Imports

In [110…

```
import pandas as pd
import seaborn as sns
import os
import matplotlib.pyplot as plt
import numpy as np
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.feature_selection import RFE
from sklearn.model_selection import GridSearchCV
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, precision_score, recall_score,balanced_accur
from sklearn.metrics import mean_squared_error,r2_score
from sklearn.metrics import confusion_matrix
import xgboost as xgb
from imblearn.under_sampling import RandomUnderSampler
from imblearn.over_sampling import SMOTE
from sklearn.metrics import classification_report
from imblearn.ensemble import BalancedRandomForestClassifier
import requests
from bs4 import BeautifulSoup
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import RandomizedSearchCV
import requests
from bs4 import BeautifulSoup
import random
from ydata_profiling import ProfileReport
np.set_printoptions(threshold=np.inf)
```

In [111…

```
# Définir le chemin du répertoire contenant vos fichiers CSV
directory = '/Users/stanislas/Downloads/To-Import-Model'

# Liste pour stocker les DataFrames de chaque fichier
dfs = []

# Parcourir tous les fichiers dans le répertoire
for filename in os.listdir(directory):
    if filename.endswith(".csv"):  # Assurez-vous que le fichier est un fichier CSV
        filepath = os.path.join(directory, filename)
        # Lire le fichier CSV dans un DataFrame et l'ajouter à la liste
        data = pd.read_csv(filepath)
        dfs.append(data)

# Concaténer tous les DataFrames dans un DataFrame global
df = pd.concat(dfs, ignore_index=True)
```

In [112…

```
players = pd.read_csv('atp_players.csv')
```

In [113…

```
top500=pd.read_excel('Top500-Modified.xlsx')
```

DF reprocessing

Loading [MathJax]/extensions/Safe.js

Basic information

In [90]:
```
df.dtypes
```

Out[90]:
```
tourney_id               object
tourney_name             object
surface                  object
draw_size               float64
tourney_level            object
tourney_date              int64
match_num                 int64
winner_id                 int64
winner_seed             float64
winner_entry             object
winner_name              object
winner_hand              object
winner_ht               float64
winner_ioc               object
winner_age              float64
loser_id                  int64
loser_seed              float64
loser_entry              object
loser_name               object
loser_hand               object
loser_ht                float64
loser_ioc                object
loser_age               float64
score                    object
best_of                   int64
round                    object
minutes                 float64
w_ace                   float64
w_df                    float64
w_svpt                  float64
w_1stIn                 float64
w_1stWon                float64
w_2ndWon                float64
w_SvGms                 float64
w_bpSaved               float64
w_bpFaced               float64
l_ace                   float64
l_df                    float64
l_svpt                  float64
l_1stIn                 float64
l_1stWon                float64
l_2ndWon                float64
l_SvGms                 float64
l_bpSaved               float64
l_bpFaced               float64
winner_rank             float64
winner_rank_points      float64
loser_rank              float64
loser_rank_points       float64
dtype: object
```

In [91]:
```
df.describe()
```

Out[91]:

	draw_size	tourney_date	match_num	winner_id	winner_seed	winner_ht	winner_age
count	191085.000000	1.919200e+05	191920.000000	191920.000000	70926.000000	175012.000000	190609.000000
mean	53.088479	1.993711e+07	79.398656	104499.014537	6.308279	184.491618	25.670153
std	36.645414	1.581156e+05	111.963129	13664.799449	5.547887	6.672384	4.053105
min	2.000000	1.967123e+07	1.000000	100001.000000	1.000000	160.000000	14.300000
25%	32.000000	1.980051e+07	11.000000	100417.000000	2.000000	180.000000	22.700000
50%	32.000000	1.993052e+07	25.000000	101733.000000	5.000000	185.000000	25.300000
75%	64.000000	2.007032e+07	94.000000	103990.000000	8.000000	188.000000	28.200000
max	128.000000	2.023113e+07	1701.000000	212428.000000	35.000000	211.000000	58.700000

8 rows × 35 columns

In [7]:

```
df.isna().sum()
```

Out[7]:

```
tourney_id                 0
tourney_name               0
surface                 2990
draw_size                835
tourney_level              0
tourney_date               0
match_num                  0
winner_id                  0
winner_seed           120994
winner_entry          175177
winner_name                0
winner_hand               12
winner_ht              16908
winner_ioc                 8
winner_age              1311
loser_id                   0
loser_seed            155821
loser_entry           163496
loser_name                 0
loser_hand                49
loser_ht               29663
loser_ioc                 72
loser_age               4657
score                      9
best_of                    0
round                      0
minutes                99653
w_ace                  96885
w_df                   96886
w_svpt                 96886
w_1stIn                96886
w_1stWon               96886
w_2ndWon               96886
w_SvGms                96885
w_bpSaved              96886
w_bpFaced              96886
l_ace                  96886
l_df                   96885
l_svpt                 96886
l_1stIn                96886
l_1stWon               96886
l_2ndWon               96886
l_SvGms                96885
l_bpSaved              96886
l_bpFaced              96886
winner_rank            35761
winner_rank_points     82984
loser_rank             44132
loser_rank_points      84612
dtype: int64
```

Keep only significant columns

In [10]:

```
df = df[['tourney_id','tourney_name','surface','draw_size','tourney_level','tourney_date
         'match_num','winner_id','winner_hand','winner_ht','winner_age','loser_id','lose
        'loser_ht','loser_age','best_of','round']]
```

Further analysis

In [13]:

```
profile = ProfileReport(df, title="Rapport d'analyse exploratoire")
```

Loading [MathJax]/extensions/Safe.js

In [14]:

```
profile
```

```
Summarize dataset:   0%|          | 0/5 [00:00<?, ?it/s]
Generate report structure:   0%|          | 0/1 [00:00<?, ?it/s]
Render HTML:   0%|          | 0/1 [00:00<?, ?it/s]
```

Overview

Dataset statistics

Number of variables	17
Number of observations	191920
Missing cells	56425
Missing cells (%)	1.7%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	24.9 MiB
Average record size in memory	136.0 B

Variable types

Text	2
Categorical	6
Numeric	9

Alerts

`winner_hand` is highly imbalanced (65.9%)	**Imbalance**
`loser_hand` is highly imbalanced (61.5%)	**Imbalance**
`best_of` is highly imbalanced (52.0%)	**Imbalance**

Out[14]:

Data Format

Datetime format for tourney_date

In [114…

```python
df['tourney_date']=df['tourney_date'].fillna(0)

# Convertir la colonne de flottants en chaînes de caractères
df['date_str'] = df['tourney_date'].astype(int).astype(str)

# Fonction pour convertir une chaîne de caractères en date, avec gestion des erreurs
def convert_to_date(date_str):
    try:
        return pd.to_datetime(date_str, format='%Y%m%d').strftime('%Y-%m-%d')
    except ValueError:
        return None  # Remplacer par NaN ou une valeur par défaut si nécessaire

# Appliquer la fonction à chaque valeur de la colonne et créer une nouvelle colonne 'dat
df['tourney_date'] = df['date_str'].apply(convert_to_date)

# Supprimer la colonne temporaire de chaînes de caractères
df.drop(columns=['date_str'], inplace=True)
```

Reducing values in the column tourney_name

In [115…

```python
sorted(df['tourney_name'].unique())
```

```
Out[115]: ['ATP Rio de Janeiro',
           'Aberavon',
           'Acapulco',
           'Adelaide',
           'Adelaide 1',
           'Adelaide 2',
           'Adelaide-2',
           'Aix en Provence',
           'Aix-en-Provence',
           'Alamo WCT',
           'Albany',
           'Algiers',
           'Amersfoort',
           'Amsterdam',
           'Amsterdam WCT',
           'Anaheim',
           'Ancona',
           'Antalya',
           'Antwerp',
           'Aptos',
           'Astana',
           'Athens',
           'Athens Olympics',
           'Atlanta',
           'Atlanta Olympics',
           'Atlanta WCT',
           'Atp Cup',
           'Auckland',
           'Australian Chps.',
           'Australian Open',
           'Australian Open-2',
           'Australian Round Robin',
           'Aviles',
           'Bahia',
           'Bakersfield WCT',
           'Baltimore',
           'Baltimore WCT',
           'Bangalore',
           'Bangkok',
           'Banja Luka',
           'Barcelona',
           'Barcelona 2',
           'Barcelona Nationals',
           'Barcelona Olympics',
           'Barcelona WCT',
           'Bari',
           'Barranquilla',
           'Basel',
           'Bastad',
           'Bastad 1',
           'Bastad WCT',
           'Beckenham',
           'Beijing',
           'Beijing Olympics',
           'Belgrade',
           'Belgrade ',
           'Belgrade 2',
           'Berkeley',
           'Berlin',
           'Bermuda',
           'Binghamton',
           'Binghamton NTL',
           'Birmingham',
           'Birmingham WCT',
```

Loading [MathJax]/extensions/Safe.js

```
               'Bloemfontein',
               'Boca Raton',
               'Boca West',
               'Bogota',
               'Bogota NTL',
               'Bologna',
               'Bologna WCT',
               'Bolzano',
               'Bombay',
               'Bordeaux',
               'Boston',
               'Boston 1',
               'Boston 2',
               'Boston WCT',
               'Bournemouth',
               'Brasilia',
               'Bremen Indoors',
               'Bretton Woods',
               'Brighton',
               'Brisbane',
               'Bristol',
               'Brussels',
               'Brussels WCT',
               'Bucharest',
               'Budapest',
               'Buenos Aires',
               'Buenos Aires NTL',
               'Buffalo',
               'Buffalo WCT',
               'Buzios',
               'Cagliari',
               'Cairo',
               'Calcutta',
               'Calgary',
               'Cambridge',
               'Canada Masters',
               'Cannes',
               'Cannes Chps',
               'Cannes WCT',
               "Cap D'Adge WCT",
               'Cape Town',
               'Cape Town WCT',
               'Caracas',
               'Caracas WCT',
               'Casablanca',
               'Casablanca WCT',
               'Catania',
               'Cedar Grove',
               'Champions Classic',
               'Charleston',
               'Charlotte',
               'Charlotte WCT',
               'Chengdu',
               'Chennai',
               'Chicago',
               'Chicago WCT',
               'Chicago-2 WCT',
               'Christchurch',
               'Cincinnati',
               'Cincinnati Masters',
               'Cleveland',
               'Cleveland WCT',
               'Cologne',
               'Cologne 1',
               'Cologne 2',
               'Cologne WCT',
               'Colombus',
               'Columbia',
               'Columbus',
               'Columbus WCT',
               'Concord Indoors',
               'Copenhagen',
               'Copenhagen WCT',
               'Coral Springs',
               'Cordoba',
               'Corpus Christi',
               'Corpus Christi NTL',
               'Costa Do Sauipe',
               'Curacao',
               'Dallas',
               'Dallas WCT',
               'Davis Cup AFR QF: ALG vs CIV',
               'Davis Cup AFR QF: CIV vs NGR',
               'Davis Cup AFR QF: KEN vs ZIM',
               'Davis Cup AFR QF: MAR vs KEN',
               'Davis Cup AFR QF: MEX vs KOR',
               'Davis Cup AFR QF: SEN vs ALG',
               'Davis Cup AFR QF: SEN vs NGR',
               'Davis Cup AFR QF: ZIM vs LBA',
               'Davis Cup AFR R1: CIV vs TUN',
               'Davis Cup AFR R1: SEN vs MAR',
               'Davis Cup AFR R1: TUN vs NGR',
               'Davis Cup AFR SF: ALG vs NGR',
               'Davis Cup AFR SF: NZL vs KOR',
               'Davis Cup AFR SF: SEN vs EGY',
               'Davis Cup AFR SF: ZIM vs MAR',
               'Davis Cup AFR SF: ZIM vs NGR',
               'Davis Cup AME F: ARG vs USA',
               'Davis Cup AME F: BRA vs CAN',
               'Davis Cup AME F: BRA vs ECU',
               'Davis Cup AME F: BRA vs MEX',
               'Davis Cup AME F: CAN vs COL',
               'Davis Cup AME F: CAN vs NZL',
               'Davis Cup AME F: CAN vs PAR',
               'Davis Cup AME F: CHI vs ARG',
               'Davis Cup AME F: CHI vs BRA',
               'Davis Cup AME F: CHI vs RSA',
               'Davis Cup AME F: COL vs USA',
               'Davis Cup AME F: COL vs VEN',
               'Davis Cup AME F: ECU vs BRA',
               'Davis Cup AME F: JPN vs AUS',
               'Davis Cup AME F: JPN vs PHI',
               'Davis Cup AME F: MEX vs AUS',
               'Davis Cup AME F: MEX vs BRA',
               'Davis Cup AME F: MEX vs NZL',
               'Davis Cup AME F: MEX vs USA',
               'Davis Cup AME F: USA vs ARG',
               'Davis Cup AME F: USA vs COL',
               'Davis Cup AME F: USA vs ECU',
               'Davis Cup AME F: USA vs MEX',
               'Davis Cup AME F: USA vs RSA',
               'Davis Cup AME PR: ARG vs ECU',
               'Davis Cup AME PR: BOL vs PER',
               'Davis Cup AME PR: BRA vs BOL',
               'Davis Cup AME PR: BRA vs PER',
               'Davis Cup AME PR: CAN vs CAR',
               'Davis Cup AME PR: CAN vs COL',
               'Davis Cup AME PR: CAN vs MEX',
               'Davis Cup AME PR: CAR vs CAN',
               'Davis Cup AME PR: CAR vs USA',
               'Davis Cup AME PR: CAR vs VEN',
               'Davis Cup AME PR: COL vs CAR',
               'Davis Cup AME PR: COL vs VEN',
               'Davis Cup AME PR: ECU vs ARG',
               'Davis Cup AME PR: MEX vs CAN',
               'Davis Cup AME PR: MEX vs CAR',
               'Davis Cup AME PR: PER vs URU',
               'Davis Cup AME PR: PHI vs JPN',
               'Davis Cup AME PR: URU vs ECU',
               'Davis Cup AME PR: USA vs MEX',
               'Davis Cup AME PR: USA vs VEN',
               'Davis Cup AME PR: VEN vs COL',
               'Davis Cup AME PR: VEN vs USA',
               'Davis Cup AME QF: ARG vs BRA',
               'Davis Cup AME QF: ARG vs URU',
               'Davis Cup AME QF: BRA vs ARG',
               'Davis Cup AME QF: BRA vs PER',
               'Davis Cup AME QF: CAN vs CAR',
               'Davis Cup AME QF: CAN vs VEN',
               'Davis Cup AME QF: CAR vs BRA',
               'Davis Cup AME QF: CAR vs CHI',
               'Davis Cup AME QF: CAR vs URU',
               'Davis Cup AME QF: CHI vs CAN',
               'Davis Cup AME QF: CHI vs COL',
               'Davis Cup AME QF: COL vs BRA',
               'Davis Cup AME QF: COL vs PER',
               'Davis Cup AME QF: ECU vs BOL',
               'Davis Cup AME QF: ECU vs CAR',
               'Davis Cup AME QF: HKG vs AUS',
               'Davis Cup AME QF: JPN vs CHN',
               'Davis Cup AME QF: JPN vs PHI',
               'Davis Cup AME QF: MEX vs CAN',
               'Davis Cup AME QF: MEX vs PER',
               'Davis Cup AME QF: MEX vs USA',
               'Davis Cup AME QF: PER vs CAN',
               'Davis Cup AME QF: URU vs BRA',
               'Davis Cup AME QF: URU vs COL',
               'Davis Cup AME QF: URU vs MEX',
               'Davis Cup AME QF: USA vs MEX',
               'Davis Cup AME QF: VEN vs BRA',
               'Davis Cup AME R1: CAR vs CUB',
               'Davis Cup AME R1: COL vs URU',
               'Davis Cup AME R1: PER vs BRA',
               'Davis Cup AME R1: VEN vs CAN',
               'Davis Cup AME R1: VEN vs MEX',
               'Davis Cup AME R1: VEN vs URU',
               'Davis Cup AME SF: ARG vs CHI',
               'Davis Cup AME SF: ARG vs PER',
               'Davis Cup AME SF: AUS vs INA',
               'Davis Cup AME SF: AUS vs JPN',
               'Davis Cup AME SF: BRA vs CHI',
               'Davis Cup AME SF: BRA vs COL',
               'Davis Cup AME SF: BRA vs URU',
               'Davis Cup AME SF: CAN vs CAR',
               'Davis Cup AME SF: CAN vs ECU',
               'Davis Cup AME SF: CAN vs MEX',
               'Davis Cup AME SF: CAR vs MEX',
               'Davis Cup AME SF: CAR vs NZL',
               'Davis Cup AME SF: CAR vs USA',
               'Davis Cup AME SF: CHI vs ARG',
               'Davis Cup AME SF: CHI vs BRA',
               'Davis Cup AME SF: CHI vs MEX',
               'Davis Cup AME SF: CHN vs JPN',
               'Davis Cup AME SF: COL vs CAN',
               'Davis Cup AME SF: COL vs CAR',
               'Davis Cup AME SF: ECU vs CAN',
               'Davis Cup AME SF: INA vs AUS',
               'Davis Cup AME SF: JPN vs AUS',
               'Davis Cup AME SF: JPN vs PHI',
               'Davis Cup AME SF: KOR vs AUS',
               'Davis Cup AME SF: MEX vs CAN',
               'Davis Cup AME SF: MEX vs NZL',
               'Davis Cup AME SF: PHI vs JPN',
               'Davis Cup AME SF: URU vs BRA',
               'Davis Cup AME SF: USA vs CAR',
               'Davis Cup AME SF: USA vs RSA',
               'Davis Cup AME SF: VEN vs CAN',
               'Davis Cup AME SF: VEN vs CAR',
               'Davis Cup AOC F: NZL vs KOR',
               'Davis Cup AOC QF: KOR vs PAK',
               'Davis Cup AOC QF: KOR vs PHI',
               'Davis Cup AOC QF: PAK vs INA',
               'Davis Cup AOC QF: PAK vs KOR',
               'Davis Cup AOC QF: PHI vs KOR',
               'Davis Cup AOC QF: SRI vs IND',
               'Davis Cup AOC R1: INA vs PHI',
               'Davis Cup AOC SF: INA vs PAK',
               'Davis Cup AOC SF: IND vs THA',
               'Davis Cup AOC SF: PHI vs KOR',
               'Davis Cup AOC SF: THA vs IND',
               'Davis Cup EUR F: AUS vs NZL',
               'Davis Cup EUR F: BRA vs ROU',
               'Davis Cup EUR F: CHN vs NZL',
               'Davis Cup EUR F: COL vs RSA',
               'Davis Cup EUR F: DEN vs AUT',
               'Davis Cup EUR F: ESP vs HUN',
               'Davis Cup EUR F: ESP vs ITA',
               'Davis Cup EUR F: ESP vs SWE',
               'Davis Cup EUR F: ESP vs TCH',
               'Davis Cup EUR F: ESP vs YUG',
               'Davis Cup EUR F: FRA vs AUT',
               'Davis Cup EUR F: FRA vs ROU',
               'Davis Cup EUR F: FRG vs ESP',
               'Davis Cup EUR F: FRG vs RSA',
               'Davis Cup EUR F: FRG vs SUI',
               'Davis Cup EUR F: FRG vs URS',
               'Davis Cup EUR F: GBR vs ISR',
               'Davis Cup EUR F: GBR vs ITA',
               'Davis Cup EUR F: GBR vs ROU',
               'Davis Cup EUR F: GBR vs RSA',
               'Davis Cup EUR F: GBR vs TCH',
               'Davis Cup EUR F: HUN vs DEN',
               'Davis Cup EUR F: HUN vs ESP',
               'Davis Cup EUR F: IND vs JPN',
               'Davis Cup EUR F: IRL vs SUI',
               'Davis Cup EUR F: ITA vs GBR',
               'Davis Cup EUR F: ITA vs ROU',
               'Davis Cup EUR F: ITA vs SWE',
               'Davis Cup EUR F: JPN vs IND',
               'Davis Cup EUR F: NZL vs AUS',
               'Davis Cup EUR F: PAK vs JPN',
               'Davis Cup EUR F: ROU vs DEN',
               'Davis Cup EUR F: ROU vs FRG',
               'Davis Cup EUR F: ROU vs TCH',
               'Davis Cup EUR F: ROU vs URS',
               'Davis Cup EUR F: SUI vs ISR',
               'Davis Cup EUR F: SWE vs HUN',
               'Davis Cup EUR F: TCH vs ESP',
               'Davis Cup EUR F: TCH vs FRA',
               'Davis Cup EUR F: TCH vs ITA',
               'Davis Cup EUR F: TCH vs SWE',
               'Davis Cup EUR F: URS vs HUN',
               'Davis Cup EUR F: URS vs ISR',
               'Davis Cup EUR F: URS vs NED',
               'Davis Cup EUR F: URS vs ROU',
               'Davis Cup EUR F: URS vs SUI',
               'Davis Cup EUR F: URS vs TCH',
               'Davis Cup EUR F: YUG vs HUN',
               'Davis Cup EUR PQ: ALG vs MAR',
               'Davis Cup EUR PQ: EGY vs MAR',
               'Davis Cup EUR PQ: IRI vs ALG',
               'Davis Cup EUR PQ: IRI vs ISR',
               'Davis Cup EUR PQ: IRI vs LIB',
               'Davis Cup EUR PQ: ISR vs LUX',
               'Davis Cup EUR PQ: ISR vs TUR',
               'Davis Cup EUR PQ: LUX vs FIN',
               'Davis Cup EUR PQ: LUX vs POR',
               'Davis Cup EUR PQ: LUX vs TUR',
               'Davis Cup EUR PQ: MAR vs TUR',
               'Davis Cup EUR PQ: NGR vs KEN',
               'Davis Cup EUR PQ: NGR vs MAR',
               'Davis Cup EUR PQ: PHI vs PAK',
               'Davis Cup EUR PQ: TUR vs IRI',
               'Davis Cup EUR PQ: TUR vs IRL',
               'Davis Cup EUR PQ: TUR vs LIB',
               'Davis Cup EUR PQ: TUR vs LUX',
               'Davis Cup EUR PR: ALG vs IRI',
               'Davis Cup EUR PR: AUT vs EGY',
               'Davis Cup EUR PR: AUT vs FIN',
               'Davis Cup EUR PR: AUT vs GBR',
               'Davis Cup EUR PR: AUT vs MON',
               'Davis Cup EUR PR: AUT vs NZL',
               'Davis Cup EUR PR: AUT vs ROU',
               'Davis Cup EUR PR: BEL vs AUT',
               'Davis Cup EUR PR: BEL vs BUL',
               'Davis Cup EUR PR: BEL vs DEN',
               'Davis Cup EUR PR: BEL vs GRE',
               'Davis Cup EUR PR: BEL vs HUN',
               'Davis Cup EUR PR: BEL vs IRL',
               'Davis Cup EUR PR: BEL vs NED',
               'Davis Cup EUR PR: BEL vs NOR',
               'Davis Cup EUR PR: BEL vs ROU',
               'Davis Cup EUR PR: BEL vs YUG',
               'Davis Cup EUR PR: BUL vs AUT',
               'Davis Cup EUR PR: BUL vs HUN',
               'Davis Cup EUR PR: BUL vs IRI',
               'Davis Cup EUR PR: BUL vs TUR',
               'Davis Cup EUR PR: DEN vs FIN',
               'Davis Cup EUR PR: DEN vs GRE',
               'Davis Cup EUR PR: EGY vs BUL',
               'Davis Cup EUR PR: EGY vs GBR',
               'Davis Cup EUR PR: EGY vs IRL',
               'Davis Cup EUR PR: EGY vs LIB',
               'Davis Cup EUR PR: EGY vs POR',
               'Davis Cup EUR PR: EGY vs TUR',
               'Davis Cup EUR PR: ESP vs DEN',
               'Davis Cup EUR PR: ESP vs NED',
               'Davis Cup EUR PR: ESP vs NOR',
               'Davis Cup EUR PR: ESP vs URS',
               'Davis Cup EUR PR: FIN vs EGY',
```

Loading [MathJax]/extensions/Safe.js

```
               'Davis Cup EUR PR: FIN vs MAR',
               'Davis Cup EUR PR: FIN vs POL',
               'Davis Cup EUR PR: FIN vs TUR',
               'Davis Cup EUR PR: FRA vs BEL',
               'Davis Cup EUR PR: FRA vs URS',
               'Davis Cup EUR PR: FRG vs DEN',
               'Davis Cup EUR PR: FRG vs ISR',
               'Davis Cup EUR PR: FRG vs NOR',
               'Davis Cup EUR PR: FRG vs SUI',
               'Davis Cup EUR PR: GBR vs IRI',
               'Davis Cup EUR PR: GRE vs DEN',
               'Davis Cup EUR PR: GRE vs ESP',
               'Davis Cup EUR PR: GRE vs FIN',
               'Davis Cup EUR PR: GRE vs POR',
               'Davis Cup EUR PR: GRE vs URS',
               'Davis Cup EUR PR: HUN vs BEL',
               'Davis Cup EUR PR: HUN vs NED',
               'Davis Cup EUR PR: IRI vs SUI',
               'Davis Cup EUR PR: IRL vs BUL',
               'Davis Cup EUR PR: IRL vs IRI',
               'Davis Cup EUR PR: IRL vs NOR',
               'Davis Cup EUR PR: IRL vs POR',
               'Davis Cup EUR PR: IRL vs SWE',
               'Davis Cup EUR PR: ISR vs AUT',
               'Davis Cup EUR PR: ISR vs FIN',
               'Davis Cup EUR PR: ISR vs IRI',
               'Davis Cup EUR PR: ISR vs MON',
               'Davis Cup EUR PR: ISR vs SUI',
               'Davis Cup EUR PR: ITA vs DEN',
               'Davis Cup EUR PR: ITA vs POL',
               'Davis Cup EUR PR: JPN vs IND',
               'Davis Cup EUR PR: MAR vs BEL',
               'Davis Cup EUR PR: MAR vs LUX',
               'Davis Cup EUR PR: MAR vs NOR',
               'Davis Cup EUR PR: MAR vs YUG',
               'Davis Cup EUR PR: MON vs DEN',
               'Davis Cup EUR PR: MON vs EGY',
               'Davis Cup EUR PR: MON vs GBR',
               'Davis Cup EUR PR: MON vs ISR',
               'Davis Cup EUR PR: MON vs LUX',
               'Davis Cup EUR PR: MON vs NGR',
               'Davis Cup EUR PR: MON vs URS',
               'Davis Cup EUR PR: NED vs DEN',
               'Davis Cup EUR PR: NED vs FIN',
               'Davis Cup EUR PR: NED vs FRA',
               'Davis Cup EUR PR: NED vs GRE',
               'Davis Cup EUR PR: NED vs ISR',
               'Davis Cup EUR PR: NED vs NOR',
               'Davis Cup EUR PR: NED vs YUG',
               'Davis Cup EUR PR: NOR vs IRI',
               'Davis Cup EUR PR: NOR vs TUR',
               'Davis Cup EUR PR: PHI vs THA',
               'Davis Cup EUR PR: POL vs FIN',
               'Davis Cup EUR PR: POL vs FRG',
               'Davis Cup EUR PR: POL vs HUN',
               'Davis Cup EUR PR: POL vs IRI',
               'Davis Cup EUR PR: POL vs NOR',
               'Davis Cup EUR PR: POL vs POR',
               'Davis Cup EUR PR: POR vs FRA',
               'Davis Cup EUR PR: POR vs IRL',
               'Davis Cup EUR PR: POR vs MON',
               'Davis Cup EUR PR: POR vs SUI',
               'Davis Cup EUR PR: ROU vs BEL',
               'Davis Cup EUR PR: SRI vs PAK',
```

Loading [MathJax]/extensions/Safe.js

```
 'Davis Cup EUR PR: SUI vs AUT',
 'Davis Cup EUR PR: SUI vs EGY',
 'Davis Cup EUR PR: SUI vs FRA',
 'Davis Cup EUR PR: SUI vs FRG',
 'Davis Cup EUR PR: SUI vs GBR',
 'Davis Cup EUR PR: SUI vs IRI',
 'Davis Cup EUR PR: SUI vs ISR',
 'Davis Cup EUR PR: SUI vs RHO',
 'Davis Cup EUR PR: SUI vs YUG',
 'Davis Cup EUR PR: SWE vs MON',
 'Davis Cup EUR PR: SWE vs POL',
 'Davis Cup EUR PR: TCH vs IRL',
 'Davis Cup EUR PR: TCH vs NED',
 'Davis Cup EUR PR: TPE vs PHI',
 'Davis Cup EUR PR: URS vs MON',
 'Davis Cup EUR PR: YUG vs BUL',
 'Davis Cup EUR PR: YUG vs GRE',
 'Davis Cup EUR PR: YUG vs POR',
 'Davis Cup EUR PR: YUG vs ROU',
 'Davis Cup EUR QF: ALG vs ESP',
 'Davis Cup EUR QF: AUT vs ALG',
 'Davis Cup EUR QF: AUT vs DEN',
 'Davis Cup EUR QF: AUT vs ESP',
 'Davis Cup EUR QF: AUT vs FRA',
 'Davis Cup EUR QF: AUT vs GRE',
 'Davis Cup EUR QF: AUT vs NOR',
 'Davis Cup EUR QF: AUT vs POR',
 'Davis Cup EUR QF: BEL vs FRG',
 'Davis Cup EUR QF: BEL vs ISR',
 'Davis Cup EUR QF: BEL vs TCH',
 'Davis Cup EUR QF: BEL vs URS',
 'Davis Cup EUR QF: BUL vs BEL',
 'Davis Cup EUR QF: BUL vs EGY',
 'Davis Cup EUR QF: BUL vs FIN',
 'Davis Cup EUR QF: BUL vs FRG',
 'Davis Cup EUR QF: BUL vs SEN',
 'Davis Cup EUR QF: DEN vs BEL',
 'Davis Cup EUR QF: DEN vs NED',
 'Davis Cup EUR QF: DEN vs POL',
 'Davis Cup EUR QF: EGY vs HUN',
 'Davis Cup EUR QF: EGY vs NED',
 'Davis Cup EUR QF: EGY vs TCH',
 'Davis Cup EUR QF: EGY vs YUG',
 'Davis Cup EUR QF: ESP vs BUL',
 'Davis Cup EUR QF: ESP vs FRG',
 'Davis Cup EUR QF: ESP vs GBR',
 'Davis Cup EUR QF: ESP vs SWE',
 'Davis Cup EUR QF: ESP vs YUG',
 'Davis Cup EUR QF: FIN vs BUL',
 'Davis Cup EUR QF: FIN vs DEN',
 'Davis Cup EUR QF: FIN vs IRL',
 'Davis Cup EUR QF: FRA vs AUT',
 'Davis Cup EUR QF: FRA vs ESP',
 'Davis Cup EUR QF: FRA vs FIN',
 'Davis Cup EUR QF: FRA vs SUI',
 'Davis Cup EUR QF: FRA vs YUG',
 'Davis Cup EUR QF: FRG vs AUT',
 'Davis Cup EUR QF: FRG vs EGY',
 'Davis Cup EUR QF: FRG vs ESP',
 'Davis Cup EUR QF: FRG vs GBR',
 'Davis Cup EUR QF: FRG vs IRL',
 'Davis Cup EUR QF: FRG vs SWE',
 'Davis Cup EUR QF: FRG vs URS',
 'Davis Cup EUR QF: GBR vs AUT',
 'Davis Cup EUR QF: GBR vs IRL',
 'Davis Cup EUR QF: GBR vs POR',
 'Davis Cup EUR QF: GBR vs ROU',
 'Davis Cup EUR QF: GRE vs AUT',
 'Davis Cup EUR QF: GRE vs FIN',
 'Davis Cup EUR QF: HUN vs EGY',
 'Davis Cup EUR QF: HUN vs FRG',
 'Davis Cup EUR QF: HUN vs LUX',
 'Davis Cup EUR QF: HUN vs URS',
 'Davis Cup EUR QF: HUN vs YUG',
 'Davis Cup EUR QF: HUN vs ZIM',
 'Davis Cup EUR QF: IND vs JPN',
 'Davis Cup EUR QF: IRI vs RSA',
 'Davis Cup EUR QF: IRL vs MON',
 'Davis Cup EUR QF: IRL vs NED',
 'Davis Cup EUR QF: IRL vs YUG',
 'Davis Cup EUR QF: ISR vs BEL',
 'Davis Cup EUR QF: ISR vs NED',
 'Davis Cup EUR QF: ISR vs POL',
 'Davis Cup EUR QF: ISR vs ROU',
 'Davis Cup EUR QF: ITA vs AUT',
 'Davis Cup EUR QF: ITA vs BUL',
 'Davis Cup EUR QF: ITA vs MON',
 'Davis Cup EUR QF: ITA vs NED',
 'Davis Cup EUR QF: ITA vs YUG',
 'Davis Cup EUR QF: JPN vs PAK',
 'Davis Cup EUR QF: MAR vs SUI',
 'Davis Cup EUR QF: MAR vs URS',
 'Davis Cup EUR QF: MON vs HUN',
 'Davis Cup EUR QF: MON vs IRL',
 'Davis Cup EUR QF: MON vs ISR',
 'Davis Cup EUR QF: MON vs POL',
 'Davis Cup EUR QF: MON vs POR',
 'Davis Cup EUR QF: MON vs TCH',
 'Davis Cup EUR QF: MON vs URS',
 'Davis Cup EUR QF: NED vs ESP',
 'Davis Cup EUR QF: NED vs IRL',
 'Davis Cup EUR QF: NED vs ROU',
 'Davis Cup EUR QF: NGR vs NED',
 'Davis Cup EUR QF: NOR vs AUT',
 'Davis Cup EUR QF: NOR vs FRA',
 'Davis Cup EUR QF: NOR vs ROU',
 'Davis Cup EUR QF: PAK vs PHI',
 'Davis Cup EUR QF: PHI vs IND',
 'Davis Cup EUR QF: POL vs FRA',
 'Davis Cup EUR QF: POL vs ITA',
 'Davis Cup EUR QF: POL vs ROU',
 'Davis Cup EUR QF: POR vs EGY',
 'Davis Cup EUR QF: POR vs HUN',
 'Davis Cup EUR QF: ROU vs AUT',
 'Davis Cup EUR QF: ROU vs FRG',
 'Davis Cup EUR QF: ROU vs GRE',
 'Davis Cup EUR QF: ROU vs IRI',
 'Davis Cup EUR QF: ROU vs ISR',
 'Davis Cup EUR QF: ROU vs POL',
 'Davis Cup EUR QF: ROU vs TCH',
 'Davis Cup EUR QF: SUI vs BEL',
 'Davis Cup EUR QF: SUI vs ESP',
 'Davis Cup EUR QF: SUI vs GRE',
 'Davis Cup EUR QF: SUI vs HUN',
 'Davis Cup EUR QF: SUI vs SEN',
 'Davis Cup EUR QF: SUI vs ZIM',
 'Davis Cup EUR QF: SWE vs AUT',
 'Davis Cup EUR QF: SWE vs ESP',
```

Loading [MathJax]/extensions/Safe.js

```
'Davis Cup EUR QF: SWE vs FRG',
'Davis Cup EUR QF: SWE vs NED',
'Davis Cup EUR QF: TCH vs POL',
'Davis Cup EUR QF: TCH vs POR',
'Davis Cup EUR QF: TCH vs SWE',
'Davis Cup EUR QF: THA vs PAK',
'Davis Cup EUR QF: TPE vs NZL',
'Davis Cup EUR QF: TUR vs FRA',
'Davis Cup EUR QF: TUR vs ROU',
'Davis Cup EUR QF: TUR vs URS',
'Davis Cup EUR QF: URS vs BEL',
'Davis Cup EUR QF: URS vs CAN',
'Davis Cup EUR QF: URS vs MON',
'Davis Cup EUR QF: URS vs YUG',
'Davis Cup EUR QF: YUG vs ESP',
'Davis Cup EUR QF: YUG vs ISR',
'Davis Cup EUR QF: YUG vs ITA',
'Davis Cup EUR QF: YUG vs NZL',
'Davis Cup EUR QF: YUG vs SWE',
'Davis Cup EUR R1: ALG vs BUL',
'Davis Cup EUR R1: AUT vs MAR',
'Davis Cup EUR R1: AUT vs NZL',
'Davis Cup EUR R1: AUT vs RSA',
'Davis Cup EUR R1: BEL vs BUL',
'Davis Cup EUR R1: BEL vs POL',
'Davis Cup EUR R1: BEL vs TCH',
'Davis Cup EUR R1: BUL vs BEL',
'Davis Cup EUR R1: BUL vs CYP',
'Davis Cup EUR R1: BUL vs DEN',
'Davis Cup EUR R1: BUL vs ESP',
'Davis Cup EUR R1: BUL vs TUR',
'Davis Cup EUR R1: CHN vs INA',
'Davis Cup EUR R1: CYP vs IRL',
'Davis Cup EUR R1: DEN vs POR',
'Davis Cup EUR R1: DEN vs TCH',
'Davis Cup EUR R1: DEN vs URS',
'Davis Cup EUR R1: EGY vs ALG',
'Davis Cup EUR R1: EGY vs GRE',
'Davis Cup EUR R1: EGY vs LUX',
'Davis Cup EUR R1: EGY vs MLT',
'Davis Cup EUR R1: EGY vs NOR',
'Davis Cup EUR R1: EGY vs POL',
'Davis Cup EUR R1: EGY vs TCH',
'Davis Cup EUR R1: ESP vs ALG',
'Davis Cup EUR R1: ESP vs NED',
'Davis Cup EUR R1: ESP vs RHO',
'Davis Cup EUR R1: FIN vs BEL',
'Davis Cup EUR R1: FIN vs CYP',
'Davis Cup EUR R1: FIN vs DEN',
'Davis Cup EUR R1: FIN vs IRL',
'Davis Cup EUR R1: FIN vs POR',
'Davis Cup EUR R1: FIN vs SWE',
'Davis Cup EUR R1: FRA vs GBR',
'Davis Cup EUR R1: FRG vs DEN',
'Davis Cup EUR R1: FRG vs NZL',
'Davis Cup EUR R1: FRG vs SUI',
'Davis Cup EUR R1: GBR vs AUT',
'Davis Cup EUR R1: GBR vs FRA',
'Davis Cup EUR R1: GRE vs BEL',
'Davis Cup EUR R1: GRE vs FRG',
'Davis Cup EUR R1: GRE vs HUN',
'Davis Cup EUR R1: GRE vs LUX',
'Davis Cup EUR R1: GRE vs NED',
'Davis Cup EUR R1: GRE vs NOR',
'Davis Cup EUR R1: GRE vs SYR',
'Davis Cup EUR R1: GRE vs URS',
'Davis Cup EUR R1: HUN vs POL',
'Davis Cup EUR R1: HUN vs URS',
'Davis Cup EUR R1: IRI vs EGY',
'Davis Cup EUR R1: IRI vs ISR',
'Davis Cup EUR R1: IRI vs ROU',
'Davis Cup EUR R1: IRL vs BEL',
'Davis Cup EUR R1: IRL vs LUX',
'Davis Cup EUR R1: IRL vs TUR',
'Davis Cup EUR R1: ISR vs NED',
'Davis Cup EUR R1: ISR vs NOR',
'Davis Cup EUR R1: ITA vs AUT',
'Davis Cup EUR R1: ITA vs BEL',
'Davis Cup EUR R1: ITA vs BUL',
'Davis Cup EUR R1: ITA vs HUN',
'Davis Cup EUR R1: ITA vs TCH',
'Davis Cup EUR R1: LIB vs MAR',
'Davis Cup EUR R1: LUX vs BUL',
'Davis Cup EUR R1: LUX vs IRL',
'Davis Cup EUR R1: LUX vs MON',
'Davis Cup EUR R1: LUX vs NOR',
'Davis Cup EUR R1: MAR vs FIN',
'Davis Cup EUR R1: MAR vs HUN',
'Davis Cup EUR R1: MAR vs POL',
'Davis Cup EUR R1: MAR vs SWE',
'Davis Cup EUR R1: MLT vs IRL',
'Davis Cup EUR R1: MON vs BUL',
'Davis Cup EUR R1: MON vs IRL',
'Davis Cup EUR R1: MON vs LUX',
'Davis Cup EUR R1: MON vs MAR',
'Davis Cup EUR R1: MON vs POR',
'Davis Cup EUR R1: MON vs ZIM',
'Davis Cup EUR R1: NED vs CAN',
'Davis Cup EUR R1: NED vs FIN',
'Davis Cup EUR R1: NED vs NOR',
'Davis Cup EUR R1: NGR vs NOR',
'Davis Cup EUR R1: NOR vs AUT',
'Davis Cup EUR R1: NOR vs DEN',
'Davis Cup EUR R1: NOR vs IRL',
'Davis Cup EUR R1: NOR vs POR',
'Davis Cup EUR R1: NOR vs SEN',
'Davis Cup EUR R1: PAK vs SRI',
'Davis Cup EUR R1: POL vs BEL',
'Davis Cup EUR R1: POL vs FIN',
'Davis Cup EUR R1: POL vs GRE',
'Davis Cup EUR R1: POL vs HUN',
'Davis Cup EUR R1: POL vs YUG',
'Davis Cup EUR R1: POL vs ZIM',
'Davis Cup EUR R1: POR vs ISR',
'Davis Cup EUR R1: POR vs LUX',
'Davis Cup EUR R1: POR vs MON',
'Davis Cup EUR R1: POR vs NED',
'Davis Cup EUR R1: POR vs TUN',
'Davis Cup EUR R1: POR vs TUR',
'Davis Cup EUR R1: POR vs ZIM',
'Davis Cup EUR R1: ROU vs DEN',
'Davis Cup EUR R1: ROU vs EGY',
'Davis Cup EUR R1: ROU vs NED',
'Davis Cup EUR R1: ROU vs SUI',
'Davis Cup EUR R1: RSA vs IRI',
'Davis Cup EUR R1: SEN vs MON',
'Davis Cup EUR R1: SEN vs TUN',
'Davis Cup EUR R1: SUI vs FRA',
'Davis Cup EUR R1: SUI vs FRG',
'Davis Cup EUR R1: SUI vs GBR',
'Davis Cup EUR R1: SWE vs ESP',
'Davis Cup EUR R1: SWE vs FRA',
'Davis Cup EUR R1: SWE vs NZL',
'Davis Cup EUR R1: SWE vs RHO',
'Davis Cup EUR R1: SYR vs TUR',
'Davis Cup EUR R1: TCH vs BRA',
'Davis Cup EUR R1: TUN vs SUI',
'Davis Cup EUR R1: TUR vs BEL',
'Davis Cup EUR R1: TUR vs BUL',
'Davis Cup EUR R1: TUR vs GRE',
'Davis Cup EUR R1: TUR vs LUX',
'Davis Cup EUR R1: TUR vs ZIM',
'Davis Cup EUR R1: URS vs GRE',
'Davis Cup EUR R1: URS vs HUN',
'Davis Cup EUR R1: YUG vs FRA',
'Davis Cup EUR R1: YUG vs GBR',
'Davis Cup EUR R1: YUG vs NOR',
'Davis Cup EUR R1: YUG vs NZL',
'Davis Cup EUR R1: YUG vs POL',
'Davis Cup EUR R1: YUG vs TUN',
'Davis Cup EUR R1: ZIM vs POL',
'Davis Cup EUR SF: ARG vs AUS',
'Davis Cup EUR SF: AUS vs TCH',
'Davis Cup EUR SF: AUT vs HUN',
'Davis Cup EUR SF: AUT vs ISR',
'Davis Cup EUR SF: AUT vs ROU',
'Davis Cup EUR SF: AUT vs SUI',
'Davis Cup EUR SF: AUT vs URS',
'Davis Cup EUR SF: BEL vs HUN',
'Davis Cup EUR SF: BRA vs ESP',
'Davis Cup EUR SF: BRA vs TCH',
'Davis Cup EUR SF: BUL vs SUI',
'Davis Cup EUR SF: BUL vs YUG',
'Davis Cup EUR SF: COL vs MEX',
'Davis Cup EUR SF: DEN vs EGY',
'Davis Cup EUR SF: DEN vs ROU',
'Davis Cup EUR SF: ESP vs FRA',
'Davis Cup EUR SF: ESP vs GBR',
'Davis Cup EUR SF: ESP vs IRL',
'Davis Cup EUR SF: ESP vs MON',
'Davis Cup EUR SF: ESP vs ROU',
'Davis Cup EUR SF: ESP vs USA',
'Davis Cup EUR SF: FIN vs NED',
'Davis Cup EUR SF: FRA vs BUL',
'Davis Cup EUR SF: FRA vs ESP',
'Davis Cup EUR SF: FRA vs GBR',
'Davis Cup EUR SF: FRA vs ITA',
'Davis Cup EUR SF: FRA vs TCH',
'Davis Cup EUR SF: FRG vs BEL',
'Davis Cup EUR SF: FRG vs HUN',
'Davis Cup EUR SF: FRG vs IND',
'Davis Cup EUR SF: FRG vs TCH',
'Davis Cup EUR SF: GBR vs BRA',
'Davis Cup EUR SF: GBR vs ESP',
'Davis Cup EUR SF: GBR vs FRA',
'Davis Cup EUR SF: GBR vs FRG',
'Davis Cup EUR SF: GBR vs ROU',
'Davis Cup EUR SF: GBR vs SUI',
'Davis Cup EUR SF: HUN vs ESP',
'Davis Cup EUR SF: HUN vs ISR',
'Davis Cup EUR SF: HUN vs ITA',
'Davis Cup EUR SF: HUN vs SUI',
```

Loading [MathJax]/extensions/Safe.js

```
               'Davis Cup EUR SF: HUN vs TCH',
               'Davis Cup EUR SF: IND vs FRG',
               'Davis Cup EUR SF: IND vs JPN',
               'Davis Cup EUR SF: IND vs NZL',
               'Davis Cup EUR SF: IND vs PAK',
               'Davis Cup EUR SF: IRL vs FIN',
               'Davis Cup EUR SF: ISR vs FRG',
               'Davis Cup EUR SF: ISR vs HUN',
               'Davis Cup EUR SF: ISR vs SUI',
               'Davis Cup EUR SF: ITA vs ESP',
               'Davis Cup EUR SF: ITA vs FRA',
               'Davis Cup EUR SF: ITA vs HUN',
               'Davis Cup EUR SF: ITA vs SUI',
               'Davis Cup EUR SF: ITA vs SWE',
               'Davis Cup EUR SF: ITA vs TCH',
               'Davis Cup EUR SF: ITA vs URS',
               'Davis Cup EUR SF: JPN vs THA',
               'Davis Cup EUR SF: MEX vs COL',
               'Davis Cup EUR SF: MON vs DEN',
               'Davis Cup EUR SF: NED vs ISR',
               'Davis Cup EUR SF: NED vs URS',
               'Davis Cup EUR SF: NZL vs IND',
               'Davis Cup EUR SF: NZL vs JPN',
               'Davis Cup EUR SF: PAK vs THA',
               'Davis Cup EUR SF: POL vs URS',
               'Davis Cup EUR SF: POR vs AUT',
               'Davis Cup EUR SF: ROU vs EGY',
               'Davis Cup EUR SF: ROU vs FRA',
               'Davis Cup EUR SF: ROU vs GBR',
               'Davis Cup EUR SF: ROU vs ITA',
               'Davis Cup EUR SF: ROU vs NZL',
               'Davis Cup EUR SF: ROU vs SWE',
               'Davis Cup EUR SF: ROU vs YUG',
               'Davis Cup EUR SF: RSA vs COL',
               'Davis Cup EUR SF: RSA vs ITA',
               'Davis Cup EUR SF: SUI vs NED',
               'Davis Cup EUR SF: SWE vs CHI',
               'Davis Cup EUR SF: SWE vs ESP',
               'Davis Cup EUR SF: SWE vs FRG',
               'Davis Cup EUR SF: SWE vs ITA',
               'Davis Cup EUR SF: TCH vs AUS',
               'Davis Cup EUR SF: TCH vs FRA',
               'Davis Cup EUR SF: TCH vs FRG',
               'Davis Cup EUR SF: TCH vs HUN',
               'Davis Cup EUR SF: TCH vs ROU',
               'Davis Cup EUR SF: TCH vs URS',
               'Davis Cup EUR SF: THA vs NZL',
               'Davis Cup EUR SF: URS vs AUT',
               'Davis Cup EUR SF: URS vs ESP',
               'Davis Cup EUR SF: URS vs FRA',
               'Davis Cup EUR SF: URS vs ITA',
               'Davis Cup EUR SF: URS vs SWE',
               'Davis Cup EUR SF: URS vs TCH',
               'Davis Cup EUR SF: URS vs YUG',
               'Davis Cup EUR SF: USA vs ESP',
               'Davis Cup EUR SF: YUG vs ROU',
               'Davis Cup Eastern QF: CHN vs HKG',
               'Davis Cup Eastern QF: CHN vs KOR',
               'Davis Cup Eastern QF: HKG vs CHN',
               'Davis Cup Eastern QF: JPN vs PHI',
               'Davis Cup Eastern QF: NZL vs TPE',
               'Davis Cup Eastern QF: THA vs PHI',
               'Davis Cup Eastern SF: JPN vs CHN',
               'Davis Cup Eastern SF: NZL vs CHN',
               'Davis Cup Finals F: AUS vs ITA',
               'Davis Cup Finals F: CAN vs AUS',
               'Davis Cup Finals F: CAN vs ESP',
               'Davis Cup Finals F: RTF vs CRO',
               'Davis Cup Finals QF: ARG vs ESP',
               'Davis Cup Finals QF: AUS vs CAN',
               'Davis Cup Finals QF: AUS vs NED',
               'Davis Cup Finals QF: CAN vs FIN',
               'Davis Cup Finals QF: CRO vs ESP',
               'Davis Cup Finals QF: CZE vs AUS',
               'Davis Cup Finals QF: GBR vs GER',
               'Davis Cup Finals QF: GER vs CAN',
               'Davis Cup Finals QF: ITA vs CRO',
               'Davis Cup Finals QF: ITA vs NED',
               'Davis Cup Finals QF: ITA vs USA',
               'Davis Cup Finals QF: RTF vs SWE',
               'Davis Cup Finals QF: SRB vs GBR',
               'Davis Cup Finals QF: SRB vs KAZ',
               'Davis Cup Finals QF: SRB vs RUS',
               'Davis Cup Finals RR: ARG vs CHI',
               'Davis Cup Finals RR: ARG vs CRO',
               'Davis Cup Finals RR: ARG vs GER',
               'Davis Cup Finals RR: ARG vs SWE',
               'Davis Cup Finals RR: AUS vs BEL',
               'Davis Cup Finals RR: AUS vs COL',
               'Davis Cup Finals RR: AUS vs FRA',
               'Davis Cup Finals RR: AUS vs GBR',
               'Davis Cup Finals RR: AUS vs HUN',
               'Davis Cup Finals RR: AUS vs SUI',
               'Davis Cup Finals RR: BEL vs AUS',
               'Davis Cup Finals RR: BEL vs COL',
               'Davis Cup Finals RR: CAN vs CHI',
               'Davis Cup Finals RR: CAN vs ITA',
               'Davis Cup Finals RR: CAN vs KAZ',
               'Davis Cup Finals RR: CAN vs SWE',
               'Davis Cup Finals RR: CRO vs AUS',
               'Davis Cup Finals RR: CRO vs ESP',
               'Davis Cup Finals RR: CRO vs FIN',
               'Davis Cup Finals RR: CRO vs HUN',
               'Davis Cup Finals RR: CRO vs NED',
               'Davis Cup Finals RR: CRO vs RUS',
               'Davis Cup Finals RR: CRO vs SWE',
               'Davis Cup Finals RR: CRO vs USA',
               'Davis Cup Finals RR: CZE vs KOR',
               'Davis Cup Finals RR: ESP vs CAN',
               'Davis Cup Finals RR: ESP vs CZE',
               'Davis Cup Finals RR: ESP vs ECU',
               'Davis Cup Finals RR: ESP vs KOR',
               'Davis Cup Finals RR: ESP vs RTF',
               'Davis Cup Finals RR: ESP vs RUS',
               'Davis Cup Finals RR: ESP vs SRB',
               'Davis Cup Finals RR: FRA vs AUS',
               'Davis Cup Finals RR: FRA vs BEL',
               'Davis Cup Finals RR: FRA vs CZE',
               'Davis Cup Finals RR: FRA vs GBR',
               'Davis Cup Finals RR: FRA vs GER',
               'Davis Cup Finals RR: FRA vs JPN',
               'Davis Cup Finals RR: FRA vs SRB',
               'Davis Cup Finals RR: FRA vs SUI',
               'Davis Cup Finals RR: GBR vs CZE',
               'Davis Cup Finals RR: GBR vs FRA',
               'Davis Cup Finals RR: GBR vs KAZ',
               'Davis Cup Finals RR: GBR vs NED',
               'Davis Cup Finals RR: GBR vs SUI',
               'Davis Cup Finals RR: GER vs AUS',
               'Davis Cup Finals RR: GER vs AUT',
               'Davis Cup Finals RR: GER vs BEL',
               'Davis Cup Finals RR: GER vs CHI',
               'Davis Cup Finals RR: ITA vs ARG',
               'Davis Cup Finals RR: ITA vs CAN',
               'Davis Cup Finals RR: ITA vs CHI',
               'Davis Cup Finals RR: ITA vs COL',
               'Davis Cup Finals RR: ITA vs CRO',
               'Davis Cup Finals RR: ITA vs SWE',
               'Davis Cup Finals RR: KAZ vs NED',
               'Davis Cup Finals RR: KAZ vs SWE',
               'Davis Cup Finals RR: KOR vs CAN',
               'Davis Cup Finals RR: KOR vs SRB',
               'Davis Cup Finals RR: NED vs FIN',
               'Davis Cup Finals RR: NED vs USA',
               'Davis Cup Finals RR: RTF vs ECU',
               'Davis Cup Finals RR: SRB vs AUT',
               'Davis Cup Finals RR: SRB vs CAN',
               'Davis Cup Finals RR: SRB vs CZE',
               'Davis Cup Finals RR: SRB vs GER',
               'Davis Cup Finals RR: SRB vs JPN',
               'Davis Cup Finals RR: SRB vs KOR',
               'Davis Cup Finals RR: SWE vs CHI',
               'Davis Cup Finals RR: USA vs CAN',
               'Davis Cup Finals RR: USA vs COL',
               'Davis Cup Finals RR: USA vs FIN',
               'Davis Cup Finals RR: USA vs GBR',
               'Davis Cup Finals RR: USA vs ITA',
               'Davis Cup Finals RR: USA vs KAZ',
               'Davis Cup Finals RR: USA vs NED',
               'Davis Cup Finals SF: AUS vs CRO',
               'Davis Cup Finals SF: CRO vs SRB',
               'Davis Cup Finals SF: FIN vs AUS',
               'Davis Cup Finals SF: GBR vs ESP',
               'Davis Cup Finals SF: ITA vs CAN',
               'Davis Cup Finals SF: ITA vs SRB',
               'Davis Cup Finals SF: RTF vs GER',
               'Davis Cup Finals SF: RUS vs CAN',
               'Davis Cup G1 F: ARG vs USA',
               'Davis Cup G1 F: AUT vs GBR',
               'Davis Cup G1 F: CHI vs USA',
               'Davis Cup G1 F: INA vs IND',
               'Davis Cup G1 F: INA vs JPN',
               'Davis Cup G1 F: INA vs KOR',
               'Davis Cup G1 F: MAS vs IND',
               'Davis Cup G1 F: URS vs NED',
               'Davis Cup G1 F: USA vs CHI',
               'Davis Cup G1 PO: ARG vs ECU',
               'Davis Cup G1 PO: ARG vs VEN',
               'Davis Cup G1 PO: AUT vs DEN',
               'Davis Cup G1 PO: AUT vs UKR',
               'Davis Cup G1 PO: BAH vs CAN',
               'Davis Cup G1 PO: BAH vs COL',
               'Davis Cup G1 PO: BAH vs MEX',
               'Davis Cup G1 PO: BAH vs VEN',
               'Davis Cup G1 PO: BAR vs ECU',
               'Davis Cup G1 PO: BAR vs URU',
               'Davis Cup G1 PO: BLR vs GEO',
               'Davis Cup G1 PO: BLR vs MKD',
               'Davis Cup G1 PO: BLR vs NED',
               'Davis Cup G1 PO: BLR vs POL',
               'Davis Cup G1 PO: BLR vs POR',
               'Davis Cup G1 PO: BLR vs SVK',
```

Loading [MathJax]/extensions/Safe.js

```
 'Davis Cup G1 PO: BRA vs CHI',
 'Davis Cup G1 PO: BRA vs PER',
 'Davis Cup G1 PO: BUL vs ROU',
 'Davis Cup G1 PO: CAN vs ARG',
 'Davis Cup G1 PO: CAN vs CHI',
 'Davis Cup G1 PO: CAN vs DOM',
 'Davis Cup G1 PO: CAN vs VEN',
 'Davis Cup G1 PO: CHI vs CAN',
 'Davis Cup G1 PO: CHI vs COL',
 'Davis Cup G1 PO: CHI vs PER',
 'Davis Cup G1 PO: CHI vs URU',
 'Davis Cup G1 PO: CHN vs INA',
 'Davis Cup G1 PO: CHN vs PAK',
 'Davis Cup G1 PO: CHN vs THA',
 'Davis Cup G1 PO: CHN vs TPE',
 'Davis Cup G1 PO: COL vs ARG',
 'Davis Cup G1 PO: COL vs MEX',
 'Davis Cup G1 PO: CRO vs POR',
 'Davis Cup G1 PO: DEN vs ESP',
 'Davis Cup G1 PO: DOM vs CHI',
 'Davis Cup G1 PO: ECU vs BAH',
 'Davis Cup G1 PO: ECU vs MEX',
 'Davis Cup G1 PO: ECU vs PER',
 'Davis Cup G1 PO: FIN vs HUN',
 'Davis Cup G1 PO: FIN vs NOR',
 'Davis Cup G1 PO: FIN vs POL',
 'Davis Cup G1 PO: GBR vs ISR',
 'Davis Cup G1 PO: GBR vs POL',
 'Davis Cup G1 PO: GBR vs ROU',
 'Davis Cup G1 PO: GHA vs IRL',
 'Davis Cup G1 PO: HKG vs TPE',
 'Davis Cup G1 PO: HUN vs BEL',
 'Davis Cup G1 PO: HUN vs MAR',
 'Davis Cup G1 PO: HUN vs UKR',
 'Davis Cup G1 PO: INA vs CHN',
 'Davis Cup G1 PO: INA vs KOR',
 'Davis Cup G1 PO: INA vs LIB',
 'Davis Cup G1 PO: INA vs NZL',
 'Davis Cup G1 PO: INA vs TPE',
 'Davis Cup G1 PO: IND vs CHN',
 ...]
```

In [116…

```python
def replace_tourney_name(df):
    mask1 = df['tourney_name'].str.contains('Davis Cup')
    df.loc[mask1, 'tourney_name'] = 'Davis Cup'

    mask2 = df['tourney_name'].str.contains('Olympics')
    df.loc[mask2, 'tourney_name'] = 'Olympics'

    mask3 = df['tourney_name'].str.contains('Australian Open')
    df.loc[mask3, 'tourney_name'] = 'Australian Open'

    mask4 = df['tourney_name'].str.contains('Adelaide')
    df.loc[mask4, 'tourney_name'] = 'Adelaide'

    mask5 = df['tourney_name'].str.contains('Washington')
    df.loc[mask5, 'tourney_name'] = 'Washington'

    mask6 = df['tourney_name'].str.contains('Barcelona')
    df.loc[mask6, 'tourney_name'] = 'Barcelona'

    mask7 = df['tourney_name'].str.contains('Belgrade')
    df.loc[mask7, 'tourney_name'] = 'Belgrade'
```

Loading [MathJax]/extensions/Safe.js

```
    mask8 = df['tourney_name'].str.contains('Cincinnati')
    df.loc[mask8, 'tourney_name'] = 'Cincinnati'

    mask9 = df['tourney_name'].str.contains('Hamburg')
    df.loc[mask9, 'tourney_name'] = 'Hamburg'

    mask10 = df['tourney_name'].str.contains('Indian Wells')
    df.loc[mask10, 'tourney_name'] = 'Indian Wells'

    mask11 = df['tourney_name'].str.contains('Madrid')
    df.loc[mask11, 'tourney_name'] = 'Madrid'

    mask12 = df['tourney_name'].str.contains('Miami')
    df.loc[mask12, 'tourney_name'] = 'Miami'

    mask13 = df['tourney_name'].str.contains('Monte Carlo')
    df.loc[mask13, 'tourney_name'] = 'Monte Carlo'

    mask14 = df['tourney_name'].str.contains('New York')
    df.loc[mask14, 'tourney_name'] = 'New York'

    mask15 = df['tourney_name'].str.contains('Rome')
    df.loc[mask15, 'tourney_name'] = 'Rome'

    mask16 = df['tourney_name'].str.contains('Shanghai')
    df.loc[mask16, 'tourney_name'] = 'Shanghai'

    mask17 = df['tourney_name'].str.contains('Toronto')
    df.loc[mask17, 'tourney_name'] = 'Toronto'

    mask18 = df['tourney_name'].str.contains('Janeiro')
    df.loc[mask18, 'tourney_name'] = 'Rio de Janeiro'

    return df
```

In [117…

```
df = replace_tourney_name(df)
```

Handle missing values

For surface

In [118…

```
df[df['surface'].isna()]['tourney_name'].unique()
```

Out[118]:

```
array(['Davis Cup', 'Perth', 'Bristol', 'Manchester', 'Cape Town',
       'Mexico City', 'Calcutta', 'Port Elizabeth', 'San Juan', 'Phoenix',
       'Cannes', 'Los Angeles SoCal Chps', 'Oakland', 'Woodside',
       'Helsinki', 'Saltsjoebaden', 'San Antonio Collegiate',
       'Montana Vermala', 'Senigallia', 'Quebec City', 'Bucharest',
       'Mamaia', 'Las Vegas', 'Madrid', 'Valencia', 'Cannes Chps',
       'Hanau', 'Tournament of Champions WCT', 'Roanoke', 'Dublin',
       'Cedar Grove', 'Christchurch', 'Oslo', 'New Orleans WCT', 'Omaha',
       'Paramus', 'Calgary', 'Salt Lake City', 'Washington', 'Tokyo WCT',
       'Tokyo', 'Freeport', 'Shreveport', 'New York', 'Istanbul',
       'Quebec WCT', 'Vancouver WCT', 'Casablanca WCT', 'Seattle',
       'Tanglewood', 'Montreal WCT', 'Alamo WCT', 'Los Angeles',
       'Des Moines', 'Kansas City', 'Sacramento', 'Quebec', 'New Delhi',
       'Djkarta'], dtype=object)
```

In [119…

```
# Repérer les valeurs vides dans la colonne 'surface'
missing_surface = df[df['surface'].isnull()]

# Parcourir les lignes avec une surface manquante
```

```
for index, row in missing_surface.iterrows():
    # Récupérer la valeur de 'tourney_name' pour cette ligne
    tourney_name = row['tourney_name']

    # Trouver les lignes avec le même 'tourney_name' qui ont une valeur de 'surface'
    matching_rows = df[(df['tourney_name'] == tourney_name) & ~(df['surface'].isnull())]

    # S'il y a des lignes correspondantes, prendre la première valeur de 'surface' et la
    if not matching_rows.empty:
        surface_value = matching_rows.iloc[0]['surface']
        df.at[index, 'surface'] = surface_value
```

For winner height

In [120…

```
# Repérer les valeurs vides dans la colonne 'winner_ht'
missing_winner_ht = df[df['winner_ht'].isnull()]

# Parcourir les lignes avec une 'winner_ht' manquante
for index, row in missing_winner_ht.iterrows():
    # Récupérer la valeur de 'winner_id' pour cette ligne
    winner_id = row['winner_id']

    # Rechercher la ligne correspondante dans le DataFrame 'players'
    player_info = players[players['player_id'] == winner_id]

    # Si la ligne correspondante existe dans le DataFrame 'players' et qu'elle a une val
    if not player_info.empty and not pd.isnull(player_info.iloc[0]['height']):
        # Récupérer la valeur de 'height' et remplacer la valeur NaN dans 'winner_ht'
        height_value = player_info.iloc[0]['height']
        df.at[index, 'winner_ht'] = height_value
```

For loser height

In [121…

```
# Repérer les valeurs vides dans la colonne 'winner_ht'
missing_loser_ht = df[df['loser_ht'].isnull()]

# Parcourir les lignes avec une 'winner_ht' manquante
for index, row in missing_loser_ht.iterrows():
    # Récupérer la valeur de 'winner_id' pour cette ligne
    loser_id = row['loser_id']

    # Rechercher la ligne correspondante dans le DataFrame 'players'
    player_info = players[players['player_id'] == loser_id]

    # Si la ligne correspondante existe dans le DataFrame 'players' et qu'elle a une val
    if not player_info.empty and not pd.isnull(player_info.iloc[0]['height']):
        # Récupérer la valeur de 'height' et remplacer la valeur NaN dans 'winner_ht'
        height_value = player_info.iloc[0]['height']
        df.at[index, 'loser_ht'] = height_value
```

For loser age

In [122…

```
# Convertir 'dob' en datetime si ce n'est pas déjà fait
players['dob'] = pd.to_datetime(players['dob'])

df['tourney_date'] = pd.to_datetime(df['tourney_date'])

# Calculer l'âge à partir de 'tourney_date' et 'dob'
def calculate_age(row):
    if pd.isnull(row['loser_age']):
        loser_id = row['loser_id']
```

Loading [MathJax]/extensions/Safe.js

```
        player_info = players[players['player_id'] == loser_id]
        if not player_info.empty and not pd.isnull(player_info.iloc[0]['dob']):
            dob = player_info.iloc[0]['dob']
            tourney_date = row['tourney_date']
            return (tourney_date - dob).days // 365
    return row['loser_age']

# Appliquer la fonction calculate_age pour remplir les valeurs manquantes de 'loser_age'
df['loser_age'] = df.apply(calculate_age, axis=1)
```

Drop remaining NaN

In [123…

```
df=df.dropna(subset=['surface','winner_ht','winner_age','loser_ht','loser_age'])
```

Second column filter to keep only the essentials

In [125…

```
df=df[['tourney_name','surface','winner_id','winner_hand','winner_ht','winner_age',
    'loser_id','loser_hand','loser_ht','loser_age','round']]
```

Creation of 2 dataframes : one for victories and one for losses

By doing this, I duplicate my number of columns and put myself in a player versus opponent perspective. Each match will be interpreted from the winner's point of view as a victory and then from the loser's point of view as a defeat.

In [147…

```
df_winner = df.copy()
df_loser = df.copy()
```

In [148…

```
df.columns
```

Out[148]:

```
Index(['tourney_name', 'surface', 'winner_id', 'winner_hand', 'winner_ht',
       'winner_age', 'loser_id', 'loser_hand', 'loser_ht', 'loser_age',
       'round'],
      dtype='object')
```

Rename columns to have one player and his opponent

In [149…

```
df_winner=df_winner.rename(columns={'winner_id':'Player','winner_hand':'Player_Hand','wi
                         'winner_age':'Player_Age','loser_id':'Opponent','loser_hand':'O
                          'loser_ht':'Opponent_Height','loser_age':'Opponent_Age'})
```

Players here are the winners so Result=1

In [150…

```
df_winner['Result']=1
```

Rename columns to have one player and his opponent

In [151…

```
df_loser=df_loser.rename(columns={'loser_id':'Player','loser_hand':'Player_Hand','loser_
                         'loser_age':'Player_Age','winner_id':'Opponent','winner_hand':'
                          'winner_ht':'Opponent_Height','winner_age':'Opponent_Age'})
```

Players here are the losers so Result=0

Loading [MathJax]/extensions/Safe.js

```
In [152…  df_loser['Result']=0
```

```
In [153…  df_loser=df_loser[['tourney_name','surface','Player','Player_Hand','Player_Height','Play
```

Concat to get the final dataframe

```
In [158…  df=pd.concat([df_winner,df_loser])
```

Instead of leaving two columns for the same feature, create a differential.

```
In [160…  df
```

Out[160]:

	tourney_name	surface	Player	Player_Hand	Player_Height	Player_Age	Opponent	Opponent_Hand
0	Brisbane	Hard	105453	R	178.0	29.0	106421	R
1	Brisbane	Hard	106421	R	198.0	22.8	104542	R
2	Brisbane	Hard	105453	R	178.0	29.0	104871	R
3	Brisbane	Hard	104542	R	188.0	33.7	200282	R
4	Brisbane	Hard	106421	R	198.0	22.8	105683	R
...	...	...	...	...	...	...	...	...
191915	Tour Finals	Hard	105453	R	178.0	24.8	104925	R
191916	Tour Finals	Hard	103819	R	185.0	33.2	104925	R
191917	Davis Cup	Clay	104542	R	188.0	29.5	104527	R
191918	Davis Cup	Clay	103819	R	185.0	33.2	104792	R
191919	Davis Cup	Clay	104755	R	185.0	28.4	103819	R

311238 rows × 12 columns

```
In [161…  df['Hand_Opposition'] = df['Player_Hand'] + ' vs ' + df['Opponent_Hand']
```

```
In [162…  df['Height_Differential'] = df['Player_Height']-df['Opponent_Height']
```

```
In [163…  df['Age_Difference'] = df['Player_Age']-df['Opponent_Age']
```

```
In [165…  df=df[['tourney_name','surface','Player','Opponent','Hand_Opposition','Height_Differenti
```

```
In [168…  df.to_csv('df.csv')
```

Players reprocessing

Sélection des joueurs présents dans top500

Add Full Name

```
In [166…  players['Name'] = players['name_first'] + ' ' + players['name_last']
```

Loading [MathJax]/extensions/Safe.js

Keep only the players in the ATP Top 500 beginning of 2024

In [167…

```
players = players[players['Name'].isin(top500['Player'])]
```

In [171…

```
players=players.rename(columns={'Name':'Player'})
```

Get full data : Players characteristics and Rankings from ATP

In [172…

```
full_data=pd.merge(players,top500,on='Player',how='left')
```

In [173…

```
full_data
```

Out[173]:

	player_id	name_first	name_last	hand	dob	ioc	height	wikidata_id	Player	Rank
0	100644	Alexander	Zverev	R	1970-01-01 00:00:00.019970420	GER	198.0	Q13990552	Alexander Zverev	6
1	104527	Stan	Wawrinka	R	1970-01-01 00:00:00.019850328	SUI	183.0	Q193661	Stan Wawrinka	74
2	104755	Richard	Gasquet	R	1970-01-01 00:00:00.019860618	FRA	185.0	Q209436	Richard Gasquet	118
3	104792	Gael	Monfils	R	1970-01-01 00:00:00.019860901	FRA	193.0	Q186429	Gael Monfils	54
4	104918	Andy	Murray	R	1970-01-01 00:00:00.019870515	GBR	190.0	Q10125	Andy Murray	67
...	...	...	...	...	...	...	...	...	...	...
185	210097	Ben	Shelton	L	1970-01-01 00:00:00.020021009	USA	NaN	Q108532383	Ben Shelton	17
186	210136	Mark	Lajal	R	1970-01-01 00:00:00.020030512	EST	NaN	NaN	Mark Lajal	195
187	210150	Jakub	Mensik	U	1970-01-01 00:00:00.020050901	CZE	NaN	Q102228055	Jakub Mensik	87
188	210234	Juncheng	Shang	L	1970-01-01 00:00:00.020050101	CHN	NaN	Q106466705	Juncheng Shang	139
189	210506	Alex	Michelsen	U	1970-01-01 00:00:00.020040825	USA	NaN	NaN	Alex Michelsen	76

190 rows × 11 columns

In [174…

```
full_data.to_csv('players_information.csv')
```

In [403…

```
to_predict = full_data[['player_id','Player','height','hand','dob','Rank','Official Poin
```

In [404…

```
to_predict=to_predict.dropna(subset=['height'])
```

In [405…

```
to_predict['dob'] = to_predict['dob'].apply(convert_to_date)
```

Encodage des variables textuelles devant avoir les mêmes valeurs entre colonnes

In [433…

```
# Créer un ensemble de tous les ID uniques
unique_ids = set(df['Player_1_id']).union(set(df['Player_2_id'])).union(set(df['Winner']
# Créer un dictionnaire pour mapper chaque ID unique à un nombre unique
id_to_number = {id_: i for i, id_ in enumerate(unique_ids)}

# Remplacer les valeurs des colonnes Player_1_id, Player_2_id et Winner par les nombres
df['Player_1_id'] = df['Player_1_id'].map(id_to_number)
df['Player_2_id'] = df['Player_2_id'].map(id_to_number)
df['Winner'] = df['Winner'].map(id_to_number)
```

In [434…

```
df
```

Out[434]:

	tourney_name	surface	Player_1_id	Player_2_id	Hand_Opposition	Height_Differential	Age_Difference
0	Canada Masters	Hard	1724	1926	R vs R	-21.0	6.3
8	Stuttgart	Grass	823	818	R vs R	13.0	-0.1
11	Metz	Hard	205	537	R vs R	-5.0	4.9
13	Chennai	Hard	907	130	L vs R	-5.0	-9.8
14	Mexico City	Clay	159	17	R vs R	0.0	-2.4
...	...	...	...	...	...	...	...
210737	Stuttgart	Clay	221	986	R vs R	10.0	9.1
210738	Washington	Hard	1995	259	R vs R	15.0	0.0
210739	Toulouse	Hard	2023	1812	R vs R	-5.0	-2.1
210740	Australian Open	Hard	626	771	R vs R	5.0	1.4
210741	Metz	Hard	893	912	R vs R	5.0	0.2

112951 rows × 10 columns

In [435…

```
numeric = ['Height_Differential','Age_Difference', 'Ranking_Gap', 'Ranking_Points_Differ
string = ['tourney_name','surface','Hand_Opposition']
```

In [436…

```
for col in df.columns:
    if col in numeric:
        # Standardisation des données numériques
        scaler = StandardScaler()
        df[col] = scaler.fit_transform(df[[col]])
    elif col in string:
        # Encodage des données de chaîne
        encoder = LabelEncoder()
        df[col] = encoder.fit_transform(df[col])
```

Remise en forme de Winner

In [452…

```
df['Winner'] = df['Winner'].replace({29: 0, 378: 1})
```

Récupération de df

In [453…

```
df
```

Out[453]:

	tourney_name	surface	Player_1_id	Player_2_id	Hand_Opposition	Height_Differential	Age_Difference
0	51	3	1724	1926	8	-2.284599	1.221184
8	1857	2	823	818	8	1.424725	-0.023975
11	1789	3	205	537	8	-0.539035	0.948806
13	55	3	907	130	4	-0.539035	-1.911170
14	1790	1	159	17	8	0.006454	-0.471454
...	...	...	...	...	...	...	...
210737	1857	1	221	986	8	1.097432	1.765942
210738	1886	3	1995	259	8	1.642921	-0.004520
210739	1876	3	2023	1812	8	-0.539035	-0.413088
210740	17	3	626	771	8	0.551943	0.267859
210741	1789	3	893	912	8	0.551943	0.034392

112951 rows × 10 columns

In [454…

```
data = df.copy()
```

In [455…

```
data.to_csv('data.csv')
```

In [456…

```
df = data
```

Données d'entraînement et de test

In [457…

```
X=df.drop(columns=['Winner'])
y=df['Winner']
```

In [458…

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=123)
```

1er modèle : RandomForest

In [459…

```
rdf = RandomForestClassifier(n_estimators=100, random_state=42)
```

In [460…

```
rdf.fit(X_train,y_train)
```

Out[460]:

```
▼        RandomForestClassifier
RandomForestClassifier(random_state=42)
```

In [461…

```
y_pred = rdf.predict(X_test)
```

In [462…

```
print('Classification report:\n', classification_report(y_test, y_pred))
```

```
Classification report:
               precision    recall  f1-score   support

           0       0.73      0.76      0.74     12509
           1       0.68      0.66      0.67     10082

    accuracy                           0.71     22591
   macro avg       0.71      0.71      0.71     22591
weighted avg       0.71      0.71      0.71     22591
```

In [465…

```
importances = rdf.feature_importances_
importances
```

Out[465]:

```
array([0.13019673, 0.03287846, 0.14151982, 0.14300902, 0.02881643,
       0.09643659, 0.14001425, 0.14611234, 0.14101636])
```

In [475…

```
param_grid = {
    'n_estimators': [50, 100, 150],
    'max_depth': [None, 10, 20],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4],
    'bootstrap': [True, False]
}
```

In [476…

```
# Instancier RandomizedSearchCV pour effectuer la recherche aléatoire sur la grille d'hy
rdf_up = RandomizedSearchCV(estimator=rdf, param_distributions=param_grid, n_iter=50, cv
```

In [477…

```
rdf_up.fit(X_train,y_train)
```

Out[477]:

```
▸        RandomizedSearchCV
▸ estimator: RandomForestClassifier
      ▸ RandomForestClassifier
```

In [478…

```
y_pred = rdf_up.predict(X_test)
```

In [479…

```
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)

# Afficher les performances
print("Accuracy: {:.2f}".format(accuracy))
print("Precision: {:.2f}".format(precision))
print("Recall: {:.2f}".format(recall))
```

```
Accuracy: 0.71
Precision: 0.67
Recall: 0.67
```

2e modèle : xgBoost

In [466…

```
# Créer un objet DMatrix pour l'entraînement et le test
dtrain = xgb.DMatrix(X_train, label=y_train)
dtest = xgb.DMatrix(X_test, label=y_test)
```

In [467…

```
# Définir les paramètres du modèle
params = {
    'objective': 'binary:logistic',  # pour un problème de classification binaire
    'eval_metric': ['logloss', 'error'],  # métriques d'évaluation
    'max_depth': 3,  # profondeur maximale de l'arbre
    'eta': 0.1,  # taux d'apprentissage
    'gamma': 0.5  # paramètre de réduction de perte minimale
}
```

In [468…

```
# Entraîner le modèle
num_rounds = 100  # nombre d'itérations
model = xgb.train(params, dtrain, num_rounds)
```

In [469…

```
# Faire des prédictions sur l'ensemble de test
y_pred = model.predict(dtest)
y_pred_binary = [1 if p >= 0.5 else 0 for p in y_pred]  # conversion en classes binaires
```

In [470…

```
accuracy = accuracy_score(y_test, y_pred_binary)
precision = precision_score(y_test, y_pred_binary)
recall = recall_score(y_test, y_pred_binary)

# Afficher les performances
print("Accuracy: {:.2f}".format(accuracy))
print("Precision: {:.2f}".format(precision))
print("Recall: {:.2f}".format(recall))
```

```
Accuracy: 0.66
Precision: 0.62
Recall: 0.60
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [471…

```
to_predict
```

Out[471]:

	player_id	Player	height	hand	dob	Rank	Official Points
0	100644	Alexander Zverev	198.0	R	1997-04-20	6	5085
1	104527	Stan Wawrinka	183.0	R	1985-03-28	74	797
2	104755	Richard Gasquet	185.0	R	1986-06-18	118	538
3	104792	Gael Monfils	193.0	R	1986-09-01	54	937
4	104918	Andy Murray	190.0	R	1987-05-15	67	845
...	...	...	...	...	...	...	...
168	208502	Dominic Stricker	180.0	L	2002-08-16	112	557
171	209070	Arthur Cazaux	183.0	R	2002-08-23	86	701
172	209098	Hamad Medjedovic	185.0	R	2003-07-18	116	541
180	209857	Leandro Riedi	191.0	R	2002-01-27	160	396
181	209950	Arthur Fils	185.0	R	2004-06-12	44	1028

146 rows × 7 columns

In [63]:

```
df
```

```
Out[63]:
         tourney_id  surface  tourney_level  winner_id  winner_hand  winner_age  loser_id  loser_hand  loser_ht
      0        7825        3              0     105453            2    0.821557    106421           2   2.062667
      1        7825        3              0     106421            2   -0.708139    104542           2   0.560875
      2        7825        3              0     105453            2    0.821557    104871           2   0.560875
      3        7825        3              0     104542            2    1.981165    200282           2  -0.190021
      4        7825        3              0     106421            2   -0.708139    105683           2   1.762309
    ...         ...      ...            ...        ...          ...         ...       ...         ...        ...
 191915        7056        3              2     104925            2    0.426797    105453           2  -0.940917
 191916        7056        3              2     104925            2    0.426797    103819           2   0.110337
 191917        7084        1              1     104527            2    0.969592    104542           2   0.560875
 191918        7084        1              1     104792            2    0.624177    103819           2   0.110337
 191919        7084        1              1     103819            2    1.857802    104755           2   0.110337

102934 rows × 15 columns
```

```
In [ ]:
```

Loading [MathJax]/extensions/Safe.js