Multivariate Analysis

Isabella Chittumuri

9/29/2020

3.11 Use the calcium data in Table 3.4:

```
# Create Table 3.4 calcium matrix
calcium <- matrix(c(35,35,40,10,6,20,35,35,35,30,3.5,4.9,30,2.8,2.7,2.8,4.6,10.9,8,1.6,2.8,2.7,4.38,3.2
dimnames(calcium) <- list(NULL, c("V1", "V2", "V3"))
```

(a) Find the generalized sample variance |S| as in (3.77).

```
# Generalized sample variance
calcium_cov <- cov(calcium)
det(calcium_cov)
```

```
## [1] 459.9555
```

(b) Find the total sample variance tr(S) as in (3.78).

```
# Total sample variance
sum(diag(calcium_cov))
```

```
## [1] 213.043
```

3.17 Define the following linear combinations for the variables in Table 3.4:

$z_1 = y_1 + y_2 + y_3$
$z_2 = 2y_1 - 3y_2 + 2y_3$
$z_3 = -y_1 - 2y_2 - 3y_3$

(a) Find $\bar{z}$ and S_z using (3.62) and (3.64).

```r
# zbar
calcium_ybar <- apply(calcium,2, mean); calcium_ybar
```

```
##     V1     V2     V3
## 28.100  7.180  3.089
```

```r
calcium_A <- matrix(c(1, 1, 1, 2, -3, 2, -1, -2, -3), nrow = 3, ncol = 3, byrow = T); calcium_A
```

```
##      [,1] [,2] [,3]
## [1,]    1    1    1
## [2,]    2   -3    2
## [3,]   -1   -2   -3
```

```r
calcium_zbar <- calcium_A %*% calcium_ybar; calcium_zbar
```

```
##         [,1]
## [1,]  38.369
## [2,]  40.838
## [3,] -51.727
```

```r
# S_z sample variance
calcium_sample_var <- calcium_A %*% calcium_cov %*% t(calcium_A); calcium_sample_var
```

```
##           [,1]     [,2]      [,3]
## [1,]  323.6376  19.2526 -460.9770
## [2,]   19.2526 588.6710  104.0717
## [3,] -460.9770 104.0717  686.2697
```

(b) Find R_z from S_z using (3.37).

```r
library(matlab)
```

```
##
## Attaching package: 'matlab'
```

```
## The following object is masked from 'package:stats':
##
##     reshape
```

```
## The following objects are masked from 'package:utils':
##
##     find, fix
```

```
## The following object is masked from 'package:base':
##
##     sum
```

```r
# $R_z$
diag(calcium_sample_var, names=T)
```

```
## [1] 323.6376 588.6710 686.2697
```

```r
n <- sqrt(diag(calcium_sample_var,names = T))
calcium_D <- diag(n); calcium_D
```

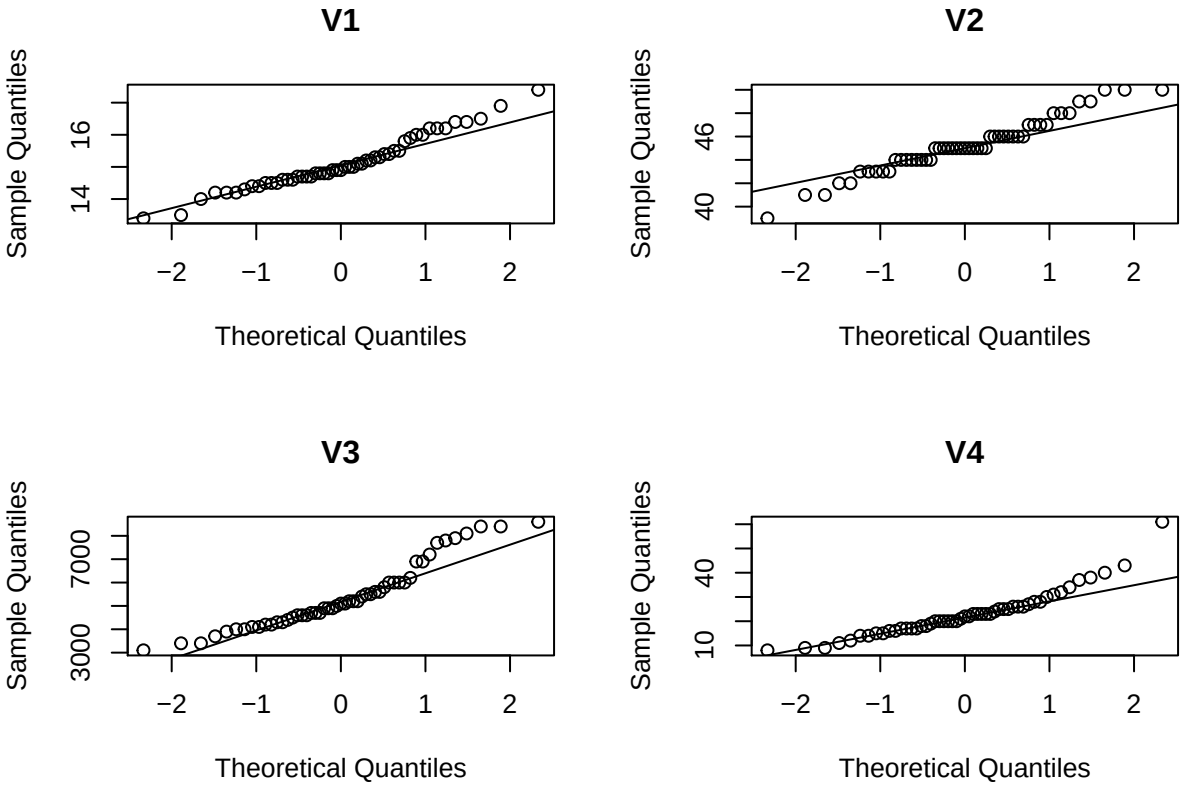
```
##          [,1]     [,2]     [,3]
## [1,] 17.98993  0.00000  0.00000
## [2,]  0.00000 24.26254  0.00000
## [3,]  0.00000  0.00000 26.19675
```

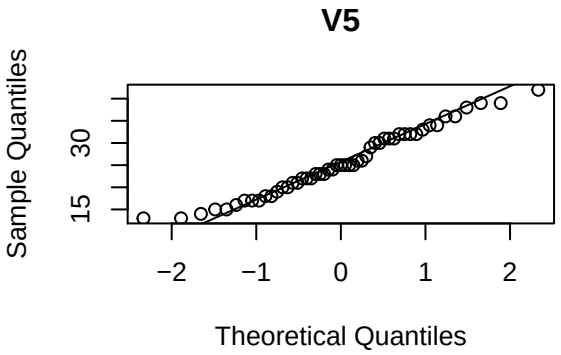
```r
calcium_D_inv <- solve(calcium_D)

calcium_Rz <- calcium_D_inv %*% calcium_sample_var %*% calcium_D_inv; calcium_Rz
```

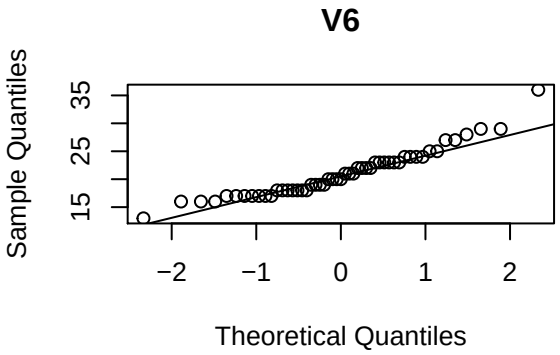
```
##             [,1]       [,2]       [,3]
## [1,]  1.00000000 0.04410862 -0.9781430
## [2,]  0.04410862 1.00000000  0.1637378
## [3,] -0.97814302 0.16373782  1.0000000
```

4.23 The data are given in Table 4.2. Check each of the six variables for univanate normality using the following tests: Q-Q plots, histograms, 2-dimensional scatter plots, chi-square plot, and Shapiro-Wilks test.

```r
hematol <- read.table("T4_2_HEMATOL.dat")

# Q-Q plots
par(mfrow=c(2,2))
for (i in 1:6) {
  qqnorm(hematol[,i],main=names(hematol)[i])
  qqline(hematol[,i])
}
```

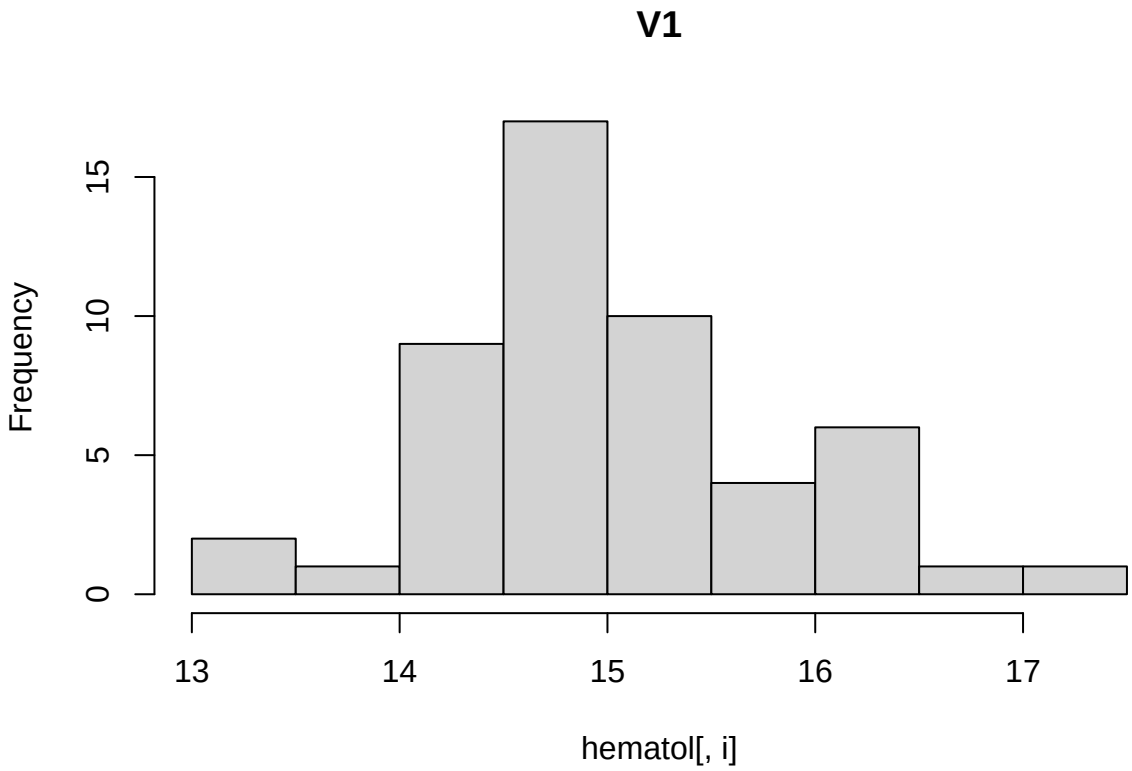

V1
Sample Quantiles
14
16
−2
−1
0
1
2
Theoretical Quantiles
V2
Sample Quantiles
40
46
−2
−1
0
1
2
Theoretical Quantiles
V3
Sample Quantiles
3000
7000
−2
−1
0
1
2
Theoretical Quantiles
V4
Sample Quantiles
10
40
−2
−1
0
1
2
Theoretical Quantiles

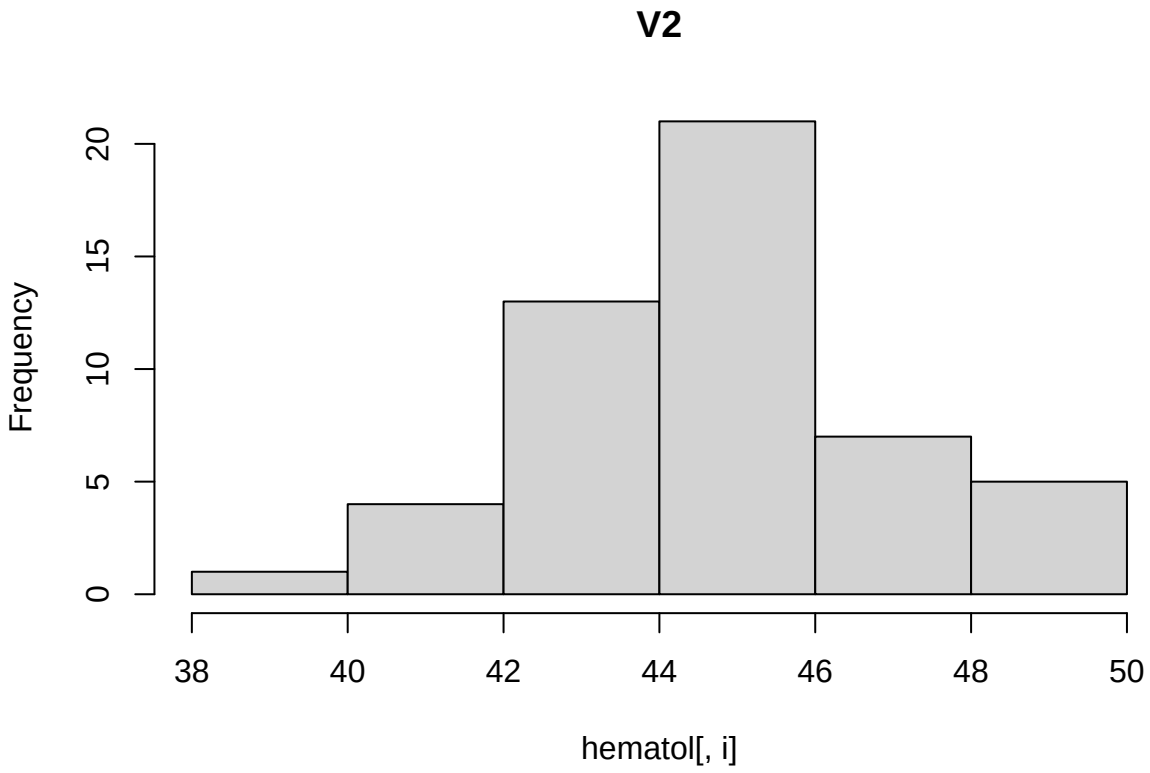


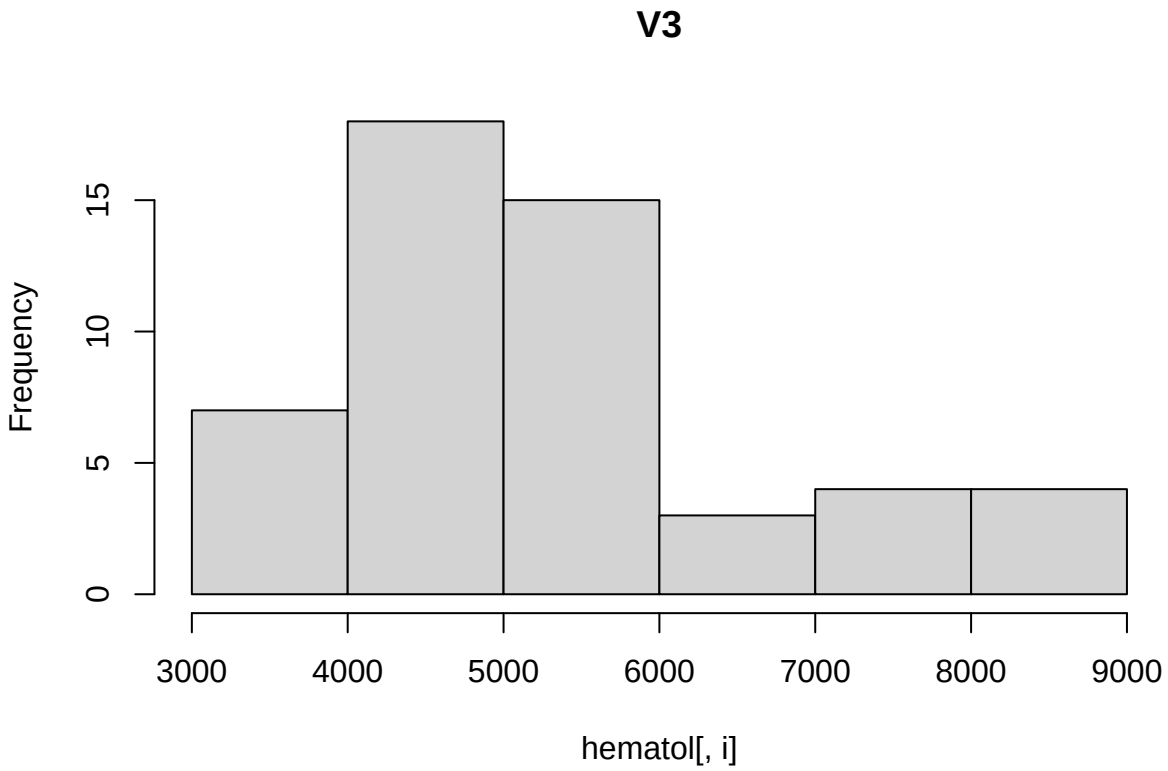


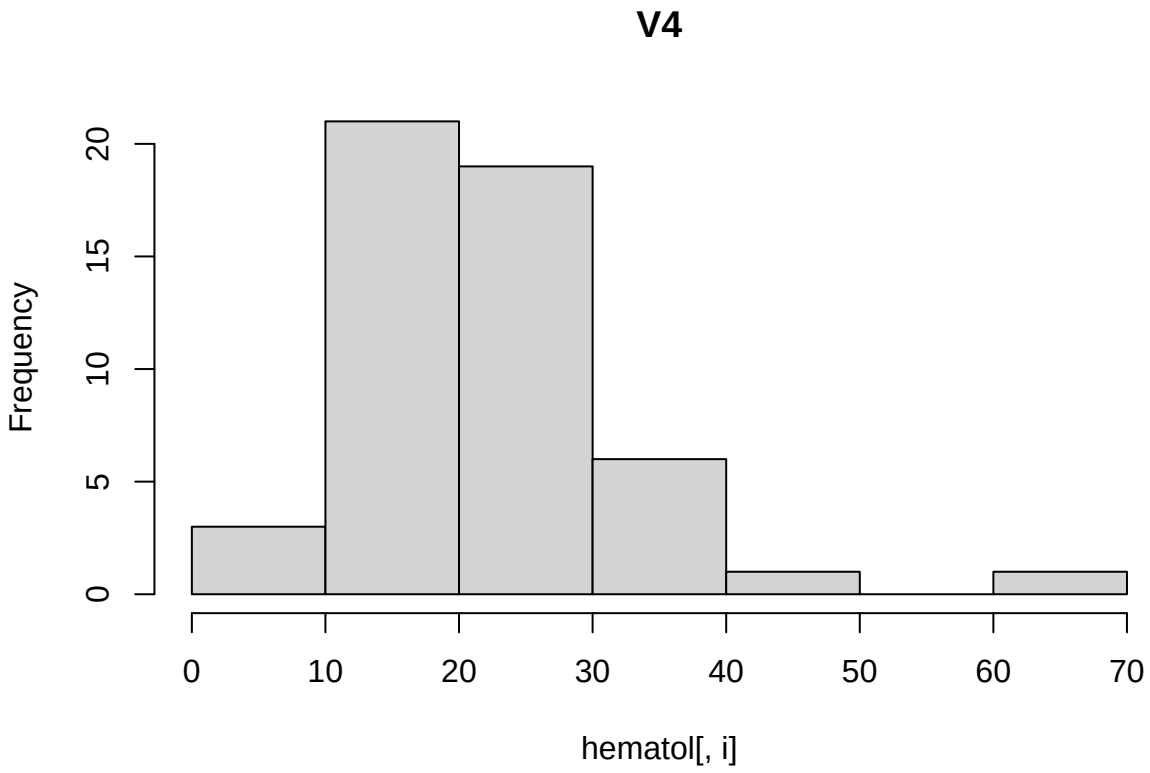
The Q-Q plots tell us that most of the variables are normal with a few deviations towards the right and left ends of the plots. The hematology variables that seem the more normal are "Hemoglobin Concentration" (V1), "Lymphocyte Count" (V4), and "Neutrophil Count" (V5).

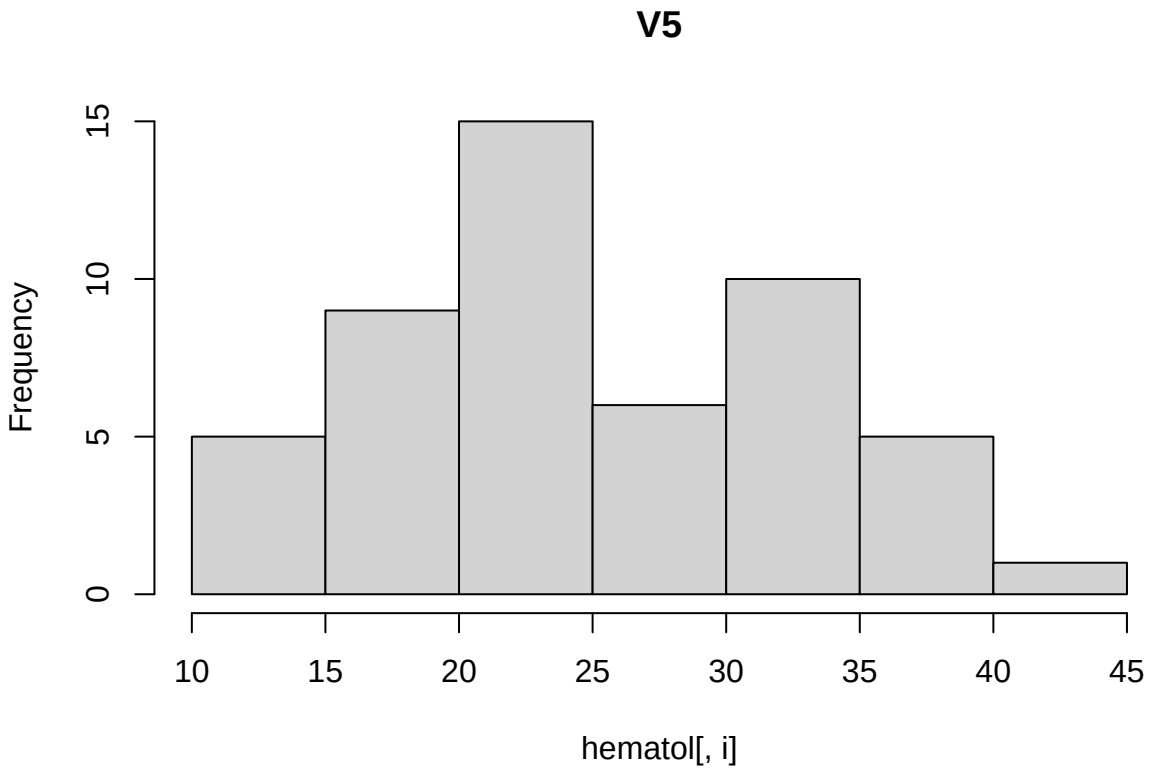
```
# Histograms
for (i in 1:6) {
  hist(hematol[,i],main=names(hematol)[i])
}
```

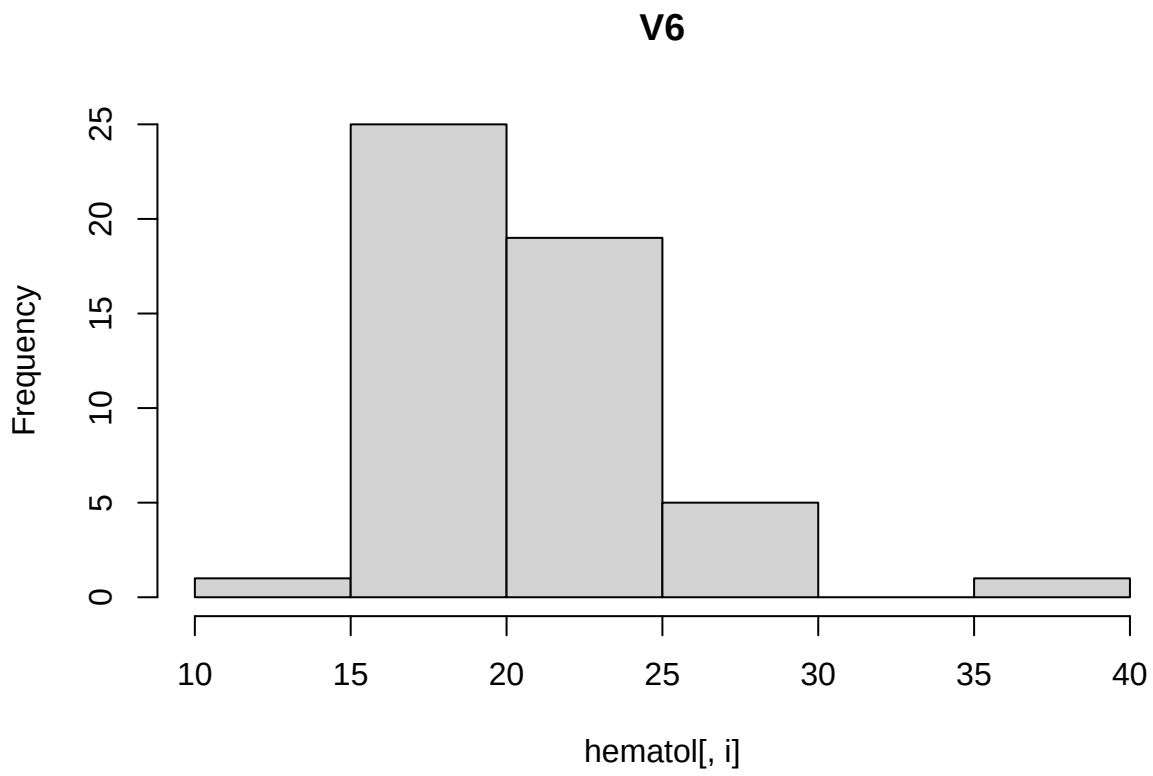

V1
Frequency
0
5
10
15
13
14
15
16
17
hematol[, i]


V2
Frequency
0
5
10
15
20
38
40
42
44
46
48
50
hematol[, i]


V3
Frequency
0
5
10
15
3000
4000
5000
6000
7000
8000
9000
hematol[, i]


V4
Frequency
0
5
10
15
20
0
10
20
30
40
50
60
70
hematol[, i]

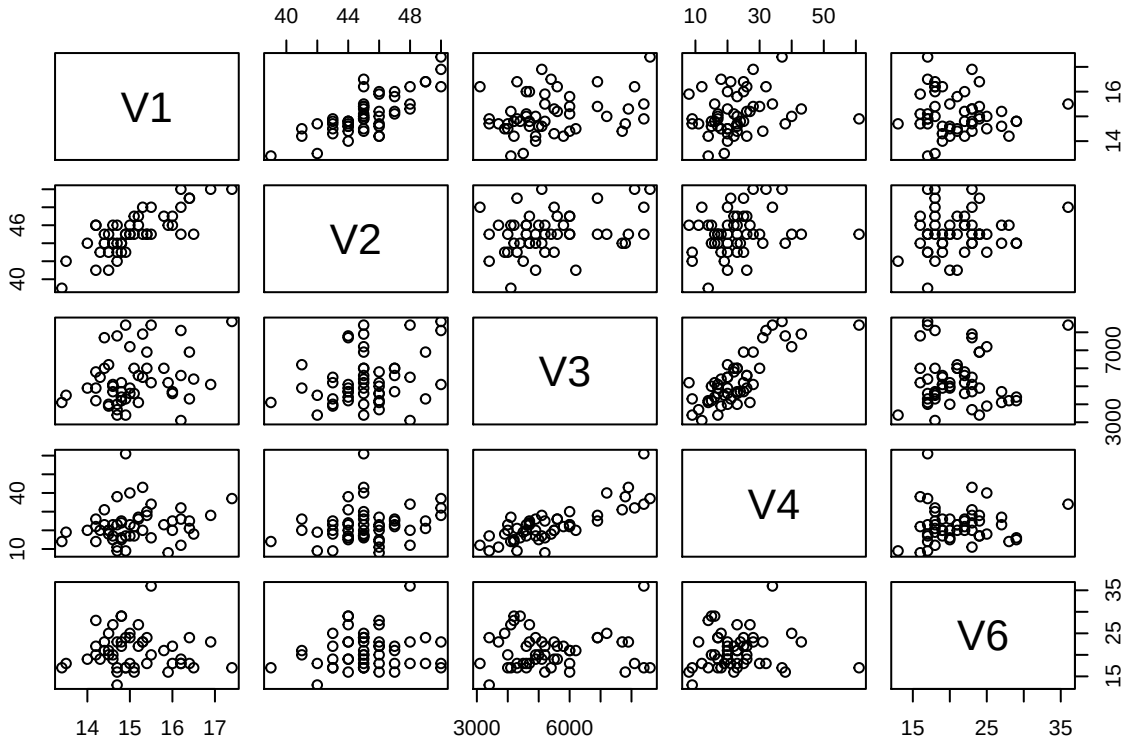

V5
Frequency
0
5
10
15
10
15
20
25
30
35
40
45
hematol[, i]




```
par(mfrow=c(1,1))
```

The Histogram plots show us that most of the variables are skewed either to the right or left. “Hemoglobin Concentration” (V1), “Packed Cell Volume” (V2), and “Neutrophil Count” (V5) appear to have a normal distribution.

```
# Pairwise scatter plots
pairs(hematol[,-5])
```



The pairwise scatter plot allows us to see the relationship between any two variables from the hematology dataset. From this, we can see that the relationship between "Hemoglobin Concentration" (V1) and "Packed Cell Volume" (V2), as well as "White Blood Cell Count" (V3) and "Lymphocyte Count" (V4) are positively linear. The other pairwise relationships seem to have little to no relationship.

```
# Chi Square plot
chisqplot <- function(data=hematol[,-5],percent=50,alpha=0.05)
{
  # Vector of the means
  xbar <- apply(data,2,mean)

  # Unbiased variance-covariance matrix & "deviation" vector
  S <- var(data)
  Sinv <- solve(S)
  ssize <- nrow(data)
  nvars <- ncol(data)
  xdel <- data - rep(1,ssize) %*% t(xbar)
  xdel <- as.matrix(xdel)

  cat("\nObs.","  ","Stat.distance\n")
  count <- 0
  sqd <- numeric(ssize)
  chsq <- numeric(ssize)

  # percentile point
  qcp <- qchisq(percent/100,nvars)
```

```
  for (i in 1:ssize)
  {
    # squared distance
    sd <- xdel[i, ] %*% Sinv %*% xdel[i, ]

    # flag obs. outside the contour
    cat("\n",i,ifelse(i<10," ",""),":  ",round(sd,3),ifelse(sd>qcp ," +",""))
    if ( sd<=qcp ) count <- count+1

    sqd[i] <- sd
    chsq[i] <- qchisq(1-(ssize-i+0.5)/ssize,nvars)
  }

  plot(chsq,sort(sqd))
  abline(0,1)   #add reference line

  cat("\nThe proportion of observations falling into the ",percent,"% prob. contour is:\n",sep="")
  cat("  ",round(count/ssize,3))
}

#Compare % of observations inside the contour to the corresponding chisq percentile
chisqplot(percent=50)
```

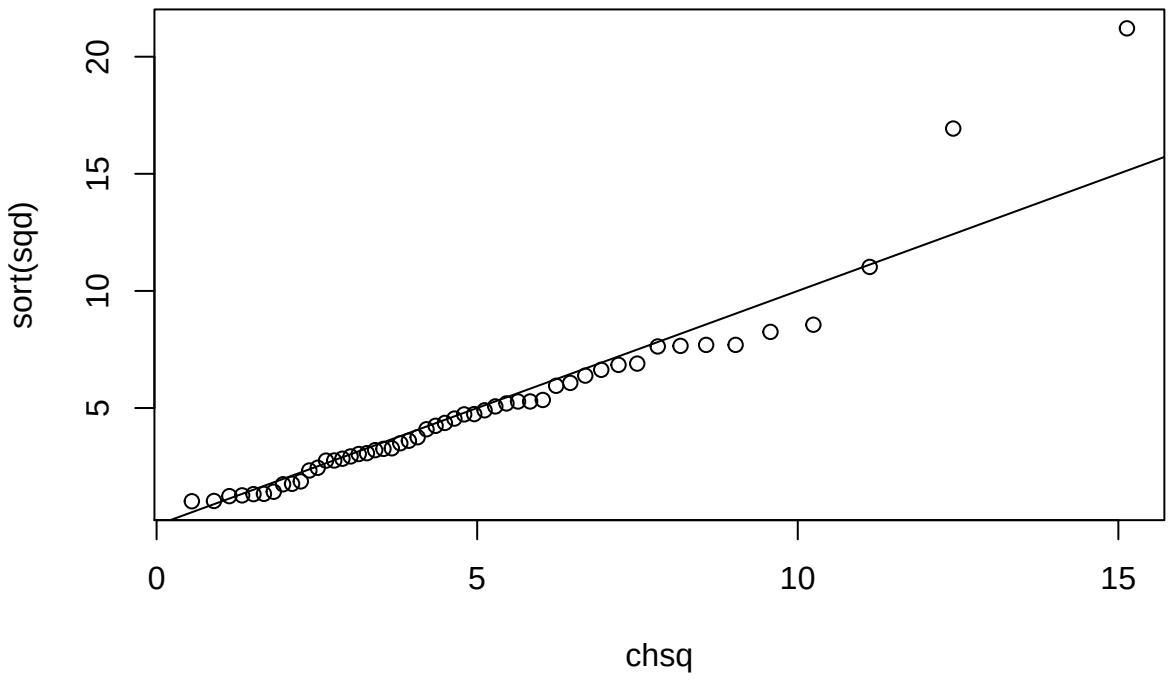
```
##
## Obs.    Stat.distance
##
##  1   :   7.696  +
##  2   :   3.603
##  3   :   4.728  +
##  4   :   2.449
##  5   :   1.239
##  6   :   3.202
##  7   :   5.949  +
##  8   :   5.068  +
##  9   :   6.632  +
##  10  :   16.932  +
##  11  :   1.336
##  12  :   7.653  +
##  13  :   1.324
##  14  :   4.902  +
##  15  :   3.754
##  16  :   6.384  +
##  17  :   8.251  +
##  18  :   2.759
##  19  :   1.271
##  20  :   6.898  +
##  21  :   7.695  +
##  22  :   3.281
##  23  :   11.025  +
##  24  :   1.869
##  25  :   5.28  +
##  26  :   4.093
##  27  :   1.764
```

```
##  28  :   3.037
##  29  :   2.334
##  30  :   2.934
##  31  :   5.278  +
##  32  :   3.256
##  33  :   1.032
##  34  :   5.196  +
##  35  :   3.495
##  36  :   1.744
##  37  :   1.425
##  38  :   8.557  +
##  39  :   1.02
##  40  :   4.74  +
##  41  :   2.754
##  42  :   3.072
##  43  :   7.629  +
##  44  :   4.363  +
##  45  :   4.241
##  46  :   5.345  +
##  47  :   21.211  +
##  48  :   6.843  +
##  49  :   2.835
##  50  :   4.549  +
##  51  :   6.073  +

##
## The proportion of observations falling into the 50% prob. contour is:
##    0.51
```

```
chisqplot(percent=75)
```

```
##
## Obs.    Stat.distance
##
##  1   :   7.696  +
##  2   :   3.603
##  3   :   4.728
##  4   :   2.449
##  5   :   1.239
##  6   :   3.202
##  7   :   5.949
##  8   :   5.068
##  9   :   6.632  +
##  10  :   16.932  +
##  11  :   1.336
##  12  :   7.653  +
##  13  :   1.324
##  14  :   4.902
##  15  :   3.754
##  16  :   6.384
##  17  :   8.251  +
##  18  :   2.759
##  19  :   1.271
```

```
##  20  :   6.898  +
##  21  :   7.695  +
##  22  :   3.281
##  23  :   11.025  +
##  24  :   1.869
##  25  :   5.28
##  26  :   4.093
##  27  :   1.764
##  28  :   3.037
##  29  :   2.334
##  30  :   2.934
##  31  :   5.278
##  32  :   3.256
##  33  :   1.032
##  34  :   5.196
##  35  :   3.495
##  36  :   1.744
##  37  :   1.425
##  38  :   8.557  +
##  39  :   1.02
##  40  :   4.74
##  41  :   2.754
##  42  :   3.072
##  43  :   7.629  +
##  44  :   4.363
##  45  :   4.241
##  46  :   5.345
##  47  :   21.211  +
##  48  :   6.843  +
##  49  :   2.835
##  50  :   4.549
##  51  :   6.073
```




```
##
## The proportion of observations falling into the 75% prob. contour is:
##    0.765
```

```
chisqplot(data=hematol[,-5],percent=80)
```

```
##
## Obs.    Stat.distance
##
##  1   :   7.696  +
##  2   :   3.603
##  3   :   4.728
##  4   :   2.449
##  5   :   1.239
##  6   :   3.202
##  7   :   5.949
##  8   :   5.068
##  9   :   6.632
##  10  :   16.932  +
##  11  :   1.336
##  12  :   7.653  +
##  13  :   1.324
##  14  :   4.902
##  15  :   3.754
##  16  :   6.384
```

```
##  17  :   8.251  +
##  18  :   2.759
##  19  :   1.271
##  20  :   6.898
##  21  :   7.695  +
##  22  :   3.281
##  23  :   11.025  +
##  24  :   1.869
##  25  :   5.28
##  26  :   4.093
##  27  :   1.764
##  28  :   3.037
##  29  :   2.334
##  30  :   2.934
##  31  :   5.278
##  32  :   3.256
##  33  :   1.032
##  34  :   5.196
##  35  :   3.495
##  36  :   1.744
##  37  :   1.425
##  38  :   8.557  +
##  39  :   1.02
##  40  :   4.74
##  41  :   2.754
##  42  :   3.072
##  43  :   7.629  +
##  44  :   4.363
##  45  :   4.241
##  46  :   5.345
##  47  :   21.211  +
##  48  :   6.843
##  49  :   2.835
##  50  :   4.549
##  51  :   6.073

##
## The proportion of observations falling into the 80% prob. contour is:
##    0.824
```

The Chi-Square plot shows us the proportion of observations falling into the 50%, 75% and 80% probability contour are 0.51, 0.765, and 0.824 respectively. All three contour values are close to the percentile value. This tells us that the observed counts and the counts we expect are the same.

```
# Formal test for normality (Shapiro-Wilks)
shapiro.test(hematol$V1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hematol$V1
## W = 0.96373, p-value = 0.1203
```

```
shapiro.test(hematol$V2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hematol$V2
## W = 0.96548, p-value = 0.1427
```

```
shapiro.test(hematol$V3)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hematol$V3
## W = 0.92316, p-value = 0.002748
```

```
shapiro.test(hematol$V4)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hematol$V4
## W = 0.90139, p-value = 0.0004682
```

```
shapiro.test(hematol$V5)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hematol$V5
## W = 0.97136, p-value = 0.2516
```

```
shapiro.test(hematol$V6)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  hematol$V6
## W = 0.93174, p-value = 0.005807
```

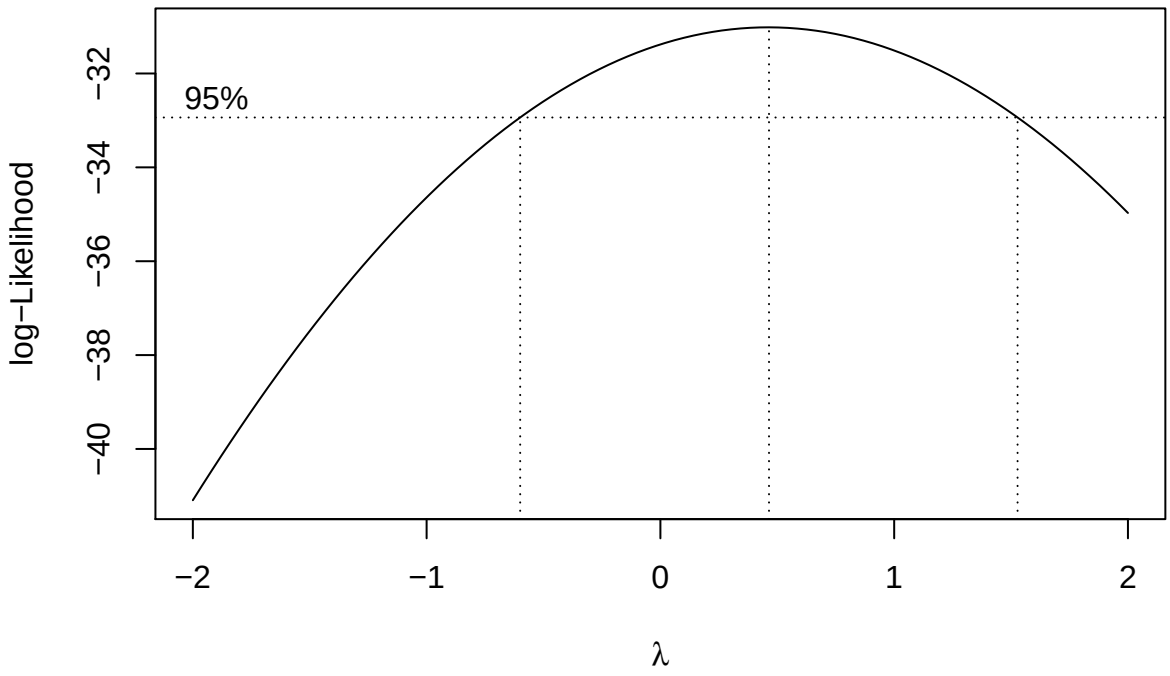
The null hypothesis for the Shapiro-Wilk test is that the data is normally distributed. Variables "White Blood Cell Count" (V3), "Lymphocyte Count" (V4), and "Serum Lead Concentration" (V6) all have a p-value less than the alpha level 0.05. This means that we reject the null hypothesis that those three variables are normally distributed.

In conclusion, based upon the results of our various plots and tests, Variables "Hemoglobin Concentration" (V1), "Packed Cell Volume" (V2), and "Neutrophil Count" (V5) are the most normally distributed variables among the hematology dataset. However, of these variables, the least skewed is "Neutrophil Count" (V5), with a p-value of 0.2516.

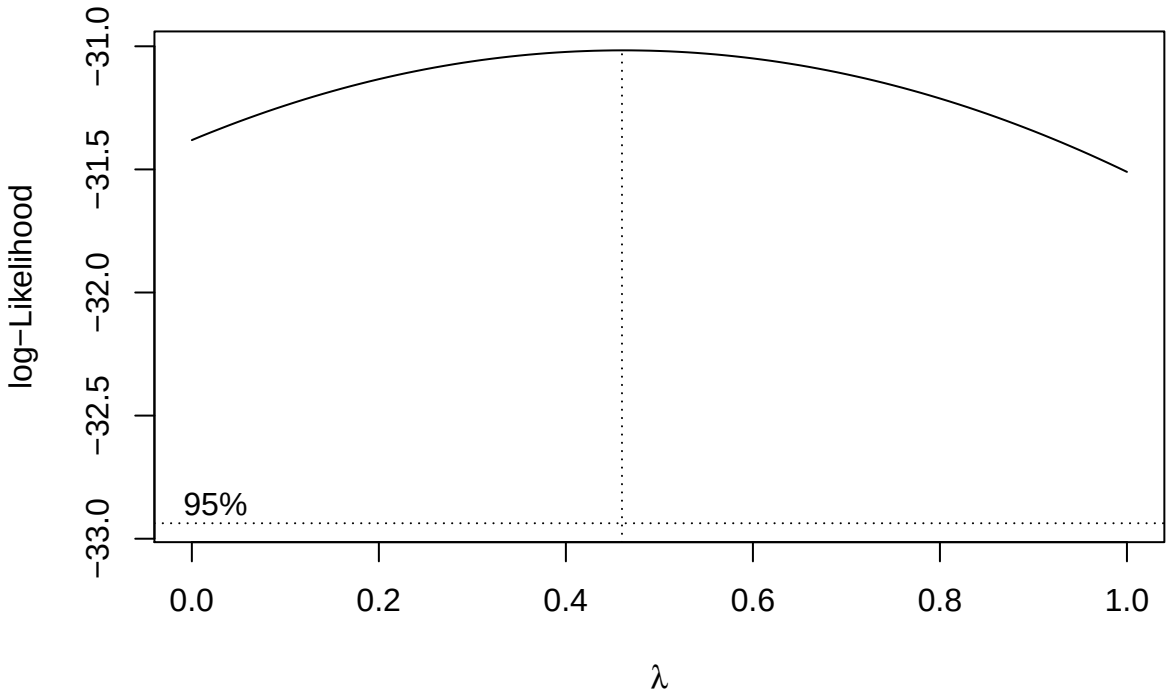
4.25 Use the glucose data in Table 3.9.

(a) Use the methods in Section 4.5.1 to find the optimal univariate transformation to normality for each of the glucose measurements obtained one hour after sugar intake (x1, x2, and x3).

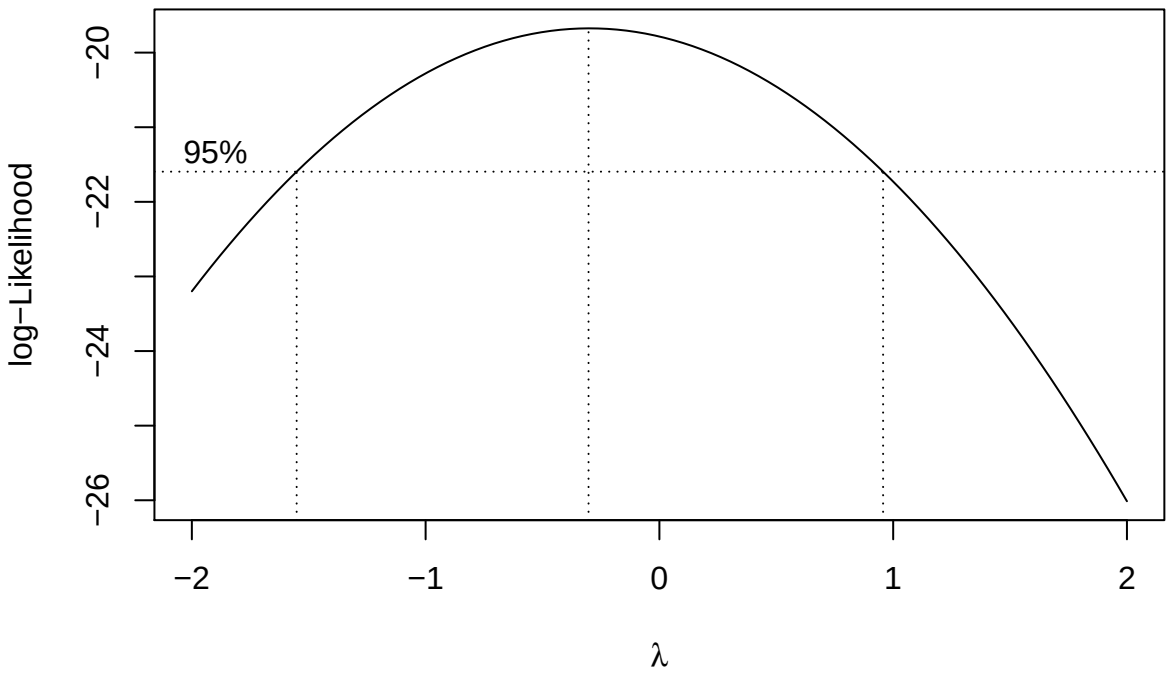
```
# Univariate Transform to normality with Box-Cox

# Get glucose table
t39<- read.table("T3_9_GLUCOSE.dat")
g <- as.data.frame(t39)
glucose <- g[,4:6]
names(glucose) <- c("x1", "x2", "x3")

library(MASS)

# Box-Cox Tranformations
boxcox(lm(x1 ~ 1, data=glucose))
```



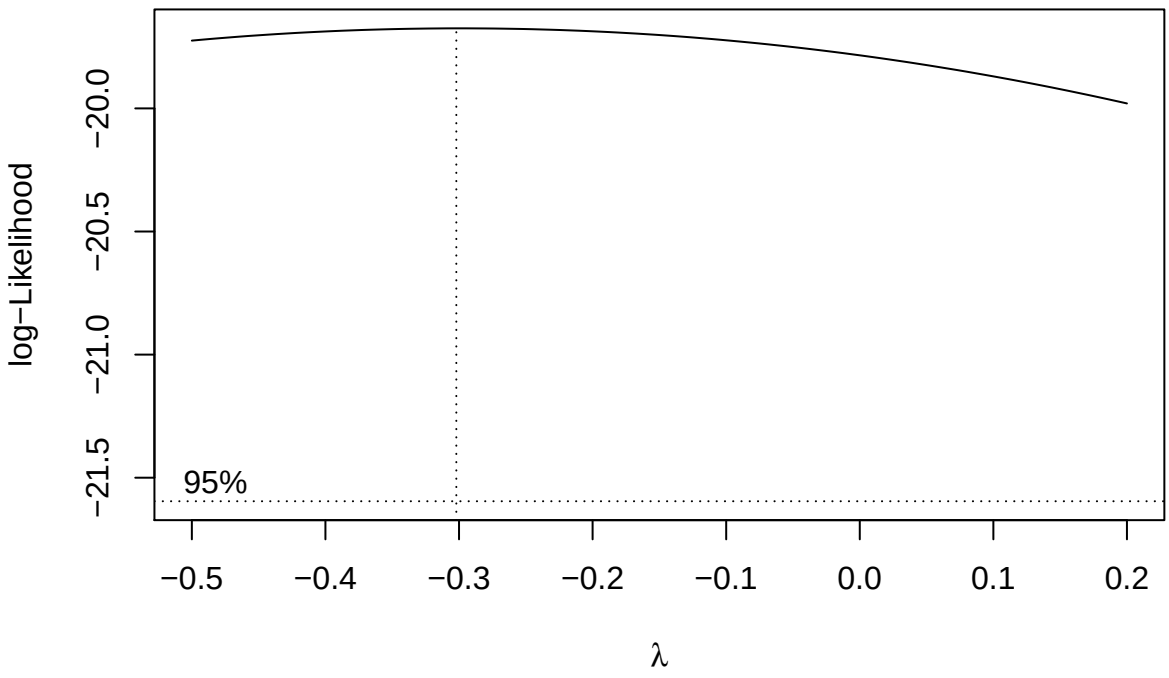

```
boxcox(lm(x1 ~ 1, data=glucose), lambda=seq(0,1,1/100))
```



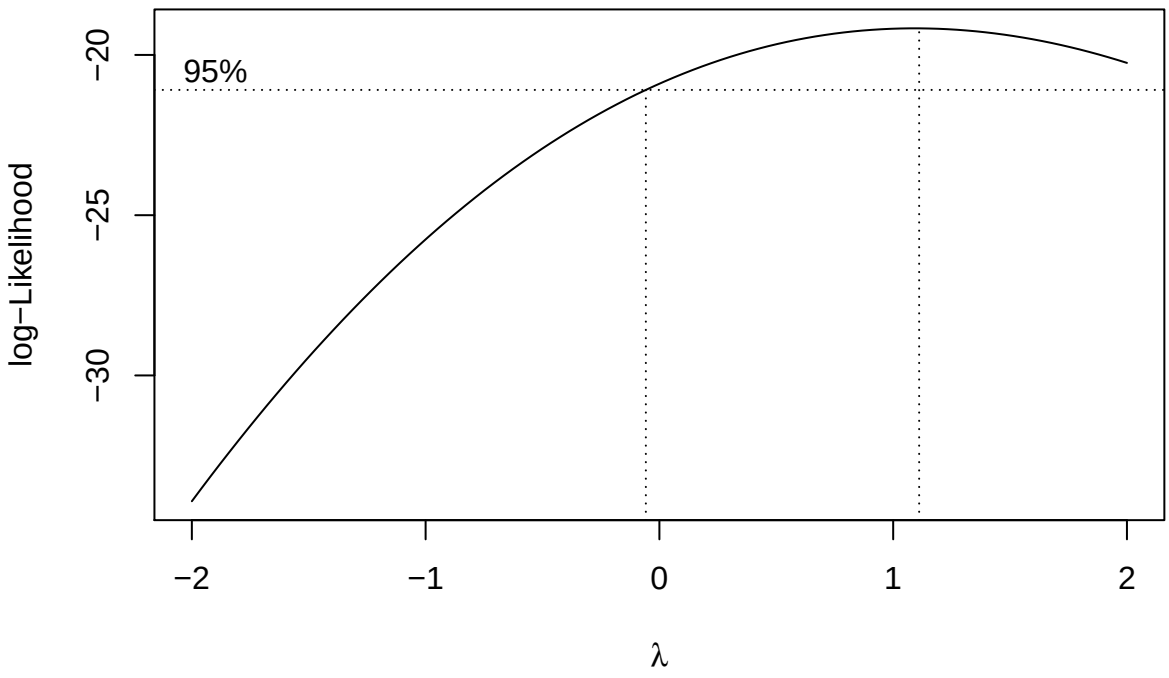

```
# lamda = 0.5

boxcox(lm(x2 ~ 1, data=glucose))
```



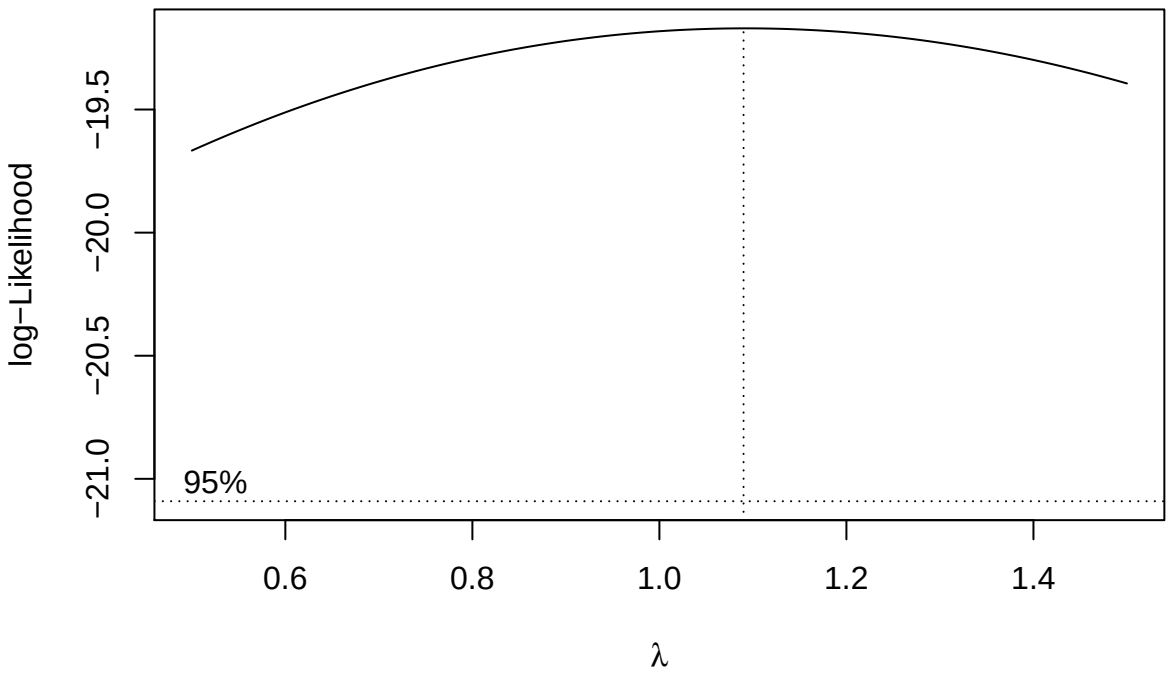

```
boxcox(lm(x2 ~ 1, data=glucose), lambda=seq(-.5,.2,1/100))
```




```
# lamda = -0.3

boxcox(lm(x3 ~ 1, data=glucose))
```




```
boxcox(lm(x3 ~ 1, data=glucose), lambda=seq(.5,1.5,1/100))
```




```
# lamda = 1.1
```

The Box-Cox univariate tranformations of x_1, x_2, and x_3 gives us λ of 0.5, -0.3, and 1.1 respectively.

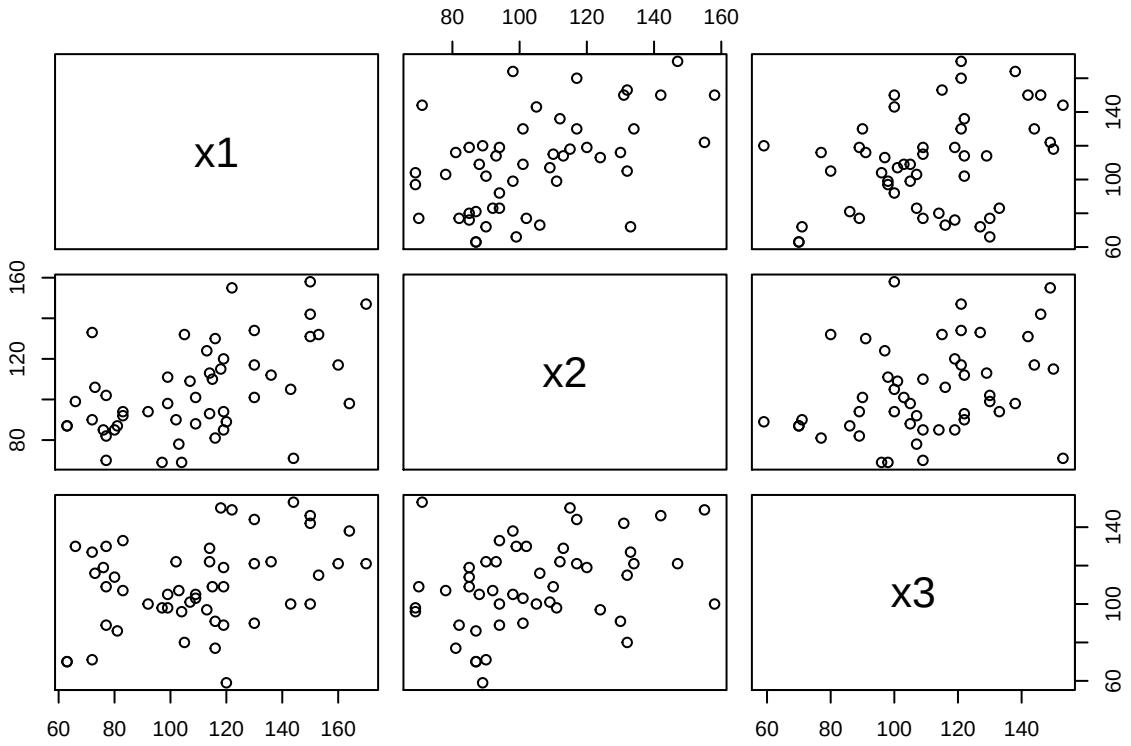
(b) Use the methods in Section 4.5.2 to find the optimal multivariate transformation to 3-variate normality for the glucose measurements obtained one hour after sugar intake (x1, x2, and x3).

```
# Multivariate Transform to 3-variate normality

library(carData)
library(car)

(pt <- powerTransform(glucose))
```

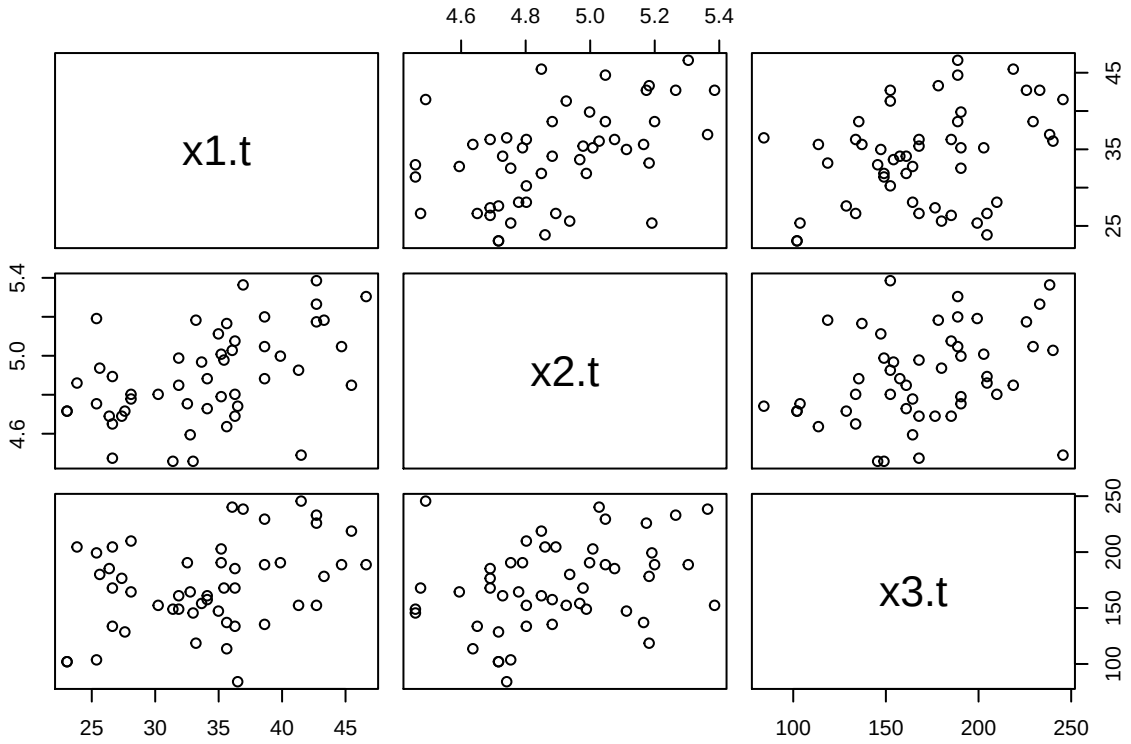
```
## Estimated transformation parameters
##         x1         x2         x3
## 0.67870695 0.02417438 1.11672502
```

```
glucose.t <- data.frame(glucose, x1.t = (glucose$x1^pt$lambda[1]-1)/pt$lambda[1])
glucose.t <- data.frame(glucose.t, x2.t = (glucose$x2^pt$lambda[2]-1)/pt$lambda[2])
glucose.t <- data.frame(glucose.t, x3.t = (glucose$x3^pt$lambda[3]-1)/pt$lambda[3])
```

```r
# Now lets take a look at the pairs plot before and after the transformation. Only viewing the new vari
pairs(glucose[,1:3])
```




```r
pairs(glucose.t[,4:6])
```



The multivariate power tranformations of x_1, x_2, and x_3 gives us λ of 0.7, 0, and 1.1 respectively.

(c) How do the transformations obtained from the two approaches compare?

```
# Using lamda to tranform the data

# Univariate Tranformations
a = 0.5
glucose$x1_uni = (glucose$x1)^a
b = -0.3
glucose$x2_uni = (glucose$x2)^b
c = 1.1
glucose$x3_uni = (glucose$x3)^c

# Multivariate Tranformations
aaa = 0.7
glucose$x1_multi = (glucose$x1)^aaa
bbb = 0
glucose$x2_multi = (glucose$x2)^bbb
ccc = 1.1
glucose$x3_multi = (glucose$x1)^ccc

head(glucose, 10)
```

```
##     x1  x2  x3    x1_uni    x2_uni   x3_uni x1_multi x2_multi x3_multi
```

```
## 1   97  69  98  9.848858 0.2807665 155.0061 24.58896        1 153.2671
## 2  103  78 107 10.148892 0.2706273 170.7348 25.64402        1 163.7272
## 3   66  99 130  8.124038 0.2519471 211.5133 18.77936        1 100.3456
## 4   80  85 114  8.944272 0.2637390 183.0608 21.48637        1 123.9935
## 5  116 130  91 10.770330 0.2321758 142.8715 27.86895        1 186.5966
## 6  109 101 103 10.440307 0.2504399 163.7272 26.68078        1 174.2485
## 7   77 102 130  8.774964 0.2497008 211.5133 20.91912        1 118.8885
## 8  115 110 109 10.723805 0.2441081 174.2485 27.70056        1 184.8279
## 9   76  85 119  8.717798 0.2637390 191.9118 20.72858        1 117.1912
## 10  72 133 127  8.485281 0.2305921 206.1504 19.95872        1 110.4246
```

Both univarite and multivarite transformations have the absolute difference in λ values of 0.2, 0.3, and 0 for x_1, x_2, and x_3 respectively. Looking at a glimpse of the new glucose dataset, we can see that after both transformations, the values for each variable are closer in range with one another than they were before. This shows that their distribution is more normal.