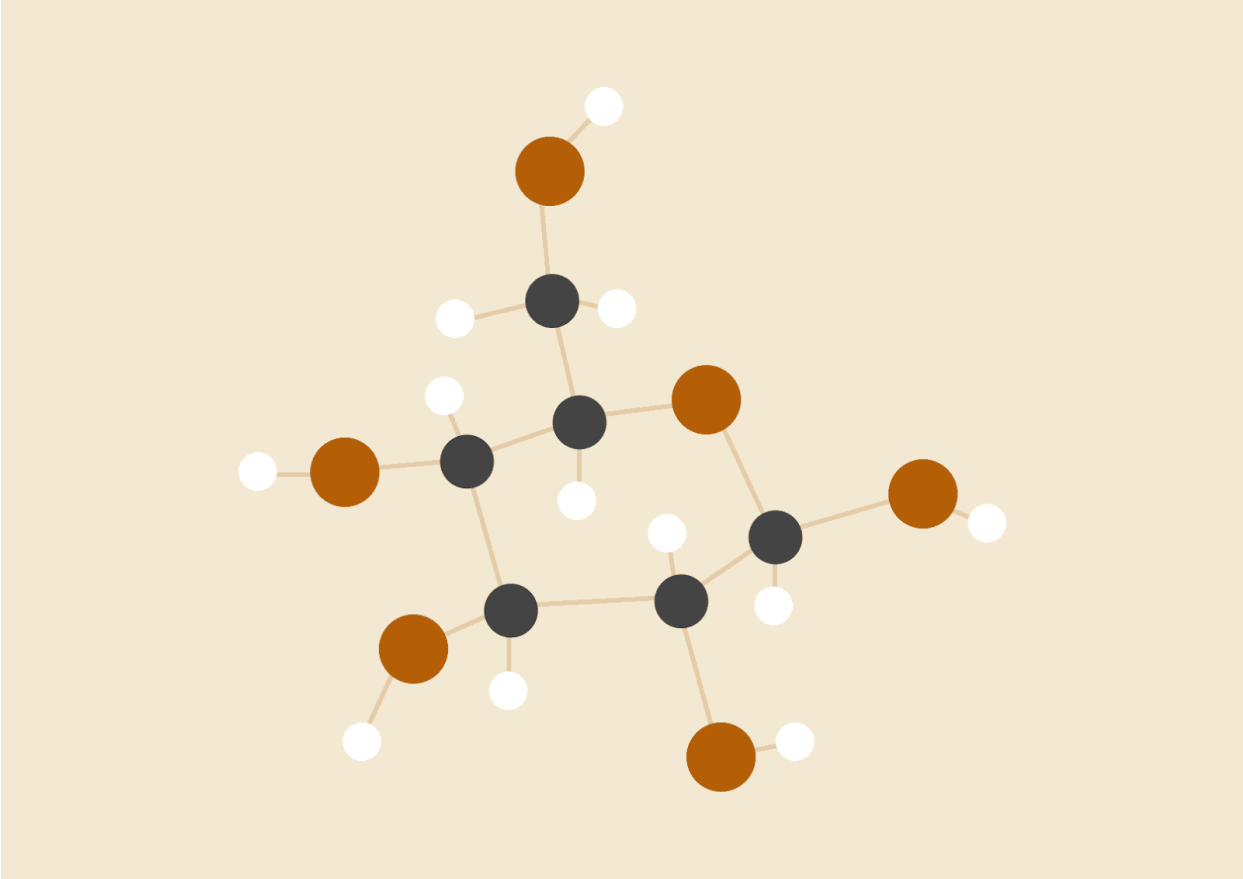
Rapport de projet de prédiction en NBA



Ayoub BOUCHACHIA

18/05/2023

IA– Master 2

INTRODUCTION

Ce projet "Prediction in the NBA" a pour objectif de prédire le vainqueur de la saison de la NBA pour l'année 2022/2023. Le processus de prédiction se divise en trois étapes principales : trouver le vainqueur de la conférence Est, trouver le vainqueur de la conférence Ouest, puis déterminer le champion de la NBA en fonction des deux vainqueurs de conférence.

DATASET

Le dataset récupéré à partir du site "https://www.basketball-reference.com/playoffs/NBA_2023_games.html" contient les informations relatives à l'horaire et aux résultats des matchs des séries éliminatoires de la NBA pour la saison 2022/2023. Le jeu de données comprend un total de 82 lignes, ce qui correspond au nombre de matchs joués pendant les séries éliminatoires.

Les attributs présents dans le dataset sont les suivants :

a. Date : La date à laquelle le match a été joué.
b. Start (ET) : L'heure de début du match (heure de l'Est).
c. Visitor/Neutral : L'équipe visiteuse ou neutre (équipe qui joue à l'extérieur ou dans une arène neutre).
d. PTS1 : Le nombre de points marqués par l'équipe visiteuse ou neutre.
e. Home/Neutral : L'équipe à domicile ou neutre (équipe qui joue à domicile ou dans une arène neutre).
f. PTS2 : Le nombre de points marqués par l'équipe à domicile ou neutre.
g. Attend. : L'affluence du public lors du match (nombre de spectateurs présents).
h. Arena : Le nom de l'arène où le match a été joué.
i. Notes : Des notes ou informations supplémentaires sur le match.

Ces informations seront essentielles pour entraîner notre modèle de prédiction. Les attributs tels que les points marqués par chaque équipe, l'équipe à domicile et l'équipe visiteuse, ainsi que les dates, nous permettront d'extraire des caractéristiques significatives pour effectuer nos prédictions. la figure suivante présente un échantillon de dataset:

Playoffs Schedule Share & Export ▾

Date	Start (ET)	Visitor/Neutral	PTS	Home/Neutral	PTS		Attend.	Arena	Notes
Sat, Apr 15, 2023	1:00p	Brooklyn Nets	101	Philadelphia 76ers	121	Box Score	20,913	Wells Fargo Center	
Sat, Apr 15, 2023	3:30p	Atlanta Hawks	99	Boston Celtics	112	Box Score	19,156	TD Garden	
Sat, Apr 15, 2023	6:00p	New York Knicks	101	Cleveland Cavaliers	97	Box Score	19,432	Rocket Mortgage Fieldhouse	
Sat, Apr 15, 2023	8:30p	Golden State Warriors	123	Sacramento Kings	126	Box Score	18,253	Golden 1 Center	
Sun, Apr 16, 2023	3:00p	Los Angeles Lakers	128	Memphis Grizzlies	112	Box Score	18,487	FedEx Forum	
Sun, Apr 16, 2023	5:30p	Miami Heat	130	Milwaukee Bucks	117	Box Score	17,381	Fiserv Forum	
Sun, Apr 16, 2023	8:00p	Los Angeles Clippers	115	Phoenix Suns	110	Box Score	17,071	Footprint Center	
Sun, Apr 16, 2023	10:30p	Minnesota Timberwolves	80	Denver Nuggets	109	Box Score	19,628	Ball Arena	
Mon, Apr 17, 2023	7:30p	Brooklyn Nets	84	Philadelphia 76ers	96	Box Score	20,958	Wells Fargo Center	

En utilisant ces données historiques, nous serons en mesure de former notre modèle d'apprentissage supervisé de type classification binaire pour prédire les vainqueurs des matchs futurs, des séries de matchs de la conférence Est et Ouest, ainsi que le champion final de la NBA pour la saison 2022/2023.

Il conviendra de nettoyer et de prétraiter les données, en s'assurant de traiter correctement les valeurs manquantes, de convertir les attributs pertinents en formats appropriés et de créer des ensembles de données d'entraînement et de test pour évaluer les performances de notre modèle de prédiction.

Prétraitement de dataset

Le prétraitement effectué sur le dataset récupéré à partir du site "https://www.basketball-reference.com/playoffs/NBA_2023_games.html" comprend plusieurs étapes afin de préparer les données pour l'entraînement du modèle de prédiction :

1. Suppression des colonnes : La colonne "Start (ET)" qui indique l'heure de début du match, la colonne "Attend." qui représente l'affluence du public, la colonne "Arena" qui contient le nom de l'arène où le match a été joué, ainsi que la colonne "Notes" qui fournit des informations supplémentaires sur le match, sont supprimées du dataset. Ces informations ne sont pas nécessaires pour notre tâche

de prédiction.

2. Création de la colonne "résultat" : Les deux colonnes "PTS1" (points marqués par l'équipe visiteuse/neutre) et "PTS2" (points marqués par l'équipe à domicile/neutre) sont remplacées par une seule colonne "résultat". La valeur de cette colonne sera déterminée en comparant les points marqués par les deux équipes. Si les points marqués par l'équipe visiteuse sont supérieurs aux points marqués par l'équipe à domicile, la valeur de "résultat" sera 1, sinon elle sera 0. Cela permettra de créer une classification binaire pour notre modèle de prédiction.

3. Extraction du jour de la semaine et du jour du mois : À partir de la colonne "Date", nous extrayons le jour de la semaine (ex : lundi, mardi, etc.) et le jour du mois (ex : 1, 2, 3, etc.) pour fournir des informations temporelles supplémentaires à notre modèle.

4. Suppression de la colonne "Date" : La colonne "Date" est supprimée du dataset car nous avons déjà extrait les informations temporelles pertinentes dans les étapes précédentes.

5. Suppression des matchs non joués : Les matchs pour lesquels les colonnes "PTS1" et "PTS2" sont vides (indiquant que le match n'a pas encore été joué) sont supprimés du dataset. Cela garantit que nous n'incluons que les matchs pour lesquels nous avons des résultats concrets dans notre ensemble de données.

En effectuant ces étapes de prétraitement, nous simplifions et préparons les données pour l'entraînement de notre modèle de prédiction. Les colonnes inutiles sont supprimées, les informations pertinentes sont extraites et les matchs non joués sont éliminés, ce qui permet de créer un ensemble de données cohérent pour notre tâche de prédiction du vainqueur de la NBA. le résultat obtenu est présentée dans la figure suivante :

	Visitor/Neutral	Home/Neutral	result	day	day_name
0	Brooklyn Nets	Philadelphia 76ers	0	15	6
1	Atlanta Hawks	Boston Celtics	0	15	6
2	New York Knicks	Cleveland Cavaliers	1	15	6
3	Golden State Warriors	Sacramento Kings	0	15	6
4	Los Angeles Lakers	Memphis Grizzlies	1	16	7
5	Miami Heat	Milwaukee Bucks	1	16	7
6	Los Angeles Clippers	Phoenix Suns	1	16	7
7	Minnesota Timberwolves	Denver Nuggets	0	16	7
8	Brooklyn Nets	Philadelphia 76ers	0	17	1
9	Golden State Warriors	Sacramento Kings	0	17	1

Encodage des chaînes de caractères

L'encodage des chaînes de caractères en nombres entiers et la normalisation des données sont deux étapes essentielles du prétraitement des données pour le projet de prédiction dans la NBA. Voici comment ces étapes sont appliquées :

Codage des chaînes de caractères en nombres entiers : Pour traiter les attributs contenant des chaînes de caractères tels que les noms des équipes, il est nécessaire de les encoder en nombres entiers. Cela permet de représenter chaque nom d'équipe par un identifiant unique. Par exemple, si nous avons les équipes "Los Angeles Lakers", "Denver Nuggets" et "Miami Heat", nous pouvons leur attribuer les identifiants 1, 2 et 3 respectivement. Ainsi, au lieu de travailler avec des chaînes de caractères, notre modèle utilisera des nombres entiers pour représenter les équipes. le résultat de cette étape est présenté dans la figure suivante :

	Visitor/Neutral	Home/Neutral	result	day	day_name
0	0	6	0	15	6
1	14	9	0	15	6
2	3	2	1	15	6
3	13	12	0	15	6
4	8	4	1	16	7
...	...	...	...	...	...
77	1	9	0	25	4
78	10	8	0	26	5
79	9	1	0	27	6
80	8	10	0	28	7
81	1	9	0	29	1

82 rows × 5 columns

Normalisation des données : La normalisation des données est une technique couramment utilisée pour mettre à l'échelle les valeurs numériques dans un intervalle spécifique. Dans ce projet, nous appliquerons la normalisation min-max à toutes les colonnes, à l'exception de la colonne "resultat". La normalisation min-max redimensionne les valeurs d'une colonne entre 0 et 1 en utilisant la formule suivante pour chaque valeur :

valeur_normalisée = (valeur - valeur_min) / (valeur_max - valeur_min)

Cela garantit que toutes les valeurs numériques sont mises à la même échelle, ce qui facilite l'apprentissage du modèle et améliore sa performance. Cependant, la colonne "resultat" (qui représente notre variable cible) n'est pas normalisée, car elle contient déjà des valeurs binaires (0 et 1).

En normalisant les données, nous évitons les problèmes liés à des écarts importants entre les valeurs de différentes colonnes. Cela permet également de préserver les

relations entre les valeurs et de garantir que les modèles d'apprentissage peuvent interpréter correctement les données.

	Visitor/Neutral	Home/Neutral	result	day	day_name
0	0.000000	0.400000	0	0.482759	0.833333
1	0.933333	0.600000	0	0.482759	0.833333
2	0.200000	0.133333	1	0.482759	0.833333
3	0.866667	0.800000	0	0.482759	0.833333
4	0.533333	0.266667	1	0.517241	1.000000
...	...	...	...	...	...
77	0.066667	0.600000	0	0.827586	0.500000
78	0.666667	0.533333	0	0.862069	0.666667
79	0.600000	0.066667	0	0.896552	0.833333
80	0.533333	0.666667	0	0.931034	1.000000
81	0.066667	0.600000	0	0.965517	0.000000

82 rows × 5 columns

Ensemble de données de formation et de test

La division de l'ensemble de données en ensembles de formation et de test est une étape cruciale dans la construction de modèles de prédiction. Cette division nous permet d'entraîner le modèle et d'évaluer leur performance sur des données indépendantes et non utilisées lors de l'entraînement. Voici comment cette division est réalisée :

Préparation des variables d'entrée (x) et de sortie (y) : À partir du dataset (dataframe), nous sélectionnons les colonnes qui serviront de variables d'entrée pour notre modèle ('day_name', 'day', 'Visitor/Neutral' et 'Home/Neutral') et nous sélectionnons ‘result’ comme une variable de sortie. Nous les convertissons en tableaux NumPy pour faciliter

la manipulation.

Séparation en ensembles de formation et de test : Nous utilisons la fonction train_test_split fournie par une librairie telle que scikit-learn pour diviser notre ensemble de données en ensembles de formation et de test. Nous utilisons un ratio de division de 80% pour l'ensemble d'entraînement (x_train, y_train) et de 20% pour l'ensemble de test (x_test, y_test). Cela signifie que 80% des données seront utilisées pour l'entraînement du modèle et 20% pour l'évaluation de sa performance.

Régression logistique

Une fois que nous avons divisé notre ensemble de données en ensembles de formation et de test, nous pouvons procéder à l'entraînement du modèle en utilisant l'ensemble de formation. En utilisant scikit-learn, nous pouvons importer la classe LogisticRegression pour créer une instance de notre modèle de régression logistique.

Lors de l'entraînement, notre modèle ajuste ses paramètres en utilisant l'algorithme de régression logistique afin de trouver les meilleures valeurs qui minimisent l'erreur entre les valeurs prédites et les valeurs réelles.

En utilisant l'ensemble de test et les fonctions d'erreur Mean Squared Error (MSE) et le Mean Absolute Error (MAE , le résultat de test de notre modèle est affiché dans le tableau suivant.

Fonction d'Erreur	valeur
Mean Squared Error (MSE)	0.302
Mean Absolute Error (MAE)	0.442

Réseau de neurones artificiels

En plus de la régression logistique nous avons entraîné un modèle de réseau de neurones artificiels multicouches. Nous avons utilisé une architecture avec quatres couches : une

couche d'entrée, deux couches cachées et une couche de sortie.

La couche d'entrée, définie par input_shape=(4,), indique que notre modèle s'attend à recevoir des données d'entrée avec quatre caractéristiques qui sont 'day_name', 'day', 'Visitor/Neutral' et 'Home/Neutral'. Les couches cachées, définies par Dense(32, activation='relu') et Dense(16, activation='relu'), comportent respectivement 32 et 16 neurones avec la fonction d'activation ReLU. Ces couches sont responsables de l'apprentissage des caractéristiques complexes des données. Enfin, la couche de sortie, définie par Dense(1, activation='sigmoid'), comporte un neurone avec la fonction d'activation sigmoïde. Cette couche est utilisée pour effectuer une classification binaire en produisant une probabilité entre 0 et 1.

Nous avons utilisé l'optimiseur Adam avec la fonction de perte binary_crossentropy pour minimiser l'erreur entre les prédictions et les étiquettes réelles.

Les résultats obtenus sont présentés dans le tableau suivant :

métriques	valeur
binary_crossentropy	0.67
accuracy	0.64

RÉSULTATS

Nous remarquons que la finale pour la conférence Est et Ouest se déroule sur une série de sept matchs. le calendrier des deux finales sont présentés dans les deux figures suivante:

Eastern Conference Finals ▲	Boston Celtics trail Miami Heat	Series Stats
Game 1	Wed, May 17	Miami Heat @ Boston Celtics
Game 2	Fri, May 19	Miami Heat @ Boston Celtics
Game 3	Sun, May 21	Boston Celtics @ Miami Heat
Game 4	Tue, May 23	Boston Celtics @ Miami Heat
Game 5	Thu, May 25	Miami Heat @ Boston Celtics
Game 6	Sat, May 27	Boston Celtics @ Miami Heat
Game 7	Mon, May 29	Miami Heat @ Boston Celtics

Western Conference Finals ▲	Denver Nuggets lead Los Angeles Lakers	Series Stats
Game 1	Tue, May 16	Los Angeles Lakers @ Denver Nuggets
Game 2	Thu, May 18	Los Angeles Lakers @ Denver Nuggets
Game 3	Sat, May 20	Denver Nuggets @ Los Angeles Lakers
Game 4	Mon, May 22	Denver Nuggets @ Los Angeles Lakers
Game 5	Wed, May 24	Los Angeles Lakers @ Denver Nuggets
Game 6	Fri, May 26	Denver Nuggets @ Los Angeles Lakers
Game 7	Sun, May 28	Los Angeles Lakers @ Denver Nuggets

Avant d'effectuer des prédictions à l'aide des modèles, il est nécessaire de commencer par encoder et normaliser les données provenant des 14 matchs (7 de la conférence de l'Est et 7 de la conférence de l'Ouest).

Ensuite, nous procédons à la prédiction des résultats pour les 7 matchs de chaque conférence et calculons le nombre de matchs remportés par chaque équipe.

L'équipe qui remporte le plus de matchs lors de chaque finale de conférence sera considérée comme le vainqueur le plus probable.

Les prédictions des vainqueurs générées par le modèle de réseau neuronal, la régression logistique et le calcul statistique sont présentées dans le tableau ci-dessous.

Selon les résultats, il semble que les Boston Celtics soient les favoris pour remporter la Conférence Est, tandis que les Denver Nuggets sont les favoris pour remporter la Conférence Ouest. En ce qui concerne la finale entre les Boston Celtics et les Denver

Nuggets, les modèles prédisent que les Denver Nuggets seront les champions.

métriques	Réseau de neurones artificiels	Régression logistique	Statistique
le vainqueur de la conférence Est	Boston Celtics (2, 5)	Boston Celtics (3, 4)	Boston Celtics
le vainqueur de la conférence Ouest	Denver Nuggets (5, 2)	Denver Nuggets (4, 3)	Denver Nuggets
le champion NBA 2022/2023	Denver Nuggets	Denver Nuggets	