Untitled

Charlene Garridos

2023-04-03

```
url <- "https://raw.githubusercontent.com/msuiitdmsgabriel/datasets-regression/main/salespeople.csv"
```

```
salespeople <- read.csv(url)
local_salespeople <- read.csv("salespeople.csv")
```

create df_report function

```
df_report <- function(df){paste("This dataframe contains", nrow(df), "rows and", ncol(df), "columns. The
```

```
df_report(mtcars)
```

```
## [1] "This dataframe contains 32 rows and 11 columns. There are 0 NA entries"
```

```
paste("This is how the paste", "functioning works")
```

```
## [1] "This is how the paste functioning works"
```

installing packages

```
my_packages <- c("MASS", "DescTools", "dplyr")
install.packages(my_packages)
```

```
## Installing packages into 'C:/Users/User/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)
```

```
## Error in contrib.url(repos, "source"): trying to use CRAN without setting a mirror
```

```
installed.packages("MASS")
```

```
##      Package LibPath Version Priority Depends Imports LinkingTo Suggests
##      Enhances License License_is_FOSS License_restricts_use OS_type Archs
##      MD5sum NeedsCompilation Built
```

```
my_packages <- c("DescTools", "dplyr")
```

```
installed.packages("my_packages")
```

```
##      Package LibPath Version Priority Depends Imports LinkingTo Suggests
##      Enhances License License_is_FOSS License_restricts_use OS_type Archs
##      MD5sum NeedsCompilation Built
```

```
library(MASS)
```

```
help(package = "MASS")
```

The pipe operator

```
sales <- subset(salespeople, subset = sales < 500)
```

```
mean(sales$sales)
```

```
## [1] 388.6684
```

```
mean(subset(salespeople, subset = sales <500) $ sales)
```

```
## [1] 388.6684
```

```
mean(subset(salespeople$sales, subset = salespeople$sales<500))
```

```
## [1] 388.6684
```

Load magrittr library to get the pipe operator

```
library(magrittr)
```

#Use the pipe operator to lay out the steps more logically

```
subset(salespeople$sales, subset = salespeople$sales < 500) %>%
mean()
```

```
## [1] 388.6684
```

```
library(magrittr)
subset(salespeople, subset = sales < 500)$sales %>%
mean()
```

```
## [1] 388.6684
```

```r
salespeople$sales %>% # start with all data
  subset(subset = salespeople$sales < 500) %>% # get the subsetted data
  mean() %>% # take the mean value
  round() # round to the nearest time
```

```
## [1] 389
```

Errors, warnings and messages

```r
subset(salespeople, subset = sales = 700)
```

```
## Error: <text>:1:36: unexpected '='
## 1: subset(salespeople, subset = sales =
##                                         ^
```

```r
subset(salespeople, subset = sales == 700)
```

```
## [1] promoted      sales         customer_rate performance
## <0 rows> (or 0-length row.names)
```

```r
head[salespeople]
```

```
## Error in head[salespeople]: object of type 'closure' is not subsettable
```

```r
head(salespeople)
```

```
##   promoted sales customer_rate performance
## 1        0   594          3.94           2
## 2        0   446          4.06           3
## 3        1   674          3.83           4
## 4        0   525          3.62           2
## 5        1   657          4.40           3
## 6        1   918          4.54           2
```

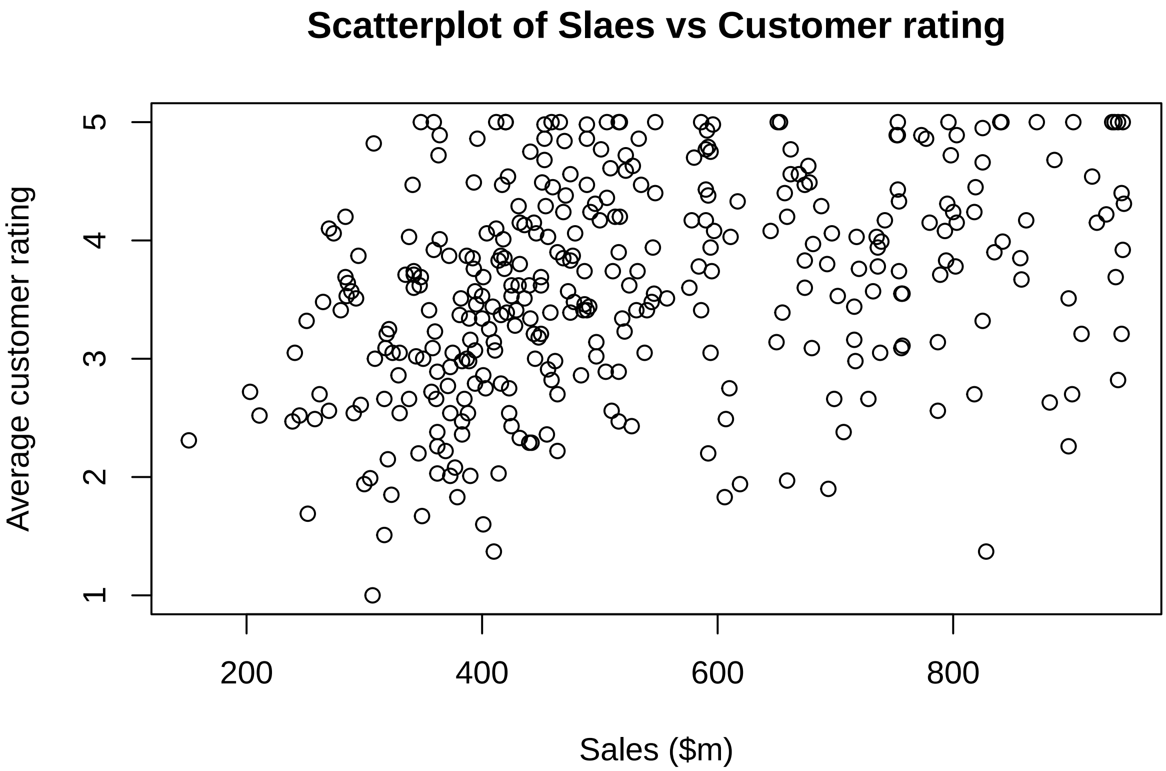
```r
salespeople[1,0]
```

```
## data frame with 0 columns and 1 row
```

plotting and graphing

scatter plot of customer_rate against sales

```
plot(x =  salespeople$sales, y =  salespeople$customer_rate, xlab = "Sales ($m)", ylab = "Average custo
```

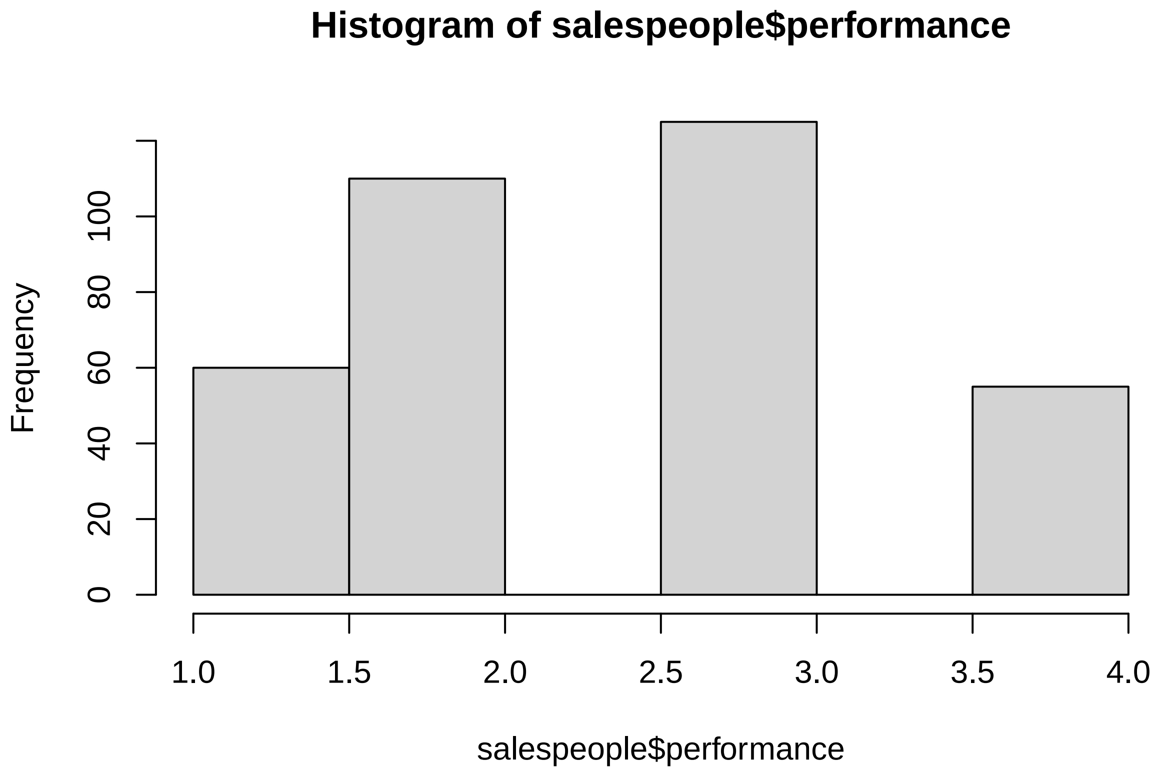


Convert performance ratings back to numeric data type for histogram

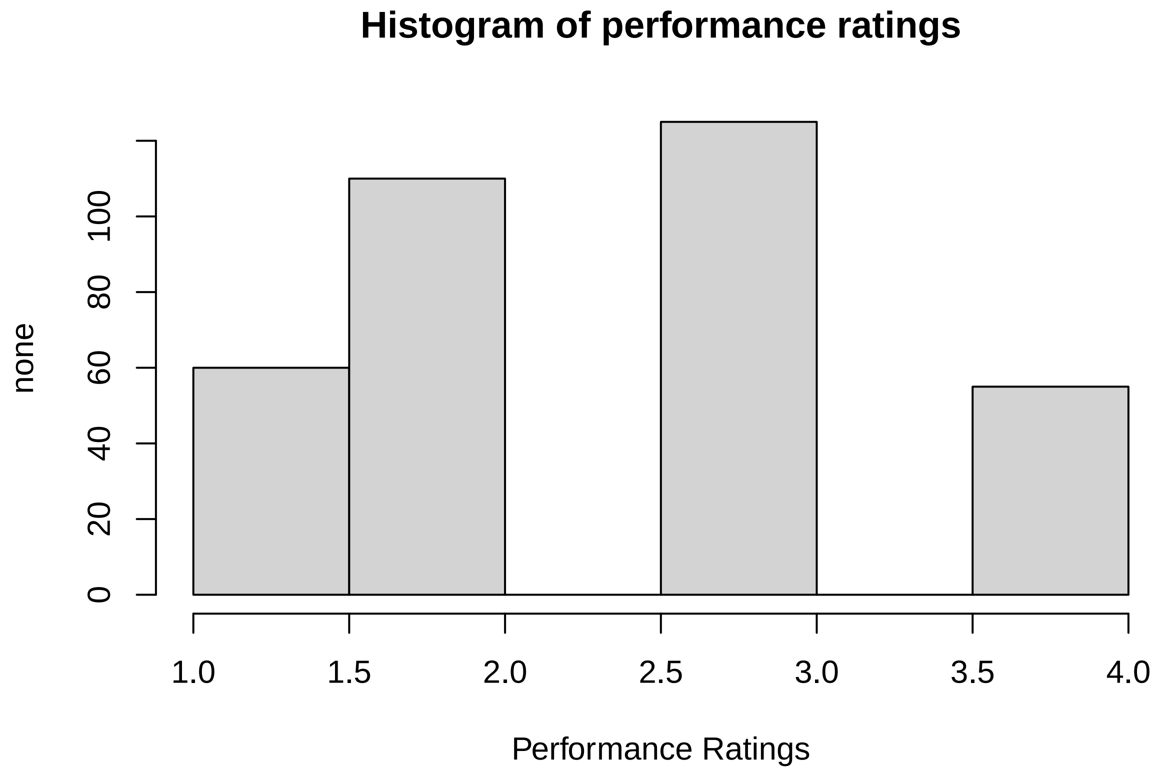
```
salespeople$performance <- as.numeric(salespeople$performance)
```

#histogram of performance ratings

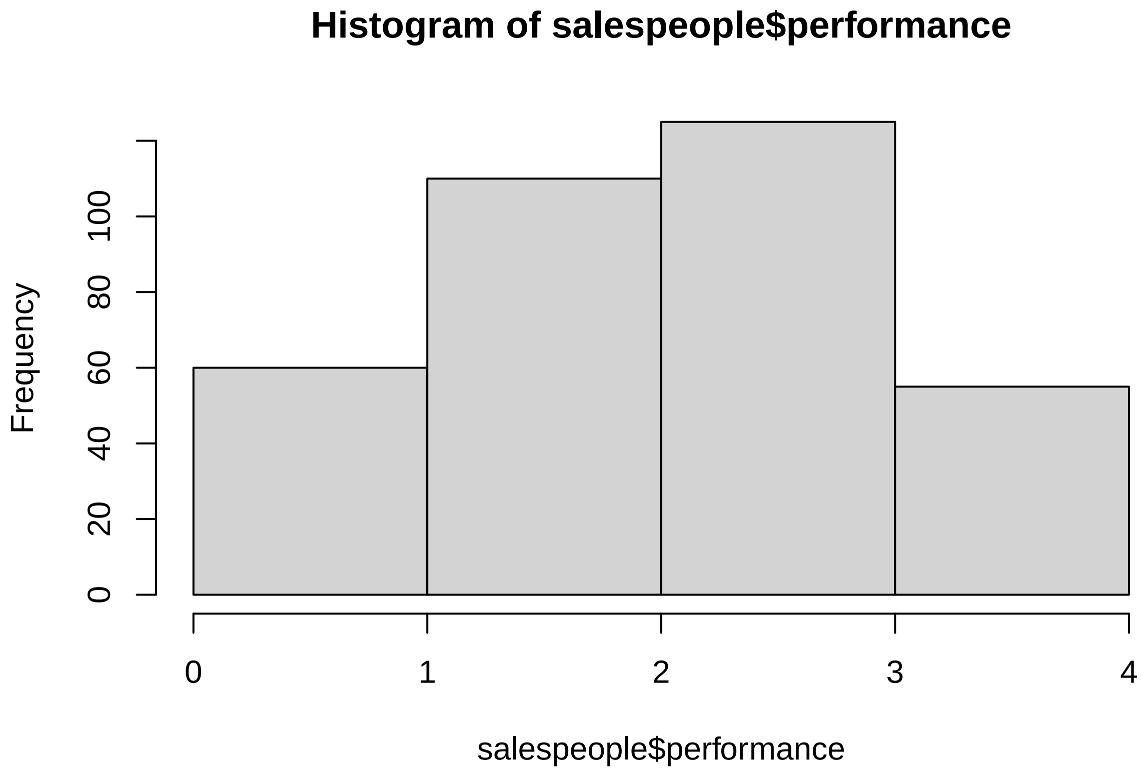
```
hist(salespeople$performance)
```



```
hist(salespeople$performance, xlab = "Performance Ratings", ylab = "none", main = "Histogram of performa
```

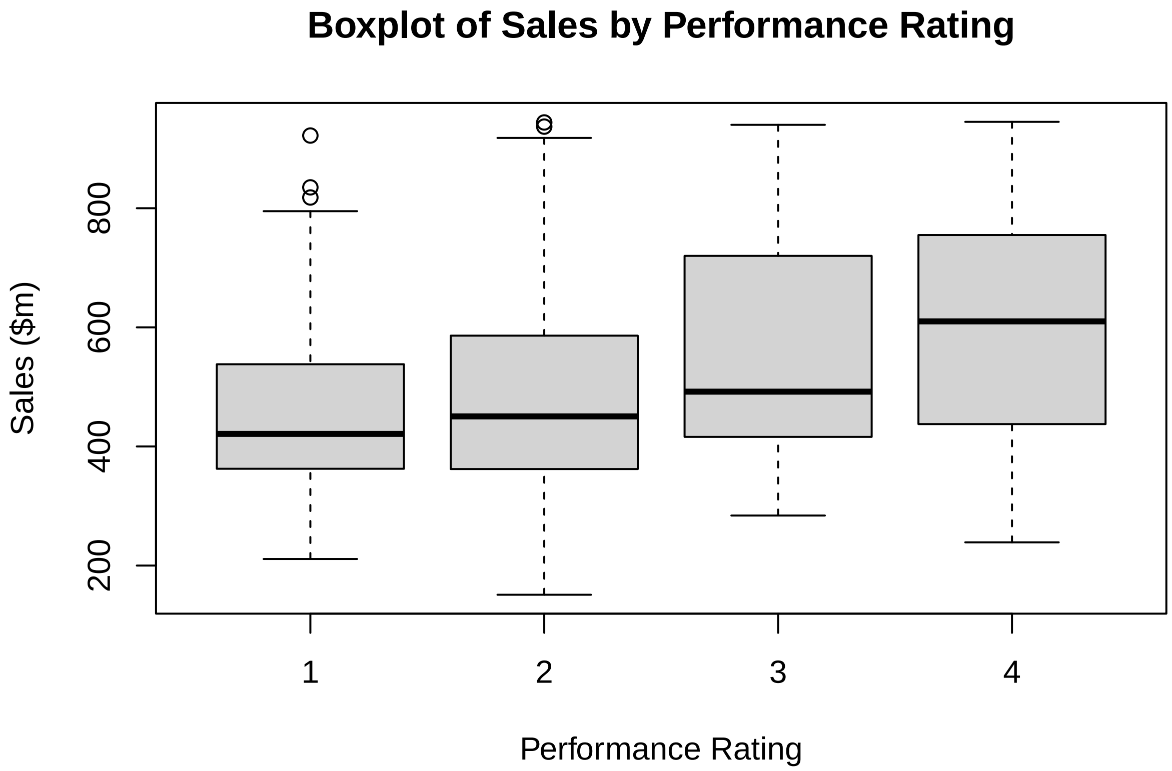


```
hist(salespeople$performance, breaks = 0:4)
```



box plot of sales by performance rating

```
boxplot(formula = sales ~ performance, data = salespeople,
        xlab = "Performance Rating", ylab = "Sales ($m)",
        main = "Boxplot of Sales by Performance Rating")
```



Specialist plotting and graphing packages

convert performance and promotion to categorical

```
install.packages(
   "ggplot2",
   repos = c("http://rstudio.org/_packages",
   "http://cran.rstudio.com")
)
```

```
## Installing package into 'C:/Users/User/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)

## Warning: unable to access index for repository http://rstudio.org/_packages/bin/windows/contrib/4.1:
##   cannot open URL 'http://rstudio.org/_packages/bin/windows/contrib/4.1/PACKAGES'

## package 'ggplot2' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\User\AppData\Local\Temp\RtmpyaqMIa\downloaded_packages
```

```
library(GGally)
```

```
## Loading required package: ggplot2
```

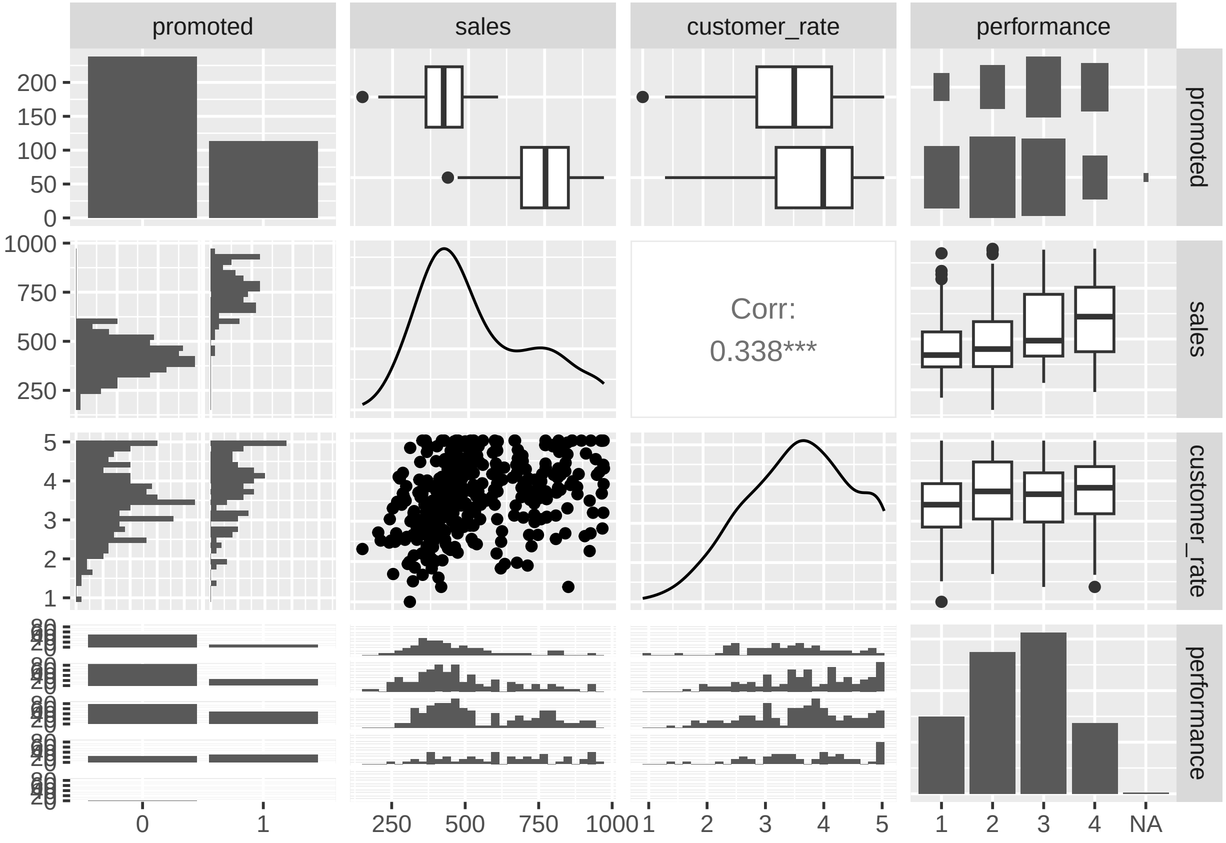
```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
salespeople$promoted <- as.factor(salespeople$promoted)
salespeople$performance <- as.factor(salespeople$performance)
```

Pairplot of salespeople

```
GGally::ggpairs(salespeople)
```

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


promoted
sales
customer_rate
performance
Corr:
0.338***
NA