Machine Learning
Report of Assignment - 5
George Koshy - gxk140830

Question:
Below is the list of the new classifiers learned:

- k-Nearest Neighbors
- Bagging
- Random Forests
- AdaBoost
- Gradient Boosting.

In this assignment, you will test the performance of classifiers on any 5 UCI datasets of your choice. You are free to use the same ones as in assignment 3 or choose different ones.

Language used:
Python

Libraries Used:
Latest dev version of sklearn.

IDE used:
Jupyter

Analysis of running all the algorithms on the chosen datasets:

Dataset	Number of total instances	Number of Attributes	How many fold cross-validation	KNN Accuracy	Bagging	Random Forest	AdaBoost	Gradient Boosting	Best accuracy
Credit Card Approval	3000	23	10	0.761333	0.784667	0.812000	0.822333	0.823333	Ada/ Gradient
Phishing	11055	30	10	0.937557	0.945701	0.961991	0.930317	0.936652	Random Forest
Transfusion	748	4	10	0.918919	0.932432	0.905405	0.918919	0.918919	Bagging
Breast Cancer Diagnostic	569	31	10	0.785714	0.982143	0.964286	0.964286	0.964286	Bagging

Dataset	Number of total instances	Number of Attributes	How many fold cross-validation	KNN Accuracy	Bagging	Random Forest	AdaBoost	Gradient Boosting	Best accuracy
Ionosphere Data	351	34	10	0.971429	0.971429	0.971429	0.971429	0.971429	All Similar

Observations:

- As seen various datasets gives slightly varying accuracies.
- Some algorithms perform very well for some datasets and some perform even bad than assigning classes randomly(ideally 50%)
- Observe that the data sets are not large, data science requires that the data sets are at least a little big for better results, more the sample size the better it is. We see some of the chosen datasets have very less number of instances.
- **The most consistent algorithm observed across datasets was Gradient Boosting and AdaBoost and sometimes Bagging**. No matter what the dataset was, Gradient Boosting/ Boosting gave a pretty high accuracy(Not the best all the time) Advantage of boosted trees are about modeling, because boosted trees are derived by optimizing a objective function, basically it can be used to solve almost all objective you can write gradient out. So it performs significantly well for all data sets.
- **Random Forest perform well** because it takes the votes of multiple trees built into consideration.
- **The worst performing algorithms turned out to be KNN and Bagging (this is again at times and it again varies with the dataset).**
- The algorithms for this assignment except **KNN** are considered better than many others because of their depth. **KNN** is not suggested because of the ambiguities it has. We have to choose the value of **K,** there might be a local minima and not a global minima while calculating the centroids, etc.

Conclusion:

Ranking the algorithm(for our specific 5 datasets.) according to their usefulness if correlation between the attributes in the dataset is not known.

1. Gradient Boosting
2. AdaBoost
3. Random Forest
4. Bagging
5. KNN.