UDACITY

‹ Return to Classroom

Investigate a Dataset

REVIEW

HISTORY

Meets Specifications

Congratulations! You passed in this project!

In the review, I made some comments and also suggested a few links for further reading if you want to go deeper into a few aspects seen in this project. I hope they are useful to you :)

All the best and keep on learning!

Code Functionality

All code is functional and produces no errors when run. The code given is sufficient to reproduce the results described.

The project uses NumPy arrays and Pandas Series and DataFrames where appropriate rather than Python lists and dictionaries. Where possible, vectorized operations and built-in functions are used instead of loops.

Very well done using built-in functions, like `head()`, `info()`, `describe()` and many others!

Further reading

Here is the documentation and tutorial for several other functionalities that we can find within the Pandas module that are very useful.

The code makes use of functions to avoid repetitive code. The code contains good comments and variable names, making it easy to read.

Improvement suggestion

There is some repetitive code that could be removed if we add a function. The green bar plots share the exact same code, the only changes being the name of the disease. So a possible function that can be useful to plot all of them is:

```
def plot_disease_noshow(column_name = ""):
    df.groupby(column_name)['no_show'].value_counts(normalize = 'true').plot(kind =
'bar', color = "green")
    plt.title("Comparison between those who showed to those who did not according to
{}".format(column_name));
    plt.ylabel("percentage of {}".format(column_name)) # Plot to show the comparison
of the disease
```

This way, you can do each of the plots just passing different `column_name`s:

```
plot_disease_noshow(column_name = "diabetes")
```

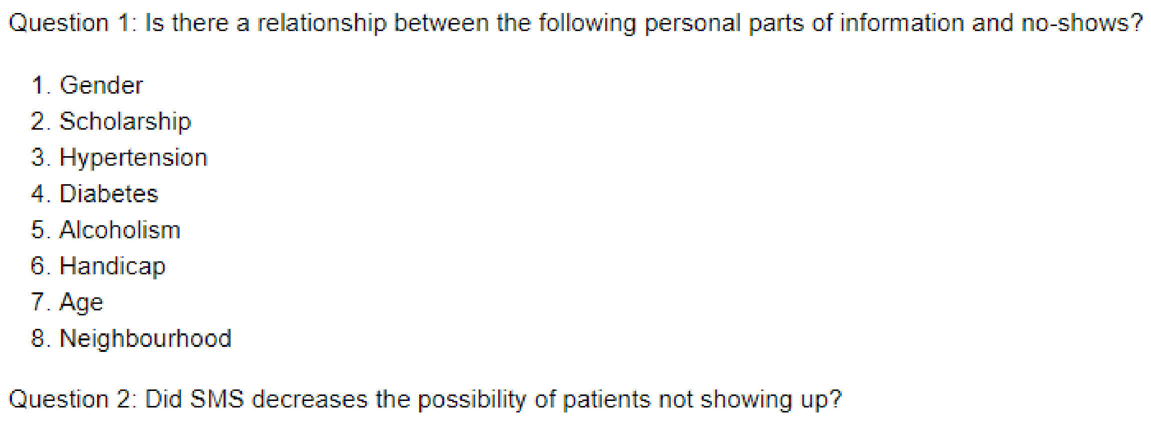
```
plot_disease_noshow(column_name = "alcoholism")
```

and so on!

Quality of Analysis

The project clearly states one or more questions, then addresses those questions in the rest of the analysis.

Two main questions were discussed in the project:



Question 1: Is there a relationship between the following personal parts of information and no-shows?

1. Gender
2. Scholarship
3. Hypertension
4. Diabetes
5. Alcoholism
6. Handicap
7. Age
8. Neighbourhood

Question 2: Did SMS decreases the possibility of patients not showing up?

Data Wrangling Phase

The project documents any changes that were made to clean the data, such as merging multiple files, handling missing values, etc.

The data wrangling phase was documented using markdown and inline comments.

Suggestion

When dealing with the Age values, you have removed some records using

```
#Taking rows with range from 0 to 100 in Age columns.
df = df[(df['age']>=0) & (df['age']<=100)]
```

This is a gerat solution when there is a small number of such records. However, when the number of missing/wrong values is larger (say 20%-30% of the whole dataset), removing those records might have a negative effect on our analysis, as we lose information about the other variables as well (imagine you remove rows that have only Age missing values, but all the other columns are complete). A better solution in these cases would be to impute the missing values rather than removing them. You can do this using either the median or mean value of the distribution of the column. You can also use more advanced imputation techniques that are already implemented in libraries such as missingpy.

Exploration Phase

The project investigates the stated question(s) from multiple angles. At least three variables are investigated using both single-variable (1d) and multiple-variable (2d) explorations.

Both kinds of exploration were done using histograms and bar plots. Excellent!

The project's visualizations are varied and show multiple comparisons and trends. Relevant statistics are computed throughout the analysis when an inference is made about the data.

At least two kinds of plots should be created as part of the explorations.

You have used two kinds of plots in your exploration and met this requirement!

Further reading

There are many more kinds of plots and the more kinds you know, the better you become in identifying the best one for your specific use case. Check here for 44 types of plots that you may find interesting! You can also look here to matplotlib plots with their implementations in Python!

Conclusions Phase

The results of the analysis are presented such that any limitations are clear. The analysis does not state or imply that one change causes another based solely on a correlation.

The conclusion step summarizes very well the findings you obtained during this data exploration and a limitation and a future work suggestion were mentioned. Great job!

Suggestion

In the last suggestion, you say

> for further analysis, if we add one more variable to the relationship maybe we will find a correlation.

It would be great if you also mention one variable you think would be helpful to find that correlation. This way, the reader of this report - maybe a colleague or boss - may be able to look for this specific data in order to help the research.

Communication

Reasoning is provided for each analysis decision, plot, and statistical summary.

Visualizations made in the project depict the data in an appropriate manner that allows plots to be readily interpreted.

The visualizations are properly labeled and titled.

Further reading

Now that you are familiar with matplotlib, you should feel free to learn about other plotting libraries. One of them is called plotly in which you can make interactive plots rather than the static ones obtained using matplotlib and seaborn. Check it out!

DOWNLOAD PROJECT

RETURN TO PATH

Rate this review

START