Filleule Quentin

**Projet PVE**

**Estimation de la valeur des**

**Joueurs de foot**

Une image contenant Graphique, clipart, dessin, cercle

Description générée automatiquement

2023/2024

**Présentation du projet PVE :**

PVE pour Player Value Estimator est un projet révolutionnaire qui combine la puissance de l'intelligence artificielle avec une expertise approfondie du monde du football pour offrir une solution d'estimation de la valeur des joueurs.

Conçu spécifiquement pour répondre aux besoins des clubs et des agents de joueurs, ce système utilise les statistiques clés, et d'autres facteurs influents. En résulte une évaluation précise de la valeur d'un joueur, fournissant ainsi un outil indispensable pour guider les décisions stratégiques en matière de recrutement mais pas uniquement. En effet le fait d’avoir un outil impartial et fiable permet également d’obtenir des arguments pour de potentielles négociations.

**Spécification technique du projet :**

Le projet est basé autour du dataset Masood Ahmed : Football Players Data, il comprend des données sur plus de 17 000 joueurs avec une cinquantaine de caractéristiques pour chaque. La première étape du projet fut donc de l’exploration et le traitement de ces données.

**Exploration et traitement des données :**

Apres importation du dataset, la première tâche à réaliser est de récupérer la liste et le type de toutes les données pour en comprendre la structure, vous trouverez les détails en annexe. Ensuite, on s’occupe de traiter les données pour qu’elles soient utilisable par notre futur algorithme et qu’il n’y est pas de problème. Cela peut être la cas l’lorsqu’il manque des informations, on peut alors remplacer cette information de façons cohérentes quand il manque peut.

Il est possible remplacé la valeur et le salaire de certains joueurs par la médiane de toutes les caractéristiques par exemple. De cette façon nous pouvons utiliser ces données tout en conservant une certaine cohérence. Dans le cas de joueurs qui n’avaient pas de clause de libération de contrat, ces valeurs ont été remplacé par la valeur 0.

Cependant quand il manque une donnée sur un grand nombre de joueurs, il est plus judicieux de l’ignorer et de ne pas l’utiliser pour éviter des erreurs d’interprétation. C’est le cas pour les informations concernant la sélection nationale du joueur. En effet peut d’entre eux a déjà eu une sélection, il est donc préférable de ne pas prendre en compte ces données.

Il est également nécessaire de remplacer certaines données pour qu’elles soient plus compatibles avec le modèle, c’est souvent le cas pour les données de type objet. Dans notre cas il y en a plusieurs mais seulement une seule est susceptible de nous intéresser. Pour pouvoir l’utiliser, nous allons procéder à un encodage et nous allons remplacer toutes les positions par un numéro. Le poste de défenseur central (center back ou CB) devient le poste 1 et ainsi de suite.

**Pertinence des données : Matrice de corrélation**

Une matrice de corrélation est utilisée pour évaluer la pertinence des données en identifiant les relations linéaires entre différentes variables. En particulier, elle mesure la force et la direction des corrélations entre les caractéristiques du jeu de données. Dans notre cas, nous allons évidemment déterminer la corrélation en la colonne value\_euro et les autres. Cette analyse permet alors de détecter des relations fortes entre des variables.

Pour des raisons pratiques vous retrouvez l’image en pièce jointe puisqu’il s’agit d’une matrice de 52x52 qui est donc impossible à afficher d’une façon convenable. Voici cependant une liste des éléments importants que trouvés lors de l’analyse :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Figure - Paramètres avec une corrélation de 0.4 ou plus

Le seuil de 0.4 est arbitraire mais permet de bien séparer les données en catégories pour l’entrainement du modèle. Cette première catégorie (figure 1) sera représentée tel quel pour l’apprentissage. Au contraire de la seconde catégorie (figure 2) qui sera elle utiliser en fusionnant les éléments pour leur donner plus de poids.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Figure - Paramètres avec une corrélation de 0.2 à 0.4

**Feature Engineering :**

Le "Feature Engineering" est le processus de création, de sélection et de transformation de variables dans un ensemble de données afin d'améliorer les performances des modèles d'apprentissage automatique. L'objectif est d'extraire des informations pertinentes des données brutes ou de représenter les caractéristiques de manière à ce qu'elles soient plus efficaces pour le modèle.

Dans notre cas la sélection est déjà effectuée, nous allons alors nous pencher sur le processus de création de nouvelles variables, il permet de créer de nouveaux paramètres sur lesquels l’algorithme peut s’entraîner en utilisant les anciens. Un exemple simple dans notre cas est le rapport entre le potentiel du joueur et son niveau actuel, on réutilise alors les données « potential » et « overall\_rating ». Dans ce cas, les deux paramètres utilisés sont dans la catégorie 1, on peut donc supposer qu’ils se suffisent à eux même.

Il est également possible de créer de nouvelles données en combinant plusieurs d’entre elles qui ont un coefficient de corrélation faible. C’est par exemple le cas ici avec un attribut « shoot » qui combine les données suivantes :

« shot\_power » / « long\_shots » / « penalties » / « finishing » / « volleys » / « curve » / « freekick\_accuracy »

**Entrainement du modèle :**

La partie entraînement du modèle se déroule en trois parties, tout d'abord, on définit les caractéristiques d’apprentissage, telles que l'âge, la note globale, le potentiel, et d'autres. On donne également la variable cible à prédire, à savoir la valeur en euros.

Ensuite, le jeu de données est divisé en deux parties distinctes : les caractéristiques (X) et la variable cible (y). Vient alors l'amélioration du modèle, celle-ci se fait en adoptant une approche de forêt aléatoire avec 100 arbres de décision.

Enfin, pour évaluer la performance du modèle de manière robuste, une validation croisée à 10 plis est utilisée, chaque pli est alternativement utilisé comme ensemble de tests et d'entraînement.

Une fois le modèle entraîné, il est nécessaire de connaître ses performances, pour cela il, existe des métriques de performance. Dans notre cas, la racine carrée de l'erreur quadratique moyenne (RMSE) représente l’écart moyen entre l’estimation de la valeur du joueur et sa valeur réelle. Le coefficient de détermination (R²) correspond lui à la précision du modèle.

Ces résultats sont ensuite affichés sur la moyenne des 10 plis. Ce processus permet d'obtenir une évaluation globale de la performance du modèle de régression sur différentes sections du jeu de données.

**Résultats du modèle :**

Dans un premier temps, pour comprendre comment fonctionne le modèle, une analyse des caractéristiques (features) appliqué à l’ensemble de données a été effectué.

Après l’entraînement du modèle de forêt aléatoire sur les features présenté précédemment, l’importance des caractéristiques sont extraites du modèle entraîné. Ces importances sont organisées dans un DataFrame, où chaque ligne représente une caractéristique associée à son poids dans la prédiction :

Une image contenant texte, capture d’écran, ligne

Description générée automatiquement

On remarque que la feature avec le plus d’importance est la release\_clause avec un coefficient à 0.76. Cela confirme les valeurs trouvées avec la matrice de corrélation puisque c’est bien la donnée qui possède le plus gros score et de loin. D’autres données sont moins importantes mais servent à l’affinement des résultats comme lo note du joueur ou son potentiel.

Finalement nous avons une précision à 0.9896 et une erreur moyenne de 555 304 €.

Pour avoir accès au code, je vous propose de vous rendre sur le dépôt GitHub suivant :

https://github.com/Quentin-Filleule/Projet\_IA\_Foot

**Annexe**

**Liste des données du dataset**

* **full\_name**: Full name of the player.
* **name**: Name of the player.
* **birth\_date**: Date of birth of the player.
* **age**: Age of the player.
* **height\_cm**: Player's height in centimeters.
* **weight\_kgs**: Player's weight in kilograms.
* **positions**: Positions the player can play.
* **nationality**: Player's nationality.
* **overall\_rating**: Overall rating of the player in FIFA.
* **potential**: Potential rating of the player in FIFA.
* **value\_euro**: Market value of the player in euros.
* **wage\_euro**: Weekly wage of the player in euros.
* **preferred\_foot**: Player's preferred foot.
* **international\_reputation(1-5)**: International reputation rating from 1 to 5.
* **weak\_foot(1-5)**: Rating of the player's weaker foot from 1 to 5.
* **skill\_moves(1-5)**: Skill moves rating from 1 to 5.
* **release\_clause\_euro**: Release clause of the player in euros.
* **national\_team**: National team of the player.
* **national\_rating**: Rating in the national team.
* **national\_team\_position**: Position in the national team.
* **national\_jersey\_number**: Jersey number in the national team.
* **crossing**: Rating for crossing ability.
* **finishing**: Rating for finishing ability.
* **heading\_accuracy**: Rating for heading accuracy.
* **short\_passing**: Rating for short passing ability.
* **volleys**: Rating for volleys.
* **dribbling**: Rating for dribbling.
* **curve**: Rating for curve shots.
* **freekick\_accuracy**: Rating for free kick accuracy.
* **long\_passing**: Rating for long passing.
* **ball\_control**: Rating for ball control.
* **acceleration**: Rating for acceleration.
* **sprint\_speed**: Rating for sprint speed.
* **agility**: Rating for agility.
* **reactions**: Rating for reactions.
* **balance**: Rating for balance.
* **shot\_power**: Rating for shot power.
* **jumping**: Rating for jumping.
* **stamina**: Rating for stamina.
* **strength**: Rating for strength.
* **long\_shots**: Rating for long shots.
* **aggression**: Rating for aggression.
* **interceptions**: Rating for interceptions.
* **positioning**: Rating for positioning.
* **vision**: Rating for vision.
* **penalties**: Rating for penalties.
* **composure**: Rating for composure.
* **marking**: Rating for marking.
* **standing\_tackle**: Rating for standing tackle.
* **sliding\_tackle**: Rating for sliding tackle.