CAPSTONE PROJECT REPORT
YELP – RECOMMENDING TOP 10 FEATURES FOR BUSINESS SUCCESS

Team Members:
Arun Jaganathan - ajagana
Neela Niranjani Vengateshwaran - nvengat
Manjusha Trilochan Awasthi - mawasth

GitHub Link: https://github.com/arun5493/Yelp_Rating_Prediction

NOTE: We have covered the following three topics in our implementation :

- Model inter-comparison, diagnostics, design of data science experiments
- Generalized Linear Models (Linear Regression, Logistic Regression, Multinomial Regression)
- Deep Learning (Artificial Neural Network)

BI USE CASE:

Yelp has a rich base of user ratings for various businesses, users, user reviews, tips and checkin's. Using this dataset, we can apply various business intelligence techniques and answer questions like:

1) Perform NLP and guess the rating for a business based on the user review text
2) Use Convolution Neural Network to classify the photos uploaded by user for businesses.
3) Perform time series analysis and predict which businesses can go out of business based on their past performance and reviews it has received.
4) Recommend business features that new business owners need to focus in order to have a successful business.

In our project, we will be dealing with the below use cases :

- Recommending top 10 business attributes to new restaurant owners that lead to high star ratings. We focus on the problem from the perspective of new restaurant and try to identify the key business attributes that would influence the overall performance and hence the success of the restaurant. For our use case, we have considered the business overall rating as a measure of its overall performance.
- Artificial Neural Networks are known for its high accuracy in prediction. Thus, we have used an Artificial Neural Network and experimented its performance with different hidden layers.

DATA SET:

Yelp has a rich base of user ratings for various businesses. The yelp_academic_dataset_business.json dataset contains important attributes for around 61k business. We will be working on this dataset for our use case. To keep the problem at hand tractable, we will be focusing on restaurants based in Nevada, US.

The business meta-data structure is as follows:

```
{
    "business_id":"encrypted business id",
    "name":"business name",
    "neighborhood":"hood name",
    "address":"full address",
    "city":"city",
    "state":"state -- if applicable --",
    "postal code":"postal code",
    "latitude":latitude,
    "longitude":longitude,
    "stars":star rating, rounded to half-stars,
    "review_count":number of reviews,
    "is_open":0/1 (closed/open),
    "attributes":["an array of strings: each array element is an attribute"],
    "categories":["an array of strings of business categories"],
    "hours":["an array of strings of business hours"],
```

```
    "type": "business"
}
```

PERFORMANCE METRIC : We will use accuracy and RMSE to compare the performance of models.

DATA PRE-PROCESSING: Since our business use case focuses on Nevada-based restaurants, we have initially filtered out all non-restaurant & non-Nevada based businesses.

We figured out that business dataset has overall 88 features (including the sub-features like 'categories', 'attributes'). We are interested in identifying top 10 features that affect the '**star**' attribute.

The complete set of features is:

'business_id','address','city','state','postal_code','name','review_count','stars','type','BikeParking','BusinessAccepts Bitcoin','BusinessAcceptsCreditCards','WheelchairAccessible','Caters','GoodForKids','HasTV','OutdoorSeating','Rest aurantsDelivery','RestaurantsGoodForGroups','RestaurantsReservations','RestaurantsTakeOut','RestaurantsTable Service','BusinessParking_garage','BusinessParking_street','BusinessParking_validated','BusinessParking_lot','Busi nessParking_valet','RestaurantsPriceRange','Ambience_romantic','Ambience_intimate','Ambience_classy','Ambien ce_hipster','Ambience_divey','Ambience_touristy','Ambience_trendy','Ambience_upscale','Ambience_casual','Alco hol','WiFi','Music_DJ','Music_Background','Music_Karaoke','Music_Live','Music_Video','Music_Jukebox','Categories _Restaurant','Categories_Food','Categories_NightLife','Categories_Bars','Categories_AmericanTraditional','Catego ries_FastFood','Categories_Pizza','Categories_Sandwiches','Categories_Coffee&Tea','Categories_Italian','Categori es_Burgers','Categories_Mexican','Categories_AmericanNew','Categories_Chinese','Categories_Breakfast&Brunch' ,'Categories_SpecialityFoods','Categories_Cafes','Categories_Hotels','Categories_Desserts','Categories_Japanese',' Categories_IceCreams','Categories_ChickenWings','Categories_SeaFood','Categories_SushiBars','Categories_Spor tsBars','Categories_Beer','Categories_Wine','Categories_Delis','Categories_Asian','Categories_Salad','Categories_ Med','Categories_Barbeque','Categories_Indian','Categories_SteakHouses','Categories_Thai','Categories_Diners',' Categories_French','Categories_Greek','Categories_Vegetarian','Categories_Buffet','Categories_GlutenFree','Categ ories_Soup','Categories_Vegan'

Next, we removed the features which do not affect the star rating of the restaurants like 'business_id','address','city','state','postal_code','name' and 'type'.

We further observed that some of the features were very sparsely populated. For example, Categories_Soup data was available only for 514 restaurants. So, we considered only those features for which data was available in atleast 50% of the dataset.

In the feature set, we had many features that had a positive or negative response like Ambience_romantic=True/False. These were converted to numerical values of 0 or 1. Similarly, categorical variables like NoiseLevel=average/loud/quiet were converted to numerical values 0,1 and 2.

Next, we filtered out the features that were highly correlated. We used Pearson correlation to find similar attributes and removed the attributes having correlation greater than 70%.

After the filtering & preprocessing stage, the feature set was drilled down to following 28 features:
['review_count', 'stars', 'BikeParking', 'BusinessAcceptsBitcoin', 'BusinessAcceptsCreditCards', 'WheelchairAccessible', 'Caters', 'GoodForKids', 'HasTV', 'OutdoorSeating', 'RestaurantsDelivery', 'RestaurantsGoodForGroups', 'RestaurantsReservations', 'RestaurantsTakeOut', 'RestaurantsTableService', 'BusinessParkingTypes', 'RestaurantsPriceRange', 'Ambience_Offered', 'Alcohol', 'WiFi', 'Music_Offered', 'Categories_Offered', 'Categories_NightLife', 'Categories_Bars', 'Categories_AmericanTraditional', 'Categories_FastFood', 'Categories_Pizza', 'Categories_Sandwiches', 'Categories_Italian']

Also, before building models, we performed Normalization on the feature sets. This was needed since the feature set has a variety of data, some are binary true false while others are categorical.

If a feature has a variance that is orders of magnitude larger than others, then it might dominate the star rating and make the estimator unable to learn from other features correctly as expected. Hence we performed normalization to achieve better model performance.

FEATURE SELECTION:

Our next task was to identify the top 15 features that significantly affect the star rating and hence the success of the restaurant. We used different feature selection techniques, compared their results and identified the top 10 features that were commonly identified by all techniques.

- First, we used **`VarianceThreshold`** provided by `sklearn` to remove all the low-variance features. We observed that 9 features had low variance and after removing these features we were left with below features :
  ['review_count', 'stars', 'BikeParking', 'WheelchairAccessible', 'Caters', 'GoodForKids', 'HasTV', 'OutdoorSeating', 'RestaurantsReservations', 'RestaurantsTakeOut', 'RestaurantsTableService', 'BusinessParkingTypes', 'RestaurantsPriceRange', 'Ambience_Offered', 'Alcohol', 'WiFi', 'Music_Offered', 'Categories_Offered', 'Categories_Italian']

- Next, we applied **Univariate Feature Selection** which examines each feature individually and determines the strength of the relationship of the feature with the response variable. We used the chi2 scoring function since our features and classes are both discrete.
  We performed univariate feature selection using sklearn's SelectKBest function.
  The top 15 features predicted were :
  ['review_count', 'stars', 'BikeParking', 'WheelchairAccessible', 'GoodForKids', 'HasTV', 'OutdoorSeating', 'RestaurantsTakeOut', 'RestaurantsTableService', 'BusinessParkingTypes', 'RestaurantsPriceRange', 'Ambience_Offered', 'Alcohol', 'Music_Offered', 'Categories_Offered']

- Next, we performed **Recursive Feature Elimination** which selects features by recursively considering smaller and smaller sets of features.
  For this technique, we used the RFE function provided by sklearn.
  The top 15 features predicted by this technique were :
  ['stars', 'BikeParking', 'WheelchairAccessible', 'Caters', 'GoodForKids', 'HasTV', 'OutdoorSeating', 'RestaurantsReservations', 'RestaurantsTakeOut', 'RestaurantsTableService', 'RestaurantsPriceRange', 'Ambience_Offered', 'WiFi', 'Music_Offered', 'Categories_Offered']

- Bagged decision trees like Random Forest and **Extra Tree Classifiers** can be used to estimate the importance of features. We used ExtraTreesClassifier provided by sklearn package.
  # The top 15 features identified by this technique were :
  ['review_count', 'stars', 'BikeParking', 'WheelchairAccessible', 'Caters', 'GoodForKids', 'HasTV', 'OutdoorSeating', 'RestaurantsReservations', 'RestaurantsTakeOut', 'RestaurantsTableService', 'BusinessParkingTypes', 'RestaurantsPriceRange', 'Ambience_Offered', 'Alcohol', 'Music_Offered']

We observed that the following 10 features were common amongst all three results :
['HasTV', 'RestaurantsPriceRange', 'Music_Offered', 'OutdoorSeating', 'Ambience_Offered', 'BikeParking', 'RestaurantsTakeOut', 'GoodForKids', 'WheelchairAccessible', 'RestaurantsTableService']

Hence we concluded that these are top 10 features affecting the rating of a restaurant in Nevada.

BUILDING MODEL TO TEST SELECTED FEATURES:

The next task was to test the strength of features that were identified in the previous step.

We classified the data using models such as Naive Bayes, Logistic Regression, Ordinal Regression, Multinomial Regression, Logistic Regression and Support Vector Machines.

- **Naive Bayes Classifier** : We split the dataset to training and test set in the ratio of 80:20. Then we trained the data set using the top 10 features and predicted the model output for test dataset. The accuracy for this model was : 0.65

  Using 10-fold cross validation, the accuracy obtained was :
  Accuracy of the Naive Bayes Model using 10-Fold validation :

	Score
1	0.634860
2	0.620865
3	0.614504
4	0.673028
5	0.611959
6	0.669211
7	0.670483
8	0.620865
9	0.675159
10	0.653503

  Mean accuracy of the Naive Bayes Model : Score 0.644444

- **Ordinal Regression:** We split the dataset to training and test set in the ratio of 70:30. Then we trained the data set using the top 10 features and predicted the model output for test dataset. The accuracy for this model was : 0.578

- **Logistic Regression :** We split the dataset to training and test set in the ratio of 70:30. Then we trained the data set using the top 10 features and predicted the model output for test dataset.
  The accuracy for this model was : 1.00

  Running the binary logistic regression model for our dataset gives an accuracy of 100% but this is a case of overfitting due to the addition of many dummy variables (splitting the star rating column which consists of 5 labels - 1,2,3,4,5 into 5 different binary columns one each for the star rating). Hence, we use multinomial logistic regression which better explains the response variable for our dataset.

- **Multinomial Regression :** We split the dataset to training and test set in the ratio of 70:30. Then we trained the data set using the top 10 features and predicted the model output for test dataset. The accuracy for this model was : 0.589

- **Support Vector Machine :** We split the dataset to training and test set in the ratio of 70:30. Then we trained the data set using the top 10 features and predicted the model output for test dataset. The RMSE reported for this model was : 0.4833

- **Linear Regression :** We split the dataset to training and test set in the ratio of 70:30. Then we trained the data set using the top 10 features and predicted the model output for test dataset. The RMSE reported for this model was : 0.4551

HYPOTHESIS TESTING : We performed Wilcoxon signed-rank test & Paired t test on our models.
For Wilcoxon test, the p-value is obtained as 0.0019 which is less than 0.05. Hence, we reject null hypothesis which states that there is no significant difference between the medians of the error rates of the two classifiers - SVM and ORDINAL REGRESSION models.

For the paired t test, since the p value is of the order of 10^(-16) which is less than 0.05 (alpha value), we reject the null hypothesis which claims that the true difference in means of the two error percentages is equal to 0.

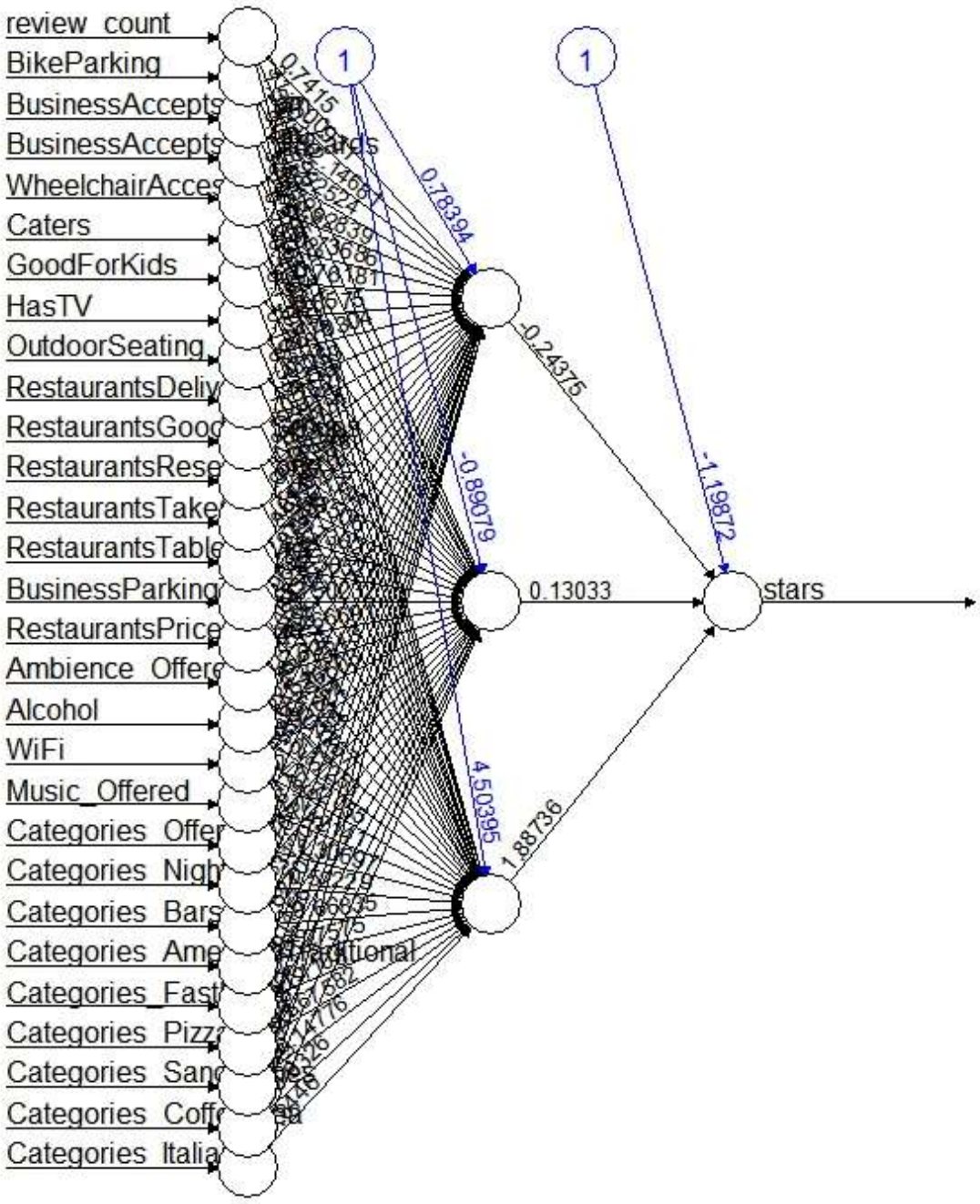
BUILDING ARTIFICIAL NEURAL NETWORK FOR STAR RATING PREDICTION:

Neural Networks (NN) is a data mining technique used for classification and clustering. It is modeled after the neural structure of the human brain. NN usually learns by examples. If NN is supplied with sufficient examples, it can perform classification and discover new patterns in data. Basic NN consists of three layers, input, output and hidden layer. Each layer can have multiple nodes. Nodes from input layer are connected to the nodes in the hidden layer. Nodes from hidden layer are connected to the nodes in the output layer. Those connections are assigned weights. One of the most popular NN algorithms is back propagation algorithm. After choosing the weights of the network randomly, the back propagation algorithm is used to compute the necessary corrections.
The algorithm can be decomposed in the following four steps:

i) Feed-forward computation
ii) Back Propogation to the outer layer
iii) Back propogation to the hidden layer
iv) Weight updates

Aritifical neural networks are trained using back propogation method and it can approximate almost any functions. All numeric attributes were taken as inputs for this. We used a trial and error method to find the number of hidden layers required to get the lowest RMSE and found that 3 hidden layers were needed to get the lowest RMSE value. The value obtained was 0.020. When the hidden layers were increased, the performance (in terms of time) of the neural network also reduced. So, this is the reason why we chose 3 as the Hidden Layer number.

The neural network obtained is as follows:



CONCLUSION: We used different techniques to identify the top 10 features that have the highest influence on the restaurant star rating. All three techniques I.e Univariate Feature Selection, Recursive Feature Selection and Extra Tree Classifiers produced 10 features in common : *RestaurantsTableService , HasTV , RestaurantsPriceRange , OutdoorSeating, Ambience_Offered, BikeParking, RestaurantsTakeOut, GoodForKids, WheelchairAccessible, Alcohol*'

We also built and compared various models for predicting the star rating using the top 10 features and found that Naive Bayes classifier gave the highest accuracy among classifications and Linear Regression gave the lowest RMSE among regression.

We studied Artificial Neural Network technique and experimented its performance with different hidden layers. We observed that for our dataset, model with 3 hidden layers gave the lowest error.