BASKETBALL Playoffs Qualifcation

Machine Learning Project - Group 12

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

José Isidro - 202006485, Marcos Costa - 202108869,
Rodrigo Moucho - 202108855

TABLE OF CONTENTS

01

BUSINESS UNDERSTANDING

What's our objective?

OBJECTIVE

Our goal is to predict which teams will qualify for the **playoffs** in a given year.

How does the WNBA work?

- 2 conferences: Eastern and Western
- 1st Part - Each team plays against each other
- 2nd Part - Best 4 teams of each conference play the playoffs
- The playoffs will use a format of elimination, with quarter-finals, semi-finals and finals.
- We are given **10** consecutive years of data for the WNBA league and want to predict which teams will make the playoffs in year **XX**, using only the data available at the start of that season.

DATA MINING GOALS

Metrics to evaluate the success of the model

Accuracy

Precision

AUC

Thus, it was defined that the model would be considered successful if:

Accuracy > 65%
Precision > 65%
AUC > 65%

02

DATA UNDERSTANDING

What data do we have?

Dataset Description

Data classes and respective description

Teams	Each record corresponds to a team in a specific season, and contains that team's information and performance metrics like which phase the team made it to, how many field goals were made or attempted, etc. Includes the target variable "playoff".
Players	Contains each player's biographical data, such as weight and height, along with profile data like the position they play.
Players_teams	Connects the players and teams table, thus depicting what players played in each team in what season, as well as the statistic of that player in that season, like minutes played, points and assists.
Awards_players	Indicates the awards and prizes given out each season and that player who won them.

Dataset Description

Data classes and respective description

Series_post Records every series results, namely the winner and loser teams, and the number of losses and wins of the winning team.

Teams_post Contains the results of every team in the post-season/playoffs, namely the number of wins and losses.

Coaches Contains information on each coach and which team they coached.

Coaches in Players

The coaches are not only in the coaches table but also in the players table, even though they were never once players and thus require placeholder values to be included in the players table.

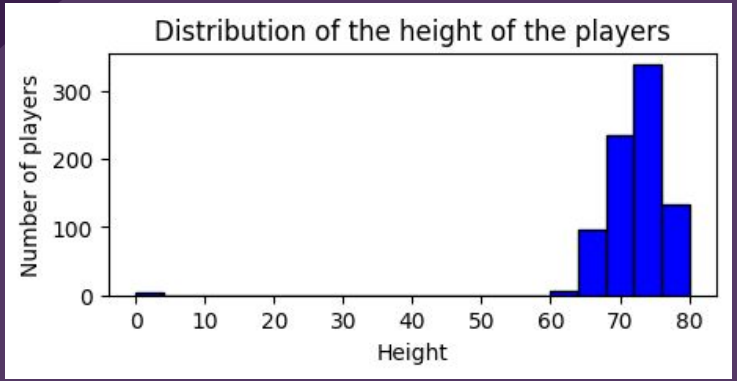
Why the redundancy?

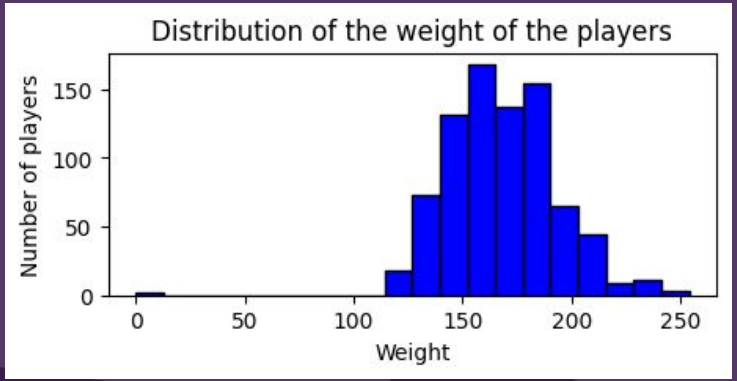
We believe it's because the awards table only refers to players, but there are awards that are meant for coaches, e.g. 'Coach of the Year'. Likely to avoid modifying the dataset's class diagram, all the coaches are added to the players table as a workaround.

Action?

Any players analysis excludes the coaches that have never played.

Players' height and weight





Upon analysing the distribution of the height and weight of the players (excluding coaches), we observe that:

- The height and weight distributions show incorrect values at or near zero.
- The weight distribution includes a few players with weights that are slightly higher than the majority.

Conclusion

Values close to zero appear to be outliers and likely represent incorrect data entries. In contrast, the higher weights seem reasonable and are considered valid data.

Team's Playoffs Qualification Rate

We calculate the rate that a team qualifies to the playoffs like this:

N° of years where they qualified / N° of years they team played

Conclusion

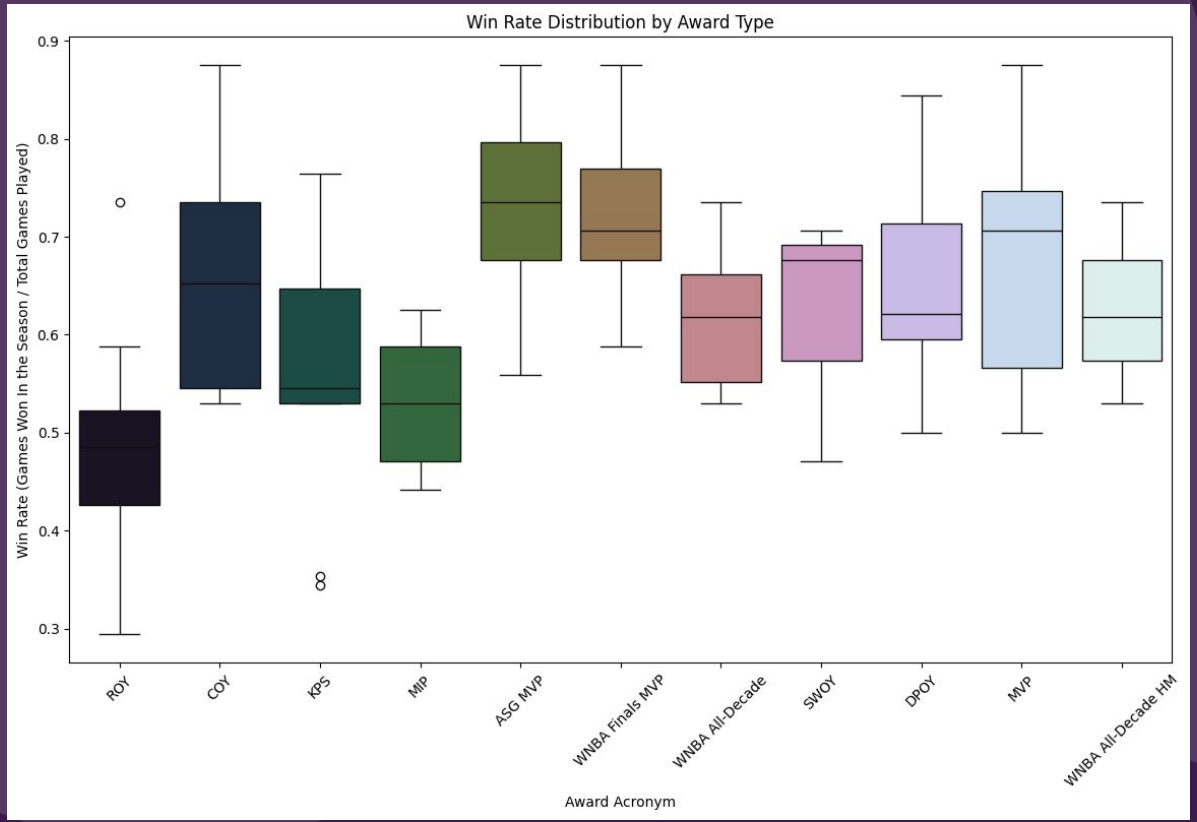
We can see from the plot that some teams are very likely to qualify to the playoffs, while other teams have very low chances of qualifying, assuming the pattern presented continues.

Significance of Awards in Team Performance

We aimed to evaluate the predictive accuracy of awards in relation to team performance based on award type.

The graph illustrates win rates by award, suggesting that certain awards serve as strong performance indicators.

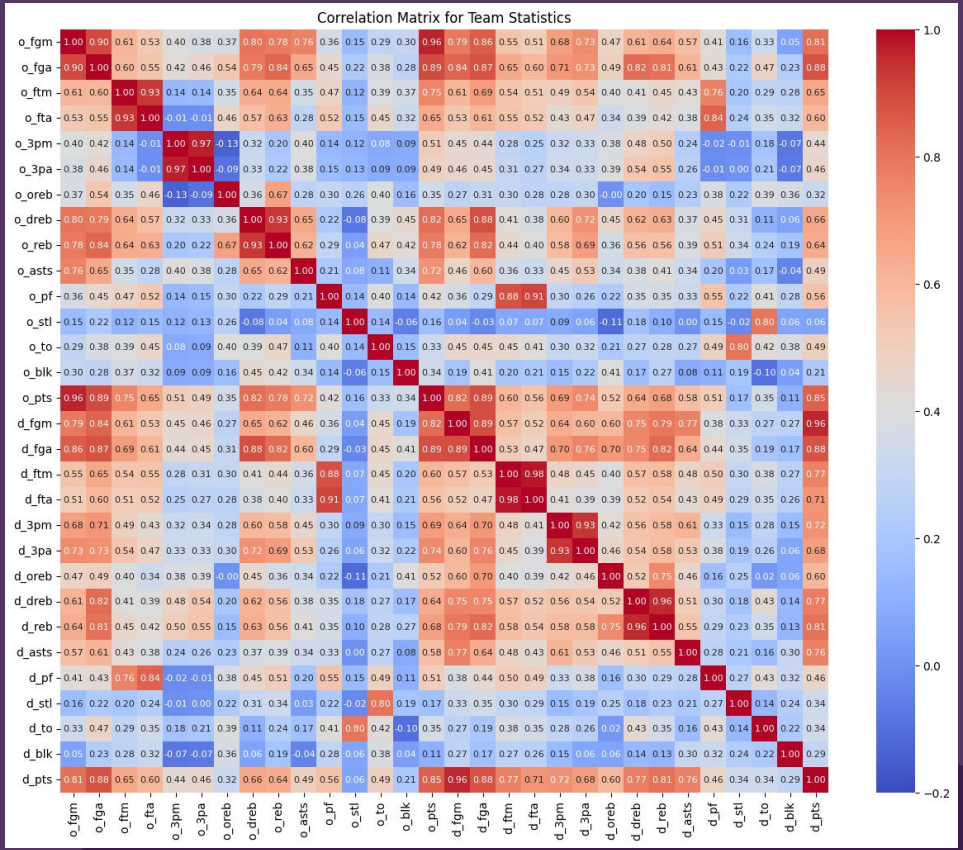
Teams with players who received impactful awards, such as ASG MVP, tend to achieve higher win rates.



Correlation between attributes of Teams

Since the table Teams has a lot of attributes, we graph the correlation between attributes of a subset of attributes in the table Teams.

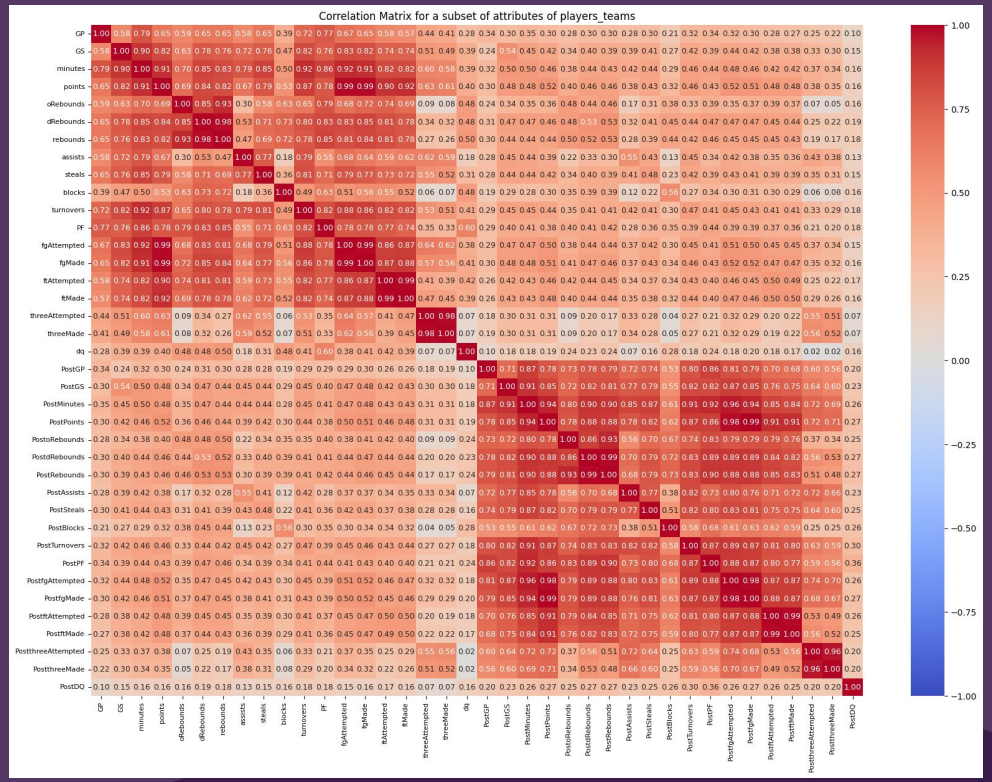
- The number of actions (eg. goals) attempted and made have very high correlation > 0.95
  - For example, ‘o_fgm’ and ‘o_fga’;
  - Teams likely have similar rate of success when they attempt, for example, a goal;
- Other correlated attributes:
  - Points scored with goals made
  - etc.



Correlation between attributes of Players_teams

This slide presents the correlation matrix for a subset of attributes from the players_teams table:

- Overall, the attributes show high correlations.
- The attributes can be roughly divided into two groups, where the average correlation within each group is the strongest.
- This grouping indicates that dimensionality reduction for this table may be more straightforward than for the Teams table. We can achieve this by either removing or combining highly correlated attributes.



03

DATA PREPARATION

Preparing the dataset

Data Integration

- Objective: Compile a single table showcasing each team by year, along with all relevant statistics.
- In order to do this we had to find the best way to aggregate all the important information from all of the tables into the main one

For example:

- Awards table was merged by giving values to each award and making a mean of the points of the players that belonged to a team in a certain year.
- Players table was merged by making a mean of the average attributes (height, weight, age, ...) of the players that belonged to a team in a certain year

Data Quality

- The .csv file represents actual WNBA seasons, provided by the course professors, and can be considered highly accurate.
- Regarding completeness, there were a small amount of outliers and missing values.
- Consistency wise the data was very uniform, although there were some errors (Kim Perrot Award for example)
- The data guaranteed uniqueness since there were no duplicated records.
- The data described real WNBA seasons so the quality and reliability were assured

Redundancy

- Several redundant features were found in many of the data tables given so they had to be removed in order to avoid possible problems when using prediction models

  For example:

- lgID, lgIDWinner, lgIDLoser - always equal to WNBA,
- franchID - same to teamID

  A correlation matrix was also developed in order to find highly correlated features that could be dropped, for example:

- Offensive and Defensive rebounds could be dropped because they correlated highly to the Rebounds feature

Missing Data / Outliers

- Some player attributes, like height and weight, had missing values or extreme outliers. Instead of deleting these records, we estimated missing values by calculating an average BMI based on similar players' data.

- Columns that were consistently empty or held the same value for all entries, such as firstSeason and lastSeason in the player table, were removed to streamline the dataset.

Data transformation for algorithm compatibility

- We mapped the playoff attribute, which is the target variable, to 1s and 0s.
- We used LabelEncoder from sklearn library, to encode all the string variables so that the models could interpret them.

Feature Engineering from tabular data

Award_Points

A mapping of points for each award players can receive

Player_Efficiency

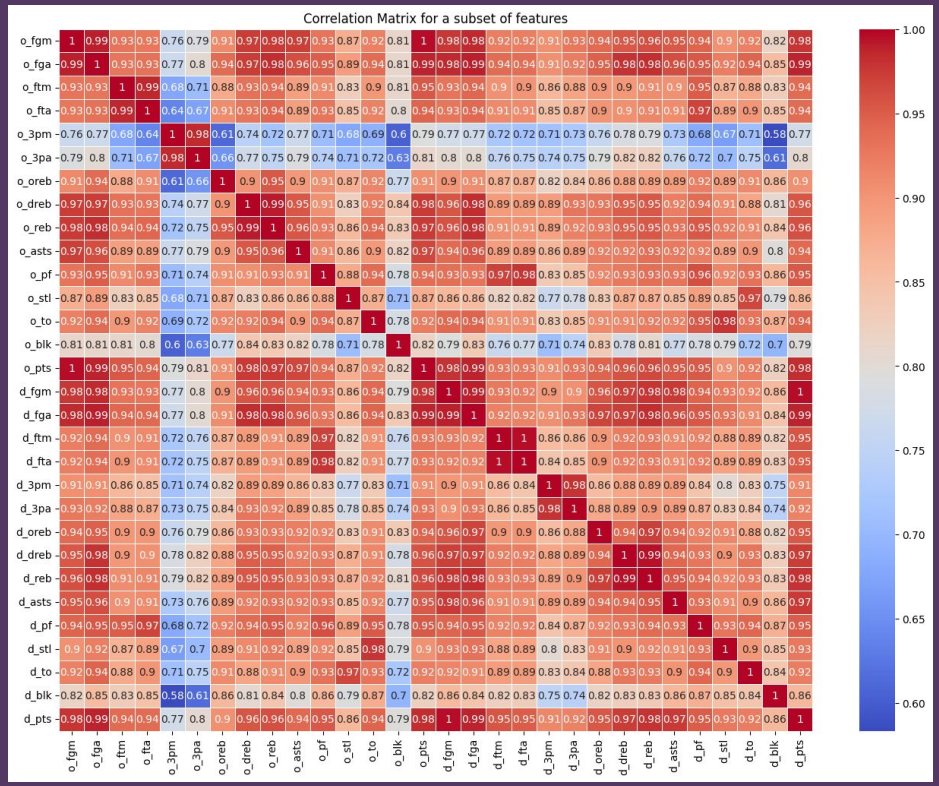
A score representing how the player performed on a given season

Team_Efficiency

A score representing how the team performed on a given season

The final dataset includes average team statistics and player performance metrics from the previous year, focusing on players currently on the team.

Feature Selection



- After testing with the entire dataset:
- Removed attributes with high correlation between them in the teams dataset
- We also removed other attributes manually, and checked if it improved performance

04 RESULTS

Analysis of the results

Diversity of Tasks and Algorithms

Due to the nature of the problem, we have only used Classification Algorithms.

Decision Tree

K-Nearest Neighbors

Support Vector Machine

Logistic Regression

Gradient Boosting

Ada Boost

Random Forest

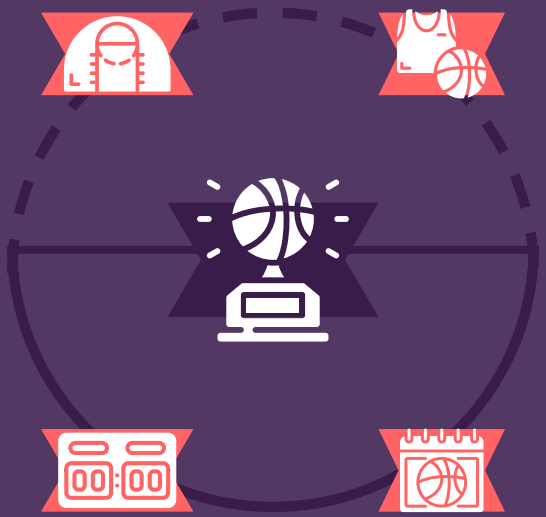
Neural Network

Parameter Tuning

Examples of some parameters that were tuned:

Number of estimators

Eg.: In Random Forest it would be the number of trees



Maximum Depth

Maximum depth of the tree

Learning Rate

Controls the step size at each iteration

Maximum Functions

Maximum number of function evaluations, specific to the **'lbfgs'** solver

Training vs Test

Data Splitting by Year:

- Training Set: Includes all data from years earlier than target year
- Test Set: Includes data from the specific target year
- Eg.: training set: [1-9]; test set [10]

This ensures the model is trained only on past data and tested on unseen future data, avoiding data leakage.

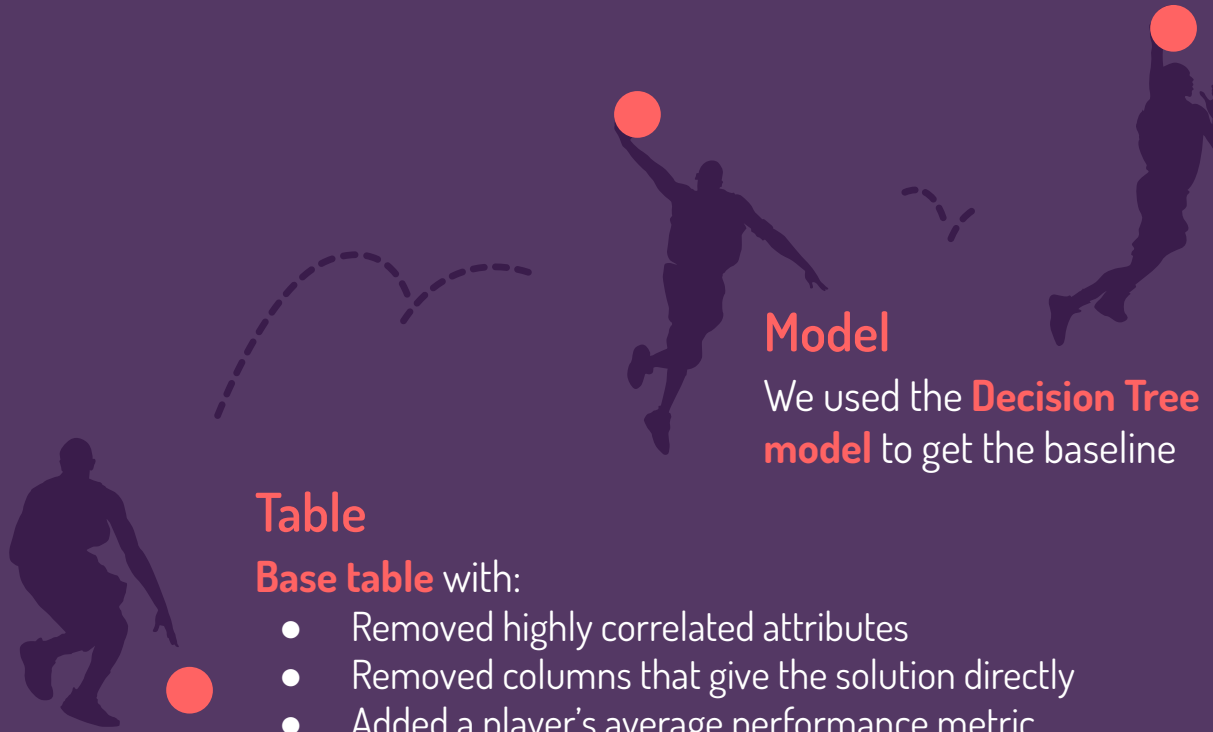
SMOTE is applied to the training set to balance the classes

Performance Measures

Our main objectives are to measure how often the model correctly predicts whether a team made the playoffs or not, how precise it is in the positive cases and assess the model's ability to differentiate between teams across varying thresholds.

- Accuracy: Proportion of correctly predicted outcomes.
- Precision: Ratio of true positives to all predicted positives.
- AUC: Measures overall model performance across all thresholds

Performance Baseline



Baseline

The metrics obtained converged around **50%**

Model

We used the **Decision Tree model** to get the baseline

Table

Base table with:

- Removed highly correlated attributes
- Removed columns that give the solution directly
- Added a player's average performance metric.

Analysis of Results

We tested a big variety of classifier models and researched them in order to understand what were the best ones to use.

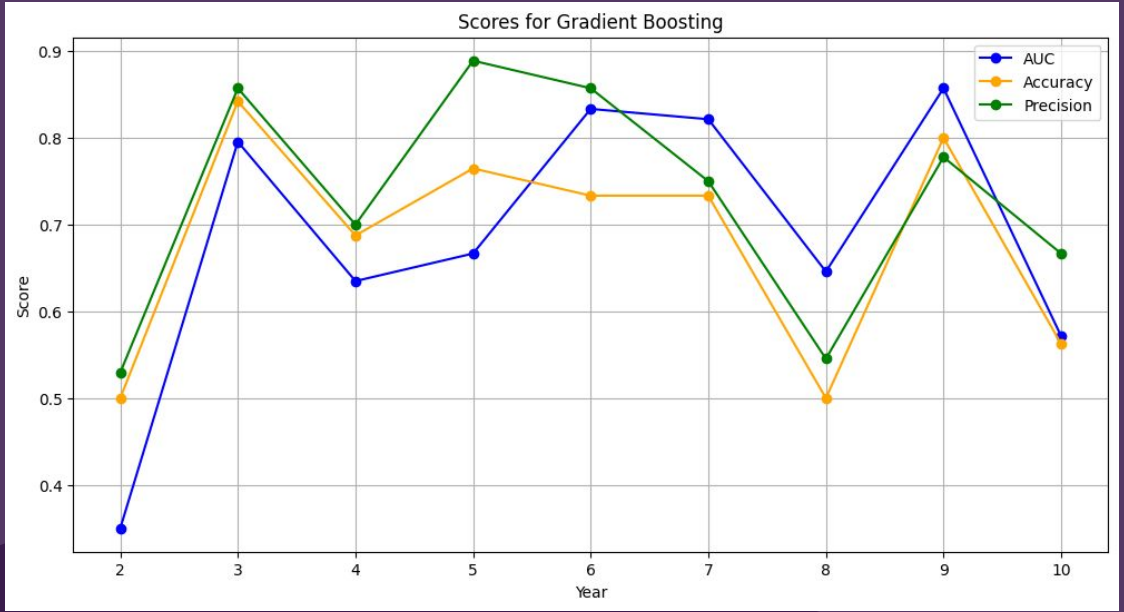
The four models that yielded the best results were:

- **Gradient Boosting**
- **Random Forest**
- **Logistic Regression**
- **Ada Boost**
- **Cat Boost**

An ensemble of these classifiers was developed but the results were not as good as expected so we excluded it.

Analysis of Results

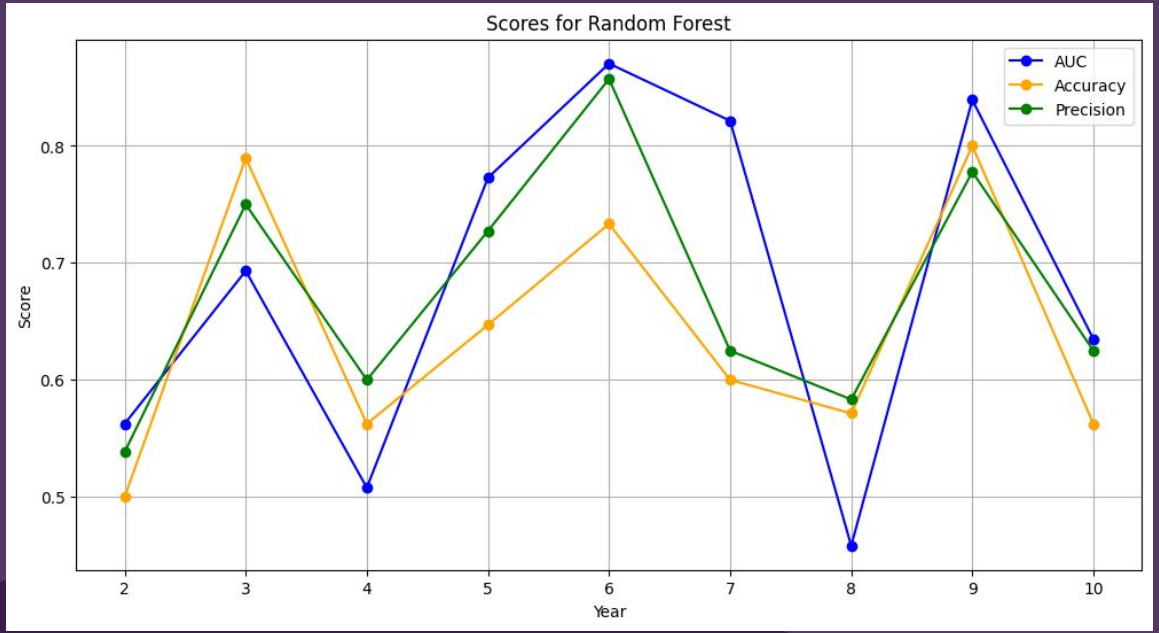
- Gradient Boosting combines multiple weak learners to create a powerful predictive model, achieving the highest precision and accuracy in our analysis



ACCURACY	PRECISION	AUC
0.70	0.74	0.70

Analysis of Results

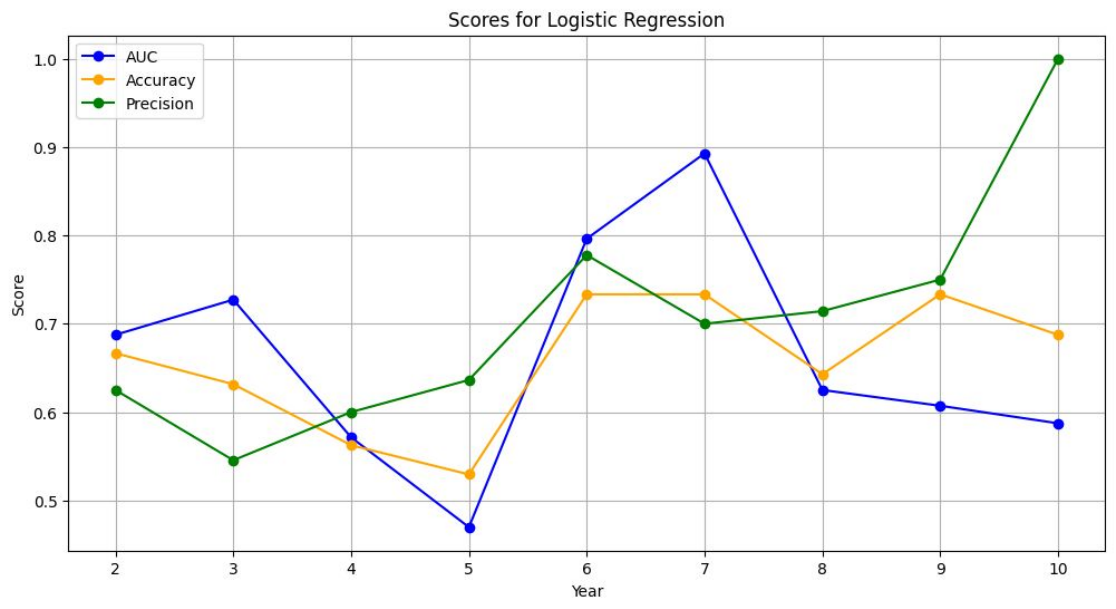
- Random Forest leverages an ensemble of decision trees, excelling in AUC by effectively distinguishing between classes.



ACCURACY	PRECISION	AUC
0.65	0.69	0.69

Analysis of Results

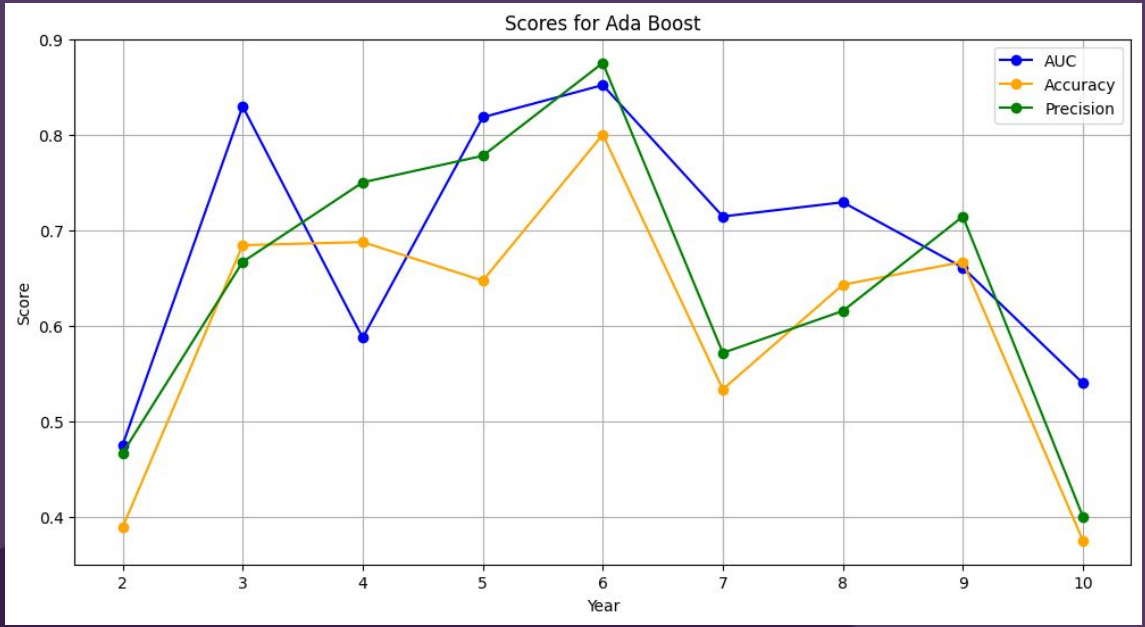
- Logistic Regression offers a simple yet reliable approach, delivering solid performance across accuracy, precision, and AUC as a strong baseline



ACCURACY	PRECISION	AUC
0.65	0.70	0.67

Analysis of Results

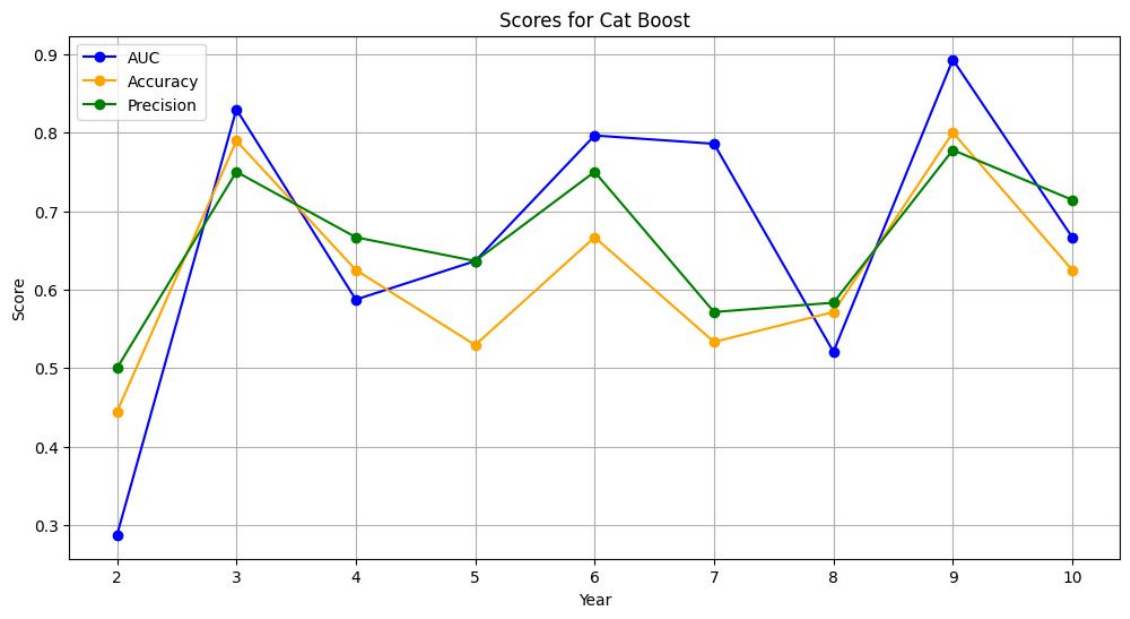
- AdaBoost enhances weak classifiers iteratively, providing balanced performance and complementing other ensemble models in our evaluation.



ACCURACY	PRECISION	AUC
0.63	0.68	0.71

Analysis of Results

- CatBoost stands out for its unique handling of categorical features, bias reduction through ordered boosting, and efficient training using symmetric trees, making it highly effective for structured data.



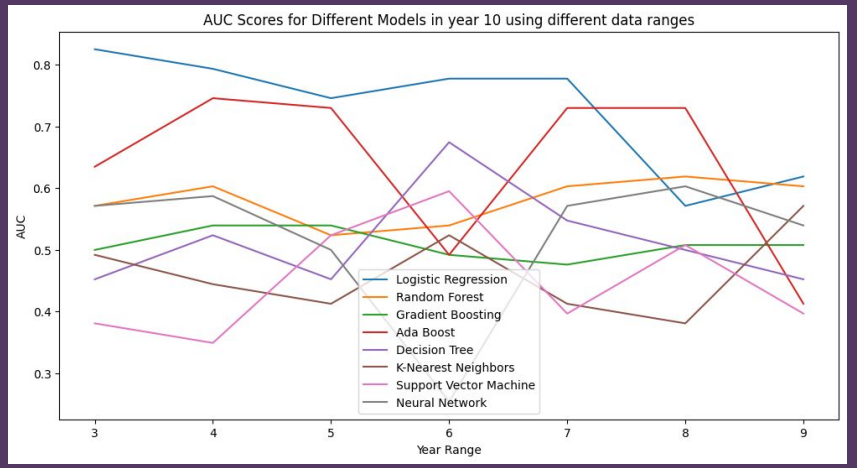
ACCURACY	PRECISION	AUC
0.62	0.65	0.67

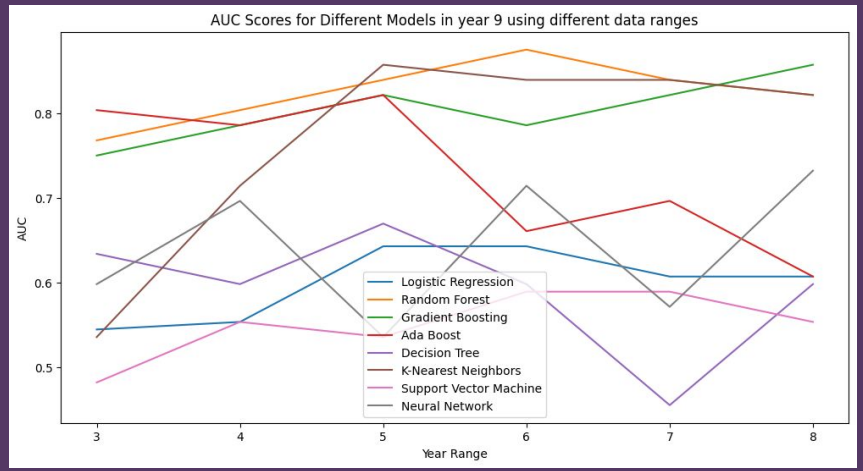
Analyses - Overfitting

We tested the models for years 9 and 10 using **different training ranges**.

Conclusion: In year 10, the AUC starts to drop when using more than 8 years of data. However, results vary across models, so no definite conclusion can be drawn.

Still, we noticed a trend where past a certain point, **adding more data has little impact on model performance.**





Project Management

At the beginning of the semester, a plan and methodology were established and followed to optimize the development of the project.

Tasks were **divided** among team members: data cleaning, exploration, preparation, and predictive modeling.

Development followed an **incremental**, **collaborative** approach for continuous project improvement.

Frequent updates were made across all stages: data preparation, model testing or parameter tuning.

Project Management: Tools

- **Github** - code repository, branches with different approaches
- **Python Notebooks** - code and documentation in one place
- **Discord** - main communication channel, work distribution

Kaggle Submissions

For the Kaggle submissions our strategy was:

- Test different models and test results
- Explore the best submission by tweaking with the error probabilities

Machine Learning Tools Used

Pandas

Extracted, interpreted and merged the datasets

Matplotlib

Plotted graphics for the analysis of the data

Seaborn

Constructed heatmaps to visualize the corr. matrix

Scikit-learn

Used various models to determine the best for our predictions

Imbalance-learn

Used SMOTE for data balancing

THANKS

Machine Learning Project - Group 12

U.PORTO

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

José Isidro - 202006485, Marcos Costa - 202108869,
Rodrigo Moucho - 202108855