DAT102x: Predicting County-Level Rents

Michael K. - December 2019

Executive summary

This document resumes the work made during the DAT102x competition.

A dataset containing demographic and socioeconomic information regarding the United States was analysed, and a machine learning method was trained in order to predict the median gross rent at the county level.

The present document is divided into 4 sections, which contain all relevant information about the methodology that was followed in order to properly reach the project's goal.

Data description contains an exhaustive description of data. This is: the size of the dataset used in the different stages of this project and its composition. The semantic meaning of each feature is also described.

Data analysis and cleaning gathers a collection of all resources that were used to analyze the data and the decisions that were made in order to prepare it for the next stage. Some interesting visualizations are included to support the analysis and to improve reader's experience.

Regression experiments synthesizes the work made at building the model for the median gross rent prediction. Different regressors were tried and several combinations of its more relevant parameters were tested. The performance of all of these variants were compared and decisions were made in order to choose for the most promising methods. A scaling stage was added as a possibility prior training the regression algorithm.

Conclusions summarizes the most important observations and remarks about the work made in the different parts of this project.

Data description

The data available for this project comprises of two different sets[1] meant for training and testing a regression method. The former of these datasets is composed of 1563 elements and the latter by 1575. Both of them have 43 features, with a variety of socioeconomic and demographic estimators about the United States, grained at county level. Train set also contains a column with the information of median gross rent for each county, which is used as label.

The features can be grouped into different categories, according to the type of information they provide about the data.

The description for each feature and its categorization is provided here[2]:

ID

- county_code - Unique identifier for each county
- state - Unique identifier for each state
- population - Total population

Housing

- renter_occupied_households - Count of renter-occupied households
- pct_renter_occupied - Percent of occupied housing units that are renter-occupied
- evictions - Number of eviction judgments in which renters were ordered to leave in a given area and year
- rent_burden - Median gross rent as a percentage of household income

Ethnicity

- pct_white - Percent of population that is White alone and not Hispanic or Latino
- pct_af_am - Percent of population that is Black or African American alone and not Hispanic or Latino
- pct_hispanic - Percent of population that is of Hispanic or Latino origin
- pct_am_ind - Percent of population that is American Indian and Alaska Native alone and not Hispanic or Latino
- pct_asian - Percent of population that is Asian alone and not Hispanic or Latino
- pct_nh_pi - Percent of population that is Native Hawaiian and Other Pacific Islander alone and not Hispanic or Latino
- pct_multiple - Percent of population that is two or more races and not Hispanic or Latino

---

[1] Actually, as provided by the instructors there are 3 different files; train_features, test_features and train_labels. When a 'train set' is mentioned in this document, it refers at the concatenation of train_features and train_label files.
[2] This information was extracted from DataScienceCapstone, available at this site.

- pct_other - Percent of population that is other race alone and not Hispanic or Latino

Economic

- poverty_rate - Percent of the population with income in the past 12 months below the poverty level
- rucc - Rural-Urban Continuum Codes "form a classification scheme that distinguishes metropolitan counties by the population size of their metro area, and nonmetropolitan counties by degree of urbanization and adjacency to a metro area. The official Office of Management and Budget (OMB) metro and nonmetro categories have been subdivided into three metro and six nonmetro categories. Each county in the U.S. is assigned one of the 9 codes." (USDA Economic Research Service)
- urban_influence - Urban Influence Codes "form a classification scheme that distinguishes metropolitan counties by population size of their metro area, and nonmetropolitan counties by size of the largest city or town and proximity to metro and micropolitan areas." (USDA Economic Research Service)
- economic_typology - County Typology Codes "classify all U.S. counties according to six mutually exclusive categories of economic dependence and six overlapping categories of policy-relevant themes. The economic dependence types include farming, mining, manufacturing, Federal/State government, recreation, and nonspecialized counties. The policy-relevant types include low education, low employment, persistent poverty, persistent child poverty, population loss, and retirement destination." (USDA Economic Research Service)
- pct_civilian_labor - Civilian labor force, annual average, as percent of population.
- pct_unemployment - Unemployment, annual average, as percent of population

Health

- pct_uninsured_adults - Percent of adults without health insurance
- pct_uninsured_children - Percent of children without health insurance
- pct_adult_obesity - Percent of adults who meet clinical definition of obese
- pct_adult_smoking - Percent of adults who smoke
- pct_diabetes - Percent of population with diabetes
- pct_low_birthweight - Percent of babies born with low birth weight
- pct_excessive_drinking - Percent of adult population that engages in excessive consumption of alcohol
- pct_physical_inactivity - Percent of adult population that is physically inactive
- air_pollution_particulate_matter_value - Fine particulate matter in $\mu g/m^3$
- homicides_per_100k - Deaths by homicide per 100,000 population
- motor_vehicle_crash_deaths_per_100k - Deaths by motor vehicle crash per 100,000 population
- heart_disease_mortality_per_100k - Deaths from heart disease per 100,000 population
- pop_per_dentist - Population per dentist
- pop_per_primary_care_physician - Population per Primary Care Physician

Demographic

- pct_female - Percent of population that is female
- pct_below_18_years_of_age - Percent of population that is below 18 years of age
- pct_aged_65_years_and_older - Percent of population that is aged 65 years or older
- pct_adults_less_than_a_high_school_diploma - Percent of adult population that does not have a high school diploma
- pct_adults_with_high_school_diploma - Percent of adult population which has a high school diploma as highest level of education achieved
- pct_adults_with_some_college - Percent of adult population which has some college as highest level of education achieved
- pct_adults_bachelors_or_higher - Percent of adult population which has a bachelor's degree or higher as highest level of education achieved
- birth_rate_per_1k - Births per 1,000 of population
- death_rate_per_1k - Deaths per 1,000 of population

Data analysis and cleaning

Data types

To further process the data, it is important to understand the data type for each feature.

There are three different data types in the dataset: *categorical*, *integer* and *float*. The first data type is the case of *county_code*, *state*, *rucc*, *urban_influence* and *economic_typology*. The second group is composed by *heart_disease_mortality_per_100k* only and the rest of features are *float* type.

The label, *gross_rent*, is also an *integer*.

All categorical data will be converted into integer values to avoid type problems at training stage.

Data statistics

In order to have a very quick overview of numerical features composition in the training set, the **Table 1** summarizes its most meaningful statistics, this is: mean, standard deviation, minimum and maximum value.

feature	count[3]	mean	std	min	max
population	1562	108.340.684	374.522.903	269.000	10.020.287
renter_occupied_households	1562	14.904.620	62.559.473	64.000	1.760.277
pct_renter_occupied	1562	28.526	8.122	7.279	73.008
evictions	1235	397.411	1.522.801	-1.000	29.251.000
rent_burden	1562	28.538	4.670	9.909	49.665
pct_white	1562	769	203	10	995
pct_af_am	1562	89	144	0	756
pct_hispanic	1562	92	142	0	987
pct_am_ind	1562	18	75	0	816
pct_asian	1562	13	27	0	418
pct_nh_pi	1562	1	3	0	85
pct_multiple	1562	18	16	0	184
pct_other	1562	1	2	0	20
poverty_rate	1562	12.183	5.784	0	38.792
pct_civilian_labor	1562	471	71	186	996
pct_unemployment	1562	63	23	12	242

[3] *Count* refers to the amount of not nan values in that column.

pct_uninsured_adults	1560	220	67	53	520
pct_uninsured_children	1560	89	41	18	327
pct_adult_obesity	1560	305	44	133	474
pct_adult_smoking	1344	212	64	31	513
pct_diabetes	1560	107	23	33	180
pct_low_birthweight	1446	83	21	30	182
pct_excessive_drinking	1100	165	51	32	419
pct_physical_inactivity	1560	277	53	104	446
air_pollution_particulate_matter_value	1542	11.637	1.534	7.209	14.992
homicides_per_100k	613	5.752	4.298	-80	26.920
motor_vehicle_crash_deaths_per_100k	1372	21.715	10.721	3.140	110.450
heart_disease_mortality_per_100k	1562	275.483	57.828	76.000	511.000
pop_per_dentist	1447	3.421.829	2.538.671	340.000	25.169.000
pop_per_primary_care_physician	1448	2.508.304	1.960.312	279.000	16.740.000
pct_female	1560	499	24	314	564
pct_below_18_years_of_age	1560	229	35	82	415
pct_aged_65_years_and_older	1560	168	45	36	488
pct_adults_less_than_a_high_school_diploma	1562	146	67	19	536
pct_adults_with_high_school_diploma	1562	346	71	74	536
pct_adults_with_some_college	1562	303	52	114	477
pct_adults_bachelors_or_higher	1562	205	92	64	788
birth_rate_per_1k	1562	11.621	2.756	3.654	29.035
death_rate_per_1k	1562	10.415	2.772	961	24.281
gross_rent	1562	701,142	192,883	351	1.827

Table 1.: Summary statistics for the numerical features in training set.

It is observed that several features contain nan values, and some of them in large quantities, as it is the case of *homicides_per_100k* and *pct_excessive_drinking*. This has to be taken cared in a future processing stage.

Also, to understand what is the distribution of average gross rent values, its histogram is plotted in **Figure 1**. This distribution is of great importance in the present analysis, given that it is the target value for the prediction problem. It can be seen from this representation that although it ranges between 351 and 1827, most values are contained between the interval [500, 900].

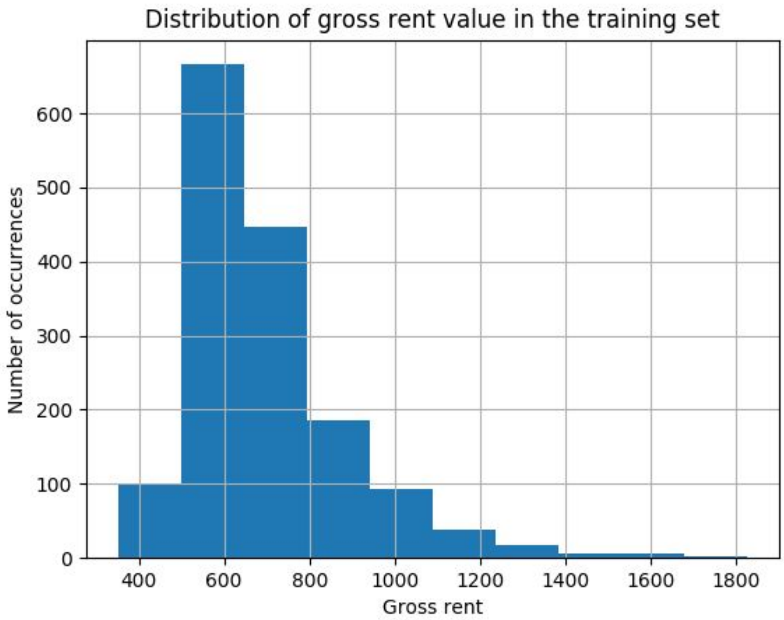

Figure 1.: Distribution of gross rent in the training set.

Nan removal

As it was previously discussed, there are several features with missing data. This has to be solved before the data is fed to a regression algorithm. The possible ways to solve this problem are either imputation or data removal.

Table 2 summarizes the quantity of missing data for all features, in the totality of training and testing datasets. To simplify the visualization, only values greater than zero are shown.

feature	missing data
evictions	640
pct_uninsured_adults	5
pct_uninsured_children	5
pct_adult_obesity	5
pct_adult_smoking	433
pct_diabetes	5
pct_low_birthweight	226
pct_excessive_drinking	919
pct_physical_inactivity	5
air_pollution_particulate_matter_value	33
homicides_per_100k	1888
motor_vehicle_crash_deaths_per_100k	367

pop_per_dentist	221
pop_per_primary_care_physician	205
pct_female	5
pct_below_18_years_of_age	5
pct_aged_65_years_and_older	5

Table 2.: Missing data in the totality of training and testing datasets.

Table 2 shows that some features have only a few missing values and other ones have hundreds of them. To decide if the data is removed or imputed, a criteria was chosen regarding the quantity of this missing data.

If the missing data is less than 1% of the total dataset, the data is imputed with the median value for that feature. In other case, the feature is removed completely from the dataset and it is not taken into account to build the predictive model.

Feature engineering

Although the *state* feature was changed to numerical, the numbers still represent categories. Under this representation, if *a* and *b* are states, and *a* > *b*, it doesn't say anything about the problem and it could potentially confuse a classifier.

Inspired by one of the questions of first challenge, where a relationship was stated between the number of counties in a state and the average gross rent in that state, a decision was made regarding the *state* feature label, to improve its correlation with the target.

Figure 2 shows the average of gross rent values in a state as a function of the number of counties in that state. It can be seen very clearly that the greater the number of counties in a state, the average gross rent decreases.

A decision was made to change the content of *state* feature from current value to the number of counties in that state. This new feature is also numeric, but it will have a greater relevance at building our predictive model.

Pearson correlation

In order to study the impact of each feature in the target, the Pearson correlation coefficient is calculated. This coefficient ranges between -1 and 1. For a given pair of variables, if the Pearson coefficient is close to 1 (in absolute value) means the variables are very correlated, if it is close to 0 means there is not a linear correlation between them.

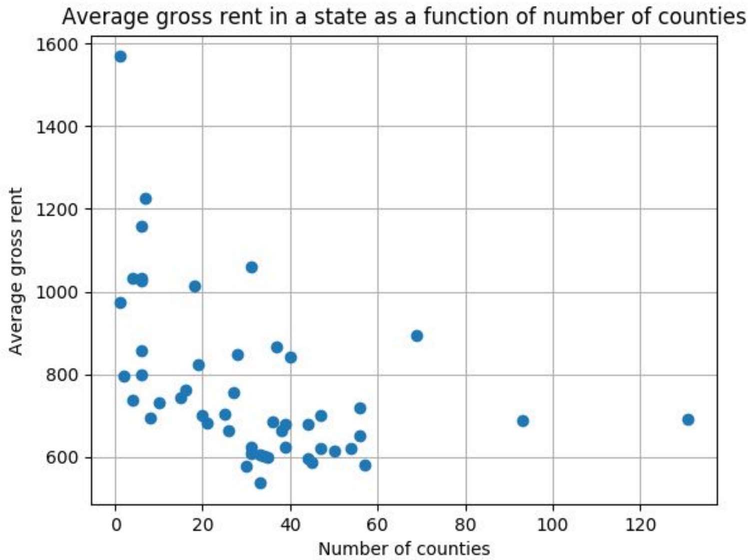

Figure 2.: Average gross rent in a state as a function of number of counties in that state. It can be seen a clear relationship between these variables.

Table 3 shows the correlation coefficient for all features. From this, it can be seen that the features *pct_adults_bachelors_or_higher*, *pct_adults_with_high_school_diploma*, *pct_asian*, *death_rate_per_1k* and *pct_physical_inactivity* have the highest correlation with *gross rent* target.

feature	correlation
pct_af_am	0,01614720824
pct_am_ind	0,01794168821
pct_adults_with_some_college	0,01941370217
county_code	0,02444943498
pct_below_18_years_of_age	0,04622110176
birth_rate_per_1k	0,04699824541
pct_female	0,04821064918
pct_uninsured_children	0,07066753309
air_pollution_particulate_matter_value	0,09197105707
pct_unemployment	0,09424046291
state	0,10976144252
pct_excessive_drinking	0,11644841886
pct_uninsured_adults	0,16328965334
pct_hispanic	0,17045651803
pct_low_birthweight	0,17705312513
pct_nh_pi	0,20137541197

pct_white	0,22698499685
pop_per_primary_care_physician	0,22929729470
rent_burden	0,23721288828
urban_influence	0,24066566531
pct_civilian_labor	0,24429800537
pct_multiple	0,25597729005
pct_renter_occupied	0,27861266160
pop_per_dentist	0,28682216465
pct_adults_less_than_a_high_school_diploma	0,30660820290
pct_adult_smoking	0,30921045260
economic_typology	0,31462872172
evictions	0,32565331287
homicides_per_100k	0,33721943193
pct_other	0,33844493102
renter_occupied_households	0,34515449481
poverty_rate	0,35254800298
pct_aged_65_years_and_older	0,38670474671
population	0,39834315536
heart_disease_mortality_per_100k	0,42647467817
pct_diabetes	0,43776454589
rucc	0,44102798855
pct_adult_obesity	0,47049944497
motor_vehicle_crash_deaths_per_100k	0,50036101685
pct_physical_inactivity	0,57879028201
death_rate_per_1k	0,58919107415
pct_asian	0,59207654303
pct_adults_with_high_school_diploma	0,60153972356
pct_adults_bachelors_or_higher	0,67910051157

Table3.: Pearson correlation coefficient for all features and target.

This analysis, not only gives a great insight on how the features are important to predict the target, but it also allows to rank the features according to their relevance.

Regression experiments

After the data is cleaned and preprocessed, the training set is split into two: a larger set that will be used to train the regressor algorithm and a smaller one used to validate the results. For the rest of this section, these two sets are called *training set*[4] and *validation set*.

Several experiments were made regarding the different combinations available to train a successful regressor. These possibilities involved the type of regressor being used, the hyperparameters of this regressor, the size of the training/validation split, if a scaling stage was used and the number of features that were fed into the algorithm. All the performances were always measured as the average in a 5-fold cross validation.

About the regression algorithm, only two were tested; Random Forest and AdaBoost. The performances were very similar in both, so AdaBoost was chosen. The reason for this is that AdaBoost is a more complex classifier with more parameters to tune in order to improve regression accuracy.

The results obtained in the combination of different hyperparameters and the use of a scaling method were not conclusive. The most meaningful parameters were believed to be the number of features that were used to train the algorithm and the size of training and validation sets.

For a given combination of hyperparameters, **Figure 3** shows the performance of an AdaBoost regressor with different sizes of training/validation splits. The x axis is the number of features used in training; if *k* features were used, it means the *k* most meaningful features according to **Table 3** were selected.

[4] Although this a naming abuse, it should be clear to the reader what set are we referring to with *training set*, in each occasion.

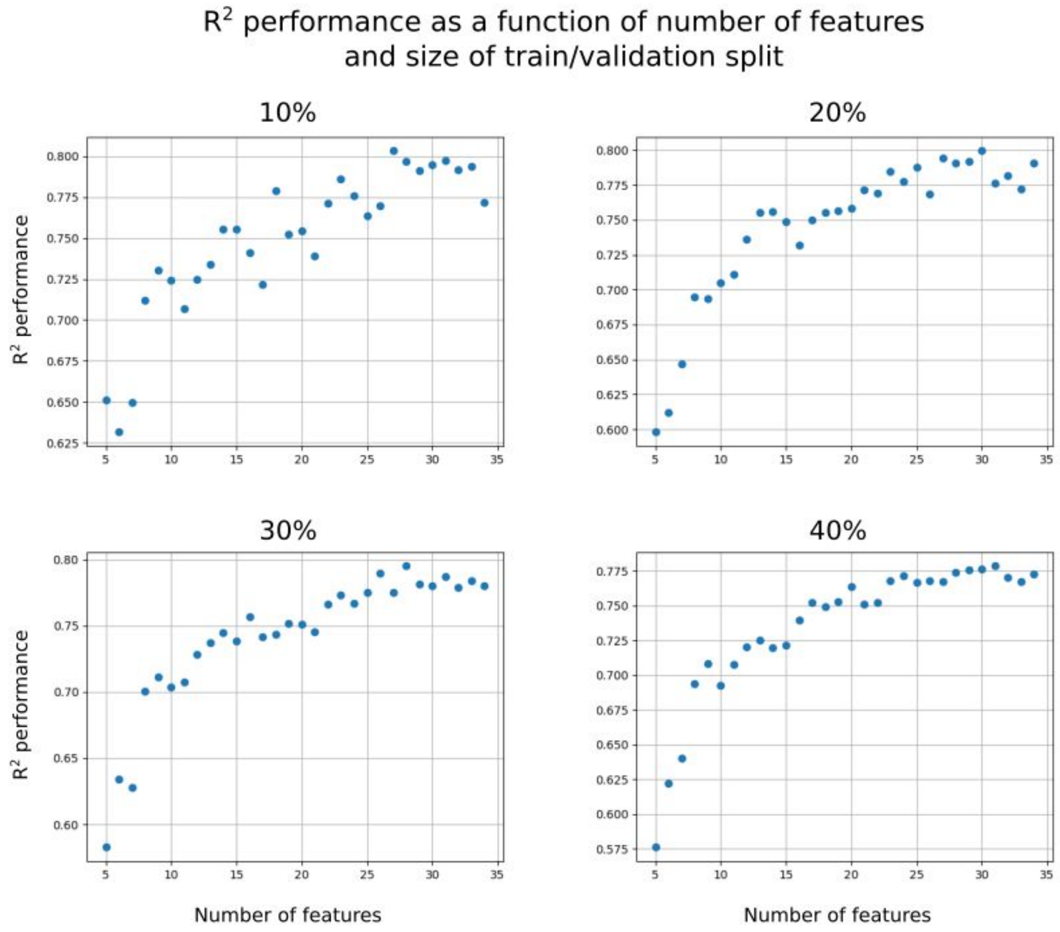

Figure 3.: R^2 performance as a function of number of features. Each plot represents a different value for train/validation split, as it is stated in its title.

From this Figure it can be seen that a value of features between 25 and 35 achieves the optimal performance in all cases. Also, it is interesting to see how the performance grows as the number of features increases, it appears to be an exponential growth that reaches an asymptote.

As for the size of the training/validation sets, larger values produce a more stable growth, which is desirable. Instability in this scenario can be interpreted as overfitting. But larger values in train/validation split are also related to a decrease in performance. A trade off between performance and stability is needed to propose a solution.

Given that the algorithm will be tested in a large testing set (larger that the one used for training), and that the number of possible submissions is limited, it is needed to enlarge the size of validation set as much as possible without affecting the performance. For this, a value of 30% is selected.

Doing so, and selecting the number of features at 30, the trained AdaBoost regressor reaches an accuracy of 77% in the testing set.

Conclusions

Data cleaning and preprocessing is a very important stage in order to prepare the data to be fed into a classification/regression algorithm. Among other methods, feature engineering is an interesting resource to improve the correlation of features with the target.

It is possible to train a machine learning regression algorithm in order to predict the gross rent value at county level, with the data that was given for this problem.

The performance depends on a variety of hyperparameters, some of them have a larger impact than others. Choosing the most adequate split size to train the algorithm is crucial, given that the algorithm can overfit or its performance can be compromised.