Yufei_HW3

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com (http://rmarkdown.rstudio.com).

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
#Section A
#Question 1
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyvers
e 1.2.1 ──
```

```
## ✔ ggplot2 3.2.1     ✔ purrr   0.3.3
## ✔ tibble  2.1.3     ✔ dplyr   0.8.3
## ✔ tidyr   1.0.0     ✔ stringr 1.4.0
## ✔ readr   1.3.1     ✔ forcats 0.4.0
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conf
licts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library(dplyr)
#create the area code column
set.seed(1050)
Area_code1 <-sample(LETTERS, 1000, replace=TRUE)
Area_code2 <-sample(LETTERS, 1000, replace=TRUE)

#create the company column
set.seed(1050)
Company1<- sample("Alpha", 1000, replace = TRUE)
Company2<- sample("Beta", 1000, replace = TRUE)


#create the employee height column
set.seed(1005)
height1 <- sample(rnorm(1000,mean=160,sd=5), replace = TRUE)
height2 <- sample(rnorm(1000,mean=170,sd=5),replace=TRUE)

#create the data frame
df1 <- tibble("Area_Code"=Area_code1, "Company"=Company1,"Employee_H
eight"=height1)
df2 <- tibble("Area_Code"=Area_code2, "Company"=Company2,"Employee_H
eight"=height2)
df3 <-full_join(df1, df2)
```

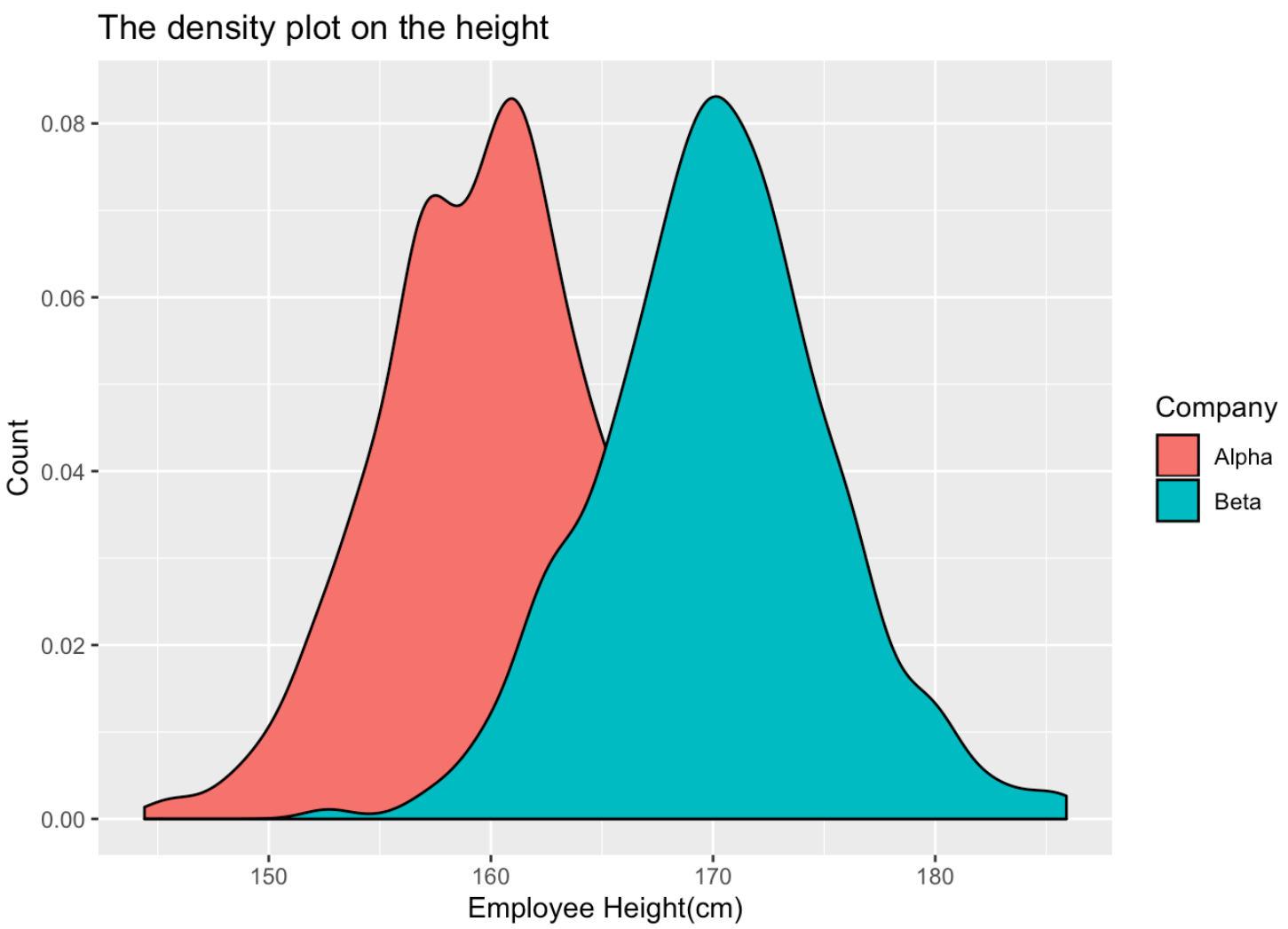
```
## Joining, by = c("Area_Code", "Company", "Employee_Height")
```

```
df3
```

```
## # A tibble: 2,000 x 3
##    Area_Code Company Employee_Height
##    <chr>     <chr>              <dbl>
##  1 Z         Alpha               160.
##  2 W         Alpha               158.
##  3 H         Alpha               159.
##  4 S         Alpha               156.
##  5 P         Alpha               163.
##  6 Z         Alpha               160.
##  7 M         Alpha               162.
##  8 D         Alpha               159.
##  9 Z         Alpha               158.
## 10 B         Alpha               156.
## # … with 1,990 more rows
```

```
plot1 <- ggplot(data=df3)+
  geom_density(mapping=aes(x=df3$Employee_Height,fill=Company))+
scale_fill_discrete() +
scale_y_continuous() +
xlab("Employee Height(cm)")+
ylab("Count")+
  ggtitle("The density plot on the height")

plot1
```




```
# Question 2
#part 1
#create the new datafram
# caculate the average height for each company
target1 <-group_by(df1,Company)
target2 <- summarize(target1, AvgHeight=mean(df1$Employee_Height,na.
rm=TRUE))
target2
```

```
## # A tibble: 1 x 2
##   Company AvgHeight
##   <chr>       <dbl>
## 1 Alpha        160.
```

```
target3 <-group_by(df2,Company)
target4 <- summarize(target3, AvgHeight=mean(df2$Employee_Height,na.
rm=TRUE))
target4
```

```
## # A tibble: 1 x 2
##   Company AvgHeight
##   <chr>       <dbl>
## 1 Beta         170.
```

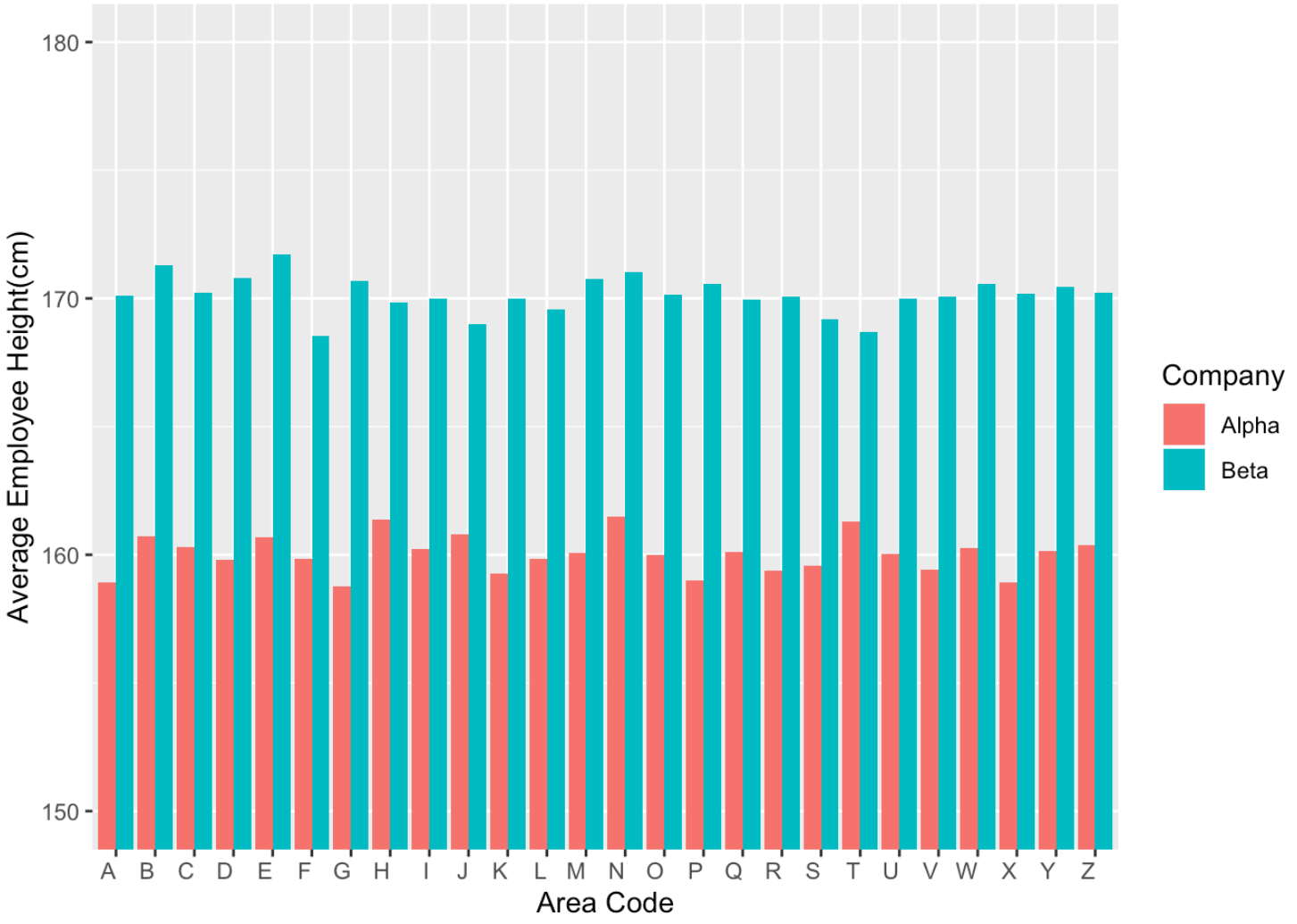
```
df4 <- full_join(target2,target4,by = c("Company","AvgHeight"))
df4
```

```
## # A tibble: 2 x 2
##   Company AvgHeight
##   <chr>       <dbl>
## 1 Alpha        160.
## 2 Beta         170.
```

```
#part 2
new <- df3 %>% group_by(Company,Area_Code) %>%
  summarise(AvgHeight=mean(Employee_Height,na.rm=TRUE))
new
```

```
## # A tibble: 52 x 3
## # Groups:   Company [2]
##    Company Area_Code AvgHeight
##    <chr>   <chr>         <dbl>
##  1 Alpha   A              159.
##  2 Alpha   B              161.
##  3 Alpha   C              160.
##  4 Alpha   D              160.
##  5 Alpha   E              161.
##  6 Alpha   F              160.
##  7 Alpha   G              159.
##  8 Alpha   H              161.
##  9 Alpha   I              160.
## 10 Alpha   J              161.
## # … with 42 more rows
```

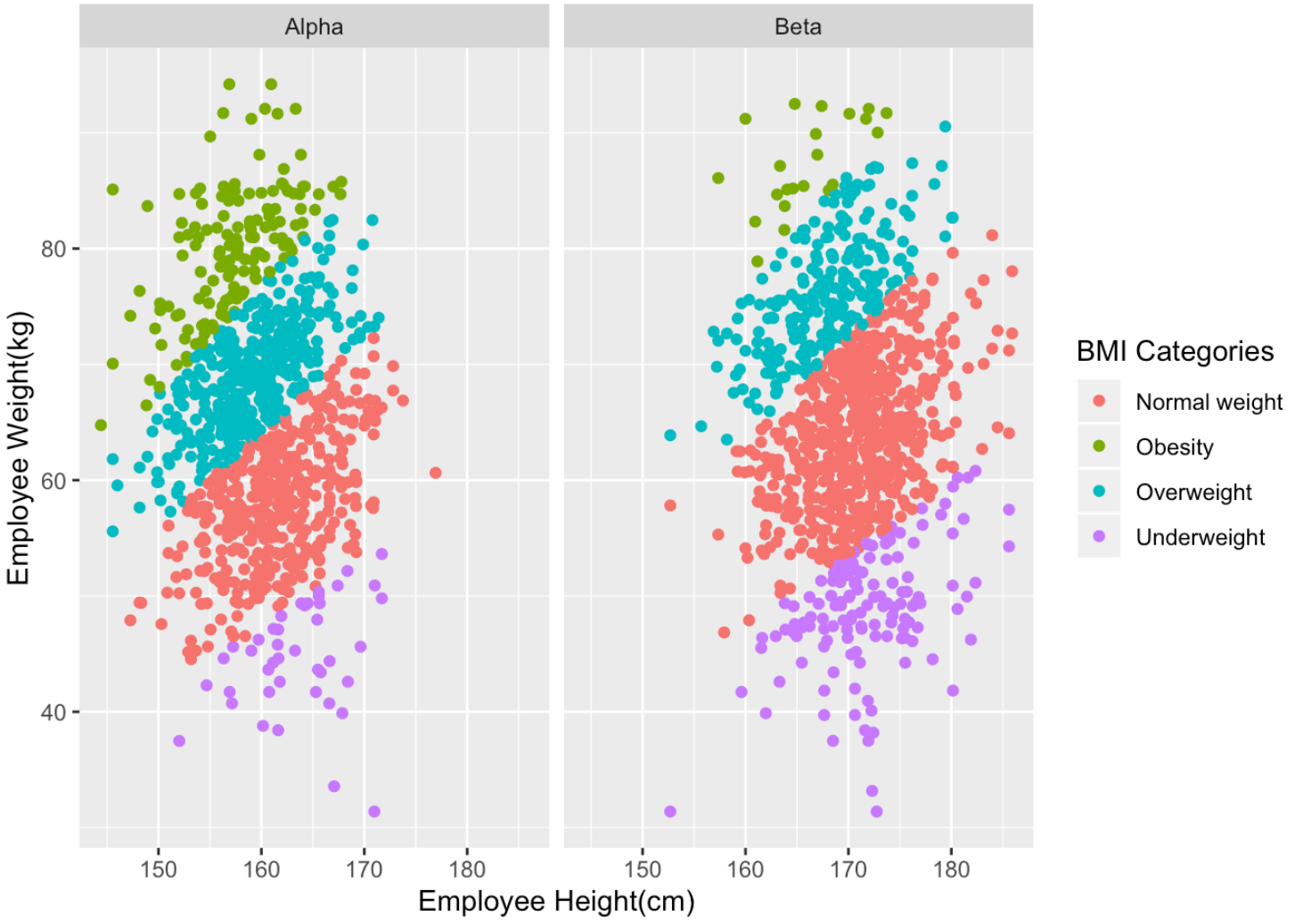
```
#plot the target chart
plot2 <- ggplot(data=new)+
  geom_bar(mapping=aes(x=Area_Code,y=AvgHeight,fill=Company),stat="i
dentity",position ="dodge")+
  scale_x_discrete() +
scale_fill_discrete() +
scale_y_continuous() +
theme(axis.text.x=element_text(angle = 360, hjust = 1))+
coord_cartesian(ylim=c(150,180))+
xlab("Area Code")+
ylab("Average Employee Height(cm)")
plot2
```




```
#Question 3
#create the random employee weight and add it to the previous datafr
ame
set.seed(1000)
df5<- mutate(df3, "Employee_Weight(kg)"=sample(rnorm(2000,mean=65,sd
=10), replace = TRUE))
df6<- mutate(df5,"BMI"=`Employee_Weight(kg)`/((Employee_Height/100)^
2))
df7<- mutate(df6,BMI_Categories= ifelse(BMI <=18.5,"Underweight",ife
lse(BMI>18.5 & BMI <= 25,"Normal weight",ifelse(BMI >25 & BMI<=30 ,"
Overweight", ifelse(BMI >30, "Obesity",""))))))
df7
```

```
## # A tibble: 2,000 x 6
##    Area_Code Company Employee_Height `Employee_Weight…   BMI BMI_
Categories
##    <chr>     <chr>              <dbl>              <dbl> <dbl> <chr
>
##  1 Z         Alpha               160.               66.8  26.1 Over
weight
##  2 W         Alpha               158.               64.9  25.9 Over
weight
##  3 H         Alpha               159.               66.2  26.4 Over
weight
##  4 S         Alpha               156.               81.8  33.7 Obes
ity
##  5 P         Alpha               163.               85.0  32.1 Obes
ity
##  6 Z         Alpha               160.               50.5  19.7 Norm
al weight
##  7 M         Alpha               162.               79.2  30.1 Obes
ity
##  8 D         Alpha               159.               49.3  19.6 Norm
al weight
##  9 Z         Alpha               158.               61.8  24.9 Norm
al weight
## 10 B         Alpha               156.               71.1  29.2 Over
weight
## # … with 1,990 more rows
```

```
#create the target plot
df7 %>% ggplot(mapping=aes(df7$Employee_Height,df7$`Employee_Weight(
kg)`))+
  geom_point(mapping=aes(color=BMI_Categories))+
  facet_grid(.~Company)+
  xlab("Employee Height(cm)")+
  ylab("Employee Weight(kg)")+
  labs(color="BMI Categories")
```




```
#Section B
#Problem 1
library(tidyverse)
library(plyr)
```

```
## ------------------------------------------------------------
--------
```

```
## You have loaded plyr after dplyr - this is likely to cause proble
ms.
## If you need functions from both plyr and dplyr, please load plyr
first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------
--------
```

```
##
## Attaching package: 'plyr'
```

```
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise
,
##     summarize
```

```
## The following object is masked from 'package:purrr':
##
##     compact
```

```
library(dplyr)
library(haven)
#upload the dataset
dataset1 <-  read_xpt('https://wwwn.cdc.gov/Nchs/Nhanes/2015-2016/DE
MO_I.XPT')

#filter the ratio without missing values
new111 <- dataset1[!is.na(dataset1$INDFMPIR),]
#remove the ratio's decimals
new111$INDFMPIR <- trunc(new111$INDFMPIR)

#count1
dataset2 <- group_by(new111,RIDRETH1,INDFMPIR)
dataset3 <- dataset2 %>% dplyr::count(RIDRETH1,name="count1")
dataset3 %>% dplyr::filter(!is.na(INDFMPIR))
```

```
## # A tibble: 30 x 3
## # Groups:   RIDRETH1, INDFMPIR [30]
##    RIDRETH1 INDFMPIR count1
##       <dbl>    <dbl>  <int>
##  1        1        0    666
##  2        1        1    512
##  3        1        2    265
##  4        1        3     98
##  5        1        4     55
##  6        1        5     69
##  7        2        0    343
##  8        2        1    351
##  9        2        2    156
## 10        2        3    117
## # … with 20 more rows
```

```
# caculate the proportion of each ethnic families among all families at each annual family income value
dataset4 <- plyr::ddply(dataset3,.(INDFMPIR),transform,prop1=count1/sum(count1))
dataset4 %>% dplyr::filter(!is.na(INDFMPIR)) %>% group_by( RIDRETH1)
```

```
## # A tibble: 30 x 4
## # Groups:   RIDRETH1 [5]
##    RIDRETH1 INDFMPIR count1 prop1
##       <dbl>    <dbl>  <int> <dbl>
##  1        1        0    666 0.288
##  2        2        0    343 0.148
##  3        3        0    367 0.159
##  4        4        0    669 0.289
##  5        5        0    270 0.117
##  6        1        1    512 0.207
##  7        2        1    351 0.142
##  8        3        1    783 0.316
##  9        4        1    501 0.202
## 10        5        1    330 0.133
## # … with 20 more rows
```

```
#caculate the proportion of each ethnic families among all families
dataset5 <- group_by(new111,RIDRETH1)
dataset6 <- dataset5 %>% dplyr::count(RIDRETH1,name="count2")

library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
## The following object is masked from 'package:purrr':
##
##     transpose
```

```
setDT(dataset6)[,prop2:=count2/sum(count2)]
dataset6
```

```
##    RIDRETH1 count2     prop2
## 1:        1   1665 0.1866801
## 2:        2   1132 0.1269201
## 3:        3   2877 0.3225698
## 4:        4   1881 0.2108981
## 5:        5   1364 0.1529319
```

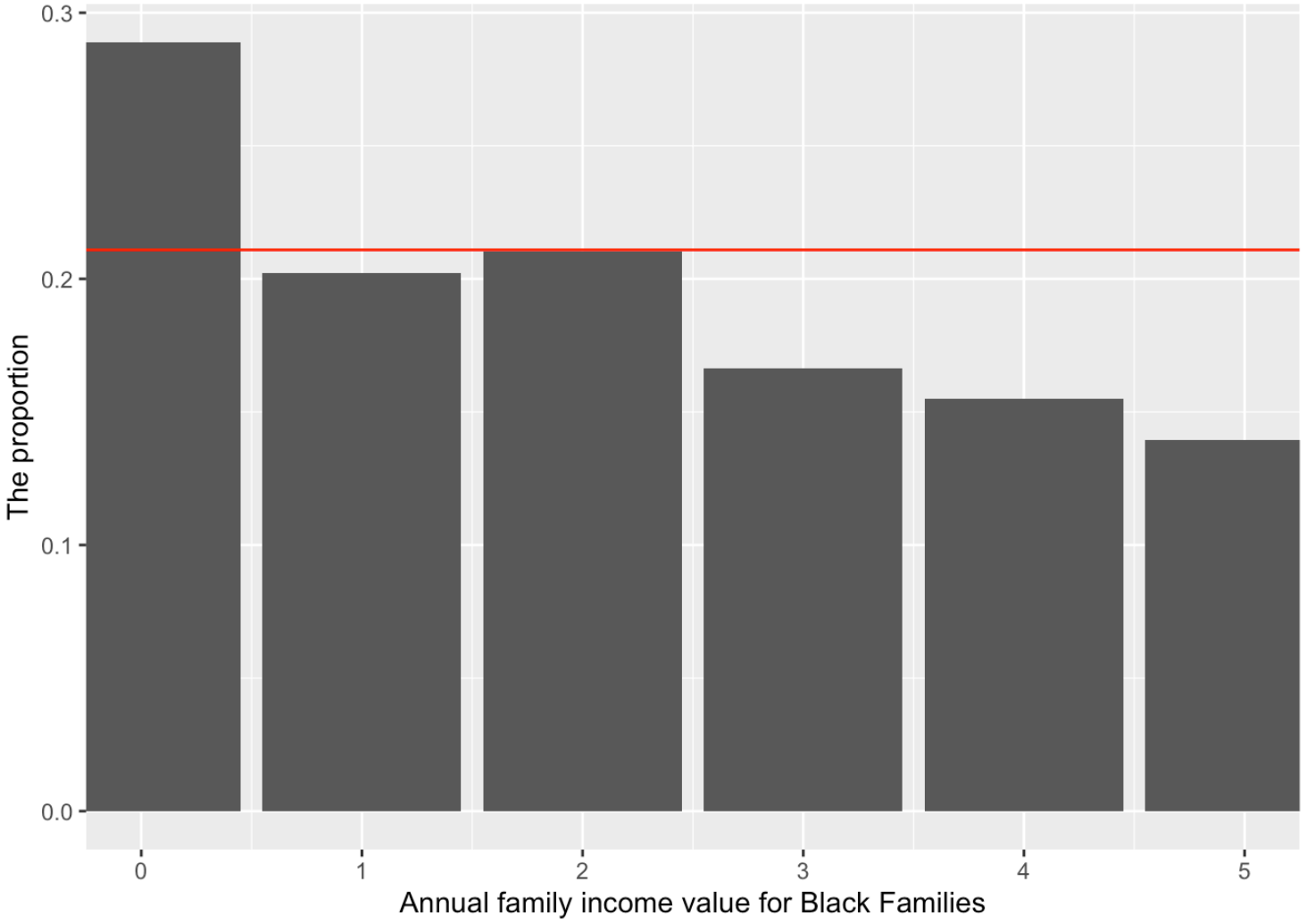
```
#get the target data frame
dataset7 <- full_join(dataset4,dataset6)
```

```
## Joining, by = "RIDRETH1"
```

```
dataset7 %>% dplyr::filter(!is.na(INDFMPIR)) %>% select(RIDRETH1,IND
FMPIR,prop2,prop1)
```

```
##    RIDRETH1 INDFMPIR     prop2      prop1
## 1         1        0 0.1866801 0.28768898
## 2         2        0 0.1269201 0.14816415
## 3         3        0 0.3225698 0.15853132
## 4         4        0 0.2108981 0.28898488
## 5         5        0 0.1529319 0.11663067
## 6         1        1 0.1866801 0.20670166
## 7         2        1 0.1269201 0.14170367
## 8         3        1 0.3225698 0.31610820
## 9         4        1 0.2108981 0.20226080
## 10        5        1 0.1529319 0.13322568
## 11        1        2 0.1866801 0.18088737
## 12        2        2 0.1269201 0.10648464
## 13        3        2 0.3225698 0.35631399
## 14        4        2 0.2108981 0.21023891
## 15        5        2 0.1529319 0.14607509
## 16        1        3 0.1866801 0.11475410
## 17        2        3 0.1269201 0.13700234
## 18        3        3 0.3225698 0.41803279
## 19        4        3 0.2108981 0.16627635
## 20        5        3 0.1529319 0.16393443
## 21        1        4 0.1866801 0.09353741
## 22        2        4 0.1269201 0.12074830
## 23        3        4 0.3225698 0.44557823
## 24        4        4 0.2108981 0.15476190
## 25        5        4 0.1529319 0.18537415
## 26        1        5 0.1866801 0.05655738
## 27        2        5 0.1269201 0.07704918
## 28        3        5 0.3225698 0.48032787
## 29        4        5 0.2108981 0.13934426
## 30        5        5 0.1529319 0.24672131
```

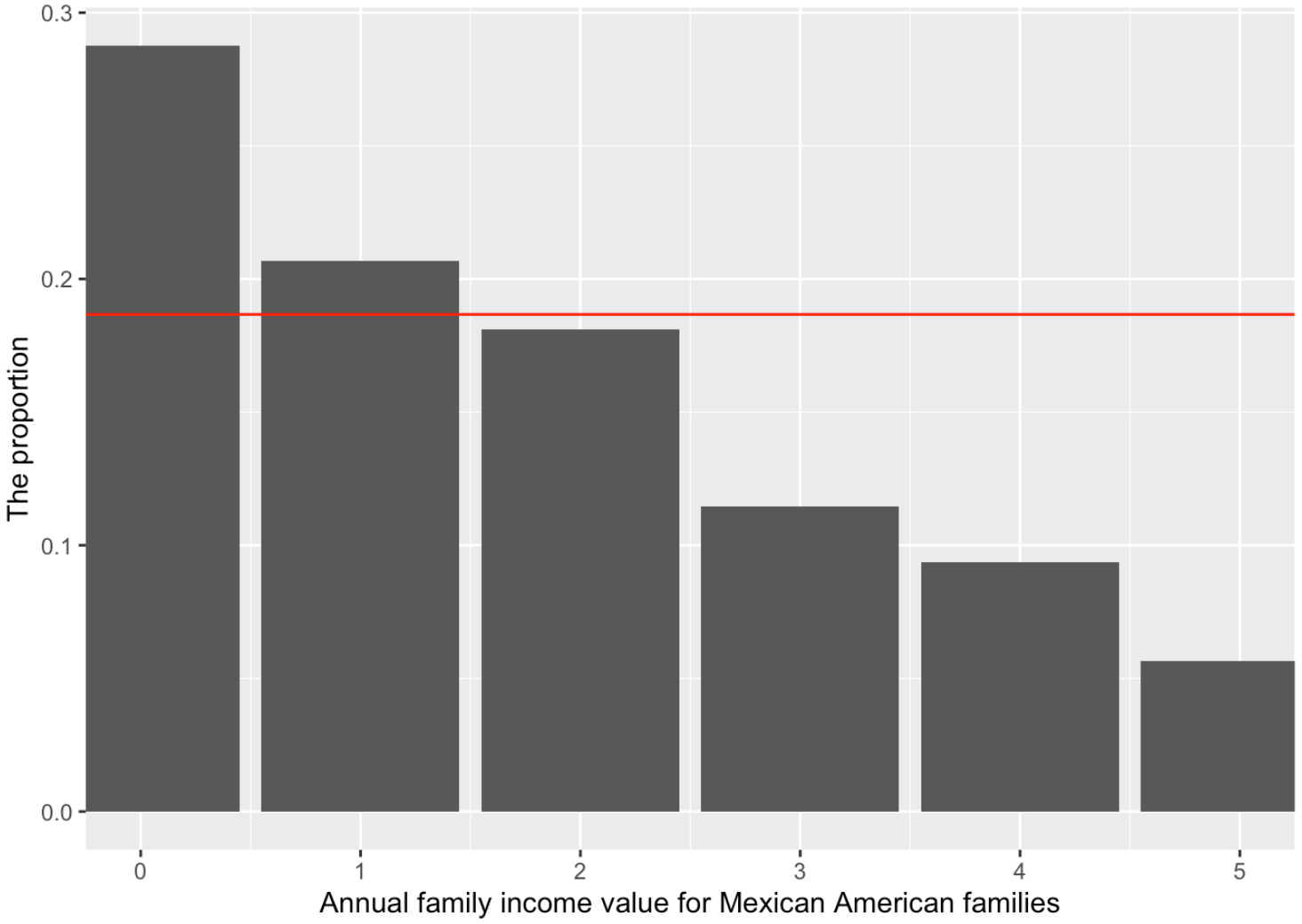
```
#plot1 for black families (4)
dataset7 %>% dplyr::filter(RIDRETH1==4) %>%
  ggplot()+
  geom_bar(mapping=aes(INDFMPIR,prop1),stat="identity")+
  coord_cartesian(xlim=c(0:5))+
  geom_hline(yintercept=0.2108981,color="red")+
  xlab("Annual family income value for Black Families")+
  ylab("The proportion")
```



Answer

Most balck families are under the poverty. The proportions of the annual family income values (0) are above the the proportion of familiy over all familities, and the proportion of “0” annual family income value is the highest number in the plot.

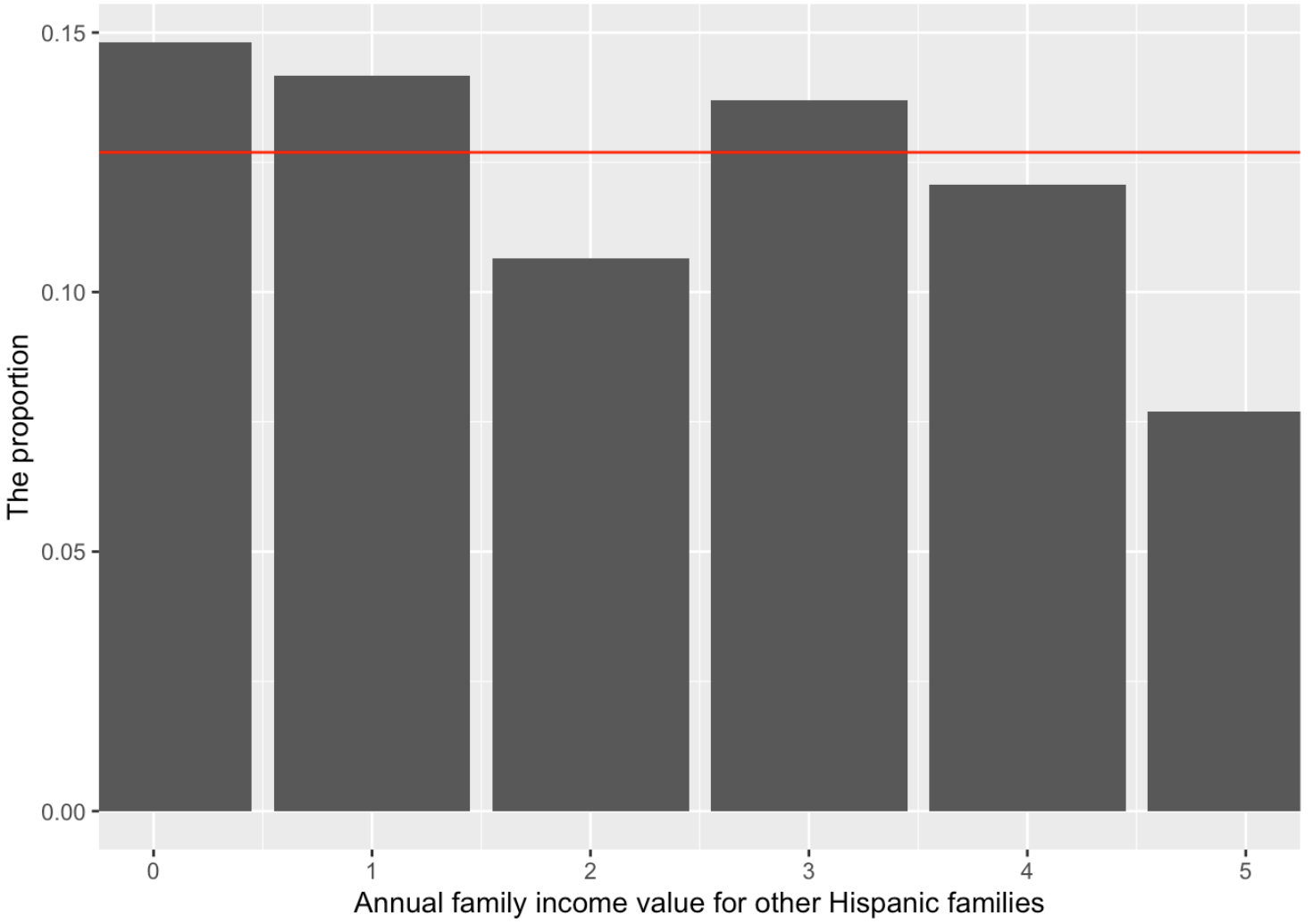
```
#plot2 for Mexican American families (1)
dataset7 %>% dplyr::filter(RIDRETH1==1) %>%
  ggplot()+
  geom_bar(mapping=aes(INDFMPIR,prop1),stat="identity")+
  coord_cartesian(xlim=c(0:5))+
  geom_hline(yintercept=0.1866801,color="red")+
  xlab("Annual family income value for Mexican American families")+
  ylab("The proportion")
```



Answer

Most Mexican American families are under the poverty. The proportions of the annual family income values (0 and 1) are above the the proportion of familiy over all familities, and the proportion of “0” annual family income value is the highest number in the plot.

```
#plot3 for other Hispanic families (2)
dataset7 %>% dplyr::filter(RIDRETH1==2) %>%
  ggplot()+
  geom_bar(mapping=aes(INDFMPIR,prop1),stat="identity")+
  coord_cartesian(xlim=c(0:5))+
  geom_hline(yintercept=0.1269201,color="red")+
  xlab("Annual family income value for other Hispanic families")+
  ylab("The proportion")
```



Answer

Most other Hispanic families are over poverty. The proportions of the annual family income values (0,1,3) are above the the proportion of famility over all familities, and the proportion of “0” annual family income value is the highest number in the plot.

```
```