Term Project Data
Jordan Montgomery
QMB6358
Due 11/8/2020

Are Emily and Greg More Employable Than Lakisha and Jamal?

Building a regression model to identify resume attributes which correlate to receiving a call from an employer.

Project Scope – UPDATE:

Upon review of the data using Python, I have decided to only use 17 of the 26 exploratory variables for the analysis with 'call' – whether or not the resume garnered a call back – being the dependent variable:

1	**name**
2	**gender**
3	**ethnicity**
4	**quality**
5	**city**
6	**jobs**
7	**experience**
8	**holes**
9	**computer**

10	**college**
11	**minimum**
12	**equal**
13	**wanted**
14	**reqexp**
15	**reqeduc**
16	**reqcomp**
17	**industry**

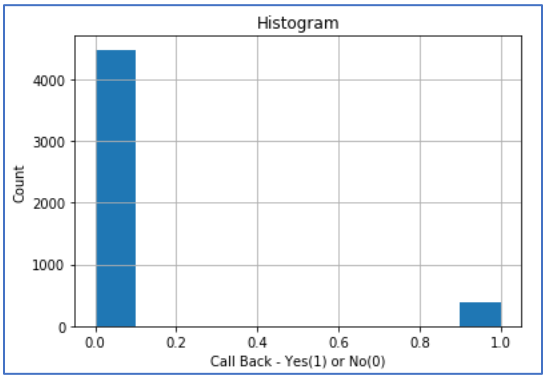
18	**honors**
19	**volunteer**
20	**military**
21	**school**
22	**email**
23	**special**
24	**requirements**
25	**reqcomm**
26	**reqorg**

In the **ResumeNames.xlsx** file, the data is defined and sorted on the **Variable Details** tab.

The **Project_Data_-_Analysis.py** file in the repository contains code to display some summary statistics and analysis of the data in Python. Screenshots of the code and charts are below as well:

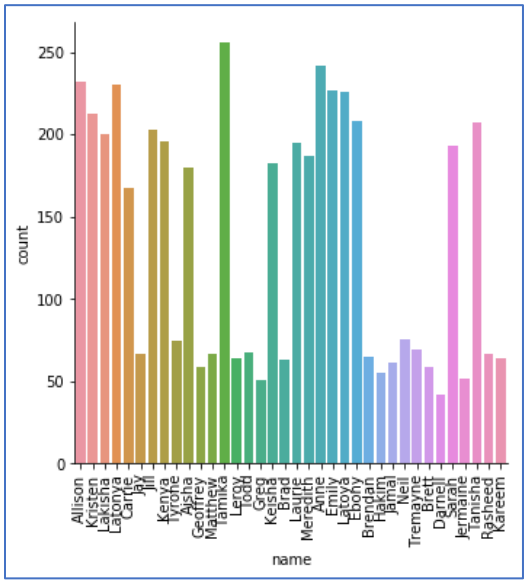
```
##################################################
# Import Modules.
##################################################


import os # To set working directory
import pandas as pd # To read and inspect data
import seaborn as sns


##################################################
# Set Working Directory.
##################################################


# Find out the current directory.
os.getcwd()
# Change to a new directory.
os.chdir('C:\\Users\\jorda\\OneDrive\\Documents\\UCF\\QMB6358\\QMB6358F20-JSM\\Term Project')
# Check that the change was successful.
os.getcwd()
```

```
##################################################
# 1) Load Data.
##################################################


ResumeNames = pd.read_csv('ResumeNames.csv')

print(ResumeNames)

# Removing data which will not be used in analysis
ResumeNames.drop(columns=['honors','volunteer','military','school','email','special','requirements','reqcomm','reqorg'])


##################################################
# 2) Summary statistics for dependent variable.
##################################################


# Changing "no" to "0" and "yes" to "1" in "call" column to use in histogram
ResumeNames.loc[ResumeNames["call"]=="no","call"] = 0
ResumeNames.loc[ResumeNames["call"]=="yes","call"] = 1

pd.DataFrame.hist(ResumeNames[['call']])
pl.title("Histogram")
pl.xlabel("Call Back - Yes(1) or No(0)")
pl.ylabel("Count")

ResumeNames.call.describe()
```




```
In [11]: ResumeNames.call.describe()
Out[11]:
count    4870.000000
mean        0.080493
std         0.272083
min         0.000000
25%         0.000000
50%         0.000000
75%         0.000000
max         1.000000
Name: call, dtype: float64
```

```
##################################################
# 3) Summary statistics for explanatory variables.
##################################################


# Plotting 'histogram' of Resume Names > 'name' column in dataset
# since it is the primary exploratory variable for the analysis
ResumeNames_plot = sns.factorplot(x="name", kind="count", data=ResumeNames)
ResumeNames_plot.set_xticklabels(rotation=90)
```




```
# Summary statistics of other exploratory variables
ResumeNames.name.describe()
ResumeNames.gender.describe()
ResumeNames.ethnicity.describe()
ResumeNames.quality.describe()
ResumeNames.city.describe()
ResumeNames.jobs.describe()
ResumeNames.experience.describe()
ResumeNames.holes.describe()
ResumeNames.computer.describe()
ResumeNames.college.describe()
ResumeNames.minimum.describe()
ResumeNames.equal.describe()
ResumeNames.wanted.describe()
ResumeNames.reqexp.describe()
ResumeNames.reqeduc.describe()
ResumeNames.reqcomp.describe()
ResumeNames.industry.describe()


ResumeNames.describe()
```

References

Arel-Bundock, V. (2007). *Vincent Arel-Bundock's Github projects.* Retrieved from R Datasets: https://vincentarelbundock.github.io/Rdatasets/datasets.html