

Key Outputs

Name: Naga Satya Silpa Annadevara

A20517818

Load the housing.csv data into R:

The screenshot shows the RStudio interface. In the top-left corner, there's a file titled "Untitled1". The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help, and Addins. The top-right corner shows a "Project: (None)" indicator. The bottom navigation bar has tabs for Files, Plots, Packages, Help, Viewer, and Presentation.

In the center, the "Console" tab is active, displaying R code and its output. A red box highlights the command and its output:

```
> data = read.table(file = "c:/users/satya/OneDrive/Desktop/housing.csv", header = T, sep = ",")> str(data)
'data.frame': 5000 obs. of 7 variables:
 $ Avg.Income      : num 79545 79249 61287 63345 59982 ...
 $ Avg.Area.Age    : num 5.68 6 5.87 7.19 5.04 ...
 $ Avg.Area.Number.of.Rooms: num 7.01 6.73 8.51 5.59 7.84 ...
 $ Avg.Area.Number.of.Bedrooms: num 4.09 3.09 5.13 3.26 4.23 4.04 3.41 2.42 2.3 6.1 ...
 $ Area.Population : num 23087 40173 36882 34310 26354 ...
 $ Price           : num 1059034 1505891 1058988 1260617 630943 ...
 $ Address         : chr "208 Michael Ferry Apt. 674\nLaurabury, NE 37010-5101" "188 Johnson Views Sui
te 079\nLake Kathleen, CA 48958" "9127 Elizabeth Stravenue\nDanieltown, WI 06482-3489" "USS Barnett\nFPO AP 44820"
...> |
```

In the top-right panel, the "Environment" tab is selected. It shows a variable named "data" with the description "5000 obs. of 7 variables". A red box highlights this entry. The "Global Environment" dropdown is also visible.

1.checked for missing values

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Untitled1 data

Environment History Connections Tutorial

Project: (None)

Data

- data 5000 obs. of 7 variables
- na_count 7 obs. of 1 variable

Showing 1 to 4 of 5,000 entries, 7 total columns

Console Terminal Background Jobs

```
R 4.3.0 - ~/~>
> str(data)
'data.frame': 5000 obs. of 7 variables:
$ Avg..Area.Income : num 79545 79249 61287 63345 59982 ...
$ Avg..Area.House.Age : num 5.68 6 5.87 7.19 5.04 ...
$ Avg..Area.Number.of.Rooms : num 7.01 6.73 8.51 5.59 7.84 ...
$ Avg..Area.Number.of.Bedrooms: num 4.09 3.09 5.13 3.26 4.23 4.04 3.41 2.42 2.3 6.1 ...
$ Area.Population : num 23087 40173 36882 34310 26354 ...
$ Price : num 1059034 1505891 1058988 1260617 630943 ...
$ Address : chr "208 Michael Ferry Apt. 674\nLaurabury, NE 37010-5101"
"188 Johnson Views Suite 079\nLake Kathleen, CA 48958" "9127 Elizabeth Stravenue\nDanieltown,
WI 06482-3489" "USS Barnett\nFPO AP 44820"
> na_count = sapply(data,function(y) sum(length(which(is.na(y)))))
> na_count = data.frame(na_count)
> na_count
```

	na_count
Avg..Area.Income	0
Avg..Area.House.Age	0
Avg..Area.Number.of.Rooms	0
Avg..Area.Number.of.Bedrooms	0
Area.Population	0
Price	0
Address	0

> |

Files Plots Packages Help Viewer Presentation

2. Then handled non-numeric data. Removed column 'Address':

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Untitled1 data

Environment History Connections Tutorial

Project: (None)

Data

- data 5000 obs. of 6 variables
- na_count 7 obs. of 1 variable

Showing 1 to 1 of 5,000 entries, 6 total columns

Console Terminal Background Jobs

```
R 4.3.0 - ~/~>
> str(data)
'data.frame': 5000 obs. of 7 variables:
$ Avg..Area.Income : num 79545 79249 61287 63345 59982 ...
$ Avg..Area.House.Age : num 5.68 6 5.87 7.19 5.04 ...
$ Avg..Area.Number.of.Rooms : num 7.01 6.73 8.51 5.59 7.84 ...
$ Avg..Area.Number.of.Bedrooms: num 4.09 3.09 5.13 3.26 4.23 4.04 3.41 2.42 2.3 6.1 ...
$ Area.Population : num 23087 40173 36882 34310 26354 ...
$ Price : num 1059034 1505891 1058988 1260617 630943 ...
$ Address : chr "208 Michael Ferry Apt. 674\nLaurabury, NE 37010-5101"
"188 Johnson Views Suite 079\nLake Kathleen, CA 48958" "9127 Elizabeth Stravenue\nDanieltown,
WI 06482-3489" "USS Barnett\nFPO AP 44820" ...
> na_count = sapply(data,function(y) sum(length(which(is.na(y)))))
> na_count = data.frame(na_count)
> na_count
```

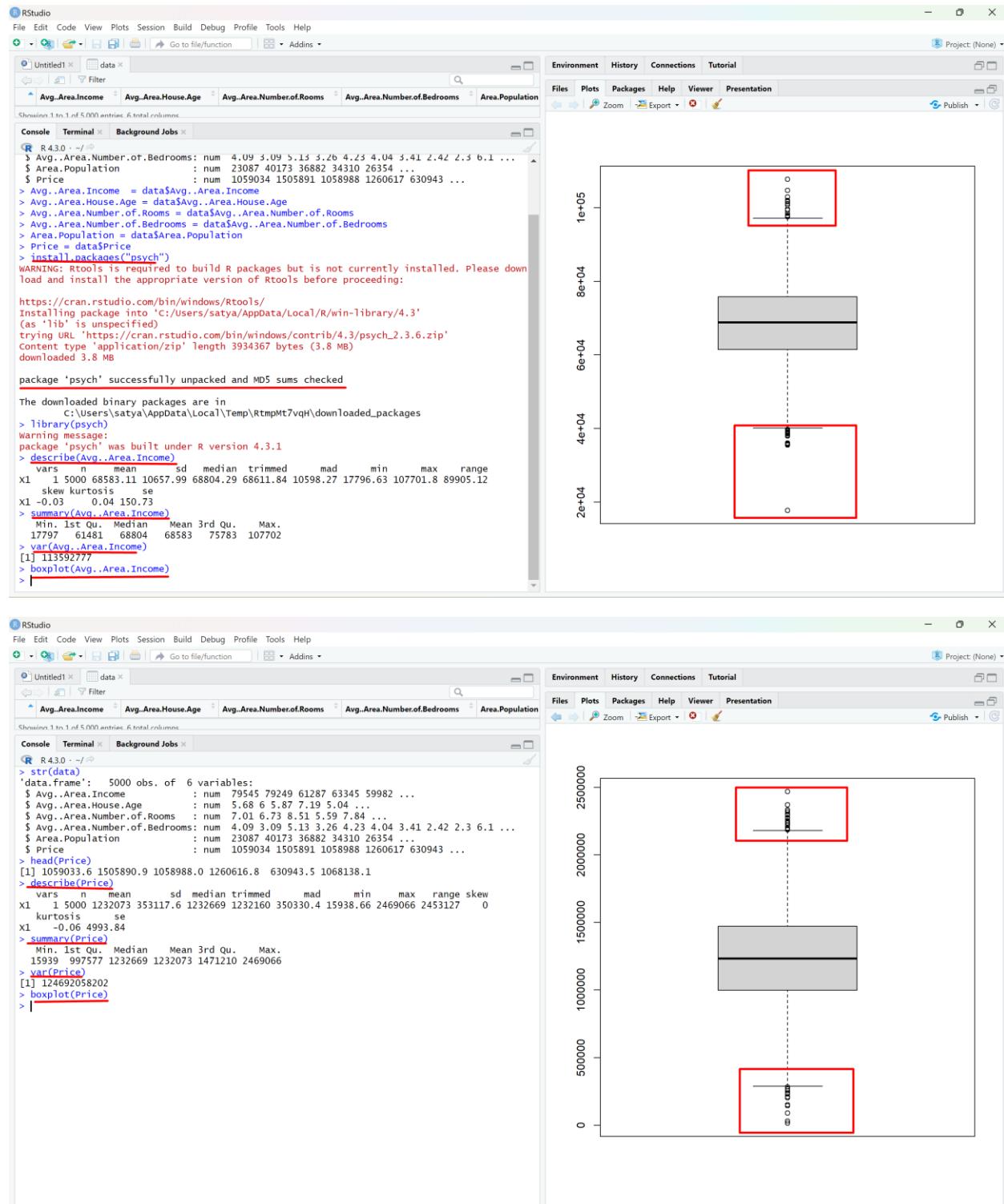
	na_count
Avg..Area.Income	0
Avg..Area.House.Age	0
Avg..Area.Number.of.Rooms	0
Avg..Area.Number.of.Bedrooms	0
Area.Population	0
Price	0
Address	0

```
> data = data[, !(names(data) %in% c("Address"))]
Error in exists(cachekey, where = .rs.workingDataEnv, inherits = FALSE) :
  invalid first argument
Error in assign(cachekey, frame, .rs.CachedDataEnv) :
  attempt to use zero-length variable name
> str(data)
'data.frame': 5000 obs. of 6 variables:
$ Avg..Area.Income : num 79545 79249 61287 63345 59982 ...
$ Avg..Area.House.Age : num 5.68 6 5.87 7.19 5.04 ...
$ Avg..Area.Number.of.Rooms : num 7.01 6.73 8.51 5.59 7.84 ...
$ Avg..Area.Number.of.Bedrooms: num 4.09 3.09 5.13 3.26 4.23 4.04 3.41 2.42 2.3 6.1 ...
$ Area.Population : num 23087 40173 36882 34310 26354 ...
$ Price : num 1059034 1505891 1058988 1260617 630943 ...
```

#column 'Address' has been removed, Now we have only 6 variables

Files Plots Packages Help Viewer Presentation

3. Performed Descriptive statistics on columns 'Price & Income':



4. Validate the claim/Assumption by using Hypothesis testing:

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Untitled1 data

Avg..Area.Income Avg..Area.House.Age Avg..Area.Number.of.Rooms Avg..Area.Number.of.Bedrooms Area.Population

Show rows 1 to 1 of 5,000 entries. 6 total columns

Console Terminal Background Jobs

```
R 4.3.0 - ~/ ~> > library(PASWR2)
Loading required package: lattice
Warning message:
package 'PASWR2' was built under R version 4.3.1
> z.test(Price,NULL,alternative = "two.sided",mu = 1230000,sigma.x= sd(Price),sigma.y=NULL,conf.level=0.95)
One Sample z-test

data: Price
z = 0.41504, p-value = 0.6781
alternative hypothesis: true mean is not equal to 1230000
95 percent confidence interval:
1222285 1241860
sample estimates:
mean of x
1232073
```

Environment History Connections Tutorial

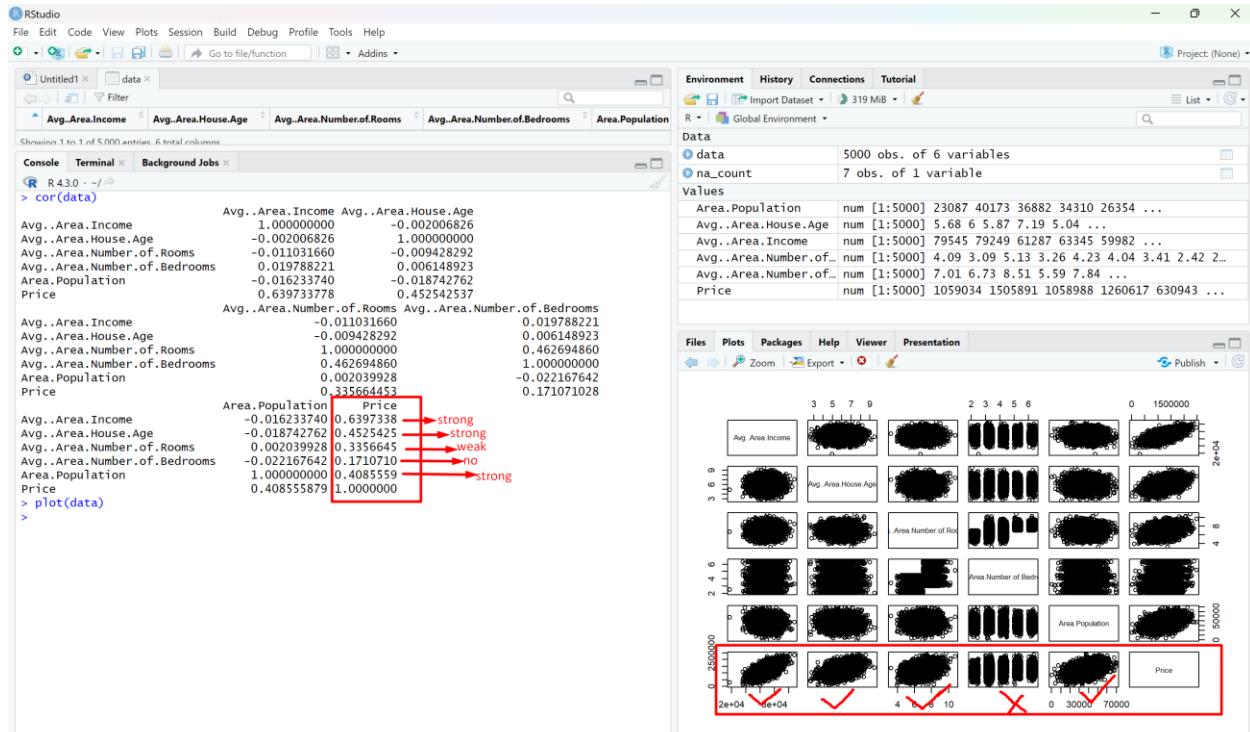
R Project: (None)

Data

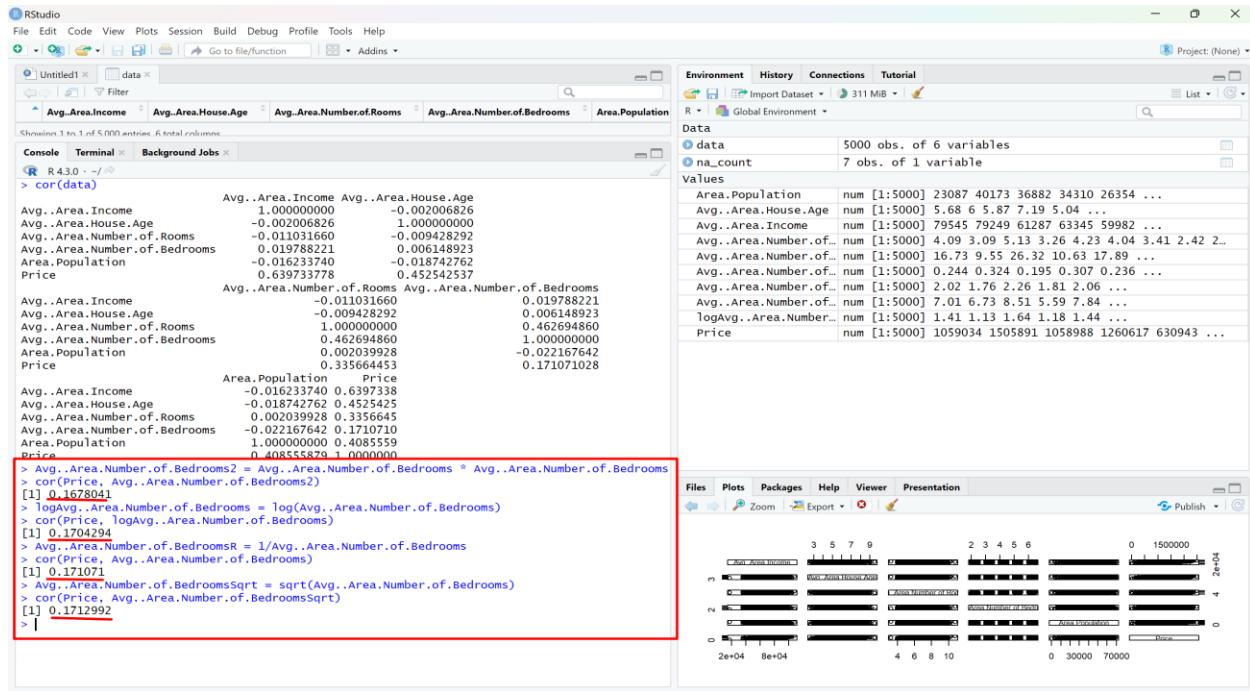
data	5000 obs. of 6 variables
na_count	7 obs. of 1 variable
Values	
Area.Population	num [1:5000] 23087 40173 36882 34310 26354 ...
Avg..Area.House.Age	num [1:5000] 5.68 6 5.87 7.19 5.04 ...
Avg..Area.Income	num [1:5000] 79545 79249 61287 63345 59982 ...
Avg..Area.Number.of.Rooms	num [1:5000] 4.09 3.09 5.13 3.26 4.23 4.04 3.41 2.42 2...
Avg..Area.Number.of.Bedrooms	num [1:5000] 7.01 6.73 8.51 5.59 7.84 ...
Price	num [1:5000] 1059034 1505891 1058988 1260617 630943 ...

Files Plots Packages Help Viewer Presentation

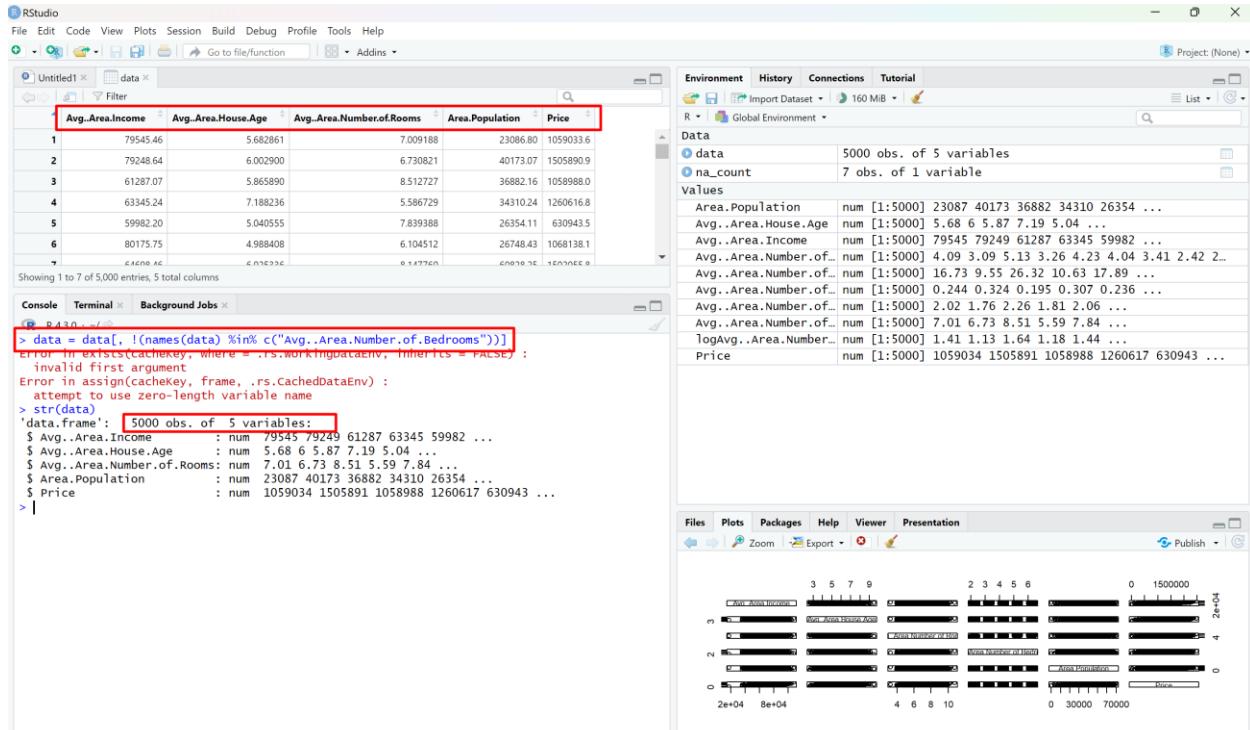
5. Handling non-linearity: Examine linear relation b/w X & y variables (check correlations)



Performed transformation on X-variable: Avg..Area.Number.of.Bedrooms



dropping the x-variable: Avg..Area.Number.of.Bedrooms:



6. data split by N-fold cross evaluation:

The screenshot shows the RStudio interface. In the top-left, there's a code editor window titled 'Untitled1' containing R code. The code includes `library(caret)`, `set.seed(123)`, and `train.control = trainControl(method= "cv", number =5)`. In the top-right, the 'Environment' pane is open, showing the global environment with objects like 'data', 'na_count', and 'train.control'. The 'train.control' object is highlighted with a red box. Below the environment pane, the 'Values' section lists various variables such as 'Area.Population', 'Avg.Area.Income', etc. At the bottom of the interface, there are tabs for 'Files', 'Plots', 'Packages', 'Help', 'Viewer', and 'Presentation'.

7. Build 5 different multiple linear regression models to compare them:

Building Full model by using all x-variables & target variable: M1

This screenshot shows the RStudio interface again. The code editor window now contains the command `> m1_full = train(Price ~ Avg.Area.Income + Avg.Area.House.Age + Avg.Area.Number.of.Rooms + Area.Population, data = data, method = "lm", trControl = train.control)` followed by `> print(m1_full)`. The output shows the results of a Linear Regression model. It indicates 5000 samples and 4 predictors. A summary of sample sizes is provided: 4000, 4000, 4000, 4000, 4000. The resampling results show RMSE, Rsquared, and MAE values. The 'train' function was run with cross-validation (5 fold). The environment pane shows the 'm1_full' object has been created. The 'Values' section below lists variables like 'Area.Population', 'Avg.Area.House...', etc. The bottom navigation bar includes 'Files', 'Plots', 'Packages', 'Help', 'Viewer', and 'Presentation'.

Building base model with just one x-variable & target variable: M2

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal Background Jobs

```
R 4.3.0 - /-
> m1_full = train(Price ~ Avg..Area.Income + Avg..Area.House.Age + Avg..Area.Number.of.Rooms + Area.Population, data = data, method = "lm", trControl = train.control)
> print(m1_full)
Linear Regression

5000 samples
4 predictor

No pre-processing
Resampling: cross-Validated (5 fold)
Summary of sample sizes: 4000, 4000, 4000, 4000, 4000
Resampling results:

RMSE R-squared MAE
101251.2 0.9178551 81478.7

Tuning parameter 'intercept' was held constant at a value of TRUE
> m2_base = train(Price ~ Avg..Area.Income, data = data, method = "lm", trControl = train.control)
> print(m2_base)
Linear Regression

5000 samples
1 predictor

No pre-processing
Resampling: cross-Validated (5 fold)
Summary of sample sizes: 4000, 4000, 4000, 4000, 4000
Resampling results:

RMSE R-squared MAE
271418.6 0.4091903 217378.1

Tuning parameter 'intercept' was held constant at a value of TRUE
>
```

Environment History Connections Tutorial

Import Dataset 393 MB

R Global Environment

Data

- data 5000 obs. of 5 variables
- m1_full Large train (24 elements, 2.3 MB)
- m2_base Large train (24 elements, 1.9 MB)
- m3 List of 24
- m4 List of 24
- m5 List of 24
- na_count 7 obs. of 1 variable
- train.control List of 27

Values

Area.Population	num [1:5000]	23087 40173 36882 34310 26354 ...
Avg..Area.House...	num [1:5000]	5.68 6 5.87 7.19 5.04 ...
Avg..Area.Income	num [1:5000]	79545 79249 61287 63345 59982 ...
Avg..Area.Number...	num [1:5000]	4.09 3.09 5.13 3.26 4.23 4.04 3.41 ...
Avg..Area.Number...	num [1:5000]	16.73 9.55 26.32 10.63 17.89 ...
Avg..Area.Number...	num [1:5000]	0.244 0.324 0.195 0.307 0.236 ...
Avg..Area.Number...	num [1:5000]	2.02 1.76 2.26 1.81 2.06 ...
Avg..Area.Number...	num [1:5000]	7.01 6.73 8.51 5.59 7.84 ...
logAvg..Area.Number...	num [1:5000]	1.41 1.13 1.64 1.18 1.44 ...
Price	num [1:5000]	1059034 1505891 1058988 1260617 63 ...

Files Plots Packages Help Viewer Presentation

Building m3 by backward method: M3

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Untitled1 data

	Avg..Area.Income	Avg..Area.House.Age	Avg..Area.Number.of.Rooms	Area.Population	Price
1	79545.46	5.682861	7.009188	23086.80	1059033.6
2	79248.64	6.002900	6.730821	40173.07	1505890.9
3	61287.07	5.865890	8.512727	36882.16	1058988.0

Showing 1 to 3 of 5,000 entries. 5 total columns

Console Terminal Background Jobs

```
R 4.3.0 - /-
> m3 = train(Price ~ Avg..Area.Income + Avg..Area.House.Age + Avg..Area.Number.of.Rooms + Area.Population, data = data, method = "leapBackward", trControl = train.control)
> print(m3)
Linear Regression with Backwards Selection

5000 samples
4 predictor

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 4000, 4000, 4000, 4000, 4000
Resampling results across tuning parameters:

rvmax RMSE R-squared MAE
2 219153.4 0.6145658 174509.38
3 158776.8 0.7976385 126698.30
4 101243.6 0.9178375 81453.42

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was rvmax = 4.
> coef(m3$finalModel, 3)
(Intercept) Avg..Area.Income Avg..Area.House.Age Area.Population
-1.772988e+06 2.145624e+01 1.644960e+05 1.521688e+01
```

Environment History Connections Tutorial

Import Dataset 365 MB

R Global Environment

Data

- base Large train (24 elements, 1.9 MB)
- data 5000 obs. of 5 variables
- full Large train (24 elements, 2.3 MB)
- m3 List of 24
- na_count 7 obs. of 1 variable
- train.control List of 27

Values

Area.Population	num [1:5000]	23087 40173 36882 34310 26354 ...
Avg..Area.House.Age	num [1:5000]	5.68 6 5.87 7.19 5.04 ...
Avg..Area.Income	num [1:5000]	79545 79249 61287 63345 59982 ...
Avg..Area.Number.of...	num [1:5000]	4.09 3.09 5.13 3.26 4.23 4.04 3.41 2.42 ...
Avg..Area.Number.of...	num [1:5000]	16.73 9.55 26.32 10.63 17.89 ...
Avg..Area.Number.of...	num [1:5000]	0.244 0.324 0.195 0.307 0.236 ...
Avg..Area.Number.of...	num [1:5000]	2.02 1.76 2.26 1.81 2.06 ...
Avg..Area.Number.of...	num [1:5000]	7.01 6.73 8.51 5.59 7.84 ...
logAvg..Area.Number...	num [1:5000]	1.41 1.13 1.64 1.18 1.44 ...
Price	num [1:5000]	1059034 1505891 1058988 1260617 630943 ...

Files Plots Packages Help Viewer Presentation

Building m4 by forward method: M4

```

R 4.3.0 - ~/r/
> m4 = train(Price ~ Avg..Area.Income + Avg..Area.House.Age + Avg..Area.Number.of.Rooms + Area.Population, data = data, method = "leapForward", trControl = train.control)
> print(m4)
Linear Regression with Forward Selection
5000 samples
4 predictor

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 4000, 4000, 4000, 4000, 4000
Resampling results across tuning parameters:

  nvmax RMSE Rsquared MAE
  2     219040.5 0.6151058 174398.97
  3     158760.0 0.7980723 126686.38
  4     101165.9 0.9180211 81401.83

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was nvmax = 4.
> coef(m4$finalModel, 3)
(Intercept) Avg..Area.Income Avg..Area.House.Age Area.Population
-1.772988e+06 2.145624e+01 1.644960e+05 1.521688e+01
> |

```

building m5 by forward method: M5

```

R 4.3.0 - ~/r/
> m5 = train(Price ~ Avg..Area.Income + Avg..Area.House.Age + Avg..Area.Number.of.Rooms + Area.Population, data = data, method = "leapSeq", trControl = train.control)
> print(m5)
Linear Regression with Stepwise Selection
5000 samples
4 predictor

No pre-processing
Resampling: cross-Validated (5 fold)
Summary of sample sizes: 4000, 4000, 4000, 4000, 4000
Resampling results across tuning parameters:

  nvmax RMSE Rsquared MAE
  2     219094.4 0.6145526 174434.45
  3     158734.3 0.7978116 126707.79
  4     101201.0 0.9180291 81465.73

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was nvmax = 4.
> coef(m5$finalModel, 3)
(Intercept) Avg..Area.Income Avg..Area.House.Age Area.Population
-1.772988e+06 2.145624e+01 1.644960e+05 1.521688e+01
> |

```

7. Model diagnosis:

F-test on M4:

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Untitled1

Console Terminal Background Jobs

```
R 4.3.0 - /~/
> m4_ftest = train(Price ~ Avg..Area.Income + Avg..Area.House.Age + Area.Population, data = data, method = "lm", trControl = train.control)
> summary(m4_ftest)

Call:
lm(formula = .outcome ~ ., data = dat)

Residuals:
    Min      1Q  Median      3Q     Max 
-555822 -106450    726 107403  575415 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.773e+06 2.175e+04 -81.53 <2e-16 ***
Avg..Area.Income 2.146e+01 2.107e-01 101.85 <2e-16 ***
Avg..Area.House.Age 1.645e+05 2.265e+03 72.63 <2e-16 ***
Area.Population 1.522e+01 2.263e-01 67.25 <2e-16 ***

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 158700 on 4996 degrees of freedom
Multiple R-squared:  0.7981, Adjusted R-squared:  0.7979 
F-statistic: 6581 on 3 and 4996 DF, p-value: < 2.2e-16
```

Environment History Connections Tutorial

base

5000 obs. of 5 variables

full

Large train (24 elements, 2.3 MB)

m3

List of 24

m4

List of 24

m4_ftest

Large train (24 elements, 2.2 MB)

m4_res

Large lm (12 elements, 1.5 MB)

m5

List of 24

na_count

7 obs. of 1 variable

train.control

List of 27

Values

Area.Population num [1:5000] 23087 40173 36882 34310 26354 ...

Avg..Area.House.Age num [1:5000] 5.68 6 5.87 7.19 5.04 ...

Avg..Area.Income num [1:5000] 79545 79249 61287 63345 59982 ...

Avg..Area.Number.of... num [1:5000] 4.09 3.09 5.13 3.26 4.23 4.04 3.41 2.42 2...

Avg..Area.Number.of... num [1:5000] 16.73 9.52 26.32 10.63 17.89 ...

Avg..Area.Number.of... num [1:5000] 0.244 0.324 0.195 0.307 0.236 ...

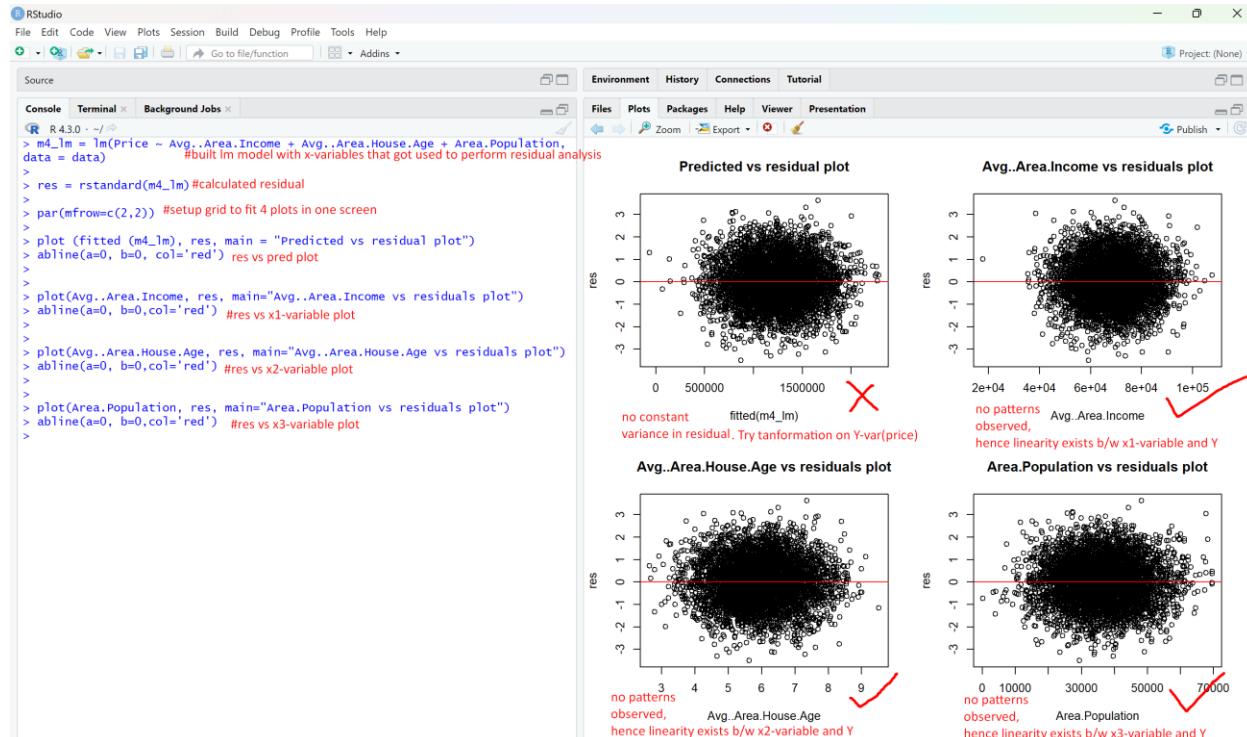
Avg..Area.Number.of... num [1:5000] 2.02 1.76 2.26 1.81 2.06 ...

Avg..Area.Number.of... num [1:5000] 7.01 6.73 8.51 5.59 7.84 ...

logAvg..Area.Number... num [1:5000] 1.41 1.13 1.64 1.18 1.44 ...

Files Plots Packages Help Viewer Presentation

Residual analysis on M4:



Transformation on Y-variable & rebuilding the model again:

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Console Terminal Background Jobs

```

> Pricelog = log(Price) #applied log transformation on Price (Y) variable
> data[, 'Pricelog'] = Pricelog #added new y-var (Pricelog) to the data
> data = data[, !(names(data) %in% c("Price"))] removed old y-var(Price) from the data
> str(data) #checked the data whether the changes made
'data.frame': 5000 obs. of 5 variables:
 $ Avg..Area.Income : num 79545 79249 61287 62345 59982 ...
 $ Avg..Area.House.Age : num 5.68 6 5.87 7.19 5.04 ...
 $ Avg..Area.Number.of.Rooms: num 7.01 6.73 8.51 5.59 7.84 ...
 $ Avg..Area.Population : num 23087 40173 36882 34310 26354 ...
 $ Pricelog : num 13.9 14.2 13.9 14 13.4 ...
>
> m4_lm = lm(Pricelog ~ Avg..Area.Income + Avg..Area.House.Age + Area.Population, data = data)
> summary(m4_lm) #built model m4 again with new
Error in summary(m4_lm) : could not find function "summary" Y-var(Pricelog)
> summary(m4_lm)

Call:
lm(formula = Pricelog ~ Avg..Area.Income + Avg..Area.House.Age +
    Area.Population, data = data)

Residuals:
    Min      Q1 Median      Q3      Max 
-3.2412 -0.0886  0.0126  0.1076  0.4898 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1.125e+01 2.422e-02 464.30 <2e-16 ***
Avg..Area.Income 1.946e-05 2.346e-07 82.96 <2e-16 ***
Avg..Area.House.Age 1.501e-01 2.522e-03 59.52 <2e-16 ***
Area.Population 1.376e-05 2.520e-07 54.62 <2e-16 ***

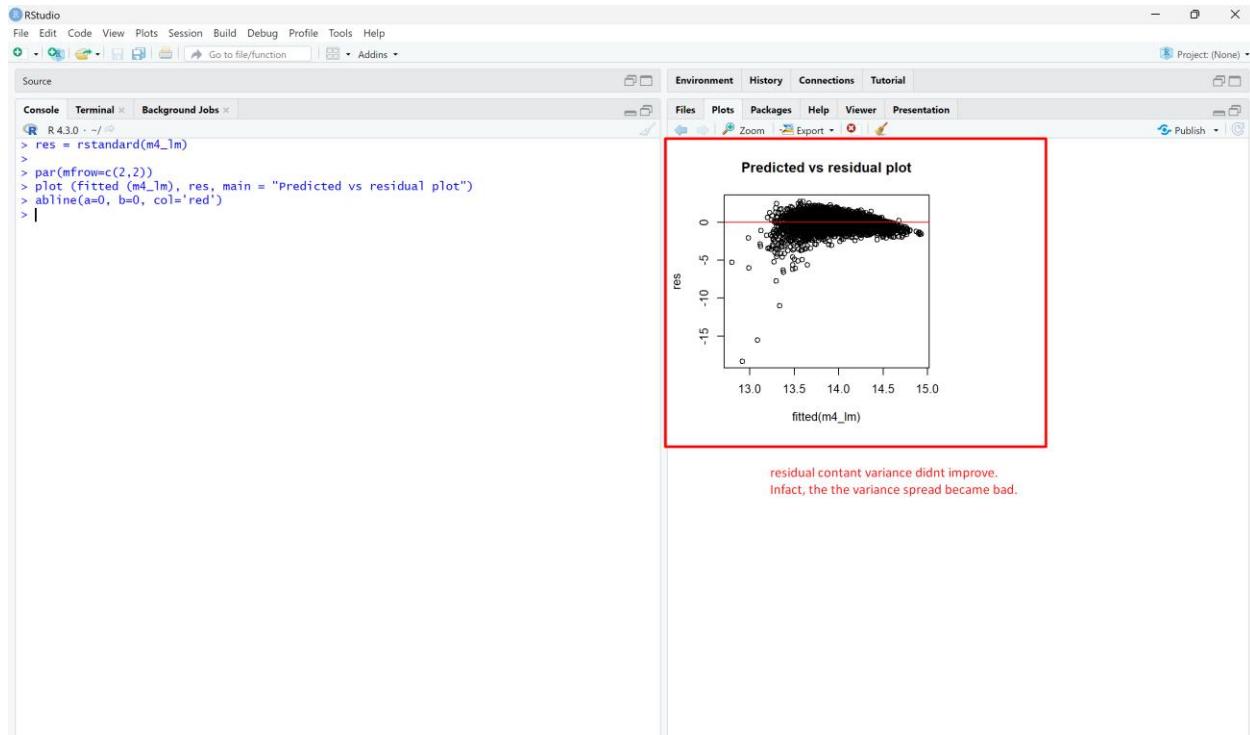
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1768 on 4996 degrees of freedom
Multiple R-squared:  0.7243, Adjusted R-squared:  0.7241 
F-statistic: 4375 on 3 and 4996 DF, p-value: < 2.2e-16

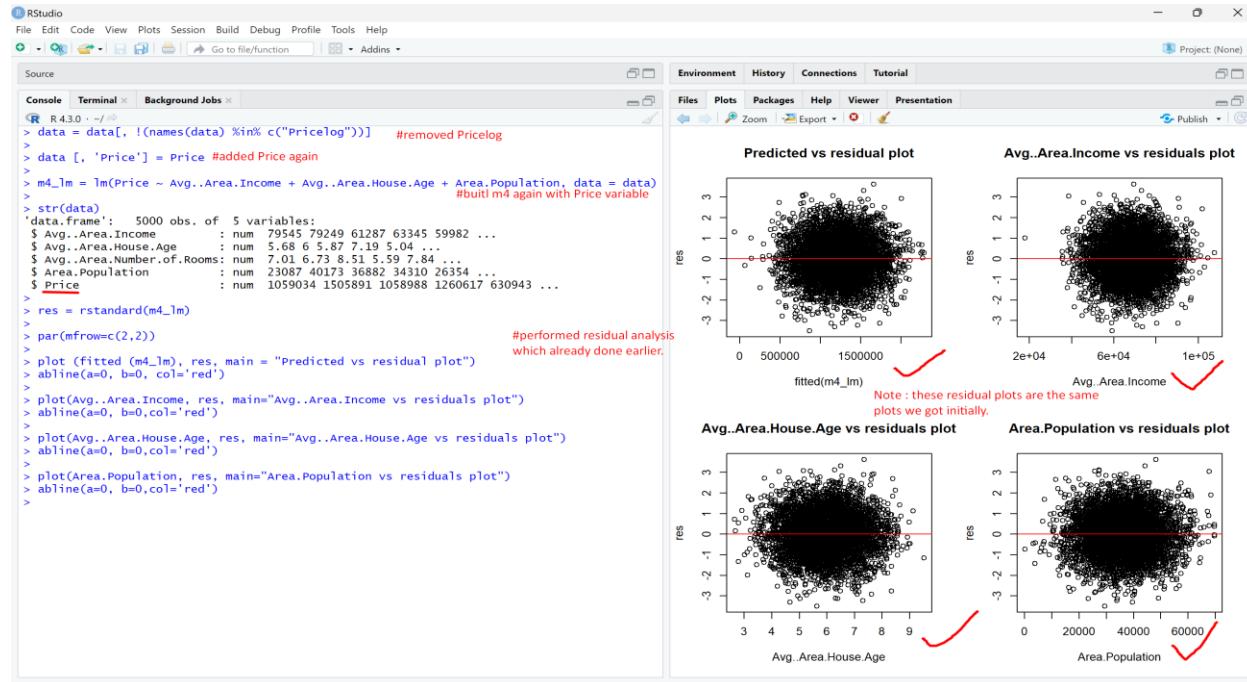
```

Files Plots Packages Help Viewer Presentation

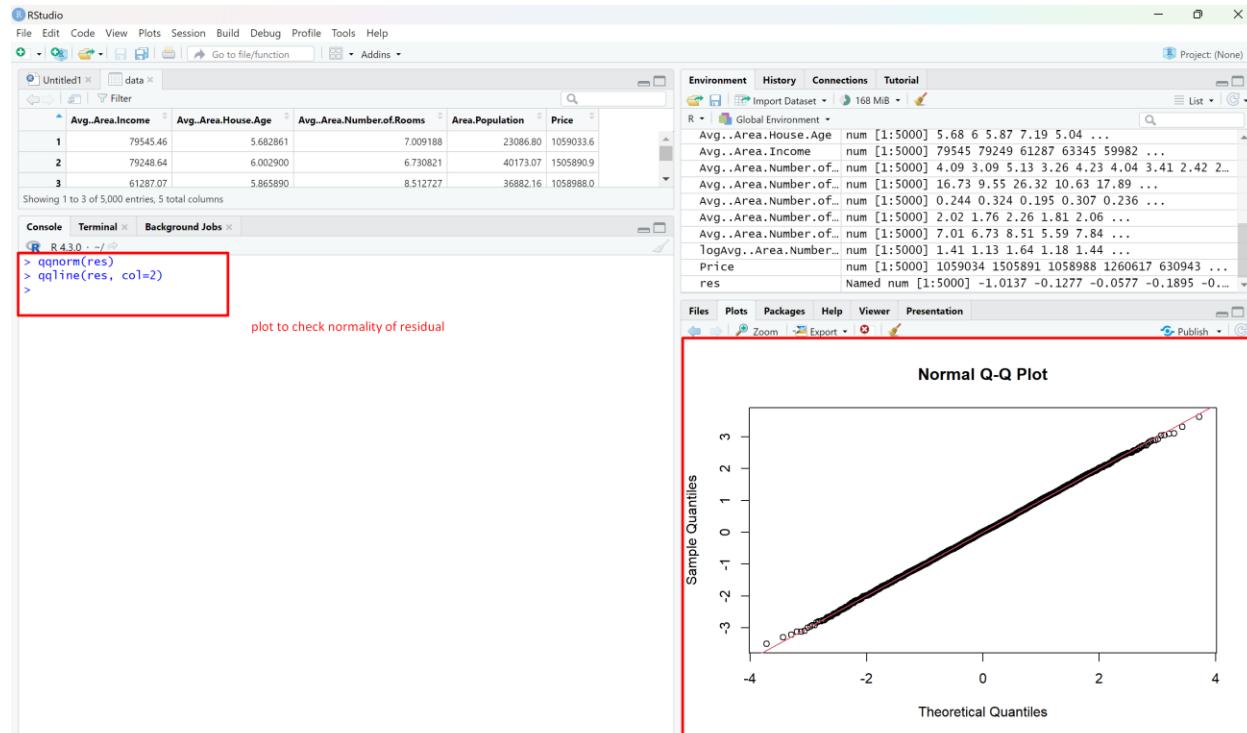
Re-perform residual analysis on m4_lm again:



As residual constant variance got much more worse Deleting the Pricelog variable and re-adding the original Price variable to the data and rebuilding the model again with original Price variable:



Qqplot:



8. Checking multicollinearity for the selected model above (M4) by calculating VIF:

VIF for M4:

The screenshot shows the RStudio interface with the following details:

- Console:** Displays the command `vif(m4_res)` and its output, which shows the VIF values for three variables: Avg.Area.Income (1.000269), Avg.Area.House.Age (1.000357), and Area.Population (1.000616). These values are highlighted with a red box.
- Data View:** Shows a small table with three rows of data for these variables.
- Environment View:** Shows the global environment with objects like base, data, full, m3, m4, m4_ftest, m4_lm, m5, na_count, and train.control.
- Values View:** Shows the data frames for Area.Population, Avg.Area.House.Age, Avg.Area.Income, Avg.Area.Number.of.Rooms, Avg.Area.Number.of.Beds, Avg.Area.Number.of.Baths, Avg.Area.Number.of.Balconies, Avg.Area.Number.of.Balconies, LogAvg.Area.Number.of.Beds, Price, and res.

9. Checking for Influential points by cooks.distance

247 influential points were observed and removed from the data and created new data called data_no_influential

The screenshot shows the RStudio interface with the following details:

- Console:** Displays the code used to identify and remove influential points. It includes commands like `n = nrow(data)`, `threshold = 4/n`, `influence_measures_all = data.frame(influence.measures(m4_lm)\$infmat)`, `influential_measures = influence_measures_all[influence_measures_all\$cook.d > threshold,]`, `influential_measures_index = as.numeric(rownames(influential_measures))`, `data_no_influential = data[-influential_measures_index,]`, and `str(data_no_influential)`. The output shows that 4753 obs. of 5 variables were removed, and the resulting data frame has 247 observations. This part is highlighted with a red box.
- Data View:** Shows the original data frame with 5000 rows and 5 columns.
- Environment View:** Shows the global environment with objects like data_no_influential, influence_measure, influential_measur, m1_full, m2_base, m3, m4, m4_ftest, m4_lm, m5, na_count, and train.control.
- Plots View:** Displays a diagnostic plot titled "Diagnostic Plots" showing the distribution of Cook's distance across 5000 indices. The x-axis is labeled "Index" and ranges from 0 to 5000. The y-axis is labeled "Cook's d" and ranges from 0.0 to 0.6. A horizontal line at approximately 0.1 indicates the threshold.

Rebuild the m4 again with the name “final model” with the new data (data_no_influential_points):

```
R> final_model_m6 = lm(Price ~ Avg..Area.Income + Avg..Area.House.Age + Area.Population, data = data_no_influential)
> summary(final_model_m6)

Call:
lm(formula = Price ~ Avg..Area.Income + Avg..Area.House.Age +
   Area.Population, data = data_no_influential)

Residuals:
    Min      1Q  Median      3Q     Max 
-438726 -101047    790 100924 448444 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.782e+06  2.057e+04 -86.63 <2e-16 ***
Avg..Area.Income 2.139e+01  1.991e-01 107.47 <2e-16 ***
Avg..Area.House.Age 1.673e+05 2.142e+03  78.13 <2e-16 ***
Area.Population 1.513e+01  2.139e-01  70.75 <2e-16 ***

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 142900 on 4749 degrees of freedom
Multiple R-squared:  0.8236, Adjusted R-squared:  0.8234 
F-statistic: 7389 on 3 and 4749 DF,  p-value: < 2.2e-16
```

RMSE for the final model M6:

```
R> y_pred = predict.glm(final_model_m6,data_no_influential)
> y_obs = data_no_influential[,5]
> rmse_final_model_m6 = sqrt((y_obs - y_pred)^%^(y_obs-y_pred) / nrow(data_no_influential))
> rmse_final_model_m6
[1,] 142848.8
```

figure margins too large

