Project 3

SDS348 Spring 2021

In [ ]:
```
### Yanyu Yang yy8439
```

Exploratory Data Analysis

The dataset I will be using is the combined dataset that I used in both projects 1 and 2. The observations were my top 50 Osu! plays as of March 17, 2021.

In [12]:
```
# Import package
import pandas as pd
import seaborn as sn
import matplotlib.pyplot as plt
```

In [4]:
```
# Import dataset
osu = pd.read_csv("C:\\Users\\ta1je\\OneDrive\\Documents\\sds 348\\project\\osu.csv")
```

In [6]:
```
# head
osu.head()
```

Out[6]:

	song	difficulty_name	accuracy	pp	top_pp	length	bpm	difficulty	overweightness	dif
0	Koto no Ha (TV Size)	Gu's Insane	96.90	54	52	88	145	3.91	0	
1	&Z (TV size)	Insane	97.01	47	92	88	158	4.20	169	
2	unravel	Hard	91.70	45	44	203	135	3.52	0	
3	BRAVE JEWEL (TV Size)	Hyper	95.05	44	72	89	192	3.98	3	
4	IGNITE (TV size ver.)	Hard	99.06	44	52	87	171	3.43	12	

In [8]:
```
# info
osu.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 10 columns):
song                    50 non-null object
difficulty_name         50 non-null object
accuracy                50 non-null float64
pp                      50 non-null int64
top_pp                  50 non-null int64
length                  50 non-null int64
bpm                     50 non-null int64
difficulty              50 non-null float64
overweightness          50 non-null int64
difficulty_category     50 non-null object
dtypes: float64(2), int64(5), object(3)
memory usage: 4.0+ KB
```

The osu data set has 10 variables with a total of 50 observations.

The numeric variables are accuracy, pp, top_pp, length, bpm, difficulty, and overweightness. The categorical variables are song, difficulty_name, and difficulty_category.

In [ ]:
```
### Summary Statistics
```

In [9]:
```
osu.describe()
```

Out[9]:

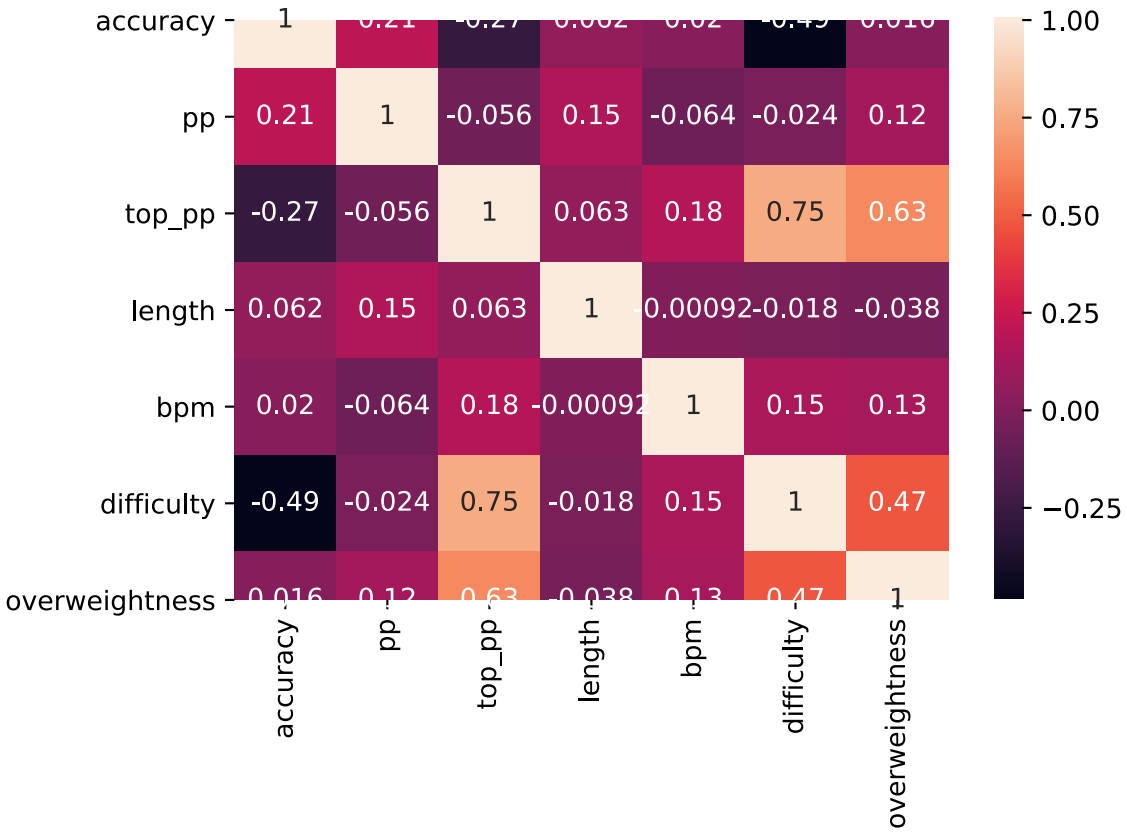
	accuracy	pp	top_pp	length	bpm	difficulty	overweightness
count	50.000000	50.000000	50.000000	50.000000	50.0000	50.000000	50.000000
mean	94.671200	34.840000	59.060000	111.180000	163.8400	3.716800	9.580000
std	3.204321	5.775953	17.944825	44.729888	28.1674	0.374468	31.021447
min	86.470000	28.000000	33.000000	56.000000	79.0000	3.090000	0.000000
25%	92.722500	30.000000	48.000000	87.000000	145.0000	3.450000	0.000000
50%	95.145000	33.000000	53.500000	90.000000	170.0000	3.615000	0.000000
75%	97.122500	38.000000	64.500000	116.000000	184.0000	3.955000	2.000000
max	100.000000	54.000000	121.000000	242.000000	222.0000	4.740000	169.000000

The table above shows the count, mean, standard deviation, percentiles, and quartiles for numeric variables in the dataset.

Correlation Matrix

In [15]:
```
# correlation matrix
osu_corr = osu.corr()
```

In [30]:
```
osu_corr_map = sn.heatmap(osu_corr, annot = True)
plt.show()
```

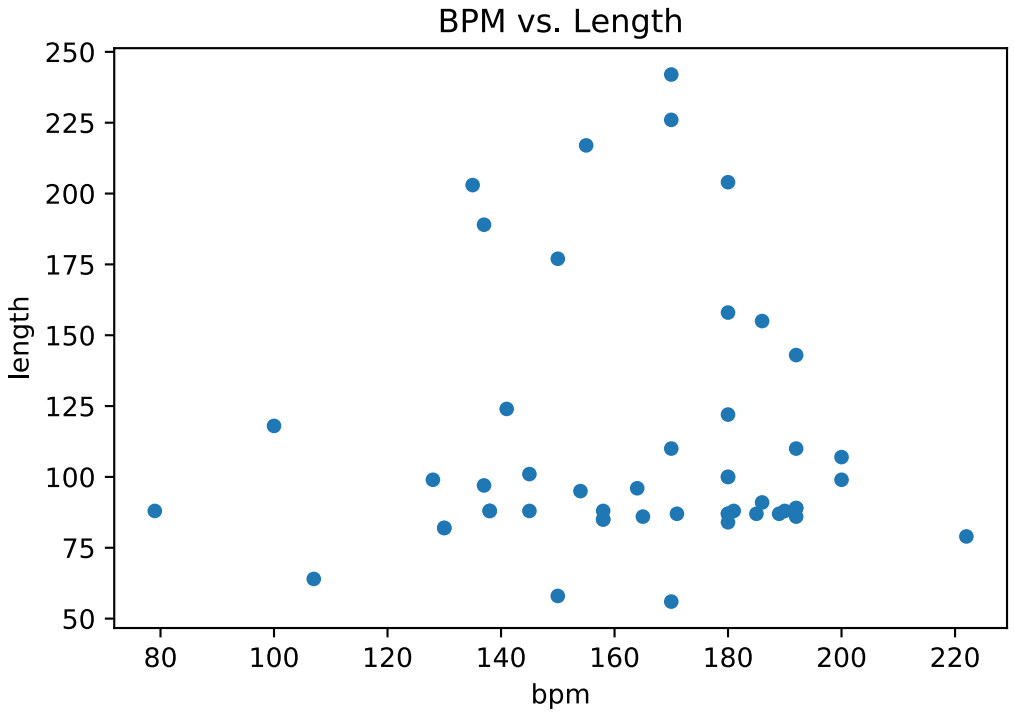


The correlation matrix of variables shows that accuracy and overweightness have the strongest negative correlation. On the other hand, top_pp and difficulty have the strongest positive correlation.

Scatterplot

In [36]:
```
# scatterplot of bpm vs length
osu.plot.scatter(x = 'bpm', y = 'length')
plt.title('BPM vs. Length')
```

Out[36]:
```
Text(0.5, 1.0, 'BPM vs. Length')
```

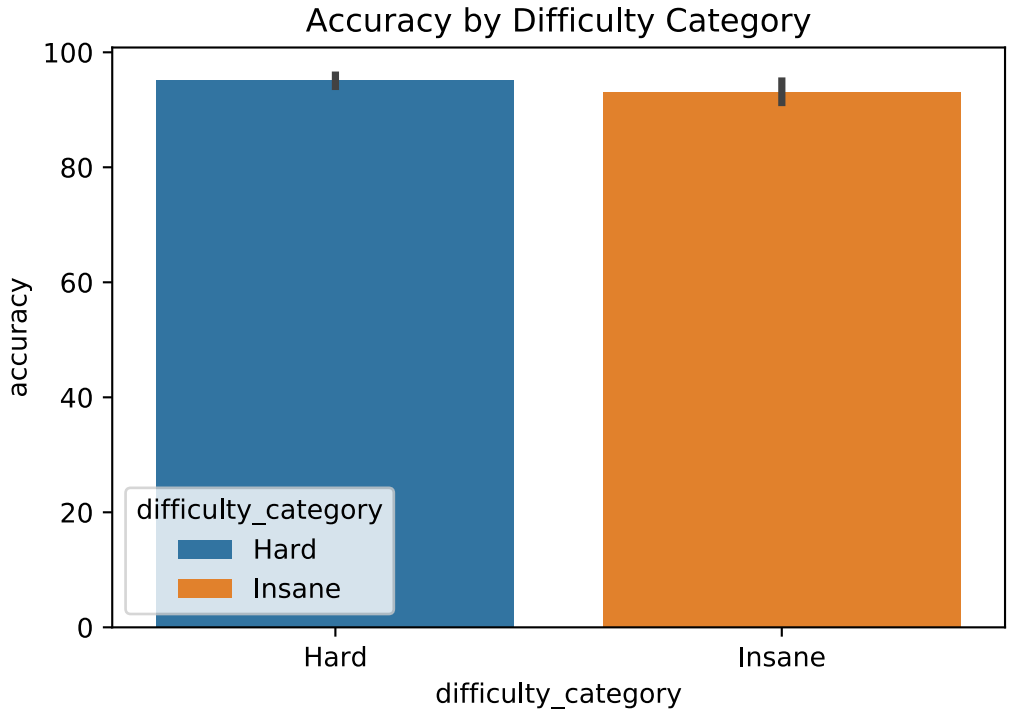


The scatter plot above shows the relationship between beats per minute and length of each map in the dataset.

Bar Plot

In [49]:
```
# bar plot of accuracy by difficulty category
sn.barplot(y = 'accuracy', x = 'difficulty_category', hue = 'difficulty
_category', data = osu, dodge = False)
plt.title('Accuracy by Difficulty Category')
```

Out[49]:
```
Text(0.5, 1.0, 'Accuracy by Difficulty Category')
```



According to the bar plot of accuracy by difficulty category, there does not appear to be much of a difference between Hard and Insane. However, Insane maps have a greater standard deviation for accuracy compared to Hard maps.