EDA

Ying

2020/11/9

Lecture notes of Exploratory Data Analysis (EDA)

library(plyr)  
  
auto\_data <- read.table("auto.txt", header=FALSE, sep="\t")  
#auto\_data  
  
auto\_data <- rename(auto\_data, c(  
 "V1"="MPG",  
 "V2"="Cylinders",  
 "V3"="Displacement",  
 "V4"="Horsepower",  
 "V5"="Weight",  
 "V6"="Acceleration",  
 "V7"="ModelYear",  
 "V8"="Origin",  
 "V9"="CarName"  
 ))

x <- auto\_data[[3]]  
mean(x)

## [1] 193.4259

var(x)

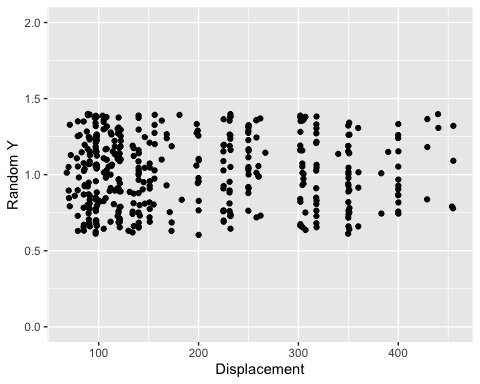
## [1] 10872.2

n <- length(x)  
n

## [1] 398

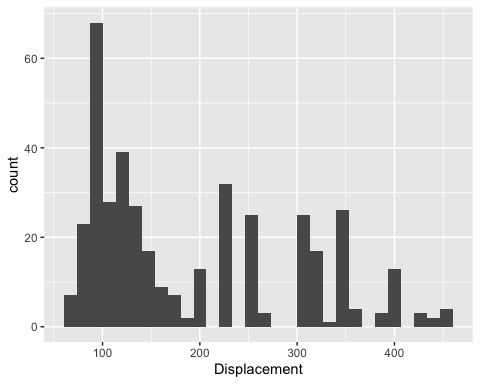
jitter plot, lossless

library(ggplot2)  
ggplot(auto\_data, aes(x=Displacement,y=rep(1,n)))+ geom\_jitter() + ylim(0,2) + ylab("Random Y")

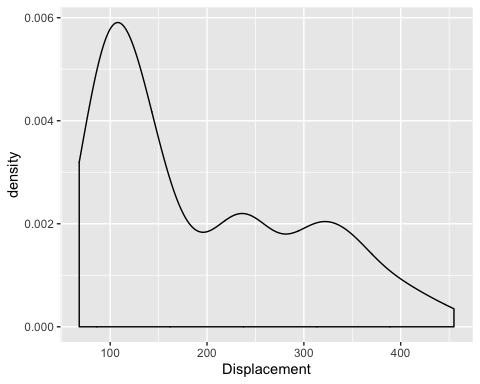


ggplot(auto\_data, aes(x=Displacement)) + geom\_histogram()

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



ggplot(auto\_data, aes(x=Displacement)) + geom\_density()



colMeans(auto\_data[,1:7], na.rm = TRUE)

## MPG Cylinders Displacement Horsepower Weight   
## 23.514573 5.454774 193.425879 104.469388 2970.424623   
## Acceleration ModelYear   
## 15.568090 76.010050

cov(auto\_data[,1:7], use = "na.or.complete")

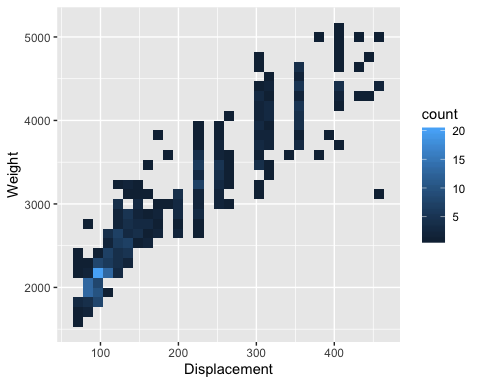
## MPG Cylinders Displacement Horsepower Weight  
## MPG 60.918142 -10.352928 -657.5852 -233.85793 -5517.4407  
## Cylinders -10.352928 2.909696 169.7219 55.34824 1300.4244  
## Displacement -657.585207 169.721949 10950.3676 3614.03374 82929.1001  
## Horsepower -233.857926 55.348244 3614.0337 1481.56939 28265.6202  
## Weight -5517.440704 1300.424363 82929.1001 28265.62023 721484.7090  
## Acceleration 9.115514 -2.375052 -156.9944 -73.18697 -976.8153  
## ModelYear 16.691477 -2.171930 -142.5721 -59.03643 -967.2285  
## Acceleration ModelYear  
## MPG 9.115514 16.691477  
## Cylinders -2.375052 -2.171930  
## Displacement -156.994435 -142.572133  
## Horsepower -73.186967 -59.036432  
## Weight -976.815253 -967.228457  
## Acceleration 7.611331 2.950462  
## ModelYear 2.950462 13.569915

cor(auto\_data[,1:7], use = "na.or.complete")

## MPG Cylinders Displacement Horsepower Weight  
## MPG 1.0000000 -0.7776175 -0.8051269 -0.7784268 -0.8322442  
## Cylinders -0.7776175 1.0000000 0.9508233 0.8429834 0.8975273  
## Displacement -0.8051269 0.9508233 1.0000000 0.8972570 0.9329944  
## Horsepower -0.7784268 0.8429834 0.8972570 1.0000000 0.8645377  
## Weight -0.8322442 0.8975273 0.9329944 0.8645377 1.0000000  
## Acceleration 0.4233285 -0.5046834 -0.5438005 -0.6891955 -0.4168392  
## ModelYear 0.5805410 -0.3456474 -0.3698552 -0.4163615 -0.3091199  
## Acceleration ModelYear  
## MPG 0.4233285 0.5805410  
## Cylinders -0.5046834 -0.3456474  
## Displacement -0.5438005 -0.3698552  
## Horsepower -0.6891955 -0.4163615  
## Weight -0.4168392 -0.3091199  
## Acceleration 1.0000000 0.2903161  
## ModelYear 0.2903161 1.0000000

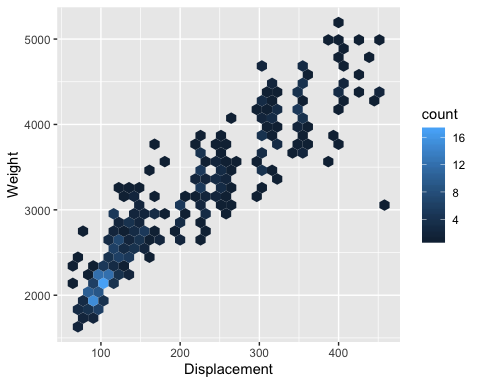
A 2d histogram generalised the univariate in the natural wayas the count of data points falling inside a given two-dimensional area.

ggplot(auto\_data, aes(x=Displacement,y=Weight)) + geom\_bin2d()

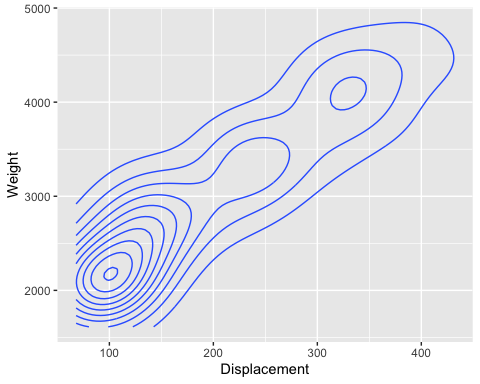


the area need not be a rectangle!

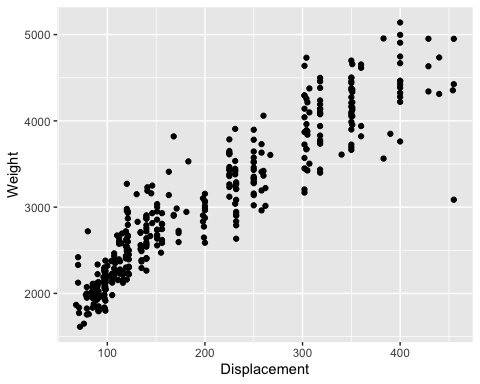
ggplot(auto\_data, aes(x=Displacement,y=Weight)) + geom\_hex()



ggplot(auto\_data, aes(x=Displacement,y=Weight)) + geom\_density\_2d()



ggplot(auto\_data, aes(x=Displacement,y=Weight)) + geom\_point()



ggplot(auto\_data, aes(x=Displacement,y=Weight)) + geom\_point(aes(color=Acceleration),size=auto\_data$Horsepower/50)+ ggtitle("Point Size is Proportional to Horsepower")

## Warning: Removed 6 rows containing missing values (geom\_point).

