Proposing An Ensemble-based Model Using Data Clustering and Machine Learning Algorithms for Effective Predictions

by

Fateme Azimlu Shanajani

A Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of

Master of Applied Science in Electrical and Computer Engineering

in

The Faculty of Engineering and Applied Science

University of Ontario Institute of Technology
Oshawa, Ontario, Canada

August 2019

THESIS EXAMINATION INFORMATION

Submitted by: Fateme Azimlu Shanjani

Masters of Applied Science in Electrical and Computer Engineering

Thesis title: Proposing an Ensemble-based Model Using Data Clustering and Machine Learning Algorithms for Effective Predictions

An oral defense of this thesis took place on August 12, 2019 in front of the following examining committee:

Examining Committee:

Chair of Examining Committee	Dr. Ying Wang
Research Supervisors	Dr. Shahryar Rahnamayan and Dr. Masoud Makrehchi
Examining Committee Member	Dr. Khalid Elgazzar
External Examiner	Dr. Khalil El-Khatib

The above committee determined that the thesis is acceptable in form and content and that a satisfactory knowledge of the field covered by the thesis was demonstrated by the candidate during an oral examination. A signed copy of the Certificate of Approval is available from the School of Graduate and Postdoctoral Studies.

Abstract

One of the most important tasks in machine learning is prediction. Data scientists use various regression methods to find the most appropriate and accurate model applicable for each type of datasets. This study proposes a meta-model to improve prediction accuracy. In common methods different models are applied to the whole dataset to find the best model with the highest accuracy. This means, a global model is developed for the entire dataset. In the proposed approach, first, we cluster data using different methods and we have used algorithm-based and expert-based clustering. Algorithm-based clustering such as K-means, DBSCAN, agglomerative hierarchical clustering algorithms. For expert-based clustering, we use expert knowledge to group datasets based on the important features which are selected by experts. Then, for each clustering method and for each generated cluster, we apply different machine learning models including linear and polynomial regressions, SVR, neural network, genetic programming and other techniques and select the most accurate prediction model per cluster. In every cluster, the number of samples in each cluster is reduced compared to the number of samples in the original dataset and consequently, by decreasing the number of samples in each cluster, the model is prone to lose its accuracy. On the other hand, customizing a model for each sub-dataset increases the capability of offering more effective prediction, compared to a situation where one model is fitted to the whole dataset. That is why the proposed model can be categorized as in an ensemble-based group due to the fact that the prediction is performed based on the collaboration of various models over clusters of sub-datasets. Moreover, granularity of the proposed method is better for parallelization purposes. This means, it can be parallelized in a more efficient way. As our main case study, we used real-estate data with more than 21,000 instances and 20 features to improve house price prediction. However, this approach is applicable to other large datasets. In order to examine its capability, we applied the proposed method on two other datasets; agricultural dataset with 10 features and more than 7,000 instances and also Facebook comments volume dataset, which contains roughly 41,000 samples with 54 features. For the first dataset, the new approach reduces error value from 0.14 to 0.087 for K-means clustering and 0.086 for grouping based on human knowledge. With respect to our second case study, the water evaporation data did not obtain considerable improvement in accuracy; however, in some sub-datasets there was an improvement in accuracy.

Keywords: Data mining, Machine learning, Clustering, Regression, Prediction, Symbolic regression, Genetic programming, Neural network, , Fusion, Ensemble model, Comparative study.

Author's Declaration

I hereby declare that this thesis consists of original work of which I have authored.
This is a true copy of the thesis, including any required final revisions, as accepted
by my examiners.
I authorize the University Of Ontario Institute Of Technology to lend this thesis to
Other Institutions or individuals for the purpose of scholarly research.
I further authorize University of Ontario Institute of Technology to reproduce this thesis
by photocopying or by other means, in total or in part, at the request of other institutions
or individuals for the purpose of scholarly research. I understand that my thesis will be
made electronically available to the public.

Fateme Azimlu Shanjani

Statement of Contribution

The proposed method described in chapter 3, has been introduced in:

Azimlu, F., Rahnamayan, S., Makrehchi, M., House Price Prediction Using Customized Fusion-based Clustering and Machine Learning Techniques, *WIML (NIPS side workshop)**,** Montreal, 2019*.

Part of the work described in Chapter 4 has been presented as:

Azimlu, F., Rahnamayan, S., Makrehchi, M., Comparative Study on Symbolic Regression via Genetic Programming and Conventional Data Mining Regression Techniques *, **ICCSE**, Toronto, 2019.*

As the first author of the paper, I performed the experiments, majority design of the model, and writing of the manuscript.

Acknowledgements

This research would not have been possible without the support of my supervisors, family, and friends.

First, I would like to thank my supervisors, Dr. Rahnamayan and Dr. Makrehchi, for their support, kind advice, and continuous encouragement.

I like to express my gratitude to my mother and my aunt who have ever assisted me with their kind moral and emotional supports.

I must express my profound gratitude to my sister, who supported me in all aspects.

Finally, I would like to thank my friends in Nature Inspired Computational Intelligence (NICI) Lab, Social Computing and Collective Intelligence Lab (SCiLab), and my friends in ECSE and CS departments for motivating me and their kind assistance when I need any help.

Thanks for all your encouragement.

Contents

List of Tables

List of Figures

ACRONYMS

ANN: Artificial Neural Network

CRP: Chinese Restaurant Process Clustering

DBSCAN: Density-Based spatial clustering of Applications with Noise

DEC: Deep Embedded Clustering

GP: Genetic Programming

HCA: Hierarchical Cluster Analysis

HAC: Hierarchical Agglomerative Clustering

KNN: K Nearest Neighbors

Mem: Member

ML: Machine Learning

MLP: Multi Layer Perceptron

NMAE: Normalized Mean Absolute Error

OLS: Ordinary Least Squares

RMSE: Root Mean Square Error

RPF: Regression on Feature Projections

SSE: Sum of Squared Errors

SVM: Support Vector Machine

SVR: Support Vector Regression

Chapter 1

Introduction

1.1 Introduction to Machine Learning Clustering, Regression and Prediction Techniques

Prediction is a key task in machine learning. Scientists use different data mining techniques to discover patterns in a given dataset to predict future-related values. The first pattern recognition was completely a human-based task, based on Bayes' theorem which is offered by Thomas Bayes (1701-1761). This theorem describes conditional probability for occurrence of an event based on prior knowledge of other conditions and events which can affect that intended event. The publication of Bayes' theorem in 1763, opened the new horizons on more accurate predictions [1]. Legendre proposed the first regression approach, least squares technique, in 1805 [2] and after Gauss mentioned it in his book in 1809 [3]; it became a well-known approach and widely used technique. Interestingly, both used this method to predict the orbital parameters of small astronomical objects such as comets and asteroids, which are rotating around the sun and can hardly be observed. In 1821, Gauss improved his proposed method to Gauss-Markov theorem [4] which became an important reference for other scientific predictions. After this, many mathematical models were provided, which improved predictions; however, the milestone in scientific regression was with the invention of computers in the 1940s. Their quick development allowed humans the ability to fulfill complicated calculations. Another milestone was the expansion of internet usage after 1990, which not only generated, but exchanges a large amount of data that was not possible to be analyzed by using traditional methods. As a result, data mining techniques were born as powerful tools to assist humans in analyzing large amounts of growing data. Data mining is a collection of techniques for discovering hidden patterns in large data sets [5] and is the conjunction of machine learning techniques, databases, and statistics. Computerized modern data processing was supported by other improvements in computer science, such as clustering analysis, neural networks, evolutionary algorithms in the 1950s and after that point, decision trees in the 1960s and finally, support vector machines in the 1990s. Arthur Samuel, a computer gaming and artificial intelligence scientist, proposed the term ”machine learning” in 1959 [6]. Machine learning (ML) includes various algorithms, which are capable of learning from data and can make predictions for a target variables [7]. This consists of different tasks such as regression, clustering, and classifying. In our research, we study differnt clustering, regression and prediction techniques, which are defined in section 2.1.

All scientific fields are affected by the invention and improvement of data mining and machine learning, including physics, biology, health and medical science, social science and economy. In a report published by the European Public Real Estate Association [8], it was demonstrated that real-estate in all its aspects includes nearly 20% of economic activities. Price estimation has an important role in the economics, marketing and even in politics. Events such as war, or natural disasters, which are not predictable long term, can have an impact on a a given economy. As many parameters can cause significant changes in prices, even in normal conditions, price estimating is not an easy task. In recent years after improving the machines, computational power and the internet potential to facilitate generating and collecting massive data, machine learning techniques have become a fundamental tool for price prediction [9], [10]. We can see machine learning effects even in most unpredictable tasks such as auction price estimation [11]. The most interesting aspect of machine learning is the potential generalization of created methods. When a new approach is examined in a specific case study dataset, it is sometimes capable of being generalized to other datasets with completely different features and structure.

1.2 Motivation and Hypothesis

In regression and prediction task, we try to create a mathematical equation that fits in both training and test data. When trying to fit a model in training data with highest accuracy, we should be careful of over-fitting, which can result in decreasing accuracy in test data and weak prediction. When we review the data mining and machine learning research works, in most cases, the predictive model is applied to the whole dataset. It is useful for creating a model because we have more instances for training and it also prevents the model from over-fitting. On the other hand, when there is a high diversity in the values of differnt features, it is hard to fit a model into a dataset and create a model with low error. For example, in some medical predictions there is a high diversity in the age of the patients or in one of the most famous machine learning problems, house price predictions, there is a high diversity in house prices and it is not easy to fit a model into the whole data. Therefore, we proposed a hypothesis that if we cluster data with machine learning techniques before prediction, and assign similar samples in smaller sub-datasets, then apply ML methods to each dataset, as we customized the model, we can enhance prediction accuracy. Moreover, we can examine differnt predictive models for each created sub-dataset and select the best model for each case. Then, we can calculate the overall error for the whole dataset by averaging. In this study, other than conventional regression and prediction models, we also benefit from employing Genetic programming as a symbolic regression tool and compare it with other prediction methods. As house price datasets contain differnt features with high variations, for our first case study, we used house sales data for King County, in the USA with 21,614 instances and 20 features such as the house size, location, the house condition and number of bedrooms and bathrooms [12]. Moreover, to evaluate our proposed method for small size datasets with completely different structure, we also applied it to an agricultural dataset. We chose a smalled sized dataset, which has both a smaller number of samples and variables with 10 features and 7549 samples. The goal for the second experiment is predicting water evaporation based on known features such as temperature, wind and humidity, in different locations, in India [13]. In addition, to examine our approach for high dimensional datasets, as the third case study, we considered the Facebook comments volume dataset, which includes approximately 41,000 samples with 54 features [14]. The goal is to be able to predict the number of comments that each post may receive.

1.3 Thesis Outline

This study is based on the hypothesis which is defined in this chapter, that clustering a dataset before prediction, and creating customized models may improve prediction accuracy. The second chapter includes a background review of the clustering and prediction methods that we employed in our study. This chapter, first, defines three of the most common machine learning clustering techniques and subsequently explains the five models, which provided the best estimation for house price prediction among various models that we primarily examined for house price predictions. In addition, this chapter reviews the research works on methods related to house price estimation and our proposed method. Chapter three provides more details about the proposed method, utilized methods, evaluation technique that we used and our method advantages and challenges. Our experiments and results are provided in chapter four. In this chapter, we initially defined the setting for each clustering and prediction model. We then focused on our main case study, house price dataset results, which demonstrate the proposed method potential in improving prediction accuracy and its challenges. Finally, we provided the results of two other datasets, which we utilized to examine the results' variations when we change the data size and number of features. Chapter five consists of the comparison among the different experiment results and conclusions, which are followed by future work suggestions. At the end, in the appendix, the details regarding the examined datasets samples are provided.

Chapter 2

Background and Literature Review

2.1 Background Review

The following section provides a background review of the machine learning fundamental concepts that are specific to the research work done. We discuss the two principal tasks in machine learning: Clustering and Regression. First, we present three fundamental clustering methods and other conventional techniques. Then, we review seven predictive models utilized in our research.

2.1.1 Clustering Algorithms

Clustering is a principal technique extensively used for investigating the intrinsic data structure in machine learning and pattern recognition. Clustering is an unsupervised machine learning task that partitions the data that has not been labelled, classified or grouped. Most of the conventional methods focus on modeling the similarity among data points in a way that instances in each cluster are more similar to their cluster members than to those which are assigned in other clusters [15]. Based on the nature and structure of dataset, various

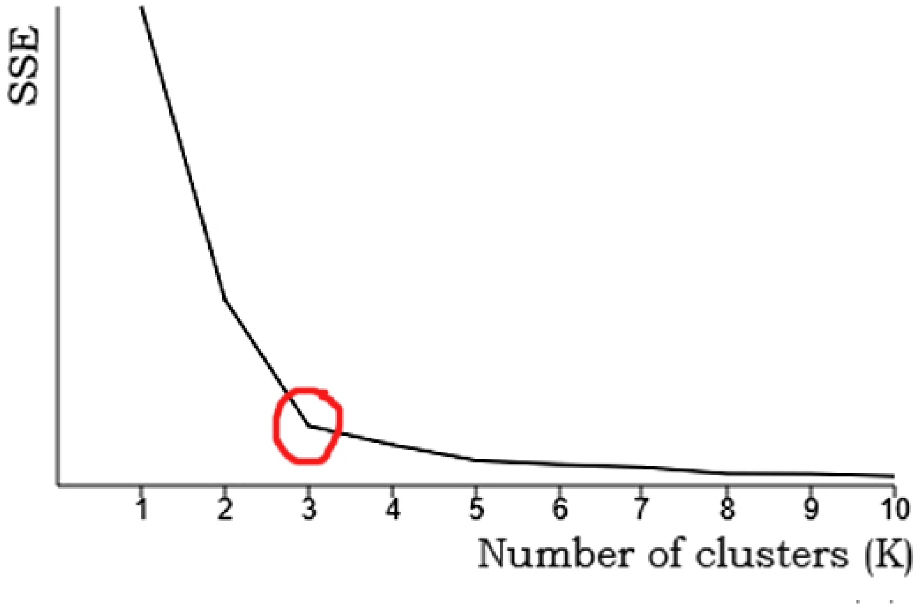


Figure 2.1: Elbow method for determining optimum value for K.

clustering methods such as; K-means, DBSCAN, Mean-shift, Spectral clustering,and Hierarchical agglomerative clustering, should be examined to recognize the hidden patterns in data and its structure successfully [16]. For instance, DBSCAN is very powerful in detecting the arbitrarily shaped clusters. It is also can detect the completely surrounded by, but not connected to, a different cluster. But when two or more neighbour clusters are connected and have different density, DBSCAN method fails in clustering. In contrast to DBSCAN, K-means has the potential to partition the data points which are closer to each group but it is weak in detect the surrounded clusters, arbitrarily shaped clusters and the neighbour clusters which are very closed to each other. Mean-shift method is also weak in detecting surrounded clusters and arbitrarily shaped ones, but can partition close clusters. In addition, it does not need pre-specified number of clusters. Hierarchical agglomerative methods are successful in grouping the arbitrarily shaped clusters and some of surrounded clusters but it cannot partition the near neighbour clusters. Therefore, we applied different common clustering methods on our dataset and selected the most accurate ones.

In order to apply clustering models, we employed scikit-learn [17] libraries. It is a Python module incorporating software for medium-scale supervised and unsupervised tasks in machine learning. The scikit-learn libraries include different algorithms in classification, regression and clustering.

K-means Clustering

K-means was the first clustering technique that we utilized. It is an extensively used clustering technique that minimizes the average squared distance between instances in the same cluster. This technique groups instances into a predefined number (k) of clusters based on the nearest mean distance of each data point to the cluster members [18]. Hugo Steinhaus used this method for the first time in 1956 [19]. Stuart Lloyd [20] provided the first standard algorithm for it that Forgy menthioned it in his research [21], but for the first time, James MacQueen proposed the term ”k-means” [22].

If we have a data points set: $(X_1, X_2, ..., X_n)$ that every data point, X_n is a real vector, K-means cluster the dataset with n instances, into k sub-sets which K is a number smaller than n; $S = S_1, S_2, ..., S_k$. The K-means algorithm group data by trying to categorize data points in k clusters of equal variance. Therefore, the K-means objective is given in Eq. 2.1.

Min Sum of $\mid\mid X - \mu \mid\mid^2$:

$$arg_s min \sum_{i=1}^{k} \sum_{X \in S_i} \mid\mid X - \mu_i \mid\mid^2 = arg_s min \sum_{i=1}^{k} \mid S_i \mid Var S_i \tag{2.1}$$

Where μ_i is the mean of points in S_i . Moreover, we can write the Eq. 2.1 in other shape of minimizing the squared distance of each pair of samples in each cluster. Then the objective is Eq. 2.2.

$$arg_s min \sum_{i=1}^{k} 1/(2 \mid S_i \mid) \sum_{X \in S_i} \mid\mid X - Y \mid\mid^2 \tag{2.2}$$

K-means clustering has some challenges, we need to realize the best value for number of clusters, k. Based on the number of instances, we can change k manually to experimentally find the proper value. If majority of the samples are assigned in the same group and very low number of instances in other groups, then we increase the number of clusters to determine if the group with large number of instances breaks down to smaller groups. Moreover, there are some techniques that we can utilize to find out the proper value for K. For instance, the elbow method and Silhouette score are effective methods to validate the optimal number of clusters for K-means .

In the **elbow method** [23], we apply K-means on the dataset for an estimated range of values of k (for instance, k from 2 to 10), and for every value of k consider the sum of squared errors (SSE). Then, plot a line chart of the SSE for different assigned k. If the line chart is similar to an arm, like Figure 2.1, then the elbow of the arm is the best value of k. The reason is that we need to minimize SSE. But SSE has a decreasing trend to zero when K increases (the SSE is equal to zero when each data point is considered as a cluster therefore, k is equal to the number of instances in the dataset it results in the error equal to zero). Therefore the goal is finding a small value for k which has a low SSE, and the elbow displays the point that SSE starts to decrease significantly by increasing k.

Silhouette scoring method [24] can be utilized to evaluate the clustering by calculating and scoring the distance between the resulting clusters data points. The silhouette score shows the closeness of each instances in cluster to the other members in the neighboring clusters. Silhouette value is based on sum of average distances of data points in each cluster comparing to average distances to members of other clusters. This method, creates a scoring parameter with value in range of -1 to 1. In the other words, the Silhouette score is a measure of how similar a sample is to its own cluster compared to other clusters. Silhouette score close to +1 shows that the data point is far away from the other clusters. Score of 0 implies that the instance is very close to the division between two clusters and negative score shows that sample is assigned to the wrong cluster.

DBSCAN Clustering

Density-based spatial clustering of applications with noise (DBSCAN), is a clustering algorithm which detects core samples of high density areas and expands clusters from them. It is practically beneficial for type of data that include groups of samples with similar density. First time, Martin Ester, Hans-Peter Kriegel et. al suggested this method in 1996 [25]. DBSCAN searches the each data point's neighbourhood in a circle with ϵ radius around the data point and count the number of other data points in this circle if it is equal or larger than the value of the special parameter, minPts, the central point is considered a core point and member of the cluster. All other points which are reachable from the core points of each cluster, are also member of that cluster. Otherwise, any other points are outliers or noise points. This technique is illustrated in Figure2.2. DBSCAN performs perfect in partitioning of high density clusters versus low density ones in data point space of dataset. In addition, it is powerful in detecting outliers in the dataset. On the other hand, it is weak when a given dataset contains varying densities areas. While DBSCAN is great at separating high density clusters from low density clusters, other disadvantage of DBSCAN is that it cannot partition the clusters of similar density which are placed beside each other. In addition, it does not work accurate in high dimensional datasets which include many features.

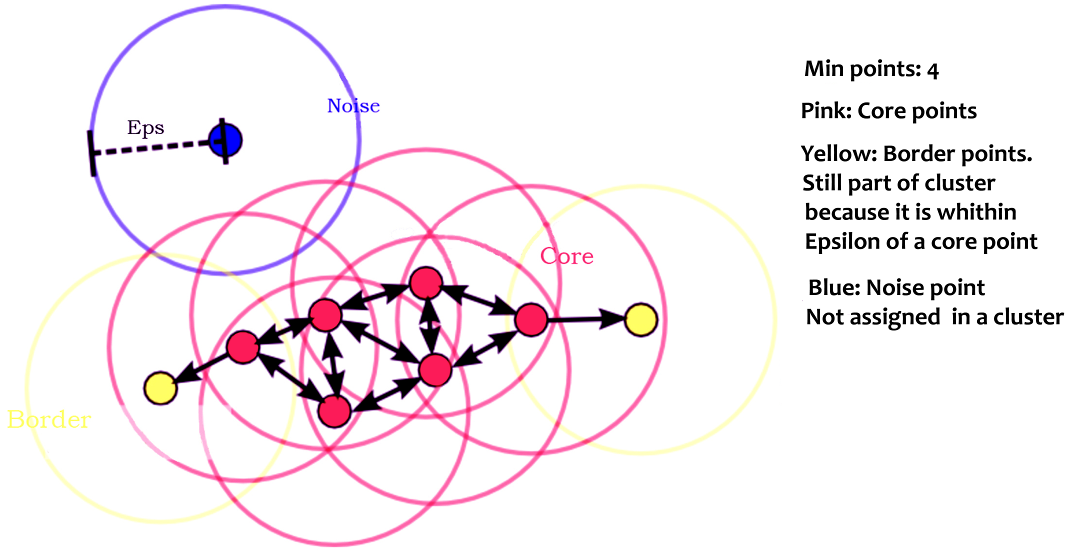


Figure 2.2: DBSCAN clustering method (adapted from [26]).

Hierarchical Agglomerative Clustering (HAC)

Hierarchical clustering [27], [28] is one of the most common clustering techniques. It builds clusters by combining clusters or splitting them. This hierarchy creating clusters can be illustrated like a tree or dendrogram. This clustering technique is divided into two types: Agglomerative and Divisive. Divisive is a top-down approach that initially, all data points are assigned to one cluster. Recursively, this cluster and other creates ones are splited hierarchy. Agglomerative method, which we utilized in our study, is a bottom-up approach that initially, each instance is in its own cluster, and pairs of clusters are merged successively to create new larger clusters. At the end, all of them merge to the one cluster which includes all samples. This method can also depicted like an upside down tree. The whole data set which is a univalent cluster is the root of the created tree. The tree branches are the middle the leaves are data points that can be considered a cluster with only one

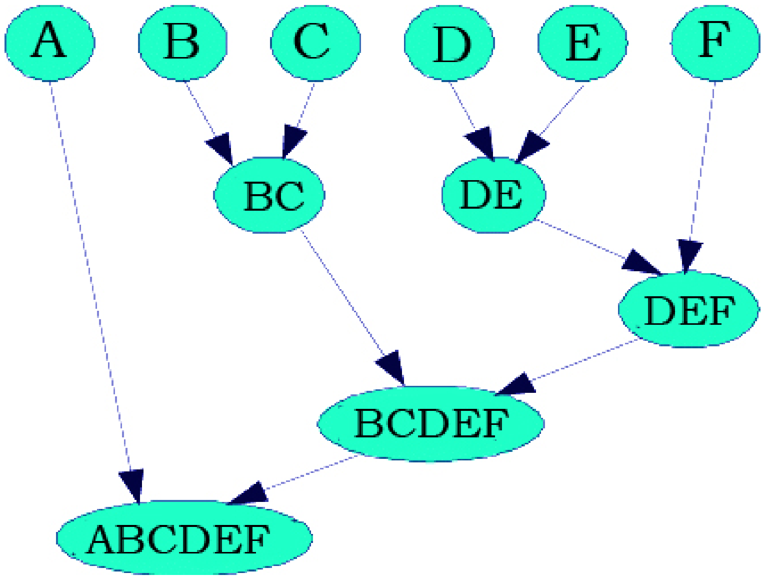


Figure 2.3: An example of Hierarchical Agglomerative Clustering method.

member. In HAC, the linkage criteria is the metric employed for the merging procedure. It uses differnt linkage strategies: ward, complete linkage, average linkage and single linkage. Ward method, minimizes the sum of squared distances between created clusters. In other words, it minimizes the variance and can be considered similar to the K-means objective function but works with an agglomerative hierarchical procedure.

Complete linkage (or maximum linkage), minimizes the maximum distances between all couple of samples in each cluster. Average linkage minimizes the average of the differences within all pairs of instances in clusters. Single linkage minimizes the distance between the closest pair of samples.

HAC process needs large space and it is computationally expensive approach. Moreover, because HAC has complexity of $O(n^3)$ and requires very large memory, it is very slow even for medium size datasets and it fails in clustering huge datasets.

Other Conventional Clustering Methods

There are many other clustering methods other than above mentioned techniques, such as mean shift, spectral clustering, Chinese restaurant, etc.

Mean shift clustering detects clumps in a homogeneous environment. It uses an algorithm which finds the centroids based on the mean of the data points in each selected group. The algorithm keeps the nearest samples to the centroids and calculate the new center of the group this method is very similar to the K-means but it employs mean shift vectors that is calculated for each centroid that points towards a region of the maximum increase in the density of instances.

Chinese Restaurant Process Clustering (CRP) is a process similar to the order that costumers sit on different tables (clusters) in a Chinese restaurant [29]. The first person sits down at a table (the first cluster). The second person that comes in, sits at the first table with probability $1/(1+\alpha)$ or at another table with probability $\alpha/(1+\alpha)$. The i^{th} customer selects an previously selected table with probability proportional to the number of customers already are sitted at that table, or sits at a new table with a probability proportional to the value α as shown in Eq. 2.1.1 [30]. If the i^{th} person selects the table z_i then:

$$p(z_i = k \mid z_i, \alpha) = \begin{cases} \mathrm{N}_{k,(-i)}/(N+\alpha-1) \text{ , if k is occupied, i.e.} N_k > 0, \\ \alpha/(N+\alpha-1) \text{ , if k is a new table, i.e. } k = k^* = K+1 \end{cases} \tag{2.3}$$

Where $z_i = (z_1, z_2, ..., z_{i-1}, z_{i+1}, ..., z_N)$

and $N_{k,(-i)}$ is the number of customers seated at table excluding customer i. A new customer prefers to select a table with more number of customers comparing to other tables. Therefore, the Eq. 2.1.1 demonstrates that the CRP process can be explained by a rich-getricher property in which the probability of being assigned to a group improves by proportion to the number of people already have sat at that table.

2.1.2 Regression Models and Prediction Techniques

Regression is a supervised machine learning technique that creates a model or mapping function f from the input variables (X) to predict output values of a desired target (Y) when the target values are continuous. Whenever there is a new input data (X), the output variable Y = f(X) for predicted value. There are numerous regression and prediction techniques. Each method has its own importance, advantages, disadvantages and limitations. We employed the most common approaches for the prediction stage of our research an selected the most accurate ones for the rest of our study.

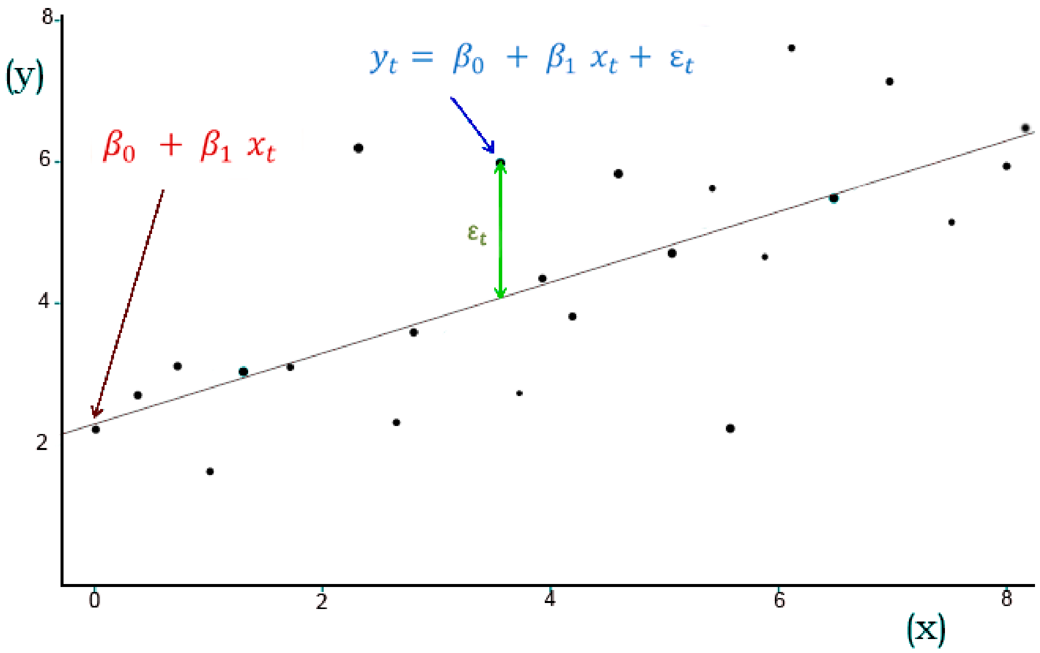


Figure 2.4: An example for a linear regression .

Linear Regression

Linear regression is a supervised machine learning linear approach for modeling the relationship between dependent variable (Y) which is the target feature and one or more independent variables (X). The best fit to this data is straight line which follows the Eq. 2.4 and depicted in the Figure 2.4.

$$y_t = \beta_0 X + \beta_1 + \epsilon_t \tag{2.4}$$

In Eq. 2.4, β_0 and β_1 are coefficients of equation and the goal in creating linear model is to find this coefficients. Linear prediction models are usually fitted by the least square method, but sometimes, they can be fitted using other approaches. For example, we can minimize the lack of fit with least absolute deviations regression. Linear regression is an effective technique in many problems such as forecasting or error reduction, explain variation in the response variable and recognizing the correlation between target feature and dependent variables even for some explanatory variables which may have no linear correlation with the predicted values. In addition, when dataset contains a large number of variables, it is probable to obtain low prediction accuracy for the test data, comparing to the training error. Over generalization or over fitting can cause this problem. Ridge and Lasso regressions use approaches such as L1 and L2 Regularization to reduce over fitting and model complexity which may happens in simple linear regression. When given dataset contains some noise points, the model fits the pattern considering this noise data. Consequently, provides good score for training but fails in the test prediction and generalization of created model. Because fitting the model to all points including noises, cause a high variance in model and over fitting. One solution can be L1 (Lasso) or L2 (Ridge) regularization.

Lasso Model

Least absolute shrinkage and selection operator or LASSO, execute variable selection and regularization to improve the prediction accuracy. First time, it was suggested in geophysics field [31] but later, Robert Tibshirani developed the method and named it LASSO [32]. It is a Linear Model that is trained with L1 regularization approach. When some features have weights closer to zero and does not have an important effect on prediction, L1 method shrink the related coefficients to zero. In this method, a cost function is added that provides penalty of the importance of the coefficients. As shown in Eq. 2.5, for N samples, the penalty term λ regularizes the coefficients. When the coefficients have large value, the optimization function will be penalized:

$$MinL(x,y), L(x,y) = \sum_{i=1}^{N}(y_i - h_\theta(x_i))^2 + \lambda \sum_{i=1}^{N} \mid \theta_i \mid \tag{2.5}$$

Therefore, LASSO shrinks the coefficients and is benefitial in reducing over fitting. In addition, it can performs well in feature selection. when λ is close to zero, the cost function is alike to the linear regression cost function and the model will resemble simple linear regression model.

Ridge Model

Ridge regression which is also recognized Tikhonov regularization which is proposed by Andrey Tikhonov. It also is a Linear Model that is trained with L2 regularization method. Similar to LASSO, in Ridge method, a cost function is added to consider penalty of square of the value of the coefficients. As shown in Eq. 2.6, for N samples, the penalty term λ regularizes the coefficients:

$$MinL(x,y), L(x,y) = \sum_{i=1}^{N}(y_i - h_\theta(x_i))^2 + \lambda \sum_{i=1}^{N} \theta_i^2 \tag{2.6}$$

Again, the penalty term λ, regularizes the coefficients with large values. Ridge regression eliminates that coefficients and results to decreasing the model complexity. Similar to

LASSO, when we have small λ, near to zero, on the features, the model is alike linear regression model.

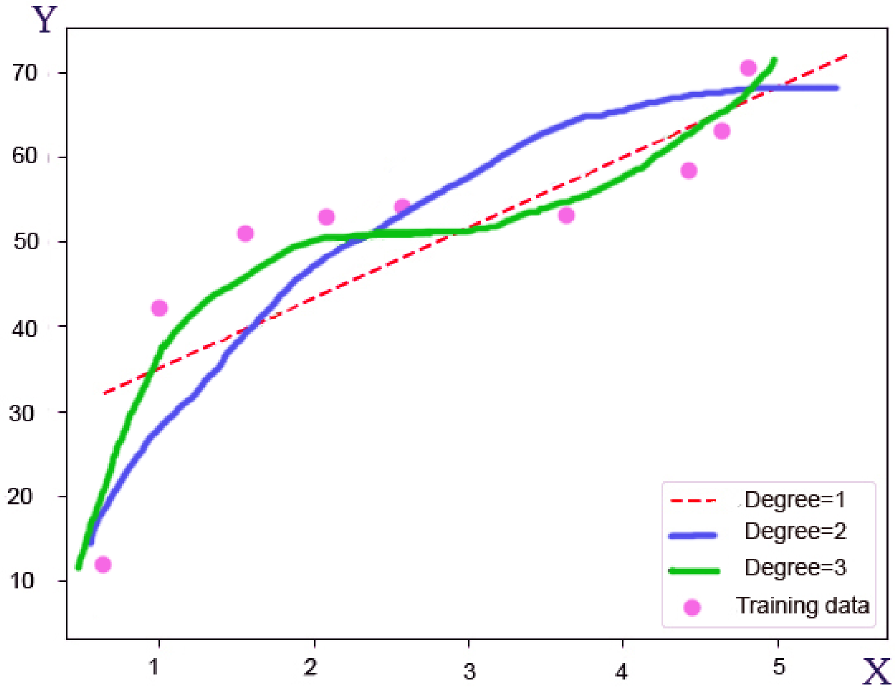


Figure 2.5: Comparison of linear and differnt order of polynomial regressions.

Polynomial Models

In some cases, the straight line which presents linear model, cannot fit the given data points and find an accurate patterns in the data. In this regressions we obtain low accuracy in RMSE and R^2 score for linear models. In some datasets, increasing the order of predicting model and as a result, improving the complexity of the model may solve the under fitting problem. We only need to create a model similar to the linear regression and only add higher order of dependent variables to the equation such as Eq. 2.9 which target variable, y, is defined as an nth degree polynomial model of explanatory variable x.

$$y_t = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon_t \tag{2.7}$$

$$y_t = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \epsilon_t \tag{2.8}$$

$$y_t = \beta_0 + \beta_1 X + \beta_2 X^2 + ... + \beta_n X^n + \epsilon_t \tag{2.9}$$

Usually we do not need the equations with the decrees higher than 2 or 3 similar to Equations Eq. 2.7 and Eq. 2.8. As Figure 2.5 demonstrates, increasing the order of the model may increase the problem of over fitting in training data.

Support Vector Regression (SVR)

Support vector regression, SVR, considers the data as a series of points within a space. The model is a hyperplane with maximum margin such that maximum number of data points are located within that margin [33]. SVR approach is similar to support vector machine, SVM, classification algorithm [34]. Instead of minimizing the error rate in simple linear model, SVR fit the error within a specified border. As depicted in Figure 2.6, SVR objective is to create the hyperplane that contains maximum number of points within the margin. The concept of a SVM was first proposed in 1964 by Aizerman et. al [35]. It was the first time that was suggested we can employ learning machine to classify data with a very high number of features, similar to the capabilities of the human brain that can cop with many features. The hyperplane in SVM is the partition line between the different data classes but in SVR it is similar to the linear model that created to predict the target variable. In addition, we use kernel function to map a low dimension data into a high dimension space. We consider two boundary lines create a margin and support vectors are the instances with minimum distance to the boundary lines. For linear predictions we use radial basis function (RBF) but for none-liner regression we utilize every dot/inner product $\langle w, x_i \rangle$ kernel to map the data to the dimensional feature space. The goal is minimizing the Eq. 2.1.2.

$$\text{Minimize (1/2 } \| \; W \; \|^2 + C\sum_{i=1}^{n}(\zeta_i^* + \zeta_i))$$

$$\text{Subject to} \begin{cases} y_i - \langle w, x_i \rangle - b \leq \epsilon + \zeta_i^* \\ \langle w, x_i \rangle + b - y_i \leq \epsilon + \zeta_i \end{cases} \qquad (2.10)$$

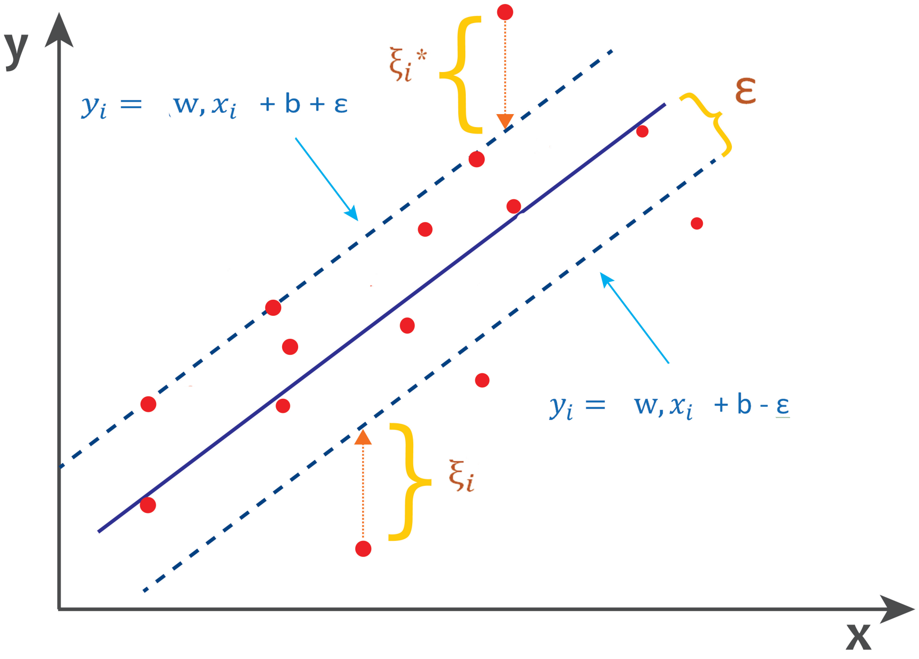


Figure 2.6: One dimensional support vector regression (SVR) model.

In Eq. 2.1.2, w is the weight vector which is learned from data, ζ_i is the distance between the bound lines and predicted variables outside the bounds. C is a constraint value which is the controlling parameter for the penalty applied on data points located outside the bounds to reduce over fitting.

Artificial Neural Network (ANN)

Artificial neural networks are sets of the most important machine learning approaches. The Idea is inspired by human brain and its neural systems and tries to simulate the brain learning. This systems learn to accomplish tasks by using examples without any specific programs or special rules similar to the brain that learns from experience. For the first time, Warren McCulloch and Walter Pitts [36] suggested a mathematical model for neural networks using threshold logic algorithms. After that, many scientists developed regression and classification methods based on ANN. In 1992, White gathered many articles about ANN and advanced statistics [37]. In the early stage of using ANN the computers were not powerful enough to accomplish ANN tasks beneficially but still many research works employed it for accurate regressions comparing to other conventional methods [38]. After improving GPU systems in recent decade, again it become widely used in machine learning and deep learning tasks [39], [40].

An ANN is a technique that connect the basic unites or nodes called artificial neurons which create a neural networks or ”perceptrons”. As illustrated in Figur 2.7, a multi-layer perceptron (MLP) consists of different layers of neurons, at least, three layers, that includes an input layer, middle hidden layer(s) and output layer. Neural network models can be considered as a function with input vector X and Output vector (target feature), Y. Neuron’s network function f(x) is defined as a combination of $g_i(x)$ functions. Many combination of functions can be used but usually it is the nonlinear weighted summation of functions. Mathematically it can be defined as $f(X) = H\sum_{i=1} w_i X_i g_i(X)$ which w_i is the weight of each function and H is activation function that consists of some predefined functions,

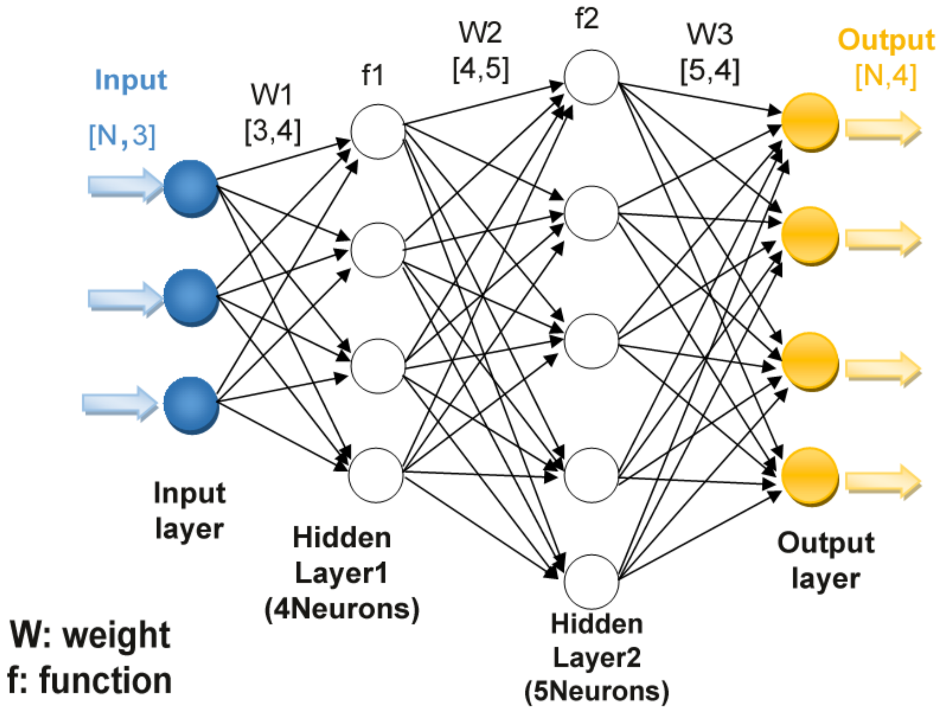


Figure 2.7: A multi-layer perceptron with 4 input neurons, 2 hidden layers and 3 output neurons.

such as Sigmoid function, the hyperbolic tangent, Softargmax or normalized exponential function and rectifier function. The activation function is responsible to create smooth transition with input values variations in such a way that minor change in input only causes a small change in output.

Symbolic Regression

In symbolic regression, we try to create the model, which best fits the measured data [41]. In 1985, Cramer proposed one of the first tree-structured evolutionary algorithms that could be used in basic symbolic regression. John Koza [42] was the first person, who developed

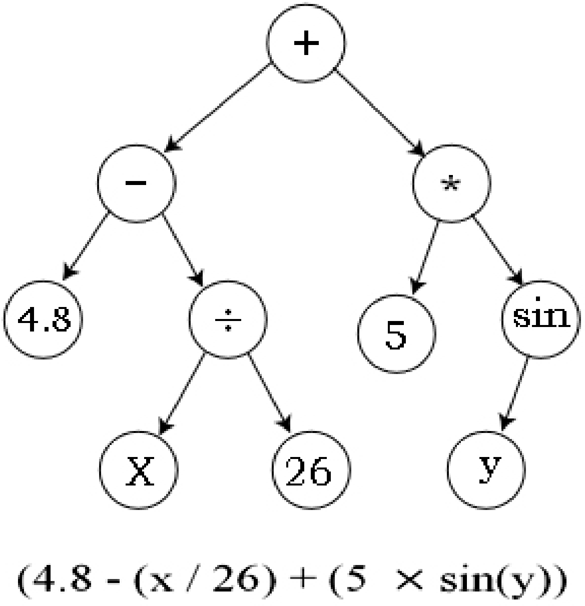


Figure 2.8: Creating equation with GP using LISP language. LISP can create tree structure relationships.

Genetic Programming in LISP, one of the earliest programming languages, and also it was shown that GP is a powerful tool in problem solving, including symbolic regression. In addition, John Koza showed GP can be applied in automatic functions by discovering an approximate value for the impulse response function in linear time invariant systems. It was a great improvement in machine learning regression methods [43]. In 1995, using a numerical approach, Iba presented a novel variant to GP, which merges an adaptive search of tree structures in GP, and a system recognition method to the discover local parameters by using statistical search [44]. Moreover, Montana proposed a new type of genetic programming that can use data constraints by restricting generated programs, avoids large computational time. After Montana's proposed method, the GP became more powerful and fast in solving the multi-dimensional least-squares regression problems [45]. After the first international conference on GP at Stanford University in 1996, many scientists such as Sian studied a

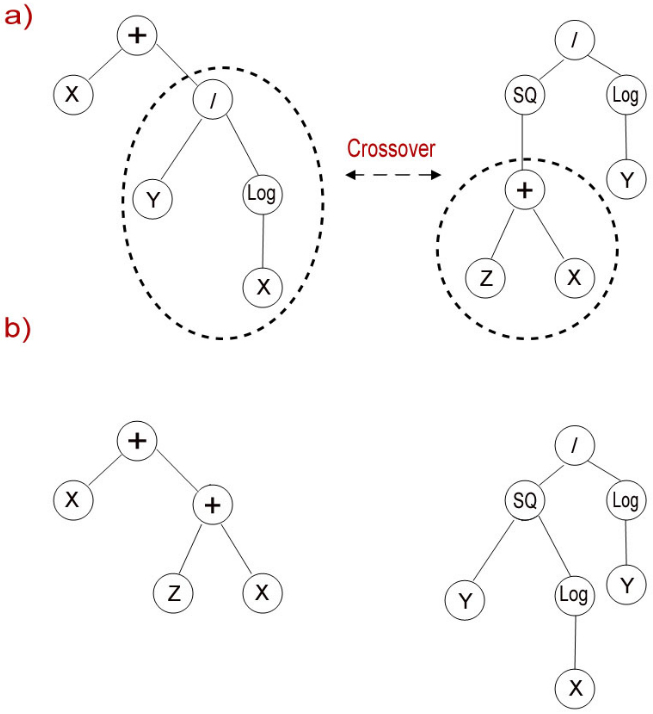


Figure 2.9: Example of creating new equations in GP by crossover. (a) shows the parents which by applying crossover and switching two parts of their equations, we create the new offspring (b).

distributed method for parallel GP on the internet in order to use the computational power of internet for solving hard problems [46]. In the year 2000, Augusto used Read's linear code for tree structures to increase simplicity and to enhance GP's performance in solving symbolic regression problems [47]. Many researchers have tried to discover the correlation of genetic programming results with the improvement of population diversity [48], [49], which resulted interested in utilizing GP in different fields; this has opened a new horizon in evolutionary computing. In 1997, Willis tackled engineering applications by employing GP [50]. Gustafson improved GP for symbolic regression [51]. GP is not only valuable in theory but also it has proven its effectiveness in regression in practice [52]. Regres-

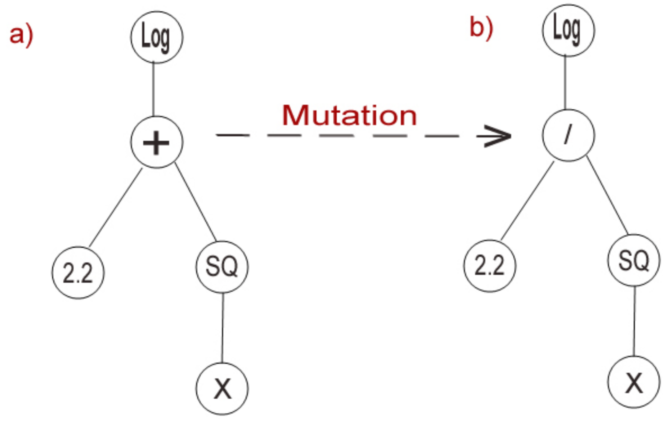


Figure 2.10: Example of creating new equations in GP by mutation. (a) shows a parents which by applying mutation and changing one parts of its equations, we create the new offspring (b).

sions with GP has many advantages and also faced with some challenges [53]. It is proud of its wide search area and the power of discovering functions which are combination of simple functions for complicated systems. Symbolic regression can be very accurate and has the potential to reveal hidden system characteristics [54]. The most challenging part of symbolic regression is recognizing the most appropriate model that gives a prediction for given data and is expressed analytically utilizing the minimal set of input variables and the given set of fundamental operations. In conventional regression methods, data analysts try to fit pre-defined models (linear or polynomial equations) on a given dataset. In these techniques, the function is known and we only need to find coefficients of that function. Based on the nature of the given dataset, we cannot always obtain the acceptable accuracy. Sometimes each part of data fits a different function or the pattern is very complicated in its global shape. In contrast, symbolic regression attempts to not only find models structure but also discover the model parameters that infer the model from the data [47]. This task can be done by employing genetic programming [55]. As depicted in Figure 2.8, GP is a useful and effective technique for creating symbolic equations, computer programs or models using a process similar to evolution in nature, by utilizing selection, crossover (Figure 2.9), and mutation (Figure 2.10). It is a method of programming which utilizes the idea of biological evolution and natural selection to solve complicated problems. Initial expressions which are pieces of programs, formed by randomly combining mathematical building pieces including mathematical operators, constant values, analytic functions and variables, contest each other and only the most proper programs can survive. New equations are then formed by recombining previously survived equations and compete with another piece of programs in the next generation. As the result, through many iterations, it continuously approaches to the best solution which has the best fitness for the given data. Therefore, GP can be used in creating a data-driven model [56]. In contrast to conventional data mining regression models with limited global search abilities, GP provides a strong capability of the global search item of the general structure of the model [57].

2.2 Literature Review and Related Works

In most management and decision making tasks, analyzing the available data and evidences is essential to make correct predictions. No method exists with hundred percent prediction accuracy but data analysts try to improve accuracy by using innovative techniques. Regression is one of the most important and beneficial techniques in data analysis which made predictions more accurate (and faster in some cases). In many cases linear models can provide acceptable accuracy. Using linear models is usually practical because of their low computational cost and avoiding over fitting problem [58]. But sometimes, linear models cannot provide the desired accuracy [59] and more complicated models are needed to predict target value with required accuracy, there are many none-linear prediction techniques that it is not easy to figure out which one can create best model for a given dataset [60]. One approach is that examine different methods and select the best one for the dataset [61] or we can search the successful applications for similar datasts [62]. Price estimation has an remarkable role in various businesses. It includes several economic activities such as; sale and purchase , transfer money, financing, tax assessment, estate settlement and finantial investments. Population transfer settlement of new citizens and Money transfer. Consequently, modeling the house price has been a target of many research works in recent yeras [63]. Some early house price predictions use semi-parametric estimation. First, they create the predictive model, then, the predictions of the models are compared together by calculating the distribution of the predicted price and use it for estimating the associated prediction intervals [64]. Another conventional model which first created for house price prediction, is hedonic regression (hedonic demand theory) to estimate house price. Assumption for linear regression models is that the property value is a weighted summation of home characteristics. This regression do not perform well for outliers which are very common in house sale. In addition, they cannot address non-linearity within the data. This method which still is one of common techniques, consider the effect of differnt parameters that impact the target value [65]. In economics, hedonic regression is a ascertained preference technique to predict demand or value. This method split the object being researched into its constituent parameters. After an determinative vector is assigned to each parameter, the model calculate the participation ratio of each characteristic. It considers a nonlinear logarithmic relationship between the price and independent variables. Hedonic models usually use conventional statistic rules and regression analysis [66]. More generalized models, including sales regulation nets, are special cases of hedonic models. Another primary models for predictions is Fuzzy logic. This technique assign real numbers between 0 and 1 to truth values of variables and considers concept of partial truth values. Fuzzy logic models are employed in discovering, interpreting and utilising ambiguous data with lack certainty. Therefore, house price predictions based on this method can provide a fuzzy set output [67]. Many regression models, including house price predictions use ordinary least squares (OLS) models to fit data. This approach can be used to figure out the most important features such as nearest neighbor transactions [68]. House price prediction usually can be done by comparing similar cases with the target house therefor, K nearest neighbor is another effective models in house price prediction [69], [70]. KNN is used in Regression on Feature Projections (RFP) to improve accuracy. RPF employs two averaging procedures applied on dataset sequentially to predict the target variable. Other method, case based reasoning (CBR) is not very successful for many datasets and even house price prediction needs complicated models but for the data set with many features, such as house price predictions, it is capable to decrease the prediction error comparing to other conventional models such as KNN [71]. In recent years, neural network became one of the most effective and accurate techniques in regression especially for large size datasets that provide enough training data for creating an accurate model. Many researches employed this technique to improve accuracy in one of the most popular regression problems, house price prediction. Artificial neural networks overrides many conventional regression methods such as hedonic which was the most reliable technique in house price estimation [72]. Especially when researchers added images to house data, the problem become very complicated that could not be solved with out utilizing neural network [73], [74].

We proposed a model that if we cluster data before applying prediction models, we may improve the prediction accuracy. It is conventional in research works that employ more than one technique sequentially and create a model to enhance predictions [75]. In addition, clustering can be used for feature selection and removing outliers. Bekoulis et. al proposed a method to enhance house price estimation accuracy [76]. They used textual advertisements to broken down the estimation task into three sub-problems. First, in sequential labeling, the model should identify the important entities of the house and dependencies between them (e.g., rooms, bathroom) from classifieds. Second, structure the identified parts into a tree-like format. This steps can be done either one by one, in a pipeline approach, or simultaneously in a joint model. Soni, Ansari and Sharma suggested a combined method for heart disease prediction [77]. They employed a clustering method as a prepossessing step before applying classifiers such as KNN and neural network on the data. Actually they employed clustering to group similar elements to reduce the data size to obtain the optimal subset of feature, adequate for heart disease prediction. Quackenbush also used classifications based on clustering for micro array data [78]. As genes have some elements, grouping them can enhance the classifying accuracy. The most similar work to our proposed method is the Wang et. al's approach in their research [79]. In this method, a fuzzy cluster module, categorizes a given data into differnt groups with homogeneity of the clusters, assigning relevant data in the same cluster and dissimilarity between clusters, and samples in different clusters should be as disparate as possible. After the training set clustered into various subsets, ANN Utilized for prediction. considering the fact that the size and complexity of each training subset is decreased, the efficiency and accuracy of subsequent neural network improved.

Chapter 3

Proposed Method

In following chapter we are going to define the details of our proposed approach and explain the techniques that we utilized to examine our hypothesis.

3.1 Ensemble Based Prediction Using Clustering and Regression

When there is high diversity in values of differnt features of a dataset, fitting a single model to the whole dataset may not be easy. especially when the dataset has complicated structure or each part of dataset has differnt structure, a fitted model to the whole dataset may not have acceptable accuracy. In this cases, we can split data to sub-groups and apply proper model to each group. To test our proposed method, we utilized different clustering and prediction techniques sequentially. This section provides both big picture of whole process and the details of the proposed method. The main contribution of this study is improving prediction accuracy by customising prediction models for sub-datasets or groups which are created based on similarity between the members of each group. The process is depicted Figure 3.1.

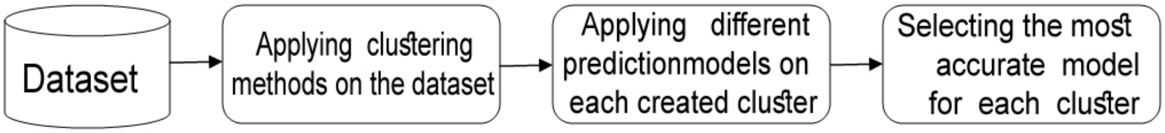

Dataset
Applying clustering methods on the dataset
Applying different predictionmodels on each created cluster
Selecting the most accurate model for each cluster

Figure 3.1: Summary of the proposed method.

3.1.1 Applied Techniques

Usually in the estimation task, various models are applied to the whole dataset to find the best model with a higher accuracy. But in real world, especially when we have high volume of samples, it is probable that we have high diversity in the samples characteristics for example, it may not be accurate if we model price estimation of all houses in range of prices in 100K dollar to 10M dollar. In addition, we compare a small house with a larger one, located in a big lot with many bedrooms, bathrooms and swimming pool. Therefore, it may not be appropriate to create a single model to predict both group of prices, cheap ones and very expensive ones because it may not be an accurate model to fulfill the whole dataset. Therefore, if we group the instances to different categories based on the similar characteristics and then find the best model for each group, we may be able to make more reliable models. But how we can find the best categories to group the data properly? We can group the instances based on the most important features. For this task, we can rely on experts knowledge to know which features are the most important ones. For example, for grouping diseases we can ask a doctor or for predicting house price we can ask real-estate agents to know which feature is the main factor to estimate the house price. For instance, location, size, number of bedrooms, etc. On the other hand, when we have many features, it may not be easy for human to find the most similar instances. In this cases, data scientists employ computing machines to carry out this difficult task using clustering methods. In clustering techniques we create sets of entities in such a way that items in the same cluster are more similar to each other than the objects in other groups. Based on the technique that we use for grouping, there are many different clustering methods which result in different type of clusters.

In the first stage of experiments we apply various prediction methods, such as, linear and polynomial regressions, SVR, Neural network (multilayer perceptron), genetic programming and other methods on the entire data on order to find the best model with the lowest error in estimation the target value.

For the second stage, we apply our proposed method. Our approach consists of two steps which is depicted in Figure 3.2.In the first step, we cluster or group data using **Algorithm-based clustering** or **Expert-based grouping**.

In **Algorithm-based clustering**, we use different machin learning clustering methods such as k-means, DBSCAN and agglomeration hierarchical clustering algorithms.

In **Expert-based grouping**, we group data based on the human knowledge.

In the second step of our proposed method, for each generated cluster or group, we apply different prediction methods that we have utilized in the first stage for the whole data. Now we can compare the prediction accuracy of different models on each group and select the best one. Moreover, we can compare the best model with the most accurate model in the first stage. Many parameters affect each model's accuracy such as nature of data, the number of features and the size of the dataset (the number of instances in the dataset). Therefore, we studied the proposed method's performance on three different dataset with completely different nature and characteristics: 1. Our main dataset is house sales data for King County in the USA, 2. Agricultural water evaporation dataset, and 3. The Facebook comments dataset. When a new sample, such as a new house comes to the market and we need to predict the target value, first, we assign it to one of the created clusters, based on

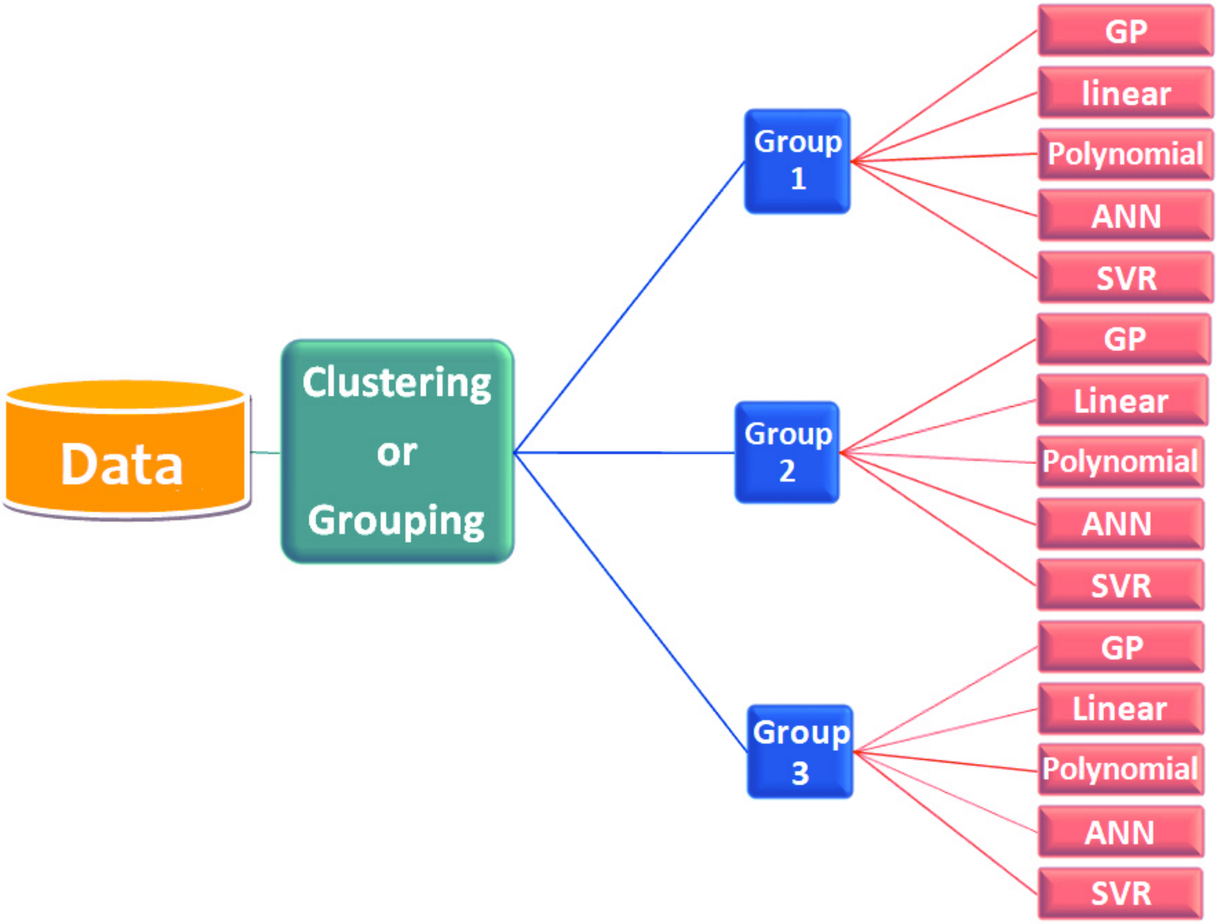


Figure 3.2: Configuration of the proposed method: after clustering, we apply different prediction techniques to each group.

the object's distance from the centroid of the clusters. Then, Using the best model that previously found for the closest cluster to the new object to predict the target value such as the house price. The process is illustrated in Figure 3.3.

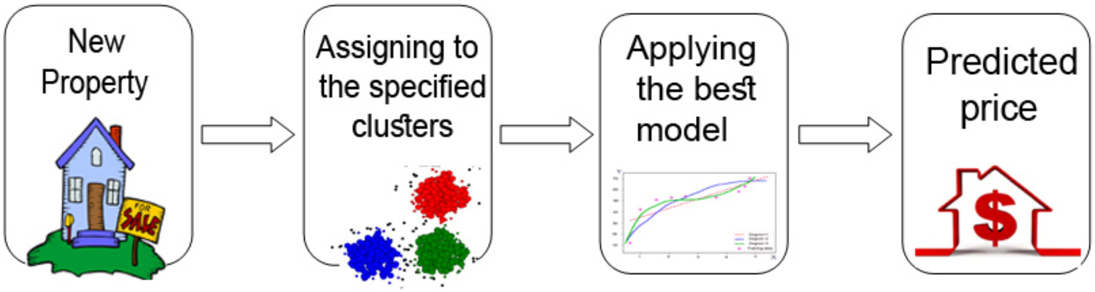


Figure 3.3: Configuration of the proposed method for new samples. First, we assign it to one of the created clusters, based on the object's distance from the centroid of the clusters. Then, using the best model that previously found for the closest cluster to the new object to predict the target value such as the house price

3.1.2 The Evaluation of Model Performance

In this study, we applied different prediction and regression methods on three different datasets to find out if the process is applicable to different datasets to find the best model with a higher prediction accuracy for each case, in order to provide a comprehensive analysis regarding the proposed method abilities and dataset types. For calculating accuracy, we consider the normalized relative error which is calculated by the Eq. 3.1:

$$Er = \mid y_e - y_r \mid / y_r \tag{3.1}$$

(Where in Eq. 3.1 Er, y_r and y_e are the error, real value, and estimated value, respectively.) If we consider the average error of n instances, we call it normalized mean absolute error, NMAE which is one of the most common metrics for evaluating accuracy of continuous variables, Eq. 3.2:

$$Er = 1/n \sum_{j=1}^{n} \mid y_e - y_r \mid / y_r \tag{3.2}$$

MAE measures the average value of the errors in predicted values, without considering their direction. It is the absolute disparity between prediction and real value in the test sample of the absolute differences between prediction and real value. The advantage of relative error and NMAE is that it puts error in prediction into perspective and observable insight. This equation provides the error for the whole process. To avoid over fitting, we used cross validation method. In addition, we split data 70 percent for training and 30 percent for testing data. If the model has problem of over fitting in training data, we can not get a good accuracy in the test data.

3.2 The Proposed Method Advantages and limitations

Advantages

Our proposed method improves the prediction accuracy because we specialize the model for the selected similar samples in each cluster. In addition, as the whole dataset is more complicated comparing to the sub-datasets, the created models for each cluster is less complicated and can be found faster and it's processing is easier especially when we have limitations in power of our computer's processors, this approach is helpful. Moreover, after clustering finding a model for each sub-dataset, can be done in parallel and prediction for small size data is faster.

Data Size

Since after clustering, each cluster has lower number of samples comparing to the original dataset, this method is only applicable on voluminous datasets with large number of instances. Then, after clustering we still have sufficient number of samples to build a proper model for each cluster.

Computational Cost

The proposed method is able to improve the accuracy but on the the other hand, the process over all is time consuming and computationally expensive. Clustering techniques, such as K-means are very expensive and usually need a lot of time to run. Moreover the next step, studying different prediction methods, requires additional time. Therefore, this method is not proper in applications with time limitation such as real-time systems and it is only beneficial when the goal is only improving the accuracy.

3.3 Summary

- A novel approach for improving prediction accuracy is proposed.
- The first stage is applying various models to the whole dataset to find the best models with higher accuracy.
- The second stage applying our method which has two steps: first step is clustering or grouping data. We can do this task with two different approaches: **Algorithm-based clustering** or **Expert-based grouping**.

  In Algorithm-based clustering, we utilize different methods such as k-means, DBSCAN, agglomeration hierarchical clustering algorithms.
- Expert-based grouping is grouping data based on human knowledge that we ask experts to select the most important variables and we group data based on the values of that feature.
- The second step in our approach is utilizing various estimation methods that we examined in the first stage, for each cluster. We employed linear and polynomial re-

gressions, SVR, neural network, genetic programming and other methods to find the most accurate model.

- As the case study, we have used three different datasets to study if the proposed method is applicable on different datasets.

Chapter 4

Experiments and Results

4.1 Clustering Models Setting

For clustering models, Dendrogram, K-means , mean shift clustering and DBDScan, which are mentioned in Chapter 2, we employed scikit-learn libraries. Its codes and adds-on are available in its web-page [80]. We applied K-means , DBDScan, mean shift clustering and different Agglomerative clustering methods on our three datasets. Moreover, we tried to cluster data using Chinese Restaurant Process Clustering (CRP). In CRP we can control the number of samples in each group but in this way, we lose accuracy because the order of the samples can effect the clusters but we desire that create clusters only based on their similarity. By utilizing CRP techniques for our three detests, we could not reach to acceptable accuracy for any of datasets. In addition, mean shift clustering did not perform as well as the other methods. To evaluate the resulting clusters, we used Silhouette method, which is introduced in the Section (2.2.2).

4.1.1 K-means Setting

We employed scikit-learn K-means library for the first clustering technique and set variant number of clusters, 2-10 for The house price and water evaporation datasets. For the Facebook comments dataset which has a large number of instances, we used different number of clusters, 2-20. In the next step, Silhouette method, which is defined in section 2.1.1, reveals the best number of clusters, K. We set the default tolerance for stopping criteria (tol) which is 0.0001. In addition, in our experiment all features have equal weights and we selected K-means initialization scheme. This technique uses a simple semi-randomized seeding method to acquire optimal clustering which not only is faster, but also is more accurate than completely random initialization. K-means initial centroids are not selected randomly and the first cluster centres are chosen to be distant from each other from the beginning [81].

4.1.2 DBSCAN Clustering

As discussed in section 2.2.2, DBSCAN technique separates the high density areas as clusters. We chose euclidean metric to measure the distances between the points. eps or ϵ, is the maximum distance between two samples to be placed in the same cluster. In this experiment, eps parameter varies from 0.6 to 0.01 to find the best value, when eps is relatively large, close to 0.6, more than 90 percent of samples are categorized in one group. Moreover, we set the min-samples value or minPts, the number of instances in a neighborhood for each sample to be considered as a core point, varies from 3 to 6.

4.1.3 Hierarchical Agglomerative clustering

Hierarchical agglomerative clustering creates small groups considering the most similar and closest samples. Then, step-wise by combining the most similar clusters, creates other new clusters. This hierarchy of clusters can be illustrated as a tree or dendrogram. At the first levels, we have more number of clusters with more similarity in each group and at the higher levels, we have lower number of clusters with larger group size including more members. As mentioned in section 2.2.2, in this technique, we can use different linkage types including Ward, complete, average, and single linkage. In this experiment, we used euclidean distance and checked all different linkage types. As a consequence, we realized the Ward linkage which minimizes the variance of the merged clusters, is the most proper linkage type for house price datasets. Other linkages result in assigning 80 percent of samples in one group. In addition, utilizing Silhouette method, we studied the different number of clusters 3 to 9 to find the best number of clusters. Moreover, we can check the number of members in different number of clusters to find the best value for number of clusters.

4.2 Prediction Models Setting

We applied different common prediction techniques on house price data and selected the most accurate ones to utilize in our proposed method. To be able to compare the methods, if any setting was needed, we used the default setting for prediction models. The following section explains the setting details for each prediction technique.

4.2.1 Artificial Neural Network

There are many regression and prediction techniques in machine learning. Artificial neural network (ANN) is the first method which is utilized in this study. We used multi layer perceptron (MLP) which consists of five layers of nodes: an input layer, three hidden layers, and one output layer. Every node presents a neuron. A nonlinear activation function is assigned to each node. The number on neurons is equal to the number of dataset features which for the first dataset is 20, the second one is 10 and the last one is 54. To create a model for training data, MLP employs a supervised learning method called backpropagation. Its compound layers and nonlinear activation function differentiate MLP from a linear perceptron. In our MLP we used Sigmoid activation function and for each dataset, and it is trained on L1 loss (which stands for Least Absolute Deviations).

4.2.2 Support Vector Regression

SVR is a supervised learning model with a learning algorithm which analyzes data used for classification or regression analysis. For our SVR we used shrinking heuristic. kernel function converts the given data into the higher dimensional feature space to make the linear data separation possible. We choose RBF kernel (this parameter indicates the kernel type that we use in the algorithm. It can be linear, polynomial, rbf, sigmoid, precomputed, or a callable. The default value is rbf) and the coefficient C (the regularization parameter) is equal to 1.0. C is a parameter which affects the trade-off between complexity and proportion of non-separable instances. The degree of the polynomial kernel function for SVR is 3, gamma is 0.22 this parameter controls the support vectors influences (large value for gamma causes large bias and low variance in models), the independent term in kernel function, f_0 is 0, tolerance for stopping criteria (tol) is 0.001.

4.2.3 Linear model, Lasso, and Ridge

Simple linear regression fits a linear model with two or three coefficients to minimize the residual sum of squares between the instances in the dataset, and the target values predicted by the linear model. For the linear model we set Boolean intercept fit. For the Ridge model, the Regularization strength, alpha, is 0.1 and tol is 0.001, Ridge coefficients minimize a penalized residual sum of squares (L2). The Lasso is a linear prediction model that finds sparse coefficients and it can create models with a few number of parameter. Lasso can reduce the number of variables in the way that the given solution is dependent. Moreover, in certain conditions, it consider the set of non zero weights but we did not assigned weights to the features. Lasso contains a linear model trained with L1. in this study, we set alpha equal to 1.0 and tol equal to 0.0001.

In SVR, Ridge and Lasso model, we used default values for coefficients as we changed parameters and there was no remarkable improvement in accuracy, we stayed with above-mentioned default values.

4.2.4 Polynomial Model

In the utilized polynomial model, the degree of features is 2. In addition, only interaction features, features that are products of at most degree distinct input features, are acquired. Moreover, we set a bias column, the feature in which all polynomial powers are zero, (similar to the intercept term in a linear model).

4.2.5 Genetic Programming

GP is a useful and effective technique for creating symbolic equations. For GP we used Eureqa [82] to create the models which have the best fit to our data. Eureqa not only has the ability to discover the functions, it also has the power to find the relevant coefficients of that function. For creating the model, we consider most of the available functions including: addition, subtraction, multiplication, division, sine , cosine, tangent, power, exponential, logarithm, arcsine, arccosine, arctangent and hyperbolic functions. GP begin with a initial population of random programs. Sometimes GP results in local solution (local maximum or minimum) which is not a globally optimal or even a good solution. High number of runs, may be useful to generate an acceptable result. In addition, increasing the initial population size, diversity of functions and in creasing mutation rate may solve this problem. but using Eureqa, it is not possible to edit the size of the population and individuals size, crossover and mutation rate.

4.3 Experiments

4.3.1 Case Study 1: House Price Prediction

To evaluate our proposed approach, we examined it on three different datasets. For our main case study, real-estate data, we tried to enhance house price estimation. As our proposed method seems applicable to other large datasets. To examine this capability, we applied the proposed approach on two other datasets; agricultural dataset and Facebook comments volume dataset. In this section is dedicated to the experimental results of applying our method on these three datasets.

4.3.1.1 House Price Dataset

Our first case study is house price estimation. As many variables influence price negotiations, high dimensionality of the house price datasets challenges the prediction of the final price of each home. For our study, we used house sales data for King County in the USA [12]. It contains 21,614 instances and 20 features such as price, number of bedrooms, number of bathrooms, house size, floors (number of floors), condition, grade, building and renovation date and location. The dataset details is provided in appendix. This is a multivariate dataset with both real and integer values with no sparsity. Before creating models, we study the data structure, first, we check the features correlation which is depicted in the Figure 4.1. There is not any strong correlation between features and price. Moreover, we can plot the price distribution.

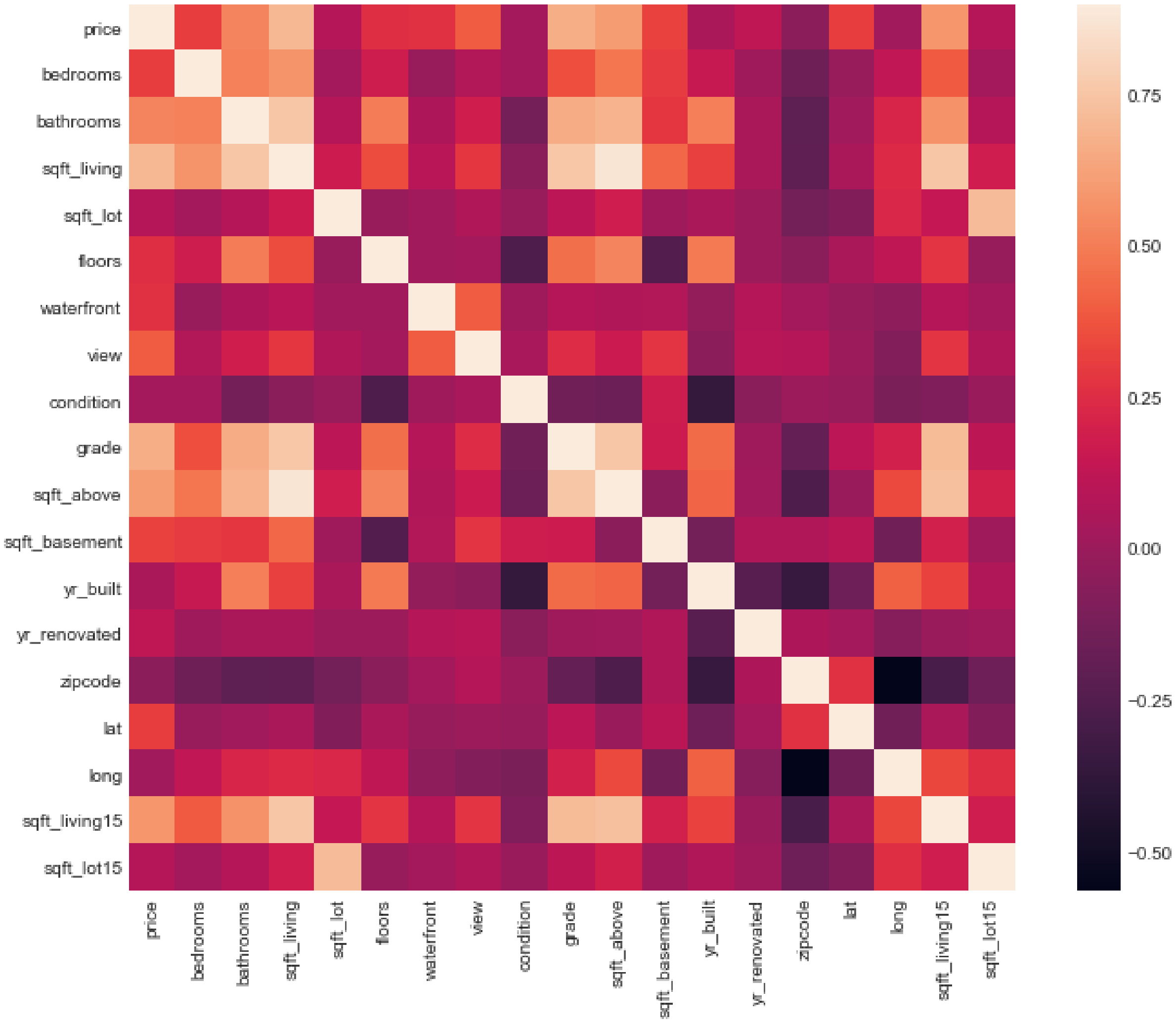


Figure 4.1: Correlation between the house price dataset features. This gives us an insight about the dataset features. If any feature is highly correlated or not correlated at all to all other features.

As illustrated in Figure 4.2, the mean price, μ, is 540088.14 Dollar and the data standard deviation, σ, is 367118.70 Dollar. Both the plot shape and the σ value, confirm that this dataset has considerable dispersion. Usually, a large dispersion, makes it hard to fit a

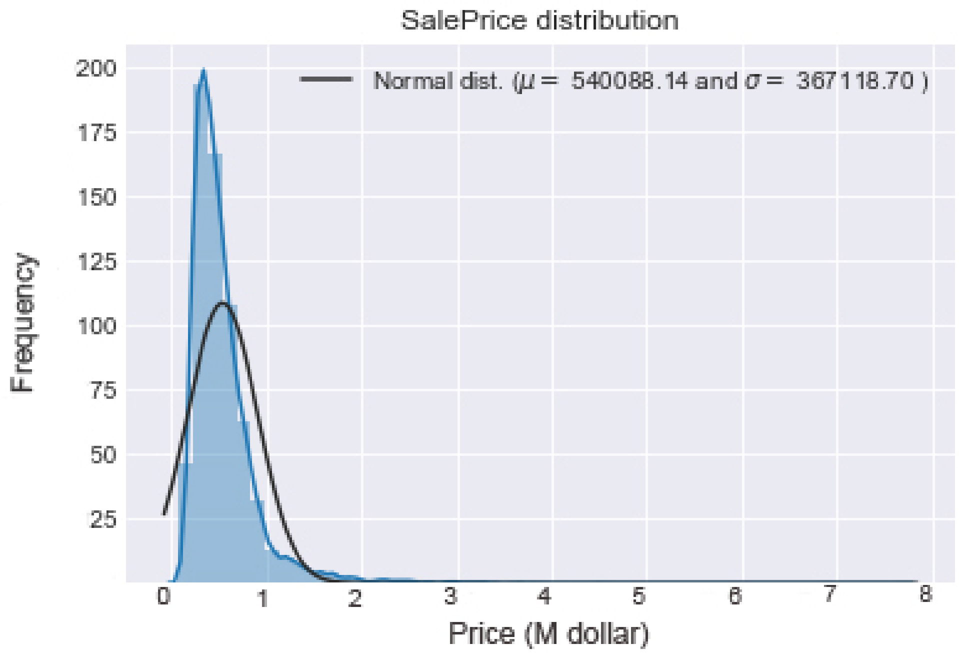


Figure 4.2: Comparison of price distribution in the house price dataset and the normal distribution. The prices are based on US dollar, μ is the mean price and σ is the data standard deviation. Both the plot shape and the σ value, confirm that this dataset has considerable dispersion and our data does not follow a normal distribution.

simple model on entire data. In addition, we can compare the price values in house price dataset probability plot with a normal distribution which has a linear form. The comparison is shown in the Figure 4.3. It is apparent that this dataset do not follow a linear or a polynomial curve. Only the right side of the plot which presents lower prices shows good fit to the linear normal probability. Further more, we can compare the logarithm value of the price as a variable with normal distribution. Figure 4.4 and the low value of σ which is only 0.53, reveals that the logarithm of price has better fit to the normal distribution. Moreover, Figure 4.5 the probability plot of the logarithm value of the price, has a better fit to the normal distribution. When we have a highly diverse dataset with a huge dispersion, that turns to be normal distribution with a small dispersion, we man conclude that there is

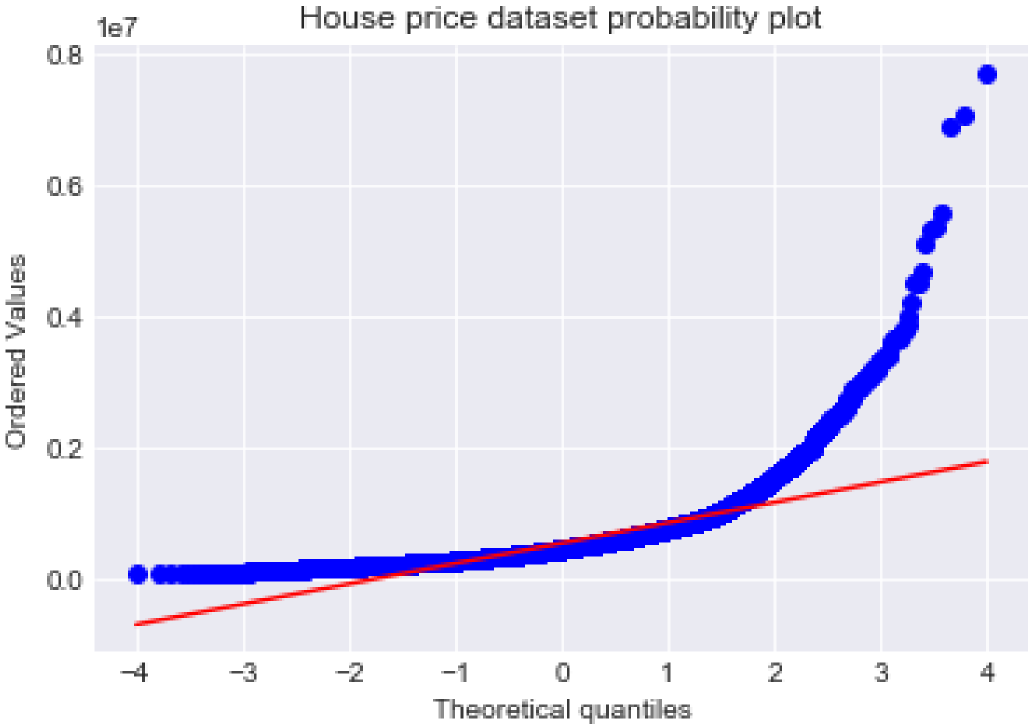


Figure 4.3: Comparison of house price dataset probability with a normal structure. The plot reveals that this dataset do not follow a linear or a polynomial curve. Only the right side of the plot which presents lower prices shows good fit to the linear normal probability.

a considerable diversity in the order of values. In our case the house price varies within four order of magnitudes. Therefore, our proposed method may work effectively for this dataset.

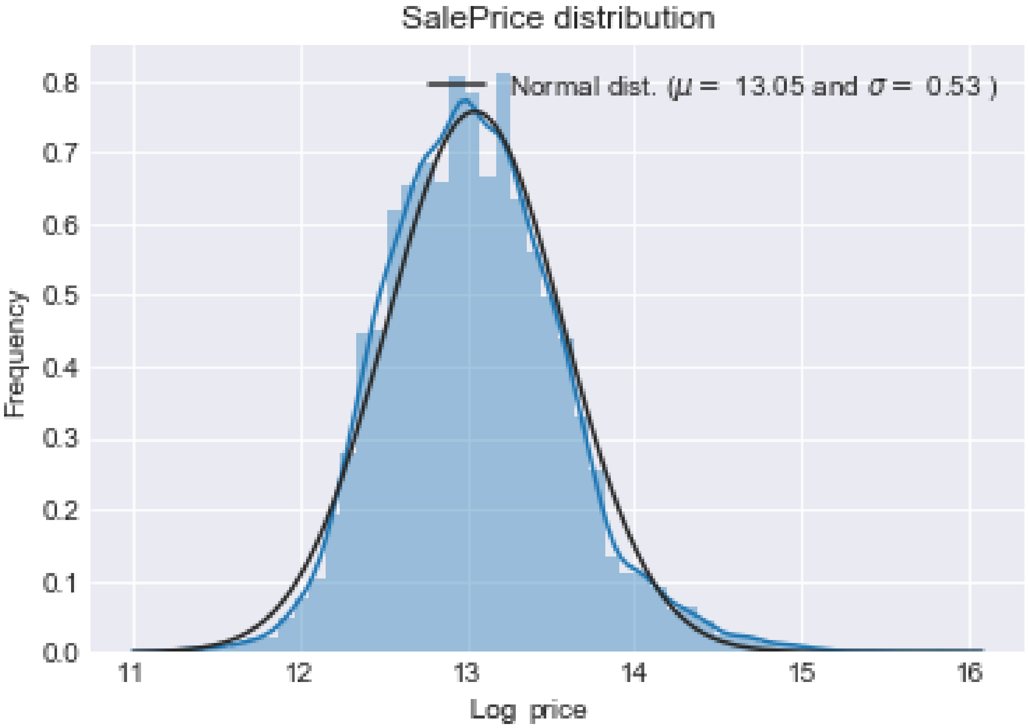


Figure 4.4: Comparison of logarithm of house price distribution in the house price dataset and the normal distribution. The prices are based on US dollar and horizontal axis is based on Log(price), μ is the mean of Log(price) and σ is the data standard deviation. The plot and the low value of σ, shows that the logarithm of price has better fit to the normal distribution and confirms that there is a considerable diversity in the order of values in our dataset.

4.3.1.2 House Price Prediction Experiments

In the first step of the estimation task, various models which are defined in the section 2.1.2, such as linear and polynomial regressions, SVR, neural network and genetic programming and some other methods, are applied on the whole dataset. We were interested to find out how the results may change if we create smaller sub-sets of our data without any condition. Therefore, we randomly divided the dataset to smaller groups with 2000 and 1200 members to investigate the effect of the data size on prediction accuracy for each model. As expected,

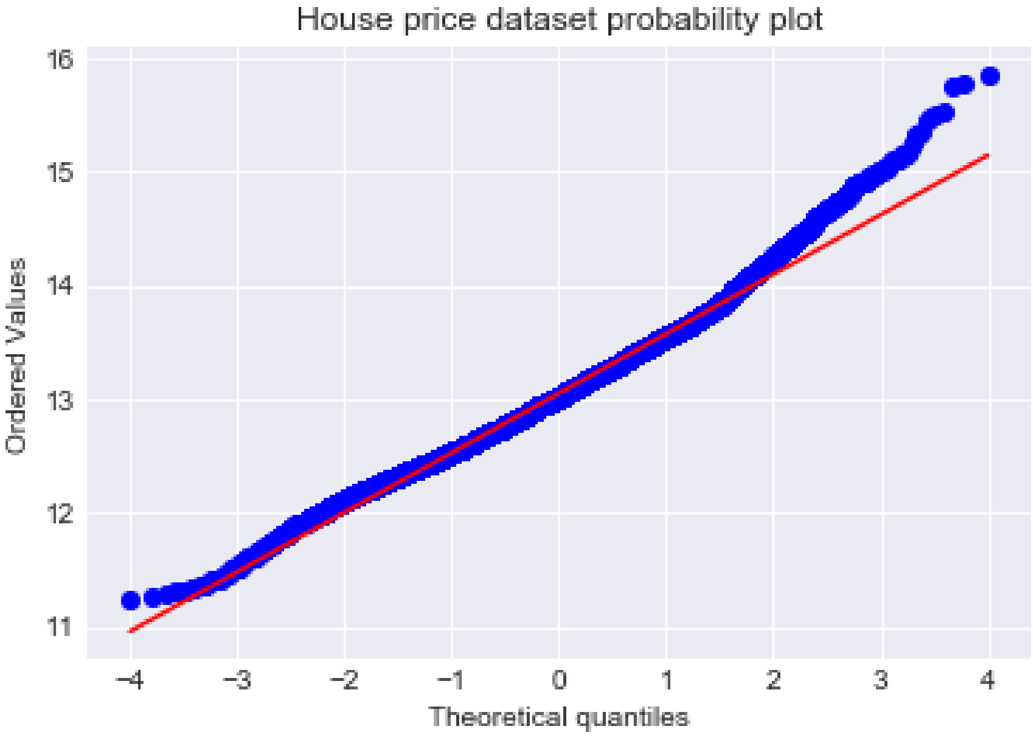


Figure 4.5: Comparison of house price dataset probability with logarithm of the price and a normal structure. The probability plot of the logarithm value of the price, has a better fit to the normal distribution. It shows that data has a highly diversity and a huge dispersion

the accuracy declines by reducing data size, but the decreasing accuracy rate and improving the error, varies widely for different models. Table 4.1 and Figure 4.8 display the relative errors for each prediction model. For calculating error, we consider the normalized relative error which is calculated by the Eq. 3.1 which is defined in chapter 3.

For symbolic regression, we create an equation using GP. Equation 4.1 is an example of one of the models that GP generate for the house price dataset.

$$price = a + (b + c * waterfront + d * grade + f * condition + g * sqftliving + h* \\ sin(i + j * lat) - sin(k + l * long) * sin(m + n * lat))^p \tag{4.1}$$

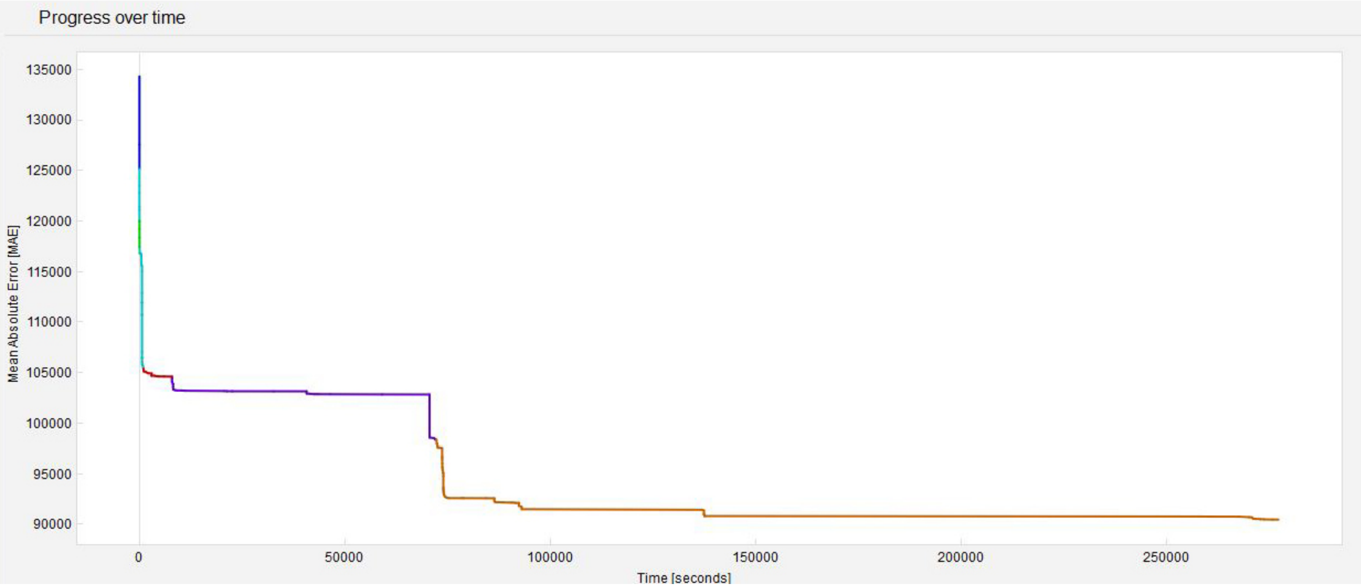


Figure 4.6: House price prediction error during generating the symbolic equation with GP.

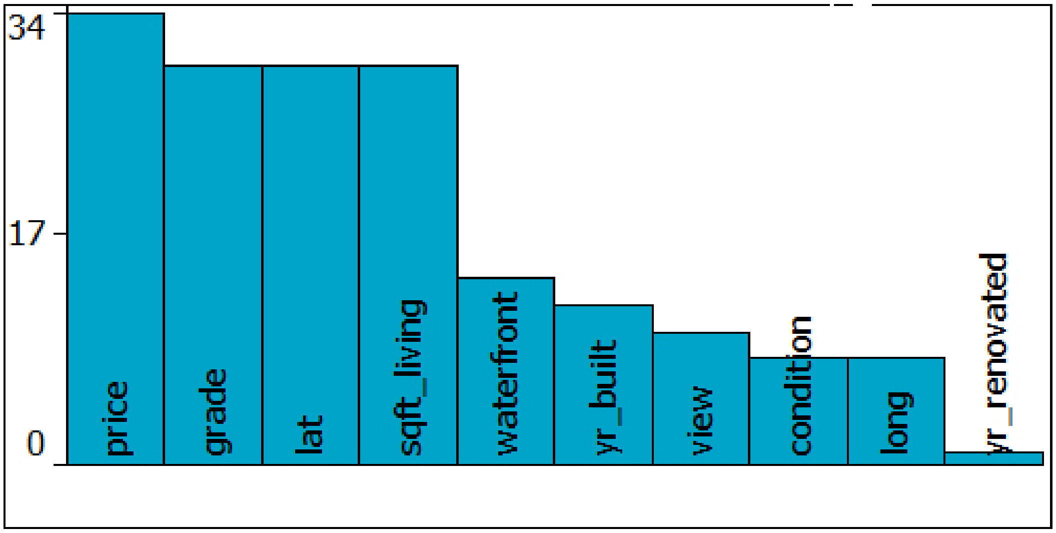


Figure 4.7: Number of models each variable appears in number of Occurrences of each variable (across all models generated before the final one).

(Which a=71287.47, b=8.655, c=4.1929, d=0.8507, f=0.390097, g=0.0010888, h=1.59789, i=5.17567, j=11.532, k=6.277, l=7.101, m=5.1757, n=11.532, p=4.3536)

The Figure 4.6 demonstrates the decreasing the prediction error during generating the

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**	**SVR**
Whole Data	**0.14**	0.175	0.22	0.246	0.248	0.251	0.343
G1: 2000-2500	**0.17**	0.19	0.261	0.25	0.244	0.28	0.375
G2: 1200-2000	0.25	**0.22**	0.267	0.254	0.243	0.287	0.388
G3: 1200	0.33	**0.27**	0.32	0.31	0.35	0.38	0.399

Table 4.1: Different methods relative error in house price prediction dataset and randomly created smaller sub-sets. Group 1 has 2000-2500 instances, Group 2 has 1200-2000 instances and Group 3 has 1200 instances.

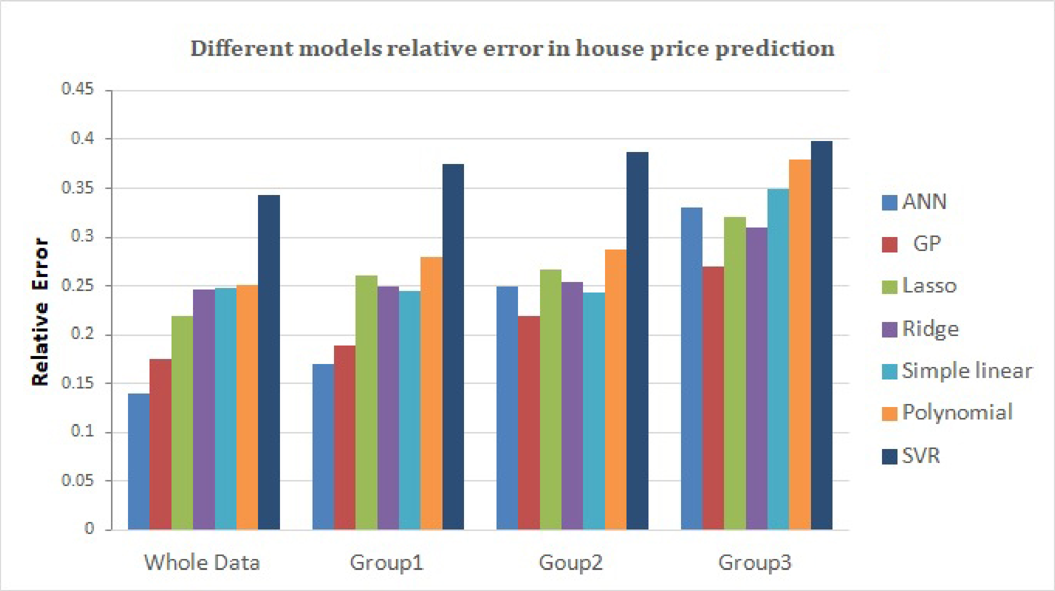


Figure 4.8: Different methods relative error in house price prediction dataset and randomly created smaller sub-sets. Group 1 has 2000-2500 instances, Group 2 has 1200-2000 instances and Group 3 has 1200 instances.

the symbolic equation with GP to the final one, Equation 4.1. Figure 4.7 shows the Variable Occurrences which is the number of GP models each variable appears in number

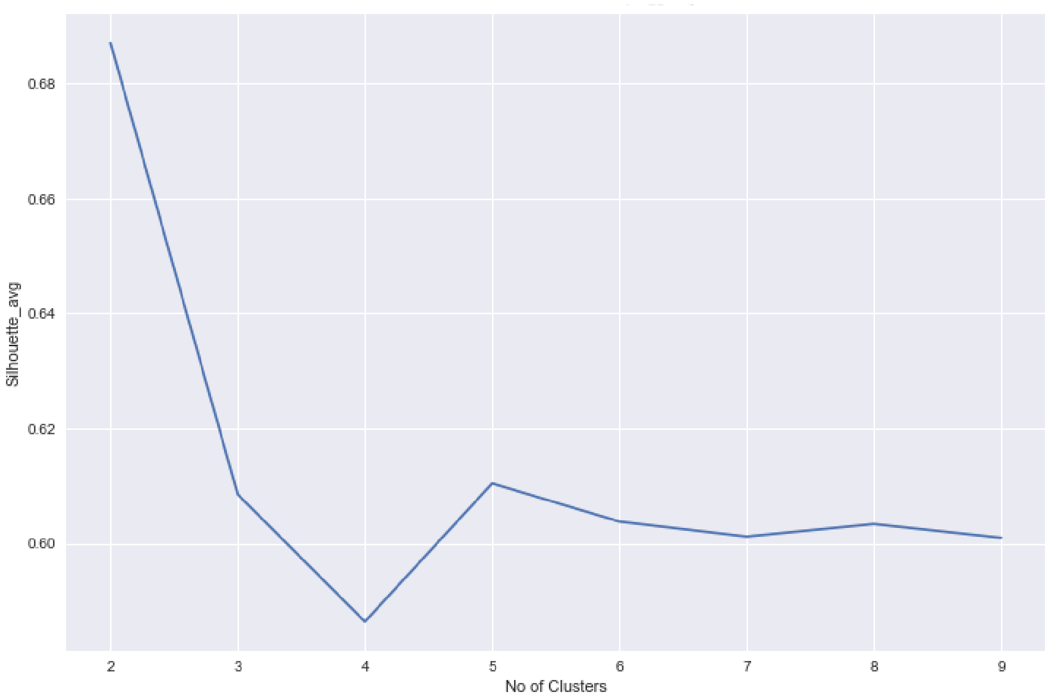


Figure 4.9: Silhouette Score for K-means clustering applied on house price dataset.

of Occurrences of each variable (across all models generated before the final one). This demonstrates the importance of each feature in symbolic regression.

K-means Clustering

The next step is Clustering data using K-means technique to know how the prediction may change if we have smaller datasets which its members in a cluster have similarity. The first step of applying K-means is determining the number of clusters. Using the Silhouette score which is defined in section 2.1.1, we can discover the most efficient number of clusters, K. Figure 4.9reveals that the best number of clusters in K-means for house price data is four. K-means created 4 sub-datasets: Group 1 has more than 2500 members, Group 2 has 2000-2500 instances, Group 3 has 1200-2000 instances and Group 4 has less than 1200 samples. Surprisingly, compared to other approaches, when the number of instances in a smaller datasets is very low (smaller than 1200 members), GP shows better perfor-

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**	**SVR**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251	0.343
G1: >2500 mem	**0.06**	0.14	0.2389	0.243	0.241	0.228	0.28
G2: 2000-2500	**0.13**	0.143	0.2389	0.2565	0.2544	0.2410	0.3109
G3: 1200-2000	**0.1445**	0.145	0.2365	0.2527	0.2517	0.271	0.3132
G4: <1200	0.23	**0.15**	0.2409	0.2588	0.2660	0.346	0.3282
Average	**0.09**	0.144	0.239	0.253	0.252	0.23	0.308
Overall				0.087			

Table 4.2: Different methods relative error in house price prediction dataset and smaller sub-sets created by **K-means** . Group 1 has more than 2500 members, Group 2 has 2000-2500 instances, Group 3 has 1200-2000 instances and Group 4 has 1200 samples. If we select the best model for each cluster, the total error for prediction after clustering, is **0.087**.

mance. Table 4.2 and Figure 4.10 compare the relative error for the employed models and the last row in Table 4.2, shows the average error for each model if we use clustering. In addition, if we select the best model for each group and then calculate the average, the total error for prediction through clustering, is **0.087**. K-means results in Table 4.2 indicate an important point that in all sub-datasets which include large number of instances, ANN has better performance especially in the most voluminous cluster, its accuracy is surprising. All prediction models except Lasso, could decrease error for this cluster. None of linear models have better average error comparing to the entire dataset predictions. The most surprising results belongs to GP. For all K-means clusters it has very good performance and even its average accuracy is not lower than The best model before clustering, ANN. K-means clustering significantly has improved GP's performance in prediction. In

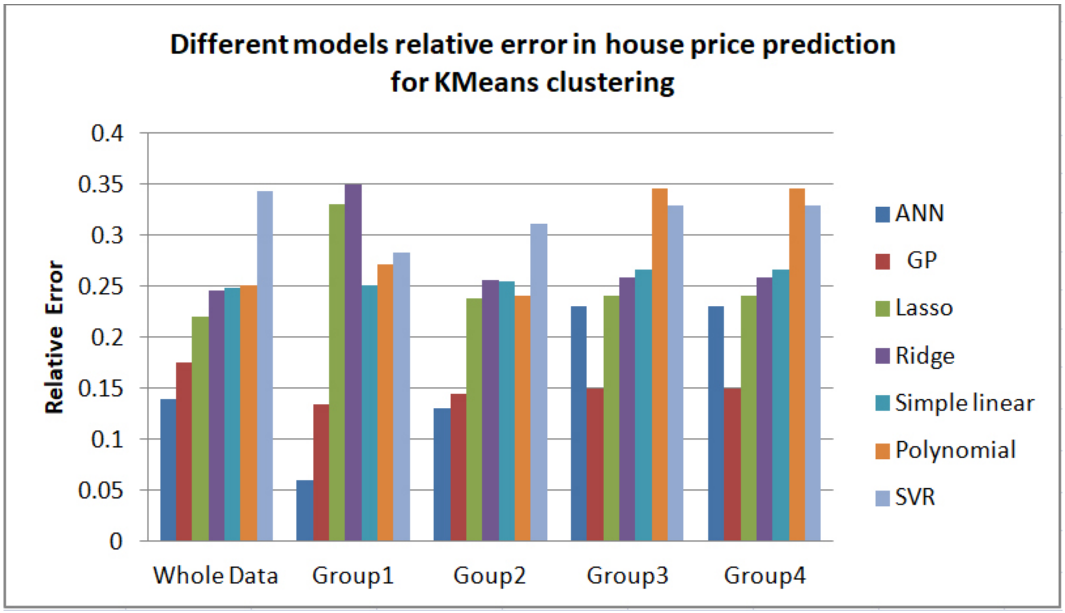


Figure 4.10: Different methods relative error in house price prediction dataset and small sub-sets created by **K-means** . Group 1 has more than 2500 members, Group 2 has 2000-2500 instances, Group 3 has 1200-2000 members and Group 4 has 1200 samples.

the smallest sub-dataset GP is the most accurate method. In addition, we wanted to find the way to improve the accuracy in prediction for created clusters with K-means. Therefore, we conduct another experiment. In this experiment, before clustering, first, we sort data to investigate if the results will improve? We sorted data based on the houses grades before applying K-means but we obtained the same results. Moreover, experiments with sorted data based on price, house size, and location, did not improve the prediction.

DBSCAN Clustering

DBSCAN is the second method which is employed. Comparing to K-means , the advantage of DBSCAN is that there is no need to initiate the number of clusters and this method automatically clusters data points based on their similarity (how close the samples are to-

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
G1 : 17000	**0.12**	0.172	0.218	0.223	0.236	0.24
G2 : 3000	**0.16**	0.168	0.23	0.244	0.25	0.28
G3 : <1000	0.19	**0.179**	0.26	0.27	0.28	0.31
Average	**0.13**	0.171	0.228	0.228	0.24	0.249
Overall			0.129			

Table 4.3: Different methods relative error in house price prediction dataset and smaller sub-sets created by **DBSCAN**. Group 1 includes approximately 17000 instances, Group 2 has around 3000 instances and group 3 has less than 1000 members. SVR error waslarger than 0.35 for all clusters. If we select the best model for each cluster, the total error for prediction after clustering, is **0.129**.

gether). DBSCAN created three clusters but unfortunately, most data points are assigned to one group. Group 1 includes approximately 17000 instances, Group 2 has around 3000 samples and group 3 has less than 1000 members. The Table 4.3 and the bar plot in Figure 4.11 compare the error of each prediction method for each created cluster. As SVR did not have acceptable performance with error larger than 35 percent for all clusters, its results have not been included in DBSCAN table and plots and the rest of experiments. If we select the best model for each cluster, the overall error is 12.9. Table 4.3 shows even DBSCAN clustering has improved prediction accuracy especially for ANN and GP, The average errors do not present considerable change. Similar to ANN and GP, Ridge, simple linear and polynomial models have very small improvement after applying DBSCAN. But it is surprising that again in small sub-dataset, GP has lower error comparing to other

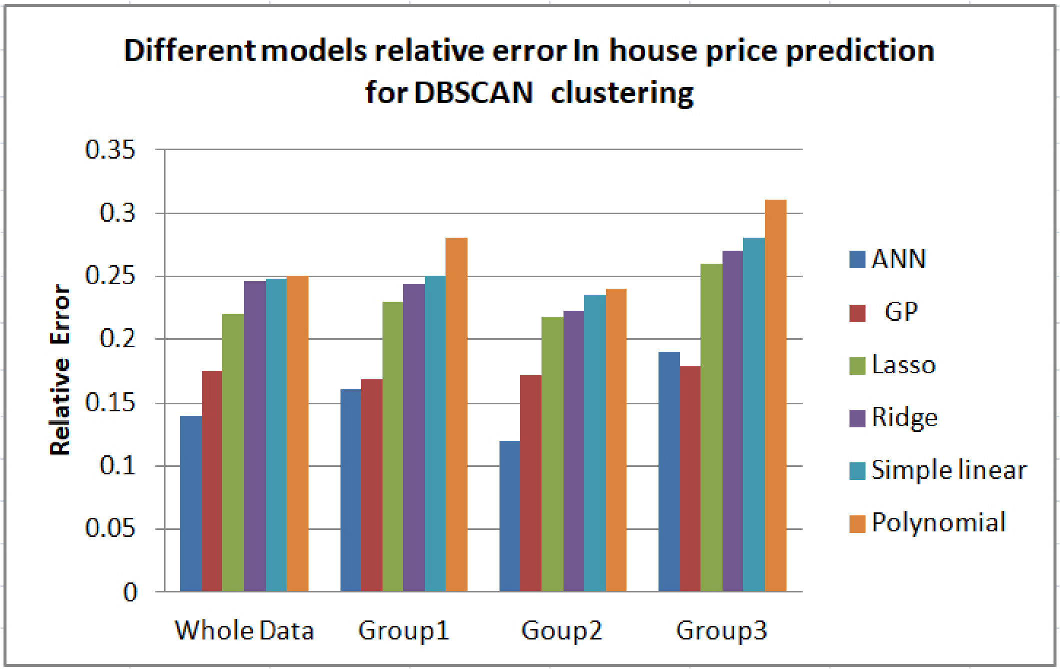


Figure 4.11: Different methods relative error in house price prediction dataset and smaller sub-sets created by **DBSCAN**. Group1 has more than approximately 17000 members, Group2 has around 3000 instances, Group 3 has less than 1000 instances.

models.

Hierarchical Agglomerative Clustering

Considering previous experiments with two different clustering techniques, it is apparent that the clustering method can considerably effect the results. Therefore, we Employed another well known clustering method; hierarchical agglomerative clustering or HAC. In this technique at the first stage, the most similar instances, being grouped in the first set of clusters in this stage, we have large number of clusters with low number of members. In next steps, the smaller clusters are combined based on their similarity and in subsequent steps, new clusters are created until to reach one large cluster that includes all data points. In section 2.1.1 we mentioned that we can use different linkage type: Ward, complete, average, and single linkage. We achieved better results using ward and average linkage.

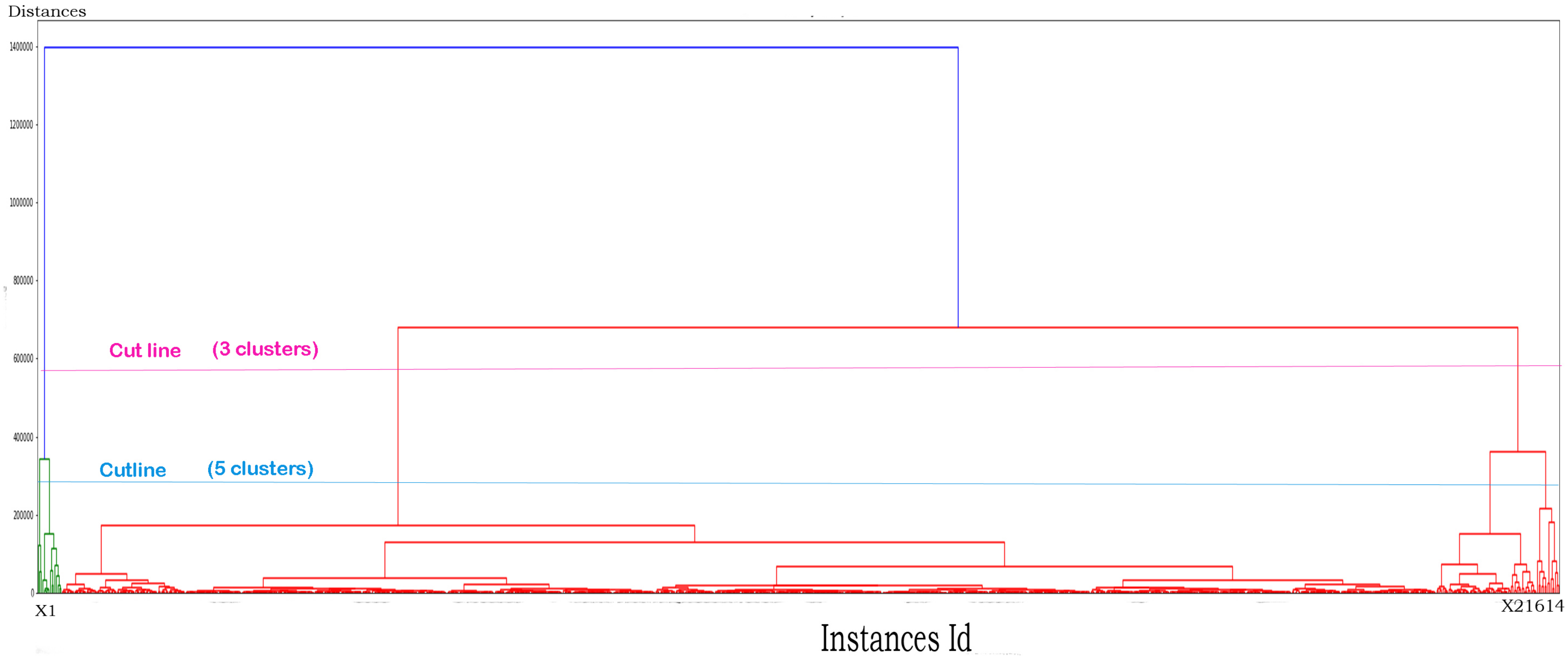


Figure 4.12: Hierarchical clustering dendrogram for house price dataset, created by average linkage method. The horizontal pink line shows a cut line which creates three clusters and using the blue line, generates five groups. It is obvious that there is no possibility to find a proper cutting line to create four clusters.

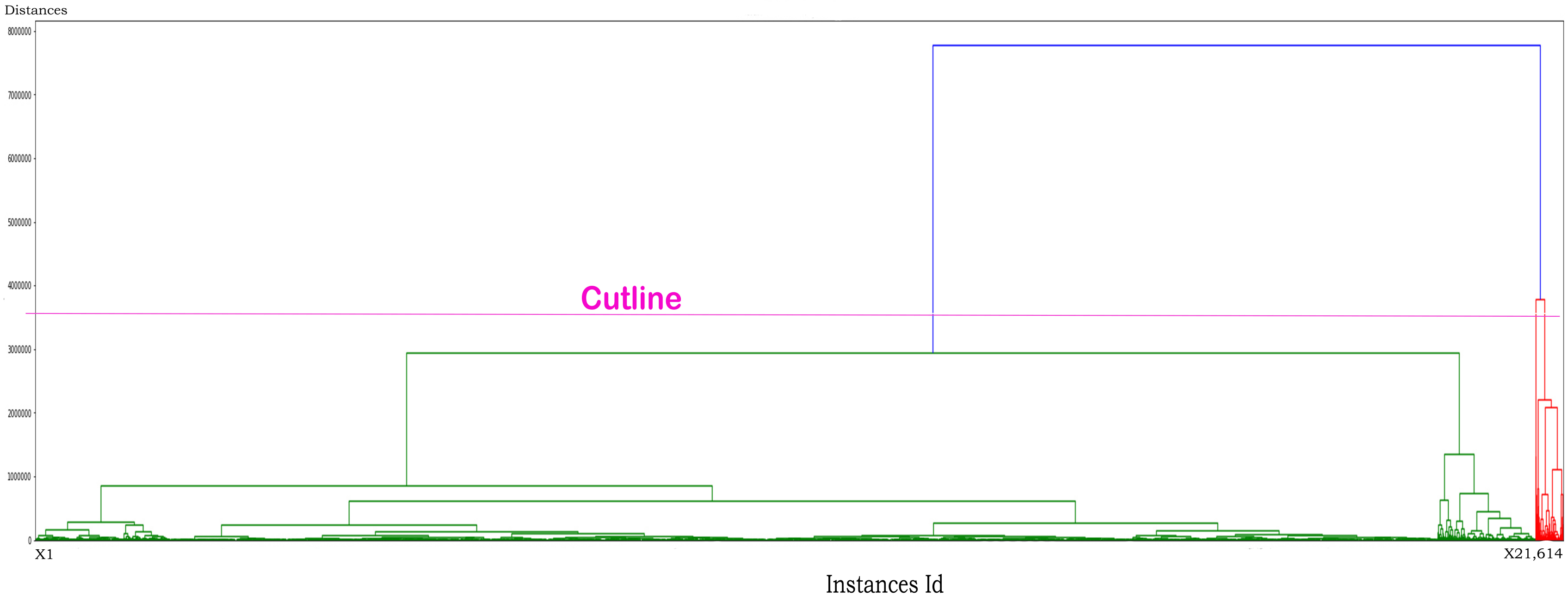


Figure 4.13: Hierarchical clustering dendrogram for house price dataset, created by ward linkage method. The horizontal pink line shows a cut line which creates three clusters.

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
G1: 12000	0.18	**0.17**	0.22	0.223	0.232	0.25
G2: 7000	0.22	**0.21**	0.25	0.241	0.242	0.264
G3: 2000	**0.22**	**0.22**	0.23	0.244	0.221	0.267
average	0.206	**0.20**	0.243	0.236	0.231	0.252
Overall			0.172			

Table 4.4: Different methods relative error in house price prediction dataset and smaller sub-sets created by **HAC**, **Average linkage**. Group 1 includes approximately 12000 instances, Group 2 has around 7000 instances and group 3 has approximately 2000 members. If we select the best model for each cluster, the total error for prediction after clustering, is **0.172**.

In others, more than 80 percent of the data points were assigned to only one cluster. The hierarchical average method is visualised in Figure 4.12 and Figure 4.13.

The advantage of HAC is that it provide us different options to choose the number of clusters easily by using its dendrogram (tree shape) chart. As depicted in Figure 4.12, by selecting different cutting lines, we can obtain various number of clusters with having insight and approximate estimation about the number of members in each cluster. But disadvantage of this method is that sometimes you can not find a proper cut line to obtain the desired number of clusters. Table 4.4 and Figure 4.14 show the results of house price prediction using HAC average linkage. The dendrograms in Figure 4.12 and Fig-

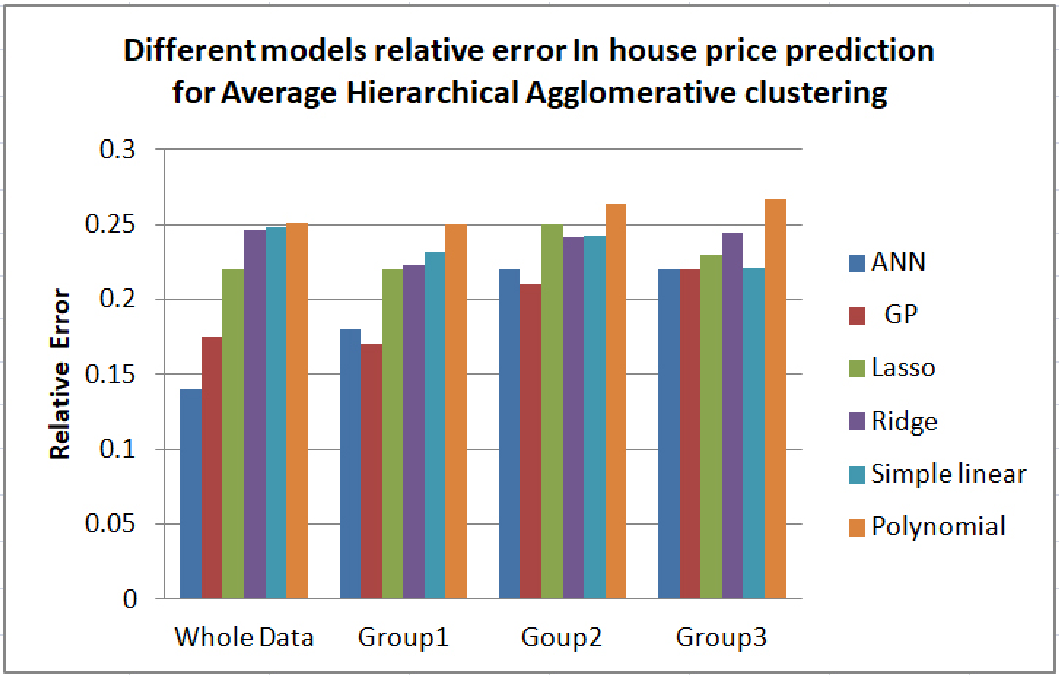


Figure 4.14: House price prediction error using average linkage in hierarchical clustering method. Group1 includes approximately 12000 instances, Group2 has around 7000 instances and group3 has approximately 2000 members.

ure 4.13 demonstrate different clustering structures for two different linkage methods. It is expected to gain different results for the ward linkage method which are displayed in Table 4.5 and Figure 4.15. Table 4.4 and 4.5 indicate that HAC could not improve the price prediction. For the largest sub-clusters with 12000 and 14000 members, as a big training data is available, we expected better performance for ANN, but the larger error comparing to the predictions for the whole data, affirm that HAC can not detect close data points for this dataset.

Human Knowledge Base Grouping

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
G1:14000	**0.18**	0.182	0.244	0.254	0.251	0.259
G2: 2000	0.24	**0.213**	0.250	0.259	0.268	0.284
G3:2000	0.24	0.242	0.252	0.254	**0.230**	0.278
average	0.22	**0.21**	0.246	0.256	0.245	0.262
Overall			0.178			

Table 4.5: Different methods relative error in house price prediction dataset and smaller sub-sets created by **HAC**, **ward linkage**. Group 1 includes approximately 14000 instances, Group 2 has around 2000 instances and group 3 has approximately 2000 members. If we select the best model for each cluster, the total error for prediction after clustering, is **0.178**.

Usually, when people want to do prediction or estimation, they categorize the items in their mind and compare every item with their knowledge about each category. This fact encouraged us to instead of using machine for clustering, ask some experts to help the machine in grouping (clustering) the dataset. In fact, human select the distinguishing features to split the data. Therefore, we asked some agents to select the most important features that they would use if they want to cluster the data points. In the next step, we grouped data based on the selected features. First, we grouped data based on the price and equal price intervals. Actually, in data processing, we can not use the target feature for clustering we only did this experiment to study the results. Therefore, Three sub-datasets created. The Groups based on price, have approximately 5000, 13000 and 3000 members respectively. The Groups based on grade, have approximately 6300, 9000 and 5700 members respectively. The results are displayed in the Table 4.6 and Figure 4.16. If we select the best

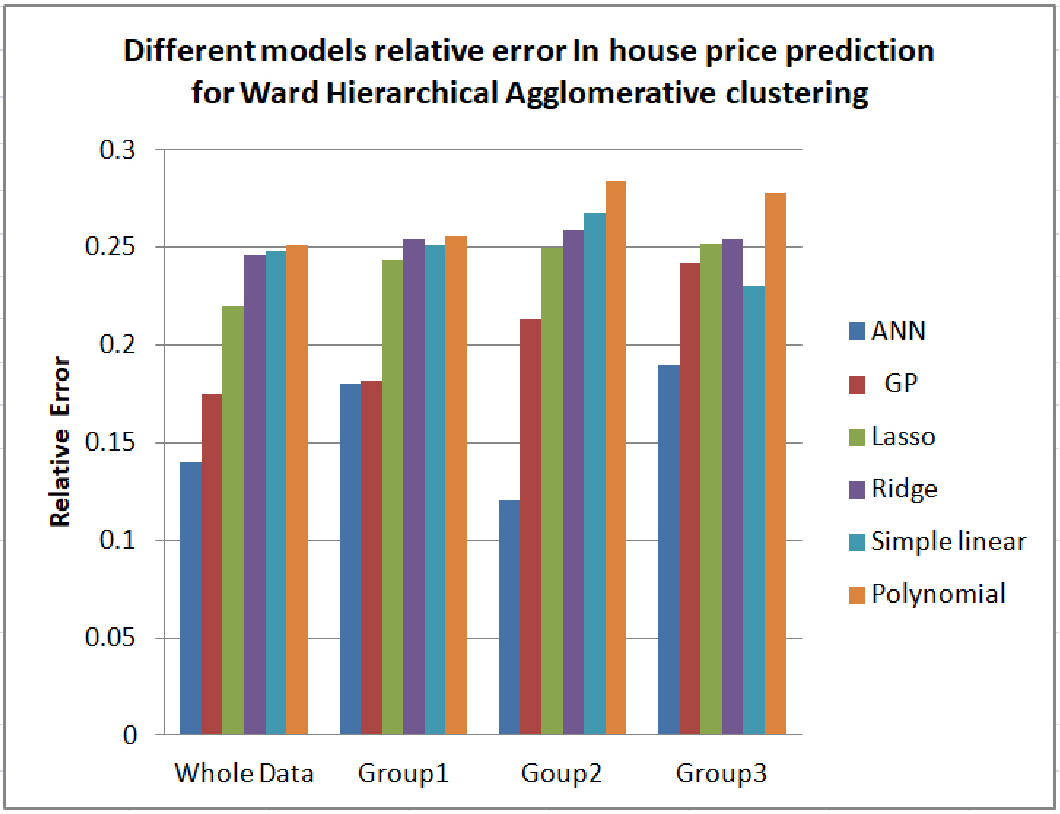


Figure 4.15: House price prediction error using ward linkage in hierarchical clustering method. Group1 includes approximately 14000 instances, Group2 has around 2000 instances and group3 has approximately 2000 members.

model for each group, the overall error for the price grouping is 0.6 and for grade base grouping is 0.1. In both predictions based on price and grade, for most of the models, We observe improvement only in the middle group, but in average we had better accuracy than prediction on the whole data. The the prediction accuracy for group 2 in grouping based on grade, is surprising especially for ANN and linear model. But none of the models in average can overcome the K-means average prediction accuracy.

Then, we grouped data again, based on the price and the house grade, but in the way to have approximately equal number of instances in each group. Therefore, three sub-datasets created with around 7000 members in each group. The results are displayed in the Ta-

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**	**SVR**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251	0.343
group1(prc)	**0.15**	0.18	0.28	0.27	0.26	0.29	0.38
Group2(prc)	**0.11**	0.16	0.14	0.16	**0.01**	0.21	0.27
Group3(prc)	**0.18**	**0.18**	0.28	0.27	0.27	0.30	0.36
Average(prc)	0.129	0.168	0.193	0.2	**0.07**	0.241	0.309
Overall (prc)	0.06						
group1(grd)	**0.18**	0.25	0.24	0.22	0.229	0.28	0.417
Group2(grd)	**0.012**	0.16	0.15	0.12	**0.01**	0.22	0.282
Group3(grd)	**0.19**	0.26	0.25	0.22	0.219	0.28	0.373
Average(grd)	**0.11**	0.21	0.22	0.20	0.128	0.254	0.347
Overall (grd)	0.1						

Table 4.6: House price prediction error, using grouping based on the selling Price (prc) and the house grade (grd) with equal price and grade value intervals. The groups based on price, have approximately 5000, 13000 and 3000 members respectively. The groups based on grade, have approximately 6300, 9000 and 5700 members respectively. If we select the best model for each group, the overall error for the price grouping is **0.06** and for grade base grouping is **0.1**.

ble 4.7 and Figure 4.17. If we select the best model for each group, the overall error for the price grouping is 0.08 and for grade base grouping is 0.13. Selecting the Equal number of samples for each group decreases the accuracy for the middle sub-group but in average, we observe a little decrease in error.

As mentioned above, we can not cluster data based on the house price because it is the

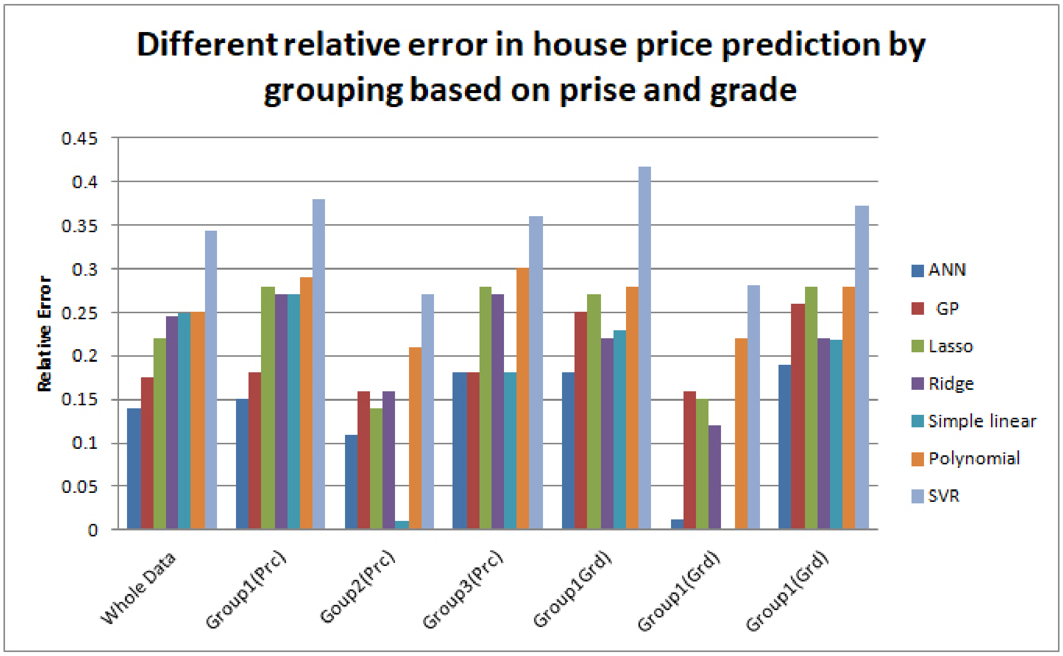


Figure 4.16: House price prediction error, using grouping based on the selling Price (prc) and the house grade (grd)with equal price and grade value intervals. The Groups based on price, have approximately 5000, 13000 and 3000 members respectively. The Groups based on grade, have approximately 6300, 9000 and 5700 members respectively.

feature that we need to predict. Therefore, we used another strategy. First, we split the train and test data, and remove the price column in the test data, then, we predict the price by applying prediction models on train data, now, we can use the predicted price for test data in grouping based on predicted price. The results of this scheme which are provided in Table 4.8 show that even groping based on predicted price can not improve accuracy for symbolic regression with GP, still it is practical for other models and we can obtain overall error equal to 0.09 which demonstrates very better accuracy than predictions on the whole dataset. This results confirm that even this scheme's accuracy is not better that K-means , but it is applicable to obtain better prediction comparing to the whole data. Some of the real-estate agents expressed that the location is the most important parameter. Therefore,

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**	**SVR**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251	0.343
group1(prc)	**0.12**	0.27	0.188	0.19	0.199	0.24	0.225
Group2(prc)	0.1	0.18	0.11	0.12	**0.009**	0.22	0.127
Group3(prc)	**0.12**	0.28	0.13	0.13	0.134	0.25	0.361
Average(prc)	**0.113**	0.24	0.142	14.6	0.114	0.23	0.237
Overall (prc)				0.08			
group1(grd)	**0.13**	0.32	0.27	0.22	0.229	0.26	0.401
Group2(grd)	0.021	**0.17**	0.20	0.20	0.08	0.22	0.328
Group3(grd)	**0.13**	0.31	0.27	0.22	0.19	0.26	0.346
Average(grd)	**0.093**	0.26	0.24	0.213	0.16	0.24	0.358
Overall (grd)				0.13			

Table 4.7: House price prediction error, using grouping based on the selling Price(prc) and the house grade(grd) with equal members in each group. All groups have around 7000 members. If we select the best model for each group, the overall error for the price grouping is **0.08** and for grade base grouping is **0.13**.

we grouped instances based on zipcode (region-based). Again, all prediction methods were examined for each created group the results are shown in Table 4.9 and Figure 4.18. For all sub-datasets, ANN has the best performance. Therefore, If we select the best model for each group, the overall error for the zipcode grouping is the same as the average error for ANN, 0.142. However, in some sub-groups we get lower error, but in total accuracy for all models we obtain approximately the same accuracy for the whole data.

As we could not gain improvement by grouping based on the zip codes, we studied the

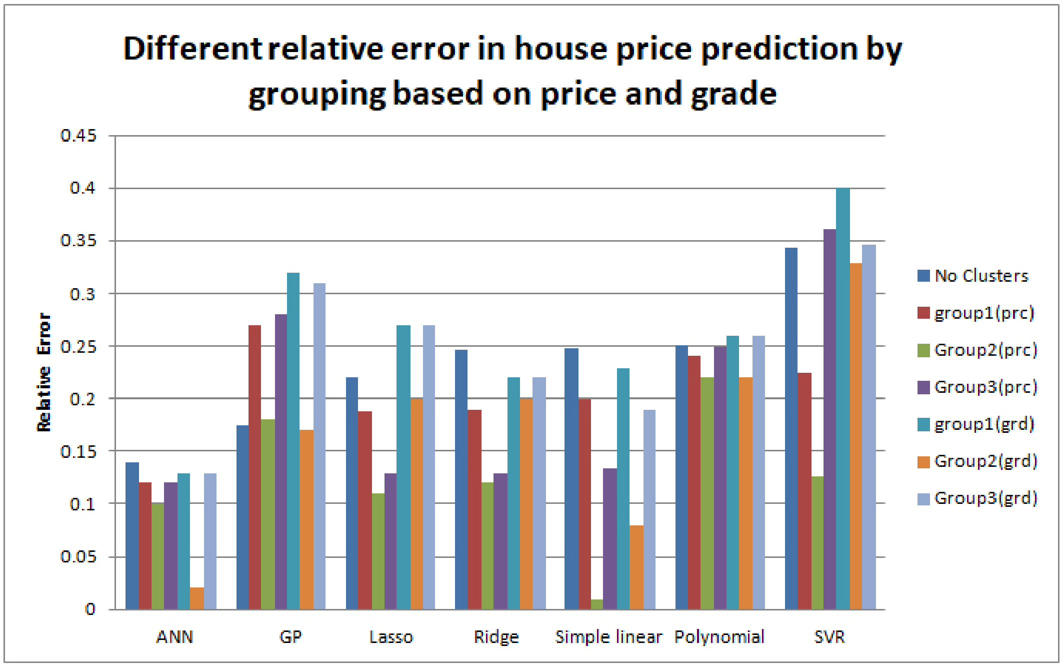


Figure 4.17: House price prediction error, using grouping based on the selling Price (prc) and the house grade (grd) with equal members in each group. All groups have around 7000 members.

same method of grouping and parameters with lower number of groups and this time we created only three groups. The improvement in prediction can be seen in the Table 4.10 and Figure 4.19. If we select the best method for each group, the total error is 0.12. This results apparently show that the larger sub-groups provide better accuracy, but still the results of grouping based on location are not more accurate than K-means and the house grade grouping.

We have studied that how grouping by the selling price can effect the results. Therefore, we grouped data based on both zip code and the price. Thus, we divided each group which is created based on the location, to other three groups based on the selling price. consequently, at the end we have nine sub-groups with approximately 2300 member in each. Again, we apply all prediction methods on each created sub-group. The results are pro-

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**	**SVR**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251	0.343
Group1(prc)	**0.14**	0.31	0.2	0.2	0.21	0.26	0.3
Group2(prc)	0.12	0.2	0.13	0.14	**0.01**	0.22	0.15
Group3(prc)	**0.13**	0.31	0.15	0.15	0.14	0.26	0.4
Average	0.13	0.22	0.16	0.163	**0.12**	0.246	0.28
Overall				0.09			

Table 4.8: House price prediction error, using grouping based on the predicted selling Price(prc) with equal members in each group. All groups have around 7000 members. If we select the best model for each group, the overall error is **0.09**.

vided in the Table 4.11 and Figure 4.20. Moreover, if we select the best model for each group, the overall error is 0.123. As demonstrated in Table 4.1, when the number of instances decrease, the error increases dramatically. Accordingly, we also created sub-groups based on both zip code and price, but we only created six sub-groups with larger number of members in each sub-group. For this purpose, first, we grouped the whole data to three groups based on the location, then, we divided each group to two sub-groups based on the selling price. We were not surprised when the results changed by increasing the number of samples. The outcome is provided in Table 4.12 and Figure 4.21 and for this experiment we reached to the total error equal to 0.087 which is considerable lower than the total error when we have nine groups. For all prediction methods we can see an improvement in prediction, but the change is not significant for the polynomial regression in both six and nine groups. The prediction with neural network is very sensitive to the number of instances in each sub-group and we can see when we have more samples in each sub-group for the same

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
Group 1	**0.13**	0.16	0.210	0.22	0.22	0.24
Group 2	**0.13**	0.15	0.22	0.22	0.23	0.235
Group 3	**0.14**	0.16	0.218	0.23	0.24	0.256
Group 4	**0.18**	0.17	0.232	0.25	0.254	0.26
Group 5	**0.17**	0.17	0.23	0.248	0.25	0.278
Group 6	**0.15**	0.16	0.231	0.25	0.25	0.255
Average	**0.142**	0.161	0.22	0.24	0.243	0.259
Overall			0.142			

Table 4.9: House price prediction error for six sub-datasets, using grouping based on the location (Zip code). If we select the best model for each group, the overall error is **0.142**.

grouping method, the error drops from 0.11 for nine groups to 0.9 which belongs to the six groups. The most surprising result is about linear predictions that they have the largest decrease in error when we split datasets only based on important features. It means that for house price dataset, when we have sets of smaller groups, which are split up by their two important variables; price and location, approximately, we can consider them, especially the middle ones on grouping, as linear systems. Both zipcode-price grouping for nine and six sub-groups, were based on having approximately equal number of samples in created groups, If we group data, for equal value intervals for variables for example, price, in some groups we will have a few instances, sometimes less than 500 members. This has negative effect on the accuracy. Consequently, for all grouping experiments based on two or more variables, we split up data in the way to have approximately equal number of samples in

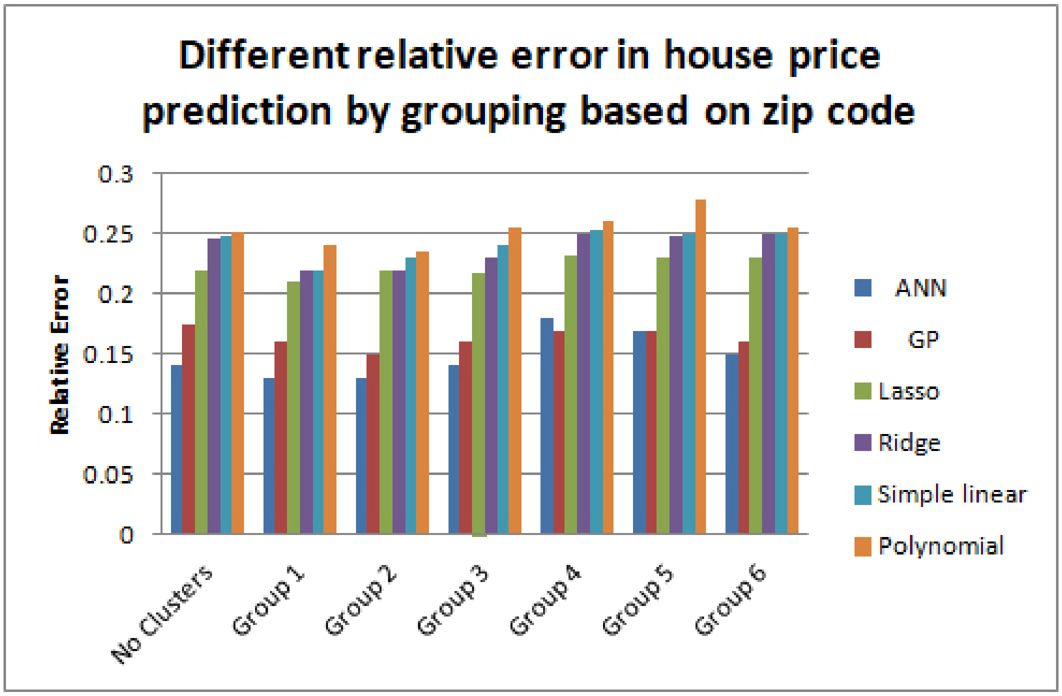


Figure 4.18: House price prediction error for six sub-datasets, using grouping based on the Location (zip code).

each sub-group.

Considering the real-estate experts knowledge, the location of the house is not the only important parameter in house price prediction. Another important variable is the size of the house which is shown as the Square feet (Sqrfeet) in the dataset. Therefore, the same as previous experiment, we group house price dataset based on both location and the house size. First, we divide the whole data to three groups considering their zip code. then, split each group to other three sub-groups regarding their size and again, we study the different prediction models performance for each sub-group. Table 4.13 and Figure 4.22 compare the results this process decreases the total error to 0.086. If we only create six sub-groups the same as what we did in previous experiment, it provides a little change in the results

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
Group1	**0.12**	0.15	0.210	0.22	0.22	0.245
Group2	**0.11**	0.14	0.22	0.22	0.23	0.24
Group3	**0.14**	0.17	0.218	0.23	0.24	0.249
Average	**0.12**	0.154	0.217	0.221	0.225	0.243
Overall	0.12					

Table 4.10: House price prediction error for three sub-datasets, using grouping based on the location (Zip code) with three groups. If we select the best model for each group, the overall error is **0.12**.

and total error which decreases to 0.085. Moreover, we were Curious to investigate how the accuracy may change if we add another important feature and level to the grouping based on features. Therefore, in addition to the two previous variables, location and the house size, we also consider the number of bedrooms. Accordingly, First we create nine groups similar to the previous experiment, then we divide each group to two other sub-groups based on the number of bedrooms. Consequently, we obtain 18 sub-groups with approximately 1100 instances in each. But we did not achieve very lower error. Some of prediction errors are provided in Table 4.14. In this experiment, in average, all prediction models had better performance except neural network and polynomial. It seems that this two models accuracy is highly dependent on the data size (number of instances). Surprisingly, GP's performance is improved considerably but unfortunately, the overall error increased to 0.125. Therefore, we decided to repeat the experiment with lower number of sub-groups which have more instances each. Accordingly, first, we grouped the whole data to three clusters based on the

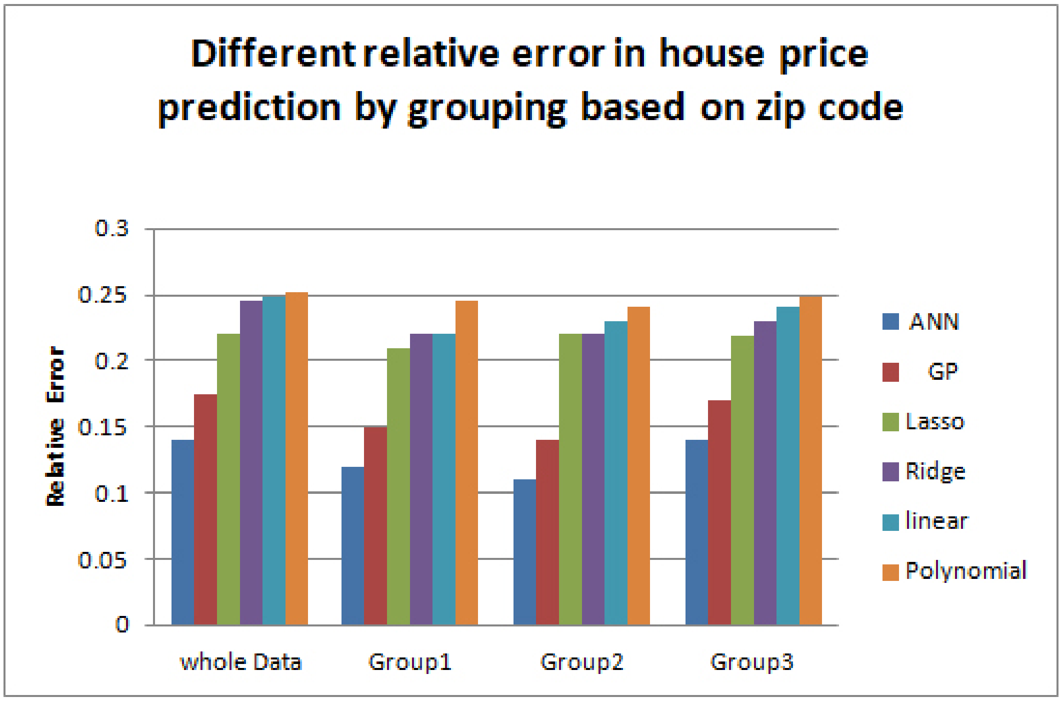


Figure 4.19: House price prediction error for three sub-datasets, using grouping based on the Location (zip code). Comparing to Figure 4.18, in this experiment we have less number of groups and more instanses in each group

location. Then, each group is divided to other two groups considering the house size and at last, each group is split to two sub-groups based on the number of bedrooms. Eventually, we have 12 sub-datasets and the results of different models for each of them is provided in Table 4.15 and Figure 4.23. Comparing this experiment with 18 groups most predictions and the overall accuracy is improved. On the other hand, it has larger error comparing to the experiment with the experiments that we created six and nine sub-groups, based on the size and the location. Again, it can be interpreted that not only neural network and polynomial methods are highly correlated to the data size, but also the linear models can be damaged by decreasing the number of instances. Even though, comparing the grouping to six and nine groups reveal that in smaller groups linear models can be more successful,

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
Group1	0.12	0.14	0.12	**0.114**	0.115	0.23
Group2	0.11	0.13	0.09	**0.08**	0.104	0.20
Group3	0.13	0.15	**0.1**	0.11	0.113	0.22
Group4	0.11	0.14	**0.1**	0.1	0.106	0.23
Group5	**0.08**	0.11	0.09	**0.08**	0.1	0.19
Group6	0.11	0.12	0.11	**0.1**	0.104	0.20
Group7	0.11	0.13	**0.1**	0.11	0.11	0.23
Group8	0.1	0.1	**0.07**	0.08	0.09	0.22
Group9	0.13	0.16	**0.1**	0.11	0.1	0.24
average	0.11	0.133	0.097	**0.096**	0.104	0.217
Overall			0.123			

Table 4.11: House price prediction error for nine groups, using grouping based on both location (Zip code) and price. If we select the best model for each group , the overall error is **0.123**.

but too much minifying the sub-groups can harm the linear models as well. In addition, the most damaged model is the polynomial one. Figure 4.23 demonstrates that not only polynomial model does not have better performance in most sub-groups but it also have Worse accuracy in some groups.

We could improve the predictions by splitting data based on important features, but still this method is not as accurate as using the K-means clustering. We wonder if there is any way to improve the accuracy in grouping based on experts knowledge. When we split data

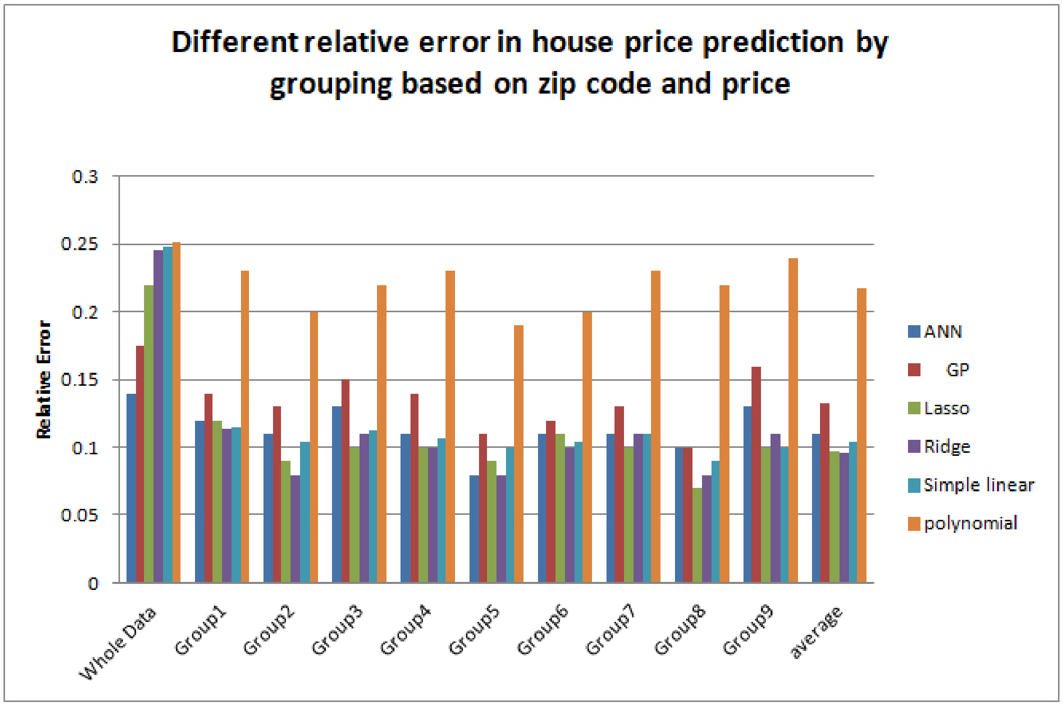


Figure 4.20: House price prediction error for nine sub-datasets, using grouping based on the Location (zip code) and the selling price.

based on important features to create sub-datasets, we cut the data sharply from boarder values. But the data points near the borders, can be member of the groups in both sides of the boarder value. According to this fact, if we assign the instances which are located near the boarders in both groups in two sides of the boarder values, it may improves the predictions. Therefore, we added the 10 percent of the data points in each group which are beside the borders to the other group on the other side. First, we applied this method on grouping based on location and price which is discussed in the Table 4.11. The prediction results for new grouping is provided in the Table 4.16. As we expected, comparing the average errors, there are improvement in all predictions even though it is not significant for all of them. Moreover, overall error is decreased from 0.123 to 0.089.

In order to evaluate adding boarder data points scheme, we repeat the experiment for group-

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
Group1	0.11	0.13	**0.1**	0.11	0.11	0.23
Group2	0.08	0.9	0.08	**0.07**	0.1	0.21
Group3	0.11	0.13	0.1	**0.09**	**0.09**	0.23
Group4	0.11	0.13	**0.1**	**0.1**	**0.1**	0.24
Group5	0.08	0.9	**0.07**	**0.07**	**0.07**	0.24
Group6	0.09	0.11	**0.08**	0.1	0.1	0.20
Average	0.096	0.113	**0.089**	0.09	0.095	0.23
Overall			0.087			

Table 4.12: House price prediction error for six groups, using grouping based on both location (zip code) and price. If we select the best model for each group, the overall error is **0.087**.

ing based on location and price which is demonstrated in the Table 4.13. The results are presented in the Table 4.17. Using this approach, improve the total accuracy and decrease the error from 0.94 to 0.086. Comparing the average errors for each model in the Table 4.17, all predictions had a little improvement.

Clusters as a New Feature

As we had success in applying prediction models in clustered data, it was motivating to investigate if we add clusters as a feature to the dataset, can we improve the prediction? Therefore, first, we cluster samples in house price dataset and assign each instances in dataset to a cluster. Then, we append another column for clusters to the dataset. This new feature presents the cluster that every data point belongs to. As it can be seen in Table 4.18

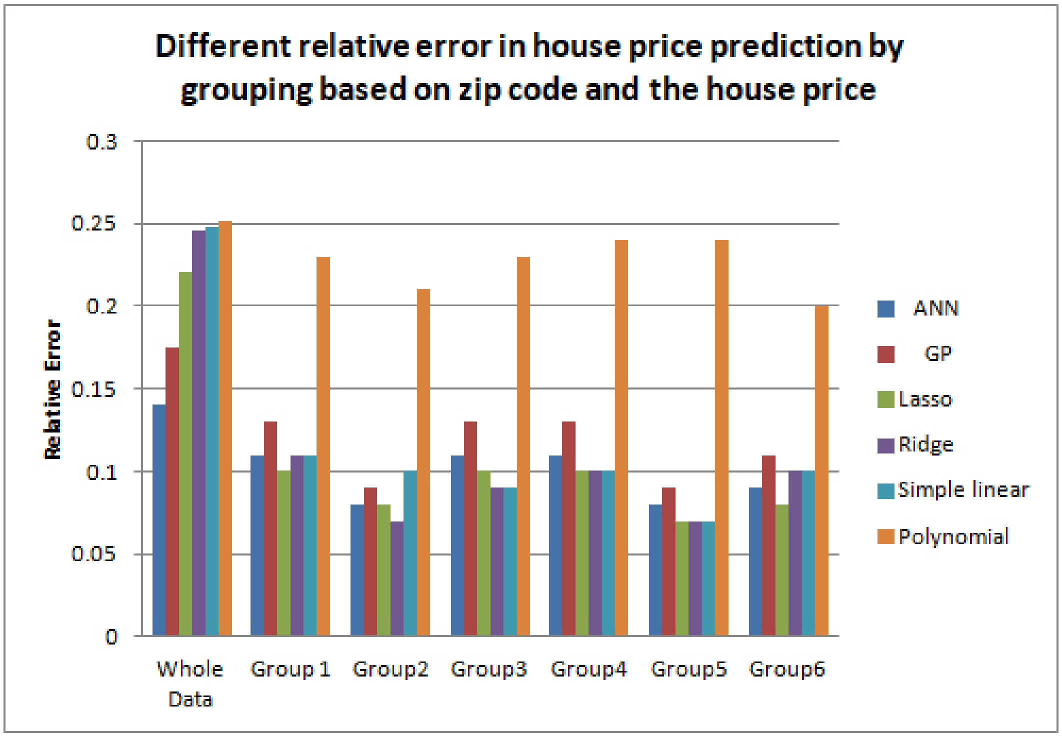


Figure 4.21: House price prediction error for six sub-datasets, using grouping based on the Location (zip code) and the selling price.

we could not obtain considerable better results.

4.3.1.3 House Price Prediction Results

The summary of results is illustrated in the Table 4.19 and Figure 4.24 for the first dataset, house price prediction. The best predictor before clustering is neural network with the error equal to 0.14. HAC could not improve accuracy for different linkages comparing to applying prediction models to the entire data. DBSCAN improved accuracy a little for ANN that the error decreased from 0.14 to 0.13 and overall to 0.129 but it is not considerable enhancement. Even we examined different values for parameters ϵ and minPts, we could not find proper clusters. When minPts is 2, we have many clusters with only a few instances when we increase 2 to 3, DBSCAN assigns most samples in one big group that can not

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
Group1	**0.12**	0.13	0.13	0.13	0.12	0.22
Group2	**0.08**	0.13	0.11	0.1	0.11	0.23
Group3	**0.09**	0.12	0.12	0.11	0.11	0.24
Group4	**0.08**	0.13	0.09	0.09	0.1	0.26
Group5	**0.11**	0.14	**0.11**	0.12	0.12	0.25
Group6	**0.08**	0.14	0.1	0.1	0.1	0.25
Group7	**0.08**	0.13	0.11	0.1	0.1	0.23
Group8	0.13	0.13	0.12	**0.11**	0.12	0.23
Group9	0.12	0.12	**0.1**	**0.1**	**0.1**	0.25
Average	**0.098**	0.13	0.11	0.106	0.108	0.24
Overall	0.094					

Table 4.13: House price prediction error for nine groups, using grouping based on both location (zip code) and the house size. If we select the best model for each group, the overall error is **0.094**.

help us to evaluate the proposed method because the big created cluster is approximately equal to the whole dataset. But K-means has acceptable performance with total error equal to 0.087. It shows that K-means can partition data points in House price dataset effectively. Tables 4.1to 4.13 and Figures 4.8 to 4.22, show that ANN is the best predictor when we have sufficient number of samples. But when the number of instances is small, for example in the clusters/groups with less than 1200 instances in our experiments, ANN fails. For all prediction models, after clustering, especially for small sub-datasets, the accuracy de-

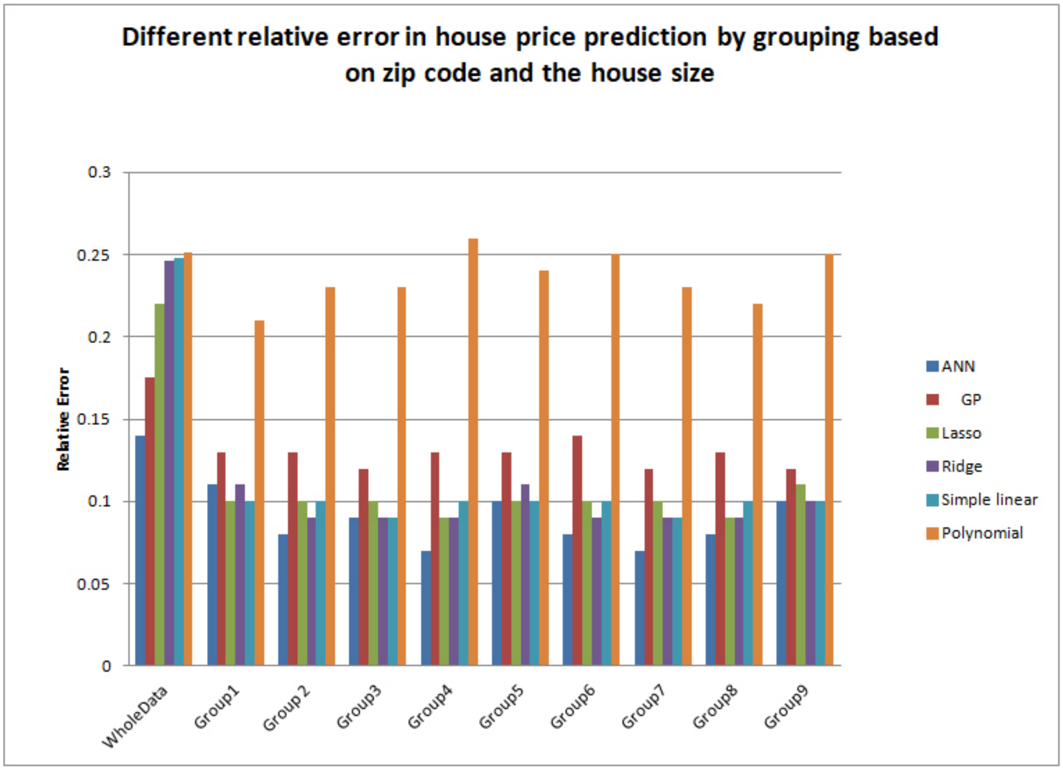


Figure 4.22: House price prediction error for nine sub-datasets, using grouping based on the Location (zip code) and house size.

creases considerably but in this cases, GP can perform well comparing to other models. In addition, Ridge, Lasso, and simple linear models have approximately similar accuracy for most experiments and it is astonishing that linear simple model is less sensitive to the number of training data. In addition, the results of grouping data based on the human knowledge and comparing their average errors in Table 4.19 and Figure 4.24, especially grouping E, grouping six groups based on location and price and grouping J, creating 9 groups based on location and the house size with adding boarder members to the groups, indicate that this method can compete the machine learning techniques that we employed for clustering. SVR cannot compete with other models regarding its large relative error. However, it is surprising that when we create smaller datasets using K-means, SVR has

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
Group1	0.15	0.121	0.17	0.17	0.18	0.26
Group2	0.16	0.13	0.17	0.18	0.19	0.22
...	...	...	...	...	...	...
Group18	0.17	0.135	0.16	0.17	0.175	0.25
Average	0.16	0.125	0.17	0.178	0.18	0.26
Overall			0.125			

Table 4.14: House price prediction error for 18 groups, using grouping based on three features; location (zip code), the house size and number of bedrooms. If we select the best model for each group, the overall error is **0.0.125**.

better accuracy for the sub-groups comparing to regression on the whole data, however, its error is larger than other models. The GP also has similar behavior and it presents a little better performance in prediction when samples have similarity.

4.3.2 Case Study 2: Water Evaporation Prediction

We examined our proposed method by applying on the house price dataset. It was incentive that if we use another dataset with smaller size and lower number of instances how the results will change. Therefore, we used an agricultural dataset to evaluate our method for low size datasets.

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
Group1	**0.12**	0.14	0.14	0.14	0.13	0.26
Group2	0.13	0.13	**0.09**	**0.09**	0.1	0.24
Group3	0.12	0.15	**0.1**	0.12	0.12	0.23
Group4	**0.09**	0.15	0.1	0.1	0.1	0.24
Group5	**0.07**	0.14	0.09	0.09	0.09	0.253
Group6	**0.09**	0.16	0.11	0.1	0.12	0.25
Group7	**0.1**	0.14	**0.1**	0.11	0.12	0.25
Group8	0.11	0.15	**0.09**	0.1	0.1	0.233
Group9	**0.11**	0.14	**0.11**	**0.11**	0.12	0.267
Group10	**0.1**	0.15	**0.1**	**0.1**	0.11	0.243
Group11	**0.09**	0.15	0.1	0.1	0.1	0.226
Group12	**0.09**	0.16	0.11	0.11	0.12	0.26
Average	**0.101**	0.146	0.11	0.112	0.11	0.257
Overall	0.095					

Table 4.15: House price prediction error for 12 groups, using grouping based on three features; location (zip code), the house size and number of bedrooms. If we select the best model for each group, the overall error is **0.095**.

4.3.2.1 Water Evaporation Dataset

According to the results of the first and the second steps of previous experiments, clustering and applying prediction models, we observed that the number of instances clearly can affect regression accuracy. Therefore, we decided to apply the same models on a dataset

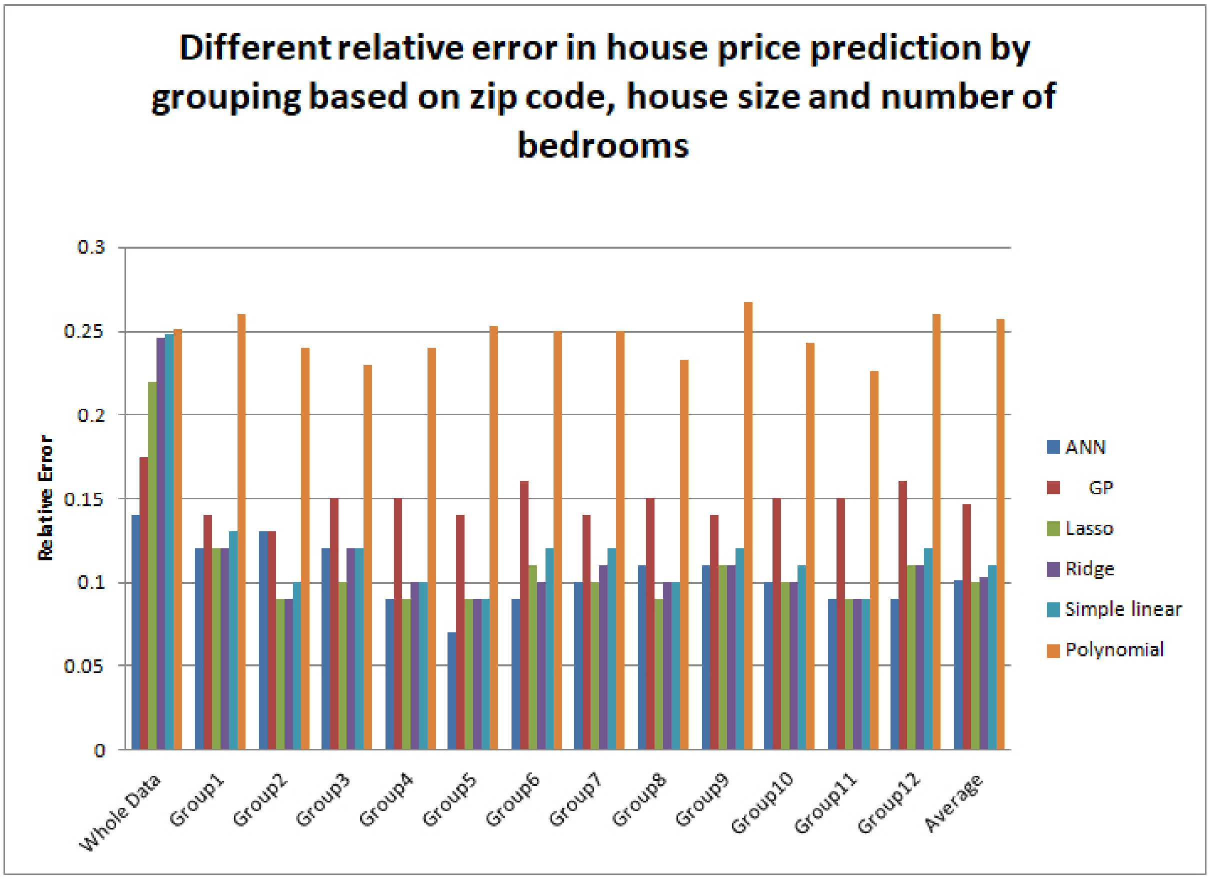


Figure 4.23: House price prediction error for 12 sub-datasets, using grouping based on the Location, house size and number of bedrooms.

with smaller size which has both a smaller number of samples and variables. The second dataset describes agricultural samples, with 10 features and 7549 samples. This is multivariate dataset which includes categorical, integer, and real variables with no sparsity [13] and [83].

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
Group1	**0.11**	0.15	0.12	**0.11**	0.12	0.22
Group2	0.1	0.13	0.1	**0.09**	0.11	0.23
Group3	**0.1**	0.12	**0.1**	**0.1**	0.1	0.24
Group4	**0.08**	0.13	0.09	0.1	0.11	0.26
Group5	**0.08**	0.12	0.09	0.09	0.09	0.254
Group6	**0.08**	0.13	0.1	0.1	0.1	0.251
Group7	**0.09**	0.14	0.1	0.1	0.11	0.233
Group8	**0.08**	0.14	0.09	0.1	0.1	0.22
Group9	**0.08**	0.13	0.1	0.1	0.11	0.25
Average A	**0.088**	0.132	0.098	0.099	0.105	0.241
Average B	0.11	0.133	**0.097**	**0.096**	0.104	0.217
Overall	0.089					

Table 4.16: House price prediction error for nine groups, using grouping based on both location (Zip code) and price and adding the boarder members to the other group. If we select the best model for each group, the overall error is **0.089**. Average A is the average error for adding boarder members and average B is the average error without adding boarder members.

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251
Group1	0.11	0.13	**0.1**	0.11	0.10	0.21
Group2	**0.08**	0.13	0.1	0.09	0.1	0.23
Group3	**0.09**	0.12	**0.1**	**0.09**	**0.09**	0.23
Group4	**0.07**	0.13	0.09	0.09	0.1	0.26
Group5	**0.1**	0.13	0.1	0.11	0.10	0.24
Group6	**0.08**	0.14	0.1	0.09	0.1	0.25
Group7	**0.07**	0.12	0.1	0.09	0.09	0.23
Group8	0.12	0.13	**0.09**	**0.09**	0.1	0.22
Group9	0.12	0.12	0.11	**0.1**	**0.1**	0.25
AverageA	**0.092**	0.127	0.098	0.095	0.097	0.237
AverageB	**0.098**	0.13	0.11	0.106	0.108	0.24
Overall	0.086					

Table 4.17: House price prediction error for nine groups, using grouping based on both location (Zip code) and the house size and adding the boarder members to the other group. If we select the best model for each group, the overall error is **0.086**. Average A is the average error for adding boarder members and average B is the average error with out adding boarder members.

Again, before creating models, we study the data structure. First, we study the features correlation which is illustrated in the Figure 4.25. It can be seen that there is not any strong correlation between features and evaporation. Moreover, we can plot the water evaporation distribution. As demonstrated in Figure 4.26, the mean of water evaporation, μ, is 8.75

Models:	ANN	GP	Lasso	Ridge	Linear	Polynomial	SVR
Original data	**0.14**	0.175	0.220	0.246	0.248	0.251	0.343
Add K-means	0.139	0.175	0.221	0.242	0.243	0.245	0.342
Add HAC	0.18	0.18	0.23	0.23	0.25	0.258	0.343
Add DBSCAN	0.135	0.18	0.22	0.22	0.24	0.248	0.345

Table 4.18: House price prediction if we add clusters as a feature to the dataset.

mm/day and the data standard deviation, σ, is 4.45 mm/day. Both the plot shape and the σ value, confirms that this dataset has considerable dispersion. In addition, If we compare the evaporation probability plot with a normal distribution which is shown in the Figure 4.27, it is apparent that this dataset do not follow a normal distribution, only the right part of the plot shows approximately a good fit to the linear normal probability plot. Further more, we can consider the shape of two above mentioned plots for the logarithm value of the target variable, evaporation. It was surprising that for logarithm value of water evaporation, we obtained the same results for the mean and the standard deviation. It means that the variation for this feature should be close to linear form. All above mentioned statistics about the evaporation dataset, confirm that it should not have considerable diversity in the order of values.

4.3.2.2 Water Evaporation Prediction Experiments

In the first step, various models which are defined in the section 2.1.2, such as linear and polynomial regressions, SVR, neural network and genetic programming are applied on the

Models:	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**	**overall**
Whole dataset	**0.14**	0.175	0.220	0.246	0.248	0.251	0.14
K-means	**0.09**	0.28	0.239	0.253	0.252	0.308	0.087
HAC,Ward	0.22	**0.21**	0.246	0.256	0.245	0.262	0.178
HAC,Average	0.206	**0.20**	0.243	0.236	0.231	0.252	0.172
DBSCAN	**0.13**	0.171	0.228	0.228	0.24	0.249	0.129
Grouping A	**0.129**	0.168	0.193	0.2	0.07	0.241	0.11
Grouping B	**0.093**	0.26	0.24	0.213	0.16	0.24	0.093
Grouping C	**0.13**	0.22	0.16	0.163	0.12	0.246	0.09
Grouping D	**0.12**	0.154	0.217	0.221	0.225	0.243	0.12
Grouping E	**0.096**0	0.113	0.089	0.09	0.095	0.23	0.087
Grouping F	0.11	0.133	**0.097**	0.096	0.104	0.217	0.123
Grouping G	0.16	**0.125**	0.17	0.178	0.18	0.26	0.125
Grouping H	**0.101**	0.146	0.10	0.103	0.11	0.257	0.095
Grouping I	**0.098**	0.13	0.11	0.106	0.108	0.24	0.098
Grouping J	**0.092**	0.127	0.098	0.095	0.097	0.237	0.086

Table 4.19: Comparison between different clustering and grouping methods effects on house price prediction. In this table, HAC has two linkage method: ward and average. Grouping A is grouping based on price. Grouping B is grouping Based on grade. Grouping C presents grouping based on predicted price. Grouping D is grouping based on location. Grouping E: 6 groups based on location and price. Grouping F: 9 groups based on location and price. Grouping G: 18 groups based on location, size and number of bedrooms. Grouping H: 12 groups based on location, size and number of bedrooms. Grouping I: 9 groups based on location and size. Grouping J: 9 groups based on location and size adding boarder members to the groups.

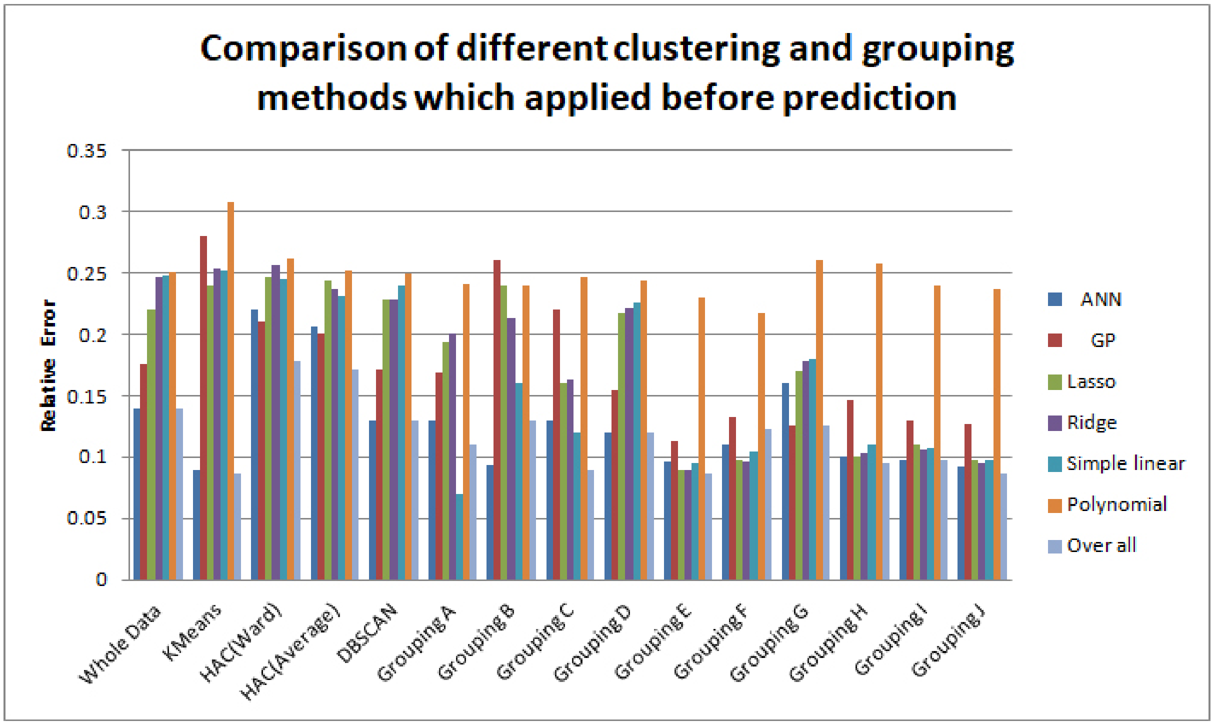


Figure 4.24: Comparison between different clustering and grouping methods effects on house price prediction. In this table, HAC has two linkage method: ward and average. Grouping A is grouping based on price. Grouping B is grouping Based on grade. Grouping C presents grouping based on predicted price. Grouping D is grouping based on location. Grouping E: 6 groups based on location and price. Grouping F: 9 groups based on location and price. Grouping G: 18 groups based on location, size and number of bedrooms. Grouping H: 12 groups based on location, size and number of bedrooms. Grouping I: 9 groups based on location and size. Grouping J: 9 groups based on location and size adding boarder members to the groups.

whole dataset. Similar to our experiment with house price dataset, we found out how the results may change if we only create smaller sub-sets of our data without any condition. Therefore, we randomly divided the dataset to smaller groups with 2000 and 1200 members to investigate the effect of the data size on prediction accuracy for each model. As

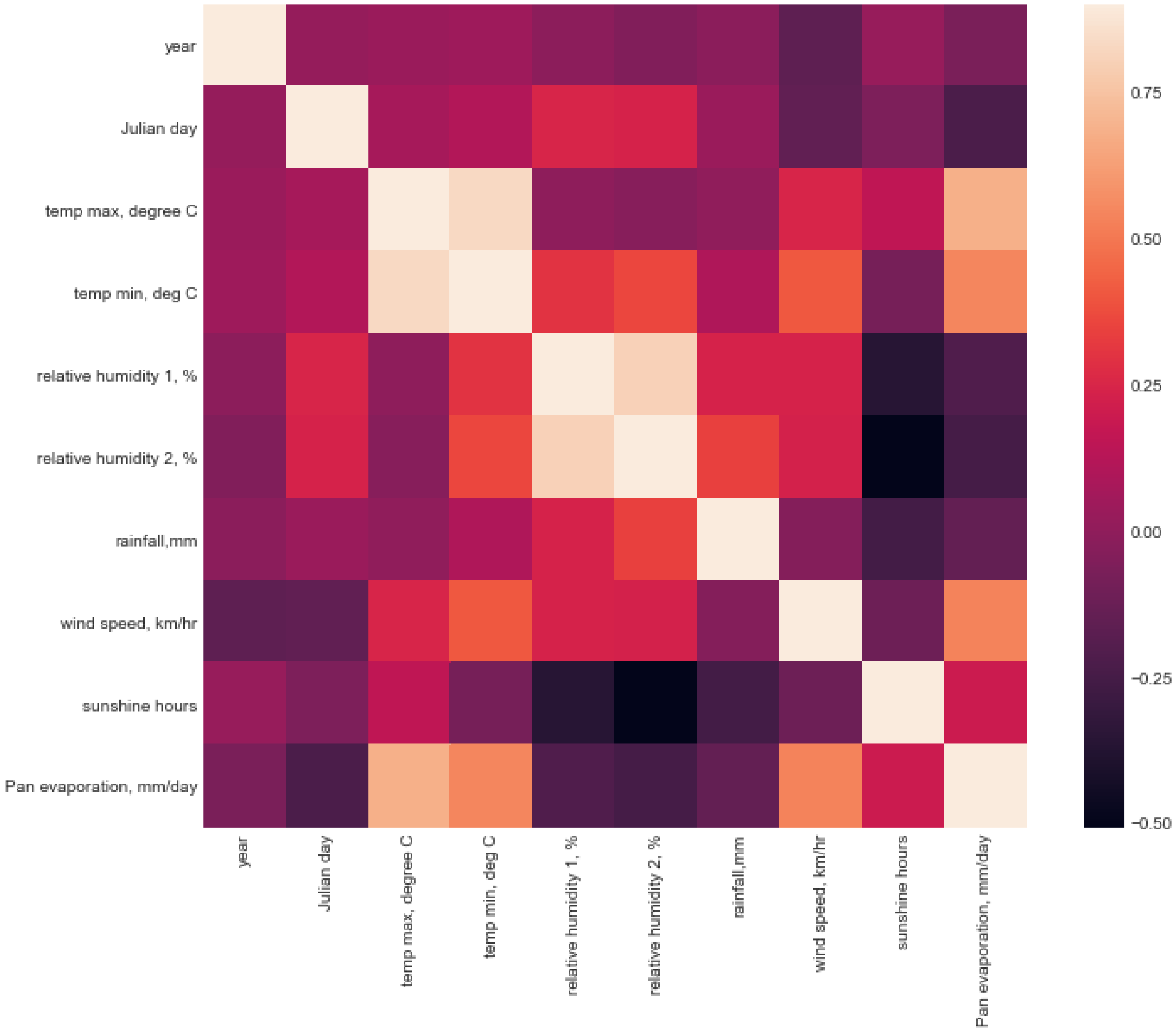


Figure 4.25: Correlation between the features in Water Evaporation dataset.

expected, the accuracy decreases by reducing data-size, but the decreasing accuracy rate and improving the error, differs for different models. Table 4.20 and Figure 4.28 display the errors belong to each prediction model. As we expected, all methods have lower accuracy in smaller datasets. Neural network and polynomial models are very sensitive to the dataset size (the number of instances) and GP has the lowest variation in error when we decrease the number of samples.

K-means Clustering

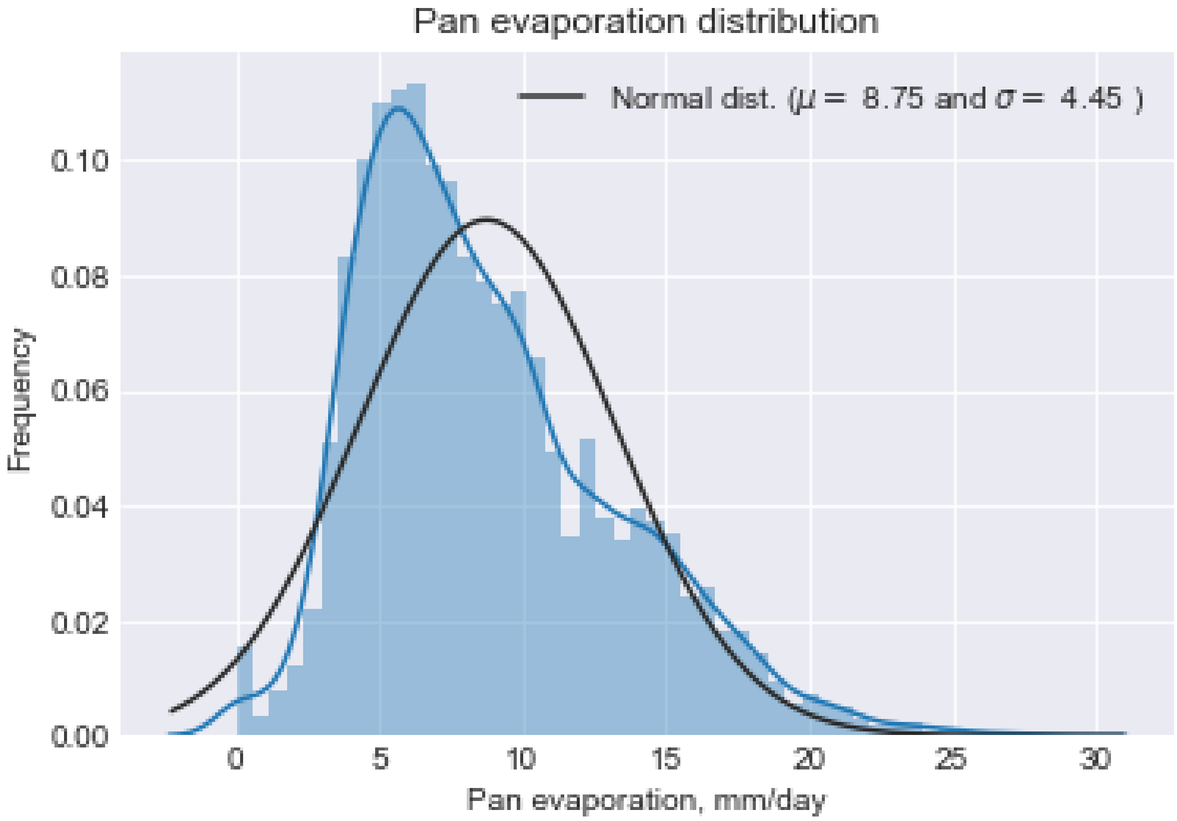


Figure 4.26: Comparison of Water evaporation distribution in the agricultural dataset and the normal distribution. The water evaporation unit is mm/day, μ is the mean of pan evaporation and σ is the data standard deviation.

The next step is Clustering data using K-means technique to create smaller datasets which members are similar together. First, we need to determine the number of clusters, K. Using the Silhouette score, we try to discover the most efficient number of clusters. Figure 4.29 reveals that the score decreases by increasing the number of clusters. but we can not select the large value for k, because when K is large, the number of instances in each cluster decreases and consequently, the accuracy will decrease. Therefore, we followed the experiment with two values for K; 4 and 5. We found out when k is 5, the average error for most of the prediction models is greater than 0.19 which comparing to the prediction errors

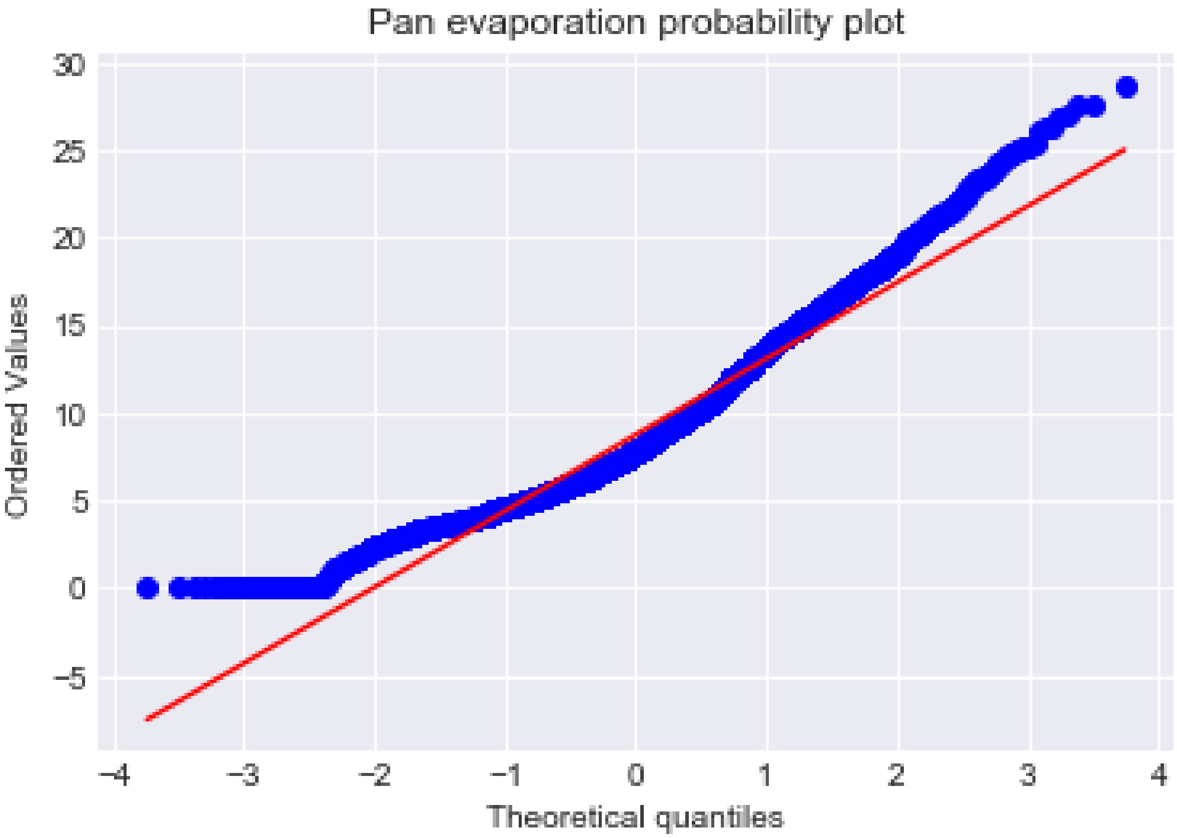


Figure 4.27: Comparison of water evaporation dataset probability with probability plot of a normal structure.

for the whole data, it is considerably large. When k is 4, we have lower errors which are provided in Table 4.21 and Figure 4.30. If we select the best model for each group and then calculate the average, the total error for prediction through clustering, is **0.111** which shows that for water evaporation dataset, this method can not provide higher accuracy comparing to applying prediction models to the whole data. Maybe the reason is the low number of instances in agricultural dataset comparing to the previous experiment with house price dataset. The other reason may be the data structure which K-means is not a proper method for clustering this data, except for the symbolic regression. The only method that shows improvement in prediction after K-means clustering, is GP even though the improvement

Models	ANN	GP	Lasso	Ridge	Linear	Polynomial	SVR
Whole Data	**0.08**	0.12	0.089	0.088	0.084	0.082	0.1
Group 1	**0.13**	0.15	0.22	0.21	0.23	0.24	0.26
Group 2	0.28	**0.18**	0.27	0.27	0.28	0.35	0.32

Table 4.20: Different methods relative error for water evaporation prediction in agricultural dataset and randomly created smaller sub-sets. group 1 has approximately 2000 instances and group 2 has around 1200 instances.

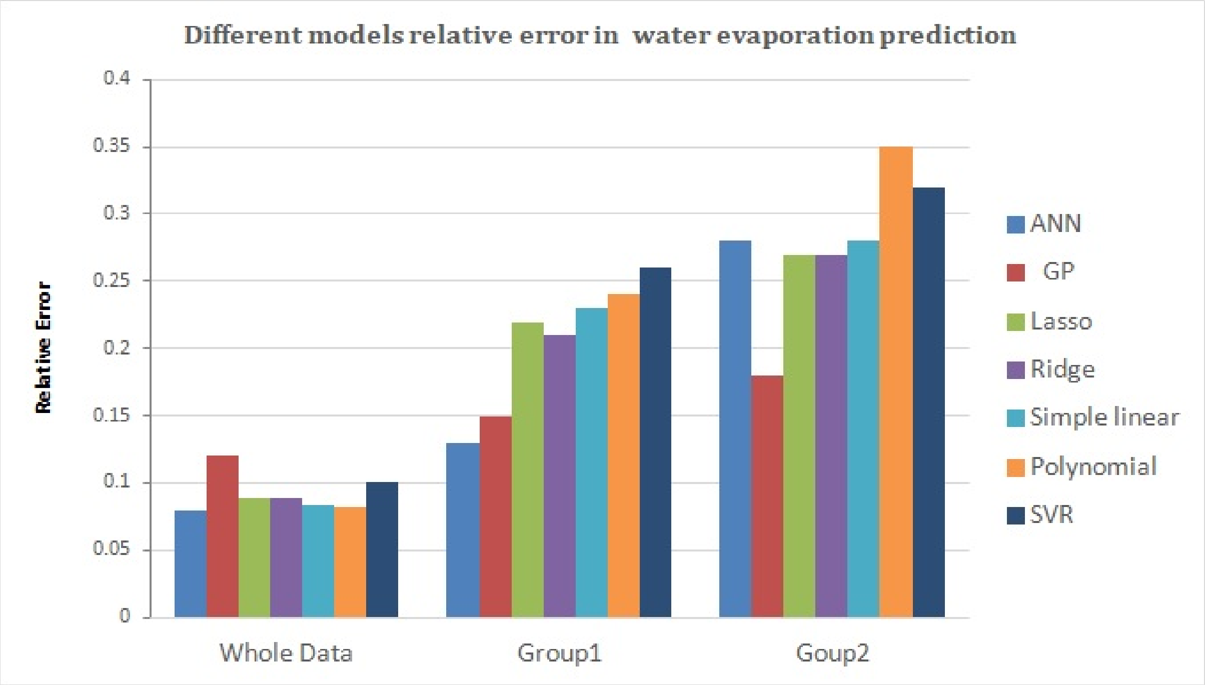


Figure 4.28: Different methods relative error in regression for the agricultural dataset and randomly created smaller sub-sets. Group 1 has approximately 2000 instances and Group 2 has around 1200 instances.

is only decreasing the error from 0.12 for the whole data to the average error 0.114 for all clusters and 0.1 for the best prediction belongs to group 2.

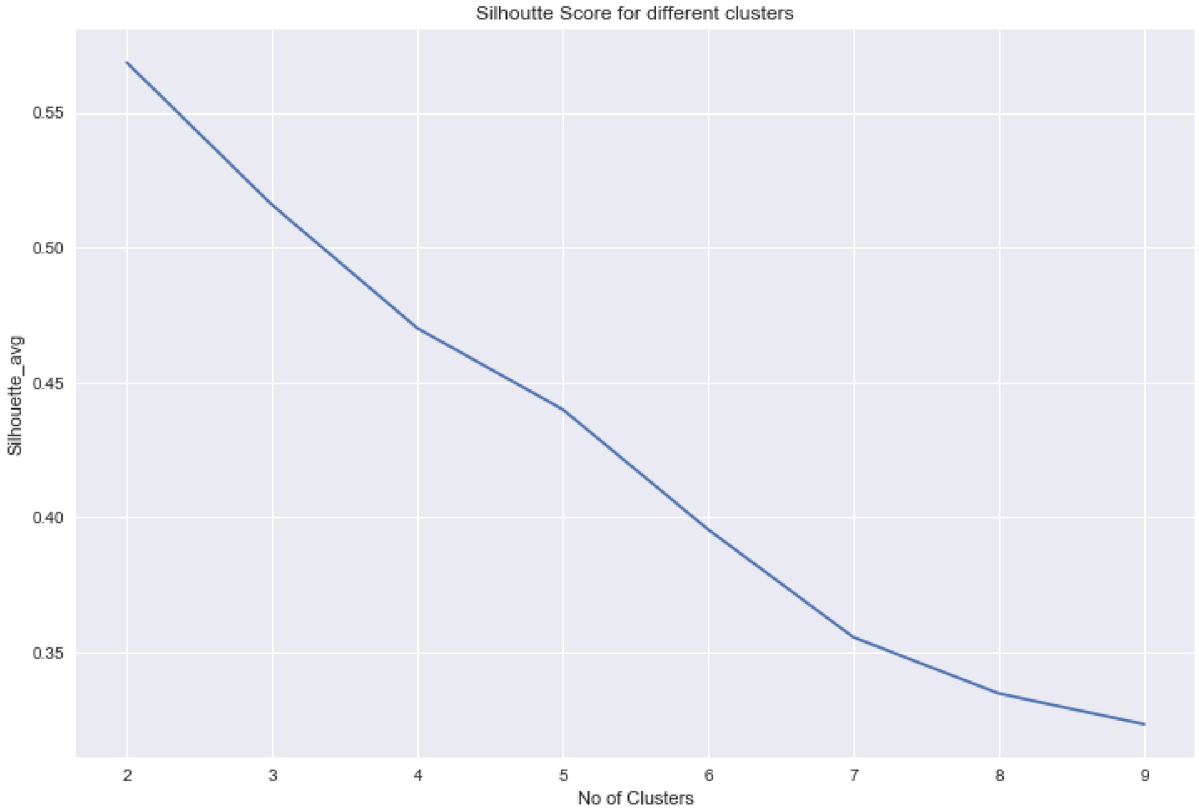


Figure 4.29: Silhouette Score for K-means clustering applied on agricultural dataset.

Hierarchical Agglomerative clustering

The same as experiment with the house price dataset, the next clustering technique we Employed is HAC. We obtained better results using ward linkage. In other linkage methods, more than 70 percent of the data points were assigned to one cluster. The hierarchical ward method applied on water evaporation data,is visualised in Figure 4.31. As depicted, by selecting different cutting lines, we can obtain various number of clusters with approximate estimation about the number of members in each cluster. The pink line in Figure 4.31, creates three clusters but it is obvious that one cluster will be very larger than the others. If we use the blue cutting line, we obtain four clusters with approximately similar number of instances in each group. Table 4.22 and Figure 4.32 show the results of different methods

Models	ANN	GP	Lasso	Ridge	Linear	Polynomial	SVR
Whole Data	0.08	**0.12**	0.089	0.088	0.084	0.082	0.1
Group1	**0.1**	0.11	0.13	0.128	0.12	0.13	0.2
Group2	**0.1**	**0.1**	0.17	0.16	0.12	0.14	0.2
Group3	**0.12**	**0.12**	0.19	0.192	0.19	0.19	0.19
Group4	0.16	**0.15**	0.23	0.23	0.22	0.25	0.29
Average	**0.112**	0.114	0.17	0.163	0.151	0.164	0.22
Overall				0.111			

Table 4.21: Different methods relative error for water evaporation prediction in agricultural dataset and four Sub-datasets created by K-means . group 1 has approximately 2500 group 2 has 2200, group 3 has 2000 and group4 includes 1000 instances. The overall prediction error after applying K-means is 0.111.

prediction for the evaporation using HAC clustering. None of prediction methods show improvement in accuracy. All methods have considerable greater error after HAC clustering only GP has low change in prediction error. Therefore, it can be concluded that our proposed method is not beneficial for this dataset or maybe the HAC can not cluster this dataset properly. Even it seems that HAC clustering was not successful, comparing its results to the randomly created groups predictions, shows that HAc clustering could improve the accuracy but not as much as to be better than the whole data predictions.

DBSCAN Clustering

We did not obtain satisfactory results with K-means and HAC clustering, therefore, another clustering method was studied on agricultural dataset. We applied DBSCAN and we studied different ϵ and min-samples (minPts) values, but surprisingly, we could not obtain

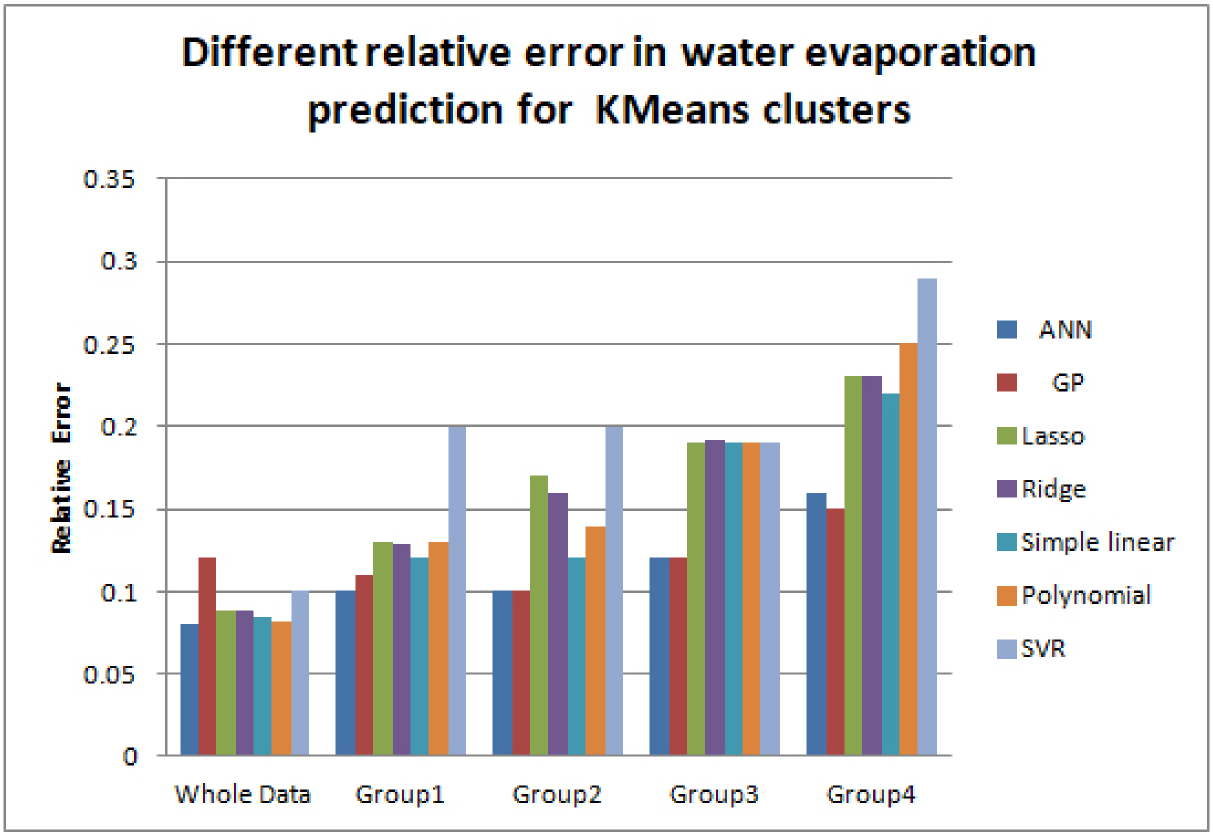


Figure 4.30: Different methods relative error for water evaporation prediction in agricultural dataset and four Sub-datasets created by K-means . group 1 has approximately 2500 group 2 has 2200, group 3 has 2000 and group 4 includes 1000 instances.

proper clusters. for different parameter values, DBSCAN results in one large cluster with more than 7000 samples and another small one with less than 700 instances. It means that this clustering method assign most data points in the same cluster. It happens when clusters are very close to each other and neighbour clusters are connected.

Human Knowledge Base Grouping

In previous experiment with the House price dataset, we had relatively good results from grouping based on the experts knowledge. Therefore, the last experiment on agricultural dataset was grouping data based on the most important feature from the specialists perspec-

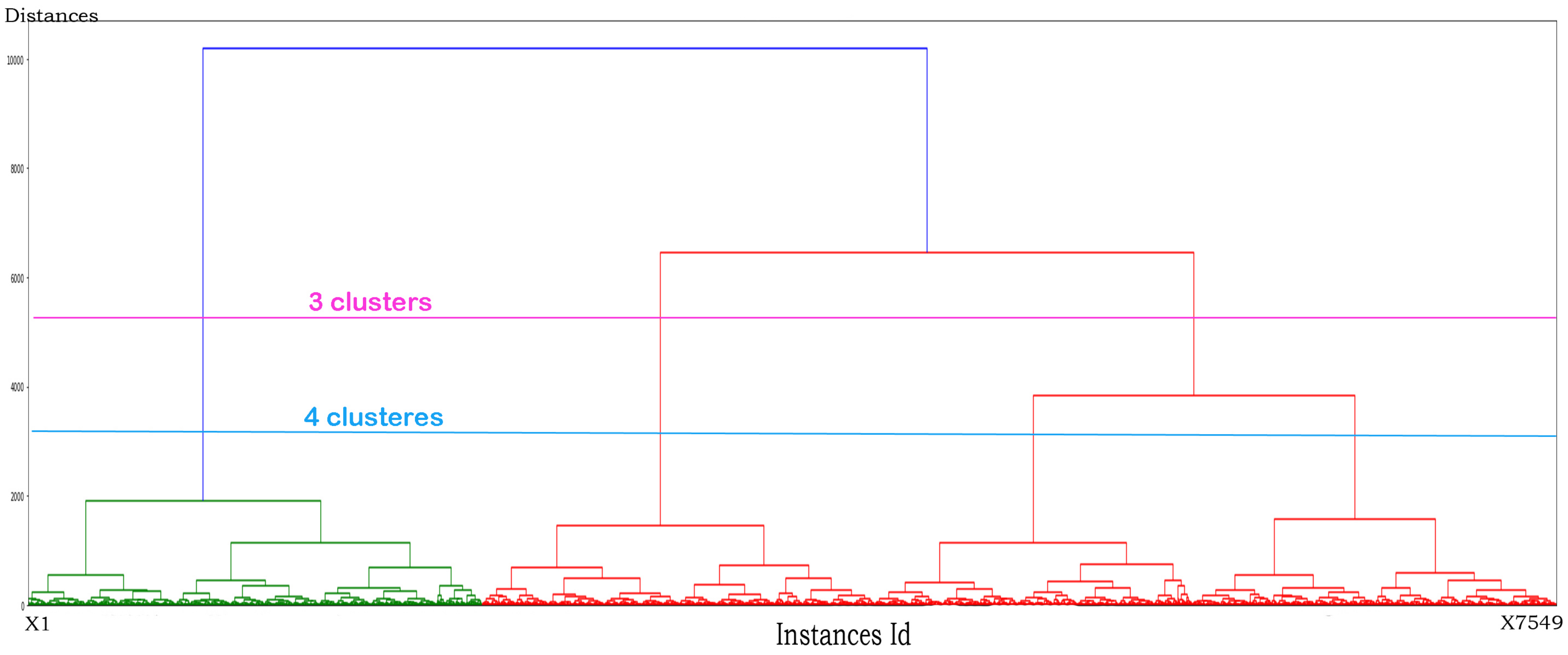


Figure 4.31: Hierarchical clustering dendrogram for agricultural dataset, created by ward linkage method. The horizontal pink line shows a cut line which creates three clusters and using the blue line, generates four groups.

Models	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**	**SVR**
Whole Data	0.08	**0.12**	0.089	0.088	0.084	0.082	0.1
Group1	0.13	**0.12**	0.15	0.14	0.14	0.15	0.22
Group2	0.14	**0.13**	0.16	0.16	0.15	0.16	0.23
Group3	0.14	**0.13**	0.15	0.17	0.17	0.16	0.22
Group4	**0.18**	0.16	0.22	0.22	0.22	0.21	0.28
Average	0.143	**0.131**	0.163	0.165	0.163	0.164	0.231
Overall				0.131			

Table 4.22: Different methods relative error for water evaporation prediction in agricultural dataset and four Sub-datasets created by HAC. group 1 has approximately 2400 group 2 and group 3 have 2100 members and group 4 includes 1200 instances. The overall prediction error after applying HAC is 0.131.

tive. Accordingly, the data is grouped based on the seasons which resulted in four groups with approximately 1900 samples in each. Again, we study different models performance for each group. Table 4.23 and Figure 4.33 reveal that comparing to the predictions for the whole data, none of the prediction models has a better performance for the sub-groups. Similar to the HAC clustering results, if we compare the sub-groups predictions to the randomly created groups, grouping based on seasons has better performance but it is weaker than the HAC clustering and the whole data in predictions.

4.3.2.3 Water Evaporation Prediction Results

The water evaporation predictions results in Table 4.24 and Figure 4.34 reveals that our proposed method is not beneficial for this dataset and for most of the prediction models

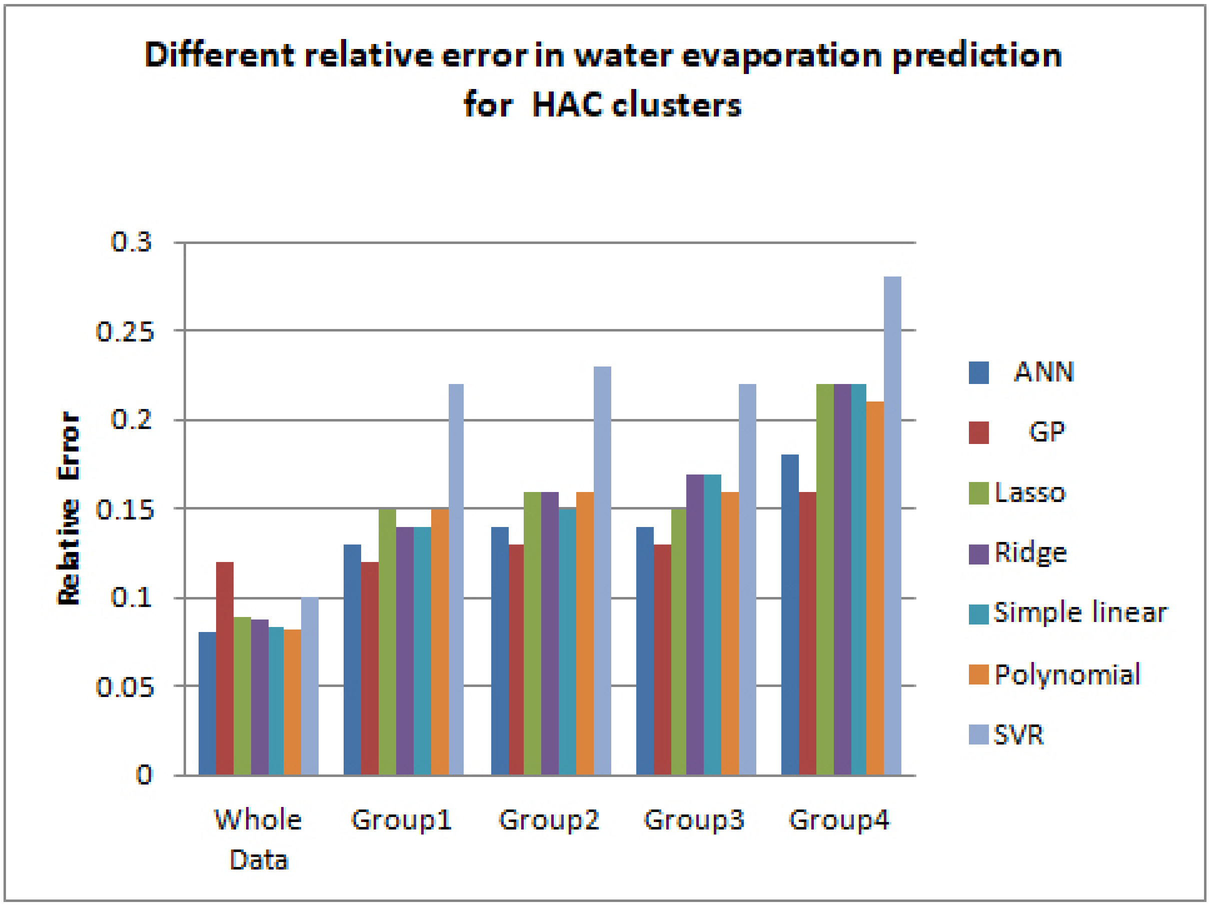


Figure 4.32: Different methods relative error for water evaporation prediction in agricultural dataset and four Sub-datasets created by HAC. group 1 has approximately 2400, group 2 and group 3 have 2100 samples and group 4 includes 1200 instances.

because the average error after clustering is larger than the predictions for the whole data. It reveals that when we split small dataset, the created clusters and groups do not contain enough number of instances for training the model accurately. The only considerable result belongs to GP predictions for K-means clustering. Similar to previous dataset, GP was successful comparing to other models in small sub-datasets. In this experiment, for K-means clustering the GP's average error is 0.114 which is lower than 0.12, even it is larger than the overall error, 0.08, for the whole data. In addition, after HAC clustering, GP perform

Models	**ANN**	**GP**	**Lasso**	**Ridge**	**Linear**	**Polynomial**	**SVR**
Whole Data	0.08	**0.12**	0.089	0.088	0.084	0.082	0.1
Group1	**0.12**	0.18	0.163	0.164	0.162	0.21	0.22
Group2	**0.13**	0.15	0.182	0.181	0.178	0.2	0.24
Group3	**0.11**	0.17	0.125	0.122	0.22	0.2	0.21
Group4	0.14	0.15	**0.134**	0.15	0.16	0.22	0.22
Average	**0.125**	0.165	0.151	0.154	0.18	0.208	0.222
Overall	0.1235						

Table 4.23: Different methods relative error for water evaporation prediction in agricultural dataset and four Sub-datasets created by grouping based on the seasons. All groups have approximately 1900 instances.

better than other models.

4.3.3 Case Study 3: Facebook Comment Volume prediction

The case studies that we used to evaluate our proposed method, had 20 or fewer features. If we have a dataset with 50 features or more, would the result be the same? In order to test this case, we used a large dataset from the social network. Prediction in social networks has an important role in social education, information spreading, politics, and economy. For instance, in marketing, it is very important to know where and when we should post the advertisements to have the highest feedback. Therefore, as our last case study, we applied our method on the Facebook Comment Volume dataset [14].

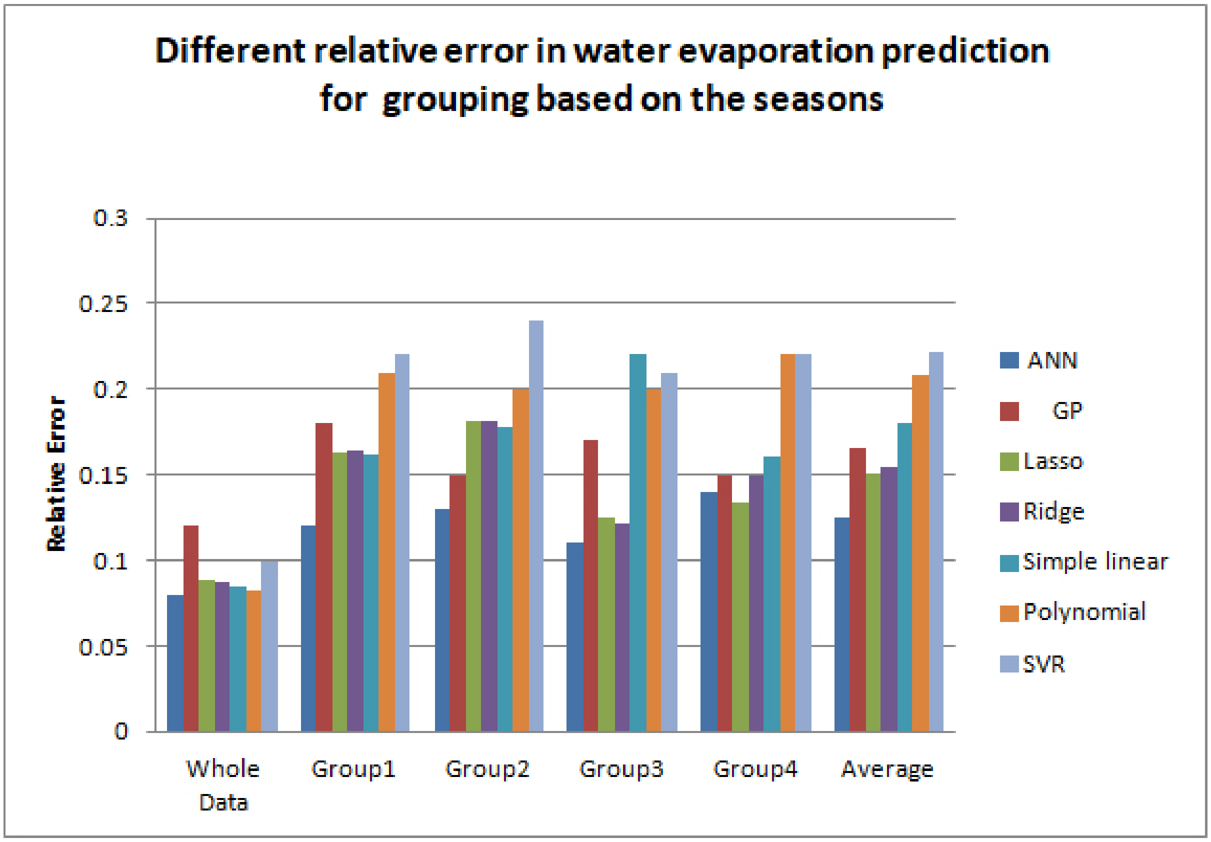


Figure 4.33: Different methods relative error for water evaporation prediction in agricultural dataset and four Sub-datasets created by grouping based on the seasons. All groups have approximately 1900 instances

4.3.3.1 Facebook Comments Dataset

The Facebook Comment Volume dataset that we used to evaluate our proposed method for large size datasets, belongs to the year 2016. It contains 40,949 samples with 54 features. The goal is predicting the number of comments that each post may have. There are high volume of instances in the Facebook dataset which is a multivariate dataset with real and integer values with no sparsity. Again, before studying prediction models, we study the data structure, first, we check the features correlation which is depicted in the Figure 4.35. It shows that there is a strong correlations between one of the features to all other variables

Models:	ANN	GP	Lasso	Ridge	Linear	Polynomial	SVR	overall
Whole Data	**0.08**	0.12	0.088	0.087	0.084	0.082	0.1	0.08
K-means	**0.112**	0.114	0.17	0.163	0.151	0.164	0.22	0.111
HAC	0.143	**0.131**	0.163	0.165	0.163	0.164	0.231	0.131
Grouping	**0.125**	0.165	0.151	0.154	0.18	0.208	0.222	0.1235

Table 4.24: Comparison between different clustering and grouping effects on water evaporation prediction. The grouping is based on seasons.

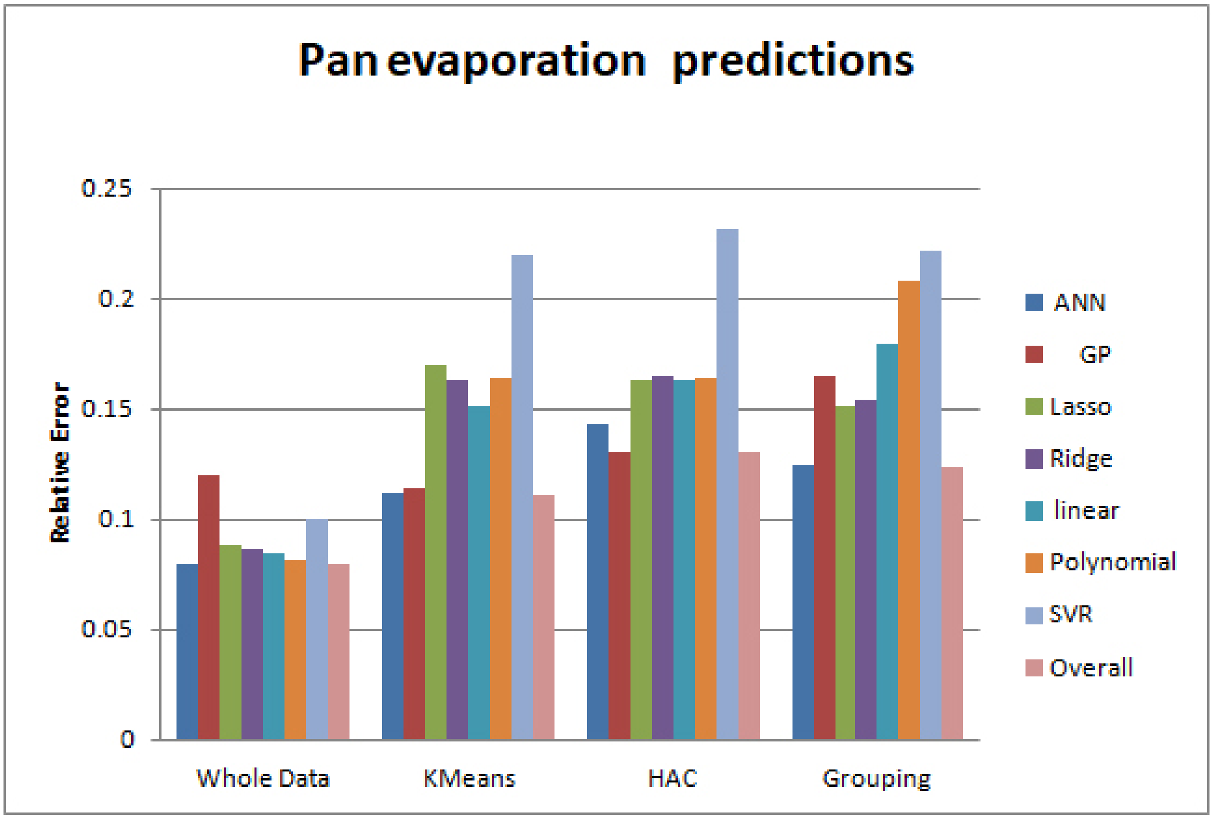


Figure 4.34: Comparison between different clustering and grouping methods effect on water evaporation prediction. The grouping is based on seasons.

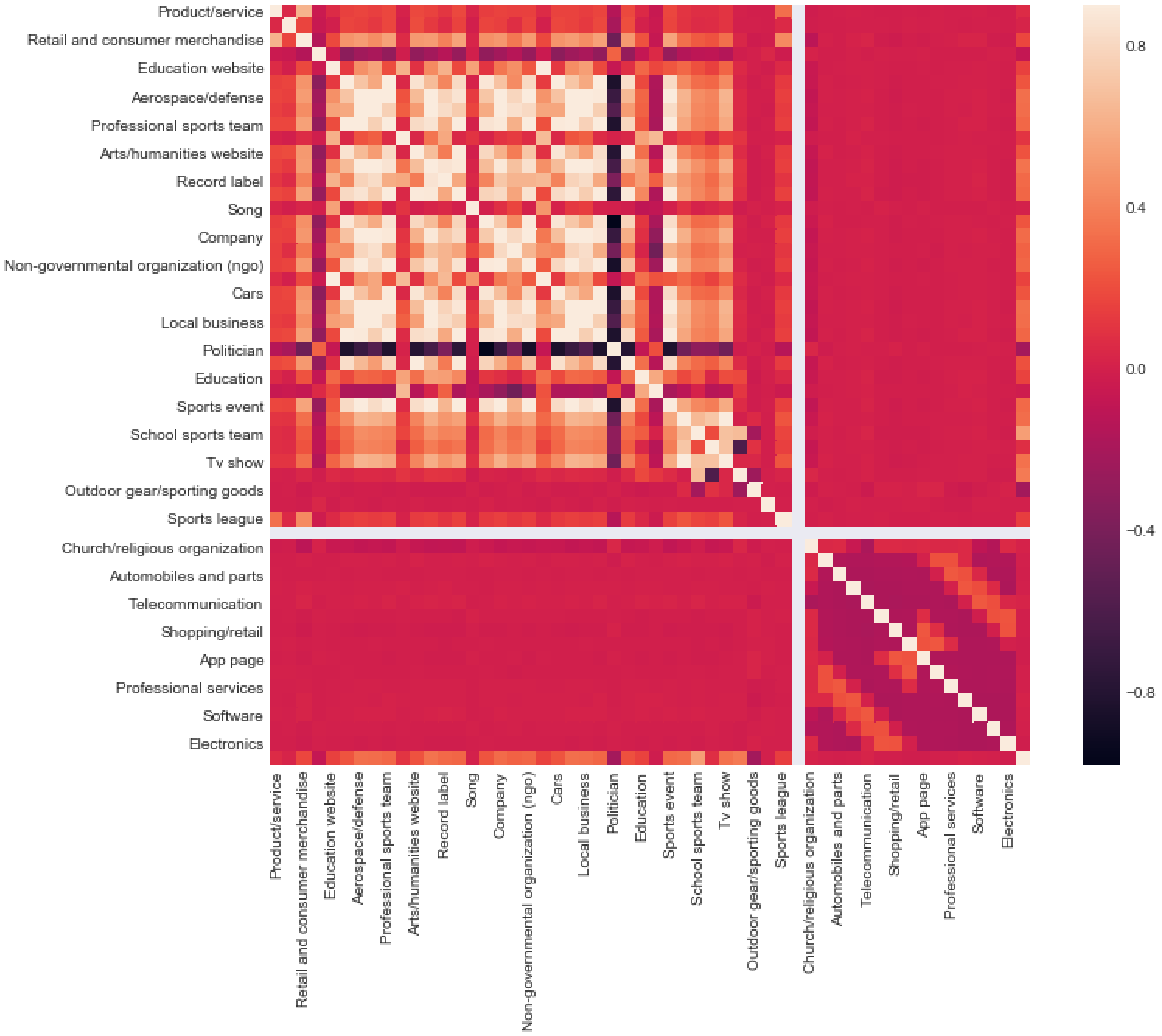


Figure 4.35: Correlation between the Facebook comments dataset features.

when we compared it to the data table, we found out that this feature is ”Entertainer” and for all instances it has the value equal to zero. Therefore, we eliminated this feature and the results are shown in 4.36. Moreover, we can plot the variables distribution similar to previous experiments. We selected the Sports event comments as a prediction target. Thus,

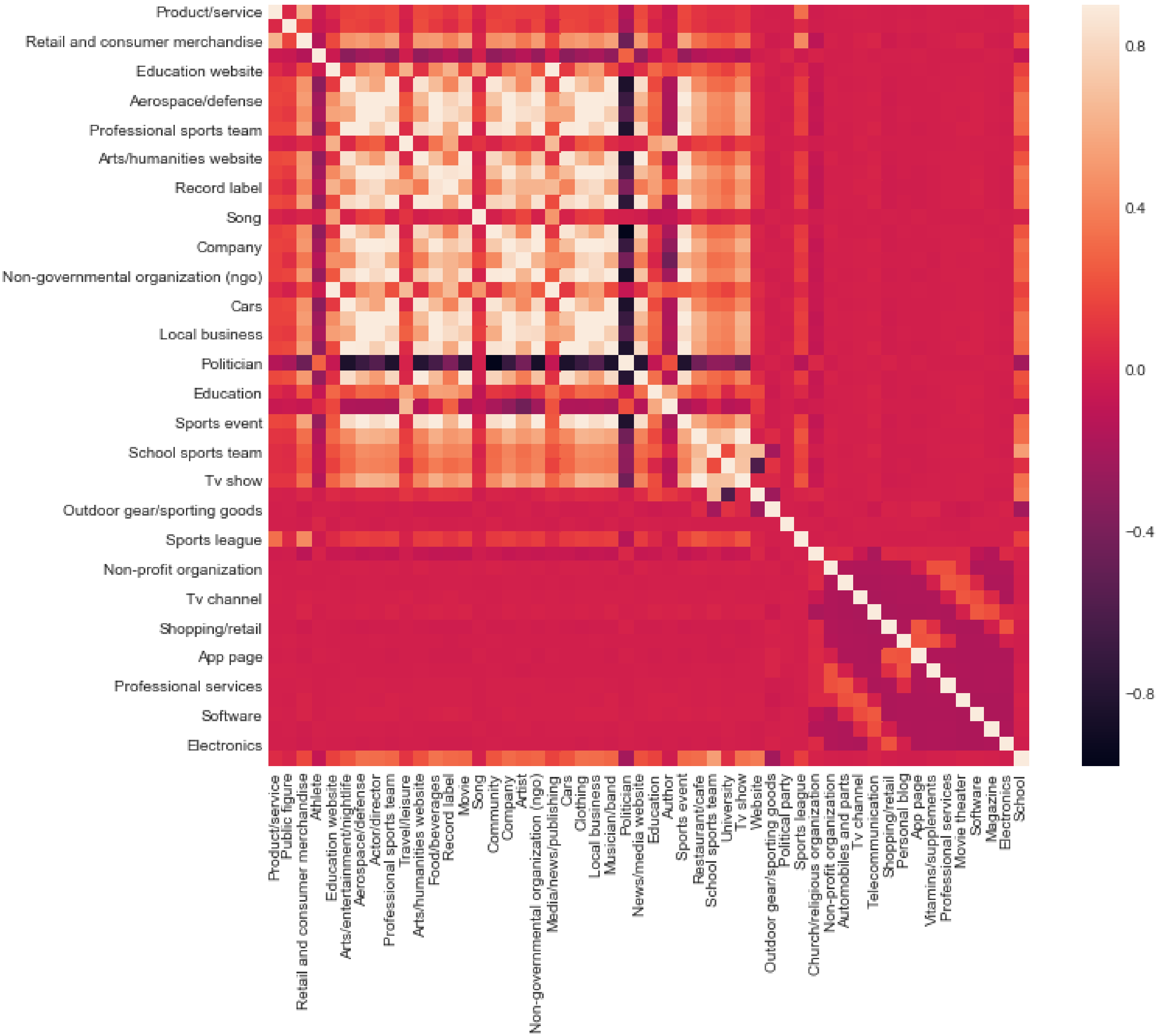


Figure 4.36: Correlation between the Facebook comments dataset features after eliminating the highly correlated feature.

we study this variable values distribution and compare it with normal distribution. As illustrated in Figure 4.37, the mean value for number of comments, μ, mu = 55.84 and the data standard deviation, σ, is 73.81. Both the plot shape and the σ value, confirm that this

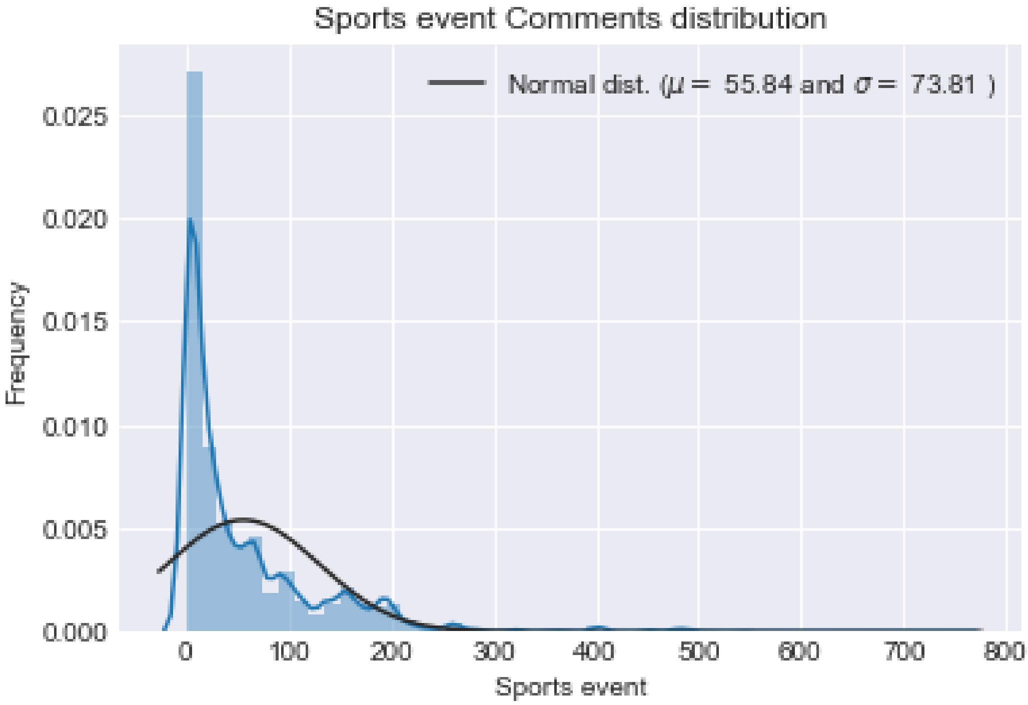


Figure 4.37: Comparison of number of Sport event comments in the Facebook dataset and the normal distribution. μ is the mean value and σ is the data standard deviation.

Models	ANN	GP	Linear	Polynomial	SVR
Whole Data	**0.1**	0.18	0.24	0.23	0.3
Group1	0.27	**0.23**	0.34	0.3	0.37
Group2	0.312	**0.27**	0.37	0.39	0.44

Table 4.25: Different relative error in Facebook comments volume predictions for random selected sub-datasets. Group 1 includes 4000 and Group 2 has 2000 instances.

dataset has considerable dispersion. Usually, a large dispersion, makes it hard to fit a model on entire data. In addition, If we compare the Sports event comments probability plot with a linear probability of a normal structure, which is shown in the Figure 4.38, it is apparent

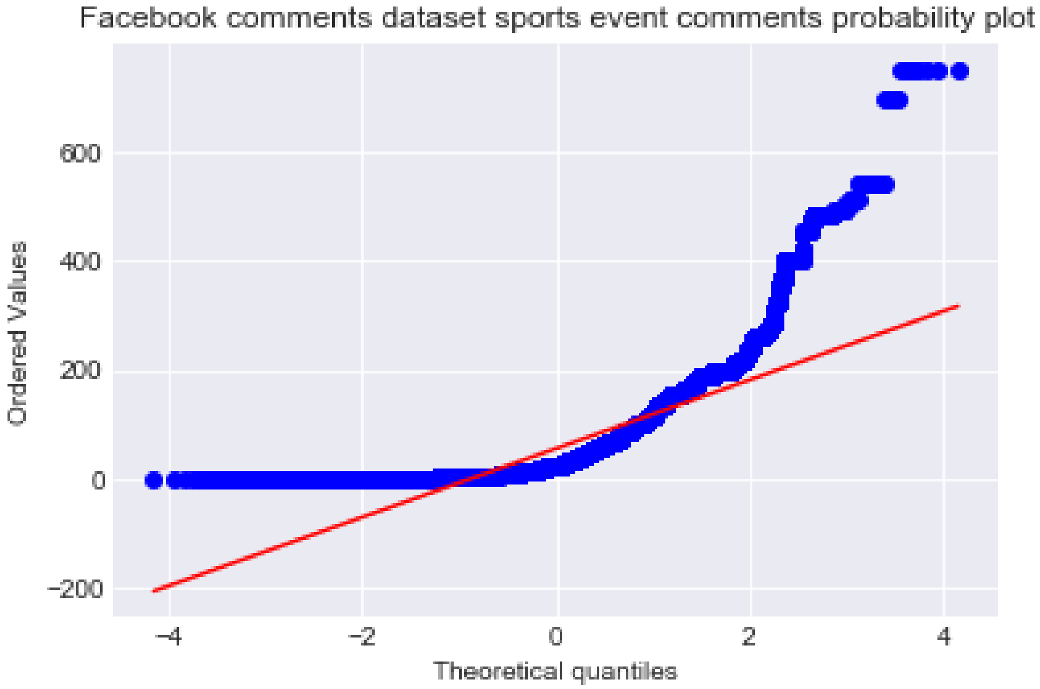


Figure 4.38: Comparison of number of Sport events comments probability in the Facebook dataset and a normal structure.

that this dataset probability do not follow a linear form (normal structure). As it is apparent in Figure 4.37 the order of the number of comments has large diversity. Thus, we also study the logarithm of the values and draw the logarithm of distribution and probability. The outcome is presented in the Figure 4.39 and Figure 4.40. Still the distribution does not fit a normal shape and has large fluctuations. Comparing to the original data probability plot, the logarithm of probability is closer to the linear normal distribution but still it shows dispersion. All above mentioned shows that we may need to create a complex prediction model for this dataset.

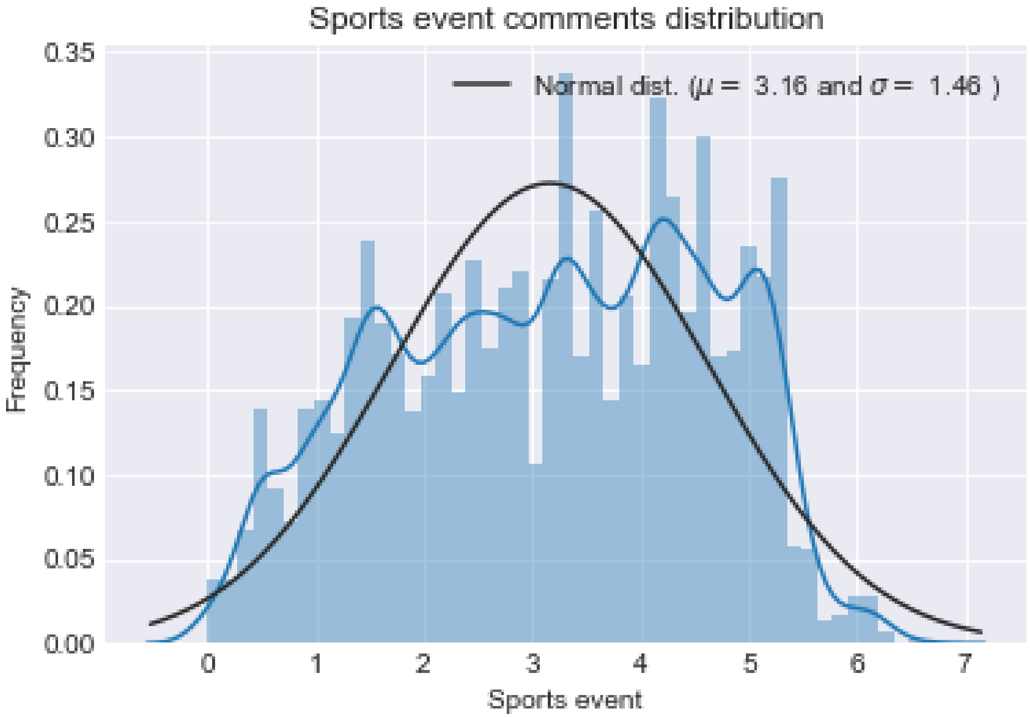


Figure 4.39: Comparison of logarithm of Sport events comments volume in the Facebook dataset and the normal distribution. μ is the mean value and σ is the data standard deviation.

4.3.3.2 Facebook Comment Volume prediction Experiments

Similar to previous experiments on the house price dataset and the agricultural dataset, first, we apply different prediction methods to the entire data. As we expected, neural network has very good performance but relatively good predictions by GP was unexpected. It is interesting to find out how the results will change if we only use a small dataset with a limited number of instances but with the same number of features. Therefore, using random selection of instances, the smaller dataset with 2000 and 4000 samples were created and tested for the same regression models to study the effect of data size on predictions. The results in Table 4.25 and Figure 4.41 reveal that neural network losses its accuracy dramatically by decreasing the size of the training data. This experiment demonstrate that

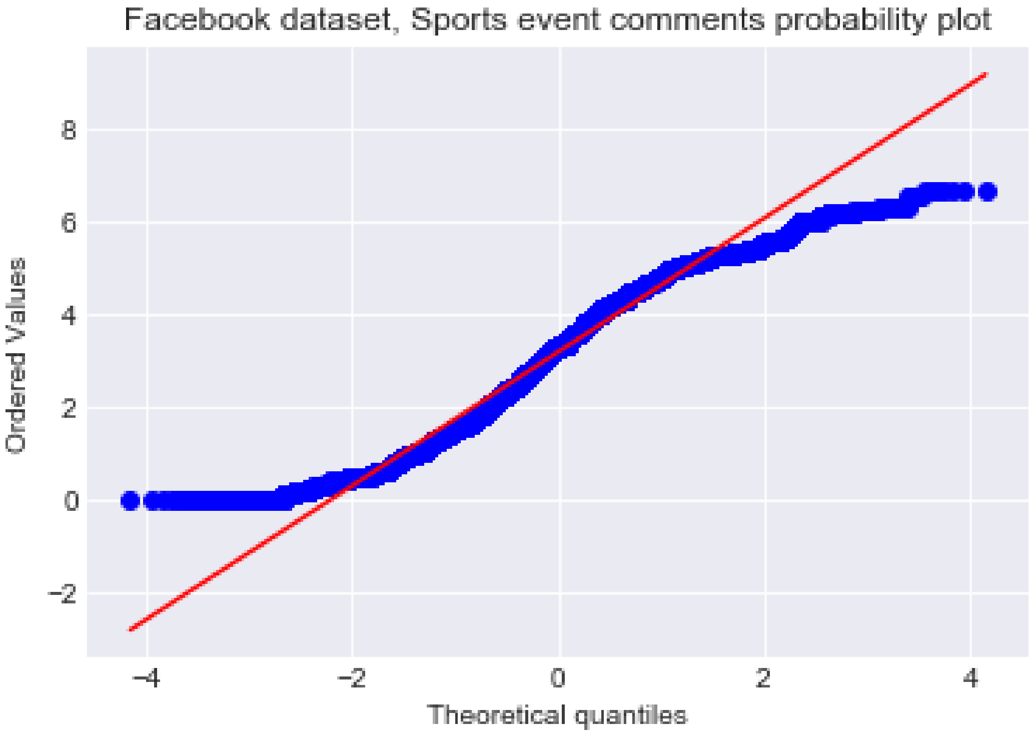


Figure 4.40: Comparison of logarithm of Sport events comments volume probability in the Facebook dataset and a normal structure.

how sensitive is this dataset to the data size. The Neural network and polynomial models are very dependent on the number of samples. When we use the whole dataset, neural network is the most accurate model, but surprisingly, when we have Small size datasets, GP performs better than other models.

K-means Clustering

We found out that in smaller groups of Facebook dataset, we loose accuracy. But if we select samples which are correlated, how the results may change? Similar to previous experiments we apply different clustering methods to examine if we can decrease the prediction error for clusters comparing to the whole dataset predictions. K-means is the first clustering method which is utilized for clustering. The first step of applying K-means is

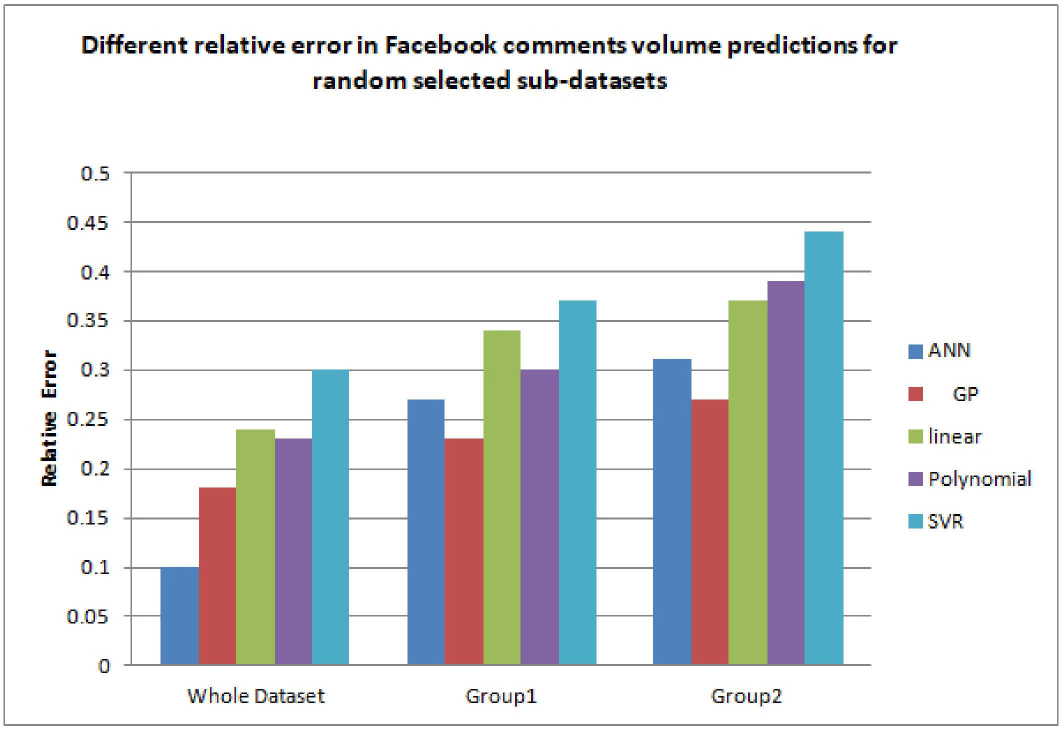


Figure 4.41: Different relative error in Facebook comments volume predictions for random selected sub-datasets.

discovering the number of clusters. Using the Silhouette score, we can determine the most efficient number of clusters. Figure 4.42 depicts that the best number of clusters for this dataset is 10. Therefore, we created 10 clusters using K-means method. The created clusters, group1 to 10, have 20143, 5042, 5332, 4852, 3894, 3267, 2853, 1807, 1364 and 1322 members respectively. Again, we examine the different prediction techniques performance for each cluster. Comparing the results which are demonstrated in Table 4.26 and Figure 4.43 confirm that clustering can improve the prediction. If we select the best model for each group, the overall error is 0.09 which is a little lower than the best prediction before clustering, which has the error equal to 0.1. Moreover, even we could not obtain lower error in all groups and models after clustering, but all studied models have improvement in overall accuracy. The most enhancement in prediction belongs to GP which its prediction

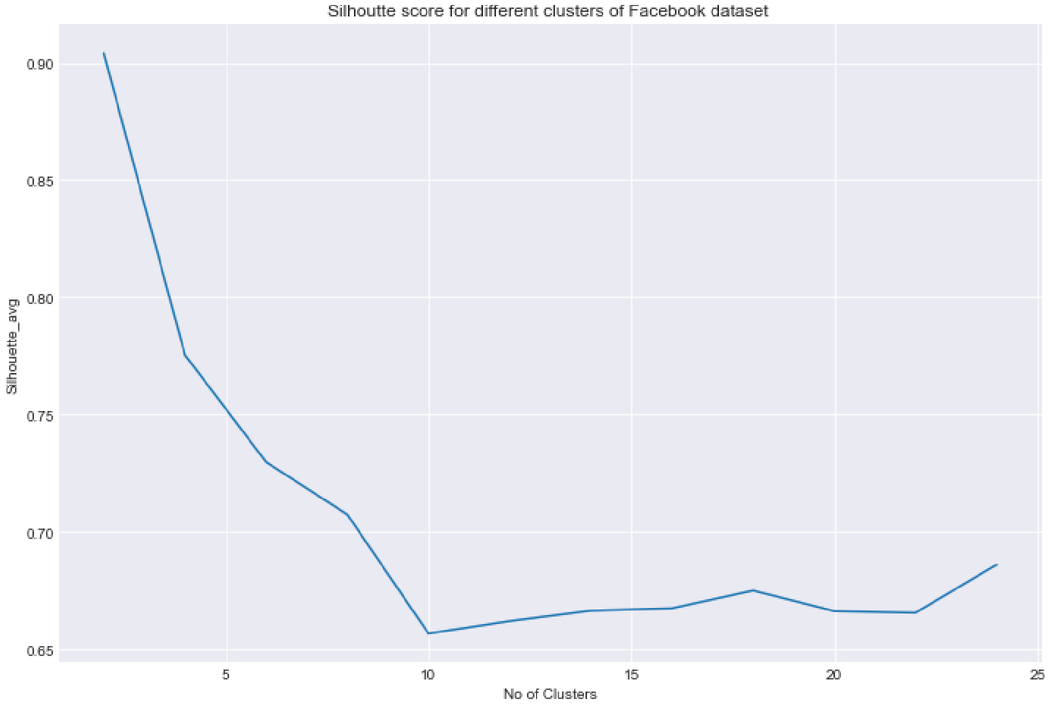


Figure 4.42: Silhoutte score for different number of clusters in Facebook dataset.

error before clustering is 0.18 and drops to 0.127 after clustering. In addition, this experiments verifies that neural network has the best performance in large clusters and all models have weak performance in relatively small size groups. But ANN is affected the most and in this cases, GP has better results comparing to other models predictions.

Hierarchical Agglomerative clustering

The K-means clustering improves prediction a little but not produces considerable enhancement in accuracy. It may be because of the lack of precision in clustering. Therefore, other clustering methods may result in more accurate predictions. HAC is one of the clustering techniques that we employed in previous experiments and again we utilize it for the Facebook comments dataset. Unfortunately, as this clustering method resulte in very large cluster which included almost all instances and some very small groups with less than 500

Models	**ANN**	**GP**	**Linear**	**Polynomial**	**SVR**
Whole Data	**0.1**	0.18	0.24	0.23	0.3
Group1	**0.07**	0.12	0.18	0.17	0.25
Group2	**0.08**	0.13	0.18	0.18	0.24
Group3	**0.09**	0.14	0.15	0.17	0.25
Group4	**0.09**	0.12	0.16	0.17	0.23
Group5	**0.08**	0.14	0.14	0.16	0.25
Group6	**0.09**	0.11	0.12	0.17	0.25
Group7	**0.09**	0.13	0.13	0.17	0.23
Group8	0.13	**0.11**	0.18	0.20	0.31
Group9	0.14	**0.11**	0.16	0.19	0.28
Group10	0.14	**0.11**	0.17	0.22	0.29
Average	**0.09**	0.127	0.17	0.18	0.261
Overall			0.09		

Table 4.26: Different relative error in Facebook comments volume predictions for different sub-datasets created by K-means . group 1 to 10, have 20143, 5042, 5332, 4852, 3894, 3267, 2853, 1807, 1364 and 1322 members respectively. If we select the best model for each group, the overall error is 0.09.

samples, we could not acquire good results in this experiment.

DBSCAN Clustering

The last utilized clustering method is DBSCAN the advantage of this method is that there is no need to initialize the number of clusters. Applying this method on Facebook dataset, resulted in 5 clusters. Unfortunately, again, most samples are assigned to one group. The

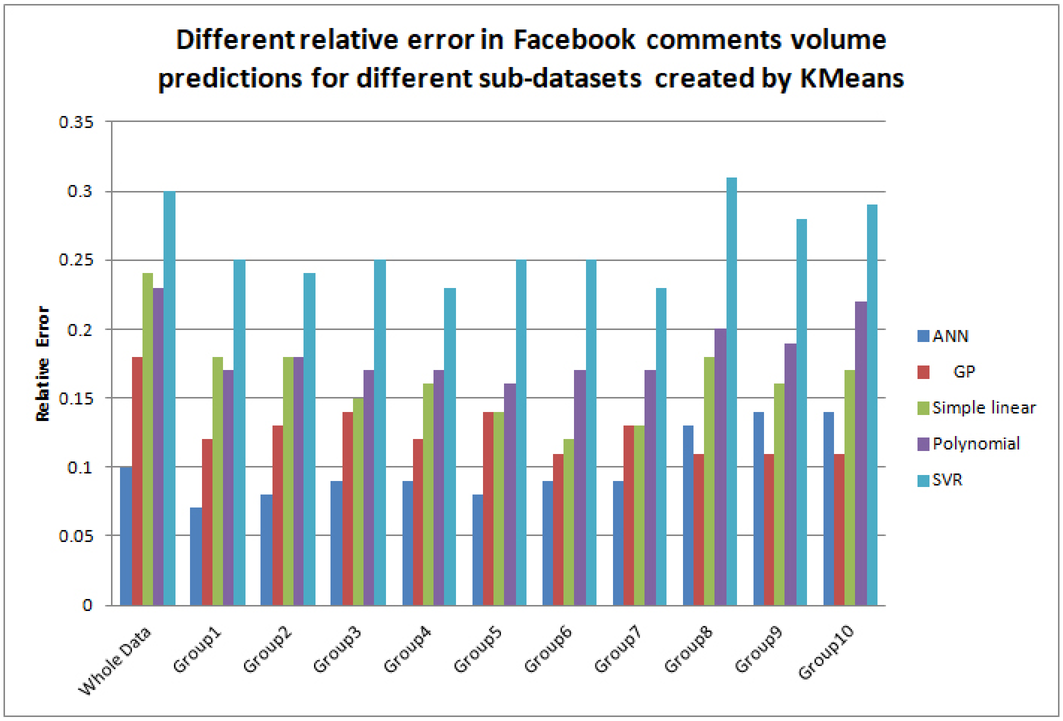


Figure 4.43: Different relative error in Facebook comments volume predictions for different sub-datasets created by K-means .

outcome was one group with 30142 instances and the other groups, group 2 to group 5, have 5317, 2524, 1720 and 1246 members respectively. Table 4.27 and Figure 4.44 compare the prediction results for five sub-datasets which are created by DBSCAn and entire data. SVR predictions has no improvement but other models predictions are more accurate after clustering. Group 4 and 5 do not have large size, the predictions for this clusters are less accurate than the entire dataset, but as group 1 has large number of instances and predictions for this group is more accurate than the whole data, in average, the total prediction after clustering has lower average error. The overall error drops from 0.1 to 0.0768, Comparing this error to total error for K-means clustering which was 0.9, reveals that DBSCAN has better performance for this dataset.

Models	**ANN**	**GP**	**Linear**	**Polynomial**	**SVR**
Whole Data	**0.1**	0.18	0.24	0.23	0.3
Group1	**0.07**	0.11	0.17	0.19	0.26
Group2	**0.08**	0.13	0.18	0.17	0.25
Group3	**0.1**	0.15	0.18	0.22	0.27
Group4	0.15	**0.12**	0.19	0.22	0.29
Group5	0.16	**0.12**	0.23	0.25	0.31
average	**0.078**	0.115	0.173	0.194	0.261
Overall	0.0768				

Table 4.27: Different relative error in Facebook comments volume predictions for different sub-datasets created by DBSCAN. group 1 to group 5, have 30142, 5317, 2524, 1720 and 1246 instances respectively. If we select the best model for each group, the overall error is 0.0768 .

Models:	**ANN**	**GP**	**Linear**	**Polynomial**	**SVR**	**overall**
Whole Dataset	**0.1**	0.18	0.24	0.23	0.3	0.1
K-means	**0.09**	0.127	0.17	0.18	0.261	0.09
DBSCAN	**0.078**	0.115	0.173	0.194	0.261	0.078

Table 4.28: Comparison between different clustering methods effects on Facebook comments volume prediction.

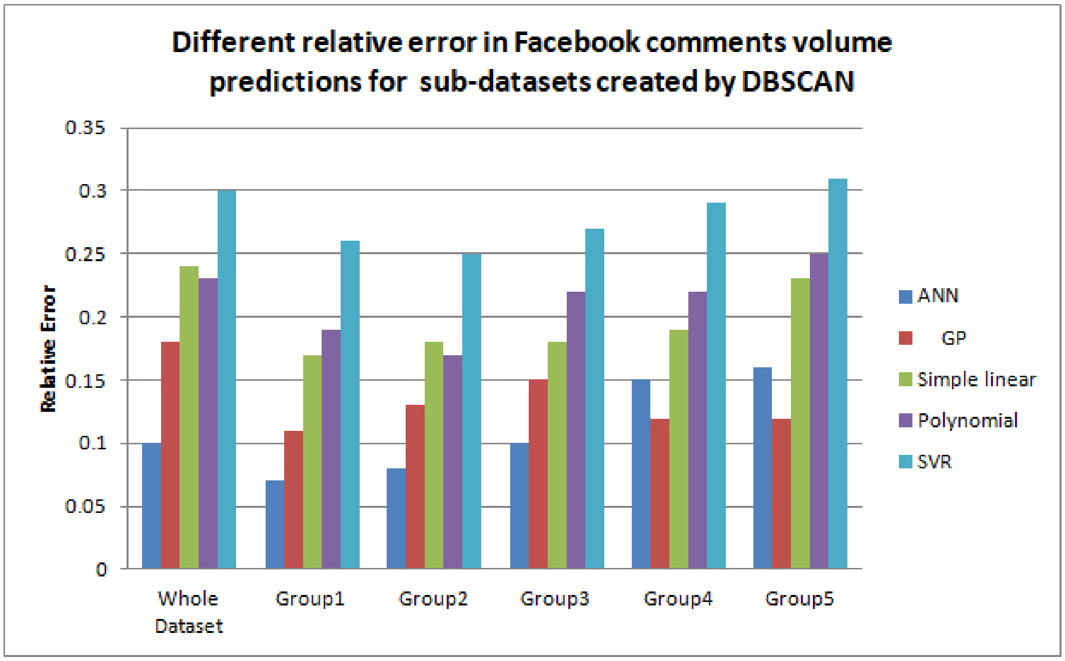


Figure 4.44: Different relative error in Facebook comments volume predictions for different sub-datasets created by DBSCAN.

4.3.3.3 Facebook Comments Volume Prediction Results

The summary of results is presented in Table 4.28 and Figure 4.45. They show how ANN has good performance when we have a big number of instances in dataset. Again, we observe that GP is more accurate comparing to other predictors except ANN. In addition, Comparing Table 4.25 and Figure 4.41 with Table 4.26 and Figure 4.43 and Table 4.27 and Figure 4.44, indicates that ANN error improves when data size decrease. In addition, Polynomial model is more sensitive to the data size comparing to SVR and a simple linear model. SVR has the lowest variation in accuracy when the data size decreases.

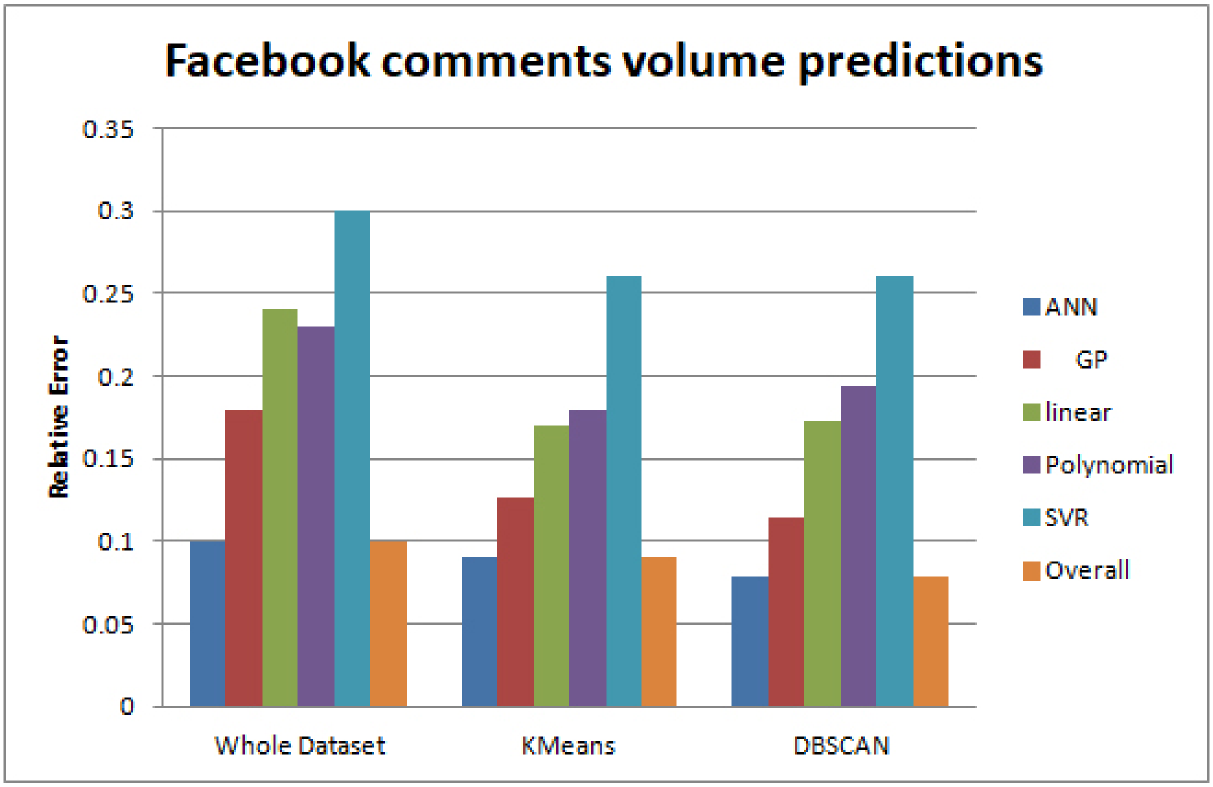


Figure 4.45: Comparison between different clustering and grouping methods effects on Facebook comments volume prediction.

4.3.4 Comparison of Three Datasets Experimental Results

Our experiments results demonstrate that the proposed method can be efficient for large size datasets. We can compare effect of differnt clustering and grouping techniques on three examined datasets; house price dataset, water evaporation data and the Facebook comments volume dataset. Table 4.29 presents the summary of the best prediction results for each dataset.

Table 4.29 shows that for clustering differnt datasets which have different structures, we need to evaluate different clustering techniques and we can not apply one clustering method to all types of datasts. More importantly, it affirms that our approach is beneficial to improve prediction accuracy for large size datasets that we examined. In the House price

Clustering method:	**Whole Dataset**	**K-means**	**DBSCAN**	**HAC**	**Grouping**
House price	0.14	0.087	0.129	0.172	0.086
Water evaporation	0.08	0.111	-	0.131	0.1235
Facebook comments	0.1	0.09	0.078	-	-

Table 4.29: Comparison between different clustering or grouping methods effects on different datasets predictions. For grouping columns, we selected the best results for grouping data based on the experts knowledge. HAC did not have acceptable performance for the Facebook dataset and DBSCAN did not perform well for Water evaporation data. We Do not have any record for expert knowledge grouping for the Facebook dataset.

prediction for the best model, the error drops from 0.14 to 0,086. The Facebook dataset shows the decrease in error from 0.1 to 0.78.

4.4 Summary

- Based on the nature of different datasets, different clustering methods may perform better for each dataset. For example, for house price data, Kmean can create proper groups and for Facebook data, DBSCAN results in better clusters.

- Our proposed method can reduce error for large datasets which after clustering, created sub-datasets can provide enough training data to fit a model with acceptable accuracy.

- Rather than clustering, we can utilize grouping based on important features based on the human knowledge. The resulted sub-groups from this approach, have good fit on linear models.

- ANN and polynomial models accuracy is very sensitive to the data size and their accuracy drops dramatically by reducing the number of samples in each sub-group. In contrast, in small size created sub-datasets GP performs better than other predictive models.

Chapter 5

Conclusion Remarks and Future Works

Improving the prediction accuracy is the goal of many machine learning researches. We proposed an ensemble-based model to increase the prediction accuracy for large size datasets. The experiments and results are discussed in chapter four and in following chapter based on our results and experiments we provide the conclusion and future work directions.

5.1 Conclusion Remarks

In this study, we modelled our proposed method and experiments to examine the hypothesis that is provided in section 1.3. It proposes that clustering data can improve the prediction accuracy because we specialize the model for the selected similar samples in each cluster. As demonstrated in predictions for randomly selected sub-datasets, Tables 4.1, 4.20 and 4.41, when we split data, the accuracy decreases because we reduce the size of training data. Therefore, when we cluster data and create sub-datasets with a lower number of instances, the prediction models are not only expected to compensate this decreasing in accuracy, but are also expected to improve the accuracy compared to the results of predictions on the whole dataset. We provided background and a literature review of machine learning models for clustering, regression and prediction and symbolic regression, which we utilized in our proposed method. In addition, we reviewed the similar works to our proposed model in chapter 2. Then, we introduced our proposed approach details in methodology section,3.1. Next, we expressed the setting of the using techniques in sections 4.1 and 4.2. The experiments in section 4.3, examine the hypothesis. We studied several prediction models for different clustering and grouping methods for three datasets.

House price prediction

House price Prediction results are illustrated in the Table 4.19 and Figure 4.24 for the first dataset. Before clustering, neural network is the most accurate model with the error equal to 0.14 in price prediction. Therefore, if our method can predict with lower error, this confirms our hypothesis. Hierarchical Agglomerative clustering was not successful in improving accuracy compared to applying prediction models to the whole data. But compared to randomly selected sub-datasets it has a better performance and demonstrates the effect of similarity between the samples in each sub-dataset. DBSCAN somewhat enhanced prediction for ANN that decreased the error from 0.14 to 0.13 and overall to 0.129, which was not a remarkable improvement. The most successful clustering technique in our method is K-means, which decreases the overall error to 0.087. It reveals that K-means can detect similarity between data points in house price dataset. The only problem with K-means is its computational time that makes it an expensive technique especially when the dataset includes a large number of instances and has too many features. Therefore, if we have time limitation, using K-means in our method, may not be effective. Based on the results which show lower error after applying K-means, we can conclude that K-means which select similar data points based on their distances to the other data points, has acceptable performance for this dataset even if it has too many features. For house price data, DBSCAN could not improve the average accuracy, but it confirms that in small sub-dataset, we can trust GP to predict more accurate than other models. Moreover, weak performance of HAC assert that all clustering techniques can not be effective for our proposed method. Tables 4.1to 4.13 and Figures 4.8 to 4.22, represent the result of applying different models on the sub-datasets of house price prediction. As shown, when we apply different models to all of the data, neural network technique has better performance and offers a lower error in prediction. It reveals that the ANN model can be trained effectively when we have a sufficient number of samples. But when the number of instances is small, creating an appropriate model fails, while, even with a small number of samples, GP can still generate a model which both fits training data and predicts the test data with acceptable accuracy. Furthermore, it is surprising that simple linear model is less sensitive to the number of training data and linear predictions in grouping based on human knowledge, which we consider one or two features for grouping and adding boarder members to the sub-groups, have better performance compared to machine learning clustering methods. This grouping scheme is faster than clustering techniques, but we need to examine all combinations of important features with a different number of members in created groups and this task is time consuming and computationally expensive. However, GP has larger overall error than some other models. It is surprising that GP presents a somewhat better performance in prediction when samples have similarity. When we randomly create smaller datasets by decreasing the number of samples, this increases the error in all models. Polynomial and neural network regressions are apparently very sensitive to the sample size. Consequently, as GP can perform better in small size datasets compared to other methods, in sub-datasets with a low number of instances, we can rely on GP predictions.

Pan evaporation prediction

The second dataset, agricultural data, can be considered a relatively small dataset compared to two other datasets that we studied, which includes both a lower number of instances and features.

The predictions results in Table 4.24 and Figure 4.34 assert that our proposed method is not useful for small datasets because when we split data into small sub-datasets, the resulting clusters and groups do not contain a sufficient number of data points for training a model accurately. The only surprising case is GP predictions for K-means clustering, which is similar to what we observed in house price data, that GP was successful compared to other models in small sub-datasets. In addition, after HAC clustering, GP performs better than other models. Moreover, it can once again be concluded from Table 4.34 and Figure 4.34, that SVR is not very sensitive to data size for this data and the same as our results for house price data, GP has the lowest change after clustering and grouping.

Facebook comments volume prediction

Considering the failure of our proposed scheme on a small dataset, a large dataset, Facebook comments volume, seems appropriate to examine our proposed method. As we expected, Table 4.28 and Figure 4.45 show how ANN performs very accurate when a dataset includes a large number of instances, even though we have a large number of features in the Facebook comments dataset. It is not surprising that in this case, neural network can perform very well; however, considering the error values in Table 4.28 and Figure 4.45, it is surprising that symbolic regression using GP, is more accurate compared to SVR, linear and Polynomial regressions. By comparing Table 4.25 and Figure 4.41 with Table 4.26 and Figure 4.43 and Table 4.27 and Figure 4.44, indicates once again how ANN accuracy decreases noticeably with sample size and confirms the GP's superiority that overcomes the conventional regression models in these cases. The most surprising outcome is that DBSCAN provides the best results even though we expected a weak performance for DBSCAN in high dimensional datasets.

Conclusion Generally speaking, we can conclude that K-means, which select similar data points based on their distances to the other data points, has acceptable performance for some datasets such as house price data, and even has 20 features. On the other hand, for different datasets based on the nature and structure of the data, we need to examine different clustering methods. As we observed in the Facebook dataset, even K-means works well, DBSCAN results in better accuracy. Moreover, if we can find the most effective features by using the features correlations to the target variable, or by using expert knowledge, grouping based on one or two features can be as accurate as the machine learning clustering. When we have a large number of features, in most cases linear and polynomial regressions do not fit well with the data. But in grouping schemes based on the data structure and feature correlations, in smaller groups, linear model or polynomial may perform excellent. It means that even if the whole dataset fits with the complicated models, small sub-datasets can be considered as a linear or polynomial model. In these cases, fitting a model and predictions can be easier and faster with acceptable accuracy. In addition, in large sub-datasets, which we have enough training data, neural network is very successful to create accurate model, but when we cannot provide large size training data, GP is powerful in creating models because it does not try to find patterns directly from training data and it initially creates random model and improves it in each generation. Therefore, for a large number of iterations, if we apply a proper mutation and crossover rate, we have the chance to discover an appropriate model. In low size datasets, GP has the problem of over-fitting but if we select a large proportion of the data, (30 to 40 percent) as the test data, we can avoid over-fitting. Even if GP is computationally expensive, it can create symbolic models for regression with reasonable accuracy, especially in cases where we are unable to gather a large volume of instances, it outperforms other models. In addition, in this research, we also provide a comparative study among GP and conventional regression methods, such as linear and polynomial models, support vector regression (SVR) and artificial neural network (ANN). Comparing GP to other common methods, as the evolutionary algorithms offer effective exploration in the search space, the final solution is a selection of models with high fitness value and their set of parameters. Therefore, it has the potential of providing an accurate regression. If we compare the GP's accuracy in the datasets, which include approximately the same number of instances, but differ in number of variables, as it is expected, regression accuracy decreases by increasing the number of features. Although, its accuracy is lower than high dimensional data, it can still perform more accurately in comparison with other models when we do not have enough samples. In addition, GP creates a understandable model (formula), which can be easily utilized in compact embedded systems. Neural network is very powerful in detecting patterns and trends in complicated, or imprecise data. Generally, intricate patterns can not be discovered by human analysis or other computer algorithms. As it works like the human brain and needs to be trained, similar to human experts, its success is very dependent on the volume of relevant training data and as we demonstrated in Tables 4.1, 4.20 and 4.41, by decreasing the data size its accuracy drops dramatically. Our experiments verified our hypothesis for large datasets that if we split data in sub-datasets based on similarities between each group, or cluster members, most models especially neural network, can create more accurate models. In addition, if we take the advantage of differnt models in each sub-dataset and select the best, overall we can reduce the prediction error. Moreover, in the small sub-datasets, we can train data with the whole data to improve the total accuracy.

The other advantage of our proposed method is that customizing models for smaller sub-datasets with more similar samples comparing to the whole data, not only improves accuracy, but is also may decrease the computational complexity because a model, which is fitted to the entire data is more complicated than customized models. Moreover, after clustering, predictions for sub-dataset can be done in parallel and prediction in a small size data is faster.

5.2 Future Works

While our prediction method demonstrated higher accuracy in predictions compared to applying the models to whole data, there are still several cases and schemes that we would like to consider for future research.

Different clustering methods

As mentioned in section 2.2.2, there are several clustering methods and none of them can work efficiently for all types of datasets. Therefore, if we examine more clustering methods, we may be able to improve the prediction accuracy. Especially for high dimensional datasets, other than conventional clustering techniques, deep embedded clustering (DEC), [84], [85], [86], can be beneficial in creating sub-datasets with high similarity. DEC is able to employ deep neural network to learn both deep feature representations and cluster placement simultaneously. DEC maps the data space into a low-dimension feature space and in each iteration optimizes the clustering objectives. This method has the potential to improve both accuracy and computational cost as it can be faster and more accurate in clustering high dimensional datasets compared to conventional methods such as K-means and DBSCAN.

Other regression and prediction methods

We applied more than ten regression models and schemes and selected the best with lower accuracy for our experiments. But there are many more methods that could be examined such as step wise regression [87], [88]. In this technique, we fit prediction models step by step by choosing predictive variables. In each step, considering pre-specified rules, a term belongs to selected variable is added or subtracted from the set of variables.

Other datasets

In this study we examined our proposed method for three datasets. Middle size data with around 21,000 instances and 20 features, small size data with approximately 7,000 samples and 10 variables and a large dataset with more than 40,000 instances and 54 features. It is not possible to find a model or regression method, which performs well for all types of datasets, but if we apply our data on different types of datasets, we can figure out that this proposed method can work accurately for what type of data with what kind of characteristics.

Using optimization methods in clustering

In many cases, in clustering we can use optimization methods to find the best clusters. For instance, as Figure 4.29 illustrates, in clustering the second dataset, by increasing the number of clusters, we obtain better clustering score. On the other hand, by increasing the number of clusters we have less number of members in each cluster and we loose accuracy. Therefore, these two objectives, clustering score and accuracy, are in conflict. Using multi-objective optimization, we can find a pareto-front of the best points, which are best solutions for both score and accuracy.

End result aggregation

We obtained one best model for each cluster, as a general result, we can consider the aggregation of the all models. Different models have differnt accuracy we can consider the weighted summation of all models.

Appendix A

A.1 Datasets

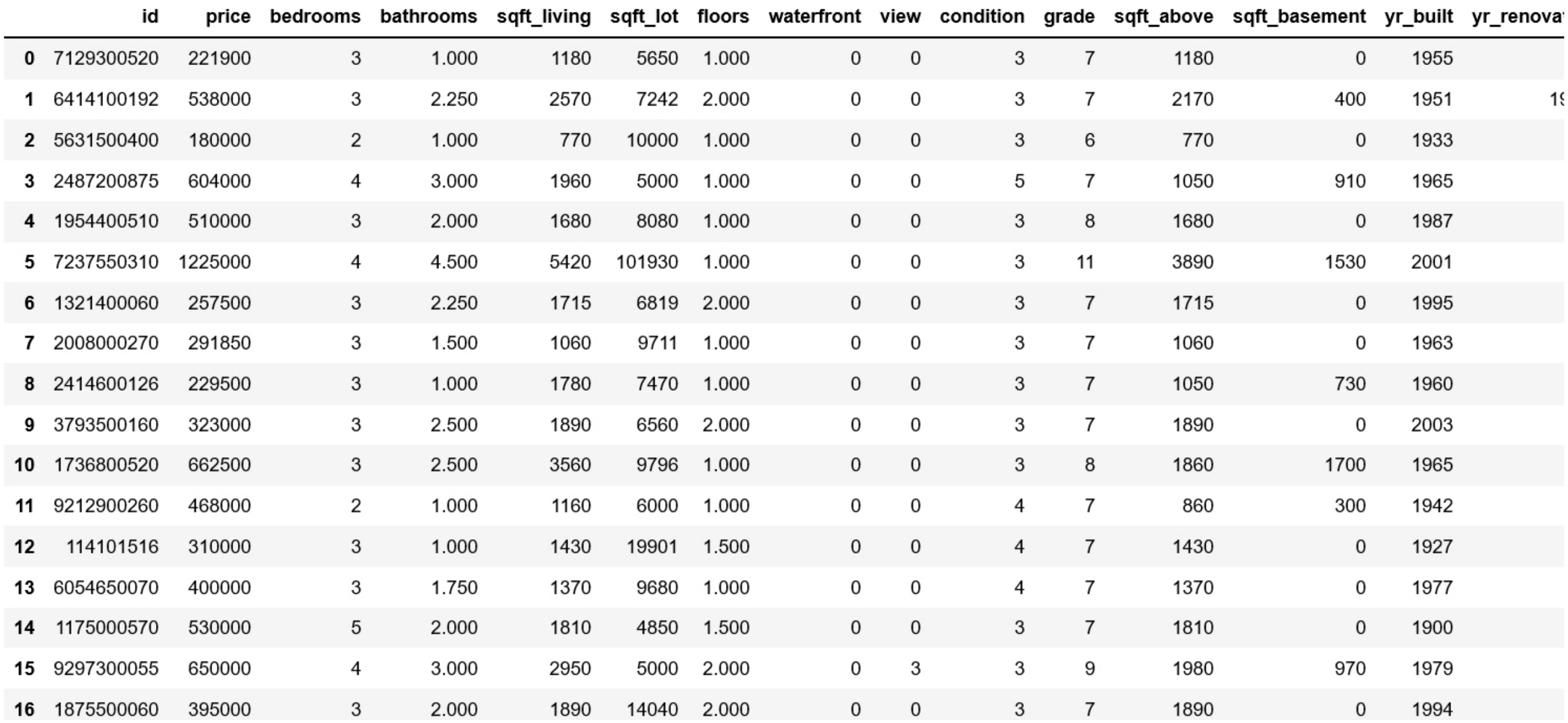
A.1.1 House Price Dataset

House sales data for King County in the USA, including 21,614 instances and 20 features Data is downloaded from Kaggle [12] which is a multivariate dataset with both real and integer values with no sparsity.

The variables are: Id, price, number of bedrooms, number of bathrooms, sqft-living, sqft-lot, floors (number of floors), waterfront, view, condition, grade, sqft-above, sqft-basement, yr-built, yr-renovated, zipcode, latitude, longitude, sqft-living15, sqft-lot15.

A.1.2 Water Evaporation Dataset

This is agricultural dataset, contains 10 features and 7549 samples. This is multivariate dataset which includes categorical, integer, and real variables with no sparsity [13] and [83]. The features are: year, Julian day, maximum temperature, maximum temperature, relative humidity 1, relative humidity 2, rainfall(mm), wind speed(km/hr), sunshine(hours), Pan



	id	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	grade	sqft_above	sqft_basement	yr_built	yr_renova
0	7129300520	221900	3	1.000	1180	5650	1.000	0	0	3	7	1180	0	1955	
1	6414100192	538000	3	2.250	2570	7242	2.000	0	0	3	7	2170	400	1951	19
2	5631500400	180000	2	1.000	770	10000	1.000	0	0	3	6	770	0	1933	
3	2487200875	604000	4	3.000	1960	5000	1.000	0	0	5	7	1050	910	1965	
4	1954400510	510000	3	2.000	1680	8080	1.000	0	0	3	8	1680	0	1987	
5	7237550310	1225000	4	4.500	5420	101930	1.000	0	0	3	11	3890	1530	2001	
6	1321400060	257500	3	2.250	1715	6819	2.000	0	0	3	7	1715	0	1995	
7	2008000270	291850	3	1.500	1060	9711	1.000	0	0	3	7	1060	0	1963	
8	2414600126	229500	3	1.000	1780	7470	1.000	0	0	3	7	1050	730	1960	
9	3793500160	323000	3	2.500	1890	6560	2.000	0	0	3	7	1890	0	2003	
10	1736800520	662500	3	2.500	3560	9796	1.000	0	0	3	8	1860	1700	1965	
11	9212900260	468000	2	1.000	1160	6000	1.000	0	0	4	7	860	300	1942	
12	114101516	310000	3	1.000	1430	19901	1.500	0	0	4	7	1430	0	1927	
13	6054650070	400000	3	1.750	1370	9680	1.000	0	0	4	7	1370	0	1977	
14	1175000570	530000	5	2.000	1810	4850	1.500	0	0	3	7	1810	0	1900	
15	9297300055	650000	4	3.000	2950	5000	2.000	0	3	3	9	1980	970	1979	
16	1875500060	395000	3	2.000	1890	14040	2.000	0	0	3	7	1890	0	1994	

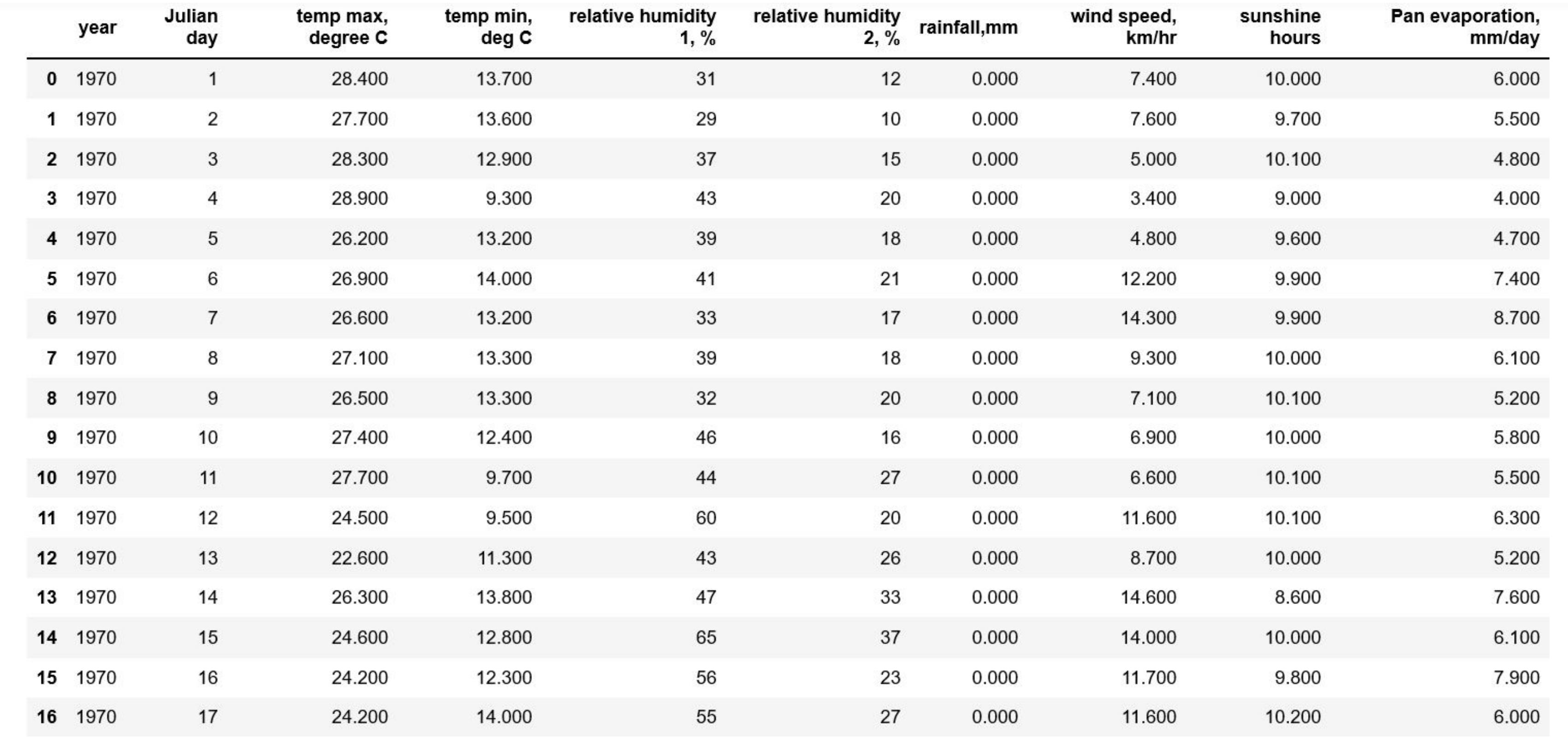
Figure A.1: Sample of house price prediction dataset .

evaporation(mm/day).

A.1.3 Facebook Comments Volume Dataset

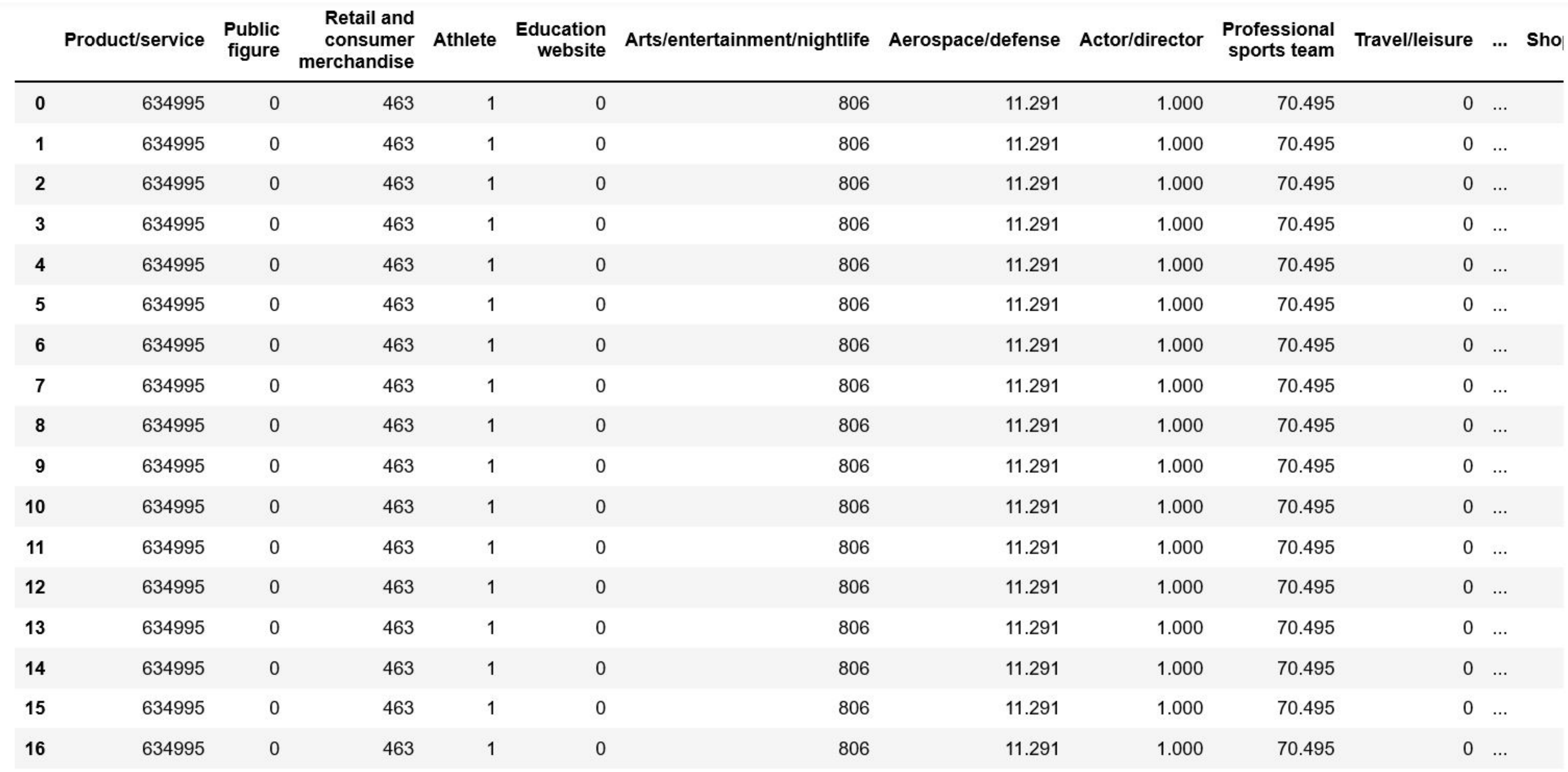
The Facebook Comment Volume dataset with 40,949 samples and 54 features includes integer, and real variables with no sparsity [14].

The features are: Product/service, Public figure, Retail and consumer, merchandise, Athlete, Education website, Arts/entertainment/nightlife, Aerospace/defense, Actor/director, Professional sports team, Travel/leisure, Arts/humanities website, Food/beverages, Record label, Movie, Song, Community, Company, Artist, Non-governmental organization (ngo), Media/news/publishing, Cars, Clothing, Local business, Musician/band, Politician, News/ media website, Education, Author, Sports event, Restaurant/cafe, School sports team, University, Tv show, Website, Outdoor gear/sporting goods, Political party, Sports league, Entertainer, Church/religious organization, Non-profit organization, Automobiles and parts, Tv channel, Telecommunication, Entertainment website, Shopping/retail, Personal blog, App page, Vitamins/supplements, Professional services, Movie, theater, Software, Magazine, Electronics, School, Just for fun, Club, Comedian, Sports venue, Sports/recreation/ activities, Publisher, Tv network, Health/medical/pharmacy, Studio, Home decor, Jewelry/watches, Writer, Health/beauty, Music video, Appliances, Computers/technology, Insurance company, Music award, Recreation/sports website, Reference website, Games/toys, Camera/ photo, Book, Producer, Landmark, Cause, Organization, Tv/movie award, Hotel, Health/medical/ pharmaceuticals, Transportation, Local/travel website, Musical instrument, Radio station, Other, Computers, Phone/tablet, Coach, Tools/equipment, Internet/software, Bank/financial institution, Society/culture, website, Small business, News personality, Teens/kids website, Government official, Photographer, Spas/beauty/personal care, Video game.



	year	Julian day	temp max, degree C	temp min, deg C	relative humidity 1, %	relative humidity 2, %	rainfall,mm	wind speed, km/hr	sunshine hours	Pan evaporation, mm/day
0	1970	1	28.400	13.700	31	12	0.000	7.400	10.000	6.000
1	1970	2	27.700	13.600	29	10	0.000	7.600	9.700	5.500
2	1970	3	28.300	12.900	37	15	0.000	5.000	10.100	4.800
3	1970	4	28.900	9.300	43	20	0.000	3.400	9.000	4.000
4	1970	5	26.200	13.200	39	18	0.000	4.800	9.600	4.700
5	1970	6	26.900	14.000	41	21	0.000	12.200	9.900	7.400
6	1970	7	26.600	13.200	33	17	0.000	14.300	9.900	8.700
7	1970	8	27.100	13.300	39	18	0.000	9.300	10.000	6.100
8	1970	9	26.500	13.300	32	20	0.000	7.100	10.100	5.200
9	1970	10	27.400	12.400	46	16	0.000	6.900	10.000	5.800
10	1970	11	27.700	9.700	44	27	0.000	6.600	10.100	5.500
11	1970	12	24.500	9.500	60	20	0.000	11.600	10.100	6.300
12	1970	13	22.600	11.300	43	26	0.000	8.700	10.000	5.200
13	1970	14	26.300	13.800	47	33	0.000	14.600	8.600	7.600
14	1970	15	24.600	12.800	65	37	0.000	14.000	10.000	6.100
15	1970	16	24.200	12.300	56	23	0.000	11.700	9.800	7.900
16	1970	17	24.200	14.000	55	27	0.000	11.600	10.200	6.000

Figure A.2: Sample of water evaporation prediction dataset .



	Product/service	Public figure	Retail and consumer merchandise	Athlete	Education website	Arts/entertainment/nightlife	Aerospace/defense	Actor/director	Professional sports team	Travel/leisure	...	Sho
0	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
1	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
2	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
3	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
4	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
5	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
6	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
7	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
8	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
9	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
10	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
11	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
12	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
13	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
14	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
15	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	
16	634995	0	463	1	0	806	11.291	1.000	70.495	0	...	

Figure A.3: Sample of Facebook comments volume prediction dataset.

Bibliography

[1] T. Bayes, "Lii. an essay towards solving a problem in the doctrine of chances. by the late rev. mr. bayes, frs communicated by mr. price, in a letter to john canton, amfr s," *Philosophical transactions of the Royal Society of London*, no. 53, pp. 370–418, 1763.

[2] A. M. Legendre, *Nouvelles méthodes pour la détermination des orbites des comètes*. F. Didot, 1805.

[3] C. F. Gauss, *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*, vol. 7. Perthes et Besser, 1809.

[4] C.-F. Gauss, *Theoria combinationis observationum erroribus minimis obnoxiae*, vol. 1. Henricus Dieterich, 1823.

[5] M. Kantardzic, "Data mining concepts, models, methods, and algorithms. a john wiley & sons," *Inc., Chichester*, 2003.

[6] R. Kohavi and F. Provost, "Glossary of terms: Machine learning," *30: 271*, vol. 274, 1998.

[7] J. McCarthy and E. A. Feigenbaum, "In memoriam: Arthur samuel: Pioneer in machine learning," *AI Magazine*, vol. 11, no. 3, pp. 10–10, 1990.

[8] "European public real estate association." `http://alturl.com/7snxx`.

[9] P. Bajari, D. Nekipelov, S. P. Ryan, and M. Yang, "Machine learning methods for demand estimation," *American Economic Review*, vol. 105, no. 5, pp. 481–85, 2015.

[10] V. Moosavi, "Urban data streams and machine learning: a case of swiss real estate market," *arXiv preprint arXiv:1704.04979*, 2017.

[11] R. E. Schapire, P. Stone, D. McAllester, M. L. Littman, and J. A. Csirik, "Modeling auction price uncertainty using boosting-based conditional density estimation," in *ICML*, pp. 546–553, 2002.

[12] "House sales data set in king county, usa." `https://www.kaggle.com/harlfoxem/housesalesprediction/version/1`, May 2014-2015.

[13] P. P. Adhikary, D. Chakraborty, N. Kalra, C. Sachdev, A. Patra, S. Kumar, R. Tomar, P. Chandna, D. Raghav, and K. Agrawal, "Pedotransfer functions for predicting the hydraulic properties of indian soils," *Soil research*, vol. 46, no. 5, pp. 476–484, 2008.

[14] K. Singh, R. K. Sandhu, and D. Kumar, "Comment volume prediction using neural networks and decision trees," in *IEEE UKSim-AMSS 17th International Conference on Computer Modelling and Simulation, UKSim2015 (UKSim2015)*, 2015.

[15] P. Berkhin, "A survey of clustering data mining techniques," in *Grouping multidimensional data*, pp. 25–71, Springer, 2006.

[16] P. Berkhin, "Survey of clustering data mining techniques. technical report," in *Accrue software*, 2002.

[17] "Scikit-learn." `https://scikit-learn.org`.

[18] J. A. Hartigan and M. A. Wong, “Algorithm as 136: A k-means clustering algorithm,” *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[19] H. Steinhaus, “Sur la division des corp materiels en parties,” *Bull. Acad. Polon. Sci*, vol. 1, no. 804, p. 801, 1956.

[20] S. Lloyd, “Least squares quantization in pcm,” *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[21] E. W. Forgy, “Cluster analysis of multivariate data: efficiency versus interpretability of classifications,” *biometrics*, vol. 21, pp. 768–769, 1965.

[22] J. MacQueen *et al.*, “Some methods for classification and analysis of multivariate observations,” in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, pp. 281–297, Oakland, CA, USA, 1967.

[23] P. Bholowalia and A. Kumar, “Ebk-means: A clustering technique based on elbow method and k-means in wsn,” *International Journal of Computer Applications*, vol. 105, no. 9, 2014.

[24] P. J. Rousseeuw, “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis,” *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[25] M. Ester and Kriegel, “A density-based algorithm for discovering clusters in large spatial databases with noise.,” in *Kdd*, vol. 96, pp. 226–231, 1996.

[26] “Medium corporation website, dbscan method.” `https://medium.com/@elutins/dbscanhow-to-use-it-8bd506293818`.

[27] L. Rokach and O. Maimon, "Clustering methods," in *Data mining and knowledge discovery handbook*, pp. 321–352, Springer, 2005.

[28] J. H. Ward Jr, "Hierarchical grouping to optimize an objective function," *Journal of the American statistical association*, vol. 58, no. 301, pp. 236–244, 1963.

[29] M. Okada, "Crpclustering: An r package for bayesian nonparametric chinese restaurant process clustering with entropy," tech. rep., PeerJ Preprints, 2018.

[30] J. Lu, M. Li, and D. Dunson, "Reducing over-clustering via the powered chinese restaurant process," *arXiv preprint arXiv:1802.05392*, 2018.

[31] F. Santosa and W. W. Symes, "Linear inversion of band-limited reflection seismograms," *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 4, pp. 1307–1330, 1986.

[32] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[33] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[34] H. Byun and S.-W. Lee, "Applications of support vector machines for pattern recognition: A survey," in *International Workshop on Support Vector Machines*, pp. 213–236, Springer, 2002.

[35] M. Aiserman, E. M. Braverman, and L. Rozonoer, "Theoretical foundations of the potential function method in pattern recognition," *Avtomat. i Telemeh*, vol. 25, pp. 917–936, 1964.

[36] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943.

[37] H. White, *Artificial neural networks: approximation and learning theory*. Blackwell Publishers, Inc., 1992.

[38] J. Misra and I. Saha, “Artificial neural networks in hardware: A survey of two decades of progress,” *Neurocomputing*, vol. 74, no. 1-3, pp. 239–255, 2010.

[39] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, “State-of-the-art in artificial neural network applications: A survey,” *Heliyon*, vol. 4, no. 11, p. e00938, 2018.

[40] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, “A survey of deep neural network architectures and their applications,” *Neurocomputing*, vol. 234, pp. 11–26, 2017.

[41] C. Zhang, “Genetic programming for symbolic regression,” *University of Tennesse, Knoxville, TN*, vol. 37996.

[42] J. R. Koza, “Genetic programming as a means for programming computers by natural selection,” *Statistics and computing*, vol. 4, no. 2, pp. 87–112, 1994.

[43] J. Koza, M. A. Keane, and J. P. Rice, “Performance improvement of machine learning via automatic discovery of facilitating functions as applied to a problem of symbolic system identification,” in *IEEE International Conference on Neural Networks*, pp. 191–198, IEEE, 1993.

[44] H. Iba, H. deGaris, and T. Sato, “A numerical approach to genetic programming for system identification,” *Evolutionary Computation*, vol. 3, no. 4, pp. 417–452, 1995.

[45] D. J. Montana, “Strongly typed genetic programming,” *Evolutionary computation*, vol. 3, no. 2, pp. 199–230, 1995.

[46] C. Sian and Fuey, “A java based distributed approach to genetic programming on the internet,” *Master’s thesis. Computer Science, University of Birmingham. Submachine-code Genetic Programming*, vol. 323, 1998.

[47] D. A. Augusto and H. J. Barbosa, “Symbolic regression via genetic programming,” in *Neural Networks, 2000. Proceedings. Sixth Brazilian Symposium on*, pp. 173–178, IEEE, 2000.

[48] E. K. Burke, S. Gustafson, and G. Kendall, “Diversity in genetic programming: An analysis of measures and correlation with fitness,” *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 1, pp. 47–62, 2004.

[49] J. M. Daida, D. J. Ward, A. M. Hilss, S. L. Long, M. R. Hodges, and J. T. Kriesel, “Visualizing the loss of diversity in genetic programming.,” in *IEEE Congress on Evolutionary Computation*, pp. 1225–1232, 2004.

[50] M.-J. Willis, H. G. Hiden, P. Marenbach, B. McKay, and G. A. Montague, “Genetic programming: An introduction and survey of applications,” in *Genetic Algorithms in Engineering Systems: Second International Conference On Innovations and Applications*, pp. 314–319, IET, 1997.

[51] S. Gustafson, E. K. Burke, and N. Krasnogor, “On improving genetic programming for symbolic regression,” in *The 2005 IEEE Congress on Evolutionary Computation*, vol. 1, pp. 912–919, IEEE, 2005.

[52] B. Worzel and R. Riolo, “Genetic programming: theory and practice,” in *Genetic Programming Theory and Practice*, pp. 1–10, Springer, 2003.

[53] E. Y. Vladislavleva *et al.*, *Model-based problem solving through symbolic regression via pareto genetic programming*. Citeseer, 2008.

[54] M. E. Kotanchek, E. Y. Vladislavleva, and G. F. Smits, "Symbolic regression via genetic programming as a discovery engine: Insights on outliers and prototypes," in *Genetic Programming Theory and Practice VII*, pp. 55–72, Springer, 2010.

[55] J. R. Koza, *Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems*, vol. 34. Stanford University, Department of Computer Science Stanford, CA, 1990.

[56] J. R. Koza, "Evolution of subsumption using genetic programming," in *Proceedings of the First European Conference on Artificial Life*, pp. 110–119, 1992.

[57] Eggermont and Jeroen, *Data mining using genetic programming: Classification and symbolic regression*. Institute for Programming research and Algorithmics, Leiden Institute of , 2005.

[58] F. A. Graybill, *Theory and application of the linear model*. No. QA279. G72 1976., Duxbury press North Scituate, MA, 1976.

[59] R. D. Cook, "Detection of influential observation in linear regression," *Technometrics*, vol. 19, no. 1, pp. 15–18, 1977.

[60] D. A. Ratkowsky and D. E. Giles, "Handbook of nonlinear regression models," 1990.

[61] R. H. Myers and R. H. Myers, *Classical and modern regression with applications*, vol. 2. Duxbury press Belmont, CA, 1990.

[62] M.-S. Chen, J. Han, and P. S. Yu, "Data mining: an overview from a database perspective," *IEEE Transactions on Knowledge and data Engineering*, vol. 8, no. 6, pp. 866–883, 1996.

[63] E. Pagourtzi, V. Assimakopoulos, T. Hatzichristos, and N. French, "Real estate appraisal: a review of valuation methods," *Journal of Property Investment & Finance*, vol. 21, no. 4, pp. 383–401, 2003.

[64] P. M. Anglin and R. Gencay, "Semiparametric estimation of a hedonic price function," *Journal of Applied Econometrics*, vol. 11, no. 6, pp. 633–648, 1996.

[65] R. A. Dubin, "Predicting house prices using multiple listings data," *The Journal of Real Estate Finance and Economics*, vol. 17, no. 1, pp. 35–59, 1998.

[66] R. Li and H. Li, "Have housing prices gone with the smelly wind? big data analysis on landfill in hong kong," *Sustainability*, vol. 10, no. 2, p. 341, 2018.

[67] C. Bagnoli and H. Smith, "The theory of fuzz logic and its application to real estate valuation," *Journal of Real Estate Research*, vol. 16, no. 2, pp. 169–200, 1998.

[68] B. Case, J. Clapp, R. Dubin, and M. Rodriguez, "Modeling spatial and temporal house price patterns: A comparison of four models," *The Journal of Real Estate Finance and Economics*, vol. 29, no. 2, pp. 167–191, 2004.

[69] Y. Demyanyk and I. Hasan, "Financial crises and bank failures: A review of prediction methods," *Omega*, vol. 38, no. 5, pp. 315–324, 2010.

[70] S. B. Imandoust and M. Bolandraftar, "Application of k-nearest neighbor (knn) approach for predicting economic events: Theoretical background," *International Journal of Engineering Research and Applications*, vol. 3, no. 5, pp. 605–610, 2013.

[71] H. A. Guvenir and I. Uysal, "Regression on feature projections," *Knowledge-Based Systems*, vol. 13, no. 4, pp. 207–214, 2000.

[72] H. Selim, “Determinants of house prices in turkey: Hedonic regression versus artificial neural network,” *Expert systems with Applications*, vol. 36, no. 2, pp. 2843–2852, 2009.

[73] O. Poursaeed, T. Matera, and S. Belongie, “Vision-based real estate price estimation,” *Machine Vision and Applications*, vol. 29, no. 4, pp. 667–676, 2018.

[74] Q. You, R. Pang, L. Cao, and J. Luo, “Image-based appraisal of real estate properties,” *IEEE Transactions on Multimedia*, vol. 19, no. 12, pp. 2751–2759, 2017.

[75] M. Efroymson, “Multiple regression analysis, w: Ralston a., wilf hs (eds), mathematical models for digital computers,” 1960.

[76] G. Bekoulis, J. Deleu, T. Demeester, and C. Develder, “An attentive neural architecture for joint segmentation and parsing and its application to real estate ads,” *Expert Systems with Applications*, vol. 102, pp. 100–112, 2018.

[77] J. Soni, U. Ansari, D. Sharma, and S. Soni, “Predictive data mining for medical diagnosis: An overview of heart disease prediction,” *International Journal of Computer Applications*, vol. 17, no. 8, pp. 43–48, 2011.

[78] J. Quackenbush, “Computational genetics: computational analysis of microarray data,” *Nature reviews genetics*, vol. 2, no. 6, p. 418, 2001.

[79] G. Wang, J. Hao, J. Ma, and L. Huang, “A new approach to intrusion detection using artificial neural networks and fuzzy clustering,” *Expert systems with applications*, vol. 37, no. 9, pp. 6225–6232, 2010.

[80] “Scikit-learn.” `http://scikit-learn.sourceforge.net`.

[81] D. Arthur and S. Vassilvitskii, “k-means++: The advantages of careful seeding,” in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1027–1035, Society for Industrial and Applied Mathematics, 2007.

[82] M. Schmidt and H. Lipson, “Eureqa (version 0.98 beta)[software],” *Nutonian, Somerville, Mass, USA*, 2013.

[83] “Indian meteorological department (imd) website.” `www.imd.gov.in`.

[84] J. Xie, R. Girshick, and A. Farhadi, “Unsupervised deep embedding for clustering analysis,” in *International conference on machine learning*, pp. 478–487, 2016.

[85] X. Guo, L. Gao, X. Liu, and J. Yin, “Improved deep embedded clustering with local structure preservation.,” in *IJCAI*, pp. 1753–1759, 2017.

[86] P. Huang, Y. Huang, W. Wang, and L. Wang, “Deep embedding network for clustering,” in *2014 22nd International Conference on Pattern Recognition*, pp. 1532–1537, IEEE, 2014.

[87] R. Jennrich and P. Sampson, “Application of stepwise regression to non-linear estimation,” *Technometrics*, vol. 10, no. 1, pp. 63–72, 1968.

[88] L. Wilkinson, “Tests of significance in stepwise regression.,” *Psychological Bulletin*, vol. 86, no. 1, p. 168, 1979.

[89] “District data labs website.” `https://medium.com/district-data-labs/machine-learning-33c9c69ef5ec`.

[90] `https://www.researchgate.net/publication/320916953_Radiance_Using_MERRA-2_Atmospheric_Data_with_Deep_Learning`.

[91] "Medium corporation website, towards data science." `https://towardsdatascience.com/polynomial-regression-bbe8b9d97491`.

[92] "Genetic programming." `http://geneticprogramming.com/about-gp/tree-based-gp/`.

[93] P. M. Anglin and R. Gencay, "Semiparametric estimation of a hedonic price function," *Journal of Applied Econometrics*, vol. 11, no. 6, pp. 633–648, 1996.

[94] J. R. Koza, *Genetic programming II, automatic discovery of reusable subprograms*. MIT Press, Cambridge, MA, 1992.

[95] A. Guven and M. Gunal, "Genetic programming approach for prediction of local scour downstream of hydraulic structures," *Journal of Irrigation and Drainage Engineering*, vol. 134, no. 2, pp. 241–249, 2008.

[96] A. Adegboye, M. Kampouridis, and C. G. Johnson, "Regression genetic programming for estimating trend end in foreign exchange market," in *Computational Intelligence (SSCI), 2017 IEEE Symposium Series on*, pp. 1–8, IEEE, 2017.

[97] V. Babovic, M. Keijzer, D. R. Aguilera, and J. Harrington, "An evolutionary approach to knowledge induction: Genetic programming in hydraulic engineering," in *Bridging the Gap: Meeting the World's Water and Environmental Resources Challenges*, pp. 1–10, 2001.

[98] C.-S. Ong, J.-J. Huang, and G.-H. Tzeng, "Building credit scoring models using genetic programming," *Expert Systems with Applications*, vol. 29, no. 1, pp. 41–47, 2005.

[99] I. Boumanchar, Y. Chhiti, F. E. MHamdi Alaoui, A. Sahibed-Dine, F. Bentiss, C. Jama, and M. Bensitel, "Multiple regression and genetic programming for coal higher heating value estimation," *International Journal of Green Energy*, vol. 15, no. 14-15, pp. 958–964, 2018.

[100] D. L. Ly and H. Lipson, "Learning symbolic representations of hybrid dynamical systems," *Journal of Machine Learning Research*, vol. 13, no. Dec, pp. 3585–3618, 2012.

[101] I.-C. Yeh, "Modeling of strength of high-performance concrete using artificial neural networks," *Cement and Concrete research*, vol. 28, no. 12, pp. 1797–1808, 1998.

[102] X. Guo, X. Liu, E. Zhu, and J. Yin, "Deep clustering with convolutional autoencoders," in *International Conference on Neural Information Processing*, pp. 373–382, Springer, 2017.

[103] "Medium corporation website, hac method." `https://towardsdatascience.com/hierarchical-clustering-c6e8243758`.