**Developing an Optimized March Madness Bracket**

Mia Alfonso, Madison Gondrella, Kevin Kelley, Matthew Lafleur

*Cain Department of Chemical Engineering, Louisiana State University, Baton Rouge Louisiana*

College basketball is a popular and growing sport across the nation. At the end of regular playing season, college basketball starts a post-season tournament commonly referred to as “March Madness.” With an average viewership of 10 million viewers per game, this tournament is a much-anticipated event where a mix of casual and diehard basketball fans watch to see which team will be name the National Collegiate Athletic Association (NCAA) champion. Estimates have shown that 60-100 million March Madness brackets are filled out each year by fans of the sport. These entries range from picking winners based on mascots, team names, regular season performance, and historical performance in the tournament.

The bracket is composed of 68 teams that qualify in various ways, either by winning their conference championship or being selected by a committee for an at-large bid. The selection committee considers a team’s overall regular season record, strength of schedule, and NET ranking when determining seedings for each region for non-automatic qualifiers. The first-four teams play in an elimination round to be slotted into the 64-team bracket. Teams are separated into 4 regions, 16 teams in each, and are balanced by the selection committee based on the criteria above. Teams play through 6 rounds of elimination to determine the national champion.

The goal of this project is to take a source of data for college basketball teams and use machine learning and data analysis to fill out a March Madness bracket based on matchup predictions from our model.

The use of data analysis and machine learning, in conjunction with other tools/methods, can be used to find meaningful insights and make predictions based off a range of data containing various types of statistics.

For this project, college basketball statistics from Kenpom.com will be used as the source of the data. This database has detailed statistics for every D-1 program, which includes offensive, defensive, and tempo ratings. These statistics are used as a general gauge for how well a team performs and can be studied to determine the probability of any one team winning a game. These ratings will be analyzed through machine learning in Python to help formulate a predictive model to allow us to best determine the winning bracket for this year’s NCAA March Madness Tournament.

Unsupervised learning will be a useful tool for this dataset as clustering and association rule learning will allow for determination of natural groupings of high, middle, and low performing teams while also giving insight on the importance of each statistic to determine potential weight factors for the given statistics.

Supervised learning will be used to determine matchup probabilities for each game of the bracket, using the historical data from our database. The model will learn patterns from the data based on a learning algorithm. Once trained, the model will be tested with unseen data and evaluated for accuracy and precision. After satisfactory training and testing, the model will be used to simulate tournament outcomes and determine the optimal bracket.

A basic Exploratory Data Analysis (EDA) will justify the data selected. The EDA will evaluate the variables in question, transforming the data, visualizing the data, and developing insights from the results. The data is detailed and readily available to be used to make accurate predictions. The data can be cleaned and checked for outliers before proceeding. The data can then be transformed, analyzed, and grouped into categories and spread between their offensive and defensive qualities. These can then be graphed and visualized to show these patterns and trends. From these trends, the results can be reviewed and manipulated to make selections of the upcoming March Madness games.

During Part 1 of the project, which was completed by February 13th, a GitHub repository was created and organized with the desired format. The team then decided data from college basketball would be used to populate an optimized predictive March Madness Bracket.

Part 2 of the project will include cleaning the data set with the learning models used in class in addition to finding patterns in the data set. This will be done using the clustering/association rule within the data. Within the coded data, annotations will be included that summarize the findings and explain what the code within the Jupyter Notebook is doing. This information will be summarized, reported and completed by February 28th although this information is to be included in the final report due mid-March.

The project will be completed by March 18th with a final report of the findings and data set including Part 2 and 3. Within Part 3, two models will be developed then adjusted with hyperparameter tuning to determine the effectiveness of each model. The final model will be tested during the SEC Tournament and March Madness.

**Unsupervised Learning Summary**

Methods

Dimensionality Reduction (DR) Techniques:

Five DR methods were tested:

1. Principal Component Analysis (PCA)
2. t-Distributed Stochastic Neighbor Embedding (t-SNE)
3. Uniform Manifold Approximation and Projection (UMAP)
4. Locally Linear Embedding (LLE)
5. Independent Component Analysis (ICA)

Clustering Techniques:

Three clustering methods were applied:

1. K-Means
2. Agglomerative Clustering
3. DBSCAN

Results and Interpretations

PCA + K-Means

* PCA reduced the dataset to two principal components, capturing most variance.
* K-Means assigned teams into four clusters, suggesting distinct styles of play based on team performance.
* The clustering boundary shows clear differentiation, with some overlap, indicating some teams have hybrid playing styles.

t-SNE + Agglomerative Clustering

* t-SNE effectively spread data points, revealing nonlinear relationships.
* Agglomerative clustering identified four clusters, potentially reflecting different playstyles (e.g., fast-paced vs. defensive-oriented teams).
* Clusters are more evenly distributed compared to PCA, suggesting better separation.

UMAP + DBSCAN

* UMAP retained both local and global structure, showing more organic groupings.
* DBSCAN primarily grouped most teams into a single cluster (label 0) with a few noise points (-1), indicating that many teams share similar performance metrics, with only a few outliers.
* This result suggests DBSCAN may not be optimal for this dataset, as basketball team performance data may not have distinct density-based clusters.

Cluster Distributions

* DBSCAN: Most teams fell into one cluster, with only a few outliers.
* K-Means: Produced four well-balanced clusters, suggesting meaningful segmentation.
* Agglomerative: Showed slightly uneven distributions but still reasonable separation.

Conclusion

* Best DR-Clustering Pairing: t-SNE with Agglomerative Clustering provided the most distinct and interpretable clusters.
* PCA & K-Means performed well but had overlapping regions.
* UMAP & DBSCAN struggled to separate teams, suggesting DBSCAN may not be ideal for this dataset.
* The clusters likely correspond to different team strategies, such as offensive vs. defensive dominance, high vs. low tempo, or balanced vs. extreme playstyles.

**Supervised Learning (Regression) Summary**

Performance Comparison

Here are the evaluation results for each model:

1. Random Forest
   * Accuracy: 0.66
   * AUC: 0.64
   * Random Forest struggled with predicting the "Away Win" (class 1), showing a lower recall for this class.
2. Logistic Regression
   * Accuracy: 0.70
   * AUC: 0.68
   * Logistic Regression provided a good balance between precision and recall but had difficulty correctly identifying "Away Win" events.
3. Gradient Boosting
   * Accuracy: 0.69
   * AUC: 0.67
   * Gradient Boosting performed similarly to Random Forest but showed a slightly better recall for the "Away Win" class.

Cross-Validation Results

The cross-validation results showed the following average accuracy across 5 folds:

* Random Forest: 0.69
* Logistic Regression: 0.73
* Gradient Boosting: 0.70

Conclusion

Best Model: Logistic Regression. Achieved the highest cross-validation accuracy (0.73) and had the best overall performance in terms of accuracy and stability. It also showed balanced performance across both classes, though there is room for improvement in predicting "Away Win."

**Model Performance Analysis**

The regression model was used to determine the win probability of the matchups that were played during the SEC Tournament. Out of the fifteen matchups that were played, the model predicted thirteen correctly and accurately identified the eventual winner before the first game was played. The model accurately predicted two of the four upsets that occurred during the tournament.

The next performance test for the model will be identifying upsets in the march madness bracket. At the time of submission, the model has identified six potential upsets for the first round of the bracket:

-Baylor over Mississippi St.

-VCU over BYU

-UC San Diego over Texas A&M

-New Mexico over Michigan St.

-Oklahoma over UConn

-Colorado St. over Memphis

At the time of submission, a completed bracket has not been completed as there are play-in games still occurring that we need to finish to rule out any other potential upsets. However, our optimized bracket will be sent to Kyle Territo as a secondary submission once the final play-in has finished (3/19).