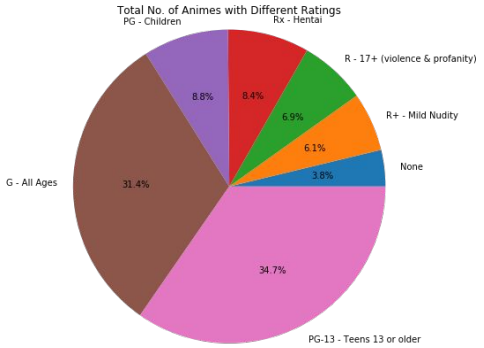
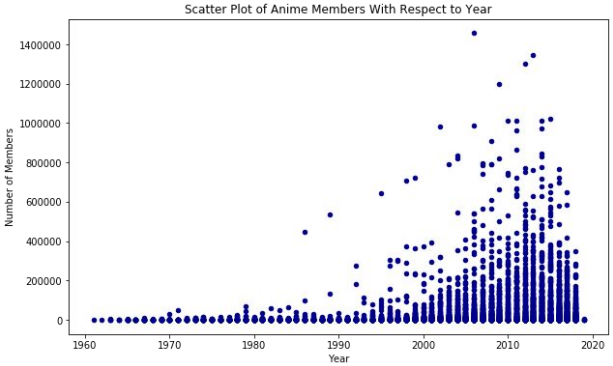
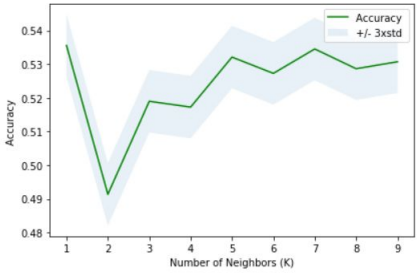
Project Poster



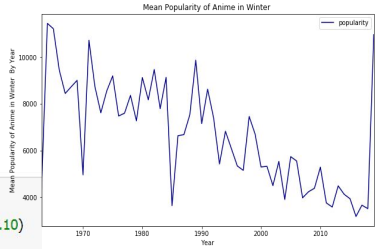


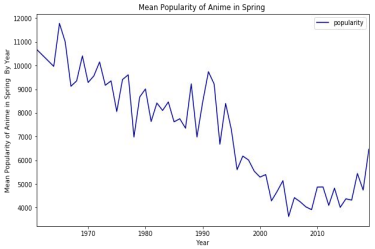

```
#Plot model accuracy for Different number of Neighbors
plt.plot(range(1,Ks),mean_acc,'g')
plt.fill_between(range(1,Ks),mean_acc - 1 * std_acc,mean_acc + 1 * std_acc, alpha=0.10)
plt.legend(('Accuracy ', '+/- 3xstd'))
plt.ylabel('Accuracy ')
plt.xlabel('Number of Neighbors (K)')
plt.tight_layout()
plt.show()
```

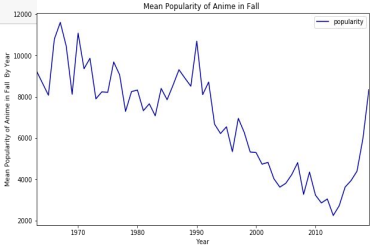


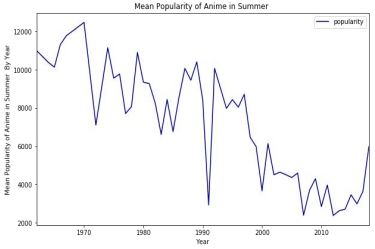

```
print( "The best accuracy was with", mean_acc.max(), "with k=", mean_acc.argmax()+1)
```

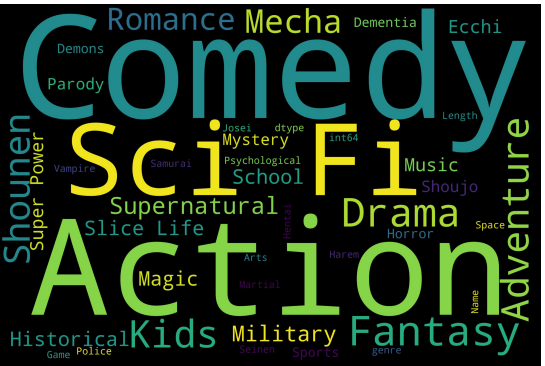
The best accuracy was with 0.5355662983425414 with k= 1












```
similar_animes(query="Naruto")
```

No Game No Life
Angel Beats!
Tokyo Ghoul
Code Geass: Hangyaku no Lelouch
Mirai Nikki (TV)

```
# It's well known that Naruto is a very popular anime, therefore, the 5 recommended options are very popular as well and
# some of them has many episodes and series such as Tokyo Ghoul and Code Geass with lots of fighting scenes.
# Personally, I would really recommend Angle Beats - a very touching anime
```

```
# When I put in a strange anime name it will cause error
similar_animes(query="You")
```

This is not a valid full name

```
# Therefore I check the available names for "slime"
part_id("slime")
```

Slime Boukenki: Umi da, Yeah! 1090
Tensei shitara Slime Datta Ken 3072

```
# After copy paste the name, it works!!
similar_animes(query="Slime Boukenki: Umi da, Yeah!")
```

Mugen no Ryvius: Illusion
Morinaga Nyuugyou x Mary to Majo no Hana
Kingdom of Chaos: Born to Kill
Fushigi na Somera-chan Special
Bamboo Blade: CM Fanfu-Fufe-Fo

User input is:

{'type': 'TV', 'source': 'Picture book', 'episodes': 12.003246304738223, 'status': 'Finished Airing', 'airing': True, 'duration': 26.237322491727266, 'rating': 'PG - Children', 'scored_by': 11460.02527973477, 'rank': 5739.031634203619, 'popularity': 7220.259566238431, 'members': 22966.402679928167, 'favorites': 311.6496062992126, 'licensor': 0.6075424782428512, 'genre': ', Yaoi, Historical, Space, Parody', 'year': 2017}

```
# Return the anime score from the input
df2 = pd.DataFrame([response_default], columns=use_cols)
df_test = df_model1.append(df2, sort = True)
X_test = df_test[use_cols].values
x_test = preprocessing.StandardScaler().fit(X_test).transform(X_test.astype(float))
score = neigh.predict([x_test[len(x_test) - 1]])
print("The predicted anime score is: " + str(score))
```

The predicted anime score is: [7]

```
# Here, an anime about Historical and Space for children which originate from Picture book has a decent score (7)
# This makes sense because this anime seems to be very interesting for children
# and would attract lots of viewers for education material
```

By Andrew, Nafi, and Eric