**Slide 1:**

**Overview**

* Business Understanding
* Data Understanding
* Data Preparation
* Exploratory Data Analysis
* Modeling
* Results
* Conclusion

**Slide 2:**

**Business Understanding**

* Problem Statement
  + The creation of an accurate model to predict restaurant ratings based on Yelp reviews
* Benefits to business owners:
  + Determine features that impact business rating.
  + Tailor specific features that will promote positive reviews.

**Slide 3:**

**Data Understanding**

* Data Source: Yelp – 9th Dataset Challenge
* Link: <https://www.kaggle.com/z5025122/yelp-csv#yelp_academic_dataset_business.csv>.
* Original Dataset: 42,153 rows, 11 metro areas

<table for initial variables>

* Used R, along with Tidy Data Principles:
  + *filter* from dplyr library
    - * State = AZ
  + *str\_detect* from stringr library
    - * Categories - contains “Restaurants”
* Revised Dataset: 7,439 rows in Phoenix Metro-area

<table for revised dataset – attributes only with frequency counts and missing data %>

**Slide 4a**

**Data Preparation**

* **Missing and special values**
  + - Converted blank values in all fields to NA
    - 104 business\_ids had “#NAME?” as a string. Since the # was immaterial, they were removed from the df
    - Left NA’s as is since not required with Decisioon Trees
    - Hold off on imputing, since decision trees allow for missing values

**Slide 4b**

1. **Data Type Conversions**
   1. Factors, logical, character, numeric, integer, etc. (see dc)
      1. Converted all attributes to factors
      2. Also, converted name, open, city stars, and food\_type to factors
      3. Dropped unused factor levels – ‘droplevels’(df$col) – drops all unused levels for that column (not the column itself)

**Slide 5**

* **String Manipulation**
  + Feature Names
    - Eliminated spa ces, replaced periods (‘.’) with commas (‘,’) and converted to all lowercase
  + Cleaned up City variable
    - Originally had 52 cities, containing duplicate and erroneous names
    - Used functions from stringr package (str\_detect, str\_trim, str\_replace), as well as case\_when
      * Cleaned up obvious errors (misspellings, duplicate names)
      * Consolidated different string combinations of city name (i.e. Phoenix, Pheonix, Central City, etc.)
      * Cities with low value counts (< 20) were assigned “Other”
      * Cities in Arizona but not in the Phoenix metro-area (since originally filtered on just state of Arizona) assigned value of “Not in Area” and later removed from dataset.

**Slide 6a**

* **Feature Selection/Reduction**
  + - Dropped the following columns:
      * Duplicates
      * Features unrelated to restaurants (i.e. hair type, appointment\_needed)
      * More than 25% missing values:
      * Temp variables I created that are no longer needed (is\_restaurant, state)
      * Other variables that I deemed are unimportant (i.e. type - all business, business hours, good\_for\_breakfast)
    - Reduced variable count from 105 to 35

**Slide 6b**

* **Feature Creation**
  + - Created New Feature: food\_types
      * Used case\_when from dplyr package and str\_detect deom stringr package to extract text from categories field and put in new column called “food\_type”
      * Since case when pulls the first field it finds, I arranged food\_types to extract in order: more detailed food types prior to wider categories, for example, xxxx before xxxx

**Slide 7**

***<Table of Variables After Cleaning>***

**Slide 8a**

**Preliminary EDA in Python**

* **Histograms**
  + Stars target variable
    - Skew and kurtosis from scipy package
      * Explain skew and kurtosis
      * Normally distributed? (not necessary for decision tree)

**Slide 8b**

* **Frequency Counts of Categorical Data**
  + using *dfsummary* function from summarytools package
    - provided the variable name, data type, all possible values, frequency percentages of valid rows at each level, small graph of frequency distributions or small histograms of numerical variables, the total number of valid and missing rows for each variable, all in a concise table format.

**Slide 9-11**

***Detailed EDA***

* + **Boxplot of stars variable**
  + **Scatter plots – look into potential correlations – facet wrap**
  + **Bar charts, including stacked bar charts (i.e. attire), facet wrap**
  + **Cross tabs (tables)**

**Slide 12**

**Preliminary Results**

* *Model Selection -* Decision Trees
* Correlations with the stars target variable.
  + - Positive correlations between ambiance variables (divey\_false, classy\_false, touristy\_false, hipster\_false, trendy\_false, intimate\_false, casual\_true, romantic\_false, and upscale\_false),
    - two of the parking variables (parking\_lot\_true, packing\_validated\_false) I would expect that a model created based on the top attributes, would allow accurate business rating predictions.

**Slide 13**

**Modeling – Random Forest**

* Model Preparation:
  + Rename levels in stars variable from 1, 1.5, 2, etc. to ‘one’, ‘one\_five’, ‘two’, etc.
  + Filter on attributes, stars, review\_count, open, city, food\_type
  + Omit n/a values – normally not required for random forests but I was getting errors
  + Randomly split into train and test sets using 60/40 ratio
* Near Zero Variances:
  + Identify near zero variances
  + Remove near zero variances from dataset
* Ran numerous models to get determine best fit:
* Pick best model type and tune hyperparameters

**Slide 15**

**References:**

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112, pp. 3-7). New York: springer.

Abbott, D. (2014). *Applied predictive analytics: Principles and techniques for the professional data analyst*. John Wiley & Sons.

[add other references]