University of Nottingham
UK | CHINA | MALAYSIA

Final Report: Prediction with Incomplete Data, with application to Automated Valuation Models

Jaesung Park

Shyjp1@nottingham.edu.cn

Supervised by Anthony Graham Bellotti

I hereby declare that this dissertation is all my own work, except as indicated in the text.

Date 26 / April / 2021

School of Computer Science University of Nottingham Ningbo China

Abstract

Dealing with incomplete observation is one of the challenging tasks in machine learning. Incomplete observation can lead to erroneous predictions, and the consequences of erroneous predictions can sometimes be disastrous. It usually arises when the data is omitted in the collection process, when it does not meet the quality control criteria, or does not exist in the first place. This dissertation starts from the one question: "What if the user has a short amount of time to give incomplete observations, but still wants valuable results?" There are still a lot of studies that handle missing observations in training sets alone. However, background research shows that there are not many studies that cover how to deal with missing observations on the testing set when the training set has completely filled.

This dissertation compares among four methods -Deletion, Mean Imputation, Regression Imputation, and custom K nearest neighbor Imputation- that deal with missing observations. Also, there are two housing property datasets to see how methods react on different structures of a dataset. The goal of this project was to provide an in-depth comparison among different methods that handle missing observation. The dissertation premises that the training set has no missing observation.

Contents

List of Figures

List of Tables

List of Abbreviations

AVM	Automated Valuation Model
RMSE	Root Mean Square Error
CV	Cross Validation
LR	Linear Regression
RR	Ridge Regression
GDA	Gradient Descent Algorithm
MAE	Mean Absolute Error
MSE	Mean Squared Error
MCAR	Missing Completely At Random
MAR	Missing At Random
MNAR	Missing Not At Random

Chapter 1

Introduction

According to the definition from Oxford Languages, "Valuation is an estimation of something's worth, especially one carried out by a professional appraiser" [0]. Ever since people have begun selling their items, they needed a professional valuation to value at a fair price. Automated Valuation Model, in the real estate market, is a software-based pricing model that produces estimated value for a property by a machine learning model. It uses historical databases that contain property information of the surrounding areas.

Before the Automated Valuation Model was released, valuations were entirely determined by human appraisers. In the late 1990s, AVM was only used by institutional investors to determine risk when purchasing loans [1]. Along with the development of technology and machine learning, AVM has been advanced rapidly and it has been becoming more popular.

Often Appraisers review 3-5 comparable properties, similar to the requested property in terms of location, physical appearance, and overall condition. There is a limitation to find truly comparable properties. On the other hand, AVM allows to estimate value based on every single element of properties, and it is not limited to the neighborhood. Moreover, AVM has additional strength of less time consuming, smaller risk of human error, and lower service price [4]. Nowadays, Multiple AVM providers offer their AVM to users, including commercial platforms like CoreLogic, Freddie Mac, and Equifax; as well as free platforms like Zillow and Trulia [3].

This chapter describes the background and motivation of this project, moreover, explains the aims and objectives.

1.1 Background

Several authors argued how appraisal estimated prices are inconsistent and how much they are biased. The recent research shows that more than 25% of rural appraisals exceed the contract price by 5%, because of appraisers' bias [2]. Sklarz and Miller (2016) also pointed out that the average standard deviations run about 14.3% for the AVMs and 11.7% for the manual appraisals on the same property [17]. Lambie, Calem, and Nakamura also mentioned that about 92% of the appraisals exceed the

purchase price or continue to be biased(18).

On the other hand, AVM estimates the price without human interference, based on a large amount of data, thereby eliminating the risk of bias and subjectivity. As even 1% in real estate property price is gigantic, one of the strengths of AVM that eliminating human bias makes AVM more attractive.

Though AVMs also have limitations. They are only as accurate as of the data behind them, in other words, they possibly are incorrect or outdated. Recent studies show that as the model requires non-standard data, AVMs are not suitable for unique property valuation [4][5]. The research shows that RMSE differences of the result with the basic data versus result with the enhanced data were from 19% to 35% [6]. Meaning, they require an accurate, fully-filled, and well-structured dataset. Also, based on the structure of the dataset and type of data, engineers encounter different challenges, since AVMs are highly dependent on the dataset and model.

To overcome these limitations of AVMs, countless studies and discussions are actively ongoing, in terms of machine learning and database: data cleaning on missing and noisy data, data transformation, adding more data, or finding better adequate ML algorithms. Thanks to these efforts, AVMs and, in October 2018, representatives from the AVM sector in Leaders Forum in the Netherlands concluded that the AVM's usefulness will continuously expand in the future [7].

1.2 Motivation

However, in operational circumstances, there are situations where: a customer has a short amount of time to get all property information and still needs to get an estimated sale price or a customer is short on budget to hire an appraiser to get the property price. Moreover, the situation also occurs when data will be missing for a new property. This dissertation is to reflect and simulate this operational problem.

The use-case shows a concrete example of these operational circumstances in Figure 1.1. Both client and estate agent meet the situation where to use AVM with incomplete data.

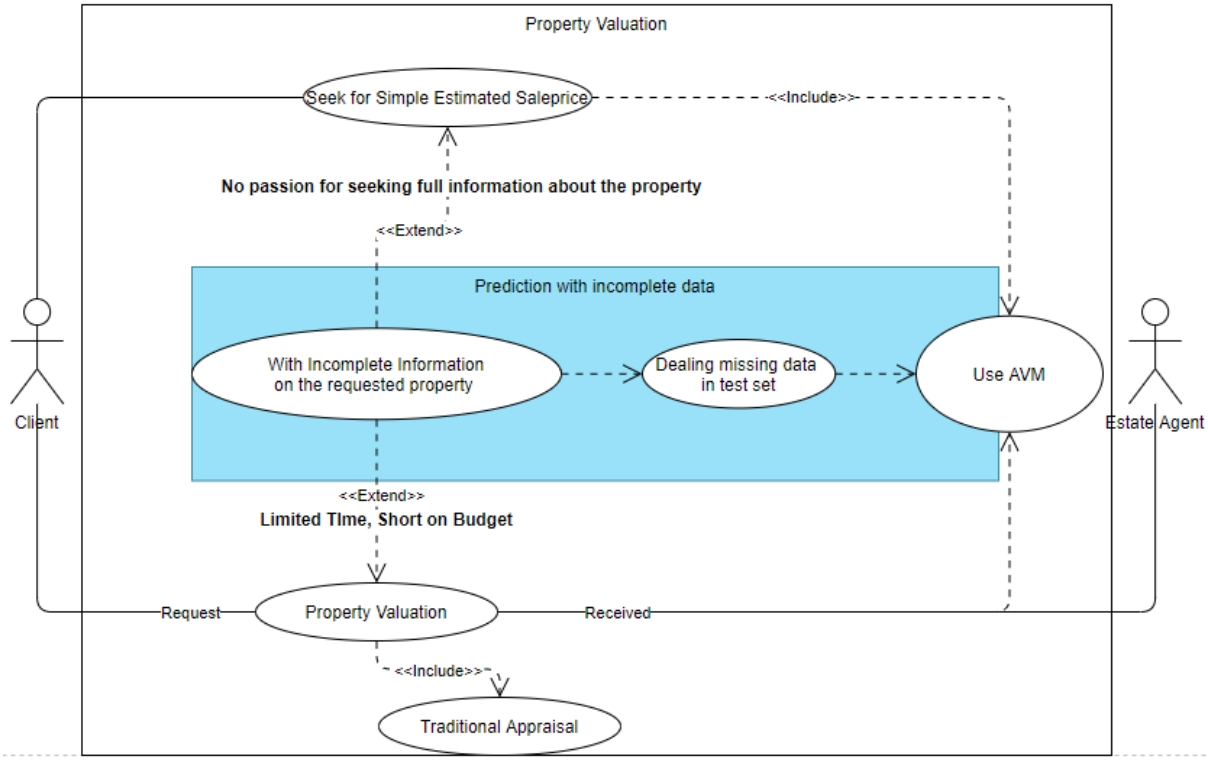


Figure 1.1. Use-case of operational circumstances

Searching through handling missing data (again, on the test set) on AVM that is on the field, it can be concluded that there are far fewer studies in progress that handling missing data in the testing set compared to those in the training set. Plus, AVM providers do not share their automated valuation models, implementations, and dataset with the public. Most of their websites only explain about AVM and why they are using them and if a client actually attempts to try AVM that they are providing, they let clients write the phone number and wait for their reply. Some AVM providers' websites manually do not let the customer leave blank on their inputs as shown in Figure 1.2. Let's say, if a client have not much time to figure out about the property, the only choice that the client has is to pick with guesses.

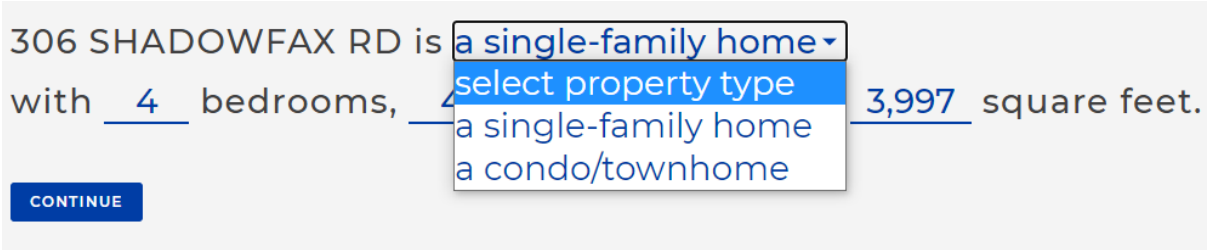


Figure 1.2. The example of one AVM provider website

Inputting inaccurate information of the property highly leads AVM to get the wrong estimated price. This specific website was one of the biggest AVM provider's websites in the US and one of the websites that actually provided AVM. However, it only asks for 4 number inputs to clients as shown in figure 1.2 when they mentioned their dataset covers 98 percent of all U.S. ZIP codes. It is hard to believe that the vast amount of data contains only 4 features, rather it comes from thoughts that if the website asks too many features, clients might decide not to use the AVM, etc. In any case, this specific AVM vendor also seems to have a way to predict when the client's data is not perfect in several ways. Thus, it is worth investigating AVM prediction with incomplete data.

1.3 Aims and Objectives

The main aim of this project is to test in operational use with limited data, then, to find the best algorithmic approach to simulate those conditions. To build a testing environment before testing, certain steps have to be accomplished: understanding dataset structure and pre-processing dataset. Those steps are also included in Chapter 4. The key objectives of this dissertation are:

1. Preprocessing the dataset to fit the model
2. Implement a cross-validation environment to obtain stable results
3. Test to find an adequate machine learning algorithm that works with the full size of the dataset.
4. Investigate imputation and non-imputation to fill up missing observations.
5. Compare different algorithmic approaches.
6. Evaluate the performance of the model with the chosen machine learning algorithm and imputation method.

1.4 Dissertation Outline

This chapter has summarized the background of AVM, provided a general overview of the automated valuation model, and the aim and objectives of this dissertation. Chapter 2 shows related research when handling missing values and Chapter 3 explains different machine learning techniques and algorithms that have been handled in this research. Chapter 4 presents the results of the implementation. Chapter 5 shows the project management. In closing, Chapter 6 expresses the reflections and considerations for future work.

Chapter 2

Related Research

As explained in the introduction, there is a limitation on stating related AVM companies' research or related studies, because when researchers study missing observation in machine learning, it is generally meant for missing data on the training set. So, Chapter 2 describes different types of missing values from the book called "statistical analysis with missing data" by Rubin and Little [20], as well as different imputation method analyses from the UCLA Statistical Consulting group [16].

2.1 Related Research

UCLA Statistical Consulting group introduced multiple imputation techniques. The goal of handling missing data they proposed was: 1. Minimize bias 2. Maximize use of available information 3. Obtain appropriate estimates of uncertainty. Also, there are three different types of missing values which are Missing completely at random (MCAR), Missing at random (MAR), and Missing not at random (MNAR) as shown in Figure 2.1. Different types of missing data require different treatments.

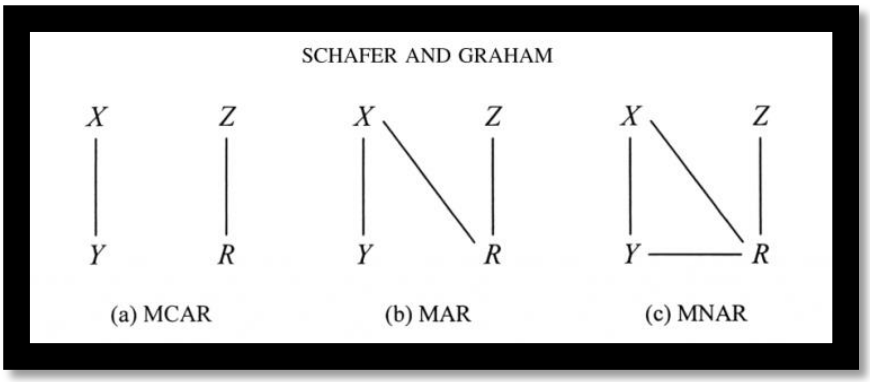


Figure 2.1. Graphical representation of MCAR, MAR, and MNAR.

X represents completely observed, Y partly missing, Z component of the causes of missingness unrelated to X and Y, R represents the missingness as explained in Rubin and Little's textbook. MCAR often happens when a subset of cases is randomly selected. In this case, the data goes missing at a completely consistent rate. MCAR is not dependent on any kind of factor. MAR is a less restrictive assumption than MCAR. In MAR, the data is missing at a certain rate, but that rate depends on some other

variable in the data. Lastly, MNAR's missingness of a certain value depends on the true value itself, so the unobserved value itself predicts missingness. When dealing with MAR and MNAR of data, it can lead to biased parameter estimates, in contrast, with MCAR, it may not reduce.

It also states that the choice of distribution, auxiliary variables, and a number of imputations can affect the quality of the imputation. Plus, the paper handled the mean imputation method and the strength and weaknesses of it. More details with mean imputation will be discussed in Chapter 3 since the mean imputation is also handled in this dissertation.

Chapter 3

Machine Learning Techniques

3.1 K-fold Cross-Validation

Cross-Validation is a resampling procedure used to evaluate models on a limited data sample. It is powerful because of several reasons: First, it allows to use of all of the data. Let's say, we have little data, splitting it into training and test set might leave us with a tiny test set. Then, the prediction cannot lead to any real conclusion. By splitting into k numbers of the fold, it allows predicting with every observation in the dataset. Secondly, it allows us to get more metrics. If we run a single evaluation, we can get only one prediction, meaning we cannot ensure that the result is reliable, because it could be predicted by chance or biased. With K-fold CV, it produces k numbers of predictions on a single run, and it allows to see whether the predictions are consistent or inconsistent. These benefits are useful, especially when proceeding with parameter tuning. By doing cross-validation, parameter tuning can be done using a single dataset.

3.2 Linear Regression

Linear regression is the most widely used algorithm because of its simplicity: easy to implement and easy to interpret the output coefficients, but still powerful [8]. Linear Regression is a supervised machine learning algorithm where the output is continuous and has a constant slope. It is used to predict continuous values rather than classification. The multivariable linear regression equation is as follows:

$$y = q_0 + q_1 x_1 + q_2 x_2 + \dots q_s x_s + e$$

The variables x represent the pieces of information with s predictor, variable q represents the weight, and e is the error term. These are the residual terms of the model.

Minimizing differences between predicted values and the actual values is the key step to optimize the linear regression model: minimizing the cost function. With linear regression equation: $H(x) = Vx + b$, V and x should be vectors and y is an outcome variable. The cost function is as follows:

$$cost(W) = \frac{1}{n}\sum_{i=1}^{n}(Vx^i - y^i)^2$$

Gradient Descent Algorithm is one of the algorithms to minimize the cost function. GDA starts from a random initial value, calculates gradient, and gives changes to W to minimize the gradient. It repeats the process on the point where gradient mostly decreased and finally finds the minimum cost function point.

$$cost(V) = \frac{1}{2n}\sum_{i=1}^{n}(Vx^i - y^i)^2$$

$$GDA = V := V - a\frac{\partial}{\partial V}cost(W)$$

The cost function is a little changed to give a clear calculation. If we substitute cost function to GDA,

$$V := V - a\frac{\partial}{\partial V}\frac{1}{2n}\sum_{i=1}^{n}(Vx^i - y^i)^2$$

If we simplify,

$$V := V - a\frac{1}{2V}\frac{1}{2n}\sum_{i=1}^{n}2(Vx^i - y^i)x^i$$

$$V := V - a\frac{1}{n}\sum_{i=1}^{n}(Vx^i - y^i)x^i$$

3.3 Ridge Regression

Ridge regression is a type of regularized linear regression that is used to data that suffers from multicollinearity or overfitting. RR adds a squared magnitude of coefficient as a penalty term to the loss function. Following is the equation of Ridge Regression:

$$\sum_{i=1}^{n}\left(y_i - \boldsymbol{\beta}_0 - \sum_{j=1}^{p}\boldsymbol{\beta}_j x_{ij}\right)^2 + a\sum_{j=1}^{p}\boldsymbol{\beta}_j^2$$

The equation can be interpreted as:

$$RSS + a\sum_{j=1}^{p}\boldsymbol{\beta}_j^2$$

Ridge regression is a sum of the residual sum of squares (RSS) + penalty term(**β**). If alpha is zero, the equation is the default OLS, but if it is greater than zero, it adds a constraint to the coefficient. This constraint leads to a minimized coefficient that tends to go to zero the greater the value of the alpha. Decreasing the coefficient stops the variance and continues the error value. Thus, Ridge regression reduces the complexity of the model, but does not reduce the number of variables, but reduces its effect. According to Scikit-Learn, ridge regression is strong when the data is less than a thousand samples or when we have more parameters than samples.

Additionally, the following is the equation of Lasso regression:

$$RSS + a\sum_{j=1}^{p}|\boldsymbol{\beta}_j|$$

By giving the absolute sum of the coefficients on the cost function, in contrast to Ridge regression's squared of the cost function, it leads to making coefficients to absolute zero when Ridge never sets the value of coefficient to absolute zero. In other words, Lasso is not punishing high values of the coefficients but as setting them to zero if it is irrelevant. Therefore, Lasso can lead to fewer features in the model.

In this dissertation, Lasso regression hasn't been used for the following reasons: Firstly, Lasso works well if there are a small number of significant features and the rest has a small influence on the response. Since our two datasets had many large parameters of about the same value or there are too few features to be removed from, Ridge regression was more suitable to the model.

3.4 Deletion of Missing Information

The deletion method is a simple way of handling missing information by deleting the missing information. Normally, when researchers use the deletion method, there are two types of deletion: Listwise versus Pairwise deletion. Listwise deletion will remove a row completely if there is a missing value for one of the variables. On the other hand,

the pairwise deletion will not omit the row and only uses information that the row has. For this dissertation, the concept of pairwise deletion is used. It is easy to think about this in a real situation where the test set has a row of client information. The pairwise deletion method lets the dataset save its size. If a client provides 10 features out of 12 features, the model deletes 2 missing features and compares only with 10 features in the training set. However, this deletion method reduces the number of features and it sometimes affects the result strongly.

3.5 Imputation

The imputation method is a widely used method when dealing with missing data. If variables are missing from the data entirely (with no bias), there is an option to omit those, but if there is some kind of potential in variables like bias, imputation is a good option to proceed. Different from eliminating the data, it leads to maintain the sample size and all variables. The imputation method is simply replacing the missing value with a substituted value.

In this dissertation, the missing observations are only located in the test set. To clarify, the imputation needs to be mapped from the train observations. There are reasons why should not be mapped from the test set: First, in a real-life situation, the model's input should be able to process one row. Taking mean or median of one row is the same value which does not do anything. Secondly, the model should create be independent of the test set. By using the test set to calculate observation, it leads the model to depend on the test set size. For instance, the number would be different when there are 10 observations and when there are 100 observations. This, mapping from a training set, logic is applied to all imputation methods in this dissertation.

3.5.1 Mean Imputation

Mean Imputation is one of the simple imputation methods that any missing observations in each feature are replaced with the mean of that specific feature in the training set. For this research, if there is missing observation in one feature, it is assumed that there is no data in that specific feature. So, for example, to get mean imputation, get a mean from a specific feature in the training set and impute to the specific feature in the testing set.

One of the biggest weaknesses of the mean imputation method is that it does not preserve the relationships among variables, because imputing mean preserves the mean of the training data. To overcome this weakness, the regression imputation

method had tested to the next.

3.5.2 Regression Imputation

Regression Imputation is an advanced imputation method that replacing missing values with a predicted result based on a regression line. Each missing feature has its model to predict its values. Let's say, there are A and B features are missing in the test set, firstly, build a model with known features in the training set, and predicted value from the model of A get imputed to feature A in the test set. Same process for feature B.

The strength of the regression imputation method is that it uses the complete data to impute values and it lets the dataset preserve relationships among variables. The drawback is that predicted values will fall directly on the regression line and it will lead to decrease variability. To overcome this weakness of Regression imputation, the custom K-nearest Neighbor comes next.

3.5.3 Custom K-Nearest Neighbor Imputation

The original K-Nearest neighbor imputation algorithms substitute the mean of K numbers of neighbors that are closest to the missing value. It uses a different type of distance metrics to calculate the distance between the feature that contains the missing value to the neighbor, in this case, the model used the Euclidean distance metric. It calculates the shortest distance between two points and the following is the formula.

$$D_e = \left(\sum_{i=1}^{n} (p_i - q_i)^2 \right)^{1/2}$$

Where D represents the distance, p and q represent two different points, and n represents the number of features. To sum up, it calculates the distance from the known features on the observation that contains an unknown variable to every single observation that is in the training set without the feature that is unknown on the test set. After calculating distances, the normal KNN imputation method would get the mean from K numbers of neighbors' feature values and substitute.

On the other hand, the custom KNN reflect the part of Monte Carlo method's concept that relies on repeated sampling to obtain the result and it doesn't get mean directly from neighbors, rather use all K numbers of neighbor to regression and get the mean of predicted results. The custom KNN method, not only allows us to get the K

numbers of the closest neighbor but also, presents more accurate and stable predictions.

Chapter 4

Implementation

This chapter specifies the implementation steps of the dataset and models. This chapter also explains the data structure (Section 4.1) and how the training set got pre-processed.

4.1 Dataset

There are two different housing datasets from the US, one from Ames' county and the other from King's county. To be easier, Ames' county dataset will be called dataset A and King's county dataset as dataset B.

Both dataset's features represent different features of housing property. Dataset A consists of both numerical and categorical values and it has a total of 2930 observations with 78 features and dataset B consists of only numerical values and it has a total of 21,613 observations with 21 features. The observations represent the number of housing properties. For additional information for features on both datasets, refer to Appendix A.

4.1.1 Pre-Processing

Unlike dataset B, dataset A had numerous categorical variables, which cannot be entered into the regression equation. One-hot encoding technique is to add a new feature according to the type of feature value and display 1 in the column corresponding to the unique value and 0 in the remaining columns. By using the *get_dummies* function from Pandas, each categorical column is transformed into three numerical columns. Then, dropping one column on each transformed set of columns increases the accuracy by adding ambiguity or noise [9]. After changed all categorical values to numeric values, the sample size of dataset A changed from (2930,83) to (2930, 160).

Secondly, there were few un-useful rows of observation on dataset A, as only 2 out of 83 features are available. Thus, those 145 numbers of observations are deleted. After deletion, the sample size got reduced from (2930,160) to (2785,160).

Figure 4.1. Example of creating a new level, 'other'

Lastly, by creating a new level and naming 'other' as shown in Figure 4.1, NAs and levels with few variables are combined into one new level. By approaching this technique, each feature can only contain major levels that have large numbers of a variable while no need to delete observation that has NA.

On the other hand, dataset B was more of the complete dataset compared to dataset A. There was not any missing value or categorical variables.

4.1.2 Coefficients

On both datasets, models were suffered from multicollinearity, in which one or more variables are linear or very nearly linear on each other. Table 4.1 shows part of the coefficient in dataset A. The intercept is 977852.8115.

Name	Lot Area	Overall Qual	Mas Vnr Area	MS Zoning other	MS SubClass_60	Street_Pave	Bsmt Unf SF	...
Coef	-4.3e-02	2.29e+03	2.80e+01	-3.75e+03	3.33e+03	1.49e+04	9.6e+12	...

Table 4.1: Coef. on different features in dataset A

There were 4 columns with an abnormally high coefficient which was 9.6e+12 when other coefficients were far less. Those four columns were 'Total Bsmt SF', 'Bsmt Unf SF', 'BsmtFin SF 1', and 'BsmtFin SF 2'. As we can assume in the similar feature names,

four features seem related. Table 4.2. shows MAPE result on deletion among those 4 features.

Environment	Full data	With 1 feature deletion	With 4 features deletion
Linear Regression on dataset A	-300.93%	89.33%	89.10%

Table 4.2: Linear Regression model MAPE results after deletion of high coefficient

The difference between the result on 1 deletion and 4 deletions seems small by 0.23%. The deletion of one feature from four features solved the issue. According to the results, we can conclude that deletion of 1 feature, Total Bsmt SF, removes major collinearity problem with the other three columns. After one column deletion, the final sample size decreased by one number of features from (2785,160) to (2785, 159).

On dataset B, multiple features had unusually high coefficients and some features are re-factored. For example, feature 'date' and feature 'yr_built' have refactored into one feature, date – year_built = 'house_age'. After deletions and refactoring, the final dataset size decreased from (21613, 21) to (21613, 12).

4.1.3 Data Preparation

For this dissertation, the dataset must have a training set with no missing values and a test set containing missing values. To see a variety of results over a different number of missing features, it needed categories. Each dataset was formatted into 7 categories. It is divided equally from 5% to 95%, and the category 15% means that the test set has 15% of known features and the remaining 85% are unknown features. The number of features on each category on both datasets is shown in Table 4.3.

Percentage of known features	# of features on set A	# of features on set B
5%	4	1
15%	12	2
30%	24	4
50%	39	6
70%	55	8
85%	67	10

95%	74	11
Full dataset	78	12

Table 4.3: The number of features on each category

There are two different approaches to leaving theoretical blanks on test sets: 1. Use-case approach 2. Systematic approach. The use-case approach is based on a use-case where a customer has a short amount of time to find out information on his property and he only finds out part of features. Hence, the 5% category contains the top 5% effortless features that users might easily find in the real world, let's say in few minutes, for instance, Month sold, Year sold on dataset A.

In a similar sense, a systematic approach is based on the ranking of predictions on each feature. After testing each feature to the sale price, the highest 5% ranked features are set into the 5% category. Thus, in category 85%, the rest 15% are the features that are ranked lowest in a systematic approach.

Ranking	Name of feature	R_squared
1	Gr Liv Area	0.548
2	Garage Area	0.410
3	Garage Cars	0.406
4	1st Flr SF	0.344
5	Exter Qual	0.329
6	Kitchen Qual	0.321
7	Year Built	0.309
8…	…	…

Table 4.4: Example of the ranking system on dataset A

To give example with Table 4.4., in the systematic approach, from ground living area to 1st floor square feet, a total of 4 are picked as 5% category. The use-case approach is more focused on realistic when the systematic approach is focused on predictions. For additional information about categories on both datasets, refer to Appendix B.

4.1.4 Learning Curve

This learning curve shows the learning performance over time, in terms of sample size. This step was necessary because if the sample size were too small, it would be pointless to investigate reasonable output. The following are learning curves on both datasets with a linear regression algorithm.

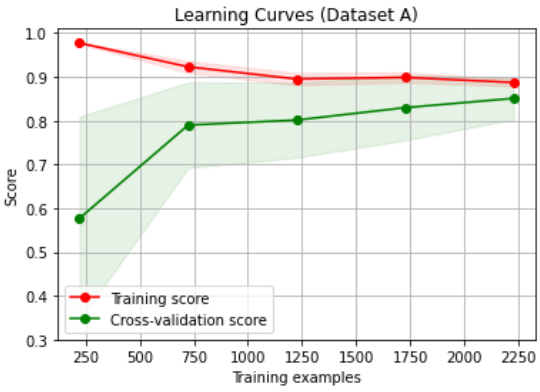


Figure 4.2 Learning Curve on A

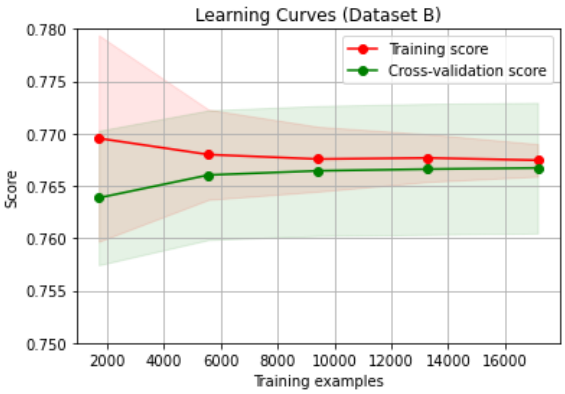


Figure 4.3 Learning Curve on B

Table 4.2 shows that there is a possibility that the result may better if we have a larger sample size. But still, the learning curve seems acceptable, especially when there is a limitation on adding more data. Table 4.3 shows that dataset B has enough sample size as the lines getting closer as they reach 6,000 samples. For both learning curves, the difference between scores seems large at the beginning by at most 0.4. But, as training examples get larger, the difference gets smaller, and the graph becomes stable for the range of 50% of the dataset to 100% of the dataset. Meaning, it would be hard to see a dramatic increase in the score if additional training examples are applied in the future. This point also seems valid to Allison's argument. Allison mentioned that ML can perform well even up to 50% missing observations (19).

4.2 AVM Results

4.2.1 Evaluation metric

This chapter gives results on different methods. There are four different regression model evaluation metrics, MAE, MSE, RMSE, and R-squared.

- Mean Absolute Error (MAE): the difference between the original and predicted values extracted by averaged the absolute difference
- Mean Squared Error (MSE): the difference between the original and predicted values extracted by squared the average difference
- Root Mean Squared Error (RMSE): error rate by the square root of MSE to give a clear view by providing a large number.

- R-squared: It is also called the coefficient of determination and it represents the percentage of the response variable variation, meaning how close the data are to the fitted regression line.

4.2.2 Comparison

4.2.2.1 Deletion of missing data

In this subsection, there are deletions of missing data models on both datasets. On this deletion section, the category of 15% means only 15% of the full data is being tested. Following table 4.5 and 4.6 are the result of dataset A with two different approaches.

%/result	**5%**	**15%**	**30%**	**50%**	**70%**	**85%**	**95%**	**100%**
MAE	47548	40018	25140	21173	18859	17907	17716	16510
MSE	490114 9788	322132 8514	158788 3672	111777 9601	981081 085	956947 544	911645 295	742568 256
RMSE	69822	56440	38876	32746	30640	30161	29532	26925
R^2	**0.226**	**0.489**	**0.754**	**0.821**	**0.837**	**0.844**	**0.845**	**0.878**

Table 4.5: Dataset A: deletion of features, Use-case approach

%/result	**5%**	**15%**	**30%**	**50%**	**70%**	**85%**	**95%**	**100%**
MAE	30464	21815	18506	18110	17611	17650	17460	16510
MSE	211497 2316	121070 7380	100823 6781	995151 484	949098 419	937962 580	891651 075	742568 256
RMSE	45740	34143	30816	30199	29936	29978	29299	26925
R^2	**0.663**	**0.803**	**0.844**	**0.847**	**0.853**	**0.850**	**0.859**	**0.878**

Table 4.6: Dataset A: deletion of features, systematic approach

Although multicollinearity has been handled by deleting one column (Chapter 4.1.3), it was worth trying Ridge Regression which is strong against multicollinearity. It seems the result is similar when there are most known features, but there is a huge difference (R_Squared < 0.437) in the low percentage of known data between the two approaches. Meaning, in a systematic approach, that several features are affecting the result hard so that even if there is only 30% of dataset presence, there's a relatively small difference to the result of the full dataset. The systematic approach shows that if there is sufficient power with a small number of features, the deletion method works, even though it lost part of its data set. Following Table 4.4 and 4.5 represent the result with dataset B.

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	220821	207776	161512	147201	136455	127333	129389	125843
MSE	122272124534	103458574057	64192553548	55916115105	45249155640	41822508173	43167136781	41000867961
RMSE	348394	320942	253118	236164	212424	204011	207214	201701
R^2	**0.093**	**0.227**	**0.524**	**0.584**	**0.663**	**0.689**	**0.679**	**0.695**

Table 4.7: Dataset B: deletion of features, Use-case approach

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	173833	164733	162541	157090	134277	135518	128476	125843
MSE	6852794471	62907564075	61967404807	57537189668	4558713531	46162559834	42542010569	41000867961
RMSE	261169	250235	248060	239660	213121	214489	205421	201701
R^2	**0.490**	**0.533**	**0.541**	**0.573**	**0.662**	**0.657**	**0.686**	**0.695**

Table 4.8: Dataset B: deletion of features, systematic approach

There is a similar trend with the result of dataset A. The high percentage of known features showed stable results on both approaches, but another big difference between the two approaches on a small percentage (<50%) of known features. When we think about the real-world scenario, it would be better to avoid prediction with a small number of features, since there is a little chance that customer's inputs are the same as the ones in systematic approach picks. Especially when the number of all features is less than 20 like our dataset B. But, the result with deletion brought out to be genuinely higher than expected, compared to other imputation methods that follow up next.

4.2.2.2 LR and RR with mean imputation

In this section, the mean imputation method is tested with Linear Regression and Ridge Regression. To clarify the steps, if a certain feature is missing on the test set, we get a mean of that certain feature from the training set and fill it into the test set. Then, fit LR or RR to predict.

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	57592	57218	48993	26810	22929	19039	18154	16510
MSE	624848	597348	445077	164229	125527	100711	958886	742568

	9203	3292	4405	5186	5296	1627	654	256
RMSE	78877	76939	66609	40324	35225	30909	30002	26925
R^2	**0.01**	**0.05**	**0.293**	**0.739**	**0.799**	**0.839**	**0.847**	**0.87**

Table 4.9: Dataset A: LR with mean imputation, Use-case approach

%/result	**5%**	**15%**	**30%**	**50%**	**70%**	**85%**	**95%**	**100%**
MAE	39777	26680	22025	20920	21165	19758	17747	16510
MSE	333900 5049	177573 2093	123188 996	116392 5880	111083 0007	102291 1675	924,944 900	742568 256
RMSE	57578	41804	34553	32841	32908	31464	29446	26925
R^2	**0.468**	**0.717**	**0.804**	**0.815**	**0.820**	**0.836**	**0.851**	**0.878**

Table 4.10: Dataset A: LR with mean imputation, systematic approach

%/result	**5%**	**15%**	**30%**	**50%**	**70%**	**85%**	**95%**	**100%**
MAE	0.415	0.405	0.319	0.310	0.274	0.209	0.200	0.196
MSE	0.279	0.264	0.159	0.150	0.115	0.072	0.066	0.064
RMSE	0.529	0.514	0.399	0.387	0.340	0.268	0.257	0.254
R^2	**-0.009**	**0.045**	**0.425**	**0.457**	**0.582**	**0.740**	**0.760**	**0.766**

Table 4.11: Dataset B: LR with mean imputation, Use-case approach

%/result	**5%**	**15%**	**30%**	**50%**	**70%**	**85%**	**95%**	**100%**
MAE	111832 51225	0.327	0.308	0.294	0.219	0.221	0.199	0.196
MSE	5.352e+ 20	0.170	0.151	0.136	0.079	0.080	0.066	0.064
RMSE	147497 32081	0.413	0.389	0.369	0.281	0.283	0.258	0.254
R^2	**-1.936**	**0.383**	**0.453**	**0.500**	**0.714**	**0.710**	**0.759**	**0.766**

Table 4.12: Dataset B: LR with mean imputation, systematic approach.

Table 4.9 – 4.12 are results of mean imputation with linear regression on both datasets with two different approaches. The first thing to notice is that results with dataset B (Table 4.11,4.12) have different units. It is because, on dataset B, only the 'price' feature had relatively large numbers compared to other features. By taking a log of the target, it compresses outliers making the distribution normal. The result overall similar to the deletion method on high percentage categories (< 70%), but it looks unstable on low

percentage categories. R-Squared is not normally linear to the percentage of features, especially on the use-case approach on both datasets.

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	58020	56562	49005	26561	21509	17877	17844	17656
MSE	6306888285	5929240846	4541210779	1652718772	1214323645	955324898	924822477	1005904918
RMSE	79236	76914	67257	40335	34550	30075	29800	30096
R^2	**0.003**	**0.06**	**0.281**	**0.741**	**0.807**	**0.848**	**0.849**	**0.855**
alpha	10.0	10.0	10.0	10.0	10.0	10.0	10.0	10.0

Table 4.13: Dataset A: RR with mean imputation, Use-case approach

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	39731	26468	20886	19961	19099	18433	17528	17656
MSE	3331558302	1768409663	1199489301	1066501674	985177202	957411147	932520836	1005904918
RMSE	57587	41810	34119	32123	30662	30442	29552	30096
R^2	**0.473**	**0.719**	**0.812**	**0.829**	**0.846**	**0.844**	**0.854**	**0.855**
	10.0	10.0	10.0	10.0	10.0	10.0	10.0	10.0

Table 4.14: Dataset A: RR with mean imputation, systematic approach

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	244299	235779	180443	172197	156498	145577	125759	125748
MSE	143483215404	132918933995	83604345680	76005852772	55506349688	47754530249	40984542518	40950694754
RMSE	377956	363940	288397	274920	235049	218062	201580	201824
R^2	**-0.065**	**0.014**	**0.380**	**0.437**	**0.588**	**0.645**	**0.697**	**0.697**

Table 4.15: Dataset B: RR with mean imputation, Use-case approach

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	189778	159415	166776	171442	148099	150639	127347	125748
MSE	9369713197 5	6740680756 6	6532033063 5	6217635076 4	4898878895 0	4972749656 4	4210517110 8	4095069475 4
RMSE	305041	259297	255026	249094	221170	222648	204839	201824
R^2	**0.307**	**0.500**	**0.515**	**0.539**	**0.636**	**0.630**	**0.688**	**0.697**
alpha	1.0	1.0	1.0	1.0	1.0	1.0	1.0	125748

Table 4.16: Dataset B: RR with mean imputation, systematic approach

Table 4.13 – 16 also represents the mean imputation but with Ridge regression. This time log of the price wouldn't be necessary, because the L2 penalty from Ridge does a similar job for the model. Based on background research, the mean imputation is one of the popular imputation methods that people usually suggest. It is still powerful when there are small numbers of missing observations (for instance, 95% known in category), but unlike the expectation, the result seems disappointing especially from 5 to 30%.

Overall, the difference between LR and RR was small, and it shows that mean imputation acts worse than deletion of missing data. The reason why is that each variable in a dataset has much more information and mean imputation reduces the variance of the data and that it changes the variance of the dataset. Followings are the histograms of the true price distribution, along with distributions of predictions in the full dataset, half, 30%, 5%, respectively on dataset A.

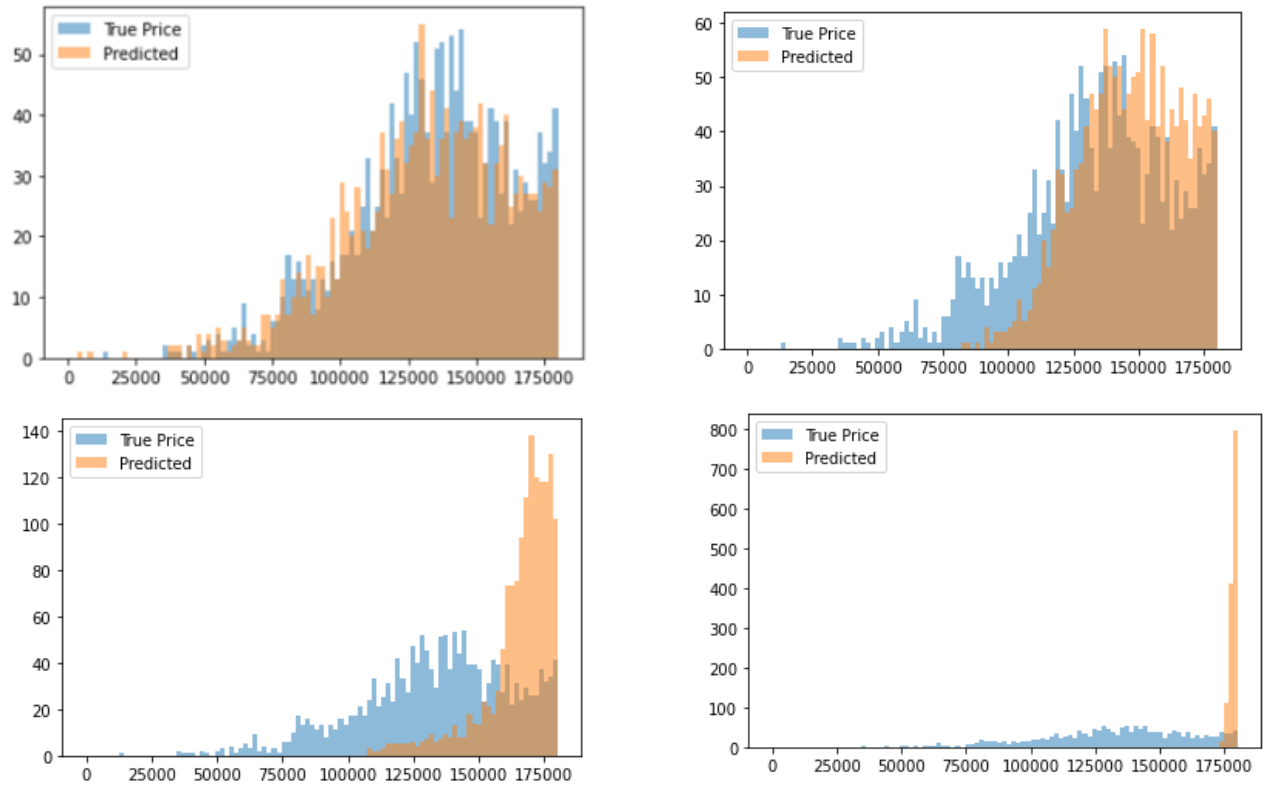


Figure 4.4. Mean imputation histogram of full dataset to 5%, left to right.

Three histograms show that the fewer features are known, the more the overall shape of the prediction graph is tilted to the right. Regardless of the shape of the actual price graph, the predictions on all four histograms are getting narrow while keeping their shape.

4.2.2.3 Regression imputation

The previous testing shows small differences between LR and RR, thereby the regression imputation is tested with linear regression.

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	47527	39973	25049	21233	18788	17554	17530	16510
MSE	4959045614	3215411549	1557958634	1109255910	994890369	948530235	928527939	742568256
RMSE	70132	56545	39008	32680	30822	30140	29427	26925
R^2	**0.208**	**0.491**	**0.753**	**0.822**	**0.841**	**0.845**	**0.852**	**0.878**

Table 4.17: Dataset A: regression imputation, use-case approach

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	30437	21753	18443	18044	17764	17882	17951	16510
MSE	2121715667	1228716163	977855595	993163117	949980068	980815242	927066038	742568256
RMSE	45552	34089	30788	30198	29734	30327	29861	26925
R^2	**0.667**	**0.796**	**0.845**	**0.842**	**0.848**	**0.842**	**0.854**	**0.878**

Table 4.18: Dataset A: regression imputation, systematic approach

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	220764	207737	164150	147681	131760	127877	127434	125843
MSE	122174387984	103355127772	6421312810	55411920913	44187552871	42368473764	42059419099	41000867961
RMSE	348732	320362	252886	235232	209768	205538	204834	201701
R^2	**0.095**	**0.235**	**0.524**	**0.588**	**0.673**	**0.684**	**0.687**	**0.695**

Table 4.19: Dataset B: regression imputation, use-case approach

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	173972	169101	162506	157128	135061	135285	128254	125843
MSE	68597281372	63164811465	61975756090	57587039107	46179449594	46168047442	42586501988	41000867961
RMSE	261437	253120	248100	239699	214622	214250	205680	201701
R^2	**0.487**	**0.501**	**0.540**	**0.571**	**0.658**	**0.657**	**0.684**	**0.695**

Table 4.20: Dataset B: regression imputation, systematic approach

Unlike the mean imputation method, category of 15% and 30% are on the right track. Even systematic approach on dataset A (Table 4.18) has R_Squared 0.667 on 5% category, which is highest among all methods. Although overall results aren't significantly higher than the deletion method, it is much higher than the mean imputation and surely more stable than the mean imputation method. Therefore, it is certain to use rather a regression imputation than the mean imputation especially when filling missing features in the test set.

4.2.2.4 Custom KNN Imputation

The last imputation method is the custom K-nearest neighbor imputation method. For the k value, the difference between k<5 and k=10 was most dramatic, compared to the result between k=10 and k=20 or k=15 and k=20. The R-squared difference between k=5 and k=10 was nearly .02 when the difference between k=10 and k=20,30 was less than .01. Also, the time consumption after k>20 increased vigorously from 15 minutes on each run to 35minutes. This trend also can be shown on dataset B. Thus, the k value for all testing with custom KNN imputation was set to 10.

%/result	**5%**	**15%**	**30%**	**50%**	**70%**	**85%**	**95%**	**100%**
MAE	64535	53717	48351	25493	20462	19651	17874	16510
MSE	7920335358	6023196080	4562676931	1400200489	1049586407	1007067754	925320502	742568256
RMSE	88904	77414	67418	37104	32057	30893	29600	26925
R^2	**-0.269**	**0.036**	**0.262**	**0.780**	**0.829**	**0.843**	**0.853**	**0.878**

Table 4.21: Dataset A: regression imputation, use-case approach

%/result	**5%**	**15%**	**30%**	**50%**	**70%**	**85%**	**95%**	**100%**
MAE	33483	24302	22161	20171	18366	18281	17743	16510
MSE	2446245490	1373270459	1193632045	1064611561	933564136	934139646	897138288	742568256
RMSE	49319	36630	34159	106461 1561	30010	30130	29035	26925
R^2	**0.599**	**0.784**	**0.802**	**0.826**	**0.850**	**0.851**	**0.859**	**0.878**

Table 4.22: Dataset A: regression imputation, systematic approach

%/result	**5%**	**15%**	**30%**	**50%**	**70%**	**85%**	**95%**	**100%**

MAE	0.443	0.422	0.331	0.334	0.293	0.202	0.198	0.196
MSE	0.318	0.282	0.171	0.177	0.137	0.067	0.065	0.064
RMSE	0.563	0.530	0.414	0.421	0.370	0.259	0.255	0.254
R^2	**-0.110**	**0.013**	**0.399**	**0.380**	**0.522**	**0.764**	**0.762**	**0.766**

Table 4.23: Dataset B: regression imputation, use-case approach

%/result	5%	15%	30%	50%	70%	85%	95%	100%
MAE	0.379	0.344	0.330	0.324	0.286	0.217	0.188	0.196
MSE	0.224	0.186	0.171	0.165	0.126	0.079	0.064	0.064
RMSE	0.473	0.431	0.413	0.406	0.345	0.281	0.231	0.254
R^2	**0.167**	**0.300**	**0.404**	**0.425**	**0.556**	**0.722**	**0.706**	**0.766**

Table 4.24: Dataset B: regression imputation, systematic approach

From the results of the table from 4.21 to 4.24, the custom KNN imputation method performs significantly worse in the 5% category. Mid-range (15%-50%) performance is similar to linear regression and deletion, but high percentages (85% – 95%) scored highest among all methods especially on dataset B. The result seems understandable because when the mean imputation and regression imputation treat observations in a missing feature as a whole, custom KNN imputation counts each into each. For example, pretend there are 3 missing observations on both the mean imputation model and custom KNN model, the mean imputation model will substitute the same 1 variable into 3 observations. On the other hand, the KNN model treats each observation separately, so that 3 different variables are substituted into 3 observations and that gives the model diversity of the data.

Chapter 5

Project Management

A Gantt chart of the work plan is presented in Appendix C. In the first semester, the main focus of the project was on data pre-processing, coding, and understanding algorithms. In the second semester, the main focus was implementing advanced imputation methods like custom KNN, testing, and collecting valuable results from implemented models.

Some methods like the cross-validation method or K nearest neighbor imputation method weren't planned at the beginning of this project. Because of the nature of the

research, multiple plans were added and removed during the project. The deletion method's shrinking size of the dataset problem led to implementing the mean imputation method, the mean imputation method's removing relationship between variables problem led to implementing regression imputation method, and so on. In the past, I simply knew how to run a Schkit-learn model. But, as progressing this project, the supervisor suggested new techniques, at the same time, I began to understand why these new techniques are necessary. In the same sense, the project was first planned to use one dataset, but, in the second semester, dataset B is added to get more variety on the result.

To sum up, the project management was well managed to obtain valuable results.

Chapter 6

Conclusions and Reflections

This dissertation explores how AVM behaves when there is missing information on the requested property. The deletion method worked relatively well, compared to other methods when the mean imputation methods performed worse. According to the result, it seems to remove features is resulting in better than confusing the relationship between features. Overall, in a range of 5-50% category (95 – 50% missing information), it did not bring admired results and if that is the case in a real circumstance, it is wise to use the deletion method. Plus, on 70-95% of categories, the custom KNN imputation method performed significantly well, especially when the dataset has fewer features like dataset B.

This project was progressed during the outbreak of the COVID-19. That led me to proceed with the whole research outside of the campus and honestly, it seemed harsh to communicate with the supervisor only with email and video call, especially when my first language is not English. I would like to thank my supervisor, Dr. Anthony Bellotti. He even used the online whiteboard to explain concepts of techniques or algorithms and I haven't felt any shortage of guidance during the project.

At the beginning of this project, I was a student who can barely run a simple ML model. Because of this module, I learned the whole idea of data pre-processing and techniques, gained experience in implementing custom models, learned how to visually represent the results, and more.

For the future, I am planned to explore more about this topic via GitHub, with the neural network, custom weighted the nearest neighbor and more.

References

[1] Kok,Nils; Koponen, Elija-Leena; Martinez-Barbosa, Carmen Adriana. Big Data in Real Estate? From Manual Appraisal to Automated Valuation (2017)

[2] Andrew Fortelny, Dr. Richard Reed. The increasing use of automated valuation models in Australian mortgage market (2005)

[3] Alexander N.Bogin; Jessica Shui. Appraisal Accuracy and Automated Valuation Models in Rural Area (2020)

[4] Faishal Ibrahim, Muhammad; Jam Cheng, Fook; How Eng, Kheng. Automated valuation model: an application to the public housing resale market in Singapore 357-373 (2005)

[5] William J.McCluskey, Michael Mccord, Peadar Thomas Davis, Martin Haran, D.Mclhatton. Prediction accuracy in mass appraisal: A comparison of modern approaches (2013)

[6] Peter Rossini; Paul John Kershaw. Automated valuation model accuracy: some empirical testing (2008)

[7] Sander Scheurwater, Automated Valuation Models – which future? (2018)

[8] Su X; Yan X; Tsai C. Linear regression, Volume 4, issue 3, pp. 275-294 (2012)

[9] Nikolas M; Imputations: Benefits, Risks, and a Method for Missing Data (2013)

[10] Geron, A. Hands-on machine learning with Scikit-Learn, Keras and TensorFlow: concepts, tools, and techniques to build intelligent systems (2nd ed.). page 92-140 (2019)

[11] Tan Yigitcanlar. Smart cities: an effective urban development and management model? (2015)

[12] Gregory S. Yovanof, George N. Hazapis. An architectural framework and enabling wireless technologies for digital cities and intelligent urban environments (2009)

[13] Yaling Zheng. Machine Learning with incomplete information (2011)

[14] Joachim Schork. Regression imputation Stochastic vs. Deterministic and R example https://statisticsglobe.com/regression-imputation-stochastic-vs-deterministic/

[15] Grigorios Papageorgiou, Stuart W Grant, Johanna J M Takkenberg, Mostafa M Mokhles. Statistical primer: how to deal with missing data in scientific research? (2018)

[16] UCLA Statistical Consulting Group. Multiple Imputation In Stata. See: http://stats.idre.ucla.edu/stata/seminars/mi_in_stata_pt1_new/. Accessed 10-April-2021

[17] Sklarz, M. and N. Miller, "Adjusting Loan to Value (LTV) Ratios to Reflect Value Uncertainty" Working Paper, August 1, (2016) See http://collateralanalytics.com/adjusting-loan-to-value-ltv-ratios-to-reflect-valueuncertainty/ Accessed 1-March-2021

[18] Calem, P., L. Lambie-Hanson, and L. Nakamura, "Information Losses in Home Purchase Appraisals" Federal Reserve Bank of Philadelphia, Working Paper 15-11, March, (2015)

[19] Allison P. D "Missing Data Techniques for Structural Equation Modeling" University of Pennsylvania (2003)

[20] Rubin D. B., Little R. J., "Statistical Analysis with Missing Data, Second Edition" Wiley Series (2002)

Appendix A

Feature Variables Description (dataset A)

The description below is detailed explanation on each feature variable to give more understanding of the dataset.

MSSubClass: Identifies the type of dwelling involved in the sale.
MSZoning: Identifies the general zoning classification of the sale.
LotFrontage: Linear feet of street connected to property
LotArea: Lot size in square feet
Street: Type of road access to property
Alley: Type of alley access to property
LotShape: General shape of property
LandContour: Flatness of the property
Utilities: Type of utilities available
LotConfig: Lot configuration
LandSlope: Slope of property
Neighborhood: Physical locations within Ames city limits
Condition1: Proximity to various conditions
Condition2: Proximity to various conditions (if more than one is present)
BldgType: Type of dwelling
HouseStyle: Style of dwelling
OverallQual: Rates the overall material and finish of the house
OverallCond: Rates the overall condition of the house
YearBuilt: Original construction date
YearRemodAdd: Remodel date (same as construction date if no remodeling or additions)
RoofStyle: Type of roof
RoofMatl: Roof material
Exterior1st: Exterior covering on house
Exterior2nd: Exterior covering on house (if more than one material)
MasVnrType: Masonry veneer type
MasVnrArea: Masonry veneer area in square feet
ExterQual: Evaluates the quality of the material on the exterior
ExterCond: Evaluates the present condition of the material on the exterior
Foundation: Type of foundation
BsmtQual: Evaluates the height of the basement
BsmtCond: Evaluates the general condition of the basement

BsmtExposure: Refers to walkout or garden level walls
BsmtFinType1: Rating of basement finished area
BsmtFinSF1: Type 1 finished square feet
BsmtFinType2: Rating of basement finished area (if multiple types)
BsmtFinSF2: Type 2 finished square feet
BsmtUnfSF: Unfinished square feet of basement area
TotalBsmtSF: Total square feet of basement area
Heating: Type of heating
HeatingQC: Heating quality and condition
CentralAir: Central air conditioning
Electrical: Electrical system
1stFlrSF: First Floor square feet
2ndFlrSF: Second floor square feet
LowQualFinSF: Low quality finished square feet (all floors)
GrLivArea: Above grade (ground) living area square feet
BsmtFullBath: Basement full bathrooms
BsmtHalfBath: Basement half bathrooms
FullBath: Full bathrooms above grade
HalfBath: Half baths above grade
Bedroom: Bedrooms above grade (does NOT include basement bedrooms)
Kitchen: Kitchens above grade
KitchenQual: Kitchen quality
TotRmsAbvGrd: Total rooms above grade (does not include bathrooms)
Functional: Home functionality
GarageType: Garage location
GarageYrBlt: Year garage was built
GarageFinish: Interior finish of the garage
GarageCars: Size of garage in car capacity
GarageArea: Size of garage in square feet
GarageQual: Garage quality
GarageCond: Garage condition
PavedDrive: Paved driveway
WoodDeckSF: Wood deck area in square feet
OpenPorchSF: Open porch area in square feet
EnclosedPorch: Enclosed porch area in square feet
3SsnPorch: Three season porch area in square feet
ScreenPorch: Screen porch area in square feet
PoolArea: Pool area in square feet
PoolQC: Pool quality
Fence: Fence quality
MiscFeature: Miscellaneous feature not covered in other categories
MiscVal: $Value of miscellaneous feature

MoSold: Month Sold (MM)
YrSold: Year Sold (YYYY)
SaleType: Type of sale
SaleCondition: Condition of sale

Feature Variables Description (dataset B)

house_age: Property age, date – year built

sqft_living: Square footage of the living area

grade: Overall grade

condition: Overall condition

sqft_living15: Living room area in 2015(implies renovations)

bathrooms: number of bathrooms

view: views that has been viewed

sqft_basement: Square footage of the basement

floors: total floors(levels) in house

waterfront: House which has a view to a waterfront

renovated: year of renovation

bedrooms: number of bedrooms

Appendix B

The description below is showing each category's designated feature variables. The category 5% meant for 5% of data is presence and rest are missing. **In 70-95%, listed features are the ones that are known for both datasets and both approaches.**

Dataset A

Based on the Use-Case selection

5%

'Exterior 1st_HdBoard', 'Exterior 1st_MetalSd', 'Exterior 1st_Plywood', 'Exterior 1st_VinylSd', 'Exterior 1st_Wd Sdng', 'Exterior 1st_other','Lot Area','Mo Sold','Yr Sold'

15%

'Exterior 1st_HdBoard', 'Exterior 1st_MetalSd', 'Exterior 1st_Plywood', 'Exterior 1st_VinylSd', 'Exterior 1st_Wd Sdng', 'Exterior 1st_other','Lot Area','Mo Sold','Yr Sold','Pool Area','Year Remod/Add','Bedroom AbvGr', 'Bldg Type_Duplex', 'Bldg Type_Twnhs', 'Bldg Type_TwnhsE', 'Bldg Type_other','Fireplaces', 'House Style_1Story', 'House Style_2Story', 'House Style_SLvl', 'House Style_other', 'Sale Condition_Normal', 'Sale Condition_Partial', 'Sale Condition_other','Roof Style_Hip', 'Roof Style_other'

30%

'Exterior 1st_HdBoard','Exterior 1st_MetalSd', 'Exterior 1st_Plywood', 'Exterior 1st_VinylSd', 'Exterior 1st_Wd Sdng', 'Exterior 1st_other','Lot Area','Mo Sold','Yr Sold','Pool Area','Year Remod/Add','Bedroom AbvGr', 'Bldg Type_Duplex', 'Bldg Type_Twnhs','Bldg Type_TwnhsE', 'Bldg Type_other','Fireplaces', 'House Style_1Story', 'House Style_2Story', 'House Style_SLvl','House Style_other', 'Sale Condition_Normal', 'Sale Condition_Partial', 'Sale Condition_other','Roof Style_Hip','Roof Matl_other', 'Roof Style_other', 'Exter Cond_TA', 'Exter Cond_other', 'Exter Qual_Gd', 'Exter Qual_TA', 'Exter Qual_other', 'Heating QC_Gd', 'Heating QC_TA', 'Heating QC_other', 'Heating_other','Paved Drive_Y','Paved Drive_other', 'BsmtFin Type 2_Rec','BsmtFin Type 2_Unf', 'BsmtFin Type 2_other', '1st Flr SF', 'Land Slope_Mod', 'Land Slope_Sev', 'Lot Config_CulDSac', 'Lot Config_Inside', 'Lot Config_other', 'Foundation_CBlock', 'Foundation_PConc', 'Foundation_other', 'MS SubClass_160', 'MS SubClass_20', 'MS SubClass_30', 'MS SubClass_50','MS SubClass_60', 'MS SubClass_70', 'MS SubClass_other', 'Low Qual Fin SF'

50%

'Exterior 1st_HdBoard','Exterior 1st_MetalSd', 'Exterior 1st_Plywood', 'Exterior 1st_VinylSd', 'Exterior 1st_Wd Sdng', 'Exterior 1st_other','Lot Area','Mo Sold','Yr Sold','Pool Area','Year Remod/Add','Bedroom AbvGr', 'Bldg Type_Duplex', 'Bldg Type_Twnhs','Bldg Type_TwnhsE', 'Bldg Type_other','Fireplaces', 'House Style_1Story','House Style_2Story', 'House Style_SLvl','House Style_other', 'Sale Condition_Normal', 'Sale Condition_Partial', 'Sale Condition_other','Roof Style_Hip','Roof Matl_other', 'Roof Style_other', 'Exter Cond_TA', 'Exter Cond_other', 'Exter Qual_Gd', 'Exter Qual_TA', 'Exter Qual_other', 'Heating QC_Gd','Heating QC_TA', 'Heating QC_other', 'Heating_other', 'Paved Drive_Y','Paved Drive_other', 'BsmtFin Type 2_Rec','BsmtFin Type 2_Unf', 'BsmtFin Type 2_other', '1st Flr SF', 'Land Slope_Mod', 'Land Slope_Sev', 'Lot Config_CulDSac', 'Lot Config_Inside', 'Lot Config_other', 'Foundation_CBlock', 'Foundation_PConc', 'Foundation_other', 'MS SubClass_160','MS SubClass_20', 'MS SubClass_30', 'MS SubClass_50','MS SubClass_60', 'MS SubClass_70', 'MS SubClass_other', 'Low Qual Fin SF', 'Mas Vnr Area', 'Mas Vnr Type_None', 'Mas Vnr Type_Stone', 'Mas Vnr Type_other', 'Neighborhood_CollgCr', 'Neighborhood_Crawfor', 'Neighborhood_Edwards', 'Neighborhood_Gilbert', 'Neighborhood_IDOTRR', 'Neighborhood_Mitchel', 'Neighborhood_NAmes', 'Neighborhood_NWAmes', 'Neighborhood_NridgHt', 'Neighborhood_OldTown', 'Neighborhood_Sawyer', 'Neighborhood_SawyerW', 'Neighborhood_Somerst', 'Neighborhood_other','Open Porch SF', 'Garage Yr Blt', 'Gr Liv Area', 'Half Bath','Kitchen AbvGr', 'Kitchen Qual_TA', 'Kitchen Qual_other', 'Land Contour_HLS', 'Land Contour_Low', 'Land Contour_Lvl','Year Built'

70%

In 70-95%, listed features are the ones that are known

'Misc Val', 'Overall Qual', 'Utilities_other', 'Wood Deck SF', 'Condition 1_Feedr', 'Condition 1_Norm', 'Condition 1_other', 'Condition 2_other', 'Electrical_SBrkr', 'Electrical_other', 'Enclosed Porch', 'Exterior 2nd_HdBoard', 'Full Bath', 'Fence_other','Fireplace Qu_TA', 'Fireplace Qu_other', 'MS Zoning_RM', 'MS Zoning_other', '2nd Flr SF', '3Ssn Porch', 'Alley_other','BsmtFin SF 1','BsmtFin SF 2','Bsmt Cond_Gd', 'Bsmt Cond_TA', 'Bsmt Cond_other', 'Bsmt Exposure_Gd', 'Bsmt Exposure_Mn', 'Bsmt Exposure_No', 'Bsmt Exposure_other', 'Bsmt Full Bath','Bsmt Half Bath', 'Bsmt Qual_Gd', 'Bsmt Qual_TA', 'Bsmt Qual_other', 'Bsmt Unf SF', 'Sale Type_other'

85%

'Misc Val', 'Overall Qual', 'Utilities_other', 'Wood Deck SF', 'Condition 1_Feedr', 'Condition 1_Norm', 'Condition 1_other','Condition 2_other', 'Electrical_SBrkr', 'Electrical_other', Enclosed Porch', 'Exterior 2nd_HdBoard','Full Bath', 'Fence_other','Sale Type_other'

95%

'Misc Val','Overall Qual', 'Utilities_other','Sale Type_other'

Based on the systemetic selection

5%

'Gr Liv Area', 'Garage Area', 'Garage Cars', '1st Flr SF'

15%

'Gr Liv Area', 'Garage Area', 'Garage Cars', '1st Flr SF','Exter Qual_Gd', 'Exter Qual_TA', 'Exter Qual_other', 'Kitchen Qual_TA', 'Kitchen Qual_other','Year Built', 'Mas Vnr Area', 'Foundation_CBlock', 'Foundation_PConc', 'Foundation_other','Fireplace Qu_TA', 'Fireplace Qu_other','Fireplaces','BsmtFin SF 1'

30%

'Gr Liv Area', 'Garage Area', 'Garage Cars', '1st Flr SF','Exter Qual_Gd', 'Exter Qual_TA', 'Exter Qual_other', 'Kitchen Qual_TA', 'Kitchen Qual_other','Year Built', 'Mas Vnr Area', 'Foundation_CBlock', 'Foundation_PConc', 'Foundation_other', Fireplace Qu_TA', 'Fireplace Qu_other','Fireplaces','BsmtFin SF1','Neighborhood_CollgCr','Neighborhood_Crawfor','Neighborhood_Edwards', Neighborhood_Gilbert', 'Neighborhood_IDOTRR','Neighborhood_Mitchel', 'Neighborhood_NAmes', 'Neighborhood_NWAmes','Neighborhood_NridgHt','Neighborhood_OldTown', Neighborhood_Sawyer','Neighborhood_SawyerW', 'Neighborhood_Somerst', 'Neighborhood_other', 'Bsmt Qual_Gd', 'Bsmt Qual_TA', 'Bsmt Qual_other','Garage Yr Blt','BsmtFin Type 1_GLQ', 'BsmtFin Type 1_LwQ', 'BsmtFin Type 1_Rec','BsmtFin Type 1_Unf', 'BsmtFin Type 1_other','Overall Qual', 'Garage Finish_RFn', 'Garage Finish_Unf','Garage Finish_other','MS SubClass_160', 'MS SubClass_20', 'MS SubClass_30', 'MS SubClass_50','MS SubClass_60', 'MS SubClass_70', 'MS SubClass_other', 'Sale Condition_Normal', 'Sale Condition_Partial','Sale Condition_other','Mas Vnr Type_None','Mas Vnr Type_Stone', 'Mas Vnr Type_other', 'Exterior 2nd_HdBoard','Exterior 2nd_MetalSd', 'Exterior 2nd_Plywood', 'Exterior 2nd_VinylSd','Exterior 2nd_Wd Sdng', 'Exterior 2nd_other', 'Wood Deck SF','Foundation_CBlock'

50%

'Gr Liv Area', 'Garage Area', 'Garage Cars', '1st Flr SF','Exter Qual_Gd', 'Exter Qual_TA', 'Exter Qual_other', 'Kitchen Qual_TA', 'Kitchen Qual_other','Year Built',Ms Vnr Area', 'Foundation_CBlock', 'Foundation_PConc', 'Foundation_other', Fireplace Qu_TA', 'Fireplace Qu_other','Fireplaces','BsmtFin SF1','Neighborhood_CollgCr','Neighborhood_Crawfor','Neighborhood_Edwards', Neighborhood_Gilbert', 'Neighborhood_IDOTRR','Neighborhood_Mitchel', 'Neighborhood_NAmes', 'Neighborhood_NWAmes','Neighborhood_NridgHt','Neighborhood_OldTown',Neighborhood_Sawyer','NeighborhoodSawyerW', 'Neighborhood_Somerst', 'Neighborhood_other', 'Bsmt Qual_Gd', 'Bsmt Qual_TA', 'Bsmt Qual_other','Garage Yr Blt','BsmtFin Type 1_GLQ', 'BsmtFin Type 1_LwQ', 'BsmtFin Type 1_Rec', 'BsmtFin Type 1_Unf', 'BsmtFin Type 1_other','Overall Qual', 'Garage Finish_RFn', 'Garage Finish_Unf','Garage Finish_other','MS SubClass_160', 'MS SubClass_20', 'MS SubClass_30', 'MS SubClass_50', 'MS SubClass_60', 'MS SubClass_70', 'MS

SubClass_other', 'Sale Condition_Normal', 'Sale Condition_Partial','Sale Condition_other','Mas Vnr Type_None','Mas Vnr Type_Stone', 'Mas Vnr Type_other', 'Exterior 2nd_HdBoard','Exterior 2nd_MetalSd', 'Exterior 2nd_Plywood', 'Exterior 2nd_VinylSd', 'Exterior 2nd_Wd Sdng', 'Exterior 2nd_other', 'Wood Deck SF','Foundation_CBlock', 'Garage Type_BuiltIn', 'Garage Type_Detchd', 'Garage Type_other','Exterior 1st_HdBoard', 'Exterior 1st_MetalSd', 'Exterior 1st_Plywood', 'Exterior 1st_VinylSd', 'Exterior 1st_Wd Sdng', 'Exterior 1st_other', 'Heating QC_Gd','Heating QC_TA','Bsmt Full Bath', 'Half Bath','Full Bath','Open Porch SF' 'Lot Shape_Reg', 'Lot Shape_other','Bsmt Exposure_Gd', 'Bsmt Exposure_Mn','Bsmt Exposure_No', 'Bsmt Exposure_other','Lot Area', 'MS Zoning_RM', 'MS Zoning_other','Roof Style_Hip','Roof Style_other','Paved Drive_Y', 'Paved Drive_other', 'Garage Qual_TA', 'Garage Qual_other'

70%

In 70-95%, listed features are the ones that are known

'Condition 2_other','Pool Area','Low Qual Fin SF', 'Mo Sold', 'Land Contour_HLS', 'Land Contour_Low','Land Contour_Lvl','Street_Pave', 'Lot Config_CulDSac', 'Lot Config_Inside', 'Lot Config_other','House Style_1Story', 'House Style_2Story', 'House Style_SLvl','House Style_other','Condition 1_Feedr', 'Condition 1_Norm', 'Condition 1_other','BsmtFin Type 2_Rec', 'BsmtFin Type 2_Unf', 'BsmtFin Type 2_other', 'Exter Cond_TA', 'Exter Cond_other','Land Slope_Mod', 'Land Slope_Sev','Misc Feature_other','Utilities_other','Yr Sold','Bsmt Half Bath','3Ssn Porch','Bldg Type_Duplex', 'Bldg Type_Twnhs','Bldg Type_TwnhsE', 'Bldg Type_other','BsmtFin SF 2','Year Remod/Add'

85%

''Condition 2_other','Pool Area','Low Qual Fin SF', 'Mo Sold','Land Contour_HLS', 'Land Contour_Low', 'Land Contour_Lvl','Street_Pave', 'Lot Config_CulDSac', 'Lot Config_Inside', 'Lot Config_other', 'House Style_1Story', 'House Style_2Story', 'House Style_SLvl','House Style_other','Condition 1_Feedr', 'Condition 1_Norm', 'Condition 1_other','BsmtFin Type 2_Rec','BsmtFin Type 2_Unf', 'BsmtFin Type 2_other', 'Exter Cond_TA', 'Exter Cond_other'

95%

'Condition 2_other','Pool Area','Low

Dataset B

Based on the Use-case selection

5% 'bedrooms'

15% 'bedrooms','view'

30% 'bedrooms','view','grade','waterfront'

50% 'bedrooms', 'sqft_living15','bathrooms', 'view','grade', 'waterfront

70% **In 70-95%, listed features are the ones that are known**

'bedrooms', 'sqft_living15','bathrooms', 'view'

85% 'bedrooms', 'sqft_living15'

95% 'bedrooms'

Based on the systemetic selection

5% sqft_living'

15% 'sqft_living','grade'

30% 'sqft_living','sqft_above','grade','sqft_living15'

50% 'sqft_living','sqft_above','grade','sqft_living15','bathrooms', 'view'

70% **In 70-95%, listed features are the ones that are known**

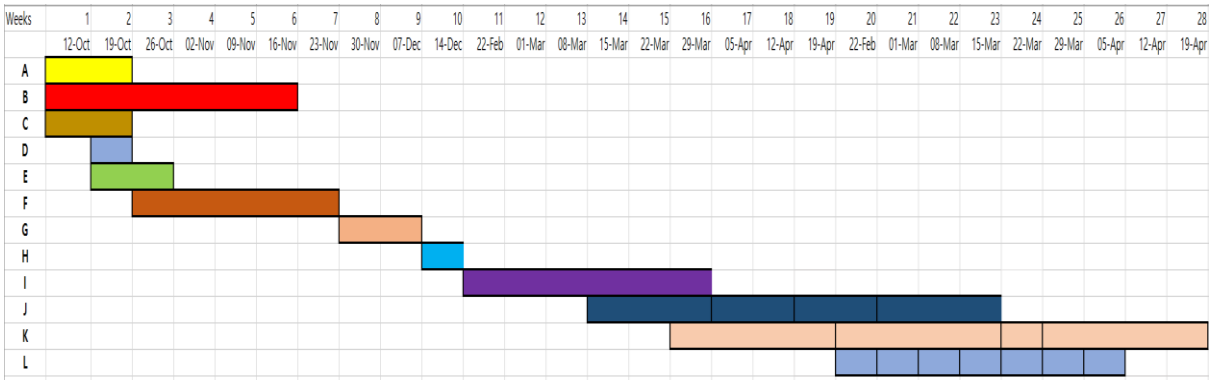
'condition', 'house_age','zipcode', 'sqft_lot15'

85%'condition', 'house_age'

95%'condition'

Appendix C

Gantt Chart



A. Write and submit the proposal

B. Background research on different techniques

C. Test the environments

D. Data preprocessing / mean imputation

E. Regression imputation

F. Collecting results / bug fixing

G. Cross validation

H. write interim report

I. Custom KNN

J. Fixing bug and error from all previous methods and collecting results

K. Background research and Writing the final report

L. Testing all again with the new dataset