Fall 2023
CSCE 320 - Principles of Data Science
Homework #3

Problem 1 – Lasso regression (35%)

Using the dataset 'auto_mpg.csv' included in the assignment, perform lasso regression to predict car mileage as a function of the remaining variables in the dataset (excluding car name).

- **a. 15% credit.** Generate a plot of the MSE on validation data as a function of the regularization parameter via 10-fold cross-validation (see figure 6.12 in the ISL book).
- **b. 20% credit.** Generate a plot of the regression coefficients as a function of the regularization parameter (see figure 6.12 in the ISL book).

Problem 2 – Logistic regression (35%)

Using the dataset 'diabetes.csv' included in the assignment, develop a logistic regression model to predict whether a patient has developed diabetes as a function of the remaining variables in the dataset.

- **a. 15% credit.** Estimate the classification rate on validation data using k-fold cross validation. Generate a plot of the average classification rate across folds for different values of k (2,3,..10).
- **b. 20% credit.** Interpret the regression coefficients of the model. These coefficients will vary somewhat from fold to fold, but they should be somewhat consistent. This will require you to develop some familiarity with the individual predictors and how they are related to diabetes.

Problem 3 – k-means clustering (30%)

Using the image 'soccer.jpg' included in the assignment, perform k-means clustering to vector-quantize pixels according to their RGB color.

a. **15% credit.** Reconstruct the image using the colors in the codebook for values of k = 1,2,...,0. Generate a color JPEG image for each of the reconstructed images, and display all the images in a single 3x3 mosaic figure. Discuss which codewords emerge from the image as the codebook length increases?

b. **15% credit.** Generate a plot that shows the sum-squared-error (SSE) between the reconstructed image and the original image as a function of k, the number of clusters. To do so, repeat part (a) several times and, for each k, record the clustering that gives the minimum SSE. Can you make sense of this plot?

Common to all problems and sections: If you want to get a full grade, discuss your results in detail. We assume that everyone in class is capable of developing code and figuring out how to use ML libraries. It is when you design experiments and interpret the results that you show that you have learned. This skill is much harder than writing code, but it is the skill that will separate you from wannabe data scientists.

Submission guidelines

File naming convention

Please submit a separate ZIP file for each problem:

1. Problem 1: LastnameFirstname_p1.zip (e.g., GutierrezRicardo_p1.zip)
2. Problem 2: LastnameFirstname _p2.zip
3. …

Each ZIP file should include:

1. A PDF report (LastnameFirstname _p1.pdf)
2. A Jupyter notebook (LastnameFirstname _p1.ipynb)
3. Any data files your code needs in order to run (e.g., CSV files). No naming convention needed as long as it is consistent with your code.

PDF reports:

- Each PDF report will be graded in isolation from the associated code. Thus, please make sure the PDF includes all the necessary figures.
- Please use language that is appropriate for a technical report. Informal language (e.g., something you would write in an email to a friend) is not appropriate. Neither is the use of hyperbole. Technical terms should also be used rigorously (e.g., the term 'significant' is only appropriate when supported by an appropriate test statistic).
- Stuff you can do to lower your grade:
  - Use unwarranted[1] precision on numerical results.
  - Use lots of text fonts, font sizes, colors, and ***lots*** of Emphasis Everywhere.
    - Change the paragraph alignment.
  - Refuse to use a spelchercker. (because that's for wimps)
  - Ignore the file naming conventions above (this one is a sure winner)
  - Include snippets of your code, especially with dark backgrounds (it's Halloween!)
  - Include screenshots of numerical results, especially when you can see the pixelation. For example, see Figure 1.
- Things you can do to earn 5% extra credit
  - Not do stuff that would lower your grade (see above)
  - Start each section of a problem on a new page[2].
  - Include a caption for each figure, describing its contents.
  - Discuss each figure and cross-reference it, just like we are doing with Figure 1.
  - Use **clearly identifiable** section headings (if unsure, refer to the syllabus)
  - Each figure has its axes labeled and includes a legend whenever possible
  - The legend is easily interpretable (e.g., {veggies, meat, …} instead of {class 1, class 2, …}
  - Use colors that make the figures easy to interpret
  - Each figure and its text are properly sized (i.e., the opposite of Figure 1.)

---

[1] Reporting a classification rate of 45.343543875514441% when you have 100 examples in the dataset is incorrect; the best you can do in that case is 1%.

[2] If you are concerned about saving trees, remember we're not printing the PDFs.

Figure 1. Look at my p!

Code:

- Please make sure your code runs before submitting it. This includes making sure the ZIP file has all the required data files.
- You will be allowed to make minor modifications (e.g., TypeError, NameError) to your code if it does not run initially. However, any modification to your submitted code will result in a 50% reduction on the code grade.
- Please add comments to your code to assist us in reviewing it.

You are free to use any code or notebooks that are available online, as long as you:

- Acknowledge the source,
- Provide a link to it (e.g., URL),
- Describe what the code does, and
- Describe how you modified it to serve your purpose

If you use existing code/notebooks without proper citations, your submission will be treated as a case of **plagiarism** (see Syllabus).