COMS 4771 HW1

Due: Sun Feb 24, 2019 at 11:59pm

You are allowed to write up solutions in groups of (at max) three students. These group members don't necessarily have to be the same from previous homeworks. Only one submission per group is required by the due date on Gradescope. Name and UNI of all group members must be clearly specified on the homework. No late homeworks are allowed. To receive credit, a typesetted copy of the homework pdf must be uploaded to Gradescope by the due date. You must show your work to receive full credit. Discussing possible solutions for homework questions is encouraged on piazza and with peers outside your group, but every group must write their own individual solutions. You should cite all resources (including online material, books, articles, help taken from specific individuals, etc.) you used to complete your work.

1 **[A simple property of the gradient]** If $f : \mathbb{R}^d \to \mathbb{R}$ is a differentiable function, then one often hears that the gradient

$$\nabla f(x) = \Big(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_d}\Big)$$

points in the direction of steepest ascent of f provided that $\nabla f \neq 0$. In other words, this says for any unit vector u one has the bound

$$\frac{d}{dt} f(x+tu)\Big|_{t=0} \leq \frac{d}{dt} f(x+tv)\Big|_{t=0}$$

where $v = \nabla f(x)/\|\nabla f(x)\|$. Prove this inequality and show that it is strict unless $u = v$.

(hint: show that $(d/dt)f(x+tu)|_{t=0} = \nabla f(x) \cdot u$, and use the fact that for any two vectors x and y, we have: $x \cdot y \leq \|x\|\|y\|$.)

2 **[Designging optimal classifiers]** In this problem you are going to show what are the optimal binary classifier for some other popular error functions. Recall from class that the Bayes classifier is the optimal solution to

$$\min_f \;\; \mathbb{E}_{(X,Y)} \big[\mathbf{1}\big[f(X) \neq Y\big]\big], \tag{1}$$

where $f : \mathcal{X} \to \{-1, 1\}$ is a binary classifier.

However we can propose other objective functions.

(i) Assume we are trying to find the optimal binary classifier for the following error function

$$\min_f \;\; \mathbb{E}_{(X,Y)}\big[\max\{0, 1 - Yf(X)\}\big]$$

How is this problem similar to Eq. (1)?

(Hint: think of the cases where $f(X)$ and Y match.)

(ii) Similar to the proof seen in class, condition on $X = x$ and express the previous expectation as the sum of two terms.

(iii) What relationship between $\Pr[Y = 1|X = x]$ and $\Pr[Y = -1|X = x]$ needs to hold for $f^*(x) = 1$?

(iv) Argue that in general

$$f^*(x) = \text{sign}\left(\Pr[Y = 1|X = x] - \frac{1}{2}\right).$$

How does this function relate to the optimal Bayes Classifier?

(v) Now suppose that we are tying to optimize

$$\min_f \ \mathbb{E}_{(X,Y)}\left[(1 - Yf(X))^2\right].$$

Here we allow f to output real values (relaxing the notion of f being a classifier). Using the same ideas as before, show that in this case

$$f^*(x) = 2\Pr[Y = 1|X = x] - 1.$$

3 **[Strange consequences of high dimensionality]** As discussed in class, we often represent our data in high dimensions. Thus to understand our data better and design effective prediction algorithms, it is good to understand how things behave in high dimensions. Obviously, since we cannot visualize or imagine high dimensional spaces, we often tend to rely on how data behave in one-, two- or three-dimensions and extrapolate how they may behave in hundreds of dimensions. It turns out that our low dimensional intuition can be very misleading about data and distributions in high dimensional spaces. In this problem we will explore this in more detail.

Consider the Gaussian distribution with mean μ and identity covariance I_d in $\mathbb{R}^d$. Recall that the density assigned to any point $x \in R^d$, then becomes

$$p(x) = (2\pi)^{-d/2} \exp\left\{ - \|x - \mu\|^2/2\right\}.$$

(i) Show that when $x = \mu$, x gets assigned the highest density.

(This, of course, makes sense: the Gaussian density peaks at its mean and thus $x = \mu$ has the highest density.)

(ii) If mean has the highest density, it stands to reason that if we draw a large i.i.d. sample from the distribution, then a large fraction of the points should lie close to the mean. Let's try to verify this experimentally. For simplicity, let mean $\mu = 0$ (covariance is still I_d). Draw 10,000 points i.i.d. from a Gaussian $N(0, I_d)$.

To see how far away a sampled datapoint is from the mean, we can look at the distance $\|x - \mu\|^2 = \|x\|^2$ (that is, the squared length of the sampled datapoint, when mean is zero). Plot the histogram of squared length of the samples, for dimensions $d = 1, 2, 3, 5, 10, 50$ and 100. You should plot the all these histograms on the same figure for a better comparison.

What interesting observations do you see from this plot? Do you notice anything strange when the samples that were drawn from the high dimensional Gaussian distribution? Do most of the samples lie close to the mean?

(iii) Let's mathematically derive where we *expect* these samples to lie. That is, calculate

$$\mathbb{E}_{x\sim N(0,I_d)}\Big[\|x\|^2\Big].$$

Is the empirical plot in part (ii) in agreement with the mathematical expression you derived here?

(iv) This "strangeness" is not specific to Gaussian distribution, you can observe something similar even for the simplest of distributions in high dimensions. Consider the uniform distribution over the cube $[-1, 1]^d$. Just like in part (ii), draw 10,000 i.i.d. samples from this d-dimensional cube with uniform density, and plot the histogram of how far away from the origin the sample points lie. (do this for $d = 1, 2, 3, 5, 10, 50$ and 100, again on the same plot).

Recall that the cube has side length of 2, while most of the high-dimensional samples have length of far more than 2! This means even though you are drawing uniformly from the cube, most of your samples lie in the corners (and not the interior) of the cube!

(v) Again, calculate the expected (squared) length of the samples. That is, calculate

$$\mathbb{E}_{x\sim \text{unif}([-1,1]^d)}\Big[\|x\|^2\Big].$$

Does the plot in part (iv) in agreement with the expression you derive here?

4 **[An alternate learning paradigm]** In class you have seen that when building classifiers, one wants to minimize the expected classification error over a distribution $\mathcal{D}$. That is, we want to find the classifier f that minimizes:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}\Big[\mathbf{1}[f(x)\neq y]\Big]. \qquad (2)$$

Since this quantity is not estimable in practice (since we don't know $\mathcal{D}$ and only have access to finite samples drawn from it), it is usually approximated via its empirical equivalent:

$$\frac{1}{|S|}\sum_{(x,y)\in S}\mathbf{1}[f(x)\neq y]. \qquad (3)$$

This latter quantity is the training error if S is the training set, and is the testing error if S is the testing set.

However for certain applications, obtaining a positively and negatively labelled samples S is not possible. Consider, for example, the problem of modelling user preferences based on news-feed that gets shown. Very simply, if a user interacts with a particular news item (such as they clicked and read it) shows that they are interested in the contents of the article, thus providing a positive label. But if a user does not interact with a particular news item, it is not clear whether the user dislikes the contents of the article, or simply didn't get around to viewing it. In such a scenario obtaining a good quality negatively labelled data sample is not possible. We thus need a slightly different learning paradigm where the training samples obtained are only either labelled as positive examples, or they are simply unlabeled examples. We can model this as follows:

- $\mathcal{D}$ is an unknown distribution over $\mathbb{R}^D \times\{0,1\} \times \{0,1\}$. $(x, y, s) \sim \mathcal{D}$ is a sample, where x is the input feature vector, y is the true label, and s (the "selection" variable) is whether x was interacted with (ie, selected) or not. Note that only x and s are observed.

- $\Pr[s = 1 \mid x, y = 0] = 0$, that is, a negatively labelled x is never selected.
- Given y, s and x are conditionally independent. That is, which x gets selected (given that, say, x positively labelled) is chosen independently.

The goal of this problem is to find an empirical estimator of (2) similar to (3) but using the unlabeled and positive data only.

(i) Prove that $\Pr[y = 1 \mid x] = \frac{\Pr[s=1|x]}{\Pr[s=1|y=1]}$.

(ii) Using (i) prove that $\Pr[y = 1 \mid x, s = 0] = \frac{1-\Pr[s=1|y=1]}{\Pr[s=1|y=1]} \frac{\Pr[s=1|x]}{1-\Pr[s=1|x]}$.

For the rest of the problem, assume that both quantities on the RHS can be estimated from (x, s) data only. This is trivially true for $\Pr[s = 1 \mid x]$ (since it does not depend on y). And while estimating $\Pr[s = 1 \mid y = 1]$ with only(x, s) data is nontrivial, it can be done under suitable conditions.

(iii) Letting p denote the PDF of $\mathcal{D}$ show that:

$$\begin{aligned}\mathbb{E}_{(x,y)\sim\mathcal{D}}\big[\mathbf{1}[f(x) \neq y]\big] = \int_x & p(x, s = 1)\mathbf{1}[f(x) \neq 1] \\ & + p(x, s = 0)(\Pr[y = 1 \mid s = 0, x]\mathbf{1}[f(x) \neq 1] \\ & \quad + \Pr[y = 0 \mid s = 0, x]\mathbf{1}[f(x) \neq 0])dx\end{aligned}$$

(iv) Using parts (ii) and (iii) suggest an empirical estimator of (2) similar to (3) but that uses only (x, s) data.

Hint: Try viewing unlabeled points as part positive and part negative. That is, replace unlabeled points by two "partial" points. One that is positive with weight $w(x)$ and one negative with weight $1 - w(x)$.

5 **[Designing socially aware classifiers]** Traditional Machine Learning research focuses on simply improving the accuracy. However, the model with the highest accuracy may be discriminatory and thus may have undesirable social impact that unintentionally hurts minority groups[1]. To overcome such undesirable impacts, researchers have put lots of effort in the field called Computational Fairness in recent years.

Two central problems of Computational Fairness are: (1) what is an appropriate definition of fairness that works under different settings of interest? (2) How can we achieve the proposed definitions without sacrificing on prediction accuracy?

In this problem, we will focus on some of the ways we can address the first problem. There are two categories of fairness definitions: individual fairness[2] and group fairness[3]. Most works in the literature focus on the group fairness. Here we will study some of the most popular group fairness definitions and explore them empirically on a real-world dataset.

Generally, group fairness concerns with ensuring that group-level statistics are same across all groups. A group is usually formed with respect to a feature called the **sensitive attribute**.

[1] see e.g. **Machine Bias** by Angwin et al. for bias in recidivism predication, and **Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification** by Buolamwini and Gebru for bias in face recognition

[2] see e.g. **Fairness Through Awareness** by Dwork et al.

[3] see e.g. **Equality of Opportunity in Supervised Learning** by Hardt et al.

Most common sensitive features include: gender, race, age, religion, income-level, etc. Thus, group fairness ensures that statistics across the sensitive attribute (such as across, say, different age groups) remain the same.

For simplicity, we only consider the setting of binary classification with a single sensitive attribute. Unless stated otherwise, we also consider the sensitive attribute to be binary. (Note that the binary assumption is only for convenience and results can be extended to non-binary cases as well.)

Notations:

Denote $X \in \mathbb{R}^d,\ A \in \{0,1\}$ and $Y \in \{0,1\}$ to be three random variables: non-sensitive features of an instance, the instance's sensitive feature and the target label of the instance respectively, such that $(X, A, Y) \sim \mathcal{D}$. Denote a classifier $f : \mathbb{R}^d \to \{0,1\}$ and denote $\hat{Y} := f(X)$.

For simplicity, we also use the following abbreviations:

$$\mathbb{P} := \mathbb{P}_{(X,A,Y)\sim D} \qquad \text{and} \qquad \mathbb{P}_a := \mathbb{P}_{(X,a,Y)\sim D}$$

We will explore the following are three fairness definitions.

- *Demographic Parity (DP)*

$$\mathbb{P}_0[\hat{Y} = \hat{y}] = \mathbb{P}_1[\hat{Y} = \hat{y}] \qquad \forall \hat{y} \in \{0,1\}$$

(equal positive rate across the sensitive attribute)

- *Equalized Odds (EO)*

$$\mathbb{P}_0[\hat{Y} = \hat{y} \mid Y = y] = \mathbb{P}_1[\hat{Y} = \hat{y} \mid Y = y] \qquad \forall \hat{y},\ y \in \{0,1\}$$

(equal true positive- and true negative-rates across the sensitive attribute)

- *Predictive Parity (PP)*

$$\mathbb{P}_0[Y = y \mid \hat{Y} = \hat{y}] = \mathbb{P}_1[Y = y \mid \hat{Y} = \hat{y}] \qquad \forall \hat{y},\ y \in \{0,1\}$$

(equal positive predictive- and negative predictive-value across the sensitive attribute)

Part 0: The basics.

(i) Why is it not enough to just remove the sensitive attribute A from the dataset to achieve fairness as per the definitions above? Explain with a concrete example.

Part 1: Sometimes, people write the same fairness definition in different ways.

(ii) Show that the following two definitions for *Demographic Parity* is equivalent under our setting:

$$\mathbb{P}_0[\hat{Y} = 1] = \mathbb{P}_1[\hat{Y} = 1] \iff \mathbb{P}[\hat{Y} = 1] = \mathbb{P}_a[\hat{Y} = 1] \qquad \forall a \in \{0,1\}$$

(iii) Generalize the result of the above equivalence and state an analogous equivalence relationship of two equality when $A \in \mathbb{N}$, and $\hat{Y} \in \mathbb{R}$.

Part 2: In this part, we will explore the COMPAS dataset (available in `hw1data.zip`). The task is to predict two year recidivism. Download the COMPAS dataset from the class's website. In this dataset, the target label Y is `two_year_recid` and the sensitive feature A is `race`.

(iv) Develop the following classifiers: (1) MLE based classifier, (2) nearest neighbor classifier, and (3) naive-bayes classifier, for the given dataset.

For MLE classifier, you can model the class conditional densities by a Multivariate Gaussian distribution. For nearest neighbor classifier, you should consider different values of k and the distance metric (e.g. L_1, L_2, L_∞). For the naive-bayes classifier, you can model the conditional density for each feature value as count probabilities.

(you may use builtin functions for performing basic linear algebra and probability calculations but you should write the classifiers from scratch.)

You must submit your code to Courseworks to receive full credit.

(v) Which classifier (discussed in previous part) is better for this prediction task? You must justify your answer with appropriate performance graphs demonstrating the superiority of one classifier over the other. Example things to consider: how does the training sample size affects the classification performance.

(vi) To what degree the fairness definitions are satisfied for each of the classifiers you developed? Show your results with appropriate performance graphs.

For each fairness measure, which classifier is the most fair? How would you summarize the difference of these algorithms?

(vii) Choose any one of the three fairness definitions. Describe a real-world scenario where this definition is most reasonable and applicable. What are the potential disadvantage(s) of this fairness definition?

(You are free to reference online and published materials to understand the strengths and weaknesses of each of the fairness definitions. Make sure cite all your resources.)

(viii) [Optional problem, will not be graded] Can an algorithm simultaneously achieve high accuracy and be fair and unbiased on this dataset? Why or why not, and under what fairness definition(s)? Justify your reasoning.