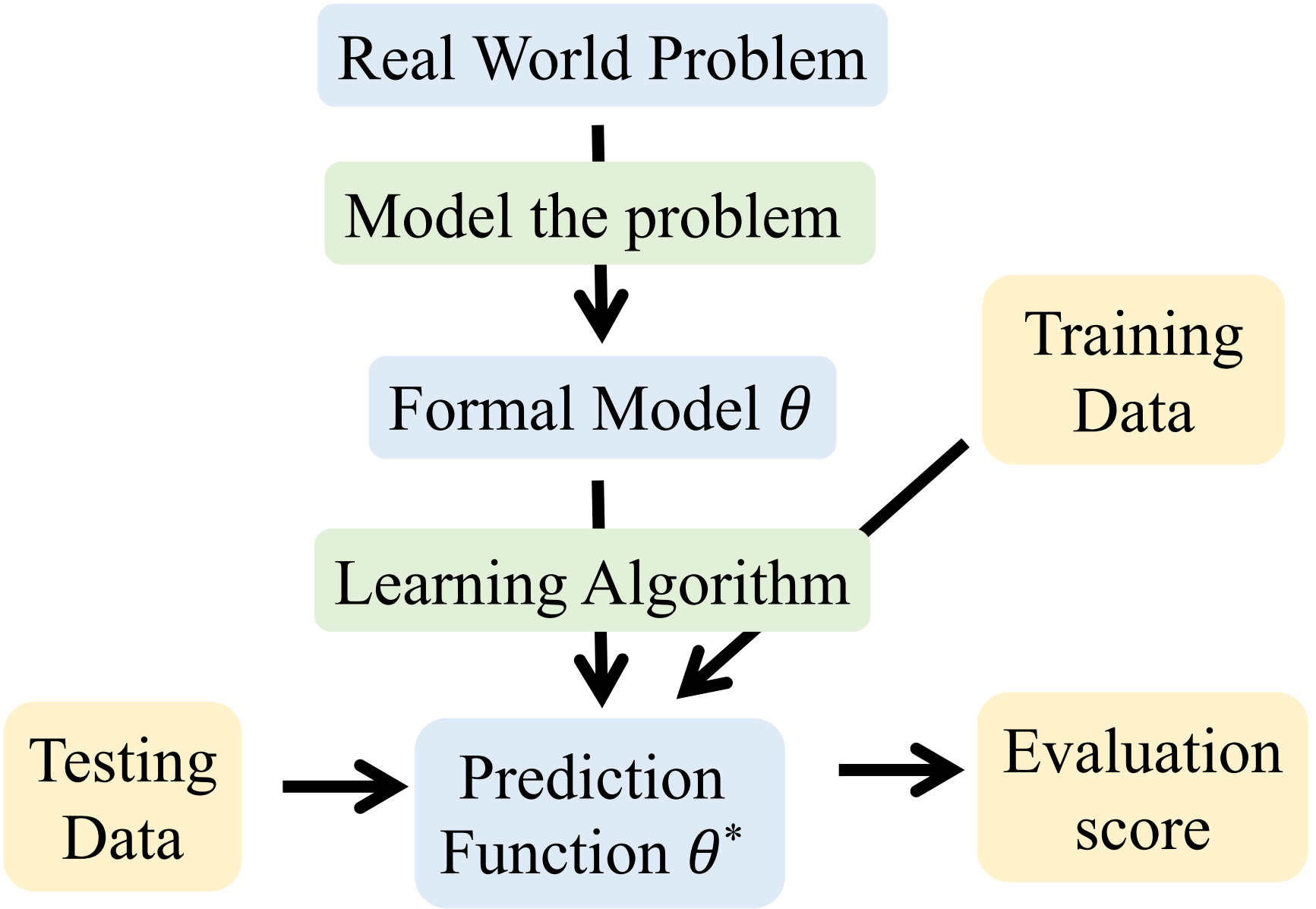
Gradient Ascent

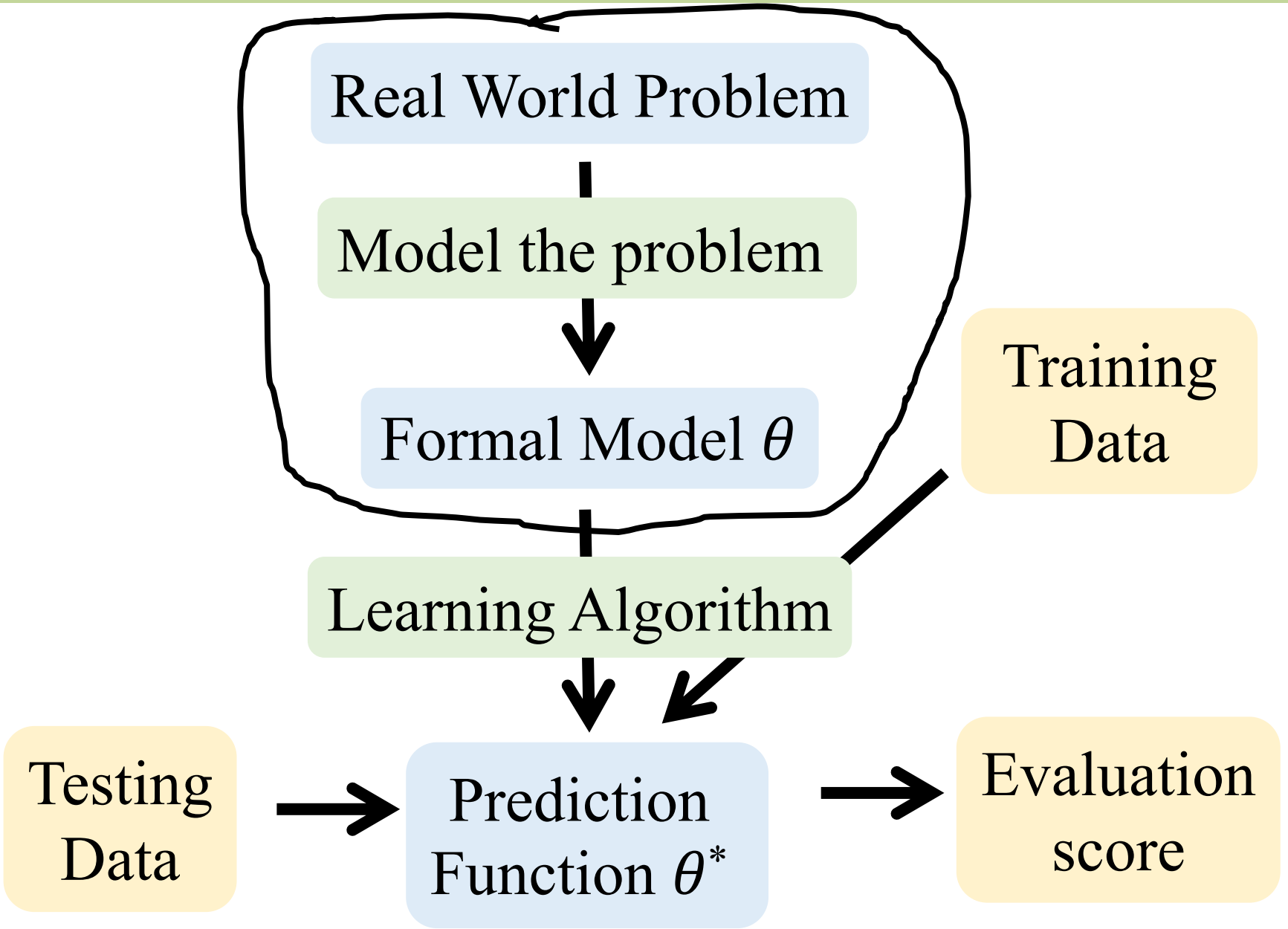
Noah Arthurs
CS109, Stanford University

Review

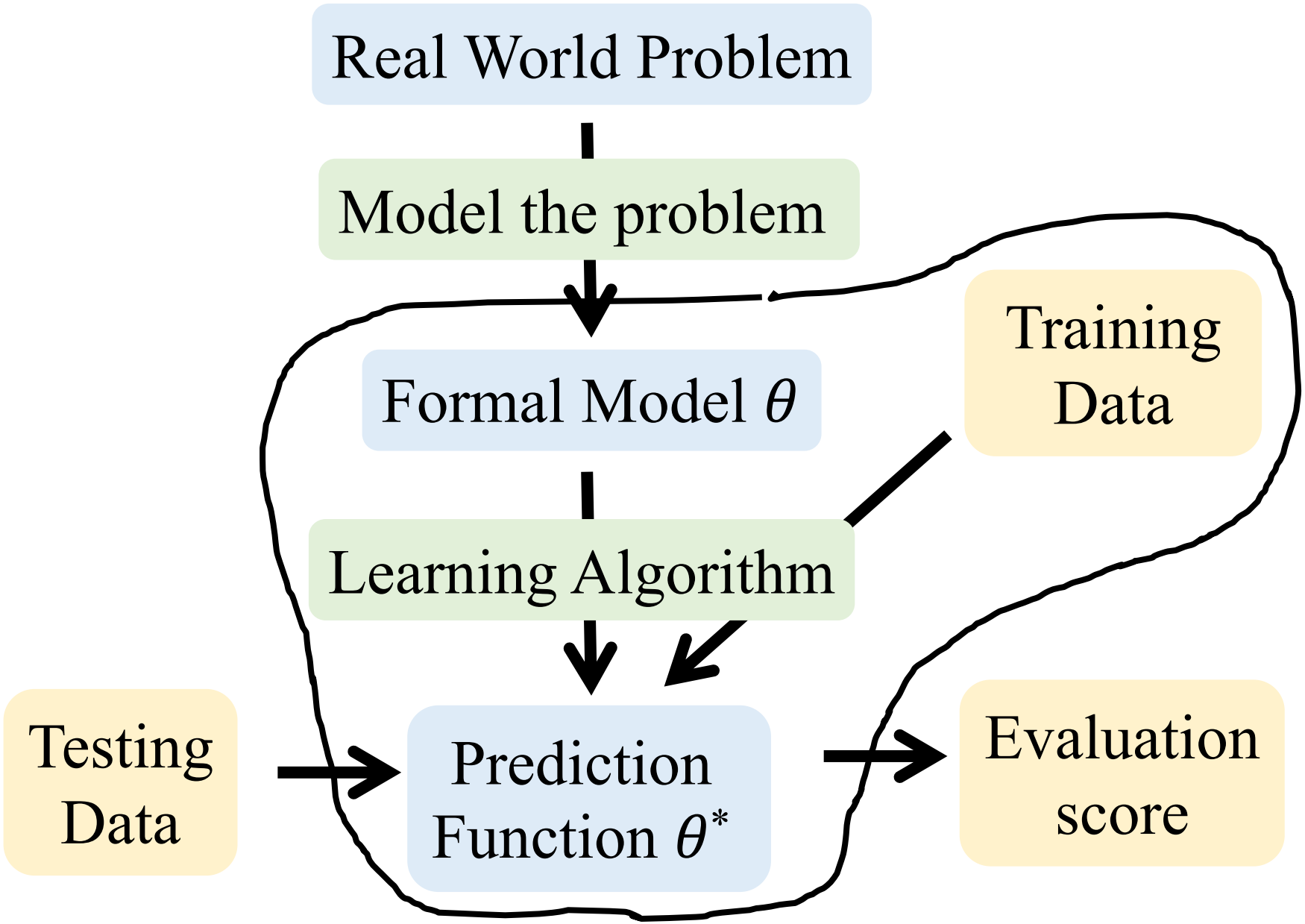
Supervised Learning



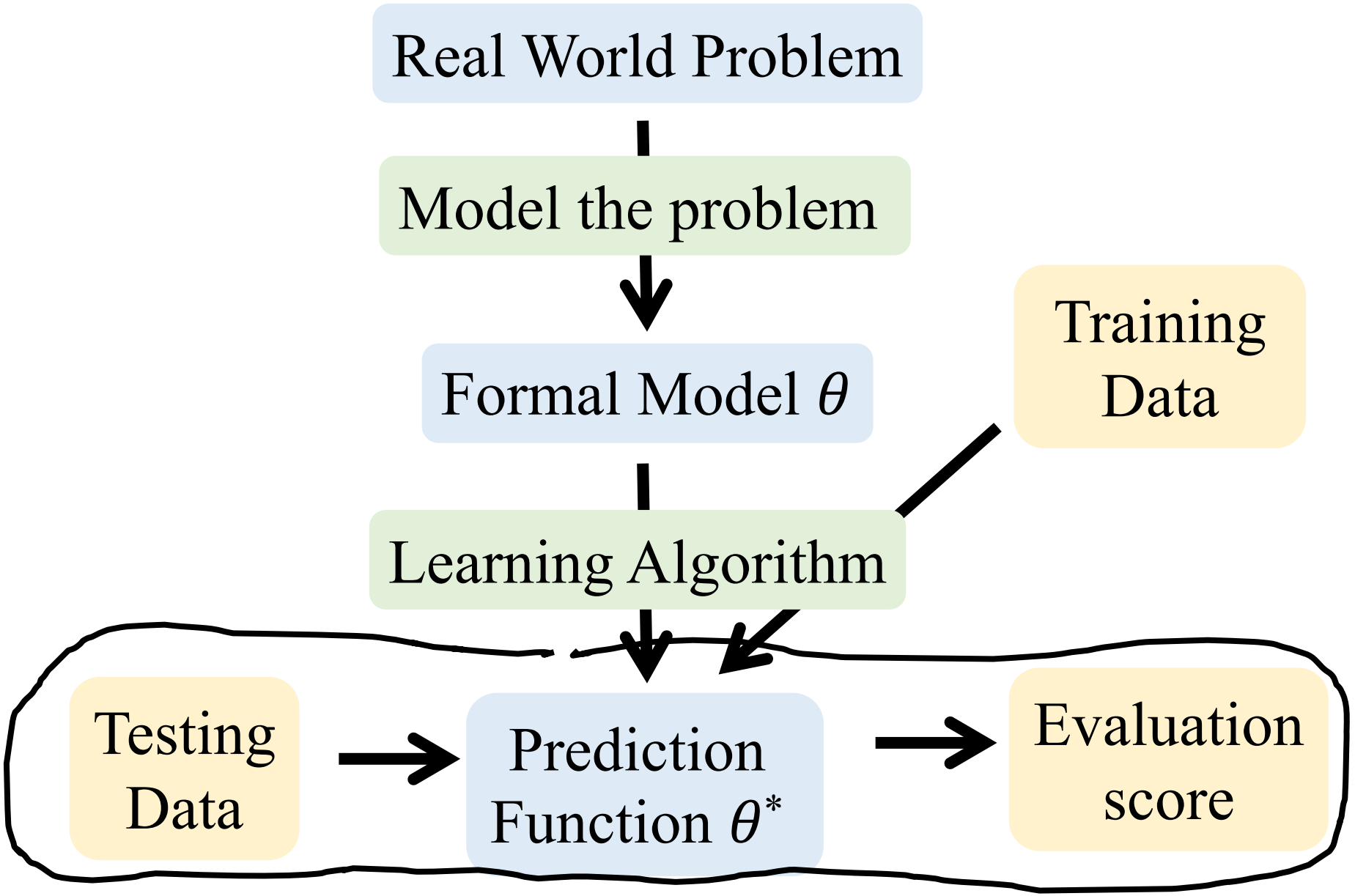
Modelling


Real World Problem
Model the problem
Formal Model θ
Training Data
Learning Algorithm
Testing Data
Prediction Function θ^*
Evaluation score

Training


Real World Problem
Model the problem
Formal Model θ
Learning Algorithm
Training Data
Testing Data
Prediction Function θ^*
Evaluation score

Testing


Real World Problem
Model the problem
Formal Model θ
Training Data
Learning Algorithm
Testing Data
Prediction Function θ^*
Evaluation score

Likelihood of Data

- Consider *n* I.I.D. random variables X_1, X_2, ..., X_n
  - X_i is a sample from density function $f(X_i \mid \theta)$
    - Note: now explicitly specify parameter θ of distribution
  - We want to determine how "likely" the observed data $(x_1, x_2, ..., x_n)$ is based on density $f(X_i \mid \theta)$
  - Define the **Likelihood function**, $L(\theta)$:

$$L(\theta) = \prod_{i=1}^{n} f(X_i \mid \theta)$$

    - This is just a product since X_i are I.I.D.
  - Intuitively: what is probability of observed data using density function $f(X_i \mid \theta)$, for some choice of θ

Maximum Likelihood Estimator

- The **Maximum Likelihood Estimator** (MLE) of θ, is the value of θ that maximizes $L(\theta)$
  - More formally: $\theta_{MLE} = \arg\max_{\theta} L(\theta)$
  - More convenient to use **log-likelihood function**, $LL(\theta)$:

$$LL(\theta) = \log L(\theta) = \log \prod_{i=1}^{n} f(X_i \mid \theta) = \sum_{i=1}^{n} \log f(X_i \mid \theta)$$

  - θ that maximizes $LL(\theta)$ also maximizes $L(\theta)$
    - Formally: $\arg\max_{\theta} LL(\theta) = \arg\max_{\theta} L(\theta)$
    - Similarly, for any positive constant c (not dependent on θ):

$$\arg\max_{\theta}(c \cdot LL(\theta)) = \arg\max_{\theta} LL(\theta) = \arg\max_{\theta} L(\theta)$$

Maximum Likelihood

$$L(\theta) = \prod_{i=1}^{n} f(X_i \mid \theta)$$

$$LL(\theta) = \sum_{i=1}^{n} \log f(X_i|\theta)$$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\ LL(\theta)$$

Option #1: Straight optimization

Computing the MLE

- General approach for finding MLE of θ
  - Determine formula for $LL(\theta)$
  - Differentiate $LL(\theta)$ w.r.t. (each) θ: $\frac{\partial LL(\theta)}{\partial \theta}$
  - To maximize, set $\frac{\partial LL(\theta)}{\partial \theta} = 0$
  - Solve resulting (simultaneous) equations to get θ_{MLE}
    - Make sure that derived $\hat{\theta}_{MLE}$ is actually a maximum (and not a minimum or saddle point). E.g., check $LL(\theta_{MLE} \pm \varepsilon) < LL(\theta_{MLE})$
      - This step often ignored in expository derivations
      - So, we'll ignore it here too (and won't require it in this class)

Bernoulli PMF

$$X \sim \text{Ber}(p)$$

$$f(X = x|p) = p^x(1-p)^{1-x}$$

Maximizing Likelihood with Bernoulli

- Consider I.I.D. random variables X_1, X_2, ..., X_n
  - $X_i \sim \text{Ber}(p)$
  - Probability mass function, $f(X_i \mid p)$, can be written as:

$$f(X_i \mid p) = p^{x_i}(1-p)^{1-x_i} \quad \text{where } x_i = 0 \text{ or } 1$$

  - Likelihood: $L(\theta) = \prod_{i=1}^{n} p^{X_i}(1-p)^{1-X_i}$
  - Log-likelihood:

$$LL(\theta) = \sum_{i=1}^{n} \log\left(p^{X_i}(1-p)^{1-X_i}\right) = \sum_{i=1}^{n} \left[X_i(\log p) + (1-X_i)\log(1-p)\right]$$

$$= Y(\log p) + (n-Y)\log(1-p) \quad \text{where } Y = \sum_{i=1}^{n} X_i$$

  - Differentiate w.r.t. p, and set to 0:

$$\frac{\partial LL(p)}{\partial p} = Y\frac{1}{p} + (n-Y)\frac{-1}{1-p} = 0 \quad \Rightarrow \quad p_{MLE} = \frac{Y}{n} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Maximizing Likelihood with Poisson

- Consider I.I.D. random variables X_1, X_2, ..., X_n
  - $X_i \sim \text{Poi}(\lambda)$
  - PMF: $f(X_i \mid \lambda) = \frac{e^{-\lambda}\lambda^{x_i}}{x_i!}$ Likelihood: $L(\theta) = \prod_{i=1}^{n} \frac{e^{-\lambda}\lambda^{X_i}}{X_i!}$
  - Log-likelihood:

$$LL(\theta) = \sum_{i=1}^{n} \log\left(\frac{e^{-\lambda}\lambda^{X_i}}{X_i!}\right) = \sum_{i=1}^{n} \left[-\lambda \log(e) + X_i \log(\lambda) - \log(X_i!)\right]$$

$$= -n\lambda + \log(\lambda)\sum_{i=1}^{n} X_i - \sum_{i=1}^{n} \log(X_i!)$$

  - Differentiate w.r.t. λ, and set to 0:

$$\frac{\partial LL(\lambda)}{\partial \lambda} = -n + \frac{1}{\lambda}\sum_{i=1}^{n} X_i = 0 \quad \Rightarrow \quad \lambda_{MLE} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Maximizing Likelihood with Normal

- Consider I.I.D. random variables X_1, X_2, ..., X_n
  - $X_i \sim N(\mu, \sigma^2)$
  - PDF: $f(X_i \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(X_i-\mu)^2/(2\sigma^2)}$
  - Log-likelihood:

$$LL(\theta) = \sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi}\sigma} e^{-(X_i-\mu)^2/(2\sigma^2)}\right) = \sum_{i=1}^{n} \left[-\log(\sqrt{2\pi}\sigma) - (X_i-\mu)^2/(2\sigma^2)\right]$$

  - First, differentiate w.r.t. μ, and set to 0:

$$\frac{\partial LL(\mu,\sigma^2)}{\partial \mu} = \sum_{i=1}^{n} 2(X_i-\mu)/(2\sigma^2) = \frac{1}{\sigma^2}\sum_{i=1}^{n}(X_i-\mu) = 0$$

  - Then, differentiate w.r.t. σ, and set to 0:

$$\frac{\partial LL(\mu,\sigma^2)}{\partial \sigma} = \sum_{i=1}^{n} -\frac{1}{\sigma} + 2(X_i-\mu)^2/(2\sigma^3) = -\frac{n}{\sigma} + \sum_{i=1}^{n}(X_i-\mu)^2/(\sigma^3) = 0$$

Being Normal, Simultaneously

- Now have two equations, two unknowns:

$$\frac{1}{\sigma^2}\sum_{i=1}^{n}(X_i-\mu)=0 \qquad -\frac{n}{\sigma}+\sum_{i=1}^{n}(X_i-\mu)^2/(\sigma^3)=0$$

- First, solve for μ_{MLE}:

$$\frac{1}{\sigma^2}\sum_{i=1}^{n}(X_i-\mu)=0 \;\Rightarrow\; \sum_{i=1}^{n}X_i=n\mu \;\Rightarrow\; \mu_{MLE}=\frac{1}{n}\sum_{i=1}^{n}X_i$$

- Then, solve for σ^2_{MLE}:

$$-\frac{n}{\sigma}+\sum_{i=1}^{n}(X_i-\mu)^2/(\sigma^3)=0 \;\Rightarrow\; n\sigma^2=\sum_{i=1}^{n}(X_i-\mu)^2$$

$$\sigma^2_{MLE}=\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu_{MLE})^2$$

- Note: μ_{MLE} unbiased, but σ^2_{MLE} biased

Maximizing Likelihood with Uniform

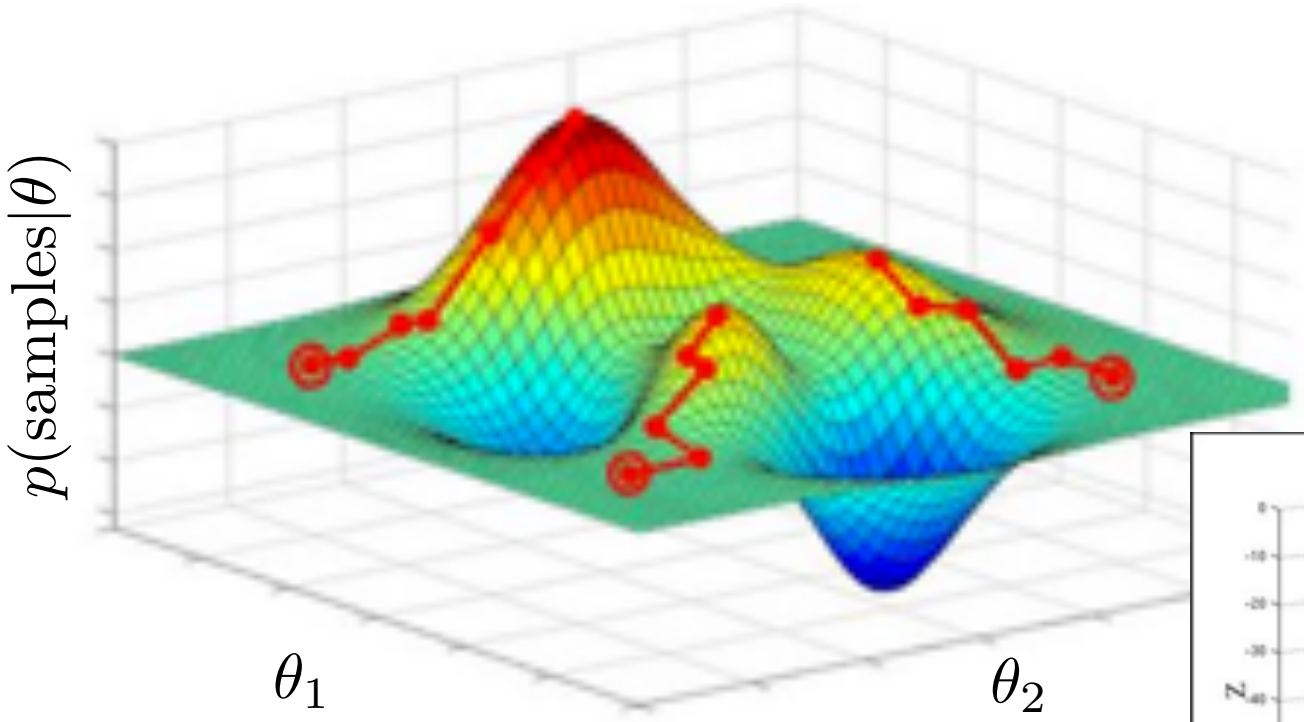
- Consider I.I.D. random variables X_1, X_2, ..., X_n
  - $X_i \sim \text{Uni}(\alpha, \beta)$
  - PDF: $f(X_i \mid \alpha, \beta) = \begin{cases} \frac{1}{\beta - \alpha} & \alpha \le x_i \le \beta \\ 0 & \text{otherwise} \end{cases}$
  - Likelihood: $L(\theta) = \begin{cases} \left(\frac{1}{\beta - \alpha}\right)^n & \alpha \le x_1, x_2, \ldots, x_n \le \beta \\ 0 & \text{otherwise} \end{cases}$
    - Constraint $\alpha \le x_1, x_2, \ldots, x_n \le \beta$ makes differentiation tricky
    - Intuition: want interval size $(\beta - \alpha)$ to be as small as possible to maximize likelihood function for each data point
    - But need to make sure all observed data contained in interval
      - If all observed data not in interval, then $L(\theta) = 0$
  - Solution: $\alpha_{MLE} = \min(x_1, \ldots, x_n)$ $\beta_{MLE} = \max(x_1, \ldots, x_n)$

Small Samples = Problems

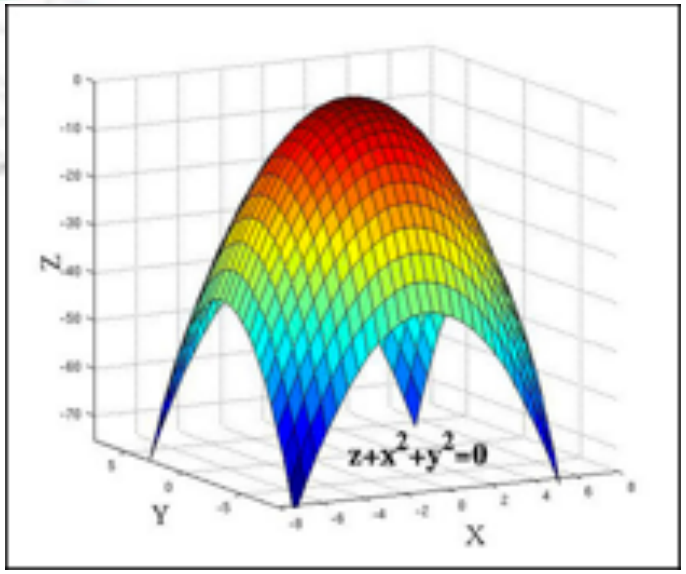
- How do small samples affect MLE?
  - In many cases, $\mu_{MLE} = \frac{1}{n}\sum_{i=1}^{n} X_i$ = sample mean
    - Unbiased. Not too shabby…
  - As seen with Normal, $\sigma^2_{MLE} = \frac{1}{n}\sum_{i=1}^{n}(X_i - \mu_{MLE})^2$
    - Biased. Underestimates for small *n* (e.g., 0 for n = 1)
  - As seen with Uniform, $\alpha_{MLE} \geq \alpha$ and $\beta_{MLE} \leq \beta$
    - Biased. Problematic for small *n* (e.g., $\alpha = \beta$ when n = 1)
  - Small sample phenomena intuitively make sense:
    - Maximum likelihood $\Rightarrow$ best explain data we've seen
    - Does not attempt to generalize to unseen data

Argmax
Option #2: Gradient Ascent

Gradient Ascent



Especially good if function is convex



Walk uphill and you will find a local maxima
(if your step size is small enough)

Gradient Ascent

Repeat many times

$$\theta_j^{\text{new}} = \theta_j^{\text{old}} + \eta \cdot \frac{\partial LL(\theta^{\text{old}})}{\partial \theta_j^{\text{old}}}$$

This is some **profound** life philosophy

Walk uphill and you will find a local maxima
(if your step size is small enough)

End Review

Review: Maximum Likelihood Algorithm

1. Decide on a model for the likelihood of your samples. This is often using a PMF or PDF.

2. Write out the log likelihood function.

3. State that the optimal parameters are the argmax of the log likelihood function.

4. Use an optimization algorithm to calculate argmax

Review: Maximum Likelihood Algorithm

1. Decide on a model for the likelihood of your samples. This is often using a PMF or PDF.

2. Write out the log likelihood function.

3. State that the optimal parameters are the argmax of the log likelihood function.

4. Calculate the derivative of LL with respect to theta

5. Use an optimization algorithm to calculate argmax

Gradient Ascent

Initialize: $\theta_j = 0$ **for all** $0 \leq j \leq m$

Calculate all θ_j

Gradient Ascent

Initialize: $\theta_j = 0$ for all $0 \leq j \leq m$

Repeat many times:

gradient[j] = 0 for all $0 \leq j \leq m$

Calculate all gradient[j]*'s based on data*

θ_j += η * gradient[j] for all $0 \leq j \leq m$

Linear Regression Lite

Predicting CO_2

X_1 = Temperature

X_2 = Elevation

X_3 = CO_2 level yesterday

X_4 = GDP of region

X_5 = Acres of forest growth

Y = CO_2 levels

Predicting CO_2 (simple)

X = CO_2 level

Y = Average Global Temperature

N training datapoints

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

Linear Regression Lite Model

$$Y = \theta \cdot X + Z \qquad Z \sim N(0, \sigma^2) \qquad Y|X \sim N(\theta X, \sigma^2)$$

1) Write Likelihood Fn

N training datapoints

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

Model

$$Y|X \sim N(\theta X, \sigma^2)$$

First, calculate Likelihood of the data

$$L(\theta) = \prod_{i=1}^{n} f(y^{(i)}, x^{(i)}|\theta)$$

Let's break up this joint

Shorthand for:

$$f(Y = y^{(i)}, X = x^{(i)}|\theta)$$

1) Write Likelihood Fn

N training datapoints

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

Model

$$Y|X \sim N(\theta X, \sigma^2)$$

First, calculate Likelihood of the data

$$L(\theta) = \prod_{i=1}^{n} f(y^{(i)}, x^{(i)}|\theta)$$

Let's break up this joint

$$= \prod_{i=1}^{n} f(y^{(i)}|x^{(i)}, \theta) f(x^{(i)})$$

$f(x^{(i)})$ is independent of θ

$$= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y^{(i)} - \theta x^{(i)})^2}{2\sigma^2}} f(x^{(i)})$$

Definition of $f(y^{(i)}|x^{(i)})$

2) Write Log Likelihood Fn

N training datapoints: $(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \ldots (\mathbf{x}^{(n)}, y^{(n)})$

Likelihood function: $L(\theta) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y^{(i)} - \theta x^{(i)})^2}{2\sigma^2}} f(x^{(i)})$

Second, calculate Log Likelihood of the data

$$
\begin{aligned}
LL(\theta) &= \log L(\theta) \\
&= \log \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y^{(i)} - \theta x^{(i)})^2}{2\sigma^2}} f(X^{(i)}) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y^{(i)} - \theta x^{(i)})^2}{2\sigma^2}} + \sum_{i=1}^{n} \log f(X^{(i)}) \\
&= n \log \frac{1}{\sqrt{2\pi}} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y^{(i)} - \theta x^{(i)})^2 + \sum_{i=1}^{n} \log f(x^{(i)})
\end{aligned}
$$

3) State MLE as Optimization

N training datapoints: $(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \ldots (\mathbf{x}^{(n)}, y^{(n)})$

Log Likelihood: $LL(\theta) = n \log \frac{1}{\sqrt{2\pi}} - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y^{(i)} - \theta x^{(i)})^2 + \sum_{i=1}^{n} \log f(x^{(i)})$

Third, celebrate!

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} - \sum_{i=1}^{n} (y^{(i)} - \theta x^{(i)})^2$$

4) Find derivative

N training datapoints: $(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \ldots (\mathbf{x}^{(n)}, y^{(n)})$

Goal: $\hat{\theta} = \underset{\theta}{\text{argmax}} - \sum_{i=1}^{n} (y^{(i)} - \theta x^{(i)})^2$

Fourth, optimize!

$$\frac{\partial LL(\theta)}{\partial \theta} = \frac{\partial}{\partial \theta} - \sum_{i=1}^{n} (y^{(i)} - \theta x^{(i)})^2$$

$$= -\sum_{i=1}^{n} \frac{\partial}{\partial \theta} (y^{(i)} - \theta X^{(i)})^2$$

$$= -\sum_{i=1}^{n} 2(y^{(i)} - \theta x^{(i)})(-x^{(i)})$$

$$= \sum_{i=1}^{n} 2(y^{(i)} - \theta x^{(i)})(x^{(i)})$$

5) Run optimization code

N training datapoints: $(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \ldots (\mathbf{x}^{(n)}, y^{(n)})$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} - \sum_{i=1}^{n} (y^{(i)} - \theta x^{(i)})^2$$

$$\frac{\partial LL(\theta)}{\partial \theta} = \sum_{i=1}^{n} 2(y^{(i)} - \theta x^{(i)})(x^{(i)})$$

Gradient Ascent

Initialize: $\theta_j = 0$ for all $0 \leq j \leq m$

Repeat many times:

gradient[j] = 0 for all $0 \leq j \leq m$

Calculate all gradient[j]*'s based on data and current setting of theta*

θ_j += η * gradient[j] for all $0 \leq j \leq m$

Linear Regression (simple)

```
Initialize: θ = 0
```

```
Repeat many times:
    gradient = 0
    Calculate gradient based on data
    θ += η * gradient
```

Linear Regression (simple)

```
Initialize: θ = 0

Repeat many times:
    gradient = 0
    For each training example (x, y):
        Update gradient for current training example
    θ += η * gradient
```

Linear Regression (simple)

```
Initialize: θ = 0

Repeat many times:
    gradient = 0
    For each training example (x, y):
        gradient += 2(y - θ x)(x)
    θ += η * gradient
```

Linear Regression

Predicting CO_2

X_1 = Temperature

X_2 = Elevation

X_3 = CO_2 level yesterday

X_4 = GDP of region

Y = CO_2 levels

Linear Regression

Problem: Predict real value Y based on observing variable X

Model: Linear weight every feature

$$\begin{aligned}\hat{Y} &= \theta_1 X_1 + \cdots + \theta_m X_m + \theta_{m+1} \\ &= \theta^T \mathbf{X}\end{aligned}$$

Training: Gradient ascent to chose the best thetas to describe your data

$$\hat{\theta}_{MLE} = \underset{\theta}{\operatorname{argmax}} \; -\sum_{i=1}^{n} (Y^{(i)} - \theta^T \mathbf{x}^{(i)})^2$$

Linear Regression

Initialize: $\theta_j = 0$ for all $0 \leq j \leq m$

Repeat many times:

gradient[j] = 0 for all $0 \leq j \leq m$

For each training example (x, y):

For each parameter j:

gradient[j] += $(y - \theta^T x)(-x[j])$

θ_j += η * gradient[j] for all $0 \leq j \leq m$

Predicting CO_2

$$Y = CO_2 \text{ levels}$$

$$\hat{Y} = \theta_1 X_1 + \cdots + \theta_m X_m + \theta_{m+1}$$
$$= \theta^T \mathbf{X}$$

X_1 = Temperature

X_2 = Elevation

X_3 = CO_2 level yesterday

X_4 = GDP of region

$\theta_1 = -2.3$

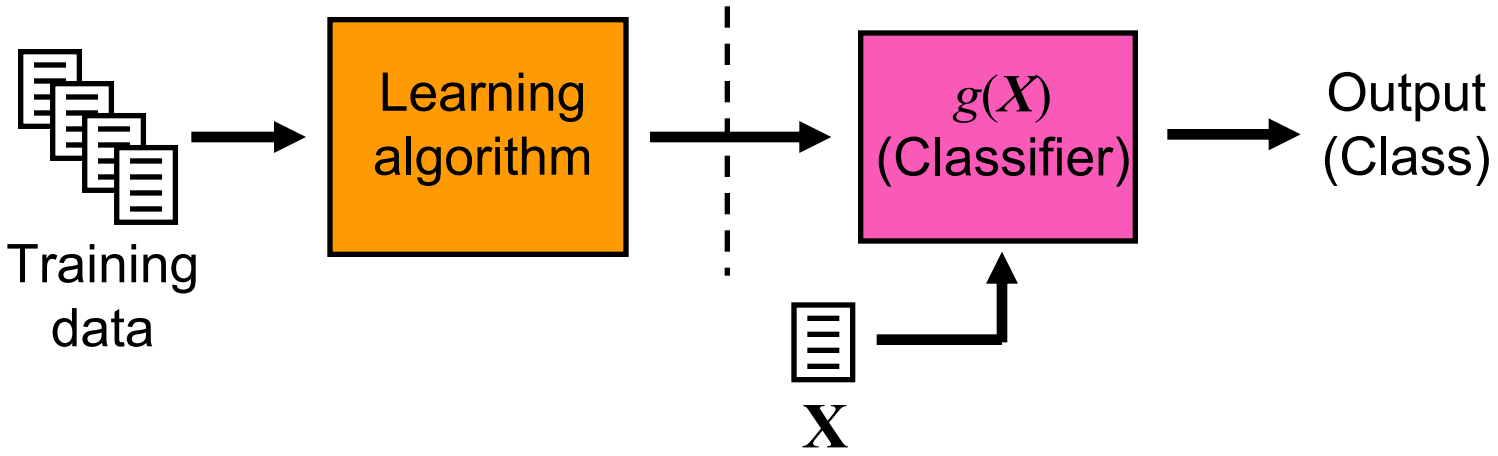
$\theta_2 = +1.2$

$\theta_3 = +10.2$

$\theta_4 = +3.3$

$\theta_5 = +95.4$

The Machine Learning Process



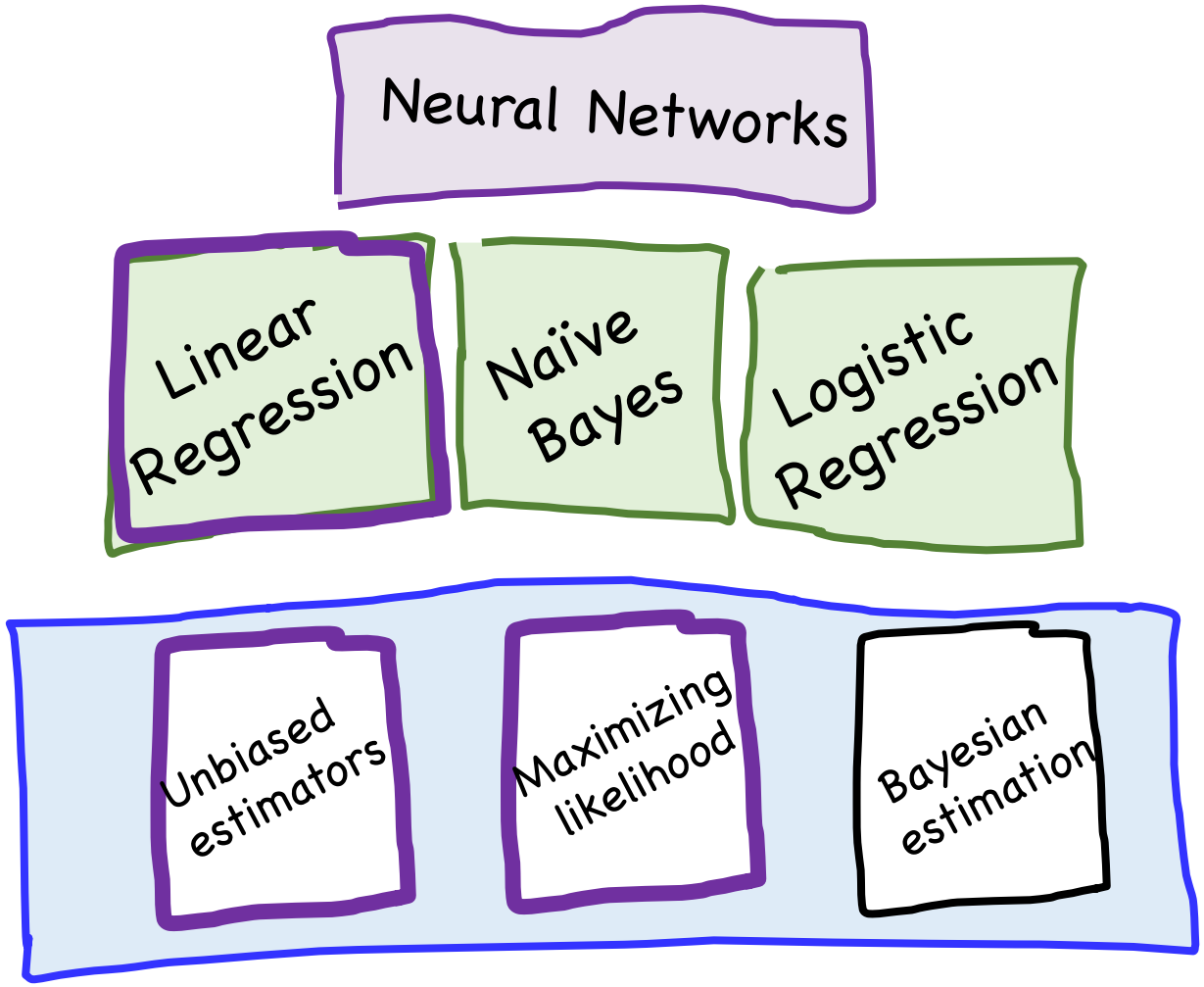
- Training data: set of *N* pre-classified data instances
  - *N* training pairs: $(\boldsymbol{x}^{(1)}, y^{(1)})$, $(\boldsymbol{x}^{(2)}, y^{(2)})$, …, $(\boldsymbol{x}^{(n)}, y^{(n)})$
    - Use superscripts to denote *i*-th training instance
- Learning algorithm: method for determining $g(\boldsymbol{X})$
  - Given a new input observation of $\boldsymbol{x} = x_1, x_2, \ldots, x_m$
  - Use $g(\boldsymbol{x})$ to compute a corresponding output (prediction)

Stretch!

Maximum A Posteriori

Noah Arthurs
CS109, Stanford University

Our Path


Neural Networks
Linear Regression
Naïve Bayes
Logistic Regression
Unbiased estimators
Maximizing likelihood
Bayesian estimation

Something rotten
in the world of MLE

Need a Volunteer

Two Envelopes

- I have two envelopes, will allow you to have one
  - One contains $X, the other contains $2X
  - Select an envelope
    - Open it!
  - Now, would you like to switch for other envelope?
  - To help you decide, compute E[$ in other envelope]
    - Let Y = $ in envelope you selected

$$E[\$ \text{ in other envelope}] = \frac{1}{2} \cdot \frac{Y}{2} + \frac{1}{2} \cdot 2Y = \frac{5}{4} Y$$

  - Before opening envelope, think either equally good
  - So, what happened by opening envelope?
    - And does it really make sense to switch?

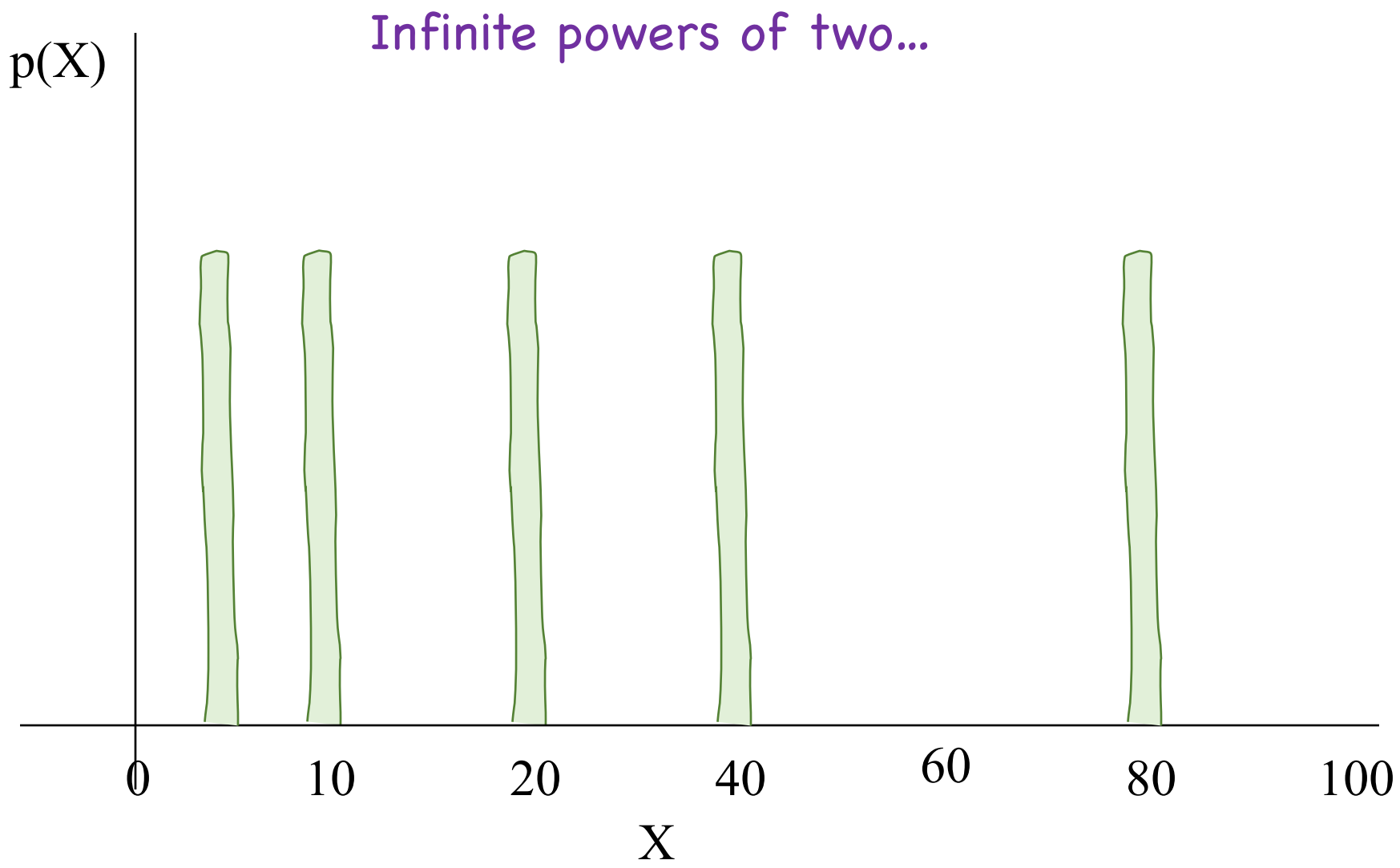
Thinking Deeper About Two Envelopes

- The “two envelopes” problem set-up
  - Two envelopes: one contains $X, other contains $2X
  - You select an envelope and open it
    - Let Y = $ in envelope you selected
    - Let Z = $ in other envelope

$$E[Z \mid Y] = \frac{1}{2} \cdot \frac{Y}{2} + \frac{1}{2} \cdot 2Y = \frac{5}{4} Y$$

  - E[Z | Y] above assumes all values X (where $0 < X < \infty$) are equally likely
    - Note: there are infinitely many values of X
    - So, not true probability distribution over X (doesn’t integrate to 1)

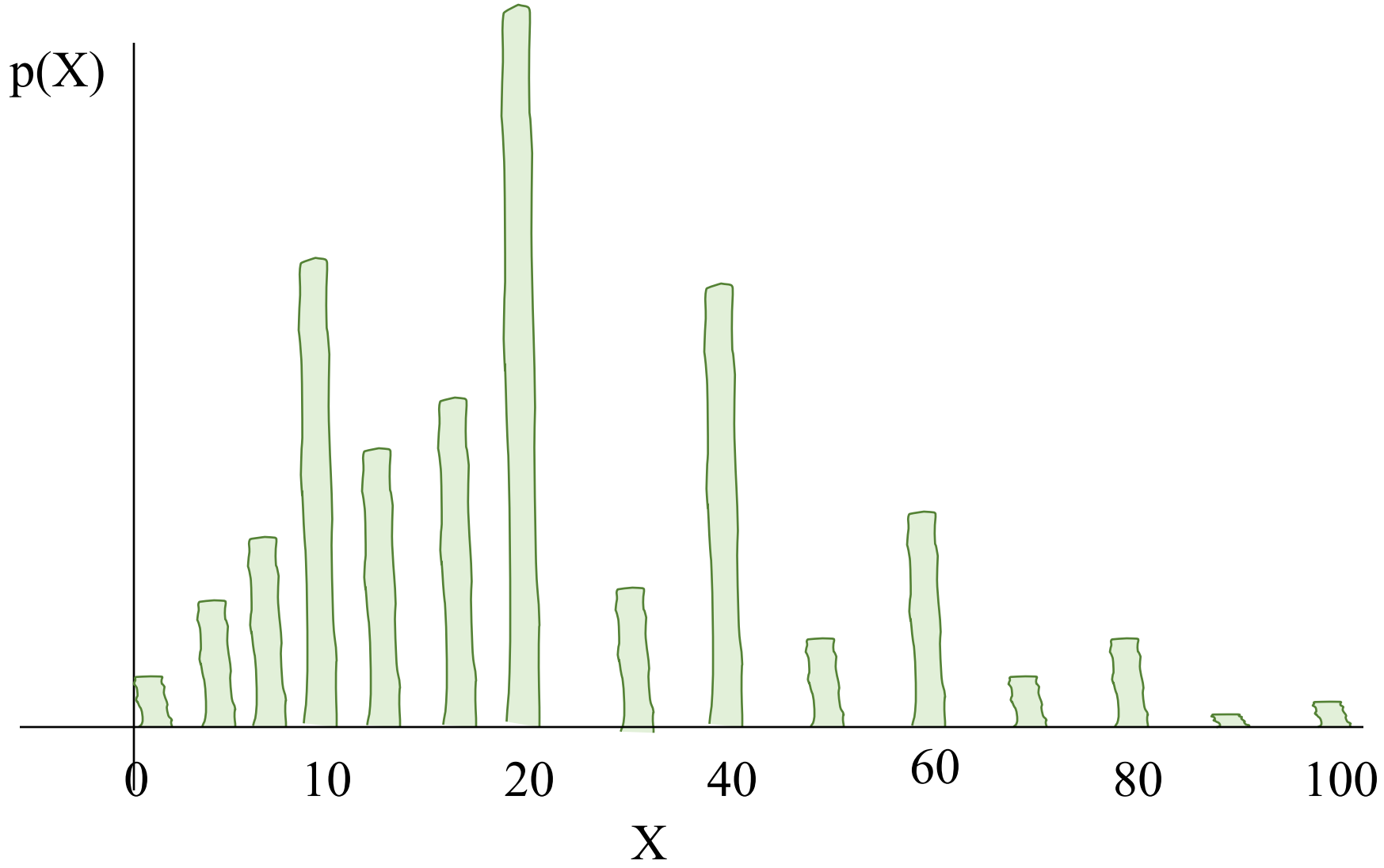
All Values are Equally Likely?



Subjectivity of Probability

- Belief about contents of envelopes
  - Since implied distribution over X is not a true probability distribution, what is our distribution over X?
    - *Frequentist*: play game infinitely many times and see how often different values come up.
    - Problem: I only allow you to play the game *once*
  - Bayesian probability
    - Have prior belief of distribution for X (or anything for that matter)
    - Prior belief is a *subjective* probability
      - By extension, *all* probabilities are subjective
    - Allows us to answer question when we have no/limited data
      - E.g., probability a coin you've never flipped lands on heads
    - As we get more data, prior belief is "swamped" by data

Subjectivity of Probability



The Envelope, Please

- *Bayesian*: have prior distribution over X, P(X)
  - Let Y = $ in envelope you selected
  - Let Z = $ in other envelope
  - Open your envelope to determine Y
  - If Y > E[Z | Y], keep your envelope, otherwise switch
    - No inconsistency!
  - Opening envelope provides data to compute P(X | Y) and thereby compute E[Z | Y]
  - Of course, there's the issue of how you determined your prior distribution over X…
    - Bayesian: Doesn't matter how you determined prior, but you *must* have one (whatever it is)
    - Imagine if envelope you opened contained $20.01

Envelope Summary:
Probabilities are beliefs
Incorporating prior beliefs is useful

Priors for Parameter Estimation?

Flash Back: Bayes Theorem

- Bayes' Theorem (θ = model parameters, D = data):

"Posterior" "Likelihood" "Prior"

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

  - Likelihood: you've seen this before (in context of MLE)
    - Probability of data given probability model (parameter θ)
  - Prior: before seeing any data, what is belief about model
    - I.e., what is *distribution* over parameters θ
  - Posterior: after seeing data, what is belief about model
    - After data D observed, have posterior distribution $p(\theta \mid D)$ over parameters θ conditioned on data. Use this to predict new data.

MLE vs MAP

Data: $x^{(1)}, \ldots, x^{(n)}$

Maximum Likelihood Estimation

$$\hat{\theta}_{MLE} = \underset{\theta}{\operatorname{argmax}}\ f(X^{(1)} = x^{(1)}, \ldots, X^{(n)} = x^{(n)} | \theta)$$

$$= \underset{\theta}{\operatorname{argmax}} \left(\sum_i \log f(X^{(i)} = x^{(i)} | \theta) \right)$$

Maximum A Posteriori

$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}}\ f(\Theta = \theta | X^{(1)} = x^{(1)}, \ldots, X^{(n)} = x^{(n)})$$

Notation Shorthand

MAP, without shorthand

$$\hat{\theta}_{MAP} = \underset{\theta}{\text{argmax}}\ f(\Theta = \theta | X^{(1)} = x^{(1)}, \ldots, X^{(n)} = x^{(n)})$$

Our shorthand notation

θ is shorthand for the event: $\Theta = \theta$

$x^{(i)}$ is shorthand for the event: $X^{(i)} = x^{(i)}$

MAP, now with shorthand

$$\hat{\theta}_{MAP} = \underset{\theta}{\text{argmax}}\ f(\theta | x^{(1)}, \ldots, x^{(n)})$$

MLE vs MAP

Data: $x^{(1)}, \ldots, x^{(n)}$

Maximum Likelihood Estimation

$$\hat{\theta}_{MLE} = \underset{\theta}{\operatorname{argmax}}\ f(x^{(1)}, \ldots, x^{(n)}|\theta)$$

$$= \underset{\theta}{\operatorname{argmax}} \left(\sum_i \log f(x^{(i)}|\theta) \right)$$

Maximum A Posteriori

$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}}\ f(\theta|x^{(1)}, \ldots, x^{(n)})$$

Most important slide of today

Maximum A Posteriori

data: $x^{(1)}, \ldots, x^{(n)}$ $\qquad$ $\hat{\theta}_{MAP} = \underset{\theta}{\text{argmax}}\ f(\theta | x^{(1)}, \ldots, x^{(n)})$

---

likelihood

$$\hat{\theta}_{MAP} = \underset{\theta}{\text{argmax}}\ \frac{f(x^{(1)}, x^{(2)}, \ldots, x^{(n)} | \theta) g(\theta)}{h(x^{(1)}, x^{(2)}, \ldots x^{(n)})}$$

posterior

prior

Maximum A Posteriori

data: $x^{(1)}, \ldots, x^{(n)}$ $\qquad$ $\hat{\theta}_{MAP} = \underset{\theta}{\text{argmax}}\ f(\theta | x^{(1)}, \ldots, x^{(n)})$

$$\hat{\theta}_{MAP} = \underset{\theta}{\text{argmax}}\ \frac{g(\theta) f(x^{(1)}, x^{(2)}, \ldots, x^{(n)} | \theta)}{h(x^{(1)}, x^{(2)}, \ldots x^{(n)})}$$

$$= \underset{\theta}{\text{argmax}}\ \frac{g(\theta) \prod_{i=1}^{n} f(x^{(i)} | \theta)}{h(x^{(1)}, x^{(2)}, \ldots, x^{(n)})}$$

$$= \underset{\theta}{\text{argmax}}\ g(\theta) \prod_{i=1}^{n} f(x^{(i)} | \theta)$$

$$= \underset{\theta}{\text{argmax}}\ \left(\log(g(\theta)) + \sum_{i=1}^{n} \log(f(x^{(i)} | \theta)) \right)$$

Maximum A Posteriori

Estimated parameter

Log prior

$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}} \left(\log(g(\theta)) + \sum_{i=1}^{n} \log(f(x^{(i)}|\theta)) \right)$$

Chose the value of theta that maximizes:

Sum of log likelihood

MLE vs MAP

Data: $x^{(1)}, \ldots, x^{(n)}$

Maximum Likelihood Estimation

$$\hat{\theta}_{MLE} = \underset{\theta}{\operatorname{argmax}}\ f(x^{(1)}, \ldots, x^{(n)}|\theta)$$

$$= \underset{\theta}{\operatorname{argmax}} \left(\sum_i \log f(x^{(i)}|\theta) \right)$$

Maximum A Posteriori

$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}}\ f(\theta|x^{(1)}, \ldots, x^{(n)})$$

$$= \underset{\theta}{\operatorname{argmax}} \left(\log(g(\theta)) + \sum_{i=1}^{n} \log(f(x^{(i)}|\theta)) \right)$$

Gotta get that intuition

$P(\theta \mid D)$ For Bernoulli

- Prior: θ ~ Beta(*a*, *b*); data = {*n* heads, *m* tails}
- Estimate *p, aka* θ

$$\hat{\theta}_{MAP} = \underset{\theta}{\text{argmax}}\ f(\theta|\text{data}) = \underset{\theta}{\text{argmax}}\ f(\text{data}|\theta)g(\theta)$$

This is the beta PDF

$$= \underset{\theta}{\text{argmax}}\ \log g(\theta) + \log f(\text{data}|\theta)$$

This is ???

$P(\theta \mid D)$ For Bernoulli

- Prior: θ ~ Beta(a, b); data = {n heads, m tails}
- Estimate p, *aka* θ

$$\hat{\theta}_{MAP} = \underset{\theta}{\text{argmax}}\ f(\theta|\text{data}) \qquad = \underset{\theta}{\text{argmax}}\ f(\text{data}|\theta)g(\theta)$$

This is the beta PDF

$$= \underset{\theta}{\text{argmax}}\ \log g(\theta) + \log f(\text{data}|\theta)$$

Product of thetas and (1-theta)s

$$= \underset{\theta}{\text{argmax}} \log \left[\frac{1}{\beta}\theta^{a-1}(1-\theta)^{b-1}\right] + n \log f(\text{heads}|\theta) + m \log f(\text{tails}|\theta)$$

$$= \underset{\theta}{\text{argmax}} \log \frac{1}{\beta} + (a-1)\log\theta + (b-1)\log(1-\theta) + n\log\theta + m\log(1-\theta)$$

$$= \underset{\theta}{\text{argmax}} (a-1+n)\log\theta + (b-1+m)\log(1-\theta)$$

$P(\theta \mid D)$ For Bernoulli

- Prior: θ ~ Beta(a, b); D = {n heads, m tails}
- Estimate p, *aka* θ

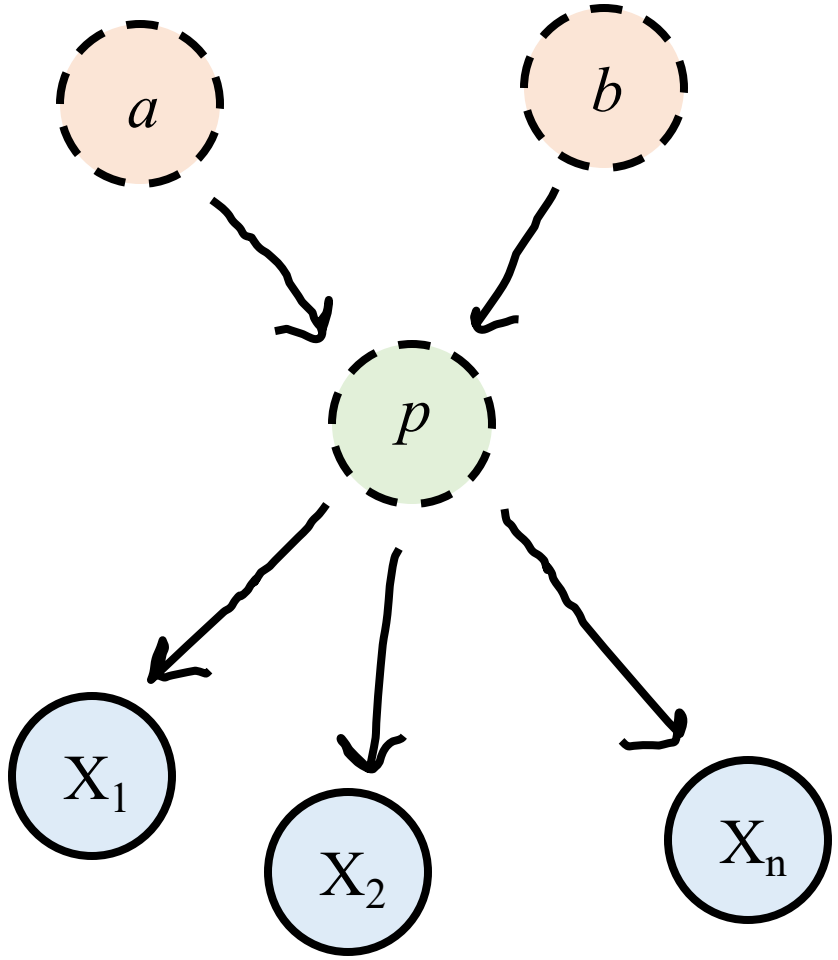
$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}}\ f(\theta|\text{data})$$

$$= \underset{\theta}{\operatorname{argmax}} (a-1+n)\log\theta + (b-1+m)\log(1-\theta)$$

$$= \frac{n+a-1}{n+m+a+b-2}$$

That's the mode of the updated beta

Hyper Parameters



Hyperparameter

a, b are fixed

Prior

$p \sim \text{Beta}(a, b)$

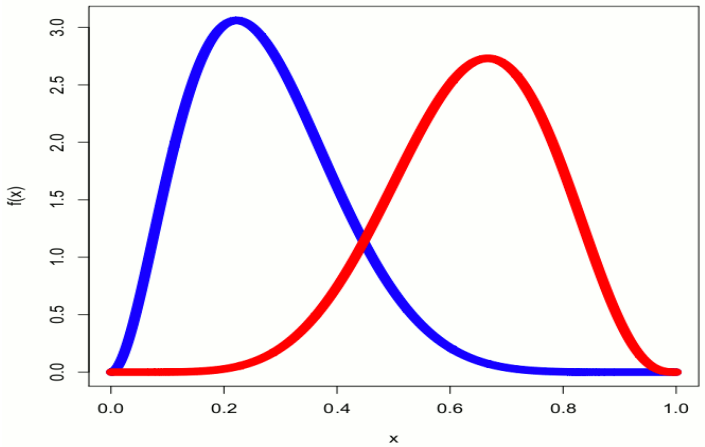
Data distribution

$X_i \sim \text{Bern}(p)$

MAP will estimate the most likely value of p for this model

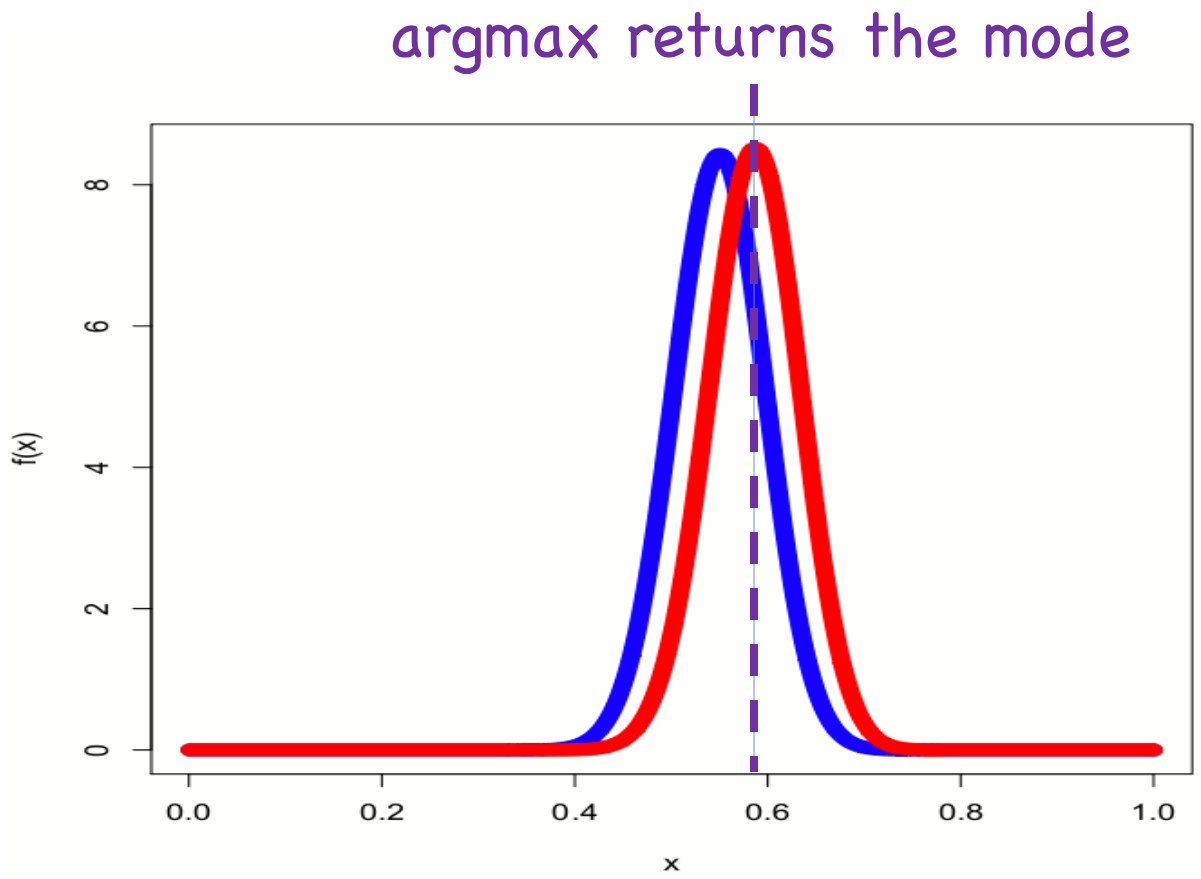
Where'd Ya Get Them P(θ)?

- θ is the probability a coin turns up heads
- Model θ with 2 different priors:
  - $P_1(\theta)$ is Beta(3,8) (blue)
  - $P_2(\theta)$ is Beta(7,4) (red)
- They look pretty different!



- Now flip 100 coins; get 58 heads and 42 tails
  - What do posteriors look like?

It's Like Having Twins



- As long as we collect enough data, posteriors will converge to the true value!