STACKED DENOISING AUTOENCODER BASED PRICE PREDICTION AND CLUSTERING OF REALESTATE PROPERTIES

by

Balachandhar Tirunelveli Nallasivan

Submitted in partial fulfillment of the requirements
for the degree of Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
September 2018

Contents

List of Tables

List of Figures

Abstract

This work develops a framework for residential real estate price prediction and clustering properties in Nova Scotia province. It differs from other studies by covering large geographic area and by defining submarkets. We also used stacked denoising autoencoder to reduce the dimensionality and used two-level clustering approach (self organizing map and K-means) for clustering real estate properties. In addition, we test with different submarkets, different training period, different geographical features, and regularizer to improve the accuracy of the model.

Acknowledgements

First, I would like to express my greatest gratitude to my supervisor, Dr.Stan Matwin, for your patience, guidance, encouragement, advice and support throughout my time in my Master program. Thank you for taking me as your student and providing me precious opportunities to work on the projects. I am so proud to have a supervisor who offers brilliant ideas, invaluable suggestions and directions on my research and studies. Your patient guidance and insightful comments significantly contributed to this thesis.

My sincere gratitude also goes to Ashley Wu, for your constant support, encouragement and constructive suggestions. Your great support and kind advice has greatly helped the accomplishment of the work in this thesis. Thank you for trusting me and always being patient with me.

Thanks to all my friends and lab mates in the Big Data Institute: Xiang, Xuan, Ahmad, Behrouz, Habibeh, Lulu, David, Parsa, Casey, Fateha, Baifan for the discussions and lots of things I learned from you and for all the fun we have had during the past two years. And I would like to thank Dalhousie University for providing a nice environment to do research and study.

I am thankful to my family that always loves and cares about me. To my Mom and Dad, I am really proud to be your child. Thank you for encouraging and supporting me to pursue my dreams. And I am grateful to my sister and friends. Thank you for the love and companionship.

Chapter 1

Introduction

A significant interest has been witnessed in predicting the value of real estate properties. This helps investors to choose the right investment portfolio and invest at right time. Predicting the price of real estate properties will also help sellers to find the right time to sell. Machine learning based valuation models could consider local geographic information to predict the price of properties. Automated valuation models outperformed traditional real estate appraisals.

If we could reflect the conditions in real estate market accurately and valuate the properties in timely fashion, investors and lenders would be among the key beneficiaries. In recent years, machine learning techniques has been used for analyzing real estate data and predicting the prices. This has been done in more and more cities around the world [1].

Currently, properties are valuated based on the sales price. Sales price of proximal similar properties are considered to valuate the property. For example, if a home in one neighbourhood is sold for a certain price, then, similar kind of homes in that neighbourhood would use this sales price as comparable value to substantiate its current value. Properties in the neighbourhood could be of any type - Single family

houses, Apartments, Condo properties. Sales can only be compared with specific type of properties. The underlying problem here is not every neighbourhood would have sales in every fiscal year. So, predicting the market value of a property is difficult with very less or no sales records.

In real estate property price prediction problems, the variable to be predicted does not depend only on variables which are in property listing but also depends on unknown variables which are not known during the training. These unknown variables are sometimes collected by regional government as open source information. This includes mainly geographical features and other amenities in every neighbourhood of the province. This is the main problem in real estate market predictions.

Real estate price prediction problems are approached as regression problem.(i.e.). predicting the price of property P given the attributes X associated with it. These features include house features and features about the neighbourhood. House features includes number of bedrooms, number of bathrooms, living area space, grade etc. Neighbourhood features include average commute time to work, proximity to hospitals, schools and parks etc. This problem has strong spacial structure associated with it which could help to improve the property price prediction, if exploited.

Given a particular neighbourhood, price of property differs based on the size of the house. A small house with one bedroom and one bathroom will be less expensive than a large house with three or four bedroom with 3 bathrooms. In addition to property features, price of the property is also dependent on geographic attributes. One of the most important attribute to be considered is "desirability" of the neighbourhood. The price of a house is determined by comparing with similar kind of houses in the same neighbourhood. For example, a house will have similar price value compared to houses with similar set of features in that neighbourhood but not in other neighbourhood. If a house is located in upscale locality, then, it will have higher "desirability" compared to houses in poor neighbourhood. This implies the houses in upscale locality will have

higher prices compared to poor neighbourhood. Spacial smoothness is associated with the desirability of the neighbourhood: desirability of a location changes as we move from one neighbourhood to another. So, the price will also change according to that. This can be considered as 2D surface with desirability as $2nd$ co-ordinate. Thus, desirability plays a major role in predicting the price of a real estate property. Properties can be clustered to find similar neighbourhoods, to find similar properties and to find how gradually desirability changes from one neighbourhood to another.

To this end, one might ask questions such as the following to predict the price of properties and to cluster them.

1. What kind of regression technique will help to predict the price of properties?

2. What are all the housing features that contribute to the price of the properties?

3. What are all the geographic features that contribute to the price prediction of the properties?

4. Is it possible to cluster the properties and rank the similar kind of properties? If clustering the properties is possible, how many clusters do we obtain overall?

In short, there is still a need to model and predict the property price of all the real estate properties in a province which should be better than automated valuation models. This model should consider current market sales data as one of the feature so the model could learn the current market scenarios. For the purpose of this research, it is necessary to conduct supervised learning and measure the accuracy performance of different regression algorithms. This research also meant to discover previously unknown patterns in the data using unsupervised learning methods.

In summary, this thesis aims to predict the price of real estate properties, in the context of supervised learning and also to cluster the real estate propeties, in the context of unsupervised learning.

To this end, we have proposed the use of a multilayer perceptron configured to act as a Stacked Denoising Autoencder (SDAE),

1.prior to the application of a regression algorithms, a supervised learning approach is explored.

2.prior to the application of K-means, Self-organizing map (SOM), an unsupervised learning approach is explored.

Successive layers of Stacked Denoising Autoencoder may learn increasingly high-level features. While the first layer might learn some first-order features from input data (such as important housing characteristics), a second layer may learn some grouping of first-order features (for instance, by learning given configurations of housing characteristics that correspond to geographic area).

The evaluations on Nova Scotia real estate dataset shows that it is possible to predict the real estate property prices and identify patterns using the proposed approach.

The upcoming chapters in the thesis are organized as follows: research previously conducted in the field of real estate data analysis, using machine learning algorithms are summarized in Chapter 2. The adaptation of known regression and clustering methods to analyze real estate data is presented in Chapter 3. The main contribution of this thesis i.e., is the utility of Stacked Denoising Autoencoders(SDAE) to obtain a optimized representation of input features describing real estate properties is detailed in Chapter 4, along with the visualizations from resulting regression and clustering algorithms. Chapter 5 draws conclusions of the research by providing inference from the results of the proposed framework; and discusses future research directions.

Chapter 2

Background

The discussion in this chapter begins with previous work in real estate data analysis using Machine learning techniques. This provides solid idea that machine learning methods are applied in real estate data analysis and forecasting property prices. Furthermore, the chapter also summarizes the research conducted using different clustering techniques. This chapter concludes by summarizing all the work done in real estate data analysis. This chapter also borrows some literature from economics because the problem of predicting real estate property prices has long history in economics and investments.

2.1 Property agents and economists

The job of predicting the price of properties is usually done by real estate companies such as Realtors, Remax and Royal LePage in Canada and economics department in universities. To predict the price of properties one needs to consider many attributes about the properties and geographic features about the neighourhood. They often predict the property prices throughout the country and segregated by province or

major cities. They predict the price of properties based on variety of factors: Inflation, economic performance, Market behavior, Market sentiment etc. However these predictions are generic and not for individual properties. To have better granular view, it is more important to do prediction separately in every neigbourhood.

Average price forecast	2016	2016 Annual percentage change	2017 Projected	2017 Annual percentage change	2018 Forecast	2018 Annual percentage change
Canada	490,051	10.9	510,400	4.2	503,100	-1.4
British Columbia	691,111	8.6	708,500	2.5	708,600	0.0
Alberta	394,512	0.4	398,600	1.0	397,500	-0.3
Saskatchewan	300,299	0.7	295,100	-1.7	295,100	0.0
Manitoba	277,493	2.6	287,300	3.5	290,300	1.0
Ontario	534,899	15.4	586,600	9.7	573,700	-2.2
Quebec*	283,306	2.9	295,900	4.4	308,400	4.2
New Brunswick	162,716	1.4	167,400	2.9	170,400	1.8
Nova Scotia	221,161	0.8	230,300	4.1	235,000	2.0
Prince Edward Island	178,603	10.7	198,300	11.0	200,000	0.9
Newfoundland	257,974	-6.6	252,200	-2.2	247,300	-1.9

** Provincial weighted average price for Quebec does not affect unweighted national average price calculations. Information on Quebec's weighted average price calculation can be found at: http://www.fciq.ca/immobilier-statistiques-definitions.php*

Table 2.1: Canadian real estate property price forecast[Replicated from fciq.ca]

Average weighted housing price in every province for the year 2016, 2017 and 2018 is shown in Table 2.1. They also show percentage change of average weighted housing price in between two consecutive years.

These general forecasts are not helpful for the buyers and sellers to understand the predictions for a specific neighbourhood. So, it is better to do the prediction of property price based upon its features: Number of bathrooms, Number of bedrooms, Total square foot living area, Outbuilding etc.

2.2 Hedonic regression analysis and Pricing models

Machine learning techniques have been used to forecast the price of real estate properties in many cities, all over the world. Every city has different unique features that contribute to the movement of real estate property prices. These features can be ranked by training a model. Many research groups have published papers relevant to real estate property valuation methods [11][14][17][16]. This adds to the literature and development of real estate property valuation models. These models considered both geographic and spatial features to develop localized model for every neighbourhood.

A residential property can be viewed as combination of many different characteristics that contributes to the overall value of the property. These contributions can be measured and validated to calculate the ultimate value of the property. The concept of finding the feature importance actually dates back to mid-20th century. The individual traits and its contribution to the overall value of the product was investigated by Court [4]. He developed a model to define the price index for automobiles [4]. This kind of models are called hedonic pricing and later improved by Rosen and Lancaster [5] [6].

Microeconomic theory is developed by Lancaster [6]. He investigated the valuation of different products and goods based on different features of the product. He also applied microeconomic theory to real estate and other assets in financial industry. Rosen [5] also investigated similar kind of models to predict the price of the property. He developed functions that relate property characteristics and property value. Rosen concluded that "Hedonic prices are defined as the implicit prices of attributes and are revealed to economic agents from observed prices of differentiated products and the specific amounts of characteristics associated with them" [5].

Rosen considered many housing features to predict the property value. Few important features he considered are: Living room area, Land area, Number of bathrooms, Grade of the house etc. Rosen's principles revolutionized the fundamentals of housing pricing strategy [5].

The features of geographical area and its contribution in the value of the property was investigated by Can [7]. His theory considered house as a multi-dimensional object differentiated into many different attributes; both internal and external. He mainly investigated the impact of neighbourhood in hedonic pricing models. He considered many features about neighbourhood and geographical area such as: percentage of non-white population, unemployment rate in neighbourhood, median household income and percentage of population under poverty level. These variables helped the hedonic pricing model to include current market scenario and made it more accurate in housing price prediction [7].

The existing hedonic pricing regression was expanded by proving that attribute prices are different in different markets by Bajic[8]. He proved that unified hedonic pricing model for all real estate properties in the province will lack in accuracy. He also insisted to develop separate hedonic pricing model for every sub-markets. (i.e) separate models for every neighbourhood [8].

The concept of Bajic has been followed by many other researchers. They proved the fact that having separate models for sub-markets could improve the accuracy of hedonic models. Bajic's model was expanded by Goodman by considering school quality attribute in every neighbourhood [9].

Location specific hedonic models was investigated and developed by Sirmans, Macpherson and Zietz [10]. They considered different geographic borders such as Forward sortation area, Zip code, Neighbourhood and calculated the accuracy of hedonic regression models. They analyzed 25 different Hedonic pricing models to predict housing value [10].

The 20 Characteristics Appearing Most Often in Hedonic Pricing Model Studies				
Variable	Appearances	# Times Positive	# Times Negative	# Times Not Significant
Lot Size	52	45	0	7
Ln Lot Size	12	9	0	3
Square Feet	69	62	4	3
Ln Square Feet	12	12	0	0
Brick	13	9	0	4
Age	78	7	63	8
# of Stories	13	4	7	2
# of Bathrooms	40	34	1	5
# of Rooms	14	10	1	3
Bedrooms	40	21	9	10
Full Baths	37	31	1	5
Fire place	57	43	3	11
Air-conditioning	37	34	1	2
Basement	21	15	1	5
Garage Spaces	61	48	0	13
Deck	12	10	0	2
Pool	31	27	0	4
Distance	15	5	5	5
Time on Market	18	1	8	9
Time Trend	13	2	3	8
** reproduced from Sirman, Macpherson and Zietz (2005)*				

Table 2.2: Stephen Malpezzi's review of Hedonic Models

In Stephen Malpezzi's review of Hedonic Models (2002), he experimented with vast variety of real estate datasets and found the important features that contribute to the price of real estate properties. Top 20 important attributes that contributes to the prediction of property price based on the malpezzi's review of many different hedonic pricing models was shown in Table 2.2. List of attributes Malpezzi chose from Table 2.2 to run his hedonic pricing models was shown in Table 2.3.

Malpezzi's List of Housing Characteristics	
1	Rooms, in the aggregate, and by type (bedrooms, bathrooms, etc.)
2	Floor area of the unit
3	Structure type (single family, attached or detached, if multifamily the number of unites in the structure, number of floors)
4	Type of heating and cooling systems
5	Age of the unit
6	Other structural features, such as the presence of basements, fireplaces, garages, etc.
7	Major categories of structural materials, and quality of finish
8	Neighborhood variables, perhaps an overall neighborhood rating, quality of schools, socioeconomic characteristics of the neighborhood
9	Distance to the central business district, and perhaps to sub-centers of employment; access to shopping, schools and other important amenities
10	Among characteristics of the tenant that affect prices: length of tenure (especially for renters), whether utilities are included in rent; and possibly racial or ethnic characteristics (if these are hypothesized to affect the price per unit of housing serviced faced by the occupant)
11	Date of data collection (especially if the data are collected over a period of months or years)
(Taken from "Hedonic Pricing Models: A Selective and Applied Review" Stephen Malpezzi. Prepared for Housing Economics: Essays in Honor of Duncan Maclennan. April 10, 2002)	

Table 2.3: Malpezzi's List of Housing Characteristics

Below literature review focused on studies that compares different real estate price prediction models for price prediction. The studies we review are given below:

1. A 2004 study of Fairfax Country prices in 1967 through 1991
2. A 2009 study of Los Angeles prices in 2004
3. A 2010 study of Louisville prices in 1999
4. A 2003 study of Auckland prices in 1996

2.3 A 2004 study of Fairfax Country prices in 1967 through 1991

In this study, Bradford Case *et.al.,* [11] build a model to predict the property values specially for single family residences in Fairfax County, Virginia. The data were provided by tax assessors and it contains 60000 records. Different characteristics about the properties are included as features. This includes housing features and GPS coordinates.

Totally, four different models were built in this study.

1. Model 1 is built based on ordinary least square method. It included all the housing features including latitude and longitude to develop the trend surface.

2. Model 2 is based on linear regression. All the housing features are used to develop the linear function. Few variants of the model were also developed which were non-linear with space and time as parameters.

3. Based on previous literatures, Model 3 is also called geostatistical and kriging models. Kriging models are also called as Gaussian process regression [12]. In this model, interpolated values are modeled by Gaussian process using prior covariance's. Many different linear models are developed and co-variance matrix of error term is modeled.

The model was $Y = \beta X + \mu$, where μ is drawn from normal distribution $\mu \sim N(0, \sigma^2 K)$. K is assumed to have a constant main diagonal and be positive definite (we follow the text, equation 8, page 176) [11], X is the set of input attributes, Y be the output value(predicted price). This is a standard hedonic formulation, except that the errors are allowed to be correlated. Estimate for β could be represented as $\beta = (Y' K^{(-1)} X)^{-1} X' K^{-1} Y$. The above function is the maximum likelihood estimated by Dublin [11] for hedonic price regression.(we follow the text, page 176) [11]. Variants of model 3 were developed by excluding and including open data source. The open source data were from tax assessor. The data included census tract information and other geographical attributes such as GPS coordinates of retail shops, schools, colleges [11].

4. Model 4 is set of linear models built separately for every sub-markets. Sub-markets are identified using K-means clustering algorithm.

These models are built by different people and accuracy has been compared. Conclusion drawn from this experiment: One of the Ordinary Least Squares(OLS) model has 6% higher error rate than all the other models [11].

2.4 A 2009 study of Los Angeles prices in 2004

In this study, Factor graphs to capture underlying relational structure in data was developed by Sumit Chopra *et.al.,* [14]. These factor graphs were used by relational regression algorithm to find the underlying relation. Five different models have been developed and compared in this experiment.

1. Model 1 - $K-$nearest neighbour model with value of K equal to 90.
2. Model 2 - Linear regression model with $L2$ regularizer.
3. Model 3 - Locally-weighted regression.
4. Model 4 - Neural network with 2 hidden layers.
5. Model 5 - Relational factor graph.

Accuracy of the model is pretty low for model 1 and very high for model 5.

With most methods, the model is fed with the data of a single sample which is processed independently of the other samples. However a relational structure could be possessed by the data generated by many real world applications. The value of variables associated with each sample not only depends on features specific to that sample, but also on the features and variables of other related samples [13].

Relational factor graph are graphical models that involves iteratively refining the prediction of a sample by propagating information from other samples that are connected to it [3].

Relational factor graph is a bipartite graph which is used to link the price of the house with similar houses in that neighbourhood. By doing this, we could explore the hidden relationship between output price and some sample features [13]. This shows, relational factor graph could predict the property prices accurately compared to other Machine learning techniques [14].

2.5 A 2010 study of Louisville prices in 1999

In this work, Variants of Hedonic pricing models were compared by Steven C. Bourassa *et.al.,* [15]. The data comprises 13,000 Single family residential properties sales in the year 1999 in Louisville. Many important housing features such as Square foot area, lot size, house age are used to predict the price of single family residential properties.

Four different models were developed and compared.

1. Model 1 - An Ordinary Least Squares(OLS) model.
2. Model 2 - A 2-stage Ordinary Least Squares(OLS) model.
3. Model 3 - A Geostatistical model.
4. Model 4 - A Trend surface method. (Only five features were considered. (i.e.). Age, Latitude, Longitude, Lot size, Square foot area.

Different machine learning models has been iteratively developed and tested for the entire market and for different submarkets. This study shows us many exciting results. This study shows narrowly defined submarkets improve the accuracy of the model than broadly defined submarkets. Geostatistical model is considered to be more accurate model than all the other models. Geostatistical models are considered as group of statistical technique used to interpolate a random field based on the values of nearby locations [15].

This study also has its own limitations. This study is only based on one year data and there are only $13,000$ records which is relatively small compared to all other studies [15].

2.6 A 2003 study of Auckland prices in 1996

In this study, housing prices in Auckland, NewZealand was studied by Steven C. Bourassa *et.al.,* [17]. They tried to study the impact of submarkets in the accuracy

of regression algorithms.

They developed regression algorithm to predict the price of properties and did experiments by introducing submarkets as one of the indicator variable. They trained the regression algorithm using the whole city data and then they introduced submarkets of various sizes and repeated the experiment. Their results show that definition of submarket is important in regression studies. It is clear that narrowly defined submarket were not improving the accuracy: for example, every single residential property could be considered as a submarket and all the price history could be used for training. But it is known fact that this model won't learn about the real estate market in general.

Finding the submarket is one of the important attributes to be experimented with. Submarkets are also called sales groups. These sales groups can be defined based on geographic area. For example, we can define all the residential properties within a range as submarkets. They had also defined submarkets as PCA based K-means clusters and compared the results. In this study, Principal component analysis (PCA) is used as attribute transformation technique which is applied to the data before training any clustering method. The final clusters or submarkets were referred as PCA based K-means clusters.

Five models were developed.

1. Model 1 - This model considered data from entire market to train the regression algorithms.

2. Model 2 - This model is exactly similar to model 1, but considered sales group as indicator variable.

3. Model 3 - A separate model for every sales group.

4. Model 4 - A model for first PCA based K-means clusters.

This study explored the importance of submarkets in real estate price prediction [17].

2.7 Contributions of this work

This work improved the literature in many ways.

1. Sales price is normalized among all the properties in that submarket (i.e.). considering the current market trend to predict the price of properties which has no sales record.

2. Experimented with the use of Stacked denoising autoencoder and verified the improvement in regression performance.

3. Submarket is defined on Forward sortation area (FSA) [1], Neighbourhood and Zipcodes. These outcomes were compared.

Some limitations of the study include:

1. Data is proprietary to Property Valuation Services Corporation (PVSC).

2. The framework developed can be used for other real estate valuation studies but it is not fully automated. Example: Hyper-parameter tuning should be done manually based on the data.

[1]FSA is a collection of zip codes in a specific geographic region. First three letters of a zip code defines the name of the FSA [19]. For example B3H would be the name of the FSA which comprises the geographical regions of zip codes B3H 1R2, B3H1R3, B3H1R4 etc

Chapter 3

Price prediction and clustering

A significant solution to predict the value of residential and commercial properties is to employ a supervised learning paradigm. Given a task to predict the value of real estate properties in a province, it demands us to build regression model for the whole province.

On the other hand, we could apply unsupervised learning paradigms to cluster the properties in a province.

The focus of this thesis is to investigate:

1. the feasibility of using traditional machine learning algorithms to predict the real estate property prices.

2. the use of partition and model based clustering methods that borrow principles of competitive learning as well-known examples of unsupervised learning algorithms.

To investigate the first hypothesis, many regression algorithms are chosen and training is conducted. Different testing methodologies has been applied by dividing and

conquering the training process with geographic area as a key factor. The algorithms that we use to train the system are Lasso linear regression [20], Ridge linear regression [21], Random forest regression [25], Ada boost [25], KNN regression [22], Location specific linear regression [25], Decision tree [23], Support vector regression [24].

Genetic programming is used to optimize the hyper parameters are optimized to generate better accuracy. All the hyper-parameters of baselines are selected using validation dataset.

To investigate the second hypothesis, we use competitive learning to cluster the real estate data. Through competitive learning, input data is summarized by a group of prototype vectors. A finite set of randomly initialized prototypes compete to respond to an input vector [28]. The prototype that responds more strongly would be considered as a "winner". Through training, we could make sure that each prototype specializes itself to respond to different type of input vectors. By this way, clusters are formed. Algorithms that uses competitive learning principles are vector quantization (prototypes are plaster centroids) and self-organizing maps (prototypes are neuron weights) [27]. In self organizing map, a set of similar neurons are trained such that each neuron organizes its synaptic weights towards representing its input pattern.

In this chapter, regression solutions from Random forest, KNN regression, Adaboost, Lasso linear regression, Ridge linear regression, Location specific linear regression, Decision tree, Support vector regression and clustering solutions from Self organizing map and KNN are discussed. Friedman's test has been performed which helps us to choose better learning paradigm. These results act as benchmark to develop contribution of this research.

3.1 Real estate datasets

PVSC is a Nova Scotia not-for-profit organization whose mandate is performing assessment of property values in the province, which helps the municipalities to generate property taxes. PVSC is dealing with large data volumes, maintaining more than 17 years of historical data about characteristics of every single residential and commercial property in the province. The data is highly heterogeneous, including structured data collected and managed in a large DBMS, GIS data in a geographical category management system, and even image data. Furthermore, abundant Open Data exists on the Nova Scotia open data portal. This data resource calls for exploitation of the use data integration and analysis methods, resulting in better models and services for both the public and the government.

PVSC assesses the value of properties and submits the report to 51 municipalities in Nova Scotia. Municipality uses these reports for taxation purpose. This data comprises of information about attributes of every house. They have also provided assessment and appeal information. All these data are maintained in an Oracle database. For every year, approximately 600000-700000 property valuation records are generated.

For the purpose of this research, I classify the Nova Scotia real estate dataset into two broad categories (i.e.) discrete and continuous. Attributes that have finite set of values are represented as discrete. Attributes which are continuous in nature are considered continuous variables, e.g. Square foot area. For the purpose of this research, we did the preprocessing [1] in all 15 years data (2001 - 2015). We divided the feature set into two different types: House profile and geographic features.

[1]Preprocessing refers to binarization of discrete variables and normalization of continuous variables to [-1,+1] intervel as mentioned in section 3.2

Feature set from PVSC data (Nova Scotia real estate data) combined with set of geographical attributes taken from Nova Scotia open data portal is shown in Table 3.1. [29].

Table 3.1: Features

Feature Type	Feature
House profile	Neighbourhood
	Municipality
	land area in acres
	Year built/ Effective year
	Floor area
	Living room square foot
	Grade
	Stories
	External wall
	Style
	Heat
	Building square foot
	Fuel
	Total number of rooms
	Total number of bedrooms
	Total number of bathrooms
	Total number of half bathrooms
	Basement
	Basement area
	Garage area

	Land lake front
	Land farm
	Land forest
	Land semi urban
	Land urban
	Outbuilding area
	Unfinished area
Geographic features	Proximity to Schools
	Proximity to Hospitals
	Proximity to Retail stores
	Income level

3.2 Representation of learning algorithms

This section describes the manner in which dataset is represented to the learning algorithms. The property information matrices from the discrete and the continuous set, are normalized to [-1,+1] interval. Normalizing the input to [-1,1] would create stronger gradients. Since data is centered around 0, convergence is usually faster and derivatives are higher. We use neural network models (Stacked Denoising Autoencoder and Self Organizing Map) to reduce the dimensionality and to cluster the data. So, this normalization process would improve the learning [30].

In summary, discrete variables are those that contain binary information (0 or 1). If there are K values in a nominal variable, then K binary variables are created as a end result of that nominal variable. On the contrary, continuous variables are those which describes continued function of values. Example: Square foot of living area.

3.3 Regression algorithms:

To investigate the first hypothesis of the thesis we use traditional machine learning regression algorithms. Below is the brief discussion of the algorithms considered.

3.3.1 Linear regression

Linear regression is a classical method that fits a line/plane through the data in the space of features [41].

Theory: If there are "m" features then the hypothesis is of the form,

$$\mathrm{f}_t(x) = w_1x_1 + w_2x_2 \quad + w_3x_3 + w_4x_4 \quad + w_5x_5 + w_6x_6 \quad + \ldots\ldots\ldots\ldots + w_mx_m$$

$$= w_0 + \sum_{j=1}^{m} w_jx_j$$

The $w's$ are the weights which will be assigned by training linear regression algorithm using our data. Many different methods exist to assign weights optimally. We chose least square method which seeks to find the weight vector to minimize,

$$Err(w) = \sum_{i=1}^{n} (y_i - w^T x_i)^2$$

Here, w and x are column vectors of size $m + 1$.

Rewriting the above equation in matrix notation,

$$f_w X = X_w$$

$$Err(W) = (Y - {X_w}^T)(Y - X_w)$$

Where, X is the $n * m$ matrix of input data and Y is the $n * 1$ vector of output data and w is the $m * 1$ vector of weights. To minimize the error, take the derivative with respect to w to get a system of m equations with m unknowns:

$$\partial Err(w)/\partial w = -2X^T(Y - Xw)$$

Now, solving the equation for W, we get,

$$X^T(Y - Xw) = 0$$

$$X^T = X^T Xw$$

$$\widehat{w} = (X^T X)^{-1} X^T Y$$

where, $\widehat{w}$ are the estimated weights from the closed form solution. These weights are estimated iteratively using gradient descent approach.

Given an initial weight vector w_0, for $k = 1, 2, 3, 4....m$, $w_{k+1} = w_k - \alpha_k \partial Err(w_k)/\partial w_k$, and end when $|w_{k+1} - w_k| < \epsilon$.

Here, parameter $\alpha_k > 0$ is the learning rate for iteration k [41].

3.3.2 Lasso regression

LASSO is a regression algorithm that performs variable selection and regularization to improve prediction accuracy of the trained statistical model. Lasso regression model are originally derived for least square models and to explain the predictability of estimator [42].

Theory:

Consider a sample with N inputs, each has p covariates and single outcome. Let y_i be the outcome and $x_i := (x_1, x_2, x_3, x_4....x_p)^T$ be the covariate vector for the ith

case. To make the above definition true, the objective of lasso regressor is to solve,

$$\min_{\beta_0, \beta} \left(\frac{1}{N} \sum_{i=1}^{N} (y_i - \beta_0 - x_i^T \beta)^2 \right) \text{ subject to } \sum_{j=1}^{p} |\beta_j| <= t.$$

t is a pre-specified parameter that applies the regularization. Let $X_{(ij)} = (x_i)_j$ and x_i^T is the ith row of X, then X is called covariant matrix.

$$\min_{\beta_0, \beta} \left(\frac{1}{N} ||y - \beta_0 - X\beta||_2^2 \right) \text{ subject to } ||\beta||_1 \leq t$$

where $||\beta||_p = \left(\sum_{i=1}^{N} |\beta_i|^p \right)^{(1/p)}$ is the standard l_p norm.

Since $\widehat{\beta_0} = \overline{y} - \overline{x}^T \beta$, so that,

$$y_i - \overline{\beta_0} - x_i^T \beta = y_i - (\overline{y} - \overline{x}^T \beta) - x_i^T \beta = (y_i - \overline{y}) - (x_i - \overline{x})^T \beta$$

Centered variables can be used to create covariance depending upon the measurement scale.

We can rewrite the above equation as,

$$\min_{\beta \in R^p} \left\{ 1/N ||y - X\beta||_2^2 \right\} \text{ subject to } ||\beta||_1 \leq t.$$

in the lagrangian form, $\min_{\beta \in R^p} \left\{ 1/N ||y - X\beta||_2^2 + \lambda ||\beta||_1 \right\}$
with t and λ data dependent.

3.3.3 Ridge linear regression

Ridge regression is also known as weight decay method. It is considered as very useful tool for multi-collinearity problem. It can be derived from least squares with restriction on its parameters. The standard approach to solve the linear equation is implied as:

For a known matrix A and vector b, we wish to find a vector x by defining,

$b = Ax$

is known as linear least squares and seek to minimize the sum of squared residual,

$||Ax - b||^2$,

A regularization term can be added to the above minimization equation:

$||Ax - b||^2 + ||Rx||^2$

R is known as Tikhonov matrix. Usually R would be chosen as a multiple of identity matrix. This is known as L_2 regularization.

For a multivariate normal distribution of x, variables can be transformed as shown in the below equation:

$\widehat{x} = (A^T A + R^T R)^{-1} A^T b$

The effect of regularization may be varied via the scale of matrix R. $\widehat{x}$ is considered as explicit solution of x. One can seek an x to minimize,

$||Ax - b||_P^2 + ||x - x_0||_Q^2$

The generalized problem has optimum solution x^* which can be solved explicitly using the formula,

$x^* = (A^T PA + Q)^{-1}(A^T Pb + Qx_0)$.

or equivalently

$x^* = x_0 + (A^T P A + Q)^{-1}(A^T P(b - Ax_0))$.

where x_0 is the expected value of x, Q is the inverse covariance of x, P is the inverse covariance matrix of b. Q could be formulated using Tikhonov matrix using the formula $Q = R^T R$ and is considered as Whitening filter. [43]

3.3.4 Support vector regression

Support vector machine constructs a hyperplane in high or infinite dimensional space which could be used for regression task. One of the major advantage of using support vector machine for regression task is that it can do the regression task using a polynomial function or Gaussian function instead of linear functions in high dimensional space. Linear SVR, Polynomial SVR and Gaussian SVR are trained to predict the real estate property prices. [44]

Theory: Support vector regression is convex minimization problem. In SVR, number of deviations is maximized until the actual targets are within the stripe, epsilon. Given a training data $(x_1, y_1), (x_2, y_2), ..., (x_l, y_l) \subset X * R$, where X denotes the input patterns. In ϵ - support vector regression our goal is to find a function $f(x)$ with at most ϵ -deviation from the target value y_i in training data [44]. Let us define the linear function $f(x)$ as $f(x) =< w, x > +b$ with $w \in X, b \in R$. We can define this as convex optimization problem:

minimize $1/2(||w||)^2$

subject to $y_i - < w, x_i > -b <= \epsilon$ and $< w, x_i > +b - y_i <= \epsilon$

The above equation assumes that function f approximates all pairs (x_i, y_i) with ϵ precision. This assumes that convex optimization problem is feasible to solve. Vapnik and Cortes introduce slack variables e_i and $e_i{}^*$ to handle the constraints of

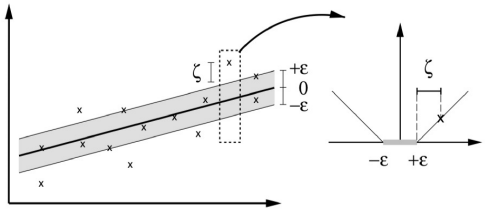


Figure 3.1: The soft margin loss setting for a linear SVM. [44]

optimization problem. Hence the below equation is derived:

minimize $1/2(||w||)^2 + C\sum_{i=1}^{l}(e_i + e_i{}^*)$

subject to $y_i - < w, x_i > -b <= \epsilon + e_i, \quad < w, x_i > +b - y_i <= \epsilon + e_i{}^*,$ and $e_i, e_i{}^* >= 0$

The constant C should be greater than 0. This denotes the way it would tolerate the deviations larger than ϵ.

3.3.5 K-Nearest Neighbours

K-Nearest Neighbours is non-parametric lazy learning method. Input consists of k nearest neighbours in the feature space. Output is the average of the target variable of k-nearest neighbours [47].

Approach: It stores all the input feature vector and the target variable in memory. For each unlabeled input feature vector, K-nearest neighbours are chosen from the training set based on the Euclidean distance in m-dimensional feature space. If all the features are normalized, each feature will have equal weight in the calculation of Euclidean distance and so there won't be any dominating features. This is explained in detail in section 3.2. For example, consider two input vectors X and W, their distance is defined by the below formula,

$$d(x, w) = \sqrt{\sum_{i=1}^{m}(x_i - w_i)^2}$$

In previous studies it has been shown that there is a very strong spacial dependency based on geographical distance and it has significant impact on the predicted value [47].

KNN regression algorithm:

Predict(X, Y, x) $//X$: training data, Y : target variable of X, x : unknown sample

for $i = 1$ to m do

Compute distance $d(X_i, x)$

end for

Compute set I containing indices for the smallest distances $d(X_i, x)$.

Return average predicted value for Y_i where $i \in I$.

3.3.6 AdaBoost algorithm

AdaBoost also known as Adaptive boosting is used in conjunction to many other machine learning algorithms to improve performance. Output of many different weak learners are combined into a weighted sum that represents the output of boosted regressor. Main advantage of AdaBoost algorithm is final model would converge to be a strong learner as long as the performance of each weak learners are slightly better than random guessing [45].

Theory:

Given a set X and Y, let P be the probability distribution on (X, Y). An N-sample is a sequence $< (x_1, y_1), ...(x_n, y_n) >$ with (x_k, y_k) belongs to (X, Y); We call all $Samp_N$ the set of all $N - samples$ and $Samp = Union_N(Samp_N)$. Given a class

of functions $H \subseteq \{f|f : X- > \{0,1\}\}$, a learning algorithm A on H is a function $A : Samp- > H$ [45].

Let $S =< (x_1, y_1), ..., (x_n, y_n) >$ be a $N - sample$ and let $h = A_s$ be the hypothesis output of A on input S, the empirical error $\widehat{\epsilon}$ of A on S is,

$$\widehat{\epsilon} = \frac{\#\{(x_k, y_k)|y_k \neq h(x_k)\}}{N}$$

while the generalization error is,

$$\epsilon_g = P\{y \neq h(x)\}.$$

Under weak conditions on H choosing elements of (X, Y) randomly and independently according to P, for sufficiently large samples, the empirical error is close to generalization error with high probability. For this reason, a good learning algorithm should minimize the empirical error. Often this is a difficult task because of large amount of computational resources, computationally efficient algorithms are moderately accurate.

ADABOOST Algorithm:

Input: a N-sample $S =< (x_1, y_1), ...(x_N, Y_N) >$, a distribution D on S, a learning algorithm A, an integer T.

Initialize: the weight vector $w_i^1 = D(i)$ for $i = 1, 2, 3, ...N$.

Do for $t = 1, 2, 3, ..., T$

1. Set $p^{(t)} = w^{(t)} / \sum_{i=1}^{N} w_i^{(t)}$.
2. Choose randomly with distribution $p^{(t)}$ the sample $S^{(t)}$ from S; call the learning

algorithm A, and get the hypothesis $h_t = A_s^{(t)}$.

3. Calculate the error $\epsilon_t = \sum_{i=1}^{N} p_i^{(t)} |h_t(x_i) - y_i|$.

4. Calculate $\beta_t = \epsilon_t / (1 - \epsilon_t)$.

5. Set the new weights vector to be: $w_i^{(t+1)} = w_i^{(t)} \beta_t^{1-|h^{(t)}(x_i)-y_i|}$.

Output the hypothesis $h_f = \left(\sum_{k=1}^{T} (log(1/\beta_t) h_t(x) - 1/2 \sum_{k=1}^{T} (log(1/\beta_t) \right)$

An upper bound to the error $\epsilon = \sum_{k=1}^{N} D(k).|h_f(x_k) - y_k)|$ of the hypothesis h_f output is bounded by,

$\epsilon <= 2^T \prod_{t=1}^{T} \sqrt{\epsilon_t (1 - \epsilon(t)}$

3.3.7 Random forest regressor

Random forests or random regression trees are an ensembling learning method for classification and regression problems. They are trained by developing multitude of regression trees at training time and output the mean prediction of the individual trees as output [46].

The introduction of random forests was first made in a paper by Leo Breiman [68]. In this paper, he describes a method to build forest of uncorrelated trees using CART procedure and it also explains ways to combine with randomized node optimization and bagging. This paper forms the basis for modern day random forest. Few concepts described in this paper includes:

1. Using out-of-bag error as an estimate of the generalization error. 2. Measuring variable importance through permutation. The report also offers the first theoretical result for random forests in the form of a bound on the generalization error which depends on the strength of the trees in the forest and their correlation.

Theory:

Random forest is based on general technique of bootstrap aggregating or bagging. Given a training set $X = x_1, x_2, x_3.....x_n$ with output target variables $Y =$

$y_1, y_2, y_3, y_4, y_5.....y_n$, bagging algorithm is a logic which repeatedly selects random sample of training sets and fits the trees to these samples [46].

For $b = 1, 2, 3....B$, recursively execute the below two steps,

1) Sample "n" training samples with replacement from the actual dataset(X, Y). Let's call this (X_b, Y_b).

2) Train a regression tree f_b on X_b, Y_b. Every regression tree is built by selecting random subset of attributes for selecting the best split on each tree node.

Predictions of individual trees are averaged to predict the output of unseen samples (x^{new}) .

$$\widehat{f} = 1/B \sum_{b=1}^{B} f_b(x^{new})$$

Bootstraping procedure decreases the variance of the model which leads to better model performance without increasing the bias. Predictions using single tree are highly sensitive to noise from training data whereas predictions from average of many trees is not., as long as trees are not correlated. Training many trees might create strongly correlated trees so bootstrap sampling is used as a way to de-correlate the trees by randomly training them with different training sets.

Uncertainty of prediction is can be measured using standard deviation of prediction which is given by below formula,

$$\sigma = \sqrt{\sum_{b=1}^{B} (f_b(x^{new}) - \widehat{f})^2 / B - 1}$$

The number of trees, B, is a free parameter. Typically, a few hundred to several thousand trees are used, depending on the size and nature of the training set. An optimal number of trees B can be found using cross-validation, or by observing the out-of-bag error: the mean prediction error on each training sample X_i, using only

the trees that did not have X_i in their bootstrap sample. The training and test error tend to level off after some number of trees have been fit [46].

3.4 Clustering algorithms

Clustering is a method to group similar objects as one cluster. Objects in one cluster is more similar to each other than to objects in other clusters. To cluster the real estate properties, we train K-means clustering and Self organizing map clustering algorithms to cluster the properties in Nova Scotia province.

3.4.1 K-means Clustering

K-means clustering algorithm is a centroid based partitioning method. K initial seeds are required to group N exemplars. These seeds represent the centroid of the K clusters [47].

Theory:

The algorithm randomly assigns a position to cluster seeds and begins by calculating the pairwise distance from each exemplar to all the cluster seeds. The position of the cluster centroid is updated after every epoch based on the exemplars in that cluster. Formally, consider a training set of N samples: $x_1, x_2, x_3, x_4....x_n$. Assume $c_1, c_2.....c_k$ be the centroids such that $x_i, c_i \in R^n$. Clustering label update and centroid updates as follows,

$$l_i = \arg\min_{\mathbf{j}} ||x_i - c_j||^2$$

$$c_j = \sum_{i=1}^{m} (l_i = j).x_i / \sum_{i=1}^{m} (l_i = j)$$

where l_i is the cluster label in ith exemplar. The efficacy of the result depends heavily of the value of K and the distance function. Optimum number of clusters should be estimated by testing the cluster qualities within a desired number of clusters (K). Cluster qualities can be measured using silhouette value. A high silhouette value of +1 denotes clusters are well separated. Conversely, a silhouette value of -1 denotes clusters overlap with each other. The algorithm is run multiple times with different K value and the average silhouette value is computed. [47].

3.4.2 Self-organizing Maps

Self-organizing map is a model-based data analysis method. SOM excites the same neuron for similar kind of inputs. By doing this, similar input samples are grouped to belong to similar models. Distribution of data is summarized by a framework for unsupervised learning. Every input data is mapped to a best matching unit updating the neighbourhood and project that into a 1D or 2D topology. Formally SOM algorithm for n-dimensional input K can be summarized as follows:

Theory:

1. Initialize a 2D lattice of neurons each with $n-$dimensional weight vector. We know that the lattice and neighbours of its neurons are predefined. Lattice is defined as a 2D map and the weight vector defines the distance to every input vector.

$$K = \{k_i | k_i \in R^n\}$$

2. Begin SOM training to align the neural map.

(a) For each training exemplar $k_i \in \{K\}$ find the Euclidean distance metric and map the input to BMU (Best Matching Unit). By doing this for all the inputs, data

is clustered. Create a list of Best Matching Units and its corresponding data instance in B

$$BM(b_i) = \{k_j \in \{K\} | \arg\min_{\mathbf{i}}(||k_j - b_i||^2)\}$$

(b) At each iteration the weight vector within the neighbourhood function decreases linearly. For neurons outside the neighbourhood the weight remains unchanged.

$$\vec{w}_{m_i}^{\mathrm{t}} = \vec{w}_{m_i}^{\mathrm{t-1}} + \alpha * ||\widehat{BM(m_i)} - \vec{w}_{m_i}^{\mathrm{t-1}}||^2$$

Where $\widehat{BM(m_i)}$ is the average of all the data mapped to a specific Best Matching Unit(BMU). Whenever SOM finds the BMU for an input vector all neurons within the lattice neighbourhood is updated. The size of neighbourhood incrementally decrease as training epochs increase.

3. Repeat step 2 until $N_f < N_c$. This denotes the fine training of SOM.

4. Repeat step 3 with recent neighbourhood radius value. (R_s)

Post training, we have all the inputs mapped to BMU's based on the bset hit. Using the resulting vector, the most frequently 'hit' neurons can be identified [48].

3.5 Experimental setup

3.5.1 Regression experimental setup

To forecast the price of properties in Nova Scotia, Machine learning models are developed by training with previous year data and testing with subsquent year data. For example, If the model is trained using 2011 data, then it is tested using 2012 data.

Likewise, different models are developed considering 2011 to 2016 data as training data. Average Normalized mean square error of all these regression models were used to calculate the regression accuracy percentage.

Normalized mean square error = $1/n(\sum_{i=1}^{n}(x_i - y_i)^2)/(1/n - 1(\sum_{i=1}^{n}(x_i - \bar{x})^2))$

where $\bar{x} = 1/N \sum_{i=1}^{n} x_i$; x is the assessment price of the property and y is the approximation of x or in other words it is the property price predicted by the model.

Prediction accuracy or regression accuracy percentage = (1−Normalized mean square error) $*$ 100

3.5.2 Clustering experimental setup

Properties in Nova Scotia has been clustered using two-step clustering approach (SOM and K-means clustering). Iteratively, we ran the training process many times with varying number of epochs to find the best outcome. Silhouette value is used to measure and find the best model and optimized number of clusters. The weigths of neurons generated in SOM is passed as feature attributes to K-means for further clustering.

To quantify the extent of disimilarity between clusters, we use Normalized Mutual Information (NMI). NMI value varies between 0 to 1. Two clusters are said to be distinct if the value of NMI is 0. Two clusters are said to be same if the value of NMI is 1. Dissimilarity is the complement of NMI.

The degree of uniqueness between the clusters is defined by the formula,

Exclusive rate $(\mathrm{ER})_i = \sum_{j=1}^{n} D_{ij}$, where "n" is the number of clusters and D_{ij} denotes the dissimilarity between clusters.

The above equation help us to answer - "How different is this cluster compared to all other cluster?"

3.6 Baseline results and discussion

Results have been generated to predict the property price for future years using regression techniques mentioned in section 3.3. Average Normalized mean square error of all these models are shown as results in this section. Properties in Nova Scotia has also been clustered using clustering techniques mentioned in section 3.4.

3.6.1 Regression results

Normalized mean square error is used to measure the error of the model which predicts the property prices in Nova Scotia.

Normalized mean square error $= 1/n(\sum_{i=1}^{n}(x_i - y_i)^2)/(1/n - 1(\sum_{i=1}^{n}(x_i - \bar{x})^2))$

where $\bar{x} = 1/N \sum_{i=1}^{n} x_i$; x is the assessment price of the property and y is the approximation of x or in other words it is the property price predicted by the model.

Prediction accuracy or regression accuracy percentage = (1−Normalized mean square error) $*$ 100

Normalized mean square error might be greater than 1 if the model is performing worse than the baseline[2]. Fortunately none of the models developed in this experiment performed worse than the baseline. The prediction accuracy generated by random forest and ada boost algorithms by passing the raw input is shown in figure 3.2. The prediction accuracy generated by different regression models are shown in figure 3.3. Comparing the regression results in figure 3.2 and figure 3.3, we could confirm that there is a slight increase in prediction accuracy if we do preprocessing[3] . The above result is the average value of prediction accuracy for seven consecutive years (2011 - 2017). Machine learning model is trained using one-year data and tested using subsequent year data. For example, Train the model using 2015 data and predict the

[2]Typically, the mean of the assessment property price on the training set is used as baseline.

[3]Preprocessing refers to binarization of discrete variables and normalization of continuous variables to [-1,+1] intervel as mentioned in section 3.2

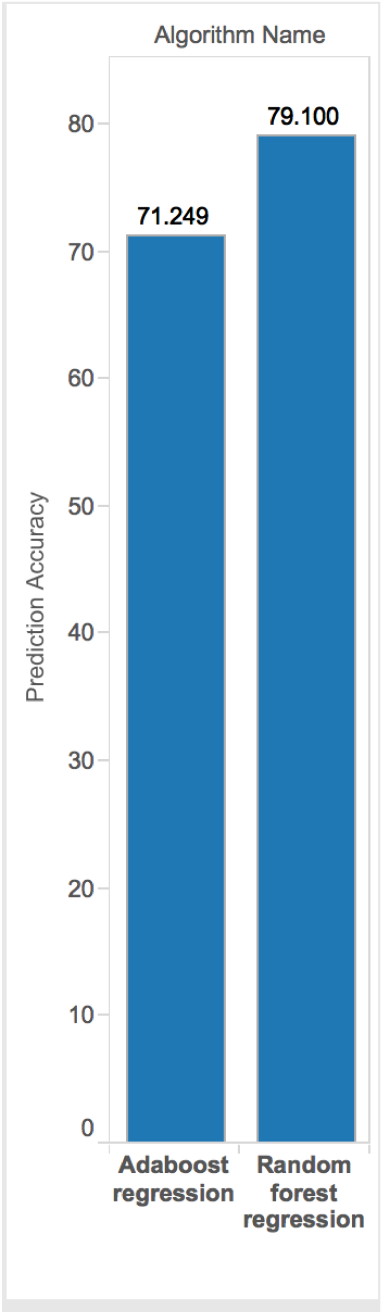


Figure 3.2: Regression algorithm prediction accuracy comparison - using raw data as input

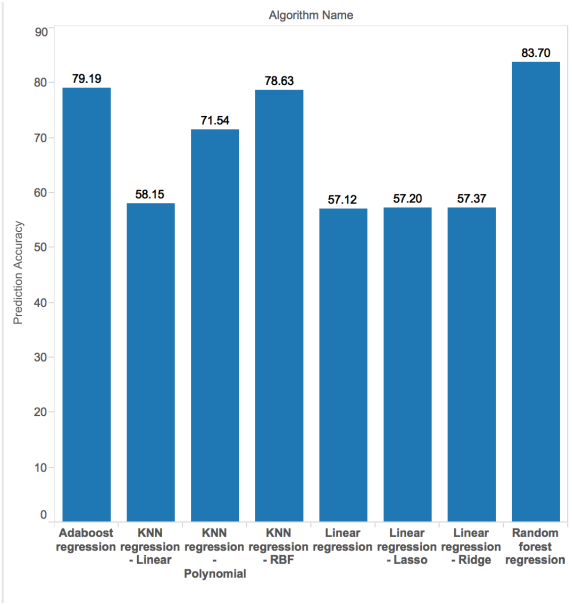


Figure 3.3: Regression algorithm prediction accuracy comparison

property price for 2016 data. Random forest regression model has high regression percentage compared to all other regression models.

Along with regression results, the features which contributed to predict the property price has been noticed. For a forest, the impurity decrease from each feature can be averaged and the features are ranked according to this measure. This is how we extracted top five features that have high contribution to the price of the property. Figure 3.4 shows the top five features that has high contribution in predicting the price of the property.

There are approximately 520000 residential properties in Nova Scotia. I have compared the difference between predicted price and actual assessment price. Figure 3.5 shows the count of properties in each bucket[4].

Real estate market is highly dynamic and the price changes drastically every year.

[4]Each bucket holds set of properties based on the percentage difference between predicted price and assessment price.

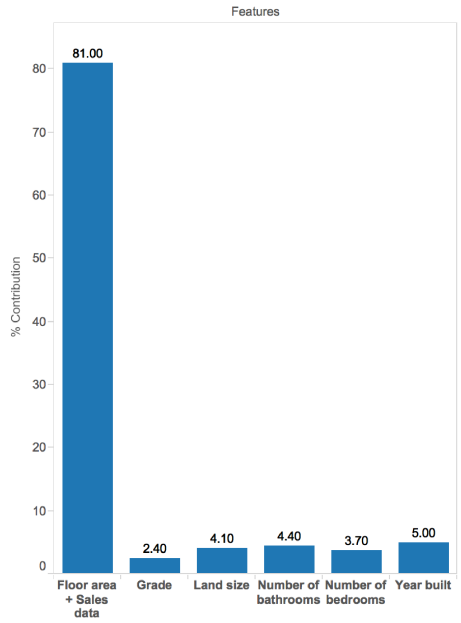


Figure 3.4: Contribution by features to predict property prices

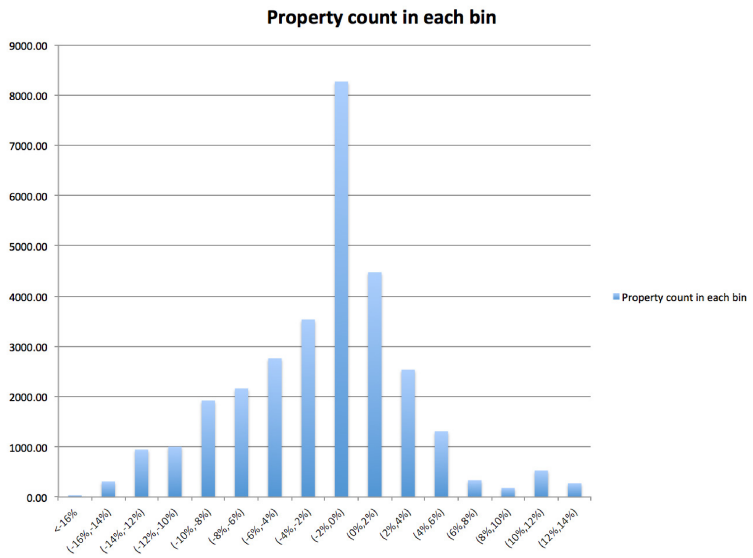


Figure 3.5: Count of Properties in each bucket

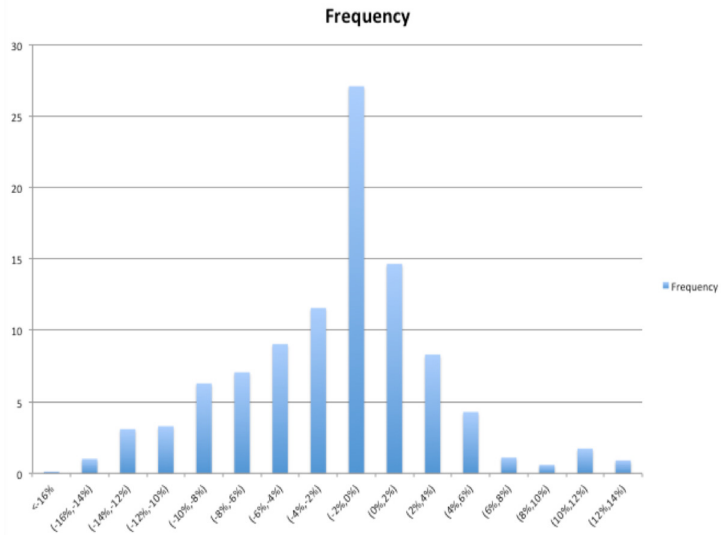


Figure 3.6: Sales price vs predicted price - Percentage of properties in each bin

To make sure our machine learning model will predict the property prices without any dip in accuracy, I compared the difference between predicted price and sales price. There are over 30538 properties in Nova Scotia with valid sales record in the year 2015. Figure 3.6 shows the percentage of properties in each bin[5] based on the difference between predicted price and sales price. (properties which are sold in that year are considered.)

We also developed regression model (Random forest regression model) separately for every municipality. Prediction accuracy is shown separately for every municipality in figure 3.7.

[5]Each bin holds set of properties based on the percentage difference between predicted price and sales price.

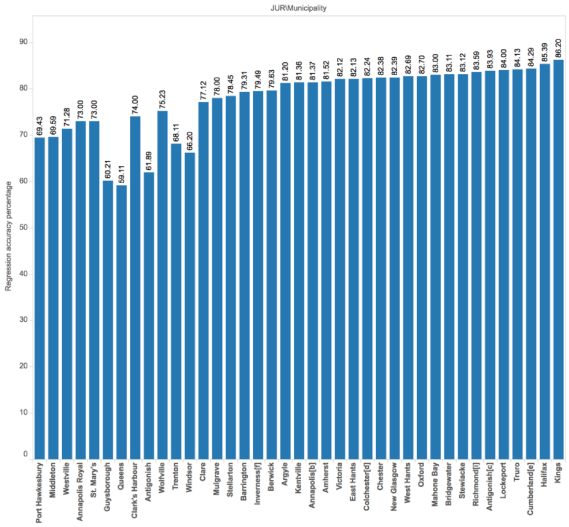


Figure 3.7: Property price prediction accuracy in every municipality

3.6.2 Clustering results

To quantify the extent of disimilarity between clusters, we use Normalized Mutual Information (NMI). NMI value varies between 0 to 1. Two clusters are said to be distinct if the value of NMI is 0. Two clusters are said to be same if the value of NMI is 1. Dissimilarity is the complement of NMI.

The degree of uniqueness between the clusters is defined by the formula,

Exclusive rate $(\text{ER})_i = \sum_{j=1}^{n} D_{ij}$, where "n" is the number of clusters and D_{ij} denotes the dissimilarity between clusters.

The above equation help us to answer - "How different is this cluster compared to all other cluster?" The average exclusive rate of Two-step-clustering is 23.30. Average exclusive rate of 23.30 refers to the average dissimilarity between the clusters. If the proposed clustering framework could generate average dissimilarity or exclusive rate greater than 23.30 then we could prove that the proposed clustering framework is better than the baseline clustering approach.

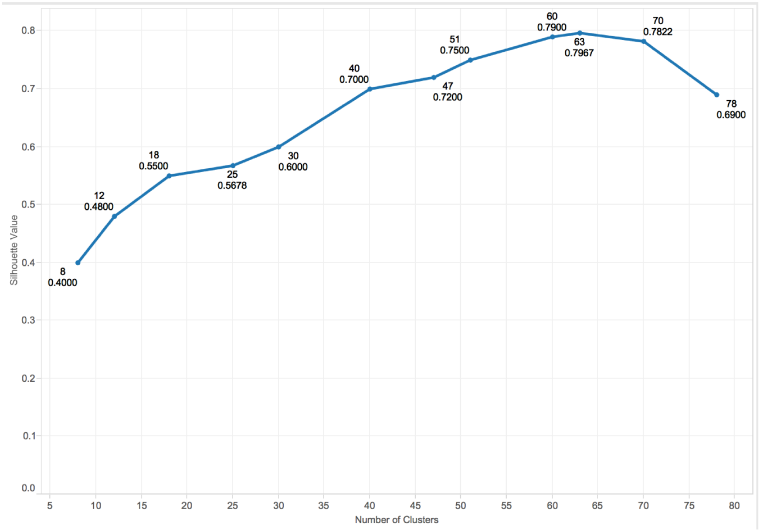


Figure 3.8: SOM clustering - Number of clusters vs Silhouette value

Two step clustering of SOM and K-means generates 63 clusters and 35 clusters with silhouette value 0.79 and 0.84 respectively is shown in figure 3.8 and figure 3.9.

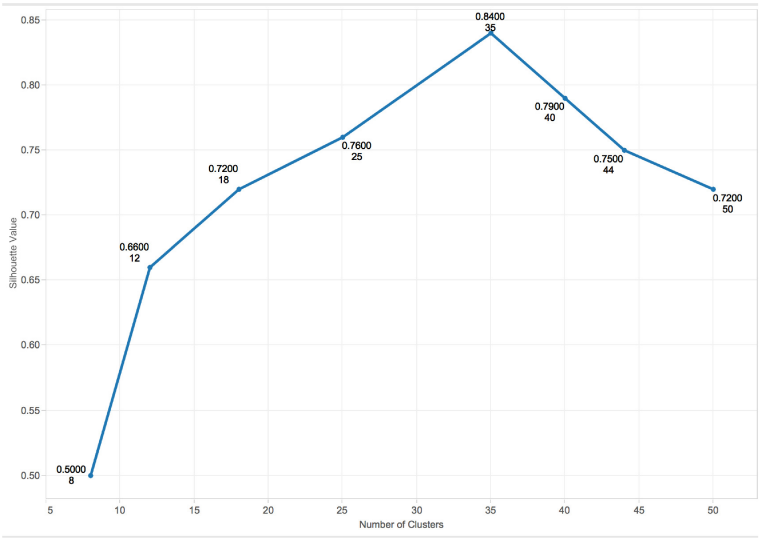


Figure 3.9: K-means clustering - Number of clusters vs Silhouette value

3.6.3 Summary

Henceforth in this research, Random forest is continued to be used as regression method and K-means, SOM are continued to be used as clustering method. In order to improve the prediction and clustering ability of high dimensional real estate data the use of encoding is investigated.

Chapter 4

Proposed method: SDA based regression and clustering framework

In this chapter, the proposed framework to improve the prediction of property prices and clustering of properties by using an encoding mechanism is explained (see Figure 4.1). The hypothesis of this research is to use Stacked Denoising Autoencoder in providing higher level representation of inputs which also improves the performance of regression and clustering. Random forest is trained with the encoded inputs to predict the property price. The clustering is carried out as a two-level approach where we first cluster the real estate properties in Nova Scotia using the SOM, and then, the SOM is clustered using KNN. Using Stacked denoising autoencoder to encode the inputs has several advantages. Some of them are mentioned below [54],

1. It reduces the training time and data storage space required.

2. Removal of multi-collinearity improves the performance of the machine learning model. If two or more explanatory variables are highly linearly related then it is

called multi-collinear variables.

3. It becomes easier to visualize the data when reduced to very low dimensions such as 2D or 3D.

The uniqueness of this approach is to drop the output layer of the stacked denoising autoencoder post training, and introduce regression and clustering algorithms. The corresponding architecture is illustrated below (Figure 4.1):

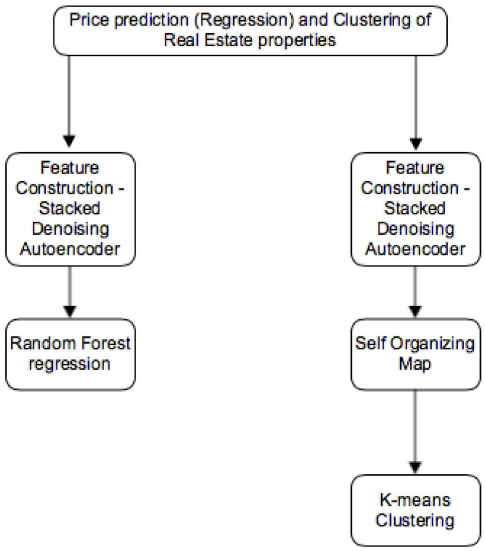


Figure 4.1: Proposed framework Architecture

4.1 Theory of Stacked denoising autoencoder

Stacked Denoising Autoencoder is an extension of Stacked Autoencoder. Denoising Autoencoders are stacked so the output of the first denoising autoencoder is passed as input to second and so on. Each denoising autoencoder is trained to reduce the reconstruction error [53]. Once the first k layers are trained, we can train the $k+1$-th layer.

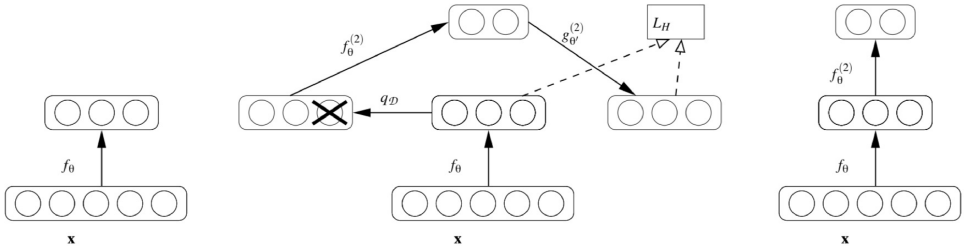


Figure 4.2: Stacked Denoising Autoencoder diagram [31]

The Denoising Autoencoder algorithm for $n-$dimensional input D, can be summarized as follows [53]:

1. Initialize random weights to the network.

2. Corrupt the input vector by injecting noise. Noise is stochastically added to the input data. Stochastic corruption process randomly selects some of the inputs and sets that to zero. The denoising autoencoder then tries to predict these corrupted inputs and correct them. This can be done by considering the uncorrupted inputs and capturing the joint distribution between corrupted and uncorrupted inputs (Gibbs sampling).

3. Define the encoding function $E = f(\vec{w}.\ \vec{d} + \vec{b})$ where $f(x) = (2/(1+e^{-2x})) - 1$, $\vec{d}$ is the neuron input vector, $\vec{w}$ is the weight vector between two consecutive layers of the Multi-Layer Perceptron and $\vec{b}$ is the bias vector. Encoding function is defined

as a mapping from R^n to R^h such that $h < n$.

4. Define the decoding function D that maps $E(d)$ to d. This maps the encoded output to output layer. In this step, stochastically we try to undo the corruption and reconstruct the input.

5. Learning continues until maximum number of training epochs or optimal minima is reached.

Many denoising autoencoders are stacked and executed in sequence. This help us to define much better higher level representation [49].

Types of corruption considered:

1. Additive isotropic Gaussian noise $(GS) : x \mid x\ N(x, \sigma^2 I)$;

2. Masking noise (MN) : a fraction v of the elements of x (chosen at random for each example) is forced to 0;

3. Salt-and-pepper noise (SP) : a fraction v of the elements of x (chosen at random for each example) is set to their minimum or maximum possible value (typically 0 or 1) according to a fair coin flip [49].

4.2 Procedure

As discussed earlier, the goal is to explore ways to combine Stacked denoising autoencoder with regression and clustering algorithms to improve the performance in real estate property price prediction and clustering. The Stacked Denoising Autoencoder, reconstructs the input vector using Multi-Layer perceptron configuration. This ensures to encode the most important attributes and finally this improves the price prediction. The resulting encoded features are input to regression and clustering algorithms.

We train both regression and clustering models using the dataset which contains information about all the residential properties in Nova Scotia. We combine these in-

formation with few other geographical factors such as proximity to schools, hospitals, retail shops etc. Complete list of housing and geographical attributes we use to train the regression and clustering algorithms is shown in Table 3.1. Housing profile data is gathered by Property Valuation services corporation whereas geographical features are gathered from Nova Scotia open data portal. We train the regression and clustering algorithms using one-year data and the subsequent year's data is used for testing the model.

Parameter configuration for the proposed framework is shown in Table 4.1. Parameters mentioned in the table could be used to repeat the experiment in the future.

Table 4.1: Parameter configuration for the proposed framework

Parameter configuration for the proposed framework			
Module	Software	Parameter List	Parameter Value
Preprocessing	Python	Normalization	[-1,+1]
Encoding	Python; Tensorflow package	Epochs; Number of hidden layers; Activation function; Back propagation; Error	500; 2; tansig; Conjugate gradient; Used both Mean square error and Negative log likelihood
Random Forest regression	Python	Number of estimators; Criterion	1024; Mean squared error
K-means clustering	Python	Initialization; Maximum iterations	10; 300

Self organizing map	Python	Initial weights; Lattice size; Neighbourhood radius; Convergence	Principal components initialization; $5\sqrt{Numberofsamples}$; N_c:[0.2*long edge,0.05*short edge], N_f : [0.05*short edge, 0.05*short edge]; Until codebook ceases to change; N_s: 0.5*short edge

Stacked denoising autoencoder empirically finds the number of neurons in each hidden layer by optimizing the hyper-parameters using genetic algorithm. The number of neurons predicted for hidden layer 1 is used as an upper bound while empirically trying to predict the number of neurons in second hidden layer. Input (as mentioned in section 3.2) is passed to hidden layer 1 and hidden layer 2 and then the input is regenerated by which reconstruction error (Negative Log Likilihood) is measured as shown in figure 4.3. The output of hidden layer 2 is passed to the next autoencoder which is stacked to the right of the first autoencoder (see figure 4.3). We could stack any number of autoencoders until there is an improvement in the reconstruction error. The reconstructed input (output of Stacked denoising autoencoder) is passed to regression and clustering algorithms.

Negative log likelihood function is $-\sum_{j=1}^{N} y_j log\widehat{(y_j)}$

$\widehat{y}$ represents discrete probability distribution over the possible values of the observation. y can also be interpreted as a probability distribution over the same space, that just happens to give all of its probability mass to a single outcome.

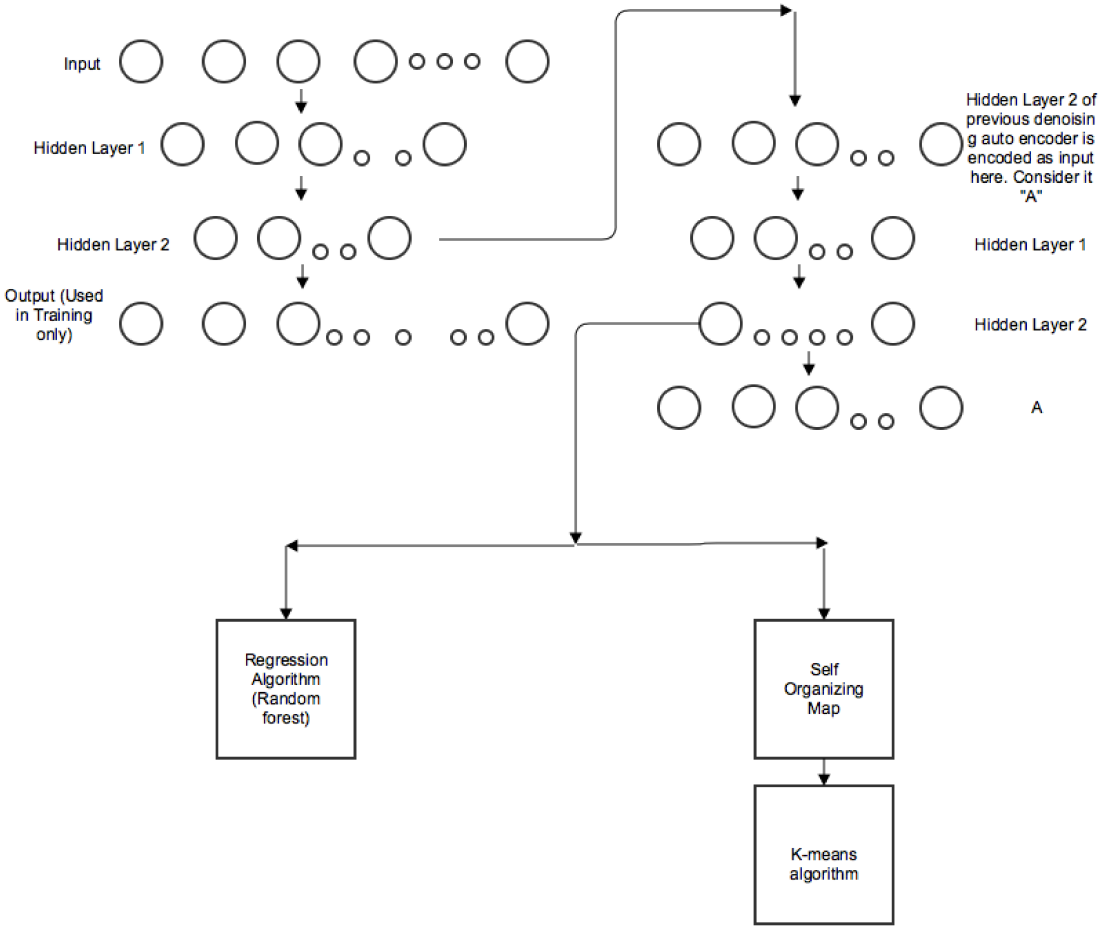


Figure 4.3: Proposed method: SDA based regression and clustering framework

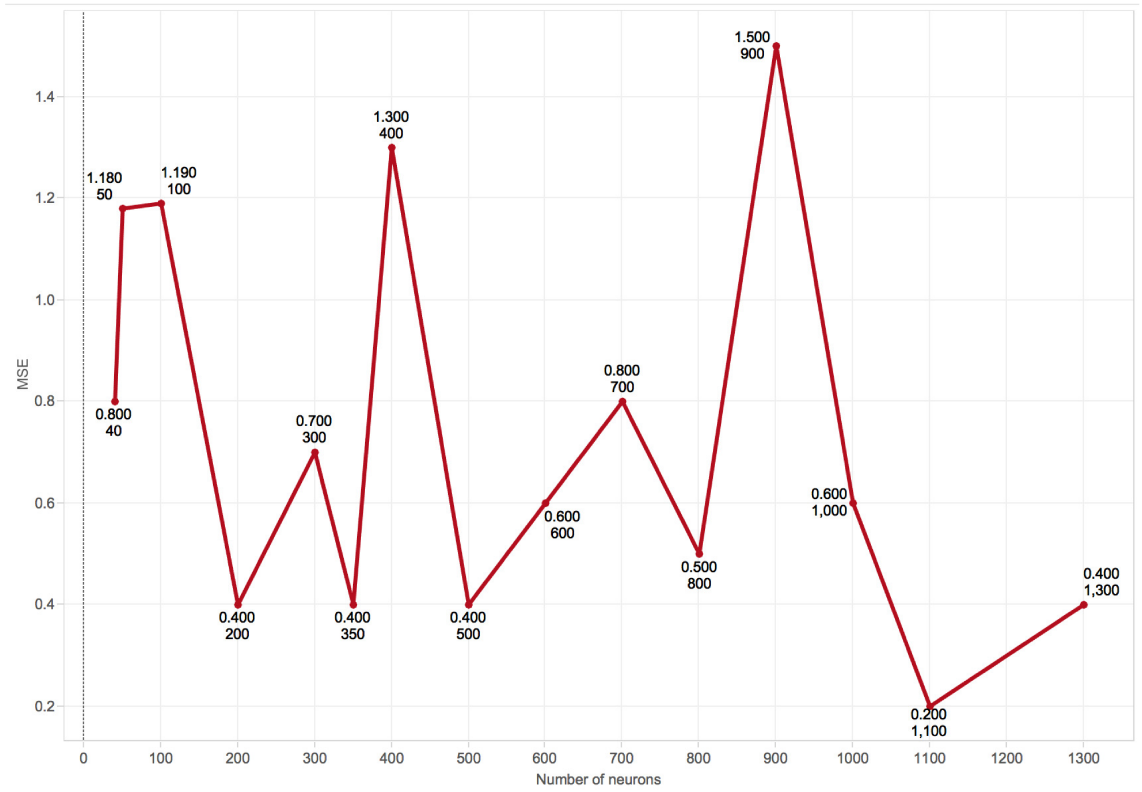


Figure 4.4: Obtaining Stacked Denoising Autoencoder architecture: Number of Neurons vs Mean square error - Hidden Layer 1.

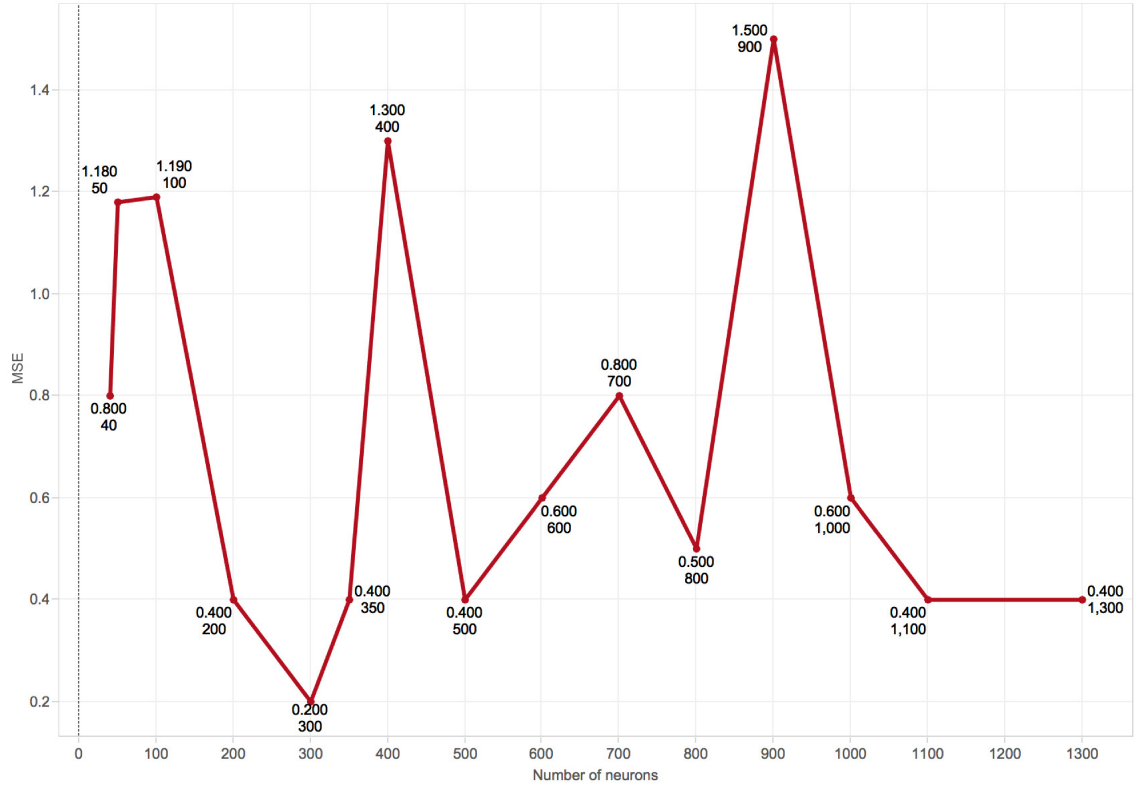


Figure 4.5: Obtaining Stacked Denoising Autoencoder architecture: Number of Neurons vs Mean square error - Hidden Layer 2.

The number of neurons in first hidden layer and second hidden layer is predicted based on mean square error and the corresponding graph is shown in figure 4.4 and figure 4.5. The number of neurons needed to process the input data in first hidden layer with least mean square error is shown figure 4.4. Likewise, the number of neurons needed in the second hidden layer to process the output of hidden layer 1 with least mean square error is shown in figure 4.5. The is no specific pattern in figure 4.4 and figure 4.5 because autoencoder uses greedy search to empirically fix the number of neurons [66].

4.2.1 Regression results (generated using dimensionality reduced dataset)

The change in reconstruction error with respect to learning rate when we tried to reduce the attributes from 2594 to 300 is shown in figure 4.6(a). We could notice that reconstruction error is as low as 9.9 when the learning rate is 1. The change in reconstruction error with respect to learning rate when we tried to reduce the attributes from 2594 to 150 is shown in figure 4.6(b). We could notice that reconstruction error is as low as 23.8 when the learning rate is 1.

Average regression accuracy of property price prediction for the year 2011, 2012, 2013, 2014 and 2015 is shown in figure 4.7. This figure also explains that regression is done separately for every municipality.

Real estate market is highly dynamic and the price changes drastically every year. To make sure our machine learning model will predict the property prices without any dip in accuracy, I compared the difference between predicted price and sales price. There are over 30538 properties in Nova Scotia with valid sales record. Percentage of properties in each bin based on the difference between predicted price and sales price is shown in figure 4.8 (properties which are sold in that year are considered.).

Comparing the prediction accuracy in figure 3.2, 3.3 and 4.9, we could conclude that regression result generated after SDA is better compared to regression results generated without SDA or with raw input.

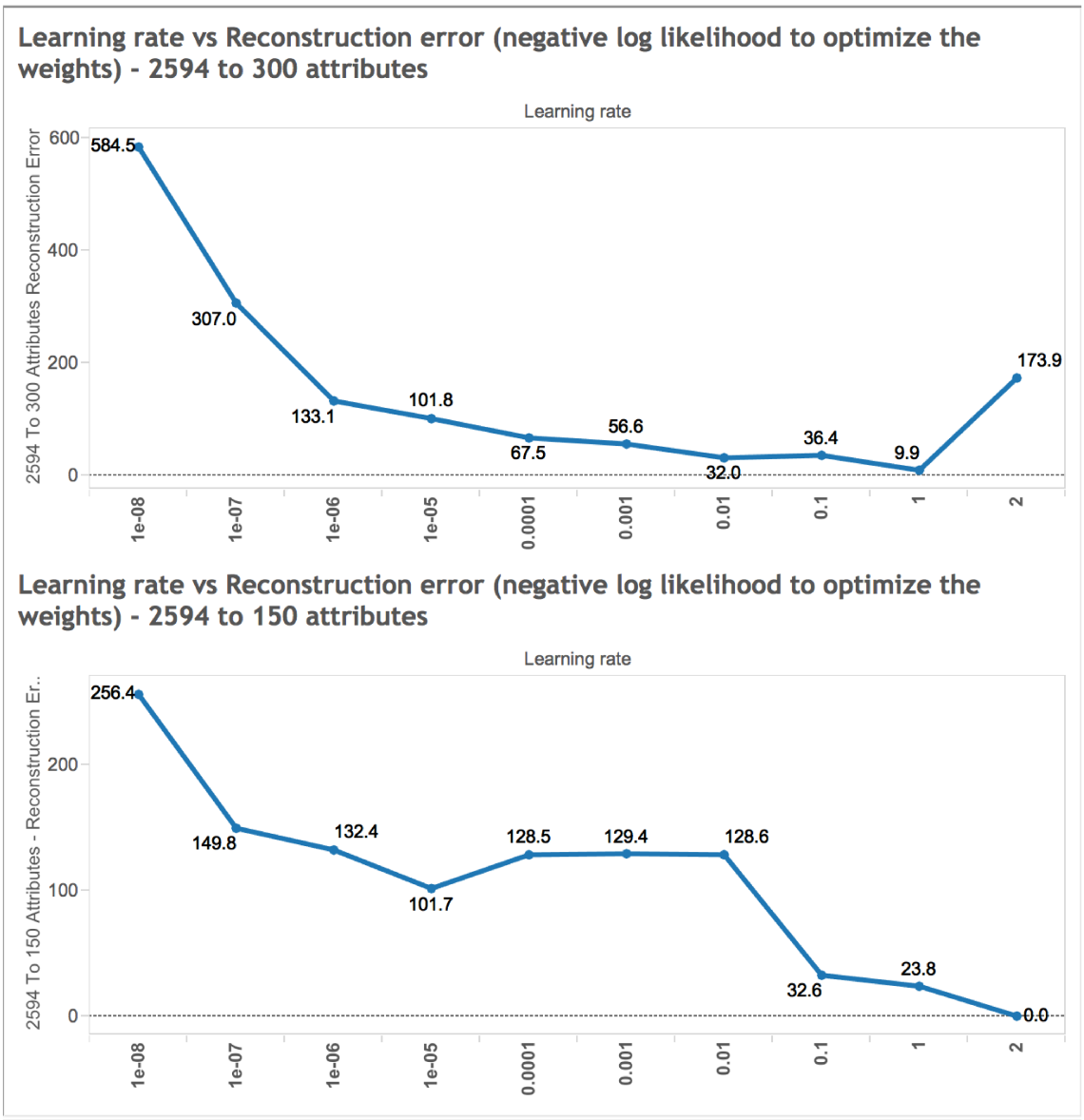


Figure 4.6: Learning rate vs Reconstruction error

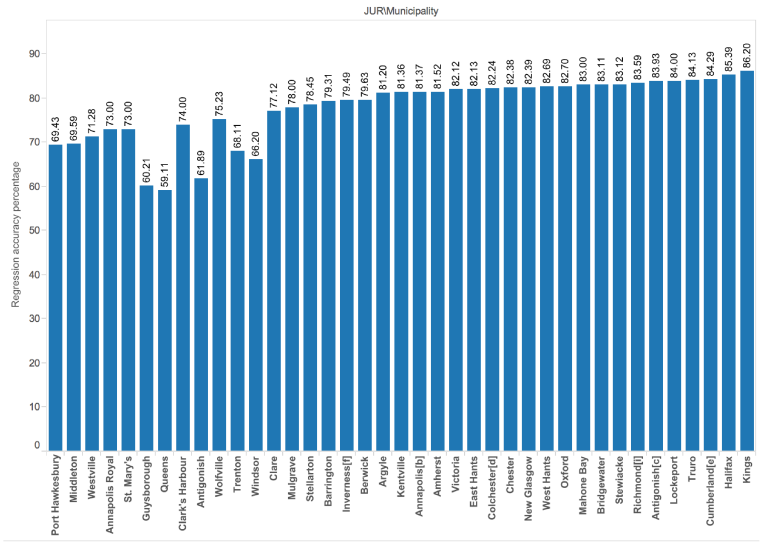


Figure 4.7: Regression accuracy specific to Municipalities in Nova Scotia

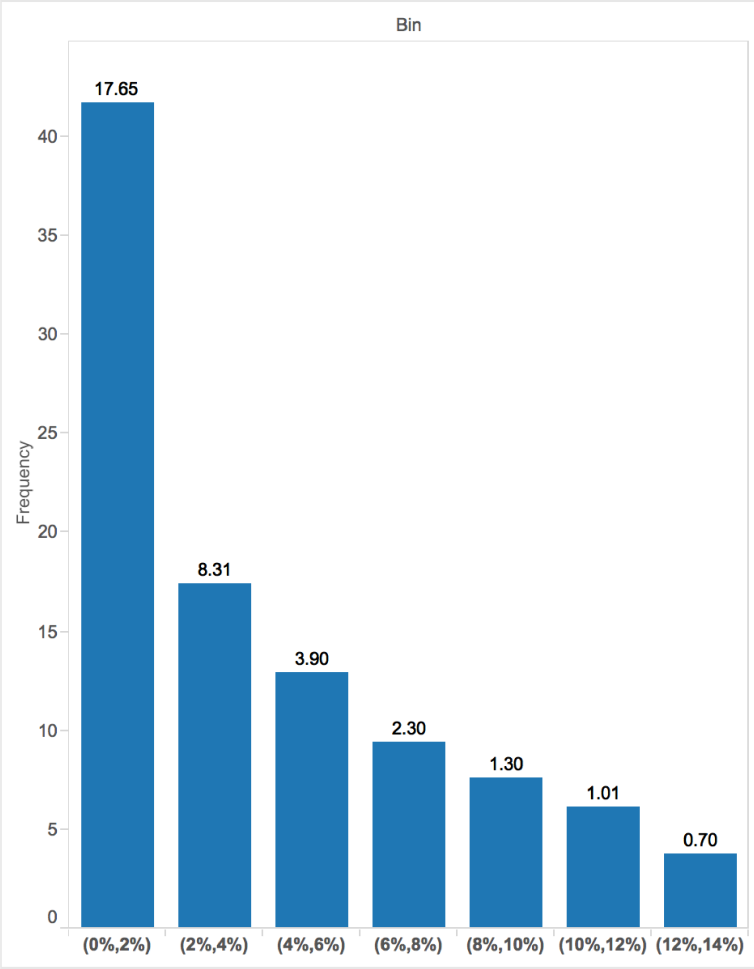


Figure 4.8: Percentage change between predicted regression price and actual assessment price

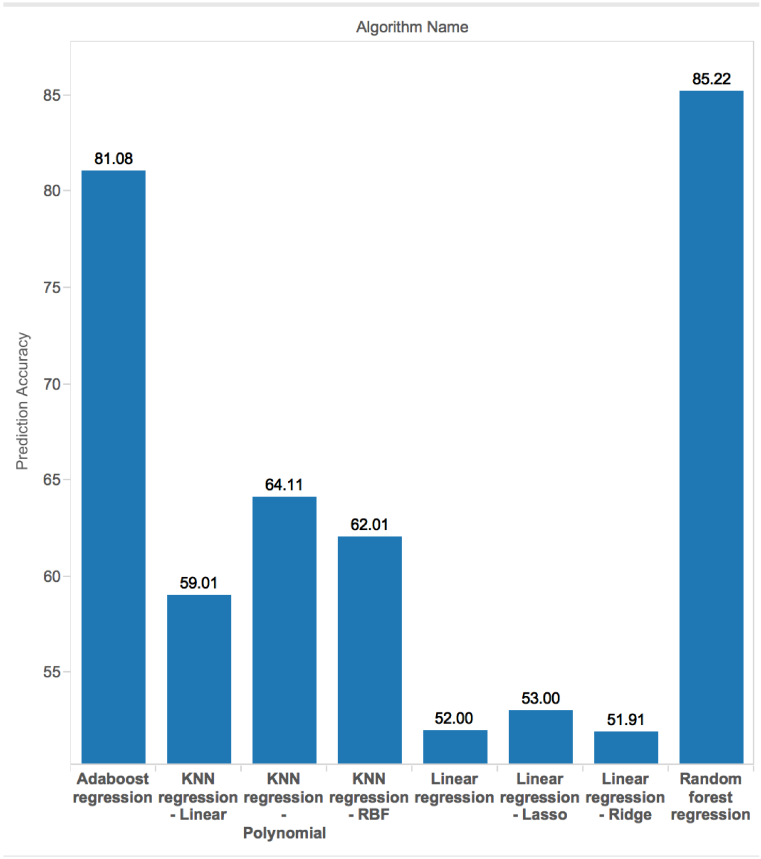


Figure 4.9: Prediction accuracy for regression algorithms using SDA

4.2.2 Clustering results (generated using dimensionality reduced dataset)

In this section, I have consolidated all the clustering results generated by the proposed framework. Dimensionality reduced dataset is passed as input to SOM Clustering algorithm. Every neuron in SOM represents a cluster of properties. The resulting weights generated by neurons in SOM were passed to K-means for further clustering. Correlation between number of clusters and silhouette value generated by experimenting with SOM algorithm is shown in figure 4.10. This figure shows that 55 clusters could be generated with maximum silhouette value of 0.83.

Correlation between number of clusters and Silhouette value generated by experimenting with K-means algorithm is shown in figure 4.11. This figure shows that 30 clusters could be generated with maximum silhouette value of 0.84. By this we conclude that the approximately 530000 Nova Scotia provincial real estate data could be divided into 30 clusters. Two level clustering approach helps us to reduce the number of clusters that facilitates quantitative analysis of SOM map and data. The main benefit of this approach is that computational load decreases drastically, allowing us to cluster large datasets within a limited time. Another benefit is we could see the granular view (see appendix A) and it enables us to inspect final individual structures closely when we visualize the clusters which in turn provides us with better understandability about the residential properties in the province.

To quantify the extent of disimilarity between clusters, we use Normalized Mutual Information (NMI). NMI value varies between 0 to 1. Two clusters are said to be distinct if the value of NMI is 0. Two clusters are said to be same if the value of NMI is 1. Dissimilarity is the inverse of NMI.

The degree of uniqueness between the clusters is defined by the formula,

Exclusive rate $(\mathrm{ER})_i = \sum_{j=1}^{n} D_{ij}$, where "n" is the number of clusters and D_{ij}

denotes the dissimilarity between clusters.

The above equation help us to answer - "How different is this cluster compared to all other clusters?" The average exclusive rate of SDA-based-Two-step-clustering is 46.81. This value is approximately 2 times the value generated by two-step clustering without using Stacked Denoising autoencoder (refer section 3.6.2). This shows SDA based two-step clustering is performing better than two-step clustering.

The view of clusters in different Forward sortation area (FSA) in Nova Scotia province is shown in appendix (Figure A.1 to A.17). All these figures confirm that clustering algorithm produces acceptable results because we could visually confirm that it clusters residential properties in a neighbourhood (mostly similar kind of houses) into same cluster. For example, consider figure 4.12(a) that shows the clustering of properties in B3L. B3L has 5 neighborhoods which are clustered into 7 clusters. Houses in the same street/neighbourhood has almost same kind of attributes which would place them in same cluster. This pattern is visually visible in almost all the FSA.

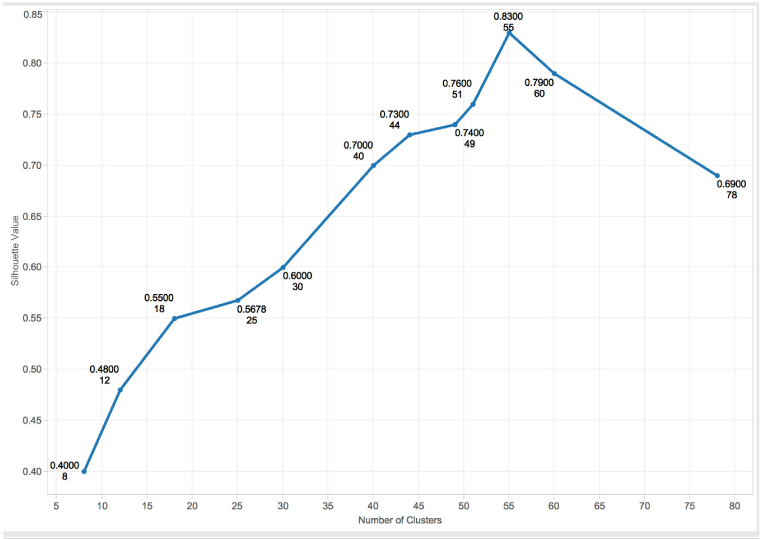


Figure 4.10: SOM based clustering - Number of Clusters vs Silhouette value

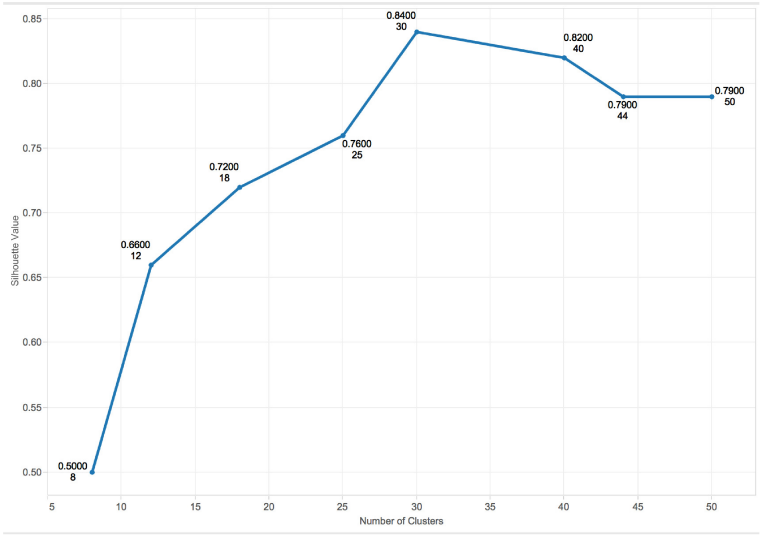


Figure 4.11: KNN based clustering - Number of Clusters vs Silhouette value

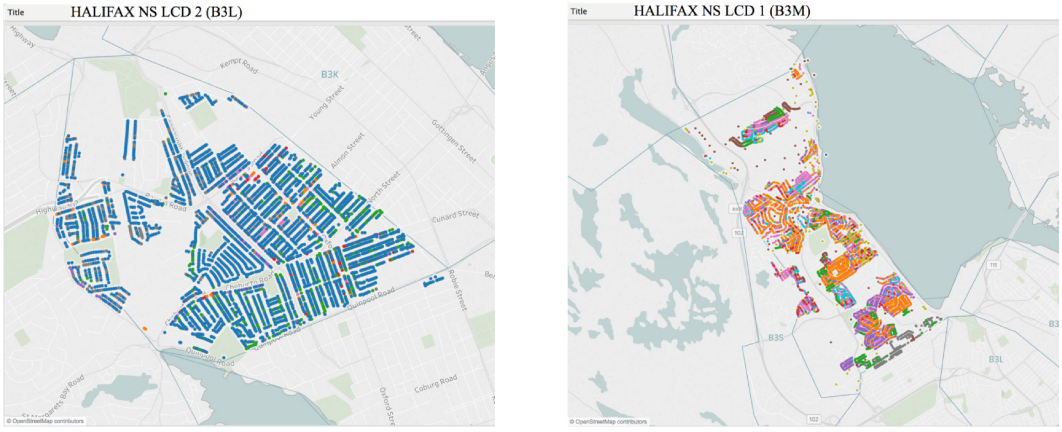
It is necessary to test the statistical significance of our results. We employ Friedman's test and Two-tailed T-test to understand the improvement of Stacked denoising autoencoder based two stage clustering approach. To quantify the extent of disimilarity between clusters, we use Normalized Mutual Information (NMI). NMI value varies between 0 to 1. Two clusters are said to be distinct if the value of NMI is 0. Two clusters are said to be same if the value of NMI is 1. Dissimilarity is the inverse of NMI.

$$D_{ij} = \left\{ \begin{array}{c} \text{0,if i=j, same cluster} \\ < 1, \text{ if i} \simeq \text{j, clusters with some similarity} \\ \text{1, if i} \neq \text{j, distinct clusters} \end{array} \right\}$$

The degree of uniqueness is defined by the formula,

Exclusive rate $(\text{ER})_i = \sum_{j=1}^{n} D_{ij}$, where "n" is the number of clusters and D_{ij} denotes the dissimilarity between clusters.

The above equation help us to answer - "How different is this cluster compared to all other cluster?"



(a) B3L (b) B3M

Figure 4.12: B3L (left) and B3M (right) Clustering result view

Table 4.2: Results of dissimilarity from the proposed clustering framework

Results of dissimilarity from the proposed clustering framework					
Dataset	Average ER Two-step-clustering	Average ER SDA-based-Two-step-clustering	T-test	Friedman's test	Overall benefit(Subtract Average ER of SDA based Two step Clustering and Average ER of Two step clustering)
Nova Scotia real estate dataset(2001-2015)	23.30	46.81	$1.32e^{-25}$	$1.62e^{-6}$	23.5

The proposed framework is better than SOM and it is statistically proved as shown in table 4.2. If the p-values (column 4 in table 4.2) are < 0.05, it is concluded that the improvement in Exclusive rate is statistically significant.

Chapter 5

Conclusion and Future Work

We have designed a framework which has low error rate in unseen data in both regression and clustering. Comparing prediction accuracy of random forest regressor using raw input (figure 3.2), prediction accuracy of random forest regressor without SDA (figure 3.3) and prediction accuracy of random forest regressor with SDA (figure 4.9), we conclude that the prediction accuracy of random forest regression model improved by approximately 5% after we reduce the dimensionality using SDA. The framework developed in this thesis will help us to predict the property value considering the current market scenarios. In Regression, Random forest model showed high prediction accuracy than all the other linear models, support vectors and Adaboost models. In Clustering, two-level clustering approach helped us to generate high Silhouette value for optimal number of clusters.

Stacked denoising autoencoder helped us to reduce the dimensionality of the data. This helped us to reduce the running time and improve the accuracy of both regression and clustering algorithms.

To achieve optimal performance, property sales data is one of the most important attribute we considered as mentioned in Chapter 3. Not all properties would have

sales data in a neighbourhood or Forward sortation area in a particular year. So, we formally defined a sales value based on the other property attributes and existing sales record to all the residential properties in a particular geographic vicinity and tested the regression models. This gave us optimal regression and clustering performance. These results can be achieved by considering forward sortation area (FSA) [1] as submarket (Geographic vicinity).

Future research of real estate data analysis has lot of areas to explore.

1. Many different dimensionality reduction algorithms can be applied and tested for performance improvement.

2. Submarkets has highest error when most of the properties in that geographical vicinity has high price. Any important attribute missing here?

3. Can we consider Latitude and Longitude as features instead of defining submarkets? If so, do they improve the performance measure of Machine Learning algorithms?

4. Can we consider training a regression algorithm separately for every cluster generated by SOM? If we replace neighbourhood by cluster, how do they impact the accuracy of regression techniques.

Few more general ideas that could improve real estate property price prediction is development of open source product to fit in and analyze any real estate dataset.

[1] FSA is a collection of zip codes in a specific geographic region. First three letters of a zip code defines the name of the FSA [19]. For example B3H would be the name of the FSA which comprises the geographical regions of zip codes B3H 1R2, B3H1R3, B3H1R4 etc

Bibliography

[1] *Zillow: Machine learning and data disrupt real estate* https://www.zdnet.com/article/zillow-machine-learning-and-data-in-real-estate/.

[2] *Canadian real estate outlook 2017* http://www.moneysense.ca/spend/real-estate/canadian-real-estate-market-outlook-2017/.

[3] Sumit Chopra, Trivikraman Thampy, John Leahy, Andrew Caplin, Yann LeCun. *Discovering the Hidden Structure of House Prices with a Non-Parametric Latent Manifold Model*. 13th International Conference on Knowledge Discovery and Data Mining (KDD), San Jose CA, August 2007.

[4] Court, A. T. *Hedonic Price Indexes With Automotive Examples*. The Dynamics of Automobile Demand , 1939, New York, General Motors.

[5] Rosen, S., *Hedonic Prices and Implicit Markets: Product Differentiation in Pure Competition*. Journal of Political Economy, 1974, 82, 34-55.

[6] Lancaster, Kevin, *A New Approach to Consumer Theory*. Journal of Political Economy, 1966, 74, 132-157.

[7] Can, A., *Measurement of Neighborhood Dynamics in Urban House Prices*. Economic Geography, 1990, 66, 254-72.

[8] Bajic, V., *Housing Market Segmentation and Demand for Housing Attributes.* Journal of the American Real Estate and Urban Economics Association, 1985, 13, 58-75.

[9] Goodman, Allen C., Thibodeau, Thomas G., *Housing Market Segmentation and Hedonic Prediction Accuracy.* Journal of Housing Economics, 2003, 12, 181-201.

[10] Sirmans, Stacy, Macpherson, David A., Zietz, Emily N., *The Composition of Hedonic Pricing Models.* Journal of Real Estate Literature, 2005.

[11] Bradford Case, John Clapp, Robin Dubin, and Mauricio Rodriguez, *Modeling spatial and temporal house price patterns: A comparison of four models.* Journal of Real Estate Finance and Economics 29 (2004), no. 2, 167–191.

[12] Geoff Bohling *http://www.people.ku.edu/ gbohling/cpe940/Kriging.pdf*

[13] Sumit Chopra, Trivikaraman Thampy, John Leahy, Andrew Caplin, Yann Le-Cun *Factor Graphs for Relational Regression* Journal of Real Estate Finance and Economics 29 (2004), no. 2, 167–191.

[14] Sumit Prakash Chopra, *Factor graphs for relational regression.* Ph.D. thesis, New York University, 2009.

[15] Jennifer A. Hoeting, Richard A. Davis, Andrew Merton *Model Selection for Geostatistical Models.* Department of Statistics, Colorado State University, Fort Collins, Colorado 80523-1877 USA 2P.O. Box 999, K5–12, Richland, Washington 99352 USA, Ecological Applications, 16(1), 2006, pp. 87–98

[16] Timothy J. Fik, David C. Ling, and Gordon F Mulligan, *Modeling spatial variation in housing prices: A variable interaction approach.* Real Estate Economics 31 (2003), 623 – 646.

[17] Steven C. Bourassa, Martin Hoesli, and Vincent S. Peng, *Do housing submarkets really matter?*. Tech. report, Universite De Geneve, Geneva, Switzerland, 2003.

[18] Steven C. Bourassa, Eva Cantoni, and Martin Hoesli, *Predicting house prices with spatial dependence: A comparison of alternative approaches*. Journal of Real Estate Research 32 (2010), no. 2, 139–159.

[19] Forward Sortation Area definition *https://www.ic.gc.ca/eic/site/bsf-osb.nsf/eng/br03396.html*

[20] Greg Ridgeway, David Madigan, and Thomas Richardson *Boosting Methodology for Regression Problems*. Box 354322, Department of Statistics University of Washington, Seattle, WA 98195, 1999.

[21] *Regularization with Ridge penalties, the Lasso, and the Elastic Net for Regression with Optimal Scaling Transformations. https://openaccess.leidenuniv.nl/bitstream/handle/1887/12096/04.pdf*

[22] *http://chem-eng.utoronto.ca/ datamining/Presentations/KNN.pdf*

[23] *https://www.stat.cmu.edu/ cshalizi/350/lectures/22/lecture-22.pdf*

[24] *https://alex.smola.org/papers/2004/SmoSch04.pdf*

[25] Nissan Pow, Emil Janulewicz, Liu (Dave) Liu. *Applied Machine Learning Project 4 Prediction of real estate property prices in Montreal(2011).*

[26] R. J. Shiller. *"Understanding recent trends in house prices and home ownership."* National Bureau of Eco- nomic Research, Working Paper 13553, Oct. 2007. DOI: 10.3386/w13553. [Online]. Available: http://www.nber.org/papers/w13553.

[27] *http://labs.seas.wustl.edu/bme/raman/Lectures/Lecture10CompetitiveLearning.pdf*

[28] Van Hulle M.M. (2012) *Self-organizing Maps. In: Rozenberg G., Bäck T., Kok J.N. (eds) Handbook of Natural Computing. Springer, Berlin, Heidelberg*

[29] Open source data portal *https://data.novascotia.ca/*

[30] Yan Lecunn, Leon Buttou, Genevieve B.Orr, Klaus-Robert Muller *Efficient back-propagation,* http://yann.lecun.com/exdb/publis/pdf/lecun-98b.pdf

[31] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol *Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion*, http://www.jmlr.org/papers/volume11/vincent10a/vincent10a.pdf, Journal of Machine Learning Research 11 (2010) 3371-3408, 2010.

[32] Kanus *"Predicting a house's selling price through inflating its previous selling price."* JORS 60(3): 339-347 (2009)

[33] Hakan Kusan, Osman Aytekin, Ilker Özdemir, *"The use of fuzzy logic in predicting house selling price."* Expert Syst. Appl. 37(3): 1808-1813 (2010).

[34] Aaron Ng, *"Machine Learning for a London Housing Price Prediction Mobile Application."* Expert Syst. Appl. ,June, 2015

[35] Alex Seutin and Ian Jones *"Using Machine Learning to Predict Housing Prices Given Multivariate Input."* Expert Syst. Appl. ,June, 2015

[36] Byeonghwa Parka, Jae KwonBaeb *"Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data."* Volume 42, Issue 6, 15 April 2015, Pages 2928-2934.

[37] GuJirong, ZhuMingcang, Jiang Liuguangyan, *"Housing price forecasting based on genetic algorithm and support vector machine."* Volume 38, Issue 4, April 2011, Pages 3383-3386.

[38] Xibin Wang, Junhao Wen, Yihao Zhang, Yubiao Wang, *"Real estate price forecasting based on SVM optimized by PSO."* Volume 125, Issue 3, February 2014, Pages 1439-1443.

[39] Byeonghwa Park, Jae Kwon Bae, *"Using machine learning algorithms for housing price prediction."* Expert Systems with Applications: An International Journal archive, Volume 42 Issue 6, April 2015.

[40] Vilius Kontrimas Antanas Verikas, *"The mass appraisal of the real estate by computational intelligence."* Expert Systems with Applications: An International Journal archive, Volume 39, Issue 9, July 2012, Pages 8369-8379.

[41] Schneider A, Hommel G, Blettner M. *Linear Regression Analysis: Part 14 of a Series on Evaluation of Scientific Publications. Deutsches Ärzteblatt International. 2010;107(44):776-782. doi:10.3238/arztebl.2010.0776.*

[42] Robert Tibshirani,Stanford University, USA *Regression shrinkage and selection via the lasso: a retrospective,* https://pdfs.semanticscholar.org/6b5e/99c128b9cd7b7fbc817a2843a47ce8a1c35d.pdf

[43] *https://ncsswpengine.netdnassl.com/wpcontent/themes/ncss/pdfNCSS/RidgeRegression.pdf*

[44] Alex J. Smola and Bernhard Scholkopf *A tutorial on support vector regression* 2004 Kluwer Academic Publishers.

[45] Robert E. Schapire, *Explaining AdaBoost* http://rob.schapire.net/papers/explainingadaboost.pdf

[46] Gilles Louppe *UNDERSTANDING RANDOM FORESTS,https://arxiv.org/pdf/1407.7502.pdf*

[47] Kiri Wagstaff Claire Cardie Seth Rogers Stefan Schroedl *Constrained K-means Clustering with Background Knowledge, https://web.cse.msu.edu/ cse802/notes/ConstrainedKmeans.pdf*

[48] Teuvo Kohenen *Exploration of very large databases using Self Organizing maps*

[49] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol *Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion*

[50] Jae Joon Ahn, Hyun Woo Byun,Kyong Joo Oh,Tae Yoon Kimb1, *"Using ridge regression with genetic algorithm to enhance real estate appraisal forecasting."* Expert Systems with Applications: An International Journal archive, Volume 39, Issue 9, July 2012, Pages 8369-8379.

[51] Vasilios Plakandaras, Rangan Gupta, Periklis Gogas, Theophilos Papadimitrioua, *"Forecasting the U.S. real house price index."* Expert Systems with Applications: An International Journal archive, Volume 45, February 2015, Pages 259-267.

[52] Hengshu Zhu, Hui Xiong, Fangshuang Tang, Qi Liu, Yong Ge, Enhong Chen, Yanjie Fu *"Days on Market: Measuring Liquidity in Real Estate Markets."* KDD '16, August 13-17, 2016,2016ACM.ISBN978-1-4503-4232-2/16/08.

[53] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol *Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion* Journal of Machine Learning Research 11 (2010) 3371-3408

[54] Chen Xing,Li Ma, and Xiaoquan Yang *Stacked Denoise Autoencoder Based Feature Extraction and Classification for Hyperspectral Images*

[55] Magdalena Graczyk, Tadeusz Lasota, Bogdan Trawinski, Krysztof Trawinski *"Comparison of Bagging, Boosting and Stacking Ensembles Applied to Real Estate Appraisal."* Expert Systems with Applications: An International Journal archive, Volume 45, February 2015, Pages 259-267.

[56] Fernando Bac, Victor Lobo1, and Marco Painho *"Self-organizing Maps as Substitutes for K-Means Clustering."* V .S. Sunderam et al. (Eds.): ICCS 2005, LNCS 3516, pp. 476 – 483, 2005.

[57] Osama Abu Abbas *"Comparisons between data clustering algorithms."* The international arab journal of Information technology, Vol: 5, No.3, July 2008.

[58] Juha Vesnto and Esa Alhoniemi *"Clustering of the Self-Organizing map."* IEEE transaction on neural network, Vol:11, No:3, May, 2000.

[59] T.Kohonen *"Self-Organizing maps"* Germany: Springer,1995, vol:30.

[60] D.Pyle *"Data preparation for Data Mining."* San Fancisco, CA: Morgan Kaufmann, 1999.

[61] J. Vesanto *"SOM-based data visualization methods."* Intell. Data analysis, vol: 3, no. 2, pp 111-126, 1999.

[62] G. Karypis, E-H Han, and V. Kumar *"Clustering properties of hierarchical self-organizing maps."* IEEE computing, vol:32, pp, 68-94,Aug 1999.

[63] Aristidis Likas, Nikos Vlassis *"The global k-means clustering algorithm."* Pattern Recognition, Elsevier, Jakob J. Verbeek, Volume 36, Issue 2, February 2003, Pages 451-461.

[64] Kyoung jae Kim, Hyunchu Ahn *"A recommender system using GA K-means clustering in an online shopping market."* Elsevier, Expert Systems with Applications, Volume 34, Issue 2, February 2008, Pages 1200-1209.

[65] K.-L.Du *"Clustering: A neural network approach."* Neural Networks, Elsevier, Volume 23, Issue 1, January 2010, Pages 89-107.

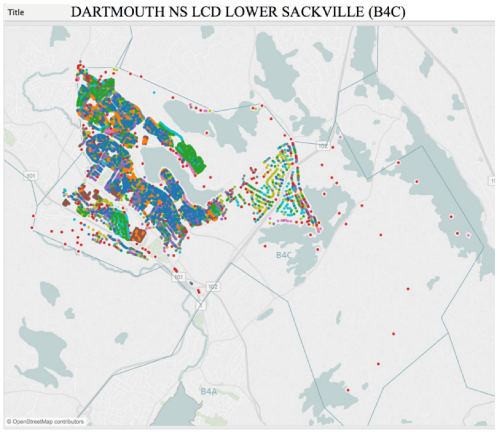
[66] D. Rajashekar *"One class learning problem with autoencoder"*

[67] Jiahui Mo, Melody Y.Kiang, Peng Zou, Yijun Lid *"A two-stage clustering approach for multi-region segmentation."* Elsevier, Expert Systems with Applications,Volume 37, Issue 10, October 2010, Pages 7120-7131.

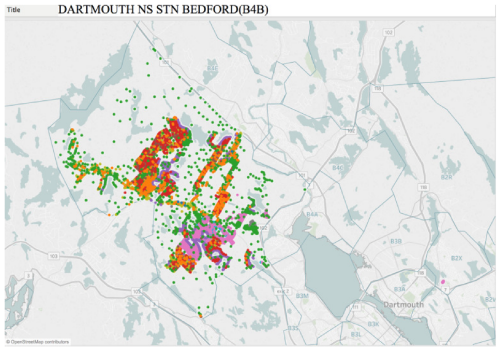
[68] Leo Breiman *Random Forests* Machine Learning, 45, 5–32, 2001 Kluwer Academic Publishers. Manufactured in The Netherlands,2001.

Appendix: Clustering visualization

Following are the visualizations (one per FSA) from evaluation of the proposed framework on the PVSC real estate dataset.

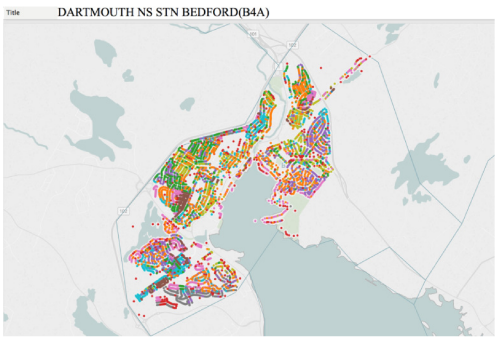


(a) B4C

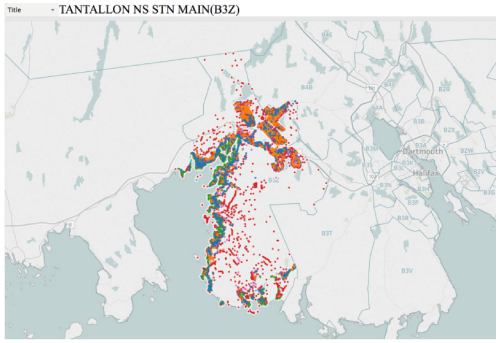


(b) B4B

Figure 1: B4C (left) and B4B (right) Clustering result view

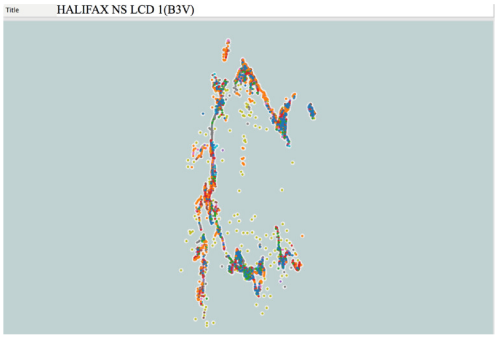


(a) B4A

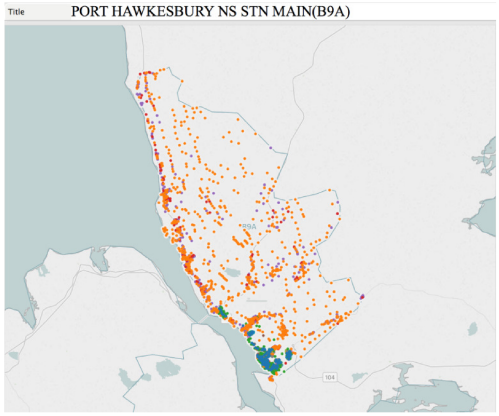


(b) B3Z

Figure 2: B4A (left) and B3Z (right) Clustering result view

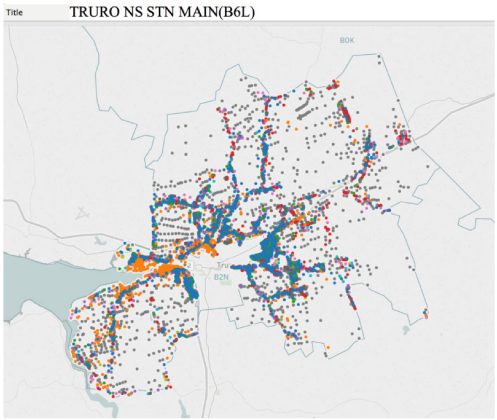


(a) B3V

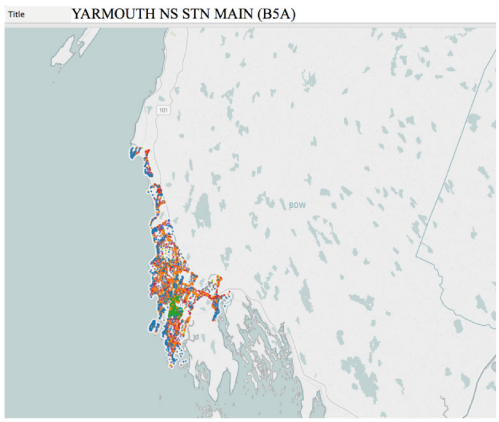


(b) B9A

Figure 3: B3V (left) and B9A (right) Clustering result view

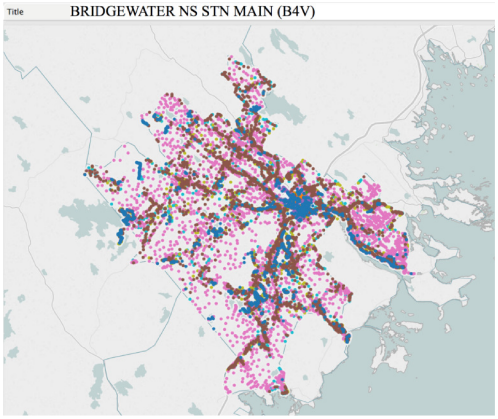


(a) B6L

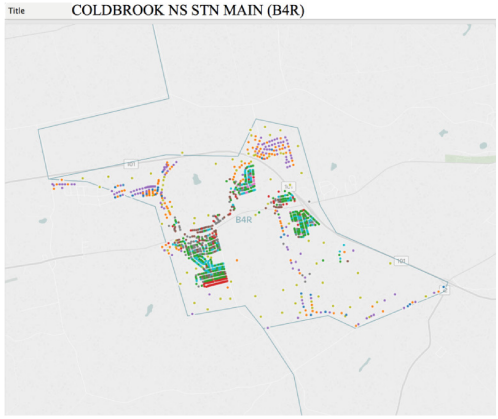


(b) B5A

Figure 4: B6L (left) and B5A (right) Clustering result view

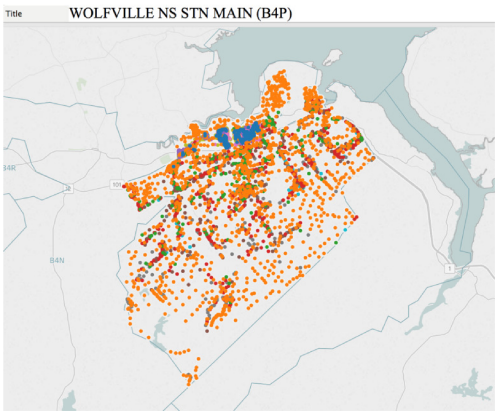


(a) B4V

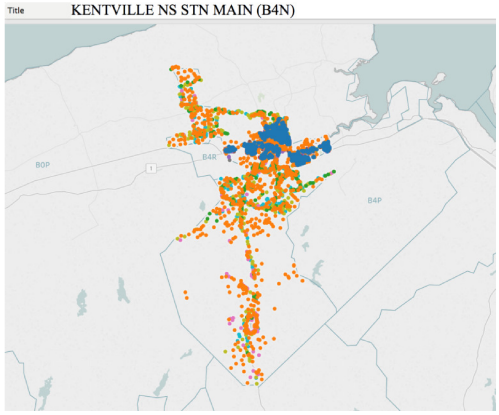


(b) B4R

Figure 5: B4V (left) and B4R (right) Clustering result view

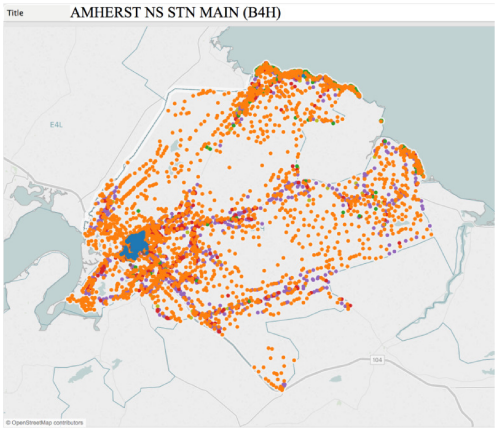


(a) B4P

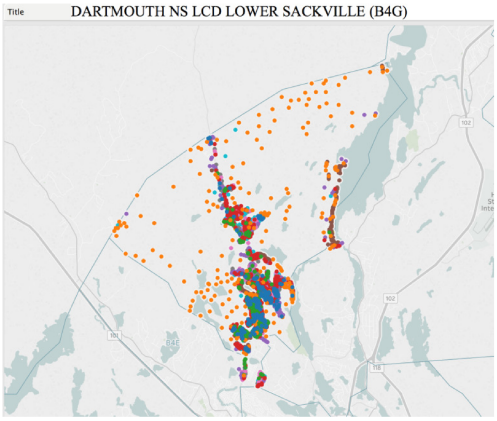


(b) B4N

Figure 6: B4P (left) and B4N (right) Clustering result view

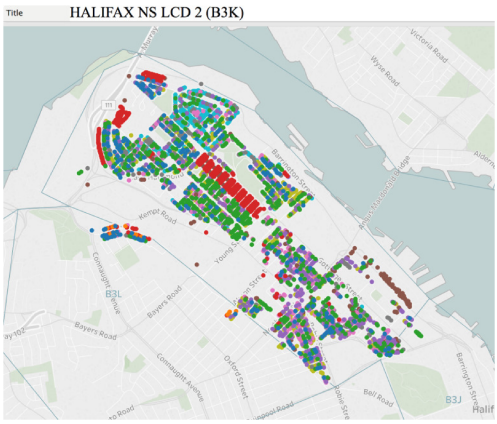


(a) B4H

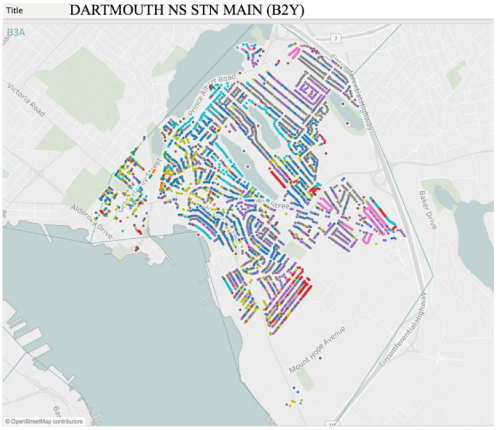


(b) B4G

Figure 7: B4H (left) and B4G (right) Clustering result view

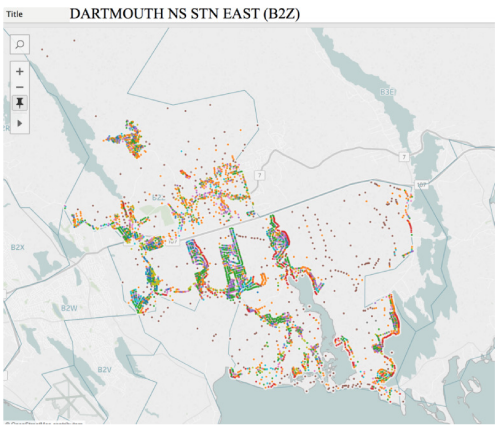


(a) B3K

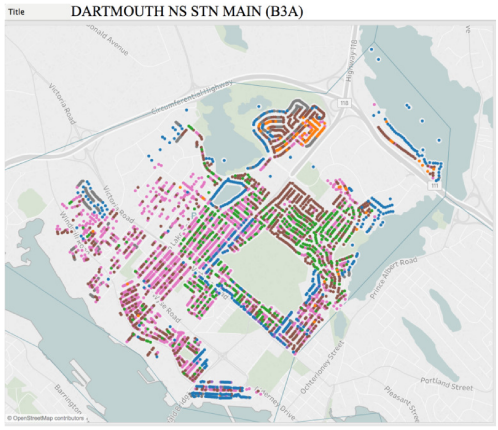


(b) B2Y

Figure 8: B3K (left) and B2Y (right) Clustering result view

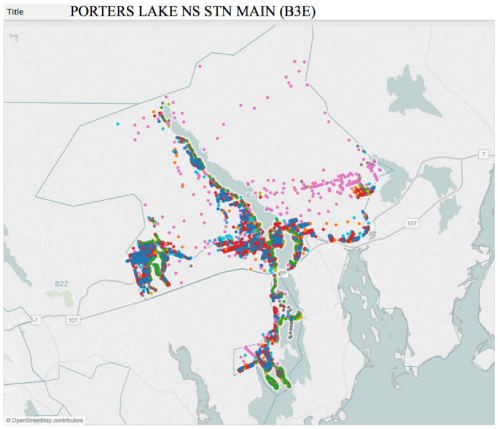


(a) B2Z

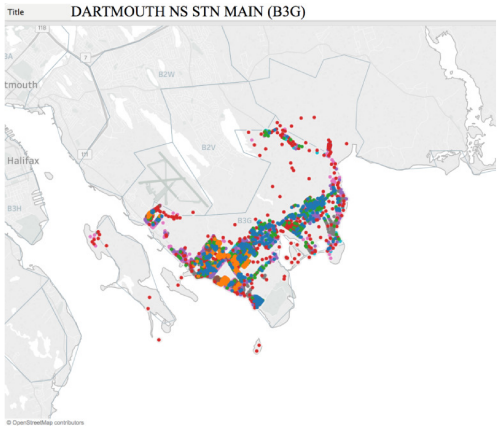


(b) B3A

Figure 9: B2Z (left) and B3A (right) Clustering result view

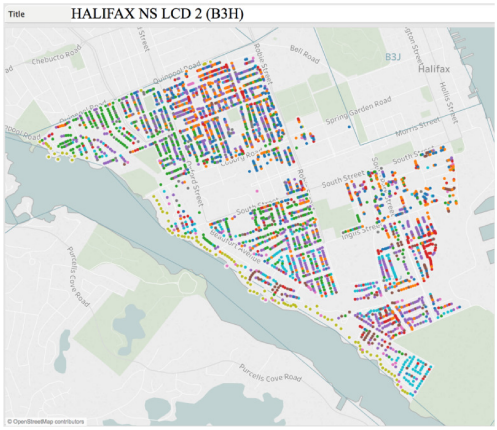


(a) B3E

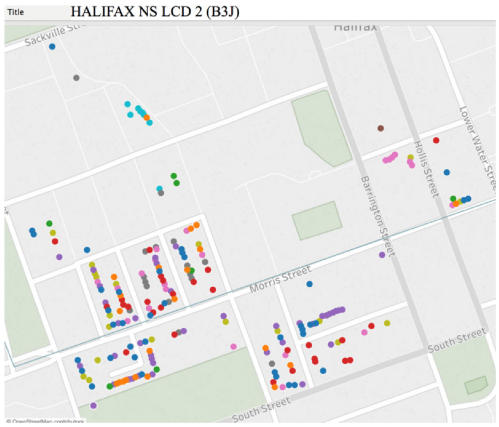


(b) B3G

Figure 10: B3E (left) and B3G (right) Clustering result view

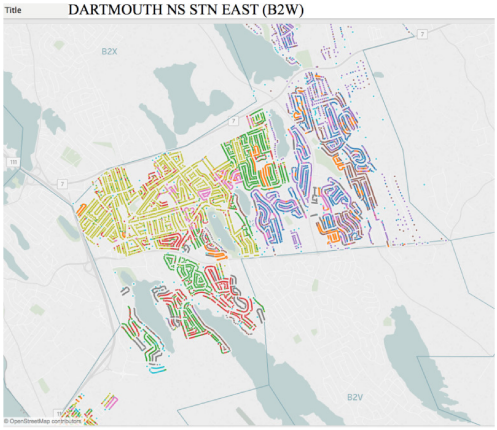


(a) B3H

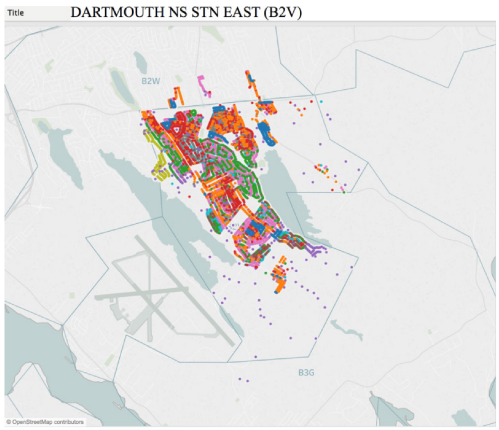


(b) B3J

Figure 11: B3H (left) and B3J (right) Clustering result view

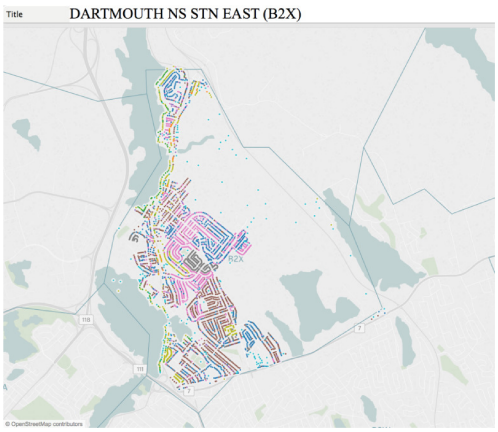


(a) B2W

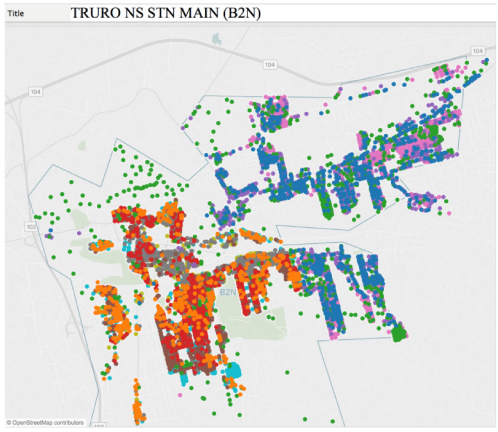


(b) B2V

Figure 12: B2W (left) and B2V (right) Clustering result view

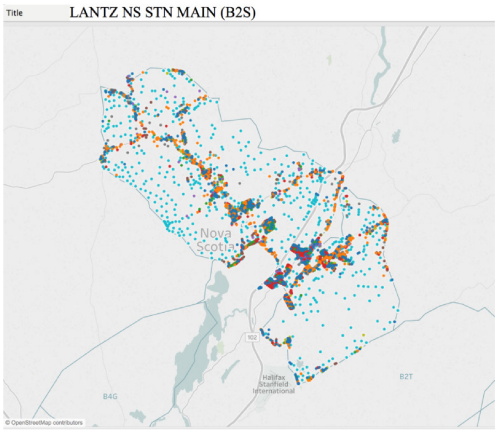


(a) B2X

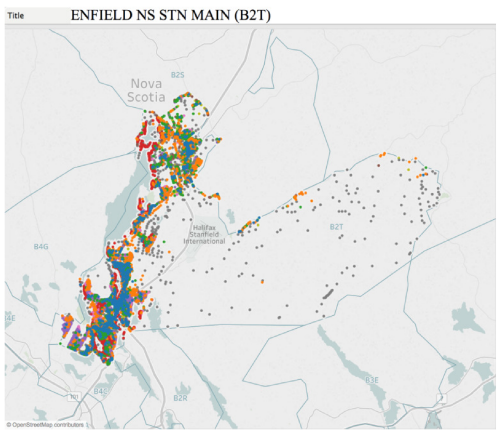


(b) B2N

Figure 13: B2X (left) and B2N (right) Clustering result view

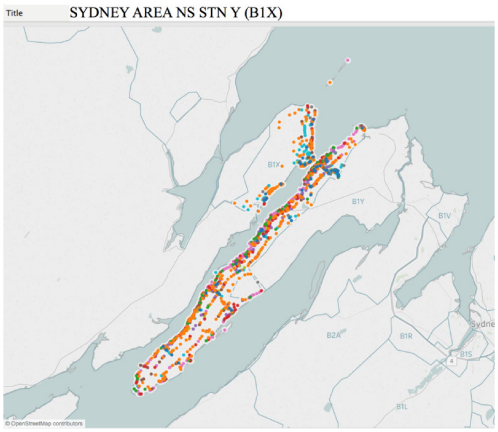


(a) B2S

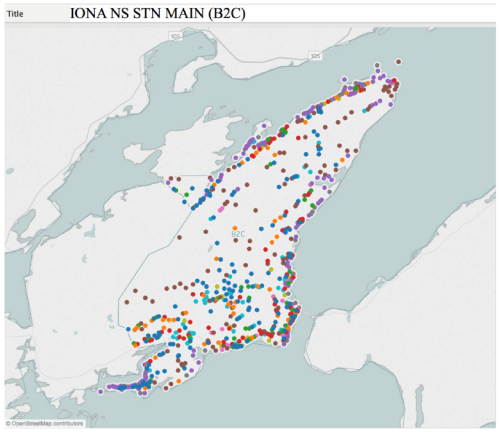


(b) B2T

Figure 14: B2S (left) and B2T (right) Clustering result view

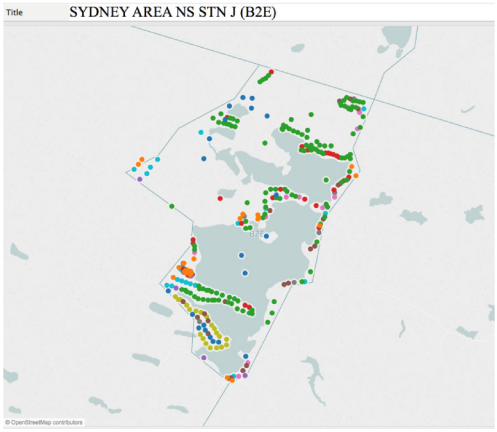


(a) B1X

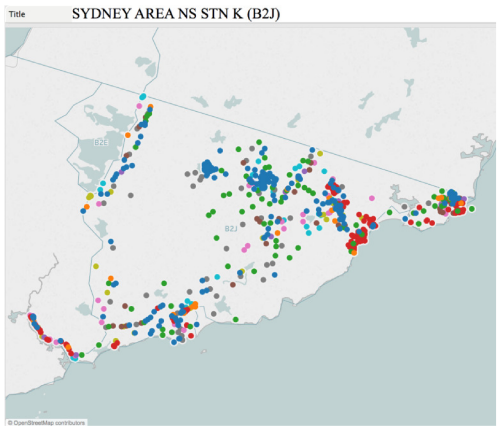


(b) B2C

Figure 15: B1X (left) and B2C (right) Clustering result view



(a) B2E



(b) B2J

Figure 16: B2E (left) and B2J (right) Clustering result view

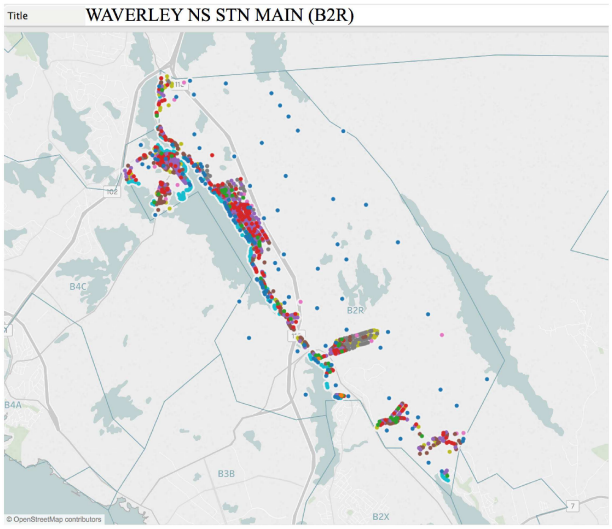


Figure 17: B2R Property clustering