ECO394D Probability and Statistics Homework 1

Sibo Ding

Summer 2023

```
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(ggplot2)
```

Problem 1

Part A

```
##           david.bowie
## daft.punk     0     1
##         0 0.925 0.912
##         1 0.075 0.088
```

Part B

```
xtabs(~ johnny.cash + pink.floyd, data = plays) |>
  prop.table(margin = 2)

##             pink.floyd
## johnny.cash          0          1
##           0 0.94503203 0.89517154
##           1 0.05496797 0.10482846
```

$$P(johnny.cash = 1 | pink.floyd = 1) = 0.10$$
$$P(johnny.cash = 1 | pink.floyd = 0) = 0.05$$

These numbers are pretty close, which means different results of Pink Floyd do not change the results of Johnny Cash very much. Therefore, Pink Floyd and Johnny Cash are independent.

Problem 2

Part A

```
sb <- read.csv("superbowl.csv")

xtabs(~ danger, data = sb) |> prop.table()

## danger
##     FALSE      TRUE
## 0.6963563 0.3036437
```

$$P(danger = TRUE) = 0.30$$

```
xtabs(~ danger + funny, data = sb) |>
  prop.table(margin = 2) |>
  round(2)

##        funny
## danger  FALSE TRUE
##   FALSE  0.88 0.61
##   TRUE   0.12 0.39
```

$$P(danger = TRUE|funny = TRUE) = 0.39$$
$$P(danger = TRUE|funny = FALSE) = 0.12$$

Ads using humor are more likely to feature danger.

Part B

```
xtabs(~ animals, data = sb) |> prop.table()

## animals
##     FALSE      TRUE
## 0.6275304 0.3724696
```

$$P(animals = TRUE) = 0.37$$

```
xtabs(~ animals + use_sex, data = sb) |>
  prop.table(margin = 2) |>
  round(2)

##        use_sex
## animals FALSE TRUE
##   FALSE  0.63 0.62
##   TRUE   0.37 0.38
```

$$P(animals = TRUE|use\_sex = TRUE) = 0.38$$
$$P(animals = TRUE|use\_sex = FALSE) = 0.37$$

`use_sex` and `animals` look nearly independent.

Part C

```
xtabs(~ celebrity, data = sb) |> prop.table()
```

```
## celebrity
##     FALSE      TRUE
## 0.7125506 0.2874494
```

$$P(celebrity = TRUE) = 0.29$$

```
xtabs(~ celebrity + patriotic, data = sb) |>
  prop.table(margin = 2) |>
  round(2)

##           patriotic
## celebrity FALSE TRUE
##     FALSE  0.71 0.71
##     TRUE   0.29 0.29
```

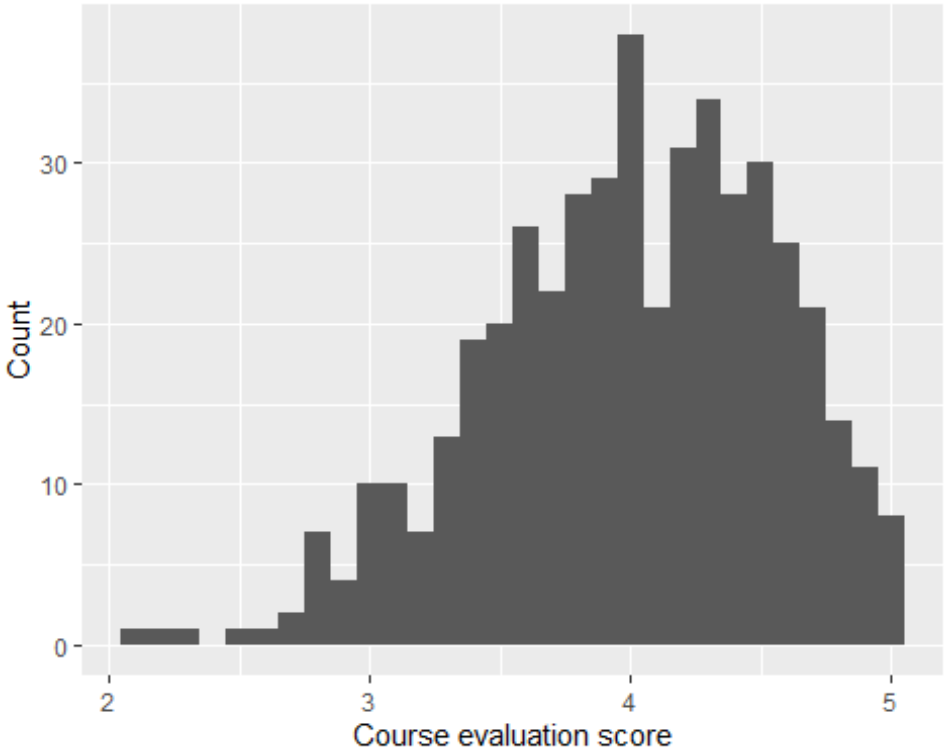
$$P(celebrity = TRUE|patriotic = TRUE) = 0.29$$
$$P(celebrity = TRUE|patriotic = FALSE) = 0.29$$

`celebrity` and `patriotic` look nearly independent.

Problem 3

Part A

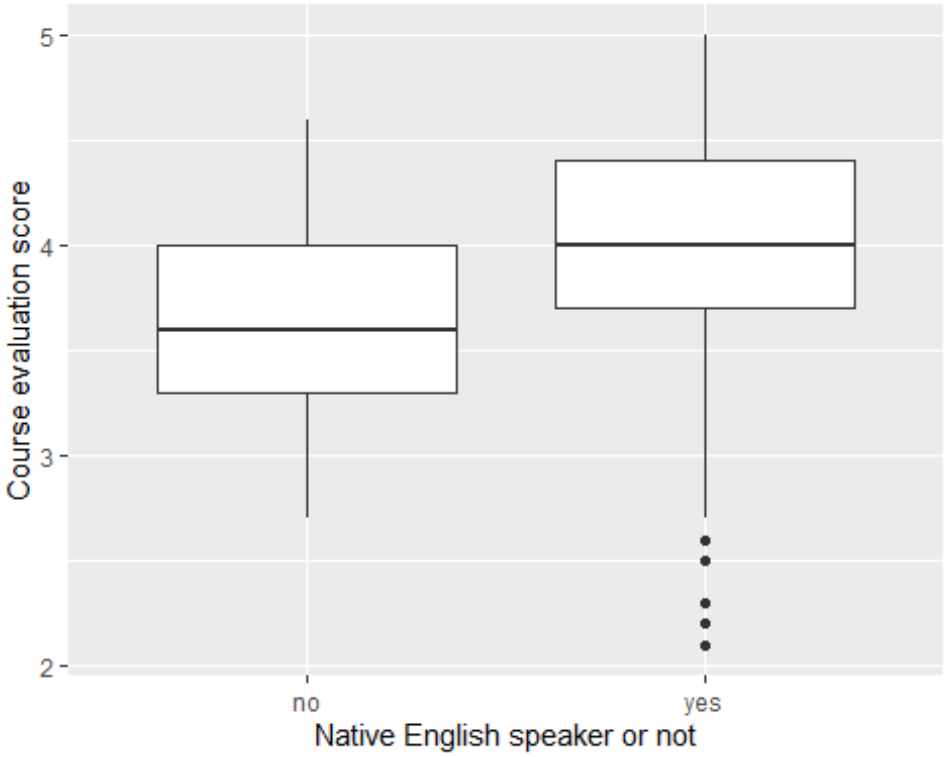
```
profs <- read.csv("profs.csv")

profs |> ggplot(aes(eval)) +
  geom_histogram() +
  xlab("Course evaluation score") +
  ylab("Count")

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



x-axis shows different course evaluation scores. y-axis shows the number of occurrences of the corresponding score.

Part B

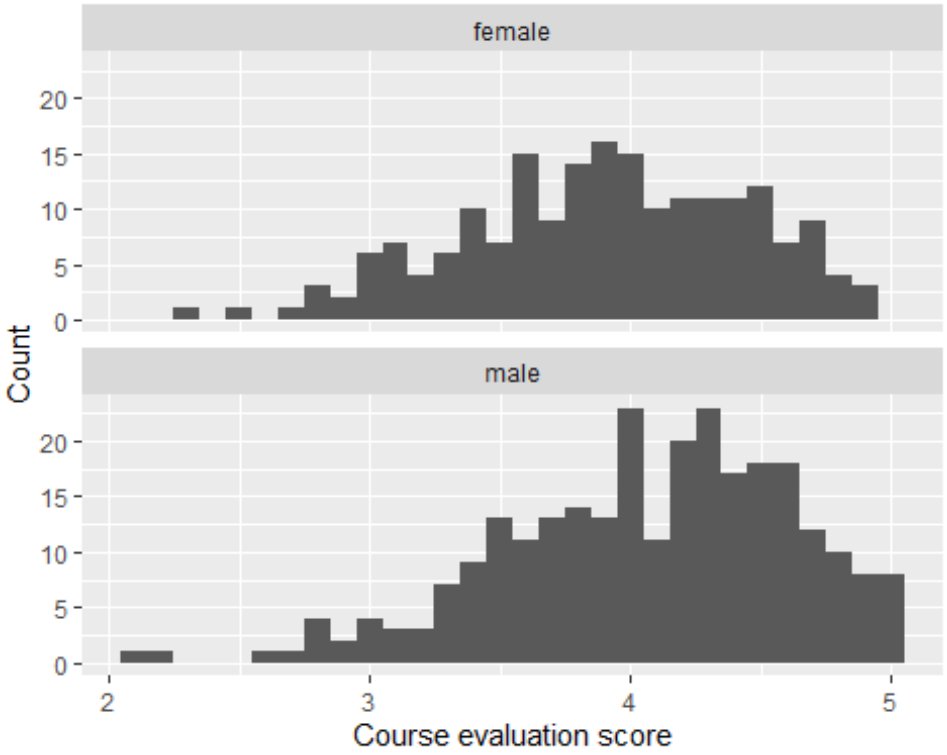
```
profs |> ggplot(aes(native, eval)) +
  geom_boxplot() +
  xlab("Native English speaker or not") +
  ylab("Course evaluation score")
```



Summary distribution of course evaluation scores based on professors' native languages.

Part C

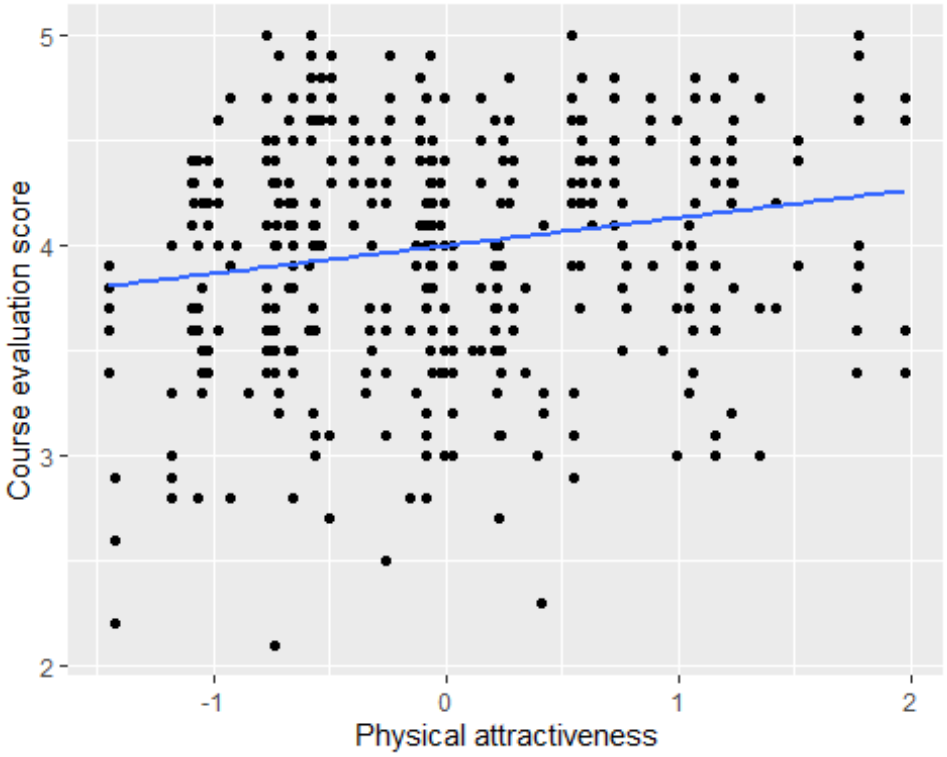
```
profs |> ggplot(aes(eval)) +
  geom_histogram() +
  facet_wrap(~ gender, nrow = 2) +
  xlab("Course evaluation score") +
  ylab("Count")

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



x-axis shows different course evaluation scores. y-axis shows the number of occurrences of the corresponding score, for either male or female.

Part D

```
profs |> ggplot(aes(beauty, eval)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  xlab("Physical attractiveness") +
  ylab("Course evaluation score")

## `geom_smooth()` using formula = 'y ~ x'
```



Each point corresponds to its coordinate.

Problem 4

```
sat <- read.csv("utsat.csv")

# Define a function that returns a data frame of summaries
fn_summary <- function(x){
  data.frame(mean = mean(x),
             sd = sd(x),
             IQR = IQR(x),
             percentile_05 = quantile(x, 0.05),
             percentile_25 = quantile(x, 0.25),
             median = median(x),
             percentile_75 = quantile(x, 0.75),
             percentile_95 = quantile(x, 0.95)) |>
    round(2)
}

# Create 3 data frames of summaries
a <- sat |> summarize(fn_summary(SAT.V))
b <- sat |> summarize(fn_summary(SAT.Q))
c <- sat |> summarize(fn_summary(GPA))

# Concatenate 3 data frames and output as a table
a |> rbind(b) |> rbind(c) |> knitr::kable()
```

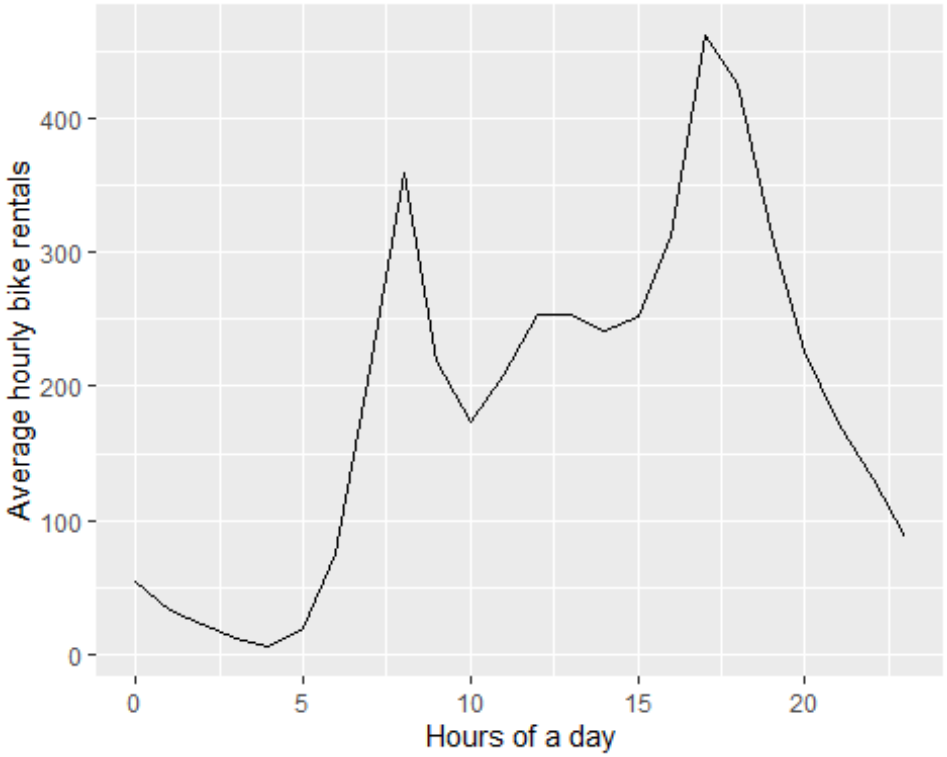
mean	sd	IQR	percentile_05	percentile_25	median	percentile_75	percentile_95
595.05	83.77	110.00	460.00	540.00	590.00	650.00	730.00
619.98	83.08	120.00	480.00	560.00	620.00	680.00	760.00
3.21	0.48	0.72	2.36	2.87	3.25	3.59	3.92

Key summary statistics for SAT Verbal, SAT Quantitative, and GPA.

Problem 5

Plot A

```
bs <- read.csv("bikeshare.csv")

bs |> group_by(hr) |>
  summarize(mean_total = mean(total)) |>
  ggplot(aes(hr, mean_total)) +
  geom_line() +
  xlab("Hours of a day") +
  ylab("Average hourly bike rentals")
```

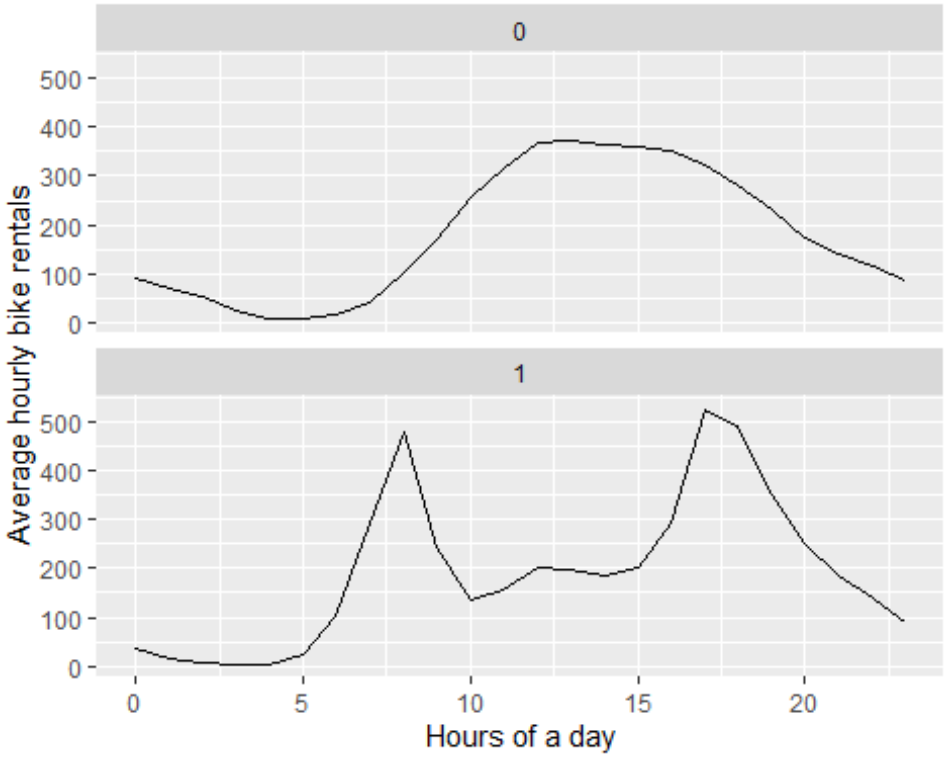


The line plot shows the time-series (hourly) change of bike rentals of a day (24 hours).
Lesson: There are 2 peaks at 8:00 and 17:00.

Plot B

```
bs |> group_by(workingday, hr) |>
  summarize(mean_total = mean(total)) |>
  ggplot(aes(hr, mean_total)) +
  geom_line() +
  facet_wrap(~ workingday, nrow = 2) +
  xlab("Hours of a day") +
  ylab("Average hourly bike rentals")
```

```
## `summarise()` has grouped output by 'workingday'. You can override
using the
## `.groups` argument.
```

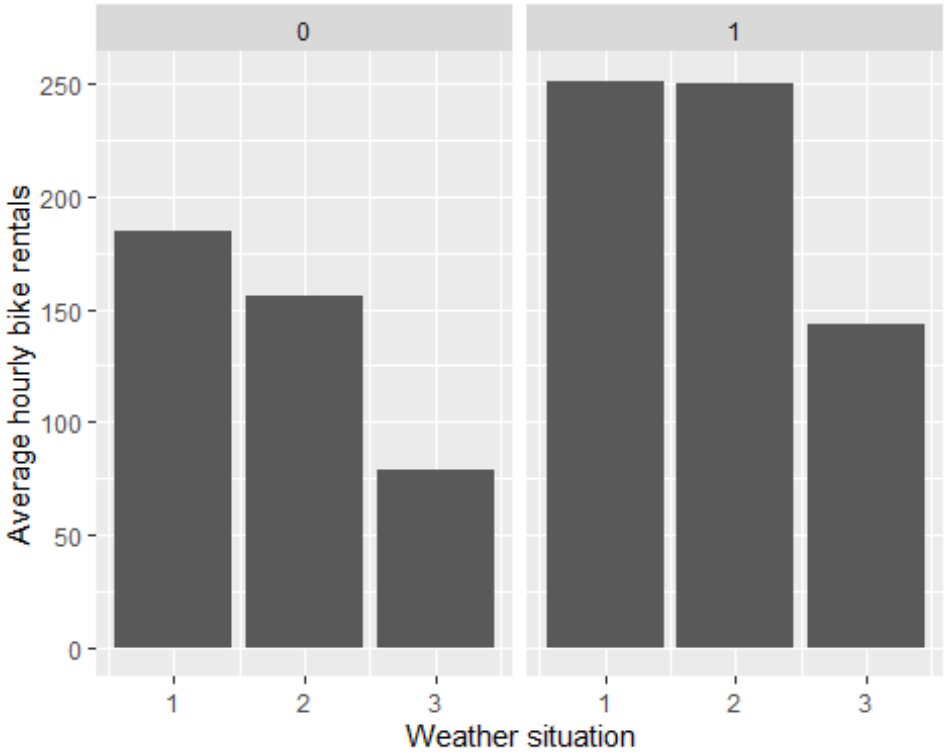


The line plots show the time-series (hourly) change of bike rentals of a day (24 hours). There are 2 groups, "1" represents working day, "0" otherwise.
Lesson: On working days, there are 2 peaks at 8:00 and 17:00; on non-working days, the change is flatter, 12:00 - 16:00 is higher, 3:00 - 6:00 is lower.

Plot C

```
bs |> filter(hr == 9) |>
  group_by(weathersit, workingday) |>
  summarize(mean_total = mean(total)) |>
  ggplot(aes(weathersit, mean_total)) +
  geom_col() +
  facet_wrap(~ workingday) +
  xlab("Weather situation") +
  ylab("Average hourly bike rentals")
```

```
## `summarise()` has grouped output by 'weathersit'. You can override
using the
## `.groups` argument.
```



The bar plots show the bike rentals at 9:00 at different weather situations. There are 2 groups, “1” represents working day, “0” otherwise.
Lesson: Bike rentals are higher on working days. Bike rentals are higher on good-weather days. The difference of bike rentals between weather situation “1” and “2” is smaller on working days than that on non-working days.