SimpleLinearRegression.R

shashanksoni092

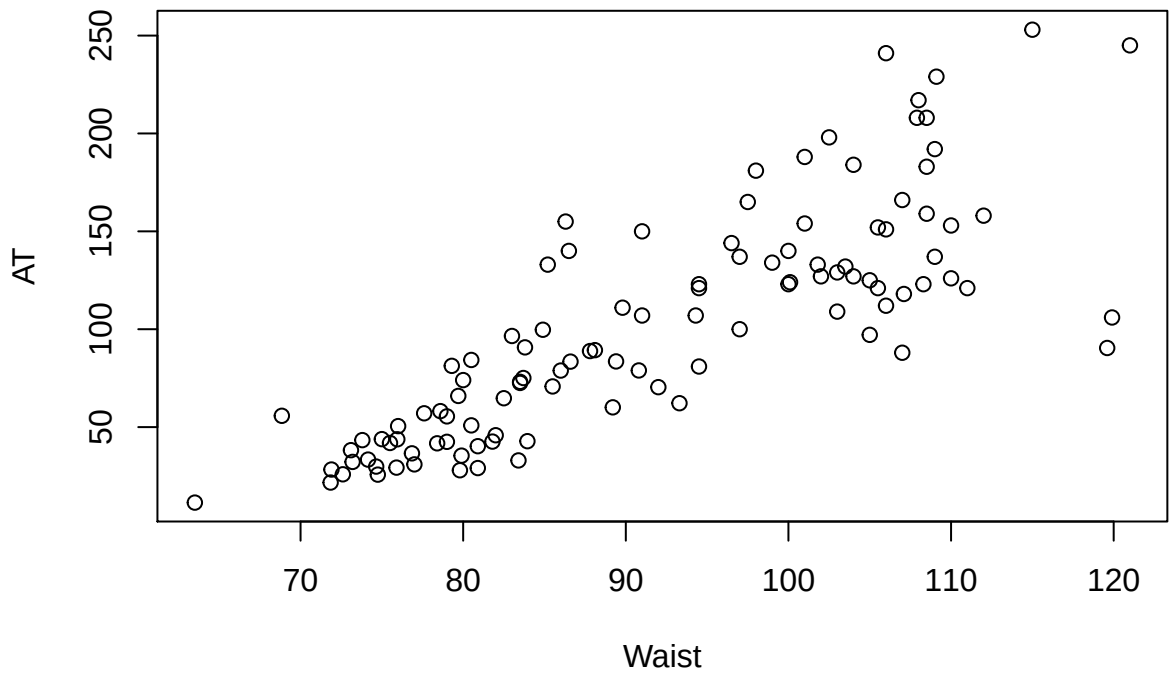
Sat Sep 22 23:13:45 2018

```
#Linear Regression
##Simple Linear Regression
```

```
data<-read.csv("/home/shashanksoni092/DataScienceWorkspace/DataScienceProjectsPdf/LinearRegression/wc-a
dim(data)
```

```
## [1] 109   2
```

```
View(data)
attach(data)
plot(Waist,AT)#To scatter the points ie, a scatterplot
```




```
# Correlation coefficient value for Waist and FAT Data
cor(AT,Waist)
```

```
## [1] 0.8185578
```

```
cor(Waist,AT)
```

```
## [1] 0.8185578
```

```r
class(data)#To know the datatype of data
```

```
## [1] "data.frame"
```

```r
#To know the structure of the data
str(data)
```

```
## 'data.frame':    109 obs. of  2 variables:
##  $ Waist: num  74.8 72.6 81.8 84 74.7 ...
##  $ AT   : num  25.7 25.9 42.6 42.8 29.8 ...
```

```r
#to know the standard deviation of waist
sd(Waist)
```

```
## [1] 13.55912
```

```r
#To get the summary of the data(Business Decision)
summary(data)
```

```
##      Waist             AT        
##  Min.   : 63.5   Min.   : 11.44  
##  1st Qu.: 80.0   1st Qu.: 50.88  
##  Median : 90.8   Median : 96.54  
##  Mean   : 91.9   Mean   :101.89  
##  3rd Qu.:104.0   3rd Qu.:137.00  
##  Max.   :121.0   Max.   :253.00
```

```r
#Time to create a LinearModel for the data

#Syntax for that is lm(y~x)
model1<-lm(AT~Waist)
summary(model1)
```

```
##
## Call:
## lm(formula = AT ~ Waist)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -107.288  -19.143   -2.939   16.376   90.342
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -215.9815    21.7963  -9.909   <2e-16 ***
## Waist          3.4589     0.2347  14.740   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 33.06 on 107 degrees of freedom
## Multiple R-squared:   0.67,  Adjusted R-squared:  0.667
## F-statistic: 217.3 on 1 and 107 DF,  p-value: < 2.2e-16
```

```r
confint(model1,level = 0.95)
```

```
##                   2.5 %     97.5 %
## (Intercept) -259.190053 -172.77292
## Waist          2.993689    3.92403
```

```r
predict(model1)
```

```
##          1          2          3          4          5          6
##  42.568252  35.131704  66.953210  74.389758  42.222366  32.537559
##          7          8          9         10         11         12
##  63.840237  72.487385   3.656083  37.207020  32.710502  43.432966
##         13         14         15         16         17         18
##  36.861134  57.268404  50.350685  22.160981  46.718883  40.492936
##         19         20         21         22         23         24
##  39.282335  46.545940  49.831856  63.840237  60.381377  92.548770
##         25         26         27         28         29         30
##  67.644982 102.233576  83.555735  62.456693  81.480420  69.374412
##         31         32         33         34         35         36
##  72.833271  88.744024  98.082945  93.240542 136.822170 110.880725
##         37         38         39         40         41         42
##  98.774717 140.281029  60.727263  57.268404  72.833271  46.891826
##         43         44         45         46         47         48
##  62.456693  83.209849  71.103842 154.462353 110.188953 110.880725
##         49         50         51         52         53         54
##  59.689606  58.306062  94.624085  73.870929  78.713332  45.162396
##         55         56         57         58         59         60
##  55.193088  55.884860  87.706367  82.518078  79.750990  73.525043
##         61         62         63         64         65         66
##  52.426001  77.675674  60.035492 158.612984 197.698095 198.735753
##         67         68         69         70         71         72
## 117.798443 148.928178 147.198748 154.116467 154.116467 133.363311
##         73         74         75         76         77         78
## 119.527873 129.904451 157.575326 129.904451 140.281029 143.739889
##         79         80         81         82         83         84
## 150.657608 161.034186 142.010459 164.493045 164.493045 171.410764
##         85         86         87         88         89         90
## 159.304756 143.739889 167.951905 159.304756 202.540498 161.034186
##         91         92         93         94         95         96
## 121.257303 148.928178 122.986732 110.880725 119.527873 147.198748
##         97         98         99        100        101        102
## 150.657608 126.445592  98.774717 138.551600 150.657608 161.380072
##        103        104        105        106        107        108
## 181.787342 133.363311 130.250337 106.730093 136.130398 157.229440
##        109
## 159.304756
```

```r
predict(model1,interval="predict")
```

```
## Warning in predict.lm(model1, interval = "predict"): predictions on current data refer to _future_ r
```

```
##           fit         lwr       upr
## 1   42.568252 -23.7607107 108.89721
## 2   35.131704 -31.3249765 101.58838
## 3   66.953210   0.9383962 132.96802
## 4   74.389758   8.4385892 140.34093
## 5   42.222366 -24.1122081 108.55694
## 6   32.537559 -33.9671546  99.04227
## 7   63.840237  -2.2056980 129.88617
## 8   72.487385   6.5213726 138.45340
## 9    3.656083 -63.5036005  70.81577
```

```
## 10   37.207020 -29.2125284 103.62657
## 11   32.710502 -33.7909536  99.21196
## 12   43.432966 -22.8821078 109.74804
## 13   36.861134 -29.5645231 103.28679
## 14   57.268404  -8.8518878 123.38870
## 15   50.350685 -15.8605336 116.56190
## 16   22.160981 -44.5537679  88.87573
## 17   46.718883 -19.5452517 112.98302
## 18   40.492936 -25.8701771 106.85605
## 19   39.282335 -27.1012331 105.66590
## 20   46.545940 -19.7208032 112.81268
## 21   49.831856 -16.3867039 116.05042
## 22   63.840237  -2.2056980 129.88617
## 23   60.381377  -5.7022296 126.46498
## 24   92.548770  26.6894200 158.40812
## 25   67.644982   1.6367253 133.65324
## 26  102.233576  36.3862036 168.08095
## 27   83.555735  17.6622091 149.44926
## 28   62.456693  -3.6039202 128.51731
## 29   81.480420  15.5758571 147.38498
## 30   69.374412   3.3819768 135.36685
## 31   72.833271   6.8700310 138.79651
## 32   88.744024  22.8729233 154.61513
## 33   98.082945  32.2335934 163.93230
## 34   93.240542  27.3829016 159.09818
## 35  136.822170  70.8074775 202.83686
## 36  110.880725  45.0222774 176.73917
## 37   98.774717  32.9260237 164.62341
## 38  140.281029  74.2316072 206.33045
## 39   60.727263  -5.3524301 126.80696
## 40   57.268404  -8.8518878 123.38870
## 41   72.833271   6.8700310 138.79651
## 42   46.891826 -19.3697083 113.15336
## 43   62.456693  -3.6039202 128.51731
## 44   83.209849  17.3145658 149.10513
## 45   71.103842   5.1264122 137.08127
## 46  154.462353  88.2365608 220.68815
## 47  110.188953  44.3321471 176.04576
## 48  110.880725  45.0222774 176.73917
## 49   59.689606  -6.4019262 125.78114
## 50   58.306062  -7.8017094 124.41383
## 51   94.624085  28.7694706 160.47870
## 52   73.870929   7.9158100 139.82605
## 53   78.713332  12.7922191 144.63445
## 54   45.162396 -21.1255054 111.45030
## 55   55.193088 -10.9531208 121.33930
## 56   55.884860 -10.2525800 122.02230
## 57   87.706367  21.8313711 153.58136
## 58   82.518078  16.6191807 148.41697
## 59   79.750990  13.8363291 145.66565
## 60   73.525043   7.5672497 139.48284
## 61   52.426001 -13.7565798 118.60858
## 62   77.675674  11.7478144 143.60353
## 63   60.035492  -6.0520617 126.12304
```

```
## 64  158.612984  92.3252791 224.90069
## 65  197.698095 130.6020356 264.79416
## 66  198.735753 131.6127559 265.85875
## 67  117.798443  51.9163563 183.68053
## 68  148.928178  82.7776990 215.07866
## 69  147.198748  81.0701043 213.32739
## 70  154.116467  87.8956245 220.33731
## 71  154.116467  87.8956245 220.33731
## 72  133.363311  67.3800865 199.34653
## 73  119.527873  53.6378248 185.41792
## 74  129.904451  63.9494297 195.85947
## 75  157.575326  91.3035349 223.84712
## 76  129.904451  63.9494297 195.85947
## 77  140.281029  74.2316072 206.33045
## 78  143.739889  77.6524810 209.82730
## 79  150.657608  84.4844833 216.83073
## 80  161.034186  94.7082219 227.36015
## 81  142.010459  75.9424508 208.07847
## 82  164.493045  98.1096934 230.87640
## 83  164.493045  98.1096934 230.87640
## 84  171.410764 104.9030239 237.91850
## 85  159.304756  93.0062808 225.60323
## 86  143.739889  77.6524810 209.82730
## 87  167.951905 101.5079578 234.39585
## 88  159.304756  93.0062808 225.60323
## 89  202.540498 135.3163441 269.76465
## 90  161.034186  94.7082219 227.36015
## 91  121.257303  55.3584733 187.15613
## 92  148.928178  82.7776990 215.07866
## 93  122.986732  57.0783023 188.89516
## 94  110.880725  45.0222774 176.73917
## 95  119.527873  53.6378248 185.41792
## 96  147.198748  81.0701043 213.32739
## 97  150.657608  84.4844833 216.83073
## 98  126.445592  60.5155029 192.37568
## 99   98.774717  32.9260237 164.62341
## 100 138.551600  72.5199497 204.58325
## 101 150.657608  84.4844833 216.83073
## 102 161.380072  95.0485136 227.71163
## 103 181.787342 115.0691257 248.50556
## 104 133.363311  67.3800865 199.34653
## 105 130.250337  64.2926425 196.20803
## 106 106.730093  40.8795247 172.58066
## 107 136.130398  70.1222603 202.13854
## 108 157.229440  90.9628890 223.49599
## 109 159.304756  93.0062808 225.60323
```

```r
# R-squared value for the above model is 0.667.
# we may have to do transformation of variables for better R-squared value
# Applying transformations

# Logarthmic transformation
reg_log <- lm(AT ~ log(Waist))  # Regression using logarthmic transformation
summary(reg_log)
```

```
##
## Call:
## lm(formula = AT ~ log(Waist))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -98.473 -18.273  -2.374  14.538  90.400
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1328.34      95.92  -13.85   <2e-16 ***
## log(Waist)    317.14      21.26   14.92   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.8 on 107 degrees of freedom
## Multiple R-squared:  0.6753, Adjusted R-squared:  0.6723
## F-statistic: 222.6 on 1 and 107 DF,  p-value: < 2.2e-16
```

```
confint(reg_log,level=0.95)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1518.4980 -1138.1860
## log(Waist)    274.9936   359.2775
```

```
predict(reg_log,interval="predict")
```

```
## Warning in predict.lm(reg_log, interval = "predict"): predictions on current data refer to _future_ r
```

```
##           fit         lwr       upr
## 1    39.82816 -26.0086206 105.66493
## 2    30.57278 -35.4293466  96.57490
## 3    68.41096   2.9414522 133.88047
## 4    76.63877  11.2344683 142.04307
## 5    39.40361 -26.4402572 105.24748
## 6    27.27955 -38.7868379  93.34593
## 7    64.90236  -0.6004886 130.40520
## 8    74.55421   9.1351144 139.97331
## 9   -11.89961 -78.9452004  55.14599
## 10   33.18296 -32.7702511  99.13618
## 11   27.50016 -38.5618285  93.56215
## 12   40.88704 -24.9322576 106.70634
## 13   32.74942 -33.2117928  98.71064
## 14   57.36532  -8.2202874 122.95093
## 15   49.23319 -16.4587235 114.92511
## 16   13.75359 -52.6063320  80.11351
## 17   44.87886 -20.8772091 110.63493
## 18   37.27231 -28.6078692 103.15250
## 19   35.77184 -30.1346300 101.67831
## 20   44.67001 -21.0892604 110.42929
## 21   48.61480 -17.0859270 114.31552
## 22   64.90236  -0.6004886 130.40520
## 23   60.95783  -4.5864306 126.50210
## 24   95.87610  30.5530649 161.19914
## 25   69.18541   3.7228117 134.64800
## 26  105.67798  40.3579039 170.99806
```

```
## 27   86.49485  21.1446565 151.84503
## 28   63.33043  -2.1884217 128.84928
## 29   84.28995  18.9299349 149.64997
## 30   71.11329   5.6672036 136.55937
## 31   74.93424   9.5179301 140.35056
## 32   91.94092  26.6093922 157.27246
## 33  101.51422  36.1960545 166.83238
## 34   96.58637  31.2644225 161.90832
## 35  138.40137  72.9033276 203.89942
## 36  114.18079  48.8422401 179.51933
## 37  102.21199  36.8938286 167.53014
## 38  141.49540  75.9656230 207.02518
## 39   61.35450  -4.1854096 126.89441
## 40   57.36532  -8.2202874 122.95093
## 41   74.93424   9.5179301 140.35056
## 42   45.08757 -20.6653089 110.84045
## 43   63.33043  -2.1884217 128.84928
## 44   86.12843  20.7766947 151.48016
## 45   73.02952   7.5988570 138.46018
## 46  153.87447  88.1921218 219.55682
## 47  113.50889  48.1725126 178.84526
## 48  114.18079  48.8422401 179.51933
## 49   60.16301  -5.3901071 125.71612
## 50   58.56735  -7.0040382 124.13874
## 51   98.00216  32.6819690 163.32235
## 52   76.07161  10.6634002 141.47982
## 53   81.32605  15.9507484 146.70135
## 54   42.99426 -22.7911351 108.77966
## 55   54.94750 -10.6678717 120.56288
## 56   55.75549  -9.8497616 121.36075
## 57   90.85916  25.5245643 156.19377
## 58   85.39432  20.0393844 150.74925
## 59   82.44076  17.0714867 147.81004
## 60   75.69294  10.2820718 141.10381
## 61   51.69480 -13.9630768 117.35268
## 62   80.20740  14.8257200 145.58909
## 63   60.56067  -4.9879960 126.10933
## 64  157.40805  91.6746644 223.14144
## 65  188.88302 122.5499339 255.21611
## 66  189.67752 123.3259492 256.02908
## 67  120.82261  55.4560561 186.18917
## 68  149.10094  83.4822641 214.71961
## 69  147.59435  81.9945150 213.19419
## 70  153.57822  87.9000000 219.25644
## 71  153.57822  87.9000000 219.25644
## 72  135.27686  69.8082566 200.74546
## 73  122.46156  57.0862606 187.83686
## 74  132.12126  66.6797282 197.56278
## 75  156.52834  90.8079697 222.24871
## 76  132.12126  66.6797282 197.56278
## 77  141.49540  75.9656230 207.02518
## 78  144.55954  78.9958046 210.12327
## 79  150.60040  84.9623781 216.23842
## 80  159.45127  93.6868594 225.21568
```

```
## 81  143.03117  77.4846874 208.57766
## 82  162.34750  96.5372255 228.15778
## 83  162.34750  96.5372255 228.15778
## 84  168.06181 102.1545358 233.96909
## 85  157.99317  92.2510144 223.73533
## 86  144.55954  78.9958046 210.12327
## 87  165.21753  99.3596089 231.07545
## 88  157.99317  92.2510144 223.73533
## 89  192.57375 126.1534437 258.99406
## 90  159.45127  93.6868594 225.21568
## 91  124.09208  58.7073648 189.47679
## 92  149.10094  83.4822641 214.71961
## 93  125.71426  60.3194659 191.10905
## 94  114.18079  48.8422401 179.51933
## 95  122.46156  57.0862606 187.83686
## 96  147.59435  81.9945150 213.19419
## 97  150.60040  84.9623781 216.23842
## 98  128.93394  63.5170394 194.35084
## 99  102.21199  36.8938286 167.53014
## 100 139.95216  74.4385304 205.46579
## 101 150.60040  84.9623781 216.23842
## 102 159.74209  93.9731705 225.51100
## 103 176.44474 110.3796271 242.50985
## 104 135.27686  69.8082566 200.74546
## 105 132.43823  66.9941064 197.88236
## 106 110.12788  44.8005734 175.45519
## 107 137.77893  72.2869570 203.27090
## 108 156.23456  90.5184906 221.95063
## 109 157.99317  92.2510144 223.73533
```

```
# R-squared value for the above model is 0.6723.
# we may have to do different transformation better R-squared value
# Applying different transformations

# Exponential model
reg_exp<-lm(log(AT) ~ (sqrt(Waist*Waist*Waist))) # regression using Exponential model
summary(reg_exp)
```

```
##
## Call:
## lm(formula = log(AT) ~ (sqrt(Waist * Waist * Waist)))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.09333 -0.24324  0.02966  0.23507  0.84142
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 1.9918266  0.1620426   12.29   <2e-16 ***
## sqrt(Waist * Waist * Waist) 0.0027568  0.0001782   15.47   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3625 on 107 degrees of freedom
## Multiple R-squared:  0.691,  Adjusted R-squared:  0.6881
```

```
## F-statistic: 239.3 on 1 and 107 DF,  p-value: < 2.2e-16
```

```
# R-squared value has increased from 0.67 to 0.7071
# Higher the R-sqaured value - Better chances of getting good model
# for Waist and addipose Tissue
```

```
```