## 主成分分析

In [1]:
```
import warnings
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegressionCV
from sklearn.metrics import confusion_matrix
from sklearn.decomposition import PCA
from sklearn.datasets import load_breast_cancer
import seaborn
warnings.simplefilter('ignore')
```

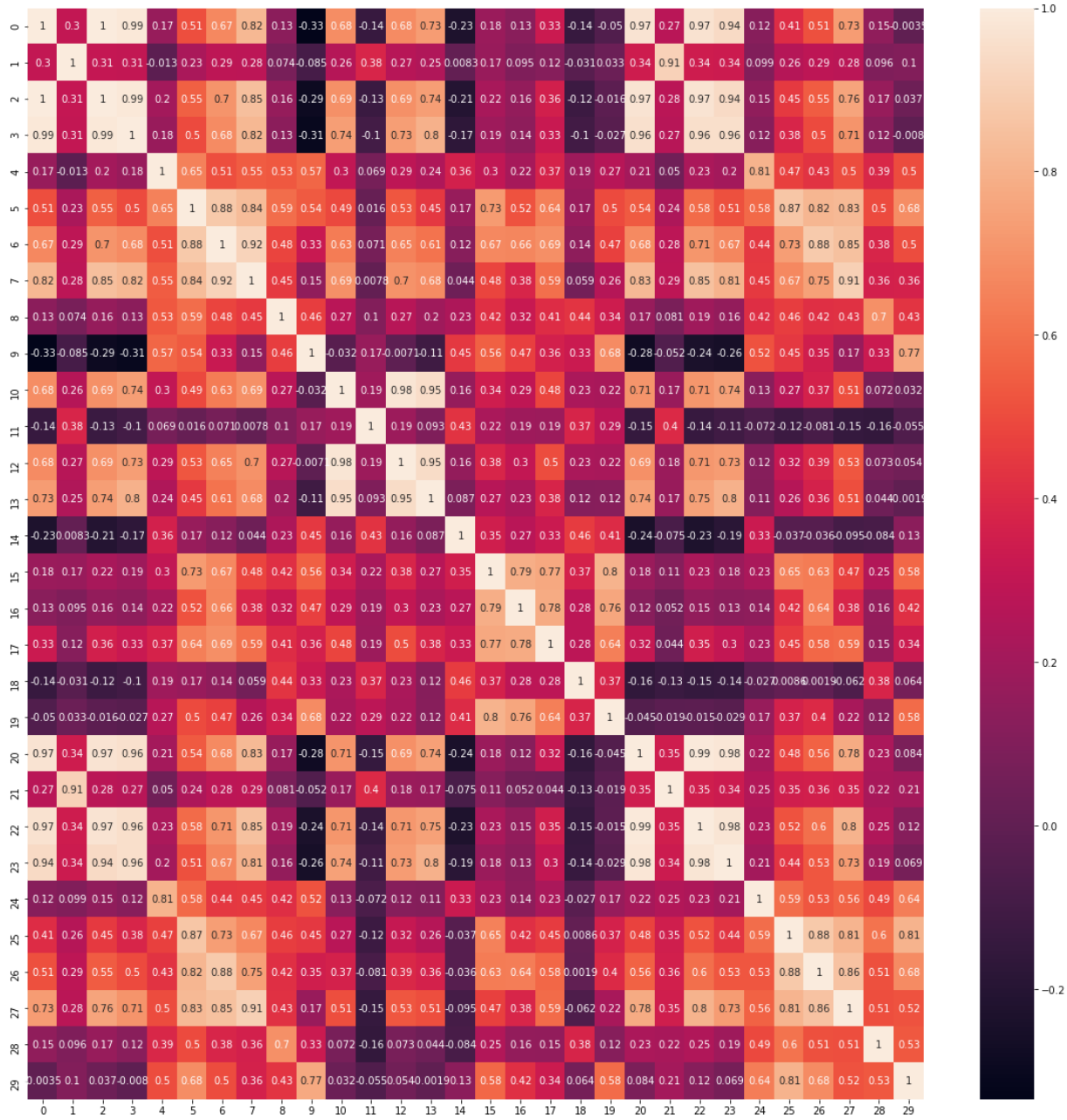
In [2]:
```
data_breast_cancer = load_breast_cancer()
X = pd.DataFrame(data_breast_cancer["data"],columns=data_breast_cancer["feature_names
y = pd.DataFrame(data_breast_cancer["target"],columns=["target"])

X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)

sc = StandardScaler()
X_train_std = sc.fit_transform(X_train)
X_test_std = sc.transform(X_test)
```

In [3]:
```
lr = LogisticRegressionCV(cv=10, random_state=0)
lr.fit(X_train_std,y_train)
print('train score:',lr.score(X_train_std,y_train))
print('test score:',lr.score(X_test_std,y_test))
```

```
train score: 0.9882629107981221
test score: 0.972027972027972
```

In [4]:
```
plt.figure(figsize=(20, 20))
seaborn.heatmap(pd.DataFrame(X_train_std).corr(), annot=True)
```

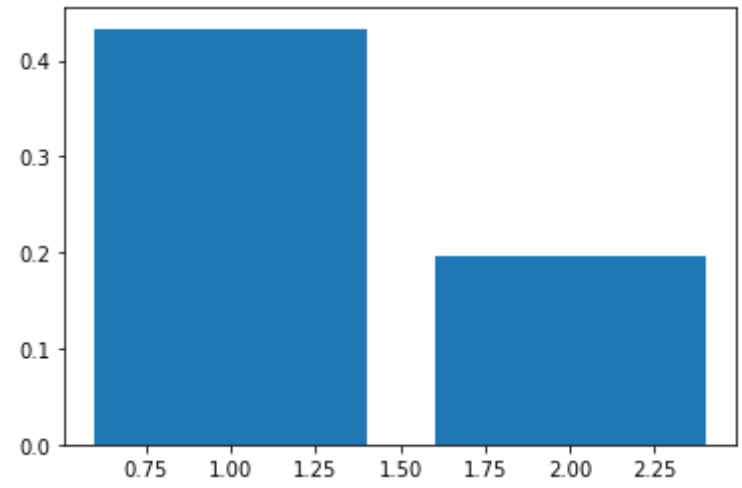
Out[4]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x26207f29908>
```



In [9]:

```
pca = PCA(n_components=2)
pca.fit(X_train_std)
plt.bar([n for n in range(1, len(pca.explained_variance_ratio_)+1)], pca.explained_va
```

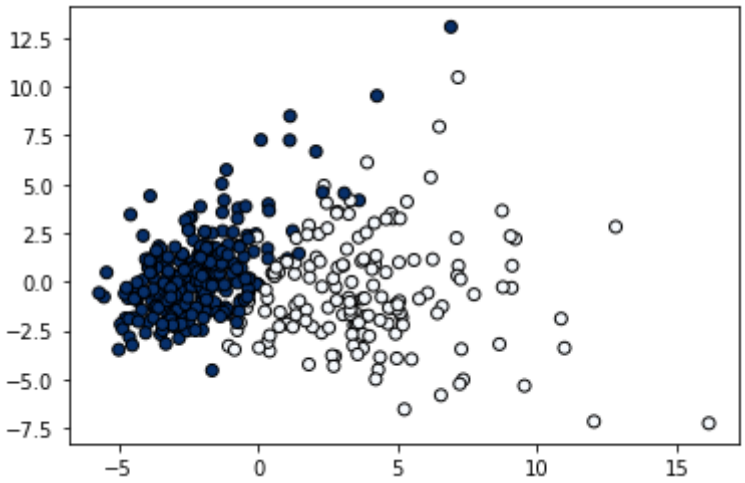
Out[9]: <BarContainer object of 2 artists>



In [10]:
```
X_train_pca = pca.fit_transform(X_train_std)
X_test_pca = pca.fit_transform(X_test_std)

lr = LogisticRegressionCV(cv=10, random_state=0)
lr.fit(X_train_pca,y_train)
print('train score:',lr.score(X_train_pca,y_train))
print('test score:',lr.score(X_test_pca,y_test))
plt.scatter(X_train_pca[:, 0], X_train_pca[:, 1],c = y_train.values.flatten(),edgecol
```

```
train score: 0.9647887323943662
test score: 0.916083916083916
```

Out[10]:
```
<matplotlib.collections.PathCollection at 0x26208d13408>
```




乳がんデータを主成分分析を行って次元削減したデータを、ロジスティック回帰で分類した。
2つの主成分だけで、データの分散の6割近くを担っていたため、すべてのデータを用いて行った場合とほぼ遜色なく学習ができた。
3つに増やした場合も行ったが、それほどスコアに変化はなかった。
欠点としては、より学習の中身がブラックボックス化してしまうため、説明性がひくくなる。