Charles Hwang
Rachel Jordan
STAT 470
Fall 2022

Report: Determining a Resource-Efficient Model for Residential Real Estate Valuation

Abstract

The field of real estate assessment has evolved in the past several decades with the rapid rise of new technology and statistical methods. Ongoing innovations in computing and predictive analytics allow public assessor's offices to more accurately and efficiently appraise real estate using methods that can capture widespread geographic areas rather than individual properties. Interpretations of these methods also provide additional context and information on possible reasons behind underlying trends that were not previously available. Finally, the development of these methods also allows data to be visualized in new ways with spatial features. This report analyzes data from the City of Milwaukee and attempts to build a model to predict property values with several variables.

Section 1: Introduction

The project we chose for the Statistical Consultation course is Project 7: Determining a Resource-Efficient Model for Residential Real Estate Valuation. The basis of the course is to complete a semester-long statistical project with the guidance of an external vendor, or "client", who presents some real-world problem and desires that we use various statistical methods to solve it by our interpretation. The final deliverables include the code containing all analysis performed on the datasets provided and this report.

The motivation for choosing this particular project is that it appeared to have a high degree of applicability and use in the real world. Many of the other projects provided were similar to each other in topic as they were all science-related (some had the same client). The nature of the problem also ensured the datasets would have large sample sizes (namely, a high rows-to-columns ratio) which would be easier to analyze.

Section 1.1: The Client

Our client for this project is Colin Williams, Real Estate Valuation Modeler within the City of Milwaukee Assessor's Office (CMAO). We were appreciative because Colin is a recent Loyola alumnus and has also worked with students from this course in previous years, and thus is knowledgeable about the course outline and objectives as well as university dates and timelines of the semester. This was helpful in determining check-in dates and frequency and allowed us to work on the project with everything that was needed. For example, some other groups were not able to obtain certain data from their client during the first half of the semester which may have initially limited the amount of analysis they were able to perform.

CMAO conducts assessments of real estate values of properties across the city of Milwaukee, Wisconsin as required by law and various state and local statutes. As there is a lot of real estate across the city, CMAO uses various "mass appraisal techniques" to expedite the process. These can be helpful to save time and resources, but year-over-year changes from evolving real estate and migration trends and

aging properties may cause these to become less accurate over time. Building models with recent data can be helpful in incorporating these trends and providing more accurate predictions.

Section 1.2: The Problem

Real estate is assessed primarily to calculate property taxes. Assessments are generated annually based on a wide variety of criteria like area, number of bathrooms and kitchens, age, quality, condition, neighborhood, etc. Most applicable jurisdictions have an assessor's office or equivalent agency responsible for overseeing the assessment process. As a publicly funded office, it is important that assessments are accurate. Our client has shared that some residents and homeowners can be frustrated at assessments that are higher than expected and cause them to owe more in property taxes.

The main goal of the project is to provide the client with a statistical model that can predict the real estate values of properties. The outline of the project describes the ideal model as using a "resource-efficient method", which it defines as "the recommended methodology balances accuracy with the amount of resources (including technology, time, data, and skill) required to implement the methodology". In other words, the model should have relatively good accuracy while using an easy-to-interpret method, as CMAO may not have the resources (time, staffing, funding, etc.) to implement more complex types of models on real-world data. This balance resulted in the objective being to provide two models: one that is relatively simple in method and practical to use, and another that is more complex but may have higher accuracy. Initially, a hybrid or "compromise" model was an objective, but it was determined such a method would likely not have higher accuracy than a simple model while still being interpretable and is not required or necessary.

The two main metrics used in the field of assessment to measure the quality of a statistical model are the assessment sales ratio (ASR) and the coefficient of dispersion (COD). It may be helpful to review commonly used prediction statistics before proceeding into the definitions of each metric. Unlike other models that use the differences of the predicted and actual values (e.g., mean squared error is the mean of the squared differences), these two metrics use the quotients of predicted and actual real estate values of a property. The ASR is the median of these quotients, and a general rule of thumb is that ASR greater than 0.9 and less than 1.1 ($0.9 < ASR < 1.1$) are good. The median (rather than the mean) is taken to avoid outliers and influential observations having undue influence on the model which can be common in real estate values.

The COD measures uniformity of values and is derived by dividing the quotients by the ASR and taking the mean. This essentially standardizes the quotients using the ASR and results in a number between 0 and 1, but in the field of assessment, this value is usually multiplied by 100. A COD lower than 15 is considered good. Further information on how this is computed can be found in the corresponding code.

Section 2: Data Wrangling

Code for this section is located in Consulting EDA.Rmd, Fixing SFYI.Rmd, and Archive/GLM Modeling.Rmd.

The data was provided in five separate data frames: sales_data, residential_data, property_values, sfyi_data, and location_data. Sales_data is a data frame where the unit of observation is the sale of a property. This data set contains many variables describing both the sales and the characteristics of the properties; examples of variables included in this data frame are sale price, sale date, square footage,

number of bathrooms, and year built. Residential_data is a data frame consisting of all the relevant properties in Milwaukee where the unit of observation is the property. Similarly to sales_data, this data frame contains variables relating to characteristics of each property; for example, year built and square footage. Property_values is a data frame where each observation is a valuation, and the variables are the property identification numbers and valuations. Sfyi_data is a data frame containing special features of the properties where the unit of observation is the special feature (e.g., greenhouse, pool, patio, garage, etc.) and it contains variables such as type of special feature, quantity of the special feature, and quality/condition of the special feature. We were instructed to train our models on sales_data, using sale price as the response. Finally, location_data is a data frame containing location information such as neighborhood, neighborhood cluster, latitude, longitude, and geotract.

To create our testing and training data sets, we first had to join sales_data, property_values, sfyi_data, and location_data, but before beginning this process we had to wrangle property_values and sfyi_data individually. Both were in long form and needed to be pivoted wider for merging and analysis. Using the pivot_wider() function from the dplyr R package easily allowed us to restructure the property_values data frame; however, restructuring sfyi_data was much more difficult and required a more sophisticated application of the pivot_wider() function. With permission from Dr. Banerjee, fabricated example data was posted to the StackOverflow website (https://stackoverflow.com/), and members of the StackOverflow community were able to provide a coding solution that allowed us to complete the pivot. Due to the complex nature of the pivot and the desired final data frame structure, the quality and condition of the special features were not retained. The final data frame was structured so that the unit of observation was a property, and the quantity of each special feature was an individual variable. For example, the entry for one house might indicate that it has 2 parking structures, one greenhouse, and one pool. The cells that were coded as NA by the pivot_wider() function were recoded to be 0, as properties without values for each special feature once pivoted were properties that did not have the special feature. Once the property_values and sfyi_data were restructured, they were merged with location_data and then with the sales_data (itself restructured to contain only the latest sale price for each property that occurred during the year 2021) to create our final training data set.

The data was further wrangled as follows. Each categorical variable that was represented with numbers (e.g. neighborhood) was coded as a factor. Categorical variables with a natural order (e.g. quality) were appropriately labeled and leveled. Parcels sold as vacant (aka vacant lots) were dropped because these cases were not useful for home valuations and there were only 67, or 0.3% of our data. Parcels where the variable xrImprovedStatus was not 2 were also dropped as these were not houses; dropping those cases allowed us to drop the whole variable, as all remaining properties had a value of 2 for this variable. The dropped properties represented a very small proportion of our data (110 observations out of over 21,000). The variable percent airconditioned was recoded to a categorical variable with the categories “Partial”, “Full”, and “None”, as most properties were either 0% airconditioned or 100% airconditioned, with only a few appearing in between these two values. Properties that were sold for less than $1500 were dropped, as lower prices are not plausible values and would confuse our model. Similarly, properties with a “year built” before 1700 CE were dropped, as they represent impossible dates based on the history of development in Milwaukee. Minimal manipulation of data (renaming the special features columns to make them more intuitive) occurred in MS Excel. The data were then split into training and testing sets in R using an 80/20 split.

Section 3: Exploratory Data Analysis (EDA)

Section 3.1: General EDA

Code for this section is located in Consulting EDA.Rmd.

Prior to beginning the modeling process, we performed general exploratory data analysis to identify important patterns and relationships between variables.

First, we simply examined the distribution of the response variable, LastSalePrice (Figure 1).

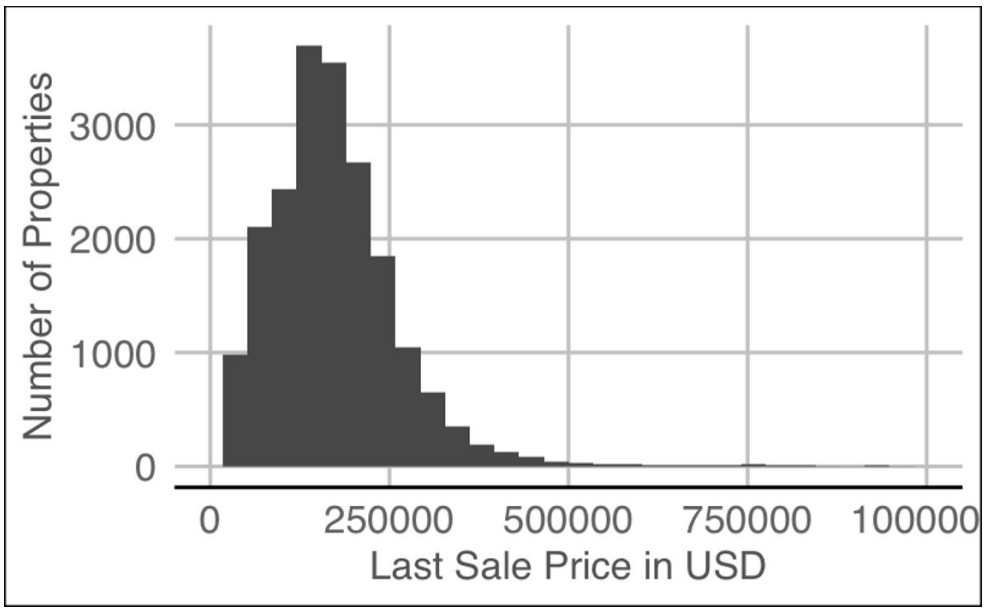


Figure 1: Histogram of Properties by Last Sale Price in USD

Figure 1 (restricted to a maximum of $1,000,000) indicates that most properties lie below $250,000 in value. Generating a five-number summary revealed a minimum last sale price of $1,500 (as properties under this amount had been removed during data cleaning), a Q1 of $115,000, a Q3 of $216,500, and a maximum of just over $2,000,000. The response clearly deviates from normality, as the existence of homes at higher price points results in a mean higher than the median; therefore, the distribution is skewed right. This distribution greatly informed our modeling choices, as will be discussed in future sections.

We were also curious about the potential effect of the year a home was built. Figure 2, a scatter plot of last sale price by year built, does not show a clearly identifiable correlation; however, it does indicate that luxury homes costing above $1 million in today's currency generally did not begin to appear in Milwaukee until the early 1900s.

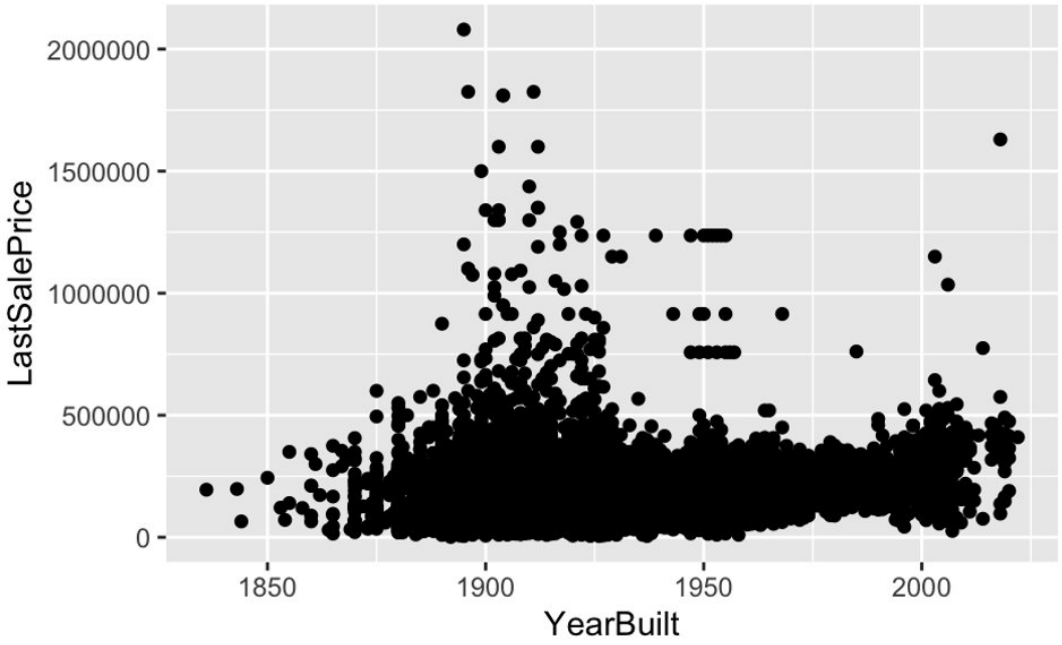


Figure 2: Scatter Plot of Last Sale Price by Year Built

Most homes in Figure 2, regardless of year built, remain under $500,000. This figure did not lead us to believe that year built was a necessary variable to include in our models. The correlation between last sale price and year built (as seen in Figure 3) calculated to be –0.08, or barely negatively correlated, a result which strengthened our conceptualization of year built as a less important variable.

Figure 3: Correlation Plot of Numerical Variables

The correlation plot generated for numerical variables is shown in Figure 3. The strongest correlations are positive; both the number of full baths and the last sale price are relatively strongly correlated with

the total finished area of the home. These findings make intuitive sense, as larger homes do generally tend to have more rooms and higher sale prices. This figure supported our intuition to prioritize total finished area and number of full baths as variables to include in our models.

Additionally, we wanted to explore the impact of neighborhood on last sale price (Figure 4), as location is intuitively a very important aspect of sale price. This figure is shown below.

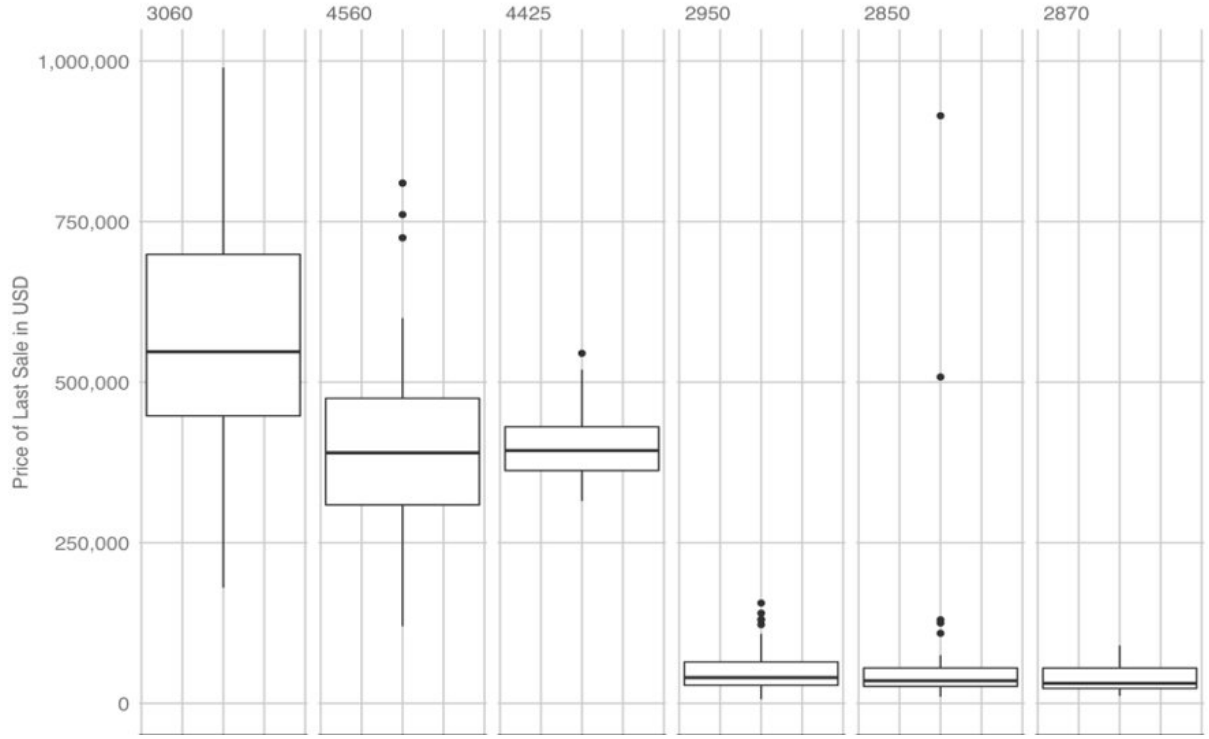


Figure 4: Boxplots of Last Sale Price in Neighborhoods with Lowest Median Sale Price and Highest Median Sale Price

The three boxplots on the left-hand side represent sale prices from the three neighborhoods with the highest median sale prices, neighborhoods 3060, 4560, and 4425. The boxplots on the right-hand side represent the last sale prices of homes in the three Milwaukee neighborhoods (2950, 2850, and 2870) with the lowest median sale prices. As the boxplots on the left- and right-hand sides do not overlap, it is clear that there can be drastic differences in sale prices between neighborhoods. Incorporating location data such as neighborhood in our model was prioritized based on these findings. The spatial nature of our data will be further explored in section 3.2 of this report.

Section 3.2: Spatial EDA

Code for this section is located in Spatial Maps.Rmd.

Since this dataset consists of real estate data with location variables like latitudinal and longitudinal coordinates, a logical step was to create some maps with residential_data which would incorporate a spatial component to the analysis. In further visualizing the data, we used the popular ggplot2 package to plot the observations' coordinates and record the values of different variables, using distinct colors for categorical variables and a color gradient for quantitative variables. The shape of Milwaukee County was included to provide a background for the points and an outline of the county. (There are only

county maps readily available in the package rather than city maps, but approximately 99.89 percent of the area of the city of Milwaukee is in Milwaukee County, with the remaining area being unincorporated areas with no real estate subject to assessment.) Since the dataset has a large number of variables, it was difficult to choose which ones to map and which ones to include in this report, but we were able to choose 12 variables (nine categorical and three quantitative) that appeared to provide meaningful analysis. Some variables had small sample sizes for certain levels, making them difficult to see and interpret on the map, so some levels were “collapsed” to create a new level with a sufficient proportion of observations, as shared in the descriptions of the applicable maps. This process was only done for the purpose of creating these maps and has no effect on any other analyses in this report.

After reviewing the variables, it seemed logical to map the NeighborhoodCluster variable first to see exactly where each of the 16 clusters were. CMAO defines several “neighborhoods” within which the real estate have characteristics similar to one another (which are not necessarily the same as the cultural definition of “neighborhood” that we commonly think about) as part of the mass appraisal techniques used to conserve time and resources. The “neighborhood cluster” is a similar variable with several neighborhoods in each cluster. We can see the different clusters in Figure 5a. Although the legend labels the clusters as 1 through 16, there does not appear to be a defined order of clusters and the numbers are only labels for classification purposes. There appear to be clusters for the downtown and outer areas and the area where Highway 59 can be seen by the absence of properties just south of the downtown area and north of neighborhood cluster #4.

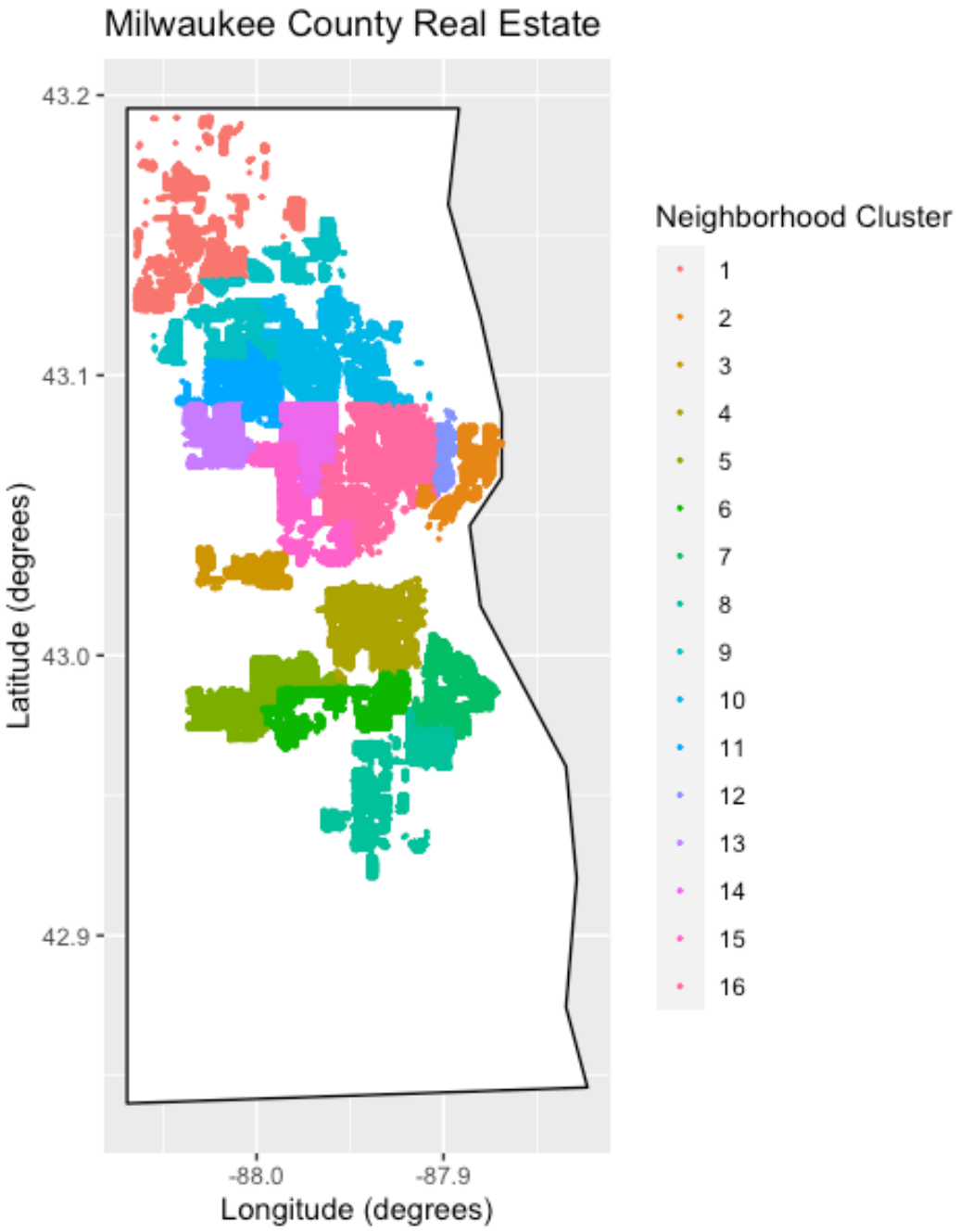


Figure 5a: Map of Milwaukee County Real Estate by Neighborhood Cluster

Proceeding in order of appearance in the dataset, the BldType variable showed the "type" of each building. Since there were many different types that only had a small proportion of observations, it was impractical to visualize all of them in the initial map and the legend was also unreasonably long. Thus, we took the five most popular types and collapsed the remaining types into an "Other" level. The map can be found in Figure 5b. We can see the outer areas tend to have more ranch- and Cape Cod-style buildings while there are more Milwaukee bungalows and residential office spaces as one moves closer

to downtown. This makes sense intuitively as we would expect a greater proportion of office spaces to be in the downtown area.

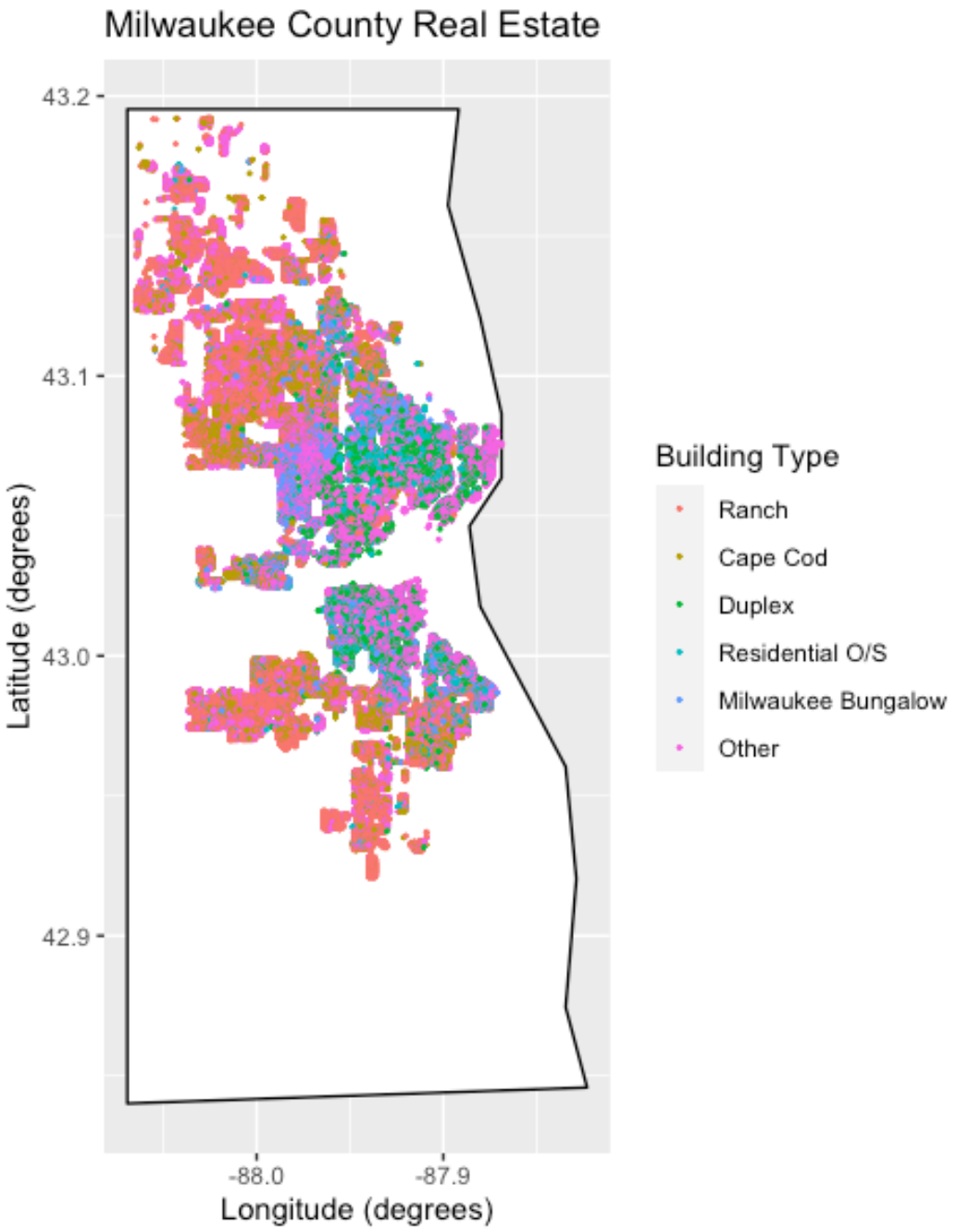


Figure 5b: Map of Milwaukee County Real Estate by Building Type

The next variable we mapped was the PhysicalCondition variable, which is an assessor's rating of the current condition of the property at the time of the appraisal. The variable in the dataset has eight levels (excellent, very good, good, average, fair, poor, very poor, and unsound), but after an initial mapping we saw it was necessary to collapse these into just three levels to retain interpretability. Even after this, we can see in Figure 5c that most properties have an "average" physical condition rating. Dr. Banerjee

advised us that the variable itself may not be contributing much to a fitted model, as it does not provide much meaningful information due to the “average” level having such a high proportion of observations. If the “average” level were split into two or more levels or if properties that were marginally rated “average” were instead pushed into a neighboring level (“good” or “fair”), this could result in the variable having more influence in the model.

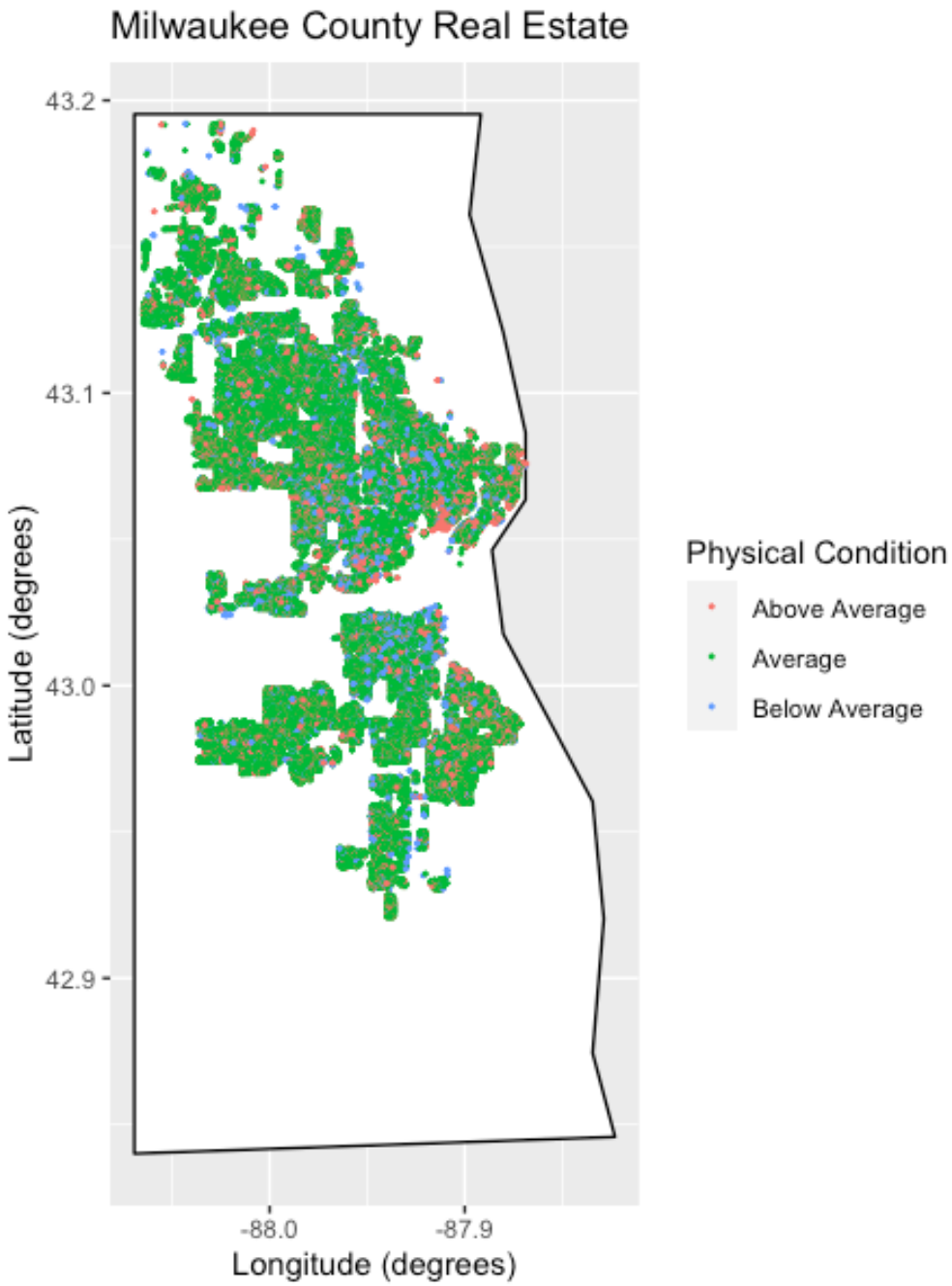


Figure 5c: Map of Milwaukee County Real Estate by Physical Condition

The Quality variable is different than the PhysicalCondition variable in that it assesses the overall building construction rather than its current condition. The variable had 18 levels ranging from “AA+” to

"E-" (specifically, "AA", "A", "B", "C", "D", and "E" succeeded by "+", no symbol, or "-"), but we again saw after the first mapping that reducing the variable to just three levels was needed to practically visualize the data. Like the PhysicalCondition variable, we can see in Figure 5d that most properties were given the quality rating of "C". Some properties in and around the downtown area have a lower rating, and we can also see a notable cluster of properties in the eastern part of the downtown area along Lake Michigan have a rating of "C+" or higher.

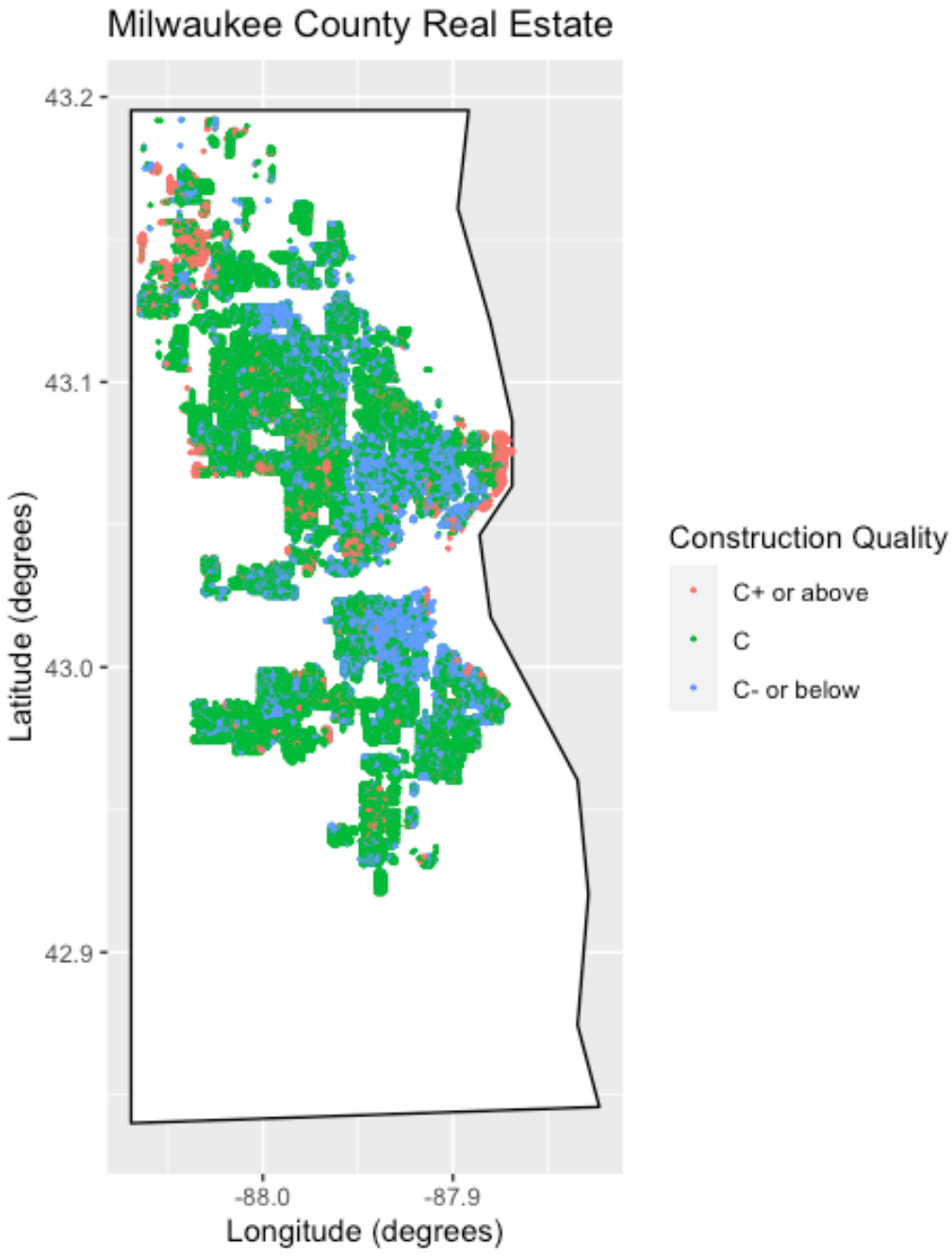


Figure 5d: Map of Milwaukee County Real Estate by Construction Quality

The YearBuilt variable is the first quantitative variable we mapped. Concurring with research done prior to these analyses, we can see in Figure 5e that most real estate in the city of Milwaukee is very old. The consensus shading appears to fall around the end of World War II when significant economic revitalization and development occurred across the United States. There also appear to be some more recently built properties at the turn of the 21st century just west of downtown, indicating further expansion and development that has been occurring over the past few decades.

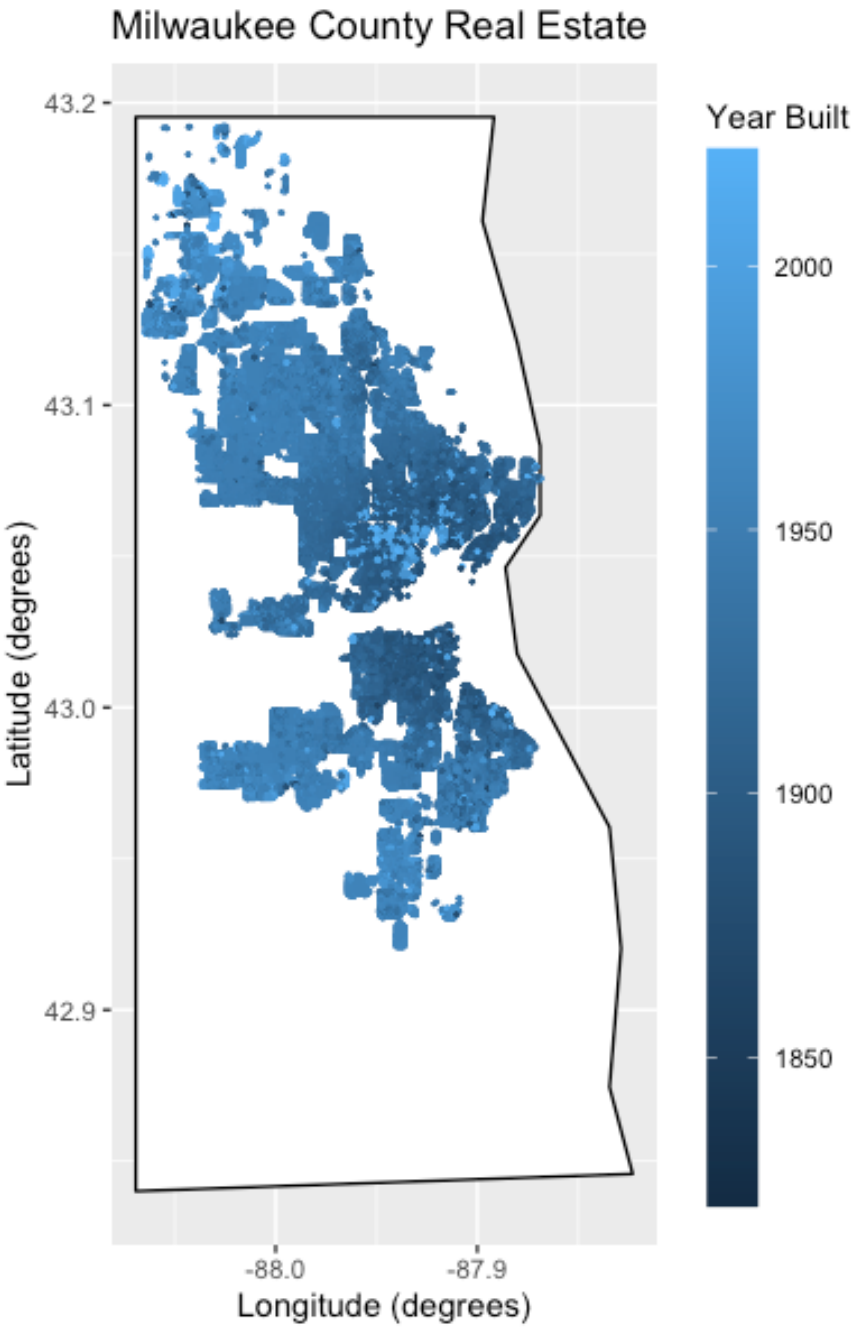


Figure 5e: Map of Milwaukee County Real Estate by Year Built

Intuitively, the FullBath, HalfBath, and Kitchen variables are somewhat correlated with each other. As discrete variables, we chose to convert them into categorical variables only for mapping to aid visualization as the ggplot() function would otherwise consider them quantitative and use a gradient scale. Most properties had anywhere between 0 and 7 bathrooms, and there was also one property each with 10 and 13 bathrooms, which would not be easily visualized if mapped as is. Thus, after collapsing all values above 3 to a "3+" level, we can see in Figure 5f that nearly all properties have either one or two bathrooms, which makes sense intuitively. Some properties around the downtown area, especially along Lake Michigan, have 3+ bathrooms, indicating larger properties.

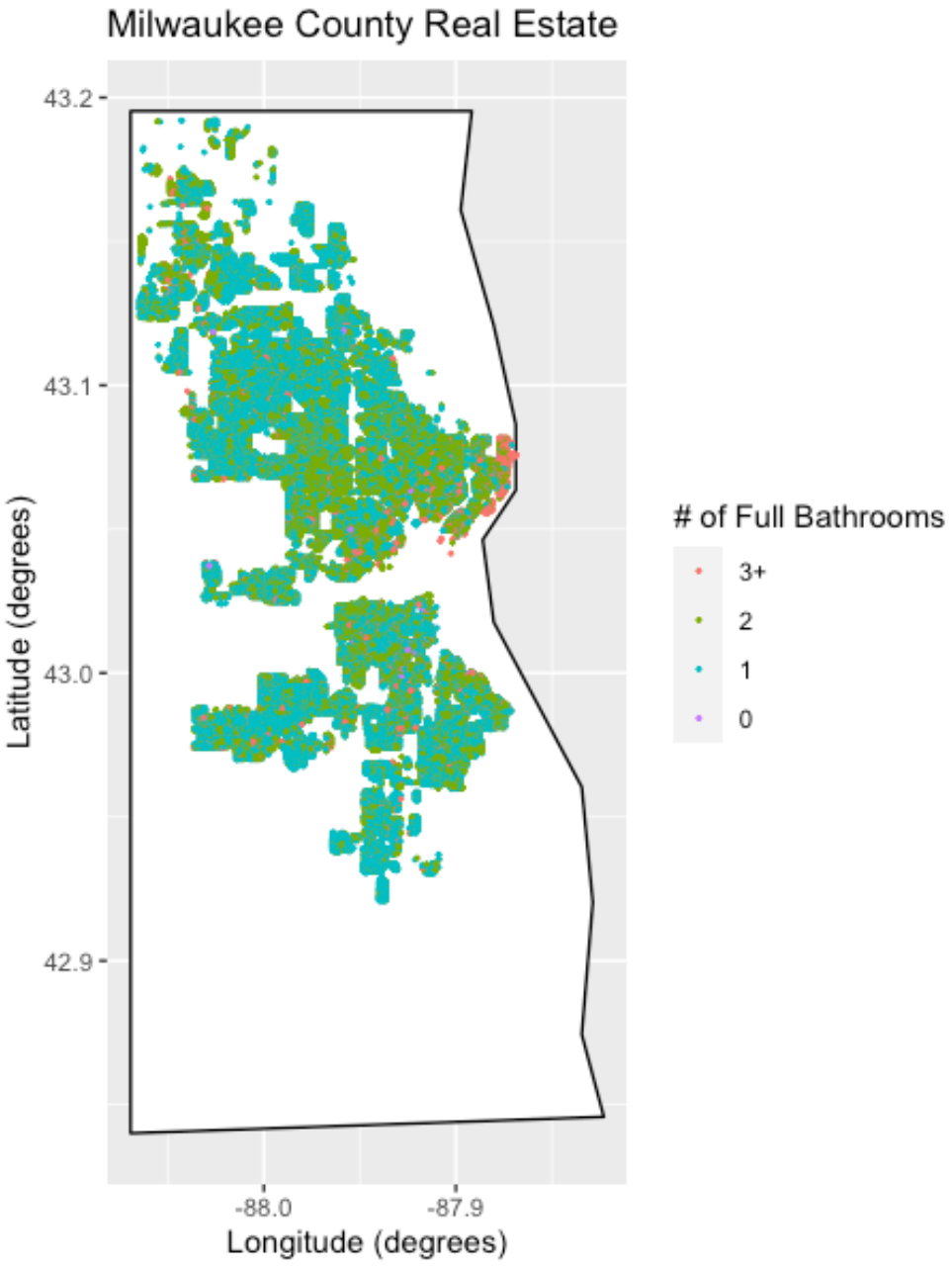


Figure 5f: Map of Milwaukee County Real Estate by Number of Full Bathrooms

We found the presence of half bathrooms in real estate in the city of Milwaukee was uncommon. However, a few properties were recorded as having two or more half bathrooms, and there was actually one property each with 6 and 9 half bathrooms. Like the FullBath variable, this necessitated that we collapse all values above 2 to a "2+" level to ensure the map would be interpretable. We can see in Figure 5g that most properties do not have a half bathroom, while some have just one. The common areas where properties had at least one half bathroom were on the outer areas of the city and again east of the downtown area along Lake Michigan.

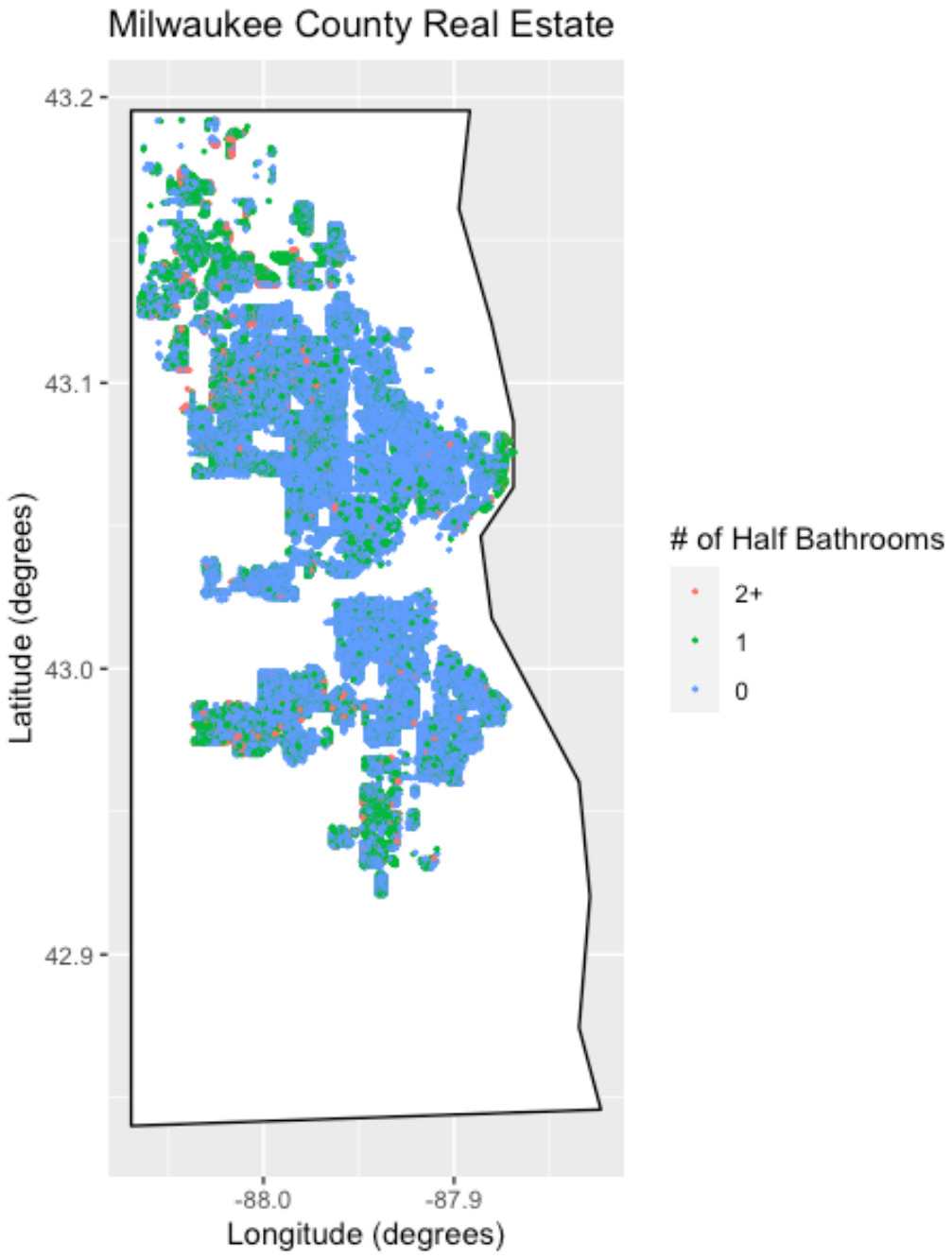


Figure 5g: Map of Milwaukee County Real Estate by Number of Half Bathrooms

After doing some preliminary analysis, we saw there were 29 properties with four kitchens and one property with five, and we collapsed these to a “3+” level for the same reason as before. We can see most properties had one kitchen, but it was surprising that many had two kitchens. We can see in Figure 5h that these properties were fairly evenly distributed across the city and each neighborhood. There does not appear to be any apparent reason there is such a large proportion of properties with two kitchens, but intuitively it can be inferred this would likely increase the assessed values of these properties.

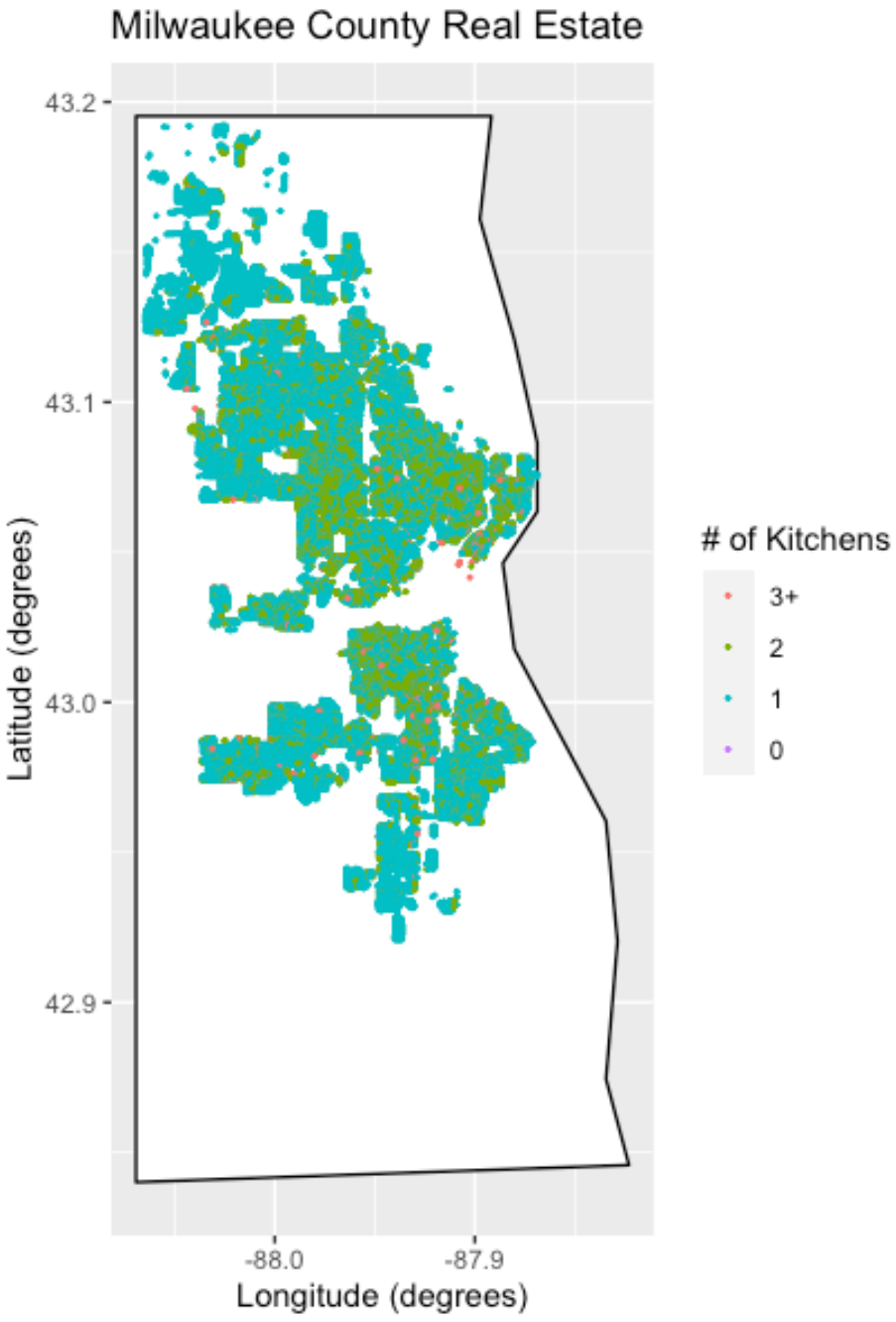


Figure 5h: Map of Milwaukee County Real Estate by Number of Kitchens

As mentioned in Section 2, the PercentAirConditoned variable was recoded to a categorical variable to measure whether a property had full, partial, or no air conditioning. The proportions of properties in each level were sufficient to provide a practical visualization which can be found in Figure 5j. We can see a roughly even split between properties with full air conditioning and no air conditioning. Looking back at the gradient scale map of property age in Figure 5e, we can see a lot of the properties with no air conditioning appear to be older while most of the properties built in the last half-century appear to have full air conditioning. The original PercentAirConditioned variable was calculated by using square area, and we can see it is rare but possible for properties to have "partial" air conditioning, especially if there is poor circulation or if some rooms are insulated from others.

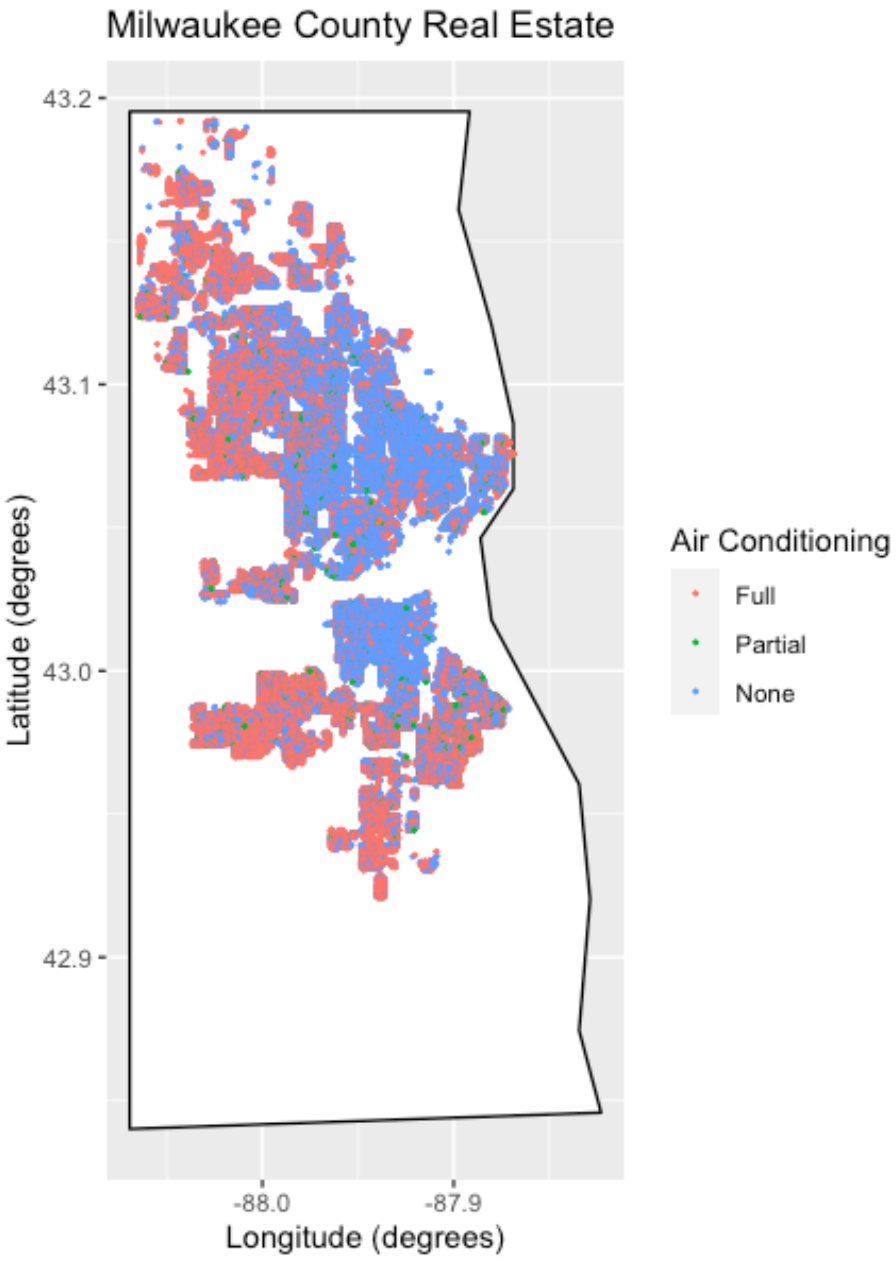


Figure 5j: Map of Milwaukee County Real Estate by Air Conditioning Status

The PrimaryWall variable measures the exterior material of a property. This variable was challenging to map as there were 14 different levels, many of which had a “moderate” number of observations—not enough for the level to be easily visible on the map, but enough to represent a considerable proportion of observations. Eventually, all but three levels were collapsed to an “Other” level for mapping. This was helpful for visualization, but we experienced some loss of information as eleven different levels were grouped together and the “Other” level had more observations than the lowest individual level (“Wood”). We can see in Figure 5k that the majority of real estate in the city of Milwaukee was constructed with aluminum/vinyl as its primary exterior wall material, with brick being a distant second.

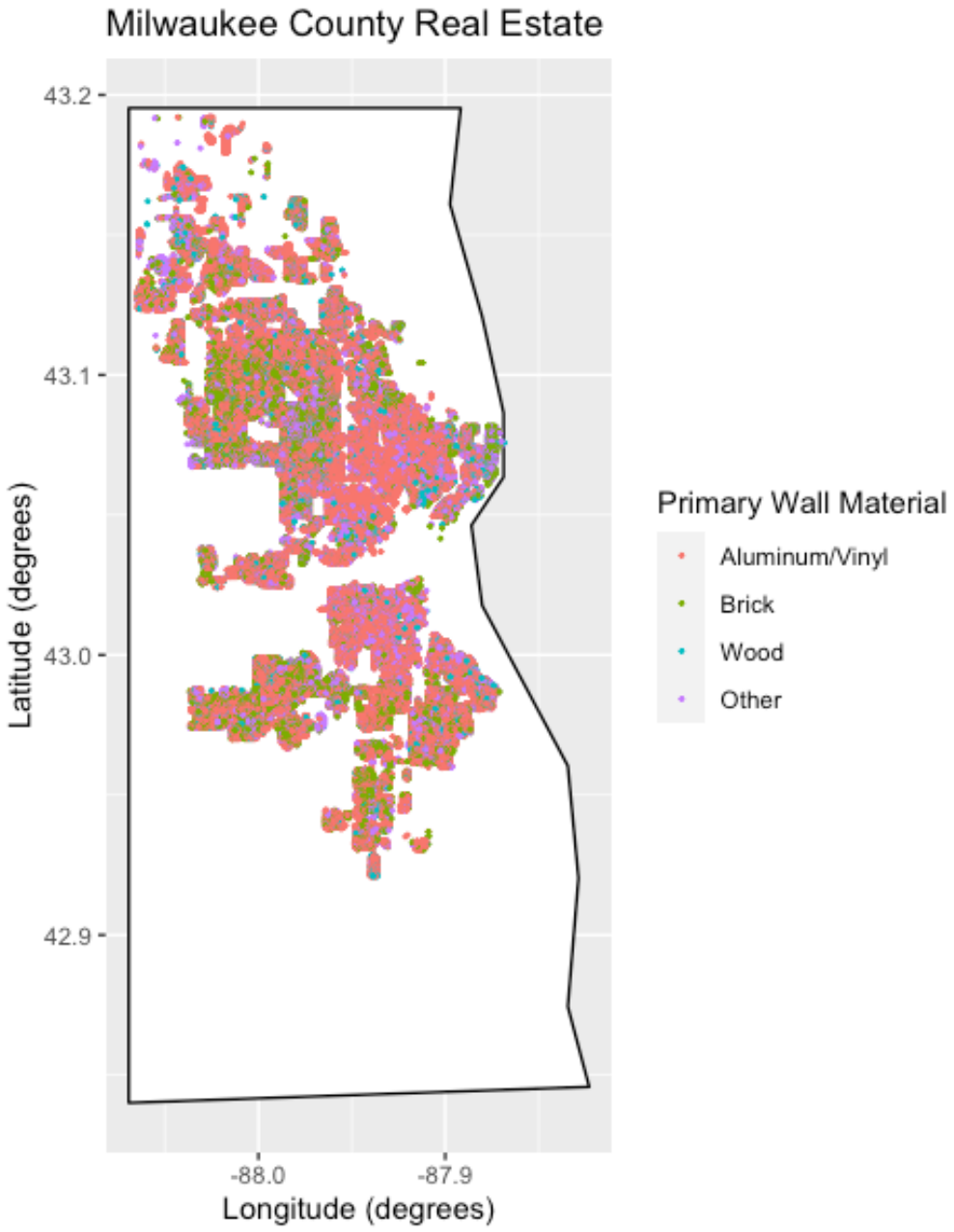


Figure 5k: Map of Milwaukee County Real Estate by Primary External Wall Material

Sale price is naturally of interest in relation to predicting assessed values. This was the only map for which the sales dataset was used, as sale price was not recorded as a variable in residential_data. Since the distribution of the variable is right-skewed due to several high-end sales, a log transformation was performed on the variable before mapping, and the updated scale can be found in Figure 5m. We can see nearly all properties were roughly in the same range of sale prices, except for some properties east of the downtown area which appear to be slightly higher and closer to the seven-figure range. There also appear to be some lower-end sales west of the downtown area.

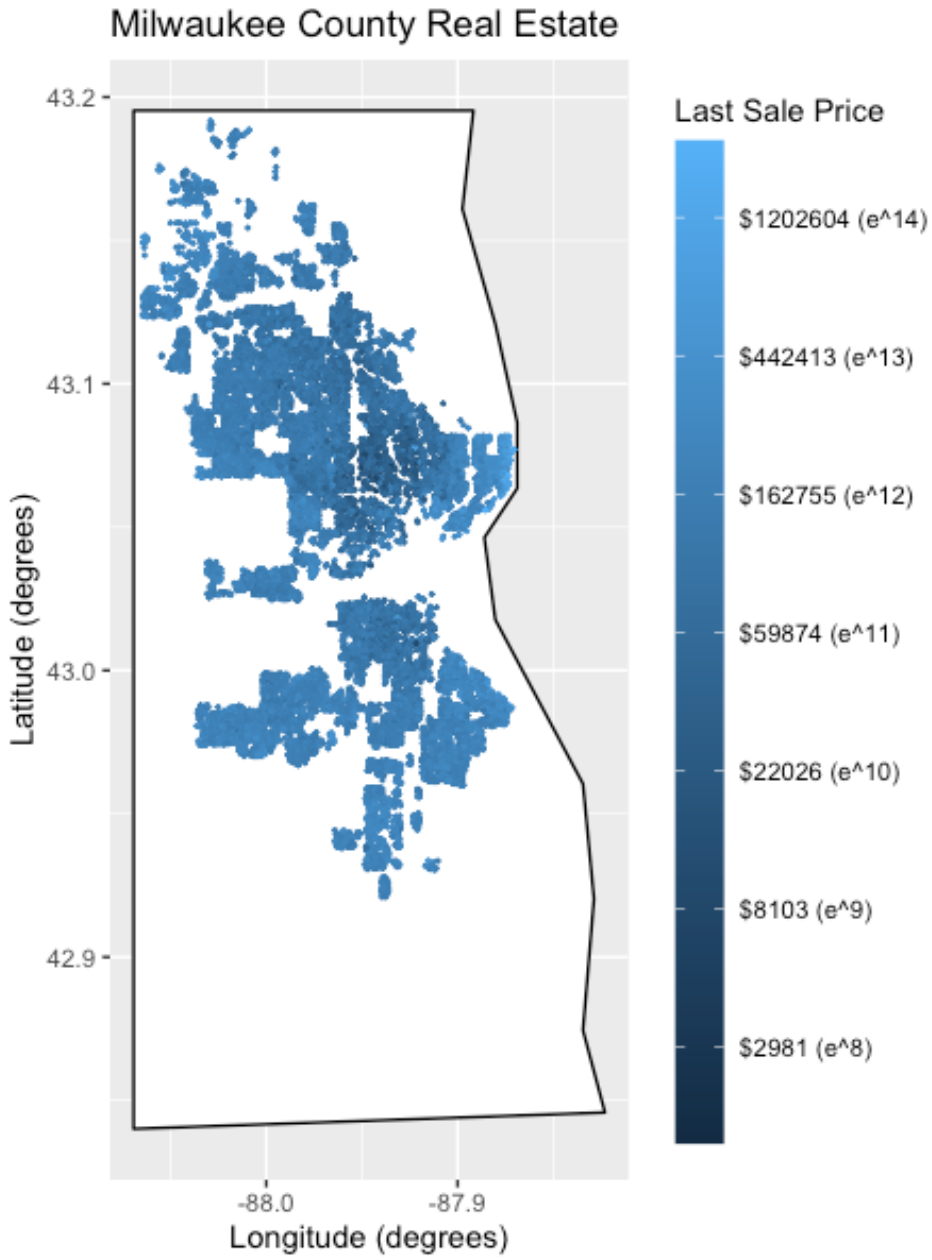


Figure 5m: Map of Milwaukee County Real Estate by Last Sale Price

Finally, the TotalArea variable was similarly right-skewed due to several mega properties, so a log transformation was performed on the variable prior to mapping. However, the distribution of this

variable is different than that of the LastSalePrice variable, and we saw the log transformation performed worse as the transformed distribution appeared to be closer to a gamma distribution rather than approximately normal. The knowledge that this variable follows a gamma distribution is later utilized in the "simple" models in Sections 4.1.1 and 4.1.2.

Even though it was a bit difficult to visualize and interpret the results, we chose to proceed with the map as there was no apparent and practical transformation that could better normalize the variable and, as this was only an observational analysis, no assumptions had been violated. A map of the log-transformed variable and updated scale can be found in Figure 5n. We can again see there is not much difference in total square area between properties, but there is some lighter shading east of the downtown area which suggests the presence of larger properties.

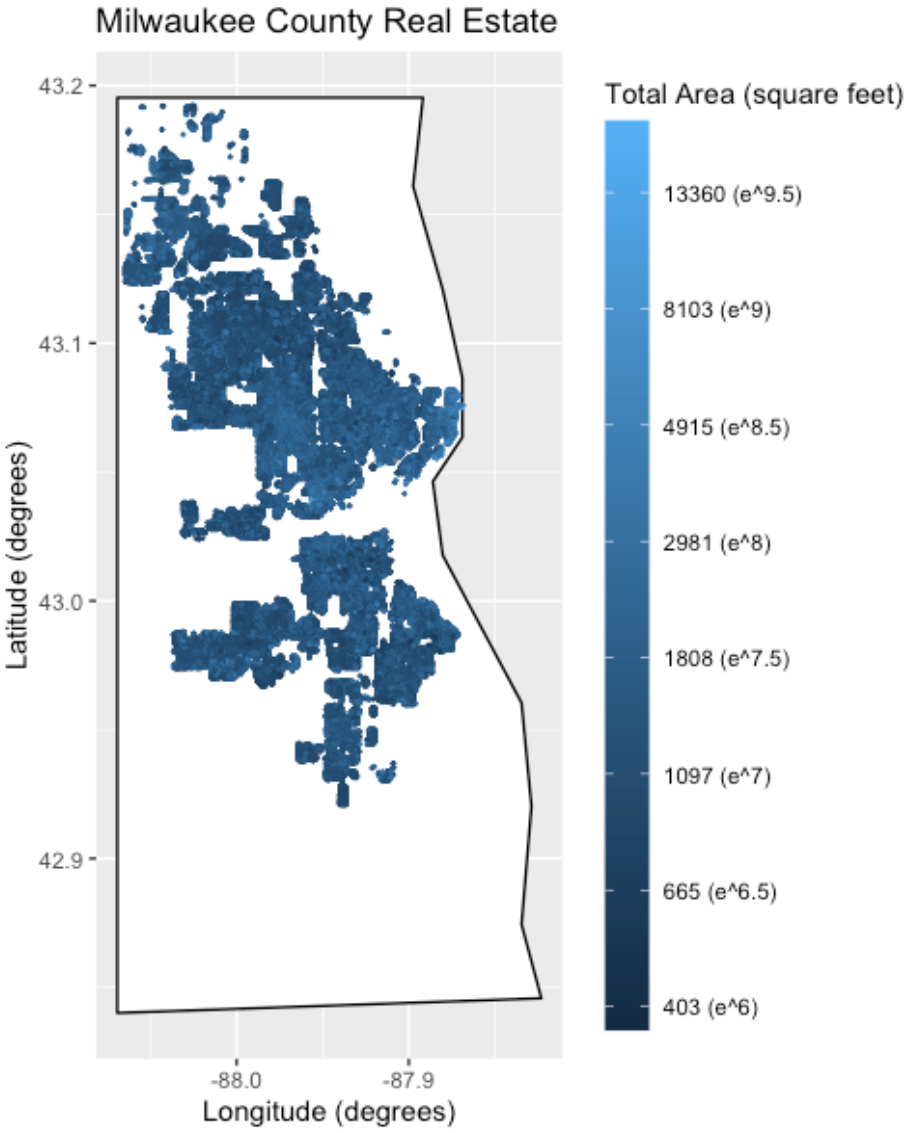


Figure 5n: Map of Milwaukee County Real Estate by Total Area (square feet)

Section 4: Our Methods

Based on the task at hand and our EDA, we decided to proceed with both a generalized linear model (GLM) and a linear model for our "simple" model and a random forest for our "complex" model. GLMs were indicated by the literature to be appropriate for real estate modeling (Deaconu et al., 2022) and allowed for the fact that our response variable was right-skewed. The selection of random forests as a modeling method is discussed in section 4.2.

Section 4.1: Simple Models

Code for this section is located in Create Test and Train.Rmd.

Prior to discussing the process for creating the "simple" model, we find it important to note that cross-validation can become an issue when modeling with categorical variables, or factors. When attempting to cross-validate our initial models, R returned an error message coming from the predict() function stating that there were "new levels" of the factor in the test data set that did not appear in the training data set. After consulting with Dr. Greg Matthews, the instructor for Loyola's Predictive Analytics course, we elected to collapse categories for the factors that were causing the error messages and recreate our test and training data sets using the collapsed categories that did allow for cross-validation.

The first variable that was causing this issue was BuildingType (e.g., mansion, bungalow, tudor, etc). We decided to remove this variable at this stage, as no building type except mansion was significant when regressing BuildingType against LastSalePrice as the sole predictor. We believe that the concept of "mansion" is still easily captured in the variables TotalFinishedArea, FullBath, HalfBath, Kitchen, and Quality, which were retained.

The second variable causing an error in the predict() function was Quality. Quality was regressed against LastSalePrice as the sole predictor, and only the higher levels of Quality were significant at alpha = 0.05. Thus, the "worse" qualities were collapsed together as "Worse". To retain information, the higher qualities were still broken out as follows: "B-", "B", and "B+" were all coded as "B", and all "A" and "AA" were collapsed similarly so that the variable contained the categories of "Worse", "B", "A", "AA".

PrimaryWall was also collapsed to group together similar wall types (e.g., "Masonry/Frame" and "PrecastMasonry" were recoded as "Masonry").

Due to collinearity issues, a singular "Parking" variable was created at this stage to capture all special features that allowed for parking on property (e.g., "garage", "frame garage", etc). Other special feature variables such as "Attic" and "RecRoom" were removed at this stage due to low counts or counts of zero.

Section 4.1.1: Gamma Family GLM using bestglm() Function

The code for this section is located in BestGLM Modeling.Rmd.

The bestglm() function is a function from the bestglm R package. This function selects the best subsets GLM model using AIC (Akaike Information Criterion) as the default information criteria (McLeod, Xu, and Lai, 2020). Our intention was to use this function prior to other attempts at model-building to hone in on the most important variables to include in the model.

To create a GLM, it is important to specify the “family” of the model. Due to the positive, continuous nature of our response variable and its right-skewed nature, we selected the Gamma family with log link, specifications which are sometimes recommended for cost data (Peraillon, 2020).

The first step to implementing the bestglm() function from the bestglm package was to change all variables to either numeric or factor variables, as required by the bestglm() function. One attempt was made to implement bestlgm using the entire training data set; however, R returned an error stating that the maximum number of parameters the function could consider was 15. To this aim, we removed special features (which were judged to be less important for inclusion than other variables based on conversations with the client) from the model which were not significant when regressed against last sale price as sole predictors. When the data set was paired down to 15 predictors, we reran the bestglm() function and obtained the model shown in Figure 6.

```
Coefficients:
                                  Estimate Std. Error t value Pr(>|t|)
(Intercept)                      3.296e+00  5.261e-01   6.266 3.81e-10 ***
PhysicalCondition2               7.027e-02  1.893e-01   0.371 0.710537
PhysicalCondition3               7.302e-02  1.672e-01   0.437 0.662368
PhysicalCondition4               2.225e-01  1.591e-01   1.398 0.162076
PhysicalCondition5               4.205e-01  1.584e-01   2.655 0.007942 **
PhysicalCondition6               5.515e-01  1.585e-01   3.479 0.000504 ***
PhysicalCondition7               7.050e-01  1.592e-01   4.429 9.53e-06 ***
PhysicalCondition8               8.977e-01  1.662e-01   5.401 6.71e-08 ***
YearBuilt                        4.028e-03  2.549e-04  15.801  < 2e-16 ***
HalfBath                         6.389e-02  9.173e-03   6.965 3.42e-12 ***
TotalFinishedArea                2.421e-04  7.753e-06  31.231  < 2e-16 ***
LandSF                           4.344e-06  1.038e-06   4.183 2.89e-05 ***
AirConditionedNone              -7.806e-02  9.974e-03  -7.826 5.35e-15 ***
AirConditionedPartial           -5.499e-02  2.418e-02  -2.274 0.022985 *
NeighborhoodCluster2            -1.390e-01  2.531e-02  -5.492 4.03e-08 ***
NeighborhoodCluster3            -3.438e-01  2.501e-02 -13.746  < 2e-16 ***
NeighborhoodCluster4            -1.576e-01  2.707e-02  -5.824 5.87e-09 ***
NeighborhoodCluster5             3.640e-01  3.698e-02   9.841  < 2e-16 ***
NeighborhoodCluster6             1.082e-01  2.588e-02   4.182 2.90e-05 ***
NeighborhoodCluster7            -3.154e-01  2.917e-02 -10.811  < 2e-16 ***
NeighborhoodCluster8             9.894e-02  2.825e-02   3.502 0.000464 ***
NeighborhoodCluster9            -6.599e-01  2.866e-02 -23.026  < 2e-16 ***
NeighborhoodCluster10            6.422e-01  3.328e-02  19.296  < 2e-16 ***
NeighborhoodCluster11            1.980e-01  3.115e-02   6.356 2.12e-10 ***
NeighborhoodCluster12           -1.237e-01  2.949e-02  -4.196 2.73e-05 ***
NeighborhoodCluster13            1.820e-01  2.501e-02   7.277 3.57e-13 ***
NeighborhoodCluster14            1.282e-01  2.642e-02   4.851 1.24e-06 ***
NeighborhoodCluster15            4.766e-01  2.742e-02  17.382  < 2e-16 ***
NeighborhoodCluster16            2.294e-01  2.587e-02   8.865  < 2e-16 ***
QualityC2                        2.234e-01  4.115e-02   5.429 5.76e-08 ***
QualityC3                        2.376e-01  1.083e-01   2.194 0.028284 *
QualityC4                       -1.658e-01  2.554e-01  -0.649 0.516150
PrimaryWallCAsphalt/Other       -7.369e-02  1.761e-02  -4.184 2.88e-05 ***
PrimaryWallCBlock                2.960e-03  5.876e-02   0.050 0.959832
PrimaryWallCBrick                8.080e-02  1.080e-02   7.480 7.81e-14 ***
PrimaryWallCFiber Cement/Hardiplank  1.064e-01  5.795e-02   1.837 0.066280 .
PrimaryWallCMasonry              6.001e-02  2.527e-02   2.375 0.017569 *
PrimaryWallCStone                1.560e-01  2.400e-02   6.501 8.20e-11 ***
PrimaryWallCStucco               1.347e-01  3.095e-02   4.351 1.36e-05 ***
PrimaryWallCWood                 3.627e-02  1.720e-02   2.108 0.035016 *
Parking                          4.375e-02  8.138e-03   5.376 7.70e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 6: Gamma Family GLM Created using bestglm() R Function

Cross-validating the GLM from Figure 6 on test data (cross-validation code can be found in Purposeful Variable Selection with Gamma GLM.Rmd) resulted in an assessment sales ratio of 1.05 and a coefficient of dispersion of 6.8. These values are each within the desirable range for these metrics; an ideal sales ratio is between 0.9 and 1.1, and an ideal COD is in the single digits. The sales ratio over 1 indicates that we are slightly overestimating sale price with this model.

Section 4.1.2: Gamma Family GLM using Purposeful Variable Selection

The code for this section is located in Purposeful Variable Selection with Gamma GLM.Rmd and LASSO and Full Model.Rmd.

Another valuable process that can be implemented to purposefully select variables for a GLM is the purposeful variable selection process presented in Agresti (2019). For clarity and ease of reference, the four steps of this process are quoted directly from Agresti (2019) below (p. 126-127):

1. Construct an initial main-effects model using explanatory variables that include the known important variables and others that show *any* evidence of being relevant when used as sole predictors (e.g., having P-value < 0.2).
2. Conduct backward elimination, keeping a variable if it is either significant at a somewhat more stringent level or shows evidence of being a relevant confounder, in the sense that the estimated effect of a key variable changes substantially when it is removed.
3. Add to the model any variables that were not included in step 1 but that are significant when adjusting for other variables in the model after step 2, since a variable may not be significantly associated with y but may make an important contribution in the presence of other variables.
4. Check for plausible interactions among variables in the model after step 3, using significance tests at conventional levels such as 0.05.

Per step 1, we regressed each variable individually against the response, dropping those which were not significant predictors at alpha = 0.2. The variables dropped at this stage were MiscSfyi, PrefabPool, Canopy, MetalShed, GreenHouse, FiberglassPool, PoolLighting, FrameGarage, and AsphaltPaving. The model resulting from step 1 was cross-validated and obtained a sales ratio of 1.05 and a COD of 6.2, similar to but slightly better than the bestglm model. Note that valuations were not considered as potential predictors when fitting the full model from step 1 because the client advised us that these were “red herrings”. Experimentally including the valuations in the model did not result in a great improvement in model metrics.

Next, step 2 was performed. A table of dropped predictors (in the order in which they were dropped) and the p-values which lead to their dropping is presented in Table 1. No significant changes in estimated effects of key variables were observed when dropping these variables.

Table 1: Variables Dropped During Backward Elimination (Step 2)

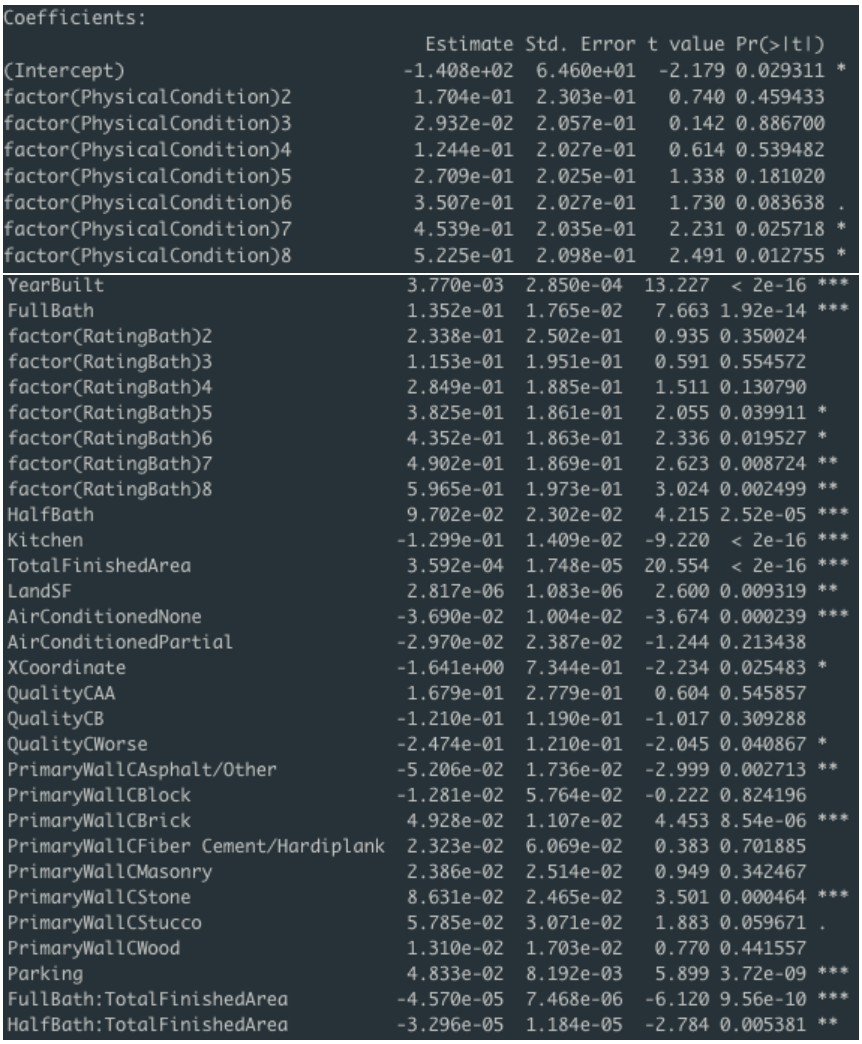
Predictor	P-Value
YCoordinate	0.948
PlasticLinedPool	0.899
ReinforcedConcreteSF	0.95
FrameShed	0.89

DivingBoard	0.822
PoolLadder	0.83
MasonAdjustment	0.20

Quality was also dropped with a p-value of 0.07 but was added back into the model when cross-validation revealed that the model containing quality performed better than the model without it (equivalent sales ratios of 1.04; COD of 6.6 vs COD of 6.7).

Per step 3, the variables dropped during step 1 were added back into the model one by one. None were significant, and none improved their model's sales ratio or COD.

Per step 4, plausible interaction terms were examined and some (PhysicalCondition*Neighborhood, FullBath*TotalFinishedArea, HalfBath*TotalFinishedArea) were significant and improved the COD to 6.4. The best model resulting from the purposeful selection process is presented below in Figure 6. Note that factor(Neighborhood) is included in the model but was removed from Figure 6 due to the large number of neighborhoods, which did not allow for a reasonably sized image to be captured. Many neighborhoods were significant, with coefficient estimates and standard errors similar to the other variables presented in Figure 7.



Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-1.408e+02	6.460e+01	-2.179	0.029311	*
factor(PhysicalCondition)2	1.704e-01	2.303e-01	0.740	0.459433	
factor(PhysicalCondition)3	2.932e-02	2.057e-01	0.142	0.886700	
factor(PhysicalCondition)4	1.244e-01	2.027e-01	0.614	0.539482	
factor(PhysicalCondition)5	2.709e-01	2.025e-01	1.338	0.181020	
factor(PhysicalCondition)6	3.507e-01	2.027e-01	1.730	0.083638	.
factor(PhysicalCondition)7	4.539e-01	2.035e-01	2.231	0.025718	*
factor(PhysicalCondition)8	5.225e-01	2.098e-01	2.491	0.012755	*
YearBuilt	3.770e-03	2.850e-04	13.227	< 2e-16	***
FullBath	1.352e-01	1.765e-02	7.663	1.92e-14	***
factor(RatingBath)2	2.338e-01	2.502e-01	0.935	0.350024	
factor(RatingBath)3	1.153e-01	1.951e-01	0.591	0.554572	
factor(RatingBath)4	2.849e-01	1.885e-01	1.511	0.130790	
factor(RatingBath)5	3.825e-01	1.861e-01	2.055	0.039911	*
factor(RatingBath)6	4.352e-01	1.863e-01	2.336	0.019527	*
factor(RatingBath)7	4.902e-01	1.869e-01	2.623	0.008724	**
factor(RatingBath)8	5.965e-01	1.973e-01	3.024	0.002499	**
HalfBath	9.702e-02	2.302e-02	4.215	2.52e-05	***
Kitchen	-1.299e-01	1.409e-02	-9.220	< 2e-16	***
TotalFinishedArea	3.592e-04	1.748e-05	20.554	< 2e-16	***
LandSF	2.817e-06	1.083e-06	2.600	0.009319	**
AirConditionedNone	-3.690e-02	1.004e-02	-3.674	0.000239	***
AirConditionedPartial	-2.970e-02	2.387e-02	-1.244	0.213438	
XCoordinate	-1.641e+00	7.344e-01	-2.234	0.025483	*
QualityCAA	1.679e-01	2.779e-01	0.604	0.545857	
QualityCB	-1.210e-01	1.190e-01	-1.017	0.309288	
QualityCWorse	-2.474e-01	1.210e-01	-2.045	0.040867	*
PrimaryWallCAsphalt/Other	-5.206e-02	1.736e-02	-2.999	0.002713	**
PrimaryWallCBlock	-1.281e-02	5.764e-02	-0.222	0.824196	
PrimaryWallCBrick	4.928e-02	1.107e-02	4.453	8.54e-06	***
PrimaryWallCFiber Cement/Hardiplank	2.323e-02	6.069e-02	0.383	0.701885	
PrimaryWallCMasonry	2.386e-02	2.514e-02	0.949	0.342467	
PrimaryWallCStone	8.631e-02	2.465e-02	3.501	0.000464	***
PrimaryWallCStucco	5.785e-02	3.071e-02	1.883	0.059671	.
PrimaryWallCWood	1.310e-02	1.703e-02	0.770	0.441557	
Parking	4.833e-02	8.192e-03	5.899	3.72e-09	***
FullBath:TotalFinishedArea	-4.570e-05	7.468e-06	-6.120	9.56e-10	***
HalfBath:TotalFinishedArea	-3.296e-05	1.184e-05	-2.784	0.005381	**

Figure 7: Gamma GLM Model from Purposeful Variable Selection

The full model was also run as a Gamma GLM; however, including more predictors did not in this case result in an improvement on the previously established best model.

Section 4.1.3: Linear Model

The code for this section is located in Purposeful Variable Selection with Gamma GLM.Rmd and LASSO and Full Model.Rmd.

To see how well it would perform on the data, a linear model was created using the variables selected during the first stage of the purposeful variable selection process. The resulting model had an assessment sales ratio of 1.04 and a COD of 5.36. Due to the excellent performance of this model, the model produced by the bestglm function was also run as a linear model, producing a sales ratio of 1.04 and a COD of 5.4. A model using all predictors except BuildingType (as BuildingType not included when collapsing variables for testing and training sets due to only one level, Mansion, being significant, information which we believe is captured by square footage) was created, which resulted in a sales ratio of 1.04 and a COD of 5.4.

The best linear model was determined to be the initial linear model. This model can be viewed in Figure 8, although the neighborhood variable was removed from the image to save space. Many neighborhoods were statistically significant at the $\alpha = 0.05$ level.

```
Call:
lm(formula = LastSalePrice ~ factor(PhysicalCondition) + factor(Neighborhood) +
    YearBuilt + FullBath + factor(RatingBath) + HalfBath + Kitchen +
    TotalFinishedArea + LandSF + AirConditioned + FrameShed +
    MasonAdjustment + PlasticLinedPool + DivingBoard + PoolLadder +
    ReinforcedConcreteSF + XCoordinate + YCoordinate + QualityC +
    PrimaryWallC + Parking, data = training)

Residuals:
    Min      1Q  Median      3Q     Max
-392905  -26386   -5060   19051 1166044

Coefficients:
                                  Estimate Std. Error t value Pr(>|t|)
(Intercept)                     -1.036e+07  9.419e+06  -1.100 0.271443
factor(PhysicalCondition)2      -2.021e+04  2.747e+04  -0.736 0.462032
factor(PhysicalCondition)3      -4.423e+04  2.454e+04  -1.802 0.071529 .
factor(PhysicalCondition)4      -3.385e+04  2.418e+04  -1.400 0.161461
factor(PhysicalCondition)5      -1.920e+04  2.416e+04  -0.795 0.426740
factor(PhysicalCondition)6      -5.695e+03  2.418e+04  -0.235 0.813826
factor(PhysicalCondition)7       1.194e+04  2.427e+04   0.492 0.622841
factor(PhysicalCondition)8       3.553e+04  2.502e+04   1.420 0.155710
YearBuilt                        5.324e+02  3.437e+01  15.489  < 2e-16 ***
FullBath                         1.200e+04  1.360e+03   8.827  < 2e-16 ***
factor(RatingBath)2              4.879e+03  2.984e+04   0.164 0.870126
factor(RatingBath)3              2.186e+04  2.327e+04   0.939 0.347640
factor(RatingBath)4              2.844e+04  2.249e+04   1.265 0.205991
factor(RatingBath)5              4.441e+04  2.220e+04   2.000 0.045484 *
factor(RatingBath)6              5.276e+04  2.223e+04   2.374 0.017631 *
factor(RatingBath)7              6.609e+04  2.229e+04   2.965 0.003032 **
factor(RatingBath)8              1.289e+05  2.353e+04   5.476 4.42e-08 ***
HalfBath                         8.076e+03  1.149e+03   7.025 2.22e-12 ***
Kitchen                         -2.950e+04  1.682e+03 -17.537  < 2e-16 ***
TotalFinishedArea                5.089e+01  1.364e+00  37.318  < 2e-16 ***
LandSF                           5.331e-01  1.299e-01   4.105 4.06e-05 ***
AirConditionedNone              -6.595e+03  1.198e+03  -5.503 3.79e-08 ***
AirConditionedPartial           -2.739e+03  2.847e+03  -0.962 0.336066
FrameShed                        8.892e+02  2.275e+03   0.391 0.695890
MasonAdjustment                  2.216e+03  1.530e+03   1.449 0.147466
PlasticLinedPool                 6.147e+03  1.460e+04   0.421 0.673691
DivingBoard                     -3.961e+04  2.427e+04  -1.632 0.102662
PoolLadder                       3.378e+04  2.272e+04   1.487 0.137039
ReinforcedConcreteSF             7.431e+04  2.376e+04   3.128 0.001763 **
XCoordinate                     -1.902e+05  8.794e+04  -2.163 0.030533 *
YCoordinate                     -1.707e+05  1.409e+05  -1.211 0.225876
QualityC2                        7.408e+04  5.564e+03  13.314  < 2e-16 ***
QualityC3                        3.804e+05  1.372e+04  27.724  < 2e-16 ***
QualityC4                        5.922e+05  3.111e+04  19.038  < 2e-16 ***
PrimaryWallCAsphalt/Other       -4.424e+03  2.071e+03  -2.136 0.032659 *
PrimaryWallCBlock               -8.485e+02  6.877e+03  -0.123 0.901800
PrimaryWallCBrick                8.492e+03  1.366e+03   6.217 5.20e-10 ***
PrimaryWallCFiber Cement/Hardiplank  3.224e+04  7.257e+03   4.442 8.96e-06 ***
PrimaryWallCMasonry              5.787e+03  3.026e+03   1.913 0.055804 .
PrimaryWallCStone                1.668e+04  2.959e+03   5.636 1.77e-08 ***
PrimaryWallCStucco               8.454e+03  3.664e+03   2.308 0.021035 *
PrimaryWallCWood                 3.350e+03  2.031e+03   1.650 0.099058 .
Parking                          6.721e+03  9.799e+02   6.859 7.21e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 8: Best Linear Model

A residual vs fitted values plot was created for this model and appears in Figure 9 below. This plot is difficult to read due to the sheer number of observations, so a random sample of 0.5% of these points was drawn and plotted (Figure 10). In Figure 10, the residuals appear to be evenly distributed around zero with no apparent pattern, indicating that a linear model is appropriate on this data. Figure 9, however, although difficult to read, gives some indication of a non-linear relationship when looking at fitted values on the lower end of the range.

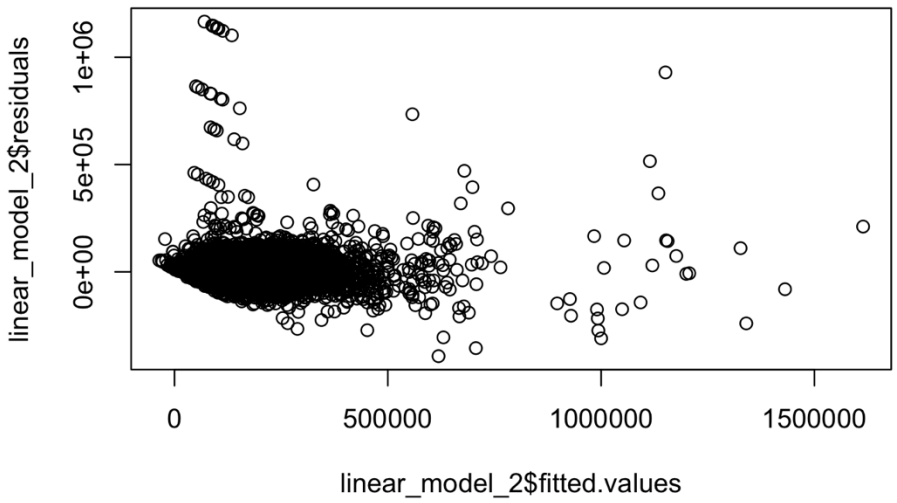


Figure 9: Residual vs Fitted Plot (Full Data Set)

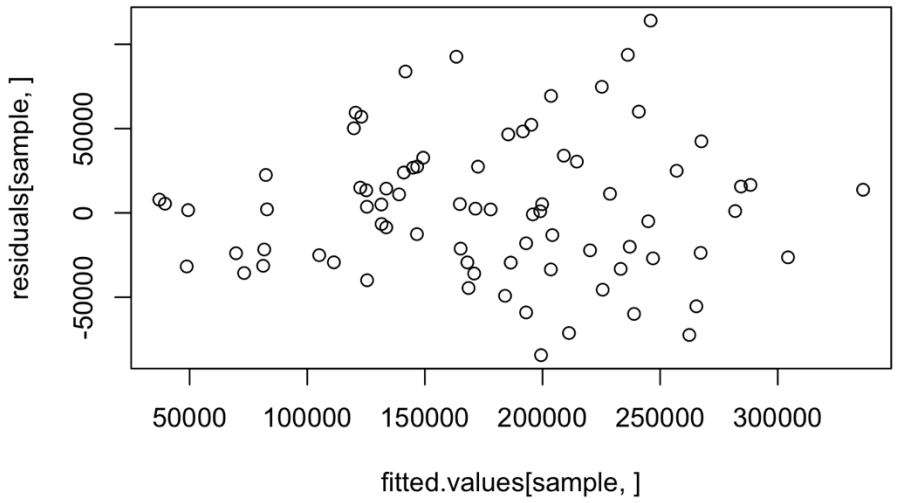


Figure 10: Residual vs Fitted Plot (Sample of Full Data Set)

Section 4.1.4: Variable Selection using LASSO

The shrinkage method LASSO was done as a variable selection method to see if it could improve the model, as LASSO has been used successfully on real estate data by others (Nowak and Smith, 2016). No

predictors were completely reduced to zero, indicating that all variables considered in the previous best model were reasonable to include. However, within the neighborhood factor, three neighborhoods were given coefficients of zero: neighborhood 2970, neighborhood 4240, and neighborhood 4420. The client may wish to examine these neighborhoods in more detail.

Section 4.1.5: Evaluating Best Simple Model

Code for this section is located in Evaluating Best Model.Rmd.

As mentioned in Section 4.1.3, it was determined that the linear model was the best of the "simple" models analyzed. The client mentioned the next steps after building a model that has acceptable metrics ($0.9 < ASR < 1.1$ and $COD < 15$) would be to see if the model performs poorly with any particular levels of variables or subsets of the data (certain neighborhoods, building styles, expensive sales, large properties, etc.). This would indicate that error in the model is potentially nonrandom and that the model may be biased toward a particular feature (or, in the field of assessment, nonuniform). This would also be evidence that building a separate model on a particular subset of data may be appropriate. The following paragraphs detail our process for finding which subsets the model performed poorly on.

Further mentioned in Section 4.1.3, we see the linear model had a ASR of 1.04 and a COD of 5.36. In determining where the model performed poorly, we calculated the ASR and COD on the applicable subset of the data and compared it to these two values while also checking whether they were outside the previously mentioned acceptable ranges. The sample size of the subset and proportion of the subset to the test set were also calculated to estimate the extensiveness of the potential nonuniformity in the model. After reviewing all variables, we identified five main subsets across four variables that the model performed poorly on.

One of the challenges encountered was determining a reasonable "cutoff" point for variables. For example, the model performed poorly with properties less than 900 square feet, but it did even worse as the threshold was lowered to 800, 700, and so on. However, as expected, the proportion of observations decreased along with the cutoff, and less properties were affected by the model's poor performance, albeit with greater severity. We can imagine the tail of a hypothetical histogram of the TotalFinishedArea variable getting thinner as we continue moving left. In most cases like this, we looked at the two metrics and the proportion of observations and made an arbitrary determination.

It is worth noting the test set is a random sample of 20 percent of the sales dataset, as mentioned in Section 2, but the sales dataset is not necessarily a random sample of the full dataset of real estate. In other words, sales of real estate may not be uniformly distributed across the city, and having a small proportion of the test set affected does not imply a similarly small proportion of real estate is affected.

Intuitively, the first variable to look at is the response variable, LastSalePrice. We found the model did quite poorly with properties sold for $20,000 or less, with ASR and COD values of 2.428144 and 53.16512, respectively. The model also did poorly with properties sold for $250,000 or more but to a lesser extent. The ASR for this subset was 0.8883501, but interestingly the COD was less than 1 at 0.1979317. The two subsets made up approximately 0.4578988 and 15.33961 percent of the test set, respectively.

The YearBuilt variable was also examined at either end, and while there did not appear to be any subsets of older properties that the model performed poorly on, real estate constructed since 2017 had ASR and COD values of 1.129211 and 27.21689, respectively. This makes sense intuitively as these properties would be hard to assess due to a lack of appraisal history and similar properties as well as being different in construction and architectural design. It is worth noting that only 0.1271941 percent (n = 5) of the test set consisted of these properties, but there are more such properties in the full sales (n = 25) and residential (n = 162) datasets. Additionally, the number of properties constructed since 2017 will logically increase over time, so we do not believe the small sample size invalidates evidence pointing to a separate model. We reiterate this in Section 5.

With guidance from our previous spatial analysis (seen in Figure 5g), the first categorical variable we looked at was the HalfBath variable. For properties with two or more half baths, the model fell outside the ASR range at 1.114048, although the COD was satisfactory at 4.866196. This makes sense intuitively as there are very few properties with two or more half baths, and confirming previous analyses, we saw the subset only made up approximately 1.653523 percent of the test set.

We were a bit cautious with evaluating the TotalFinishedArea variable with this method as the distribution is more “gamma-like” than normal, as discussed at the end of Section 3.2. We found the model did poorly with properties smaller than 900 square feet, which had a ASR of 1.15019 but a COD of 2.085207. This subset made up approximately 7.682524 percent of the test set. As previously detailed, the cutoff of 900 square feet was determined rather arbitrarily.

Overall, it appears newer properties are the main level for which there could be a separate model built. A potential solution for the TotalFinishedArea variable rather than a separate model could be a more complex type of transformation, as we did not find a common one that was interpretable (as mentioned at the end of Section 3.2). The client suggested a square root transformation, but when applied to the linear model it made only a marginal difference to ASR and COD. A separate model for the LastSalePrice variable would only be applicable to properties with previous sales data, and properties with two or more half baths appear to represent an insufficient proportion of observations to warrant a separate model, so we make no recommendation for separate models on either variable.

Section 4.2: Complex Models

For our complex model, the natural inclination after cleaning and reviewing the data was to grow a random forest. This appeared to be the most complex of the methods we were familiar with and a good fit for this type of large dataset. Random forests are also a commonly used method for mass appraisal, especially in more densely populated urban areas, and their use has been detailed in various journal articles (e.g. Levantesi and Piscopo, 2020).

Section 4.2.1: Random Forests

The code for this section is located in Random Forest.Rmd.

The process for growing a random forest consisted of using the randomForest() function in the randomForest package. As with any model, one of the main questions prior to growing this random forest was which variables to use. Since previous models had already been created, we decided to use the subsets of variables from those models as starting points and grow a random forest on each set of

variables. A random forest was also grown using all variables in the dataset, and we would choose the "best" random forest from the following sets of variables:

1. All variables *except* Valuation (33)
    - As mentioned in Section 4.1.2, valuations are considered "red herrings" and thus were not included
2. Variables from the linear model (21)
3. Variables from GLM created by bestglm *except* Quality (9)
4. Variables from GLM created by purposeful variable selection (16)
5. Arbitrary choice of variables based on what intuitively made sense (12)
    - LastSalePrice ~ BuildingType + PhysicalCondition + YearBuilt + FullBath + HalfBath + Kitchen + TotalFinishedArea + SeasonSold + AirConditioned + NeighborhoodCluster + PrimaryWall + Parking

The same seed number was used when growing each tree as well as the default selections of 500 trees and p/3 variables per split (rounded down). The ASR, COD, MSE, and r^2 for each random forest can be found in Table 2. We can see there are three random forests with ASR and/or COD values outside the ranges for acceptable models in the field of assessment ($0.9 < ASR < 1.1$ and $COD < 15$), and these values are highlighted in red.

Table 2: Results of Random Forests

	ASR	COD	MSE	r^2
Full Data (33)	1.44693388	14.9267116	3,382,055,751	0.690430
Linear Model (21)	1.43075374	16.6610998	3,416,858,940	0.687244
bestglm (9)	1.03893338	10.1856495	3,999,503,178	0.633913
PVS (16)	1.44446165	17.6974501	3,490,417,322	0.680511
Choice (12)	1.03843279	10.5961782	3,868,303,043	0.645922

We removed the three random forests with unacceptable ASR and COD values from consideration as these would be unhelpful for the purpose of prediction and for our client. Among the two remaining random forests, we can see choosing the "better" one is difficult and subjective. After extensive review of the statistics in Table 2 and the plots of the MSE and r^2 vs. number of trees, the random forest constructed from the "Choice" variables appears to be the better of the two. Although the "bestglm" random forest has a lower COD, we can see the "Choice" random forest has a marginally lower ASR as well as improved metrics for MSE and r^2. We do not proceed with further analysis here for the four random forests not chosen, but the code for computing the statistics and plots for each is included in the corresponding file.

Plotting the MSE vs. the number of trees confirmed that the MSE was unreasonably high (3,868,303,043). As previously discussed, the unique distribution may have led to some poor predictions. We can see in Figure 11 that the MSE levels off after approximately 200 trees.

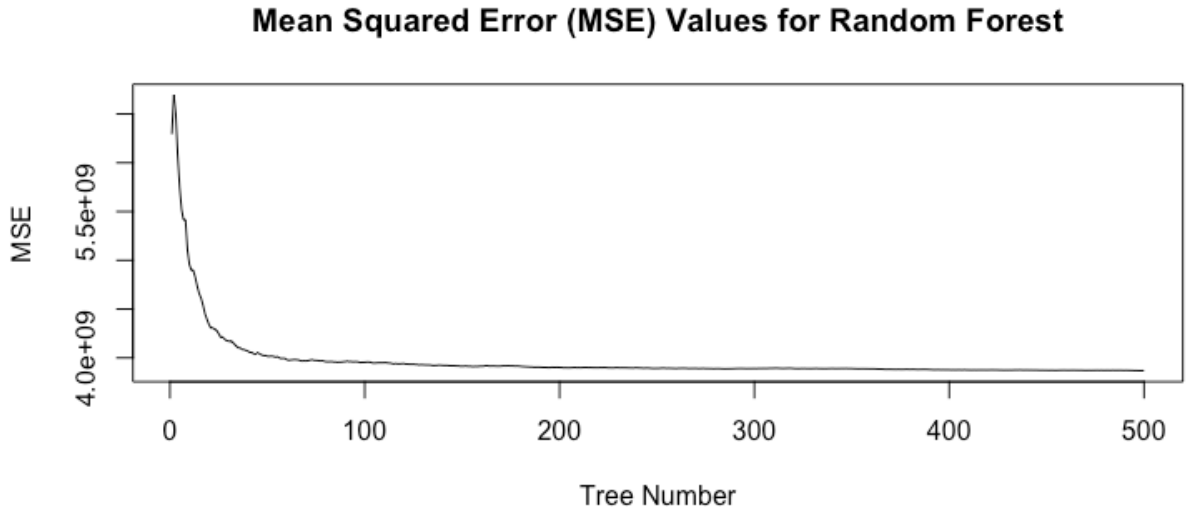


Figure 11: MSE vs. Number of Trees in Random Forest

We also plotted the r^2 of the model vs. number of trees and saw it was similarly maximized after approximately 200 trees ($r^2 = 0.65$). We can see in Figure 12 that this plot is almost a mirror image of the plot in Figure 11.

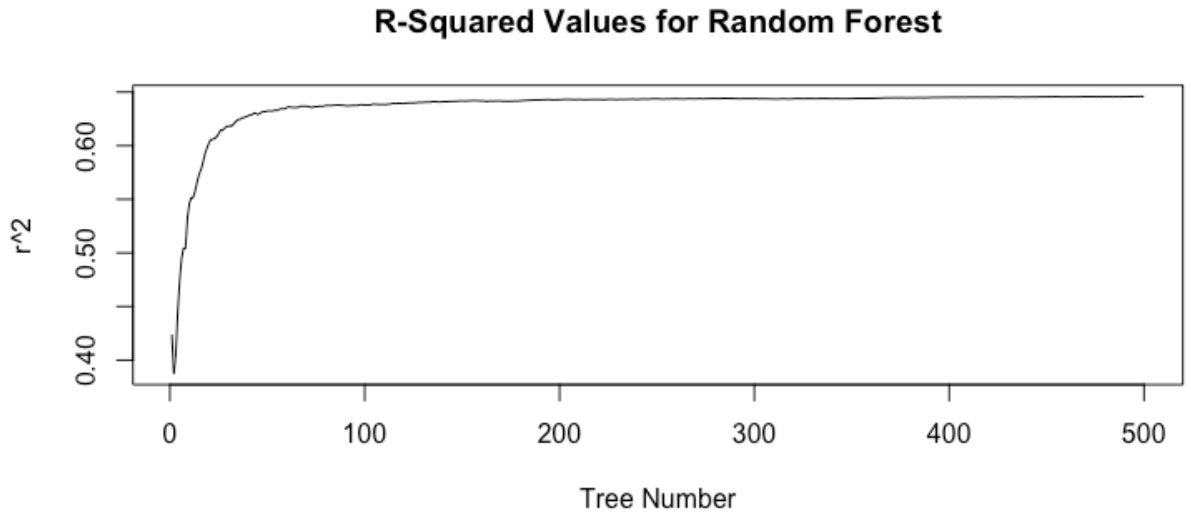


Figure 12: r^2 vs. Number of Trees in Random Forest

Finally, a variable importance plot was created using the varImpplot() function which showed the NeighborhoodCluster and TotalFinishedArea variables were the most important in the random forest. We can see from Figure 13 that the YearBuilt variable appears to be a definitive third, with the remaining variables in the model being relatively distant.

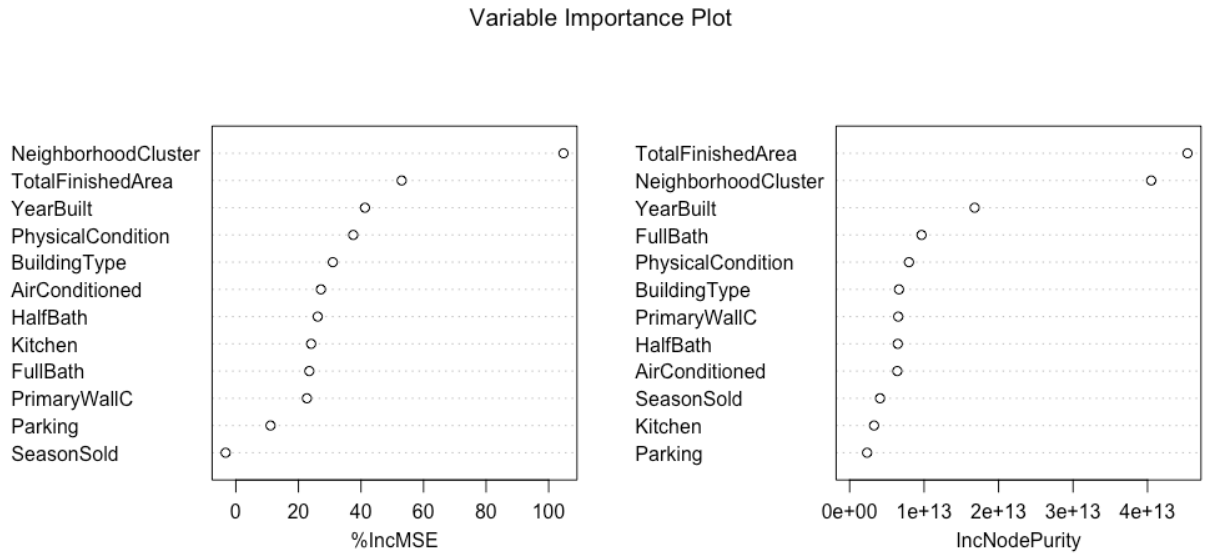


Figure 13: Variable Importance Plot for Random Forest

Section 5: Results/Recommendations for Client

After closely reviewing the results and output of each model, we recommend the linear model to our client. Recall from Section 4.1.3 that the skeleton code for the linear model is:

> lm(LastSalePrice ~ factor(PhysicalCondition) + factor(Neighborhood) + YearBuilt + FullBath + factor(RatingBath) + HalfBath + Kitchen + TotalFinishedArea + LandSF + AirConditioned + FrameShed + MasonAdjustment + PlasticLinedPool + DivingBoard + PoolLadder + ReinforcedConcreteSF + XCoordinate + YCoordinate + Quality + PrimaryWall + Parking)

This model is good for resource efficiency as there is a single regression equation and it would be easy to implement and interpret. The intercept term for a linear model can be interpreted as the theoretical "baseline" observation (even if it is unreasonable), predicting the real estate value if the value of every independent variable were set to 0. Coefficients for quantitative variables can be interpreted as the increase (or decrease if negative) in predicted real estate value for a one-unit increase in the variable holding all other variables constant. Coefficients for categorical variables can be interpreted as the increase (or decrease if negative) in predicted real estate value for a one-level increase in the variable holding all other variables constant. For example, the quality variable has multiple levels that have a defined order and going from one level to the next would result in an increase of the value of the coefficient in predicted real estate value.

We also recommend considering a separate model for newly constructed properties, as mentioned in Section 4.1.5. While there is a very small proportion of properties constructed since 2017 in the sales data, this number will only increase over time, which may become problematic if a model continues to include contrasting older properties. Intuitively, these properties are much different than the median property in the city of Milwaukee and harder to assess, especially via mass appraisal. Separate models for these types of real estate could help alleviate this problem. We recommend trying different year cutoffs for defining "very new" properties, as a separate model may behave differently, and reviewing these cutoffs annually or periodically to ensure they are up to date.

Section 6: Conclusion

There are many different things that could be done in the future to improve existing analyses or expand on the work already done. One of them is constructing the proposed hybrid model, which could incorporate methods not covered in courses we have previously taken. Another is continuing to experiment with different combinations of variables and/or interaction terms in existing models. We know there are a lot of variables in the data and thus a lot of combinations of variables, and we saw that different methods performed better with different combinations. A third, specifically for the random forests, is to experiment with different numbers of trees grown (ntree) and variables considered at each split (mtry). This tends to only change the model and prediction statistics like MSE and r^2 marginally but could have a larger effect on ASR and COD. We are optimistic that, given more time, the proposals mentioned above would be used, hopefully resulting in a more accurate and robust model.

In conclusion, this project was a good learning experience in applying methods learned in previous coursework to real-world data. A lot was learned in interacting with an external client rather than just with a professor with which there was a previous relationship, as this was more representative of a standard project in a full-time employment position rather than a class project.

Code

The code for this project is located at https://github.com/racheljordanstats/Consulting-Project.

Individual Contributions

Charles Hwang: Random Forests, creating maps/spatial EDA, looking at where our best simple model performs poorly and proposing solutions

Rachel Jordan: Creating test/train data sets, non-spatial EDA, data wrangling, bestglm() model, purposeful variable selection of GLM model, linear model, LASSO, project administration

Works Cited

Agresti, A. (2019). *An introduction to categorical data analysis* (3rd ed.). Wiley-Blackwell.

Deaconu, A., Buiga, A., & Tothăzan, H. (2022). Real estate valuation models performance in price prediction. *International Journal of Strategic Property Management*, *26*(2), 86–105. https://doi.org/10.3846/ijspm.2022.15962

Levantesi, Susanna, and Gabriella Piscopo. (2020). The Importance of Economic Variables on London Real Estate Market: A Random Forest Approach. *Risks (Basel)*, vol. 8, no. 4, p. 112. https://doi.org/10.3390/risks8040112.

McLeod A, Xu C, Lai Y (2020). _bestglm: Best Subset GLM and Regression Utilities_. R package version 0.37.3, <https://CRAN.R-project.org/package=bestglm>.

Nowak, A., & Smith, P. (2016). Textual analysis in real estate. *Journal of Applied Econometrics*, *32*(4), 896–918. https://doi.org/10.1002/jae.2550

Perraillon, M. C. (2020). *Week 7: Cost data and Generalized Linear Models*. *Health Services Research Methods I HSMP 7607*. Retrieved November 2022, from https://clas.ucdenver.edu/marcelo-perraillon/sites/default/files/attached-files/week_7_glm_and_costs_perraillon.pdf.