In [ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
```

In [ ]:
```
genomes_df = pd.read_csv("genome-scores.csv")
genomes_df
```

In [ ]:
```
relevance_df = genomes_df[["relevance"]]
unique_tags = genomes_df["tagId"].unique()
```

In [ ]:
```
relevance_df.shape
```

In [ ]:
```
relevance_scores = relevance_df.to_numpy().reshape(13176, 1128)
#relevance_scores.shape #num_movies, num_tags
relevance_df = pd.DataFrame(relevance_scores, columns=unique_tags)
```

In [ ]:
```
variance_captured = []
for i in np.arange(0, 1000, 10):
  pca = PCA(n_components=i)
  pca.fit(relevance_df)
  variance_captured.append(sum(pca.explained_variance_ratio_))
```

In [ ]:
```
plt.plot(np.arange(0, 1000, 10), variance_captured)
plt.title("Explained Variance as a Function of the Number of Principle Compone
nts Kept")
plt.xlabel("Number of Principle Components")
plt.ylabel("Explained Variance")
```

In [ ]:
```
relevance_df
```

In [ ]:
```
pca = PCA(n_components=100)
pca.fit(relevance_df)
relevances = pca.transform(relevance_df)
pca_relevances_df = pd.DataFrame(relevances)
pca_relevances_df
```

In [ ]:
```
pca_components_df
```

In [ ]:
```
movie_ids = genomes_df["movieId"].unique()
```

In [ ]:
```
final_df = pd.DataFrame(movie_ids, columns=["movieId"])
```

In [ ]:
```
final_df
```

In [ ]:
```
final_df = pd.concat([final_df, pca_relevances_df], axis=1)
```

In [ ]:
```
final_df.to_csv("genome-scores-pca'd.csv")
```

In [ ]:
```

```