18-661: Introduction to ML for Engineers

Probability, MLE/MAP, and Linear Algebra

Fall 2020

ECE – Carnegie Mellon University

Announcements

- The math quiz scores will be posted on Gradescope later this week – Entry Code: 9E882Y
- Recitation this Friday will go through the math quiz
- HW1 will be posted today, due on Fri Sept 11

Course Prerequisites

- Probability theory
- Linear algebra
- Calculus: Differentiation, Integration, Convexity
- Python programming, in particular, numpy

If you don't satisfy these pre-requisites, we strongly encourage you to take the class after reviewing introductory material

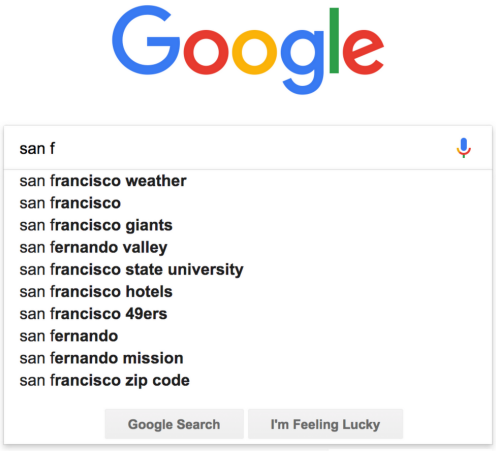
Please take the math quiz to assess your preparedness for this class

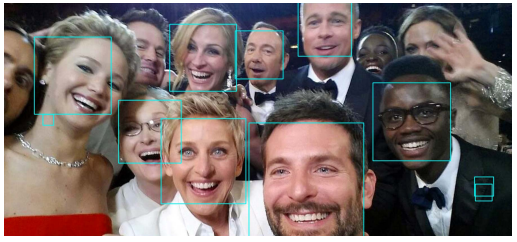
Outline

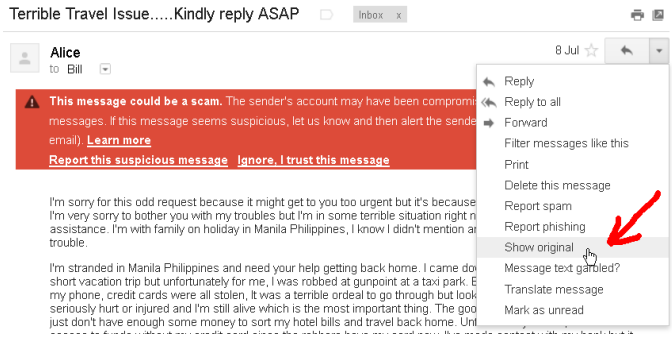
1. Recap: What is Machine Learning?
2. Probability Review
3. A Simple Learning Problem: MLE/MAP Estimation
4. Linear Algebra Review

Recap: What is Machine Learning?

Definition and Examples

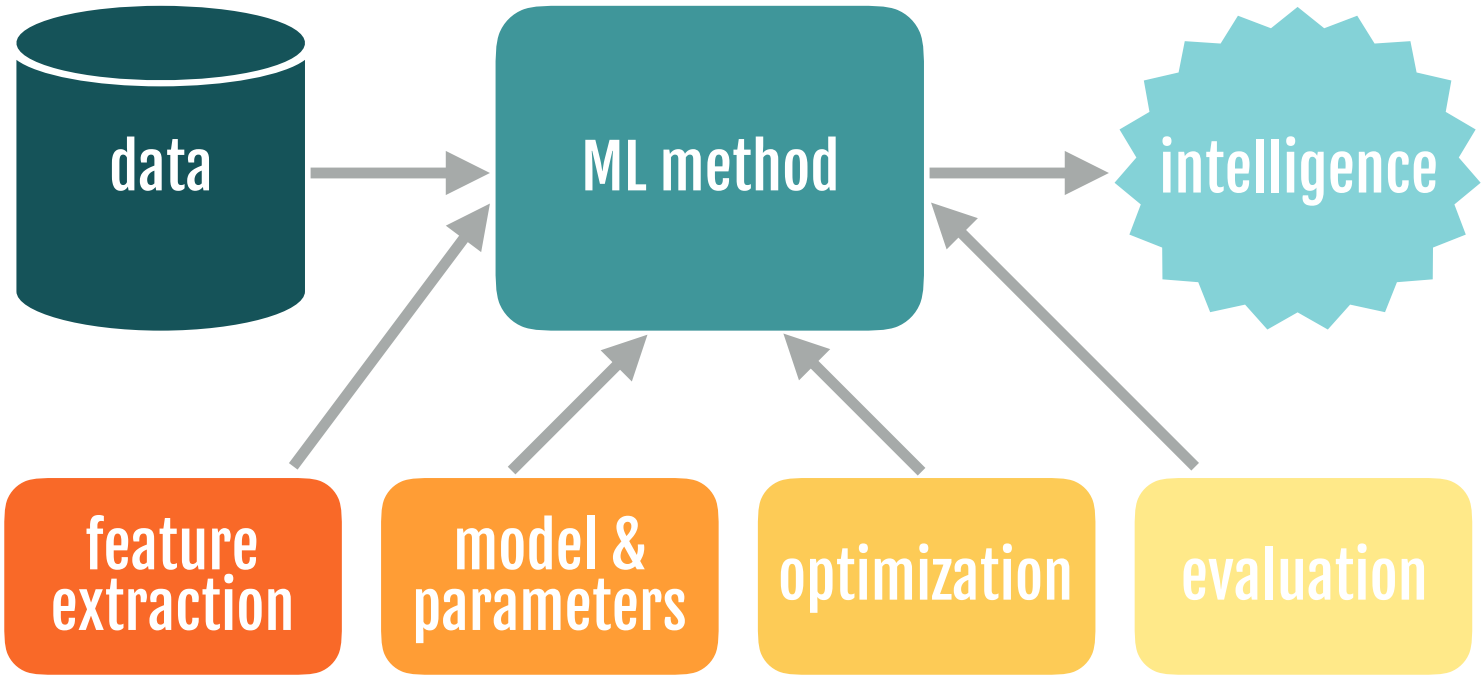






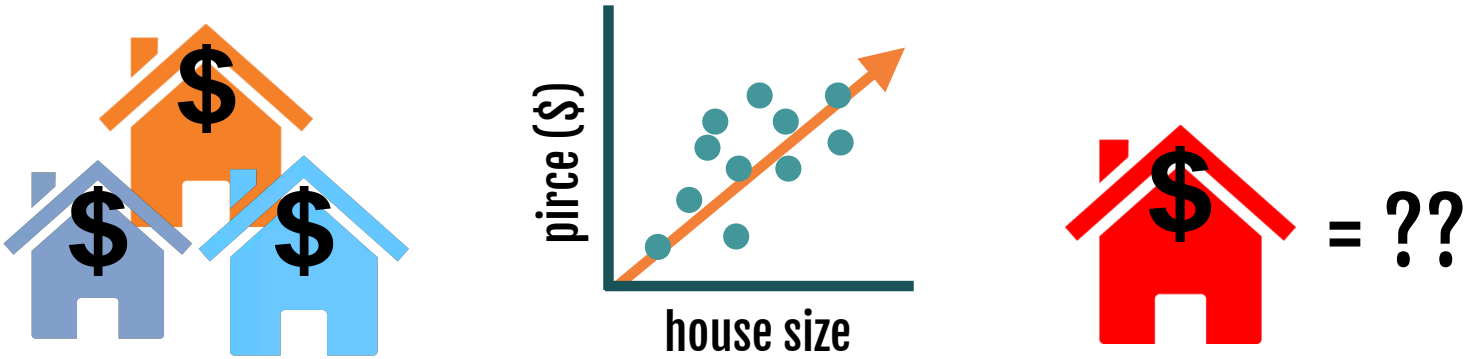
- **Machine learning is**: the study of methods that *improve their performance on some task with experience*

Goal: Choose the Right ML Method for a Given Task



Task 1: Regression

How much should you sell your house for?

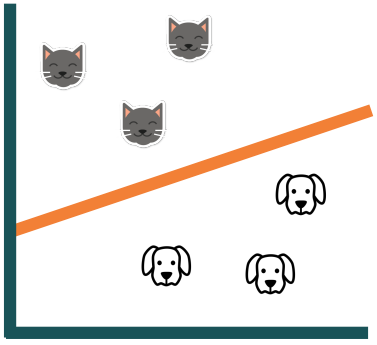


input: houses & features **learn**: $x \to y$ relationship **predict**: y (*continuous*)

Course Covers: Feature Scaling, Linear/Ridge Regression, Loss Function, SGD, Regularization, Cross Validation

Task 2: Classification

Cat or dog?



= ??

input: cats and dogs

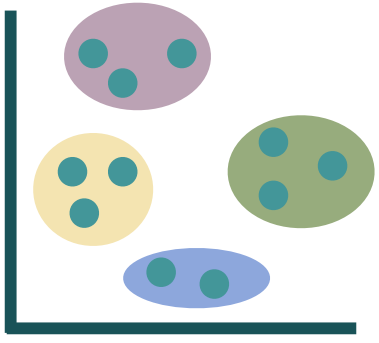
learn: $x \rightarrow y$ relationship

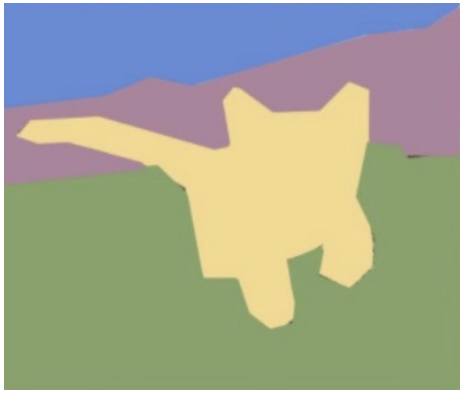
predict: y (*categorical*)

Course Covers: Naive Bayes, Logistic Regression, SVMs, Neural Nets, Nearest Neighbors, Decision Trees, Boosting

Task 3: Clustering

How to segment an image?



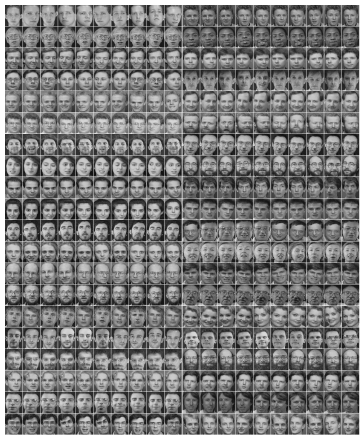


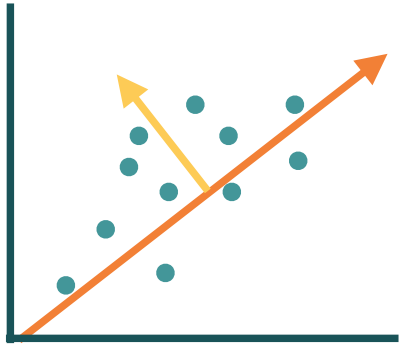
input: raw pixels $\{x\}$ **separate**: $\{x\}$ into sets **output**: cluster labels $\{z\}$

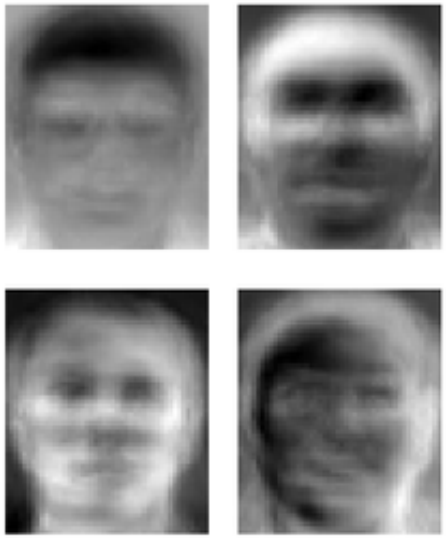
Course Covers: K-means clustering, GMMs

Task 4: Embedding

How to reduce size of dataset?







input: large dataset $\{x\}$ **find**: sources of variation **return**: representation $\{z\}$

Course Covers: Dimensionality Reduction, PCA

Outline

1. Recap: What is Machine Learning?
2. Probability Review
3. A Simple Learning Problem: MLE/MAP Estimation
4. Linear Algebra Review

Probability Review

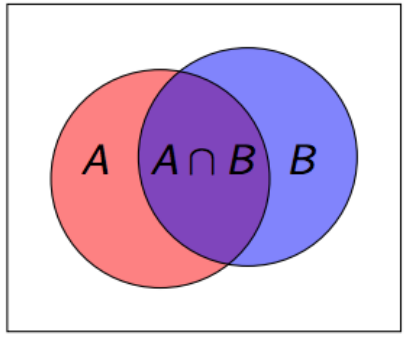
Probability Terminology

Name	Type	Symbol	Meaning
Sample Space	set	Ω, S	possible outcomes
Event Space	a set of subsets (of Ω)	$\mathcal{F}$, E	the events that have probabilities
Probability Measure	measure	P, π	assigns probabilities to events
Probability Space	a triple	$(\Omega, \mathcal{F}, P)$	

Examples:

- Rolling a fair die
  - Ω: $\{1, 2, 3, 4, 5, 6\}$
  - $\mathcal{F} = \{\{1\}, \{2\}, \ldots, \{1, 2\}, \ldots, \{1, 2, 3\}, \ldots, \{1, 2, 3, 4, 5, 6\}, \{\}\}$
  - $P(\text{rolling an odd number}) = P(\{1, 3, 5\}) = \frac{1}{2}$
- Tossing a fair coin twice
  - Ω: $\{HH, HT, TH, TT\}$
  - $\mathcal{F} = \{\{HH\}, \{HT\}, \ldots, \{HH, HT\}, \ldots, \{HH, HT, TH, TT\}, \{\}\}$
  - $P(\text{first flip is heads}) = P(\{HH\}, \{HT\}) = \frac{1}{2}$

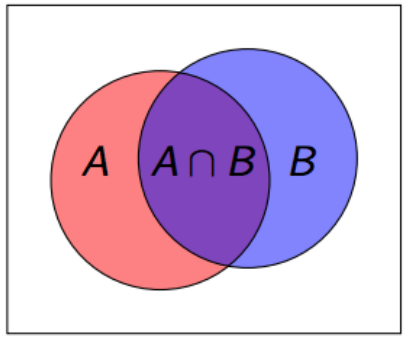
Axioms of Probability



- $0 \leq P(A) \leq 1$
- $P(\Omega) = 1$, $P(\varnothing) = 0$
- $P(A \cup B) = P(A) + P(B) - P(A \cap B)$
- Question: For two tosses of a fair coin, suppose A is the event that at least one is H, and B is the event that there is exactly one T. Then what is $P(A \cup B)$?

$$P(A \cup B) = 0.75 + 0.5 - 0.5 = 0.75$$

Conditional Probability (Bayes Rule)



- For events $A, B \in \mathcal{F}$, the conditional probability of A given B is given by:

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

- Question: For two tosses of a fair coin, what is the probability of at least one T, given that the event TT did not occur? ANS: 2/3
- Bayes rule:

$$P(B \mid A)P(A) = P(A \cap B) = P(A \mid B)P(B)$$

$$\implies P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Some Other Concepts that You Should Know

- Discrete and Continuous Random Variables
- PMF, PDF, CDF
- Expectation and Variance
- Entropy

Some of these will be covered in HW1

A Simple Learning Problem: MLE/MAP Estimation

Dogecoin

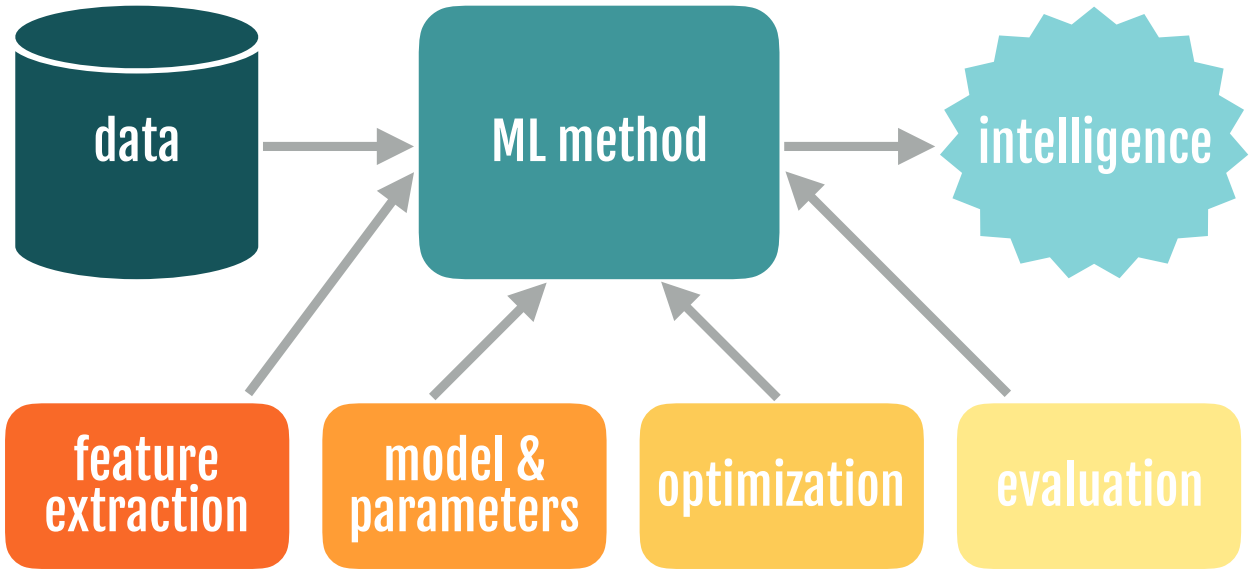
- Scenario: You find a coin on the ground.

- *You ask yourself: Is this a fair or biased coin? What is the probability that I will flip a heads?*

- You flip the coin 10 times . . .
- It comes up as 'H' 8 times and 'T' 2 times

- Can we learn the bias of the coin from this data?

Recall: Machine Learning Pipeline



Two approaches that we will discuss today:

- Maximum likelihood Estimation (MLE)
- Maximum a posteriori Estimation (MAP)

Maximum Likelihood Estimation (MLE)

- **Data**: Observed set D of n_H heads and n_T tails
- **Model**: Each flip follows a Bernoulli distribution

$$P(H) = \theta,\ P(T) = 1 - \theta,\ \theta \in [0, 1]$$

Thus, the likelihood of observing sequence D is

$$P(D \mid \theta) = \theta^{n_H}(1 - \theta)^{n_T}$$

- Question: Given this model and the data we've observed, can we calculate an estimate of θ?
- **MLE**: Choose θ that maximizes the *likelihood* of the observed data

$$\hat{\theta}_{MLE} = \arg\max_{\theta} P(D \mid \theta)$$

How to solve?

- $\log(x)$ is a monotone increasing function; will not affect the arg max

$$\begin{aligned}
\hat{\theta}_{MLE} &= \arg\max_{\theta} P(D \mid \theta) \\
&= \arg\max_{\theta} \log P(D \mid \theta) \\
&= \arg\max_{\theta} \log\left(\theta^{n_H}(1-\theta)^{n_T}\right) \\
&= \arg\max_{\theta} \underbrace{n_H \log(\theta) + n_T log(1-\theta)}_{\text{concave}}
\end{aligned}$$

- Take derivative $\frac{\partial}{\partial\theta} \log P(D \mid \theta)$ and set equal to zero

$$\begin{aligned}
0 &= \frac{\partial}{\partial\theta} n_H \log(\theta) + n_T log(1-\theta) \\
&= \frac{n_H}{\theta} - \frac{n_T}{1-\theta} \\
\implies \hat{\theta}_{MLE} &= \frac{n_H}{n_H + n_T}
\end{aligned}$$

Going back to our scenario

- You flip the coin 10 times ...
- It comes up as 'H' 8 times and 'T' 2 times

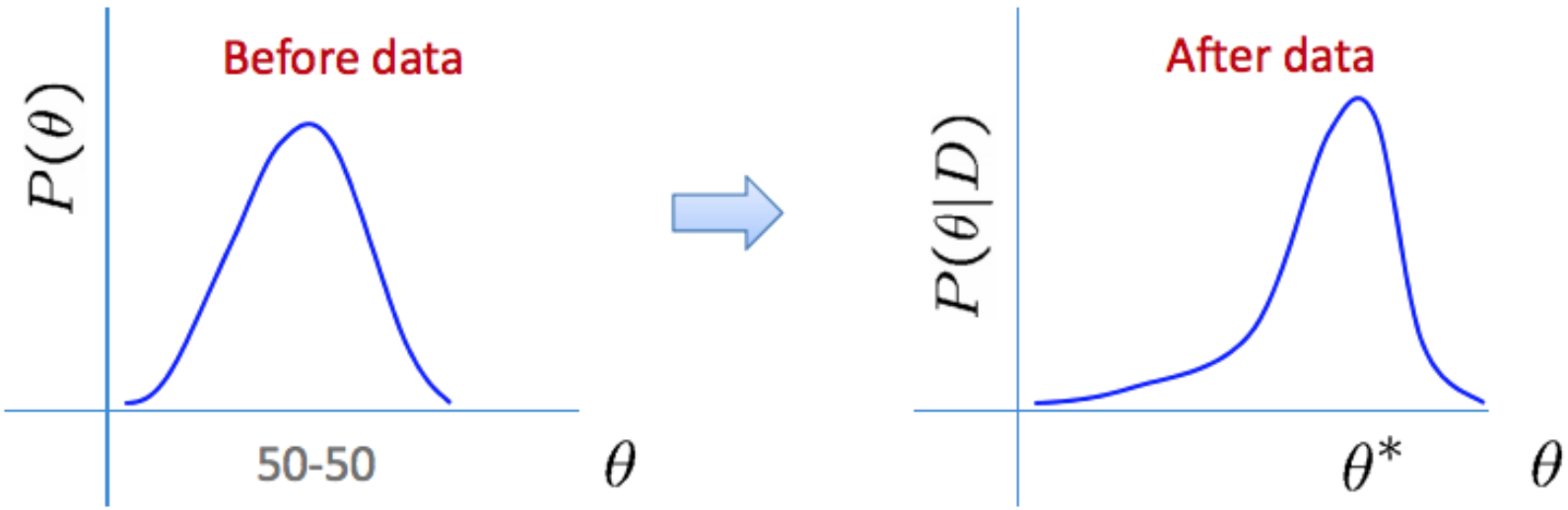
- Can we learn the bias θ of the coin from this data?

$$\hat{\theta}_{MLE} = \frac{n_H}{n_H + n_T} = 0.8$$

Here, we are trusting the data completely. But there could be too little data or noisy data

What about prior knowledge?

- We believe the coin is *supposed* to be close to 50-50
- Rather than completely "trusting" the data as-is, we want to use the data to update our prior beliefs



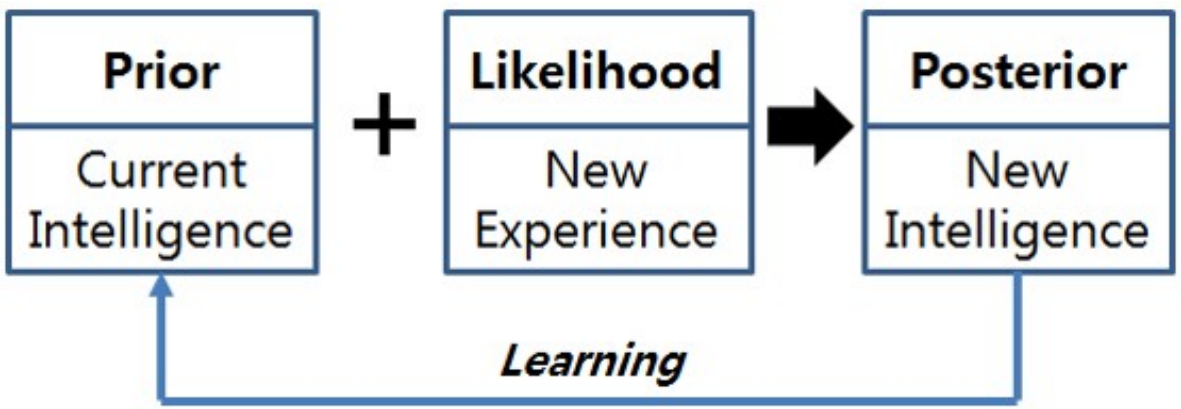
Bayesian Learning

- How to incorporate prior knowledge?
- Use Bayes' Rule:

$$P(\theta \mid D) = \frac{P(D \mid \theta)P(\theta)}{P(D)}$$

- Or, equivalently:

$$\underset{\text{posterior}}{P(\theta \mid D)} \propto \underset{\text{likelihood}}{P(D \mid \theta)} \underset{\text{prior}}{P(\theta)}$$



MAP for Dogecoin

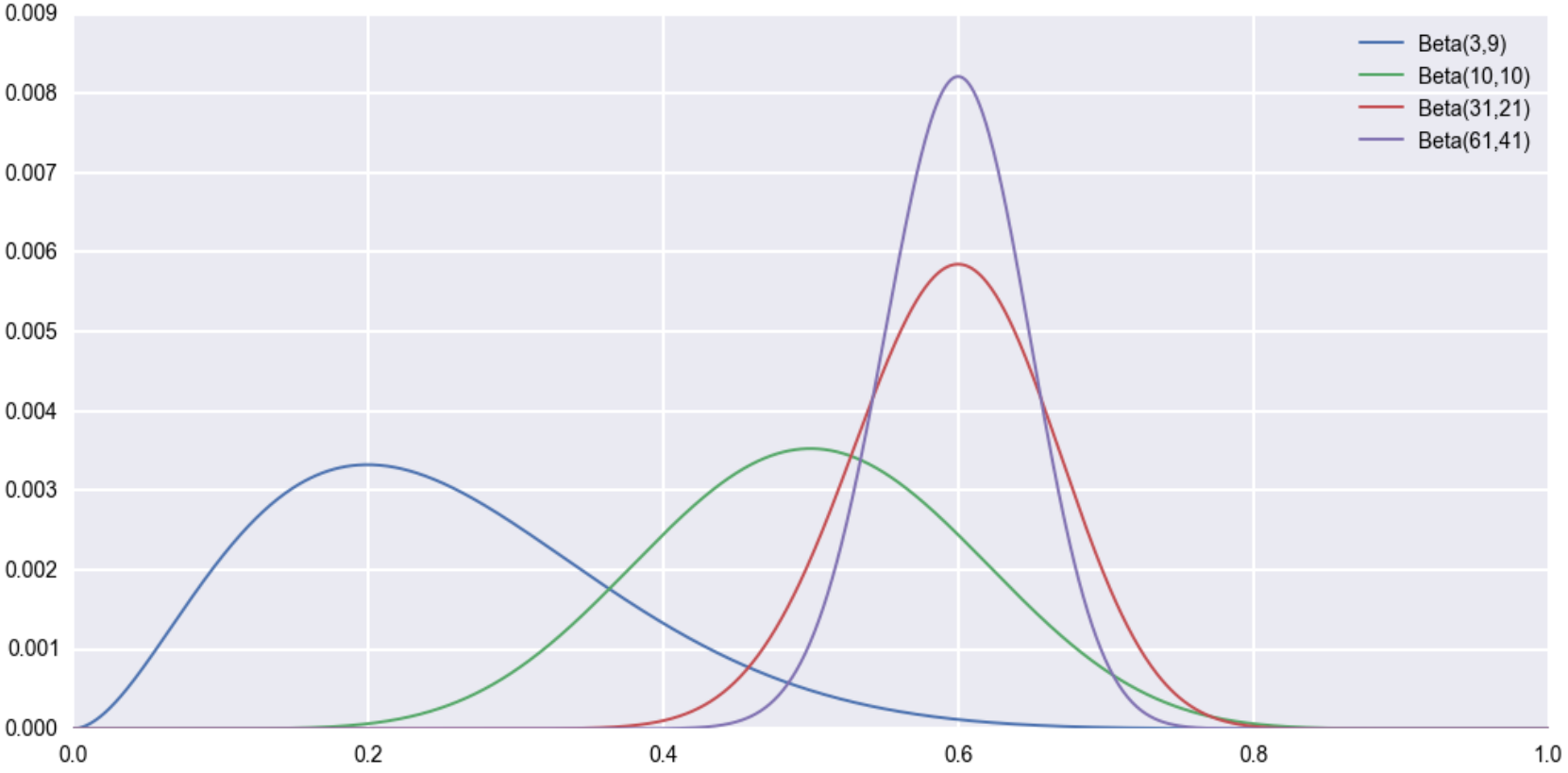
$$\hat{\theta}_{MAP} = \arg\max_{\theta} P(D \mid \theta) P(\theta)$$

- Recall that $P(D \mid \theta) = \theta^{n_H}(1-\theta)^{n_T}$
- How should we set the prior, $P(\theta)$?
- Common choice for a binomial likelihood is to use the Beta distribution, $\theta \sim Beta(\alpha, \beta)$:

$$P(\theta) = \frac{1}{B(\alpha,\beta)} \theta^{\alpha-1}(1-\theta)^{\beta-1}, \text{where } B(\alpha,\beta) = \int_0^1 \theta^{\alpha-1}(1-\theta)^{\beta-1} d\theta$$

- Interpretation: α = number of expected heads, β = number of expected tails. Larger value of $\alpha + \beta$ denotes more confidence (and smaller variance).

Beta Distribution



$\frac{\alpha}{\beta}$ controls left/right bias, $\alpha + \beta$ controls height of peak

MAP for Dogecoin

- A benefit of using the *Beta* distribution as a prior is that the posterior will also be a *Beta* distribution:

$$\begin{aligned}\hat{\theta}_{MAP} &= \arg\max_{\theta} P(D \mid \theta) P(\theta) \\ &= \arg\max_{\theta} \theta^{\alpha + n_H - 1} (1-\theta)^{\beta + n_T - 1} \\ &= \frac{\alpha + n_H - 1}{\alpha + \beta + n_H + n_T - 2}\end{aligned}$$

- Note that as $n_H + n_T \to \infty$, the effect of the prior disappears and we recover the MLE estimate

Putting it all together

$$\hat{\theta}_{MLE} = \frac{n_H}{n_H + n_T}$$

$$\hat{\theta}_{MAP} = \frac{\alpha + n_H - 1}{\alpha + \beta + n_H + n_T - 2}$$

- Suppose $\theta^* := 0.5$ and we observe: $D = \{H, H, T, T, T, T\}$
- Scenario 1: We assume $\theta \sim Beta(4, 4)$. Which is more accurate – θ_{MLE} or θ_{MAP}?
  - $\theta_{MAP} = 5/12$, $\theta_{MLE} = 1/3$
- Scenario 2: We assume $\theta \sim Beta(1, 7)$. Which is more accurate – θ_{MLE} or θ_{MAP}?
  - $\theta_{MAP} = 1/6$, $\theta_{MLE} = 1/3$

Bayesians vs. Frequentists

Why was this a ML problem?

Machine learning is: the study of methods that

improve their performance (the accuracy of the predicted probability)

on some task (predicting the probability of 'heads')

with experience (the more coin flips we see, the better our guess)

Learning involves ...

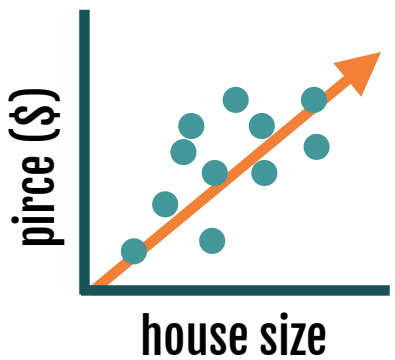
- Collect some data
  - e.g., coin flips
- Set up the problem: Choose a model / loss function
  - e.g., bernoulli model, data likelihood/ a posteriori prob.
- Solve the problem: Choose an optimization procedure
  - e.g., set derivative of log to zero and solve to find MLE/MAP

Key idea: these are *choices*. It's important to understand the implications of these choices and evaluate their trade-offs for the problem at hand.

Linear Algebra Review

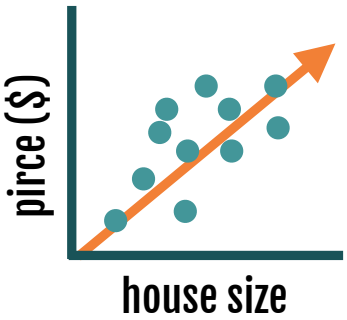
Recall: Task 1: Regression

How much should you sell your house for?



input: houses & features **learn**: $x \rightarrow y$ relationship **predict**: y (*continuous*)

Data Can be Compactly Represented by Matrices



- Learn parameters (w_1, w_0) of the orange line $y = w_1 x + w_0$

Sq.ft

$$\text{House 1: } 1000 \times w_1 + w_0 = 200,000$$

$$\text{House 2: } 2000 \times w_1 + w_0 = 350,000$$

- Can represent compactly in matrix notation

$$\begin{bmatrix} 1000 & 1 \\ 2000 & 1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_0 \end{bmatrix} = \begin{bmatrix} 200,000 \\ 350,000 \end{bmatrix}$$

Some Concepts That You Should Know

- Invertibility of Matrices and Computing Inverses
- Vector Norms – L2, Frobenius etc., Inner Products
- Eigenvalues and Eigen-vectors
- Singular Value Decomposition
- Covariance Matrices and Positive Semi-definite-ness

Excellent Resources:

- Essence of Linear Algebra YouTube Series by 3Blue1Brown
- Prof. Gilbert Strang's course at MIT

Matrix multiplication

- For two matrices $A \in \mathbb{R}^{m \times n}, B \in \mathbb{R}^{n \times p}$, their product is:

$$AB = C \in R^{m \times p} \iff C_{ij} = \sum_{k=1}^{n} A_{ik} B_{kj}$$

- Multiplication is undefined with the number of columns in $A \neq$ the number of rows in B (except in case: cA where $c \in \mathbb{R}$ is a scalar)
- Special cases:
  - Inner product: $x, y \in \mathbb{R}^n, \quad x^\top y \in \mathbb{R} = \sum_{i=1}^{n} x_i y_i$
  - Matrix-vector product: $A \in \mathbb{R}^{m \times n}, x \in \mathbb{R}^n \iff Ax \in \mathbb{R}^m$

$$A = \begin{bmatrix} | & | & & | \\ a_1 & a_2 & \dots & a_n \\ | & | & & | \end{bmatrix}, Ax \in \mathbb{R}^m = \sum_{i=1}^{n} a_i x_i$$

Important properties

- Associative: $A(BC) = (AB)C$
- Distributive: $A(B + C) = AB + AC$
- *Not* Commutative: $AB \neq BA$
- Transpose: $(AB)^\top = B^\top A^\top$

Matrix Inverse

- The *inverse* of a matrix $A \in \mathbb{R}^{n \times n}$ is a matrix $A^{-1} \in \mathbb{R}^{n \times n}$ such that:

$$AA^{-1} = A^{-1}A = I_n$$

- If A^{-1} exists, then A is called invertible or non-singular
- Matrix A is invertible iff $\det(A) \neq 0$
- Let us solve the house-price prediction problem

$$\begin{bmatrix} 1000 & 1 \\ 2000 & 1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_0 \end{bmatrix} = \begin{bmatrix} 200,000 \\ 350,000 \end{bmatrix}$$

Matrix Inverse

- Let us solve the house-price prediction problem

$$\begin{bmatrix} 1000 & 1 \\ 2000 & 1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_0 \end{bmatrix} = \begin{bmatrix} 200,000 \\ 350,000 \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} w_1 \\ w_0 \end{bmatrix} = \left(\begin{bmatrix} 1000 & 1 \\ 2000 & 1 \end{bmatrix} \right)^{-1} \begin{bmatrix} 200,000 \\ 350,000 \end{bmatrix} \tag{2}$$

$$= \frac{1}{-1000} \begin{bmatrix} 1 & -1 \\ -2000 & 1000 \end{bmatrix} \begin{bmatrix} 200,000 \\ 350,000 \end{bmatrix} \tag{3}$$

$$= \frac{1}{-1000} \begin{bmatrix} -150,000 \\ -5 \times 10^7 \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} w_1 \\ w_0 \end{bmatrix} = \begin{bmatrix} 150 \\ 50,000 \end{bmatrix} \tag{5}$$

Norms and Loss Functions

- You could have data from many houses

$$\underbrace{\begin{bmatrix} 1000 & 1 \\ 2000 & 1 \\ 1500 & 1 \\ \vdots & \vdots \\ 2500 & 1 \end{bmatrix}}_{A} \times \underbrace{\begin{bmatrix} w_1 \\ w_0 \end{bmatrix}}_{w} = \underbrace{\begin{bmatrix} 200,000 \\ 350,000 \\ 300,000 \\ \vdots \\ 450,000 \end{bmatrix}}_{y}$$

- There isn't a $w = [w_1, w_0]^T$ that will satisfy all equations
- Want to find w that minimizes the difference between Aw, y
- But since this a vector, we need an operator that can map the vector $y - Aw$ to a scalar

Norms and Loss Functions

- A vector norm is any function $f : \mathbb{R}^n \to \mathbb{R}$ with
  - $f(x) \geq 0$ and $f(x) = 0 \iff x = 0$
  - $f(ax) = |a| f(x)$ for $a \in \mathbb{R}$
  - $f(x + y) \leq f(x) + f(y)$
- e.g., ℓ_2 norm: $\|x\|_2 = \sqrt{x^\top x} = \sqrt{\sum_{i=1}^n x_i^2}$
- e.g., ℓ_1 norm: $\|x\|_1 = \sum_{i=1}^n |x_i|$
- Question: What is the ℓ_1 norm of $y - Aw$ for the following problem?

$$\underbrace{\begin{bmatrix} 1 & 1 \\ 2 & 1 \\ 1.5 & 1 \\ 2.5 & 1 \end{bmatrix}}_{A} \times \underbrace{\begin{bmatrix} 1.5 \\ 0.5 \end{bmatrix}}_{w} = \underbrace{\begin{bmatrix} 2 \\ 3.5 \\ 3 \\ 4.5 \end{bmatrix}}_{y}$$

- Answer: $\|y - Aw\|_1 = 0.5$

Matrix as Linear Transformation (see 3Blue1Brown)

- How exactly does square matrix multiplication transform vectors?
- It's columns correspond to re-scaled unit vectors

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

- Now we can express any vector as a linear combination of the above matrix-unit-vector products

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = 2 \begin{bmatrix} 1 \\ 4 \end{bmatrix} + 1 \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

$$= \begin{bmatrix} 4 \\ 11 \end{bmatrix}$$

Eigenvalues and Eigenvectors

- For $A \in \mathbb{R}^{n \times n}$, λ is an eigenvalue and $x \neq 0$ is an eigenvector if $Ax = \lambda x$.
- Eigenvalues are the roots of $\det(A - \lambda I_n) = 0$
- Eigenvalues are non-zero solutions of $Ax = \lambda x$
- Viewing A as a linear transformation
  - The vectors remain unchanged and only get re-scaled are the eigen-vectors.
  - Their scaling factors are the eigen-values!
- Question: Find the eigen-values and eigen-vectors of

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix}$$

Eigenvalues and Eigenvectors

- Question: Find the eigen-values and eigen-vectors of

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix}$$

- Eigen-values:

$$\det\left(\begin{bmatrix} 1-\lambda & 2 \\ 4 & 3-\lambda \end{bmatrix}\right) = 0$$

$$(1-\lambda)(3-\lambda) - 8 = 0$$

$$\lambda^2 - 4\lambda - 5 = 0$$

$$(\lambda - 5)(\lambda + 1) = 0$$

$$\lambda = 5, \lambda = -1$$

Eigenvalues and Eigenvectors

- Question: Find the eigen-values and eigen-vectors of

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix}$$

- Eigen-values: $\lambda = 5, \lambda = -1$
- Eigen-vectors:

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 5 \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \implies \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = -1 \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \implies \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Eigen-Value Decomposition

- Group the eigen-vectors and eigen values into the following matrices.

$$P = \begin{bmatrix} 1 & -1 \\ 2 & 1 \end{bmatrix}$$

$$\Lambda = \begin{bmatrix} 5 & 0 \\ 0 & -1 \end{bmatrix}$$

- If the eigen-vectors are linearly independent, we can express A as

$$\begin{aligned} A &= P\Lambda P^{-1} \\ &= \begin{bmatrix} 1 & -1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 5 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 2 & 1 \end{bmatrix}^{-1} \end{aligned}$$

Eigen-Value Decomposition

- Why is this useful?
- Suppose we want to find powers of A, eg. A^4
- One option, that is quite tedious is:

$$A^4 = \begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix}$$

- Instead we could use the eigen-value decomposition

$$\begin{aligned} A^4 &= P\Lambda P^{-1} P\Lambda P^{-1} P\Lambda P^{-1} P\Lambda P^{-1} \\ &= P\Lambda^4 P^{-1} \\ &= \begin{bmatrix} 1 & -1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 5^4 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 2 & 1 \end{bmatrix}^{-1} \end{aligned}$$

Singular value decomposition (SVD)

- EVD only works for square, diagonalizable matrices
- SVD works for matrices of any size! It decomposes $A \in \mathbb{R}^{m \times n}$ as follows.

$$A = U \Sigma V^\top,$$

- $U \in \mathbb{R}^{m \times m}$, $V \in \mathbb{R}^{n \times n}$ are orthogonal matrices (*i.e.* $U^\top = U^{-1}$)
- $\Sigma \in \mathbb{R}^{m \times n}$ is a diagonal matrix with *singular values* of A denoted by σ_i appearing by non-increasing order: $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{\min(m,n)} \geq 0$.
- The square singular values of A are the eigenvalues of the matrix $AA^\top$ or $A^\top A$, i.e., $\sigma_i(A) = \sqrt{\lambda_i(AA^\top)} = \sqrt{\lambda_i(A^\top A)}$
- V is the matrix of eigen-vectors of $A^T A$
- U is the matrix of eigen-vectors of AA^T

Outline

1. Recap: What is Machine Learning?
2. Probability Review
3. A Simple Learning Problem: MLE/MAP Estimation
4. Linear Algebra Review

Announcements

- Graded quizzes will be posted on Gradescope – Entry Code: 9E882Y
- Recitation this Friday will go through the math quiz
- HW1 will be posted today, due on Fri Sept 11