Institut za matematiku i informatiku

Prirodno-matematički fakultet

Univerzitet u Kragujevcu

Seminarski rad iz predmeta

Uvod u nauku o podacima

Emisija CO2 kod vozila

Studenti: Nevena Bojović 47/2018

Milica Vučić 52/2018

Profesor: dr Branko Arsić

Avgust 2021.

Sadržaj

Opis problema

Ovaj skup podataka obuhvata detalje o tome kako emisija CO2 kod vozila može varirati u zavisnosti od različitih faktora. Skup podataka je preuzet sa zvanične veb stranice otvorenih podataka vlade Kanade. Skup podataka se sastoji od ukupno 7385 redova i 12 kolona. Postoji nekoliko skraćenica koje su korišćene za opis karakteristika.

- Make = Kompanija vozila
- Model = Model vozila
- Vehicle Class = Tip vozila u zavisnosti od njihove namene, kapaciteta i težine
- Engine Size = Veličina motora koji se koristi u litrima
- Cylinders = Broj cilindara
- Transmission = Vrsta menjača
- Fuel Type = Vrsta goriva koje se koristi
- Fuel Consumption City = Potrošnja goriva na gradskim putevima (L/100 km)
- Fuel Consumption Hwy = Potrošnja goriva na autoputevima (L/100 km)
- Fuel Consumption Comb = Kombinovana potrošnja goriva (55% grad, 45% autoput) prikazana je u L/100 km
- Fuel Consumption Comb mpg = Kombinovana potrošnja goriva u gradu i na autoputu (55% grad, 45% autoput) prikazana je u miljama po galonu (mpg)

Fuel Type (Tip goriva):

- X = Obični benzin
- Z = Vrhunski benzin
- D = Dizel
- E = Etanol (E85)
- N = Tečni naftni gas (TNG)

Transmission (Menjač)

- A = Automatski
- AM = Poluautomatski
- AS = Automatski sa odabranim pomeranjem
- AV = Automatski sa kontinuiranim pomeranjem
- M = Ručno
- 3 - 10 = Broj zupčanika

Model

- 4VD/4X4 = Pogon na sve točkove
- AVD = Pogon na sve točkove
- FFV = Vozilo sa fleksibilnim gorivom
- SVB = Kratko međuosovinsko rastojanje
- LVB = Dugo međuosovinsko rastojanje

- EVB = Produženo međuosovinsko rastojanje

Eksploratorna analiza podataka (EDA) odnosi se na proces izvođenja početnih istraživanja podataka kako bi se otkrili obrasci, uočile anomalije, testirale hipoteze i proverile pretpostavke uz pomoć zbirnih statistika i grafičkih prikaza.

Cilj tokom EDA-e je da se razvije razumljivost podataka. Najlakši način za to je da se koriste pitanja kao alati za istraživanje.

Kada postavimo pitanje, ono nam usmerava pažnju na određeni deo skupa podataka i pomaže nam da odlučimo koje grafike, modele ili transformacije da napravimo.

Cilj nam je da na osnovu podataka iz ovog skupa podataka predvidimo koliku emisiju CO2 proizvodi vozilo.

Priprema podataka

Učitavanje biblioteka i dataset-a i kratka deskripcija podataka:

```
library("dplyr")
library("ggpubr")
library("carData")
library("tidyverse")
library("msm")
library("MASS")
library("fitdistrplus")
library("logspline")
library("caret")
library("exptest")
library("rpart")

library("rpart.plot")
```

```
library("data.table")


co2 = read.csv('input\\CO2.csv',sep=',', stringsAsFactors = F)

dim(co2)

## [1] 7385   12

head(co2)

##    Make      Model Vehicle.Class Engine.Size.L. Cylinders Transmission
## 1 ACURA        ILX       COMPACT            2.0         4          AS5
## 2 ACURA        ILX       COMPACT            2.4         4           M6
## 3 ACURA ILX HYBRID       COMPACT            1.5         4          AV7
## 4 ACURA    MDX 4WD   SUV - SMALL            3.5         6          AS6
## 5 ACURA    RDX AWD   SUV - SMALL            3.5         6          AS6
## 6 ACURA        RLX      MID-SIZE            3.5         6          AS6
##   Fuel.Type Fuel.Consumption.City..L.100.km.
Fuel.Consumption.Hwy..L.100.km.
## 1         Z                             9.9
6.7
## 2         Z                            11.2
7.7
## 3         Z                             6.0
5.8
## 4         Z                            12.7
9.1
## 5         Z                            12.1
8.7
## 6         Z                            11.9
7.7
##   Fuel.Consumption.Comb..L.100.km. Fuel.Consumption.Comb..mpg.
## 1                              8.5                           33
## 2                              9.6                           29
## 3                              5.9                           48
## 4                             11.1                           25
## 5                             10.6                           27
## 6                             10.0                           28
##   CO2.Emissions.g.km.
## 1                 196
## 2                 221
## 3                 136
## 4                 255
## 5                 244
## 6                 230
```

Nazivi kolona (struktura dataset-a):

```
names(co2)
```

```
##  [1] "Make"                              "Model"
##  [3] "Vehicle.Class"                     "Engine.Size.L."
##  [5] "Cylinders"                         "Transmission"
##  [7] "Fuel.Type"                         "Fuel.Consumption.City..L.100.km."
##  [9] "Fuel.Consumption.Hwy..L.100.km."   "Fuel.Consumption.Comb..L.100.km."
## [11] "Fuel.Consumption.Comb..mpg."       "CO2.Emissions.g.km."
```

Struktura skupa podataka - proveravamo koje varijable mogu biti faktor promenljive.

```
str(co2)
```

```
## 'data.frame':    7385 obs. of  12 variables:
##  $ Make                            : chr  "ACURA" "ACURA" "ACURA" "ACURA"
...
##  $ Model                           : chr  "ILX" "ILX" "ILX HYBRID" "MDX
4WD" ...
##  $ Vehicle.Class                   : chr  "COMPACT" "COMPACT" "COMPACT"
"SUV - SMALL" ...
##  $ Engine.Size.L.                  : num  2 2.4 1.5 3.5 3.5 3.5 3.5 3.7
3.7 2.4 ...
##  $ Cylinders                       : int  4 4 4 6 6 6 6 6 6 4 ...
##  $ Transmission                    : chr  "AS5" "M6" "AV7" "AS6" ...
##  $ Fuel.Type                       : chr  "Z" "Z" "Z" "Z" ...
##  $ Fuel.Consumption.City..L.100.km.: num  9.9 11.2 6 12.7 12.1 11.9 11.8
12.8 13.4 10.6 ...
##  $ Fuel.Consumption.Hwy..L.100.km. : num  6.7 7.7 5.8 9.1 8.7 7.7 8.1 9
9.5 7.5 ...
##  $ Fuel.Consumption.Comb..L.100.km.: num  8.5 9.6 5.9 11.1 10.6 10 10.1
11.1 11.6 9.2 ...
##  $ Fuel.Consumption.Comb..mpg.     : int  33 29 48 25 27 28 28 25 24 31
...
##  $ CO2.Emissions.g.km.             : int  196 221 136 255 244 230 232 255
267 212 ...
```

```
summary(co2)
```

```
##      Make               Model           Vehicle.Class      Engine.Size.L.
##  Length:7385        Length:7385        Length:7385        Min.   :0.90
##  Class :character   Class :character   Class :character   1st Qu.:2.00
##  Mode  :character   Mode  :character   Mode  :character   Median :3.00
##                                                           Mean   :3.16
##                                                           3rd Qu.:3.70
##                                                           Max.   :8.40
##    Cylinders      Transmission        Fuel.Type
##  Min.   : 3.000   Length:7385        Length:7385
##  1st Qu.: 4.000   Class :character   Class :character
```

```
##  Median : 6.000   Mode  :character   Mode  :character
##  Mean   : 5.615
##  3rd Qu.: 6.000
##  Max.   :16.000
##  Fuel.Consumption.City..L.100.km. Fuel.Consumption.Hwy..L.100.km.
##  Min.   : 4.20                    Min.   : 4.000
##  1st Qu.:10.10                    1st Qu.: 7.500
##  Median :12.10                    Median : 8.700
##  Mean   :12.56                    Mean   : 9.042
##  3rd Qu.:14.60                    3rd Qu.:10.200
##  Max.   :30.60                    Max.   :20.600
##  Fuel.Consumption.Comb..L.100.km. Fuel.Consumption.Comb..mpg.
##  Min.   : 4.10                    Min.   :11.00
##  1st Qu.: 8.90                    1st Qu.:22.00
##  Median :10.60                    Median :27.00
##  Mean   :10.98                    Mean   :27.48
##  3rd Qu.:12.60                    3rd Qu.:32.00
##  Max.   :26.10                    Max.   :69.00
##  CO2.Emissions.g.km.
##  Min.   : 96.0
##  1st Qu.:208.0
##  Median :246.0
##  Mean   :250.6
##  3rd Qu.:288.0
##  Max.   :522.0
```

Proveravamo da li postoje NA vrednosti u našem dataset-u:

```
colSums(is.na(co2))
```

```
##                              Make                             Model
##                                 0                                 0
##                     Vehicle.Class                     Engine.Size.L.
##                                 0                                 0
##                         Cylinders                       Transmission
##                                 0                                 0
##                         Fuel.Type Fuel.Consumption.City..L.100.km.
##                                 0                                 0
##  Fuel.Consumption.Hwy..L.100.km. Fuel.Consumption.Comb..L.100.km.
##                                 0                                 0
##      Fuel.Consumption.Comb..mpg.              CO2.Emissions.g.km.
##                                 0                                 0
```

Pošto ne postoje NA vrednosti, unećemo ih proizvoljno.

Promena naziva kolona:

Napomena: Potrošnja se podrazumeva da je L/100km za: ConsumptionCity, ConsumptionHwy, ConsumptionComb, ConsumptionCombMpg

```
names(co2)[names(co2)=="Fuel.Consumption.City..L.100.km."]<-"ConsumptionCity"
names(co2)[names(co2)=="Fuel.Consumption.Hwy..L.100.km."]<-"ConsumptionHwy"
names(co2)[names(co2)=="Fuel.Consumption.Comb..L.100.km."]<-"ConsumptionComb"
names(co2)[names(co2)=="Fuel.Consumption.Comb..mpg."]<-"ConsumptionCombMpg"
names(co2)[names(co2)=="Vehicle.Class"]<-"VehicleClass"
names(co2)[names(co2)=="Engine.Size.L."]<-"EngineSize"  #velicina motora
litrima
names(co2)[names(co2)=="CO2.Emissions.g.km."]<-"CO2Emissions" # u g/km
names(co2)[names(co2)=="Fuel.Type"]<-"FuelType" # u g/km
```

Dodavanje NA vrednosti u kolonama ConsumptionCity, ConsumptionHwy, FuelType:

```
ind = sample(1:dim(co2)[1], 590, replace=FALSE)
co2$ConsumptionCity[ind]=NA
sum(is.na(co2$ConsumptionCity))
```

```
## [1] 590
```

```
set.seed(654321)
ind = sample(1:dim(co2)[1], 730, replace=FALSE)
co2$ConsumptionHwy[ind]=NA
sum(is.na(co2$ConsumptionHwy))
```

```
## [1] 730
```

```
set.seed(123456)
ind = sample(1:dim(co2)[1], 152, replace=FALSE)
co2$FuelType[ind]=NA
sum(is.na(co2$FuelType))
```

```
## [1] 152
```

Sada se u skupu nalazi skoro 20% (19.93%) nedostajućih vrednosti koje ćemo popuniti [(590+730+152)/7385].

Ponovna provera NA vrednosti:

```
colSums(is.na(co2))
```

```
##               Make              Model       VehicleClass
EngineSize
```

```
##                   0                   0                   0
0
##           Cylinders        Transmission            FuelType
ConsumptionCity
##                   0                   0                 152
590
##     ConsumptionHwy    ConsumptionComb ConsumptionCombMpg
CO2Emissions
##                 730                   0                   0
0
```

Promena u *factor* promenljive: Cylinders, FuelType, VehicleClass, Make, Transmission:

```
table(co2$Cylinders)
```

```
##
##    3    4    5    6    8   10   12   16
##   95 3220   26 2446 1402   42  151    3
```

```
co2$Cylinders = factor(co2$Cylinders)

table(co2$FuelType)
```

```
##
##    D    E    N    X    Z
##  174  362    1 3567 3129
```

```
co2$FuelType = factor(co2$FuelType)


table(co2$VehicleClass)
```

```
##
##                   COMPACT                 FULL-SIZE                   MID-SIZE
##                      1022                       639                      1133
##               MINICOMPACT                   MINIVAN     PICKUP TRUCK - SMALL
##                       326                        80                       159
##  PICKUP TRUCK - STANDARD  SPECIAL PURPOSE VEHICLE STATION WAGON - MID-SIZE
##                       538                        77                        53
##    STATION WAGON - SMALL               SUBCOMPACT              SUV - SMALL
##                       252                       606                      1217
##           SUV - STANDARD               TWO-SEATER              VAN - CARGO
##                       735                       460                        22
##          VAN - PASSENGER
##                        66
```

```
co2$VehicleClass = factor(co2$VehicleClass)

table(co2$Make)

##
##         ACURA    ALFA ROMEO  ASTON MARTIN          AUDI       BENTLEY
##            72            30            47           286            46
##           BMW       BUGATTI         BUICK      CADILLAC     CHEVROLET
##           527             3           103           158           588
##      CHRYSLER         DODGE          FIAT          FORD       GENESIS
##            88           246            73           628            25
##           GMC         HONDA       HYUNDAI      INFINITI        JAGUAR
##           328           214           210           108           160
##          JEEP           KIA   LAMBORGHINI    LAND ROVER         LEXUS
##           251           231            41            85           178
##       LINCOLN      MASERATI         MAZDA MERCEDES-BENZ          MINI
##            96            61           180           419           204
##    MITSUBISHI        NISSAN       PORSCHE           RAM   ROLLS-ROYCE
##            95           259           376            97            50
##         SCION         SMART           SRT        SUBARU        TOYOTA
##            22             7             2           140           330
##    VOLKSWAGEN         VOLVO
##           197           124

co2$Make = factor(co2$Make)

table(co2$Transmission)

##
##  A10   A4   A5   A6   A7   A8   A9  AM5  AM6  AM7  AM8  AM9 AS10  AS4  AS5
AS6
##   31   65   84  789   53  490  339    4  132  445   62    3  168    2   26
1324
##  AS7  AS8  AS9   AV AV10  AV6  AV7  AV8   M5   M6   M7
##  319 1211   77  295   11  113  118   39  193  901   91

co2$Transmission = factor(co2$Transmission)
```

Provera distribucije (Normalnost distribucije) - statističke metode tj. testovi koji su pogodni za proveru normalnosti distribucije su Kolmogorov-Smirnov i Shapiro-Wilkov test. Kod Shapiro-Wilkovog testa postoji ograničenje za obim uzorka (3000-5000 podataka), dok kod Kolmogorov-Smirnovog testa ne postoji ograničenje.

```
CS = filter(co2, !is.na(co2$ConsumptionCity))
ks.test(CS$ConsumptionCity, "pnorm", mean=mean(CS$ConsumptionCity),
sd=sd(CS$ConsumptionCity))
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  CS$ConsumptionCity
## D = 0.072201, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
CH = filter(co2, !is.na(co2$ConsumptionHwy))
ks.test(CH$ConsumptionHwy, "pnorm", mean=mean(CH$ConsumptionHwy),
sd=sd(CH$ConsumptionHwy))
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  CH$ConsumptionHwy
## D = 0.077364, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
CC = filter(co2, !is.na(co2$ConsumptionComb))
ks.test(CC$ConsumptionComb, "pnorm", mean=mean(CC$ConsumptionComb),
sd=sd(CC$ConsumptionComb))
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  CC$ConsumptionComb
## D = 0.071934, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
CM = filter(co2, !is.na(co2$ConsumptionCombMpg))
ks.test(CM$ConsumptionCombMpg, "pnorm", mean=mean(CM$ConsumptionCombMpg),
sd=sd(CM$ConsumptionCombMpg))
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  CM$ConsumptionCombMpg
## D = 0.07523, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

Za sve kontinualne promenljive možemo da zaključimo da raspodela nije normalna (p nije veće od 0.05).

Sada ćemo proveriti Shapiro-Wilkovim testom na obimu uzorka od 5000:

```
shapiro.test(sample(CC$ConsumptionCity, 5000))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sample(CC$ConsumptionCity, 5000)
## W = 0.96466, p-value < 2.2e-16
```

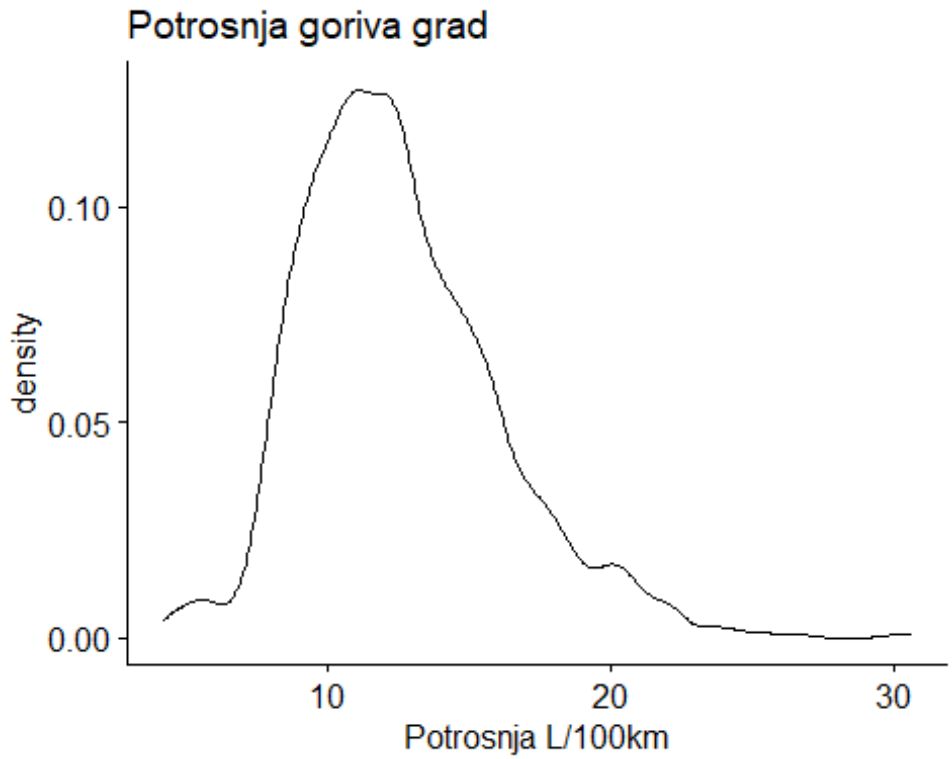
```
shapiro.test(sample(CH$ConsumptionCity, 5000))

##
##  Shapiro-Wilk normality test
##
## data:  sample(CH$ConsumptionCity, 5000)
## W = 0.96352, p-value < 2.2e-16

shapiro.test(sample(CC$ConsumptionCity, 5000))

##
##  Shapiro-Wilk normality test
##
## data:  sample(CC$ConsumptionCity, 5000)
## W = 0.96281, p-value < 2.2e-16

shapiro.test(sample(CM$ConsumptionCity, 5000))

##
##  Shapiro-Wilk normality test
##
## data:  sample(CM$ConsumptionCity, 5000)
## W = 0.96253, p-value < 2.2e-16
```

Ovaj test nam takođe potvrđuje da raspodela nije normalna, s tim što uzorak može biti maksimalno obima 5000 (p nije veće od 0.05).

Provera distribucije (Normalnost distribucije) - vizuelne metode

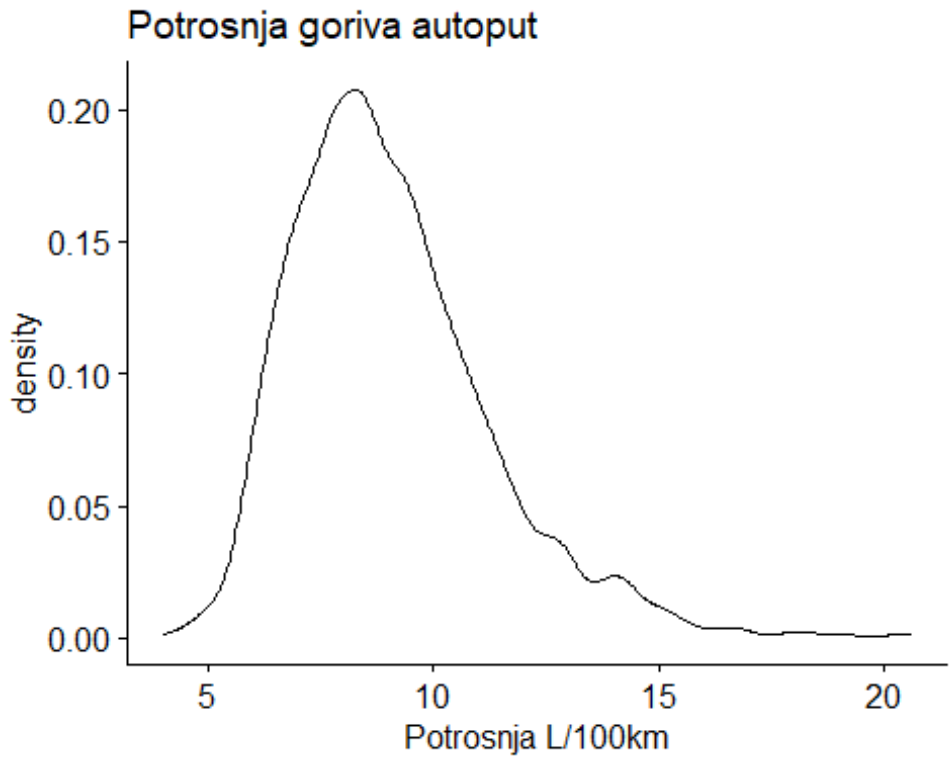
Vizuelno postoji sličnost sa Poasonovom raspodelom, što ćemo i proveriti statističkim testom nakon vizuelnih metoda.

```
ggdensity(co2$ConsumptionCity,
          main = "Potrosnja goriva grad",
          xlab = "Potrosnja L/100km")
```



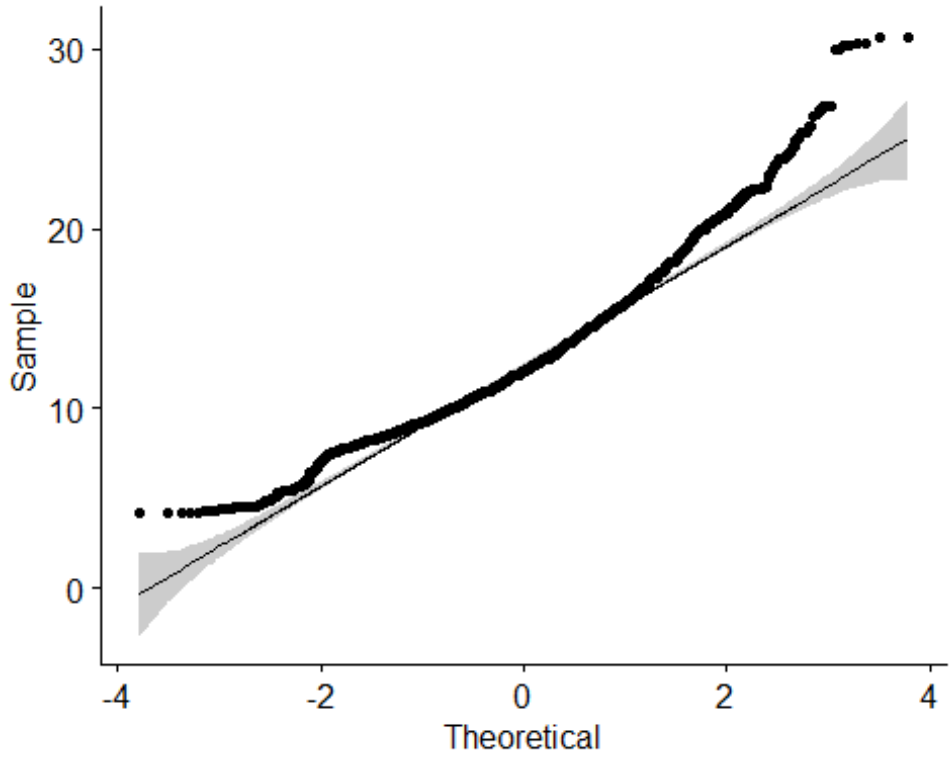
Asimetrija (skewnes) je prisutna na levoj strani.

```
ggdensity(co2$ConsumptionHwy,
          main = "Potrosnja goriva autoput",
          xlab = "Potrosnja L/100km")
```

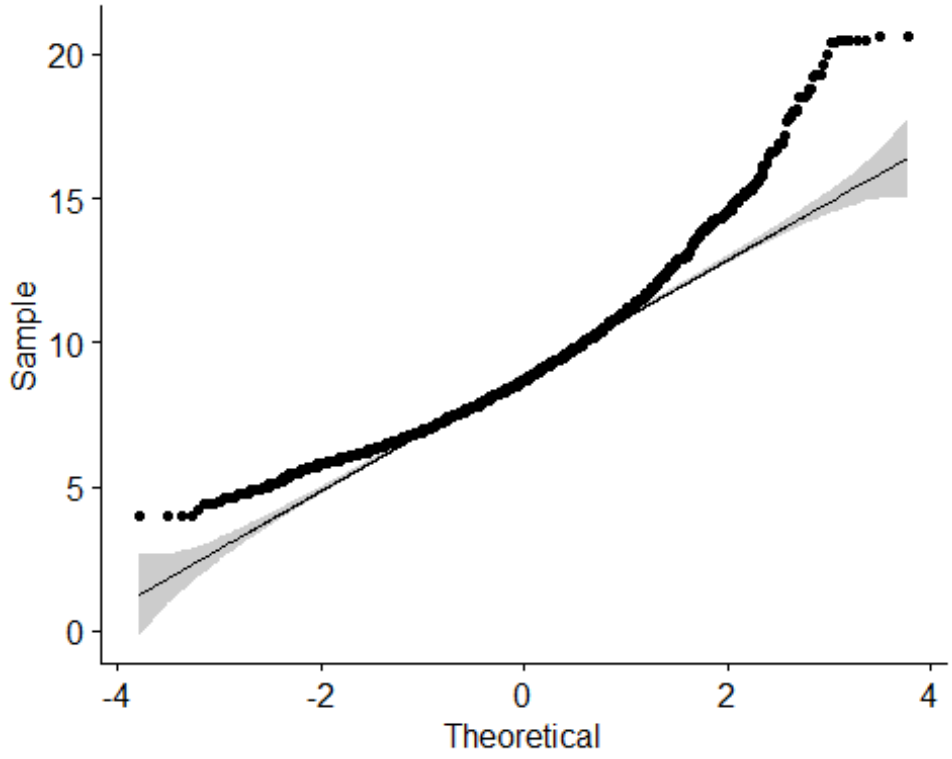


Asimetrija (skewnes) je prisutna na levoj strani.

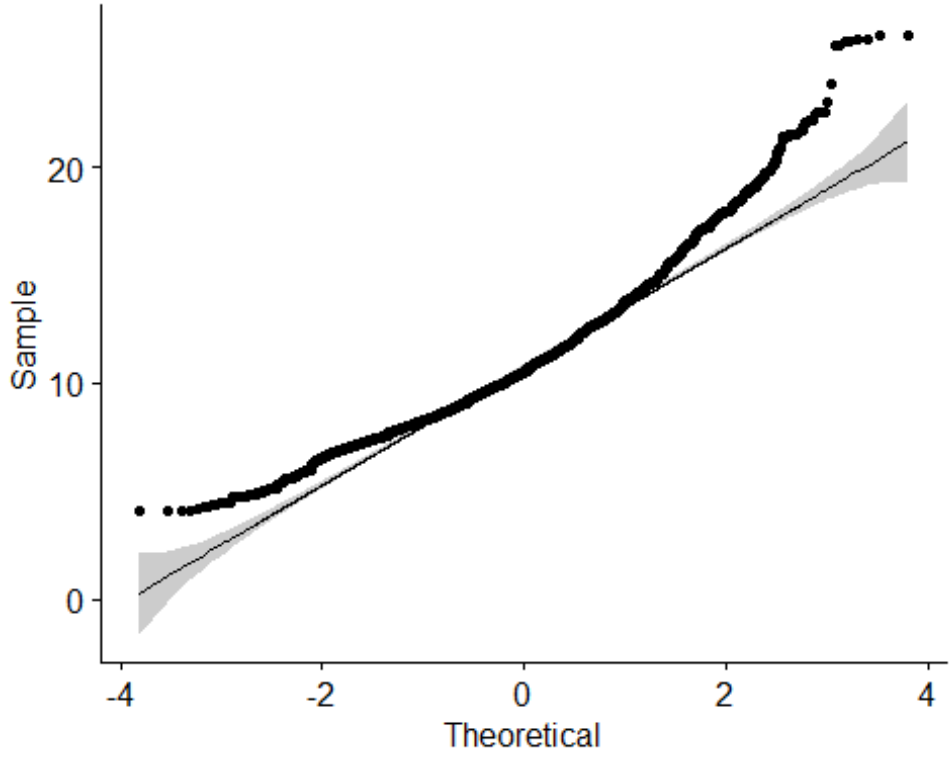
```
ggqqplot(co2$ConsumptionCity)
```



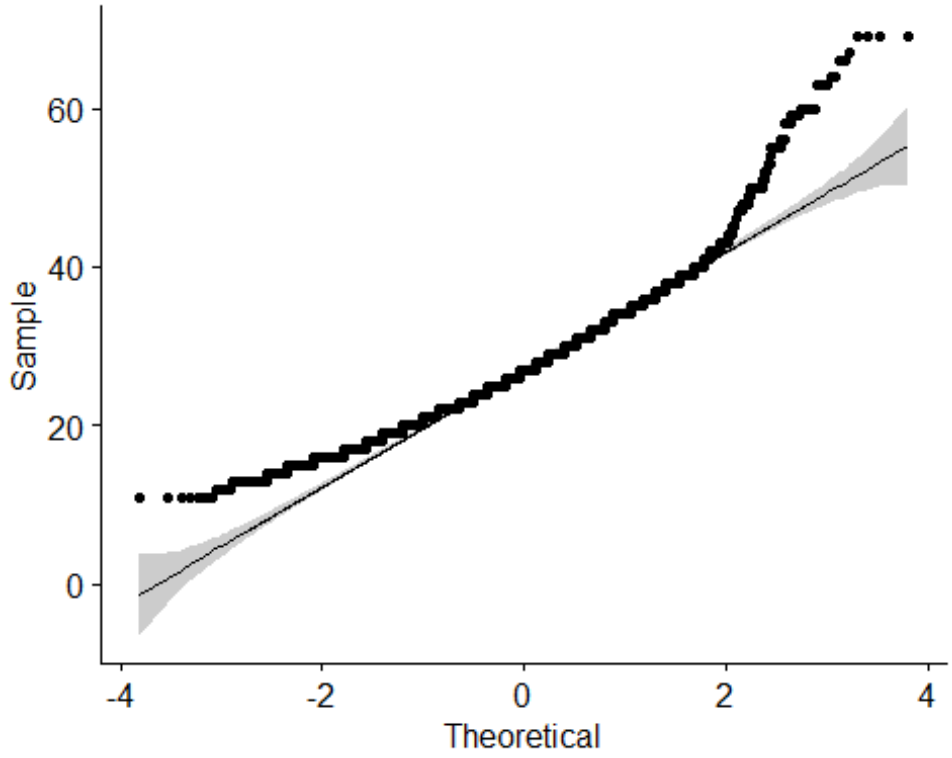
```
ggqqplot(co2$ConsumptionHwy)
```




```
ggqqplot(co2$ConsumptionComb)
```




```
ggqqplot(co2$ConsumptionCombMpg)
```



Sa grafika jasno vidimo da raspodela nije normalna, uniformna, hi-kvadrat, Studentova. Proverićemo još da li je raspodela eksponencijalna, s obzirom na izgled Q-Q plot-a.

```
ks.exp.test(CS$ConsumptionCity, nrepl=2000)

##
##  Kolmogorov-Smirnov test for exponentiality
##
## data:  CS$ConsumptionCity
## KSn = 0.42679, p-value < 2.2e-16

ks.exp.test(CH$ConsumptionHwy, nrepl=2000)

##
##  Kolmogorov-Smirnov test for exponentiality
##
## data:  CH$ConsumptionHwy
## KSn = 0.45378, p-value < 2.2e-16

ks.exp.test(CC$ConsumptionComb, nrepl=2000)

##
##  Kolmogorov-Smirnov test for exponentiality
##
## data:  CC$ConsumptionComb
## KSn = 0.43449, p-value < 2.2e-16

ks.exp.test(CM$ConsumptionCombMpg, nrepl=2000)

##
##  Kolmogorov-Smirnov test for exponentiality
##
## data:  CM$ConsumptionCombMpg
## KSn = 0.4246, p-value < 2.2e-16
```

Raspodela nije ni eksponencijalna, s obzirom na vrednost p.

Proverićemo samo još Poasonovu raspodelu, statističkim testom.

```
not_null_ConsumptionCity = filter(co2, !is.na(ConsumptionCity))
#Provera Poasonove raspodele
dispersion_test <- function(x)
{
  res <- 1-2 * abs((1 - pchisq((sum((x - mean(x))^2)/mean(x)), length(x) -
1))-0.5)

  cat("Dispersion test of count data:\n",
      length(x), " data points.\n",
      "Mean: ",mean(x),"\n",
      "Variance: ",var(x),"\n",
```

```
      "Probability of being drawn from Poisson distribution: ",
      round(res, 3),"\n", sep = "")

  invisible(res)
}
dispersion_test(not_null_ConsumptionCity$ConsumptionCity)
```

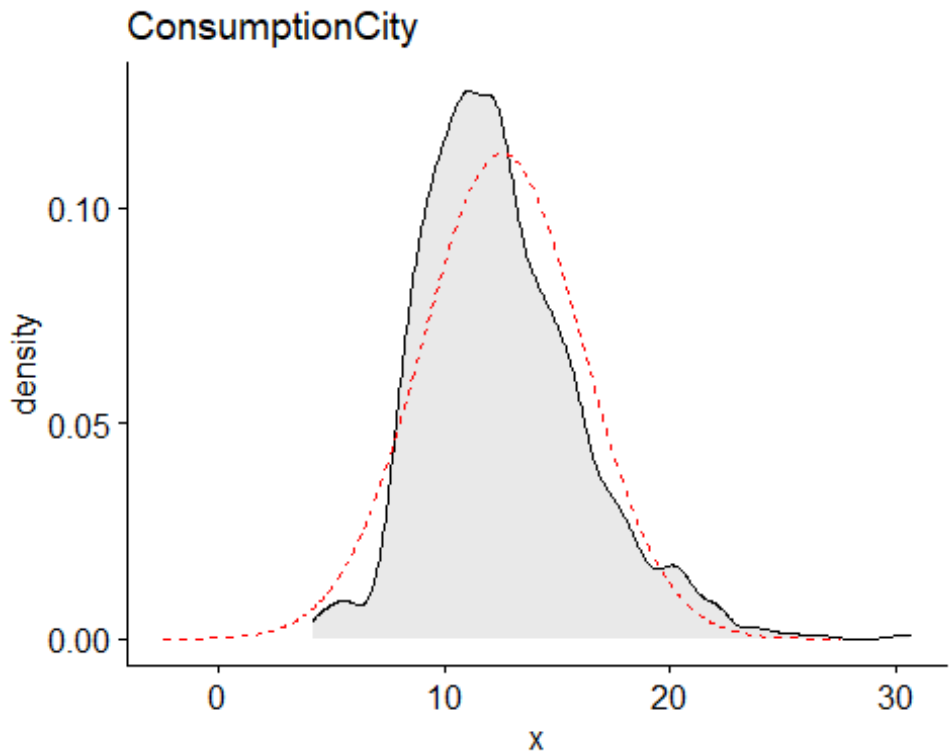
```
## Dispersion test of count data:
## 6795 data points.
## Mean: 12.55635
## Variance: 12.23603
## Probability of being drawn from Poisson distribution: 0.135
```

Verovatnoća da je Poasonova raspodela je 0.257, što nije dovoljno da se prihvati nulta hipoteza, pa je prema tome odbacujemo, prihvatamo alternativnu hipotezu, da raspodela nije Poasonova.

Popunjavanje NA vrednosti

Izgled distribucije za ComsumptionCity:

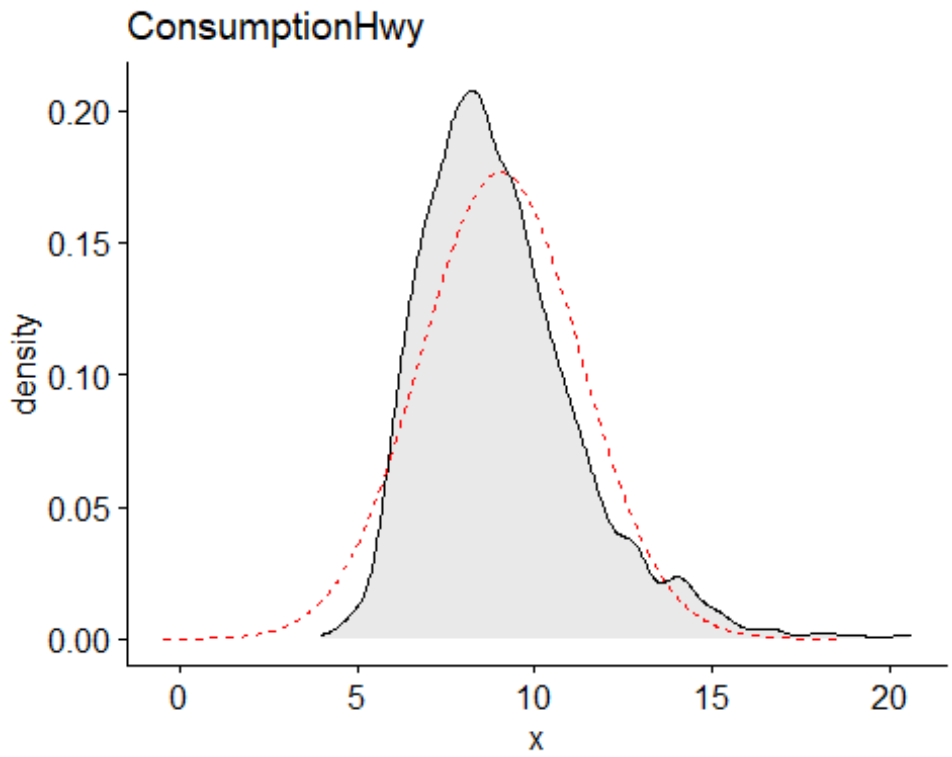
```
ggdensity(co2$ConsumptionCity, fill = "lightgray", title = "ConsumptionCity")
+
  stat_overlay_normal_density(color = "red", linetype = "dashed")
```



Pokušaćemo da svedemo ovu distribuciju na normalnu, a nakon toga ćemo da popunimo nedostajuće vrednosti srednjom vrednosti novodobijene normalne distribucije. Kasnije ćemo vrednosti vratiti na staro (inverzna funkcija logaritma).

Izgled distribucije za ComsumptionHwy

```
ggdensity(co2$ConsumptionHwy, fill = "lightgray", title = "ConsumptionHwy") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
```



Sveli smo na normalnu distribuciju, popunili srednjom vrednošću, pa vratili na originalnu distribuciju.

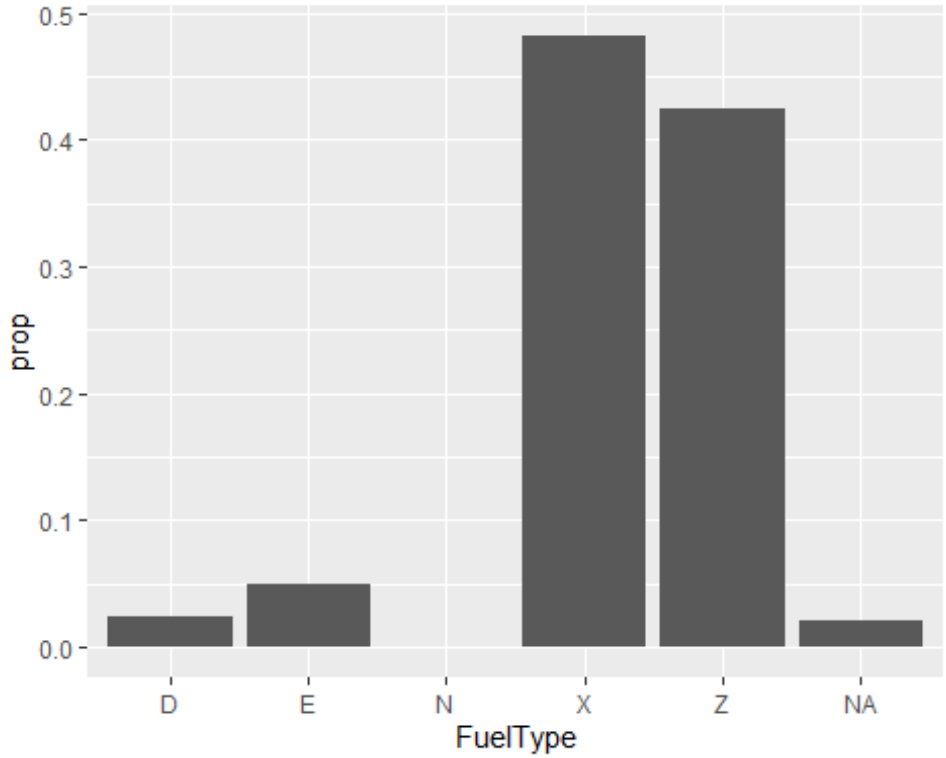
```
co2$ConsumptionCity <- log10(co2$ConsumptionCity)
co2$ConsumptionHwy <- log10(co2$ConsumptionHwy)
CityMean = mean(co2$ConsumptionCity, na.rm=TRUE)
HwyMean = mean(co2$ConsumptionHwy, na.rm=TRUE)
co2$ConsumptionCity[is.na(co2$ConsumptionCity)] = CityMean
co2$ConsumptionHwy[is.na(co2$ConsumptionHwy)] = HwyMean
co2$ConsumptionCity = 10^co2$ConsumptionCity
co2$ConsumptionHwy = 10^co2$ConsumptionHwy
```

Zaključujemo da popunjavanje NA vrednosti medianom daje bolje rezultate (manji pik na grafiku), pa se odlučujemo za taj način.

```
CityMedian = median(co2$ConsumptionCity, na.rm = TRUE)
HwyMedian = median(co2$ConsumptionHwy, na.rm=TRUE)
co2$ConsumptionCity[is.na(co2$ConsumptionCity)] = CityMedian
co2$ConsumptionHwy[is.na(co2$ConsumptionHwy)] = HwyMedian
```

Za popunjavanje NA vrednosti u koloni FuelType potrebno nam je da znamo u kom procentu se pojavljuje koja vrsta goriva.

```
ggplot(data=co2) + geom_bar(mapping=aes(x=FuelType, y=..prop.., group=1))
```



Možemo primetiti da su skoro podjednako zastupljena goriva tipa X (obični benzin) i Z (vrhunski benzin). Nedostajuće vrednosti u koloni FuelType popunićemo random tipom goriva, ili X ili Z.

Popunjavanje NA vrednosti u koloni FuelType:

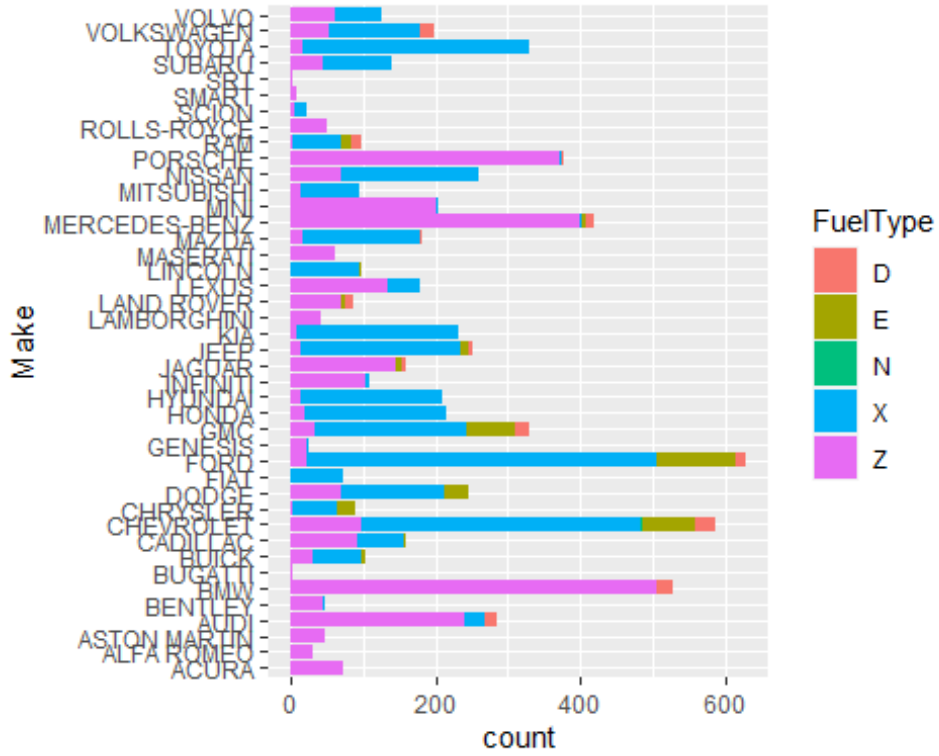
```
sum(is.na(co2$FuelType))

## [1] 152

sample_size = floor(0.5 * nrow(co2))
set.seed(42)
train_ind = sample(seq(1, nrow(co2)), size = sample_size)
co2_1 = co2[train_ind,]
co2_2 = co2[-train_ind,]
co2_1$FuelType[is.na(co2_1$FuelType)] = "X"
```

```
co2_2$FuelType[is.na(co2_2$FuelType)] = "Z"
co2 = rbind(co2_1,co2_2)
```

Analiza podataka

1) Sledeći grafik predstavlja broj vozila po tipu goriva za svaku kompaniju (Make).

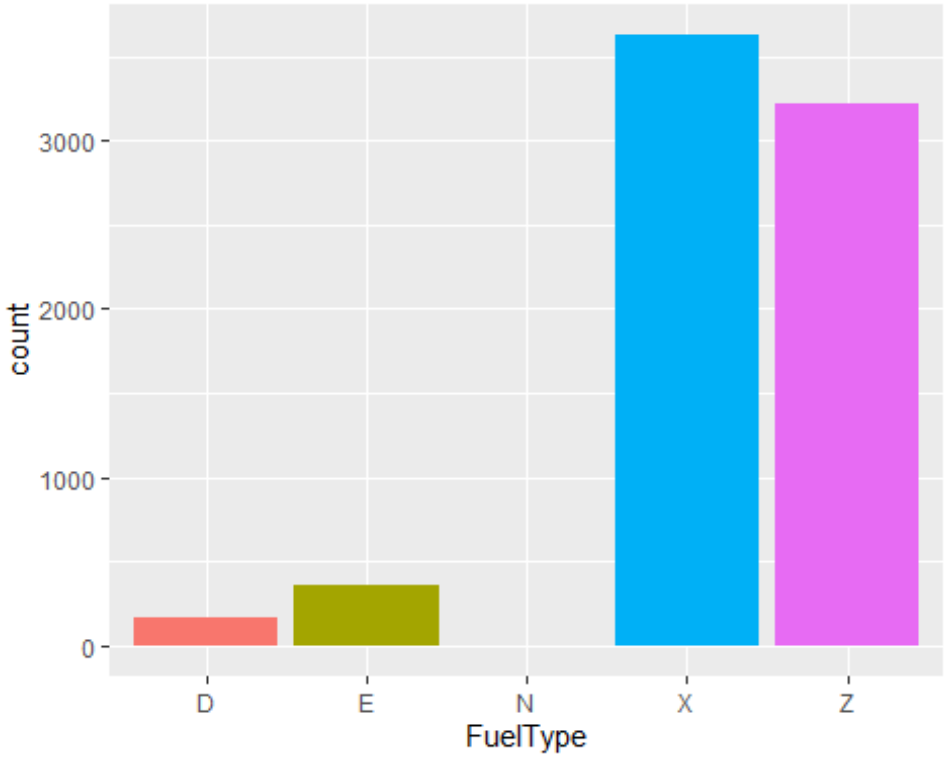
```
ggplot(data=co2) + geom_bar(mapping = aes(x=Make, fill=FuelType)) +
coord_flip()
```



Najzastupljeniji tipovi goriva u globalu su X i Z.

2) Broj automobila sa određenim tipom goriva

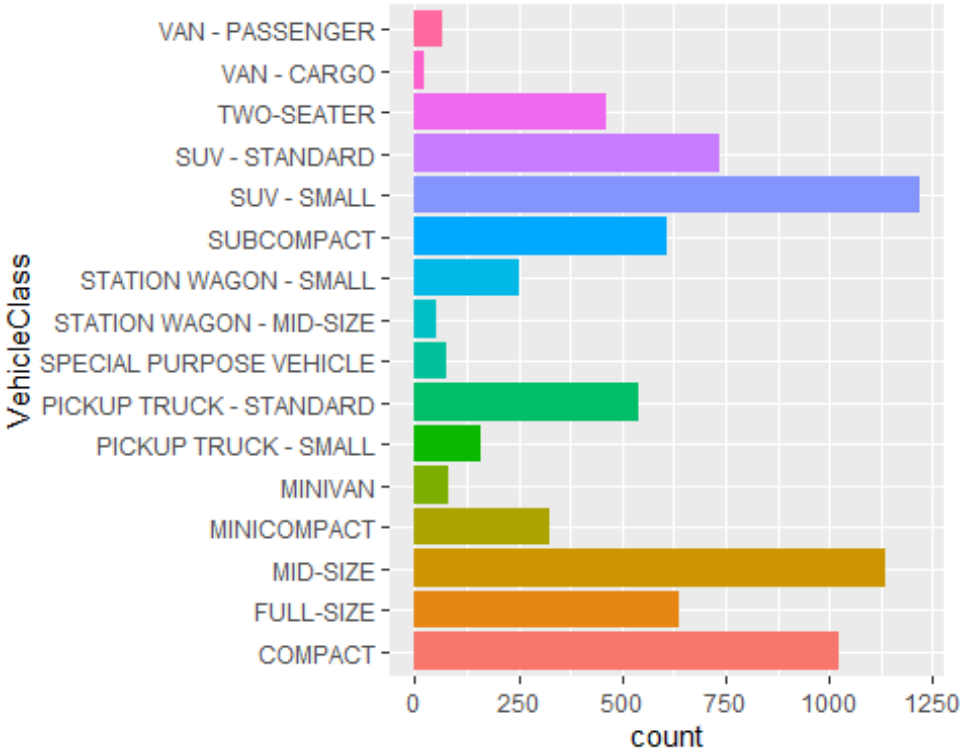
```
ggplot(data=co2) + geom_bar(mapping=aes(x=FuelType,
fill=FuelType),show.legend = FALSE)
```



Na ovom grafiku sada vidimo da su NA vrednosti popunjene.

3) Broj automobila prema klasi vozila

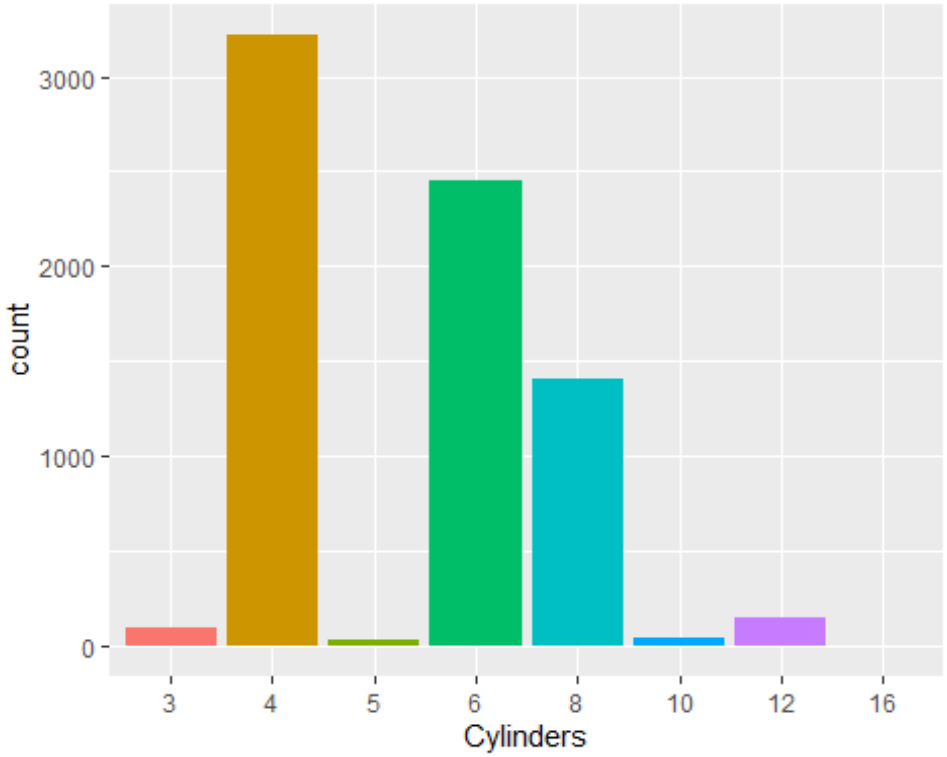
```
ggplot(data=co2) + geom_bar(mapping=aes(x=VehicleClass,
fill=VehicleClass),show.legend = FALSE) + coord_flip()
```



Najviše je zastupljen tip vozila SUV-SMALL, a najmanje VAN-CARGO.

4) Broj automobila sa određenim brojem cilindara

```
ggplot(data=co2) + geom_bar(mapping=aes(x=Cylinders,
fill=Cylinders),show.legend = FALSE)
```



Najviše vozila ima 4 cilindra, nakon čega slede vozila sa 6, a nakon toga 8 cilindara.

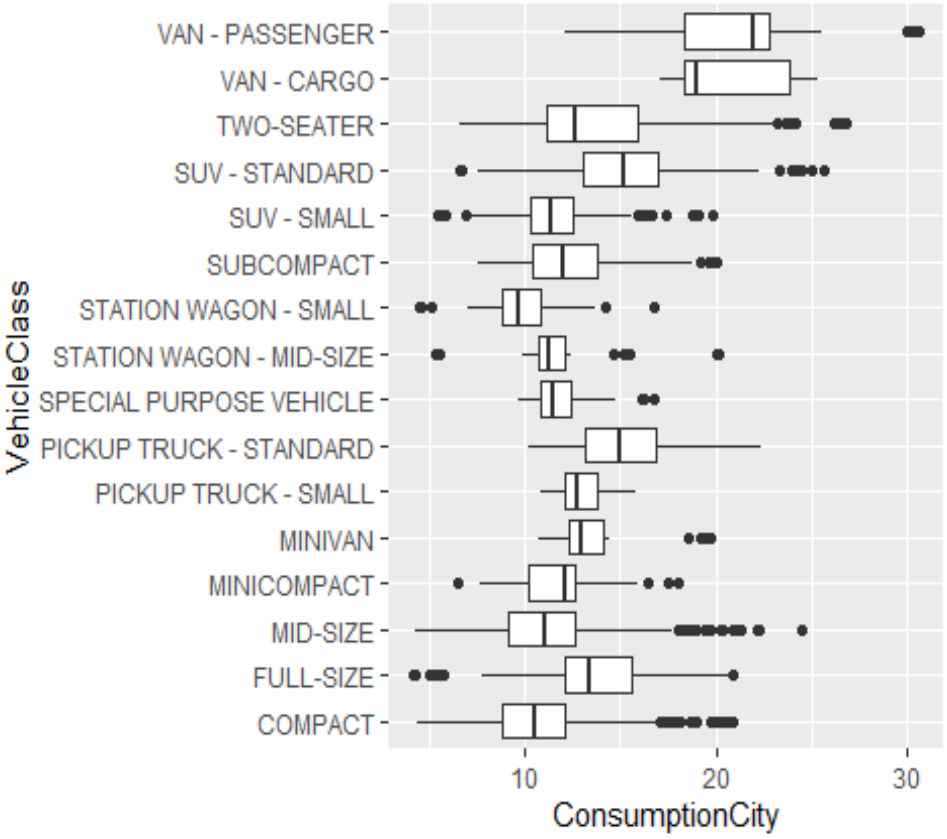
Veoma je mali broj vozila koja imaju broj cilindara 16 (3 vozila). Takođe je veoma mali broj vozila sa 3 i 5 cilindara.

```
Cylinders_ = co2 %>% group_by %>% count(Cylinders)
Cylinders_
```

```
## # A tibble: 8 x 2
##   Cylinders     n
##   <fct>     <int>
## 1 3            95
## 2 4          3220
## 3 5            26
## 4 6          2446
## 5 8          1402
## 6 10           42
## 7 12          151
## 8 16            3
```

5) Potrošnja goriva (L/100km) u gradu prema tipu vozila

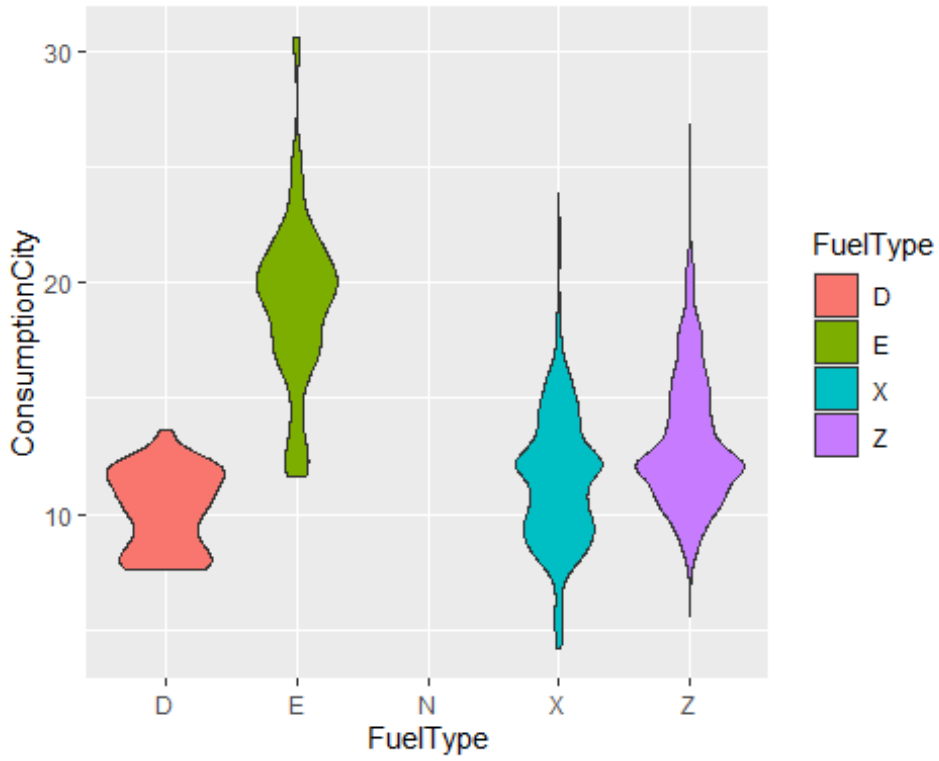
```
ggplot(data=co2,mapping = aes(x=VehicleClass, y=ConsumptionCity)) +
geom_boxplot() + coord_flip()
```



Dosta nesrazmernu potrošnju goriva imaju VAN-PASSENGER i VAN-CARGO (mediana nije na sredini kutije boxplot-a). Outlier-i su prisutni kod dosta tipova vozila, ali najviše preko gornje granice tipa COMPACT i MID-SIZE.

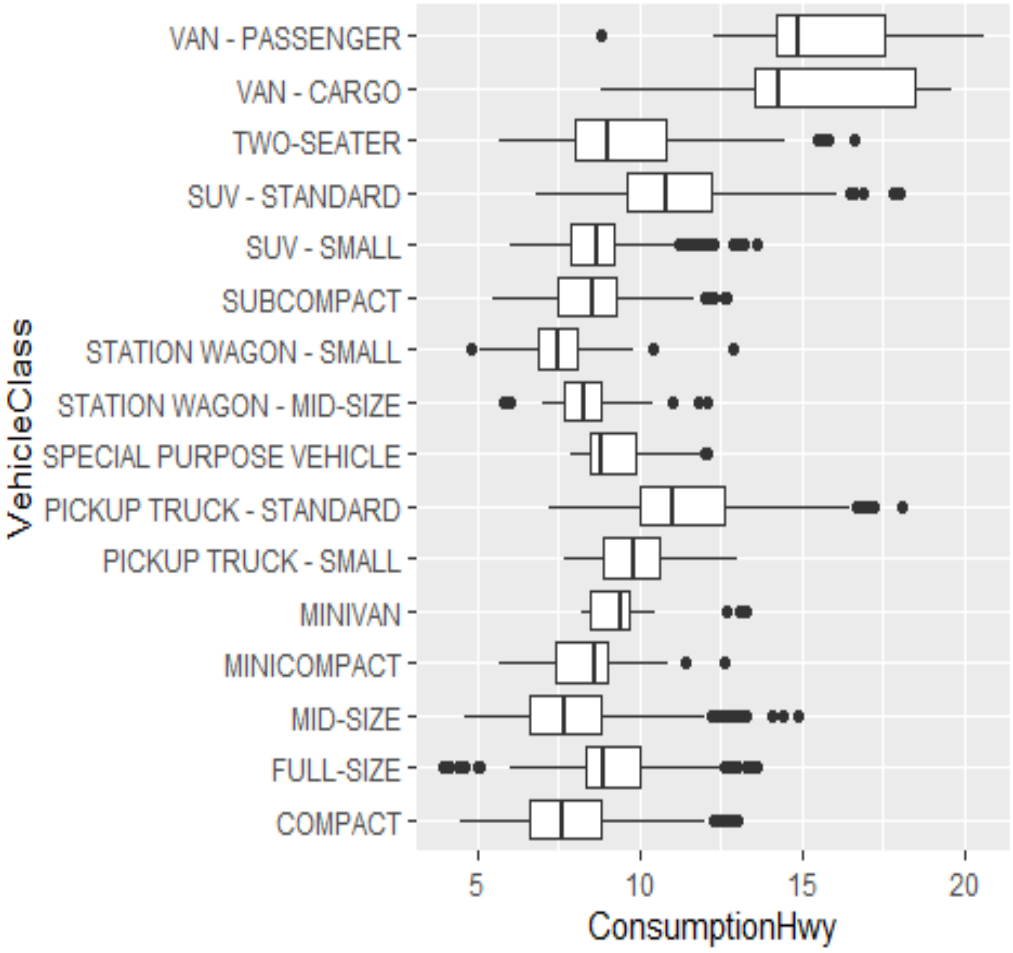
- Potrošnja goriva u gradu po tipu goriva (violinplot – drugačiji način predstavljanja)

```
ggplot(co2, aes(x=FuelType, y=ConsumptionCity, fill=FuelType)) +
  geom_violin()
```



6) Potrošnja goriva (L/100km) na autoputu prema tipu vozila

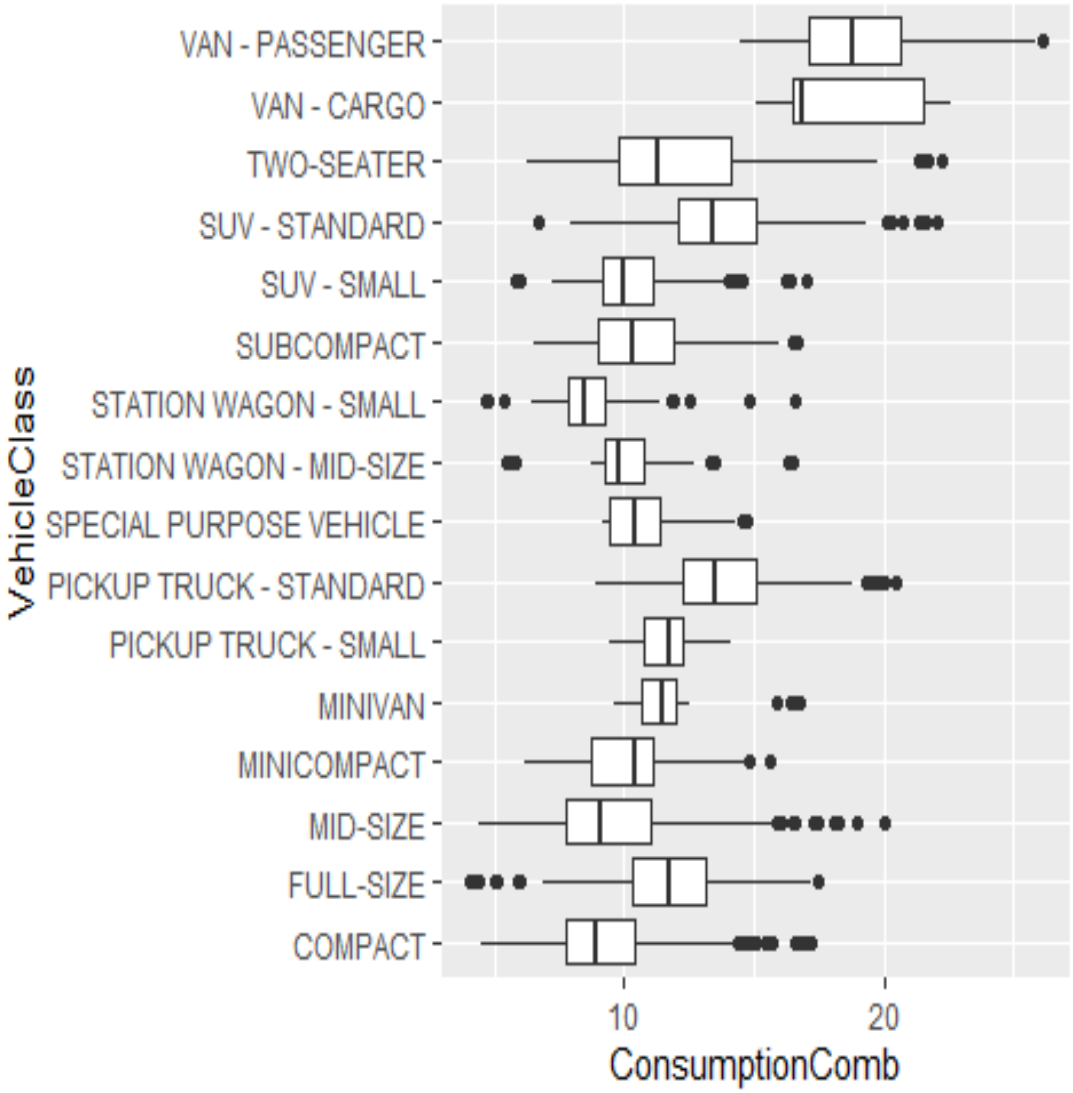
```
ggplot(data=co2,mapping = aes(x=VehicleClass, y=ConsumptionHwy)) +
geom_boxplot() + coord_flip()
```



U ovom slučaju, na autoputu, postoji više vozila sa nesrazmernom potrošnjom goriva nego u gradskoj vožnji. Osim VAN-PASSENGER i VAN-CARGO, možemo primetiti slično i kod SPECIAL PURPOSE VEHICLE, MINIVAN, MINICOMPACT, FULLSIZE (mediana nije na sredini kutije boxplot-a). Outlier-i su prisutni kod dosta tipova vozila, ali najviše preko gornje i donje granice kod tipa vozila FULL-SIZE.

7) Potrošnja goriva (L/100km) u gradu i na autoputu (kombinovano - (55% grad, 45% autoput)) prema tipu vozila

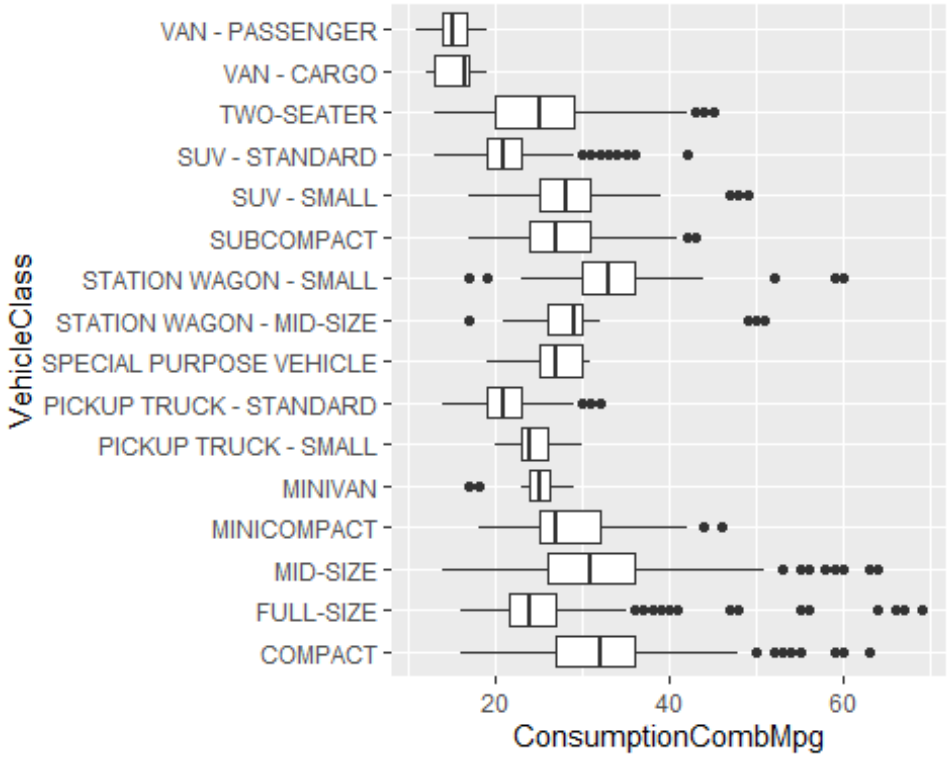
```
ggplot(data=co2,mapping = aes(x=VehicleClass, y=ConsumptionComb)) +
geom_boxplot() + coord_flip()
```



Kada posmatramo kombinovanu potrošnju goriva u odnosu na tip vozila primećujemo da većina vozila ima srazmernu potrošnju goriva, izuzev VAN-CARGO (mediana nije na sredini kutije boxplot-a). Outlier-i su prisutni kod dosta tipova vozila, ali najviše preko gornje granice ih ima COMPACT I MID-SIZE.

8) Potrošnja goriva (mpg) u gradu i na autoputu (kombinovano - (55% grad, 45% autoput)) prema tipu vozila

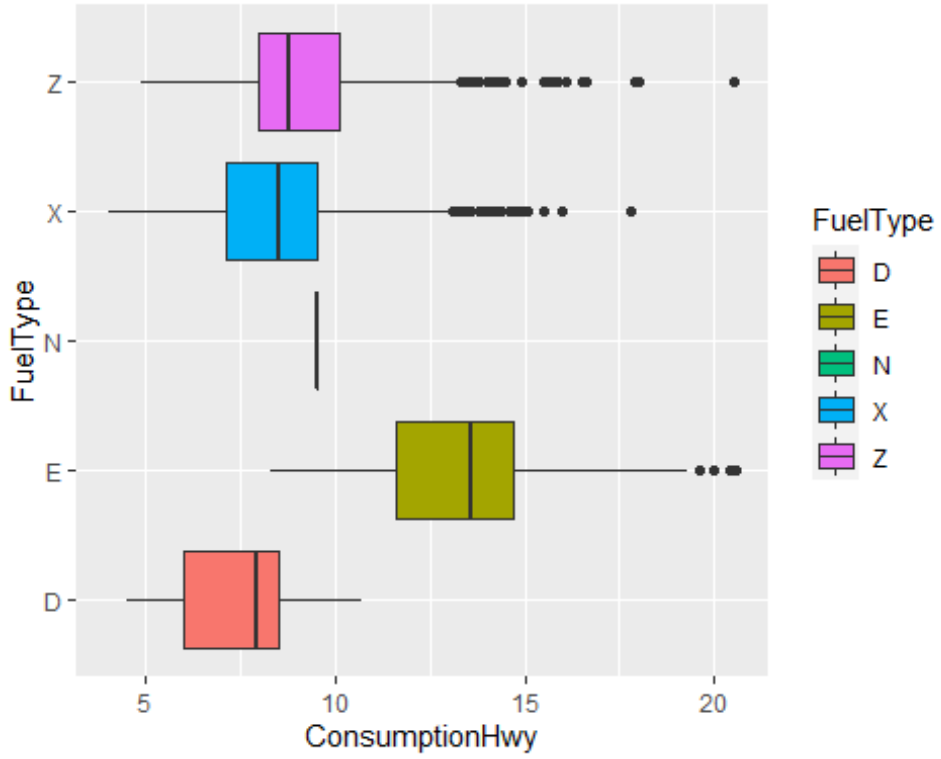
```
ggplot(data=co2,mapping = aes(x=VehicleClass, y=ConsumptionCombMpg)) +
geom_boxplot() + coord_flip()
```



Kod kombinovanu potrošnje goriva u odnosu na tip vozila mpg (milja po galonu), primećujemo da postoji dosta outlier-a kod više tipova vozila: COMPACT, FULL-SIZE, MID-SIZE, SUV-STANDARD. Potrošnja goriva je ujednačena, samo se manje istupanje primećuje kod VAN-CARGO I PICKUP TRUCK-SMALL.

9) Potrošnja goriva na autoputu po tipu goriva

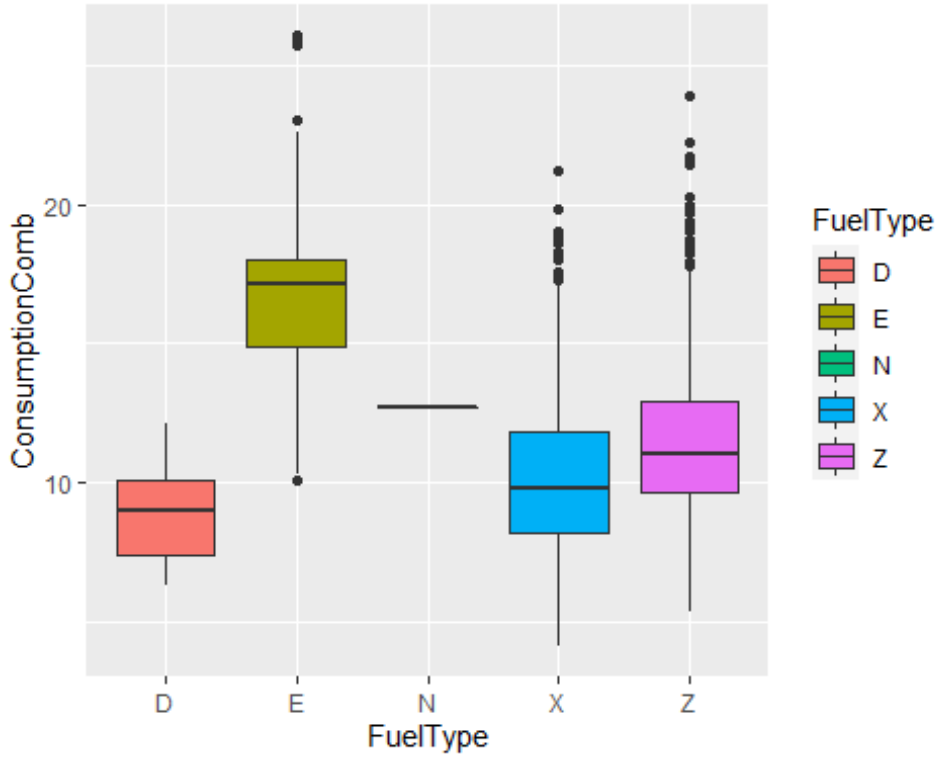
```
ggplot(data=co2,mapping = aes(x=FuelType, y=ConsumptionHwy, fill=FuelType)) +
geom_boxplot() + coord_flip()
```



Gorivo koje se najviše troši, jeste tipa E(etanol-E85), dok se najmanje troši gorivo tipa D(dizel).

10) Potrošnja goriva u gradu i na autoputu (kombinovano) po tipu goriva

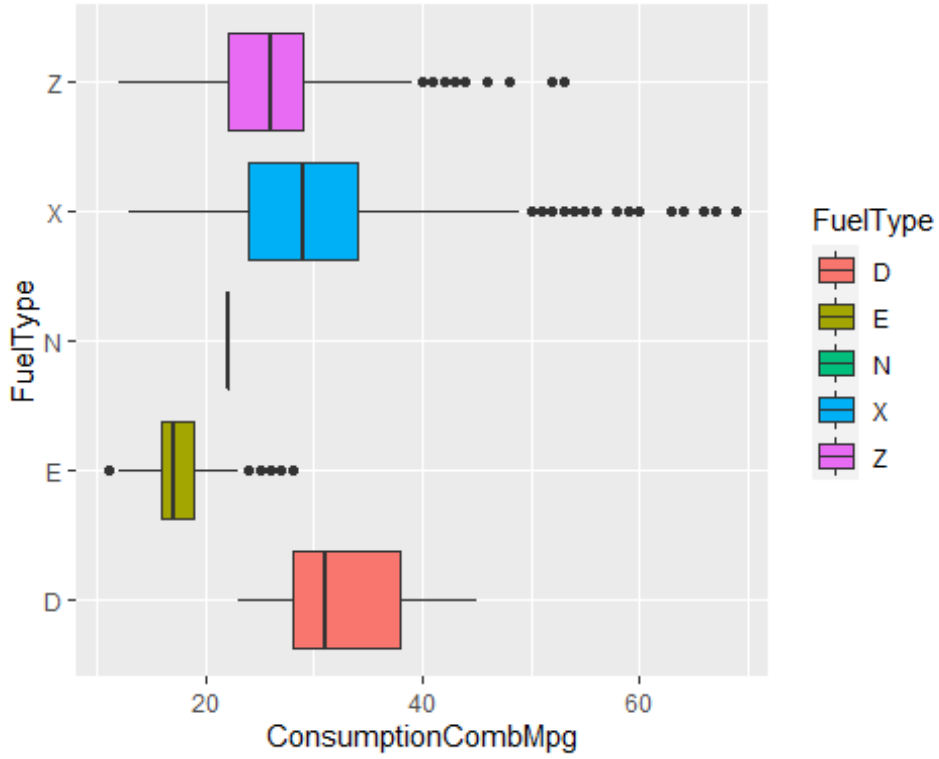
```
ggplot(data=co2,mapping = aes(x=FuelType, y=ConsumptionComb, fill=FuelType))
+ geom_boxplot()
```



Na autoputu i u gradu se najviše u proseku troši gorivo tipa E, a najmanje gorivo tipa D.

11) Potrošnja goriva (mpg) u gradu i na autoputu (kombinovano) po tipu goriva

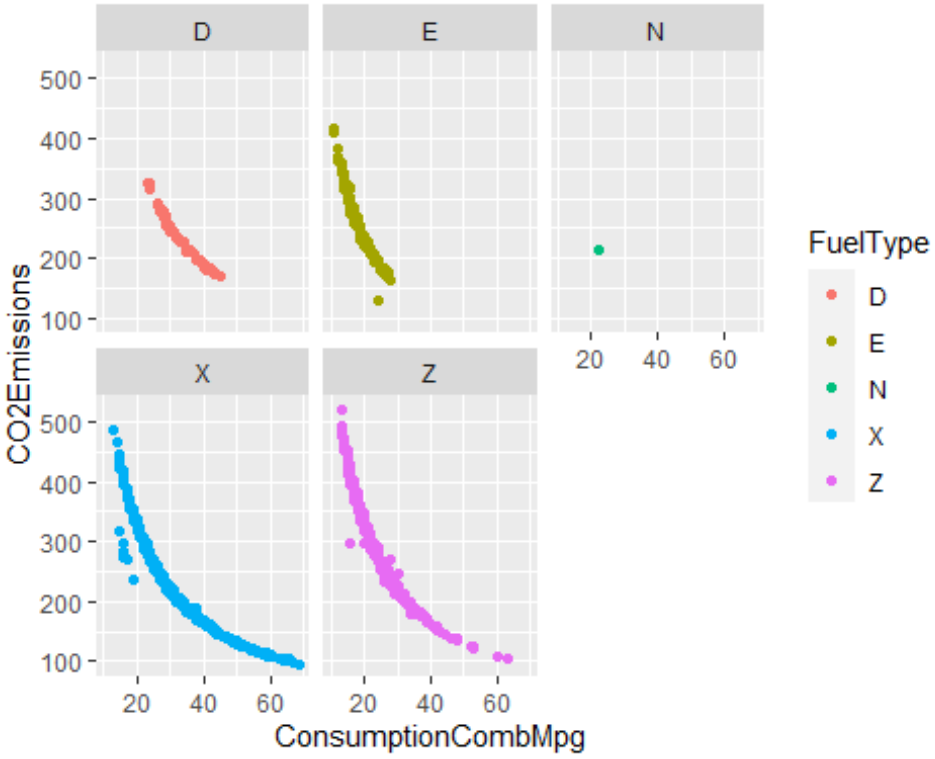
```
ggplot(data=co2,mapping = aes(x=FuelType, y=ConsumptionCombMpg,
fill=FuelType)) + geom_boxplot() + coord_flip()
```



Na autoputu i u gradu (mpg) se najmanje troši gorivo tipa E, a najviše gorivo tipa X.

12) Emisija CO2 u odnosu na potrošnju goriva(mpg) i tip goriva

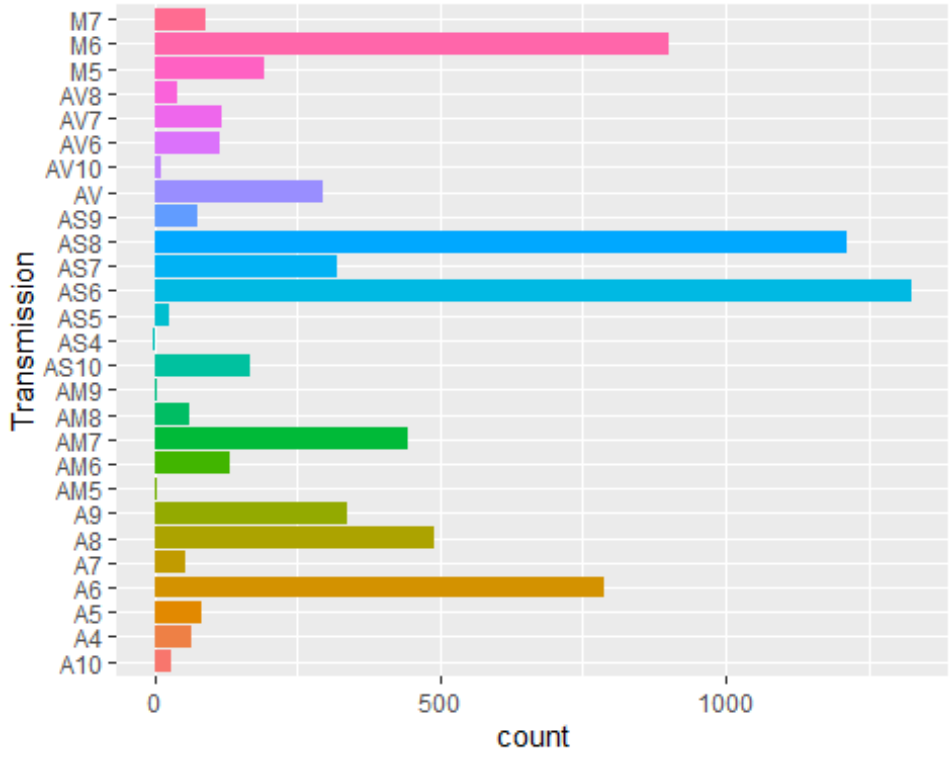
```
ggplot(data=co2, mapping=aes(x=ConsumptionCombMpg,y=CO2Emissions,
color=FuelType)) + geom_point() + facet_wrap(~FuelType, nrow=2)
```



Najveću emisiju CO2, prema tipu goriva I potrošnji goriva u gradu i na autoputu (mpg) ima gorivo tipa Z, a najmanju D, ako izuzmemo gorivo tipa N (tečni naftni gas - TNG).

13) Broj automobila prema vrsti menjača

```
ggplot(data=co2) + geom_bar(mapping=aes(x=Transmission, fill=Transmission),
show.legend = FALSE) + coord_flip() #najvise automobila sa AS6
```



Najviše vozila ima tip menjača AS6, a malo manje vozila ima tip menjača AS8, dok najmanje vozila ima tip menjača AS10, AS4 i AM5.

Broj vozila prema tipu menjača

```
transmission_ = co2 %>% group_by %>% count(Transmission)
transmission_
```

```
## # A tibble: 27 x 2
##    Transmission     n
##    <fct>        <int>
##  1 A10             31
##  2 A4              65
##  3 A5              84
##  4 A6             789
##  5 A7              53
##  6 A8             490
##  7 A9             339
##  8 AM5              4
##  9 AM6            132
## 10 AM7            445
## # ... with 17 more rows
```

Broj vozila prema tipu menjaca - sortirano

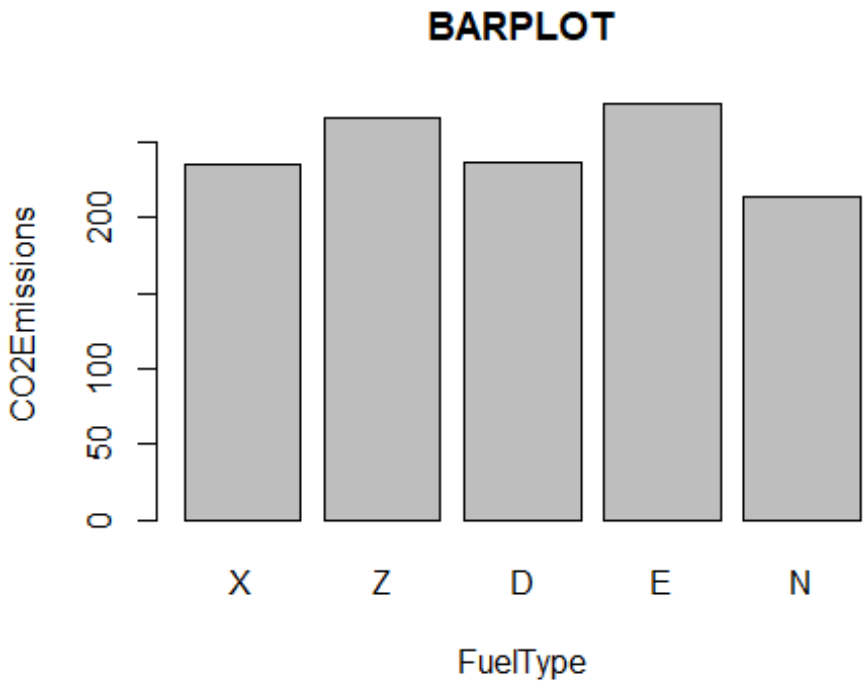
```
transmission_$Transmission <- fct_reorder(transmission_$Transmission,
transmission_$n)
```

```
transmission_ <- transmission_[order(transmission_$n, decreasing = TRUE), ]
transmission_
```

```
## # A tibble: 27 x 2
##    Transmission     n
##    <fct>        <int>
##  1 AS6           1324
##  2 AS8           1211
##  3 M6             901
##  4 A6             789
##  5 A8             490
##  6 AM7            445
##  7 A9             339
##  8 AS7            319
##  9 AV             295
## 10 M5             193
## # ... with 17 more rows
```

14) Prosečna emisija CO2 prema tipu goriva

```
m_data <- setDT(co2)[, mean(CO2Emissions), by=FuelType]
m_data[, barplot(V1, names=FuelType, main="BARPLOT", xlab="FuelType",
ylab="CO2Emissions")]
```

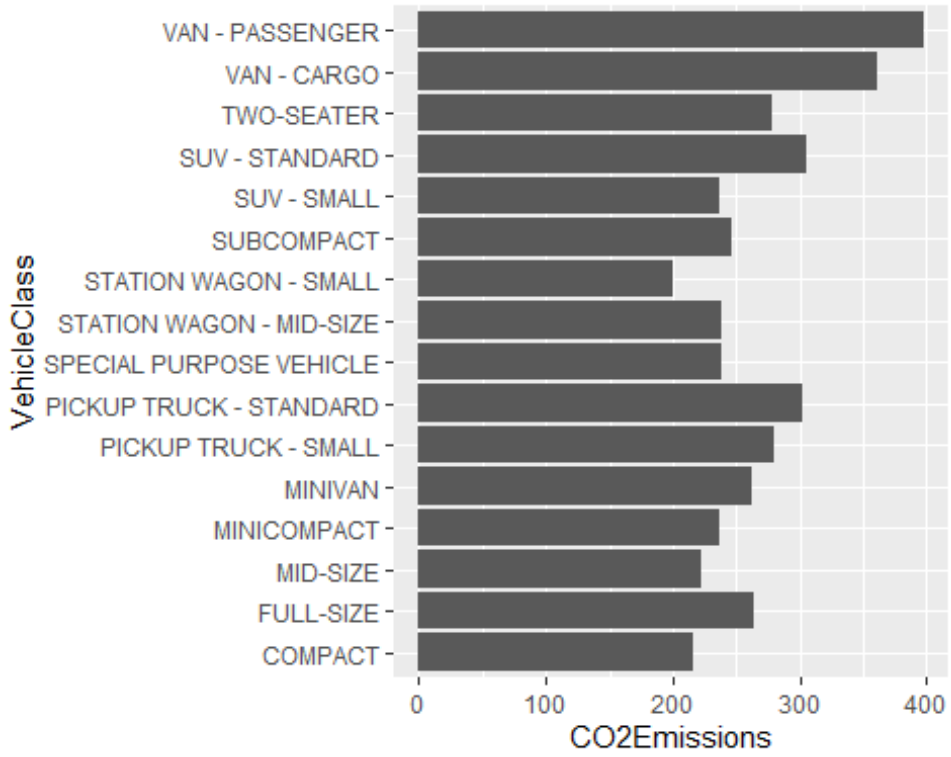


```
##      [,1]
## [1,]  0.7
## [2,]  1.9
## [3,]  3.1
## [4,]  4.3
## [5,]  5.5
```

Tip goriva E emituje najviše CO2, ali ne postoji značajna razlika u emisiji CO2 ni kod drugih tipova goriva.

15) Prosečna emisija CO2 prema tipu vozila

```
ggplot(aes(x = VehicleClass, y = CO2Emissions), data = co2) +
stat_summary(fun = "mean", geom = "bar") + coord_flip()
```



VAN-PASSENGER ima najveću emisiju CO2, dok najmanju ima STATION WAGON-SMALL.

Top 10 kompanija po broju vozila

```
group_Make = co2 %>% group_by(Make) %>% count(Make)
group_Make
```

```
## # A tibble: 42 x 2
## # Groups:   Make [42]
##    Make              n
```

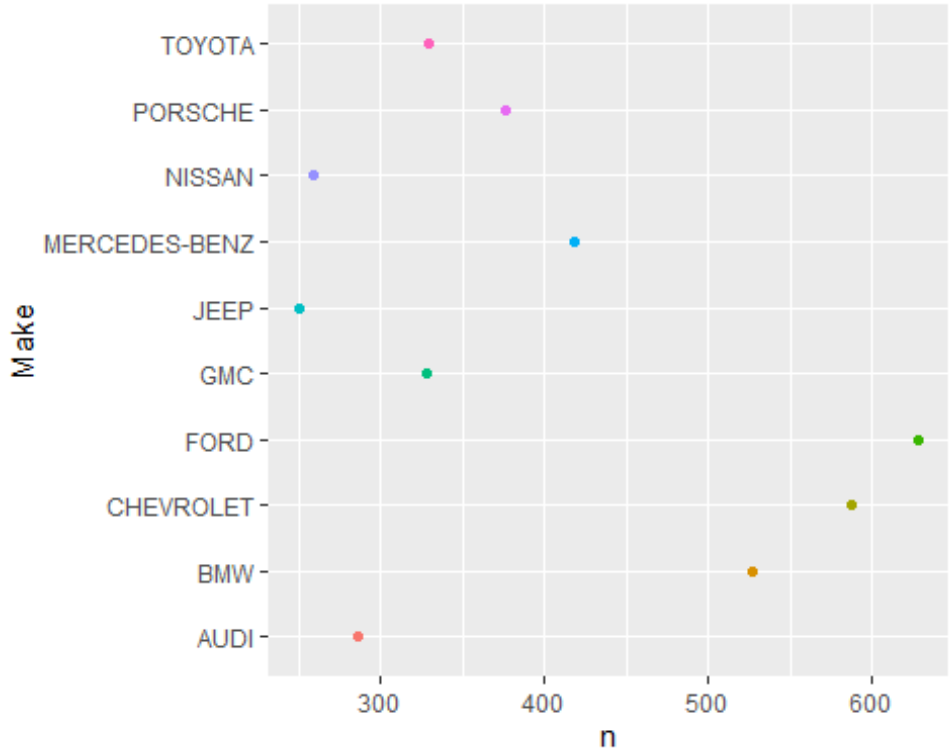
```
##    <fct>         <int>
##  1 ACURA            72
##  2 ALFA ROMEO       30
##  3 ASTON MARTIN     47
##  4 AUDI            286
##  5 BENTLEY          46
##  6 BMW             527
##  7 BUGATTI           3
##  8 BUICK           103
##  9 CADILLAC        158
## 10 CHEVROLET       588
## # ... with 32 more rows
```

```
sortirana <- group_Make[order(group_Make$n, decreasing = TRUE), ]
sortirana1 = sortirana[1:10, ]
sortirana1
```

```
## # A tibble: 10 x 2
## # Groups:   Make [10]
##    Make              n
##    <fct>         <int>
##  1 FORD            628
##  2 CHEVROLET       588
##  3 BMW             527
##  4 MERCEDES-BENZ   419
##  5 PORSCHE         376
##  6 TOYOTA          330
##  7 GMC             328
##  8 AUDI            286
##  9 NISSAN          259
## 10 JEEP            251
```

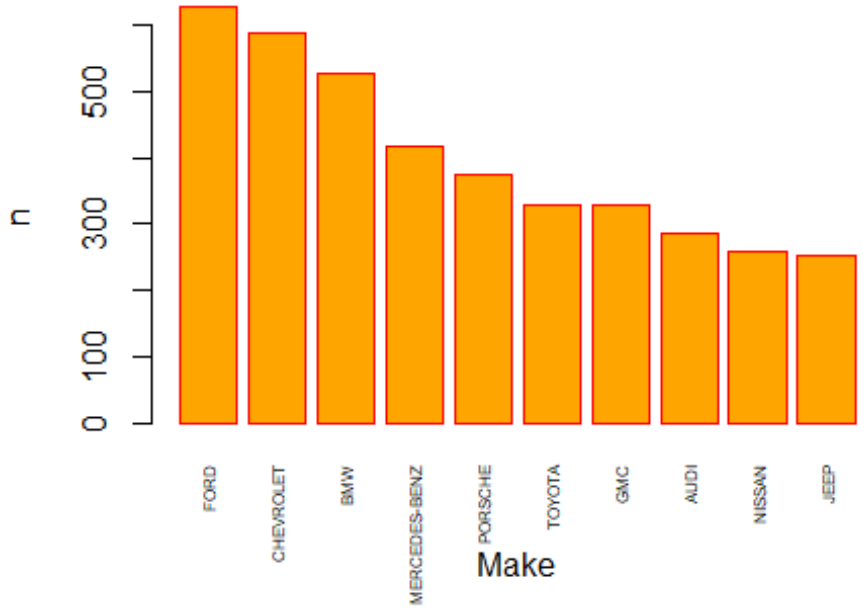
16.1) Broj vozila top 10 kompanija

```
ggplot(data=sortirana1) + geom_point(mapping=aes(x=Make,y=n, color=Make),
show.legend = FALSE) + coord_flip()
```



16.2) Broj vozila top 10 kompanija - barplot

```
barplot(sortirana1$n,names.arg=sortirana1$Make,xlab="Make",ylab="n",col="oran
ge",
        main="",border="red", cex.names = 0.5, las=3)
```

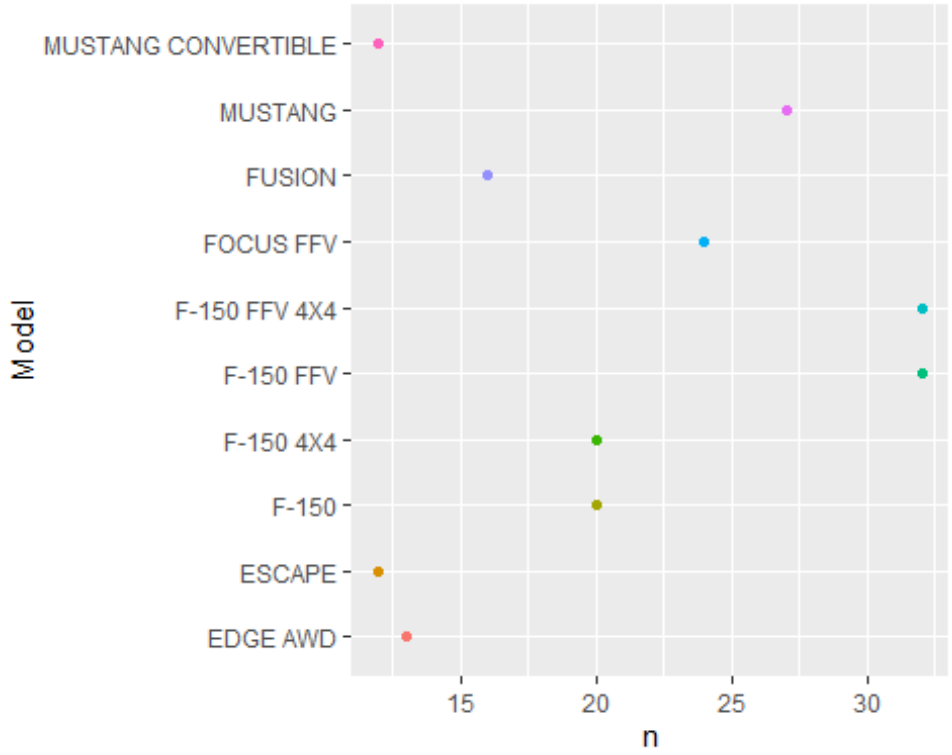


Top 10 Fordovih modela

```
ford = filter(co2, Make=="FORD")
ford_modeli = ford %>% group_by(Model) %>% count(Model)

sortirani_ford_modeli <- ford_modeli[order(ford_modeli$n, decreasing=TRUE),]
sort_ford_mod = sortirani_ford_modeli[1:10,]
```

17.1) Broj vozila top 10 Fordovih modela

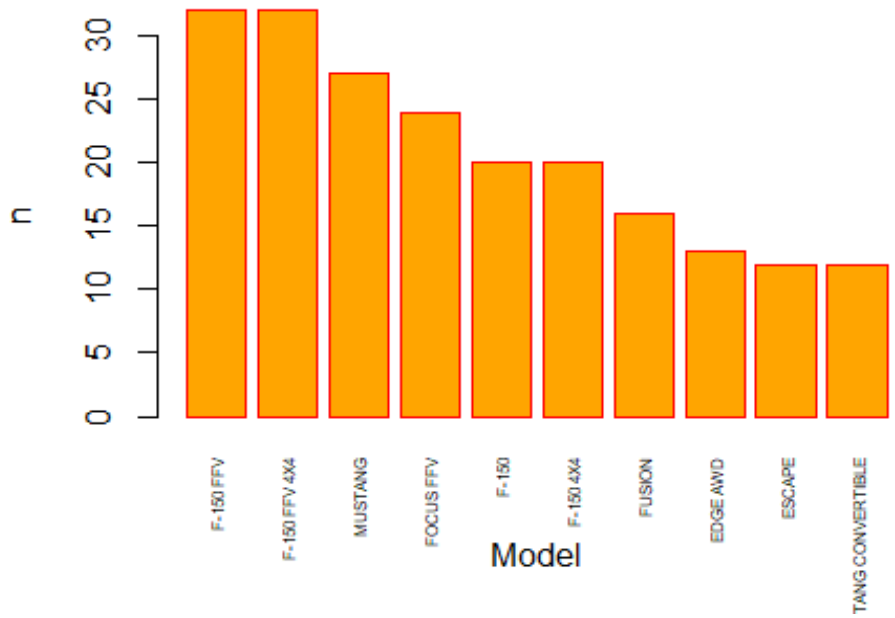
```
ggplot(data=sort_ford_mod) + geom_point(mapping=aes(x=Model,y=n,
color=Model), show.legend = FALSE) + coord_flip()
```



Najzastupljeniji Fordovi modeli su F-150 FFV I F-150 FFV 4X4, dok su najmanje zastupljeni MUSTANG CONVERTIBLE i ESCAPE.

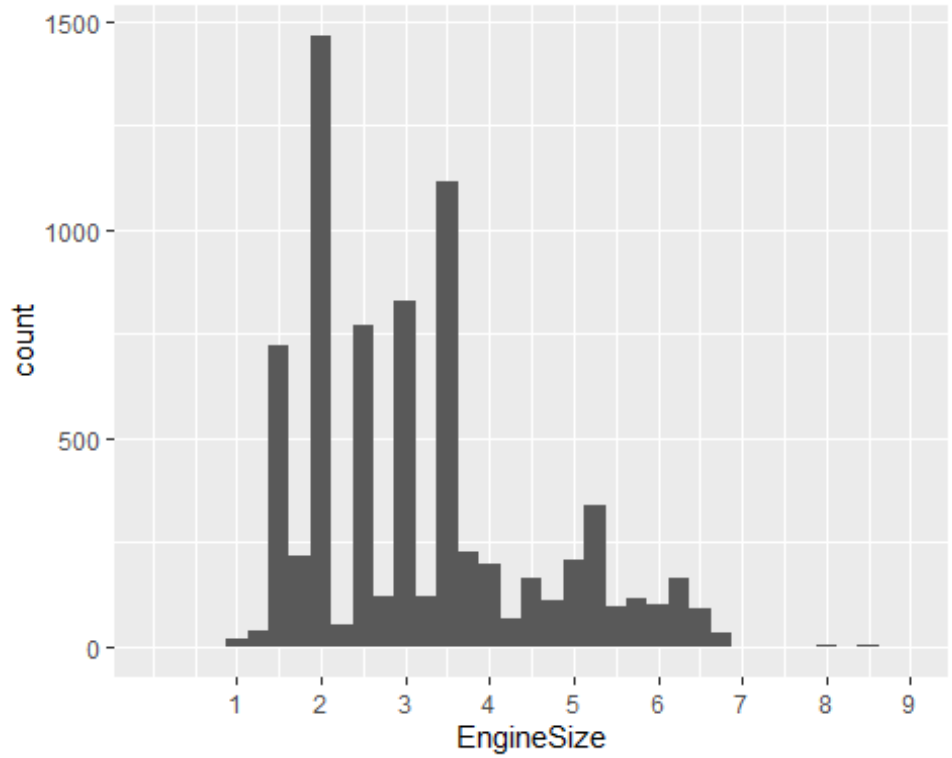
17.2) Broj vozila top 10 Fordovih modela - barplot

```
barplot(sort_ford_mod$n,names.arg=sort_ford_mod$Model,xlab="Model",ylab="n",c
ol="orange",
        main="",border="red", cex.names = 0.5, las=3)
```



18.1) Veličina motora – raspodela (u kom rangu je najzastupljeniji)

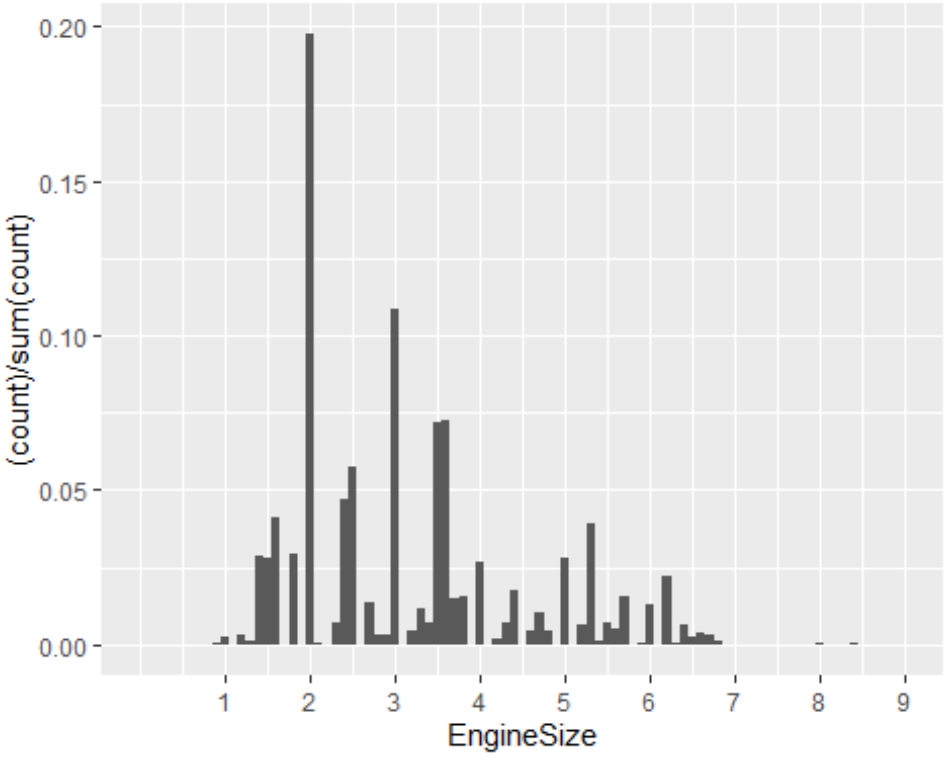
```
ggplot(data=co2) + geom_histogram(mapping = aes(x=EngineSize), binwidth =
0.25) + scale_x_continuous(limits=c(0, 9), breaks = c(1,2,3,4,5,6,7,8,9))
```



Najviše vozila ima veličinu motora u rangu od 1-4.

18.2) Veličina motora - raspodela - procentualno

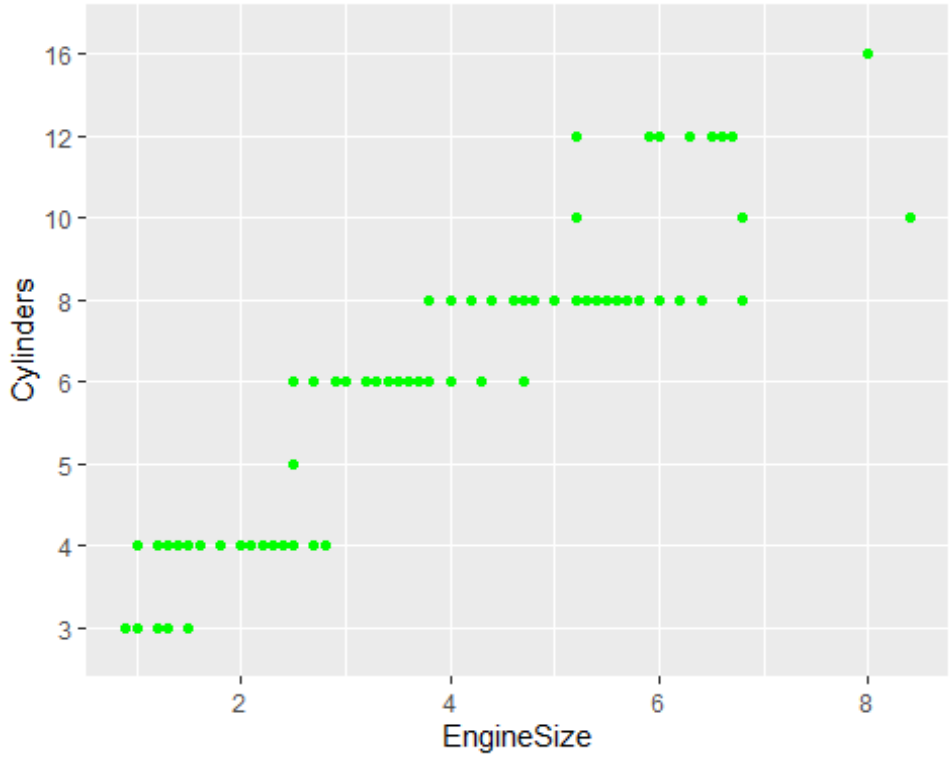
```
ggplot(co2, aes(x = EngineSize)) + geom_histogram(aes(y =
(..count..)/sum(..count..)),binwidth = 0.1) + scale_x_continuous(limits=c(0,
9), breaks = c(1,2,3,4,5,6,7,8,9))
```



Primećujemo da ima dosta vozila sa veličinom motora oko 2.

19) Broj cilindara u odnosu na veličinu motora

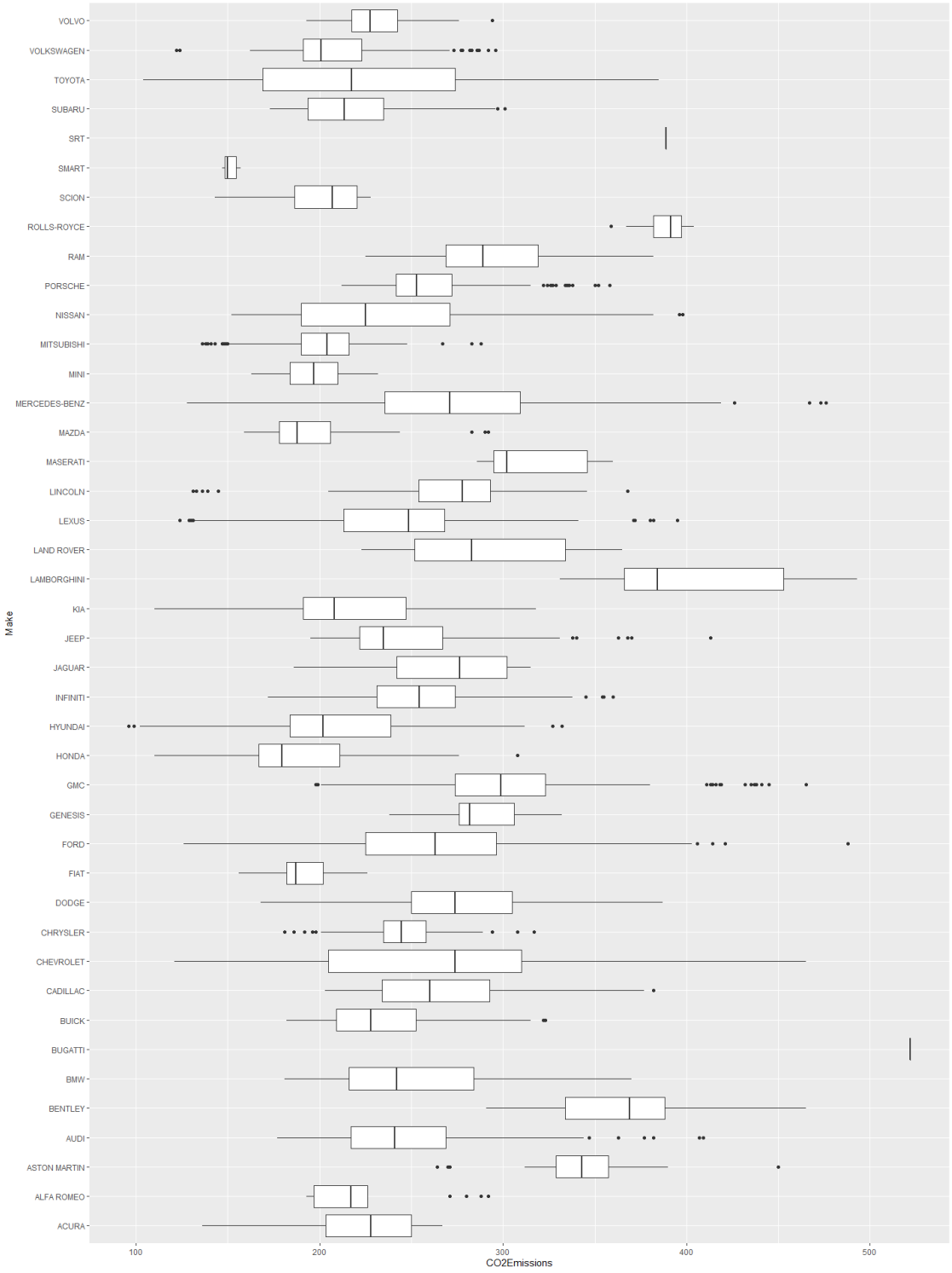
```
ggplot(data=co2) + geom_point(mapping=aes(x=EngineSize,
y=Cylinders),color="green")
```



Kako se povećava broj cilindara, tako se povećava I veličina motora, ali stepeničasto, ne linearno (sličnost sa binomnom raspodelom).

20) Emisija CO2 u odnosu na kompaniju vozila

```
ggplot(data=co2,mapping = aes(x=CO2Emissions, y=Make)) + geom_boxplot()
```




```
modeli_RR = filter(co2, Make=="ROLLS-ROYCE")
modeli_RR$CO2Emissions
```

```
##  [1] 393 390 387 375 368 387 397 401 397 382 404 386 373 370 404 403 397
368 382
## [20] 403 400 400 388 400 397 386 367 393 400 393 397 396 382 359 393 359
```

```
370 396
## [39] 396 396 373 390 390 382 397 388 400 400 382 387
```

```
modeli_SMART = filter(co2,Make=="SMART")
modeli_SMART$CO2Emissions
```

```
## [1] 147 157 150 150 147 157 152
```

Svi modeli ROLLS-ROYCE emituju vise od 350 g/km CO2. Neki modeli (outlier) MERCEDES-BENZ i FORD dostižu najveće emisije CO2. Svi modeli SMART-a emituju male količine CO2.

```
min_co2 = filter(co2, CO2Emissions==min(co2$CO2Emissions))
min_co2_Make = min_co2$Make
min_co2_Make
```

```
## [1] HYUNDAI HYUNDAI HYUNDAI HYUNDAI
## 42 Levels: ACURA ALFA ROMEO ASTON MARTIN AUDI BENTLEY BMW BUGATTI ...
VOLVO
```

Najmanja emisija CO2: HYUINDAI.

```
max_co2 = filter(co2, CO2Emissions==max(co2$CO2Emissions))
max_co2_Make = max_co2$Make
max_co2_Make
```

```
## [1] BUGATTI BUGATTI BUGATTI
## 42 Levels: ACURA ALFA ROMEO ASTON MARTIN AUDI BENTLEY BMW BUGATTI ...
VOLVO
```

Najveća emisija CO2: BUGATTI

```
co2_e = unique(co2)
sortirani_co2_modeli <- co2_e[order(co2_e$CO2Emissions, decreasing=TRUE),]
sort_co2_mod = sortirani_co2_modeli[1:5,]
sort_co2_mod
```

```
##           Make              Model VehicleClass EngineSize Cylinders
## 1:     BUGATTI             Chiron   TWO-SEATER        8.0        16
## 2:     BUGATTI             CHIRON   TWO-SEATER        8.0        16
## 3:     BUGATTI             Chiron   TWO-SEATER        8.0        16
## 4: LAMBORGHINI Aventador Roadster   TWO-SEATER        6.5        12
## 5: LAMBORGHINI Aventador Roadster   TWO-SEATER        6.5        12
```

```
##    Transmission FuelType ConsumptionCity ConsumptionHwy ConsumptionComb
## 1:          AM7        Z            26.8       16.60000             22.2
## 2:          AM7        Z            26.8       16.60000             22.2
## 3:          AM7        Z            26.8        8.78698             22.2
## 4:          AM7        Z            26.6       15.80000             21.7
## 5:          AM7        Z            26.6        8.78698             21.7
##    ConsumptionCombMpg CO2Emissions
## 1:                 13          522
## 2:                 13          522
## 3:                 13          522
## 4:                 13          493
## 5:                 13          493
```

BUGATTI najviše emituje CO2, ali također ima najveći broj cilindara, dok emisija CO2 ne zavisi od tipa goriva (podjednako se pojavljuju X i Z).

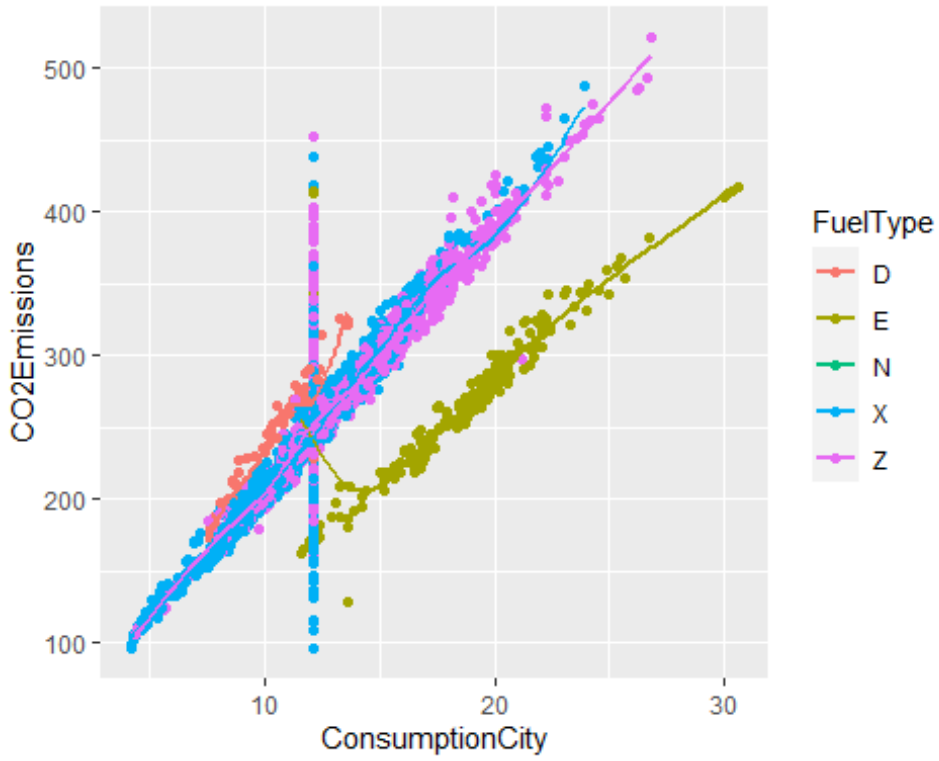
```
co2_a = unique(co2)
sortirani_co2_modeli1 <- co2_a[order(co2_a$CO2Emissions, decreasing=FALSE),]
sort_co2_mod1 = sortirani_co2_modeli1[1:5,]
sort_co2_mod1
```

```
##       Make      Model VehicleClass EngineSize Cylinders Transmission
FuelType
## 1: HYUNDAI IONIQ BLUE    FULL-SIZE        1.6         4          AM6
X
## 2: HYUNDAI IONIQ BLUE    FULL-SIZE        1.6         4          AM6
X
## 3: HYUNDAI IONIQ Blue    FULL-SIZE        1.6         4          AM6
X
## 4: HYUNDAI IONIQ Blue    FULL-SIZE        1.6         4          AM6
X
## 5: HYUNDAI      IONIQ    FULL-SIZE        1.6         4          AM6
X
##    ConsumptionCity ConsumptionHwy ConsumptionComb ConsumptionCombMpg
## 1:         4.20000        8.78698             4.1                 69
## 2:         4.20000        4.00000             4.1                 69
## 3:         4.20000        4.00000             4.1                 69
## 4:        12.08884        8.78698             4.1                 69
## 5:         4.20000        4.20000             4.2                 67
##    CO2Emissions
## 1:           96
## 2:           96
## 3:           96
## 4:           96
## 5:           99
```

Najveći broj vozila koja najmanje emituju CO2 imaju tip goriva X (običan benzin).

21) Prikaz emisije CO2 u odnosu na potrošnju goriva u gradu u zavisnosti od tipa goriva:

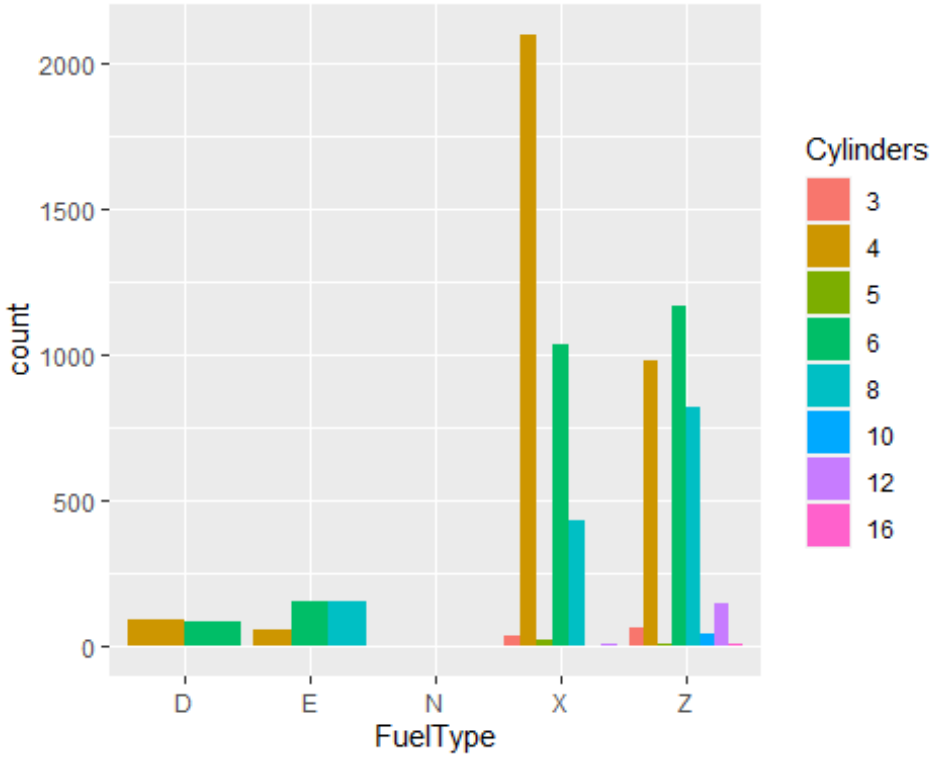
```
ggplot(data = co2, mapping = aes(x = ConsumptionCity , y = CO2Emissions,
color = FuelType)) + geom_point() +
geom_smooth(se = FALSE)

## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Primećujemo linearnu zavisnost emisije CO2 i potrošnje goriva po gradu.

22) Prikaz broja vozila sa određenim brojem cilindara i tipom goriva

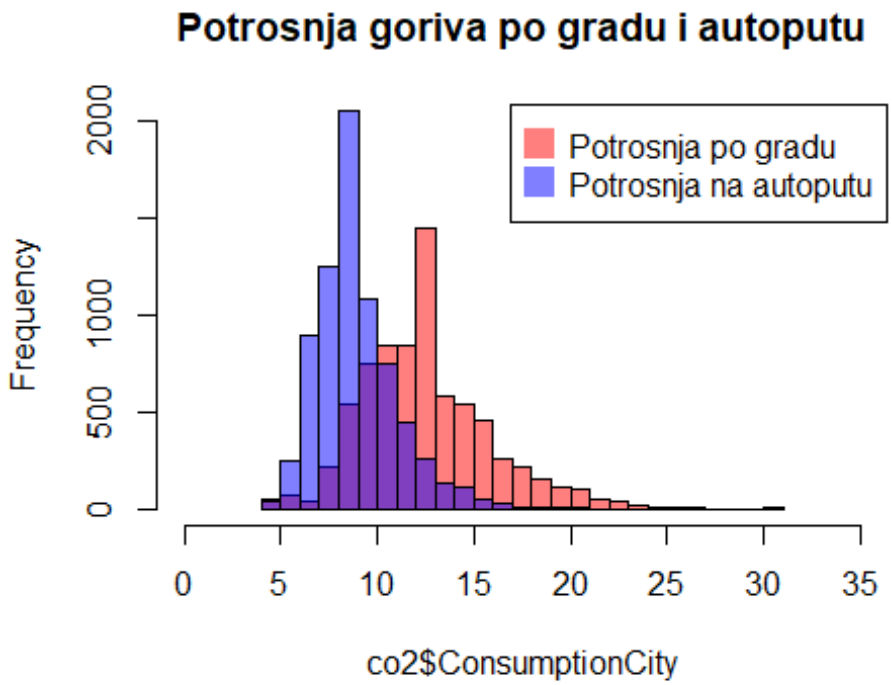
```
ggplot(data = co2) + geom_bar(mapping = aes(x = FuelType, fill = Cylinders),
position = "dodge")
```



Najzastupljenija su vozila sa manjim brojem cilindara (4, 6 i 8).

23) Potrošnja goriva po gradu i po autoputu (histogrami)

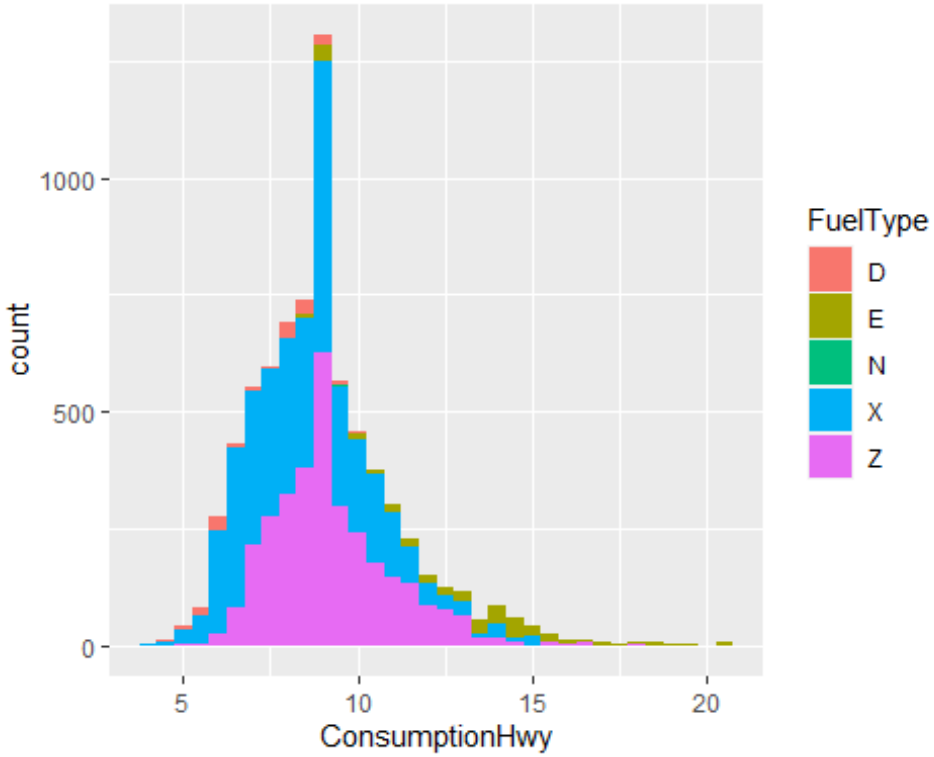
```
hist(co2$ConsumptionCity, breaks=30, xlim=c(0,35),
ylim=c(0,2000),col=rgb(1,0,0,0.5),main="Potrosnja goriva po gradu i autoputu"
)

hist(co2$ConsumptionHwy, breaks=20, xlim=c(0,35), ylim=c(0,2000),
col=rgb(0,0,1,0.5), add=T)

# Add legend
legend("topright", legend=c("Potrosnja po gradu","Potrosnja na autoputu"),
col=c(rgb(1,0,0,0.5),
     rgb(0,0,1,0.5)), pt.cex=2, pch=15 )
```



Potrošnja goriva je veća na autoputu.

24) Najdominantnije vrste goriva koje se koriste na autoputu

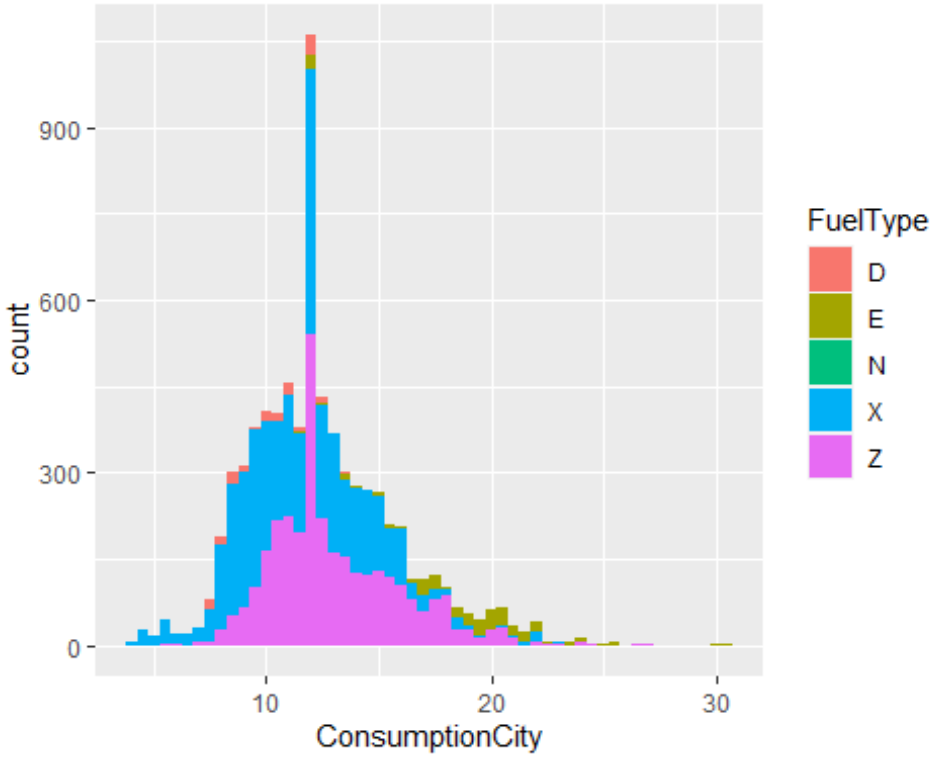
```
library("histogram")

ggplot(data = co2) + geom_histogram(mapping = aes(x = ConsumptionHwy,
fill=FuelType), binwidth = 0.5)
```



Goriva tipa X i Z su najdominantnije vrste goriva koje se koriste na autoputu.

25) Najdominantnije vrste goriva koje se koriste u gradu

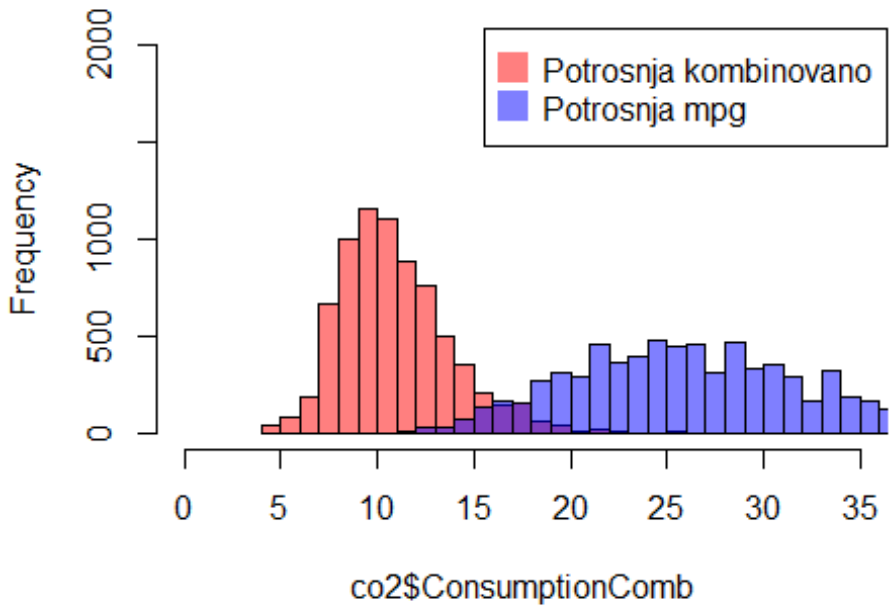
```
ggplot(data = co2) + geom_histogram(mapping = aes(x = ConsumptionCity,
fill=FuelType), binwidth = 0.5)
```



U gradu se najviše korisite X i Z vrste goriva.

26) Potrošnja goriva kombinovano i kombinovano mpg

```
hist(co2$ConsumptionComb, breaks=30, xlim=c(0,35),
ylim=c(0,2000),col=rgb(1,0,0,0.5),main="" )
hist(co2$ConsumptionCombMpg, breaks=70, xlim=c(0,70), ylim=c(0,2000),
col=rgb(0,0,1,0.5), add=T)

# Add Legend
legend("topright", legend=c("Potrosnja kombinovano","Potrosnja mpg"),
col=c(rgb(1,0,0,0.5),
     rgb(0,0,1,0.5)), pt.cex=2, pch=15 )
```



Modelovanje

Podela skupa na trening testni

```
sample_size = floor(0.6 * nrow(co2))
set.seed(42)
train_ind = sample(seq(1, nrow(co2)), size = sample_size)
co2.train = co2[train_ind,]
co2.test = co2[-train_ind,]
dim(co2.train)

## [1] 4431   12
```

1) Linearna regresija

```
model1 = lm(CO2Emissions~., data = co2.train[,-2])
summary(model1)

##
## Call:
```

```
## lm(formula = CO2Emissions ~ ., data = co2.train[, -2])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -110.881   -2.342    0.002    2.013   34.284
##
## Coefficients: (1 not defined because of singularities)
##                                    Estimate Std. Error  t value
Pr(>|t|)
## (Intercept)                       109.10308    3.69131   29.557  < 2e-
16
## MakeALFA ROMEO                      3.80795    1.71071    2.226
0.026069
## MakeASTON MARTIN                    5.47531    1.74073    3.145
0.001670
## MakeAUDI                            2.28799    1.11517    2.052
0.040260
## MakeBENTLEY                         8.32644    1.58478    5.254 1.56e-
07
## MakeBMW                             1.98638    1.06242    1.870
0.061596
## MakeBUGATTI                        38.45164    4.92700    7.804 7.45e-
15
## MakeBUICK                           3.71312    1.23686    3.002
0.002697
## MakeCADILLAC                        3.43222    1.16246    2.953
0.003169
## MakeCHEVROLET                       3.49273    1.08610    3.216
0.001310
## MakeCHRYSLER                        6.58945    1.39062    4.739 2.22e-
06
## MakeDODGE                           5.95454    1.18173    5.039 4.87e-
07
## MakeFIAT                            5.51329    1.52557    3.614
0.000305
## MakeFORD                            5.21299    1.09750    4.750 2.10e-
06
## MakeGENESIS                         8.02283    1.84740    4.343 1.44e-
05
## MakeGMC                             3.94040    1.17365    3.357
0.000794
## MakeHONDA                           3.21509    1.17151    2.744
0.006087
## MakeHYUNDAI                         4.47354    1.17430    3.810
0.000141
## MakeINFINITI                        2.12944    1.38709    1.535
0.124812
## MakeJAGUAR                          1.63344    1.21437    1.345
0.178667
## MakeJEEP                            5.12916    1.18810    4.317 1.62e-
```

```
05
## MakeKIA                              4.99780    1.15308    4.334 1.50e-
05
## MakeLAMBORGHINI                      9.42519    1.93718    4.865 1.18e-
06
## MakeLAND ROVER                       2.41975    1.30205    1.858
0.063179
## MakeLEXUS                            2.70791    1.16243    2.330
0.019877
## MakeLINCOLN                          6.29651    1.39329    4.519 6.37e-
06
## MakeMASERATI                         9.06381    1.41043    6.426 1.45e-
10
## MakeMAZDA                            2.46722    1.18559    2.081
0.037492
## MakeMERCEDES-BENZ                    2.42578    1.17880    2.058
0.039665
## MakeMINI                             1.44307    1.17587    1.227
0.219799
## MakeMITSUBISHI                       3.12067    1.40275    2.225
0.026154
## MakeNISSAN                           3.16368    1.15147    2.748
0.006030
## MakePORSCHE                          2.17192    1.11569    1.947
0.051635
## MakeRAM                              3.85761    1.41988    2.717
0.006617
## MakeROLLS-ROYCE                      6.52860    1.75712    3.716
0.000205
## MakeSCION                            1.53112    1.83744    0.833
0.404727
## MakeSMART                           -3.60075    6.22500   -0.578
0.563001
## MakeSRT                              4.88215    6.34117    0.770
0.441394
## MakeSUBARU                           2.99930    1.22491    2.449
0.014381
## MakeTOYOTA                           4.76316    1.11613    4.268 2.02e-
05
## MakeVOLKSWAGEN                       2.13482    1.16153    1.838
0.066140
## MakeVOLVO                            1.98352    1.25242    1.584
0.113323
## VehicleClassFULL-SIZE               -0.18569    0.44952   -0.413
0.679555
## VehicleClassMID-SIZE                -0.42959    0.36313   -1.183
0.236874
## VehicleClassMINICOMPACT             -1.45697    0.68227   -2.135
0.032777
## VehicleClassMINIVAN                  0.73895    0.97734    0.756
```

```
0.449643
## VehicleClassPICKUP TRUCK - SMALL        4.78294    0.79173    6.041 1.66e-
09
## VehicleClassPICKUP TRUCK - STANDARD     3.57389    0.61995    5.765 8.74e-
09
## VehicleClassSPECIAL PURPOSE VEHICLE     2.66628    1.00400    2.656
0.007944
## VehicleClassSTATION WAGON - MID-SIZE   -0.19823    1.10920   -0.179
0.858168
## VehicleClassSTATION WAGON - SMALL       0.37546    0.56096    0.669
0.503329
## VehicleClassSUBCOMPACT                 -0.35915    0.43736   -0.821
0.411586
## VehicleClassSUV - SMALL                 0.44899    0.40140    1.119
0.263391
## VehicleClassSUV - STANDARD              2.72317    0.49439    5.508 3.84e-
08
## VehicleClassTWO-SEATER                  0.29512    0.53410    0.553
0.580598
## VehicleClassVAN - CARGO                 0.34190    2.21468    0.154
0.877318
## VehicleClassVAN - PASSENGER             5.64382    1.30793    4.315 1.63e-
05
## EngineSize                              1.29642    0.29256    4.431 9.60e-
06
## Cylinders4                             -1.87837    0.94257   -1.993
0.046344
## Cylinders5                             -2.28561    1.84851   -1.236
0.216354
## Cylinders6                             -1.17799    1.09033   -1.080
0.280025
## Cylinders8                              0.45073    1.38294    0.326
0.744497
## Cylinders10                             3.19680    2.16966    1.473
0.140713
## Cylinders12                             7.98179    1.89587    4.210 2.60e-
05
## Cylinders16                                  NA         NA       NA
NA
## TransmissionA4                         -6.56737    1.94947   -3.369
0.000762
## TransmissionA5                          0.33213    1.82890    0.182
0.855904
## TransmissionA6                         -2.90112    1.53846   -1.886
0.059398
## TransmissionA7                          3.21480    1.92935    1.666
0.095734
## TransmissionA8                         -1.37395    1.57272   -0.874
0.382378
## TransmissionA9                          0.12863    1.63946    0.078
```

```
0.937467
## TransmissionAM5                 3.72774    7.55882    0.493
0.621921
## TransmissionAM6                 1.11732    1.72093    0.649
0.516210
## TransmissionAM7                 0.92257    1.63597    0.564
0.572833
## TransmissionAM8                 0.04892    1.92589    0.025
0.979735
## TransmissionAM9                 6.12001    6.30594    0.971
0.331844
## TransmissionAS10               -0.59664    1.66051   -0.359
0.719379
## TransmissionAS4                -2.11046    6.41457   -0.329
0.742163
## TransmissionAS5                -2.00175    2.16131   -0.926
0.354406
## TransmissionAS6                -0.35367    1.57069   -0.225
0.821857
## TransmissionAS7                -0.57286    1.69270   -0.338
0.735057
## TransmissionAS8                -0.49201    1.57197   -0.313
0.754302
## TransmissionAS9                 0.24332    1.79324    0.136
0.892077
## TransmissionAV                  1.03576    1.67216    0.619
0.535676
## TransmissionAV10               -1.54095    2.93967   -0.524
0.600174
## TransmissionAV6                -2.21713    1.80071   -1.231
0.218294
## TransmissionAV7                -0.45741    1.73497   -0.264
0.792069
## TransmissionAV8                -1.30858    2.03272   -0.644
0.519767
## TransmissionM5                 -1.17291    1.67503   -0.700
0.483819
## TransmissionM6                 -0.40924    1.57671   -0.260
0.795223
## TransmissionM7                  0.72438    1.86436    0.389
0.697637
## FuelTypeE                    -132.00949    0.97038 -136.039  < 2e-
16
## FuelTypeN                    -108.88985    6.07343  -17.929  < 2e-
16
## FuelTypeX                     -30.18887    0.64594  -46.736  < 2e-
16
## FuelTypeZ                     -28.57475    0.67782  -42.157  < 2e-
16
## ConsumptionCity                 0.01773    0.09390    0.189
```

```
0.850247
## ConsumptionHwy                 0.04414    0.12437    0.355
0.722692
## ConsumptionComb               18.28645    0.21839   83.732  < 2e-
16
## ConsumptionCombMpg            -1.19721    0.05101  -23.468  < 2e-
16
##
## (Intercept)                  ***
## MakeALFA ROMEO               *
## MakeASTON MARTIN             **
## MakeAUDI                     *
## MakeBENTLEY                  ***
## MakeBMW                      .
## MakeBUGATTI                  ***
## MakeBUICK                    **
## MakeCADILLAC                 **
## MakeCHEVROLET                **
## MakeCHRYSLER                 ***
## MakeDODGE                    ***
## MakeFIAT                     ***
## MakeFORD                     ***
## MakeGENESIS                  ***
## MakeGMC                      ***
## MakeHONDA                    **
## MakeHYUNDAI                  ***
## MakeINFINITI
## MakeJAGUAR
## MakeJEEP                     ***
## MakeKIA                      ***
## MakeLAMBORGHINI              ***
## MakeLAND ROVER               .
## MakeLEXUS                    *
## MakeLINCOLN                  ***
## MakeMASERATI                 ***
## MakeMAZDA                    *
## MakeMERCEDES-BENZ            *
## MakeMINI
## MakeMITSUBISHI               *
## MakeNISSAN                   **
## MakePORSCHE                  .
## MakeRAM                      **
## MakeROLLS-ROYCE              ***
## MakeSCION
## MakeSMART
## MakeSRT
## MakeSUBARU                   *
## MakeTOYOTA                   ***
## MakeVOLKSWAGEN               .
## MakeVOLVO
```

```
## VehicleClassFULL-SIZE
## VehicleClassMID-SIZE
## VehicleClassMINICOMPACT              *
## VehicleClassMINIVAN
## VehicleClassPICKUP TRUCK - SMALL     ***
## VehicleClassPICKUP TRUCK - STANDARD  ***
## VehicleClassSPECIAL PURPOSE VEHICLE  **
## VehicleClassSTATION WAGON - MID-SIZE
## VehicleClassSTATION WAGON - SMALL
## VehicleClassSUBCOMPACT
## VehicleClassSUV - SMALL
## VehicleClassSUV - STANDARD           ***
## VehicleClassTWO-SEATER
## VehicleClassVAN - CARGO
## VehicleClassVAN - PASSENGER          ***
## EngineSize                           ***
## Cylinders4                           *
## Cylinders5
## Cylinders6
## Cylinders8
## Cylinders10
## Cylinders12                          ***
## Cylinders16
## TransmissionA4                       ***
## TransmissionA5
## TransmissionA6                       .
## TransmissionA7                       .
## TransmissionA8
## TransmissionA9
## TransmissionAM5
## TransmissionAM6
## TransmissionAM7
## TransmissionAM8
## TransmissionAM9
## TransmissionAS10
## TransmissionAS4
## TransmissionAS5
## TransmissionAS6
## TransmissionAS7
## TransmissionAS8
## TransmissionAS9
## TransmissionAV
## TransmissionAV10
## TransmissionAV6
## TransmissionAV7
## TransmissionAV8
## TransmissionM5
## TransmissionM6
## TransmissionM7
## FuelTypeE                            ***
```

```
## FuelTypeN                              ***
## FuelTypeX                              ***
## FuelTypeZ                              ***
## ConsumptionCity
## ConsumptionHwy
## ConsumptionComb                        ***
## ConsumptionCombMpg                     ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.008 on 4333 degrees of freedom
## Multiple R-squared:  0.9898, Adjusted R-squared:  0.9896
## F-statistic:  4350 on 97 and 4333 DF,  p-value: < 2.2e-16
```

```
R21 = summary(model1)$r.square
RSS1 = deviance(model1) #MSE
koeficijenti1 = coefficients(model1)
fstatistics1 = summary(model1)$fstatistic[1]
R21
```

```
## [1] 0.9898352
```

```
RSS1
```

```
## [1] 156415.5
```

```
koeficijenti1
```

```
##                          (Intercept)                        MakeALFA ROMEO
##                          109.10307968                           3.80794960
##                       MakeASTON MARTIN                             MakeAUDI
##                             5.47530863                           2.28798659
##                            MakeBENTLEY                              MakeBMW
##                             8.32643505                           1.98637804
##                            MakeBUGATTI                             MakeBUICK
##                            38.45164425                           3.71312462
##                           MakeCADILLAC                         MakeCHEVROLET
##                             3.43222186                           3.49272752
##                           MakeCHRYSLER                             MakeDODGE
##                             6.58944883                           5.95454489
##                               MakeFIAT                               MakeFORD
##                             5.51328849                           5.21299252
##                            MakeGENESIS                              MakeGMC
##                             8.02283101                           3.94039686
##                              MakeHONDA                            MakeHYUNDAI
##                             3.21508879                           4.47353990
##                           MakeINFINITI                            MakeJAGUAR
##                             2.12944212                           1.63344092
##                               MakeJEEP                               MakeKIA
##                             5.12916424                           4.99779854
##                        MakeLAMBORGHINI                      MakeLAND ROVER
```

```
##                            9.42518626                             2.41974587
##                             MakeLEXUS                            MakeLINCOLN
##                            2.70791383                             6.29651214
##                           MakeMASERATI                             MakeMAZDA
##                            9.06380850                             2.46721886
##                         MakeMERCEDES-BENZ                              MakeMINI
##                            2.42577576                             1.44307217
##                           MakeMITSUBISHI                            MakeNISSAN
##                            3.12066595                             3.16368348
##                           MakePORSCHE                               MakeRAM
##                            2.17192259                             3.85760879
##                         MakeROLLS-ROYCE                             MakeSCION
##                            6.52859932                             1.53111871
##                             MakeSMART                                MakeSRT
##                           -3.60075385                             4.88214752
##                             MakeSUBARU                            MakeTOYOTA
##                            2.99930016                             4.76315585
##                           MakeVOLKSWAGEN                            MakeVOLVO
##                            2.13481912                             1.98351723
##                  VehicleClassFULL-SIZE                  VehicleClassMID-SIZE
##                           -0.18569372                            -0.42958583
##               VehicleClassMINICOMPACT                   VehicleClassMINIVAN
##                           -1.45696936                             0.73894616
##     VehicleClassPICKUP TRUCK - SMALL  VehicleClassPICKUP TRUCK - STANDARD
##                            4.78293796                             3.57388920
##  VehicleClassSPECIAL PURPOSE VEHICLE VehicleClassSTATION WAGON - MID-SIZE
##                            2.66628160                            -0.19823248
##    VehicleClassSTATION WAGON - SMALL               VehicleClassSUBCOMPACT
##                            0.37546075                            -0.35915036
##               VehicleClassSUV - SMALL           VehicleClassSUV - STANDARD
##                            0.44899040                             2.72316994
##                VehicleClassTWO-SEATER              VehicleClassVAN - CARGO
##                            0.29512077                             0.34189974
##           VehicleClassVAN - PASSENGER                             EngineSize
##                            5.64381651                             1.29642033
##                             Cylinders4                             Cylinders5
##                           -1.87836534                            -2.28561273
##                             Cylinders6                             Cylinders8
##                           -1.17799370                             0.45073307
##                            Cylinders10                            Cylinders12
##                            3.19679840                             7.98179004
##                            Cylinders16                          TransmissionA4
##                                     NA                            -6.56737138
##                          TransmissionA5                         TransmissionA6
##                            0.33213003                            -2.90112184
##                          TransmissionA7                         TransmissionA8
##                            3.21480183                            -1.37395108
##                          TransmissionA9                        TransmissionAM5
##                            0.12862874                             3.72774495
##                         TransmissionAM6                        TransmissionAM7
```

```
##                            1.11731781                             0.92256752
##                         TransmissionAM8                         TransmissionAM9
##                              0.04892340                              6.12000659
##                        TransmissionAS10                         TransmissionAS4
##                             -0.59664132                             -2.11046414
##                         TransmissionAS5                         TransmissionAS6
##                             -2.00175051                             -0.35367490
##                         TransmissionAS7                         TransmissionAS8
##                             -0.57285771                             -0.49201310
##                         TransmissionAS9                          TransmissionAV
##                              0.24331564                              1.03576034
##                        TransmissionAV10                         TransmissionAV6
##                             -1.54094521                             -2.21712939
##                         TransmissionAV7                         TransmissionAV8
##                             -0.45740885                             -1.30857720
##                          TransmissionM5                          TransmissionM6
##                             -1.17291252                             -0.40923607
##                          TransmissionM7                                FuelTypeE
##                              0.72437562                           -132.00949346
##                                FuelTypeN                                FuelTypeX
##                           -108.88984732                           -30.18887364
##                                FuelTypeZ                         ConsumptionCity
##                           -28.57475317                              0.01772886
##                         ConsumptionHwy                         ConsumptionComb
##                              0.04413847                             18.28645403
##                    ConsumptionCombMpg
##                             -1.19720813
```

```
fstatistics1
```

```
##    value
## 4349.919
```

Predikcija na testnom skupu za model1:

```
y_actual = co2.test$CO2Emissions
y_predicted = predict(model1, co2.test)
cat("\n")

RMSE(y_predicted, y_actual)
```

```
## [1] 5.720261
```

```
R2(y_predicted, y_actual)
```

```
## [1] 0.990237
```

```
MAE(y_predicted, y_actual)
```

```
## [1] 3.176749
```

Model 1: Posmatramo model linearne regresije, tako da uključimo sve feature-e osim modela vozila (Make, Vehicle Class, Engine Size, Cylinders, Transmission, Fuel Type, Fuel Consumption City, Fuel Consumption Hwy, Fuel Consumption Comb, Fuel Consumption Comb mpg) za predviđanje emisije CO2. Prisutna je velika tačnost modela (R2 = 0.9899261). Do overfitting-a nije došlo, jer nije značajna razlika Multiple R-squared=0.9899 i Adjusted R-squared=0.9896. Greška nije mnogo velika (RMSE = 5.984571, MAE = 3.312879). Pokušaćemo sada sa manjim broje feature-a.

```
model2 = lm(co2.train$CO2Emissions ~ co2.train$EngineSize, data = co2.train)
summary(model2)

##
## Call:
## lm(formula = co2.train$CO2Emissions ~ co2.train$EngineSize, data =
co2.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -113.326  -18.059   -1.451   19.156  139.710
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           133.9164     1.1524   116.2   <2e-16 ***
## co2.train$EngineSize   36.9639     0.3348   110.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30.43 on 4429 degrees of freedom
## Multiple R-squared:  0.7335, Adjusted R-squared:  0.7334
## F-statistic: 1.219e+04 on 1 and 4429 DF,  p-value: < 2.2e-16

R22 = summary(model2)$r.square
RSS2 = deviance(model2)
koeficijenti2 = coefficients(model2)
fstatistics2 = summary(model2)$fstatistic[1]
R22

## [1] 0.7334562

RSS2

## [1] 4101566

#koeficijenti2
#fstatistics2
```

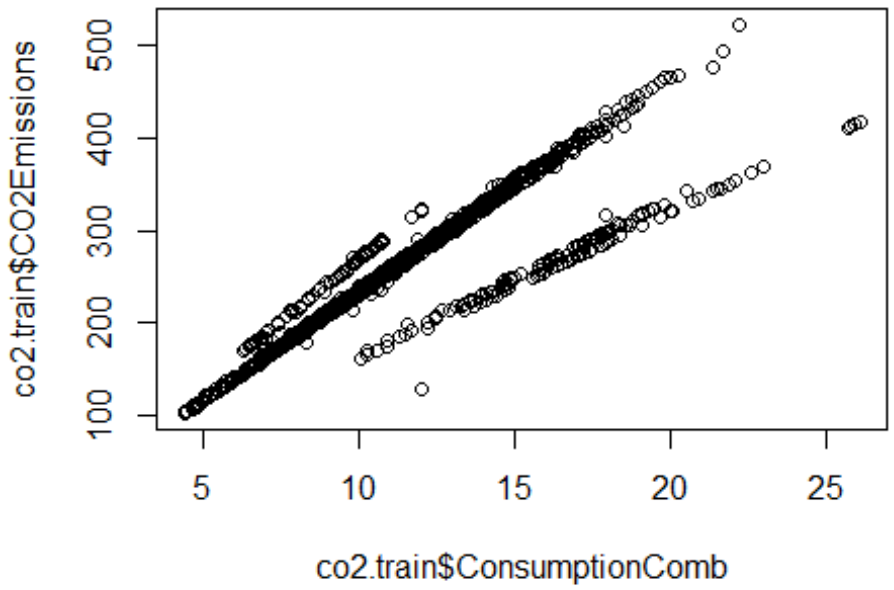
Model2: Pošto već na trening skupu vidimo da je Multiple R-squared=0.7218 dosta manji od prethodnog modela, F-statistika je dosta loša, kao i Residual standard error, nećemo raditi predikciju na testnom skupu niti metrike.

```
model3 = lm(CO2Emissions ~ ConsumptionCombMpg+Cylinders+ConsumptionComb,
data = co2.train)

summary(model3)

##
## Call:
## lm(formula = CO2Emissions ~ ConsumptionCombMpg + Cylinders +
##     ConsumptionComb, data = co2.train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -127.068   -5.732   -0.947    6.932   91.269
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        272.8515     6.6243  41.190  < 2e-16 ***
## ConsumptionCombMpg  -3.6462     0.1099 -33.168  < 2e-16 ***
## Cylinders4           2.4752     2.4373   1.016    0.310
## Cylinders5           7.0580     5.1867   1.361    0.174
## Cylinders6          16.6876     2.5532   6.536 7.04e-11 ***
## Cylinders8          40.9641     2.7098  15.117  < 2e-16 ***
## Cylinders10         68.5501     4.4860  15.281  < 2e-16 ***
## Cylinders12         78.8915     3.3041  23.877  < 2e-16 ***
## Cylinders16        172.1365    13.3473  12.897  < 2e-16 ***
## ConsumptionComb      5.6042     0.2921  19.189  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.25 on 4421 degrees of freedom
## Multiple R-squared:  0.9043, Adjusted R-squared:  0.9041
## F-statistic:  4644 on 9 and 4421 DF,  p-value: < 2.2e-16

R23 = summary(model3)$r.square
RSS3 = deviance(model3) #MSE
koeficijenti3 = coefficients(model3)
fstatistics3 = summary(model3)$fstatistic[1]
R23

## [1] 0.9043396

RSS3

## [1] 1472019
```

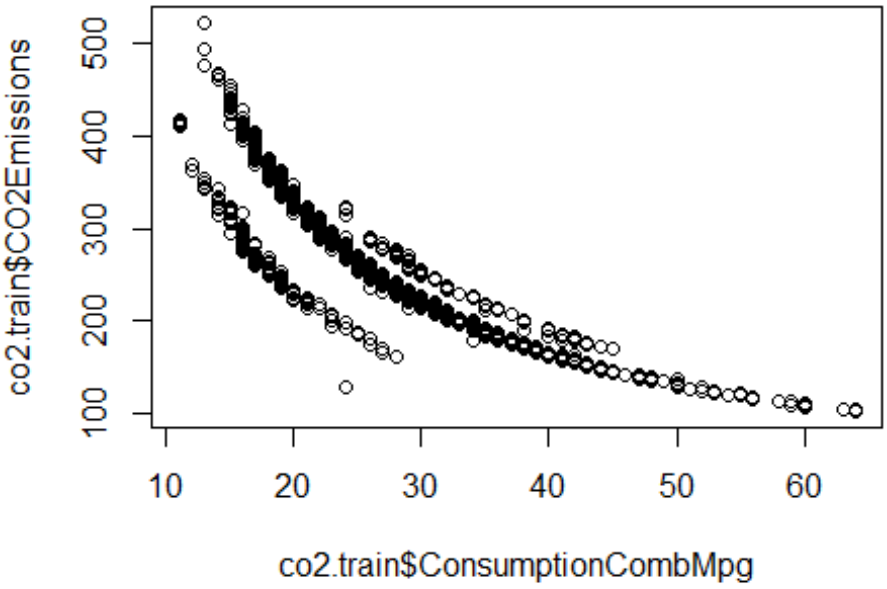
```
koeficijenti3

##        (Intercept) ConsumptionCombMpg         Cylinders4
Cylinders5
##         272.851530          -3.646219           2.475166
7.057954
##         Cylinders6         Cylinders8        Cylinders10
Cylinders12
##          16.687624          40.964127          68.550106
78.891502
##        Cylinders16    ConsumptionComb
##         172.136476           5.604182

fstatistics3

##   value
## 4643.84
```

A sada ćemo vizuelno predstaviti linearne zavisnosti:

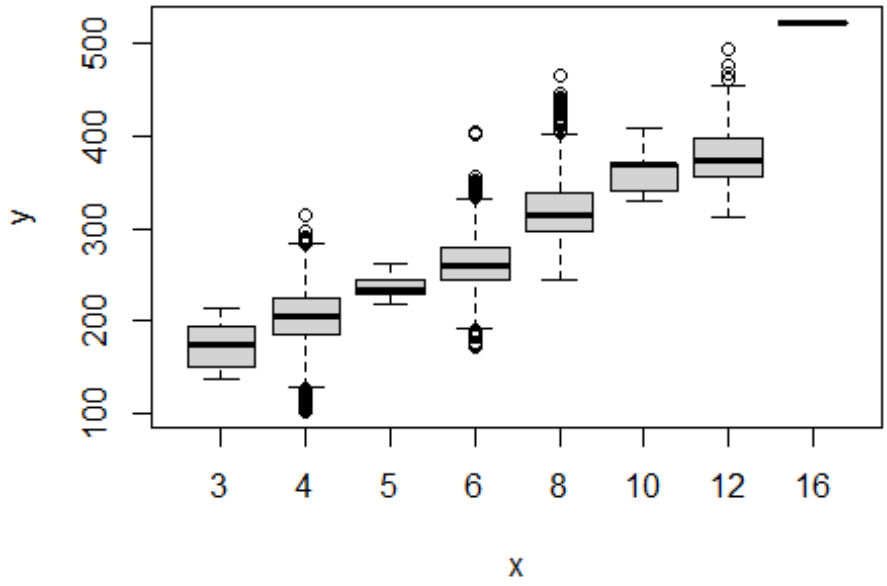
```
plot(co2.train$ConsumptionComb, co2.train$CO2Emissions)
```




```
plot(co2.train$ConsumptionCombMpg, co2.train$CO2Emissions)
```




```
plot(co2.train$Cylinders, co2.train$CO2Emissions)
```



Predikcija na testnom skupu za model3:

```
y_actual = co2.test$CO2Emissions
y_predicted = predict(model3, co2.test)
RMSE(y_predicted, y_actual)
```

```
## [1] 18.15236
```

```
R2(y_predicted, y_actual)
```

```
## [1] 0.9017863
```

```
MAE(y_predicted, y_actual)
```

```
## [1] 11.26735
```

```
#CROSS VALIDATION

tc = trainControl(method = "CV", number = 10)
modelCV = train(CO2Emissions ~ .,
                data = co2.train[,-2], method = "lm",
                trControl = tc)

modelCV
```

```
## Linear Regression
##
## 4431 samples
##   10 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 3988, 3988, 3989, 3988, 3987, 3988, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   6.099225  0.9887764  3.329281
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

Predikcija na testnom skupu za modelCV:

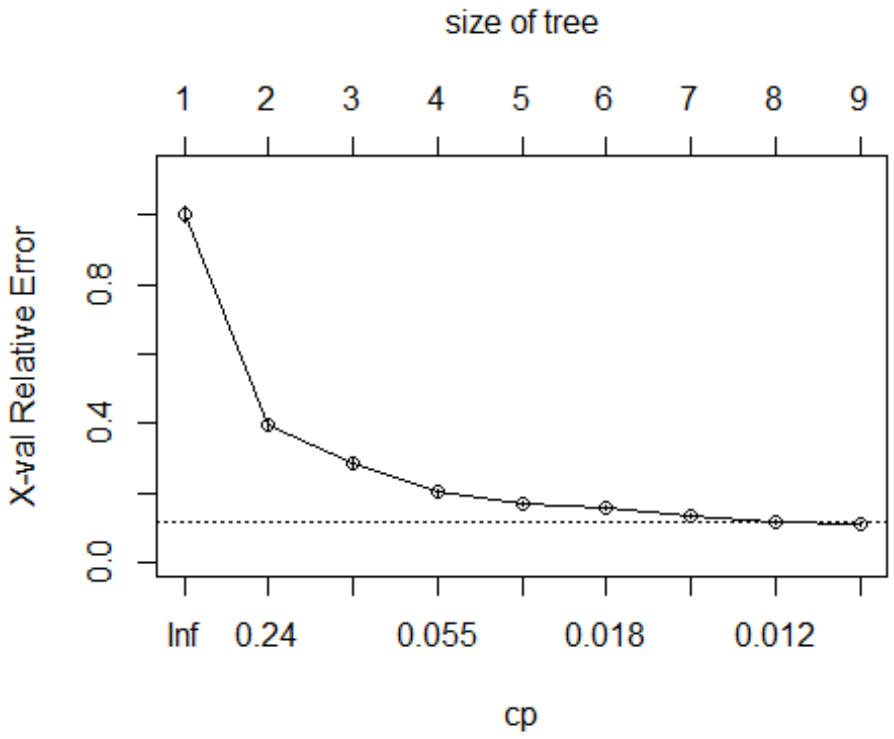
```
y_actual = co2.test$CO2Emissions
y_predicted = predict(modelCV, co2.test)
cat("\n")
```

```
RMSE(y_predicted, y_actual)

## [1] 5.720261

R2(y_predicted, y_actual)

## [1] 0.990237

MAE(y_predicted, y_actual)

## [1] 3.176749
```

2) Stabla odlučivanja

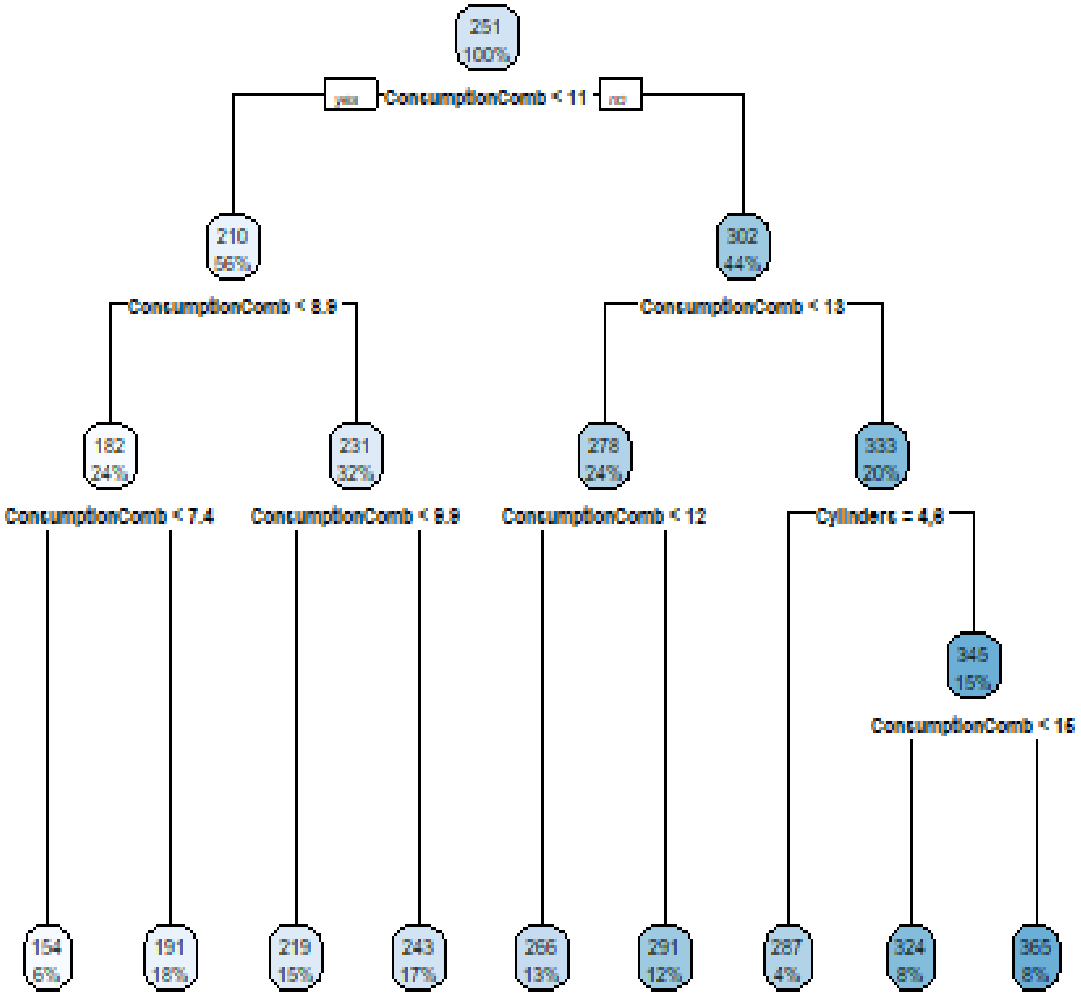
Proveravamo da li postoji overfitting:

```
model4 = rpart(CO2Emissions ~ ConsumptionCombMpg+Cylinders+ConsumptionComb,
data=co2.train)
plotcp(model4)
```

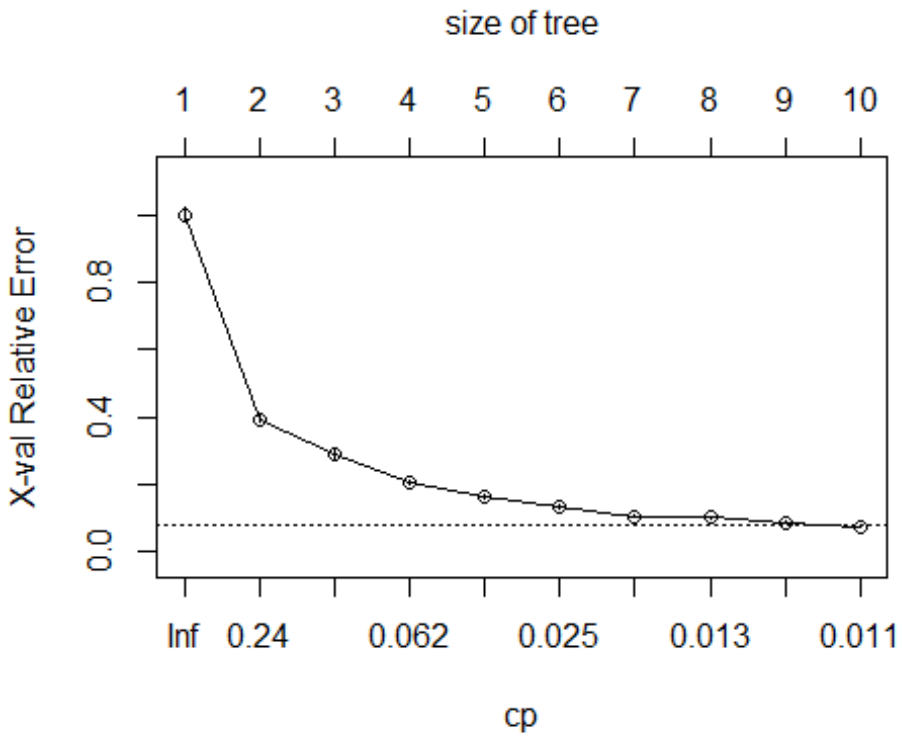


Ne postoji overfitting.

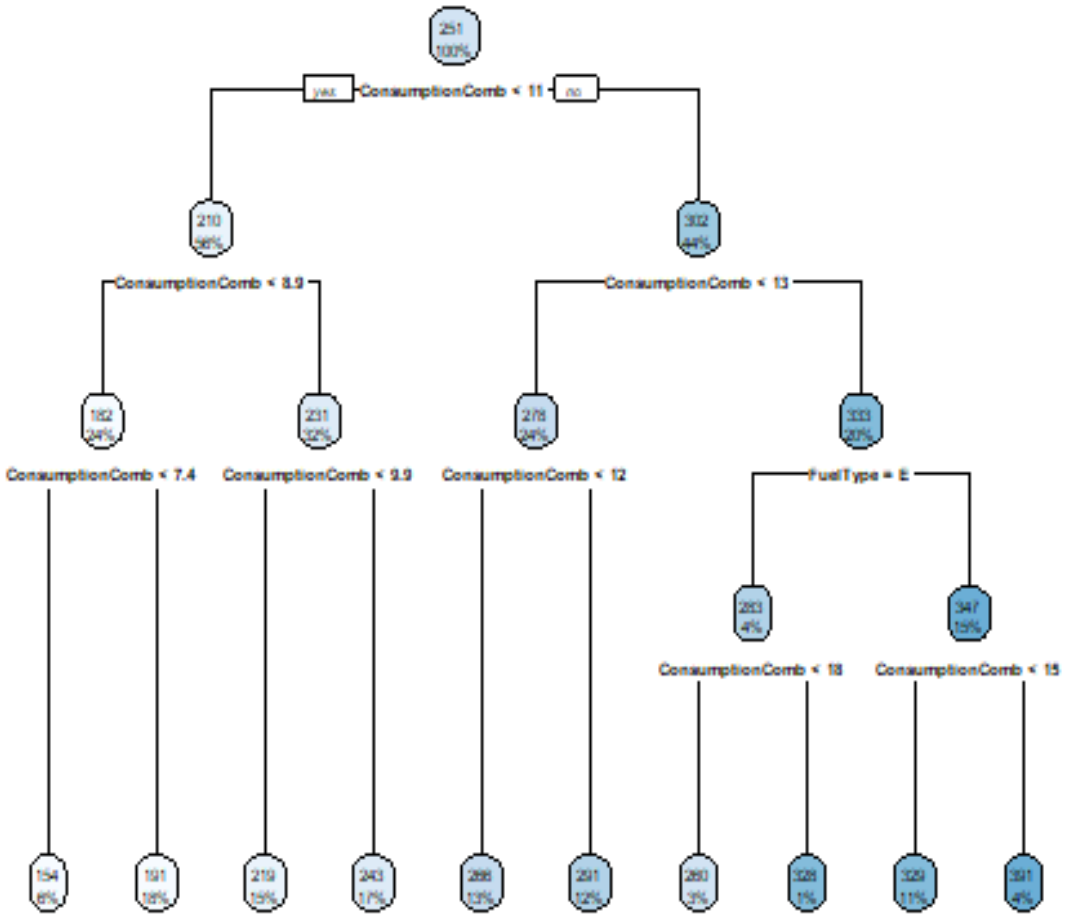
```
model4 %>% rpart.plot()
```




```
model5 = rpart(CO2Emissions ~ ., data=co2.train[,-2])
plotcp(model5)
```




```
model5 %>% rpart.plot()
```



Predikcija na testnom skupu za model4:

```
y_actual = co2.test$CO2Emissions
y_predicted = predict(model4, co2.test)
RMSE(y_predicted, y_actual)

## [1] 19.38593

R2(y_predicted, y_actual)

## [1] 0.8878129

MAE(y_predicted, y_actual)

## [1] 11.9697
```

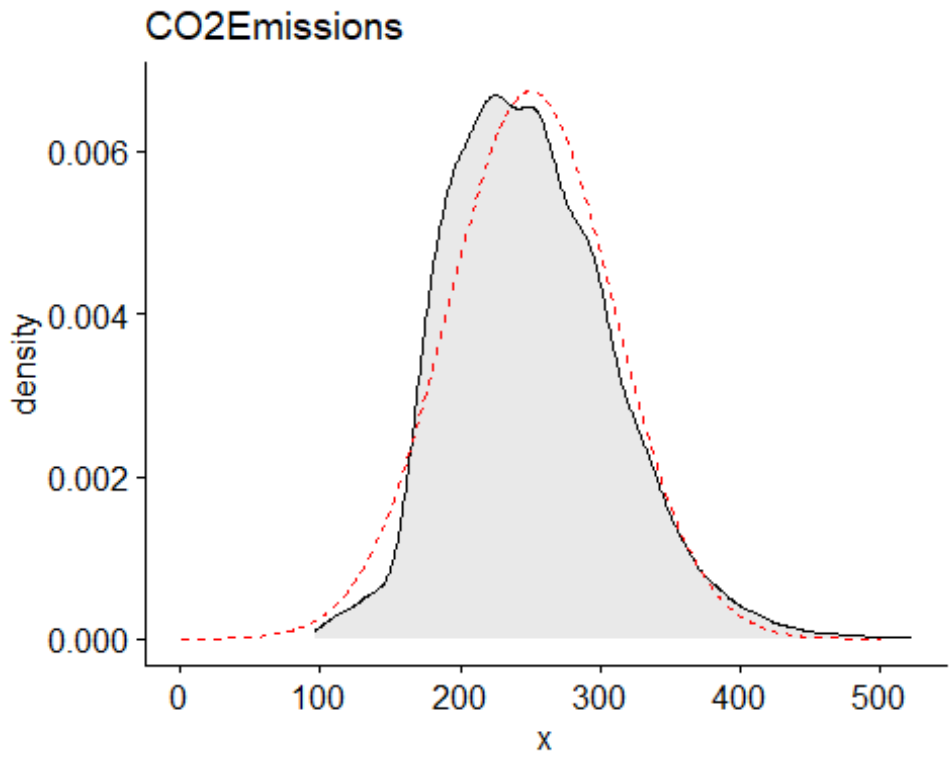
Predikcija na testnom skupu za model5:

```
y_actual = co2.test$CO2Emissions
y_predicted = predict(model5, co2.test)
RMSE(y_predicted, y_actual)

## [1] 15.11111

R2(y_predicted, y_actual)

## [1] 0.9318228

MAE(y_predicted, y_actual)

## [1] 10.30736
```

Algoritam za predviđanje koji je najoptimalniji od testiranih jeste linearna regresija (modelCV), jer ima najveću preciznost (accuracy).

3) Random Forest

```
ggdensity(co2$CO2Emissions, fill = "lightgray", title = "CO2Emissions") +
  stat_overlay_normal_density(color = "red", linetype = "dashed")
```



```
co2 = co2 %>% mutate(EmissionGroup = ifelse(CO2Emissions<150, "First",
ifelse(CO2Emissions<250, "Second", ifelse(CO2Emissions<350, "Third",
"Fourth"))))
co2
```

```
##                Make          Model          VehicleClass EngineSize
Cylinders
##    1:          FORD  FLEX AWD GTDI        SUV - STANDARD        3.5
6
##    2: MERCEDES-BENZ          B 250             MID-SIZE        2.0
4
##    3:         BUICK          REGAL             MID-SIZE        2.0
4
##    4:       PORSCHE  911 CARRERA 4           MINICOMPACT        3.0
6
##    5:          AUDI     A4 QUATTRO              COMPACT        2.0
4
##   ---
## 7381:    VOLKSWAGEN         Tiguan           SUV - SMALL        2.0
4
## 7382:    VOLKSWAGEN Tiguan 4MOTION           SUV - SMALL        2.0
4
## 7383:         VOLVO     V60 T6 AWD STATION WAGON - SMALL        2.0
4
## 7384:         VOLVO    XC60 T5 AWD           SUV - SMALL        2.0
4
## 7385:         VOLVO    XC90 T5 AWD        SUV - STANDARD        2.0
4
##       Transmission FuelType ConsumptionCity ConsumptionHwy ConsumptionComb
##    1:          AS6        X        15.70000           11.2             13.7
##    2:          AS7        Z         9.70000            6.6              8.3
##    3:          AS6        Z        11.40000            7.9              9.8
##    4:          AM7        Z        10.70000            8.3              9.6
##    5:          AS8        Z        11.00000            7.8              9.6
##   ---
## 7381:          AS8        X        10.50000            8.1              9.4
## 7382:          AS8        X        11.50000            8.7             10.2
## 7383:          AS8        Z        12.09819            7.4              9.4
## 7384:          AS8        Z        12.09819            8.3              9.9
## 7385:          AS8        Z        11.20000            8.3              9.9
##       ConsumptionCombMpg CO2Emissions EmissionGroup
##    1:                 21          322         Third
##    2:                 34          179        Second
##    3:                 29          231        Second
##    4:                 29          225        Second
##    5:                 29          221        Second
##   ---
## 7381:                 30          221        Second
## 7382:                 28          241        Second
## 7383:                 30          219        Second
```

```
## 7384:                 29          232        Second
## 7385:                 29          232        Second
```

```
library("randomForest")
```

```
modelRF = randomForest(CO2Emissions ~ ., data=co2.train[,-2])
table(co2.train$EmissionGroup)
```

```
## < table of extent 0 >
```

```
modelRF
```

```
##
## Call:
##  randomForest(formula = CO2Emissions ~ ., data = co2.train[, -2])
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##           Mean of squared residuals: 24.36594
##                     % Var explained: 99.3
```

```
y_actual = co2.test$CO2Emissions
y_predicted = predict(modelRF, co2.test)
cat("\n")
```

```
#MAE - Mean Absolute Error
mean(abs(y_actual-y_predicted))
```

```
## [1] 2.440922
```

```
MAE(y_predicted, y_actual)
```

```
## [1] 2.440922
```

```
#MSE - Mean Squared Error
mean((y_actual - y_predicted)^2)
```

```
## [1] 20.95052
```

```
#RMSE je korenovani MSE
RMSE(y_predicted, y_actual)
```

```
## [1] 4.577173
```

```
#MAPE tj. Prosecna apsolutna razlika - ove je u %
#Mean Absolute Percentage Error
mape=mean(abs((y_actual-y_predicted)/y_actual)) * 100
mape
```

```
## [1] 0.9942168
```

```
#Accuracy
print(round(100*(1-mape),2))
```

```
## [1] 0.58
```