Regression Models Competition

Outliers

```
load("data.RData")
pacman::p_load(ggplot2, knitr)
```

Executive summary

The objective of this assignment is to develop a predictor on the `SalesPrice` variable using regression. To do so, the objective is to build several regression models sing different techniques, and then choose the best among them. For this to be an efficient process, preprocessing and data filtering needs to be performed beforehand, including:

- Correctly reading the dataset.
- Removing irrelevant variables.
- Detecting collinearity and remove variables accordingly.
- Imputing the data.
- Creating new variables with interesting information from the others.

This will be properly explained later, on Section .

Among the produced models, the best we have be

This part needs to be filled the last..

Introduction & Problem Statement

Dataset Context

Two datasets were provided for this assignment, a training dataset (containing the `SalePrice` variable), and a test dataset (without the `SalePrice` variable). The former is used to train the models and to produce an estimation on the prediction power of the model, and the latter is only for testing purposes and to produce the predictions that need to be delivered along with this report. Unless stated otherwise, when the report mentions the `dataset`, it is understood to be only the training part of it.

The dataset contains 79 explanatory variables (excluding the ID and the SalePrice) for approximately 1460 observations in the training set. These variables encompass a wide variety of property characteristics, ranging from physical attributes (e.g., GrLivArea, TotalBsmtSF) and quality assessments (e.g., OverallQual, KitchenQual) to situational factors (e.g., Neighborhood, SaleCondition).

Objective and Evaluation Metric

The objective of this assignment is to build a robust predictive model for the SalePrice variable.

Housing prices naturally follow a right-skewed distribution, where a small number of expensive houses stretch the tail of the distribution, modeling the raw price can lead to violations of the normality assumption required for many statistical tests. Furthermore, in real estate valuation, relative errors are often more critical than absolute errors (e.g., a $10,000 error is significant on a $50,000 house but negligible on a $500,000 house).

To address this, and in accordance with the assignment guidelines, we aim to predict the logarithm of the sale price. Consequently, our primary metric for evaluating goodness of fit and predictive power will be the *Root Mean Squared Error* (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\big(\log(y_i) - \log(\hat{y}_i)\big)^2}$$

Minimizing this value ensures our model performs well across cheap and expensive properties alike.

Other metrics that have also been considered for the analysis are:

- **MAE:**
- **R squared:**
- **Adjusted R-squared:** Following the parsimonious assumption, simpler models are preferred over more complex ones if the increment on variability is not justified.
- **AIC:**

Methodology

To ensure the clarity and reproducibility of our results, our analysis follows a structured statistical workflow using strictly the techniques covered in this course:

1. **Exploratory Data Analysis (EDA):** We begin by visualizing the target variable and analyzing correlations to identify potential predictors and outliers.

2. **Data Preprocessing:** We address missing values (imputation) and encode categorical variables to prepare the data for linear modeling.
3. **Variable Selection:** Given the high number of predictors, we employ selection techniques (e.g., Stepwise selection or Regularization) to reduce dimensionality and prevent overfitting.
4. **Model Fitting & Validation:** We fit several candidate models and validate their performance using Out-of-Sample testing (Cross-Validation) to select the most parsimonious and accurate predictor.

Exploratory Data Analysis

The dataset originally contained 79 variables, with mixed types of data. An initial cleaning was performed to correctly read them, and after observing them and their descriptions, we created a new set of variables, which can be seen in Table 1.

Table 1: Table with newly created vars and their description.

Variable	Type	Description
`TotalBath`	Continuous	The total number of bathrooms in the house, adding, above ground and in the basement. Since toilets are less important than full-bathrooms, they are weighted by $\frac{1}{2}$.
`HouseAge`	Continuous	Age of the house in years (when sold).
`YearsSinceRemod`	Continuous	Time passed since last renewal.
`TotalSF`	Continuous	Surface over ground plus are of basement.
`TotalPorchSF`	Continuous	Added surfaces of porches, terraces, etc.
`Has2ndFloor`	Binary	If the house has a second floor or not.
`IsRemodeled`	Binary	If the house has been remodeled or not.

The main reason for creating these new variables is to reduce collinearity. We observed that some variables where tightly related and could be expressed as a single value. For example, observations with a big surface area above ground are likely to have a big surface area in their basement. Conversely, observations with low surface above ground are likely to have a smaller basement, as it is natural. For this reason, we combined both into a single variable, called `TotalSF`. Besides, it is likely that the "number shown in the advertise" is this very number, which highly impacts the decision.

Similiar reasons follow for the rest of the created variables. To ensure that their creation is relevant for the model, we computed the correlation between them and the `SalePrice` variable. Output can be seen in Figure 1.

```
plot(plot_correlation_newvars)
```

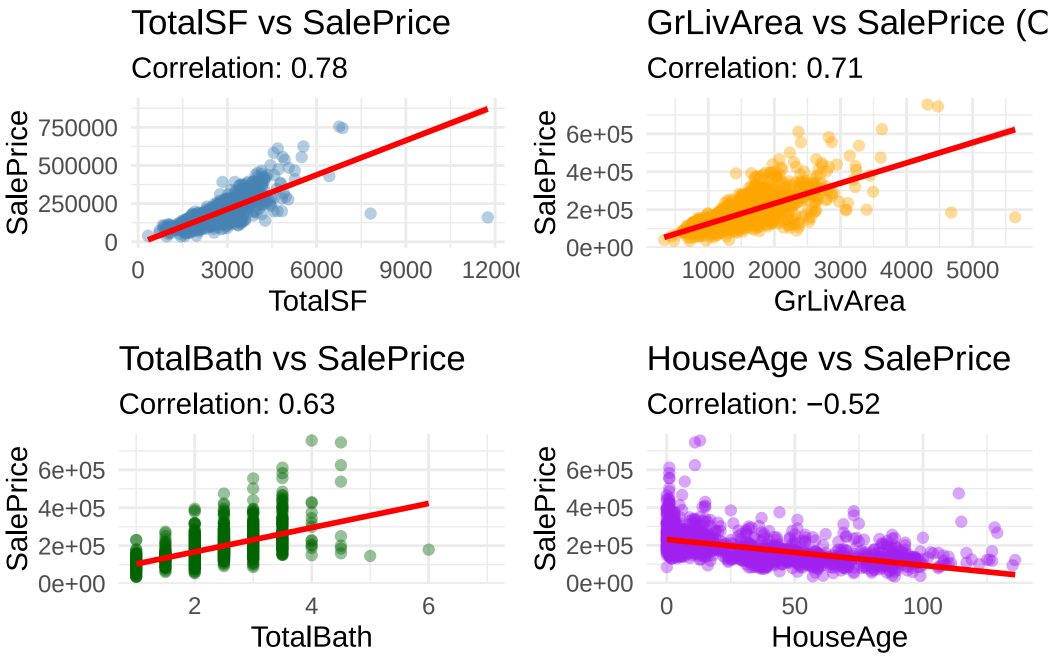

Figure 1: A correlation plot between the `SalePrice` variable and the newly created variables.

The coefficients are non-trivial, which means they can be used as predictors and they will have an impact. In consequence of the creation of these variables, others need to be modified.

Some variables can be safely removed now, as it is the case for every `Bath` variable, since the new `TotalBath` encapsulates the same information and keeping them would be redundant. In other cases, such as the `YrSold`, we need to keep it, since `HouseAge` does not fully contain all the information (consider inflation, which affects prices in general, and which is not encapsulated int the `HouseAge` variable). Their correlation is acceptable too (see ***[PLACEHOLDER FOR REFERENCE TO GLOBAL CORRPLOT, DO NOT LEAVE AS IS]***), thus it is safe to keep them both. However, since

$$\texttt{HouseAge} = \texttt{YrSold} + \texttt{YearBuilt},$$

the variable `YearBuilt` needs to be removed, otherwise there would be redundant information affecting the models.

Next, we removed variables with low variability. To do so we used `caret`'s `nearZeroVar` function, which considers any "ratio of the most common value to the second most common value" below $\frac{95}{5}$ to be of zero variance. Figure 2 shows the variables that didn't pass the filter.

```
plot(graficos_variables_eliminar_sinvariabilidad)
```

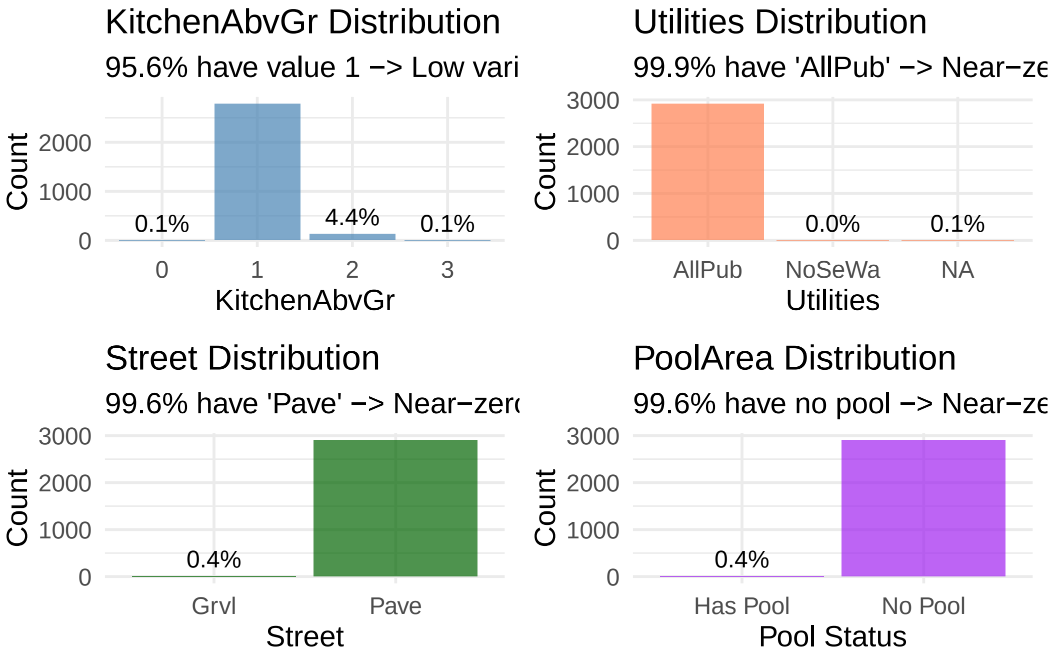

Figure 2: Variables with low-variance, which tend to explain little of the data.

Next, we focused our attention towards the response variable. As mentioned in Section , the response variable needs to be log-transformed, since the distribution is right-skewed, Figure 3 shows the effect this transformation produces on it. The log-response behaves like a normal distribution, as the QQ-Plot demonstrates, which is ideal for our modelling purposes.

```r
plot(grafico_exploratorio_respuesta)
```

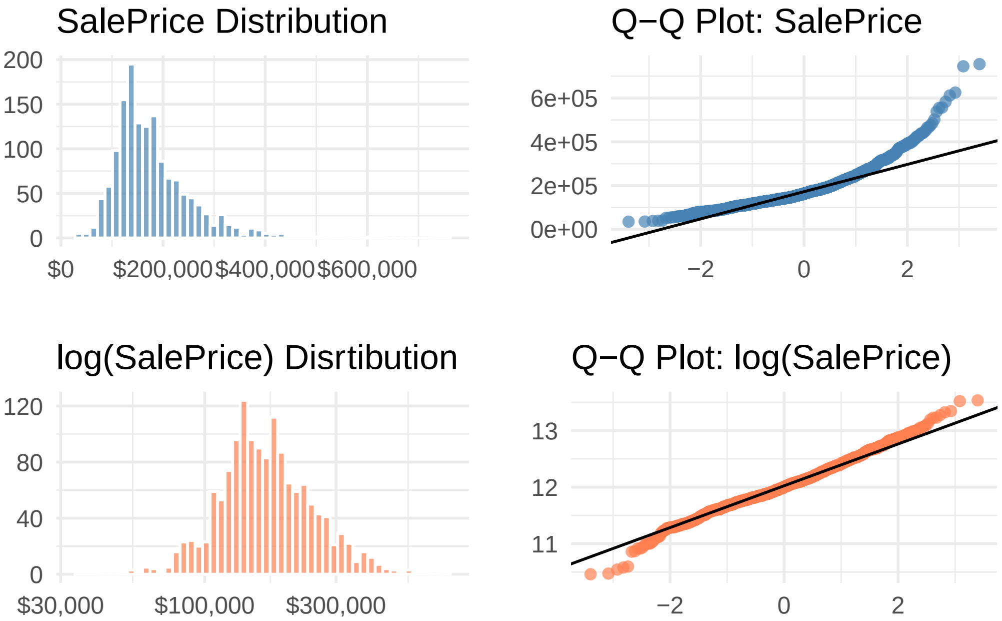

Figure 3: Comparison between the response variable as it it is against its log-scaled version.

Another possible transformation that we could have done, but chose not to, is to log-transform the `TotalSF` variable. Like money, the absolute surface area available is less important when the value is high, but has a big impact when it is low. However, in our observations, the impact of this transformation was not significant. This lead us to keep it as it is, since it is easier to interpret without transformations.

This concludes the data exploration for numerical variables. As a final note on this part, we present Figure 4, where the 10 highest correlated variables are presented. Note we chose to use Spearman's rank correlation coefficient instead of Pearson's. Motivation for this is…

We shift our focus now to categorical variables. `OverallQual`, `Neighborhood`, `ExterQual`, `KitchenQual` seemed like good candidates for predictors, so we studied their correlations with the response variable.

- The overall quality of the house shows a clear positive impact on the price. Naturally, a better house is more expensive.
- The neighborhood does not produce any conclusive result. The *Meadow* neighborhood seems to have consistent low values for prices, whereas in *North Ridge* there is a lot of variablity reflected on the length of the boxplot.

```
# plot(grafico_correlaciones_precio_predictores)
```

Figure 4: Top 10 numerical variables with correlation.

```
# plot(grafico_correlaciones_categoricas)
```

Figure 5: Correlation `SalePrice` and the categorical variables `OverallQual`, `Neighborhood`, `ExterQual` and `KitchenQual`.

- The quality of the exterior has a bigger impact on price than the quality of the kitchen, likely due to the fact that the interior can be renovated “easily”, but the exterior requires more time and effort.

An important question to answer is if the `OveraallQual`, the `KitechenQual`, and the rest of the “quality” variables are correlated among them. Figure 6 shows that, indeed, they are correlated with different magnitudes. The `SalePrice` variable has also been included in the plot.

```
# plot(graficos_heatmap_correlation)
```

Figure 6: Correlation heatmap between the quality-related categorical variables.

Interestingly, the condition of the exterior of the house does not seem to be correlated with the rest of them. In particular, is not correlated at all with the `SalePrice`. However, the exterior quality does seem to be correlated with the kitchen quality. The quality of the garage is also very correlated to its condition. Observing Figure 7, there seems to be a tendency of categorizing the garage quality into the “Typical/Average” category (88% of them). Kitchen quality and exterior quality also tends to be categorized in “Typical/Average” or in “Good” at the same time.

Figure 7 also shows the distribution of the values agglomerated by exterior and kitchen quality.

There is a clear positive tendency in the price along with the quality of the exterior (in the lower levels the price is more concentrated, whereas in the upper ones it is more stretched). This shows that values are left-skewed.

On the other hand, the exterior condition seems to have a minor impact on sale prices, since many of the values seem to be agglomerated in the “Typical/Average” or the “Good”

```
# plot(graficos_pares_altamente_correlacionados)
```

Figure 7: Pairs of highly correlated variables.

categories.

This analysis can be extended to the rest of the categorical variables. Figure 8 shows the relationship between prices and the category levels by the use of the median for each level (less affected by outliers).

Unsurprisingly, a better quality translates into a better saleprice. However, the differences vary from variable to variable. An excellent basement is far superior of an improvement over having a good one than having no basement to having a bad one, or even typical. The exterior quality shows a less exaggerated effect on the excellent level, but a more exaggerated one on the lowest-second to lowest difference.

The neighbors are also decent predictors. However, as we already saw in Figure 5, the variance is high among some of them, so the predictive power of this variable may decrease in those neighbors with high variance.

After this analysis, we decided to remove the `GarageCond`, `ExterCond` and `Exterior2nd` variables, since our analysis reports they are either unimportant or correlated to others.

Missing values

We focus now on missing values on the remaining variables. Figure 9 shows those remaining with any of them.

```
# plot(grafico_median_price_by_level)
```

Figure 8: Median price by level for the 10 most correlated variables.

```
# plot(missing_values)
```

Figure 9: Percentage of missing values across the remaining variables.

LotFrontage is the variable with the highest number of missing values, likely due to the fact that measuring this value was not always possible. Electrical, MasVnrArea and MasVnrType show a small percentage of missing values. In particular MasVnrArea and MasVnrType have the exact same number of missing values, which raises the question: are they always misssing at the same time?

```
# plot(graficos_vis_miss)
# plot(graficos_missing_upset)
```

Figure 10: Distribution of missing values.

Indeed, Figure 10 shows that the two variables are missing for the same indices. In the case of LotFrontage, there might be a hidden pattern difficult to see at first, glance, thus Figure 11.

```
# plot(graficos_missing_neighborhood)
# plot(graficos_missing_lotshape)
```

Since there is still no apparent behavior, the final question before removing it is if it affects the price sales. First, Figure 12 shows that the variable LotArea is not affected by the missing values in LotFrontage.

Figure 11: Missing values grouped by neighborhood and by lot shape. No clear pattern can be established from them.

```
# plot(graficos_missing_lotfrontage)
```

Figure 12: Distribution of the lot area grouped by the lot frontage availability. Distributions are similar, so there is no bias in missingness.

What about prices? Figure 13 shows that there is a quantifyiable difference between the two groups, best represented by their means (8% higher in the case with missing values).

Since the variable cannot be ignored (it has an effect on sale prices), imputation was required.

Data Preprocessing

Variable Selection

Model Fitting and Validation

Results & Model Comparison

Discussion: Drivers of Price

Conclusion

```
# plot(graficos_missing_price)
```

Figure 13: Distribution of prices grouped by values missing or not. Observations with missing values have a higher median, which cannot be ignored.