Code Review

Name: Nadia Baptista
Project: EDA Finance
GitHub:
https://github.com/Nads211/exploratory-data-analysis---customer-loans-in-finance489

Overall Impression

Overall a great job Nadia! You've got some good solutions for several of the tasks and have generally handled them well. The notebooks are structured well and detail your decisions and the basis for them very well done. Mainly a few points about Milestone 4 to consider.

File Structure

Good work on the file structure! Really good to see that each class has its own .py file and the use of notebook .ipynb files, although their names can be changed to something like "cleaning.ipynb" and "analysis.ipynb" to make their purpose clearer as the names are quite similar.

Another addition would be the presence of a requirements.txt or environment.yaml so users can clone your exact conda environment. You can also put the csv's into a "data" folder.

Something that needs to be removed is the credentials.yaml. Don't upload your credentials to github!

Documentation

README

Nicework. It's clear and concise with good structure and description. Try adding a table of contents, and mention git clone in the installation instructions. Look at Milestone 4 task 6 for an idea on how to expand the structure.

Docstrings & In-line Comments

Good job with these. The docstrings are well formatted and are present in every function apart from the db_utils.py file. Replace those comments with docstrings for the functions.

The comments are used well in the .ipynb files letting the markdown cells of the notebook explain the majority of the discussion and code.

Code

General code

Import statements should be organised in a consistent manner, with the preferred method being to keep the 'from' imports together, and otherwise arrange them in ascending alphabetical order.

In .py files blank lines there should be 2 blank lines between a class and imports. You can find out more here: https://peps.python.org/pep-0008/#blank-lines

Fantastic work on the classes and functions! The aim was to have reusable code for future projects so the fact there's nothing specific to the loans dataframe inside of the functions pertaining to milestone 3 on cleaning is great.

exploratory_data_analysis.ipynb

When imputing the nulls inside cols instead of repeating the code for different cols, put them in a list and loop through them, applying the function.

Fantastic work on this milestone! The visualisations and rationale behind null handling, skew transformations, outliers and collinearity were on point! Highlights were the skew transformation comparative Histograms to visualise how well each skew transform performs and the summary filter_column_info_dataframe cells.

analysis_and_visualisation.ipynb

I noticed the file you use seems to be the fully cleaned file from Milestone 3. Milestone 4 should have all the columns in the correct format and nulls handled but nothing else as we are trying to query and analyse the data so need it in the untransformed format. An example is if you skew transform the cols then the total_amount column won't have the actual amount for the loan but a reduced, transformed amount which is not the actual data.

Milestone 3 was to show that you can do all these preparation steps that may be expected of you in a job which included the skew transforms and collinearity dropping cols for ML models, but Milestone 4 is an analysis of the data so doesn't need those exact steps.

Task 1:

What you have isn't quite what the task was going for but we found that the current task 1 question is more complicated than anticipated so it will be changed soon (read the discord post regarding this task in the general chat for more info). If you want to attempt the original task I have laid out a long post of what to do. Otherwise all you have to do now is summarise what percentage of the loans are paid back compared to the total payment expected with interest, and get a prediction for 6 months into the future.

To give you a hint for the prediction that appears to be missing you need to take into account that not every loan will have 6 months left in their term, so you need to calculate the amount of months remaining (make a new col) and take that into account when making the 6 months prediction.

Task 2:

Nice work here! Try to use .isin(“Charged Off”) or .str.contains(“Charged Off”) when filtering as there is another possible value in the loan_status column of “Does not meet the credit policy. Status:Charged Off”, which you should take into consideration.

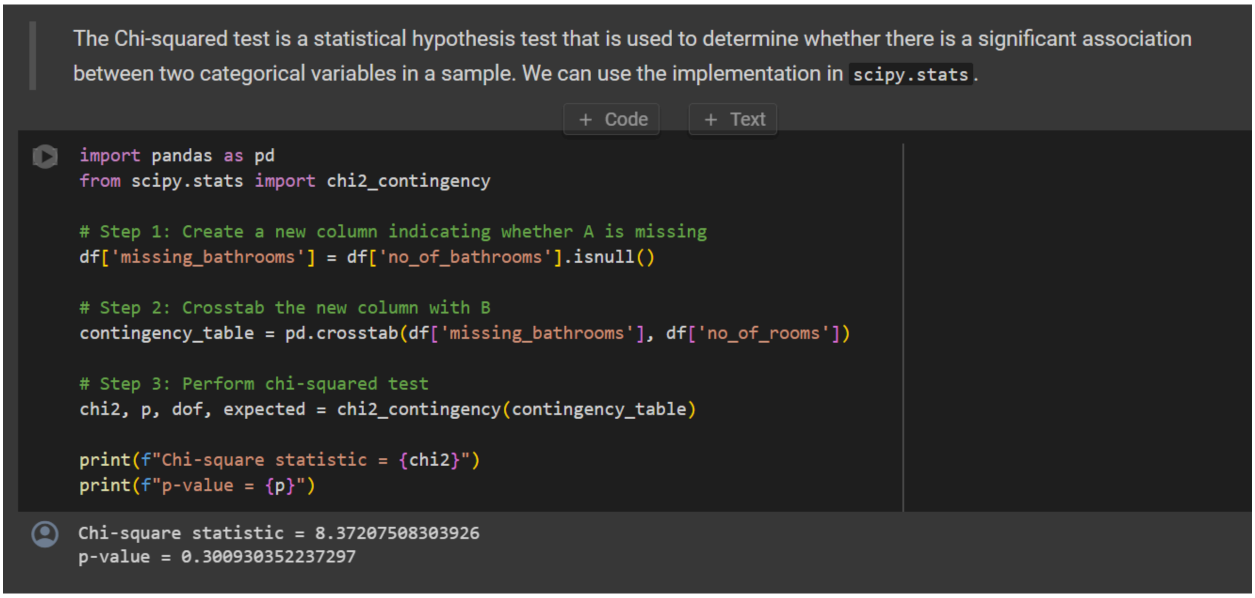
Task 3 & Task 4:

Good job. The visualisations are really clear well done!

Task 5:

Don’t see the code for this task. Its purpose was to see if there was any significance in the categorical columns for indicators of loss. So each group of customers should have their own df like for late, charged-off, on time, etc. Then the loan_status column should be encoded so it's possible to visualise it with the continuous columns in a correlation matrix to see if there's any highly correlated columns. If not that's fine but you need to show and explain that that is the case.

Also to follow the suggestions in the task, it states with the categorical cols grade, purpose and home_ownership with different values in “loan_status” you need to prove there is some significance using the chi squared test. You can see an example of this in the “End-to-End EDA” notebook here:



The Chi-squared test is a statistical hypothesis test that is used to determine whether there is a significant association between two categorical variables in a sample. We can use the implementation in `scipy.stats`.

```
import pandas as pd
from scipy.stats import chi2_contingency

# Step 1: Create a new column indicating whether A is missing
df['missing_bathrooms'] = df['no_of_bathrooms'].isnull()

# Step 2: Crosstab the new column with B
contingency_table = pd.crosstab(df['missing_bathrooms'], df['no_of_rooms'])

# Step 3: Perform chi-squared test
chi2, p, dof, expected = chi2_contingency(contingency_table)

print(f"Chi-square statistic = {chi2}")
print(f"p-value = {p}")
```

```
Chi-square statistic = 8.37207508303926
p-value = 0.300930352237297
```

This should be done between the categorical columns for each subset of loan_status to see if there is any correlation between them. So if the p-value is less than 0.05 it’s statistically significant and there is a relationship there to perhaps warrant further study (by the ‘company’ not yourself).

Overall a great project! Structure and style wise it's good and the docstrings are descriptive. It's clear how much thought you put into Milestone 3 as the notebook is exemplary. Only a couple of points about Milestone 4 really and the missing Task 5, everything else is spot on. Awesome job here Nadia!