MSA Phase 1 Submission

Josh Looker

25 July 2020

Executive Summary

Using a dataset provided by NZMSA as well as further data from the 2018 Census (from StatsNZ) and the University of Otago's 2018 Deprivation Index. The NZSMA dataset contains location and building data for each (house) observation as well as demographics for the surrounding location. It was desired to use these attributes to predict the CV of each house.

The analysis below is based on the 1050 house data points recorded, each with a corresponding 17 variables. The response variable is the CV of each house, with the remaining 16 variables being explanatory.

Following initial data analysis, different multi-linear regression models were fitted to the dataset, with the 2 best models identified based on training accuracy and their simplicity.

Note that relevant values for each model can be found in Appendix 1: Model Building, and data manipulation code in Appendix 2: Data Manipulation

Initial Data Analysis

```
summary(df_model)
```

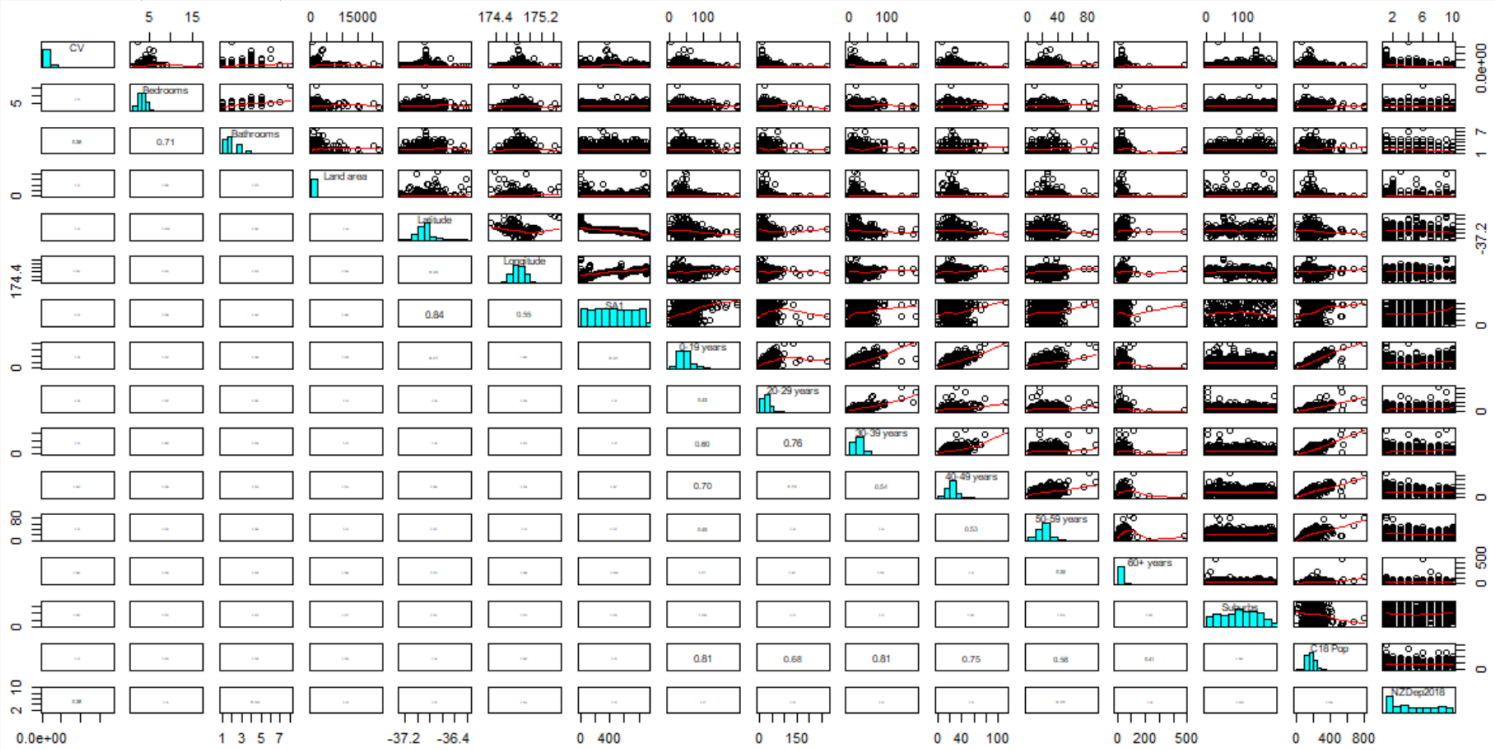
```
##        CV              Bedrooms       Bathrooms       Land area
##  Min.   :  270000   Min.   : 1.00   Min.   :1.000   Min.   :   40.0
##  1st Qu.:  780000   1st Qu.: 3.00   1st Qu.:1.000   1st Qu.:  323.0
##  Median : 1080000   Median : 4.00   Median :2.000   Median :  571.5
##  Mean   : 1388544   Mean   : 3.78   Mean   :2.074   Mean   :  857.0
##  3rd Qu.: 1600000   3rd Qu.: 4.00   3rd Qu.:3.000   3rd Qu.:  825.0
##  Max.   :18000000   Max.   :17.00   Max.   :8.000   Max.   :22240.0
##
##     Latitude        Longitude          SA1         0-19 years
##  Min.   :-37.27   Min.   :174.3   7001409:   8   Min.   :  0.00
##  1st Qu.:-36.95   1st Qu.:174.7   7009314:   5   1st Qu.: 33.00
##  Median :-36.89   Median :174.8   7004358:   4   Median : 45.00
##  Mean   :-36.89   Mean   :174.8   7005632:   4   Mean   : 47.54
##  3rd Qu.:-36.86   3rd Qu.:174.9   7001422:   3   3rd Qu.: 57.00
##  Max.   :-36.18   Max.   :175.5   7003826:   3   Max.   :201.00
##                                   (Other):1021
##   20-29 years      30-39 years   40-49 years      50-59 years
##  Min.   :  0.00   Min.   :  0   Min.   :  0.00   Min.   : 0.0
##  1st Qu.: 15.00   1st Qu.: 15   1st Qu.: 18.00   1st Qu.:15.0
##  Median : 24.00   Median : 24   Median : 24.00   Median :21.0
```

```
##  Mean   : 28.92   Mean   : 27   Mean   : 24.13   Mean   :22.6
##  3rd Qu.: 36.00   3rd Qu.: 33   3rd Qu.: 30.00   3rd Qu.:27.0
##  Max.   :270.00   Max.   :177   Max.   :114.00   Max.   :90.0
##
##    60+ years              Suburbs      NZDep2018
##  Min.   :  0.00   Remuera      : 61   Min.   : 1.000
##  1st Qu.: 18.00   Manurewa     : 38   1st Qu.: 2.000
##  Median : 27.00   Papatoetoe   : 29   Median : 5.000
##  Mean   : 29.35   St Heliers   : 29   Mean   : 5.066
##  3rd Qu.: 36.00   Mount Eden   : 26   3rd Qu.: 8.000
##  Max.   :483.00   Mount Roskill: 26   Max.   :10.000
##                   (Other)      :839
```

Looking at the data, we can see that the average number of bedrooms is 3.78, with a large upper tail of upto 17 bedrooms. For bathrooms, the expected number is 2.074 with a smaller upper tail of upto 8. Most land area sizes range between 323-825m2, with a small minimum of 40m2 and what is likely to be a large outlier of 22240m2 as the maximum. Latitude and longitude ranges are small, due to the relatively small size of Auckland. The average number in each population group (for each statistical area) is 47.54, 28.92, 27, 24.13, 22.6 and 29.35 respectively, with the 60+ year group having the largest range with a maximum of 483 (this data point is likely to be a rest home or similar and could be considered an outlier). Most statistical areas will have be between 138 to 207.8 people (according to the 2018 census) in them, though there is an area with 789 people which may indicate an apartment building, retirement village or something similar (and maybe an outlier).

Correlation and Pattern Analysis

```
pairs20x(df_model)
```



Looking at the CV histogram, it seems clear that the data should be logged as it has an obvious right skew (also mirrored in the scatter plots between age ranges and CV, and land area and CV). This may also account for the low correlation values between CV and the other variables. It is worth noting that the deprivation index and number of bathrooms are the most highly correlated to the CV value. Between the other variables, the C18 population is highly correlated with each of the age group population data (this is logical as higher population demographics means a higher total population) and the number of bedrooms and bathrooms is also highly correlated (likely because more bedrooms means more residents which would require more bathrooms).

Linear Regression Model

Looking at the original model, although we have a high R2 value, and thus have a very close-fitting model, we can see from the p-values that this is because statistical areas where used as the primary predictor for CV. In reality, real-estate companies and potential buyers/sellers may not have access to the exact SA1 that the houses they are interested in are located in, so this variable is unlikely to be able to be used in reality. The variable for address was also dropped (before any models were formed) as this would lead to overfitting in the multi-linear regression model as the model would simply predict based on the exact address and not on the general attributes of a house.

Thus, the second model was fitted (without SA1 and Address) and led to a very low R2 value of 0.4876, far too low for prediction. This is most likely due to the logged relationship discussed in the previous section between the CV and predictor attributes.

Therefore a final model was fitted between log(CV) and other attributes leading to a more respectable R2 value of 0.7061 far more suitable for prediction. A residuals plot (fitted values versus residuals) shows normality of residuals and also suggests a reasonable fit of the model. It is worth noting that the correlation between some of the attributes (as discussed in the previous section) may require interaction terms to be fitted (namely between bedrooms and bathrooms, and the demographic groups and population) but this was considered unnecessary with respect to the small accuracy gains it may have produced.

Two other models dropping longitude and then latitude and longitude respectively were considered for simplicity and only reduced accuracy by 0.0000 and 0.002 respectively. Thus, if model simplicity is desired, they may also be considered a good final model (most of the predictive power of those attributes are likely covered in suburbs attribute).

Conclusion

This analysis shows that the logged CV of a house can be predicted with reasonable confidence by the houses bedroom and bathroom number, land area, surrounding population demographic and suburb/geographic location. This model has a training accuracy of between 0.7041 to 0.7061 depending on whether the user values simplicity (of understanding) or prediction accuracy the most.

Appendix 1: Model Building

```
model <- lm(CV~.,data=df_model)
options(max.print=38)
summary(model)

##
## Call:
## lm(formula = CV ~ ., data = df_model)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3490614        0        0        0  3490614
##
## Coefficients: (187 not defined because of singularities)
##                                          Estimate Std. Error t value
## (Intercept)                             5.029e+08  7.236e+09   0.069
## Bedrooms                               -3.698e+04  1.159e+05  -0.319
## Bathrooms                               2.952e+05  1.275e+05   2.316
## `Land area`                             5.259e+02  1.555e+02   3.381
## Latitude                               -1.343e+08  4.776e+07  -2.812
## Longitude                              -3.059e+07  3.932e+07  -0.778
## SA17001131                             -7.701e+06  5.178e+06  -1.487
##                                         Pr(>|t|)
## (Intercept)                             0.944719
## Bedrooms                                0.750142
## Bathrooms                               0.022359 *
## `Land area`                             0.000989 ***
## Latitude                                0.005791 **
## Longitude                               0.438101
## SA17001131                              0.139705
##  [ reached getOption("max.print") -- omitted 1114 rows ]
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 775200 on 114 degrees of freedom
## Multiple R-squared:  0.9534, Adjusted R-squared:  0.5716
## F-statistic: 2.497 on 933 and 114 DF,  p-value: 4.202e-09

df_model2 = subset(df_model, select = -c(SA1))
model2 <- lm(CV~.,data=df_model2)
summary(model2)

##
## Call:
## lm(formula = CV ~ ., data = df_model2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2359765  -251198   -24291   190948 15389824
##
```

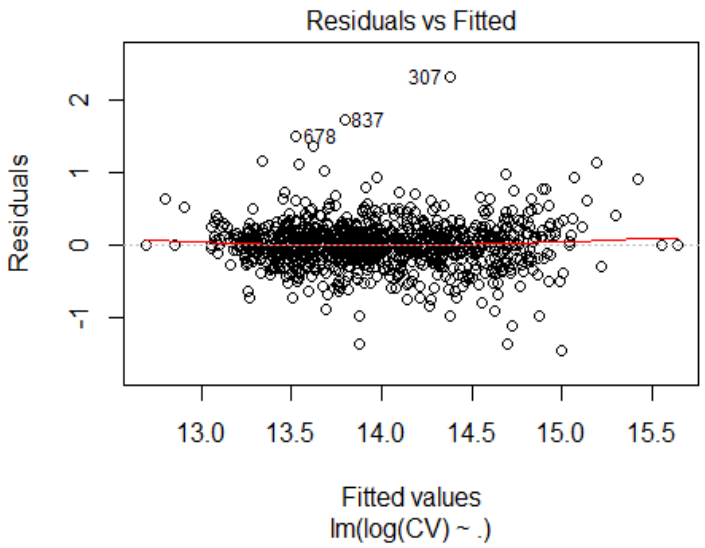
```
## Coefficients:
##                                          Estimate Std. Error t value
## (Intercept)                             6.857e+08  5.992e+08   1.144
## Bedrooms                               -6.550e+04  4.107e+04  -1.595
## Bathrooms                               3.487e+05  4.907e+04   7.106
## `Land area`                             1.620e+02  2.796e+01   5.794
## Latitude                                7.994e+06  4.161e+06   1.921
## Longitude                              -2.249e+06  3.398e+06  -0.662
## `0-19 years`                            5.152e+03  2.775e+03   1.856
##                                         Pr(>|t|)
## (Intercept)                             0.252751
## Bedrooms                                0.111128
## Bathrooms                               2.53e-12 ***
## `Land area`                             9.68e-09 ***
## Latitude                                0.055070 .
## Longitude                               0.508200
## `0-19 years`                            0.063740 .
##  [ reached getOption("max.print") -- omitted 194 rows ]
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 943800 on 847 degrees of freedom
## Multiple R-squared:  0.4863, Adjusted R-squared:  0.365
## F-statistic:  4.01 on 200 and 847 DF,  p-value: < 2.2e-16
```

```
model3 <- lm(log(CV)~.,data=df_model2)
options(max.print=76)
summary(model3)
```

```
##
## Call:
## lm(formula = log(CV) ~ ., data = df_model2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4606 -0.1374  0.0000  0.1403  2.3257
##
## Coefficients:
##                                          Estimate Std. Error t value
## (Intercept)                             1.183e+02  2.170e+02   0.545
## Bedrooms                                3.664e-02  1.488e-02   2.463
## Bathrooms                               1.093e-01  1.777e-02   6.149
## `Land area`                             8.148e-05  1.013e-05   8.045
## Latitude                                3.563e+00  1.507e+00   2.364
## Longitude                               1.461e-01  1.231e+00   0.119
## `0-19 years`                            2.071e-03  1.005e-03   2.060
## `20-29 years`                           1.639e-04  1.041e-03   0.157
## `30-39 years`                          -3.001e-03  1.294e-03  -2.319
## `40-49 years`                          -5.094e-03  1.968e-03  -2.588
## `50-59 years`                           5.229e-03  1.622e-03   3.223
```

```
## `60+ years`                                 8.374e-04  6.873e-04   1.218
## SuburbsAlfriston                            1.529e+00  7.088e-01   2.157
## SuburbsArmy Bay                             1.419e-01  4.613e-01   0.308
## SuburbsAuckland Central                     8.458e-01  4.400e-01   1.922
##                                            Pr(>|t|)
## (Intercept)                                0.585850
## Bedrooms                                   0.013965 *
## Bathrooms                                  1.20e-09 ***
## `Land area`                                2.90e-15 ***
## Latitude                                   0.018296 *
## Longitude                                  0.905551
## `0-19 years`                               0.039681 *
## `20-29 years`                              0.874953
## `30-39 years`                              0.020609 *
## `40-49 years`                              0.009809 **
## `50-59 years`                              0.001316 **
## `60+ years`                                0.223417
## SuburbsAlfriston                           0.031275 *
## SuburbsArmy Bay                            0.758452
## SuburbsAuckland Central                    0.054901 .
##  [ reached getOption("max.print") -- omitted 186 rows ]
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3419 on 847 degrees of freedom
## Multiple R-squared:  0.7061, Adjusted R-squared:  0.6367
## F-statistic: 10.17 on 200 and 847 DF,  p-value: < 2.2e-16
```

```
plot(model3, which=1)
```



```r
model4 <- lm(log(CV)~., data=df_model2[,-6])
options(max.print=50)
summary(model4)
```

```
##
## Call:
## lm(formula = log(CV) ~ ., data = df_model2[, -6])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4610 -0.1375  0.0000  0.1411  2.3268
##
## Coefficients:
##                                          Estimate Std. Error t value
## (Intercept)                             1.432e+02  5.506e+01   2.601
## Bedrooms                                3.672e-02  1.485e-02   2.472
## Bathrooms                               1.093e-01  1.776e-02   6.153
## `Land area`                             8.138e-05  1.009e-05   8.066
## Latitude                                3.547e+00  1.500e+00   2.365
## `0-19 years`                            2.062e-03  1.002e-03   2.058
## `20-29 years`                           1.508e-04  1.035e-03   0.146
## `30-39 years`                          -2.985e-03  1.286e-03  -2.322
## `40-49 years`                          -5.084e-03  1.965e-03  -2.587
## `50-59 years`                           5.224e-03  1.621e-03   3.223
##                                         Pr(>|t|)
## (Intercept)                             0.009454 **
## Bedrooms                                0.013625 *
## Bathrooms                               1.17e-09 ***
## `Land area`                             2.46e-15 ***
## Latitude                                0.018270 *
## `0-19 years`                            0.039848 *
## `20-29 years`                           0.884166
## `30-39 years`                           0.020489 *
## `40-49 years`                           0.009843 **
## `50-59 years`                           0.001316 **
##  [ reached getOption("max.print") -- omitted 190 rows ]
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3417 on 848 degrees of freedom
## Multiple R-squared:  0.7061, Adjusted R-squared:  0.6371
## F-statistic: 10.24 on 199 and 848 DF,  p-value: < 2.2e-16
```

```r
model5 <- lm(log(CV)~., data=df_model2[,c(-5:-6)])
summary(model5)
```

```
##
## Call:
## lm(formula = log(CV) ~ ., data = df_model2[, c(-5:-6)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4561 -0.1358  0.0000  0.1444  2.3351
##
## Coefficients:
##                                            Estimate Std. Error t value
## (Intercept)                               1.302e+01  3.645e-01  35.721
## Bedrooms                                  3.659e-02  1.489e-02   2.457
## Bathrooms                                 1.101e-01  1.781e-02   6.180
## `Land area`                               7.967e-05  1.009e-05   7.896
## `0-19 years`                              2.099e-03  1.004e-03   2.091
## `20-29 years`                            -5.399e-05  1.034e-03  -0.052
## `30-39 years`                            -2.839e-03  1.288e-03  -2.204
## `40-49 years`                            -5.086e-03  1.970e-03  -2.581
## `50-59 years`                             5.153e-03  1.625e-03   3.171
## `60+ years`                               8.340e-04  6.886e-04   1.211
##                                           Pr(>|t|)
## (Intercept)                                < 2e-16 ***
## Bedrooms                                  0.014208 *
## Bathrooms                                 9.95e-10 ***
## `Land area`                               8.91e-15 ***
## `0-19 years`                              0.036851 *
## `20-29 years`                             0.958359
## `30-39 years`                             0.027757 *
## `40-49 years`                             0.010020 *
## `50-59 years`                             0.001572 **
## `60+ years`                               0.226186
##  [ reached getOption("max.print") -- omitted 189 rows ]
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3426 on 849 degrees of freedom
## Multiple R-squared:  0.7041, Adjusted R-squared:  0.6351
## F-statistic:  10.2 on 198 and 849 DF,  p-value: < 2.2e-16
```

Appendix 2: Data Manipulation

Loading required packages in R for future analysis

```
library(s20x)
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 3.5.3

## -- Attaching packages ---------------------------------------------------------------
------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   0.8.3     v stringr 1.3.1
## v readr   1.3.1     v forcats 0.4.0

## Warning: package 'ggplot2' was built under R version 3.5.3

## Warning: package 'tibble' was built under R version 3.5.3

## Warning: package 'tidyr' was built under R version 3.5.3

## Warning: package 'readr' was built under R version 3.5.3

## Warning: package 'purrr' was built under R version 3.5.3

## Warning: package 'dplyr' was built under R version 3.5.3

## Warning: package 'forcats' was built under R version 3.5.3

## -- Conflicts ------------------------------------------------------------------------
---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(reticulate)

## Warning: package 'reticulate' was built under R version 3.5.3

library(reshape2)

## Warning: package 'reshape2' was built under R version 3.5.3

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths
```

Python setup for future dataframe manipulation and API usage

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import requests
import json
pd.set_option('display.max_columns', 500)
house_prices = pd.read_csv('house_prices.csv')
house_prices.head()
with open('key.txt') as fp:
  key = fp.readline()
layer_id = 104612
url = 'https://koordinates.com/services/query/v1/vector.json'
```

Using lambda and apply function/method to add in 2018 Census data to the dataframe

```
def get_cinfo(lat,lon):
  params={
    'key':key,
    'layer' : layer_id,
    'x' : lon,
    'y' : lat,
  }
  response = requests.get(url,params=params)
  return
response.json()['vectorQuery']['layers']['104612']['features'][0]['properties']['C18_CURPop'
]

house_prices['C18 Pop'] = house_prices.apply(lambda row: get_cinfo(row['Latitude'],
row['Longitude']),axis=1)
```

Appending deprivation index and census data to a new dataframe in R and writing to a csv

```
c18 <- py$house_prices['C18 Pop']
house_prices <- read_csv(file='house_prices.csv')

## Parsed with column specification:
## cols(
##   Bedrooms = col_double(),
##   Bathrooms = col_double(),
##   Address = col_character(),
##   `Land area` = col_character(),
##   CV = col_double(),
##   Latitude = col_double(),
##   Longitude = col_double(),
##   SA1 = col_double(),
##   `0-19 years` = col_double(),
##   `20-29 years` = col_double(),
##   `30-39 years` = col_double(),
##   `40-49 years` = col_double(),
##   `50-59 years` = col_double(),
##   `60+ years` = col_double(),
```

```
##   Suburbs = col_character()
## )
```

```
house_prices$`Land area` <- as.numeric(sub("\\D*(\\d+).*", "\\1", house_prices$`Land area`))
dep_index <- read_csv(file='dep_index.csv')
```

```
## Parsed with column specification:
## cols(
##   SA12018_code = col_double(),
##   NZDep2018 = col_double(),
##   NZDep2018_Score = col_double(),
##   URPopnSA1_2018 = col_double(),
##   SA22018_code = col_double(),
##   SA22018_name = col_character()
## )
```

```
house_prices['C18 Pop'] = c18['C18 Pop']
house_prices = rename(house_prices,SA1 = SA1)
dep_index = rename(dep_index,SA1 = SA12018_code)
df = left_join(house_prices,dep_index[c(1,2)],by='SA1')
df
```

```
## # A tibble: 1,051 x 16
##    Bedrooms Bathrooms Address `Land area`     CV Latitude Longitude    SA1
##       <dbl>     <dbl> <chr>         <dbl>  <dbl>    <dbl>     <dbl>  <dbl>
##  1        5         3 106 La~         714 9.60e5    -37.0      175. 7.01e6
##  2        5         3 8 Cors~         564 1.25e6    -37.1      175. 7.01e6
##  3        6         4 243 Ha~         626 1.25e6    -37.1      175. 7.01e6
##  4        2         1 2/30 H~          65 7.40e5    -36.9      175. 7.01e6
##  5        3         1 59 Isr~         601 6.30e5    -37.0      175. 7.01e6
##  6        3         1 14 Tai~         100 1.05e6    -36.9      175. 7.01e6
##  7        3         1 54 Kel~         531 2.52e6    -36.8      175. 7.00e6
##  8        3         2 39 Raw~        1024 1.40e6    -36.9      175. 7.01e6
##  9        3         2 17b Ta~          80 4.75e5    -37.0      175. 7.01e6
## 10        4         2 39a Ke~         204 6.60e5    -36.8      175. 7.00e6
## # ... with 1,041 more rows, and 8 more variables: `0-19 years` <dbl>, `20-29
## #   years` <dbl>, `30-39 years` <dbl>, `40-49 years` <dbl>, `50-59
## #   years` <dbl>, `60+ years` <dbl>, Suburbs <chr>, NZDep2018 <dbl>
```

```
write.csv(df,'expanded_dataset.csv')
```

Final data manipulation for analysis and modeling

```
df$SA1 = as.factor(df$SA1)
df$Suburbs = as.factor(df$Suburbs)
cols <- colnames(df)
cols = cols[cols!='CV']
cols = append(cols,'CV',0)
df = df[,cols]
df_model = subset(df, select = -c(Address))
```

```r
df_model[rowSums(is.na(df_model)) > 0,]
```

```
## # A tibble: 3 x 15
##       CV Bedrooms Bathrooms `Land area` Latitude Longitude SA1   `0-19 years`
##    <dbl>    <dbl>     <dbl>       <dbl>    <dbl>     <dbl> <fct>        <dbl>
## 1 1.25e6        4        NA         214    -36.8      175. 7002~           60
## 2 1.10e6        4        NA         245    -36.8      175. 7002~           60
## 3 7.40e5        1         1        2141    -36.2      175. 7001~           27
## # ... with 7 more variables: `20-29 years` <dbl>, `30-39 years` <dbl>, `40-49
## #   years` <dbl>, `50-59 years` <dbl>, `60+ years` <dbl>, Suburbs <fct>,
## #   NZDep2018 <dbl>
```

```r
df_model <- na.omit(df_model)
```