EE5907/EE5027 Programming Assignment

20% of Final Grade for EE5907; 40% of Final Grade for EE5027

Project Deadline: 11.59pm, Monday, 28 Sep, 2020

Submit the following electronic files in one zipped folder onto the "CA1 Submission" folder on LuminNUS. The zip filename **MUST** be "[name_on_matric_card]_[matric_number]_CA1.zip"

1. **The pdf file of a well-written, concise project report. The report should NOT be longer than 10 pages (font 12, single space, arial). The report filename MUST be "[name_on_matric_card]_[matric_number]_report.pdf".**
2. **Your source code folder.**
3. **Readme file containing instructions to run your code. This readme file MUST be inside your source code**
4. **Please don't include project data in your zip folder.**

Before you start, take note of the following:

1. You may discuss the assignment with your classmates, but must write the code completely on your own. Plagiarism will be severely punished.
2. You can use matlab or python.
3. The data (**spamData.mat**) can be downloaded from the LuminNUS workbin. The data is in matlab format, which can be quite easily read inside python with the right package. If you can't figure out the right package, you probably should be using matlab :)
4. For all the questions, there are publicly available software libraries that implement some versions of these classifiers. However, implementing the algorithms yourself will help you understand the theory better. Therefore you are expected to implement the classifiers yourself.
5. The evaluation criteria include organization and clarity of the report, correctness of the implementation, and performance of the classifiers.
6. Please be considerate to your GA. Be as clear as possible in your submission, e.g., having a clear readme and provide helpful comments in your code. You are more likely to get a better grade if your GA can understand your code and report!

Acknowledgement: This assignment is a variation of a problem from Kevin Murphy's "Machine Learning: A Probabilistic Perspective" textbook.

5. Number of Instances: 4601 (1813 Spam = 39.4%)
6. Number of Attributes: 58 (57 continuous, 1 nominal class label)
9. Class Distribution:
Spam 1813 (39.4%)
Non-Spam 2788 (60.6%)

Data Description

The data is an email spam dataset, consisting of 4601 email messages with 57 features. Feature descriptions are found **in this link**. We have divided the data into a training set (3065 emails) and test set (1536 emails) with accompanying labels (1 = spam , 0 = not spam).

Data Processing

One can try different preprocessing of the features. Consider the following separately:

(a) log-transform: transform each feature using $\log(x_{ij} + 0.1)$ (assume natural log)

(b) binarization: binarize features: $\mathbb{I}(x_{ij} > 0)$. In other words, if a feature is greater than 0, it's simply set to 1. If it's less than or equal to 0, it's set to 0.

Q1. Beta-binomial Naive Bayes (24%)

Fit a Beta-Binomial naive Bayes classifier on the binarized data from the Data Processing section. Since there are a lot of spam and non-spam emails, you do not need to assume any prior on the class label. In other words, the class label prior λ can be estimated using ML and you can use λ^{ML} as a plug-in estimator for testing.

On the other hand, you should assume a prior Beta(α, α) on the feature distribution (note that the two hyperparameters for the Beta prior are set to be the same). For each value of $\alpha = \{0, 0.5, 1, 1.5, 2, \cdots, 100\}$, fit the classifier on the training data and compute its error rate (i.e., **percentage of emails classified wrongly**) on the test data. For the features (i.e., when computing $p(x|y)$), please use Bayesian (i.e., posterior predictive) training and testing (see **week 3** lecture notes on "Predicting Target Class of Test Data $\tilde{x}$ Using Posterior Predictive Distribution").

Make sure you include at least the following in your report:

- Plots of training and test error rates versus α
- What do you observe about the training and test errors as α change?
- Training and testing error rates for $\alpha = 1$, 10 and 100.

Q2. Gaussian Naive Bayes (24%)

Fit a Gaussian naive Bayes classifier on the log-transformed data from the Data Processing section. Since there are a lot of spam and non-spam emails, you do not need to assume any prior on the class label. In other words, the class label prior λ can be estimated using ML and you can use λ^{ML} as a plug-in estimator for testing.

For this exercise, just use maximum likelihood to estimate the class conditional mean and variance of each feature and use ML estimates as a plug-in estimator for testing (see **week 3** lecture notes on "ML estimation of μ, σ^2" and "Predicting Target Class of Test Data $\tilde{x}$" for Strategies 1 and 2) . Make sure you include the following in your report:

- Training and testing error rates for the log-transformed data.

Q3. Logistic regression (24%)

For the log-transformed data, fit a logistic regression model with l_2 regularization (see **week 4** lecture notes on "Newton's Method for Logistic Regression" and "Exclude Bias from l_2 Regularization"). For each regularization parameter value $\lambda = \{1, 2, \cdots, 9, 10, 15, 20, \cdots, 95, 100\}$ (note the jump in interval from 10 to 15 and beyond), fit the logistic regression model on the training data and compute its error rate (i.e., **percentage of emails classified wrongly**) on the test data. Make sure you include at least the following in your report:

- Plots of training and test error rates versus λ
- What do you observe about the training and test errors as λ change?
- Training and testing error rates for $\lambda = 1$, 10 and 100.

Don't forget to include the bias term in the logistic regression and your l_2 regularization should not apply to the bias term.

Q4. K-Nearest Neighbors (24%)

For the log-transformed data, implement a KNN classifier (see **week 5** lecture notes on "Non-parametric Classification"). Use the Euclidean distance to measure distance between neighbors.

For each value of $K = \{1, 2, \cdots, 9, 10, 15, 20, \cdots, 95, 100\}$ (note the jump in interval from 10 to 15 and beyond), compute the training and test error rates (i.e., **percentage of emails classified wrongly**). Make sure you include at least the following in your report:

- Plots of training and test error rates versus K
- What do you observe about the training and test errors as K change?
- Training and testing error rates for $K = 1$, 10 and 100.

Q5. Survey (4%)

Please give an estimate of how much time you spent on this assignment. Note that you will not be given a higher or lower grade if you spend a lot of time or very little time. We just want an honest estimate. Other feedback are welcomed. Please note that regardless of positive or negative feedback, your grades won't be affected of course. In other words, as long as your report does not leave this section blank, you will get the full 4%.