In [2]:

```
'''
Patrick Ballou
ID: 801130521
ECGR 4105
Homework 4
Problem 1
'''
```

Out[2]:

```
'\nPatrick Ballou\nID: 801130521\nECGR 4105\nHomework 4\nProblem 1\n'
```

In [3]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split
from sklearn.decomposition import PCA
from sklearn import metrics
from sklearn.datasets import load_breast_cancer
from sklearn.preprocessing import MinMaxScaler, StandardScaler
from sklearn.metrics import PrecisionRecallDisplay
```

In [4]:

```
breast = load_breast_cancer()
breast_data = breast.data
breast_data.shape
breast_input = pd.DataFrame(breast_data)
```

In [5]:

```
breast_labels = breast.target
labels = np.reshape(breast_labels,(569,1))
final_breast_data = np.concatenate([breast_data, labels],axis=1)
```

In [6]:

```
breast_dataset = pd.DataFrame(final_breast_data)
features = breast.feature_names
features_labels = np.append(features, 'label')
breast_dataset.columns = features_labels
breast_dataset.head()
```

Out[6]:

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	dim
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	(
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	(
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	(
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	(
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	(

5 rows × 31 columns

In [7]:

```
x = breast_dataset[features]
Y = breast_dataset['label']
```

In [8]:

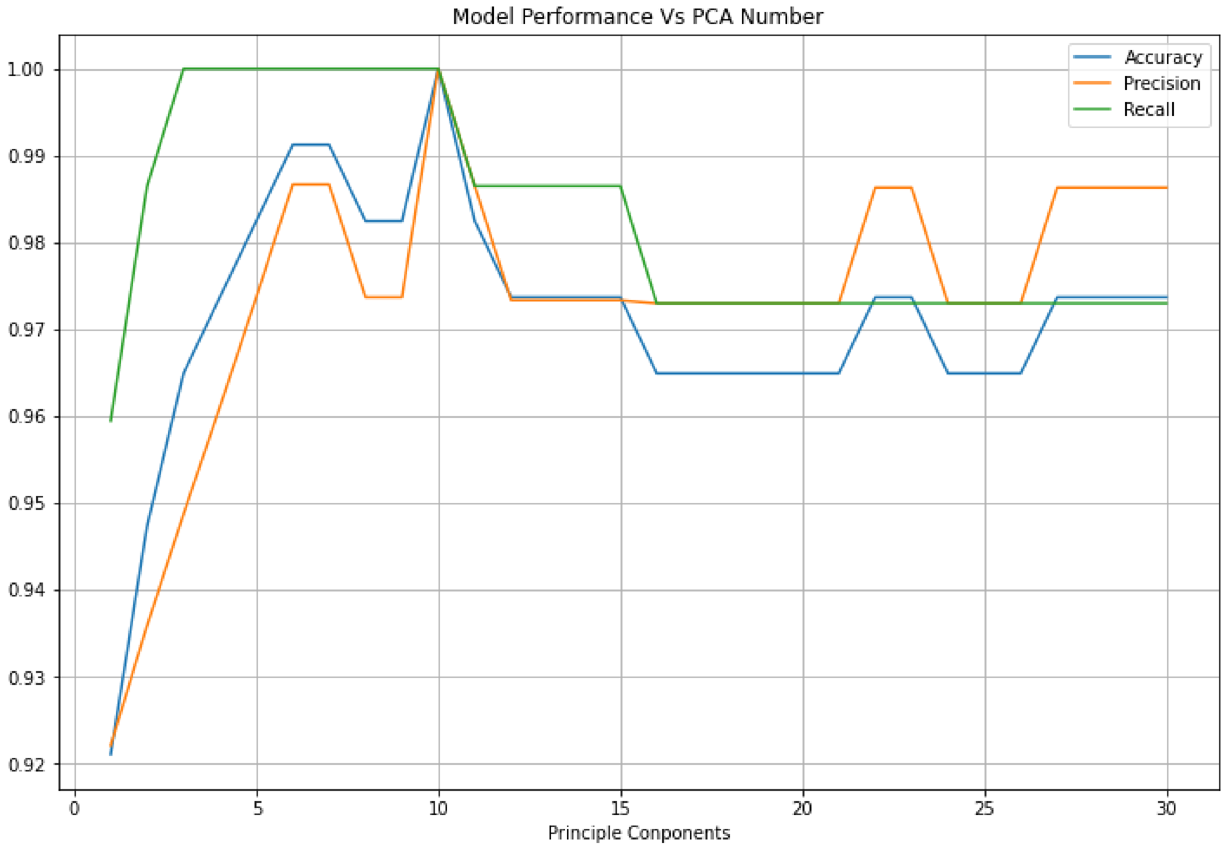
```
#min max scaler is better
#scaler = StandardScaler()
scaler = MinMaxScaler()
X = scaler.fit_transform(x)
```

In [35]:

```
accuracy_history = list()
precision_history = list()
recall_history = list()
for pca_num in range(1, 31):
    pca = PCA(n_components=pca_num)
    principalComponents = pca.fit_transform(X)
    principalDf = pd.DataFrame(data = principalComponents)

    X_train, X_test, Y_train, Y_test = train_test_split(principalDf, Y, train_size=.8,

    classifier = SVC(random_state=7, C=10, kernel="rbf")
    classifier.fit(X_train, Y_train)
    Y_pred = classifier.predict(X_test)
    accuracy_history.append(metrics.accuracy_score(Y_test, Y_pred))
    precision_history.append(metrics.precision_score(Y_test, Y_pred))
    recall_history.append(metrics.recall_score(Y_test, Y_pred))
```

In [36]:

```
plt.rcParams["figure.figsize"] = (12,8)
plt.plot(range(1, 31), accuracy_history, label="Accuracy")
plt.plot(range(1, 31), precision_history, label="Precision")
plt.plot(range(1, 31), recall_history, label="Recall")
plt.xlabel("Principle Conponents")
plt.title("Model Performance Vs PCA Number")
plt.legend()
plt.grid()
plt.show()
```



In [26]:

```
#pca_num=10 is the optimal number of components, so now we can evaluate the model with
pca = PCA(n_components=10)
principalComponents = pca.fit_transform(X)
principalDf = pd.DataFrame(data = principalComponents)

X_train, X_test, Y_train, Y_test = train_test_split(principalDf, Y, train_size=.8, ran

classifier = SVC(random_state=7, C=10, kernel="rbf")
classifier.fit(X_train, Y_train)
Y_pred = classifier.predict(X_test)
```

In [27]:

```
print(metrics.classification_report(Y_test, Y_pred))
print(metrics.confusion_matrix(Y_test, Y_pred))
plt.rcParams["figure.figsize"] = (12,8)
```

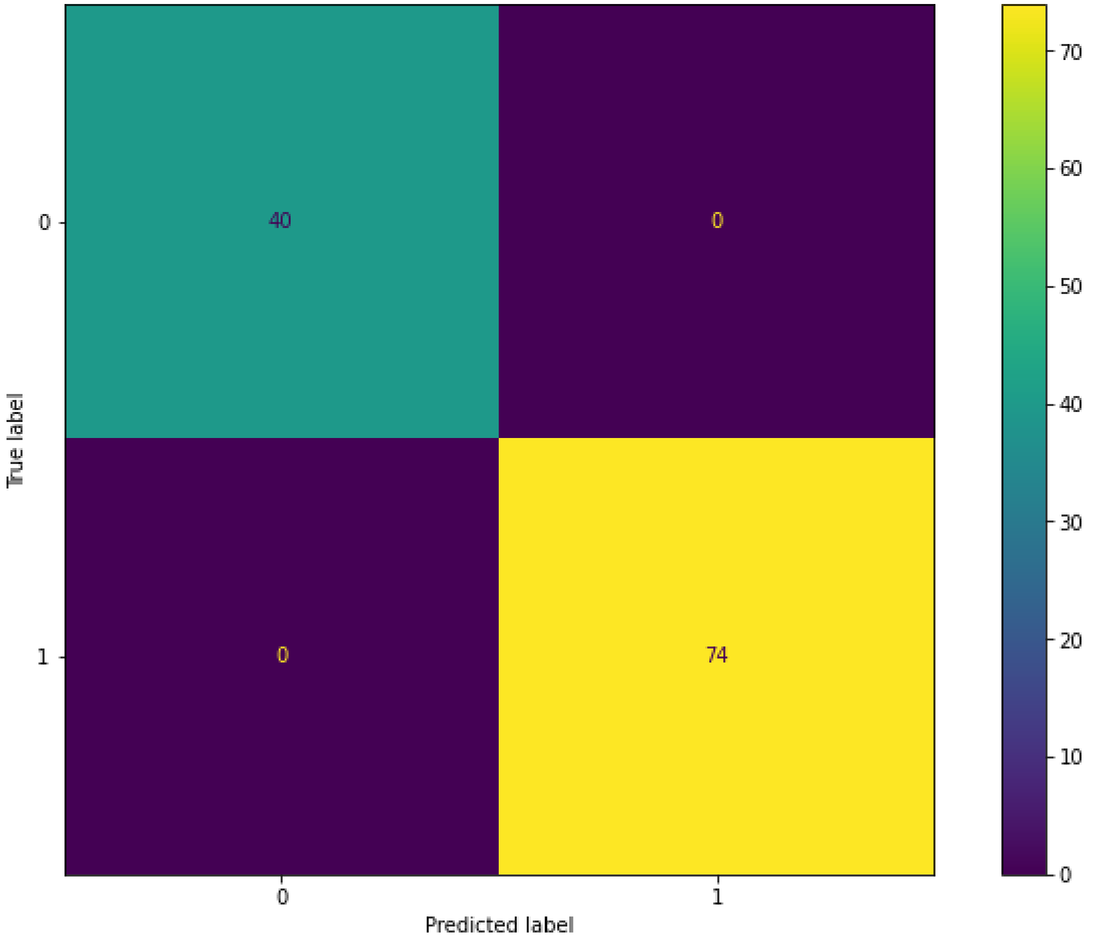
```
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00        40
         1.0       1.00      1.00      1.00        74

    accuracy                           1.00       114
   macro avg       1.00      1.00      1.00       114
weighted avg       1.00      1.00      1.00       114

[[40  0]
 [ 0 74]]
```

In [28]:

```
cm_display = metrics.ConfusionMatrixDisplay(confusion_matrix=metrics.confusion_matrix(
cm_display.plot()
```

Out[28]:

```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x257c654d4c0>
```


40
0
0
74
0
1
True label
0
1
Predicted label
70
60
50
40
30
20
10
0