MMSS 311-2 HW0

Yushi Liu

4/12/2019

```r
packages <- c("dplyr", "ggplot2", "lubridate", "stringr", "foreign")
load.packages <- function(x) {
  if (!require(x, character.only = TRUE)) {
    # character.only = TRUE specifies that the argument being passed to the function is in character ty
    install.packages(x, dependencies = TRUE)
    # setting dependencies to TRUE will also install other packages that are necessary
    library(x, character.only = TRUE) # load the package once it has been installed
  }
}
lapply(packages, load.packages)
```

```
## Loading required package: dplyr

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## Loading required package: ggplot2

## Loading required package: lubridate

##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date

## Loading required package: stringr

## Loading required package: foreign

## [[1]]
## NULL
##
## [[2]]
## NULL
```

```
##
## [[3]]
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
```

Problem 1

(a) A vector with the numbers 1–5 in order

```r
v <- c(1:5)
v
```

```
## [1] 1 2 3 4 5
```

(b) A scalar named Mindy that takes the value 12

```r
Mindy <- 12
Mindy
```

```
## [1] 12
```

(c) A 2×3 matrix with the numbers 1–6 in order by rows

```r
byrow <- matrix(1:6, nrow = 2, ncol = 3, byrow = TRUE)
byrow
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    3
## [2,]    4    5    6
```

(d)

```r
bycol <- matrix(1:6, nrow = 2, ncol = 3)
bycol
```

```
##      [,1] [,2] [,3]
## [1,]    1    3    5
## [2,]    2    4    6
```

(e)

```r
ones <- matrix(1, nrow = 10, ncol = 10)
```

(f)

```r
str <- c("THIS", "IS", "A", "VECTOR")
```

(g)

```r
sum3 <- function(a, b, c){
  return(a+b+c)
  print(a+b+c)
}
```

(h)

```r
YON <- function(n){
  if(n <= 10){
    return('Yes')
  }
  return('No')
}
```

(i)

```r
g <- rnorm(1000, mean = 10, sd = 1)
```

(j)

```r
y <- rnorm(1000, mean = 5, sd = 0.5)
```

(k)

```r
x <- NULL
for (i in 1:1000){
  x[i] <- mean(sample(g, 10, replace = TRUE))
}
```

(j)

```r
lm <- lm(y ~ x)
summary(lm)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.50676 -0.32728  0.01433  0.31737  1.59741
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.56026    0.50151  11.087   <2e-16 ***
```

```
## x            -0.05556    0.04984  -1.115    0.265
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4956 on 998 degrees of freedom
## Multiple R-squared:  0.001244,   Adjusted R-squared:  0.000243
## F-statistic: 1.243 on 1 and 998 DF,  p-value: 0.2652
```

The coefficient is 0.03 but the p-value is not less than 0.05, so y doesn't have a significant increasing trend against x.

Problem 2

```r
setwd("~/Documents/GitHub/MMSS-311-2")
pums <- read.csv("pums_chicago.csv")
dim(pums)
```

```
## [1] 50000   204
```

(b) There are 204 variables and 50000 observations.
(c) See below

```r
annual_income <- mean(pums$PINCP, na.rm = TRUE)
```

(d)

```r
pums$PINCP_LOG <- log(pums$PINCP)
```

```
## Warning in log(pums$PINCP): NaNs produced
```

NaNs produced because some of the rows for annual incomes are NaNs. (e)

```r
pums$GRAD.DUMMY <- ifelse(pums$SCHL >= 18, "grad","not grad")
```

(f)

```r
df = subset(pums, select = -c(SERIALNO))
```

(g)

```r
write.csv(df, file = 'newdata.csv')
```

(h)

```
under16 = pums[is.na(pums$ESR) == TRUE,]
pums_drop = pums[is.na(pums$ESR) == FALSE,]
employed = pums_drop[pums_drop$ESR == 1 | pums_drop$ESR == 2 , ]
unemployed = pums_drop[pums_drop$ESR == 3,]
armforce = pums_drop[pums_drop$ESR == 4 | pums_drop$ESR == 5 ,]
notinl = pums_drop[pums_drop$ESR == 6,]
```

Note that the "employed" category excludes the employed in armed forces. In words, "employed" dataframe only includes civilian employed. (i)

```
new_frame = pums_drop[pums_drop$ESR == 1 | pums_drop$ESR == 2 | pums_drop$ESR == 4 | pums_drop$ESR == 5
```

(j)

```
library(dplyr)
employed_af = select(pums, c(AGEP, RAC1P, PINCP_LOG))
```

(k)-(i) First dropped all entries containing "NA".

```
travelt = pums[is.na(pums$JWMNP) == FALSE,]$JWMNP
mean(travelt)
```

```
## [1] 34.83889
```

```
quantile(travelt, c(0.5, 0.8))
```

```
## 50% 80%
##  30  45
```

(k)-(ii)

```
cor(pums$JWMNP, pums$WAGP, use = "complete.obs")
```

```
## [1] -0.04205232
```

(k)-(iii) (iv)Scatterplot of age and log income

```
pdf("graph for hw0.pdf")
plot(x=pums$AGEP, y=pums$PINCP_LOG)
dev.off()
```

```
## pdf
##   2
```

(k)-(v) crosstab of ESR by race RAC1P

```r
cst <- table(pums$ESR, pums$RAC1P)
cst
```

```
##
##        1     2     3     4     5     6     7     8     9
##   1 12870  5786    36     0    24  1746     7  2502   521
##   2   258   147     0     0     0    31     0    66     8
##   3   794  1473     2     0     4   109     0   268    57
##   4     4     5     0     0     0     0     1     0     1
##   6  5618  5533    33     2    19   899     1  1283   240
```

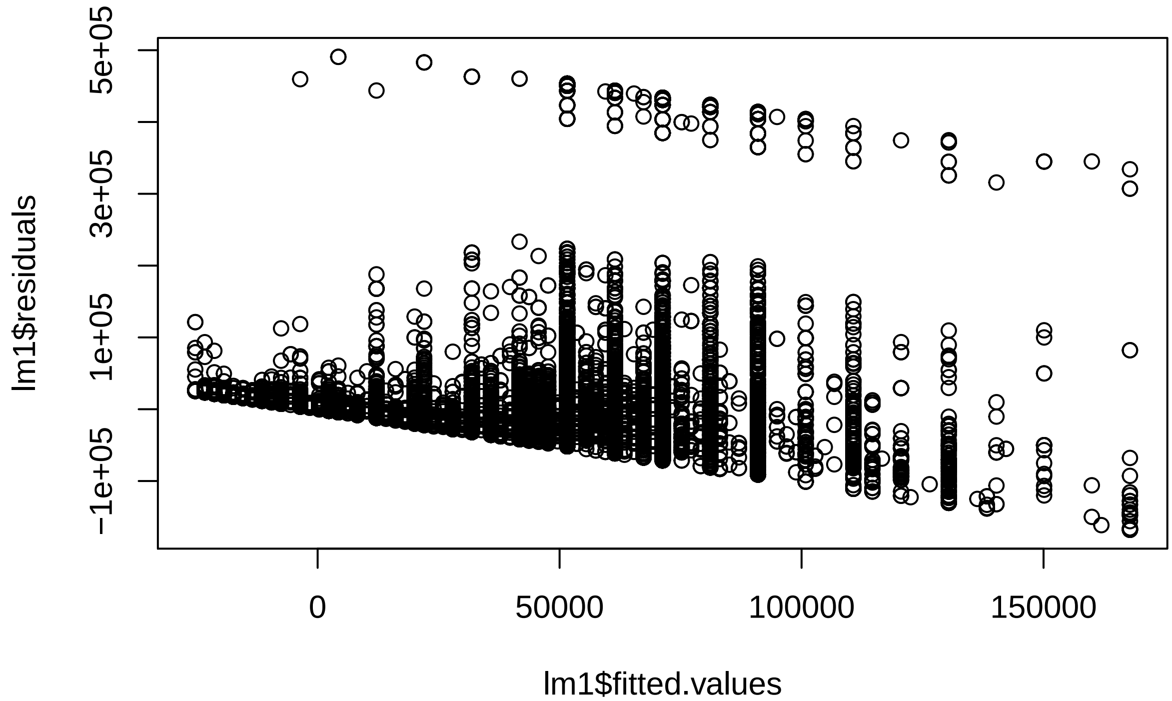
(k)-(vi) Lienar regression of WAGP on WKHP

```r
lm1 <- lm(WAGP ~ WKHP, data = pums_drop)
summary(lm1)
```

```
##
## Call:
## lm(formula = WAGP ~ WKHP, data = pums_drop)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -167856  -27577  -11577    9491  490723
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -27256.47    1253.63  -21.74   <2e-16 ***
## WKHP          1970.83      30.97   63.64   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 61490 on 26206 degrees of freedom
##   (14140 observations deleted due to missingness)
## Multiple R-squared:  0.1339, Adjusted R-squared:  0.1338
## F-statistic:  4050 on 1 and 26206 DF,  p-value: < 2.2e-16
```

(k)-(vii) Plot residuals from this regression against the fitted values

```r
plot(lm1$fitted.values,lm1$residuals)
```



The residual plot shows that there exists a linear relationship between residuals and fitted values. The distribution of residuals are not random, so there might exist omitted variable bias in this model.

(l)-(i) A linear regression of miles per gallon (mpg) on weight (wt)

```
mc <- mtcars
colnames(mtcars)
```

```
##  [1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "vs"   "am"   "gear"
## [11] "carb"
```

```
lm2 <- lm(mtcars$mpg~mtcars$wt)
summary(lm2)
```

```
##
## Call:
## lm(formula = mtcars$mpg ~ mtcars$wt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5432 -2.3647 -0.1252  1.4096  6.8727
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.2851     1.8776  19.858  < 2e-16 ***
## mtcars$wt    -5.3445     0.5591  -9.559 1.29e-10 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.046 on 30 degrees of freedom
## Multiple R-squared:  0.7528, Adjusted R-squared:  0.7446
## F-statistic: 91.38 on 1 and 30 DF,  p-value: 1.294e-10
```

(l)-(ii) First run the regression of mpg on wt for automatic transition cars.

```r
at <- mtcars[mtcars$am == 0,]
m <- mtcars[mtcars$am == 1,]
lm3 <- lm(at$mpg~at$wt)
summary(lm3)
```

```
##
## Call:
## lm(formula = at$mpg ~ at$wt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6004 -1.5227 -0.2168  1.4816  5.0610
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  31.4161     2.9467  10.661 6.01e-09 ***
## at$wt        -3.7859     0.7666  -4.939 0.000125 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.528 on 17 degrees of freedom
## Multiple R-squared:  0.5893, Adjusted R-squared:  0.5651
## F-statistic: 24.39 on 1 and 17 DF,  p-value: 0.0001246
```

Then, run the regression of mpg on wt for manual cars.

```r
lm4 <- lm(m$mpg~m$wt)
summary(lm4)
```

```
##
## Call:
## lm(formula = m$mpg ~ m$wt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4190 -1.4937 -1.2234  0.8228  6.0909
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   46.294      3.120  14.839 1.28e-08 ***
## m$wt          -9.084      1.257  -7.229 1.69e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 2.686 on 11 degrees of freedom
## Multiple R-squared:  0.8261, Adjusted R-squared:  0.8103
## F-statistic: 52.26 on 1 and 11 DF,  p-value: 1.688e-05
```

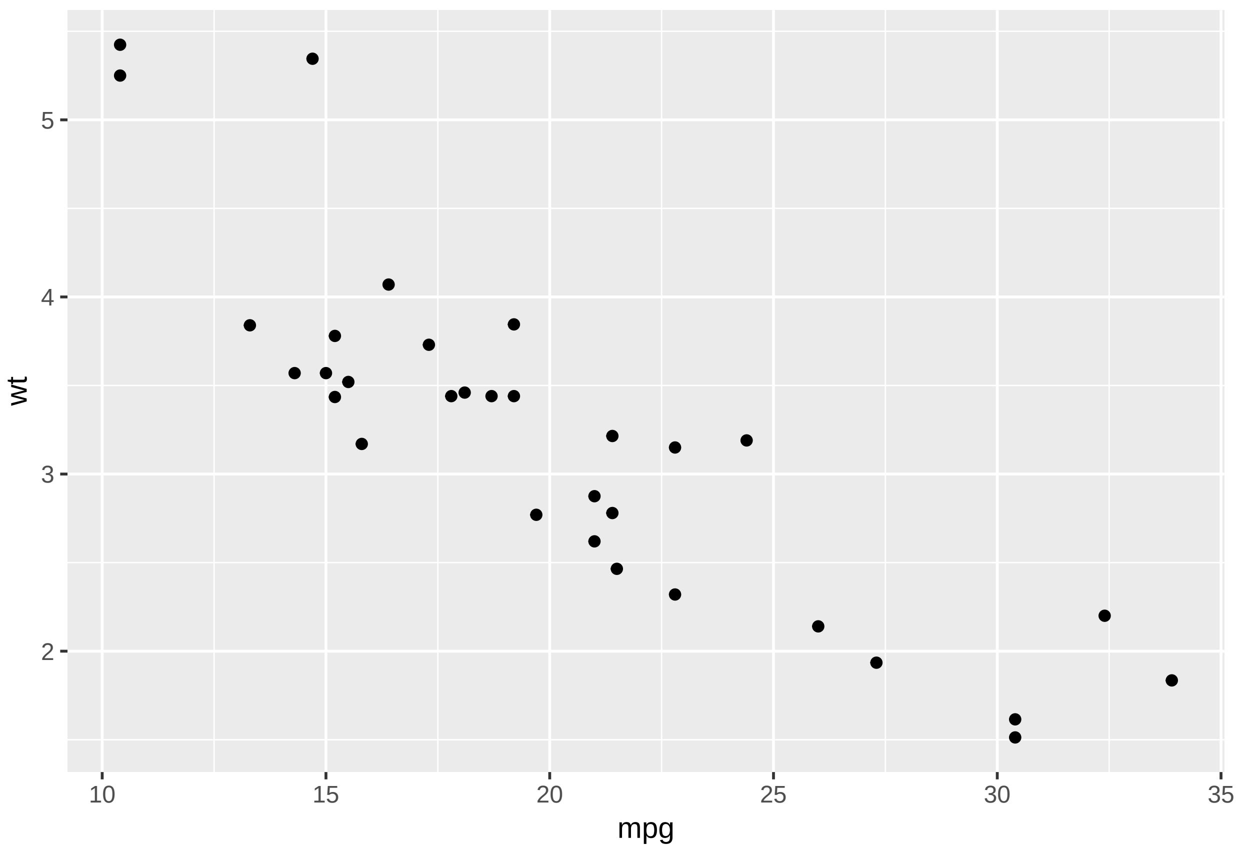
(l)-(iii)

```r
lm5 <- lm(mtcars$mpg ~ log(mtcars$hp))
summary(lm5)
```

```
##
## Call:
## lm(formula = mtcars$mpg ~ log(mtcars$hp))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.9427 -1.7053 -0.4931  1.7194  8.6460
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)      72.640      6.004  12.098 4.55e-13 ***
## log(mtcars$hp)  -10.764      1.224  -8.792 8.39e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.239 on 30 degrees of freedom
## Multiple R-squared:  0.7204, Adjusted R-squared:  0.7111
## F-statistic:  77.3 on 1 and 30 DF,  p-value: 8.387e-10
```

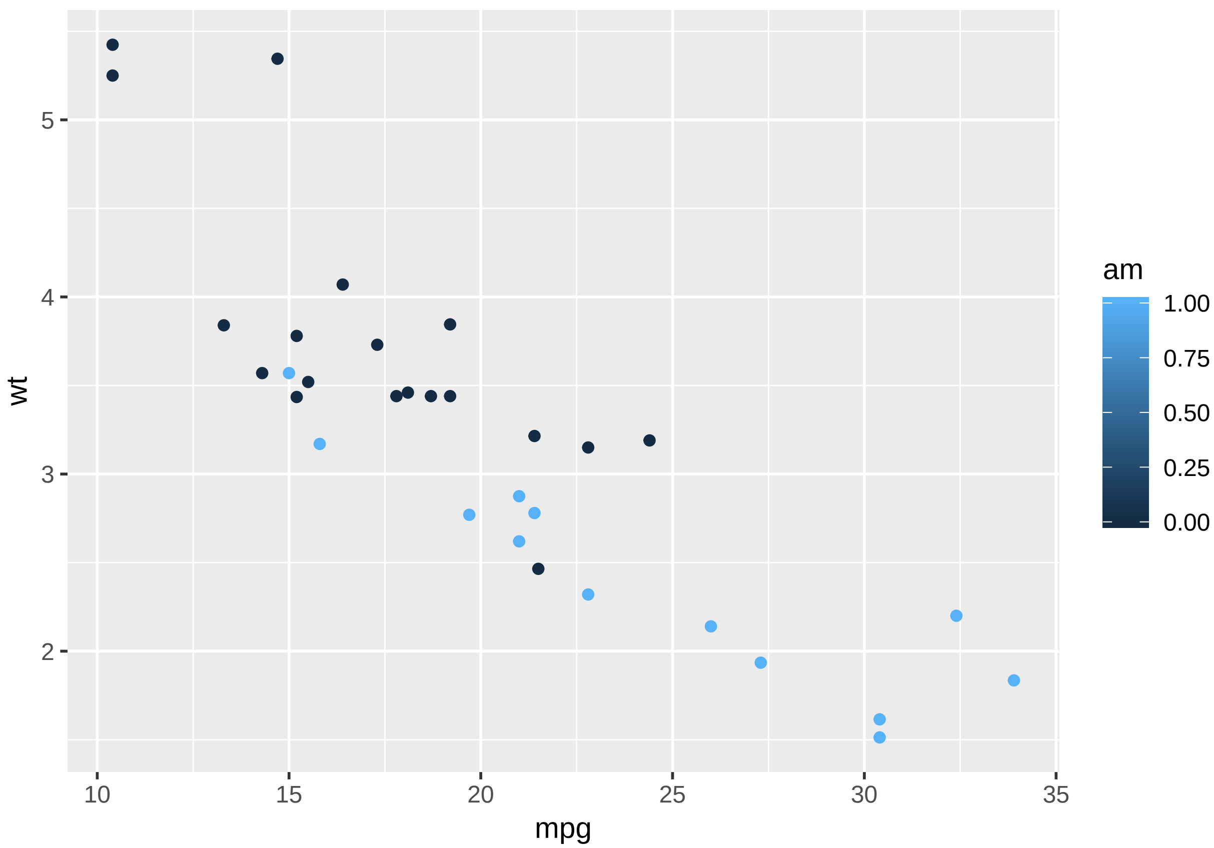
(m)-(i)

```r
mi <- ggplot(mtcars, aes(x=mpg, y=wt)) + geom_point()
show(mi)
```



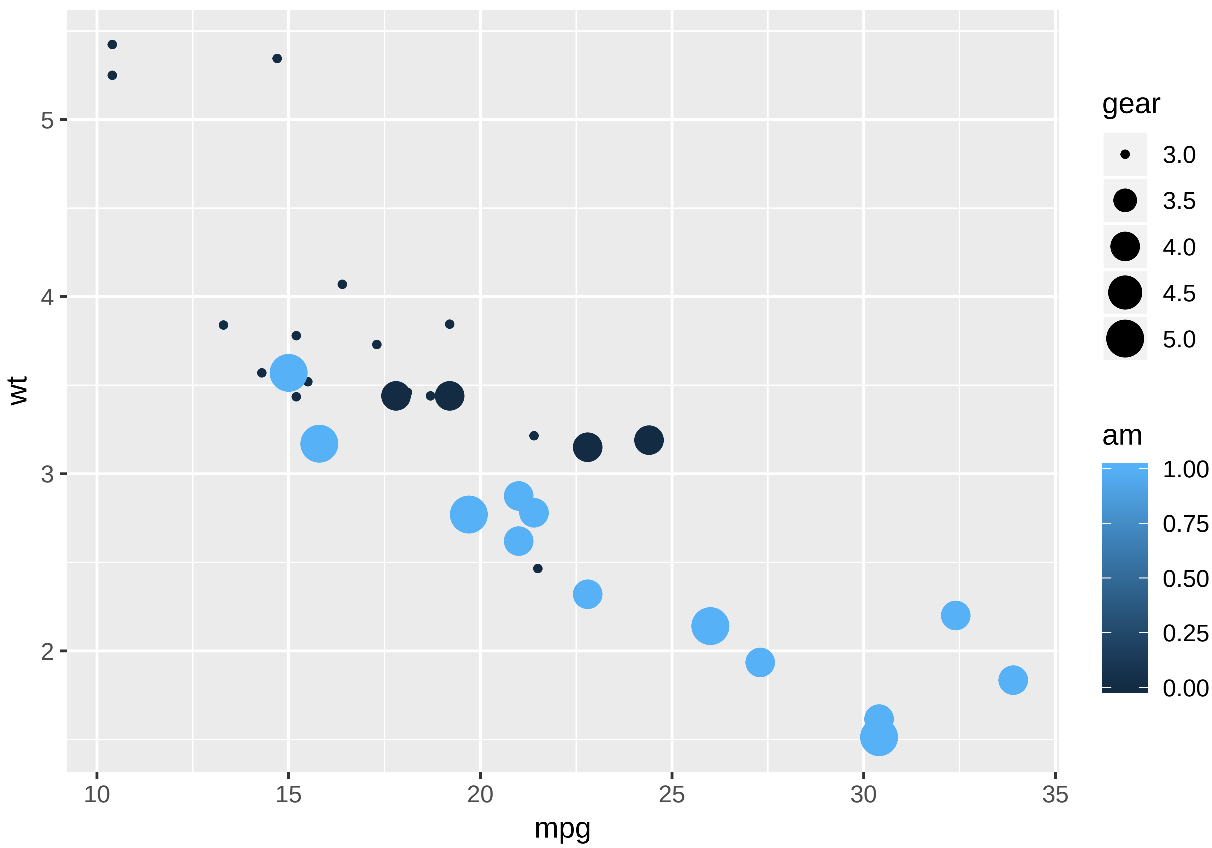
(m)-(ii)

```
ggplot(mtcars, aes(mpg, wt, color = am)) + geom_point()
```



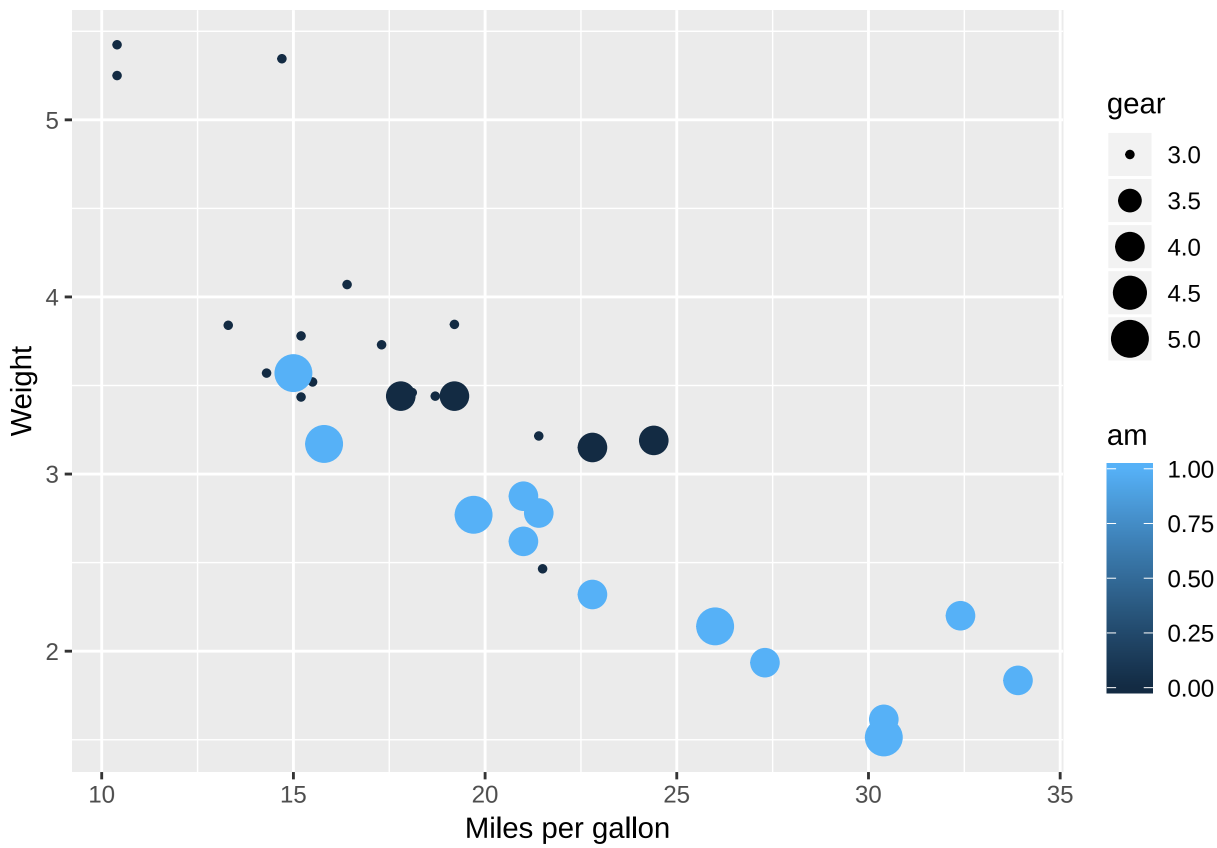
(m)-(iii)

```r
ggplot(mtcars, aes(mpg, wt, color = am, size = gear)) + geom_point()
```



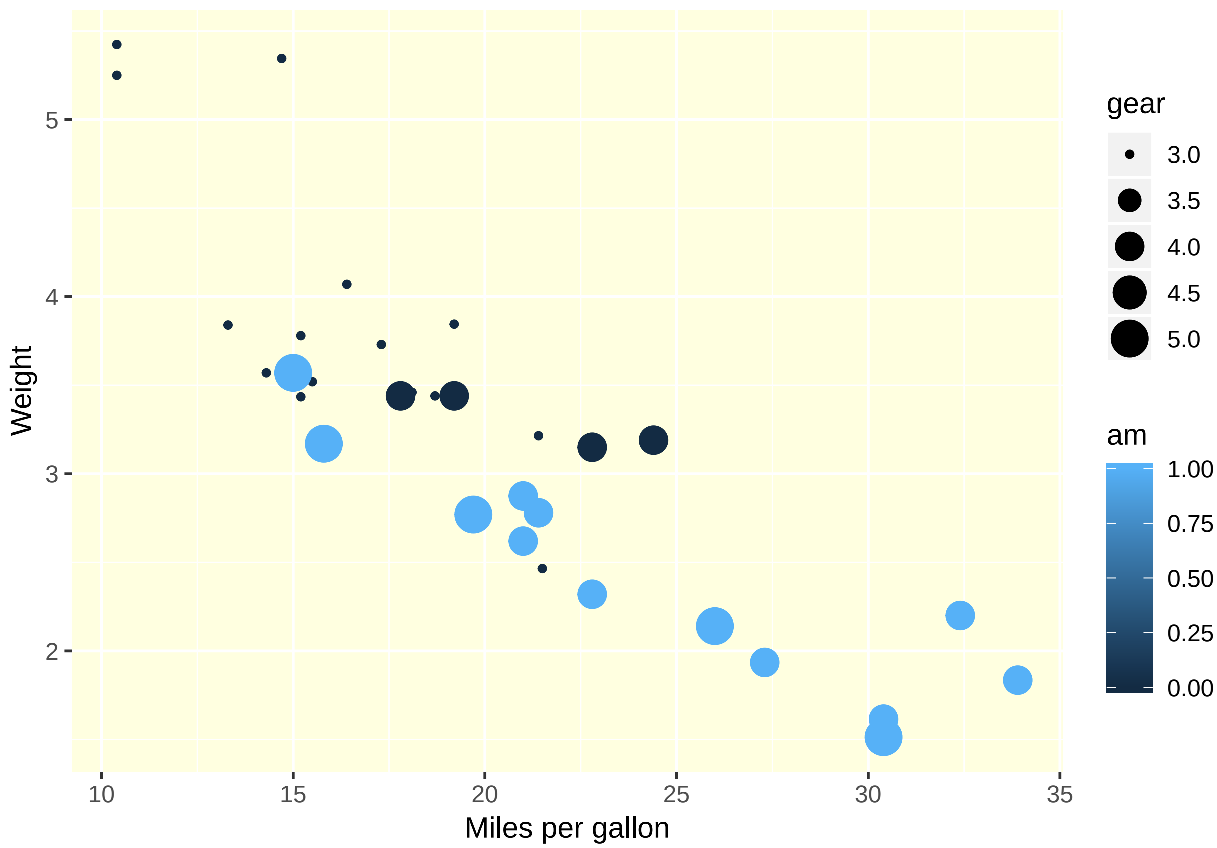
(m)-(iv)

```
ggplot(mtcars, aes(mpg, wt, color = am, size = gear)) + geom_point() + labs(x = "Miles per gallon", y =
```



(m) - (v)

```
ggplot(mtcars, aes(mpg, wt, color = am, size = gear)) + geom_point() + labs(x = "Miles per gallon", y =
```


Weight
5
4
3
2
10
15
20
25
30
35
Miles per gallon
gear
3.0
3.5
4.0
4.5
5.0
am
1.00
0.75
0.50
0.25
0.00