Homework 7

Steve Harms

October 25, 2017

Exercise 1

a)

```
#Read in the data
congressd <- data.frame(read.csv(file = "https://raw.githubusercontent.com/fivethir
tyeight/data/master/congress-age/congress-terms.csv",
                        header = T, sep = ","))
dim(congressd)
```

```
## [1] 18635    13
```

b)

```
#Summarize by age
congressd %>% summarize(mean(age))
```

```
##   mean(age)
## 1  53.31373
```

c)

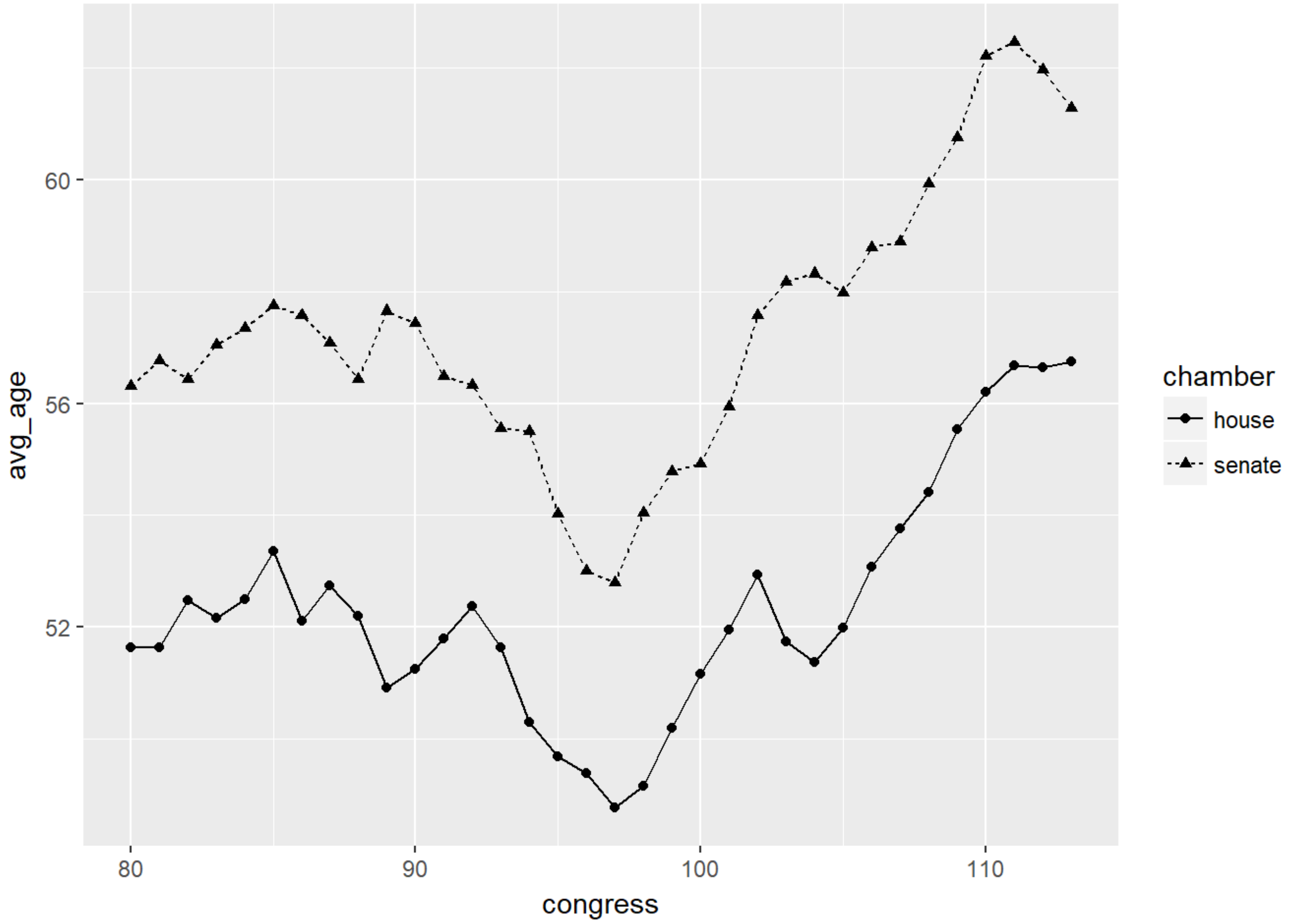
```
#group by congress and chamber, then summarize by average age again
bychamber <- congressd %>% group_by(congress, chamber) %>% summarize(avg_age = mean
(age)) %>%
  arrange(chamber) %>% select(chamber, congress, avg_age)

head(bychamber)
```

```
## Source: local data frame [6 x 3]
## Groups: congress [6]
##
##   chamber congress  avg_age
##    <fctr>    <int>    <dbl>
## 1   house       80 51.63620
## 2   house       81 51.63728
## 3   house       82 52.48489
## 4   house       83 52.16479
## 5   house       84 52.49615
## 6   house       85 53.34876
```

d)

d)

```
#plot the data, grouped by the factor chamber
dp <- ggplot(data = bychamber, aes(y= avg_age, x = congress, shape = chamber, line
type = chamber))
dpp <- dp + geom_point() + geom_line()
dpp
```



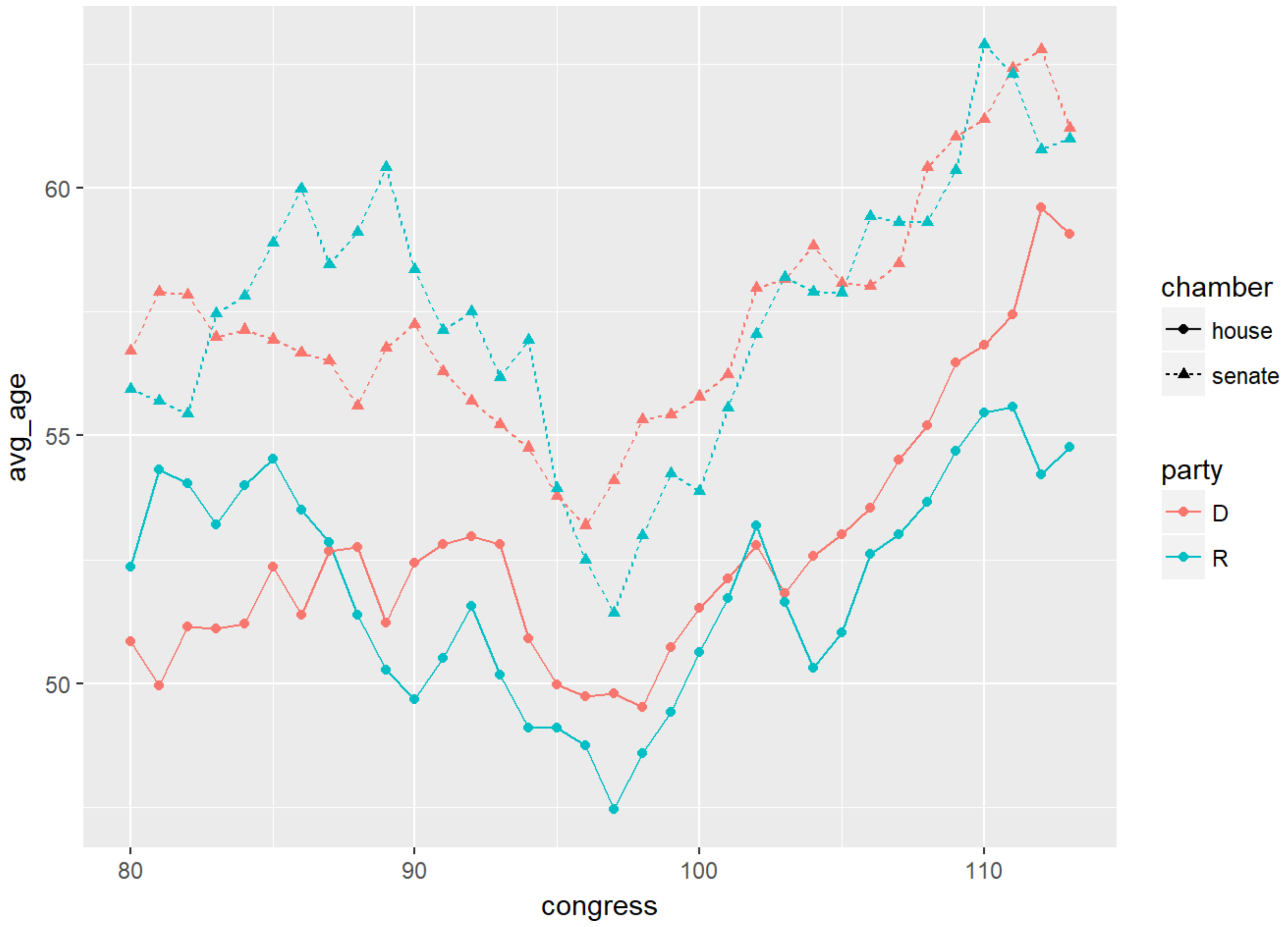
e)

```
#Group and summarize again, but this time filter out any not in republican or democ
ratic parties
#Then count our group sizes
partyaff <- congressd %>% group_by(congress, chamber, party) %>% filter(party == "R
" | party == "D") %>%
  summarize(avg_age = mean(age), n = length(party))%>% arrange(chamber, congress,
party) %>%
  select(chamber, congress, party, avg_age, n)

head(partyaff)
```

```
## Source: local data frame [6 x 5]
## Groups: congress, chamber [3]
##
##   chamber congress  party  avg_age     n
##    <fctr>    <int> <fctr>    <dbl> <int>
## 1   house       80      D 50.84091   198
## 2   house       80      R 52.34743   253
## 3   house       81      D 49.96245   269
## 4   house       81      R 54.31130   177
## 5   house       82      D 51.14876   242
## 6   house       82      R 54.03961   207
```

f)

```
#Plot the data again, this time with a color for each party
fp <- ggplot(data = partyaff, aes(y= avg_age, x = congress, shape = chamber, linet
ype = chamber, col = party))
fpp <- fp + geom_line() + geom_point()
fpp
```

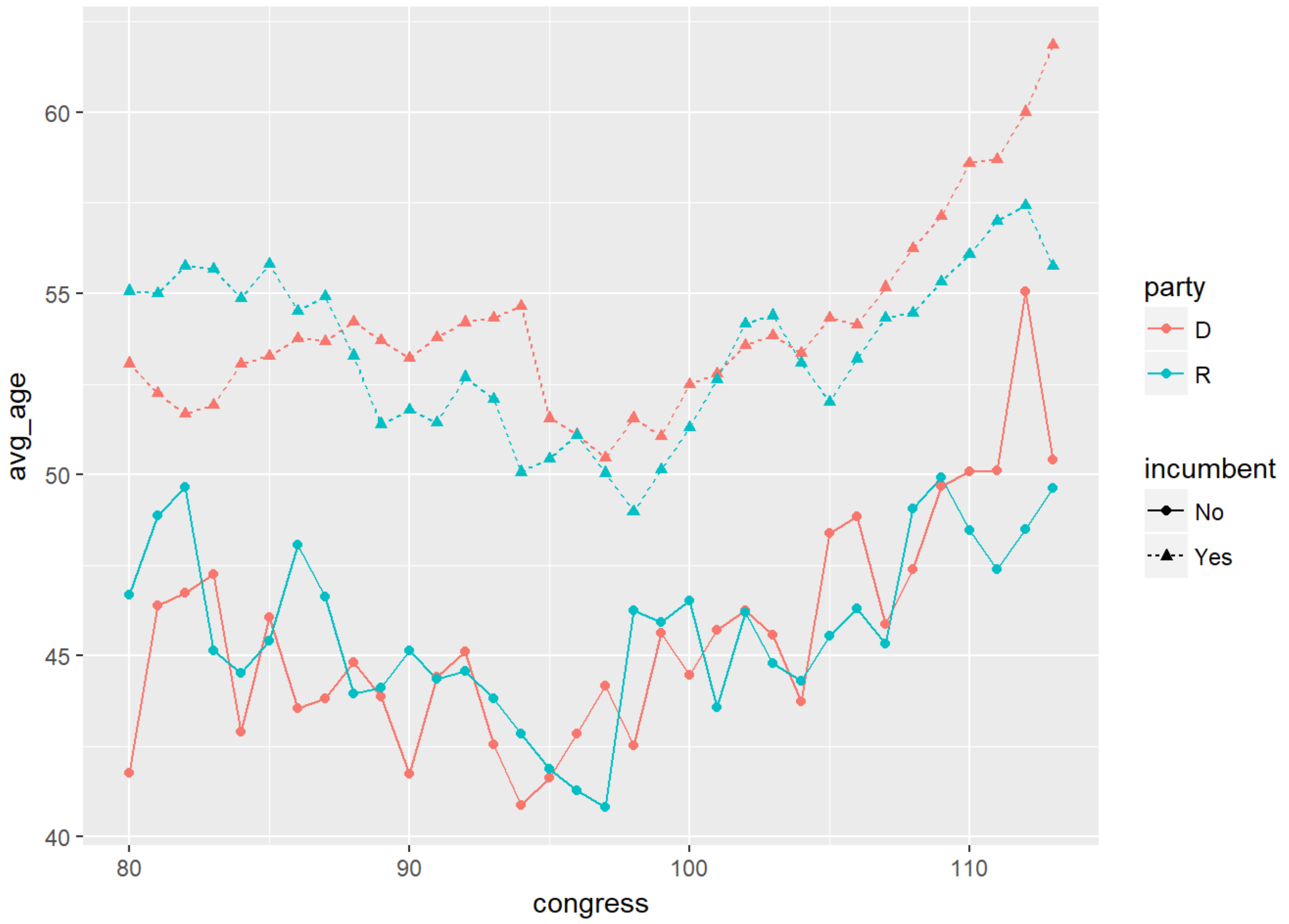


g)

```
#Group and summarize as above, but also include incumbent this time
#Filter so we only have the house and not the senate included
byinc <- congressd %>% group_by(congress, chamber, party, incumbent) %>% filter(par
ty == "R" | party == "D") %>%
  summarize(avg_age = mean(age), n = length(party))%>% arrange(chamber, congress,
party, incumbent) %>%
  select(chamber, congress, party, incumbent, avg_age, n) %>% filter(chamber == "h
ouse")

head(byinc)
```

```
## Source: local data frame [6 x 6]
## Groups: congress, chamber, party [3]
##
##   chamber congress  party incumbent  avg_age     n
##    <fctr>    <int> <fctr>    <fctr>    <dbl> <int>
## 1   house       80      D        No 41.76154    39
## 2   house       80      D       Yes 53.06792   159
## 3   house       80      R        No 46.67683    82
## 4   house       80      R       Yes 55.06667   171
## 5   house       81      D        No 46.38476   105
## 6   house       81      D       Yes 52.25305   164
```

```
#Now plot the data as above, separated by incumbent and party as factors
gp <- ggplot(data = byinc, aes(y= avg_age, x = congress, shape = incumbent, linety
pe = incumbent, col = party))
gpp <- gp + geom_line() + geom_point()
gpp
```



Exercise 2

a)

```
#split the data into a list of 3 frames based on species
iris.split <- split(iris, iris$Species)
#Check some results to make sure it worked
names(iris.split)
```

```
## [1] "setosa"     "versicolor" "virginica"
```

```
dim(iris.split$setosa)
```

```
## [1] 50  5
```

b)

```
#create a linear model for each of the 3 data frames (species)
mod <- iris.split %>% map(~ lm(Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width, data = .x))
#check one of the resulting list elements
mod$versicolor
```

```
##
## Call:
## lm(formula = Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width,
##     data = .x)
##
## Coefficients:
##  (Intercept)   Sepal.Width  Petal.Length   Petal.Width
##       1.8955        0.3869        0.9083       -0.6792
```

c)

```
#Get the summaries of the linear models from above
models <- mod %>% map(summary)
#check one of the resulting list elements
models$virginica
```

```
##
## Call:
## lm(formula = Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width,
##     data = .x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7388 -0.2183  0.0148  0.2206  0.7443
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.69988    0.53360   1.312   0.1962
## Sepal.Width   0.33034    0.17433   1.895   0.0644 .
## Petal.Length  0.94554    0.09072  10.422 1.07e-13 ***
## Petal.Width  -0.16975    0.19807  -0.857   0.3959
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.318 on 46 degrees of freedom
## Multiple R-squared:  0.7652, Adjusted R-squared:  0.7499
## F-statistic: 49.98 on 3 and 46 DF,  p-value: 1.622e-14
```

d)

```
#get coefficients from each of the 3 models
coeff <- models %>% map(coef)
coeff
```

```
## $setosa
##               Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)  2.3518898 0.39286751 5.9864707 3.034183e-07
## Sepal.Width  0.6548350 0.09244742 7.0833236 6.834434e-09
## Petal.Length 0.2375602 0.20801921 1.1420107 2.593594e-01
## Petal.Width  0.2521257 0.34686362 0.7268727 4.709870e-01
##
## $versicolor
##                Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)   1.8955395  0.5070552  3.738329 5.112246e-04
## Sepal.Width   0.3868576  0.2045449  1.891309 6.488965e-02
## Petal.Length  0.9083370  0.1654325  5.490681 1.666695e-06
## Petal.Width  -0.6792238  0.4353821 -1.560064 1.255990e-01
##
## $virginica
##                Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)   0.6998830 0.53360089  1.3116227 1.961563e-01
## Sepal.Width   0.3303370 0.17432873  1.8949086 6.439972e-02
## Petal.Length  0.9455356 0.09072204 10.4223360 1.074269e-13
## Petal.Width  -0.1697527 0.19807243 -0.8570233 3.958750e-01
```

e)

```
#Combine the 3 models' coefficients into a single data frame
newdf <- coeff %>% map_df( ~as.data.frame(.x), .id = 'Species')
newdf
```

```
##       Species   Estimate Std. Error    t value     Pr(>|t|)
## 1      setosa  2.3518898 0.39286751  5.9864707 3.034183e-07
## 2      setosa  0.6548350 0.09244742  7.0833236 6.834434e-09
## 3      setosa  0.2375602 0.20801921  1.1420107 2.593594e-01
## 4      setosa  0.2521257 0.34686362  0.7268727 4.709870e-01
## 5  versicolor  1.8955395 0.50705524  3.7383295 5.112246e-04
## 6  versicolor  0.3868576 0.20454490  1.8913091 6.488965e-02
## 7  versicolor  0.9083370 0.16543248  5.4906811 1.666695e-06
## 8  versicolor -0.6792238 0.43538206 -1.5600639 1.255990e-01
## 9   virginica  0.6998830 0.53360089  1.3116227 1.961563e-01
## 10  virginica  0.3303370 0.17432873  1.8949086 6.439972e-02
## 11  virginica  0.9455356 0.09072204 10.4223360 1.074269e-13
## 12  virginica -0.1697527 0.19807243 -0.8570233 3.958750e-01
```