**Rogers Rangers – A Predictive Analysis of   
New York City Property Sales in 2017**

Arturo Davila-Andino, James Baker,   
Giridhar Kaushik-Ramachandran, Andrew So

George Mason University

**Abstract**

New York City is a large premier metropolitan center of the United States with over 20 million residents and an important hub for international diplomacy. The data that this paper uses is a Kaggle dataset of a years’ worth of property sales in New York City in 2017.

The team seeks to fit an effective regression model to predict the value of a property within the city based on its qualitative and quantitative properties. The team initially attempted a linear regression over the dataset in the hopes of predicting the Sales Price. In our analysis we found that traditional Linear Regression reacted positively against the response variable, with a relatively high . In an attempt to improve the predictive power against the data, the team fitted a variety of both linear and nonlinear models against a tuned version of the dataset to get more cohesive results. We found that Partial Least Squares performed the strongest with a of 0.99, which has a fair predictive power against this dataset, but would need data from other years to reduce the high bias inherent in data spanning only a single years’ worth of sales. The team recommends use of traditional Linear Regression; while it does not have the strongest predictive power relative to Partial Least Squares, the initial model would very likely perform well against new data.

**Data Dictionary**

|  |  |  |
| --- | --- | --- |
|  | **Field** | **Description** |
| 1 | ADDRESS | The street address of the property as listed on the Sales File. Coop sales include the apartment number in the address field. |
| 2 | APARTMENT NUMBER | The apartment number of the property (if applicable) |
| 3 | BLOCK | The block number (up to 5 digits) |
| 4 | BOROUGH | A digit code for the borough the property is located in. In order, these are: (1) Manhattan, (2) Bronx, (3) Brooklyn, (4) Queens, and (5) Staten Island. |
| 5/6 | BUILDING CLASS (AT PRESENT/ TIME OF SALE) | The Building Classification is used to describe a property’s constructive use. The first position of the Building Class is a letter that is used to describe a general class of properties (for example “A” signifies one-family homes, “O” signifies office buildings. “R” signifies condominiums). The second position, a number, adds more specific information about the property’s use or construction style (using our previous examples “A0” is a Cape Cod style one family home, “O4” is a tower type office building and “R5” is a commercial condominium unit). The term ‘Building Class’ as used by the Department of Finance is interchangeable with the term Building Code as used by the Department of Buildings.  Refer to code here: https://www1.nyc.gov/assets/finance/jump/hlpbldgcode.html |
| 7 | BUILDING CLASS CATEGORY | Refer to code here: https://www1.nyc.gov/assets/finance/jump/hlpbldgcode.html |
| 8 | COMMERCIAL UNITS | The number of commercial units at the listed property. |
| 9 | EASE-MENT | An easement is a right, such as a right of way, which allows an entity to make limited use of another’s real property. For example: MTA railroad tracks that run across a portion of another property. |
| 10 | GROSS SQUARE FEET | The total area of all the floors of a building as measured from the exterior surfaces of the outside walls of the building, including the land area and space within any building or structure on the property. |
| 11 | LAND SQUARE FEET | The land area of the property listed in square feet. |
| 12 | LOT | The lot number (up to 4 digits) |
| 13 | NEIGHBORHOOD | Department of Finance assessors determine the neighborhood name in the course of valuing properties. The common name of the neighborhood is generally the same as the name Finance designates. However, there may be slight differences in neighborhood boundary lines and some sub-neighborhoods may not be included |
| 14 | RESIDENTIAL UNITS | The number of houses/apartments intended for use as a place of residence at the address. (https://www.lawinsider.com/dictionary/residential-unit) |
| 15 | SALE DATE | Date the property sold. |
| 16 | SALE PRICE | Price paid for the property. |
| 17/18 | TAX CLASS (AT PRESENT/TIME OF SALE) | Property in NYC is divided into 4 classes: Class 1: Most residential property of up to three units (family homes and small stores or offices with one or two apartments attached), and most condominiums that are not more than three stories.  Class 2: All other property that is not in Class 1 and is primarily residential (rentals, cooperatives and condominiums). Class 2 includes: • Sub-Class 2a (4 - 6 unit rental building); • Sub-Class 2b (7 - 10 unit rental building); • Sub-Class 2c (2 - 10 unit cooperative or condominium); and • Class 2 (11 units or more).  Class 3: Most utility property.  Class 4: All commercial and industrial properties, such as office, retail, factory buildings and all other properties not included in tax classes 1, 2 or 3.  SOURCE: https://www1.nyc.gov/site/finance/taxes/definitions-of-property-assessment-terms.page |
| 19 | TOTAL UNITS | The total number of units at the listed property. |
| 20 | YEAR BUILT | Year the structure on the property was built. |
| 21 | ZIP CODE | The property’s postal code |

**Sale Price**

In the dataset we use, the reported sale price of a property is codified by the column SALE PRICE. The team noted that the likely indicators of sale price were to be location and square footage To test the team’s anecdotal assumptions, we asked the question: **What is the best model to use to predict sale price?** The team looked to answer this question using the following workflow:

1. Condition the dataset to respond positively to regression models
2. Fit a linear regression model, , where SalePrice is the dependent variable
3. Analyze the results of the initial model
4. Fit alternative models
5. Select the strongest model from all

We approached this method by closely following the formal model of a linear regression problem by establishing the initial formula:

Where are predictors in the data dictionary (as depicted in the abstract), are coefficients, and is the error term.

**Data Processing**

The team completed the following operations against the dataset to prepare it for model usage:

* Feature Engineering
* Outlier Handling
* Skewness Adjustment

To do this, the following actions were completed against the dataset:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Field** | **Action** | **Comments** |
| 1 | ADDRESS | Removed, Converted to LATLONG, STREET | 1. Converted to LAT-LONG pairs to leverage potential geospatial utility using geocod.io API  2. Converted to STREET to factorize/cluster addresses |
| 2 | APARTMENT NUMBER | No Action |  |
| 3 | BLOCK | Factorized |  |
| 4 | BOROUGH | Factorized |  |
| 5/6 | BUILDING CLASS (AT PRESENT/ TIME OF SALE) | Factorized |  |
| 7 | BUILDING CLASS CATEGORY | Factorized |  |
| 8 | COMMERCIAL UNITS | No Action |  |
| 9 | EASE-MENT | Removed | All values were NULL |
| 10 | GROSS SQUARE FEET | No Action |  |
| 11 | LAND SQUARE FEET | No Action |  |
| 12 | LOT | No Action |  |
| 13 | NEIGHBORHOOD | No Action |  |
| 14 | RESIDENTIAL UNITS | No Action |  |
| 15 | SALE DATE | No Action |  |
| 16 | SALE PRICE | Log Transformation | Logarithm-transformed to reduce RMSE and improve model stability |
| 17/18 | TAX CLASS (AT PRESENT/TIME OF SALE) | No Action |  |
| 19 | TOTAL UNITS | No Action |  |
| 20 | YEAR BUILT | No Action |  |
| 21 | ZIP CODE | Factorized |  |

The most noteworthy action within the team’s preprocessing steps was the action taken against the ADDRESS field:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Field** | **Action** | **Comments** |
| 1 | ADDRESS | Removed, Converted to LATLONG, STREET | 1. Converted to LAT-LONG pairs to leverage potential geospatial utility using geocod.io API  2. Converted to STREET to factorize/cluster addresses |

The team used the Geocod.io API to ingest the addresses and return latitude-longitude float pairs for each record for experimental use against the model. However, the team ultimately decided against the usage of the pairs due to concerns regarding R’s ability to recognize the pairs as dependent on each other when all in the model should be independent. To resolve this issue whilst attempting to extract significance from the field, the team used a PowerShell script to scrape the street field from ADDRESS; both efforts were retained and implemented in a new canonical .csv for use beyond the scope of this project. While the team has collective expertise with converting pairs into Geohashes, we ultimately opted for STREET to maximize model readability and understanding. However, in the team’s testing, we found that the use of STREET resulted in an extreme increase in training time for our regression models, so we opted to not use either ADDRESS nor STREET for this project, but we have left it in here for potential use in the future.

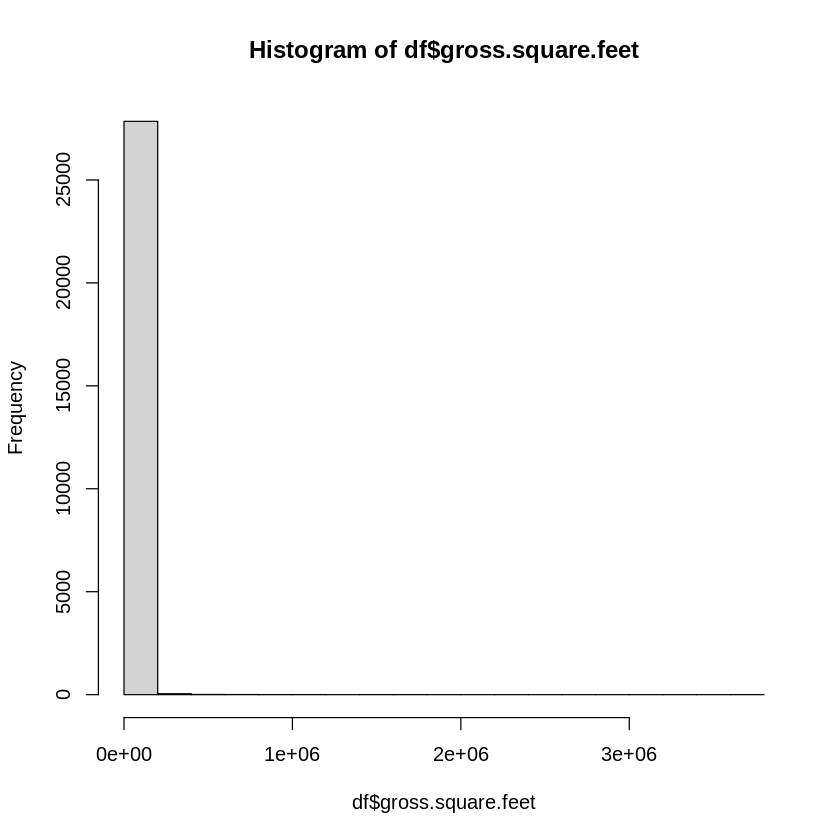
In the initial exploration, the team identified the following issues:

1. Incomplete SALE PRICE or GROSS/LAND SQUARE FEET fields
2. Nonsensical sale prices (both $0 and those uncorrelated with square footage)
3. Clear outliers
4. Skewness

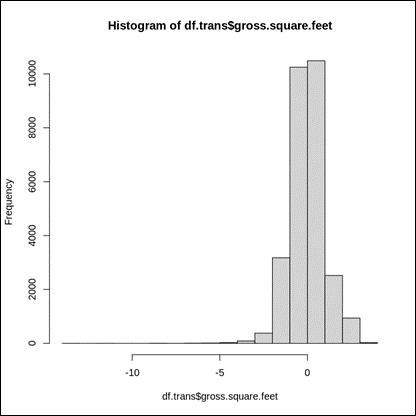
After careful deliberation, the team applied the following filters to the dataset in R:

* Eliminate incomplete observations by dropping records with NULL values in square footage and sale price.
* Reining in gross and land square feet to properties > 100 square feet.
  + This is done to remove records that are either too dirty or have no significant contribution with respect to real estate
* Using only sale prices greater than $100,000
  + This is done because price per square foot realistically would be (extremely optimistically) $1000/sq. ft in NYC.
* Log-transform the sale price to reduce RMSE

The dataset’s skewness can be clearly observed with using a histogram for gross square footage:



After performing a Box Cox transformation against the dataset, the same data elements appeared like this:



**Model Results**

The team then one-hot-encoded all categorical variables (converted to binary has/doesn’t-have predictors) for better suitability for regression. The results of all models are aggregated in the table below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Root Mean Squared Error**  **(RMSE)** | **R2** | **Mean Absolute Error**  **(MAE)** | **Runtime (≈)** |
| Linear Regression | 0.4495 | 0.7267 | 0.2885 | 24.19 sec |
| Partial Least Squares (Manual) | 0.4684 | 0.7028 | 0.3070 | 2.65 sec |
| Partial Least Squares (10 Fold CV) | 0.0074 | 0.9999 | 0.0057 | 35.20 sec |
| Regression Tree (bestFit.depth=7) | 0.1835 | 0.9544 | 0.1427 | 119.39 sec |
| Elastic Net (ELNET) | 0.6858 | 1.0000 | 0.4706 | 138.47 sec |
| MARS (Manual) | 0.4493 | 0.7270 | 0.2989 | 218.25 sec |
| MARS (10 Fold CV, bestFit.nprune=2.degree=1) | 7.67 × 10-15 | 1.0000 | 5.19 × 10-15 | 7,437.80 sec |

The team hypothesized that there exists a linear relationship between the predictors and sale price, which will allow for the fitting of a regression model. To validate this finding, we fitted a variety of both traditional linear regression models and nonlinear models to the data.

**Linear Regression**

Manual

|  |
| --- |
| Call:  lm(formula = df.lm.train$sale.price ~ ., data = df.lm.train)  Residuals:  Min 1Q Median 3Q Max  -3.8665 -0.1403 0.0474 0.2081 3.4727  Coefficients: (313 not defined because of singularities)  ---  Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  Residual standard error: 0.4214 on 18848 degrees of freedom  Multiple R-squared: 0.7671, Adjusted R-squared: 0.7586  F-statistic: 90.11 on 689 and 18848 DF, p-value: < 2.2e-16 |

For the manually trained Partial Least Squares model, it appears that only 55.81% of the variance in the dataset’s sale price is attributable to all 10 components. Furthermore, the extremely low value indicates that this is a very poor-performing model, which is an indicator of the nonlinearity or poor predictive power of the components in consideration with respect to sales price.

**Partial Least Squares**

Manual

|  |
| --- |
| Data: X dimension: 19538 1002  Y dimension: 19538 1  Fit method: kernelpls  Number of components considered: 10  TRAINING: % variance explained  1 comps 2 comps 3 comps 4 comps 5 comps 6 comps  X 14.33 21.33 27.99 35.35 41.90 45.26  df.pls.train$sale.price 42.84 49.22 54.01 57.68 60.59 64.25  7 comps 8 comps 9 comps 10 comps  X 51.49 53.73 55.18 57.00  df.pls.train$sale.price 65.68 67.41 69.11 70.35 |

The manually-trained-and-fitted partial least squares model conveys that 70.35% of the variance in the dataset’s sale price is attributable to 10 components within the model. It is interesting to note that this model performed more poorly than traditional linear regression: the RMSE is higher and the value is slightly lower. However, this doesn’t mean that the model is ***weaker*** per se. To validate this hypothesis, we then used the Caret library to complete a model fitting with a 10-fold cross validation routine:

Caret

|  |
| --- |
| Data: X dimension: 19538 1003  Y dimension: 19538 1  Fit method: oscorespls  Number of components considered: 10  TRAINING: % variance explained  1 comps 2 comps 3 comps 4 comps 5 comps 6 comps 7 comps  X 16.30 21.76 29.37 36.82 44.18 47.53 53.84  .outcome 77.36 96.36 99.08 99.64 99.82 99.93 99.95  8 comps 9 comps 10 comps  X 56.16 57.51 59.26  .outcome 99.97 99.99 99.99  41.66 sec elapsed |

The Caret and 10-fold cross-validated Partial Least Squares model conveys that 99.99% of the variance in the dataset’s sale price is attributable to 10 components within the model; furthermore, the RMSE and MAE are the smallest values of each model run. However, the team believes that this model is very likely to have been overfitted to the data, and may not perform as well against new data or other predictors.

**Tree-Based Regression**

|  |
| --- |
| 19538 samples  1003 predictor  No pre-processing  Resampling: Cross-Validated (10 fold)  Summary of sample sizes: 17585, 17584, 17583, 17584, 17584, 17584, ...  Resampling results across tuning parameters:  maxdepth RMSE Rsquared MAE  1 0.5704752 0.5574201 0.4329264  2 0.4442453 0.7318505 0.3031987  3 0.3355257 0.8470413 0.2576659  4 0.2888410 0.8867612 0.2145946  5 0.2471529 0.9171669 0.1682136  6 0.2130130 0.9385780 0.1596057  **7 0.1835312 0.9543689 0.1427259**  8 0.1835312 0.9543689 0.1427259  9 0.1835312 0.9543689 0.1427259  10 0.1835312 0.9543689 0.1427259  11 0.1835312 0.9543689 0.1427259  12 0.1835312 0.9543689 0.1427259  13 0.1835312 0.9543689 0.1427259  14 0.1835312 0.9543689 0.1427259  15 0.1835312 0.9543689 0.1427259  16 0.1835312 0.9543689 0.1427259  17 0.1835312 0.9543689 0.1427259  18 0.1835312 0.9543689 0.1427259  19 0.1835312 0.9543689 0.1427259  20 0.1835312 0.9543689 0.1427259  RMSE was used to select the optimal model using the smallest value.  The final value used for the model was maxdepth = 7.  111.66 sec elapsed |

The fitting of tree-based regression model to the dataset found that a depth of 7 was optimal by minimizing the Root-Mean-Squared Error.

The model performance is stronger than the Manual Partial Least Squares model but not as strong as the Caret-based PLS model, with an of approximately 0.95.

**Elastic Net (ELNET)**

|  |
| --- |
|  |

The ELNET model reported an of 1 with a fraction of 0.2 and a lambda of 0.001, with a relatively high RMSE and MAE; these factors together illustrate that the ELNET would not perform well against new data and is likely to be overfit.

Introducing bias to the data by adding additional predictors may improve the performance of this model in the future, particularly through the use of geospatial data such as latitude/longitude, geohashes, or the street the property resides on through one-hot-encod

**Multivariate Adaptive Regression Spline (MARS)**

Manual

|  |
| --- |
| Data: X dimension: 19538 1002  Y dimension: 19538 1  Fit method: kernelpls  Number of components considered: 10  TRAINING: % variance explained  1 comps 2 comps 3 comps 4 comps 5 comps 6 comps  X 14.33 21.33 27.99 35.35 41.90 45.26  df.pls.train$sale.price 42.84 49.22 54.01 57.68 60.59 64.25  7 comps 8 comps 9 comps 10 comps  X 51.49 53.73 55.18 57.00  df.pls.train$sale.price 65.68 67.41 69.11 70.35 |

The manually-trained-and-fitted partial least squares model conveys that 70.35% of the variance in the dataset’s sale price is attributable to 10 components within the model. It is interesting to note that this model performed more poorly than traditional linear regression: the RMSE is higher and the value is slightly lower. However, this doesn’t mean that the model is ***weaker*** per se. To validate this hypothesis, we then used the Caret library to complete a model fitting with a 10-fold cross validation routine:

Caret

|  |
| --- |
| Data: X dimension: 19538 1003  Y dimension: 19538 1  Fit method: oscorespls  Number of components considered: 10  TRAINING: % variance explained  1 comps 2 comps 3 comps 4 comps 5 comps 6 comps 7 comps  X 16.30 21.76 29.37 36.82 44.18 47.53 53.84  .outcome 77.36 96.36 99.08 99.64 99.82 99.93 99.95  8 comps 9 comps 10 comps  X 56.16 57.51 59.26  .outcome 99.97 99.99 99.99  41.66 sec elapsed |

The Caret and 10-fold cross-validated Partial Least Squares model conveys that 99.99% of the variance in the dataset’s sale price is attributable to 10 components within the model; furthermore, the RMSE and MAE are the smallest values of each model run. However, the team believes that this model is very likely to have been overfitted to the data, and may not perform as well against new data or other predictors.**Conclusion**

After conducting initial exploratory analysis, the team transformed predictors, removed outliers, performed a Box Cox transformation, and then used the filtered data to fit multiple models, both linear and nonlinear. Based on the results of our models, the team discovered that traditional linear models resulted in strong predictive power with respect to sale price, whilst nonlinear models such as ELNET and Tree-based regression resulted in weaker predictive power. A key drawback to this model is the lack of data beyond 2017, which removes the ability for the model to leverage the aspect of time series-based data to predict/classify sales on seasonality grounds. While the trained models provide useful insight on the nature of the real estate market in New York City, it is very likely that the models possess a high degree of bias as a result of both the lack of pre/post 2017 sales data and other confounding variables not accounted for that may contribute to sales price, such as economic/rental/purchase regulations or property condition/building material. A way to improve data suitability model would be to take advantage of street-based data and how neighboring properties may affect property values, as the team was unable to properly utilize the feature-engineered street predictor in a way that would introduce variance in the model without sacrificing a significant amount of time in training (from merely 20 minutes to over 12 hours). As such, the models generated from this data serve best as an initial template for ***further*** iterative development with respect to real estate price predictions, using traditional Linear Regression as a base model. While Partial Least Squares resulted in the minimal amount of errors, the team suspects that in this case, the best choice is the most simple one: a general linear model will perform better against new data versus the former since it will be resistant to both bias and variance.

# Bibliography

City of New York. (2020, November 25). *NYC Property Sales*. Retrieved from Kaggle: https://www.kaggle.com/new-york-city/nyc-property-sales

geocodio. (2020, November 25). *geocodio*. Retrieved from geocodio: https://www.geocod.io/