Assignment4

Jack

2024-10-08

```
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────────────── tidyverse 2.
0.0 ──
## ✔ dplyr     1.1.4     ✔ readr     2.1.5
## ✔ forcats   1.0.0     ✔ stringr   1.5.1
## ✔ ggplot2   3.5.1     ✔ tibble    3.2.1
## ✔ lubridate 1.9.3     ✔ tidyr     1.3.1
## ✔ purrr     1.0.2
## ── Conflicts ──────────────────────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
s
```

```
library(purrr)
```

1. Tidy data and iteration

1.1 Q1

```
impute_by_mean <- function(x) {
  mu <- mean(x,na.rm=TRUE)
  impute_f <- function(z){
    if(is.na(z)){
      return (mu)
    }else{
      return (z)
    }
  }
  return (map_dbl(x,impute_f))
}
# The function you provided uses map_dbl, which is from the purrr package in R. To run this function, y
ou will need to load the purrr library
```

1.1 Q2

```
impute_by_median <- function(x) {
  mu <- median(x,na.rm=TRUE)
  impute_f <- function(z){
    if(is.na(z)){
      return (mu)
    }else{
      return (z)
    }
  }
  return (map_dbl(x,impute_f))
}
impute_by_median(c(1,2,NA,4))
```

```
## [1] 1 2 2 4
```

1.1 Q3

```
x <- seq(0,10,0.1)
y <- map_dbl(x,~5*.x+1)
df_xy <- data.frame(x,y)
df_xy %>% head(5)
```

```
##     x   y
## 1 0.0 1.0
## 2 0.1 1.5
## 3 0.2 2.0
## 4 0.3 2.5
## 5 0.4 3.0
```

1.1 Q4

```
df_xy %>% mutate(z = map2_dbl(x,y,~.x+.y)) %>% head(5)
```

```
##     x   y   z
## 1 0.0 1.0 1.0
## 2 0.1 1.5 1.6
## 3 0.2 2.0 2.2
## 4 0.3 2.5 2.8
## 5 0.4 3.0 3.4
```

```
sometimes_missing <- function(index, value) {
  impute_f <- function(x,y){
    if(x %% 5 == 0){
      return (NA)
    }else{
      return (y)
    }
  }
  return (map2_dbl(index,value,impute_f))
}
sometimes_missing(14,25)
```

```
## [1] 25
```

```
sometimes_missing(15,25)
```

```
## [1] NA
```

```
y <- map2_dbl(row_number(df_xy),y,sometimes_missing)
df_xy_missing <- data.frame(x,y)
df_xy_missing %>% head(10)
```

```
##      x   y
## 1  0.0 1.0
## 2  0.1 1.5
## 3  0.2 2.0
## 4  0.3 2.5
## 5  0.4  NA
## 6  0.5 3.5
## 7  0.6 4.0
## 8  0.7 4.5
## 9  0.8 5.0
## 10 0.9  NA
```

1.1 Q5

```
df_xy_imputed <- df_xy_missing %>% mutate(y=impute_by_median(y))
df_xy_imputed %>% head(10)
```

```
##      x    y
## 1  0.0  1.0
## 2  0.1  1.5
## 3  0.2  2.0
## 4  0.3  2.5
## 5  0.4 26.0
## 6  0.5  3.5
## 7  0.6  4.0
## 8  0.7  4.5
## 9  0.8  5.0
## 10 0.9 26.0
```

1.2 Q1

```
library(readxl)
folder_path <- "D:/bristol/Statistical Computing and Empirical Methods/RStudioFile/RLab/Assignment4/"
file_name <- "HockeyLeague.xlsx"
file_path <- paste(folder_path,file_name,sep="")
wins_data_frame <- read_excel(file_path,sheet="Wins")
```

```
## New names:
## • `` -> `...1`
```

```
wins_data_frame %>% select(1:5) %>% head(3)
```

```
## # A tibble: 3 × 5
##   ...1   `1990`   `1991`   `1992`   `1993`
##   <chr>  <chr>    <chr>    <chr>    <chr>
## 1 Ducks  30 of 50 11 of 50 30 of 50 12 of 50
## 2 Eagles 24 of 50 12 of 50 37 of 50 14 of 50
## 3 Hawks  20 of 50 22 of 50 33 of 50 11 of 50
```

```
values <- as.character(seq(1990,2020))
wins_tidy <- wins_data_frame %>% pivot_longer(values,names_to="Year",values_to="WinsAndTotal") %>% sepa
rate(WinsAndTotal, into=c("Wins","Total"),sep="of",convert=TRUE) %>% rename(Team = "...1")
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## ℹ Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(values)
##
##   # Now:
##   data %>% select(all_of(values))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
wins_tidy %>% dim()
```

```
## [1] 248   4
```

```
wins_tidy %>% head(5)
```

```
## # A tibble: 5 × 4
##   Team  Year   Wins Total
##   <chr> <chr> <dbl> <int>
## 1 Ducks 1990     30    50
## 2 Ducks 1991     11    50
## 3 Ducks 1992     30    50
## 4 Ducks 1993     12    50
## 5 Ducks 1994     24    50
```

1.2 Q2

```
losses_data_frame <- read_excel(file_path,sheet="Losses")
```

```
## New names:
## • `` -> `...1`
```

```
losses_tidy <- losses_data_frame %>% pivot_longer(values,names_to="Year",values_to="LossesAndTotal") %
>% separate(LossesAndTotal, into=c("Losses","Total"),sep="of",convert=TRUE) %>% rename(Team = "...1")

losses_tidy %>% head(5)
```

```
## # A tibble: 5 × 4
##   Team  Year  Losses Total
##   <chr> <chr>  <dbl> <int>
## 1 Ducks 1990      20    50
## 2 Ducks 1991      37    50
## 3 Ducks 1992       1    50
## 4 Ducks 1993      30    50
## 5 Ducks 1994       7    50
```

1.2 Q3

```
hockey_df <- wins_tidy %>% inner_join(losses_tidy) %>% mutate(Draws = Total - Wins - Losses) %>% mutate
(across(c(Wins,Losses,Draws), ~.x/Total,.names="{.col}_rt"))
```

```
## Joining with `by = join_by(Team, Year, Total)`
```

```
hockey_df %>% head(5)
```

```
## # A tibble: 5 × 9
##   Team  Year   Wins Total Losses Draws Wins_rt Losses_rt Draws_rt
##   <chr> <chr> <dbl> <int>  <dbl> <dbl>   <dbl>     <dbl>    <dbl>
## 1 Ducks 1990     30    50     20     0    0.6       0.4      0
## 2 Ducks 1991     11    50     37     2    0.22      0.74     0.04
## 3 Ducks 1992     30    50      1    19    0.6       0.02     0.38
## 4 Ducks 1993     12    50     30     8    0.24      0.6      0.16
## 5 Ducks 1994     24    50      7    19    0.48      0.14     0.38
```

1.2 Q4

```
hockey_sum <- hockey_df %>% group_by(Team) %>% summarize(across(c("Wins_rt","Losses_rt","Draws_rt"),lis
t(md=median,mn=mean),.names="{substring(.col,1,1)}_{.fn}")) %>% arrange(desc(W_md))
hockey_sum %>% head(8)
```

```
## # A tibble: 8 × 7
##   Team        W_md  W_mn  L_md  L_mn  D_md  D_mn
##   <chr>      <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 Eagles     0.45  0.437 0.25  0.279 0.317 0.284
## 2 Penguins   0.45  0.457 0.3   0.310 0.133 0.232
## 3 Hawks      0.417 0.388 0.233 0.246 0.32  0.366
## 4 Ducks      0.383 0.362 0.34  0.333 0.25  0.305
## 5 Owls       0.32  0.333 0.3   0.33  0.383 0.337
## 6 Ostriches  0.3   0.309 0.4   0.395 0.267 0.296
## 7 Storks     0.3   0.284 0.22  0.283 0.48  0.433
## 8 Kingfishers 0.233 0.245 0.34  0.360 0.4   0.395
```

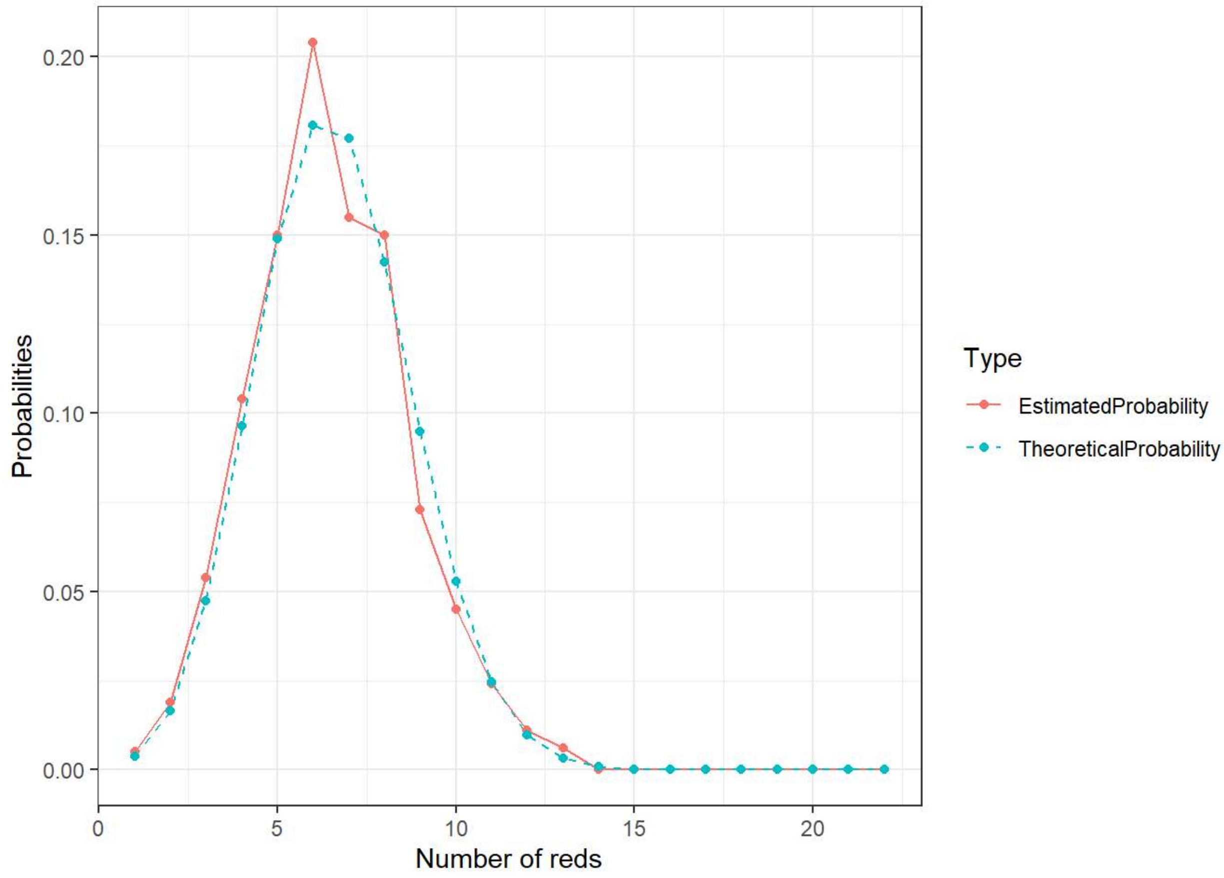
1.3 Q1

```
num_red_balls<-3
num_blue_balls<-7
total_draws<-22
prob_red_spheres<-function(z){
  total_balls<-num_red_balls+num_blue_balls
  log_prob<-log(choose(total_draws,z))+
    z*log(num_red_balls/total_balls)+(total_draws-
z)*log(num_blue_balls/total_balls)
  return(exp(log_prob))
}

num_trials <- 1000
set.seed(0)

num_reds_in_simulation <- data.frame(trial=1:num_trials) %>% mutate(sample_balls = map(.x=trial,functi
on(x){sample(10,22,replace=TRUE)})) %>% mutate(num_reds=map_dbl(sample_balls,~ sum(.x<=3))) %>% pull(n
um_reds)

prob_by_num_reds <- data.frame(num_reds=seq(22)) %>% mutate(TheoreticalProbability=prob_red_spheres(num
_reds)) %>% mutate(EstimatedProbability=map_dbl(num_reds,~sum(num_reds_in_simulation==.x))/num_trials)

print(prob_by_num_reds)
```

```
##    num_reds TheoreticalProbability EstimatedProbability
## 1         1           3.686403e-03                0.005
## 2         2           1.658881e-02                0.019
## 3         3           4.739661e-02                0.054
## 4         4           9.648595e-02                0.104
## 5         5           1.488640e-01                0.150
## 6         6           1.807635e-01                0.204
## 7         7           1.770744e-01                0.155
## 8         8           1.422919e-01                0.150
## 9         9           9.486130e-02                0.073
## 10       10           5.285129e-02                0.045
## 11       11           2.470970e-02                0.024
## 12       12           9.707380e-03                0.011
## 13       13           3.200235e-03                0.006
## 14       14           8.816975e-04                0.000
## 15       15           2.015309e-04                0.000
## 16       16           3.778704e-05                0.000
## 17       17           5.715686e-06                0.000
## 18       18           6.804388e-07                0.000
## 19       19           6.139298e-08                0.000
## 20       20           3.946691e-09                0.000
## 21       21           1.610894e-10                0.000
## 22       22           3.138106e-12                0.000
```

1.3 Q2

```
prob_by_num_reds %>% pivot_longer(cols=c("EstimatedProbability", "TheoreticalProbability"), names_to="T
ype", values_to="count") %>% ggplot(aes(num_reds,count)) + geom_line(aes(linetype=Type,color=Type)) + g
eom_point(aes(color=Type)) + scale_linetype_manual(values=c("solid","dashed")) + theme_bw() + xlab("Num
ber of reds") + ylab("Probabilities")
```



2. Conditional probability, Bayes rule and independence

2.1 Q1

$P(A) = 0.9, \quad P(A^c) = 0.1, \quad P(B|A) = 0.8, \quad P(B^c|A^c) = 0.75, \quad P(B|A^c) = 0.25$

$P(B) = P(B|A)\ cdot P(A) + P(B|A^c) \cdot P(A^c) = 0.8 \times 0.9 + 0.25 \times 0.1 = 0.745$

$P(A|B) = \frac{P(A) \cdot P(B|A)}{P(B)} = \frac{0.9 \times 0.8}{0.745} = 0.966$

2.2 Q1

Because $A \subseteq B \quad and \quad \mathbb{P}(B \backslash A) = 0, then \quad A = B\ \mathbb{P}(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B)}{P(B)} = 1$

$yes, P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(\emptyset)}{P(B)} = 0$

$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B)}{P(B)} = 1$

$P(A|\Omega) = \frac{P(A \cap \Omega)}{P(\Omega)} = \frac{P(A)}{1} = P(A)$

Let's suppose B $A \cap C$ ***is D***
$P(A \cap B \cap C) = P(B \cap D) = P(B|D) \cdot P(D) = P(B|A \cap C) \cdot P(A \cap C) = P(B|A \cap C) \cdot P(A|C) \cdot P(C)$

$P(A|B \cap C) = \frac{P(A \cap B \cap C)}{P(B \cap C)} = \frac{P(B|A \cap C) \cdot P(A|C) \cdot P(C)}{P(B|C) \cdot P(C)} = \frac{P(B|A \cap C) \cdot P(A|C)}{P(B|C)}$

2.2 Q2

Consider B as Windy, A as Cancel. Then we have P(A|B)=0.3, $P(A|B^c) = 0.1$***, P(B)=0.2,*** $P(B^c) = 0.8$***, and*** $P(A^c) = 1 - P(A)$

$P(A) = P(A|B) \cdot P(B) + P(A|B^c) \cdot P(B^c) = 0.3 \times 0.2 + 0.1 \times 0.8 = 0.14$

$P(A^c) = 1 - 0.14 = 0.86$

2.3 Q1

$\mathbb{P}(A \cap B) = \mathbb{P}((1, 1, 0)) = \frac{1}{4}$

$\mathbb{P}(A) \cdot \mathbb{P}(B) = \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$

$Then, \mathbb{P}(A \cap B) = \mathbb{P}(A) \cdot \mathbb{P}(B), \mathbb{P}(A \cap C) = \mathbb{P}(A) \cdot \mathbb{P}(C), \mathbb{P}(B \cap C) = \mathbb{P}(B) \cdot \mathbb{P}(C)$

$A \cap B \cap C = \emptyset, \mathbb{P}(A \cap B \cap C) = 0, yes, independent$

2.4 The Monty hall problem

2.4 Q1