Where should I live?
An analysis of areas in major UK cities

Applied Data Science Capstone Project – The Battle of Neighbourhoods

Introduction

It is well known that it is becoming increasingly difficult to afford a house in the UK.[1] This is particularly noticeable in southern cities such as Oxford (my current hometown), where the cost of living has been driven up significantly above the national average by a number of factors.[2] Part of the reason that these places are desirable are the easy access to venues such as museums, cafes and restaurants, and nature reserves.

Problem:

Where could someone move that would offer a similar lifestyle to their current location?

In this project, we will try find similar areas across the UK based on access to venues, and then compare the cost of housing in these areas.

This will allow people who are unable to afford to buy a house in their current area to identify similar areas around the country that might be more affordable.

Target Audience:

Young people seeking to step onto the property ladder.

Data

In order to address this problem, I needed to:

1. Identify regions of interest
2. Find house prices in each of these regions
3. Find data on venues in each of these regions

Identify regions we are interested in

To limit the amount of data we are working with, I have chosen to look at the following cities:

- Birmingham
- Bristol
- Cardiff
- Leeds
- Liverpool
- London
- Manchester
- Milton Keynes
- Newcastle
- Oxford
- York

These have a wide spread across the UK while still having reasonably sized populations, and so should be reasonably comparable. Scottish cities have unfortunately been excluded as they are absent from our price dataset, as described below.

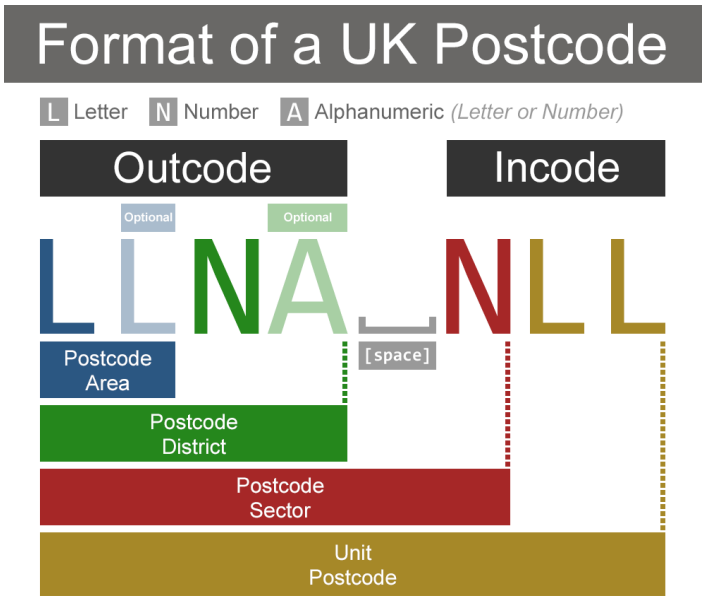
Find house prices in each of these regions

House price data were obtained from the UK Land registry,[3] using a SPARQL query that can be found in Appendix 1. While requests were attempted for Edinburgh and Glasgow, Scotland was not included in this dataset.

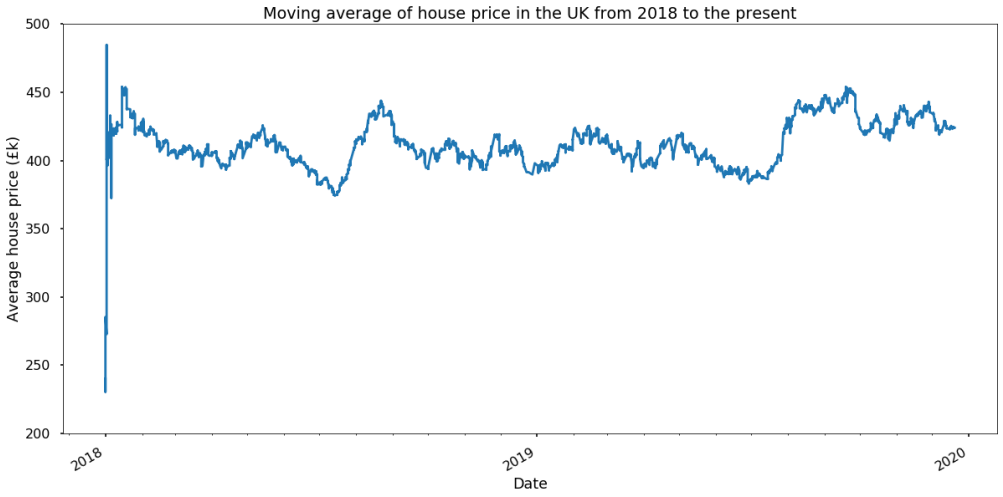
The parameters for the request were to find all house transactions from January 1 2018 to February 22 2020 in each city. Only freeholds were considered.

The separate data requests were combined in Excel and exported as a CSV file.

After tidying and cleaning the data, the postcode district of each transaction was calculated using a regular expression.[4]

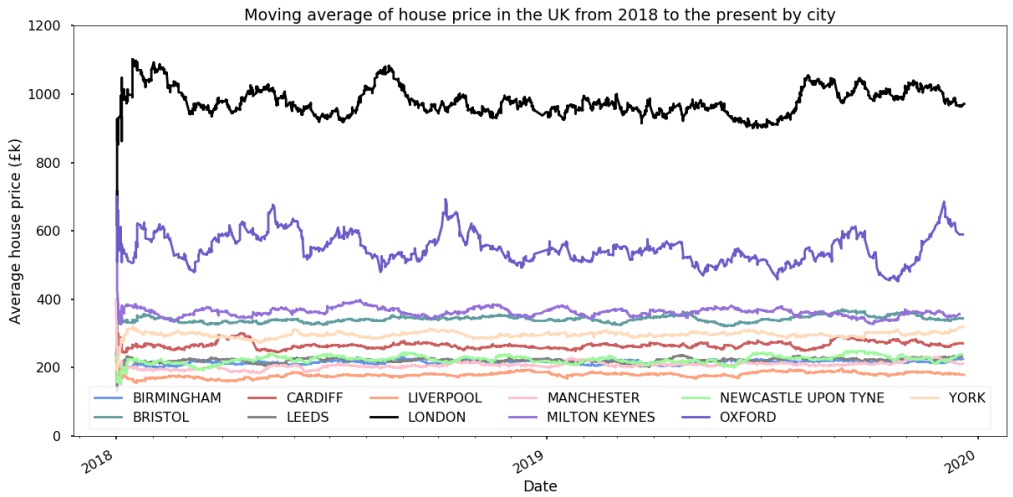


Then, the variance in house price over the time-period collected was examined using a 30-day moving average of the transactions.



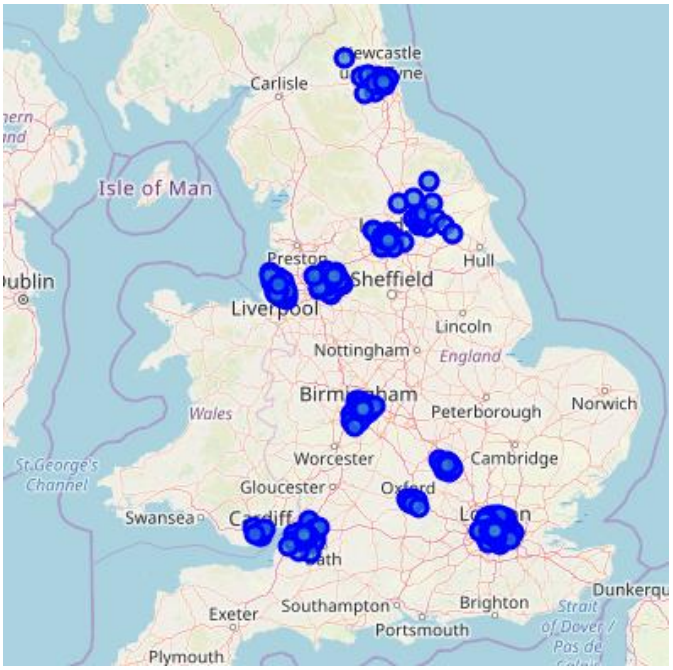
There was a spike around August/September of 2019 but otherwise the fluctuation seems to be small enough to allow for the analysis to use all the data. It should be noted that the large spike at Jan 2018 is an artefact due to the moving average operation.

Looking at the cities separately, we can see the large disparity between towns for average house prices.



Again we see that prices are relatively stable over this period. London is unsurprisingly the most expensive, with an average price hovering about £1,000,000, while places like Liverpool, Manchester and Newcastle hover around the £200,000 mark. Price variations are also more severe in the more expensive cities, although not enough to invalidate the use of the full dataset.

Latitude and longitude data for each postcode were derived from the CodePoint Open dataset.[5] As the positional data included were formatted as Eastings/Northings rather than latitude/longitude, these were converted and then averaged over district to provide a single set of positional coordinates for each outcode.



Find data on venues in each of these regions

Foursquare was used to identify the top 100 venues in a radius of 1 km from each outcode centre. This is not ideal, as the outcodes are of highly variable size. However, unfortunately the datasets containing the areas of the outcodes are not publicly available,[6] so this is a very rough approximation.

There was a high variance in number of venues in each postcode - this is an issue that could be dealt with using the true area of each outcode, however as we cannot do this we will continue with the rough approximation.

Methodology

The aim of this project was to create a tool to allow people living in more expensive postcodes to identify less expensive postcodes that offer a similar living experience (i.e. types of venues nearby).

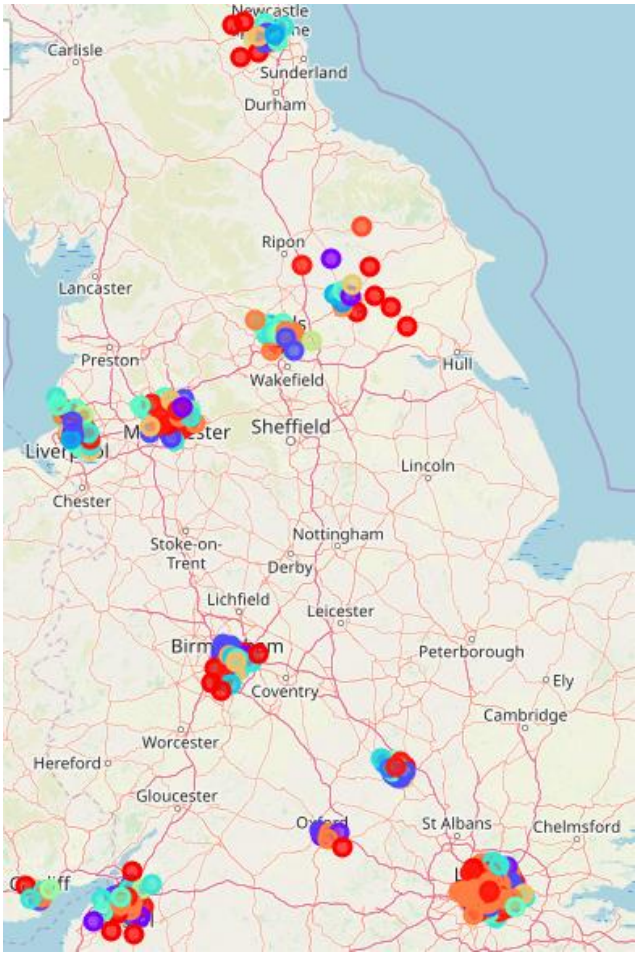
First of all, the price and location data required for the analysis was obtained, as were the types of venues within a 1 km radius of the centre of each area.

K-means clustering was then used to group the postcode districts based on the types of nearby venues.

Finally, a tool was created (using iPython widgets) that takes an outcode as an input and produces a graph detailing the cheapest areas in that outcode's cluster, and locates them on a map. This provides the user with a good starting point for exploration for looking for a property.

Results

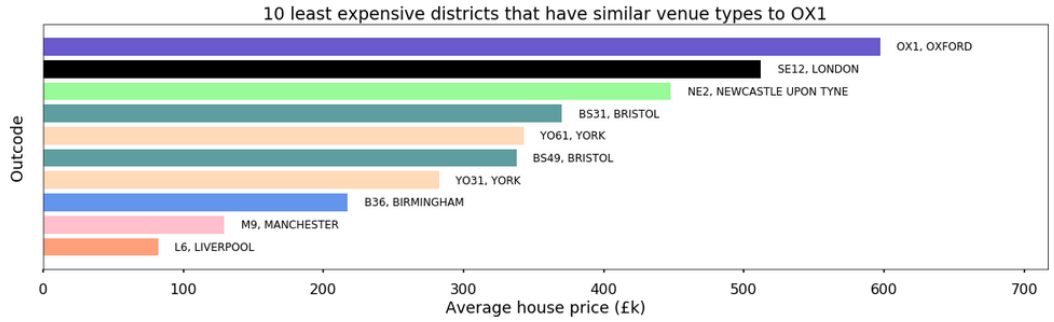
A silhouette score was used to determine the best value for K in the K-means clustering algorithm. In this instance of the code, 31 clusters were generated.



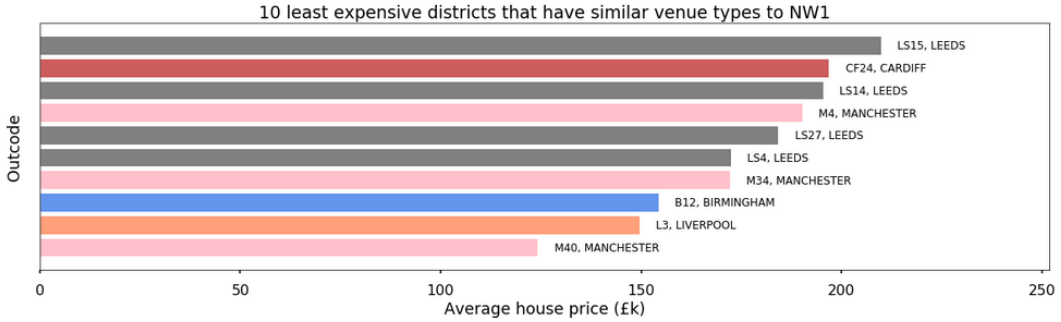
These clusters were broadly split between the cities, which bodes well for the results, as we don't want to be recommended districts within our starting city.

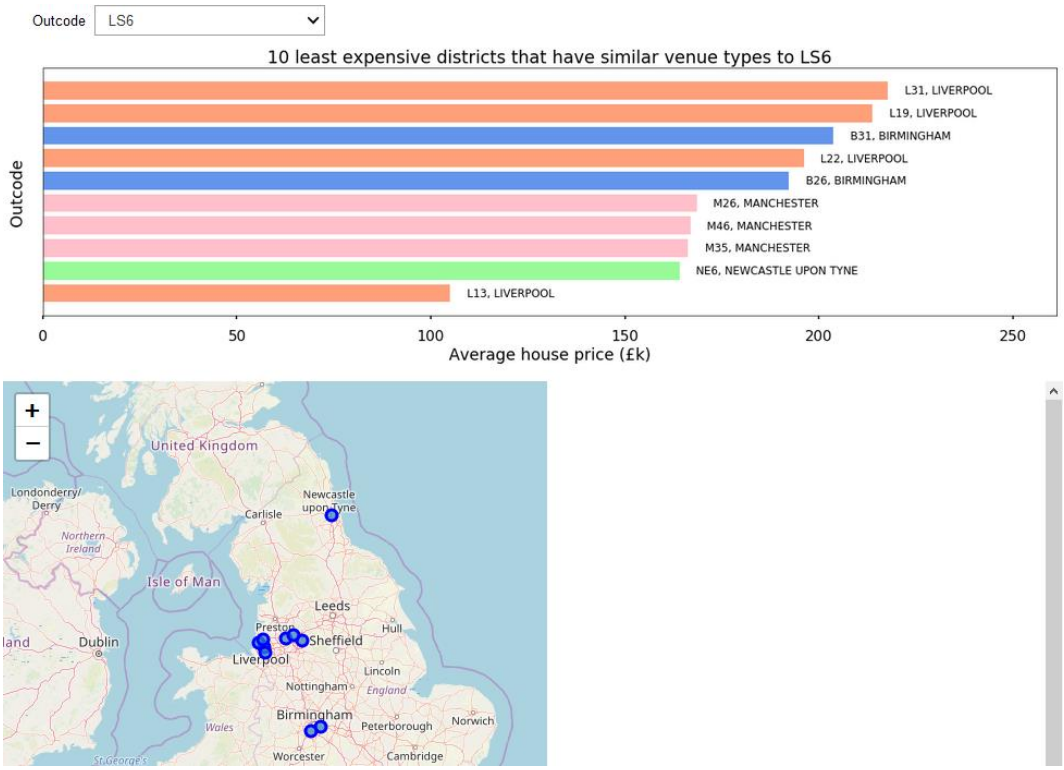
The interactive tool was then created, taking an outcode as an argument and producing a bar chart of the 10 districts with lowest average house price between 2018 and 2020, as well as a map showing the location of these districts.

Outcode OX1



Outcode NW1





Discussion

Our initial exploration of the data confirmed the premise of the analysis, i.e. more southern cities are significantly more expensive places to buy houses (see Figure 2). While most cities included in the analysis had an average house price of £200,000 to £400,000, Oxford was a notable outlier at £600,000 while London was at a whopping £1,000,000. This clearly demonstrates the need of young residents in these more expensive places to be able to find more affordable homes, without sacrificing access to amenities.

Using the tool we have generated, users can input their postcode district and the tool will suggest 'similar' areas with potentially lower house prices. For example, putting in 'OX1', the postcode district for central Oxford, will result in a suggestion of areas of Manchester, Birmingham, Liverpool, Bristol and York. For the purposes of this project, we can consider this a good proof of concept.

A small selection of cities were used for this exercise to keep the dataset to a manageable size. Additionally, the postcodes used were chosen to focus on metropolitan areas while keeping a good spread between North and South. In order to broaden the utility of this tool, we would ideally include all postcode areas in the UK.

This would, however, introduce a secondary problem - postcodes are of highly variable size, so the number of venues in each district would have to be scaled to the area. Unfortunately, the shape and size of each postcode is not freely available.[6] Future iterations of this tool could involve the purchase of these data in order to provide better clustering and therefore more accurate recommendations.

Another improvement that could be made is to further categorise the price data by property type, so users could search for e.g. semi-detached houses.

Conclusion

While this tool can of course not provide the perfect area to move to, it can provide users with some appropriate suggestions that they can further investigate. By narrowing the options, it helps our user find their perfect (affordable) house – probably in the North.

References

1. Housing affordability in England and Wales: 2016
   https://www.ons.gov.uk/peoplepopulationandcommunity/housing/bulletins/housingaffordabilityinenglandandwales/1997to2016
2. The True Cost of Living in UK Cities
   https://abcfinance.co.uk/blog/the-true-cost-of-living-in-uk-cities/
3. HM Land Registry Open Data Price Paid Dataset
   https://landregistry.data.gov.uk/app/ppd
4. Get The Data
   https://www.getthedata.com/postcode
5. Ordnance Survey - Code-Point Open
   https://www.ordnancesurvey.co.uk/business-government/products/code-point-open
6. Ordnance Survey – Code-Point with polygons
   https://www.ordnancesurvey.co.uk/business-government/products/code-point-polygons

Appendix

SPARQL query to obtain price data

```
PREFIX  xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX  text: <http://jena.apache.org/text#>
PREFIX  ppd:  <http://landregistry.data.gov.uk/def/ppi/>
PREFIX  lrcommon: <http://landregistry.data.gov.uk/def/common/>

SELECT  ?item ?ppd_propertyAddress ?ppd_transactionCategory ?ppd_transactionDate
?ppd_estateType ?ppd_hasTransaction ?ppd_pricePaid ?ppd_transactionId ?ppd_newBuild
?ppd_propertyAddressCounty ?ppd_propertyAddressDistrict ?ppd_propertyAddressLocality
?ppd_propertyAddressPaon ?ppd_propertyAddressPostcode ?ppd_propertyAddressSaon
?ppd_propertyAddressStreet ?ppd_propertyAddressTown ?ppd_propertyType ?ppd_recordStatus
WHERE
  { { ?ppd_propertyAddress
                text:query            ( lrcommon:town "( CITY )" 3000000 ) .
      ?item     ppd:propertyAddress   ?ppd_propertyAddress ;
                ppd:estateType        lrcommon:freehold ;
                ppd:transactionCategory  ppd:standardPricePaidTransaction ;
                ppd:transactionDate   ?ppd_transactionDate ;
                ppd:hasTransaction    ?ppd_hasTransaction ;
                ppd:pricePaid         ?ppd_pricePaid ;
                ppd:transactionId     ?ppd_transactionId
      FILTER ( ?ppd_transactionDate >= "2018-01-01"^^xsd:date )
      FILTER ( ?ppd_transactionDate <= "2020-02-22"^^xsd:date )
    }
    OPTIONAL
      { ?item  ppd:newBuild  ?ppd_newBuild }
    OPTIONAL
      { ?ppd_propertyAddress
                lrcommon:county  ?ppd_propertyAddressCounty
      }
    OPTIONAL
      { ?ppd_propertyAddress
                lrcommon:district  ?ppd_propertyAddressDistrict
      }
    OPTIONAL
      { ?ppd_propertyAddress
                lrcommon:locality  ?ppd_propertyAddressLocality
      }
    OPTIONAL
      { ?ppd_propertyAddress
                lrcommon:paon  ?ppd_propertyAddressPaon
      }
    OPTIONAL
      { ?ppd_propertyAddress
                lrcommon:postcode  ?ppd_propertyAddressPostcode
      }
    OPTIONAL
      { ?ppd_propertyAddress
                lrcommon:saon  ?ppd_propertyAddressSaon
      }
    OPTIONAL
      { ?ppd_propertyAddress
                lrcommon:street  ?ppd_propertyAddressStreet
      }
    OPTIONAL
      { ?ppd_propertyAddress
                lrcommon:town  ?ppd_propertyAddressTown
      }
    OPTIONAL
      { ?item  ppd:propertyType  ?ppd_propertyType }
    OPTIONAL
      { ?item  ppd:recordStatus  ?ppd_recordStatus }
    BIND(lrcommon:freehold AS ?ppd_estateType)
    BIND(ppd:standardPricePaidTransaction AS ?ppd_transactionCategory)
  }
```