Northeastern University

Module 2 Capstone Project Proposal

Priyanka Singh, Riya Parimal Dalal and V S N Sai Krishna Mohan Kocherlakota

ALY6140

Prof. Roy Wada

16/11/2023

Introduction

We have used a dataset "The Current Population Survey (CPS)" ***(ICPSR, 2019)*** is a useful dataset for socioeconomic research and various analysis. The chosen dataset consists of 5000 rows and 13 variables. The variables consist of gender, income, age, employment status, weight etc. and have several entries in it. This report aims to provide an overview of the data set chosen which will highlight major points like dimensions, questions to investigate, summary statistics, graphical visualizations, and analysis for the same.

Dimensions of Dataset

The selected dataset comprises a substantial sample size, meeting the requirement of 5,000 records and includes 13 usable features. The dimensions of the dataset play a crucial role in ensuring the robustness of the analysis that will be done. Sharing a SS of the same from the code run in Jupyter Notebook. ***(Geeksfor Geeks, 2023)***

```
In [25]: # Get the dimensions
         dimensions = excel_data.shape
         print("Dimensions of the dataset:", dimensions)

         Dimensions of the dataset: (5000, 13)
```

Variables

1. HEFAMINC – This variable depicts the Family Income of the various members of a family and is a numerical value.
2. HWHHWGT – This variable indicates the household weight.
3. HRINTSTA – This indicates the Interview Status and has 4 categories – Interviews, Type A non-interview, Type B non-interview and Type C non -interview.
4. PREXPLF – This indicates the employment status of the population whether a person is employed or non-employed.
5. PESEX – Gender of the population, male or female.
6. PENATVTY – This indicates the country of birth.
7. PRTAGE – Age of the population
8. PRDTOCC1 – This variable indicates the occupation of the people.
9. PWFMWGT – It indicates the Family weight.
10. PWLGWGT - Longitudinal Weight
11. PWORWGT - Outgoing Rotation Weight
12. PWSSWGT – Final Weight
13. PWVETWGT – Veterans Weight

Questions to Investigate

1. What is the distribution of family income in the dataset?

Analysis – We had run a code to analyze the distribution of family income from the dataset and below is the output. We wanted to get an idea of how the distribution and frequency is occurring. From the output we can see that the total income distribution count is 4179 with unique ones as 16 and top income distribution is 150,000 or more and HEFAMINC is the variable indicated in the table***. (Verma, 2020)***

```
In [27]: # Questions to Investigate:
         # Income Distribution: What is the distribution of family income in the dataset?
         # 1. Income Distribution
         income_distribution = excel_data['HEFAMINC'].describe()
         print("\nIncome Distribution:")
         print(income_distribution)
```

```
Income Distribution:
count                   4179
unique                    16
top       (16) 150,000 or more
freq                     522
Name: HEFAMINC, dtype: object
```

2. How many individuals in the dataset are employed? What is the distribution of employment statuses?

   Analysis – PREXPLF is the variable indicated in the table for employment status. From the output we can understand that the number of employed people from the dataset is 1886 and unemployed ones are 67.

```
In [28]: # 2. Employment Insights
         employment_status_counts = excel_data['PREXPLF'].value_counts()
         print("\nEmployment Status Distribution:")
         print(employment_status_counts)
```

```
Employment Status Distribution:
PREXPLF
(1) Employed      1886
(2) Unemployed      67
Name: count, dtype: int64
```

3. How is the dataset distributed by gender? What is the average age of individuals in the dataset?

   Analysis – PESEX is the variable is the dataset which depicts the gender and PRTAGE is for age variable. We had done an analysis for gender age and from the output we can see that out of the total population we have 2163 as the female and male are 2016. The average age amongst the population is 44.

```
In [29]: # 3. Demographic Analysis
gender_distribution = excel_data['PESEX'].value_counts()
average_age = excel_data['PRTAGE'].mean()
print("\nGender Distribution:")
print(gender_distribution)
print("\nAverage Age in the Dataset:", average_age)
```

```
Gender Distribution:
PESEX
(2) Female    2163
(1) Male      2016
Name: count, dtype: int64

Average Age in the Dataset: 44.08686288585786
```

4. What is the distribution of individuals based on their country of birth?

   Analysis – We wanted to carry out an analysis for country of Birth and is indicated by PENATVTY. From the output we can see that there are multiple countries and the highest births have occurred in United States. Python helps us to display the count, frequency of any variable from a dataset.

```
In [30]: # 4. Country of Birth
country_of_birth_distribution = excel_data['PENATVTY'].value_counts()
print("\nCountry of Birth Distribution:")
print(country_of_birth_distribution)
```

```
Country of Birth Distribution:
PENATVTY
(057) United States    3531
(327) Cuba               45
(073) Puerto Rico        45
(207) China              43
(303) Mexico             39
                       ...
(157) Lithuania           1
(206) Cambodia            1
(328) Dominica            1
(102) Austria             1
(310) Belize              1
Name: count, Length: 93, dtype: int64
```

5. What is the distribution of household weights?

   Analysis – From the output we can see that for Household weight distribution the count is 5000 and the mean 2388.88, the min value is 0 and max is 9065.664. The standard deviation is 1494.271 and the Inter Quartile values for 25%, 50% and 75% are 1004.822, 2904.956, 3460.50.

   Similarly, we calculated the descriptive statistics for the longitudinal weight and below are the number in the output.

```
In [31]: # 5. Weight Analysis
household_weight_distribution = excel_data['HWHHWGT'].describe()
longitudinal_weight_distribution = excel_data['PWLGWGT'].describe()
print("\nHousehold Weight Distribution:")
print(household_weight_distribution)
print("\nLongitudinal Weight Distribution:")
print(longitudinal_weight_distribution)
```

```
Household Weight Distribution:
count    5000.000000
mean     2388.882525
std      1494.271754
min         0.000000
25%      1004.822000
50%      2904.956500
75%      3460.504000
max      9065.664000
Name: HWHHWGT, dtype: float64

Longitudinal Weight Distribution:
count     5000.000000
mean      2109.146015
std       2391.514162
min          0.000000
25%          0.000000
50%        493.154950
75%       4508.279500
max      12966.150000
Name: PWLGWGT, dtype: float64
```

Summary Statistics Table for Categorical Variables

1. Income (HEFAMINC) - From the table we can see that for salary range 150,000 or more the frequency count is 522 and proportion is 0.124910 and similarly for other range the values are mentioned below. ***(Codeacademy)***

Frequency Count and Proportions for Employment Status:

HEFAMINC	Frequency Count	Proportion
(16) 150,000 or more	522	12.49%
(13) 60,000 to 74,999	493	11.8%
(15) 100,000 to 149,999	484	11.58%
(14) 75,000 to 99,999	409	9.79%
(11) 40,000 to 49,999	392	9.38%
(12) 50,000 to 59,999	318	7.61%
(09) 30,000 to 34,999	249	5.96%
(10) 35,000 to 39,999	248	5.93%
(08) 25,000 to 29,999	245	5.86%
(07) 20,000 to 24,999	219	5.24%
(06) 15,000 to 19,999	169	4.04%
(05) 12,500 to 14,999	123	2.94%
(04) 10,000 to 12,499	98	2.35%
(01) Less than $5,000	86	2.06%
(03) 7,500 to 9,999	77	1.84%
(02) 5,000 to 7,499	47	1.12%

2. Interview Status - HRINTSTA

Frequency Count and Proportions for Interview Status:

HRINTSTA	Frequency Count	Proportion
(1) Interview	4179	83.58%
(3) Type B non-interview	801	16.02%
(2) Type A non-interview	18	0.36%
(4) Type C non-interview	2	0.04%

3. Gender – PESEX

Frequency Count and Proportions for Gender:

PESEX	Frequency Count	Proportion
(2) Female	2163	51.76%
(1) Male	2016	48.24%

Summary Statistics Table for Numerical Variables

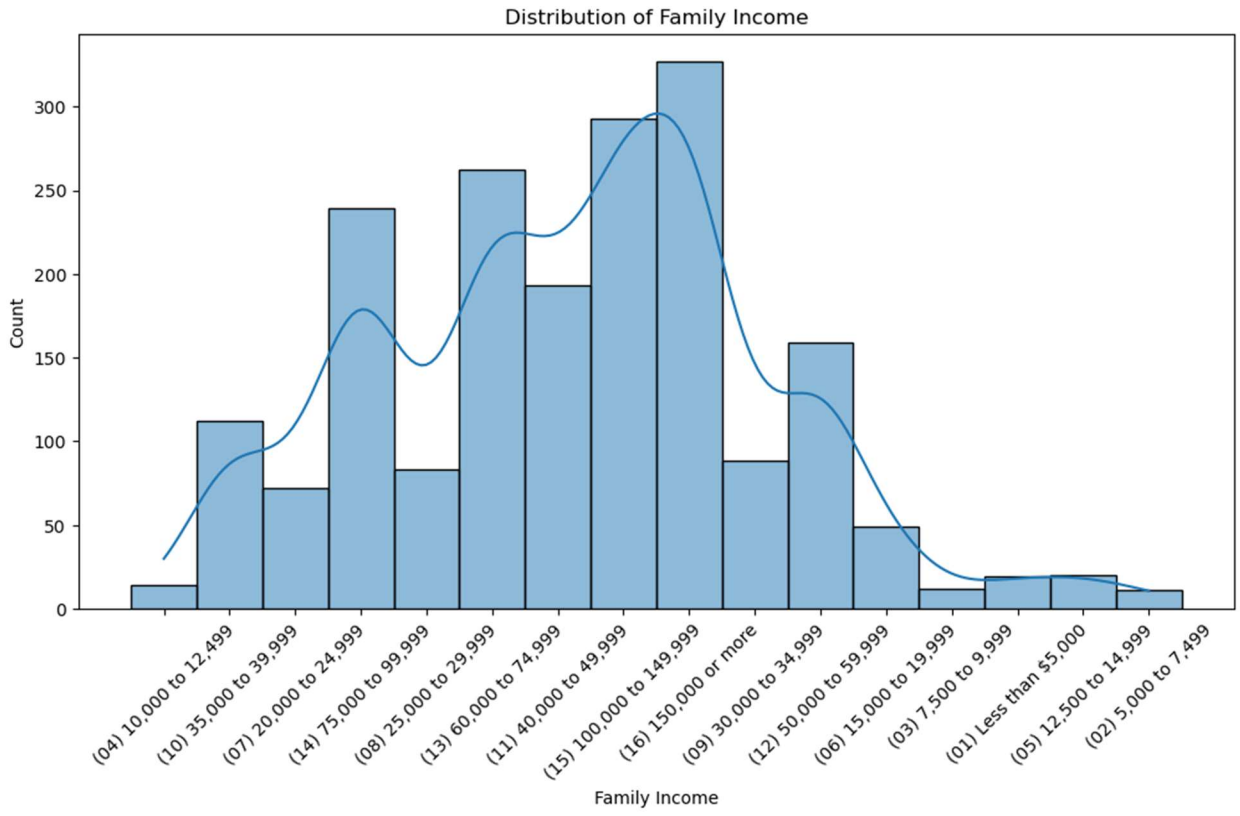
1. Summary Statistics for Numerical Variables was calculated and the variables are HWHHWGT, PRTAGE,PWFMWGT,PWLGWGT,PWORWGT. The mean, SD, min and max value, Interquartile range is mentioned for each variable.

Summary Statistics for Numerical Variables:

	HWHHWGT	PRTAGE	PWFMWGT	PWLGWGT	PWORWGT
count	5000.0	4179.0	5000.0	5000.0	5000.0
mean	2388.88252498	44.08686288585786	2423.03258354	2109.14601484	2264.14946154
std	1494.2717542252772	23.62077460635923	1537.3503658973052	2391.5141623915642	5023.167621402594
min	0.0	0.0	0.0	0.0	0.0
25%	1004.822	24.0	979.526525	0.0	0.0
50%	2904.9565000000002	46.0	2903.49	493.15495	0.0
75%	3460.504	63.0	3520.6305	4508.279500000001	0.0
max	9065.664	85.0	9639.847	12966.15	35534.11

Models

Visualization of Distribution of Family Income – We have made a graph for Family income and from the graph we can see that the highest family income is 100000 to 150000 or more.



Three models for analysis are as follows –

1. **Logistic Regression –**
   Logistic Regression serves as a fundamental Machine Learning classification algorithm, primarily designed to estimate the probability of a categorical dependent variable. In this context, the dependent variable is typically binary, encapsulating data coded as 1 (representing positive outcomes like success) or 0 (denoting negative outcomes like failure). Put simply, logistic regression models the probability, P(Y=1), as a function of the predictor variables (X).
   Rationale - Logistic Regression ***(Python,2023)*** is suitable for predicting binary outcomes, making it easy for investigating factors influencing employment status such as income, gender, age etc.

   Output –

```
Logistic Regression:
[[  0   0   0   0   0   0   0   0   0   0   0   0   1   0   0  24   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   1   1   0   5   0]
 [  0   0   0   0   0   0   0   0   0   0   1   0   1   1   0  15   1]
 [  0   0   0   0   0   0   0   0   0   0   0   0   1   0   0  16   1]
 [  0   0   0   0   0   0   0   0   0   0   0   0   1   0   0  18   1]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   3   0  32   1]
 [  0   0   0   0   0   0   0   0   0   0   0   0   2   4   0  31   1]
 [  0   0   0   0   0   0   0   0   0   0   0   0   8   3   0  38   1]
 [  0   0   0   0   0   0   0   0   0   0   0   0   4   2   0  35   2]
 [  0   0   0   0   0   0   0   0   0   0   1   0   4   5   0  34   4]
 [  0   0   0   0   0   0   0   0   0   0   0   0   6   3   0  70   6]
 [  0   0   0   0   0   0   0   0   0   0   0   0   3   7   0  70   0]
 [  0   0   0   0   0   0   0   0   0   0   2   0   5   3   0  88   2]
 [  0   0   0   0   0   0   0   0   0   0   0   0   1   6   0  62   5]
 [  0   0   0   0   0   0   0   0   0   0   0   0   4   6   0  78   4]
 [  0   0   0   0   0   0   0   0   0   0   0   0   6   6   0  80   7]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 166]]
```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	25
1	0.00	0.00	0.00	7
2	0.00	0.00	0.00	19
3	0.00	0.00	0.00	18
4	0.00	0.00	0.00	20
5	0.00	0.00	0.00	36
6	0.00	0.00	0.00	38
7	0.00	0.00	0.00	50
8	0.00	0.00	0.00	43
9	0.00	0.00	0.00	48
10	0.00	0.00	0.00	85
11	0.00	0.00	0.00	80
12	0.10	0.05	0.07	100
13	0.12	0.08	0.10	74
14	0.00	0.00	0.00	92
15	0.11	0.81	0.20	99
16	0.82	1.00	0.90	166
accuracy			0.26	1000
macro avg	0.07	0.11	0.07	1000
weighted avg	0.17	0.26	0.18	1000

```
Accuracy: 0.257
```

2. **Random Forest Classifier** - The Random Forest classifier is an ensemble method that relies on a collection of individual decision trees created from randomly chosen subsets of the training data. This algorithm consolidates the predictions of multiple decision trees to determine the ultimate class for a test object.

   Rational- Random Forest classifier to build, train, make predictions, and evaluate the model. After fitting the model to the training data (X_train and y_train), predictions are made on the test set (X_test), and model performance metrics such as confusion matrix, classification report, and accuracy score are printed for evaluation.

   Output –

```
Decision Tree Classifier:
[[  4   0   0   0   0   1   2   0   0   1   1   2   2   3   5   4   0]
 [  0   0   0   0   0   0   1   0   0   0   0   0   1   0   3   2   0]
 [  0   0   2   0   0   1   0   3   1   1   1   3   0   4   1   2   0]
 [  1   0   0   3   0   1   0   1   1   0   2   2   3   0   4   0   0]
 [  0   0   1   1   6   2   0   1   2   1   0   0   1   1   1   3   0]
 [  0   1   0   2   2   7   4   0   1   4   2   2   5   2   1   3   0]
 [  1   0   0   0   2   1  13   2   4   3   1   1   3   2   2   3   0]
 [  1   1   0   0   1   5   4  12   2   3   8   0   3   4   2   4   0]
 [  3   0   1   0   1   0   3   3  16   1   0   3   1   4   5   2   0]
 [  1   1   3   2   2   2   3   1   2  19   4   2   0   2   1   3   0]
 [  2   1   0   0   2   4   3   6   4   6  28   4   8   3   6   8   0]
 [  1   0   1   2   1   1   4   2   3   2   3  33  10   2   8   7   0]
 [  0   0   0   6   3   3   3   6   3   4   6   7  40   3   9   7   0]
 [  0   0   0   1   1   4   5   6   1   2   6   1   3  33   9   2   0]
 [  0   0   0   1   0   3   5   2   4   2   4   8   5  10  41   7   0]
 [  2   1   0   2   0   3   4   2   5   2   4   4   3   6   8  53   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 166]]
              precision    recall  f1-score   support

           0       0.25      0.16      0.20        25
           1       0.00      0.00      0.00         7
           2       0.25      0.11      0.15        19
           3       0.15      0.17      0.16        18
           4       0.29      0.30      0.29        20
           5       0.18      0.19      0.19        36
           6       0.24      0.34      0.28        38
           7       0.26      0.24      0.25        50
           8       0.33      0.37      0.35        43
           9       0.37      0.40      0.38        48
          10       0.40      0.33      0.36        85
          11       0.46      0.41      0.43        80
          12       0.45      0.40      0.43       100
          13       0.42      0.45      0.43        74
          14       0.39      0.45      0.41        92
          15       0.48      0.54      0.51        99
          16       1.00      1.00      1.00       166

    accuracy                           0.48      1000
   macro avg       0.35      0.34      0.34      1000
weighted avg       0.48      0.48      0.47      1000

Accuracy: 0.476
```

3. **Decision Tree Classifier** - A Decision Tree Classifier is a supervised machine learning algorithm used primarily for classification tasks. It operates by recursively partitioning the feature space into distinct regions or classes based on a sequence of decision rules inferred from the training data.

   Rational- Trains (using fit) and predicts with a Decision Tree Classifier. It then evaluates the model's performance on test data, displaying a confusion matrix and classification report to assess its accuracy and predictive capability for the given task.

   Output –

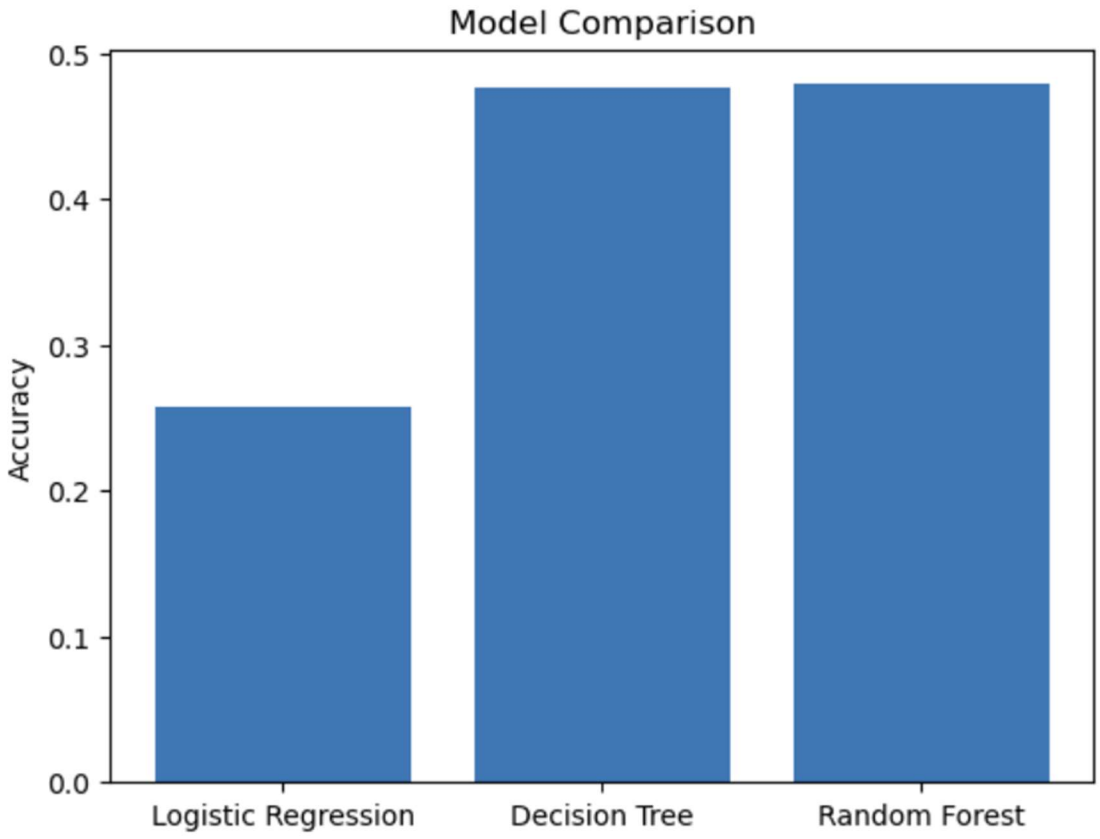
```
Random Forest Classifier:
[[  4   0   0   0   0   1   2   0   0   1   1   1   3   3   5   4   0]
 [  0   0   0   0   0   0   1   0   0   0   0   0   1   0   3   2   0]
 [  0   0   2   0   0   1   0   3   1   1   1   3   0   4   1   2   0]
 [  1   0   0   2   0   2   0   1   1   0   2   2   3   0   4   0   0]
 [  0   0   1   1   6   2   0   1   2   1   0   0   1   1   1   3   0]
 [  0   1   0   2   1   8   4   0   1   3   2   2   5   3   1   3   0]
 [  1   0   0   0   2   1  13   2   4   3   1   1   3   2   2   3   0]
 [  1   1   0   0   1   5   3  12   2   3   8   0   4   4   2   4   0]
 [  3   0   1   0   1   0   3   3  15   1   0   3   1   5   5   2   0]
 [  1   1   2   1   2   3   2   1   2  20   4   2   0   2   2   3   0]
 [  2   1   0   0   2   4   3   6   4   6  27   4   8   4   6   8   0]
 [  1   0   1   1   1   1   3   2   3   3   3  34  10   2   8   7   0]
 [  0   0   0   5   3   4   2   6   3   3   6   6  41   3   9   9   0]
 [  0   0   0   1   1   3   5   6   0   2   6   2   3  33   9   3   0]
 [  0   0   0   1   0   1   3   3   4   2   5   7   5  10  43   8   0]
 [  2   1   0   2   0   3   4   2   5   2   4   4   3   6   8  53   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 166]]
```

	precision	recall	f1-score	support
0	0.25	0.16	0.20	25
1	0.00	0.00	0.00	7
2	0.29	0.11	0.15	19
3	0.12	0.11	0.12	18
4	0.30	0.30	0.30	20
5	0.21	0.22	0.21	36
6	0.27	0.34	0.30	38
7	0.25	0.24	0.24	50
8	0.32	0.35	0.33	43
9	0.39	0.42	0.40	48
10	0.39	0.32	0.35	85
11	0.48	0.42	0.45	80
12	0.45	0.41	0.43	100
13	0.40	0.45	0.42	74
14	0.39	0.47	0.43	92
15	0.46	0.54	0.50	99
16	1.00	1.00	1.00	166
accuracy			0.48	1000
macro avg	0.35	0.34	0.34	1000
weighted avg	0.48	0.48	0.48	1000

```
Accuracy: 0.479
```

Graphical representation – A bar graph comparing the accuracies of three models: Logistic Regression, Decision Tree, and Random Forest. It utilizes the accuracy score function to compute the accuracy for each model on the test data and then visualizes these accuracies using Matplotlib. From the histogram we can see that out of all the 3 models used Random forest has the highest accuracy which lies between 0.4 to 0.5 followed by Decision tree and lastly Logistic Regression.This graph provides a visual comparison of the accuracies achieved by the three models on the test data, aiding in the assessment and comparison of their predictive performance.

Output



Conclusion

This exploratory data analysis provides initial insights into the dataset, highlighting the distributions and relationships between key variables. This preliminary exploration of the Current Population Survey dataset lays the groundwork for a comprehensive Capstone Project. The dataset's dimensions, encompassing over 5000 records and 13 variables, ensure a robust analysis. The questions help us to understand the sociodemographic aspects, guiding the investigation toward various factors influencing employment, income, and weight other critical variables.

The summary statistics table offers a concise snapshot of the dataset's central tendencies, paving the way for more in-depth analyses. The selected models—Logistic Regression and Random Forest—reflect a thoughtful approach to exploring both linear and non-linear relationships within the data. These models are poised to provide valuable insights into the complexities of the socioeconomic landscape captured by the CPS dataset.

References

1) Current Population Survey, September 2017: Volunteering and Civic Life supplement. (2019, May 20). https://www.icpsr.umich.edu/web/ICPSR/studies/37303/variables?q=employment
2) GeeksforGeeks. (2023, September 4). Pandas DF.size df.shape and Df.Ndim methods. https://www.geeksforgeeks.org/python-pandas-df-size-df-shape-and-df-ndim/
3) Verma, J. (2020, October 7). How to calculate summary statistics in Python? - AskPython. https://www.askpython.com/python/examples/calculate-summary-statistics
4) Summary Statistics for Categorical data: Summary Statistics for Categorical Data cheatsheet | Codecademy. (n.d.). Codecademy. https://www.codecademy.com/learn/stats-summary-statistics-for-categorical-data/modules/stats-summary-statistics-for-categorical-data/cheatsheet
5) How to calculate summary statistics — pandas 2.1.3 documentation. (n.d.). https://pandas.pydata.org/docs/getting_started/intro_tutorials/06_calculate_statistics.html
6) Python, R. (2023, June 26). Logistic regression in Python. https://realpython.com/logistic-regression-python/

Appendix

The appendix section will include relevant portions of code used for data preprocessing, analysis, and model building. Summary statistics or visualizations that provide more detailed insights. A concise description of each variable in the dataset, including data types and possible values. The Python Script **“Group5_ALY6140_Capstone”** and **Report “Capstone Project – Group 5”** is made and ready for submission.