Energy Performance Certificates (EPC)

EPC ratings were introduced in stages in 2007 and cover residential and non-residential buildings. Buildings will have an EPC rating if built, sold or rented after 2008.[1] Following our scope, we only look at the EPC ratings for residential buildings.

We will use the EPC data to (1) determine which property features are most predictive of energy efficiency and (2) build a labelled dataset with the known EPC ratings and heating types to use as a testing set.

Since inspectors input EPCs manually, we must be wary of errors within the data. There are no measures in place to ensure thoroughness and quality of inspections. Moreover, the EPC data overrepresents areas with a high turnover of residents rather than long-term residents since EPCs are only required when the building has a change in ownership.

Pulling EPC Data

The UK Department for Levelling Up, Housing & Communities provides public access to the Energy Performance of Buildings Data: England and Wales. They provide an API to automate pulling the EPC data. You need to sign up for an account to get your AUTH_TOKEN.

There is a limit to pulling 10,000 records simultaneously, so I iterated through each postcode in the West Midlands. I filtered the ONS UPRN Directory (August 2022) with the WMCA local authority codes and pulled all unique postcodes (see Appendix A: WMCA Local Authority Codes)

The data is updated once every six months. Our last data pull was in June 2022.

Finally, I stored all the EPC data in a .csv.

	low-energy-fixed-light-count	address	floor-height	heating-cost-potential	unheated-corridor-length	hot-water-cost-potential	construction-age-band	potential-energy-rating	mainheat-energy-eff	windows-env-eff	...	lsoa	num_meter	total_consumption	mean_counsumption	median_consumption	num_households	num_households_fuel_poverty	prop_households_fuel_poor	LATITUDE	LONGITUDE
0	NaN	Apartment 2105, Beetham Tower, 10 Holloway Cir...	2.4	133.0	5.60	184.0	England and Wales: 2003-2006	B	Average	Good	...	Birmingham 135C	1049.0	4.090256e+06	3899.195285	2766.5	668.0	99.0	14.820359	52.47544	-1.900286
1	NaN	Apartment 2201, Beetham Tower, 10 Holloway Cir...	2.4	119.0	7.50	186.0	England and Wales: 2003-2006	B	Very Poor	Good	...	Birmingham 135C	1049.0	4.090256e+06	3899.195285	2766.5	668.0	99.0	14.820359	52.47544	-1.900286
2	NaN	Apartment 2506, Beetham Tower, 10 Holloway Cir...	2.3	154.0	9.07	197.0	England and Wales: 2007-2011	B	Average	Average	...	Birmingham 135C	1049.0	4.090256e+06	3899.195285	2766.5	668.0	99.0	14.820359	52.47544	-1.900286
3	NaN	Apartment 2408, Beetham Tower, 10 Holloway Cir...	2.4	155.0	2.15	199.0	England and Wales: 2003-2006	B	Very Poor	Good	...	Birmingham 135C	1049.0	4.090256e+06	3899.195285	2766.5	668.0	99.0	14.820359	52.47544	-1.900286
4	NaN	Apartment 2502, Beetham Tower, 10 Holloway Cir...	2.4	130.0	13.40	167.0	England and Wales: 2003-2006	B	Very Poor	Good	...	Birmingham 135C	1049.0	4.090256e+06	3899.195285	2766.5	668.0	99.0	14.820359	52.47544	-1.900286

Figure 1 First 5 rows of pulled EPC data.

Data Cleaning

After pulling all the data, you will have millions of rows with over a hundred columns. The following steps explain how the data was cleaned for modelling and other analyses.

You will need to download the following Python packages: geopandas, scikit-learn and shapely.

[1] The minimum building size was 1,000m^2 from 2008, 500 m^2 from 2013 and 250 m^2 after July 2015.

(Optional) Filter houses within the West Midlands

This step is an extra measure to ensure the EPC data we pulled gives us houses in the region of interest. It can also be useful if you are looking to filter down already pulled EPC data to a specific area that cannot be easily filtered by postcode or other location information.

You will additionally require:

- Installation of geopandas, libspatialindex-dev and r-tree
- Shapefile (.shp) of the region of interest

I used geopandas used to find the houses (points) that lie in the region (polygon). This process can be slow if there are many houses.

Standardise missing data labels

The EPC data uses different missing data labels to denote different types of missing data. For our purposes, we will standardise them to the same value. This makes it easier to use functions within *pandas* and *numpy* to deal with missing data.

Clean dependent variable

The EPC data has two dependent variables of interest *current-energy-efficiency* and *current-energy-rating*. The latter is essentially the mapped version of the former.

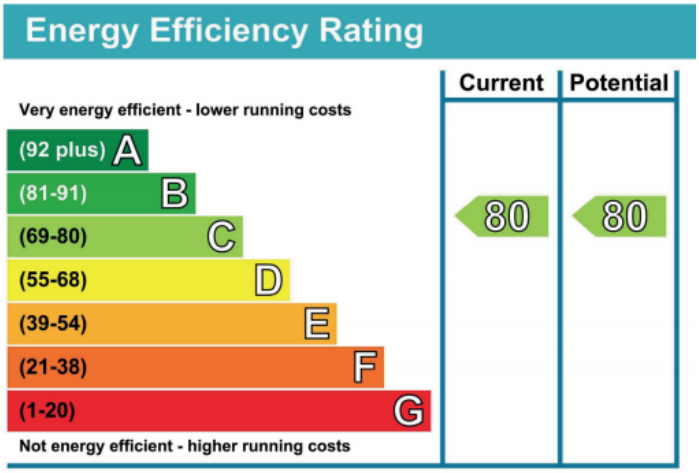

Figure 2 Breakdown of energy efficiency ratings.

We remove any missing entries since we can't make predictions without the EPC. Certain houses with energy efficiency of 0 were not given a rating, so we imputed it with 'G'.

Remove duplicate entries

The EPC database adds a new entry every time a property is bought or sold. Each property is identified by a Unique Property Reference Number (UPRN). A flat occupied by multiple tenants will have a UPRN for each tenant's room. Similarly, a house split into multiple parts occupied by different tenants will have a UPRN for each tenant's section.

We will keep the latest UPRN value since it has the most current information. Then, we will map any missing information for the following columns since we can assume that it is unlikely that these features will change over time.

- `'built-form'` : Type of house
- `'floor-level'`
- `'number-habitable-rooms'`
- `'floor-description'`
- `'roof-description'`
- `'heat-loss-corridor'`
- `'walls-description'`
- `'floor-height'`
- `'mains-gas-flag'`

Filter columns for usefulness and percentage missing

Columns with more than half missing were considered 'too many', but the cut-off point was arbitrary. We also discarded columns derived from other columns and kept columns we believed we might be able to find proxies for. See Appendix B: Removed columns from EPC data for details.

Check missing value percentage per row

We checked for this in case certain rows had virtually no data, but all the rows were mostly populated so we decided not to remove any.

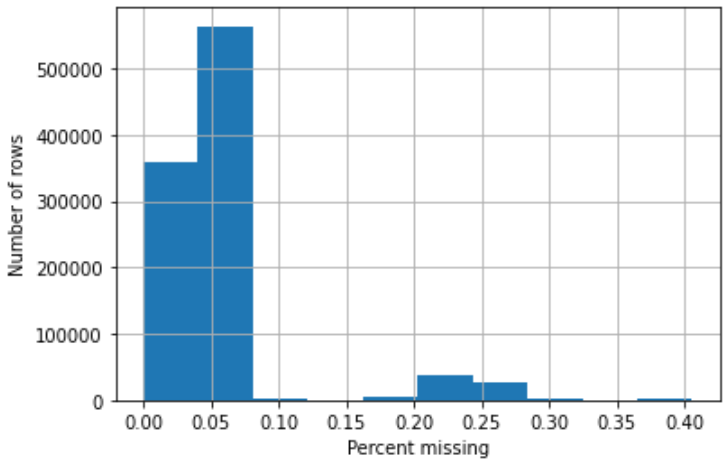

Figure 3 Histogram of the proportion of missing values in each row.

Cleaning numeric and date time columns

Since inspectors log EPC data, there is a possibility for human error in data collection and recording. Therefore, we performed a sanity check on the remaining columns to ensure all the entries were in the correct data type and had logically reasonable values.

We plotted each numeric column's distribution in a boxplot, giving a better view of the outliers in the data. We determined that all the following columns had reasonable values and did not decide to clip any of the values.

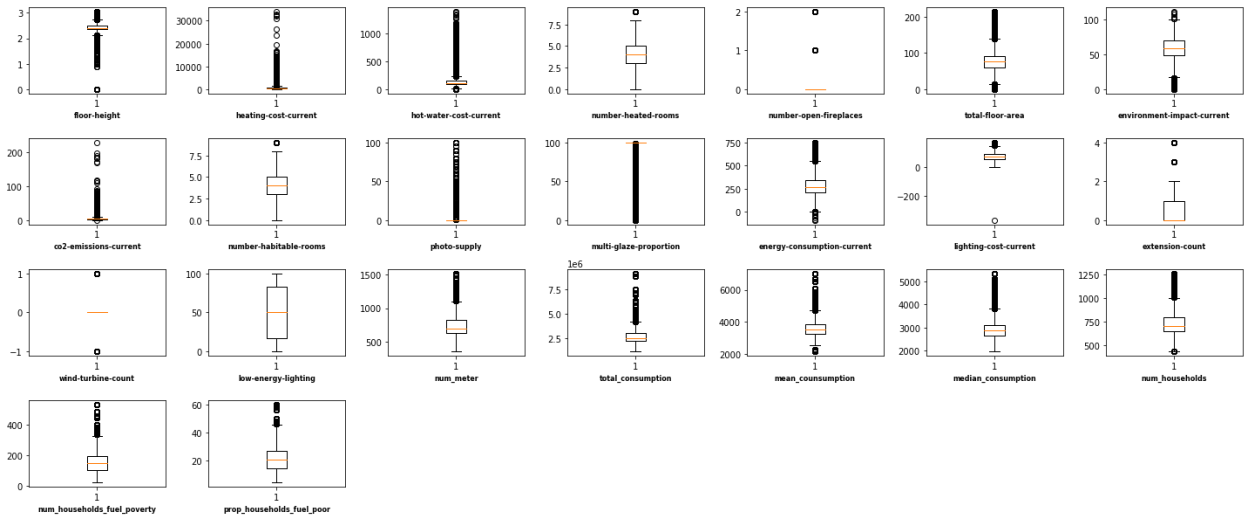

Figure 4 Boxplots showing the distribution of numeric variables.

Cleaning and encoding non-numeric data

Ideally, all textual data will be standardised into buckets of data with low cardinality for one-hot encoding the data. Particularly with the descriptions of house features, we used the methods from Sonia William's *Using machine learning to predict energy efficiency* needed to:

1. Translate Welsh to English
2. Standardise spelling, punctuation and phrasing
3. Remove unnecessary phrasing

Then, we applied CHAID (Chi-square Automatic Interaction Detector). Generally, the algorithm creates a decision tree. It randomly groups unique responses into a node and determines which split in the node brings us closer to accurately predicting the outcome.[2] We will take the nodes of the finished tree which have the optimal groupings of unique responses and use them to reduce the number of unique values in the column. We can see the items that are grouped into each category as a sanity check. You can read more in-depth about the method here.

Finally, we encoded the data based on their cardinality. We only one hot encoded the local authority and constituency because they have relatively lower cardinality. All other variables were label encoded, meaning we mapped all unique variables to an integer and replaced the text with the integer. We did this to reduce the file size.

Impute Missing Data

We used an iterative imputing process to fill in some missing data. The iterative imputer trains a regression model on the columns that contain data and uses those to predict the missing values. It does this iteratively, as the name suggests, by starting with the column with the fewest missing data points,

[2] The metric it uses is Chi-square which is a statistical test that tells us whether two categorical variables are independent.

then moving on to the following fewest missing points and so on, thus filling in the data frame. For more details on the iterative imputing process. See Sci-Kit Learn Documentation.

We also attempted a K-Nearest-Neighbors imputing method. We had a large amount of missing data and data frame size, which meant that the method required a high computation time to complete. Therefore, we abandoned the method because it was not viable within our technical and time constraints.

Determining the most predictive EPC features

We trained a simple random forest model on all the available features in the EPC dataset to determine the most important and predictive features for predicting the EPC ratings. The figure below contains the results.

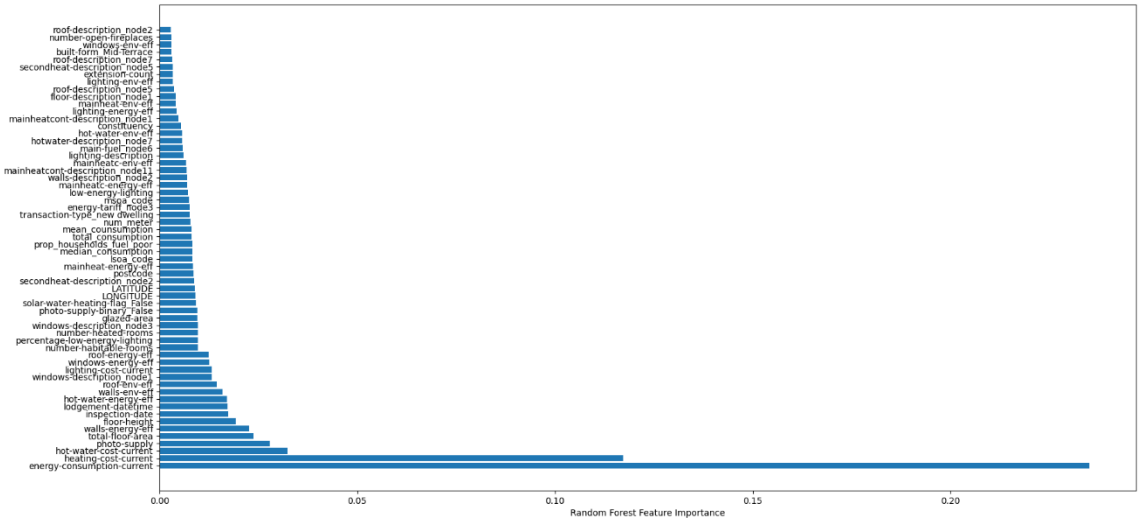

Figure 5 Feature importance from random forest model to predict energy efficiency.

The top five columns are energy consumption, heating cost, hot water cost, photo supply, and total floor area. We could only obtain a proxy value for the total floor area out of the top predictive features. Since energy consumption was the most predictive, we attempted to impute the energy consumption data using the iterative imputer, but it failed to fill in any of the energy consumption data reliably.

Build Training Data

We will use the EPC data to get the true values for the energy efficiency (*'current-energy-efficiency'* and *'current-energy-rating'*), heating types (*'mainheat-description'*) and solar panel (*'photo-supply'*) of each house, using the UPRN as the index.

The final output has 738,113 rows and 5 columns, which covers all the houses with an EPC rating in the West Midlands. We have shared a file called *cleaned_epc_data.csv,* which has the 181 columns of cleaned data from the original EPC database, including the columns we one hot encoded. See Glossary: Domestic EPCs to understand the variables.

Column Name	Column	Type
uprn	UPRN	String
current-energy-efficiency	Energy efficiency	Float
current-energy-rating	Energy rating	String
mainheat-description	Electric (1) or non-electric (0) heating	Binary
photo-supply-binary	Solar panel installed (1) or not installed (0)	Binary

Appendix

Appendix A: WMCA Local Authority Codes

The codes for all the local authorities in the West Midlands.

```
{
'E08000025': 'Birmingham',
'E08000031': 'Wolverhampton',
'E08000026': 'Coventry',
'E08000027': 'Dudley',
'E08000028': 'Sandwell',
'E08000029': 'Solihull',
'E08000030': 'Walsall',
'E07000192': 'Cannock Chase',
'E07000218': 'North Warwickshire',
'E07000219': 'Nuneaton and Bedworth',
'E07000236': 'Redditch',
'E07000220': 'Rugby',
'E06000051': 'Shropshire',
'E07000221': 'Stratford-on-Avon',
'E07000199': 'Tamworth',
'E06000020': 'Telford and Wrekin',
'E07000222': 'Warwick'
}
```

Appendix B: Removed columns from EPC data

Column	Reason	Column	Reason
'sheating-env-eff'	Too many missing	**'msoa'**	Unnecessary
'sheating-energy-eff'	Too many missing	**'lmk-key'**	Unnecessary
'flat-storey-count'	Too many missing	**'constituency-label'**	Unnecessary
'floor-env-eff'	Too many missing	**'environment-impact-potential'**	Unnecessary
'floor-energy-eff'	Too many missing	**'potential-energy-rating'**	Unnecessary
'unheated-corridor-length'	Too many missing	**'lighting-cost-potential'**	Unnecessary
'county'	Too many missing	**'co2-emissions-potential'**	Unnecessary
'heat-loss-corridor'	Too many missing	**'potential-energy-efficiency'**	Unnecessary
'flat-top-storey'	Too many missing	**'energy-consumption-potential'**	Unnecessary
'co2-emiss-curr-per-floor-area'	Derived	**'hot-water-cost-potential'**	Unnecessary
'low-energy-fixed-light-count'	Too many missing and similar variable available	**'heating-cost-potential'**	Unnecessary
'fixed-lighting-outlets-count'	Too many missing and similar variable available	**'posttown'**	Unnecessary
'local-authority-label'	Unnecessary	**'building-reference-number'**	Unnecessary
'lsoa'	Unnecessary	**‘mainheat-env-eff’**	Used to get final ratings
‘mainheatc-env-eff’	Used to get final ratings	**‘mainheatc-energy-eff’**	Used to get final ratings
‘walls-env-eff’	Used to get final ratings	**‘walls-energy-eff’**	Used to get final ratings
‘hot-water-energy-eff’	Used to get final ratings	**‘roof-env-eff’**	Used to get final ratings
‘windows-energy-eff’	Used to get final ratings		