Yelp Dataset Challenge

Goal: Predicting the ratings given to a business based on the reviews and other useful credentials of the user and the business. The ratings are integer values from 1 to 5 and thus this is a classification task.

Steps:

Preprocessing and Feature Extraction

- Read published papers on the same and decided which columns to choose as a starting point
- Joined relevant columns using Python and Spark SQL to form a single dataset and extracted it in a CSV format
- Performed sentiment analysis(removal of stop words, digits, punctuation and non-English words) to extract and create more features and stored them in a SQL table for faster and better querying
- Graphed correlation matrix and scatterlplot to understand which features have similar impact on the result and chose the best ones from them by hit and trial method by trying them in sklearn GridSearchCV

Feature Selection and Model Evaluation

- Used FeatureSelection library of sklearn to compute the best features to use and the ones that provided the most information gain
- Evaluations were done using various models like Decision Tree, Neural Net, k-nearest neighbors, Random Forest, Bagging, AdaBoost, Logistic Regression, Gradient Naïve Bayes and Gradient Boosting
- Used GridSearchCV function of ModelSelection library of sklearn to exhaustively search over specified parameter values for each model

Model Analysis

- Computed accuracy and area under ROC curve as metrics to decide the best performing model
- Random Forest model performed the best, achieving an accuracy score of 0.71, precision: 0.70, recall: 0.71 and F-1 score of 0.72
- I also found out by reviewing the confusion matrix that most of the false positives and false negatives were in predicting rating = 3. Removing it from the dataset, accuracy jumped to 91% and Logistic Regression performed the best then. This made sense as LR performs very well in classifying binary results and removing the central value of 3, LR could easily distinguish between pairs of 1,2 and 4,5 which improved accuracy by a big margin.

Additional Points:

For Preprocessing:
Using JavaScript: Sorted businesses according to the states where they are situated (done using a Dictionary) and also according to the number of reviews they have received so as to teach the model to work differently for businesses with different number of reviews.

Algorithms Designed:
I wrote the code for the k-nearest neighbors algorithm used in Python and the rest of the models used were taken from sklearn library.

Extra features designed:

Using NLTK and Python: Noun, verb, adverb and adjective count and their respective positive, negative and neutral word counts in a review
Using Python: Number of different businesses rated by a user, number of days since user is a member of Yelp

Feature Selection Tests Used:

- F-classif: Computes the Analysis of Variance F-value for the provided sample
- Select_K_best: Computes the best k features
- Chi2: Compute chi-squared stats between each non-negative feature and class
- Recursive Feature Elimination (RFE)
- Mutual_info_classif: Estimate mutual information for a discrete target variable

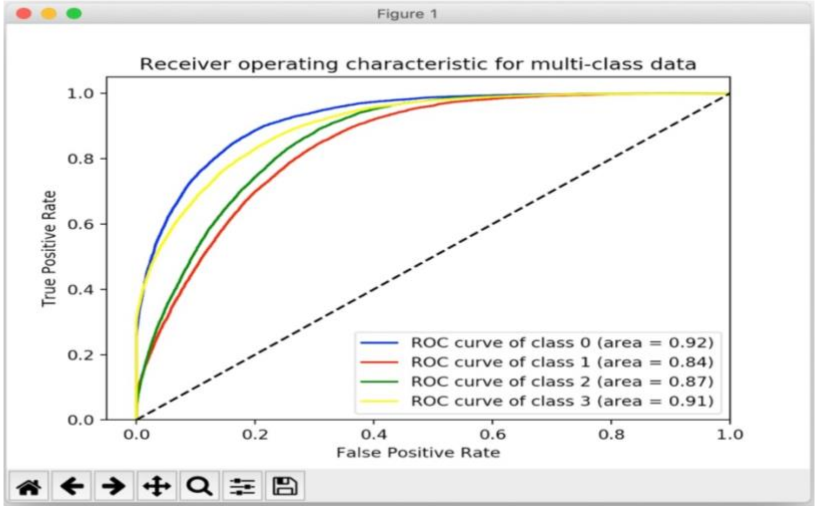
Final Features chosen:
[useful upvotes, cool upvotes, total tokens in a review, noun count, verb count, adjective count, positive words count, negative words count, neutral words count, user average stars, user yelping since, user review count]

Summarizing the Result:

Best Performing Classifier

Algorithm	Best Parameters	Average Precision	Average Recall	Average F-1 score	Accuracy
Random Forest	{'criterion': 'gini', 'max_depth': 90, 'min_samples_split': 30, 'n_estimators': 80}	0.73	0.71	0.71	0.7154375

Area under ROC Curve (AUC):
For 100% accuracy, area under the curve should be 1. This curve below shows the respective areas for different classes (1, 2, 4 and 5) of ratings predicted by the model.



Future Improvements:

- This same model can be modified and used to determine the downsides of a business when combined with the data regarding the business location, type and hours of operation
- This model can also be changed to work as a recommendation service to the users which provides them with a suggestion of their favorite dishes according to the cuisine, time of day and season