Horse Health Prediction

Machine Learning Project

Group 12
Dhanyamraju Harsh Rao
Malavade Sanskar Deepak
Parashar Kshitij
Chen Kian Leong
Mitra Ren Sachithananthan

Table of contents

Kaggle Competition

- Purpose: Predict the Health Outcome of a Horse from data about its health
- Tabular Data
- Three Class Classification Task
- Outcome can be lived, died, or euthanized
- Test Metric: Micro Averaged F1 Score
- Competition Link: https://www.kaggle.com/competitions/playground-series-s3e22/

Dataset

Training Set

Rows: 1235
Columns: 27 + target

Numerical Columns: 12
Categorical Columns: 15

Test Set

Rows: 824
Columns: 27

External Data

Data used to create the synthetic data for the competition

Rows: 299
Columns: 26 + target

Challenges Faced

Small Training Set

1. Less Data to train
2. Overfitting
3. Less Data for Validation

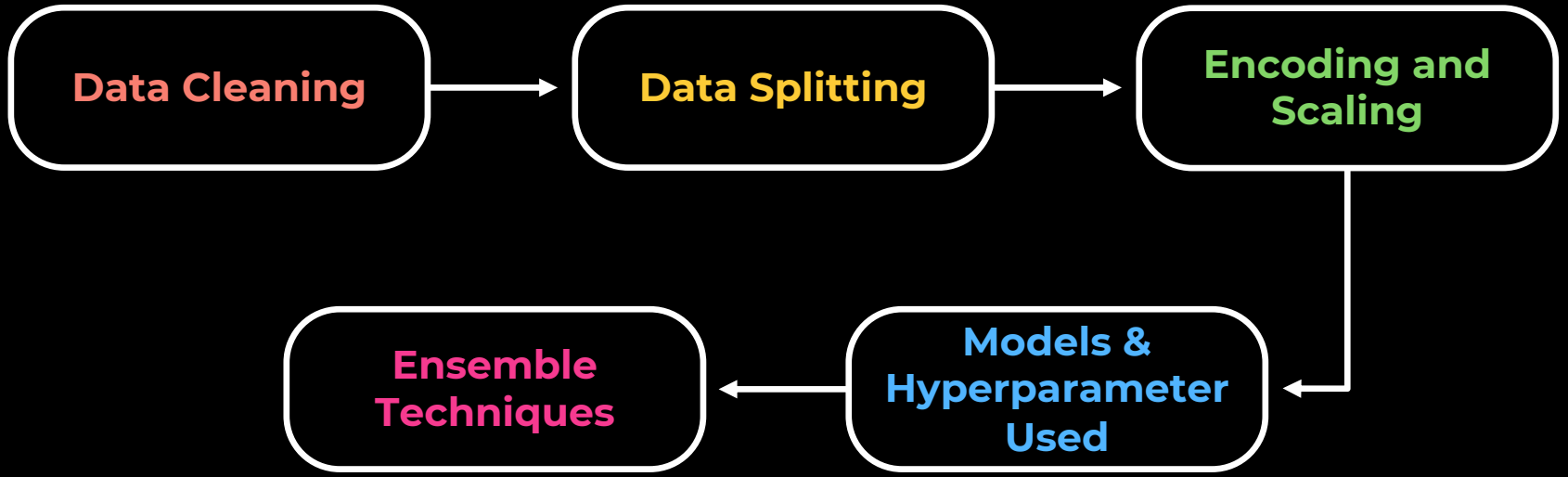
Large number of NaN values

1. Remove all NaN => 771 rows (**-38 %**)
2. Spread across both train and test

Small Test Set

1. Public Test Set is 20 % of test data (i.e, 164 rows)
2. Narrow Margins for Test F1

Pipeline


Data Cleaning
Data Splitting
Encoding and Scaling
Models & Hyperparameter Used
Ensemble Techniques

Pipeline - Data Cleaning

Drop Columns

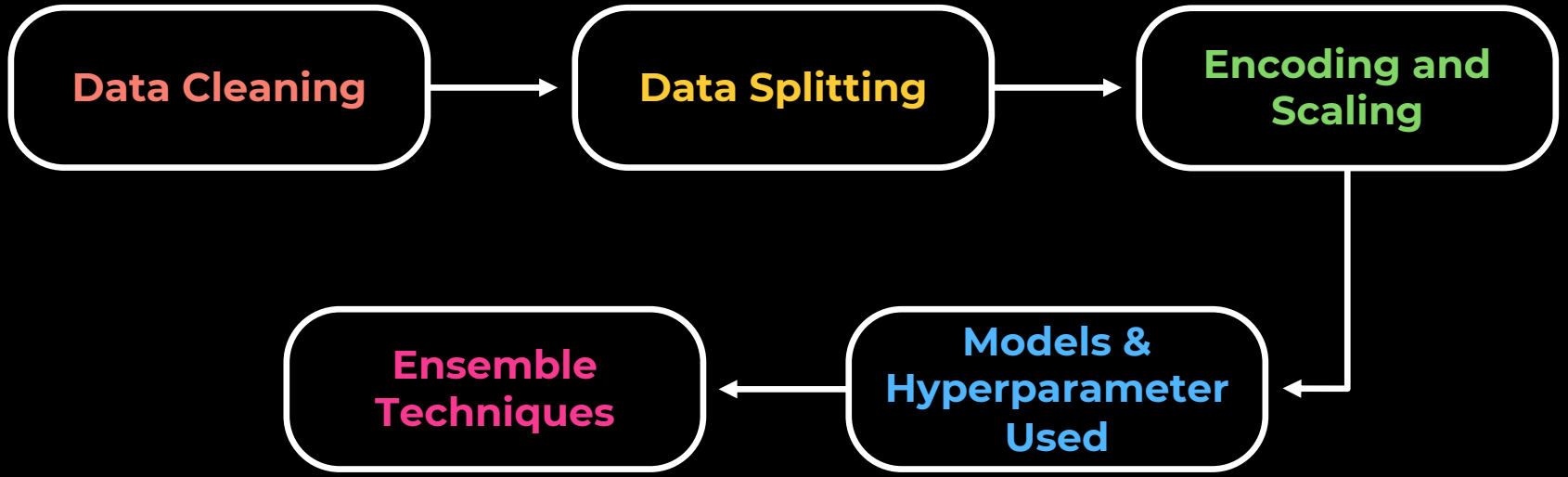
1) Hospital Number
2) Id
3) Lesion 3

NaN Values

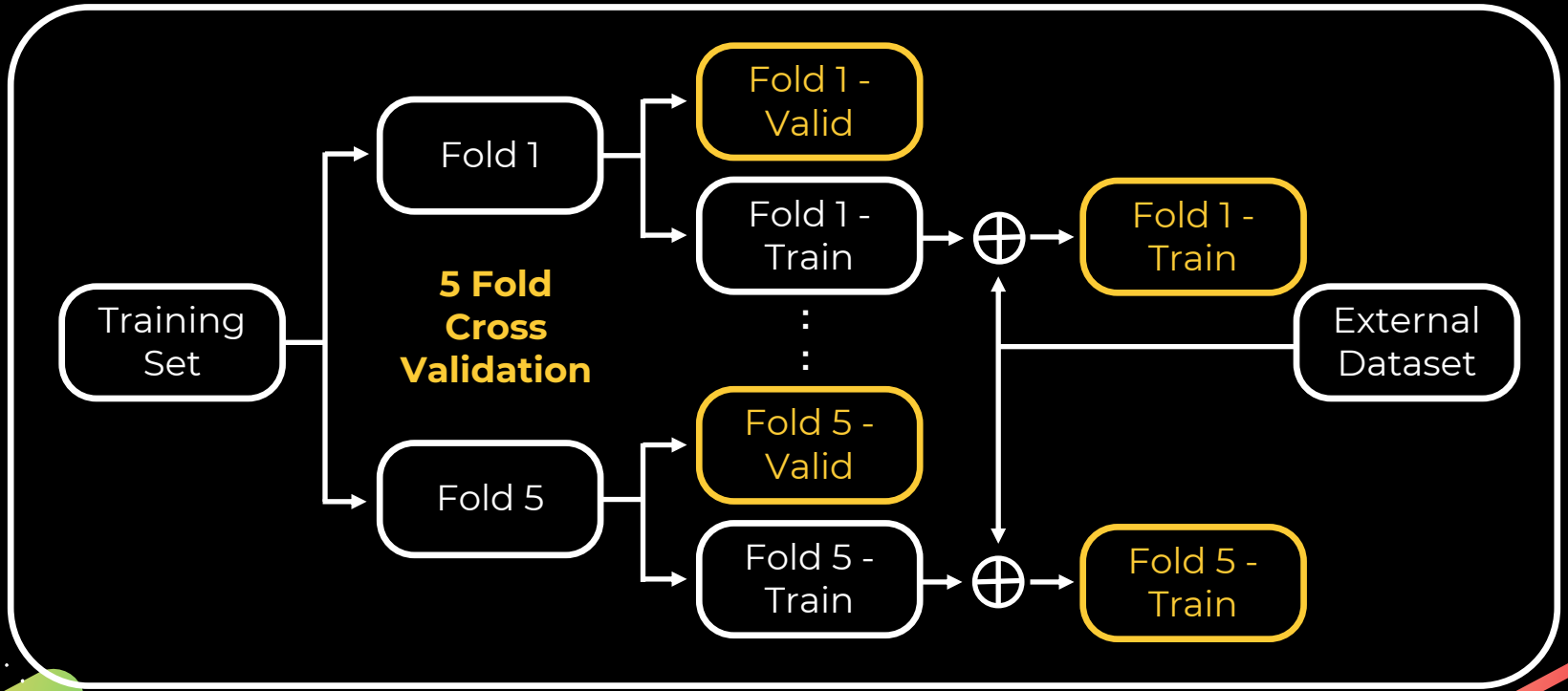
1)Replace with mode if categorical
2) Replace with median if numerical*

* There were no NaN values in numerical columns in Train dataset only in External dataset

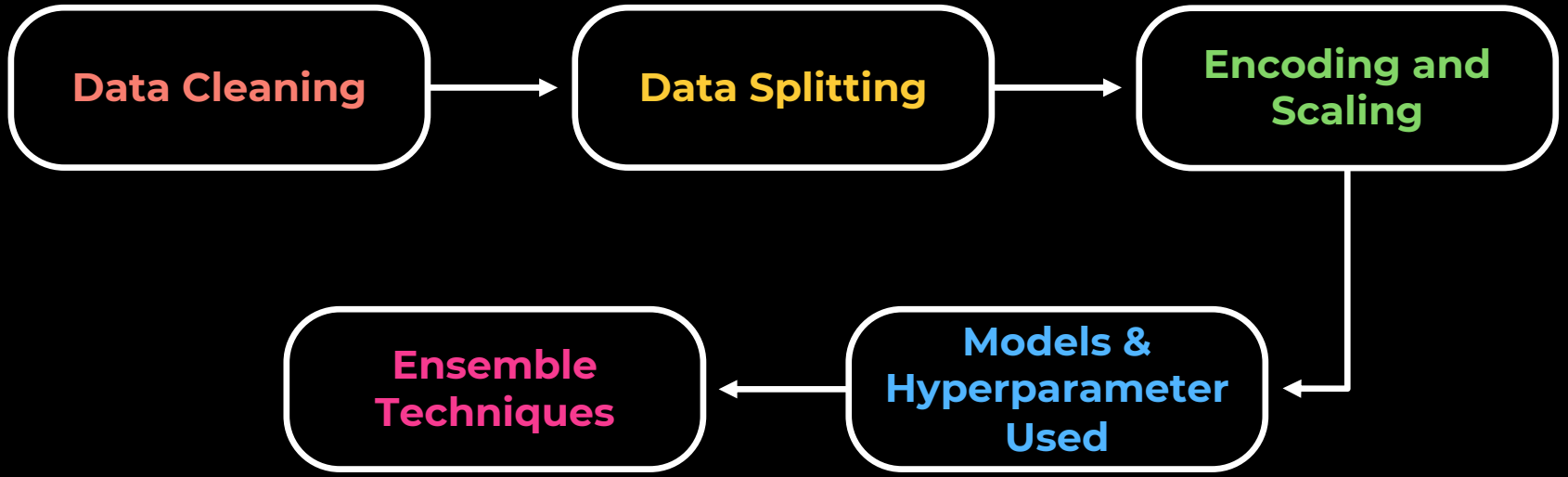
Pipeline


Data Cleaning
Data Splitting
Encoding and Scaling
Models & Hyperparameter Used
Ensemble Techniques

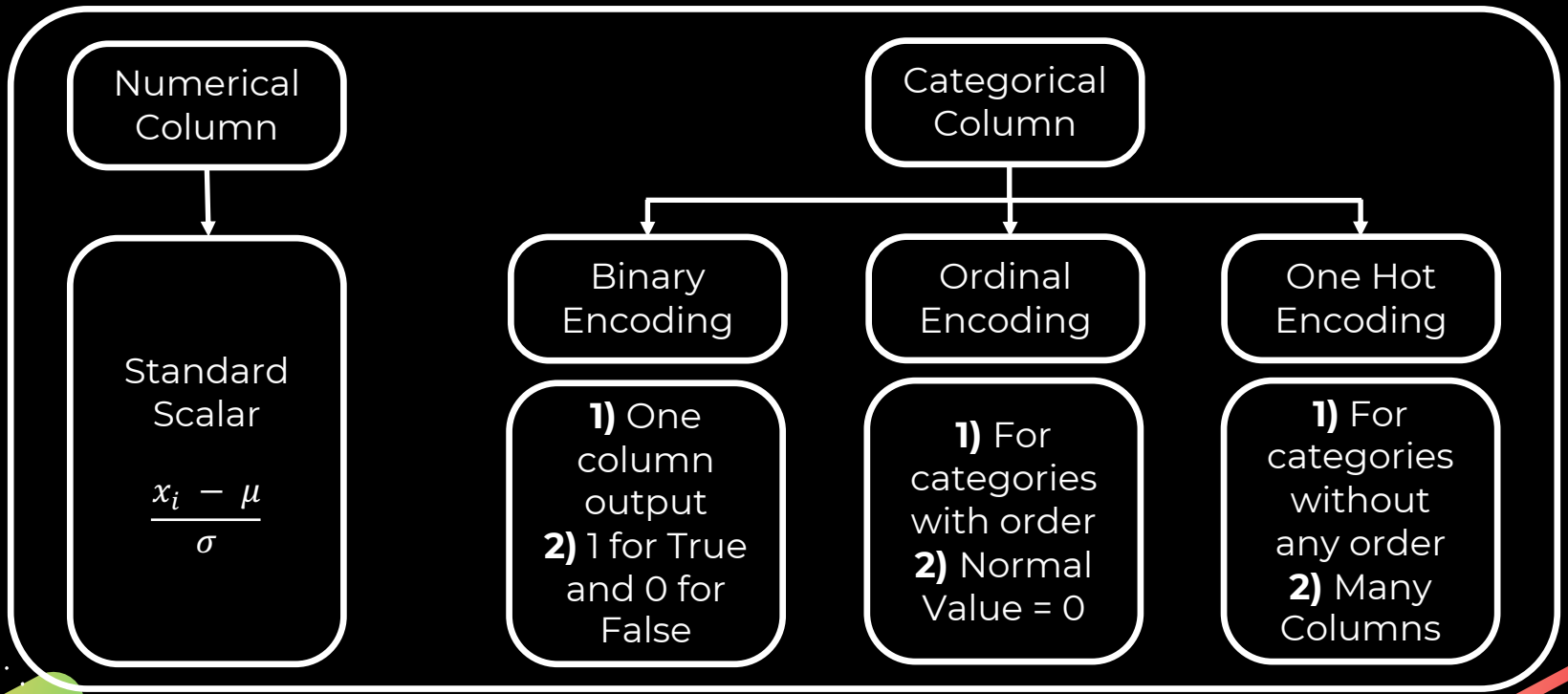
Pipeline - Data Splitting



Pipeline


Data Cleaning
Data Splitting
Encoding and Scaling
Models & Hyperparameter Used
Ensemble Techniques

Pipeline - Encoding & Scaling



Pipeline - Encoding & Scaling

Numerical Column

Standard Scalar

$$\frac{x_i - \mu}{\sigma}$$

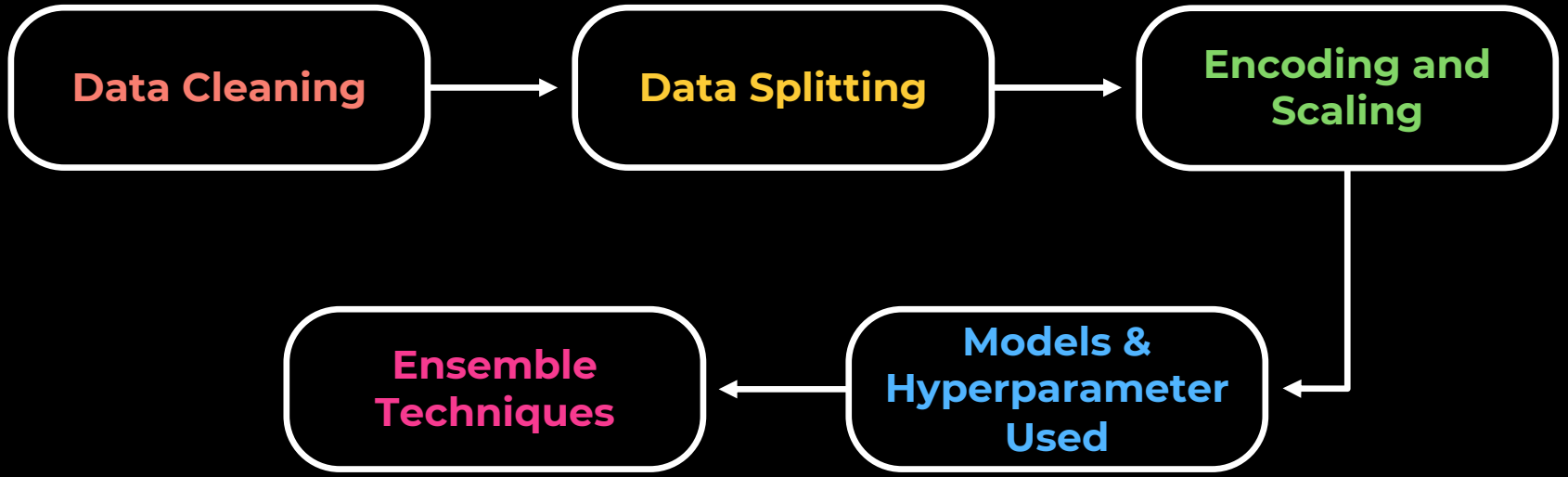
Ordinal Encoding

1) For categories with order
2) Normal Value = 0

Example:
Temp of Extremities

Cold: -2
Cool: -1
Normal: 0
Warm: 1

Pipeline


Data Cleaning
Data Splitting
Encoding and Scaling
Models & Hyperparameter Used
Ensemble Techniques

Pipeline - Models Used

Hyper parameter Tuning

1) Optuna
2) Maximize Foldwise Averaged Valid F1 Score
3) Random Seed included*

Bernoulli Naïve Bayes

1) Assumes all features are binary
2) Works well with large number of columns
3) Valid F1 ≈ 0.67

Nearest Centroid

1) Finds the centroid for each target
2) Returns the closest centroid for inference
3) Different Distance Metrics
4) Valid F1 ≈ 0.63

Random Forest

1) Voting of multiple decision trees
2) Can handle complex patterns & high dimensionality
3) Max Depth ≤ 3
4) Valid F1 ≈ 0.72

* Not Ideal but used as it allows us to experiment with different seeds quickly

Pipeline - Models Used

Neural Network

1) Capture Complex Patterns
2) Non-Linearity with Sigmoid Function
3) Dropouts to improve generalization
4) Less rows in data
5) Valid F1 ≈ 0.64

Autoencoder Neural Network

1) Dimensionality Reduction of input data
2) Trained using reconstruction error
3) Trained on both train and test data (No labels needed)
4) Valid F1 ≈ 0.65

Self Attention Neural Network

1) Interactions between features
2) No Positional Embeddings used
3) Not Nearly enough data !
4) Valid F1 ≈ 0.55

Pipeline - Models Used

XGBoost

1) Extreme Gradient Boosting
2) Inbuilt L1 & L2 Regularization
3) Tree Pruning
4) Handle missing values
5) Ignores useless columns
6) Valid F1 ≈ 0.73

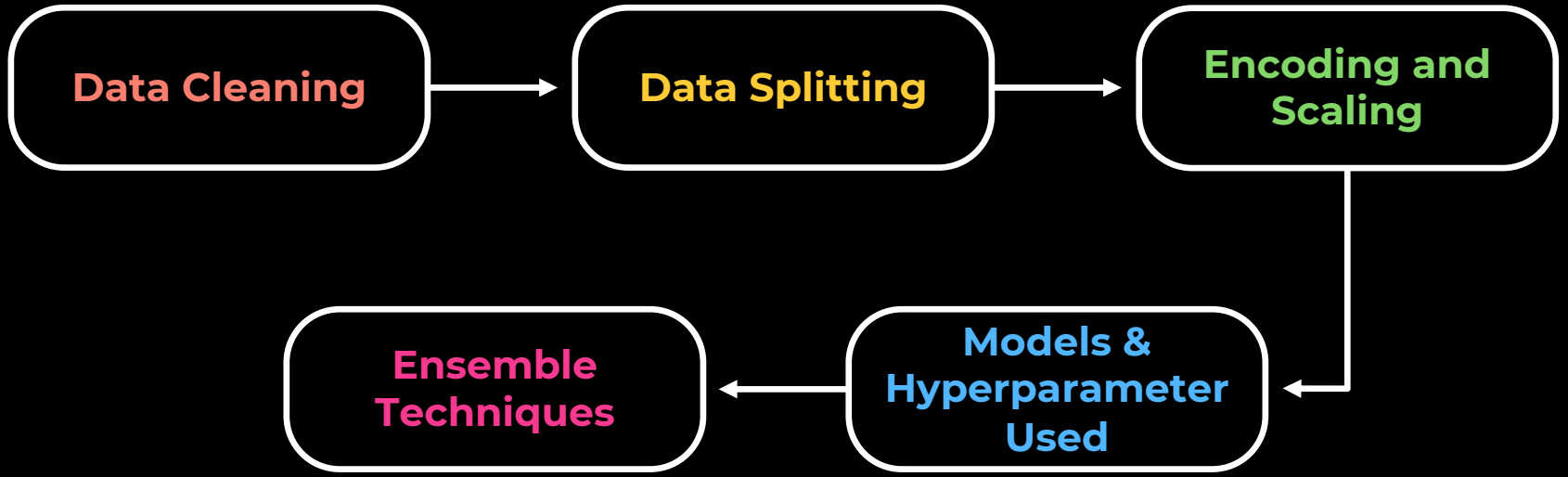
Hist GB

1) Histogram Gradient Boosting
2) Uses a histogram to bin continuous data into discrete values
3) Inbuilt L1 & L2 Regularization
4) Max Depth ≤ 3
5) Valid F1 ≈ 0.74

Light GBM

1) Advanced Hist GB
2) Leafwise Tree Growth not depth wise
3) Bundles Categorical Features into groups
4) Valid F1 ≈ 0.75

Pipeline


Data Cleaning
Data Splitting
Encoding and Scaling
Models & Hyperparameter Used
Ensemble Techniques

Pipeline - Ensemble Techniques

Voting

1) Weighted Voting of all classifiers
2) Weights optimized using Optuna*
3) Hard Voting (Nearest Centroid doesn't give probability)

Stacking

1) Decision Tree used for good explainability
2) Depth ≤ 4
3) No context on which model to choose and when

Stacking & Supplementary Data

1) Added Few Columns from original data to give model some context
2) Columns selected using SHAP values from previous models

* Led to relatively poor generalization on test data

Results

Best Model

1) Hist Gradient Boosting
2) No Ensemble Model
3) Max Depth = 3
4) Max Iteration = 99
5) Min samples per leaf = 5
6) L2 Reg = 4.2×10^{-6}
7) Learning Rate = 0.1

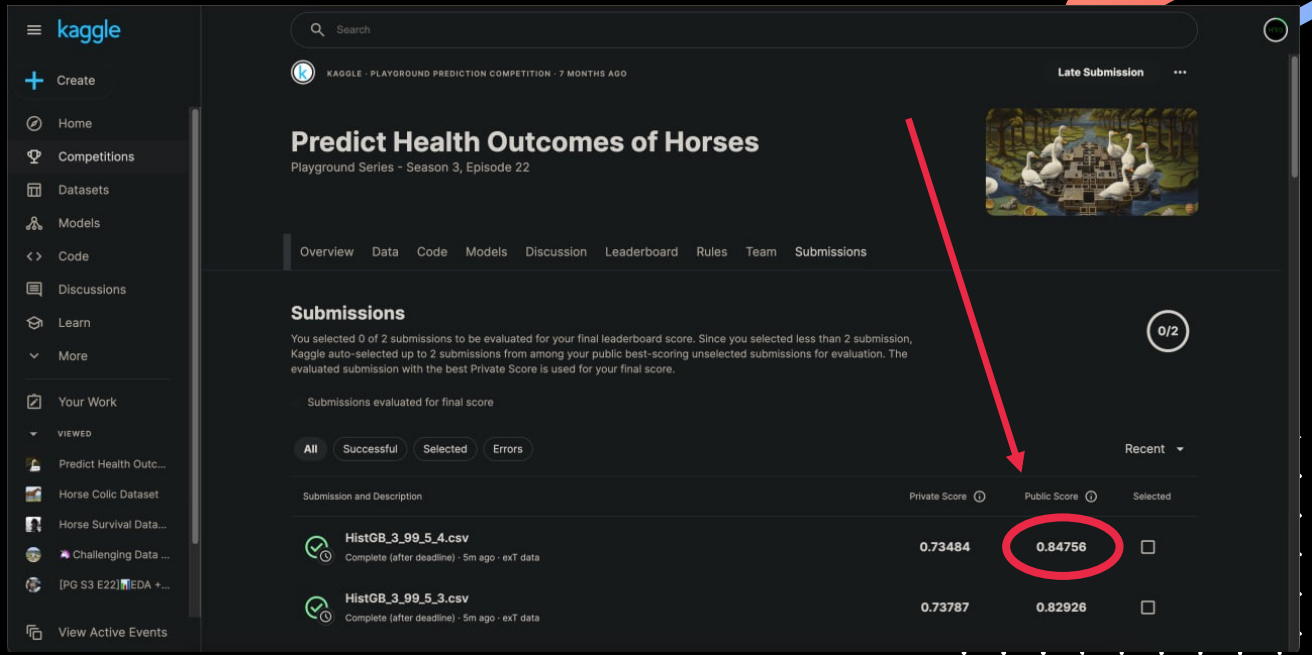
F1 Scores

1) Train F1 = 0.87
2) Valid F1 = 0.71
3) Public Test F1 = **0.84756**
4) Rank = 101
5) Total = 1535
6) **Top 6.7 %**

Results

F1 Scores

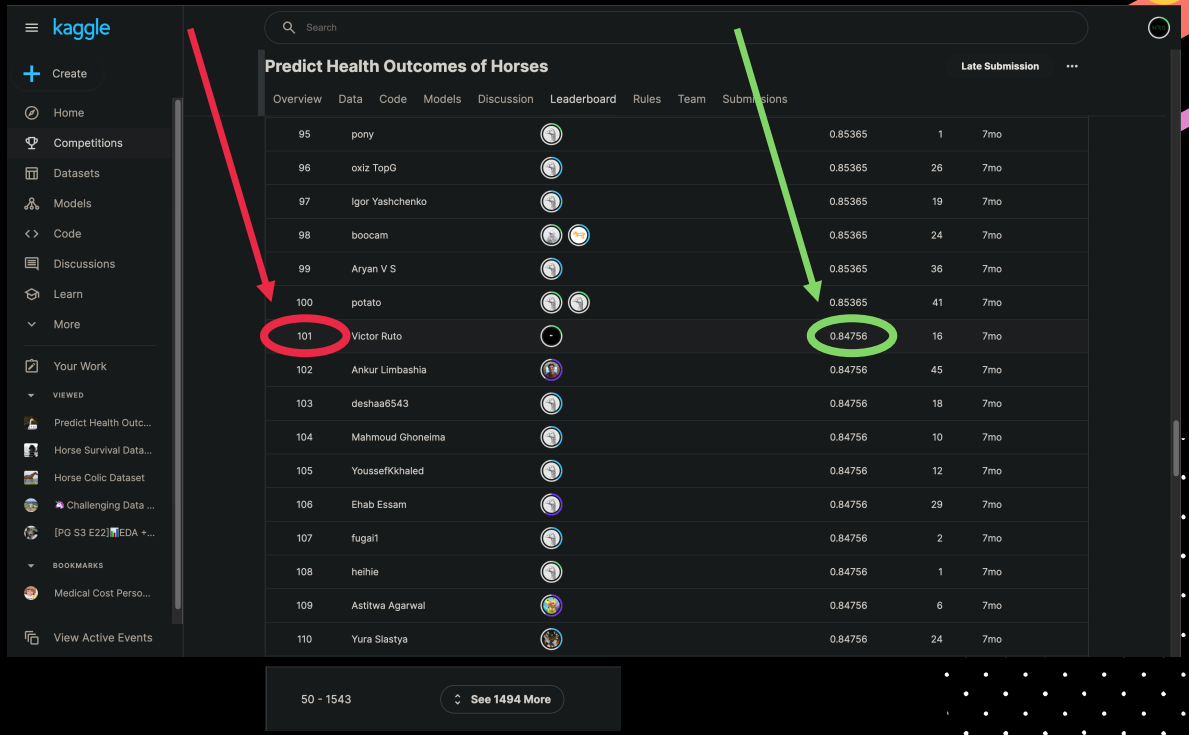
1) Train F1 = 0.87
2) Valid F1 = 0.71
3) Public Test F1 = **0.84756**
4) Rank = 101
5) Total = 1543
6) **Top 6.7 %**



Results

F1 Scores

1) Train F1 = 0.87
2) Valid F1 = 0.71
3) Public Test F1 = **0.84756**
4) Rank = 101
5) Total = 1543
6) **Top 6.7 %**



Conclusion: Challenges Faced

Small Training Set	Large number of NaN values	Small Test Set
1. Less Data to train 2. Overfitting 3. Less Data for Validation	1. Remove all NaN => 771 rows (**-38 %**) 2. Spread across both train and test	1. Public Test Set is 20 % of test data (i.e, 164 rows) 2. Narrow Margins for Test F1
1) K-Fold Cross Validation 2) External Dataset 3) Ordinal Encoding	1) Replacement with mode for categorical 2) Replacement with median for numerical	1) Large number of Experiments 2) L2 Regularization 3) Max Depth ≤ 3

Thanks !

Group 12
Dhanyamraju Harsh Rao
Malavade Sanskar Deepak
Parashar Kshitij
Chen Kian Leong
Mitra Ren Sachithananthan