****

**Superior University Lahore**

**Programming for AI (Lab)**

**Lab Task 1**

**House Price Prediction - Kaggle Competition**

**BS in Artificial Intelligence**

*Department of Software Engineering*

*Faculty of Computer Science & Information Technology*

*The Superior University, Lahore*

**Submitted by:**

|  |  |
| --- | --- |
| **Student Name:** | **Alisha Kanwal** |
| **Roll Number** | **Su92-bsaim-f23-105** |
| **Section** | **BSAI-4B** |

**Contents**

[**1.** **Introduction:** 4](#_Toc191117389)

[1.1 Problem Statement: 4](#_Toc191117390)

[1.2 Domain Understanding: 4](#_Toc191117391)

[**2.** **Dataset Overview:** 4](#_Toc191117392)

[2.1 Loading the Data: 4](#_Toc191117395)

[2.2 Exploring the Data: 4](#_Toc191117396)

[**3.** **Data Preprocessing:** 4](#_Toc191117397)

[3.1 Handling Missing Values: 4](#_Toc191117399)

[3.2 Encoding Categorical Data: 4](#_Toc191117401)

[3.3 Splitting Data into Test and Train Sets: 4](#_Toc191117402)

[**4.** **Model Implementation:** 4](#_Toc191117403)

[4.3 XG Boost Classifier 4](#_Toc191117407)

[**5.** **Sample Submission File Saving:** 4](#_Toc191117408)

[**6.** **Results** 4](#_Toc191117411)

**Report**

# **Introduction:**

## Problem Statement:

The goal of this competition is to predict the SalePrice of houses in Ames, Iowa, based on various features such as lot size, neighborhood, building type, and other house characteristics. The dataset is a modern alternative to the Boston Housing dataset and contains detailed information about the properties.

## Problem Understanding:

This is a regression problem, where the target variable (SalePrice) is continuous. The evaluation metric is Root-Mean-Squared-Error (RMSE) on the logarithm of the predicted and actual values, which ensures that errors in predicting expensive and inexpensive houses are treated equally. The challenge involves data preprocessing, feature engineering, and advanced regression techniques to build a model that generalizes well.

# **Dataset Overview:**



## Loading the Data:

The dataset consists of a **training set (train.csv)** and a **test set (test.csv)**. The training set contains the **SalePrice** column, which we use as the target variable, while the test set lacks this column, requiring us to make predictions.

The files used in this competition:

* train.csv - Training dataset containing features and house prices.
* test.csv - Test dataset without house prices.
* sample\_submission.csv - A benchmark submission file.
* data\_description.txt - Provides metadata about each feature.

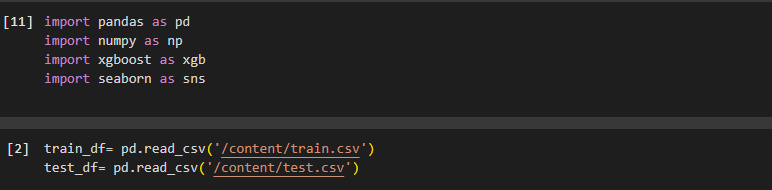


Figure 2.1.1 Importing Modules/Libraries and Loading Dataset

## Exploring the Data:

The dataset includes numerical and categorical features describing different aspects of houses. Some key attributes include:

* **Numerical Features**: LotArea, YearBuilt, TotalBsmtSF, GrLivArea, GarageCars, etc.
* **Categorical Features**: Neighborhood, HouseStyle, RoofStyle, Heating, etc.

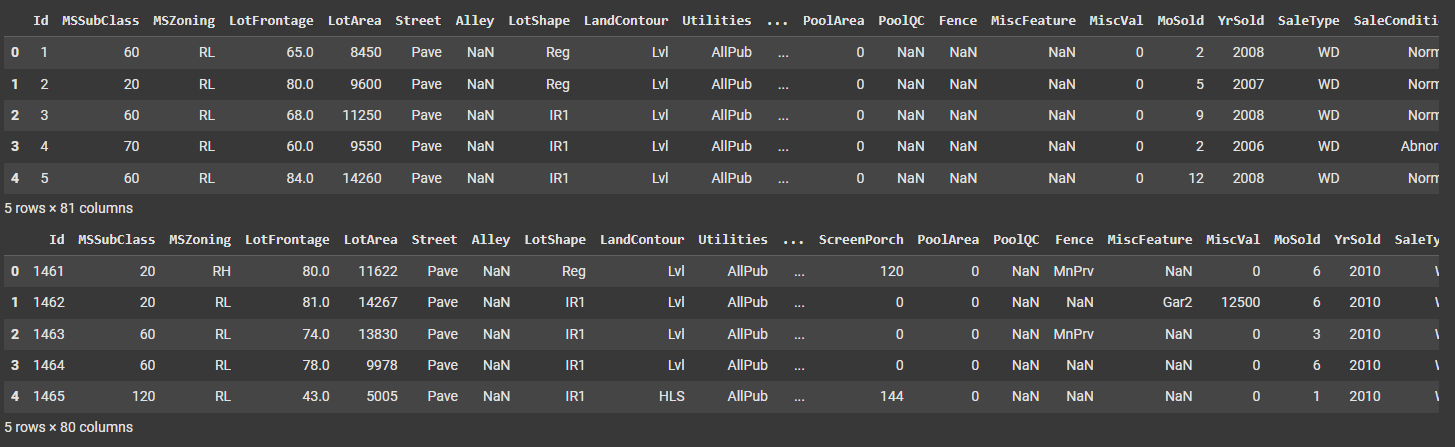


Figure 2.2.1 Viewing data through Head Function

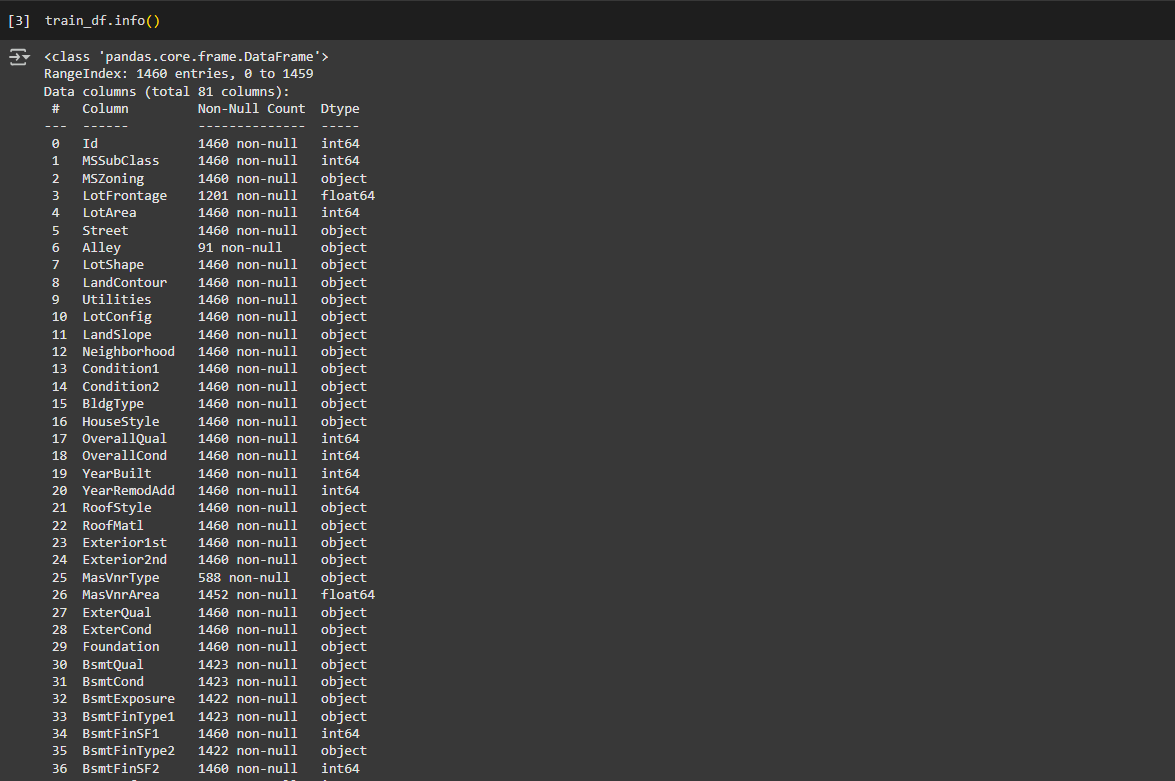


Figure .2.2 Info of Train dataset

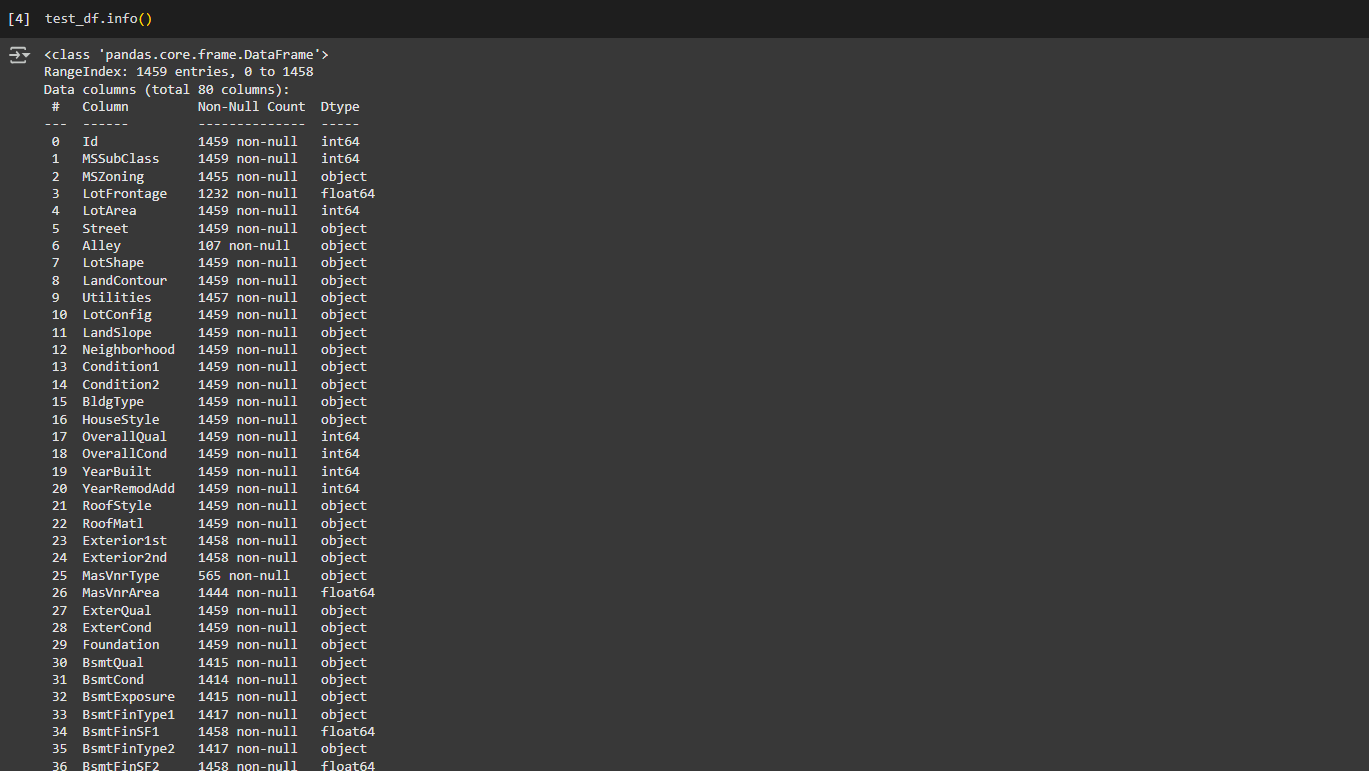


Figure .2.3 Info of Test Dataset

# **Data Preprocessing:**



## Handling Missing Values:

Missing data is handled by:

* **Numerical Columns**: Replacing missing values with the **median** of the respective column.
* **Categorical Columns**: Filling missing values with the **most frequent category** (mode).

## Encoding Categorical Data:

Categorical variables are converted into numerical format using **Label Encoding**, which assigns a unique integer to each category.

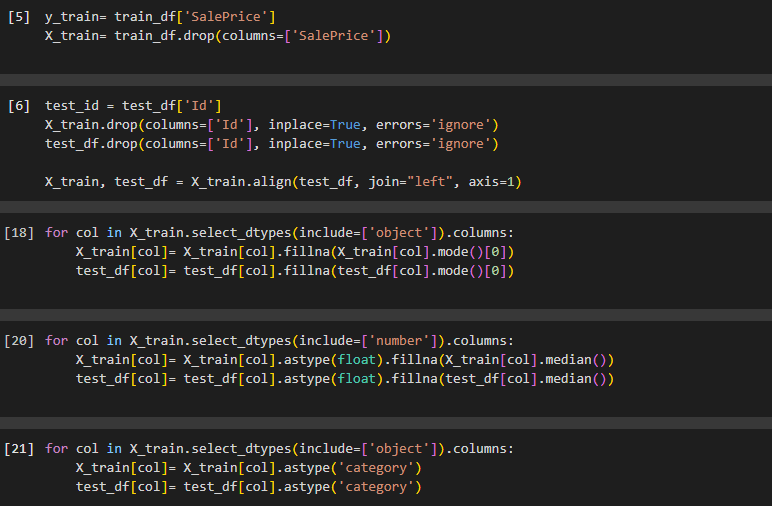


Figure 3.3.1 Preprocessing and Encoding Data

## Splitting Data into Test and Train Sets:

The dataset is split into **training and validation sets** (80%-20%) to evaluate model performance before making final predictions.

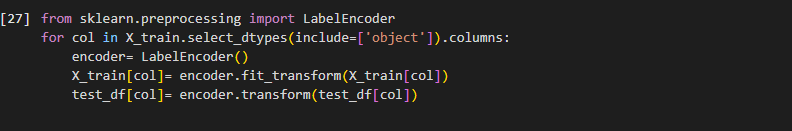


Figure 3.4.1 Splitting Data into Test and Train Sets

# **Model Implementation:**



## XG Boost Classifier:

**Gradient Boosting (XGBoost)**: A boosting method that reduces errors iteratively by adjusting weights.

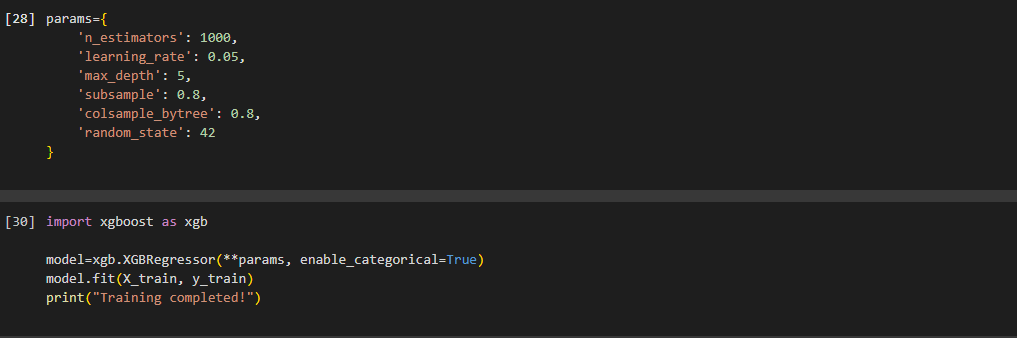


Figure .1 Models Selection and training

# **Saving Sample Submission File:**



The final predictions were stored in a CSV file following the required format:

Id,SalePrice

1461,169000.1

1462,187724.12

1463,175221.0

This file was then submitted to Kaggle for evaluation.

Figure .1.1 Sample Submission File

# **Results:**

The implemented approach successfully predicts house prices with reasonable accuracy while maintaining simplicity and efficiency. Further improvements can be made using ensembling techniques and hyper parameter tuning.

