Helwan University

Advanced Machine Learning Project

Agenda

Tip: Use links to go to a different page inside your presentation.

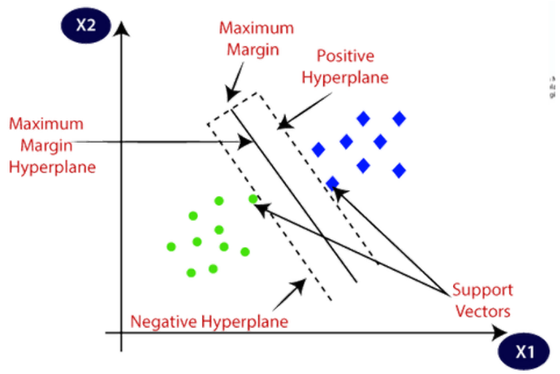
How: Highlight text, click on the link symbol on the toolbar, and select the page in your presentation you want to connect.

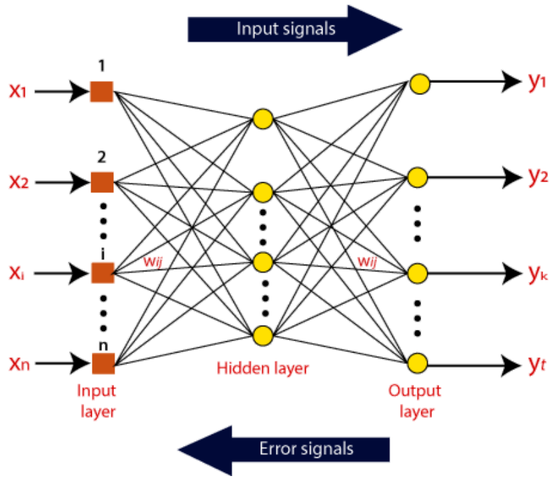
- Introduction
- Datasets
- Regression
  - Data Describtion
  - Data preparation
  - DecisionTree algorithm
  - SVM
  - ANN
- Classification
  - Data Describtion
  - Data preparation
  - DecisionTree algorithm
  - SVM
  - ANN
- Application
- Team

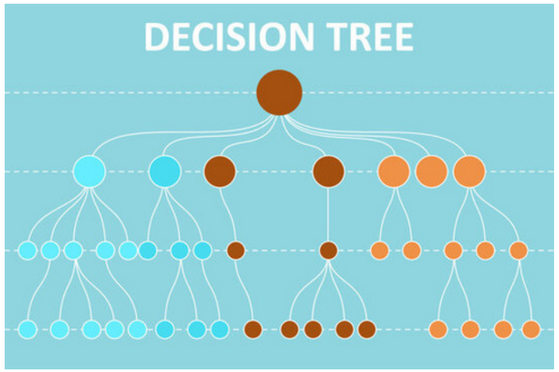
Introduction

what we did?

- In this project, we aimed to explore the capabilities of these models in solving real-world problems by working with two distinct datasets. For the classification task, we utilized the heart attack dataset, aiming to predict the likelihood of heart attacks based on various medical attributes. Meanwhile, for the regression task, we employed the house price prediction dataset, seeking to forecast housing prices based on different features.







DataSets

BACK TO AGENDA PAGE

Regression

House Sales in King County, USA

Classification

Heart Disease Classification Dataset

Regression

House Sales Dataset

Features Description

- **id -** Unique ID for each home sold
- **date -** Date of the home sale
- **price -** Price of each home sold
- **bedrooms -** Number of bedrooms
- **bathrooms -** Number of bathrooms
- **sqft_living -** Square footage of the apartment interior living space
- **sqft_lot -** Square footage of the land space
- **floors -** Number of floors
- **waterfront -** A dummy variable for whether the apartment was overlooking the waterfront or not
- **condition -** An index from 1 to 5 on the condition of the apartment,
- **grade -** An index from 1 to 13, where 1-3 falls short of building construction and design, 7 has an average level of construction and design, and 11-13 have a high quality level of construction and design.

House Sales Dataset

Features Description

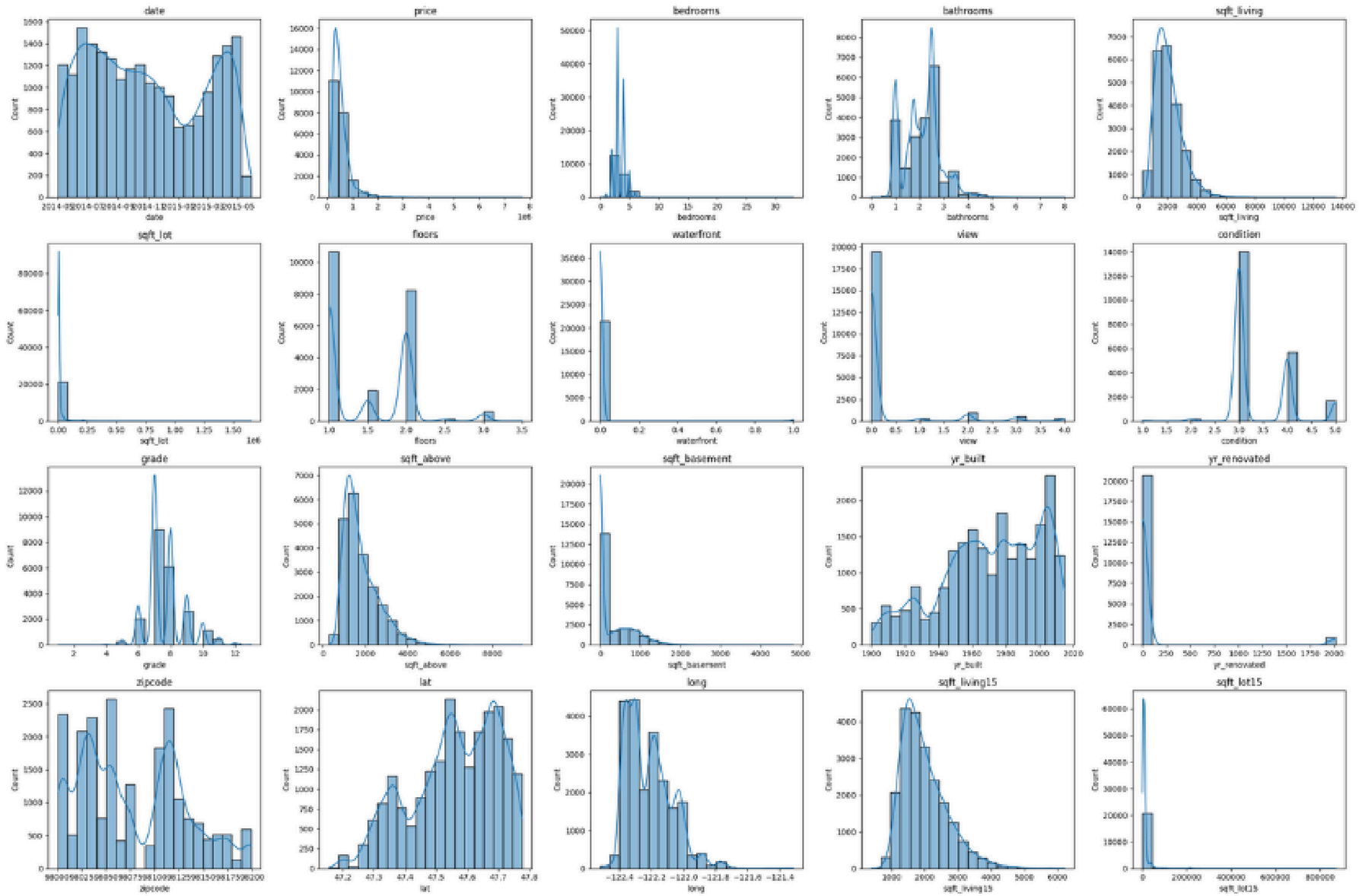
- **sqft_above -** The square footage of the interior housing space that is above ground level
- **sqft_basement -** The square footage of the interior housing space that is below ground level
- **yr_built -** The year the house was initially built
- **yr_renovated -** The year of the house's last renovation
- **zipcode -** What zipcode area the house is in
- **lat -** Lattitude
- **long -** Longitude
- **sqft_living15 -** The square footage of interior housing living space for the nearest 15 neighbors
- **sqft_lot15 -** The square footage of the land lots of the nearest 15 neighbors

House Sales Dataset

Data Before Preprocessing

- Size of data : 2.52 MB
- Number of features : 21 feature
- Number of samples : 21613 sample

Data Distribution (Before)



Data Preprocessing

In [14]:
```
df["yr_renovated"].value_counts().reset_index()
```

Out[14]:

	yr_renovated	count
0	0	20699
1	2014	91
2	2013	37
3	2003	36
4	2005	35
...	...	...
65	1954	1
66	1951	1
67	1959	1
68	1934	1
69	1944	1

70 rows × 2 columns

In [12]:
```
df["waterfront"].value_counts().reset_index()
```

Out[12]:

	waterfront	count
0	0	21450
1	1	163

In [13]:
```
df["view"].value_counts().reset_index()
```

Out[13]:

	view	count
0	0	19489
1	2	963
2	3	510
3	1	332
4	4	319

Most Values of these features (yr_renovated, view, waterfront) are zero ,so we gonna drop them

House Sales Dataset

Standard Scaler

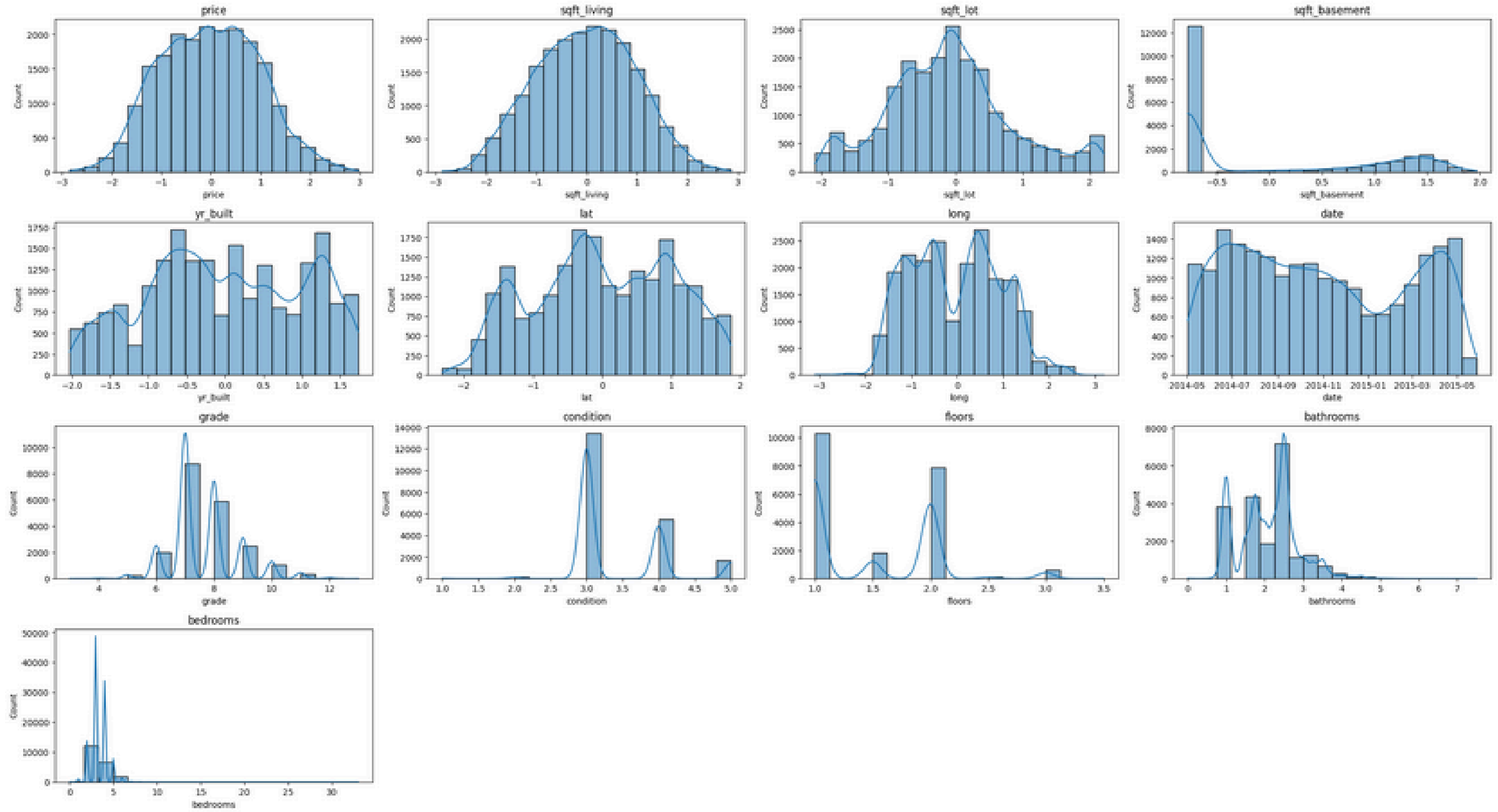
We scaling data with Standard Scaler To all column and ignore this column:
"grade", "condition", "floors", "bathrooms", "bedrooms", “date”
Because values in this column is categorical

House Sales Dataset

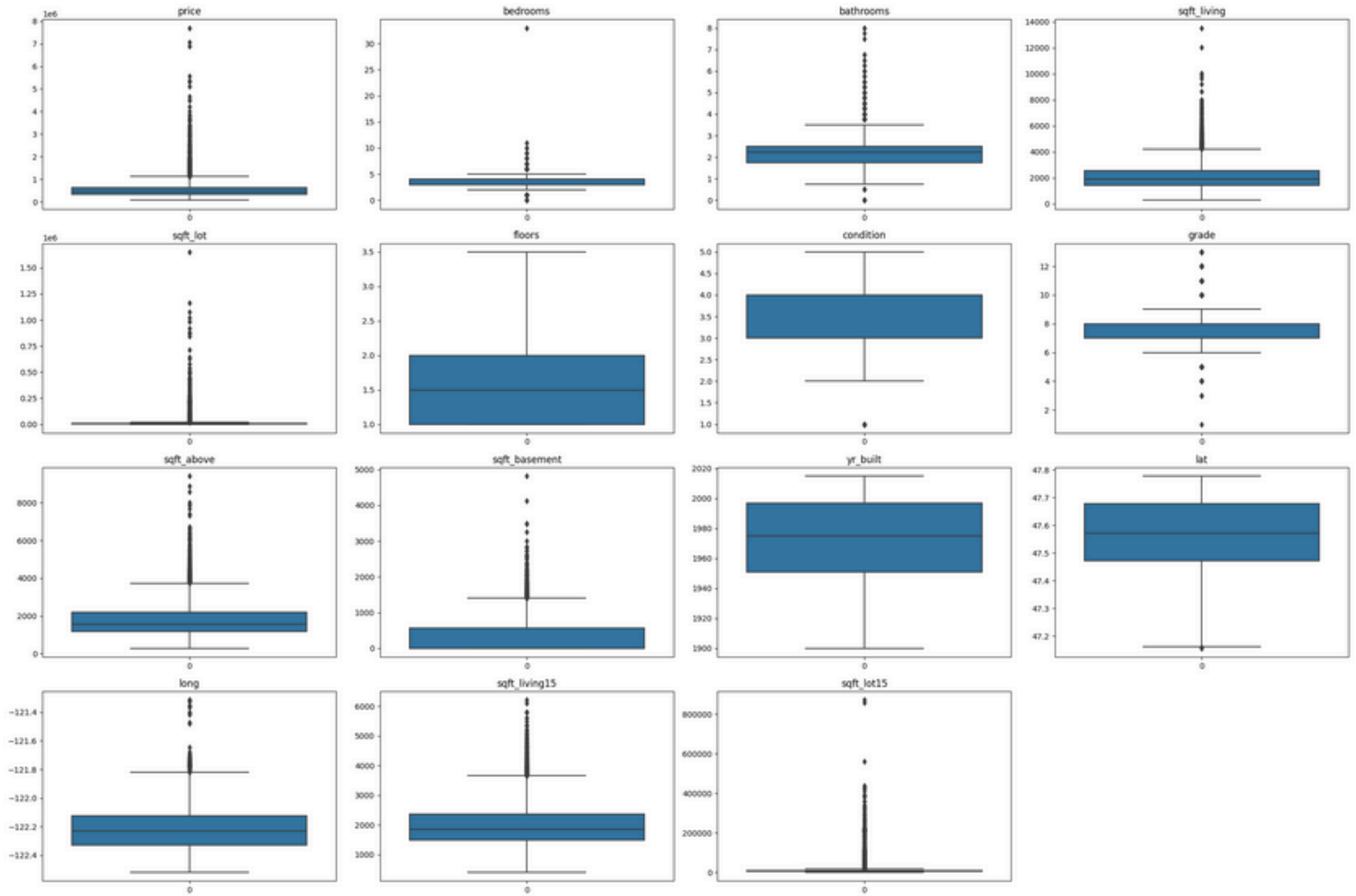
Transform

transform data with power transform (method = yeo-johnson) to make distribution of data normalized

Data Distribution (After)



Outliers (Before)

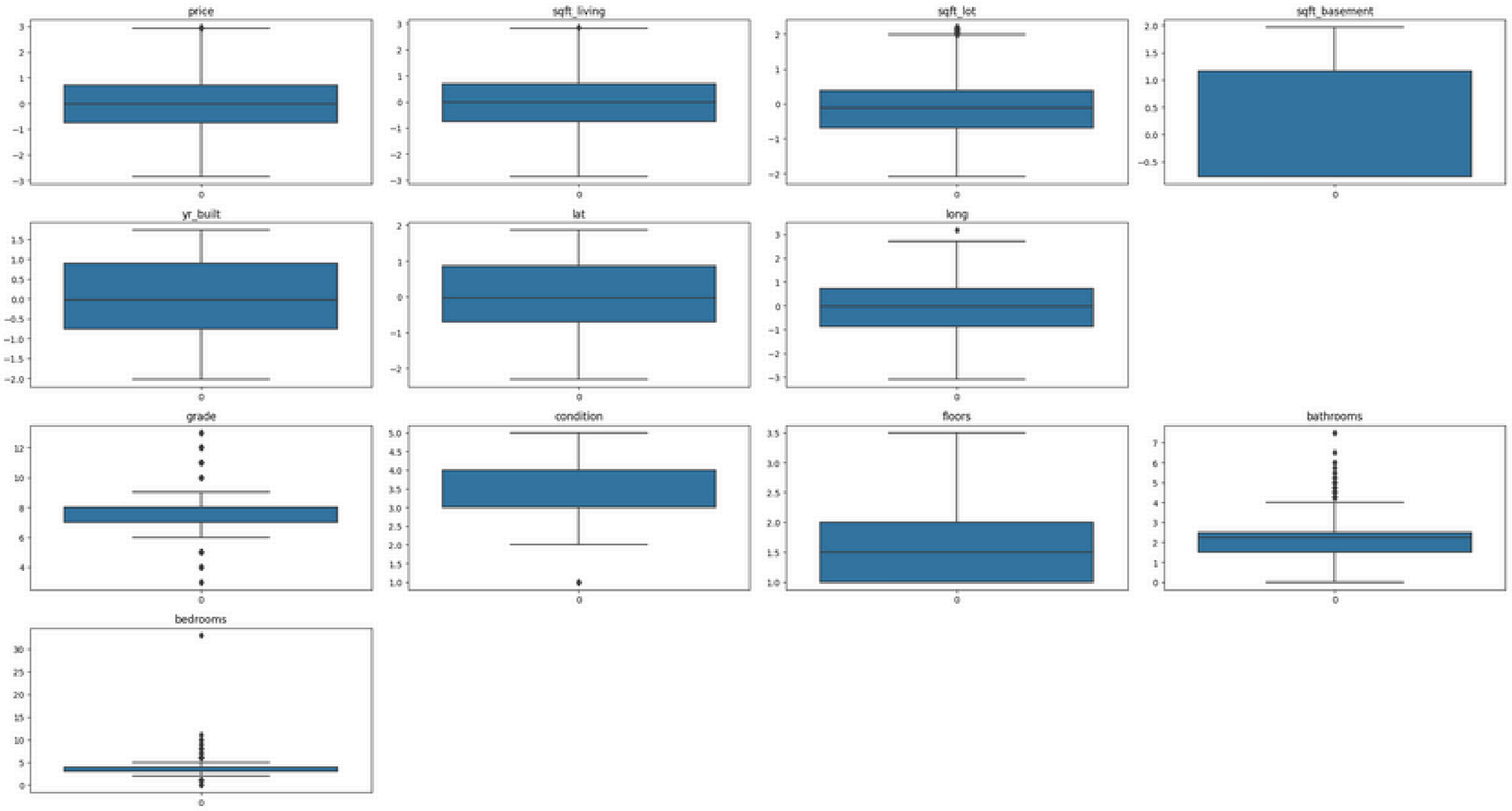


House Sales Dataset

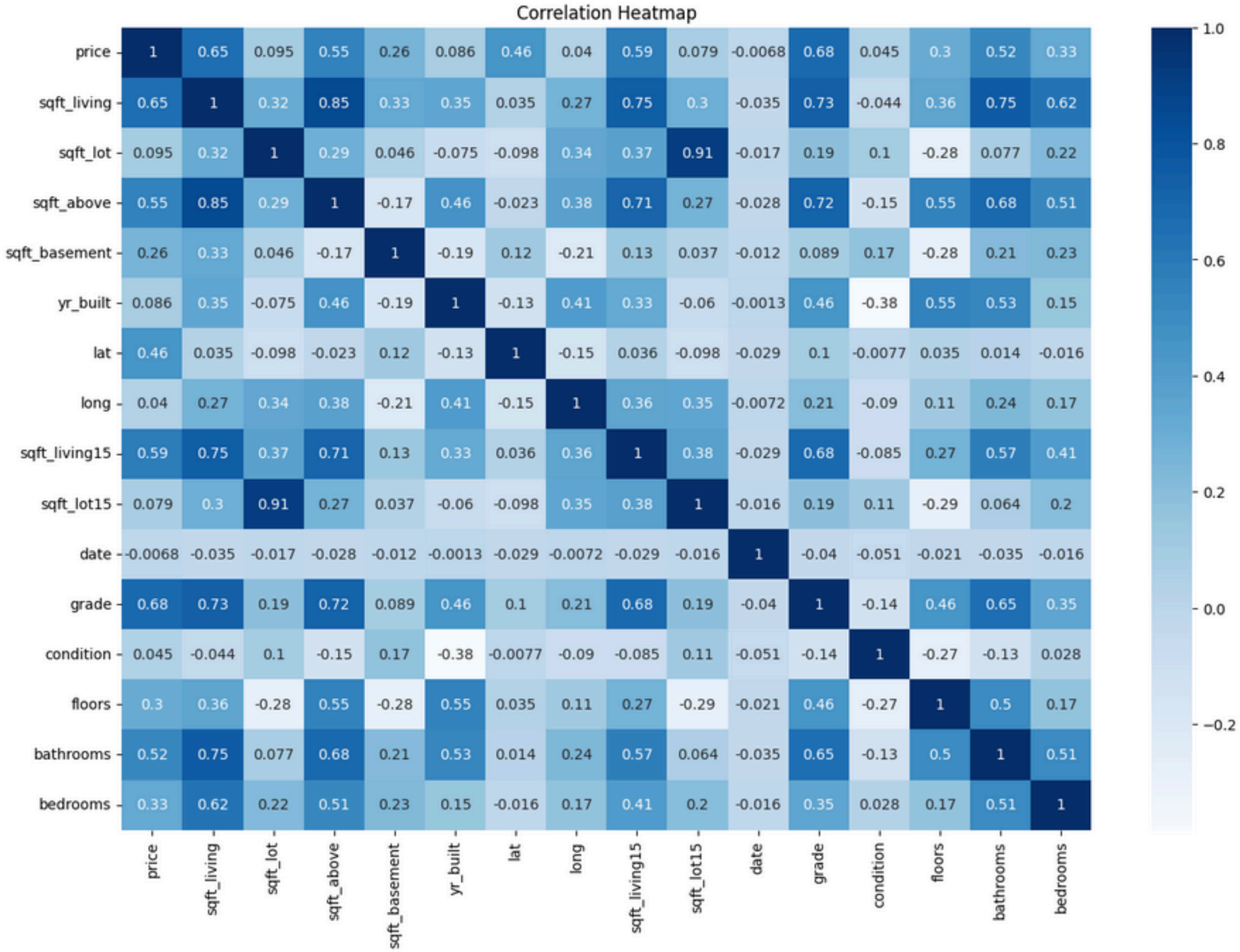
Handle Outliers

We Are Using IQR Technique To Handle Outliers
And Remove Outliers

Outliers (Before)



correlation (Before)

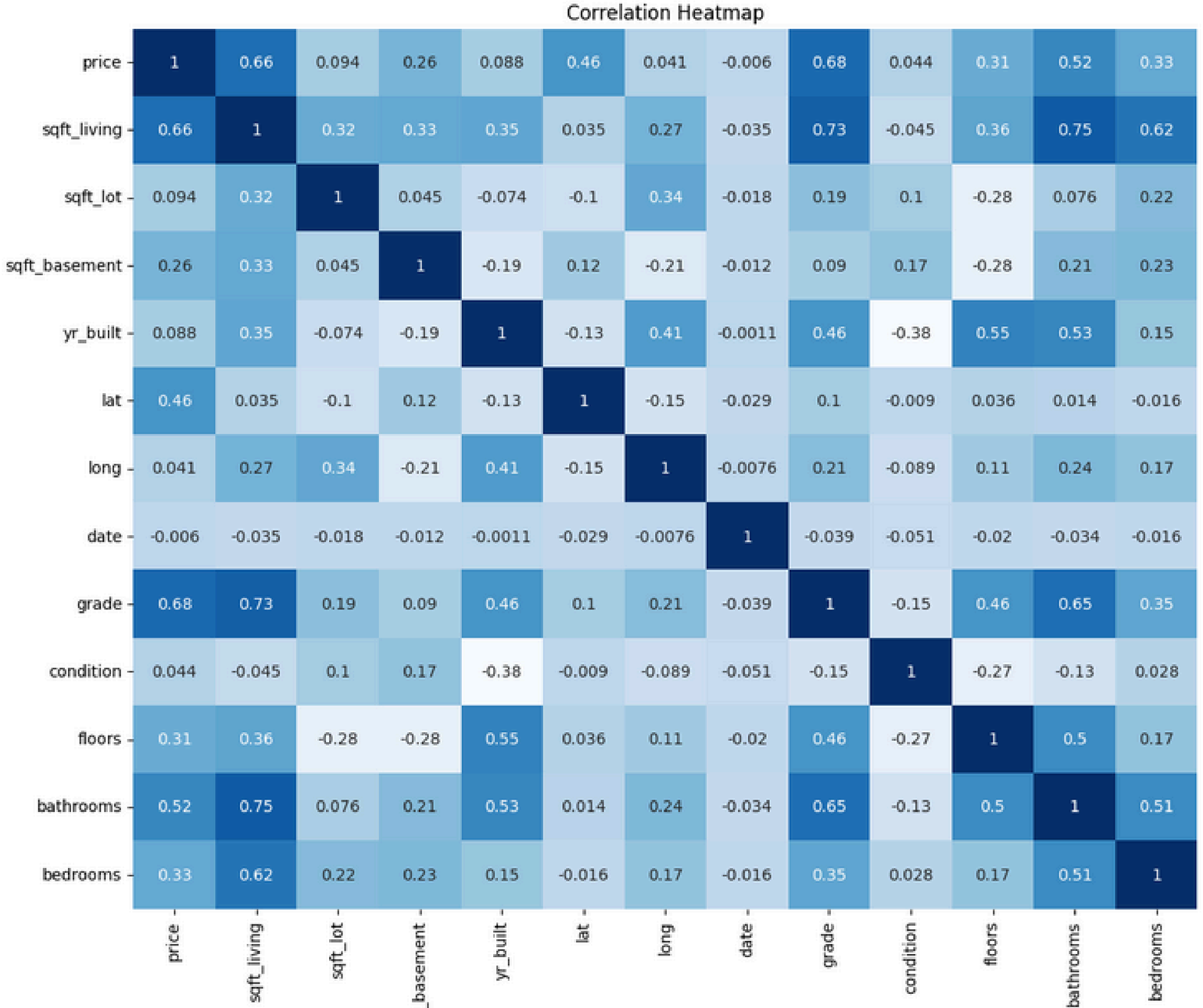


House Sales Dataset

Multicollinearity

- we droped these columns 'sqft_above','sqft_living15','grade','bathrooms'
- Due to high corellation with each other

correlation (After)



House Sales Dataset

Data After Preprocessing

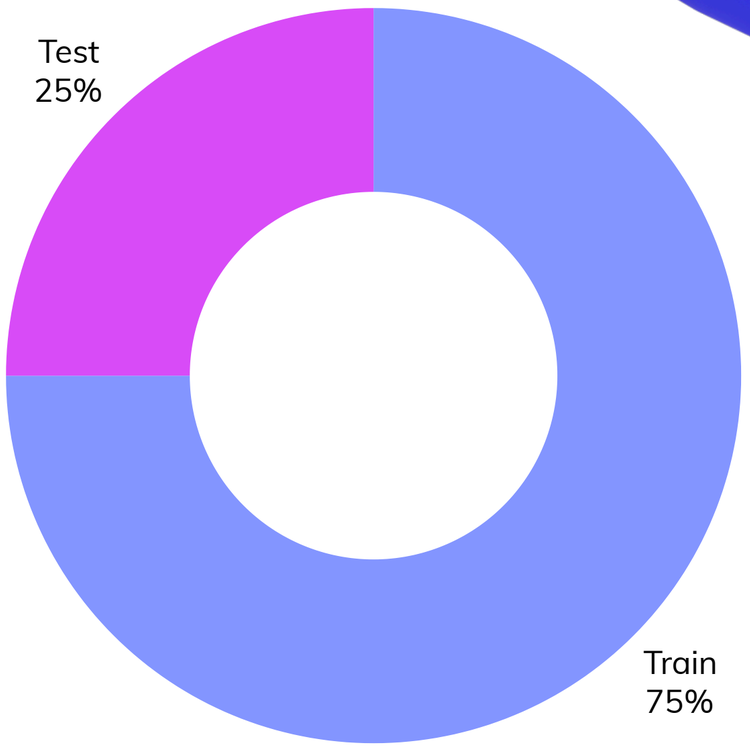
Features : 11

Samples : 20698

House Sales Dataset

Split Data

we split data using 75% for train and 25% for test



House Sales Model

DecisionTree

- Train R2_score : 88.25
- Test R2_score : 82.93
- Train mean_squared_error : 111.59
- Test mean_squared_error : 16.9

House Sales Model

DecisionTree

Cross validation score cv = 15

```
array([0.82339137, 0.84145324, 0.83847831, 0.81951225, 0.83881084,
       0.82822249, 0.80122555, 0.83946381, 0.83421607, 0.84117978,
       0.83576592, 0.84415746, 0.83293164, 0.85319055, 0.82760834])
```

- the mean of cross_val_score is 0.833307174365589 5

House Sales Model

SVM

- Train R2_score : 85.42
- Test R2_score : 84.77
- Train mean_squared_error : 14.38
- Test mean_squared_error : 15.08

House Sales Model

SVM

Cross validation score cv = 3

```
array([0.84456054, 0.84268922, 0.85224156])
```

- the mean of cross_val_score is 0.8464971035615894

House Sales Model

we use 40 epochs ,
batch size is 16 ,
validation_split 20% and
optimizer adm

ANN Model structure

Model: "sequential"

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 64)	768
dense_1 (Dense)	(None, 64)	4,160
dense_2 (Dense)	(None, 1)	65

Total params: 4,993 (19.50 KB)

Trainable params: 4,993 (19.50 KB)

Non-trainable params: 0 (0.00 B)

House Sales Model

we use Early Stop in val_loss with 5 patience

ANN Model Fit

```
Epoch 1/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - loss: 0.4258 - val_loss: 0.1632
Epoch 2/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.1810 - val_loss: 0.1540
Epoch 3/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 2s 2ms/step - loss: 0.1468 - val_loss: 0.1437
Epoch 4/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - loss: 0.1490 - val_loss: 0.1421
Epoch 5/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 2s 2ms/step - loss: 0.1349 - val_loss: 0.1404
Epoch 6/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.1329 - val_loss: 0.1365
Epoch 7/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 2s 2ms/step - loss: 0.1320 - val_loss: 0.1451
Epoch 8/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.1342 - val_loss: 0.1425
Epoch 9/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - loss: 0.1273 - val_loss: 0.1281
Epoch 10/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.1253 - val_loss: 0.1363
Epoch 11/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - loss: 0.1270 - val_loss: 0.1272
Epoch 12/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - loss: 0.1208 - val_loss: 0.1299
Epoch 13/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - loss: 0.1224 - val_loss: 0.1336
Epoch 14/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - loss: 0.1202 - val_loss: 0.1387
Epoch 15/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 2s 2ms/step - loss: 0.1203 - val_loss: 0.1294
Epoch 16/40
780/780 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - loss: 0.1186 - val_loss: 0.1312
```

House Sales Model

ANN

- Train R2_score : 87.73
- Test R2_score : 86.97
- Train mean_squared_error : 12.11
- Test mean_squared_error : 12.9

House Sales Model

DecisionTree VS SVM VS ANN

	Model	R2 Train	R2 Test	mean_squared_error score for train	mean_squared_error score for test
0	SVM	85.424628	84.772474	14.389875	15.086894
1	Decision Tree	88.253577	82.934721	11.596929	16.907675
2	ANN	87.733390	86.975763	12.110496	12.903953

Classification

Heart Disease Classification Dataset

Dataset Description

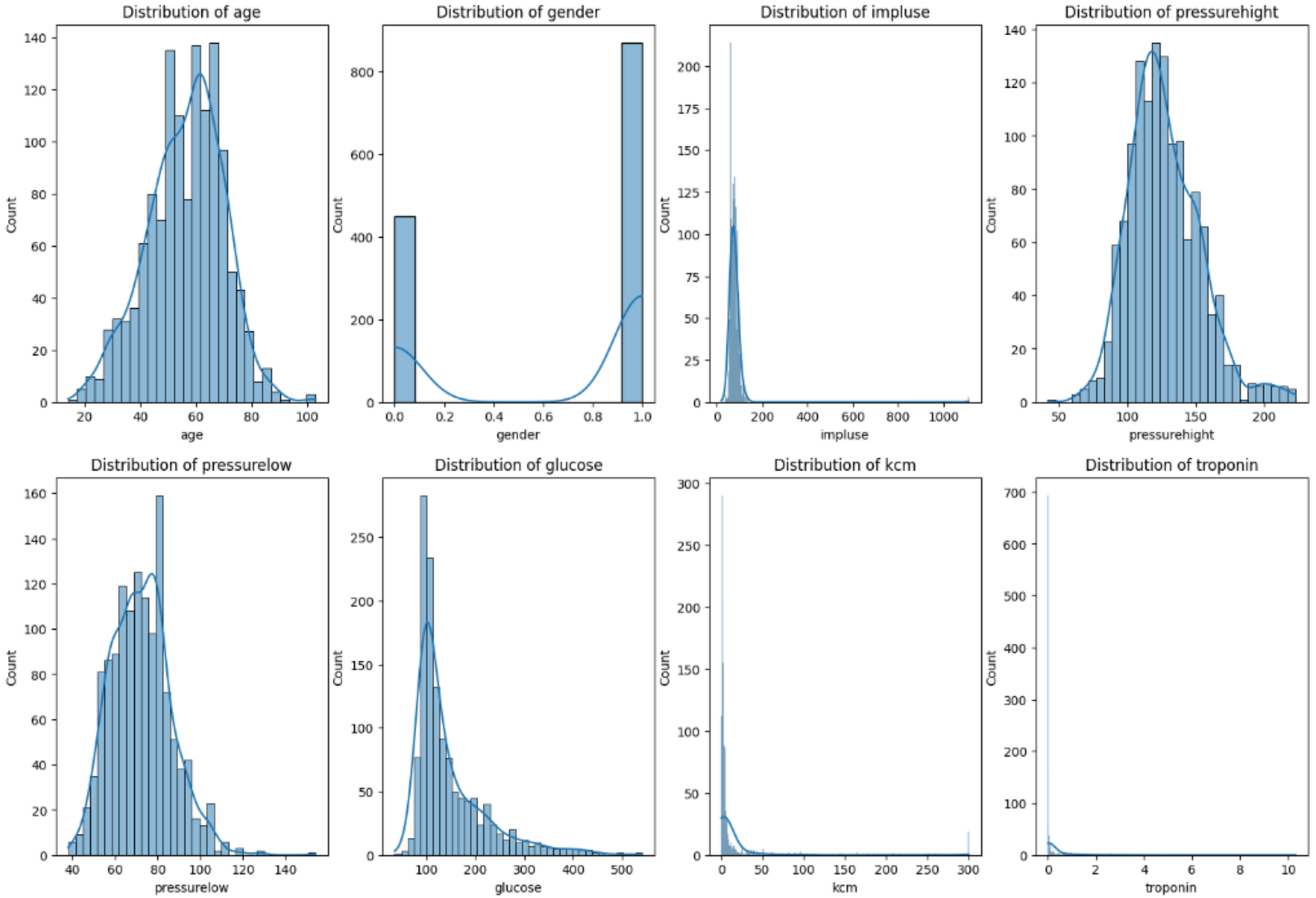
- The size of the dataset is **1319 samples**, which have nine fields, where eight fields are for input fields and one field for an output field. **Age**, **gender(0 for Female, 1 for Male)** , **heart rate (impulse)**, **systolic BP (pressure-hight)**, **diastolic BP (pressure-low)**, **blood sugar(glucose)**, **CK-MB (kcm)**, and **Test-Troponin (troponin)** are representing the input fields, while the output field pertains to the presence of **heart attack** (class), which is divided into two categories (negative and positive); negative refers to the absence of a heart attack, while positive refers to the presence of a heart attack

Data PreProcessing

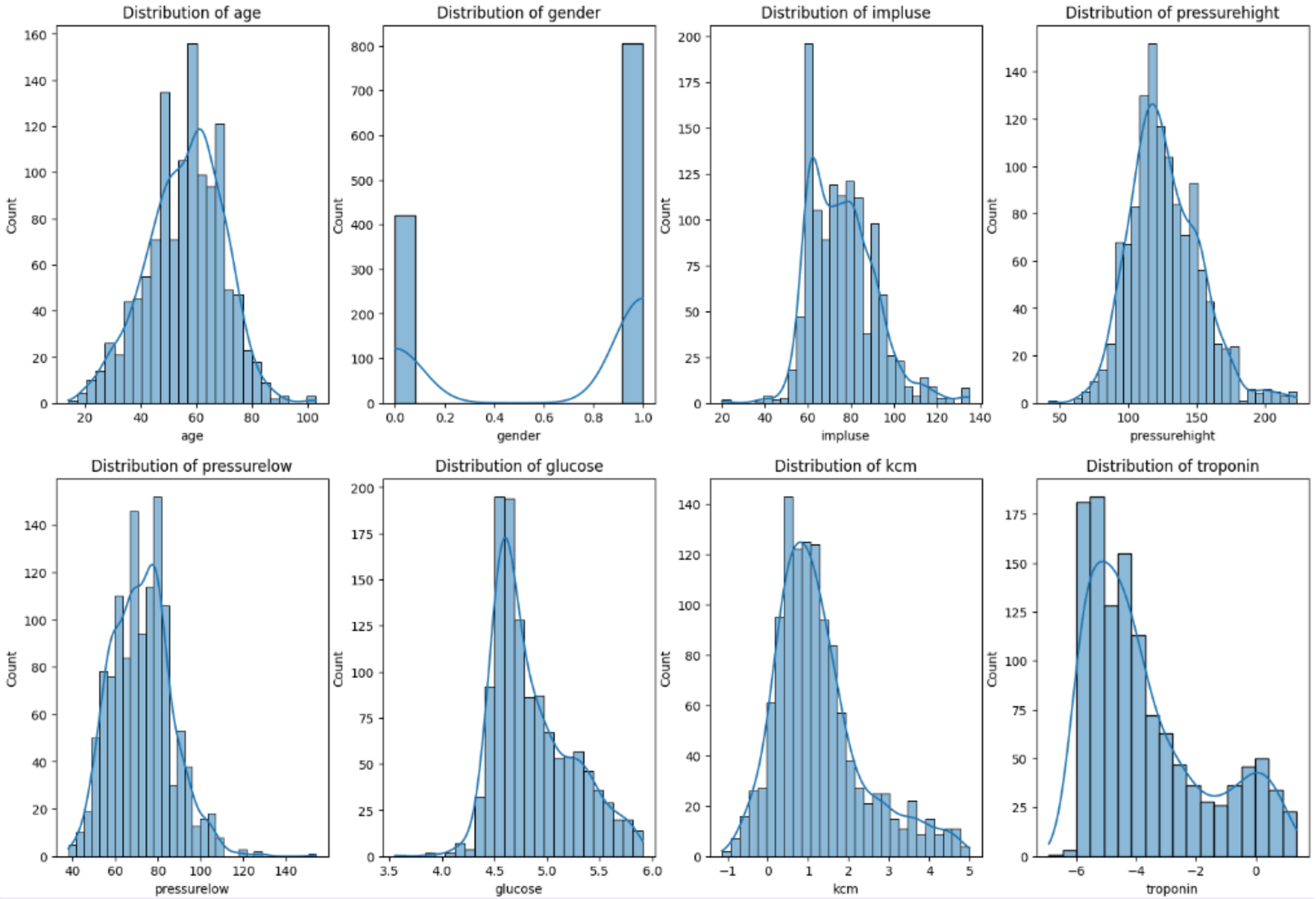
we worked on data pre-processing by focusing on:

- **Data Skew (log transform)**
- **removing outliers**
- **make sure there are not any null values**
- **Solve implanted classes problem,**

Data Distribution Example (Before)

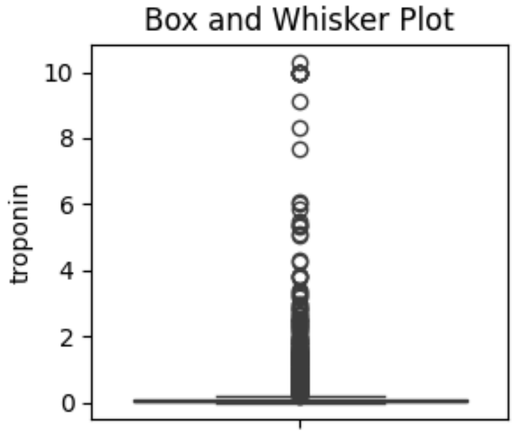


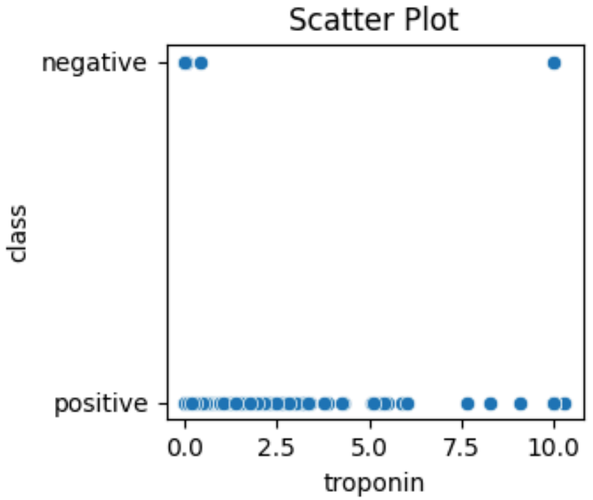
Data Distribution Example (After)

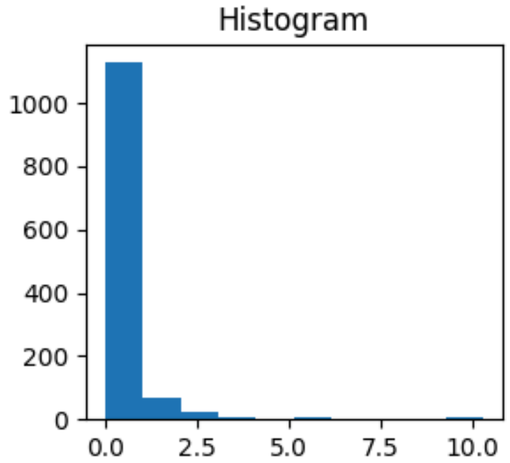


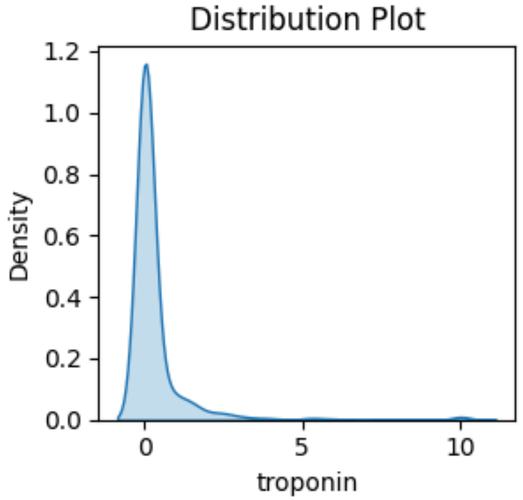
The process on 1 feature

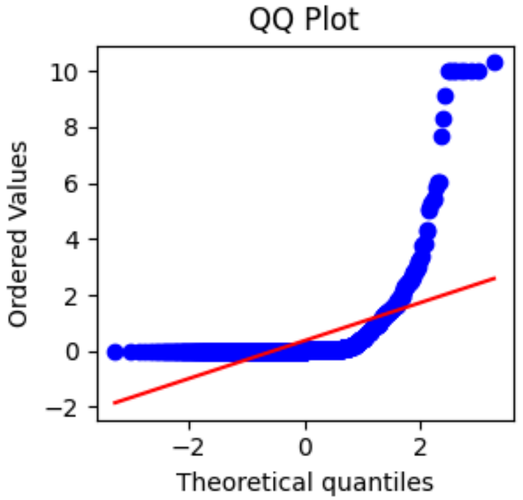
Normal Column ”troponin”





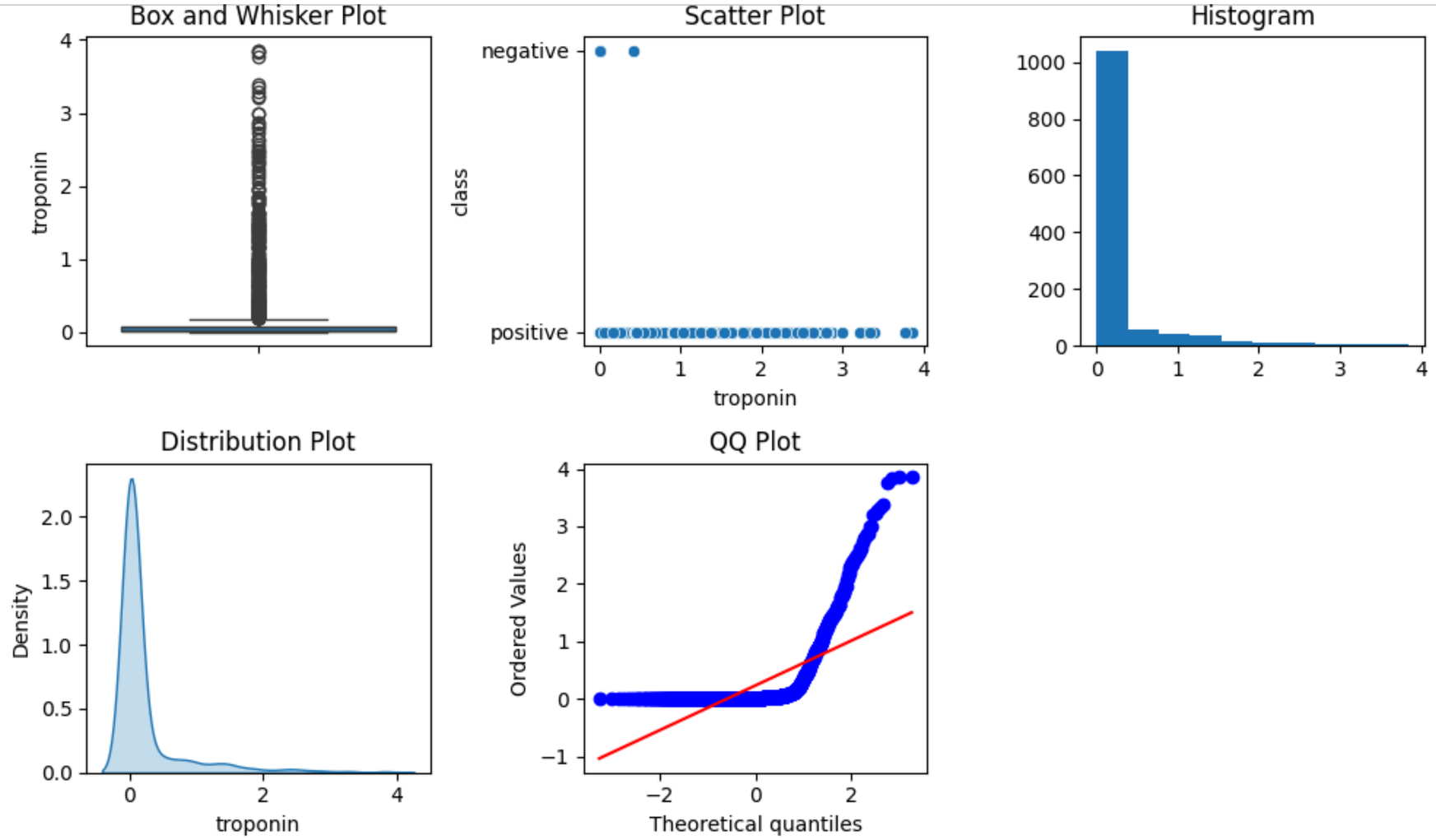






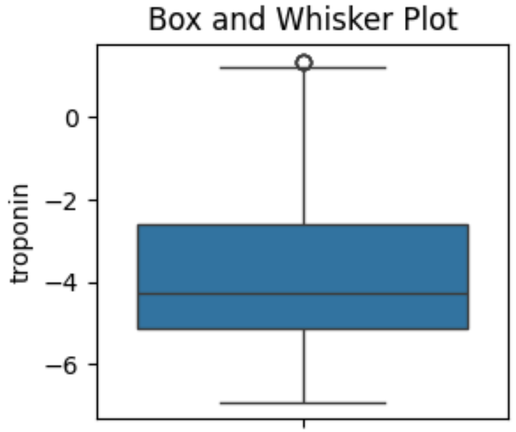
The process on 1 feature

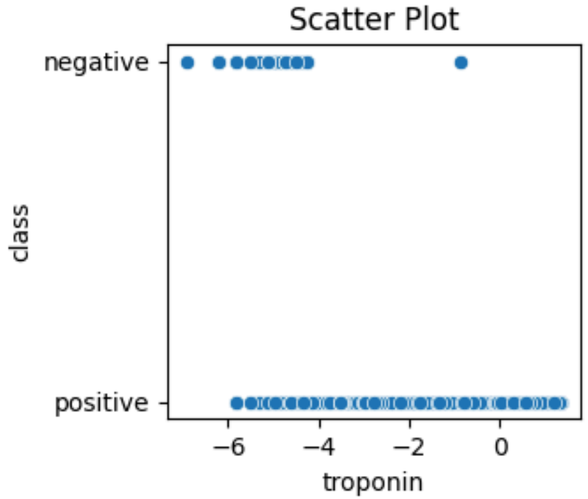
1. removing outliers

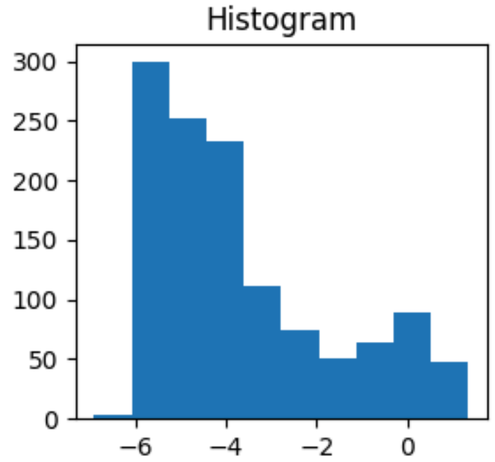


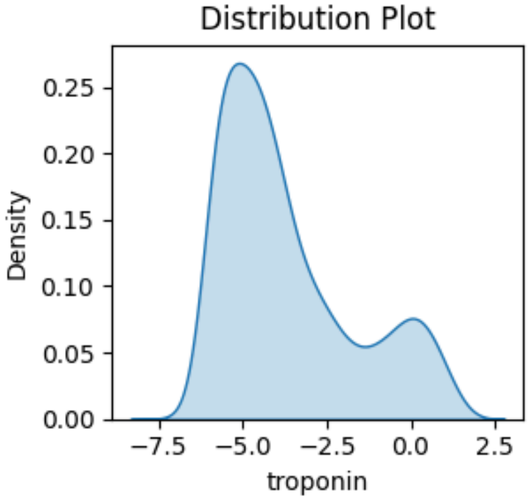
The process on 1 feature

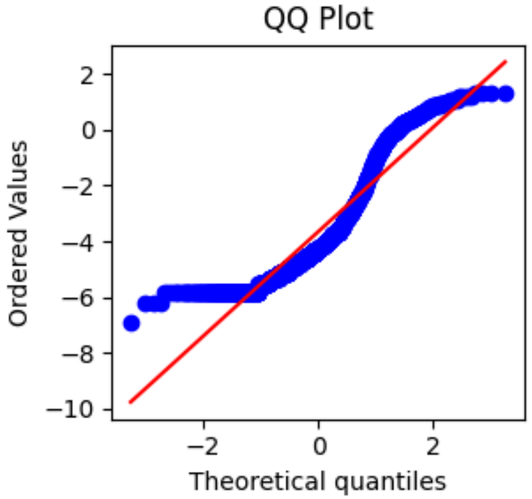
3. Log transformation





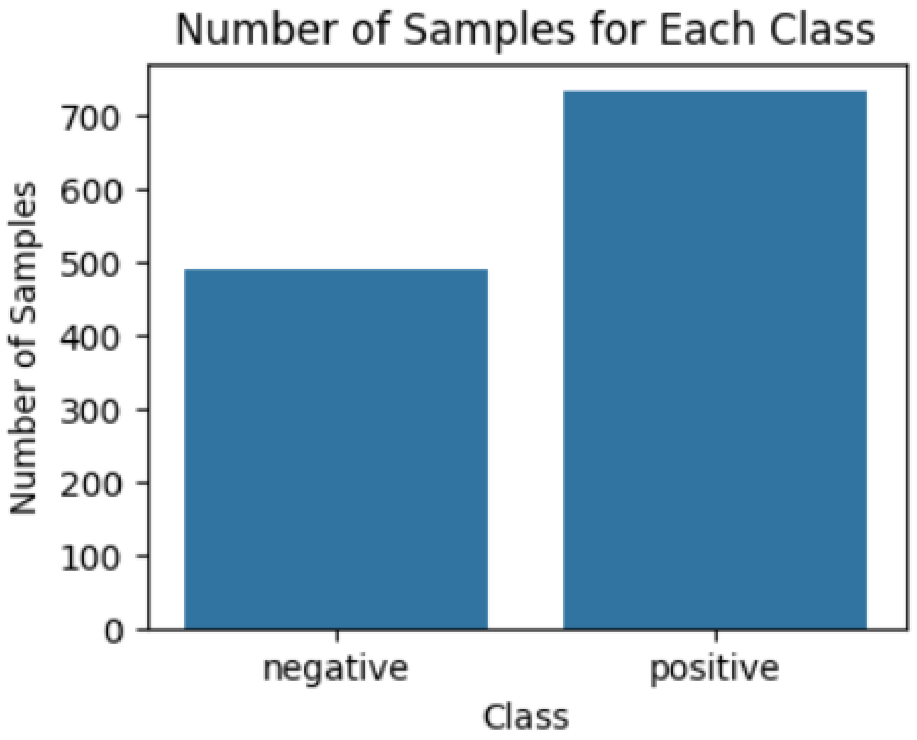


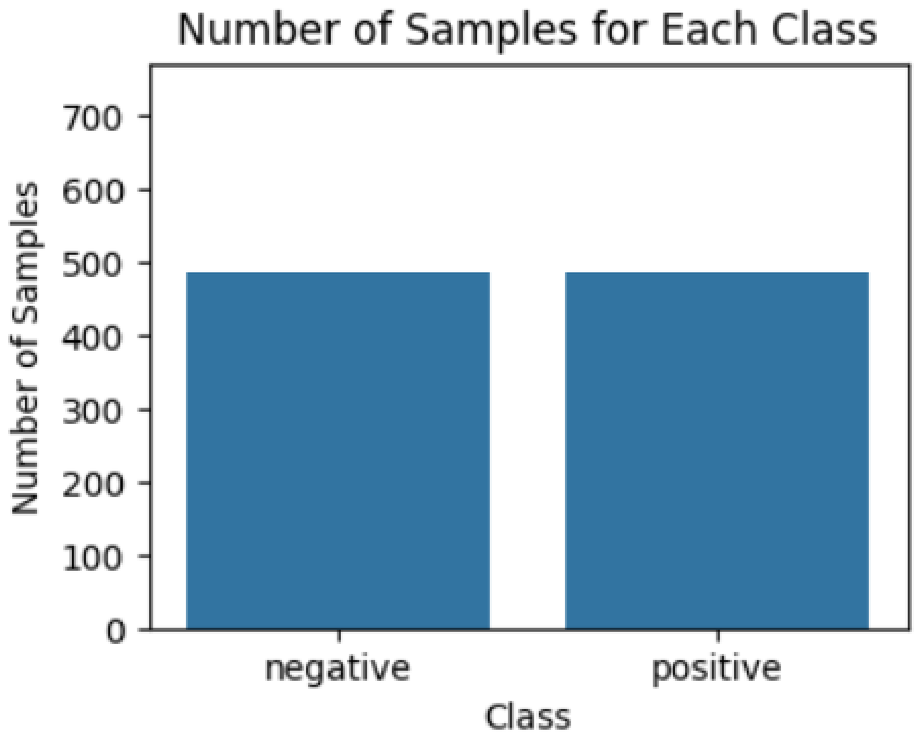




Solve implanced classes

we did it by do Down sampling

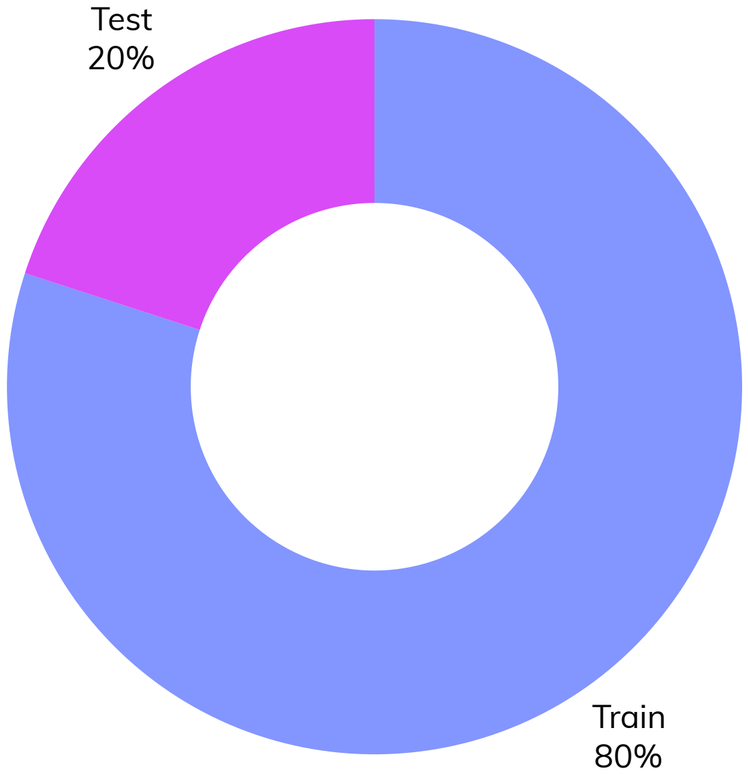




Heart Disease Classification Dataset

Split Data

we split data using **80%** for training and **20%** for testing



Model Train

Support Vector Machines (SVC)

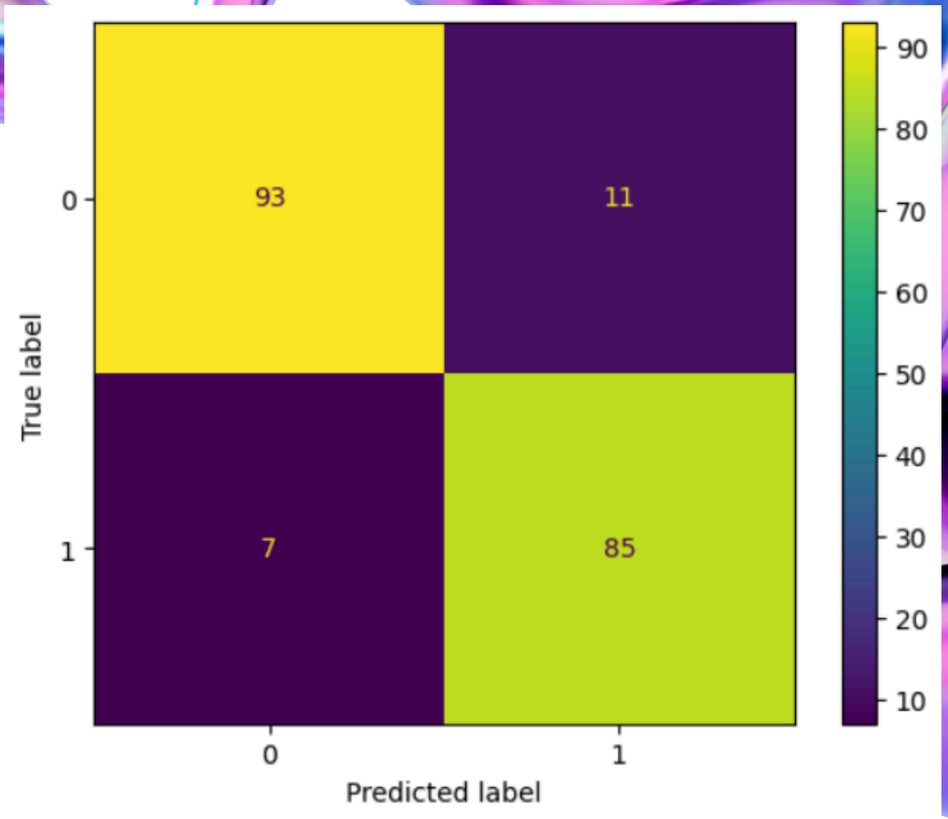
- **Accuracy score: 90.81%**
- **Evaluation**

```
param_grid = {
    'C': [9,10,12,13],
    'gamma': ['scale', 0.1, 0.2, 0.3, 0.01],
}

# Create the SVM model
svm_2 = SVC(kernel='rbf')

# Perform Grid Search Cross-Validation to find the best parameters
grid_search = GridSearchCV(svm_2, param_grid, cv=10)
grid_search.fit(X_train_scaled, y_train)
```

```
Best parameters:  {'C': 10, 'gamma': 0.1}
Test set score with best parameters:  0.9030612244897959
```



Model Train

Decision Tree Classifier (DT)

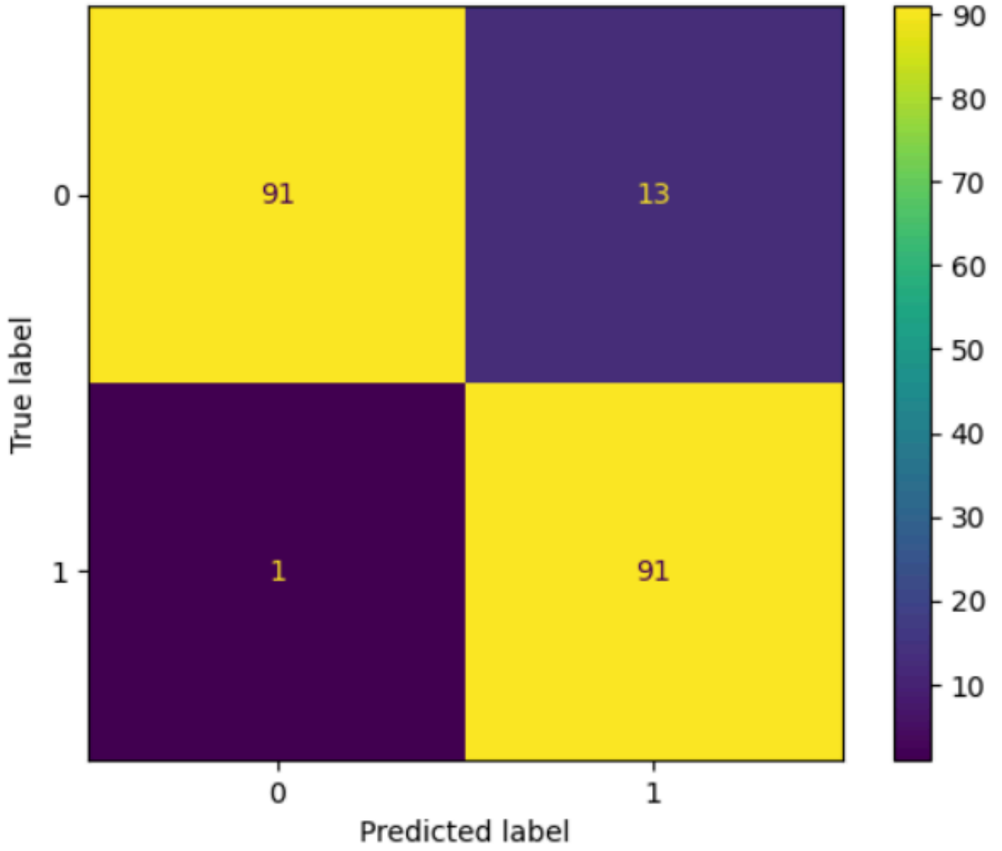
- **Accuracy score: 92.85%**
- **Evaluation**

```
param_grid = {
    'max_depth': [None, 10, 20, 30],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}

# Initialize Decision Tree Classifier
DT_2 = DecisionTreeClassifier(random_state=42)

# Initialize GridSearchCV
grid_search = GridSearchCV(DT_2, param_grid, cv=5, scoring='accuracy')

# Perform Grid Search to find the best hyperparameters
grid_search.fit(X_train_scaled, y_train)
```

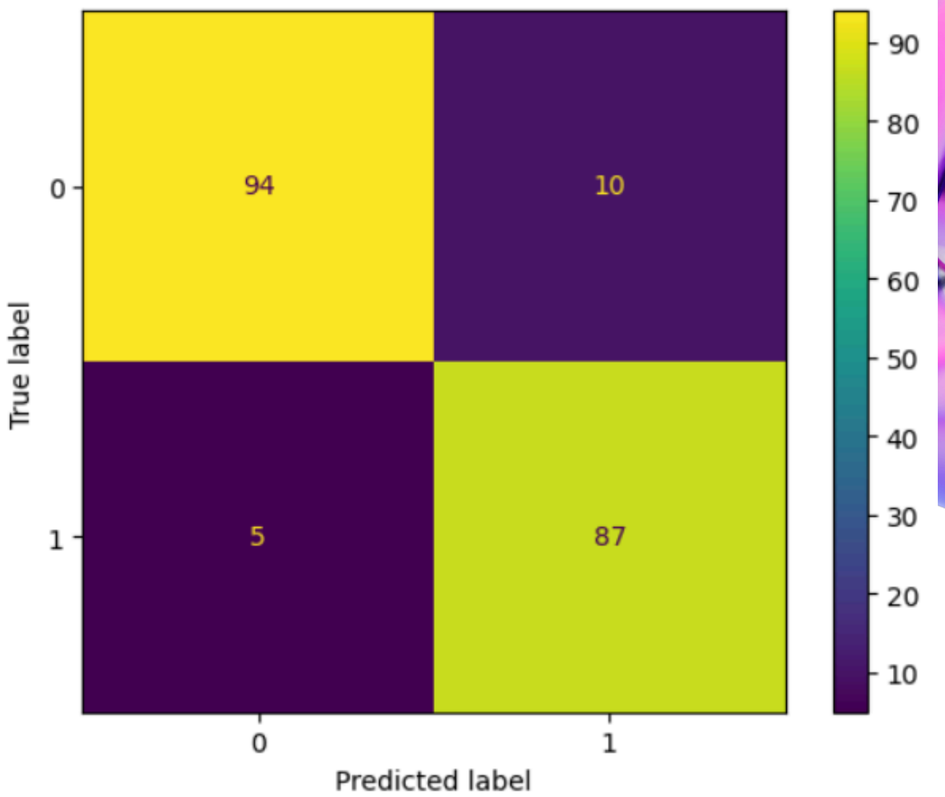



```
Best Hyperparameters: {'max_depth': None, 'min_samples_leaf': 2, 'min_samples_split': 5}
```

Model Train

Artificial Neural Networks (ANN)

- **Accuracy score: 92.34%**
- **Evaluation**

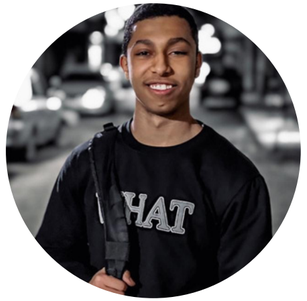


The Team

Ali Adel

Abdelhalem Ashraf

Eid Osama



Abdelrahman

Mohey Eldeen

Amr Khaled

BACK TO AGENDA PAGE