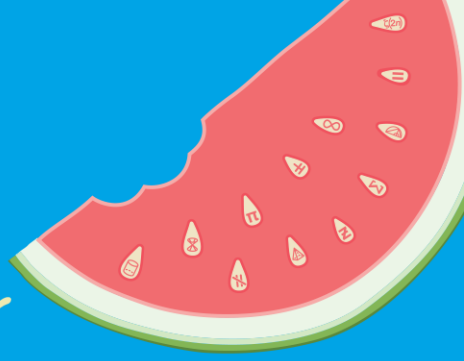
AMSI VACATION RESEARCH SCHOLARSHIPS 2021–22

Get a taste for Research this Summer



Hybrid Non-Linear Statistical Methods with Applications to Modelling House Prices

Adam Bilchouris

Supervised by Associate Professor Andriy Olenko

La Trobe University

Vacation Research Scholarships are funded jointly by the Department of Education and the Australian Mathematical Sciences Institute.

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE

Contents

Abstract

Many companies provide house price predictions, although some fail to provide reasonable estimates for anomalous houses. The aim of this project is investigating approaches to predict prices for such houses. We explore the detection of anomalous houses via an Isolation Forest, and four models to predict prices of these anomalous houses. They are a nonlinear regression model, a historical model, a hybrid model, and a weighted hybrid model which tries to express the importance of recency when using historical sales to predict a house's price.

1 Introduction

A house's price can be impacted by many factors, from the number of bedrooms and bathrooms to the suburb it is located in. Being able to predict a house's price accurately is a difficult problem, but important for many companies. For example, on realestate.com.au, a house had a prediction range of $660,000 to $960,000, and on Domain it was $740,000 to $990,000. This house had no strange features, yet the prediction range is quite large. When houses have strange features, such as a large land size, many models tend to fail. Consider this house, https://www.domain.com.au/property-profile/161-murray-road-diamond-creek-vic-3089, which has 46,539m^2 of land. No prediction is provided at the time of writing due to its unusual nature.

This project attempts to find the key features of a house, detect anomalous houses through an Isolation Forest, and then predict their prices using various methods. These methods are a nonlinear regression model, a model using previous sales of a house, a combination of the regression and previous sales, which will be called a hybrid model, and finally a weighted hybrid model, which gives more weight to more recent sales of a house. For the regression model, a Genetic Algorithm was employed to remove features that were deemed insignificant for the model, reducing the number of features needed to predict the price.

A reproducible version of the R code for this project can be found at https://github.com/AdamBilchouris/AMSI_Code.

1.1 Statement of Authorship

A. Bilchouris and A. Olenko developed the theoretical models. Under the supervision of Olenko, Bilchouris wrote code to scrape websites, analysed the data, and compared the models.

2 Data

As there was not a publicly available dataset which contained all information about houses, one had to be constructed. This was done by scraping data from Domain and realestate.com.au. The number of houses in the dataset was 8657, and information about the suburbs were introduced into the dataset through census data and general information obtained from other sources. It would be interesting to further develop the suggested approaches and apply them to large datasets. It is expected that in such cases more statistically significant results could be obtained, although most housing datasets are kept private by banks, real estate companies, and insurance companies.

2.1 Property Data

Property data was scraped from Domain using `Python 3` alongside urllib3 and Beautiful Soup 4. The web page https://www.domain.com.au/sold-listings/ contains sold properties and has several parameters including the number of bedrooms, bathrooms, parking spaces, etc. An important feature of this web page is being able to search by suburb. This allowed various suburbs of Melbourne to have sales scraped. See Appendix A for the list of suburbs.

Previous sales for 38 suburbs around Melbourne were scraped. For each suburb, houses with 1, 2, 3, 4, and 5 or more bedrooms were considered separate when obtaining data. The features scraped from these pages are:

`address, sale price, sale date, bedrooms, bathrooms, parking_spaces, land_size, land_size_unit, propertyType, url`

The `url` was used to scrape more data from the individual sales page. This data included features about the house such as a garden, air-conditioning, heating, and built-in wardrobes as well as nearby schools. Using the `address`, the `property-profile` page for each property was

determined. This page provided valuable information about previous sales of a house, which included the sale year and month, and the sale price. Unfortunately, the list of previous sales included rental history. As only three previous sales were taken per house, rental histories were sometimes included. This meant some houses had less historical sales which could be used in the models.

2.2 Median Sales Data

The median sale data from 2012 to October 2021 was scraped from realestate.com.au. This was done for every suburb used. An example page can be found at https://www.realestate.com.au/neighbourhoods/bundoora-3083-vic. This data was used to scale historical sales (before October 2021) to current prices (October 2021).

Suppose a house was sold in 2016 for some amount, Sale_{2016}. The median sale price for 2016 is Median_{2016} and for 2021, Median_{2021}.
Then, the price in 2021 can be found by:

$$\text{Sale}_{2021} = \frac{\text{Median}_{2021}}{\text{Median}_{2016}}\text{Sale}_{2016}$$

If a sale occurred in 2021, then monthly median sales were used instead. As median sales were only available up to October 2021, any sale beyond that was not adjusted.

Below are approximated smoothing functions for the non-adjusted and adjusted prices using the ratio calculated from the median sales data. In Figure 1, there is a clear upward trend with a dip between 2018 and 2020. Figure 2's smoothing function at the start of 2012 is near $1,000,000 and at the end of 2021, it is also near $1,000,000. The dip is still present and is more obvious. The dip now starts at 2016 rather than 2018. This is due to all the prices being roughly equivalent, so it cannot be masked by increasing prices.

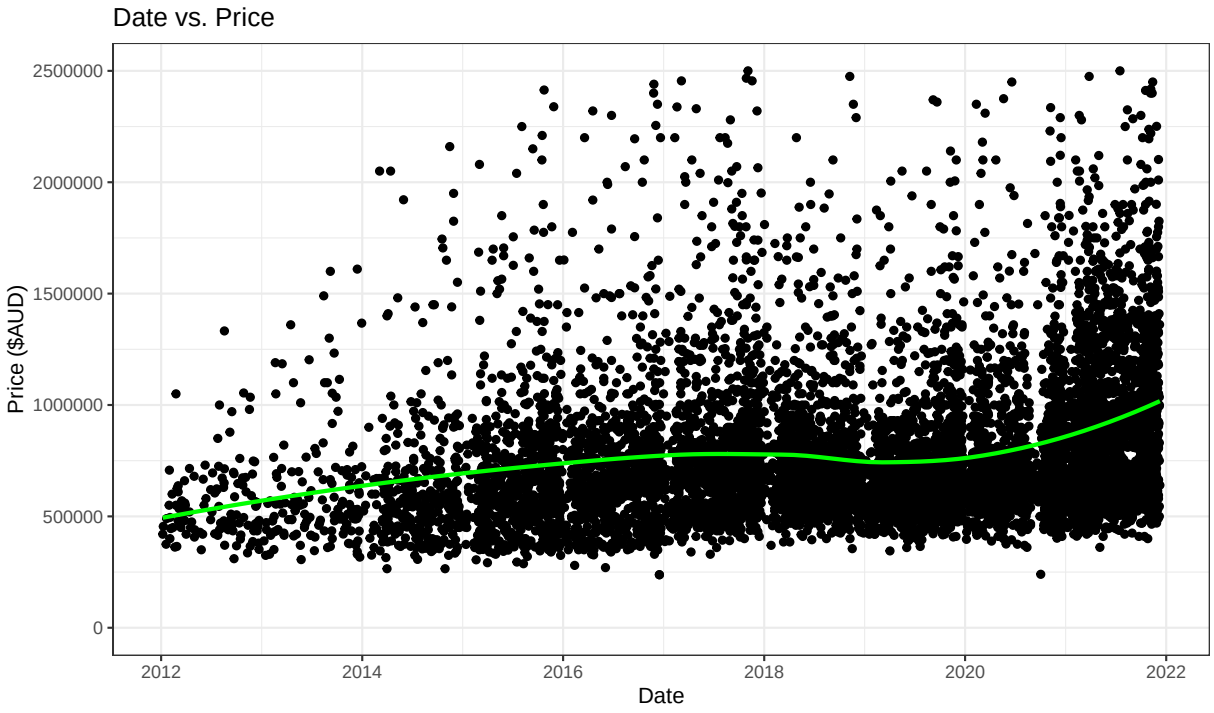


Figure 1. Sale Date vs. Non-adjusted Prices with an Approximated Smoothing Function (green).

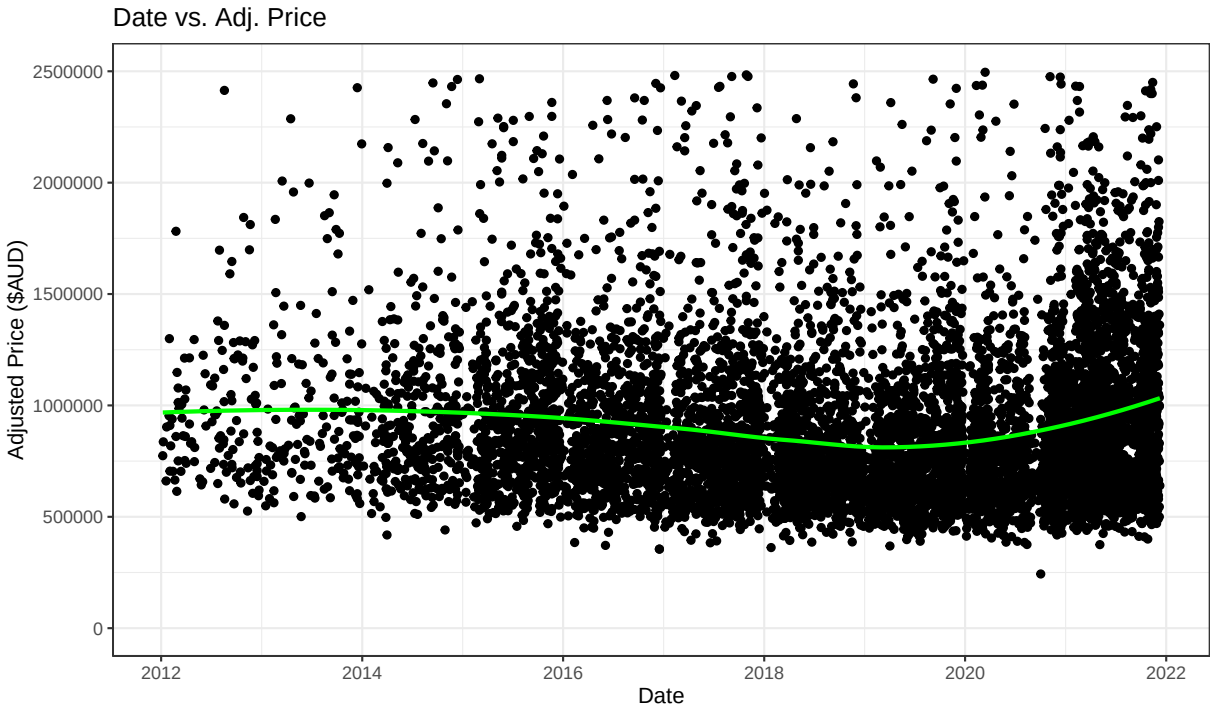


Figure 2. Sale Date vs. Adjusted Prices with an Approximated Smoothing Function (green).

2.3 Census and Suburb Data

Some basic information about suburbs was used alongside census data. The data was compiled by Michal Sniatala and can be found at https://github.com/michalsn/australian-suburbs. For each suburb the elevation, population, median income, area (in km^2), and the centre (coordinates) of the suburb were used. Using the centre of each suburb, the distances to the Melbourne CBD, Geelong, and Ballarat were calculated to see how distances to various cities impacts the price of a house. Calculating the distances on a per-suburb basis instead of per-house was a reasonable approximation for most suburbs but it can be improved by computing the true distance of each house to the cities.

3 Analysis

All analysis was done in `R` using `dyplr`, `GA` (for Genetic Algorithms), `isotree` (for Isolation Forests), and `stringr`. This analysis included the use of a Genetic Algorithm (GA) for feature selection, detection of anomalies via an Isolation Forest, and the creation of the models.

3.1 Genetic Algorithms

A metaheuristic is a general optimisation framework, where the search problem is treated as a black box, and information about the problem can only be obtained by the output of some objective, cost, or fitness function (Mirjalili et al. 2020, p. 3). A GA is one of the many metaheuristics, which also include various swarm-based algorithms such as ant-colony optimisation (Mirjalili et al. 2020, p. 3). GAs are based on an interesting approach to optimisation as they try to mimic Darwin's Natural Selection by having different generations (iterations) provide better solutions than previous generations (Givens & Hoeting 2012, p. 75). This also allows a wide variety of solutions at each generation, rather an a single one (Givens & Hoeting 2012, p. 75). Furthermore, Sivanandam & Deepa (2008, p. 343) state GAs are relatively insensitive to noise, and possess the ability to efficiently search spaces when little is known a priori, making them a powerful tool with many applications.

In GAs, every candidate solution is represented by a *chromosome* (Givens & Hoeting 2012, p. 75). A chromosome is a sequence of C symbols, from some alphabet, such as a binary

alphabet, which was used in this project. Furthermore, there are two important operations, called genetic operators, which can be done on chromosomes, namely *crossover* and *mutation* (Givens & Hoeting 2012, pp. 76-77).

Crossover is when a random position is selected between two entries in both parent chromosomes, and both are cut at that position (Givens & Hoeting 2012, p. 77). This gives a left and right segment for each parent. An offspring is created when the left segment of one parent is glued to the right of the other (Givens & Hoeting 2012, p. 77). Crossover can be seen in Figure 3, where the cut is made between the second and third values. The first offspring is created by gluing the left segment of the first parent to the right of the second.

Mutation is when values are randomly changed in the offspring given that value does not appear in either parent at the same position (Givens & Hoeting 2012, p. 77). This can be seen in Figure 4 where the third value in the offspring is a 1 but, in both parents, it is a 0.

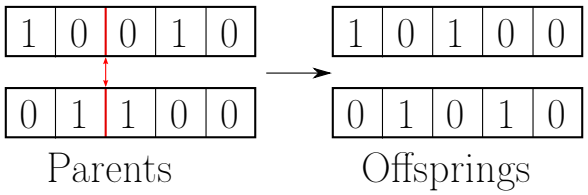


Figure 3. Crossover

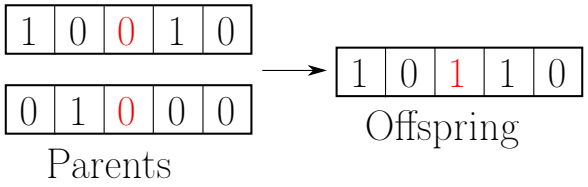


Figure 4. Mutation

In this project, GAs were used for feature selection in a nonlinear regression model. To do this, each feature in the model was a value in a chromosome where 0 meant to exclude the feature from the model, and 1 was to include it in the model (Givens & Hoeting 2012, p. 76, Sivanandam & Deepa 2008, p. 343). For example, if a model has four predictors, its chromosome will be of length 4, and an associated chromosome may be 0111, where only the first predictor is excluded from the model. For the fitness function, which is the function to be minimised, Akaike's Information Criterion (AIC) was used. From Burnham & Anderson (2003,

p. 61), it is defined as

$$\text{AIC} = -2\ln\left(\mathcal{L}\left(\hat{\theta}\middle|\, y\right)\right) + 2p$$

where $\ln\left(\mathcal{L}\left(\hat{\theta}\middle|\, y\right)\right)$ is the value of the log-likelihood at its maximum point, and p is the number of parameters in the model. This means the model is penalised when more parameters are present due to the $2p$ term, but the log-likelihood decreases when more terms are added (Burnham & Anderson 2003, p. 62). This allows there to be a balance between over and underfitting, which is the idea of parsimony (having a good fit with few features) (Burnham & Anderson 2003, p. 62). By trying to reduce the number of variables in a model, it allows for quicker prediction times.

The solution is the chromosome which represents the *best* model. This *best* regression model is then used for predicting the prices of houses. Best meaning the model with the lowest AIC.

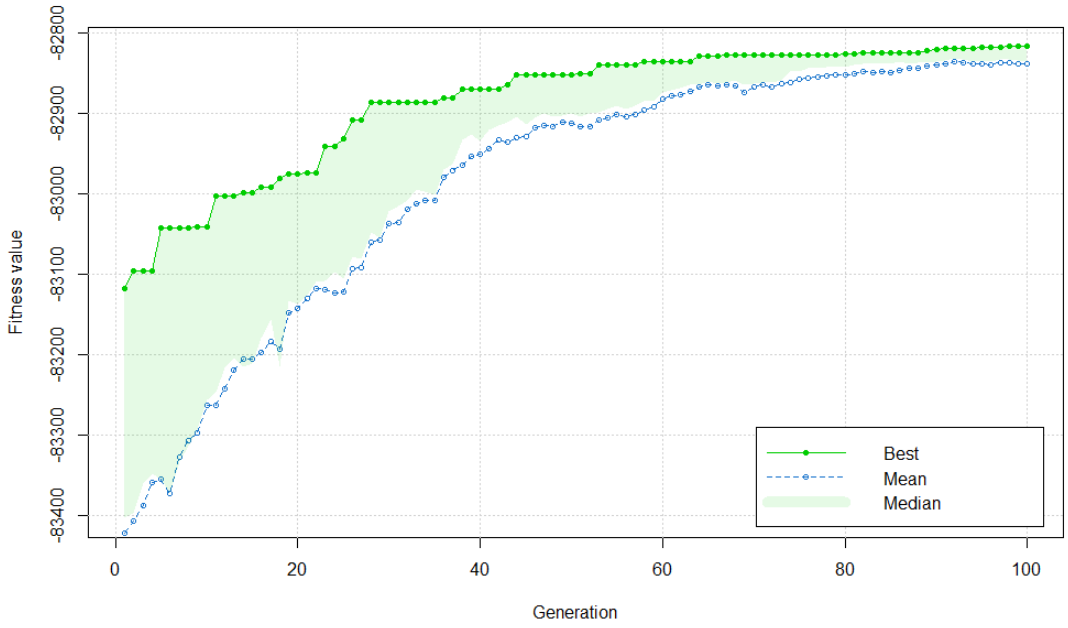


Figure 5. GA Output

Figure 5 shows the output of one of the many runs of the GA. During the earlier generations, the fitness value increases quickly, and after about 63 generations, it does not change much, meaning some near-'best' model has been found. For the selected suburbs and houses, this particular run produced some interesting results, excluding the number of bedrooms, parking

spaces, and the land size from the model, which are typically the selling points of a house. It also excluded some other features such as the interaction between the number bedrooms and bathrooms. Such results should be further investigated and interpreted alongside domain experts. The full list of included and excluded variables can be found in Appendix C.

3.2 Anomalies

Some houses can have unusual features such as a high, or low, number of bedrooms, bathrooms, or land size. These houses are classified as anomalies in this project and were detected using an Isolation Forest. Isolation Forests were proposed by Liu et al. (2008) and aim to detect anomalies through trees which differs from other methods that typically use clustering or classification methods. Isolation trees only require a sub-sample of the data to be trained which avoids *swamping* and *masking* (Liu et al. 2008, p. 5). Swamping is when normal data is classified as anomalies, and masking is when too many anomalies exist, hiding their existence (Liu et al. 2008, p. 5). As Isolation trees are constructed using sub-sampling, these two issues can be avoided as each tree in the forest only has a small piece of the total data, where some may have no anomalies at all (Liu et al. 2008, p. 5).

The parameters used in this project were based off the recommendations found in Liu et al. (2008, p. 6), which are 100 trees in the forest and a sub-sampling size of 256. The features used in the Isolation Forest were the number of bedrooms, bathrooms, parking spaces, and land size. The Isolation Forest detected 52 anomalies in the dataset, and the top three anomalies, based on their anomaly scored defined by Liu et al. (2008), are shown in Table 1.

Bedrooms	Bathrooms	Parking Spaces	Land Size
4	2	10	9496
5	2	22	5001
4	2	10	6367

Table 1. Top three anomalies

It is obvious that having 9500m^2 of land is not normal, and neither is having 22 parking spaces. Even though the top three anomalous houses share the same strange features, it is

not reasonable to say anomalous houses only have a high value for some feature (or features). There was a house which had 3 bedrooms, 1 bathroom, 8 parking spaces, and 302m^2 of land, meaning the Isolation Forest can find outliers with small values for features too.

3.3 Models

There were four models considered, a nonlinear regression model, a historical model which used previous sales of a house, a hybrid model which used both the regression and historical model, and a weighted hybrid model.

The regression model was nonlinear as it incorporated squared terms as well as interactions. As mentioned in Section 3.1, this regression model was selected using a GA. The prediction from the regression model will be denoted by P_r.

The historical model was the mean of the scaled historical sales for a property. This scaling was done through median sales as shown in Section 2.2. Suppose a house has n historical sales, and each historical sale has a price H_i, and its respective scaling ratio is R_i.
The historical model's estimate is then

$$P_H = \frac{1}{n}\sum_{i=1}^{n} R_i H_i$$

The hybrid model is simply the mean of the regression and historical models.

$$P_{\text{Hybrid}} = \frac{1}{1+n}\left(P_r + \sum_{i=1}^{n} R_i H_i\right)$$

The weighed hybrid model computes weights reflecting sale dates. It is expected that higher weights correspond to more recent historical sales of a house. The idea behind this was a sale in the past few years reflects the price today more so than five or ten years ago. A historical sale could fall into one of five bins, where each bin was 2 years in length. The first bin's range was 2012 to 2013, and the last bin's range was 2020 to 2021. The historical sales which fall into the bins have been scaled, and if more than one historical sale for a house falls into one bin, their mean is taken, meaning only one value is allowed per bin. Let w_r denote the weight of the regression price, $w_1, w_2, \ldots, w_5$ be the weights for bins 1 through 5, B_i for bin i, and $\overline{P_{H_{B_i}}}$ be the mean historical price for bin i.

$$\mathrm{P}_{\text{Wt. Hybird}} = \frac{w_r P_r + \sum_{i=1}^{5} w_i \overline{P_{H_{B_i}}}}{w_r + \sum_{i=1}^{5} w_i \mathbf{1}_{(\text{Sale in } B_i)}}$$

The weights were chosen by minimising the mean squared error (MSE).

4 Results and Discussion

The models were trained and tested using two datasets, one without and one with anomalies. These anomalies were detected using the method outlined in Section 3.2. From these datasets, 20 training and testing sets were randomly selected, and there were two response variables used, the non-adjusted price and the adjusted price. The training and testing datasets without anomalies differed than those with anomalies. As the testing set consisted of sales in 2021, the non-adjusted price and adjusted price for a house should not differ much. Once the models were trained, they were tested, and their root mean squared errors (RMSEs) were calculated for each testing set. An additional test was carried out on the anomalous houses, also using the RMSE as the metric. For the regression model, the best model (the one with the lowest AIC) was chosen out of the 20 iterations, and the weighted hybrid model used the average weights out of the 20 iterations when tested against the anomalous houses.

4.1 Results for the Subset of Data Without Anomalies

The models were trained and tested without anomalies present. As seen in Table 2, the regression and historical models perform quite poorly in comparison to the hybrid models for both the adjusted and non-adjusted prices. The weighted hybrid performs the best as it is based on recency.

When tested against the anomalous houses (Table 3), the regression model performs very poorly. As there were no anomalies present in the training data, it is very sensitive to outliers. The hybrid models performed poorly as they depend on the regression model, except for the adjusted price's weighted hybrid model, which had a low weight for the regression price in the model. The regression model had a weight of 15.5% for the non-adjusted price's weighted

hybrid model, whereas the adjusted price's was 1%. The historical model performs the best in both cases, as the historical price does not depend on the regression model.

	Adj. Price	Price
Regression	230,084.9	244,904.8
Historical	232,726.3	228,362.5
Hybrid	173,597.9	160,993.6
Weighted Hybrid	122,887.7	96,824.33

Table 2. RMSE without anomalies

	Adj. Price	Price
Regression	19,408,560	14,905,786
Historical	468,997.4	684,876
Hybrid	9,698,891	7,419,661
Weighted Hybrid	521,626.3	4,360,937

Table 3. RMSE for anomalous houses

4.2 Results for the Full Data Including Anomalies

The models were trained and tested again with anomalies present in both sets. Table 4 shows an overall improvement to Table 2 although the regression model performed worse, as outliers were present in the training and testing data. The historical and hybrid models for both prices improved, and the weighted hybrid model did not change too much.

When tested against the anomalies (Table 5), the regression RMSE is significantly lower than in Table 3 as the regression model was more robust to outliers. The historical RMSE did not change as the model is independent of any other data, except for the historical sales of a house. The hybrid models perform significantly better as the regression model performed better. The regression weight for the weighted hybrid models were very close to those when anomalies were not present in the training and testing sets.

	Adj. Price	Price
Regression	259,199.9	268,861.2
Historical	129,136.4	133,462.9
Hybrid	147,693.5	130,071.5
Weighted Hybrid	127,255.5	95,887.57

Table 4. RMSE with anomalies

	Adj. Price	Price
Regression	1,529,487	1,257,375
Historical	468,997.9	684,876
Hybrid	768,481.4	501,102.6
Weighted Hybrid	268,483.4	476,341

Table 5. RMSE for anomalous houses

From this, it is fair to conclude that having houses that are considered anomalies present

in the training sets improves the performance of the hybrid models.

4.3 Limitations of Models

A limitation of the historical, hybrid, and weighted hybrid models is the dependence on previous sales of a house. This meant if a house had no previous sales, the historical and weighted hybrid models fail, and the hybrid model is just the predicted price from the regression model.

In the weighted hybrid model, the bins are quite large as the ratios used to scale prices were only available in yearly intervals from 2012 until 2021, where it was monthly. If monthly median sales were available for many suburbs, these bins could become smaller as the monthly data would provide more granularity when performing the adjustments on prices.

In the regression model, there were very few terms that could be used due to the lack of features of the houses in the online data. This meant features such as an in-ground pool or solar panels could not be included in the model, which may increase the price of a house. Although it could not be tested, only one feature, built-in wardrobes, was common enough to include as a feature. Having more features, suburbs, and houses should help to obtain more complex models and reliable results.

5 Conclusion

The price of a house depends on many factors and cannot be predicted well using a regression model on its own. Incorporating previous sales of a house alongside information about the suburb it is in can vastly improve the quality of the prediction. Houses that are considered anomalies require a different approach to predicting prices as regression models fail, whereas using historical sales and hybrid approaches provide reasonable results.

5.1 Future Improvements

This project can be improved in several directions. For anomaly detection, more models and algorithms could be explored and compared. This would allow for anomalies to be more well defined as they would be detected multiple times by the different methods. Using information about suburbs, some ranking could be applied to them, and the same can be done for schools.

Houses in a high-ranking suburb, or close to high-ranking schools, would have their prices adjusted to cater for this. In addition to this, models could be created on a per-suburb basis as more affluent suburbs, such as Kew, differ greatly to suburbs such as Werribee. Furthermore, the year a house was built, and any renovation history were not considered in the models as no data was available. As renovations typically increase the value of a house, this information would also be useful. The introduction of times series models may prove useful as the price of houses typically went up in price, with a dip from 2018 to 2020.

6 Acknowledgements

I would like to thank Associate Professor Andriy Olenko for supervising this project and providing assistance when needed, as well as AMSI for funding the scholarship. Additionally, I would like to thank Faraz Hesari for providing ideas on how to scrape properties from Domain. Also, Michal Sniatala for making general information and basic census data for suburbs publicly available.

7 References

Burnham, KP & Anderson, DR 2003, *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*, Springer, New York.

Givens, GH & Hoeting, JA 2012, *Computational Statistics*, Wiley Series in Computational Statistics, 2nd edn, Wiley, Chicester.

Liu, FT, Ting, KM & Zhou, Z 2008, 'Isolation Forest', *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422.

Mirjalili, S, Faris, H & Aljarah, I 2020, *Evolutionary Machine Learning Techniques Algorithms and Applications: Algorithms and Applications*, Springer.

Sivanandam, SN & Deepa, SN 2008, *Introduction to Genetic Algorithms*, 1st edn, Springer, Berlin, Heidelberg.

A List of Suburbs

Below is the list of suburbs used in this project.

Airport West, Albert Park, Alphington, Altona, Balwyn, Box Hill North, Broadmeadows, Brunswick, Bundoora, Carlton, Coburg, Croydon, Diamond Creek, Doreen, Dromana, Essendon, Footscray, Glen Iris, Greensborough, Heidelberg, Hoppers Crossing, Keilor, Kew, Maribyrnong, McCrae, Melbourne, Mill Park, Preston, Reservoir, Richmond, Ringwood, Roxburgh Park, St Kilda, Sunshine, Surrey Hills, Toorak, Werribee, Williamstown

B Data Examples

A house is stored as:

```
(address,suburb,salePrice,saleDate,bedrooms,bathrooms,
parking_spaces,land_size,land_size_unit,propertyType,url,
features,schoolsDistance,schoolsCount,salesHistory,
built_in_wardrobes)
:
(address,Airport West,1099000,11-Sep-2021,2,1,1,696,m2,House,url,
Gas*-Internal Laundry*-Secure Parking*-Dishwasher*-Shed*-Garden*,
0.7-2.4-0.6,3,Sep/2021/1099000.0/SOLD,0)
```

A suburb is stored as:

```
(postcode,type,elevation,population,median_income,sqkm,latitude,
longitude,Dist_CBD,Dist_Geelong,Dist_Ballarat)
:
(3042,Major Urban locality,77,7564,35516,3.675,-37.72265,144.8782,
12.600146,66.36000,92.51653)
```

C GA Included and Excluded Variables

Included:

bathrooms, built_in_wardrobes, elevation, population, median_income, sqkm, Dist_CBD, Dist_Geelong, Dist_Ballarat, bedrooms^2, bathrooms^2, parking_spaces^2, land_size^2, built_in_wardrobes^2, elevation^2, population^2, median_income^2, sqkm^2, Dist_Geelong^2, Dist_Ballarat^2, bedrooms:parking_spaces, bedrooms:elevation, bedrooms:median_income, bedrooms:Dist_Geelong, bathrooms:land_size, bathrooms:elevation, bathrooms:population, bathrooms:Dist_CBD, bathrooms:Dist_Geelong, bathrooms:Dist_Ballarat, parking_spaces:built_in_wardrobes, parking_spaces:elevation, parking_spaces:population, parking_spaces:sqkm, land_size:built_in_wardrobes, land_size:elevation, land_size:population, land_size:median_income, land_size:sqkm, land_size:Dist_CBD, land_size:Dist_Geelong, land_size:Dist_Ballarat, built_in_wardrobes:median_income, built_in_wardrobes:sqkm, built_in_wardrobes:Dist_CBD, elevation:population, elevation:median_income, elevation:sqkm, elevation:Dist_CBD, elevation:Dist_Geelong, elevation:Dist_Ballarat, population:median_income, population:sqkm, population:Dist_CBD, population:Dist_Geelong, median_income:Dist_CBD, median_income:Dist_Geelong, median_income:Dist_Ballarat, sqkm:Dist_CBD, sqkm:Dist_Geelong, sqkm:Dist_Ballarat, Dist_CBD:Dist_Geelong, Dist_CBD:Dist_Ballarat, Dist_Geelong:Dist_Ballarat

Excluded:

bedrooms, parking_spaces, land_size, Dist_CBD^2, bedrooms:bathrooms, bedrooms:land_size, bedrooms:built_in_wardrobes, bedrooms:population, bedrooms:sqkm, bedrooms:Dist_CBD, bedrooms:Dist_Ballarat, bathrooms:parking_spaces, bathrooms:built_in_wardrobes, bathrooms:median_income, bathrooms:sqkm, parking_spaces:land_size, parking_spaces:median_income, parking_spaces:Dist_CBD, parking_spaces:Dist_Geelong, parking_spaces:Dist_Ballarat, built_in_wardrobes:elevation, built_in_wardrobes:population, built_in_wardrobes:Dist_Geelong, built_in_wardrobes:Dist_Ballarat, population:Dist_Ballarat, median_income:sqkm

D Code

The full code to scrape data can be found at https://github.com/AdamBilchouris/Domain-Scraping.

The R scripts can be found at https://github.com/AdamBilchouris/AMSI_Code.

E Code Fragments

Below are some key code fragments.

E.1 Feature Selection

```
y <- 'price'
formulaNew <-  paste0(".*. + I(", names(data5)[names(data5)!=y], "^2)+",
                      collapse="") %>%
paste(y, "~", .) %>%
substr(., 1, nchar(.)-1) %>%
as.formula

modNormalNew <- lm(formulaNew, data=train)

x <- model.matrix(modNormalNew)[, -1]
y <- model.response(model.frame(modNormalNew))

fitness <- function(s)
{
        inc <- which(s == 1)
        X <- cbind(1, x[, inc])
        mod <- lm.fit(X, y)
        class(mod) <- 'lm'
        -AIC(mod)
}

library(GA)
GANew <- ga("binary", fitness=fitness, nBits=ncol(x),
names=colnames(x), monitor=plot, popSize=100)

selectedGANew <- which(GANew@solution[1, ] == 1)
selectedNamesGANew <- names(selectedGANew)

formulaStrNew <- paste('price ~ ', selectedNamesGANew[1], sep='')
for(i in 2:length(selectedNamesGANew)) {
```

```
        formulaStrNew <- paste(formulaStrNew,
                               selectedNamesGANew[i], sep='+')
}
newFormulaNew <- as.formula(formulaStrNew)
modSelectedNew <- lm(newFormulaNew, data=train)
```

E.2 Anomaly Detection

```
library(isotree)
isoSeed <- as.numeric(as.POSIXct(Sys.time(), origin = "1970-01-01"))
isoForestAll <- isolation.forest(data5[, c(2:13)], ntrees=100,
                                 sample_size=256, seed=isoSeed)
isoPredAll <- predict(isoForestAll, data5[, c(2:13)])
predDfAll <- data.frame(isoPredAll)
names(predDfAll)[names(predDfAll) == 'isoPredAll'] <- 'pred'
predDfAll[, 'index'] <- rownames(predDfAll)
predDfAllFilter <- predDfAll[predDfAll['pred'] > 0.6, ]
orderedAll <- predDfAllFilter[order(predDfAllFilter[, 'pred'],
                                    decreasing=T),]
indicesAll <- c(orderedAll$index)
dAll <- data[unlist(indicesAll), ]
dAll['pred'] <- orderedAll$pred
dAll <- dAll[, c(1, 3, 5:8, 15, 18)]
dataIsoAll <- data5[indicesAll, ]
```