# Weekly Report (29/03/25)

## Literature Survey

| Title | Author | Year | Summary |
|---|---|---|---|
| A Unified Machine Learning Framework for Basketball Team Roster Construction: NBA and WNBA | Yuhao Ke, Ranran Bian, Rohitash Chandra | 2024 | This study introduces a framework combining unsupervised and supervised machine learning models for basketball team roster construction. Utilising PCA and clustering techniques, the framework identifies 10 distinct player clusters, with 4 classified as elite. A neural network is then employed to determine the optimal combination of player types, aiming to enhance team synergy. |
| Enhancing Basketball Game Outcome Prediction through Fused Graph Convolutional Networks and Random Forest Algorithm | Kai Zhao, Chunjie Du, Guangxin Tan | 2023 | The proposed approach applies graph convolutional networks (GCNs) combined with the random forest algorithm to predict basketball game outcomes. By transforming structured data into unstructured graphs representing interactions between teams, the study captures the spatial structure and dynamics of the league. |

## Win Prediction using XGBoost

**Methodology**: Iterative Time-Series Approach for Game Win Prediction

To predict game outcomes using team and opponent statistics, we propose an iterative time-series approach. The primary objective is to leverage player-level performance data, aggregated at the team level, to capture how team dynamics evolve over time. Instead of simply averaging or taking the median of player statistics, we normalize individual metrics on a per-minute basis and adjust for the specific lineup in each game. This ensures that player influence is proportional to their playing time, mitigating the overrepresentation of high-minute players.

Input Features

We focus on 10 key performance metrics derived from player statistics to represent overall team strength:

- Points Scored (PTS)
- Field Goals Made (FGM)
- Field Goals Attempted (FGA)
- Offensive Rebounds (OREB)
- Defensive Rebounds (DREB)
- Assists (AST)
- Steals (STL)
- Blocks (BLK)
- Turnovers (TO)
- Personal Fouls (PF)

Each player's statistics are converted to per-minute metrics, calculated as:

Stat per min= Total Stat/ Minutes Played

For every game, we sum the per-minute statistics of all players who participated, without normalizing by the number of players. This accounts for the contribution of each lineup while preserving the impact of high-performing individuals.

**Iterative Approach for Time-Series Dependency**

Basketball games between the same two teams often happen more than once within a season. To model the evolving nature of team performance and incorporate time-series dependency:

**Initial Prediction:**

For the first encounter between two teams (say Team A vs. Team B), we use the aggregated team statistics as input to the XGBoost model.

The target variable is a binary outcome indicating a win (1) or loss (0).

**Sequential Updates:**

For the next encounter between the same two teams, we leverage information from the previous matchup(s).

**The input features include:**

- Aggregated per-minute statistics for the current lineup.
- Historical predictions of wins or losses.
- A “streak” variable indicating the number of consecutive wins or losses.

**Model Iteration:**

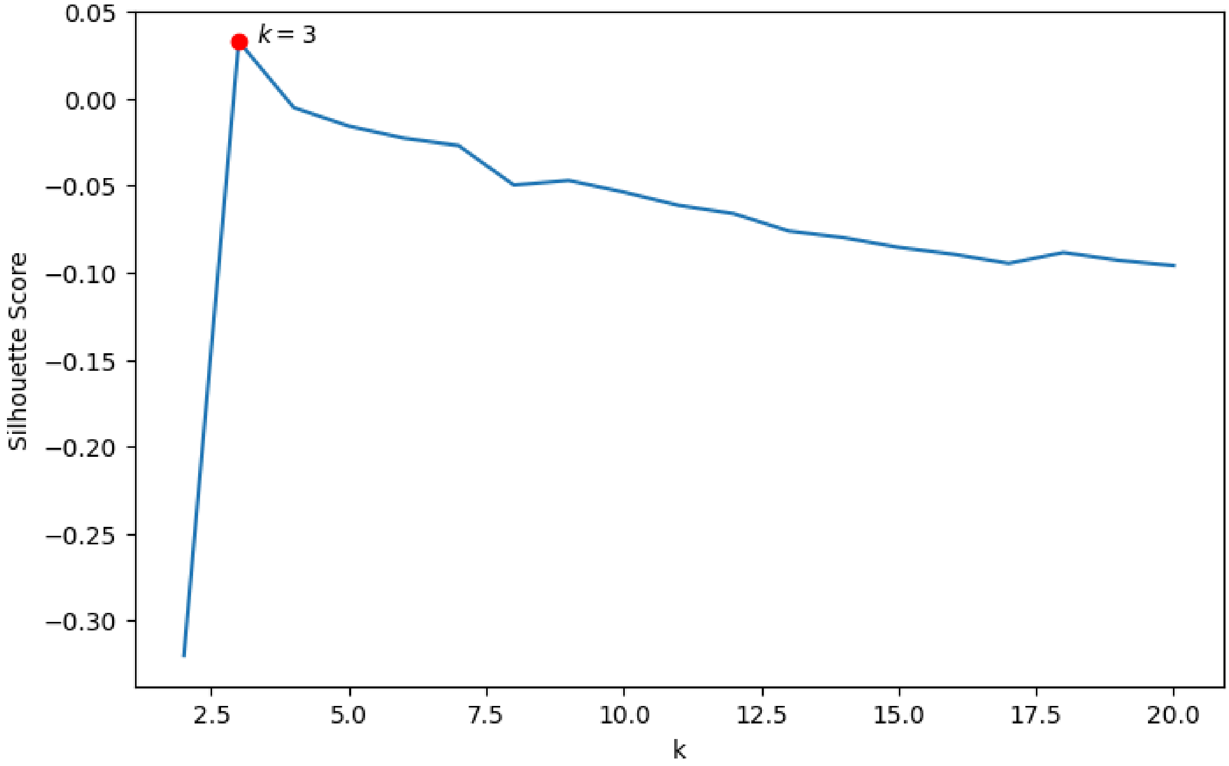
Each subsequent game iteration uses the updated team performance metrics and previous predictions, effectively modeling time-series dependency.
By incorporating historical predictions, the model adapts to changing team dynamics and captures trends over time.

**Rationale**
This method acknowledges that team performance is not static. Players' contributions vary with each lineup, and previous encounters can influence future games. By iteratively updating team metrics with historical context, we aim to improve predictive accuracy. The use of per-minute normalization ensures that key contributors with limited playtime are not overshadowed by high-minute players, while time-series modeling captures evolving dynamics.

## Subspace Clustering

We attempted to use subspace clustering to tackle the problem of finding athlete clusters given our high dimensional data. Subspace clustering has been developed specifically for tackling high dimensional data, and it tries to find clusters in subsets of features, instead of the entire feature set.

We used C. You’s implementation of Elastic Net Subspace Clustering in Python. Using unsupervised clustering metrics, we found k=3 to be the optimal number of clusters via this model. However, we did not observe much improvement compared to K-Means from just observing these metrics. We will need to observe the clusters more carefully for this model for interpretation, as they are more complicated than regular centroid based clustering. Furthermore, since our data lacks any ground truth data for individual athletes, we cannot perform any supervised evaluations of our clusters, making the task time intensive.



**Silhouette Score for different k values using the subspace clustering model**

We will aim to investigate the clusters formed by this model more carefully, as well as explore graph based clustering and the Stochastic Block model as well.