UDACITY

‹ Return to Classroom

Write a Data Science Blog Post

REVIEW

CODE REVIEW

HISTORY

Requires Changes

5 specifications require changes

Dear Learner,

Well done, you have incorporated many requirements. You have made a good analysis to unravel insights from the data.

- Blog is very informative and covers the key insights from the analysis. You have structured the blog well and well summarized.

⚠ Following updates are required

- As part of introduction in Jupyter Notebook , please add business context. In addition, you can add the list of questions you have planned to analyze
- It is important to modularize the codes, you have not used any functions in the code. Please identify codes that are repeated and construct a function, so that you can reuse them
- 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.
- It is important to segment the codes based on CRISP DM framework.

Please check the rubric comments for more details, Looking forward to your next submission

All the best

Here are some additional resources that you may find helpful 📚

- PEP8 Style Guide
- Google's Python Style Guide
- The Hitchhiker's Guide to Python

Code Functionality and Readability

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

Good work. All codes in the project notebook executed without error and placed in the GitHub repo

Tasks Required
Project code is contained in a Jupyter notebook, which demonstrates successful execution and output of the code. 🟢

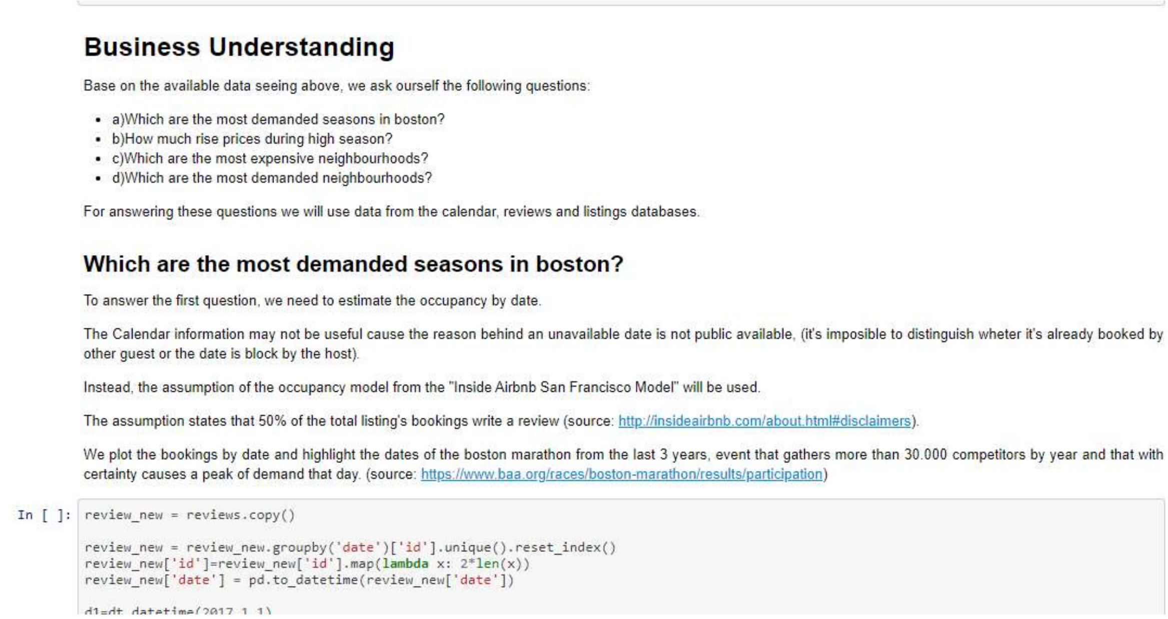
Code has easy-to-follow logical structure. The code uses comments effectively and/or Notebook Markdown cells correctly. The steps of the data science process (gather, assess, clean, analyze, model, visualize) are clearly identified with comments or Markdown cells, as well. The naming for variables and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

As part of introduction in Jupyter Notebook , please add business context. In addition, you can add the list of questions you have planned to analyze

- Adding an introduction to the analysis in the notebook is a good practice as part of documentation and will be helpful to the readers it can have
- Business Context of the analysis
- List of questions you are trying to answer with the analysis

---

Example



Business Understanding

Base on the available data seeing above, we ask ourself the following questions:

- a)Which are the most demanded seasons in boston?
- b)How much rise prices during high season?
- c)Which are the most expensive neighbourhoods?
- d)Which are the most demanded neighbourhoods?

For answering these questions we will use data from the calendar, reviews and listings databases.

Which are the most demanded seasons in boston?

To answer the first question, we need to estimate the occupancy by date.

The Calendar information may not be useful cause the reason behind an unavailable date is not public available, (it's imposible to distinguish wheter it's already booked by other guest or the date is block by the host).

Instead, the assumption of the occupancy model from the "Inside Airbnb San Francisco Model" will be used.

The assumption states that 50% of the total listing's bookings write a review (source: http://insideairbnb.com/about.html#disclaimers).

We plot the bookings by date and highlight the dates of the boston marathon from the last 3 years, event that gathers more than 30.000 competitors by year and that with certainty causes a peak of demand that day. (source: https://www.baa.org/races/boston-marathon/results/participation)

```
In [ ]: review_new = reviews.copy()

        review_new = review_new.groupby('date')['id'].unique().reset_index()
        review_new['id']=review_new['id'].map(lambda x: 2*len(x))
        review_new['date'] = pd.to_datetime(review_new['date'])

        d1=dt.datetime(2017,1,1)
```

Best practices for code structure and naming:

- Use a logical structure. Your code should be easy to follow and understand. This means using clear indentation, breaking up your code into logical blocks, and using meaningful variable names.
- Use comments effectively. Comments can help explain what your code is doing and make it easier to understand. However, don't overuse comments, or they will become cluttered and unhelpful.
- Use Markdown cells in notebooks. Markdown cells are a great way to add documentation and explanations to your notebooks. This can be helpful for explaining the data science process, as well as providing more detailed - - information about your code.
- Follow the PEP8 style guide. The PEP8 style guide is a set of guidelines for writing Python code. Following these guidelines will help make your code more readable and consistent.

Here are some additional resources that you may find helpful 📚

- PEP8 Style Guide
- Google's Python Style Guide
- The Hitchhiker's Guide to Python

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

It is important to modularize the codes, you have not used any functions in the code. Please identify codes that are repeated and construct a function, so that you can reuse them

Tasks Required

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

💡Modularizing code in Python is important for several reasons, including:

- Improved readability: Modular code is easier to read and understand because it is broken down into smaller, more manageable chunks. This makes it easier to find and fix errors, and to understand how the code works as a whole.
- Increased reusability: Modular code can be reused in other projects, which can save time and effort. This is because modular code is self-contained and does not rely on any external dependencies.
- Enhanced maintainability: Modular code is easier to maintain because it is easier to update and debug. This is because changes to modular code can be made in isolation, without affecting the rest of the code.
- Improved testability: Modular code is easier to test because it can be tested in isolation. This is because modular code is self-contained and does not rely on any external dependencies.

Further Reading 📚

- Understanding Code Reuse and Modularity in Python
  Use descriptive function names. This will help you and others understand what the function does.
  Use parameters to pass data into functions. This will make your functions more reusable.
  Return values from functions. This will make your functions more modular.

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.

It is important to segment the codes based on CRISP DM framework. Good work, you have added few sections, however please structure the code based on below

Identify the segments that are applicable to the analysis and segment your codes

- Business Understanding
- Data Understanding
- Prepare Data
- Data Modeling
- Evaluate the Results

Use markdowns to add these sections in the code

💡Tips for using CRISP-DM:

- Be flexible. The CRISP-DM process is not a rigid set of steps. You may need to adjust the process depending on the specific data mining project.
- Iterate. It is often helpful to iterate through the CRISP-DM phases multiple times. This will help you improve the quality of the data, the models, and the results.
  Get help. There are many resources available to help you with data mining. If you are struggling with a particular aspect of the process, don't be afraid to ask for help.

Further Reading 📚

- The Data Science Process (CRISP-DM) with Python

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.

Good work. You have documented the steps in the analysis. As part of documentations , we need to cover following aspects for a Data Science based analysis

Tasks

- Please highlight how you have handled missing values, if missing is not present provide a note on it
- 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.
- If Dropna is used, add a note on why this method is used instead of other methods. Add a markdown for it

Further Reading 📚

- Powering Documentation with Jupyter Notebooks

💡Here are some examples of how you can use Markdown in Jupyter Notebooks:

- Adding headings and subheadings: You can use Markdown to add headings and subheadings to your notebooks. This can help to organize your content and make it easier to read.
- Adding lists: You can use Markdown to add lists to your notebooks. This can be helpful for organizing your code or for creating a to-do list.
- Adding links: You can use Markdown to add links to your notebooks. This can be helpful for linking to external resources or for linking to other cells in your notebook.
- Adding images: You can use Markdown to add images to your notebooks. This can be helpful for adding visual aids to your notebooks or for adding screenshots of your code.
- Adding code blocks: You can use Markdown to add code blocks to your notebooks. This can be helpful for showing your code or for running your code.

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.

Relevant questions/Objective added and segmented the analysis based on the questions. Well done

Tasks Required
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. 🟢

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.

GitHub repo is well documented. You have provided all the required details in the readme file. Good work

💡Best practices for writing a GitHub README documentation:

- Start with a brief overview of the project. What does the project do? Who is it for? What are the benefits of using the project?
- Provide installation instructions. How do users install the project? What are the system requirements?
- Include a usage guide. How do users use the project? What are the different features and options?
- Provide a contributing guide. How can users contribute to the project? What are the different ways to contribute?
- Keep your README up-to-date. As you make changes to your project, make sure to update your README documentation. This will help ensure that users have the latest information about your project.

Further Reading 📚

- How to Write a Good README File for Your GitHub Project

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.

Good work. Blog is very informative and covers the key insights from the analysis. You have structured the blog well and well summarized.

- Insights align with the analysis provided in Notebook. Well done
- Blog is non-technical and easy to read and understand the insights

Further Reading 📚

- Data Science Blogs You Need To Check Out

Steps in writing data science blog posts: 💡

- Choose a niche. What are you passionate about in data science? What do you know a lot about? Once you know your niche, you can focus your blog posts on topics that your audience will be interested in.
- Write about your own experiences. People love to read about real-world applications of data science. If you've worked on any interesting projects, share your experiences with your readers.
- Keep your posts concise and informative. People are busy, so they don't have time to read long, rambling blog posts. Get to the point quickly and clearly.
- Use visuals to illustrate your points. Data visualization is a powerful way to communicate your findings to your readers. Use charts, graphs, and other visuals to make your blog posts more engaging and informative.
- Proofread your work carefully. Typos and grammatical errors will make your blog posts look unprofessional. Take the time to proofread your work before you hit publish.
  Promote your blog posts. Once you've written a great blog post, don't forget to promote it! Share it on social media, submit it to relevant websites, and reach out to other data scientists in your field.

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

Please add a welcome image for the blog

There are several reasons why you should add a welcome image to your blog: 💡

- It makes your blog more visually appealing. A well-chosen image can help to break up the text on your blog and make it more visually interesting. This can help to improve your blog's overall appearance and make it more likely that visitors will stick around to read your content.
- It can help to improve your blog's SEO. Images can help to improve your blog's search engine ranking by providing search engines with additional information about your content. This is because images can be indexed by search engines just like text.
- It can help to improve your blog's click-through rate. When you include an image in your blog's title or featured image, it can help to improve your blog's click-through rate (CTR). This is because images can help to make your blog posts more visually appealing and stand out in search results.

  It can help to improve your blog's engagement. Images can help to improve your blog's engagement by encouraging visitors to comment, share, and like your content. This is because images can help to make your content more visually appealing and engaging.

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

No long blocks found concise and well structured

💡 Why it is important to have a concise blog instead of a verbose one.

- Concise blogs are easier to read and understand. When you write a concise blog, you are getting to the point quickly and clearly. This makes your blog easier to read and understand, especially for busy readers who don't have a lot of time to spare.
- Concise blogs are more likely to be shared. When people share a blog post, they are essentially giving it a vote of confidence. They are saying that the blog post is worth their time and that others should read it too. If your blog post is long and rambling, people are less likely to share it.
- Concise blogs are more likely to rank well in search engines. Search engines like Google rank websites and blog posts based on a number of factors, including the length of the content. In general, shorter content is more likely to rank well in search engines.

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

Good use of plots and visualizations to substantiate the analysis insights in the blog.

- Insights aligns with the visualizations
- Well annotated plots

💡Best practices to substantiate analysis with clear visuals, tables, or statistic

- Use visuals that are relevant to your analysis. The visuals you choose should be clear, concise, and easy to understand. They should also be relevant to the data you are analyzing.
- Use tables and statistics to summarize your data. Tables and statistics can be a great way to present large amounts of data in a clear and concise way. They can also be used to compare different groups or to identify trends.
- Label all visuals and tables. Make sure to label all visuals and tables with clear and concise titles. This will help readers understand what the visuals and tables are showing.
- Interpret your visuals and tables. Don't just present your visuals and tables without explaining what they mean. Take the time to interpret your data and explain what it tells you.

Further Reading📚

- Visualize Data - A good read on Data Visualization techniques
  Data visualization is the process of representing data graphically in order to identify trends and patterns that would otherwise be unclear or difficult to discern.

☑ RESUBMIT

DOWNLOAD PROJECT

Learn the best practices for revising and resubmitting your project.

RETURN TO PATH

Rate this review

START