Chapter02.R

riserate

Fri Aug 4 15:46:37 2017

#Load the R packages   
library(plyr)  
library(ggplot2)  
library(reshape2)  
  
  
#Importing Automobile Fuel Efficiency Data into R  
  
  
vehicles <- read.csv(unz("~/r-code/Practical Data Science CookBook/0246OS\_Code & Data/Chapter02/code/vehicles.csv.zip", "vehicles.csv"), stringsAsFactors = F)  
  
head(vehicles)

## barrels08 barrelsA08 charge120 charge240 city08 city08U cityA08 cityA08U  
## 1 15.68944 0 0 0 19 0 0 0  
## 2 29.95056 0 0 0 9 0 0 0  
## 3 12.19557 0 0 0 23 0 0 0  
## 4 29.95056 0 0 0 10 0 0 0  
## 5 17.33749 0 0 0 17 0 0 0  
## 6 14.96429 0 0 0 21 0 0 0  
## cityCD cityE cityUF co2 co2A co2TailpipeAGpm co2TailpipeGpm comb08  
## 1 0 0 0 -1 -1 0 423.1905 21  
## 2 0 0 0 -1 -1 0 807.9091 11  
## 3 0 0 0 -1 -1 0 329.1481 27  
## 4 0 0 0 -1 -1 0 807.9091 11  
## 5 0 0 0 -1 -1 0 467.7368 19  
## 6 0 0 0 -1 -1 0 403.9545 22  
## comb08U combA08 combA08U combE combinedCD combinedUF cylinders displ  
## 1 0 0 0 0 0 0 4 2.0  
## 2 0 0 0 0 0 0 12 4.9  
## 3 0 0 0 0 0 0 4 2.2  
## 4 0 0 0 0 0 0 8 5.2  
## 5 0 0 0 0 0 0 4 2.2  
## 6 0 0 0 0 0 0 4 1.8  
## drive engId eng\_dscr feScore fuelCost08  
## 1 Rear-Wheel Drive 9011 (FFS) -1 2350  
## 2 Rear-Wheel Drive 22020 (GUZZLER) -1 4450  
## 3 Front-Wheel Drive 2100 (FFS) -1 1800  
## 4 Rear-Wheel Drive 2850 -1 4450  
## 5 4-Wheel or All-Wheel Drive 66031 (FFS,TRBO) -1 2850  
## 6 Front-Wheel Drive 66020 (FFS) -1 2250  
## fuelCostA08 fuelType fuelType1 ghgScore ghgScoreA highway08  
## 1 0 Regular Regular Gasoline -1 -1 25  
## 2 0 Regular Regular Gasoline -1 -1 14  
## 3 0 Regular Regular Gasoline -1 -1 33  
## 4 0 Regular Regular Gasoline -1 -1 12  
## 5 0 Premium Premium Gasoline -1 -1 23  
## 6 0 Regular Regular Gasoline -1 -1 24  
## highway08U highwayA08 highwayA08U highwayCD highwayE highwayUF hlv hpv  
## 1 0 0 0 0 0 0 0 0  
## 2 0 0 0 0 0 0 0 0  
## 3 0 0 0 0 0 0 19 77  
## 4 0 0 0 0 0 0 0 0  
## 5 0 0 0 0 0 0 0 0  
## 6 0 0 0 0 0 0 0 0  
## id lv2 lv4 make model mpgData phevBlended pv2 pv4  
## 1 1 0 0 Alfa Romeo Spider Veloce 2000 Y false 0 0  
## 2 10 0 0 Ferrari Testarossa N false 0 0  
## 3 100 0 0 Dodge Charger Y false 0 0  
## 4 1000 0 0 Dodge B150/B250 Wagon 2WD N false 0 0  
## 5 10000 0 14 Subaru Legacy AWD Turbo N false 0 90  
## 6 10001 0 15 Subaru Loyale N false 0 88  
## range rangeCity rangeCityA rangeHwy rangeHwyA trany UCity  
## 1 0 0 0 0 0 Manual 5-spd 23.3333  
## 2 0 0 0 0 0 Manual 5-spd 11.0000  
## 3 0 0 0 0 0 Manual 5-spd 29.0000  
## 4 0 0 0 0 0 Automatic 3-spd 12.2222  
## 5 0 0 0 0 0 Manual 5-spd 21.0000  
## 6 0 0 0 0 0 Automatic 3-spd 27.0000  
## UCityA UHighway UHighwayA VClass year youSaveSpend guzzler  
## 1 0 35.0000 0 Two Seaters 1985 -1000   
## 2 0 19.0000 0 Two Seaters 1985 -11500 T  
## 3 0 47.0000 0 Subcompact Cars 1985 1750   
## 4 0 16.6667 0 Vans 1985 -11500   
## 5 0 32.0000 0 Compact Cars 1993 -3500   
## 6 0 33.0000 0 Compact Cars 1993 -500   
## trans\_dscr tCharger sCharger atvType fuelType2 rangeA evMotor mfrCode  
## 1 NA   
## 2 NA   
## 3 SIL NA   
## 4 NA   
## 5 TRUE   
## 6 NA

labels <- read.table("~/r-code/Practical Data Science CookBook/0246OS\_Code & Data/Chapter02/code/varlabels.txt", sep = "-", header = FALSE)  
  
labels <- do.call(rbind, strsplit(readLines("~/r-code/Practical Data Science CookBook/0246OS\_Code & Data/Chapter02/code/varlabels.txt"), " - "))  
head(labels)

## [,1]   
## [1,] "atvtype"   
## [2,] "barrels08"   
## [3,] "barrelsA08"  
## [4,] "charge120"   
## [5,] "charge240"   
## [6,] "city08"   
## [,2]   
## [1,] "type of alternative fuel or advanced technology vehicle"   
## [2,] "annual petroleum consumption in barrels for fuelType1 (1)"  
## [3,] "annual petroleum consumption in barrels for fuelType2 (1)"  
## [4,] "time to charge an electric vehicle in hours at 120 V"   
## [5,] "time to charge an electric vehicle in hours at 240 V"   
## [6,] "city MPG for fuelType1 (2)"

#Exploring and Describing the Fuel Efficiency Data  
  
nrow(vehicles)

## [1] 34287

ncol(vehicles)

## [1] 74

names(vehicles)

## [1] "barrels08" "barrelsA08" "charge120"   
## [4] "charge240" "city08" "city08U"   
## [7] "cityA08" "cityA08U" "cityCD"   
## [10] "cityE" "cityUF" "co2"   
## [13] "co2A" "co2TailpipeAGpm" "co2TailpipeGpm"   
## [16] "comb08" "comb08U" "combA08"   
## [19] "combA08U" "combE" "combinedCD"   
## [22] "combinedUF" "cylinders" "displ"   
## [25] "drive" "engId" "eng\_dscr"   
## [28] "feScore" "fuelCost08" "fuelCostA08"   
## [31] "fuelType" "fuelType1" "ghgScore"   
## [34] "ghgScoreA" "highway08" "highway08U"   
## [37] "highwayA08" "highwayA08U" "highwayCD"   
## [40] "highwayE" "highwayUF" "hlv"   
## [43] "hpv" "id" "lv2"   
## [46] "lv4" "make" "model"   
## [49] "mpgData" "phevBlended" "pv2"   
## [52] "pv4" "range" "rangeCity"   
## [55] "rangeCityA" "rangeHwy" "rangeHwyA"   
## [58] "trany" "UCity" "UCityA"   
## [61] "UHighway" "UHighwayA" "VClass"   
## [64] "year" "youSaveSpend" "guzzler"   
## [67] "trans\_dscr" "tCharger" "sCharger"   
## [70] "atvType" "fuelType2" "rangeA"   
## [73] "evMotor" "mfrCode"

length(unique(vehicles[, "year"]))

## [1] 31

first\_year <- min(vehicles[, "year"])  
last\_year <- max(vehicles[, "year"])  
length(unique(vehicles$year))

## [1] 31

table(vehicles$fuelType1)

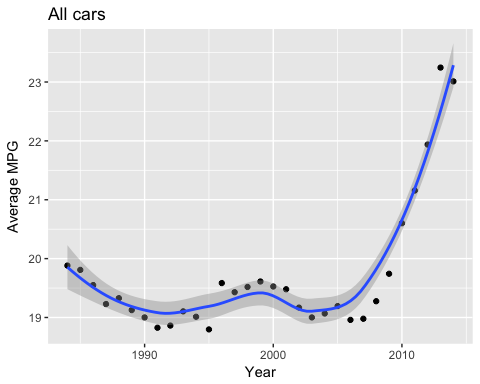
##   
## Diesel Electricity Midgrade Gasoline Natural Gas   
## 1025 56 41 57   
## Premium Gasoline Regular Gasoline   
## 8521 24587

vehicles$trany[vehicles$trany == ""] <- NA  
vehicles$trany2 <- ifelse(substr(vehicles$trany, 1, 4) == "Auto", "Auto", "Manual")  
vehicles$trany <- as.factor(vehicles$trany)  
table(vehicles$trany2)

##   
## Auto Manual   
## 22451 11825

#Analyzing Automobile Fuel Efficiency Over Time  
  
mpgByYr <- ddply(vehicles, ~year, summarise, avgMPG = mean(comb08), avgHghy = mean(highway08), avgCity = mean(city08))  
ggplot(mpgByYr, aes(year, avgMPG)) + geom\_point() + geom\_smooth() + xlab("Year") + ylab("Average MPG") + ggtitle("All cars")

## `geom\_smooth()` using method = 'loess'

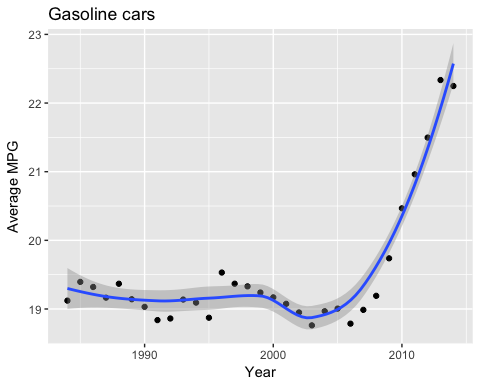


table(vehicles$fuelType1)

##   
## Diesel Electricity Midgrade Gasoline Natural Gas   
## 1025 56 41 57   
## Premium Gasoline Regular Gasoline   
## 8521 24587

gasCars <- subset(vehicles, fuelType1 %in% c("Regular Gasoline", "Premium Gasoline", "Midgrade Gasoline") & fuelType2 == "" & atvType != "Hybrid")  
mpgByYr\_Gas <- ddply(gasCars, ~year, summarise, avgMPG = mean(comb08))  
ggplot(mpgByYr\_Gas, aes(year, avgMPG)) + geom\_point() + geom\_smooth() + xlab("Year") + ylab("Average MPG") + ggtitle("Gasoline cars")

## `geom\_smooth()` using method = 'loess'



typeof(gasCars$displ)

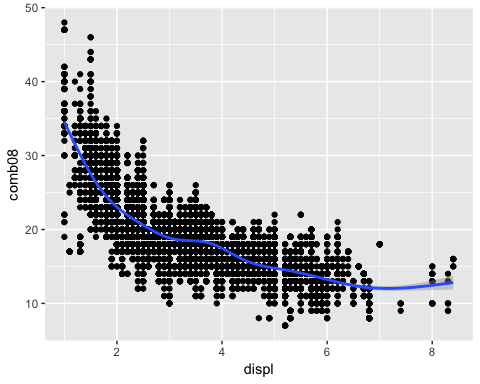
## [1] "character"

gasCars$displ <- as.numeric(gasCars$displ)  
ggplot(gasCars, aes(displ, comb08)) + geom\_point() + geom\_smooth()

## `geom\_smooth()` using method = 'gam'

## Warning: Removed 2 rows containing non-finite values (stat\_smooth).

## Warning: Removed 2 rows containing missing values (geom\_point).

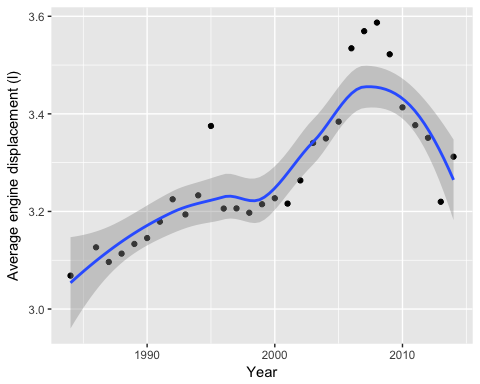


avgCarSize <- ddply(gasCars, ~year, summarise, avgDispl = mean(displ))  
ggplot(avgCarSize, aes(year, avgDispl)) + geom\_point() + geom\_smooth() + xlab("Year") + ylab("Average engine displacement (l)")

## `geom\_smooth()` using method = 'loess'

## Warning: Removed 1 rows containing non-finite values (stat\_smooth).

## Warning: Removed 1 rows containing missing values (geom\_point).



byYear <- ddply(gasCars, ~year, summarise, avgMPG = mean(comb08), avgDispl = mean(displ))  
head(byYear)

## year avgMPG avgDispl  
## 1 1984 19.12162 3.068449  
## 2 1985 19.39469 NA  
## 3 1986 19.32046 3.126514  
## 4 1987 19.16457 3.096474  
## 5 1988 19.36761 3.113558  
## 6 1989 19.14196 3.133393

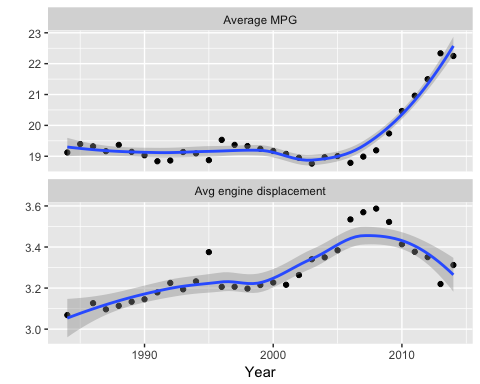
byYear2 = melt(byYear, id = "year")  
levels(byYear2$variable) <- c("Average MPG", "Avg engine displacement")  
head(byYear2)

## year variable value  
## 1 1984 Average MPG 19.12162  
## 2 1985 Average MPG 19.39469  
## 3 1986 Average MPG 19.32046  
## 4 1987 Average MPG 19.16457  
## 5 1988 Average MPG 19.36761  
## 6 1989 Average MPG 19.14196

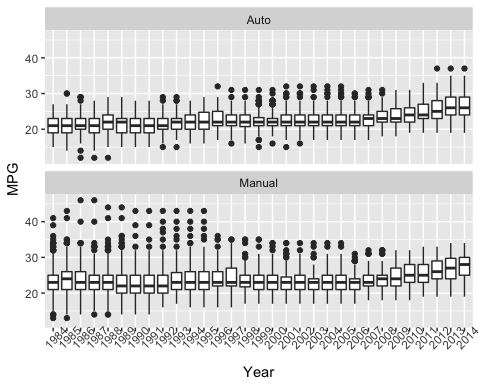
ggplot(byYear2, aes(year, value)) + geom\_point() + geom\_smooth() + facet\_wrap(~variable, ncol = 1, scales = "free\_y") + xlab("Year") + ylab("")

## `geom\_smooth()` using method = 'loess'

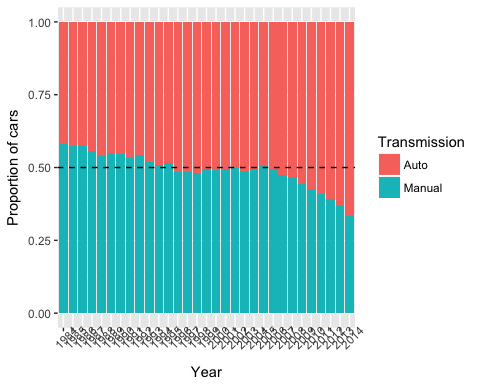
## Warning: Removed 1 rows containing non-finite values (stat\_smooth).  
  
## Warning: Removed 1 rows containing missing values (geom\_point).



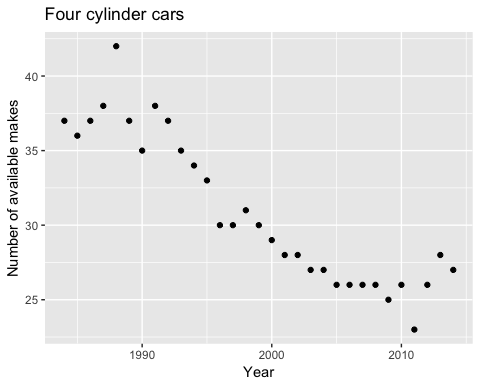
gasCars4 <- subset(gasCars, cylinders == "4")  
ggplot(gasCars4, aes(factor(year), comb08)) + geom\_boxplot() + facet\_wrap(~trany2, ncol = 1) + theme(axis.text.x = element\_text(angle = 45)) + labs(x = "Year", y = "MPG")



ggplot(gasCars4, aes(factor(year), fill = factor(trany2))) + geom\_bar(position = "fill") + labs(x = "Year", y = "Proportion of cars", fill = "Transmission") + theme(axis.text.x = element\_text(angle = 45)) + geom\_hline(yintercept = 0.5, linetype = 2)



#Investigating the Makes and Models of Automobiles  
  
carsMake <- ddply(gasCars4, ~year, summarise, numberOfMakes = length(unique(make)))  
ggplot(carsMake, aes(year, numberOfMakes)) + geom\_point() + labs(x = "Year", y = "Number of available makes") + ggtitle("Four cylinder cars")



uniqMakes <- dlply(gasCars4, ~year, function(x) unique(x$make))  
commonMakes <- Reduce(intersect, uniqMakes)  
commonMakes

## [1] "Ford" "Honda" "Toyota" "Volkswagen" "Chevrolet"   
## [6] "Chrysler" "Nissan" "Dodge" "Mazda" "Mitsubishi"  
## [11] "Subaru" "Jeep"

carsCommonMakes4 <- subset(gasCars4, make %in% commonMakes)  
avgMPG\_commonMakes <- ddply(carsCommonMakes4, ~year + make, summarise, avgMPG = mean(comb08))  
ggplot(avgMPG\_commonMakes, aes(year, avgMPG)) + geom\_line() + facet\_wrap(~make, nrow = 3)

