IITM-CS5011 : Introduction to Machine Learning — Given on: Sep 12, 10pm

Programming Assignment #1 — Due on : Sep 26, 5pm

- The goal of this assignment is to experiment with feature extraction methods, linear methods for classification and regression.
- This is an individual assignment. Collaborations and discussions with others are strictly prohibited.
- You may use Matlab, Octave or Python for your implementation. If you are using any other languages, please contact Harini before you proceed.
- You have to turn in the well documented code along with a detailed report of the results of the experiment electronically in Moodle. Typeset your report in Latex.
- Be precise for your explanations in the report. Unnecessary verbosity will be penalized.
- You have to check the Moodle discussion forum regularly for updates regarding the assignment.

Linear Classification

1. You will use a synthetic data set for the classification task. Generate two classes with 10 features each. Each class is given by a multivariate Gaussian distribution, with both classes sharing the same covariance matrix. Ensure that the covariance matrix is not spherical, i.e., that it is not a diagonal matrix, with all the diagonal entries being the same. Generate 1000 examples for each class. Choose the centroids for the classes close enough so that there is some overlap in the classes. Specify clearly the details of the parameters used for the data generation. Randomly pick 40% of each class (i.e., 400 data points per class) as a test set, and train the classifiers on the remaining 60% data When you report performance results, it should be on the left out 40%. Call this dataset at DS1.

2. For DS1, learn a linear classifier by using regression on indicator variable. Report the best fit accuracy, precision, recall and F-measure achieved by the classifier, along with the coefficients learnt.

3. For DS1, use k-NN to learn a classifier. Repeat the experiment for different values of k and report the performance for each value. Technically this is not a linear classifier, but I want you to appreciate how powerful linear classifiers can be. Do you do better than regression on indicator variables or worse? Are there particular values of k which perform better? Report the best fit accuracy, precision, recall and f-measure achieved by this classifier.

4. Now instead of having a single multivariate Gaussian distribution per class, each class is going to be generated by a mixture of 3 Gaussians. For each class, define 3 Gaussian, with first Gaussian of the first class sharing the covariance matrix with first Gaussian of the second class and so on. For both the classes, fix the mixture probability as (0.1,0.42,0.48) i.e. the sample has arisen from first gaussian with probability 0.1, second with probability 0.42 and so on. Now sample from this distribution and generate the dataset similar to question 1. Call this dataset as DS2. Now perform the experiments in questions 2 and 3 again, but now using DS2. What do you observe? Can you comment on the performance of both the classifier when you use DS1 and DS2?

Linear Regression

5. For the regression tasks, you will use the Communities and Crime Data Set from the UCI repository (http://archive.ics.uci.edu/ml/datasets/Communities+and+Crime). This is a real-life data set and as such would not have the nice properties that we expect. Your first job is to make this dataset usable, by filling in all the missing values . Use the sample mean of the missing attribute. Is this is a good choice ? What else might you use? If you have a better method, describe it, and you may use it for filling in the missing data. Turn in the complete data set.

6. Fit the above data using linear regression. Report the residual error of the best fit achieved on test data, averaged over 5 different 80-20 splits, along with the coefficients learnt.

7. Use Ridge-regression on the above data. Repeat the experiment for different values of λ. Report the residual error for each value, on test data, averaged over 5 different 80-20 splits, along with the coefficients learnt. Which value of λ gives the best fit? Is it possible to use the information you obtained during this experiment for feature selection? If so, what is the best fit you achieve with a reduced set of features?

Instructions on how to use 80-20 splits

1. Make 5 different 80-20 splits in the data and name them as $CandC-train\,\langle num\rangle\,.csv$ and $CandC-test\,\langle num\rangle\,.csv$.
2. For all 5 datasets that you have generated, learn a regression model using 80% and test it using 20%.
3. Report the average RSS over these 5 different runs.

Feature Extraction

8. You have been provided with a 3-dimensional dataset (DS3) which contains 2 classes. Perform PCA on the dataset and extract 1 feature and use the data in this projected

space to train linear regression with indicator random variables. Use the learnt model to classify the test instances. Report per-class precision, recall and f-measure. Also you have to report the 3-D plot of the dataset and the plot of the dataset in the projected space along with the classifier boundary.

9. Now use the same dataset and perform LDA on it and project the dataset to the derived feature space. Report per-class precision, recall and f-measure. Also you have to report the 3-D plot of the dataset and the plot of the dataset in the projected space along with the classifier boundary. What do you infer from these two experiments? Which feature extraction technique performs better for this scenario? Why?

Submission Instructions

Submit a single tarball/zip file containing the following files in the specified directory structure. Use the following naming convention: 'cs5011_a1_rollno.tar.gz'.

cs5011_a1_rollno

Dataset

DS1-train.csv
DS1-test.csv
DS2-train.csv
DS2-test.csv
CandC-train1.csv
CandC-test1.csv
CandC-train2.csv
CandC-test2.csv
CandC-train3.csv
CandC-test3.csv
CandC-train4.csv
CandC-test4.csv
CandC-train5.csv
CandC-test5.csv

Report

rollno-report.pdf

Code

all your code files