UT DALLAS

CS 6347

Lecture 11

Basics of Machine Learning

The Course So Far…

- What we've seen:
  - How to compactly model/represent joint distributions using graphical models
  - How to solve basic inference problems
    - Exactly: variable elimination & belief propagation
    - Approximately: LP relaxations, duality, loopy belief propagation, mean field, sampling

Next Goal

- Where we are going:
  - Given independent samples from a joint distribution, we want to estimate the graphical model that produced them
  - In practice, we typically have no idea what joint distribution describes the data
  - There might be lots of hidden variables (i.e., data that we can't or didn't observe)
  - We want the "best" model for some notion of "best"

Machine Learning

- Need a principled approach to solving these types of problems
  - How do we determine which model is better than another?
  - How do we measure the performance of our model on tasks that we care about?
- Many approaches to machine learning rephrase a learning problem as that of optimizing some objective that captures the quantities of interest

Spam Filtering

- Given a collection of emails $E_1, \ldots, E_n$ and labels $L_1, \ldots, L_n \in \{spam, not\ spam\}$ want to learn a model that detects whether or not an email is spam
  - How might we evaluate the model that we learn?

Spam Filtering

- Given a collection of emails $E_1, \ldots, E_n$ and labels $L_1, \ldots, L_n \in \{spam, not\ spam\}$ want to learn a model that detects whether or not an email is spam
  - How might we evaluate the model that we learn?
- This is an example of what is called a supervised learning problem
  - We are presented with labeled data, and our goal is to correctly predict the labels of unseen data

Performance Measures

- Classification: given a set of unseen emails, correctly label them as spam/not spam
  - Classification error defined to be the number of misclassified emails (under the model)
  - Two types of error: training and test
    - Training error: the number of misclassified emails in the labelled training set
    - Test error: the number of misclassified emails in the unseen set

Performance Measures

- Other prediction/inference tasks: choose a loss function that reflects the task you want to solve
- Density estimation: estimate the full joint distribution
  - Error could be defined using the KL divergence between the learned model and the true model
- Structure estimation: estimate the structure of the joint distribution (i.e., what independence properties does it assert)

Machine Learning Terminology

- Overfitting: the learned model caters too much to the data on which it was trained. In the worst case, the learned model corresponds exactly to the training set and assigns probability zero to all unobserved samples
- Generalization: the model should apply beyond the training set to unseen samples (independent of the true distribution)
- Cross-validation: a method of holding out some of the training data in order to limit overfitting and improve generalization
- Regularization: encode a "soft constraint" that prefers simpler models

Bias Variance Tradeoff

- The true model may not be a member of the family of models that we learn
  - Even with unlimited data, we will not recover the true solution
  - This limitation is known as bias
  - We can always choose more complicated models at the expense of computation time
- With only a few samples, many models might be a good fit
  - Small changes in the samples may result in significantly different models
  - This type of limitation is referred to as variance

The Learning Problem

- Given iid samples $x^1, \ldots, x^M$ from some probability distribution find the graphical model that best represents the samples from some family of graphical models
- This could entail
  - Structure learning: if the graph structure is unknown, we would need to learn it
  - Parameter learning: learn the parameters of the model (the parameters usually control the allowable potential functions)

Maximum Likelihood Estimation

- Fix a family of parameterized distributions
  - Each choice of the parameters produces a different distribution
  - Example: for the coloring problem on a graph G, we could treat the weights as parameters
- Given samples $x^{(1)}, \ldots, x^{(M)}$ from some unknown distribution and parameters θ…
  - The likelihood of the data is defined to be $l(\theta) = \prod_m p(x^{(m)}|\theta)$
  - Goal: find the θ that maximizes the log-likelihood
  - Example: given samples of colorings of a graph G, find the weights that maximize the likelihood of observing these colorings

Simple MLE

- A biased coin is described by a single parameter b which corresponds to the probability of seeing heads

- Given the set of samples H, H, H, H, T use MLE to estimate b

(worked out on the board)

Bayesian Inference

- MLE assumes that there exists some joint distribution $p(x, \theta)$ over possible observations and choices of the parameters, but only works with the conditional distribution $p(x|\theta)$
  - In practice, this is much easier than dealing with the whole joint distribution
  - In the coin flipping example
    - If we are told the bias, we can compute the probability that a coin comes up heads
    - To compute the joint probability, $p(x|\theta)p(\theta)$ we would need to choose a probability distribution over the biases

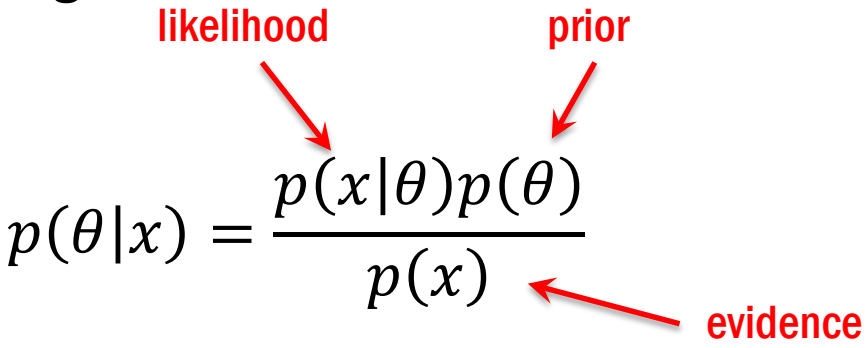
Bayesian Inference

- We could also consider the posterior probability distribution of the parameters given the evidence

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}$$

Bayesian Inference

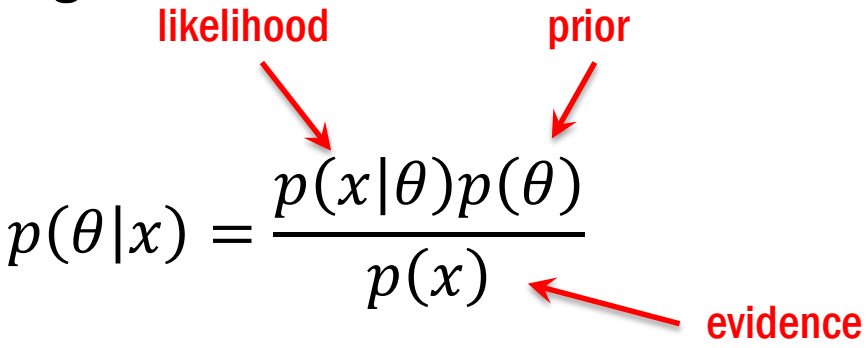
- We could also consider the posterior probability distribution of the parameters given the evidence



- Prior captures our previous knowledge about the parameters

Bayesian Inference

- We could also consider the posterior probability distribution of the parameters given the evidence



- Prior captures our previous knowledge about the parameters
- Bayesian inference computes the posterior probability distribution over θ given the observed samples
- MAP inference maximizes the posterior probability over θ

Simple MAP Inference

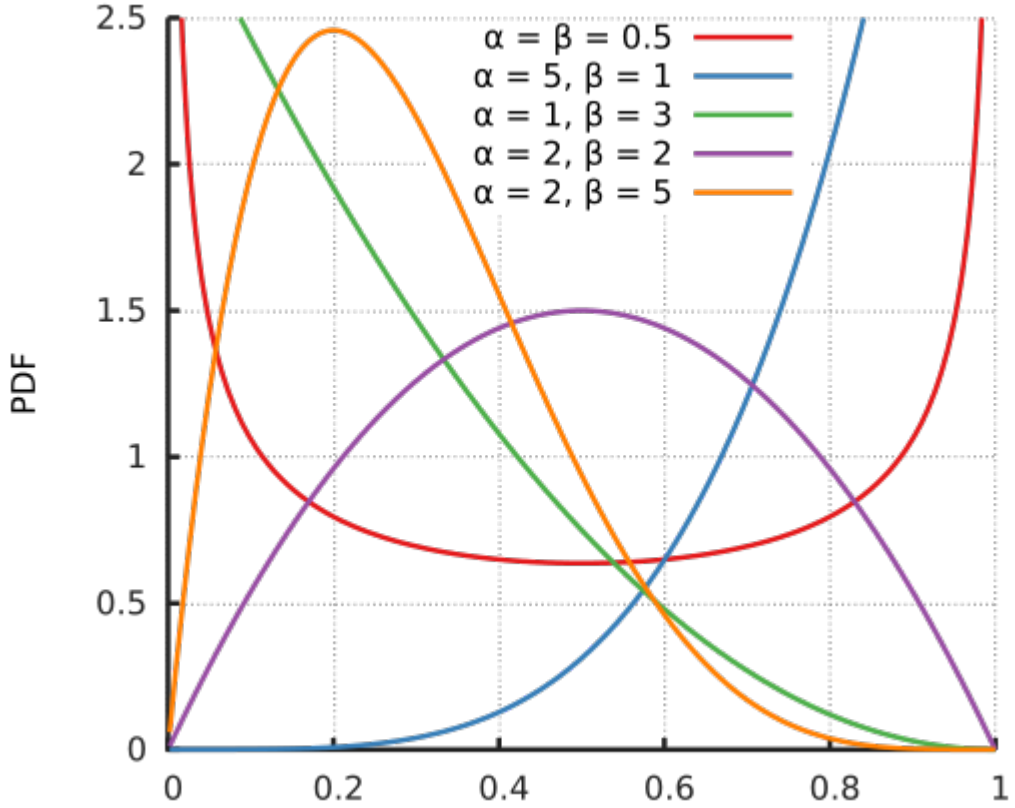
- A biased coin is described by a single parameter b which corresponds to the probability of seeing heads
- Given the set of samples H, H, H, H, T use MAP inference to estimate b
- What prior distribution should we pick for $p(b)$?

Simple MAP Inference

- A biased coin is described by a single parameter b which corresponds to the probability of seeing heads
- Given the set of samples H, H, H, H, T use MAP inference to estimate b
- What prior distribution should we pick for $p(b)$?
  - Uniform on $[0,1]$
  - Beta distribution: $p(b) \propto b^{\alpha-1}(1-b)^{\beta-1}$

(worked out on the board)

Beta Distribution



source: Wikipedia

Simple MAP Inference

- A biased coin is described by a single parameter b which corresponds to the probability of seeing heads
- Given the set of samples H, H, H, H, T use MAP inference to estimate b
- What prior distribution should we pick for $p(b)$?
- MAP inference with a uniform prior is equivalent to maximum likelihood estimation
  - Prior can be viewed as a certain kind of regularization: it preferences parameters that occur with high probability under the prior