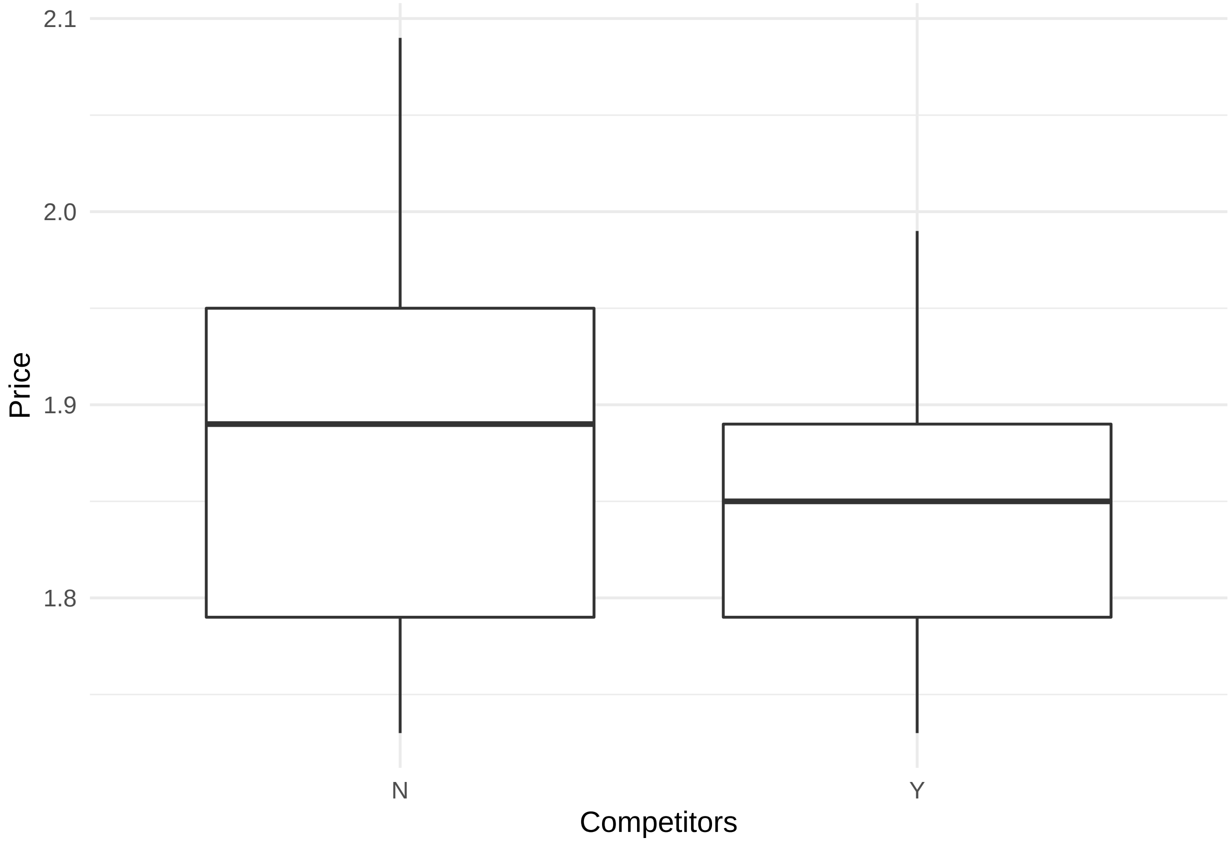
Newrmarkdown

Fjolle Gjonbalaj

8 February 2021

Problem 1

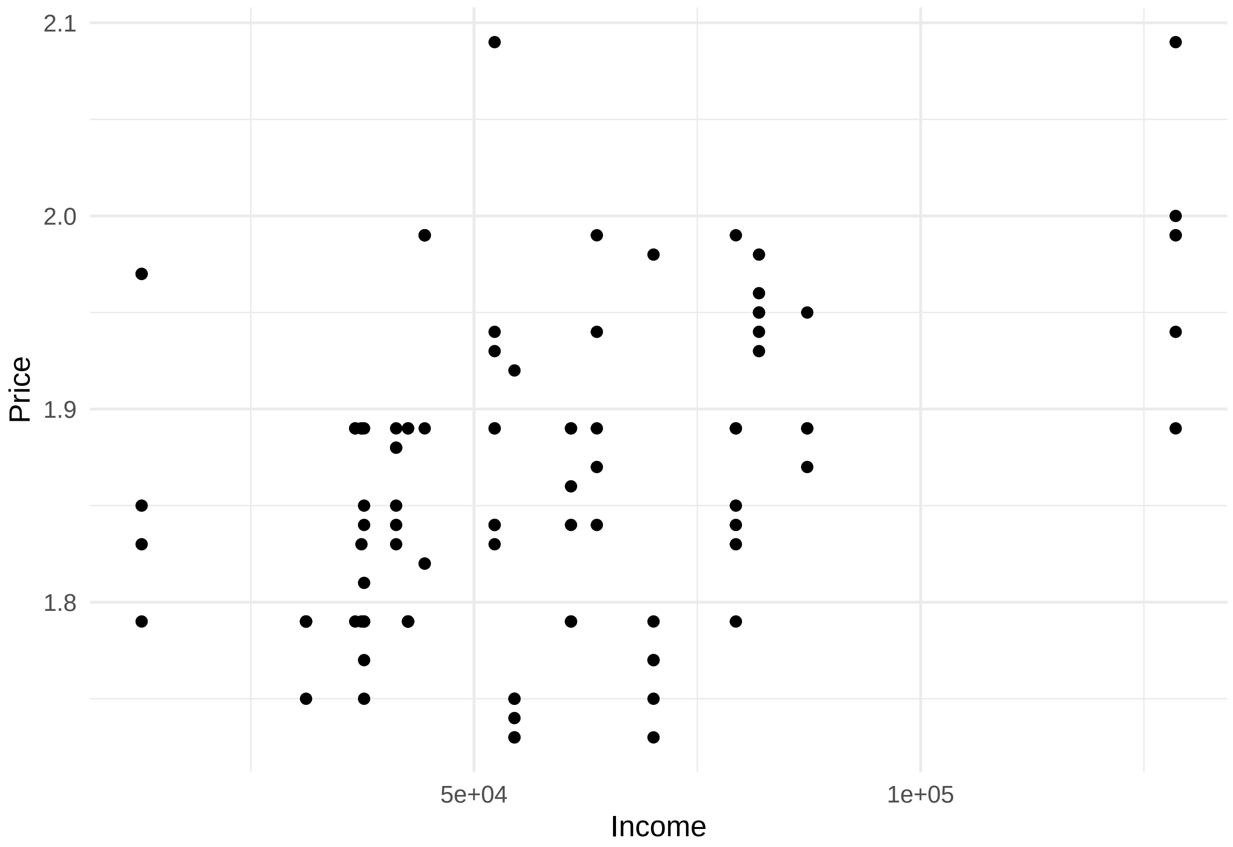
```
ggplot(data=GasPrices)+geom_boxplot(mapping=aes(x=Competitors, y=Price)) +theme_minimal()
```



Do gas stations charge more if they lack direct competition in sight?

To be able to answer this question we can look at a boxplot to see how the variable 'Competitors' affects the variable 'Price' in the data set. What we observe from the two boxplots is that the minimum price the two groups of gas stations charge is roughly the same regardless of whether there is competition or not. The difference between gas stations without competition and those with competition in sight is observed in the maximum price charged. In general, gas stations with competition charge more than those without.

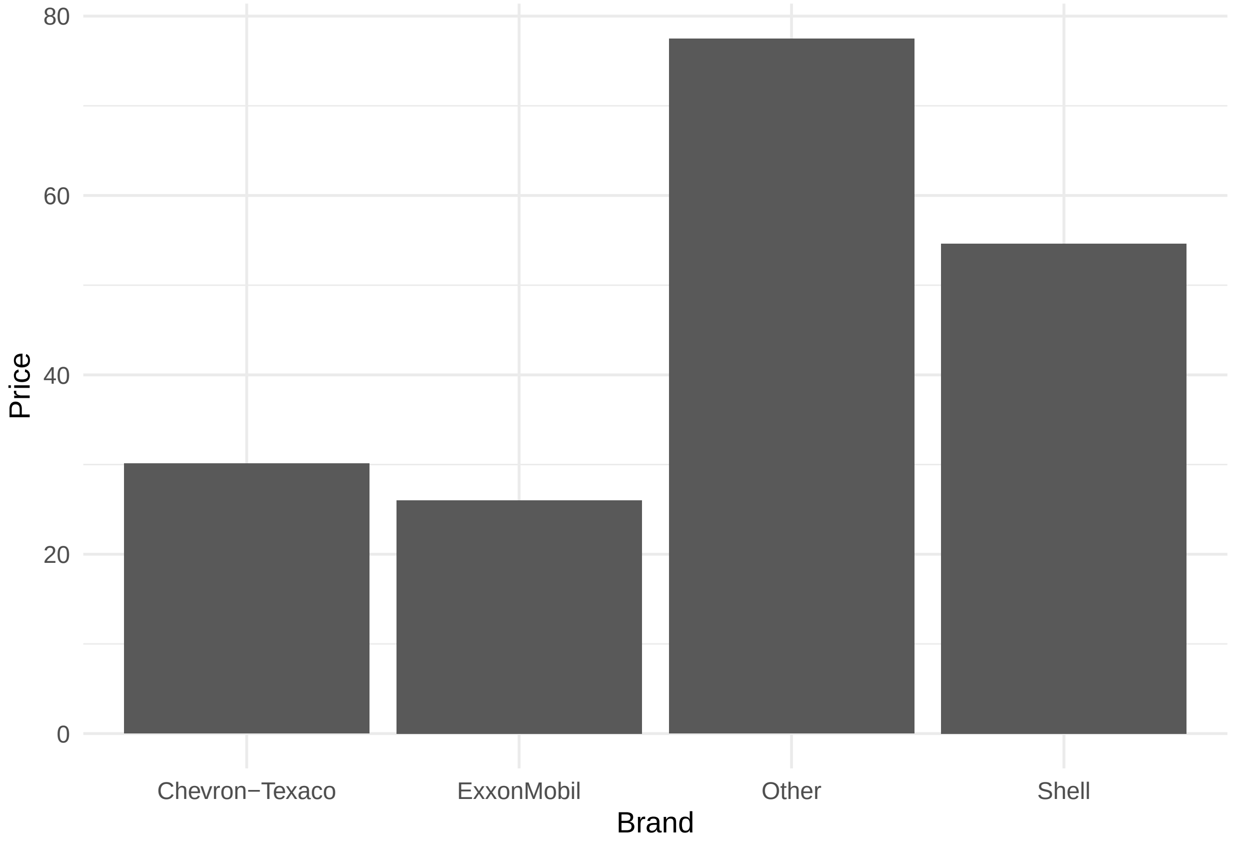
```
ggplot(data=GasPrices)+geom_point(mapping=aes(x=Income, y=Price)) +theme_minimal()
```



Do richer areas charge higher gas prices?

A scatter plot can be used to capture the relationship between gas prices and the richness of the area where the gas station is located. The data points in the scatter plot seem to be rather scattered around the entire plot, failing to show any clear connection between the variables 'Income' and 'Price'. Although this connection is not clearly expressed, eyeballing the plot gives the impression that there does exist a positive correlation between the variables, however weak that might be. To quantify this relationship a simple correlation test would be helpful.

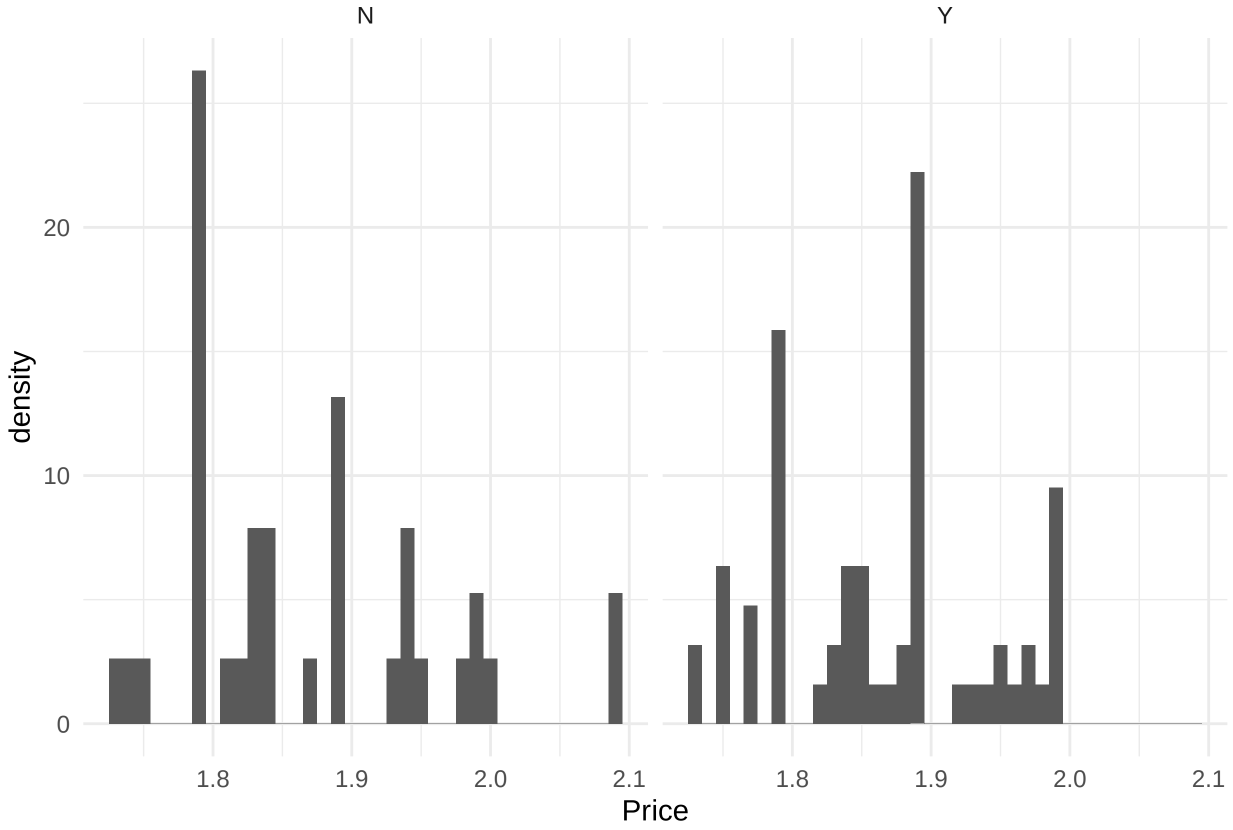
```
ggplot(data=GasPrices)+geom_col(mapping=aes(x=Brand, y=Price)) +theme_minimal()
```



Does Shell charge more than other brands?

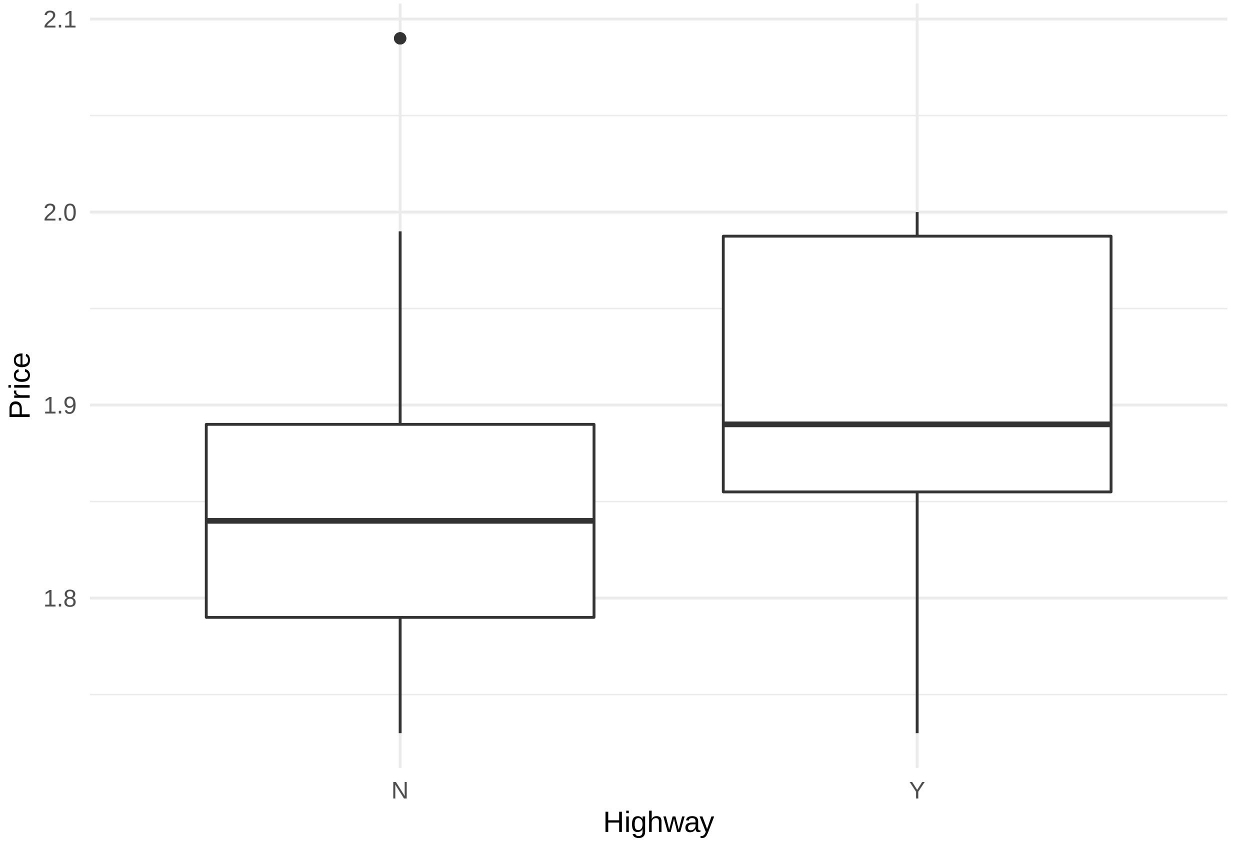
What we see in this barplot is that Shell does charge more than Chevron-Texaco and ExxonMobil. However, there are other gas station brands that charge more than Shell.

```
ggplot(data=GasPrices)+ geom_histogram(aes(x=Price, after_stat(density)),binwidth=0.01)+
  facet_wrap(~Stoplight)+theme_minimal()
```



Do gas stations at stoplights charge more? *With price on the x-axis and histogram density on the y-axis we see Gas stations that are not close to any stoplights on the left, and those that are close to a stoplight on the right. This relationship is quite unclear. It appears that there are gas stations which are not close to a stoplight, yet charge higher prices. Being near a stoplight does not seem to affect the level of prices gas stations charge.*

```
ggplot(data=GasPrices)+geom_boxplot(mapping=aes(x=Highway, y=Price)) +theme_minimal()
```



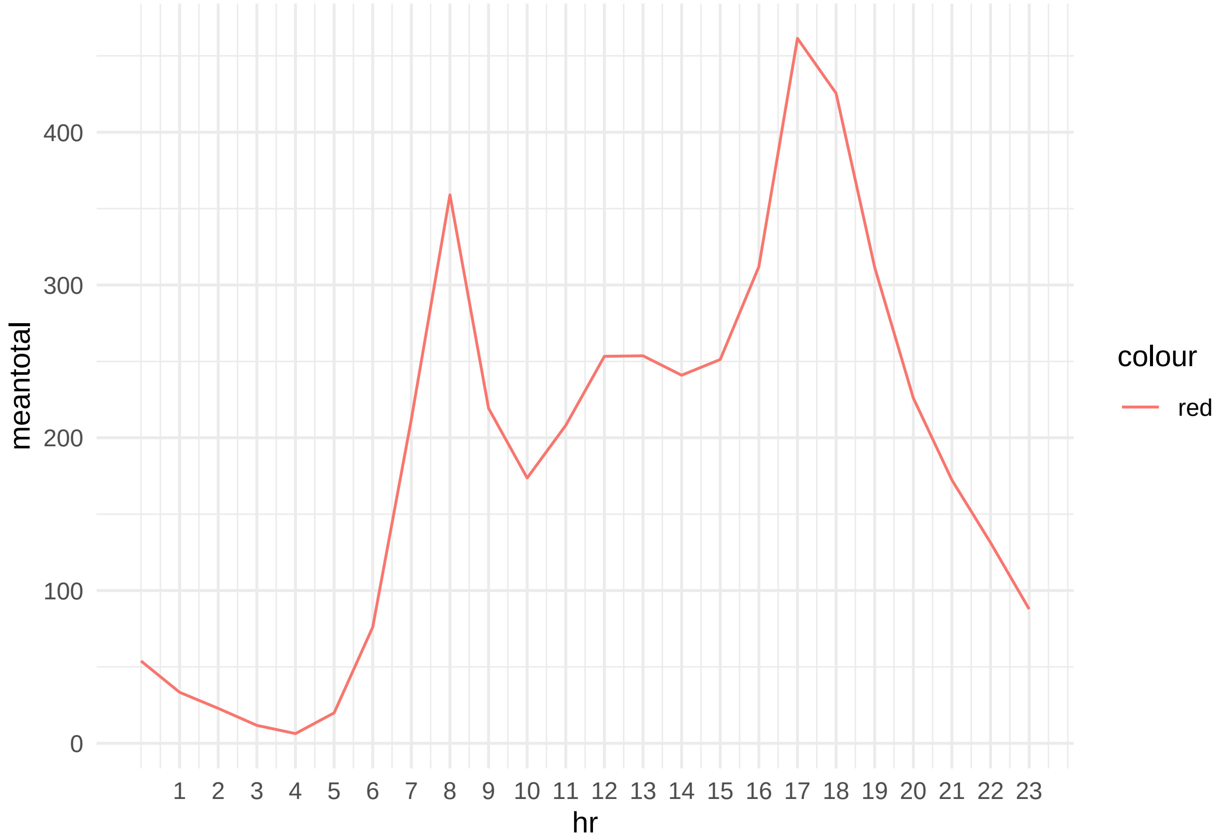
Do gas stations with direct highway access charge more? *In the boxplot above it is very clear to see that gas prices with direct access to a highway charge more than those without access to a highway. The former do not only charge higher prices on average, but also the minimum price charged is higher than that of gas stations without highway access.*

Problem 2

```
avgtotal= bikeshare %>%
  group_by(hr) %>%
  summarize(meantotal=mean(total))
```

```
## ‘summarise()‘ ungrouping output (override with ‘.groups‘ argument)
```

```
ggplot(avgtotal)+geom_line(aes(x=hr,y=meantotal, color="red")) +
  scale_x_continuous(breaks=1:23) +theme_minimal()
```

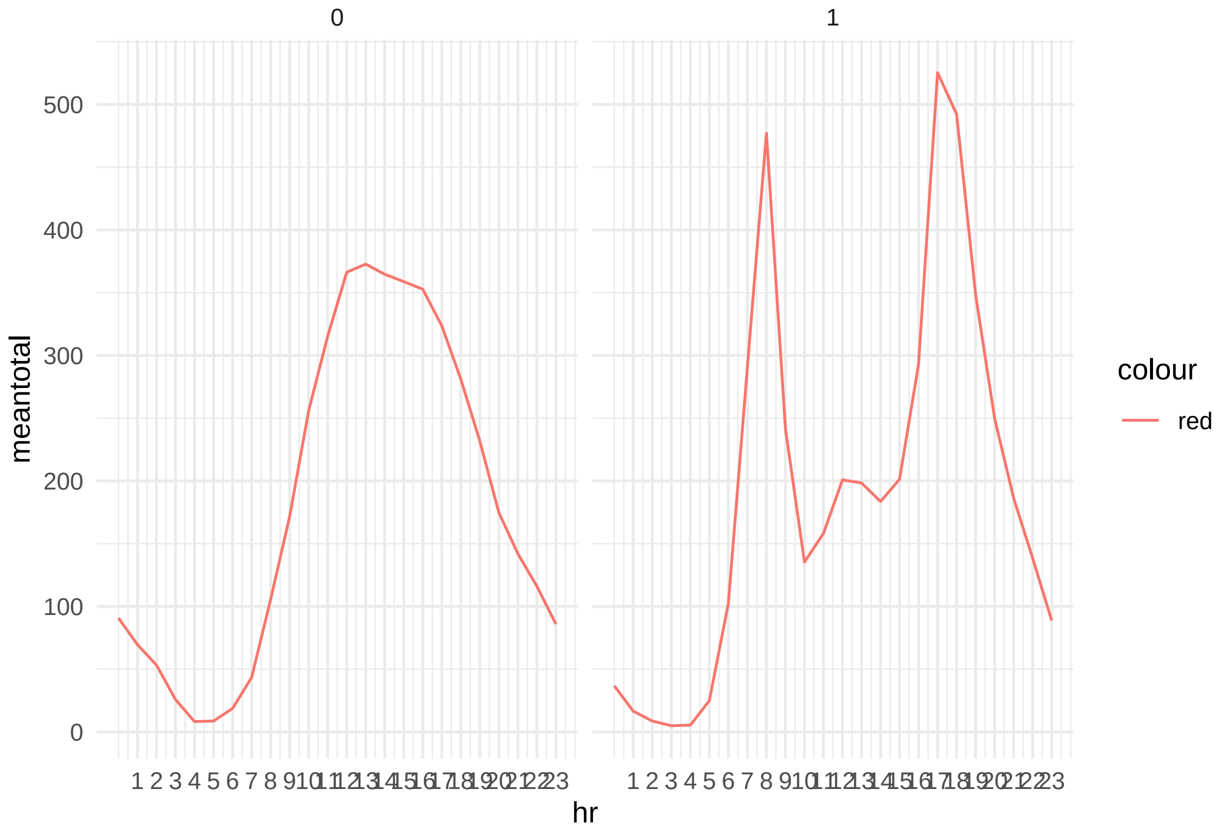


In this line graph we can see the average bike rides on the y-axis for each hour of the day on the x-axis. We oserve a spike of bike rides at 8 am, and another one at 5 pm. The two spikes are most likely happening during these two hours from people who decided to ride bikes to go and come back from work. The use of bikeshare bikes is lowest from midnight to four a'clock in the morning.

```
avgtotal= bikeshare %>%
  group_by(hr, workingday) %>%
  summarize(meantotal=mean(total))
```

```
## 'summarise()' regrouping output by 'hr' (override with '.groups' argument)
```

```
ggplot(avgtotal)+geom_line(aes(x=hr,y=meantotal, color="red")) +
  scale_x_continuous(breaks=1:23)+facet_wrap(~workingday)+theme_minimal()
```

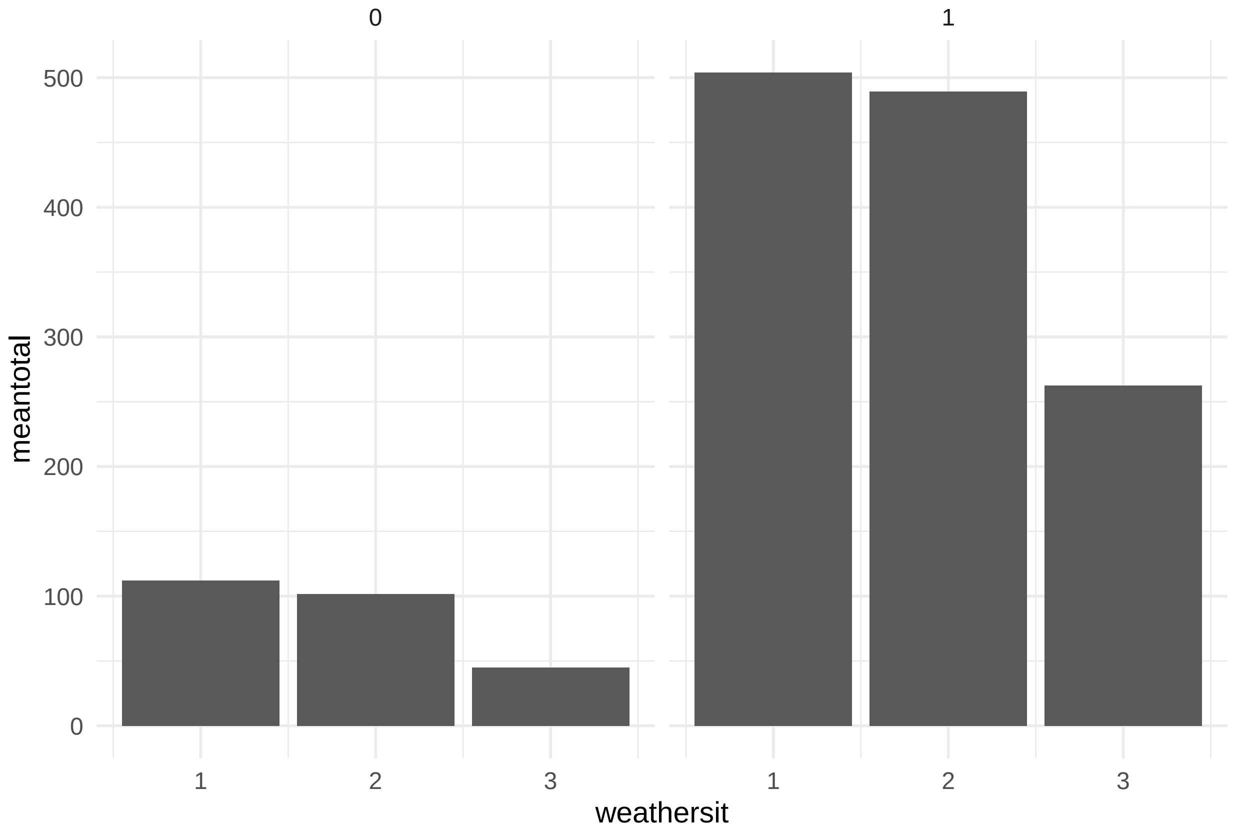


The faceted line graph showing average bike rentals versus hour of the day, faceted according to whether or not it is a working day, demontrates that there are no spikes in bike ridership during the weekend. Instead, the average bike rentals during the weekend start rising at 5 am in a steady manner, and reach a peak around 2:30 pm.

```
d1=bikeshare %>% filter(hr=='8') %>% group_by(weathersit, workingday) %>% summarize(meantotal=mean(tota
```

```
## ‘summarise()‘ regrouping output by ’weathersit’ (override with ‘.groups‘ argument)
```

```
ggplot(data=d1)+geom_col(mapping=aes(x=weathersit,y=meantotal))+
  facet_wrap(~workingday) +theme_minimal()
```



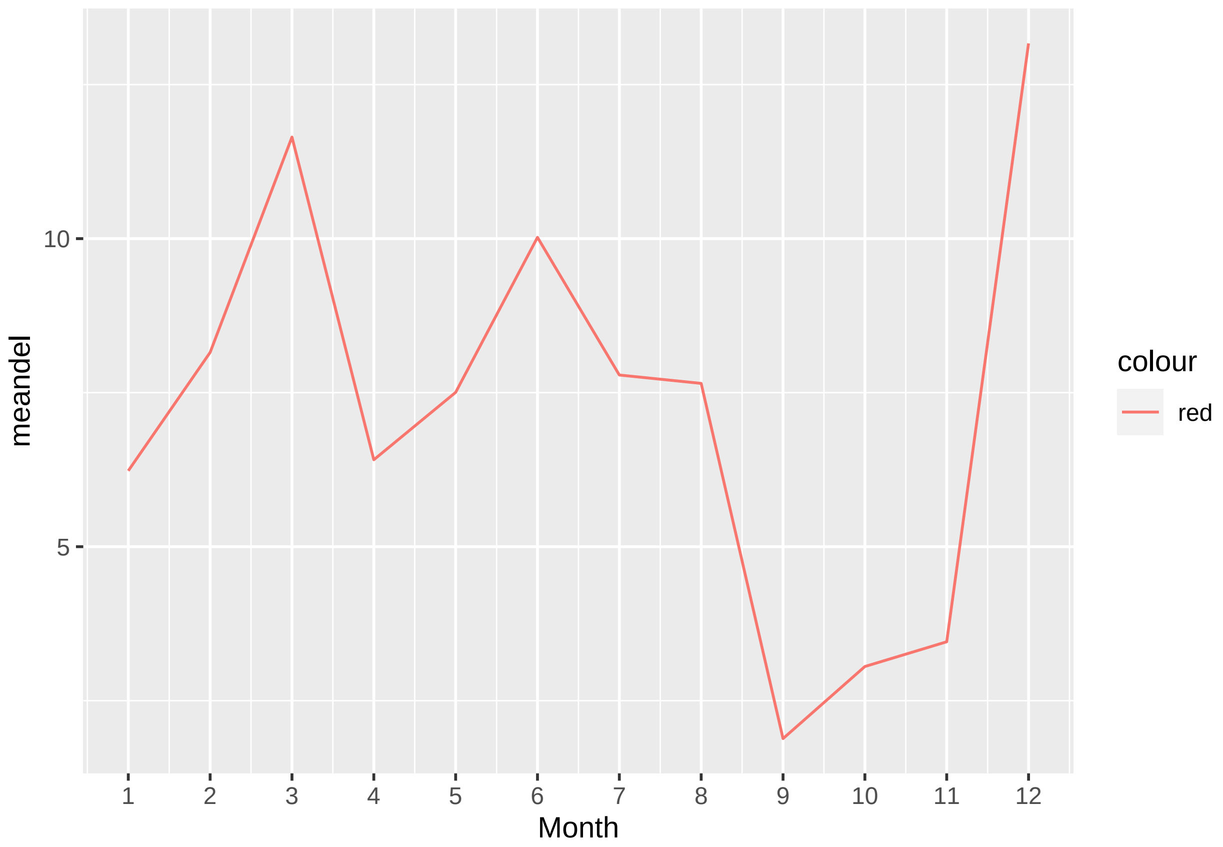
The faceted bar plot above shows the average ridership during the 8 AM hour by weather situation, faceted according to whether or not it is a working day. Weather situation has been divided into three groups in the x-axis: 1 indicates that the weather is clear and partly cloudy. 2 indicates there is mist + clouds. 3 means there was either light snow, light rain and/or thunderstorm; and 4 idicates heavy rain + ice pallets, Snow and Fog(Omitted category) It is clear to see that at 8am there are always more bike rentals on average during a working day than during the weekend. On the other hand, for both groups, the average bike ridership at 8 am is smaller when there is mist a clouds than when the weather is clear and partly cloudy; and even smaller when there is light snow or rain. Group 4 has been ommitted and serves as our base level from which we can draw comparisons.

Problem 3

```
avg1= ABIA %>%
  filter(Origin=='AUS', TaxiIn<'70') %>%
  group_by(Month) %>%
  summarize(meandel=mean(DepDelay))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ggplot(avg1)+geom_line(aes(x=Month,y=meandel, color="red")) +
  scale_x_continuous(breaks=1:12)
```

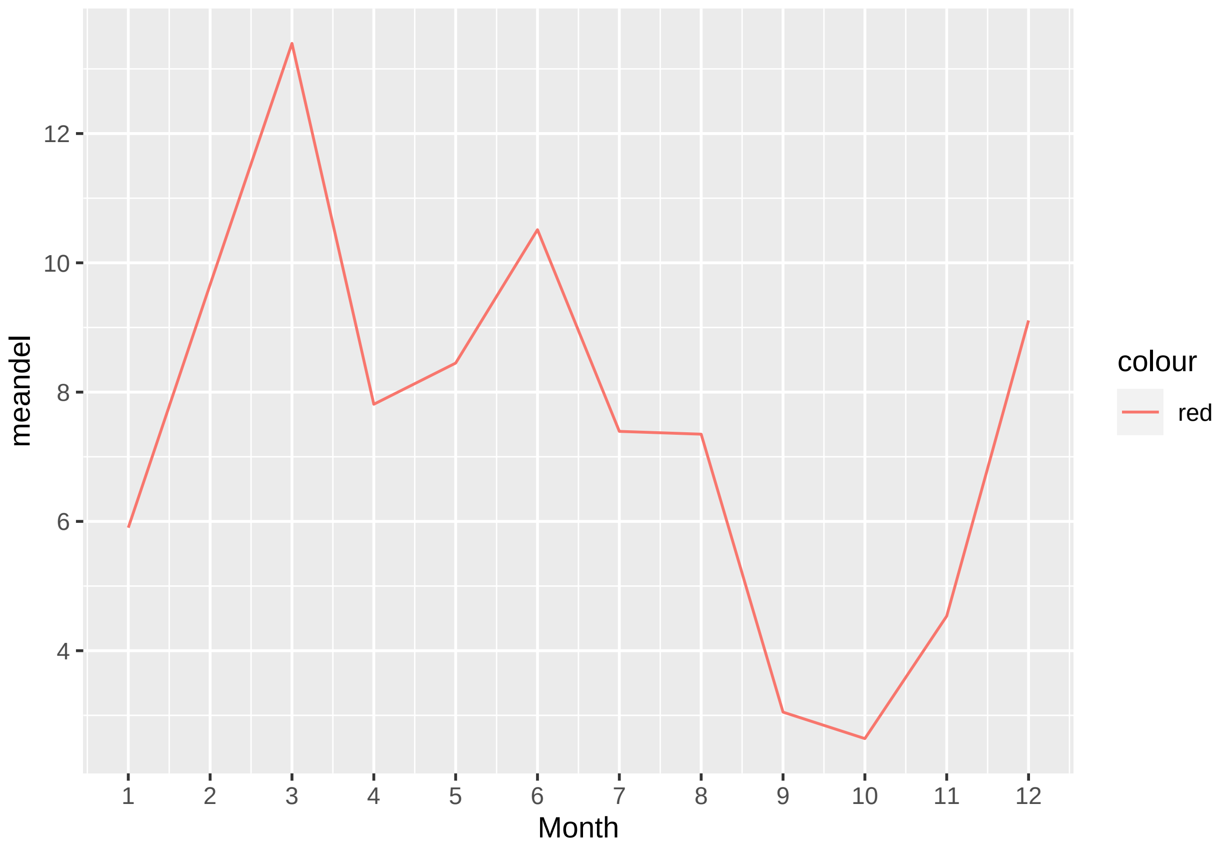


The line graph above shows the average departure delay (in minutes) of flights out of Austin, given that the Taxi-In period in minutes is less than 70. The figure shows that out of the flights which landed in Austin and had Taxi-in period of less than 70 minutes, average departure delays were the highest during the winter season, with a spike in December. The other two spikes observed are in March and in June. These are most likely due to Spring and Summer holidays when the number of flights as well as the number of people flying is the highest.

```
avg2= ABIA %>%
  filter(Origin=='AUS', TaxiIn>='70') %>%
  group_by(Month) %>%
  summarize(meandel=mean(DepDelay))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ggplot(avg2)+geom_line(aes(x=Month,y=meandel, color="red")) +
  scale_x_continuous(breaks=1:12)
```

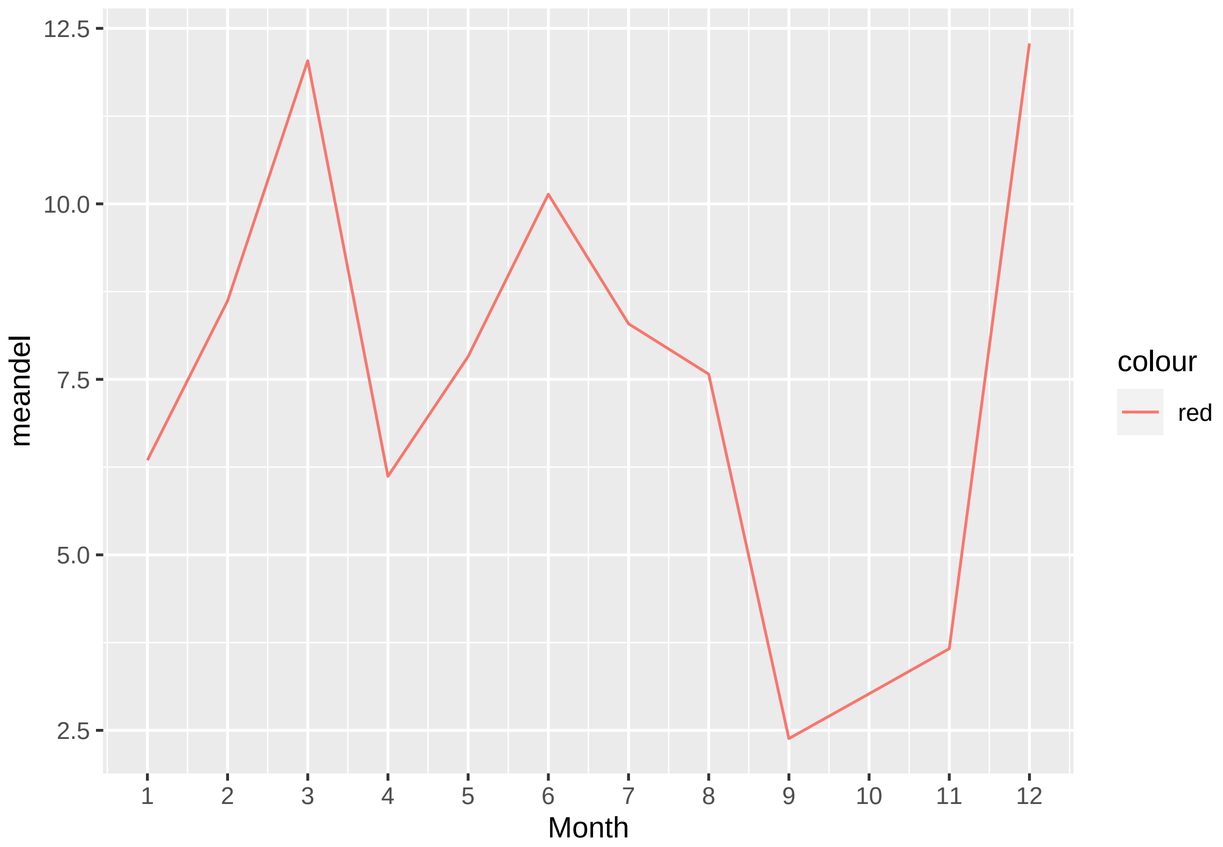


This line graph shows the average departure delay (in minutes) of flights out of Austin, given that the Taxi-In period in minutes is greater than or equal to 70. The figure tells that when the taxi-in period is greater than or equal to 70, the departure delay period is higher on average. The spikes during March, June and December remain unchanged regardless of the Taxi-In period. The difference observed between the two graphs are the months of October, November and December, with the latter graph indicating a smoother transitioning of departure delays during these less busy months.

```
avg3= ABIA %>%
  filter(Origin=='AUS', TaxiOut<'70') %>%
  group_by(Month) %>%
  summarize(meandel=mean(DepDelay))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ggplot(avg3)+geom_line(aes(x=Month,y=meandel, color="red")) +
  scale_x_continuous(breaks=1:12)
```

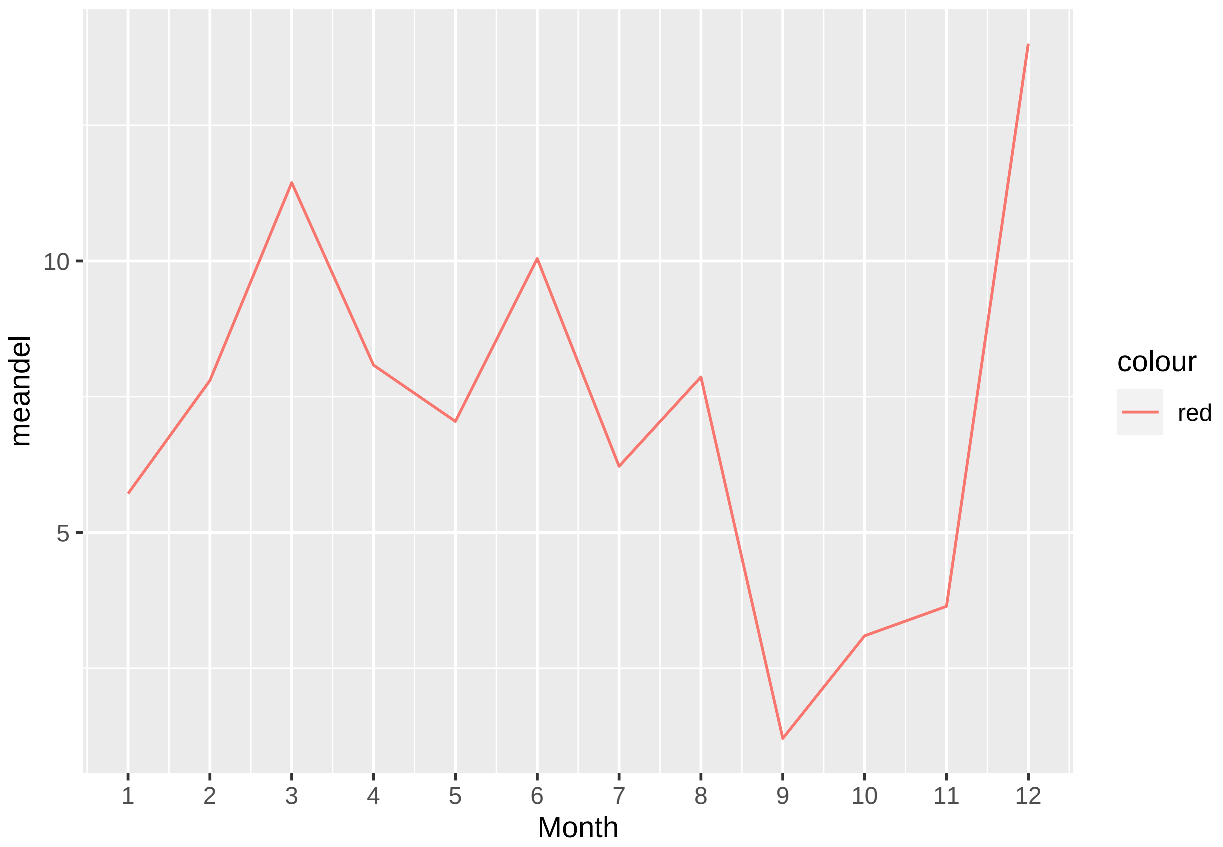


This line graph shows the average departure delay (in minutes) of flights out of Austin, given that the Taxi-Out period in minutes is less than 70. Here, again, departure delays during March, June and December are relatively high compared to other months, the highest being in December during winter holidays. These observed spikes happen regardless of the taxi-in or taxi-out period.

```
avg4= ABIA %>%
  filter(Origin=='AUS', TaxiOut>='70') %>%
  group_by(Month) %>%
  summarize(meandel=mean(DepDelay))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
ggplot(avg4)+geom_line(aes(x=Month,y=meandel, color="red")) +
  scale_x_continuous(breaks=1:12)
```



The graph with Taxi-Out period of greater than or equal to 70 indicates that the average departure delay (in minutes) of flights out of Austin is lower on average. This might be a counter-intuitive finding to what we would normally expect to find. However, a low correlation between departure delays and taxi-in & taxi-out period might be an explanation.

Problem 4

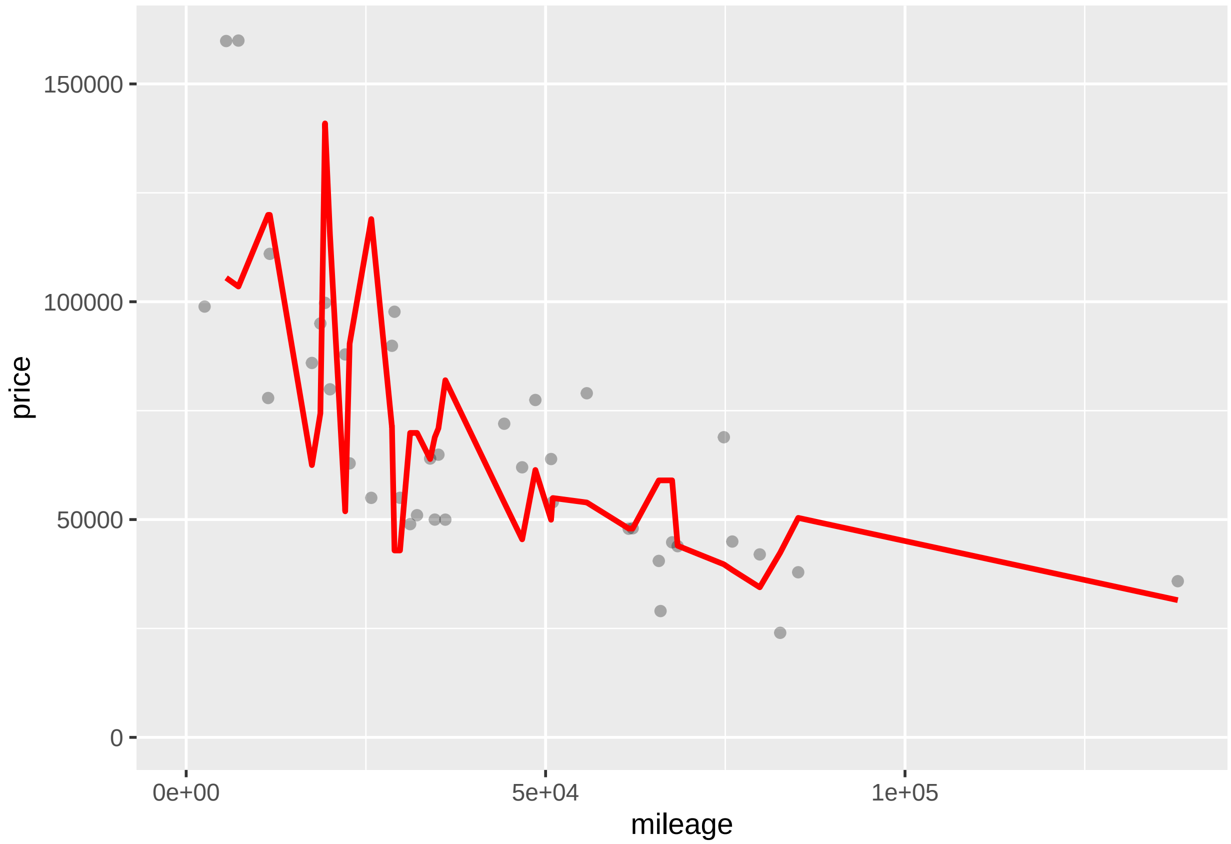
```
AMG65=filter(sclass, sclass$trim=="65 AMG")

sclass_split =  initial_split(AMG65, prop=0.8)
sclass_train = training(sclass_split)
sclass_test  = testing(sclass_split)
sclass_test = arrange(sclass_test, mileage)

knn2=knnreg(price~mileage, data=sclass_train, k=2)
rmse(knn2, sclass_test)
```

```
## [1] 32199.69
```

```
price_pred=predict(knn2, sclass_test)

sclass_t = sclass_test %>%
  mutate(price_pred)

p_test=ggplot(data=sclass_t)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```

```
## Warning: Removed 19 rows containing missing values (geom_point).
```

```
## Warning: Removed 19 row(s) containing missing values (geom_path).
```



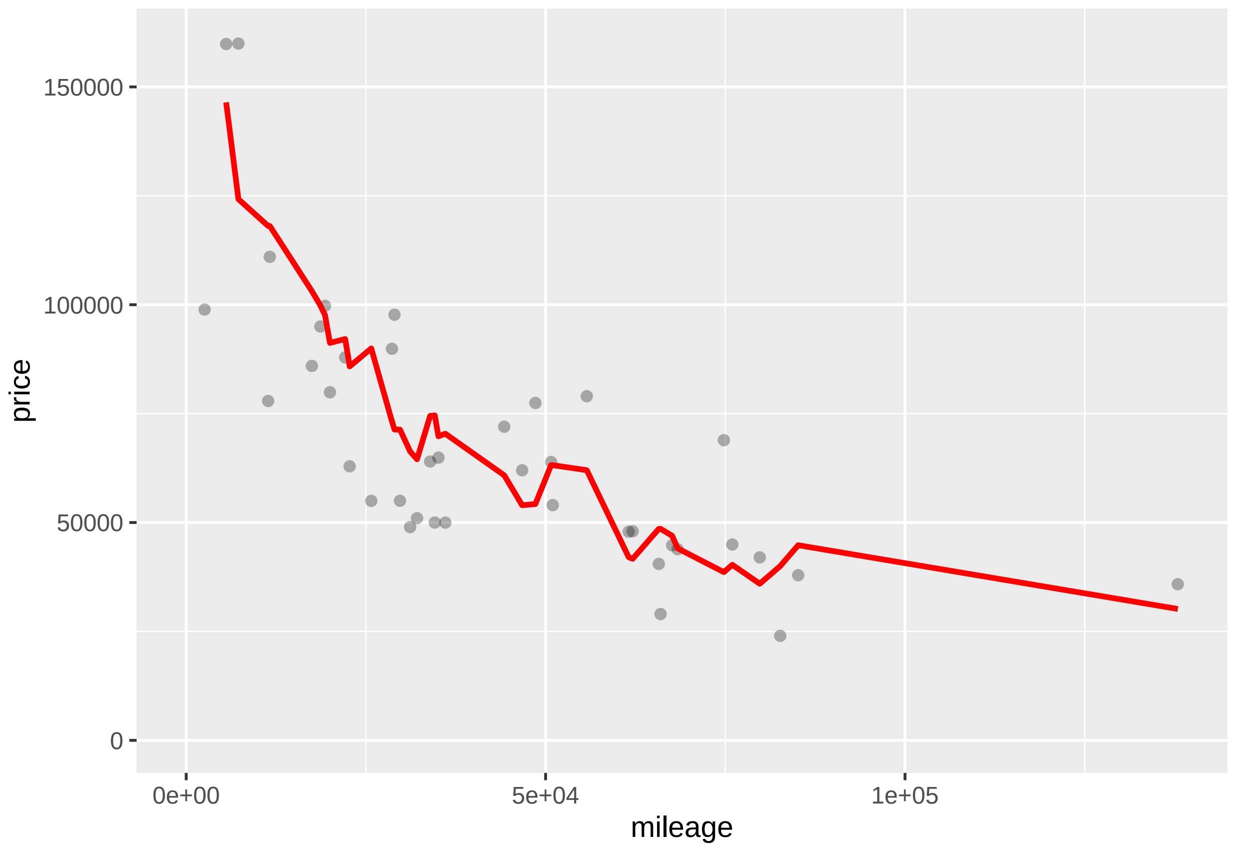
```
knn10=knnreg(price~mileage, data=sclass_train, k=10)
rmse(knn10, sclass_test)
```

```
## [1] 26180.49
```

```
price_pred=predict(knn10, sclass_test)

sclass_t2 = sclass_test %>%
  mutate(price_pred)

p_test=ggplot(data=sclass_t2)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```

```
## Warning: Removed 19 rows containing missing values (geom_point).
```

```
## Warning: Removed 19 row(s) containing missing values (geom_path).
```



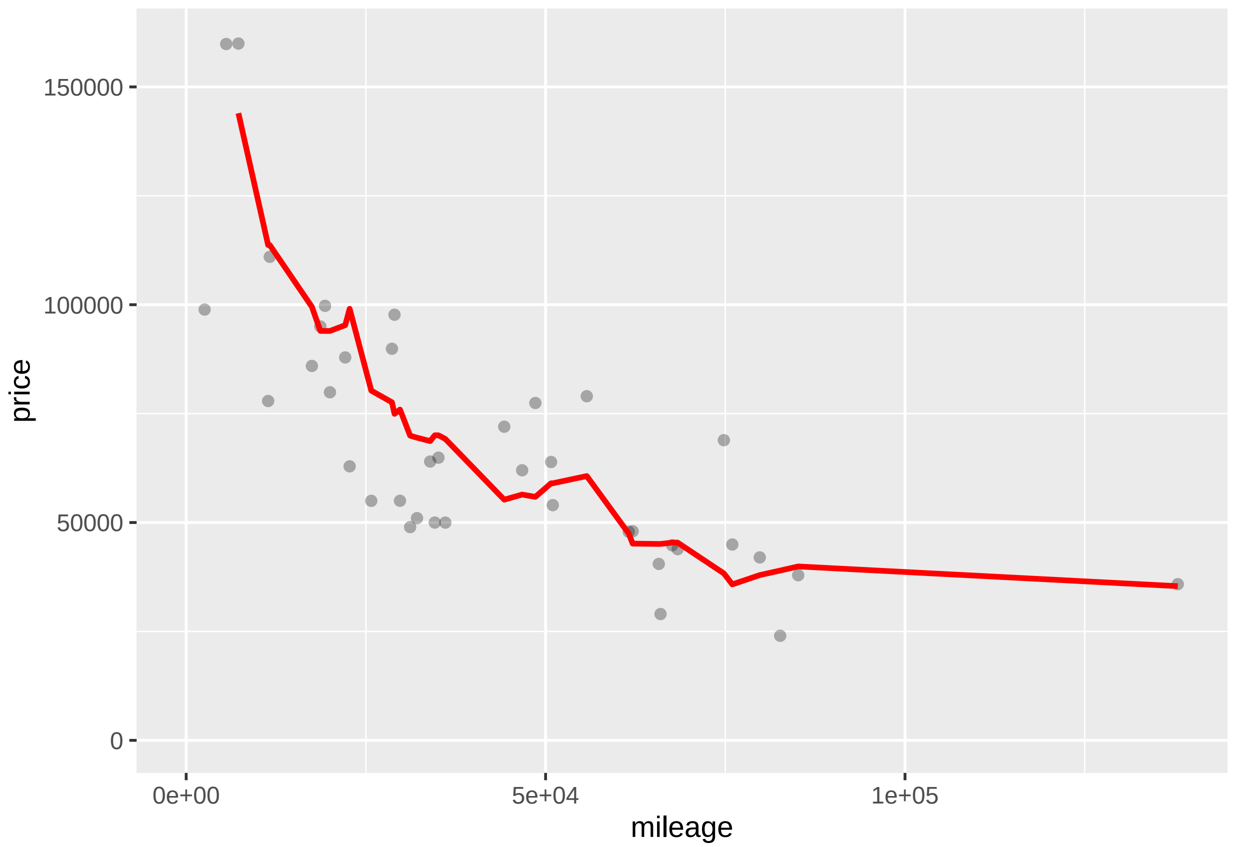
```r
knn15=knnreg(price~mileage, data=sclass_train, k=15)
rmse(knn15, sclass_test)
```

```
## [1] 25166.72
```

```r
price_pred=predict(knn15, sclass_test)

sclass_t3 = sclass_test %>%
  mutate(price_pred)

p_test=ggplot(data=sclass_t3)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```

```
## Warning: Removed 19 rows containing missing values (geom_point).
```

```
## Warning: Removed 20 row(s) containing missing values (geom_path).
```



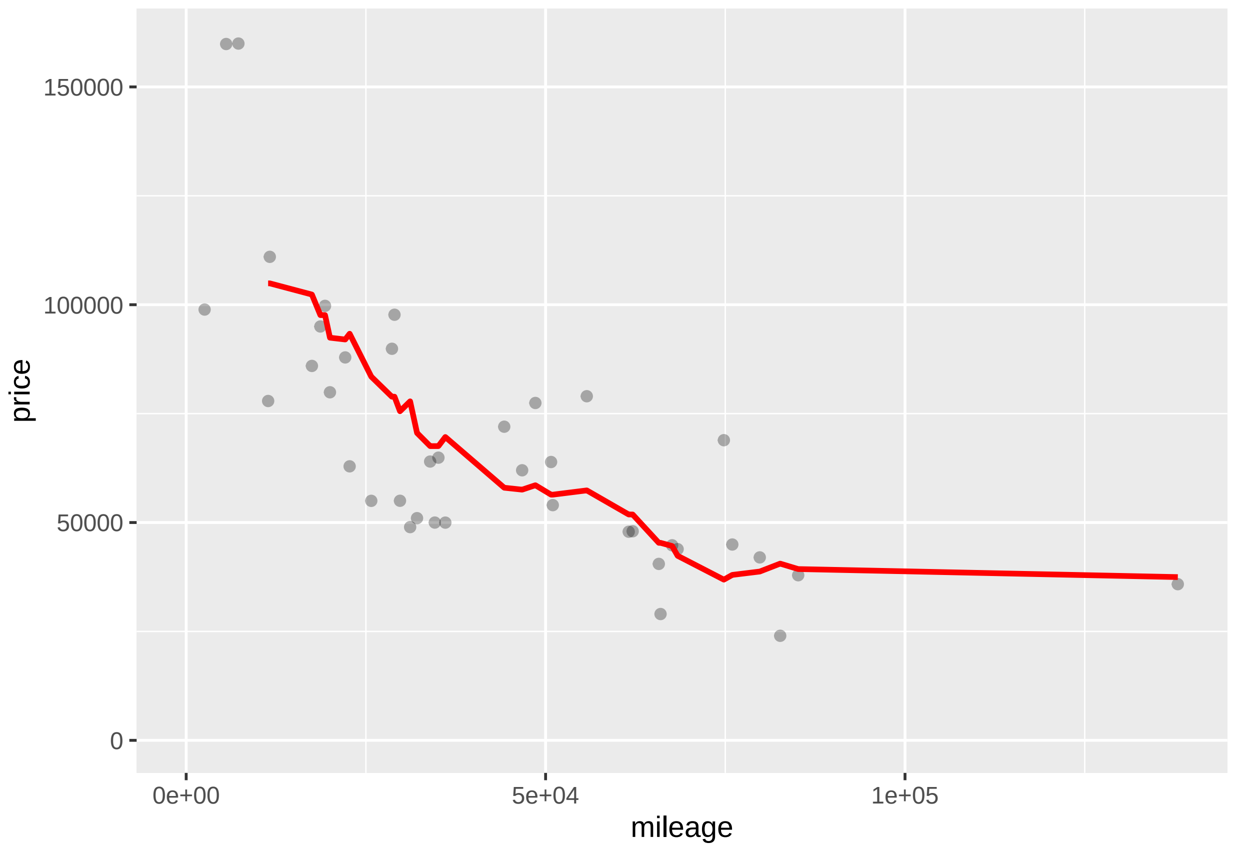
```
knn20=knnreg(price~mileage, data=sclass_train, k=20)
rmse(knn20, sclass_test)
```

```
## [1] 25169.38
```

```
price_pred=predict(knn20, sclass_test)

sclass_t4 = sclass_test %>%
  mutate(price_pred)

p_test=ggplot(data=sclass_t4)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```

```
## Warning: Removed 19 rows containing missing values (geom_point).
```

```
## Warning: Removed 21 row(s) containing missing values (geom_path).
```

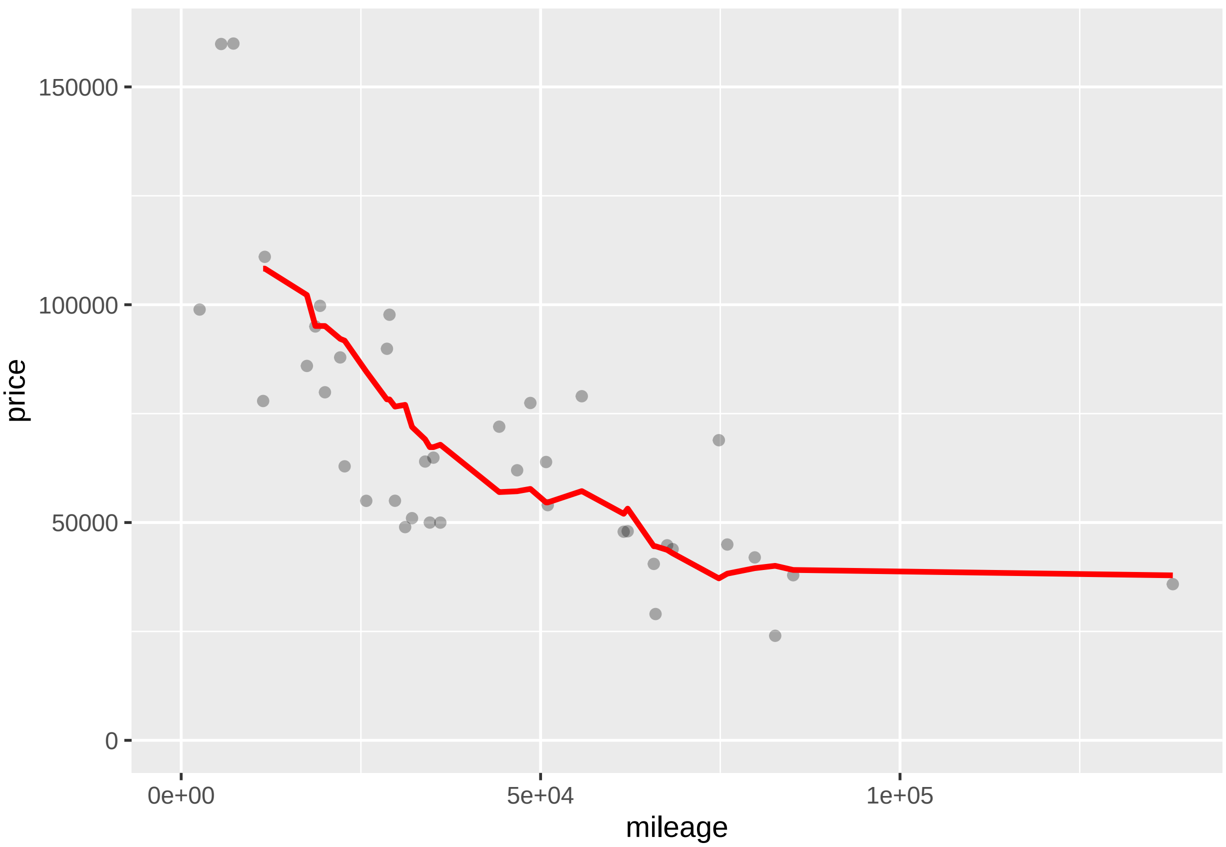


```
knn21=knnreg(price~mileage, data=sclass_train, k=21)
rmse(knn21, sclass_test)
```

```
## [1] 25615.8
```

```
price_pred=predict(knn21, sclass_test)

sclass_t5 = sclass_test %>%
  mutate(price_pred)


p_test=ggplot(data=sclass_t5)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```

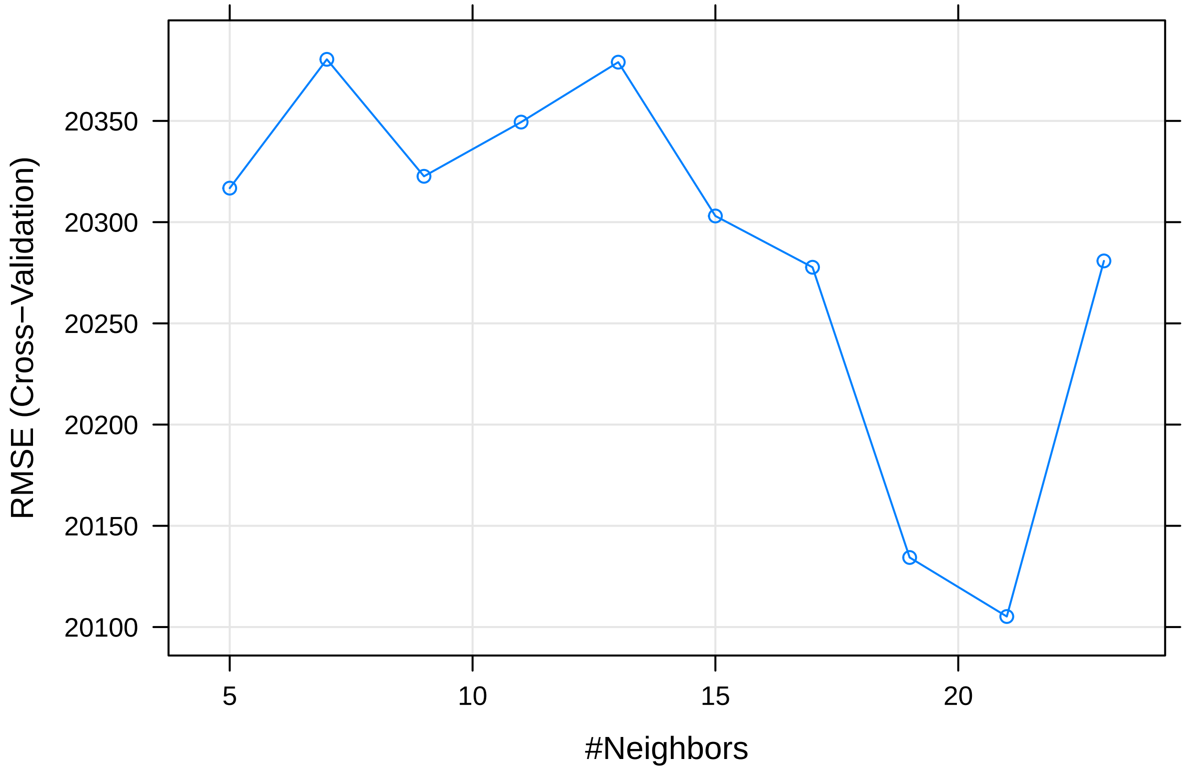
```
## Warning: Removed 19 rows containing missing values (geom_point).

## Warning: Removed 21 row(s) containing missing values (geom_path).
```



```
model <- train(price~mileage, data = sclass_train, method = "knn",
               trControl = trainControl("cv", number = 10),
               preProcess = c("center","scale"),
               tuneLength =10)
predictions <- model %>% predict(sclass_test)
RMSE(predictions, sclass_test$mileage)
```

```
## [1] 139743.9
```

```
plot(model)
```

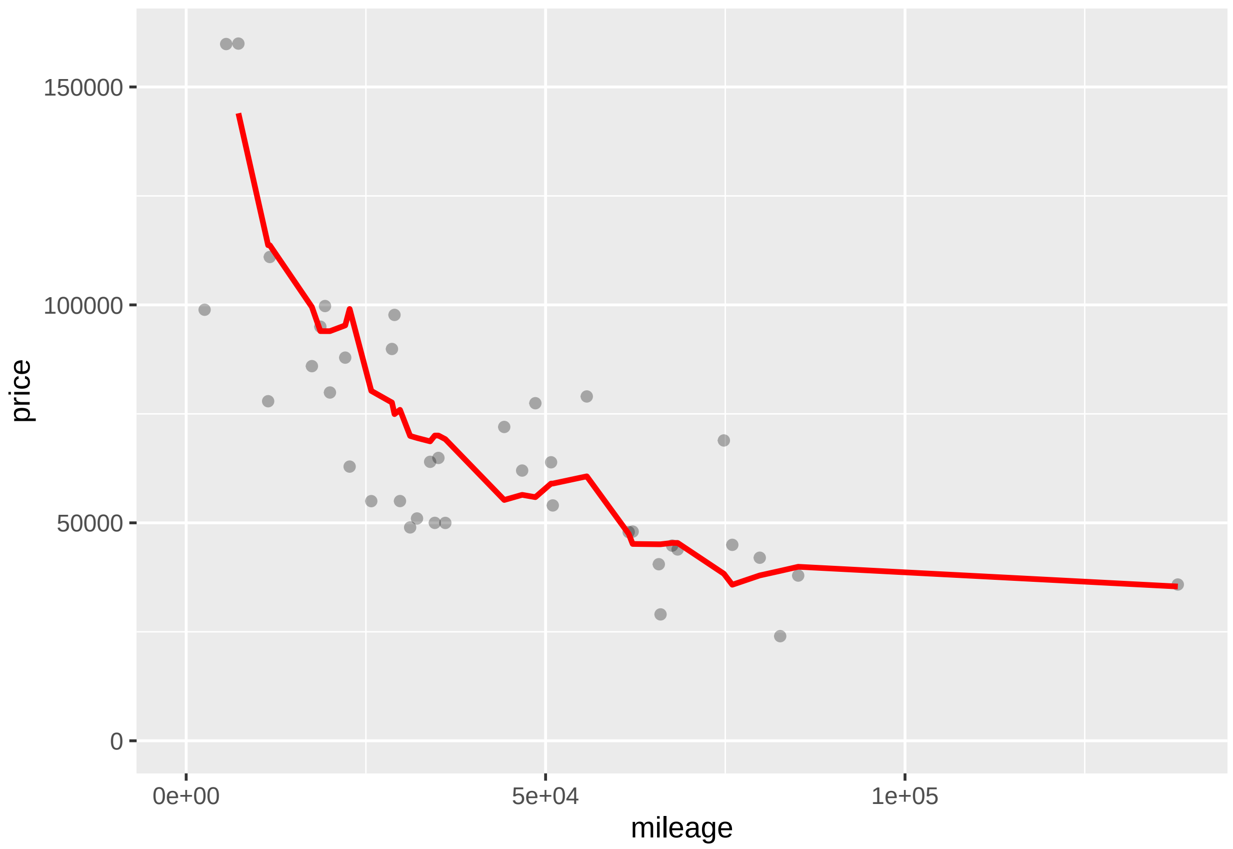


```
#FOR THE OPTIMAL LEVEL OF K:

knn18=knnreg(price~mileage, data=sclass_train, k=18)
rmse(knn18, sclass_test)
```

```
## [1] 25135.55
```

```
price_pred=predict(knn15, sclass_test)

sclass_t6 = sclass_test %>%
  mutate(price_pred)

p_test=ggplot(data=sclass_t6)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```

```
## Warning: Removed 19 rows containing missing values (geom_point).

## Warning: Removed 20 row(s) containing missing values (geom_path).
```

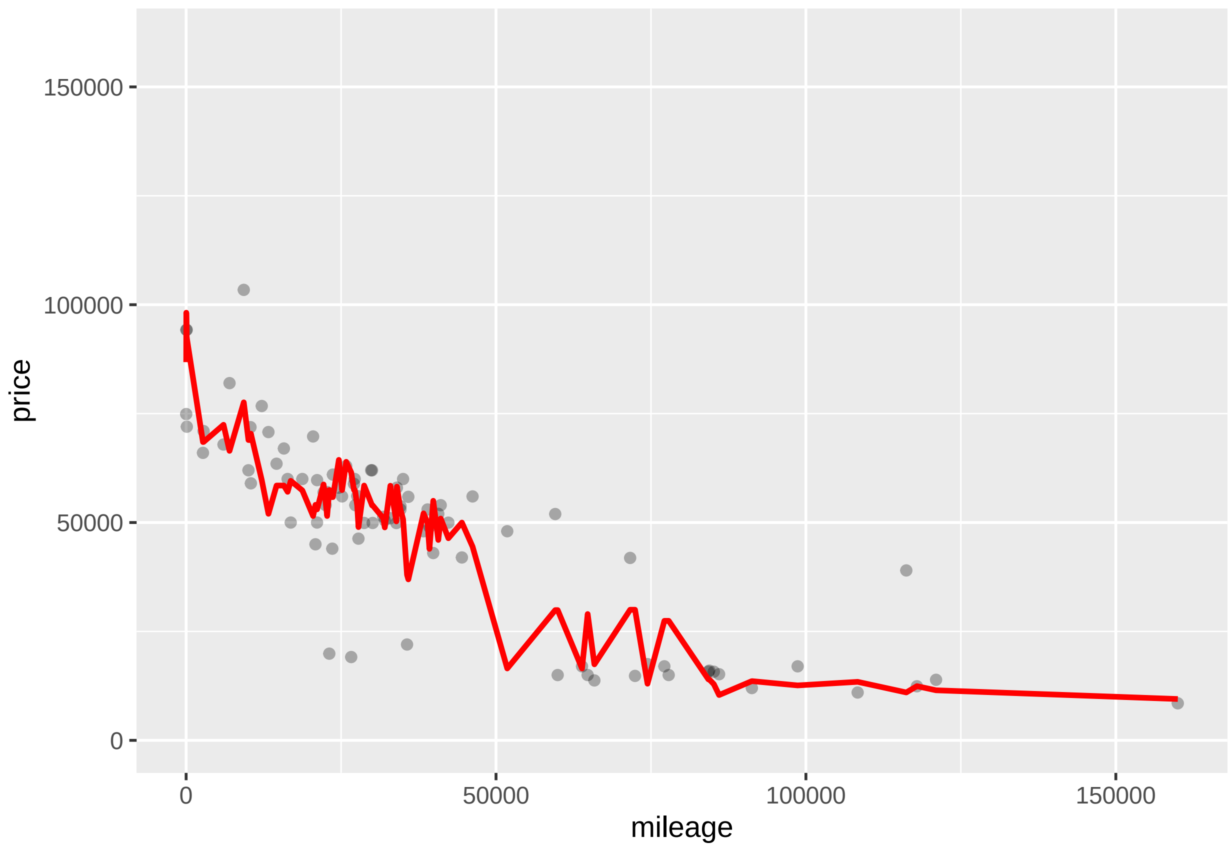


-

```
A350=filter(sclass, sclass$trim=="350")

sclass_split =  initial_split(A350, prop=0.8)
sclass_train = training(sclass_split)
sclass_test  = testing(sclass_split)
sclass_test = arrange(sclass_test, mileage)

knn2=knnreg(price~mileage, data=sclass_train, k=2)
rmse(knn2, sclass_test)
```

```
## [1] 11692.65
```

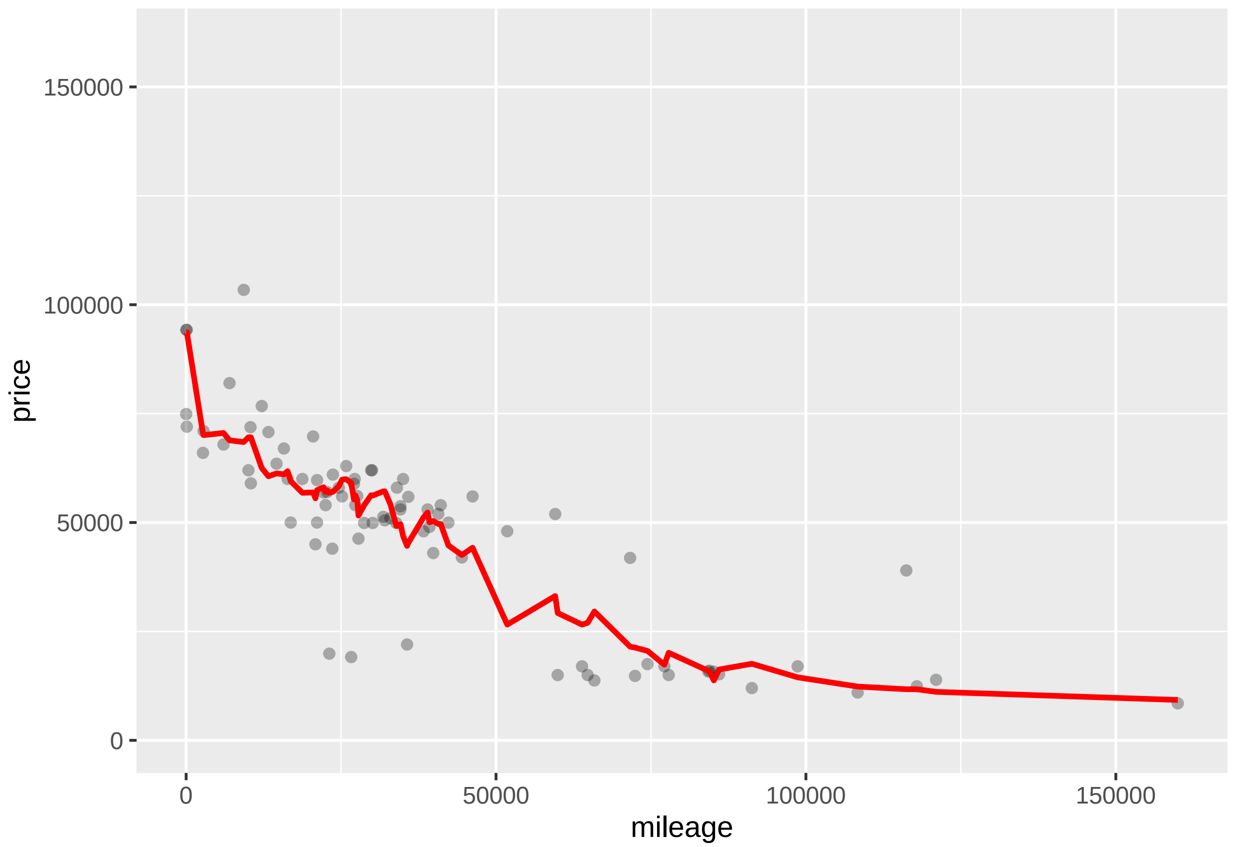
```
price_pred=predict(knn2, sclass_test)

sclass_t7 = sclass_test %>%
  mutate(price_pred)

p_test=ggplot(data=sclass_t7)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```



```
knn10=knnreg(price~mileage, data=sclass_train, k=10)
rmse(knn10, sclass_test)
```

```
## [1] 11307.64
```

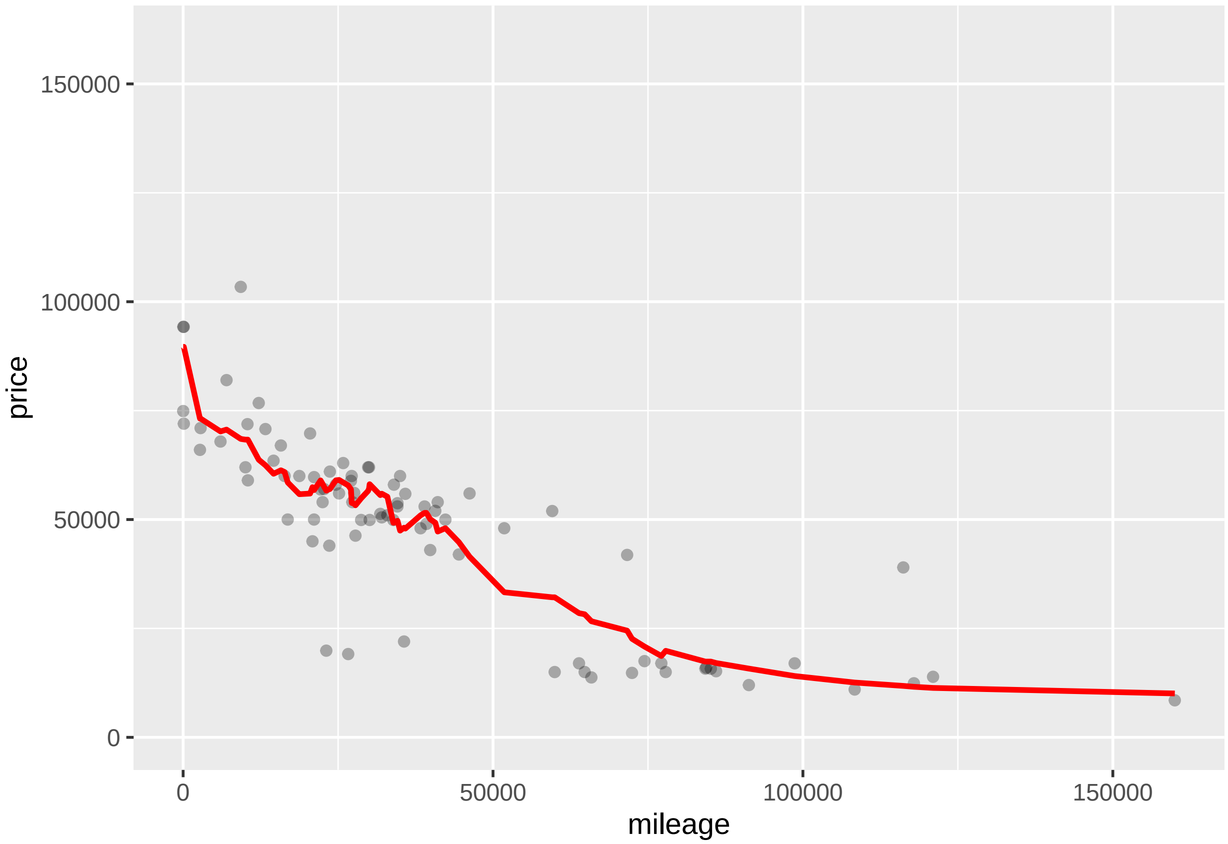
```
price_pred=predict(knn10, sclass_test)

sclass_t8 = sclass_test %>%
  mutate(price_pred)

p_test=ggplot(data=sclass_t8)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```



```
knn15=knnreg(price~mileage, data=sclass_train, k=15)
rmse(knn15, sclass_test)
```

```
## [1] 11088.48
```

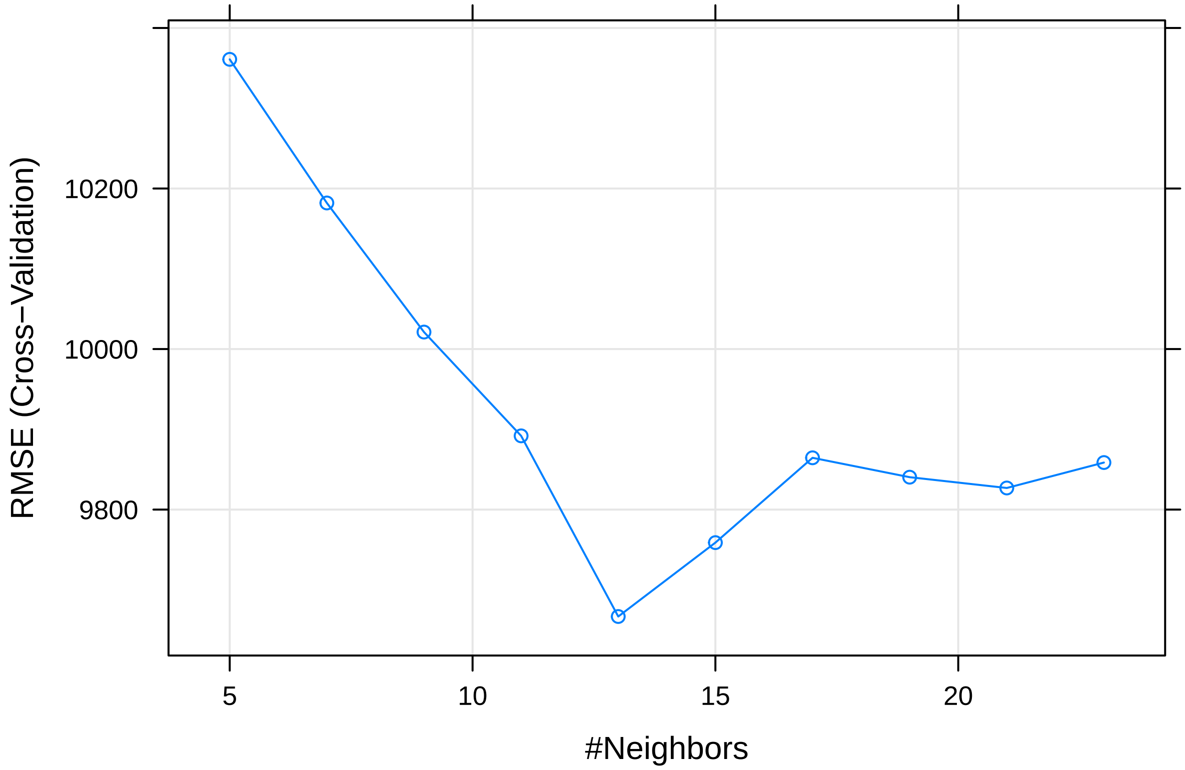
```
price_pred=predict(knn15, sclass_test)

sclass_t9 = sclass_test %>%
  mutate(price_pred)

p_test=ggplot(data=sclass_t9)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```



```
model <- train(price~mileage, data = sclass_train, method = "knn",
               trControl = trainControl("cv", number = 10),
               preProcess = c("center","scale"),
               tuneLength =10)
predictions <- model %>% predict(sclass_test)
RMSE(predictions, sclass_test$mileage)
```

```
## [1] 52044.68
```

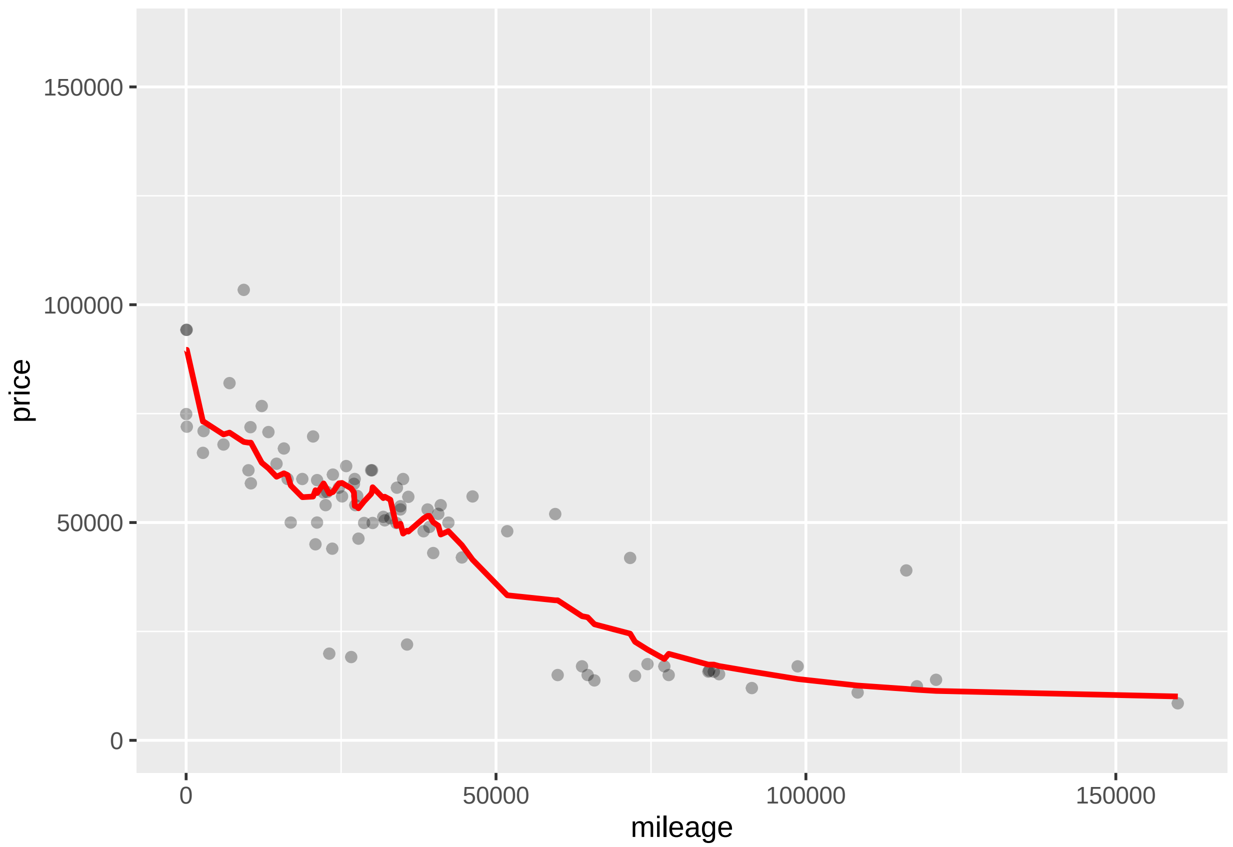
```
plot(model)
```



```
#FOR THE OPTIMAL LEVEL OF K:

knn13=knnreg(price~mileage, data=sclass_train, k=13)
rmse(knn13, sclass_test)
```

```
## [1] 11188.99
```

```
price_pred=predict(knn15, sclass_test)

sclass_t10 = sclass_test %>%
  mutate(price_pred)

p_test=ggplot(data=sclass_t10)+geom_point(mapping=aes(x=mileage, y=price), alpha=0.3)+ylim(500,160000)
p_test +geom_line(aes(x=mileage, y=price_pred), color='red', size=1)
```



QUESTION *Which trim yields a larger optimal value of K? Why do you think this is?*

Controlling for the training set we see that the trim level of 65 AMG yields a larger optimal value of K. This is because this trim level has a larger estimation variance and/or a larger estimation bias than the 350 trim level. This trim level, with respect to the 350 trim level, tends to pay little attention to the training data, causing an oversimplification of the model. That can be seen from the smoother RMSE vs. K plot in the latter with respect to 65 AMG. On the other hand, the 350 trim has a lower K, indicating that it must have either a smaller estimation bias or a smaller estimation variance than trim level 65 AMG.