# Exploratory Data Analysis

First thing to do is load any packages you may be using and load dataset (if external)

library(tidyverse)

## Registered S3 method overwritten by 'rvest':  
## method from  
## read\_xml.response xml2

## -- Attaching packages ------------------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1 v purrr 0.3.2   
## v tibble 2.1.1 v dplyr 0.8.0.1  
## v tidyr 0.8.3 v stringr 1.4.0   
## v readr 1.3.1 v forcats 0.4.0

## Warning: package 'ggplot2' was built under R version 3.6.2

## -- Conflicts ---------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

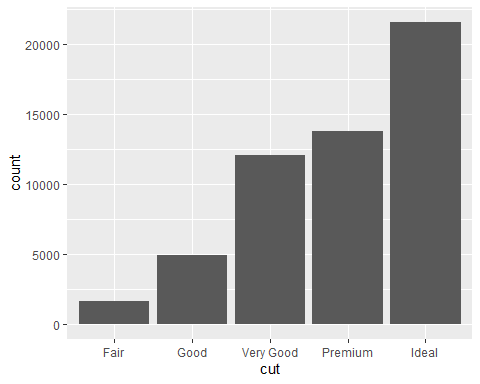
head(diamonds)

## # A tibble: 6 x 10  
## carat cut color clarity depth table price x y z  
## <dbl> <ord> <ord> <ord> <dbl> <dbl> <int> <dbl> <dbl> <dbl>  
## 1 0.23 Ideal E SI2 61.5 55 326 3.95 3.98 2.43  
## 2 0.21 Premium E SI1 59.8 61 326 3.89 3.84 2.31  
## 3 0.23 Good E VS1 56.9 65 327 4.05 4.07 2.31  
## 4 0.290 Premium I VS2 62.4 58 334 4.2 4.23 2.63  
## 5 0.31 Good J SI2 63.3 58 335 4.34 4.35 2.75  
## 6 0.24 Very Good J VVS2 62.8 57 336 3.94 3.96 2.48

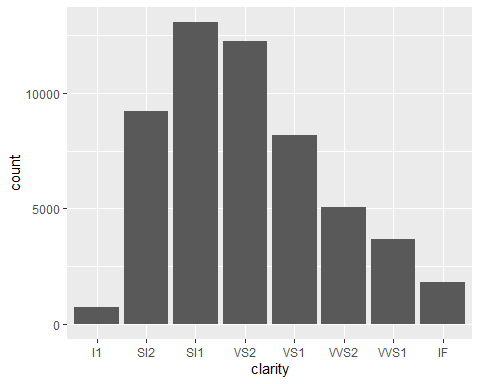
dsample <- diamonds[sample(nrow(diamonds),500),]

Displaying categorical variable (cut and calraity)

ggplot(data=diamonds, aes(cut)) +  
 geom\_bar()



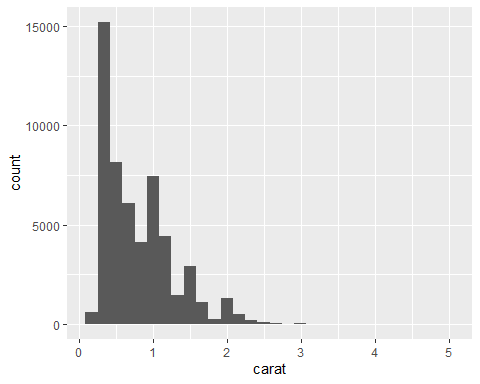
ggplot(data=diamonds, aes(clarity)) +  
 geom\_bar()



Continuous variables can have an infinte set of ordered values

ggplot(data=diamonds,aes(carat)) +  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

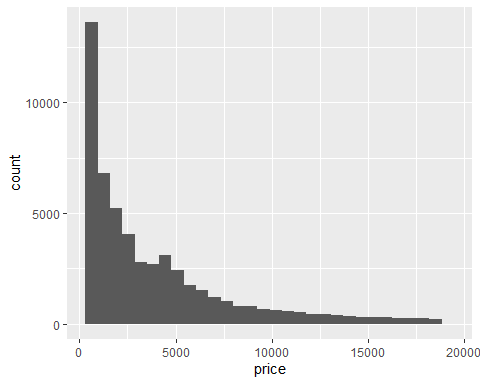


ggplot(data=diamonds,aes(carat)) +  
 geom\_histogram(binwidth=0.5)



ggplot(data=diamonds,aes(price)) +  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



ggplot(data=diamonds, aes(price, fill=color)) +  
 geom\_histogram(binwidth=500)

