Assignment 1

Kaggle Competition

Vishal Raj
Student ID: 14227627
(Github Link)

5th September 2023

36120 - Advanced Machine Learning Application
Master of Data Science and Innovation
University of Technology of Sydney

Table of Contents

1. Executive Summary

Introduction and Scope

Basketball, with its journey from college to NBA arenas, offers young athletes both challenges and chances to shine, particularly during the NBA Draft. This annual event is a nexus of hope and strategy, determining which college stars ascend to professional ranks. The multi-faceted draft process, influenced by skills, potential, performance, and occasionally, fortune, necessitates a precise, data-driven approach.

Objectives and Motivation

This project's core objective was to harness machine learning to predict a college player's likelihood of NBA selection based on seasonal statistics. This endeavour was driven by the vision of a tool benefiting diverse stakeholders. For NBA teams, the stakes are high: the choice between recruiting a future star and overlooking one. Analysts and commentators can infuse their narratives with data-driven insights. College players and their coaches get a roadmap for enhancement, while fans receive fodder for passionate debates.

Methodology and Technologies

The project blueprint involved structured stages from data gathering to model evaluation. Random Forest and XGBoost, two potent machine-learning algorithms, were employed for predictions. Python facilitated this project with its arsenal of data science tools like Pandas, Matplotlib, Seaborn, and Scikit-learn. GitHub housed the project, fostering collaboration and versioning.

Results and Impact

The endeavour's success was evident, with both models achieving ROC-AUC scores surpassing 0.9999, signifying impeccable prediction capability. The business implications are profound:

- NBA franchises, armed with these insights, can draft with precision.
- Potential NBA players gain clarity on their prospects.
- Analysts receive a robust tool for enriched discussions.
- Fans get a deeper dive into the draft's intricacies.

2. Business Understanding

a. Business Use Cases

The NBA Draft: A Pivotal Decision

The NBA Draft is a major annual event influencing players, teams, and fans. For NBA franchises, draft choices are strategic moves with long-standing effects. A prudent pick can bring a cornerstone player, while a poor decision might hinder the team's progress and financial stance.

The Shift to Data-Centric Selections

While traditional methods involving interviews and intuition still hold value, the draft process has room for data-led decisions. This project offers a predictive tool using machine learning to determine which college players stand a chance in the NBA draft.

Stakeholder Implications

- **NBA Teams:** The model acts as a supplementary tool, aiding teams in aligning draft decisions with data insights.
- **Sports Analysts:** It offers analysts a concrete foundation for draft discussions.
- **Players & Coaches:** The tool highlights areas for player improvement, guiding training directions.
- **Fans:** The model ignites fan theories and discussions, enhancing engagement.

b. Key Objectives

Filling the Analytical Void

This project strives to introduce a consistent, data-led perspective to the draft process, countering the inherent subjectivity.

Tuning to Stakeholder Needs

- **NBA Teams:** Desire a tool that harnesses player metrics to predict draft potential.
- **Analysts:** Seek empirical backup for their draft insights.
- **Players & Coaches:** Want clarity on performance metrics impacting draft chances?
- **Fans:** Crave a platform for more insightful player discussions.

Envisioning a Machine Learning Solution

The project, through machine learning, aims to curate a tool capable of predicting NBA draft outcomes using college basketball stats. It's tailored to cater to the diverse needs of every stakeholder in the NBA Draft ecosystem.

3. Data Understanding

Dataset Overview: The Foundation of Predictive Modelling

Our project leans on an exhaustive dataset capturing college basketball players' statistics, encapsulating different gameplay aspects. The dataset's richness, covering scores, assists, rebounds, and more intricate metrics like the Effective Field Goal Percentage (eFG), forms the backbone for our predictive model. The primary goal is to harness this data to derive insights that inform NBA draft choices.

Data Sources: Bridging Relevance with Depth

The data amalgamates various datasets: 'train.csv', 'test.csv', 'sample_submission.csv', and 'metadata.csv'. Their roles are distinct:

- **Train.csv:** Holds historical data inclusive of player stats and draft outcomes.
- **Test.csv:** Akin to the training set without the draft outcome, earmarked for model testing.
- **Sample_submission.csv:** A guide for Kaggle prediction submissions.
- **Metadata.csv:** Complements the primary datasets with auxiliary information for deeper analysis.

Data Inspection: An Introductory Dive

A cursory glance via the 'display_head' function offers a snapshot of the dataset, revealing potential areas like missing values or inconsistent data types that warrant rectification.

Features and Their Relevance

- **Team & Conference:** Indicates the player's affiliating team and conference, acknowledging variances in competition levels across conferences.
- **Games Played (GP):** Reflects a player's exposure during the season.
- **Offensive & Defensive Metrics:** Key indicators like 'ORtg' and 'DRtg' shed light on players' efficacy.
- **Shooting Metrics:** Stats like 'eFG%' and 'TS%' gauge shooting competency.
- **Rebounding:** Metrics such as 'ORB%' and 'DRB%' highlight a player's rebounding skill.
- **Ball Handling:** 'AST%' and 'TO_per' provide playmaking and ball control insights.
- **Defensive Playmaking:** 'stl_per' and 'blk_per' evaluate defensive disruptiveness.
- **Advanced Metrics:** 'Bpm', 'obpm', and 'dbpm' quantify a player's impact.
- **Physical & Seasonal Attributes:** Variables like 'ht', 'year', and 'type' offer context.

- **Other Metrics:** Detailed metrics like 'ast_tov', 'rimmade', and 'dunksmade' offer deeper gameplay insights.

Data Limitations: Challenges and Workarounds

Like every dataset, this one has constraints:

- **Missing Values:** Some columns, e.g., 'pick', were omitted due to significant data gaps.
- **Imbalanced Data:** The 'drafted' target variable displayed imbalance.
- **Feature Scaling:** Essential for algorithms demanding uniform scale.

We employed strategies like imputation, SMOTE, and scaling to navigate these.

Exploratory Data Analysis (EDA)

Intensive EDA fortified our understanding, including techniques like the 'display_head' function, 'target_distribution' analysis, and feature correlation assessments. This knowledge was crucial for refining our data preparation and modelling approach.

4. Data Preparation

Commencing with Data Loading

The datasets are loaded using the 'load_data' function from 'make_dataset.py'. This function ingests files like 'train.csv', 'test.csv', 'sample_submission.csv', and 'metadata.csv', setting the groundwork for the modelling phase.

Addressing Missing Values

A pivotal task is managing missing data. The 'missing_values_analysis' function pinpoints columns with absent values, quantifying the extent of the issue. Features like 'pick', with significant missing data, are omitted, while others undergo statistical imputation.

Data Imputation Techniques

The 'DataPreprocessor' class, found in 'data_preprocessor.py', is harnessed to fill gaps. Numeric columns utilise median values, and categorical ones use mode, ensuring data retention.

Enhancing with Feature Engineering

Feature engineering is the project's linchpin. Techniques employed include:

- Temporal Adjustments: The year undergoes sine and cosine transformations to capture its cyclical pattern.
- Introducing New Features: 'GP_Min_per' and 'Ortg_eFG' are created to encapsulate intricate data relationships.
- Outlier Management: Outliers are detected and removed using the Z-score method.

Balancing Data Classes

The 'apply_smote' function mitigates model bias from class imbalance. It employs the Synthetic Minority Over-sampling Technique (SMOTE) to produce synthetic samples, achieving class equilibrium.

Encoding and Data Scaling

Data encoding converts categorical columns for algorithm compatibility. The 'encode', 'scale', and 'encode_and_scale' functions manage one-hot encoding and feature scaling via the 'StandardScaler'.

Segmenting Data: Training vs. Validation

Data is partitioned into training and validation subsets using the 'train_test_split' function, ensuring distinct validation data.

Standardising Test Data

The test dataset undergoes a parallel processing procedure. Functions like 'encode_and_scale_test' and 'encode_and_scale_test_features' maintain consistency and scale equivalently.

5. Modelling

Algorithm Selection: Random Forest and XGBoost

We adopted two classifiers, Random Forest and XGBoost, based on their strengths:

- **Random Forest:** Esteemed for its robustness and interpretability, it thrives on large datasets and offers insightful feature importance.
- **XGBoost:** Celebrated for efficiency, this gradient-boosted tree model naturally handles missing data and boasts diverse tuning parameters.

Hyperparameter Optimisation

The project harnesses the 'grid_search' method, invoking 'GridSearchCV' for meticulous hyperparameter tuning for both algorithms. By prioritising ROC-AUC scores, we ensure models are performance-optimised.

Model Training Phases

Post-tuning, models are trained using:

- **Full Feature Set:** This phase assesses the raw data's predictive strength, harnessing all features.
- **Selected Features:** This phase is driven by feature importance metrics, focusing on the most influential features.

a. Approach 1: Random Forest without Feature Selection

- **Algorithm & Parameters:** Random Forest with fine-tuned 'n_estimators', 'max_depth', etc.
- **Preprocessing:** Initial steps include missing data management and scaling without explicit feature selection.
- **Training:** The model, trained on a balanced dataset via 'SMOTE', ensures unbiased learning.

b. Approach 2: Random Forest with Feature Selection

- **Algorithm & Parameters:** Like Approach 1 but further optimised for a refined feature set.
- **Feature Strategy:** A data-centric approach that retains features based on their Random Forest-determined importance. Redundant features are discarded to mitigate multicollinearity.
- **Training:** Consistent with Approach 1, the model benefits from a streamlined feature set.

c. Approach 3: XGBoost with and without Feature Selection

- **Algorithm & Parameters:** XGBoost, with parameters like 'learning_rate' fine-tuned for full and reduced feature sets.
- **Feature Strategy:** Alongside full and reduced feature sets, interaction features are introduced, and certain features undergo temporal transformations.
- **Training:** Like prior methods, it uses a balanced dataset, offering a holistic view.

Securing the Models

Post-validation, top-performing Random Forest and XGBoost models are stored via 'Joblib', ensuring ready deployment for subsequent tasks and seamless stakeholder integration.

6. Evaluation

a. Evaluation Metrics: Emphasising ROC-AUC

Our evaluation hinges on the ROC-AUC score for its multiple benefits:

- **Discrimination:** It gauges the model's skill in discerning draft likely players from others, matching our goals.
- **Imbalanced Dataset:** With 'drafted' as imbalanced, ROC-AUC, factoring sensitivity and specificity, outperforms mere accuracy.
- **Stakeholder Alignment:** This metric aligns with stakeholders, emphasising discriminative ability over accuracy.

b. Results and Analysis: Exceptional Performance

The Random Forest and XGBoost models showcased remarkable ROC-AUC scores:

- No Feature Selection:
  - Random Forest: 0.999986132207113.
  - XGBoost: 0.9999719323000272.
- Feature Selection:
  - Random Forest: Improved to 0.9999896602113071.
  - XGBoost: Rose to 0.9999779766233615.

Comparative Insights

Both models improved with feature selection, albeit marginally. While impressive, such high scores necessitate scrutiny for overfitting risks.

c. Business Impact and Benefits: Drafting Transformed

Model Revolution

This model transforms talent identification:

- **Informed Drafting:** Teams can integrate empirical predictions, potentially uncovering overlooked stars.
- **Resource Efficiency:** The model ensures better resource deployment by streamlining scouting efforts.
- **Contract Talks:** Knowing a player's promise aids in contract negotiations.
- **Fan Involvement:** Engaging fans with draft predictions can amplify their connection to the sport.

Extended Benefits

Besides NBA teams, other beneficiaries include:

- **Analysts:** They can fortify discussions with predictive insights.
- **Players/Coaches:** Recognising draft probabilities can shape focused training.
- **Advertisers/Merchandisers:** Predictions help strategise for potential NBA inductees' rising fame.

d. Data Privacy and Ethical Concerns: Treading Carefully

Ethical Data Handling

In today's data-conscious world, ethical data management is paramount:

- **Informed Players:** Even if data is public, players should know their statistics undergo such scrutiny.
- **Prediction Pitfalls:** A model isn't flawless regardless of its efficacy. Incorrect predictions might impact a player's perceived potential.

Responsible Data Application

- **Clarity:** Clear communication about model capacities is essential to manage expectations.
- **Feedback and Refinement:** Aligning predictions with actual outcomes can refine the model.
- **Bias Vigilance:** Constantly review data to negate biases that can skew fairness.
- **Ethical Use:** Ensuring the model aids but doesn't replace traditional methods ensures comprehensive player evaluation.

7. Deployment

Deployment Process: Model Serialization and GitHub Integration

Transitioning our Random Forest and XGBoost models from research to deployment involved:

- **Model Serialisation:** Using Python's 'joblib', both models were serialised post-training, ensuring easy reloading without retraining.
- **Modularisation:** The code was organised into separate modules for data cleaning, preprocessing, training, prediction, and visualisation. This modular structure promotes maintainability and clarity.
- **GitHub Integration:** The project was hosted on GitHub, endorsing collaborative development, version control, and public accessibility. Enthusiasts or NBA teams can fork, analyse, and refine as needed.

Real-world Considerations: Ensuring Robustness and Reusability

- **Data Versioning:** As new seasons unfold, platforms like DVC, combined with GitHub, could track evolving data and guarantee result reproducibility.
- **Scalability:** Our modular code design ensures seamless updates or adjustments without overhauling the system.
- **Security:** The public GitHub repository mandates rigorous security checks. Using '.gitignore' and pre-commit reviews can bolster data safety.
- **Documentation:** Comprehensive documentation, including READMEs and comments, guarantees user-friendly repository navigation and model utilisation.

Recommendations for Future Deployment

- **Continuous Integration and Testing:** Embracing CI/CD tools, such as GitHub Actions or Travis CI, can safeguard against potential bugs in future updates. Automated tests maintain model and process integrity.
- **Collaboration:** Leveraging GitHub's collaborative essence can usher in fresh insights and innovative techniques, optimising the project.
- **Feedback Loop:** Instituting a user feedback channel can spotlight model inconsistencies or areas of refinement, enhancing model resilience.

8. Conclusion

Summarising Key Findings and Insights

Our rigorous project, which predicted NBA draft chances, showcased outstanding results. The Random Forest and XGBoost models, with ROC-AUC scores nearing perfection, validated our methodology's strength. Feature engineering was pivotal, enhancing the model's depth and precision. Model validation confirmed its robustness against overfitting.

Reflecting on the Project's Success

The project met and surpassed its objectives. It crafted a reliable predictive model, benefiting a spectrum of stakeholders: NBA teams, analysts, players, and coaches. NBA franchises gain a scientific edge in draft decisions, analysts receive quantitative support, while players and coaches get a targeted training compass.

Meeting Stakeholders' Requirements

More than a technical marvel, the model offers actionable draft insights. Its potential business impact could redefine draft strategies, aligning with stakeholders' demand for a precise, actionable, and trustworthy prediction tool.

Future Work, Recommendations, and Next Steps

While the model excels, enhancements beckon:

- Temporal Features: Integrating longitudinal data could spotlight a player's evolving performance.
- Model Diversification: Venturing into Neural Networks or Gradient Boosting Machines might offer fresh insights.
- Real-world Deployment: As hinted in "Deployment", the model's real-world application promises broader validation.
- Continuous Learning: Annual NBA drafts bring fresh data; periodic model updates are imperative.
- Ethical and Regulatory Compliance: As deployment nears, addressing ethical issues, especially data privacy and fairness, is crucial.

9. References

Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32. https://doi.org/10.1023/A:1010933404324

O'Connor, K. (n.d.). The Ringer's 2023 NBA draft guide. The Ringer's 2023 NBA Draft Guide. https://nbadraft.theringer.com/

Albon, C. (2018). Machine learning with python cookbook. O'Reilly Online Learning. https://www.oreilly.com/library/view/machine-learning-with/9781491989371/ch04.html

Kuhn, M., & Johnson, K. (2019). Feature Engineering and Selection: A Practical Approach for Predictive Models (1st ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9781315108230

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794. https://doi.org/10.1145/2939672.2939785

Brownlee, J. (2020, August 17). Ordinal and one-hot encodings for Categorical Data. MachineLearningMastery.com. https://machinelearningmastery.com/one-hot-encoding-for-categorical-data/

György Kovács, Smote-variants: A python implementation of 85 minority oversampling techniques, Neurocomputing, Volume 366, 2019, Pages 352-354, ISSN 0925-2312. https://doi.org/10.1016/j.neucom.2019.06.100

Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27(8), 861-874. https://doi.org/10.1016/j.patrec.2005.10.010

Friedman, J., Hastie, T., & Tibshirani, R. (2001). The Elements of Statistical Learning. Springer Series in Statistics. https://doi.org/10.1007/978-0-387-21606-5

Han, J., Pei, J., & Kamber, M. (2012). Data Mining: Concepts and Techniques. Morgan Kaufmann. https://doi.org/10.1016/C2009-0-61819-5

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning. Springer Texts in Statistics. https://doi.org/10.1007/978-1-4614-7138-7

Provost, F., & Fawcett, T. (2013). Data Science for Business. O'Reilly Media, Inc. https://www.researchgate.net/publication/256438799_Data_Science_for_Business

Witten, I. H., Frank, E., & Hall, M. A. (2011). Data Mining: Practical Machine Learning Tools and Techniques.

Morgan Kaufmann. https://doi.org/10.1016/C2009-0-19715-5

Kelleher, J. D., Mac Namee, B., & D'Arcy, A. (2015). Fundamentals of Machine Learning for Predictive Data Analytics. MIT Press. https://mitpress.mit.edu/books/fundamentals-machine-learning-predictive-data-analytics

Friedman, J., Hastie, T., & Tibshirani, R. (2001). The Elements of Statistical Learning. Springer Series in Statistics. https://doi.org/10.1007/978-0-387-21606-5

Schapire, R. E., & Freund, Y. (2012). Boosting: Foundations and Algorithms. MIT Press. https://doi.org/10.7551/mitpress/8291.001.0001