Homework 1

Matthew Reyers, Dani Chu and Ryan Sheehan

2017-09-20

```
knitr::opts_chunk$set(echo = TRUE, warning = FALSE)
#install.packages("pacman")
pacman::p_load(knitr, tidyverse, ISLR, GGally)
data(Auto)
```

Question 1 (Chapter 2, #1, 8 marks)

(a) With a large number of data points there is less variance to account for between data sets. Because of the limited model variance we can focus instead on reducing the bias associated with each method. As flexible methods tend to produce less biased results, we would expect a flexible method to be more desirable in this circumstance.

(b) Having a small number of observations and a large number of predictors can often be problematic for highly flexible methods. Few observations indicate a great deal of variance in these methods when data sets are changed. For this reason it would make more sense to pursue a method that performs well in high model variance situations. This leads to the conclusion that a non-flexible method would be preferred.

(c) When the relationship between predictors and response is highly non-linear we expect a flexible model to outperform a non-flexible model. Reasoning for this is non-flexible models, such as regression or low order smoothing splines, tend to predict better when the underlying relationship is linear. In comparison, a high order smoothing spline (which is very flexible) can be shown to perfectly fit the data in certain conditions. Though this is not preferred for eventual testing, it provides some more intuition that a flexible model can be of more use in this scenario.

(d) When the variance of the error terms is extremely large, it is important to take an approach that is less flexible. If a flexible approach is instead used the model could end up fitting the errors instead of just fitting the data.

Question 2 (Chapter 2, #2, 6 marks)

(a) This would be a regression problem as CEO Salary is a continuous value. As we are concerned with understanding the current factors impacting current salary, a reasonable conclusion is that this is a concern of inference. There are 500 observations and 3 predictors.

(b) Here we are looking at an event that is yet to happen and are therefore looking to predict its result. As we can classify this result into either being a success or a failure, we recognize that this is a classification problem. There are 20 observations and 13 predictors.

(c) As with many market considerations, there is no money in the past and so we look to predicting the future. This problem will naturally take the form of a regression as we are looking to measure how much we can gain rather than if we will. The analysis will be conducted on 52 observations (number of weeks in 2012) and with 3 predictors.

Question 3 (Chapter 2, #9, 8 marks)

(a) Qualitative: Origin, name Quantitative: Cylinders, displacement, horsepower, weight, year, acceleration, mpg

(b) Note we made us of the inline R code functionality in R markdown to output these results.

The range of displacement is: 68, 455

The range of cylinders is: 3, 8

The range of horsepower is: 46, 230

The range of weight is: 1613, 5140

The range of year is: 70, 82

The range of acceleration is: 8, 24.8

The range of mpg is: 9, 46.6

(c)

```
# Function takes column as input and outputs mean and SD for the variable
meansd <- function(x) {
  out <- c(mean(x), sd(x))
  names(out) <- c("mean", "SD")
  out
}

mean_sd <- apply(Auto[, c("cylinders",
                          "displacement",
                          "horsepower",
                          "weight",
                          "acceleration",
                          "year",
                          "mpg")],
                 MARGIN = 2, FUN = meansd)
kable(mean_sd)
```

	cylinders	displacement	horsepower	weight	acceleration	year	mpg
mean	5.471939	194.412	104.46939	2977.5842	15.541327	75.979592	23.445918
SD	1.705783	104.644	38.49116	849.4026	2.758864	3.683737	7.805008

(d)

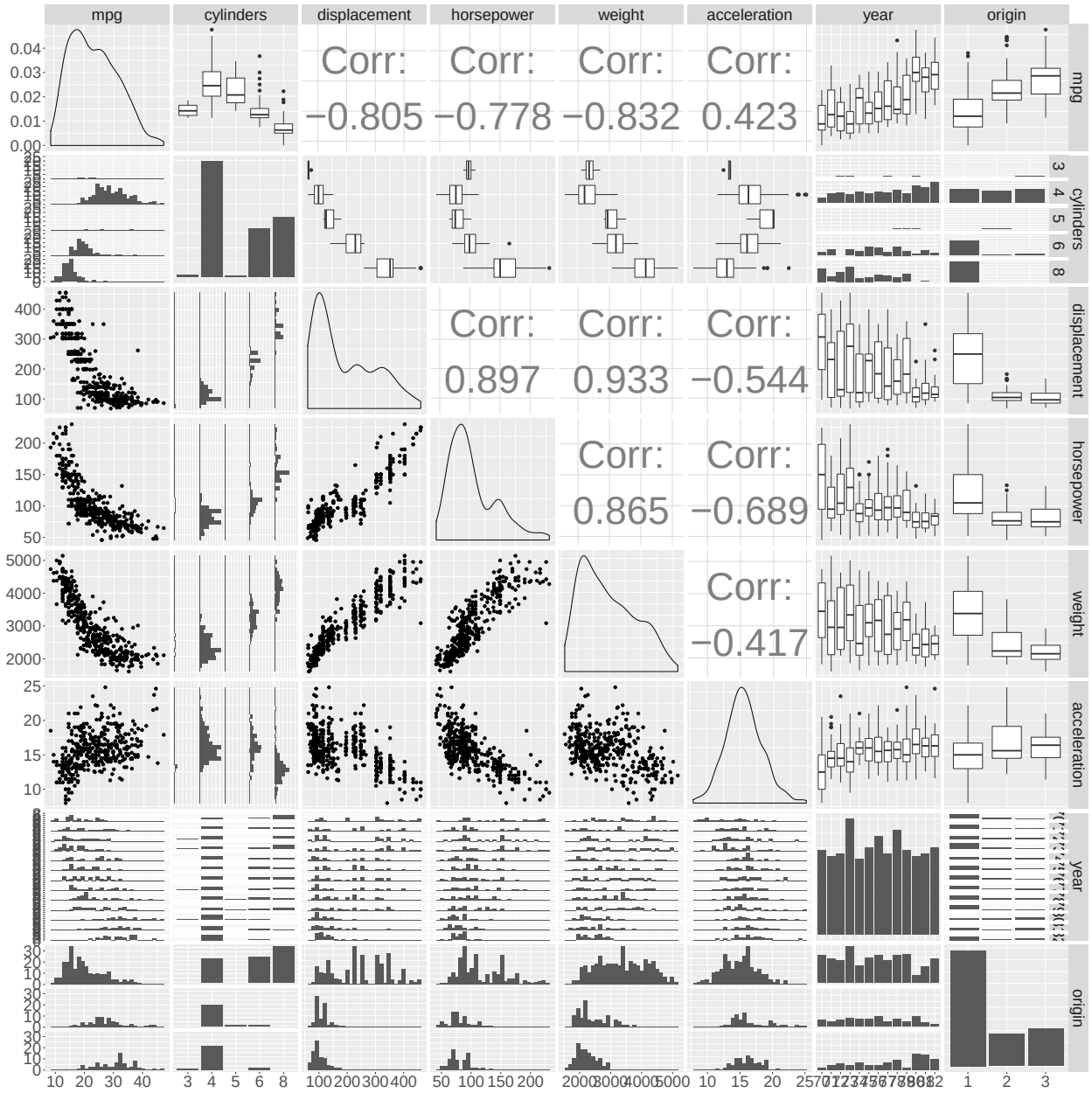
```
# Subset Data
AutoSubset <- Auto[c(1:9,86:nrow(Auto)),]

# Function that takes column as input and outputs the mean, sd, lower bound of range,
# and upper bound of range
meansdr = function(x) {
  out = c(mean(x), sd(x), range(x))
  names(out) = c("mean", "sd", "range lower", "range upper")
  out
}

subset <- apply(Auto[, c("cylinders",
                         "displacement",
```

```
                          "horsepower",
                          "weight",
                          "acceleration",
                          "year")],
                 MARGIN = 2, FUN = meansdr)
kable(subset)
```

	cylinders	displacement	horsepower	weight	acceleration	year
mean	5.471939	194.412	104.46939	2977.5842	15.541327	75.979592
sd	1.705783	104.644	38.49116	849.4026	2.758864	3.683737
range lower	3.000000	68.000	46.00000	1613.0000	8.000000	70.000000
range upper	8.000000	455.000	230.00000	5140.0000	24.800000	82.000000

(e)

```
# Use ggpairs from the GGally package
Auto %>%
  select(-name) %>%
  mutate(cylinders = cylinders %>% as.factor(),
         year = year %>% as.factor(),
         origin = origin %>% as.factor()) %>%
  ggpairs(., upper= list(continuous = wrap("cor", size = 18))) +
  theme(text = element_text(size = 26))
```



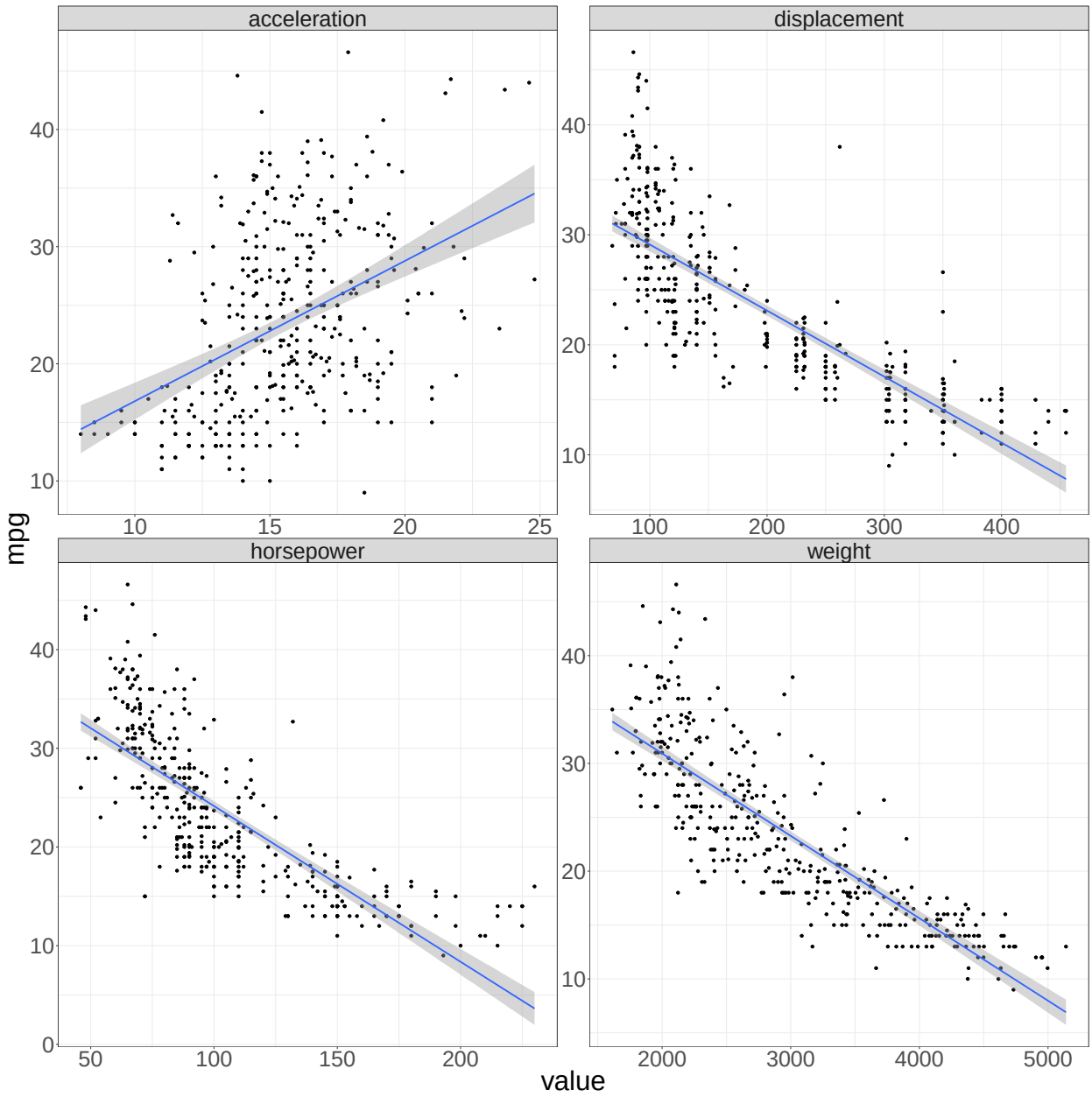
Note the plot is better viewed outside of the pdf using the zoom function in RStudio.

Some variables, such as horsepower and weight, are highly correlated. Instances in which we have highly correlated variables are generally preferable to avoid. The only time we want correlation is between a regressor and a dependent variable.

(f) There appears to be a relationship between mpg and the following: displacement, horsepower, and weight. These relationships could be argued to be linear but may be better explained using a more flexible model. If this is unavailable then a transformation of the data may lead to more efficient modelling. Additionally, both cylinders and origin are categorical variables that seem to be related to mpg.

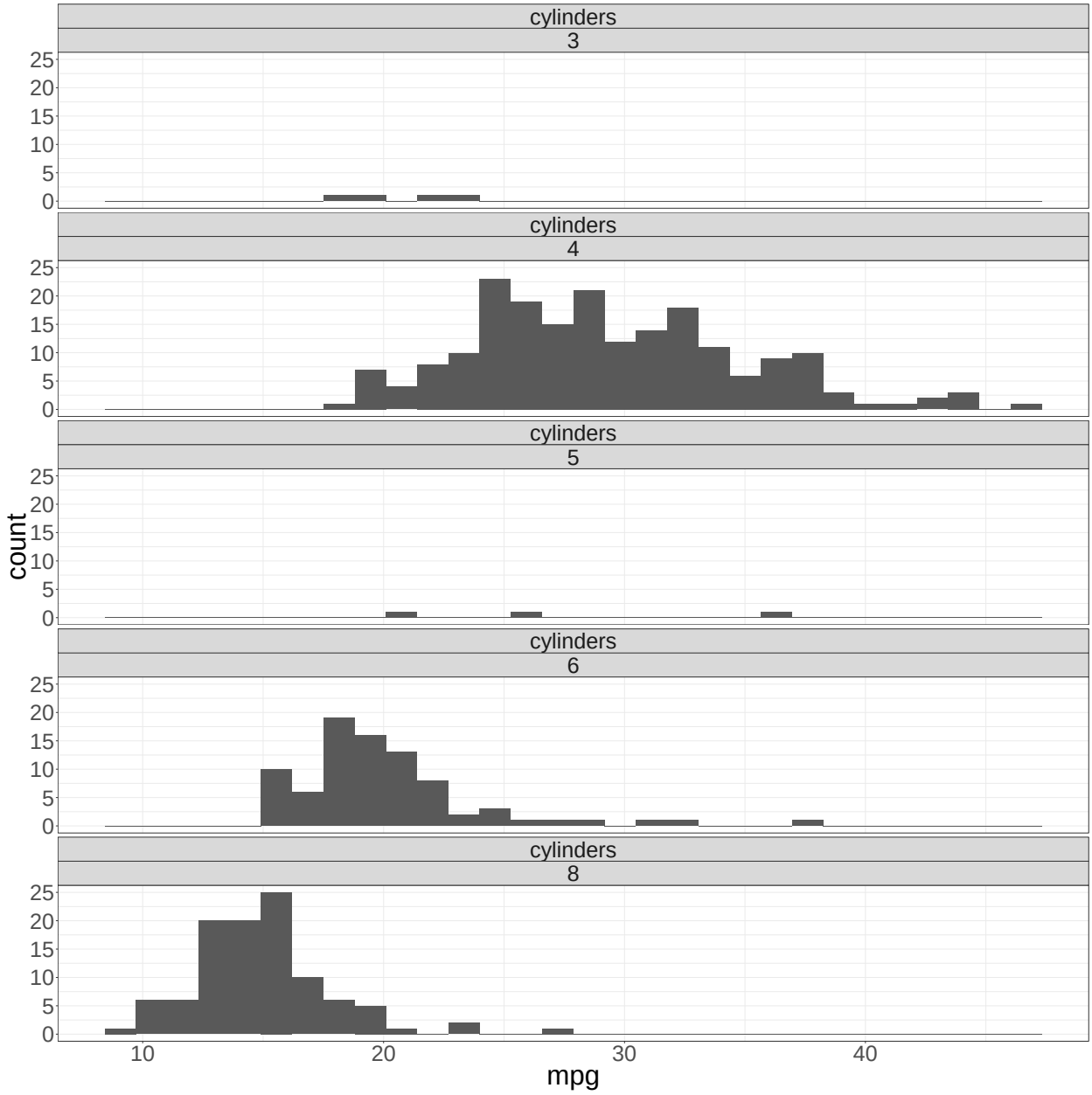
Below are graphs that depict the relationships of mpg with all the variables. The first graph is for all the continuous variables and their relationship with mpg. They are fit with a linear model.

```r
# Plot variables with mpg
Auto %>%
  select(-name, -cylinders, -origin, -year) %>%
  gather(-mpg, key = "var", value = "value") %>%
  ggplot(aes(x = value, y = mpg)) +
    geom_point() +
    stat_smooth(method = "lm") +
    facet_wrap(~ var, scales = "free") +
    theme_bw()+
  theme(text = element_text(size = 36))
```



The next graphs are for the categorical variables. Beginning with cylinders and mpg.

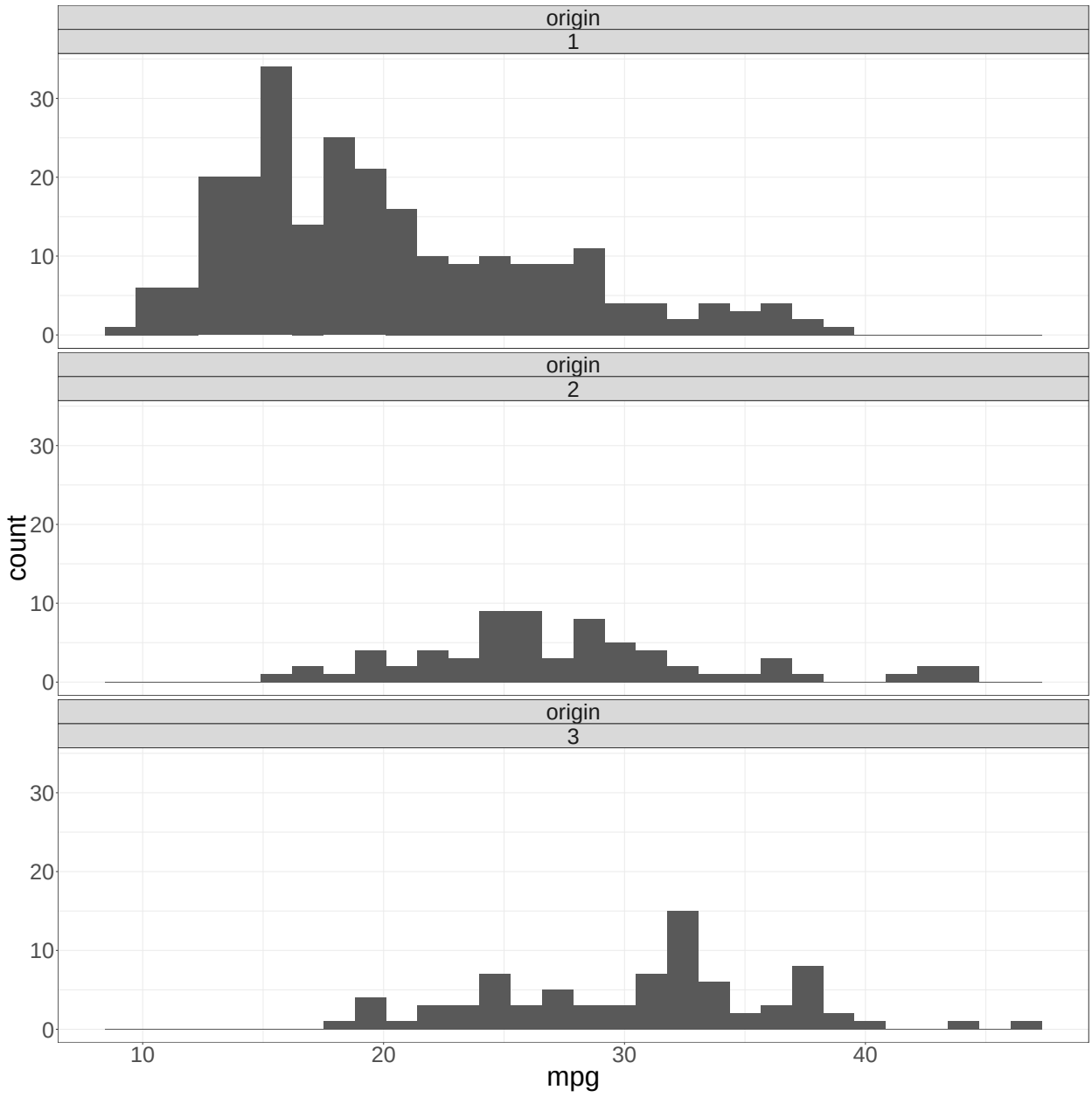
```
Auto %>%
  select(mpg, cylinders) %>%
  gather(-mpg, key = "var", value = "value") %>%
  ggplot(aes(x = mpg)) +
    geom_histogram() +
    facet_wrap(ncol=1, ~ var +value) +
    theme_bw()+
  theme(text = element_text(size = 36))
```



This next graph is origin and mpg.

```
Auto %>%
  select(mpg, origin) %>%
  gather(-mpg, key = "var", value = "value") %>%
```

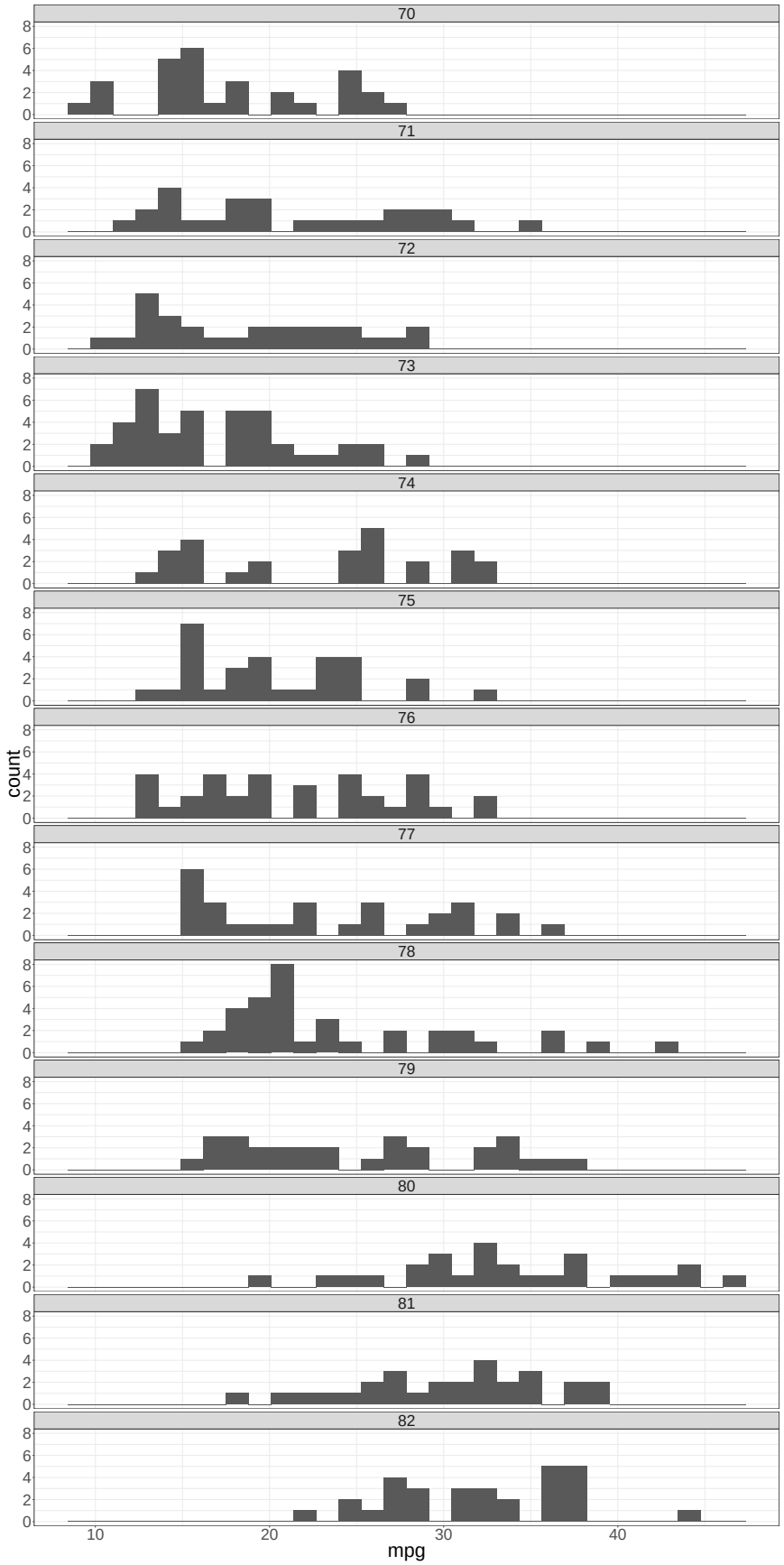
```
  ggplot(aes(x = mpg)) +
    geom_histogram() +
    facet_wrap(ncol=1, ~ var + value) +
    theme_bw()+
  theme(text = element_text(size = 36))
```



This final graph is year and mpg.

```
Auto %>%
  select(mpg, year) %>%
  gather(-mpg, key = "var", value = "value") %>%
  ggplot(aes(x = mpg)) +
    geom_histogram() +
    facet_wrap(ncol=1, ~ value) +
```

```
  theme_bw()+
theme(text = element_text(size = 36))
```


70
71
72
73
74
75
76
77
78
79
80
81
82
count
0
2
4
6
8
10
20
30
40
mpg