

# Assignment 1

*Andreu Casas (TA)*

## Problem 1

Loading packages:

```
library(dplyr)
library(ggplot2)
library(tidyr)
library(broom)
library(knitr)
```

A

```
democracy <- read.csv(file = "democracy.csv", stringsAsFactors = FALSE,
                      na.strings = ".")
```

B.

```
democracy_by_variable <-
  democracy %>%
  gather(variable, value, -COUNTRY, -CTYNAME, -REGION, -YEAR)

dem_summary_stats <-
  democracy_by_variable %>%
  group_by(variable) %>%
  summarise(min = min(value, na.rm = TRUE),
            mean = mean(value, na.rm = TRUE),
            sd = sd(value, na.rm = TRUE),
            max = max(value, na.rm = TRUE))
```

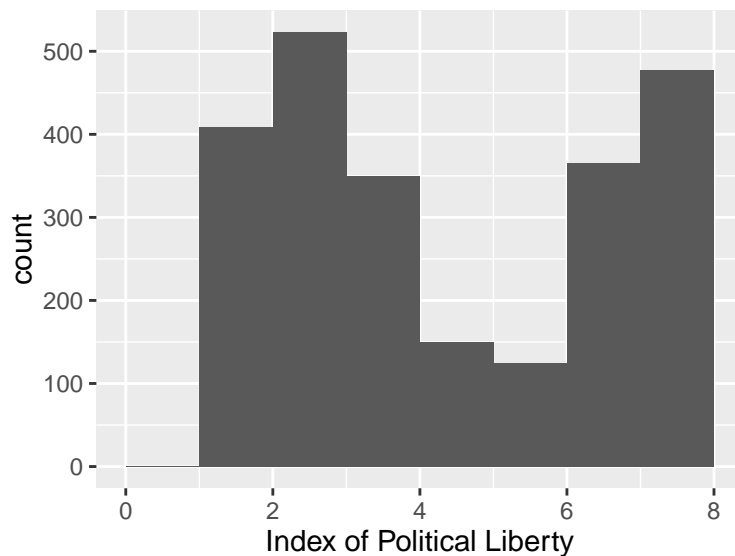
```
kable(dem_summary_stats)
```

variable	min	mean	sd	max
BRITCOL	0.00	0.2433349	0.4291476	1.00
CATH	0.00	37.1991275	38.2807940	99.00
CIVLIB	1.00	4.0762818	1.9732405	7.00
EDT	0.03	4.8533925	3.1173053	12.81
ELF60	0.00	0.3994587	0.2965374	0.93
GDPW	480.00	8876.9592826	8016.9287248	37903.00
MOSLEM	0.00	19.7358943	34.0430019	99.90
NEWC	0.00	0.4561318	0.4981322	1.00
OIL	0.00	0.1000969	0.3001656	1.00
POLLIB	1.00	3.8595248	2.2326768	7.00
REG	0.00	0.3986912	0.4896883	1.00
STRA	0.00	0.3751818	0.6979135	5.00

D.

```
ggplot(democracy, aes(x = POLLIB)) +  
  geom_histogram(binwidth = 1) +  
  xlab("Index of Political Liberty")
```

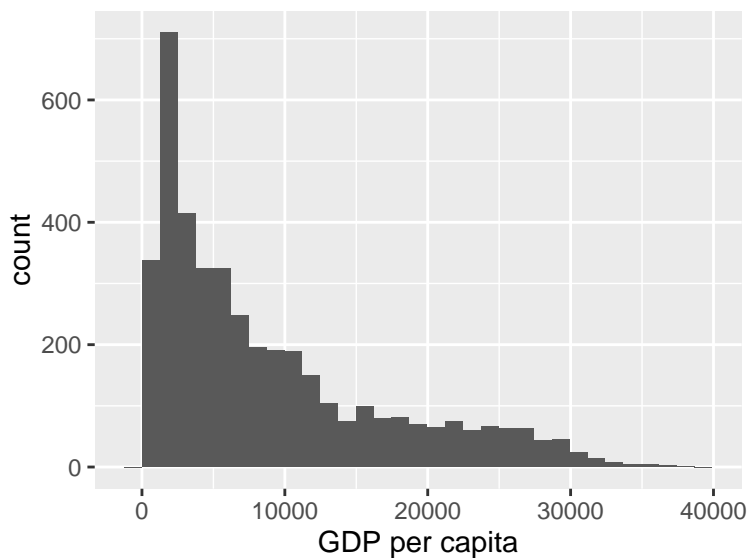
## Warning: Removed 1727 rows containing non-finite values (stat\_bin).



E.

```
ggplot(democracy, aes(x = GDPW)) +  
  geom_histogram() +  
  xlab("GDP per capita")
```

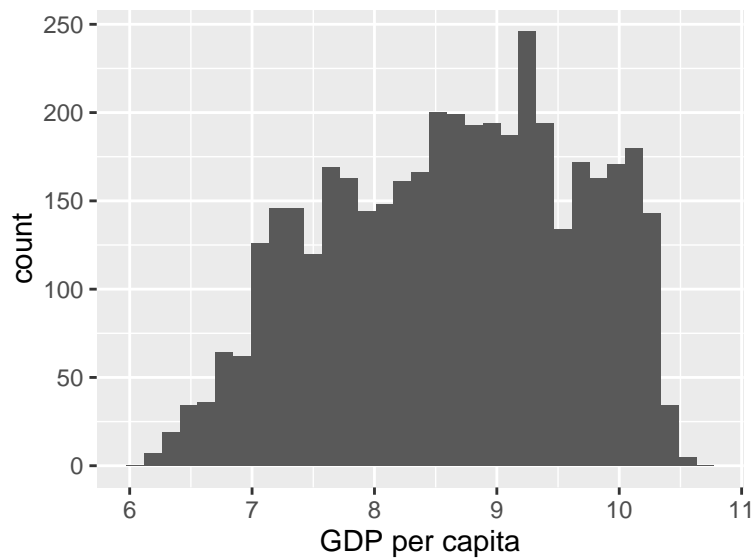
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



F.

```
ggplot(democracy, aes(x = log(GDPW))) +  
  geom_histogram() +  
  xlab("GDP per capita")
```

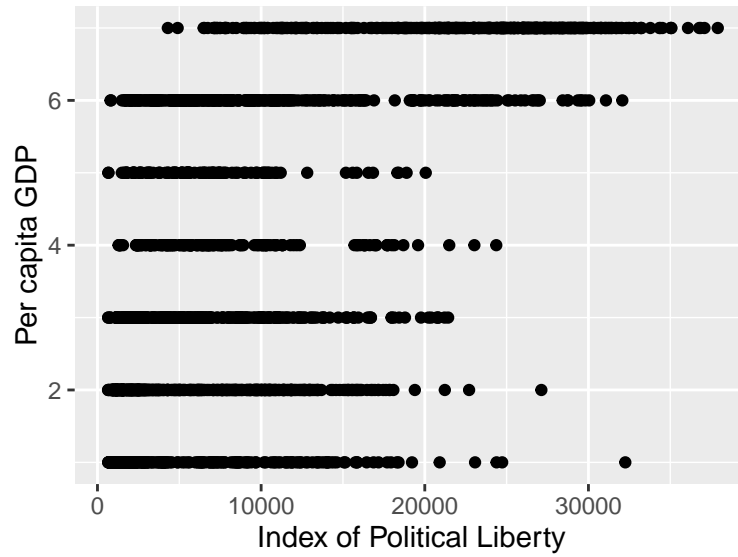
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



G.

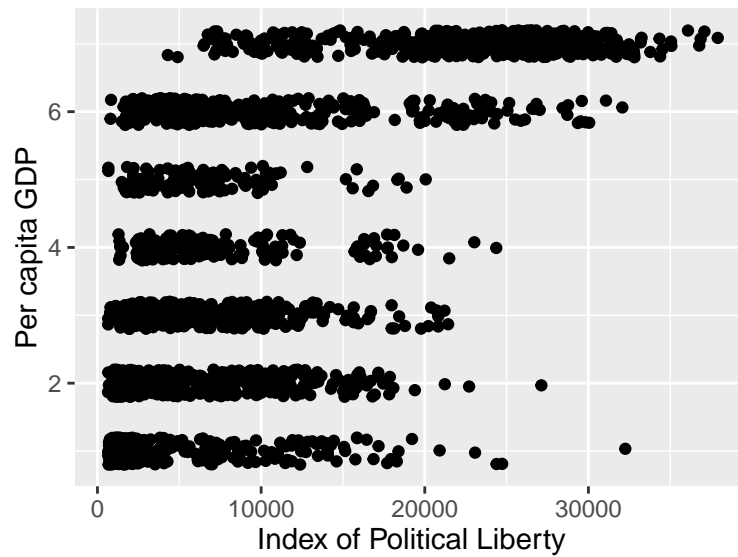
```
ggplot(democracy, aes(x = GDPW, y = POLLIB)) +  
  geom_point() +  
  ylab("Per capita GDP") +  
  xlab("Index of Political Liberty")
```

```
## Warning: Removed 1727 rows containing missing values (geom_point).
```



```
ggplot(democracy, aes(x = GDPW, y = POLLIB)) +
  geom_jitter(height = 0.5) +
  scale_y_continuous("Per capita GDP") +
  scale_x_continuous("Index of Political Liberty")
```

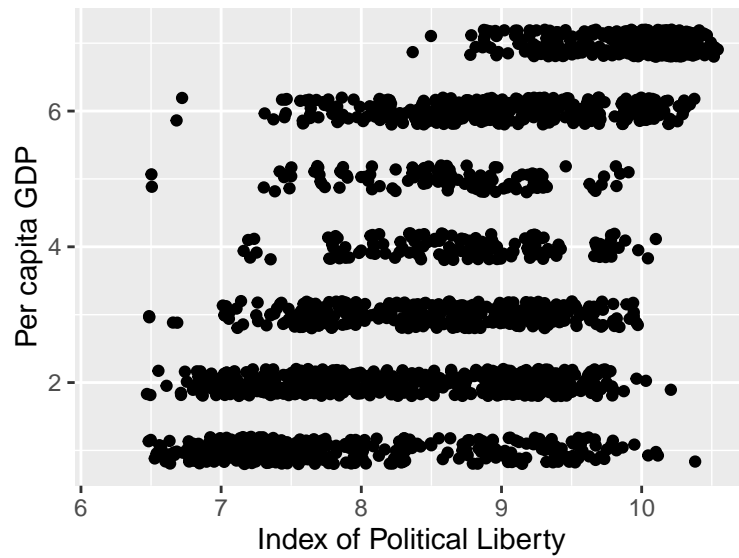
## Warning: Removed 1727 rows containing missing values (geom\_point).



I.

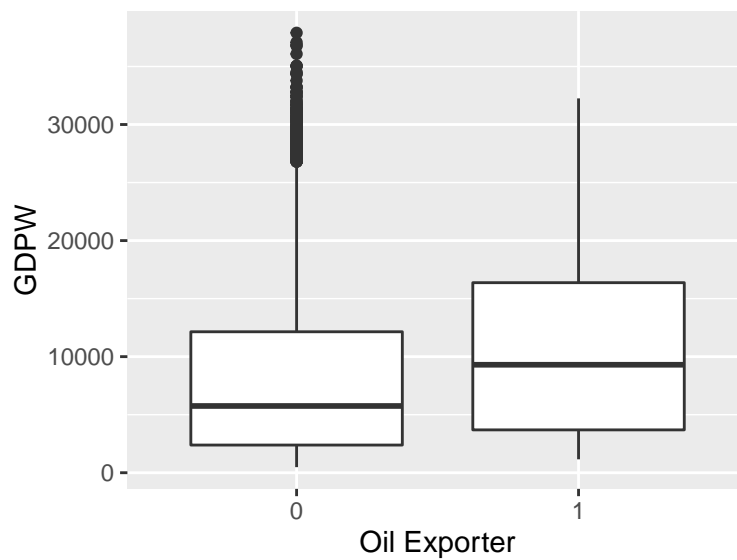
```
ggplot(democracy, aes(x = log(GDPW), y = POLLIB)) +
  geom_jitter(height = 0.5) +
  scale_y_continuous("Per capita GDP") +
  scale_x_continuous("Index of Political Liberty")
```

```
## Warning: Removed 1727 rows containing missing values (geom_point).
```



J.

```
ggplot(democracy, aes(x = factor(OIL), y = GDPW)) +  
  geom_boxplot() +  
  scale_x_discrete("Oil Exporter")
```



K.

```
catholic_gdpw <- filter(democracy, CATH > 40)$GDPW %>% mean(na.rm = TRUE)  
catholic_gdpw
```

```
## [1] 10295.13
```

```
all_gdpw <- mean(democracy$GDPW, na.rm = FALSE)
all_gdpw
```

```
## [1] 8876.959
```

```
catholic_gdpw / all_gdpw
```

```
## [1] 1.159758
```

L.

```
elf_summary <- democracy %>%
  mutate(high_elf60 = ELF60 > 0.6) %>%
  group_by(high_elf60) %>%
  summarise(gdpw_mean = mean(GDPW))
kable(elf_summary)
```

high_elf60	gdpw_mean
FALSE	11803.780
TRUE	3590.939
NA	7767.245

M.

```
ed_year <- democracy %>%
  group_by(YEAR) %>%
  summarize(ed_mean = median(EDT, na.rm = TRUE))
```

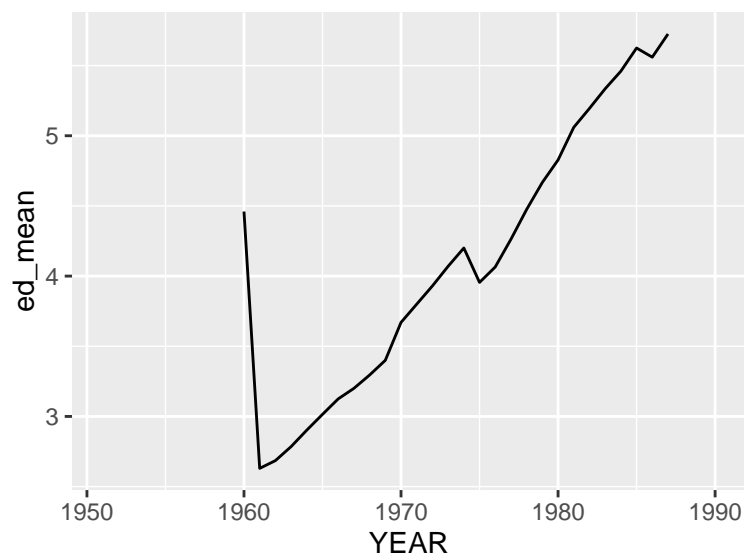
```
kable(ed_year)
```

YEAR	ed_mean
1951	NA
1952	NA
1953	NA
1954	NA
1955	NA
1956	NA
1957	NA
1958	NA
1959	NA
1960	4.4600
1961	2.6300
1962	2.6850

YEAR	ed_mean
1963	2.7850
1964	2.9025
1965	3.0150
1966	3.1250
1967	3.2000
1968	3.2950
1969	3.4000
1970	3.6700
1971	3.8000
1972	3.9300
1973	4.0700
1974	4.2000
1975	3.9550
1976	4.0650
1977	4.2625
1978	4.4750
1979	4.6675
1980	4.8275
1981	5.0600
1982	5.1950
1983	5.3350
1984	5.4600
1985	5.6250
1986	5.5600
1987	5.7250
1988	NA
1989	NA
1990	NA

```
ggplot(ed_year, aes(x = YEAR, y = ed_mean)) +  
  geom_line()
```

```
## Warning: Removed 12 rows containing missing values (geom_path).
```

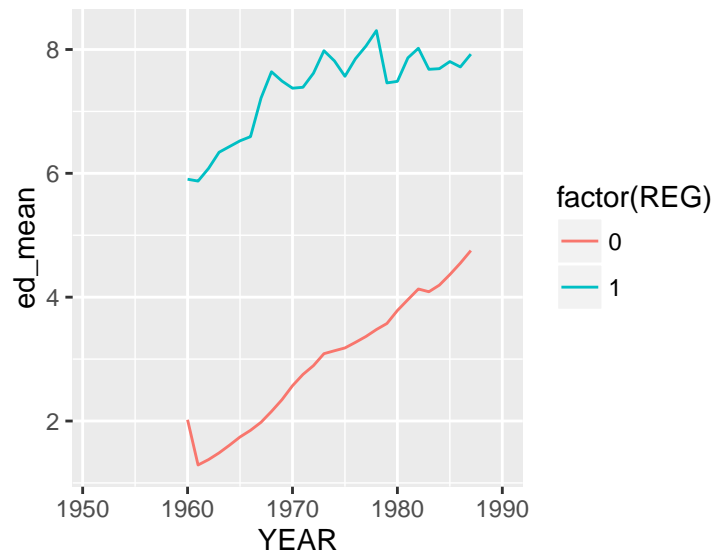


O.

```
ed_year_dem <- democracy %>%  
  group_by(YEAR, REG) %>%  
  summarize(ed_mean = median(EDT, na.rm = TRUE))
```

```
ggplot(ed_year_dem, aes(x = YEAR, y = ed_mean, col = factor(REG))) +  
  geom_line()
```

## Warning: Removed 24 rows containing missing values (geom\_path).



N.

```
democracy %>%  
  filter(YEAR == 1985, ! is.na(EDT)) %>%  
  mutate(med_edt_diff = abs(EDT - median(EDT))) %>%  
  filter(med_edt_diff == min(med_edt_diff)) %>%  
  select(CTYNAME, EDT)
```

```
##      CTYNAME  EDT  
## 1 Venezuela 5.625
```

Q.

```
dem_elf_tabl <-  
democracy %>%  
  filter(! is.na(ELF60)) %>%  
  mutate(`Country Type` = ifelse(as.logical(NEW60), "new", "old")) %>%  
  group_by(`Country Type`) %>%
```



```
summarise(elf60_p25 = quantile(ELF60, probs = 0.25),
          elf60_p75 = quantile(ELF60, probs = 0.75))
kable(dem_elf_tab1)
```

Country Type	elf60_p25	elf60_p75
new	0.42	0.75
old	0.06	0.44

## Problem 2

```
data("anscombe")
anscombe2 <- anscombe %>%
  mutate(obs = row_number()) %>%
  gather(variable_dataset, value, - obs) %>%
  separate(variable_dataset, c("variable", "dataset"), sep = 1L) %>%
  spread(variable, value) %>%
  arrange(dataset, obs)
```

A.

```
results1 <- anscombe2 %>%
  group_by(dataset) %>%
  summarise(mean_x = mean(x, na.rm = TRUE),
            sd_x = sd(x, na.rm = TRUE),
            mean_y = mean(y, na.rm = TRUE),
            sd_y = sd(y, na.rm = TRUE),
            cor_xy = cor(x,y))
results2 <- anscombe2 %>%
  group_by(dataset) %>%
  do(tidy(lm(y ~ x, data = .))) %>%
  filter(term == "x") %>%
  dplyr::select(estimate, std.error)
all_results <- left_join(results1, results2)
```

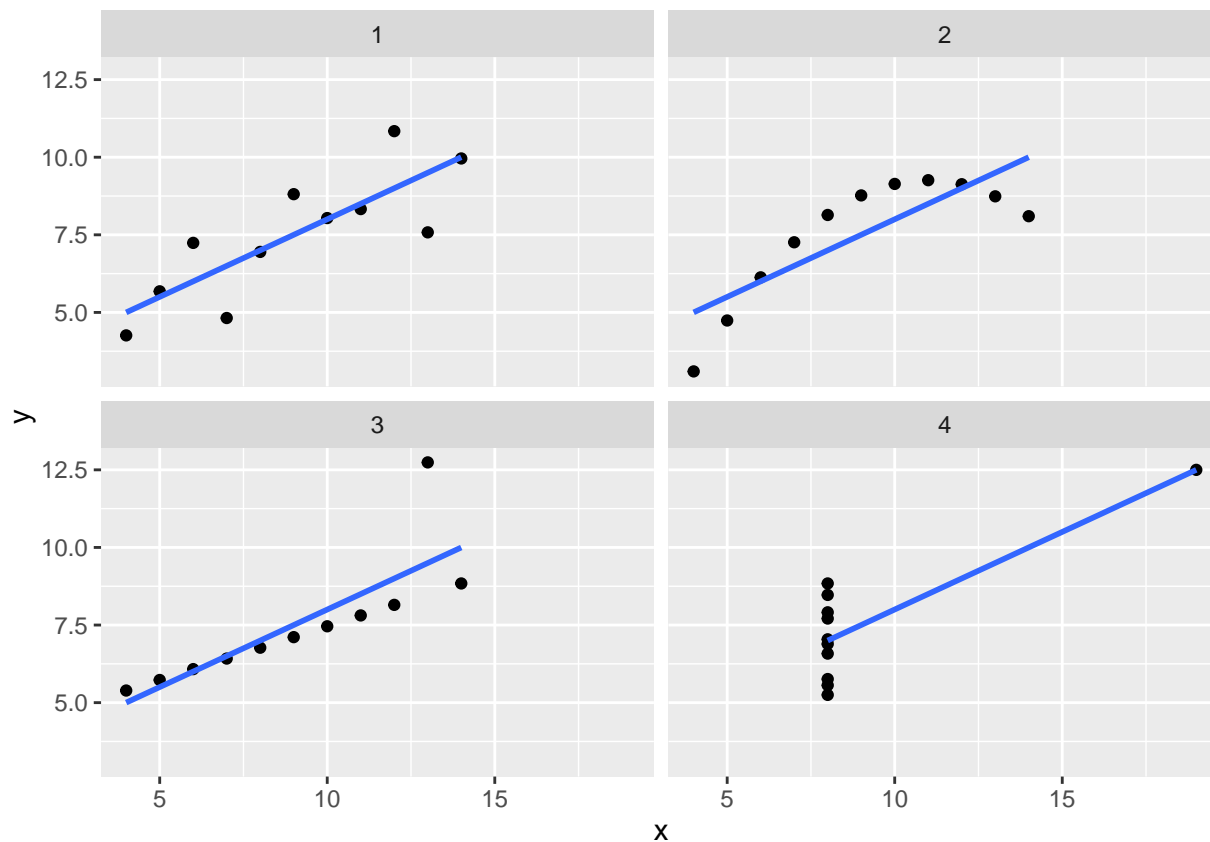
## Joining by: "dataset"

```
kable(all_results)
```

dataset	mean_x	sd_x	mean_y	sd_y	cor_xy	estimate	std.error
1	9	3.316625	7.500909	2.031568	0.8164205	0.5000909	0.1179055
2	9	3.316625	7.500909	2.031657	0.8162365	0.5000000	0.1179637
3	9	3.316625	7.500000	2.030424	0.8162867	0.4997273	0.1178777
4	9	3.316625	7.500909	2.030578	0.8165214	0.4999091	0.1178189

B.

```
ggplot(anscombe2, aes(x = x, y = y)) +  
  geom_point() +  
  geom_smooth(method = "lm", se = FALSE) +  
  facet_wrap(~ dataset)
```



### Problem 3

Load the data into R from the csv file:

```
sprinters <- read.csv("sprinters.csv")
```

A.

```
sprinters_orig <-  
  filter(sprinters,  
    year <= 2004,  
    olympics == 1)
```

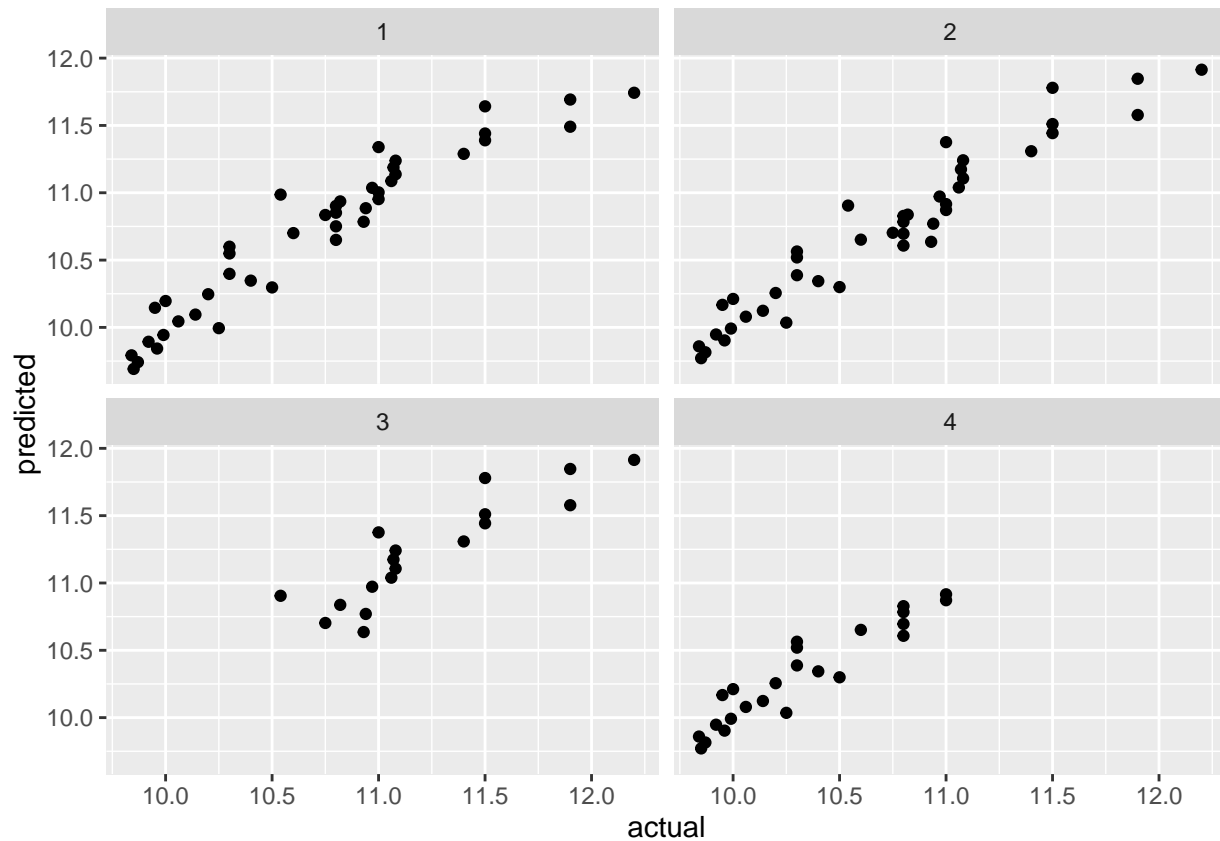
B.

```
library("dplyr")
mod1 <- lm(time ~ year + women, data = sprinters_orig)
mod2 <- lm(time ~ year * women, data = sprinters_orig)
mod3 <- lm(time ~ year, data = filter(sprinters_orig, women == 1))
mod4 <- lm(time ~ year, data = filter(sprinters_orig, women == 0))
models_list <- list(mod1, mod2, mod3, mod4)
```

C.

```
models_data <- NULL
for (i in 1:length(models_list)) {
  mod <- i
  model <- models_list[[i]]
  data <- models_list[[i]]$model
  augmented_data <- augment(model, data)
  actual_values <- augmented_data$time
  predicted_values <- augmented_data$.fitted
  res <- data.frame(actual = actual_values,
                    predicted = predicted_values,
                    model = mod)
  models_data <- rbind(models_data, res)
}
```

```
ggplot(models_data, aes(x = actual, y = predicted)) +
  geom_point() +
  facet_wrap(~ model)
```



```
models_data <- mutate(models_data, resid = actual - predicted)
models_data <- mutate(models_data, sq_resid = resid^2)
models_data %>%
  group_by(model) %>%
  summarise(rmse = sqrt(mean(sq_resid))) %>%
  filter(model == 2)
```

```
## Source: local data frame [1 x 2]
##
##   model      rmse
##   (int)    (dbl)
## 1      2 0.1624051
```

```
newdata <-
  filter(sprinters,
         year >= 2004)
models_data2 <- data.frame(
  actual = newdata$time,
  predicted = predict(mod2, newdata = newdata))
models_data2 <- mutate(models_data2, resid = actual - predicted)
models_data2 <- mutate(models_data2, sq_resid = resid^2)
models_data2 %>%
  summarise(rmse = sqrt(mean(sq_resid)))
```

```
##           rmse
## 1 0.2274526
```