CSE543/ECE563: Machine Learning (PG)
Monsoon 2022

Assignment-4 (110 points)

Release Time: 7th Nov '22, 8:00 pm Submission Time: 28th Nov '22, 11:59 pm

Instructions

- This assignment should be attempted individually. All questions are compulsory.
- **Theory.pdf**: For conceptual questions, either a typed or hand-written *.pdf* file of solutions is acceptable.
- **Code Files**: For programming questions, the use of any one programming language throughout this assignment is acceptable. For python, either *.ipynb* or *.py* file is acceptable. For other programming languages, submit the files accordingly. Make sure the submission is self-complete & replicable i.e., you are able to reproduce your results with the submitted files only.
- **Regarding Coding Exercises**: You can use modules from sklearn or statsmodels or any similar library for writing the code. Use random seed wherever applicable to retain reproducability.
- **Report.pdf**: Create a *.pdf* report of programming questions that contains your applied approach, pre-processing, assumptions, analysis, visualizations, etc.. Anything not in the report will not be evaluated. Alternatively, a well-documented *.ipynb* file with answers to all the questions may be submitted as a part of both code file and report.
- **File Submission**: Submit a *.zip* named A4_RollNo.zip (e.g., *A4_PhD22100.zip*) file containing *Theory.pdf*, *Report.pdf*, and Code files.
- **Submission Policy**: Turn-in your submission as early as possible to avoid late submissions. Expect **No Extensions**. Besides, submission within 10 min of the passing of the deadline will incur 20% penalty in the total marks of this assignment. Beyond this, late submissions will not be evaluated and hence will be awarded zero marks.
- **Resource Constraints**: In any question, if their is a resource constraint in terms of computational capabilities at your end, you are allowed to sub-sample the data (must be stratified). Make sure to exclusively mention the same in the report with proper details about the platform that didn't work for you.
- **Clarifications**: Symbols have their usual meaning. Assume the missing information. You are free to use any libraries and need not do anything from scratch unless specifically stated otherwise. Use Google Classroom for any queries. In order to keep it fair for all, no email queries will be entertained. You may attend office/TA hours for personal resolutions. No queries will be answered in Google Classroom comments when 12 hours or less are left for the submission deadline.
- **Compliance**: The questions in this assignment are structured to meet the Course Outcomes CO2, CO3, and CO4, as described in the course directory.
- **Institute Plagiarism Policy Applicable.** Both programming and theoretical questions will be subjected to strict plagiarism check.
- There could be multiple ways to approach a question. Please explain your approach briefly in the report.

1. **Decision Tree Classifier** **(25 points)**

   (a) Dataset: https://archive.ics.uci.edu/ml/datasets/heart+disease
   Only 14 attributes are relevant. Refer the dataset link and use only these attributes. Perform EDA and preprocess the dataset. (5 points)

(b) Train a decision tree using both gini index and entropy. Don't change any of other default values of the classifier. In the following models, use the criteria which gives better accuracy on test set. (5 points)

(c) Use *export_graphviz* from sklearn and pydotplus libraries to visualize the decision tree which gives better accuracy in the above part. (5 points)

(d) Train decision trees with different values of the minimum number of samples required to split an internal node. Find the best value of this hyper-parameter by using testing and training accuracy. Plot the curve between training and testing accuracy vs 'minimum number of samples required to split' to support your analysis. (10 points)

2. **Random Forest Classifier** **(28 points)**

(a) Dataset:https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
Pre-process the dataset. (5 points)

(b) Fit a random forest model with default hyperparameters. Make predictions using this model and check the classification report and the accuracy. (5 points)

(c) Tuning max_depth : Using GridSearchCV tune for max_depth. Plot training and testing accuracies vs max_depth in a single plot. Analyse the plot and write your comment(s). (5 points)

(d) Tuning n_estimators: Repeat the above part and tune n_estimators this time. (5 points)

(e) Tuning more hyper-parameters: Using GridSearchCV tune the following hyper-parameters: 'max_depth', 'min_samples_leaf', 'min_samples_split', 'n_estimators', 'max_features' (5 points)

(f) Report the classification report and accuracy of the best model based on hyper-parameter tuning of the above model. (3 points)

3. **Naive Bayes Classifier** **(12 points)**

(a) Dataset : Iris dataset : https://archive.ics.uci.edu/ml/datasets/iris. Load the dataset. After performing pre-processing use a Gaussian Naive Bayes Classifier and report the accuracy of your model. (7 points)

(b) For the Naive Bayes algorithm, what happens if one of the classes has zero training samples? Explain why it will be helpful. (5 points)

4. **PCA and k-Means Clustering** **(28 points)**

(a) Dataset : https://archive.ics.uci.edu/ml/datasets/online+retail
Explore the dataset. (5 points)

(b) After performing pre-processing reduce the dimensionality of the dataset using PCA so as to retain at least 85 percent of the explained variance. (8 points)

(c) Determine the optimum number of clusters for the technique of k-Means clustering using elbow method and WCSS (Within Cluster Sum of Squares) as the metric. (5 points)

(d) Create k-Means model using the optimum number of clusters obtained above. Visualize the clusters. (5 points)

(e) Visualize the clusters with respect to first two PCA components. (5 points)

5. **Knowledge beyond classroom** **(17 points)**

- Give the formal (mathematical) definition of Convex set and function. Describe and comment on the convexity of Ridge and LASSO Regularization. (9 points)
- Which will be a more accurate nearest neighbour classifier, a KNN with $K = 2$ or $K = 3$. Give reason(s). (3 points)
- Averaging in Ensemble Learning: Consider the predicted values of a trained model on a particular dataset D as $\{Y_i\}_{i=1}^{N}$ with mean μ and variance σ^2. Assuming that we train K models on random subsets of the dataset $\{D_i\}_{i=1}^{K}$, such that $\bigcup_{i=1}^{K} D_i = D$ and the predictions made from each of the K models have the mean μ but a variance of $L\sigma^2$ where $L > 1$. To compute the final predictions, the predictions from the K models are averaged. Find the range of K required such that the performance of the model is probabilistically better than the model trained on the complete dataset. (5 points)