The University of Texas at Austin
Department of Electrical and Computer Engineering

EE379K: Data Science Lab — Spring 2018

Lab Two

Caramanis/Dimakis Due: Monday, February 5, 3:00pm 2018.

Comments/Remarks: The data for this lab are contained in two files: `Lab2_Data.zip` and `Names.zip`.

Programming Questions

1. **Correlations**.

   - When given a data matrix, an easy way to tell if any two columns are correlated is to look at a scatter plot of each column against each other column. For a warm up, do this: Look at the data in `DF1` in `Lab2.zip`. Which columns are (pairwise) correlated? Figure out how to do this with Pandas, and also how to do this with Seaborn.
   - Compute the covariance matrix of the data. Write the explicit expression for what this is, and then use any command you like (e.g., np.cov) to compute the 4×4 matrix. Explain why the numbers that you get fit with the plots you got.
   - The above problem in reverse. Generate a zero-mean multivariate Gaussian random variable in 3 dimensions, $Z = (X_1, X_2, X_3)$ so that (X_1, X_2) and (X_1, X_3) are uncorrelated, but (X_2, X_3) are correlated. Specifically: choose a covariance matrix that has the above correlations structure, and write this down. Then find a way to generate samples from this Gaussian. Choose one of the non-zero covariance terms (C_{ij}, if C denotes your covariance matrix) and plot it vs the estimated covariance term, as the number of samples you use scales. The goal is to get a visual representation of how the empirical covariance converges to the true (or family) covariance.

2. **Outliers**. Consider the two-dimensional data in `DF2` in `Lab2.zip`. Look at a scatter plot of the data. It contains two points that look like potential outliers. Which one is "more" outlying? Propose a transformation of the data that makes it clear that the point at $(-1, 1)$ is more outlying than the point at $(5.5, 5)$, even though the latter point is"farther away" from the nearest points. Plot the data again after performing this transformation. Provide discussion as appropriate to justify your choice of transformation. *Hint: if* $\mathbf{y}$ *comes from a standard Gaussian in two dimensions (i.e., with covariance equal to the two by two identity matrix), and*

$$Q = \begin{pmatrix} 2 & \frac{1}{2} \\ \frac{1}{2} & 2 \end{pmatrix},$$

   what is the covariance matrix of the random variable $\mathbf{z} = Q\mathbf{y}$*? If you are given* $\mathbf{z}$*, how would you create a random Gaussian vector with covariance equal to the identity, using* $\mathbf{z}$*?*

3. **Even More Standard Error** (This is to be completed only after you've completed the last written exercise below). In one of the written exercises below, you derive an expression

for what is called the *Standard Error*: where β denotes the "truth," $\hat{\beta}$ denotes the value we compute using least squares linear regression, and Z and e are as in the exercise below, you find:

$$\hat{\beta} - \beta = Ze.$$

If we know the distribution of the noise (the distribution generating the noise vectors, e_i), then we know the distribution for the error, $(\hat{\beta} - \beta)$. This allows us to answer the question given in class: if we solve a regression and obtain value $\hat{\beta}$, how can we tell if it is statistically significant? The answer is: we compare the size of $\hat{\beta}$ to the spread introduced by the noise (i.e., the standard error), and we ask: what is the likelihood that the true $\beta = 0$, and what we observed was purely due to the noise.

If the noise is Gaussian (normal), i.e., $e_i \sim N(0, \sigma^2)$, and if the values of the x_i are normalized, then we expect error of the size $\sigma/\sqrt{n}$, as this is roughly the standard deviation of the expression for the error that you derive above. This means: if you have twice the data points, you should expect the error to be reduced by about 1.4 (the formula says that the standard deviation of the error would decrease by a factor of $1/\sqrt{2}$).

Compute this empirically, as follows: We will generate data for a regression problem, solve it, and see what the error is: Generate data as I did in the example from class: $x_i \sim N(0, 1)$, $e_i \sim N(0, 1)$. Generate y by $y_i = \beta_0 + x_i\beta + e_i$, where $\beta_0 = -3$ and $\beta = 0$. *Note that since $\beta = 0$, this means that y and x are unrelated! The question we are exploring here is as follows: when we solve a regression problem, we are not going to find $\hat{\beta} = 0$ – we will find that $\hat{\beta}$ takes some other values, hopefully close to zero. How do we know if the value of $\hat{\beta}$ we get is statistically meaningful?*

- By creating fresh data and each time computing $\hat{\beta}$ and recording $\hat{\beta} - \beta$, compute the *empirical standard deviation* of the error for $n = 150$ (the number we used in class). In class, in the exercise where I tried to find a linear regression of y vs. noise, we found $\hat{\beta} = -0.15$. Given your empirical computation of the standard deviation of the error, how significant is the value -0.15?
- Now repeat the above experiment for different values of n. Plot these values, and on the same plot, plot $1/\sqrt{n}$. How is the fit?

4. **Names and Frequencies**. The goal of this exercise is for you to get more experience with Pandas, and to get a chance to explore a cool data set. Download the file `Names.zip` from Canvas. This contains the frequency of all names that appeared more than 5 times on a social security application from 1880 through 2015.

   - Write a program that on input k and $XXXX$, returns the top k names from year $XXXX$.
   - Write a program that on input `Name` returns the frequency for men and women of the name `Name`.
   - It could be that names are more diverse now than they were in 1880, so that a name may be relatively the most popular, though its frequency may have been decreasing over the years. Modify the above to return the relative frequency.
   - Find all the names that used to be more popular for one gender, but then became more popular for another gender.
   - (Optional) Find something cool about this data set.

5. **Visualization Tools and Missing/Hidden Values**. Visualization is important both for exploring the data, as well as for explaining what you have done. There are a huge number of such tools now available. This exercise walks through various functionalities of matplotlib and pandas.

   - The first part of this exercise was created by Dataquest. Run through the commands given in this tutorial: https://www.dataquest.io/blog/matplotlib-tutorial/ and understand the code.
   - Suppose that you would now like to plot some of the results by state. As you will see, the state information is sometimes missing, and other times it comes in varying forms. Figure out how to aggregate the results by state. The challenge here: how many of the tweets can you (correctly) assign to a state? Note: depending on how well you want to do (i.e., how many tweets you want to correctly assign to their state), this is not an easy problem!

6. **More Visualization Tools – Optional**. This exercise was also created by Dataquest. Run through the exercise `https://www.dataquest.io/blog/python-data-visualization-libraries/` for more visualization tools, including some that allow you to plot points on a map, and also to create interactive maps (zoom in, etc.).

Written Questions

1. **Standard Error:** It is important to develop an intuition for how much error we should "expect" when we solve a particular statistical problem. As the number of sample increase, we should expect the error to decrease. But by how much? In the first lab, you generated samples from a univariate (Problem 3) and multivariate (Problem 4) Gaussian with given parameters, and then you were asked to estimate those parameters from the data you generated. In this exercise, we derive explicitly the relationship that you (should have) observed doing those exercises.

   Suppose $Z \sim N(\mu, \sigma^2)$, i.e., Z is a univariate Gaussian (a.k.a. *normal*) random variable with mean μ and variance σ^2. Suppose that you see n samples from Z, i.e., you see data $z_1, \ldots, z_n$. Let $z_{\text{avg}} = \sum_{i=1}^{n} z_i/n$ denote the sample mean. We want to answer: how close is z_{avg} to μ? Note that z_{avg} *is a random variable* so we need to quantify in a probabilistic way how close z_{avg} is to μ.

   - Suppose $Z \sim N(0, 1)$. This is also called a *standard normal random variable*. For $n = 10,000$, compute the probability that $z_{\text{avg}} > 0.1$, $z_{\text{avg}} > 0.01$, and $z_{\text{avg}} > 0.001$.
   - Now for the general case: suppose $Z \sim N(\mu, \sigma^2)$, and for general n, compute the probability that $z_{\text{avg}} - \mu > n^{-1/3}$, $z_{\text{avg}} - \mu > n^{-1/2}$, and $z_{\text{avg}} - \mu > n^{-2/3}$. For your calculations, you can let n scale if that makes things easier.

2. **More Standard Error** Consider a one dimensional regression problem, where the offset is zero. Thus, we are trying to fit a function of the form $h(x) = x \cdot \beta$. Suppose that the truth is a noisy version of this – that is, the true model according to which data are generated is:

   $$y_i = x_i \cdot \beta + e_i.$$

   Everything in the above equation is a scalar, i.e., $y_i, x_i, \beta, e_i \in \mathbb{R}$. Here, e_i represents independent noise that is not modeled by the linear relationship.

- When we have n data points, the least squares objective reads:

$$\min_{\beta} : \quad \frac{1}{n} \sum_{i=1}^{n} (x_i \beta - y_i)^2.$$

  Show that this is a quadratic function in β, that is, if we expand it, it has the form

$$A\beta^2 + B\beta + C.$$

- Compute A, B, and C explicity, i.e., as explicit functions of the data, $\{x_i, y_i\}$. Note that these should not be functions of β. Show that $A \geq 0$ regardless of the values of the data.
- Since $A \geq 0$, this is a quadratic function whose graph opens up. This means that it is convex, and therefore the solution is characterized as the solution obtained by setting the first derivative (w.r.t. β) equal to zero. Do this, and therefore explicitly solve for the solution $\hat{\beta}$. This is the one-dimensional form of what is known as the *normal equations*. *Hint: we did this problem in class.*
- Now using the one dimensional expression from the second part, and plugging in the relationship $y_i = x_i \cdot \beta + e_i$, write

$$\hat{\beta} = \beta + Z\mathbf{e},$$

  where $\mathbf{e}$ denotes the vector of all the errors, e_i, added in each stage, and where Z is a matrix of appropriate dimension. What is Z, explicitly?

(Bonus) Repeat the last two questions in the general case. That is, derive the normal equations and the standard error for the general (vector) case, where our model is

$$y_i = x_i^\top \beta + e_i,$$

where now $x_i, \beta \in \mathbb{R}^p$, and $x_i^\top \beta$ denotes the dot product.