Data Analysis Plots

Varun Krishnan, Shreya Patel, Abhishek Modoor
vk318, sp1625, avm67

4/17/2020

Abstract

We decided to perform R data analysis using Scatter, Stem-and-Leaf, Histogram, Box and Whisker, Ellipse, Residual, Quantile-Quantile Plots on the prostate dataset.

The prostate dataset provides data correlation between the level of prostate specific antigen and a number of clinical measures in men who were about to receive a radical prostatectomy

Through data analysis, we aim to observe patterns in the distribution of the data points (scatter, stem & leaf, box & whisker, histogram), the mapping accuracy to a common distribution (quantile-quantile plots), regression analysis (residual), and measure confidence (ellipse).

Materials + Methods

Data Analysis Methods:

- Scatter, Stem-and-Leaf
- Histogram
- Box and Whisker
- Ellipse
- Residual
- Quantile-Quantile

Materials Used:

- RStudio, R, prostate dataset

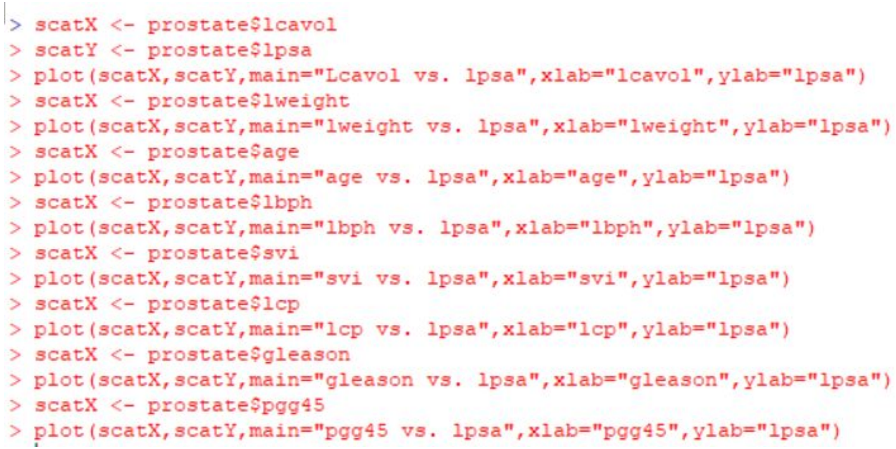
Results

<u>Scatter Plot</u>

- Scatterplots are plots where every individual data point is plotted according to their X and Y values. Scatterplots are an easy way to find out the relationship between two values by looking at the trend of the graph(positive/negative trend and strong/weak relationship)
- For the prostate dataset, we created scatterplots for every individual variable against the response lpsa (log prostate specific antigen) .

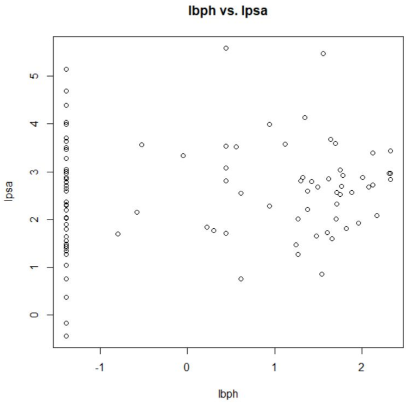
Scatter Plot code

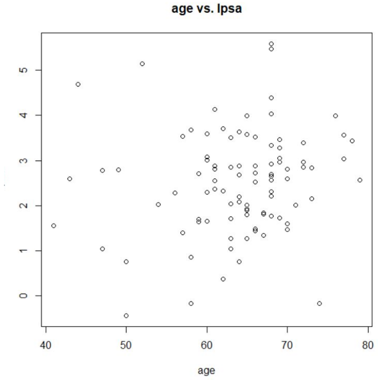
- We created scatterplots for every predictor variable plotted against the response variable lpsa.

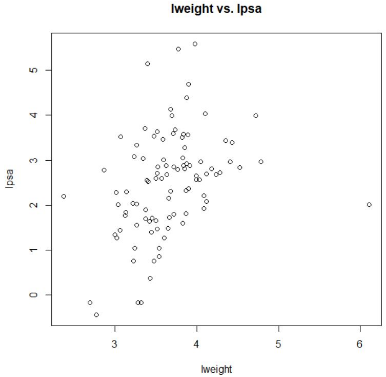


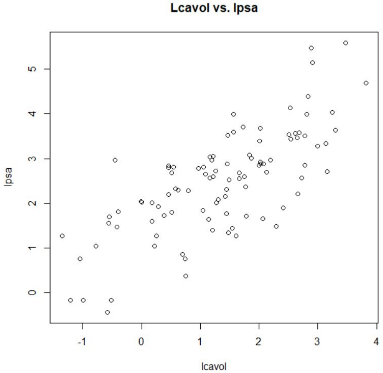
```
> scatX <- prostate$lcavol
> scatY <- prostate$lpsa
> plot(scatX,scatY,main="Lcavol vs. lpsa",xlab="lcavol",ylab="lpsa")
> scatX <- prostate$lweight
> plot(scatX,scatY,main="lweight vs. lpsa",xlab="lweight",ylab="lpsa")
> scatX <- prostate$age
> plot(scatX,scatY,main="age vs. lpsa",xlab="age",ylab="lpsa")
> scatX <- prostate$lbph
> plot(scatX,scatY,main="lbph vs. lpsa",xlab="lbph",ylab="lpsa")
> scatX <- prostate$svi
> plot(scatX,scatY,main="svi vs. lpsa",xlab="svi",ylab="lpsa")
> scatX <- prostate$lcp
> plot(scatX,scatY,main="lcp vs. lpsa",xlab="lcp",ylab="lpsa")
> scatX <- prostate$gleason
> plot(scatX,scatY,main="gleason vs. lpsa",xlab="gleason",ylab="lpsa")
> scatX <- prostate$pgg45
> plot(scatX,scatY,main="pgg45 vs. lpsa",xlab="pgg45",ylab="lpsa")
```

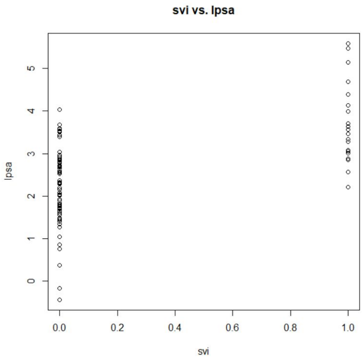
Scatter Plot graphs

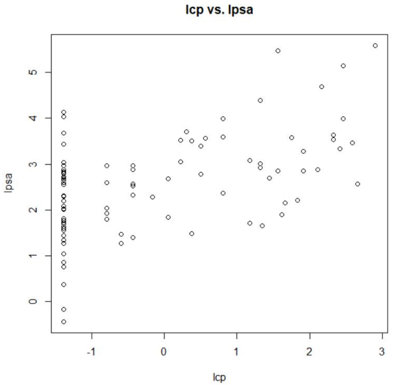


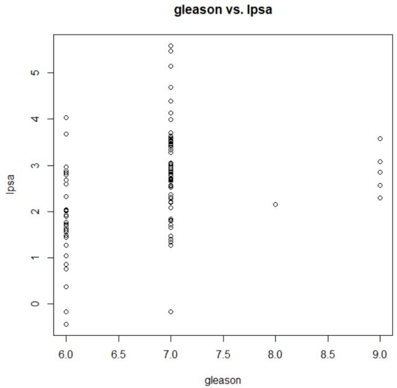


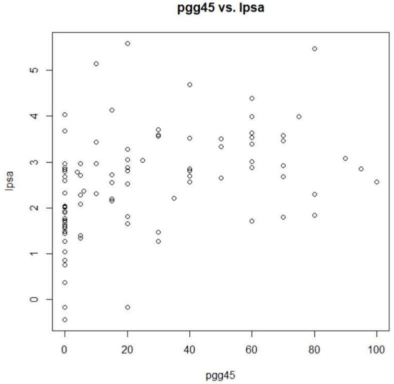












Discussion

Scatterplot

- From the graphs in the previous slide, we can see that lcavol (log cancer volume) and lweight (log prostate weight) have a strong and positive relationship with the response variable lpsa(log prostate antigen).
- The graphs also show us that categorical and scored variables in the dataset(svi and gleason) do not provide much information.
- The other graphs show that there is not much of any correlation between the predictor and response variable.

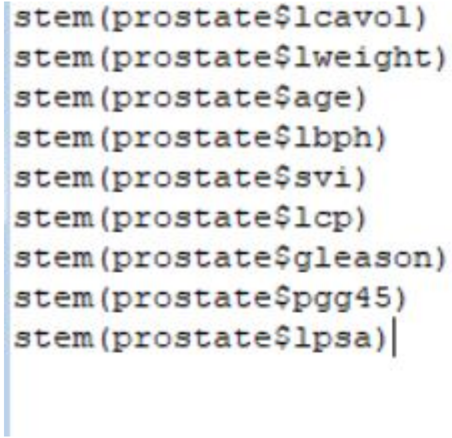
Results

Stem and Leaf Plot

- A stem and leaf plot allows us to visualize the shape of how a variable is distributed. It does this by splitting the data points into categories called stems based on the first n digits and leaves which are the rest of the digits after the first n.
- We produced stem and leaf plots for every variable in the prostate dataset.

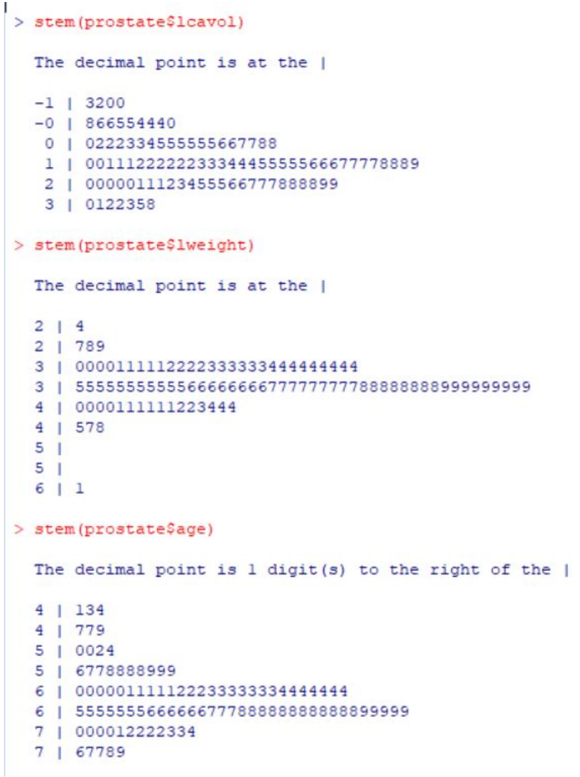
Stem and leaf plots code

- We created stem and leaf plots for every variable in the prostate dataset including the response.

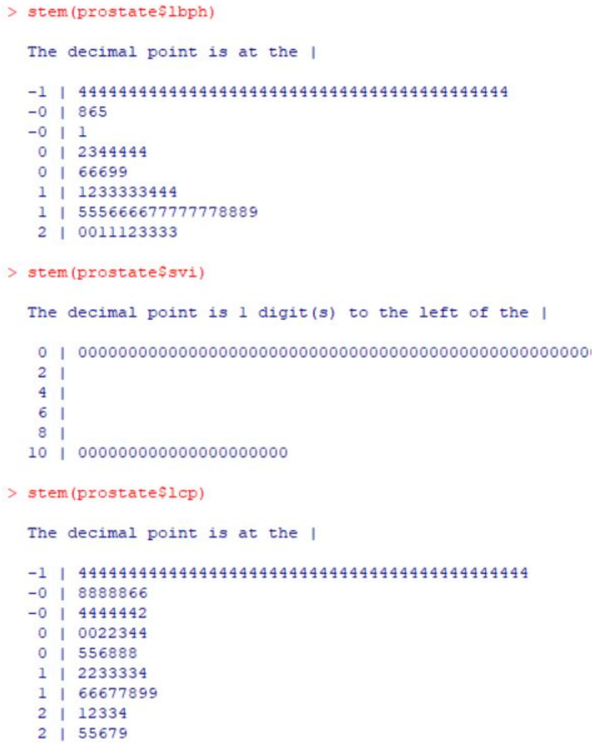


```
stem(prostate$lcavol)
stem(prostate$lweight)
stem(prostate$age)
stem(prostate$lbph)
stem(prostate$svi)
stem(prostate$lcp)
stem(prostate$gleason)
stem(prostate$pgg45)
stem(prostate$lpsa)
```

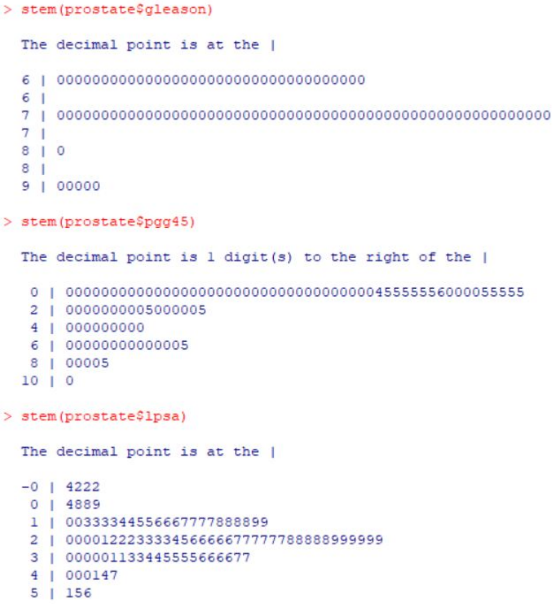
Stem and leaf plots



```
> stem(prostate$lcavol)

  The decimal point is at the |

  -1 | 3200
  -0 | 866554440
   0 | 0222334555555667788
   1 | 0011122222233344455555666777778889
   2 | 0000011123455566777888899
   3 | 0122358

> stem(prostate$lweight)

  The decimal point is at the |

  2 | 4
  2 | 789
  3 | 0000111112222333333444444444
  3 | 555555555556666666667777777778888888899999999
  4 | 0000111111223444
  4 | 578
  5 |
  5 |
  6 | 1

> stem(prostate$age)

  The decimal point is 1 digit(s) to the right of the |

  4 | 134
  4 | 779
  5 | 0024
  5 | 6778888999
  6 | 0000011111222333333344444444
  6 | 55555556666667778888888888888899999
  7 | 000012222334
  7 | 67789
```



```
> stem(prostate$lbph)

  The decimal point is at the |

  -1 | 4444444444444444444444444444444444444444444
  -0 | 865
  -0 | 1
   0 | 2344444
   0 | 66699
   1 | 1233333444
   1 | 555666677777778889
   2 | 0011123333

> stem(prostate$svi)

  The decimal point is 1 digit(s) to the left of the |

   0 | 000000000000000000000000000000000000000000000000000
   2 |
   4 |
   6 |
   8 |
  10 | 000000000000000000000

> stem(prostate$lcp)

  The decimal point is at the |

  -1 | 444444444444444444444444444444444444444444444
  -0 | 8888866
  -0 | 4444442
   0 | 0022344
   0 | 556888
   1 | 2233334
   1 | 66677899
   2 | 12334
   2 | 55679
```



```
> stem(prostate$gleason)

  The decimal point is at the |

  6 | 00000000000000000000000000000000000
  6 |
  7 | 00000000000000000000000000000000000000000000000000000000
  7 |
  8 | 0
  8 |
  9 | 00000

> stem(prostate$pgg45)

  The decimal point is 1 digit(s) to the right of the |

   0 | 0000000000000000000000000000000000045555556000055555
   2 | 0000000005000005
   4 | 000000000
   6 | 00000000000005
   8 | 00005
  10 | 0

> stem(prostate$lpsa)

  The decimal point is at the |

  -0 | 4222
   0 | 4889
   1 | 0033334455666777788899
   2 | 00001222333345666667777778888899999
   3 | 0000011334455556666 77
   4 | 000147
   5 | 156
```

Discussion

Stem and Leaf plots

- The stem and leaf plots help us deduce the shape of how every variable is distributed. Many of the variables have distributions that are close to normal and symmetric.
- The stem and leaf plots of the categorical and scored variables in the dataset(svi and gleason) do not provide any information to us other than which category is more common.
- The lbph, lcp and pgg45 are heavily right skewed to the right.

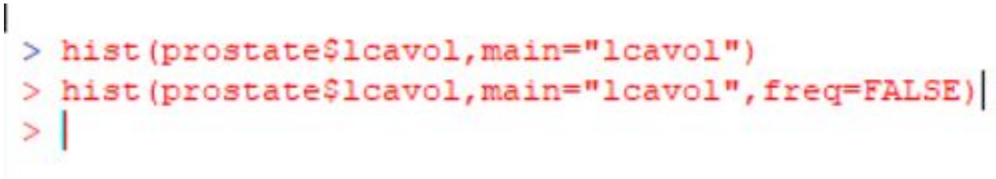
Results

Histogram

- A histogram is similar to a stem and leaf plot in that it provides very similar information about the shape of a distribution, however a histogram also allows us to plot a variable’s density.
- A density plot shows how a variable is distributed where the area of the bars have a total sum of 1. The alternative to this is a simple frequency chart.
- A frequency chart just shows the amount of points in a dataset that fall in a certain range.

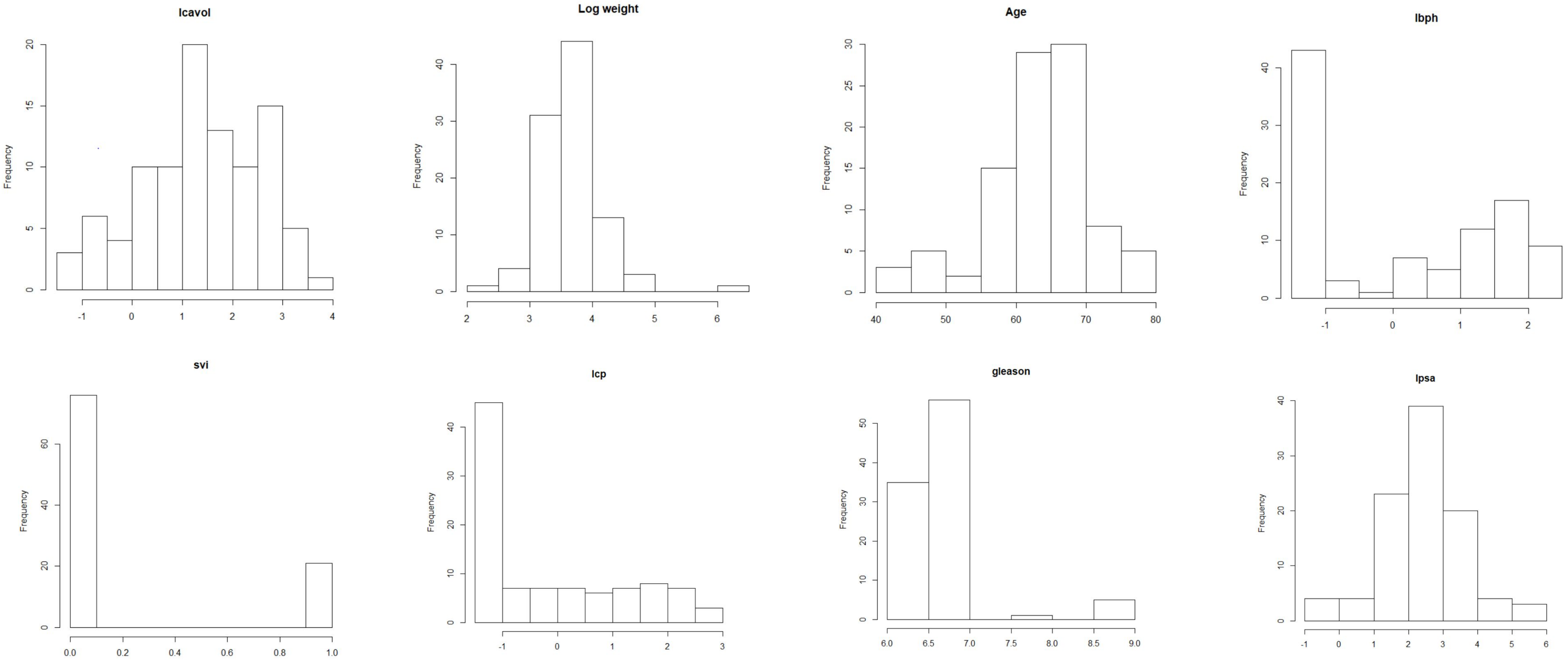
Histogram code

- We created histograms for every variable in the prostate dataset including the response.
- We also plotted density variations of the same histogram.



```
> hist(prostate$lcavol,main="lcavol")
> hist(prostate$lcavol,main="lcavol",freq=FALSE)
> 
```

Histograms


lcavol
Frequency
0
5
10
15
20
-1
0
1
2
3
4
Log weight
Frequency
0
10
20
30
40
2
3
4
5
6
Age
Frequency
0
5
10
15
20
25
30
40
50
60
70
80
lbph
Frequency
0
10
20
30
40
-1
0
1
2
svi
Frequency
0
20
40
60
0.0
0.2
0.4
0.6
0.8
1.0
lcp
Frequency
0
10
20
30
40
-1
0
1
2
3
gleason
Frequency
0
10
20
30
40
50
6.0
6.5
7.0
7.5
8.0
8.5
9.0
lpsa
Frequency
0
10
20
30
40
-1
0
1
2
3
4
5
6

Discussion

Histogram

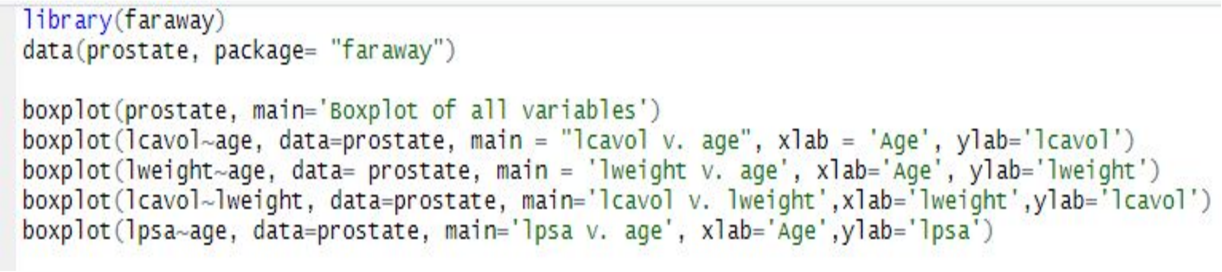
- The histograms provide information about the shape of the distribution of a variable.
- For categorical data, we can see which categories are more common among the dataset. For example, the variable svi (seminal vesicle invasion) is a boolean variable and it is more likely that this variable is false(0) in this dataset. The gleason histogram tells us that a Gleason score that is greater than 7 is unlikely.
- For the quantitative variables, we can see which range of values the data peaks at which gives us an idea of how the variable is distributed.

Results

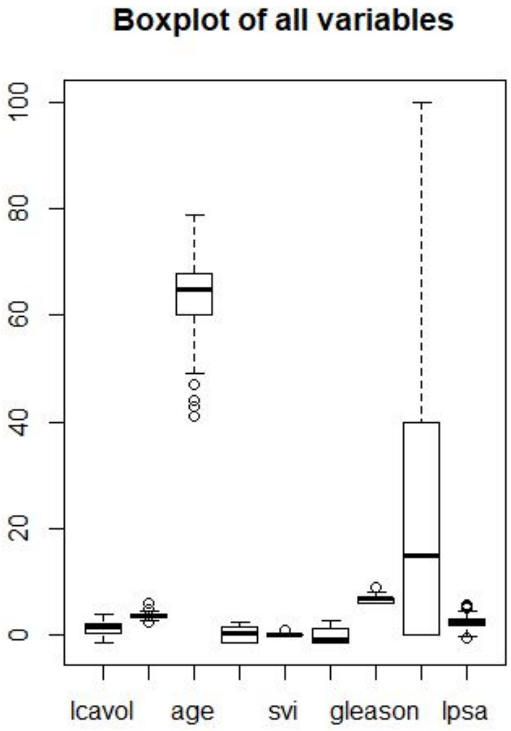
Box & Whisker Plot

- Box and whisker plot are used to show distribution of data. It is not as detailed as a histogram or stem & leaf plot, but it is more useful in discerning outliers and skewness in the data.
- For the prostate dataset, we created box & whisker plots for all the variables in the dataset, and then plots that compared the specific variables to each either in order to notice patterns.

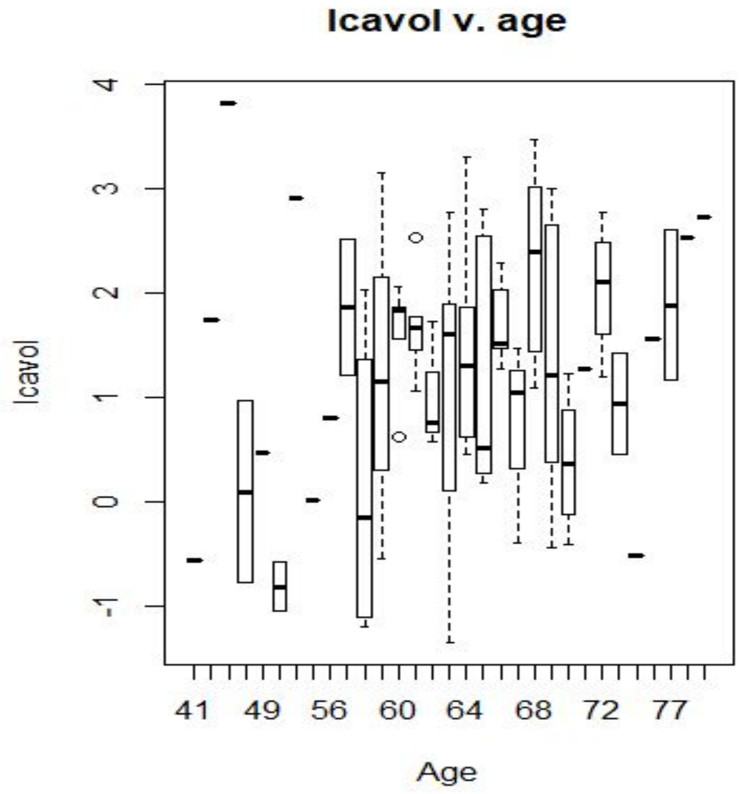
Box & Whisker Plot code

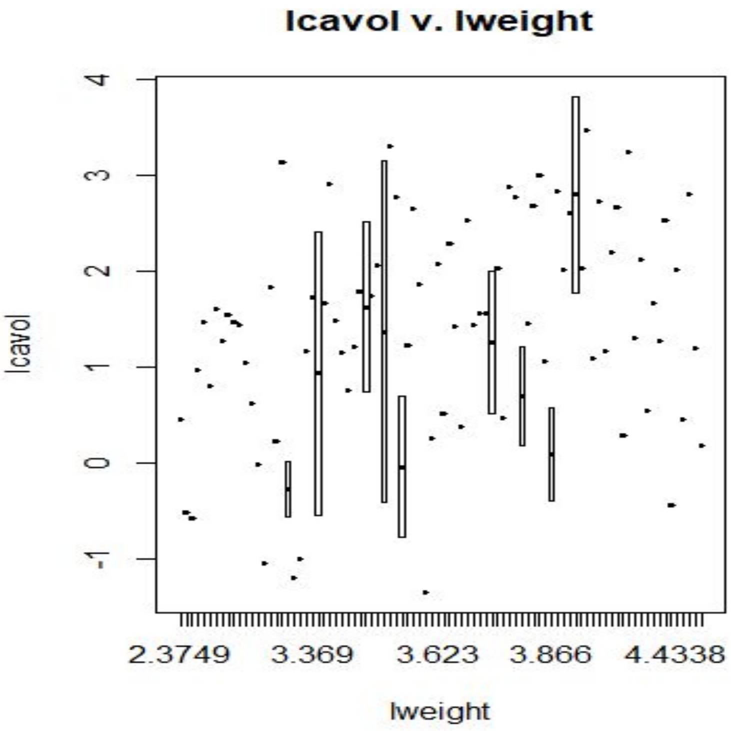


```
library(faraway)
data(prostate, package= "faraway")

boxplot(prostate, main='Boxplot of all variables')
boxplot(lcavol~age, data=prostate, main = "lcavol v. age", xlab = 'Age', ylab='lcavol')
boxplot(lweight~age, data= prostate, main = 'lweight v. age', xlab='Age', ylab='lweight')
boxplot(lcavol~lweight, data=prostate, main='lcavol v. lweight',xlab='lweight',ylab='lcavol')
boxplot(lpsa~age, data=prostate, main='lpsa v. age', xlab='Age',ylab='lpsa')
```

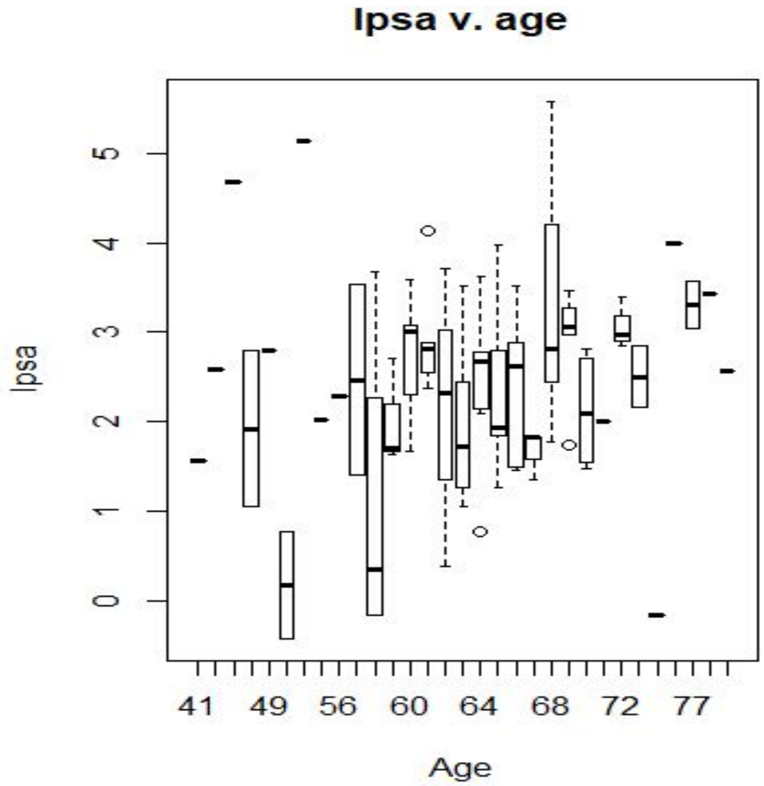


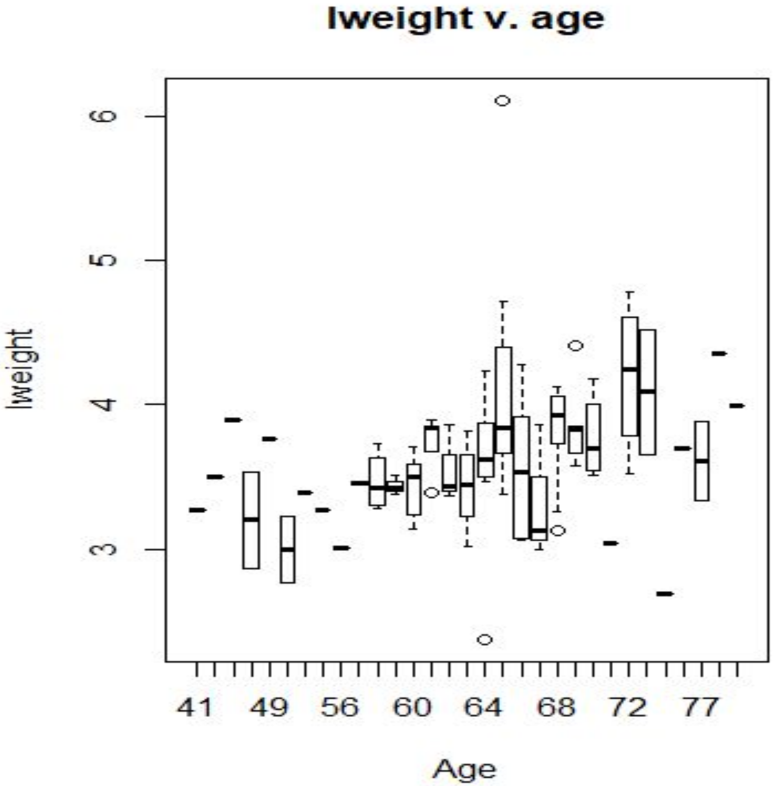
Graphs for Box & Whisker Plot





Graph for Box & Whisker Plot





Discussion

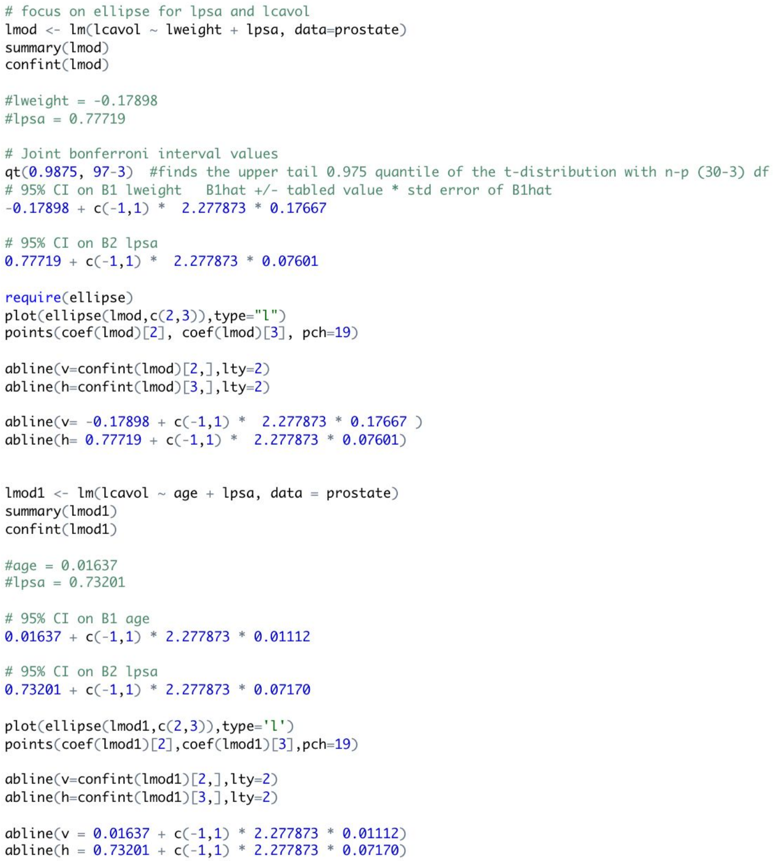
- Based on the first box & whisker plot, we can see that pgg45 has the largest distribution due to the largest box compared to the other variables. We also can see one outlier from the dataset within pgg45
- We also see that the age isn’t distributed amongst the young, it’s all distributed amongst the elderly and people above the age of 40.
- With the plot that compares lcavol to age, we see that overall the average lcavol value increases with age and the distribution of the data becomes slightly smaller (smaller boxes) as age increases.
- We also notice the same trend between lweight and age where average lweight tends to increase with age.
- However, we don’t see any patterns in distribution when comparing lcavol and lweight.
- When comparing lpsa with age, we see that the distribution is varied across and doesn’t have a general pattern in terms of increasing or decreasing linearity.

Results

Ellipsoid

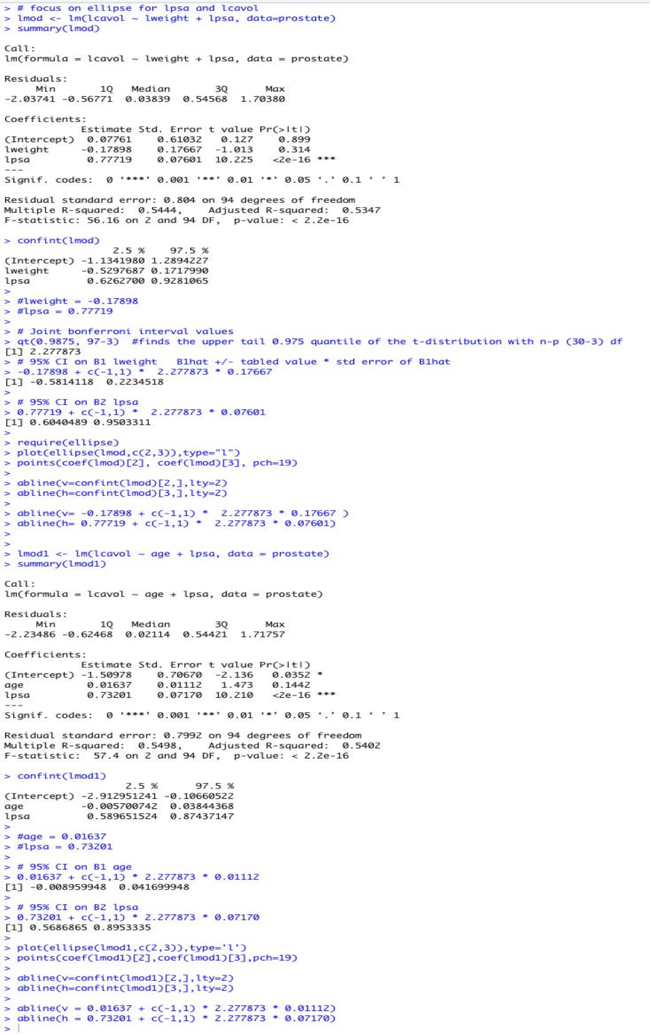
- We constructed ellipsoid CI’s by focusing on two variables at a time from the prostate dataset
- The variables we focused on were between lweight & lpsa, and age & lpsa.
- You can look at the code in the next slide in order to see how we constructed the ellipsoid CI’s.
- We did so by performing a linear regression of the variables and plotting the bonferroni beta values to construct the boundaries of our ellipsoid graph.

Code for Ellipse



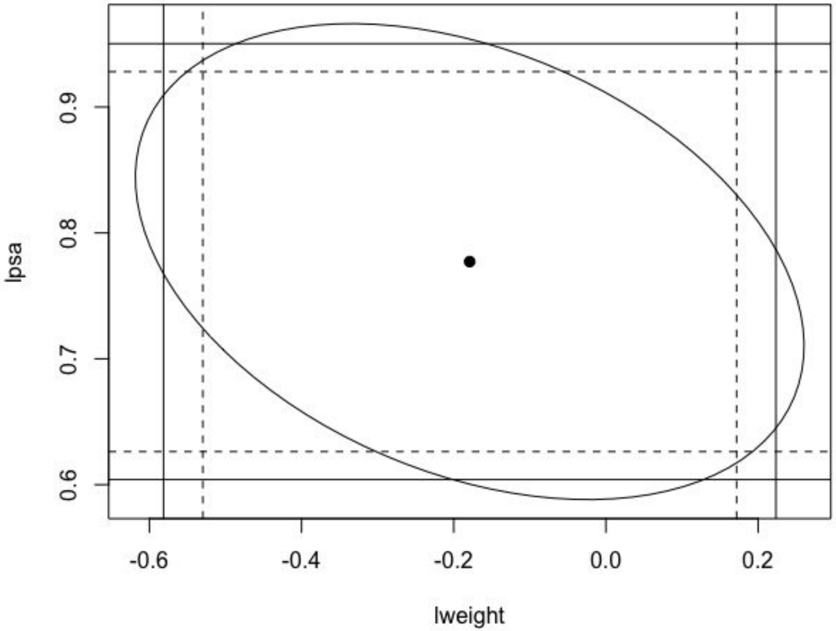
```
# focus on ellipse for lpsa and lcavol
lmod <- lm(lcavol ~ lweight + lpsa, data=prostate)
summary(lmod)
confint(lmod)

#lweight = -0.17898
#lpsa = 0.77719

# Joint bonferroni interval values
qt(0.9875, 97-3)  #finds the upper tail 0.975 quantile of the t-distribution with n-p (30-3) df
# 95% CI on B1 lweight   B1hat +/- tabled value * std error of B1hat
-0.17898 + c(-1,1) *  2.277873 * 0.17667

# 95% CI on B2 lpsa
0.77719 + c(-1,1) *  2.277873 * 0.07601

require(ellipse)
plot(ellipse(lmod,c(2,3)),type="l")
points(coef(lmod)[2], coef(lmod)[3], pch=19)

abline(v=confint(lmod)[2,],lty=2)
abline(h=confint(lmod)[3,],lty=2)

abline(v= -0.17898 + c(-1,1) *  2.277873 * 0.17667 )
abline(h= 0.77719 + c(-1,1) *  2.277873 * 0.07601)


lmod1 <- lm(lcavol ~ age + lpsa, data = prostate)
summary(lmod1)
confint(lmod1)

#age = 0.01637
#lpsa = 0.73201

# 95% CI on B1 age
0.01637 + c(-1,1) * 2.277873 * 0.01112

# 95% CI on B2 lpsa
0.73201 + c(-1,1) * 2.277873 * 0.07170

plot(ellipse(lmod1,c(2,3)),type='l')
points(coef(lmod1)[2],coef(lmod1)[3],pch=19)

abline(v=confint(lmod1)[2,],lty=2)
abline(h=confint(lmod1)[3,],lty=2)

abline(v = 0.01637 + c(-1,1) * 2.277873 * 0.01112)
abline(h = 0.73201 + c(-1,1) * 2.277873 * 0.07170)
```

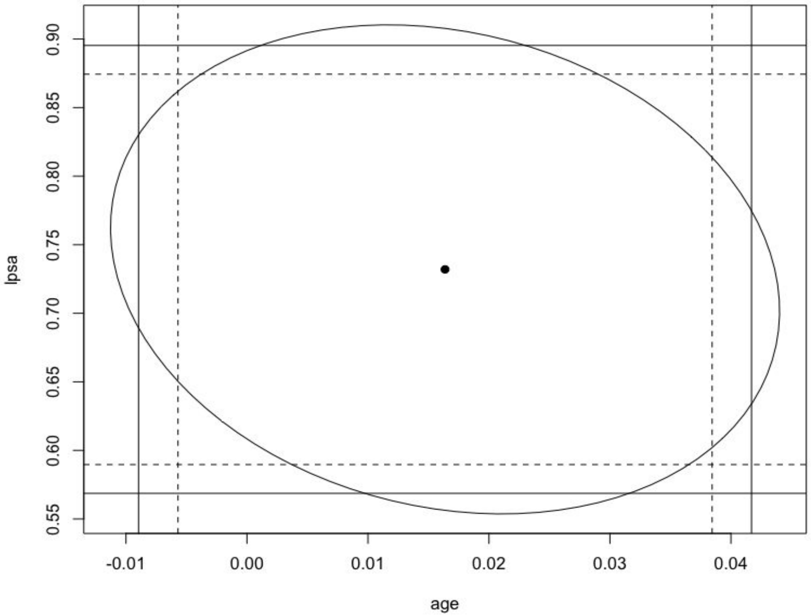
Output for Ellipse



```
> # focus on ellipse for lpsa and lcavol
> lmod <- lm(lcavol ~ lweight + lpsa, data=prostate)
> summary(lmod)

Call:
lm(formula = lcavol ~ lweight + lpsa, data = prostate)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.03741 -0.56771  0.03839  0.54568  1.70380 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.07761    0.61032   0.127    0.899    
lweight     -0.17898    0.17667  -1.013    0.314    
lpsa         0.77719    0.07601  10.225   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.804 on 94 degrees of freedom
Multiple R-squared:  0.5444,	Adjusted R-squared:  0.5347 
F-statistic: 56.16 on 2 and 94 DF,  p-value: < 2.2e-16

> confint(lmod)
                 2.5 %    97.5 %
(Intercept) -1.1341980 1.2894227
lweight     -0.5297687 0.1717990
lpsa         0.6262700 0.9281065
> 
> #lweight = -0.17898
> #lpsa = 0.77719
> 
> # Joint bonferroni interval values
> qt(0.9875, 97-3)  #finds the upper tail 0.975 quantile of the t-distribution with n-p (30-3) df
[1] 2.277873
> # 95% CI on B1 lweight   B1hat +/- tabled value * std error of B1hat
> -0.17898 + c(-1,1) *  2.277873 * 0.17667
[1] -0.5814118  0.2234518
> 
> # 95% CI on B2 lpsa
> 0.77719 + c(-1,1) *  2.277873 * 0.07601
[1] 0.6040489 0.9503311
> 
> require(ellipse)
> plot(ellipse(lmod,c(2,3)),type="l")
> points(coef(lmod)[2], coef(lmod)[3], pch=19)
> 
> abline(v=confint(lmod)[2,],lty=2)
> abline(h=confint(lmod)[3,],lty=2)
> 
> abline(v= -0.17898 + c(-1,1) *  2.277873 * 0.17667 )
> abline(h= 0.77719 + c(-1,1) *  2.277873 * 0.07601)
> 
> 
> lmod1 <- lm(lcavol ~ age + lpsa, data = prostate)
> summary(lmod1)

Call:
lm(formula = lcavol ~ age + lpsa, data = prostate)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.23486 -0.62468  0.02114  0.54421  1.71757 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.50978    0.70670  -2.136   0.0352 *  
age          0.01637    0.01112   1.473   0.1442    
lpsa         0.73201    0.07170  10.210   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7992 on 94 degrees of freedom
Multiple R-squared:  0.5498,	Adjusted R-squared:  0.5402 
F-statistic:  57.4 on 2 and 94 DF,  p-value: < 2.2e-16

> confint(lmod1)
                   2.5 %      97.5 %
(Intercept) -2.912951241 -0.10660522
age         -0.005700742  0.03844368
lpsa         0.589651524  0.87437147
> 
> #age = 0.01637
> #lpsa = 0.73201
> 
> # 95% CI on B1 age
> 0.01637 + c(-1,1) * 2.277873 * 0.01112
[1] -0.008959948  0.041699948
> 
> # 95% CI on B2 lpsa
> 0.73201 + c(-1,1) * 2.277873 * 0.07170
[1] 0.5686865 0.8953335
> 
> plot(ellipse(lmod1,c(2,3)),type='l')
> points(coef(lmod1)[2],coef(lmod1)[3],pch=19)
> 
> abline(v=confint(lmod1)[2,],lty=2)
> abline(h=confint(lmod1)[3,],lty=2)
> 
> abline(v = 0.01637 + c(-1,1) * 2.277873 * 0.01112)
> abline(h = 0.73201 + c(-1,1) * 2.277873 * 0.07170)
> 
```

Graphs for Ellipse





Discussion

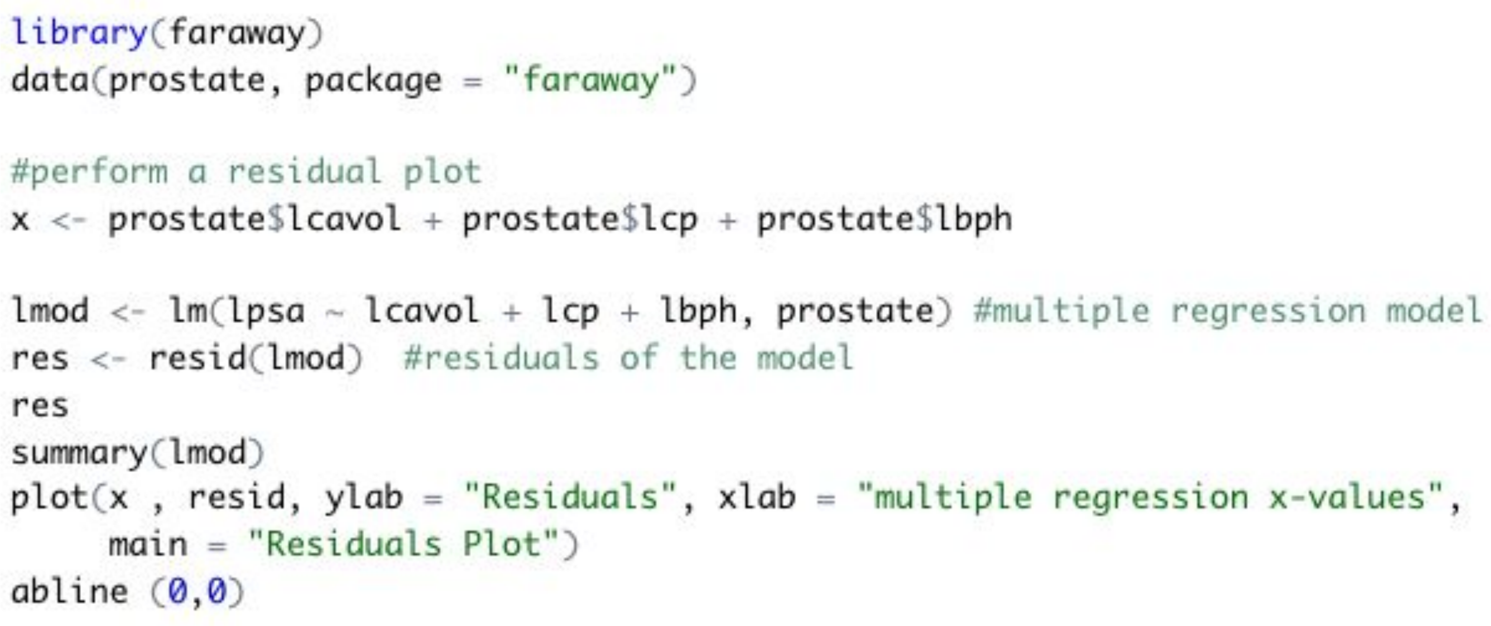
- What these ellipses tell us is the distribution of the joint intervals between lpsa & lweight, and lpsa & age.
- We notice that between lweight and lpsa , the mu values are 0.777 and -0.178, respectively. Between age and lpsa, the mu values are 0.01637 and 0.73201, respectively.
- We see that the ellipses for both regressions are rather large, which shows us that the correlation isn’t as tight as we would want it.
- In addition, the lweight and age values have intervals that include negative and positive values, so we can’t predict if they are positive or negative (which is seen in the graph).
- Meanwhile, lpsa values are between 0.604 and 0.905, telling us lpsa values could be positive for men with prostate cancer.

Results

Residual Plot

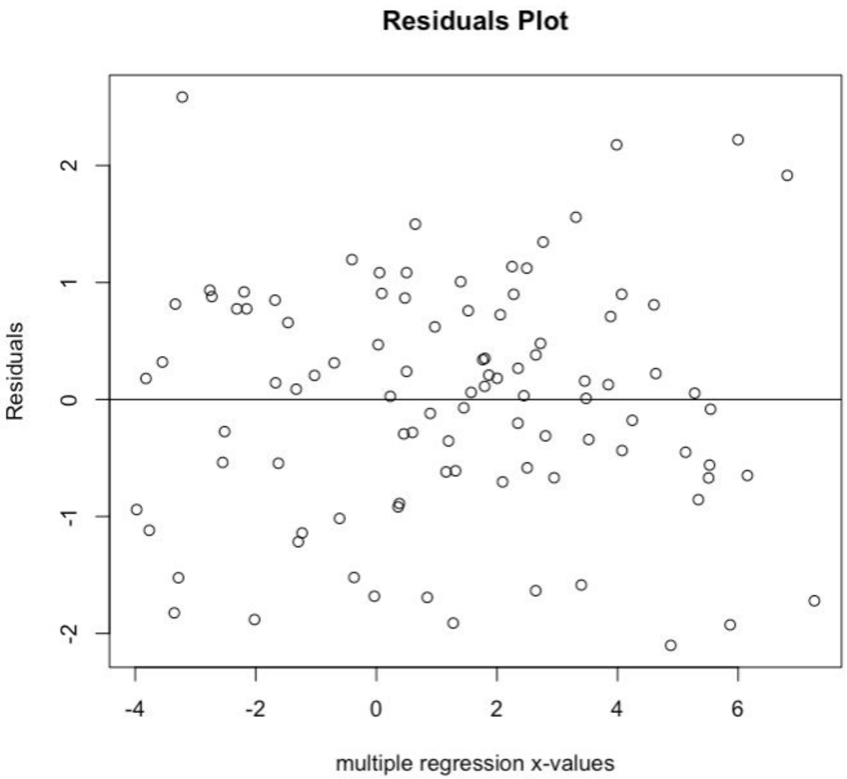
- Residual Plots are used to determine what type of regression is fit for the given data set
- For the prostate data set, a multiple linear regression model was created using lpsa as the dependant variable and (lcavol + lcp + lbph) as the multiple regression x-values
- Then the residuals for the data was computed and plotted
- The next slide provides the code and the residual plot for this data

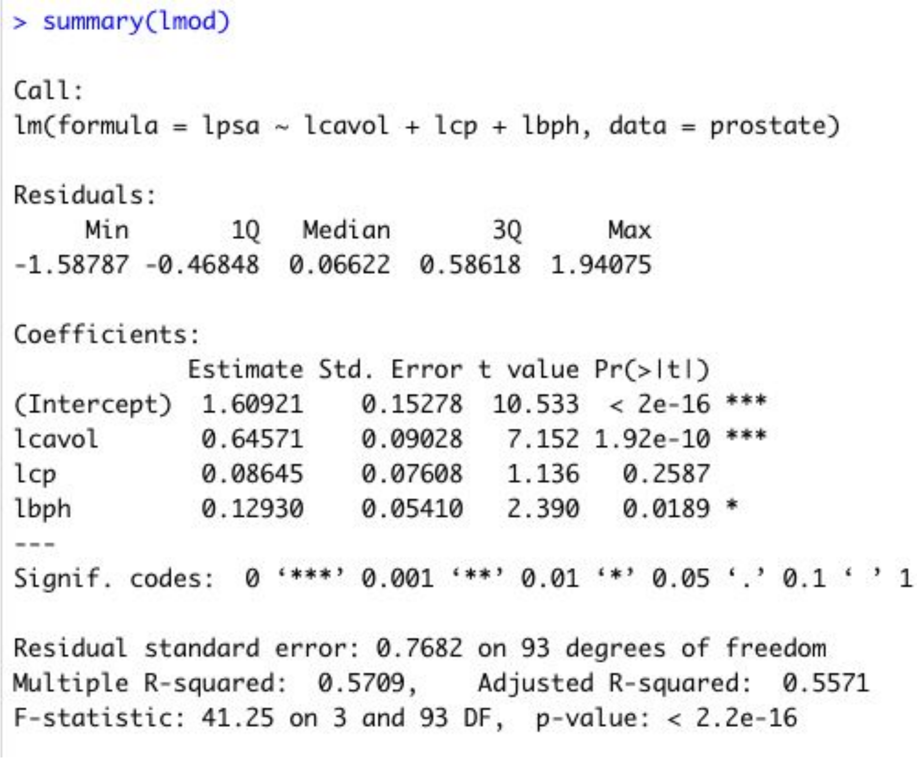
Residual Plot code



```r
library(faraway)
data(prostate, package = "faraway")

#perform a residual plot
x <- prostate$lcavol + prostate$lcp + prostate$lbph

lmod <- lm(lpsa ~ lcavol + lcp + lbph, prostate) #multiple regression model
res <- resid(lmod)  #residuals of the model
res
summary(lmod)
plot(x , resid, ylab = "Residuals", xlab = "multiple regression x-values",
     main = "Residuals Plot")
abline (0,0)
```

Graph for Residual Plot





```
> summary(lmod)

Call:
lm(formula = lpsa ~ lcavol + lcp + lbph, data = prostate)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.58787 -0.46848  0.06622  0.58618  1.94075 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.60921    0.15278  10.533  < 2e-16 ***
lcavol       0.64571    0.09028   7.152 1.92e-10 ***
lcp          0.08645    0.07608   1.136   0.2587    
lbph         0.12930    0.05410   2.390   0.0189 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7682 on 93 degrees of freedom
Multiple R-squared:  0.5709,	Adjusted R-squared:  0.5571 
F-statistic: 41.25 on 3 and 93 DF,  p-value: < 2.2e-16
```

Discussion

Residual Plot

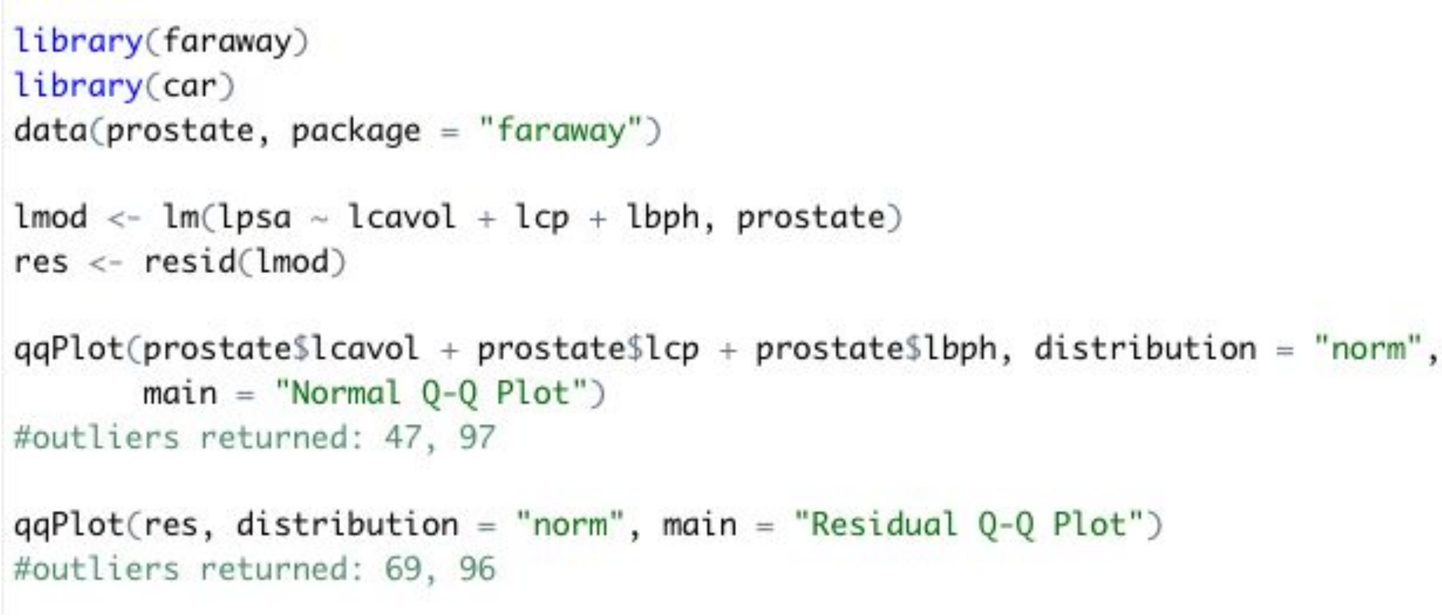
- The residual plot proves that the data is fit for a linear regression because the data points follow a random pattern. A good fit for linear regression is defined by a residual plot whose data points are randomly scattered across the horizontal axis.
- The summary of the data set shows us that residual standard error is relatively small which means that the predictions were better. This proves that the model is a good fit for the data set.

Results

Q-Q Plot

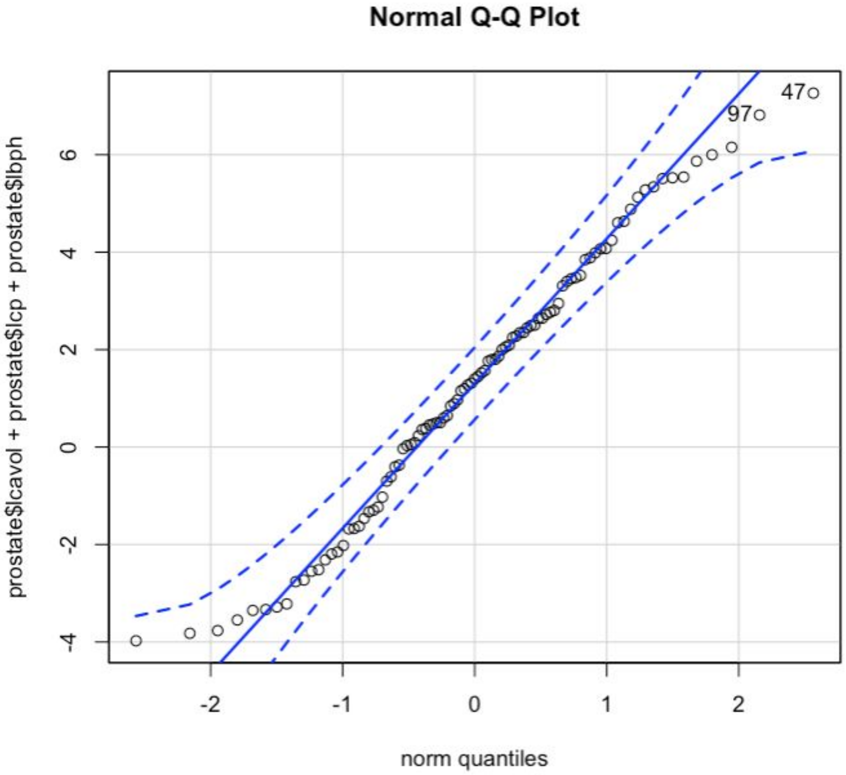
- Q-Q Plots are used to determine the distribution of one data set to the distribution of another, or for linear regression models to check for the normality of the model
- For the prostate data set, the Q-Q plot was generated to check the normality of the multiple x-values (lcavol + lcp + lbph) against a default normal quantile distribution.
- The residuals for the data set was also checked for normality where residuals were plotted against a default normal quantile distribution

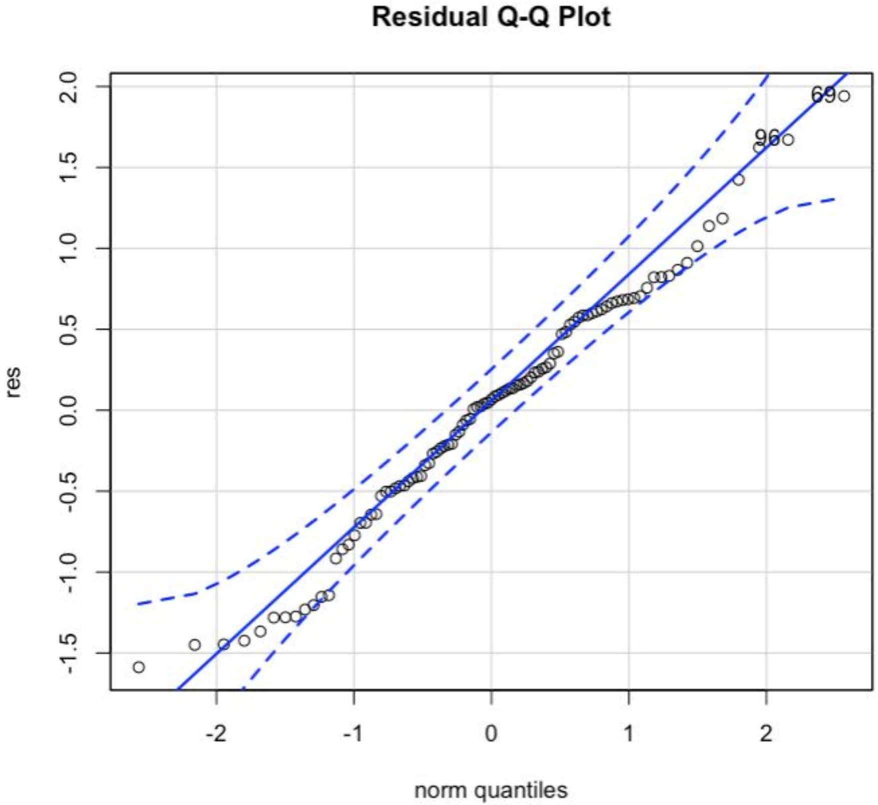
Q-Q Plot code



```r
library(faraway)
library(car)
data(prostate, package = "faraway")

lmod <- lm(lpsa ~ lcavol + lcp + lbph, prostate)
res <- resid(lmod)

qqPlot(prostate$lcavol + prostate$lcp + prostate$lbph, distribution = "norm",
       main = "Normal Q-Q Plot")
#outliers returned: 47, 97

qqPlot(res, distribution = "norm", main = "Residual Q-Q Plot")
#outliers returned: 69, 96
```

Graphs for Q-Q Plot





Discussion

Q-Q Plot

- From the graphs in the previous slide we can deduce that the data set is normal because the qqPlot function plotted a linear line for the multiple x-value dataset (lcavol + lcp + lbph). This means we are correct to assume a multiple linear regression.
- From the graph of the residuals is normal because the qqPlot function plotted a linear line for the residuals. This means that the residuals are normal so we are correct to assume that the residual plot is correct in predicting that the data set is a multiple regression model

Literature Cited

- https://rdrr.io/cran/faraway/man/prostate.html
- https://www.rdocumentation.org/packages/EnvStats/versions/2.3.1/topics/qqPlot
- https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/residuals
- https://www150.statcan.gc.ca/n1/edu/power-pouvoir/ch12/5214889-eng.htm
- https://www.statmethods.net/graphs/boxplot.html
- https://www.statmethods.net/graphs/density.html
- http://www.r-tutor.com/elementary-statistics/quantitative-data/stem-and-leaf-plot
- https://www.statmethods.net/graphs/scatterplot.html