CSE 512: Homework 3 — Due Wednesday, Oct. 7

1. **Conditional independence vs independence.**

   Tom is a blue-gray cat with a bushy tail, and Jerry is a brown mouse with a rope-like tail. After many years of fighting, they both decided to settle down, and now have thriving families. Tom has 10 kids and Jerry has 40 kids. Tom's kids are all cats like him, with bushy tails. Half of Tom's kids are blue, while the other half is gray. Jerry's kids are all brown mice, with rope-like tails.

   (a) I pick up a baby animal at random. What is the probability that ... (fill in the table)

fur \ tail	furry	rope-like
blue		
gray		
brown		

   (b) Are the features "fur color" and "tail texture" correlated, without knowing the type of animal? (Show mathematically.)

   (c) Now Tom comes over and says, "I'm very proud of my baby girl, of whom you are holding." What is the probability that (fill in the table)

fur \ tail	furry	rope-like
blue		
gray		
brown		

   (d) Are the features "fur color" and "tail texture" correlated, now that I know the animal is Tom's cherished baby daughter? (Show mathematically.)

2. **Exponential distribution.** Wait time is often modeled as an exponential distribution, e.g.

$$\mathbf{Pr}(\text{I wait} < x \text{ hours at the DMV}) = \begin{cases} 1 - e^{-\lambda x}, & x > 0 \\ 0, & x \leq 0, \end{cases}$$

   and this cumulative density function is parametrized by some constant $\lambda > 0$. A random variable X distributed according to this CDF is denoted as $X \sim \exp[\lambda]$.

   (a) In terms of λ, give the probability distribution function for the exponential distribution.

   (b) Show that if $X \sim \exp(\lambda)$, then the mean of X is $1/\lambda$ and the variance is $1/\lambda^2$.
   (You may use a symbolic integration tool such as Wolfram Alpha. If you do wish to do the integral by hand, my hint is to review integration by parts.)

   (c) Now suppose I run a huge server farm, and I am monitoring the server's ability to respond to web requests. I have m observations of delay times, $x_1, ..., x_m$, which I assume are i.i.d., distributed according to $\exp[\lambda]$ for some λ. Given these m observations, what is the maximum likelihood estimate $\hat{\lambda}$ of λ?

(d) Given the estimate of $\hat{\lambda}$ in your previous question, is $1/\hat{\lambda}$ an unbiased estimate of the mean wait time? Is $1/\hat{\lambda}^2$ an unbiased estimate of the variance in wait time?

(e) Now let's consider $x_1, ..., x_m$ drawn i.i.d. from a *truncated* exponential distribution, e.g.

$$p_{\lambda,c}(x) = \begin{cases} 0 & \text{if } x > c \text{ or } x < 0 \\ \dfrac{\lambda \exp(-\lambda x)}{1 - \exp(-\lambda c)} & \text{else.} \end{cases}$$

Using Hoefdding's inequality, give a range of values that account for the uncertainty in your guess. That is, as a function of x_i, m and δ, give a range of values $[\hat{\lambda}_{\min}, \hat{\lambda}_{\max}]$ such that

$$\mathbf{Pr}(\hat{\lambda}_{\min} \leq \mathbb{E}[X] \leq \hat{\lambda}_{\max}) \geq 1 - \delta.$$

3. **Decision theory.** I run a factory that makes widgets and gadgets. Despite best efforts, manufacturing defects can always occur. I would like to inspect each of these items individually, but the cost of inspection is pretty high, so I cannot inspect each individual widget and gadget.

The widgets and gadgets are printed on disks. A disk has a 10% chance of being warped. There are two printing presses, a blue one and a red one. The table below gives the possibility that, given a disk of a particular state printed by a particular press, a widget or gadget printed on that disk is defective.

disk \ press	red	blue
warped	30%	85%
normal	5%	0 %

(To interpret the table, the probability that a gadget is defective if it were on a disk that is not warped, and printed by a red press, is 5%.)

(a) First, we consider only the loss of quality in a product. That is, if we ship a widget or gadget is defective, we incur a loss of +1. Otherwise, we incur no losses.

i. Without inspecting anything (that is, we ship out everything we make), what is the Bayes risk of using a red press? a blue press? Which machine would I use to minimize the Bayes' risk?

ii. Without inspecting anything, what is the Minimax risk of using a red press? a blue press?

iii. Suppose I invest the effort into inspecting disks, and remove all warped disks. What is the Bayes risk of using a red press? a blue press?

iv. After removing all warped disks, what is the Minimax risk of using a red press? a blue press?

(b) Widgets are primarily used in online advertising. If they are defective, they will end up sending an ad that is undesirable. However, if they are removed, then no ad is sent out. Therefore, the revenue gained from a widget is estimated at

$$\text{revenue per widget} = \begin{cases} \$1 & \text{if widget is sold and is not defective} \\ -\$1 & \text{if widget is sold and is defective} \\ 0 & \text{if the widget is not sold.} \end{cases}$$

There is no cost to rejecting a disk, but the cost of inspection is \$1 per percent of disks inspected. (In other words, if I inspect all the disks, that cost me \$100.) Every widget that is not on a disk that was found to be warped is sold.

i. Compute the Bayes reward (e.g. the expected profit per day) as a function of $x = \mathbf{Pr}(\text{inspection})$ for widgets, when using the blue press. Compute the same for the red press.

ii. If you were a consultant for my factory, how much inspection would you recommend? Would you recommend using one press over the other for widgets?

(c) Gadgets are primarily used in medical care. If they are defective, someone will die. However, if they are removed, then someone waits a day longer to get a much-needed test. While we can never assign monetary value to a human life, in terms of insurance costs experts have estimated the following value:

$$\text{revenue per gadget} = \begin{cases} \$500 & \text{if gadget is sold and is not defective} \\ -\$10,000 & \text{if gadget is sold and is defective} \\ \$0 & \text{if the gadget is not sold.} \end{cases}$$

Again, there is no cost to rejecting a disk, and again, the cost of inspection is \$1 per percent of disks inspected. Every gadget that is not on a disk that was found to be warped is sold.

i. Compute the Bayes reward (e.g. the expected profit per day) as a function of $x = \mathbf{Pr}(\text{inspection})$ for gadgets, when using the blue press. Compute the same for the red press.

ii. If you were a consultant for my factory, how much inspection would you recommend? Would you recommend using one press over the other for gadgets?

4. **K-nearest neighbors** We will now try to use the KNN classifier to classify MNIST digits. Bear in mind that the full MNIST datasets has 60000 training symbols, so we will need to use every trick in the book to avoid memory issues.

- Load your data. It should have 4 parts: `X_train`, `X_test`, `y_train`, `y_test`. The labels `y_train` and `y_test` should have labels 1,2,...,10. *Do not prune away* data of different labels; in this exercise, we will use *all* the labels. We will also *not need to normalize any data.*
- However, to make life easier, you will need to typecast your data. The data loaded are all 8-bit integers, and can only take values 0,...,255. (Try adding 255 + 255; you won't get 510.) You need to convert them from type uint8 to type float. In MATLAB, you can typecast by just typing

```
X_train = float(X_train);
X_test = float(X_test);
```

  In Python, you can use

```
X_train = X_train.astype(float)
X_test = X_test.astype(float)
```

- First, write a function that takes in `X_train` and `y_train`, and a number $m < 60000$, and returns a train data matrix and label vector that contains only m data samples, sampled uniformly. Implement a 1-nearest-neighbor classifier, which takes a test data point, finds the closest point (in terms of Euclidean distance) in the train data set, and returns the label of that closest point. There are two ways to do this:
  - **Sacrifice memory, save computation.** Create a *distance matrix* D where D_{ij} stores the Euclidean distance between the ith training sample and jth test sample. Note that this requires all the test data samples to be known ahead of time.
  - **Sacrifice computational time, save memory.** The other approach is to compute everything "on the fly". For each incoming new data sample, compute its distance with every training sample, sort the distances, and return the label of the closest point. This is the more "realistic" approach, but for our purposes will require too much runtime, so we will forgo it.

  Using the first approach, and for $m = 10, 100, 1000, 10000$, return the error rate over the test data set.
- **$K > 1$.** Now increase K slowly from 1 to 10, and pick the label based on a majority voting system amongst the K closest neighbors. Do you see the computational time or memory becoming more burdensome with greater K? Again, for $m = 10, 100, 1000, 10000$, return the error rate over the test data set. Was the extra pain worth it?
- **Analyzing results.** For 10 test samples (one per digit) use `imshow` to plot, side by side, the digit, the training sample furthest from that digit but with the same label, and the training sample closest to that digit but with a different label. In the title, print the Euclidean distance between the test sample and the selected train sample. Interpret what you see.
- In your opinion, is KNN a reasonable way of performing handwriting digit classification? How does it compare against logistic regression or SVM?

Challenge!

In lecture we saw that if R_{NN} is the Bayes risk of a 1-NN classifier and R^* the Bayes risk of a Bayes classifier, then the Bayes risk can be used to bound the 1-NN classifier in that $R^* \leq R_{NN} \leq 2R^*(1 - R^*)$. Here we will investigate this bound more carefully, to make sure we really understand all the components of the proof.

We will analyze this bound in terms of a 2-cluster model, defined as follows:

- Y is a random variable taking values in $\{+1, -1\}$, and $\mathbf{Pr}(Y = 1) = p$.
- $X \in \mathbb{R}$ is a random variable taking any value in $\mathbb{R}$, defined by a scalar Gaussian distribution $X \sim \mathcal{N}(Y \cdot \mu, \sigma)$. That is, if $Y = 1$ then X has mean μ and variance σ^2; if $Y = -1$, then X has mean $-\mu$ and variance σ^2.

The goal will be to perform binary classification on X, and analyze how the performance of 1-NN and Bayes classifier works as we increase / decrease μ and σ.

1. **Exploring the model.** Plot some histograms of X, by drawing m datapoints and labels $x_1, ..., x_m$ and $y_1, ..., y_m$ according to our model. Use a "large enough" value of m so that the histogram well represents the model at the limit $m \to +\infty$.

   (a) **Sweep label balance.** Do this for $\mu = 2$, $\sigma = 1$ and $p = 0.1, 0.25, 0.5$

   (b) **Sweep separation width.** Repeat for $\sigma = 1$, $p = 0.5$ and $\mu = 0.1, 2.0, 10.0$.

   (c) **Sweep cluster variance.** Repeat for $\mu = 2$, $p = 0.5$ and $\sigma = 0.1, 1.0, 3.0$.

2. **Bayes classifier.** Recall that we denote $\eta(x) = \mathbf{Pr}(Y = 1|x)$.

   (a) Write out this probability (that is, find an expression for η) in terms of μ, σ, p, and x.
   Hint: Use Bayes' rule. Additionally, you can use the property that, for two different PDFs $p_{g(X)}$ and $p_{h(X)}$, that
   $$\frac{\mathbf{Pr}(g(X) = g(x))}{\mathbf{Pr}(h(X) = h(x))} = \frac{p_{g(x)}}{p_{h(x)}}.$$
   Note that in general, $\mathbf{Pr}(g(X) = g(x)) \neq p_{g(x)}$ when X is a continuous random variable! [1]

   (b) For $\mu = 1, \sigma = 1, p = 0.25$, fill out to 2 significant digits the first three columns in this table

[1] While in general the PDF does not tell us about the probability of a continuous variable taking a specific value, we can arrive at this equivalence of their ratios through the use of Radon-Nikodym derivatives. Think of it as something similar to chain rule: for two functions F and G applied over a measurable set A, assuming F and G are absolutely continuous, then $F(A) = G(A) \cdot \frac{\partial F(A)}{\partial G(A)}$. Anyway, we do not need to go into measure theory in this class, just rest assured that this operation in the hint is allowed!

x	$\mathbf{Pr}(y=1\|x)$ $(\eta(x))$	$\mathbf{Pr}(y=-1\|x)$ $(1-\eta(x))$	Bayes risk $(\beta(x))$	1-NN risk
-2				
-1				
-0.5				
0				
0.5				
1				
2				

(c) Given a new vector x drawn from this distribution, describe the action of the Bayes classifier. What is the rule in choosing a label $\hat{y} = 1$ or $\hat{y} = -1$?

(d) Write out an expression for the Bayes risk (R^*) , in terms of μ, σ, and p. Your answer may involve an integral that you do not need to evaluate.

(e) Numerically estimate the Bayes risk (R^*) for $p = 0.25$, $\sigma = 1$, $\mu = 1$. You can do this by generating 1000 points according to this distribution, calculating the Bayes risk for each point, and reporting the average.

3. **1-NN classifier.** Now we consider the limiting case of a 1-NN classifier; that is, we consider a scenario where, for any test data point x, there exists a labeled training point z_x that is arbitrarily close to x. We assume that the training and test data are drawn i.i.d., so, conditioned on their respective labels, x and z_x are not correlated.

(a) In this regime, the probability of error should somehow be high in regions where the label could be 1 or -1 with equal probability, but pretty low when the label is more likely 1 or -1. Write an expression, in terms of p, μ, σ, and x, of the "limiting error", e.g. the error of 1-NN if there always exists a labeled point arbitrarily close to x.

(b) Fill out the last column in the table above, again for $\mu = 1, \sigma = 1, p = 0.25$.

(c) Write out an expression for the 1-NN risk (R_{NN}) , in terms of μ, σ, and p. Your answer may involve an integral that you do not need to evaluate.

(d) Numerically estimate the 1-NN risk (R_{NN}) for $p = 0.25$, $\sigma = 1$, $\mu = 1$. You can do this by generating 1000 points according to this distribution, calculating the Bayes risk for each point, and reporting the average.

4. Ok, now we have all the pieces of the puzzle, and all the code snippets needed to do a more involved analysis! In particular, we want to see under what regimes we would expect the 1-NN risk to approach its lower bound (Bayes risk) or upper bound ($2R^*(1 - R^*)$).

- Write a function that takes a value σ, p, and μ, returns a numerical estimate of the Bayes risk R^* and 1-NN risk R_{NN}. Pick m "large enough" (I find $m = 1000$ works fine, but for certain regimes even fewer points is sufficient.)
- Write a function that takes a value σ, p, and μ, generates $x_1, ..., x_m$ and $y_1, ..., y_m$ according to this 2-cluster model, and using the first half of the data as a train set and the second half as a test set, returns the test misclassification error for a 1-NN classifier.

- For $\sigma = 1.$, $p = 0.25$, sweep μ as `logspace(-2,2,10) * sigma`. Plot as a function of μ the quantities R_{NN}, R^*, the upper bound $2R^*(1 - R^*)$, and the misclassification rate of the implemented 1-NN. Comment on what you see.
- Pick either 2 other values of σ or 2 other values of p and repeat this experiment. How do these other parameters affect the bound and tightness? Can you venture a guess as to what kind of scenarios would hit the upper bound, and what would hit the lower bound?