CSCC11 Week 7 Notes

Review of Baye's Rule:

- Baye's Rule: $P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$

Introduction to Estimation:

- Estimation is determining the values of some unknown variables from observed data. I.e. It is finding a single estimate of the value of an unknown parameter.

- There are 2 main types of estimation:
  1. Maximum Likelihood Estimation (MLE)
  2. Bayesian Estimation

Maximum Likelihood Estimation (MLE):

- Uses the frequentist view/ Frequentist approach which says that an event's probability is the limit of its relative frequency in many trials.

- It uses $P(D|M)$ where D is the data and M the model. It's too focused on the training data.

- Recall the likelihood function if all data points are iid is $L(\theta|D) = f(D|\theta)$

$$= \prod_{i=1}^{N} f(x_i|\theta)$$

- To find the maximum likelihood we would do:

$$\hat{\theta} = \underset{\theta}{\arg\max} \left(\prod_{i=1}^{N} f(x_i \mid \theta) \right)$$

$$= \frac{\partial}{\partial \theta} \left(\prod_{i=1}^{N} f(x_i \mid \theta) \right)$$

However, this is not ideal because getting or taking the derivative of products is very messy.

Instead, we will use the log likelihood function. We can do this because:

1. The log of a product is the sum of logs.
2. Taking the log of any function may change its values but does not change where the max of that function occurs.

Log Likelihood: $\ell(\theta) = \ln(f(D \mid \theta))$

$$= \ln\left(\prod_{i=1}^{N} f(x_i \mid \theta) \right)$$

$$= \sum_{i=1}^{N} \ln(f(x_i \mid \theta))$$

$$\therefore \hat{\theta} = \underset{\theta}{\arg\max} \; \ell(\theta)$$

Bayesian Estimation:

- Uses the Bayesian View/Bayesian Approach where probability is defined as a degree of belief in an event. This degree of belief may be based on prior knowledge.

  I.e. The idea behind Bayesian Estimation is that before we've seen any data, we already have some prior knowledge about the distribution it came from. Such prior knowledge comes from experience or past experiments.

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

$$= \frac{P(D|\theta)\,P(\theta)}{\int P(D|\theta)\,P(\theta)\,d\theta} \longrightarrow P(D) = \int P(D|\theta) \cdot P(\theta)\,d\theta$$

$P(\theta|D)$ is called the posterior distribution and it describes our knowledge of the model based on both the data and the prior.

$P(D|\theta)$ is called the likelihood distribution/function describes the likelihood of the observations assuming the data is correct.

$P(\theta)$ is called the prior distribution and it describes our assumptions about the model without having observed any data.

$P(D)$ is called the evidence. It is usually expensive to calculate and is used to normalize the posterior.

- The problem with MLE is that if there's too little data, it can overfit.

- In MLE, the observations/data, D, is treated as random vars but the parameters, θ, are not.

  In Bayesian Estimation, both the data and observation are treated as random variables.

- Since MLE depends solely on observed data, it can overfit if the data is minimal.

  E.g. Suppose I flip a fair coin 3 times and on all 3 times, it lands on heads. Then, MLE tells you that $P(H)=1$ and $P(T)=0$.

  In situations where data is sparse, having some prior information can help.

  However, unreliable priors can lead to biased models, so make sure they are well defined.

- If the Bayesian prior is non-informative, for example it is uniform over all values, then the Bayesian prediction will be very similar, if not equal, to MLE predictions.

More on Estimation:

- Estimation: Interested in finding point estimates
- Inference: Interested in $P(M|D)$.
- In this course we focus on 3 types of estimations:
  1. MLE
  2. Maximum a Posteriori (MAP)
  3. Bayes' Estimate

Maximum a Posteriori (MAP):

- $$\hat{\theta} = \underset{\theta}{\arg\max}\ P(\theta|D)$$

  $$= \underset{\theta}{\arg\max}\ P(D|\theta) \cdot \underbrace{P(\theta)}_{\text{Prior}}$$

- Note: We don't need $P(D)$ for MAP since it doesn't depend on θ and may therefore be treated as a constant for MAP estimation.

- Note: If the prior, $P(\theta)$, is uninformative (I.e. It is uniform), then MAP becomes MLE. I previously referenced this on page 4.

- $$\hat{\theta} = \underset{\theta}{\arg\max}\ P(D|\theta) \cdot P(\theta)$$

  $$= \underset{\theta}{\arg\min}\ -\ln(P(D|\theta) \cdot P(\theta))$$

  $$= \underset{\theta}{\arg\min}\ -\ln(P(D|\theta)) - \ln(P(\theta))$$

- Both MLE and MAP are optimization problems.

- Furthermore, both MAP and MLE ignore uncertainty in the parameters. This means we are choosing to put all our faith in the most probable model, but sometimes, this has surprising and undesirable consequences.

Bayes' Estimate:

- Formula: $\theta_{Bayes} = \int p(\theta|D) \cdot \theta \, d\theta$
$$= E_{\underbrace{p(\theta|D)}_{\uparrow}} [\theta]$$

This subscript is used to be explicit about which distribution we're using.

Class Conditionals:

- Here, we're modelling the distribution over the features themselves. These models are called generative models.
- E.g. In the case of binary classification, suppose we have 2 mutually-exclusive classes C_1 and C_2. The prior probability of a data vector coming from class C_1 is $P(C_1) \equiv P(y=c_1)$ and $P(C_2) \equiv P(y=C_2)$
$\equiv 1 - P(y=c_1)$

Each class has its own distribution for the feature vectors, specifically $P(x|C_1)$ and $P(x|C_2)$. (These are the data likelihood distributions for the 2 classes.)

Then, the prob of a data point can be written as:

$$P(x) = P(x, C_1) + P(x, C_2)$$
$$= P(x|C_1)P(C_1) + P(x|C_2)P(C_2)$$

- If one had such a model, one could draw data samples from the model in the following way:
  1. One would randomly choose a class according to the probabilities $P(C_1)$ and $P(C_2)$.

  2. Then, conditioned on the class, one can sample a data point, x, from the associated likelihood distribution.

- For the learning problem we are given a set of labelled training data $\{(x_i, y_i)\}$ and our goal is to learn the parameters of the generative model.

  I.e. We want to:
  1. Estimate the conditional likelihood distribution for each class.
  2. Estimate $P(C_1)$ by computing the ratio of the number of elements of class 1 to the total number of elements.

- Once we have learned the parameters of our generative model, we perform classification by comparing the posterior class probabilities: $P(C_1|x) > P(C_2|x)$?

  If $P(C_1|x) > P(C_2|x)$, then we classify the input as to belonging to class C_1.

  If $P(C_1|x) < P(C_2|x)$, then we classify the input as C_2.

  If $P(C_1|x) = P(C_2|x)$, then it's on the decision boundary.

Equivalently, we can compare their ratio to 1.

I.e. $\frac{P(C_1|x)}{P(C_2|x)} = 1 \rightarrow$ On decision boundary

$$\frac{P(C_1|x)}{P(C_2|x)} > 1 \rightarrow \text{Classify as } C_1$$

$$\frac{P(C_1|x)}{P(C_2|x)} < 1 \rightarrow \text{Classify as } C_2$$

- $P(C_i|x) = \frac{P(X|C_i) \cdot P(C_i)}{P(x)}$ ← Bayes' Rule

$$\frac{P(C_1|x)}{P(C_2|x)} = \frac{\frac{P(X|C_1) \cdot P(C_1)}{\cancel{P(x)}}}{\frac{P(X|C_2) \cdot P(C_2)}{\cancel{P(x)}}}$$

$$= \frac{P(X|C_1) \cdot P(C_1)}{P(X|C_2) \cdot P(C_2)}$$

- Note: These computations are typically done in the logarithmic domain as its faster and more numerically stable.

I.e. we check if $\log\left(\frac{P(C_1|x)}{P(C_2|x)}\right) > 0$

- Recap:
  - Class conditionals are used for generative models.
  - Want to model $P(x) = P(x, C_1) + P(x, C_2)$
  - $P(x, C_i) = \underbrace{P(C_i)}_{\text{Prior}} \cdot \underbrace{P(x \mid C_i)}_{\text{Likelihood}}$

    $= \underbrace{P(x)}_{\text{Evidence}} \cdot \underbrace{P(C_i \mid x)}_{\text{Posterior}}$
  - $$P(C_i \mid x) = \frac{P(x \mid C_i) \cdot P(C_i)}{P(x)}$$

    $$= \frac{P(x \mid C_i) \cdot P(C_i)}{\sum_{i=1}^{2} P(x \mid C_i) P(C_i)}$$

  Can be used to classify input x

- How to find/learn the priors:

  $$P(C_1) = \frac{\#\text{ of class } C_1}{N}$$

  $$P(C_2) = \frac{\#\text{ of class } C_2}{N}$$

- How to find/learn the likelihoods:
  1. Partition based on class
  2. Use all X_i's s.t. $y_i = c_j$ to learn $P(x \mid c_j)$. This is usually done via MLE.

- Class conditions can:
  1. Make predictions
  2. Generate data
     - → 1. Sample class $\hat{c}$ from prior $p(c)$.
     - → 2. Sample data $\hat{x}$ from likelihood $p(x \mid \hat{c})$.

Gaussian Class Conditionals:

- Also called Linear Discriminate Analysis (LDA).
- Assume likelihoods to be Gaussian.
- For each class i, we model the i^{th} likelihood to be:

$$N(\vec{x}, \underbrace{\vec{\mu}_i}_{\text{Mean}}, \underbrace{\Sigma_i}_{\text{Covariance}}) = \frac{1}{\sqrt{2\pi|\Sigma|^d}} \exp\left(\frac{-1}{2}(\vec{x}-\vec{\mu}_i)\Sigma_i^{-1}(\vec{x}-\vec{\mu}_i)\right)$$

Hence, $P(X_{1:N}|\mu, \Sigma) = \prod_{i=1}^{N} P(X_i|\mu, \Sigma)$

$$= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi|\Sigma|^d}} \exp\left(\frac{-1}{2}(\vec{x}-\vec{\mu}_i)\Sigma_i^{-1}(\vec{x}-\vec{\mu}_i)\right)$$

Note: d is the dimensionality of the x_i's.

- It is often easier to find the min log likelihood.

$$L(\mu, \Sigma) = -\ln(p(X_{1:N}|\mu, \Sigma))$$
$$= -\sum_i \ln(p(x_i|\mu, \Sigma))$$
$$= \sum_i \frac{(x_i-\mu)^T\Sigma^{-1}(x_i-\mu)}{2} + \frac{N}{2}\ln|\Sigma| + \frac{Nd}{2}\ln(2\pi)$$

Solving for μ and Σ involves setting $\frac{\partial L}{\partial \mu} = 0$ and $\frac{\partial L}{\partial \Sigma} = 0$.

$$\mu^* = \frac{\sum_i x_i}{N}, \quad \Sigma^* = \frac{\sum_i (x_i-\mu^*)(x_i-\mu^*)^T}{N}$$

- For the decision boundary, for simplicity, assume $P(C_1) = P(C_2) = \frac{1}{2}$.
  Then, the decision boundary occurs at $d(x) = 0$.

Recall that:

$$d(x) = \ln\left(\frac{P(C_1|x)}{P(C_2|x)}\right)$$

$$= \ln\left(\frac{P(x|C_1)\cdot \cancel{P(C_1)}}{P(x|C_2)\cdot \cancel{P(C_2)}}\right) \quad \frac{P(C_1)}{P(C_2)} = 1 \text{ since } P(C_1)=P(C_2)=\frac{1}{2}$$

$$= \ln(P(x|C_1)) - \ln(P(x|C_2))$$

$$= \underbrace{\ln\left(\frac{1}{\sqrt{2\pi}\,|\Sigma_1|}\right) - \ln\left(\frac{1}{\sqrt{2\pi}\,|\Sigma_2|}\right)}_{\leftarrow \text{ Constants w.r.t. } \vec{x}}$$

$$\underbrace{- \frac{1}{2}(\vec{x}-\vec{\mu_1})^T \Sigma_1^{-1} (\vec{x}-\vec{\mu_1}) + \frac{1}{2}(\vec{x}-\vec{\mu_2})^T \Sigma_2^{-1} (\vec{x}-\vec{\mu_2})}_{\text{Quadratic w.r.t. } \vec{x}}$$

$$= 0$$

$d(x)$ is a quadratic function.

If $\Sigma_1 = \Sigma_2$, then we have a linear decision boundary.