Decision Making Under Uncertainty

April 28, 2014

Contents

Chapter 1

Introduction

The purpose of this book is to collect all fundamental results for decision making under uncertainty in one place, much as the book by Puterman on Markov decision processes did for Markov decision process theory. In particular, the aim is to give a unified account of algorithms and theory for sequential problems and reinforcement learning. Starting from elementary statistical decision theory, we progress to the reinforcement learning problem, its formalisation and various solution methods. The end of the book focuses on the current state of the art in approximations.

The book may be read as follows ...

The problem of decision making under uncertainty

- *Modelling* our uncertainty about the world ⇒ learning
- *Optimising* our decisions given our knowlege ⇒ planning

Applications and related problems

- Optimisation: robust decisions, efficient search, planning.
- AI: modelling, learning from interaction and/or demonstration.
- Economics: Mechanism design, behavioural modelling.
- Security: Cryptography, Biometrics, Intrusion detection and response
- Biology and Medicine: Automatic experiment design, clinical trials, congitive science.

Planning and learning are connected through the exploration-exploitation trade-off.

1.1 Exploration-exploitation

1.1.1 Introduction

The exploration-exploitation trade-off

Example 1.1.1 (Selecting a restaurant)**.** *Consider the problem of selecting the restaurant to go to.*

- *You usually go to* Les Epinards*. The food there is usually to your taste and satisfactory.*
- *You heard that* King's Arm *is really good! It is tempting to try it out. But there is a risk involved.*

- *It's Friday. Do you:*
  - *Go to* Les Epinards*?*
  - *Call* King's Arm *to reserve?*
  - *Check the menu of* King's Arm *and then decide?*

The exploration-exploitation trade-off

- Exploit knowledge about the world to gain a *known* reward.
- Explore the world to learn, *potentially* getting less or more reward.
- Arises when data collection is interactive.

Formalising decision problems

- How do our decisions depend on what we want?
- How do we weigh alternatives?
- Is there a good concept of rationality?

Beliefs, learning and planning

- How can we express belief and how does belief change?
- How might we make decisions according to our beliefs?
- What if our decisions can affect our beliefs?

Why decision theory?

- Formalising trade-offs makes problems well-posed.
- Better overall solutions could be found.
- We may ignore non-essential aspects.

Reinforcement learning is the problem of learning to act in an unknown environment, solely through interaction and some limited feedback. The learner does not necessarily have access to detailed instructions on how to perform a

task. Instead, it performs *actions*, which affect the environment and obtains some *feedback*. Sometimes the actions result in *rewards*, which represent the agent's desires. The learning problem is then formulated as the problem of learning how to act to maximise total reward.

This is a fundamental problem in artificial intelligence, since frequently we can tell robots, computers and cars only *what* we want them to do, but have no idea how we can program them. We would like to simply give them a description of our goals and then let them *explore* the environment on their own to find a good solution. Since the world is partially unknown, the learner always has to decide between two general types of decisions. It can either decide to do something which will give it some known reward, or it could take a risk and try something new. This is called the exploration-exploitation tradeoff.

Of course, animals and humans do learn in this way. Through imitation, exploration and reward signals, humans shape their behaviour to achieve their goals. In fact, it has been known since the 90s that there is some connection between some reinforcement learning algorithms and mechanisms in the basal ganglia.

Finally, there are many connections between reinforcement learning and other fields. Some algorithms used for reinforcement learning are also equivalent to fundamental algorithms in other fields. In particular, the general Bayes-optimal reinforcement learning algorithm is formally identical to Bayes-optimal algorithms for the automatic design of experiments and clinical trials. In addition, algorithms developed for more specialised problems have found application in optimisation of stochastic functions and game-theoretic problems.

Outline

* Probability refresher. Measure theory; Axiomatic definition of probability; Conditional probability; Bayes' theorem; Random variables; Expectation

1. Subjective probability and utility. The notion of subjective probability; eliciting priors; the concept of utility; expected utility

2. Decision problems. maximising expected utility; maximin utility; regret;

3. Estimation. Estimation as conditioning; families of distributions that are closed under conditioning; conjugate priors; concentration inequalities; PAC and high-probability bounds; (*) Markov Chain Monte Carlo; (*) ABC estimation

* Hypothesis testing. Bayesian hypothesis testing; point hypotheses; null hypothesis testing; unconditional and conditional hypothesis test procedures; frequentist testing; the fallacy of p-values

4. Sequential sampling and optimal stopping. Sequential sampling problems; the cost of sampling; optimal stopping; Martingales

5. Automatic experiment design and bandit problems. Belief and information state; bandit problems; Markov decision processes; backwards induction

6. Reinforcement learning I: Markov decision processes and fundamental algorithms. simple learning strategies for bandit problems; Markov decision processes; value iteration; policy iteration; policy search; model-free methods
7. Reinforcement learning II: Stochastic and approximation algorithms Q-learning; approximate value iteration; approximate policy iteration
8. Reinforcement learning III: Generalised problems. continuous case; partially observable case; (*) multi-agent case
9. Project meeting.
10. Reinforcement learning IV: Bayesian algorithms Bounds on the utility; Thompson sampling; Stochastic Branch and Bound; Sparse sampling; Rollout sampling;
11. Reinforcement learning V: Bandit algorithms and regret Tree and metric bandits; UCRL; (*) Bounds for Thompson sampling
12. Project meeting.
13. Learning with expert advice Regret; Prediction with expert advice; Prediction with incomplete information; Prediction with side-information; Connections with game theory
14. Learning by demonstration; Preference Elicitation

Assessment

Exercises and feedback: 40%

- Exercises after every unit.
- Exercise sets include feedback form.
- Necessary for a good project!

Participation: 10%

- Active participation in the course.
- Corrections on course notes.

Project: 50%
Competition, presentation and report.

- Team competition using rl-glue socket API.
- Each team codes:
  - An environment (test-bed).
  - An agent.
- Agents are evaluated on all environments.

Themes

- Models for representing belief and preferences.
- Algorithms for decision making.
- Fast optimisation.
- Applications in finance.
- Decision making in animals.
- Inferring preferences and beliefs.
- Automatic design of experiments.

Chapter 2

Probability concepts

2.1 Probability

Two notions of probability

While probability is a simple mathematical construction, philosophically it has had at least two different meanings. In the classical sense, a probability distribution is a description for a truly random event. In the subjectivist sense, probabilty is merely a description for uncertainty which may or may not be due to randomness.

Classical Probability

- A *random experiment* is performed, with a given set Ω of possible outcomes. A simple example is the 2-slit experiment in physics, where a particle is generated and which can go through either one of two slits. According to our current understanding of quantum theory, it is impossible to predict which slit the particle will go through. There, the set of possible events correspond to the particle passing through one or the other slit.
- We care about the probability that the particle will go through one of the two slots in the experiment. Does it depend on where the other particles have passed through? In the 2-slit experiment, the probabilities of either event can be actually accurately calculated. However, which slit the particle will go through is fundamentally unpredictable.

Such quantum experiments are the only ones that are currently thought of as truly random (though some people disagree about that too). Any other procedure, such as tossing a coin or casting a die, is inherently deterministic and only *appears* random due to the difficulty in predicting the outcome. That is, modelling a coin toss as a random process is usually the best approximation we can make in practice, given our uncertainty about the complex dynamics involved.

Subjective Probability

- We assume that Ω is a set of possible *worlds* or realities, This set can be quite large and include anything imaginable. For example, it may include worlds where dragons are real. However, in practice one only cares about certain aspects of the world.
- We can interpret the probability of a world in Ω as a belief that it is the true world.

In such a setting there is an actual true world $\omega^* \in \Omega$, which is simply unknown. This could have been set by Nature to an arbritrary value deterministically. The probability only reflects our lack of knowledge.

2.1.1 Sets, experiments and sample spaces

Set theory definitions

A very useful way to describe a set A is as follows

$$A \triangleq \{x \mid x \text{ have property } Y\}$$

for example

$$B(c,r) \triangleq \{x \in \mathbb{R}^n \mid \|x - c\| \leq r\}$$

describes the set of points enclosed in an n-dimensional sphere of radius r with center $c \in \mathbb{R}^n$.

- If an element x *belongs* to a set A, we write $x \in A$.
- Let the *sample space* Ω be a set such that $\omega \in \Omega$ always.
- We say that A is a *subset* of B or that B *contains* A, and write $A \subset B$, iff, $x \in B$ for any $x \in A$.
- Let $B \setminus A \triangleq \{x \mid x \in B \wedge x \notin A\}$ be the set difference.
- Let $A \triangle B \triangleq (B \setminus A) \cup (A \setminus B)$ be the symmetric set difference.
- The *complement* of any $A \subset \Omega$ is $A^{\complement} \triangleq \Omega \setminus A$.
- The *empty set* is $\emptyset = \Omega^{\complement}$.
- The *union* of n sets: $A_1, \ldots, A_n$ is $\bigcup_{i=1}^{n} A_i = A_1 \cup \cdots \cup A_n$.
- The *intersection* of n sets $A_1, \ldots, A_n$ is $\bigcap_{i=1}^{n} A_i = A_1 \cap \cdots \cap A_n$.
- A and B are *disjoint* if $A \cap B = \emptyset$.

Experiments and sample spaces

Experiments

The set of possible experimental outcomes of an experiment is called the *sample space* Ω.

- Ω must contain all possible outcomes.
- Each statistician i may consider a different Ω_i for the same experiment.

Example 2.1.1. *Experiment: give medication to a patient.*

- $\Omega_1 = \{$*Recovery within a day, No recovery after a day*$\}$.
- $\Omega_2 = \{$*The medication has side-effects, No side-effect*$\}$.
- $\Omega_3 =$ *all combinations of the above.*

Product spaces

- We perform n experiments.
- Assume that the i-th experiment has sample space Ω_i.
- The *Cartesian product* or *product space* is defined as

$$\Omega_1 \times \cdots \times \Omega_n = \{(s_1, \ldots, s_n) \mid s_i \in \Omega_i, \forall i \in \{1, \ldots, n\}\} \tag{2.1.1}$$

  the set of all ordered n-tuples $(s_1, \ldots, s_n)$.
- The sample space $\prod_{i=1}^{n} \Omega_i$ can be thought of as a sample space of a *composite* experiment in which all n experiments are performed.

Identical experiment sample spaces

- In many cases, $\Omega_i = \Omega$ for all i, i.e. the sample space is identical for all individual experiment (e.g. n coin tosses).
- We then write $\Omega^n = \prod_{i=1}^{n} \Omega$.

2.1.2 Events, measure and probability

Events and probability

Probability of a set
If A is a subset of Ω, the probability of A is a measure of the chances that the outcome of the experiment will be an element of A.

Which sets?

Ideally, we would like to be able to assign a probability to *every subset of* Ω. However, for technical reasons, this is not possible.

Example 2.1.2. *Let X be uniformly distributed on $[0, 1]$.*

- *What is the probability that X will be in $[0, 1/4)$?*
- *What is the probability that X will be in $[1/4, 1]$?*
- *What is the probability that X will be a rational number?*

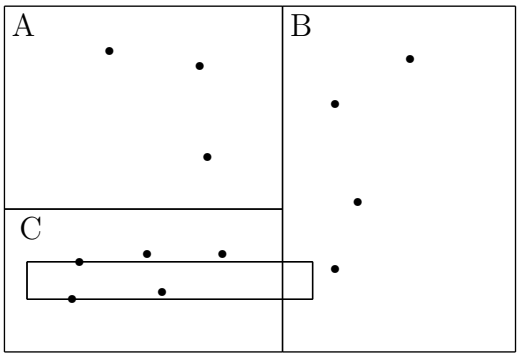


Figure 2.1: A fashionable apartment

Measure theory primer

Imagine that you have an apartment Ω composed of three rooms, A, B, C. There are some coins on the floor and a 5-meter-long red carpet. We can measure various things in this apartment.

Area

- A: $4 \times 5 = 20m^2$.
- B: $6 \times 4 = 24m^2$.
- C: $2 \times 5 = 10m^2$.

Coins on the floor

- A: 3.
- B: 4
- C: 5.

Length of red carpet

- A: $0m$
- B: $0.5m$
- C: $4.5m$.

Measure the sets: $\mathcal{F} = \{\emptyset, A, B, C, A \cup B, A \cup C, B \cup C, A \cup B \cup C\}$. It is

easy to see that the union of any sets in $\mathcal{F}$ is also in $\mathcal{F}$. In other words, $\mathcal{F}$ is closed under union. Furthermore, $\mathcal{F}$ contains the whole space Ω.

Note that all those measures have an *additive property.*

Measure and probability

As previously mentioned, the probability of $A \subset \Omega$ is a measure of the chances that the outcome of the experiment will be an element of A. Here we give a precise definition of what we mean by measure and probability.

Definition 2.1.1 (A field on Ω)**.** *A family $\mathcal{F}$ of sets, such that for each $A \in \mathcal{F}$, $A \subset \Omega$, is called a field on Ω if and only if*

1. $\Omega \in \mathcal{F}$
2. *if $A \in \mathcal{F}$, then $A^{\complement} \in \mathcal{F}$.*
3. *For any $A_1, A_2, \ldots, A_n$ such that $A_i \in \mathcal{F}$, it holds that: $\bigcup_{i=1}^{n} A_i \in \mathcal{F}$.*

From the above definition, it is easy to see that $A_i \cap A_j$ is also in the field.

Definition 2.1.2 (σ-field on Ω)**.** *A family $\mathcal{F}$ of sets, such that $\forall A \in \mathcal{F}$, $A \subset \Omega$, is called a σ-field on Ω if and only if*

1. $\Omega \in \mathcal{F}$
2. *if $A \in \mathcal{F}$, then $A^{\complement} \in \mathcal{F}$.*
3. *For any sequence $A_1, A_2, \ldots$ such that $A_i \in \mathcal{F}$, it holds that: $\bigcup_{i=1}^{\infty} A_i \in \mathcal{F}$.*

It is easy to verify that the $\mathcal{F}$ given in the apartment example satisfies these properties.

Definition 2.1.3 (Measure)**.** *A measure λ on $(\Omega, \mathcal{F})$ is a function $\lambda : \mathcal{F} \to \mathbb{R}^+$ such that*

1. $\lambda(\emptyset) = 0$.
2. $\lambda(A) \geq 0$ *for any* $A \in \mathcal{F}$.
3. *For any collection of subsets $A_1, \ldots, A_n$ with $A_i \in \mathcal{F}$ and $A_i \cap A_j = \emptyset$.*

$$\lambda\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \lambda(A_i) \tag{2.1.2}$$

It is easy to verify that the floor area, the number of coins, and the length of the red carpet are all measures. In fact, the area and length correspond to what is called a *Lebesgue measure* and the number of coins to a *counting measure.*

Definition 2.1.4 (Probability measure)**.** *A probability measure P on $(\Omega, \mathcal{F})$ is a function $P : \mathcal{F} \to [0, 1]$ such that:*

1. $P(\Omega) = 1$
2. $P(\emptyset) = 0$
3. $P(A) \geq 0$ *for any* $A \in \mathcal{F}$.

4. If $A_1, A_2, \ldots$ are disjoint then

$$P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i) \qquad \text{(union)}$$

$(S, \mathcal{F}, P)$ is called a probability space.

So, probability is just a special type of measure.

The Lebesgue measure

Definition 2.1.5 (Outer measure)**.** *Let $(\Omega, \mathcal{F}, \lambda)$ be a measure space. The outer measure of a set $A \subset \Omega$ is:*

$$\lambda^* \triangleq \inf A \subset \bigcup_k B_k \sum_k \lambda(B_k). \tag{2.1.3}$$

Definition 2.1.6 (Inner measure)**.** *Let $(\Omega, \mathcal{F}, \lambda)$ be a measure space. The outer measure of a set $A \subset \Omega$ is:*

$$\lambda_* \triangleq \lambda(\Omega) - \lambda(\Omega \setminus A). \tag{2.1.4}$$

Definition 2.1.7 (Lebesgue measurable sets)**.** *A set A is (Lebesgue) measurable if the outer and inner measures are equal.*

$$\lambda^*(A) = \lambda_*(B). \tag{2.1.5}$$

The common value of the inner and outer measure is called the Lebesgue measure[1] *$\bar{\lambda} = \lambda^*(A)$.*

2.1.3 Conditioning and independence

Independent events and conditional probability

Events correspond to sets. Thus, the probability of the event that a draw from Ω is in A is equal to the probability measure of A, $P(A)$.

Definition 2.1.8 (Independent events)**.** *Two events A, B are independent if $P(A \cap B) = P(A)P(B)$. The events in a family $\mathcal{F}$ of events are independent if for any sequence $A_1, A_2, \ldots$ of events in $\mathcal{F}$,*

$$P\left(\bigcap_{i=1}^{n} A_i\right) = \prod_{i=1}^{n} P(A_i) \qquad \text{(independence)}$$

Definition 2.1.9 (Conditional probability)**.** *The conditional probability of A when B, s.t. $P(B) > 0$, is given is:*

$$P(A \mid B) \triangleq \frac{P(A \cap B)}{P(B)}. \tag{2.1.6}$$

Of course, $P(A \cap B) = \mathbb{P}(A \mid B)\,\mathbb{P}(B)$ even if A, B are not independent.

[1] It is easy to see that $\bar{\lambda}$ is a measure.

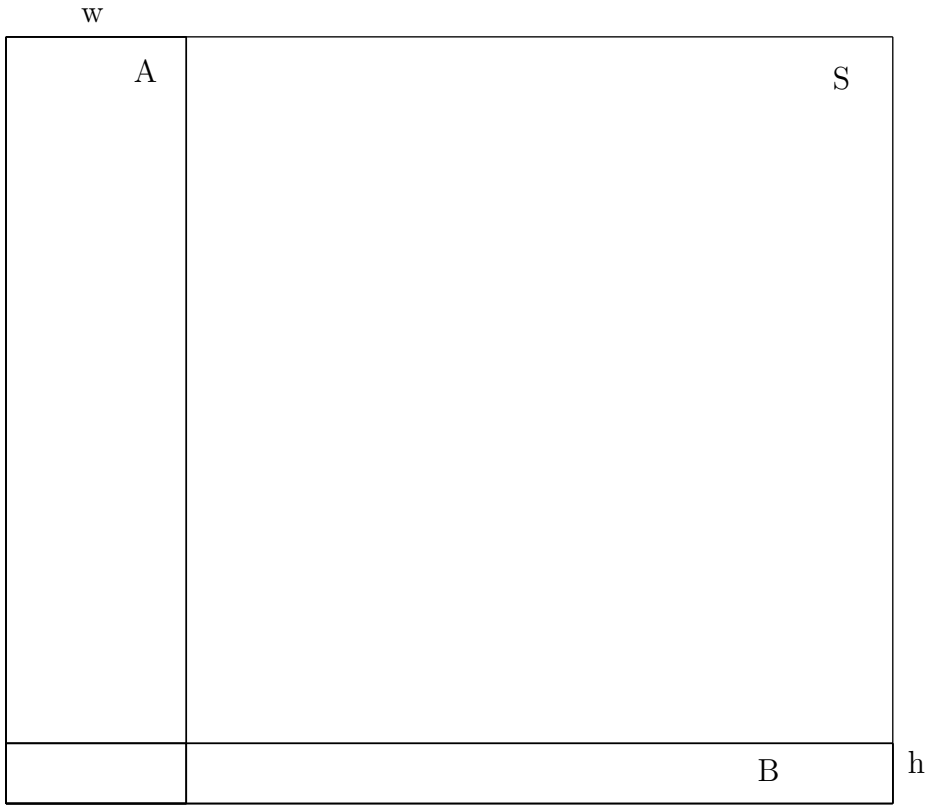


Figure 2.2: In the above case, S is a unit square and taking P to be the Lebesgue measure, we see that $P(S) = 1 \cdot 1$, $P(A) = 1 \cdot w$, $P(B) = h \cdot 1$ and $P(A \cap B) = wh$, so A and B are independent.

Bayes' theorem

The following theorem trivially follows from the above discussion. However, versions of it shall be used repeatedly throughout. For this reason we present it here together with a detailed proof.

Theorem 2.1.1 (Bayes' theorem)**.** *Let $A_1, A_2, \ldots$ be a (possibly infinite) sequence of disjoint events such that $\bigcup_{i=1}^{\infty} A_i = \Omega$ and $P(A_i) > 0$ for all i. Let B be another event with $P(B) > 0$. Then*

$$P(A_i \mid B) = \frac{P(B \mid A_i)P(A_i)}{\sum_{j=1}^{\infty} P(B \mid A_j)P(A_j)} \tag{2.1.7}$$

Proof. From (2.1.6), $P(A_i \mid B) = P(A_i \cap B)/P(B)$ and also $P(A_i \cap B) = P(B \mid A_i)P(A_i)$. Thus

$$P(A_i \mid B) = \frac{P(B \mid A_i)P(A_i)}{P(B)},$$

and we continue analyzing the denominator $P(B)$. First, due to $\bigcup_{i=1}^{\infty} A_i = \Omega$ we have $B = \bigcup_{j=1}^{\infty}(B \cap A_j)$. Since A_i are disjoint, so are $B \cap A_i$. Then from the union property of probability distributions we have

$$P(B) = P\left(\bigcup_{j=1}^{\infty}(B \cap A_j)\right) = \sum_{j=1}^{\infty} P(B \cap A_j) = \sum_{j=1}^{\infty} P(B \mid A_j)P(A_j),$$

which finishes the proof. □

Binomial coefficients

Binomial coefficients appear in a lot of different distributions. They are especially useful for combinatorial problems.

$$\binom{x}{n} \triangleq \frac{\prod_{i=0}^{n-1}(x-i)}{n!}, \qquad x \in \mathbb{R}, n \in \mathbb{N}, \tag{2.1.8}$$

and $\binom{x}{0} = 1$. It follows that

$$\binom{k}{n} = \frac{k!}{n!(k-n)!} \qquad k, n \in \mathbb{N}, k \geq n. \tag{2.1.9}$$

2.2 Random variables

Random variables

A random variable X is a special kind of random quantity, defined as a real function of outcomes in Ω. Thus, it also defines a mapping from a probability measure P on $(\Omega, \mathcal{F})$ to a probability measure P_X on $(\mathbb{R}, \mathcal{B}(\mathbb{R}))$. More precisely, we define the following.

Definition 2.2.1 (Measurable function)**.** *Let $\mathcal{F}$ on Ω be a σ-field. A function $g : \Omega \to \mathbb{R}$ is said to be measurable with respect to $\mathcal{F}$, or $\mathcal{F}$-measurable, if, for any $x \in \mathbb{R}$,*

$$\{s \in \Omega \mid g(s) \leq x\} \in \mathcal{F}.$$

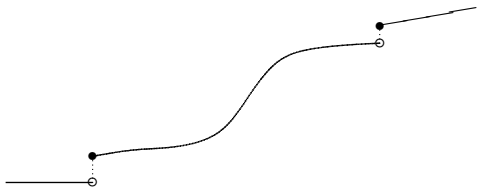


Figure 2.3: A distribution function F

Definition 2.2.2 (Random variable)**.** *Let $(\Omega, \mathcal{F}, P)$ be a probability space. A random variable $X : \Omega \to \mathbb{R}$ is a real-valued, $\mathcal{F}$-measurable function.*

***The distribution of** X*

Every random variable X induces a probability measure P_X on $\mathbb{R}$. For any $B \subset \mathbb{R}$ we define

$$P_X(B) = \mathbb{P}(X \in B) = P(\{s \mid X(s) \in B\}). \tag{2.2.1}$$

Thus, the probability that X is in B is equal to the P-measure of the points $s \in \Omega$ such that $X(s) \in B$ and also equal to the P_X-measure of B.

Here $\mathbb{P}$ is used as a *short-hand* notation.

Exercise 1. *Ω is the set of 52 playing cards. $X(s)$ is the value of each card $(1, 10$ for the ace and figures respectively). What is the probability of drawing a card s with $X(s) > 7$?*

(Cumulative) Distribution functions

Definition 2.2.3 ((Cumulative) Distribution function)**.** *The distribution function of a random variable X is the function $F : \mathbb{R} \to \mathbb{R}$:*

$$F(t) = \mathbb{P}(X \leq t). \tag{2.2.2}$$

Properties

- If $x \leq y$, then $F(x) \leq F(y)$.
- F is right-continuous.
- At the limit,
$$\lim_{t \to -\infty} F(t) = 0, \qquad \lim_{t \to \infty} F(t) = 1.$$

2.2.1 Discrete and continuous random variables

Types of distributions

On the real line, there are two types of distributions for a random variable. Here, once more, we employ the $\mathbb{P}$ notation as a shorthand for the probability of general events involving random variables, so that we don't have to deal with the measure notation. The two following examples should give some intuition.

Discrete distributions

$X : \Omega \to \{x_1, \ldots, x_n\}$ takes n discrete values (n can be infinite). The probability function of X is

$$f(x) \triangleq \mathbb{P}(X = x),$$

defined for $x \in \{x_1, \ldots, x_n\}$. For any $B \subset \mathbb{R}$:

$$P_X(B) = \sum_{x_i \in B} f(x_i).$$

In addition, we write $\mathbb{P}(X \in B)$ to mean $P_X(B)$.

Continuous distributions

X has a continuous distribution if there exists a *probability density function* f s.t. $\forall B \subset \mathbb{R}$:

$$P_X(B) = \int_B f(x)\,\mathrm{d}x.$$

It is possible that X has neither a continuous, nor a discrete distribution.

2.2.2 Random vectors

Generalisation to $\mathbb{R}^m$

We can generalise to random *vectors* in a Euclidean space. Once more, there are two special cases of distributions for the random vector $X = (X_1, \ldots, X_n)$.

Discrete distributions

$$\mathbb{P}(X_1 = x_1, \ldots, X_m = x_m) = f(x_1, \ldots, x_m)$$

Continuous distributions

For $B \subset \mathbb{R}^m$

$$\mathbb{P}\{(X_1, \ldots, X_m) \in B\} = \int_B f(x_1, \ldots, x_m)\,\mathrm{d}x_1 \cdots \mathrm{d}x_m$$

Measure-theoretic notation

The previously seen special cases can be handled with a unified notation if we take advantage of the fact that probability is only a particular type of measure. As a first step, we note that summation can also be seen as integration with respect to the counting measure and that Riemann integration is integration with respect to the Lebesgue measure.

Integral with respect to a measure μ

Introduce the common notation $\int \cdots \mathrm{d}\mu(x)$, where μ is a measure. Let some real function $g : \Omega \to \mathbb{R}$. Then for any subset $B \subset \Omega$ we can write

- Discrete case: f is the probability function and we choose the *counting measure* for μ, so:

$$\sum_{x \in B} g(x) f(x) = \int_B g(x) f(x) \, \mathrm{d}\mu(x)$$

  Roughly speaking, the counting measure $\mu(\Omega)$ is equal to the number of elements in Ω.

- Continuous case: f is the probability density function and we choose the *Lebesgue measure* for μ, so:

$$\int_B g(x) f(x) \, \mathrm{d}x = \int_B g(x) f(x) \, \mathrm{d}\mu(x)$$

  Roughly speaking, the Lebesgue measure $\mu(S)$ is equal to the volume of S.

In fact, since probability is a measure in itself, we do not need to complicate things by using f and μ at the same time! This allows us to use the following notation.

Lebesgue-Stiletjes notation

If P is a probability measure on $(\Omega, \mathcal{F})$ and $B \subset \Omega$, and g is $\mathcal{F}$-measurable, we write the probability that $g(x)$ takes the value B can be written equivalently as:

$$\mathbb{P}(g \in B) = P_g(B) = \int_B g(x) \, \mathrm{d}P(x) = \int_B g \, \mathrm{d}P. \tag{2.2.3}$$

Intuitively, $\mathrm{d}P$ is related to densities in the following way. If P is a measure on Ω and is absolutely continuous with respect to another measure μ, then $p \triangleq \frac{\mathrm{d}P}{\mathrm{d}\mu}$ is the (Radon-Nikodyn) derivative of P with respect to μ. We write the integral as $\int gp \, \mathrm{d}\mu$. If μ is the Lebesgue measure, then p coincides with the probability density function.

Marginal distributions and independence

Although this is a straightforward outcome of the set-theoretic definition of probability, we also define the marginal explicitly for random vectors.

Marginal distribution

The marginal distribution of $X_1, \ldots, X_k$ from a set of variables $X_1, \ldots, X_m$, is

$$\mathbb{P}(X_1, \ldots, X_k) \triangleq \int \mathbb{P}(X_1, \ldots, X_k, X_{k+1} = x_{k+1}, \ldots, X_m = x_m) \, \mathrm{d}\mu(x_{k+1}, \ldots, x_m). \tag{2.2.4}$$

In the above, $\mathbb{P}(X_1, \ldots X_k)$ can be thought of as the probability measure for any events related to the random vector $(X_1, \ldots, X_k)$. Thus, it defines a probability measure over $\left(\mathbb{R}^k, \mathfrak{B}\left(\mathbb{R}^k\right)\right)$. In fact, let $Y = (X_1, \ldots, X_k)$ and $Z = (X_{k+1}, \ldots, X_m)$ for simplicity. Then define $Q(A) \triangleq \mathbb{P}(Z \in A)$, with $A \subset \mathbb{R}^{m-k-1}$. Then the above can be re-written as:

$$\mathbb{P}(Y \in B) = \int_{\mathbb{R}^{m-k-1}} \mathbb{P}(Y \in B \mid Z = z) \, \mathrm{d}Q(z).$$

Similarly, $\mathbb{P}(Y \mid Z = z)$ can be thought of as a function mapping from values of Z to probability measures. Let $P_z(B) \triangleq \mathbb{P}(Y \in B \mid Z = z)$ be this measure corresponding to a particular value of z. Then we can write

$$\mathbb{P}(Y \in B) = \int_{\mathbb{R}^{m-k-1}} \left(\int_B \mathrm{d}P_z(y) \right) \mathrm{d}Q(z).$$

Independence

If X_i is independent of X_j for all $i \neq j$:

$$\mathbb{P}(X_1, \ldots, X_m) = \prod_{i=1}^{M} \mathbb{P}(X_i), \qquad f(x_1, \ldots, x_m) = \prod_{i=1}^{M} g_i(x_i) \tag{2.2.5}$$

2.2.3 Moments

There are some simple properties of the random variable under consideration which are frequently of interest in statistics. Two of those properties are *expectation* and *variance*.

Expectation

Definition 2.2.4. *The expectation $\mathbb{E}(X)$ of any random variable $X : \Omega \to R$,where R is a vector space, with distribution P_X is defined by*

$$\mathbb{E}(X) \triangleq \int_R t \, \mathrm{d}P_X(t), \tag{2.2.6}$$

as long as the integral exists.

Furthermore,

$$\mathbb{E}[g(X)] = \int g(t)\,\mathrm{d}P_X(t),$$

for any function g.

Variance

Definition 2.2.5. *The variance $\mathbb{V}(X)$ of any random variable $X : \Omega \to \mathbb{R}$ with distribution P_X is defined by*

$$\begin{aligned}\mathbb{V}(X) &\triangleq \int_{-\infty}^{\infty} [t - \mathbb{E}(X)]^2 \,\mathrm{d}P_X(t) \\ &= \mathbb{E}\left\{[X - \mathbb{E}(X)]^2\right\} \\ &= \mathbb{E}(X^2) - \mathbb{E}^2(X).\end{aligned} \tag{2.2.7}$$

When $X : \Omega \to R$ with R an arbitrary vector space, the above becomes the *covariance matrix*:

$$\begin{aligned}\mathbb{V}(X) &\triangleq \int_{-\infty}^{\infty} [t - \mathbb{E}(X)]\,[t - \mathbb{E}(X)]^\top \,\mathrm{d}P_X(t) \\ &= \mathbb{E}\left\{[X - \mathbb{E}(X)]\,[X - \mathbb{E}(X)]^\top\right\} \\ &= \mathbb{E}(XX^\top) - \mathbb{E}(X)\,\mathbb{E}(X)^\top.\end{aligned} \tag{2.2.8}$$

Divergences

One useful idea is KL-divergences on measures.

Definition 2.2.6. *KL-Divergence*

$$D\left(P \parallel Q\right) \triangleq \int \frac{\mathrm{d}P}{\mathrm{d}Q}\,\mathrm{d}P. \tag{2.2.9}$$

Empirical distributions

Definition 2.2.7. *Let $x^n = (x_1, \ldots, x_n)$ drawn from a product measure $x^n \sim P^n$ on the measurable space $(\mathcal{X}^n, \mathfrak{F}_n)$. Let $\mathfrak{S}$ be any σ-field on $\mathcal{X}$. Then empirical distribution of x^n is defined as*

$$\hat{P}_n(B) \triangleq \frac{1}{n}\sum_{t=1}^{n} \mathbb{I}\left\{x_t \in B\right\}. \tag{2.2.10}$$

2.3 Conclusion

Recommended further reading

Most of this material is based on DeGroot [1970]. See Kolmogorov and Fomin [1999] for a really clear exposition of measure, starting from rectangle areas (developed from course notes in 1957). Also see Savage [1972] for a verbose, but interesting and rigorous introduction to subjective probability. More technical books, such as Ash and Doleéans-Dade [2000] are not very approachable by non-math graduates.

Summary

- *Sample space* Ω contains all possible *outcomes* of an experiment.
- σ-field $\mathcal{F}$ s.t. $\forall A, B \in \mathcal{F}$, $A \subset \Omega$, $A \cup B \in \mathcal{F}$, $\Omega \in \mathcal{F}$.
- Measurable space $(\Omega, \mathcal{F})$, measure space $(\Omega, \mathcal{F}, \mu)$.
- *Measure* $\mu : \mathcal{F} \to \mathbb{R}$ such that $\mu(\emptyset) = 0$, and $\mu(A_i) \geq 0$ for any $A_i \in \mathcal{F}$. For *disjoint* A_i, $\mu\left(\bigcup_i A_i\right) = \sum_i \mu(A_i)$.
- Probability space $(\Omega, \mathcal{F}, P)$, with *probability measure* P such that $P(\Omega) = 1$.
- *Probability* that $x \in A$:

$$\mathbb{P}(x \in A) \triangleq P(A) = \int_A \mathrm{d}P(t), \qquad A \subset \Omega$$

- *Expectation* of $X : \Omega \to Z$

$$\mathbb{E}(X) \triangleq \int_\Omega X(t)\,\mathrm{d}P(t) = \int_Z u\,\mathrm{d}P_X(u)$$

- *Conditional probability*

$$\mathbb{P}(A \mid B) = \frac{\mathbb{P}(A, B)}{\mathbb{P}(B)}, \quad P(A \mid B) = \frac{P(A \cap B)}{P(B)},$$

- *Marginal* distribution

$$\mathbb{P}(B) = \sum_i \mathbb{P}(B, A = i), \quad \sum_i \mathbb{P}(A = i) = 1,$$

$$P(B) = \sum_i P(B \cap A_i), \quad \bigcup_i A_i = \Omega.$$

- If A, B are *independent*

$$\mathbb{P}(A, B) = \mathbb{P}(A)\,\mathbb{P}(B), \quad P(A \cap B) = P(A)P(B).$$

Chapter 3

Subjective probability and utility

3.1 Subjective probability

In order to make decisions, we need to be able to make predictions about the possible outcomes of each decision. Usually, we have *uncertainty* about what those outcomes are. This can be due to *stochasticity*, which is frequently used to model games of chance and inherently unpredictable physical phenomena. It can also be due to *partial information*, a characteristic of many natural problems. For example, it might be hard to guess at any one moment how much change you have in your wallet, whether you will be able to catch the next bus, or to remember where you left your keys.

In either case, this uncertainty can be expressed as a *subjective belief*. This does not have to correspond to reality. For example, some people believe, quite inaccurately, that if a coin comes up tails for a long time, it is quite likely to come up heads very soon. Or, you might quite happily believe your keys are in your pocket, only to realise that you left them at home as soon you arrive at the office.

In this book, we assume the view that subjective beliefs can be modelled as *probabilities*. This allows us to treat uncertainty due to stochasticity and due to partial information in a unified framework. In doing so, we shall treat each part of the problem as specifying a space of possible outcomes. What we wish to do is to find a *consistent way* of defining probabilities in the space of outcomes.

3.1.1 Relative likelihood

Let us consider the simple example of guessing whether a tossed coin will come up head, or tails. Let $\mathcal{S}$ be the sample space, and let $A \subset \mathcal{S}$ be the set of tosses where the coin comes up heads, and $B \subset \mathcal{S}$ be the set of tosses where it comes up tails. Here $A \cap B = \emptyset$, but there may be some other events such as the coin becoming lost, so it does not necessarily hold that $A \cup B = \mathcal{S}$. Nevertheless, we only care about whether A is more likely to occur than B. We can express that via the concept of relative likelihood:

Definition 3.1.1. *(The relative likelihood of two events A and B)*

- *If one thinks that A is more likely than B, then we write $A \succ B$, or equivalently $B \prec A$.*
- *If one thinks A is as likely as B, then we write $A \asymp B$.*

We also use $\succsim$ and $\precsim$ for at least as likely as and for no more likely than.

Let us now speak more generally about the case where we have defined an appropriate σ-field $\mathcal{F}$ on $\mathcal{S}$. Then each element $A_i \in \mathcal{F}$ will be a subset of $\mathcal{S}$. Furthermore, we have defined a relative likelihood relation for all elements $A_i \in \mathcal{F}$.[1]

As we would like to use the language of probability to talk about likelihoods, we would like to be able to define a probability measure that agrees with our given relations.

[1] More formally, we can define three classes: $C_\succ, C_\prec, C_\asymp \subset \mathcal{F}^2$ such that a pair $(A_i, A_j) \in C_R$ if an only if it satisfies the relation $A_i R A_j$, where $R \in \{\succ, \prec, \asymp\}$. It is easy to see that the three classes form a partition of $\mathcal{F}^2$.

A probability measure $P : \mathcal{F} \to [0, 1]$ is said to *agree* with a relation $A \precsim B$, if it has the property that: $P(A) \leq P(B)$ if and only if $A \precsim B$, for all $A, B \in \mathcal{F}$.

Of course, there are many possible measures that can agree with a given relation. It could even be that a given relational structure is incompatible with any possible probability measure. For that reason, we shall have to make some assumptions about relative likelihoods of events.

3.1.2 Subjective probability assumptions

Our beliefs must be *consistent*. This can be achieved if they satisfy some assumptions. First of all, it must always be possible to say whether one event is more likely than the other. Consequently, we are not allowed to claim ignorance.

Assumption 3.1.1 (SP1)**.** *For any pair of events $A, B \in \mathcal{F}$, one of the following must hold: Either $A \succ B$, $A \prec B$, or $A \eqsim B$.*

If we can partition A, B in such a way that each part of A is less likely than its counterpart in B, then A is less likely than B. For example, let A_1 be the event that it rains tomorrow morning and A_2 the event that does not. Let B_1, B_2 be the corresponding events for the afternoon. If it is more likely that it rains than it does not, both in the morning and the afternoon, then it is more likely that it rains sometime in the day that not. This is formalised by the following assumption.

Assumption 3.1.2 (SP2)**.** *Let $A = A_1 \cup A_2$, $B = B_1 \cup B_2$ with $A_1 \cap A_2 = B_1 \cap B_2 = \emptyset$. If $A_i \precsim B_i$ for $i = 1, 2$ then $A \precsim B$.*

We also require the simple technical assumption that any event $A \in \mathcal{F}$ is at least as likely as the empty event $\emptyset$, which never happens.

Assumption 3.1.3 (SP3)**.** *If $\mathcal{S}$ is the certain event, and $\emptyset = \mathcal{S}^{\complement}$ the empty set, then: $\emptyset \precsim A$ and $\emptyset \prec S$.*

As it turns out, these assumptions are sufficient for proving the following theorems DeGroot [1970]. The first theorem tells us that our belief must be consistent with respect to transitivity.

Theorem 3.1.1 (Transitivity)**.** *For all events A, B, D, if $A \precsim B$ and $B \precsim D$, then $A \precsim D$.*

The second theorem says that if two events have a certain relation, then their negations have the converse relation.

Theorem 3.1.2 (Complement)**.** *For any A, B: $A \precsim B$ iff $A^{\complement} \succ B^{\complement}$.*

Finally, note that if $A \subset B$, then it must be the case that whenever A happens, B must happen and hence B must be at least as likely as A. This is demonstrated in the following theorem.

Theorem 3.1.3 (Fundamental property of relative likelihoods)**.** *If $A \subset B$ then $A \precsim B$. Furthermore, $\emptyset \precsim A \precsim S$ for any event A.*

Since we are dealing with σ-fields, we need to introduce properties for infinite sequences of events. While these are not necessary if the field $\mathcal{F}$ is finite, it is good to include them for generality.

Assumption 3.1.4 (SP4)**.** *If $A_1 \supset A_2 \supset \cdots$ is a decreasing sequence of events in $\mathcal{F}$ and $B \in \mathcal{F}$ is such that $A_i \succsim B$ for all i, then $\bigcap_{i=1}^{\infty} A_i \succsim B$.*

As a consequence, we obtain the following dual theorem:

Theorem 3.1.4. *If $A_1 \subset A_2 \subset \cdots$ is an increasing sequence of events in $\mathcal{F}$ and $B \in \mathcal{F}$ is such that $A_i \precsim B$ for all i, then $\bigcup_{i=1}^{\infty} A_i \precsim B$.*

We are now able to state a theorem for the unions of infinite sequences of disjoint events.

Theorem 3.1.5. *If $(A_i)_{i=1}^{\infty}$ and $(B_i)_{i=1}^{\infty}$ are infinite sequences of disjoint events in $\mathcal{F}$ such that $A_i \precsim B_i$ for all i, then $\bigcup_{i=1}^{\infty} A_i \precsim \bigcup_{i=1}^{\infty} B_i$.*

Exercise 2. *Here we prove that a probability measure P always satisfies the stipulated assumptions.*

(i) For any events $P(A) > P(B)$, $P(A) < P(B)$ or $P(A) = P(B)$.

(ii) If A_i, B_i are partitions of A, B, $\forall i P(A_i) \leq P(B_i) \Rightarrow P(A) \leq P(B)$.

(iii) For any A, $P(\emptyset) \leq P(A)$ and $P(\emptyset) < P(\mathcal{S})$

Solution. Part (i) is trivial, as $P : \mathcal{F} \to [0,1]$. Part (ii) follows from $P(A) = P(\bigcup_i A_i) = \sum_i P(A_i) \leq \sum_i P(B_i) = P(B)$. Part (iiI) $P(\emptyset) = 0$, $P(A) \geq 0$. Also, $P(\mathcal{S}) = 1$. □

3.1.3 Assigning unique probabilities

In many cases, and particularly when $\mathcal{F}$ is a finite field, there is a large number of probability distributions agreeing with our relative likelihoods.

How can we assign probabilities to events in an unambiguous manner?

Example 3.1.1. *Consider $\mathcal{F} = \left\{\emptyset, A, A^{\complement}, \mathcal{S}\right\}$ and say $A \succ A^{\complement}$. Consequently, $P(A) > 1/2$. But this is insufficient for assigning a specific value to $P(A)$.*

Let A be an interval on the real line, with length $\lambda(A)$.

Definition 3.1.2 (Uniform distribution)**.** *$x : \mathcal{S} \to [0,1]$ has a uniform distribution on $[0,1]$ if, for any subintervals A, B of $[0,1]$,*

$$(x \in A) \precsim (x \in B) \quad \textit{iff} \quad \lambda(A) \leq \lambda(B)$$

This means that *any* larger interval is more likely than *any* smaller interval. Now we shall connect the uniform distribution to the original sample space $\mathcal{S}$ by assuming that there is some function with uniform distribution.

Assumption 3.1.5 (SP5)**.** *There exists a random variable $x : \mathcal{S} \to [0,1]$ with a uniform distribution in $[0,1]$.*

Constructing the probability distribution

We can now use the uniform distribution to create a unique probability measure that agrees with our likelihood relation. First, we have to map each event in $\mathcal{S}$ to an equivalent event in $[0, 1]$.

Theorem 3.1.6 (Equivalent event)**.** *For any event $A \in \mathcal{F}$, there exists some $\alpha \in [0, 1]$ such that $A \approx (X \in [0, \alpha])$.*

This means that we can now define the probability of an event A by matching it to a specific equivalent event on $[0, 1]$.

Definition 3.1.3 (The probability of A)**.** *Given any event A, define $P(A)$ to be the α with $A \approx (X \in [0, \alpha])$.*

Hence

$$A \approx (X \in [0, P(A)]).$$

The above is sufficient to show the following theorem.

Theorem 3.1.7 (Relative likelihood and probability)**.** *If assumptions SP1-SP5 are satisfied, then the probability measure P defined above is unique. Furthermore, for any two events A, B, $A \precsim B$ iff $P(A) \leq P(B)$.*

3.1.4 Conditional likelihoods

Conditional likelihood

So far we have only considered the problem of forming opinions about which events are more likely *a priori*. However, we also need to have a way to incorporate evidence which may adjust our opinions. For example, while we ordinarily may think that $A \precsim B$, we may have additional information D, given which we think the opposite is true. We can formalise this through the notion of conditional likelihoods.

Example 3.1.2. *Say that A is the event that it rains in Gothenburg, Sweden tomorrow. We know that Gothenburg is quite rainy due to its oceanic climate, so we set $A \succsim A^{\complement}$. Now, let us try and incorporate some additional information. Let D denote the fact that good weather is forecast. I personally believe that $(A \mid D) \precsim (A^{\complement} \mid D)$, i.e. that good weather is more probable than rain, given the evidence of the weather forecast.*

Conditional likelihoods

Define $(A \mid D) \precsim (B \mid D)$ to mean that B is at least as likely as A when it is known that D has occurred.

Assumption 3.1.6 (CP)**.** *For any events A, B, D,*

$$(A \mid D) \precsim (B \mid D) \quad \textit{iff} \quad A \cap D \precsim B \cap D.$$

Theorem 3.1.8. *If a relation $\precsim$ satisfies assumptions SP1 to SP5 and CP, then P is the unique probability distribution such that:*

For any A, B, D such that $P(D) > 0$,

$$(A \mid D) \precsim (B \mid D) \quad \textit{iff} \quad P(A \mid D) \leq P(B \mid D)$$

Definition 3.1.4 (Conditional probability)**.**

$$P(A \mid D) = \frac{P(A \cap D)}{P(D)} \tag{3.1.1}$$

3.1.5 Probability elicitation

Probability elicitation is the problem of quantifying the subjective probabilities that a particular individual uses. One of the simplest, and most direct, methods, is to simply ask. However, because we cannot simply ask somebody to completely specify a probability distribution, we can ask for this distribution iteratively.

Example 3.1.3 (Temperature prediction)**.** *Let τ be the temperature tomorrow at noon in Gothenburg. What are your estimates?*

Eliciting the prior / forming the subjective probability measure P

- Select temperature x_0 s.t. $(\tau \leq x_0) \eqsim (\tau > x_0)$.
- Select temperature x_1 s.t. $(\tau \leq x_1 \mid \tau \leq x_0) \eqsim (\tau > x_1 \mid \tau \leq x_0)$.

Note that, necessarily, $P(\tau \leq x_0) = P(\tau > x_0) = p_0$. Since $P(\tau \leq x_0) + P(\tau > x_0) = P(\tau \leq x_0 \cup \tau > x_0) = P(\tau \in \mathbb{R}) = 1$, it follows that $p_0 = 1/2$. Similarly, $P(\tau \leq x_1 \mid \tau \leq x_0) = P(\tau > x_1 \mid \tau \leq x_0) = 1/4$.

Updating beliefs

Although we always start with a particular belief, this belief must be adjusted when we receive new evidence. In probabilistic inference, the updated beliefs are simply the probability of future events conditioned on observed events. This idea is captured neatly by Bayes' theorem, which links together the prior probability of events $P(A_i)$ with their posterior probability $P(A_i \mid B)$ given some event B and the probability $P(B \mid A_i)$ of observing the evidence B given that hypothesis A_i is true.

Theorem 3.1.9 (Bayes' theorem)**.** *Let $A_1, A_2, \ldots$ be a (possibly infinite) sequence of disjoint events such that $\bigcup_{i=1}^{n} A_i = \mathcal{S}$ and $P(A_i) > 0$ for all i. Let B be another event with $P(B) > 0$. Then*

$$P(A_i \mid B) = \frac{P(B \mid A_i)P(A_i)}{\sum_{j=1}^{n} P(B \mid A_j)P(A_j)} \tag{3.1.2}$$

Proof. By definition, $P(A_i \mid B) = P(A_i \cap B)/P(B)$, and $P(A_i \cap B) = P(B \mid A_i)P(A_i)$, so:

$$P(A_i \mid B) = \frac{P(B \mid A_i)P(A_i)}{P(B)}, \tag{3.1.3}$$

As $\bigcup_{i=1}^{n} A_i = \mathcal{S}$, we have $B = \bigcup_{j=1}^{n}(B \cap A_j)$. Since A_i are disjoint, so are $B \cap A_i$. As P is a probability, the union property and an application of 3.1.3 gives

$$P(B) = P\left(\bigcup_{j=1}^{n}(B \cap A_j)\right) = \sum_{j=1}^{n} P(B \cap A_j) = \sum_{j=1}^{n} P(B \mid A_j)P(A_j).$$

□

A simple exercise in updating beliefs

The area of Germany
Form a subjective probability for the area a of Germany in km^2.

A_1 : $a < 10^5$ km^2

A_2 : $a \in [10^5, 2.5 \cdot 10^5)$ km^2

A_3 : $a \in [2.5 \cdot 10^5, 5 \cdot 10^5)$ km^2

A_4 : $a \in [5 \cdot 10^5, 10^6)$ km^2

A_5 : $a \geq 10^6$ km^2.

Choose $P(A_i)$ for all i.

Additional information

- The EU's largest country is France ($6.7 \cdot 10^5$km^2) and the smallest is Malta with 316km^2.
- Germany is the 4th largest of the 27 EU states
- UK ($2.4 \cdot 10^5$km^2) is the 8th largest EU state

The correct answer is A_3, since $a = 3.57 \cdot 10^5$

3.2 Utility theory

While probability can be used to describe how likely an event is, utility can be used to describe how desirable it is. More concretely, our subjective probabilities are numerical representations of our beliefs and information. They can be taken to represent our "internal model" of the world. By analogy, our utilities are numerical representations of our tastes and preferences. Even if they are not directly known to us, we assume that our actions are such that we act so as to obtain maximum utility, in some sense.

3.2.1 Rewards and preferences

Rewards

Consider that we have to choose a *reward* r from a set R of possible rewards. While the elements of R may be abritrary, we shall in general find that we prefer some rewards to others. In fact, some elements of R may not even be desirable. As an example, R might be a set of tickets to different musical events, or a set of financial commodities.

Preferences

Example 3.2.1 (Musical event tickets)**.** *We have a set of tickets R, and we must choose the ticket $r \in R$ we prefer best.*

- *Case 1: R are tickets to different music events at the same time, at equally good halls with equally good seats and the same price. Here preferences simply coincide with the preferences for a certain type of music or an artist.*
- *Case 2: R are tickets to different events at different times, at different quality halls with different quality seats and different prices. Here, preferences may depend on all the factors.*

Example 3.2.2 (Route selection)**.** *We have a set of alternate routes and must pick one.*

- *R contains two routes, one short and one long, of the same quality.*
- *R contains two routes, one short and one long, but the long route is more scenic.*

Preferences among rewards

We will treat preferences in a similar manner as we have treated probabilities. That is, we will define a linear ordering among possible rewards.

Let $a, b \in R$ be two rewards. When we *prefer* a to b, we write $a \succ^* b$. Conversely, when we like a *less* than b we write $a \prec^* b$. If we like a *as much* as b, we write $a \asymp^* b$. We also use $\succsim^*$ and $\precsim^*$ for *I like at least as much as* and for *I don't like any more than*, respectively.

Assumption 3.2.1. *We make the following assumptions about the preference relations.*

(i) For any $a, b \in R$, one of the following holds: $a \succ^* b$, $a \prec^* b$, $a \asymp^* b$.

(ii) If $a, b, c \in R$ are such that $a \precsim^* b$ and $b \precsim^* c$, then $a \precsim^* c$.

The first assumption means that we must always be able to decide between any two rewards. It may seem that it does not always hold in practice, since humans are frequently indecisive. This could be attributed to the difficulty of computing the value of the relation **?**. Consider an algorithm that, taking r, r' as input returns $R \in \{\succ^*, \prec^*, \asymp^*\}$. If the algorithm does not halt, then the preference relation is not defined.

The second assumption is a bit stronger. It is in fact possible to create a preference relation that will be cyclic.

Example 3.2.3 (Counter example for transitive preferences)**.** *Consider for example vector rewards in $\mathcal{R} = \mathbb{R}^2$. Given rewards $r_i = (a_i, b_i)$, our preference relation is:*

- *$r_i \succ^* r_j$ if $a_i \geq a_j$ and $|b_i - b_j| < \epsilon$*
- *$r_i \succ^* r_j$ if $b_i \geq b_j + \epsilon'$.*

This may correspond for example to an employer deciding to hire one of two employees, i, j, depending on their experience (a) or their school grades (b). Since grades are not very reliable, if two people have grades, then we prefer the one with the most experience. However, that may lead to a cycle. Consider a sequence of candidates $i = 1, \ldots, n$, such that each candidate satisfies $b_i = b_{i+1} + \delta$, with $\delta < \epsilon$ and $a_i > a_{i+1}$. Then clearly, we must always prefer r_i to r_{i+1}. However, if $\delta n > \epsilon$, we will prefer r_n to r_1.

3.2.2 Preferences among distributions

When we cannot select rewards directly

In most problems, we cannot choose the rewards directly. Rather, we must make some decision, and then obtain a reward depending on this decision. Since we may be uncertain about the outcome of a decision, we can specify our uncertainty regarding the rewards obtained by a decision in terms of a probability distribution.

Example 3.2.4 (Route selection).

- *Each reward $r \in R$ is the time it takes to travel from A to B.*
- *We prefer shorter times.*
- *There are two routes, P_1, P_2.*
- *Route P_1 takes 10 minutes when the road is clear, but 30 minutes when the traffic is heavy. The probability of heavy traffic on P_1 is q_1.*
- *Route P_2 takes 15 minutes when the road is clear, but 25 minutes when the traffic is heavy. The probability of heavy traffic on P_2 is q_2.*

Preferences among probability distributions

Consequently, we have to define preferences between probability distributions, rather than rewards. We use the same notation as before. Let P_1, P_2 be two distributions on $(R, \mathcal{F}_R)$. If we *prefer* P_1 to P_2, we write $P_1 \succ^* P_2$. If w like P_1 *less* than P_2, write $P_1 \prec^* P_2$. If we like P_1 *as much* as P_2, we write $P_1 \asymp^* P_2$. Finally, we also use $\succsim^*$ and $\precsim^*$ do denote strict preference relations.

3.2.3 Utility

The concept of utility allows us to create a unifying framework, such that given a particular set of rewards and probability distributions on them, we can define preferences among distributions automatically. The first step is to define utility as a way to define a preference relation among rewards.

Definition 3.2.1 (Utility). *The utility is a function $U : R \to \mathbb{R}$, such that for all $a, b \in R$*

$$a \succsim^* b \quad \textit{iff} \quad U(a) \geq U(b), \tag{3.2.1}$$

The above definition is very similar to how we defined relative likelihood in terms of probability. For a given utility function, its expectation for a distribution over rewards as:

Definition 3.2.2 (Expected utility)**.** *The expected utility of a distribution P on R is:*

$$\mathbb{E}_P(U) = \int_R U(r)\,\mathrm{d}P(r) \tag{3.2.2}$$

Finally, we make the assumption that the utility function is such that the expected utility remains consistent with the preference relations between all probability distributions we are choosing between.

Assumption 3.2.2. *The expected utility hypothesis The utility of P is equal to the expected utility of the reward under P. Consequently,*

$$P \succsim^* Q \quad \textit{iff} \quad \mathbb{E}_P(U) \geq \mathbb{E}_Q(U). \tag{3.2.3}$$

Example 3.2.5. *Consider the following decision problem. You have the option of entering a lottery, for 1 CU, that gives you a prize of 10 CU. The probability of winning is 0.01. This can be formalised by making it a choice between P, where you do not enter the lottery and Q, which represents entering the lottery. Now we can calculate the expected utility for each choice. This is simply $\mathbb{E}(U \mid P) = \sum_r U(r)P(r)$ and $\mathbb{E}(U \mid Q) = \sum_r U(r)Q(r)$ respectively.*

r	U(r)	P	Q
did not enter	0	1	0
paid 1 CU and lost	−1	0	0.99
paid 1 CU and won 10	9	0	0.01

Table 3.1: A simple gambling problem

	P	Q
$\mathbb{E}(U \mid \cdot)$	0	−0.9

Table 3.2: Expected utility for the gambling problem

Monetary rewards

Example 3.2.6. *Choose between the following two gambles:*

1. *The reward is 500,000 with certainty.*
2. *The reward is 2,500,000 with probability* 0.10*. It is 500,000 with probability* 0.89*, and 0 with probability* 0.01*.*

Example 3.2.7. *Choose between the following two gambles:*

1. *The reward is 500,000 with probability* 0.11*, or 0 with probability* 0.89*.*
2. *The reward is: 2,500,000 with probability* 0.1*, or 0 with probability* 0.9*.*

Exercise 3. *Show that if gamble 1 is preferred in the first example, gamble 1 must also be preferred in the second example.*

The St. Petersburg Paradox

A simple game [Bernoulli, 1713]

- A *fair coin* is tossed until a head is obtained.
- If the first head is obtained on the n-th toss, our reward will be 2^n currency units.

How much are you willing to pay, to play this game once?

- The probability to stop at round n is 2^{-n}.
- Thus, the expected monetary gain of the game is

$$\sum_{n=1}^{\infty} 2^n 2^{-n} = \infty.$$

- If your utility function were linear you'd be willing to pay any amount to play.

3.2.4 Measuring utility

Experimental measurement of utility

Example 3.2.8. *We shall try and measure the utility of all monetary rewards in some interval* $[a, b]$.

Let $\langle a, b\rangle$ *denote a lottery ticket that yields* a *or* b *CU with equal probability. Consider the following sequence:*

1. *Find* x_1 *such that receiving* x_1 *CU with certainty is equivalent to receiving* $\langle a, b\rangle$.
2. *Find* x_2 *such that receiving* x_2 *CU with certainty is equivalent to receiving* $\langle a, x_1\rangle$.
3. *Find* x_3 *such that receiving* x_3 *CU with certainty is equivalent to receiving* $\langle x_1, b\rangle$.
4. *Find* x_4 *such that receiving* x_4 *CU with certainty is equivalent to receiving* $\langle x_2, x_3\rangle$.

If $x_1 \neq x_4$, then your preferences do not meet the requirements of a utility function.

Exercise 4. 1. *Specify an amount* a*, then observe random value* Y.

2. *If* $Y \geq a$*, receive* Y.
3. *If* $Y < a$*, receive random reward* X *with known distribution (independent of* Y*).*
4. *Show that you should choose* a *s.t.* $U(a) = \mathbb{E}[U(X)]$.

3.2.5 Convex and concave utility functions

Convex functions

Definition 3.2.3. *A function g is convex on A if, for any points $x, y \in A$, and any $\alpha \in [0,1]$:*

$$\alpha g(x) + (1-\alpha)g(y) \geq g[\alpha x + (1-\alpha)y]$$

Theorem 3.2.1 (Jensen's inequality)**.** *If g is convex on $\mathcal{S}$ and $x \in \mathcal{S}$ with measure $P(A) = 1$ and $\mathbb{E}(x)$ and $\mathbb{E}[g(x)]$ exist, then:*

$$\mathbb{E}[g(x)] \geq g[\mathbb{E}(x)]. \tag{3.2.4}$$

Example 3.2.9. *If the utility function is convex, then we choose a gamble giving a random gain x rather than one giving a fixed gain $\mathbb{E}(x)$. Thus, a convex utility function implies risk-taking.*

Concave functions

Definition 3.2.4. *A function g is concave on $\mathcal{S}$ if, for any points $x, y \in \mathcal{S}$, and any $\alpha \in [0,1]$:*

$$\alpha g(x) + (1-\alpha)g(y) \leq g[\alpha x + (1-\alpha)y]$$

For concave functions, an analogue of Jensen's inequality holds (in the other direction). If the utility function is concave, then we choose a gamble giving a fixed gain $\mathbb{E}[X]$ rather than one giving a random gain X. Consequently, a concave utility function implies risk aversion.

Example 3.2.10 (Insurance)**.** *The act of buying insurance can be related to concavity of our utility function.*

Let x be the insurance cost, h our insurance cover and ϵ the probability of needing the cover. Then we are going to buy insurance if the utility of losing x with certainty is greater than the utility of losing $-h$ with probability ϵ.

$$U(-x) > \epsilon U(-h) + (1-\epsilon)U(0). \tag{3.2.5}$$

The company has a linear utility, and fixes the premium x high enough for

$$x > \epsilon h. \tag{3.2.6}$$

Consequently, we see that from (3.2.6) *that $U(-\epsilon h) \geq U(-x)$, as U is an increasing function. Thus, from* (3.2.5) *we obtain $U(-\epsilon h) > \epsilon U(-h) + (1-\epsilon)U(0)$. Now the $-\epsilon h$ term is the utility of our expected monetary loss, while the right hand side is our expected utility. Consequently if the inequality holds, our utility function is (at least locally) concave.*

3.3 Summary

Summary

- We can subjectively indicate which events we think are more likely.
- Using relative likelihoods, we can define a *subjective probability* P for all events.

- Similarly, we can subjectively indicate *preferences for rewards*.
- We can determine a *utility function* for all rewards.
- Hypothesis: we prefer the probability distribution (over rewards) with the highest *expected utility*.
- Concave utility functions imply *risk aversion* (and convex, risk-taking).

Chapter 4

Decision problems

4.1 Introduction

In this chapter we describe how to actually formulate statistical decision problems. The simplest such problem arises when we have a choice between a number of different decisions, where each decision gives us different *rewards* with different probabilities. If these probabilities are known, then the framework of expected utility maximisation gives a solution to the problem. An example includes *gambling*, where we must choose between a number of possible lotteries, each one having different payoffs and winning probabilities.

Another classical setting is parameter estimation. Therein, we stipulate the existence of a parameterised *law of nature*, and we wish to choose a best-guess set of parameters for the law through measurements and some prior information, such as for example determining the gravitational attraction constant from observing planetary movements. These measurements are always obtained through experiments, and the automatic design of those experiment is a topic we shall consider in later chapters.

Finally, these decisions will necessarily depend on our prior information, even if have access to some additional measurements. The last section of this chapter will examine how sensitive our decisions are to the prior, and how we can choose a prior distribution so that our decisions are robust.

4.2 Rewards that depend on the outcome of an experiment

Consider the problem of choosing between a two different types of tickets in raffle. Each type of ticket gives you the chance to win a different prize. The first is a bicycle and the second is a tea set. As most people opt for the bicycle, your chance of actually winning it is much smaller. However, if you prefer winning a bicycle to winning the tea set, it is not clear what choice you should make in the raffle. The above is the quintessential example for problems where the reward that we obtain depends not only on our decisions, but also in the outcome of an *experiment*.

This problem can be viewed more generally, for scenarios where the reward you receive depends not only on your own choice, but also some other, unknown fact in the world. This may be something completely uncontrollable, and hence you only have to make the best possible guess.

More formally, we must make a decision $d \in \mathcal{D}$ *before* knowing the outcome ω of an experiment with outcomes in Ω. After the experiment is performed, we obtain a *reward* $r \in \mathcal{R}$ which depends on both the outcome of the experiment ω and our decision. As discussed in the previous chapter, our preferences for some rewards over others is determined by a *utility* function $U : \mathcal{R} \to \mathbb{R}$, such that we prefer r to r' if and only if $U(r) \geq U(r')$. Now, however, we cannot choose rewards directly. Another example, which will be used throughout this section, is the following.

Example 4.2.1 (Taking the umbrella)**.** *We must decide whether to take an umbrella to work. Our reward is a combination of whether we get wet and the amount of objects that we carry. We would rather not get wet and not carry too many things, which can be made more precise by choosing an appropriate utility*

function. For example, we might put a value of -1 *for carrying the umbrella, a value of* -10 *for getting wet. In this example, the only events of interest are whether it rains or not.*

4.2.1 Formalisation of the problem setting

We are now ready to formulate the problem setting more precisely.

Assumption 4.2.1 (Outcomes)**.** *There exists a probability measure* P *on* $(\Omega, \mathfrak{F}_\Omega)$ *such that the probability of the random outcome* ω *being in* $A \subset \Omega$ *is:*

$$\mathbb{P}(\omega \in A) = P(A), \qquad \forall A \in \mathfrak{F}_\Omega. \tag{4.2.1}$$

This probability is completely independent of any decision that we make.

Assumption 4.2.2 (Utilities)**.** *Our rewards satisfy all our assumptions from Chapter 3: Preferences are transitive, all rewards are comparable and there exists a utility function* U*, measurable with respect to* $\mathfrak{F}_\mathcal{R}$ *such that* $U(r') \geq U(r)$ *iff* $r \succ^* r'$*.*

Since the random outcome ω does not depend on our decision d, we must find a way to connect the two. This can be formalised via a reward function, so that the reward that we obtain (whether we get wet or not) depends on both our decision (to take the umbrella) and the random outcome (whether it rains).

Definition 4.2.1 (Reward function)**.** *A reward function* $\rho : \Omega \times \mathcal{D} \to \mathcal{R}$ *defines the reward we obtain if we select* $d \in \mathcal{D}$ *and the experimental outcome is* $\omega \in \Omega$*:*

$$r = \rho(\omega, d). \tag{4.2.2}$$

The decision space might be arbitrarily more complex than the one we have seen so far. For example, our decisions may be distributions over simple decisions, or functions whose value depends on future events. We shall examine those problems later in the chapter.

When we discussed the problem of choosing between distributions, in section 3.2.2, we had directly defined probability distributions on rewards. We can now formulate our problem in that setting. First, we define a set of distributions $\{P_d \mid d \in \mathcal{D}\}$ on the reward space $(\mathcal{R}, \mathfrak{F}_\mathcal{R})$, such that the decision d amounts to choosing a particular distribution P_d on the rewards.

Example 4.2.2 (Rock/Paper/Scissors)**.** *Consider a simple game of rock-paper-scissors, where your opponent plays a move at the same time as you, so that you cannot influence his move. The opponents moves are thus* $\Omega = \{\omega_R, \omega_P, \omega_S\}$*.*

You have studied your opponent for some time and you believe *that he is most likely to play rock* $P(\omega_R) = 3/6$*, somewhat likely to play paper* $P(\omega_P) = 2/6$*, and less likely to play scissors:* $P(\omega_S) = 1/6$*. Your decision set is your own moves:* $\mathcal{D} = \{d_R, d_P, d_S\}$*, for rock, paper, scissors, respectively. The reward set is* $\mathcal{R} = \{\mathit{Win}, \mathit{Draw}, \mathit{Lose}\}$*.*

What is the probability of each reward, for each decision you make? Taking the example of d_R*, we see that you win if the opponent plays scissors with probability* $1/6$*, you lose if the opponent plays paper (*$2/6$*), and you draw if he plays rock (*$3/6$*). Consequently, we can convert the outcome probabilities to reward probabilities for every decision:*

$$
\begin{aligned}
P_{d_R}(\mathit{Win}) &= 1/6, & P_{d_R}(\mathit{Draw}) &= 3/6, & P_{d_R}(\mathit{Lose}) &= 2/6\\
P_{d_P}(\mathit{Win}) &= 3/6, & P_{d_P}(\mathit{Draw}) &= 2/6, & P_{d_P}(\mathit{Lose}) &= 1/6\\
P_{d_S}(\mathit{Win}) &= 2/6, & P_{d_S}(\mathit{Draw}) &= 1/6, & P_{d_S}(\mathit{Lose}) &= 3/6.
\end{aligned}
$$

Of course, what we play depends on our own utility function. If we prefer winning over drawing or losing, we could for example have the utility function $U(\mathit{Win}) = 1$, $U(\mathit{Draw}) = 0$, $U(\mathit{Lose}) = -1$. *Then, since* $\mathbb{E}_d U = \sum_{\omega \in \Omega} U(\omega, d) P_d(\omega)$, *we have*

$$
\begin{aligned}
E_{d_R} U &= -1/6\\
E_{d_P} U &= 2/6\\
E_{d_S} U &= -1/6
\end{aligned}
$$

More generally, every decision that we make creates a corresponding probability distribution on rewards.

The probability measure induced by decisions

For every $d \in \mathcal{D}$, the function $\rho : \Omega \times \mathcal{D} \to \mathcal{R}$ induces a probability distribution P_d on $\mathcal{R}$. In fact, for any $B \in \mathfrak{F}_{\mathcal{R}}$:

$$P_d(B) \triangleq \mathbb{P}(\rho(\omega, d) \in B) = P(\{\omega \mid \rho(\omega, d) \in B\}). \tag{4.2.3}$$

The above equation requires that the following technical assumption is satisfied:

Assumption 4.2.3. *The sets* $\{\omega \mid \rho(\omega, d) \in B\}$ *must belong to* $\mathfrak{F}_\Omega$. *That is,* ρ *must be* $\mathfrak{F}_\Omega$*-measurable for any* d.

In other words, while the outcome of the experiment is independent of the decision, the distribution of rewards is effectively chosen by our decision, as before. However, this structure allows us to clearly distinguish the controllable from the random part of the rewards.

In either case, we employ the expected utility hypothesis (Assumption 3.2.2). Thus, we should choose the decision that results in the highest expected utility.

Expected utility

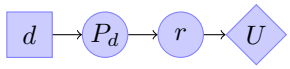
The expected utility of any decision $d \in \mathcal{D}$ under P is: the expected utility is:

$$\mathbb{E}_{P_d}(U) = \int_R U(r) \, \mathrm{d}P_d(r) = \int_\Omega U[\rho(\omega, d)] \, \mathrm{d}P(\omega) \tag{4.2.4}$$

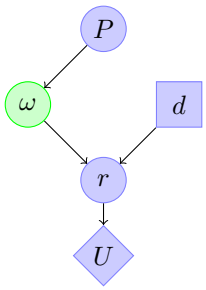
From now on, we shall use the simple notation

$$U(P, d) \triangleq \mathbb{E}_{P_d} U \tag{4.2.5}$$

to denote the expected utility of d under distribution P.



(a) The combined decision problem



(b) The separated decision problem

Figure 4.1: Decision diagrams for the combined and separated formulation of the decision problem. Squares denote decision variables, diamonds denote utilities. All other variables are denoted by circles. Arrows denote the flow of dependency.

$\rho(\omega, d)$	d_1	d_2
ω_1	dry, carrying umbrella	wet
ω_2	dry, carrying umbrella	dry
$U[\rho(\omega, d)]$	d_1	d_2
ω_1	0	-10
ω_2	0	1
$\mathbb{E}_P(U \mid d)$	0	-1.2

Table 4.1: Rewards, utilities, expected utility for 20% probability of rain.

Instead of viewing the decision as effectively choosing a distribution over rewards (Fig. 4.1(a)) we can separate the random part of the process from the deterministic part (Fig. 4.1(b)) by considering a measure P on some space of outcomes Ω, such that the reward depends on both d and the outcome $\omega \in \Omega$ through a function $\rho(\omega, d)$. The optimal decision is of course always the $d \in \mathcal{D}$ maximising $\mathbb{E}(U \mid P_d)$.

The dependency structure of this problem in either formulation can be visualised in the *decision diagram* shown in Figure 4.1(a).

Example 4.2.3. *You are going to work, and it might rain. The forecast said that the probability of rain* (ω_1) *was* 20%. *What do you do?*

- d_1*: Take the umbrella.*
- d_2*: Risk it!*

The reward of a given outcome and decision combination, as well as the expected utility is given in table 4.1.

4.2.2 Decision diagrams

Decision diagrams are also known as *decision networks* or *influence diagrams*. Like the examples shown in Figure 4.1, they are used to show dependencies between different variables. In general, these include the following types of nodes:

- Choice nodes, denoted by squares. These are nodes whose values the decision maker can directly choose. Sometimes there is more than one decision maker involved.
- Value nodes, denoted by diamonds. These are the nodes that the decision maker is interested in influencing. The utility of the decision maker is always a function of the value nodes.
- Circle nodes are used to denote all other types of variables. These include deterministic, stochastic, known or unknown variables.

The nodes are connected via directed edges. These denote the dependencies between nodes. For example, in Figure 4.1(b), the reward is a function of both ω and d, i.e. $r = \rho(\omega, d)$, while ω depends only on the probability distribution P. Typically, there must be a path from a choice node to a value node, otherwise nothing the decision maker can do will influence its utility. Nodes belonging to or observed by different players will usually be denoted by different lines or colors. In Figure 4.1(b), ω, which is not observed, is shown in a lighter color.

4.2.3 Statistical estimation*

This is especially the case in statistical problem of *parameter estimation,* such as estimating the covariance matrix of a Gaussian random variable. A simple example is estimating the distribution of votes in an election from a small sample, given below.

Example 4.2.4 (Voting)**.** *Let us say for example that you wish to estimate the number of votes for different candidates in an election. The* unknown parameters *of the problem mainly include: the percentage of likely voters in the population, the probability that a likely voter is going to vote for each candidate. One simple way to estimate this is by polling.*

Consider a nation with k political parties. Let $\omega = (\omega_1, \dots, \omega_k) \in [0,1]^k$ be the voting percentages for each party. We wish to make a guess $d \in [0,1]^k$. How should we guess, given a distribution $P(\omega)$? How should we select U and ρ? This depends on what our goals is, when we make the guess.

If we wish to give a reasonable estimate about all the k parties votes, we can use the squared error: First, set $\rho(\omega, d) = (\omega_1 - d_1, \dots, \omega_k - d_k)$, our error vector $r \in [0,1]^k$. Then we set $U(r) \triangleq -\|r\|^2$, where $\|r\|^2 = \sum_i |x_i|^2$.

If on the other hand, we just want to predict the winner of the election, then the actual percentages of all individual parties are not important. In that case, we can set $\rho(\omega, d) = 1$ if $\arg\max_i \omega_i = \arg\max_i d_i$ and 0 otherwise, and $U(r) = r$.

- The unknown outcome of the experiment ω is called a *parameter.*
- The set of outcomes Ω is called the *parameter space.*

Losses and risks

In such problems, it is common to specify a loss instead of a utility. This is usually the negative utility:

Definition 4.2.2 (Loss)**.**

$$\ell(\omega, d) = -U[\rho(\omega, d)]. \tag{4.2.6}$$

Given the above, instead of the expected utility, we consider the expected loss, or risk.

Definition 4.2.3 (Risk)**.**

$$\sigma(P, d) = \int_{\Omega} \ell(\omega, d) \, \mathrm{d}P(\omega). \tag{4.2.7}$$

Of course, the optimal decision is d minimising σ.

4.3 Bayes decisions

The decision which maximises the expected utility under a particular distribution P, is called the *Bayes-optimal* decision, or simply the *Bayes decision*. The probability distribution P is supposed to reflect all our uncertainty about the problem.

Definition 4.3.1 (Bayes-optimal utility)**.** *Consider an outcome (or parameter) space Ω, decision space $\mathcal{D}$, and a utility function $U : \Omega \times \mathcal{D} \to \mathbb{R}$. For any probability distribution P on Ω, the Bayes-optimal utility $U^*(P)$ is defined as the smallest upper bound on $U(P, d)$ for all decisions $d \in \mathcal{D}$. That is,*

$$U^*(P) = \sup_{d \in D} U(P, d). \tag{4.3.1}$$

Remark 4.3.1 (The sup notation)**.** *A clarification of the supremum notation is in order. When we say that*

$$M = \sup_{x \in A} f(x),$$

then: (i) $M \geq f(x)$ for any $x \in A$. In other words, M is an upper bound on $f(x)$. (ii) for any $M' > M$, there exists some $x' \in A$ s.t. $M' < f(x')$. In other words, there exists no smaller upper bound than M. When the function f has a maximum, then the supremum is identical to the maximum.

As can be seen from Figure **??**, for absolute loss, the optimal decision is to choose the d that is closest to the most likely ω. However, for quadratic loss, Figure **??** appears to indicate that the optimal choice should be equal to the expected value of ω. This is actually true in general for quadratic loss, and for $d, \omega \in \mathbb{R}$, as shall be seen from the following example.

Example 4.3.1 (Quadratic loss)**.** *Now consider $\Omega = \mathbb{R}$ with measure P and $\mathcal{D} = \mathbb{R}$. For any point $\omega \in \mathbb{R}$, we define the utility as:*

$$U(\omega, d) = -|\omega - d|^2. \tag{4.3.2}$$

The optimal decision maximises

$$U(P, d) = -\int_{\mathbb{R}} |\omega - d|^2 \, \mathrm{d}P(\omega).$$

Then, as long as $\partial/\partial d|\omega - d|^2$ *is measurable with respect to* $\mathfrak{F}_{\mathbb{R}}$

$$\begin{aligned}
\frac{\partial}{\partial d} \int_{\mathbb{R}} |\omega - d|^2 \, \mathrm{d}P(\omega) &= \int_{\mathbb{R}} \frac{\partial}{\partial d} |\omega - d|^2 \, \mathrm{d}P(\omega) && (4.3.3)\\
&= 2 \int_{\mathbb{R}} (d - \omega) \, \mathrm{d}P(\omega) && (4.3.4)\\
&= 2 \int_{\mathbb{R}} d \, \mathrm{d}P(\omega) - 2 \int_{\mathbb{R}} \omega \, \mathrm{d}P(\omega) && (4.3.5)\\
&= 2d - 2\,\mathbb{E}(\omega), && (4.3.6)
\end{aligned}$$

so the expected utility is maximised for $d = \mathbb{E}(\omega)$.

4.3.1 Convexity of the Bayes-optimal utility*

We shall show now the expected utility is linear. Consequently, the Bayes-utility is convex with respect to the distribution P. This firstly implies that there is a unique "worst" distribution P, against which we cannot do very well. Secondly, we can approximate the Bayes-utility very well for all possible distributions by generalising from a small number of distributions. In order to define linearity and convexity, we must first introduce the concept of a mixture of distributions.

A mixture of distributions

Consider two probability measures P, Q on $(\Omega, \mathfrak{F}_\Omega)$.

These define two alternative distributions for ω. For any P, Q and $\alpha \in [0, 1]$, we define

$$Z_\alpha \triangleq \alpha P + (1 - \alpha) Q \tag{4.3.7}$$

to mean the probability measure such that $Z_\alpha(A) = \alpha P(A) + (1 - \alpha) Q(A)$ for any $A \in \mathfrak{F}_\Omega$.

Remark 4.3.2 (Linearity of the expected utility)**.** *If* Z_α *is as defined in* (4.3.7)*, then, for any* $d \in \mathcal{D}$*:*

$$U(Z_\alpha, d) = \alpha U(P, d) + (1 - \alpha) U(Q, d). \tag{4.3.8}$$

Proof. This follows from the linearity of expectation, i.e.

$$\begin{aligned}
U(Z_\alpha, d) &= \int_\Omega U(\omega, d) \, \mathrm{d}Z_\alpha(\omega) && (4.3.9)\\
&= \alpha \int_\Omega U(\omega, d) \, \mathrm{d}P(\omega) + (1 - \alpha) \int_\Omega U(\omega, d) \, \mathrm{d}Q(\omega) && (4.3.10)\\
&= \alpha U(P, d) + (1 - \alpha) U(Q, d). && (4.3.11)
\end{aligned}$$

□

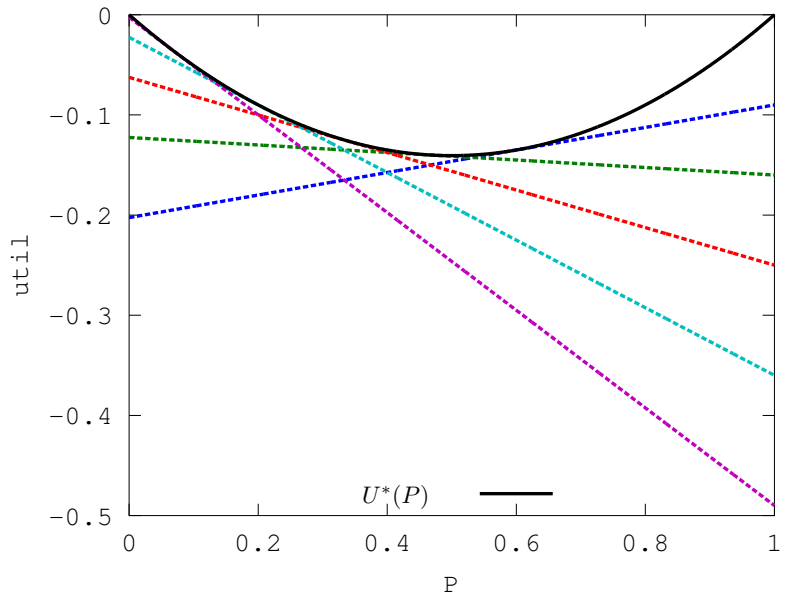


Figure 4.2: A strictly convex Bayes utility.

Theorem 4.3.1. *For probability measures P, Q on Ω and any $\alpha \in [0,1]$,*

$$U^*[Z_\alpha] \leq \alpha U^*(P) + (1-\alpha)U^*(Q), \tag{4.3.12}$$

where $Z_\alpha = \alpha P + (1-\alpha)Q$.

Proof. From the definition of the expected utility (4.3.8), for any decision $d \in \mathcal{D}$,

$$U(Z_\alpha, d) = \alpha U(P,d) + (1-\alpha)U(Q,d).$$

Hence, by definition (4.3.1) of the Bayes-utility:

$$\begin{aligned} U^*(Z_\alpha) &= \sup_{d\in D} U(Z_\alpha, d) \\ &= \sup_{d\in D} [\alpha U(P,d) + (1-\alpha)U(Q,d)]. \end{aligned}$$

Use $\sup_x [f(x)+g(x)] \leq \sup_x f(x) + \sup_x g(x)$ to bound r.h.s:

$$\begin{aligned} U^*[Z_\alpha] &\leq \alpha \sup_{d\in D} U(P,d) + (1-\alpha) \sup_{d\in D} U(Q,d) \\ &= \alpha U^*(P) + (1-\alpha)U^*(Q). \end{aligned}$$

□

Convexity of the Bayes utility

As we have proven, the expected utility is linear with respect to P. Thus, for any fixed decision d we obtain one of the lines in Fig. 4.2. Due to the theorem just proved, the Bayes risk is concave. Furthermore, the minimising decision

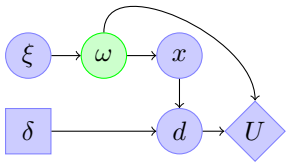


Figure 4.3: Statistical decision problem with observations

for any P is tangent to the risk at the point $(P, U^*(P))$. If we take a decision that is optimal with respect to some P, but the distribution is in fact $Q \neq P$, then we are not far from the optimal when P and Q are close and U^* is smooth. Consequently, we can trivially lower bound the Bayes utility by examining a finite set of decisions $\hat{D}$:

$$U^*(P) \geq \max_{d \in \hat{D}} U(P, d) \forall P$$

In addition, we can upper-bound the Bayes utility as follows. Take any two distributions P_1, P_2 in the set of allowed distributions. Then, the following upper bound holds

$$U^*(\alpha P_1 + (1-\alpha)P_2) \leq \alpha U^*(P_1) + (1-\alpha)U^*(P_2)$$

due to convexity. The two bounds suggest an algorithm for successive approximation of the Bayes risk, by looking for the largest gap between the lower and the upper bounds.

4.4 Decision problems with observations

So far we have only examined problems where the outcomes were drawn from some fixed distribution. This distribution constituted our subjective belief about what the unknown parameter is. Now, we examine the case where we can obtain some observations that depend on the unknown ω before we make our decision. These observations should give us more information about ω, before making a decision. Intuitively, we should be able to make decisions by simply considering the posterior distribution. The following section will investigate whether this is true.

Obtaining information

In this setting, we once more need to take some decision $d \in \mathcal{D}$ so as to maximise expected utility. As before, we have a prior distribution ξ on some parameter $\omega \in \Omega$, representing what we know about ω. Consequently, the expected utility of any fixed decision d is going to be $\mathbb{E}_\xi(U \mid d)$.

However, it might be possible to obtain more infomation about ω before making a decision. In particular, each ω corresponds to a *model* of the world ψ_ω. This is expressed as a probability distribution over the observation space $\mathcal{S}$, such that $\psi_\omega(X)$ is the probability that the observation is in $X \subset \mathcal{S}$. The set of parameters Ω thus defines a family of models:

$$\Psi \triangleq \{\psi_\omega \mid \omega \in \Omega\}. \tag{4.4.1}$$

Now, consider the case where we take an observation x from the true model ψ_{ω^*} before having to make a decision. We can represent the dependency of our decisions on the observations by making our decision a function of x:

Definition 4.4.1 (Decision function)**.** *A decision function $\delta : \mathcal{S} \to \mathcal{D}$ maps from the set of possible observations $\mathcal{S}$ to the set of possible decisions.*

The expected utility of a decision function δ is:

$$U(\xi, \delta) \triangleq \mathbb{E}_\xi \{U[\omega, \delta(x)]\} = \int_\Omega \left(\int_{\mathcal{S}} U[\omega, \delta(x)] \, \mathrm{d}\psi_\omega(x) \right) \mathrm{d}\xi(\omega). \tag{4.4.2}$$

When the set of decision functions includes all fixed decisions, then there is a decision function δ^* at least as good as the best fixed decision d^*. More formally:

Remark 4.4.1. *Let $\mathscr{D}$ denote the set of decision functions $\delta : \mathcal{S} \to \mathcal{D}$. If, $\forall d \in \mathcal{D}$ $\exists \delta \in \mathscr{D}$ such that $\delta(x) = d$ $\forall x \in \mathcal{S}$, then $\sup_{\delta \in \mathscr{D}} \mathbb{E}_\xi(U \mid \delta) \geq \sup_{d \in \mathcal{D}} \mathbb{E}_\xi(U \mid d)$.*

Proof. The proof follows by setting $\mathscr{D}_0$ to be the set of fixed decision functions. The result follows since $\mathscr{D}_0 \subset \mathscr{D}$. □

This is the standard Bayesian framework for decision making. It may be slightly more intuitive in some case to use the notation $\psi(x \mid \omega)$, in order to emphasize that this is a conditional distribution. However, there is no technical difference between the two notations.

Example 4.4.1. *Consider the problem of deciding whether or not to go to a particular restaurant. Let $\Omega = [0, 1]$ with $\omega = 0$ meaning the food is in general horrible and $\omega = 1$ meaning the restaurant is great. Let $x_1, \ldots, x_n$ be n expert opinions in $\mathcal{S} = \{0, 1\}$ about the restaurant. Under our model, the probability of observing $x_i = 1$ when the quality of the restaurant is ω is given by $\psi_\omega(1) = \omega$ and conversely $\psi_\omega(0) = 1-\omega$. The probability of observing a particular[1] sequence x of length n is*

$$\psi_\omega(x) = \omega^s (1 - \omega)^{n-s}$$

with $s = \sum_{i=1}^n x_i$.

Maximising utility when making observations

Statistical procedures based on the notion that a distribution can be assigned to any parameter in a statistical decision problem, as we are assuming here, are called *Bayesian statistical methods*. The scope of these methods has been the subject of much discussion in the statistical literature. See e.g. Savage [1972].

In the following, we shall look at different expressions for the expected utility. We shall overload the utility operator U for various cases: when the parameter is fixed, when the parameter is random, when the decision is fixed, and when the decision depends on the observation x and thus is random as well.

[1] We obtain a slightly different probability under the binomial model, but the end result is the same.

Expected utility of a fixed decision d with $\omega \sim \xi$
We first consider the expected utility of taking a fixed decision $d \in \mathcal{D}$, when $\mathbb{P}(\omega \in A) = \xi(A)$. This is the case we have dealt with so far.

$$U(\xi, d) \triangleq \mathbb{E}_\xi(U \mid d) = \int_\Omega U(\omega, d) \, \mathrm{d}\xi(\omega). \tag{4.4.3}$$

Expected utility of a decision function δ with fixed $\omega \in \Omega$
Now assume that ω is fixed, but instead of selecting a decision directly, we select a decision that depends on the random observation x, which is distributed according to ψ_ω on $\mathcal{S}$. We do this by defining a function $\delta : \mathcal{S} \to \mathcal{D}$.

$$U(\omega, \delta) = \int_{\mathcal{S}} U(\omega, \delta(x)) \, \mathrm{d}\psi_\omega(x). \tag{4.4.4}$$

Expected utility of a decision function δ with $\omega \sim \xi$
Now we generalise to the case where ω is distributed with measure ξ. Note that the expectation of the previous expression (4.4.4) is by definition written as:

$$U(\xi, \delta) = \int_\Omega U(\omega, \delta) \, \mathrm{d}\xi(\omega), \qquad U^*(\xi) \triangleq \sup_\delta U(\xi, \delta) = U(\xi, \delta^*). \tag{4.4.5}$$

Bayes decision rules

We wish to construct the Bayes decision rule, that is, the decision function with maximal ξ-expected utility. However, doing so by examining all possible decision functions is hard, because (usually) there are many more decision functions than decisions. It is however, easy to find the Bayes decision for each possible observation.

Theorem 4.4.1. *If U is non-negative or bounded, then we can reverse the integration order of*

$$U(\xi, \delta) = \mathbb{E}\left\{U[\omega, \delta(x)]\right\} = \int_\Omega \int_{\mathcal{S}} U[\omega, \delta(x)] \, \mathrm{d}\psi_\omega(x) \, \mathrm{d}\xi(\omega),$$

which is the normal form, to obtain the risk in extensive form.

$$U(\xi, \delta) = \int_{\mathcal{S}} \int_\Omega U[\omega, \delta(x)] \, \mathrm{d}\xi(\omega \mid x) \, \mathrm{d}f(x), \tag{4.4.6}$$

where $f(x) = \int_\Omega \psi_\omega(x) \, \mathrm{d}\xi(\omega)$.

Proof. To prove this when U is non-negative, we shall use Tonelli's theorem. First we need to construct an appropriate product measure. Note that the original is written Let $p(x \mid \omega) \triangleq \frac{\mathrm{d}\psi_\omega(x)}{\mathrm{d}\nu(x)}$ be the Radon-Nikodym derivative of ψ_ω with respect to some dominating measure ν on $\mathcal{S}$. Similarly, let $p(\omega) \triangleq \frac{\mathrm{d}\xi(\omega)}{\mathrm{d}\mu(x)}$ be the corresponding derivative for ξ. Now, the utility can be written as:

$$U(\xi, \delta) = \int_\Omega \int_\mathcal{S} U[\omega, \delta(x)] p(x \mid \omega) p(\omega) \,\mathrm{d}\nu(x) \,\mathrm{d}\mu(\omega) \tag{4.4.7}$$

$$= \int_\Omega \int_\mathcal{S} h(\omega, x) \,\mathrm{d}\nu(x) \,\mathrm{d}\mu(\omega). \tag{4.4.8}$$

Clearly, if U is non-negative, then h is non-negative. Then, Tonelli's theorem applies and

$$U(\xi, \delta) = \int_\mathcal{S} \int_\Omega h(\omega, x) \,\mathrm{d}\mu(\omega) \,\mathrm{d}\mu(x) \tag{4.4.9}$$

$$= \int_\mathcal{S} \int_\Omega p(x \mid \omega) p(\omega) \,\mathrm{d}\mu(\omega) \,\mathrm{d}\nu(x) \tag{4.4.10}$$

$$= \int_\mathcal{S} \int_\Omega p(\omega \mid x) \,\mathrm{d}\mu(\omega) p(x) \,\mathrm{d}\nu(x) \tag{4.4.11}$$

$$= \int_\mathcal{S} \left[\int_\Omega p(\omega \mid x) \,\mathrm{d}\mu(\omega) \right] p(x) \,\mathrm{d}\nu(x) \quad = \int_\mathcal{S} \left[\int_\Omega \mathrm{d}\xi(\omega \mid x) \right] \mathrm{d}f(x), \tag{4.4.12}$$

where $p(x) = \mathrm{d}f(x)/\,\mathrm{d}\nu(x)$. □

We can construct an optimal decision function δ^* as follows. For any specific observed $x \in S$, we set $\delta^*(x)$ to:

$$\delta^*(x) \triangleq \arg\max_{d \in D} \mathbb{E}_\xi(U \mid x, d) = \arg\max_{d \in D} \int_\Omega U(\omega, d) \,\mathrm{d}\xi(\omega \mid x).$$

So now we can plug δ^* in the extensive form to obtain:

$$\int_\mathcal{S} \int_\Omega U[\omega, \delta^*(x)] \,\mathrm{d}\xi(\omega \mid x) \,\mathrm{d}f(x) = \int_\mathcal{S} \left\{ \min_d \int_\Omega U[\omega, d] \,\mathrm{d}\xi(\omega \mid x) \right\} \mathrm{d}f(x).$$

Consequently, there is no need to completely specify the decision function before we have seen x. In particular, this would create problems when $\mathcal{S}$ is large.

Definition 4.4.2 (Prior distribution)**.** *The distribution ξ is called the prior distribution of ω.*

Definition 4.4.3 (Marginal distribution)**.** *The distribution f is called the (prior) marginal distribution of x.*

Definition 4.4.4 (Posterior distribution)**.** *The conditional distribution $\xi(\cdot \mid x)$ is called the posterior distribution of ω.*

Bayes decision rule.

The *optimal decision* given x, is the optimal decision with respect to the *posterior* $\xi(\omega \mid x)$. Thus, we do not need to pre-compute the complete Bayes-optimal decision rule.

4.4.1 Calculating posteriors

Posterior distributions for multiple observations

We now consider how we can re-write the posterior distribution over Ω incrementally. Assume that we observe $x^n \triangleq x_1, \ldots, x_n$. We have a prior ξ on Ω. For the observations, we write:

Observation probability given history x^{n-1} and parameter ω

$$\psi_\omega(x_n \mid x^{n-1}) = \frac{\psi_\omega(x^n)}{\psi_\omega(x^{n-1})}$$

Now we can write the posterior as follows:

Posterior recursion

$$\xi(\omega \mid x^n) = \frac{\psi_\omega(x^n)\xi(\omega)}{f(x^n)} = \frac{\psi_\omega(x_n \mid x^{n-1})\xi(\omega \mid x^{n-1})}{f(x_n \mid x^{n-1})}. \tag{4.4.13}$$

Here $f(\cdot \mid \cdot) = \int_\Omega \psi_\omega(\cdot \mid \cdot)\, \mathrm{d}\xi(\omega)$ is a marginal distribution.

Posterior distributions for multiple independent observations

Now we consider the case where, given the parameter, the next observation does not depend on the history: If $\psi(x_n \mid \omega, x^{n-1}) = \psi_\omega(x_n)$ then $\psi_\omega(x^n) = \prod_{k=1}^n \psi_\omega(x_k)$. Then:

Posterior recursion with conditional independence

$$\xi_n(\omega) \triangleq \xi_0(\omega \mid x^n) = \frac{\psi_\omega(x^n)\xi_0(\omega)}{f_0(x_n)} \tag{4.4.14}$$

$$= \xi_{n-1}(\omega \mid x_n) = \frac{\psi_\omega(x_n)\xi_{n-1}(\omega)}{f_{n-1}(x_n)}, \tag{4.4.15}$$

where we define ξ_t to be the belief at time t. Here $f_n(\cdot \mid \cdot) = \int_\Omega \psi(\cdot \mid \cdot, \omega)\, \mathrm{d}\xi_n(\omega)$ is the marginal distribution with respect to the n-th posterior.

Conditional independence allows us to write the posterior update as an identical recursion at each time t. We shall take advantage of that when we look at *conjugate prior* distributions. For such models, the recursion involves a particularly simple parameter update.

Quick summary

- We want to make a decision against an unknown parameter ω.
- The risk is the negative expected utility.
- The Bayes risk is the minimum risk, and it is concave with respect to the distribution of ω.
- Our decisions can depend on observations, via a decision function.
- We can construct a complete decision function by computing $U(\xi, \delta)$ for all *decision functions* (normal form).
- We can instead wait until we observe x and compute $U[\xi(\cdot \mid x), d]$ for all *decisions* (extensive form).
- The posterior given multiple observations can be computed recursively using independence.

Chapter 5

Estimation

5.1 Calculation of posterior distributions

In the previous unit, we have seen how to make optimal decisions with respect to a given risk function and belief. However, one important question is how belief can be calculated. It is one thing to say that we simply calculate the posterior distribution of a parameter and another thing to actually *do it*. In this unit, we shall look at cases when calculating the posterior distributions of parameters is easy. This occurs when the posterior distribution can be expressed by a function that belongs to the same family as the prior distribution, no matter what the observations are.

In the Bayesian setting, we need to calculate posterior distributions of parameters given data. The basic problem can be stated as follows. Let $\mathscr{P} \triangleq \{P_\omega \mid \omega \in \Omega\}$ be a family of probability measures on $(\mathcal{S}, \mathfrak{F}_\mathcal{S})$ and ξ be our prior probability measure on $(\Omega, \mathfrak{F}_\Omega)$. Given some data $x \sim P_{\omega^*}$, with $\omega^* \in \Omega$, how can we estimate ω^*? The Bayesian answer is, instead of guessing a single ω^*, to estimate the posterior distribution $\xi(\cdot \mid x)$. In general, the posterior measure is a function $\xi(\cdot \mid x) : \mathfrak{F}_\Omega \to [0, 1]$, with:

$$\xi(B \mid x) = \frac{\int_B P_\omega(x) \, \mathrm{d}\xi(\omega)}{\int_\Omega P_\omega(x) \, \mathrm{d}\xi(\omega)}. \tag{5.1.1}$$

The main question is how to calculate this posterior for any value of x in practice. One imagines that if x is a complicated object, this may be a difficult job. However, there exist distribution families and priors such that this calculation is very easy. This happens when a summary of the data that does contains all necessary information can be calculated easily. Formally, this is captured via the concept of a sufficient statistic.

5.2 Sufficient statistics

Sometimes we want to summarise the data we have observed. This can happen when the data is a long sequence of simple observations $x^t = (x_1, \ldots, x_t)$, with $x_k \in \mathbb{R}$. It may also be useful to do so when we have a single observation x, such as a high-resolution image. For some applications, it maybe sufficient to only calculate a really simple function of the data, such as the sample mean, defined below:

Definition 5.2.1 (Sample mean)**.** *The sample-mean $\bar{x}_t : \mathbb{R}^t \to \mathbb{R}$ of a sequence x_k is defined as:*

$$\bar{x}_t \triangleq \frac{1}{t} \sum_{k=1}^{t} x_k, \qquad x_k \in \mathbb{R}. \tag{5.2.1}$$

This summary, or any other function of the observations is called a *statistic*. In particular, we would be interested in using statistics with which we could completely replace all the real data in our calculations, without losing any information. Such statistics are called *sufficient*.

5.2.1 Formalisation of sufficient statistics

We consider the standard probabilistic setting. Let $\mathcal{S}$ be a sample space and Ω be a parameter space defining a family of measures on $\mathcal{S}$:

$$\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}.$$

In addition, we must also define an appropriate prior distribution ξ on the parameter space Ω.

Example 5.2.1. *Consider a sequence of random variables x_k taking values in $\mathcal{X} = \{0, 1\}$. Let $\Omega = \left\{0, \frac{1}{2}, 1\right\}$ be a family of Bernoulli distributions with parameter ω:*

$$\mathbb{P}(x_k = 1 \mid \omega) = P_\omega(\{1\}) = \omega,$$

such that $x_k \sim P_\omega$ for all k. A suitable prior distribution could be defined via $\xi(\omega) = \theta_i$ for some vector $\boldsymbol{\theta} \in \mathbb{\Delta}^3$ on the three-dimensional simplex.

Definition 5.2.2. *Let Ξ be a set of prior distributions on Ω, $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$ be a family of distributions on $\mathcal{S}$. A statistic $T : \mathcal{S} \to \mathcal{Z}$ is a sufficient statistic for $\langle \mathscr{P}, \Xi \rangle$ if:*

$$\xi(\omega \mid x) = \xi(\omega \mid x'), \tag{5.2.2}$$

for any prior $\xi \in \Xi$ and any $x, x' \in \mathcal{S}$ such that:

$$T(x) = T(x').$$

Theorem 5.2.1. *A statistic $T : \mathcal{S} \to \mathcal{Z}$ is sufficient for a family $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$ of probability functions on $\mathcal{S}$ iff there exist functions $u : \mathcal{S} \to (0, \infty)$, and $v : \mathcal{Z} \times \Omega \to [0, \infty)$ such that $\forall x \in \mathcal{S}, \omega \in \Omega$:*

$$P_\omega(x) = u(x)v[T(x), \omega], \qquad u > 0, v \geq 0. \tag{5.2.3}$$

Proof. The proof will be for the general case. The case when Ω is finite is technically simpler and is left as an exercise. Assume the existence of u, v. Then for $B \in \mathfrak{F}_\Omega$:

$$\begin{aligned} \xi(B \mid x) &= \frac{\int_B u(x)v[T(x), \omega] \, \mathrm{d}\xi(\omega)}{\int_\Omega u(x)v[T(x), \omega] \, \mathrm{d}\xi(\omega)} \\ &= \frac{\int_B v[T(x), \omega] \, \mathrm{d}\xi(\omega)}{\int_\Omega v[T(x), \omega] \, \mathrm{d}\xi(\omega)}. \end{aligned}$$

If $T(x) = T(x')$, then the above is also equal to $\xi(\omega \mid x')$, so T is a sufficient statistic.

Conversely, let T be a sufficient statistic. Let μ be a dominating measure on $\mathcal{S}$ so that $p(\omega) \triangleq \frac{\mathrm{d}\xi(\omega)}{\mathrm{d}\mu(\omega)}$ and take the derivative at $B \to \omega$ so that:

$$p(\omega \mid x) \triangleq \frac{\mathrm{d}\xi(\omega \mid x)}{\mathrm{d}\mu(\omega)} = \frac{P_\omega(x)p(\omega)}{\int_\Omega P_\omega(x) \, \mathrm{d}\xi(\omega)}$$

Consequently, we can write:

$$P_\omega(x) = \frac{p(\omega \mid x)}{p(\omega)} \int_\Omega P_\omega(x) \, \mathrm{d}\xi(\omega).$$

Since T is sufficient, there is some function $g : \mathcal{Z} \times \Omega$ such that $p(\omega \mid x) = g[T(x), x]$. Consequently, we can factorise P_ω as:

$$P_\omega(x) = v[T(x), \omega]u(x),$$

where $u(x) = \int_\Omega P_\omega(x) \, \mathrm{d}\xi(\omega)$ and $v[T(x), w] = g[T(x), w]/\xi(\omega)$. □

With this factorisation, u is the only factor that depends directly on x.

Example 5.2.2. *Suppose $x^t = (x_1, \dots, x_t)$ is a random sample from a Bernoulli distribution with parameter ω. Then the joint probability is*

$$P_\omega(x^t) = \prod_{k=1}^{t} P_\omega(x_k) = \omega^{s_t}(1-\omega)^{t-s_t}$$

with $s_t = \sum_{k=1}^{t} x_k$ being the number of times 1 has been observed until time t. Then the statistic:

$$T(x^t) = \sum_{k=1}^{t} x_k.$$

satisfies (5.2.3) *with $u(x) = 1$, while $P_\omega(x^t)$ only depends on the data trough the statistic $s_t = T(x^t)$.*

5.2.2 Exponential families

Many well-known distributions such as the Gaussian, Bernoulli and Dirichlet distribution are members of the exponential family of distributions. All such distributions are factorisable in the manner shown below, while at the same time they have fixed-dimension sufficient statistics.

Definition 5.2.3. *A distribution family $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$ with P_ω a probability density or a probability function defined on the sample space $\mathcal{S}$, is said to be an exponential family if for any $x \in \mathcal{S}$, $\omega \in \Omega$:*

$$P_\omega(x) = a(\omega)b(x) \exp\left[\sum_{i=1}^{k} g_i(\omega)h_i(x)\right]. \tag{5.2.4}$$

Remark 5.2.1. *Among families of distributions satisfying certain regularity conditions, only exponential familes have a fixed-dimension sufficient statistic. That is, there is some constant $C < \infty$ such that:*

$$\sup_{X \in \mathfrak{X}_\mathcal{S}} \dim(\{T(x) \mid x \in \mathcal{S}\}) \leq C. \tag{5.2.5}$$

Because of this property, exponential family distributions admit so-called *conjugate* prior distribution families. These have the property that any posterior distribution calculated will remain within the conjugate family. Frequently, because of the simplicity of the statistic used, calculation of the conjugate posterior parameters is very simple.

5.3 Conjugate priors

In this section, we examine some well-known conjugate families. First, we give sufficient conditions for the of existence of conjugate family of priors for a given distribution family and statistic.

Remark 5.3.1. *If a family $\mathscr{P}$ of distributions on $\mathcal{S}$ has a sufficient statistic $T : \mathcal{S} \to \mathcal{Z}$ of fixed dimension for any $x \in \mathcal{S}$, then there exists a conjugate family of priors $\Xi = \{\xi_\alpha \mid \alpha \in A\}$, where A is a set of possible parameters for the prior distribution, such that:*

1. *$P_\omega(x)$ is proportional to some $\xi_\alpha \in \Xi$:*

$$\forall x \in S, \exists \xi_\alpha \in \Xi, c > 0 \ : \ \int_B P_\omega(x) \, \mathrm{d}\xi_\alpha(\omega) = c\xi_\alpha(B), \forall B \in \mathfrak{F}_\Omega$$

2. *The family is closed under multiplication:*

$$\forall \xi_1, \xi_2 \in \Xi, \quad \exists \xi_\alpha \in \Xi, c > 0$$

such that:

$$\xi_\alpha = c\xi_1\xi_2.$$

5.3.1 Bernoulli-Beta conjugate pair

The Bernoulli-Beta conjugate pair of families is the simplest example. It is useful for problems where we wish to measure success rates of independent trials. First, we shall give details on the Bernoulli distribution. Then, we shall define the Beta distribution and describe its conjugate relation to the Bernoulli.

Bernoulli distribution

The Bernoulli distribution is a discrete distribution with outcomes taking values in $\{0, 1\}$. It is ideal for modelling the outcomes of independent random trials with fixed probability of success.

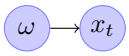


Figure 5.1: Bernoulli graphical model

Definition 5.3.1 (Bernoulli distribution)**.** *The Bernoulli distribution is discrete, with outcomes $\mathcal{S} = \{0, 1\}$, prameter $\omega \in [0, 1]$, and probability function:*

$$P_\omega(u) = \begin{cases} \omega, & u = 1 \\ 1 - \omega, & u = 0 \end{cases} = \omega^u (1 - \omega)^{1-u}.$$

If x_t is distributed according to a Bernoulli distribution with parameter ω, we write $x_t \sim \mathcal{Bern}(\omega)$. The Bernoulli distribution can be extended to $\mathcal{S} = \{0, 1\}^t$ by modelling each out come as independent. Then $P_\omega(x^t) = \prod_{k=1}^t P_\omega(x_k)$. In that case, it directly related to the Binomial distribution.

Definition 5.3.2 (Binomial Distribution)**.** *Let us denote the total number of* 1*'s observed until time* t *by* $s_t = \sum_{k=1}^{t} x_k$*. Then* $\mathbb{P}(s_t = k \mid \omega) = \binom{t}{k}\omega^k(1-\omega)^{t-k}$ *is the probability that,* k *out of* t *trials will be positive. Remember that* $\binom{x}{k} = \prod_{i=0}^{k-1}(x-i)/(1+i)$*. If* s_t *is drawn from a binomial distribution with parameters* ω, t*, we write* $s_t \sim \mathit{Binom}(\omega, t)$*.*

The difference between the two distributions, is that the Bernoulli is a distribution on a sequence of outcomes, while the binomial a distribution on the total number of positive outcomes.

Example 5.3.1. *A fair coin toss can be modelled as a Bernoulli distribution with* $\omega = \frac{1}{2}$*.*

Thus, the Bernoulli distribution is the simplest possible parametric model. If the ω parameter is known, then all the observations are independent of each other. However, this is not the case when ω is unknown. For example, let $\Omega = \{\omega_1, \omega_2\}$. Then $\mathbb{P}(x^t) = \sum_{\omega\in\Omega} \mathbb{P}(x^t \mid \omega)\,\mathbb{P}(\omega) = \sum_{\omega} \prod_{k=1}^{t} \mathbb{P}(x_k \mid \omega)\,\mathbb{P}(\omega)$.

Beta distribution

The Beta distribution is a distribution on the interval $[0, 1]$. It has two parameters that determine the density of the observations. Because the outcomes of this distribution can be used to define the parameter of the Bernoulli distribution, we can now call the distribution's outcomes ω and its parameter α.

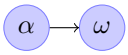


Figure 5.2: Beta graphical model

Definition 5.3.3 (Beta distribution)**.** *The Beta distribution has outcomes* $\omega \in \Omega = [0, 1]$ *and parameters* $\alpha_0, \alpha_1 > 0$*,* $\alpha = (\alpha_1, \alpha_0)$*. It is defined via its probability density function:*

$$f(\omega \mid \alpha) = \frac{\Gamma(\alpha_0 + \alpha_1)}{\Gamma(\alpha_0)\Gamma(\alpha_1)}\omega^{\alpha_1 - 1}(1-\omega)^{\alpha_0 - 1}, \tag{5.3.1}$$

where Γ *is the gamma function. If* x_t *is distributed according to a Beta distribution with parameters* α_1, α_0*, we write:* $x_t \sim \mathit{Beta}(\alpha_1, \alpha_0)$*.*

The Beta family defines a family of probability measures $\mathscr{P} = \{P_\alpha \mid \alpha_0, \alpha_1 > 0\}$, with the probability of $A \subset [0, 1]$: $\mathbb{P}(\omega \in A \mid \alpha) = P_\alpha(A) \triangleq \int_A p(\omega \mid \alpha)\,\mathrm{d}\omega$.

A Beta distribution with parameter α has expectation $\mathbb{E}(x \mid \alpha) = \alpha_1/\|\alpha\|_1$.

Example 5.3.2. *The parameter* $\omega \in [0, 1]$ *of a randomly selected coin can be modelled as a Beta distribution peaking around* $1/2$*. Usually one assumes that coins are fair. However, not all coins are exactly the same. Thus, it is possible that each coin deviates slightly from fairness. We can use a Beta distribution to model how likely (we think) different values* ω *of coin parameters are.*

Figure 5.3 shows the density of a Beta distribution for four different parameter vectors. When $\alpha_0 = \alpha_1 = 1$, the distribution is equivalent to a uniform one.

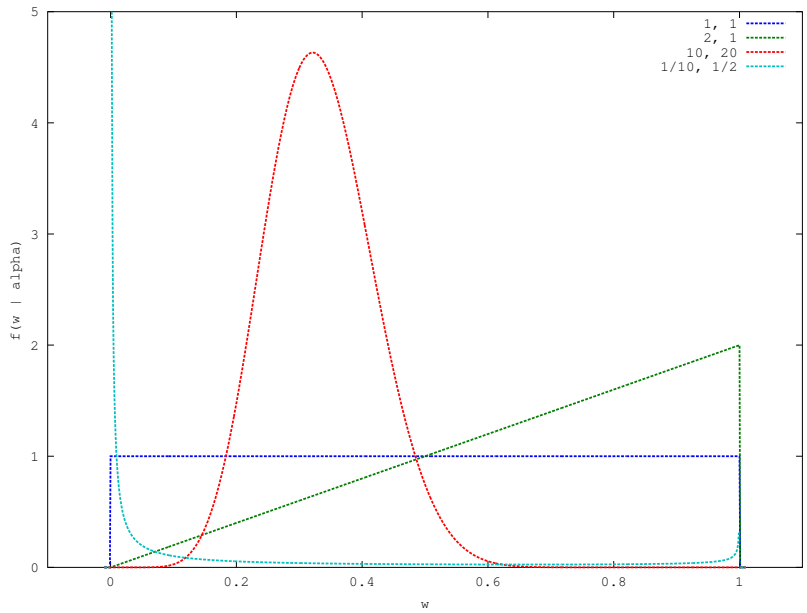


Figure 5.3: Four example Beta densities

The Beta distribution is useful for expressing probabilities of random variables in bounded intervals. In particular, since probabilities of events take values in $[0, 1]$, the Beta distribution is an excellent choice for expressing uncertainty about a probability.

Beta prior for Bernoulli distributions

One simple idea is to encode our uncertainty about an unknown parameter of the Bernoulli distribution, is to use a Beta distribution. The main idea is to assume that the Bernoulli parameter $\omega \in [0, 1]$ is unknown but fixed. We define a Beta prior distribution for ω to represent our uncertainty. This can be summarised by a paramter α and we write $\xi_0(B) \triangleq \int p(\omega \mid \alpha)\, \mathrm{d}\omega$ for our prior distribution ξ_0.

The posterior probability in that case is

$$p(\omega \mid x^t, \alpha) = \frac{\prod_{k=1}^t P_\omega(x_k) p(\omega \mid \alpha)}{\int_\Omega \prod_{k=1}^t P_\omega(x_k) p(\omega \mid \alpha)\, \mathrm{d}\omega} \tag{5.3.2}$$

$$\propto \omega^{s_{t,1}+\alpha_1-1}(1-\omega)^{s_{t,0}+\alpha_0-1} \tag{5.3.3}$$

and $f(x) \propto g(x)$ means that $f(x) = cg(x)$ for some constant c.

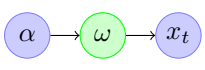


Figure 5.4: Beta-Bernoulli graphical model.

Coin tossing

A simple illustration of the use of the Beta-Bernoulli model is to estimate the bias of a particualr coin. The following figure shows a sequence of beliefs at times $0, 10, 100, 1000$ respectively, from a coin with bias $\omega = 0.6$. To demonstrate

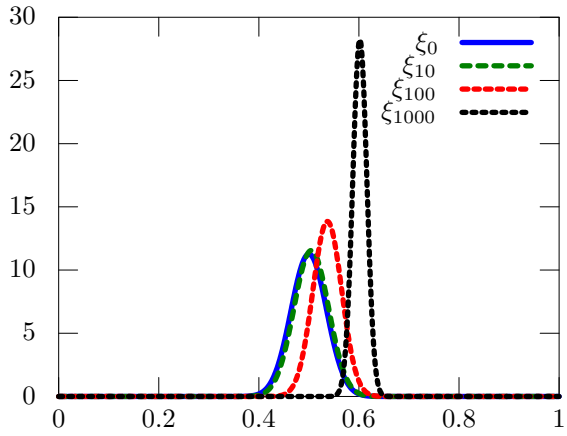


Figure 5.5: Changing beliefs as we observe tosses from a coin with probability $\omega = 0.6$ of heads.

how belief changes, we can perform the following simple experiment. Imagine a coin such that, when it is tossed, it has a probability 0.6 of coming heads every time it is tossed, independently of previous outcomes. Thus, the distribution of outcomes is a Bernoulli distribution with parameter $\omega = 0.6$.

We wish to form an accurate belief about how biased the coin is, under the assumption that the outcomes are Bernoulli with parameter ω. Our initial belief, ξ_0, is modelled as a Beta distribution on the parameter space $\Omega = [0, 1]$, with parameters $\alpha_0 = \alpha_1 = 100$. This places a strong prior on the coin being close to fair. However, we still allow for the possibility that the coin is biased.

Due to the strength of our prior, after 10 observations, the situation has not changed much and the belief ξ_{10} is very close to the starting one. However, after 100 observations, our belief has now shifted towards 0.6, the true bias of the coin. After a total of 1000 observations, our belief is centered very close to 0.6, and is now much more concentrated, reflecting the fact that we are almost certain about the value of ω.

5.4 Credible intervals

Credible interval

According to our current belief ξ, there is a certain subjective probability that the unknown parameter ω takes a certain value. We can use this to construct intervals where we think the unknown parameter is most likely to be.

Definition 5.4.1. *Given some probability measure ξ on Ω representing our*

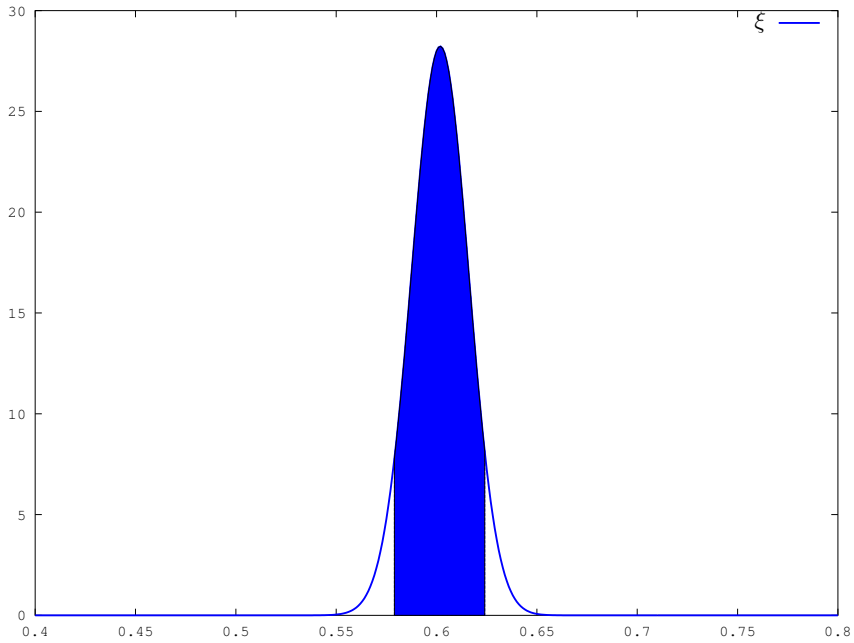


Figure 5.6: 90% credible interval after 1000 observations from a Bernoulli with $\omega = 0.6$.

belief and some interval $A \subset \Omega$,

$$\xi(A) = \int_A \mathrm{d}\xi = \mathbb{P}(\omega \in A \mid \xi).$$

is our subjective belief that the unknown parameter ω *is in* A. *If* $\xi(A) = s$, *then we say that* A *is an* s*-credible interval (or set), or an interval of size (or measure)* s.

Constructing the credible interval.
For prior distributions on $\mathbb{R}$, constructing an s-credible interval is usually done by finding $\omega_u, \omega_l \in \mathbb{R}$ such that

$$\xi([\omega_l, \omega_u]) = s.$$

However, *any* choice of A such that $\xi(A) = s$ is valid.

Figure 5.6 shows the 90% credible interval. (The measure of A under ξ is $\xi(A) = 0.9$.)

What is the probability that the true value of ω will be within a particular credible interval? This will depend on how well our prior ξ_0 matches the true distribution from which the parameter ω was drawn.

Reliability of credible intervals

Assume ϕ, ξ_0 are probability measures on the parameter set Ω, where our prior belief is ξ_0 and ϕ is the actual distribution of $\omega \in \Omega$. Each ω defines a measure P_ω on the observation set $\mathcal{S}$. Let us construct a credible interval $A_t \subset \Omega$ (which is a random variable $A_t : S^t \to \mathscr{P}_\Omega$) such that it has measure $s = \xi_t(A_t)$ for all t. Finally, let $Q \triangleq \int_\omega P_\omega \, \mathrm{d}\phi(\omega)$ be the marginal distribution on $\mathcal{S}$. Then the probability that the credible interval A_t will not include ω is

$$Q\left(\left\{x^t \in S^t \mid \omega \notin A_t\right\}\right).$$

The main question is how this failure probability relates to s, t and ξ_0. There are some results that show that

$$Q\left(\left\{x^t \in S^t \mid \omega \in A_t\right\}\right) \leq (1-s)[1 + D\left(\xi_0 \parallel \phi\right) \mathcal{O}(t^{-1/2})].$$

Let us now design an experiment for examining how often a typical credible interval includes the parameter we are interested in. In order to do so, we will have Nature draw the parameter from some arbitrary distibution ϕ, which may differ from our own assumed prior distribution ξ_0.

Experimental testing of a credible interval

Given a probability family $\mathscr{P} = \{P_\omega \mid w \in \Omega\}$.
Nature chooses distribution ϕ over Ω.
We choose another distribution ξ_0 over Ω.
for $k = 1, \ldots, n$ **do**
  Draw $\omega_k \sim \phi$.
  Draw $x^T \mid \omega_k \sim P_{\omega_k}$.
  for $t = 1, \ldots, T$ **do**
    Calculate $\xi_t(\cdot) = \xi_0(\cdot \mid x^t)$ for all t.
    Calculate A_t, the 50% CI for all t.
    Check failure: $\epsilon_{t,k} = \mathbb{I}\{\omega_k \notin A_t\}$
  end for
end for
Average over all k: $\epsilon_t = \frac{1}{n} \sum_{k=1}^n \epsilon_{t,k}$.

We performed this experiment for $n = 1000$ trials and for $T = 100$ observations per trial. Figure 5.7 illustrates what happens when $\phi = \xi_0$. We see that the credible interval is always centered around our initial mean guess and that it is quite tight. Figure 5.8 shows the average number of times the credible interval A_t around our estimated mean did not match the actual value of ω_k. Since the measure of our interval A_t is always $\xi_t(A_t) = 1/2$, we expect our error probability to be $1/2$. This is always the case.

On the other hand, Figure 5.9 illustrates what happens when $\phi \neq \xi_0$. In fact in that case, $\phi(\omega) = \delta(\omega - 0.6)$, so that $\omega_k = 0.6$ for all trials k. We see that the credible interval is always centered around our initial mean guess and that it is always quite tight. Figure 5.10 shows the average number of failures. We see that initially, due to the fact that our prior is different from the distribution from which the ω_k are selected, we make many more mistakes. However, eventually, our prior is swamped by the data and our error rate converges to 50%.

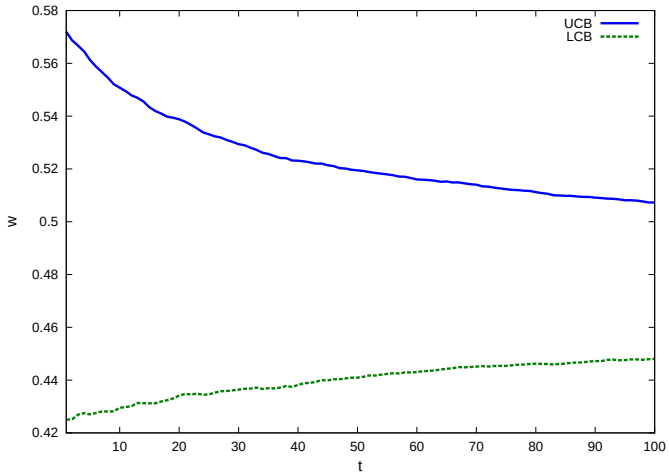


Figure 5.7: 50% credible intervals for a prior $\mathcal{Beta}(10, 10)$, matching the distribution of ω.

5.5 Concentration inequalities

The sample mean revisited

It is interesting to consider the case calculating a sample mean. We have seen that, for the Beta-Bernoulli conjugate prior, it is a simple enough matter to calculate a posterior distribution. From that, we can obtain a credible interval on the expected value of the unknown Bernoulli distribution. However, we would like to do the same for arbitrary distributions on $[0, 1]$, rather than the Bernoulli, which is defined on $\{0, 1\}$.

Example 5.5.1 (Sample mean)**.** *Let $x^t = (x_1, \ldots, x_t)$, with $x^t \sim P$, $\mathbb{E}_P x_k = \mu \in [0, 1]$ for all k, with P an unknown distribution with support in $[0, 1]$. Let*

$$\bar{x}_t \triangleq \frac{1}{t} \sum_{k=1}^{t} x_k.$$

be the sample mean. We are going to look at a number of well-known inequalities that allow to bound $|\hat{x}_t - \mu|$.

The Markov inequality

Theorem 5.5.1 (Markov inequality)**.** *If $X \sim P$, with P a distribution on $[0, \infty)$, then:*

$$\mathbb{P}(X \geq t) \leq \frac{\mathbb{E}\, X}{t}, \tag{5.5.1}$$

where $\mathbb{P}(X \geq t) = P(\{x \mid x \geq t\})$.

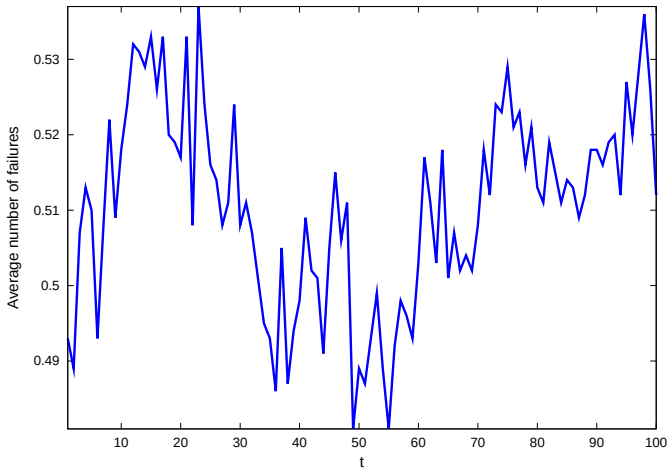


Figure 5.8: Failure rate of 50% CI for a prior $\mathcal{Beta}(10, 10)$, matching the distribution of ω.

Proof. The expectation of X is:

$$\begin{aligned}
\mathbb{E}\, X &= \int_0^\infty x \,\mathrm{d}P(x) && (5.5.2)\\
&= \int_0^t x \,\mathrm{d}P(x) + \int_t^\infty x \,\mathrm{d}P(x) && (5.5.3)\\
&\geq 0 + \int_t^\infty t \,\mathrm{d}P(x) && (5.5.4)\\
&= tP(\{x \mid x \geq t\}) = t\,\mathbb{P}(X \geq t). && (5.5.5)
\end{aligned}$$

□

Consequently, we have that $\mathbb{P}(|\bar{x}_t - p| \geq \epsilon) \leq \mathbb{E}\,|\bar{x}_t - p|/\epsilon$. For $X \in [0, 1]$, we obtain the bound

$$\mathbb{P}(|\bar{x}_t - p| \geq \epsilon) \leq 1/\epsilon.$$

Can we do better? We might, if we take advantage of the following trick. For monotonic f,

$$\mathbb{P}(X \geq t) = \mathbb{P}(f(X) \geq f(t)) \tag{5.5.6}$$

as $\{x \mid x \geq t\} = \{x \mid f(x) \geq f(t)\}$. Thus, we can apply the Markov inequality in a large number of contexts.

The Chebyshev inequality

Theorem 5.5.2. *Let X be a random variable with expectation $\mu = \mathbb{E}\, X$ and variance $\sigma^2 = \mathbb{V}\, X$. Then, for all $k > 0$:*

$$\mathbb{P}(|X - \mu| \geq k\sigma) \leq k^{-2}. \tag{5.5.7}$$

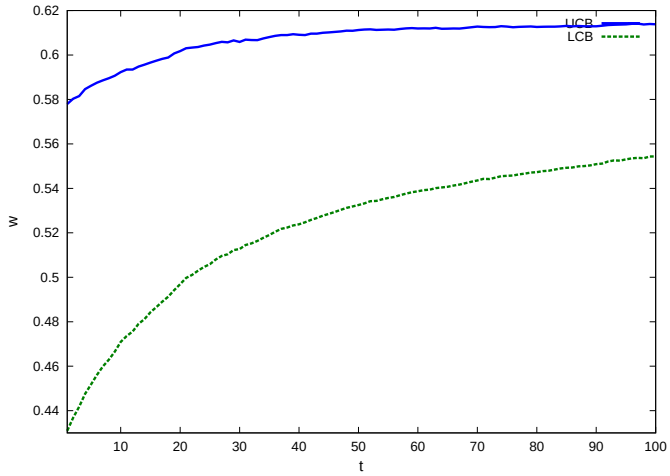


Figure 5.9: 50% credible intervals for a prior $\mathcal{Beta}(10, 10)$, when $\omega = 0.6$.

Proof.

$$\begin{aligned}\mathbb{P}(|X-\mu| \geq k\sigma) &= \mathbb{P}\left(\frac{|X-\mu|}{k\sigma} \geq 1\right) = \mathbb{P}\left(\frac{|X-\mu|^2}{k^2\sigma^2} \geq 1\right) \\ &\leq \mathbb{E}\left(\frac{(X-\mu)^2}{k^2\sigma^2}\right) = \frac{\mathbb{E}(X-\mu)^2}{k^2\sigma^2} = k^{-2}\end{aligned}$$

□

Example 5.5.2 (Application to sample mean)**.** *It is easy to show that the sample mean has expectation μ and variance σ_x^2/t, where $\sigma_x^2 = \mathbb{V}\, x_k$. Consequently:*

$$\mathbb{P}\left(|\bar{x}_t - \mu| \geq k\sigma_x/\sqrt{t}\right) \leq k^{-2}.$$

Setting $\epsilon = k\sigma_x/\sqrt{t}$ we get $k = \epsilon\sqrt{t}/\sigma_x$ and hence

$$\mathbb{P}(|\bar{x}_t - \mu| \geq \epsilon) \leq \frac{\sigma_x^2}{\epsilon^2 t}.$$

Chernoff-Hoeffding bounds

The previous inequality can be quite loose. In fact, one can prove tighter bounds for the estimation of an expected value. All these bounds rest upon a different application of the Markov inequality, due to Chernoff.

Main idea of Chernoff bounds.
Let $S_t = \sum_{k=1}^t X_k$, with $X_k \sim P$ independently, i.e. $X^t \sim P^t$. By definition, from Markov's inequality we obtain in turn, for any $\alpha > 0$

$$\mathbb{P}(S_t \geq u) = \mathbb{P}(e^{\alpha S_t} \geq e^{\alpha u}) \leq e^{-\alpha u}\, \mathbb{E}\, e^{\alpha S_t} = e^{-\alpha u} \prod_k \mathbb{E}\, e^{\alpha X_k}. \tag{5.5.8}$$

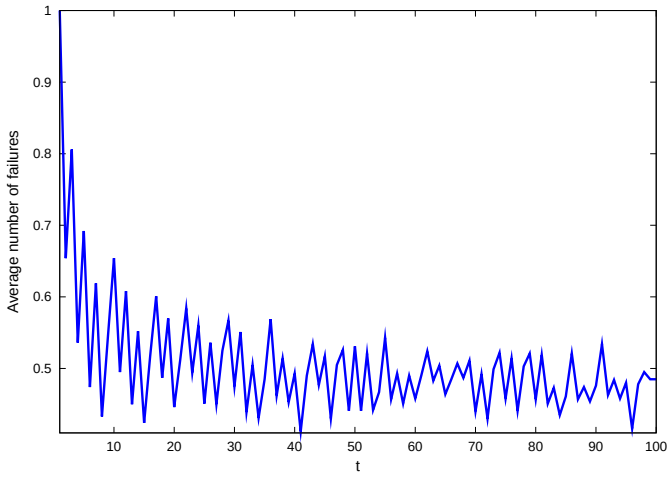


Figure 5.10: Failure rate of 50% CI for a prior $\mathcal{B}eta(10, 10)$, when $\omega = 0.6$.

Theorem 5.5.3. *Hoeffding inequality (Hoeffding [1963], Theorem 2) Let $x_k \sim P_k$ with $x_k \in [a_k, b_k]$ with $\mathbb{E}\, X_k = \mu_k$. Then*

$$\mathbb{P}\left(\bar{x}_t - \mu \geq \epsilon\right) \leq \exp\left(-\frac{2t^2\epsilon^2}{\sum_{k=1}^{t}(b_k - a_k)^2}\right), \tag{5.5.9}$$

where $\bar{x}_t = \frac{1}{t}\sum_{k=1}^{t} x_k$ and $\mu = \frac{1}{t}\sum_{k=1}^{t} \mu_k$.

Proof. Use (5.5.8), setting $X_k = x_k - \mu_k$ so that $S_t = t(\bar{x}_t - \mu)$ and $u = t\epsilon$. Then:

$$\begin{aligned}\mathbb{P}(\bar{x}_t - \mu \geq \epsilon) &= \mathbb{P}(S_t \geq u) \\ &\leq e^{-\theta u}\prod_{k=1}^{t}\mathbb{E}\, e^{\theta X_k} = e^{-\theta t\epsilon}\prod_{k=1}^{t}\mathbb{E}\, e^{\theta(x_k-\mu)}.\end{aligned}$$

Applying Jensen's inequality directly to the expectation does not help. However, we can use convexity in another way. Let $f(x)$ be the linear upper bound on $e^{\theta x}$ on the interval $[a, b]$, i.e.

$$f(x) = \frac{b-x}{b-a}e^{\theta a} + \frac{x-a}{b-a}e^{\theta b} \geq e^{\theta x}.$$

Then obviously $\mathbb{E}\,\theta x \leq \mathbb{E}\, f(x)$. Applying this to the above we get

$$\mathbb{P}(\bar{x}_t - \mu \geq \epsilon) \leq e^{-\theta t\epsilon}\prod_{k=1}^{t}\mathbb{E}\, e^{\theta(x_k-\mu)}.$$

Now

$$e^{\theta(x_k-\mu_k)} \leq \frac{e^{-\theta\mu_k}}{b_k - a_k}\left\{(b_k - \mu_k)e^{\theta a_k} + (\mu_k - a_k)e^{\theta b_k}\right\} = e^{F(\theta_i)},$$

where

$$F(\theta_i) = -\theta_i p_i + \ln(1 - p_i + p_i e^{\theta_i}),$$
$$\theta_i = \theta(b_i - a_i), p_i = \frac{\mu_i - a_i}{b_i - a_i}.$$

Taking derivatives and computing the Taylor expansion, we get

$$\mathbb{E}\, e^{\theta(x_k - \mu_k)} \leq e^{\frac{1}{8}\theta^2 (b_k - a_k)^2}$$
$$\mathbb{P}(\bar{x}_t - \mu \geq \epsilon) \leq e^{-\theta t\epsilon + \frac{1}{8}\theta^2 \sum_{k=1}^{t} (b_k - a_k)^2}.$$

This is minimised at $\theta = 4t\epsilon / \sum_{k=1}^{t}(b_k - a_k)^2$. Plugging it into the above, we obtain the required result. □

Example 5.5.3 (Application to sample mean)**.** *For $x_k \in [0, 1]$:*

$$\mathbb{P}\left(|\bar{x}_t - \mu| \geq \epsilon\right) \leq 2e^{-2t\epsilon^2}$$

5.6 Approximate Bayesian approaches

Monte-Carlo inference

Estimating expectations.
Let $f : S \to [0, 1]$ and P a measure on S. Then

$$\mathbb{E}_P f = \int_S f(x)\, \mathrm{d}P(x). \tag{5.6.1}$$

Estimating expectations is relatively easy, as long as we can generate samples from P. Then, we can our error in estimating its expectation by using the Hoeffding bound.

Corollary 5.6.1. *Let $\hat{f}_n = \frac{1}{n}\sum_t f(x_t)$ with $x_t \sim P$ and $f : \mathcal{S} \to [0, 1]$. Then:*

$$P\left(\left\{x^n \in S^n \;\middle|\; |\hat{f}_n - \mathbb{E}\, f| \geq \epsilon\right\}\right) \leq 2e^{-2n\epsilon^2}. \tag{5.6.2}$$

This technique is simple and fast. However, we frequently cannot sample from P, but only from some alternative distribution Q. Then it is hard to bound our error.

Another interesting application of this technique is the calculation of posterior distributions.

Example 5.6.1 (Calculation of posterior distributions)**.** *Assume a probability family $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$ and a prior distribution ξ on Ω such that we can draw $\omega \sim \xi$. The posterior distribution can be written according to* (5.1.1)*. The nominator can be written as*

$$\int_B P_\omega(x)\, \mathrm{d}\xi(\omega) = \int_\Omega \mathbb{I}\left\{\omega \in B\right\} P_\omega(x)\, \mathrm{d}\xi(\omega) = \mathbb{E}_\xi\left[\mathbb{I}\left\{\omega \in B\right\} P_\omega(x)\right]. \tag{5.6.3}$$

Similarly, the denominator can be written as $\mathbb{E}_\xi[P_\omega(x)]$. If P_ω is bounded, then the errors can be bounded too.

An extension of this approach involves Markov chain Monte-Carlo (MCMC) methods. These are sequential sampling procedures, where data is sampled iteratively. At the k-th iteration, we obtain a sample $x^{(k)} \sim Q_k$, where Q_k depends on the previous sample drawn, $x^{(k-1)}$. Although under mild conditions $Q_k \to P$, there is no easy way to determine *a priori* when the procedure has converged.

5.6.1 Approximate Bayesian Computation

Many times, we have a more fundamental problem. The family of models that we consider An approach that has been popularised by

Algorithm 1 ABC Rejection Sampling

1: $\omega_k \sim \xi$.
2: $\hat{x}^{(k)} \sim M_{\omega_k}$.
3: If $D[T(x), T(\hat{x}^{(k)})] \leq \epsilon$ accept ω_k as a sample from $\xi(\omega \mid x)$.

5.7 Other conjugate families

5.7.1 Conjugates for the normal distribution

Normal distribution

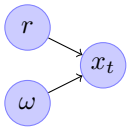


Figure 5.11: Normal graphical model

- Continuous distribution, outcomes: $\mathcal{S} = \mathbb{R}$.
- Parameters: mean $\omega \in \mathbb{R}$, precision $r \in \mathbb{R}^+$.
- Probability density function: $f(x_t \mid \omega, r) = \frac{r}{\sqrt{2\pi}} \exp\left(-\frac{r}{2}(x_t - \omega)^2\right)$
- Independence: $f(x^t \mid \omega, r) = \prod_{k=1}^{t} f(x_k \mid \omega, r) = \left(\frac{r}{\sqrt{2\pi}}\right)^n \exp\left(-\frac{r}{2}\sum_{k=1}^{t}(x_k - \omega)^2\right)$

Normal prior for normal distribution with known precision, unknown mean

The simplest normal estimation problem occurs when we only need to estimate the mean, but we assume that the variance, or equivalently the precision, is known. For Bayesian estimation, it is convenient to assume that the mean ω is drawn from *another* normal distribution with known mean.

The model

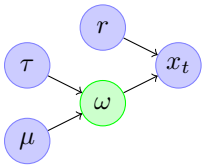


Figure 5.12: Normal with unknown mean, graphical model

- Known precision r.
- Unknown mean ω with assumed distribution $\omega \sim \mathcal{N}(\mu, \tau)$.
- μ, τ are the mean and precision of our belief about ω, respectively.

Theorem 5.7.1. *Let x^t be a random sample from $\mathcal{N}(\omega, r)$. If $\omega \sim \xi_0 = \mathcal{N}(\mu, \tau)$, then $\xi_t = \mathcal{N}(\mu', \tau')$, with*

$$\mu' = \frac{\tau\mu + nr\bar{x}_t}{\tau'}, \qquad \tau' = \tau + nr, \tag{5.7.1}$$

and $\bar{x}_t \triangleq \frac{1}{t}\sum_{k=1}^{t} x_k$.

Gamma distribution

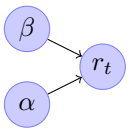


Figure 5.13: Gamma graphical model

The Gamma distribution is a distribution on the interval $[0, \infty)$. It has two parameters that determine the density of the observations.

- Continuous distribution, outcomes $r \in \Omega = [0, \infty)$.
- Parameters $\alpha, \beta > 0$.
- Probability density function: $f(r \mid \alpha) = \frac{\beta^\alpha}{\Gamma(\alpha)} r^{\alpha-1} e^{-\beta r}$, for $r > 0$.
- $\Gamma(\alpha) = \int_0^\infty u^{\alpha-1} e^{-u} \, \mathrm{d}u$.

Relations to other distributions

For $\alpha = 1$, $\beta > 0$ one obtains an *exponential* distribution with parameter β.

For $n \in \mathbb{N}$ and $\alpha = n/2$, $\beta = 1/2$ one obtains a χ^2 distribution with n degrees of freedom. Also, if x^n are i.i.d. standard normal, then $\sum_{k=1}^{n} x_t^2$ has a χ^2 distribution with n-degrees of freedom.

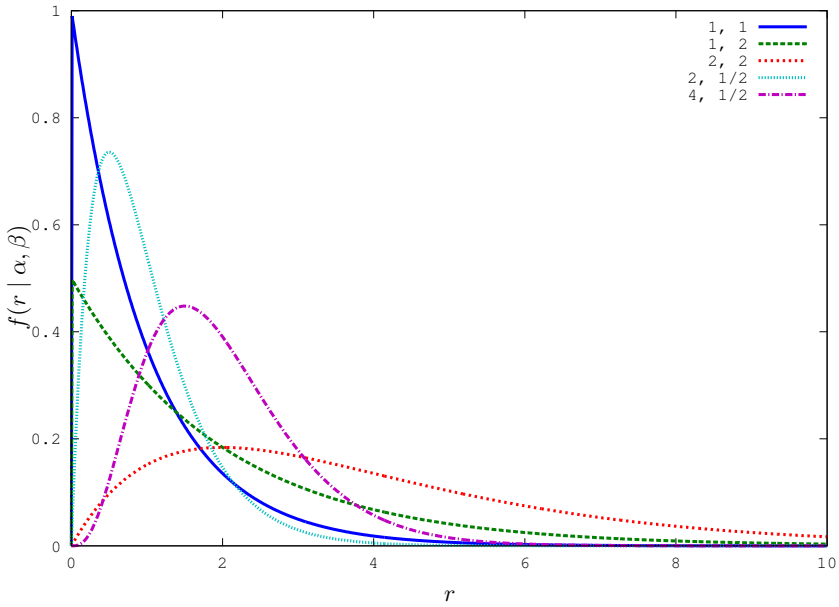


Figure 5.14: Example Gamma densities

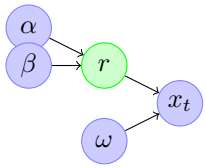


Figure 5.15: Normal with unknown precision, graphical model

Gamma prior for the precision

Theorem 5.7.2. *Given a sample x^n from a normal distribution with mean ω and unknown precision with Gamma prior $\xi_0(r) \triangleq f(r \mid \alpha, \beta)$, the posterior distribution is*

$$\xi_n(r) \triangleq \xi_0(r \mid x^n) = f(r \mid \alpha', \beta'), \tag{5.7.2}$$

where $\alpha' = \alpha + \frac{n}{2}$, $\beta' = \beta + \frac{1}{2}\sum_{i=1}^{n}(x_i - \omega)^2$.

Normals with unknown precision and unknown mean

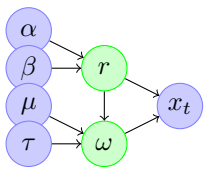


Figure 5.16: Normal with unknown mean and precision, graphical model

Theorem 5.7.3. *Given a sample x^n from a normal distribution with unknown mean ω and precision r, whose prior joint distribution satisfies*

$$\omega \mid r \sim \mathcal{N}(\mu, \tau r), \qquad r \sim \mathcal{G}am(\alpha, \beta), \tag{5.7.3}$$

the posterior distribution is

$$\omega \mid r \sim \mathcal{N}\left(\frac{\tau\mu + n\bar{x}}{\tau + n}, (\tau + n)r\right), \quad r \sim \mathcal{G}am\left(\alpha + \frac{n}{2}, \beta + \frac{1}{2}\sum_{i=1}^{n}(x_i - \bar{x})^2 + \frac{\tau n(\bar{x} - \mu)^2}{2(\tau + n)}\right). \tag{5.7.4}$$

Interesting properties

1. While $\omega \mid r$ has normal distribution, the marginal distribution of ω is not normal. In fact, it can be shown that it has a t-distribution.
2. There is a dependence of ω on r. This is true for our posteriors, even if ω, r are independent in our prior.

The marginal predictive distribution

For a normal distribution with mean m, precision r, we have

$$f(x \mid m, r) \propto r^{1/2} \exp\left(-\frac{r}{2}(m - x)^2\right).$$

For a prior $M|R = r \sim \mathcal{N}(\mu, \nu r)$ and $R \sim \mathcal{G}am(\alpha, \beta)$, we have the following joint distribution for the mean and precision:

$$\xi(m, r) \propto r^{1/2} e^{-(\nu r/2)(m-\mu)^2} r^{\alpha-1} e^{-\beta r}. \tag{5.7.5}$$

Now we can write the posterior marginal as

$$\begin{aligned}
\xi(x) &= \int f(x \mid m, r)\, \mathrm{d}\xi(m, r) && (5.7.6)\\
&\propto \int r e^{-\frac{r}{2}(m-x)^2} e^{-(\nu r/2)(m-\mu)^2} r^{\alpha-1} e^{-\beta r}\, \mathrm{d}(m, r) && (5.7.7)\\
&= \int r^{\alpha} e^{-\beta r} \int e^{-\frac{r}{2}(m-x)^2 - (\nu r/2)(m-\mu)^2}\, \mathrm{d}(m, r) && (5.7.8)\\
&= \int r^{\alpha} e^{-\beta r} \left(\int_{-\infty}^{\infty} e^{-\frac{r}{2}\left[(m-x)^2 + \nu(m-\mu)^2\right]}\, \mathrm{d}m\right) \mathrm{d}r && (5.7.9)\\
&= \int r^{\alpha} e^{-\beta r} e^{-\frac{\nu r}{2(\nu+1)}(\mu-x)^2} \sqrt{\frac{2\pi}{r(1+\nu)}}\, \mathrm{d}r && (5.7.10)
\end{aligned}$$

5.7.2 Conjugates for multivariate distributions

The binomial distribution can be extended to the multinomial and the normal distribution on the real line can be extended to the multivariate normal distribution. Fortunately, multivariate extensions exist for their corresponding conjugate priors as well.

Multinomial-Dirichlet conjugates

Multinomial distribution

The multinomial distribution is the extension of the binomial distribution to more than an arbitrary number of outcomes. Consider an outcome set $S = \{1, \dots, K\}$. This is a common model for independent random trials with a finite number of possible outcomes, such as repeated dice throws, multi-class classification problems, etc.

We now perform n trials, such that the outcome of each trial is independent of the rest. This is the an extension of a sequence of n Bernoulli trials, but with a larger set of possible outcomes in each trial.

The *multinomial* distribution gives us the probability of obtaining the i-th outcome n_i times, given that we perform a total of n trials.

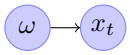


Figure 5.17: Multinomial graphical model

- Discrete distribution, outcomes $x_t \in S = \{1, \dots, K\}$.
- Parameter $\omega \in \mathbb{R}_+$, $\|\omega\|_1 = 1$. Due to the second constraint, given $\omega_1, \dots, \omega_{K-1}$, the value of ω_K is fully determined. However, we shall not make use of that fact.
- This is a distribution with i.i.d outcomes: $\mathbb{P}(x_t = i \mid \omega) = \omega_i$ for all i. Note that this is the multiple outcome extension of the Bernoulli distribution, which only has two possible outcomes.
- Let us denote the number of times the i-th outcome was observed until time t by $n_{t,i} \triangleq \sum_{k=1}^{t} \mathbb{I}\{x_k = i\}$. Then:

$$\mathbb{P}(n_t \mid \omega) = \frac{t!}{\prod_{i=1}^{K} n_{t,i}!} \prod_{i=1}^{K} \omega_i^{n_{t,i}}. \tag{5.7.11}$$

Dirichlet distribution

The Dirichlet distribution is the multivariate extension of the Beta distribution.

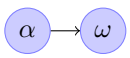


Figure 5.18: Dirichlet graphical model

The Dirichlet distribution is a distribution on the interval $[0, 1]^K$. It has a vector parameter that determines the density of the observations.

- Continuous distribution, outcomes $\omega \in \Omega = \mathbb{\Delta}^K$, i.e. $\|\omega\|_1 = 1$ and $\omega_i \geq 0$. In other words, all of the mass is on the positive $K-1$ dimensional simplex in $\mathbb{R}^K$.

- Parameter vector $\alpha \in \mathbb{R}_+^K$.
- Probability density function:

$$f(\omega \mid \alpha) = \frac{\Gamma(\sum_{i=1}^K \alpha_i)}{\prod_{i=1}^K \Gamma(\alpha_i)} \prod \omega_i^{\alpha_i - 1}, \tag{5.7.12}$$

  $\Gamma(x) = \int_0^\infty u^{x-1} e^{-u} \, \mathrm{d}u$.

Dirichlet prior for multinomial distributions

Figure 5.19: Dirichlet-multinomial graphical model.

- We assume ω is unknown, but fixed.
- We observe $x^t = (x_1, \ldots, x_t)$.
- Our prior is determined by $\mathcal{D}ir(\alpha)$: $\xi_0(\omega) \triangleq f(\omega \mid \alpha)$
- Our posterior is

$$\xi_t(\omega) \propto \prod_{i=1}^K \omega_i^{n_{t,i} + \alpha_i - 1} \tag{5.7.13}$$

  where $n_{t,i} = \sum_{k=1}^t \mathbb{I}\{x_k = i\}$.

Multivariate normal conjugate families

Mutltivariate normal distribution

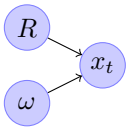


Figure 5.20: Multivariate normal graphical model

- Continuous distribution, outcomes: $S = \mathbb{R}^K$.
- Parameters: mean $\omega \in \mathbb{R}^K$, precision $\boldsymbol{R} \in \mathbb{R}^{K \times K}$, with $x' \boldsymbol{R} x > 0$ for any $x \neq 0$.
- Probability density function:

$$f(x_t \mid w, r) = (2\pi)^{-K/2} |\boldsymbol{R}|^{1/2} \exp\left(-\frac{1}{2} (x_t - \omega)' \boldsymbol{R} (x_t - \omega) \right). \tag{5.7.14}$$

- Independence: $f(x^t \mid \omega, \boldsymbol{R}) = \prod_{k=1}^t f(x_k \mid \omega, \boldsymbol{R})$.

Wishart distribution

The Wishart distribution is a *matrix distribution* on $\mathbb{R}^{K\times K}$ with n degrees of freedom and precision matrix $\boldsymbol{T} \in \mathbb{R}^{K\times K}$.

Construction of the Wishart distribution

1. $x^n \sim \mathcal{N}(\omega, \boldsymbol{T})$, $\omega \in \mathbb{R}^K$, $\boldsymbol{R} \in \mathbb{R}^{K,K}$.
2. $\bar{x}_t \triangleq \frac{1}{n}\sum_{i=1}^n x_i$.
3. $\boldsymbol{S} = \sum_{i=1}^n (x_i - \bar{x}_t)(x_i - \bar{x}_t)'$.

Then $\boldsymbol{S}$ has a Wishart distribution with $n-1$ degrees of freedom and parameter matrix $\boldsymbol{T}$.

Probability density function

For any $\boldsymbol{V} \in \mathbb{R}^{K\times K}$, with $\boldsymbol{V} > 0$.

$$f(\boldsymbol{V} \mid n, \boldsymbol{T}) \propto |\boldsymbol{T}|^{n/2}|\boldsymbol{V}|^{(n-K-1)/2} e^{-\frac{1}{2}\operatorname{trace}(\boldsymbol{T}\boldsymbol{V})}. \tag{5.7.15}$$

is the density for a Wishart distribution with $n > K-1$ degrees of freedom and precision matrix $\boldsymbol{T}$.

Definition 5.7.1. *The trace of a $n \times n$ square matrix A is*

$$\operatorname{trace}(A) \triangleq \sum_{i=1}^n a_{ii}.$$

Normal-Wishart conjugate prior

Theorem 5.7.4. *Given a sample x^n from a multivariate normal distribution in $\mathbb{R}^K$ with unknown mean $\omega \in \mathbb{R}^K$ and precision $\boldsymbol{R} \in \mathbb{R}^{K\times K}$, whose prior joint distribution satisfies:*

$$\omega \mid \boldsymbol{R} \sim \mathcal{N}(\mu, \tau\boldsymbol{R}), \qquad \boldsymbol{R} \sim \mathcal{W}\!\mathit{ish}(\alpha, \boldsymbol{T}), \tag{5.7.16}$$

with $\tau > 0$, $\alpha > k-1$, $\boldsymbol{T} > 0$, the posterior distribution is

$$\omega \mid \boldsymbol{R} \sim \mathcal{N}\left(\frac{\tau\mu + n\bar{x}}{\tau+n}, (\tau+n)\boldsymbol{R}\right), \tag{5.7.17}$$

$$\boldsymbol{R} \sim \mathcal{W}\!\mathit{ish}\left(\alpha+n, \boldsymbol{T} + \boldsymbol{S} + \frac{\tau n}{\tau+n}(\mu-\bar{x})(\mu-\bar{x})'.\right), \tag{5.7.18}$$

$\boldsymbol{S} = \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})'$.

5.8 Summary

- Family of observation distributions: $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$ on $\mathcal{S}$.

- Family of parameter priors Ξ on Ω.
- A *statistic* T is a function giving a *summary* of observations in $\mathcal{S}$.
- A *sufficient* statistic T summarises *all* relevant information for distinguishing between distributions in a family.
- Ξ is *conjugate* to $\mathscr{P}$ if $\xi(\cdot \mid x) \in \Xi$ for all $\xi \in \Xi, x \in \mathcal{S}$.
- A credible interval/set A has a measure $\xi(A \mid x)$, which represents our belief that the true parameter ω is in A, given observations $x \in \mathcal{S}$ and our prior belief $\xi \in \Xi$.
- A confidence interval of measure $\xi(A \mid x) = s$ fails with probability $1 - s$ if $\omega \sim \xi$.
- The beta distribution is good for modelling parameters / observations in $[0, 1]$.
- The gamma distribution is good for modelling parameters / observations in $[0, \infty)$.
- The Dirichlet and Wishart distributions are their multivariate extensions.
- Parametric conjugate pairs: Binomial/Beta, Normal/Normal-Gamma, Multinomial/Dirichlet, Multivariate normal / Normal-Wishart, Uniform/Pareto.

Chapter 6

Hypothesis testing

6.1 Decision problems

Hypothesis testing as a decision problem.

Hypothesis testing is a special type of decision problem, where the decision space $\mathcal{D}$ is a set of hypotheses about the distribution that generates the observations.

Observations

Consider a problem where we obtain an observation $x \in \mathcal{S}$. $\mathcal{S}$ can be a finite product space: $\mathcal{S} = \mathcal{Z}^n$, or it can be the space of all sequences of some observations $z_t \in \mathcal{Z}$: $\mathcal{S} = \mathcal{Z}^* \triangleq \bigcup_{k=1}^{\infty} \mathcal{Z}^k$.

Models

We have a set of probability measures on $\mathcal{S}$, $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$, indexed by ω. We wish to choose from a set of different hypotheses about ω. The measures P_ω do not necessarily have to be within the same parametric family. They must however define measures on CX.

Decision space $\mathcal{D}$

Each decision $d \in \mathcal{D}$ corresponds to a *hypothesis* about which is the correct ω. More specifically, $d \in \mathcal{D}$ corresponds to the hypothesis that $\omega \in \Omega_d$, with $\Omega_d \subset \Omega$ for all d.

Loss function $\ell : \Omega \times D \to \mathbb{R}$

If ω^* is the true parameter, then the loss of decision d is $\ell(\omega^*, d)$.

Decision function.

Definition 6.1.1. *A decision function $\delta : \mathcal{S} \to \mathcal{D}$ selects a decision $\delta(x)$ for any observation $x \in \mathcal{S}$.*

The expected loss of a decision function for a given parameter

If the true parameter is ω^*, the expected loss of a decision function is

$$\sigma(\omega^*, \delta) \triangleq \mathbb{E}_{\omega^*}[\ell(\omega^*, \delta)] = \int_{\mathcal{S}} \ell(\omega, \delta(x)) \, \mathrm{d}P_{\omega^*}(x). \tag{6.1.1}$$

This is the risk of the decision function for the case when the value of the parameter is $\omega^* \in \Omega$. Since ω^* is unknown, the risk is also unknown. However, we can calculate the risk functional for all ω.

The (subjective) expected loss (or risk) of a decision function
For *any* probability measure μ on Ω, our expected loss is:

$$\sigma(\mu, \delta) = \int_{\Omega} \sigma(\omega, \delta)\, \mathrm{d}\mu(\omega) = \int_{\Omega} \int_{\mathcal{S}} [\ell(\omega, \delta(x))]\, \mathrm{d}\mu(\omega). \tag{6.1.2}$$

This holds both when $\mu = \psi$, the distribution from which ω is actually drawn, or when $\mu = \xi$, our prior belief. Thus, the meaning of the above equation depends on what ξ is. If ω is drawn from ξ prior to our experiment, then (6.1.2) coincides with the true risk. If ξ is our subjective belief about ω, but ω is drawn from some other distribution ψ prior to our experiment, then the above expression is only subjective, since generally $\rho(\xi, \delta) \neq \rho(\psi, \delta)$ if $\psi \neq \xi$.

Decision problems with two points

To make the above discussion more concrete, consider the following simple problem, where $\Omega = \{\omega_1, \omega_2\}$ and where we have two decisions, i.e. $D = \{d_1, d_2\}$. In *simple hypothesis testing* problems, each decision d_i corresponds to selecting ω_i as the true parameter and we only suffer a loss when we have made the wrong choice. The loss function is shown in Table 6.1.

$\ell(\omega, d)$	d_1	d_2
ω_1	0	c_1
ω_2	c_2	0

Table 6.1: Cost function of a simple hypothesis testing problem

Consider that we want to decide whether ω_1 or ω_2 is true. We can do so after observing x. We construct a decision function δ, such that for any observed value, we choose d_1, or d_2. For any selected decision function, there will be some probability that we make the wrong decision. Thus, the decision function will have an expected cost associated with it. In fact, let $\alpha_1(\delta), \alpha_2(\delta)$ be the probabilities of making the wrong decision using δ for the two cases, $\omega^* = \omega_1$ and $\omega^* = \omega_2$. Given a probability for ω, we can calculate the risk of our decision function. More specifically:

- Let $\mu(\omega)$ be some (either true or subjective) prior on ω.
- For each $\omega \in \Omega$, we define $\mathbb{P}(x \mid w)$.
- We observe x and then choose decision $\delta(x)$.
- Let $\alpha_1(\delta)$ be the conditional probability that we choose d_2 when $\omega^* = \omega_1$.

$$\alpha_1(\delta) \triangleq \mathbb{P}(\delta(x) \neq d_1 \mid \omega_1)$$

- Let $\alpha_2(\delta)$ be the conditional probability that we choose d_1 when $\omega^* = \omega_2$.

$$\alpha_2(\delta) \triangleq \mathbb{P}(\delta(x) \neq d_2 \mid \omega_2)$$

- Let $b_1 \triangleq c_1 \mu(\omega_1)$ and $b_2 \triangleq c_2 \mu(\omega_2)$.

The risk of δ is:

$$\rho(\mu, \delta) = b_1\alpha_1(\delta) + b_2\alpha_2(\delta) \tag{6.1.3}$$

Theorem 6.1.1 (Neymann-Pearson lemma)**.** *For any $b_1, b_2 > 0$, let δ^* be a decision function such that*

$$\delta^*(x) = d_1, \qquad \text{if } b_1 P_{\omega_1}(x) > b_2 P_{\omega_2}(x) \tag{6.1.4}$$

$$\delta^*(x) = d_2, \qquad \text{if } b_1 P_{\omega_1}(x) < b_2 P_{\omega_2}(x), \tag{6.1.5}$$

and either of d_1, d_2 otherwise. Then, for any other δ:

$$\sigma(\mu, \delta^*) = b_1\alpha_1(\delta^*) + b_2\alpha_2(\delta^*) \leq b_1\alpha(\delta) + b_2\alpha_2(\delta) = \sigma(\mu, \delta)$$

Interpretation.

- When $\mu = \psi$, the distribution from which ω^* is drawn, $\rho(\mu, \delta)$ is the *actual* risk.
- When $\mu = \xi$, our subjective prior belief about ω^*, then $\rho(\mu, \delta)$ is the *subjective* risk.

Bayesian decisions

Construction of the decision function

We saw in chapter 3 (Decision problems) that the optimal decision function is the one that minimises the risk relative to the current posterior. So, we can have the following procedure.

1. Given a family $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$, decision space $\mathcal{D}$, loss function $\ell : \Omega \times D \to \mathbb{R}$.
2. Select a prior $\xi(\omega)$.
3. Observe x, and obtain posterior $\xi'(\omega) \triangleq \xi(\omega \mid x) \propto P_\omega(x)\xi(\omega)$.
4. Select $d^* = \arg\min_d \sigma(\xi(\omega), d)$.

Thus, the optimal decision function is

$$\delta^*(x) = \arg\min_d \sigma(\xi(\omega|x), d).$$

Open problems

- How do we select ξ? Ideally we would like to be able to select ξ in an objective way. However, this is usually not possible. What is done instead, is to select a ξ with certain symmetry or invariance properties, to select a ξ which is maximally pessimistic in some sense, or to actually attempt to elicit a prior ξ from experts. However, strict frequentists would say that a prior does not even exist in a single experiment.
- What is the true risk? This is a big question. The true risk depends on how fast the posterior distribution converges to the true value and how far it is from it initially. However, in general the true risk is close to the Bayes risk, also because of the convexity property of the Bayes risk.

6.2 Unconditional and conditional errors

Error probabilities

Recall that the risk of decision rule δ under μ for the two-point problem is

$$\sigma(\mu, \delta) = b_1\alpha_1(\delta) + b_2\alpha_2(\delta),$$

where

$$\alpha_i(\delta) \triangleq \mathbb{P}(\delta(x) \neq d_i \mid \omega_i), \qquad b_i = c_i\mu(\omega_i). \tag{6.2.1}$$

Definition 6.2.1 (Unconditional error)**.** *When $c_1 = c_2 = 1$, then $\rho(\mu, \delta)$ is the unconditional error of decision rule δ. In other words,*

$$\rho(\mu, \delta) = \sum_i \mu(\omega_i)\,\mathbb{P}(\delta(x) \neq d_i \mid \omega_i)$$

is the a priori probability *that the rule δ will make the wrong decision, given that ω is drawn from μ. Once more, the above can be interpreted as a* subjective *quantity, if $\mu = \xi$, our belief about ω, or as an* actual *quantity, if $\mu = \psi$, the true distribution of ω.*

Classical decisions

These types of decisions are sometimes called frequentist. However, the decisions we shall talk about here are optimal in a worst-case sense.

Construction of the decision function

1. Given a family $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$, decision space $\mathcal{D}$, loss function $\ell : \Omega \times D \to \mathbb{R}$.
2. We have no information on $\psi(\omega)$, not even a subjective $\xi(\omega)$.

3. Solution: minimise worst-case risk. For any μ:

$$\sigma(\mu, \delta) = \sum_i \mu(\omega_i) c_i \, \mathbb{P}(\delta(x) \neq d_i \mid \omega_i). \tag{6.2.2}$$

$$\leq \sum_i \mu(\omega_i) \max_j c_j \, \mathbb{P}(\delta(x) \neq d_j \mid \omega_j). \tag{6.2.3}$$

$$= \max_i c_i \, \mathbb{P}(\delta(x) \neq d_i \mid \omega_i). \tag{6.2.4}$$

The most powerful test for two-point problems

- Minimise bound by equating $c_1 \, \mathbb{P}(\delta(x) \neq d_1 \mid \omega_1) = c_2 \, \mathbb{P}(\delta(x) \neq d_2 \mid \omega_2)$.
- Find $X \subset \mathcal{S}$ such that $c_1 P_{\omega_1}(S) = c_2 P_{\omega_2}(\mathcal{S} \setminus X)$.
- $\delta(x) = d_2$ if $x \in X$ and d_1 otherwise.
- Then $\mathbb{P}(\delta(x) \neq d_1 \mid \omega_1) = P_{\omega_1}(S)$.

Conditional error

Assume we observe $x \sim w$ and let $\mu(\omega)$ be a prior distribution on Ω. Let us define

$$\mu(\omega \mid x) = P_\omega(x)\mu(\omega)/\mu(x),$$

to be the posterior probability of ω.

Definition 6.2.2 (Conditional error)**.** *Given the above, $\rho(\mu(\cdot \mid x), \delta)$ is the conditional error of decision rule δ if we have observed x. In other words,*

$$\rho(\mu(\cdot \mid x), \delta(x)) = \sum_i \mu(\omega_i \mid x) \, \mathbb{P}(\delta(x) \neq d_i \mid \omega_i)$$

is the posterior probability *that the rule δ will make the wrong decision, given that ω is drawn from μ.*

Once more, the above can be interpreted as a *subjective* quantity, if $\mu = \xi$, our belief about ω, or as an *actual* quantity, if $\mu = \psi$, the true distribution of ω.

Desired properties of reported errors

Assume that we have a procedure which:

1. Given μ, δ and *before* seeing x, gives us an unconditional error estimate α.
2. Given μ, δ and *after* seeing x, gives us a conditional error estimate $\alpha(x)$.

Desirable properties

$$\alpha = \sigma(\psi, \delta) \tag{6.2.5}$$
$$\alpha(x) = \sigma(\psi(\cdot \mid x), \delta(x)). \tag{6.2.6}$$

6.3 Two-point test with Bernoulli trials

Example 6.3.1. *Bernoulli trials*

1. *Take a sample* $x = x_1, \dots, x_n$ *from* $\mathbb{P}(x_k \mid w)$.
2. *Let* $s_n = \sum_{k=1}^{n} x_k$.
3. *From the binomial formula,*

$$\mathbb{P}(s_t = k \mid w) = \binom{t}{k} w^k (1-w)^{t-k}, \quad \binom{x}{k} = \prod_{i=0}^{k-1} (x-i)/(1+i). \tag{6.3.1}$$

Recall that $\binom{x}{k} = \prod_{i=0}^{k-1}(x-i)/(1+i)$.

The problem

- We need to decide wether $d_1 : \omega = \omega_1$, or $d_2 : \omega = \omega_2$.
- Assume that the costs of mistakes are $c_1, c_2 = 1$ for simplicity.
- Use rule δ to select $\delta(x)$ after you observe x.

Bayesian Bernoulli trial test

- Select a prior parameter $\pi \in [0, 1]$, so that $\xi(\omega_1) = \pi = 1 - \xi(\omega_2)$.
- Observe x and calculate:

$$\xi(\omega_1 \mid x) = \frac{\pi \, \mathbb{P}(x \mid \omega_1)}{\pi \, \mathbb{P}(x \mid \omega_1) + (1-\pi) \, \mathbb{P}(x \mid \omega_2)}. \tag{6.3.2}$$

- Construct decision function

$$\delta(x) = \begin{cases} d_1, & \xi(\omega_1 \mid x) > \xi(\omega_2 \mid x) \\ d_2, & \xi(\omega_1 \mid x) < \xi(\omega_2 \mid x) \\ d_1 \text{w.p. } 1/2, & \xi(\omega_1 \mid x) = \xi(\omega_2 \mid x) \end{cases}$$

In other words, this decision function contains a no-decision region of measure 0, corresponding to the case where both posterior probabilities are equal.

Properties

- The procedure has unconditional error $O(e^{-t})$.
- The conditional error is $\max_k \xi(\omega_k) + O(e^{-t})$.

Classical Bernoulli trial test

- Select error rates α_1, α_2.
- These can depend on the sample size: e.g. $\alpha_1(t) = b_1 t^{-1/2}$, $\alpha_2(t) = b_2 t^{-1/2}$.
- Select a subset $S \subset \mathcal{S}$ such that $P_{\omega_1}(S) = \alpha_1$ and $P_{\omega_2}(\mathcal{S} \setminus S) = \alpha_2$, if possible.
- Observe x and use decision function

$$\delta(x) = \begin{cases} d_1, & x \in S \\ d_2, & x \notin S. \end{cases}$$

To construct S, traditionally $\alpha_1 = \alpha_2$ and we choose a *critical value* c such that $\mathbb{P}(s > c \mid \omega_1) = 1 - \mathbb{P}(s < c \mid \omega_2)$, with $s = \sum_k x_k$ being the *test statistic*. Note that $c \approx \frac{t}{2}(\omega_1 + \omega_2)$. However, any arbitrary S can be chosen.

Properties

- The unconditional error is $\max\{\alpha_1, \alpha_2\}$.
- The conditional error is $\alpha_{\delta(x)}$. *Thus, it only depends on whether* $x \in S$.

Summary and additional remarks

- The decision rule is chosen a priori.
- The unconditional error/risk does not depend on the data.
- The conditional error/risk depends on the data.
- Bayesian procedures minimise *expected loss*.
- Classical procedures minimise *worst-case expected loss*.
- The Bayesian procedure should be consistent with classical ones.
- By selecting a *worst-case prior* we can create a Bayesian analogue of classical procedures.

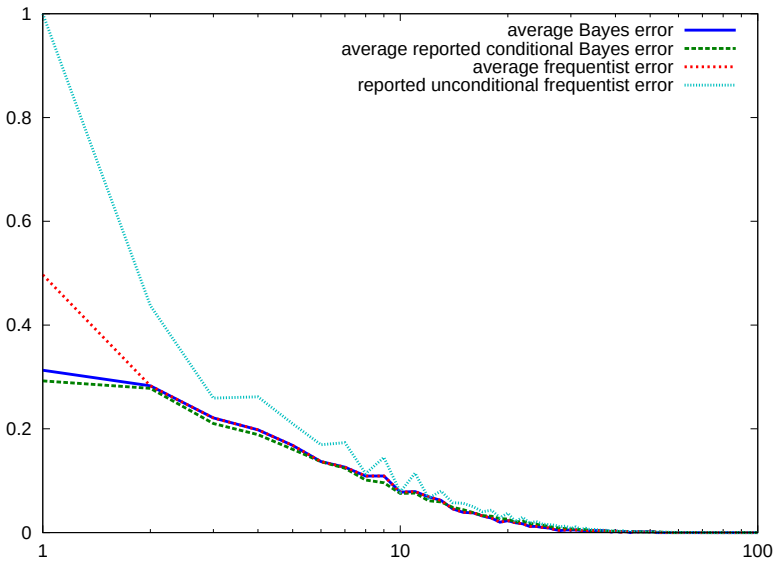


Figure 6.1: $\psi(\omega_1) = 1/2$, $\omega_1 = 1/3$, $\omega_2 = 3/4, \xi(\omega_1) = 1/2$.

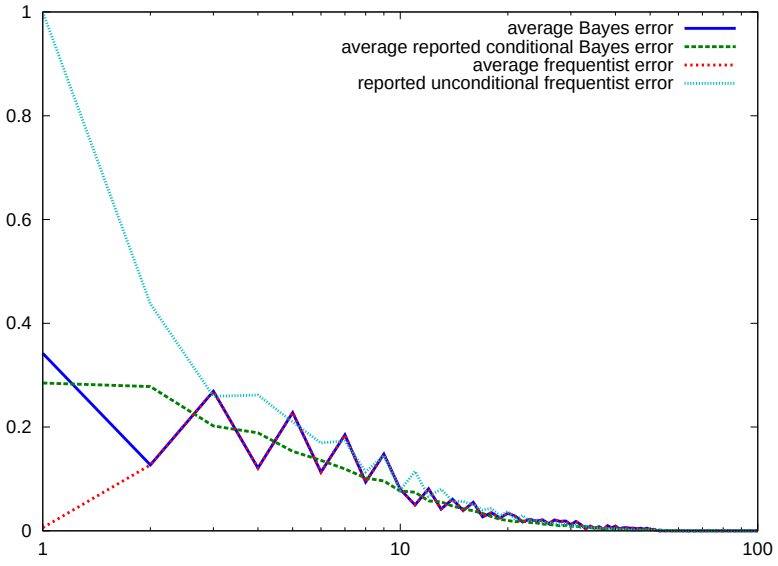
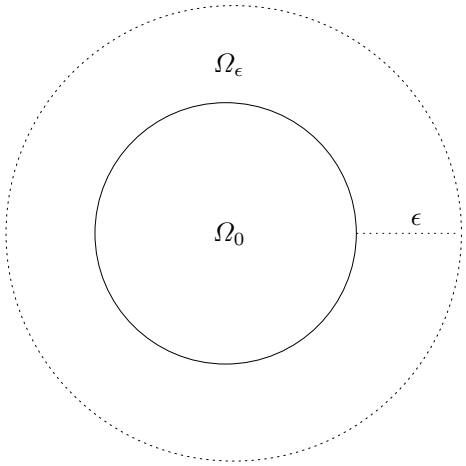


Figure 6.2: $\psi(\omega_1) = 99/100$, $\omega_1 = 1/3$, $\omega_2 = 3/4, \xi(\omega_1) = 1/2$.



6.4 Null hypothesis tests

The problem of null hypothesis tests

- Very frequently we wish to decide whether something is true or not.
- However, we have *no specific alternative hypothesis.*
- Consider for example we want to choose whether: $d_0 : \omega^* \in \Omega_0$, or $d_1 : \omega^* \notin \Omega_0$.
- Most of the time we can only decide $d_0 : \omega^* \in \Omega_0$, or $d_1 : \omega^* \notin \Omega_\epsilon$, with $\Omega_\epsilon \supset \Omega_0$.
- In the simplest case the null hypothesis is a single point: $\Omega_0 = \{\omega_0\}$.

Construction of the null hypothesis test

Consider the following problem. We have a distribution in some family $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$ with unknown parameter ω. We have two hypotheses: $\omega = \omega_0$ and $\omega \neq \omega_0$. After obtaining a sample $x = z_1, \ldots, z_n$ from P_ω, we need to decide for $d_0 \in \mathcal{D} : \omega = \omega_0$ or $d_1 : \omega \neq \omega_0$. The loss of a wrong decision is 1.

A simple decision rule

Let us choose some $S \subset \mathcal{S}$, such that $P_{\omega_0}(S) = \alpha$. We can now define the following decision rule:

$$\delta(x) = \begin{cases} d_0, & \text{if } x \notin S, \\ d_1, & \text{if } x \in S. \end{cases}$$

The unconditional error

This rule has unconditional error α when $\omega = \omega_0$. In other words, the probability that we will make a mistake when employing this rule, and $\omega = \omega_0$, is α. However, there is no guarantee for the error when $\omega \neq \omega_0$, without additional assumptions. In particular, we need to specify μ, and $\mathbb{P}(x \in S \mid \omega \neq \omega_0)$

$$\sigma(\mu, \delta) = \mu(\omega = \omega_0) P_{\omega_0}(S) + \mu(\omega \neq \omega_0)\, \mathbb{P}(x \in S \mid \omega \neq \omega_0), \tag{6.4.1}$$

where μ is either a subjective distribution ξ or the actual distribution ψ.

Bayesian approach

The Bayesian approach once more involves us having a subjective belief about the nature of the problem. If, of course, we somehow know that ω is selected according to some ψ, then we can set $\xi = \psi$, and we will be optimal in expectation.

(Subjective) prior

$$\pi_0 \triangleq \xi(\omega = \omega_0) \tag{6.4.2}$$

$$\phi(Z) \triangleq \xi(\omega \in Z \mid \omega \neq \omega_0), \qquad Z \subset \Omega. \tag{6.4.3}$$

(Subjective) posterior

$$\pi_0(x) \triangleq \xi(\omega = \omega_0 \mid x) = \frac{\pi_0 P_\omega(x)}{\pi_0 P_\omega(x) + (1 - \pi)\phi(x)} \tag{6.4.4}$$

$$\phi(x) = \int_\Omega P_\omega(x)\, \mathrm{d}\phi(\omega). \tag{6.4.5}$$

Example 6.4.1. *Consider Bernoulli trials where we want to test whether $\omega^* = \omega_0$, or $\omega^* \neq \omega_0$. We can use a Beta density with parameters β_0, β_1 for the prior distribution in the case of $\omega^* \neq \omega_0$:*

$$\phi(Z) = \int_Z f(\omega \mid \beta),$$

with

$$f(\omega \mid \beta) = \frac{\Gamma(\beta_0 + \beta_1)}{\Gamma(\beta_0)\Gamma(\beta_1)} w^{\beta_1 - 1} (1 - w)^{\beta_0 - 1}$$

and posterior distribution also being a Beta density with parameters $\beta_0 + t - s_t, \beta_1 + s_t$, with $s_t = \sum_{k=1}^{t} x_k$.

Classical approach

Since we only have one hypothesis, $\omega = \omega_0$, we must *construct* an alternative hypothesis against which we can measure our other error.

Constructing the alternative hypothesis

1. Select $\Omega_\epsilon \subset \Omega$, to be the ϵ-enlargement of Ω_0.
2. When $\Omega_0 = \{\omega_0\}$, $\Omega_\epsilon = \{\omega \in \Omega \mid \|w - \omega_0\| < \epsilon\}$.

We select $d_1 : \omega^* \notin \Omega_\epsilon$ when $x \in S$ and $d_0 : \omega^* = \omega_0$ when $x \notin S$. Then the risk is:

$$\rho(\omega_0, \delta) = P_{\omega_0}(S),$$

and

$$\rho(\Omega \setminus \Omega_\epsilon, \delta) \leq \max\left\{P_\omega(x) \mid \omega \in \Omega \setminus \Omega_\epsilon\right\}.$$

Example 6.4.2. *Consider Bernoulli trials where we want to test whether* $\omega^* = \omega_0$*, or* $\omega^* \neq \omega_0$*.*

The test

- Select an error rate $\alpha_1 > 0$.
- Set $s_t = \sum_{k=1}^{t} x_k$.
- Decision rule, with $c_1 < t\omega_0 < c_2 \in [0, t]$:

$$\delta(x) = \begin{cases} d_0 & \text{if } s_t \in [c_1, c_2], \\ d_1 & \text{otherwise.} \end{cases}$$

Let Q_ω^t be the probability measure on $\mathbb{N}$ arising from the binomial distribution with parameters t, w.

Unconditional error

$$\mathbb{P}(\delta(x) \neq d_0 \mid \omega^* = \omega_0) = 1 - Q_{\omega_0}^t([c_1, c_2]) \tag{6.4.6}$$

$$\mathbb{P}(\delta(x) \neq d_1 \mid |\omega^* - \omega_0| > \epsilon) \leq \max\left\{\mathbb{P}(s_t \in [c1, c2] \mid \omega^* = w) \mid |w - \omega_0| > \epsilon\right\} \tag{6.4.7}$$

To calculate this, we look at two cases. The first case is:

$$\mathbb{P}(s_t \in [c_1, c_2] \mid \omega^* < \omega_0 - \epsilon) \leq \max \left\{ Q_\omega^t([c_1, c_2]) \;\middle|\; \omega \in [0, \omega_0 - \epsilon) \right\}$$

and assuming $\epsilon > \omega_0 - c_1/t$

$$\mathbb{P}(s_t \in [c_1, c_2] \mid \omega^* < \omega_0 - \epsilon) \leq Q_{\omega_0 - \epsilon}^t([c_1, c_2]).$$

The second case is:

$$\begin{aligned} \mathbb{P}(s_t \in [c_1, c_2] \mid \omega^* > \omega_0 + \epsilon) &\leq \max \left\{ Q_\omega^t([c_1, c_2]) \;\middle|\; \omega \in (\omega_0 + \epsilon, 1] \right\} \\ &= Q_{\omega_0 + \epsilon}^t([c_1, c_2]). \end{aligned}$$

where we assumed $\epsilon > c_2/t - \omega_0$.

Consequently, to obtain a test with good guarantees on the ϵ-enlargement of the null hypothesis, we need $c_1 \geq (\omega_0 - \epsilon)t$ and $c_2 \geq (\omega_0 + \epsilon)t$.

6.5 The fallacy of P-values

P-values are frequently misused in applications. In fact, P-values are a side-effect of constructing a statistical test in order for it to have a certain error rate α. They are a *uniformly distributed* random variable when the null-hypothesis is true. Thus, by rejecting the null-hypothesis only when the P-value is smaller than α, you guarantee that the type I error is exactly α. Let us discuss this point further with an example.

P-values in simple two-point tests.

Example 6.5.1.
- *Exponential distribution with parameter ω:* $f_\omega(x) = we^{-wx}$.
- $d_1 : \omega = 1$, $d_2 : \omega = 2$.

A classical test
We observe $x \in \mathbb{R}_+$. We use the decision function

$$\delta(x) = \begin{cases} d_1, & x > c \\ d_2, & x \leq c. \end{cases}$$

These leads to the following unconditional errors

$$\alpha_1 \triangleq \mathbb{P}(\delta(x) \neq d_1 \mid \omega = 1) = \mathbb{P}(x \leq c \mid \omega = 1) = \int_{u=0}^{c} e^{-x} = 1 - e^{-c} \tag{6.5.1}$$

$$\alpha_2 \triangleq \mathbb{P}(\delta(x) \neq d_2 \mid \omega = 2) = \mathbb{P}(x > c \mid \omega = 2) = \int_{u=c}^{\infty} 2e^{-2x} = \frac{1}{2} e^{-2c}. \tag{6.5.2}$$

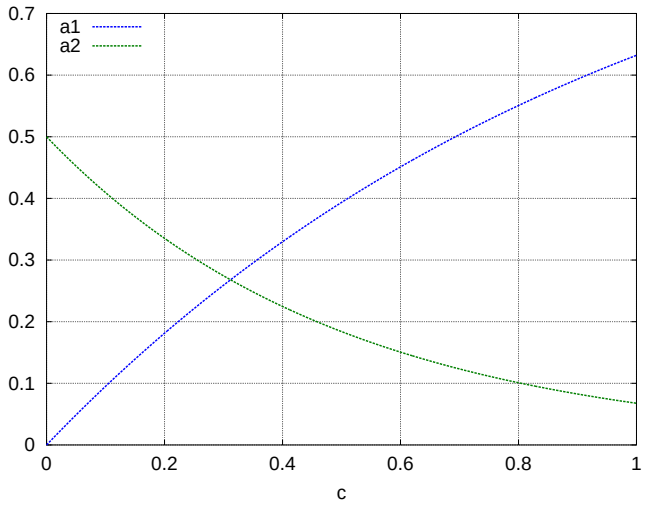


Figure 6.3: Unconditional errors for different critical values

Selecting c

For uniformly bounding the error, set $\alpha_1 = \alpha_2$:

What is a P-value?

Definition 6.5.1. *Assume we want to perform a test with type I error α_1. If $p : \mathcal{S} \to [0, 1]$ is such that:*

$$\mathbb{P}(p(x) \leq \alpha_1 \mid \omega_1) = \alpha_1, \tag{6.5.3}$$

then $p(x)$ is a p-value for observations x under ω_1.

The uniform property of p-values.
$p(x)$ is uniformly distributed in $[0, 1]$ under ω_1

Consequently, $p(x)$ gives *no information* about the validity of ω_1. It also does not tell you how much the observations x might match alternative hypotheses.

Summary

- A test is a decision rule.
- Unconditional / conditional errors are properties of decision rules.
- Null-hypothesis tests need an ϵ-enlargement for guarantees.
- There is always a (possibly negligible) region where no guarantees can be made.

- Classical (frequentist) methods are generally worst-case and unconditional.
- Bayesian (subjective) methods are generally expected-case and conditional.
- P-values are only a tool to implement classical tests and carry no information.

Further reading

- Bayesian methods can be extended to worst-case approaches.
- Classical methods can also report conditional errors.
- Distribution-free methods can be used for more general problems.
- Pre-validation

Chapter 7

Sequential sampling

7.1 Gains from sequential sampling

The idea of sequential sampling

We wish to buy a large number of items in bulk. However, some portion of the items may be defective. We calculate (somehow) that if we test 100 items and 10 or more are faulty, then we should not buy. As testing is expensive, instead of testing all 100 items, we test sequentially. If, at any point, we have 10 faulty items or 91 good items, we can stop testing. This produces a decision of the same quality as testing all 100 items, but at a fraction of the cost.

Sequential sampling

A sequential sample from some unknown distribution P is generated as follows. At time t, we have observed $x_1, \dots, x_t$, generated by P. These are not necessarily independently and identically distributed samples. At any time t, we can either *stop sampling* or obtain one *more* observation x_{t+1}. A sample obtained in this way is called a *sequential sample*. More formally,

Definition 7.1.1. *A sequential sampling procedure on a probability space* $(\mathcal{X}^*, \mathfrak{B}(\mathcal{X}^*), P)$ *involves a decision function* $f : \mathcal{X}^* \to \{0, 1\}$*, such that we stop sampling at time* t *if and only if* $f(x^t) = 1$*, otherwise we obtain a new sample*

$$x_{t+1} \mid x^t \sim P(\cdot \mid x^t).$$

Sampling with costs

We once more consider problems where we have some observations $x_1, x_2, \dots$, with $x_t \in \mathcal{X}$, which are drawn from some distribution with parameter $\omega \in \Omega$, or more precisely from a family $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$, such that each $(\mathcal{X}^*, \mathfrak{B}(\mathcal{X}^*), P_\omega)$ is a probability space for all $\omega \in \Omega$. Once more, we have a prior probability measure ξ on $\mathfrak{B}(\Omega)$ for the unknown parameter, and we wish to take a decision $d \in \mathcal{D}$ that maximises the expected utility according to our utility function $U : \Omega \times \mathcal{D} \to \mathbb{R}$.

In the classical case, we obtain a complete sample of fixed size n, $x^n = (x_1, \dots, x_n)$ and calculate a posterior measure $\xi(\cdot \mid x^n)$. We then take the decision maximising the expected utility according to our posterior.

So far, we have mainly considered decision problems where the sample size was fixed. However, frequently the sample size can also be part of the decision. Since normally larger sample sizes give us more information, in this case the decision problem is only meaningful when obtaining new samples has a cost.

Fixed sample size

Given x^n, find $d \in D$ maximising:

$$\mathbb{E}_\xi(U \mid d, x^n) = \int_\Omega U(w, d) \, \mathrm{d}\xi(\omega \mid x^n). \tag{7.1.1}$$

Samples with costs

Consider now the case where each sample taken incurs a cost c. Then, our utility is

$$\mathbb{E}_\xi(U \mid d, x^n) = \int_\Omega U(w, d) \, \mathrm{d}\xi(\omega \mid x^n) - cn. \tag{7.1.2}$$

Example 7.1.1. *Consider the following decision problem.*

- *Parameters:* $\Omega = \{\omega_1, \omega_2\}$.
- *Decisions:* $D = \{d_1, d_2\}$.
- *Observation distribution* $f_i(k) = \mathbb{P}(x_t = k \mid \omega = \omega_i)$ *for all* t *with*

$$f_1(1) = 1 - \alpha, \qquad f_1(2) = 0, \qquad f_1(3) = \alpha, \tag{7.1.3}$$
$$f_2(1) = 0, \qquad f_2(2) = 1 - \alpha, \qquad f_2(3) = \alpha. \tag{7.1.4}$$

- *Utility:* $U(\omega_i, d_j) = 0$, *for* $i = j$ *and* $b < 0$ *otherwise.*
- *Prior:* $\mathbb{P}(\omega = \omega_1) = \xi = 1 - \mathbb{P}(\omega = \omega_2)$.

In this problem, it is immediately possible to distinguish f_1 from f_2 when you observe $x_t = 1$ or $x_t = 2$. However, the values $x_t = 3$ provide no information.

- So, the expected utility of stopping if you have only observed 3s is ξb.
- Otherwise, it equals 0.

7.1.1 An example sequential problem

A procedure taking n observations

- The probability of observing $x_t = 3$ for all $t = 1, \ldots, n$ is α^n.
- The total value $V(n)$ of the optimal procedure taking n observations is

$$V(n) = \xi b \alpha^n - cn. \tag{7.1.5}$$

We can find the optimal value more or less easily. Since V is a nice function, an approximate minimiser can be found by viewing n as a continuous variable. Taking derivatives:

$$n^* = \left[\log \frac{c}{\xi b \log \alpha} \right] \frac{1}{\log \alpha} \tag{7.1.6}$$

$$V(n^*) = \frac{c}{\log \alpha} \left[1 + \log \frac{c}{\xi b \log \alpha} \right] \tag{7.1.7}$$

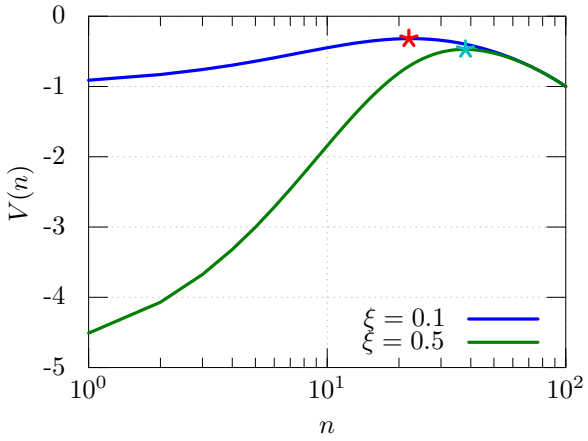


Figure 7.1: The value of taking exactly n observations under two different beliefs, for $\alpha = 0.9$, $b = -10$, $c = 10^{-2}$.

Definition 7.1.2 (The geometric series)**.** *The sum $\sum_{k=0}^{n} x^k$ is called the geometric series and has the property*

$$\sum_{k=0}^{n} x^k = \frac{x^{n+1} - 1}{x - 1}. \tag{7.1.8}$$

Taking derivatives with respect to x can result in other useful formulae.

A sequential procedure stopping after at most n^* steps.

- If $t < n^*$, use the stopping rule $s(x^t) = 1$ iff $x_t = 3$.
- In other words, stop as soon as you observe a 3, or until you reach $t = n^*$.
- Our posterior after stopping is, in this case, the same as in the standard procedure.
- However the number of observations n is random.

Since the probability of $x_t = 3$ is always the same for both ω_1 and ω_2, we have:

$$\mathbb{E}(n) = \mathbb{E}(n \mid \omega = \omega_1) = \mathbb{E}(n \mid \omega = \omega_2) < n^*$$

We can calculate the expected number of steps as follows:

$$\mathbb{E}(n) = \mathbb{E}(n \mid \omega = \omega_1) = \sum_{t=1}^{n^*} t\, \mathbb{P}(n = t \mid \omega = \omega_1) \tag{7.1.9}$$

$$= \sum_{t=1}^{n^*-1} t\alpha^{t-1}(1-\alpha) + n^*\alpha^{n^*-1} = \frac{1 - \alpha^{n^*}}{1 - \alpha}, \tag{7.1.10}$$

from the *geometric series*. The value of this procedure is:

$$\bar{V} = \xi b \alpha^{n^*} - c\,\mathbb{E}(n) \tag{7.1.11}$$

and from the definition of n^*:

$$\bar{V} = \frac{c}{\alpha - 1} + \frac{c}{\log \alpha}\left[1 + \frac{c}{\xi b(1-\alpha)}\right]. \tag{7.1.12}$$

To simplify this, note that $e^x \geq 1 + x \Rightarrow x \geq \log(1+x) \Rightarrow \alpha - 1 \geq \log(\alpha) \Rightarrow \frac{1}{\alpha-1} \leq \frac{1}{\log(\alpha)}$.

An unbounded sequential procedure

- Use stopping rule $s(x^t) = 1$ iff $x_t = 3$.
- In other words, stop as soon as you observe a 3.

Once we observe $x_t = 3$, we can make a decision that has value 0. So, the value of the unbounded sequential procedure is just

$$-c\,\mathbb{E}(n).$$

$$\mathbb{E}(n) = \sum_{t=1}^{\infty} t\,\mathbb{P}(n = t \mid \omega = \omega_1) \tag{7.1.13}$$

$$= \sum_{t=1}^{\infty} t\alpha^{t-1}(1-\alpha) = \frac{1}{1-\alpha}. \tag{7.1.14}$$

The value of this procedure is:

$$\bar{V} = \xi b \alpha^{n^*} - c\,\mathbb{E}(n) \tag{7.1.15}$$

and from the definition of n^*:

$$\bar{V} = \frac{c}{\alpha - 1} + \frac{c}{\log \alpha}\left[1 + \frac{c}{\xi b(1-\alpha)}\right]. \tag{7.1.16}$$

To simplify this, note that $e^x \geq 1 + x \Rightarrow x \geq \log(1+x) \Rightarrow \alpha - 1 \geq \log(\alpha) \Rightarrow \frac{1}{\alpha-1} \leq \frac{1}{\log(\alpha)}$.

Summary

- Bounded procedures are (in expectation) better than fixed-sampling procedures.
- Unbounded procedures are (in expectation) better than bounded procedures.
- An unbounded procedure may end up costing much more than taking a decision without observing any data.
- Essentially, with an unbounded procedure, we disregard the amount spent to time t.

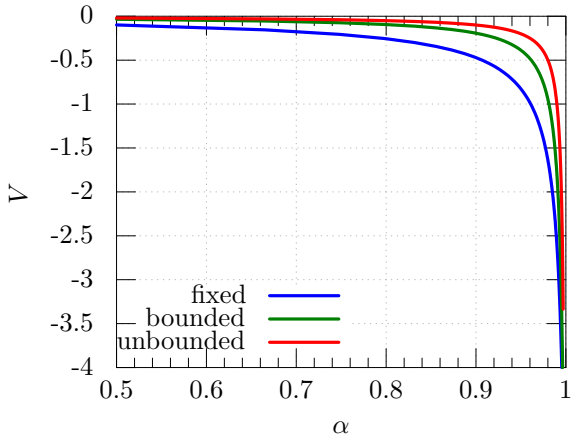


Figure 7.2: The value of three strategies for $\xi = 1/2$, $b = -10$, $c = 10^{-2}$ and varying α. Higher values of α imply a longer time before the true ω is known.

7.2 Sequential decision procedures

Definition 7.2.1 (Sequential decision procedure)**.** *A sequential decision procedure $\delta = (s, d)$ is tuple composed of*

1. *A stopping rule $s : \mathcal{X}^* \to \{0, 1\}$.*
2. *A decision rule $d : \mathcal{X}^* \to D$.*

Stopping rule

The stopping rule s specifies whether, at any given time, we should stop and make a decision in $\mathcal{D}$ or take one more sample. That is, if

$$s(x^t) = 1,$$

stop, otherwise observe x_{t+1}.

Decision rule

Once we have stopped (i.e. $s(x^t) = 1$), we choose the decision

$$d(x^t).$$

Deterministic stopping rules

If the stopping rule s is deterministic, then, for any t, there exists some $B_t \subset \mathcal{X}^t$ such that

$$s(x^t) = \begin{cases} 1, & \forall x^t \in B_t \\ 0, & \forall x^t \notin B_t. \end{cases} \tag{7.2.1}$$

Cylinder sets
For $r > t$, B_t can be also regarded as a subset of S_r: If $x^r \in B_t$, and y^r is such that $y_i = x_i$ for $i = 1, \dots, t$, then $y^r \in B_t$.

7.3 Calculating the expected utility of a sequential decision procedure

Once more, consider a distribution family $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$ and a prior ξ over $\mathfrak{B}(\Omega)$. For a decision set $\mathcal{D}$, a utility function $U : \Omega \times \mathcal{D} \to \mathbb{R}$, and a sampling cost c, the risk of a sequential decision procedure is the expected decision utility minus the expected sampling cost.

$$U(\xi, \delta) = \mathbb{E}_\xi \left\{U[\omega, \delta(x^n)] - nc\right\} \tag{7.3.1}$$

$$= \sum_{n=1}^{\infty} \int_{B_n} \mathbb{E}_\xi[U(\omega, \delta(x^n)) \mid x^n] \, \mathrm{d}P(x^n \mid \xi) - \sum_{n=1}^{\infty} \mathbb{P}(B_n \mid \xi) nc \tag{7.3.2}$$

$$= \sum_{n=1}^{\infty} \int_{B_n} \left\{ \int_\Omega U[\omega, d(x^n)] \, \mathrm{d}\xi(\omega \mid x^n) \right\} \mathrm{d}P(x^n \mid \xi) - \sum_{n=1}^{\infty} \mathbb{P}(B_n \mid \xi) nc. \tag{7.3.3}$$

Although it may seem difficult to evaluate this, it can be done by a simple dynamic programming technique called *backwards induction*.

Definition 7.3.1 (Bounded sequential decision procedure)**.** *A sequential decision procedure δ is bounded if there is a positive integer H such that $\mathbb{P}(n \leq H) = 1$.*

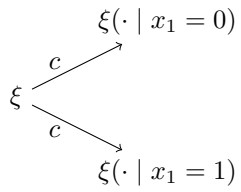
- If δ is H-bounded, then we shall take at most H samples.
- At stage H, we will have observed some sequence x^H, which gives rise to a posterior $\xi(\omega \mid x^H)$.
- Since we *must* stop at H, we must choose a decision d maximising

$$\mathbb{E}_\xi[U(\omega, d) \mid x^H] = \int_\Omega U(\omega, d) \, \mathrm{d}\xi(\omega \mid x^H)$$

- The respective *value* (expected utility) is

$$V_0[\xi(\cdot|x^H)] \triangleq \sup_{d \in \mathcal{D}} \mathbb{E}_\xi[U(\omega, d) \mid x^H],$$

  where V_0 denotes the decision's expected utility.
- What about the previous step?



A simple two-stage problem

Consider $\mathcal{X} = \{0, 1\}$ and a prior ξ on the ω parameter of $\mathcal{Bern}(w)$. We wish to either decide now on a parameter ω, or take one more observation, at cost c, before deciding. Thus, the problem has two stages.

- *We begin with a prior* ξ at the first stage. There are two possible outcomes for the **second stage**:
  1. If we have observed $x_1 = 0$: then our value is $V_0[\xi(\cdot \mid x_1 = 0)]$.
  2. If we have observed $x_1 = 1$: then our value is $V_0[\xi(\cdot \mid x_1 = 1)]$.
- At the first stage,
  1. Stop with value $V_0(\xi)$.
  2. Pay cost c for value:

$$V_0[\xi(\cdot \mid x_1)], \qquad \text{with } P_\xi(x_1) = \int_\Omega P_\omega(x_1) \,\mathrm{d}\xi(w)$$

  So the expected of continuing for one more step is

$$V_1(\xi) \triangleq \int_{\mathcal{X}} V_0[\xi(\cdot \mid x_1)] \,\mathrm{d}P_\xi(x_1).$$

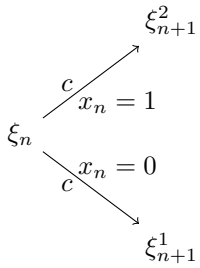
- Thus, the overall value for this problem is:

$$\max \left\{ V_o(\xi), \sum_{x_1=0}^{1} V_o[\xi(\cdot \mid x_1)] P_\xi(x_1) - c. \right\}$$

The above is simply the maximum between the value of stopping immediately, and the value of continuing for one more step. This procedure can be applied recursively for multi-stage problems.

Multi-stage problems

$$\begin{aligned}
V_0(\phi) &= \sup_{d \in \mathcal{D}} \int_\Omega U(\omega, d) \,\mathrm{d}\phi(\omega) && \text{(Immediate value)} \\
\xi_{n+1}(\cdot) &= \xi_n(\cdot \mid x_n) = \xi(\cdot \mid x^n) && \text{(posterior)} \\
\mathbb{E}_{\xi_n} V_0[\xi_{n+1}] &= \int_{\mathcal{X}} V_0[\xi_n(\cdot \mid x_n)] \,\mathrm{d}\xi_n(x_n) && \text{(Next-step value)} \\
\sigma_1(\xi_n) &= \min \left\{ V_0(\xi_n), \mathbb{E}_{\xi_n} V_0(\xi_{n+1}) - c. \right\}. && (7.3.4)
\end{aligned}$$



7.4 Backwards induction

The main idea expressed in the previous section is to start from the last stage of our decision problem, where the utility is known, and then move backwards. At each stage, we know the probability of reaching different points in the next stage, as well as their values. Consequently, we can compute the value of any point in the current stage as well. This notion is formalised below.

Theorem 7.4.1 (Backwards induction)**.** *The utility of a k-bounded optimal procedure with prior ξ_0 is $V_k(\xi_0)$ and is given by the recursion:*

$$V_{j+1}(\xi_n) = \max \left\{ V_0(\xi_n), \mathbb{E}_{\xi_n} V_j(\xi_{n+1}) - c \right\}. \tag{7.4.1}$$

After observing x^{k-j}, the value of the optimal continuation is $V_j(\xi_{k-j})$.

The above theorem essentially gives a recursive calculation of the risk of the k-bounded optimal procedure. In fact, that procedure will have the following property.

Theorem 7.4.2. *The optimal k-bounded procedure stops at time t if*

$$V_0(\xi_t) \geq V_{k-t}(\xi_t)$$

and chooses d maximising $\mathbb{E}_{\xi_t} U(\omega, d)$, otherwise takes one more sample.

Finally, it is always useful to look ahead one more step, as shown by the following theorem.

Theorem 7.4.3. *For any probability measure ξ on Ω,*

$$V_n(\xi) \leq V_{n+1}(\xi). \tag{7.4.2}$$

7.5 Unbounded sequential decision procedures

Given the monotonicity of the value of bounded procedures (7.4.2), one may well ask what is the value of unbounded procedures.

The value of unbounded sequential decision procedures

- Let δ be an unbounded procedure.

- The value of the procedure is

$$V(\xi, \delta) \triangleq \mathbb{E}_\xi \left\{ V_0[\xi(\cdot \mid x^n)] - cn \right\}, \tag{7.5.1}$$

  where n is the random number of samples taken by δ. This is random because the observations x are random; δ itself can be deterministic.

Definition 7.5.1. *Let $B_{>k} \subset \mathcal{X}^*$ be the set of sequences such that δ takes more than k samples. Then δ is regular if $V(\xi, \delta) \geq V_0(\xi)$ and if, for all $x^n \in B_{>n}$ and for all $n \in \mathbb{N}$,*

$$\mathbb{E}[V(\xi, \delta) \mid x^n] > V_0[\xi(\cdot \mid x^n)] - cn. \tag{7.5.2}$$

In other words, if δ specifies that at least one observation should be taken, then the value of δ is greater than the value of choosing a decision without any observation. Furthermore, whenever δ specifies that another observation should be taken, the expected value of continuing must be smaller than the value of stopping. If the procedure is not regular, then there may be stages where the procedure specifies that sampling should be continued, though the value would be increased by stopping.

Theorem 7.5.1. *If δ is not regular, then there exists a regular δ' such that $V(\xi, \delta') \geq V(\xi, \delta)$.*

Proof. First, consider the case that $V(\xi, \delta) \leq V_0(\xi)$. This is equivalent to considering δ' to be the regular procedure which chooses $d \in D$ without any observations.

Now consider the case that $V(\xi, \delta) < V_0(\xi)$. Let δ' be the procedure which stops as soon as the observed x^n does not satisfy (7.5.2).

If, for x^n, δ stops, then both sides of (7.5.2) are equal. Consequently, δ' stops before n.

Finally, let

$$B_k(\delta) = \{x \in \mathcal{X}^* \mid n = k\} \tag{7.5.3}$$

be the set of observations such that exactly k samples are taken by rule δ and

$$B_{\leq k}(\delta) = \{x \in \mathcal{X}^* \mid n \leq k\} \tag{7.5.4}$$

be the set of observations such that at most k samples are taken by rule δ. Then

$$\begin{aligned} V(\xi, \delta) &= \sum_{k=1}^{\infty} \int_{B_k(\delta')} \{V_0[\xi(\cdot \mid x^k) - ck]\} \, \mathrm{d}\xi(x^k) \\ &\geq \sum_{k=1}^{\infty} \int_{B_k(\delta')} \mathbb{E}_\xi\{V[\xi, \delta \mid x^k]\} \, \mathrm{d}\xi(x^k) \\ &\sum_{k=1}^{\infty} \mathbb{E}_\xi\{V(\xi, \delta) \mid n(\delta') = k \, \mathbb{P}_\xi(\delta' = n)\} = V(\xi, \delta). \end{aligned}$$

□

7.6 The sequential probability ratio test

A two-point sequential decision problem.
As an illustration, consider a sequential decision problem where we must decide for one out of two possible parameters ω_1, ω_2.

- Observations $x_t \in \mathcal{X}$
- Distribution family: $\mathscr{P} = \{P_\omega \mid \omega \in \Omega\}$, probability spaces $(\mathcal{X}^*, \mathfrak{B}(\mathcal{X}^*), P_\omega)$.
- Parameter set: $\Omega = \{\omega_1, \omega_2\}$.
- Decision set: $D = \{d_1, d_2\}$.
- Prior $\xi = \mathbb{P}(\omega = \omega_1)$.
- Sampling cost $c > 0$.

$U(\omega, d)$	d_1	d_2
ω_1	0	λ_1
ω_2	λ_2	0

Table 7.1: The utility function, with $\lambda_1, \lambda_2 < 0$

The immediate value is:

$$V_0(\xi) = \max\{\lambda_1 \xi, \lambda_2(1-\xi)\}. \tag{7.6.1}$$

The worst-case immediate value, i.e. the minimum, is attained when both terms are equal. Consequently, setting $\lambda_1 \xi = \lambda_2(1-\xi)$, which gives $\xi = \lambda_2/(\lambda_1 + \lambda_2)$. Replacing in (7.6.1) gives:

$$V_0(\xi) \geq \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}.$$

Let Δ' denote the set of procedures δ which take at least one observation and define:

$$V'(\xi) = \sup_{\delta \in \Delta'} V(\xi, \delta). \tag{7.6.2}$$

Then the ξ-expected utility $V^*(\xi)$ must satisfy:

$$V^*(\xi) = \max\{V_0(\xi), V'(\xi)\}. \tag{7.6.3}$$

As we showed in Section **??**, V' is a convex function of ξ. Let

$$\xi^* \triangleq \{\xi \mid \sigma_0(\xi) \leq \sigma'(\xi)\}, \tag{7.6.4}$$

be the set of priors where it is optimal to terminate sampling.

The sequential probability ratio test

If $\xi \in (\xi_L, \xi_H)$, then it is optimal to take at least one more sample. Otherwise, it is optimal to make an immediate decision with risk $\rho_0(\xi)$.

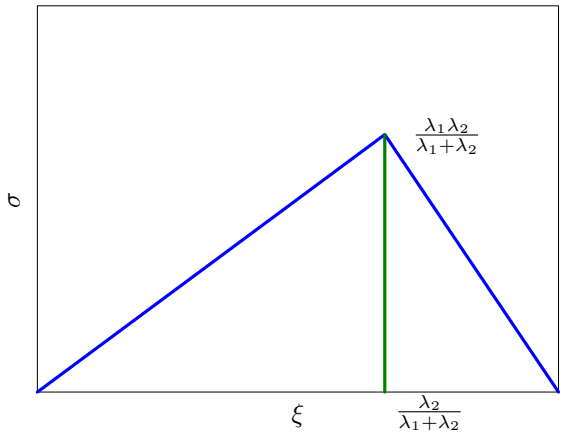


Figure 7.3: The immediate risk

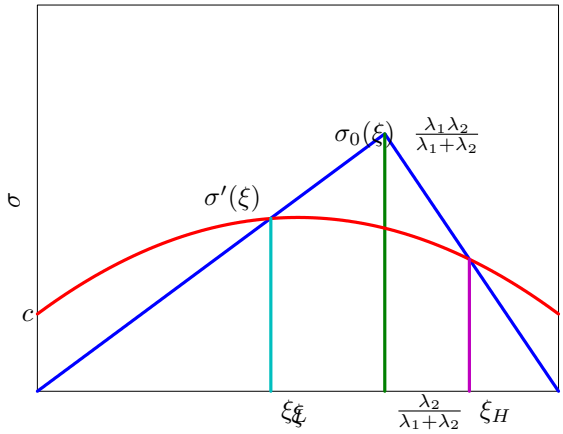


Figure 7.4: The risk of the optimal continuation σ' versus stopping σ_0.

Our posterior at time t can be written as

$$\xi_t = \frac{\xi P_{\omega_1}(x^t)}{\xi P_{\omega_1}(x^t) + (1-\xi) P_{\omega_2}(x^t)}.$$

Then, for any posterior, the optimal procedure is:

- If $\xi_L < \xi_t < \xi_H$, take one more sample.
- If $x_L \geq \xi_t$, stop and decide d_2.
- If $x_H \leq \xi_t$, stop and decide d_1.

If we let

$$A = \frac{\xi(1-\xi_H)}{(1-\xi)\xi_H}, \qquad B = \frac{\xi(1-\xi_L)}{(1-\xi)\xi_L},$$

then the optimal procedure can be restated as:

Take another observation as long as

$$A < \frac{P_{\omega_2}(x^t)}{P_{\omega_1}(x^t)} < B.$$

If the first inequality is violated, choose d_1. If the second inequality is violated, choose d_2.

Practical considerations

In general, determining ξ_L, ξ_H, or A, B, is *not trivial*. This is because we need to first calculate the risk of the optimal decision function that takes at least one more observation. Nevertheless, for a given pair A, B, the following frequentist property holds.

Frequentist property

$$\mathbb{P}(d_1 \mid \omega = \omega_2) \leq A\, \mathbb{P}(d_1 \mid \omega = \omega_1) \tag{7.6.5}$$
$$\mathbb{P}(d_2 \mid \omega = \omega_2) \geq B\, \mathbb{P}(d_2 \mid \omega = \omega_1) \tag{7.6.6}$$

Indeed, one may instead simply select A, B so as to satisfy the above property with a certain ratio.In the worst case scenario, the above are equalities. Since

$$\mathbb{P}(d_1 \mid \omega = \omega_i) + \mathbb{P}(d_2 \mid w = \omega_i) = 1, \qquad i = 1, 2,$$

we have four equations with four unknowns which we can solve to obtain

$$\mathbb{P}(d_1 \mid \omega_1) = \frac{B-1}{1-A}, \qquad \mathbb{P}(d_2 \mid \omega_2) = \frac{A(B-1)}{B-A}. \tag{7.6.7}$$

Setting $A = 1/K$, $B = K$, we get

$$\mathbb{P}(d_2 \mid \omega_1) = \mathbb{P}(d_1 \mid \omega_2) = \frac{1}{K+1}. \tag{7.6.8}$$

Consequently, we can simply choose K so as to achieve a certain error probability. The same effect can be achieved by choosing an appropriate prior ξ, so that the SPRT not only has easily-verifiable frequentist properties, but can also be derived from Bayesian principles.

7.6.1 Wald's theorem

An important tool in the analysis of SPRT as well as other procedures that stop at random times is the following theorem by Wald.

Theorem 7.6.1 (Wald's theorem)**.** *Let $z_1, z_2, \dots$ be a sequence of i.i.d. random variables with measure G, such that $\mathbb{E}\, z_i = m$ for all i. Then for any sequential procedure with $\mathbb{E}\, n < \infty$:*

$$\mathbb{E} \sum_{i+1}^{n} z_i = m\, \mathbb{E}\, n. \tag{7.6.9}$$

Proof.

$$\begin{aligned}
\mathbb{E} \sum_{i=1}^{n} z_i &= \sum_{k=1}^{\infty} \int_{B_k} \sum_{i=1}^{k} z_i \, \mathrm{d}G^k(z^k) \\
&= \sum_{k=1}^{\infty} \sum_{i=1}^{k} \int_{B_k} z_i \, \mathrm{d}G^k(z^k). \\
&= \sum_{i=1}^{\infty} \sum_{k=i}^{\infty} \int_{B_k} z_i \, \mathrm{d}G^k(z^k) \\
&= \sum_{i=1}^{\infty} \int_{B_{\geq i}} z_i \, \mathrm{d}G^i(z^i) \\
&= \sum_{i=1}^{\infty} \mathbb{E}(z_i)\, \mathbb{P}(n \geq i) = m\, \mathbb{E}\, n.
\end{aligned}$$

□

We now consider application of this theorem to the SPRT. Let $z_i = \log \frac{P_{\omega_2}(x_i)}{P_{\omega_1}(x_i)}$. Consider the equivalent formulation of the SPRT which uses

$$a < \sum_{i=1}^{n} z_i < b$$

as the test. Using Wald's theorem and the previous properties and assuming $c \approx 0$, we obtain the following approximately optimal values for a, b:

$$a \approx \log c - \log \frac{I_1 \lambda_2 (1-\xi)}{\xi} \qquad b \approx \log \frac{1}{c} - \log \frac{I_2 \lambda_1 \xi}{1-\xi}, \tag{7.6.10}$$

where $I_1 = -\mathbb{E}(z \mid \omega = \omega_1)$ and $I_2 = \mathbb{E}(z \mid \omega = \omega_2)$ is the *information*, better known as the *KL divergence*. If the cost c is very small, then the information terms vanish and we can approximate the values by $\log c$ and $\log \frac{1}{c}$.

7.7 Martingales

Martingales are a fundamentally important concept in the analysis of stochastic processes. The main idea of a martingale is that the expectation of a random variable at time $t+1$ only depends on the value of another variable at time t.

An example of a martingale sequence is when x_t is the amount of money you have at a given time, and where at each time-step t you are making a gamble such that you lose or gain 1 currency unit with equal probability. Then, at any step t, it holds that $\mathbb{E}(x_{t+1} \mid x_t) = x_t$. This concept can be generalised as follows.

1.

2. A random variable $y_n : \mathcal{S}^n \to \mathbb{R}$.

Definition 7.7.1. *Let $x^n \in \mathcal{S}^n$ be a sequence of observations $x^n \in \mathcal{S}^n$ with distribution P_n, and $y_n : \mathcal{S}^n \to \mathbb{R}$ be a random variable. Then the sequence $\{y_n\}$ is a martingale with respect to $\{x_n\}$ if for all n the expectation*

$$\mathbb{E}(y_n) = \int_{S^n} y_n(x^n) \, \mathrm{d}P_n(x^n) \tag{7.7.1}$$

exists and

$$\mathbb{E}(y_{n+1} \mid x^n) = y_n \tag{7.7.2}$$

holds with probability 1. If $\{y_n\}$ is a martingale with respect to itself, i.e. $y_i(x) = x$, then we call it simply a martingale.

Definition 7.7.2. *Similarly, sequence $\{y_n\}$ is a super-martingale if $\mathbb{E}(y_{n+1} \mid x^n) \leq y_n$ and a sub-martingale if $\mathbb{E}(y_{n+1} \mid x^n) \geq y_n$, w.p. 1.*

7.7.1 Doob martingales

At a first glance, it might appear that martingales are not very frequently encountered, apart from some niche applications. Actually, we can always construct a martingale from any sequence of random variables as follows.

Let $f : \mathcal{S}^m \to \mathbb{R}$ be some function that we are interested in, such that its expectation $\mathbb{E}(f \mid x^n)$ exists. Now, let $y_n : \mathcal{S}^n \to \mathbb{R}$ with $n \leq m$ be :

$$y_n(x^n) = \mathbb{E}[f \mid x^n].$$

Then $\mathbb{E}(y_{n+1} \mid x^n) = y_n$.

7.7.2 The Azuma-Hoeffding inequality

Definition 7.7.3. *A sequence $\{y_n\}$ is a martingale difference sequence with respect to $\{x_n\}$ if*

$$\mathbb{E}(y_{n+1} \mid x^n) = 0 \qquad \textit{with probability 1.} \tag{7.7.3}$$

Theorem 7.7.1. *Let b_k be a random variable depending on x^{k-1}. If $\{y_k\}$ is a martingale difference sequence with $y_k \in [b_k, b_k + c_k]$ w.p. 1, then setting $s_k = \sum_{i=1}^k y_i$ it holds that*

$$\mathbb{P}(s_n \geq t) \leq \exp\left(\frac{-2t^2}{\sum_{i=1}^n c_i^2}\right). \tag{7.7.4}$$

Example 7.7.1 (Estimating a mean)**.** *Let we decide to stop if*

7.8 Markov processes

Definition 7.8.1 (Markov Process)**.** *Let $(S, \mathfrak{B}(S))$ be a measurable space and define the random sequence $\{x_n\}$ such that*

$$\mathbb{P}(x_t \in A \mid x_{t-1}, \dots, x_1) = \mathbb{P}(x_t \in A \mid x_{t-1}). \tag{7.8.1}$$

x_n is called the state of the Markov process at time n.

If $\mathbb{P}(x_t \in A \mid x_{t-1} = x) = \tau(A \mid x)$ where $\tau : \mathfrak{B}(S) \times S \to [0, 1]$, *transition kernel*. In that case, $\{x_n\}$ is a stationary Markov process.

Example 7.8.1 (Markov processes)**.** *The following are examples of Markov processes. Can you think of more?*

- *The state of a Turing machine.*
- *The state in a complete game tree.*
- *Posterior parameters.*
- *The information state defined in the backwards induction tree.*

A Markov chain stopping problem

- Consider a stationary process with state space S.
- The transition kernel is a matrix $\boldsymbol{\tau}$.
- At time t, we are at state $x_t = z$ and we can either:
  - Terminate and receive reward $b(z)$.
  - Pay $c(z)$ and continue to a random state x_{t+1}.
- There maybe $S_0, S_1 \subset S$ where we may be forced to stop or continue sampling.

All the problems considered so far are stationary! This is true no matter whether the next observation is generated from the posterior state ξ_n or the true distribution P_ω.

Summary

- Sequential sampling is always better than a fixed sample size.
- Unbounded procedures are better than bounded procedures.
- Bounded procedures can be calculated using backwards induction.
- Unbounded procedures can be approximated as the limit of a sequence of bounded procedures.
- The sequential probability ratio test (SPRT) is a type of unbounded sequential decision procedure.

- The error probabilities of the SPRT $A < P_{\omega_2}(x^t)/P_{\omega_1}(x^t) < B$ are approximately $A, 1/B$. For sample cost $c \to 0$, the near-optimal values are $A = c$, $B = 1/c$.
- Martingales are a special type of sequence of random variables such that $\mathbb{E}(y_{n+1} \mid x^n) = y_n(x^n)$.
- Concentration inequalities can be derived for martingales.
- All the above problems can be modelled as Markov processes.

Chapter 8

Experiment design and Markov decision processes

8.1 Introduction

This unit describes the very general formalism of Markov decision processes (MDPs) for dealing with various problems in sequential decision making. Thus a Markov decision process can be used to model stochastic path problems, stopping problems, reinforcement learning problems, experiment design problems, and control problems.

8.1.1 Experiment design: examples

The problem of experimental design originally arose in the statistical literature when considering the problem of how to best allocate treatments with unknown efficacy to patients. The problem, originally considered by Chernoff [1959, 1966], informally can be stated as follows.

We have a number of treatments of unknown efficacy. When a new patient arrives, we must choose one of them. There are two possible, slightly different, goals: either maximise the number of cured patients, or discover the best treatment. The two different problems can be formalised as follows.

Example 8.1.1 (Adaptive treatment allocation)**.** *Consider k treatments to be administered to T volunteers. To each volunteer only a single treatment can be assigned. At the t-th trial, we perform some experiment $a_t \in \{1, \ldots, k\}$ and obtain a reward $r_t = 1$ if the result is successful and 0 otherwise. We wish to choose actions maximising $\sum_t r_t$.*

Example 8.1.2 (Adaptive hypothesis testing)**.** *We are given a hypothesis set $\Omega = \{\omega_1, \omega_2\}$, a prior ψ_0 on Ω, a decision set $\mathcal{D} = \{d_1, d_2\}$ and a utility function $\Omega : \mathcal{D} \times \Omega \to \mathbb{R}$. One hypothesis $\omega \in \Omega$ is true. We can choose from a set of k possible experiments to be performed over T trials. At the t-th trial, we choose experiment $a_t \in \{1, \ldots, k\}$ and observe outcome $x_t \in \mathcal{X}$, with $x_t \sim P_\omega$ drawn from the true hypothesis. Our posterior is*

$$\xi_t(\omega) \triangleq \xi_0(\omega \mid a_1, \ldots, a_t, x_1, \ldots, x_t).$$

The reward is $r_t = 0$ for $t < T$ and

$$r_T = \max_{d \in D} \mathbb{E}_{\xi_T}(U \mid d).$$

Again, we wish to maximise $\sum_t r_t$.

Both formalizations correspond to so-called *bandit problems* which we take a closer look at in the following section.

8.2 Bandit problems

The simplest bandit problem is the stochastic n-armed bandit. We are faced with n different one-armed bandit machines, such as those found in casinos. In this problem, at time t, you have to choose one *action* (i.e. machine) $a_t \in \mathcal{A} = \{1, \ldots, n\}$. The assumption is that each time t you play a machine, you receive a bounded reward r_t, with fixed expected value $\omega_i = \mathbb{E}(r_t \mid a_t = i)$. Unfortunately, you do not know ω_i. How do you then choose arms so as to maximise the total expected reward?

The stochastic n-armed bandit problem

- Actions $\mathcal{A} = \{1, \ldots, n\}$.
- Expected reward $\mathbb{E}(r_t \mid a_t = i) = \omega_i$.
- Select actions to maximise

$$\sum_{t=0}^{T} \gamma^t r_t,$$

with discount factor $\gamma \in [0, 1]$, horizon $T \geq 0$.

One idea is to apply the Bayesian decision-theoretic framework we have developed earlier to maximise the reward in expectation. More specifically, given the horizon $T \in (0, \infty]$ and the discount factor $\gamma \in (0, 1]$, we define our utility from time t to be:

$$U_t = \sum_{k=1}^{T-t} \gamma^k r_{t+k}. \tag{8.2.1}$$

Decision-theoretic approach

- Assume $r_t \mid a_t = a \sim P_{\omega,i}$, with $\omega \in \Omega$.
- Define prior ξ on Ω.
- Select actions to maximise $\mathbb{E}_\xi U_t = \mathbb{E}_\xi \sum_{k=1}^{T-t} \gamma^k r_{t+k}$.

8.2.1 Bernoulli bandits

As a simple illustration, consider the case when the reward for choosing one of the n actions is either 0 or 1, with some fixed, yet unknown probability depending on the chosen action. This can be modelled in the standard Bayesian framework using the Beta-Bernoulli conjugate prior. More specifically, we can formalise the problem as follows.

Consider n Bernoulli distributions with unknown parameters ω_i $(i = 1, \ldots, n)$ such that

$$r_t \mid a_t = i \sim \mathit{Bern}(\omega_i), \qquad \mathbb{E}(r_t \mid a_t = i) = \omega_i. \tag{8.2.2}$$

Each Bernoulli distribution thus corresponds to the distribution of rewards obtained from each bandit that we can play. In order to apply the statistical decision theoretic framework, we have to quantify our uncertainty about the parameters ω in terms of a probability distribution. We model our belief

for each bandit's parameter ω_i as a Beta distribution $\mathcal{Beta}(\alpha_i, \beta_i)$, with density $f(\omega \mid \alpha_i, \beta_i)$ so that

$$\xi(\omega_1, \ldots, \omega_n) = \prod_{i=1}^{n} f(\omega_i \mid \alpha_i, \beta_i).$$

Recall that the posterior of a Beta prior is also a Beta. Let

$$N_{t,i} \triangleq \sum_{k=1}^{t} \mathbb{I}\{a_k = i\}$$

be the number of times we played arm i and

$$\hat{r}_{t,i} \triangleq \frac{1}{N_{t,i}} \sum_{k=1}^{t} r_t \mathbb{I}\{a_k = i\}$$

be the *empirical reward* of arm i at time t. We can let this equal 0 when $k_{t,i} = 0$. Then, the posterior distribution for the parameter of arm i is

$$\xi_t = \mathcal{Beta}(\alpha_i + N_{t,i}\hat{r}_{t,i} \ , \ \beta_i + N_{t,i}(1 - \hat{r}_{t,i})).$$

Since $r_t \in \{0, 1\}$ the possible states of our belief given some prior are $\mathbb{N}^{2n}$.

The state of the bandit problem is the state of our belief. A sufficient statistic for our belief is the number of times we played each bandit and the total reward from each bandit. Thus, our state at time t is entirely described by our priors α, β (the initial state) and the vectors

$$N_t = (N_{t,1}, \ldots, N_{t,i}) \tag{8.2.3}$$

$$\hat{r}_t = (\hat{r}_{t,1}, \ldots, \hat{r}_{t,i}). \tag{8.2.4}$$

At any time t, we can calculate the probability of observing $r_t = 1$ or $r_t = 0$ if we pull arm i as:

$$\xi_t(r_t = 1 \mid a_t = i) = \frac{\alpha_i + N_{t,i}\hat{r}_{t,i}}{\alpha_i + \beta_i + N_{t,i}}$$

The next state is well-defined and depends only on the current state. For this reason, the decision-theoretic n-armed bandit problem can be formalised as a *Markov decision process*.

The number of states of this particular bandit problem is countable for $\{0, 1\}$ rewards, though for a finite horizon T it is of order $(2n)^T$. In the general case, the number of states is uncountable.

8.2.2 Decision-theoretic bandit process

The basic bandit process can be seen in Figure 8.1. The decision-theoretic bandit process can be formalised as follows.

Definition 8.2.1. *Let $\mathcal{A}$ be a set of actions, not necessarily finite. Let Ω be a set of possible parameter values, indexing a family of probability measures $\mathscr{P} = \{P_{\omega,a} \mid \omega \in \Omega, a \in \mathcal{A}\}$. There is some $\omega \in \Omega$ such that, whenever we take action $a_t = a$, we observe reward r_t with probability measure:*

$$P_{\omega,a}(R) \triangleq \mathbb{P}_\omega(r_t \in R \mid a_t = a). \tag{8.2.5}$$

Let ξ_1 be a prior distribution on Ω and let the posterior distributions be defined as:

$$\xi_{t+1}(B) \propto \int_B P_{\omega,a_t}(r_t \mid s_t)\, \mathrm{d}\xi_t(\omega). \tag{8.2.6}$$

The next belief is random, since it depends on the random quantity r_t. In fact, the probability of next rewards if $a_t = a$ is given by:

$$P_{\xi_t,a}(R) \triangleq \int_\Omega P_{\omega,a}(R)\, \mathrm{d}\xi_t(\omega) \tag{8.2.7}$$

Finally, as ξ_{t+1} deterministically depends on ξ_t, a_t, r_t, the probability of obtaining a particular next belief is the same as the probability of obtaining the corresponding rewards leading to the next belief. In more detail, we can write:

$$\mathbb{P}(\xi_{t+1} = \xi \mid \xi_t, a_t) = \int_{\mathcal{R}} \mathbb{I}\left\{\xi_{t+1}(\cdot \mid a_t, r_t = r) = \xi\right\} \mathrm{d}P_{\xi_t,a}(r). \tag{8.2.8}$$

In practice, although multiple reward sequences may lead to the same beliefs, we frequently ignore that possibility for simplicity. Then the process becomes a tree.

Since the next belief only depends on the current belief, action and reward, it satisfies the Markov property. Consequently, a decision-theoretic bandit process can be modelled as a Markov decision process.

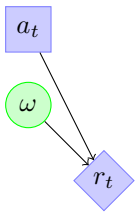


Figure 8.1: The basic bandit process. The decision maker selects a_t, while the parameter ω of the process is hidden. It then obtains reward r_t. The process repeats for $t = 1, \ldots, T$.

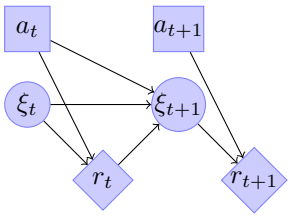


Figure 8.2: The decision-theoretic bandit process. While ω is not known, at each time step t we maintain a belief ξ_t on Ω. The reward distribution is then defined through our belief.

8.3 Markov decision processes and reinforcement learning

Reinforcement learning

Bandit problems are one of the simplest instances of reinforcement learning problems. Informally, speaking, these are problems of learning how to act in an unknown environment, only through interaction with the environment and limited reinforcement signals.

The reinforcement learning problem.
Learning to act in an *unknown* environment, by interaction and reinforcement.

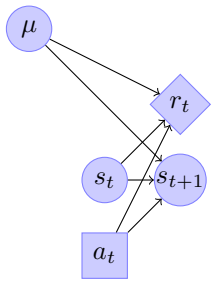
Generally, we assume that the environment that we are acting in has an underlying state s_t, which changes with time t. At the same time, the agent chooses actions a_t. We usually assume that the environment is such that its next state s_{t+1} only depends on its current state s_t and the last action taken by the agent, a_t. In addition, the agent observes a reward signal r_t, and its goal is to maximise the total reward during its lifetime.

This problem is hard even in seemingly simple problems, like n-armed bandits, where the underlying state never changes. However, in many real-world applications, the state is not directly observed. Instead, we may simply have some measurements x_t, which give only partial information about the true underlying state.

Reinforcement learning problems typically fall into one of the following three groups: (1) Markov decision processes (MDPs), where the state is known; (2) Partially observable MDPs (POMDPs), where the state is hidden; and (3) potentially partially observable Markov games, where the next state also depends on the move of other agents. While all of these problem *descriptions* are different, in the Bayesian setting, they all can be reformulated as MDPs, albeit with a very large state space.

Markov decision processes (MDP).
At each time step t:

- We observe *state* $s_t \in \mathcal{S}$.
- We take *action* $a_t \in \mathcal{A}$.
- We receive a *reward* $r_t \in \mathbb{R}$.



Markov property of the reward and state distribution

$$\mathbb{P}_\mu(s_{t+1} \mid s_1, a_1, \ldots, s_t, a_t) = \mathbb{P}_\mu(s_{t+1} \in S \mid s_t, a_t) \tag{8.3.1}$$
$$\mathbb{P}_\mu(r_t \mid s_1, a_1, \ldots, s_t, a_t) = \mathbb{P}_\mu(r_t \in R \mid s_t, a_t) \tag{8.3.2}$$

where $S \subset \mathcal{S}$ and $R \subset \mathcal{R}$.

More formally, we have the following definition.

Definition 8.3.1. *A Markov decision process μ is a tuple $\mu = \langle \mathcal{S}, \mathcal{A}, \mathscr{P}, \mathscr{R} \rangle$, such that $\mathscr{P} = \{P(\cdot \mid s, a) \mid s \in \mathcal{S}, a \in \mathcal{A}\}$ is a collection of probability measures on $\mathcal{S}$, indexed in $\mathcal{S} \times \mathcal{A}$ and $\mathscr{R} = \{\rho(\cdot \mid s, a) \mid s \in \mathcal{S}, a \in \mathcal{A}\}$ is a collection of probability measures on $\mathcal{R}$, such that:*

$$P(S \mid s, a) = \mathbb{P}_\mu(s_{t+1} \in S \mid s_t = s, a_t = a) \tag{8.3.3}$$
$$\rho(R \mid s, a) = \mathbb{P}_\mu(r_t \in R \mid s_t = s, a_t = a). \tag{8.3.4}$$

For simplicity, we shall also use

$$r_\mu(s, a) = \mathbb{E}_\mu(r_{t+1} \mid s_t = s, a_t = a), \tag{8.3.5}$$

for the expected reward.

Of course, the transition and reward distributions are different for different environments μ. For that reason, we shall usually subscript the relevant probabilities and expectations with μ.

Dependencies of rewards

Sometimes it is more convenient to have rewards that depend on the next state as well, i.e.

$$r_\mu(s, a, s') = \mathbb{E}_\mu(r_{t+1} \mid s_t = s, a_t = a, s_{t+1} = s'), \tag{8.3.6}$$

though this is complicats notation considerably since now the reward is obtained on the next time step. However, we can always replace this with the expected reward for a given state-action pair:

$$r_\mu(s, a) = \mathbb{E}_\mu(r_{t+1} \mid s_t = s, a_t = s) \tag{8.3.7}$$
$$= \sum_{s' \in \mathcal{S}} P_\mu(s' \mid s, a) r_\mu(s, a, s') \tag{8.3.8}$$

In fact, it is notationally more convenient to have rewards that only depend on the current state:

$$r_\mu(s) = \mathbb{E}_\mu(r_t \mid s_t = s). \tag{8.3.9}$$

Many times, for simplicity, we shall only consider the latter case.

The agent

The environment does not exist in isolation. The actions are taken by an agent, who is interested in obtaining high rewards. Instead of defining an algorithm for choosing actions directly, we define an algorithm for computing policies, which define distributions on actions.

The agent's policy π

$$\mathbb{P}^\pi(a_t \mid s_t, \ldots, s_1, a_{t-1}, \ldots, a_1) \quad \text{(history-dependent policy)}$$
$$\mathbb{P}^\pi(a_t \mid s_t) \quad \text{(Markov policy)}$$

In some sense, the agent is defined by its *policy* π, which is a conditional distribution on actions given the history. The policy π is otherwise known as a *decision function* or a *strategy*. In general, the policy can be history-dependent. In certain cases, however, there are optimal policies that are Markov. This is for example the case with additive utility functions.

Definition 8.3.2 (Utility)**.** *Given a horizon T and a discount factor $\gamma \in (0, 1]$, the utility can be defined as*

$$U_t \triangleq \sum_{k=0}^{T-t} \gamma^k r_{t+k} \tag{8.3.10}$$

The agent wants to to find π *maximising* the *expected total future reward*

$$\mathbb{E}_\mu^\pi U_t = \mathbb{E}_\mu^\pi \sum_{k=0}^{T-t} \gamma^k r_{t+k}. \quad \text{(expected utility)}$$

For simplicity, we shall also use $r_\mu^\pi(s) = \mathbb{E}_\mu^\pi(r_{t+1} \mid s_t = s)$ for the expected reward at a given state, or simply $r(s)$ when clear from context.

8.3.1 Value functions

It is frequently useful to employ the following abbreviations. These will be employed in the development of both theory and algorithms and are simply the expected utility for a given policy and MDP conditioned on different states and/or actions.

State value function

$$V_{\mu,t}^\pi(s) \triangleq \mathbb{E}_\mu^\pi(U_t \mid s_t = s) \tag{8.3.11}$$

State-action value function

$$Q_{\mu,t}^\pi(s, a) \triangleq \mathbb{E}_\mu^\pi(U_t \mid s_t = s, a_t = a) \tag{8.3.12}$$

It is also useful to define the optimal policy and optimal value functions for a given MDP. In the following, a star indicated optimal quantities. The *optimal policy* π^*

$$\pi^*(\mu) : V_{t,\mu}^{\pi^*(\mu)}(s) \geq V_{t,\mu}^{\pi}(s) \quad \forall \pi, t, s \tag{8.3.13}$$

dominates all other policies π everywhere in $\mathcal{S}$.

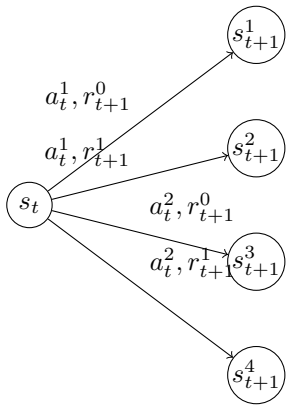
The *optimal value function* V^*

$$V_{t,\mu}^*(s) \triangleq V_{t,\mu}^{\pi^*(\mu)}(s), \quad Q_{t,\mu}^*(s) \triangleq Q_{t,\mu}^{\pi^*(\mu)}(s,a). \tag{8.3.14}$$

is the value function of the optimal policy π^*.

Finding the optimal policy when μ is known

When the MDP μ is known, the expected utility of any policy can be calculated. Therefore, one could find the optimal policy by brute force. However, there are faster methods which can be employed. First, there are iterative/offline methods where an optimal policy is found for all states of the MDP. These either try to estimate the optimal value function directly, or try to iteratively improve a policy until it is optimal. The second type of methods tries to find an optimal policy online. That is, the optimal actions are estimated only for states which can be visited in the future starting from the current state. However, the same main ideas are used in all of these algorithms.



8.4 Finite horizon, undiscounted problems

The conceptually simplest type of problems are finite horizon problems where $T < \infty$ and $\gamma = 1$. The first thing we shall try to do is to evaluate a given policy for a given MDP.

8.4.1 Policy evaluation

Policy evaluation

An optimal policy

> An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision. – Bellman.

The value function of a policy π (for $\gamma = 1, T < \infty$) can be determined by the following recursion, noting that $U_{t+1} = \sum_{k=1}^{T-t} r_{t+k}$:

$$\begin{aligned} V_{\mu,t}^{\pi}(s) &\triangleq \mathbb{E}_{\mu}^{\pi}(U_t \mid s_t = s) && (8.4.1)\\ &= \sum_{k=0}^{T-t} \mathbb{E}_{\mu}^{\pi}(r_{t+k} \mid s_t = s) && (8.4.2)\\ &= \mathbb{E}_{\mu}^{\pi}(r_t \mid s_t = s) + \mathbb{E}_{\mu}^{\pi}(U_{t+1} \mid s_t = s) && (8.4.3)\\ &= \mathbb{E}_{\mu}^{\pi}(r_t \mid s_t = s) + \sum_{i \in \mathcal{S}} V_{\mu,t+1}^{\pi}(i)\, \mathbb{P}_{\mu}^{\pi}(s_{t+1} = i | s_t = s). && (8.4.4) \end{aligned}$$

Note that

$$\mathbb{P}_{\mu}^{\pi}(s_{t+1} = i | s_t = s) = \sum_{a \in \mathcal{A}} \mathbb{P}_{\mu}(s_{t+1} = i | s_t = s, a_t = a)\, \mathbb{P}^{\pi}(a_t = a | s_t = s). \tag{8.4.5}$$

This derivation directly gives a number of *policy evaluation algorithms*.

Direct policy evaluation as given by Algorithm 2 is very simple. We calculate the probability of reaching any state from any other state at different times, and then add up the expected reward we would get in that state under our policy. It is easy to see that $\hat{V}_t(s) = V_{\mu,t}^{\pi}(s)$.

Algorithm 2 Direct policy evaluation

1: **for** $s \in \mathcal{S}$ **do**
2: **for** $t = 0, \ldots, T$ **do**
3:
$$\hat{V}_t(s) = \sum_{k=t}^{T} \sum_{j \in \mathcal{S}} \mathbb{P}_{\mu}^{\pi}(s_k = j \mid s_t = s)\, \mathbb{E}_{\mu}^{\pi}(r_k \mid s_k = j).$$
4: **end for**
5: **end for**

8.4.2 Monte-Carlo policy evaluation

Another conceptually simple algorithm is Monte-Carlo policy evaluation shown as Algorithm 3. The idea is that instead of summing over all possible states to be visited, we just draw states from the Markov chain defined jointly by the policy and the Markov decision process. Unlike direct policy evaluation the algorithm needs a parameter K, the number of trajectories to generate. Nevertheless, this is a very useful method, employed within a number of more complex algorithms.

Algorithm 3 Monte-Carlo policy evaluation

for $s \in \mathcal{S}$ **do**
  for $k = 0, \dots, K$ **do**
    Choose initial state s_1.
    for $t = 1, \dots, T$ **do**
      $a_t \sim \pi(a_t \mid s_t)$ // Take action
      Observe reward r_t and next state s_{t+1}.
      Set $r_{t,k} = r_t$.
    end for
    Save total reward:

$$\hat{V}_k(s) = \sum_{t=1}^{T} r_{t,k}.$$

  end for
  Calculate estimate:

$$\hat{V}(s) = \frac{1}{K} \sum_{k=1}^{K} \hat{V}_k(s).$$

end for

Remark 8.4.1. *The estimate $\hat{V}$ of the Monte Carlo evaluation algorithm satisfies*

$$\|V - \hat{V}\|_\infty \leq \sqrt{\frac{\ln(2|\mathcal{S}|/\delta)}{2K}} \qquad \textit{with probability } 1 - \delta$$

Proof. From Hoeffding's inequality we have for any state s that

$$\mathbb{P}\left(|\hat{V}(s) - V(s)| \geq \sqrt{\frac{\ln(2|\mathcal{S}|/\delta)}{2K}} \right) \leq \delta/|\mathcal{S}|.$$

Consequently, using a union bound of the form $P(A_1 \cup A_2 \cup \ldots \cup A_n) \leq \sum_i P(A_i)$ gives the required result. □

Finally, the backwards induction algorithm shown as Algorithm 4 is similar to the backwards induction algorithm we saw for sequential sampling problems. However, here we are only evaluating a policy rather than finding the optimal policy. This algorithm is slightly less generally applicable than the Monte-Carlo method because it implicitly assumes that the policy we evaluate only takes actions depending on the current state.

Algorithm 4 Backwards induction policy evaluation

For each state $s \in S$, for $t = 1, \dots, T-1$:

$$\hat{V}_t(s) = r_\mu^\pi(s) + \sum_{j \in S} \mathbb{P}_\mu^\pi(s_{t+1} = j \mid s_t = s) \hat{V}_{t+1}(j), \tag{8.4.6}$$

with $\hat{V}_T(s) = r_\mu^\pi(s)$.

Theorem 8.4.1. *The backwards induction algorithm gives estimates $\hat{V}_t(s)$ satisfying*

$$\hat{V}_t(s) = V_{\mu,t}^\pi(s) \tag{8.4.7}$$

Proof. For $t = T - 1$, the result is obvious. We can prove the remainder by induction. Let (8.4.7) hold for all $t \geq n + 1$. Now we prove that it holds for n. Note that from the recursion (8.4.6) we have:

$$\begin{aligned}\hat{V}_t(s) &= r_\mu(s) + \sum_{j \in S} \mathbb{P}_{\mu,\pi}(s_{t+1} = j \mid s_t = s)\hat{V}_{t+1}(j) \\ &= r(s) + \sum_{j \in S} \mathbb{P}_{\mu,\pi}(s_{t+1} = j \mid s_t = s)V^\pi_{\mu,t+1}(j) \\ &= r(s) + \mathbb{E}_{\mu,\pi}(U_{t+1} \mid s_t = s) \\ &= \mathbb{E}_{\mu,\pi}(U_t \mid s_t = s) = V^\pi_{\mu,t}(s),\end{aligned}$$

where the second equality is by the induction hypothesis, the third and fourth equalities are by the definition of the utility, and the last by definition of $V^\pi_{\mu,t}$. □

As you can see, the proof follows the same outline as the expansion of the policy value.The same theorem can be proven for history-dependent policies.

8.4.3 Finite horizon backwards induction

Backwards induction as given in algorithm 5 is the first non-naive algorithm for finding an optimal policy for the case where there are T stages. It is basically identical to the backwards induction algorithm we saw in Chapter 7.

Algorithm 5 Finite-horizon backwards induction

Input μ, set $\mathcal{S}_T$ of states reachable within T steps.
Initialise $V_T(s) := \max_a r(s, a)$, for all $s \in \mathcal{S}_T$.
for $n = T - 1, T - 2, \ldots, 1$ **do**
  for $s \in \mathcal{S}_n$ **do**
    $\pi_n(s) = \arg\max_a r(s, a) + \sum_{s' \in \mathcal{S}_{n+1}} P_\mu(s' \mid s, a)V_{n+1}(s')$
    $V_n(s) = r(s, a) + \sum_{s' \in \mathcal{S}_{n+1}} P_\mu(s' \mid s, \pi_n(s))V_{n+1}(s')$
  end for
end for
Return $\pi = (\pi_n)_{n=1}^T$.

Theorem 8.4.2. *For T-horizon problems, backwards induction is optimal, i.e.*

$$V_n(s) = V^*_{\mu,n}(s) \tag{8.4.8}$$

Proof. Note that the proof below also holds for $r(s, a) = r(s)$. First we show that $V_t \geq V^*_t$. For $n = T$ we evidently have $V_T(s) = \max_a r(s, a) = V^*_{\mu,T}(s)$. Now assume that for $n \geq t + 1$, (8.4.8) holds. Then it also holds for $n = t$, since

for any policy π'

$$\begin{aligned} V_t(s) &= \max_a \left\{ r(s,a) + \sum_{j \in \mathcal{S}} P_\mu(j \mid s,a) V_{t+1}(j) \right\} \\ &\geq \max_a \left\{ r(s,a) + \sum_{j \in \mathcal{S}} P_\mu(j \mid s,a) V^*_{\mu,t+1}(j) \right\} \quad \text{(by induction assumption)} \\ &\geq \max_a \left\{ r(s,a) + \sum_{j \in \mathcal{S}} P_\mu(j \mid s,a) V^{\pi'}_{\mu,t+1}(j) \right\} \\ &\geq V_t^{\pi'}(s). \end{aligned}$$

This holds for any policy π', including $\pi' = \pi$, the policy returned by backwards induction. Then:

$$V^*_{\mu,t}(s) \geq V^{\pi}_{\mu,t}(s) = V_t(s) \geq V^*_{\mu,t}(s).$$

□

Remark 8.4.2. *A similar theorem can be proven for arbitrary $\mathcal{S}$. This requires using* sup *instead of* max *and proving the existence of a π' that is arbitrary-close in value to $\boldsymbol{v}$.*

8.5 Infinite-horizon

When problems have no fixed horizon, they usually can be modelled as infinite horizon problems, sometimes with help of a *terminating state*, whose visit terminates the problem, or discounted rewards, which indicate that we care less about rewards further in the future. When reward discounting is exponential, these problems can be seen as undiscounted problems with random and geometrically distributed horizon. For problems with no discounting and no termination states there are some complications in the definition of optimal policy. However, we defer discussion of such problems to Chapter **??**.

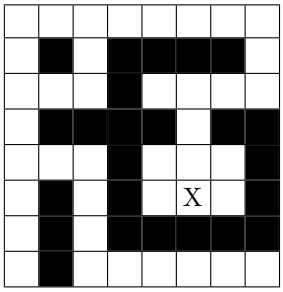
8.5.1 Examples

We begin with some examples.

Shortest-path problems

Deterministic shortest-path problems

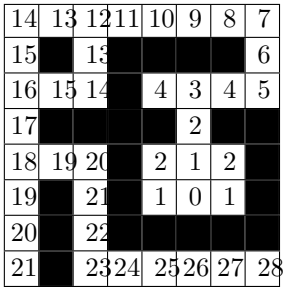
Consider an agent moving in a maze, aiming to get to some terminating goal state X. That is, when reaching this state, the agent cannot move anymore, and receives a reward of 0. In general, the agent can move deterministically in the four cardinal directions, and receives a negative reward at each time step. Consequently, the optimal policy is to move to X as quickly as possible.



Properties

- $\gamma = 1$, $T \to \infty$.
- $r_t = -1$ unless $s_t = X$, in which case $r_t = 0$.
- $\mathbb{P}_\mu(s_{t+1} = X | s_t = X) = 1$.
- $\mathcal{A} = \{\text{North}, \text{South}, \text{East}, \text{West}\}$
- Transitions are deterministic and walls block.

Solving the shortest path problem can be done simply by looking at the distance of any point to X. Then the reward obtained by the optimal policy starting from any point, is simply the negative distance. The optimal policy simply moves to the state with the smallest distance to X.

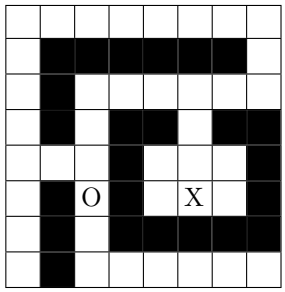


Properties

- $\gamma = 1$, $T \to \infty$.
- $r_t = -1$ unless $s_t = X$, in which case $r_t = 0$.
- The length of the shortest path from s equals the negative value of the optimal policy.
- Also called *cost-to-go*.

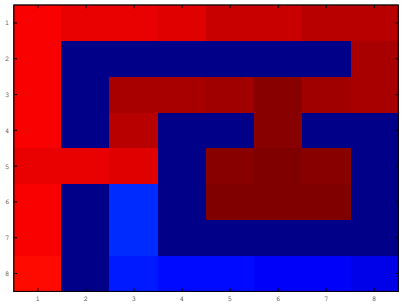
Stochastic shortest path problem with a pit

Now assume the shortest path problem with stochastic dynamics. That is, at each time-step there is a small probability ω that move to a random direction. In addition, there is a pit O, that is a terminating state with a reward of -100.

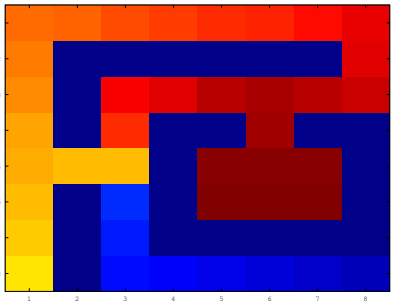


Properties

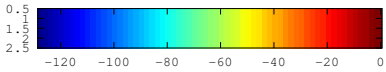
- $\gamma = 1$, $T \to \infty$.
- $r_t = -1$, but $r_t = 0$ at X and -100 at O and episode ends.
- $\mathbb{P}_\mu(s_{t+1} = X | s_t = X) = 1$.
- $\mathcal{A} = \{\text{North}, \text{South}, \text{East}, \text{West}\}$
- Moves to a random direction with probability ω. Walls block.



(a) $\omega = 0.1$



(b) $\omega = 0.5$



(c) value

Figure 8.3: Pit maze solutions for two values of ω.

Randomness changes the solution significantly in this environment. When ω is relatively small, it is worthwhile (in expectation) for the agent to pass past the pit, even though there is a risk of falling in and getting a reward of -100. In the example given, even starting from the third row, the agent prefers taking the short-cut. For high enough ω, the optimal policy avoids approaching the pit. Still, the agent prefers jumping in the pit, than being trapped at the bottom of the maze forever.

Continuing problems

Finally, many problems have no natural terminating state, but are continuing *ad infinitum*. Frequently, we model those problems using a utility that discounts future rewards exponentially. As an example, consider the following inventory management problem. There are K storage locations, and each location i can store n_i items. At each time-step there is a probability ϕ_i that a client tries to buy an item from location i, where $\sum_i \phi_i \leq 1$. If there is an item available,

when this happens, you gain reward 1. There are two types of actions, one for ordering a certain number u units of stock, paying $c(u)$. Further one may move u units of stock from one location i to another location j, paying $\psi_{ij}(u)$.

An easy special case is when $K = 1$, and we assume that deliveries happen once every m timesteps, and each time-step a client arrives with probability ϕ. Then the state set $\mathcal{S} = \{0, 1, \ldots, n\}$ corresponds to the number of items we have, the action set $\mathcal{A} = \{0, 1, \ldots, n\}$ to the number of items we may order. The transition probabilities are given by $P(s'|s, a) = \binom{m}{d}\phi^d(1-\phi)^{m-d}$, where $d = s + a - s'$, for $s + a \leq n$.

8.5.2 Markov chain theory for discounted problems

We first consider MDPs with discounted rewards, and only turn later to algorithms and bounds for the undiscounted case.

Discounted total reward.
Our utility in this case is:

$$U_t = \lim_{T\to\infty} \sum_{k=t}^{T} \gamma^k r_k, \qquad \gamma \in (0, 1)$$

For simplicity, in the following we assume that rewards only depend on the current state instead of both state and action. It can easily be verified that results still hold in the latter case. We use the following notation:

- $\boldsymbol{v}^\pi = (\mathbb{E}^\pi(U_t \mid s_t = s))_{s\in\mathcal{S}}$ is a vector in $\mathbb{R}^{|\mathcal{S}|}$ representing the value of policy π.
- $\boldsymbol{P}_{\mu,\pi}$ is a transition matrix in $\mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$ for policy π.
- Sometimes we will use $p(j|s, a)$ as a shorthand for $\mathbb{P}_\mu(s_{t+1} = j \mid s_t = s, a_t = a)$.
- $\boldsymbol{r}$ is a reward vector in $\mathbb{R}^{|\mathcal{S}|}$.
- The space of value functions $\mathcal{V}$ is a Banach space (i.e., a complete, normed vector space) equipped with the norm

$$\|\boldsymbol{v}\| = \sup\{|\boldsymbol{v}(s)| \mid s \in \mathcal{S}\}$$

For infinite-horizon discounted MDPs, stationary policies are sufficient. This can be proven by induction, using arguments similar to other proofs given here. For a detailed set of proofs, see Puterman [1994].

Definition 8.5.1. *A policy π is stationary if $\pi(a_t \mid s_t) = \pi(a_n \mid s_n)$ for all n, t.*

We now present a set of important results that link Markov decision processes to linear algebra.

Remark 8.5.1. *We can use the Markov chain kernel $\boldsymbol{P}$ to write the expected reward vector as*

$$\boldsymbol{v}^\pi = \sum_{t=0}^{\infty} \gamma^t \boldsymbol{P}_{\mu,\pi}^t \boldsymbol{r} \tag{8.5.1}$$

Proof.

$$\begin{aligned} V^\pi(s) &= \mathbb{E}\left(\sum_{t=0}^{\infty} \gamma^t r_t \;\middle|\; s_0 = s \right) \\ &= \sum_{t=0}^{\infty} \gamma^t \, \mathbb{E}(r_t | s_0 = s) \\ &= \sum_{t=0}^{\infty} \gamma^t \sum_{i \in \mathcal{S}} \mathbb{P}(s_t = i \mid s_0 = s) \, \mathbb{E}(r_t \mid s_t = i). \end{aligned}$$

Since for any distribution vector $\boldsymbol{p}$ over $\mathcal{S}$, we have $\mathbb{E}_{\boldsymbol{p}} r_t = \boldsymbol{p}^\top \boldsymbol{r}$, the result follows. □

Exercise 5. *Show that the expected discounted total reward of any given policy is equal to the expected undiscounted total reward with a finite, but random horizon T. In particular, let T be distributed according to a geometric distribution on $\{1, 2, \ldots\}$ with parameter $1 - \gamma$. Then show that:*

$$\mathbb{E} \lim_{T \to \infty} \sum_{k=0}^{T} \gamma^k r_k = \mathbb{E}\left(\sum_{k=0}^{T} r_k \;\middle|\; T \sim \mathit{Geom}(1-\gamma) \right).$$

The value of a particular policy can be expressed as a linear equation. This is an important result, as it has led to a number of successful algorithms that employ linear theory.

Theorem 8.5.1. *For any stationary policy π, $\boldsymbol{v}^\pi$ is the unique solution of*

$$\boldsymbol{v} = \boldsymbol{r} + \gamma \boldsymbol{P}_{\mu,\pi} \boldsymbol{v}. \tag{8.5.2}$$

In addition, the solution is:

$$\boldsymbol{v}^\pi = (\boldsymbol{I} - \gamma \boldsymbol{P}_{\mu,\pi})^{-1} \boldsymbol{r}. \tag{8.5.3}$$

To prove this we will need the following important theorem.

Theorem 8.5.2. *For any bounded linear transformation $\boldsymbol{A} : S \to S$ on a normed linear space S (i.e., there is $c < \infty$ s.t. $\|\boldsymbol{A}x\| := \sup_i \sum_j a_{i,j} \leq c\|x\|$ for all $x \in S$ with spectral radius $\sigma(\boldsymbol{A}) \lim_{n\to\infty} := \|\boldsymbol{A}^n\|^{1/n} < 1$), $\boldsymbol{A}^{-1}$ exists and is given by*

$$\boldsymbol{A}^{-1} = \lim_{T \to \infty} \sum_{n=0}^{T} (\boldsymbol{I} - \boldsymbol{A})^n. \tag{8.5.4}$$

Proof of Theorem 8.5.1. First note that by manipulating the infinite sum in Remark 8.5.1, one obtains $\boldsymbol{r} = (\boldsymbol{I} - \gamma \boldsymbol{P}_{\mu,\pi})\boldsymbol{v}^\pi$. Since $\|\gamma \boldsymbol{P}_{\mu,\pi}\| < 1 \cdot \|\boldsymbol{P}_{\mu_\pi}\| = 1$, the inverse

$$(\boldsymbol{I} - \gamma \boldsymbol{P}_{\mu,\pi})^{-1} = \lim_{n\to\infty} \sum_{t=0}^{n} (\gamma \boldsymbol{P}_{\mu,\pi})^t$$

exists by Theorem 8.5.2. It follows that

$$\boldsymbol{v} = (\boldsymbol{I} - \gamma \boldsymbol{P}_{\mu,\pi})^{-1}\boldsymbol{r} = \sum_{t=0}^{\infty} \gamma^t \boldsymbol{P}_{\mu,\pi}^t \boldsymbol{r} = \boldsymbol{v}^\pi,$$

where the last step is by Remark 8.5.1 again. □

8.5.3 Optimality equations

Let us now look at the backwards induction algorithms in terms of operators. We introduce the operator of a policy, which is the one-step backwards induction operation for a fixed policy, and the Bellman operator, which is the equivalent operator for the optimal policy. If a value function is optimal, then it satisfies the Bellman optimality equation.

Definition 8.5.2 (Policy and Bellman operator)**.** *The linear operator of a policy π is:*

$$\mathscr{L}_\pi \boldsymbol{v} \triangleq \boldsymbol{r} + \gamma \boldsymbol{P}_\pi \boldsymbol{v} \tag{8.5.5}$$

The (non-linear) Bellman operator in the space of value functions $\mathcal{V}$ is defined as:

$$\mathscr{L} \boldsymbol{v} \triangleq \sup_\pi \{\boldsymbol{r} + \gamma \boldsymbol{P}_\pi \boldsymbol{v}\}, \qquad \boldsymbol{v} \in \mathcal{V} \tag{8.5.6}$$

We now show that the Bellman operator satisfies the following monotonicity properties with respect to an arbitrary value vector $\boldsymbol{v}$.

Theorem 8.5.3. *Let $\boldsymbol{v}^* := \sup_\pi \boldsymbol{v}^\pi$. Then for any bounded $\boldsymbol{r}$, it holds that for $\boldsymbol{v} \in \mathcal{V}$:*

(1) If $\boldsymbol{v} \geq \mathscr{L}\boldsymbol{v}$, then $\boldsymbol{v} \geq \boldsymbol{v}^*$.

(2) If $\boldsymbol{v} \leq \mathscr{L}\boldsymbol{v}$, then $\boldsymbol{v} \leq \boldsymbol{v}^*$.

(3) If $\boldsymbol{v} = \mathscr{L}\boldsymbol{v}$, then $\boldsymbol{v}$ is unique and $\boldsymbol{v} = \sup_\pi \boldsymbol{v}^\pi$. Therefore, $\boldsymbol{v} = \mathscr{L}\boldsymbol{v}$ is called the Bellman optimality equation.

Proof. We first prove (1). A simple proof by induction over n shows that for any π

$$\boldsymbol{v} \geq \boldsymbol{r} + \gamma \boldsymbol{P}_\pi \boldsymbol{v} \geq \sum_{k=0}^{n-1} \gamma^k \boldsymbol{P}_\pi^k \boldsymbol{r} + \gamma^n \boldsymbol{P}_\pi^n \boldsymbol{v}.$$

Since $\boldsymbol{v}^\pi = \sum_{t=0}^{\infty} \gamma^t \boldsymbol{P}_\pi^t \boldsymbol{r}$ it follows that

$$\boldsymbol{v} - \boldsymbol{v}^\pi \geq \gamma^n \boldsymbol{P}_\pi^n \boldsymbol{v} - \sum_{k=n}^{\infty} \gamma^k \boldsymbol{P}_\pi^k \boldsymbol{r}.$$

The first-term on the right-hand side can be bounded by arbitrary $\epsilon/2$ for large enough n. Also note that

$$\sum_{k=n}^{\infty} \gamma^k \boldsymbol{P}_\pi^k \boldsymbol{r} \geq -\frac{\gamma^n \boldsymbol{e}}{1-\gamma},$$

with $\boldsymbol{e}$ being a unit vector, so this can be bounded by $\epsilon/2$ as well. So for any $\pi, \epsilon > 0$:

$$\boldsymbol{v} \geq \boldsymbol{v}^\pi - \epsilon,$$

so

$$\boldsymbol{v} \geq \sup_\pi \boldsymbol{v}^\pi.$$

An equivalent argument shows that

$$\boldsymbol{v} \leq \boldsymbol{v}^\pi + \epsilon,$$

proving (2). Putting together (1) and (2) gives (3). □

We eventually want show that repeated application of the Bellman operator converges to the optimal value. As a preparation, we need the following theorem.

Theorem 8.5.4 (Banach Fixed-Point theorem)**.** *Suppose $\mathcal{S}$ is a Banach space (i.e. a complete normed linear space) and $T : \mathcal{S} \to \mathcal{S}$ is a contraction mapping (i.e. $\exists \gamma \in [0, 1)$ s.t. $\|Tu - Tv\| \leq \gamma \|u - v\|$ for all $u, v \in \mathcal{S}$). Then*

- *there is a unique $u^* \in U$ s.t. $Tu^* = u^*$, and*
- *for any $u^0 \in \mathcal{S}$ the sequence $\{u^n\}$:*

$$u^{n+1} = Tu^n = T^{n+1}u^0$$

converges to u^*.

Proof. For any $m \geq 1$

$$\begin{aligned}\|u^{n+m} - u^n\| &\leq \sum_{k=0}^{m-1} \|u^{n+k+1} - u^{n+k}\| = \sum_{k=0}^{m-1} \|T^{n+k}u^1 - T^{n+k}u^0\| \\ &\leq \sum_{k=0}^{m-1} \gamma^{n+k} \|u^1 - u^0\| = \frac{\gamma^n(1-\gamma^m)}{1-\gamma} \|u^1 - u^0\|.\end{aligned}$$

□

Theorem 8.5.5. *For $\gamma \in [0, 1)$ the Bellman operator $\mathscr{L}$ is a contraction mapping in $\mathcal{V}$.*

Proof. Let $\boldsymbol{v}, \boldsymbol{v}' \in \mathcal{V}$. Consider $s \in \mathcal{S}$ such that $\mathscr{L}\boldsymbol{v}(s) \geq \mathscr{L}\boldsymbol{v}'(s)$, and let

$$a_s^* \in \arg\max_{a \in \mathcal{A}} \left\{ r(s) + \sum_{j \in \mathcal{S}} \gamma p_\mu(j \mid s, a) \boldsymbol{v}(j) \right\}.$$

Then

$$\begin{aligned}0 \leq \mathscr{L}\boldsymbol{v}(s) - \mathscr{L}\boldsymbol{v}'(s) &\leq \sum_{j\in S} \gamma p(j \mid s, a_s^*)\boldsymbol{v}(j) - \sum_{j\in S} \gamma p(j \mid s, a_s^*)\boldsymbol{v}'(j) \\ &= \gamma \sum_{j\in S} p(j \mid s, a_s^*)[\boldsymbol{v}(j) - \boldsymbol{v}'(j)] \\ &\leq \gamma \sum_{j\in S} p(j \mid s, a_s^*)\|\boldsymbol{v} - \boldsymbol{v}'\| = \gamma\|\boldsymbol{v} - \boldsymbol{v}'\|.\end{aligned}$$

Repeating the argument for s such that $\mathscr{L}\boldsymbol{v}(s) \leq \mathscr{L}\boldsymbol{v}'(s)$, we obtain

$$|\mathscr{L}\boldsymbol{r}(s) - \mathscr{L}\boldsymbol{r}'(s)| \leq \gamma\|\boldsymbol{r} - \boldsymbol{r}'\|.$$

Taking the supremum, we the required result follows. □

Theorem 8.5.6. *For discrete $\mathcal{S}$, bounded $\boldsymbol{r}$, and $\gamma \in [0, 1)$*

(i) there is a unique $\boldsymbol{v}^* \in \mathcal{V}$ such that $\mathscr{L}\boldsymbol{v}^* = \boldsymbol{v}^*$ and such that $\boldsymbol{v}^* = V_\mu^*$,

(ii) for any stationary policy π, there is a unique $\boldsymbol{v} \in \mathcal{V}$ such that $\mathscr{L}_\pi \boldsymbol{v} = \boldsymbol{v}$ and $\boldsymbol{v} = V_\mu^\pi$.

Proof. As the Bellman operator $\mathscr{L}$ is a contraction by Theorem 8.5.5, application of the fixed-point Theorem 8.5.4 shows that there is a unique $\boldsymbol{v}^* \in \mathcal{V}$ such that $\mathscr{L}\boldsymbol{v}^* = \boldsymbol{v}^*$. This is also the optimal value function due to Theorem 8.5.5.The second part of the theorem follows from the first part when considering only a single policy π (which then is optimal). □

8.5.4 MDP Algorithms

Let us now look at three basic algorithms for solving a known Markov decision proces. The first, *value iteration*, is a simple extension of the backwards induction algorithm to the infinite horizon case.

Value iteration

In this version of the algorithm, we assume that rewards are depepndent only on the state. An algorithm for the case where reward only depends on the state can be obtained by replacing $r(s, a)$ with $r(s)$.

Algorithm 6 Value iteration

Input μ, $\mathcal{S}$.
Initialise $\boldsymbol{v}_0 \in \mathcal{V}$.
for $n = 1, 2, \ldots$ **do**
  for $s \in \mathcal{S}_n$ **do**
    $\pi_n(s) = \arg\max_{a\in\mathcal{A}} \left\{r(s, a) + \gamma \sum_{s'\in\mathcal{S}} P_\mu(s' \mid s, a)\boldsymbol{v}_{n-1}(s')\right\}$
    $\boldsymbol{v}_n(s) = r(s, \pi_n(s)) + \gamma \sum_{s'\in\mathcal{S}} P_\mu(s' \mid s, \pi_n(s))\boldsymbol{v}_{n-1}(s')$
  end for
  break if `termination-condition` is met
end for
Return π_n, V_n.

Theorem 8.5.7. *The value iteration algorithm satisfies*

- $\lim_{n\to\infty} \|\boldsymbol{v}_n - V^*\| = 0$.
- *For each* $\epsilon > 0$ *there exists* $N_\epsilon < \infty$ *such that for all* $n \geq N_\epsilon$

$$\|\boldsymbol{v}_{n+1} - \boldsymbol{v}_n\| \leq \epsilon(1-\gamma)/2\gamma. \tag{8.5.7}$$

- *For* $n \geq N_\epsilon$ *the policy* π_ϵ *that takes action* $\arg\max_a r(s,a) + \gamma \sum_j p(j|s,a)\boldsymbol{v}_n(s')$ *is* ϵ*-optimal.*
- $\|\boldsymbol{v}_{n+1} - V^*_\mu\| < \epsilon/2$ *for* $n > N_\epsilon$.

Proof. The first two statements follow from the fixed-point Theorem 8.5.4. Now note that

$$\|V^{\pi_\epsilon}_\mu - V^*_\mu\| \leq \|V^{\pi_\epsilon}_\mu - \boldsymbol{v}_n\| + \|\boldsymbol{v}_n - V^*_\mu\|$$

We can bound these two terms easily:

$$\begin{aligned}
\left\|V^{\pi_\epsilon}_\mu - \boldsymbol{v}_{n+1}\right\| &= \left\|\mathscr{L}_{\pi_\epsilon} V^{\pi_\epsilon}_\mu - \boldsymbol{v}_{n+1}\right\| \\
&\leq \left\|\mathscr{L}_{\pi_\epsilon} V^{\pi_\epsilon}_\mu - \mathscr{L}\boldsymbol{v}_{n+1}\right\| + \|\boldsymbol{v}_{n+1} - \boldsymbol{v}_{n+1}\| \\
&= \left\|\mathscr{L}_{\pi_\epsilon} V^{\pi_\epsilon}_\mu - \mathscr{L}_{\pi_\epsilon}\boldsymbol{v}_{n+1}\right\| + \|\mathscr{L}\boldsymbol{v}_{n+1} - \mathscr{L}\boldsymbol{v}_n\| \\
&\leq \gamma \left\|V^{\pi_\epsilon}_\mu - \boldsymbol{v}_{n+1}\right\| + \|\boldsymbol{v}_{n+1} - \boldsymbol{v}_n\|.
\end{aligned}$$

An analogous argument gives the same bound for the second term $\|\boldsymbol{v}_n - V^*_\mu\|$. Then, rearranging we obtain

$$\|V^{\pi_\epsilon} - \boldsymbol{v}_{n+1}\| \leq \frac{\gamma}{1-\gamma}\|\boldsymbol{v}_{n+1} - \boldsymbol{v}_n\|, \qquad \|\boldsymbol{v}_{n+1} - V^*_\mu\| \leq \frac{\gamma}{1-\gamma}\|\boldsymbol{v}_{n+1} - \boldsymbol{v}_n\|,$$

and the third and fourth statements follow from the second statement. □

Theorem 8.5.8 (Value iteration monotonicity)**.** *Let* $\mathcal{V}$ *be a value function space with Bellman operator* $\mathscr{L}$*. Then:*

1. *Let* $\boldsymbol{v}, \boldsymbol{v}' \in \mathcal{V}$ *with* $\boldsymbol{v}' \geq \boldsymbol{v}$*. Then* $\mathscr{L}\boldsymbol{v}' \geq \mathscr{L}\boldsymbol{v}$*.*
2. *Let* $\boldsymbol{v}_{n+1} = \mathscr{L}\boldsymbol{v}_n$*. If there is an* N *s.t.* $\mathscr{L}\boldsymbol{v}_N \leq \boldsymbol{v}_N$*, then* $\mathscr{L}\boldsymbol{v}_{N+k} \leq \boldsymbol{v}_{N+k}$ *for all* $k \geq 0$ *and similarly for* $\geq$*.*

Proof. Let $\pi \in \arg\max_\pi \boldsymbol{r} + \gamma \boldsymbol{P}_{\mu,\pi}\boldsymbol{v}$. Then

$$\mathscr{L}\boldsymbol{v} = \boldsymbol{r} + \gamma\boldsymbol{P}_{\mu,\pi}\boldsymbol{v} \leq \boldsymbol{r} + \gamma\boldsymbol{P}_{\mu,\pi}\boldsymbol{v}' \leq \max_{\pi'} \boldsymbol{r} + \gamma\boldsymbol{P}_{\mu,\pi'}\boldsymbol{v}',$$

where the first inequality is due to the fact that $\boldsymbol{P}\boldsymbol{v} \geq \boldsymbol{P}\boldsymbol{v}'$ for any $\boldsymbol{P}$.

For the second part,

$$\mathscr{L}\boldsymbol{v}_{N+k} = \boldsymbol{v}_{N+k+1} = \mathscr{L}^k\mathscr{L}\boldsymbol{v}_N \leq \mathscr{L}^k\boldsymbol{v}_N = \boldsymbol{v}_{N+k}.$$

since $\mathscr{L}\boldsymbol{v}_N \leq \boldsymbol{v}_N$ by assumption and consequently $\mathscr{L}^k\mathscr{L}\boldsymbol{v}_N \leq \mathscr{L}^k\boldsymbol{v}_N$ by part one of the theorem. □

Thus, value iteration converges monotonically to V^*_μ if $\boldsymbol{r}^0$ satisfies condition 1 of the theorem. If $r \geq 0$, it is sufficient to set $\boldsymbol{r}^0 = \boldsymbol{0}$. Then $\boldsymbol{v}$ is always a lower bound on the optimal value function.

Theorem 8.5.9. *Value iteration converges linearly at rate γ and $O(\gamma^n)$. In addition, for $r \in [0,1]$ and $\boldsymbol{r}^0 = \mathbf{0}$*

$$\|\boldsymbol{v}_n - V_\mu^*\| \leq \frac{\gamma^n}{1-\gamma},$$
$$\|V_\mu^{\pi_n} - V_\mu^*\| \leq \frac{2\gamma^n}{1-\gamma}.$$

Proof. The first part follows from the contraction property (Theorem 8.5.5):

$$\|\boldsymbol{v}_{n+1} - \boldsymbol{v}^*\| = \|\mathscr{L}\boldsymbol{v}_n - \mathscr{L}\boldsymbol{v}^*\| \leq \gamma\|\boldsymbol{v}_n - \boldsymbol{v}^*\|. \tag{8.5.8}$$

Now divide by γ^n to obtain the $O(\gamma^n)$ property. □

Policy iteration

Unlike value iteration, *policy iteration* attempts to iteratively improve a given policy. At each iteration, it calculates the value of the current policy. At the next step, it tries to obtain an improvement by calculating a policy that is greedy with respect to the previous value function. The algorithm described below can be extended to the case when the reward also depends on the action, by replacing $\boldsymbol{r}$ with the policy-dependent reward vector $\boldsymbol{r}_\pi$. The following theorem can also be easily extended to this case.

Algorithm 7 Policy iteration

Input μ, $\mathcal{S}$.
Initialise $\boldsymbol{v}_0$.
for $n = 1, 2, \ldots$ **do**
  $\pi_{n+1} = \arg\max_\pi \{\boldsymbol{r} + \gamma \boldsymbol{P}_\pi \boldsymbol{v}_n\}$ `// policy improvement`
  $\boldsymbol{v}_{n+1} = V_\mu^{\pi_{n+1}}$ `// policy evaluation`
  break if $\pi_{n+1} = \pi_n$.
end for
Return $\pi_n, \boldsymbol{v}_n$.

Theorem 8.5.10. *Let $\boldsymbol{v}_n, \boldsymbol{v}_{n+1}$ be the value vectors generated by policy iteration. Then $\boldsymbol{v}_n \leq \boldsymbol{v}_{n+1}$.*

Proof. From the policy improvement step

$$\boldsymbol{r} + \gamma \boldsymbol{P}_{\pi_{n+1}} \boldsymbol{v}_n \geq \boldsymbol{r} + \gamma \boldsymbol{P}_{\pi_n} \boldsymbol{v}_n, = \boldsymbol{v}_n$$

where the equality is due to the policy evaluation step for π_n. Rearranging, we get $\boldsymbol{r} \geq (\boldsymbol{I} - \gamma \boldsymbol{P}_{\pi_{n+1}})\boldsymbol{v}_n$ and hence

$$(\boldsymbol{I} - \gamma \boldsymbol{P}_{\pi_{n+1}})^{-1} \boldsymbol{r} \geq \boldsymbol{v}_n,$$

noting that the inverse is positive. Since the left side equals $\boldsymbol{v}_{n+1}$ by the policy evaluation step for π_{n+1}, the theorem follows. □

Corollary 8.5.1. *If $\mathcal{S}, \mathcal{A}$ are finite, then policy iteration terminates after a finite number of iterations and returns an optimal policy.*

Proof. There is only a finite number of policies, and since policies in policy iteration are monotonically improving, the algorithm must stop after finitely many iterations. Finally, the last iteration satisfies

$$\boldsymbol{v}_n = \max_{\pi} \boldsymbol{r} + \gamma \boldsymbol{P}_\pi \boldsymbol{v}_n. \tag{8.5.9}$$

Thus $\boldsymbol{v}_n$ solves the optimality equation. □

Modified policy iteration

The astute reader will have noticed that it may be not necessary to fully evaluate the improved policy. In fact, we can take advantage of that to speed up policy iteration. Thus, a simple variant of policy iteration involves doing only a k-step update for the policy evaluation step. For $k = 1$, the algorithm becomes identical to value iteration.

Algorithm 8 Modified policy iteration

Input μ, $\mathcal{S}$.
Initialise $\boldsymbol{v}_0$.
for $n = 1, 2, \ldots$ **do**
  $\pi_n = \arg\max_\pi \boldsymbol{r} + \gamma \boldsymbol{P}_\pi \boldsymbol{v}_{n-1}$ `// policy improvement`
  $\boldsymbol{v}_n = \mathscr{L}_{\pi_n}^k \boldsymbol{v}_{n-1}$ `// partial policy evaluation`
  break if $\pi_n = \pi_{n+1}$.
end for
Return $\pi_n, \boldsymbol{v}_n$.

Geometric view

Definition 8.5.3. *Difference operator The difference operator is defined as*

$$\mathscr{B}\boldsymbol{v} \triangleq \max_{\pi} \{\boldsymbol{r} + (\gamma \boldsymbol{P}_\pi - \boldsymbol{I})\boldsymbol{v}\} = \mathscr{L}\boldsymbol{v} - \boldsymbol{v}. \tag{8.5.10}$$

Thus the optimality equation can be rewritten as

$$\mathscr{B}\boldsymbol{v} = 0. \tag{8.5.11}$$

For any $\boldsymbol{v} \in \mathcal{V}$, define:

$$\Pi_{\boldsymbol{v}} \triangleq \arg\max_{\pi \in \Pi} \{\boldsymbol{r} + (\gamma \boldsymbol{P}_\pi - \boldsymbol{I})\boldsymbol{v}\}$$

to be the set of $\boldsymbol{v}$-improving policies.

Theorem 8.5.11. *For any $\boldsymbol{v}, \boldsymbol{v}' \in \mathcal{V}$ and $\pi \in \Pi_{\boldsymbol{v}}$*

$$\mathscr{B}\boldsymbol{v} \geq \mathscr{B}\boldsymbol{v} + (\gamma \boldsymbol{P}_\pi - \boldsymbol{I})(\boldsymbol{v}' - \boldsymbol{v}). \tag{8.5.12}$$

Proof. By definition, $\mathscr{B}\boldsymbol{v}' \geq \boldsymbol{r} + (\gamma \boldsymbol{P}_\pi - \boldsymbol{I})\boldsymbol{v}'$, while $\mathscr{B}\boldsymbol{v} = \boldsymbol{r} + (\gamma \boldsymbol{P}_\pi - \boldsymbol{I})\boldsymbol{v}$. Subtracting the latter from the former gives the result. □

Equation (8.5.12) is similar to the convexity equation. In fact, we can have a nice geometric view of this.

Theorem 8.5.12. *Let $\{\boldsymbol{v}_n\}$ be the sequence of value vextors obtained from policy iteration. Then for any $\pi \in \Pi_{\boldsymbol{v}_n}$,*

$$\boldsymbol{v}_{n+1} = \boldsymbol{v}_n - (\gamma \boldsymbol{P}_\pi - \boldsymbol{I})^{-1} \mathscr{B} \boldsymbol{v}_n. \tag{8.5.13}$$

Proof. By definition, we have for $\pi \in \Pi_{\boldsymbol{v}_n}$

$$\begin{aligned} \boldsymbol{v}_{n+1} &= (\boldsymbol{I} - \gamma \boldsymbol{P}_\pi)^{-1} \boldsymbol{r} - \boldsymbol{v}_n + \boldsymbol{v}_n \\ &= (\boldsymbol{I} - \gamma \boldsymbol{P}_\pi)^{-1} [\boldsymbol{r} - (\boldsymbol{I} - \gamma \boldsymbol{P}_\pi) \boldsymbol{v}_n] + \boldsymbol{v}_n. \end{aligned}$$

Since $\boldsymbol{r} - (\boldsymbol{I} - \gamma \boldsymbol{P}_\pi)\boldsymbol{v}_n = \mathscr{B}\boldsymbol{v}_n$ the claim follows. □

Temporal-Difference Policy Iteration

Temporal-difference policy iteration replaces the next-step value with an approximation $\boldsymbol{v}_n$. First, policy improvement is used to obtain the next policy given our approximation:

$$\mathscr{L}_{\pi_{n+1}} \boldsymbol{v}_n = \mathscr{L} \boldsymbol{v}_n. \tag{8.5.14}$$

The method uses the temporal difference error, defined as:

$$d_n(i, j) = \boldsymbol{v}_n(i) - [\boldsymbol{r}(i) + \gamma \boldsymbol{v}_n(j)]. \qquad \text{(temporal difference error)}$$

This can be seen as the difference in the estimate when we move from state i to state j. Note the similarity to the difference operator in modified policy iteration. The idea of the temporal-difference policy iteration is to use d_n as the one-stage reward for a λ-discounted problem. Then one can write the value vector and update for this problem as

$$\boldsymbol{\tau}_n(i) = \sum_{t=0}^{\infty} \mathbb{E}_{\pi_n, \mu} \left[(\gamma\lambda)^m d_n(s_t, s_{t+1}) \mid s_0 = i \right], \tag{8.5.15}$$

$$\boldsymbol{v}_{n+1} = \boldsymbol{v}_n + \boldsymbol{\tau}_n. \tag{8.5.16}$$

Summarizing, we obtain the following algorithm:

Algorithm 9 Temporal-Difference Policy Iteration

Input μ, $\mathcal{S}$, λ.
Initialise $\boldsymbol{v}_0$.
for $n = 1, 2, \ldots$ **do**
  $\pi_n = \arg\max_\pi \boldsymbol{r} + \gamma \boldsymbol{P}_\pi \boldsymbol{v}_{n-1}$ `// policy improvement`
  $\boldsymbol{v}_n = \boldsymbol{v}_{n-1} + \boldsymbol{\tau}_k$ `// temporal difference update.`
  break if $\pi_n = \pi_{n=1}$.
end for
Return $\pi_n, \boldsymbol{v}_n$.

In fact, $\boldsymbol{v}_{n+1}$ is the unique fixed point of the following equation:

$$\mathscr{D}_n \boldsymbol{v} \triangleq (1 - \lambda) \mathscr{L}_{\pi_{n+1}} \boldsymbol{v}_n + \lambda \mathscr{L}_{\pi_{n+1}} \boldsymbol{v}. \qquad \text{(fixed point)}$$

In other words, the new value vector is moved only partially towards the direction of the Bellman update. In fact, for $\lambda = 1$, this becomes identical to standard policy iteration. For $\lambda = 0$, one obtains standard value iteration.

Linear programming

We shall now formulate the problem of finding an optimal value function for a given Markov decision process in terms of a linear program. Recall that if

$$\boldsymbol{v} \geq \mathscr{L}\boldsymbol{v}$$

then $\boldsymbol{v} \geq V^*_\mu$. In order to transform this into a linear program, we must first define a scalar function to minimise. We can do this by selecting some arbitrary distribution on the states $\boldsymbol{y} \in \Delta^{|\mathcal{S}|}$. Then we can write the following linear program.

Primal linear program

$$\min_{\boldsymbol{v}} \boldsymbol{y}^\top \boldsymbol{v},$$

such that

$$\boldsymbol{v}(s) - \gamma \boldsymbol{p}_{s,a}^\top \boldsymbol{v} \geq r(s,a), \qquad \forall a \in \mathcal{A}, s \in \mathcal{S}.$$

Note that the inequality condition is equivalent to $\boldsymbol{v} \geq \mathscr{L}\boldsymbol{v}$. Consequently, the problem is to find the smallest $\boldsymbol{v}$ that satisfies this inequality. When $\mathcal{A}, \mathcal{S}$ are finite, it is easy to see that this will be the optimal value function and the Bellman equation is satisfied.

It also pays to look at the dual linear program, which is in terms of a maximisation. This time, instead of finding the minimal upper bound on the value function, we find the maximal cumulative discounted state-action visits $x(s,a)$ that are consistent with the transition kernel of the process.

Dual linear program

$$\max_x \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} r(s,a) x(s,a)$$

such that $x \in \mathbb{R}_+^{|\mathcal{S} \times \mathcal{A}|}$ and

$$\sum_{a \in \mathcal{A}} x(j,a) - \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \gamma p(j \mid s,a) x(s,a) = y(j) \qquad \forall j \in \mathcal{S}.$$

with $\boldsymbol{y} \in \Delta^{|\mathcal{S}|}$.

In this case, x can be interpreted as the discounted sum of state-action visits, as proved by the following theorem.

Theorem 8.5.13.

$$x_\pi(s,a) = \mathbb{E}_{\pi,\mu}\left\{\sum \gamma^n \mathbb{I}\{s_t = s, a_t = a \mid s_0 \sim y\}\right\}$$

is a feasible solution to the dual problem. On the other hand, if x is a feasible solution to the dual problem then $\sum_a x(s,a) > 0$. Finally, if we define the

strategy

$$\pi(a \mid s) = \frac{x(s,a)}{\sum_{a' \in \mathcal{A}} x(s,a')}$$

then $x_\pi = x$ is a feasible solution.

The equality condition ensures that x is consistent with the transition kernel of the Markov decision process. Consequently, the program can be seen as search among all possible cumulative state-action distributions to find the one giving the highest total reward.

Summary

Markov decision processes
can represent : Shortest path problems, stopping problems, experiment design problems, multi-armed bandit problems, reinforcement learning problems.

Backwards induction (aka value iteration)

- In the class of dynamic programming algorithms.
- Tractable when either the state space $\mathcal{S}$ or the horizon T are small (finite).

Optimal decisions and Bayesian reinforcement learning

- A known environment is represented as an MDP.
- Bandit problems can be solved by representing them as infinite-state MDPs.
- In general, an unknown environment can be represented as a distribution over MDPs.
- The decision problem can again be formulated as an infinite-state MDP.

8.6 Further reading

See the last chapter of [DeGroot, 1970] for further information on the MDP formulation of bandit problems in the decision theoretic setting. This was explored in more detail in Duff's PhD thesis [Duff, 2002]. When the number of (information) states in the bandit problem is finite, Gittins [1989] has proven

that it is possible to formulate simple index policies. However, this is not generally applicable. Easily computable, near-optimal heuristic strategies for bandit problems will be given in Chapter 12. The decision-theoretic solution to the unknown MDP problem will be given in Chapter 11.

Further theoretical background on Markov decision processes, including many of the theorems in Section 8.5, can be found in [Puterman, 1994]. Chapter 2 of Bertsekas and Tsitsiklis [1996] gives a quick overview of MDP theory from the operator perspective. The introductory reinforcement learning book of Sutton and Barto [1998] also explains the basic Markov decision process framework.

Chapter 9

Reinforcement learning and stochastic approximation

9.1 Introduction

In this chapter, we consider the actual problem of reinforcement learning. Up to now, we have only examined a solution method for bandit problems, which are a special case of the general reinforcement learning problem. The Bayesian decision-theoretic solution is to *reduce* the bandit problem to a *Markov decision process* which can then be solved with backwards induction.

We also have seen that Markov decision processes can be used to *describe environments* in more general reinforcement learning problems. When our knowledge of the MDP describing these problems is perfect, then we can employ a number of standard algorithms to find the optimal policy. However, in the actual reinforcement learning problem, the model of the environment is *unknown*. However, as we shall see later, ideas from both cases can be used to solve the general reinforcement learning problem.

The main idea we explore in this chapter is that we can perform approximate dynamic programming algorithms, by replacing the actual unknown dynamics of the Markov decision process with estimates. The estimates can be improved by drawing samples from the actual environment, either by actually acting within the environment or using a simulator. In both cases we end up with a number of algorithms that can be used for reinforcement learning. Although may not be performing as well as the Bayes-optimal solution, these have a low enough computational complexity that they are worth investigating in practice.

9.1.1 Bandit problems

The stochastic n-armed bandit problem

Let us return to the example of bandit problems. As before, we have n actions corresponding to probability distributions P_i on the real numbers.

$$\mathscr{P} = \{P_i \mid i = 1, \dots, n\}.$$

At each time-step t we select an action a_t, obtaining a random reward distributed according to:

$$r_t \mid a_t = i \sim P_i.$$

Our objective is to find a policy π maximising the expected total reward.

$$\mathbb{E}_\pi U_t = \mathbb{E}_\pi \sum_{k=t}^{T} r_k, \qquad a_t^* \triangleq \max \{\mathbb{E}(r_t \mid a_t = i) \mid i = 1, \dots, n\}.$$

Had we known the distribution, we could simply always the maximising action, as the expected reward of the i-th action can be easily calculated from P_i and the reward only depends on our current action. The situation is similar when $\mathscr{P}$ is a parametric family unknown parameter ω^*, outlined below.

$$\mathscr{P} = \{P_i(\cdot \mid \omega) \mid \omega \in \Omega\}, \qquad r_t \mid a_t = i, \omega^* = \omega \sim P_i(r \mid \omega^*). \tag{9.1.1}$$

If in addition we have a subjective belief ξ over Ω, we could (as explained in Sec. 8.2) in principle calculate the policy maximising the ξ-expected utility:

$$\mathbb{E}_\xi^\pi U_t = \mathbb{E}_\xi^\pi \sum_{k=t}^{T} r_k. \tag{9.1.2}$$

This of course will now have to be a history-dependent policy. In the remainder of this section, we shall examine algorithms algorithms which eventually convergence to the optimal action, but for which we cannot always guarantee a good initial behaviour.

9.1.2 Estimation and Robbins-Monro approximation

The basic idea of the Robbins-Monro stochastic approximation algorithm [Robbins and Monro, 1951] is to maintain a set of *point estimates* of a parameter we want to approximate and perform *random* steps that on average move towards the solution, in a way to be made more precise later. It can in fact be seen as a generalisation of stochastic gradient descent.

Algorithm 10 Robbins-Monro bandit algorithm

1: **input** Step-sizes $(\alpha_t)_t$, initial estimates $(\mu_{i,0})_i$, policy π.
2: **for** $t = 1, \ldots, T$ **do**
3: Take action $a_t = i$ with probability $\pi(i \mid a_1, \ldots, a_{t-1}, r_1, \ldots, r_{t-1})$.
4: Observe reward r_t.
5: $\mu_{t,i} = \alpha_{i,t} r_t + (1 - \alpha_{i,t})\mu_{i,t-1}$ `// estimation step`
6: $\mu_{t,i} = \mu_{j,t-1}$ for $j \neq i$.
7: **end for**
8: **return** $\boldsymbol{\mu_T}$

A simple Robbins-Monro algorithm for the n-armed bandit problem is given in Algorithm 10. The input is a particular policy π, that gives us a probability over the next actions given the observed history, a set of initial estates $\mu_{i,0}$ for the bandit means, and a sequence of step sizes α.

If you examine the updates carefully, you will be able to find what the cost function you are trying to minimise is. This simple update rule can be seen as trying to minimise the expected squared error between your estimated reward, and the random reward obtained by each bandit. Consequently, the variance of the reward of each bandit plays an important role.

The step-sizes α must obey certain constraints in order for the algorithm to work, in particular it must decay neither too slowly, nor too fast. There is one particular choice, for which our estimates are in fact the mean estimate of the expected value of the reward for each action i, which is a natural choice if the bandits are stationary.

The other question is what policy to use to take actions. We must take all actions often enough, so that we have good estimates for the expected reward of every bandit. One simple way to do it is to play the apparently best bandit most of the time, but to sometimes select bandits randomly. This is called ϵ-greedy action selection. This ensures that all actions are tried a sufficient number of times.

Definition 9.1.1. *ϵ-greedy action selection*

$$\hat{\pi}^*_\epsilon \triangleq (1 - \epsilon_t)\hat{\pi}^*_t + \epsilon_t \mathcal{U}nif(\mathcal{A}), \tag{9.1.3}$$

$$\hat{\pi}^*_t(i) = \mathbb{I}\left\{i \in \hat{\mathcal{A}}^*_t\right\} / |\hat{\mathcal{A}}^*_t|, \qquad \hat{\mathcal{A}}^*_t = \arg\max_{i \in \mathcal{A}} \mu_{t,i} \tag{9.1.4}$$

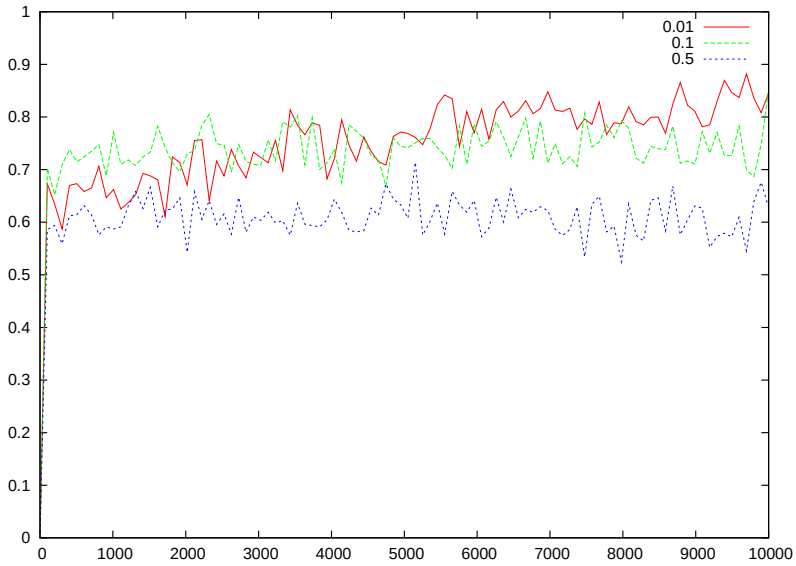


Figure 9.1: $\epsilon_t = 0.1$, $\alpha \in \{0.01, 0.1, 0.5\}$.

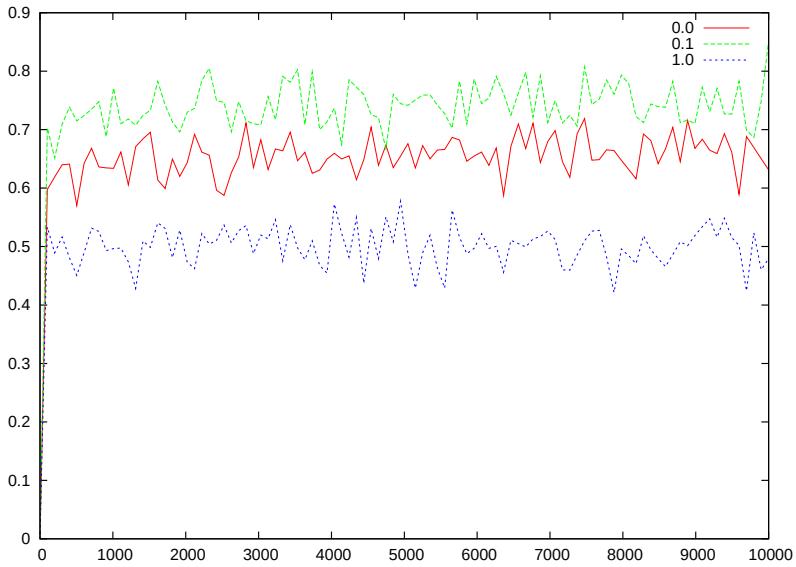


Figure 9.2: $\epsilon_t = \epsilon$, $\alpha = 0.1$.

The main two parameters of the algorithm are randomness ϵ-greedy action selection and the step-size. Figures 9.1 and 9.2 show the average reward obtained, if we keep the step size α or the randomness ϵ fixed, respectively. We see that there the choice of values really affects convergence.

For a fixed ϵ, we find that larger values of α tend to give a better result eventually, while smaller values have a better initial performance. This is a

natural trade-off, since large α appears to "learn" fast, but it also "forgets" quickly. That is, for a large α, our estimates mostly depend upon the last few rewards observed.

Things are not so clear-cut for the choice of ϵ. We see that the choice of $\epsilon = 0$, is significantly worse than $\epsilon = 0.1$. So, that appears to suggest that there is an optimal level of exploration. How should that be determined? Ideally, we should be able to to use the decision-theoretic solution seen earlier, but perhaps a good heuristic way of choosing ϵ may be good enough.

9.1.3 The theory of the approximation

Here we quickly review some basic results of stochastic approximation theory. Complete proofs can be found in Bertsekas and Tsitsiklis [1996]. The main question here is Consider the algorithm

$$\mu_{t+1} = \mu_t + \alpha_t z_{t+1}. \tag{9.1.5}$$

Here μ_t is our estimate, α_t is a step-size and z_t is an observation. Let $h_t = \{\mu_t, z_t, \alpha_t, \ldots\}$ be the history of the algorithm.

There are two types of convergence conditions. The first focuses on continuity and smoothness properties, and the second on contraction properties of operators. Here, we concentrate on the first case, which also has applications in other areas.

Assumption 9.1.1. *Assume a function $f : \mathbb{R}^n \to \mathbb{R}$ such that:*

(i) $f(x) \geq 0$ *for all* $x \in \mathbb{R}^n$.

(ii) (Lipschitz derivative) *f is continuously differentiable (i.e. the derivative ∇f exists and is continuous) and $\exists L > 0$ such that:*

$$\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\|, \qquad \forall\ x, y \in \mathbb{R}^n$$

(iii) (Pseudo-gradient) $\exists c > 0$ *such that:*

$$c \|\nabla f(\mu_t)\|^2 \leq -\nabla f(\mu_t)^\top \mathbb{E}(z_{t+1} \mid h_t), \qquad \forall\ t.$$

(iv) $\exists K_1, K_2 > 0$ *such that*

$$\mathbb{E}(\|z_{t+1}\|^2 \mid h_t) \leq K_1 + K_2 \|\nabla f(\mu_t)\|^2$$

Condition (ii) is a very basic condition for convergence. It basically ensures that the function is well-behaved, so that gradient-following methods can easily find the minimum. Condition (iii) combines two assumptions in one. Firstly, that expected direction of update always decreases cost, and secondly that the squared norm of the gradient is not too large relative to the size of the update. Finally, condition (iv) ensures that update is bounded in expectation relative to the gradient.

One can see how putting together the last two conditions ensures that the expected direction of our update is correct, and that its norm is bounded.

Theorem 9.1.1. *For the algorithm*

$$\mu_{t+1} = \mu_t + \alpha_t z_{t+1},$$

where $\alpha_t \geq 0$ *satisfy*

$$\sum_{t=0}^{\infty} \alpha_t = \infty, \qquad \sum_{t=0}^{\infty} \alpha_t^2 < \infty, \tag{9.1.6}$$

and under Assumption 9.1.1, with probability 1:

1. *The sequence* $\{f(\mu_t)\}$ *converges.*
2. $\lim_{t\to\infty} \nabla f(\mu_t) = 0$.
3. *Every limit point* μ^* *of* μ_t *satisfies* $\nabla f(\mu^*) = 0$.

A demonstration

Figure 9.3 demonstrates the convergence, or lack thereof, of our estimates μ_t to the expected value of a random variable with mean 0.5, for three different step-size schedules, with update direction:

$$z_{t+1} = x_{t+1} - \mu_t.$$

The first one, $\alpha_t = 1/t$, satisfies both assumptions. The second one, $\alpha_t = 1/\sqrt{t}$, reduces too slowly, and the third one, $\alpha_t = t^{-3/2}$, approaches zero too fast.

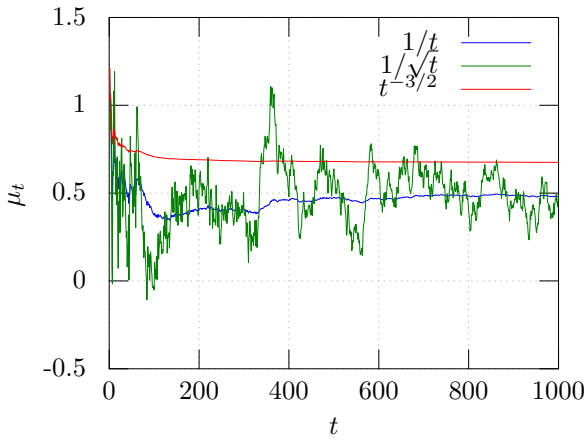


Figure 9.3: Estimation of the expectation of $x_t \sim \mathcal{N}(0.5, 1)$ using use three step-size schedules.

9.2 Dynamic problems

The dynamic setting presents one essential difference. Our policy now affects which sequences of states we observe. Otherwise, the algorithmic structure remains the same and is described below.

Algorithm 11 Generic reinforcement learning algorithm

input Update-rule $f : \Theta \times \mathcal{S}^2 \times \mathcal{A} \times \mathscr{R} \to \Theta$, initial parameters $\theta_0 \in \Theta$, policy $\pi : \mathcal{S} \times \Theta \to \mathfrak{D}(\mathcal{A})$.
for $t = 1, \dots, T$ **do**
  $a_t \sim \pi(\cdot \mid \theta_t, s_t)$ `// take action`
  Observe reward r_{t+1}, state s_{t+1}.
  $\theta_{t+1} = f(\theta_t, s_t, a_t, r_{t+1}, s_{t+1})$ `// update estimate`
end for

Questions

- What should we estimate? For example, θ_t could be describing a posterior distribution over MDPs, or a distribution over parameters.
- What policy should we use? For example, we could try and use the Bayes-optimal policy with respect to θ, or some heuristic policy.

Example 9.2.1 (The chain task)**.** *The chain task has two actions and five states, as shown in Fig. 9.4. The reward in the leftmost state is* 0.2 *and* 1.0 *in the rightmost state, and zero otherwise. The first action (dashed, blue) takes you to the right, while the second action (solid, red) takes you to the first state. However, there is a probability* 0.2 *with which the actions have the opposite effects. The value function of the chain task for a discount factor* $\gamma = 0.95$ *is shown in Table 9.1.*

The chain task is a very simple, but well-known task, used to test the efficacy of reinforcement learning algorithms. A variant of this task, with action-dependent rewards was used by [Dearden et al., 1998].

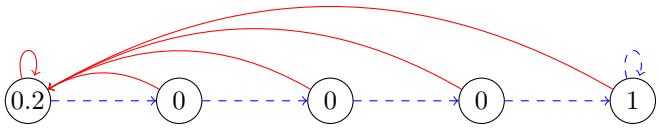


Figure 9.4: The chain task

s	s_1	s_2	s_3	s_4	s_5
$V^*(s)$	7.6324	7.8714	8.4490	9.2090	10.209
$Q^*(s,1)$	7.4962	7.4060	7.5504	7.7404	8.7404
$Q^*(s,2)$	7.6324	7.8714	8.4490	9.2090	10.2090

Table 9.1: The chain task's value function for $\gamma = 0.95$

9.2.1 Monte-Carlo policy evaluation and iteration

The simplest algorithm is Monte-Carlo policy evaluation. In the standard setting, we can the value function for every state by approximating the expectation with the sum of rewards obtained over multiple trajectories starting from each state. The k-th trajectory starts from some initial state $s_0 = s$ and the next states are sampled as follows

$$a_t^{(k)} \sim \pi(a_t \mid h_t).r_t^{(k)} \quad \sim \mathbb{P}_\mu(r_t \mid s_t^{(k)}, a_t^{(k)}) s_{t+1}^{(k)} \sim \mathbb{P}_\mu(s_{t+1} \mid s_t^{(k)}, a_t^{(k)}). \quad (9.2.1)$$

Then the value function satisfies

$$V_\mu^\pi(s) \triangleq \mathbb{E}_\mu^\pi(U \mid s_1 = s) \approx \frac{1}{K} \sum_{k=1}^{K} \sum_{t=1}^{T} r_t^{(k)},$$

where $r_t^{(k)}$ is the sequence of rewards obtained from the k-th trajectory.

Algorithm 12 Stochastic policy evaluation

```
input Initial parameters v_0, Markov policy π.
for s ∈ S do
    s_1 = s.
    for k = 1, ..., K do
        Run policy π for T steps.
        Observe utility U_k = Σ_t r_t.
        Update estimate v_{k+1}(s) = v_k(s) + α_k(U_k − v_k(s))
    end for
end for
return v_K
```

For $\alpha_k = 1/k$ and iterating over all $\mathcal{S}$, this is the same as Monte-Carlo policy evaluation.

A well-known algorithm for getting an optimal policy is policy iteration, Algorithm 7 in Section 8.5.4. This consists of estimating the value of a particular policy, and then trying to get an improved policy using this value. We can still apply the same principle for the case where we cannot exactly evaluate a policy. This is called approximate policy iteration. Unfortunately, approximate policy iteration does not necessarily converge without strong conditions on each approximation step.

Algorithm 13 Approximate policy iteration

```
input Initial parameters v_0, inital Markov policy π_0, stochastic estimator
    f.
for i = 1, ..., N do
    Get estimate v_i = f(v_{i−1}, π_{i−1}).
    Calculate new policy π_i = arg max_π L v_i.
end for
```

Monte Carlo update

Note that $s_1, \dots, s_T$ contains $s_k, \dots, s_T$. This suggests that we could update the value of all encountered states, as we also have the utility starting from each state. We call this algorithm every-visit Monte-Carlo.

Algorithm 14 Every-visit Monte-Carlo update

input Initial parameters $\boldsymbol{v}_k$, trajectory $s_1, \dots, s_T$, rewards $r_1, \dots, r_T$ visit counts n.
for $t = 1, \dots, T$ **do**
  $U_t = \sum_{t=1}^{T} r_t$.
  $n_t(s_t) = n_{t-1}(s_t) + 1$
  $\boldsymbol{v}_{t+1}(s_t) = \boldsymbol{v}_t(s) + \alpha_{n_t(s_t)}(s_t)(U_t - \boldsymbol{v}_t(s_t))$
  $n_t(s) = n_{t-1}(s)$, $\boldsymbol{v}_t(s) = \boldsymbol{v}_{t-1}(s)$ $\forall s \neq s_t$.
end for
return $\boldsymbol{v}_K$

For a proper Monte-Carlo estimate, when the environment is stationary $\alpha_{n_t(s_t)}(s_t) = 1/n_t(s_t)$. Nevertheless, this type of estimate can be biased, as can be seen by the following example.

Example 9.2.2. *Consider a two-state chain with* $\mathbb{P}(s_{t+1} = 1 \mid s_t = 0) = \delta$ *and* $\mathbb{P}(s_{t+1} = 1 \mid s_t = 1) = 1$*, and reward* $r(1) = 1$*,* $r(0) = 0$*. Then the every-visit estimate is biased.*

Let us consider the discounted setting. Then value of the second state is $1/(1-\gamma)$ and the value of the first state is $\sum_k (\delta\gamma)^k = 1/(1-\delta\gamma)$. Consider the every-visit Monte-Carlo update. The update is going to be proportional to the number of steps you spend in that state. In order to avoid the bias, we must instead look at only the first visit to every state. This eliminates the dependence between states.

Unbiased Monte-Carlo update

Algorithm 15 First-visit Monte-Carlo update

input Initial parameters $\boldsymbol{v}_1$, trajectory $s_1, \dots, s_T$, rewards $r_1, \dots, r_T$, visit counts n.
for $t = 1, \dots, T$ **do**
  $U_t = \sum_{t=1}^{T} r_t$.
  $n_t(s_t) = n_{t-1}(s_t) + 1$
  $\boldsymbol{v}_{t+1}(s_t) = \boldsymbol{v}_t(s) + \alpha_{n_t(s_t)}(s_t)(U_t - \boldsymbol{v}_t(s_t))$ if $n_t(s_t) = 1$.
  $n_t(s) = n_{t-1}(s)$, $\boldsymbol{v}_t(s) = \boldsymbol{v}_{t-1}(s)$ otherwise
end for
return $\boldsymbol{v}_{T+1}$

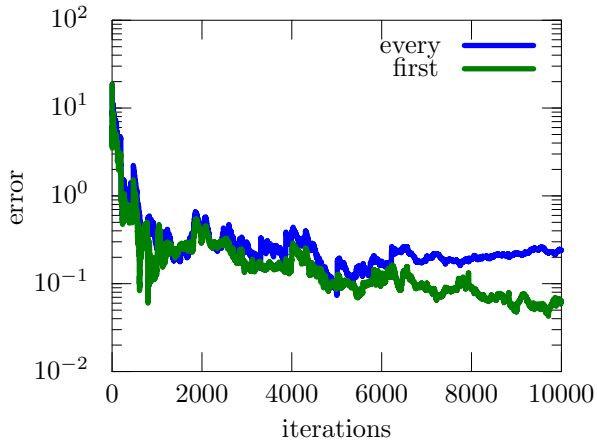


Figure 9.5: Error as the number of iterations n increases, for first and every visit Monte Carlo estimation.

9.2.2 Temporal difference methods

The main idea of temporal differences is to use partial samples of the utility and replace the remaining sample from time t with an estimate of the expected utility after time t. Since there maybe no particular reason to choose a specific t, frequently an exponential distribution t's is used. Let us first look at the usual update when we have the complete utility sample U_k. The full stochastic update is of the form:

$$\boldsymbol{v}_{k+1}(s) = \boldsymbol{v}_k(s) + \alpha(U_k - \boldsymbol{v}_k(s)),$$

Using the *temporal difference error* $d(s_t, s_{t+1}) = \boldsymbol{v}(s_t) - [\boldsymbol{r}(s_t) + \gamma \boldsymbol{v}(s_{t+1})]$, we obtain the update:

$$\boldsymbol{v}_{k+1}(s) = \boldsymbol{v}_k(s) + \alpha \sum_t \gamma^t d_t, \qquad d_t \triangleq d(s_t, s_{t+1}) \tag{9.2.2}$$

Stochastic, incremental, update:

$$\boldsymbol{v}_{t+1}(s) = \boldsymbol{v}_t(s) + \alpha \gamma^t d_t. \tag{9.2.3}$$

We have now converted the full stochastic update into an incremental update that is nevertheless equivalent to the old update. Let us see how we can generalise this to the case where we have a mixture of temporal differences.

Temporal difference algorithm with eligibility traces.

TD(λ).
Recall the temporal difference update when the MDP is given in analytic form.

$$\boldsymbol{v}_{n+1}(i) = \boldsymbol{v}_n i + \boldsymbol{\tau}_n i, \quad \boldsymbol{\tau}_n(i) \triangleq \sum_{t=0}^{\infty} \mathbb{E}_{\pi_n, \mu} \left[(\gamma\lambda)^m d_n(s_t, s_{t+1}) \mid s_0 = i \right].$$

We can convert this to a stochastic update, which results in the well-known TD(λ) algorithm for policy evaluation.

$$\boldsymbol{v}_{n+1}(s_t) = \boldsymbol{v}_n(s_t) + \alpha \sum_{k=t}^{\infty} (\gamma\lambda)^{k-t} d_k. \tag{9.2.4}$$

Unfortunately, this algorithm is only possible to implement offline due to the fact that we are looking at future values.

This problem can be fixed by the backwards-looking Online TD(λ) algorithm. The main idea is to backpropagate changes in future states to previously encountered states. However, we wish to modify older states less than more recent states.

Algorithm 16 Online TD(λ)

input Initial parameters $\boldsymbol{v}_k$, trajectories (s_t, a_t, r_t)
$\boldsymbol{e}_0 = \mathbf{0}$.
for $t = 1, \ldots, T$ **do**
  $d_t \triangleq d(s_t, s_{t+1})$ `// temporal difference`
  $\boldsymbol{e}_t(s_t) = \boldsymbol{e}_{t-1}(s_t) + 1$ `// eligibility increase`
  for $s \in \mathcal{S}$ **do**
    $\boldsymbol{v}_{t+1}(s_t) = \boldsymbol{v}_t(s) + \alpha_t \boldsymbol{e}_t(s) d_t$. `// update all eligible states`
  end for
  $\boldsymbol{e}_{t+1} = \lambda \boldsymbol{e}_t$
end for
return $\boldsymbol{v}_T$

Figure 9.6: Eligibility traces

9.2.3 Stochastic value iteration methods

The main problem we had seen so far with Monte-Carlo based simulation is that we normally require a complete sequence of rewards before updating values. However, in value iteration, we can simply perform a backwards step from all the following states in order to obtain a utility estimate. This idea is explored in stochastic value iteration methods.

The standard value iteration algorithm performs a sweep over the complete state space at each iteration. However, could perform value iteration over an arbitrary sequence of states. For example, we can follow a sequence of states generated from a particular policy. This lends to the idea of *simulation-based* value iteration.

Such state sequences must satisfy various technical requirements. In particular, the policies that generate those state sequences must be *proper* for episodic problems. That is, that all policies should reach a terminating state with probability 1. For discounted non-episodic problems, this is easily achieved by using a geometric distribution for termination time. This ensures that all policies will

be proper. Alternatively, of course, we could simply select starting states with an arbitrary schedule, as long as all states are visited infinitely often in the limit.

However, value iteration also requires the Markov decision process model. The question is whether it is possible to replace the MDP model with some arbitrary estimate. This estimate can itself be obtained via simulation. This leads to a whole new family of stochastic value iteration algorithms. The most important and well-known of these is Q-learning, which uses a trivial empirical MDP model.

Simulation-based value iteration

First, however, we shall discuss the extension of value iteration to the case where we obtain state data from simulation. This allows us to concentrate our estimates to the most useful states.

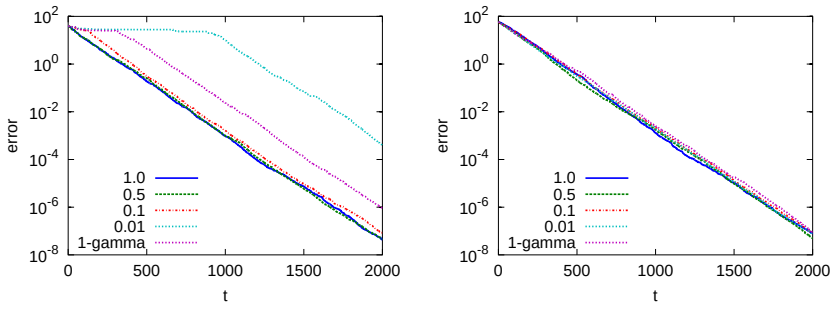
Algorithm 17 shows a generic simulation-based value iteration algorithm, with a uniform restart distribution $\mathcal{U}nif(\mathcal{S})$ and termination probability ϵ.

Algorithm 17 Simulation-based value iteration

Input μ, $\mathcal{S}$.
Initialise $s_t \in \mathcal{S}, \boldsymbol{v}_0 \in \mathcal{V}$.
for $t = 1, 2, \ldots$ **do**
  $s = s_t$.
  $\pi_t(s) = \arg\max_a r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathbb{P}_\mu(s'|s,a) \boldsymbol{v}_{t-1}(s')$
  $\boldsymbol{v}_t(s) = r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathbb{P}_\mu(s'|s,\pi_t(s)) \boldsymbol{v}_{t-1}(s')$
  $s_{t+1} \sim (1-\epsilon) \cdot \mathbb{P}(s_{t+1} \mid s_t = a, \pi_t, \mu) + \epsilon \cdot \mathcal{U}nif(\mathcal{S})$.
end for
Return π_n, V_n.

In the following figures, we can see the error in value function estimation in the chain task when using simulation-based value iteration. It is always a better idea to use an initial value $\boldsymbol{v}_0$ that is an upper bound on the optimal value function, if such a value is known. This is due to the fact that in that case, convergence is always guaranteed when using simulation-based value iteration, as long as the policy that we are using is proper.[1]

[1] In the case of discounted non-episodic problems, this amounts to a geometric stopping time distribution, after which the state is drawn from the initial state distribution.



(a) Pessimistic (b) Optimistic

Figure 9.7: Simulation-based value iteration with pessimistic initial estimates ($\boldsymbol{v}_0 = 0$) and optimistic initial estimates ($\boldsymbol{v}_0 = 20 = 1/(1-\gamma)$), for varying ϵ. Errors indicate $\|\boldsymbol{v}_t - V^*\|_1$.

As can be seen in Figure 9.7, the value function estimation error of simulation-based value iteration is highly dependent upon the initial value function estimate $\boldsymbol{v}_0$ and the exploration parameter ϵ. It is interesting to see uniform sweeps ($\epsilon = 1$) result in the lowest estimation error in terms of the value function L_1 norm.

Q-learning

Simulation-based value iteration can be suitably modified for the actual reinforcement learning problem. Instead of relying on a model of the environment, we replace arbitrary random sweeps of the state-space with the actual state sequence observed in the real environment. We also use this sequence as a simple way to estimate the transition probabilities.

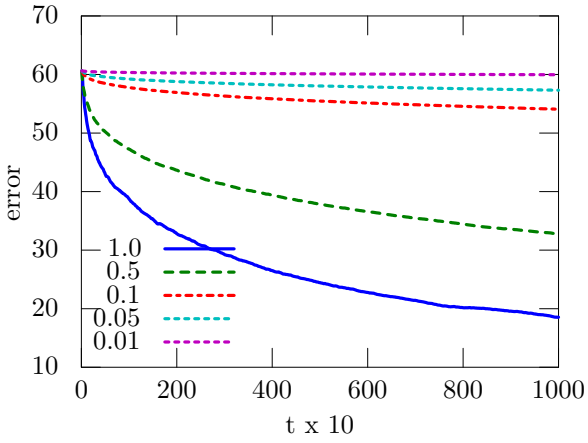
Algorithm 18 Q-learning

1: Input μ, $\mathcal{S}, \epsilon_t, \alpha_t$.
2: Initialise $s_t \in \mathcal{S}, \boldsymbol{q}_0 \in \mathcal{V}$.
3: **for** $t = 1, 2, \ldots$ **do**
4: $\quad s = s_t$.
5: $\quad a_t \sim \hat{\pi}^*_{\epsilon_t}(a \mid s_t, \boldsymbol{q}_t)$
6: $\quad s_{t+1} \sim \mathbb{P}(s_{t+1} \mid s_t = a, \pi_t, \mu)$.
7: $\quad \boldsymbol{q}_{t+1}(s_t, a_t) = (1 - \alpha_t)\boldsymbol{q}_t(s_t, a_t) + \alpha_t[r(s_t) + \boldsymbol{v}_t(s_{t+1})]$, where $\boldsymbol{v}_t(s) = \max_{a \in \mathcal{A}} \boldsymbol{q}_t(s, a)$.
8: **end for**
9: Return π_n, V_n.

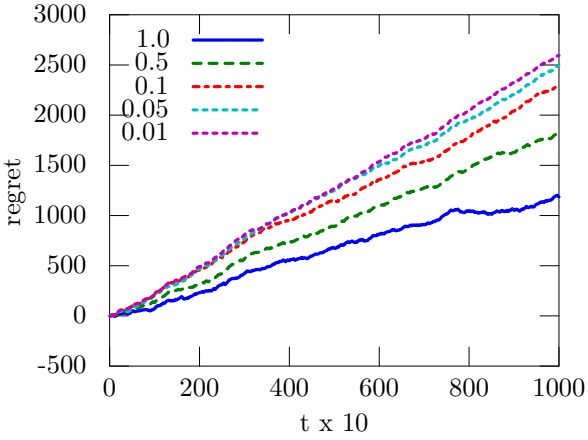
The result is Q-learning, one of the most well-known and simplest algorithms in reinforcement learning. In light of the previous theory, it can be seen as a stochastic value iteration algorithm, where at every step t, given the partial observation (s_t, a_t, s_{t+1}) you have an approximate transition model for the MDP which is as follows:

$$P(s'|s_t, a_t) = \begin{cases} 1, & \text{if } s_{t+1} = s' \\ 0, & \text{if } s_{t+1} \neq s'. \end{cases} \tag{9.2.5}$$

Even though this model is very simplistic, it still seems to work relatively well in practice, and the algorithm is simple to implement. In addition, since we cannot arbitrarily select states in the real environment, we replace the state-exploring parameter ϵ with a time-dependent exploration parameter ϵ_t for the policy we employ on the real environment.



(a) Error



(b) Regret

Figure 9.8: Q-learning with $\boldsymbol{v}_0 = 1/(1-\gamma)$, $\epsilon_t = 1/n_{s_t}$, $\alpha_t \in \alpha n_{s_t}^{-2/3}$.

Figure 9.8 shows the performance of the basic Q-learning algorithm for the Chain task, in terms of value function error and regret. In this particular implementation, we used a polynomially decreasing exploration parameter and step size. Both of these depend on the number of visits to a particular state.

Of course, one could get any algorithm in between pure Q-learning and pure stochastic value iteration. In fact, variants ofthe Q-learning algorithm using eligibility traces (see Section 201) can be formulated in this way.

Generalised stochastic value iteration

Finally, we can generalise the above ideas to the following algorithm. This is an online algorithm, which can be applied directly to a reinforcement learning problem and it includes simulation-based value iteration and Q-learning as special cases. There are three parameters associated with this algorithm. The first is ϵ_t, the exploration amount performed by the policy we follow. The second is α_t, the step size parameter. The third one is σ_t, the state-action distribution. The final parameter is the MDP estimator $\widehat{\mu}_t$. This includes both an estimate of the transition probabilities $\mathbb{P}_{\widehat{\mu}_t}(s' \mid s, a)$ and of the expected reward $r_{\widehat{\mu}_t}(s, a)$.

Algorithm 19 Generalised stochastic value iteration

Input $\widehat{\mu_0}$, $\mathcal{S}, \epsilon_t, \alpha_t$.
Initialise $s_1 \in \mathcal{S}, \boldsymbol{q}_1 \in \mathcal{Q}, \boldsymbol{v}_0 \in \mathcal{V}$.
for $t = 1, 2, \ldots$ **do**
  $a_t \sim f(\hat{\pi}^*_{\epsilon_t}(a \mid s_t, \boldsymbol{q}_t))$
  Observe s_{t+1}, r_{t+1}.
  $\widehat{\mu}_t = \hat{\mu}_{t-1} \mid s_t, a_t, s_{t+1}, r_{t+1}$. `// update MDP estimate.`
  for $s \in \mathcal{S}$, $a \in \mathcal{A}$ **do**
    With probability $\sigma_t(s, a)$ do:

$$\boldsymbol{q}_{t+1}(s, a) = (1 - \alpha_t)\boldsymbol{q}_t(s, a) + \alpha_t \left[r_{\widehat{\mu}_t}(s, a) + \gamma \sum_{s' \in \mathcal{S}} \mathbb{P}_{\widehat{\mu}_t}(s' \mid s, a) \boldsymbol{v}_t(s') \right].$$

    otherwise $\boldsymbol{q}_{t+1}(s, a) = \boldsymbol{q}_t(s, a)$.
    $\boldsymbol{v}_{t+1}(s) = \max_{a \in \mathcal{A}} \boldsymbol{q}_{t+1}(s, a)$,
  end for
end for
Return π_n, V_n.

It is instructive to examine special cases for these parameters. For the case when $\sigma_t = 1$, $\alpha_t = 1$, and when $\widehat{\mu}_t = \mu$, we obtain standard value iteration.

For the case when $\sigma_t(s, a) = \mathbb{I}\{s_t = s \wedge a_t = a\}$ and

$$\mathbb{P}_{\widehat{\mu}_t}(s_{t+1} = s' \mid s_t = s, a_t = a) = \mathbb{I}\{s_{t+1} = s' \mid s_t = s, a_t = a\},$$

it is easy to see that we obtain Q-learning.

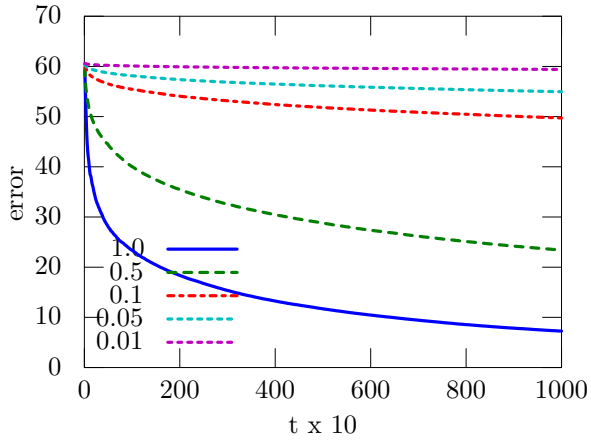
Finally, if we set $\sigma_t(s, a) = \boldsymbol{e}_t(s, a)$, then we obtain a stochastic eligibility-trace Q-learning algorithm similar to $Q(\lambda)$.

Examples

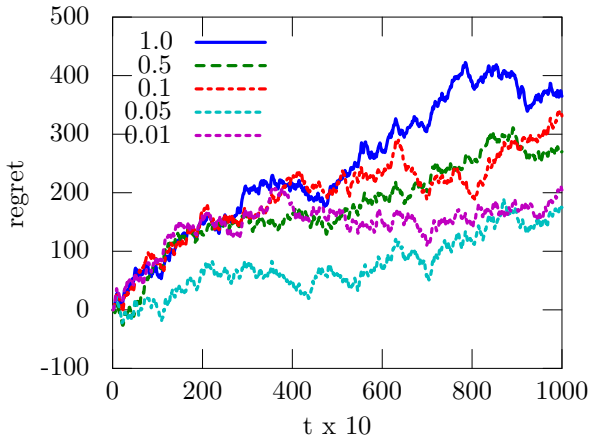
The following examples use a Dirichlet-based estimate for the transition probabilities. If $n_t(s, a, s')$ is the number of times we have visited state s' after taking action a in state s, then the marginal proability of the next step, with a prior $\mathcal{D}ir(\alpha)$ is

$$\frac{n_t(s, a, s') + \alpha}{n_t(s, a) + \alpha|\mathcal{S}|}.$$

Now we can combine this with the simple Q-learning update, where we only modify the value of the current state-action pair. The error and regret are shown in figures 9.9(a) and 9.9(b)



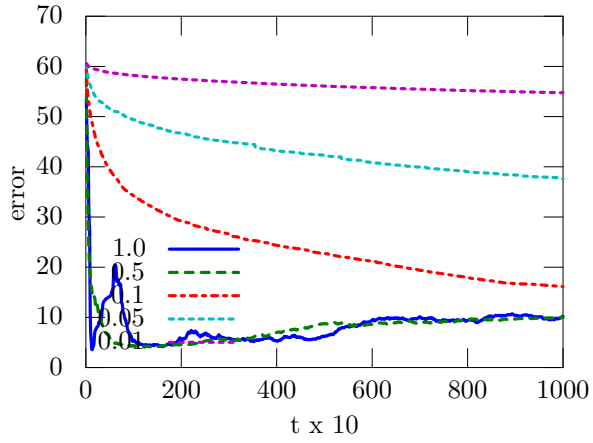
(a) Error



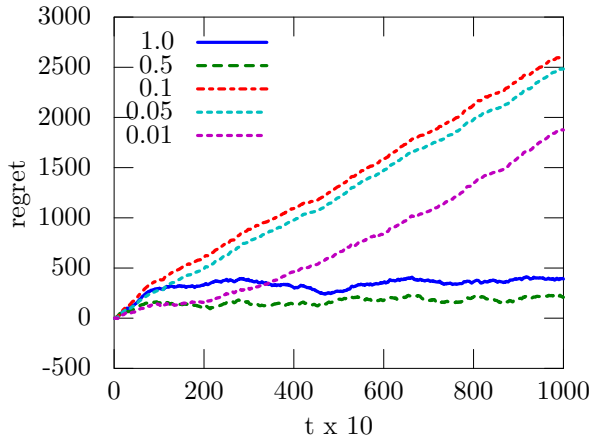
(b) Regret

Figure 9.9: Mode-based Q-learning, i.e. GSVI with Dirichlet model and single-step updates: $\boldsymbol{v}_0 = 1/(1-\gamma)$, $\epsilon_t = 1/n_{s_t}$, $\alpha_t \in \alpha n_{s_t}^{-2/3}$.

In the following example, we again use a Dirichlet estimate but we perform a uniform sweep over the state space, i.e. $\sigma_t = 1$. 9.10(b) and 9.10(a)



(a) Error



(b) Regret

Figure 9.10: GSVI with Dirichlet model estimation and a uniform sweep over the state-space with $\boldsymbol{v}_0 = 1/(1-\gamma)$, $\epsilon_t = 1/n_{s_t}$, $\alpha_t \in \alpha n_{s_t}^{-2/3}$.

Chapter 10

Approximate dynamic programming

10.1 Introduction

In this chapter, we consider approximate dynamic programming. This includes all methods with approximations in the maximisation step, methods where the value function used is approximate, or methods where the policy used is some approximation to the optimal policy.

We first consider the case where we have an approximate value function. Let $\boldsymbol{u} \in \mathcal{V}$ be an approximate optimal value function obtained via some arbitrary method. Then we can define the greedy policy with respect to it as follows:

Definition 10.1.1 ($\boldsymbol{u}$-greedy policy and value function)**.**

$$\pi^*_{\boldsymbol{u}} \in \arg\max_{\pi} \mathscr{L}_\pi \boldsymbol{u}, \qquad \boldsymbol{v}^*_{\boldsymbol{u}} = \mathscr{L} \boldsymbol{u}, \tag{10.1.1}$$

where $\pi : \mathcal{S} \to \mathfrak{D}(\mathcal{A})$ maps from states to action distributions.

Although previously policies did not need to be stochastic, here we are explicitly considering stochastic policies to facilitate the approximations. Nevertheless, frequently, we cannot actually perform this maximisation if the state or action space are very large. So we define ϕ, a distribution on $\mathcal{S}$, and parametrised sets of value functions $\mathcal{V}_\Theta$ and policies $\mathit{\Pi}_\Theta$.

Parameteric value function estimation

$$\mathcal{V}_\Theta = \{\boldsymbol{v}_\theta \mid \theta \in \Theta\}, \qquad \theta^* \in \arg\min_{\theta \in \Theta} \|\boldsymbol{v}_\theta - \boldsymbol{u}\|_\phi \tag{10.1.2}$$

where $\|\cdot\|_\phi \triangleq \int_{\mathcal{S}} |\cdot| \, \mathrm{d}\phi$.

In other words, we find the value function best matching the approximate value function $\boldsymbol{u}$. If $\boldsymbol{u} = V^*$ then we end up getting the best possible approximation with respect to the distribution ϕ.

Parameteric policy estimation

$$\mathit{\Pi}_\Theta = \{\pi_\theta \mid \theta \in \Theta\}, \qquad \theta^* \in \arg\min_{\theta \in \Theta} \|\pi_\theta - \pi^*_{\boldsymbol{u}}\|_\phi \tag{10.1.3}$$

where $\pi^*_{\boldsymbol{u}} = \arg\max_{\pi \in \mathit{\Pi}} \mathscr{L}_\pi \boldsymbol{u}$

Example 10.1.1. *A simple case is when ϕ does not support $\mathcal{S}$, that is it only takes positive values for some states $s \in \mathcal{S}$.*

10.1.1 Error bounds

If the approximate value function $\boldsymbol{u}$ is close to V^* then the greedy policy with respect to $\boldsymbol{u}$ is close to optimal. For a finite state and action space, the following holds.

Theorem 10.1.1. *Consider a finite MDP μ with discount factor $\gamma < 1$ and a vector $\boldsymbol{u} \in \mathcal{V}$ such that $\left\|\boldsymbol{u} - V_\mu^*\right\|_\infty = \epsilon$. If π is the $\boldsymbol{u}$-greedy policy then*

$$\left\|V_\mu^\pi - V_\mu^*\right\|_\infty \leq \frac{2\gamma\epsilon}{1-\gamma}.$$

In addition, $\exists \epsilon_0 > 0$ s.t. if $\epsilon < \epsilon_0$, then π is optimal.

Proof. Recall that $\mathscr{L}$ is the one-step Bellman operator and $\mathscr{L}_\pi$ is the one-step policy operator on the value function. Then

$$\begin{aligned}
\left\|V^\pi - V^*\right\|_\infty &= \left\|\mathscr{L}_\pi V^\pi - V^*\right\|_\infty \\
&\leq \left\|\mathscr{L}_\pi V^\pi - \mathscr{L}_\pi \boldsymbol{u}\right\|_\infty + \left\|\mathscr{L}_\pi \boldsymbol{u} - V^*\right\|_\infty \\
&\leq \gamma \left\|V^\pi - \boldsymbol{u}\right\|_\infty + \left\|\mathscr{L}\boldsymbol{u} - V^*\right\|_\infty \\
&\leq \gamma \left\|V^\pi - V^*\right\|_\infty + \gamma \left\|V^* - \boldsymbol{u}\right\|_\infty + \gamma \left\|\boldsymbol{u} - V^*\right\|_\infty \\
&\leq \gamma \left\|V^\pi - V^*\right\|_\infty + 2\gamma\epsilon.
\end{aligned}$$

This proves the first part.

For the second part, note that the state and action sets are finite. Consequently, the set of policies is finite. Thus, there is some $\epsilon_0 > 0$ such that the best sub-optimal policy is ϵ_0-close to the optimal policy in value. So, if $\epsilon < \epsilon_0$, the obtained policy must be optimal. □

10.1.2 Features

Frequently, when dealing with large, or complicated spaces, it pays to project the state and action observations onto a feature space $\mathcal{X}$. In that way, we can make problems much more manageable. Generally speaking, a feature mapping is defined as follows.

Feature mapping $f : \mathcal{S} \times \mathcal{A} \to \mathcal{X}$.
For $\mathcal{X} \subset \mathbb{R}^n$, the feature mapping can be written in vector form:

$$f(s,a) = \begin{bmatrix} f_1(s,a) \\ \dots \\ f_n(s,a) \end{bmatrix} \tag{10.1.4}$$

What sort of functions should we use? A common idea is to use a set of smooth functions, that are focused around a single point. One of the most usual examples are radial basis functions.

Example 10.1.2 (Radial Basis Functions)**.** *Let d be a metric on $\mathcal{S} \times \mathcal{A}$ and $\{(s_i, a_i) \mid i = 1, \dots, n\}$. Then we define each element of f as:*

$$f_i(s,a) \triangleq \exp\left\{-d[(s,a),(s_i,a_i)]\right\}. \tag{10.1.5}$$

These function are sometimes called kernels.

Another common type of functions are binary functions. These effectively discretise a continuous space through either a cover or a partition.

Definition 10.1.2. *The collection of sets $\mathscr{G}$ is a cover of X iff $\bigcup_{S\in\mathscr{G}} S \supset X$.*

Definition 10.1.3. *The collection of sets $\mathscr{G}$ is a partition of X iff*

1. *$\mathscr{G}$ is a cover of X*
2. *If $S \neq R \in \mathscr{G}$ then $S \cap R = \emptyset$.*
3. *$\bigcup_{S\in\mathscr{G}} S = X$.*

In reinforcement learning, these types of feature functions corresponding to partitions are usually referred to as tilings.

Example 10.1.3 (Tilings)**.** *Let $\mathscr{G} = \{X_1, \dots, X_n\}$ be a partition of $\mathcal{S} \times \mathcal{A}$ of size n. Then:*

$$f_i(s,a) \triangleq \mathbb{I}\{(s,a) \in X_i\}. \tag{10.1.6}$$

Multiple tilings create a cover. These can be used without many difficulties with most discrete reinforcement learning algorithms.

10.2 Approximate policy iteration

Approximate policy ieration

The main idea of approximate policy iteration is to replace the exact Bellman operator $\mathscr{L}$ with an approximate version $\hat{\mathscr{L}}$ and the exact value of the policy with an approximate version. In fact, in the policy improvement step, we simply try to get as close as possible to the best possible improvement, in a restricted set of policies, using an approximate operator. Similarly, in the policy evaluation step, we try to get as close as possible to the actual value of the improved policy.

Algorithm 20 Generic approximate policy iteration algorithm

input Initial value function $\boldsymbol{v}_0$, approximate Bellman operator $\hat{\mathscr{L}}$, approximate value estimator $\hat{V}$.
for $k = 1, \dots$ **do**
  $\pi_k = \arg\min_{\pi\in\hat{\Pi}} \left\| \hat{\mathscr{L}}_\pi \boldsymbol{v}_{k-1} - \mathscr{L}\boldsymbol{v}_{k-1} \right\|$ `// policy improvement`
  $\boldsymbol{v}_k = \arg\min_{\boldsymbol{v}\in\hat{\mathcal{V}}} \|\boldsymbol{v} - V_\mu^{\pi_k}\|$ `// policy evaluation`
end for

Theoretical gurantees

Assumption 10.2.1. *Consider a discounted problem with discount factor γ and iterates $\boldsymbol{v}_k, \pi_k$ such that:*

$$\|\boldsymbol{v}_k - V^{\pi_k}\|_\infty \leq \epsilon, \qquad \forall k \tag{10.2.1}$$

$$\left\|\mathscr{L}_{\pi_{k+1}}\boldsymbol{v}_k - \mathscr{L}\boldsymbol{v}_k\right\|_\infty \leq \delta, \qquad \forall k \tag{10.2.2}$$

Theorem 10.2.1 (Bertsekas and Tsitsiklis [1996], proposition 6.2)**.** *Under Assumption 10.2.1*

$$\limsup_{k\to\infty} \|V^{\pi_k} - V^*\|_\infty \leq \frac{\delta + 2\gamma\epsilon}{(1-\gamma)^2}. \tag{10.2.3}$$

10.2.1 Estimation building blocks

Lookahead policies

Given an approximate value function $\boldsymbol{u}$, the transition model of the MDP P_μ and expected rewards r_μ we can always find the improving policy given in Def. 10.1.1 via the following single-step lookahead.

Single-step lookahead

$$\pi_{\boldsymbol{q}}(a \mid i) > 0 \qquad \text{iff } a \in \arg\max_{a' \in \mathcal{A}} q(i, a') \tag{10.2.4}$$

$$q(i,a) \triangleq r_\mu(i,a) + \gamma \sum_{j \in \mathcal{S}} P_\mu(j \mid i, a)\boldsymbol{u}(j). \tag{10.2.5}$$

We are however not necessarily limited to the first-step. By looking T steps forward into the future we can improve both our value function and policy estimates.

T-step lookahead

$$\pi(i; \boldsymbol{q}_T) = \arg\max_{a \in \mathcal{A}} \boldsymbol{q}_T(i,a), \tag{10.2.6}$$

where $\boldsymbol{u}_k$ is recursively defined as:

$$q_k(i,a) = r_\mu(i,a) + \gamma \sum_{j \in \mathcal{S}} P_\mu(j \mid i, a)\boldsymbol{u}_{k-1}(j) \tag{10.2.7}$$

$$\boldsymbol{u}_k(i) = \max\{q_k(i,a) \mid a \in \mathcal{A}\} \tag{10.2.8}$$

and $\boldsymbol{u}_0 = \boldsymbol{u}$.

In fact, taking $\boldsymbol{u} = \boldsymbol{0}$, this recursion is identical to solving the k-horizon problem and at the limit we obtain solution to the original problem. In the general case, our value function estimation error is bounded by $\gamma^k \|\boldsymbol{u} - V^*\|$.

Rollout policies

As we have seen in Section 8.4.2 one way to obtain an the approximate value function of an arbitrary policy π is to use Monte Carlo estimation That is, to simulate K sequences of state-action-reward tuples by running the policy on the MPD. More specifically, we have the following rollout estimate.

Rollout estimate of the q-factor

$$q(i,a) = \frac{1}{K_i} \sum_{k=1}^{K_i} \sum_{t=0}^{T_k - 1} r(s_{t,k}, a_{t,k}),$$

where $s_{t,k}, a_{t,k} \sim \mathbb{P}^{\pi}_{\mu}(\cdot \mid s_0 = i, a_0 = a)$, and $T_k \sim \mathit{Geom}(1-\gamma)$.

This results in a set of samples of $\boldsymbol{q}$-factors. We now find a parametric policy that approximates the optimal policy with respect to our samples, $\pi^*_{\boldsymbol{q}}$. For a finite number of actions, this fitting can be seen as a classification problem. Once more, we define a distirbution ϕ on the states, over which we wish to perform the minimisation.

Rollout policy estimation.
Given a set of samples $q(i,a)$ for $i \in \hat{S}$, we estimate

$$\min_{\boldsymbol{\theta}} \left\| \pi_{\boldsymbol{\theta}} - \pi^*_{\boldsymbol{q}} \right\|_{\phi},$$

for some ϕ on $\hat{S}$.

10.2.2 The value estimation step

We can now attempt to fit a parametric approximation to a given value function $\boldsymbol{v}$ or $\boldsymbol{q}$. The simplest way to do so is via a generalised linear model. A natural parametrisation for the value function is to use a generalised linear model on a set of features. Then the value function is a linear function of the features with parameters $\boldsymbol{\theta}$. More precisely, we can define the following model.

Generalised linear model using features (or kernel)
Feature mapping $f : \mathcal{S} \to \mathbb{R}^n$, parameters $\boldsymbol{\theta} \in \mathbb{R}^n$.

$$\boldsymbol{v}_{\boldsymbol{\theta}}(s) = \sum_{i=1}^{n} \theta_i f_i(s) \tag{10.2.9}$$

In order to fit a value function, we first pick a set of *representative states* $\hat{S}$ to fit our value function $\boldsymbol{v}_{\boldsymbol{\theta}}$ to $\boldsymbol{v}$. We can then estimate the optimal parameters via gradient descent.

Fitting a value function.

$$c(\boldsymbol{\theta}) = \sum_{s \in \hat{S}} c_s(\boldsymbol{\theta}), \qquad c_s(\boldsymbol{\theta}) = \phi(s) \left\| \boldsymbol{v}_{\boldsymbol{\theta}}(s) - \boldsymbol{v}(s) \right\|_p^{\kappa}. \tag{10.2.10}$$

Example 10.2.1. *The case $p = 2$, $\kappa = 2$ In this case the square root and κ cancel out and we obtain*

$$\nabla_{\boldsymbol{\theta}} c_s = \phi(s) \sum_{a \in \mathcal{A}} \nabla_{\boldsymbol{\theta}} [\boldsymbol{v}_{\boldsymbol{\theta}}(s) - \boldsymbol{v}(s)]^2 = 2[\boldsymbol{v}_{\boldsymbol{\theta}}(s) - \boldsymbol{v}(s)] \nabla_{\boldsymbol{\theta}} \boldsymbol{v}_{\boldsymbol{\theta}},$$

where $\nabla_{\boldsymbol{\theta}} \boldsymbol{v}_{\boldsymbol{\theta}}(s) = f(s)$. *Taking partial derivatives* $\partial/\partial\theta_j$, *leads to the update rule:*

$$\theta_j' = \theta_j - 2\alpha\phi(s)[\boldsymbol{v}_{\boldsymbol{\theta}}(s) - \boldsymbol{v}(s)]f_j(s). \tag{10.2.11}$$

10.2.3 Policy estimation

A natural parametrisation for the policy is to use a generalised linear model on a set of features. Then the policy can be described (up to scaling) as a linear function of the features with parameters $\boldsymbol{\theta}$. More precisely, we can define the following model.

Generalised linear model using features (or kernel).
Feature mapping $f : \mathcal{S} \to \mathbb{R}^n$, parameters $\boldsymbol{\theta} \in \mathbb{R}^n$.

$$\pi_{\boldsymbol{\theta}}(a \mid s) = \frac{g(s,a)}{h(s)}, \qquad g(s,a) = \sum_{i=1}^{n} \theta_i f_i(s,a), \quad h(s) = \sum_{b \in \mathcal{A}} g(s,b) \tag{10.2.12}$$

We are performing the intermediate step of estimating g first, because we need to make sure that the policy is a distribution over actions. An alternative method would be to directly constrain the policy parameters so the result is always a distribution, but that would require a more complex optimisation method.

In order to fit a policy, we first pick a set of representative states $\hat{S}$ and then we find a $\pi_{\boldsymbol{\theta}}$ that approximates π. In order to do so, we can define an appropriate cost function and then estimate the optimal parameters via some arbitrary optimisation method.

Fitting a policy through a cost function.

$$c(\boldsymbol{\theta}) = \sum_{s \in \hat{S}} c_s(\boldsymbol{\theta}), \qquad c_s(\boldsymbol{\theta}) = \phi(s) \left\| \pi_{\boldsymbol{\theta}}(\cdot \mid s) - \pi(\cdot \mid s) \right\|_p^{\kappa}. \tag{10.2.13}$$

The function $\phi : \mathcal{S} \to \mathbb{R}_+$ is a weighting on the state space, such that we put more weight in more "important" states. Choosing the weights and the set of representative states $\hat{S}$ is an interesting problem. A good choice is to relate those to the state distribution under different policies. One method to minimise the cost function is to use gradient descent. The gradient of this cost function is

$$\nabla_{\boldsymbol{\theta}} c = \sum_{s \in \hat{S}} \nabla_{\boldsymbol{\theta}} c_s.$$

We obtain different results for different norms. Mainly three cases are of interest: $p = 1, p = 2, p \to \infty$. Here, we only consider the first one.

The case $p = 1$, $\kappa = 1$.
The derivative can be written as:

$$\nabla_{\boldsymbol{\theta}} c_s = \phi(s) \sum_{a \in \mathcal{A}} \nabla_{\boldsymbol{\theta}} |\pi_{\boldsymbol{\theta}}(a \mid s) - \pi(a \mid s)|,$$
$$\nabla_{\boldsymbol{\theta}} |\pi_{\boldsymbol{\theta}}(a \mid s) - \pi(a \mid s)| = \nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}}(a \mid s) \operatorname{sgn}[\pi_{\boldsymbol{\theta}}(a \mid s) - \pi(a \mid s)]$$

The policy derivative in turn is

$$\pi_{\boldsymbol{\theta}}(a \mid s) = \frac{h(s) \nabla_{\boldsymbol{\theta}} g(s,a) - \nabla_{\boldsymbol{\theta}} h(s) g(s,a)}{h(s)^2},$$

with $\nabla_{\boldsymbol{\theta}} h(s) = \left(\sum_{b \in \mathcal{A}} f_i(s,b)\right)_i$ and $\nabla_{\boldsymbol{\theta}} g(s,a) = f(s,a)$. Taking partial derivatives $\partial / \partial \theta_j$, leads to the update rule:

$$\theta_j' = \theta_j - \alpha \phi(s) \left(\pi_{\boldsymbol{\theta}}(a \mid s) \sum_{b \in \mathcal{A}} f_j(s,b) - f_j(s,a) \right). \tag{10.2.14}$$

Iterating over (s,a) pairs with a decreasing step-size α according to the stochastic approximation assumptions, should ensure convergence.

Exercise 6. *Find the derivative for the two other cases, specifically:*

1. $p = 2$, $\kappa = 2$.

2. $p \to \infty$, $\kappa = 1$.

Alternative cost functions. It is frequently a good idea to add a *penalty term* to the cost function. The purpose of this is to prevent overfitting of the parameters to a small number of observations. Frequently, this is done by constraining the parameters to be small, via a penalty term of the form $\|\boldsymbol{\theta}\|^q$.

10.2.4 Rollout-based policy iteration methods

One idea for estimating the value function is to simply perform rollouts, while the policy itself is estimated in parametric form, as suggested in Bertsekas and Tsitsiklis [1996]. The first practical algorithm in this direction was Rollout Sampling Approximate Policy iteration Dimitrakakis and Lagoudakis [2008]. The main idea is to concentrate rollouts in interesting parts of the state space.

Algorithm 21 Rollout Sampling Approximate Policy Iteration.

for $k = 1, \dots$ **do**
  Select a set of representative states $\hat{S}_k$
  for $n = 1, \dots$ **do**
    Select a state $s_n \in \hat{S}_k$ maximising $U_n(s)$ and perform a rollout.
    If $\hat{a}^*(s_n)$ is optimal w.p. $1 - \delta$, put s_n in $\hat{S}_k(\delta)$ and remove it from $\hat{S}_k$.
  end for
  Calculate $\boldsymbol{q}_k \approx Q^{\pi_k}$ from the rollouts.
  Train a classifier $\pi_{\boldsymbol{\theta}_{k+1}}$ on the set of states $\hat{S}_k(\delta)$ with actions $\hat{a}^*(s)$.
end for

The main idea is to concentrate rollouts on promising states. We can use the empirical state distribution to select starting states. We always choose the state s with the highest upper bound $U_n(s)$. More specifically, we employ a Hoeffding bound to select the state with the largest gap between actions. We stop rolling out states where we are certain to have found the best action. This is done by applying the Hoeffding bound to gaps between actions.

10.2.5 Least Squares Methods

The main idea is to formulate the problem in linear form, using a feature space.

Least square value estimation

Recall that the solution of

$$\boldsymbol{v} = \boldsymbol{r} + \gamma \boldsymbol{P}_{\mu,\pi} \boldsymbol{v} \tag{10.2.15}$$

is the value function of π and can be obtained via

$$\boldsymbol{v} = (\boldsymbol{I} - \gamma \boldsymbol{P}_{\mu,\pi})^{-1} \boldsymbol{r}. \tag{10.2.16}$$

However, in this setting, we do not have access to the transition matrix. In addition, when the state space is continuous (e.g. $\mathcal{S} \subset \mathbb{R}^n$), the transition matrix becomes a general transition kernel. In addition, while up to now the set of value functions $\mathcal{V}$ was a Euclidean subset, now $\mathcal{V}$ becomes a Hilbert space.

In general, we deal with this case via projections. We project down from teh infinite-dimensional Hilbert space to one with finite-dimensions. We assume that there is a projection that is complex enough for us to be able to recover the original value function.

Projection.
Setting $\boldsymbol{v} = \boldsymbol{\Phi}\boldsymbol{\theta}$ where $\boldsymbol{\Phi}$ is a feature matrix and $\boldsymbol{\theta}$ is a parameter vector we have

$$\boldsymbol{\Phi}\boldsymbol{\theta} = \boldsymbol{r} + \gamma \boldsymbol{P}_{\mu,\pi} \boldsymbol{\Phi}\boldsymbol{\theta} \tag{10.2.17}$$

$$\boldsymbol{\theta} = \left[(\boldsymbol{I} - \gamma \boldsymbol{P}_{\mu,\pi}) \boldsymbol{\Phi} \right]^{-1} \boldsymbol{r} \tag{10.2.18}$$

Replacing the inverse with the *pseudo-inverse*, with $\boldsymbol{A} = (\boldsymbol{I} - \gamma \boldsymbol{P}_{\mu,\pi})\boldsymbol{\Phi}$

$$\tilde{\boldsymbol{A}}^{-1} \triangleq \boldsymbol{A}^\top \left(\boldsymbol{A}\boldsymbol{A}^\top\right)^{-1},$$

gives us an estimate for the parameters. If the inverse exists, then it is equal to the pseudoinverse. The main idea that makes this work is to calculate everything on the empirical transition matrix, the empirical rewards and the empirical feature vectors.

Empirical constructions.
Given a set of data points $\{(s_i, a_i, r_i, s_i') \mid i = 1, \ldots, n\}$, which may not be consecutive, we define:

1. $\boldsymbol{r} = (r_i)_i$.
2. $\boldsymbol{\Phi}_i = f(s_i, a_i)$, $\boldsymbol{\Phi} = (\boldsymbol{\Phi}_i)_i$.
3. $\boldsymbol{P}_{\mu,\pi} = \boldsymbol{P}_\mu \boldsymbol{P}_\pi$, $\boldsymbol{P}_{\mu,\pi}(i,j) = \mathbb{I}\{j = i+1\}$

We are now ready to define some algorithms. We begin with an algorithm that estimates an approximate value function for some policy π given some data D and a feature mapping f.

Algorithm 22 LSTDQ - Least Squares Temporal Differences on $\boldsymbol{q}$-factors

input data $D = \{(s_i, a_i, r_i, s_i') \mid i = 1, \ldots, n\}$, feature mapping f, policy π
$\boldsymbol{\theta} = (\boldsymbol{\Phi}(\widetilde{\boldsymbol{I} - \gamma \boldsymbol{P}_{\mu,\pi}}))^{-1} \boldsymbol{r}$

This algorithm is sufficient for performing approximate policy iteration by plugging it into the generic API algorithm to estimate a value function. Since LSTDQ returns $\boldsymbol{q}$-factors, our next policy can simply be greedy with respect to the value estimatees.

Algorithm 23 LSPI - Least Squares Policy Iteration

input data $D = \{(s_i, a_i, r_i, s_i') \mid i = 1, \ldots, n\}$, feature mapping f
Set π_0 arbitrarily.
for $k = 1, \ldots$ **do**
  $\boldsymbol{\theta}_k = LSTDQ(D, f, \pi_{k-1})$.
  $\pi_k = \pi^*_{\boldsymbol{\Phi}\boldsymbol{\theta}_k}$.
end for

10.3 Approximate Value Iteration

Approximate algorithms can also be defined for backwards induction. The general algorithmic structure remains the same. We only need to replace the exact steps with approximations. Usually this is necessary when the value function cannot be updated everywhere exactly, possibly because our value function representations are not complex enough to capture the true value function.

10.3.1 Approximate backwards induction

The first algorithm is approximate backwards induction. Let us start with the basic backwards induction algorithm:

$$V_t^*(s) = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \, \mathbb{E}_\mu \left(V_{t+1}^* \mid s_t = s, a_t = a \right) \right\} \tag{10.3.1}$$

This essentially the same both for finite and infinite-horizon problems. Now assume that the set of functions $\mathcal{V}$ that you can use to approximate the value functions is not rich enough, so none of its members will correspond to the left side of (10.3.1). Consider then the following value function approximation.

Let our estimate at time t be $\boldsymbol{v}_t \in \mathcal{V}$, with $\mathcal{V}$ being a set of parametrised functions. Let $\hat{V}_t$ be our one-step update given the value function approximation at the next step, $\boldsymbol{v}_{t+1}$. Then $\boldsymbol{v}_t$ will be the closest approximation in that set.

Iterative approximation

$$\hat{V}_t(s) = \max_{a \in \mathcal{A}} \left\{ r(s,a) + \gamma \sum_{s'} P_\mu(s' \mid s, a) \boldsymbol{v}_{t+1}(s') \right\} \tag{10.3.2}$$

$$\boldsymbol{v}_t = \arg\min \left\{ \left\| \boldsymbol{v} - \hat{V}_t \right\| \;\middle|\; \boldsymbol{v} \in \mathcal{V} \right\} \tag{10.3.3}$$

Any algorithm can be used to perform the above minimisation, including gradient descent. Now consider the case where $\boldsymbol{v}$ is a parametrised function with parameters $\boldsymbol{\theta}$. Then it is sufficient for us to maintain the parameter $\boldsymbol{\theta}_t$ at time t. These can be updated with a gradient scheme at every step. In the online case, our next-step estimates can be given by gradient descent:

Online gradient estimation

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \alpha_t \nabla_{\boldsymbol{\theta}} \left\| \boldsymbol{v}_t - \hat{V}_t \right\| \tag{10.3.4}$$

This gradient descent algorithm can also be made stochastic, if we sample from the probability distribution given in the iterative approximation. The enxt sections give some examples.

10.3.2 State aggregation

Partitions, or tiling of the state space, inevitably lead to what is called *state aggregation*. That is, multiple different states are seen as identical by the algorithm. Unfortunately, it is very rarely the case that aggregated states really are identical. Nevertheless, as we can see in the example below, aggregation significantly simplifies the estimation problems.

Aggregated estimate.
Let $\mathscr{G} = \{S_0, S_1, \dots, S_n\}$ be a partition of $\mathcal{S}$, with $S_0 = \emptyset$ and $\boldsymbol{\theta} \in \mathbb{R}^n$ and let $f_k(s_t) = \mathbb{I}\{s_t \in S_k\}$. Then the approximate value function is

$$\boldsymbol{v}(s) = \boldsymbol{\theta}(k), \qquad \text{if } s \in S_k,\ k \neq 0. \tag{10.3.5}$$

That is, the value of every state corresponds to the value of the k-th set in the partition. Of course, this is only a very rough approximation if the sets S_k are very large. However, this is a very nice approach to use for gradient descent updates, as only one parameter needs to be updated at every step.

Online gradient estimate.
Consider the case $\|\cdot\| = \|\cdot\|_2^2$. For $s_t \in S_k$:

$$\boldsymbol{\theta}_{t+1}(k) = (1-\alpha)\boldsymbol{\theta}_t(k) + \alpha \max_{a \in \mathcal{A}} r(s_t, a) + \gamma \sum_j P(j \mid s_t, a)\boldsymbol{v}_t(s) \tag{10.3.6}$$

For $s_t \notin S_k$:

$$\boldsymbol{\theta}_{t+1}(k) = \boldsymbol{\theta}(k). \tag{10.3.7}$$

Of course, whenever we perform the estimation online, we are limited to estimation on the sequence of states s_t that we visit. Consequently, estimation on other states may not be very good. It is indeed possible that we will suffer from oscillation problems.

10.3.3 Representative states

A rather different idea is to choose only some representative states on which to perform the approximation. The main assumption is that the value of all other states can be represented as a convex combination of the value of the representative states.

Representative states approximation.
Let $\hat{S}$ be a set of n representative states and $\boldsymbol{\theta} \in \mathbb{R}^n$ and a feature mapping f:

$$\sum_{i=1}^{n} f_i(s) = 1, \qquad \forall s \in \mathcal{S}.$$

The feature mapping is used to perform the convex combination. For any given state s, it has higher value for representative states i which are "closer" to it. In general, the feature mapping is fixed, and we just want to find a set of parameters for the values of the representative states.

We focus here on the online estimate. At time t, for each representative state i, we obtain a new estimate of its value function and plug it back in.

Representative state update.
For $i \in \hat{S}$:

$$\boldsymbol{\theta}_{t+1}(i) = \max_{a \in \mathcal{A}} \left\{ r(i,a) + \gamma \int \boldsymbol{v}_t(s) \, \mathrm{d}P(s \mid i, a) \right\} \tag{10.3.8}$$

with

$$\boldsymbol{v}_t(s) = \sum_{i=1}^{n} f_i(s) \boldsymbol{\theta}_t(i). \tag{10.3.9}$$

When the summation is not possible, we may instead approximate the expectation with a Monte-Carlo method. One particular problem with this method arises when the transition kernel is very sparse. Then we are basing our estimates on approximate values of other states, which may be very far from any other representative state.

Bellman error methods

The problems with the representative state update can be alleviated through Bellman error minimisation. The idea here is to obtain as a *consistent* value function as possible. The basic Bellman error minimisation is as follows:

$$\min_{\boldsymbol{\theta}} \|\boldsymbol{v}_{\boldsymbol{\theta}} - \mathscr{L} \boldsymbol{v}_{\boldsymbol{\theta}}\| \tag{10.3.10}$$

This is different from the approximate backwards induction algorithm we saw previously, since the same parameter $\boldsymbol{\theta}$ appears in both sides of the equality. Furthermore, if the norm has support in all of the state space and the approximate value function space contains the actual set of value functions then the minimum is 0 and we obtain the optimal value function.

Gradient update.
When the norm is

$$\|\boldsymbol{v}_{\boldsymbol{\theta}} - \mathscr{L} \boldsymbol{v}_{\boldsymbol{\theta}}\| = \sum_{s \in \hat{S}} D_{\boldsymbol{\theta}}(s)^2, \quad D_{\boldsymbol{\theta}}(s) = \boldsymbol{v}_{\boldsymbol{\theta}}(s) - \max_{a \in \mathcal{A}} \int_{\mathcal{S}} \boldsymbol{v}_{\boldsymbol{\theta}}(j) \, \mathrm{d}P(j \mid s, a). \tag{10.3.11}$$

then the gradient update becomes

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \alpha D_{\boldsymbol{\theta}_t}(s_t) \nabla_{\boldsymbol{\theta}} D_{\boldsymbol{\theta}_t}(s_t) \tag{10.3.12}$$

$$\nabla_{\boldsymbol{\theta}} D_{\boldsymbol{\theta}_t}(s_t) = \nabla_{\boldsymbol{\theta}} \boldsymbol{v}_{\boldsymbol{\theta}_t}(s_t) - \int_{\mathcal{S}} \nabla_{\boldsymbol{\theta}} \boldsymbol{v}_{\boldsymbol{\theta}_t}(j) \, \mathrm{d}P(j \mid s_t, a_t^*) \tag{10.3.13}$$

$$a_t^* = \arg\max_{a \in \mathcal{A}} \left\{ r(s_t, a) + \gamma \int_{\mathcal{S}} \boldsymbol{v}_{\boldsymbol{\theta}}(j) \, \mathrm{d}P(j \mid s_t, a) \right\} \tag{10.3.14}$$

We can also construct a Q-factor approximation for the case where no model is available. This is going to be simply done by replacing P with the empirical transition observed at time t.

A litany of approximation algorithms

- Fitted Q-iteration Antos et al. [2008b].
- Fitted value iteration Munos and Szepesvári [2008].
- Rollout sampling policy iteration Dimitrakakis and Lagoudakis [2008]
- State aggregation Singh et al. [1995], Bernstein [2007]
- Bellman error minimisation Antos et al. [2008a], Dimitrakakis [2013], Ghavamzadeh and Engel [2006]
- Least-squares methods Bradtke and Barto [1996], Boyan [2002], Lagoudakis and Parr [2003].

Chapter 11

Bayesian reinforcement learning

11.1 Introduction

Bayesian reinforcement learning connects all elements previously seen in the course. Firstly, how to express uncertainty and preferences via probabilities and utilities. Secondly, how to make decisions under uncertainty, including how to make decisions maximising the utility in different conditions. Thirdly, how to adjust our subjective belief in the face of new evidence. Fourthly, optimal experiment design: how to make decisions in problems where our decisions can affect the evidence we obtain. These problems can be modelled as Markov decision processes. We also consider the problem of finding optimal policies for Markov decision processes.

In the previous two chapters, we have considered stochastic algorithms for acting within Markov decision processes. These stochastic analogues of deterministic MDP algorithms can also be used in the context of "learning" the optimal policy while acting in the MDP itself, even if the MDP parameters are not known. In the case where the MDP is very large, or the state/action spaces are continuous, it is necessary to approximate it. These can be used in conjunction with stochastic approximations.

Now, however, we come full circle to the setting of subjective probability and utility. We shall try and solve the reinforcement learning problem directly. Here, we are acting in an MDP which is not known, but we have a subjective belief about what the MDP is.

11.2 Bayesian reinforcement learning

The reinforcement learning problem can be formulated as the problem of learning to act in an unknown environment, only by interaction and reinforcement. All of those elements of the definition are important. Firstly and foremostly it is a *learning* problem. Consequently, we have only partial prior knowledge about the environment we are acting in. This knowledge is arrived at via *interaction* with the environment. We do not have a fixed set of data to work with, but we must actively explore the environment to understand how it works. Finally, there is an intrinsic *reinforcement* that punishes some behaviours and rewards others. We can formulate some of these problems as Markov decision processes.

Markov decision processes (MDP) as an environment.
We are in some *environment* μ, where at each time, we: step t:

- Observe *state* $s_t \in \mathcal{S}$.
- Take *action* $a_t \in \mathcal{A}$.
- Receive *reward* $r_t \in \mathbb{R}$.

In these types of problems, the environment state and our action fully determines the distribution of the immediate reward, as well as that of the next state, as described in Definition 8.3.1. When μ is unknown, the probability of the immediate reward is given by $P_\mu(r_t \mid s_t, a_t)$ and that of next state by $P_\mu(s_{t+1} \mid s_t, a_t)$.

However, we now assume that we do not known μ. The structure of the unknown MDP process is shown in Figure 11.1 below,

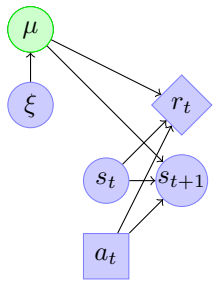


Figure 11.1: The unknown Markov decision process

The optimal policy for a given μ

When μ is known, we wish to find a *policy* $\pi : \mathcal{S} \to \mathcal{A}$ maximising the *utility* in expectation. This requires us to solve the maximisation problem $\max_\pi \mathbb{E}^\pi_\mu U$, where the utility is an additive function of rewards, $U = \sum_{t=1}^T r_t$. When μ is *known*, we can use standard algorithms, such as value or policy iteration. However, knowing μ is contrary to the problem definition.

In Chapter 9 we have seen a number of stochastic approximation algorithms which allow us to learn the optimal policy for a given MDP eventually. However, these generally give few guarantees on the performance of the policy while learning. How can we create an algorithm for optimal learning MDPs? This should trades off exploring the environment to obtain further knowledge, and simultaneously exploiting its knowledge.

The solution is rather simple, conceptually. Within the subjective probabilistic framework, we only need to define a prior belief ξ on the set of MDPs $\mathcal{M}$, and then find the policy that maximises the expected utility with respect to the prior. The value of information is automatically taken into account in this model.

This should not be too surprising, as we have previously seen it in two Bayes-optimal construction. The first was the simple optimal stopping procedure in Section 7.4, which introduced the backwards induction algorithm. The second was the optimal experiment design problem, which resulted in the bandit Markov decision process of Section 8.2. Let us now formulate the reinforcement learning problem as a Bayesian maximisation problem.

Let ξ be a prior over $\mathcal{M}$ and $\mathit{\Pi}$ be a set of policies. Then the expected utility of the optimal policy is:

$$U^*_\xi \triangleq \max_{\pi \in \mathit{\Pi}} \mathbb{E}(U \mid \pi, \xi) = \max_{\pi \in \mathit{\Pi}} \int_{\mathcal{M}} \mathbb{E}(U \mid \pi, \mu) \, \mathrm{d}\xi(\mu) \tag{11.2.1}$$

Finding the optimal policy is not easy as in general the optimal policy π must now map from *complete histories* to actions: *Planning* must take into account *future learning*.

Policy types
We use Π to denote the set of all policies. We use Π_k to denote the set of k-order Markov policies. Important special cases are the set of *blind* policies Π_0 and the set of *memoryless* policies Π_1. A policy in $\pi \in \bar{\Pi}_k \subset \Pi_k$ is *stationary*, when $\pi(A \mid s_{t-k+1}^t, a_{t-k+1}^{t-1}) = \pi(A \mid s^k, a^{k-1})$ for all t.

Generally speaking, the Bayes-optimal policies are history-dependent, as shown by the following counterexample.

Example 11.2.1. *Consider two MDPs, μ_1, μ_2 with states $\mathcal{S} = \{1\}$ and actions $\mathcal{A} = \{1, 2\}$. In the i-th MDP, whenever you take action $a_t = i$, you obtain reward $r_t = 1$, otherwise you obtain reward 0.The expected utility of a memoryless policy taking action i with probability $\pi(i)$ would be*

$$\mathbb{E}_\xi^\pi U = T \sum_i \xi(\mu_i)\pi(i),$$

for horizon T. Consequently, if your prior is not uniform, you select the action corresponding to the MDP with the highest prior probability. Then, the maximal expected utility is:

$$\max_{\pi \in \Pi_1} \mathbb{E}_\xi^\pi U = T \max_i \xi(\mu_i).$$

In this case, we are certain which one is the right MDP as soon as we take one action. We can then follow the policy which selects the apparently best action at first, and then switches to the best action for the MDP we have seen. Then, our utility is simply $\max_i \xi(\mu_i) + (T - 1)$.

Given the above general remarks, let us now discuss how the optimal policies can be constructed. Firstly, we must examine how to update the belief. Given that, we shall examine methods for obtaining near-optimal policies.

11.2.1 Updating the belief

Strictly speaking, in order to update our belief, we must condition the prior distribution on all the information. This includes the sequence of observations up to this at point in time, including the states s^t, actions a^{t-1}, and rewards r^{t-1}, as well the policy π that we followed. Let $D_t = \left\langle s^t, a^{t-1}, r^{t-1} \right\rangle$ be the observed data to time t. Then

$$\xi(B \mid D_t, \pi) = \frac{\int_B \mathbb{P}_\mu^\pi(D_t) \, \mathrm{d}\xi(\mu)}{\int_{\mathcal{M}} \mathbb{P}_\mu^\pi(D_t) \, \mathrm{d}\xi(\mu)}. \tag{11.2.2}$$

However, as we shall see in the following remark, we can usually ignore the policy in the calculations.

Remark 11.2.1. *The dependence on the policy can be removed, since the posterior is the same for all policies that put non-zero mass on the observed data: Let $D_t \sim \mathbb{P}_\mu^\pi$. Then it is easy to see that $\forall \pi' \neq \pi$ such that $\mathbb{P}_\mu^{\pi'}(D_t) > 0$,*

$$\xi(B \mid D_t, \pi) = \xi(B \mid D_t, \pi').$$

Finally, since we are dealing with MDPs, the posterior calculation is easy to perform incrementally. This also more clearly demonstrates why there is no dependence on the policy. Let ξ_t be the (random) posterior at time t. Then, the next-step belief is going to be:

$$\xi_{t+1}(B) \triangleq \xi(B \mid D_{t+1}) = \frac{\int_B \mathbb{P}_\mu^\pi(D_t)\,\mathrm{d}\xi(\mu)}{\int_{\mathcal{M}} \mathbb{P}_\mu^\pi(D_t)\,\mathrm{d}\xi(\mu)} \tag{11.2.3}$$

$$= \frac{\int_B \mathbb{P}_\mu(s_{t+1}, r_t \mid s_t, a_t)\pi(a_t \mid s^t, a^{t-1}, r^{t-1})\,\mathrm{d}\xi(\mu \mid D_t)}{\int_{\mathcal{M}} \mathbb{P}_\mu(s_{t+1}, r_t \mid s_t, a_t)\pi(a_t \mid s^t, a^{t-1}, r^{t-1})\,\mathrm{d}\xi(\mu \mid D_t)} \tag{11.2.4}$$

$$= \frac{\int_B \mathbb{P}_\mu(s_{t+1}, r_t \mid s_t, a_t)\,\mathrm{d}\xi_t(\mu)}{\int_{\mathcal{M}} \mathbb{P}_\mu(s_{t+1}, r_t \mid s_t, a_t)\,\mathrm{d}\xi_t(\mu)} \tag{11.2.5}$$

The above calculation is easy to perform for arbitrarily complex MDPs when the set $\mathcal{M}$ is finite. The posterior calculation is also simple under certain conjugate priors.

Exercise 7. *A practical case is when we have an independent belief over the transition probabilities of each state-action pair. Consider the case where we have n states and k actions. Similar to the product-prior in the bandit case in Section 8.2, we assign a probability (density) $\xi_{s,a}$ to the probability vector $\boldsymbol{\theta}_{(s,a)} \in \mathbb{\Delta}^n$. We can then define our joint belief on the $(nk) \times n$ matrix $\boldsymbol{\Theta}$ to be*

$$\xi(\boldsymbol{\Theta}) = \prod_{s \in \mathcal{S}, a \in \mathcal{A}} \xi_{s,a}(\boldsymbol{\theta}_{(s,a)}).$$

(i) Derive the updates for a product-Dirichlet prior on transitions.

(ii) Derive the updates for and a product-Normal-Gamma prior on rewards.

(iii) What would be the meaning of using a Normal-Wishart prior on rewards?

11.3 Finding Bayes-optimal policies

The problem of policy optimisation in the Bayesian case is much harder than in the known-MDP case. This is simply because of the history dependence, which has two effects. Firstly, it makes the policy space much larger, as we need to consider history dependent policies. However, even we consider only memoryless policies, it does not make dynamic programming easier.

In this section, we first consider two simple heuristics for finding optimal policies. Then we examine policies which try and construct upper and lower bounds on the expected utility. Finally, we consider finite-lookahead backwards induction, that uses the same upper and lower bounds to perform efficient tree search.

11.3.1 The expected MDP heuristic

One simple heuristic is to simply calculate the expected MDP for a given belief ξ:

$$\bar{\mu}_\xi \triangleq \mathbb{E}_\xi\, \mu.$$

Then, we simply calculate the optimal memoryless policy for $\bar{\mu}_\xi$:

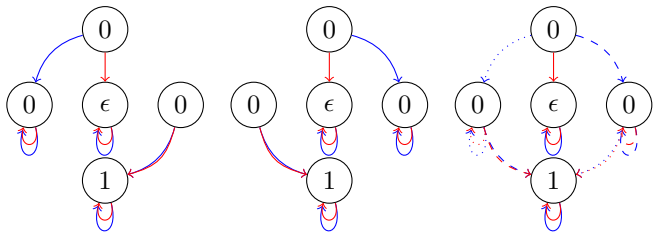
$$\pi^*(\bar{\mu}_\xi) \in \arg\max_{\pi \in \Pi_1} V^\pi_{\bar{\mu}_\xi},$$

where $\Pi_1 = \left\{\pi \in \Pi \;\middle|\; \mathbb{P}_\pi(a_t \mid s^t, a^{t-1}) = \mathbb{P}_\pi(a_t \mid s_t)\right\}$. Finally, we execute $\pi^*(\bar{\mu}_\xi)$ on the real MDP. The algorithm can be written as follows. Unfortunately, this

Algorithm 24 The expected MDP heuristic

for $k = 1, \ldots$ **do**
  $\mu_k \triangleq \mathbb{E}_{\xi_{t_k}} \mu$.
  $\pi_k \approx \arg\max_\pi \mathbb{E}^\pi_{\mu_k} U$.
  for $t = 1 + T_{k-1}, \ldots, T_k$ **do**
    Observe s_t.
    Update belief $\xi_t(\cdot) = \xi_{t-1}(\cdot \mid s_t, a_{t-1}, r_{t-1}, s_{t-1})$.
    Take action $a_t \sim \pi_k(a_t \mid s_t)$.
    Observe reward r_t.
  end for
end for

approach may be far from the optimal policy in Π_1, as shown by the following counterexample.



(a) μ_1 (b) μ_2 (c) $\bar{\mu}_\xi$

Figure 11.2: The two MDPs and the expected MDP from example **??**

.

Example 11.3.1 (Counterexample[1])**.** *In this example, illustrated in Figure 11.2, $\mathcal{M} = \{\mu_1, \mu_2\}$ is the set of MDPs, and the belief is $\xi(\mu_1) = \theta$, $\xi(\mu_2) = 1 - \theta$. All transitions are deterministic, and there are two actions, the blue and the red action. When we calculate the expected MDP, we see that now the state with reward 1 is reachable.*

Consequently, when $T \to \infty$, the $\bar{\mu}_\xi$-optimal policy is not optimal in Π_1 if:

$$\epsilon < \frac{\gamma\theta(1-\theta)}{1-\gamma}\left(\frac{1}{1-\gamma\theta} + \frac{1}{1-\gamma(1-\theta)}\right)$$

In this example, $\bar{\mu}_\xi \notin \mathcal{M}$.

[1] Based on one by Remi Munos

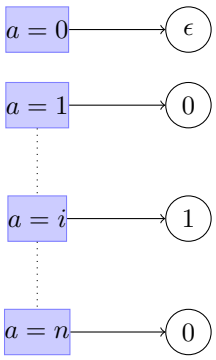


Figure 11.3: The MDP μ_i from example 11.3.2

11.3.2 The maximum MDP heuristic

An alternative idea is to simply pick the maximum-probability MDP, as shown in Algorithm 25. This at least guarantees that the MDP that you are acting optimally for is actually within the set of MDPs. However, it may still be the case that the resulting policy is sub-optimal, as shown by the following counterexample.

Algorithm 25 The maximum MDP heuristic

for $k = 1, \dots$ **do**
  $\mu_k \triangleq \arg\max_\mu \xi_{t_k}(\mu)$.
  $\pi_k \approx \arg\max_\pi \mathbb{E}^\pi_{\mu_k} U$.
  for $t = 1 + T_{k-1}, \dots, T_k$ **do**
    Observe s_t.
    Update belief $\xi_t(\cdot) = \xi_{t-1}(\cdot \mid s_t, a_{t-1}, r_{t-1}, s_{t-1})$.
    Take action $a_t \sim \pi_k(a_t \mid s_t)$.
    Observe reward r_t.
  end for
end for

Example 11.3.2 (Counterexample for $\hat{\mu}^*_\xi \triangleq \arg\max_\mu \xi(\mu)$)**.** *Let the MDP set be* $\mathcal{M} = \{\mu_i \mid i = 1, \dots, n\}$ *with* $\mathcal{A} = \{0, \dots, n\}$*. In all MDPs,* a_0 *gives a reward of* ϵ *and the MDP terminates. In the* i*-th MDP, all other actions give you a reward of* 0 *apart from the* i*-th action which gives you a reward of* 1*. Then the MDP terminates. The MDP is visualised in Figure 11.3.*

For this problem, the ξ*-optimal policy takes action* i *iff* $\xi(\mu_i) \geq \epsilon$*, otherwise takes action* 0*. On the other hand, the* $\hat{\mu}^*_\xi$*-optimal policy takes* $a = \arg\max_i \xi(\mu_i)$*. Thus, this policy is sub-optimal if* $\max_i \xi(\mu_i) < \epsilon$*.*

For smooth beliefs, $\bar{\mu}_\xi$ is close to $\hat{\mu}^*_\xi$, and in this case, those heuristics might be reasonable. However, they can be shown to be sub-optimal even for very simple stopping problems.

11.3.3 Bounds on the expected utility

Given that these heuristics are incorrect, what can we actually do? The first thing to try is to calculate the expected utility of some arbitrary policy. As it turns out, this operation is relatively simple in the Bayesian case, even when the set of MDPs is infinite.

Policy evaluation is particularly simple in Bayesian MDP problems. We simply apply the basic utility theory definitions. We first define the Bayes-value function of a policy π to be the expected utility under that policy and our belief ξ.

Expected utility of a policy π for a belief ξ

$$V_\xi^\pi(s) \triangleq \mathbb{E}_\xi^\pi(U \mid s_t = s) \tag{11.3.1}$$

$$= \int_{\mathcal{M}} \mathbb{E}_\mu^\pi(U \mid s_t = s)\, \mathrm{d}\xi(\mu) \tag{11.3.2}$$

$$= \int_{\mathcal{M}} V_\mu^\pi(s)\, \mathrm{d}\xi(\mu) \tag{11.3.3}$$

Algorithm 26 Bayesian Monte-Carlo policy evaluation

input policy π, belief ξ
for $k = 1, \ldots, K$ **do**
  $\mu_k \sim \xi$.
  $\boldsymbol{v}_k = V_{\mu_k}^\pi$
end for
$\boldsymbol{u} = \frac{1}{K} \sum_{k=1}^K \boldsymbol{v}_k$.
return $\boldsymbol{u}$.

The value of any policy gives us a natural lower bound on the Bayes-optimal value function.

We can also get the following upper bounds:

$$V_\xi^* \triangleq \sup_\pi \mathbb{E}_\xi^\pi(U) = \sup_\pi \int_{\mathcal{M}} \mathbb{E}_\mu^\pi(U)\, \mathrm{d}\xi(\mu) \tag{11.3.4}$$

$$\leq \int_{\mathcal{M}} \sup_\pi V_\mu^\pi\, \mathrm{d}\xi(\mu) = \int_{\mathcal{M}} V_\mu^*\, \mathrm{d}\xi(\mu) \triangleq V_\xi^+ \tag{11.3.5}$$

Bounds on $V_\xi^* \triangleq \max_\pi \mathbb{E}(U \mid \pi, \xi)$

Given the previous development, it is easy to see that the following inequalities always hold:

$$V_\xi^\pi \leq V_\xi^* \leq V_\xi^+, \qquad \forall \pi. \tag{11.3.6}$$

These bounds are geometrically demonstrated in Fig. 11.4. They are entirely analogous to the Bayes risk bounds of Sec. **??**.

Algorithm 27 Bayesian Monte-Carlo upper bound

input policy π, belief ξ
for $k = 1, \dots, K$ **do**
  $\mu_k \sim \xi$.
  $\boldsymbol{v}_k = V^*_{\mu_k}$
end for
$\boldsymbol{u}^* = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{v}_k$.
return $\boldsymbol{u}^*$.

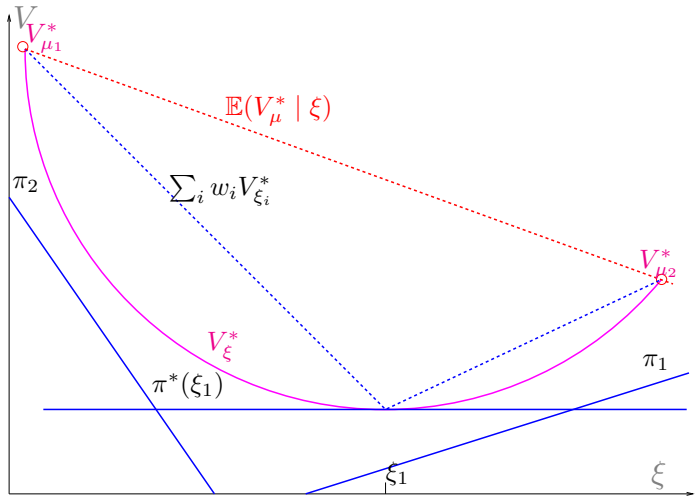
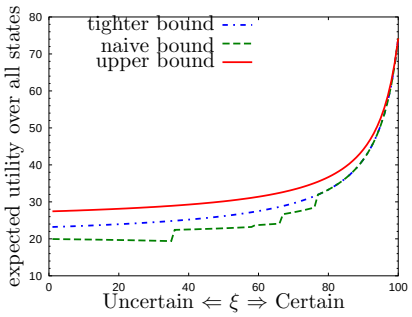


Figure 11.4: A geometric view of the bounds

11.3.4 Tighter lower bounds

One idea to get a better lower bound is to simply find better policies. This idea was explored in Dimitrakakis [2011].



The main idea was to maximise try and find the best memoryless policies. This can be done approximately by assuming that the belief is nearly constant over time, and performing backwards induction on n MDPs simultaneously. While this greedy procedure might not find the optimal memoryless policy, it still improves the lower bounds considerably.

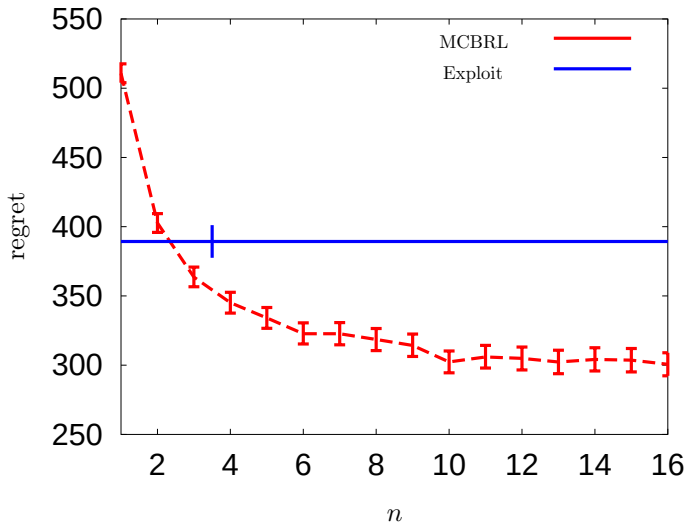
The multi-MDP backwards induction procedure simply involves calculating the expected utility of a particular policy over all MDPs.

$$Q^{\pi}_{\xi,t}(s,a) \triangleq \int_{\mathcal{M}} \left\{ \bar{R}_{\mu}(s,a) + \gamma \int_{\mathcal{S}} V^{\pi}_{\mu,t+1}(s') \, \mathrm{d}\mathcal{T}^{s,a}_{\mu}(s') \right\} \mathrm{d}\xi(\mu) \tag{11.3.7}$$

The algorithm greedily performs backwards induction as shown in Algorithm 28. However, this is not an optimal procedure, since the belief at any time-step t is not constant. Indeed, as the policy is memoryless, $\xi(\mu \mid s_t, \pi) \neq \xi(\mu \mid s_t, \pi')$. This is because the probability of being at a particular state is different under different policies and at different time-steps (e.g. if you consider periodic MDPs). For the same reason, this type of backwards induction may not converge in the manner of value iteration.

Algorithm 28 Multi-MDP backwards induction

MMBI$\mathcal{M}, \xi, \gamma, T$
Set $V_{\mu,T+1}(s) = 0$ for all $s \in \mathcal{S}$.
for $t = T, T-1, \ldots, 0$ **do**
  for $s \in \mathcal{S}, a \in \mathcal{A}$ **do**
    Calculate $Q_{\xi,t}(s,a)$ from (11.3.7) using $\{V_{\mu,t+1}\}$.
  end for
  for $s \in \mathcal{S}$ **do**
    $a^*_{\xi,t}(s) \in \arg\max_{a \in \mathcal{A}} Q_{\xi,t}(s,a)$.
    for $\mu \in \mathcal{M}$ **do**
      $V_{\mu,t}(s) = Q_{\mu,t}(s, a^*_{\xi,t}(s))$.
    end for
  end for
end for



MCBRL: Application to Bayesian RL

1. For $i = 1, \ldots$
2. At time t_i, sample n MDPs from ξ_{t_i}.
3. Calculate best memoryless policy π_i wrt the sample.
4. Execute π_i until $t = t_{i+1}$.

Relation to other work

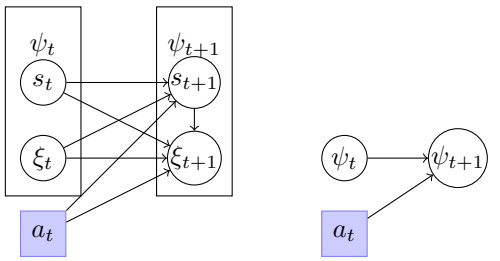
- For $n = 1$, this is equivalent to the Thompson sampling used by Strens Strens [2000].
- Unlike BOSS Asmuth et al. [2009] it does not become more optimistic as n increases. BOSS takes multiple samples and constructs the most optimistic MDP possible in this set.
- BEETLEPoupart et al. [2006], Poupart and Vlassis [2008] is a belief-sampling approach.It examines a set of possible future beliefs and approximates the value of each belief with a lower bound. In essence, it then creates the set of policies which are optimal with respect to these bounds.
- Furmston and Barber Furmston and Barber [2010] use approximate inference to estimate policies. These use the expectation-maximisation view Toussaint et al. [2006] of reinforcement learning.

Generalisations

- Policy search for improving lower bounds.
- Search enlarged class of policies
- Examine all history-based policies.

The augmented MDP

We are given an inital belief ξ_0 on a set of MDPs $\mathcal{M}$. Each $\mu \in \mathcal{M}$ is a tuple $(\mathcal{S}, \mathcal{A}, P_\mu, \boldsymbol{r})$, with state space $\mathcal{S}$, action space $\mathcal{A}$, transition kernel P_μ and reward vector $\boldsymbol{r}$. We now construct the following augmented Markov decision process: $(\mathcal{S} \times \Xi, \mathcal{A}, P, \boldsymbol{r})$, with factorised transition probabilities:



(a) The complete MDP model

(b) Compact form of the model

Figure 11.5: Belief-augmented MDP

The optimal policy for the augmented MDP is the ξ-optimal for the original problem.

$$P(s_{t+1} \in S \mid \xi_t, s_t, a_t) \triangleq \int_S P_\mu(s_{t+1} \in S \mid s_t, a_t) \, \mathrm{d}\xi_t(\mu) \tag{11.3.8}$$

$$\xi_{t+1}(\cdot) = \xi_t(\cdot \mid s_{t+1}, s_t, a_t) \tag{11.3.9}$$

and reward $r_t = \rho(s_t, a_t)$. In the above,

- ξ_t our belief over MDPs $\mu \in \mathcal{M}$ at time t.
- s_t is the observed state of the unknown MDP at time t.
- P_μ is the transition kernel of the MDP μ.
- a_t is our action at time t.
- For simplicity, we assume that r_t be known.

One of the first treatments of this idea was due to Bellman Bellman [1957]. Although the idea was well-known in the statistical community DeGroot [1970], the popularisation of the idea in reinforcement learning was achieved with Duff's thesis Duff [2002].

Belief-augmented MDP tree structure

Given a belief over MDPs, we can create an *augmented* MDP with state space $\Psi = \mathcal{S} \times \Xi$. This has a pseudo-tree structure (since belief states might repeat). Consider an MDP family $\mathcal{M}$ with $\mathcal{A} = \{a^1, a^2\}$, $\mathcal{S} = \{s^1, s^2\}$.

$$\psi_t = (s_t, \xi_t)$$

11.3.5 Stochastic branch and bound

Branch and bound

Branch and bound is a general technique for solving large problems. It can be applied in all cases where upper and lower bounds on the value of solution sets can be found

Value bounds

Let upper and lower bounds $\boldsymbol{q}^+$ and $\boldsymbol{q}^-$ such that:

$$\boldsymbol{q}^+(\psi, a) \geq Q^*(\psi, a) \geq \boldsymbol{q}^-(\psi, a) \tag{11.3.10}$$

$$\boldsymbol{v}^+(\psi) = \max_{a \in \mathcal{A}} Q^+(\psi, a), \qquad \boldsymbol{v}^-(\psi) = \max_{a \in \mathcal{A}} Q^-(\psi, a). \tag{11.3.11}$$

$$\boldsymbol{q}^+(\psi, a) = \sum_{\psi'} p(\psi' \mid \psi, a) \left[r(\psi, a, \psi') + V^+(\psi') \right] \tag{11.3.12}$$

$$\boldsymbol{q}^-(\psi, a) = \sum_{\psi'} p(\psi' \mid \psi, a) \left[r(\psi, a, \psi') + V^-(\psi') \right] \tag{11.3.13}$$

Remark 11.3.1. *If $\boldsymbol{q}^-(\psi, a) \geq \boldsymbol{q}^+(\psi, b)$ then b is sub-optimal at ψ.*

Stochastic branch and bound for belief tree search Dimitrakakis [2010, 2008]

- (Stochastic) Upper and lower bounds on the values of nodes (via Monte-Carlo sampling)
- Use upper bounds to expand tree, lower bounds to select final policy.
- Sub-optimal branches are *quickly* discarded.

11.4 Partially observable Markov decision processes

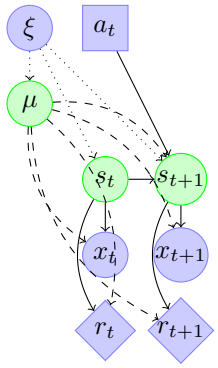
In most real applications, the state of the system is not observed.

Partially observable Markov decision processes (POMDP)

When acting in μ, each time step t:

- The system *state* $s_t \in \mathcal{S}$ is not observed.
- We receive an *observation* $x_t \in \mathcal{X}$ and a *reward* $r_t \in \mathcal{R}$.
- We take *action* $a_t \in \mathcal{A}$.

- The system transits to state s_{t+1}.



Definition 11.4.1. *Partially observable Markov decision process (POMDP) A POMDP $\mu \in \mathcal{M}_P$ is a tuple $(\mathcal{X}, \mathcal{S}, \mathcal{A}, P)$ where $\mathcal{X}$ is an observation space, $\mathcal{S}$ is a state space, $\mathcal{A}$ is an action space, and P is a conditional distribution on observations, states and rewards. The following Markov property holds:*

$$\mathbb{P}_\mu(s_{t+1}, r_t, x_t \mid s_t, a_t, \ldots) = P(s_{t+1} \mid s_t, a_t) P(x_t \mid s_t) P(r_t \mid s_t) \tag{11.4.1}$$

Belief state in POMDPs when μ is known

In POMDPs, we can similarly define a *belief state* summarising our knowledge. This takes the form of a probability distribution on the hidden state variable s_t. *If μ defines starting state probabilities, then the belief is not subjective*

Belief ξ

For any distribution ξ on $\mathcal{S}$, we define:

$$\xi(s_{t+1} \mid a_t, \mu) \triangleq \int_{\mathcal{S}} P_\mu(s_{t+1} \mid s_t a_t) \, \mathrm{d}\xi(s_t) \tag{11.4.2}$$

When there is no ambiguity, we shall use ξ to denote arbitrary marginal distributions on states and state sequence givne the belief ξ.

Belief update

$$\xi_t(s_{t+1} \mid x_{t+1}, r_{t+1}, a_t, \mu) = \frac{P_\mu(x_{t+1}, r_{t+1} \mid s_{t+1}) \xi_t(s_{t+1} \mid a_t, \mu)}{\xi_t(x_{t+1} \mid a_t, \mu)} \tag{11.4.3}$$

$$\xi_t(s_{t+1} \mid a_t, \mu) = \int_{\mathcal{S}} P_\mu(s_{t+1} \mid s_t, a_t, \mu) \, \mathrm{d}\xi_t(s_t) \tag{11.4.4}$$

$$\xi_t(x_{t+1} \mid a_t, \mu) = \int_{\mathcal{S}} P_\mu(x_{t+1} \mid s_{t+1}) \, \mathrm{d}\xi_t(s_{t+1} \mid a_t, \mu) \tag{11.4.5}$$

Example 11.4.1. *If $\mathcal{S}, \mathcal{A}, \mathcal{X}$ are finite, and then we can define*

- $\partial_t(j) = P(x_t \mid s_t = j)$
- $\boldsymbol{A}_t(i, j) = P(s_{t+1} = j \mid s_t = i, a_t)$.
- $\boldsymbol{b}_t(i) = \xi_t(s_t = i)$

We can then use Bayes theorem:

$$\boldsymbol{b}_{t+1} = \frac{\operatorname{diag}(\boldsymbol{p}_{t+1})\boldsymbol{A}_t\boldsymbol{b}_t}{\boldsymbol{p}_{t+1}^\top \boldsymbol{A}_t \boldsymbol{b}_t}, \tag{11.4.6}$$

When the POMDP μ is unknown

$$\xi(\mu, s^t \mid x^t, a^t) \propto P_\mu(x^t \mid s^t, a^t) P_\mu(s^t \mid a^t) \xi(\mu) \tag{11.4.7}$$

Cases

- Finite $\mathcal{M}$.
- Finite $\mathcal{S}$
- General case

Strategies for POMDPs

- Bayesian RL on POMDPs? *EXP inference and planning*
- Approximations and stochastic methods.
- Policy search methods.

Chapter 12

Distribution-free reinforcement learning

12.1 Introduction

The Bayesian framework requires specifying a prior distribution ξ_0. For many reasons, we may frequently be unable to specify such a prior distribution. In addition, as we have seen, the Bayes-optimal solution is frequently intractable. Here we shall take a look at a number of *heuristic* algorithms that do not require specifying a prior distribution. Instead, they employ the heuristic of "optimism under uncertainty" to select policies. This idea is very similar to heuristic search algorithms, such as A^*. All these algorithms select the policy with the highest optimistic value, i.e. with the highest upper bound in its value. The upper bound can be interpreted as the maximum utility we could obtain by playing this policy now, even if we have to switch policies later. In general we want the upper bound to

1. Be as tight as possible
2. Hold with high probability.

We begin with an introduction to these ideas in bandit problems, when the objective is to maximise total reward. We then expand this discussion to structured bandit problems, which have many applications in optimisation. Finally, we look at the case of maximising total reward in unknown MDPs. The same main ideas can be used, but the very definition of an optimal MDP policy is not trivial when we wish to maximise total reward. For this reason, we shall go over the various optimality criteria we can use. We then briefly discuss a nearly-optimal reinforcement learning algorithm.

12.2 Bandit problems

First of all, let us remind the reader of the stochastic bandit problem. We have a choice between a set of K bandits, corresponding to an action set $\mathcal{A} = \{1, \dots, K\}$. The expected reward we get when we play the i-th bandit is $\mu_i \triangleq \mathbb{E}(r_t \mid a_t = i)$ What we wish to do is to maximise the total reward during our life-time $\sum_{t=1}^{T} r_t$, where T is may be random. What is a good heuristic strategy?

Let π_t define a probability distribution over the arms at time t. Let $\mu^* \triangleq \max_i \mu_i$ be the highest average reward we can achieve and let π^* be the policy that always plays the arm with the highest average reward. It is important to recognise that maximising total reward is equivalent to minimising total regret with respect to that policy

Definition 12.2.1 (Total regret)**.** *The (total) regret of a policy π relative to the optimal fixed policy π^* is:*

$$L_T(\pi) \triangleq \sum_{t=1}^{T} r_t^* - r_t^{\pi}. \tag{12.2.1}$$

where $r_t^* \triangleq r_t^{\pi^*}$ is the reward obtained by π^* and r_t^{π} are the rewards generated by the policy π

Here we are comparing with the best fixed policy. This can be seen as the policy that would be played by oracle that knows which is the best bandit on average. We could also compare with the oracle that knows exactly how much reward each bandit would give at different times, but it is hard to prove anything meaningful about that, as the rewards at each time-step are by definition unpredictable. Taking expectations, we have that the expected (total) regret is

$$\mathbb{E}\, L_T(\pi) \triangleq T\mu^* - \mathbb{E}_\pi \sum_{t=1}^{T} r_t. \tag{12.2.2}$$

Now we consider a simple algorithm that uses the empirical average rewards obtained by each bandit.

Empirical average

$$\hat{\mu}_{t,i} \triangleq \frac{1}{n_{t,i}} \sum_{k=1}^{t} r_{k,i}\, \mathbb{I}\left\{a_k = i\right\}, \qquad n_{t,i} \triangleq \sum_{k=1}^{t} \mathbb{I}\left\{a_k = i\right\}.$$

Using the empirical averages directly is not a very good idea, because you might get stuck with a sub-optimal bandit. A better idea is to play bandits optimistically. That is, as long as a particular bandit has a significant chance of being the best, you play it. One way to implement this is through the following algorithm, which assumes that the rewards are bounded, i.e. that $r_t \in \mathcal{R} \subset \mathbb{R}$.

Algorithm 29 Optimistic initial values

Input $\mathcal{A}$, $\mathcal{R}$
$r_{\max} \triangleq \max \mathcal{R}$
for $t = 1, \ldots$ **do**
  $u_{t,i} = \frac{n_{t-1,i}\hat{\mu}_{t-1,i} + r_{\max}}{n_{t-1,i}+1}$
  $a_t = \arg\max_{i \in \mathcal{A}} u_{t,i}$
end for

If you analyse this decision rule carefully, you can see that the algorithm chooses the arm with maximal $\hat{\mu}_i + O(1/n_i)$. That is, it adds a small bonus value to each arm, depending on how many times the arm has been played.

A simple analysis in the deterministic case

When there is no randomness, the algorithm is easy to analyse. Firstly, it must hold that $r_{t,i} = \mu_{t,i}$ for all bandits. Secondly, note that $u_{t,i} \geq \mu_i$ for all t, i. At time t, we will play arm i only if $u_{t,i} \geq u_{t,j}$ for all j. However, $u_{t,j} \geq \mu_j$ and so a necessary condition for us to play arm i is that it $u_{t,i} > \mu_j$ for all arms, including the optimal arm. From this, we obtain that we play i at most

$$n_{t,i} \leq \frac{r_{\max}}{\Delta_i}$$

times, where $\Delta_i \triangleq \mu^* - \mu_i$. This is easy to see since, for us to play i it must hold that

$$\begin{aligned}\frac{n_{t,i}\hat{\mu}_{t,i} + r_{\max}}{n_{t,i}+1} &\geq \mu^* \\ n_{t,i}\hat{\mu}_{t,i} + r_{\max} &\geq \mu^*(n_{t,i}+1) \\ r_{\max} - \mu^* &\geq \Delta_i n_{t,i}\end{aligned}$$

Since every time we play i we lose Δ_i, the regret is

$$L_T \leq \sum_{i \neq j} \Delta_i \frac{r_{\max} - \mu^*}{\Delta_i} = (K-1)(r_{\max} - \mu^*).$$

Unfortunately this algorithm does not have very good properties in the stochastic case. However, an other algorithm, with a choice of actions based upon concentration inequalities, does.

12.2.1 UCB

The idea of UCB is to

Algorithm 30 UCB1

Input $\mathcal{A}$, $\mathcal{R}$
$\hat{\mu}_{0,i} = r_{\max}$, $\forall i$.
for $t = 1, \ldots$ **do**
  $u_{t,i} = \hat{\mu}_{t-1,i} + \sqrt{2\frac{\ln t}{n_{t-1,i}}}$.
  $a_t = \arg\max_{i \in \mathcal{A}} u_{t,i}$
end for

Note that the decision rule in this case uses the heuristic bound of the form $\hat{\mu}_i + O(\sqrt{\ln t / n_i})$ to select the action.

Theorem 12.2.1 (Auer et al Auer et al. [2002])**.** *The expected regret of UCB1 after T rounds is at most*

$$c_1 \sum_{i:\mu_i < \mu^*} \left(\frac{\ln T}{\Delta_i}\right) + c_2 \sum_{j=1}^{K} \Delta_j$$

Proof. First we prove that

$$\mathbb{E}\, n_{t,i} \leq O\left(\frac{\ln T}{\Delta_i^2}\right)$$

Then we note that the expected regret can be written as

$$\sum_{i:\mu_i < \mu^*} \Delta_i \,\mathbb{E}\, n_{t,i}$$

due to Wald's identity. □

Let $B_{t,s} = \sqrt{(2 \ln t)/s}$. Then we can prove $\forall c \in \mathbb{Z}$:

$$\begin{aligned}
n_{T,i} &= 1 + \sum_{t=K+1}^{T} \mathbb{I}\{a_t = i\} \\
&\leq c + \sum_{t=K+1}^{T} \mathbb{I}\{a_t = i \wedge n_{t-1,i} \geq c\} \\
&\leq c + \sum_{t=K+1}^{T} \mathbb{I}\left\{\hat{\mu}^*_{n^*_{t-1}} + B_{t-1,n^*_{t-1}} \leq \max \hat{\mu}_{n_i(t-1),i} + B_{t-1,n_i(t-1)}\right\} \\
&\leq c + \sum_{t=K+1}^{T} \mathbb{I}\left\{\min_{0<s<t} \hat{\mu}^*_s + B_{t-1,s} \leq \max_{c \leq s_i < t} \hat{\mu}_{s_i,i} + B_{t-1,s_i}\right\} \\
&\leq c + \sum_{t=1}^{\infty} \sum_{s=1}^{t-1} \sum_{s_i=c}^{t-1} \mathbb{I}\{\hat{\mu}^*_s + B_{t-1,s} \leq \hat{\mu}_{s_i,i} + B_{t-1,s_i}\}
\end{aligned}$$

When the indicator function is true one of the following holds:

$$\hat{\mu}^*_s \leq \mu^* - B_{t,s} \tag{12.2.3}$$
$$\hat{\mu}_{s_i,i} \geq \mu_i + B_{t,s_i} \tag{12.2.4}$$
$$\mu^* < \mu_i + 2B_{t,s_i} \tag{12.2.5}$$

Proof idea

- Bound the probability of the first two events.
- Choose c to bound the last term.

From Hoeffding bound:

$$\mathbb{P}(\hat{\mu}^*_s \leq \mu^* - B_{t,s}) \leq e^{-4 \ln t} = t^{-4} \tag{12.2.6}$$
$$\mathbb{P}(\hat{\mu}_{s_i,i} \geq \mu_i + B_{t,s_i}) \leq e^{-4 \ln t} = t^{-4} \tag{12.2.7}$$

Setting $c = \lceil (8 \ln n)/\Delta_i^2 \rceil$ makes the last event false as $s_i \geq c$.

$$\mu^* - \mu_i - 2B_{t,s_i} = \mu^* - \mu_i - 2\sqrt{(2 \ln t)/s_i} \geq \mu^* - \mu_i - \Delta_i = 0.$$

Summing up all the terms completes the proof.

12.3 Structured bandit problems

Bandits and optimisation

- Continuous stochastic functionsKocsis and Szepesvári [2006], Auer et al. [2007], Bubeck et al. [2011]
- Constrained deterministic distributed functionsOttens et al. [2012]

First ideaAuer et al. [2007]

Solve a sequence of discrete bandit problems.
At epoch i, we have some interval A_i

- Split the interval A_i in k regions $A_{i,j}$
- Run UCB on the k-armed bandit problem.
- When a region is sub-optimal with high probability, remove it!

Tree bandits Bubeck et al. [2011]

Create a tree of coverings, with (h, i) being the i-th node at depth h. $\mathcal{D}$ are the descendants and $\mathcal{C}$ the children of a node.

At time t we pick node H_t, I_t. Each node is picked at most once.

$$n_{h,i}(T) \triangleq \sum_{t=1}^{T} \mathbb{I}\left\{(H_t, I_t) \in \mathcal{D}(h,i)\right\} \qquad \text{(visits of } (h,i)\text{)}$$

$$\widehat{\mu}_{h,i}(T) \triangleq \frac{1}{n_{h,i}(T)} \sum_{t=1}^{T} r_t \, \mathbb{I}\left\{(H_t, I_t) \in \mathcal{C}(h,i)\right\} \qquad \text{(reward from } (h,i)\text{)}$$

$$C_{h,i}(T) \triangleq \widehat{\mu}_{h,i}(T) + \sqrt{\frac{2 \ln T}{n_{h,i}(T)}} + n u_1 \rho^h \qquad \text{(confidence bound)}$$

$$B_{h,i}(T) \triangleq \min\left\{C_{h,i}(T), \max_{(h+1,j) \in \mathcal{C}(h,i)} B_{h+1,j}\right\} \qquad \text{(child bound)}$$

12.4 Reinforcement learning problems

12.4.1 Optimality Criteria

In all previous cases, we assumed a specific discount rate, or horizon for our problem. Now we shall examine different choices and how they affect the existence of an optimal policy. As mentioned previously, the following two views of discounted reward processes are equivalent.

Infinite horizon, discounted
Discount factor γ such that

$$U_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k} \qquad \Rightarrow \mathbb{E}\, U_t = \sum_{k=0}^{\infty} \gamma^k \, \mathbb{E}\, r_{t+k} \tag{12.4.1}$$

Geometric horizon, undiscounted

At each step t, the process terminates with probability $1-\gamma$:

$$U_t^T = \sum_{k=0}^{T-t} r_{t+k}, \quad T \sim \mathit{Geom}(1-\gamma) \quad \Rightarrow \mathbb{E}\, U_t = \sum_{k=0}^{\infty} \gamma^k \,\mathbb{E}\, r_{t+k} \tag{12.4.2}$$

$$V_\gamma^\pi(s) \triangleq \mathbb{E}(U_t \mid s_t = s)$$

The expected total reward criterion

$$V_t^{\pi,T} \triangleq \mathbb{E}_\pi\, U_t^T, \qquad V^\pi \triangleq \lim_{T\to\infty} V^{\pi,T} \tag{12.4.3}$$

Dealing with the limit

- Consider μ s.t. the limit exists $\forall\pi$.

$$V_+^\pi(s) \triangleq \mathbb{E}_\pi \left(\sum_{t=1}^{\infty} r_t^+ \;\middle|\; s_t = s \right), \quad V_-^\pi(s) \triangleq \mathbb{E}_\pi \left(\sum_{t=1}^{\infty} r_t^- \;\middle|\; s_t = s \right) \tag{12.4.4}$$

$$r_t^+ \triangleq \max\{-r, 0\}, \qquad r_t^- \triangleq \max\{r, 0\}. \tag{12.4.5}$$

- Consider μ s.t. $\exists\pi^*$ for which V^{π^*} exists and

$$\lim_{T\to\infty} V^{\pi^*,T} = V^{\pi^*} \geq \limsup_{T\to\infty} V^{\pi,T}.$$

- Use optimality criteria sensitive to the divergence rate.

The average reward (gain) criterion

The gain g

$$g^\pi(s) \triangleq \lim_{T\to\infty} \frac{1}{T} V^{\pi,T}(s) \tag{12.4.6}$$

$$g_+^\pi(s) \triangleq \limsup_{T\to\infty} \frac{1}{T} V^{\pi,T}(s), \qquad g_-^\pi(s) \triangleq \liminf_{T\to\infty} \frac{1}{T} V^{\pi,T}(s) \tag{12.4.7}$$

If $\lim_{T\to\infty} \mathbb{E}(r_T \mid s_0 = s)$ exists then it equals $g^\pi(s)$.

Let Π be the set of all history-dependent, randomised policies.

Using our overloaded symbols, we have that π^* is *total reward optimal* if

$$V^{\pi^*}(s) \geq V^\pi(s) \qquad \forall s \in \mathcal{S}, \pi \in \Pi.$$

π^* is *discount optimal* for $\gamma \in [0, 1)$ if

$$V_\gamma^{\pi^*}(s) \geq V_\gamma^\pi(s) \qquad \forall s \in \mathcal{S}, \pi \in \Pi.$$

π^* is *gain optimal* if

$$g^{\pi^*}(s) \geq g^\pi(s) \qquad \forall s \in \mathcal{S}, \pi \in \Pi.$$

Overtaking optimality

π^* is *overtaking optimal* if

$$\liminf_{T\to\infty} \left[V^{\pi^*,T}(s) - V^{\pi,T}(s)\right] \geq 0 \qquad \forall s \in \mathcal{S}, \pi \in \Pi.$$

However, no overtaking optimal policy may exist.

π^* is *average-overtaking optimal* if

$$\liminf_{T\to\infty} \frac{1}{T}\left[V^{\pi^*,T}(s) - V_+^\pi(s)\right] \geq 0 \qquad \forall s \in \mathcal{S}, \pi \in \Pi.$$

Sensitive discount optimality

π^* is *n-discount optimal* for $n \in \{-1, 0, 1, \ldots\}$ if

$$\liminf_{\gamma\uparrow 1} (1-\gamma)^{-n}\left[V_\gamma^{\pi^*}(s) - V_\gamma^\pi(s)\right] \geq 0 \qquad \forall s \in \mathcal{S}, \pi \in \Pi.$$

A policy is *Blackwell optimal* if $\forall s$, $\exists \gamma^*(s)$ such that

$$V_\gamma^{\pi^*}(s) - V_\gamma^\pi(s) \geq 0, \qquad \forall \pi \in \Pi, \gamma^*(s)\gamma\gamma < 1.$$

Lemma 12.4.1. *If a policy is m-discount optimal then it is n-discount optimal for all $n \leq m$.*

Lemma 12.4.2. *Gain optimality is equivalent to -1-discount optimality.*

12.4.2 UCRL

An upper-confidence bound algorithm

Confidence region M_t such that

$$\mathbb{P}(\mu \notin M_t) < \delta \tag{12.4.8}$$

Optimistic value for policy π:

$$V_+^\pi(M_t) \triangleq \max\left\{V_\mu^\pi \;\middle|\; \mu \in M_t\right\} \tag{12.4.9}$$

UCRL Jacksh et al. [2010] outline

- At round k, start time t_k, calculate M_{t_k}.
- Choose $\pi_k \in \arg\max_\pi V_+^\pi(M_{t_k})$.
- Execute π_k, observe rewards and update model until t_{k+1}.

The confidence region

Let M_t be a set of plausible MDPs for time t with transitions τ s.t.:

$$\left\| \boldsymbol{P}(\cdot \mid s,a) - \hat{\boldsymbol{P}}_t(\cdot \mid s,a) \right\|_1 \leq \sqrt{\frac{n \ln T}{N_t(s,a)}}, \qquad \forall s \in \mathcal{S}, a \in \mathcal{A}, \tag{12.4.10}$$

where $\hat{\boldsymbol{P}}_t(\cdot \mid s,a)$ is the empirical transition probability.

Then $\mathbb{P}(\mu \in M_t) > 1 - nkT^{-2}$, via a bound due to Weissman Weissman et al. [2003].

Changing set of plausible MDPs

- This implies that we may have to switch policies.
- We do so when $N_t(s,a)$ doubles for some s, a .

Calculating the upper bound

In effect, create an *augmented MDP*

$$Q_t(s,a) = r(s,a) + \max \left\{ \sum_{s' \in \mathcal{S}} \boldsymbol{P}(s' \mid s,a) V_{t+1}(s') \;\middle|\; \|\boldsymbol{P} - \hat{\boldsymbol{P}}\|_1 \leq \epsilon \right\} \tag{12.4.11}$$

$$V_t(s) = \max_{a \in \mathcal{A}} Q_t(s,a) \tag{12.4.12}$$

Comparison with Bayesian upper bound

High-probability value function bound

$$V_+^* = \max \left\{ V_\mu^* \;\middle|\; \mu \in M_t \right\}, \qquad \mathbb{P}(\mu^* \in M_t) \geq 1 - \delta.$$

Highly credible value function bound

$$V_+^* = \max \left\{ V_\mu^* \;\middle|\; \mu \in M_t \right\}, \qquad \xi_t(M_t) \geq 1 - \delta.$$

Bayesian value function bound (e.g. Dimitrakakis [2011])

$$V_+^* = \int_{\mathcal{M}} V_\mu^* \, \mathrm{d}\xi_t(\mu) \qquad \xi_t = \xi_0(\cdot \mid s_t, r_t, \ldots)$$

12.4.3 Bibliographical remarks

Different optimality criteria are treated in detail in Puterman [1994] Chapter 5.

.1 Symbols

.2 Index

Index

.3 Glossary

Bibliography

A. Antos, C. Szepesvári, and R. Munos. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1):89–129, 2008a.

Andrè Antos, Rémi Munos, and Csaba Szepesvari. Fitted q-iteration in continuous action-space mdps. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*. MIT Press, Cambridge, MA, 2008b.

Robert B. Ash and Catherine A. Doleéans-Dade. *Probability & Measure Theory*. Academic Press, 2000.

J. Asmuth, L. Li, M. L. Littman, A. Nouri, and D. Wingate. A Bayesian sampling approach to exploration in reinforcement learning. In *UAI 2009*, 2009.

P. Auer, R. Ortner, and C. Szepesvari. Improved Rates for the Stochastic Continuum-Armed Bandit Problem. In *Learning Theory*, volume 4539 of *Lecture Notes in Computer Science*, page 454. Springer, 2007.

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2/3):235–256, 2002.

Richard Ernest Bellman. A problem in the sequential design of experiments. *Sankhya*, 16:221–229, 1957.

A. Bernstein. Adaptive state aggregation for reinforcement learning. Master's thesis, Technion – Israel Institute of Technolog01y, 2007.

Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

J.A. Boyan. Technical update: Least-squares temporal difference learning. *Machine Learning*, 49(2):233–246, 2002.

S.J. Bradtke and A.G. Barto. Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22(1):33–57, 1996.

Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. X-armed bandits. *Journal of Machine Learning Research*, 12:1655–1695, 2011.

Herman Chernoff. Sequential design of experiments. *Annals of Mathematical Statistics*, 30(3):755–770, 1959.

Herman Chernoff. Sequential Models for Clinical Trials. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol.4*, pages 805–812. Univ. of Calif Press, 1966.

Richard Dearden, Nir Friedman, and Stuart J. Russell. Bayesian Q-learning. In *AAAI/IAAI*, pages 761–768, 1998. URL `citeseer.ist.psu.edu/dearden98bayesian.html`.

Morris H. DeGroot. *Optimal Statistical Decisions*. John Wiley & Sons, 1970.

Christos Dimitrakakis. Tree exploration for Bayesian RL exploration. In *Computational Intelligence for Modelling, Control and Automation, International Conference on*, pages 1029–1034, Wien, Austria, 2008. IEEE Computer Society. ISBN 978-0-7695-3514-2. doi: http://doi.ieeecomputersociety.org/10.1109/CIMCA.2008.32.

Christos Dimitrakakis. Complexity of stochastic branch and bound methods for belief tree search in Bayesian reinforcement learning. In *2nd international conference on agents and artificial intelligence (ICAART 2010)*, pages 259–264, Valencia, Spain, 2010. ISNTICC, Springer.

Christos Dimitrakakis. Robust bayesian reinforcement learning through tight lower bounds. In *European Workshop on Reinforcement Learning (EWRL 2011)*, number 7188 in LNCS, pages 177–188, 2011.

Christos Dimitrakakis. Monte-carlo utility estimates for bayesian reinforcement learning. In *IEEE 52nd Annual Conference on Decision and Control (CDC 2013)*, 2013. arXiv:1303.2506.

Christos Dimitrakakis and Michail G. Lagoudakis. Rollout sampling approximate policy iteration. *Machine Learning*, 72(3):157–171, September 2008. doi: 10.1007/s10994-008-5069-3. Presented at ECML'08.

Michael O'Gordon Duff. *Optimal Learning Computational Procedures for Bayes-adaptive Markov Decision Processes*. PhD thesis, University of Massachusetts at Amherst, 2002.

Milton Friedman and Leonard J. Savage. The expected-utility hypothesis and the measurability of utility. *The Journal of Political Economy*, 60(6):463, 1952.

Thomas Furmston and David Barber. Variational methods for reinforcement learning. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 9 of *JMLR : W&CP*, pages 241–248, Chia Laguna Resort, Sardinia, Italy, 2010.

Mohammad Ghavamzadeh and Yaakov Engel. Bayesian policy gradient algorithms. In *NIPS 2006*, 2006.

C. J. Gittins. *Multi-armed Bandit Allocation Indices*. John Wiley & Sons, New Jersey, US, 1989.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, March 1963.

Thomas Jacksh, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11:1563–1600, 2010.

Levente Kocsis and Csaba Szepesvári. Bandit based Monte-Carlo planning. In *Proceedings of ECML-2006*, 2006.

AN Kolmogorov and SV Fomin. *Elements of the theory of functions and functional analysis*. Dover Publications, 1999.

M.G. Lagoudakis and R. Parr. Least-squares policy iteration. *The Journal of Machine Learning Research*, 4:1107–1149, 2003.

R. Munos and C. Szepesvári. Finite-time bounds for fitted value iteration. *The Journal of Machine Learning Research*, 9:815–857, 2008.

Brammert Ottens, Christos Dimitrakakis, and Boi Faltings. DUCT: An upper confidence bound approach to distributed constraint optimization problems. In *AAAI 2012*, 2012.

P. Poupart, N. Vlassis, J. Hoey, and K. Regan. An analytic solution to discrete Bayesian reinforcement learning. In *ICML 2006*, pages 697–704. ACM Press New York, NY, USA, 2006.

Pascal Poupart and Nikos Vlassis. Model-based Bayesian reinforcement learning in partially observable domains. In *International Symposium on Artificial Intelligence and Mathematics (ISAIM)*, 2008.

Marting L. Puterman. *Markov Decision Processes : Discrete Stochastic Dynamic Programming*. John Wiley & Sons, New Jersey, US, 1994.

H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407, 1951.

Leonard J. Savage. *The Foundations of Statistics*. Dover Publications, 1972.

S. Singh, T. Jaakkola, and M.I. Jordan. Reinforcement learning with soft state aggregation. *Advances in neural information processing systems*, pages 361–368, 1995.

Malcolm Strens. A Bayesian framework for reinforcement learning. In *ICML 2000*, pages 943–950, 2000.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

Marc Toussaint, Stefan Harmelign, and Amos Storkey. Probabilistic inference for solving (PO)MDPs, 2006.

T. Weissman, E. Ordentlich, G. Seroussi, S. Verdu, and M.J. Weinberger. Inequalities for the L_1 deviation of the empirical distribution. *Hewlett-Packard Labs, Tech. Rep*, 2003.