

<b>Autor:</b>	Radosław Smoter
<b>Numer sprawozdania:</b>	3
<b>Data oddania:</b>	14.05.2022
<b>Prowadzący:</b>	dr inż. Krzysztof Schiff

# Probabilistyka w zastosowaniach technicznych

# 1 Kody

- 1 Załaduj plik FacebookNarcissism.dat z katalogu korzystając z funkcji read.delim().

```
# Załadowanie tabeli.
facebookData <- read.delim("FacebookNarcissism.dat", header = T);
```

- 2 Korzystając z funkcji ggplot() utwórz obiekt graficzny graph z pliku załadowanego wcześniej. Użyj aes(NPAC\_R\_Total, Rating) w celu wyznaczenia osi x i y, a następnie dodaj wizualizację korzystając z funkcji geom\_point(). Wyświetlenie rysunku otrzymasz używając funkcji graph+geom\_point().

```
# Stworzenie grafu z odpowiednimi osiami X, Y.
graph <- ggplot(facebookData, aes(NPAC_R_Total, Rating));
#Utworzenie punktów na grafie.
graph + geom_point();
```

- 3 Wykonaj:

- 3.1 examData<-read.delim(„Exam Anxiety.dat”, header=TRUE)
- 3.2 scatter<-ggplot(examData, aes(Anxiety, Exam, colour=Gender))
- 3.3 scatter+geom\_point+geom\_smooth(method="lm")
- 3.4 scatter+geompoint()
   
+geom\_smooth(method="lm",aes(fill=Gender),alpha=0.1) +labs(x="Exam
   
Anxiety",y="Exam performance%", colour="Gender")

```
# Wczytaj dane.
examData <- read.delim("Exam Anxiety.dat", header = T);
# Utworź graf, kolor wyników zależy od wartości z nagłówka Gender.
scatter <- ggplot(examData, aes(Anxiety, Exam, color = Gender));
# Wyświetl graf wraz z punktami oraz jako zakres określony liniami
ciągłymi, w postaci liniowej. Kolor wypełnienia niech zależy od wartości
Gender, przezroczystość równa 0,1.
scatter + geom_point() + geom_smooth(method = "lm", aes(fill = Gender),
alpha = 0.1);
```

- 4 Wczytaj plik DownloadFestival.dat do obiektu festivaldata

- 4.1 festivalHistogram<-ggplot(festivalData, aes(day1))
   
+opts(legend.position="none")
- 4.2 festivalHistogram+geom\_histogram(binwidth=0.4)+labs(x="Hygiene
   
(Day 1 of festival)", y="Frequency")

```
# Wczytaj dane.
festivalData <- read.delim("DownloadFestival.dat", header = T);
# Stwórz histogram dla osi day1, bez legendy.
festivalHistogram <- ggplot(festivalData, aes(day1)) +
theme(legend.position = "none");
```

```
# Wyświetl histogram z daną szerokością oraz opisami.
festivalHistogram + geom_histogram(binwidth = 0.4) + labs(x = "Hygiene
Day 1 of the festival", y = "Frequency");
```

## 5 Korzystając z ggplot() oraz geom\_boxplot() na obiekcie festivalData.

```
# Stwórz graf z osiami gender (X) i day1 (Y)
festivalBoxplot <- ggplot(festivalData, aes(gender, day1));
# Wyświetl jako wykres pudełkowy z zadanymi opisami osi
festivalBoxplot + geom_boxplot() + labs(x = "Gender", y = "Hygiene(Day 1
of the festival)");
```

## 6 Utwórz obiekt density<-ggplot(festivalData, aes(day1))

### 6.1 następnie density+geom\_density()+labs(x="Hygiene (Day 1 of Festival)", y="Density Estimate")

```
# Wykres gęstości prawdopodobieństwa (rozkładu prawdopodobieństwa)
density <- ggplot(festivalData, aes(day1));
# Wyświetl wykres gęstości prawdopodobieństwa z zadanymi opisami osi
density + geom_density() + labs(x = "Hygiene(Day 1 of the festival)", y =
"Density Estimate");
```

## 7 chickFlick<-read.delim("ChickFlick.dat",header=TRUE)

### 7.1 bar<-ggplot(chickFlick, aes(film,arousal))

### 7.2 bar+stat\_summary(fun.y=mean, geom="bar", fill="White", colour="Black")+stat\_summary(fun.data=mean\_cl\_normal, geom="pointrange")+ labs(x="Film", y="Mean Arousal")

```
# Załaduj dane.
chickFlick <- read.delim("ChickFlick.dat", header = T);
# Utwórz graf z danymi osiami.
bar <- ggplot(chickFlick, aes(film, arousal));
# Wyświetl graf jako wykres słupkowy, z białym wypełnieniem.
bar + stat_summary(fun = "mean", geom = "bar", fill = "White", color =
"Black") + labs(x = "Film", y = "Mean Arousal");
# Funkcja stat_summary() z argumentem fun.data = "mean_cl_normal" nie
działa.
```

## 8 bar<-ggplot(chickFlick, aes(film, arousal, fill=gender))

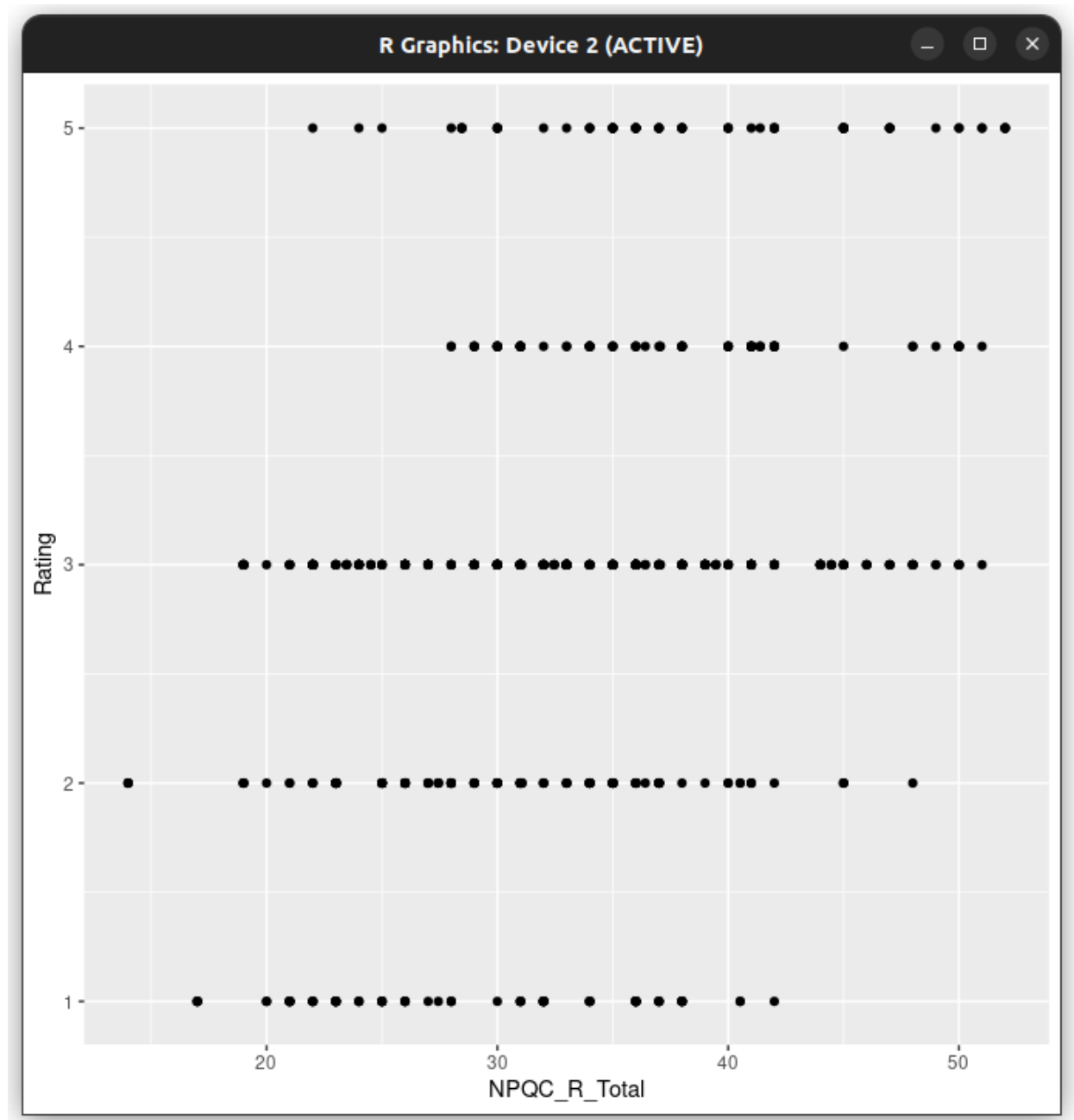
### 8.1 bar+stat\_summary(fun.y=mean, geom="bar", position="dodge")+stat\_summary(fun.data=mean\_cl\_normal, geom="errorbar", position=position\_dodge(width=0.9), width=0.2)+labs(x="Film", y="Mean Arousal", fill="Gender")

```
# Stwórz wykres słupkowy dla takich kategorii.
bar <- ggplot(chickFlick, aes(film, arousal, fill = gender));
# Wyświetl wykres z zadanymi parametrami.
bar + stat_summary(fun = "mean", geom = "bar", position = "dodge") +
stat_summary(fun = "mean", geom = "errorbar", position =
```

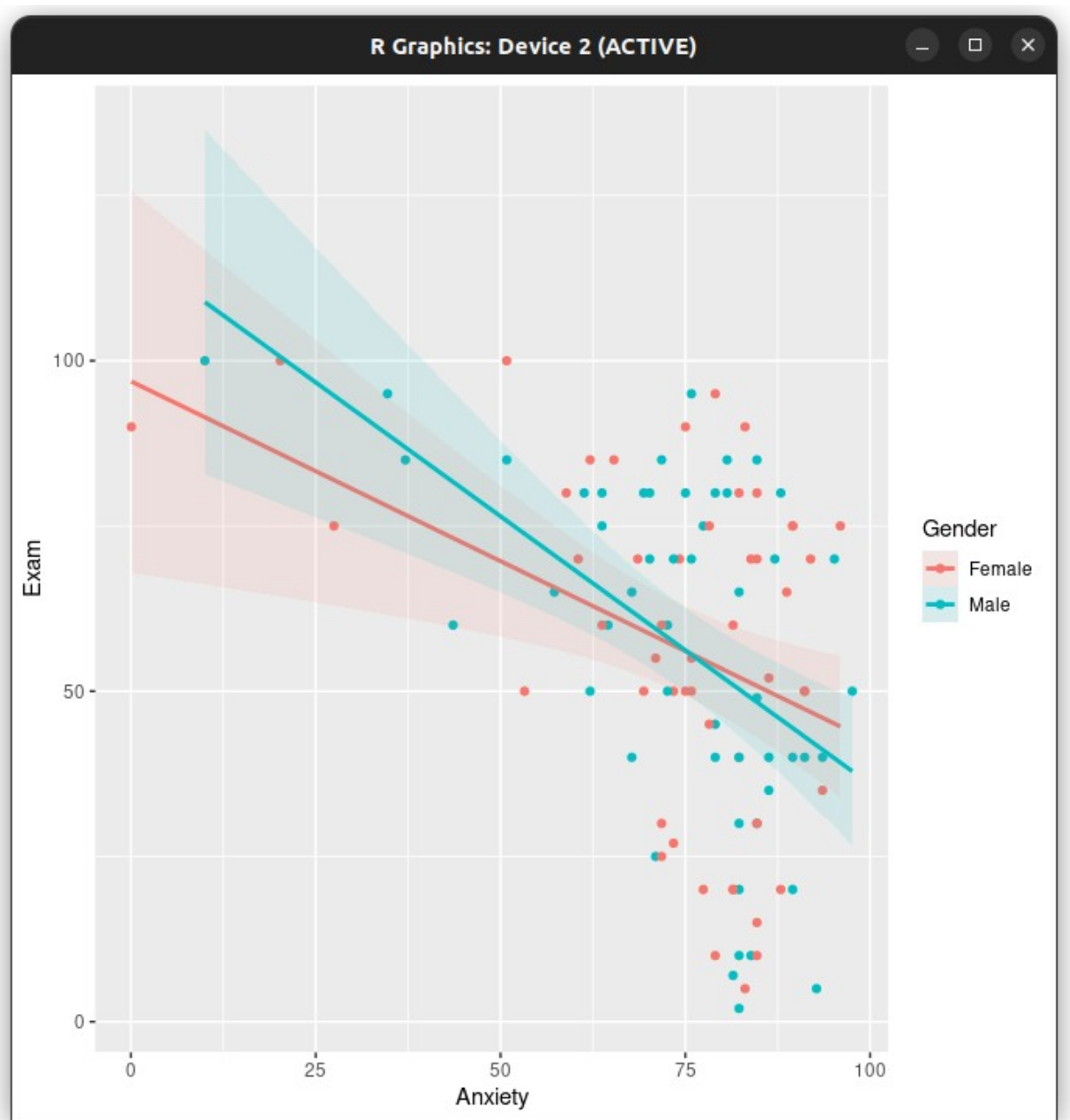
```
position_dodge(width = 0.9), width = 0.2) + labs(x = "Film", y = "Mean
Arousal", fill = "Gender");
```

## 2 Wyniki

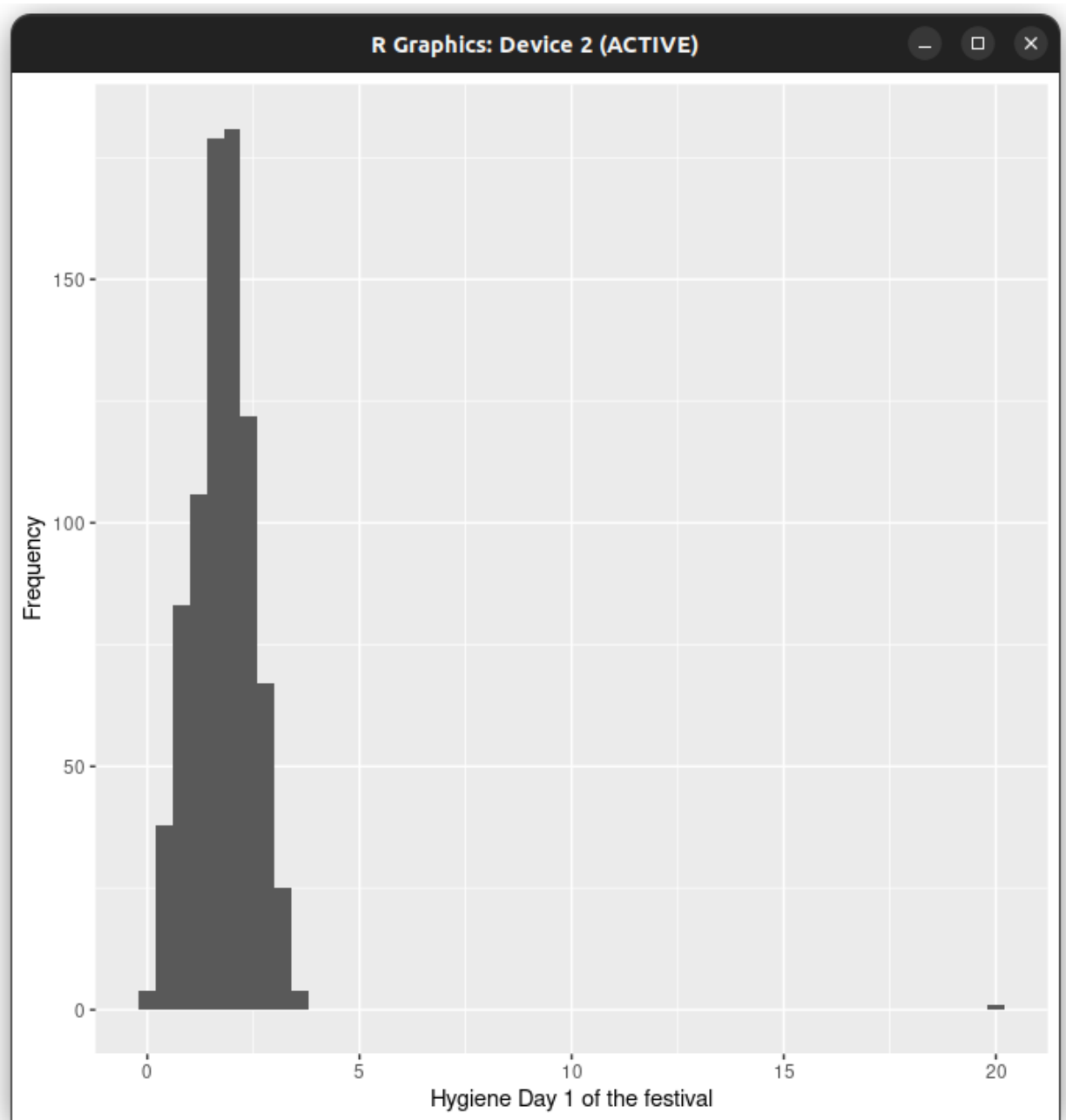
- 1.
- 2.



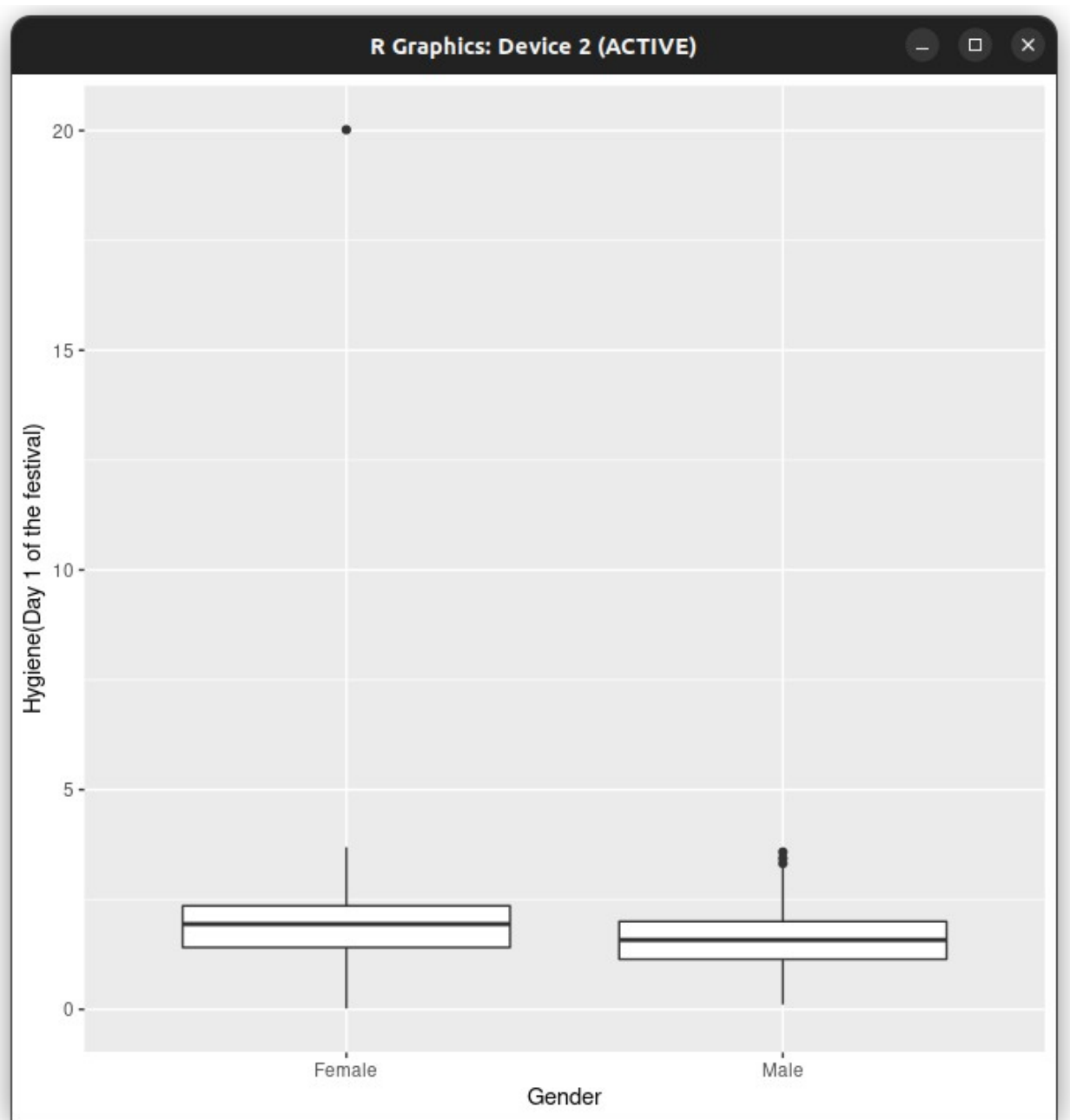
3.



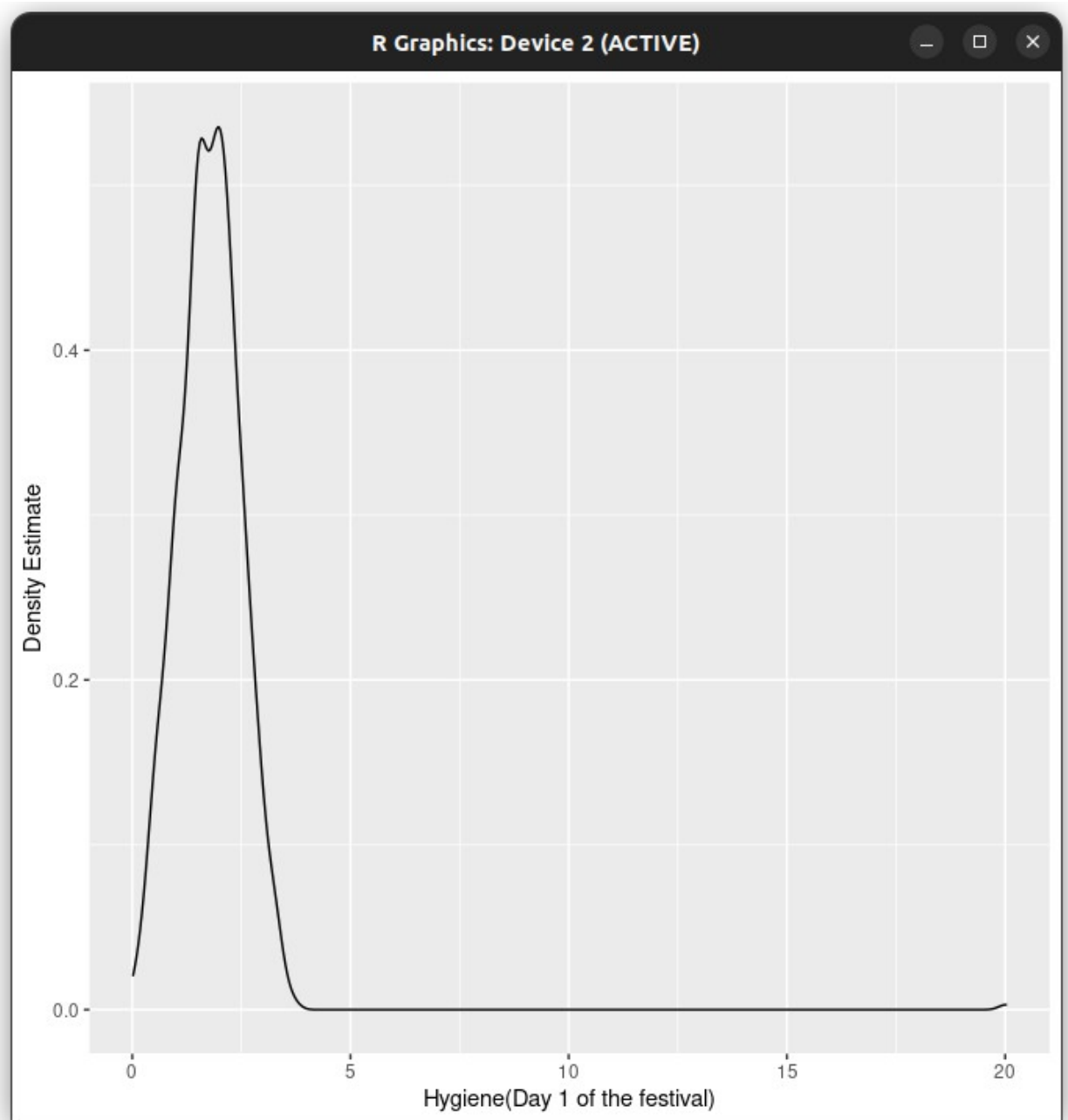
4.



5.

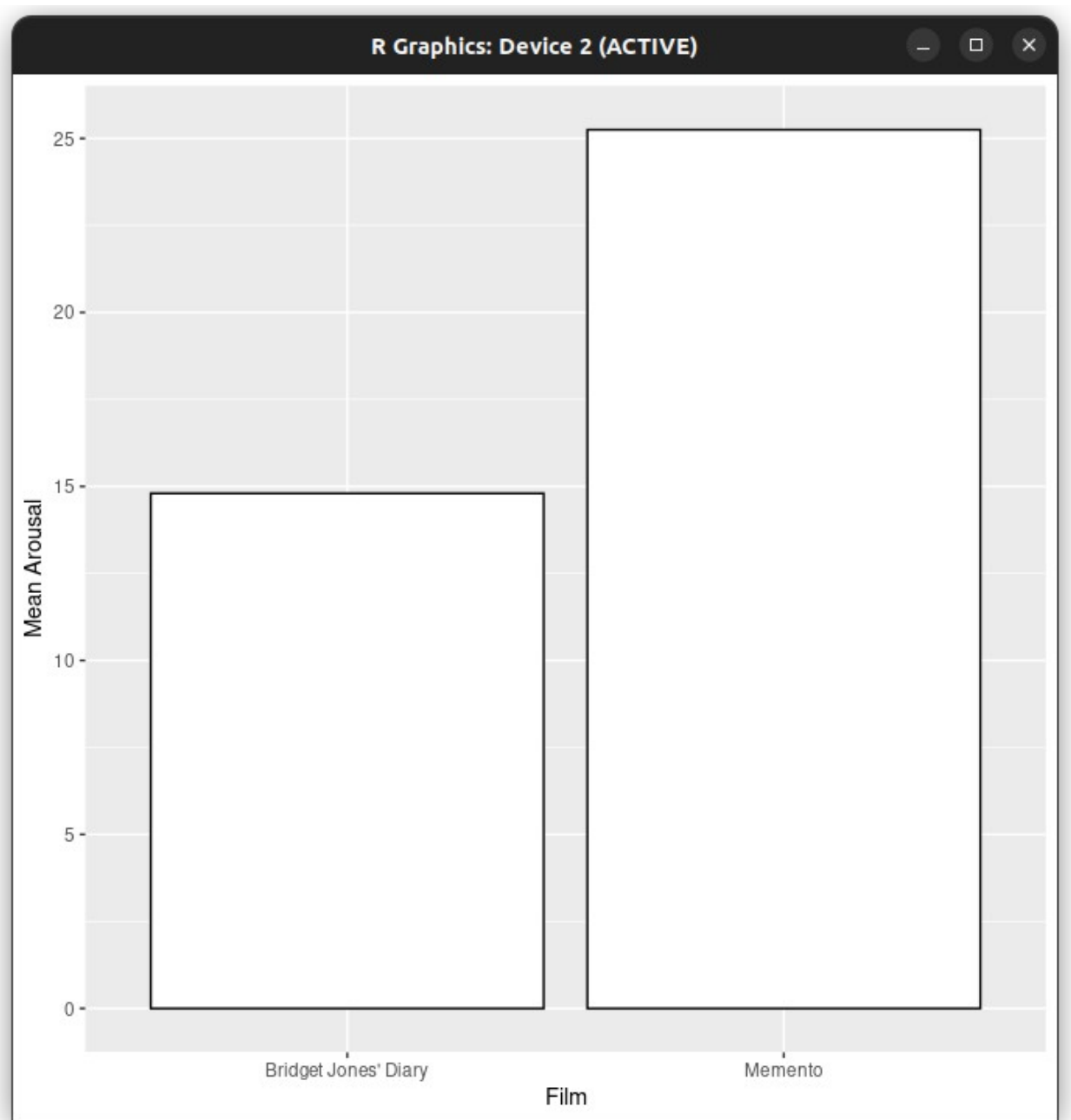


6.





7.



8.

