Problem Set 3

Electronic submission to Gradescope due **10am Thursday 10/19**. You are strongly encouraged to submit a homework with a partner—that is, submit one homework with both of your names. If you submit with a partner, **you must use a different partner than in previous problem sets!**

*[You may discuss these problems with classmates. Your submission must be the original work of you and your partner, and you must understand everything that is written on your submission. We suggest that you write solutions using LaTeX—see the course website for a latex solution template.]*

1. Chernoff Bound applications. For each question, be sure to specify which Chernoff bound you use, and give an actual numerical result:

   (a) Suppose you are conducting a poll to forecast the outcome of an election. Assume you randomly sample 10,000 voters uniformly at random from the population of voters. What is the probability that the true percentage of the population who support candidate X differs from the results of your poll by more than an additive $\pm 2\%$? What about differing by more than $5\%$? What about more than $10\%$? [To clarify, if the true percentage who support the candidate is $28\%$, then the result of the poll is within an additive $\pm 2\%$ if it yields a result between $26\%$ and $30\%$.]

   (b) How many random voters must you sample if you want to guarantee that with probability at least 0.95 (over the randomness in the choice of who you poll), you end up with an estimate of the percent of people who will vote for candidate X that is accurate to within an additive $\pm 1\%$? How does this change if you know that the actual fraction who support candidate X is very small, say at most $5\%$?

2. For a random variable $X$ distributed according to a Poisson distribution of expectation $\lambda \geq 0$, for all integers $k \geq 0$, by definition $\Pr[X = k] = \frac{e^{-\lambda} \lambda^k}{k!}$. For independent random variables, $X, Y$, with $X$ distributed according to a Poisson distribution of expectation $\lambda_1$, and $Y$ distributed according to a Poisson distribution of expectation $\lambda_2$, show that $X + Y$ is distributed according to a Poisson distribution of expectation $\lambda_1 + \lambda_2$ by computing the moment generating function of the Poisson distribution, and then arguing that the moment generating function of $X + Y$ is equal to that of the Poisson distribution of expectation $\lambda_1 + \lambda_2$ (and hence, these two distributions must be equal).

3. Suppose a class has $n$ students. On a given day, assume that if the $i$th student is asked “What is the temperature today”, her response is drawn from a Gaussian (Normal) distribution whose mean is the actual temperature, and variance is $\sigma_i^2$ (and her guess is independent from those of the other students).

   (a) If $\sigma_i = 1$ for all $i$, how accurate will the average of the $n$ guesses be? [Feel free to use the fact that the sum of independent Gaussians is Gaussian: namely for $X_i$ distributed according to $N(\mu_i, \sigma_i^2)$, if the $X_i$ are independent, then $\sum_i X_i$ is distributed according to $N(\sum_i \mu_i, \sum_i \sigma_i^2)$.]
(b) If \( \sigma_i = 1 \) for all \( i \), how accurate can we expect the median of the guesses to be? Specifically, specify some function \( d(n) \), and prove a result of the following form: with probability at least \( 3/4 \), the median will be within distant \( d(n) \) of the true temperature. [Your expression for \( d(n) \) should be of the form \( d(n) = \Theta(1/n^c) \) for some constant \( c \). Feel free to ignore constant factors in the expression \( d(n) \), though for full credit, you must get the asymptotic expression for \( d(n) \) correct, namely you must get the best exponent that is possible for such a result.]

(c) Suppose \( \sigma_i^2 = 1 \) for \( i \leq n/2 \), and \( \sigma_i^2 = n \) for \( i > n/2 \). Are we better-off returning the mean of the guesses, or the median? Support your claim with a Chernoff bound or two. [Specifically, argue as in the previous parts about how accurate the mean and median is.]

(d) BONUS: Suppose you are given the list of \( \sigma_i^2 \)'s, and hence you know the variance of each of the \( n \) guesses. What is the “best” estimate of the temperature, where, for example “best” could be defined as minimizing the expected square of the error of the guess.

(e) DOUBLE BONUS: Suppose you are given the set of \( \sigma_i^2 \)'s, but are not told which student has each variance. What is the “best” estimate of the temperature? [This is more of a research problem/food for thought, and we will not grade it.]

4. (How to Avoid Fake Science) Most statistical tools and scientific methodologies were designed around the assumption that a scientist would first make a hypothesis, and then gather data, and then test the hypothesis. Now, much of science is conducted by repeatedly probing the same dataset, i.e. looking at the data, then making a hypothesis, then looking at it again, then making a new hypothesis, etc. In this problem, you will investigate how subtle variations in how this process is conducted result in significant variations in the statistical validity of the results. To make this setting closer to home, we will formulate this process from the perspective of a machine learning person who is trying to tweak their learning algorithm to get better results (think hyper-parameter tuning for deep neural nets), though the identical story could also be told in the ‘formulating scientific hypotheses’ language.

Suppose we are trying to build a binary classifier for some task. Specifically, assume there is a distribution of labeled examples (e.g. the distribution over all photos), and we want to build a binary classifier (e.g. identifying whether a photo contains a cat). Formally, there is a distribution \( X \) over labeled pairs \((x, y)\), where \( y \in \{0, 1\} \) is the label, and we hope to learn a classifier \( f \) such that \( \Pr_{(x,y) \sim X}[f(x) = y] \) is close to 1.

(a) Suppose we have a dataset of \( n \) labeled examples drawn from the distribution \( X \). Before we look at the data, we make a list of \( L \) different learning algorithms that we will try (e.g. 1. Linear Regression, 2. Linear Regression with L0 regularization, 3. Convolutional Neural Net, 4,...). Then, for each learning algorithm, we train it on the first half of our dataset, \((x_1, y_1), \ldots, (x_{n/2}, y_{n/2})\). This will yield \( f_1, \ldots, f_L \), namely one model/labeling function for each algorithm. Next, we evaluate each of these labeling functions on the remaining \( n/2 \) datapoint. Suppose we pick the function \( f_i \) from this set that had the best performance on the test set, and decide to deploy that model. How many different algorithms can be in our list before we should expect that the performance on the test set will deviate significantly from the actual performance. Namely, how large can \( L \) be while ensuring that the test error of the top-performing model, \( f_i \), will be close to \( \Pr_{(x,y) \sim X}[f_i(x) = y] \)? (Essentially, this question is asking how many algorithms you
can try, as a function of \( n \), before you should no longer trust your results.) [For this part, and the next two parts, you should prove one of the following two statements: Option 1) For some constant \( c \), provided \( L \leq 2^{cn} \) then with probability at least 0.9 over the randomness in the draws of the examples, the output classifier, \( f_i \), has the property that 
\[
\left| Pr_{(x,y) \leftarrow X} [f_i(x) = y] - TestPerformance(f_i) \right| < 0.1.
\]
Option 2) For some constant \( c \), there exists a distribution \( X \) and choice of the learning algorithms such that for \( L > n^c \), with probability at least 0.5, the classifier with the best test performance, \( f_i \), has the property that 
\[
\left| Pr_{(x,y) \leftarrow X} [f_i(x) = y] - TestPerformance(f_i) \right| > 0.1.
\]

(b) Unfortunately, most people don’t make this list of learning algorithms ahead of time. Instead, they decide on the second learning algorithm based on how well the first algorithm worked, and decide on the \( i \)th algorithm based on the performances of all the earlier ones. Suppose we are in the setting of the previous problem, but we choose the \( i \)th algorithm as a function of how well the first \( i-1 \) functions \( f_1, \ldots, f_{i-1} \) perform on the TRAINING set. Now that we have \( L \) models, \( f_1, \ldots, f_L \), suppose we test them all on the TEST set, and pick the \( f_i \) with the best test set error. How large can \( L \) be before the test error performance of that \( f_i \) might differ significantly from \( Pr_{(x,y) \leftarrow X} [f_i(x) = y] \)?

(c) Now, suppose we choose the \( i \)th learning algorithm as a function of the performance of the first \( i-1 \) functions, \( f_1, \ldots, f_{i-1} \), on the TEST set. Again, we pick the \( f_i \) with the best test set error. How large can \( L \) be before the test error performance of that \( f_i \) might differ significantly from \( Pr_{(x,y) \leftarrow X} [f_i(x) = y] \)?

(d) (Double Bonus) Think more about how one can avoid this sort of “overfitting” in these adaptive data analysis settings where one repeatedly probes the same dataset. This is a very recent and active area of research, and I’d be happy to chat more about it!!!.