

```
In [1]: import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn import metrics
from sklearn.metrics import accuracy_score
```

```
In [2]: # Import data set csv
data_set = pd.read_csv('filtered_data_set.csv')
```

```
In [3]: # Set X and y columns
X = data_set[['valence', 'acousticness', 'danceability', 'duration_ms', 'energy', 'exp
            'instrumentalness', 'key', 'liveness', 'loudness', 'mode', 'speechiness',
y = data_set['popularity']
```

```
In [4]: # Create the X training and testing set, and Y training and testing set where 70%
# are for the training set and the rest to the testing set.
x_testing_set, x_training_set, y_testing_set, y_training_set = train_test_split()
```

```
In [5]: # Create linear regression object
lm = LinearRegression()
```

```
In [6]: # Fit the linear model to the training data
lm.fit(x_training_set, y_training_set)
```

```
Out[6]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

```
In [7]: # Find the predicted y values
predictions = lm.predict(x_testing_set)
```

```
In [8]: # Compare the predicted outputs to a threshold value
scaled_predictions = []
for i in range(0, len(predictions)):
    if(predictions[i] >= 0.5):
        scaled_predictions.append(1)
    else:
        scaled_predictions.append(0)
```

```
In [9]: # Calculate Accuracy
accuracy_score(y_testing_set, scaled_predictions)
```

```
Out[9]: 0.779128672745694
```

```
In [ ]:
```

