COMPSCI 671D: Homework 2 Solutions

January 2019

1 Maximum Entropy Configuration

a) Consider flipping three different (weighted) coins. The first coin lands on heads with probability $p = 0.5$ while the other two land on heads with probabilities $p = 0.7$ and $p = 0.1$ respectively. Calculate the entropy of the flipping results. Which coin has the largest entropy?
Solution:

1 . $H_1 = -[0.5log(0.5) + 0.5log(0.5)] = 1$

2 . $H_2 = -[0.7log(0.7) + 0.3log(0.3)] = 0.8813$

3 . $H_3 = -[0.1log(0.1) + 0.9log(0.9)] = 0.4690$

$H_1 > H_2 > H_3$, the first coin has the largest entropy.

b) Show that for a discrete random variable X with n possible values and probability mass function $p(X)$, its maximum entropy is $log(n)$. (Hint: use Jensen's Inequality: if X is a random variable and ϕ is a convex function, then $\phi(E[X]) \leq E[\phi(X)]$).
Solution:

$$H(X) = -\sum_{i=1}^{n} p(X_i)log(p(X_i)) = \sum_{i=1}^{n} p(X_i)log(\frac{1}{p(X_i)})$$

$$\leq log(\sum_{i=1}^{n} p(X_i)\frac{1}{p(X_i)}) = log(n)$$

The maximum is achieved when $p(X_i) = \frac{1}{n}$. Note that $log(x)$ is concave instead of convex, so we should reverse the two sides of Jensen's Inequality.

c) The concept of entropy you have already seen can be extended to continuous random variables; this extension is known as **differential entropy**. It is given by the formula

$$H(x) = -\int p(x)log(p(x))dx.$$

Show that the normal distribution

$$g(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

gives the maximum differential entropy among all arbitrary probability density functions $f(x)$ of a random variable x with variance σ^2. (Hints: show that $H(g(x)) = -\int_{-\infty}^{\infty} f(x)log(g(x))dx$ and then prove $H(g(x)) - H(f(x)) \geq 0$ using Jensen's Inequality).
(P.S. You may also solve b) and c) using Lagrange Multipliers if you know how to do that.)
Solution:
Assume that $g(x)$ and $f(x)$ have the mean since entropy is translation invariant.

$$\begin{aligned}
H(g(x)) &= -\int_{-\infty}^{\infty} g(x)log(g(x))dx \\
&= -\int_{-\infty}^{\infty} g(x)log(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}})dx \\
&= -\int_{-\infty}^{\infty} g(x)log(\frac{1}{\sqrt{2\pi\sigma^2}})dx - \int_{-\infty}^{\infty} g(x)\frac{(x-\mu)^2}{2\sigma^2}dx \\
&= -log(\frac{1}{\sqrt{2\pi\sigma^2}})\int_{-\infty}^{\infty} g(x)dx - \frac{1}{2\sigma^2}\int_{-\infty}^{\infty} g(x)(x-\mu)^2dx \\
&= -log(\frac{1}{\sqrt{2\pi\sigma^2}})\int_{-\infty}^{\infty} g(x)dx - \frac{1}{2\sigma^2}\int_{-\infty}^{\infty} g(x)(x-\mu)^2dx \\
&= -log(\frac{1}{\sqrt{2\pi\sigma^2}}) \cdot 1 - \frac{1}{2\sigma^2} \cdot \sigma^2 \\
&= -log(\frac{1}{\sqrt{2\pi\sigma^2}})\int_{-\infty}^{\infty} f(x)dx - \frac{1}{2\sigma^2}\int_{-\infty}^{\infty} f(x)(x-\mu)^2dx \\
&= -\int_{-\infty}^{\infty} f(x)log(g(x))dx
\end{aligned}$$

Therefore,

$$\begin{aligned}
H(g(x)) - H(f(x)) &= -\int_{-\infty}^{\infty} f(x)log(g(x))dx + \int_{-\infty}^{\infty} f(x)log(f(x))dx \\
&= -\int_{-\infty}^{\infty} f(x)log(\frac{g(x)}{f(x)})dx \\
&\geq -log(\int_{-\infty}^{\infty} f(x)\frac{g(x)}{f(x)}dx) = 0
\end{aligned}$$

As a result, for any probability density function $f(x)$ with fixed variance σ^2, we have $H(g(x)) \geq H(f(x))$.

2 Bias and Variance of Random Forest

a) Suppose we have some observed data X with corresponding observed labels Y. We know $Y = f(x) + \epsilon$ where ϵ has mean 0 and variance σ^2. Now we want to approximate $f(x)$ with $\hat{f}(x)$. Derive the bias-variance decomposition of mean square error; that is, show that

$$E[(y - \hat{f}(x))^2] = (\text{Bias}[\hat{f}(x)])^2 + \text{Var}[\hat{f}(x)] + \sigma^2$$

where

$$\text{Bias}[\hat{f}(x)] = E[\hat{f}(x) - f(x)]$$

and

$$\text{Var}[\hat{f}(x)] = E[(\hat{f}(x) - E[\hat{f}(x)])^2]$$

Solution:

$$E[(y - \hat{f}(x))^2] = E[(f(x) + \epsilon - \hat{f}(x) + E\hat{f}(x) - E\hat{f}(x))^2]$$

$$= E[(f(x) - E\hat{f}(x))^2] + E[\epsilon^2] + E[(\hat{f}(x) - E\hat{f}(x))^2]$$

$$+2E[\epsilon(f(x) - E\hat{f}(x))] + 2E[\epsilon(\hat{f}(x) - E\hat{f}(x))] + 2E[(f(x) - E\hat{f}(x))(\hat{f}(x) - E\hat{f}(x))]$$

We can derive that

$$E[(f(x) - E\hat{f}(x))^2] = (f(x) - E\hat{f}(x))^2 = \text{Bias}[\hat{f}(x)]^2$$

$$E[\epsilon^2] = \text{Var}[\epsilon] + E[\epsilon]^2 = \sigma^2$$

$$E[(\hat{f}(x) - E\hat{f}(x))^2] = \text{Var}[\hat{f}(x)]$$

$$E[\epsilon(f(x) - E\hat{f}(x))] = E[\epsilon](f(x) - E\hat{f}(x)) = 0$$

$$E[\epsilon(\hat{f}(x) - E\hat{f}(x))] = E[\epsilon]E[\hat{f}(x) - E\hat{f}(x)] = 0$$

$$E[(f(x) - E\hat{f}(x))(\hat{f}(x) - E\hat{f}(x))] = (f(x) - E\hat{f}(x))(E\hat{f}(x) - E\hat{f}(x)) = 0$$

As a result,

$$E[(y - \hat{f}(x))^2] = (\text{Bias}[\hat{f}(x)])^2 + \text{Var}[\hat{f}(x)] + \sigma^2$$

b) Show that the decision tree algorithm can achieve zero bias. In other words, show that for a given dataset $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_n, y_n)\}$ with no collision, i.e. there is no $1 \leq i < j \leq n$ such that $\mathbf{x}_i = \mathbf{x}_j$ while $y_i \neq y_j$, there will always exist a decision tree whose training error is 0. Note that the decision tree under consideration here is not a single algorithm such as CART, but rather the hypothesis space of all possible decision trees. (Hint: start from $n = 1$ and proceed by induction).
Solution:
When $n = 1$, it's trivial. Suppose the result holds for $n = k$. In other words, we can find a decision tree that achieves 0 training error on dataset with size

less or equal to k. Now consider a dataset sized k+1. In this new dataset we can definitely find a feature on which at least two data points have different values. Now we can split on this feature and the k+1 data points will be partitioned into more than 2 groups each with size less or equal to k. For each of them, we can find a perfect decision tree. Then we can replace the leaves of the root with these trees to get a new decision tree achieving 0 training error.

c) Given N identically distributed random variables $X_1, X_2, ..., X_N$ each with variance σ^2 and pairwise correlation coefficient $\rho > 0$, obtain the variance of the random variable $\bar{X}$ where $\bar{X}$ is defined to be the average of the N X_i's.
Solution:

$$\begin{aligned}
\mathrm{Var}(\bar{X}) &= \mathrm{Var}(\frac{\sum_{i=1}^{n} X_i}{n}) \\
&= \frac{1}{n^2}\mathrm{Var}(\sum_{i=1}^{n} X_i) \\
&= \frac{1}{n^2}(\sum_{i=1}^{n}\mathrm{Var}(X_i) + \sum_{i=1}^{n}\sum_{j=1, j\neq i}^{n}\mathrm{Cov}(X_i, X_j)) \\
&= \frac{1}{n^2}(n\sigma^2 + n(n-1)\rho\sigma^2) \\
&= \rho\sigma^2 + \frac{(1-\rho)\sigma^2}{n}
\end{aligned}$$

d) Using inspiration from the result of the previous question, explain why random forests uses bagging (averaging of models) and random subspaces. In addition, provide justification for why you think decision trees are usually chosen as base classifiers over other types of base classifiers? (Your explanations should focus on how these methods reduce variance by reducing a certain term in the previous question.)
Solution:
Bagging increases n and consequently decreases the term $\frac{(1-\rho)\sigma^2}{n}$. Random subspaces make different trees look at different features which help decorrelate the classifiers and therefore decrease the $\rho\sigma^2$. There are two good properties of decision trees. First, decision trees can achieve 0 bias which enables the reduction of variance really improve the performance. Second, decision tree without pruning tend to "fit the noise". When they are trained on different part of a dataset, they are less likely to be correlated. Using decision tree as base classifier decreases ρ.

3 Variable Importance of Recidivism Prediction

In this question, we will try to replicate a controversial analysis by the ProPublica news organization (Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing, 2016). ProPublica's main claim was that the COMPAS model used throughout the US court system depends on race, even after conditioning on age and criminal history. In other words, they claimed that a model that predicted whether someone would be arrested depended on their race, given age and criminal history. This caused a huge commotion and now ProPublica's work is often cited in popular books and in the media - many people you meet will know about this conclusion. A preprocessed version of the ProPublica dataset is in the attachment, and you can use it in the following tasks. For this problem, you may use any programming language you like and any toolkit supporting random forest and variable importance. You don't need to tune the parameters, using the default parameter given by the toolkit should be fine.

a) Train a random forest on the ProPublica dataset to predict recidivism. You will predict whether someone is arrested within 2 years based on their age, criminal history, and race. Use 4/5 of the dataset to train the model and the rest to test, what's the test accuracy and f1 score?

b) Measure the variable importance of the random forest you have just trained for the race variable. Is race important in your model? Here, we want to know whether race is important given age and criminal history.

c) Remove the two features with the highest variable importance (but don't remove race), does this change the relative importance of race? If you remove only one of these two variables, does the variable importance of race change?

d) Now train a linear classifier on the dataset and print the coefficients of all features. People often use the values of the coefficient of a variable in a linear model to indicate whether the variable is important. In fact, this is exactly what ProPublica did. Is the scale of these coefficients consistent with the result you got from earlier parts of this question?

e) Do you agree with ProPublica's analysis? Provide a discussion. If your analysis agrees, state how you came to that conclusion. If your analysis disagrees, state where you think ProPublica went wrong. (Keep in mind: could it really be that this famous well-cited study is incorrect? You are welcome to do a literature search on this topic if you think it would be helpful.)
Solution(e):
Not agree! Applying linear model directly on the dataset can be problematic. Because
(1) The results may be predictable from some features nonlinearly but linear models can only catch the linear correlation.
(2) The features are in different scales, one should standardize it before using linear model
(3) Different features can be correlated to each other(It's called confounding).