

Failed.
Correction: Please recompute the simultaneous intervals in Question 3.

Assignment 2

Mohsen Pirmoradian, Ahmed Alhasan, Asad Enver, Ali Etminan, Mubarak Hussain
11/28/2019

Question 1: Test of outliers

```
## $Outliers
##   SAM   PNG  KORN
## 35.01 30.51 26.17

## $`Outliers after adjustment`
##   SAM   PNG
## 35.01 30.51
```

- a)
- Because confidence interval 0.001 means that there's a 0.1% chance of getting outliers, and because we are testing outliers in different races (7) there is a chance of getting different results with this threshold.
 - Therefore, it is **adviseable to do multiple testing correction**. The simplest method to do this is Bonferroni adjustment which divide alpha by the number of tests (in this case 7)
 - 0.1% is reasonable enough to reduce the change in getting different results, since there is trade off between not classifying any country as outlier (using low alpha) and capturing many outliers but with different results from different races (using high alpha)

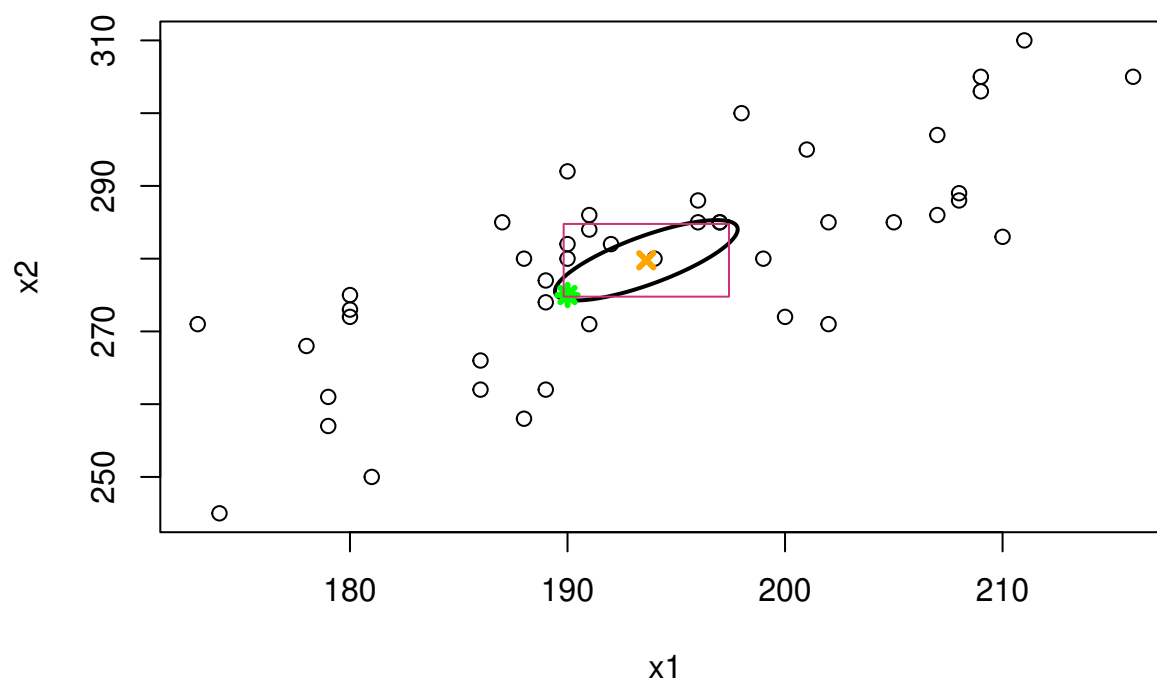
Though still not a very accurate wording, I would say, instead, that Mahalanobis penalises the long axes instead of the shorts.

- b)
- Because the distribution of the records between each two races are in elipsoid shape its wise to use Mahalanobis to measure the distance for such distribution, however Mahalanobis **penalize** the distance on the short axis (or it give less weight to distances along the long axis) and because North Korea lies further way on the short axis it is treated as far as a country with longer distance from the center of the ellipse but on the long axis.

In fact, we do not need multiple testing correction here. We are only performing one hypothesis testing on each data point. Think about it this way: it does not matter if you have a 10-million-row or 1000-row data set, if these data perfectly normal, you will still have about 0.1% of the data points identified as outliers using this test (by Result 4.7 in text book). So you have perfect control on how much data you want to remove by controlling alpha. If you have a 10-million-row data, for example, you can always say "I don't think there are 10000 (that is 0.1%) outliers; I think only 100 of them are", and go ahead to set alpha = 0.001%. The moment you set your new alpha, you know your false discovery rate is about 0.001% if the rest of your data were normal enough. So there is no need to force yourself to divide your alpha by 10-million: divide or not, you still have full control.

On the other hand, if you t-test each column seperately without Bonferroni correction, by setting alpha = 0.1%, you cannot say that about 0.1% of data points will be removed anymore: in other words, in this case, you lost control on your false discovery rates. This is why univariate t-testings on multiple columns independently require Bonferroni correction.

Question 2: Test, confidence region and confidence intervals for a xbar vector
a)



The orange cross is the mean of sample and the green point is the location of μ_0 . As it is clear the point is inside the ellipse, therefore this vector contains plausible values for μ .

- Because T^2 is smaller than the critical value at $\alpha = 5\%$ we can not reject the null hypotheses and we can conclude that the population means of the male birds are plausible means for the female birds.

b)

The related T^2 and *Benferroni* intervals for μ_1 :

```
##      T2_mu1_Lo T2_mu1_Hi Bon_mu1_Lo Bon_mu1_Hi
## [1,]  189.4217 197.8227  189.8216  197.4229
```

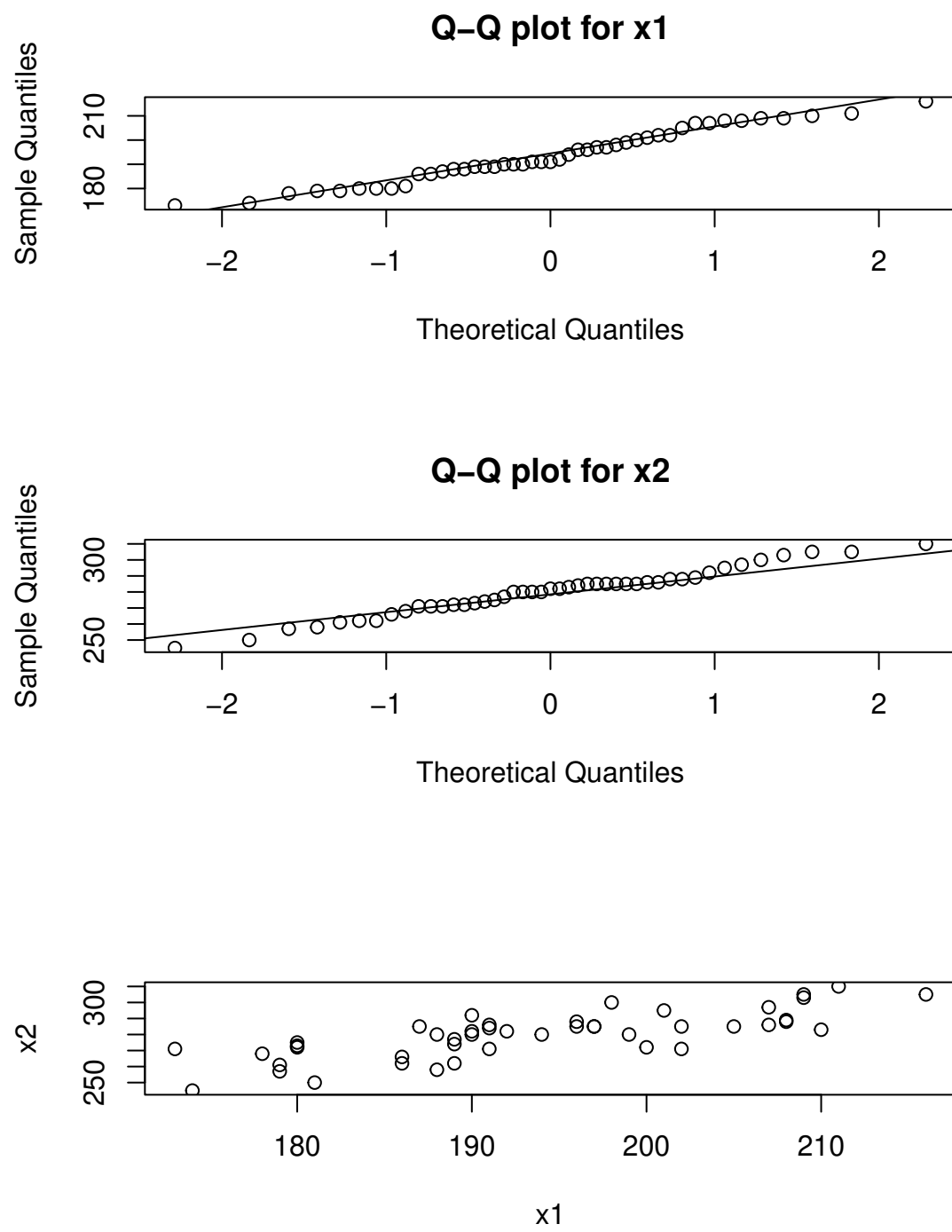
The related T^2 and *Benferroni* intervals for μ_2 :

```
##      T2_mu2_Lo T2_mu2_Hi Bon_mu2_Lo Bon_mu2_Hi
## [1,]  274.2564 285.2992  274.7819  284.7736
```

Correct.

As it can be identified from these values the Benferroni intervals are shorter than those calculated from T^2 . Refer to the book: "The simultaneous confidence intervals (T^2) are ideal for "data snooping." The confidence coefficient $1 - \alpha$ remains unchanged for any choice of \mathbf{a} , so linear combinations of the components μ_i that merit inspection based upon an examination of the data can be estimated.

c)



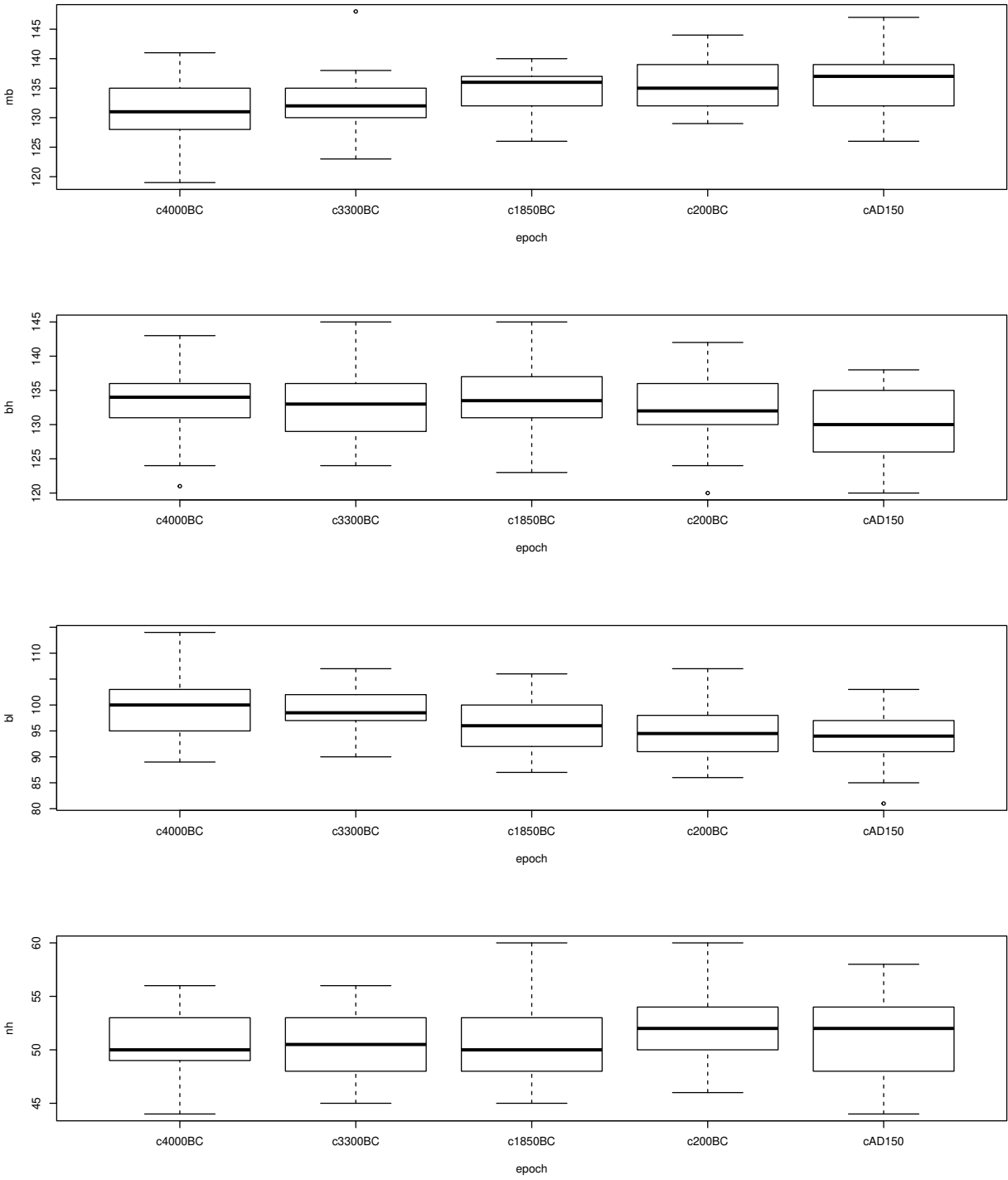
The Q-Q plot for both variables (x_1 & x_2) illustrate a linear trend. The scatterplot of two variables also indicates a linear relationship between these two features. These linear trends can lead us to this conclusion that the population can be considered as normal.

Question 3: Comparison of xbar vectors (one{way MANOVA)

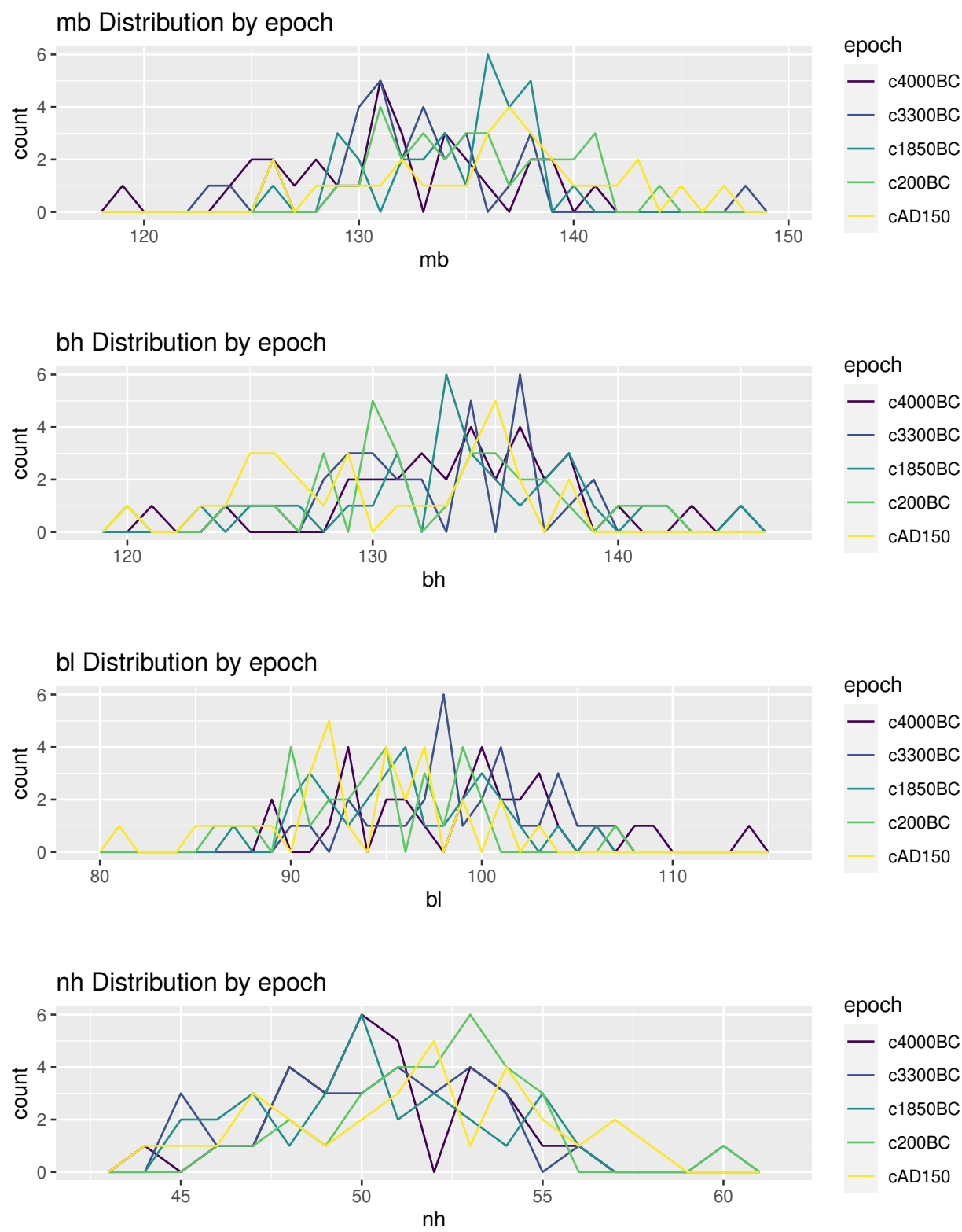
```
# fit manova model
sk.mod <- lm(cbind(mb, bh, bl, nh) ~ epoch, data=Skulls)
sk.mod

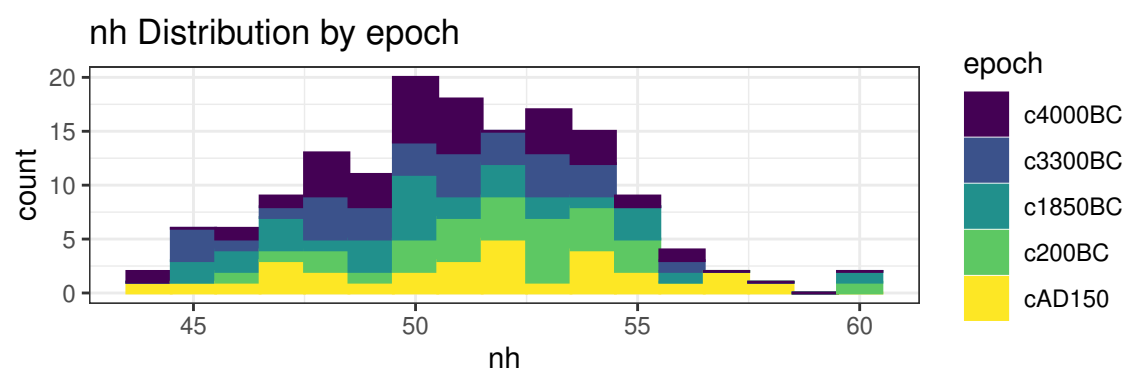
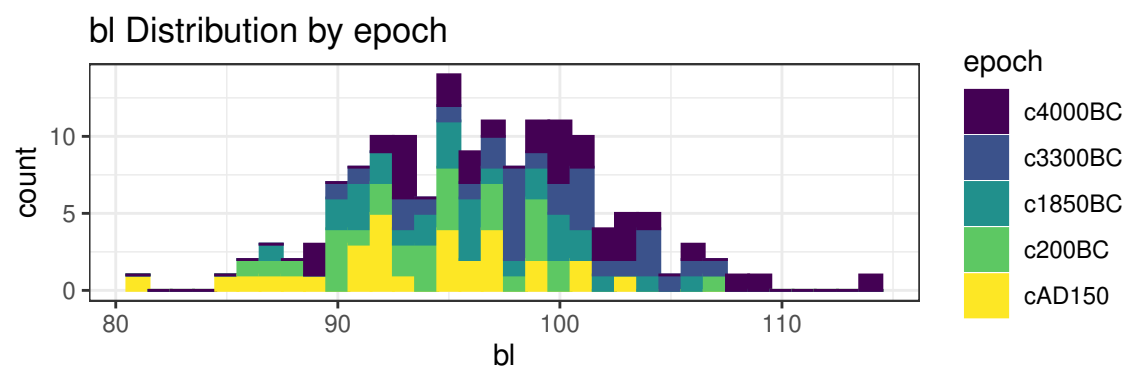
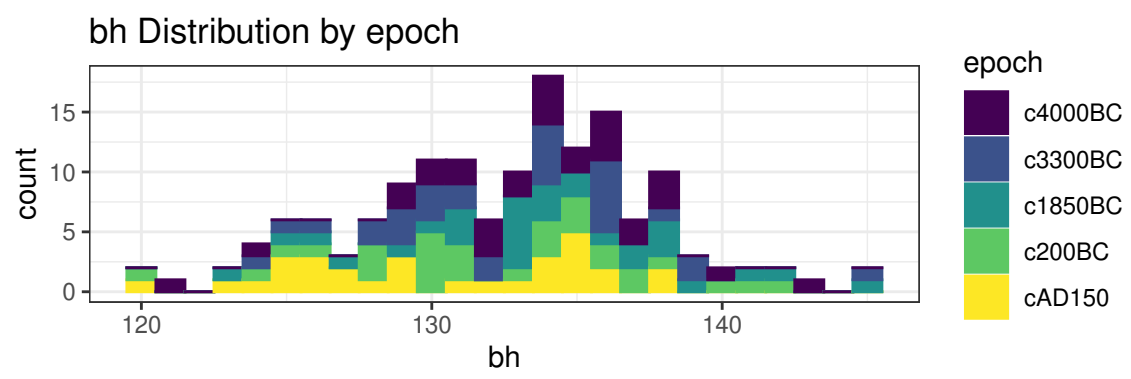
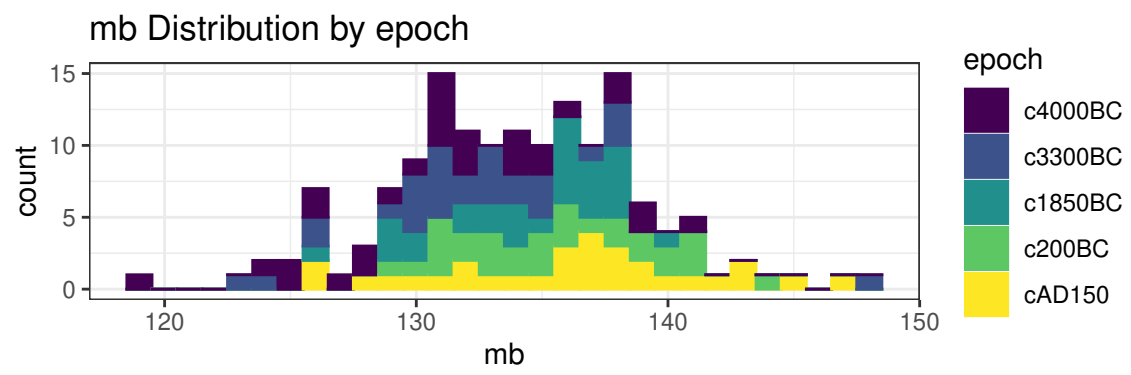
##
## Call:
## lm(formula = cbind(mb, bh, bl, nh) ~ epoch, data = Skulls)
##
## Coefficients:
##           mb           bh           bl           nh
## (Intercept) 133.97333 132.54667  96.46000  50.93333
## epoch.L      4.02663  -2.19251  -5.01748   1.07517
## epoch.Q     -0.46325  -1.26504  -0.08909   0.12472
## epoch.C     -0.46380  -0.78003   1.07517  -0.83273
## epoch^4      0.34263   0.80479  -0.66136  -0.41833
```

Boxplots:

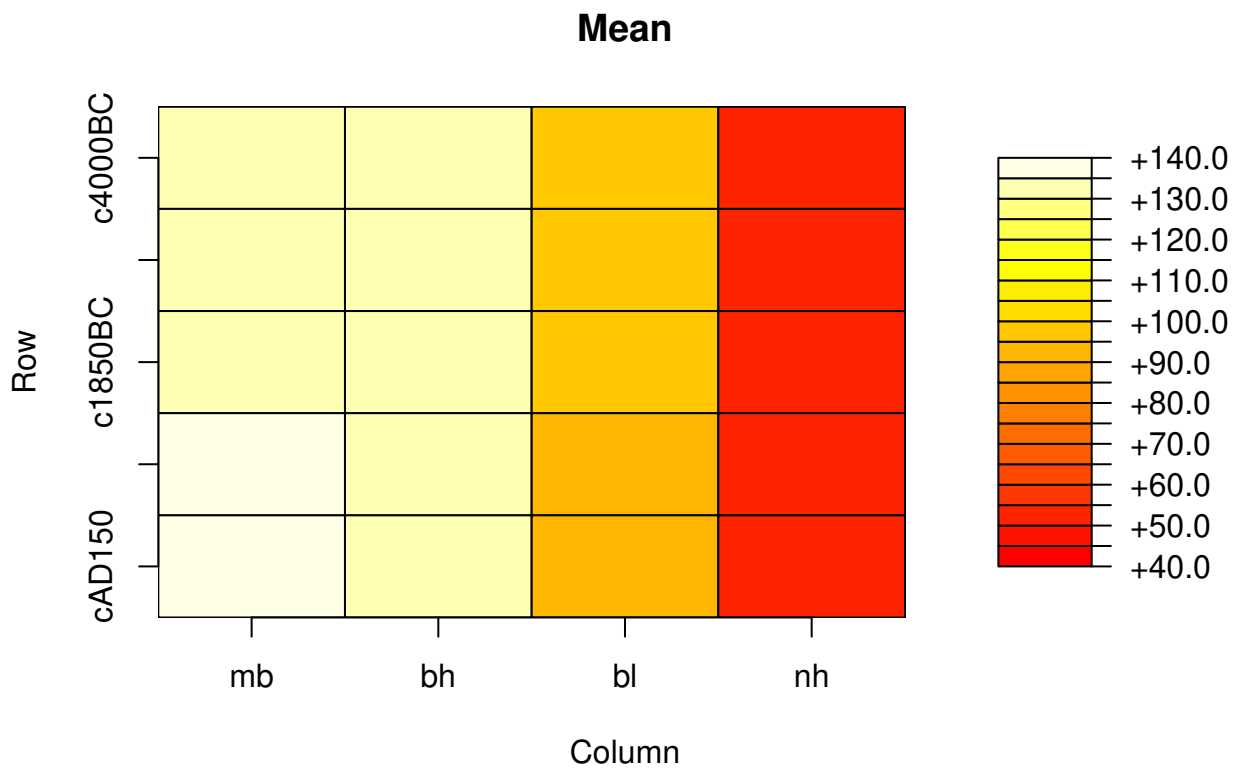


Features distributions with respect to the epochs:

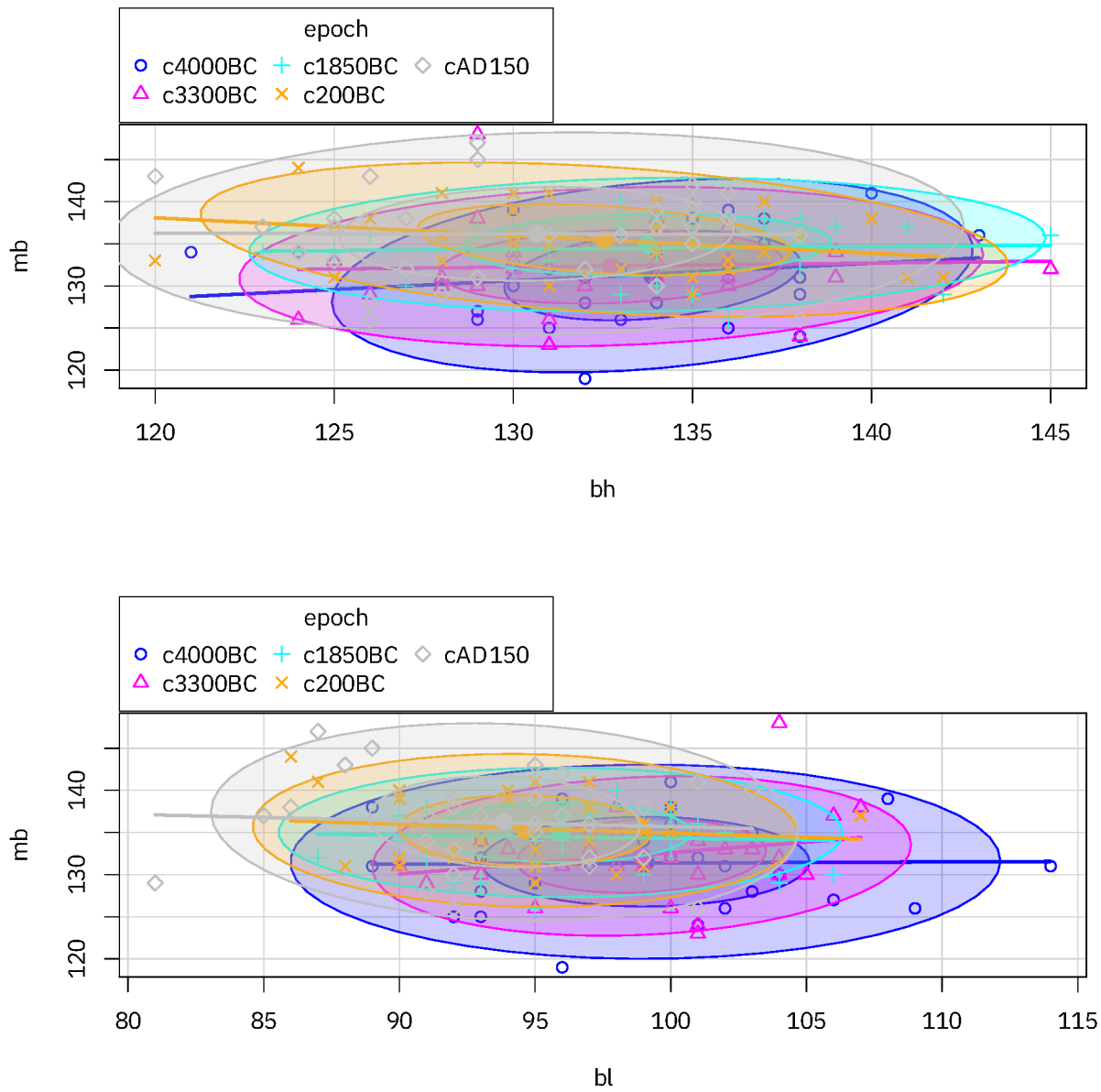


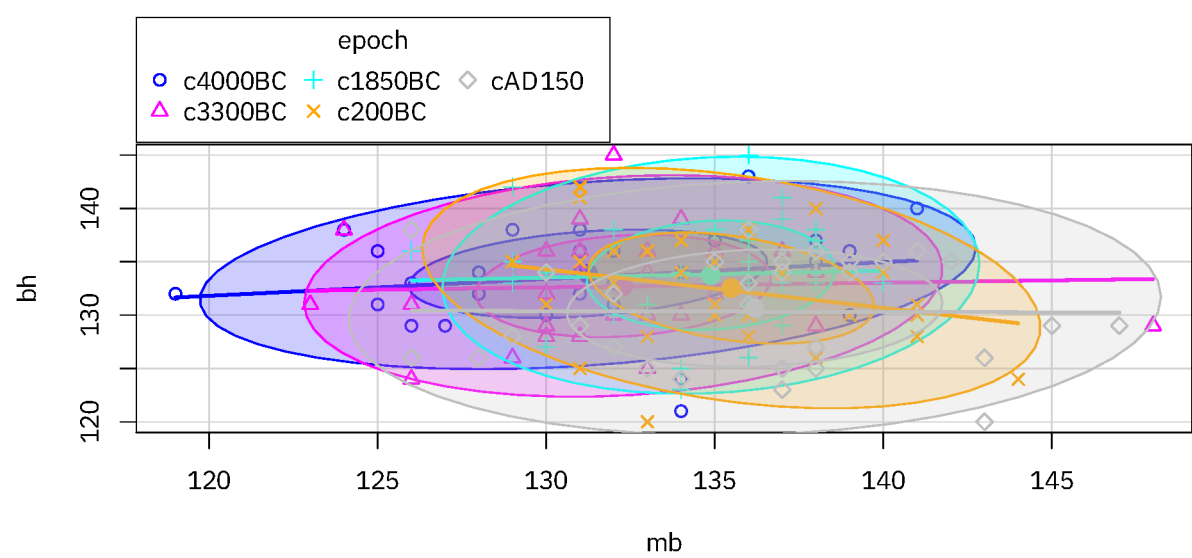
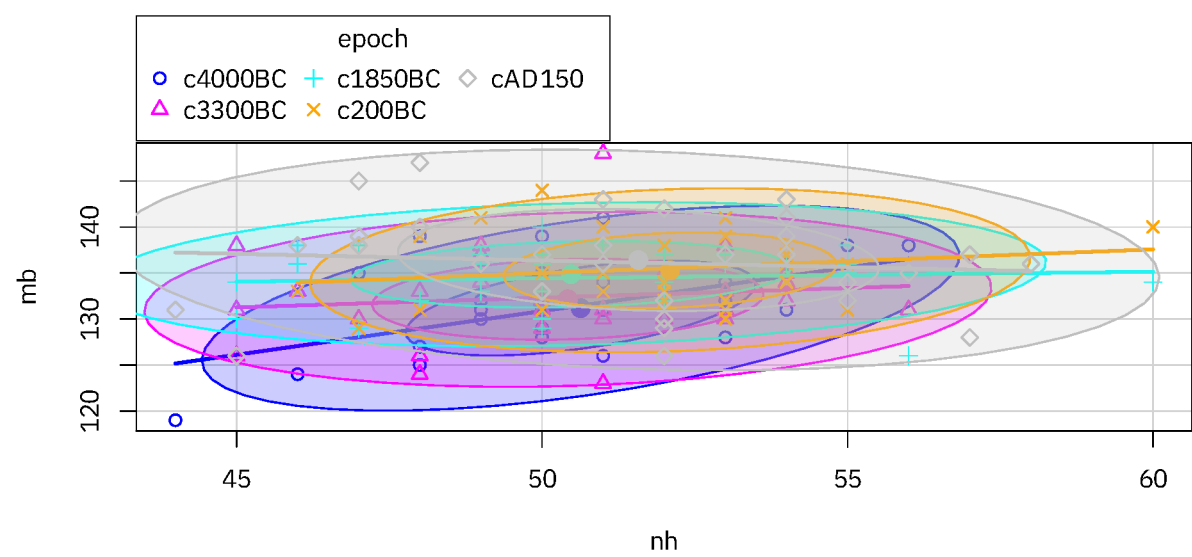


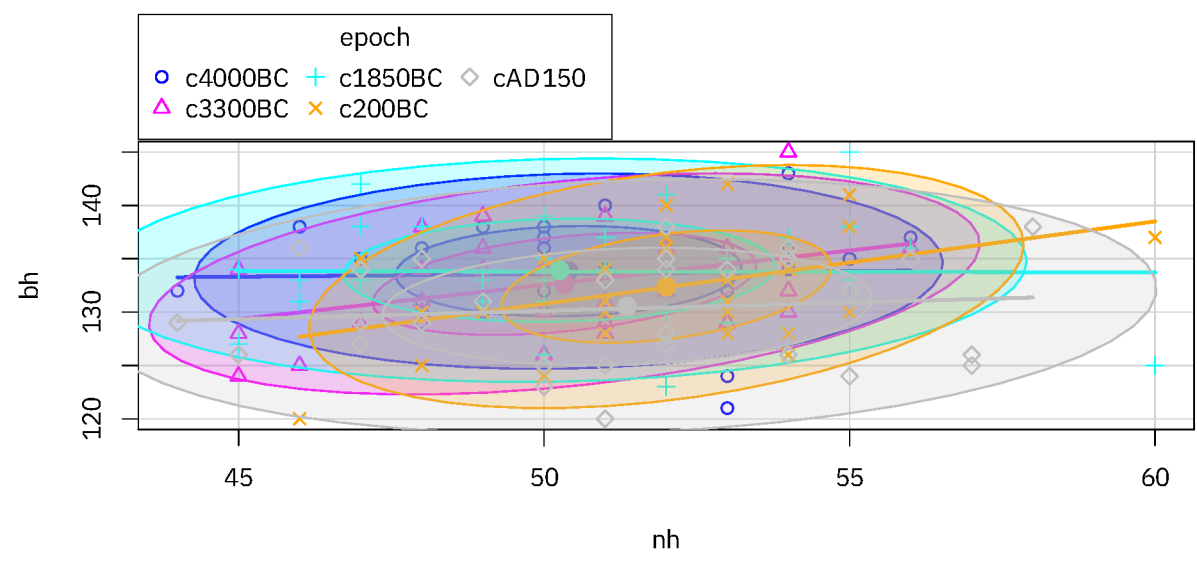
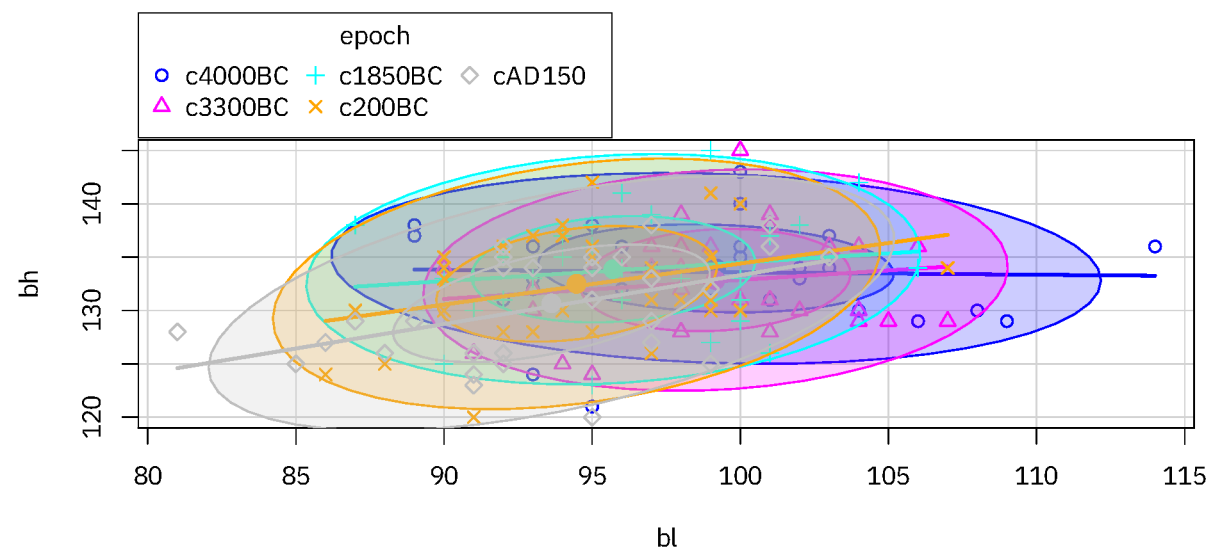
xbar matrix indicating the mean value for each feature with respect to epochs:

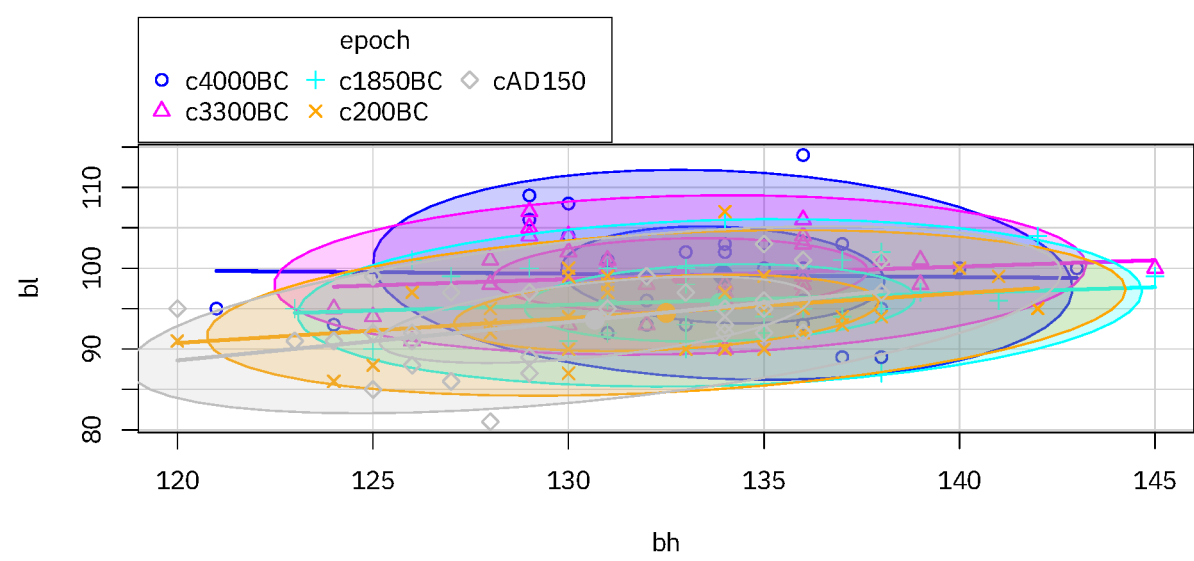
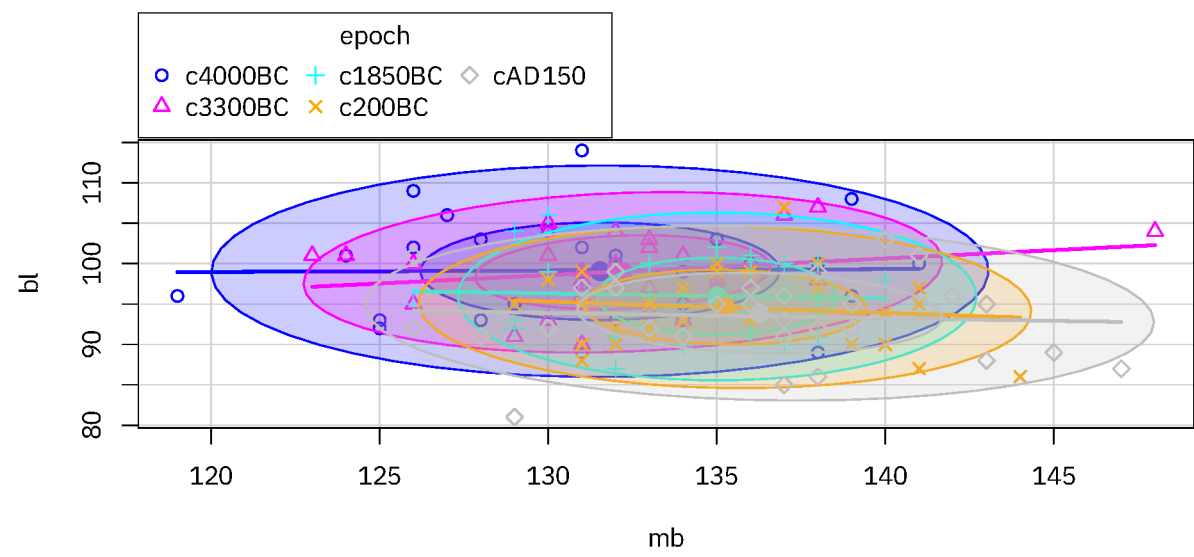


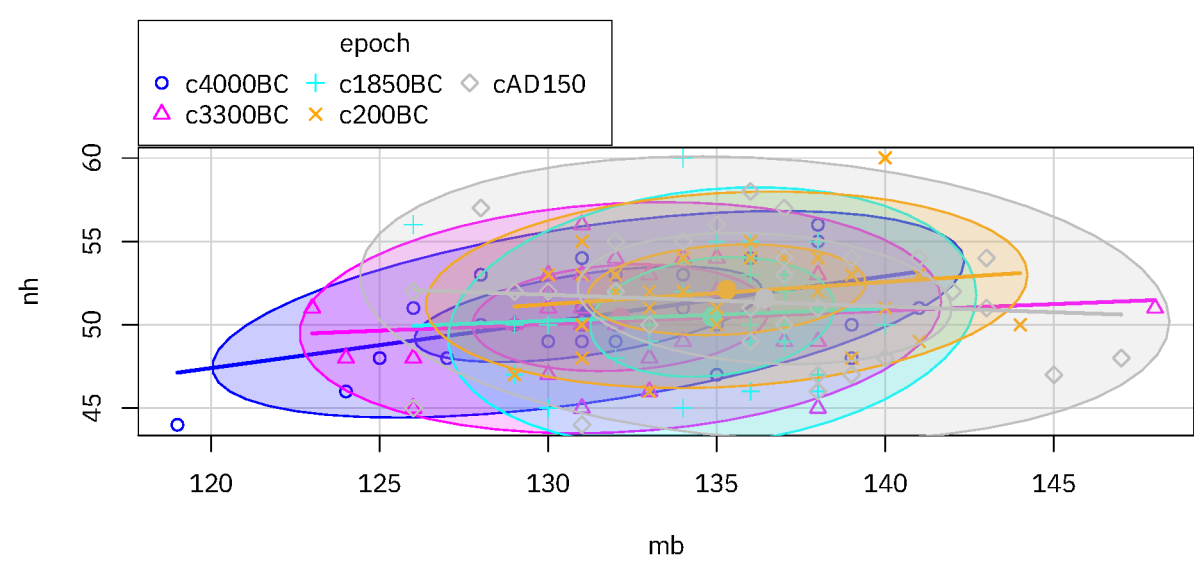
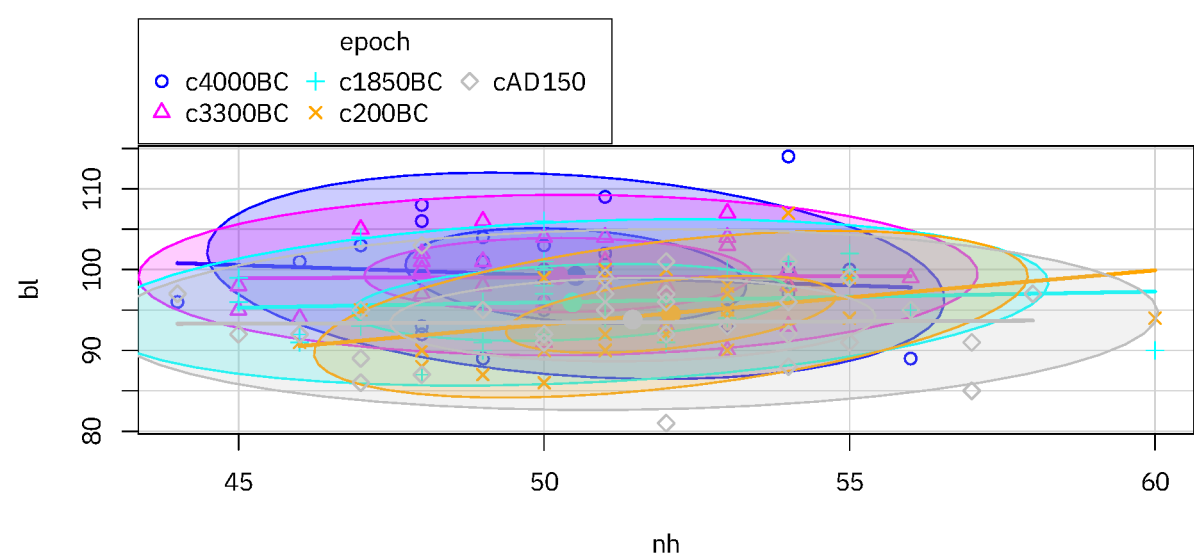
Scatter plots of features conditioning on epochs. The following plots illustrate the relationship between the different features mutually and the ellipses indicates the distribution of those relationships considering each epoch:

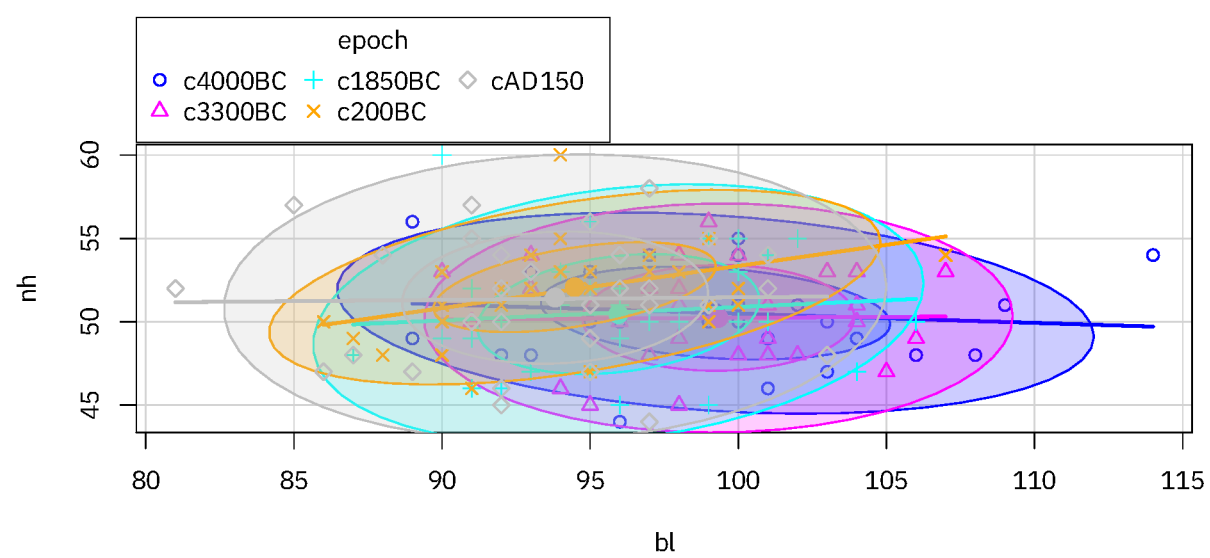
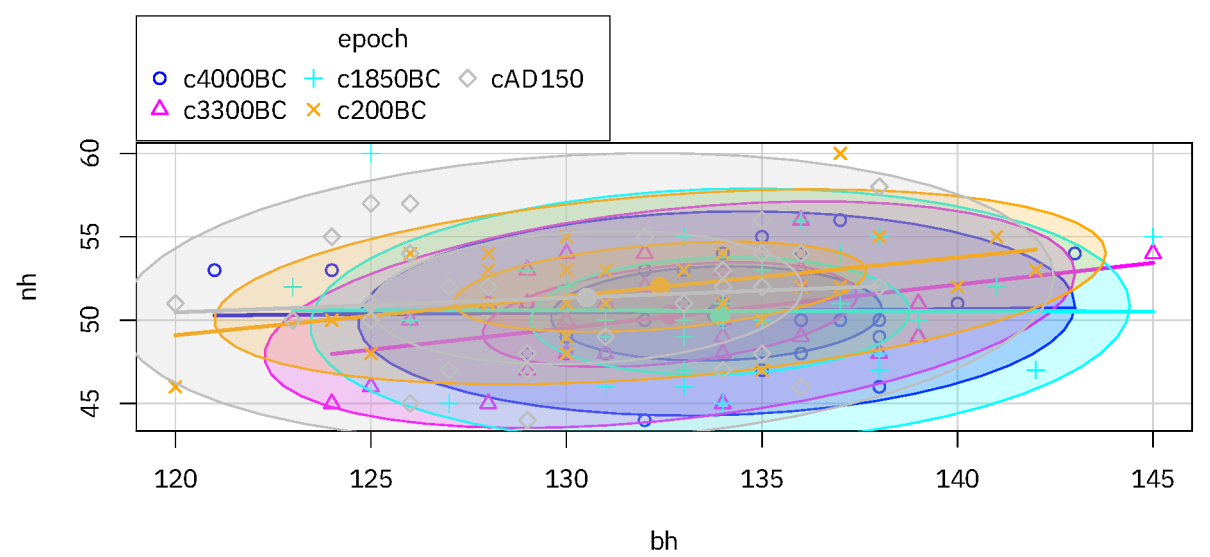












(b)

1)Summary of the manova model

```
## Response mb :
##           Df Sum Sq Mean Sq F value    Pr(>F)
## as.factor(epoch)  4  502.83  125.707   5.9546 0.0001826 ***
## Residuals      145 3061.07   21.111
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Response bh :
##           Df Sum Sq Mean Sq F value    Pr(>F)
## as.factor(epoch)  4  229.9   57.477   2.4474 0.04897 *
## Residuals      145 3405.3   23.485
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Response bl :
##           Df Sum Sq Mean Sq F value    Pr(>F)
## as.factor(epoch)  4  803.3  200.823   8.3057 4.636e-06 ***
## Residuals      145 3506.0   24.179
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Response nh :
##           Df Sum Sq Mean Sq F value    Pr(>F)
## as.factor(epoch)  4   61.2   15.300   1.507 0.2032
## Residuals      145 1472.1   10.153
```

2)Different tests for MANOVA

2-1)Hotelling-Lawley

```
##           Df Hotelling-Lawley approx F num Df den Df    Pr(>F)
## as.factor(epoch)  4           0.48182   4.231    16   562 8.278e-08 ***
## Residuals      145
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2-2)Roy

```
##           Df      Roy approx F num Df den Df    Pr(>F)
## as.factor(epoch)  4 0.4251    15.41     4   145 1.588e-10 ***
## Residuals      145
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

All correct.

2-3)Pillai

```
##           Df Pillai approx F num Df den Df    Pr(>F)
## as.factor(epoch)  4 0.35331    3.512    16   580 4.675e-06 ***
## Residuals      145
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2-4)Wilks

```
##              Df    Wilks approx F num Df den Df    Pr(>F)
## as.factor(epoch)  4 0.66359   3.9009      16 434.45 7.01e-07 ***
## Residuals        145
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3)Pairwise comparisons

3-1)4000BC, 3300BC and 185BC

```
##              Df    Pillai approx F num Df den Df    Pr(>F)
## as.factor(epoch)  1 0.027674  0.39135      4     55 0.8139
## Residuals        58
```

3-2)4000BC, AD150

```
##              Df    Pillai approx F num Df den Df    Pr(>F)
## as.factor(epoch)  1 0.36182   7.7956      4     55 4.736e-05 ***
## Residuals        58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4)Confidence interval

```
##              47.6 %      52.4 %
## mb:(Intercept) 133.95094774 133.99571892
## mb:epoch.L      3.97657786   4.07668925
## mb:epoch.Q      -0.51330852  -0.41319712
## mb:epoch.C      -0.51385642  -0.41374502
## mb:epoch^4       0.29257650   0.39268790
## bh:(Intercept) 132.52305603 132.57027730
## bh:epoch.L      -2.24530750  -2.13971752
## bh:epoch.Q      -1.31783153  -1.21224156
## bh:epoch.C      -0.83282348  -0.72723350
## bh:epoch^4       0.75199228   0.85758225
## bl:(Intercept)  96.43604280  96.48395720
## bl:epoch.L      -5.07105048  -4.96391063
## bl:epoch.Q      -0.14265700  -0.03551716
## bl:epoch.C       1.02160448   1.12874433
## bl:epoch^4      -0.71492975  -0.60778991
## nh:(Intercept)  50.91780926  50.94885741
## nh:epoch.L       1.04046152   1.10988729
## nh:epoch.Q       0.09000903   0.15943480
## nh:epoch.C      -0.86744600  -0.79802023
## nh:epoch^4      -0.45304290  -0.38361713

##              47.6 % 52.4 %

## Response mb :
##
## Call:
## lm(formula = mb ~ epoch, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

You have misunderstood the question.

The question means that you should compute the interval described in Result 6.5.

Hint: The diagonal of the W matrix is the same as the "Residuals" column given by `print(manova(...))`


```

## -12.3667 -2.5000 0.1667 2.6333 15.6333
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 133.9733    0.3752 357.118 < 2e-16 ***
## epoch.L      4.0266    0.8389   4.800 3.9e-06 ***
## epoch.Q     -0.4633    0.8389  -0.552 0.582
## epoch.C     -0.4638    0.8389  -0.553 0.581
## epoch^4      0.3426    0.8389   0.408 0.684
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.595 on 145 degrees of freedom
## Multiple R-squared:  0.1411, Adjusted R-squared:  0.1174
## F-statistic: 5.955 on 4 and 145 DF, p-value: 0.0001826
##
##
## Response bh :
##
## Call:
## lm(formula = bh ~ epoch, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.600  -3.200   0.200   3.375  12.300
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 132.5467    0.3957 334.983 <2e-16 ***
## epoch.L     -2.1925    0.8848  -2.478 0.0144 *
## epoch.Q     -1.2650    0.8848  -1.430 0.1549
## epoch.C     -0.7800    0.8848  -0.882 0.3794
## epoch^4      0.8048    0.8848   0.910 0.3645
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.846 on 145 degrees of freedom
## Multiple R-squared:  0.06325, Adjusted R-squared:  0.0374
## F-statistic: 2.447 on 4 and 145 DF, p-value: 0.04897
##
##
## Response bl :
##
## Call:
## lm(formula = bl ~ epoch, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.5000  -3.1667  -0.0333   3.5000  14.8333
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  96.46000    0.40149 240.255 < 2e-16 ***
## epoch.L     -5.01748    0.89776  -5.589 1.1e-07 ***

```

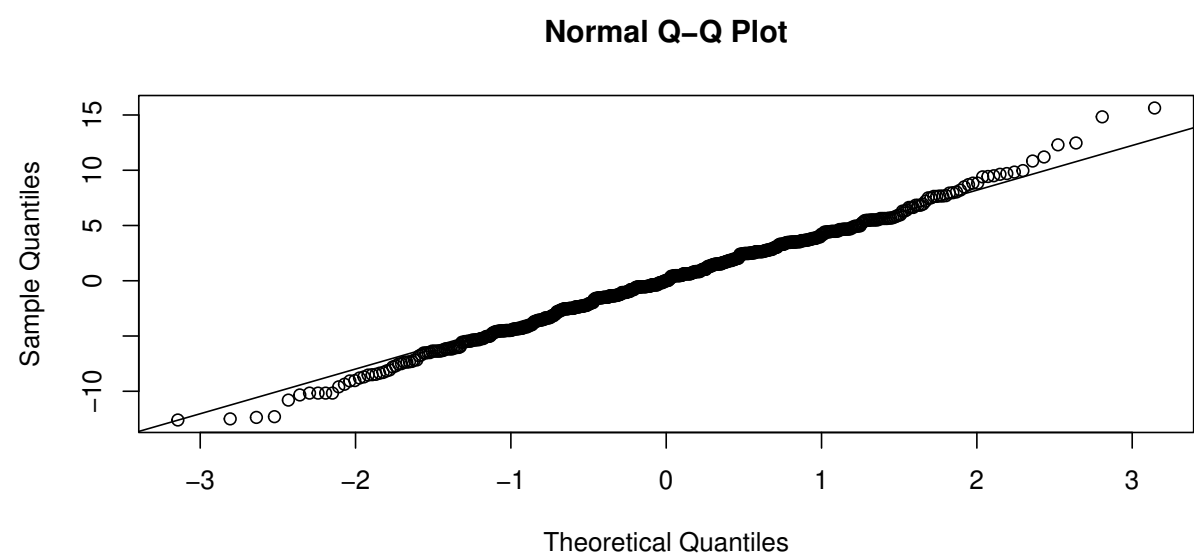
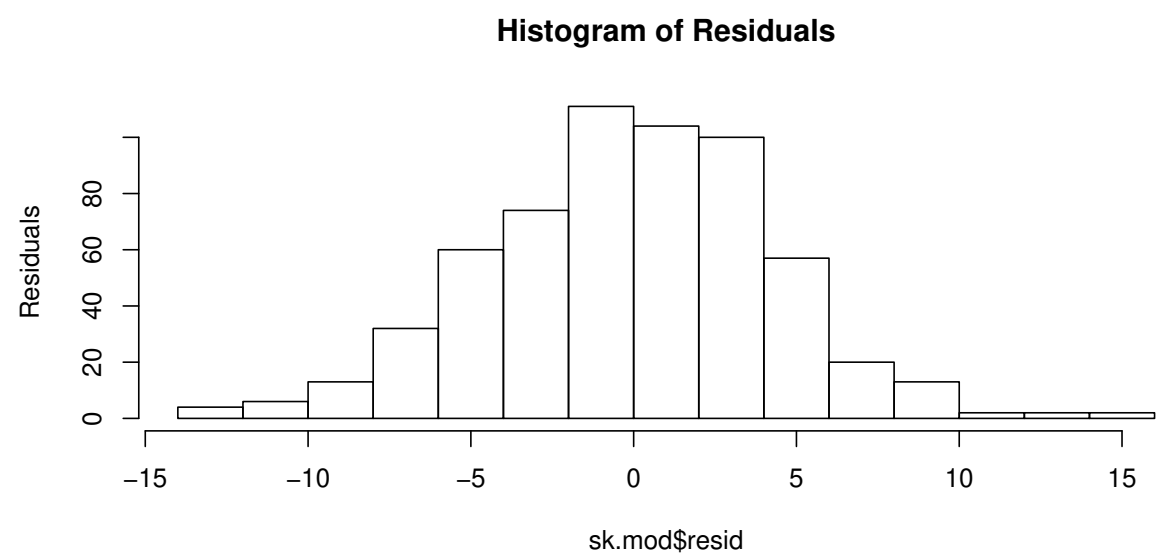
```

## epoch.Q      -0.08909    0.89776  -0.099    0.921
## epoch.C       1.07517    0.89776   1.198    0.233
## epoch^4      -0.66136    0.89776  -0.737    0.463
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.917 on 145 degrees of freedom
## Multiple R-squared:  0.1864, Adjusted R-squared:  0.164
## F-statistic: 8.306 on 4 and 145 DF,  p-value: 4.636e-06
##
##
## Response nh :
##
## Call:
## lm(formula = nh ~ epoch, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7.3667 -2.1667  0.0333  2.4333  9.4333
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  50.9333    0.2602  195.775  <2e-16 ***
## epoch.L       1.0752    0.5817   1.848   0.0666 .
## epoch.Q       0.1247    0.5817   0.214   0.8305
## epoch.C      -0.8327    0.5817  -1.431   0.1545
## epoch^4      -0.4183    0.5817  -0.719   0.4732
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.186 on 145 degrees of freedom
## Multiple R-squared:  0.03991,    Adjusted R-squared:  0.01343
## F-statistic: 1.507 on 4 and 145 DF,  p-value: 0.2032

```

Analysis:

Applying MANOVA to the given variables gives strong evidence that the epoch means of these variables differ since as the epochs become further separated in time, the skull measurements become increasingly different, especially in bl.



Analysis:

Yes, our histogram of residuals shows that the mean is zero. Since our residuals are normal, it means that our assumption is valid and model inference (confidence intervals, model predictions) should also be valid.

The data come from normal distribution.

References

- <http://www.biostathandbook.com/multiplecomparisons.html>
- <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2907892/>
- <http://users.stat.umn.edu/~helwig/notes/mvmean-Notes.pdf>

Appendix

```
knitr::opts_chunk$set(echo = TRUE)

# Question 1
setwd("E:/LiU/1st Semester/732A97-MStatistics/Assignments/2")
records <- read.table("T1-9.DAT")
colnames(records) <- c("Country", "m100", "m200", "m400", "m800", "m1500", "m3000", "Marathon")

# in the previous assignment we used abs(x-xbar(x))
# thats why we didnt get North Korea among the outliers
centered <- apply(records[, -1], 2, function(x){x-mean(x)})
rownames(centered) <- records[[1]]

Cov <- cov(records[, -1])
mah <- ((centered) %*% solve(Cov)) %*% t(centered)
dist <- (sort(diag(mah), decreasing = TRUE))

df <- dim(records[, -1])[2]-1
alpha <- 0.001

chi <- pchisq(dist, df, lower.tail = FALSE)
outliers <- list("Outliers" = round(dist[which(chi < alpha)], 2))
outliers

Bonferroni <- list("Outliers after adjustment" = round(dist[which(chi < (alpha/7))], 2))
Bonferroni

# Question 2
setwd("E:/LiU/1st Semester/732A97-MStatistics/Assignments/2")
birds <- read.table("T5-12.DAT")
colnames(birds) <- c("Tail Length", "Wing Length")
birds
x1 <- birds$`Tail Length`
x2 <- birds$`Wing Length`
n = nrow(birds)
p = ncol(birds)
alpha <- 0.05
crit_value <- sqrt((p*(n-1)/(n*(n-p))) * qf(1-alpha, p, n-p))

mu0 = c(190, 275)

Mean = apply(birds, 2, mean)
Cov = cov(birds)

angles <- seq(0, 2*pi, length.out=200)
#eigen values and eigen vectors of covariance-variance matrix
eigVal <- eigen(Cov)$values
eigVec <- eigen(Cov)$vectors
```

```

ellBase <- cbind(sqrt(eigVal[1])*crit_value*cos(angles), sqrt(eigVal[2])* crit_value*sin(angles))

ellRot <- eigVec%*%t(ellBase)

plot(x1,x2)
lines((ellRot+Mean)[1,], (ellRot + Mean)[2,], asp=1, type='l', lwd=2,
      main= "100(1-a)% Confidence Ellipsoid", xlab="x1", ylab="x2")
points(Mean[1], Mean[2], pch=4, col="orange", lwd=3)
points(mu0[1], mu0[2], pch=8, col="green", lwd=3)

bon11 <- Mean[1] - qt(1-alpha/(2*p), n-1)*sqrt(Cov[1,1]/n)
bon12 <- Mean[1] + qt(1-alpha/(2*p), n-1)*sqrt(Cov[1,1]/n)
bon21 <- Mean[2] - qt(1-alpha/(2*p), n-1)*sqrt(Cov[2,2]/n)
bon22 <- Mean[2] + qt(1-alpha/(2*p), n-1)*sqrt(Cov[2,2]/n)
rect(bon11, bon21, bon12, bon22, border= "violetred3")

T2_mu1_Lo <- mean(x1)- sqrt((p*(n-1)/(n* (n-p))) * qf(1-alpha, p, n-p)*var(x1))
T2_mu1_Hi <- mean(x1)+ sqrt((p*(n-1)/(n* (n-p))) * qf(1-alpha, p, n-p)*var(x1))
T2_mu2_Lo <- mean(x2)- sqrt((p*(n-1)/(n* (n-p))) * qf(1-alpha, p, n-p)*var(x2))
T2_mu2_Hi <- mean(x2)+ sqrt((p*(n-1)/(n* (n-p))) * qf(1-alpha, p, n-p)*var(x2))

Bon_mu1_Lo <- mean(x1) - qt(1-0.025/2, n-1)* sqrt(var(x1)/n)
Bon_mu1_Hi <- mean(x1) + qt(1-0.025/2, n-1)* sqrt(var(x1)/n)
Bon_mu2_Lo <- mean(x2) - qt(1-0.025/2, n-1)* sqrt(var(x2)/n)
Bon_mu2_Hi <- mean(x2) + qt(1-0.025/2, n-1)* sqrt(var(x2)/n)

cbind(T2_mu1_Lo, T2_mu1_Hi,Bon_mu1_Lo , Bon_mu1_Hi)
cbind(T2_mu2_Lo, T2_mu2_Hi,Bon_mu2_Lo , Bon_mu2_Hi)
qqnorm(x1, main = "Q-Q plot for x1")
qqline(x1)
qqnorm(x2, main = "Q-Q plot for x2")
qqline(x2)
plot(x1, x2)

# Question 3
library(heplots)
library(ggplot2)
data(Skulls)
data <- Skulls
# fit manova model
sk.mod <- lm(cbind(mb, bh, bl, nh) ~ epoch, data=Skulls)
sk.mod
#Boxplots
par(mfrow = c(4, 1))
boxplot(mb ~ epoch, data=data)
boxplot(bh ~ epoch, data=data)
boxplot(bl ~ epoch, data=data)
boxplot(nh ~ epoch, data=data)

par(mfrow = c(4, 1))

```

```

ggplot(Skulls, aes(mb, colour = epoch)) +
  geom_freqpoly(binwidth = 1) + labs(title="mb Distribution by epoch")

ggplot(Skulls, aes(bh, colour = epoch)) +
  geom_freqpoly(binwidth = 1) + labs(title="bh Distribution by epoch")

ggplot(Skulls, aes(bl, colour = epoch)) +
  geom_freqpoly(binwidth = 1) + labs(title="bl Distribution by epoch")

ggplot(Skulls, aes(nh, colour = epoch)) +
  geom_freqpoly(binwidth = 1) + labs(title="nh Distribution by epoch")

c <- ggplot(Skulls, aes(x=mb, fill=epoch, color=epoch)) +
  geom_histogram(binwidth = 1) + labs(title="mb Distribution by epoch")
c + theme_bw()

c <- ggplot(Skulls, aes(x=bh, fill=epoch, color=epoch)) +
  geom_histogram(binwidth = 1) + labs(title="bh Distribution by epoch")
c + theme_bw()

c <- ggplot(Skulls, aes(x=bl, fill=epoch, color=epoch)) +
  geom_histogram(binwidth = 1) + labs(title="bl Distribution by epoch")
c + theme_bw()

c <- ggplot(Skulls, aes(x=nh, fill=epoch, color=epoch)) +
  geom_histogram(binwidth = 1) + labs(title="nh Distribution by epoch")
c + theme_bw()

#Mean matrix
epoch = levels(Skulls$epoch)

Mean = matrix(0, nrow = length(epoch), ncol = 4)
rownames(Mean) = levels(Skulls$epoch)
colnames(Mean) = colnames(Skulls)[-1]
for (i in epoch) {
  for (j in colnames(Mean)) {
    Mean[i,j] = mean(Skulls[,j][Skulls[1] == i])
  }
}
library(plot.matrix)
par(mar=c(5.1, 4.1, 4.1, 4.1))
plot(Mean, breaks = 20)

#Scatterplots
scatterplot(mb ~ bh|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")
scatterplot(mb ~ bl|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")
scatterplot(mb ~ nh|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")

scatterplot(bh ~ mb|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")
scatterplot(bh ~ bl|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")
scatterplot(bh ~ nh|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")

```

```

scatterplot(bl ~ mb|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")
scatterplot(bl ~ bh|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")
scatterplot(bl ~ nh|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")

scatterplot(nh ~ mb|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")
scatterplot(nh ~ bh|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")
scatterplot(nh ~ bl|epoch, data=Skulls, ellipse=TRUE, levels=0.68, smooth=FALSE, legend.coords="topright")

manova_1 <- manova(cbind(mb, bh, bl, nh) ~ as.factor(epoch), data= data)

summary.aov(manova_1)
summary(manova_1, test = "Hotelling-Lawley")
summary(manova_1, test = "Roy")
summary(manova_1, test = "Pillai")
summary(manova_1, test = "Wilks")
#Pairwise comparisons

#4000BC, 3300BC and 185BC
manova_2 <- manova(cbind(mb, bh, bl, nh) ~ as.factor(epoch), data= data,
                  subset= as.factor(epoch) %in% c("c4000BC", "c3300BC", "c1850BC"))

#4000BC, AD150
manova_3 <- manova(cbind(mb, bh, bl, nh) ~ as.factor(epoch), data= data,
                  subset= as.factor(epoch) %in% c("c4000BC", "cAD150"))

summary(manova_2)
summary(manova_3)
intervals <- confint(sk.mod, level = 0.95/length(coef(sk.mod)))
intervals

intervals <- confint(manova_1, level = 0.95/length(coef(manova_1)))
intervals

sk.mod <- lm(cbind(mb, bh, bl, nh) ~ epoch, data=data)
#sk.mod
summary(sk.mod)

#Histogram of Residuals
hist(sk.mod$resid, main="Histogram of Residuals",
     ylab="Residuals")

#Q-Q Plot
qqnorm(sk.mod$resid)
qqline(sk.mod$resid)

```