FIT5196 Data wrangling S2 2017

Assessment 2: Parsing and Cleansing Raw Data

Please carefully review all the requirements below to ensure you have a good understanding of what is required for your assessment.

1. **Due Date**
2. **Instructions & Brief**
3. **Assessment Resources**
4. **Assessment Criteria**
    1. **Grading Rubric**
    2. **Penalties**
5. **How to Submit**

---

1. Due Date

This specific due date and time can be **viewed below in the Grading Summary**.

---

2. Assessment description

The real estate markets, like those in Sydney and Melbourne, present an interesting opportunity for data analysts to analyze and predict where property prices are moving towards. Prediction of property prices is becoming increasingly important and beneficial. Property prices are a good indicator of both the overall market condition and the economic health of a country. This assessment assumes that you, as a data analyst, are required to wrangle a large set of property sales records stored in an unknown format and with unknown data quality issues. This assessment contains two major tasks that are specified as follows:

Task 1. Parsing the property sales data stored in “data.dat”:

- Examine and load the data into a Pandas DataFrame.
- Parse the loaded data so that each sales record has the following attributes:

Attribute	Description
date	Date of the property sold, e.g., 20140502T000000
price	Property sold price
bedrooms	Number of bedrooms
bathrooms	Number of bathrooms, the value of which can be either an integer or a fraction ending with .25, . and .75. For example, 0.5 accounts for a room with a toilet but no shower

sqft_living	Square footage of the property's interior living space, it is equal to the sum of basement area (i.e sqft_basement) and the above living area (i.e., sqft_above)
sqft_lot	Square footage of the land space
floors	Number of floors
waterfront	Whether the property was overlooking the waterfront or not
view	An index from 0 to 4 of how good the view of the property was
condition	An index from 1 to 5 on the condition of the property.
sqft_above	The square footage of the interior living space that is above ground level
sqft_basement	The square footage of the interior living space that is below ground level
yr_built	The year the property was initially built
yr_renovated	The year of the property's last renovation
street	The street address of the property, e.g., “3140 Franklin Ave E”
city	The city where the property is, e.g., “Seattle”
statezip	The zip code area where the property is, which contains state and zip code, separated by a spac For example, “WA 98115”, where WA is the abbreviation of Washington and the number is the zi code.
country	The country where the property is, e.g., “USA”

After the data is parsed and loaded into Pandas, you should have a DataFrame where each row is a sales record and each column is one of the attributes listed in the table above. All the columns should be labeled with attribute names shown in the table (Note: it is not allowed to change the attribute names!) and also have a proper data type (e.g., integer, string, float, date, etc). Now, the loaded data should be ready for Task 2.

Task 2. Auditing and cleansing the loaded data: In this task, you are required to inspect and audit the data to identify the data problems, and then fix the problems. Different generic and major data problems could be found in the data might include:

- Lexical errors, e.g., typos and spelling mistakes
- Irregularities, e.g., abnormal data values and data formats
- Violations of the Integrity constraint.
- Outliers
- Duplications
- Missing values
- Inconsistency, e.g., inhomogeneity in values and types in representing the same data

Hint: You might need to use non-graphical (e.g., statistics) and graphical (e.g., different plots) methods to explore the data in order to identify those problems. You might also need to refer to the table above for the description of the attributes.

In order to finish this assessment, you should **write Python code (Hint:** use existing Python packages as possible as you can**) in a Jupyter Notebook** with proper comments. Your notebook will take "data.dat" as input and generate a CSV file containing the cleaned data. It is important to structure your Jupyter notebook in such a way that it clearly shows how you identify and solve each problem found in the data. Some solutions could be subjective, e.g., when dealing with missing values and outliers, different people might choose different methods, however, you will have to justify your choice of methods using markdown cells.

This is an individual assignment and worth 30% of your total mark for FIT5196.

3. Assessment Resources

Before you start writing your code, you will need to read the following materials:

- While using Jupyter Notebook, you should consider the use of different cells to make notes as you go. However, be precise and concise, please do not include things that are not relevant to the tasks in your notebook.
- Standards for commenting code are available online (e.g. **https://www.python.org/dev/peps/pep-0008/#comments).** You must ensure that you can clearly explain what your code does with clear comments.
- The work required to finish this assessment should be your own. If you use resources elsewhere, make sure that you properly acknowledge them in your notebook.You may need to review the FIT citation style tutorial to make you're familiar with appropriate citing and referencing for this assessment. Also, review the demystifying citing and referencing for help.

and download the data file:

- data.dat

---

4. Assessment Criteria

The following outlines the criteria which you will be assessed against.

4.1 Grading Rubric

Specific grading details for this assessment are:

- Programming and results: (80%)
  - The submitted code should work without any errors and produces the required results correctly.
  - All the errors should be identified and resolved.
- Quality of code, commenting and notebook: (20%)
  - The code should be well structured and properly commented. A high-quality code is clean, explicit, consistent, efficient, easy to read and **well use of existing libraries**
  - The methods used in each task should be well justified in the Jupyter notebook.
  - The notebook should be structured in a logical way so that it clearly shows how students finish the tasks in the assessment.

Detailed rubric can be found here

4.2 Penalties

- Late submission: for all assessment items handed in after the official due date, and without an approved extension, a 5% penalty applies to the student's mark for each day after the due date (including weekends, and public holidays) for up to 10 days. **Assessment items handed in after 10 days will not be considered!**
- Submission: please do follow **Section 5 How to Submit** to submit your assignment. Otherwise, a 5% penalty will be applied.

---

5. How to Submit

Once you have completed your work, take the following steps to submit your work.

1. Only two files to be submitted:
    1. A **Jupyter Notebook file**, whose extension must be ".ipynb". The file name must be in the following format: **surname_studentID_ass2.ipynb.**
    2. The **CSV file** generated by your jupyter notebook, which contains the cleaned data. The file name should be in the following format: **surname_studentID_ass2_data.csv**

3. Two files should be uploaded individually, rather than in a zip file.

2. Click the **Add Submission** button below to submit and upload your assignment. **Please do remember to accept the submission statement! Only the submitted assignments will be marked. Those shown as a draft won't be marked!**

If you need further guidance on how to submit an assessment item, please review the **Submitting an assessment** overview. If you need further assistance, please go to **Help & Support**.

Submission status

Attempt number	This is attempt 1.
Submission status	No attempt
Grading status	Not graded
Due date	Sunday, 24 September 2017, 1:55 AM
Time remaining	19 days 11 hours
Last modified	-
Submission comments	Comments (0)

Add submission

Make changes to your submission

NAVIGATION

Dashboard
- Site home

Site pages

Current unit

FIT5196-S2-2017

Participants

Unit Updates

Unit Information

Consultation Times

Forums

My Assessments

Academic Integrity Tutorials

Student Academic Integrity Policy

Assessment 1: Text Preprocessing

Assessment 2: Parsing and Cleansing Raw Data

Assessment 1 Patent Data

Assessment 2 Raw Data

How to submit an assignment in Moodle

Notebook template for assessment 1

Assessment 2 Rubric

Jump to Content

My.Monash

IT Academic Integrity

IT Student Portal

ADMINISTRATION

Unit administration