MOBILE PHONE PRICE CLASSIFICATION

TATIKSHA SINGH
BANA 200

GOAL

Determine what variables contribute the most to price range for mobile phones so that Bob can leverage this information for his new mobile phone company in order to compete with existing companies like Apple and Samsung

DEPENDENT VARIABLE

Price range

INDEPENDENT VARIABLES

- Battery Power
- Bluetooth (yes/no)
- Front Camera Megapixels
- Primary Camera Megapixels
- Clock speed
- Dual Sim Card
- 4G (yes/no)
- 3G (yes/no)
- Internal Memory (GB)
- Mobile Weight
- No of cores of processor
- Touch Screen (yes/no)
- Wi-Fi (yes/no)
- Talk Time

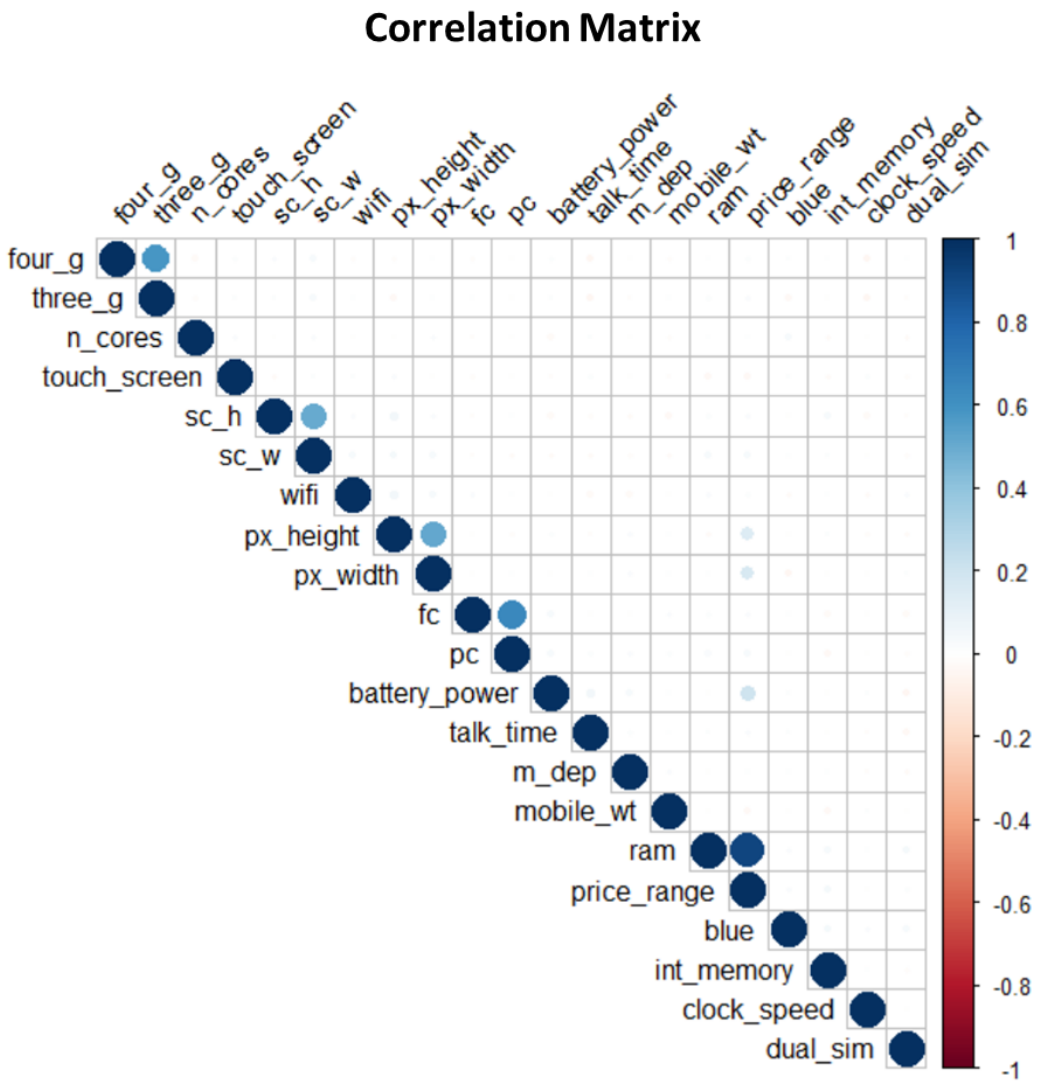
New Independent Variables

- Aspect Ratio
- Screen Diagonal (mm)
- Screen Diagonal (px)
- Pixels Per Inch (PPI)

DESCRIPTIVE STATISTICS

```
battery_power         blue          clock_speed       dual_sim           fc
 Min.   : 501.0   Min.   :0.000   Min.   :0.500   Min.   :0.0000   Min.   : 0.000
 1st Qu.: 851.8   1st Qu.:0.000   1st Qu.:0.700   1st Qu.:0.0000   1st Qu.: 1.000
 Median :1226.0   Median :0.000   Median :1.500   Median :1.0000   Median : 3.000
 Mean   :1238.5   Mean   :0.495   Mean   :1.522   Mean   :0.5095   Mean   : 4.309
 3rd Qu.:1615.2   3rd Qu.:1.000   3rd Qu.:2.200   3rd Qu.:1.0000   3rd Qu.: 7.000
 Max.   :1998.0   Max.   :1.000   Max.   :3.000   Max.   :1.0000   Max.   :19.000
     four_g         int_memory        m_dep          mobile_wt        n_cores
 Min.   :0.0000   Min.   : 2.00   Min.   :0.1000   Min.   : 80.0   Min.   :1.000
 1st Qu.:0.0000   1st Qu.:16.00   1st Qu.:0.2000   1st Qu.:109.0   1st Qu.:3.000
 Median :1.0000   Median :32.00   Median :0.5000   Median :141.0   Median :4.000
 Mean   :0.5215   Mean   :32.05   Mean   :0.5018   Mean   :140.2   Mean   :4.521
 3rd Qu.:1.0000   3rd Qu.:48.00   3rd Qu.:0.8000   3rd Qu.:170.0   3rd Qu.:7.000
 Max.   :1.0000   Max.   :64.00   Max.   :1.0000   Max.   :200.0   Max.   :8.000
       pc           px_height         px_width           ram           sc_h
 Min.   : 0.000   Min.   :   0.0   Min.   : 500.0   Min.   : 256   Min.   : 5.00
 1st Qu.: 5.000   1st Qu.: 282.8   1st Qu.: 874.8   1st Qu.:1208   1st Qu.: 9.00
 Median :10.000   Median : 564.0   Median :1247.0   Median :2146   Median :12.00
 Mean   : 9.916   Mean   : 645.1   Mean   :1251.5   Mean   :2124   Mean   :12.31
 3rd Qu.:15.000   3rd Qu.: 947.2   3rd Qu.:1633.0   3rd Qu.:3064   3rd Qu.:16.00
 Max.   :20.000   Max.   :1960.0   Max.   :1998.0   Max.   :3998   Max.   :19.00
      sc_w          talk_time        three_g         touch_screen       wifi
 Min.   : 0.000   Min.   : 2.00   Min.   :0.0000   Min.   :0.000   Min.   :0.000
 1st Qu.: 2.000   1st Qu.: 6.00   1st Qu.:1.0000   1st Qu.:0.000   1st Qu.:0.000
 Median : 5.000   Median :11.00   Median :1.0000   Median :1.000   Median :1.000
 Mean   : 5.767   Mean   :11.01   Mean   :0.7615   Mean   :0.503   Mean   :0.507
 3rd Qu.: 9.000   3rd Qu.:16.00   3rd Qu.:1.0000   3rd Qu.:1.000   3rd Qu.:1.000
 Max.   :18.000   Max.   :20.00   Max.   :1.0000   Max.   :1.000   Max.   :1.000
  price_range
 Min.   :0.00
 1st Qu.:0.75
 Median :1.50
 Mean   :1.50
 3rd Qu.:2.25
 Max.   :3.00
```

CORRELATION MATRIX BETWEEN ALL VARIABLES



Detecting Multicollinearity

```
> vif(M)
battery_power          blue   clock_speed      dual_sim            fc
     1.012356      1.013269      1.010437      1.013217      1.734805
       four_g   int_memory         m_dep     mobile_wt      n_cores
     1.511596      1.016749      1.008915      1.007367      1.013893
            pc     px_height      px_width           ram          sc_h
     1.730049      1.377745      1.375716      1.016798      1.378235
          sc_w     talk_time       three_g  touch_screen          wifi
     1.371423      1.012087      1.508982      1.007630      1.014818
```

Correlation between RAM and Price Range

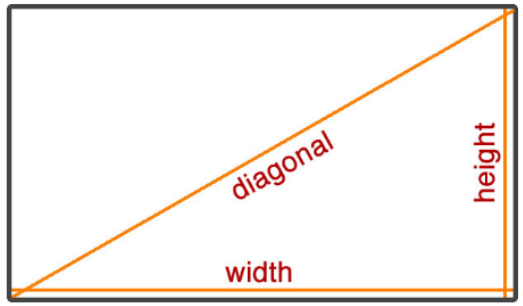
```
        Pearson's product-moment correlation

data:  train$ram and train$price_range
t = 90.422, df = 1498, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.9111024 0.9268110
sample estimates:
      cor
0.9193222
```

INDEPENDENT VARIABLE #1 – SCREEN DIAGONAL IN CM (SC_D)

Using Pythagorean Theorem to calculate the mobile phone's screen width

$$\text{diagonal} = \sqrt{\text{width}^2 + \text{height}^2}$$



train | ppi | sc_d | sc_d_MM | sc_d_inch | stb

Filter

ram	sc_h	sc_w	talk_time	three_g	touch_screen	wifi	price_range	sc_d
2549	9	7	19	0	0	1	1	11
2631	17	3	7	1	1	0	2	17
2603	11	2	9	1	1	0	2	11
2769	16	8	11	1	0	0	2	18
1411	8	2	15	1	1	0	1	8
1067	17	1	10	1	0	0	1	17
3220	13	8	18	1	0	1	3	15
700	16	3	5	1	1	1	0	16
1099	17	1	20	1	0	0	0	17
513	19	10	12	1	0	0	0	21
3946	5	2	7	0	0	0	3	5
3826	14	9	13	1	1	1	3	17
1482	18	0	2	1	0	0	1	18
2680	7	1	4	1	0	1	2	7
373	14	9	3	1	0	1	0	17
568	17	15	11	1	1	1	0	23
3554	10	9	19	1	0	1	3	13
3752	10	2	18	1	1	0	3	10

Environment | History | Connections

To Console | To Source

```
library(tidyr)
diag_calc <- round(sqrt(train$sc_h^2 +
   train$sc_w^2))
view(diag_calc)
sc_d <- train %>%
mutate(sc_d = diag_calc)
View(sc_d)
diag_inch <- diag_cal/2.54
diag_inch <- diag_calc/2.54
```

Files | Plots | Packages | Help | Viewer

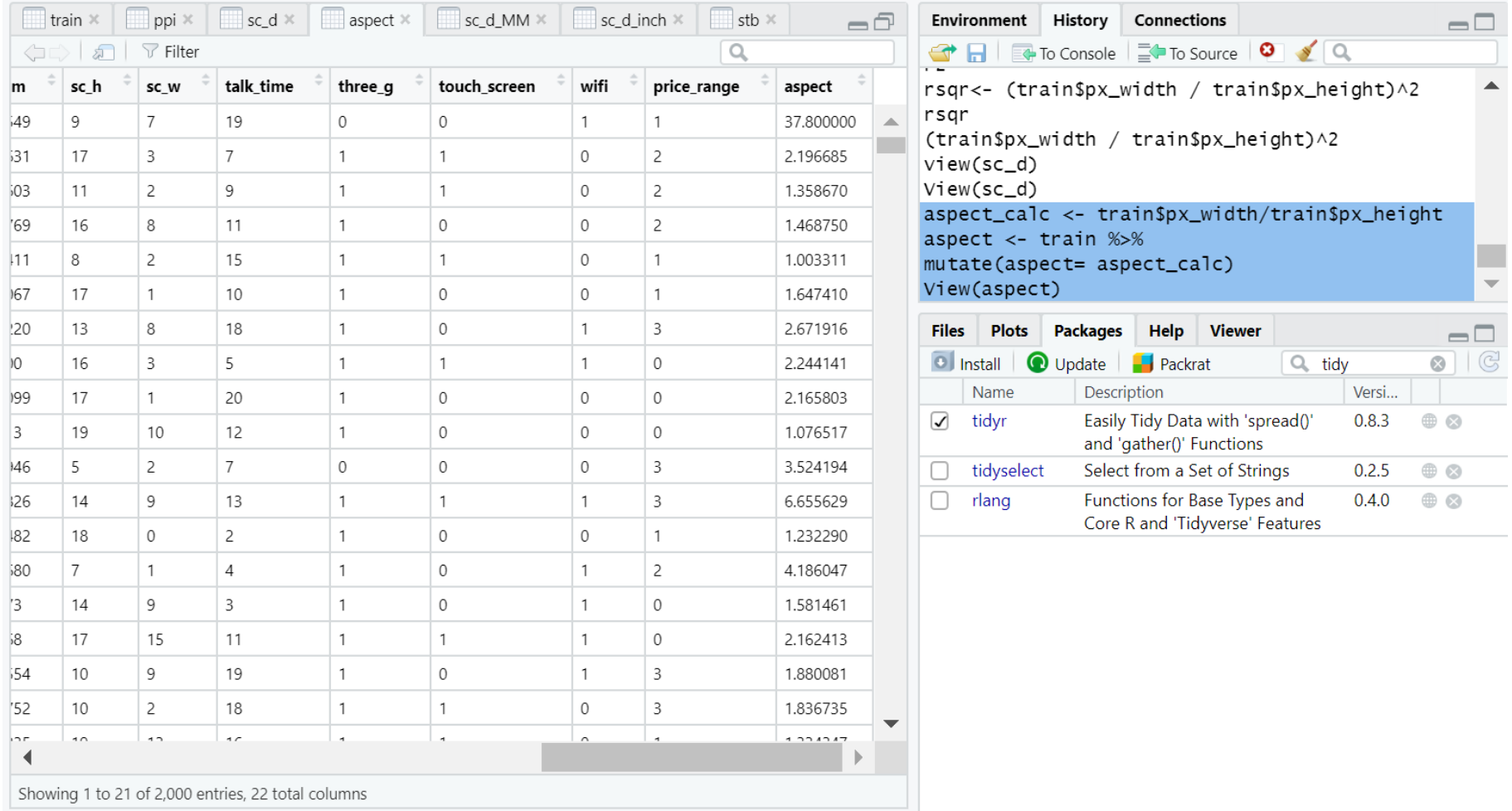
Install | Update | Packrat | tidy

Name	Description	Versi...
tidyr	Easily Tidy Data with 'spread()' and 'gather()' Functions	0.8.3
tidyselect	Select from a Set of Strings	0.2.5
rlang	Functions for Base Types and Core R and 'Tidyverse' Features	0.4.0

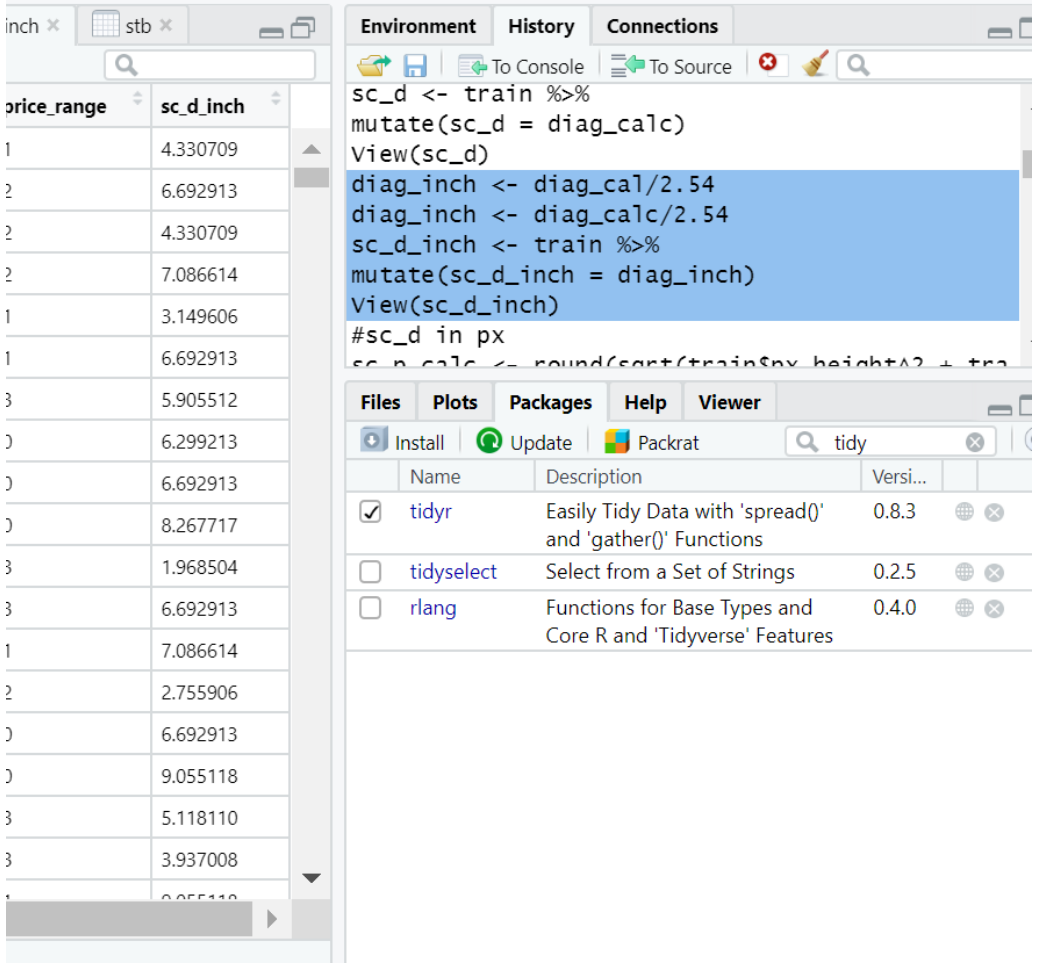
Picture of Pythagoras Theorem Formula and diagram sourced from:
Furey, Edward "Pixels Per Inch PPI Calculator"; CalculatorSoup, *https://www.calculatorsoup.com* - Online Calculators

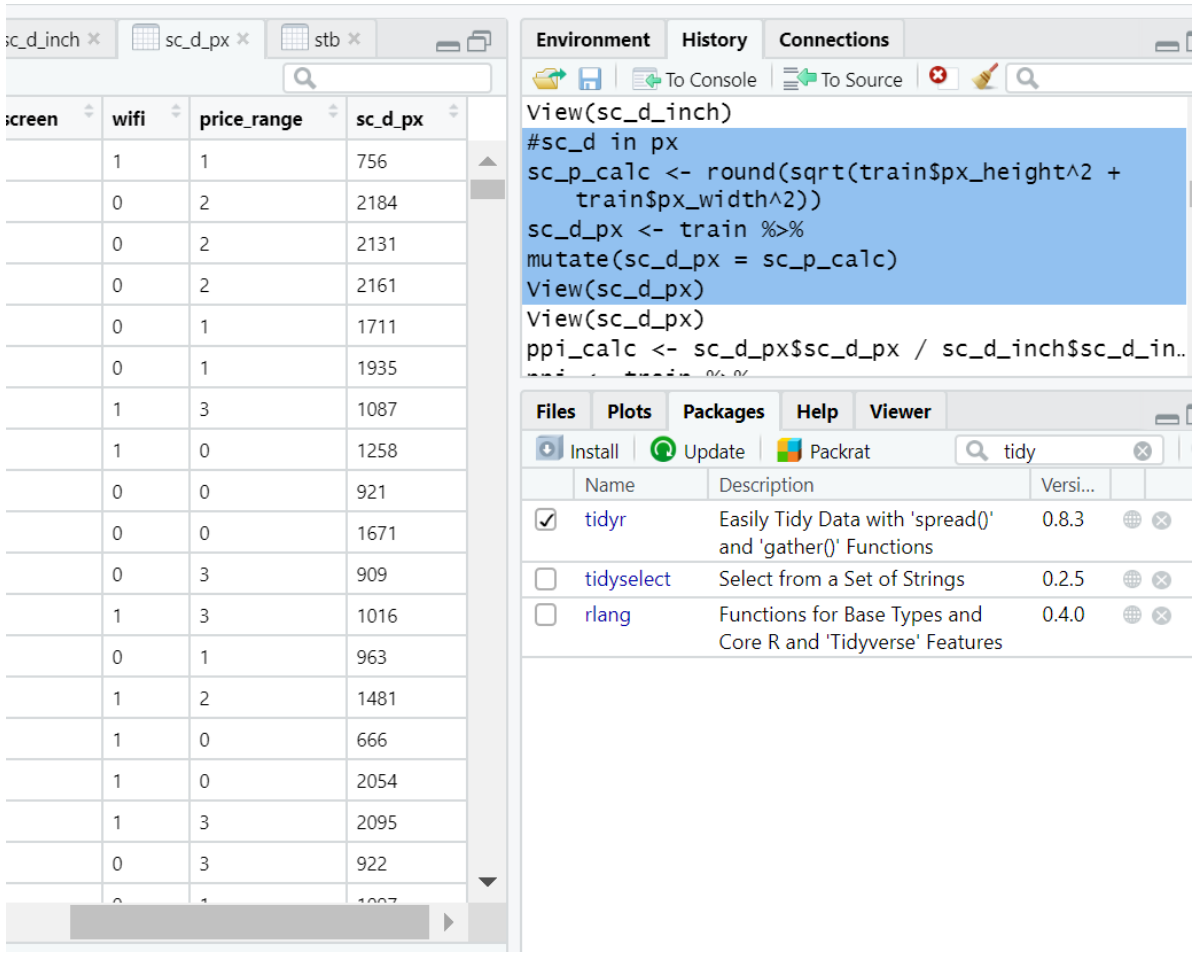
INDEPENDENT VARIABLE #2- ASPECT RATIO

$$\text{Aspect Ratio} = \frac{\text{Width in Pixels}}{\text{Height in Pixels}}$$



INDEPENDENT VARIABLES TO CREATE PIXELS PER INCH (PPI) -
- SCREEN DIAGONAL IN INCH (SC_D_INCH)
- SCREEN DIAGONAL IN PIXELS (SC_D_PX)





INDEPENDENT VARIABLE #3- PIXELS PER INCH (PPI)

$$\text{PPI} = \frac{\text{diagonal in pixels}}{\text{diagonal in inches}}$$

PPI is an indicator of how many pixels are within a 1- inch line for a phone display screen

train | ppi | sc_d_MM | sc_d_inch | stb

Filter

m	sc_h	sc_w	talk_time	three_g	touch_screen	wifi	price_range	ppi
49	9	7	19	0	0	1	1	174.56727
31	17	3	7	1	1	0	2	326.31529
03	11	2	9	1	1	0	2	492.06727
69	16	8	11	1	0	0	2	304.94111
11	8	2	15	1	1	0	1	543.24250
67	17	1	10	1	0	0	1	289.11176
20	13	8	18	1	0	1	3	184.06533
0	16	3	5	1	1	1	0	199.70750
99	17	1	20	1	0	0	0	137.60824
3	19	10	12	1	0	0	0	202.11143
46	5	2	7	0	0	0	3	461.77200
26	14	9	13	1	1	1	3	151.80235
82	18	0	2	1	0	0	1	135.89000
80	7	1	4	1	0	1	2	537.39143
3	14	9	3	1	0	1	0	99.50824
8	17	15	11	1	1	1	0	226.83304
54	10	9	19	1	0	1	3	409.33077
52	10	2	18	1	1	0	3	234.18800

Environment | History | Connections

To Console | To Source

```
View(sc_d_px)
View(sc_d_px)
ppi_calc <- sc_d_px$sc_d_px /
   sc_d_inch$sc_d_inch
ppi <- train %>%
mutate(ppi = ppi_calc)
View(ppi)
ppi_calc <- round(sc_d_px$sc_d_px /
   sc_d_inch$sc_d_inch)
View(ppi)
```

Files | Plots | Packages | Help | Viewer

Install | Update | Packrat | tidy

	Name	Description	Versi...
☑	tidyr	Easily Tidy Data with 'spread()' and 'gather()' Functions	0.8.3
☐	tidyselect	Select from a Set of Strings	0.2.5
☐	rlang	Functions for Base Types and Core R and 'Tidyverse' Features	0.4.0

Use of formula sourced from:
Furey, Edward "Pixels Per Inch PPI Calculator";
CalculatorSoup, *https://www.calculatorsoup.com* - Online Calculators

DESCRIPTIVE STATISTICS FOR NEW VARIABLES

Pixels Per Inch (PPI)

```
      ppi
Min.    :  57.32
1st Qu.: 179.36
Median : 262.91
Mean   : 309.62
3rd Qu.: 380.73
Max.   :1278.13
```

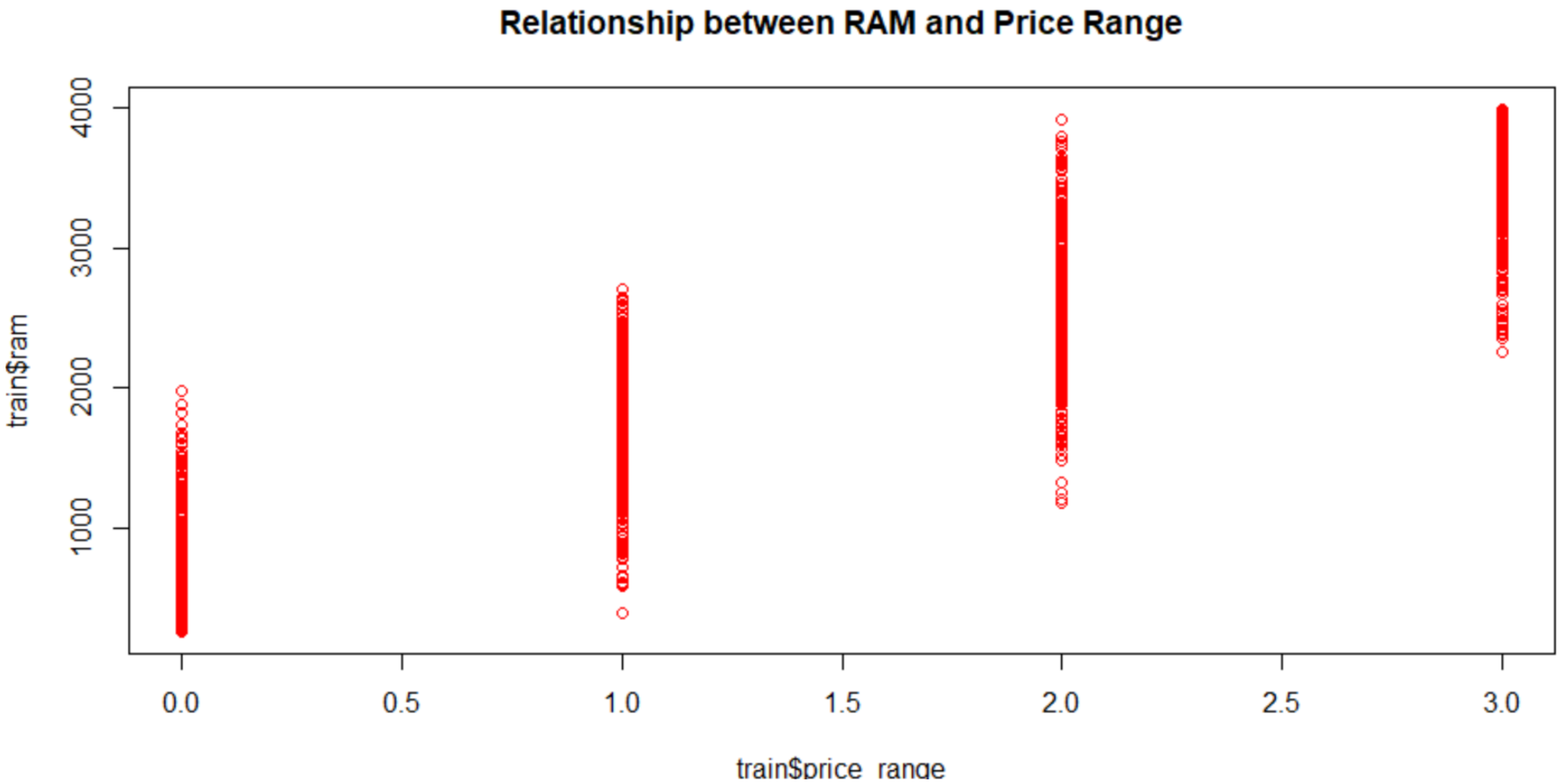
Aspect Ratio

```
     aspect
Min.    :1.001
1st Qu.:1.303
Median :1.919
Mean   :  Inf
3rd Qu.:3.860
Max.   :  Inf
```

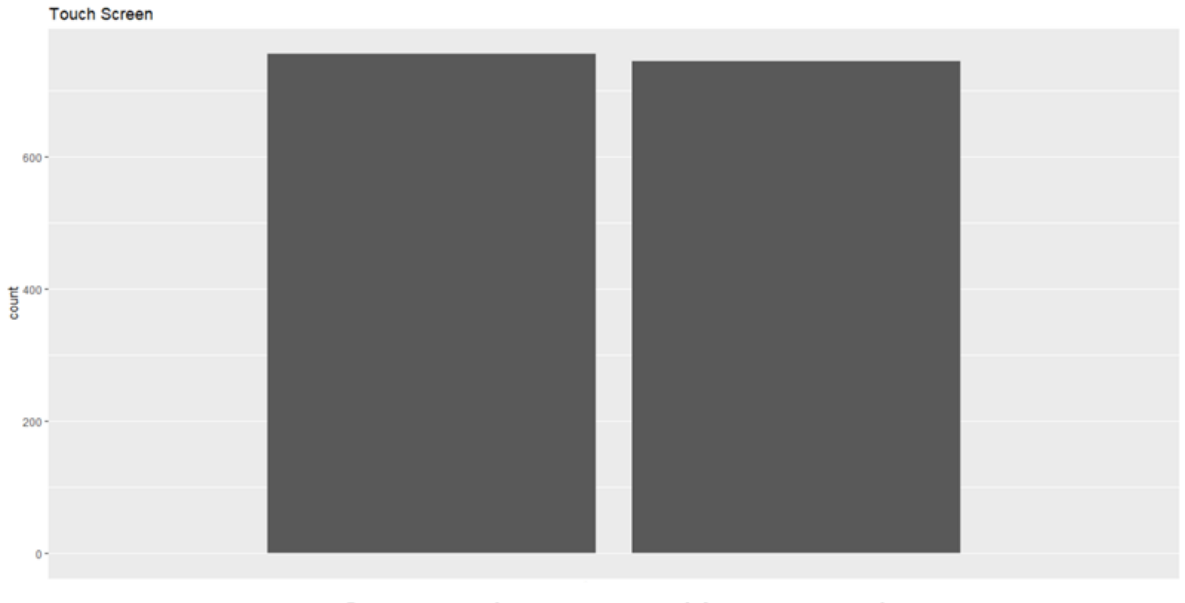
Screen Diagonal (cm)

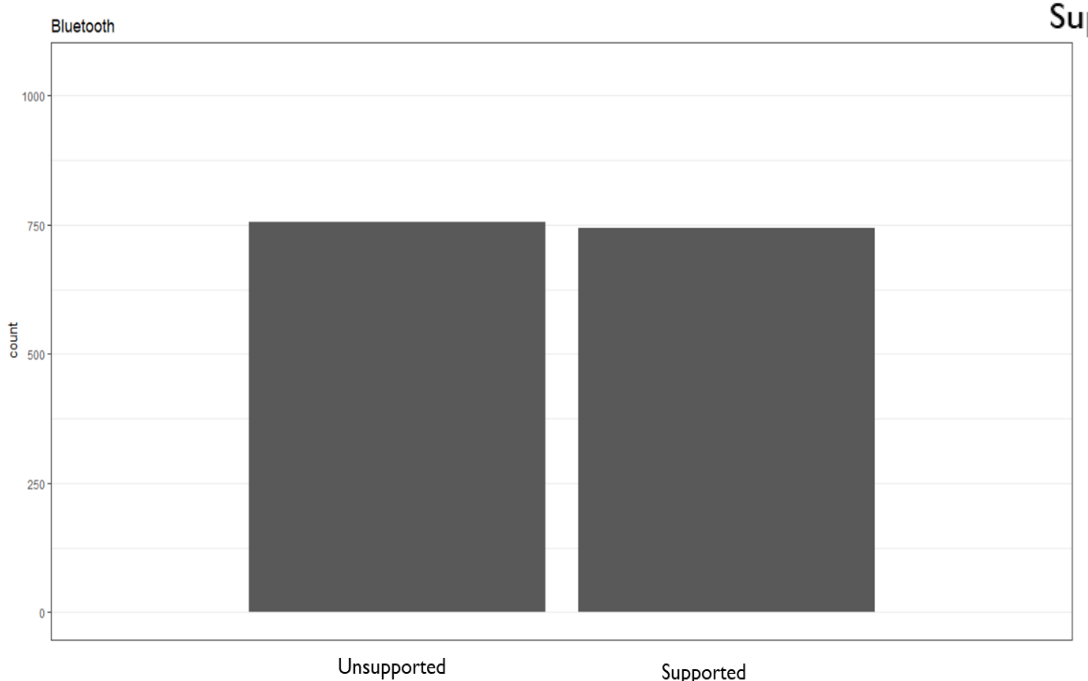
```
      sc_d
Min.    : 5.00
1st Qu.: 9.00
Median :14.00
Mean   :13.93
3rd Qu.:18.00
Max.   :26.00
```

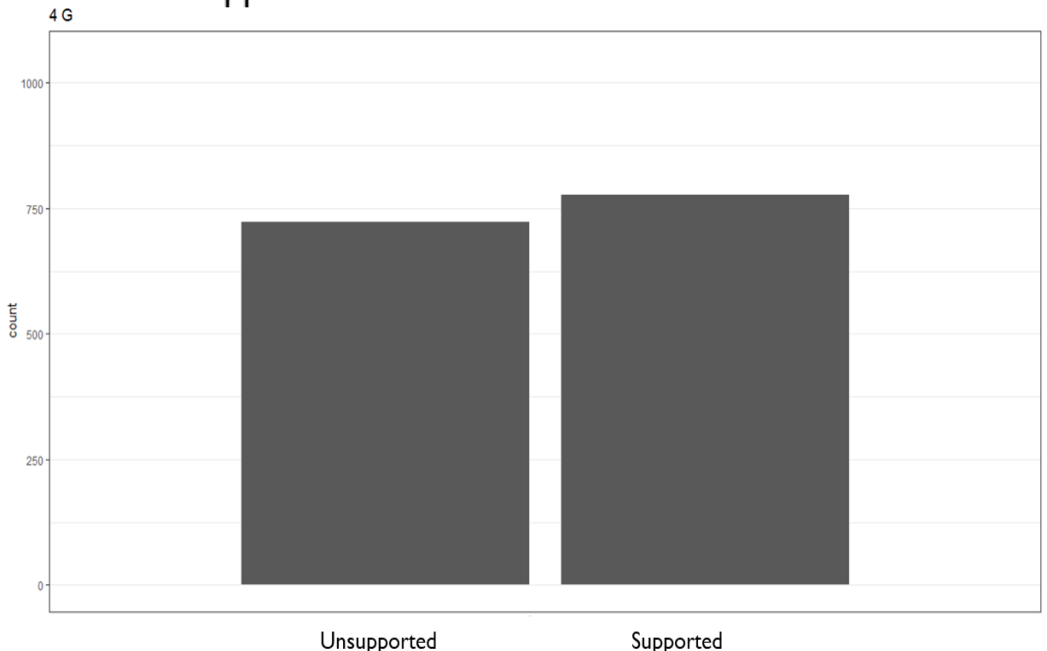
SCATTER PLOT



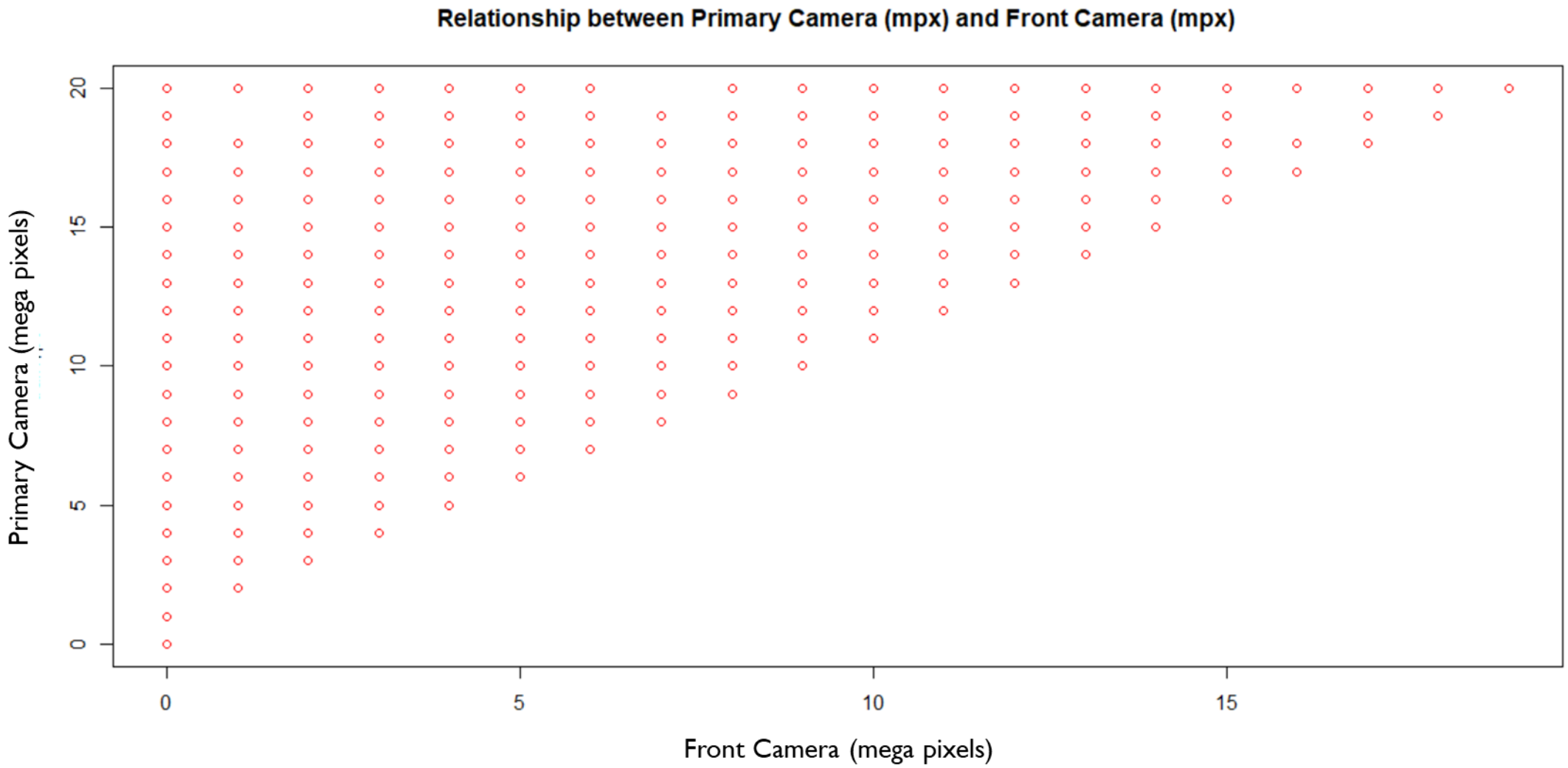
VISUALIZATIONS



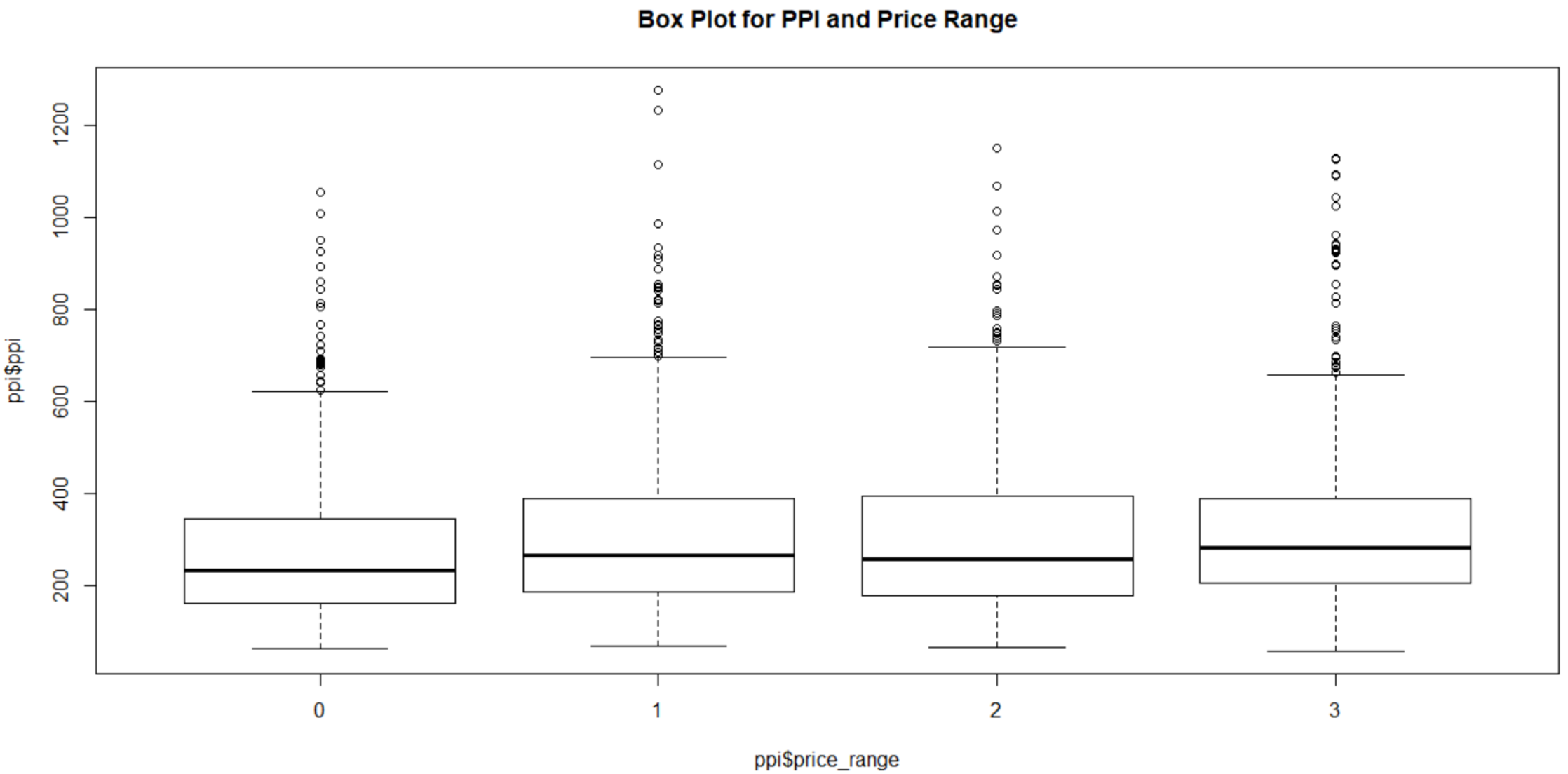




VISUALIZATIONS



VISUALIZATIONS



CLASSIFCATION ANALYSIS

```
Call:
rpart(formula = price_range ~ ., data = df.train1, method = "class",
    parms = list(split = "information"))
  n= 1400

          CP nsplit rel error    xerror       xstd
1 0.33429395      0 1.0000000 1.0441883 0.01497518
2 0.19308357      1 0.6657061 0.6657061 0.01797056
3 0.17098943      2 0.4726225 0.4841499 0.01725260
4 0.01825168      3 0.3016330 0.3218060 0.01533495
5 0.01056676      6 0.2468780 0.2776177 0.01454760
6 0.01000000      7 0.2363112 0.2612872 0.01422081

Variable importance
          ram battery_power    int_memory          sc_w          sc_h      px_width
           79             6             3             3             2             2
        m_dep     mobile_wt     px_height            pc
            2             1             1             1

Node number 1: 1400 observations,    complexity param=0.3342939
  predicted class=3  expected loss=0.7435714  P(node) =1
    class counts:   348   342   351   359
   probabilities: 0.249 0.244 0.251 0.256
  left son=2 (715 obs) right son=3 (685 obs)
  Primary splits:
      ram           < 2217.5 to the left,  improve=661.553900, (0 missing)
      battery_power < 1309   to the left,  improve= 36.655080, (0 missing)
      px_width      < 1296   to the left,  improve= 25.434400, (0 missing)
      px_height     < 642    to the left,  improve= 17.398140, (0 missing)
      sc_w          < 9.5    to the left,  improve=  7.405097, (0 missing)
  Surrogate splits:
      sc_w          < 10.5   to the left,  agree=0.541, adj=0.061, (0 split)
      int_memory    < 42.5   to the left,  agree=0.534, adj=0.048, (0 split)
      battery_power < 1282   to the left,  agree=0.532, adj=0.044, (0 split)
      sc_h          < 12.5   to the left,  agree=0.528, adj=0.035, (0 split)
      m_dep         < 0.25   to the right, agree=0.526, adj=0.031, (0 split)

Node number 2: 715 observations,    complexity param=0.1930836
  predicted class=0  expected loss=0.5132867  P(node) =0.5107143
    class counts:   348   297    70     0
   probabilities: 0.487 0.415 0.098 0.000
  left son=4 (314 obs) right son=5 (401 obs)
  Primary splits:
      ram           < 1106   to the left,  improve=223.349400, (0 missing)
      battery_power < 1438.5 to the left,  improve= 38.492140, (0 missing)
      px_height     < 642    to the left,  improve= 33.045980, (0 missing)
      px_width      < 1081.5 to the left,  improve= 24.333820, (0 missing)
      n_cores       < 4.5    to the left,  improve=  6.549474, (0 missing)
  Surrogate splits:
      mobile_wt   < 91.5   to the left,  agree=0.580, adj=0.045, (0 split)
      px_width    < 591.5  to the left,  agree=0.578, adj=0.038, (0 split)
      int_memory  < 54.5   to the right, agree=0.575, adj=0.032, (0 split)
      pc          < 1.5    to the left,  agree=0.573, adj=0.029, (0 split)
      clock_speed < 2.85   to the right, agree=0.568, adj=0.016, (0 split)

Node number 3: 685 observations,    complexity param=0.1709894
  predicted class=3  expected loss=0.4759124  P(node) =0.4892857
    class counts:     0    45   281   359
   probabilities: 0.000 0.066 0.410 0.524
  left son=6 (311 obs) right son=7 (374 obs)
  Primary splits:
      ram           < 3013.5 to the left,  improve=193.948100, (0 missing)
      battery_power < 1353   to the left,  improve= 44.863200, (0 missing)
      px_width      < 1281   to the left,  improve= 38.586400, (0 missing)
      px_height     < 955    to the left,  improve= 27.844770, (0 missing)
      int_memory    < 11.5   to the left,  improve=  6.232142, (0 missing)
  Surrogate splits:
      battery_power < 579    to the left,  agree=0.562, adj=0.035, (0 split)
      int_memory    < 4.5    to the left,  agree=0.556, adj=0.023, (0 split)
      sc_h          < 18.5   to the right, agree=0.552, adj=0.013, (0 split)
      mobile_wt     < 84.5   to the left,  agree=0.550, adj=0.010, (0 split)
      pc            < 1.5    to the left,  agree=0.549, adj=0.006, (0 split)

Node number 4: 314 observations
  predicted class=0  expected loss=0.09872611  P(node) =0.2242857
    class counts:   283    31     0     0
   probabilities: 0.901 0.099 0.000 0.000

Node number 5: 401 observations,    complexity param=0.01825168
  predicted class=1  expected loss=0.3366584  P(node) =0.2864286
    class counts:    65   266    70     0
   probabilities: 0.162 0.663 0.175 0.000
  left son=10 (171 obs) right son=11 (230 obs)
  Primary splits:
      battery_power < 1108.5 to the left,  improve=51.650560, (0 missing)
      ram           < 1508.5 to the left,  improve=45.468900, (0 missing)
      px_height     < 727    to the left,  improve=30.668250, (0 missing)
      px_width      < 1085.5 to the left,  improve=26.712840, (0 missing)
      n_cores       < 4.5    to the left,  improve= 6.629337, (0 missing)
  Surrogate splits:
      px_height  < 458.5  to the left,  agree=0.606, adj=0.076, (0 split)
      mobile_wt  < 191.5  to the right, agree=0.589, adj=0.035, (0 split)
      int_memory < 3.5    to the left,  agree=0.584, adj=0.023, (0 split)
      m_dep      < 0.85   to the right, agree=0.584, adj=0.023, (0 split)
      px_width   < 746.5  to the left,  agree=0.581, adj=0.018, (0 split)
```

CLASSIFCATION ANALYSIS

```
Node number 6: 311 observations
  predicted class=2  expected loss=0.2861736  P(node) =0.2221429
    class counts:     0    45   222    44
   probabilities: 0.000 0.145 0.714 0.141

Node number 7: 374 observations
  predicted class=3  expected loss=0.157754  P(node) =0.2671429
    class counts:     0     0    59   315
   probabilities: 0.000 0.000 0.158 0.842

Node number 10: 171 observations,    complexity param=0.01825168
  predicted class=1  expected loss=0.374269  P(node) =0.1221429
    class counts:    58   107     6     0
   probabilities: 0.339 0.626 0.035 0.000
  left son=20 (64 obs) right son=21 (107 obs)
  Primary splits:
      ram           < 1541   to the left,  improve=44.664400, (0 missing)
      px_height     < 1158   to the left,  improve=15.847780, (0 missing)
      px_width      < 1481   to the left,  improve=15.011850, (0 missing)
      blue          < 0.5    to the right, improve= 5.448381, (0 missing)
      battery_power < 1007.5 to the left,  improve= 4.145287, (0 missing)
  Surrogate splits:
      sc_h        < 18.5   to the right, agree=0.655, adj=0.078, (0 split)
      m_dep       < 0.15   to the left,  agree=0.649, adj=0.062, (0 split)
      clock_speed < 2.55   to the right, agree=0.643, adj=0.047, (0 split)
      pc          < 19.5   to the right, agree=0.643, adj=0.047, (0 split)
      px_height   < 76     to the left,  agree=0.643, adj=0.047, (0 split)

Node number 11: 230 observations,    complexity param=0.01825168
  predicted class=1  expected loss=0.3086957  P(node) =0.1642857
    class counts:     7   159    64     0
   probabilities: 0.030 0.691 0.278 0.000
  left son=22 (171 obs) right son=23 (59 obs)
  Primary splits:
      ram           < 1896.5 to the left,  improve=33.009440, (0 missing)
      px_width      < 1110   to the left,  improve=23.652520, (0 missing)
      px_height     < 698.5  to the left,  improve=19.398550, (0 missing)
      battery_power < 1485.5 to the left,  improve=12.508940, (0 missing)
      n_cores       < 4.5    to the left,  improve= 6.385996, (0 missing)
  Surrogate splits:
      battery_power < 1119.5 to the right, agree=0.748, adj=0.017, (0 split)
      clock_speed   < 2.85   to the left,  agree=0.748, adj=0.017, (0 split)

Node number 20: 64 observations
  predicted class=0  expected loss=0.234375  P(node) =0.04571429
    class counts:    49    15     0     0
   probabilities: 0.766 0.234 0.000 0.000

Node number 21: 107 observations
  predicted class=1  expected loss=0.1401869  P(node) =0.07642857
    class counts:     9    92     6     0
   probabilities: 0.084 0.860 0.056 0.000

Node number 22: 171 observations
  predicted class=1  expected loss=0.1754386  P(node) =0.1221429
    class counts:     7   141    23     0
   probabilities: 0.041 0.825 0.135 0.000

Node number 23: 59 observations,    complexity param=0.01056676
  predicted class=2  expected loss=0.3050847  P(node) =0.04214286
    class counts:     0    18    41     0
   probabilities: 0.000 0.305 0.695 0.000
  left son=46 (21 obs) right son=47 (38 obs)
  Primary splits:
      px_width      < 1061   to the left,  improve=16.929900, (0 missing)
      battery_power < 1511.5 to the left,  improve=11.423470, (0 missing)
      px_height     < 695    to the left,  improve= 9.319454, (0 missing)
      n_cores       < 2.5    to the left,  improve= 3.904247, (0 missing)
      mobile_wt     < 104.5  to the right, improve= 2.749839, (0 missing)
  Surrogate splits:
      px_height     < 695    to the left,  agree=0.831, adj=0.524, (0 split)
      mobile_wt     < 172    to the right, agree=0.729, adj=0.238, (0 split)
      n_cores       < 2.5    to the left,  agree=0.729, adj=0.238, (0 split)
      battery_power < 1490   to the left,  agree=0.712, adj=0.190, (0 split)
      ram           < 2193.5 to the right, agree=0.712, adj=0.190, (0 split)

Node number 46: 21 observations
  predicted class=1  expected loss=0.2380952  P(node) =0.015
    class counts:     0    16     5     0
   probabilities: 0.000 0.762 0.238 0.000

Node number 47: 38 observations
  predicted class=2  expected loss=0.05263158  P(node) =0.02714286
    class counts:     0     2    36     0
   probabilities: 0.000 0.053 0.947 0.000
```

CLASSIFCATION ANALYSIS

```
> dtree$cptable
          CP nsplit rel error    xerror       xstd
1 0.33429395      0 1.0000000 1.0441883 0.01497518
2 0.19308357      1 0.6657061 0.6657061 0.01797056
3 0.17098943      2 0.4726225 0.4841499 0.01725260
4 0.01825168      3 0.3016330 0.3218060 0.01533495
5 0.01056676      6 0.2468780 0.2776177 0.01454760
6 0.01000000      7 0.2363112 0.2612872 0.01422081
```

```
> names(dt_prac)
 [1] "frame"               "where"               "call"
 [4] "terms"               "cptable"             "method"
 [7] "parms"               "control"             "functions"
[10] "numresp"             "splits"              "variable.importance"
[13] "y"                   "ordered"
```

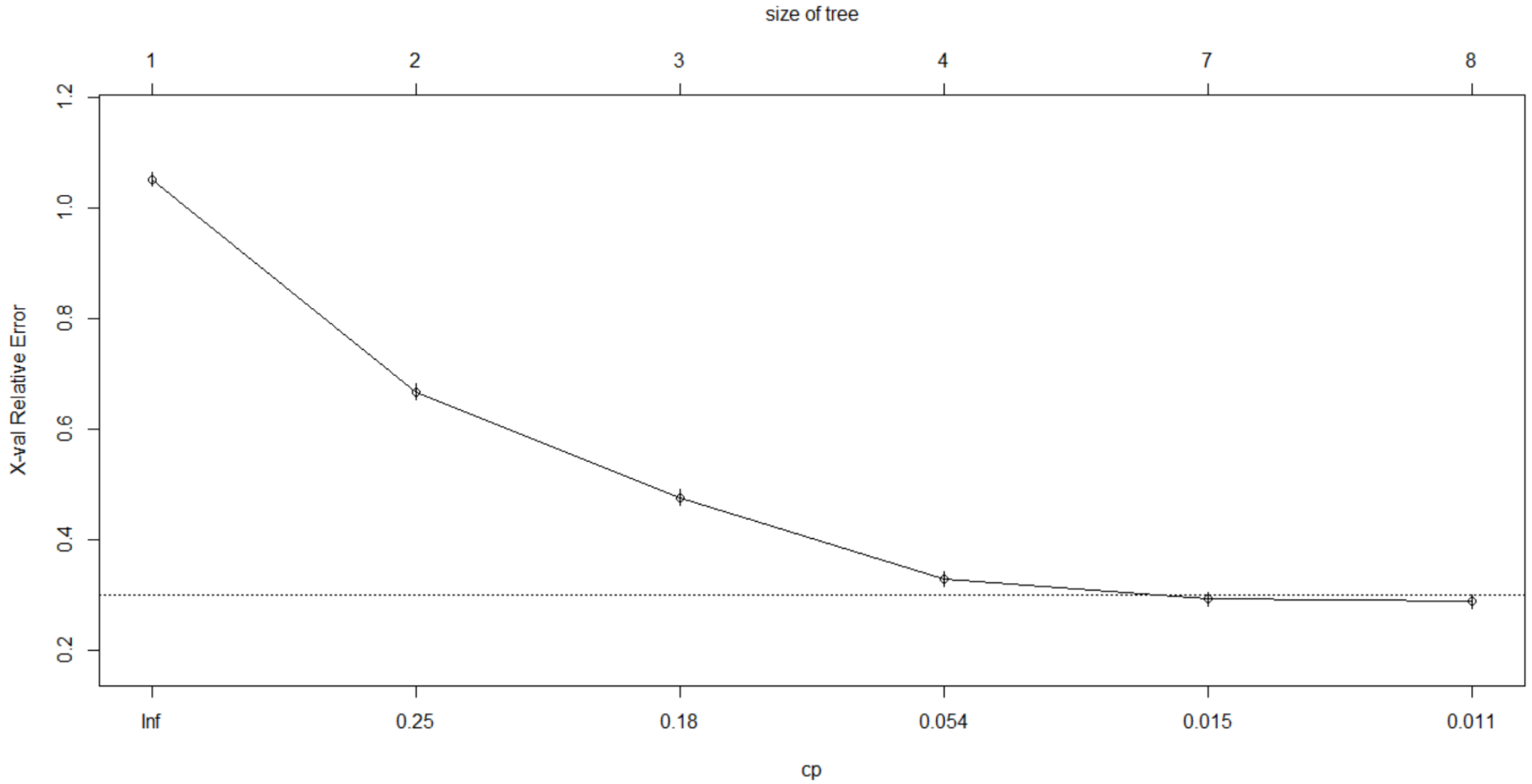
Confusion Matrix

Actual \ Predicted	0	1	2	3
0	145	7	0	0
1	24	113	21	0
2	0	22	98	29
3	0	0	24	117

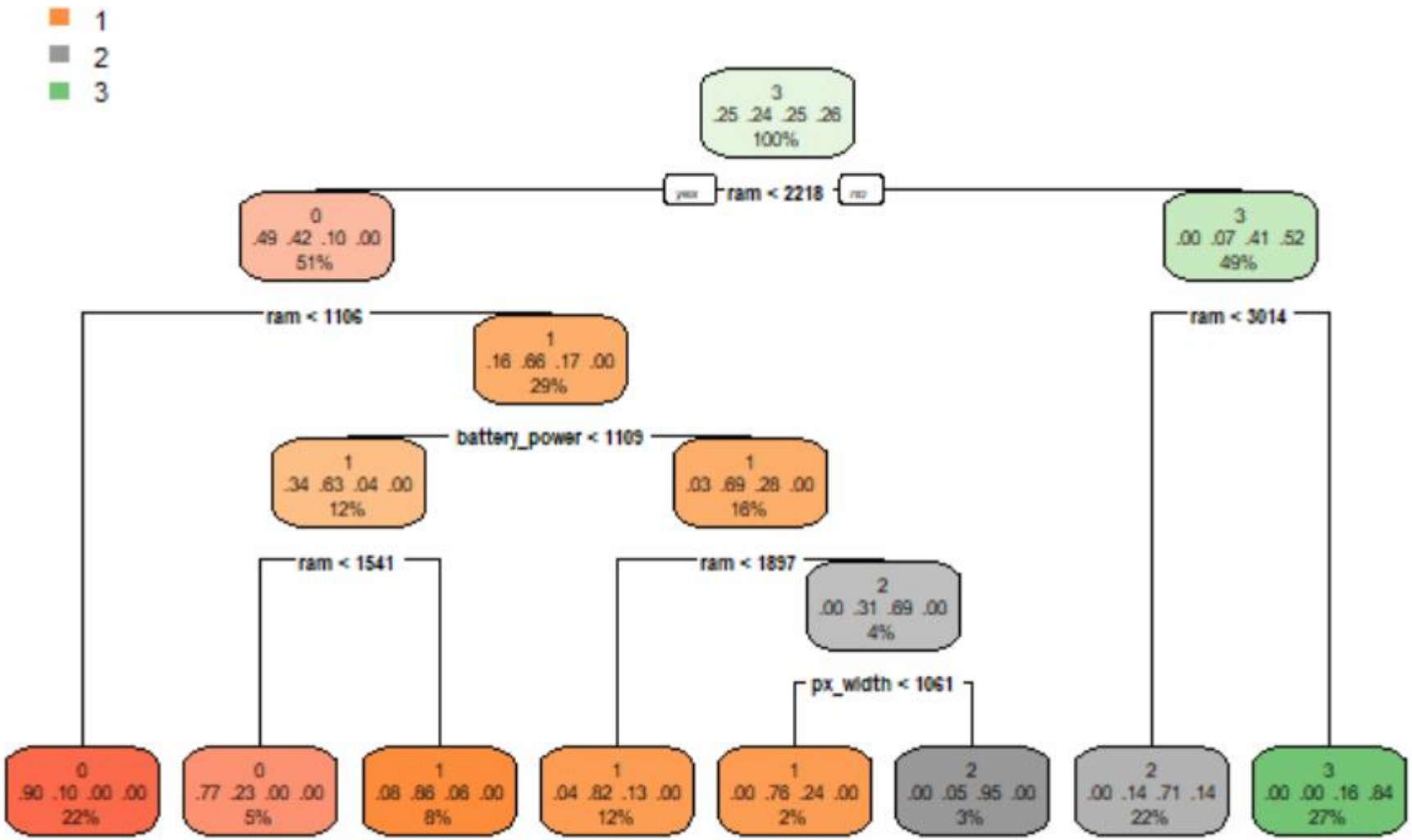
```
> table(df.validate$price_range)

  0   1   2   3
152 158 149 141
```

CLASSIFCATION ANALYSIS



CLASSIFCATION ANALYSIS- DECISION TREE



CLASSIFCATION ANALYSIS- RANDOM FOREST

```
Call:
 randomForest(formula = price_range ~ ., data = df.train1, importance = TRUE,      na.action = na.roughfix)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 6

          Mean of squared residuals: 0.09832107
                    % Var explained: 92.2
```

	IncNodePurity
battery_power	98.243717
blue	2.806492
clock_speed	16.508190
dual_sim	2.689221
fc	13.229535
four_g	2.437942
int_memory	23.138825
m_dep	12.631894
mobile_wt	27.270536
n_cores	11.756247
pc	16.879528
px_height	55.007889
px_width	70.856617
ram	1325.079672
sc_h	16.094625
sc_w	19.502842
talk_time	17.393161
three_g	2.308455
touch_screen	2.541299
wifi	2.293385

KEY INSIGHTS

1. There is some correlation between independent variables (3G/4G, FC/PC, PX_Height/PX_Weight, SC_H/SC_W) which could suggest multicollinearity and affect the analysis, therefore a test for Multicollinearity was conducted using VIF (Variance Inflation Factor)
   a) However, the result shows a factor less than 2 therefore, there are no signs of multicollinearity
2. There is strong, positive correlation (R=0.92) between RAM and Price Range . Therefore, as RAM increases, price and price range increases as well
3. Most mobile phones in this dataset had touch screens and supported 4G which is a necessity in the mobile phone market. Bob must factor this into the production of his mobile phones
4. There is some relationship between the front camera and primary camera. The max megapixels in the front camera is 19 and very few phones had front facing cameras as their primary camera on a mobile phone.
5. The average megapixel for the front camera was only 4.3, however, to be a competitor in the market, Bob should consider making his front camera at least 5 mega pixels
6. PPI for mobile phones relate to the concentration of pixels on the display. As PPI increases, the display quality increases which enhances the user's experience on the mobile phone. Less PPI shows less pixels and that is when images start to look 'pixelated' and have low quality.
   a) The box plot for each price range shows increasing PPI for higher price ranges, however, there are also a lot of outliers. The dataset was not fully accurate to determine the true PPI using height and width in pixels and inches, therefore, the results for PPI should not be fully exercised.
7. Through Classification analysis, RAM was found to be the variable with the most importance of 79 following that was Battery power which was 6.
8. From the Confusion Matrix, 0 class for price range had the least amount of FP and FN, while 2 class for price range had the highest amount of FP and FN
9. RAM that is more than 3014 has to be priced high (price range 3) and RAM that is less than 1106 has to be priced low (price range 0) based on the decision tree

False Positives:
0 → 24
1 → 7 + 22 = 29
2 → 21 + 24 = 45
3→ 29

False Negatives:
0 → 7
1 → 24 + 21 = 45
2 → 22 + 29 = 51
3 → 24

APPENDIX

Definitions

battery_power → Total energy a battery can store in one time measured in mAh
Blue → Has bluetooth or not
clock_speed → speed at which microprocessor executes instructions
dual_sim → Has dual sim support or not
Fc → Front Camera mega pixels
four_g → Has 4G or not
int_memory → Internal Memory in Gigabytes
m_dep → Mobile Depth in cm
mobile_wt → Weight of mobile phone
n_cores → Number of cores of processor
Pc → Primary Camera mega pixels
px_heightPixel Resolution Height
px_width → Pixel Resolution Width
Ram → Random Access Memory in Mega Bytes
sc_h → Screen Height of mobile in cm
sc_w → Screen Width of mobile in cm
talk_time → longest time that a single battery charge will last when you are
three_g → Has 3G or not
touch_screen → Has touch screen or not
Wifi → Has wifi or not
Pixels Per Inch → Measure of pixel density or resolution
Screen Diagonal → Screen Diagonal of mobile in mm and pixels
Aspect Ratio → size of the pixels on the phone screen
Price_range → This is the target variable with value of 0(low cost), 1(medium cost), 2(high cost) and 3(very high cost).

Data Source and Definitions taken from:

Sharma, A. (2018, January 28). Mobile Price Classification. Retrieved from https://www.kaggle.com/iabhishekofficial/mobile-price-classification#train.csv

APPENDIX

```
> library(readxl)
> train <- read_excel("~/Business Analytics/Project 4/train.xlsx")
> View(train)
> names(train)
 [1] "battery_power" "blue"          "clock_speed"   "dual_sim"      "fc"
 [6] "four_g"        "int_memory"    "m_dep"         "mobile_wt"     "n_cores"
[11] "pc"            "px_height"     "px_width"      "ram"           "sc_h"
[16] "sc_w"          "talk_time"     "three_g"       "touch_screen"  "wifi"
[21] "price_range"
> summary(train)
```

```
battery_power        blue          clock_speed      dual_sim          fc
 Min.   : 501.0   Min.   :0.000   Min.   :0.500   Min.   :0.0000   Min.   : 0.000
 1st Qu.: 851.8   1st Qu.:0.000   1st Qu.:0.700   1st Qu.:0.0000   1st Qu.: 1.000
 Median :1226.0   Median :0.000   Median :1.500   Median :1.0000   Median : 3.000
 Mean   :1238.5   Mean   :0.495   Mean   :1.522   Mean   :0.5095   Mean   : 4.309
 3rd Qu.:1615.2   3rd Qu.:1.000   3rd Qu.:2.200   3rd Qu.:1.0000   3rd Qu.: 7.000
 Max.   :1998.0   Max.   :1.000   Max.   :3.000   Max.   :1.0000   Max.   :19.000
     four_g         int_memory        m_dep         mobile_wt        n_cores
 Min.   :0.0000   Min.   : 2.00   Min.   :0.1000   Min.   : 80.0   Min.   :1.000
 1st Qu.:0.0000   1st Qu.:16.00   1st Qu.:0.2000   1st Qu.:109.0   1st Qu.:3.000
 Median :1.0000   Median :32.00   Median :0.5000   Median :141.0   Median :4.000
 Mean   :0.5215   Mean   :32.05   Mean   :0.5018   Mean   :140.2   Mean   :4.521
 3rd Qu.:1.0000   3rd Qu.:48.00   3rd Qu.:0.8000   3rd Qu.:170.0   3rd Qu.:7.000
 Max.   :1.0000   Max.   :64.00   Max.   :1.0000   Max.   :200.0   Max.   :8.000
       pc           px_height        px_width          ram            sc_h
 Min.   : 0.000   Min.   :   0.0   Min.   : 500.0   Min.   : 256   Min.   : 5.00
 1st Qu.: 5.000   1st Qu.: 282.8   1st Qu.: 874.8   1st Qu.:1208   1st Qu.: 9.00
 Median :10.000   Median : 564.0   Median :1247.0   Median :2146   Median :12.00
 Mean   : 9.916   Mean   : 645.1   Mean   :1251.5   Mean   :2124   Mean   :12.31
 3rd Qu.:15.000   3rd Qu.: 947.2   3rd Qu.:1633.0   3rd Qu.:3064   3rd Qu.:16.00
 Max.   :20.000   Max.   :1960.0   Max.   :1998.0   Max.   :3998   Max.   :19.00
      sc_w          talk_time        three_g        touch_screen        wifi
 Min.   : 0.000   Min.   : 2.00   Min.   :0.0000   Min.   :0.000   Min.   :0.000
 1st Qu.: 2.000   1st Qu.: 6.00   1st Qu.:1.0000   1st Qu.:0.000   1st Qu.:0.000
 Median : 5.000   Median :11.00   Median :1.0000   Median :1.000   Median :1.000
 Mean   : 5.767   Mean   :11.01   Mean   :0.7615   Mean   :0.503   Mean   :0.507
 3rd Qu.: 9.000   3rd Qu.:16.00   3rd Qu.:1.0000   3rd Qu.:1.000   3rd Qu.:1.000
 Max.   :18.000   Max.   :20.00   Max.   :1.0000   Max.   :1.000   Max.   :1.000
 price_range
Min.   :0.00
1st Qu.:0.75
Median :1.50
Mean   :1.50
3rd Qu.:2.25
Max.   :3.00
```

APPENDIX

```
  # Simple named list:
  list(mean = mean, median = median)

  # Auto named with `tibble::lst()`:
  tibble::lst(mean, median)

  # Using lambdas
  list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
This warning is displayed once per session.
> corrplot(train, method="circle", type="upper", order="hclust",tl.srt=45)
Error in corrplot(train, method = "circle", type = "upper", order = "hclust",  :
  could not find function "corrplot"
> library(corrplot)
corrplot 0.84 loaded
> corrplot(train, method="circle", type="upper", order="hclust",tl.srt=45)
Error in matrix(if (is.null(value)) logical() else value, nrow = nr, dimnames = list(rn,
 :
  length of 'dimnames' [2] not equal to array extent
> round(cor(df), 8)
Error in cor(df) : supply both 'x' and 'y' or a matrix-like 'x'
> corr<- round(cor(df), 8)
Error in cor(df) : supply both 'x' and 'y' or a matrix-like 'x'
> corr<- round(cor(train), 8)
> ggcorrplot(corr)
Error in ggcorrplot(corr) : could not find function "ggcorrplot"
> corrplot(corr, method="circle")
> corrplot(corr, method="circle", type="upper", order="hclust", tl.srt=45, tl.cex= 1)
> g1 <- corrplot(corr, method="circle", type="upper", order="hclust", tl.srt=45, tl.cex=
 1, tl.col="black")
> g1 <- corrplot(corr, method="circle", type="upper", order="hclust", tl.srt=45, tl.cex=
 1, tl.col="black", title= "Correlation Matrix")
>
```

Environment | History | Connections

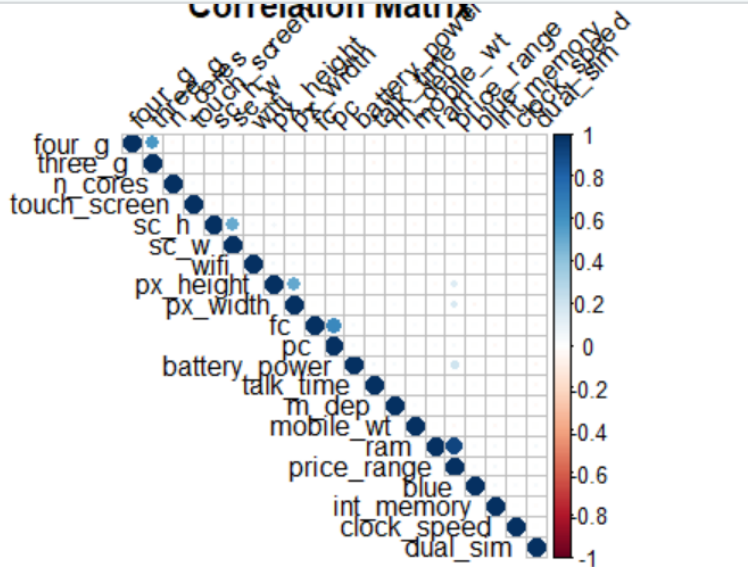
Import Dataset | List

Global Environment

Data	
corr	num [1:21, 1:21] 1 0.0113 0.0115 -0.04…
g1	num [1:21, 1:21] 1 0.5842 -0.0297 0.01…
train	2000 obs. of 21 variables

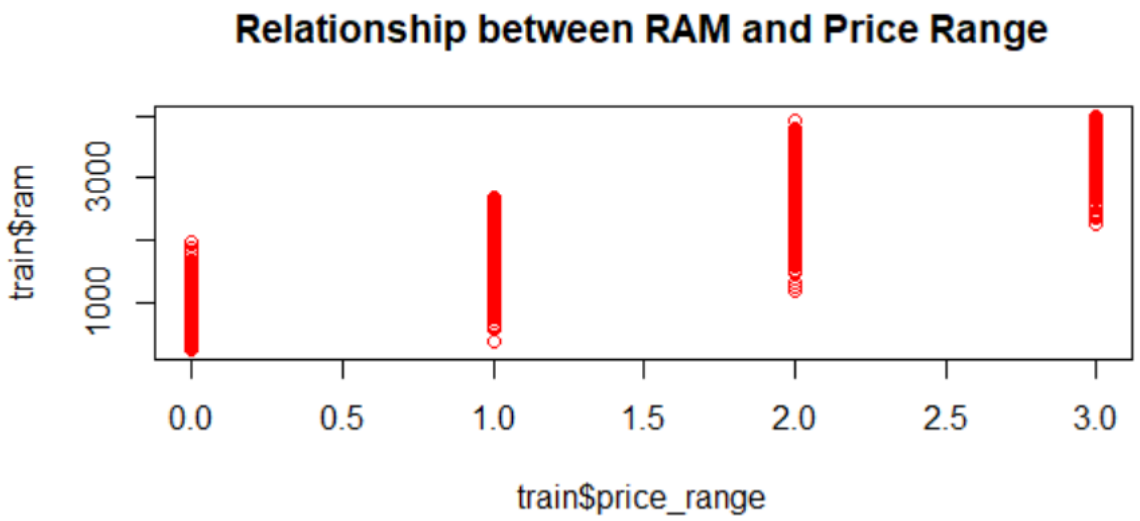
Files | Plots | Packages | Help | Viewer

Zoom | Export



APPENDIX

```
Console   Terminal ×   Jobs ×
~/Business Analytics/Project 4/Project 4/
      pc       px_height        px_width            ram           sc_h
1.730049        1.377745        1.375716        1.016798        1.378235
    sc_w       talk_time         three_g    touch_screen            wifi
1.371423        1.012087        1.508982        1.007630        1.014818
> scatter(train$ram, train$price_range, main="Relationship between RAM and Pr
ice Range", col="red")
Error in scatter(train$ram, train$price_range, main = "Relationship between R
AM and Price Range",  :
  could not find function "scatter"
> plot(train$ram, train$price_range, main="Relationship between RAM and Price
 Range", col="red")
> plot(train$price_range, train$ram, main="Relationship between RAM and Price
 Range", col="red")
```



APPENDIX

```
> View(test)
> library(rpart.plot)
> set.seed(123)
> table(train$class)
< table of extent 0 >
Warning message:
Unknown or uninitialised column: 'class'.
> table(train$price_range)

  0   1   2   3
500 500 500 500
> dt_prac <- rpart(train$price_range ~ ., data= train, method="class")
> View(dt_prac)
> summary(dt_prac)
Call:
rpart(formula = train$price_range ~ ., data = train, method = "class")
  n= 2000

          CP nsplit rel error    xerror       xstd
1 0.33333333      0 1.0000000 1.0520000 0.01216476
2 0.19400000      1 0.6666667 0.6673333 0.01490711
3 0.15800000      2 0.4726667 0.4760000 0.01428444
4 0.01822222      3 0.3146667 0.3293333 0.01285789
5 0.01200000      6 0.2600000 0.2926667 0.01234035
6 0.01000000      7 0.2480000 0.2886667 0.01227926

Variable importance
          ram battery_power       px_width     px_height          sc_w
           85             4              3             3             1
   int_memory     mobile_wt          sc_h
            1             1             1

Node number 1: 2000 observations,    complexity param=0.3333333
```

```
View(aspect)
View(sc_d_px)
library(rpart)
library(readxl)
test <- read_excel("C:/Users/Tatiksha/Desktop/t…
View(test)
library(rpart.plot)
set.seed(123)
table(train$class)
table(train$price_range)
dt_prac <- rpart(train$price_range ~ ., data=
   train, method="class")
View(dt_prac)
summary(dt_prac)
dtree$cptable
dt_prac$cptable
plotcp(dt_prac)
dt.pruned <- prune(dt_prac, cp=0.0125)
train$price_range(dtree$cptable)
train$price_range(dt_prac$cptable)
class(dt_prac$cptable)
names(dt_prac)
prp(dt.pruned, type=2, extra=104,
fallen.leaves=TRUE, main="Decision Tree")
```

Files | Plots | Packages | Help | Viewer

Zoom | Export

figure margins too large

APPENDIX

train | dt_prac | test

Filter

	id	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep
1	1	1043	1	1.8	1	14	0	5	0.1
2	2	841	1	0.5	1	4	1	61	0.8
3	3	1807	1	2.8	0	1	0	27	0.9
4	4	1546	0	0.5	1	18	1	25	0.5
5	5	1434	0	1.4	0	11	1	49	0.5
6	6	1464	1	2.9	1	5	1	50	0.8
7	7	1718	0	2.4	0	1	0	47	1.0
8	8	833	0	2.4	1	0	0	62	0.8

Showing 1 to 10 of 1,000 entries, 21 total columns

Console | Terminal | Jobs

~/Business Analytics/Project 4/Project 4/

```
+     fallen.leaves=TRUE, main="Decision Tree")
> rpart.plot(dt.pruned, extra = 106)
There were 28 warnings (use warnings() to see them)
> rpart.plot(dt.pruned, extra = 106, tweak=1.6)
Warning message:
extra=106 but the response has 4 levels (only the 2nd level is displayed)
> rpart.plot(dt.pruned, extra = 106, tweak=1.3)
Warning message:
extra=106 but the response has 4 levels (only the 2nd level is displayed)
> rpart.plot(dt.pruned, extra = 106, tweak=1)
Warning message:
extra=106 but the response has 4 levels (only the 2nd level is displayed)
>
```

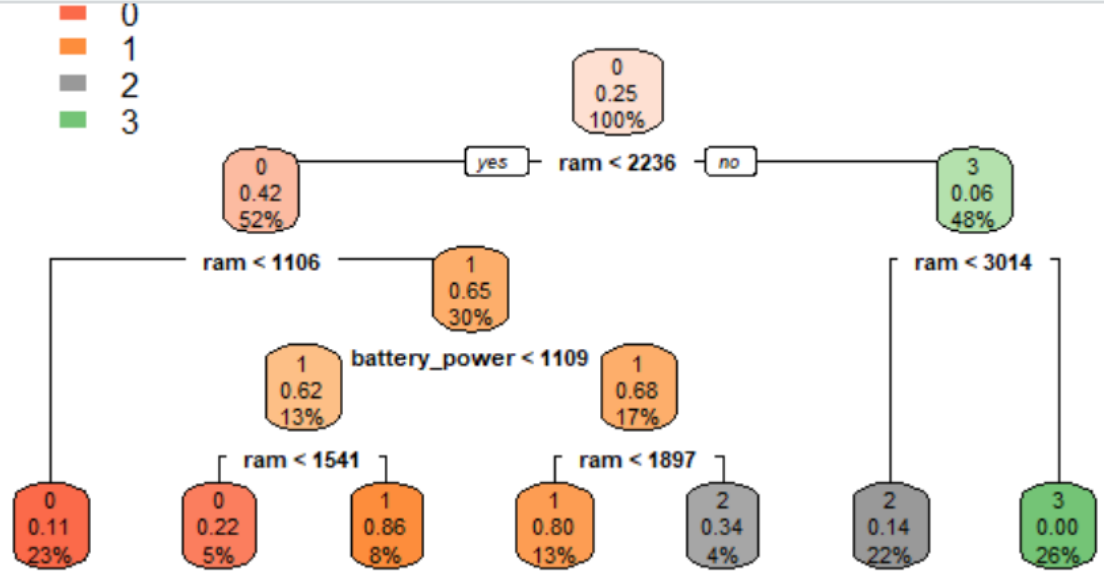
Environment | History | Connections

To Console | To Source

```
dt.pruned <- prune(dt_prac, cp=0.0125)
train$price_range(dtree$cptable)
train$price_range(dt_prac$cptable)
class(dt_prac$cptable)
names(dt_prac)
prp(dt.pruned, type=2, extra=104,
fallen.leaves=TRUE, main="Decision Tree")
rpart.plot(dt.pruned, extra = 106)
rpart.plot(dt.pruned, extra = 106, tweak=1.6)
rpart.plot(dt.pruned, extra = 106, tweak=1.3)
rpart.plot(dt.pruned, extra = 106, tweak=1)
```

Files | Plots | Packages | Help | Viewer

Zoom | Export | Publish



APPENDIX

```
> dtree.perf <- table(df.validate$price_range, dtree.pred, dnn=c("Actua
l", "Predicted"))
> dtree.perf
      Predicted
Actual   0   1   2   3
     0 145   7   0   0
     1  24 113  21   0
     2   0  22  98  29
     3   0   0  24 117
```

```
> forest.perf <- table(df.validate$price_range, forest.pred,
+                      dnn=c("Actual", "Predicted"))
> forest.perf
      Predicted
Actual 0.0338999999999999 0.0440333333333333 0.0442666666666666 0.0532666666666666 0.0653666666666666
     0                  1                  1                  1                  1                  1
      Predicted
Actual 0.0674666666666667 0.0709999999999999 0.0758333333333333 0.0760333333333333 0.0795333333333333
     0                  1                  1                  1                  1                  1
      Predicted
Actual 0.0894666666666667 0.0924333333333333 0.0993 0.1007 0.1022 0.106266666666667 0.106866666666667
     0                  1                  1      1      1      1                  1                  1
      Predicted
Actual 0.107733333333333 0.1094 0.111966666666667 0.113666666666667 0.115166666666667 0.116833333333333
     0                  1      1                  1                  1                  1                  1
      Predicted
Actual 0.129133333333333 0.129366666666667 0.131033333333333 0.133466666666667 0.133833333333333
     0                  1                  1                  1                  1                  1
      Predicted
```

APPENDIX

```
Console   Terminal ×   Jobs ×
~/Business Analytics/Project 4/Project 4/
y and RAM", xlab="RAM", ylab="Internal Memory",pch=19)
> p3 <- ggplot(train, aes(x=four_g, fill=four_g)) +
+     theme_bw() +
+     geom_bar() +
+     ylim(0, 1050) +
+     labs(title = "4 G") +
+     scale_x_discrete(labels = c('Not Supported','Supported'))
> p3
> barchart(train$touch_screen,train, title="Touch Screen", col=blue)
Warning message:
In barchart.numeric(train$touch_screen, train, title = "Touch Screen",  :
  explicit 'data' specification ignored
```

APPENDIX

train | aspect | sc_d | dt_prac

Filter

px_height	px_width	ram	sc_h	sc_w	talk_time	three_g	touch_screen	wifi	price
495	574	3838	9	2	7	1	0	1	
282	1358	1614	17	12	3	1	1	1	
1176	1220	2842	16	12	2	1	0	0	
1673	1759	3970	16	8	18	1	0	1	
710	1179	2844	7	5	18	1	1	0	
361	511	3148	18	7	6	1	1	0	
332	970	1507	5	0	4	0	0	1	
874	1264	2479	9	2	15	1	1	0	

Showing 1 to 10 of 1,500 entries, 21 total columns

Environment | History | Connections

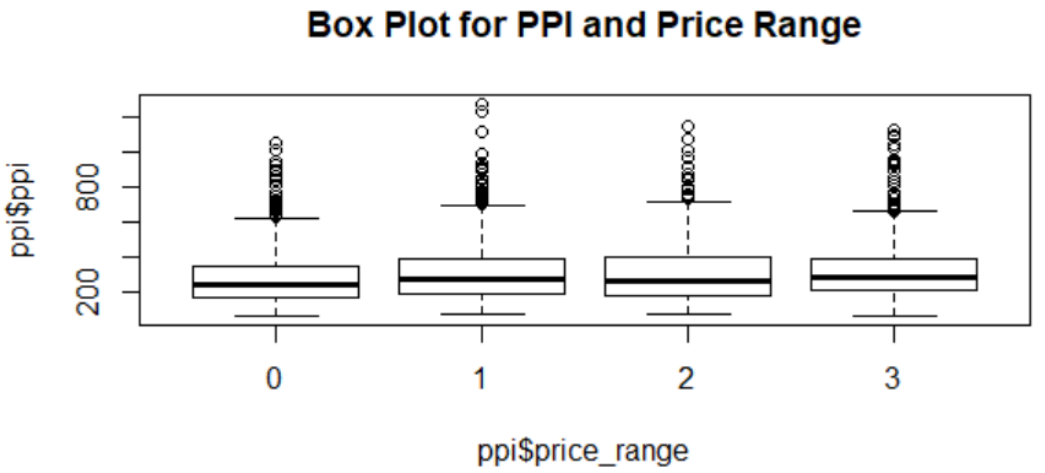
Import Dataset | List

Global Environment

sc_d_px	2000 obs. of 22 variables
stb	2000 obs. of 22 variables
test	374 obs. of 21 variables
train	1500 obs. of 21 variables
x_test	374 obs. of 20 variables
x_train	1500 obs. of 20 variables
Values	
aspect_calc	num [1:2000] 37.8 2.2 1.36 1.47 1 ...

Files | Plots | Packages | Help | Viewer

Zoom | Export | Publish



Console | Terminal | Jobs

~/Business Analytics/Project 4/Project 4/

```
> df2<-subset(ppi, price_range=="1",select=c(22))
> boxplot(df2, main="Box Plot when Price Range is 1", ylab="PPI")
> df2<-subset(ppi, price_range=="2",select=c(22))
> boxplot(df2,main="Box Plot when Price Range is 2", ylab="PPI")
> df2<-subset(ppi, price_range=="3",select=c(22))
> boxplot(df2,main="Box Plot when Price Range is 3", ylab="PPI")
> boxplot(ppi$price_range~ppi$ppi, main="Box Plot for PPI and Price Range")
> boxplot(ppi$ppi~ppi$price_range, main="Box Plot for PPI and Price Range")
> barchart(train$touch_screen, train)
Warning message:
In barchart.numeric(train$touch_screen, train) :
  explicit 'data' specification ignored
>
```

APPENDIX

train | aspect | sc_d | dt_prac

Filter

	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile
1	707	0	2.1	1	2	0	25	0.8	13
2	1986	0	0.5	1	3	1	51	0.1	16
3	947	0	0.8	1	0	1	22	0.1	19
4	1589	1	1.4	0	3	1	26	0.1	10
5	640	0	0.6	0	0	0	20	0.1	16
6	764	1	1.2	1	1	0	13	1.0	15
7	592	0	0.8	1	1	0	52	0.9	10
8	1923	0	1.3	0	3	0	24	1.0	86

Showing 1 to 10 of 1,500 entries, 21 total columns

Console | Terminal | Jobs

~/Business Analytics/Project 4/Project 4/

```
p between Primary Camera (mpx) and Front Camera (mpx"))"
> plot(train$pc, train$fc, col=red, main="Relationship between Primary Camera
 (mpx) and Front Camera (mpx)")
Error in plot.xy(xy, type, ...) : object 'red' not found
> plot(train$pc, train$fc, col="red:, main="Relationship between Primary Came
ra (mpx) and Front Camera (mpx)")
Error: unexpected symbol in "plot(train$pc, train$fc, col="red:, main="Relati
onship"
> plot(train$pc, train$fc, col="red", main="Relationship between Primary Came
ra (mpx) and Front Camera (mpx)")
> plot(train$fc, train$pc, col="red", main="Relationship between Primary Came
ra (mpx) and Front Camera (mpx)")
>
```

Environment | History | Connections

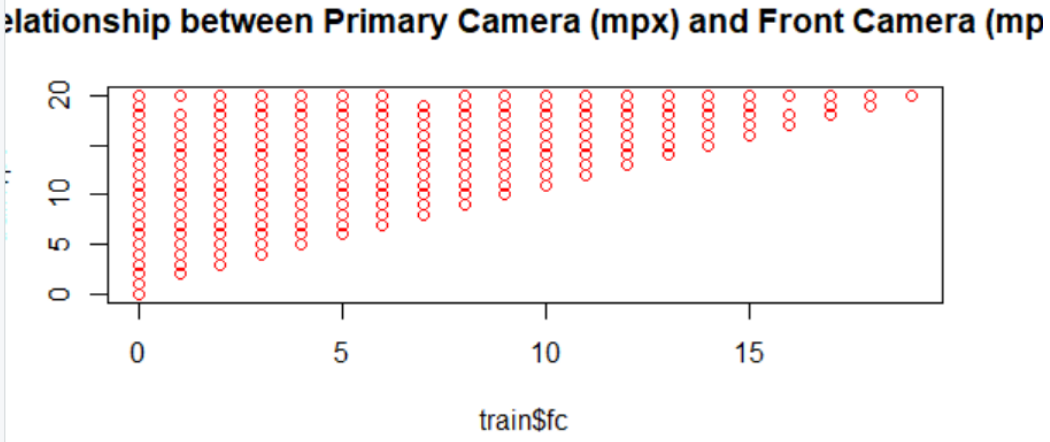
Import Dataset | List

Global Environment

sc_d_px	2000 obs. of 22 variables
stb	2000 obs. of 22 variables
test	374 obs. of 21 variables
train	1500 obs. of 21 variables
x_test	374 obs. of 20 variables
x_train	1500 obs. of 20 variables
Values	
aspect_calc	num [1:2000] 37.8 2.2 1.36 1.47 1 ...

Files | Plots | Packages | Help | Viewer

Zoom | Export | Publish



APPENDIX

```
> M <- lm(train$price_range~.,data=train)
> VIF(mapply(function, ...))
Error: unexpected ',' in "VIF(mapply(function,"
> VIF(M)
Error in VIF(M) : could not find function "VIF"
> vif(M)
battery_power          blue   clock_speed      dual_sim            fc
     1.012356      1.013269      1.010437      1.013217      1.734805
       four_g    int_memory         m_dep     mobile_wt      n_cores
     1.511596      1.016749      1.008915      1.007367      1.013893
            pc     px_height      px_width           ram          sc_h
     1.730049      1.377745      1.375716      1.016798      1.378235
          sc_w     talk_time      three_g  touch_screen          wifi
```