Spring 5-20-2019

Data Mining Techniques for Predicting Real Estate Trends

David Vargason
La Salle University, vargasond1@student.lasalle.edu

David Vargason
Capstone Submission – Final
April 16, 2019

Data Mining Techniques for Predicting Real Estate Trends

I. Abstract

A wide variety of businesses and government agencies support the U.S. real estate market. Examples would include sales agents, national lenders, local credit unions, private mortgage and title insurers, and government sponsored entities (Freddie Mac and Fannie Mae), to name a few. The financial performance and overall success of these organizations depends in large part on the health of the overall real estate market. According to the National Association of Home Builders (NAHB), the construction of one single-family home of average size creates the equivalent of nearly 3 new jobs for a year (Greiner, 2015). The economic impact is significant, with residential construction and related activities contributing approximately 5 percent to overall gross domestic product. With these data points in mind, the ability to accurately predict housing trends has become an increasingly important function for organizations engaged in the real estate market.

The government bailouts of Freddie Mac and Fannie Mae in July 2008, following the severe housing market collapse which began earlier that year, serve as an example of the risks associated with the housing market. The housing market collapse had left the two firms, which at the time owned or guaranteed about $5 trillion of home loans, in a dangerous and uncertain financial state (Olick, 2018). Countrywide Home Loans, Indy Mac, and Washington Mutual Bank are a few examples of mortgage banks that did not survive the housing market collapse and subsequent recession. In the wake of the financial crisis, businesses within the real estate market have recognized that predicting the direction of real estate is an essential business requirement. A business acquisition by Radian Group, the Philadelphia-based mortgage insurance company, illustrates the

importance of predictive modeling for the mortgage industry. In January 2019, Radian Group acquired Five Bridges Advisors, a Maryland-based firm which develops data analytics and econometric predictive models leveraging artificial intelligence and machine learning techniques (Blumenthal, 2019).

This paper will initially provide an overview of data mining, including its history, range of learning styles, data preparation best practices, description of training and test data, and measures for evaluating of results. The remainder of this paper is organized as follows: The literature review section will survey three peer-reviewed research articles that examine the uses and limitations of data mining and machine learning for predicting real estate related metrics. Following the literature review section, this paper will proceed to test a variety of machine learning algorithms with the goal of accurately predicting three real estate performance metrics: single family housing starts, supply of housing at current sales rate, and residential building permits.

Data mining will be performed using Weka, a free software package that supports initial data preparation and discretization, classification, regression, clustering, and evaluation of prediction results. Specific machine learning algorithms to be evaluated include Random Forest, multi-layer perceptron, logistic regression, OneR, and J48 (aka C4.5). Historical economic data used for this project (all publicly available) was obtained from the Moody's Analytics website (www.economy.com) using a subscription service. A complete listing of all data variables used for this project is shown in the appendix.

The complete process of data mining will be presented: initial data preparation, discretization, testing algorithms, and evaluating results will be explored in detail. A selection of algorithms with the highest accuracy and comprehensibility will be further tested under an "actuals versus expectations" scenario. The goal of this test is to determine how each model would have performed if placed into actual production in a business environment five years ago. To evaluate predictive accuracy, the selected models will be re-evaluated on five years of holdout data covering the period October 2013 through September 2018. Prediction results during the holdout period will be evaluated to determine the most accurate learning models for prediction.

II. Overview of Data Mining

A. Definition and Brief History of Data Mining

Data Mining is defined as the practical, non-theoretical technique of finding structural patterns in data, automatically or semi automatically, and using that information to make predictions (Witten et al, 2011). Witten emphasizes that structural patterns are explicit in nature. In other words, the discovered patterns help explain something about the data being evaluated. Patterns must also be meaningful and significant enough to add value, which is usually an economic value added to a business problem. Data mining involves the use of machine learning algorithms to scan through large quantities of data until a structural pattern is discovered, typically without any human involvement.

While data mining is the process of discovering patterns in data, machine learning can be defined as the computational methods (i.e. algorithms) used to perform the process of making accurate predictions from past information (Mohri et al, 2018). Machine learning encompasses the design of operationally efficient and accurate prediction algorithms. The concept of "machine learning" traces its history back to 1950 and the research of a British mathematician named Alan Turing. Turing authored a research paper titled *Computing Machinery and Intelligence* in which he examined the creation of intelligent machines and how artificial intelligence could be measured (Anyoha, 2017). Computing power in the 1950's limited Turing's ability to implement his ideas. Since then, advancements in computing power have created an environment where machine learning techniques have the potential to make significant economic contributions.

Data evaluated by machine learning algorithms can be structured, represented by conventional rows and columns, or unstructured, which includes information sources such as video, pictures, and text messages. Machine learning algorithms can evaluate and derive insight from unstructured data, however, this paper will focus on historical economic data that is structured. An example of machine learning on unstructured data would include image recognition on social media sites to glean insight into the success of product advertising. An athletic apparel company, such as Adidas, could scan social media sites to obtain a count of people wearing their apparel as identified by the Adidas brand logo.

B. Learning Styles in Data Mining

Data mining typically employs four different styles of machine learning: classification, numeric prediction, clustering, and association (Witten et al, 2011). Classification learning is the process of predicting the class (or category) of unobserved data after being presented and evaluating a set of classified examples (i.e. training data). Classification learning is considered supervised, where the algorithm functions under the supervision of correct outcomes for each of the training data examples. Numeric prediction is often considered a variation of classification, where the predicted outcome is a numeric value instead of a category. Linear regression would be an example of a classification numeric algorithm. Clustering is essentially the concept of grouping items that naturally go together. Clustering is considered unsupervised learning, where the algorithm learns from training data without being provided any actual results. The clustering algorithm must evaluate training data and determine if patterns exist on its own. A common clustering algorithm is k-means, which functions by assigning instances to the nearest cluster center using the Euclidean distance, or straight-line distance between two data points. In association learning, there is no specified class being predicted. Instead, association learning evaluates training data with the goal of finding interesting structures (or associations) which enable it to predict attributes, which are typically non-numeric. This paper will focus primarily on classification learning, with a more detailed description of classification algorithms provided in the literature review and research sections.

C. Data Preparation

Data preparation could be considered the most important phase of the data mining process. The author of *Data Preparation for Data Mining* emphasizes the importance by stating "data preparation and the data survey lead to an understanding of the data that allows the right model to be built, and built right the first time" (Pyle, 1999, p. 9). The primary objective of data preparation is to arrange an input data file, in which the natural order of the data is minimally disturbed and appropriately formatted for the purposes of the data miner. An incomplete data preparation phase can lead to significant frustration and time wasted during the model building phase. When data preparation has been

completed successfully, the process of running algorithms and building models is a relatively small part of the overall data mining effort (Pyle, 1999).

The first step in data preparation is surveying the data file being considered for mining. Pyle proposes that the goal of surveying is to answer three important questions:

1. *What is in the data set?*

   Evaluate the general structure of the data, including the number of rows and columns and the type of data contained in each column (numeric or nominal). Graphical visualizations of the data can help identify outliers, which may indicate potential errors in the data file.

2. *Can I get my questions answered?*

   Does the data file contain the appropriate attributes needed for prediction? If the goal is to predict housing starts, it might not make sense to include data entirely unrelated to real estate or the economy.

3. *Where are the danger areas?*

   Missing values can pose a significant danger area. Many machine learning algorithms will simply ignore missing values, which assumes there is no significance to a missing value. A number of algorithms are unable to handle missing values, with examples being decision tree classifier Id3 and rule classifier PRISM. When evaluating missing values, it is also important to consider if there is a legitimate reason for the existence of missing values.

Once the data set has been evaluated and potential danger areas have been identified and resolved, the next step is to structure the data file for the specific machine learning software tool being used. As previously mentioned, the data mining tool used for this paper is a free software package called Weka. The typical process for importing data into Weka would initially involve surveying the data in Microsoft Excel, cleaning the data as appropriate, saving the file as a comma-separated value (CSV) format, and finally importing into Weka. The Weka Explorer dashboard provides a listing of all attributes in the file, a statistical summary of each attribute, and a histogram providing a visual depiction of the range of values. The class, or attribute being predicted, is defaulted to the last attribute (furthest to the right) in the file. However, Weka does provide the option of choosing a different class upon selecting a classifier algorithm.

Weka provides the option for discretization of numeric attributes into nominal attributes. It can be useful to create two versions of data files in Weka: one file containing data in its original numeric format and another version with all fields fully discretized. Having both nominal and numeric file versions ready for data mining can allow for more efficient testing of various algorithms. Certain classification algorithms, such as PRISM and Id3, will only evaluate nominal attributes. Simple linear regression can only handle numeric attributes.

D. Test Options

Weka provides a variety of options for establishing the training and test data. Training and test data serve two different functions: training data is used determine the classifiers and test data is used to calculate the error rate of the final prediction method (Witten et al, 2011). Cross-validation is a commonly chosen testing method, which divides the dataset into a specific number of training and testing datasets (folds). For example, 10-fold cross validation splits the dataset into ten partitions containing approximately the same number of observations. In this example, training is conducted on 9 out of 10 partitions, with the 10^{th} partition held out as the test set. This process is repeated 10 times, so that each partition is evaluated once as the test set. Research has shown that 10-folds are about the right number to obtain the best estimate of error (Witten et al, 2011).

E. Evaluating Data Mining Results

Weka provides a summary of model accuracy results in the classifier output window, which is described below:

- *Correctly Classified Instances* – provides a quick synopsis of the number and percentage of instances correctly classified.
- *Mean Absolute Error* – provides the average of all individual errors without considering their sign. The error represents the average distance between each predicted point and the actual observations. The best possible value is zero.
- *Root Mean Squared Error* – a measure of accuracy similar to MAE but with the square root taken, which gives larger errors a disproportionate influence. RMSE is a useful accuracy measure when large errors are undesirable. Values closer to 0 are better.

- *Confusion Matrix* – provides a visual depiction of classifier predictions, with correct predictions shown along the top-left to bottom-right diagonal. Within this paper, confusion matrices have been color formatted to provide ease of interpretation. Cells within the main diagonal indicating a correct prediction have been colored dark green. Light green cells are located adjacent to the main diagonal, indicating an incorrect (but generally close) prediction result. Cells located more than one prediction unit from the main diagonal are colored orange, indicating a severely incorrect prediction.
- *Correlation Coefficient* – a measure of statistical correlation for numeric prediction that ranges from 0 to 1, where 0 would indicate no correlation and 1 would indicate perfect correlation.

For this project, the most important accuracy results are provided by mean absolute error, root mean squared error, and the confusion matrix. Economic data evaluated for this project will be fully discretized into nominal attributes, which consequently excludes numeric prediction algorithms and accuracy results obtained from correlation coefficient.

III. Literature Review

The literature section will review three peer-reviewed research articles pertaining to machine learning techniques and the real estate market. Each research article explores a different aspect of the real estate market: valuation modeling of multifamily rental properties, identifying real estate investment opportunities using machine learning, and forecasting residential housing values. The articles provide significant insight into the range of business applications for machine learning techniques within the broader real estate market.

A. Big Data in Real Estate? From Manual Appraisal to Automated Valuation

Property valuations have historically required the very manual process of evaluating the subject property and surrounding area, along with an evaluation of comparable transactions within neighboring areas ("comps"). Research has shown that when comparing the post-appraisal transaction price with the initial property appraisal, the appraised valuation is (on average) more than 12% higher or lower than the actual

transaction price (Kok et al., 2017). Manual appraisals typically lag the market, which can result in artificially low property valuations during real estate bull markets and the opposite during bear markets. Both traits can create an inefficient real estate market for buyers and sellers.

In their study, Kok, Koponen, and Martínez-Barbosa provide a detailed description of data preparation and modeling techniques with the goal of developing an automated valuation model (AVM) based on machine learning techniques. The author's research focuses on multi-family rental properties and supported by a dataset containing nearly 54,000 multi-family property records. The dataset used for modeling includes traditional real estate metrics including construction permits, vacancy rates, and mortgage interest rates. Interestingly, the authors also chose to include attributes on local crime rates, nearby music venues, water recreation within 30 minutes, and S&P 500 market data. The goal of including data beyond traditional real estate was to obtain a proxy for neighborhood vibrancy, which seems like a reasonable approach. The authors support their attribute inclusion decisions by explaining that machine learning does not have to be limited to 10 to 15 variables that someone deems important. A key advantage of machine learning, supported by modern computer processing power, is an almost unlimited number of variables can be used by the model.

Kok, Koponen, and Martínez-Barbosa continue by explaining the machine learning approaches used for their research article. Real estate asset valuation data is continuous, rather than categorical, and therefore the authors chose to focus on regression decision tree algorithms for prediction. In their opinion, the advantages of using decision trees include ease of interpretation, key statistical evaluations that are easily calculated, and rapid computer processing times relative to other machine learning techniques. This type of machine learning operates by minimizing the variance of regression between dependent and independent variables, with the goal of determining an order of importance. Each node of the decision tree contains an explanatory variable, with the most (best) explanatory variable located at the first node (root node). After the root node is determined, the order of importance is calculated again, with subsequent explanatory variables building out the branches of the decision tree.

Decision trees do have distinct limitations, including the risk of overfitting caused by overly complex decision tree models reflecting the presence of noise within the training data. Overfitting can result in models that produce poor predictions when using unseen data. Underfitting represents another risk of decision tree algorithm, which typically occurs when the tree has not been built out sufficiently. To avoid the potential drawbacks of decision trees, the authors chose to focus on two learning algorithms that include a modeling feature called *ensemble of trees*, which can be described as taking an average prediction from multiple decision trees. The first algorithm tested was Random Forest, which the authors describe as being very good at reducing variance and minimizing the risk of overfitting. The other ensemble model tested was XGBoost, which is a gradient boosting algorithm that builds many small decision trees from random samples of the training data.

Model testing was conducted using 70/30 cross-validation, which refers to using 70% of the data as training data and 30% as the test set. Cross-validation is a testing approach that can reduce the risk of underfitting or overfitting. Model performance was measured by evaluating two key performance indicators:

1. R^2 = measures the explained variance "robustness" of the model. Values closer to 1 (or 100%) indicate better model fit.
2. MSE = mean squared error, which measures model accuracy. Lower values in the MSE represent more accurate results.

How successful were Random Forest and XGBoost at predicting multi-family rental valuations? The authors tested different combinations of variable transformations, which resulted in four different result panels. Across all four testing panels, XGBoost generally tested highest with an R^2 ranging from 0.73 to 0.92 and MSE ranging from 8.6 to 19.1. Random Forest results show R^2 ranging from 0.62 to 0.86 and MSE ranging from 9.8 to 22.3. Summarizing their results, the authors argue that automated valuation models developed through machine learning techniques are a viable tool for the real estate sector.

There are several important takeaways from this research article, with implications for the main research goal of this paper:

- Including a wide range of data, beyond typical real estate metrics, can provide interesting results or relationships that have previously never been examined. This is demonstrated by Exhibit 6 in the research article, which shows a rank-order of explanatory variables. The variables "Green space within 30 minutes" and "Music events within 3 minutes" were near the top of the ranking.
- Regression trees can be a good machine learning technique for predicting continuous variables. The risk of overfitting can be overcome by testing algorithms that feature an *ensemble of trees*, with Random Forest being an example. Weka contains the Random Forest algorithm. XGBoost is not available in the current download version of Weka.
- Interestingly, the scope of the research article is closely aligned with the research goals of this capstone paper. Cross-validation will be utilized for model testing, consistent with the methodology used in this research article. The Random Forest algorithm, which the authors describe as minimizing the risk of overfitting, is included among the algorithms tested.

B. Housing Value Forecasting Based on Machine Learning Methods

The premise of this research article is, if housing values can be accurately predicted by machine learning methods, then government agencies can use that technology to engage in reasonable urban planning with positive economic and social benefits (Mu et al., 2014). Housing values in the Boston, MA suburbs are analyzed for this study. Housing value trends for that metropolitan area exhibit significant non-linear characteristics, which can present a challenge for machine learning techniques. The authors chose three machine learning algorithms for their study: support vector machine, least squares support vector machine, and partial least squares. These algorithms fall under the machine learning category of artificial neural networks, which is a prediction system inspired by the neural networks found in living organisms.

Support vector machine (SVM) is a supervised learning algorithm that can be used for both classification and regression. In Weka, this algorithm is named SMO and SMOreg which are both listed under the "functions" classifier menu. Although SVM can be used for both linear and non-linear regression, the authors describe non-linear regression as the primary advantage of this particular algorithm. SVM solves non-linear

problems by utilizing a "kernel trick" function that, in simplified terms, transforms non-linearly separable data into data structures that can be evaluated by linear classification, thereby improving model accuracy with reduced computing processing requirements. Another benefit of SVM is the required data sample size is small in relation to the complexity of the prediction problem.

Least squares support vector machine (LSSVM) is an enhanced version of SVM, overcoming hurdles related to the selection of parameters, model complexity, and poor computational performance when processing large datasets. LSSVM is being used for prediction problems within a wide range of industries. Successful LSSVM applications in healthcare and industry include the identification of tumors from magnetic resonance spectroscopy signals, air pollution prediction, and marketing and financial engineering (Suykens, 2002).

Partial least squares (PLS) is a less complex algorithm, relative to SVM and LSSVM, that has strong explanatory capability in solving problems with multiple variables (Mu et al., 2014). In general terms, PLS operates by identifying the best function matching within the sample data, as determined by minimized sum of the squared errors. Model accuracy is supported through the extraction of maximum information from training data, along with the algorithm's ability to separate sample noise from normal information. PLS is often chosen for regression modeling in circumstances when the number of variables exceeds the number of sample points. An important benefit of PLS is that is can efficiently simplify the data structure, analyze correlation, and build the regression model all at the same time.

Housing valuation data for the Boston area was obtained from the Machine Learning Repository (UCI) which contains a variety of datasets for free download. Housing value attributes include crime rate data, the proportion of non-retail business, and highway accessibility. Similar to the prior research article, the authors of this study included a wide range of data beyond typical real estate metrics. Data preparation involved removing missing values from the training data. As previously described, missing values can pose a significant hazard to the data mining process since many machine learning algorithms are unable to handle them. The final training data set contained 400 samples and the test data contained 52 samples.

How successful were SVM, LSSVM, and PLS at predicting housing values in the Boston suburbs? PLS scored the lowest accuracy results, with the MSE equal to 25.05 and computer processing time equal to 0.746 seconds. The authors commented that PLS was obviously worse at forecasting from non-linear data, describing the prediction situation as "not very ideal". In second place was LSSVM, which produced an MSE equal to 15.13 and required 20.37 seconds of processing time. The champion at predicting Boston housing values was SVM, which returned an MSE equal to 10.74 and only required 0.46 seconds to get it done.

There are several important takeaways from this research article:

- The article reinforces the value of including a wide range of data, including variables that would seem only marginally related to real estate. Crime rates, business activity, and access to highways are all important considerations when people purchase homes.
- The author's decision to focus entirely on Boston, MA could be a flaw in their research design. Home valuation trends can differ significantly between metropolitan areas, examples being San Francisco and Cleveland. Including additional metropolitan areas in the analysis would have strengthened the credibility of their results.
- The Boston housing data used for this research paper was obtained from the UCI Machine Learning Repository website. The UCI website provides a dataset summary, which indicates the data was originally compiled back in 1993. More recent housing value data, in which the 2008 housing crisis is fully reflected, would be more relevant for a research paper that was published in 2014.
- Based on the positive prediction results, SVM will be included among the machine learning algorithms evaluated in the research section of this paper. LSSVM does not appear to be available in the current download version of Weka.

C. Identifying Real Estate Opportunities Using Machine Learning

The goal of this research article is to identify opportunities for real estate investment by developing machine learning models that can identify properties which are listed significantly below market price (Baldominos et al, 2018). Investors interested in the housing market could leverage this type of model for arbitrage investing, which seeks

to profit from an imbalance in price. The geographic region of focus is the Salamanca district of Madrid, Spain, which the authors describe as home to 150,000 people and one of the wealthiest locations in Spain. Machine learning algorithms tested for this study include ensembles of regression trees, k-nearest neighbor, support vector regression (SVM regression), and multi-layer perceptron. The international perspective, extensive use of data visualizations, and wide range of data provided for an interesting and relevant read with important takeaways.

There are many factors that influence the value of a house, including the number of bedrooms, bathrooms, square footage, location, proximity to schools, nearby parks, and transportation. The combination of these factors can contribute to the main force determining the price of a house, which is essentially consumer demand. In an efficient market, the price of a product reflects all available information. The authors explain several reasons why the real estate market can be inefficient, resulting in an actual price that is significantly below the expected price. In one scenario, the house has remained on the market for an extended period of time, during which the seller has failed recognize that the property has increased in value. In another scenario, the seller intentionally sets the price lower than the expected value to facilitate a quick sale. Mortgage delinquency and the risk of foreclosure could be factors driving this scenario. The authors believe these two scenarios represent opportunities for investors to take advantage of market inefficiencies and make an immediate profit.

Data used for this research encompasses high-end residential properties in the Salamanca district of Madrid, Spain. Housing data was limited to properties valued at more than one million euros, which is equivalent to about $1.1 million U.S. dollars as of March 2019. The authors emphasize that while property features may vary (i.e. pool, garden, parking spaces), they consider the sample data to be relatively homogeneous and representing the prime market (as opposed to subprime). A wide variety of property attributes were chosen for this study. A selection of attributes includes the construction year, floor area, elevator access, community costs, box room, garden, and whether or not the property has a swimming pool. An initial exploratory data analysis of the attributes, leveraging the Pearson correlation coefficient, revealed that the square footage of the house was the primary variable affecting the price of the property. In total, the sample

dataset contains a total of 2,266 properties with values ranging from €1 million to €90 million euros, with the average equal to €2 million and median of about €1.66 million euros.

Four different machine learning algorithms were chosen for testing. Similar to the prior research article, SVM regression with a "kernel trick" function was selected for this study. The second machine learning technique chosen was k-nearest neighbor, a lazy algorithm that computes the distance of the class to all existing instances in the dataset. Prediction is determined by identifying the instances that are closest to the class being predicted, typically using the Euclidean distance method. The authors point out that k-nearest neighbor may not perform well when evaluating high-dimensional data with binary attributes. High-dimensional data is defined as data with a large number of attributes or characteristics. The next machine learning technique chosen is ensemble of regression trees, with Random Forest being an example of this type. The final learning technique chosen is multi-layer perceptron, which is an artificial neural network algorithm that uses the concept of backpropagation to classify instances. In general terms, backpropagation is defined as the process of determining the suitable gradient descent needed in the calculation of attribute weights used in neural networks (Witten et al, 2011). All four learning algorithms were tested using 5-fold cross validation. The authors did not provide an explanation for using 5-fold rather than 10-fold, which is generally the preferable number of folds for obtaining the best estimate of error.

Which models performed best for predicting the expected price? K-nearest neighbor returned the lowest (best) MSE with 4.044 and an R^2 equal to 0.3598. Looking back at the first research article where R^2 results ranged from 0.62 to 0.92, an R^2 score of just under 0.36 does not provide much confidence in the robustness of the model. Multi-layer perceptron came in second place with an MSE equal to 4.2262 and an R^2 equal to 0.3067. While both scores are slightly worse than k-nearest neighbor, the differences are minimal enough to justify additional testing of both models. Ensemble of regression trees and SVM regression came in third and fourth place, respectively. MSE results for both models were both slightly worse than the second-place finisher. However, their respective R^2 results were meaningfully worse with scores of 0.1253 and 0.0664.

There are several important takeaways from this research article:

- All three research articles discuss the importance of evaluating a wide variety of data attributes. This research article is unique because of its focus on high-end properties costing more than $1 million dollars. For this category of house, property attributes you would not normally think of are included in the dataset (i.e. elevator in the house).
- The research article utilized a variety of maps and data visualizations to tell the story. A similar design and use of visualizations will be incorporated into the research section of this paper.
- The author's decision to focus entirely on a small region in Spain could limit the credibility and usability of their findings. If the goal is to identify arbitrage opportunities in real estate, it would seem necessary from a credibility perspective to include a wider geographic area to capture a greater number of real estate transactions.

IV. Data Mining Techniques for Predicting Real Estate Trends

A. Introduction

A common theme in the literature review section is the wide range of applications for machine learning within the real estate market. While predicting opportunities for arbitrage would seem both interesting and potentially lucrative, the research section of this paper will focus on predicting real estate trends with practical applications for businesses and government agencies. For example, a mortgage bank might forecast housing starts to determine future capital requirements, interest revenue, and future loss projections. A construction firm would likely be interested in predicting the supply of housing at the current sales rate. Local governments may be interested in forecasting residential building permits for urban planning purposes.

In this section, a variety of machine learning algorithms will be evaluated for ability to accurately predict three real estate performance metrics: single family housing starts, supply of housing at current sales rate, and residential building permits. Training and test data include a wide variety of real estate and economic attributes, represented monthly beginning in January 1976 through September 2013. For each predicted metric,

the algorithms with the highest accuracy and comprehensibility will be further tested using five years of holdout data covering the period October 2013 through September 2018. The goal of this test is to determine how the most successful prediction models would have performed if placed into actual usage in a business environment five years ago.

B. Data Preparation

Historical economic data used for this project was obtained from the Data Buffet within the Moody's Analytics website (www.economy.com) using a subscription service. A complete listing of data variables is shown in the Appendix section. Similar to the three research papers, this project utilized a wide range of data beyond typical real estate and economic performance metrics. Well known housing variables include the Federal Housing Finance Agency (FHFA) Home Price Index, the National Association of Realtors (NAR) Housing Affordability Index, the U.S. Bureau of Labor Statistics (BLS) Consumer Price Housing Index, and the historical 30-year fixed mortgage rate. General economic variables include historical rates for U.S. Treasury Bills, Gross Domestic Product (GDP), and the M1 Money Stock. A wide variety of miscellaneous economic variables were evaluated, such as the BLS Consumer Price Gasoline Index, Consumer Price Prescription Drug Index, and U.S. Census Bureau (BOC) population growth forecasts for immigration and within population age cohorts. Manufacturing variables that would seem to be unrelated to real estate were also included, such as the U.S. Federal Reserve (FRB) Railroad Rolling Stock Manufacturing Index, Ship and Boat Building Index, and Iron and Steel Products Index.

The Data Buffet provides the option of downloading data in monthly, quarterly, semi-annual, or annual intervals. Monthly data was chosen, rather than quarterly or annual, to fully capture seasonal variations and provide the maximum number of data points for evaluation. Data was initially reviewed for missing or invalid records and none were found. Economic data reported in nominal dollar amounts was converted to percent change from prior month, since comparisons between dollars in 1976 and 2018 would need to be adjusted for inflation. The final data file was converted to a comma-separated value (CSV) file for importing into Weka.

Discretization was performed by converting continuous numeric attributes into nominal attributes, with five bins chosen rather than the Weka default option of ten bins. Choosing the right number of discretization bins requires some trial and error. Five bins were selected as a reasonable number for testing purposes, as an excessively wide number of bins can increase the risk of overfitting. The final data check involved a visual inspection of the histogram produced for each variable, which is shown on the Weka preprocess screen. Histograms provide a visual depiction of the range of values and can help identify any significant outliers.

C. Single Family Housing Starts

This monthly reported real estate metric represents the annualized number of single-family housing units (as opposed to multi-family townhouses) for which construction activity has commenced during the reporting period. Single-family units, which represent the largest share of the U.S. housing market, are the primary focus for many housing-related businesses, including mortgage bankers and insurers. Predicting single-family housing starts is likely an essential component of managing those businesses. For example, a mortgage insurance company would forecast housing starts to determine future capital requirements, premium revenue, and loss projections.

Exhibit 1 contains the monthly reported single-family housing starts from January 1976 through September 2018. As depicted in the graph, monthly housing starts were generally on an upward trend from the early 1990's through mid-2007. The sudden drop-off in activity corresponds with the financial crisis period, with a gradual recovery and return to an upward trend starting in early 2012. The question we will attempt to answer is: can machine learning algorithms, using real estate and other economic data from 1976 through 2013, accurately predict single family housing starts? In addition, will the most accurate learning models on the training and test data also be successful at prediction when evaluating five years of holdout data?

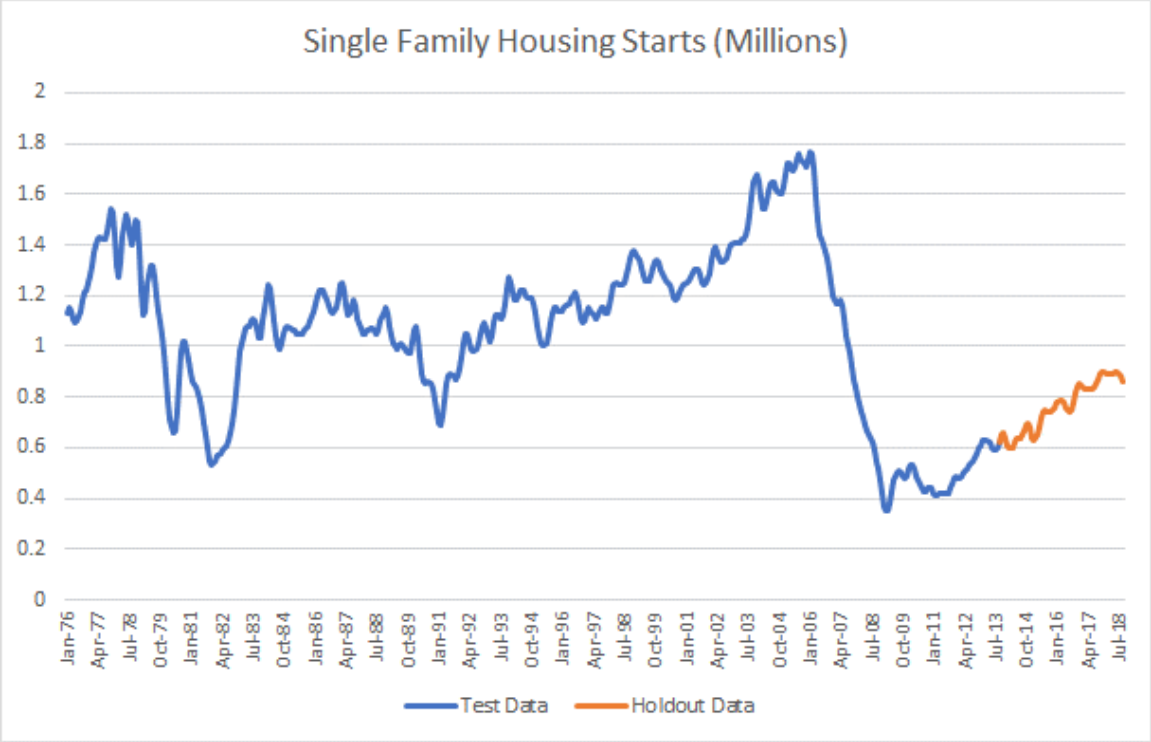

Exhibit 1: Single family housing starts (annualized in millions) from January 1976 to September 2018.

Two series of tests were performed for predicting single family housing starts. The first series of tests were conducted on data containing 70 attributes, which included a broad range of real estate related metrics along with data generally unrelated to the real estate market. The second series of tests were performed on a reduced number of attributes (12) that are more specific to real estate. The reason for this data adjustment was to test the accuracy impact from using two significantly different data files. The list of variables used for both series of tests is shown in the Appendix. Machine learning was performed using 10-fold cross validation. Outlined below is a description of the machine learning algorithms used for predicting single family housing starts (source = Weka documentation and Witten et al, 2011).

- **Decision Trees**
    - **J48** – decision tree algorithm that selects attributes which have the smallest entropy, which can also be described as having the largest information gain.

    - **Random Forest** – supervised algorithm that builds an ensemble of random decision trees, with the final prediction model considering the combined result of all trees.
    - **ID3** – decision tree algorithm based on the concept of divide-and-conquer.
- **Rule Classifiers**
    - **JRip** – rule learner based on repeated incremental pruning to reduce errors (Ripper).
    - **OneR** – uses the minimum-error attribute for prediction, with the model generated being the rules learned.
    - **Prism** – classification rule that uses concept of separate-and-conquer to continually add clauses to rules until 100% accuracy is obtained for each rule.
- **Functions**
    - **Logistic Regression** – linear logistic regression
    - **Simple Logistic** – similar to logistic regression, but with built-in ability to select attributes for evaluation.
    - **Multi-Layer Perceptron** – artificial neural network algorithm that uses the concept of backpropagation to classify instances.
    - **SMO** – Sequential Minimal Optimization algorithm for linear vectors utilizing a "kernel trick" function.
    -

- **Nearest Neighbor**
    - **IB1** – nearest-neighbor classifier (lazy learner) that uses Euclidean distance to predict the class
    - **IBk** – k-nearest-neighbors classifier (lazy learner) where the number of nearest neighbors (k) can be specified by the user. Weka default = 1.

When evaluating data mining results, it can be useful to compare each algorithm's performance against an appropriate strawman. The appropriate strawman for a data mining problem should be an algorithm, preferably a simple, clear-box method that can serve as a baseline. OneR would seem to be an appropriate choice for strawman, since it

is a comparatively simple method that always chooses one minimum-error attribute for prediction. Accuracy results from the more sophisticated models will serve as challengers to the strawman.

Exhibit 2 contains prediction accuracy results for Test 1, which used the 70-attribute data file. The percentage of correctly classified instances ranged from 59.4% to 91.4%. The mean absolute error ranged from 0.0344 to 0.2447 and the root mean squared error ranged from 0.1771 to 0.4031. With the notable exception of OneR, the prediction accuracy results all appear generally successful, as indicated by the percentage of correctly classified instances greater than 80%, mean absolute errors below 0.10, and root mean squared errors mostly below 0.25. K-nearest neighbor (IBk) was tested with three different user-selected values for the number of neighbors. A small value for K will generally provide the most flexible fit, exhibiting high variance but low bias (Zakka, 2016). On the other hand, higher values for K are typically more resistant to outliers by averaging more voters in each prediction. The most accurate prediction results for IBk were obtained when the number of neighbors (k) was set to 1, with incrementally lower results as k was increased to 3 and then 5.

Based on the model accuracy results depicted in Exhibit 2, simple logistic, multi-layer perceptron, IB1, and IBk (k = 1) were selected as the most successful algorithms for testing on five years of holdout data. All four algorithms produced a correctly classified percentage of greater than 90%. For comparison, the strawman algorithm produced 59.4% correctly classified instances, underlying the strong relative performance of the four algorithms selected for additional testing.

Test Data - January 1976 through September 2013

	Correctly Classified	Mean Absolute Error	Root Mean Squared Error
Decision Trees			
J48	83.0%	0.0768	0.2381
Random Forest	89.8%	0.0900	0.1794
ID3	84.8%	0.0541	0.2325
Rule Classifiers			
JRip	83.0%	0.0854	0.2403
OneR	59.4%	0.1625	0.4031
Prism	80.8%	0.0464	0.2154
Functions			
Logistic Regression	85.2%	0.0591	0.2320
Simple Logistic	91.4%	0.0545	0.1771
Multi-layer Perceptron	90.9%	0.0447	0.1809
SMO	88.5%	0.2447	0.3229
Nearest Neighbor			
IB1	91.4%	0.0344	0.1856
IBk (k = 1)	91.2%	0.0387	0.1857
IBk (k = 3)	90.3%	0.0532	0.1674
IBk (k = 5)	86.8%	0.0742	0.1916

Exhibit 2: Prediction accuracy results from single family housing starts – Test 1

Holdout Data - October 2013 through September 2018

	Correctly Classified	Mean Absolute Error	Root Mean Squared Error
Tested Models with Highest Accuracy			
OneR	8.3%	0.3667	0.6055
Simple Logistic	6.7%	0.3736	0.5729
Multi-layer Perceptron	8.3%	0.3675	0.5822
IB1	8.3%	0.3667	0.6055
IBk	8.3%	0.3662	0.6027

Exhibit 3: Prediction accuracy results from single family housing starts – Test 1 holdout data.

Exhibit 3 contains prediction accuracy results when re-evaluating the most successful models from Test 1 against five years of holdout data, which encompasses the period October 2013 through September 2018. The OneR strawman algorithm was also tested on holdout data to provide a baseline comparison. The models have not seen this data, which provides an "actuals versus expectations" scenario assuming the models were

placed into actual business usage. Accuracy results are significantly lower on the holdout data, with the percentage of correctly classified instances all below 10%. The range of mean absolute and root mean squared errors are all significantly higher (worse). Prediction results are nearly identical to the OneR strawman, signifying that the more sophisticated algorithms are no better than a simple, baseline method.

	Units Predicted				
Actual Units (Millions)	**A**	**B**	**C**	**D**	**E**
A = Up to 0.634 units	73	1	0	0	0
B = 0.634 to 0.918 units	2	41	5	0	0
C = 0.918 to 1.202 units	0	3	169	8	0
D = 1.202 to 1.486 units	0	0	13	92	5
E = 1.486 or greater units	0	0	0	4	37

Summary of Prediction Results

412	90.9%	**Correct Predictions**
41	9.1%	**Incorrect but Close**
0	0.0%	**Incorrect - Severe**

Exhibit 4: Multi-layer perceptron confusion matrix results on Test 1 data (453 instances)

	Units Predicted				
Actual Units (Millions)	**A**	**B**	**C**	**D**	**E**
A = Up to 0.634 units	5	0	0	0	0
B = 0.634 to 0.918 units	44	0	10	1	0
C = 0.918 to 1.202 units	0	0	0	0	0
D = 1.202 to 1.486 units	0	0	0	0	0
E = 1.486 or greater units	0	0	0	0	0

Summary of Prediction Results

5	8.3%	**Correct Predictions**
54	90.0%	**Incorrect but Close**
1	1.7%	**Incorrect - Severe**

Exhibit 5: Multi-layer perceptron confusion matrix results on Test 1 holdout data (60 instances)

The confusion matrix also provides insight into prediction accuracy, providing a visual depiction of results which may offer useful information not apparent in other performance indicators. Exhibit 4 shows the confusion matrix results for multi-layer perceptron algorithm on the full test data. Nearly all predictions (90.9%) are either

correct or hover close to the top-left to bottom-right diagonal indicating a correct prediction. Exhibit 5 shows the confusion matrix results from multi-layer perceptron on the holdout data. While only 5 out of 60 instances were correctly predicted, nearly all the remaining 55 instances (all but one) are within one cell of the correct prediction diagonal. Notice the absence of severely incorrect predictions, which appear as points within the orange-colored cells. This would suggest that, although the model is far from perfect, it does produce prediction results that are not far removed from actual outcomes.

The second series of tests were performed on a reduced number of attributes (12) that are more specific to real estate. The goal of this test is to determine if using fewer attributes that are more closely related to real estate could improve prediction accuracy. Exhibit 6 contains accuracy results using the 12 attribute test data. Nearly all the performance accuracy indicators are less favorable when compared to the original test using 70 attribute data. On the other hand, when the three most successful models from this test are re-evaluated on 12 attribute holdout data, the results (Exhibit 7) appear modestly more favorable compared to the original holdout data test which used 70 attributes. These results illustrate the importance of testing a variety of attributes and being willing to try new approaches.

Test Data - January 1976 through September 2013

	Correctly Classified	Mean Absolute Error	Root Mean Squared Error
Decision Trees			
J48	81.0%	0.0908	0.2429
Random Forest	87.6%	0.0932	0.2045
ID3	79.5%	0.0706	0.2588
Rule Classifiers			
JRip	78.1%	0.1159	0.2627
OneR	55.8%	0.1766	0.4202
Prism	76.8%	0.0670	0.2588
Functions			
Logistic Regression	77.7%	0.0979	0.2809
Simple Logistic	83.9%	0.0890	0.2224
Multi-layer Perceptron	84.8%	0.0723	0.2269
SMO	83.0%	0.2470	0.3267
Nearest Neighbor			
IB1	80.8%	0.0768	0.2772
IBk (k = 1)	83.9%	0.0746	0.2346
IBk (k = 3)	83.2%	0.0965	0.2227
IBk (k = 5)	81.9%	0.1105	0.2308

Exhibit 6: Prediction accuracy results from single family housing starts – Test 2

Holdout Data - October 2013 through September 2018

	Correctly Classified	Mean Absolute Error	Root Mean Squared Error
Tested Models with Highest Accuracy			
OneR	6.7%	0.3733	0.6110
Random Forest	10.0%	0.3342	0.4672
Simple Logistic	18.3%	0.3338	0.5605
Multi-layer Perceptron	16.7%	0.3336	0.5584

Exhibit 7: Prediction accuracy results from single family housing starts – Test 2 holdout data.

In conclusion, test results for predicting single family housing starts could be characterized as a partial success. The models built using test data all exhibited accuracy results that significantly exceeded the strawman algorithm. Accuracy results showed strong predictive ability, demonstrated by an average percentage of correctly classified (excluding the strawman) equal to 85.0%. Accuracy results were less encouraging when the most successful models were re-evaluated on recent holdout data. A potential

explanation for low accuracy results on the holdout data could be provided by Exhibit 1, which graphs single family housing starts from 1976 through 2018. The health of the U.S. economy changed dramatically in mid-2007, as illustrated by the collapse in housing starts that persisted through 2009. Training and test data include data points during the recession, which could be overly influencing each model and thereby causing unexpected prediction results when evaluating holdout data. Despite the potential impact from recession-era data, accuracy results on the holdout data are not far removed from actual outcomes, as evidenced by the confusion matrix results. Simple logistic and multi-layer perceptron were the most successful algorithms for both test scenarios.

D. Supply New Homes at Current Sales Rate

This metric represents how long the current inventory of houses for-sale would last given the current sales rate and assuming no additional houses were constructed (Federal Reserve Bank of St. Louis, 2019). The statistic measures the size of the for-sale housing inventory in relation to the number of houses currently being sold. Realtor agencies may find forecasting this metric useful, since the supply of homes for-sale could be necessary for predicting future business activity and staffing levels. Construction firms may closely monitor this metric to predict future demand, since a shrinking supply of new homes could increase demand for residential construction. The ability to accurately predict housing supply could have practical business applications across a variety of different enterprises.

Exhibit 8 depicts the historical trend of new home supply at the current sales rate at each period. Starting in the late 1970's a general declining trend in the supply of new homes began to take shape, with occasional increases in volatility that appear to approximately correspond with recession periods. The 2007 housing market collapse and recession caused the supply of new homes to increase dramatically, peaking at nearly a year of supply at the then-current sales rate. While the economic recovery brought some stability to this metric, recent periods have seen the supply of houses gradually increase. Test data for this prediction exercise will encompass 453 months starting in January 1976 through September 2013. Holdout data contains 60 months of observations starting in October 2013 through September 2018. Test and holdout data comprise of 70 attributes

along with the class being predicted, representing the same real estate and economic data used in the prior exercise.

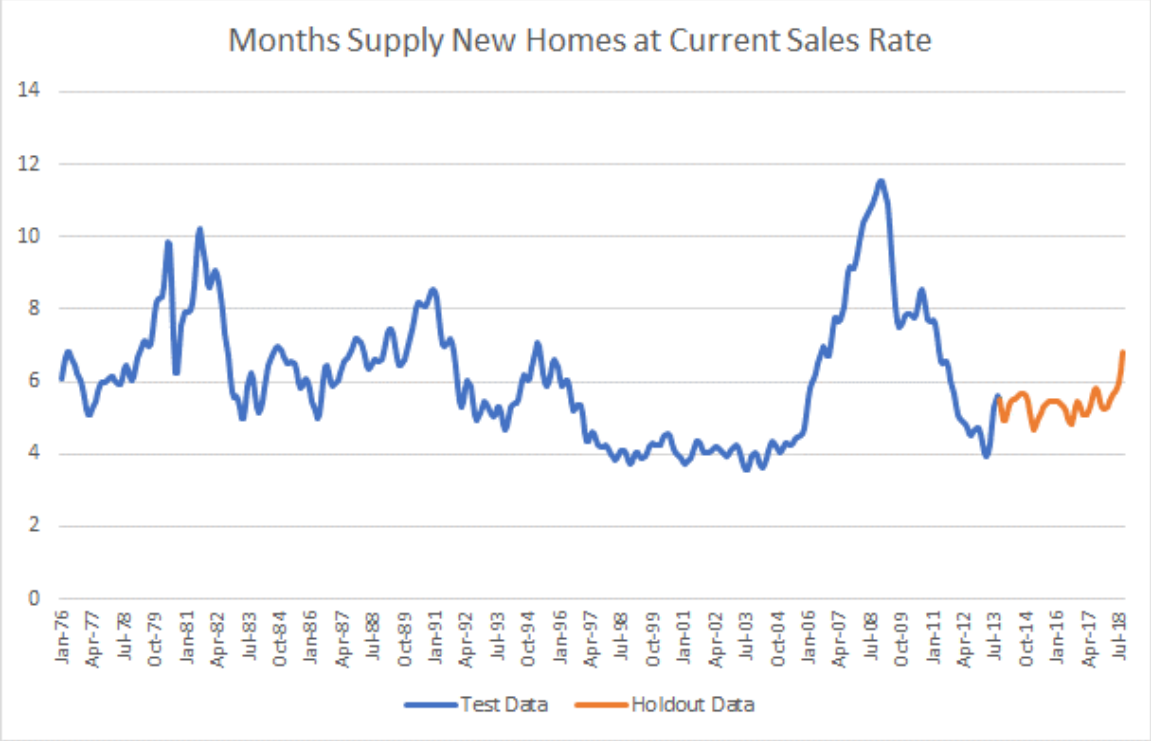

Exhibit 8: Supply new homes at current sales rate, January 1976 to September 2018.

Exhibit 9 contains accuracy results for the same twelve machine learning algorithms. Once again, OneR serves as our strawman with more sophisticated models serving as challengers to the strawman. The attribute that OneR chose as the minimum-error predictor is *birth rate per 1,000 U.S. residents*, which at first glance might seem generally intuitive that a correlation exists between birth rates and the supply of new homes. OneR exhibited a percentage of correctly classified instances of just about 50%, which is the lowest accuracy of all algorithms tested. The other, more sophisticated methods returned an average of 84% correctly classified instances. Visual inspection of the accuracy indicators reveals that Random Forest, simple logistic, IB1, and IBk (k = 1) are the most successful models. It should be noted that multi-layer perceptron required a significant amount of processing time; over an hour of processing time in Weka was required for this test. Computer processing time could be a valid consideration, along with prediction accuracy, for determining model success.

In addition to accuracy measurements, models should be evaluated for comprehensibility. In other words, does the model seem logical and will people trust it? Of the four models selected for testing on holdout data, Random Forest and IB1 could be considered black box models because Weka provides very little insight into how the models were constructed. Weka does provide model details for simple logistic and multi-layer perceptron, with simple logistic being the more coherent model for interpretation. A review of the simple logistic model reveals some interesting elements: *rental vacancy rate*, *percent change in housing stock*, and *distressed share of repeat home sales* are at or near the top of the model construct, indicating their high level of importance for predicting the supply of new homes. From a model user's perspective, it seems logical and believable that those three attributes could influence the supply of new homes.

Test Data - January 1976 through September 2013

	Correctly Classified	Mean Absolute Error	Root Mean Squared Error
Decision Trees			
J48	79.9%	0.0722	0.2364
Random Forest	88.7%	0.0944	0.1795
ID3	83.9%	0.0474	0.2177
Rule Classifiers			
JRip	73.7%	0.0978	0.2744
OneR	50.3%	0.1656	0.4069
Prism	72.2%	0.0581	0.2410
Functions			
Logistic Regression	85.2%	0.0509	0.2096
Simple Logistic	89.6%	0.0473	0.1681
Multi-layer Perceptron	85.9%	0.0484	0.1922
SMO	86.3%	0.2258	0.3158
Nearest Neighbor			
IB1	89.0%	0.0368	0.1918
IBk (k = 1)	89.8%	0.0394	0.1855
IBk (k = 3)	86.8%	0.0532	0.1663
IBk (k = 5)	83.7%	0.0761	0.1918

Exhibit 9: Prediction accuracy results for supply new homes at current sales rate – test data.

Holdout Data - October 2013 through September 2018

	Correctly Classified	Mean Absolute Error	Root Mean Squared Error
Tested Models with Highest Accuracy			
OneR	8.3%	0.3056	0.5528
Random Forest	53.3%	0.2193	0.3163
Simple Logistic	48.3%	0.1743	0.3913
IB1	53.3%	0.1556	0.3944
IBk (k = 1)	55.0%	0.1498	0.3716

Exhibit 10: Prediction accuracy results for supply new homes at current sales rate – holdout data.

Exhibit 10 contains accuracy results for the four selected algorithms on the holdout data, which covers the period October 2013 through September 2018. At first glance, accuracy results appear significantly better relative to the prior holdout test on single family housing starts. The four models all produced a percentage correctly classified hovering right around 52%. The OneR strawman produced a correctly classified percentage of 8.3%, which provides a measure of confidence in the more sophisticated algorithms which all handily beat the baseline.

When evaluating model success, it's important to take a deeper dive beyond the output statistics. Exhibit 11 is the confusion matrix for Random Forest, which produced a correctly classified percentage of 53.3% and the lowest root mean square error of 0.3163. 32 of the 60 instances were correctly predicted. 26 of the 28 incorrect predictions are within one cell of the diagonal indicating a correct prediction. The key takeaway: 58 out of 60 instances (96.7%) were either correctly predicted or within one cell of being correctly predicted.

	Months Predicted					
Actual Months (Millions)	**A**	**B**	**C**	**D**	**E**	**E**
A = Up to 4.90 months	3	2	0	0	0	0
B = 4.90 to 6.23 months	22	29	2	0	0	0
C = 6.23 to 7.56 months	2	0	0	0	0	0
D = 7.56 to 8.88 months	0	0	0	0	0	0
D = 8.88 to 10.21 months	0	0	0	0	0	0
E = 10.21 or greater months	0	0	0	0	0	0

Summary of Prediction Results

32	53.3%	**Correct Predictions**
26	43.3%	**Incorrect but Close**
2	3.3%	**Incorrect - Severe**

Exhibit 11: Random Forest confusion matrix results on holdout data (60 instances)

In conclusion, the results indicate that machine learning methods can be leveraged for predicting the supply of new homes. More sophisticated algorithms displayed prediction accuracy results that easily defeated the OneR strawman. Model comprehension was evaluated by reviewing the model constructed by simple logistic, which revealed attribute selections that seem logical and believable. Holdout data results were mixed. The four selected models produced a percentage of correctly classified instances that hovered around 52%, yet easily defeating the strawman's 8.3% accuracy result. More encouraging, the Random Forest confusion matrix indicates that nearly all incorrect predictions were one cell away from the diagonal indicating a correct prediction. The model constructed by Random Forest appears to have predictive ability that could have practical applications for businesses that need to forecast the supply of new homes.

E. Residential Single-Family Housing Permits

This metric represents the number of new residential housing units authorized by building permits (Federal Reserve Bank of St. Louis, 2019). Housing permit trends would likely be beneficial for government agencies engaged in urban planning and needing to predict future demand for public utilities. Exhibit 12 shows housing permit activity from 1976 through 2018, with the units shown for each month representing an annualized number in millions. The graph appears generally aligned to single family housing starts, which makes sense considering that a building permit must be obtained

prior to construction. However, a side-by-side comparison reveals greater monthly volatility in the number of single-family housing starts.

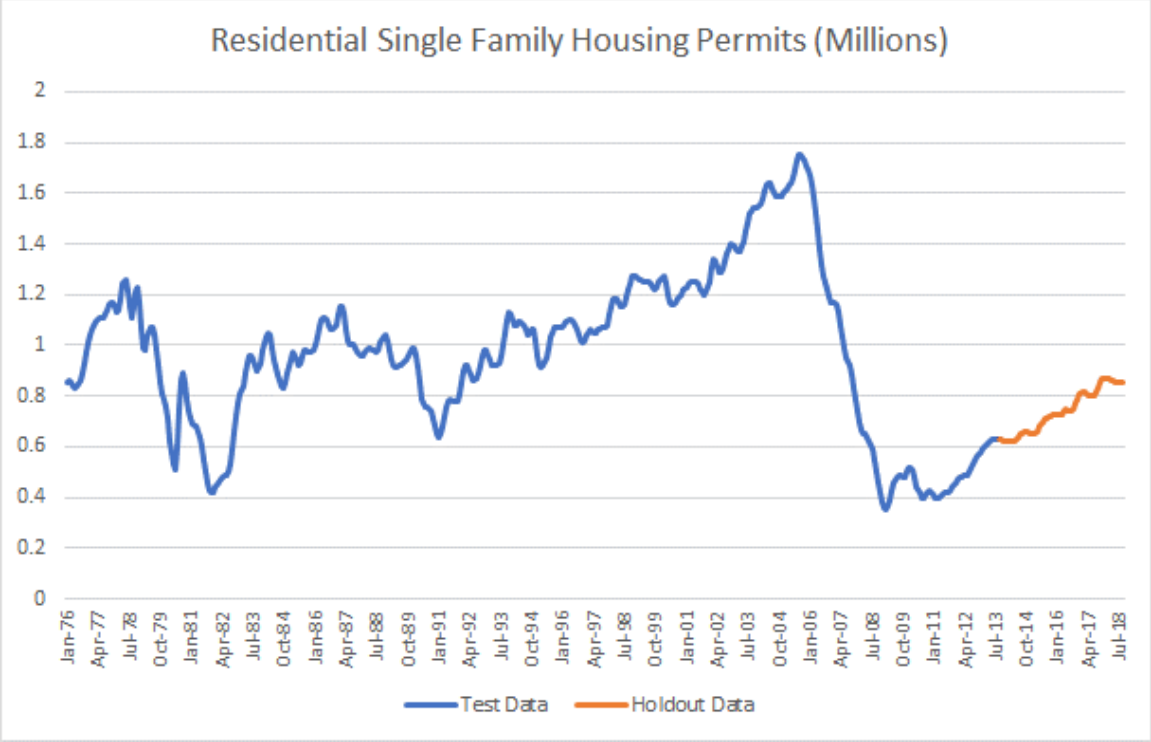

Exhibit 12: Residential single-family housing permits (annualized in millions), January 1976 to Sept. 2018.

Exhibit 13 shows the distribution of the monthly percentage change in annualized housing permits from 1976 through 2018. This graph is generated using the same data as Exhibit 12 but shows the distribution of percentage change in the annualized units for each month. This information could be equally useful for an organization that needs to forecast housing permits. For example, will housing permit activity over the next five years exhibit a similar distribution of activity as seen over a historical time-period? Can machine learning methods be used to accurately predict this activity?

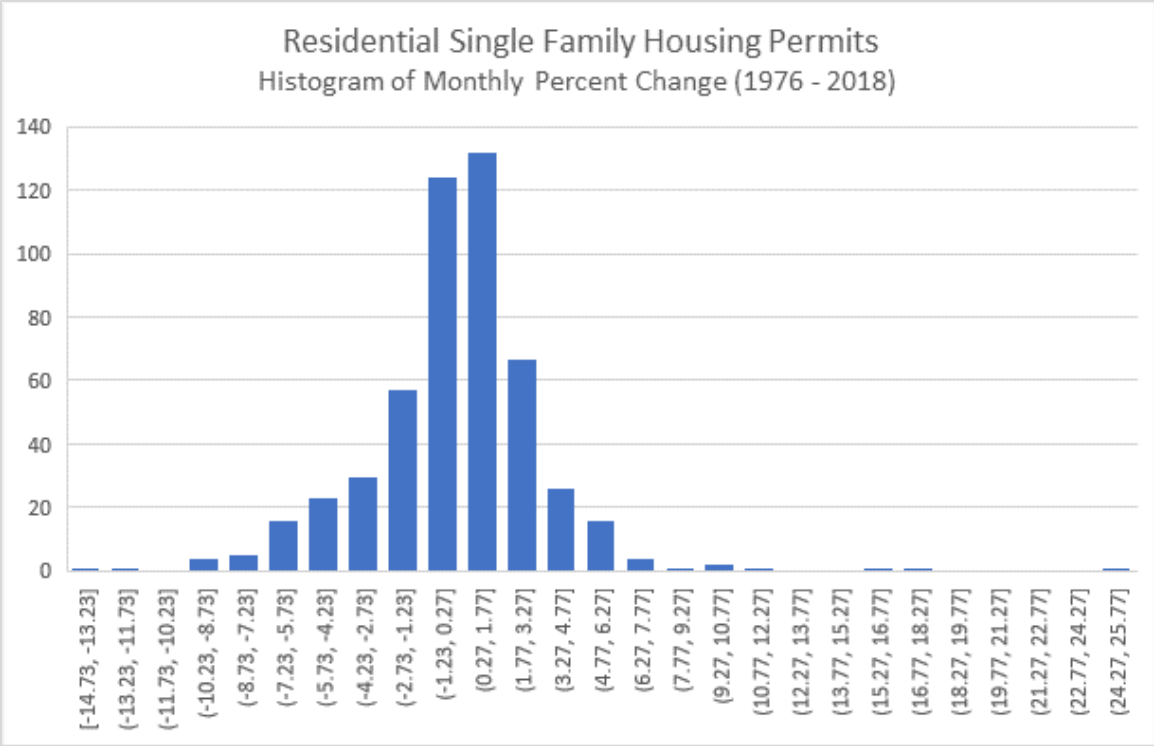

Exhibit 13: Histogram distribution of monthly change in permit activity, January 1976 to September 2018.

Exhibit 14 lists the machine learning algorithms used for this analysis and accuracy results using test data from 1976 through 2013. Excluding the OneR strawman, the average percentage of correctly classified instances was 70.6% and the average mean absolute error is equal to 0.1353. Interestingly, OneR performed within the range of more sophisticated algorithms, with a correctly classified percentage of 63.8% and mean absolute error of 0.1448. The three rule classifiers all performed within a narrow range, with the model generated by each classifier providing a potential explanation for the consistency. Evaluating the OneR model for comprehensibility revealed that it chose *wood products production* as the minimum-error predictor. JRip and Prism, both rule classifiers, also chose *wood products production* within their respective selection of rules.

The selection of *wood products production* was not limited to rule classifiers. J48 is a clear-box decision-tree algorithm in which the higher portions of the tree contain the most predictive attributes. J48 selected *wood products production* as the highest decision node in the tree. The ID3 algorithm, which is another clear-box decision tree learner, also selected *wood products production* at the top of the decision tree model. The wood products category (NACIS 321) is defined as business establishments engaged in sawing

logs into lumber and similar products such as plywood, trusses, and wood flooring (U.S. Census Bureau, 2012). These are all wood materials used to build a house. It seems intuitive that a predictive relationship could exist between lumber/plywood production and residential housing permits.

The most successful algorithms were Random Forest, multi-layer perceptron, IB1, and IBk (k = 1). Those four algorithms, along with OneR, were tested on five years of holdout data to determine how each model would have performed if placed into actual production in a business environment five years ago.

Test Data - January 1976 through September 2013

	Correctly Classified	Mean Absolute Error	Root Mean Squared Error
Decision Trees			
J48	68.9%	0.1385	0.3276
Random Forest	78.6%	0.1452	0.2502
ID3	66.4%	0.1276	0.3572
Rule Classifiers			
JRip	66.0%	0.1678	0.3167
OneR	63.8%	0.1448	0.3805
Prism	62.5%	0.1037	0.3220
Functions			
Logistic Regression	58.7%	0.1660	0.4045
Simple Logistic	68.9%	0.1381	0.2967
Multi-layer Perceptron	77.0%	0.0924	0.2713
SMO	72.0%	0.2525	0.3352
Nearest Neighbor			
IB1	81.9%	0.0724	0.2691
IBk (k = 1)	82.3%	0.0749	0.2601
IBk (k = 3)	79.5%	0.1128	0.2437
IBk (k = 5)	72.0%	0.1455	0.2742

Exhibit 14: Prediction accuracy results for housing permits – test data.

Holdout Data - October 2013 through September 2018

	Correctly Classified	Mean Absolute Error	Root Mean Squared Error
Tested Models with Highest Accuracy			
OneR	70.0%	0.1200	0.3464
Random Forest	61.7%	0.1982	0.3087
Multi-layer Perceptron	45.0%	0.2140	0.4320
IB1	51.7%	0.1933	0.4397
IBk (k = 1)	61.7%	0.1704	0.3918

Exhibit 15: Prediction accuracy results for housing permits – holdout data.

Exhibit 15 contains accuracy results for the four selected algorithms on the five years of holdout data. OneR appears to be the overall winner, with 70.0% correctly classified instances and a mean absolute error equal to 0.1200. The OneR strawman beat out the more sophisticated algorithms, providing additional evidence for wood products production having some level of predictive ability for housing permits. Exhibit 16 contains the confusion matrix for OneR, showing 70% correct predictions and the remaining 30% located one cell away from the main diagonal. The simplicity of OneR also speaks to the concept of Occam's Razor, where simple scientific theories are preferable to complex theories (Witten et al, 2011). In addition to strong accuracy results, the model produced by OneR is simple to explain and comprehend by people who may be less familiar with machine learning methods.

	Predicted Percent Change				
Actual Percent Change	**A**	**B**	**C**	**D**	**E**
A = Less than negative 6.9%	0	0	0	0	0
B = Negative 6.9% to 0.9%	0	31	7	0	0
C = 0.9% to 8.7%	0	11	11	0	0
D = 8.7% to 16.6%	0	0	0	0	0
E = 16.6% or greater	0	0	0	0	0

Summary of Prediction Results

42	70.0%	**Correct Predictions**
18	30.0%	**Incorrect but Close**
0	0.0%	**Incorrect - Severe**

Exhibit 16: OneR confusion matrix results on holdout data (60 instances)

Looking back at the holdout data results, the second-place algorithm was IBk with the number of neighbors set to one. The distribution of data points in the histogram (Exhibit 13) show few outliers in the data, lending support for a smaller k value. Random Forest had the same percentage of correctly predicted instances as IBk (61.7%) but had a modestly higher mean absolute error of 0.1982. Multi-layer perceptron experienced a significant decline in accuracy compared to test data results, with 45% correctly classified and a mean absolute error of 0.2140.

In conclusion, the results suggest that machine learning methods can accurately predict the percentage change in monthly housing permits. In an interesting twist, the OneR strawman defeated more sophisticated algorithms. Taking a broader view, three algorithms (OneR, Random Forest, and IBk) each showed encouraging accuracy results on the holdout data. If either of those three models had been used for actual forecasting purposes five years ago, the resulting permit predictions over the subsequent period would have been more than 61% accurate.

V. Summary

Within this analytics capstone paper, a variety of machine learning algorithms were evaluated for their ability to accurately predict three real estate performance metrics: single family housing starts, supply of housing at current sales rate, and residential building permits. The most successful models were re-evaluated on five years of holdout data, which simulated the impact of placing them into actual forecast usage in a business environment. Random Forest and multi-layer perceptron were consistently chosen as the most accurate algorithms on training and test data. When evaluating holdout data, model accuracy varied considerably between each real estate metric being predicted. However, a closer review of confusion matrices revealed predictions that, in most cases, were either accurate or one cell away from a correctly predicted outcome. The most successful holdout data tests were on housing permit data, with OneR, Random Forest and k-nearest neighbor all exhibiting prediction accuracy above 61%. In a larger sense, the literature review section and prediction results from the three tests confirm that machine learning methods can be successfully used to predict real estate trends and housing valuations.

The data preparation, algorithm testing, model evaluations described in this paper required significant time to undertake. For example, the multi-layer perceptron algorithm required over an hour of processing time to evaluate 71 attributes with 453 data instances. Testing different data combinations might require an entire working day. In a fast-paced business environment, the luxury of time may not always be readily available. What are the key considerations for building a forecasting model when time and resources are limited?

- **Data Preparation**
  Does the data file contain the appropriate attributes needed for prediction?
  Is data formatted consistently? For example, it can be problematic to merge monthly data observations with quarterly or annual data.

- **Data Quality**
  Cleansing of missing and invalid data can require significant time. Ideally, choose data from trustworthy sources where data quality has already been addressed.

- **Data Mining Tool**
  Weka is a free download but can be unstable with very large datasets.
  R has many data mining packages but requires programming knowledge.

- **Algorithm Testing**
  Test as many algorithms as possible. Clear box methods preferable over black box if accuracy is equal. Clear box methods are comprehensible to people.

- **Processing Time**
  Neural network algorithms (i.e. multi-layer perceptron) can take over an hour to process. Decision tree algorithms (i.e. J48) required less than a minute for this project.

- **Testing Different Versions of Data**
  Weka provides numerous options for discretization (i.e. number of bins).
  Adding or removing columns to test different data combinations can be beneficial but requires time.

VI. Appendix

List of Attributes
Rental_Vacancy_Rate
Distressed_Share_of_Repeat_Home_Sales
FHFA_Home_Price_Index_%_Change
New_Home_Sales_Median_Price_%_Change
Housing_Stock_%_Change
Months_supply_new_homes_at_current_sales_rate
Residential_Multi_Family_Permits_%_Change
Conventional_Mortgages_Average_Loan_to_Value_Ratio
Housing_Affordability_ Index_%_Change
PPI_Total_%_Change
PPI_Core_%_Change
CPI_Total_%_Change
CPI_Core_%_Change
CPI_New_Cars_%_Change
CPI_Used_Vehicles_%_Change
CPI_Housing_Index_%_Change
CPI_Male_Apparel_Index_%_Change
CPI_Female_Apparel_Index_%_Change
CPI_Footwear_Apparel_Index_%_Change
CPI_Medical_Care_Index_%_Change
CPI_Prescription_Drugs_Index_%_Change
CPI_Gasoline_Index_%_Change
CPI_Furniture_and_Bedding_Index_%_Change
CPI_Alcoholic_Beverages_Index_%_Change
Prices_Received_by_Farmers_Total_%_Change
Prices_Received_by_Farmers_Crops_%_Change
Prices_Received_by_Farmers_Livestock_%_Change
Gold_Price_Troy_Oz_%_Change
M1_Money_Stock_%_Change
M2_Money_Stock_%_Change
M2_Savings_Deposits_Commercial_Banks_%_Change
Consumer_Credit_Outstanding_%_Change
Net_Immigration_Per_1000_US_Residents
Birth_Rate_Per_1000_US_Residents
Ages_15_19_Annualized_Growth
Ages_20_24_Annualized_Growth
Ages_25_29_Annualized_Growth
Ages_30_34_Annualized_Growth
Ages_35_39_Annualized_Growth
Ages_40_44_Annualized_Growth
Ages_45_49_Annualized_Growth
Ages_50_54_Annualized_Growth
Ages_55_59_Annualized_Growth
Ages_60_64_Annualized_Growth
Ages_65_plus_Annualized_Growth
Industrial_Production_%_Change
Electric_and_Gas_Utilities_Production_%_Change
Durable_Manufacturing_%_Change
Nondurable_Manufacturing_%_Change
Textile_Production_%_Change
Paints_Soaps_Toiletries_Production_%_Change
Wood_Products_Production_%_Change
Appliance_Production_%_Change
Railroad_Rolling_Stock_Manufacturing_%_Change
Furniture_Production_%_Change
Ship_and_Boat_Building_%_Change
Oil_and_Gas_Extraction_%_Change
Coal_Mining_Production_%_Change
Iron_and_Steel_Products_Production_%_Change
Fixed_30_Year_Mortgage_Rate
Bank_Prime_Interest_Rate
US_Treasury_1_Year_Rate
US_Treasury_3_Year_Rate
US_Treasury_5_Year_Rate
US_Treasury_10_Year_Rate
Inflation_Adjusted_GDP_Annualized
Petroleum_US_Demand_%_Change
Effective_Federal_Government_Personal_Tax_Rate
Effective_Federal_Government_Corporate_Tax_Rate
single_family_housing_starts

Single Family Housing Prediction - Data for Test 2
Rental_Vacancy_Rate
Distressed_Share_of_Repeat_Home_Sales
FHFA_Home_Price_Index_%_Change
New_Home_Sales_Median_Price_%_Change
Housing_Stock_%_Change
Months_supply_new_homes_at_current_sales_rate
Residential_Multi_Family_Permits_%_Change
Conventional_Mortgages_Average_Loan_to_Value_Ratio
Housing_Affordability_ Index_%_Change
Fixed_30_Year_Mortgage_Rate
Inflation_Adjusted_GDP_Annualized
single_family_housing_starts

Appendix Exhibit 1: Attributes used for testing

VII. References

Anyoha, R. (2017, August 28). "The History of Artificial Intelligence." Harvard University Science in the News. Retrieved from: *http://sitn.hms.harvard.edu/flash/2017/history-artificial-intelligence/*

Baldominos, A., Blanco, I., Moreno, A. J., Iturrarte, R., Bernárdez, Ó, & Afonso, C. (2018). "Identifying Real Estate Opportunities Using Machine Learning." Applied Sciences, 8, 2321st ser. doi:10.3390/app8112321

Blumenthal, J. (2019, January 16). "Radian acquires Maryland software developer." Philadelphia Business Journal. Retrieved from: *https://www.bizjournals.com/philadelphia/news/2019/01/16/radian-buys-five-bridges-advisors-mortage-insuranc.html*

Greiner, B. (2015, August 20). "There's No Place Like Home - The Housing Market and Economic Growth." Forbes Magazine. Retrieved from: *https://www.forbes.com/sites/billgreiner/2015/08/20/theres-no-place-like-home-the-housing-market-and-economic-growth/#543d6cd87404*

Kok, N., Koponen, E.L., Martínez-Barbosa, C. (2017). "Big Data in Real Estate? From Manual Appraisal to Automated Valuation." The Journal of Portfolio Management. Retrieved from: *https://sustainable-finance.nl/upload/researches/Kok-et-al_Big-Data-in-Real estate.pdf*

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). Foundations of machine learning. Cambridge, MA: The MIT Press.

Mu, J., Fang, W., Zhang, A. (2014). "Housing Value Forecasting Based on Machine Learning Methods." Abstract and Applied Analysis, Volume 2014, Article ID 648047. Retrieved from: *https://dx.doi.org/10.1155/2014/648047*

Olick, D. (2018, September 6). "Decade after housing crash, Fannie Mae and Freddie Mac are Uncle Sam's cash cows." Retrieved from: *https://www.cnbc.com/2018/09/05/fannie-mae-freddie-mac-are-uncle-sams-cash-cows-a-decade-after-crash.html*

Pyle, D. (1999). Data preparation for data mining. San Francisco, CA: Morgan Kaufmann.

Sankara Subbu, Ramesh, "Brief Study of Classification Algorithms in Machine Learning" (2017). CUNY Academic Works. http://academicworks.cuny.edu/cc_etds_theses/679

Suykens, J. (2002). "Least Squares Support Vector Machines." NATO-ASI Learning Theory and Practice. Retrieved from: *https://www.esat.kuleuven.be/sista/natoasi/suykens.pdf*

U.S. Bureau of the Census. (2012). Industry Statistics Portal: NAICS: 321 - Wood product manufacturing. Retrieved from: https://www.census.gov/econ/isp/sampler.php?naicscode=321&naicslevel=3#

U.S. Bureau of the Census and U.S. Department of Housing and Urban Development, Monthly Supply of Houses in the United States [MSACSR], retrieved from FRED, Federal Reserve Bank of St. Louis; https://fred.stlouisfed.org/series/MSACSR, March 9, 2019.

Witten, I. H., Frank, E., & Hall, M. A. (2011). Data mining: Practical machine learning tools and techniques. Amsterdam: Elsevier.

Zakka, K. (2016, July 13). "A Complete Guide to K-Nearest-Neighbors with Applications in Python and R." Retrieved from: *https://kevinzakka.github.io/2016/07/13/k-nearest-neighbor/*