Homework 5

Due: May 4, 2018 @ 11:59pm

Instructions

For this assignment, you will be guided through the process of building a regression model that predicts the market value of condominiums in New York City using a dataset published by the New York City Department of Finance.

Obtain the Github repository you will use to complete homework 5 that contains a starter RMarkdown file named `homework_5.Rmd`, which you will use to do your work and write-up when completing the questions below. Remember to fill in your name at the top of the RMarkdown document and be sure to save, commit, and push (upload) frequently to Github so that you have incremental snapshots of your work. When you're done, follow the How to submit section below to setup a Pull Request, which will be used for feedback.

About the dataset

This dataset reports on the market valuations of condominiums in New York City for Fiscal Year 2011/2012. The data[1] comes from the New York City Department of Finance and was made available to the general public on the NYC OpenData website (https://opendata.cityofnewyork.us/). The official description for the dataset is as follows:

> Condominiums and cooperatives are valued as if they were residential rental apartments. Income information from similar rental properties is applied to determine value. The Department of Finance (DOF) chooses similar properties to value condos and coops. Properties are selected based on a combination of factors such as: land location, income levels, building age and construction and exemptions and subsidies.

The data in the files `housing_train.rds` and `housing_test.rds` was adapted from the cleaned and aggregated version by data scientist Jared Lander (https://www.jaredlander.com/data).

Variable	Description
`boro`	Borough where building is located. New York City is divided into 5 boroughs, Manhattan, The Bronx, Brooklyn, Queens, and Staten Island.
`neighborhood`	Neighborhood of building location. The neighborhood name is assigned by the New York City Department of Finance, and in most cases is the same as the neighborhood's common name.
`class`	Building classification code assigned by the New York City Department of Finance. There are four building classifications for the condominiums in the dataset, rental, walk-up, elevator, and co-op.
`year_built`	Year the building was built
`units`	Total number of units in the building
`sqft`	Gross square footage of the building
`value_per_sqft`	Total market value per squarefoot of the land and building

[1] Data set aggregated from the following sources:
https://data.cityofnewyork.us/Finances/DOF-Condominium-Comparable-Rental-Income-Manhattan/dvzp-h4k9
https://data.cityofnewyork.us/Finances/DOF-Condominium-Comparable-Rental-Income-Brooklyn-/bss9-579f
https://data.cityofnewyork.us/Finances/DOF-Condominium-Comparable-Rental-Income-Queens-FY/jcih-dj9q
https://data.cityofnewyork.us/Property/DOF-Condominium-Comparable-Rental-Income-Bronx-FY-/3qfc-4tta
https://data.cityofnewyork.us/Finances/DOF-Condominium-Comparable-Rental-Income-Staten-Is/tkdy-59zg

Questions

The main goal of this homework assignment is to build a model that predicts the market value per squarefoot — the variable `value_per_sqft` — of condominiums in New York City. We should not expect to be able to construct a model with 100% precision, but we would like to uncover trends in the data. This allows us to pose the driving question for our analysis as follows:

> What are key factors that affect the overall price of condominiums in New York City?

When building and evaluating predictive models, it is standard protocol to split your dataset into a **training dataset** and a **test dataset**. This has already been done for you, with the training dataset loaded into the variable `housing_train` and the testing dataset loaded into `housing_test`. You will be using `housing_train` for most of the homework to build and cross-validate your models. Once you've selected your model, as a final step you will use it to predict the `value_per_sqft` column in the dataset stored in `housing_test`.

For your convenience, the helper function `rep_kfold_cv(data, k, model, cv_reps)` is loaded into your R environment and will run the code that cross-validates your models.[2] This function requires four inputs: `data` is a tibble of your training dataset, `k` is an integer specifying the number of folds for cross-validation, `model` is a formula written in the format used for the `lm()` function (`money ~ work + time` for example), and `cv_reps` is an integer specifying how many times to repeat the k-fold cross-validation to improve your statistical averages.

---

1. Create the following visualizations to explore the dataset:
   - A histogram of `value_per_sqft` faceted over boroughs of New York City
   - Boxplots of `units` (y axis) for the different boroughs (x axis) plotted two different ways: in a normal scale and in a `log10()` scale along the y axis (see http://r4ds.had.co.nz/graphics-for-communication.html#replacing-a-scale for how to scale the axes)
   - Boxplots of `sqft` (y axis) for the different boroughs (x axis) plotted two different ways: in a normal scale and in a `log10()` scale along the y axis
   - Scatterplots of `value_per_sqft` (y axis) versus `units` (x axis) using `log10()` scaling for `units`. Facet over two variables: boroughs **and** condominium classification.

   Based on your plots so far, which variables (columns) in the dataset seem to have the strongest overall impact on the condominium values?

2. The boxplots of `units` in the previous question should reveal extreme outliers in the plot. Since our goal is to model general trends and not precise values, fitting to these data points may skew our model in an unhelpful way. Filter the dataset to remove these outliers (there are 3 in all).

   Note: Besides these 3 extreme points, there are other potential outliers that you might consider removing. If you detect others, you are welcome to remove them to see if it helps you build your model, provided you explain why you're removing them.

3. To begin, try building univariate (single variable) models and see how they compare with each other. Use `value_per_sqft` as your response variable and then try `boro`, `class`, `units`, and `sqft` for your explanatory variable (this means you will try out 4 different models). Plug these models into the k-fold cross-validation function `rep_kfold_cv()` with `k = 10` and `cv_reps = 3`. Compare the mean-square error (MSE), both unadjusted and adjusted, and R^2 for these models, noting that models with better predictive power will have lower MSE and higher R^2 scores. Which model did the best so far?

---

[2] For those that are interested in seeing how you would implement k-fold cross-validation using the `tidyverse` packages, the code for the function `rep_kfold_cv()` can be found in the file `repeated_kfold_cross_validation.R` distributed with your Github repo.

These next three questions can be completed for extra credit, provided you've already completed the first three questions in full. These will guide you through building a more complicated multivariate model and making predictions with your final model on the `housing_test` dataset.

Answering each question correctly will give you some extra credit, but you must complete them in order. This means you shouldn't skip question 4 and just try and answer questions 5 and 6. Submissions that skip over or provide incomplete answers for these questions will not receive any extra credit.

4. Build and cross-validate at least 3 multivariate models that predict `value_per_sqft`, using the k-fold cross-validation parameters `k = 10` and `cv_reps = 3`. An example of a multivariate model is `value_per_sqft ~ boro + units`. You may also want to consider including interaction terms (see http://r4ds.had.co.nz/model-basics.html#interactions-continuous-and-categorical for a quick review). For example, you might try `value_per_sqft ~ boro + class * sqft`. Which of your models performs the best? Is it significantly better than your best model in the last question?

5. Now that you've selected your model, train it on the full dataset:

   ```
   final_model <- lm(model_formula, data = housing_train)
   ```

   where `model_formula` is the best performing model from the previous question.

   To predict values in the testing set, use `add_predictions()` from the `modelr` package to put the model predictions directly into your testing dataset. Then calculate the mean-square error for the predictions:

   ```
   test_predictions %>%
     summarize(mse = sum((value_per_sqft - pred)^2)/n())
   ```

   This score is useful because it is absolute and allows you to compare how well your model performs against all other model types. Can you do better than a MSE score of 2030.34?

6. To wrap up, evaluate how well your model obeys the conditions for least squares linear regression, which are summarized on page 238 of the *Introductory Statistics with Randomization and Simulation* textbook. Make two plots to inspect how well your model conforms to the requirements for linear regression:

   - To evaluate the residual spread, make a scatterplot of `(value_per_sqft - pred)` (y axis) versus `pred` (x axis)
   - To inspect whether the residual distribution is nearly normal, make a Q-Q plot of `(value_per_sqft - pred)`.

   Explain whether your model obeys the conditions for least squares linear regression.

How to submit

When you are ready to submit, be sure to save, commit, and push your final result so that everything is synchronized to Github. Then, navigate to **your copy** of the Github repository you used for this assignment. You should see your repository, along with the updated files that you just synchronized to Github. Confirm that your files are up-to-date, and then do the following steps:

1. Click the *Pull Requests* tab near the top of the page.
2. Click the green button that says "New pull request".
3. Click the dropdown menu button labeled "base:", and select the option `starting`.
4. Confirm that the dropdown menu button labeled "compare:" is set to `master`.
5. Click the green button that says "Create pull request".
6. Give the *pull request* the following title: `Submission: Homework 5, FirstName LastName`, replacing `FirstName` and `LastName` with your actual first and last name.
7. In the messagebox, write: `My homework submission is ready for grading @shuaibm @jkglasbrenner`.
8. Click "Create pull request" to lock in your submission.

Cheatsheets

You are encouraged to review and keep the following cheatsheets handy while working on this assignment:

- RStudio cheatsheet
- RMarkdown cheatsheet
- RMarkdown reference
- `ggplot2` cheatsheet
- Data transformation cheatsheet
- Data import cheatsheet