EE16A - Lecture 10 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

Circuit Properties

- Voltage: Energy spent per unit charge to move some charge from point A to point B.

$$V_{volts} = \frac{E_{joules}}{Q_{coulombs}} \tag{1}$$

- Current: The change in charge per unit time

$$I_{amps} = \frac{dQ}{dt_{sec}} \tag{2}$$

- Resistor: Conductor that converts energy to heat (dissipates energy). Therefore, it causes current to always flow into the plus side of the voltage across the resistor.

Kirchoff's Voltage Law (KVL):
The directed sum of potential differences (V) in a closed circuit must = 0

Kirchoff's Current Law (KCL):
The amount of current entering a point must = the amount of current exiting that point

Ohm's Law:
The amount of current entering a point must = the amount of current exiting that point

$$V_{volts} = \frac{I_{Amps}}{R_{Ohms}}$$

Resistance:
Resistance = Resistivity * Length/Cross-sectional-area

$$R = \rho \cdot \frac{L_{cm}}{A_{cm^2}}$$

2D Touch Screen

- Need Voltage (V), Current (I), Resistance (R)
- Calculate resistance of full touch screen $R = \rho \cdot \frac{L}{A}$)
- Get Current: $I = \frac{V}{R}$
- Voltage Divider: $V = \frac{L_1}{L_2}V$

EE16A - Lecture 12 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

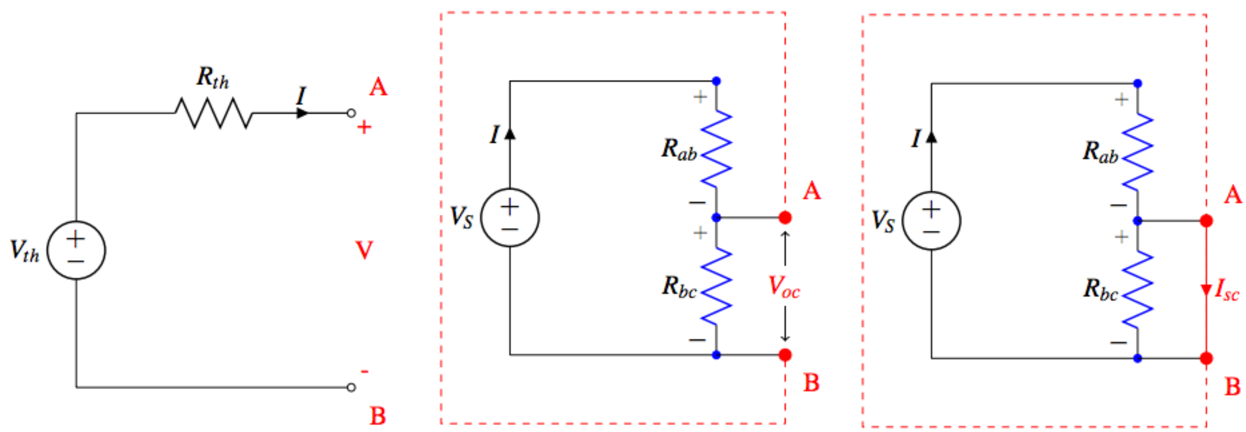
Power

- Energy disappated over time: $P = \frac{Energy}{Time}$

Power Equations:

$$P = VI = I^2R = \frac{V^2}{R} \tag{1}$$

Thevenin Equivalence



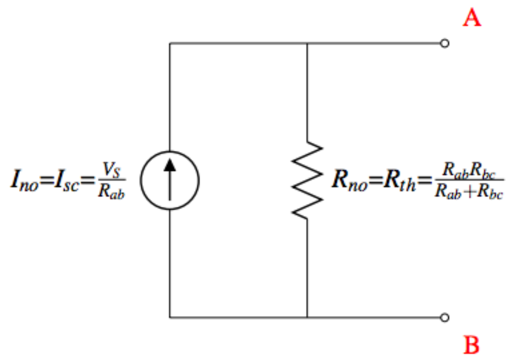
- Open Circuit: Find voltage drop between two nodes in complete (normal) circuit (V_{oc}). No current is flowing (no voltage drop over resistor).
- Short Circuit: Find current between two nodes, skip all voltage drops/resistors/subcircuits between them (I_{sc})

Thevenin Equations:

$$V_{th} = V_{oc} \tag{2}$$

$$R_{th} = \frac{V_{th}}{I_{sc}} \tag{3}$$

Norton Equivalence



- Open Circuit: Find voltage drop between two nodes in complete (normal) circuit $V_{oc} = I_{no} \cdot R_{no}$
- Short Circuit: Find current between two nodes, skip all voltage drops/resistors/subcircuits between them (I_{sc})

Norton Equations:

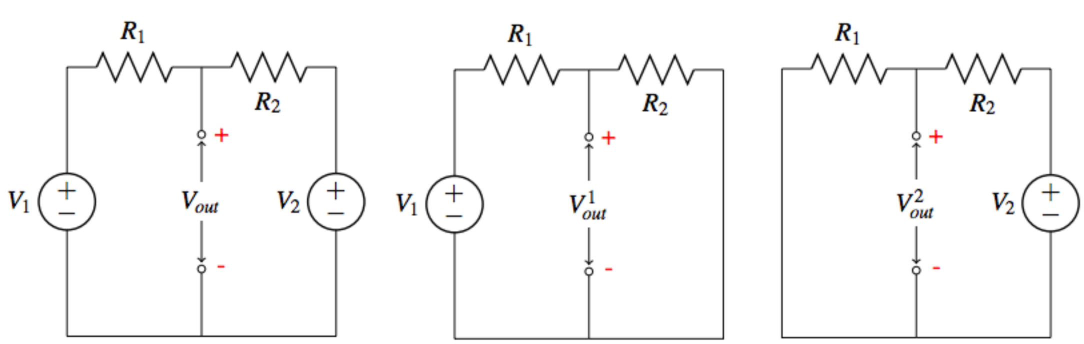
$$I_{no} = I_{sc} = \frac{V_S}{R_{ab}} \tag{4}$$

$$R_{no} = \frac{V_{oc}}{I_{no}} = \frac{R_{ab}R_{bc}}{R_{ab}+R_{bc}} = R_{th} = \frac{V_{th}}{I_{th}} \tag{5}$$

Thevenin/Norton Summary:

- Thevenin/Norton circuits have the same relationship between Voltage, V, in the circuit and the Current, I, coming out of it as the original circuits do
- They do not say anything about inside the actual circuit
- Ex. Power consumed by original circuit does not equal the power consumed by the T/N circuit (Power is non-linear)

Superposition



1. For each source (V_{src}, or I_{src}) k:
   a. Set all other sources to 0
      - V_{src} to short circuit (no voltage drop/increase)
      - I_{src} to open circuit (no current)
   b. Compute V_{out} due to source k
2. Sum up all V_{out_k}'s for all k sources

EE16A - Lecture 13 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

2D Resistance Touch Screens

- Measure y-pos:
  - Conductive strips on top plate (top: E_1, bottom: E_2)
  - Fig1. Connect Voltage source between E_1, E_2
  - Get voltage between E_2, E_4 (open circuit) V_{out} over the second resistor of the top plate ($V_{out} = V_2$)
  - Use V_out in a voltage divider equation to get the x position
- Measure y-pos:
  - Conductive strips on top plate (left: E_3, bottom: E_4)
  - Fig2. Connect Voltage source between E_3, E_4
  - Get voltage between E_2, E_4 (open circuit) V_{out} over the second resistor of the bottom plate ($V_{out} = V_4$)
  - Use V_out in a voltage divider equation to get the x position

Passive Sign Convention:

- Current must flow into the positive side of a voltage drop

Capacitors

- 2 pieces of conductive material (can have current/resistance)
- Separated by non-conductive substance with permittivity, ε
- Applied voltage builds up positive charge on one piece and negative charge on the other
- Capacitance: relationship between charge stored and voltage across it (measure in Farads (coulombs/volt))
- Once $Q = CV$ on surface of plate, current stops flowing

Parallel Capacitance

- Sum Capacitances (same as series resistors)
- $C_{par} = C_1 + C_2 + ... + C_n$

Series Capacitance

- Inverse of sum of inverses: (same as parallel resistors)
- $C_{ser} = (C_1 C_2 \cdots C_n)/(C_1 + C_2 + ... + C_n) = C_1 || C_2$

Capacitance Equations:

- d = distance between parallel plates
- ε = permittivity of substance between plates
- A = cross sectional area of plate

$$C = \frac{Q}{V} = \frac{\varepsilon \cdot A}{d} \tag{1}$$

$$I = C \cdot \frac{dV}{dt} \tag{2}$$

$$E = \frac{1}{2}CV^2 \tag{3}$$

$$C_{par} = C_1 + C_2 + ... + C_n \tag{4}$$

$$C_{ser} = C_1||C_2 = \frac{C_1C_2}{C_1 + C_2} \tag{5}$$

EE16A - Lecture 14 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

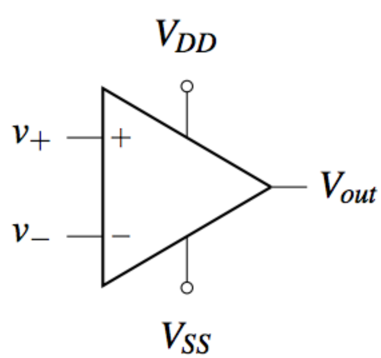
Switch Diagrams

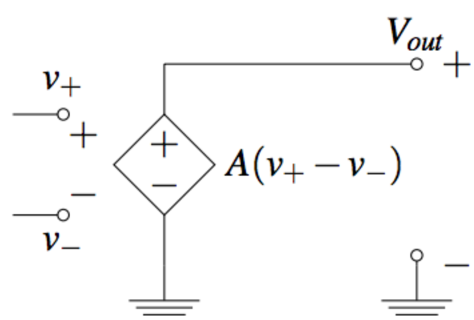
- Analyze each phase of the diagram with one switched as ideal wires and off switches as open circuits
- Capacitors have no resistance in the scope of this class
- Total charge is conserved between switch phases

Charge Sharing

- Phase 1: Accumulate charge on variable capacitor
- Phase 2: Distribute charge between variable and fixed capacitor
  a. Calculate total charge in Phase 1 (C_{fix} should be 0, C_{var} should be Q)
  b. Calculate total charge in Phase 2 in terms of V_{out} ($Q_{fix} = C_{fix}V_{out}$, $Q_{var} = C_{var}V_{out}$)
  c. Equate two charges and solve for V_{out}
- Any resistor in a capacitor circuit will discharge capacitors

OpAmp





Example at (1:05:28)

- Equivalent to dependent voltage source that measures voltage between V^+ and V^- (open circuit), and produces a voltage $V_{out} = A(V^+ - V^-)$
- Needs to be connected to 2 power sources ($V_{PP} = V_{DD}, V_{NN} = V_{SS}$)
- A is ideally an infinite scale factor, which would theoretically cause infinite V_{out}, but V_{out} is restricted to the range of ($V_{SS} = V_{NN} < V_{out} < V_{DD} = V_{PP}$)

OpAmp Equations and Properties:

- 5 terminals: $V^+, V^-, V_{out}, V_{DD}, V_{SS}$

$$V_{out} = A(V^+ - V^-) \tag{1}$$

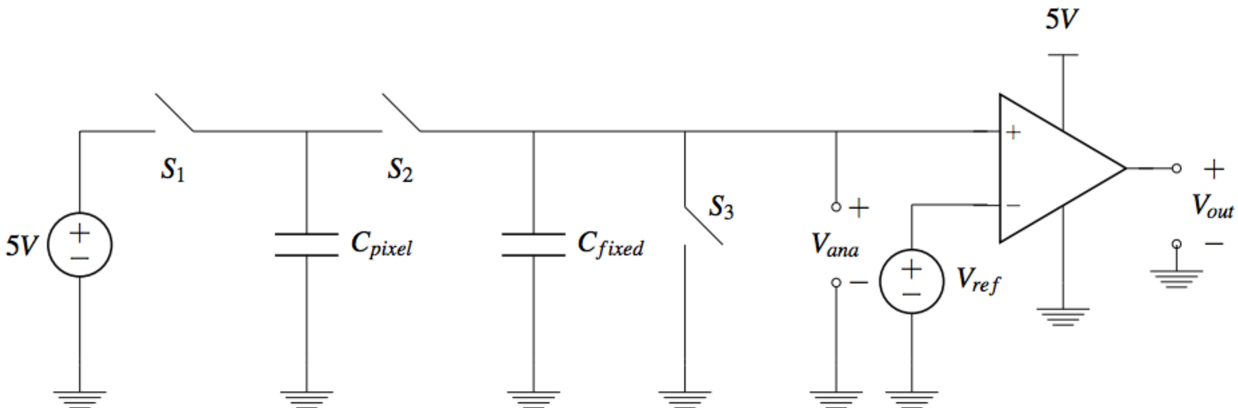
$$V_{SS} < V_{out} < V_{DD} \tag{2}$$

Capacitor Touch Screen

Variable Capacitance:

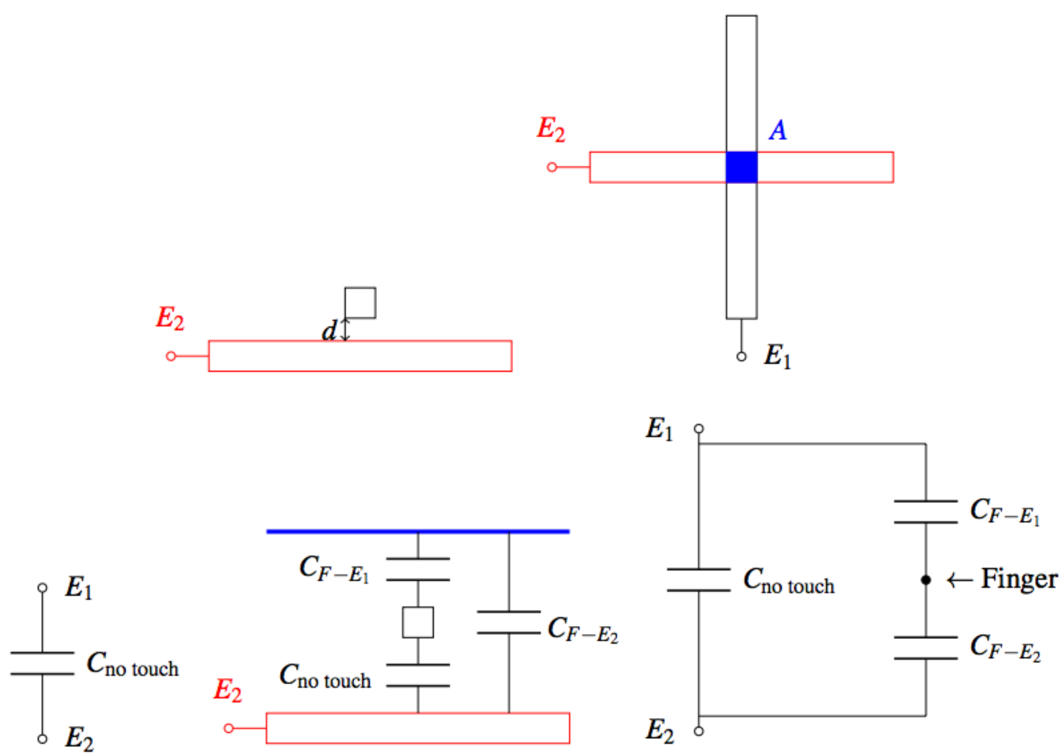
- Determines if finger has touched
  a. Charge variable capacitor to a fixed voltage
  b. Because $Q = CV$, capacitance varies, and voltage is fixed, charge Q now varies.
  c. Move varying Q onto another fixed capacitor (parallel)
  d. Because $V = \frac{Q}{C}$, Q varies, and C is fixed, this is the varying voltage

Using an OpAmp



- Purpose:
  a. Takes analog voltage created by switch/capacitor system, and force it to be either $0V$ or $5V$
  b. Uses an open circuit with infinite resistance (no voltage drop/current)
- Want V_{out} to be V_{max} on touch, and 0 on no touch

Detecting Finger Touch



- Pixel is intersection of capacitors
- Connect charge sharing parallel capacitors to OpAmp:
  - $V_{var} = V_{ana} = V^{+}$, $V_{ref} = V^{-}, V_{DD} = V_{max}$, V_{SS} = ground = 0
  - Choose V_{ref} and C_{fix} s.t. $V_{ana} > V_{ref}$ on touch, and $V_{ana} < V_{ref}$ when not touching
    * OpAmp forces V_{DD} and V_{SS}
  - $C_{notouch}$ is $\frac{\varepsilon A}{d}$: 1 capacitor between (E_1, E_2)
  - C_{touch} is $C_{notouch} + (C_{fin}, E_1)||(C_{fin}, E_2)$: finger adds two more capacitors in parallel

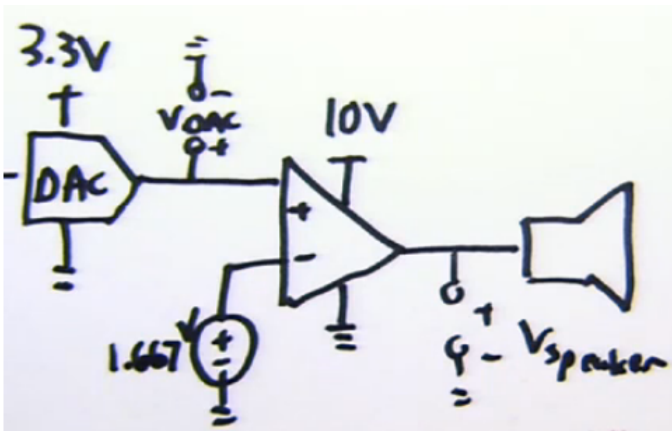
$$\boxed{\begin{aligned} C_{notouch} &= \frac{\varepsilon A}{d} \quad (3) \\ C_{touch} &= C_{notouch} + (C_{fin}, E_1)||(C_{fin}, E_2) \quad (4) \end{aligned}}$$

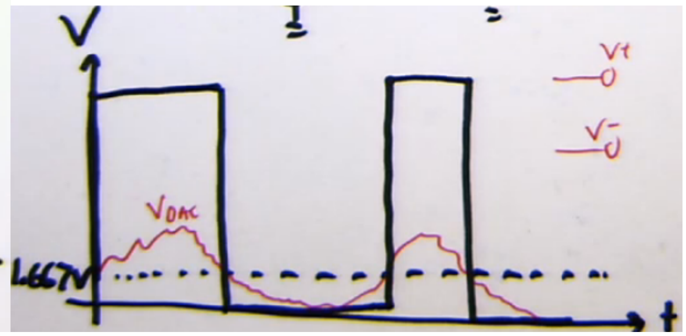
EE16A - Lecture 15 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

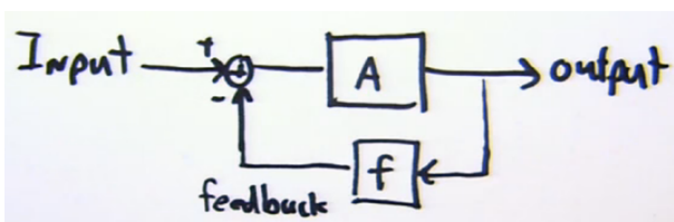
Scaling OpAmps





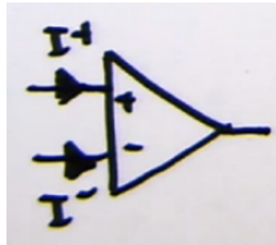
- Example at 46:53
- DAC: $V_{max} = 3.3V$, $V_{min} = 0$
- Speakers can be modeled with an 8Ω resistor to ground
- Need OpAmp to scale V_{DAC} to the range of the speaker without creating a blocking pattern (shown above)

Negative Feedback Loop



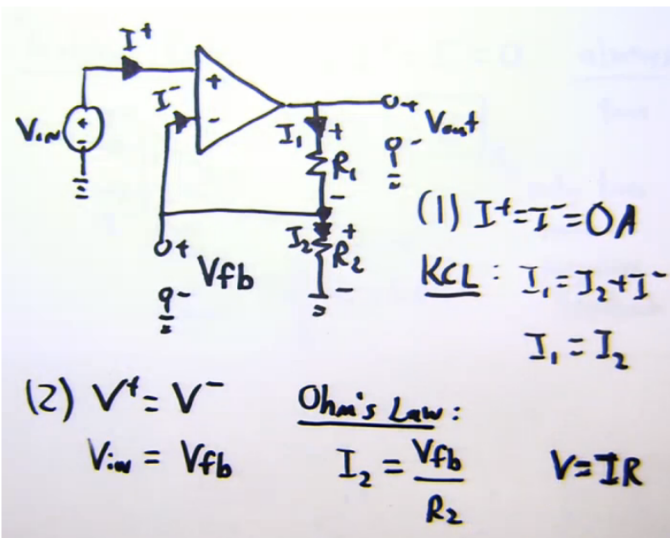
- Compare input against scaled feedback (from output)
- Feed result into a linear gain (A) to produce the output
- Send output into scaled feedback loop to compare against input
- If feedback > input, error is negative, output is pushed down, which causes the feedback to go down... etc.
- If feedback < input, error is positive, output is pushed up, which causes the feedback to go up... etc.
- Big Picture: Feedback will approach the input

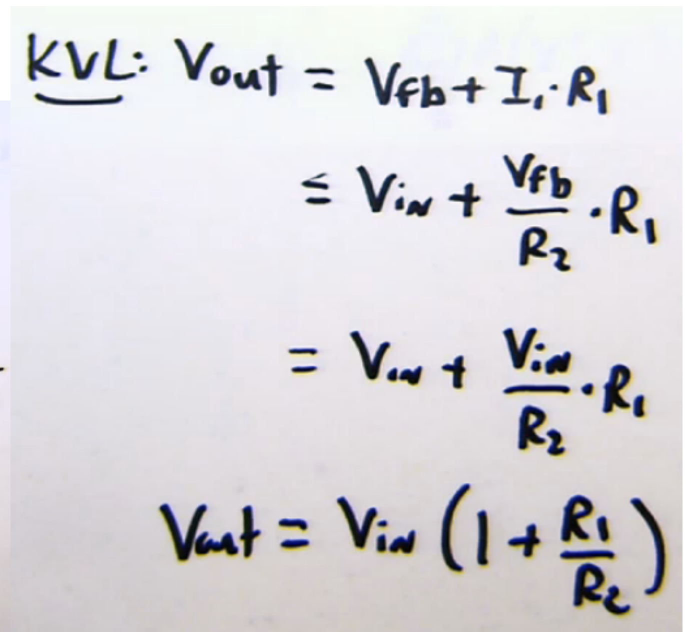
Golden Rules



1. $I^+ = I^- = 0$ (always true for ideal op amps)
2. $V^+ = V^-$ (only true with negative feedback)

OpAmp Negative Feedback Loop Analysis





- Input voltage V_{in} connected to OpAmp, with V_{out}
- Negative Feedback loop with R_1 connected to ground through parallel R_2 in and V_{fb} between V_{out} and V^-
- Because V^- is an open circuit: $I_1 = I_2$
- GR 1. $I^+ = I^- = 0$: KCL at node between R_1, R_2: $I_1 = I_2$ (because $I^- = 0$)
- GR 2. $V^+ = V^-$: $V_{in} = V_{fb}$
- Ohm's Law on R_2 because $V_2 = V_{fb}$: $I_2 = \frac{V_{fb}}{R_2} = \frac{V_{in}}{R_2} = I_1$
- KVL of path from ground to V_{out} through R_1 : $V_{out} = V_{fb} + I_1 R_1 = V_{in} + \frac{V_{in} R_1}{R_2} = V_{in}(1 + \frac{R_1}{R_2})$

OpAmp Negative Feedback Loop Equations

$$I^+ = I^- = 0 \tag{1}$$

$$V^+ = V^- \tag{2}$$

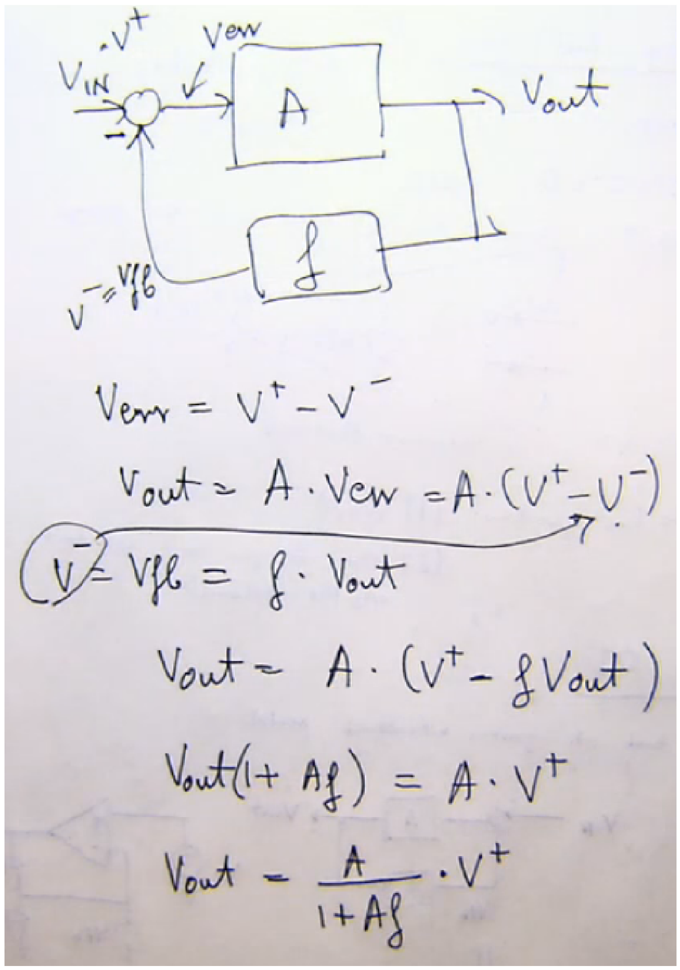
$$V_{out} = V_{in}(1 + \frac{R_1}{R_2}) \tag{3}$$

EE16A - Lecture 16 Notes

Name: Felix Su   SID: 25794773

Spring 2016   GSI: Ena Hariyoshi

Negative Feedback Loop



Usually V^- is negative:

- $V_{err} = V_{in} - V_{fb}$
- V_{fb} unchanged, $A > 0$, $f > 0$: V_{in} increases $\Rightarrow$ V_{err} increases $\Rightarrow$ V_{out} increases $\Rightarrow$ V_{fb} increases $\Rightarrow$ V_{err} decreases... V_{in} stabilizes
- Eq 1. $V_{out} = \frac{A}{1+Af} \cdot V^+$: Derivation at 27:34
  - Use resistors to determine f, which in turn determines the scaling factor of $\frac{V_{out}}{V_{in}}$
  - V_{out} scales V_{in} by a factor of $\frac{A}{1+Af}$
  - $V^- = V_{fb} = fV_{out} = \frac{fA}{1+Af} \cdot V^+$ (approaches 1 as $A \to \infty$)
- V_{out} approaches $\frac{1}{f} \cdot V^+$ as $A \to \infty$: Amps does not have to be very precise, just very large

- **Changes to V_{in} should cause V_{err} to decrease (V_{fb} increases if V_{in} increases and decreases if V_{in} decreases)**

NFL Equations

$$V_{out} = \frac{A}{1+Af} \cdot V^{+}; \lim_{A\to\infty} V_{out} = \frac{1}{f} \cdot V^{+} \tag{1}$$

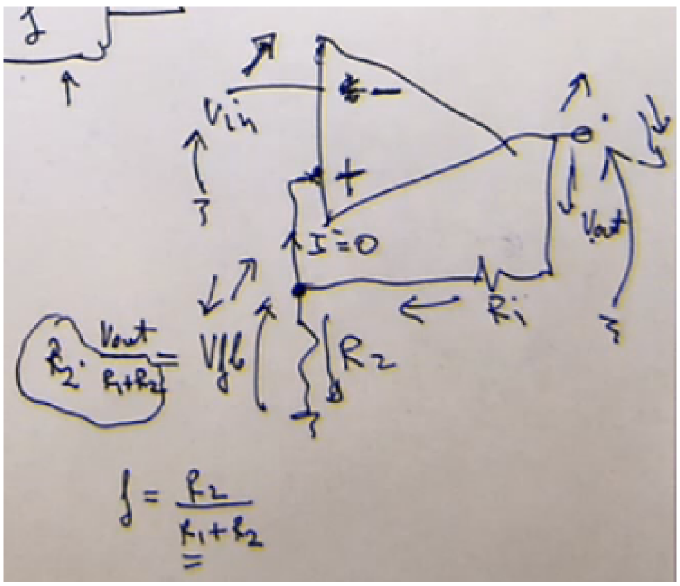
$$V^{-} = \frac{fA}{1+Af} \cdot V^{+}; \lim_{A\to\infty} V^{-} = V^{+} \tag{2}$$

Positive Feedback Loop

Usually V^{-} is positive:

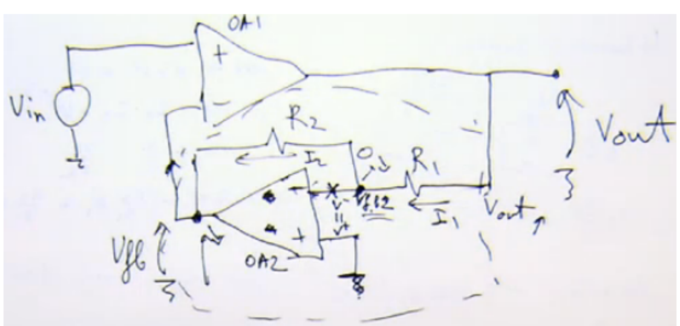
- V_{in} continuously increases (V^{+} is positive): V_{out} hits max rail
- V_{in} continuously decreases (V^{-} is negative): V_{out} hits min rail

Determining if Negative Feedback Exists



1. GR 1. $I^{-} = 0$, so analyze circuit between V_{out} and V_{fb}
2. $V_{fb} = \frac{R_2}{R_1 R_2} V_{out}$: $f = \frac{R_2}{R_1 R_2}$
3. Negative feedback exists

Analyzing complex OpAmp Circuits



- Use GR 1 to locate open circuits ($I = 0$)

- Split OpAmps by open circuits and V_{out}
- Determine if negative feedback loop exists (If positive feedback loop exists, V_{out} will hit the rails)
- Apply GR 2 with KCL to determine current flows
- Analyze individual voltage drops across each resistor from V_{in} to V_{fb} to V_{out} to get V_{out}

Example OA_2 : Inverting Amplifier

- Inverts the sign and is determined by the ratio of the two resistors
- $V_{fb} = -\frac{R_2}{R_1} \cdot V_{out}$

Find Polarity of OpAmp

- Polarity of inverse amplifier should be the opposite sign of the disturbance (If V^+ is inserted as positive, the polarity of OpAmp must be negative to generate a proper negative feedback)

EE16A - Lecture 17 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

Check Negative Feedback

Inverting Amplifier

1. Set any input to 0 (Replace all voltage sources with short circuits, all current sources with open circuits)
2. Dink the output
   - Check how V_err and V_in are changing (increase/decrease)
   - Positive feedback loop: dink causes V_out to slam into a rail
   - Negative Feedback Loop: dink minimized V_err and causes V_out to track V_in
3. Allows you to apply GR 2: $V^+ = V^-$

Check Negative Feedback

Design Example 1 Using only R's & op-amps & voltage sources, implement a current source whose value is proportional to a central voltage

1. State the Goal: Voltage Controlled Current Source (VCCS)
   - Control Voltage (V_c) to create a fixed current ($g_m V_c$ (g_m = Amps/V))
   - Value of current source depends on control voltage
   - 4 terminals (2 terminal for voltage, 2 terminals to create current at the output)
   - 2 separate circuits, but no current between the two circuits
2. Describe a Strategy: Block Diagram
   - Need relationship between voltage and current (Ohm's Law)
   - Measure voltage → pass through resister → measure output current
3. Implement the strategy: Voltage Source and Resistor
   - Simplest circuit that satisfies the criteria of the strategy and goal: Voltage source and resistor
     - Only 2 nodes (1 voltage source) - need 4
     - Create additional loop after resistor (at 0 volts to get $I = \frac{V}{R}$ and not allow current to flow into it: OpAmp)
   - OpAmp
     - Connect the circuit after the R to one of the inputs of the OpAmp to divert the current away from ground, but maintain $0I$ and $0V$ (O^+)
     - Connect the other input to ground
     - Connect the OpAp output terminal as the remaining (O^-)
4. Verify That the Strategy Works
   - If you dink the output O^-, the input from the original circuit will always increase, which means it must be V^- and the other has to be V^+ to maintain negative feedback.

- GR1: $I^+ = I^- = 0$; GR 2:$V_c = V_R$ (KVL)
- $I = \frac{V_R}{R} = \frac{V_c}{R}$, By design, I flows through output terminals in original goal ($I = \frac{V_c}{R} = g_m V_c$)
- V_c is not open circuit, like goal wants, so add a buffer (opamp) between voltage source and rest of the circuit

EE16A - Lecture 19 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

Positioning (Acoustic Locationing)

- Need $d+1$ known points/distances to determine position in d dimension
- 2D - intersection of 3 circles

Keep track of Time

- Determine distance from time it takes signal to arrive at receiver divided by the speed of the signal
- Need reference time t_o
- Synchronized clocks between receiver and beacon

Time Sampling

- Continuous signal is a function $a(t)\mathbb{R} \mapsto \mathbb{R}$
- Sample the signal every T seconds (sampling period (secs))
- Sample frequency $\frac{1T}{H}z$
- This defines a new **discrete time signal** that is defined only the integers: $a(t), \mathbb{Z} \mapsto \mathbb{R}$
- $a_[n] = a_{CT}(nT_s)$ discrete to continuous
- Compute delay between receiver and beacon in terms of samples and map make to continuous time using the time period (inverse of frequency)
- $y'[n] = y[n-k]$ implies that y' shifts y to the right by k discrete time samples (k sample delay)

Signal Delay Linear Algebra:

- Stack all values of discrete time signals into a vector
- Account for delay by cyclicly shifting down
- Use a Circulant Matrix to do the cyclic shift
- Example:
  - Let $y =$ signal received, $a, b, c =$ signal from beacons a,b,c
  - Let S^{N_k} Circulant Matrices that represent delay from beacon k to receiver
  - Each signal has an attenuation coeff. (not in scope of class)
  - $y = \alpha S^{N_A} a + \beta S^{N_B} b + \gamma S^{N_C} c$
  - Find $N_A, N_B, N_C \rightarrow$ delay $_A$, delay $_B$, delay $_C \rightarrow$ distance $_A$, distance $_B$, distance $_C$
  - Want all $S^{N_k} k$ to be nearly mutually orthogonal
  - Orthogonal vectors have dot product of 0

Orthogonality

- To measure 'similarity' of vectors, use dot products $\rightarrow$ inner product
- Inner product: $\langle \vec{x}, \vec{y} \rangle = x^T y^*$
- y^*: complex conjugate of y
- Complex conjugate of $a + ib = a - ib$ where $i = \sqrt{-1}$
- Complex values not in scope of class, so inner product = dot product for non-complex vectors
- Dot product: $x \cdot y = x^T y$

EE16A - Lecture 20 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

Positioning

Time of Flight

- Beacon sends signal of certain shape with signal a(t) at time t
- Receiver gets signal of same general shape y(t) some time later
- ToF = time it takes from signal sent to signal received
- Derive distance using $d = v\Delta t$
- Convert continuous time signal to a discrete time signal (with sampling period of $T_s = sec/sample$) where $a_{DT}[n] = a_{CT}(NT_s)$ and the sampling frequency is $f_s = \frac{1}{T_s} = Hz(cycles/sec)$

Discrete Time Delay

> 1. To find N_a, look at all possible shifts $S^k a$ where $k = 0, 1, \cdots, N-1$ and find the value of k that gives the closest match between received signal and very shifted version of original signal a
>   - compare y w/ $a, Sa, S^2a, \cdots, S^{N-1}a$
> 2. Compute k that maximizes the absolute value of the inner product between y and $S^k a$ (Max($|\langle y, S^k a, |\rangle$)
> 3. $k = N_a$ or the delay shift between the received and original signal

- Receive delayed version of original signal $a[n]$ (discrete time): $y[n] = \alpha a[n - N_a]$ where delay is N_a
- Delay is N_a samples so get delay time Δt by multiplying by period T_s ($\Delta t = N_a T_s$)
- It follows that the distance: $d = v\Delta t = vN_aT_s$
- Do not know N_a, so look at all possible shifts $S^k a$ where $k = 0, 1, \cdots, N-1$ and find the value of k that gives the closest match between received signal and very shifted version of original signal a
  - y compare w/ $a, Sa, S^2a, \cdots, S^{N-1}a$
  - Compute k that maximizes the absolute value of the inner product between y and $S^k a$ (Max($|\langle y, S^k a, |\rangle$)
  - $k = N_a$ or the delay shift between the received and original signal (not perfect, but close)

Periodic Signals

- Beacon signal a is periodic w/ period N (N-periodic) ($a[n - N] = a[n], \forall n \in \mathbb{Z}$)
- If $y[n] = \alpha a[n - N_A]$, y is also $N-periodic$ because shifting does not change periodicity (circular shift)

Circular Shift:

$$a[n] = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}, a[n-1] = \begin{bmatrix} a_2 \\ a_0 \\ a_1 \end{bmatrix} \tag{1}$$

Shift Matrix Example:

$$S = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, a[n-1] = Sa[n] \tag{2}$$

Shift Matrix General:

$$\begin{bmatrix} a_{N-1} \\ a_0 \\ \vdots \\ a_{N-2} \end{bmatrix} = \begin{bmatrix} 0_{N-1}^T & 1 \\ I_{N-1} & 0_{N-1} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{N-1} \end{bmatrix} \tag{3}$$

Model Delay of Signal a by Shift N_a

$$y = S^{N_A} a \tag{4}$$

Circular Shift of N-periodic signl by N samples $\implies I$

$$S^N = I \tag{5}$$

Shift Transpose is its Inverse

$$S^T = S^{-1} \tag{6}$$

Inner Product

> 1. Scope of the class: Inner products are in $\mathbb{R}^n$, so $\langle x, y \rangle = x^T y = x \cdot y$
> 2. Inner product = dot product

- Elements $x, y, z \in \mathscr{V}$ where $\mathscr{V}$ = vector space in $\mathbb{R}^n$ or $\mathbb{C}^n$
- Inner product is a function that has a domain that is a pair of elements in vector space $\langle x, y \rangle : \mathscr{V} \times \mathscr{V}$ and maps these values from $\mathbb{R}^n$ or $\mathbb{C}^n$ to $\mathbb{R}$ or $\mathbb{C}$
- In $\mathbb{C}^n$, $\langle x, y \rangle = x^T y^*$
- In $\mathbb{R}^n$, $\langle x, y \rangle = x^T y = x \cdot y$

Inner Product Properties
Hermitian/Conjugate Symmetry (*=complex conjugate)

$$\langle x, y \rangle = \langle y, x^* \rangle \tag{7}$$

Symmetry/Commutativity (If vector space $\in \mathbb{R}$)

$$\langle x, y \rangle = \langle y, x \rangle \tag{8}$$

Distributive

$$\langle x, y + z \rangle = \langle x, y \rangle + \langle x, z \rangle \tag{9}$$

Scaling

$$\langle \alpha x, y \rangle = \alpha \langle x, y \rangle \tag{10}$$

Non-negativity

$$\langle x, x \rangle \geq 0 \text{ w/ equality } \iff x = 0 \tag{11}$$

Norm

> 1. 2-Norm Definition ($\|x\|_2$)
>   - $\|x\| = \sqrt{\langle x, x\rangle}$
>   - $\|x\|^2 = \langle x, x\rangle$
> 2. Norm and inner product: $\langle a, b\rangle = \|a\|\|b\| \cos\theta$ where $\theta =$ angle between a, b
> 3. For comparison between signals, find shift that will maximize: $\cos\theta = \frac{\langle a,b\rangle}{\|a\|\|b\|}$
> 4. Norm = magnitude

- Gives sense of size to elements in vector space
- Relationship with cosine of angle between vectors
- Vector a, b with angles w.r.t x-axis of α, β respectively
- Angle between the vectors is θ
- $a = [a_1, a_2], b = [b_1, b_2]$
- $a_1 = \|a\| \cos\alpha, a_2 = \|a\| \sin\alpha; b_1 = \|b\| \cos\beta, b_2 = \|b\| \sin\beta$
- $\langle a, b\rangle = [a_1, a_2][b_1, b_2]^T = a_1 b_1 + a_2 b_2 = \|a\| \cos\alpha \|b\| \cos\beta + \|a\| \sin\alpha \|b\| \sin\beta$
- $= \|a\|\|b\|(\cos\alpha\cos\beta + \sin\alpha\sin\beta)$
- $= \|a\|\|b\|(\cos(\beta - \alpha)$ because $(\cos(\beta - \alpha) = \cos\alpha\cos\beta + \sin\alpha\sin\beta)$
- $= \|a\|\|b\| \cos\theta$

Non-Negativity

$$\|x\| \geq 0 \text{ w/ equality } \iff x = 0 \tag{12}$$

Scaling Property

$$\|\alpha x\| = |\alpha|\|x\| \tag{13}$$

Triangle Inequality

$$\|x + y\| \leq \|x\| + \|y\| \text{ w/ equality } \iff x = \alpha y, \alpha > 0 \tag{14}$$

Norm and Inner Product:

$$\langle a, b\rangle = \|a\|\|b\| \cos\theta \text{ where } \theta = \text{ angle between } a, b \tag{15}$$

Angle between Cartesian Vectors

$$\cos\theta = \frac{\langle a, b\rangle}{\|a\|\|b\|} \tag{16}$$

EE16A - Lecture 21 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

Determine Distance from Beacon

1. Sent signal (periodic) = $a[n] = a[n+N], \forall n \in \mathbb{Z}$
2. Received signal (Time delayed) = $y[n] = \alpha a[n - N_A]$
3. Find time delay N_A by finding the maximum inner product of each shifted version of the original signal and the received signal ($max(|\langle S^k a, y\rangle|), k = 1, \cdots, N$) where resulting $k = N_A$
4. Determine distance (d_A) using time delay N_A and given speed of wave signal
5. **Note:** When doing inner products to get correct shifts $S^k, k =$ shift, you want everything to be as close as possible to mutually orthogonal with shifted version, so only the inner product of the original signal with itself will be large (peak) and everything else will be noticably smaller

- Beacon A sends periodic signal with period N ($a[n] = a[n+N], \forall n \in \mathbb{Z}$)
- Receive attenuated/timed delayed version of the sent signal
  - Goal: Figure out time delay and derive distance using the given speed of the wave being emitted
- Need 3 beacons to pinpoint one point on a 2D plane
- Find d_A, d_B, d_C from signals A, B, C which are all $N - periodic$
- Receiver gets some linear combination the 3 signals ($y[n] = \alpha a[n - N_A] + \beta b[n - N_B] + \gamma c[n - N_C] = \alpha S^{N_A} a + \beta S^{N_B} b + \gamma S^{N_C} c$)
- Linear waves: Superposition principles applies (Electromagnetic and acoustic waves are approx. linear)
- Determine $N_A, N_B, N_C \rightarrow d_A, d_B, d_C$, by using $max(\langle S^k a, y\rangle), k = 1, \cdots, N$

Vector Model of a Signal Period:

$$y = \begin{bmatrix} y[n] \\ \vdots \\ y[N-1] \end{bmatrix} \tag{1}$$

Triangulation

1. When subtracting vectors, arrow point towards "positive" value
2. **Minimizing Error**: Find x s.t. $\|\varepsilon\|^2$ is minimized where $\varepsilon = b - Ax$ (Result data we receive $Ax = b - \varepsilon$)
3. $\varepsilon \perp a_1$ & $\varepsilon \perp a_2$: ε has to be orthogonal to each of the columns of the A matrix

- Retrieve position after distances from each beacon received

- $\|a_n - x\|^2 = \langle a_n - x, a_n - x\rangle = \langle a_n, a_n\rangle - 2\langle a_n, x\rangle + \langle x, x\rangle = \|a\|^2 + 2\langle a_n, x\rangle + \|x\|^2 = d_n^2$
- For 3-beacon system, subtract d_2 and d_3 from d_1:
  * $2\langle a_2, x\rangle - 2\langle a_1, x\rangle = 2\langle a_2 - a_1, x\rangle = d_1^2 - d_2^2 + \|a_2\|^2 - \|a_1\|^2$
  * $2\langle a_3, x\rangle - 2\langle a_1, x\rangle = 2\langle a_3 - a_1, x\rangle = d_1^2 - d_3^2 + \|a_3\|^2 - \|a_1\|^2$
- Matrix multiplication and solve for b_1, b_2
  - $Ax = b$ has a unique soln. if $a_2^T - a_1^T$ and $a_3^T - a_1^T$ are lin. independent
  - As long as the three beacons are not co-linear, the above will be lin. ind.
  - Use more beacons to do pairwise subtractions and account for error. This will just make A an $n \times 2$ matrix, and we will get $b \in \mathbb{R}^n$
- **Problem:** $b \notin$ span of cols of A because there is error in the signals.
  - $Ax = b$ has no exact soln.
  - b is some vector outside the subspace of $x_1A_1 + x_2A_2$ (outside of the plane formed by the two vectors of A)
  - Error $\varepsilon = b - Ax$
  - Find x s.t. $\|\varepsilon\|^2$ is minimized
  - $Ax = a_1x_1 + a_2x_2$, which is a line between the two vectors (a_1, a_2).
  - Choose point on Ax that creates a perpendicular line from that point to b in order to get the smallest ε
  - ε is perpendicular to the plane of the subspace between a_1, a_2

Matrix Multiplication Step for Triangulation

$$\begin{bmatrix} 2(a_2^T - a_1^T) \\ 2(a_3^T - a_1^T) \end{bmatrix} x = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{2}$$

Basic Projection Problem

> 1. $\langle \vec{a}, \varepsilon = b - \vec{a}x\rangle = 0$, Solve for x
> 2. Best estimate for $\vec{b}$ in the least square error sense is $\hat{b} = \vec{a}x - \frac{\vec{a}^T\vec{b}}{\vec{a}^Ta}\vec{a}$

- Have one dimensional vector $\vec{a}$ and $\vec{b}$ outside of the subspace.
- Want to get value x on $\vec{a}$ s.t. $\varepsilon = b - \vec{a}x$ is minimized.
  - Optimal x makes ε orthogonal to $\vec{a}$
  - $\varepsilon \perp a \implies \langle \vec{a}, \varepsilon\rangle = 0 \implies \langle \vec{a}, b - \vec{a}x\rangle = 0$
- Solve for x:
  - $\langle \vec{a}, \vec{b}\rangle - \langle \vec{a}, \vec{a}x\rangle = 0$
  - $\langle \vec{a}, \vec{b}\rangle - x\langle \vec{a}, \vec{a}\rangle = 0$
  - $x = \frac{\langle \vec{a}, \vec{b}\rangle}{\langle \vec{a}, \vec{a}\rangle} = \frac{\vec{a}^T\vec{b}}{\vec{a}^Ta}$
- Best estimate for $\vec{b}$ in the least square error sense is $\hat{b} = \vec{a}x - \frac{\vec{a}^T\vec{b}}{\vec{a}^Ta}\vec{a}$
- Can derive Cauchy-Schwarz from $\|\varepsilon\| = \|\vec{b} - \hat{b}\|^2 \geq 0$

Cauchy-Schwarz Inequality

$$|\langle \vec{a}, \vec{b}\rangle| \leq \|\vec{a}\|\|\vec{b}\| \tag{3}$$

Multi-Dimension Projection Problem

> 1. $x = (A^T A)^{-1} A^T b$

- $A_{n\times m} x_{m\times 1} = b_{n\times 1}$
- Let $A = \begin{bmatrix} a_1 & a_2 \end{bmatrix}$
- Optimal x makes ε orthogonal to each a_n (column vec) of A
  - $\varepsilon \perp a_n \implies \langle \vec{a_n}, \varepsilon \rangle = \langle \vec{a_n}, b - Ax \rangle = \vec{a_n}^T (\vec{b} - Ax) = 0$
- Solve for x: $A^T(\vec{b} - Ax) = 0$
  - $A^T(\vec{b} - Ax) = 0 \implies A^T b - A^T A x = 0 \implies A^T A x = A^T b$
  - $A^T A$ is invertible iff A has full column rank (lin. indep. columns)
  - $x = (A^T A)^{-1} A^T b$

Solve for x:

$$\begin{bmatrix} \vec{a_1}^T \\ \vdots \\ \vec{a_n}^T \end{bmatrix} (\vec{b} - Ax) = 0 \tag{4}$$

EE16A - Lecture 22 Notes

Name: Felix Su   SID: 25794773

Spring 2016   GSI: Ena Hariyoshi

Least Squares Geometric Argument

> 1. Best estimate (1D): $\hat{b} = \frac{\vec{a}^T\vec{b}}{\vec{a}^T\vec{a}}\vec{a}$ where $\frac{\vec{a}^T\vec{b}}{\vec{a}^T\vec{a}}$= best scalar x
> 2. Best estimate (Multi-D): $x = (A^TA)^{-1}A^Tb$
> 3. Error Vector: $\varepsilon = \vec{b} - A\vec{x}$ because $\vec{b} \notin$ span$\{\vec{a_1}, \ldots, \vec{a_n}\}$
> 4. Goal: Minimize $\|\varepsilon\|^2$ over all possible $\vec{x}$
> 5. Best $\vec{x}$ ensures that $\varepsilon \perp \sum_{i=1}^{n} \alpha_i\vec{a}_i \forall \alpha_i \in \mathbb{R} \implies \varepsilon \perp a_n$ where a_n = each column vector of A
> 6. $(A^TA)^{-1}$ exists iff A has full col. rank (if A^TA is invertible, can't solve), $n >> m \implies$ this will happen.

- Dim(A)=$n \times m$, Dim(x)=$m \times 1$, Dim(b)=$n \times 1$
  - Generally $n > m$
- **Claim:** A^TA is invertible iff A has linearly independent columns.
- **Proof:** Show Null(A^TA) = Null(A)
  - A^TA guaranteed to be square $m \times m$
  - Forward: $\vec{q} \in$ Null(A), $\exists \vec{q} \neq 0 \implies \vec{q} \in$ Null(A^TA)
    - Because $A\vec{q} = 0 \implies A^TA\vec{q} = A^T0 = 0$
  - Backward (Converse): $\vec{r} \in$ Null(A^TA), $\exists \vec{r} \neq 0 \implies \vec{r} \in$ Null(A)
    - Because $A^TA\vec{r} = 0 \implies \vec{r}^TA^TA\vec{r} = \langle A\vec{r}, A\vec{r}\rangle = \|A\vec{r}\|^2 = \vec{r}^T0 = 0$ (**Transpose of a dot product reverses the order of the operands in the product**)
    - $\|A\vec{r}\|^2 = 0 \implies A\vec{r} = 0 \implies \vec{r} \in$ Null(A)
  - Proves Null(A^TA) and Null(A) have the same null space.
- Only vector in null space of Null(A^TA) is $\vec{0} \implies$ columns of A are linearly independent.

Inverse Formula for 2×2 Matrix:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \tag{1}$$

Line of Best Fit

- Example: $t_1 = 0, t_2 = 1, t_2 = 2$ and $y(t) = -1, y(t) = -3, y(t) = 1$, Find $y(t) = x_1t + x_2$
- Set up $A\vec{x} = y(\vec{t_n})$
- Solve for A^TA

Line of Best Fit Example

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} y(t_1) = -1 \\ y(t_2) = -3 \\ y(t_3) = 1 \end{bmatrix} \tag{2}$$

Find $A^T A$

$$\begin{bmatrix} 0 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 1 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 5 & 3 \\ 3 & 3 \end{bmatrix} \tag{3}$$

Find $(A^T A)^{-1}$

$$\frac{1}{6} \begin{bmatrix} 3 & -3 \\ -3 & 5 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{5}{6} \end{bmatrix} \tag{4}$$

Find $x = (A^T A)^{-1} A^T b$

$$\begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{5}{6} \end{bmatrix} \begin{bmatrix} 0 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -1 \\ -3 \\ 1 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & 0 & \frac{1}{2} \\ \frac{5}{6} & \frac{2}{3} & \frac{1}{6} \end{bmatrix} \begin{bmatrix} -1 \\ -3 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \end{bmatrix} \tag{5}$$

Plug in to solve for $y(t)$

$$y(t) = x_1 t + x_2 = t - 2 \tag{6}$$

Find $\varepsilon = \vec{b} - A\vec{x}$

$$\varepsilon = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} \tag{7}$$

Determine $\|\varepsilon\|^2$

$$\|\varepsilon\|^2 = 1^2 + (-2)^2 + 1^2 = 6 \tag{8}$$

Least Squares Calculus Argument

> 1. Smallest $\|\varepsilon\|^2$ is the sum of the squares of the distance between each actual measurement (point) and the line of best fit.

Least Squares Calculus:

$$\|\varepsilon\|^2 = \sum_{i=1}^{n(=3)} \varepsilon^2 = \sum_{i=1}^{n(=3)} (x_1 t_i + x_2 - y(t_i))^2 = (x_2 + 1)^2 + (x_1 + x_2 + 3)^2 + (2x_1 + x_2 - 1)^2 \tag{9}$$

Find minimum w.r.t x_1:

$$\frac{\partial \|\varepsilon\|}{\partial x_1} = 0 + 2(x_1 + x_2 + 3) + 4(2x_1 + x_2 - 1) = 0 \implies 10x_1 + 6x_2 = -2 \tag{10}$$

Find minimum w.r.t x_2:

$$\frac{\partial \|\varepsilon\|}{\partial x_2} = 2(x_2 + 1) + 2(x_1 + x_2 + 3) + 2(2x_1 + x_2 - 1) = 0 \implies x_1 + x_2 = -1 \tag{11}$$

Solve for x_1, x_2:

$$x_1 = -x_2 - 1, 10(-x_2 - 1) + 6x_2 = -2, x_2 = -2, x_1 = 1 \tag{12}$$

EE16A - Lecture 23 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

Matching Pursuit Setup

1. Received signal $y = \sum_k \alpha_k S^{N_k} \vec{z}_k$ is a linear combination of delayed versions of the signals sent z_k where S^{N_k} is the circular shift matrix
2. Sparse Representation: The linear combination (received signal) of the delayed messages contains very few terms relative to the max number of broadcasted signals.

- Scenario: A lot of players are trying to communicate with you, but you can only receive a few at the time.
- Say you have$L = 2000$ players (trying to communicate), each sends a message $\vec{z}_k \in \mathbb{R}^{N=400}$
- Usually $L > N$ and typically $L >> N$
- Received vector $y = \sum_k \alpha_k S^{N_k} \vec{z}_k$, which is a linear combination of delayed versions of the signals sent z_k where S^{N_k} is the circular shift matrix.
- Sparse Representation: The linear combination (received signal) of the delayed messages contains very few terms relative to the 2000 possible terms.

Sparsity (Using Dictionary)

1. Use a larger, more specific dictionary to make sparser messages

- The larger the dictionary is, the more specific descriptions can be and the less words you need to use to convey a message.
- Dictionary contains all the circularly shifted forms of the messages, so it will be of size $L \times N$ (no. of msgs $\times$ no. of possible shifts = msg. vec length)
  - Called a **Redundant Dictionary**
  - $D = \{\vec{z}_1 S_{z_1} \dots \vec{z}_1 S_{z_1}^{N-1} | \vec{z}_2 S_{z_2} \dots \vec{z}_2 S_{z_2}^{N-1} | \dots \vec{z}_n S_{z_n}^{N-1}\}$
  - Assume: D is complete and includes N linearly independent vectors
  - Assume: Every vector is unit length, $\|\vec{z}_1\| = 1$
  - Cannot Assume: Orthogonality
- Now let $D = \{\phi_1, \dots, \phi_T\}, T =$ Dictionary size
- Need to figure out which subset of ϕ_k's is represented in the received signal y.
- Simplification: Assume no delay, so $y = \sum_k \alpha_k \vec{z}_k$ and $D = \{\phi_1 = \vec{z}_1, \dots, \phi_L = \vec{z}_L\}$
- Residuals will approach 0.

Matching Pursuit Algorithm

> 1. Take observation vector y as the initial residual vector $r^{[0]}$
> 2. Project each residual $r^{[m-1]}$ onto the space of each of the vectors in the dictionary D and pick the maximum projection. (Choose a $\phi_k \in D$ s.t. $\vec{v}_m = \text{ argmax}_k |\langle \vec{r}^{[m-1]}, \phi_k \rangle|$)
> 3. Write out equation for residual: $r^{[m-1]} = \langle \vec{r}^{[m-1]}, \vec{v}_m \rangle \vec{v}_m + \vec{r}^{[m]}$ where $r^{[m]} \perp \vec{v}_m$
> 4. Decompose the next residual ($r^{[m]}$)
> 5. Resulting signal at iteration M is $\vec{y} = \sum_{m=0}^{M-1} \langle r^{[m]}, \vec{v}_{m+1} \rangle \vec{v}_{m+1} + \vec{r}^{[M]}$

- A greedy algorithm: Step by step. At each step, search for a local optimum to reach a global optimum.
- Goal: Estimate $y \in \mathbb{R}^N$ by finding the main signal contributors
  - Let $M =$ no. of terms in the estimation (signal contributors)
  - Let $D = \{\phi_l\}, L = 1, \ldots, L$ where $L \geq M$ and $L > N (\phi_L \in \mathbb{R}^N)$
  - D is complete and $\|\phi_l\| = 1$
- At each step you have **Residue of** y=$\vec{r}^{[0]}$ (unaccounted portion)
- Step 0: $\vec{r}^{[0]} = y$
- Step 1: Choose a $\phi_k \in D$ s.t. $\vec{v}_1 = \text{ argmax}_k |\langle y, \phi_k \rangle|$
  - Choose ϕ_k that gives the largest projection of y onto ϕ_k
  - Write $y = \langle \vec{y}_1, \vec{v}_1, \vec{\rangle} v_1 + \vec{r}^{[1]}$ where $r^{[1]} \perp \vec{v}_1$
  - $\|\vec{y}\|^2 = \|\langle \vec{y}_1, \vec{v}_1, \vec{\rangle} v_1\|^2 + \|\vec{r}^{[1]}\|^2 = |\langle \vec{y}_1, \vec{v}_1, \rangle|^2 \|\vec{v}_1\|^2 + \|\vec{r}^{[1]}\|^2$
  - $\|\vec{y}\|^2 = |\langle \vec{y}_1, \vec{v}_1, \rangle|^2 + \|\vec{r}^{[1]}\|^2$ (becuase $\vec{v}_1$ has unit length)
- Step 2: Decompose $\vec{r}^{[1]}$: Find the vector in D that best represents $\vec{r}^{[1]}$
  - Choose a $\phi_k \in D$ s.t. $\vec{v}_2 = \text{ argmax}_k |\langle \vec{r}^{[1]}, \phi_k \rangle|$
  - $\vec{r}^{[1]} = \langle \vec{r}^{[1]}, \vec{v}_2 \rangle \vec{v}_2 + \vec{r}^{[2]}$ where $r^{[2]} \perp \vec{v}_2$
  - $r^{[2]}$ may not be (probably not) $\perp \vec{v}_1$
- Step m: Decompose $\vec{r}^{[m-1]}$
  - $\vec{v}_y^{[m-1]} = \langle \vec{r}^{[m-1]}, \vec{v}_m \rangle \vec{v}_m + \vec{r}^{[m]}$
  - To stop at iteration M, express $\vec{y} = \sum_{m=0}^{M-1} \langle r^{[m]}, \vec{v}_{m+1} \rangle \vec{v}_{m+1} + \vec{r}^{[M]}$
- Residuals approach 0
  - $\|\vec{y}\|^2 = \sum_{m=0}^{M-1} |\langle r_y^{[M]}, \vec{v}_{m+1} \rangle|^2 + \|\vec{r}_y^{[M]}\|^2$
  - Fixed term on left
  - Sum of positive terms on right plus risidual
  - As sum increases last residual must decrease for each step

Another View of the Matching Pursuit Algorithm

> 1. If A_m has Orthonormal Columns: $A_m^T A_m = I$

- Start with $y = r^{[m]}$ (observed measurement)
- Create a matrix A_m at each step that has the columns of the vectors in the dictionary you have obtained up to that point. (Matrix of principal directions up to this step)
- Start from $m = 1$: while $r^{[m]} \neq 0$, look for principal direction: $\vec{v}_m = \text{argmax}_k |\langle r^{[m]}, \vec{z}_k \rangle|$
  - argmax: Scan all indices k, on k for which the inner product is the largest, use that inner product and assign $\vec{v}_m = \vec{z}_k$ as the mth principal direction
- Augment matrix of principal directions: $A_m = \left[A_{m-1} | \vec{v}_m\right]$
- Project y onto sol. space of A_m: $\vec{y} = A_m \alpha_m + \vec{\varepsilon}$
- Approximate y in Least Squares sense
  - $\vec{y}_m = A_m \hat{\alpha}_m$ where $\hat{\alpha}_m$ is the LS soln to $A_m \alpha_m = \vec{y}$
  - $\hat{\alpha}_m = (A_m^T A_m)^{-1} A_m^T \vec{y}$
  - $\vec{y}_m = A_m (A_m^T A_m)^{-1} A_m^T \vec{y}$
  - $\vec{r}_m = \vec{y} - \vec{y}_m$
  - Problem: When columns of A_m, the principal directions $\vec{v}_k$ are not orthogonal, the LS soln, is very costly.
- Keep iterating forward: $m = m + 1$ until end.

EE16A - Lecture 24 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

Review

- Redundant (Overcomplete) Dictionary of vectors $D = \{\phi_l\}, \phi_l \in \mathbb{R}^N$
- N = length of each player's message
- $\text{span}\{\phi_l\} = \mathbb{R}^N$
- D contains no fewer than N lin. ind. vectors
- $y = \sum_{l=1}^{m-1} \alpha_l \vec{z}_l$
  - $\vec{z}_l$: message form the lth player
  - α_l: scalars
  - Simplified model: ignoring time delay S^{N_k}
  - of the $L = 2000$ players, very few are talking to us at any given time window
  - Assume: All $\|\phi_l\| = 1$, if not, normalize
- Use Matching Pursuit Algorithm to solve for y
  - Continuously match with the best vector in the dictionary
  - Minimize residual vector each time using local optimization

Another View of Matching Pursuit Algorithm

- Initialize:
  - Residual$=$ y at iteration 0: $r_0 = y$
  - $A_0 = [\quad]$, placeholder matrix for the matched distionary vectors
- While $r_m \geq \varepsilon$, continue (Here: $\varepsilon = 0$)
  1. Look for the vector that gives the best match: $k = \text{argmax}_i |\langle r_m, \vec{z}_i, \rangle|$
  2. k is the largest projection between the current residual and all the vectors in the dictionary D.
  3. $\vec{v}_m = \vec{z}_k$ is the vector in the dictionary the gives the projection k
  4. Augment matrix A with $\vec{v}_m$ as a new column: $A_m = \begin{bmatrix} A_{m-1} & \vec{v}_m \end{bmatrix}$
  5. Model received signal as a linear combination of the columns of the A matrix up to this point + error term:
     - $\vec{y} = A_m \vec{\alpha}_m + \vec{r}_m$
     - $A_1 \alpha_1$ is the best estimate at iteration 1 of the received signal $\vec{y}$: $A_m \alpha_M = \vec{y}_m$
     - Best α for answer is given by the Least Squares formula: $\alpha_m = (A_M^T A_m)^{-1} A_m^T \vec{y}$
  6. $\vec{r}_m = \vec{y} - \vec{y}_m = y - A_m \alpha_m$
- **Problem:**
  - Matrix A keeps getting wider on every iteration

- Eacch iteration you have to compute $(A_m^T A_m)^{-1}$, which is too costly

- **Solution:**
  - Try to construct matrix A at each iteration that has *orthonormal* columns
  - $\alpha_m = (A_M^T A_m)^{-1} A_m^T \vec{y}$
  - $\vec{y}_m = A_m (A_M^T A_m)^{-1} A_m^T \vec{y}$
  - If A_m has orthonormal cols, (if all $\vec{v}_k$'s are orthonormal): $A_m^T A_m = T$
  - Thus, $\vec{y}_m = A_m(A_m^T y) = \sum_{i=1}^{m} \langle \vec{v}_i, \vec{y} \rangle \vec{v}_i$

Gram-Schmidt Orthogonlization (Orthonormalization)

- Have $\vec{v}_1, \ldots, \vec{v}_m$ lin. ind. vectors $V_m = \text{span}\{\vec{v}_1, \ldots, \vec{v}_m\}$
- **Claim:** We can find another set of vectors $\vec{z}_1, \ldots, \vec{z}_m$ that are mutually orthogonal and for which $Z_m = \text{span}\{\vec{z}_1, \ldots, \vec{z}_m\} = \text{span}\{\vec{v}_1, \ldots, \vec{v}_m\} = V_m$
  - Once we have $\vec{z}_1, \ldots, \vec{z}_m$, we can construct vectors $\vec{q}_i = \frac{\vec{z}_i}{\|\vec{z}_i\|}, i = 1, \ldots, m$
  - $Q_m = \text{span}\{\vec{q}_1, \ldots, \vec{q}_m\} = V_m = Z_m$ where $\langle \vec{q}_l, \vec{q}_k \rangle = 1$ when $k = l$ and $\langle \vec{q}_l, \vec{q}_k \rangle = 0$ when $k \neq l$
  - All $\vec{q}_i$'s are orthonormal
- **Proof** (By construction):
  - $\vec{v}_1, \vec{v}_2, \vec{v}_3$ are linearly independent
  - Let $\vec{z}_1 = \vec{v}_1$: 1st principal direction of the orthogonal set
  - $\vec{z}_2 = \vec{v}_2 - \alpha \vec{z}_1$ s.t. $\vec{z}_2 \perp \vec{z}_1$
    * $\langle \vec{z}_2, \vec{z}_1 \rangle = \langle \vec{v}_2 - \alpha \vec{z}_1, \vec{z}_1 \rangle = 0$
    * $\langle \vec{v}_2, \vec{z}_1 \rangle - \alpha \langle \vec{z}_1, \vec{z}_1 \rangle = 0 \implies \frac{\langle \vec{v}_2, \vec{z}_1 \rangle}{\langle \vec{z}_1, \vec{z}_1 \rangle}$
    * $\vec{z}_2 = \vec{v}_2 - \frac{\langle \vec{v}_2, \vec{z}_1 \rangle}{\langle \vec{z}_1, \vec{z}_1 \rangle} \vec{z}_1$
    * $\vec{z}_2$ is the 2nd principal direction in the orthogonal set
  - $\vec{z}_2 = \vec{z}_3 - \alpha_1 \vec{z}_1 - \alpha_2 \vec{z}_2 = \vec{v}_3 - \sum_{l=1}^{2} \alpha_l \vec{z}_l$ s.t. $\vec{z}_3 \perp \vec{z}_1 \& \vec{z}_3 \perp \vec{z}_2$
    * $\langle \vec{z}_3, \vec{z}_1 \rangle = \langle \vec{v}_3 - \alpha_1 \vec{z}_1 - \alpha_2 \vec{z}_2, \vec{z}_1 \rangle = 0$
    * $\langle \vec{z}_3, \vec{z}_2 \rangle = \langle \vec{v}_3 - \alpha_1 \vec{z}_1 - \alpha_2 \vec{z}_2, \vec{z}_2 \rangle = 0$
    * Solve for $\vec{z}_3 = \vec{v}_3 - \frac{\langle \vec{v}_3, \vec{z}_1 \rangle}{\langle \vec{z}_1, \vec{z}_1 \rangle} \vec{z}_1 - \frac{\langle \vec{v}_3, \vec{z}_2 \rangle}{\langle \vec{z}_2, \vec{z}_2 \rangle} \vec{z}_2$
    * $\vec{z}_3$ is the 3nd principal direction in the orthogonal set
  - Continuing in this way: $\vec{z}_n = \vec{v}_n - \sum_{i=1}^{n-1} \frac{\langle \vec{v}_n, \vec{z}_i \rangle}{\langle \vec{z}_i, \vec{z}_i \rangle} \vec{z}_i$
  - Gets the Orthogonal set, but still need to normalize
    * Get orthonormalized set $\vec{q}_1, \ldots, \vec{q}_m$ where $\vec{q}_l = \frac{\vec{z}_l}{\|\vec{z}_l\|}$
    * Orthonormal Representation of Principal Direction:
      · $\alpha_l = \frac{\langle \vec{v}_m, \vec{z}_l \rangle}{\langle \vec{z}_l, \vec{z}_l \rangle}$
      · $\vec{z}_l = \|\vec{z}_l\| \vec{q}_l$
      · $\vec{v}_m = \vec{z}_m + \sum_{l=1}^{m-1} \alpha_l \vec{z}_l = \|\vec{z}_m\| \vec{q}_m + \sum_{l=1}^{m-1} \alpha_l \|\vec{z}_l\| \vec{q}_l = \sum_{l=1}^{m} \alpha_l \|\vec{z}_l\| \vec{q}_l = \sum_{l=1}^{m} r_{lm} \vec{q}_l$
  - This gets $A = QR$ where:
    * R is composed of elements r_{lm} and is upper triangular
    * Q is the orthonormalized matrix $\begin{bmatrix} \vec{q}_1 & \ldots & \vec{q}_m \end{bmatrix}$
    * A is the original principal direction matrix $\begin{bmatrix} \vec{v}_1 & \ldots & \vec{v}_m \end{bmatrix}$
    * $Q^T Q = I$

Gram Schmidt Orthogonalization:

1st principal direction in the orthogonal set

$$\vec{z}_1 = \vec{v}_1 \tag{1}$$

n-th principal direction in the orthogonal set

$$\vec{z}_n = \vec{v}_n - \sum_{i=1}^{n-1} \frac{\langle \vec{v}_n, \vec{z}_i \rangle}{\langle \vec{z}_i, \vec{z}_i \rangle} \vec{z}_i \tag{2}$$

Orthonormal Representation of Principal Direction
R is composed of elements r_{lm} and is upper triangular
Q is the orthonormalized matrix $\begin{bmatrix} \vec{q}_1 & \dots & \vec{q}_m \end{bmatrix}$
A is the original principal direction matrix $\begin{bmatrix} \vec{v}_1 & \dots & \vec{v}_m \end{bmatrix}$

$$A = QR \tag{3}$$

$$\vec{v}_m = \sum_{l=1}^{m} r_{lm} \vec{q}_l \tag{4}$$

EE16A - Lecture 25 Notes

Name: Felix Su SID: 25794773

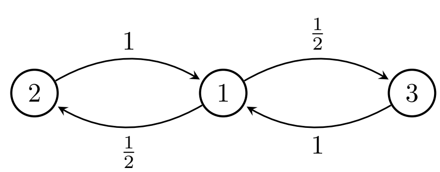
Spring 2016 GSI: Ena Hariyoshi

Pagerank

> 1. Webpages represented by nodes, linked to each other
> 2. Normalize Link Weights: ($\frac{\text{Node Score}}{\text{Out-Degree}}$)
> 3. State at time $n+1 = \vec{s}[n+1] = A\vec{s}[n]$ where A is the **transition matrix** and $\vec{s}$ is the **state vector**
> 4. A is a Markov Matrix
>   - All entries in A are non-negative
>   - Each column of A sums to 1

- Normalize outdegrees of each node, by dividing the weight of the node by the total number of out-links from that node (Node Score/Out-Degree)
  - Node j has n_j out links
  - Weight due to that node would be $\frac{x_j}{n_j}$ where $x =$ node's score
  - Node j's contribution to node k's score, to which it has a link, is $\frac{x_j}{n_j}$
- $A\vec{s}* \vec{s}* = 1I\vec{s}* \implies 1I\vec{s}* - A\vec{s}* = 0 \implies (I-A)\vec{s}* = 0$
- $\vec{s}*$ is non-zero and $(I-A)\vec{s}* = 0$, so $\vec{s}* \in \text{Null}((I-A))$
  - $\text{Null}((I-A))$ is non-trivial ($(I-A)$ does not have an empty null space) $\implies (I-A)$ does not have full rank

Simple Pagerank Example:



$$\vec{s}[n+1] = A\vec{s}[n] = \begin{bmatrix} 0 & 1 & 1 \\ 1/2 & 0 & 0 \\ 1/2 & 0 & 0 \end{bmatrix} \vec{s}[n] \tag{1}$$

Distribution approaches $\vec{s}*$ s.t.:

$$\vec{s}* = A\vec{s}* \tag{2}$$

From $(I-A)\vec{s}* = 0$

$$I - A = \begin{bmatrix} 1-0 & -1 & -1 \\ -1/2 & 1-0 & 0 \\ -1/2 & 0 & 1-0 \end{bmatrix} = \begin{bmatrix} 1 & -1 & -1 \\ -1/2 & 1 & 0 \\ -1/2 & 0 & 1 \end{bmatrix} \tag{3}$$

Eigenvalue representation of $\vec{s}* = A\vec{s}*$ **for** $\lambda = 1$

$$A\vec{v}_1 = \lambda\vec{v}_1 = \vec{v}_1 \tag{4}$$

Eigenvalues and Eigenvectors

> 1. **Concept:** Transition matrix scales $\vec{v}_1$ in its original direction by a factor of λ
> 2. Every transition matrix A will have $\lambda = 1$ as an eigenvalue and a non-negative vector $\vec{v}_1$ associated with that eigenvalue, where the entries of that eigenvector sum to 1.
> 3. Eigenvector $\vec{v}_n$ associated with eigenvalue λ_n is the vector that solves $(I - A)\vec{v}_n = 0$

- Solution to $\vec{s}[n+1] = A\vec{s}[n]$
  - Need Initial State vector $\vec{s}[0]$
- General Solution: $\vec{s}[n] = \alpha_1 \lambda_1^n \vec{v}_1 + \alpha_2 \lambda_2^n \vec{v}_2$
  - Linear combination of eigenvalues and eigenvectors
  - $\vec{s}[n+1] = \alpha_1 \lambda_1^{n+1} \vec{v}_1 + \alpha_2 \lambda_2^{n+1} \vec{v}_2$
  - $A\vec{s}[n] = A(\alpha_1 \lambda_1^n \vec{v}_1 + \alpha_2 \lambda_2^n \vec{v}_2) = \alpha_1 \lambda_1^n A\vec{v}_1 + \alpha_2 \lambda_2^n A\vec{v}_2 = \alpha_1 \lambda_1^n \lambda_1 \vec{v}_1 + \alpha_2 \lambda_2^n \lambda_2 \vec{v}_2$
  - $= \alpha_1 \lambda_1^{n+1} \vec{v}_1 + \alpha_2 \lambda_2^{n+1} \vec{v}_2 = \vec{s}[n+1]$
- Find α_1 and α_2
  - $\vec{s}[0] = \alpha_1 \vec{v}_1 + \alpha_2 \vec{v}_2 = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} \vec{s}_1[0] \\ \vec{s}_2[0] \end{bmatrix}$
- To find the Limiting State Distribution, investigate the linear combination of the eigenvectore and eigenvalues as $n \to \infty$
  - $\lambda_2^n \to 0$ as $n \to \infty$
  - $\lambda_1^n = 1 \forall n$
  - $\lim_{n \to \infty} \vec{s}[n] = \alpha_1 \lambda_1^n \vec{v}_1 = \alpha \vec{v}_1$

Distribution Vector at State n

$$\vec{s}[n] = A^n \vec{s}[0] \tag{5}$$

EE16A - Lecture 26 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

Review Eigenvalues/Eigenvectors

1. $A\vec{v}_n = \lambda_n \vec{v}_n$
   - Scaling factor λ: **Eigenvalue**
   - Vector $\vec{v}_n$: **Eigenvector**
2. For a matrix $M_{n\times n}$ you can have at most n eigenvalues
3. $(A - \lambda I)\vec{v} = 0, x \neq 0 \implies A - \lambda I$ has a non trivial null space and $\det(A - \lambda I) = 0$
   - M is non-invertible
   - M has a non-trivial null space
   - $\det M = 0$
4. If matrix A is symmetric ($A^T = A$), eigenvectors are independent and orthogonal

- Given $\vec{v}_1$ and λ_1 of matrix M_1, find another eigenvalue/eigenvector pair
  - $\vec{v}_2$ is also an eigenvector of M_1 if $\vec{v}_2 = \alpha\vec{v}_1$
    * $M_1\vec{v}_2 = M_1(\alpha\vec{v}_1) = \alpha(M\vec{v}_1) = \alpha(\lambda_1\vec{v}_1) = \lambda_1(\alpha\vec{v}_1) = \lambda_1(\vec{v}_2)$
    * So, $\lambda_2 = \lambda_1$
  - Any eigenvalue $\implies$ an eigenspace (linear combinations of any valid eigenvector works for that eigenvalue)
  - $(A - \lambda I)\vec{v} = 0 \implies A - \lambda I$ has a non-trivial null space
    * $A - \lambda I$ is non-invertible, so the transformation destroys some information (lose a dimension)
    * Invertible matrix (trivial null space) performs a transformation (scale x value)

Linear Transformations

- Given a linear transformation $y = ax + b = 3x$, find eigenvectors and eigenvalues for this transformation
  - Vector in the same direction (**colinear**) with the transformation vector, $\lambda_1 = 1, \vec{v}_1 = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$
  - Vector **orthogonal** to the transformation vector $\lambda_2 = -1, \vec{v}_2 = \begin{bmatrix} -3 \\ 1 \end{bmatrix}$

Determinant

1. The "oriented" volume of the polygon obtained by applying the matrix M onto a unit hypercube.

- Because $A - \lambda I$ is non-invertible, the transformation destroys some information (lose a dimension), so the volume is smaller

Inverse Formula

$$M^{-1} = \frac{1}{\det M}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \tag{1}$$

Solve for Eigenvalues/Eigenvectors

Solve for λ

$$\det(A - \lambda I) = 0 \tag{2}$$

Determinant of a 2×2 matrix

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc \tag{3}$$

Determinant of a $n \times n$ matrix

Mult. each elem. in the first row by the det. of the $n-1 \times n-1$ matrix not in that element's row or column
Take an **alternating sum** of the products from the previous step

$$\det \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = a \times \det \begin{bmatrix} e & i \\ f & h \end{bmatrix} - b \times \det \begin{bmatrix} d & i \\ f & g \end{bmatrix} + c \times \det \begin{bmatrix} d & h \\ e & g \end{bmatrix} \tag{4}$$

Plug resulting Eigenvalues λ_i into $A - \lambda_i I$

$$B_i = A - \lambda_i I = \begin{bmatrix} a_{11} - \lambda_i & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} - \lambda_i & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} - \lambda_i \end{bmatrix} \tag{5}$$

Solve for Eigenvectors $\vec{v}_i$ s.t.

This will be the linear combination of the columns of B_i that cause $B_i \vec{v}_i = 0$

$$(A - \lambda_i I)\vec{v}_i = B_i \vec{v}_i = 0 \tag{6}$$

Properties of Determinants

1. If you **scale a row/col** of a matrix by α, the determinant of the matrix is **multiplied by** α
2. If you **add a scalar multiple of a row/col to any other row/col**, the determinant **doesn't change**
3. If you **swap rows**, the determinant is **multiplied by** -1
4. Determinant of an **upper triangular matrix** is the **product of its pivots**
   - Take unit hypercube, multiply first dimension by a_1, second dimension by a_2...
   - Volum of a hypercube is the product of all its dimensions
5. Generic determinant of $\det(A - \lambda I)$ is $\lambda^n + \alpha_1 \lambda^{n-1} + \cdots + \lambda_n = 0$
   - The max number of roots is n
6. If $\lambda_1 \neq \lambda_2$: eigenspace(λ_1)$\cap$ eigenspace(λ_2)$= \vec{0}$
7. If eigenvectors of A span $\mathbb{R}^n$,$\vec{x} \in \mathbb{R}^n$ can be expressed as $\vec{x} = \sum_{i=1}^{n} \alpha_i \vec{v}_i$
8. $A\vec{x} = \sum_{i=1}^{n} \alpha_i \lambda_i \vec{v}_i$
9. $A^n \vec{x} = \sum_{i=1}^{n} \alpha_i \lambda_i^n \vec{v}_i$ where $\vec{x} = \sum_{i=1}^{n} \alpha_i \vec{v}_i$

- $\vec{x} = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 & \cdots & \vec{v}_n \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix}$
- $\begin{bmatrix} \vec{v}_1 & \vec{v}_2 & \cdots & \vec{v}_n \end{bmatrix}^{-1} \vec{x} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix}$

EE16A - Lecture 27 Notes

Name: Felix Su   SID: 25794773

Spring 2016   GSI: Ena Hariyoshi

Review

1. **Eigenvalue-Eigenvector Eq.**: $A\vec{v} = \lambda\vec{v}$

Diagonlization

- Set of eigenvectors ($\vec{v}_i$) and eigenvalues (λ_i) for $i = 1, \ldots, N$ s.t. $\begin{bmatrix} A\vec{v}_1 & \cdots & A\vec{v}_N \end{bmatrix} = \begin{bmatrix} \lambda\vec{v}_1 & \cdots & \lambda\vec{v}_N \end{bmatrix}$
  - Left side: Take out A
  - Right side: Take out diagonal eigenvalue matrix (λ's)
  - $\implies A \begin{bmatrix} \vec{v}_1 & \cdots & \vec{v}_N \end{bmatrix} = \begin{bmatrix} \vec{v}_1 & \cdots & \vec{v}_N \end{bmatrix} \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}$
  - So, $AV = V\Lambda$
  - Postmultiply by V^{-1} on both sides to get $A = V\Lambda V^{-1}$

Summary Equation for Eigen Decomposition of Matrix A:

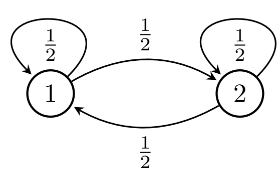
$$AV = V\Lambda \text{ where } V = \begin{bmatrix} \vec{v}_1 & \cdots & \vec{v}_N \end{bmatrix} \text{ and } \Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix} \tag{1}$$

$$A = V\Lambda V^{-1} \tag{2}$$

Pagerank

- $A = V\Lambda V^{-1}$
- $s[n] = A^n s[0]$
- $A^n = V\Lambda^n V^{-1}$
  - If $|\lambda_i| > 1 \to \lambda_i^n$ keeps growing
  - If $|\lambda_i| < 1 \to \lambda_i^n$ decays towards 0
  - If $|\lambda_i| = 1 \to \lambda_i^n$ stays the same

Example:
Pagerank Example



$A = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$
1 linearly independent col/row $\implies$ rank(A) = 1
One of the eigenvalues has to be zero: $\lambda_1 = 0$, the other is non-zero: $\lambda_2 = n$
0 eigenvalue corresponds to singular matrix
A is a projection matrix onto vector $\vec{r} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$
Projection matrix $P = \frac{\vec{r}\vec{r}^T}{\vec{r}^T\vec{r}}$
A is symmetric
A is column and row-stochastic (doubly-stochastic)
column-stochastic matrix = Markoc matrix $A \geq 0$ (non-negative)
$A - \lambda I = \begin{bmatrix} \frac{1}{2} - \lambda & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} - \lambda \end{bmatrix}$
$\det(A - \lambda I) = (\frac{1}{2} - \lambda)^2 - (\frac{1}{2})^2 = 0 \implies \lambda(1-\lambda)0$, so, $\lambda = 0, 1$
Markov matrices always have 1 as an eigenvalue (All other eigenvalues are smaller), so, as teh system continues to run, the other eignvalues decay to 0 and the eigenvector corresponding to the $\lambda_i = 1$ will be the **important score** for the website
$\lambda_1 = 0, \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} = 0 \implies \vec{v}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$
$\lambda_2 = 1, \begin{bmatrix} \frac{1}{2} - 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} - 1 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix} = 0 \implies \vec{v}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$
$V = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$
$\Lambda = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$
$V^{-1} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}$
$A = V\Lambda V^{-1} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 \end{bmatrix}\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}\begin{bmatrix} \vec{w}_1^T \\ \vec{w}_2^T \end{bmatrix} = \lambda_1\vec{v}_1\vec{w}_1^T + \lambda_2\vec{v}_2\vec{w}_2^T$
$A^n = V\Lambda^n V^{-1} = \sum_{i=1}^{N} \lambda_i^n \vec{v}_i\vec{w}_i^T$

General Form of nth Power of a Matrix

$$A^n = V\Lambda^n V^{-1} = \sum_{i=1}^{N} \lambda_i^n \vec{v}_i\vec{w}_i^T \tag{3}$$

Importance Score for Pagerank

Pagerank Importance Score
Converges to the value $A^n s[0]$ where:

$$\lim_{n\to\infty} A^n = \lambda_i\vec{v}_i\vec{w}_i^T = \vec{v}_i\vec{w}_i^T \text{ where } \lambda_i = 1 \text{ if } |\lambda_i| < 1 \forall i = 1, \ldots, N-1 \tag{4}$$

Preview of Next Lecture

- If $s[n+1] = As[n]$, and $\vec{q}[n] = V^{-1}s[n] \implies \vec{s}[n] = V\vec{q}[n]$
- Then, $V\vec{q}[n+1] = AV\vec{q}[n] \implies \vec{q}[n+1] = V^{-1}AV\vec{q}[n]$

- Because $A = V\Lambda V^{-1} \implies V^{-1}\Lambda V = \Lambda$
- $\therefore \vec{q}[n+1] = \Lambda\vec{q}[n]$
  - Decouples dynamic system

EE16A - Lecture 28 Notes

Name: Felix Su SID: 25794773

Spring 2016 GSI: Ena Hariyoshi

Decoupling State Space Equation

1. Summary

- State Space equation (SSE): $\vec{s}[n+1] = A\vec{s}[n]$
- $\vec{s}[n] = V\vec{q}[n] \implies \vec{s}[n+1] = V\vec{q}[n+1]$
  - SSE becomes $V\vec{q}[n+1] = AV\vec{q}[n]$
  - $A = V\Lambda V^{-1} \implies V^{-1}AV = \Lambda$
  - Premultiply by $V^{-1} \implies \vec{q}[n+1] = V^{-1}AV\vec{q}[n]$
- So, you have a new equation: $\vec{q}[n+1] = \Lambda\vec{q}[n]$
- Good because:
  - Original SSE: $\vec{s}_k[n+1] = \begin{bmatrix} a_{k1} & a_{k2} & \cdots & a_{kN} \end{bmatrix} \begin{bmatrix} \vec{s}_1[n] & \vec{s}_2[n] & \cdots & \vec{s}_N[n] \end{bmatrix}^T = \sum_{l=1}^{N} \vec{a}_{kl}\vec{s}_k[n]$
  - Can't solve this independently of other state variables, because they are coupled and appear in the right hand side of the equation
  - Shift from $\vec{s}$ to $\vec{q}$ = decoupled state variables
  - Can solve for the state $\vec{q}$ independently of the others

Change of Basis:

$$\begin{bmatrix} \vec{q}_1[n+1] \\ \vdots \\ \vec{q}_k[n+1] \\ \vdots \\ \vec{q}_N[n+1] \end{bmatrix} = \begin{bmatrix} \lambda_1 & & & & 0 \\ & \ddots & & & \\ & & \lambda_k & & \\ & & & \ddots & \\ 0 & & & & \lambda_N \end{bmatrix} \begin{bmatrix} \vec{q}_1[n] \\ \vdots \\ \vec{q}_k[n] \\ \vdots \\ \vec{q}_N[n] \end{bmatrix} = \begin{bmatrix} \lambda_1\vec{q}_1[n] \\ \vdots \\ \lambda_k\vec{q}_k[n] \\ \vdots \\ \lambda_N\vec{q}_n[n] \end{bmatrix} \tag{1}$$

Decoupled State Evolution Equation

$$\vec{q}_k[n+1] = \lambda_k\vec{q}_k[n], \text{ for } k = 1, \ldots, N \tag{2}$$

Solving independent Decoupled States
Given $\vec{s}[0]$, $\vec{q}[0] = V\vec{s}[0]$, so:

$$\vec{q}_k[n] = \lambda_k^n\vec{q}_k[0] \text{ for } k = 1, \ldots, N \tag{3}$$

Solving $\vec{q}_k$

- $\vec{s}[n] = V\vec{q}[n] \to \vec{q}[n] = V^{-1}\vec{s}[n]$
- We know initial state $\vec{s}[0]$ so we also know initial state $\vec{q}[0] \to \vec{q}[0] = V^{-1}\vec{s}[0]$
- So, we have $\vec{q}_k[0]$ (known) and the new state evolution equation $\vec{q}_k[n+1] = \lambda_k\vec{q}_k[n]$
- This means, $\vec{q}_k[n] = \lambda_k^n\vec{q}_k[0]$ for $k = 1, \ldots, N$

Basis Transformation

- Canonical basis: set of vectors e_i that form the identity matrix
- Want to use another set of lin. indep. vectors $\vec{z_1}, \ldots, \vec{z}_n$ as our principal axes
  - $\vec{s} = Z\vec{x}^{[Z]}$ where $Z = span\{\vec{z_1}, \ldots, \vec{z}_n\}$
  - Want to express vecotr x in terms of vectors in z
  - Z is a square matrix of linearly independent cols, so it is an invertible matrix
  - $\vec{x}^{[Z]}$ represents $\vec{x}$ in the corrdinate sys. given by the z's
- $x = Z\vec{x}^{[Z]} \implies \vec{x}^{[Z]} = Z^{-1}\vec{x}$, one to one mapping between x and $x^{[Z]}$ because Z is invertible.

Representation of a Linear Transformation in a new Coordinate System

- $\vec{y} = A\vec{x}$ where $\vec{x} = Z\vec{x}^{[Z]}$ and $\vec{y} = Z\vec{y}^{[Z]}$
- So, $Z\vec{y}^{[Z]} = AZ\vec{x}^{[Z]}$ and Z is invertible, so,
- $\vec{y}^{[Z]} = Z^{-1}AZ\vec{x}^{[Z]} \to y = Ax$
  - A takes the vector x and maps it to the vector y in the orginal coordinate system
  - In the new coordinate system, the same linear transformation maps the coordinate vector for x into the coordinate vector for y by $A^{[Z]} = Z^{-1}AZ$, the transfomation matrix in the new coordinate system
- If $Z = V$ (eigenvector matrix for A) where $A = V\Lambda V^{-1}$
  - Then the coordinate transformation diagonlizes A
  - $A^{[Z]} = V^{-1}AV = V^{-1}V\Lambda V^{-1}V = \Lambda$
  - Switching to this new coordinate system and diagonalizing matrix A decouples the state variables in the new coordinate system

From Canonical Coordinate System to Coordinate System of Z

$$\vec{y}^{[Z]} = A^{[Z]}\vec{x}^{[Z]} \text{ where } A^{[Z]} = Z^{-1}AZ \tag{4}$$

Transformations Across Coordinate Systems

$$y \to y\vec{y}^{[Z]} = Z^{-1}y \tag{5}$$

$$x \to x\vec{x}^{[Z]} = Z^{-1}x \tag{6}$$

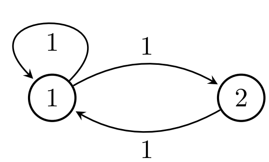
$$A \to A^{[Z]} = Z^{-1}AZ \tag{7}$$

If $Z = V$ (eigenvector matrix for A) where $A = V\Lambda V^{-1}$

$$A^{[Z]} = \Lambda \tag{8}$$

Example:
Fibonacci Sequence

$$\vec{s}[0] = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$



$\vec{s}[n+1] = A\vec{s}[n]$
$A = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$
$\vec{s}[1] = A\vec{s}[0] = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \vec{s}[2] = A\vec{s}[1] = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad \vec{s}[3] = A\vec{s}[2] = \begin{bmatrix} 3 \\ 2 \end{bmatrix} \quad \vec{s}[4] = A\vec{s}[3] = \begin{bmatrix} 5 \\ 3 \end{bmatrix} \quad \vec{s}[5] = A\vec{s}[4] = \begin{bmatrix} 8 \\ 5 \end{bmatrix}$
Eigendecomposition of A
Find $(\lambda_1, \vec{v}_1)$ $(\lambda_2, \vec{v}_2)$
$\vec{s}[n] = \alpha_1 \lambda_1^n \vec{v}_1 + \alpha_2 \lambda_2^n \vec{v}_2$ where α_1, α_2 are determined by initial state: $\vec{s}[0] = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ s.t. $\begin{bmatrix} \vec{v}_1 & \vec{v}_2 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ **Solve**
$A - \lambda I$ $A - \lambda I = \begin{bmatrix} 1-\lambda & 1 \\ 1 & -\lambda \end{bmatrix}$ where $\det(A - \lambda I) = -\lambda(1-\lambda) - 1 = la^2 - \lambda - 1 = 0$
Roots: $\frac{1 \pm \sqrt{5}}{2}$, $\lambda_1 = \frac{1-\sqrt{5}}{2}$, $\lambda_2 = \frac{1+\sqrt{5}}{2}$ (**Golden Ratio**)