Mathematical Explanation of the Player Selection Program

Introduction

This document provides a mathematical explanation of how the player selection program works. The program aims to select a subset of players from a larger dataset to maximize a total objective score based on specified criteria while satisfying certain constraints. This problem is formulated as a **Binary Integer Linear Programming (BILP)** problem.

Decision Variables

We define binary decision variables to represent whether a player is selected:

Let:

- n be the total number of players in the dataset.
- For each player i (where $i = 1, 2, \ldots, n$), we define a binary variable:

$$x_i = \begin{cases} 1, & \text{if player } i \text{ is selected} \\ 0, & \text{otherwise} \end{cases}$$

Objective Function

The objective is to maximize the total team score based on the selected criteria and their corresponding weights.

Let:

- m be the number of criteria selected.
- For each criterion j (where $j = 1, 2, \ldots, m$):
  - c_{ij} be the value of criterion j for player i.
  - w_j be the weight assigned to criterion j.

The individual score for player i is calculated as:

$$\text{Score}_i = \sum_{j=1}^{m} w_j \times c_{ij}$$

The total team score is the sum of individual scores of the selected players:

$$\text{Total Team Score} = \sum_{i=1}^{n} x_i \left(\sum_{j=1}^{m} w_j c_{ij} \right)$$

Objective Function:

$$\max_{x_i} \quad \sum_{i=1}^{n} x_i \left(\sum_{j=1}^{m} w_j c_{ij} \right)$$

Constraints

Constraints are conditions that the selected players must collectively satisfy. They can be inequalities or equalities involving player attributes.

For each constraint k:

Let:

- a_{ik} be the value of the attribute involved in constraint k for player i.
- b_k be the value specified in the constraint.

The constraint can be one of the following types:

Less Than or Equal To ($\leq$)

$$\sum_{i=1}^{n} x_i a_{ik} \leq b_k$$

Greater Than or Equal To ($\geq$)

$$\sum_{i=1}^{n} x_i a_{ik} \geq b_k$$

Equal To ($=$)

$$\sum_{i=1}^{n} x_i a_{ik} = b_k$$

Decision Variable Bounds

Each decision variable x_i is binary:

$$x_i \in \{0, 1\}, \quad \forall i = 1, 2, \ldots, n$$

Complete Mathematical Model

Putting it all together, the optimization problem is:

Maximize:

$$\max_{x_i} \quad \sum_{i=1}^{n} x_i \left(\sum_{j=1}^{m} w_j c_{ij} \right)$$

Subject to:

For each constraint k:

- If the constraint is of type $\leq$:

$$\sum_{i=1}^{n} x_i a_{ik} \leq b_k$$

- If the constraint is of type $\geq$:

$$\sum_{i=1}^{n} x_i a_{ik} \geq b_k$$

- If the constraint is of type $=$:

$$\sum_{i=1}^{n} x_i a_{ik} = b_k$$

Decision Variables:

$$x_i \in \{0,1\}, \quad \forall i = 1, 2, \ldots, n$$

Explanation with an Example

Suppose we have:

Criteria

- Goals (`goals`) with weight $w_1 = 5$.
- Interceptions (`interceptions`) with weight $w_2 = 2$.
- Fouls (`fouls`) with weight $w_3 = -3$ (negative because fouls are undesirable).

Constraints

- Total appearances (`appearances`) must be at least 10:

$$\sum_{i=1}^{n} x_i \times \text{appearances}_i \geq 10$$

- Red cards (`red_cards`) must be at most 1:

$$\sum_{i=1}^{n} x_i \times \text{red\_cards}_i \leq 1$$

- Clean sheets (`clean_sheets`) must be at least 4:

$$\sum_{i=1}^{n} x_i \times \text{clean\_sheets}_i \geq 4$$

Objective Function

$$\max_{x_i} \quad \sum_{i=1}^{n} x_i \left(5 \times \text{goals}_i + 2 \times \text{interceptions}_i - 3 \times \text{fouls}_i\right)$$

Constraints

$$\begin{cases} \sum_{i=1}^{n} x_i \times \text{appearances}_i \geq 10 \\ \sum_{i=1}^{n} x_i \times \text{red\_cards}_i \leq 1 \\ \sum_{i=1}^{n} x_i \times \text{clean\_sheets}_i \geq 4 \\ x_i \in \{0,1\}, \quad \forall i = 1, 2, \ldots, n \end{cases}$$

Interpretation

- **Decision Variables (x_i)**: Determine which players are included in the team.
- **Objective Function**: Calculates the total score of the team based on the selected criteria and their weights. The goal is to maximize this total score.
- **Constraints**: Ensure that the selected team meets specific requirements (e.g., minimum number of appearances, maximum number of red cards).
- **Solution**: The optimization solver finds the combination of players (values of x_i) that maximizes the objective function while satisfying all the constraints.

Mathematical Steps in the Program

1. **Input Collection**:
   - Collect weights w_j for each selected criterion j.
   - Collect constraint values b_k and types for each constraint k.
2. **Data Preparation**:
   - Prepare the data matrix c_{ij} containing players' statistics for each criterion.
   - Prepare the attribute matrix a_{ik} for constraints.
3. **Model Formulation**:
   - Formulate the objective function and constraints as shown above.
4. **Optimization**:
   - Use an optimization solver (e.g., PuLP with CBC) to solve the BILP problem.
5. **Result Extraction**:
   - Retrieve the values of x_i to determine which players are selected.
   - Calculate individual scores for each selected player:

$$\text{Score}_i = \sum_{j=1}^{m} w_j \times c_{ij}$$

Important Notes

- **Binary Variables**:
  - The decision variables are binary because a player can either be selected ($x_i = 1$) or not ($x_i = 0$).
- **Linear Programming**:
  - The problem is linear because both the objective function and constraints are linear functions of the decision variables.
- **Integer Programming**:
  - Since decision variables are integers (specifically 0 or 1), it is an integer programming problem.
- **Constraints Types**:
  - Constraints can be equality ($=$) or inequality ($\leq$ or $\geq$), depending on the user's requirements.

Summary

The program solves the following optimization problem:

Maximize:

$$\text{Total Team Score} = \sum_{i=1}^{n} x_i \left(\sum_{j=1}^{m} w_j c_{ij} \right)$$

Subject to:

$$\begin{cases} \sum_{i=1}^{n} x_i a_{ik} \leq / = / \geq \ b_k, & \text{for each constraint } k \\ x_i \in \{0,1\}, & \forall i = 1, 2, \ldots, n \end{cases}$$