Investigate a Dataset

Requires Changes

2 specifications require changes

Dear student,
Since the previous review the following section (s) that **REQUIRED CHANGES** have been successfully updated and **MET SPECIFICATIONS**. The section(s) is/are:

- Both UNivariate and MUltivariate Explorations are presented
- Each step of analysis and visualization contains explanations enabling the report to flow smoothly just as a story!
- All visualizations are titled, labeled.

As much as the project has improved since the previous review. There are still two more sections that require changes. These are:

- The report has one type of visualization multiple times. It is required to have at least two different types of visualizations for variety.
- A conclusion is provided but there are no limitations present within it. Ensure to add the limitations.

For the sections that **REQUIRE CHANGES** and sections where necessary comments, recommendations, and suggestions have been provided to help better understand the requirements and assist with helping to pass this project successfully.

All the best for the next submission!
Cheers!

Code Functionality

✓

- All code is functional and produces no errors when run.

- **The code given is sufficient to reproduce the results described.**

Good start!
The following requirements for this section **MEETS SPECIFICATIONS!**:

- The code runs well without any errors presented/ outputted.
- The code is properly formatted/documented and is sufficient enough to produce the results described

RECOMMENDATION

- For better readability, it is better to remove the **TIPS** and **HEADERS** for better readability.
- Secondly, remove the Udacity instruction comments that are TO DOs

IMPORTANT NOTE FOR CODING PROJECTS(FRIENDLY REVIEWER ADVICE)

When programming code, as developers we do encounter the dilemma of "annoying" bugs/errors/issues even indentation and formatting issues
. Many big companies and professional developers actually look at this and take it heavily into consideration. It is always recommended practice good object-oriented programming skills and learn to debug and format the code after finishing.
Now, there are many ways to do this project, but it is best suited and recommended by Udacity to do it in the **Jupyter-Notebook** because all code, visualizations, and explanations can be done in one space.
However, the drawback with Jupyter-Notebook is that sometimes debugging errors(if they occur) can be difficult.
Luckily, IDEs such as Spyder-IDE and PyCharm-IDE have been updated and can now integrate the **Juptyter Notebook** extension `ipynb` within them. This makes debugging, formating, and editing a whole lot easier than before!

NOTE This is just for the Python programming language. Though it can also be applied to future, Data Analysts, Sciences, and Python projects as well!

✓

- **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.**

Good work! The following requirements for this section **MEETS SPECIFICATIONS!**:

- Project successfully uses Numpy arrays, Pandas Series and Data frames where appropriate
- Practices effeciency in cases where vectorized operations and built in functions are used instead of "for" loops!

Suggestion

There are many built-in methods that can be used in the field of Python and Data Analysis/Sciences but here are a few which are commonly used and applicable for these projects:

- Boolean-Indexing
- Group-by
- Value-Counts
- Series.map
- Working-with-text-data

NOTE The project is not limited only to these methods mentioned above. It all depends on the analysis conducted. There are many other methods that can be used. Ensure to check the official : Pandas-Data-Analysis-Documentation

✓

- **The code makes use of at least 1 function to avoid repetitive code.**
- **The code contains good comments and meaningful variable names, making it easy to read.**

Good work! The following requirements for this section **MEETS SPECIFICATIONS!**:

- The Code contains at least ONE re-usable function to negative repeptitive code
- The project contains good variable names
- The code has comments/docstrings where necessary in the project

Friendly Reviewer Suggestion ONE

Glas to see a function in the project. As for a suggestion for the future for creating a function the easiest way is by plotting a visualization. This is helpful as if multiple visualizations are being plotted the function comes in handy instead of writing repetitive code.
An example can be demonstrated below you can fix the code up in your way but this is a rough template to use to create the function if you wish. The example will be creating a function for a histogram:

```
 def hist_plot_by(x, xlabel, ylabel):
''' A function to draw a histgram '''
     ax = plt.subplot(1,1,1)
     ax.hist(x)
     ax.set_xlabel(xlabel)
     ax.set_ylabel(ylabel)
   plt.show()
```

According to me and my advice to students, this is the easiest function to implement in the project. Obviously, you would not need to use this. A function can be created from any step of code. Secondly, to add to the point stray `for` loops are a " no-no " too! It is more efficient to put them in a function!
Finally, remember this is just a suggestion to add the most efficient function easily in this project! The function already in the project was well done!

Friendly Reviewer Suggestion TWO

As a developer/ programmer, it is good practice to add comments and/or docstrings to your code. Comments and/or docstrings are important because they briefly tell the reader what type of variable, function, or execution is being done within the code block. Comments and/ or docstrings do not need to be paragraphs, but a good sentence or two will do.
No matter how many lines of code it may be from one line of code to multiple lines of code, it is necessary.

This article is a very good read as it pertains to this section Using-Python-comments-and-docstrings.
Finally, it is good to know the difference when to use both!

Quality of Analysis

✓

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

Good work! The following requirements for this section **MEETS SPECIFICATIONS!**:

- The project has an Introductionary Paragraph with the questions that is very appealing to readers. It gives the agenda to the Project and what to expect when reading through the document.

TIP

Great Introductionary paragraphs that are properly written provide excellent anticipation for the reader and what to expect. Do this for all projects in the future as it gives insight.

Data Wrangling Phase

✓

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

Good work! The following requirements for this section **MEETS SPECIFICATIONS!**:

- The project successfully shows the Data Wrangling steps in code which are complete, valid, accurate and consistent.
- The Data Wrangling section is documented in writing and coincides with the coding steps used to wrangle the data.

Suggestion

The most important aspect of Data Wrangling is to clean or transform the data preparing it for analysis.

One main issue is having missing data while conducting analysis, which can provide skewed/biased results. Luckily there are a few methods that Pandas provide to deal with these issues:

- The first thing to do is to always Identify the missing values within the dataset. The few steps after this explain how to deal with the missing data
- If there are columns with a few rows of missing data the Dropna methodcould be used to drop the missing rows.
- If there are rows with missing data the Fillna-method can be used instead of dropping them completely (This method can vary with the data and the project)
- The final option is if there are way too many missing values within a column or f it is not wanted, it is best to drop the column completely using the Drop-column-method

Data Wrangling does not only involve Identifying and dealing with missing values but also involves in transforming the data to a more effective state to target the analysis. Here are other wrangling methods:

- Binning or Cutting Groups continuous or numerical values into smaller groups or 'bins'
- Pandas-DummiesTransforms categorical data into dummy/indicator variables
- Working-with-text-data
- Creating new columns

All these methods can be applied to this project!

Exploration Phase

✓

- **The project investigates the stated question(s) from multiple angles.**
- **The project explores at least three variables in relation to the primary question. This can be an exploratory relationship between three variables of interest, or looking at how two independent variables relate to a single dependent variable of interest.**
- **The project performs both single-variable (1d) and multiple-variable (2d) explorations.**

Good work! The following requirements for this section **MEETS SPECIFICATIONS!**:

- The project successfully investigates/explores the analysis from multiple angles
- Has at least THREE cases of Univariate explorations (1D). Where only Single Variables are explored
- Has at least THREE cases of Multi-variable explorations (2D). Where Two or More Variables are explored.

SUGGESTION

Here is a good refresher to remind us what is Univariate and Multivariable explorations and how to create both simple but efficient univariate and multivariable explorations via code with examples:

1. Simple-Univariate-Explorations
2. SImple-Multivariable-Explorations

ADVICE TO CONSIDER: Quite a bit of student, even analysts have a misconception that visualization is automatically a multivariable exploration. This is not true! There are different types of visualizations for each type of exploration whether it be univariate or multivariable(Look at the sources for more details). Just keep this in mind when conducting explorations in this project or any future Data Analysis/Sciences projects.

- **The project's visualizations are varied and show multiple comparisons and trends.**
- **At least two kinds of plots should be created as part of the explorations.**
- **Relevant statistics are computed throughout the analysis when an inference is made about the data.**

Good work! The following requirements for this section do meet the criterion:

- The project's visualizations are appealing by showing multiple comparisions, trends in different cases and different variable combinations
- Each visualization appears to be relevant where necessary in making an inference about the data.

As seen the multiple plots are currently present. However, they are of the same type which is **Bar charts**.

Required

At least two different kinds of plots should be given that visualize the data in different ways.

Comments / Recommendations

The report has given multiple plots which all display the data in the same way. A more variety of visualizations is required to be used to display the data in other ways. There are many visualizations to choose from:

- Histograms
- Count plots
- Box plots
- Line graphs
- Pie charts

- Scatter plots
- Heat Maps
- 3d visualizations
- List goes on

SUGGESTION 2

These are the two most popular libraries for plotting visualizations in the Python programming world ensure to check out their galleries for more variety:

- Matplotlib-gallery
- Sea-born-gallery

NOTE:

- One key thing to understand when plotting visualizations is which visualization is best for the type of data. For example, the main data types are numerical and string/categorical. There are different types of visualizations that can be applicable to both. Keep this in mind when plotting!

To meet specifications

- Provide two different types of visualizations that depict the data in different ways

Conclusions Phase

- **The Conclusions have reflected on the steps taken during the data exploration.**
- **The Conclusions have summarized the main findings in relation to the question(s) provided at the beginning of the analysis accurately.**
- **The project has pointed out where additional research can be done or where additional information could be useful.**
- **The conclusion should have at least 1 limitation explained clearly.**
- **The analysis does not state or imply that one change causes another based solely on a correlation.**

Good work! The following requirement(s) does meet some of the criteria for this section:

- The Conclusion successfully summarizes the findings from the question(s) asked through the analysis.

Required

Almost had it. The reason this section has not met the specifications is that the project has not outlined any limitations within the conclusion section.

Comments/ Recommendation

The Limitations subsection can be placed after the conclusion. Limitations are the challenges that you personally face during this project or could have faced in the long run. To give you an idea to address the limitations or challenges you personally faced while doing this project, here are some factors to consider and can be addressed:

- Was the Missing data too much to handle?

- Was the data sufficient to prove your findings without wrangling much data?
- Was there too much missing data that possibly will skew the analysis
- Were methods difficult to find and implement for this project
- Did you as the student personally find it difficult to achieve certain tasks such as coding steps, writing explanations

These are examples of common limitations, that many persons face while doing this project, but everyone has different experiences explain yours!

REMINDER (ON HOW TO DOCUMENT THE CONCLUSION SECTION)

The proper structure to write a substantial-conclusion is:

1. Summarizing the findings from the questions explored
2. A limitations subsection
3. A reference (OPTIONAL)

To meet specifications

- Explain and give justification to the limitations or challenges you personally faced while doing this project.

Communication

✓

- **The code should have ideally the following sections: Introduction; Questions; Data Wrangling; Exploratory Data Analysis; Conclusions, Limitation.**
- **Reasoning is provided for each analysis decision, plot, and statistical summary.**
- **Interpretation of plots and application of statistical tests should be correct and without error.**
- **Comments are used within the code cells.**
- **Documented the flow of analysis in the mark-down cells.**

Good work! The following requirements for this section **MEETS SPECIFICATIONS!**:

- Reasoning is provided for each bit of analysis, visualization, and decision-making.
- Interpretations and explanations of creating and understanding the visualizations before plotting and after plotting.
- The written explanations documented are identifiable in Markdown cells

The report contains explanations for each step and each visualization making it flow smoothly just like a story!

Friendly Reviewer SUGGESTIONS/ TIPS

1. When explaining visualizations it is always helpful to provide statistical and numerical figures from the graph and include them within the reasoning.
2. Before implementing code within the code blocks. Create a Markdown cell and give an explanation of what you are about to do. The main question that always arises is why is code being written and what is it solving?

NOTE:

- Analyzing data is like writing a story there should be an explanation of why each step is being done. This includes both code cells and visualizations. It is just for you the storyteller to make sure there is a flow to it!
  Try to keep this in mind every time when doing a Data analysis report in fact, this is applicable to every project you do

within Udacity and the real world!

✓

Visualizations made in the project depict the data in an appropriate manner (i.e., has appropriate labels, scale, legends, and plot type) that allows plots to be readily interpreted.

Good work! The following requirements for this section **MEETS SPECIFICATIONS!**

Most of the visualizations contain the following:

- The x-axis and y-axis are properly labeled in each visualization.
- Each visualization has a title.
- Have a legend if necessary
- All axis data points are readable

REMINDER: The project requires more visualizations to be added. Ensure that when they are created that they follow the pointers mentioned above !

RESUBMIT

DOWNLOAD PROJECT

Best practices for your project resubmission

Ben shares 5 helpful tips to get you through revising and resubmitting your project.

Watch Video (3:01)

RETURN TO PATH

Rate this review

START