Statistical Methods - Final Project: House Prices - Regressing Sales Price

Donninelli Adriano Insaghi Edoardo Zappi Piero Zappia Edoardo

Introduction

This project aims to develop statistical models capable of predicting house sale prices (it's then a regression task) from a set of covariates describing every aspect of the property. To do it we leverage the "House Prices - Advanced Regression Techniques" dataset from Kaggle [1]. This dataset comprises 1460 samples of property sales situated in Ames, Iowa. For each one, 79 explanatory variables are collected, varying from the year of construction to the type of foundations used.

We begin by exploring the data and doing the necessary preprocessing steps. After that, we simplify the data using a dimensionality reduction technique (PCA) to gain insights into how the target variable relates to other factors. Next, we suggest up to four statistical models: LM, GAM, RF, and a basic NN. We didn't take into consideration generalised linear models such as the logistic regression or the Poisson regression because of the response variable's type. We fine-tune each model and compare their behaviours using cross-validation since there is no predefined validation dataset at disposal.

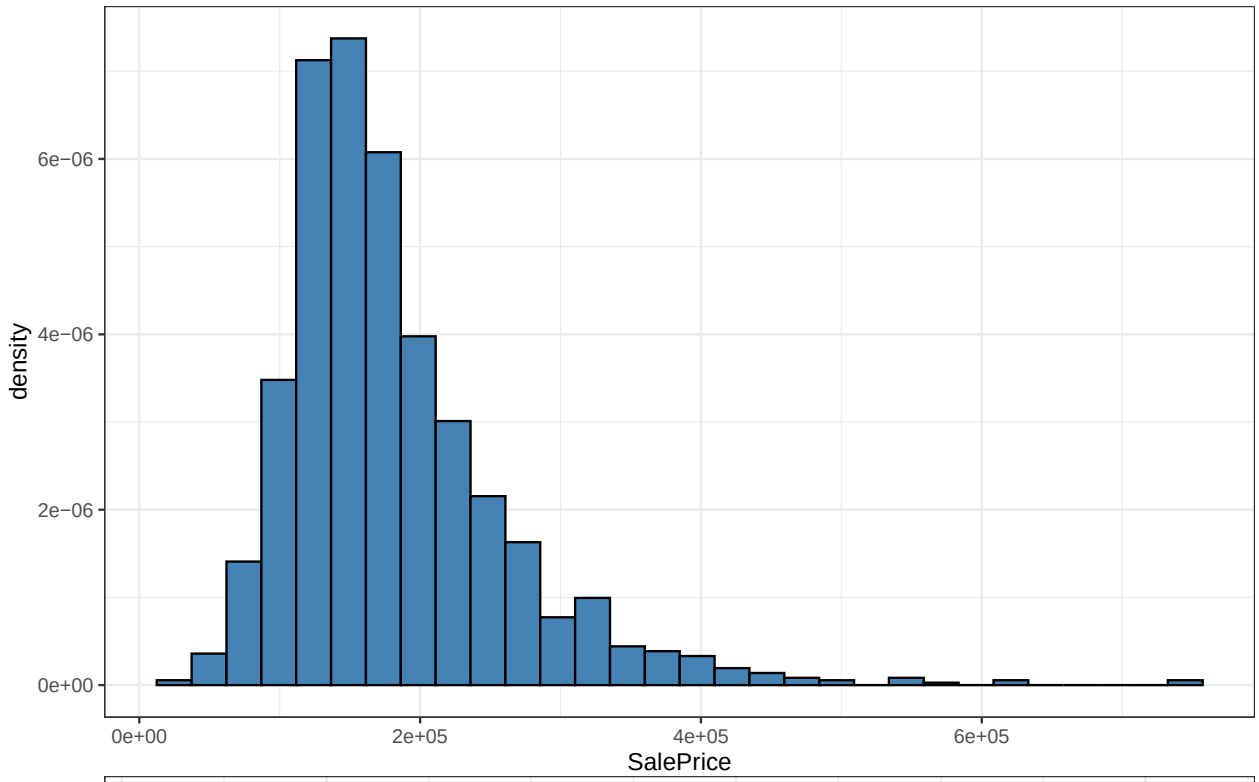
Data Exploration

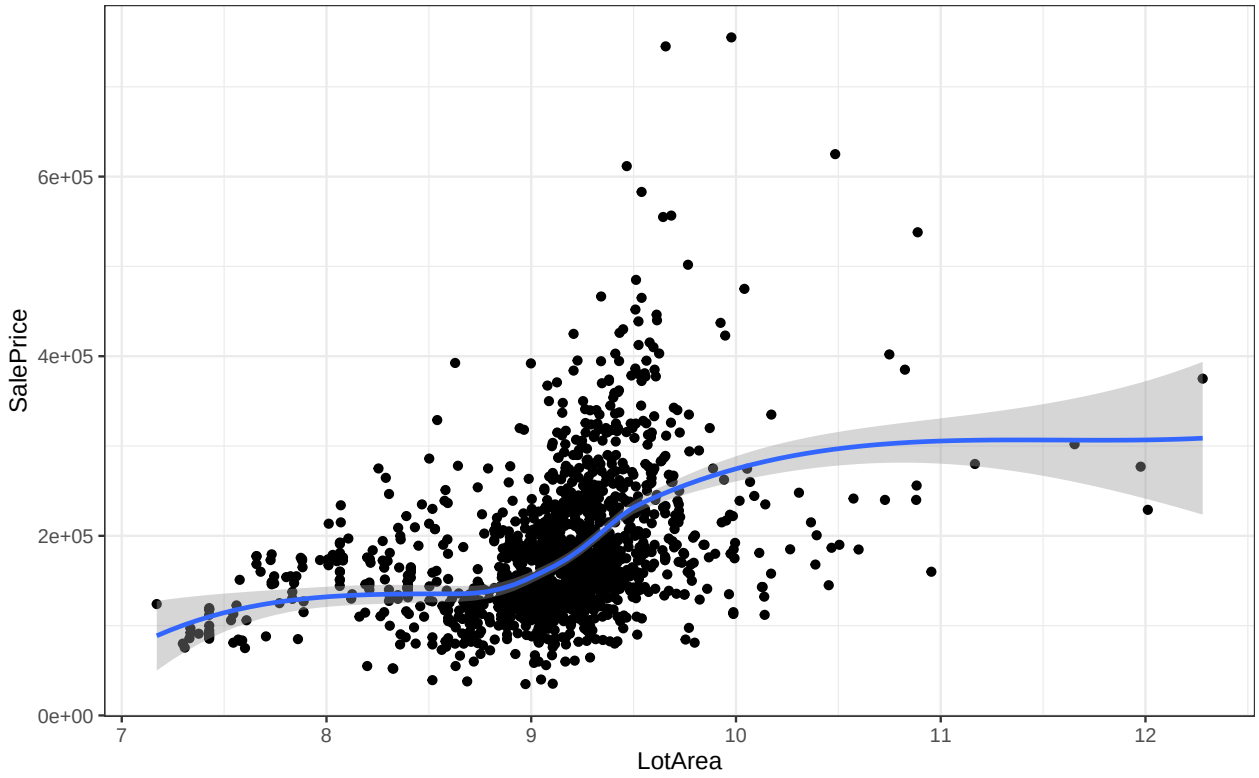
The first step is to explore the data. As mentioned in the introduction, the dataset consists of 1460 observations and 79 covariates, which is a substantial number of variables, and poses the challenge of selecting which ones play a role in determining the response variable, and which do not.

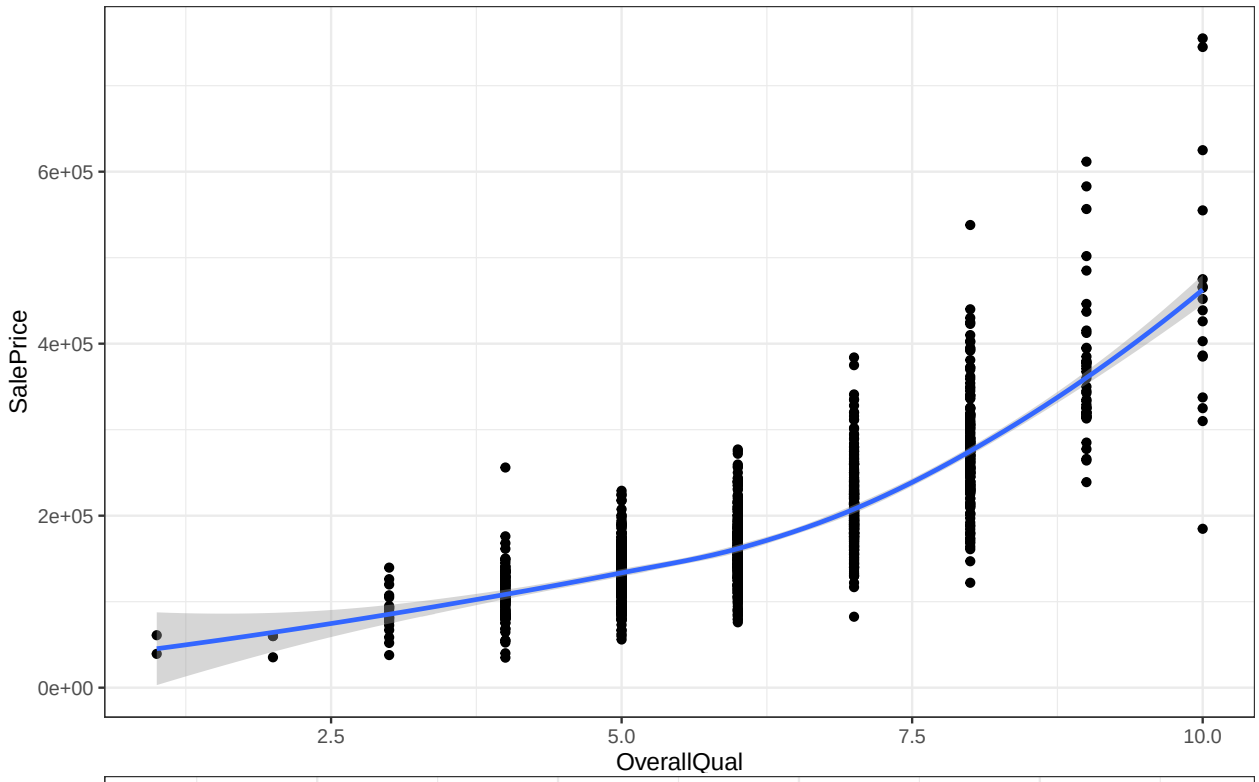
The preprocessing of the data consisted in a number of steps: we first decided to remove from the dataset all the variables with an excessive number of NAs and fill those which had a reasonable amount of NAs (<100) with the mean or the mode of the remaining observations, depending on whether such variables were numerical or categorical. We then proceeded to scale the extremely right skewed numerical variables using a logarithmic transformation, which helps normalizing the variables and making them more easily explainable. Lastly, given the substantial number of categorical variables with dozens, if not hundreds, of categories, we decided to dump the least occurring variables into a separate category. This procedure helps keeping the degrees of freedom under control and makes the models more easily explainable.

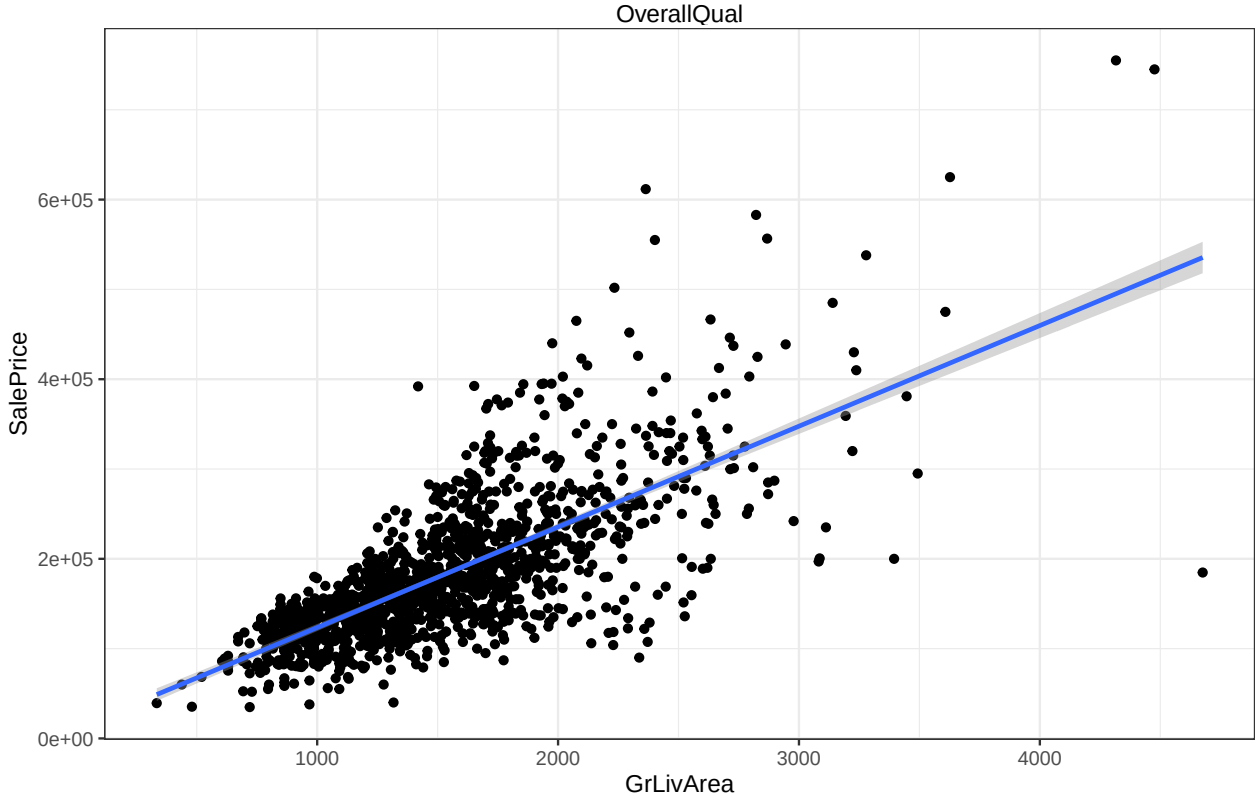
We also considered taking the logarithm of the response variable, which also appears to be right skewed, but this does not improve significantly the explanatory power of the models, and we therefore decided to leave it as it is preserving the interpretability of the models abiding by the principle expressed by the Occam's Razor.

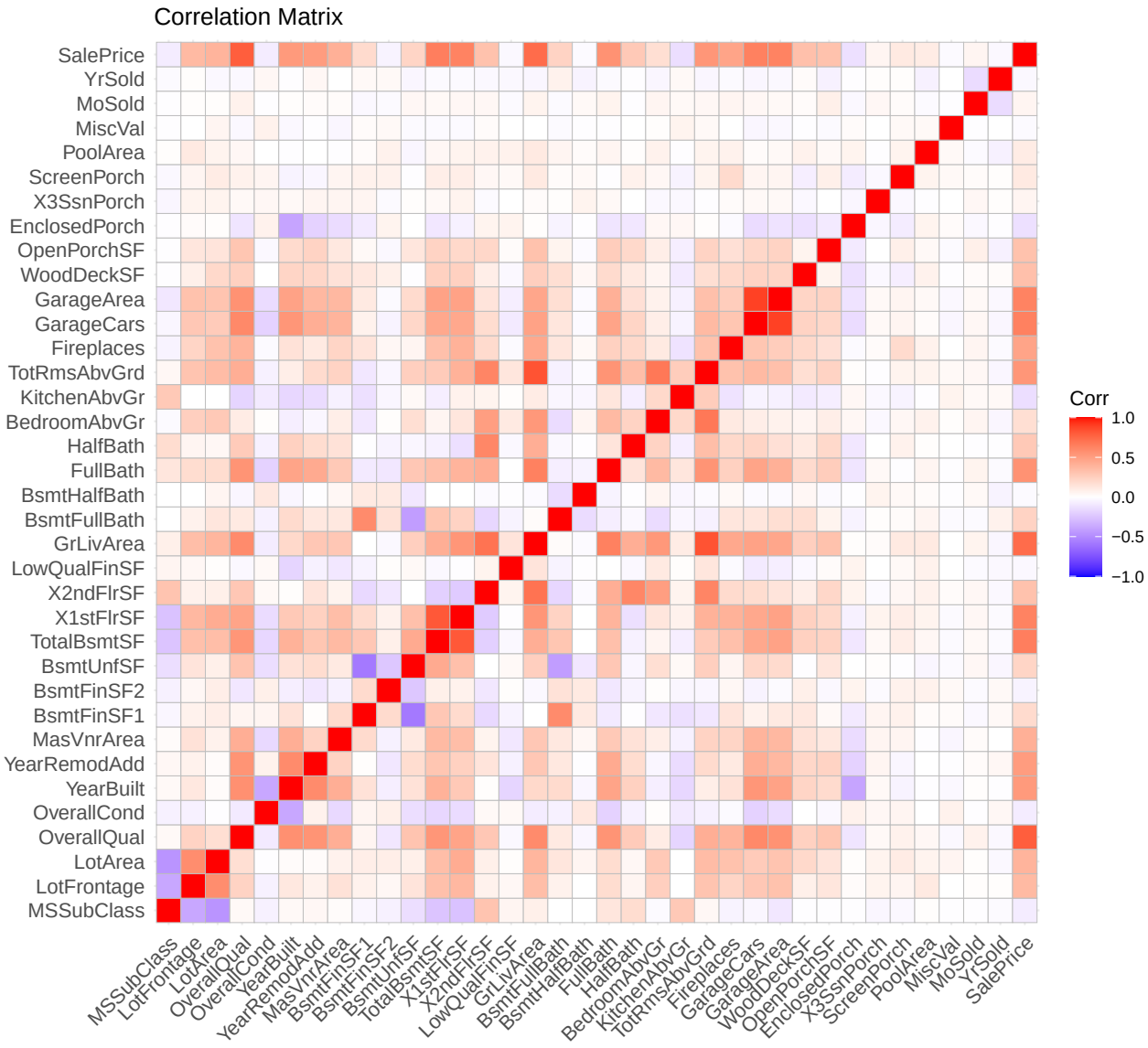
We observed the correlation matrix to explore how the numerical variables are related to each other and have a first look at how they are correlated to the response variable. We then show a few scatterplots of the response variables against some interesting covariate, looking for any sort of relationship with the house price.


density
6e−06
4e−06
2e−06
0e+00
0e+00
2e+05
4e+05
6e+05
SalePrice


SalePrice
6e+05
4e+05
2e+05
0e+00
7
8
9
10
11
12
LotArea


SalePrice
6e+05
4e+05
2e+05
0e+00
2.5
5.0
7.5
10.0
OverallQual


SalePrice
6e+05
4e+05
2e+05
0e+00
1000
2000
3000
4000
GrLivArea

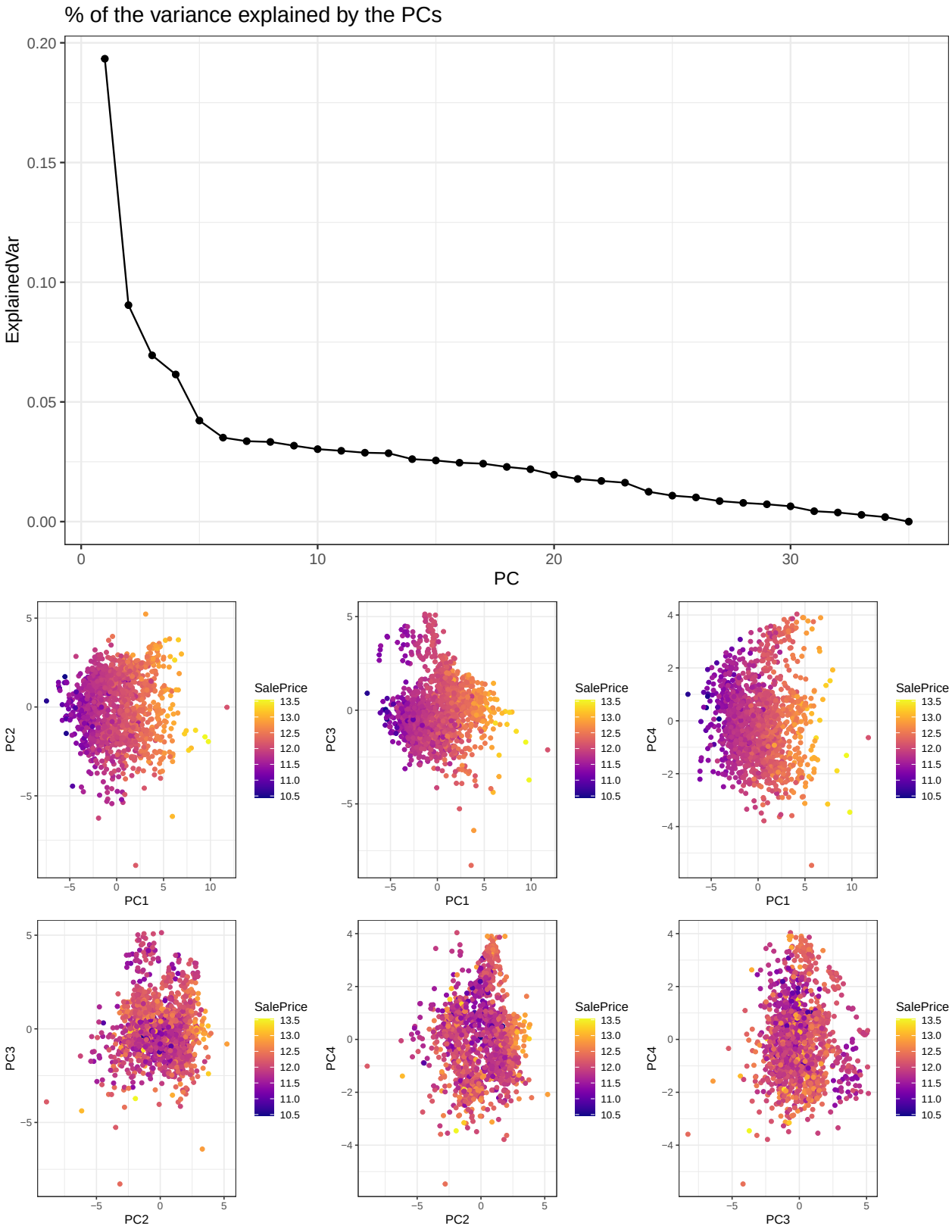


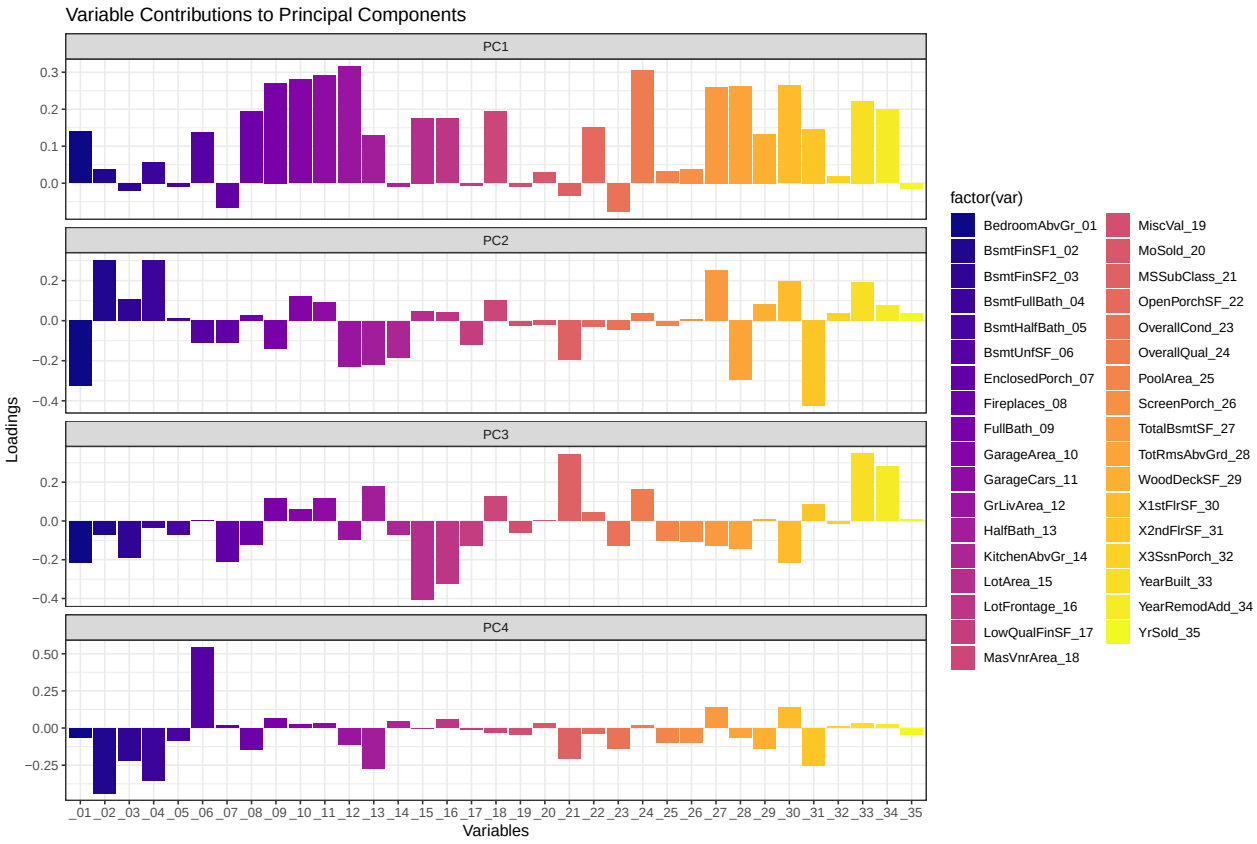
Dimensionality Reduction

When the number of covariates is high, it may be beneficial to put in place techniques that aim at reducing the dimensionality of the dataset, while preserving as much information as possible. We decided to use the principal component analysis, a linear dimensionality reduction tool that works by projecting the data onto the eigenvectors of the covariance matrix of the data, to see whether the dataset can be summarized with less variables.

The elbowplot in the figure represents the proportion of the variance explained by each principal component. The results are not extremely exciting, because ideally we would like to see the first few principal components explaining the majority of the variance in our data, which is not what is happening here.

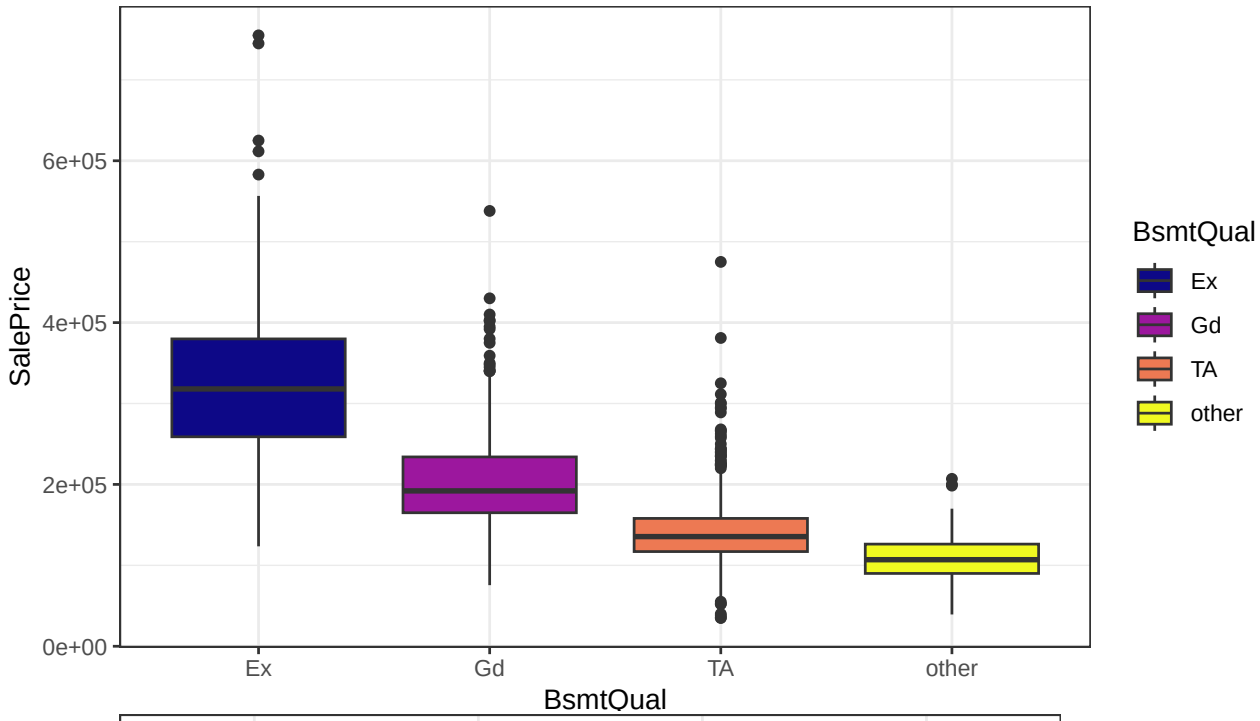
We then plotted the first four principal components against each other, highlighting the value of the Sale Price in the plots, looking for interesting patterns. Apparently, only the first principal component seems to explain something about the response variable. The last plot includes the loadings of the variables on the first few principal components, here we are particularly interested in which variables contribute noticeably to the first component. These variables have to be observed closely since they are likely to influence the price of the house.

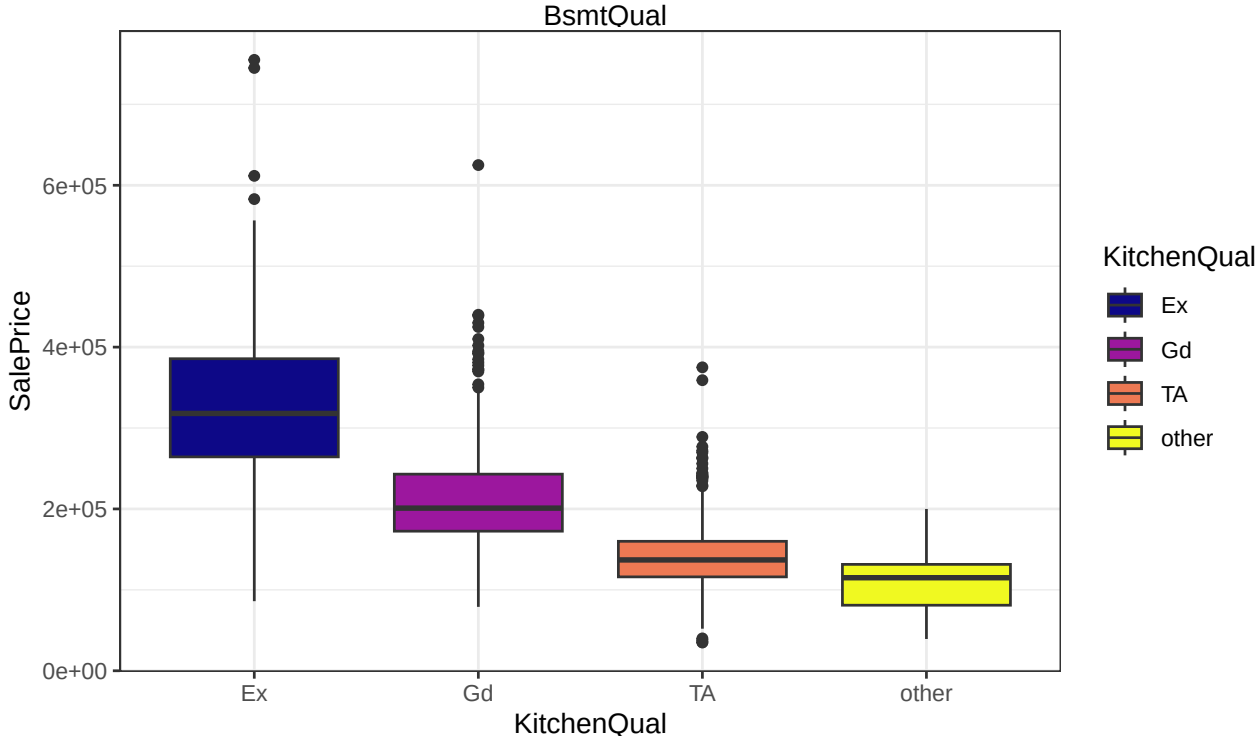

% of the variance explained by the PCs
ExplainedVar
PC
SalePrice
PC1
PC2
PC3
PC4

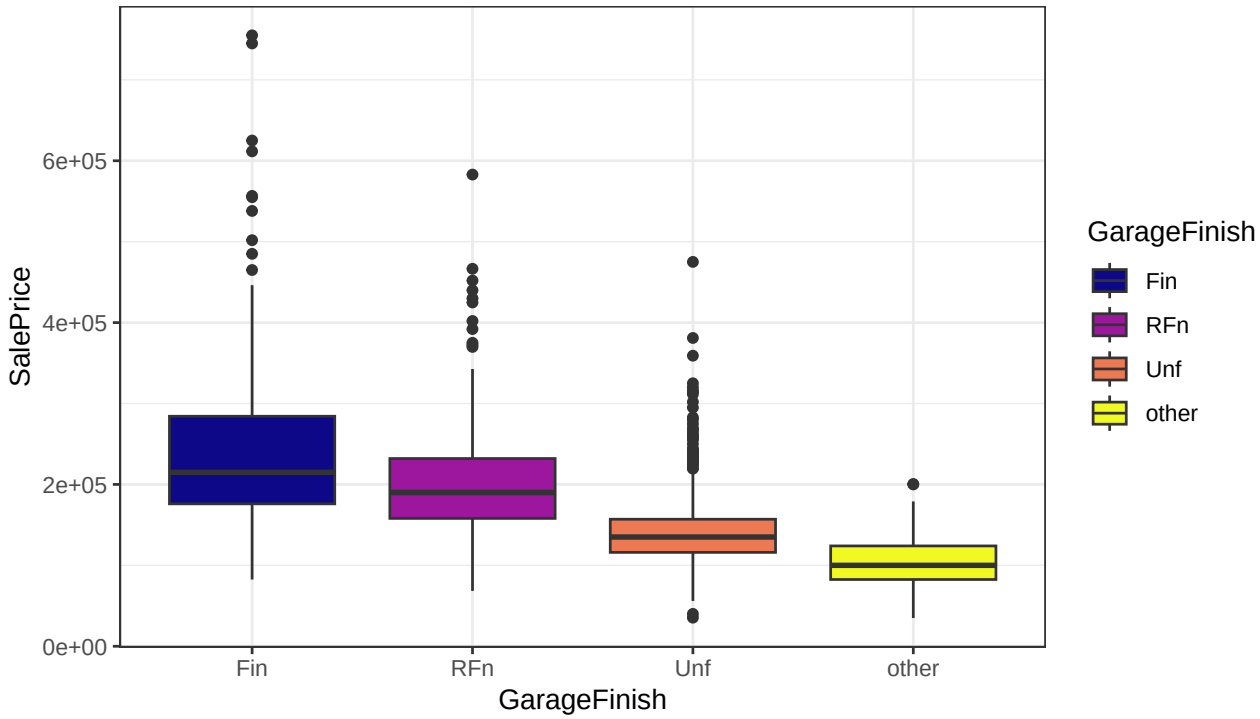


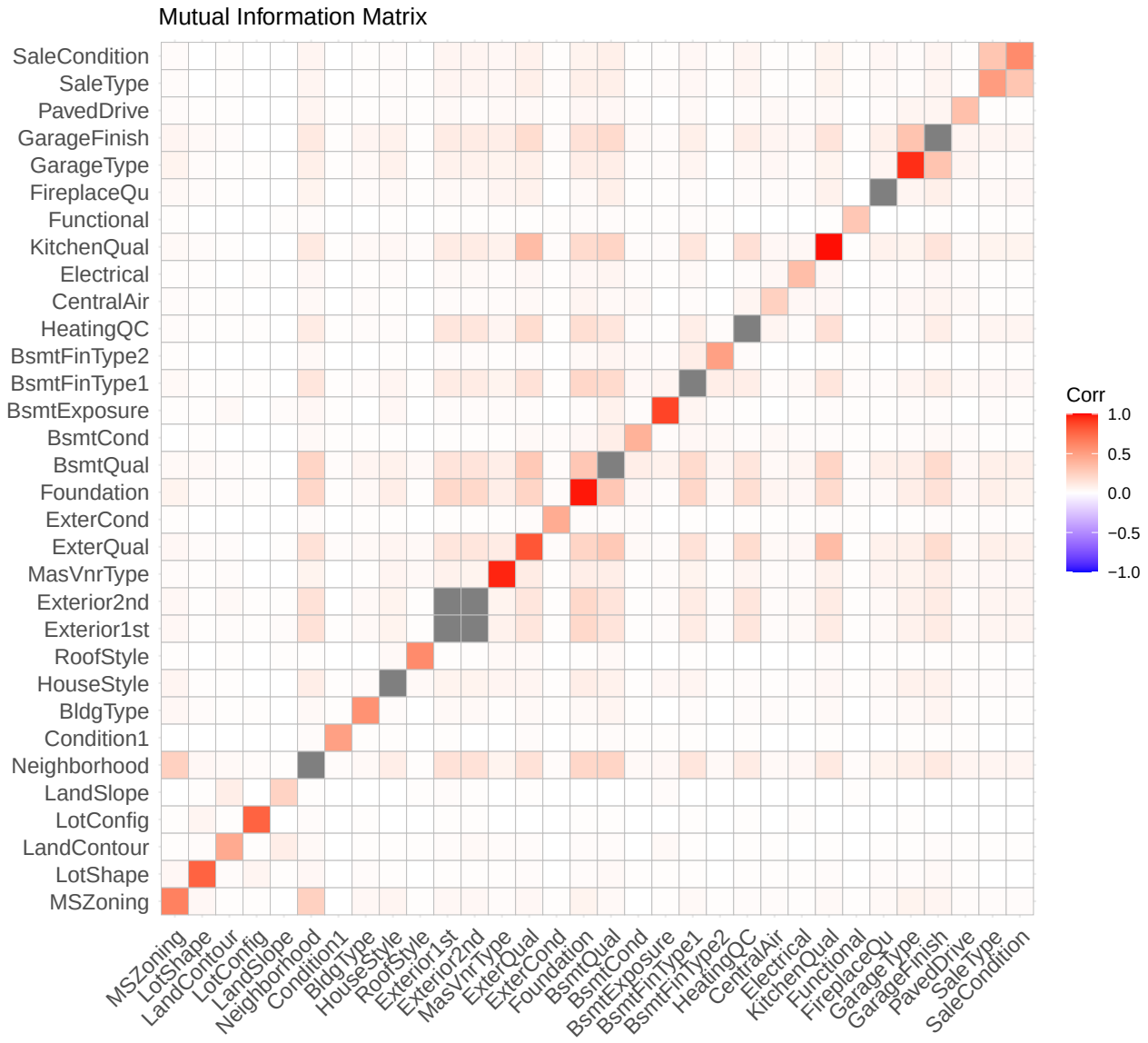
We concluded the exploratory analysis of the data by taking a look at the categorical variables. We plotted the boxplot of the Sale Price conditioned to every class of the categorical variables, looking for significant differences among them. For clarity we just decided to show the plots for some interesting covariates.

The last plot represents the mutual information matrix for the categorical covariates. It conveys information about the dependence between two variables, and we adopted it as a way of assessing correlation between these variables.


SalePrice
6e+05
4e+05
2e+05
0e+00
Ex
Gd
TA
other
BsmtQual
BsmtQual
Ex
Gd
TA
other


SalePrice
6e+05
4e+05
2e+05
0e+00
Ex
Gd
TA
other
KitchenQual
KitchenQual
Ex
Gd
TA
other


SalePrice
6e+05
4e+05
2e+05
0e+00
Fin
RFn
Unf
other
GarageFinish
GarageFinish
Fin
RFn
Unf
other



Linear Model

In this paragraph we discuss the building process of the linear model we employed to predict the response variable `SalePrice`.

The goal was to obtain a model showcasing a good trade-off between its performance and its complexity. In fact, since the dataset we had at disposal consisted of many different variables, the first task was to select among them the covariates to use for our model, by choosing the ones which would lead to better performances in terms of explained variability of the response variable (higher *adjusted* R^2). Moreover, we wanted the vast majority of the terms to be statistically significant and therefore, at the same time, we selected the covariates trying to avoid multicollinearity problems. Finally, in order to address the model's complexity, interpretability and to avoid overfitting, we simplified as much as possible the model's structure, while maintaining consistent results in terms of *adjusted* R^2.

In order to reduce from the beginning the risk of having multicollinearity problems, we first decided to remove from the dataset some variables which were highly correlated to/dependent from other variables; among a set of correlated variables, we kept in the dataset the one which was the most correlated to the response variable and therefore more useful to predict it. We then selected the covariates to remove by looking at

the correlation matrix (for the numerical variables) and the mutual information matrix (for the categorical variables).

Starting from a linear model containing all the remaining variables in the dataset as covariates (*adjusted* $R^2 = 0.8683$), we used the `stepAIC` function (part of the `MASS` package) to select the linear model composed by the best combination of covariates with respect to the AIC (Akaike Information Criterion).

The AIC is a measure that balances the goodness of fit of a model with its complexity, penalizing the models that are too complex: better fits correspond to smaller AIC values; the goal was to find a model that fitted the data well while avoiding overfitting.

The `stepAIC` function systematically adds or removes predictor variables from the model to find the combination that minimizes the AIC. The process involves iteratively fitting models, assessing their AIC, and deciding whether to add or remove variables based on the AIC improvement.

The `stepAIC` function returned the model which minimized the AIC: this model consisted of 30 covariates, so it was way less complex than the starting one; it showcased a good performance as well, since it had *adjusted* $R^2 = 0.8696$. We also computed the *variance inflation factor* (VIF) associated to each predictor to spot eventual collinearity and we obtained $VIF_j < 4 \; \forall j$ (way below the critical value 10). Finally, the ANOVA suggested to keep this simpler model instead of the initial more complex one.

Indeed, we obtained a list of 30 selected variables to use as covariates of our linear model. In order to simplify the model we got using the `stepAIC` function, we then tried to fit a LASSO model using these variables, to see if we could remove some more predictors by shrinking their coefficients to zero. In this way we were able to exclude four more variables from our list of covariates; we then again used the `stepAIC` function starting from a linear model containing the remaining selected variables as predictors and obtained the following final linear model.

```
##
## Call:
## lm(formula = SalePrice ~ LotArea + LandSlope + Condition1 + BldgType +
##     OverallQual + YearBuilt + YearRemodAdd + RoofStyle + BsmtExposure +
##     BsmtFinSF1 + BsmtFinSF2 + TotalBsmtSF + HeatingQC + LowQualFinSF +
##     GrLivArea + BsmtFullBath + BedroomAbvGr + KitchenAbvGr +
##     KitchenQual + Functional + Fireplaces + GarageArea + WoodDeckSF +
##     ScreenPorch + PoolArea, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -394216  -13448    -705   12688  223596
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -1.019e+06  1.192e+05  -8.546  < 2e-16 ***
## LotArea            1.154e+04  2.248e+03   5.134 3.23e-07 ***
## LandSlopeMod       7.377e+03  3.934e+03   1.875 0.060988 .
## LandSlopeother    -9.169e+03  9.136e+03  -1.004 0.315717
## Condition1Norm     7.023e+03  3.503e+03   2.005 0.045169 *
## Condition1other   -8.242e+03  4.343e+03  -1.898 0.057933 .
## BldgTypeTwnhsE    -9.692e+03  3.613e+03  -2.683 0.007389 **
## BldgTypeother     -7.238e+03  3.955e+03  -1.830 0.067441 .
## OverallQual        1.234e+04  1.046e+03  11.798  < 2e-16 ***
## YearBuilt          2.366e+02  4.013e+01   5.896 4.64e-09 ***
## YearRemodAdd       2.367e+02  5.631e+01   4.203 2.80e-05 ***
## RoofStyleHip       4.803e+03  2.089e+03   2.299 0.021666 *
## RoofStyleother    -2.144e+03  5.567e+03  -0.385 0.700220
## BsmtExposureNo    -3.029e+03  2.360e+03  -1.284 0.199517
```

```
## BsmtExposureother  1.168e+04  2.754e+03   4.241 2.37e-05 ***
## BsmtFinSF1         1.434e+03  3.385e+02   4.235 2.43e-05 ***
## BsmtFinSF2        -8.165e+02  4.428e+02  -1.844 0.065421 .
## TotalBsmtSF        2.250e+01  2.577e+00   8.728  < 2e-16 ***
## HeatingQCGd       -4.541e+03  2.390e+03  -1.900 0.057629 .
## HeatingQCTA       -6.873e+03  2.246e+03  -3.059 0.002260 **
## HeatingQCother    -3.223e+03  4.702e+03  -0.686 0.493073
## LowQualFinSF      -5.272e+01  1.674e+01  -3.149 0.001673 **
## GrLivArea          6.302e+01  2.820e+00  22.352  < 2e-16 ***
## BsmtFullBath       4.190e+03  1.967e+03   2.130 0.033373 *
## BedroomAbvGr      -6.839e+03  1.315e+03  -5.202 2.25e-07 ***
## KitchenAbvGr      -1.422e+04  4.915e+03  -2.892 0.003883 **
## KitchenQualGd     -4.603e+04  3.524e+03 -13.063  < 2e-16 ***
## KitchenQualTA     -4.702e+04  4.153e+03 -11.322  < 2e-16 ***
## KitchenQualother  -3.809e+04  6.666e+03  -5.714 1.34e-08 ***
## FunctionalTyp      1.581e+04  5.301e+03   2.982 0.002913 **
## Functionalother   -4.602e+03  6.335e+03  -0.726 0.467753
## Fireplaces         3.439e+03  1.481e+03   2.322 0.020396 *
## GarageArea         2.783e+01  4.969e+00   5.600 2.57e-08 ***
## WoodDeckSF         1.879e+01  6.783e+00   2.769 0.005690 **
## ScreenPorch        2.910e+01  1.448e+01   2.010 0.044656 *
## PoolArea           6.887e+01  2.080e+01   3.311 0.000954 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29470 on 1422 degrees of freedom
## Multiple R-squared:  0.8659, Adjusted R-squared:  0.8626
## F-statistic: 262.2 on 35 and 1422 DF,  p-value: < 2.2e-16
```

The variables we selected as covariates for the final linear model we developed are then the following ones:

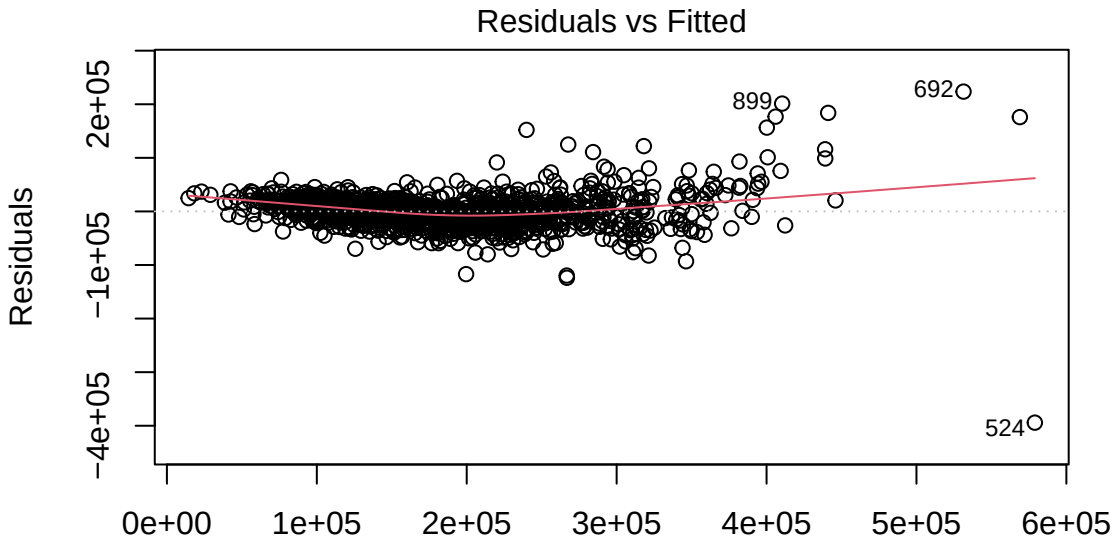
```
##  [1] "LotArea"      "LandSlope"    "Condition1"   "BldgType"     "OverallQual"
##  [6] "YearBuilt"    "YearRemodAdd" "RoofStyle"    "BsmtExposure" "BsmtFinSF1"
## [11] "BsmtFinSF2"   "TotalBsmtSF"  "HeatingQC"    "LowQualFinSF" "GrLivArea"
## [16] "BsmtFullBath" "BedroomAbvGr" "KitchenAbvGr" "KitchenQual"  "Functional"
## [21] "Fireplaces"   "GarageArea"   "WoodDeckSF"   "ScreenPorch"  "PoolArea"
```

We can underline the fact that some of these variables are the ones we noticed to contribute noticeably to the first component in the *dimensionality reduction* paragraph.

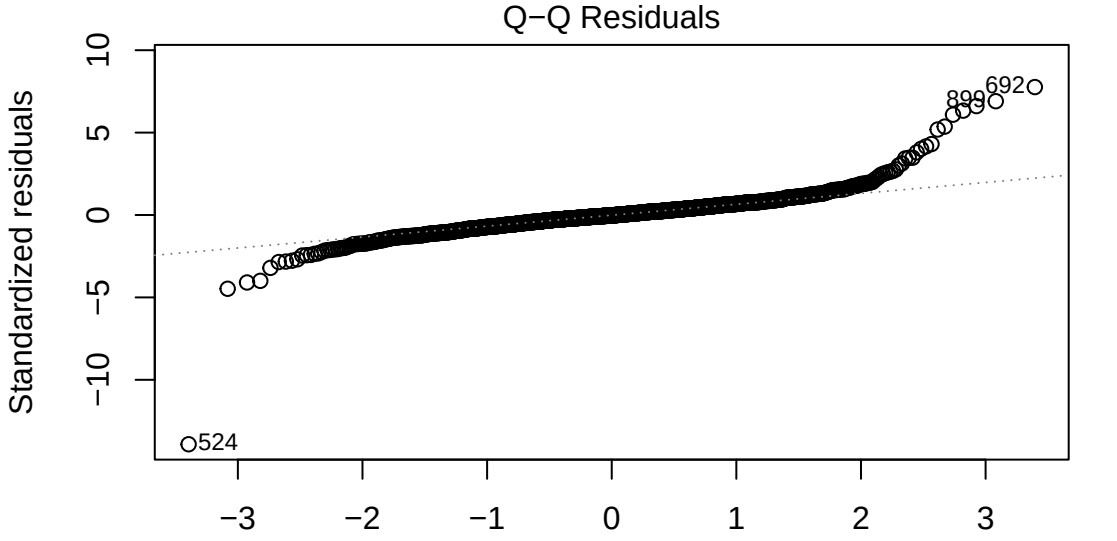
This liner model has 25 predictors and *adjusted* $R^2 = 0.8626$; moreover, by looking at the p-values in the model's summary, we can see that the terms are in general statistically significant. We also computed the *variance inflation factor* (VIF) associated to each predictor to spot eventual collinearity and we obtained $VIF_j < 4 \; \forall j$.

Following this procedure, we therefore built a linear model which showcases a good trade-off between its performance and its complexity:

- its covariates are selected to minimize the AIC;
- its performance is good in terms of *adjusted* R^2 (moreover, its R^2 is not far from the complete model's one, which was approximately equal to 0.88);
- it does not have multicollinearity problems;
- we were able to simplify the model's structure by reducing the number of predictors, while maintaining consistent results in terms of the model's performance.



By plotting the residuals against the fitted values, we can see that the linearity and the homeschedasticity assumptions are overall met; however, we can spot some outliers depending on the dataset.



By looking at the qqplot, the normality assumption could be questioned; as already mentioned, we tried to fit the model using the logarithm of the response variable to fix this problem, but it didn't improve the situation.

We decided to not include interaction terms in our linear model, since they would have decreased its interpretability, increased its complexity (and hence the risk of overfitting and multicollinearity) and its performance was already satisfying.

```
knitr::knit_exit()
```

1. Anna Montoya D. House prices - advanced regression techniques. 2016.