Variable Selection

- Consider a MLR model $Y \sim 1 + X_1 + X_2 + \cdots + X_p$ where we have p non-intercept predictors. Latter on some slides, I drop the intercept for notational simplicity.

- In many applications nowadays, we have a lot of explanatory variables, i.e., p is large and we could even have $p >> n$, but only a small portion of the p variables are believed to be relevant to Y.

- Of interest is to find the following subset of the p predictors

$$S = \{j : \quad \beta_j = 0\}.$$

- In some applications, the key question we need to answer is to identify this set S, e.g., which variables among the p variables are really effective for boosting the sales (Y).
- If our goal is simply to do well on prediction, then should we care about variable selection?

Recall that the LS estimate $\hat{\boldsymbol{\beta}}$ is unbiased, i.e., estimates for the irrelevant variables $\hat{\beta}_j$ (with $j \in S^c$) will eventually go to zero anyway.

To understand this, let's examine the training and the test errors.

Test vs Training Error

- Training data $(\mathbf{x}_i, y_i)_{i=1}^n$
- Test data $(\mathbf{x}_i, y_i^*)_{i=1}^n$ is an independent (imaginary) data set collected at the same location $\mathbf{x}_i$'s (aka, in-sample prediction).
- Assume the data are indeed from a linear model

$$\mathbf{y}_{n\times 1},\ \mathbf{y^*}_{n\times 1} \text{ i.i.d. } \sim \text{N}_n\left(\boldsymbol{\mu}, \sigma^2 \mathbf{I}_n\right),\ \text{where } \mathbf{X}\boldsymbol{\beta} = \boldsymbol{\mu}.$$

Or equivalently, write

$$\begin{aligned} \mathbf{y} &= \mathbf{X}\boldsymbol{\beta} + \mathbf{e}, \\ \mathbf{y}^* &= \mathbf{X}\boldsymbol{\beta} + \mathbf{e}^* \\ \mathbf{e}_{n\times 1},\ \mathbf{e^*}_{n\times 1} \text{ i.i.d. } &\sim \text{N}_n\left(\mathbf{0}, \sigma^2 \mathbf{I}_n\right). \end{aligned}$$

Note that the two error terms $\mathbf{e}_{n\times 1}$ and $\mathbf{e^*}_{n\times 1}$ are independent.

$$\begin{aligned}
\mathbb{E}[\text{Test Err}] &= \mathbb{E}\|\mathbf{y}^* - \mathbf{X}\hat{\boldsymbol{\beta}}\|^2 \\
&= \mathbb{E}\|(\mathbf{y}^* - \mathbf{X}\boldsymbol{\beta}) + (\mathbf{X}\boldsymbol{\beta} - \mathbf{X}\hat{\boldsymbol{\beta}})\|^2 \\
&= \mathbb{E}\|\mathbf{y}^* - \boldsymbol{\mu}\|^2 + \mathbb{E}\|\mathbf{X}\boldsymbol{\beta} - \mathbf{X}\hat{\boldsymbol{\beta}}\|^2 \\
&= \mathbb{E}\|\mathbf{e}^*\|^2 + \text{Tr}\big(\mathbf{X}\text{Cov}(\hat{\boldsymbol{\beta}})\mathbf{X}^t\big) \\
&= n \cdot \sigma^2 + \text{Tr}\mathbf{H} = n \cdot \sigma^2 + p \cdot \sigma^2
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}[\text{Train Err}] &= \mathbb{E}\|\mathbf{y} - \hat{\mathbf{y}}\|^2 = \mathbb{E}\|(\mathbf{I} - \mathbf{H})\mathbf{y}\|^2 \\
&= \text{Tr}\big((\mathbf{I} - \mathbf{H})\text{Cov}(\mathbf{y})(\mathbf{I} - \mathbf{H})^t\big) \\
&= \sigma^2\text{Tr}(\mathbf{I} - \mathbf{H}) = (n - p) \cdot \sigma^2
\end{aligned}$$

So Test Err increases with p and Training Err decreases with p. When adding more variables, i.e., p gets large, although training error gets smaller, but the test error will be large, since the error for estimating p covariates accumulates.

So even if our goal is purely prediction, it is not the more the X variables, the better the prediction. We should remove some irrelevant variables.

The analysis on the previous slides seems to indicate that the best model (i.e., the one with the smallest expected test error) is the intercept-only model with $p = 0$.

This, of course, is not true. The previous analysis is based on the assumption that the mean of $\mathbf{y}$ is in the column space of $\mathbf{X}$, i.e., there exists some coefficient vector $\boldsymbol{\beta}$, such that $\boldsymbol{\mu} = \mathbf{X}\beta$. In general, when we run a linear regression model using only a subset of the columns of $\mathbf{X}$, there will be an additional bias term.

Model Index $\boldsymbol{\gamma}$

- Index each model (i.e., each subset of the p variables) by a p-dimensional binary vector $\boldsymbol{\gamma}$.

  $$\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_p), \quad \gamma_j = 0/1,$$

  where $\gamma_j = 1$ indicates that X_j is included (in the model); 0, otherwise.

- So There are totally 2^p subsets or models. In particular,

  $$\boldsymbol{\gamma} = (1, 1, \ldots, 1)$$

  refers to the full model including all p variables (largest dim), and

  $$\boldsymbol{\gamma} = (0, 0, \ldots, 0)$$

  refers to the intercept-only model (smallest dim).

Suppose $\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$, where $\boldsymbol{\mu}$ denotes the mean of $\mathbf{y}$. If we fit the data $\mathbf{y}$ with respect to model γ, i.e., we fit a linear model with a sub-design-matrix $\mathbf{X}_\gamma$ where $\mathbf{X}_\gamma$ contains only columns from $\mathbf{X}$ such that $\gg_j = 1$.

We can show that the Test Err and Train Error for model γ are

$$\begin{aligned} \mathbb{E}[\text{Test Err}] &= n \cdot \sigma^2 + p \cdot \sigma^2 + \text{Bias}_\gamma \\ \mathbb{E}[\text{Train Err}] &= n \cdot \sigma^2 - p \cdot \sigma^2 + \text{Bias}_\gamma \end{aligned}$$

Bigger model (i.e., p large), small Bias, but large variance ($p\sigma^2$);

Smaller model (i.e., p small), large Bias, but small variance ($p\sigma^2$).

So to reduce the test error (i.e., prediction error), the key is to find the best trade-off between Bias and Variance.

Testing-based Procedures

- Backward

  1. Start with all the predictors in the model.
  2. Remove the predictor with highest p-value great than α_0 (most insignificant).
  3. Refit the model, and repeat the above process.
  4. Stop when all p-values $\leq \alpha_0$.

- α_0 is often set to be 15% or 20%.

- Forward
  1. Start with the intercept-only model.
  2. For all predictors not in the model, check their p-value if being added to the model. Add the one with the lowest p-value $\leq \alpha_0$ (most significant);
  3. Refit the model, and repeat the above process;
  4. Stop when no predictors can be added

- Pros and Cons
  - Main advantage: Computation cost is low.
  - Due to the "one-at-a-time" nature of adding/dropping variables, this type of procedures does not compare all possible models. So it's possible to miss the "optimal" model.
  - It's not clear how to choose α_0, the cut-off for p-values.

Criterion-based Procedures

1. Score each model;

2. Use a search algorithm to find the optimal one.

- Model selection criteria/scores often take the following form

  $$\text{Training error} + \text{Complexity-penalty}.$$

  In the context of linear regression models, complexity of a model increases with the number of predictor variables (i.e., $p_{\boldsymbol{\gamma}}$).

- Why not R^2 or RSS?

Model Selection Criteria

- AIC/BIC

$$\begin{aligned} \textsf{AIC} &: -2 \times \textsf{loglik}_\gamma + 2p_{\boldsymbol{\gamma}} \\ \textsf{BIC} &: -2 \times \textsf{loglik}_\gamma + (\log n)p_{\boldsymbol{\gamma}} \end{aligned}$$

where $p_{\boldsymbol{\gamma}}$ denotes the number of predictors included in model $\boldsymbol{\gamma}$.

For linear regression model

$$-2 \times \textsf{loglik}_\gamma = n \log \frac{\textsf{RSS}_{\boldsymbol{\gamma}}}{n}.$$

Note that when n is large, adding an additional predictor costs a lot more in BIC than AIC. So AIC tends to pick a bigger model than BIC.

- Adjusted R^2 for model $\boldsymbol{\gamma}$

$$\begin{aligned} R_a^2 &= 1 - \frac{\mathsf{RSS}/(n - p_{\boldsymbol{\gamma}} - 1)}{\mathsf{TSS}/(n-1)} \\ &= 1 - R^2 \left(\frac{n-1}{n - p_{\boldsymbol{\gamma}} - 1} \right) \\ &= 1 - \frac{\hat{\sigma}^2_{\boldsymbol{\gamma}}}{\hat{\sigma}^2_0} \end{aligned}$$

- Mallow's C_p

$$\frac{\mathsf{RSS}_{\boldsymbol{\gamma}}}{\hat{\sigma}^2} + 2p_{\boldsymbol{\gamma}} - n$$

where $\hat{\sigma}^2$ is the estimate of the error variance from the full model.

Mallow's C_p behaves very similar to AIC.

Search Algorithms

- Leaps and bounds: return the global optimal solution, but only feasible for less than 50 variables.

  1. Find p models which are the ones with the smallest RSS among models of the same size; [a]

  2. Then evaluate the score on the p models and report the optimal one.

[a] Note that at step 1, we do not need to visit every model. For example, suppose we've know that RSS(X1, X2) $<$ RSS(X3, X4, X5, X6), then we do not need to visit any size 2 or 3 sub-models of (X3, X4, X5, X6), which can be leaped over.

- Greedy algorithms: fast, but only return a local optimal solution (which might be good enough in practice).
  - Backward: start with the full model and sequentially delete predictors until the score does not improve.
  - Forward: start with the null model and sequentially add predictors until the score does not improve.
  - Stepwise: consider both deleting and adding one predictor at each stage.