

Predicting House Prices

Oliver Tomondy, Friedrich Winkelbauer

28/12/2021

Contents

1	Ziele	1
2	Libraries	2
3	Datenaufbereitung	2
4	Explorative Datenanalyse	3
4.0.1	Überblick	4
4.0.2	Zustand der Häuser nach Baujahr	5
4.0.3	Verteilung der Preise	6
4.0.4	Durschnittlicher Preis der Häuser nach Baujahr	7
4.0.5	Preis der Häuser nach Wohnfläche und Zustand	8
4.0.6	Verteilung der Preise nach Anzahl der Stockwerke	9
4.0.7	Verteilung der Preise nach Anzahl der Zimmer	10
4.0.8	Preis der Häuser nach Lage	11
4.0.9	Korrelation der einzelnen Merkmale mit Preis	13
5	Modellierung	14
5.1	Datenvorbereitung	14
5.2	Linear Regression	14
5.3	Random Forest	15
5.4	Neural Network	15
5.5	Ergebnisse	16

1 Ziele

- Das Ziel dieser Arbeit ist es, einen Datensatz mit Seattle Häuser zu analysieren und mit verschiedenen Machine Learning Modellen den Preis der Häuser in der Stadt Seattle vorherzusagen.
- Anschließend wird das beste Modell auch als Webservice deployed.
- Die Dashboard in `dashboard/dashboard.html` fasst die wichtigsten Erkenntnisse zusammen.

2 Libraries

```
#install.packages("corrplot")
#install.packages(c("cowplot", "ggraph", "rnaturalearth", "rnaturalearthdata"))
#install.packages("Metrics")
library(zoo, quietly = TRUE)
library(corrplot, quietly = TRUE)
library(tidyverse, quietly = TRUE)
library(tidygraph, quietly = TRUE)
library(igraph, quietly = TRUE)
library(ggplot2, quietly = TRUE)
library(ggraph, quietly = TRUE)
library(rnaturalearth, quietly = TRUE)
library(rnaturalearthdata, quietly = TRUE)
library(caret, quietly = TRUE)
library(randomForest, quietly = TRUE)
library(nnet, quietly = TRUE)
library(e1071, quietly = TRUE)
library(gbm, quietly = TRUE)
library(Metrics, quietly = TRUE)
```

3 Datenaufbereitung

Zuerst lesen wir die Daten ein. Wir verwenden dafür `read_delim` anstatt `read_csv` um den Spaltentyp zu schätzen.

```
data = read_delim("data/house_sales.csv", delim=",")
data = data %>% as_tibble()
```

Unser Datensatz enthält keine fehlenden oder infiniten Werte.

```
apply(data, 2, function(x) any(is.na(x) | is.infinite(x)))
```

##	id	date	price	bedrooms	bathrooms
##	FALSE	FALSE	FALSE	FALSE	FALSE
##	sqft_living	sqft_lot	floors	waterfront	view
##	FALSE	FALSE	FALSE	FALSE	FALSE
##	condition	grade	sqft_above	sqft_basement	yr_built
##	FALSE	FALSE	FALSE	FALSE	FALSE
##	yr_renovated	zipcode	lat	long	sqft_living15
##	FALSE	FALSE	FALSE	FALSE	FALSE
##	sqft_lot15				
##	FALSE				

4 Explorative Datenanalyse

Unsere Datensatz enthält Informationen über 21.613 Häuser in der US-amerikanischen Stadt Seattle. Jedes Haus ist durch eine ID gekennzeichnet und ist durch 19 Merkmale beschrieben. Unten findet man einen Überblick dieser Merkmale.

```
summary(data)
```

```
##           id           date           price
## Length:21613      Min.   :2014-05-02 00:00:00      Min.   : 75000
## Class :character  1st Qu.:2014-07-22 00:00:00      1st Qu.: 321950
## Mode  :character  Median :2014-10-16 00:00:00      Median : 450000
##                               Mean  :2014-10-29 04:38:01      Mean   : 540088
##                               3rd Qu.:2015-02-17 00:00:00      3rd Qu.: 645000
##                               Max.   :2015-05-27 00:00:00      Max.   :7700000
## bedrooms      bathrooms      sqft_living      sqft_lot
## Min.   : 0.000      Min.   :0.000      Min.   : 290      Min.   : 520
## 1st Qu.: 3.000      1st Qu.:1.750      1st Qu.: 1427      1st Qu.: 5040
## Median : 3.000      Median :2.250      Median : 1910      Median : 7618
## Mean   : 3.371      Mean   :2.115      Mean   : 2080      Mean   : 15107
## 3rd Qu.: 4.000      3rd Qu.:2.500      3rd Qu.: 2550      3rd Qu.: 10688
## Max.   :33.000      Max.   :8.000      Max.   :13540      Max.   :1651359
## floors      waterfront      view      condition
## Min.   :1.000      Min.   :0.000000      Min.   :0.0000      Min.   :1.000
## 1st Qu.:1.000      1st Qu.:0.000000      1st Qu.:0.0000      1st Qu.:3.000
## Median :1.500      Median :0.000000      Median :0.0000      Median :3.000
## Mean   :1.494      Mean   :0.007542      Mean   :0.2343      Mean   :3.409
## 3rd Qu.:2.000      3rd Qu.:0.000000      3rd Qu.:0.0000      3rd Qu.:4.000
## Max.   :3.500      Max.   :1.000000      Max.   :4.0000      Max.   :5.000
## grade      sqft_above      sqft_basement      yr_built
## Min.   : 1.000      Min.   : 290      Min.   : 0.0      Min.   :1900
## 1st Qu.: 7.000      1st Qu.:1190      1st Qu.: 0.0      1st Qu.:1951
## Median : 7.000      Median :1560      Median : 0.0      Median :1975
## Mean   : 7.657      Mean   :1788      Mean   : 291.5      Mean   :1971
## 3rd Qu.: 8.000      3rd Qu.:2210      3rd Qu.: 560.0      3rd Qu.:1997
## Max.   :13.000      Max.   :9410      Max.   :4820.0      Max.   :2015
## yr_renovated      zipcode      lat      long
## Min.   : 0.0      Min.   :98001      Min.   :47.16      Min.   : -122.5
## 1st Qu.: 0.0      1st Qu.:98033      1st Qu.:47.47      1st Qu.: -122.3
## Median : 0.0      Median :98065      Median :47.57      Median : -122.2
## Mean   : 84.4      Mean   :98078      Mean   :47.56      Mean   : -122.2
## 3rd Qu.: 0.0      3rd Qu.:98118      3rd Qu.:47.68      3rd Qu.: -122.1
## Max.   :2015.0      Max.   :98199      Max.   :47.78      Max.   : -121.3
## sqft_living15      sqft_lot15
## Min.   : 399      Min.   : 651
## 1st Qu.:1490      1st Qu.: 5100
## Median :1840      Median : 7620
## Mean   :1987      Mean   : 12768
## 3rd Qu.:2360      3rd Qu.: 10083
## Max.   :6210      Max.   :871200
```

4.0.1 Überblick

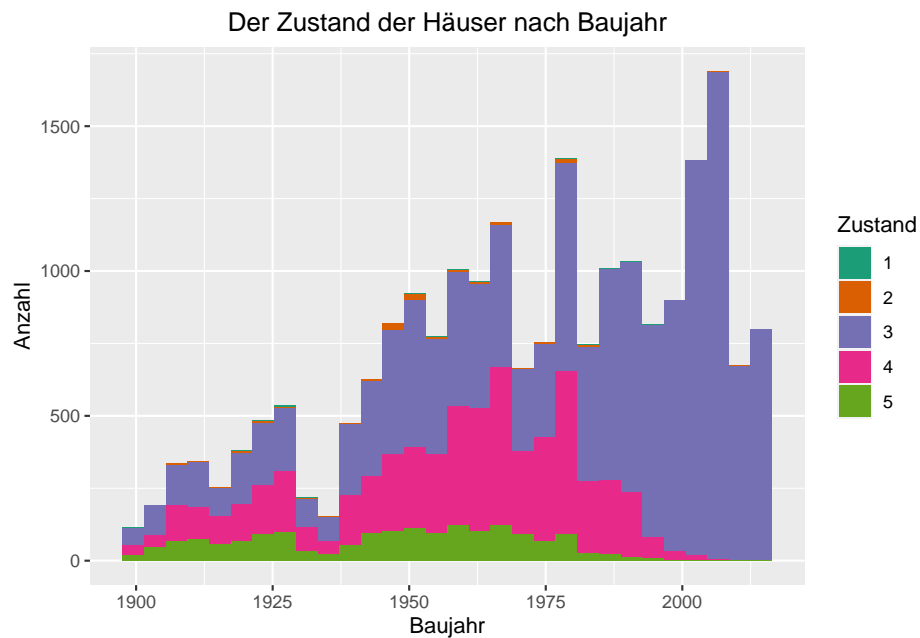
- Der durchschnittliche Preis eines Hauses im Datensatz beträgt 540.084 US-Dollar. Das teuerste Haus kostet 7.7 Millionen US-Dollar.
- Die Wohnfläche beträgt durchschnittlich 2.080 Quadraftfuß, was ca. 193 Quadratmeter ist.
- Die Median Größe eines Grundstücks beträgt 7.618 Quadratfuß, wobei das größte Grundstück 1.651.359 Quadratfuß hat.
- Die Häuser in unserem Datensatz haben außerdem durchschnittlich 3.4 Zimmer und 2.25 Badezimmer.
- Von Mehr als 20 Tausend Häuser liegen nur 163 am Wasser.
- Das älteste Haus wurde im Jahr 1900 gebaut. Der durchschnittliche Alter der Häuser im Datensatz beträgt 50 Jahre.

Schauen wir uns nun weitere Statistiken graphisch an. Da das Ziel dieser Arbeit die Erstellung mehrerer Modelle für die Vorhersage der Hauspreise ist, wird der Fokus dieser visuellen Datenanalyse auf der Variable **Preis** liegen.

4.0.2 Zustand der Häuser nach Baujahr

```
data %>%  
  ggplot(aes(x=yr_built, fill=as.factor(condition))) +  
  geom_histogram() +  
  ggtitle("Der Zustand der Häuser nach Baujahr") +  
  xlab("Baujahr") +  
  ylab("Anzahl") +  
  scale_fill_brewer(palette = "Dark2") +  
  theme(plot.title = element_text(hjust = 0.5)) +  
  labs(fill="Zustand")
```

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

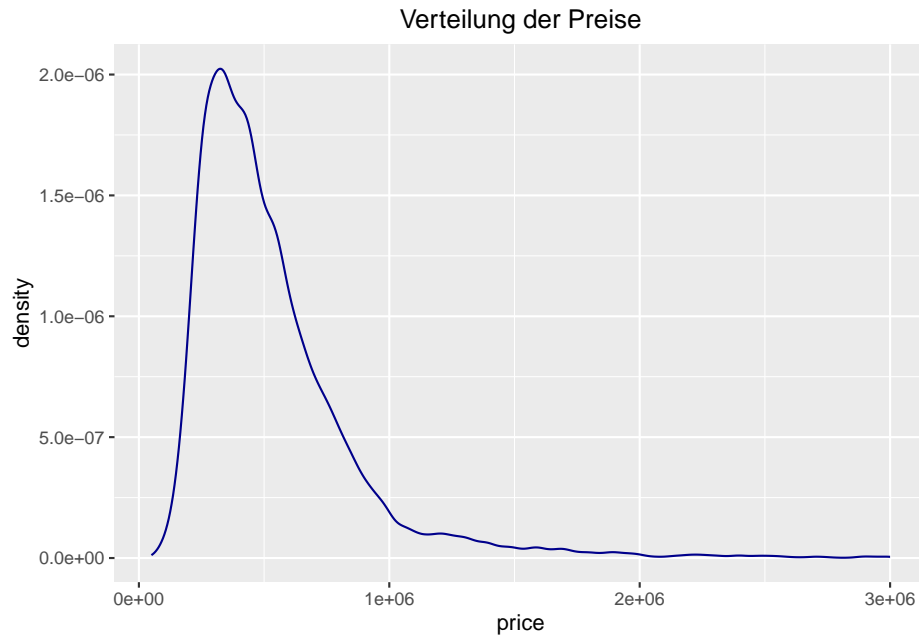


- Das Diagramm zeigt den Zustand der Häuser nach Baujahr.
- Es ist zu beobachten, dass die meisten Häuser im durchschnittlichen Zustand sind.
- Zudem ist die Mehrheit der Häuser zum Kauf ein Neubau, gebaut in den letzten 50 Jahren.

4.0.3 Verteilung der Preise

```
data %>%
  ggplot(aes(x=price)) +
  geom_line(stat="density", color="darkblue") +
  xlim(50000, 3000000) +
  ggtitle("Verteilung der Preise") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 45 rows containing non-finite values (stat_density).
```



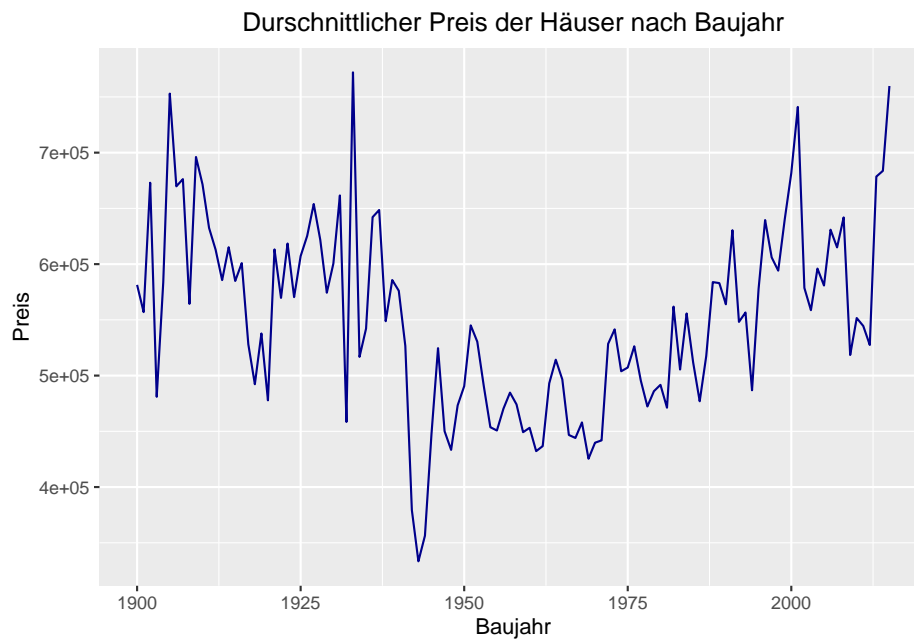
- Auf dem Diagramm sehen wir die Verteilung der Preise für Häuser in unserem Datensatz.
- Die Verteilung folgt einer ungefähren F-Verteilung.
- Die Mehrheit der Häuser kostet zwischen 320.000 und 645.000 US-Dollar.

```
summary(data$price)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  75000   321950   450000   540088   645000  7700000
```

4.0.4 Durchschnittlicher Preis der Häuser nach Baujahr

```
data %>%
  ggplot(aes(x=yr_built,y=price)) +
  geom_line(stat = "summary", fun = "mean", color="darkblue") +
  ggtitle("Durschnittlicher Preis der Häuser nach Baujahr") +
  xlab("Baujahr") + ylab("Preis") +
  scale_fill_brewer(palette = "Dark2") +
  theme(plot.title = element_text(hjust = 0.5))
```

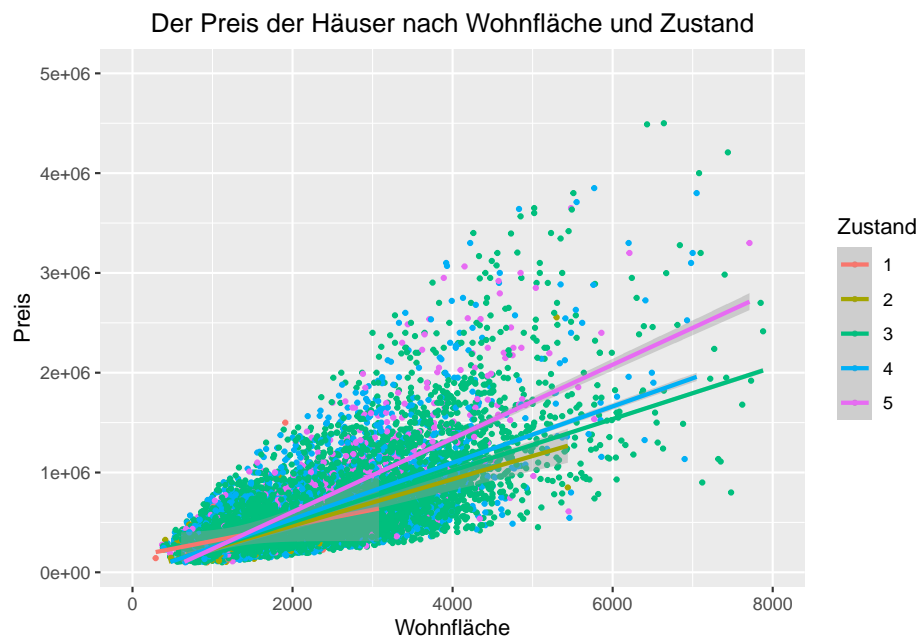


- Auf dem Diagramm sehen wir den durchschnittlichen Verkaufspreis der Häuser nach dem Baujahr.
- Auffällig ist, dass Häuser, die zwischen 1900 und 1930 gebaut wurden, durchschnittlich einen höheren Preis haben als Häuser, die zwischen den Jahren 1945 und 1980 gebaut wurden. Erst ganz junge Häuser, die am Ende des 20. Jahrhunderts und am Anfang des 21. Jahrhunderts gebaut wurden, sind wieder teurer.

4.0.5 Preis der Häuser nach Wohnfläche und Zustand

```
data %>%
  ggplot(aes(x=sqrt_living,y=price, colour =as.factor(condition))) +
  geom_point(size=0.8) +
  ggtitle("Der Preis der Häuser nach Wohnfläche und Zustand") +
  xlab("Wohnfläche") +
  ylab("Preis") +
  scale_fill_brewer(palette = "Dark2") +
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_smooth(method=lm) +
  ylim(100000, 5000000) +
  xlim(0, 8000) +
  labs(color="Zustand")
```

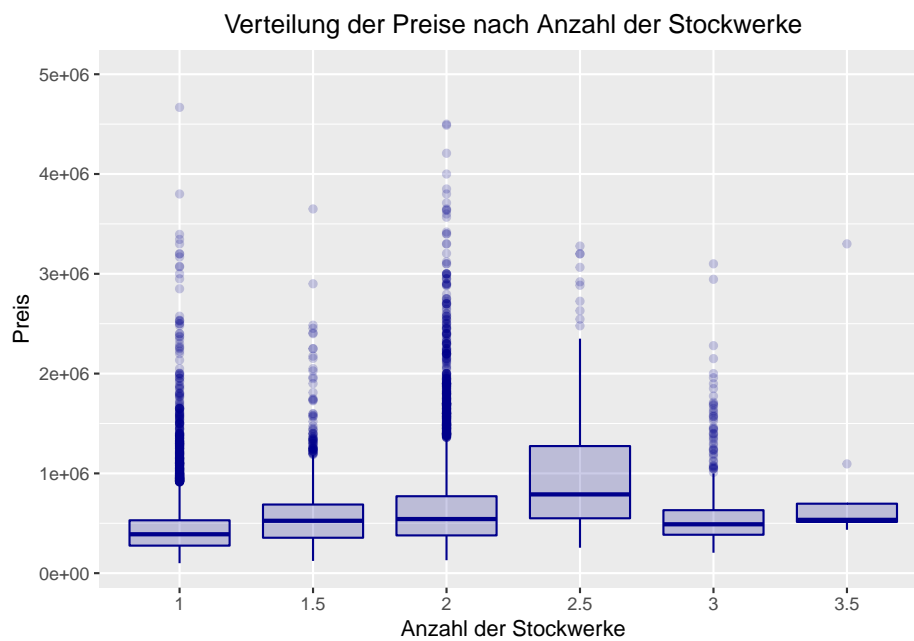
'geom_smooth()' using formula 'y ~ x'



- Das Diagramm zeigt den Preis der Häuser nach Wohnfläche und Zustand.
- Es lässt sich deutlich erkennen, dass mit steigender Wohnfläche auch der Preis für ein Haus steigt.
- Zudem lässt es sich anhand von Trendgeraden erkennen, dass der Preis für Häuser in einem besseren Zustand steiler ansteigt, als Preis für Häuser in einem schlechteren Zustand.

4.0.6 Verteilung der Preise nach Anzahl der Stockwerke

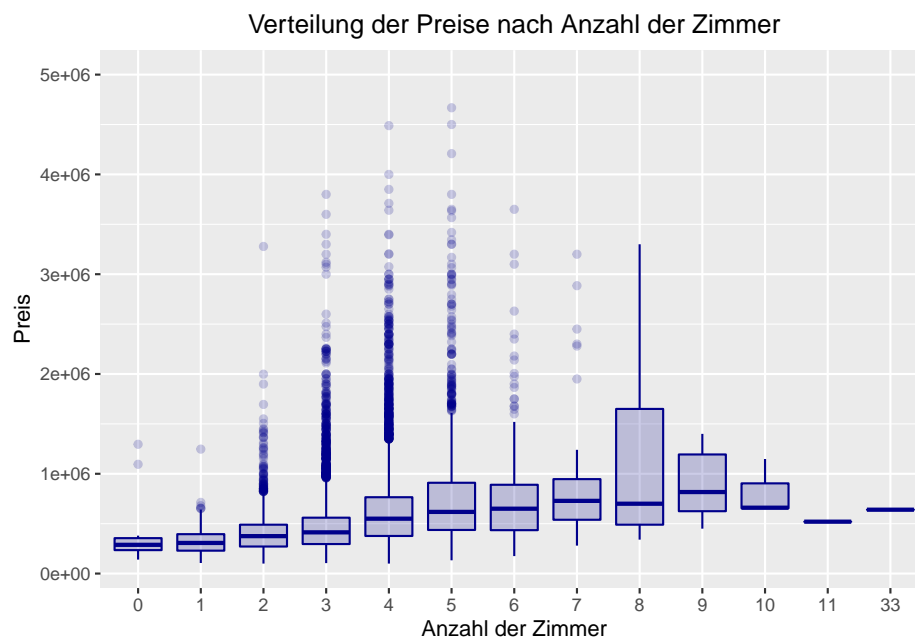
```
data %>%
  ggplot(aes(x=as.factor(floors), y=price)) +
  geom_boxplot(color="darkblue", fill="darkblue", alpha=0.2) +
  ylim(100000, 5000000) +
  ggtitle("Verteilung der Preise nach Anzahl der Stockwerke") +
  xlab("Anzahl der Stockwerke") + ylab("Preis") +
  scale_colour_brewer(palette = "Dark2") +
  theme(plot.title = element_text(hjust = 0.5))
```



- Das Diagramm zeigt die Verteilung der Hauspreise nach der Anzahl der Stockwerke.
- Es lässt sich erkennen, dass je mehr Stockwerke das Haus hat, desto höher der Preis ist. Interessenterweise gilt dieser Trend nur bis zu 2.5 Stockwerken. Wenn ein Haus 3 oder 3.5 Stockwerke hat, ist der Preis durchschnittlich niedriger als bei Häusern mit nur 2.5 Stockwerken.
- Wir können zudem viele Ausreißer nach oben beobachten. Das könnte sich in Vorhersagemodellen negativ auf die Performance auswirken. (RMSE wird deutlich größer sein als MAE)
- Vielleicht lässt sich eine deutlichere Tendenz bei der Anzahl der Zimmer feststellen.

4.0.7 Verteilung der Preise nach Anzahl der Zimmer

```
data %>%
  ggplot(aes(x=as.factor.bedrooms), y=price)) +
  geom_boxplot(color="darkblue", fill="darkblue", alpha=0.2) +
  ylim(100000, 5000000) +
  ggtitle("Verteilung der Preise nach Anzahl der Zimmer") +
  xlab("Anzahl der Zimmer") + ylab("Preis") +
  scale_colour_brewer(palette = "Dark2") +
  theme(plot.title = element_text(hjust = 0.5))
```



- Auf der Grafik können wir die Verteilung der Hauspreise nach der Anzahl der Zimmer beobachten.
- Es gibt einen klaren aufsteigenden Trend: Also je mehr Zimmer ein Haus hat, desto mehr wird er wahrscheinlich kosten.
- Wir können zudem viele Ausreißer nach oben beobachten. Das könnte sich in Vorhersagemodellen negativ auf die Performance auswirken. (RMSE wird deutlich größer sein als MAE)
- Offensichtlich gibt es in unserem Datensatz einen Ausreißer, der wahrscheinlich nur eine Fehleingabe war. Ein Haus mit 33 Zimmer und Wohnfläche nur 1620 Quadratfuß. Wir entfernen es aus dem Datensatz.

```
data %>% subset(bedrooms == 33)
```

```
## # A tibble: 1 x 21
##   id    date                price bedrooms bathrooms sqft_living sqft_lot floors
##   <chr> <dtm>                <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1 2402~ 2014-06-25 00:00:00 6.4e5      33        1.75      1620      6000      1
```

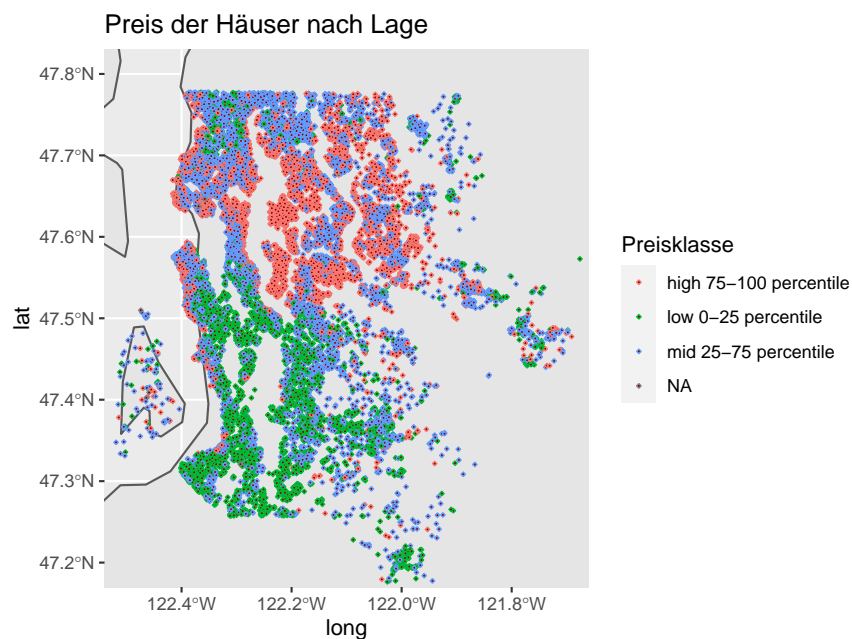
```
## # ... with 13 more variables: waterfront <dbl>, view <dbl>, condition <dbl>,
## #   grade <dbl>, sqft_above <dbl>, sqft_basement <dbl>, yr_built <dbl>,
## #   yr_renovated <dbl>, zipcode <dbl>, lat <dbl>, long <dbl>,
## #   sqft_living15 <dbl>, sqft_lot15 <dbl>
```

```
data = data %>% subset(bedrooms != 33)
```

4.0.8 Preis der Häuser nach Lage

```
data = data %>% mutate(pricecat = case_when(
  price < 321950 ~ 'low 0-25 percentile',
  price < 645000 ~ 'mid 25-75 percentile',
  price > 645000 ~ 'high 75-100 percentile'
))

world <- ne_countries(scale = "medium", returnclass = "sf")
ggplot(data = world) +
  geom_sf() +
  geom_point(data = data, aes(x = long, y = lat, col=as.factor(pricecat)), size = 0.5,
    shape = 23, fill = "darkred") +
  ggtitle("Preis der Häuser nach Lage")+
  coord_sf(xlim = c(-122.5, -121.7), ylim = c(47.20, 47.8)) +
  labs(color="Preisklasse")
```

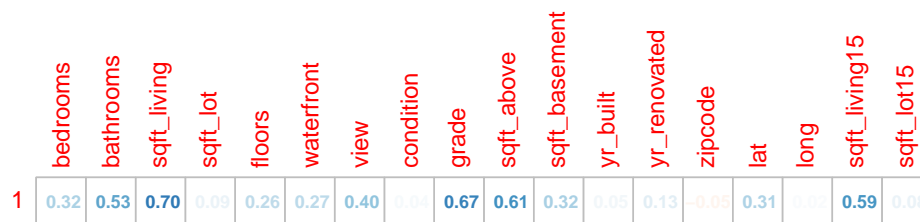


- Auf der geografischen Karte können wir die Lage der Häuser im Datensatz sehen, gefärbt nach Preisklasse.

- Wir können beobachten, dass die teuersten Häuser (rot) näher zum Stadtzentrum liegen, als billigere Häuser (grün). Häuser, die preismäßig in der Mitte liegen (blaue), sind in der Stadt ungefähr regelmäßig verteilt.

4.0.9 Korrelation der einzelnen Merkmale mit Preis

```
datacor = data %>% select(-c("id", "date", "pricecat"))
corrplot(cor(datacor$price, datacor), method="number", diag = FALSE, tl.cex = 1,
          number.cex=0.75, cl.pos = "n")
```



- Als letztes schauen wir die Korrelation einzelner Merkmale mit dem Preis.
- Wir können beobachten, dass die stärkste Korrelation mit dem Preis die Merkmale: Wohnfläche, Grade, Anzahl der Badezimmer und Aussicht gegeben ist.
- Fast keine oder nur eine sehr schwache Korrelation mit dem Preis haben die Merkmale: Zustand, Baujahr, Renovierungsjahr, ZIP-Kode, Longitude und Latitutde.

5 Modellierung

- Zuerst wird ein fester Seed gesetzt, sodass die Ergebnisse gleich bleiben.

```
set.seed(1500)
```

- Da unsere Modelle den Preis der Häuser vorhersagen sollen, handelt es sich um **Regression** und daher werden Metriken
 - **MAE** (Mean Absolute Error),
 - **RMSE** (Root Mean Squared Error),
 - **MAPE** (Mean Absolute Percentage Error),
 - **R2** Score gemessen.

5.1 Datenvorbereitung

- Wegen der Erkenntnisse aus der Korrelationsanalyse werden einige Spalten aus den Daten entfernt.
- Zusätzlich werden unnötige Spalten ('id', 'date') entfernt.

```
data = data %>% select(-c(id,date,condition,zipcode,lat,long,pricecat,sqft_lot, condition,yr_built,yr_r
```

Die Daten werden im nächsten Schritt skaliert. (außer der Ziel-Variable -> Bewusste Entscheidung, die Modelle werden dadurch nicht schlechter, und wir müssen MAE, RSME, sowie API Antworten nicht zurückskalieren.)

```
orig = data
data = orig
price = data$price;
data = scale(data %>% select(-price));
# Wir speichern die Skalierungen, damit wir im Web Service unskalierte Daten als Input haben können
scaled_center = attr(data, 'scaled:center')
scaled_scale = attr(data, 'scaled:scale')
save(scaled_center, file ="webservice/scaled_center.rda")
save(scaled_scale, file ="webservice/scaled_scale.rda")

data = cbind(data, price)
data = as_tibble(data)
```

Die Daten werden zuerst in Test- und Trainingsdaten aufgeteilt.

```
part = createDataPartition(data$price, times = 2, p = 4/5)
train = data[train$Resample1,]
test = data[-part$Resample1,]
```

5.2 Linear Regression

Das erste Modell, das verwendet wird, ist eine lineare Regression.

```

model_r_linearModel = lm(price ~ . , data = train)
pred_r_linearModel = predict(model_r_linearModel, test)
stats_r_linearModel = data.frame(
  rmse = round(rmse(test$price, pred_r_linearModel),2),
  mae = round(mae(test$price, pred_r_linearModel),2),
  mape = round(mape(test$price, pred_r_linearModel),2)*100,
  r2_squared = round(summary(model_r_linearModel)$r.squared,2))
stats_r_linearModel

```

```

##          rmse          mae mape r2_squared
## 1 222890.5 150943.1   31         0.6

```

5.3 Random Forest

Die zweite Methode ist ein Random Forest und wird einmal mit 101 Bäumen und einmal mit 501 Bäumen ausgeführt, um die Auswirkung der Erhöhung zu sehen.

```

model_r_randomForest = randomForest(price ~ ., data = train, ntrees=101)
pred_r_randomForest = predict(model_r_randomForest, test)
stats_r_randomForest = data.frame(
  rmse = round(rmse(test$price, pred_r_randomForest),2),
  mae = round(mae(test$price, pred_r_randomForest),2),
  mape = round(mape(test$price, pred_r_randomForest),2)*100,
  r2_squared = round(mean(model_r_randomForest$rsq),2))
stats_r_randomForest

```

```

##          rmse          mae mape r2_squared
## 1 198735.1 125189    26         0.71

```

```

#model_r_randomForest1 = randomForest(price ~ ., data = train, ntrees=501)
#pred_r_randomForest1 = predict(model_r_randomForest, test)
#stats_r_randomForest1 = RMSE(test$price, pred_r_randomForest)
#Ergebnis: RSME=0.6

```

Da die höhere Anzahl der Bäume keine Verbesserung bewirkt, wird das mit 101 Bäumen erzeugte Modell weiterverwendet.

5.4 Neural Network

Das dritte Modell ist ein Neural Network. Wie beim Random Forest werden mehrere Settings durchprobiert:

```

model_r_nnet = nnet(price ~ ., data = train,
                    size = 100, MaxNWts = 10000, trace = FALSE, maxit = 100)
pred_r_nnet = predict(model_r_nnet, test)
stats_r_nnet = data.frame(
  rmse = round(rmse(test$price, pred_r_nnet),2),
  mae = round(mae(test$price, pred_r_nnet),2),
  mape = round(mape(test$price, pred_r_nnet),2)*100,
  r2_squared = round(cor(test$price, pred_r_nnet) ^ 2,2))

```

```
## Warning in cor(test$price, pred_r_nnet): the standard deviation is zero
```

```
stats_r_nnet
```

```
##           rmse           mae mape r2_squared  
## 1 648249.8 539502.2 100           NA
```

```
#model_r_nnet1 = nnet(price ~ ., data = train,  
#                      size = 200, MaxNWts = 10000, trace = FALSE, maxit = 200)  
#pred_r_nnet1 = predict(model_r_nnet, test)  
#stats_r_nnet1 = RMSE(test$price, pred_r_nnet)  
#Ergebnis: RMSE=1  
  
#model_r_nnet2 = nnet(price ~ ., data = train,  
#                      size = 500, MaxNWts = 15000, trace = FALSE, maxit = 500)  
#pred_r_nnet2 = predict(model_r_nnet, test)  
#stats_r_nnet2 = RMSE(test$price, pred_r_nnet)  
#Ergebnis: RSME=1
```

Da der Fehler für alle drei Varianten gleich ist, wird das ursprüngliche Modell beibehalten.

5.5 Ergebnisse

Vergleich der Modelle mittels MAE, RMSE und R2 Squared Werten:

	MAE	MAPE	RMSE	R2-Squared
Lineare Regression	1.5094314×10^5	31	2.2289051×10^5	0.6
Random Forest	1.2518903×10^5	26	1.9873515×10^5	0.71
Neural Net	5.3950219×10^5	100	6.4824975×10^5	NA

- Der mittlere absolute Fehler ist bei allen Modellen ziemlich groß, bei randomForest aber nur um ca. 190.000 Tausend US-Dollar- also die mittlere Höhe der Abweichung der Vorhersage von der Beobachtung. Bei einem durchschnittlichen Preis der Häuser von 540.084 Tausend US-Dollar stellt es eine ziemlich große Abweichung dar (28%)
- Wie erwartet, verursachen viele (Preis-)Ausreißer einen viel höheren RMSE als MAE. Die Modelle könnten eventuell verbessert werden, wenn mehrere Faktoren in Betracht gezogen wären.
- Das beste Modell RandomForest erklärt mehr als 70% der Streuung (R2 Squared), was ziemlich gut ist.
- Unter der Annahme, dass nur die hier gezeigten Daten und Modelle zur Verfügung stehen, ist das beste Modell: Random Forest.

Das beste Modell wird abgespeichert:, damit es in Webservice verwendet werden kann.

```
save(model_r_randomForest,  
      file = "webservice/model.rda")
```