data analysis exercise

Jessie Chen

1/22/2023

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.

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:

```
getwd()
```

```
## [1] "/Users/jessiechen/Desktop/interviews"
```

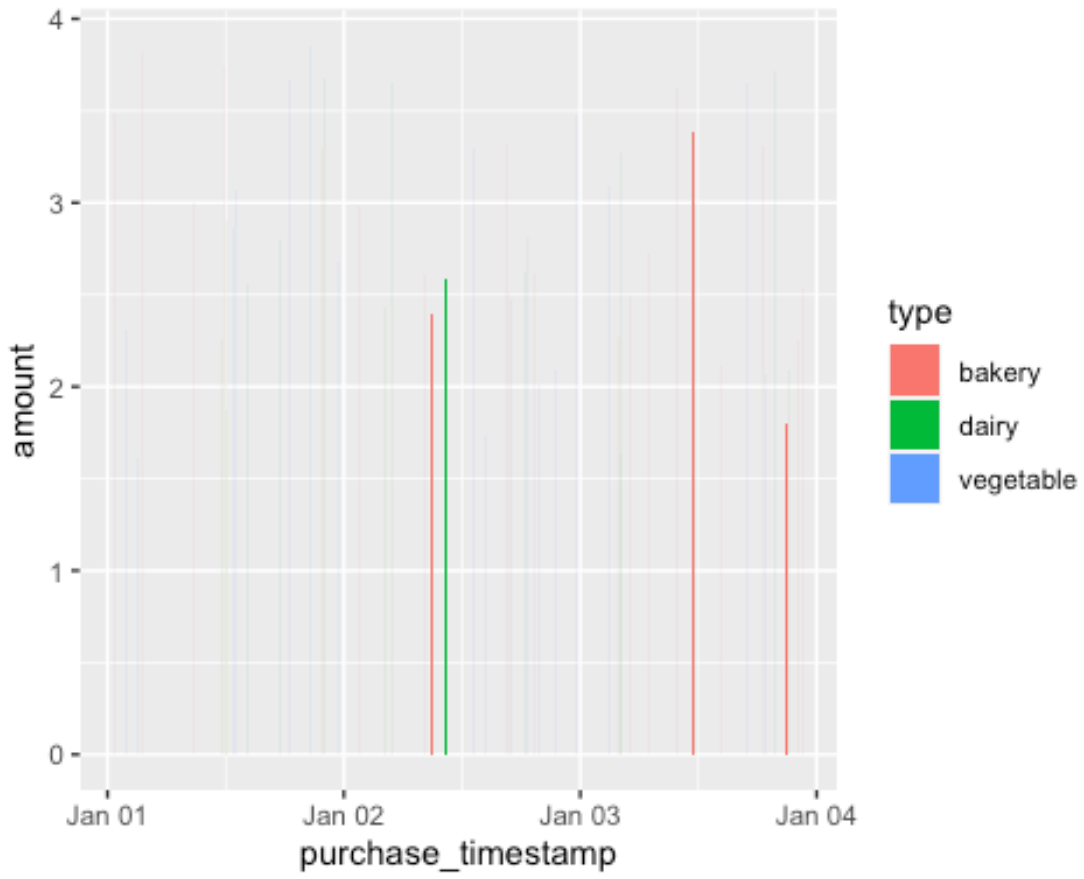
```
library(yaml)
mapping <- read_yaml("mapping.yml")
#read csv files
csv_files = c("detroit_purchases.csv", "new_york_purchases.csv")

# use plyr to remap values
library(plyr)
df1 <- read.csv(csv_files[1])
df2 <- read.csv(csv_files[2])

cat_name = names(mapping)

#map df2 to the other by yml file
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

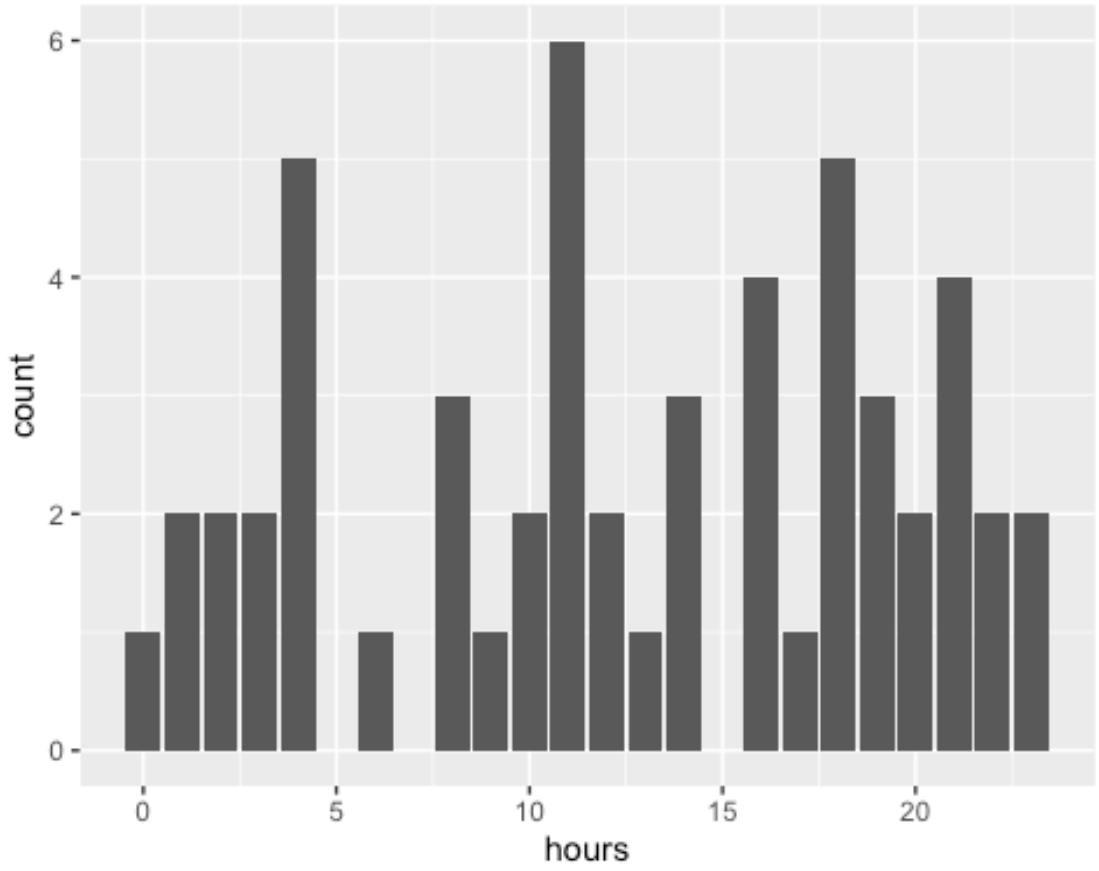
```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

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

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

df_t<-merge(df2, stack(mapping),by.x = 'type',by.y = 'values')
df_t<-subset(df_t,select=-c(type))
df_t<-df_t %>% dplyr::rename(type= ind)

# in df1 prices are chars, in df2 they are doubles
df1$amount <- as.double(sub("\\$", "", df1$amount))

#convert date to EST
df1$purchase_timestamp <- as.POSIXct(df1$purchase_timestamp,
                              format="%Y-%m-%d %H:%M:%S", tz='EST')
df_t$purchase_timestamp <- as.POSIXct(df_t$purchase_timestamp,
                                      format="%Y-%m-%d %H:%M:%S",
                                      tz='EST')



#Q1 merge data frames
df_all <- rbind(df1, df_t)

#Q2 Filter the data such that it only contains transactions for 1/2/2023.
df_s <- df_all[as.Date(df_all$purchase_timestamp) == "2023-01-02", ]

#Q3 i:Create a bar chart that is total revenue in each product line for 1/2.
library(ggplot2)
ggplot(df_all, aes(fill=type, y=amount, x=purchase_timestamp)) +
    geom_bar(position="dodge", stat="identity")
```

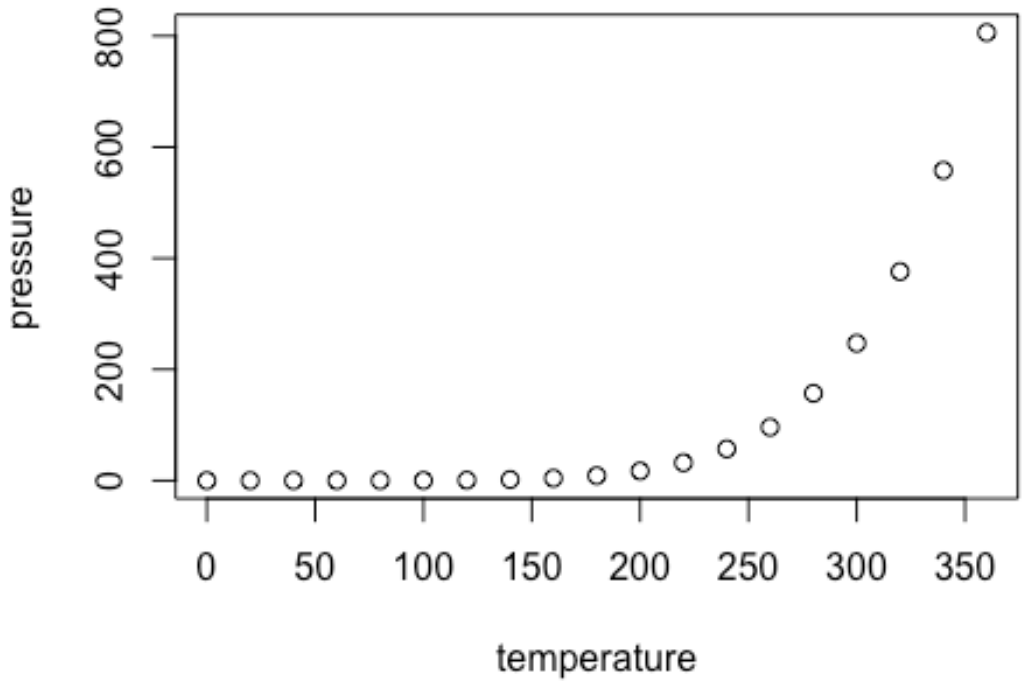



```r
#ii:Create a histogram for the number of items purchased for each hour of the day on 1/2.
#divide data in each hour
df_hours <- strptime(df_all$purchase_timestamp, format="%Y-%m-%d %H:%M:%S")
hours <-  as.numeric(format(df_hours, format="%H"))
#graph number of items purchased for each hour of the day on 1/2
ggplot(df_all, aes(hours )) + geom_bar()
```



Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.