Statistic for AI and Data Science

Coursework 3, Version 1

1   Introduction

This document outlines the requirements for the coursework. Section 2 introduces the data and section 3 gives the analysis and submission requirements. Programming hints are in Section 5.

2   The Data on English Property Prices

The file `average-property-price.csv` has information about the average price of properties (there are 4 types of properties) over 36 months in different areas of the country. The dataset contains the following fields:

Name	Description
Date	A date, which is the first of the month, between September 1st 2016 and August 1st 2019. 36 months in total.
Area	The name of an area (or region – see below)
Code	The code for the area (or region – see below)
Detached	Average sale price of a detached property in this area in the month
Semi	Same, for semi-detached property.
Terraced	Same, for a terraced property
Flat	Same, for a flat.

Understanding the Region / Areas

The Area Codes are defined by the Office for National Statistics (ONS) and reflect the hierarchy of government in the UK. We have simplified the data so it is not necessary to understand the full ONS codes, but if you want to understand the full details read https://en.wikipedia.org/wiki/ONS_coding_system.

In the provided data:

- Each area of the country is covered twice, at different levels.
- Level 1 is a <u>region</u>. There are 9 regions in England (examples are 'London', 'South East', 'South West').
- Level 2 is a local government <u>area</u>. There are 4 different types of local government areas, though the distinction is not important in our analysis. Each area belongs to one of the regions.

The following table shows the prefixes of the Area Code and the corresponding level.

Code Prefix	Classification	Level	Description
E12	English Region	1	One of 9 different English regions
E10	County	2	Local government area. Parts of a region.
E09	London Borough	2	
E08	Metropolitan Boroughs	2	
E06	English unitary authority	2	

Relationship Between Area and Region

A separate file `location-codes.csv` shows which region each area belongs to. Note that here is some overlap between the two files, such as the names of the areas and regions. This file can be used to determine which of the 9 regions each area is in.

3   Required Analysis

Complete the analyses and submit your work as described in the following table.

Requirement	Marks
1. Submit the work in a well-presented Jupyter notebook	
a. The notebook must have a title, table of contents, numbered sections and subsections. It must make appropriate use of markdown to describe how the data is being manipulated. This commentary must be readable (by a property expert) with the code cells hidden.	20
b. Code runs without error when the kernel is re-initialised and all the code is executed in order. The code should be clear and it should be easy to see that it is correct.	
2. Load and prepare the data	
a. Separate the region data (with 'E12' prefix codes) from the area data (other prefixes). *Both region data and area data are needed below.*	
b. Add to each area price record the region to which the area belongs, ensuring that all areas have been assigned a region.	15
c. Check the data for missing values in the region or area data; decide how to act. giving a brief justification.	
3. Plot trends of the prices, considering only flats and detached properties. *Use the region data in this section.*	
a. Look at the trends in prices in the 9 regions over the 36 months, making appropriate choices of the number of plots to use to show trends clearly.	20
b. Comment on the trends. Compare the regions, the two property types and different times of year.	
4. Calculate the change in prices between July 2017 and July 2018, looking only at flats. *Use the area data in this section*.	
a. Calculate the change in the price in each area.	
b. Choose a way (or ways) to visualise the change in area prices in each region.	25
c. Comment on the results, describing any patterns that you see.	
5. Investigate whether there is evidence that the change in the price of flats has affected all the regions similarly. *Use the area data in this section*	
a. Cross-tabulate the number of areas in which the price has increased and the number has decreased, by region	
b. Use a chi-square test to determine whether there is evidence that the regions differ	20
c. Interpret the results you obtain, including what you can and cannot determine from the result.	

4   Feedback

No feedback is provided until after the last hand-in date (i.e. one week after the deadline). The following forms of feedback will be provided:

1. A sample answer.
2. General review comments on the main strengths and weaknesses for the class as a whole (written or in a review lecture)
3. The grades obtained by your answer on each of the detailed criteria. Clarifying comments may be added.

5   Resources and Pandas Programming Hints

This section is here to help. There are no extra requirements added in this section, so ignore it if you wish.

5.1   Reading Dates with Pandas

The Pandas library supports dates, which are represented as the pandas by the `datetime` type. Date are complex, with many different formats; our data uses the UK 'day first' format, writing 01/07/2017 for the first of July 2017.

Here is the code for reading the date correctly. The data is first read without parsing the date (which is just held as a string). The date is then parsed and inserted as a new column, before the old column is dropped.

```
prices = pd.read_csv('average-property-price.csv')
prices = prices.assign(Month = pd.to_datetime(
    prices['Date'], dayfirst=True)).drop(labels='Date', axis=1)
```

Note that the date is used as a data value, not as an index. Consider why this is.

5.2   Adding a Region to Each Area

Requirement 2 b) asks you to add the region to each area in the area price data. For example, the area 'Waltham Forest' with area code E09000031. One of the rows in the location-codes.csv file has:

	AuthorityCode	AuthorityName	RegionCode	RegionName
124	E09000031	Waltham Forest	E12000007	London

So we can see that 'Waltham Forest' is in the 'London' region.

There are many ways to add the region name to each row of the area data. Here are some:

1. Merge the tables. We did not cover this in the course, so you need to read the documentation at: https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.merge.html
2. Use a dictionary mapping from area code to the name of the region. Use the dictionary in a function that is applied (function 'apply') to each row, assigning (function 'assign') a new column.
3. Apply a function to look-up the region name for an area code directly in a data frame of location codes. Again, the functions are 'apply' and 'assign'.

There are in turn may ways to create a dictionary. A data frame with two columns can be turned into a dictionary using the 'to_dict' function; alternatively, apply a function to each row of a location code data frame, adding entries to a dictionary.

5.3 Creating the Cross Tabulation

The cross tabulation table has the following general shape (note that it will not look exactly like this in Pandas):

		Region			
		Region1	Region2	Region3	Region4
Change	Increase				
	Decrease				

Recall that the core ideas of a cross-tabulation are that

- The values of some variables are combined (counting is the default combinator)
- The values of other variables become headers (on either axis)

To obtain the table above, we need to start from a dataframe that includes columns in the following general form:

Area	Region	Change
A1	Region1	Increase
A2	Region3	Decrease
A3	Region2	Decrease
A4	Region2	Increase
A5	Region1	Increase
A6	Region3	Decrease

The cross-tabulation is created by combining the area values and using the region and changes values as headings.

5.4 Calculating the Change in Price

In the provided data the prices for different dates are in different rows. The general form of the data is shown below.

Area	Month	Flat
A1	01/07/2017	200,000
A2	01/07/2017	300,000
A3	01/07/2017	400,000
A1	01/07/2018	220,000
A2	01/07/2018	330,000
A3	01/07/2018	440,000

One route towards calculating the price different is to reorganise this into the following table:

Area	Flat2017	Flat2018
A1	200,000	220,000
A2	300,000	330,000
A3	400,000	440,000

One route towards this is first to split the table by date:

2017 data frame   2018 data frame

Area	Month	Flat
A1	01/07/2017	200,000
A2	01/07/2017	300,000
A3	01/07/2017	400,000

Area	Month	Flat
A1	01/07/2018	220,000
A2	01/07/2018	330,000
A3	01/07/2018	440,000

In these data frames, there is only a single row for each area so this can be used to retrieve a value. this can be done using assign / apply; another approach (not covered) is to make the area an index (set_index method) and then use the merge method.

5.5   Chi-Squared Test

See the sample notebook.