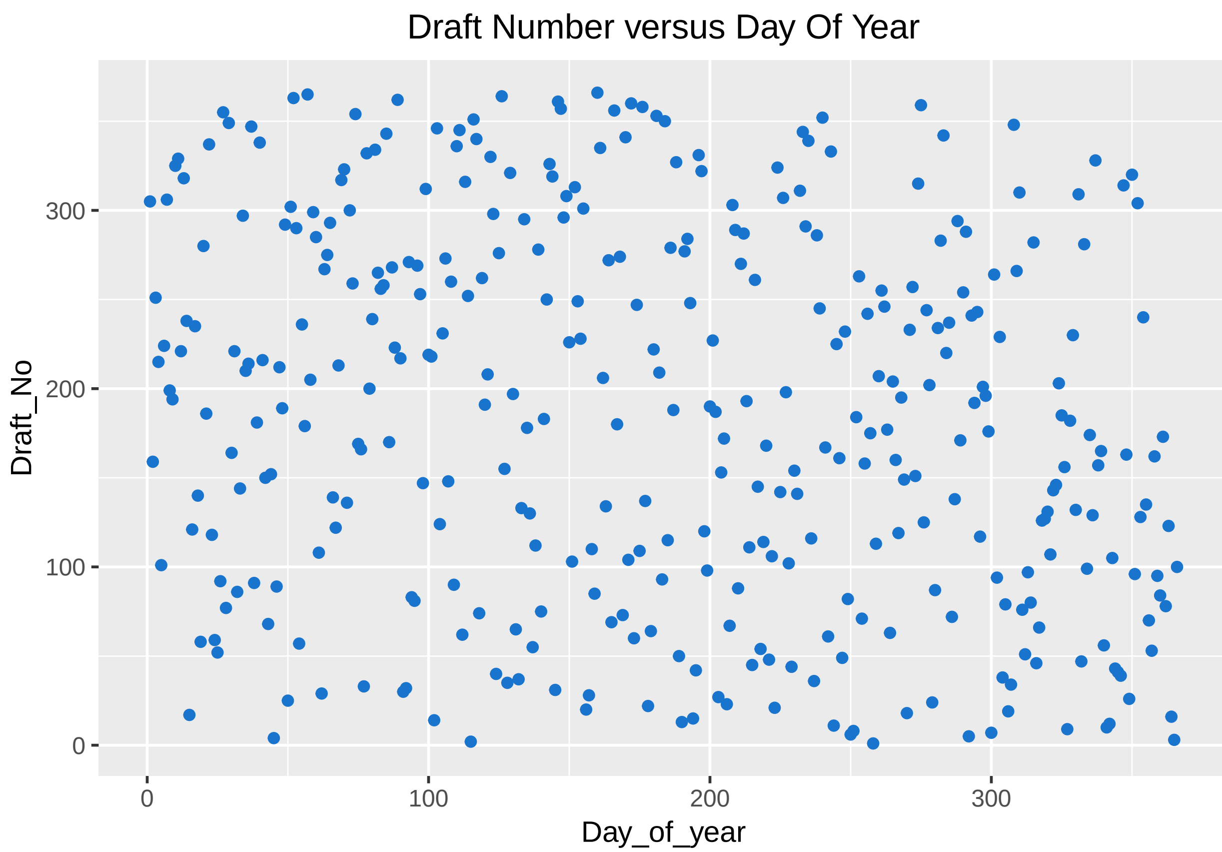
Computational Statistics Computer Lab 5

Nahid Farazmand (nahfa911), Aashana Nijhawan (aasni448)

February 19, 2019

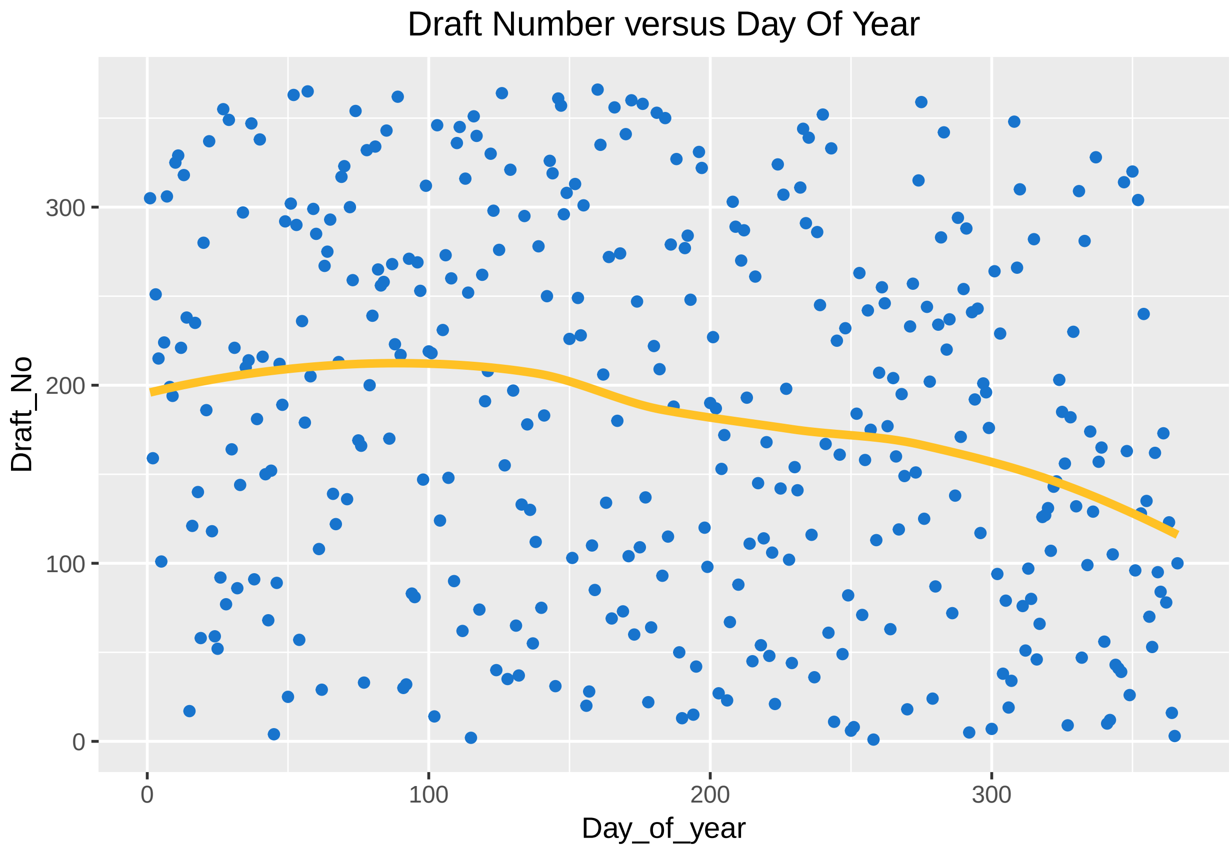
Question 1: Hypothesis testing

1.1. Scatterplot of Draft Number versus Day Of Year



Based on this plot there is no relation between Draft number and Day of year and the lottery looks random.

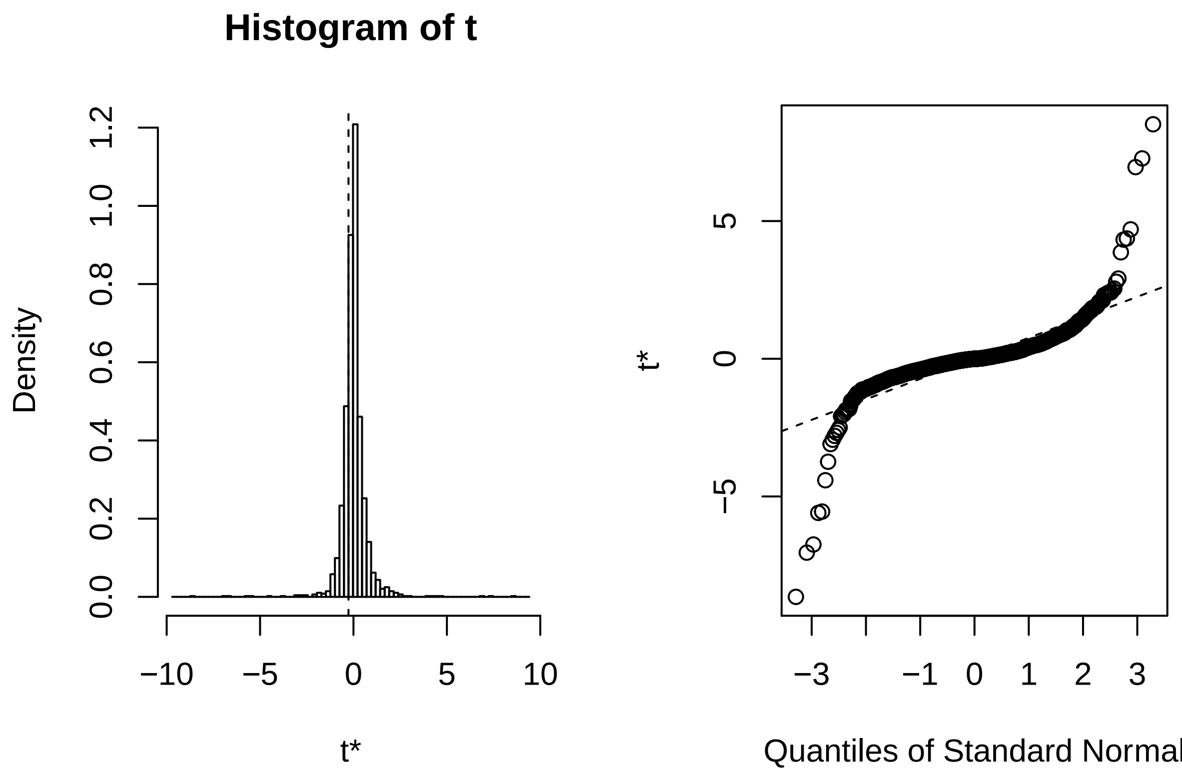
1.2. Curve of estimated Draft_No versus Day_Of_Year



1.3. Estimating the distribution of T by using a non-parametric bootstrap with B = 2000

$T = \frac{\hat{Y}(X_b)-\hat{Y}(X_a)}{X_b-X_a}$ Where $X_b = argmax_x Y(X)$ and $X_a = argmin_x Y(X)$

NOTE: If this value is signifcantly greater than zero, then there should be a trend in the data and the lottery is not random.



P-Value:

```
## [1] 0.453
```

By looking at the histogram of T we can obviously see that most of the cases are around zero. Besides the pi-value >> 0.05 which means we cannot reject the null hypothesis; so based on our test the lottery is random.

1.4. Permutation test with statistics T

```
## [1] 0.4575
```

Based on the result of permutation test, again, we can say that the lottery is random.

1.5. Crude estimation of power

At first we Generate data samples that satisfy H_1 (obviously non-random datasets). And then we implement our test for all datasets. Then we calculate the proportion of cases that our test reject H_0 (Where p-value is less than 0.05).

```
##     alpha p_value
## 1     0.1   0.015
## 2     0.2   0.025
## 3     0.3   0.015
## 4     0.4   0.000
## 5     0.5   0.000
```

```
## 6     0.6   0.005
## 7     0.7   0.000
## 8     0.8   0.000
## 9     0.9   0.000
## 10    1.0   0.000
## 11    1.1   0.000
## 12    1.2   0.000
## 13    1.3   0.000
## 14    1.4   0.005
## 15    1.5   0.000
## 16    1.6   0.000
## 17    1.7   0.000
## 18    1.8   0.000
## 19    1.9   0.000
## 20    2.0   0.005
## 21    2.1   0.000
## 22    2.2   0.005
## 23    2.3   0.000
## 24    2.4   0.000
## 25    2.5   0.000
## 26    2.6   0.005
## 27    2.7   0.005
## 28    2.8   0.005
## 29    2.9   0.000
## 30    3.0   0.005
## 31    3.1   0.005
## 32    3.2   0.010
## 33    3.3   0.005
## 34    3.4   0.005
## 35    3.5   0.000
## 36    3.6   0.010
## 37    3.7   0.010
## 38    3.8   0.000
## 39    3.9   0.005
## 40    4.0   0.000
## 41    4.1   0.005
## 42    4.2   0.000
## 43    4.3   0.005
## 44    4.4   0.010
## 45    4.5   0.000
## 46    4.6   0.015
## 47    4.7   0.005
## 48    4.8   0.005
## 49    4.9   0.000
## 50    5.0   0.000
## 51    5.1   0.005
## 52    5.2   0.010
## 53    5.3   0.005
## 54    5.4   0.010
## 55    5.5   0.010
## 56    5.6   0.015
## 57    5.7   0.005
## 58    5.8   0.005
## 59    5.9   0.010
```

```
## 60    6.0   0.005
## 61    6.1   0.000
## 62    6.2   0.010
## 63    6.3   0.000
## 64    6.4   0.005
## 65    6.5   0.005
## 66    6.6   0.000
## 67    6.7   0.005
## 68    6.8   0.010
## 69    6.9   0.015
## 70    7.0   0.025
## 71    7.1   0.015
## 72    7.2   0.020
## 73    7.3   0.005
## 74    7.4   0.005
## 75    7.5   0.005
## 76    7.6   0.020
## 77    7.7   0.005
## 78    7.8   0.010
## 79    7.9   0.005
## 80    8.0   0.015
## 81    8.1   0.025
## 82    8.2   0.005
## 83    8.3   0.060
## 84    8.4   0.025
## 85    8.5   0.015
## 86    8.6   0.015
## 87    8.7   0.010
## 88    8.8   0.005
## 89    8.9   0.015
## 90    9.0   0.005
## 91    9.1   0.025
## 92    9.2   0.030
## 93    9.3   0.020
## 94    9.4   0.045
## 95    9.5   0.010
## 96    9.6   0.015
## 97    9.7   0.015
## 98    9.8   0.035
## 99    9.9   0.020
## 100  10.0   0.010
```

Percent of correct rejections

```
## power = 99%
```

As we can see power is high so the quality of our test statistics is good.

Appendix

```
#--- Question 1:
#-- 1.1
```

```r
library(readxl)
data <- read_excel('lottery.xls',sheet = 1)

library(ggplot2)
ggplot(data = data)+
geom_point(mapping = aes(x = Day_of_year,y = Draft_No),color = 'dodgerblue3') +
ggtitle('Draft Number versus Day Of Year') +
theme(plot.title = element_text(hjust = 0.5))
#-- 1.2
set.seed(123456)
model <- loess(formula = Draft_No~Day_of_year
               , data = data)
Y_hat <- predict(object = model, newdata = data)
data2 <- data.frame(Y_hat = Y_hat, Day_of_year = data$Day_of_year)
ggplot()+
geom_point(data = data, mapping = aes(x = Day_of_year,y = Draft_No)
           ,color = 'dodgerblue3') +
geom_line(data = data2,mapping = aes(x = Day_of_year,y = Y_hat)
          ,color = 'goldenrod1'
          ,size = 1.5) +
ggtitle('Draft Number versus Day Of Year') +
theme(plot.title = element_text(hjust = 0.5))

#-- 1.3.
library(boot)
stat1 <- function(data,vn){
  data <- as.data.frame(data[vn,])
  X_b <- data[which.max(data$Draft_No),]$Day_of_year
  X_a <- data[which.min(data$Draft_No),]$Day_of_year
  model <- loess(formula = Draft_No~Day_of_year
                 , data = data)
  Y_hat <- predict(object = model, newdata = data)
  Y_hat_a <- Y_hat[X_a]
  Y_hat_b <- Y_hat[X_b]
  (Y_hat_b - Y_hat_a)/(X_b - X_a)
}

res <- boot(data = data, statistic = stat1
                        # Ordinary nonparametric bootstrap (default)
            , R = 2000, sim = "ordinary")


plot(res)
# P-Value
mean(abs(res$t0) < abs(res$t-mean(res$t)))
#-- 1.4. Permutation test
P_per_test <- function(data,R){
  res <- boot(data = data, statistic = stat1, R = R)
  X_b <- data[which.max(data$Draft_No),]$Day_of_year
  X_a <- data[which.min(data$Draft_No),]$Day_of_year
  model <- loess(formula = Draft_No~Day_of_year
                 , data = data)
  Y_hat <- predict(object = model, newdata = data)
```

```r
  Y_hat_a <- Y_hat[X_a]
  Y_hat_b <- Y_hat[X_b]
  T_x <- (Y_hat_b - Y_hat_a)/(X_b - X_a)
               # two-sided test
  length(which(abs(res$t) > abs(T_x)))/R
}

P_per_test(data,2000)

#-- 1.5. Estimate of the power
library(matrixStats)
X <- data$Day_of_year
alph <- seq(0.1,10,0.1)
output <- data.frame(alpha = 0,p_value = 0)

for(i in 1:length(alph)){
  Bet <- rnorm(n = length(X),mean = 183,sd = 10)
  Y <- rowMaxs(cbind(matrix(rowMins(cbind(matrix(alph[i]*X+Bet,ncol = 1),366))
                            ,ncol = 1),0))
  data5 <- data.frame(Day_of_year = X, Draft_No = Y)
  output[i,] <- c(alph[i],P_per_test(data5,200))
}

output
power = length(which(output$p_value < 0.05))/length(alph)
cat(paste0('power = ',power*100,'%'))
```