Chapter 6: Comparing more than two means

6.1 a)

```
# Be sure to set your working directory when providing a relative path
dataset8 <- read.csv('eyeColor.csv')

ttestData <- subset(dataset8,
                    dataset8$Group == 'Blue'|
                    dataset8$Group == 'Brown')
```

6.1 b) $H_0: \quad \mu_1 = \mu_2$ $\qquad H_1: \quad \mu_1 \neq \mu_2$

6.1 c)

```
blue <- subset(ttestData$Score, ttestData$Group == 'Blue')
brown <- subset(ttestData$Score, ttestData$Group == 'Brown')
t.test(blue, brown, var.equal = TRUE) # p-value: 0.1401
```

6.1 d) The p-value is 0.1401, which is higher than the 0.05 required to reject H_0. H_0 is not rejected with 95% confidence. You can be 95% confident that the mean of the blue group is the same as the mean of the brown group. There is a risk of a type-II error.

6.1 e) The content of `dummyBrown` is a 0 for blue eyes, and a 1 for brown eyes. This kind of variable is called a dummy variable.

```
dummyBrown <- as.numeric(ttestData$Group == 'Brown')
ttestData <- cbind(ttestData, dummyBrown)
```

6.1 f)

```
ttestreg <- lm(formula = Score ~ dummyBrown, data = ttestData)
```

6.1 g)

```
summary(ttestreg) # p-value dummyBrown: 0.14
```

6.2 a) $H_0: \quad \mu_1 = \mu_2 = \mu_3 = \mu_4$ $\qquad H_1: \quad \mu_1 \neq \mu_2 \neq \mu_3 \neq \mu_4$

6.2 b) $df_M = k - 1 = 4 - 1 = 3$ $\qquad df_R = n - k = 222 - 4 = 218$

6.2 c) Critical F-value: 2.646

```
df1 <- 4 - 1 # 3
df2 <- nrow(dataset8) - 4 # 218
qf(p = 0.95, df1 = df1, df2 = df2) # 2.646
```

6.2 d)

```
anovaResult <- aov(formula = Score ~ Group, data = dataset8)
```

6.2 e) F-value: 2.894 p-value: 0.0362

```
summary(anovaResult)
# F-value: 2.894
# p-value: 0.0362
```

6.2 f) The p-value is 0.0362, which is lower than the 0.05 required to reject H_0. The sample F-value is 2.894, which is higher than the critical F-value of 2.646. H_0 is rejected with 95% confidence. The means of the four groups are not equal to each other. There is a risk of 5% for a type-I error.

6.3 a)

```
dummyBrown <- as.numeric(dataset8$Group == 'Brown') # Brown eyes
dataset8 <- cbind(dataset8, dummyBrown)

dummyBlue <- as.numeric(dataset8$Group == 'Blue') # Blue eyes
dataset8 <- cbind(dataset8, dummyBlue)

dummyGreen <- as.numeric(dataset8$Group == 'Green') # Green eyes
dataset8 <- cbind(dataset8, dummyGreen)
```

6.3 b) anovaReg <- lm(formula = Score ~ dummyBrown + dummyBlue + dummyGreen, data = dataset8)

6.3 c) F-value: 2.894 p-value: 0.0362

```
summary(anovaReg)
# F-value: 2.894
# p-value: 0.0362
# R-squared: 0.0383
```

6.3 d) Yes, the results are the same.

6.4 a)

```
ancovaReg <- lm(formula = Score ~ dummyBrown + dummyBlue +
                          dummyGreen + initialScore,
                data = dataset8)
```

6.4 b) F-value: 2.252 p-value: 0.064

```
summary(ancovaReg)
# F-value: 2.252
# p-value: 0.064
# R-squared: 0.0398
```

6.4 c) The p-value is 0.0645, which is higher than the 0.05 required to reject H_0. H_0 is not rejected with 95% confidence. The means are equal to each other if you consider the initial score as a covariate. There is a 5% change of a type-I error.

6.4 d) The p-value is of the coefficient of initial score is 0.5539, which means that it is not significantly different from zero. This means that the initial score is not a good predictor of the actual score.

6.4 e) R^2 `anovaReg` : 0.0383 R^2 `ancovaReg` : 0.0398

The `ancovaReg` regression model explains more variation in the outcome variable score.

6.4 f) The groups, together with the initial score, explain 3.98% of the variance in the dependent variable score.

6.4 g) AIC `anovaReg` : 865.54 AIC `ancovaReg` : 867.18

```
AIC(anovaReg) # AIC: 865.54
AIC(ancovaReg) # AIC: 867.18
```

6.4 h) The AIC value of the `anovaReg` regression model is the lowest, which means that the `anovaReg` model fits the data better than the `ancovaReg` regression model. This means that the model without the covariate is a better model. You may have already seen this, since the covariate in the `ancovaReg` model was not a good predictor of the score.

6.5 a)

```r
#  Be sure to set your working directory when providing a relative path
load('iowa.RData')
```

6.5 b) The `iowa` data consists of payments made by the state of Iowa. Payments are assigned to fiscal years that run from July 1 through June 30, and are numbered for the calendar year in which they end. The fiscal year is divided into fiscal periods with 1 being July and 12 being June. The fiscal year also includes a hold-over period for payments made after year end for good and services received on or before June 30.

6.5 c) Rows: 12279009 Columns: 22

```r
nrow(iowa) # 12279009 rows
ncol(iowa) # 22 columns
```

6.5 d) Unique services: 8

```r
unique(iowa$Service) # 8 unique services
table(iowa$Service)
```

6.5 e) Service: Human Services Rows: 6682159

6.5 f) Number of rows that show a difference: 2624607

```r
iowa$Payment.Issue.Date <- as.Date(iowa$Payment.Issue.Date,format= '%m/%d/%Y')
iowa$Invoice.Date <- as.Date(iowa$Invoice.Date,format = '%m/%d/%Y')

length(which(iowa$Payment.Issue.Date != iowa$Invoice.Date)) # 2624607
```

6.5 g)

```r
dataDif <- data[which(iowa$Payment.Issue.Date != data$Invoice.Date),]
```

6.5 h)

```r
dataDif$dif.days <- dataDif$Payment.Issue.Date - dataDif$Invoice.Date
dataDif$dif.days <- as.numeric(dataDif$dif.days)
```

6.5 i)

Minimum: −3651
Mean: 21.815
Maximum: 36529
Upper quartile: 33
Lower quartile: 4
Standard deviation: 89.24

```
min(dataDif$dif.days)
max(dataDif$dif.days)
mean(dataDif$dif.days)
quantile(dataDif$dif.days)
sd(dataDif$dif.days)
```

6.5 j) The default histogram does not provide much information due to the fact that R specifies a very wide *x-axis*.

```
hist(dataDif$dif.days)
```

6.5 k)

```
hist(dataDif$dif.days[dataDif$dif.days > quantile(dataDif$dif.days, 0.05) &
      dataDif$dif.days < quantile(dataDif$dif.days, 0.95)], breaks = 100)
```

6.5 l)

```
dataDif2 <- dataDif[(dataDif$dif.days > (-1)) & (dataDif$dif.days <= 365),]
```

6.5 m)

```
plot(dataDif2$Amount,dataDif2$dif.days)
```

6.5 n) Correlation: −0.0025

```
cor.test(dataDif2$Amount,dataDif2$dif.days)
```

6.5 o) The p-value of the correlation test against the value zero is 3.148e-05, which is sufficient enough to reject H_0 with 95% confidence. This implies that there is, with 95% certainty, a correlation between the the time between invoice and payment, and the amount that is paid.

6.5 p) Administration and regulation: 26.452
Agriculture and natural resources: 28.275
Capital: 35.490
Economic development: 25.815
Education: 23.868
Human services: 18.508
Justice system: 25.478

```
aggregate(dataDif2$dif.days, by = list(dataDif2$Service), FUN = mean)
```

6.5 q) p-value: < 2e-16

Conclusion: The p-value is lower than 0.05, so you can reject H_0 with 95% confidence. This means that the means of all expense categories are not equal. There is a 5% type change of a type-I error.

```
aovRes <- aov(dif.days ~ Service, data = dataDif2)
summary(aovRes)
```

6.5 r) All means, except for the means of the justice system expenses and the economic development expenses, show a p-value below 5% and can thus be regarded to differ from each other.

```
tukeyRes <- TukeyHSD(aovRes)
```

6.5 s) An ANOVA assumes the dependent variable to be continuous. Some examples of appropriate analyses could be:

```
# Poisson regression and then interpret the predictors
poissonReg <- glm(dif.days ~ Expense.Category, family = poisson, data = dataDif2)
summary(poissonReg)

# Kruskal Wallis test and Dunn test to compare individual groups
kruskRes <- kruskal.test(dif.days ~ Service, data = dataDif2)
# install.packages('FSA'); library(FSA)
dunn.res <-dunnTest(dataDif2$dif.days, dataDif2$Service)
```

6.5 t) It is a bad idea, the p-value is affected by the number of samples. The higher the sample, the lower the p-value gets. In other words, p-values lose their meaning quite quickly (unless they are non-significant).