ST 437/537: Applied Multivariate and Longitudinal Data Analysis

Multivariate Normal, Assessment of Normality, and Outlier Detection

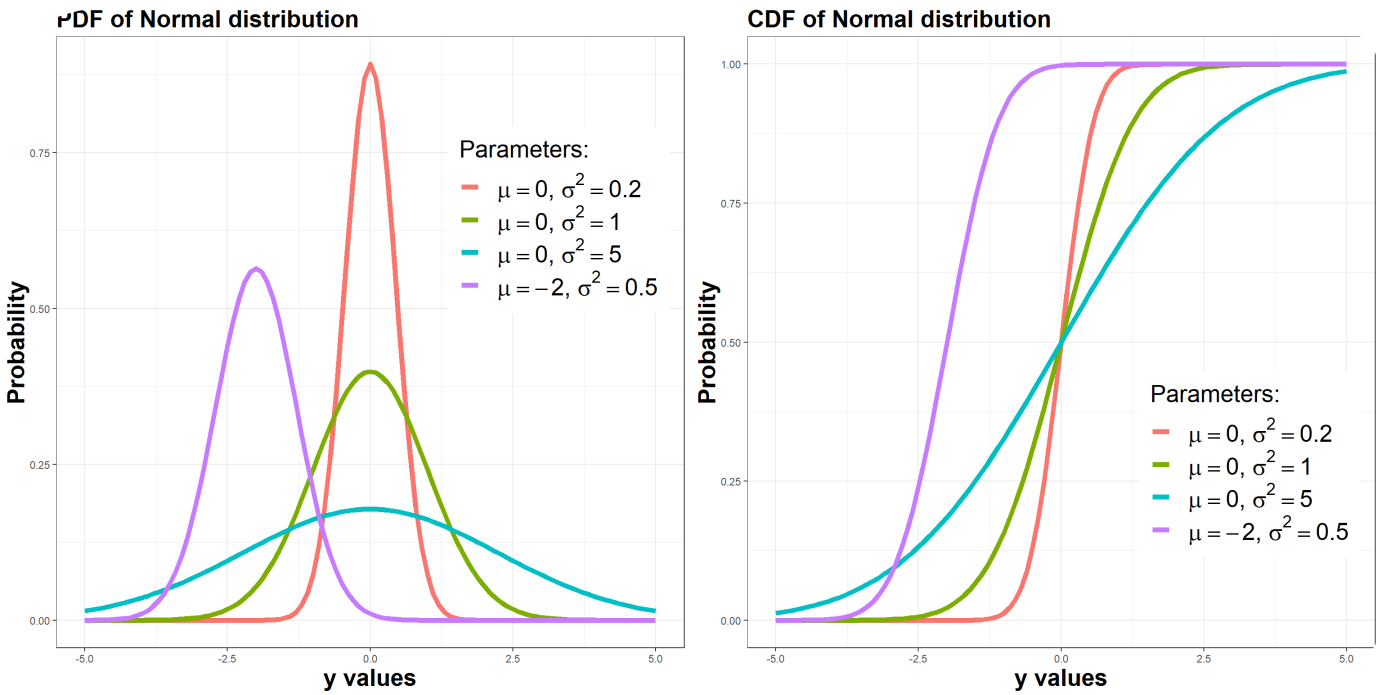
Arnab Maity
NCSU Department of Statistics
SAS Hall 5240 919-515-1937 amaity[at]ncsu.edu

Review of univariate normal distribution

Suppose we are modeling a continuous random variable (scalar) Y with a bell shaped pdf with mean μ and variance σ^2. We say that Y follows a normal distribution, that is, $Y \sim N(\mu, \sigma^2)$, if the pdf is

$$f_Y(y; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\{-\frac{1}{2\sigma^2}(y-\mu)^2\}, \qquad -\infty < y < \infty.$$

Let us now visualize the normal distribution $N(\mu, \sigma^2)$ for different values of μ, σ^2.



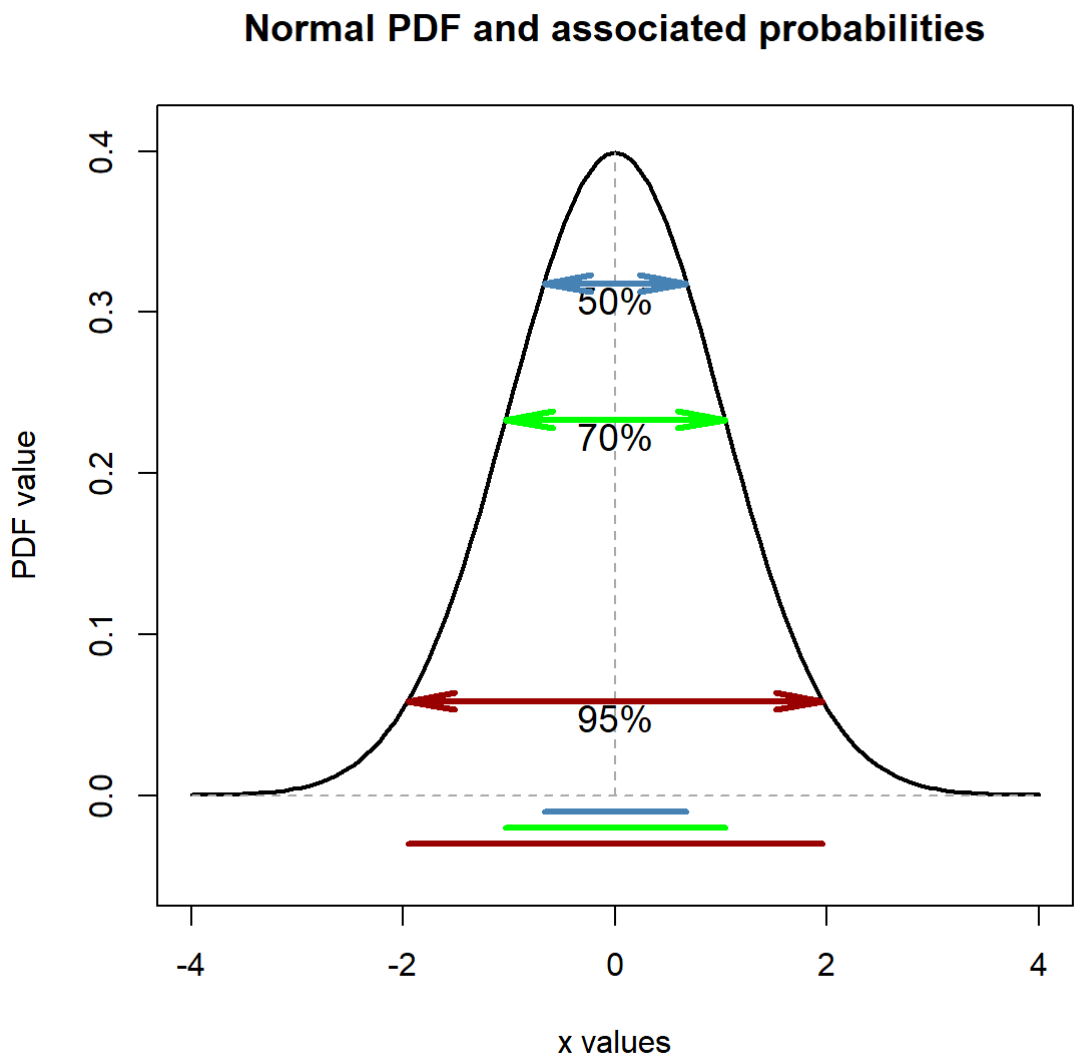
Notes:

- If $Y \sim N(0, \sigma^2)$ then $E(Y) = \mu$ and $Var(Y) = \sigma^2$.

- **Standard Normal Distribution** is $Z \sim N(0,1)$. Any normal distribution can be standardized using $Z = \frac{Y-\mu}{\sigma}$.
- The function $\phi(\cdot)$ is often used to denote the *pdf of the standard normal*.
- The function $\Phi(\cdot)$ is often used to denote the *cdf of the standard normal* (no closed form)

$$\Phi(z) = P(Z \leq z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt$$

- Any normal distribution can be created from a standard normal distribution using $Y = \mu + Z\sigma$. Specifically, if $Z \sim N(0,1)$ then $\mu + Z\sigma \sim N(\mu, \sigma^2)$.
- Each interval has a associated probality:



R functions: R requires that you specify the mean and standard deviation, rather than mean and variance.

The R functions related to normal distribution are

a. **PDF**: dnorm(x, mean, sd, log = FALSE)
b. **CDF**: pnorm(q, mean, sd, lower.tail = TRUE, log.p = FALSE)
c. **Quantiles**: qnorm(p, mean, sd, lower.tail = TRUE, log.p = FALSE)
d. **Random number**: rnorm(n, mean, sd)

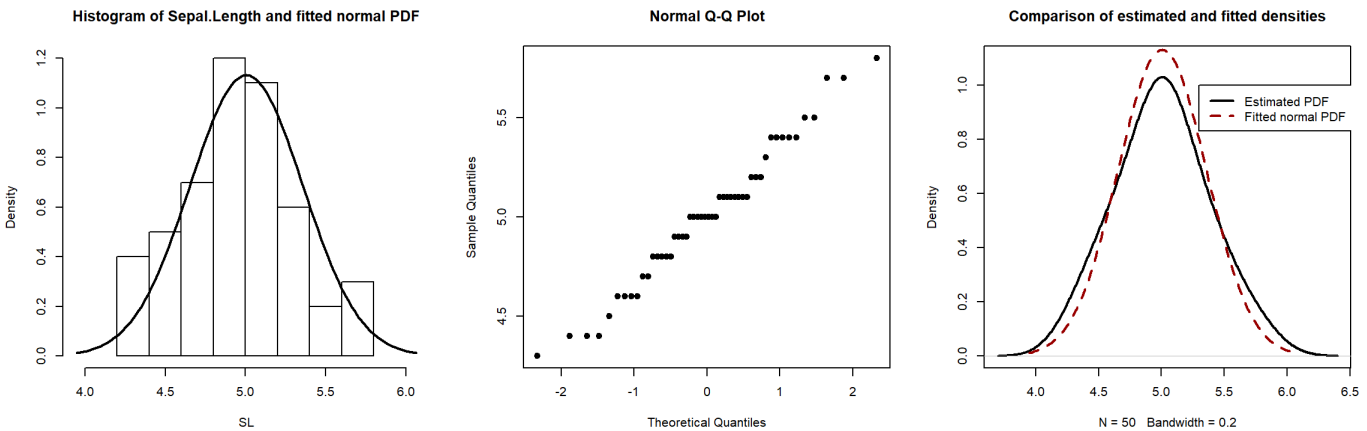
Here the arguments are:

- `x , q` : the value at which to compute the probability PMF of CDF
- `mean` : mean μ
- `sd` : standard deviation σ
- `p` : probability, it must be between 0 and 1
- `n` : the number of times to repeat the experiment.
- `log , log.p` : logical; if TRUE, probabilities p are given as $log(p)$.
- `lower.tail` : logical; if TRUE (default), probabilities are $P[Y \leq x]$, otherwise, $P[Y > x]$.

Let us consider the sepal length of setosa flowers. We will create the following plots.

- Relative frequency histogram and overlay with the PDF of a $N(\mu, \sigma^2)$ distribution, where μ and σ^2 are replaced by $\bar{x}$ and s^2 (left panel)
- Normal Q-Q plot, where sorted data, $x_{(1)} \leq \ldots \leq x_{(n)}$, are plotted against normal quantiles, $\Phi^{-1}\{(1-0.5)/n\}, \ldots, \Phi^{-1}\{(n-0.5)/n\}$ (middle panel)
- Estimated density function from the data overlayed with the PDF of a $N(\mu, \sigma^2)$ distribution, where μ and σ^2 are replaced by $\bar{x}$ and s^2 (right panel)

```
# Extract the species
species <- iris$Species

# Only take the setosa flowers
setosa <- iris[species == "setosa", 1:4]

# Extract only sepal.length (the first column)
SL <- setosa[, 1]

# sample size
n <- length(SL)

# sample mean and sd
xbar <- mean(SL)
s <- sd(SL)


# Create a 1x3 plotting window
par(mfrow = c(1,3))

# Plot a histogram and
# Overlay a normal distribution
xx <- seq(xbar - 3*s, xbar + 3*s, len=101)
yy <- dnorm(xx,  mean = mean(SL), sd = sd(SL))

hist(SL, probability = T, xlim = range(xx),
     main = "Histogram of Sepal.Length and fitted normal PDF")
lines(xx, yy, lwd=2)

# Q-Q plot to assess normality
qqnorm(SL, pch=19)

# Estimated density vs, normal pdf
den.est <- density(SL, bw = 0.2)
plot(den.est, lwd=2, col="black", ylim = c(0, 1.1),
     main = "Comparison of estimated and fitted densities")
lines(xx, yy, lwd=2, lty=2, col="#990000")
legend(x = 5.3, y = 1,
       legend = c("Estimated PDF", "Fitted normal PDF"),
       col = c("black", "#990000"), lwd = 2, lty = 1:2)
```



Assessing univariate normality: We can use graphical (e.g., Q-Q plot) as well as hypothesis testing techniques.

Many formal statistical tests have been developed. See the article **[Yap and Sim (2011). Comparisons of various types of normality tests] (https://www.tandfonline.com/doi/full/10.1080/00949655.2010.520163)** for a numerical comparison between various tests.

- Shapiro-Wilks test, and Shapiro–Francia test: the later test is a simplification of the former; they show similar power to each other. These two are among the more powerful normality tests.
- Kolmogorov-Smirnov (K-S) test, and Lilliefors corrected K-S test: the later test is usually preferred among the two tests.
- Cramer von Mises test, and Anderson-Darling test (a modification of the CVM test): based on weighted difference between the empirical and theoretical CDFs.
- Person's Chi-squared test: a goodness-of-fit test, not highly recommended for continuous distributions.
- Jarque-Bera test and D'Agostino-Pearson omnibus tests: moment based tests.

Overall, Shapiro-Wilk test shows a robust performance against a wide variety of alternatives.

```
shapiro.test(SL)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  SL
## W = 0.9777, p-value = 0.4595
```

Since the p-value is large (e.g., larger than 5%), we can say that normality assumption for the data is justified.

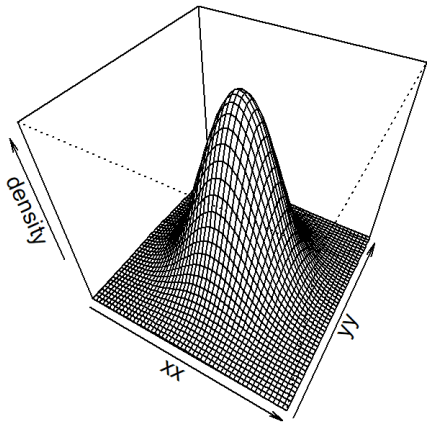
The **[nortest] (https://cran.r-project.org/web/packages/nortest/index.html)** package implements a few of the tests mentioned above.

Bivariate and Multivariate normal distributions

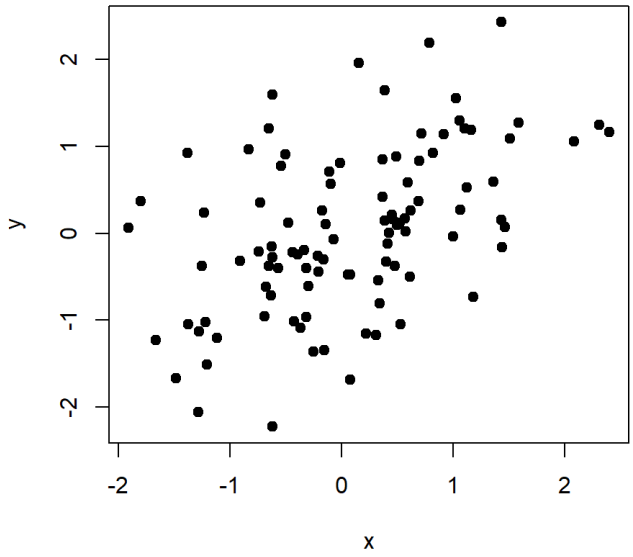
The random vector $\boldsymbol{X}_{2\times 1} = (X_1, X_2)^T$ follows a bivariate normal (Gaussian) distribution with mean vector $\boldsymbol{\mu} = (\mu_1, \mu_2)^T$ and variance-covariance (positive definite) matrix $\boldsymbol{\Sigma}$ and denoted as $\boldsymbol{X} \sim N_2(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ if its probability density function is

$$f(\boldsymbol{x}) = (2\pi)^{-1} |\boldsymbol{\Sigma}|^{-1/2} \exp\{-(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}-\boldsymbol{\mu})/2\}.$$

PDF of a bivariate normal distribution



A random sample of size 100



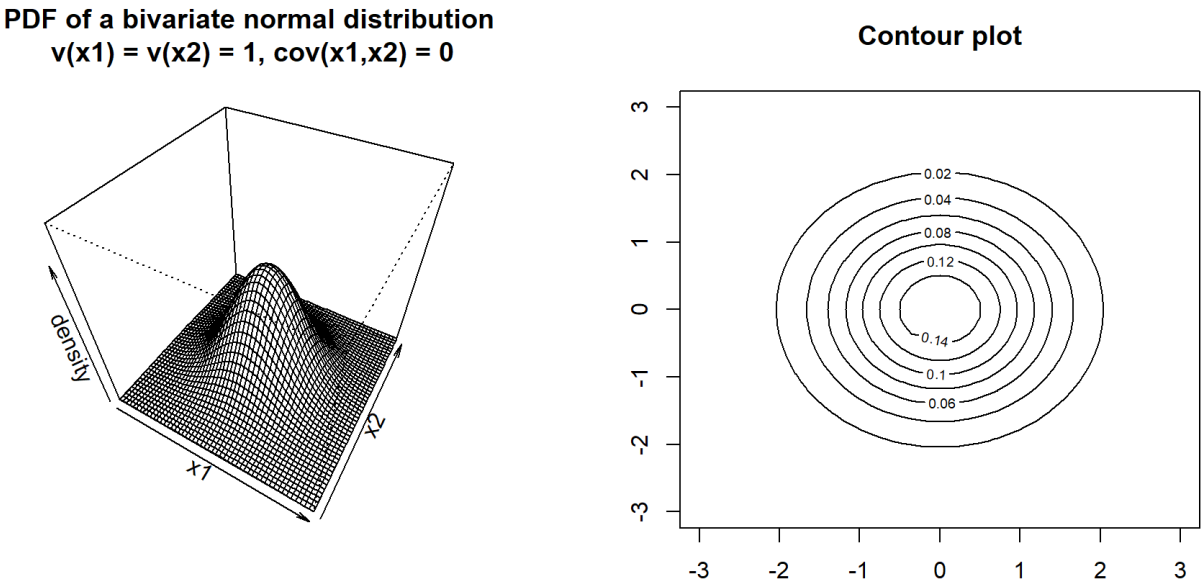
The shape of the PDF (and that of the scatterplot of a random sample generated from the distribution) is determined by the variance-covariance matrix of $\boldsymbol{X}$. An easy was to visualize the PDF of a bivariate distribution is to plot the constant probability density contours.

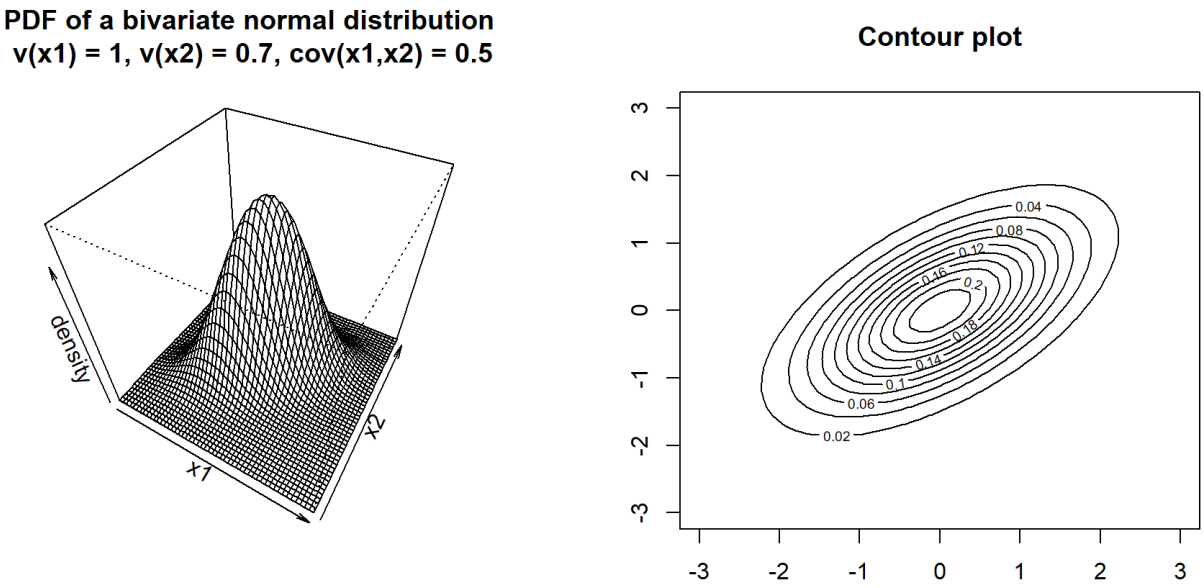
Constant probability density contours

We define the constant probability density contour (also called constant-density contour) of a bivariate normal PDF to be the set of vectors $\boldsymbol{x}$ such that $f(\boldsymbol{x})$ is constant, that is,

$$\{\boldsymbol{x} : (\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}) = c\}$$

for a specific c. These sets are ellipses that are centered around $\boldsymbol{\mu}$, and the major and minor axes are $c\sqrt{\lambda_i}\boldsymbol{e}_i$, where λ_i are the eigenvalues and $\boldsymbol{e}_i$ are the corresponding eigenvectors of $\boldsymbol{\Sigma}$.





More generally, a random vector $\boldsymbol{X} = (X_1, \ldots, X_p)^T$ is said for follow a multivariate normal distribution $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu}$ is a $p \times 1$ vector and $\boldsymbol{\Sigma}$ is positive definite matrix, if the PDF of $\boldsymbol{X}$ is

$$f(\boldsymbol{x}) = (2\pi)^{-p/2} |\boldsymbol{\Sigma}|^{-1/2} \exp\{-(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})/2\}.$$

We can show that $E(\boldsymbol{X}) = \boldsymbol{\mu}$ and that $cov(\boldsymbol{X}) = \boldsymbol{\Sigma}$.

Some properties of the multivariate normal distribution:

- The constant probability density contours are ellipsoids

- Zero covariance implies the compoments of $\boldsymbol{X}$ are independent (**ONLY** when $\boldsymbol{X}$ is multivariate normal)
- When $\boldsymbol{\mu} = \mathbf{0}_p$ and $\boldsymbol{\Sigma} = \boldsymbol{I}_p$, we say that we have a **standard multivariate normal distribution**, $\boldsymbol{Z} \sim N_p(0_p, I_p)$.
- All subsets of $\boldsymbol{X}$ also follow multivariate normal distribution.
- If $\boldsymbol{X}$ follows a multivariate normal distribution, then any linear combination of $\boldsymbol{X}$ follow multivariate normal distribution. Specifically, if $\boldsymbol{X} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then $\boldsymbol{AX} \sim N_q(\boldsymbol{A\mu}, \boldsymbol{A\Sigma A}^T)$ for any matrix $\boldsymbol{A}$.
- $(\boldsymbol{X} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{X} - \boldsymbol{\mu}) \sim \chi^2_p$, where χ^2_p is the chi-square distribution with p degrees of freedom.

Mahalanobis distance

The quantity

$$d^2 = (\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})$$

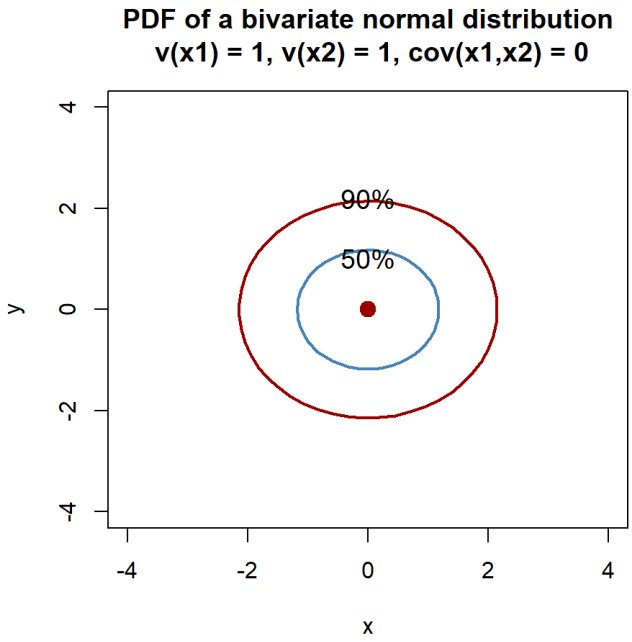
is called the Mahalanobis squared distance between $\boldsymbol{x}$ and $\boldsymbol{\mu}$.

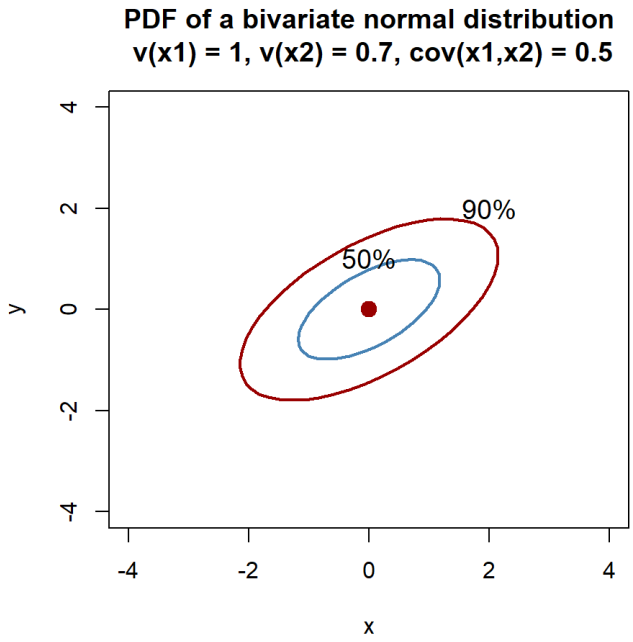
Using the last property, we can compute the probability observing data within any constant-density contours. Specifically, consider the constant-density ellipse $E_c = \{\boldsymbol{x} : (\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) = c\}, c > 0$. Then

$$Pr(\boldsymbol{X} \in E_c) = G_p(c),$$

where $G_p(c)$ is the CDF of a χ^2_p distribution.

We show 50% and 90% contours below for two bivariate normal distributions.





Sampling distribution of $\bar{X}$ and S

Recall that, for univariate normal distribution, if $X_1, \ldots, X_n$ form a random sample from $N(\mu, \sigma^2)$, then

$$\bar{X} \sim N(\mu, \sigma^2/n), \text{ and } \frac{n-1}{\sigma^2} S^2 \sim \chi^2_{n-1},$$

where S^2 is the sample variance. We also know that

$$\bar{X} \text{ and } S^2 \text{ are independent.}$$

We have similar results for multivariare normal distribution.

Exact distribution of $\bar{X}$ and S

Suppose $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ form a random sample from a $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ distribution. Then

- $\bar{\boldsymbol{X}}$ has a $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}/n)$ distribution.
- $(n-1)\boldsymbol{S}$ has a **Wishart** distribution with $n-1$ degrees of freedom (a generalization of χ^2 distribution).
- $\bar{\boldsymbol{X}}$ and $\boldsymbol{S}$ are independent.

Large sample results analogous to univariate normal also exist. Recall that, if $X_1, \ldots, X_n$ form a random sample from $N(\mu, \sigma^2)$, then Central Limit Theorem says

$$\bar{X} \text{ approximately has a } N(\mu, \sigma^2/n) \text{ distribution.}$$

Similar results hold for multivariate normal distribution.

Large sample results

Suppose $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ form a random sample from a population (can be different from normal) with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. When the sample size n is large,

- $\bar{\boldsymbol{X}}$ has an *approximate* $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}/n)$ distribution (multivariate CLT).
- $(\boldsymbol{X} - \boldsymbol{\mu})^T \boldsymbol{S}^{-1} (\boldsymbol{X} - \boldsymbol{\mu})$ has an *approximate* χ^2_p distribution (also need $n - p$ large; note that we replaced $\boldsymbol{\Sigma}$ with $\boldsymbol{S}$).

Evaluating multivariate normality

Many of the techniques typically used in multivariate statistics assume that the parent distribution is multivariate normal or that the sample size sufficiently large (in which case the normality assumption is less crucial). However, the quality of the inferences relies on how close the parent distribution is to the multivariate normal. Thus it is essential to validate the normality assumption. Nevertheless, it is difficult to assess multivariate normality; in practice, we investigate the univariate and bivariate distributions and determine how close they are to normality assumptions.

Check univariate normality

Usual univariate analysis for each variable, such as normal Q-Q plot and statistical tests for normality can be done.

Recall, if $\boldsymbol{X}$ is multivariate normal, then each component is univariate normal as well.If we reject normality for one of the variables, then $\boldsymbol{X}$ can not be multivariate normal.

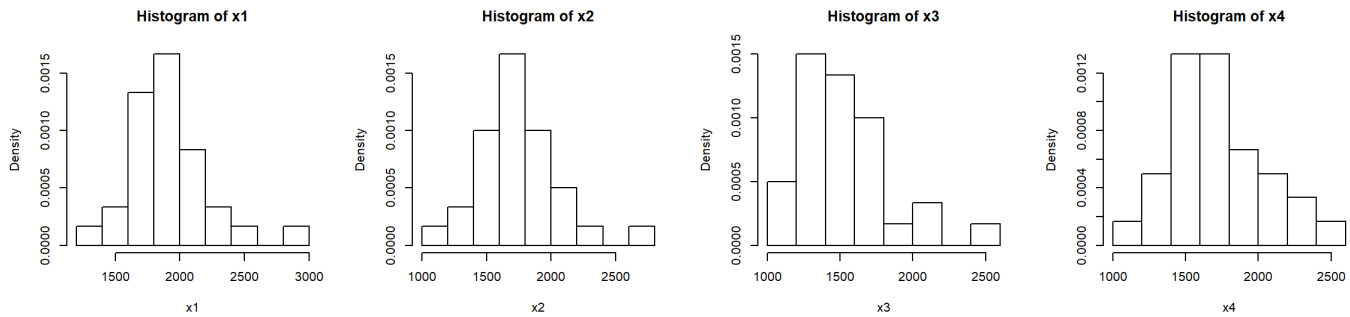
Let us consider the dataset **[Table 4.3 in Johnson and Wichern (2007). Applied Multivariate Analysis.] (data/T4-3.DAT)** where four measures of stiffness $x_1, \ldots, x_4$ are measured of each of the $n = 30$ boards.

```
# Reading the data set
dat <- read.table("data/T4-3.DAT", header = F)
colnames(dat) <- c("x1", "x2", "x3", "x4", "d2")

n <- nrow(dat)
p <- ncol(dat) - 1

# snapshot
head(dat)
```

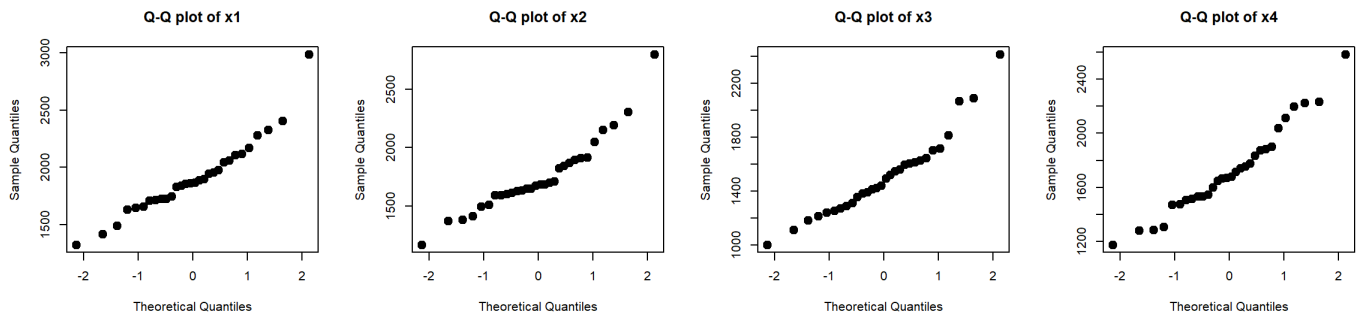
```
##     x1   x2   x3   x4   d2
## 1 1889 1651 1561 1778 0.60
## 2 2403 2048 2087 2197 5.48
## 3 2119 1700 1815 2222 7.62
## 4 1645 1627 1110 1533 5.21
## 5 1976 1916 1614 1883 1.40
## 6 1712 1712 1439 1546 2.22
```

The first four columns provide the four maesured variables. Let us construct their relative frequency histograms and normal Q-Q plots.

```
par(mfrow = c(1,4))
for(ii in 1:4){
  hist(dat[, ii], probability = T,
       xlab = paste("x", ii, sep=""),
       main = paste0("Histogram of x", ii))
}
```




```
par(mfrow = c(1,4))
for(ii in 1:4){
  qqnorm(dat[, ii],
       main = paste0("Q-Q plot of x", ii), pch=19, cex=1.5)
}
```



These marginal distributions appear somewhat close to normal. Applying the Shapiro-Wilks test to each of the variables reveals a little more.

```
# Shapiro-Wilks test for each column using the apply function
# See ?apply for more details
apply(dat[,1:4], 2, shapiro.test)
```

```
## $x1
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.93068, p-value = 0.05118
##
##
## $x2
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.91274, p-value = 0.01746
##
##
## $x3
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.93258, p-value = 0.05751
##
##
## $x4
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.96127, p-value = 0.3337
```

It seems that X_2 may be violating the normality assumption while X_1 and X_3 has small p-values.

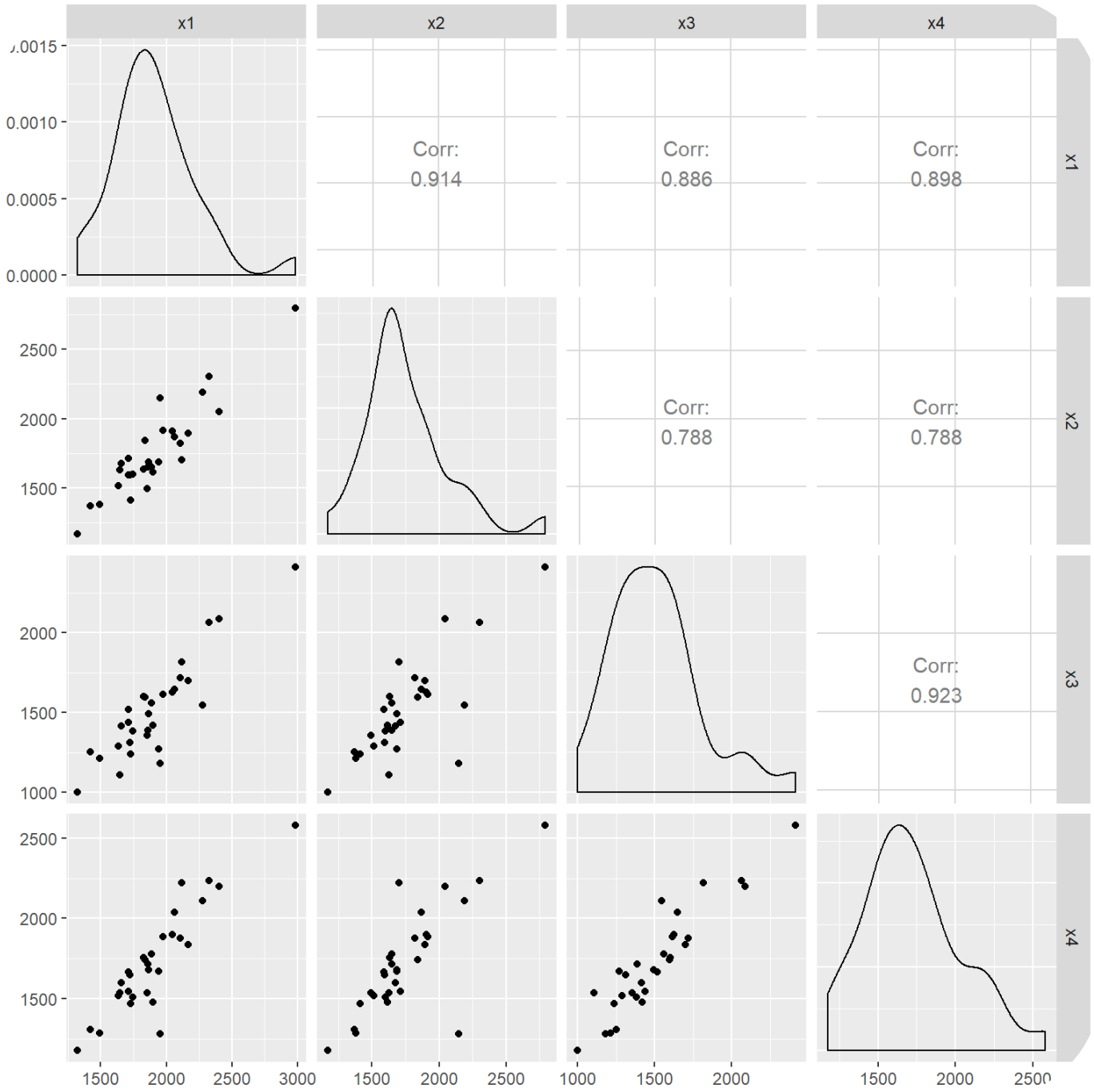
In general, just checking univariate plots is not enough. This is because even if individual variables are normally distributed, their joint distribution may not be normal.

Check scatterplots

If the data indeed are generated from normal, the constant-density contours must be ellipses. Thus, the scatterplots should also confrom to this structure.

Plotting scatterplots and pairs-plot of the data will also reveal any unusual shape (or outliers) in the data set. Overlaying “data ellipses” (constant-density contours estimated from the data assuming normality) on top of scatterplots are useful in this situation.

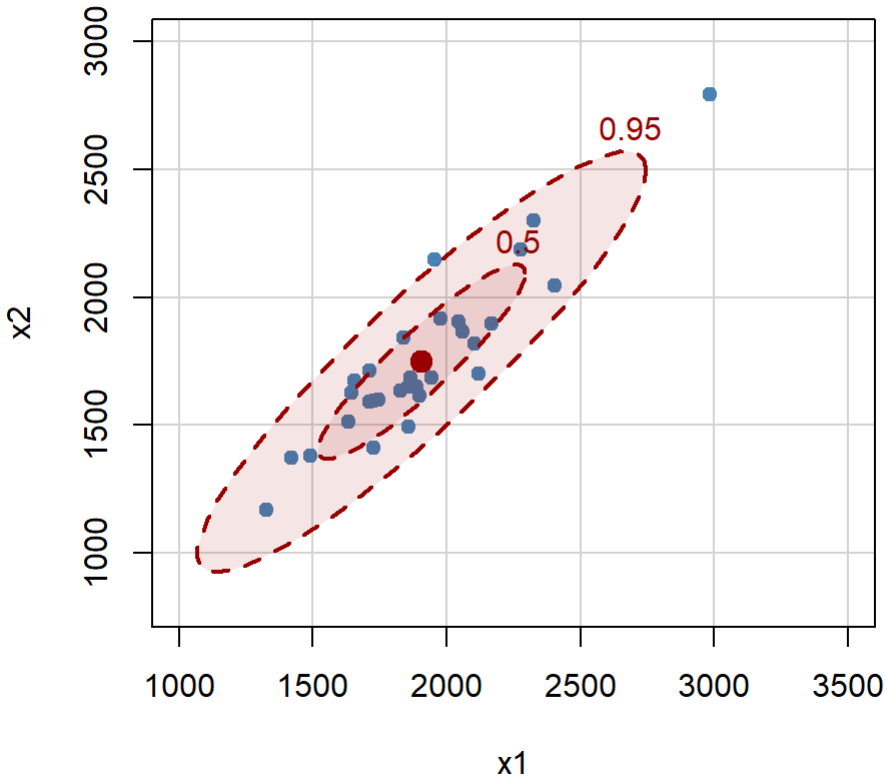
```
library(GGally)
ggpairs(dat[,1:4])
```



The diagonals show the density estimates for each variables.

The data ellipses can be drwan using the `dataEllipse` function in the `car` package.

```
library(car)

x1 <- dat[, 1]
x2 <- dat[, 2]

dataEllipse(x1, x2,
            xlim = c(1000, 3500), ylim = c(800, 3000),
            pch=19, col = c("steelblue", "#990000"), lty=2,
            ellipse.label=c(0.5, 0.95), levels = c(0.5, 0.95),
            fill=TRUE, fill.alpha=0.1)
```



By default, the 50% and 95% ellipses are drawn. See the documentation using `?dataEllipse` for more customization options.

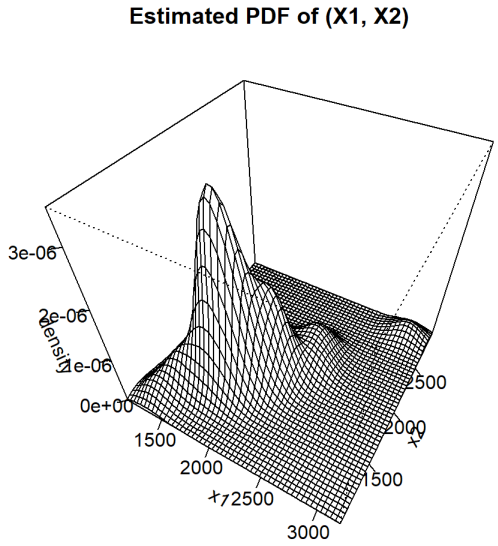
We can see the data cloud does have a elliptical shape. However, there is one point that might be an outlier.

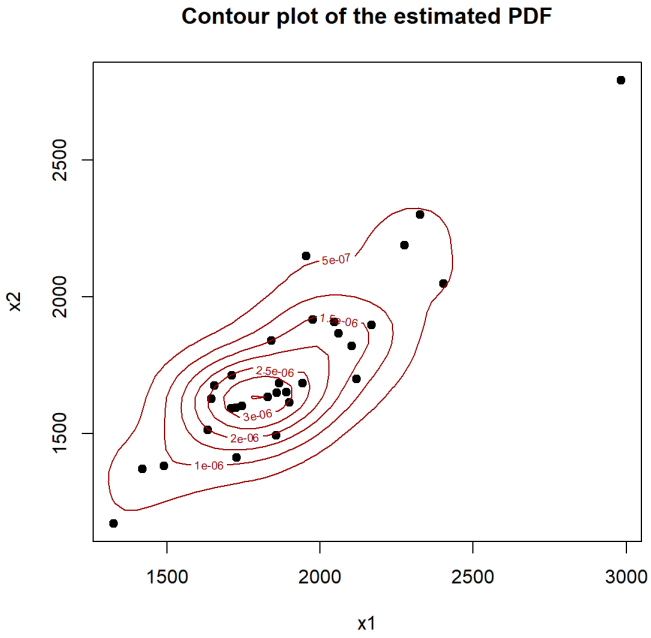
We can also estimate the PDF of each pair of variables. This can be done using the `bkde2D()` function in the `KernSmooth` package. Visualization can be done using `persp()` function in base R.

```
library("KernSmooth")
```

```
## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009
```

```
# Estimate bivariate density
den.est <- bkde2D(dat[, 1:2], bandwidth = apply(dat[, 1:2], 2, dpik))

# Plot the density
par(mfrow = c(1,2))
persp(x = den.est$x1, y = den.est$x2,
      z = den.est$fhat,
      xlab = "x1", ylab = "x2", zlab = "density",
      phi = 45, theta = 30, ticktype = "detailed",
      main = "Estimated PDF of (X1, X2)")

# Create a contour plot of the estimated density
plot(dat[, 1:2], xlab = "x1", ylab = "x2", pch=19,
     main = "Contour plot of the estimated PDF")
contour(x = den.est$x1, y = den.est$x2,
            z = den.est$fhat, add = TRUE, col="#990000")
```





Construct a chi-square plot

Given sample data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, the chi-square plot is constructed using the following steps:

- For each i, compute the Mahalanobis squared distance

$$d_i^2 = (\boldsymbol{x}_i - \bar{\boldsymbol{x}})^T \boldsymbol{s}^{-1} (\boldsymbol{x}_i - \bar{\boldsymbol{x}}),$$

  where $\bar{\boldsymbol{x}}$ and $\widehat{\boldsymbol{s}}$ are observed values of the sample mean and covariance matrix, respectively.

- If the data are indeed generted from a normal distribution, then the d_i^2 values should follow a χ_p^2 (here $p = 4$) distribution. Thus, we plot the ordered d_i^2 values,

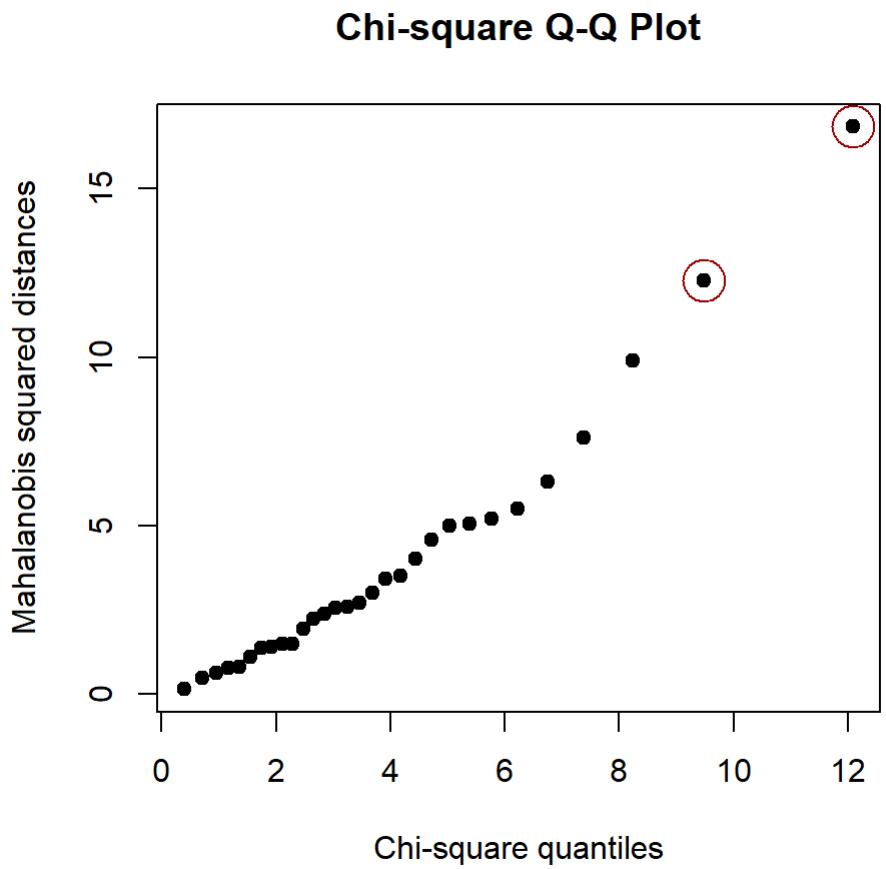
  $$d_{(1)}^2 \leq \ldots \leq d_n^2$$

  versus the theoretical quantiles of the χ_p^2 distribution

  $$q_p(\frac{1-0.5}{n}), \ldots, q_p(\frac{n-0.5}{n}).$$

If the multivariate normality assumption is correct, then the points should follow a straight line. A systematic curved pattern will suggest a departure from normality. One or two points that show large deviations from the linear trend might be outliers and would warrant further investigation.

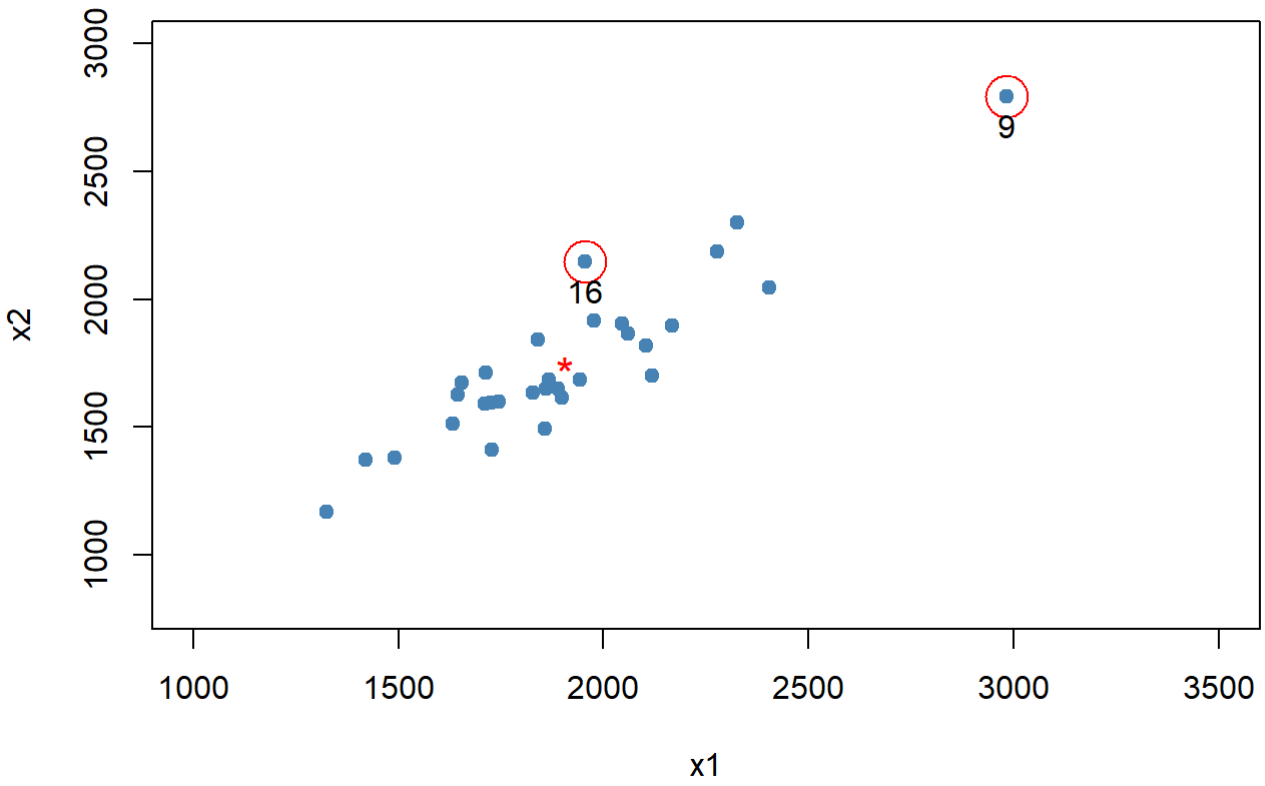
```
# A function to create a chi-square plot
chisquare.plot <- function(x, mark){
  # x: a n x p data matrix
  # mark: number of extreem points to mark

  # number of variables
  p <- ncol(x)
  # sample size
  n <- nrow(x)

  # xbar and s
  xbar <- colMeans(x)
  s <- cov(x)

  # Mahalanobis dist
  x.cen <- scale(x, center = T, scale = F)
  d2 <- diag( x.cen %*% solve(s) %*% t(x.cen) )

  # chi-sq quantiles
  qchi <- qchisq((1:n - 0.5)/n, df = p)

  # sorted d^2 value
  sortd <- sort(d2)

  # plot
  plot(qchi, sortd, pch=19, xlab = "Chi-square quantiles", ylab = "Mahalanobis square
d distances", main = "Chi-square Q-Q Plot")

  # Mark the top three points with heighest distance values
  points(qchi[(n-mark+1):n], sortd[(n-mark+1):n], cex=3, col="#990000")
}

# Call the function and draw the chi-square plot; mark
# two points with highest distance
chisquare.plot(x = dat[, 1:4], mark = 2)
```



Let us look at the top few distance values along with the z-scores for each variable.

x1	x2	x3	x4	z1	z2	z3	z4	d2	ID
1954	2149	1180	1281	0.15	1.25	-1.09	-1.38	**16.85**	16
2983	2794	2412	2581	**3.31**	**3.28**	**2.98**	**2.65**	**12.26**	9
2276	2189	1547	2111	1.14	1.38	0.12	1.2	9.9	21
2119	1700	1815	2222	0.66	-0.16	1.01	1.54	7.62	3
2326	2301	2065	2234	1.29	1.73	1.83	1.58	6.28	29
2403	2048	2087	2197	1.53	0.94	1.91	1.46	5.48	2

```
plot(x1, x2,
            xlim = c(1000, 3500), ylim = c(800, 3000),
            pch=19, col = c("steelblue"))
points(x = x1[c(9, 16)], y = x2[c(9, 16)], col="red", cex=3)
text(x = x1[c(9, 16)], y = x2[c(9, 16)], labels = c(9, 16), pos=1)
points(mean(x1), mean(x2), pch = "*", cex=1.4, col="red")
```



It seems that observations 16 and 9 are outliers. Observation 9 is easy to notice since it is visible in scatterplots as well. However, observation 16 is hidden within the data cloud and is only visible in the chi-square plot.

Johnson and Wichern (2007) recommend that once we find an outlier, we must try to access the real specimens and re-examine them whenever possible to determine the reason behind the unusual observations.

Perform formal tests for multivariate normality

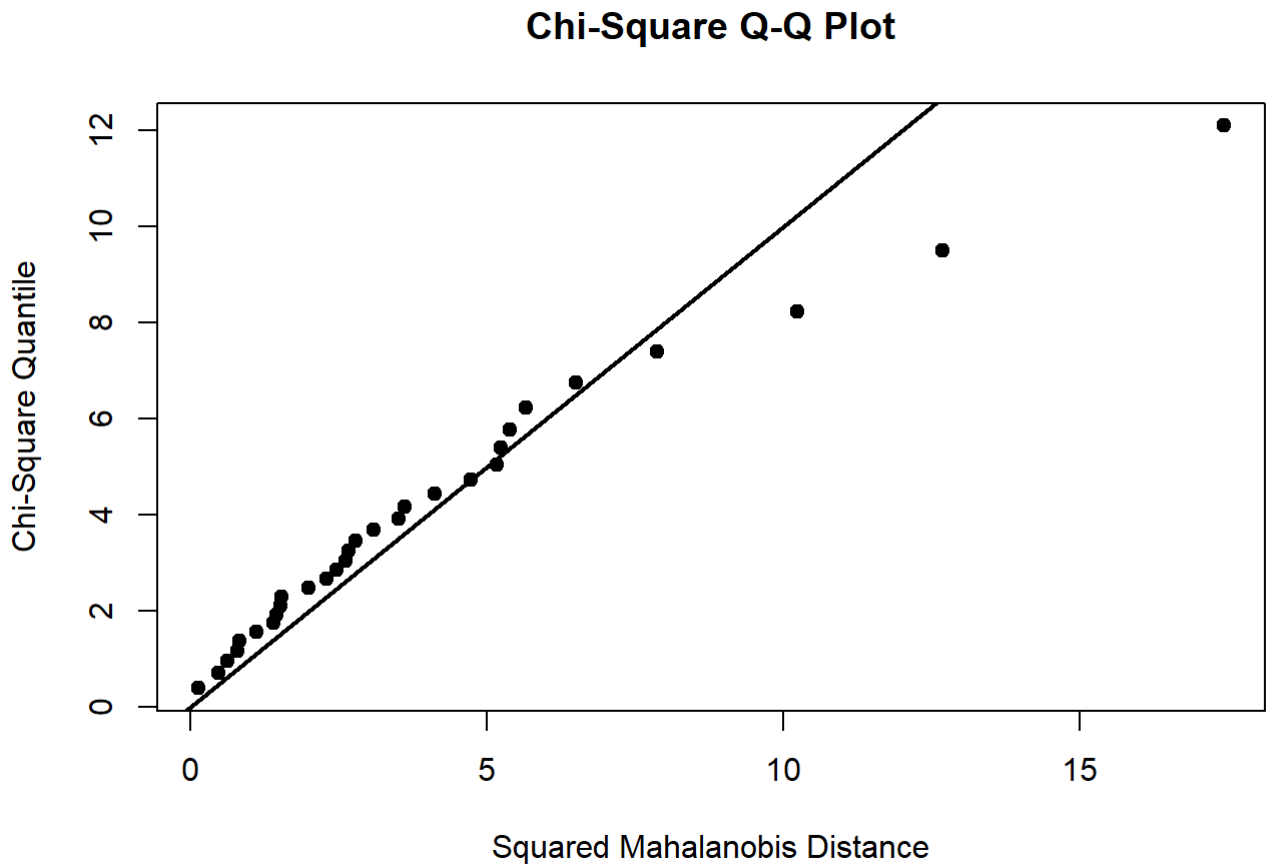
Several tests for assessing multivariate normality are available:

- Mardia's Skewness test
- Mardia's Kurtosis test
- Henze-Zirkler test
- Royston test
- Doornik-Hansen test
- Szekely – Rizzo's Energy test
- Singh's classical and robust tests

There are several articles, such as **[A Powerful Test for Multivariate Normality] (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3927875/)**, **[On tests for multivariate normality and associated simulation studies] (https://www.tandfonline.com/doi/abs/10.1080/10629360600878449)** and **[Comparison of some multivariate normality tests: A simulation study] (https://doi.org/10.21833/ijaas.2016.12.011)**, that provide numerical comparison of various tests. It seems the *Henze-Zirkler* and *Royston* tests are preffered.

The R package `MVN` implements several of these tests.

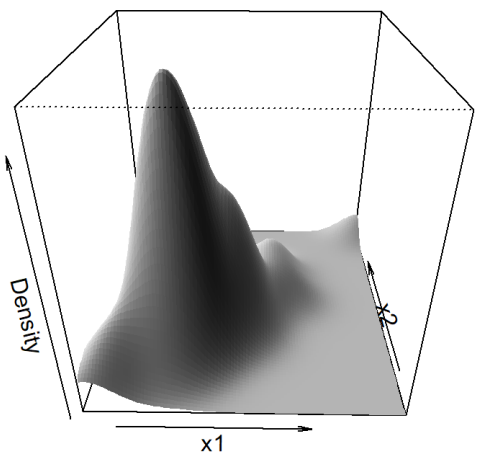
```
library(MVN)

# Perform Royston's test and create a chi-square plot
mvn(dat[, 1:4], mvnTest = "royston", multivariatePlot = "qq")
```

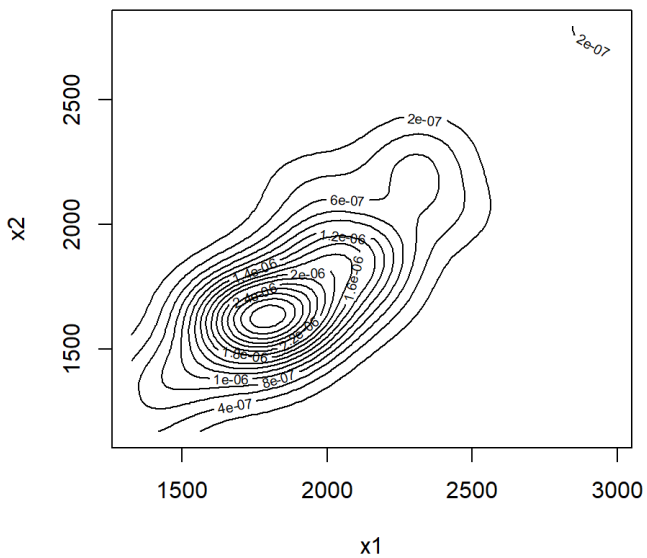


```
## $multivariateNormality
##      Test        H     p value MVN
## 1 Royston 9.823858 0.009534665  NO
##
## $univariateNormality
##           Test  Variable Statistic   p value Normality
## 1 Shapiro-Wilk    x1        0.9307    0.0512    YES
## 2 Shapiro-Wilk    x2        0.9127    0.0175    NO
## 3 Shapiro-Wilk    x3        0.9326    0.0575    YES
## 4 Shapiro-Wilk    x4        0.9613    0.3337    YES
##
## $Descriptives
##     n     Mean  Std.Dev Median  Min  Max    25th    75th      Skew
## x1 30 1906.100 324.9866 1863.0 1325 2983 1715.25 2057.25 1.0380842
## x2 30 1749.533 318.6065 1680.0 1170 2794 1595.50 1888.75 1.1435912
## x3 30 1509.133 303.1783 1466.0 1002 2412 1295.75 1623.75 0.9800274
## x4 30 1724.967 322.8436 1674.5 1176 2581 1520.25 1880.75 0.5978431
##       Kurtosis
## x1  2.03586397
## x2  1.94986381
## x3  0.99683699
## x4 -0.04626509
```

We can easily create bivariate density estimates and contour plots using thie function.

```
par(mfrow = c(1,2))
# Henze-Zirkler test and construct perspective plot
result1 <- mvn(dat[, 1:2], mvnTest = "hz", multivariatePlot = "persp")

# Henze-Zirkler test and contour plot
result2 <- mvn(dat[, 1:2], mvnTest = "hz", multivariatePlot = "contour")
```






```
result1
```

```
## $multivariateNormality
##            Test       HZ  p value MVN
## 1 Henze-Zirkler 0.567119 0.226055 YES
##
## $univariateNormality
##           Test  Variable Statistic   p value Normality
## 1 Shapiro-Wilk    x1        0.9307    0.0512    YES
## 2 Shapiro-Wilk    x2        0.9127    0.0175    NO
##
## $Descriptives
##     n     Mean  Std.Dev Median  Min  Max    25th    75th     Skew Kurtosis
## x1 30 1906.100 324.9866   1863 1325 2983 1715.25 2057.25 1.038084 2.035864
## x2 30 1749.533 318.6065   1680 1170 2794 1595.50 1888.75 1.143591 1.949864
```

```
result2
```

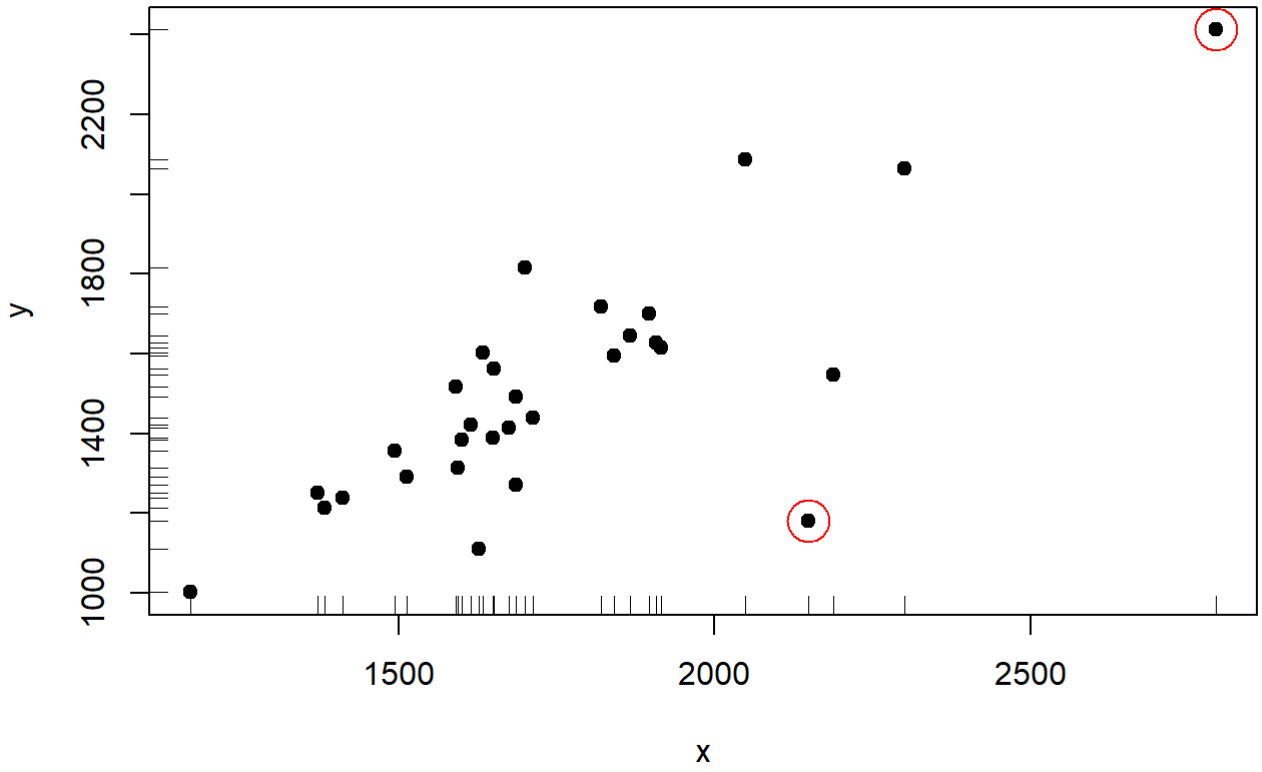
```
## $multivariateNormality
##            Test       HZ  p value MVN
## 1 Henze-Zirkler 0.567119 0.226055 YES
##
## $univariateNormality
##           Test  Variable Statistic   p value Normality
## 1 Shapiro-Wilk    x1        0.9307    0.0512    YES
## 2 Shapiro-Wilk    x2        0.9127    0.0175    NO
##
## $Descriptives
##     n     Mean  Std.Dev Median  Min  Max    25th    75th     Skew Kurtosis
## x1 30 1906.100 324.9866   1863 1325 2983 1715.25 2057.25 1.038084 2.035864
## x2 30 1749.533 318.6065   1680 1170 2794 1595.50 1888.75 1.143591 1.949864
```

See the vignette **[MVN: An R Package for Assessing Multivariate Normality] (https://cran.r-project.org/web/packages/MVN/vignettes/MVN.pdf)** for details of the full capabilities of the `MVN` package.

Outlier detection

Outliers can be viewed as unusual data points that do not seem to follow the pattern of variability produced by other observations. Johnson and Wichern suggest inspecting data to detect outliers whenever possible visually.

Univariate outliers can be detected using a dot plot or boxplot. However, it might be more complicated for multivariate data. See the scatterplot below.



One outlier is detached from the rest of the data; both its coordinates are large relative to the rest of the values. The other outlier is hard to detect from univariate plots (e.g., boxplot) as each of its components has typical values.

We start with discussing two extensions of the univariate boxplot to the bivariate situation.

Bivariate boxplot

A bivariare analogue of the usual boxplot is proposed in the article **[Goldberg and Iglewicz (1992). Bivariate Extensions of the Boxplot](https://amstat.tandfonline.com/doi/abs/10.1080/00401706.1992.10485280)**. The `bvbox()` function in the `MVA` package implements this method.

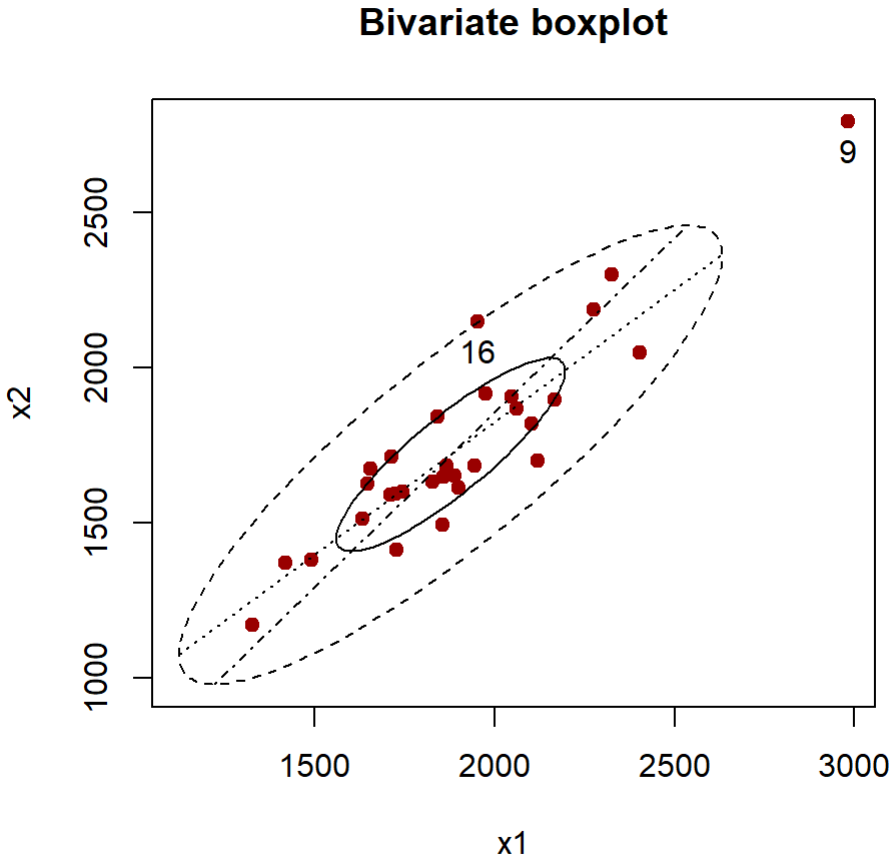
Let us look at the variables x_1 and x_2 from the lumber stiffnes data discussed before.

```
library(MVA)
```

```
## Loading required package: HSAUR2
```

```
## Loading required package: tools
```

```
bvbox(dat[, 1:2],
      pch=19, col="#990000",
      xlab = "x1", ylab = "x2", main ="Bivariate boxplot"
     )
text(dat[c(9, 16), 1], dat[c(9, 16), 2], pos= 1, labels = c(9, 16))
```



The bivariate boxplot consists of the following:

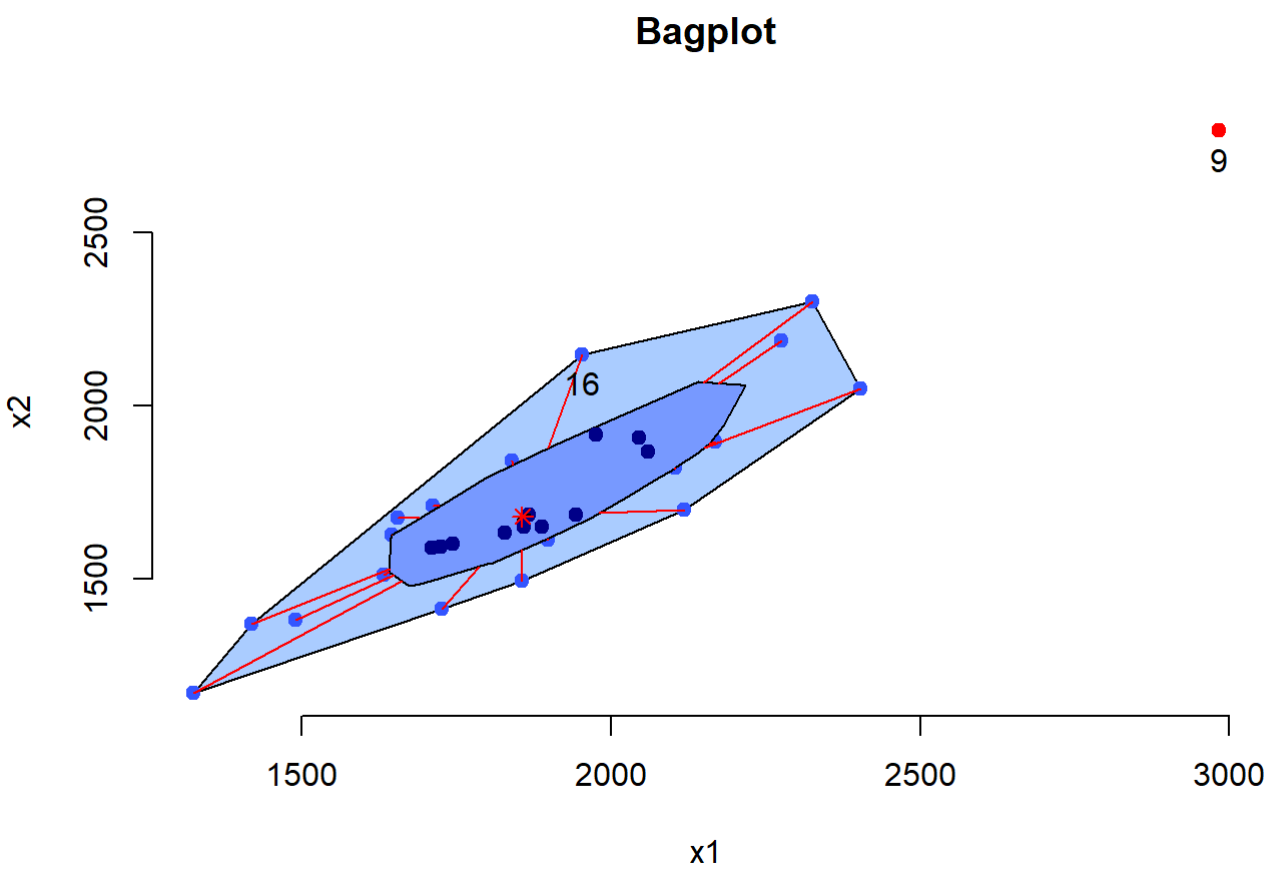
- Two concentric ellipses, the inner ellipse (called the “hinge”“) contains 50% of the data, and the outer ellipse (called the”fence“”) determines potential outliers. These ellipses are drawn based on robust measures of location, scale, and correlation, and a constant, D, that determines the distance of the fence from the hinge. Goldberg and Iglewicz (1992) propose to use $D = 7$ so that the outer ellipse forms an approximate 99% confidence bound.
- Resistant (robust) regression lines of both y on x and x on y are drawn. Their intersection shows the location estimator.

It seems observation 16 is an outlier. However, observation 9 is on the fence.

Bagplot

Another bivariate extension of the usual boxplot, called bagplot, has been suggested in the article **[Rousseeuw, Ruts and Tukey (1999). The Bagplot: A Bivariate Boxplot] (https://www.tandfonline.com/doi/abs/10.1080/00031305.1999.10474494)**

```
# Example of a Bagplot
library(aplpack)
bagplot(dat[, 1], dat[, 2],
        xlab = "x1",
        ylab = "x2",
        main = "Bagplot",
        pch = 19, cex = 1)
text(dat[c(9, 16), 1], dat[c(9, 16), 2], pos= 1, labels = c(9, 16))
```



The bagplot is based on the concept of *halfspace location depth* of a point relative to a bivariate dataset, which extends the univariate concept of rank. The plot consists of the following:

- An inner convex polygon, called the “bag,” containing 50% of the data points (with the largest depth).
- The outer polygon, called the “fence” is created by magnifying the bag by a factor of three. The fence separates inliers from outliers. The fence is not plotted, but the outliers are plotted in red. The observations between the bag and the fence are shown using a lighter color.

The bagplot visualizes the location, spread, correlation, skewness, and tails of the data. It is not limited to elliptical (e.g., multivariate normal) distributions.

Johnson and Wichern suggests the following steps for detecting outliers:

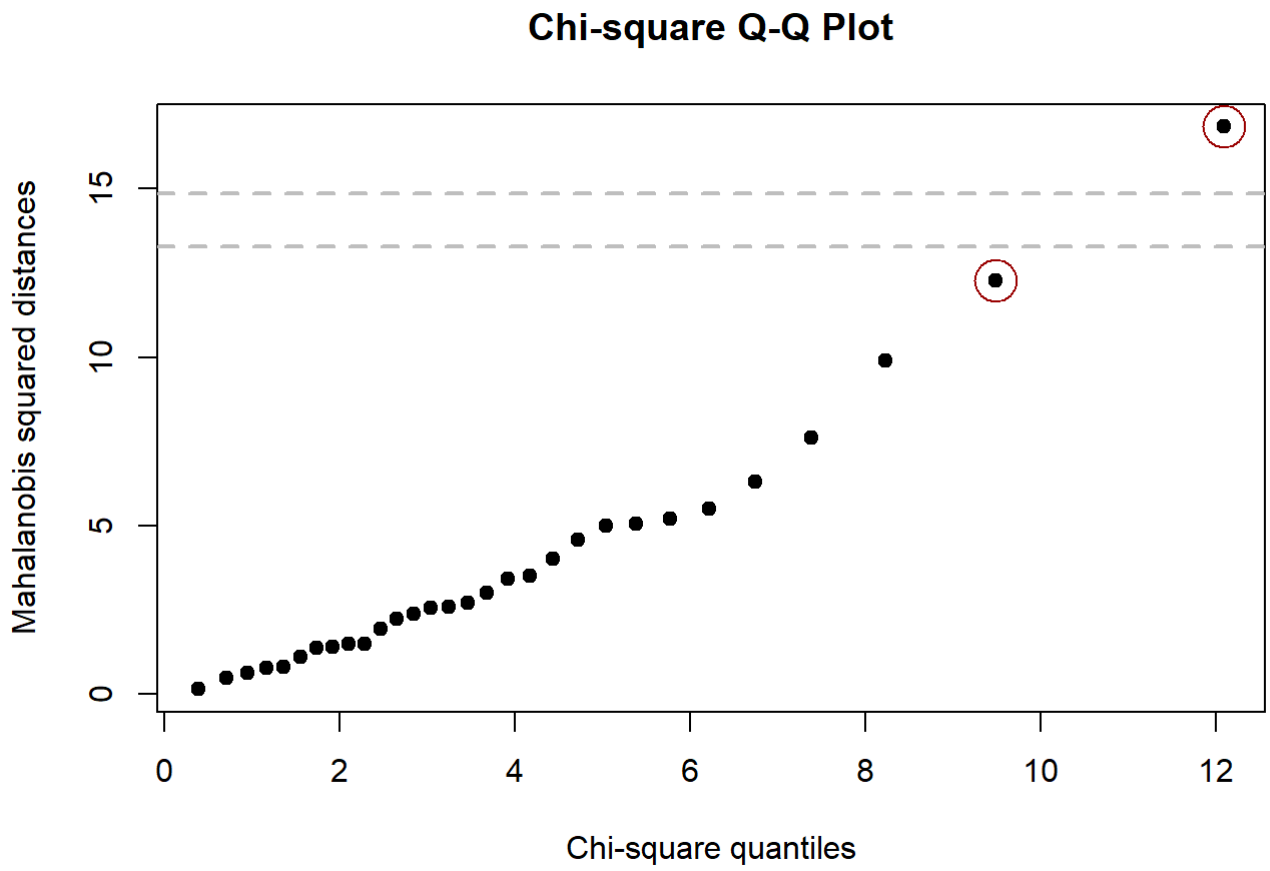
- Construct dotplot/boxplot of each variable
- Make scatterplots for each pair of variables
- Calculate standardized values for each variable

$$z_k = \frac{x_k - \bar{x}_k}{\sqrt{s_k^2}}.$$

  Examine the standardized values for extreme values. This depends of the sample size as well as number of variables. Even is the data came from a normal distribution, we can expect 1% data to exceed 3.

- Calculate the Mahalanobis squared distances $(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^T \boldsymbol{S}^{-1} (\boldsymbol{x}_i - \bar{\boldsymbol{x}})$ and create chi-square plot. Examine the points with unusually large distance values. Here "large" is measured by an appropriate percentile of χ_p^2 distrubition, e.g., 0.005 or 0.01 quantiles.

```
chisquare.plot(x = dat[, 1:4], mark = 2)
abline(h=qchisq(0.995, df = 4), col="grey", lty=2, lwd=2)
abline(h=qchisq(0.99, df = 4), col="grey", lty=2, lwd=2)
```



The top and bottom lines correspond to 0.005 or 0.01 quantiles, respectively.

Main page: **ST 437/537: Applied Multivariate and Longitudinal Data Analysis (https://maityst537.wordpress.ncsu.edu/)**