STATS 769 Lab 00

Chase Robertson

2022-07-20

```
cars <- read.csv("car-imports.csv")
head(cars)
```

```
##    Month Unit.Qty Country        vfd        cif Imports.Qty
## 1 201901      NMB Austria  2,511,892  2,574,059          42
## 2 201901      NMB Belgium  1,886,314  1,963,103          53
## 3 201901      NMB   China  2,675,272  2,799,368         121
## 4 201901      NMB Czechia  7,251,350  7,665,974         256
## 5 201901      NMB Germany 16,902,288 17,320,488         308
## 6 201901      NMB  France  7,755,592  8,188,301         238
```

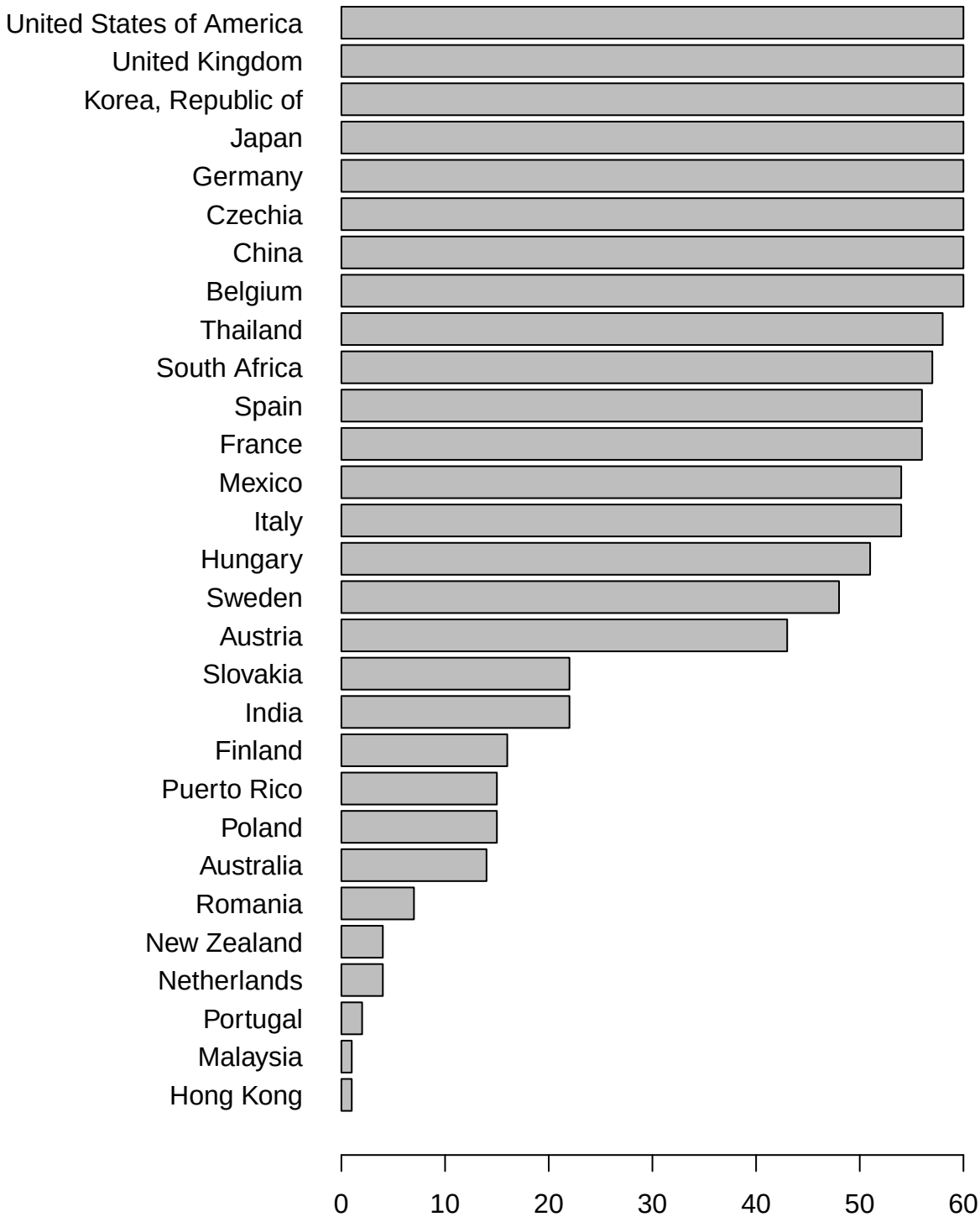
```
cars2 <- cars
cars2$vfd <- as.numeric(gsub(",", "", cars$vfd))
cars2$cif <- as.numeric(gsub(",", "", cars$cif))
cars2$Imports.Qty <- as.numeric(cars$Imports.Qty)
```

```
## Warning: NAs introduced by coercion
```

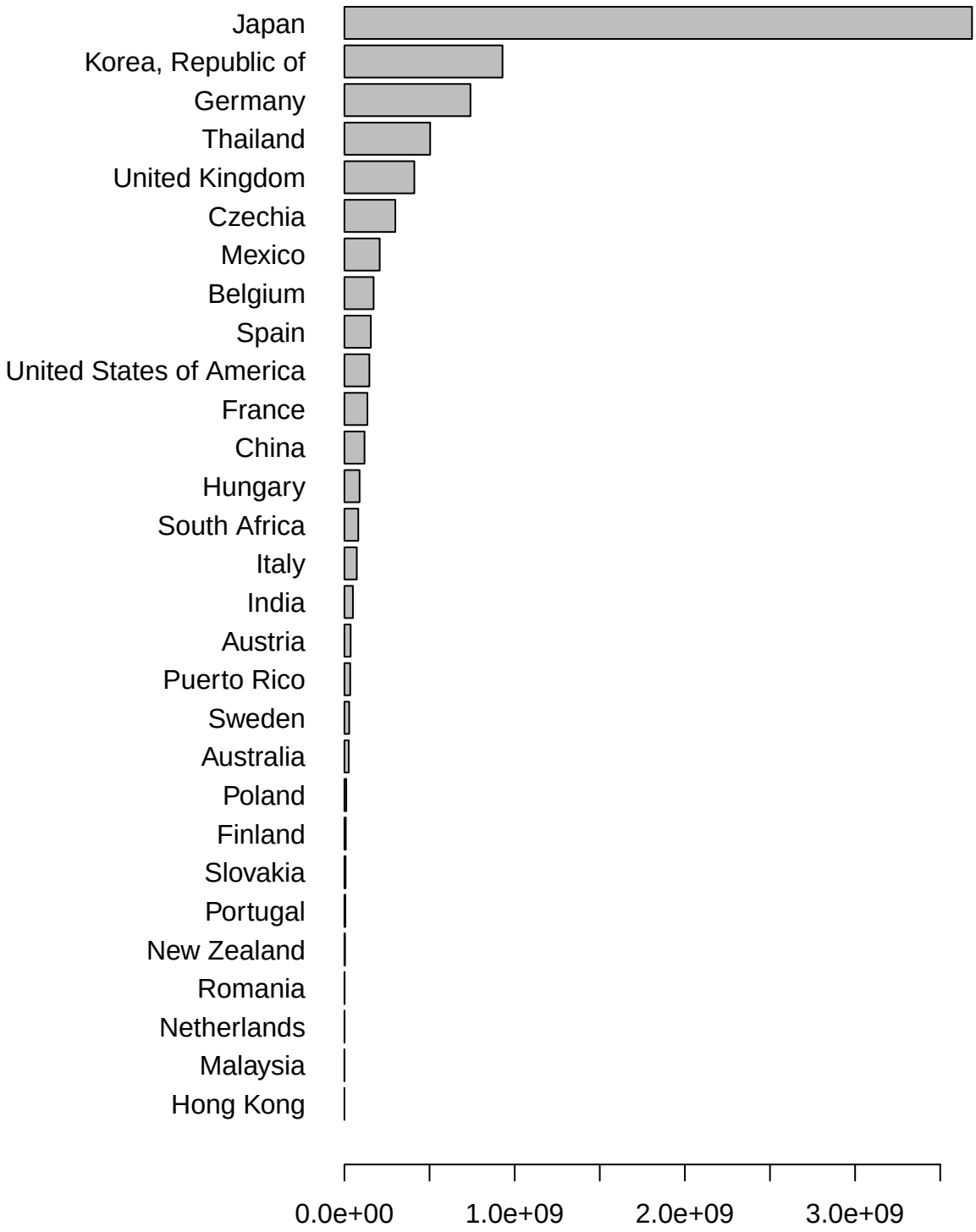
```
cars2$Month <- as.Date(paste0(cars$Month, "01"), format="%Y%m%d")
str(cars2)
```

```
## 'data.frame':    1080 obs. of  6 variables:
##  $ Month      : Date, format: "2019-01-01" "2019-01-01" ...
##  $ Unit.Qty   : chr  "NMB" "NMB" "NMB" "NMB" ...
##  $ Country    : chr  "Austria" "Belgium" "China" "Czechia" ...
##  $ vfd        : num  2511892 1886314 2675272 7251350 16902288 ...
##  $ cif        : num  2574059 1963103 2799368 7665974 17320488 ...
##  $ Imports.Qty: num  42 53 121 256 308 238 192 119 4 6 ...
```

```
tab <- table(cars2$Country)
par(mar=c(2,12,1,1))
barplot(tab[order(tab)], horiz=T, las=1)
```




```
tab <- tapply(cars2$vfd, cars2$Country, sum)
par(mar=c(2,12,1,1))
barplot(tab[order(tab)], horiz=T, las=1)
```



Propose log transformation of vfd totals to linearise bar lengths.

```
# narrow scope to Germany, scale vfd to millions NZD, sort by Month
germany <- cars2[cars2$Country == 'Germany',]
germany$vfd <- germany$vfd / 1000000
germany <- germany[order(germany$Month),]

# reserve final 10% for testing
```

```r
n <- nrow(germany)
pivot <- n - (n %/% 10)
train <- germany[1:pivot,]
test <- germany[-(1:pivot),]

mean_model <- mean(train$vfd)
linear_model <- lm(vfd ~ Month, data=train)
mean_pred <- rep(mean_model, 6)
fit_pred <- predict(linear_model, test)

rmse <- function(actual, predicted) {
    sqrt(mean((actual - predicted) ^ 2))
}

cat("RMSE for mean:", fill=T)
```

```
## RMSE for mean:
```

```r
rmse(test$vfd, mean_pred)
```

```
## [1] 6.257842
```

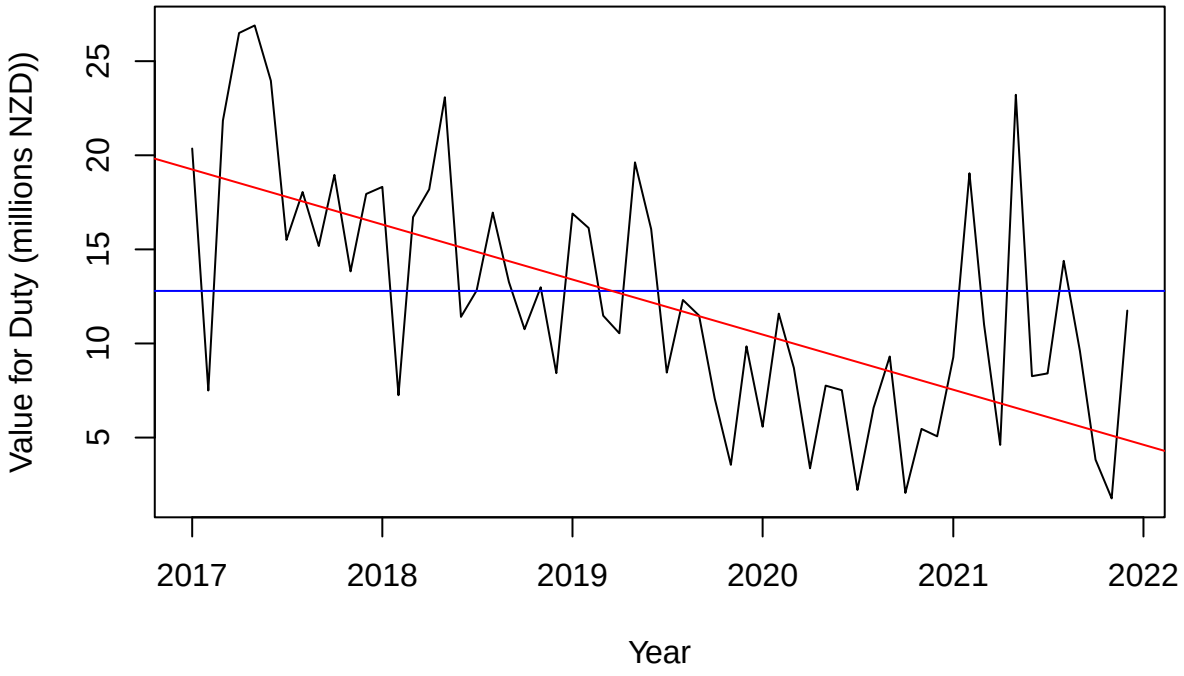
```r
cat("RMSE for linear model:", fill=T)
```

```
## RMSE for linear model:
```

```r
rmse(test$vfd, fit_pred)
```

```
## [1] 5.087111
```

```r
plot(vfd ~ Month, data=germany, type='l',
     xlab='Year', ylab='Value for Duty (millions NZD))')
abline(h=mean_model, col='blue')
abline(linear_model, col='red')
```



I think the model should use the Import.Qty as predictor for a better fit with the same complexity.

Summary