Lecture 26: Positive semidefinite matrices, tensor products, Spectral graph analysis

Tuesday, November 24, 2015 9:30 AM

# POSITIVE SEMI-DEFINITE MATRICES

## WHY CARE?

**Reading**: Meyer §7.6, Strang ch. 6

**Definition:**

Symmetric matrix $A$ is positive definite $\Longleftrightarrow$ Eigenvalues all $> 0$

$\Updownarrow$ we proved this!

$x^TAx > 0$ for all $x \neq \vec{0}$

Symmetric matrix $A$ is positive semi-definite $\Longleftrightarrow$ Eigenvalues all $\geq 0$

$\Updownarrow$

$x^TAx \geq 0$ for all $x \neq \vec{0}$

**Ways to show that $A \succeq 0$**

- compute all e-values, check $\geq 0$ (lame)
- write $A = B + C$ with $B \succeq 0$, $C \succeq 0$ (or $B+C+D$, etc)
- show that $x^TAx \geq 0$ for all $\vec{x}$

**Ways to show $A \not\succeq 0$**

- find an e-value $< 0$
- find $\vec{x}$ so $x^TAx < 0$
- show that a **submatrix** of $A$ is not $\succeq 0$
- check $\mathrm{Det}(A) < 0$ $\left(= \prod_{j=1} \lambda_j\right)$

Maybe also consider rank(A), ...

**Recall**: A function $T: V \rightarrow W$ is linear if

- $T(x+y) = T(x) + T(y)$ for all $x, y \in V$,
- $T(\alpha x) = \alpha T(x)$ for all scalars $\alpha$.

Linear functions $\longleftrightarrow$ Matrices

$T(\alpha x) = \alpha T(x)$ for all scalars $\alpha$.

Linear functions $\xleftrightarrow[\text{for } V \,\&\, W]{\text{w/ fixed bases}}$ Matrices

Not all functions are linear... but calculus lets us give affine approximations to differentiable functions.

Calculus $\longrightarrow$ linearity

$$f(x) - f(x_0) \approx f'(x_0)\cdot(x - x_0)$$

for $f: \mathbb{R} \to \mathbb{R}$ differentiable

Or for $f: \mathbb{R}^n \to \mathbb{R}^m$ differentiable,

$$\vec{f}(\vec{y}) - \vec{f}(y_0) \approx \frac{\partial \vec{f}}{\partial x_1}(\vec{y}_0)\cdot(\vec{y}-\vec{y}_0)_1 + \cdots + \frac{\partial \vec{f}}{\partial x_n}(\vec{y}_0)\cdot(\vec{y}-\vec{y}_0)_n$$

$$= \begin{pmatrix} | & | & & | \\ \frac{\partial \vec{f}}{\partial x_1} & \frac{\partial \vec{f}}{\partial x_2} & \cdots & \frac{\partial \vec{f}}{\partial x_n} \\ | & | & & | \end{pmatrix} \begin{pmatrix} | \\ \vec{y} - \vec{y}_0 \\ | \end{pmatrix}$$

$$\sum_{j=1}^{n} \frac{\partial \vec{f}}{\partial x_j}\cdot \vec{e}_j^{\,T}$$

But often, an affine approximation isn't enough!

Eg., studying local optima requires a quadratic approx$^{n}$:

$$f(x) = f(x_0) + f'(x_0)\cdot(x - x_0) + \frac{1}{2} f''(x_0)\cdot(x-x_0)^2 + O((x-x_0)^3)$$

Single-variable functions

local minimum $\longleftrightarrow$ $f'(x_0) = 0,\ f''(x_0) > 0$

local maximum $\longleftrightarrow$ $f'(x_0) = 0,\ f''(x_0) < 0$

(if $f''(x_0) \neq 0$)

Multi-variable functions $f: \mathbb{R}^n \to \mathbb{R}$

$$f(y) = f(y_0) + \sum_j \frac{\partial f}{\partial x_j}(y_0)\cdot(y-y_0)_j + \frac{1}{2}\sum_{j,k} \frac{\partial^2 f}{\partial x_j \partial x_k}\cdot(y-y_0)_j (y-y_0)_k + O(\|y-y_0\|^3)$$

For $f: \mathbb{K}^n \to \mathbb{K}$, $y_0$ is a local minimum if

$$\frac{\partial f}{\partial x_j}(y_0) = 0 \quad \forall j = 1, \dots, n$$

and $$\underbrace{\frac{1}{2}\sum_{j,k} \frac{\partial^2 f}{\partial x_j \partial x_k}(y_0) \cdot (y-y_0)_j (y-y_0)_k}_{\text{This is a quadratic form in the variables } (y-y_0)_1, \dots, (y-y_0)_n} > 0$$

This is a quadratic form in the variables $(y-y_0)_1, \dots, (y-y_0)_n$

**Definition:** A quadratic form is a degree-two homogeneous polynomial.

$x^2+2xy+y^2$ ✓ $\qquad$ $x^2+2x$ ✗

$x^2-2xz+2y^2-2yz+z^2$ ✓ $\qquad$ $x^2+2xy+y$ ✗

**Observe:** Quadratic forms $\updownarrow$ Symmetric matrices

**Corollary:** Symmetric matrices let us study homogeneous quadratic functions.

$$x^2+2xy+y^2 = (x\ y)\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}$$

It also equals $(x\ y)\begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}$ and $(x\ y)\begin{pmatrix} 1 & 3/2 \\ 1/2 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}$, but the symmetric form is nicer.

**Examples:**

- $f(x,y) = x^2 + (2b)xy + y^2$
  $$= (x\ y)\underbrace{\begin{pmatrix} 1 & b \\ b & 1 \end{pmatrix}}_{A}\begin{pmatrix} x \\ y \end{pmatrix}$$

  $\lambda_1 + \lambda_2 = Tr(A) = 2$

  $\lambda_1 \cdot \lambda_2 = Det(A) = 1 - b^2$

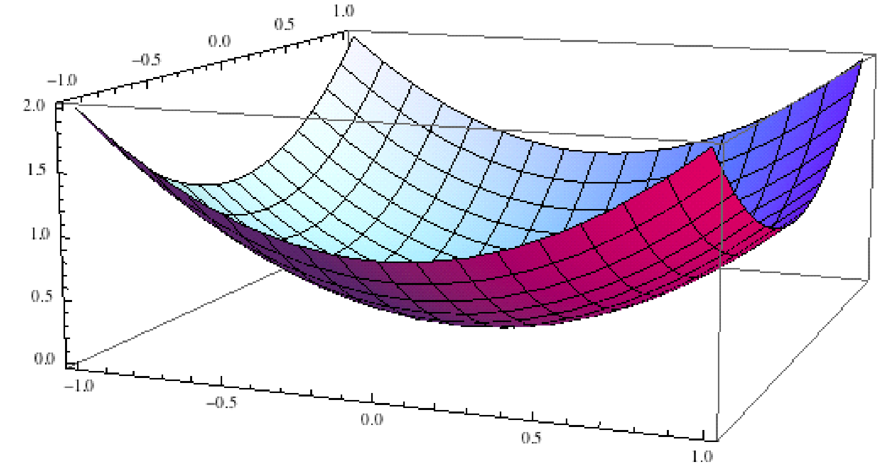
  $\Rightarrow A \succ 0$ if $b < 1$

  $A \succeq 0$ if $b \leq 1$

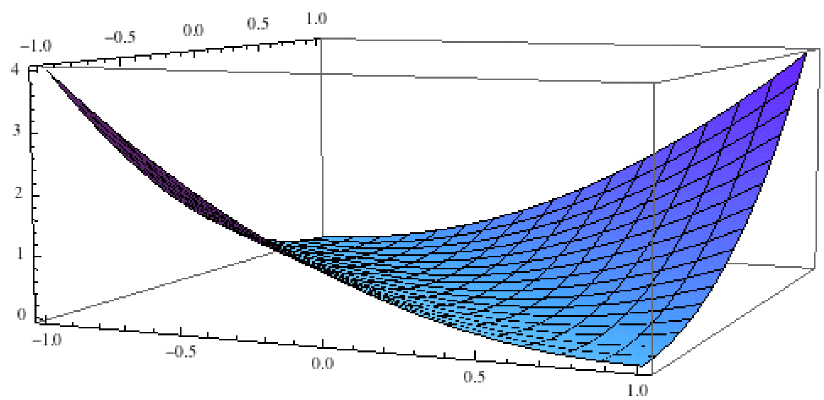
  $A \not\succeq 0$ if $b > 1$

  (A is indefinite)

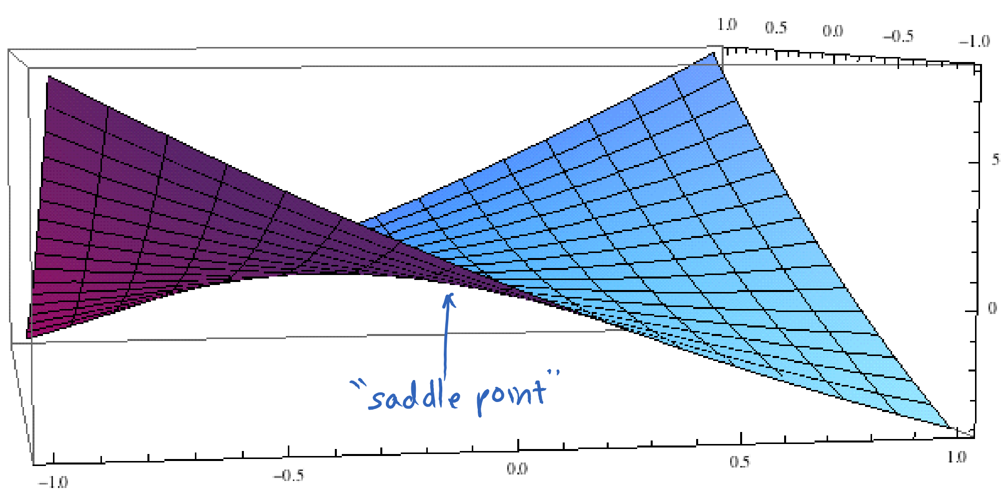
```
b = 0;
Plot3D[x^2 + 2 b x y + y^2, {x, -1, 1}, {y, -1, 1}]
```




```
b = 1;
Plot3D[x^2 + 2 b x y + y^2, {x, -1, 1}, {y, -1, 1}]
```




```
b = 3;
Plot3D[x^2 + 2 b x y + y^2, {x, -1, 1}, {y, -1, 1}]
```




$$\begin{pmatrix} 1 & b \\ b & 1 \end{pmatrix} = \begin{pmatrix} 0 & b \\ b & 0 \end{pmatrix} + I$$

| | Eigenvalues | Eigenvectors |
|---|---|---|
| | $1+b$ | $(1, 1)$ |
| | $1-b$ | $(1, -1)$ |

$\begin{pmatrix} 0 & b \\ b & 0 \end{pmatrix} + I \qquad 1-b \qquad (1,-1)$

$$\Rightarrow \begin{pmatrix} 1 & b \\ b & 1 \end{pmatrix} = (1+b)\cdot \tfrac{1}{2}\begin{pmatrix} 1 \\ 1 \end{pmatrix}\begin{pmatrix} 1 & 1 \end{pmatrix} + (1-b)\cdot \tfrac{1}{2}\begin{pmatrix} 1 \\ -1 \end{pmatrix}\begin{pmatrix} 1 & -1 \end{pmatrix}$$

$$\Rightarrow f(x,y) = \begin{pmatrix} x & y \end{pmatrix}\begin{pmatrix} 1 & b \\ b & 1 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = x^2 + y^2 + 2bxy$$

$$= (1+b)\cdot\left(\tfrac{1}{\sqrt{2}}(1,1)\cdot(x,y)\right)^2$$

$$+ (1-b)\cdot\left(\tfrac{1}{\sqrt{2}}(1,-1)\cdot(x,y)\right)^2$$

$$= \frac{1+b}{2}(x+y)^2 + \frac{1-b}{2}(x-y)^2$$

**Observe:** Any quadratic form can be expressed as a sum of squares like this, and it is $\geq 0$ iff all coefficients are $\geq 0$.

By writing $A = \lambda_1 v_1 v_1^T + \cdots + \lambda_n v_n v_n^T$,

$$x^T A x = \sum_j \lambda_j (v_j \cdot x)^2 \quad ✓$$

**Claim:** $A \succeq 0 \iff A = M^T M$ for some matrix $M$.

Proof:

If $A = M^T M$, $x^T A x = \|Mx\|^2 \geq 0$

$\Rightarrow A \succeq 0$. ✓

If $A \succeq 0$, $A = \sum_j \lambda_j v_j v_j^T$.

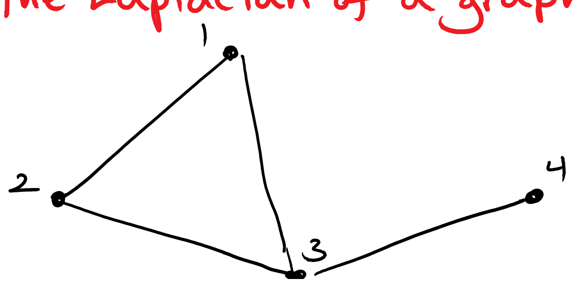
Let $M = \begin{pmatrix} \text{---} \sqrt{\lambda_1}\, v_1^T \text{---} \\ \text{---} \sqrt{\lambda_2}\, v_2^T \text{---} \\ \vdots \end{pmatrix} \Rightarrow M^T M = A$ ✓

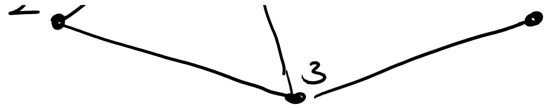
$\square$

(**Remark:** This also proves that $\underset{n\times n}{A} \succeq 0$ if and only if $A$ is the Gram matrix for a set of $n$ vectors.)

**Example:**

The Laplacian of a graph






$$L = \begin{pmatrix} \text{degrees along diagonal} & -\text{edges} \\ -\text{edges} & \end{pmatrix} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \end{array}$$

$L = E E^T$ for the incidence matrix $E$

$$E = \sum_{\substack{\text{edges} \\ (u,v)}} (e_u - e_v) e_{(u,v)}^T$$

$$\Rightarrow L \succeq 0$$

$$x^T L x = \sum_{\substack{\text{edges} \\ (u,v)}} (x_u - x_v)^2$$

**Question**: Is $L$ positive definite?

Answer: No! $L\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} = 0$.

**Observe**: For $f: \mathbb{R}^n \to \mathbb{R}$, a "critical point" (where all derivatives $\frac{\partial f}{\partial x_j} = 0$) is a local minimum if

$$\begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} & \\ \vdots & & \ddots \end{pmatrix}$$

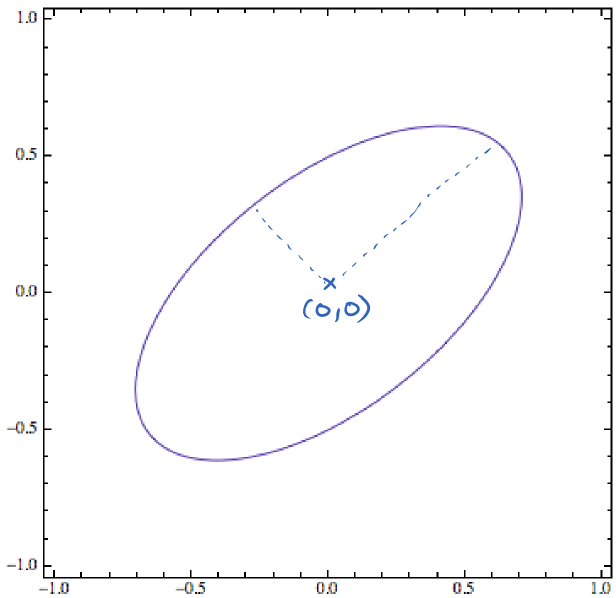
is positive definite.

**Geometrically**:

Quadratic forms assoc. to positive definite matrices $\longleftrightarrow$ Ellipses and ellipsoids

```
In[1]:= A = (3  -2
             -2  4);
        Eigenvalues[A // N]
        Eigensystem[A] // N // Transpose // MatrixForm
        ContourPlot[{x, y}.A.{x, y} == 1, {x, -1, 1}, {y, -1, 1}]

Out[2]= {5.56155, 1.43845}

Out[3]//MatrixForm=
        (5.56155  {-0.780776, 1.}
         1.43845  {1.28078, 1.})
```

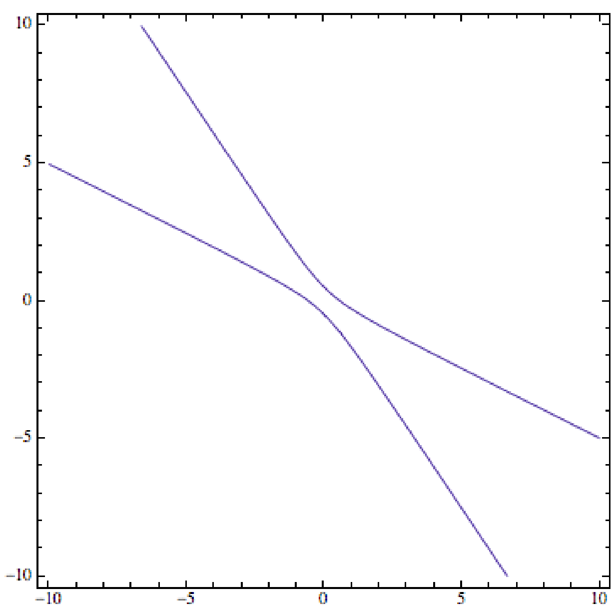
Out[4]=




ratio $\frac{\text{longest}}{\text{shortest}}$ axis

$\leftrightarrow$ condition number

```
A = (3  4
     4  4);
Eigenvalues[A // N]
ContourPlot[{x, y}.A.{x, y} == 1, {x, -10, 10}, {y, -10, 10}]

{7.53113, -0.531129}
```



Ellipsoids also correspond to inner products/norms where some directions are weighted differently than others.

# Example: Gradient Descent

**Goal**: Solve $A\vec{x} = \vec{b}$.

Define the cost function

$$C(\vec{x}) = \| Ax - b \|^2$$
$$= \| A(x - x^*) \|^2 \quad \text{where } Ax^* = b$$
$$= (x - x^*)^T \underbrace{A^T A}_{\text{positive semi-definite}} (x - x^*)$$

**Algorithm**: Repeat until convergence:

$$x^t = x^{t-1} - \alpha \left( \frac{\partial C}{\partial x_1}(x^{t-1}), \dots, \frac{\partial C}{\partial x_n}(x^{t-1}) \right)$$

i.e., step in the direction opposite the largest increase in $C$.

$$\frac{\partial C}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ (Ax - b)^T (Ax - b) \right]$$
$$= \frac{\partial}{\partial x_j} \sum_i (Ax - b)_i^2$$
$$= 2 \sum_i (Ax - b)_i \cdot \frac{\partial}{\partial x_j} \left[ (Ax - b)_i \right]$$
$$= 2 \sum_i (Ax - b)_i \cdot a_{ij}$$

$$\Rightarrow \left( \frac{\partial C}{\partial x_1}, \dots, \frac{\partial C}{\partial x_n} \right) = 2 \sum_i \underbrace{(Ax - b)_i}_{(Ax-b)^T e_i} \cdot \underbrace{(a_{i1}, \dots, a_{in})}_{e_i^T A}$$
$$= 2 (Ax - b)^T \left( \sum_i e_i e_i^T \right) A$$
$$= 2 (Ax - b)^T A$$

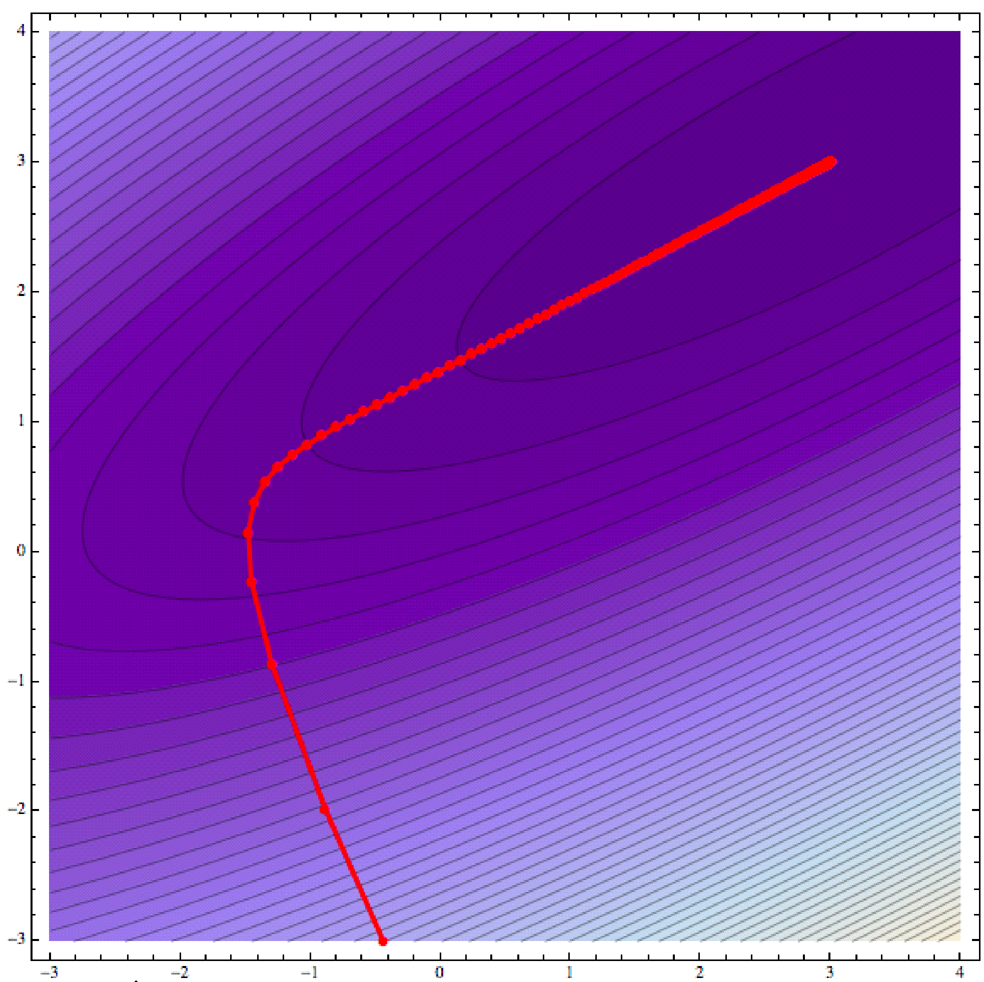
(As you might guess!)

**Example**:

```
A = ( 2     -1.5 )
    ( -1.5   4   );
b = A.{3, 3};
Eigenvalues[A]

α = .01;
x = {0, -4};
list = {x};
For[k = 1, k ≤ 1000, k++,
  x = x - 2 α (A.x - b).A;
  AppendTo[list, x];
 ];
"The solution is:"
x
```

```
{4.80278, 1.19722}

The solution is:

{3., 3.}
```

```
"Display the results:";
iter = ListPlot[list, PlotRange → 3 {{-1, 1}, {-1, 1}}, Joined → True,
   PlotStyle → {Red, Thickness[.005]}, PlotMarkers → Automatic];
contour = ContourPlot[Norm[A.{x, y} - b]^2, {x, -3, 4}, {y, -3, 4}, Contours → 50];
Show[contour, iter]
```




For small enough α, this has to converge since there is

For small enough $\alpha$, this has to converge, since there is a unique local and global minimum.

**Observe**: Smaller condition number
$\Rightarrow$ Rounder ellipse $\Rightarrow$ Faster convergence

**Remark**: Although not generally the fastest way of solving a set of linear equations, the gradient descent approach is very robust, and **many** variants are used in **many** applications.

## One more motivation for positive semi-definite matrices: Semi-definite programming

This was touched on in the homework....

SDPs $\max \ \mathrm{Trace}(AB)$ are everywhere!

$\quad\quad$ s.t. $A \succeq 0$

They can be solved efficiently and have a notion of duality roughly because the set of p.s.d. matrices is convex.

# TENSOR PRODUCTS

Inputs: $V \cong \mathbb{R}^m$, $W \cong \mathbb{R}^n$

Output: $V \otimes W \cong \mathbb{R}^{m \cdot n}$

**Tensor products of vectors**

$$\begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \otimes \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} p_1 \cdot q_1 \\ p_1 \cdot q_2 \\ p_2 \cdot q_1 \\ p_2 \cdot q_2 \end{pmatrix} \in \mathbb{R}^4$$

$$\begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix} \otimes \begin{pmatrix} q_1 \\ q_2 \end{pmatrix} = \begin{pmatrix} p_1 \cdot q_1 \\ p_1 \cdot q_2 \\ p_2 \cdot q_1 \\ p_2 \cdot q_2 \\ p_3 \cdot q_1 \\ p_3 \cdot q_2 \end{pmatrix} \in \mathbb{R}^{6 = 3 \times 2}$$

Properties:

- $u \otimes v + u \otimes w = u \otimes (v + w)$
- $u \otimes w + v \otimes w = (u + v) \otimes w$
- $u \otimes u' + v \otimes v' \neq (u+v) \otimes (u' + v')$
  $= u \otimes u' + v \otimes v' + u \otimes v' + v \otimes u'$
- $(\alpha u) \otimes v = u \otimes (\alpha v) = \alpha (u \otimes v)$
- If $\{\vec{u}_1, \dots, \vec{u}_m\}$ is a basis for $\mathcal{U}$, and $\{\vec{v}_1, \dots, \vec{v}_n\}$ a basis for $V$, then $\{\vec{u}_i \otimes \vec{v}_j\}_{\substack{i=1..m \\ j=1..n}}$ is a basis for $\mathcal{U} \otimes V$

# Tensor products of operators

How to randomize a coin:

multiply by $\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}$

since $\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix} \begin{pmatrix} q \\ 1-q \end{pmatrix} = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix}$

How to randomize the $2^{nd}$ of 2 coins:

multiply by $M = \begin{pmatrix} 1/2 & 1/2 & & \\ 1/2 & 1/2 & \multicolumn{2}{c}{0} \\ & & 1/2 & 1/2 \\ \multicolumn{2}{c}{0} & 1/2 & 1/2 \end{pmatrix}$

since $M \begin{pmatrix} a \\ b \\ c \\ d = 1-a-b-c \end{pmatrix} \begin{array}{l} HH \\ HT \\ TH \\ TT \end{array} = \begin{pmatrix} (a+b)/2 \\ (a+b)/2 \\ (c+d)/2 \\ (c+d)/2 \end{pmatrix}$

And the first coin?

$$\begin{pmatrix} 1/2 & 0 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/2 \\ \dots \end{pmatrix} \begin{pmatrix} a \\ b \\ \dots \end{pmatrix} \begin{array}{l} HH \\ HT \end{array} \begin{pmatrix} (a+c)/2 \\ (b+d)/2 \\ \dots \end{pmatrix}$$

And the first coin:

$$\begin{pmatrix} 1/2 & 0 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/2 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \begin{matrix} HH \\ HT \\ TH \\ TT \end{matrix} \begin{pmatrix} (a+c)/2 \\ (b+d)/2 \\ (a+c)/2 \\ (b+d)/2 \end{pmatrix}$$

**Definition:**

$$A \otimes B = \begin{pmatrix} a_{11} B & a_{12} B & \cdots \\ a_{21} B & a_{22} B & \\ & \vdots & \ddots \end{pmatrix}$$

$$\Rightarrow I \otimes \begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix} = \begin{pmatrix} 1/2 & 1/2 & & \\ 1/2 & 1/2 & & 0 \\ & & 1/2 & 1/2 \\ 0 & & 1/2 & 1/2 \end{pmatrix}$$

$$\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix} \otimes I = \begin{pmatrix} I/2 & I/2 \\ I/2 & I/2 \end{pmatrix}$$

$$= \frac{1}{2} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \quad \checkmark$$

$$\boxed{(A \otimes B)_{(i,j),(k,l)} = A_{ik} B_{jl}}$$

**Properties:**

- $(A \otimes B)(\vec{u} \otimes \vec{v}) = (A\vec{u}) \otimes (B\vec{v})$

Proof:

$$\left[(A \otimes B)(u \otimes v)\right]_{(i,j)} = \sum_{(k,l)} (A \otimes B)_{(i,j),(k,l)} (u \otimes v)_{(k,l)}$$
$$= \sum_{k,l} A_{ik} B_{jl} u_k v_l$$
$$= \left(\sum_k A_{ik} u_k\right)\left(\sum_l B_{jl} v_l\right)$$
$$= (A\vec{u})_i (B\vec{v})_j$$
$$= \left[(A\vec{u}) \otimes (B\vec{v})\right]_{(i,j)}$$

- $(A \otimes B) \cdot (C \otimes D) = (AC) \otimes (BD)$

(Exercise)

- $\mathrm{Trace}(A \otimes B) = \mathrm{Trace}(A) \cdot \mathrm{Trace}(B)$

(Exercise)

- Eigenvalues & eigenvectors:

If $A\vec{u} = \lambda\vec{u}$

$B\vec{v} = \sigma\vec{v}$,

then $(A\otimes B)(\vec{u}\otimes\vec{v}) = (\lambda\vec{u})\otimes(\sigma\vec{v})$

$= (\lambda\sigma)(\vec{u}\otimes\vec{v})$

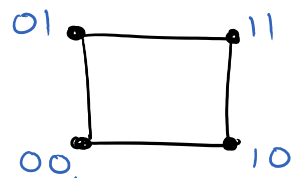
— and this accounts for all $m\cdot n$ eigenvectors

## Example: THE HYPERCUBE

$n=1$:

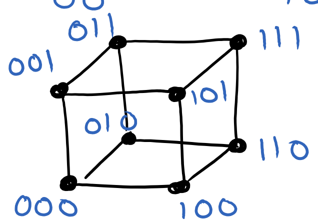

vertices → bit strings of length $n$

$n=2$:

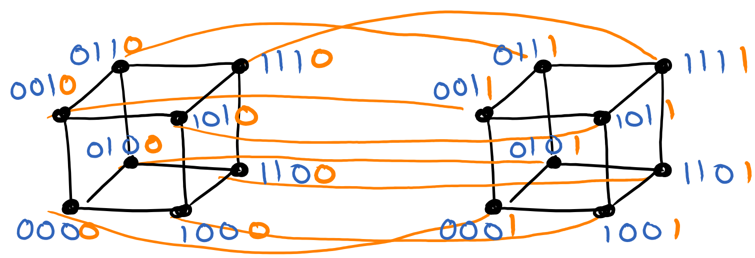



$x \underset{\text{adjacent to}}{\sim} y \longleftrightarrow$ they differ in one coordinate

$n=3$:




$n=4$:




Adjacency matrix:

$$A = \sum_{\text{edges}(u,v)} \left(\vec{e}_u\vec{e}_v^{\,T} + \vec{e}_v\vec{e}_u^{\,T}\right)$$

$$A_{H_1} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad A_{H_2} = \begin{array}{c|c} & \begin{matrix} 00 & 10 & 01 & 11 \end{matrix} \\ \begin{matrix} 00 \\ 10 \\ 01 \\ 11 \end{matrix} & \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

$$A_{H_3} = \begin{array}{c|c} & \begin{matrix} 000 & 100 & 010 & 110 & 001 & 101 & 011 & 111 \end{matrix} \\ \begin{matrix} 000 \\ 100 \\ 010 \\ 110 \\ 001 \\ 101 \\ 011 \\ 111 \end{matrix} & \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & & & \\ 1 & 0 & 0 & 1 & & 1 & & \\ 1 & 0 & 0 & 1 & & & 1 & \\ 0 & 1 & 1 & 0 & & & & 1 \\ 1 & & & & 0 & 1 & 1 & 0 \\ & 1 & & & 1 & 0 & 0 & 1 \\ & & 1 & & 1 & 0 & 0 & 1 \\ & & & 1 & 0 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

$$\Rightarrow A_{H_n} = I_2 \otimes A_{H_{n-1}} + A_{H_1} \otimes I_{2^{n-1}}$$

$$= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \otimes I_2^{\otimes(n-1)} + I_2 \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \otimes I_2^{\otimes(n-2)} + \cdots + I_2^{\otimes(n-1)} \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

**Problem:** What are the eigenvalues of $A_{H_n}$?

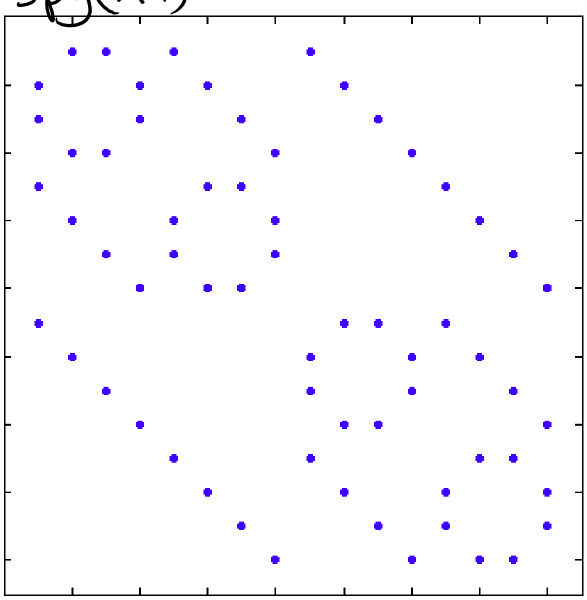
Step one: experiment!

```
>> X = [0 1; 1 0];
I = eye(2);
>> kron(X,I)

ans =

     0     0     1     0
     0     0     0     1
     1     0     0     0
     0     1     0     0

>> kron(I,X)

ans =

     0     1     0     0
     1     0     0     0
     0     0     0     1
     0     0     1     0

>> X2 = kron(X,I) + kron(I,X)

X2 =

     0     1     1     0
     1     0     0     1
     1     0     0     1
     0     1     1     0
```

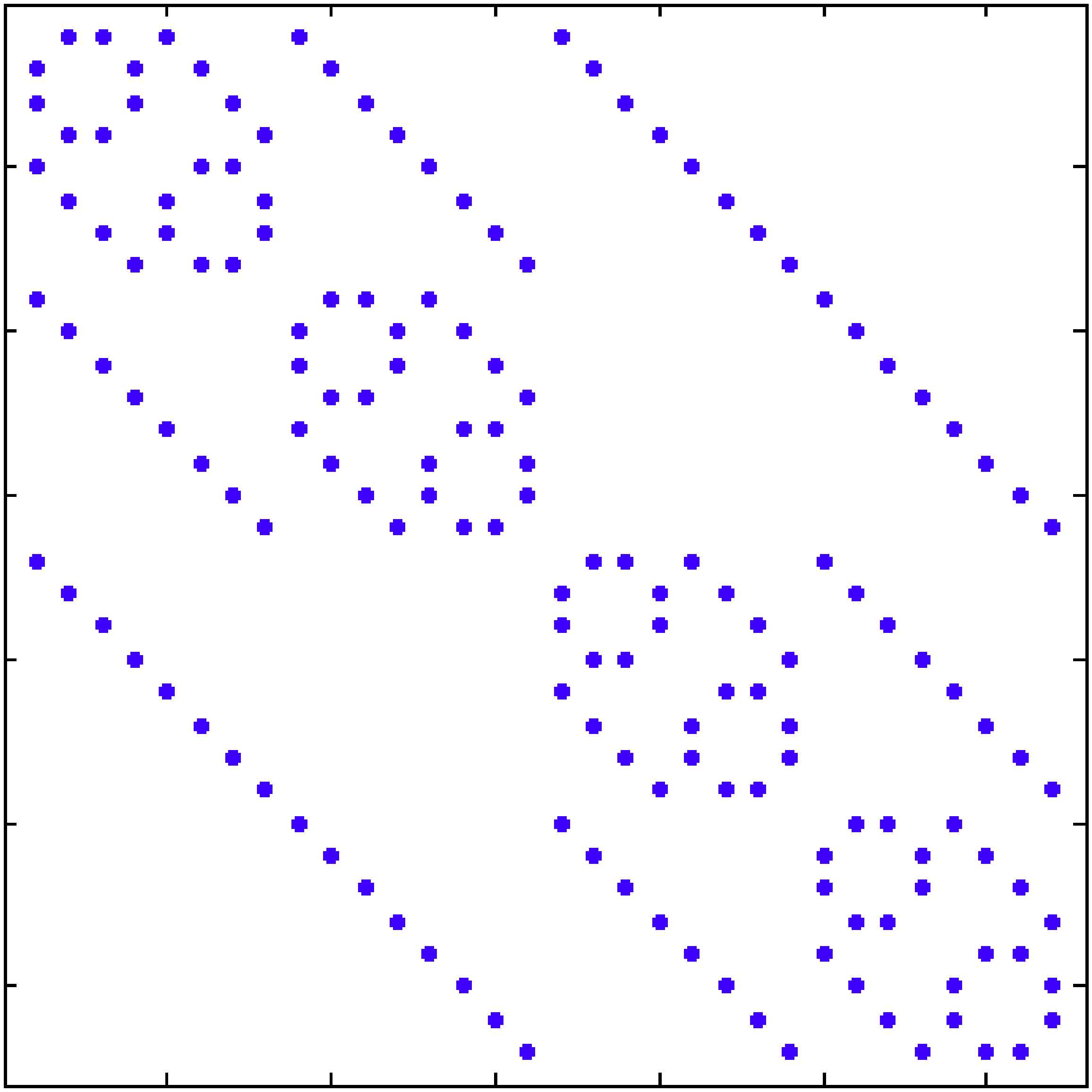
```
>> X3 = kron(X,eye(2^2)) + kron(I,X2)

X3 =

     0     1     1     0     1     0     0     0
     1     0     0     1     0     1     0     0
     1     0     0     1     0     0     1     0
     0     1     1     0     0     0     0     1
     1     0     0     0     0     1     1     0
     0     1     0     0     1     0     0     1
     0     0     1     0     1     0     0     1
     0     0     0     1     0     1     1     0
```

```
>> X4 = kron(X,eye(2^3)) + kron(I,X3)

X4 =

     0     1     1     0     1     0     0     0     1     0     0     0     0     0     0     0
     1     0     0     1     0     1     0     0     0     1     0     0     0     0     0     0
     1     0     0     1     0     0     1     0     0     0     1     0     0     0     0     0
     0     1     1     0     0     0     0     1     0     0     0     1     0     0     0     0
     1     0     0     0     0     1     1     0     0     0     0     0     1     0     0     0
     0     1     0     0     1     0     0     1     0     0     0     0     0     1     0     0
     0     0     1     0     1     0     0     1     0     0     0     0     0     0     1     0
     0     0     0     1     0     1     1     0     0     0     0     0     0     0     0     1
     1     0     0     0     0     0     0     0     0     1     1     0     1     0     0     0
     0     1     0     0     0     0     0     0     1     0     0     1     0     1     0     0
     0     0     1     0     0     0     0     0     1     0     0     1     0     0     1     0
     0     0     0     1     0     0     0     0     0     1     1     0     0     0     0     1
     0     0     0     0     1     0     0     0     1     0     0     0     0     1     1     0
     0     0     0     0     0     1     0     0     0     1     0     0     1     0     0     1
     0     0     0     0     0     0     1     0     0     0     1     0     1     0     0     1
     0     0     0     0     0     0     0     1     0     0     0     1     0     1     1     0
```

spy(X4):



spy(X5):



Step two: Solve for the eigenvalues!

**Proposition**: If $A$'s eigenvalues are $\lambda_1, \dots, \lambda_m$ and $B$'s eigenvalues are $\sigma_1, \dots, \sigma_n$, then the eigenvalues of $A \otimes I_n + I_m \otimes B$ are

$$\lambda_1+\sigma_1, \lambda_1+\sigma_2, \dots, \lambda_1+\sigma_n,$$
$$\lambda_2+\sigma_1, \dots, \lambda_2+\sigma_n,$$
$$\dots,$$
$$\lambda_m+\sigma_1, \dots, \lambda_m+\sigma_n$$

i.e. all sums (eigenvalue of $A$) + (eigenvalue of $B$).

Proof:

① Eigenvalues of $A \otimes I_n$:

$$\underbrace{\lambda_1, \lambda_1, \dots, \lambda_1}_{n \text{ times}}, \dots, \underbrace{\lambda_m, \dots, \lambda_m}_{n \text{ times}}$$

since $I_n$ has e-value 1 with multiplicity $n$.

② Eigenvalues of $I_m \otimes B$:

$\sigma_1$ ($m$ times), ..., $\sigma_n$ ($m$ times)

③ $A \otimes I_n$ and $I_m \otimes B$ commute!

$$\begin{aligned}(A \otimes I)(I \otimes B) &= (AI) \otimes (IB) \\ &= A \otimes B \\ &= (I \otimes B)(A \otimes I)\end{aligned}$$

$\Rightarrow$ they can be simultaneously diagonalized.

If $A\vec{u}_i = \lambda_i \vec{u}_i$

$B\vec{v}_j = \sigma_j \vec{v}_j$,

then $(A \otimes I + I \otimes B)(\vec{u}_i \otimes \vec{v}_j) = (\lambda_i + \sigma_j)(\vec{u}_i \otimes \vec{v}_j)$ ✓ □

**Theorem**: The eigenvalues of the adjacency matrix for the $n$-dimensional hypercube $H_n$ are:

$n$

$n-2$ w/multiplicity $\binom{n}{1} = n$

$n-4$ w/multiplicity $\binom{n}{2}$

$\vdots$

$-n+2$

$-n$

## Motivations/applications for positive semi-definite matrices:

Quadratic forms and local optima

Cholesky factorization

$A = M^T M$

Ellipsoids  condition # = eccentricity

inner products

Convex set and semi-definite programming

Iterative solvers for $Ax=b$

- gradient descent