Analysis of NBA Player Stats

Darshan Singh
University of the Witwatersrand
2441307

1 Data Cleaning

Missing values were investigated for things like the Attempts and whether it logically matches the % (e.g., if 3PA was 0, it would make sense for the percentage to also be 0). There were no NaNs so leaving those values at 0 is fine.

Players with very few games played (e.g., < 10) can be considered “odd" as their “per game" stats are not representative. Thus, they were removed. 62 players had < 10 games played, leaving 405 remaining 1.

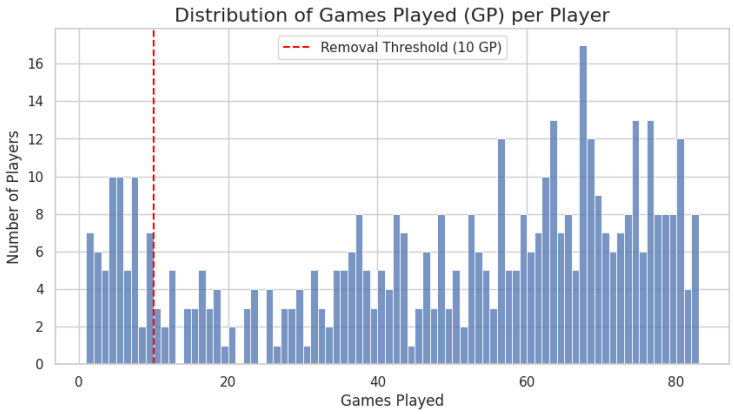

Figure 1: Distribution of Games Played (GP) per Player. The red dashed line indicates the removal threshold at 10 games.

For feature preparation all numeric features were scaled using `StandardScaler` and categorical features were one-hot encoded. Scaling was necessary for k-Means and Neural nets like the Autoencoders to prevent large magnitude features (like ‘Salary’) dominating the model.

2 Dimensionality Reduction

Q2.1(a) An unsupervised neural network that learns a compressed latent representation (encoding) by trying to reconstruct its own input. Trained by minimising reconstruction error (MSE).

For the hybrid techniques below, the autoencoder was first used as a non-linear feature extractor. The `X_processed.shape[1]`– dimensional data was fed into the trained encoder, and the output of its final 32D hidden layer was extracted. This dense layer representation captures the most salient features of the data. These were then used as the input for the SOM, t-SNE and UMAP.

38th Conference on Neural Information Processing Systems (NeurIPS 2024) Track on Datasets and Benchmarks.

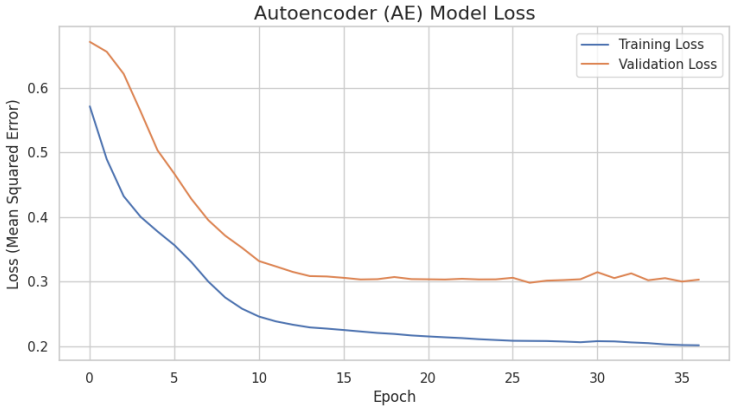

Figure 2: Autoencoder (AE) training and validation loss (MSE).

Q2.1(b) The 32D data from the autoencoder was used to train the 20x20 SOM. Each of the 405 players was then mapped to their 'Best Matching Unit' (BMU) on the grid. The final 2D representation for each player consists of the (x, y) coordinates of their winning neuron (see Fig. 5).

Q2.1(c) The T-SNE algorithm was applied to the 32D latent space from the AE. A standard `perplexity=30` was used to generate the final 2D representations.

Q2.1(d) UMAP was applied to the 32-dimensional AE latent space, using standard parameters (`n_neighbors=15`, `min_dist=0.1`) to produce the final 2D representation.

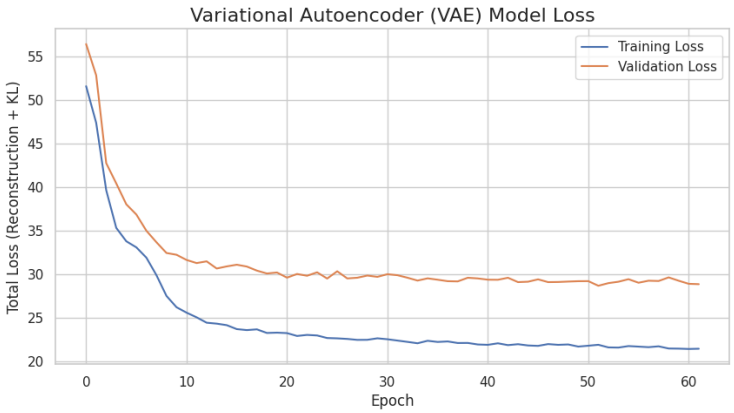

Figure 3: Variational Autoencoder (VAE) training and validation loss (Reconstruction + KL Divergence).

Q2.1(e) The 84-dimensional input was encoded into a 64-dimensional layer, which then outputted the parameters (`z_mean` and `z_log_var`) for a 2-dimensional latent distribution. The model was trained using a custom loss function, which is the sum of the **reconstruction loss** (MSE) and the **Kullback-Leibler (KL) divergence**. The KL divergence acts as a regulariser, pushing the latent distributions to resemble a standard normal distribution, which encourages in a smoother and more continuous latent space. The loss plot Fig. 3 shows steady convergence for both training and validation, indicating a successful training process. To get a stable 2D representation for plotting, the mean (z_mean) of the learned distributions was taken for each player, resulting in the final (405, 2) matrix.

2.1 Question 2.2

This plot was done to generate an observation of what the appropriate number of clusters for k-Means should be. The AE+UMAP representation was plotted for a range of `k` values (from 2 to 10).

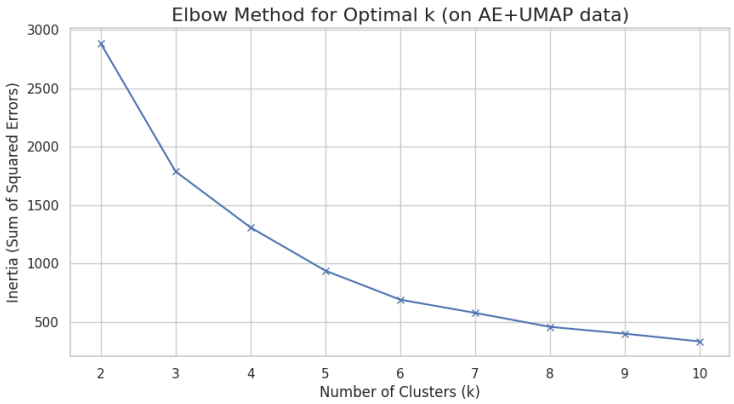

Figure 4: Elbow Method plot using AE+UMAP data, showing inertia vs. number of clusters (k). The 'elbow' is observed at k=4.

As seen in Fig. 4, there is a distinct rate of decrease in inertia at the 'elbow' `k=4` . So any clusters beyond four provides diminishing returns. So the optimal number of clusters was then selected to be 4. As will be seen below it should capture the primary groupings within the data. AE/VAE: The

Figure 5: Comparison of k-Means (k=4) clustering results applied to the 2D representations from each dimensionality reduction technique.

clusters produced by the standard Autoencoder and the VAE show significant overlap. The k-Means algorithm struggles to find clear boundaries, resulting in poorly defined, spherical groups that do not appear to represent distinct player types.

AE + SOM: The Self-Organising Map maps players to a rigid grid structure. While this organises the data, the resulting clusters are 'blocky' and artificial, reflecting the grid itself rather than natural groupings within the data.

AE + t-SNE/AE + UMAP: In stark contrast, the AE+t-SNE and AE+UMAP methods yield a dramatically different result. The clusters are dense, non-spherical, and well-separated. This visual evidence strongly suggests that these non-linear techniques, when combined with an autoencoder for feature extraction, are far superior at identifying the underlying groupings in the player data.

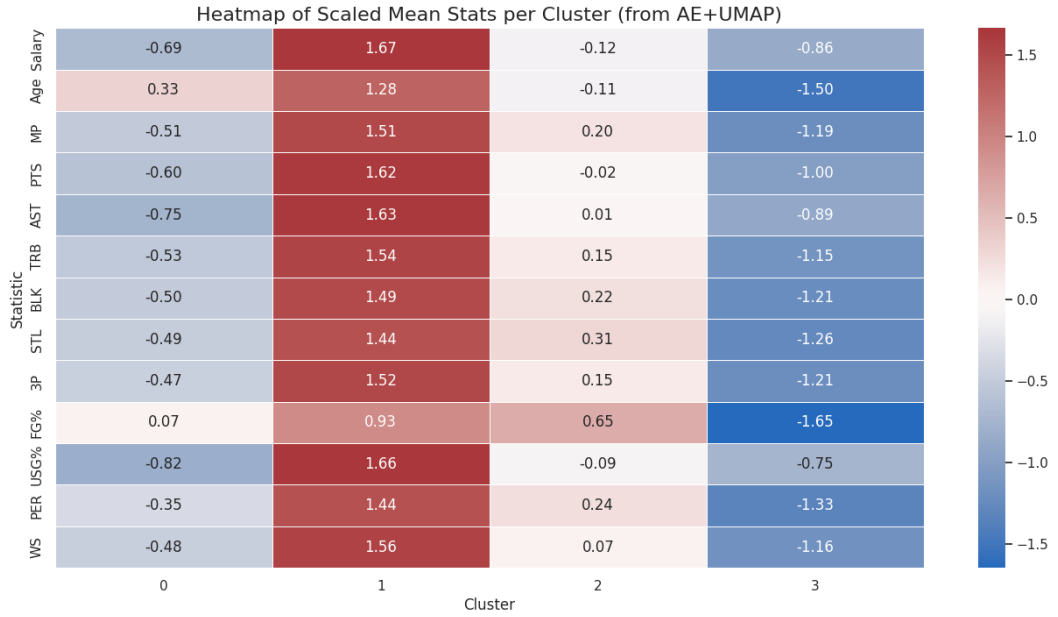

Figure 6: Heatmap of scaled mean statistics per cluster derived from the AE+UMAP representation. Values indicate standard deviations from the overall mean.

Table 1: Position Distribution per Cluster (AE+UMAP)

Cluster_UMAP	C	PF	PG	SF	SG
0	16	35	8	31	31
1	22	15	20	11	15
2	42	15	21	18	34
3	3	11	17	19	21

As seen in Fig. 5 AE + UMAP provides the most insightful clustering which is why it was chosen for the above heatmap Fig. 6 . Based on the clustering it is quite clear which persona each cluster represents.

Cluster 0 (Low-Usage Veterans / Bench Roles): This cluster shows slightly above-average `Age` (+0.33), suggesting experience. However, all key performance metrics (`MP`, `PTS`, `AST`, `TRB`, `BLK`, `STL`, `3P`, `PER`, `WS`) and contribution metrics (`Salary`, `USG%`) are moderately below average (mostly between -0.35 and -0.82 standard deviations). Their `FG%` (+0.07) is average. This profile points towards veteran players who fill bench or limited roles, providing experience but not significant statistical production.

Cluster 1 (Elite All-Around Stars): This group stands out dramatically, showing significantly above-average values (often > +1.4 standard deviations) across almost all categories. This clearly identifies the league's established stars and highest impact players, who excel in scoring, playmaking, contribute across the board, command high salaries, and

Cluster 2 (Efficient Finishers / Mid-Tier Roles): Players in this cluster are close to the league average in terms of `Age` (-0.11), `Salary` (-0.12), usage (`USG%` -0.09), and overall production (`PTS` -0.02, `AST` +0.01, `TRB` +0.15, `WS` +0.07). Their defining characteristic is notably efficient scoring, reflected in the high positive deviation for `FG%` (+0.65). They also play slightly above-average minutes (`MP` +0.20) and contribute positively to efficiency metrics (`PER` +0.24) and defence (`BLK` +0.22, `STL` +0.31). This suggests players who primarily score efficiently near the basket or fill specific roles effectively without being high-usage stars.

Cluster 3 (Young Developing Players / Deep Bench): This cluster is clearly defined by its significantly below-average `Age` (-1.50). Correspondingly, all performance and contribution metrics are substantially below the mean (mostly between -0.75 and -1.65 standard deviations). This profile

strongly indicates young players still developing or players at the end of the bench with minimal playing time and statistical impact.

3 Conclusion

As demonstrated by the comparative scatter plots, the standard AE and VAE produced overlapping and indistinct clusters. The AE+SOM's output was constrained by its grid structure. In contrast, the AE+UMAP technique produced dense, well-separated clusters that, upon inspection, corresponded to distinct and meaningful player archetypes. This demonstrates that using an autoencoder for non-linear feature extraction, followed by a manifold learning technique like UMAP for the final reduction, is a highly effective strategy for uncovering the complex, underlying structure within player data.

References

[1] Basketball-Reference.com. (2024). "2023-24 NBA Advanced Stats". Retrieved October 22, 2025, from `https://www.basketball-reference.com/leagues/NBA_2024_advanced.html`