A Methodological Framework for Latent Role Discovery and Market Inefficiency Analysis

Your Team Name

October 18, 2025

Abstract

This document outlines a revised, multi-phase methodological framework for our capstone project. Moving away from the direct analysis of systematic bias and its associated confounding problems , this new approach leverages unsupervised learning to first discover latent player roles. These data-driven roles then serve as the foundation for a robust, multi-stage analysis to (1) interpret the new roles, (2) determine their market value, (3) identify player-level inefficiencies, and (4) test for systematic biases in the remaining salary unexplained by performance or role.

1 Phase 1: Discovering Latent Player Roles

The project's foundation is the hypothesis that pre-defined positions (e.g., "Center") are insufficient. We aim to discover a player's "true" role based on their on-court performance data.

1.1 The Feature Matrix X

Let our dataset consist of n players. For each player i, we define a feature vector $\mathbf{x}_i \in \mathbb{R}^p$, where p is the number of performance metrics.

$$\mathbf{X} = \begin{bmatrix} --- & \mathbf{x}_1 & --- \\ --- & \mathbf{x}_2 & --- \\ & \vdots & \\ --- & \mathbf{x}_n & --- \end{bmatrix}$$

The features in $\mathbf{x}_i$ will be rate-based and position-agnostic (e.g., 'True Shooting %', 'Rebound Rate', 'Assist %') to ensure a fair comparison of player style and efficiency . Official position labels will be explicitly excluded from $\mathbf{X}$.

1.2 Standardization

As clustering algorithms are distance-based, we will first standardize $\mathbf{X}$ to create $\mathbf{X}'$, where each feature (column) has a mean of 0 and a standard deviation of 1.

$$x'_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$$

where μ_j and σ_j are the mean and standard deviation of feature j.

1.3 Clustering Model

We will partition the n players into K distinct roles using a suitable clustering algorithm based on their standardized feature vectors $\mathbf{x}'_i$. The primary candidates for this are **K-Means** and **Gaussian Mixture Models (GMMs)**.

- **K-Means Clustering** provides "hard" assignments by partitioning the data to minimize the within-cluster sum of squares (WCSS). It is a distance-based approach where each player is assigned to exactly one role. The objective function is:

$$\arg\min_{S} \sum_{k=1}^{K} \sum_{\mathbf{x}'_i \in S_k} ||\mathbf{x}'_i - \boldsymbol{\mu}_k||^2$$

  where S_k is the set of players in cluster k and $\boldsymbol{\mu}_k$ is the mean (centroid) of that cluster.

- **Gaussian Mixture Models (GMMs)** are a probabilistic, model-based alternative. A GMM assumes the data is generated from a mixture of K different Gaussian distributions (our roles). It provides "soft" assignments, giving the probability $P(k|\mathbf{x}'_i)$ that player i belongs to each role k. This may be advantageous for representing players who are hybrids of two or more roles.

We will evaluate both methods based on cluster interpretability and separation. For the final analysis, we will use the "hard" assignment from K-Means or the role with the highest probability from the GMM. The final output will be a mapping $C : i \to k$, assigning each player i to their primary latent role k.

2 Phase 2: Role Interpretation and Validation

Before analysis, we must interpret and label the K discovered roles. We will use the traditional position labels (which we previously excluded) as a validation tool.

2.1 Labeling Latent Roles

We will construct a contingency table (or heatmap) to cross-reference our latent roles $k \in \{1, ..., K\}$ against the official position labels Z_{pos} (e.g., 'Guard', 'Forward', 'Center'). This allows us to understand the composition of each cluster. For example, a cluster that is 90% 'Centers' can be confidently labeled "Traditional Big," while a cluster that is 50% 'Shooting Guards' and 50% 'Small Forwards' might be labeled "3-and-D Wing."

3 Phase 3: Salary Analysis

We now introduce the salary variable, Y_i, for each player. We will likely use its natural logarithm, $\log(Y_i)$, to account for the skewed distribution of salaries.

3.1 Market Valuation of Roles

Our first analysis is a direct valuation of the discovered roles. For each role $k \in \{1, ..., K\}$, we will compute a measure of central tendency for salary:

$$\text{Median Value}(k) = \text{Median}\left(\log(Y_i) \mid C(i) = k\right)$$

This answers the question: "What is the median market price for this specific skillset?"

3.2 Identifying Market Inefficiencies

To find over/under-valued players, we will build K separate regression models—one for each role. For a given role k, we model:

$$\log(Y_i) = f_k(\mathbf{x}_i) + \epsilon_{i,k} \quad \forall i \text{ such that } C(i) = k$$

Here, f_k is a model (e.g., linear regression) trained *only* on the players belonging to role k, and $\mathbf{x}_i$ is the original performance feature vector. The residual for each player is:

$$\hat{\epsilon}_{i,k} = \log(Y_i) - \hat{f}_k(\mathbf{x}_i)$$

This residual $\hat{\epsilon}_{i,k}$ is our **Performance-and-Role-Adjusted Disparity**. A player with a large negative residual ($\hat{\epsilon}_{i,k} \ll 0$) is a ”bargain” (market inefficiency). This residual becomes the outcome variable for our final phase.

4 Phase 4: Explaining Market Inefficiencies

This final phase revisits our original project goal: detecting systematic market bias . We now have a robust metric for ”mispricing” ($\hat{\epsilon}$) that controls for both performance (X) and role (C). We now test if this mispricing is systematically correlated with non-performance, contextual factors.

4.1 The Contextual Matrix Z

We define a new matrix $\mathbf{Z}$ of m contextual variables (the same variables we originally wished to test for bias).

$$\mathbf{z}_i = [z_{i,1}, z_{i,2}, ..., z_{i,m}]$$

These variables include factors like ’Age’, ’Nationality’, ’Draft Status’, ’Team Market Size’, etc. Note that this Z matrix is separate from the X performance matrix.

4.2 Modeling the Disparity

We will build a single, interpretable linear model where the outcome variable is the residual $\hat{\epsilon}_i$ (pooled from all K models) and the predictors are the contextual variables in $\mathbf{z}_i$:

$$\hat{\epsilon}_i = g(\mathbf{z}_i) + \nu_i$$

Which can be expressed as:

$$\hat{\epsilon}_i = \beta_0 + \beta_1 z_{i,1} + \beta_2 z_{i,2} + \cdots + \beta_m z_{i,m} + \nu_i$$

The coefficients β_j from this model provide a direct estimate of systematic bias. For example, a statistically significant, positive coefficient $\beta_{\text{Draft\_Status}}$ would imply that higher draft status leads to a player being paid *more* than their performance and role would otherwise warrant.

4.3 Justification of the Method

This multi-stage framework is statistically sound and avoids the two primary issues identified in our initial proposal.

1. **It is not circular.** The original flaw was in modeling $log(Y) = f(X, Z)$ and then testing the residuals against Z. Our new method *never* includes the contextual variables $\mathbf{z}_i$ in the salary model f_k. The residual $\hat{\epsilon}_i$ is therefore not, by construction, uncorrelated with $\mathbf{z}_i$. We are performing a valid, two-stage regression.

2. **It mitigates the X-Z confounding.** The second, more difficult problem was the confounding between performance (X) and traditional positions (Z_{pos}) . A simple model $log(Y) = f(X)$ would produce biased residuals, as X itself is correlated with position. Our method solves this by:

   - (a) Using clustering (Phase 1) to create new latent roles (C) that are defined *by* performance, effectively capturing the complex interaction between X and Z_{pos}.
   - (b) Building role-specific models (Phase 3) $f_k(\mathbf{x}_i)$. This is a sophisticated, non-linear way of controlling for performance, as it compares a player's salary *only* against peers with a similar skillset.

Because our residual $\hat{\epsilon}_i$ has been ”cleaned” of the influence of both performance (X) and role (C), we can now validly test if this remaining ”mispricing” is systematically explained by $\mathbf{z}_i$.