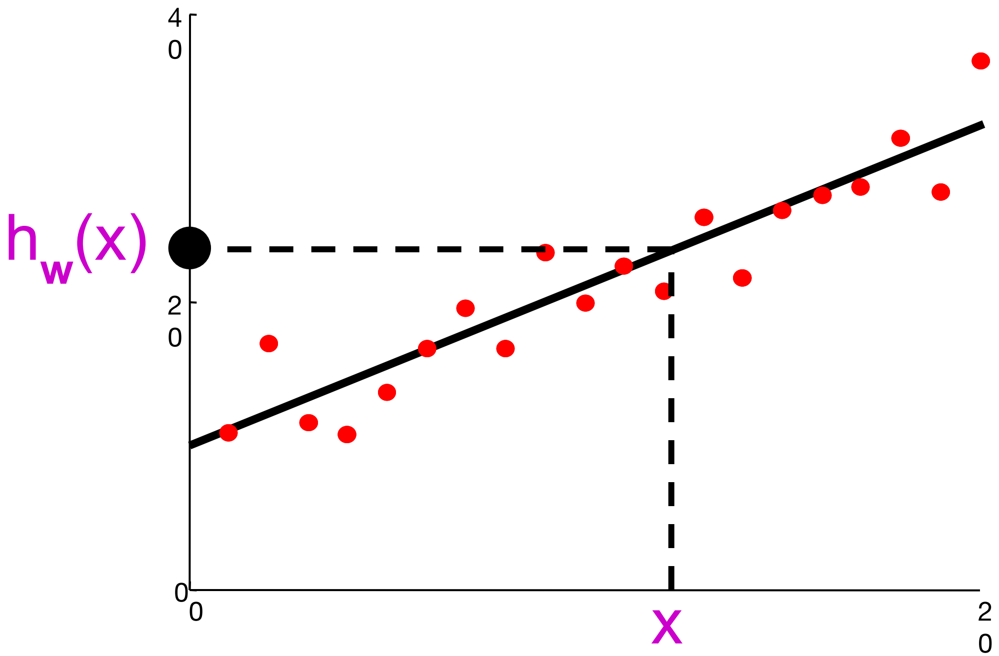
CS 188: Artificial Intelligence

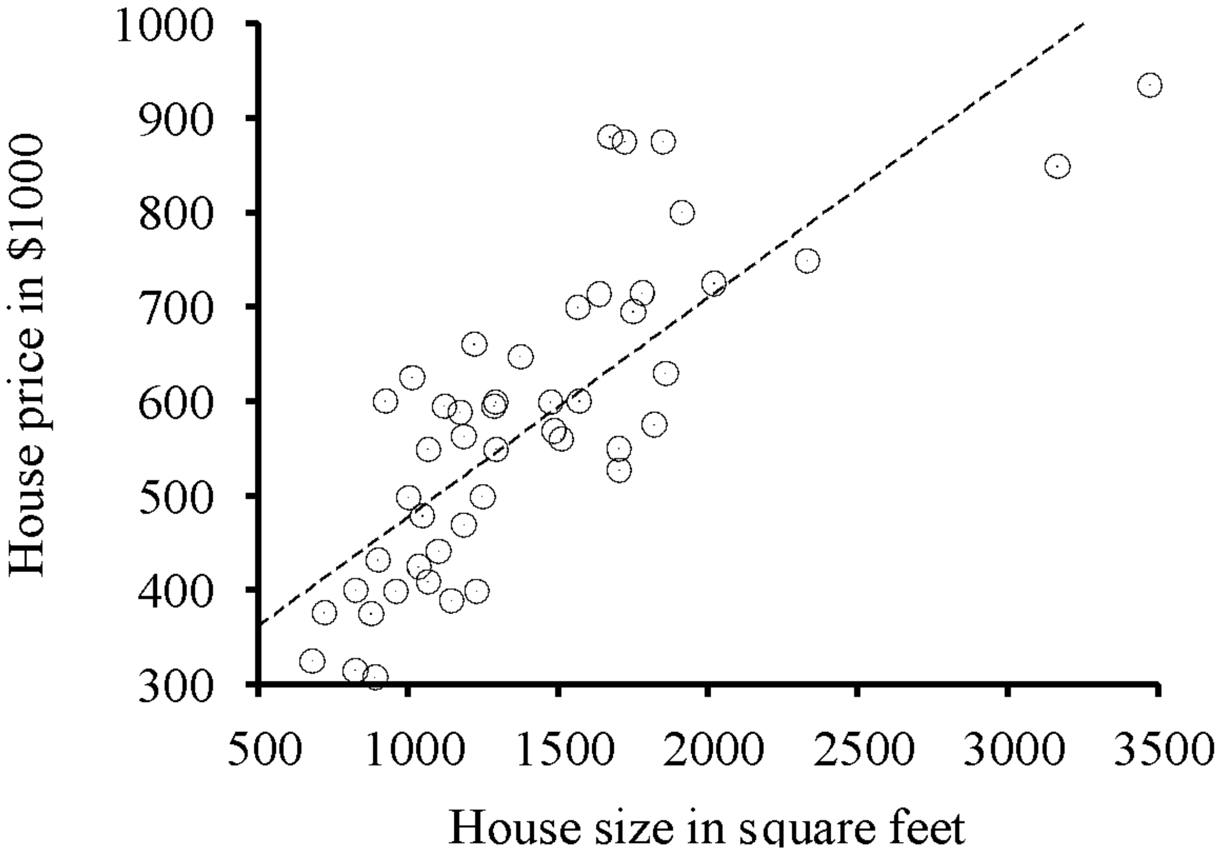
Learning IV: Statistical learning & Naïve Bayes

Instructor: Stuart Russell and Dawn Song --- University of California, Berkeley

Recap: Linear regression



Prediction: $h_w(x) = w_0 + w_1 x$



Berkeley house prices, 2009

Recap: Linear Regression

- What’s the loss function?

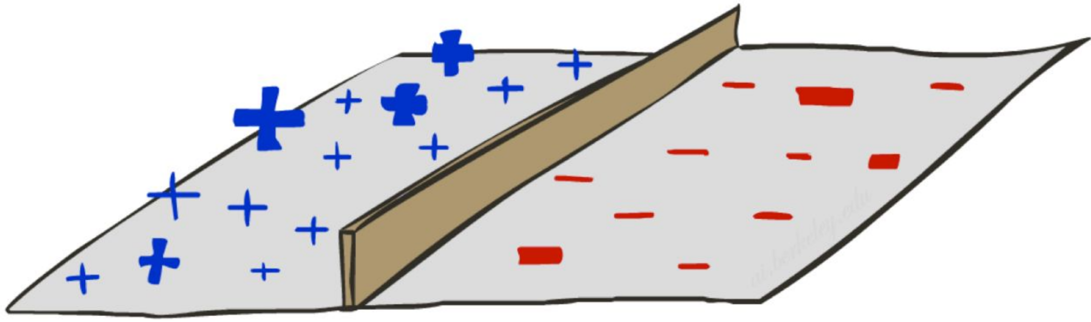
- How to find w* to minimize loss function?

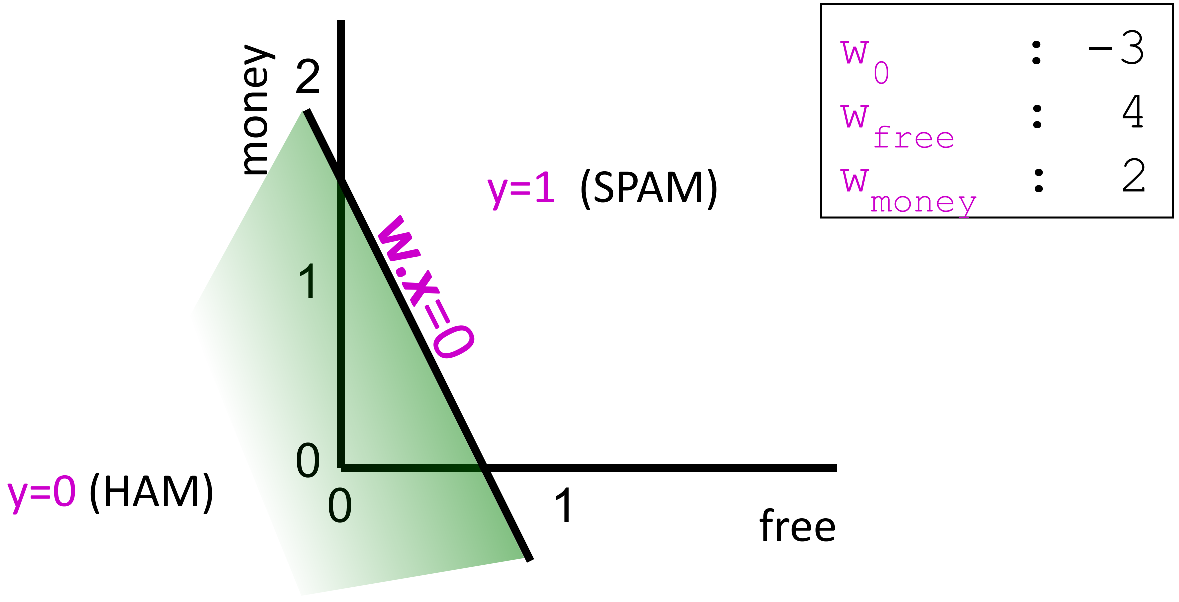
Recap: Linear Regression

- L2 loss function: sum of squared errors over all examples
  - $\text{Loss} = \Sigma_j (y_j - h_w(x_j))^2 = \Sigma_j (y_j - (w_0 + w_1 x_j))^2$
- Find w to minimize loss function (over N examples):
  - $w_1 = [N\Sigma_j x_j y_j - (\Sigma_j x_j)(\Sigma_j y_j)]/[N\Sigma_j x_j^2 - (\Sigma_j x_j)^2]$ and $w_0 = 1/N\ [\Sigma_j y_j - w_1 \Sigma_j x_j]$
- For the general case where x is an n-dimensional vector
  - **X** is the data matrix (all the data, one example per row); **y** is the column of labels
  - $\mathbf{w}^* = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

Recap: Perceptron

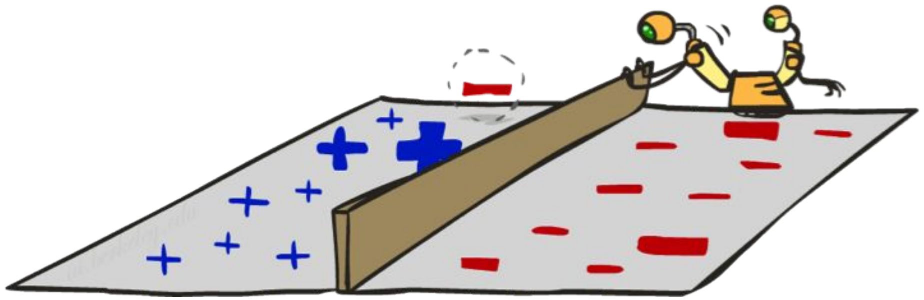
- A ***threshold perceptron*** is a single unit that outputs
  - $y = h_{\mathbf{w}}(\mathbf{x}) = 1$ when $\mathbf{w.x} \geq 0$
    $= 0$ when $\mathbf{w.x} < 0$
- In the input vector space
  - Examples are points $\mathbf{x}$
  - The equation $\mathbf{w.x}=0$ defines a ***hyperplane***
  - One side corresponds to $y=1$
  - Other corresponds to $y=0$
- Quiz:
  - What's the direction for w?



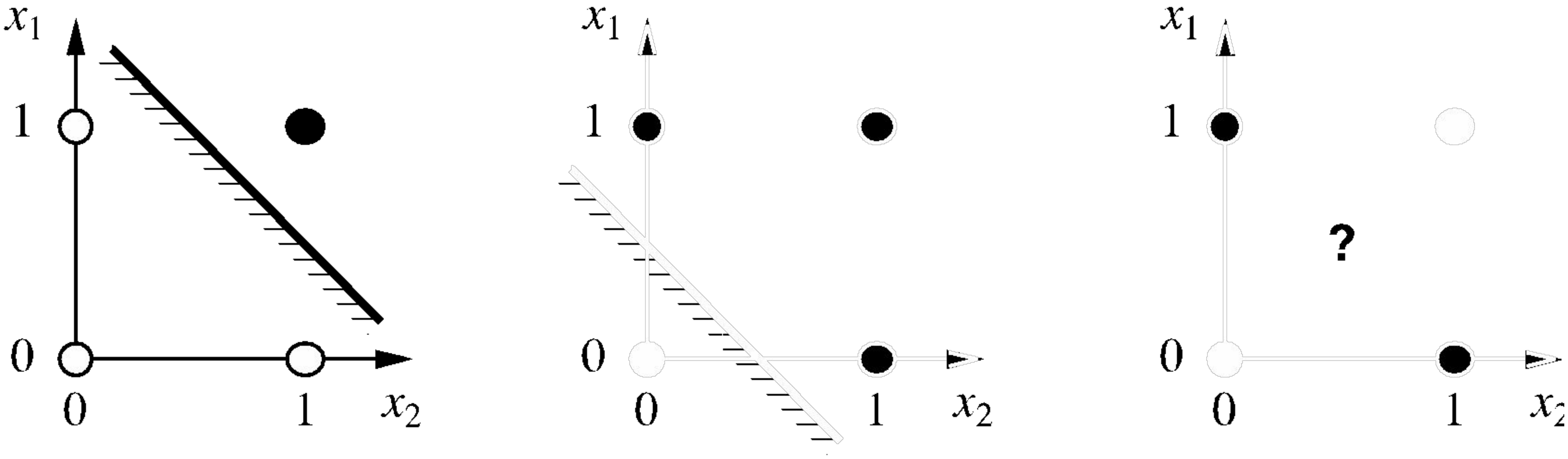


Recap: Perceptron learning rule

- If true $y \neq h_{\mathbf{w}}(\mathbf{x})$ (an error), adjust the weights
- If $\mathbf{w.x} < 0$ but the output should be $y=1$
  - This is called a ***false negative***
  - Should ***increase*** weights on ***positive*** inputs
  - Should ***decrease*** weights on ***negative*** inputs
- If $\mathbf{w.x} > 0$ but the output should be $y=0$
  - This is called a ***false positive***
  - Should ***decrease*** weights on ***positive*** inputs
  - Should ***increase*** weights on ***negative*** inputs
- The ***perceptron learning rule***:
  - $\mathbf{w} \leftarrow \mathbf{w} + \underline{\alpha}\ \underline{(y - h_{\mathbf{w}}(\mathbf{x}))}\ \mathbf{x}$



Perceptrons hopeless for XOR function



(a) x_1 **and** x_2 (b) x_1 **or** x_2 (c) x_1 **xor** x_2

Basic questions

- Which hypothesis space H to choose?
- How to measure degree of fit?
- How to trade off degree of fit vs. complexity?
  - "*Ockham's razor*"
- How do we find a good h?
- How do we know if a good h will predict well?

Classical stats/ML: Minimize loss function

- Which hypothesis space H to choose?
  - *E.g., linear combinations of features*: $h_w(x) = w^T x$
- How to measure degree of fit?
  - *Loss function, e.g., squared error* $\Sigma_j (y_j - w^T x)^2$
- How to trade off degree of fit vs. complexity?
  - *Regularization: complexity penalty, e.g.,* $||w||^2$
- How do we find a good h?
  - *Optimization (closed-form, numerical); discrete search*
- How do we know if a good h will predict well?
  - *Try it and see (cross-validation, bootstrap, etc.)*

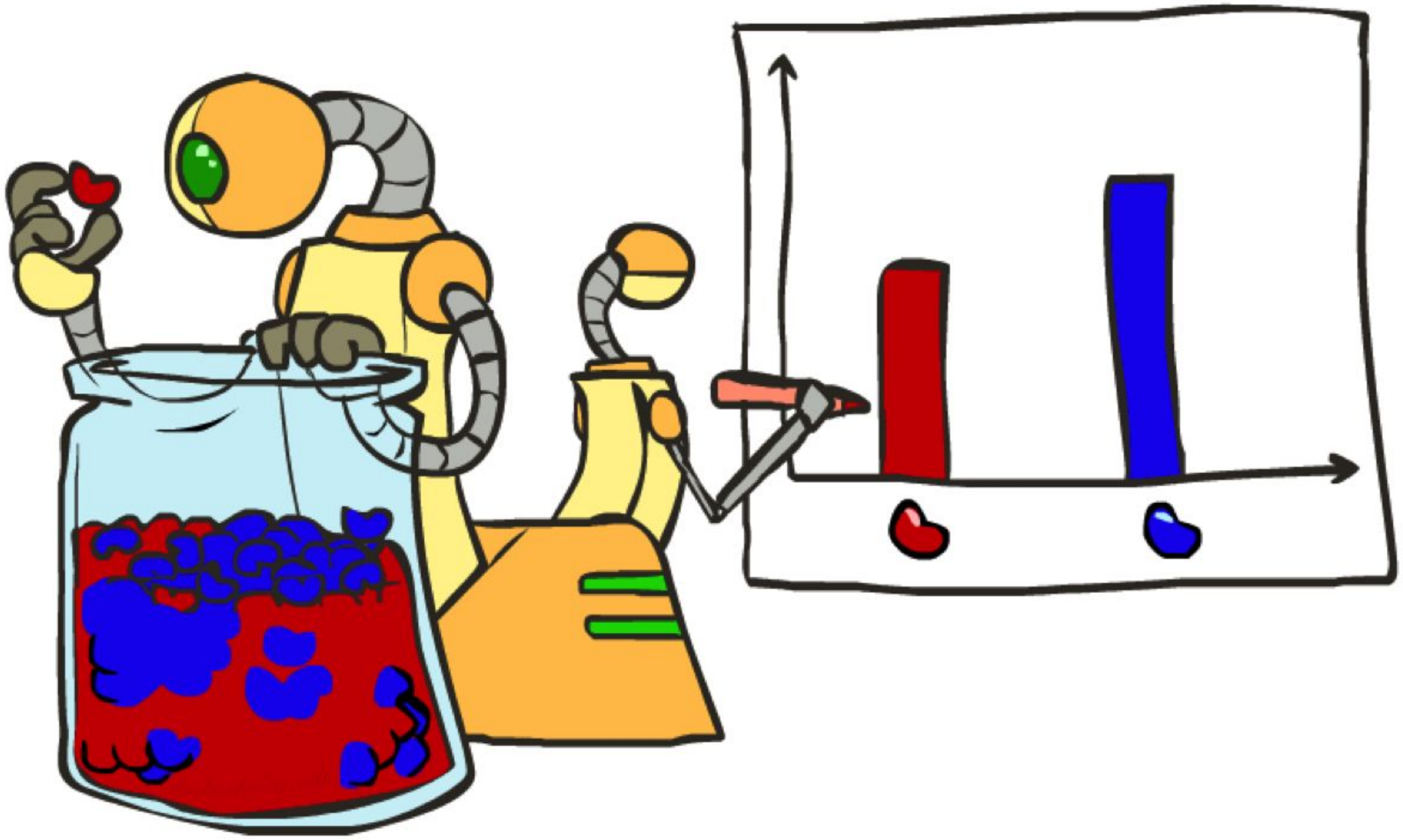
Probabilistic: Max. likelihood, max. a priori

- Which hypothesis space H to choose?
  - *Probability model* P(y | x,h) , e.g., $Y \sim N(w^T x, \sigma^2)$
- How to measure degree of fit?
  - *Data likelihood* $\prod_j P(y_j \mid x_j, h)$
- How to trade off degree of fit vs. complexity?
  - *Regularization or prior*: $\mathrm{argmax}_h\ P(h) \prod_j P(y_j \mid x_j, h)$ (*Max a Priori*)
- How do we find a good h?
  - *Optimization (closed-form, numerical); discrete search*
- How do we know if a good h will predict well?
  - *Empirical process theory (generalizes Chebyshev, CLT, PAC…);*
  - *Key assumption is (i)id*

Bayesian: Computing posterior over H

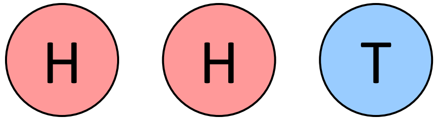
- Which hypothesis space H to choose?
  - *All hypotheses with nonzero a priori probability*
- How to measure degree of fit?
  - *Data probability, as for MLE/MAP*
- How to trade off degree of fit vs. complexity?
  - *Use prior, as for MAP*
- How do we find a good h?
  - ***Don't!*** *Bayes predictor* $P(y|x,D) = \Sigma_h P(y|x,h) P(D|h) P(h)$
- How do we know if a good h will predict well?
  - ***Silly question! Bayesian prediction is optimal!!***

Parameter Estimation



Maximum Likelihood Parameter Estimation

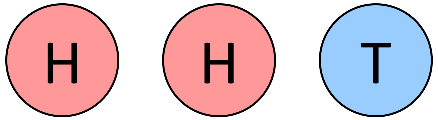
- Estimating the distribution of a random variable
  - E.g., here is a coin; what is the probability θ of heads?
- Evidence $\mathbf{x} = x_1,\ldots,x_N$
  - E.g., three independent coin tosses X_1=heads, X_2=heads, X_3=tails



- Likelihood: probability of the evidence $P(x_1,\ldots,x_N ; \theta)$
  - E.g., $P(X_1$=heads, X_2=heads, X_3=tails ; $\theta)$ = ____________
- Maximum likelihood: What value θ_{ML} maximizes the likelihood?
- Log likelihood: $L(\mathbf{x}; \theta) = \log P(\mathbf{x}; \theta)$
  - E.g., $L(\mathbf{x}; \theta)$ = ________________
- θ_{ML} also maximizes the log likelihood and it's easier to differentiate
- $\partial L/\partial\theta$ = ________________
- θ_{ML} = _________
- For h heads and t tails, θ_{ML} = _____________

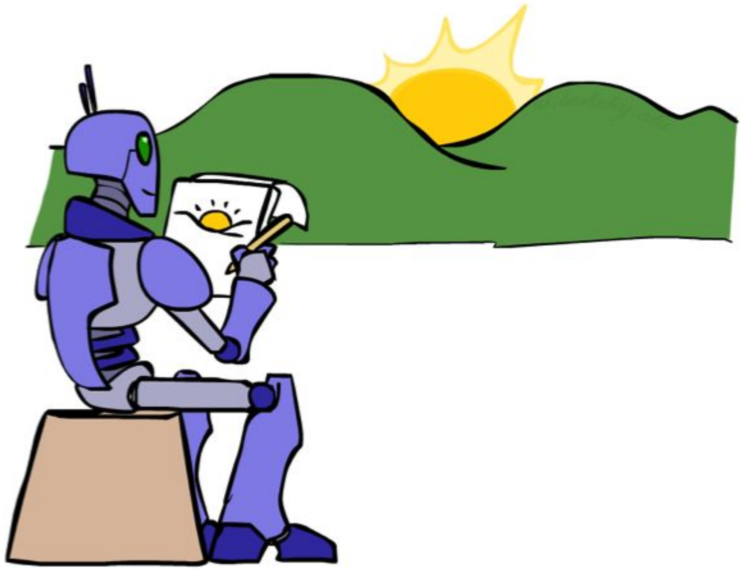
Maximum Likelihood Parameter Estimation

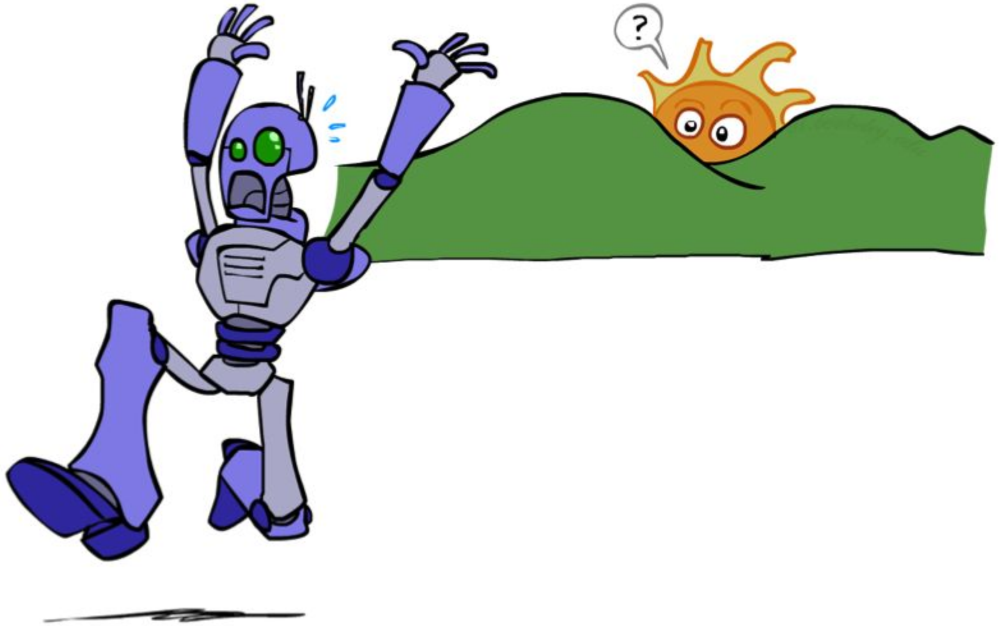
- Estimating the distribution of a random variable
  - E.g., here is a coin; what is the probability θ of heads?
- Evidence $\mathbf{x} = x_1,\ldots,x_N$
  - E.g., three independent coin tosses X_1=heads, X_2=heads, X_3=tails



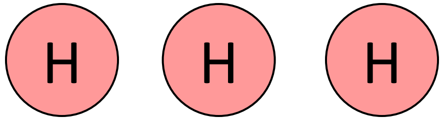
- Likelihood: probability of the evidence $P(x_1,\ldots,x_N ; \theta)$
  - E.g., $P(X_1=\text{heads}, X_2=\text{heads}, X_3=\text{tails} ; \theta) = \theta^2(1-\theta)$
- Maximum likelihood: What value θ_{ML} maximizes the likelihood?
- Log likelihood: $L(\mathbf{x}; \theta) = \log P(\mathbf{x}; \theta)$
  - E.g., $L(\mathbf{x}; \theta) = 2 \log\theta + \log(1-\theta)$
- θ_{ML} also maximizes the log likelihood and it's easier to differentiate
- $\partial L/\partial\theta = 2/\theta - 1/(1-\theta) = 0$
- $\theta_{ML} = 2/3$
- For h heads and t tails, $\theta_{ML} = h/(h+t)$

Unseen Events





Laplace Smoothing



- Suppose we see three heads: is a $\theta_{ML} = 0$ a reasonable estimate?
- Laplace smoothing with strength α:
  - Pretend you saw every outcome α times before starting
  - $\theta_{Lap} = (h+\alpha)/[(h+\alpha) + (t+\alpha)]$
  - $\quad = (3+\alpha)/(3+2\alpha)$
  - In general, for a K-valued variable:
  - $\theta_k = (N_k+\alpha) / \Sigma_k(N_k+\alpha) = (N_k+\alpha) / (N + K\alpha)$
  - For $\alpha >> N$, θ_k tends to $1/K$ (uniform prior)
  - For $\alpha << N$, θ_k tends to N_k /N (ML estimate)

Probabilistic Classification

Example: Spam Filter

- Input: an email
- Output: spam/ham

- Setup:
  - Get a large collection of example emails, each labeled "spam" or "ham"
  - Note: someone has to hand label all this data!
  - Want to learn to predict labels of new, future emails

- Features: The attributes used to make the ham / spam decision
  - Words: FREE!
  - Text Patterns: $dd, CAPS
  - Non-text: SenderInContacts
  - …

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virture of its nature as being utterly confidencial and top secret. …

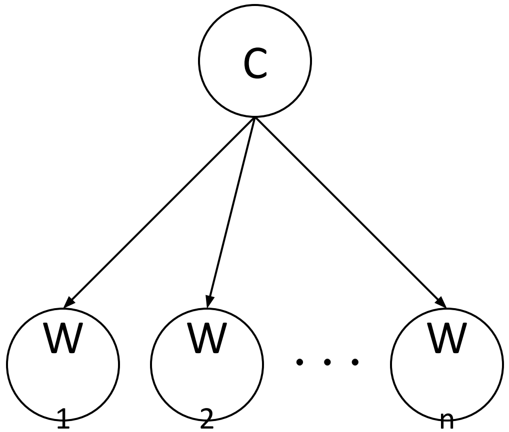
TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99

Ok, Iknow this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.

Bayes net model for ham/spam

- Class C of a document is spam or ham, with prior P(C)
- ***Bag-of-words*** model: Each word W_i in the document is generated independently from a class-specific distribution $P(W_i \mid C)$ over words
- This is an example of a ***naïve Bayes*** model



$$P(C,W_1,\ldots,W_n) = P(C) \prod_i P(W_i \mid C)$$

Inference for Naïve Bayes

- A Naïve Bayes model is a polytree, so solvable in linear time
- To compute posterior distribution for class C given a document:
- $P(C \mid w_1,\ldots,w_n) = \alpha\, P(C,w_1,\ldots,w_n) = \alpha\, P(C) \prod_i P(W_i \mid C)$
- I.e., multiply $n+1$ numbers, for each value of C, then normalize

Computing the class probabilities

	P(w|spam)	P(w|ham)	Cum LogSpam	Cum LogHam
Word	0.33333	0.66666	-1.1	-0.4

$\alpha[e^{-76.0}, e^{-80.5}] = [0.989, 0.011]$

Parameter learning for Naïve Bayes

- We need to estimate the following parameters:
  - $P(C) = [\theta_C, 1-\theta_C]$, the prior over classes
    - ML estimate: relative frequencies in training set
  - $P(W_i \mid C)$, the distribution for each word position given the class
    - For the bag-of-words model, this is the same for all positions
    - Parameters are $\theta_{k|c} = P(W_i{=}k \mid C{=}c)$ for each class c and each dictionary entry k
    - E.g., $\theta_{\text{"you"}|\text{spam}} = 0.00881$ $\quad\theta_{\text{"you"}|\text{ham}} = 0.00304$
    - Estimated by measuring frequency of occurrence in ham and spam
    - Need Laplace smoothing! Many dictionary words may not appear in training set