UDACITY

< Return to Classroom

Write a Data Science Blog Post

REVIEW

CODE REVIEW

HISTORY

Meets Specifications

Awesome 👏👏

Congratulations on passing this project. 🏆🏆

This project was not easy but you have done it gracefully. This is all because of your hard work and continuous evaluation. But still don't relax, keep exploring and learning from the references provided. Try to read more related articles on CRISP-DM. Try to solve and practice this kind of data analysis on Kaggle. This will bring a habit and will develop a mental model.

Extra Materials

The links below, provide more insights to these processes:

- CRISP-DM Overview- Data Science Project Management [Article]
- Manage your next Data Science Project with CRISP-DM METHOD [Article]
- Six steps in CRISP-DM the standard data mining process [Article]

You can also share your project on LinkedIn and ask the audience for necessary feedback or open the project for anyone to collaborate. This way you will find many interesting connections and engagement with others.

I wish you good luck. Looking forward to your success.
For any queries, you can ask on Knowledge Portal as well.

Stay ! Stay Safe

DON'T FORGET TO RATE MY WORK AS PROJECT REVIEWER! YOUR FEEDBACK IS VERY HELPFUL AND APPRECIATED.

Code Functionality and Readability

All the project code is contained in a Jupyter notebook, which demonstrates successful execution and output of the code.

Your code runs without errors. 👏

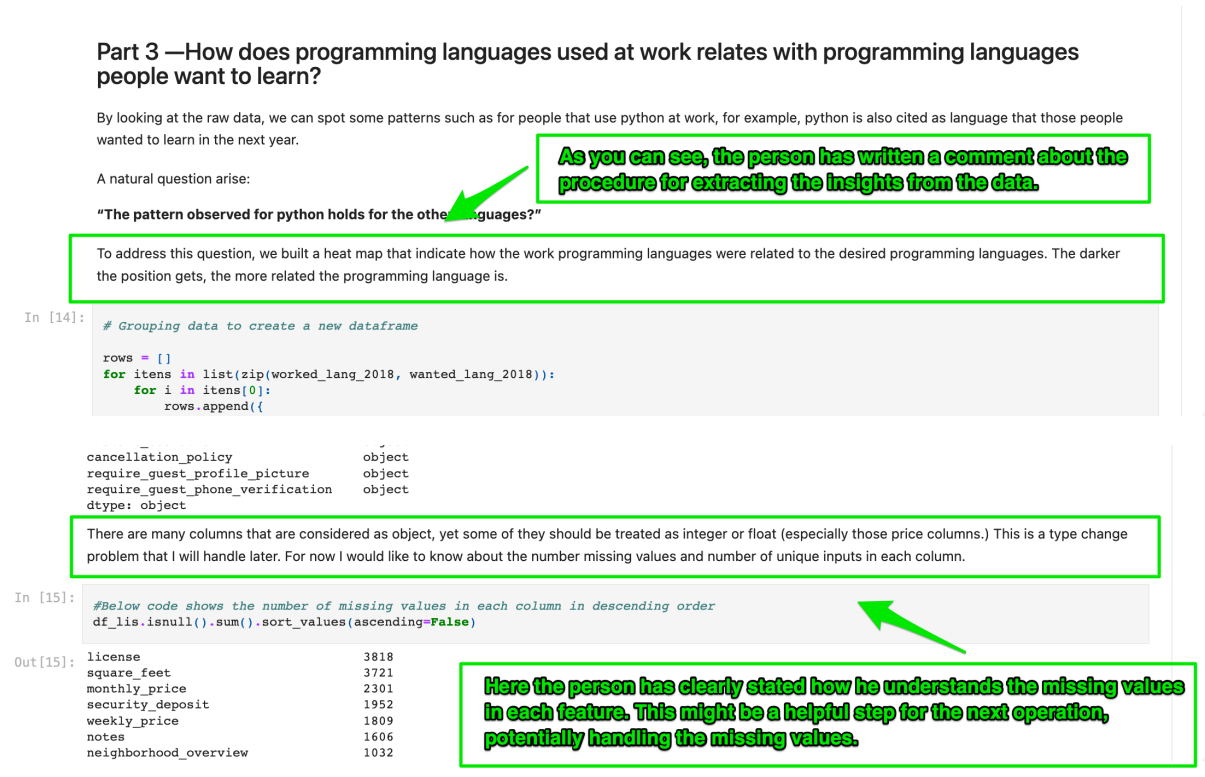
As a whole, you've accomplished a wonderful job of creating coding solutions throughout the project.

Code has easy-to-follow logical structure. The code uses comments effectively and/or Notebook Markdown cells correctly. The steps of the data science process (gather, assess, clean, analyze, model, visualize) are clearly identified with comments or Markdown cells, as well. The naming for variables and functions should be according to PEP8 style guide.

✅ Code has easy-to-follow logical structure. The code uses comments effectively and/or Notebook Markdown cells correctly.

Good Work! your comments are constructive in understanding the flow of the analysis. 💯

Markdown text is essential to making sure that our notebooks are easy to use and understand section by section. The **Markdown text is not a replacement for line or block comments** in the code cells but rather a place to provide a broader context for the code. **Here is an example:**



This results in a significant improvement in your notebook readability and explainability. 💪

✅ The steps of the data science process (gather, assess, clean, analyze, model, visualize) are clearly identified with comments or Markdown cells, as well

You have nicely segmented the notebook into several sections of Data Science processes.

Searching through the smaller section is accessible and generally identifiable when people quickly scroll through your work.

✅ The naming for variables and functions should be according to PEP8 style guide.

Your variables are correctly named and follow the pep8 guidelines.

You can also use the `pycodestyle` package to check your code against the pep8 styling guidelines. **Here is what you need to do:**

- First, install the package pycodestyle.

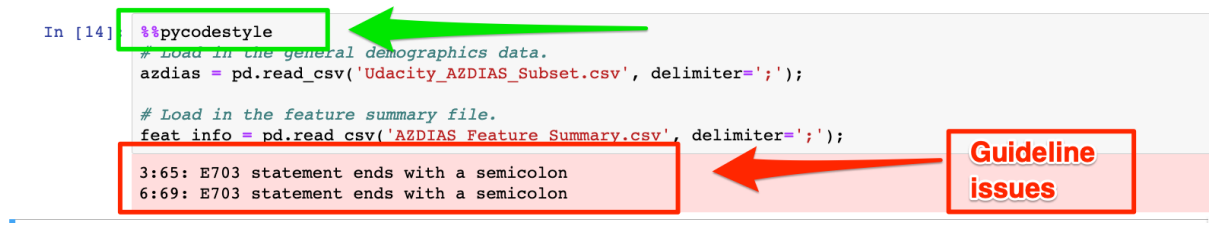
```
!pip install pycodestyle pycodestyle_magic
!pip install flake8
%load_ext pycodestyle_magic
```

NOTE: THIS NEEDS TO BE EXECUTED IN A NOTEBOOK CELL.

- Then you have to place the following command on the top of any code cell and execute it.

```
%%pycodestyle
```

Here is an example:



Here is the correction:

Useful References:

Jupyter Notebook Best Practices [Article]

Jupyter notebook shortcuts [Article]

PEP 8 -- Style Guide for Python Code [Documentation]

Code is well documented and uses functions and classes as necessary. All functions include document strings. DRY principles are implemented.

✅ Code is well documented and uses functions and classes as necessary.

Good job on implementing the function. This way you will be obeying the DRY principle.

Functions enable programmers to break down or decompose a problem into smaller chunks, each of which performs a particular task. Apart from that, you have nicely demonstrated the use of single-line comments to document your codes. Well Done! 👏

✅ All functions include document strings

All your functions contain a DOCSTRING. Good work!

Docstrings, or documentation strings, are valuable pieces of documentation that explain the functionality of any function or module in your code.

Useful References:

Python Docstrings [Article]

Docstring vs Comments [Article]

Do not Repeat Yourself [Youtube Video]

Data

Project follows the CRISP-DM process outlined for questions through communication. This can be done in the README or the notebook. If a question does not require machine learning, descriptive or inferential statistics should be used to create a compelling answer to a particular question.

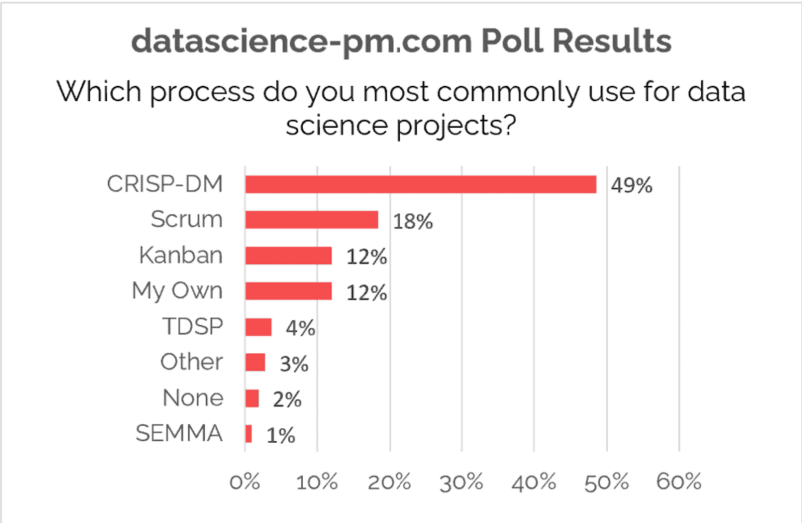
✅ Project follows the CRISP-DM process outlined for questions through communication

Your analysis does an exceptional job of following the CRISP-DM process. 💯

It is clear what part of the process you are on in each section of the notebook, which is challenging to do in a notebook. Nice job!

CRISP-DM is Still the Most Popular Framework for Executing Data Science Projects.

Based on the survey performed by KDnuggets, nearly half of the respondents most commonly use CRISP-DM. This was followed by Scrum, Kanban and "My Own". **See results below.**



Categorical variables are handled appropriately for machine learning models (if models are created). Missing values are also handled appropriately for both descriptive and ML techniques. Document why a particular approach was used, and why it was appropriate for a particular situation.

✅ Categorical variables and Missing values are handled appropriately for machine learning models.

Good job on properly handling the categorical and missing values.

Certain sklearn estimators do not work if the data contains missing values since their mathematical operations demand the data to have all values in numeric format. For instance, if you directly try to fit the Standscaler object without handling missing values then you will face the following error:

```
ValueError: Input contains NaN, infinity or a value too large for dtype('float64')
```

So it becomes necessary to handle the missing values before any fitting operation.

✅ Document why a particular approach was used, and why it was appropriate for a particular situation.

Great job on documenting the approach you have taken to handle the missing values and categorical values.

In a real case scenario, this is a **very critical operation** and you can be questioned on the approach you have taken.

Useful Resources:

- Simple Methods to deal with Categorical Variables in Predictive Modeling [article]
- Missing Values in Data - Statistics Solutions [article]
- How to Handle Missing Data -- Towards Data Science [article]

Analysis, Modeling, Visualization

In the Jupyter Notebook, there are between 3-5 questions asked, related to the business or real-world context of the data. Each question is answered with appropriate visualization, table, or statistic.

✅ There are between 3-5 questions asked, related to the business or real-world context of the data

Excellent job! You have come up with some fascinating business contextual questions.

Your questions are easy to comprehend. It is really important to grab attention of the stakeholders since our objective is to help stakeholders in decision making and help them grow their business verticals.

✅ Each question is answered with appropriate visualization, table, or statistic.

You have come up with really nice hypothesis around the analysis. 💯

Excellent work! You took the support of several visualizations as well as the statistical metrics to arrive at the conclusion.

Useful Resources

Data Visualization using Python for Machine Learning and Data science [article]

Data Analysis, Visualization, and Interpretation [Video]

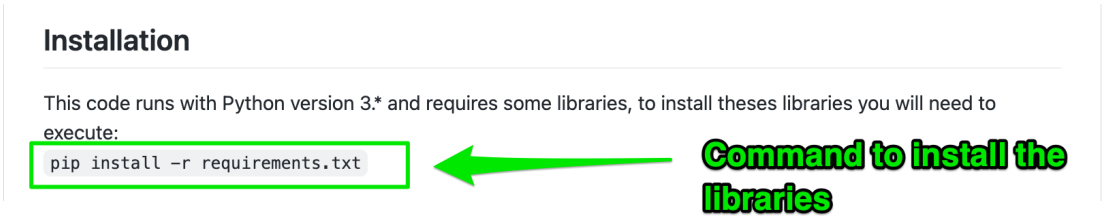
Github Repository

Student must have a Github repository of their project. The repository must have a README.md file that communicates the libraries used, the motivation for the project, the files in the repository with a small description of each, a summary of the results of the analysis, and necessary acknowledgements. Students should not use another student's code to complete the project, but they may use other references on the web including StackOverflow and Kaggle to complete the project.

Awesome! You have added all the following sections successfully in your readme file. 💯

✅ Communicates the libraries used

You can also create a `requirements.txt` file and mention how visitors can install libraries from it. Here is an example:



Useful Resources:

Why and How to make a Requirements.txt [Article]

✅ Motivation for the project
✅ Files in the repository
✅ Summary of the results
✅ Acknowledgments

Your README looks great!

Your `Table of Contents` clearly lays out the installation, project motivation, file descriptions, results, and acknowledgments. This will ease the process of navigating to the correct section.

Useful References:

Manage your data science project structure in early stage [Article]

How to write a good readme for your github project? [Article]

Blog Post

Student must have a blog post on a platform of their own choice (can be on their website, a Medium post or Github blog post). The post should not dive into technical details or difficulties of the analysis - this should be saved for Github. The post should be understandable for non-technical people from many fields.

✅ Student must communicate their results clearly

Awesome work on the blog post.

I really liked how you structured your blog post. Your post clearly communicates findings to non-technical stakeholders. You did a good job of providing a clear structure from the intro, the questions of interest, and a conclusion.

✅ The post should not dive into technical details or difficulties of the analysis. The post should be understandable for non-technical people from many fields.

Your post clearly communicates findings in a layman's manner. 👍

This way all the non-technical audience or stakeholders would be able to understand your insights and would be able to interpret the visualization and statistical metrics.

Other Remarks:

If it is hard to explain something without mentioning technical terms then hyperlink those keywords with appropriate articles or documentation. This way people who aren't aware of those words, can go through the documentation.

A blog post from the instructor [An Article by Instructor]

Art of Storytelling [Article]

Student must have a title and image to draw readers to their post.

✅ The blog post has a proper title

✅ The blog post has an image to draw readers to their post

Nice job with the title and image. 👍

Your title is concise and encompasses the overall context of the blog post. Great to see you using the subtitle section. That will surely help in providing context more clearly.

Apart from that, the attached image is spot on. A thumbnail like this will indeed grab the visitor's attention.

Useful posts:

How you write headlines [Article]

Suggestion:

Use the following sites to pull some good thumbnails.

https://pixabay.com/ https://stocksnap.io/

There are no long, ongoing blocks of text without line breaks or images for separation anywhere in the post.

✅ There are no long, ongoing blocks of text without line breaks or images for separation anywhere in the post.

You did a great job of making sure your post was broken into short paragraphs with clear ideas in each.

Staring at large amounts of texts clumped together without visuals and whitespace is exhausting for a reader.

Each question is clearly stated and each answer includes a clear visual, table, or statistic.

✅ Each question is answered with a clear visual, table, or statistic that provides how the data supports or disagrees with some hypothesis that could be formed by each question of interest

The discussion around the visualizations and table is really done well. 💯

Each question was followed with a visual of what provided a potential solution to the question.

DOWNLOAD PROJECT

RETURN TO PATH

Rate this review

START