Week 2 (& 3): Reviews

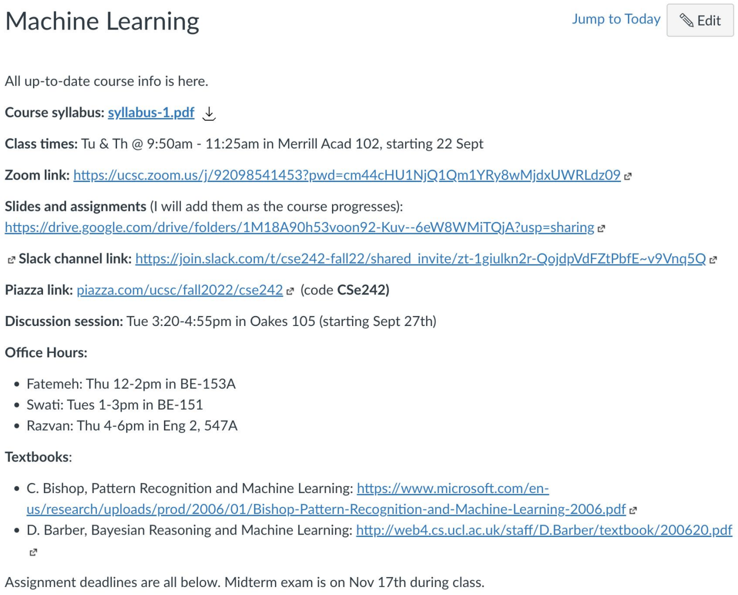
- Probability review
  - Bayesian Learning

- Linear Algebra
  - Principle Component Analysis

- News:
  - On Saturday, we added 3 additional questions to Assignment 1
  - Assignment 1 due Oct 3rd
  - Discussion session today @ 3:20-4:55pm in Oakes 105

Razvan Marinescu

- **News:**
  - On Saturday, we added 3 additional questions to Assignment 1
    - Existing questions did not change
  - Assignment 1 due Oct 3rd – do start working on it!
  - Discussion session today @ 3:20-4:55pm in Oakes 105
    - How to run a jupyter notebook, how to submit the assignment on gradescope, revision of lectures
  - All up-to-date info is on the Canvas home page



Machine Learning

All up-to-date course info is here.

Course syllabus: syllabus-1.pdf

Class times: Tu & Th @ 9:50am - 11:25am in Merrill Acad 102, starting 22 Sept

Zoom link: https://ucsc.zoom.us/j/92098541453?pwd=cm44cHU1NjQ1Qm1YRy8wMjdxUWRLdz09

Slides and assignments (I will add them as the course progresses):
https://drive.google.com/drive/folders/1M18A90h53voon92-Kuv--6eW8WMiTQjA?usp=sharing

Slack channel link: https://join.slack.com/t/cse242-fall22/shared_invite/zt-1giulkn2r-QojdpVdFZtPbfE~v9Vnq5Q

Piazza link: piazza.com/ucsc/fall2022/cse242 (code **CSe242)**

Discussion session: Tue 3:20-4:55pm in Oakes 105 (starting Sept 27th)

Office Hours:

- Fatemeh: Thu 12-2pm in BE-153A
- Swati: Tues 1-3pm in BE-151
- Razvan: Thu 4-6pm in Eng 2, 547A

Textbooks:

- C. Bishop, Pattern Recognition and Machine Learning: https://www.microsoft.com/en-us/research/uploads/prod/2006/01/Bishop-Pattern-Recognition-and-Machine-Learning-2006.pdf
- D. Barber, Bayesian Reasoning and Machine Learning: http://web4.cs.ucl.ac.uk/staff/D.Barber/textbook/200620.pdf

Assignment deadlines are all below. Midterm exam is on Nov 17th during class.

Probability Review

Probability Review

- Based on experiment: outcome space Ω containing all possible atomic outcomes
  - {1,2,3,4,5,6} for a fair die
- Each outcome (atom) has probability density or mass (discrete vs. continuous spaces)
  - $p(X = 1) = 1/6$
- Event is a subset of Ω
  - Example event "Die lands on odd numbers": {1,3,5}
- P(event) is sum (or integral) over event's atoms
- Random variable *V* is a function that maps Ω to (usually) *R*
- *V*=value is an event, P(*V*) is a distribution

Example

- Roll a fair 6-sided die and then flip that many fair coins.
- What is Ω?

Example

- Roll a fair 6-sided die and then flip that many fair coins.
- What is Ω?
- Ω={(1,H), (1,T), (2, HH), (2, HT), …, (6,TTTTTT)}

Event F = “the die lands on an even number”

Example

- Roll a fair 6-sided die and then flip that many fair coins.
- What is Ω?
- Ω={(1,H), (1,T), (2, HH), (2, HT), …, (6,TTTTTT)}

Event F = "the die lands on an even number"

  - F = {(2, HH), (2, HT), …, (4, HHHH) … (4, TTTT), … (6, HHHHHH), … , (6,TTTTTT)}

- Number of heads is a random variable

Expectation

- What is the expected number of heads? Expectation of *V* is

$$\mathbb{E}[V] = \sum_{\text{atoms } a} \mathbb{P}(a) \cdot V(a)$$

In previous example, atoms are (1, H) or (2, HT)

Expectation and Variance

- Expectation for discrete random variables

$$\mathrm{E}[X] = \sum_{i=1}^{\infty} x_i\, p_i$$

- Expectation for continuous random variables

$$\mathrm{E}[X] = \int_{-\infty}^{\infty} x f(x)\, dx.$$

- Variance

$$\mathrm{Var}(X) = \mathrm{E}\left[(X - \mathrm{E}[X])^2\right]$$

Variance expansion

$$\begin{aligned}
\mathrm{Var}(X) &= \mathrm{E}\left[(X - \mathrm{E}[X])^2\right] \\
&= \mathrm{E}\left[X^2 - 2X\,\mathrm{E}[X] + \mathrm{E}[X]^2\right] \\
&= \mathrm{E}\left[X^2\right] - 2\,\mathrm{E}[X]\,\mathrm{E}[X] + \mathrm{E}[X]^2 \\
&= \mathrm{E}\left[X^2\right] - \mathrm{E}[X]^2
\end{aligned}$$

Independence and conditional probability

- Events A and B independent iff:

  $$P(A \text{ and } B) = P(A) \bullet P(B)$$

- Conditional probability of A given B

  $$P(A \mid B) = P(A \text{ and } B) / P(B)$$

- So,

  $$P(A \text{ and } B) = P(A \mid B) \bullet P(B)$$

  $$P(B \text{ and } A) = P(B \mid A) \bullet P(A)$$

- Bayes Rule:

  $$P(A \mid B) = P(B \mid A)\, P(A) / P(B)$$

Expectation and sum-rule

- Expectations add: $E(V_1 + V_2) = E(V_1) + E(V_2)$
- Rule of conditioning: (sum rule)

if events $e_1, e_2, \ldots, e_k$ **partition** Ω then:

$$P(\text{event}) = \sum P(e_i)\, P(\text{event} \mid e_i)$$

$$= \sum P(e_i \text{ and event})$$

$$E(\text{randVar}) = \sum P(e_i)\, E(\text{randVar} \mid e_i)$$

Expected number of heads

- E(# heads) $= \sum_{r=1}^{6} P(\text{roll} = r) E(\# \text{heads} \mid \text{roll} = r)$

$$= \frac{1}{6}\left(\frac{1+2+3+4+5+6}{2}\right)$$

$$= \frac{21}{12} = 1.75$$

- Joint Distributions factor:

  If $\Omega = (S \times T \times U)$ then $P(S=s, T=t, U=u)$ is

  $P(S=s)\ P(T=t \mid S=s)\ P(U=u \mid S=s, T=t)$

  (can draw one at a time with conditioning)

- Conditional distributions are distributions:

  $P(A \mid B) = P(A \text{ and } B) / P(B)$, so also:

  $P(A \mid B, C) = P(A \text{ and } B \mid C) / P(B \mid C)$

Bayesian Learning

Bayes Rule for Learning

RVs

- Assume joint distribution $P(\boldsymbol{X}=\boldsymbol{x}, Y=y)$
- Want $P(Y=y \mid \boldsymbol{X}=\boldsymbol{x})$ for each label y on a new instance $\boldsymbol{x}$ (here $(\boldsymbol{x},y)$ is an atom)
- $P(y \mid \boldsymbol{x}) = P(\boldsymbol{x} \mid y) \bullet P(y) \,/\, P(\boldsymbol{x})$ Bayes rule
  - Posterior (over Y)
  - Likelihood (of X given Y)
  - Prior (over Y)
  - Normalization constant or Partition function (often intractable)

Bayes Rule for Learning

- $P(y \mid \boldsymbol{x})$ proportional to $P(\boldsymbol{x} \mid y) \bullet P(y)$
- From data, learn $P(\boldsymbol{x} \mid y)$ and $P(y)$
- Predict label *y* with largest product

How to learn probabilities

- Street hustler takes bets on coin flips
- You see HTH, what is probability that next flip is H? What is P(H) for coin?

(don’t be shy)

Frequentist solution

- Street hustler takes bets on coin flips
- You see HTH. What is P(H) for the coin?
  - (can assume coin is not necessarily fair)

JOIN QUIZ

https://quizizz.com/join

Please use your full name at sign-in, not a nickname

Frequentist solution

- You are a street hustler taking bets on coin flips
- *θ = p(H) (not necessarily 0.5)*
- You see HTH. What is P(H) for the coin?
  - (can assume coin is not necessarily fair)
  - p(HTH) = p(H)p(T)p(H) = $\theta(1-\theta)\theta = \theta^2(1-\theta)$
  - Maximise p(HTH) by setting the derivative to zero: ∂p(HTH)/ $\partial\theta = 0$
  - ∂p(HTH)/ $\partial\theta = 2\theta - 3\theta^2 = 0 \Rightarrow \theta = 2/3 = p(H)$

Frequentist

- Frequentist maximizes likelihood
- Likelihood function $L(\theta) = P(HTH \mid \theta)$
- Frequentist performs $\theta^* = \text{argmax}_{\theta} L(\theta)$
- The probability $P(HTH \mid \theta=2/3)$ is
  - $P(HTH \mid \theta=2/3) = P(H \mid \theta=2/3)P(T \mid \theta=2/3)P(H \mid \theta=2/3) = 2/3 * 1/3*2/3 = 4/27$

Bayesian Parameter Estimation

- Have prior distribution $P(\theta)$ on $\theta = P(H)$; two phase experiment, pick θ then flip 3 times

- Posterior on θ is given by Bayes' rule:

$$P(\theta \mid HTH) = P(HTH \mid \theta) \cdot P(\theta) \, / \, P(HTH)$$

- In this case,

$$\theta^2(1-\theta)P(\theta) \, / \, \text{normalization}$$

Bayesian examples

- In Bayesian methods, we need to set a prior
- Example Prior:

  $P(\theta=0) = P(\theta=1/2) = P(\theta=1) = 1/3$:

- $\theta^2(1-\theta)$ $P(\theta)$ is 0, 1/24, and 0 for these three cases

  posterior $P(\theta=1/2 \mid HTH) = 1$

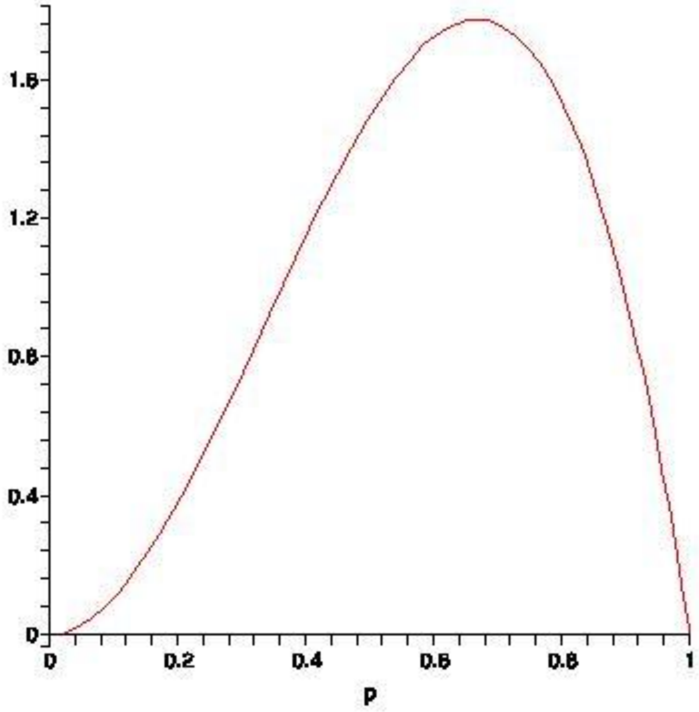
Bayesian examples

- Prior density: $P(\theta) = 1$ for $0 \leq \theta \leq 1$:

$\theta^2(1-\theta)\ P(\theta)$ is $\theta^2(1-\theta)$ for $0 \leq \theta \leq 1$

posterior $P(\theta \mid HTH)$ is $12\ \theta^2(1-\theta)$

Posterior plot



- Max at 2/3
- Average is 3/5
- 3/5 = (2+1)/(3+2)
- Not a coincidence!

  Laplace’s rule of succession - add one fictitious observation of each class

Bayes' Estimation

- Treat parameter θ as a random var with the prior distribution $P(\theta)$, see training data Z,

P(*ML model* | Data) = P(*Model*) P*(*Data | *Model)* / P*(*Z *)*

Posterior *Prior* *Data likelihood* *constant*

P*(θ | Z)* *proportional to* P*(θ)* P*(*Z | *θ)*

Posterior *Prior* *Data likelihood*

Bayes' Estimation

- Treat parameter θ' as a random var with the prior distribution $P(\theta)$, use fixed data Z (RV S)

- Maximum Likelihood (ML):
  - $\theta_{ML} = \text{argmax}_{\theta'} P(Z \mid \theta = \theta')$

- Maximum a Posteriori (MAP):
  - $\theta_{MAP} = \text{argmax}_{\theta'} P(\theta = \theta' \mid Z)$
    $= \text{argmax}_{\theta'} P(Z \mid \theta = \theta') P(\theta = \theta') / P(Z)$

Use for learning

RVs

- Draw enough data so that $P(Y=y \mid X=\boldsymbol{x})$ estimated for every possible $(\boldsymbol{x},y)$ pair

- This takes lots of data – curse of dimensionality …rote learning

- Another approach: a class of models

- Think of each model *m* as a way of generating the training set Z of $(\boldsymbol{x},y)$ pairs

The "Data Experiment"

- Prior $P(M=m)$ on model space

- Models gives $P(Z=z \mid M=m)$ (here data Z is both *y*'s and ***x***'s)

Joint experiment (if data i.i.d. given *m*)

$$P(\{(\mathbf{x}_i, y_i)\}, m) = P(m) \prod_i (P(\mathbf{x}_i | m) P(y_i \mid \mathbf{x}_i, m))$$

Bayesian model selection

- **Prior** $P(m)$ over models
- Each model gives $P(Z \mid m)$
- **Posterior** $P(m \mid Z) = P(Z \mid m)\, P(m) / P(Z)$

- *Max. likelihood*: m having max $P(Z \mid m)$

- *Max. a'posteriori*: m having max $P(m \mid Z)$

Discriminative and Generative models

- **Generative model**: $P((\boldsymbol{x}, y) \mid m)$
  - Tells how to generate examples (both instances and labels)

- **Discriminative model**: $P(y \mid m, \boldsymbol{x})$
  - Tells how to create labels from instances, (like linear regression)
  - **Discriminate function**: predict y = $f(\boldsymbol{x})$,
  - often $f(\boldsymbol{x}) = \text{argmax}_y\ f_y(\boldsymbol{x})$.

More on Generative approach

- Generative approach models P($\boldsymbol{x}$, y / m)

- Learn P($\boldsymbol{x}$ | y,m) and use Bayes' rule

  P(y | $\boldsymbol{x}$,m) = P($\boldsymbol{x}$ | y,m) P(y / m) / P($\boldsymbol{x}$/m)

- Need model for P($\boldsymbol{x}$ | y, m)

More on Generative approach

- Need model for P($\boldsymbol{x}$ | y, m)

- One common assumption:

  P($\boldsymbol{x}$ | y,m) Gaussian

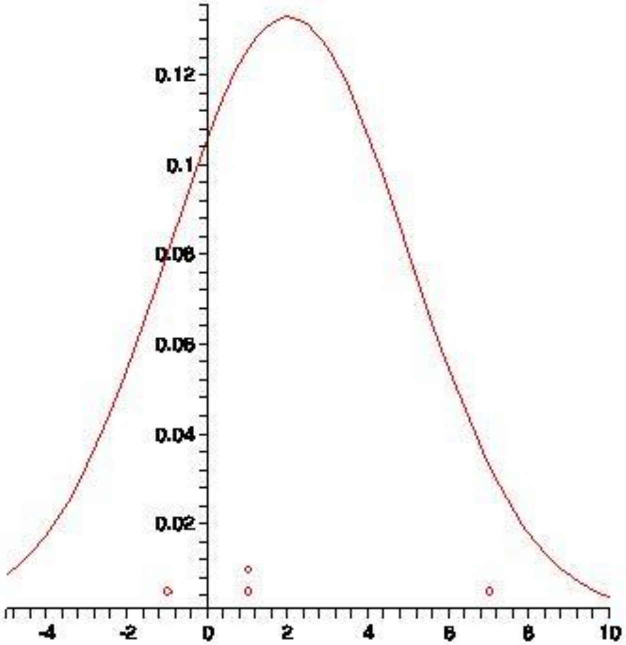
  P(y | m) Bernoulli (biased coin flip)

- How to learn (fit) Gaussian from data?

1 dimensional Gausians

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2} \right)$$

- Maximum likelihood estimate has
  sample mean $\mu = (1/n) \sum_i x_i$
  and
  sample variance $\sigma^2 = (1/n) \sum_i (x_i - \mu)^2$
  $= E[(x - \mu)^2]$
  $= E[x^2] - \mu^2$
- What (μ, σ^2) best fits [-1, 1, 1, 7] ?


0.12
0.1
0.08
0.06
0.04
0.02
-4
-2
0
2
4
6
8
10

Multivariate Gaussians

$$P(\boldsymbol{x}) = \frac{1}{(2\pi)^{n/2} \, |\Sigma|^{1/2}} \exp(-\frac{1}{2}[\boldsymbol{x} - \mu]^T \Sigma^{-1} [\boldsymbol{x} - \mu])$$

- Mean vector μ, covariance matrix Σ, entries of covariance matrix are variances (or co-variances), $\sigma^2{}_{i,j}$=E[(x_i- μ_i) (x_j- μ_j)]
  (subscripts are indices into vectors)

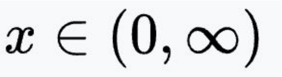
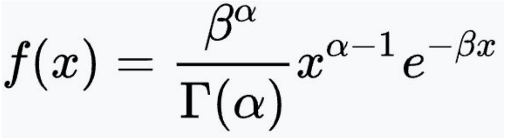
Estimating Gaussians (maximum likelihood)

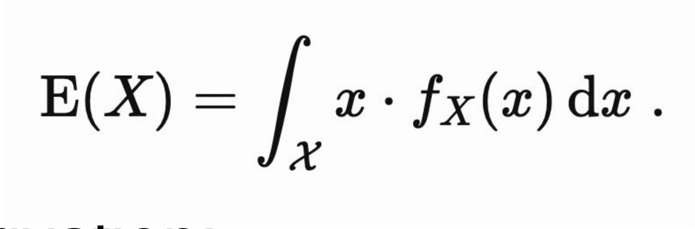
- Maximum likelihood estimate: argmax $_{\boldsymbol{\mu}}$ p($\boldsymbol{x}$| $\boldsymbol{\mu}$, σ)
  - $\boldsymbol{\mu}^* = (\sum_k \boldsymbol{x_k}) / n$ (Exercise: prove this)
- For covariance we get
  - $\sigma^2_{i,j} = (\sum_k (x_{k,i} - \mu_i) (x_{k,j} - \mu_j)) / n$
  - this is biased: use "n-1" for normalization
- If domain d-dimensional:
  - d parameters for $\boldsymbol{\mu}$
  - $d(d+1)/2$ parameters for $\sigma^2_{i,j}$'s
  - For each class!
  - Many parameters "requires" lots of data

Common tricks

- Share same Σ for all classes
- Assume diagonal Σ's for each class
- Assume shared $\Sigma = c\boldsymbol{I}$ (spherical)
  - This leads to the simple mean-based linear classifier if data balanced

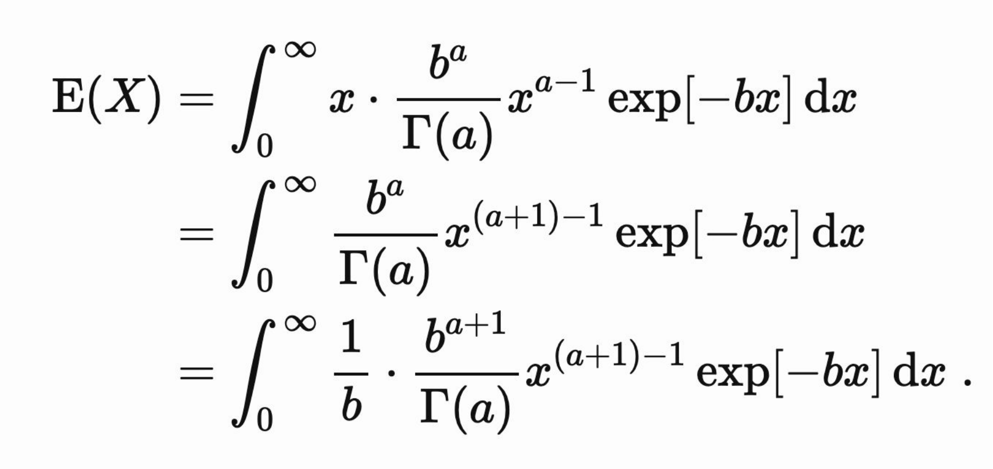
Exercise: Expectation of a Gamma random variable

- Support:  $x \in (0, \infty)$
- Gamma probability density function:  $f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$
- Expectation:



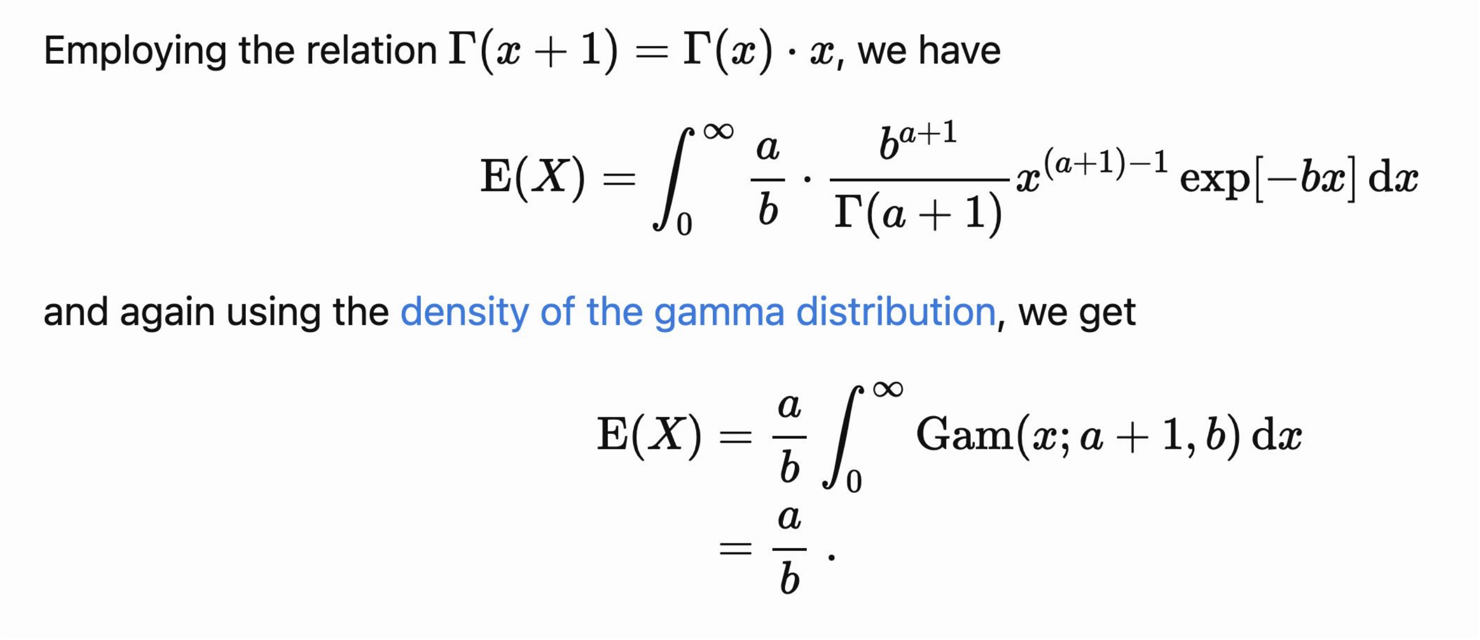
$$\mathrm{E}(X) = \int_{\mathcal{X}} x \cdot f_X(x) \, \mathrm{d}x \; .$$

- Derivation:



$$\begin{aligned} \mathrm{E}(X) &= \int_0^\infty x \cdot \frac{b^a}{\Gamma(a)} x^{a-1} \exp[-bx] \, \mathrm{d}x \\ &= \int_0^\infty \frac{b^a}{\Gamma(a)} x^{(a+1)-1} \exp[-bx] \, \mathrm{d}x \\ &= \int_0^\infty \frac{1}{b} \cdot \frac{b^{a+1}}{\Gamma(a)} x^{(a+1)-1} \exp[-bx] \, \mathrm{d}x \; . \end{aligned}$$

Exercise: Expectation of a Gamma random variable



Employing the relation $\Gamma(x+1) = \Gamma(x) \cdot x$, we have

$$\mathrm{E}(X) = \int_0^\infty \frac{a}{b} \cdot \frac{b^{a+1}}{\Gamma(a+1)} x^{(a+1)-1} \exp[-bx] \, \mathrm{d}x$$

and again using the density of the gamma distribution, we get

$$\begin{aligned} \mathrm{E}(X) &= \frac{a}{b} \int_0^\infty \mathrm{Gam}(x; a+1, b) \, \mathrm{d}x \\ &= \frac{a}{b} \; . \end{aligned}$$

Week 2: Reviews

- Linear Algebra
  - Principal Component Analysis

- News:
  - Assignment deadline extended to Oct 5th.
  - **Late days policy:** Each student has a total of five late days for use on assignments. You can extend each assignment due date by either 24 hours or 2 days.
  - Additional late days might only be considered for very special circumstances. Email my TAs to explain your circumstances, and they will decide if they grant an exception.

Razvan Marinescu

Linear Algebra

- A matrix:

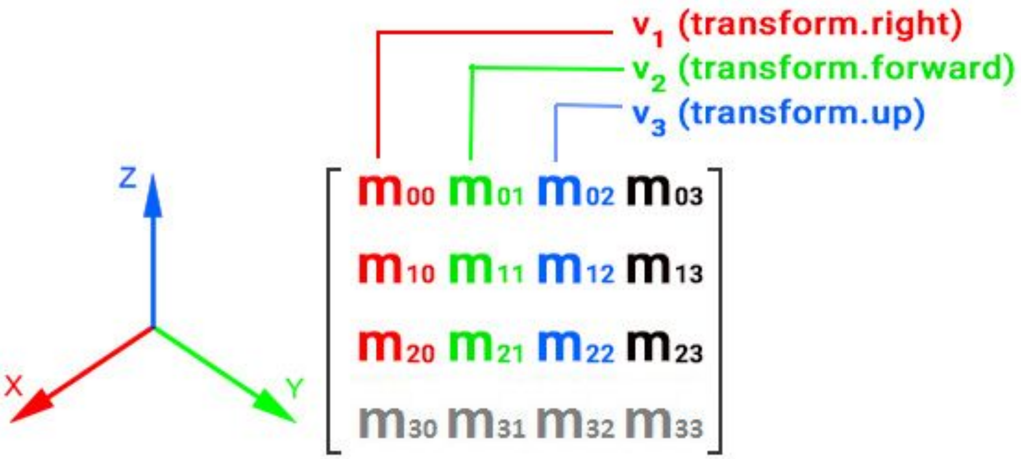
$$A \in \mathbb{R}^{m \times n} \text{ : a matrix with } m \text{ rows and } n \text{ columns}$$

- A vector:

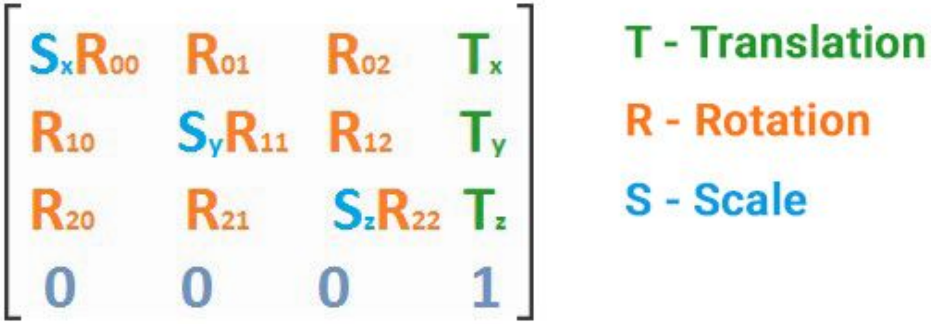
$$x \in \mathbb{R}^{n} \text{ : a vector with } n \text{ entries}$$

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \qquad x = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_i \\ \dots \\ x_n \end{bmatrix}$$

Matrices always represent a **linear** transformation



$$\begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \\ m_{30} & m_{31} & m_{32} & m_{33} \end{bmatrix}$$



$$\begin{bmatrix} S_xR_{00} & R_{01} & R_{02} & T_x \\ R_{10} & S_yR_{11} & R_{12} & T_y \\ R_{20} & R_{21} & S_zR_{22} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

T - Translation

R - Rotation

S - Scale

Linear Algebra

- To differentiate from a scalar, we often use bolded $\mathbf{x}$
- An element (i_{th}) is denoted as x_i
- An element in a matrix is denoted as $a_{i,j} \text{ or } A_{i,j}$
- j_{th} column of A: $a_j \text{ or } A_{:,j}$
- i_{th} row of A: $a_i^\top \text{ or } A_{i,:}$

Matrix multiplication

$A \in \mathbb{R}^{m \times n}, B \in \mathbb{R}^{n \times p};$

$C = AB \in \mathbb{R}^{m \times p}$

$$C_{ij} = \sum_{k=1}^{n} A_{ik} B_{kj}$$

Inner product $x^\top \cdot y \in \mathbb{R}$

Outer product $x \cdot y^\top \in \mathbb{R}^{m \times n}$

Matrix multiplication

$$Ax = \begin{bmatrix} a_1^\top x \\ a_2^\top x \\ \dots \\ \dots \\ a_m^\top x \end{bmatrix}$$

Matrix multiplication

- Associative property

$$(AB)C = A(BC)$$

- Distributive property

$$A(B + C) = AB + AC$$

Operation

Identity matrix I $\quad AI = A = IA$

Diagonal matrix $D : D_{ij} = 0, \text{ if } \; i \neq j$

Transpose $(A^\top)_{ij} = A_{ji}$

$$(A^\top)^\top = A$$

$$(AB)^\top = B^\top A^\top$$

$$(A + B)^\top = A^\top + B^\top$$

Operation

Trace of a matrix $$\mathrm{tr}(A) = \sum_{i=1}^{n} A_{ii}$$

$$\mathrm{tr}(A) = \mathrm{tr}(A^\top)$$

$$\mathrm{tr}(A + B) = \mathrm{tr}(A) + \mathrm{tr}(B)$$

$$\mathrm{tr}(cA) = c\mathrm{tr}(A), c \in \mathbb{R}$$

Operation

Norm

$$||x||_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$$

$$||x||_2^2 = x^\top x$$

$$||x||_1 = \sum_{i}^{n} |x_i$$

$$||x||_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}$$

$$||x||_\infty = \max_i |x_i|$$

$$||A||_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} A_{ij}^2} = \sqrt{\mathrm{tr}(A^\top A)}$$

Operation

Rank $\mathrm{rank}(A)$

Inverse $A^{-1} : A^{-1}A = I$

$$(A^{-1})^{-1} = A$$

$$(AB)^{-1} = B^{-1}A^{-1}$$

Orthogonal $x^\top y = 0$

Operation

Projection: Use PCA as an example shortly

Eigenuevalues and Eigenvectors

$$Ax = \lambda x, x \neq 0$$

x = eigenvectors
λ = eigenvalues

Eigenvalue decomposition

$$\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{-1}$$

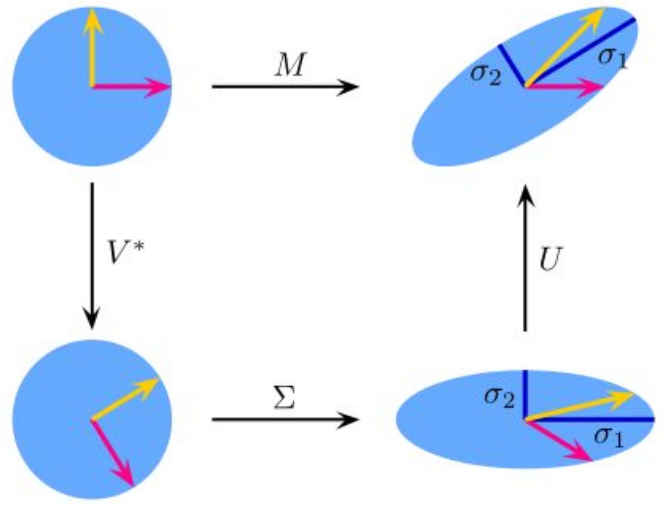
- A has to be square (n x n) and diagonalizable
- Q = eigenvectors of A
- Λ = diagonal matrix with eigenvalues of A

Singular value decomposition

$$\mathbf{A}_{nxp} = \mathbf{U}_{nxn}\ \mathbf{S}_{nxp}\ \mathbf{V}^{\mathbf{T}}_{pxp}$$

- U = eigenvectors of AA^T
- V = eigenvectors of A^TA
- S = matrix containing singular values on the diagonal
- Singular values = square root of eigenvalues of either AA^T or A^TA

- As opposed to eigenvalue decomposition, SVD can be applied to any matrix

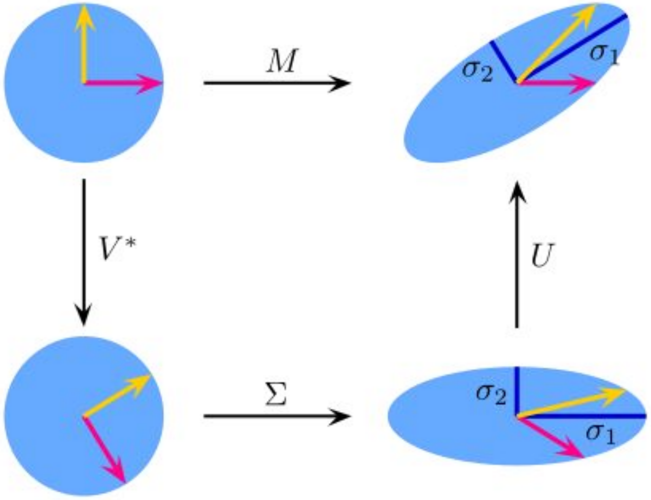


$$M = U \cdot \Sigma \cdot V^*$$

Singular value decomposition

$$\mathbf{A}_{nxp} = \mathbf{U}_{nxn}\, \mathbf{S}_{nxp}\, \mathbf{V^T}_{pxp}$$

- U = eigenvectors of AA^T
- V = eigenvectors of A^TA
- S = matrix containing singular values on the diagonal
- Singular values = square root of eigenvalues of either AA^T or A^TA



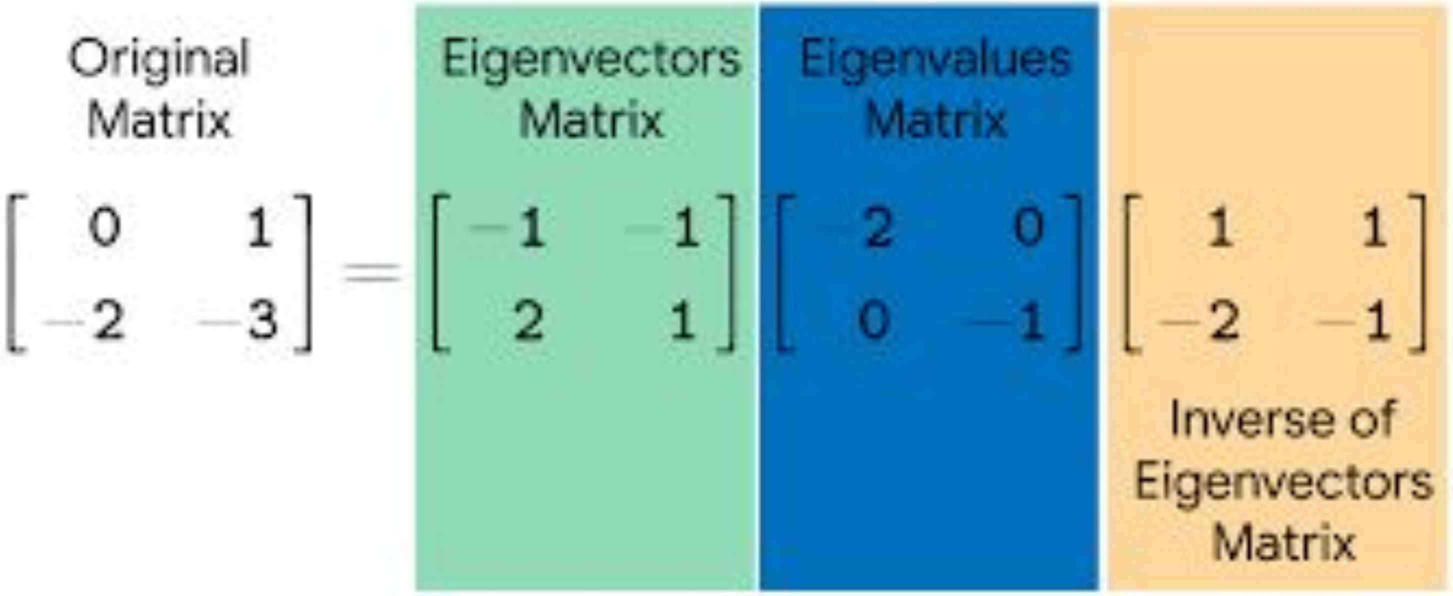
$$M = U \cdot \Sigma \cdot V^*$$

How to compute the SVD:

- U: Solve for eigenvectors and eigenvalues of AA^T: $(AA^T - \lambda I)u = 0$
- V: Solve for eigenvectors of A^TA: $(A^TA - \lambda I)v = 0$
- S is the diagonal matrix containing the square-roots of λ (eigenvalues of AA^T)

SVD example

$$\mathbf{A}_{\boldsymbol{nxp}} = \mathbf{U}_{\boldsymbol{nxn}}\,\mathbf{S}_{\boldsymbol{nxp}}\,\mathbf{V}^{\mathbf{T}}_{\boldsymbol{pxp}}$$



$$\begin{array}{ccccc} \text{Original Matrix} & & \text{Eigenvectors Matrix} & \text{Eigenvalues Matrix} & \\ \begin{bmatrix}0 & 1\\ -2 & -3\end{bmatrix} & = & \begin{bmatrix}-1 & -1\\ 2 & 1\end{bmatrix} & \begin{bmatrix}-2 & 0\\ 0 & -1\end{bmatrix} & \begin{bmatrix}1 & 1\\ -2 & -1\end{bmatrix} \\ & & & & \text{Inverse of Eigenvectors Matrix} \end{array}$$

Matrix Calculus

Derivative w.r.t. a matrix A:

Suppose a function f:$R^{m \times n}$ -> R takes a matrix *A* as inputs

$$\partial_A f(A) \in \mathbb{R}^{m\times n} = \begin{bmatrix} \frac{\partial f(A)}{\partial A_{11}} & \frac{\partial f(A)}{\partial A_{12}} & \cdots & \frac{\partial f(A)}{\partial A_{1n}} \\ \frac{\partial f(A)}{\partial A_{21}} & \frac{\partial f(A)}{\partial A_{22}} & \cdots & \frac{\partial f(A)}{\partial A_{2n}} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f(A)}{\partial A_{m1}} & \frac{\partial f(A)}{\partial A_{m2}} & \cdots & \frac{\partial f(A)}{\partial A_{mn}} \end{bmatrix}$$

Matrix Calculus

Gradient of a function *f:* R^n -> R w.r.t. a vector

$$\partial_x f(x) = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \\ \dots \\ \frac{\partial f(x)}{\partial x_n} \end{bmatrix}$$

Matrix Calculus

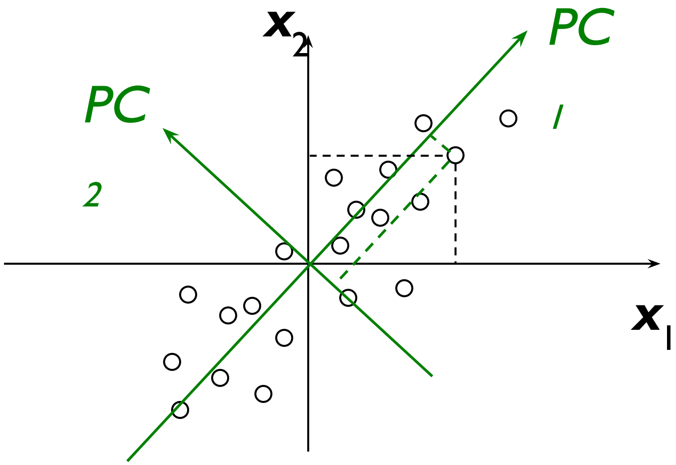
The Hessian of a function *f:* R^n -> R is the matrix of double derivatives

$$\partial_x^2 f(x) \in \mathbb{R}^{n \times n} = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \cdots & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial^2 x_n} \end{bmatrix}$$

Let’s try

Principal Component Analysis

- One of the most widely used feature construction/selection techniques is Principal Component Analysis (PCA)
  - PCA constructs new features that are linear combinations of given features
- Computed eigenvectors and eigenvalues hold useful information
- Often used for dimensionality reduction, finding the intrinsic linear structure in the data



Given features 1 and 2 (x_1, x_2)

Computed features 1 and 2 (green axes)

$$\mathrm{PC}_1 = \underset{y}{\operatorname{argmax}}(y^T X)(X^T y)$$

(maximize variance of points projected onto unit vector y)

A vector projection to another:

$$\mathbf{x}^\top u$$

Now with n vectors

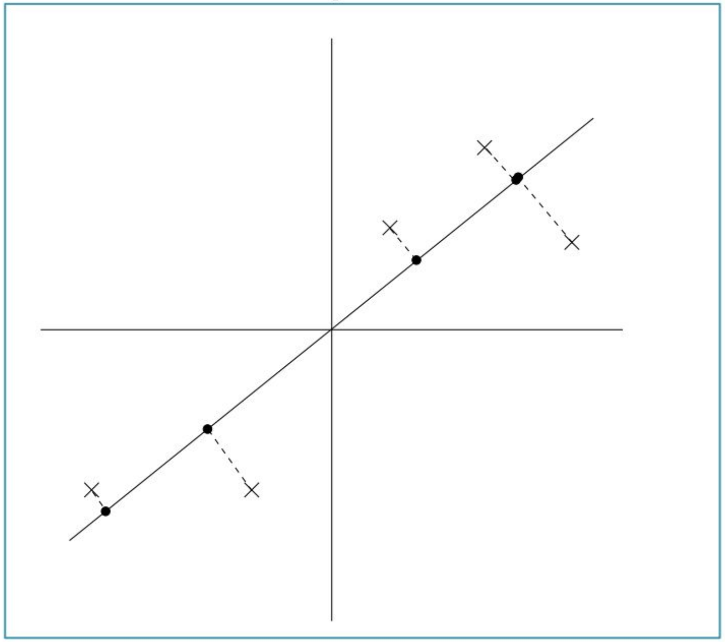
$$\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(n)}$$

Which u represent them the best?

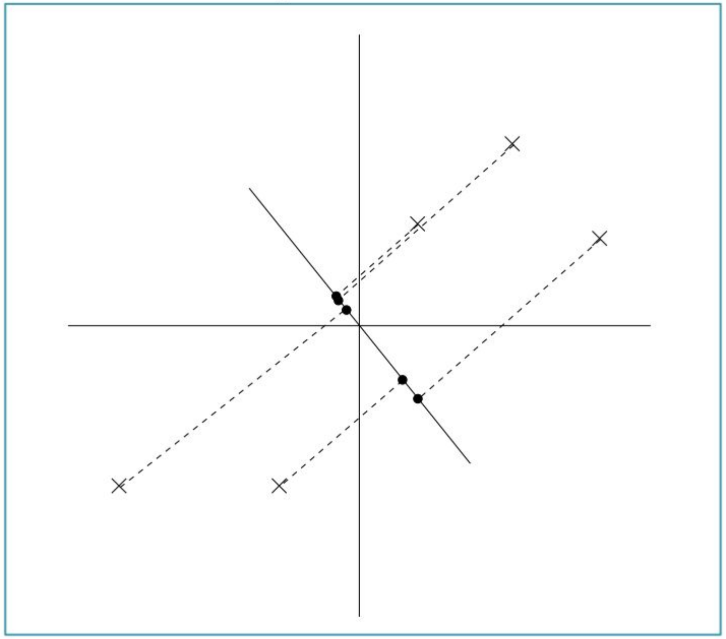
Quiz: maximizing the variance

- Consider the two projections below, which maximizes the data diversity?

Option A



Option B



Idea: project all onto u, but preserve as much variation as possible

Maximize

$$\frac{1}{N}\sum_{n=1}^{N}\left(u^\top \mathbf{x}^{(n)} - u^\top \bar{\mathbf{x}}\right)^2$$

Assume centered, $\mathbf{x} = \mathbf{0}$, so variance becomes

$$\begin{aligned}
\frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{u}^{\mathsf{T}}\mathbf{x}^{(n)}\right)^2 &= \frac{1}{N}\sum_{n=1}^{N}\mathbf{u}^{\mathsf{T}}\mathbf{x}^{(n)} \bullet (\mathbf{x}^{(n)})^{\mathsf{T}}\mathbf{u} \\
&= \mathbf{u}^{\mathsf{T}}\underbrace{\left(\frac{1}{N}\sum_{n=1}^{N}\mathbf{x}^{(n)}(\mathbf{x}^{(n)})^{\mathsf{T}}\right)}_{\text{data covariance matrix, } \mathbf{S}}\mathbf{u} \\
&= \mathbf{u}^{\mathsf{T}}\mathbf{S}\mathbf{u}
\end{aligned}$$

- Constraining $u^\top u = 1$
- Lagrange multiplier (will cover again at SVM)

$$u^\top S u + \lambda(u^\top u - 1)$$

- Optimize (1st order derivatives in u, set to 0)

$$Su = \lambda u$$

- u *is the eigenvector!*

- Top K most important directions?
- Top K eigenvectors!

$$Su_k = \lambda_k u_k$$

- Application: Reduce redundancy (collapse redundant features).
- Finds new way of encoding examples
- Normalize old features

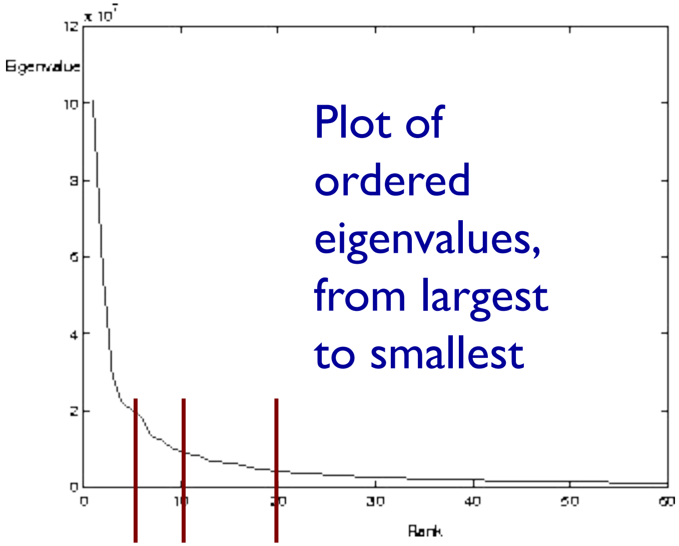
- Uses a linear transformation:
- New features are projections (how much you weight on each direction)

$$x^\top u_1, x^\top u_2, ..., x^\top u_K$$

- Projecting gives K new features

PCA for dimensionality reduction

- So the eigenvalues can give clues to the intrinsic dimensionality of the data, or at least provide a way to more efficiently approximate high-dimensional data with lower-dimensional feature vectors
- For example:



Plot of ordered eigenvalues, from largest to smallest

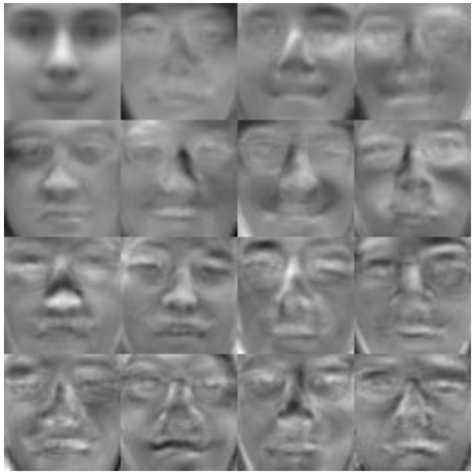
60-dimensional data (60 eigenvectors and eigenvalues)

Many of the eigenvalues are small, meaning that their associated eigenvectors don't contribute much to the representation of the data

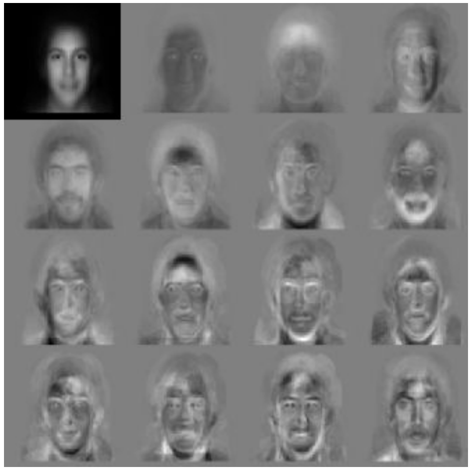
We can choose a cutoff – say, only use the first 20 eigenvectors (or 10, or 5)

Face recognition via "Eigenfaces"

- A well-known technique for face recognition based on computing eigenvectors of a training set of face images, i.e., Eigenfaces



Eigenfaces 1

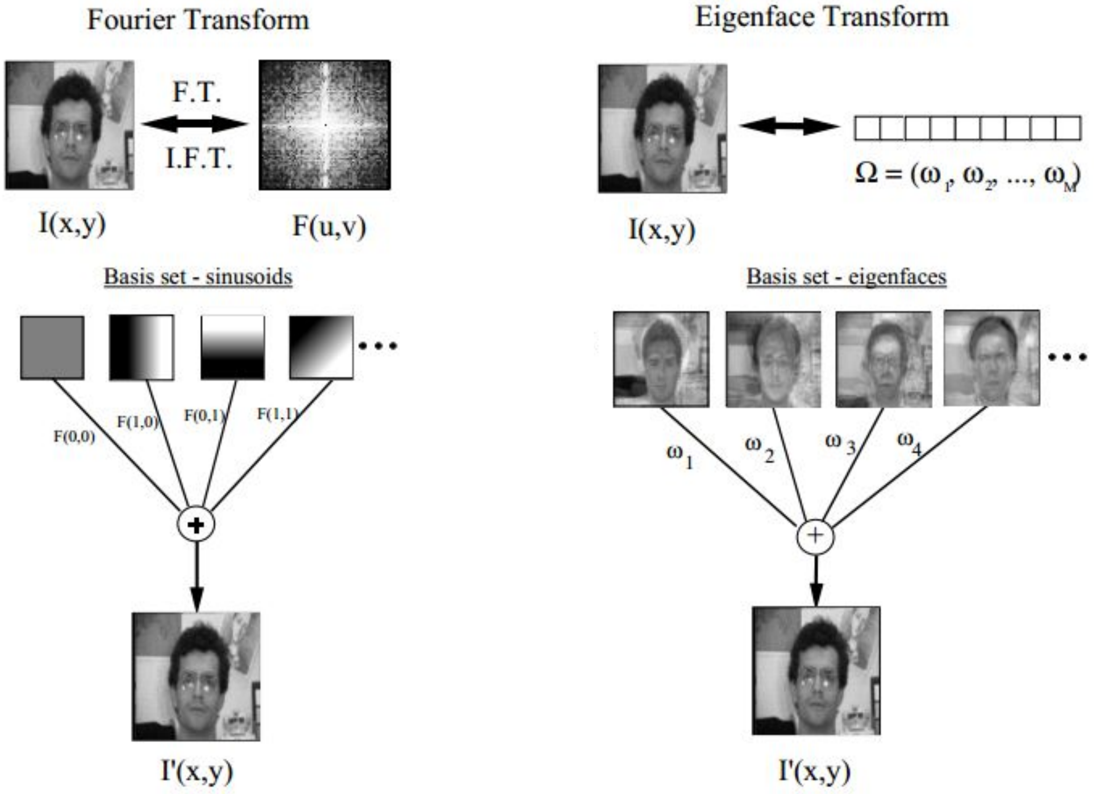


Eigenfaces 2

Keep in mind: an image is just an N-dimensional point or vector (where N = rows × cols)

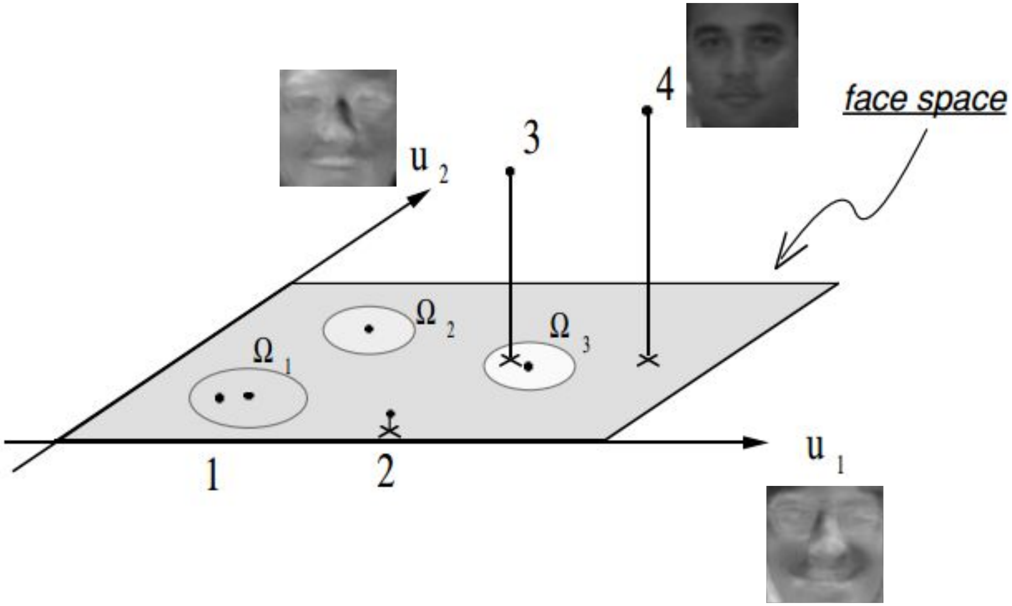
Face recognition via "Eigenfaces"

- Eigenvectors (eigenfaces) can be thought of as *basis vectors* for reconstructing data (face images)



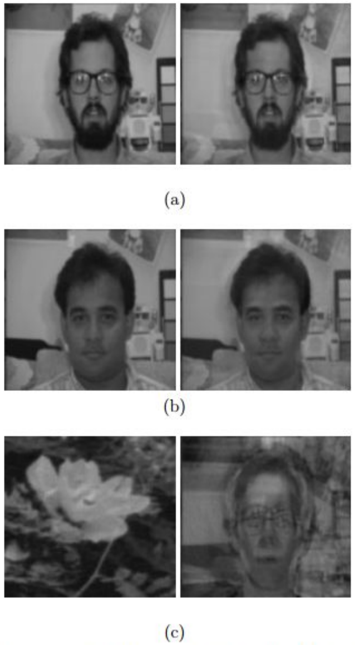
Face recognition via “Eigenfaces”

- The Eigenfaces span a (relatively) low-dimensional *face space*, representing all possible face images
- A new (unknown) face image is projected into the face space (reconstructed by the Eigenfaces)
  - The distance between the face image and its reconstruction is the *distance from face space*



Unknown face

Projection into face space

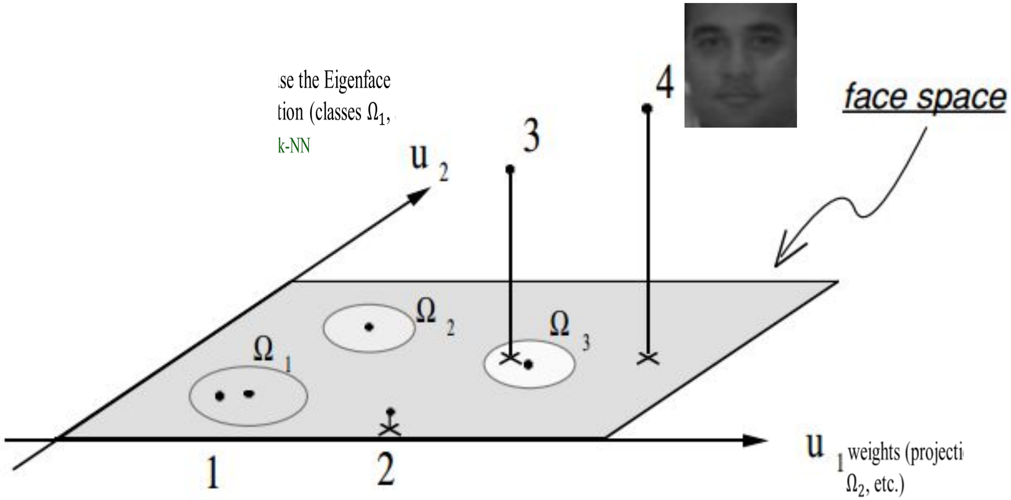

(a)

(b)

(c)

Face recognition via "Eigenfaces"

- The *distance from face space* measure could be used for face detection: Does this image (or part of an image) look like a face?
- If yes, then use the Eigenface weights (projections) as features for classification (classes Ω_1, Ω_2, etc.)
  - E.g., using k-NN



JOIN QUIZ

https://quizizz.com/join

Please use your full name at sign-in, not a nickname