Capstone 1: Analytic Report and Research Proposal- ATP matches 2019

1.0 Background and Data information

1.1 ATP Tour Background

ATP stands for Association of Tennis Professionals and is the "main men's tennis governing body" and the “official ranking system in men’s professional tennis.” It has been around since 1973, ranking players in a ranking system that will show the players performance and allow the player to enter different tournament world wide.

There are different tournaments including the ATP finals, and there is the ranking of each individual player. Player with the most points have a higher rank. Points are acquired by playing in each tournament, and most importantly the ranking of the players are changed every so often. That is because the points are replace every year from the following year. Example, if a player had low ranking points the previous year, they have the opportunity the following year to get a higher rank by winning more points and winning more matches.

1.2 The Data: “ ATP_MATCHES_2019”

This is a data of the "ATP matches 2019", which cover the ATP tennis matches from Dec 2018 to Feb 2019. The tourney are in: Brisbane, Doha, Pune, Auckland, Sydney, Australian Open, David Cup QLS R1, Cordoba, Montepellier, Sofia, Buenos Aires, New York, Rotterdam, Delray Beach, Marseille, Rio de Janeiro, Acapulco, Dubai, and Sao Paulo.

We will take a look at the:

- The data it self

- My reaserch questions
  - 2.0 Does the age of a player have an effect on the possibilty between winning or losing?
  - 3.0 Correlation between age and ranking
  - 4.0 Compare each tourney's highest and lowest ranking points: For Winners
- Conclusion
- Propose Further Research
- References

1.3 Where does the data come from?

This data is downloaded from GitHub, a Tennis dataset by Jeff Sackmann.

1.4 Here we can see how big the data is, the different columns, and what type of information is provided

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from collections import Counter
from itertools import groupby
import matplotlib.gridspec as gridspec
%matplotlib inline
```

In [2]:
```
df = pd.read_csv('atp_matches_2019.csv',usecols=['tourney_name','tourney_date','winner_name','winner_ioc
df
```

Out[2]:

	tourney_name	tourney_date	winner_name	winner_ioc	winner_age	loser_name	loser_ioc	loser_age	minutes	winner_rank	winner_r
0	Brisbane	20181231	Denis Kudla	USA	26.37	Taylor Harry Fritz	USA	21.17	144.0	63.0	
1	Brisbane	20181231	John Millman	AUS	29.55	Tennys Sandgren	USA	27.44	153.0	38.0	
2	Brisbane	20181231	Grigor Dimitrov	BUL	27.63	Yoshihito Nishioka	JPN	23.26	79.0	19.0	
3	Brisbane	20181231	Yasutaka Uchiyama	JPN	26.40	Ugo Humbert	FRA	20.51	90.0	185.0	
4	Brisbane	20181231	Jeremy Chardy	FRA	31.88	Jan Lennard Struff	GER	28.68	99.0	40.0	
...	...	...	...	...	...	...	...	...	...	...	...
646	Sao Paulo	20190225	Christian Garin	CHI	22.74	Leonardo Mayer	ARG	31.78	107.0	92.0	
647	Sao Paulo	20190225	Casper Ruud	NOR	20.18	Hugo Dellien	BOL	25.69	53.0	108.0	
648	Sao Paulo	20190225	Guido Pella	ARG	28.78	Laslo Djere	SRB	23.73	109.0	48.0	
649	Sao Paulo	20190225	Christian Garin	CHI	22.74	Casper Ruud	NOR	20.18	79.0	92.0	
650	Sao Paulo	20190225	Guido Pella	ARG	28.78	Christian Garin	CHI	22.74	83.0	48.0	

651 rows × 13 columns

In [3]:
```
df.shape
```

Out[3]:
```
(651, 13)
```

Analyze statistical values in the data

In [4]:
```
df.describe() #to see the numbers accross, and there are some null values
```

Out[4]:

	tourney_date	winner_age	loser_age	minutes	winner_rank	winner_rank_points	loser_rank	loser_rank_points
count	6.510000e+02	651.000000	651.000000	646.000000	648.000000	648.000000	639.000000	639.000000
mean	2.018901e+07	27.873318	27.767650	111.373065	63.771605	1462.862654	87.998435	970.103286
std	3.018114e+03	4.870999	4.730917	41.435166	53.982798	1626.540395	77.231068	938.011405
min	2.018123e+07	18.480000	17.230000	19.000000	1.000000	19.000000	1.000000	4.000000
25%	2.019011e+07	23.610000	23.720000	80.000000	24.000000	627.000000	41.000000	525.500000
50%	2.019020e+07	28.200000	28.010000	102.500000	53.000000	905.000000	72.000000	729.000000
75%	2.019022e+07	31.660000	31.260000	133.000000	90.000000	1560.000000	108.000000	1045.000000
max	2.019022e+07	39.880000	39.970000	305.000000	449.000000	9135.000000	593.000000	9045.000000

In [5]:
```
df[df.isnull().any(axis=1)] #looking for missing values in the data
```

Out[5]:

	tourney_name	tourney_date	winner_name	winner_ioc	winner_age	loser_name	loser_ioc	loser_age	minutes	winner_rank	winner_r
36	Doha	20181231	Guillermo Garcia Lopez	ESP	35.58	Mubarak Al Harrasi	QAT	23.72	61.0	105.0	
45	Doha	20181231	Marco Cecchinato	ITA	26.25	Guido Pella	ARG	28.62	NaN	20.0	
63	Pune	20181231	Steve Darcis	BEL	34.80	Roberto Carballes Baena	ESP	25.77	93.0	NaN	
73	Pune	20181231	Steve Darcis	BEL	34.80	Michael Mmoh	USA	20.97	189.0	NaN	
79	Pune	20181231	Steve Darcis	BEL	34.80	Malek Jaziri	TUN	34.95	93.0	NaN	
82	Pune	20181231	Ivo Karlovic	CRO	39.84	Steve Darcis	BEL	34.80	115.0	100.0	
191	Australian Open	20190114	Grigor Dimitrov	BUL	27.67	Janko Tipsarevic	SRB	34.56	153.0	21.0	
268	Davis Cup QLS R1: AUS vs BIH	20190201	Alexei Popyrin	AUS	19.49	Nerman Fatic	BIH	24.27	60.0	124.0	
282	Davis Cup QLS R1: CZE vs NED	20190201	Robin Haase	NED	31.82	Jiri Lehecka	CZE	17.23	118.0	54.0	
284	Davis Cup QLS R1: GER vs HUN	20190201	Alexander Zverev	GER	21.79	Peter Nagy	HUN	26.88	69.0	3.0	
286	Davis Cup QLS R1: GER vs HUN	20190201	Philipp Kohlschreiber	GER	35.30	David Szintai	HUN	21.65	98.0	32.0	
342	Montpellier	20190204	Jo Wilfried Tsonga	FRA	33.80	Ugo Humbert	FRA	20.61	NaN	210.0	
369	Sofia	20190204	Matthew Ebden	AUS	31.19	Adrian Andreev	BUL	17.73	139.0	45.0	
370	Sofia	20190204	Fernando Verdasco	ESP	35.22	Alexandar Lazarov	BUL	21.25	51.0	26.0	

	tourney_name	tourney_date	winner_name	winner_ioc	winner_age	loser_name	loser_ioc	loser_age	minutes	winner_rank	winner_r
383	Sofia	20190204	Marton Fucsovics	HUN	26.99	Roberto Bautista Agut	ESP	30.81	NaN	47.0	
389	Buenos Aires	20190211	Guido Pella	ARG	28.74	Francisco Cerundolo	ARG	20.50	95.0	50.0	
418	New York	20190211	Brayden Schnur	CAN	23.61	Jack Mingjie Lin	CAN	19.83	76.0	154.0	
560	Rio De Janeiro	20190218	Laslo Djere	SRB	23.72	Aljaz Bedene	SLO	29.59	NaN	90.0	
568	Acapulco	20190225	Mackenzie Mcdonald	USA	23.86	Emilio Nava	USA	17.23	52.0	71.0	
579	Acapulco	20190225	Alex De Minaur	AUS	20.02	Feliciano Lopez	ESP	37.43	NaN	26.0	

1.4.1 Drop all Null values, to get a more accurate output

In [6]:
```
df= df.dropna()
df.describe() #now the counts equall all across all columns
```

Out[6]:

	tourney_date	winner_age	loser_age	minutes	winner_rank	winner_rank_points	loser_rank	loser_rank_points
count	6.310000e+02	631.000000	631.00000	631.000000	631.000000	631.000000	631.000000	631.000000
mean	2.018906e+07	27.822884	27.84206	111.703645	63.622821	1468.088748	88.259905	971.036450
std	2.964424e+03	4.826948	4.65326	41.446458	53.988715	1632.357226	77.634097	942.720969
min	2.018123e+07	18.480000	18.22000	19.000000	1.000000	19.000000	1.000000	4.000000
25%	2.019011e+07	23.610000	23.75500	80.500000	24.000000	631.000000	41.000000	523.500000
50%	2.019020e+07	28.130000	28.10000	103.000000	53.000000	905.000000	72.000000	729.000000
75%	2.019022e+07	31.650000	31.27000	133.000000	89.000000	1572.500000	108.500000	1045.000000
max	2.019022e+07	39.880000	39.97000	305.000000	449.000000	9135.000000	593.000000	9045.000000

2.0 Does the age of a player have an effect on the possibilty between

winning or losing?

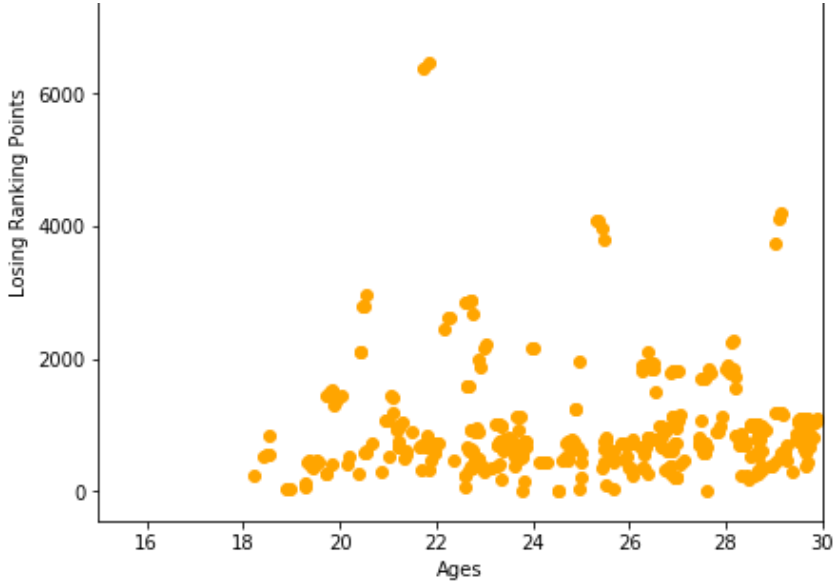
I think that everyone that looks at a player's age automatically think that the older the player is the more experience they have, and the more experience they have the greater the chances of winning. Or the younger they are, new talent more energy to the court.

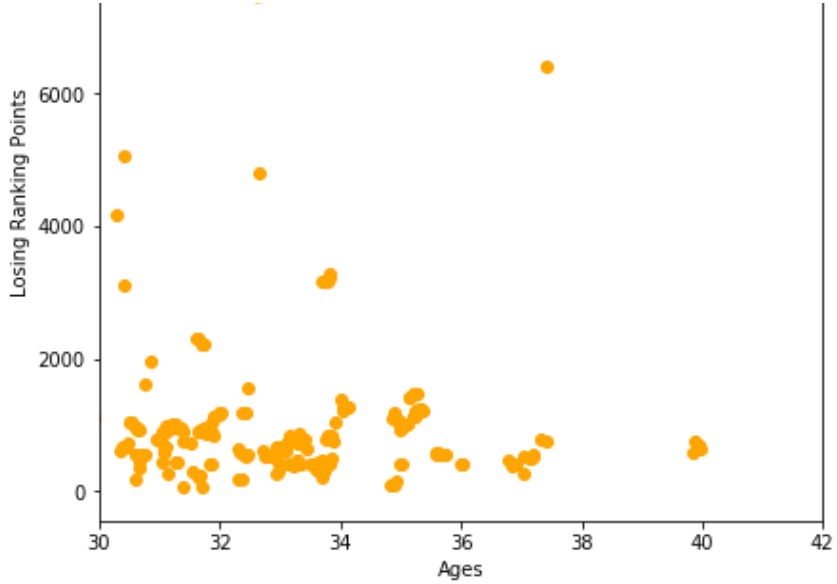
But what age in tennis represent a good chance of having a good ranking in tennis. Next, we will compare age with ranking points.

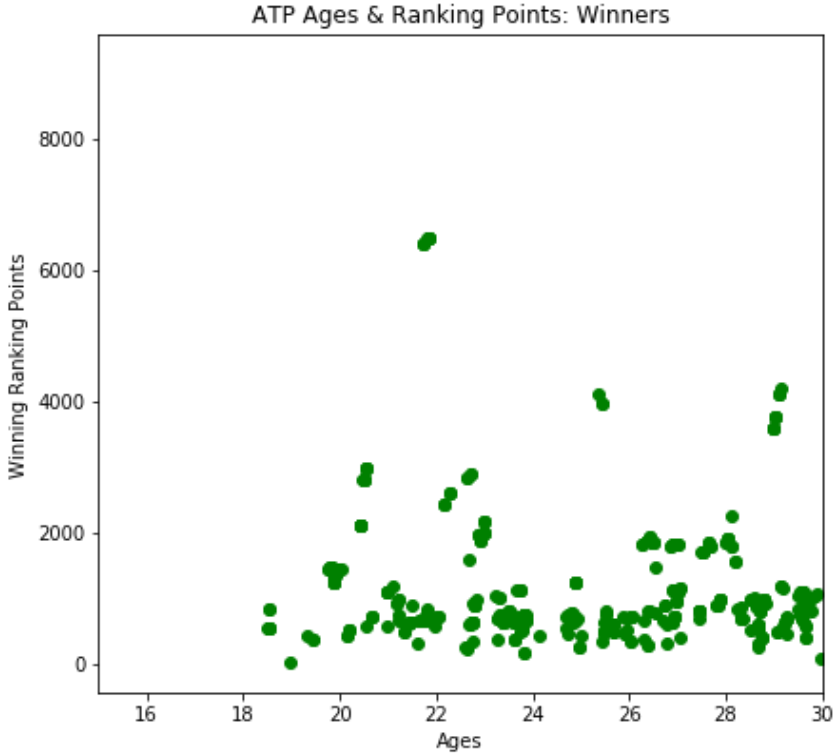
2.1 Scatter Graph: Age and Rank

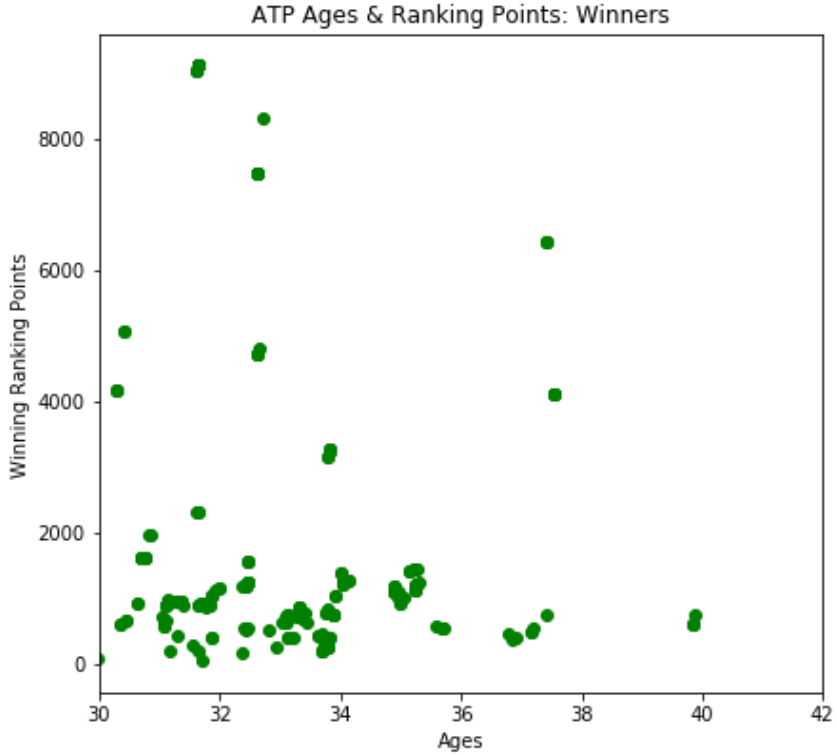
In [7]:
```
plt.figure(figsize=(15, 14))


plt.subplot(2,2,1)          #(row, col, postition)
plt.scatter(x=df['loser_age'], y=df['loser_rank_points'], color= 'orange')
plt.xlim(15,30)
plt.ylabel('Losing Ranking Points')
plt.xlabel('Ages')
plt.title('ATP Ages & Ranking Points: Losers')

plt.subplot(2,2,3)
plt.scatter(x=df['winner_age'], y=df['winner_rank_points'], color= 'green')
plt.xlim(15,30)
plt.ylabel('Winning Ranking Points')
plt.xlabel('Ages')
plt.title('ATP Ages & Ranking Points: Winners')

plt.subplot(2,2,2)
plt.scatter(x=df['loser_age'], y=df['loser_rank_points'], color= 'orange')
plt.xlim(30,42)
plt.ylabel('Losing Ranking Points')
plt.xlabel('Ages')
plt.title('ATP Ages & Ranking Points: Losers')

plt.subplot(2,2,4)
plt.scatter(x=df['winner_age'], y=df['winner_rank_points'], color= 'green')
plt.xlim(30,42)
plt.ylabel('Winning Ranking Points')
plt.xlabel('Ages')
plt.title('ATP Ages & Ranking Points: Winners')

plt.show
print("Winner's Ranking Points Standard Deviation:",(df['winner_rank_points'].std(ddof=1)))
print("Loser's Ranking Points Standard Deviation:",(df['loser_rank_points'].std(ddof=1)))
```

```
Winner's Ranking Points Standard Deviation: 1632.3572258382071
Loser's Ranking Points Standard Deviation: 942.7209691384879
```









The more points accumulated by each player the better their ranking status will be in the tournament. Here we can see what ages had a

high points in winning ranking points, and what ages had a high point in the losing ranking points.

In the winning ranking points, ages between 30-34 have the highest ranking points. But there are a couple around age 22 that do have a good ranking points.

2.2 Gather the Mean, Median, and Mode of the winners and losers age

In [8]:
```
#gather mean and median of the winning ages and losing ages
w_mean= np.mean(df['winner_age'])
l_mean= np.mean(df['loser_age'])
w_med= np.median(df['winner_age'])
l_med= np.median(df['loser_age'])
```

In [9]:
```
#Gather mode for both winners and losers
import statistics
w_mode= statistics.mode(df['winner_age'])
#Loser's age have more than one mode
freqs = groupby(Counter(df['loser_age']).most_common(), lambda x:x[1])
l_mode= [val for val,count in next(freqs)[1]]
```

In [10]:
```
#Chart to show Mean, Median, and Mode of the ages of the tennis players
chart = {'Age_Mean':[w_mean,l_mean], 'Age_Median':[w_med,l_med], 'Age_Mode':[w_mode,l_mode]}
df_chart = pd.DataFrame(chart, index =['Winner','Loser'])
df_chart
```

Out[10]:

	Age_Mean	Age_Median	Age_Mode
Winner	27.822884	28.13	32.62
Loser	27.842060	28.10	[28.68, 28.74]

Here we can conclude that mean and median doesn't really tell us much about if there is a meaning between age and winning or losing, but we do see a difference in mode between winner and loser. The mode, tells us that a player that is of 32.62 of age is represented the most in the winning ranking/points, and a player with 28.68 and 28.74 of age is represented the most in losing ranking/points.

3.0 Correlation between age and ranking

Further more, we can conclude if age has an effect on the possibility of having a good ranking by calculating the correlation

Correlation between Winner age and Winner's ranking points

In [11]:
```
from scipy.stats import pearsonr
corr, _ = pearsonr(df['winner_age'],df['winner_rank_points'] )
print('Pearsons correlation: %.3f' % corr)
```

```
Pearsons correlation: 0.112
```

Correlation between Loser age and Loser's ranking points

In [12]:
```
corr, _ = pearsonr(df['loser_age'],df['loser_rank_points'] )
print('Pearsons correlation: %.3f' % corr)
```

```
Pearsons correlation: 0.040
```

There is a very low (weak) correlation between the two, so age doesn't affect points. As the age increase the points can either decrease or increase.

4.0 Compare each tourney's highest and lowest ranking points: For Winners

In [13]:
```
df_hl= df[['tourney_name','winner_rank_points']]
```

Tourney's maximum winner ranking points: Top 10

In [14]:
```
w_max= df_hl.groupby(['tourney_name']).max().sort_values(by= 'winner_rank_points',ascending=False)
w_max.head(10)
```

Out[14]:

tourney_name	winner_rank_points
Australian Open	9135.0
Doha	9045.0
Acapulco	8320.0
Davis Cup QLS R1: GER vs HUN	6475.0
Delray Beach	5060.0
Pune	4710.0
Dubai	4190.0
Rotterdam	4100.0
Buenos Aires	3960.0
Brisbane	3590.0

In [15]:
```
w_max.mean()
```

Out[15]:
```
winner_rank_points    2982.733333
dtype: float64
```

In [16]:
```
w_max.std(ddof=1)
```

Out[16]:
```
winner_rank_points    2504.755008
dtype: float64
```

Tourney's minimum ranking points: Top 10

In [17]:
```
w_min= df_hl.groupby(['tourney_name']).min().sort_values(by= 'winner_rank_points')
w_min.head(10)
```

Out[17]:

tourney_name	winner_rank_points
Sao Paulo	19.0
Auckland	56.0
Davis Cup QLS R1: SWE vs COL	72.0
Cordoba	152.0
Davis Cup QLS R1: JPN vs CHN	177.0
Brisbane	200.0
Rotterdam	244.0
Montpellier	245.0
Australian Open	252.0
Davis Cup QLS R1: BEL vs BRA	265.0

In [18]:
```
w_min.mean()
```

Out[18]:
```
winner_rank_points    456.966667
dtype: float64
```

In [19]:
```
w_min.std(ddof=1)
```

Out[19]:
```
winner_rank_points    402.501273
dtype: float64
```

Here we can see which tournament will have a more competitive match. The higher the point the more competitive and high of importance is the tournament.

Both in the min points and max points, there is high standard deviation. Having a high standard deviation, data points are going to be scatter away from the mean. There will be a wide variety of scores across the data from different tournaments.

Conclusion

Remember that a player ranking points, really doesn't represent the 'true' players' ranking. Let's say the player didn't play one year or was absent for a couple of months, they lose the opportunity of earning points, and therefor fall behind in the rankings.

This data just shows us a year of ATP matches for 2019, and the player's ranking points. Is a small sample and not very accurate when it comes to ranking players, but we can see the difference between ages and the player's ranking points. The data set is very scatter around, with a high standard deviation.

There is a possibility that a young player could hit a high ranking point, but it seems like the best ranking were hit with players between age 30 and 34. Which 32.62, is what our mode came out to be for the most age represented in the winner's age.

There is also a low correlation between age, and ranking points. Which shows that age does not affect the possibility of winning or losing.

In the Australian Open, the highest ranking point was 9135.0. And the top three ranking players are from Serbia, Spain, Germany.

Why can this data be important? Tennis is a very high paying sports. Some fanatics of the game will always have their favorite players for personal reasons, some like to watch only the good players play, some would like to attend the most competitive tournaments, and some even bet on matches, and/or players.

Propose Further Research

Lets say we need to estimate a ranking point for the winner with a certain age. We would like to get a regression analytics that can predict winning rank / losing given an age. As seen above our data is scatter, and so a non-linear line should be formed. Python has a libaray, scikit-learn, that can help compute a prediction for an outcome of age to rank creating a line through the scatter graph.

References

1-https://ftw.usatoday.com/2018/08/atp-wta-tennis-rankings-how-do-they-work-faq-federer-serena-nadal (https://ftw.usatoday.com/2018/08/atp-wta-tennis-rankings-how-do-they-work-faq-federer-serena-nadal)

2-https://www.atptour.com/ (https://www.atptour.com/)

3-https://en.wikipedia.org/wiki/Association_of_Tennis_Professionals (https://en.wikipedia.org/wiki/Association_of_Tennis_Professionals)