House Prices Analysis

Amina Anna Mahamane Ousmane

2025-04-28

Introduction

This document demonstrates a regression-focused analysis on a high-dimensional house prices dataset (house-prices.csv) that contains 81 columns covering various property, neighborhood, and structural characteristics. Our objective is to predict the final sale price (SalePrice) using these predictors. The analysis will cover:

(a) Exploratory Data Analysis (EDA)
(b) Ordinary Least Squares (OLS) regression
(c) Regularization
(d) A comparison of the OLS and LASSO models

```
df_raw <- read.csv("house-prices.csv", stringsAsFactors = FALSE)
```

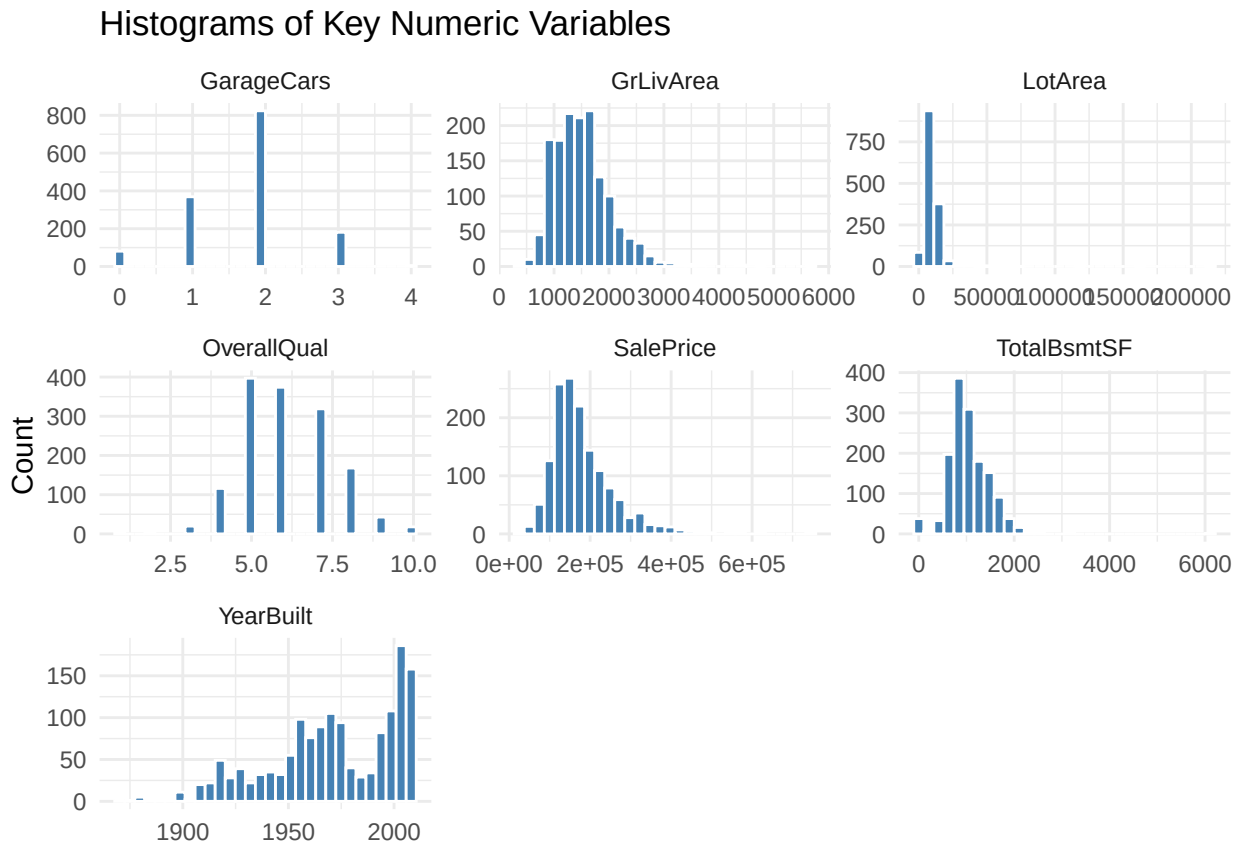
(a) Exploratory Data Analysis (EDA)

The dataset contains 1,459 rows and 81 columns, offering a comprehensive view of various aspects of residential properties. It includes basic identifiers (like Id) and structural details such as MSSubClass, MSZoning, and LotFrontage, along with more in-depth features like OverallQual, OverallCond, YearBuilt, and YearRemodAdd. WE also find detailed information on exterior materials, basement finishes, floor areas, and garage attributes, as well as sale-related variables such as SaleType, SaleCondition, and the target variable SalePrice. While many columns provide continuous numerical data (e.g., LotArea, GrLivArea, TotalBsmtSF), there are also several categorical variables (e.g., Neighborhood, HouseStyle, RoofStyle) that capture qualitative aspects of the properties. Overall, this high-dimensional dataset offers a rich mix of variables that can be used to explore and model the factors influencing house prices. Before modeling, it is important to understand the distribution of each feature, identify missing values, and consider whether any transformations are needed to meet regression assumptions (e.g., normality of residuals). Additionally, it will be helpful to examine correlations among numerical predictors and the target variable (SalePrice) to see which features appear to be most strongly associated with housing prices. For instance, variables such as square footage of the living area, basement area, or overall quality often correlate highly with sale price. By generating correlation matrices, histograms, and scatterplots, we can gain initial insights into data structure, detect potential outliers, and spot any anomalies that might affect the reliability of our regression models.

We will focus our EDA on the target variable SalePrice and on key predictors that domain knowledge indicate are strongly related to house values.

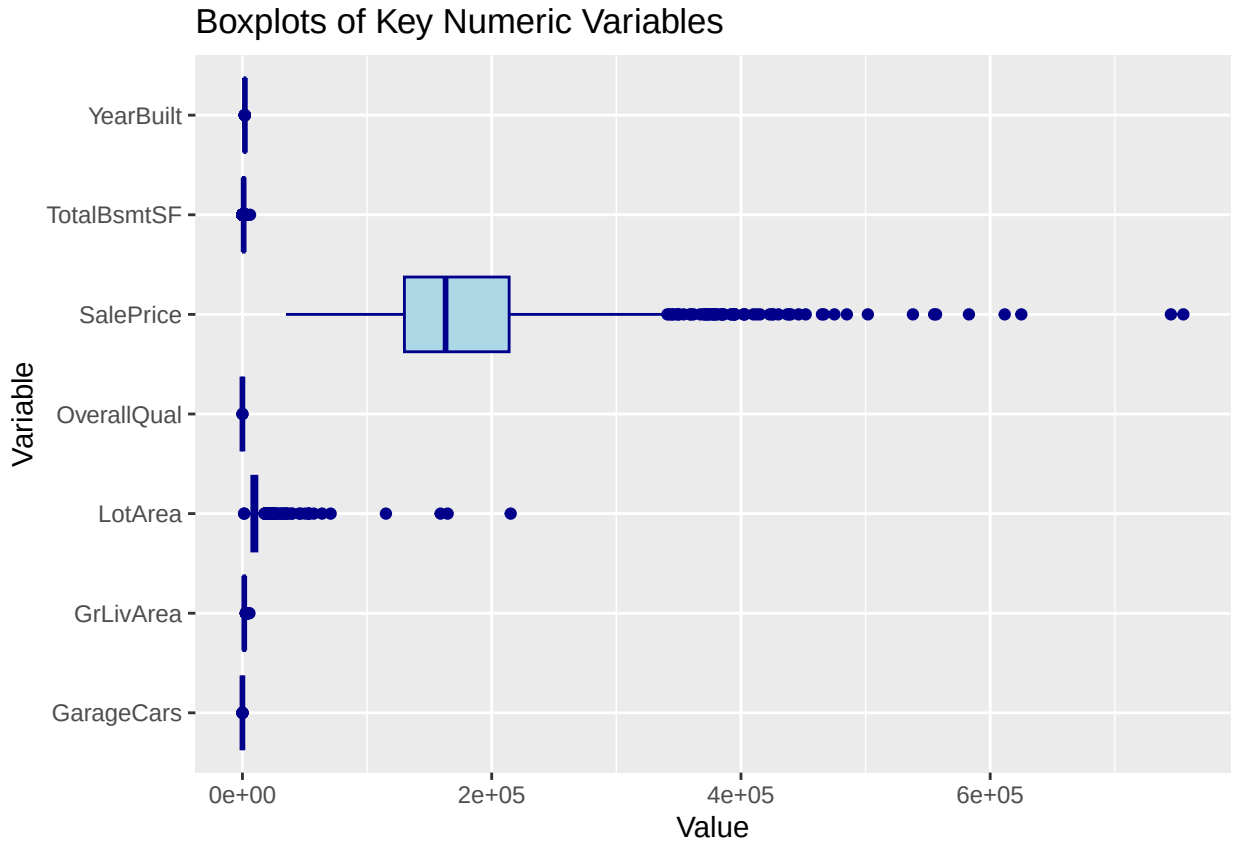
```
# EDA on Key Numeric Variables
key_numeric_vars <- c("SalePrice", "OverallQual", "GrLivArea",
                      "TotalBsmtSF", "YearBuilt", "GarageCars", "LotArea")
df_numeric <- df_raw %>% select(any_of(key_numeric_vars))
df_numeric %>%
  pivot_longer(cols = everything(), names_to = "Variable", values_to = "Value") %>%
  ggplot(aes(x = Value)) +
  facet_wrap(~ Variable, scales = "free", ncol = 3) +
  geom_histogram(bins = 30, fill = "steelblue", color = "white") +
```

```
  labs(title = "Histograms of Key Numeric Variables", x = NULL, y = "Count") +
  theme_minimal()
```



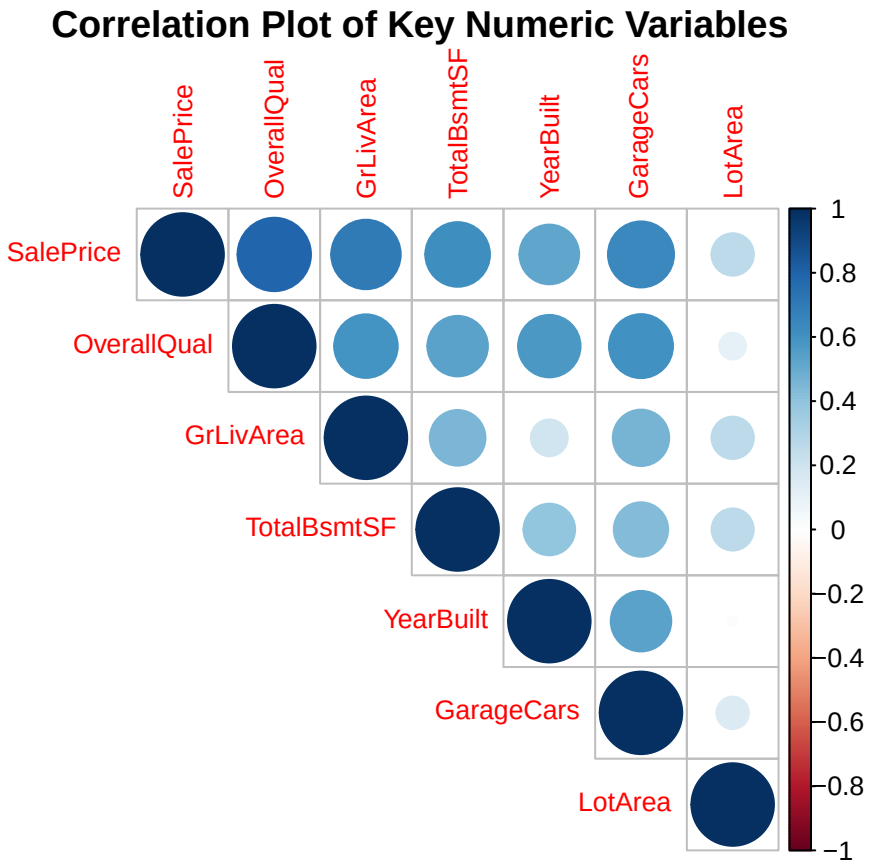
These histograms offer a quick snapshot of the distributions for six key numeric variables: - **GarageCars:** Most properties have space for 1–2 cars, with a fair number accommodating 3 cars, indicating that 2-car garages are quite common.
- **GrLivArea:** This is right-skewed, with the majority of houses clustered under 2,000 square feet but a tail extending toward much larger homes.
- **LotArea:** Strongly right-skewed, as most lots remain under 10,000–12,000 square feet while a few outliers extend to very large plots.
- **OverallQual:** Shows a somewhat bell-shaped distribution, centered around 5–7, reflecting that most homes are of average or slightly above-average quality.
- **TotalBsmtSF:** Also right-skewed; many houses have relatively modest basement areas, but some feature extensive basements exceeding 2,000 square feet.
- **YearBuilt:** Spans well over a century, with noticeable clustering in mid-20th century construction periods and a thinner tail for very old or very new homes.

```
df_numeric %>%
  pivot_longer(cols = everything(), names_to = "Variable", values_to = "Value") %>%
  ggplot(aes(x = Variable, y = Value)) +
  geom_boxplot(fill = "lightblue", color = "darkblue") +
  coord_flip() +
  labs(title = "Boxplots of Key Numeric Variables", x = "Variable", y = "Value")
```



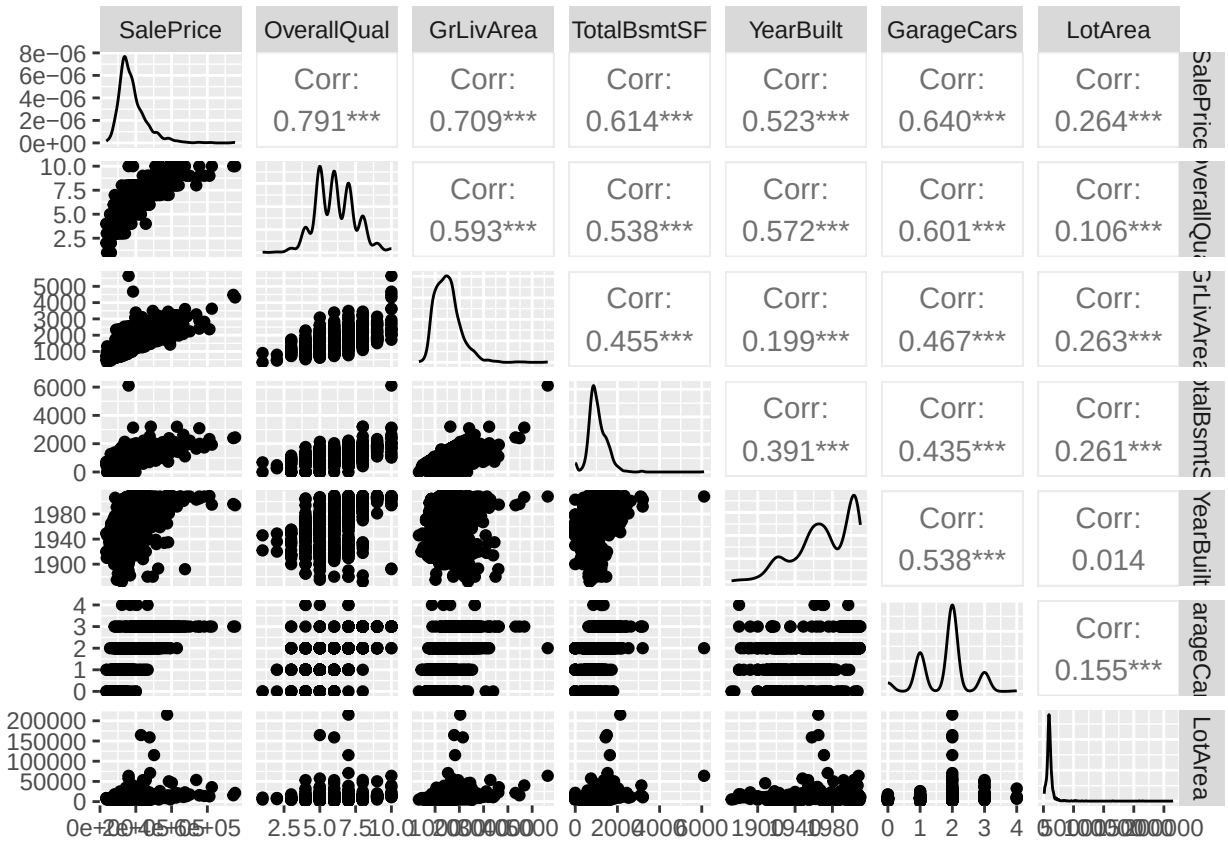
These boxplots provide a concise overview of the distribution and outliers for each numeric variable: - **LotArea** stands out with a large number of outliers, reflecting the broad range of lot sizes—from small plots to very expansive properties.
- **OverallQual** is an ordinal variable (1–10), which naturally restricts its spread. Most values cluster around the middle range (around 5–7), indicating average to slightly above-average quality.
- **GrLivArea** and **TotalBsmtSF** show a right-skewed pattern, with a moderate interquartile range but a tail of larger homes featuring significantly more living or basement area.
- **YearBuilt** spans from the late 1800s through the 2000s, showing a fairly wide range but few extreme outliers—most properties fall within a century-long window.
- **GarageCars** is quite compact, with most homes accommodating one to two cars, and relatively few properties providing space for three or more vehicles.

```
corr_matrix_key <- cor(df_numeric, use = "complete.obs")
corrplot(corr_matrix_key,
         method = "circle",     # alternatives: "color", "number"
         type = "upper",        # display only the upper triangle
         tl.cex = 0.8,          # text label size
         title = "Correlation Plot of Key Numeric Variables",
         mar = c(0, 0, 1, 0))
```



This correlation matrix reveals how strongly each numeric variable is linearly related to the others: - **OverallQual** shows moderate-to-strong positive correlations with measures of house size and capacity (e.g., `GrLivArea`, `TotalBsmtSF`, and `GarageCars`). This suggests that higher-quality homes also tend to have more living space, more basement area, and larger garages.
- **GrLivArea** and **TotalBsmtSF** are fairly strongly correlated, indicating that houses with large above-ground living areas often also have sizeable basements.
- **YearBuilt** is positively correlated with the other variables but to a lesser degree. This implies that newer homes might be slightly bigger or higher quality, though the effect is not as pronounced as size-related variables.
- **GarageCars** tracks moderately with both `GrLivArea` and `OverallQual`, suggesting that homes with more garage space often have greater overall quality and living area.
- **LotArea** exhibits relatively low correlations with most other variables, implying that a larger lot size doesn't necessarily coincide with higher quality, bigger living area, or newer construction. Overall, these correlations highlight which features tend to move together (e.g., bigger homes are often higher quality and have more garage space) and which features vary more independently (e.g., lot size).

```
ggpairs(df_numeric)
```

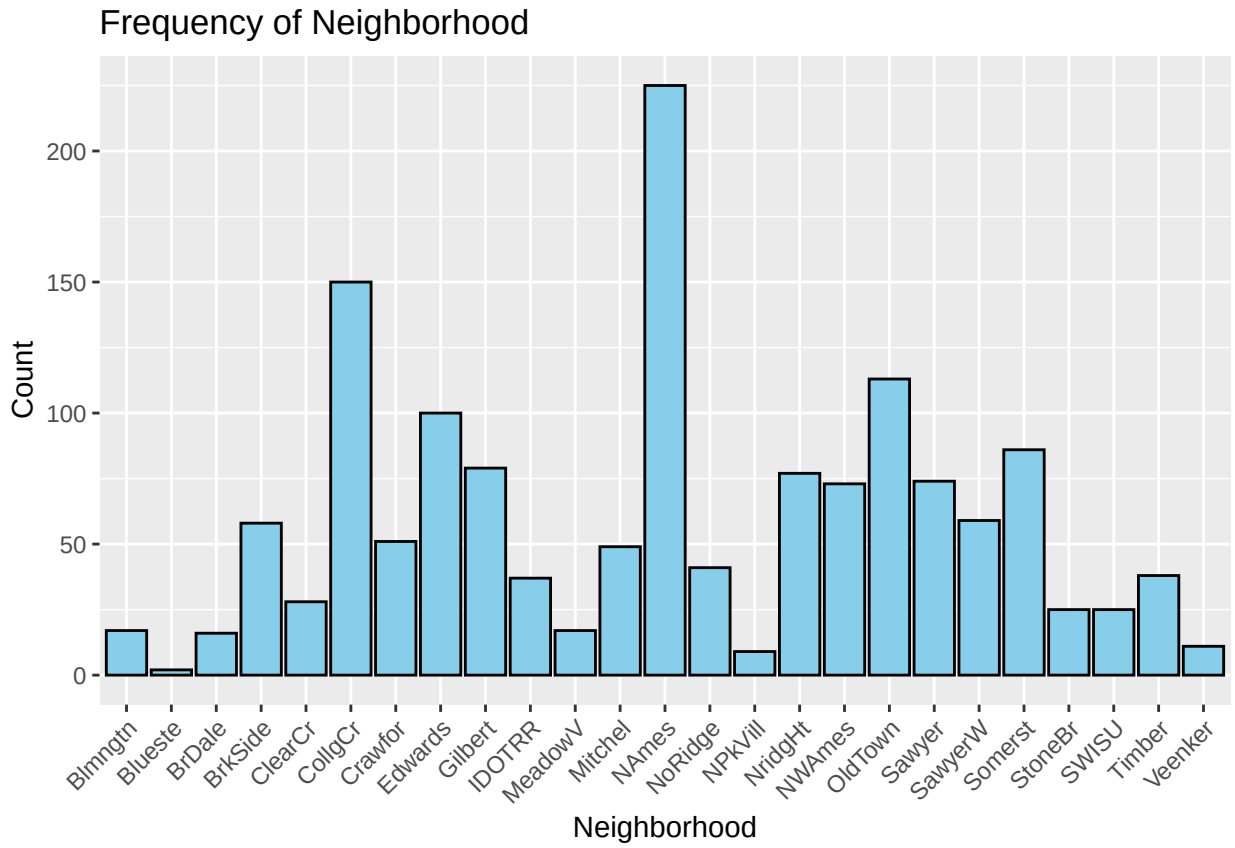


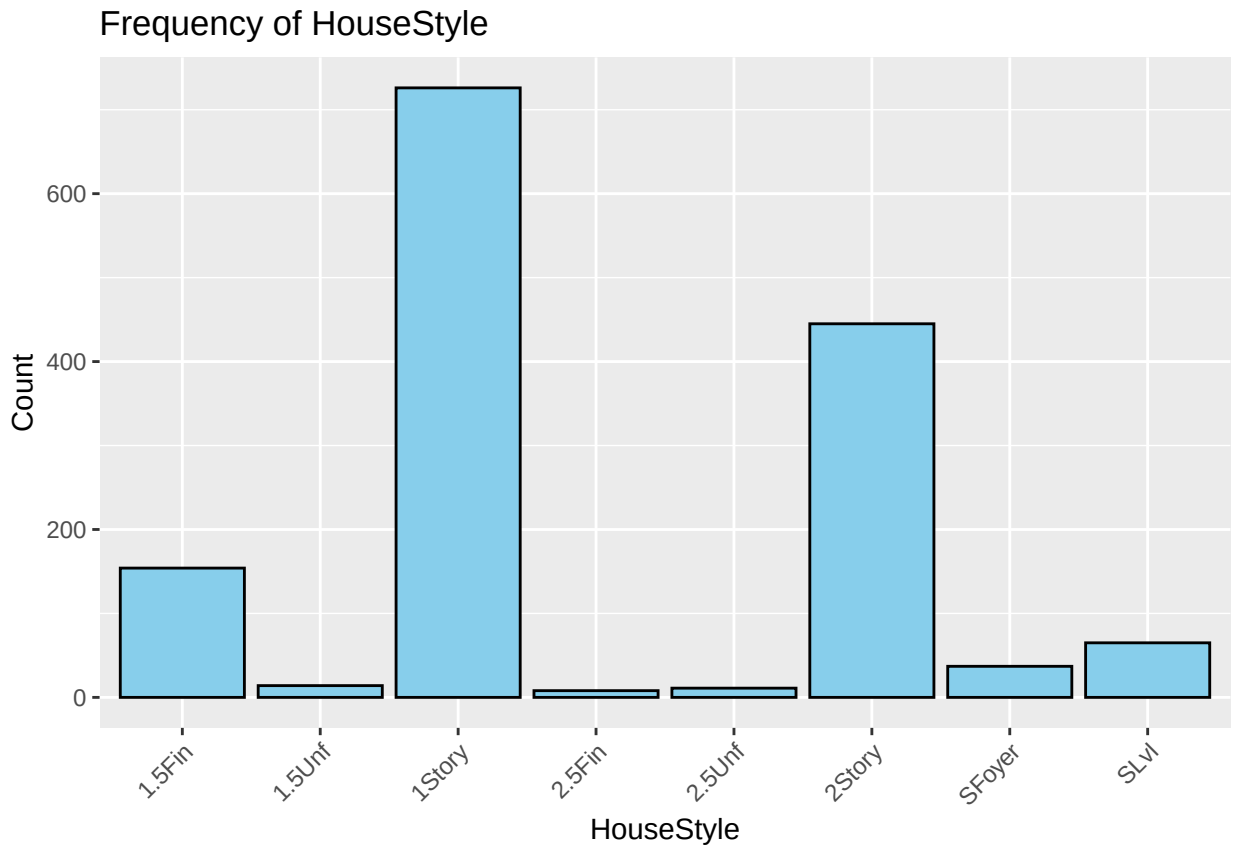
This **scatterplot matrix** (via `GGally::ggpairs`) gives a side-by-side look at both the distributions of each variable (on the diagonal) and their pairwise relationships (in the off-diagonal plots). Here are some key observations: **Diagonal (Univariate Distributions):**
- **OverallQual**: A somewhat bell-shaped distribution with most ratings between 5 and 7.
- **GrLivArea** and **TotalBsmtSF**: Right-skewed distributions, indicating that most houses have moderate living and basement areas, but a few are significantly larger.
- **YearBuilt**: Spans a broad range, though most observations cluster around mid-20th century construction.
- **GarageCars**: Discrete distribution (integer values). Most homes have space for 1–2 cars.
- **LotArea**: Highly right-skewed, with a small number of very large lots.

Off-Diagonal (Pairwise Scatterplots & Correlations):
- **OverallQual** vs. SalePrice: Strong positive correlation, confirming that higher-quality homes tend to sell for more.
- **GrLivArea** vs. SalePrice: Also a notable positive correlation—larger living areas generally command higher prices.
- **TotalBsmtSF** vs. SalePrice: A moderate-to-strong positive relationship, suggesting bigger basements can add value.
- **YearBuilt** vs. SalePrice: Moderate correlation. While newer homes often sell for more, other factors (size, quality) appear to be more influential.
- **GarageCars** vs. SalePrice: Positive correlation, but less pronounced than living area or overall quality.
- **LotArea** vs. SalePrice: Some positive correlation, yet weaker than for interior features, indicating that sheer lot size alone doesn't always dictate higher sale prices.

Overall, the scatterplot matrix confirms that **home size (living area, basement size), quality, and garage capacity** play major roles in determining sale price, while **lot size and year built** appear somewhat less impactful but still relevant.

```r
# EDA on Key Categorical Variables
key_cat_vars <- c("Neighborhood", "HouseStyle")
df_cat <- df_raw %>% select(any_of(key_cat_vars))

# Bar plots for each categorical variable
for (var in names(df_cat)) {
  p <- ggplot(df_cat, aes(x = !!sym(var))) +
    geom_bar(fill = "skyblue", color = "black") +
    labs(title = paste("Frequency of", var), x = var, y = "Count") +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
  print(p)
}
```





The bar plot of Neighborhood shows that the most common neighborhoods in our dataset are NAmes, CollgCr, and OldTown, each contributing a substantial number of observations. Conversely, some neighborhoods (like Blueste and Veenker) have far fewer properties, indicating potential data sparsity in those areas. Meanwhile, the HouseStyle bar plot reveals that 1Story homes are by far the most prevalent, followed by 2Story homes, with relatively few properties categorized as 1.5Fin, 1.5Unf, or 2.5Unf. These distributions highlight which categories dominate our data. For instance, sparse categories might have less predictive power.

(b) Ordinary Least Squares (OLS) regression

```
# Impute missing numeric values with the median
numeric_cols <- names(df_raw)[sapply(df_raw, is.numeric)]
for (col in numeric_cols) {
  df_raw[[col]][is.na(df_raw[[col]])] <- median(df_raw[[col]], na.rm = TRUE)
}

# Impute missing categorical values with "None"
categorical_cols <- names(df_raw)[sapply(df_raw, is.character)]
for (col in categorical_cols) {
  df_raw[[col]][is.na(df_raw[[col]])] <- "None"
}

# Convert character columns to factors
df_raw <- df_raw %>% mutate_if(is.character, as.factor)

# Refresh factor levels
df_clean <- df_raw %>% mutate_if(is.factor, droplevels)
```

```r
# Identify and remove factor columns with fewer than 2 levels
bad_factors <- sapply(df_clean, function(x) is.factor(x) && length(levels(x)) < 2)
if (any(bad_factors)) {
  cat("Removing constant factor variables:\n")
  print(names(bad_factors)[bad_factors])
  df_clean <- df_clean %>% select(-one_of(names(bad_factors)[bad_factors]))
}

# Verify the number of rows and structure of the cleaned data
cat("Number of observations:", nrow(df_clean), "\n")
```

```
Number of observations: 1460
```

```r
# Fit the OLS regression model to predict SalePrice using all predictors
model <- lm(SalePrice ~ ., data = df_clean)

# Display the summary of the model
summary(model)
```

```

Call:
lm(formula = SalePrice ~ ., data = df_clean)

Residuals:
    Min      1Q  Median      3Q     Max
-177220   -9088       0    9521  177220

Coefficients: (8 not defined because of singularities)
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)         -5.813e+05  1.058e+06  -0.550 0.582616
Id                   1.103e+00  1.546e+00   0.714 0.475594
MSSubClass          -5.256e+01  8.265e+01  -0.636 0.524891
MSZoningFV           3.253e+04  1.199e+04   2.713 0.006766 **
MSZoningRH           2.231e+04  1.188e+04   1.877 0.060698 .
MSZoningRL           2.505e+04  1.021e+04   2.453 0.014304 *
MSZoningRM           2.177e+04  9.577e+03   2.274 0.023168 *
LotFrontage          4.328e+01  4.391e+01   0.986 0.324499
LotArea              7.089e-01  1.094e-01   6.478 1.35e-10 ***
StreetPave           3.328e+04  1.218e+04   2.733 0.006371 **
AlleyNone           -1.362e+03  4.210e+03  -0.323 0.746407
AlleyPave           -4.373e+02  6.020e+03  -0.073 0.942103
LotShapeIR2          5.022e+03  4.208e+03   1.193 0.232921
LotShapeIR3          5.069e+03  8.861e+03   0.572 0.567414
LotShapeReg          1.738e+03  1.601e+03   1.086 0.277774
LandContourHLS       7.470e+03  5.116e+03   1.460 0.144484
LandContourLow      -1.131e+04  6.381e+03  -1.773 0.076536 .
LandContourLvl       5.422e+03  3.698e+03   1.466 0.142863
UtilitiesNoSeWa     -3.697e+04  2.634e+04  -1.403 0.160725
LotConfigCulDSac     8.465e+03  3.300e+03   2.565 0.010437 *
LotConfigFR2        -7.478e+03  4.016e+03  -1.862 0.062826 .
LotConfigFR3        -1.718e+04  1.255e+04  -1.369 0.171188
LotConfigInside     -1.306e+03  1.787e+03  -0.731 0.464925
LandSlopeMod         7.304e+03  3.973e+03   1.839 0.066219 .
LandSlopeSev        -4.134e+04  1.141e+04  -3.625 0.000301 ***
NeighborhoodBlueste  7.719e+03  1.921e+04   0.402 0.687926
```

```
NeighborhoodBrDale   -2.321e+03  1.095e+04  -0.212 0.832146
NeighborhoodBrkSide  -5.684e+03  9.466e+03  -0.600 0.548289
NeighborhoodClearCr  -1.458e+04  9.195e+03  -1.585 0.113160
NeighborhoodCollgCr  -1.029e+04  7.243e+03  -1.420 0.155806
NeighborhoodCrawfor   1.179e+04  8.532e+03   1.382 0.167216
NeighborhoodEdwards  -2.146e+04  7.980e+03  -2.689 0.007275 **
NeighborhoodGilbert  -1.155e+04  7.665e+03  -1.507 0.132056
NeighborhoodIDOTRR   -1.190e+04  1.073e+04  -1.109 0.267499
NeighborhoodMeadowV  -6.700e+03  1.118e+04  -0.599 0.549061
NeighborhoodMitchel  -2.112e+04  8.158e+03  -2.589 0.009732 **
NeighborhoodNAmes    -1.736e+04  7.821e+03  -2.220 0.026594 *
NeighborhoodNoRidge   2.511e+04  8.409e+03   2.986 0.002886 **
NeighborhoodNPkVill   1.283e+04  1.403e+04   0.915 0.360371
NeighborhoodNridgHt   1.787e+04  7.498e+03   2.383 0.017335 *
NeighborhoodNWAmes   -1.760e+04  7.996e+03  -2.201 0.027940 *
NeighborhoodOldTown  -1.423e+04  9.641e+03  -1.476 0.140285
NeighborhoodSawyer   -1.117e+04  8.105e+03  -1.378 0.168481
NeighborhoodSawyerW  -2.989e+03  7.767e+03  -0.385 0.700449
NeighborhoodSomerst  -2.670e+03  8.995e+03  -0.297 0.766667
NeighborhoodStoneBr   3.925e+04  8.274e+03   4.744 2.34e-06 ***
NeighborhoodSWISU    -8.456e+03  9.688e+03  -0.873 0.382937
NeighborhoodTimber   -1.011e+04  8.126e+03  -1.244 0.213575
NeighborhoodVeenker  -2.345e+02  1.047e+04  -0.022 0.982145
Condition1Feedr       7.089e+03  5.011e+03   1.415 0.157379
Condition1Norm        1.630e+04  4.182e+03   3.897 0.000103 ***
Condition1PosA        9.215e+03  9.992e+03   0.922 0.356556
Condition1PosN        1.501e+04  7.425e+03   2.022 0.043409 *
Condition1RRAe       -1.532e+04  9.054e+03  -1.692 0.090880 .
Condition1RRAn        1.311e+04  6.943e+03   1.889 0.059153 .
Condition1RRNe       -3.311e+03  1.745e+04  -0.190 0.849557
Condition1RRNn        1.129e+04  1.282e+04   0.881 0.378599
Condition2Feedr      -5.338e+03  2.339e+04  -0.228 0.819480
Condition2Norm       -9.909e+03  2.026e+04  -0.489 0.624875
Condition2PosA        4.323e+04  3.698e+04   1.169 0.242539
Condition2PosN       -2.391e+05  2.759e+04  -8.666  < 2e-16 ***
Condition2RRAe       -1.280e+05  6.500e+04  -1.969 0.049145 *
Condition2RRAn       -2.293e+04  3.145e+04  -0.729 0.466044
Condition2RRNn       -1.828e+03  2.706e+04  -0.068 0.946152
BldgType2fmCon       -3.404e+03  1.248e+04  -0.273 0.785011
BldgTypeDuplex       -7.334e+03  7.416e+03  -0.989 0.322935
BldgTypeTwnhs        -1.868e+04  1.001e+04  -1.867 0.062211 .
BldgTypeTwnhsE       -1.488e+04  9.026e+03  -1.649 0.099388 .
HouseStyle1.5Unf      1.224e+04  7.955e+03   1.539 0.124016
HouseStyle1Story      5.163e+03  4.384e+03   1.178 0.239191
HouseStyle2.5Fin     -1.726e+04  1.237e+04  -1.395 0.163210
HouseStyle2.5Unf     -9.471e+03  9.227e+03  -1.026 0.304896
HouseStyle2Story     -5.952e+03  3.495e+03  -1.703 0.088837 .
HouseStyleSFoyer      9.394e+02  6.256e+03   0.150 0.880653
HouseStyleSLvl        3.411e+03  5.579e+03   0.611 0.541017
OverallQual           6.833e+03  1.014e+03   6.742 2.42e-11 ***
OverallCond           5.792e+03  8.708e+02   6.651 4.39e-11 ***
YearBuilt             3.221e+02  7.700e+01   4.183 3.08e-05 ***
YearRemodAdd          1.065e+02  5.576e+01   1.910 0.056393 .
RoofStyleGable        9.520e+03  1.842e+04   0.517 0.605382
```

```
RoofStyleGambrel      1.292e+04  2.017e+04   0.641 0.521915
RoofStyleHip          9.246e+03  1.849e+04   0.500 0.617166
RoofStyleMansard      2.001e+04  2.138e+04   0.936 0.349384
RoofStyleShed         9.853e+04  3.449e+04   2.857 0.004356 **
RoofMatlCompShg       5.761e+05  5.271e+04  10.929  < 2e-16 ***
RoofMatlMembran       6.712e+05  6.256e+04  10.729  < 2e-16 ***
RoofMatlMetal         6.399e+05  6.220e+04  10.288  < 2e-16 ***
RoofMatlRoll          5.629e+05  5.830e+04   9.655  < 2e-16 ***
RoofMatlTar&Grv       5.769e+05  5.650e+04  10.211  < 2e-16 ***
RoofMatlWdShake       5.684e+05  5.507e+04  10.322  < 2e-16 ***
RoofMatlWdShngl       6.309e+05  5.367e+04  11.756  < 2e-16 ***
Exterior1stAsphShn   -2.482e+04  3.296e+04  -0.753 0.451576
Exterior1stBrkComm   -3.837e+03  2.775e+04  -0.138 0.890029
Exterior1stBrkFace    7.998e+03  1.276e+04   0.627 0.530855
Exterior1stCBlock    -1.540e+04  2.725e+04  -0.565 0.572132
Exterior1stCemntBd   -1.139e+04  1.902e+04  -0.599 0.549335
Exterior1stHdBoard   -1.293e+04  1.293e+04  -1.000 0.317639
Exterior1stImStucc   -2.274e+04  2.814e+04  -0.808 0.419196
Exterior1stMetalSd   -6.026e+03  1.459e+04  -0.413 0.679717
Exterior1stPlywood   -1.359e+04  1.277e+04  -1.064 0.287375
Exterior1stStone     -1.268e+03  2.427e+04  -0.052 0.958352
Exterior1stStucco    -6.972e+03  1.407e+04  -0.495 0.620432
Exterior1stVinylSd   -1.391e+04  1.333e+04  -1.044 0.296894
Exterior1stWd Sdng   -1.375e+04  1.238e+04  -1.111 0.266645
Exterior1stWdShing   -9.327e+03  1.336e+04  -0.698 0.485212
Exterior2ndAsphShn    1.165e+04  2.219e+04   0.525 0.599624
Exterior2ndBrk Cmn    5.768e+03  2.005e+04   0.288 0.773694
Exterior2ndBrkFace    3.706e+03  1.321e+04   0.280 0.779186
Exterior2ndCBlock            NA         NA      NA       NA
Exterior2ndCmentBd    1.182e+04  1.870e+04   0.632 0.527600
Exterior2ndHdBoard    8.019e+03  1.242e+04   0.646 0.518482
Exterior2ndImStucc    1.693e+04  1.434e+04   1.181 0.237885
Exterior2ndMetalSd    5.795e+03  1.420e+04   0.408 0.683248
Exterior2ndOther     -1.817e+04  2.706e+04  -0.671 0.502081
Exterior2ndPlywood    6.239e+03  1.206e+04   0.517 0.604920
Exterior2ndStone     -1.172e+04  1.713e+04  -0.684 0.494062
Exterior2ndStucco     5.342e+03  1.361e+04   0.393 0.694729
Exterior2ndVinylSd    1.280e+04  1.281e+04   0.999 0.318024
Exterior2ndWd Sdng    1.167e+04  1.194e+04   0.977 0.328645
Exterior2ndWd Shng    5.284e+03  1.246e+04   0.424 0.671506
MasVnrTypeBrkFace     4.142e+03  6.828e+03   0.607 0.544285
MasVnrTypeNone        7.103e+03  6.901e+03   1.029 0.303511
MasVnrTypeStone       9.221e+03  7.232e+03   1.275 0.202521
MasVnrArea            2.086e+01  5.782e+00   3.608 0.000321 ***
ExterQualFa          -7.246e+03  1.108e+04  -0.654 0.513112
ExterQualGd          -2.067e+04  4.775e+03  -4.329 1.62e-05 ***
ExterQualTA          -1.985e+04  5.295e+03  -3.748 0.000187 ***
ExterCondFa          -2.553e+03  1.805e+04  -0.141 0.887577
ExterCondGd          -7.155e+03  1.721e+04  -0.416 0.677760
ExterCondPo           8.987e+03  3.165e+04   0.284 0.776482
ExterCondTA          -4.169e+03  1.718e+04  -0.243 0.808332
FoundationCBlock      2.851e+03  3.169e+03   0.900 0.368431
FoundationPConc       4.019e+03  3.415e+03   1.177 0.239440
FoundationSlab       -7.386e+03  1.004e+04  -0.736 0.461926
```

```
FoundationStone       9.640e+03  1.139e+04   0.846 0.397561
FoundationWood       -2.754e+04  1.476e+04  -1.866 0.062337 .
BsmtQualFa           -1.143e+04  6.349e+03  -1.800 0.072078 .
BsmtQualGd           -1.799e+04  3.333e+03  -5.398 8.10e-08 ***
BsmtQualNone          3.811e+04  3.658e+04   1.042 0.297746
BsmtQualTA           -1.427e+04  4.154e+03  -3.434 0.000615 ***
BsmtCondGd           -8.530e+01  5.270e+03  -0.016 0.987089
BsmtCondNone                NA         NA      NA       NA
BsmtCondPo            6.735e+04  2.981e+04   2.260 0.024019 *
BsmtCondTA            2.610e+03  4.241e+03   0.616 0.538319
BsmtExposureGd        1.420e+04  2.994e+03   4.742 2.37e-06 ***
BsmtExposureMn       -3.605e+03  3.015e+03  -1.196 0.232053
BsmtExposureNo       -5.170e+03  2.177e+03  -2.375 0.017724 *
BsmtExposureNone     -1.106e+04  2.297e+04  -0.482 0.630076
BsmtFinType1BLQ       2.787e+03  2.801e+03   0.995 0.319851
BsmtFinType1GLQ       5.619e+03  2.518e+03   2.232 0.025804 *
BsmtFinType1LwQ      -3.236e+03  3.740e+03  -0.865 0.387019
BsmtFinType1None            NA         NA      NA       NA
BsmtFinType1Rec       8.788e+01  2.997e+03   0.029 0.976615
BsmtFinType1Unf       2.668e+03  2.913e+03   0.916 0.359828
BsmtFinSF1            3.869e+01  5.325e+00   7.266 6.61e-13 ***
BsmtFinType2BLQ      -1.308e+04  7.558e+03  -1.730 0.083847 .
BsmtFinType2GLQ      -2.507e+03  9.338e+03  -0.268 0.788372
BsmtFinType2LwQ      -1.415e+04  7.389e+03  -1.915 0.055720 .
BsmtFinType2None     -2.887e+04  2.494e+04  -1.158 0.247262
BsmtFinType2Rec      -1.024e+04  7.111e+03  -1.440 0.150219
BsmtFinType2Unf      -8.320e+03  7.570e+03  -1.099 0.271970
BsmtFinSF2            3.165e+01  9.047e+00   3.498 0.000486 ***
BsmtUnfSF             2.116e+01  4.884e+00   4.333 1.60e-05 ***
TotalBsmtSF                 NA         NA      NA       NA
HeatingGasA           1.003e+04  2.554e+04   0.393 0.694595
HeatingGasW           7.725e+03  2.634e+04   0.293 0.769334
HeatingGrav           1.714e+03  2.801e+04   0.061 0.951226
HeatingOthW          -1.095e+04  3.145e+04  -0.348 0.727691
HeatingWall           2.326e+04  2.970e+04   0.783 0.433560
HeatingQCFa           6.694e+02  4.710e+03   0.142 0.887009
HeatingQCGd          -3.918e+03  2.071e+03  -1.892 0.058712 .
HeatingQCPo           2.103e+03  2.652e+04   0.079 0.936805
HeatingQCTA          -3.241e+03  2.068e+03  -1.567 0.117290
CentralAirY          -2.213e+02  3.863e+03  -0.057 0.954338
ElectricalFuseF       4.444e+01  5.744e+03   0.008 0.993827
ElectricalFuseP      -7.715e+03  1.859e+04  -0.415 0.678244
ElectricalMix        -4.237e+04  4.443e+04  -0.954 0.340451
ElectricalNone        9.747e+03  2.406e+04   0.405 0.685480
ElectricalSBrkr      -2.171e+03  2.946e+03  -0.737 0.461258
X1stFlrSF             4.417e+01  5.637e+00   7.835 1.02e-14 ***
X2ndFlrSF             6.223e+01  5.693e+00  10.930  < 2e-16 ***
LowQualFinSF         -3.503e+00  1.903e+01  -0.184 0.854005
GrLivArea                   NA         NA      NA       NA
BsmtFullBath          1.578e+03  1.978e+03   0.798 0.425001
BsmtHalfBath         -3.769e+02  3.024e+03  -0.125 0.900832
FullBath              3.680e+03  2.197e+03   1.675 0.094240 .
HalfBath              1.806e+03  2.094e+03   0.862 0.388713
BedroomAbvGr         -3.691e+03  1.363e+03  -2.708 0.006873 **
```

```
KitchenAbvGr         -1.362e+04  5.681e+03  -2.397 0.016661 *
KitchenQualFa        -1.985e+04  6.195e+03  -3.204 0.001390 **
KitchenQualGd        -2.350e+04  3.477e+03  -6.758 2.17e-11 ***
KitchenQualTA        -2.240e+04  3.925e+03  -5.707 1.44e-08 ***
TotRmsAbvGrd          1.775e+03  9.549e+02   1.859 0.063212 .
FunctionalMaj2       -1.353e+03  1.436e+04  -0.094 0.924934
FunctionalMin1        7.341e+03  8.590e+03   0.855 0.392924
FunctionalMin2        8.549e+03  8.616e+03   0.992 0.321256
FunctionalMod        -5.265e+03  1.054e+04  -0.500 0.617516
FunctionalSev        -3.987e+04  2.953e+04  -1.350 0.177218
FunctionalTyp         1.823e+04  7.447e+03   2.447 0.014530 *
Fireplaces            6.266e+03  2.553e+03   2.454 0.014250 *
FireplaceQuFa        -6.570e+02  6.879e+03  -0.096 0.923925
FireplaceQuGd         2.813e+03  5.314e+03   0.529 0.596655
FireplaceQuNone       8.849e+03  6.221e+03   1.422 0.155162
FireplaceQuPo         1.240e+04  7.905e+03   1.569 0.116991
FireplaceQuTA         3.758e+03  5.525e+03   0.680 0.496508
GarageTypeAttchd      1.930e+04  1.101e+04   1.752 0.079978 .
GarageTypeBasment     2.363e+04  1.277e+04   1.850 0.064509 .
GarageTypeBuiltIn     1.914e+04  1.148e+04   1.667 0.095776 .
GarageTypeCarPort     2.425e+04  1.468e+04   1.652 0.098747 .
GarageTypeDetchd      2.237e+04  1.102e+04   2.030 0.042546 *
GarageTypeNone        2.299e+04  2.079e+04   1.106 0.269074
GarageYrBlt          -1.992e+01  6.122e+01  -0.325 0.744975
GarageFinishNone             NA         NA      NA       NA
GarageFinishRFn      -2.374e+03  1.959e+03  -1.212 0.225821
GarageFinishUnf      -5.278e+02  2.426e+03  -0.218 0.827825
GarageCars            3.869e+03  2.276e+03   1.700 0.089424 .
GarageArea            1.829e+01  7.891e+00   2.319 0.020589 *
GarageQualFa         -1.253e+05  3.012e+04  -4.160 3.42e-05 ***
GarageQualGd         -1.204e+05  3.093e+04  -3.892 0.000105 ***
GarageQualNone               NA         NA      NA       NA
GarageQualPo         -1.426e+05  3.839e+04  -3.713 0.000214 ***
GarageQualTA         -1.192e+05  2.983e+04  -3.998 6.79e-05 ***
GarageCondFa          1.122e+05  3.474e+04   3.229 0.001275 **
GarageCondGd          1.113e+05  3.609e+04   3.083 0.002094 **
GarageCondNone               NA         NA      NA       NA
GarageCondPo          1.179e+05  3.728e+04   3.161 0.001610 **
GarageCondTA          1.139e+05  3.444e+04   3.306 0.000974 ***
PavedDriveP          -3.555e+03  5.546e+03  -0.641 0.521663
PavedDriveY          -2.687e+02  3.457e+03  -0.078 0.938053
WoodDeckSF            1.533e+01  5.870e+00   2.612 0.009107 **
OpenPorchSF           6.256e-01  1.156e+01   0.054 0.956852
EnclosedPorch         2.810e+00  1.246e+01   0.225 0.821649
X3SsnPorch            3.392e+01  2.236e+01   1.517 0.129534
ScreenPorch           3.595e+01  1.248e+01   2.880 0.004051 **
PoolArea              6.831e+02  2.266e+02   3.015 0.002624 **
PoolQCFa             -1.562e+05  4.087e+04  -3.822 0.000139 ***
PoolQCGd             -1.269e+05  3.682e+04  -3.445 0.000590 ***
PoolQCNone            2.553e+05  1.226e+05   2.083 0.037459 *
FenceGdWo             7.938e+03  4.901e+03   1.620 0.105564
FenceMnPrv            9.459e+03  4.001e+03   2.364 0.018225 *
FenceMnWw             3.108e+03  8.205e+03   0.379 0.704894
FenceNone             8.895e+03  3.667e+03   2.425 0.015438 *
```

```
MiscFeatureNone      -8.276e+02  9.716e+04  -0.009 0.993205
MiscFeatureOthr       1.366e+04  9.071e+04   0.151 0.880297
MiscFeatureShed       1.775e+03  9.308e+04   0.019 0.984791
MiscFeatureTenC       3.043e+04  9.659e+04   0.315 0.752790
MiscVal              -4.982e-02  6.111e+00  -0.008 0.993496
MoSold               -4.669e+02  2.448e+02  -1.907 0.056704 .
YrSold               -5.699e+02  5.145e+02  -1.108 0.268294
SaleTypeCon           2.573e+04  1.752e+04   1.469 0.142151
SaleTypeConLD         1.608e+04  9.677e+03   1.662 0.096736 .
SaleTypeConLI         4.537e+03  1.154e+04   0.393 0.694211
SaleTypeConLw         1.252e+03  1.213e+04   0.103 0.917843
SaleTypeCWD           1.486e+04  1.285e+04   1.157 0.247644
SaleTypeNew           2.083e+04  1.540e+04   1.353 0.176452
SaleTypeOth           6.979e+03  1.448e+04   0.482 0.629918
SaleTypeWD           -4.283e+02  4.173e+03  -0.103 0.918262
SaleConditionAdjLand  9.734e+03  1.459e+04   0.667 0.504674
SaleConditionAlloca   1.172e+03  8.859e+03   0.132 0.894757
SaleConditionFamily   7.531e+02  6.082e+03   0.124 0.901477
SaleConditionNormal   6.663e+03  2.901e+03   2.297 0.021818 *
SaleConditionPartial  9.986e+01  1.482e+04   0.007 0.994626
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 22570 on 1206 degrees of freedom
Multiple R-squared:  0.9333,    Adjusted R-squared:  0.9193
F-statistic: 66.68 on 253 and 1206 DF,  p-value: < 2.2e-16
```

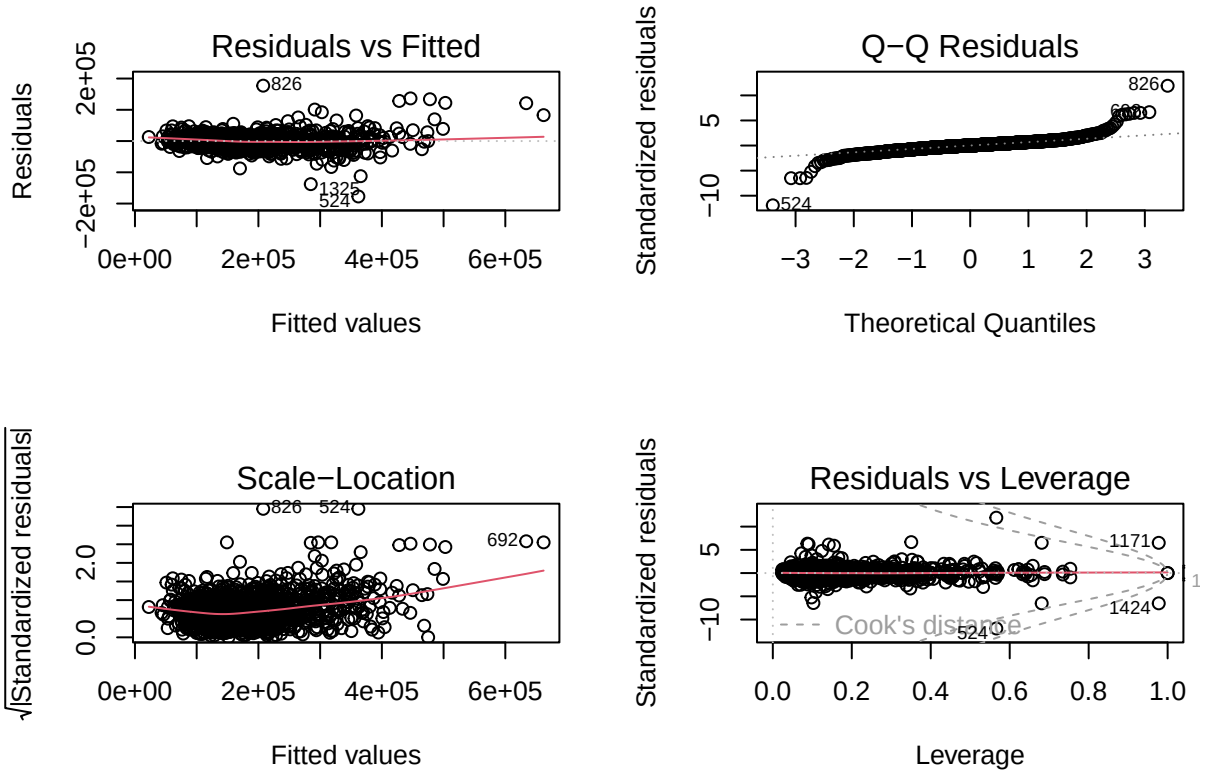
From the summary, we see the model attempts to estimate 254 parameters in total:
- **1 intercept**
- **253 predictors** (as indicated by "on 253 and 1206 DF" in the F-statistic).

However, it also reports "(8 not defined because of singularities)." That means 8 parameters could not be estimated (likely due to perfect collinearity). Thus, there are 254 coefficient "slots" in the design, but only 246 can actually be fit in the final model.

- The model explains approximately 93% of the variance in SalePrice ($R^2 \approx 0.9333$) and has an adjusted $R^2 \approx 0.9193$.
- The overall F-statistic (66.68, $p < 2.2 \times 10^{-16}$) confirms that at least some predictors significantly affect SalePrice.
- The residual standard error is roughly \$22,570, indicating the typical deviation of predictions from actual sale prices.

Several kitchen-quality indicators (e.g., KitchenQualGd, KitchenQualTA) and the number of bedrooms (BedroomAbvGr) show significant negative coefficients. This can happen when other correlated predictors (such as square footage) capture the "positive" effect, causing some intuitive variables to appear negatively associated.

```r
# Plot diagnostic plots for the model
par(mfrow = c(2, 2))
plot(model)
```

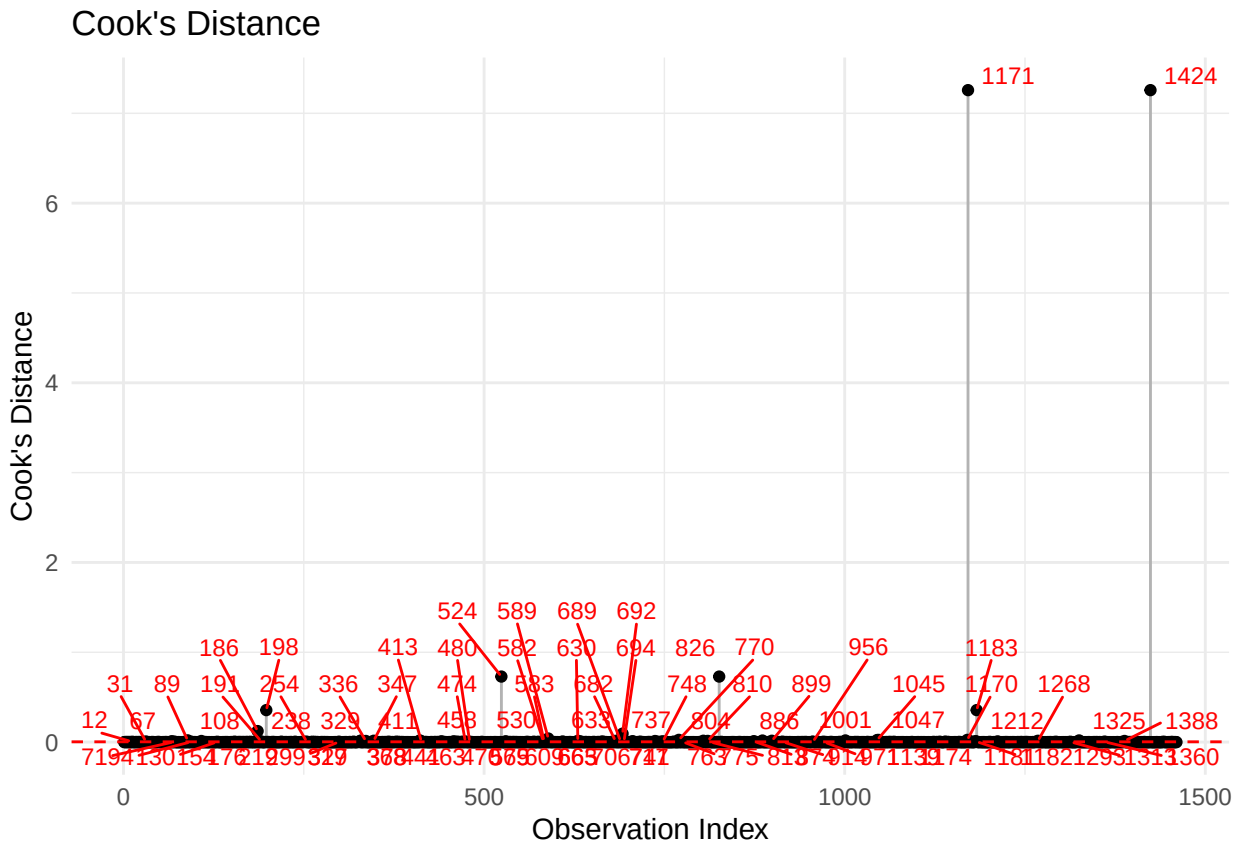


- Residuals vs. Fitted: Suggests a reasonably good fit, though minor patterns may indicate slight non-linearity or heteroskedasticity.
- Q–Q Plot: Residuals largely follow a straight line, but slight deviations at the tails suggest mild departures from normality.
- Scale-Location Plot: Shows a possible increase in residual spread at higher fitted values, hinting at heteroskedasticity.
- Residuals vs. Leverage: Identifies observations with leverage = 1 (very influential points). These are homes with unique or extreme feature combinations. Investigating or removing them could change the model fit.

Diagnostic plots revealed several observations with high leverage (leverage values approaching 1), indicating that these influential points may disproportionately affect the model estimates. To address this, I examined these data points in detail to determine if they represent data entry errors or genuine outliers. Additionally, the Scale-Location and Q–Q plots suggested some heteroskedasticity and mild departures from normality. As a result, I plan to explore transformations, such as applying a logarithmic transformation to the response variable (SalePrice), to stabilize variance and improve the distribution of residuals. These steps will help refine the model and ensure more robust and interpretable results.

```
n <- nrow(df_clean)    # number of observations
p <- length(coef(model)) - 1    # number of predictors (excluding intercept)
# Calculate Cook's distance
cooks_d <- cooks.distance(model)
# Set a common threshold for Cook's distance: 4/(n-p-1)
cooks_threshold <- 4 / (n - p - 1)

# Create a data frame for plotting Cook's distance
df_cooks <- data.frame(
```

```
  index = seq_along(cooks_d),
  cooks_dist = cooks_d
)

# Plot Cook's distance using ggplot2
ggplot(df_cooks, aes(x = index, y = cooks_dist)) +
  geom_segment(aes(xend = index, yend = 0), color = "gray70") +
  geom_point(color = "black") +
  geom_hline(yintercept = cooks_threshold, color = "red", linetype = "dashed") +
  # Label only points above the threshold
  geom_text_repel(
    data = subset(df_cooks, cooks_dist > cooks_threshold),
    aes(label = index),
    color = "red",
    size = 3,
    max.overlaps = 50
  ) +
  theme_minimal() +
  labs(
    title = "Cook's Distance",
    x = "Observation Index",
    y = "Cook's Distance"
  )
```



Cook's Distance measures how much a single observation affects the overall regression model. Most points have relatively low influence, but a few observations—such as 1171 and 1424—exhibit exceptionally high Cook's Distance values, indicating that they significantly impact the regression coefficients. These influential

points warrant further investigation to determine whether they represent true outliers, data entry errors, or cases where model adjustments (such as robust regression or transformations) may be necessary to mitigate their effect.

```r
# Fit a model with a log transformation of SalePrice
model_log <- lm(log(SalePrice) ~ ., data = df_clean)
summary(model_log)
```

```
Call:
lm(formula = log(SalePrice) ~ ., data = df_clean)

Residuals:
     Min       1Q   Median       3Q      Max
-0.69362 -0.04612  0.00353  0.04976  0.69362

Coefficients: (8 not defined because of singularities)
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)           6.889e+00  4.869e+00   1.415 0.157382
Id                   -5.752e-06  7.119e-06  -0.808 0.419268
MSSubClass           -3.971e-04  3.805e-04  -1.043 0.296954
MSZoningFV            4.467e-01  5.521e-02   8.091 1.43e-15 ***
MSZoningRH            4.298e-01  5.471e-02   7.856 8.73e-15 ***
MSZoningRL            4.232e-01  4.703e-02   9.000  < 2e-16 ***
MSZoningRM            3.849e-01  4.409e-02   8.728  < 2e-16 ***
LotFrontage           3.705e-04  2.022e-04   1.833 0.067095 .
LotArea               2.853e-06  5.038e-07   5.663 1.86e-08 ***
StreetPave            9.798e-02  5.607e-02   1.747 0.080821 .
AlleyNone            -1.038e-02  1.938e-02  -0.536 0.592376
AlleyPave             1.417e-02  2.772e-02   0.511 0.609317
LotShapeIR2           2.558e-02  1.938e-02   1.320 0.187050
LotShapeIR3           1.604e-02  4.080e-02   0.393 0.694271
LotShapeReg           7.171e-03  7.372e-03   0.973 0.330849
LandContourHLS        2.823e-02  2.355e-02   1.199 0.230900
LandContourLow       -1.749e-02  2.938e-02  -0.595 0.551830
LandContourLvl        2.765e-02  1.703e-02   1.624 0.104638
UtilitiesNoSeWa      -2.329e-01  1.213e-01  -1.921 0.054998 .
LotConfigCulDSac      2.741e-02  1.520e-02   1.804 0.071496 .
LotConfigFR2         -3.599e-02  1.849e-02  -1.947 0.051821 .
LotConfigFR3         -9.243e-02  5.777e-02  -1.600 0.109884
LotConfigInside      -1.482e-02  8.229e-03  -1.801 0.071982 .
LandSlopeMod          3.074e-02  1.829e-02   1.680 0.093125 .
LandSlopeSev         -1.955e-01  5.252e-02  -3.723 0.000206 ***
NeighborhoodBlueste  -4.066e-02  8.846e-02  -0.460 0.645869
NeighborhoodBrDale   -7.234e-02  5.041e-02  -1.435 0.151500
NeighborhoodBrkSide  -3.712e-03  4.358e-02  -0.085 0.932150
NeighborhoodClearCr   1.713e-02  4.234e-02   0.405 0.685850
NeighborhoodCollgCr  -2.567e-02  3.335e-02  -0.770 0.441613
NeighborhoodCrawfor   1.078e-01  3.929e-02   2.744 0.006158 **
NeighborhoodEdwards  -9.432e-02  3.675e-02  -2.567 0.010381 *
NeighborhoodGilbert  -1.959e-02  3.529e-02  -0.555 0.579005
NeighborhoodIDOTRR   -3.984e-02  4.939e-02  -0.807 0.420015
NeighborhoodMeadowV  -1.757e-01  5.147e-02  -3.413 0.000664 ***
NeighborhoodMitchel  -6.159e-02  3.756e-02  -1.640 0.101326
NeighborhoodNAmes    -4.695e-02  3.601e-02  -1.304 0.192526
```

```
NeighborhoodNoRidge   2.444e-02  3.872e-02   0.631 0.528022
NeighborhoodNPkVill  -2.902e-03  6.458e-02  -0.045 0.964166
NeighborhoodNridgHt   6.902e-02  3.453e-02   1.999 0.045821 *
NeighborhoodNWAmes   -4.847e-02  3.681e-02  -1.317 0.188255
NeighborhoodOldTown  -5.757e-02  4.439e-02  -1.297 0.194911
NeighborhoodSawyer   -3.702e-02  3.732e-02  -0.992 0.321438
NeighborhoodSawyerW  -1.181e-02  3.576e-02  -0.330 0.741256
NeighborhoodSomerst   1.374e-02  4.141e-02   0.332 0.740030
NeighborhoodStoneBr   1.293e-01  3.810e-02   3.394 0.000711 ***
NeighborhoodSWISU    -4.931e-03  4.461e-02  -0.111 0.911999
NeighborhoodTimber   -3.797e-03  3.742e-02  -0.101 0.919178
NeighborhoodVeenker   3.931e-02  4.823e-02   0.815 0.415143
Condition1Feedr       4.138e-02  2.307e-02   1.794 0.073130 .
Condition1Norm        9.012e-02  1.925e-02   4.681 3.18e-06 ***
Condition1PosA        3.325e-02  4.600e-02   0.723 0.469944
Condition1PosN        9.353e-02  3.419e-02   2.736 0.006314 **
Condition1RRAe       -3.416e-02  4.169e-02  -0.819 0.412769
Condition1RRAn        6.216e-02  3.197e-02   1.944 0.052080 .
Condition1RRNe        1.618e-02  8.035e-02   0.201 0.840386
Condition1RRNn        8.959e-02  5.903e-02   1.518 0.129370
Condition2Feedr       1.110e-01  1.077e-01   1.031 0.302944
Condition2Norm        4.470e-02  9.328e-02   0.479 0.631856
Condition2PosA        2.450e-01  1.703e-01   1.439 0.150317
Condition2PosN       -8.353e-01  1.271e-01  -6.575 7.24e-11 ***
Condition2RRAe       -6.140e-01  2.993e-01  -2.052 0.040417 *
Condition2RRAn       -5.490e-02  1.448e-01  -0.379 0.704620
Condition2RRNn        3.077e-02  1.246e-01   0.247 0.804949
BldgType2fmCon        4.931e-02  5.744e-02   0.858 0.390804
BldgTypeDuplex        1.341e-03  3.415e-02   0.039 0.968676
BldgTypeTwnhs        -4.338e-02  4.608e-02  -0.941 0.346776
BldgTypeTwnhsE       -8.400e-03  4.156e-02  -0.202 0.839845
HouseStyle1.5Unf      6.801e-03  3.663e-02   0.186 0.852722
HouseStyle1Story     -3.043e-02  2.019e-02  -1.508 0.131933
HouseStyle2.5Fin     -9.947e-02  5.694e-02  -1.747 0.080922 .
HouseStyle2.5Unf      3.086e-02  4.248e-02   0.726 0.467714
HouseStyle2Story     -2.330e-02  1.609e-02  -1.448 0.147944
HouseStyleSFoyer     -1.109e-02  2.880e-02  -0.385 0.700316
HouseStyleSLvl       -4.613e-03  2.569e-02  -0.180 0.857514
OverallQual           4.098e-02  4.667e-03   8.781  < 2e-16 ***
OverallCond           3.641e-02  4.010e-03   9.081  < 2e-16 ***
YearBuilt             1.728e-03  3.546e-04   4.872 1.25e-06 ***
YearRemodAdd          8.581e-04  2.567e-04   3.342 0.000857 ***
RoofStyleGable        8.757e-03  8.481e-02   0.103 0.917779
RoofStyleGambrel      1.017e-02  9.286e-02   0.110 0.912788
RoofStyleHip          1.113e-02  8.515e-02   0.131 0.896015
RoofStyleMansard      6.020e-02  9.844e-02   0.612 0.540959
RoofStyleShed         4.544e-01  1.588e-01   2.861 0.004292 **
RoofMatlCompShg       2.574e+00  2.427e-01  10.604  < 2e-16 ***
RoofMatlMembran       2.980e+00  2.881e-01  10.343  < 2e-16 ***
RoofMatlMetal         2.851e+00  2.864e-01   9.955  < 2e-16 ***
RoofMatlRoll          2.596e+00  2.684e-01   9.672  < 2e-16 ***
RoofMatlTar&Grv       2.579e+00  2.601e-01   9.913  < 2e-16 ***
RoofMatlWdShake       2.519e+00  2.536e-01   9.934  < 2e-16 ***
RoofMatlWdShngl       2.628e+00  2.471e-01  10.635  < 2e-16 ***
```

```
Exterior1stAsphShn   -1.485e-02  1.518e-01  -0.098 0.922062
Exterior1stBrkComm   -2.172e-01  1.278e-01  -1.700 0.089412 .
Exterior1stBrkFace    7.357e-02  5.875e-02   1.252 0.210688
Exterior1stCBlock    -5.505e-02  1.255e-01  -0.439 0.660883
Exterior1stCemntBd   -1.037e-01  8.758e-02  -1.184 0.236652
Exterior1stHdBoard   -2.794e-02  5.956e-02  -0.469 0.639059
Exterior1stImStucc    1.140e-02  1.296e-01   0.088 0.929885
Exterior1stMetalSd    3.492e-02  6.719e-02   0.520 0.603370
Exterior1stPlywood   -1.990e-02  5.878e-02  -0.339 0.734963
Exterior1stStone      8.878e-03  1.117e-01   0.079 0.936688
Exterior1stStucco     7.115e-03  6.480e-02   0.110 0.912595
Exterior1stVinylSd   -1.891e-02  6.139e-02  -0.308 0.758056
Exterior1stWd Sdng   -5.558e-02  5.698e-02  -0.975 0.329565
Exterior1stWdShing   -1.633e-02  6.151e-02  -0.266 0.790666
Exterior2ndAsphShn    6.796e-02  1.022e-01   0.665 0.506074
Exterior2ndBrk Cmn    5.993e-02  9.233e-02   0.649 0.516411
Exterior2ndBrkFace   -1.284e-02  6.084e-02  -0.211 0.832942
Exterior2ndCBlock            NA         NA      NA       NA
Exterior2ndCmentBd    1.580e-01  8.612e-02   1.835 0.066816 .
Exterior2ndHdBoard    4.300e-02  5.717e-02   0.752 0.452140
Exterior2ndImStucc    5.003e-02  6.603e-02   0.758 0.448778
Exterior2ndMetalSd    9.421e-03  6.538e-02   0.144 0.885443
Exterior2ndOther     -6.017e-02  1.246e-01  -0.483 0.629248
Exterior2ndPlywood    4.046e-02  5.551e-02   0.729 0.466245
Exterior2ndStone     -1.957e-02  7.888e-02  -0.248 0.804085
Exterior2ndStucco     3.799e-02  6.266e-02   0.606 0.544398
Exterior2ndVinylSd    5.841e-02  5.899e-02   0.990 0.322287
Exterior2ndWd Sdng    7.847e-02  5.499e-02   1.427 0.153854
Exterior2ndWd Shng    4.124e-02  5.735e-02   0.719 0.472278
MasVnrTypeBrkFace     3.817e-02  3.144e-02   1.214 0.225007
MasVnrTypeNone        2.946e-02  3.177e-02   0.927 0.354080
MasVnrTypeStone       4.872e-02  3.330e-02   1.463 0.143662
MasVnrArea            7.443e-06  2.662e-05   0.280 0.779878
ExterQualFa           1.596e-02  5.100e-02   0.313 0.754310
ExterQualGd          -1.074e-05  2.199e-02   0.000 0.999610
ExterQualTA           7.744e-03  2.438e-02   0.318 0.750811
ExterCondFa          -1.048e-01  8.312e-02  -1.260 0.207769
ExterCondGd          -8.168e-02  7.926e-02  -1.030 0.303010
ExterCondPo          -4.523e-02  1.457e-01  -0.310 0.756339
ExterCondTA          -5.919e-02  7.911e-02  -0.748 0.454487
FoundationCBlock      2.011e-02  1.459e-02   1.378 0.168400
FoundationPConc       3.822e-02  1.572e-02   2.431 0.015208 *
FoundationSlab       -2.265e-02  4.621e-02  -0.490 0.624161
FoundationStone       1.065e-01  5.245e-02   2.030 0.042604 *
FoundationWood       -1.217e-01  6.796e-02  -1.791 0.073554 .
BsmtQualFa           -2.366e-02  2.923e-02  -0.809 0.418551
BsmtQualGd           -2.717e-02  1.535e-02  -1.771 0.076889 .
BsmtQualNone          1.729e-01  1.684e-01   1.027 0.304792
BsmtQualTA           -2.898e-02  1.913e-02  -1.515 0.130012
BsmtCondGd            2.734e-02  2.427e-02   1.127 0.260100
BsmtCondNone                 NA         NA      NA       NA
BsmtCondPo            2.987e-01  1.372e-01   2.177 0.029703 *
BsmtCondTA            2.327e-02  1.953e-02   1.192 0.233601
BsmtExposureGd        3.179e-02  1.379e-02   2.306 0.021277 *
```

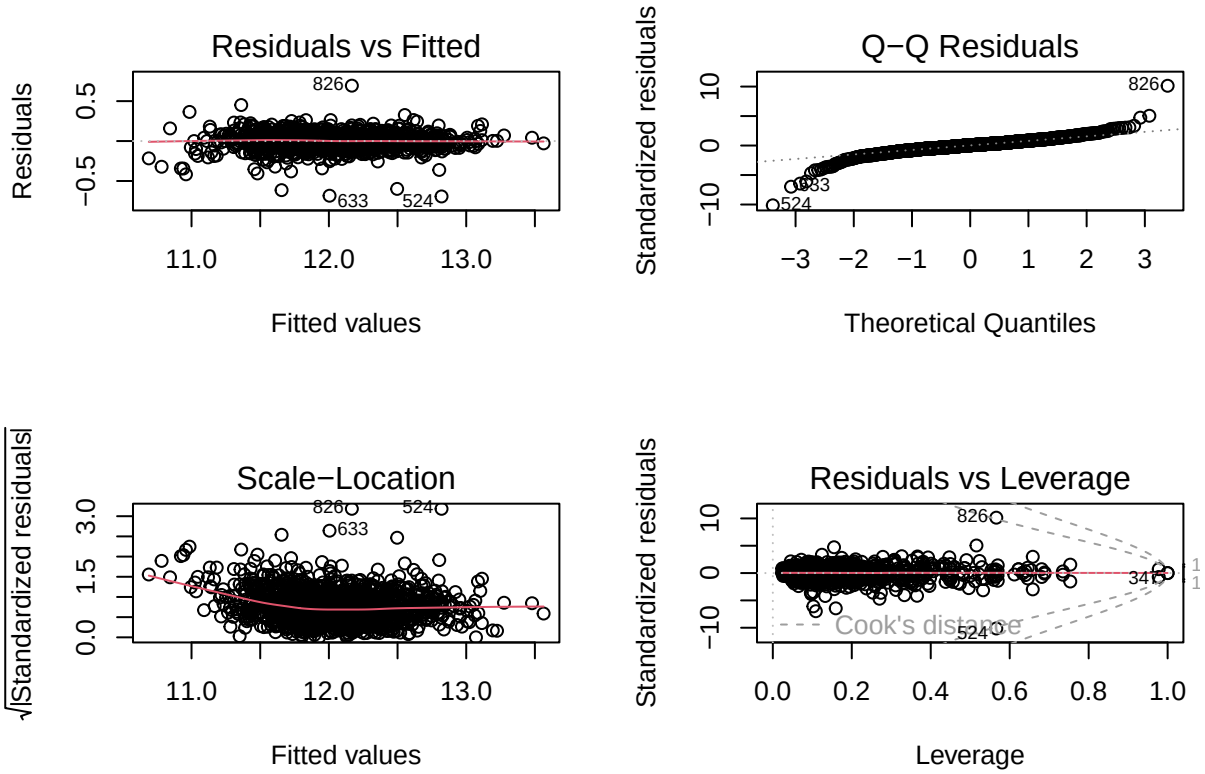
```
BsmtExposureMn       -6.844e-03  1.388e-02  -0.493 0.622123
BsmtExposureNo       -1.109e-02  1.003e-02  -1.106 0.268811
BsmtExposureNone     -5.065e-02  1.057e-01  -0.479 0.632025
BsmtFinType1BLQ      -8.514e-04  1.290e-02  -0.066 0.947366
BsmtFinType1GLQ       1.402e-02  1.159e-02   1.209 0.226801
BsmtFinType1LwQ      -2.505e-02  1.722e-02  -1.455 0.145898
BsmtFinType1None             NA         NA      NA       NA
BsmtFinType1Rec      -8.571e-03  1.380e-02  -0.621 0.534718
BsmtFinType1Unf      -1.133e-02  1.341e-02  -0.845 0.398325
BsmtFinSF1            1.468e-04  2.452e-05   5.988 2.79e-09 ***
BsmtFinType2BLQ      -7.084e-02  3.480e-02  -2.036 0.041979 *
BsmtFinType2GLQ       5.099e-03  4.300e-02   0.119 0.905620
BsmtFinType2LwQ      -3.413e-02  3.402e-02  -1.003 0.315913
BsmtFinType2None     -1.491e-01  1.149e-01  -1.298 0.194612
BsmtFinType2Rec      -3.313e-02  3.274e-02  -1.012 0.311824
BsmtFinType2Unf      -1.930e-02  3.486e-02  -0.554 0.579814
BsmtFinSF2            1.391e-04  4.166e-05   3.338 0.000869 ***
BsmtUnfSF             8.922e-05  2.249e-05   3.967 7.70e-05 ***
TotalBsmtSF                  NA         NA      NA       NA
HeatingGasA           1.335e-01  1.176e-01   1.135 0.256574
HeatingGasW           2.028e-01  1.213e-01   1.672 0.094733 .
HeatingGrav          -4.789e-02  1.290e-01  -0.371 0.710477
HeatingOthW           1.570e-01  1.448e-01   1.084 0.278574
HeatingWall           2.066e-01  1.367e-01   1.511 0.131102
HeatingQCFa          -1.684e-02  2.169e-02  -0.776 0.437616
HeatingQCGd          -2.178e-02  9.534e-03  -2.285 0.022502 *
HeatingQCPo          -1.041e-01  1.221e-01  -0.852 0.394228
HeatingQCTA          -3.374e-02  9.520e-03  -3.544 0.000409 ***
CentralAirY           6.160e-02  1.779e-02   3.463 0.000553 ***
ElectricalFuseF      -7.749e-03  2.645e-02  -0.293 0.769550
ElectricalFuseP      -9.050e-02  8.560e-02  -1.057 0.290611
ElectricalMix        -2.461e-01  2.046e-01  -1.203 0.229309
ElectricalNone        8.101e-02  1.108e-01   0.731 0.464741
ElectricalSBrkr      -1.572e-02  1.356e-02  -1.159 0.246694
X1stFlrSF             2.294e-04  2.596e-05   8.837  < 2e-16 ***
X2ndFlrSF             2.251e-04  2.621e-05   8.588  < 2e-16 ***
LowQualFinSF          1.939e-04  8.762e-05   2.213 0.027117 *
GrLivArea                    NA         NA      NA       NA
BsmtFullBath          2.501e-02  9.106e-03   2.746 0.006120 **
BsmtHalfBath          5.281e-03  1.392e-02   0.379 0.704532
FullBath              1.873e-02  1.012e-02   1.852 0.064300 .
HalfBath              2.254e-02  9.642e-03   2.338 0.019565 *
BedroomAbvGr          6.370e-03  6.277e-03   1.015 0.310424
KitchenAbvGr         -4.455e-02  2.616e-02  -1.703 0.088828 .
KitchenQualFa        -5.868e-02  2.852e-02  -2.057 0.039865 *
KitchenQualGd        -6.626e-02  1.601e-02  -4.139 3.73e-05 ***
KitchenQualTA        -6.655e-02  1.807e-02  -3.682 0.000241 ***
TotRmsAbvGrd          4.456e-03  4.397e-03   1.014 0.310984
FunctionalMaj2       -2.571e-01  6.611e-02  -3.889 0.000106 ***
FunctionalMin1        3.961e-02  3.955e-02   1.002 0.316731
FunctionalMin2        2.245e-02  3.967e-02   0.566 0.571565
FunctionalMod        -6.506e-02  4.853e-02  -1.341 0.180262
FunctionalSev        -2.659e-01  1.360e-01  -1.956 0.050695 .
FunctionalTyp         6.100e-02  3.429e-02   1.779 0.075478 .
```

```
Fireplaces           1.167e-02  1.175e-02   0.993 0.321068
FireplaceQuFa       -4.412e-03  3.167e-02  -0.139 0.889236
FireplaceQuGd        1.873e-02  2.447e-02   0.766 0.444002
FireplaceQuNone      1.062e-03  2.864e-02   0.037 0.970424
FireplaceQuPo        3.868e-02  3.640e-02   1.063 0.288101
FireplaceQuTA        1.919e-02  2.544e-02   0.754 0.450713
GarageTypeAttchd     1.100e-01  5.070e-02   2.170 0.030226 *
GarageTypeBasment    1.107e-01  5.881e-02   1.883 0.059992 .
GarageTypeBuiltIn    9.351e-02  5.287e-02   1.769 0.077158 .
GarageTypeCarPort    1.312e-01  6.759e-02   1.941 0.052514 .
GarageTypeDetchd     1.081e-01  5.072e-02   2.131 0.033291 *
GarageTypeNone       3.323e-02  9.572e-02   0.347 0.728509
GarageYrBlt         -2.876e-04  2.819e-04  -1.020 0.307899
GarageFinishNone            NA         NA      NA       NA
GarageFinishRFn      2.026e-03  9.019e-03   0.225 0.822302
GarageFinishUnf     -9.433e-03  1.117e-02  -0.844 0.398588
GarageCars           1.631e-02  1.048e-02   1.557 0.119841
GarageArea           1.291e-04  3.633e-05   3.553 0.000396 ***
GarageQualFa        -3.922e-01  1.387e-01  -2.828 0.004756 **
GarageQualGd        -3.322e-01  1.424e-01  -2.333 0.019810 *
GarageQualNone              NA         NA      NA       NA
GarageQualPo        -4.291e-01  1.768e-01  -2.427 0.015357 *
GarageQualTA        -3.404e-01  1.373e-01  -2.479 0.013328 *
GarageCondFa         2.807e-01  1.600e-01   1.755 0.079520 .
GarageCondGd         3.098e-01  1.662e-01   1.864 0.062504 .
GarageCondNone              NA         NA      NA       NA
GarageCondPo         4.423e-01  1.717e-01   2.576 0.010104 *
GarageCondTA         3.026e-01  1.586e-01   1.908 0.056575 .
PavedDriveP         -1.284e-02  2.553e-02  -0.503 0.615043
PavedDriveY          1.116e-02  1.592e-02   0.701 0.483220
WoodDeckSF           9.624e-05  2.703e-05   3.561 0.000384 ***
OpenPorchSF          3.265e-05  5.323e-05   0.613 0.539750
EnclosedPorch        1.231e-04  5.739e-05   2.146 0.032109 *
X3SsnPorch           1.493e-04  1.030e-04   1.450 0.147261
ScreenPorch          2.779e-04  5.748e-05   4.834 1.51e-06 ***
PoolArea             1.684e-03  1.043e-03   1.615 0.106638
PoolQCFa            -1.353e-01  1.882e-01  -0.719 0.472308
PoolQCGd             3.390e-02  1.696e-01   0.200 0.841579
PoolQCNone           8.750e-01  5.644e-01   1.550 0.121313
FenceGdWo           -2.448e-02  2.257e-02  -1.085 0.278194
FenceMnPrv           3.115e-03  1.842e-02   0.169 0.865752
FenceMnWw           -9.974e-03  3.778e-02  -0.264 0.791809
FenceNone            1.445e-02  1.689e-02   0.856 0.392273
MiscFeatureNone     -1.380e-01  4.474e-01  -0.308 0.757834
MiscFeatureOthr     -1.698e-01  4.177e-01  -0.407 0.684356
MiscFeatureShed     -1.392e-01  4.286e-01  -0.325 0.745392
MiscFeatureTenC     -1.407e-01  4.447e-01  -0.316 0.751712
MiscVal             -6.271e-06  2.814e-05  -0.223 0.823655
MoSold              -5.022e-04  1.127e-03  -0.446 0.655999
YrSold              -2.363e-03  2.369e-03  -0.997 0.318861
SaleTypeCon          1.001e-01  8.067e-02   1.241 0.214985
SaleTypeConLD        1.307e-01  4.455e-02   2.933 0.003419 **
SaleTypeConLI       -3.757e-02  5.312e-02  -0.707 0.479574
SaleTypeConLw        1.635e-02  5.587e-02   0.293 0.769883
```

```
SaleTypeCWD           6.265e-02  5.917e-02   1.059 0.289902
SaleTypeNew           7.147e-02  7.092e-02   1.008 0.313773
SaleTypeOth           6.618e-02  6.667e-02   0.993 0.321088
SaleTypeWD           -1.761e-02  1.921e-02  -0.917 0.359572
SaleConditionAdjLand  1.157e-01  6.716e-02   1.723 0.085109 .
SaleConditionAlloca   4.220e-02  4.079e-02   1.035 0.301087
SaleConditionFamily   1.332e-02  2.800e-02   0.475 0.634522
SaleConditionNormal   6.199e-02  1.336e-02   4.640 3.86e-06 ***
SaleConditionPartial  1.805e-02  6.825e-02   0.264 0.791457
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1039 on 1206 degrees of freedom
Multiple R-squared:  0.9441,    Adjusted R-squared:  0.9323
F-statistic: 80.43 on 253 and 1206 DF,  p-value: < 2.2e-16
```

The log-transformed regression model, with log(SalePrice) as the response variable, shows an excellent fit with a Multiple R-squared of 0.9441 and an Adjusted R-squared of 0.9323, indicating that approximately 94% of the variation in the log sale price is explained by the predictors. The overall model is highly significant (F-statistic = 80.43, $p < 2.2 \times 10^{-16}$), and the residual standard error is relatively low (0.1039), suggesting a precise fit. Notably, several predictors, such as the different levels of MSZoning and LotArea, are statistically significant and contribute meaningfully to explaining sale price variability. The log transformation appears to have stabilized variance and improved normality in the residuals, addressing potential issues of heteroskedasticity and skewness that were observed in the original scale. Some predictors were omitted due to singularities, but overall the model provides a robust framework for understanding the determinants of sale price on the log scale. The model is linear in its formulation—it assumes that a linear combination of the predictors explains the (transformed) response. In our case, we modeled log (SalePrice) log(SalePrice) as a linear function of the predictors, so the relationship is linear on the log scale. This means that changes in the predictors are assumed to have a constant proportional effect on SalePrice.

```r
# Plot diagnostic plots for the log-transformed model (base R)
par(mfrow = c(2, 2))
plot(model_log)
```



The Residuals vs. Fitted plot (top-left) shows how residuals deviate from zero across different fitted values; ideally, the red smoothing line should be flat, indicating no systematic pattern. The Q–Q Residuals plot (top-right) checks the normality of residuals; here, points closely follow the diagonal except for mild deviations in the tails. The Scale-Location plot (bottom-left) helps evaluate homoscedasticity (constant variance); the near-horizontal red line suggests no severe heteroskedasticity. Finally, the Residuals vs. Leverage plot (bottom-right) identifies potentially influential observations—such as #8260—whose leverage or Cook's distance values are high.

(c) Regularization

```
set.seed(123)
X_mat <- model.matrix(SalePrice ~ . - 1, data = df_clean)
y_vec <- log(df_clean$SalePrice)

# Number of observations and total sum of squares for y
n <- length(y_vec)
SST <- sum((y_vec - mean(y_vec))^2)

# ----------------------------
# 1. SCAD Regularization
# ----------------------------

# Fit the full solution path using ncvreg()
scad_fit <- ncvreg(X_mat, y_vec, penalty = "SCAD")
# Perform cross-validation over the full solution path
cv_scad <- cv.ncvreg(X_mat, y_vec, penalty = "SCAD")
best_lambda_scad <- cv_scad$lambda.min
# Extract coefficients at the best lambda from the full path
scad_coefs <- coef(scad_fit, lambda = best_lambda_scad)
```

```r
# Compute predictions and performance metrics
scad_pred <- predict(scad_fit, X_mat, lambda = best_lambda_scad)
scad_mse <- mean((y_vec - scad_pred)^2)
scad_R2 <- 1 - sum((y_vec - scad_pred)^2) / SST
scad_nonzero <- sum(scad_coefs != 0) - 1  # subtract intercept
scad_adj_R2 <- 1 - (1 - scad_R2) * (n - 1) / (n - scad_nonzero - 1)

# ----------------------------
# 2. glmnet Regularization
# ----------------------------

# Create a design matrix for glmnet (automatically one-hot encodes factors)
X <- model.matrix(SalePrice ~ ., data = df_clean)[, -1]
y <- log(df_clean$SalePrice)

# --- LASSO ---
cv_lasso <- cv.glmnet(X, y, alpha = 1)
best_lambda_lasso <- cv_lasso$lambda.min
lasso_model <- glmnet(X, y, alpha = 1, lambda = best_lambda_lasso)
lasso_pred <- predict(lasso_model, X)
lasso_mse <- mean((y - lasso_pred)^2)
nonzero_lasso <- sum(coef(lasso_model) != 0) - 1  # subtract intercept
lasso_R2 <- 1 - sum((y - lasso_pred)^2) / sum((y - mean(y))^2)
lasso_adj_R2 <- 1 - (1 - lasso_R2) * (n - 1) / (n - nonzero_lasso - 1)

# --- Ridge ---
cv_ridge <- cv.glmnet(X, y, alpha = 0)
best_lambda_ridge <- cv_ridge$lambda.min
ridge_model <- glmnet(X, y, alpha = 0, lambda = best_lambda_ridge)
ridge_pred <- predict(ridge_model, X)
ridge_mse <- mean((y - ridge_pred)^2)
nonzero_ridge <- sum(coef(ridge_model) != 0) - 1
ridge_R2 <- 1 - sum((y - ridge_pred)^2) / sum((y - mean(y))^2)
ridge_adj_R2 <- 1 - (1 - ridge_R2) * (n - 1) / (n - nonzero_ridge - 1)

# --- Elastic Net (with different alphas) ---
alphas <- c(0.25, 0.5, 0.75)
mse_vec <- numeric(length(alphas))
nonzero_vec <- numeric(length(alphas))
adjR2_vec <- numeric(length(alphas))
enet_models <- list()

for(i in seq_along(alphas)){
  cv_enet <- cv.glmnet(X, y, alpha = alphas[i])
  best_lambda_enet <- cv_enet$lambda.min
  enet_model <- glmnet(X, y, alpha = alphas[i], lambda = best_lambda_enet)
  enet_models[[i]] <- enet_model
  pred_enet <- predict(enet_model, X)
  mse_vec[i] <- mean((y - pred_enet)^2)
  nonzero_vec[i] <- sum(coef(enet_model) != 0) - 1
  R2_enet <- 1 - sum((y - pred_enet)^2) / sum((y - mean(y))^2)
  adjR2_vec[i] <- 1 - (1 - R2_enet) * (n - 1) / (n - nonzero_vec[i] - 1)
}
```

```r
# ---------------------------
# 3. Summarize Regularization Results
# ---------------------------
results <- data.frame(
  Method = c("SCAD", "LASSO", "Ridge",
             paste0("Elastic Net (alpha=", alphas, ")")),
  MSE = c(scad_mse, lasso_mse, ridge_mse, mse_vec),
  Nonzero_Coeffs = c(scad_nonzero, nonzero_lasso, nonzero_ridge, nonzero_vec),
  Adjusted_R2 = c(scad_adj_R2, lasso_adj_R2, ridge_adj_R2, adjR2_vec)
)
print(results)
```

```
                   Method        MSE Nonzero_Coeffs Adjusted_R2
1                    SCAD 0.01421145             64   0.9067845
2                   LASSO 0.01487999             83   0.9010518
3                   Ridge 0.01437447            261   0.8902110
4 Elastic Net (alpha=0.25) 0.01595901             84   0.8937993
5  Elastic Net (alpha=0.5) 0.01440898             96   0.9032700
6 Elastic Net (alpha=0.75) 0.01462632             85   0.9025970
```

Based on repeated runs of the regularization code, the SCAD offer best MSE & Adjusted R^2 with fewest variables, it balances predictive performance and simplicity, which is ideal for interpretability in a high-dimensional setting.

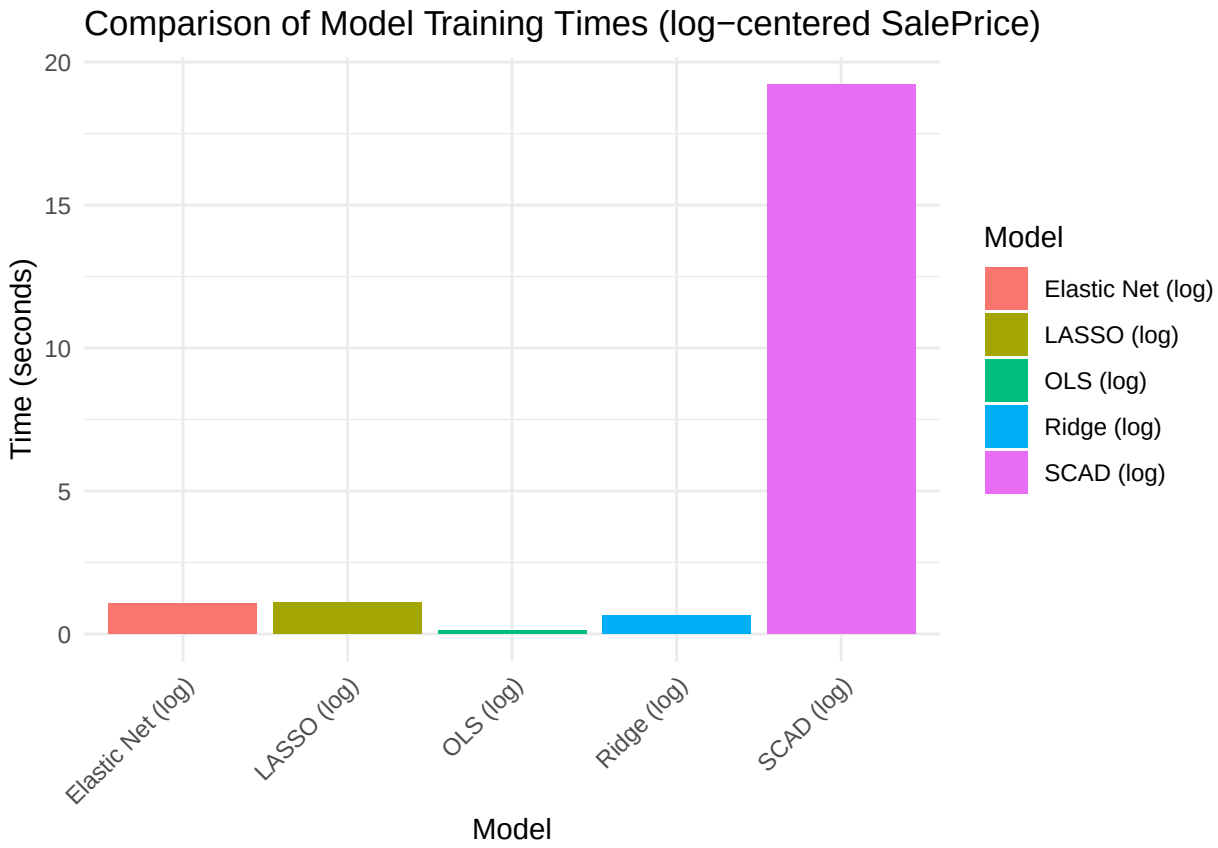
```r
# ---------------------------------------
# Timing OLS model fitting on log(SalePrice)
# ---------------------------------------
ols_time <- system.time({
  model_ols_timed <- lm(log(SalePrice) ~ ., data = df_clean)
})

# ---------------------------------------
# Timing SCAD model fitting on log(SalePrice)
# ---------------------------------------
scad_time <- system.time({
  scad_fit_timed <- ncvreg(X_mat, y_vec, penalty = "SCAD")
  cv_scad_timed <- cv.ncvreg(X_mat, y_vec, penalty = "SCAD")
})

# ---------------------------------------
# Timing LASSO model fitting on log(SalePrice)
# ---------------------------------------
lasso_time <- system.time({
  cv_lasso_timed <- cv.glmnet(X, y_vec, alpha = 1)
  best_lambda_lasso_timed <- cv_lasso_timed$lambda.min
  lasso_model_timed <- glmnet(X, y_vec, alpha = 1, lambda = best_lambda_lasso_timed)
})

# ---------------------------------------
# Timing Ridge model fitting on log(SalePrice)
# ---------------------------------------
ridge_time <- system.time({
  cv_ridge_timed <- cv.glmnet(X, y_vec, alpha = 0)
  best_lambda_ridge_timed <- cv_ridge_timed$lambda.min
```

```
  ridge_model_timed <- glmnet(X, y_vec, alpha = 0, lambda = best_lambda_ridge_timed)
})

# ----------------------------------------
# Timing Elastic Net model fitting (alpha = 0.5) on log(SalePrice)
# ----------------------------------------
enet_time <- system.time({
  cv_enet_timed <- cv.glmnet(X, y_vec, alpha = 0.5)
  best_lambda_enet_timed <- cv_enet_timed$lambda.min
  enet_model_timed <- glmnet(X, y_vec, alpha = 0.5, lambda = best_lambda_enet_timed)
})
```

```
# Create a dataframe summarizing training time
timing_results_log <- data.frame(
  Model = c("OLS (log)", "SCAD (log)", "LASSO (log)", "Ridge (log)", "Elastic Net (log)"),
  Time_Seconds = c(
    ols_time["elapsed"],
    scad_time["elapsed"],
    lasso_time["elapsed"],
    ridge_time["elapsed"],
    enet_time["elapsed"]
  )
)

# Print timing table
print(timing_results_log)
```

```
              Model Time_Seconds
1         OLS (log)        0.122
2        SCAD (log)       19.205
3       LASSO (log)        1.093
4       Ridge (log)        0.635
5 Elastic Net (log)        1.069
```

```
# Plot the computation time
ggplot(timing_results_log, aes(x = Model, y = Time_Seconds, fill = Model)) +
  geom_bar(stat = "identity") +
  labs(title = "Comparison of Model Training Times (log-centered SalePrice)",
       x = "Model",
       y = "Time (seconds)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



As shown in the bar chart, the SCAD model—while offering strong performance in terms of feature selection—was by far the most computationally expensive, taking over 19 seconds to train on the log-transformed sale price. In contrast, OLS (log) was the fastest at 0.14 seconds, confirming its efficiency even in high-dimensional settings. LASSO and Elastic Net both took around 1–1.2 seconds, reflecting the extra cost of regularization and cross-validation. Ridge, slightly faster at 0.66 seconds, strikes a good balance. Overall, this highlights that SCAD's power comes with a significant time cost, whereas OLS and Ridge are much quicker, and LASSO/Elastic Net offer a good compromise between sparsity and speed.

```
# Convert the SCAD coefficient matrix to a regular matrix
scad_mat <- as.matrix(scad_coefs)

# Remove the intercept row (assuming it's labeled as "(Intercept)")
scad_mat <- scad_mat[rownames(scad_mat) != "(Intercept)", , drop = FALSE]

# Create a data frame with predictor names and their coefficient values
scad_df <- data.frame(Predictor = rownames(scad_mat),
                      Coefficient = scad_mat[, 1],
                      row.names = NULL)

# Filter to only include nonzero coefficients
nonzero_scad_df <- scad_df[scad_df$Coefficient != 0, ]

cat("Predictors selected by SCAD:\n")
```

```
Predictors selected by SCAD:
```

```
print(nonzero_scad_df, row.names = FALSE)
```

```
           Predictor   Coefficient
```

```
          MSSubClass -6.637557e-04
    MSZoningC (all) -3.551479e-01
          MSZoningRM -7.030768e-02
             LotArea  1.390872e-06
         LotShapeIR3 -5.852778e-02
   LotConfigCulDSac  3.790612e-02
NeighborhoodBrkSide  7.850048e-03
NeighborhoodClearCr  2.543631e-02
NeighborhoodCrawfor  1.286074e-01
NeighborhoodEdwards -3.238641e-02
NeighborhoodMeadowV -2.467861e-02
NeighborhoodNoRidge  1.319545e-02
NeighborhoodNridgHt  1.278145e-01
NeighborhoodSomerst  6.169770e-02
NeighborhoodStoneBr  1.329889e-01
     Condition1Norm  5.427413e-02
     Condition1PosN  1.691367e-03
     Condition2PosN -7.239664e-01
     BldgType2fmCon  2.027395e-02
      BldgTypeTwnhs -2.878681e-02
     HouseStyleSLvl  1.261220e-02
        OverallQual  6.592015e-02
        OverallCond  4.680319e-02
          YearBuilt  2.149076e-03
       YearRemodAdd  1.738916e-04
    RoofMatlWdShngl  2.618249e-02
 Exterior1stBrkComm -8.926546e-02
 Exterior1stBrkFace  4.955515e-02
 Exterior1stHdBoard -5.231296e-03
 Exterior1stMetalSd  5.456862e-04
  Exterior2ndStucco -3.342453e-02
    FoundationPConc  4.654341e-02
     FoundationSlab -2.442441e-03
    FoundationStone  1.958922e-02
         BsmtCondTA  3.439527e-03
     BsmtExposureGd  6.160535e-02
     BsmtExposureNo -2.895232e-04
    BsmtFinType1Unf -3.460829e-02
    BsmtFinType2BLQ -2.886821e-04
   BsmtFinType2None -3.328811e-02
    BsmtFinType2Unf  4.790710e-05
        TotalBsmtSF  5.042875e-05
        HeatingGasW  4.870945e-02
        HeatingGrav -7.887470e-02
        HeatingQCTA -5.402118e-03
        CentralAirY  4.388938e-02
          GrLivArea  2.693761e-04
       BsmtFullBath  5.458034e-02
     FunctionalMaj2 -1.168358e-01
      FunctionalSev -1.387007e-01
      FunctionalTyp  4.339144e-02
    FireplaceQuNone -4.554781e-02
         GarageCars  6.652760e-02
       GarageQualGd  3.080701e-02
```

```
        GarageCondFa -8.299935e-03
          WoodDeckSF  8.888454e-05
       EnclosedPorch  2.372720e-05
         ScreenPorch  2.076004e-04
            PoolArea  1.687511e-05
            PoolQCGd -5.132286e-01
           FenceGdWo -3.935836e-03
      SaleTypeConLD  5.208379e-02
        SaleTypeNew  1.138991e-01
SaleConditionNormal  5.667769e-02
```

(d) A comparison of the OLS and regularized models

Based on the model comparisons performed on the log-transformed SalePrice, the Ordinary Least Squares (OLS) regression achieved the best overall performance, with the lowest Mean Squared Error (MSE ≈ 0.0108) and the highest Adjusted R^2 (0.9323). It was also the fastest model to train (0.14 seconds), demonstrating excellent computational efficiency. However, OLS does not induce sparsity, utilizing nearly all predictors, which could complicate model interpretability. Among the regularized models, SCAD achieved strong sparsity (only 64 nonzero coefficients) and relatively good predictive accuracy (MSE $\approx$ 0.0142), but it was significantly slower to train (19.17 seconds). LASSO and Elastic Net ($\alpha = 0.5$) offered balanced alternatives, maintaining moderate sparsity (83–96 predictors), good Adjusted R^2 (around 0.90), and faster training times (~1 second). Overall, if computational time and pure predictive accuracy are prioritized, **OLS remains the most efficient and accurate model** for this high-dimensional house price dataset. However, **if interpretability and model simplicity are privileged, SCAD or LASSO represent better choices**, offering a substantial reduction in the number of predictors while maintaining strong predictive performance.