TMA4315: Compulsory exercise 1 (title)

Group 0: Name1, Name2 (subtitle)

23.10.2018

Contents

Part 1

Bold

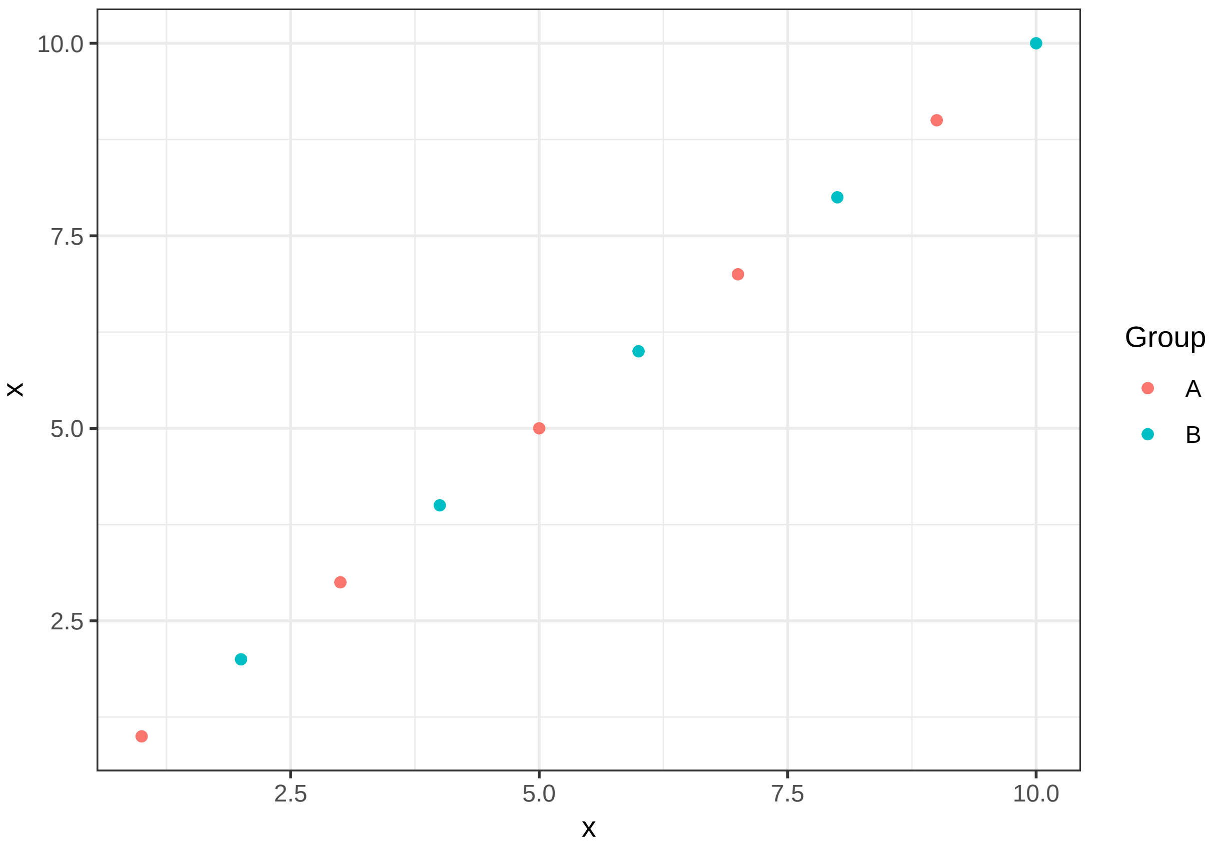
italic

To get a pdf file, make comments of the lines with the “html_document” information, and make the lines with the “pdf_document” information regular, and vice versa.

a)

Your answer for part 1a)

```r
# some R code for part 1a)
library(ggplot2)
ggplot(data.frame(x = 1:10, Group = rep(c("A", "B"), 5)), aes(x = x,
    y = x, col = Group)) + geom_point() + theme_bw()
```



The following is a numbered list:

1. First
2. Second
3. Third

And this is an unnumbered list:

- GLM
- rocks

Equations can be made like this:

$$\mathbf{Y} = \mathbf{X}\beta + \varepsilon$$

Part 2

a)

According to the χ^2-test performed by the author, there is no evidence against the assumption of independence between goals scored by the home and the away team.

```
filepath <- "https://www.math.ntnu.no/emner/TMA4315/2018h/eliteserien2018"
eliteserie <- read.table(file = filepath, header = TRUE, colClasses = c("character",
    "character", "numeric", "numeric"))
contigency_table <- table(eliteserie$ya, eliteserie$yh)
contigency_table
```

```
##
##      0  1  2  3  4  5
##   0  8 19 10 13  6  2
##   1 18 26 14 10  5  2
##   2  3 15 13  7  3  0
##   3  1  5  4  2  1  0
##   4  1  2  1  0  0  0
##   6  0  1  0  0  0  0
```

```
chisq.test(contigency_table)
```

```
##
##  Pearson's Chi-squared test
##
## data:  contigency_table
## X-squared = 16.387, df = 25, p-value = 0.9028
```

As we can see in our χ^2-test, the p-value (0.9028) is high, which supports the H_0-hypoteses that the goals scored by the home and the away team is independent. We would have to consider some dependency if the p-value had been under our siginficance value at 0.05.

b)

```
calculate_points <- function(data_eliteserie) {
    result_table <- data.frame(team <- unique(data_eliteserie$home),
        position <- seq(1, length(team), by = 1), goal_for <- rep(0,
            length(team)), goal_against <- rep(0, length(team)), goal_score <- rep(0,
            length(team)), points <- rep(0, length(team)))
    colnames(result_table) <- c("Team", "Position", "GF", "GA", "GD",
        "Points")
    for (i in 1:length(data_eliteserie$home)) {
        if (data_eliteserie$yh[i] > data_eliteserie$ya[i]) {
            index = which(result_table$Team == data_eliteserie$home[i])
            result_table[index, ]$Points = result_table[index, ]$Points +
                3
        } else if (data_eliteserie$yh[i] < data_eliteserie$ya[i]) {
            index = which(result_table$Team == data_eliteserie$away[i])
            result_table[index, ]$Points = result_table[index, ]$Points +
                3
        } else {
            result_table[which(result_table$Team == data_eliteserie$home[i]),
                ]$Points = result_table[which(result_table$Team == data_eliteserie$home[i]),
                ]$Points + 1
            result_table[which(result_table$Team == data_eliteserie$away[i]),
                ]$Points = result_table[which(result_table$Team == data_eliteserie$away[i]),
                ]$Points + 1
        }
        result_table[which(result_table$Team == data_eliteserie$home[i]),
            ]$GF = result_table[which(result_table$Team == data_eliteserie$home[i]),
            ]$GF + data_eliteserie$yh[i]
        result_table[which(result_table$Team == data_eliteserie$home[i]),
            ]$GA = result_table[which(result_table$Team == data_eliteserie$home[i]),
            ]$GA + data_eliteserie$ya[i]
        result_table[which(result_table$Team == data_eliteserie$away[i]),
            ]$GF = result_table[which(result_table$Team == data_eliteserie$away[i]),
            ]$GF + data_eliteserie$ya[i]
        result_table[which(result_table$Team == data_eliteserie$away[i]),
            ]$GA = result_table[which(result_table$Team == data_eliteserie$away[i]),
            ]$GA + data_eliteserie$yh[i]
    }
    result_table$GD <- result_table$GF - result_table$GA
    ordered_table <- result_table[order(-result_table$Points, -result_table$GD),
        ]
    i <- 1
    for (team in ordered_table$Team) {
        result_table[which(result_table$Team == team), ]$Position <- i
        i <- i + 1
    }
    return(result_table)
}
result_table <- calculate_points(eliteserie)
result_table <- result_table[order(-result_table$Points, -result_table$GD),
    ]
print(result_table)
```

```
##                  Team Position GF GA  GD Points
## 10          Rosenborg        1 43 20  23     52
## 11              Brann        2 36 23  13     48
## 1               Molde        3 48 30  18     43
## 12          Haugesund        4 36 28   8     41
## 8          Ranheim_TF        5 38 40  -2     38
## 13         Vaalerenga        6 35 37  -2     36
## 3                 Odd        7 35 29   6     34
## 14            Tromsoe        8 35 33   2     33
## 6         Sarpsborg08        9 39 34   5     32
## 7        Kristiansund       10 32 35  -3     31
## 4          BodoeGlimt       11 28 30  -2     27
## 2       Stroemsgodset       12 38 38   0     26
## 9         Lillestroem       13 26 37 -11     25
## 16            Stabaek       14 29 43 -14     23
## 5               Start       15 24 42 -18     23
## 15 Sandefjord_Fotball       16 24 47 -23     15
```

Here is a the table ordered after points, where GF is goales scored, GA is goals against and GD is the goal balance.

c

```
library(myglm)
goals <- c(eliteserie$yh, eliteserie$ya)
X <- matrix(data = 0, nrow = 384, ncol = 17)
colnames(X) <- c("Intercept", "HomeAdvantage", unique(eliteserie$home)[-4])
for (i in 1:length(eliteserie$home)) {
    X[i, 1] = 1
    X[i, 2] = 1
    home_index <- which(colnames(X) == eliteserie$home[i])
    away_index <- which(colnames(X) == eliteserie$away[i])
    X[i, home_index] <- 1
    X[i, away_index] <- -1
```

```r
}
for (i in 1:length(eliteserie$away)) {
    X[i + length(eliteserie$home), 1] = 1
    home_index <- which(colnames(X) == eliteserie$home[i])
    away_index <- which(colnames(X) == eliteserie$away[i])
    X[i + length(eliteserie$home), home_index] <- -1
    X[i + length(eliteserie$home), away_index] <- 1
}
strength_param <- myglm(goals ~ -1 + X)
# strength_param <- strength_param[order(-strength_param)]
names(strength_param) <- substring(names(strength_param), 2)
strength_param
```

```
##          Intercept      HomeAdvantage              Molde
##        0.100321807        0.402062206        0.279399199
##      Stroemsgodset                Odd              Start
##        0.049792126        0.100120614       -0.225757649
##        Sarpsborg08       Kristiansund         Ranheim_TF
##        0.097677349        0.012552907        0.008502727
##        Lillestroem          Rosenborg              Brann
##       -0.132621109        0.367125310        0.225775206
##          Haugesund         Vaalerenga            Tromsoe
##        0.141301460        0.014730410        0.060581141
## Sandefjord_Fotball            Stabaek
##       -0.291683130       -0.147940567
```

```r
glm_betas <- glm(goals ~ -1 + X, family = "poisson")$coefficients
# glm_betas <- glm_betas[order(-glm_betas)]
glm_betas
```

```
##          XIntercept      XHomeAdvantage              XMolde
##         0.100304451         0.402067992         0.279263635
##      XStroemsgodset                XOdd              XStart
##         0.049656842         0.099974799        -0.225883539
##        XSarpsborg08       XKristiansund         XRanheim_TF
##         0.097552927         0.012376355         0.008342814
##        XLillestroem          XRosenborg              XBrann
##        -0.132857461         0.366955583         0.225678120
##          XHaugesund         XVaalerenga            XTromsoe
##         0.141120567         0.014465238         0.060348164
## XSandefjord_Fotball            XStabaek
##        -0.291864636        -0.148046737
```

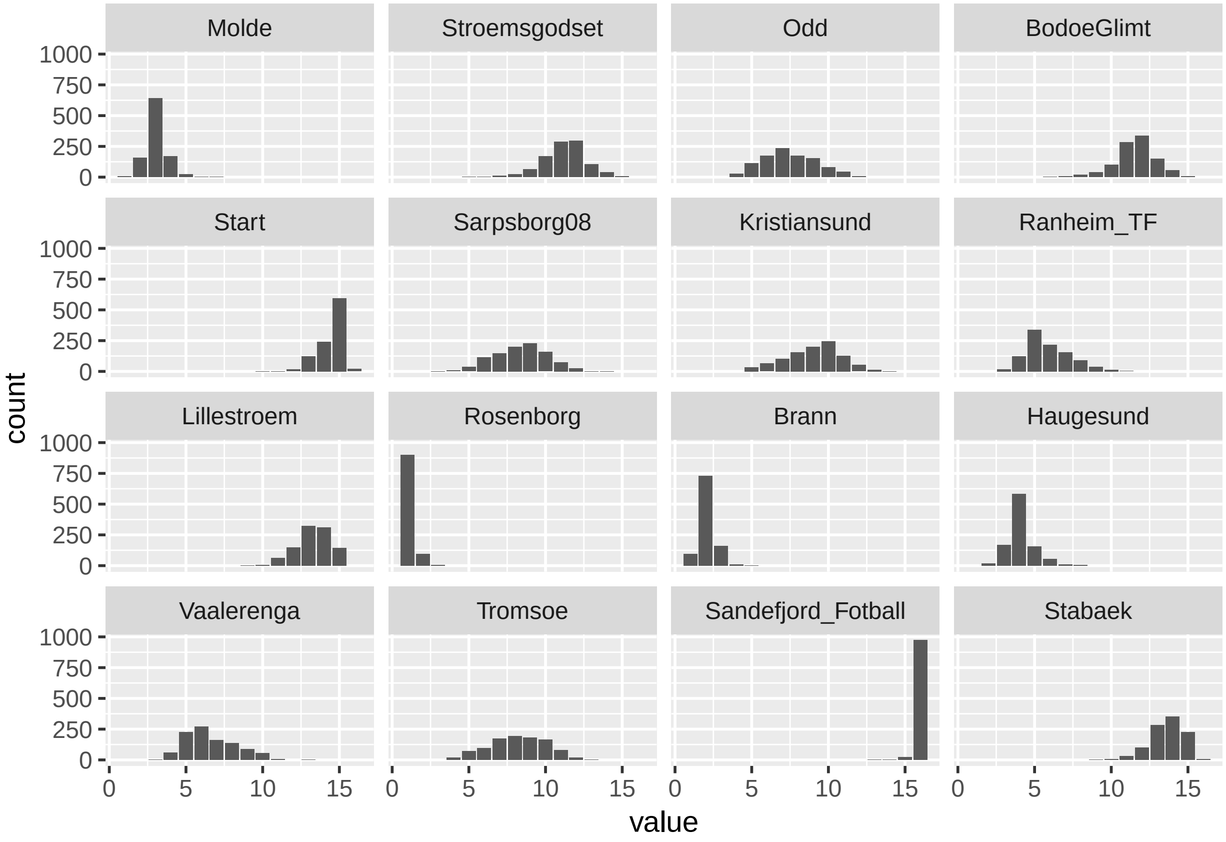
```r
library("reshape2")
set.seed(42)
filepath <- "https://www.math.ntnu.no/emner/TMA4315/2018h/unplayed2018"
eliteserie_unplayed <- read.table(file = filepath, header = TRUE, colClasses = c("character",
    "character"))
simulate_season_end <- function(data_unplayed_matches, strength_param) {
    for (i in 1:length(data_unplayed_matches$home)) {
        intercept <- strength_param["Intercept"]
        home_advantage <- strength_param["HomeAdvantage"]
        strength_hometeam <- strength_param[data_unplayed_matches$home[i]]
        strength_awayteam <- strength_param[data_unplayed_matches$away[i]]
        if (data_unplayed_matches$home[i] == "BodoeGlimt") {
            strength_hometeam <- 0
        } else if (data_unplayed_matches$away[i] == "BodoeGlimt") {
            strength_awayteam <- 0
        }
        data_unplayed_matches$yh[i] <- rpois(1, exp(strength_hometeam +
            intercept + home_advantage - strength_awayteam))
        data_unplayed_matches$ya[i] <- rpois(1, exp(strength_awayteam +
            intercept - strength_hometeam))
    }
    return(data_unplayed_matches)
}

Points <- c()
for (i in 1:1000) {
    simulated_results <- simulate_season_end(eliteserie_unplayed, strength_param)
    eliteserie_finished <- rbind(eliteserie, simulated_results)
    standings <- calculate_points(eliteserie_finished)
    write.table(standings, paste("Standings\\Standings_", toString(i),
        ".txt", sep = ""), sep = "\t", quote = FALSE)
}
```

```
library("ggplot2")
library("reshape2")
n <- 1000
teams <- unique(eliteserie$home)
Positions <- matrix(0, nrow = length(teams), ncol = n)
Points <- matrix(0, nrow = length(teams), ncol = n)
rownames(Positions) <- c(teams)
rownames(Points) <- c(teams)
for (i in 1:n) {
    standings <- read.table(file = paste("Standings\\Standings_", toString(i),
        ".txt", sep = ""), header = TRUE, colClasses = c("numeric", "character",
        "numeric", "numeric", "numeric", "numeric"))
    Positions[, i] <- standings$Position
    Points[, i] <- standings$Points
}
Positions <- melt(Positions, id.vars = c("Teams"))
Mean_points <- rowMeans(Points)
Mean_points
```

```
##              Molde      Stroemsgodset                Odd
##             53.887             35.304             42.338
##         BodoeGlimt              Start        Sarpsborg08
##             34.802             27.855             40.696
##       Kristiansund         Ranheim_TF        Lillestroem
##             39.994             46.089             30.629
##          Rosenborg              Brann          Haugesund
##             64.588             58.814             50.239
##         Vaalerenga            Tromsoe Sandefjord_Fotball
##             44.477             41.268             19.565
##            Stabaek
##             29.785
```

```
p <- ggplot(data = data.frame(Positions), aes(x = value)) + geom_bar() +
    facet_wrap(~Var1)
p
```


Molde
Stroemsgodset
Odd
BodoeGlimt
Start
Sarpsborg08
Kristiansund
Ranheim_TF
Lillestroem
Rosenborg
Brann
Haugesund
Vaalerenga
Tromsoe
Sandefjord_Fotball
Stabaek
count
value