# ****数据降维****

在进行数据挖掘或者机器学习时，我们面临的数据往往是高维数据。相较于低维数据，高维数据为我们提供了更多的信息和细节，也更好的描述了样本；但同时，很多高效且准确的分析方法也将无法使用。处理高维数据和高维数据可视化是数据科学家们必不可少的技能。解决这个问题的方法便是降低数据的维度。在数据降维时，要使用尽量少的维度来表达较多原数据的特性和结构。

## ****PCA****

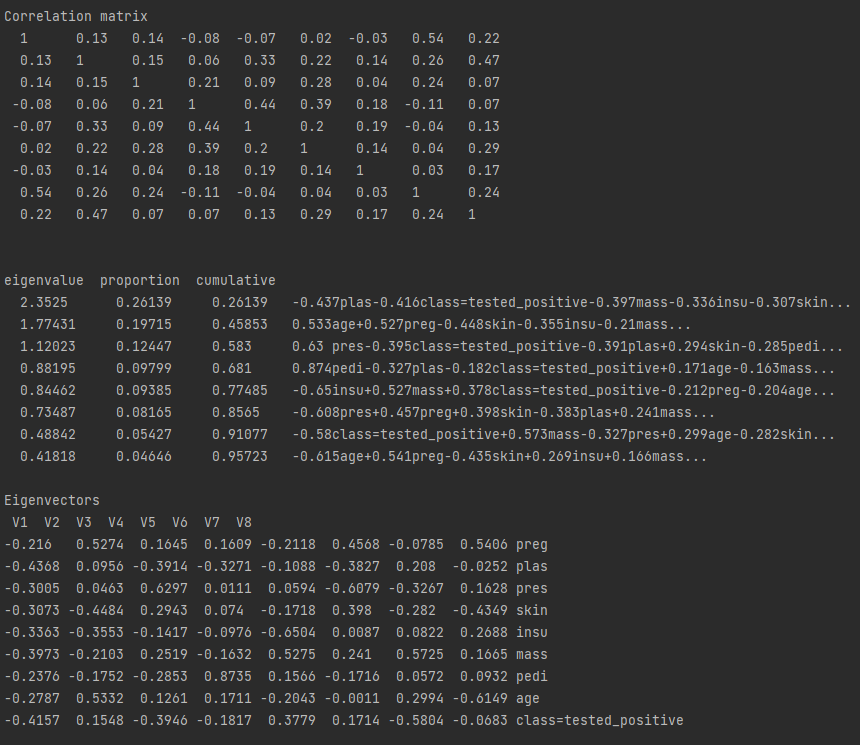
主成分分析（Principal Component Analysis, PCA）是一种使用最广泛的数据降维算法。PCA的主要思想是将n维特征映射到k维上，这k维是全新的正交特征也被称为主成分，是在原有n维特征的基础上重新构造出来的k维特征。PCA的工作就是从原始的空间中顺序地找一组相互正交的坐标轴，新的坐标轴的选择与数据本身是密切相关的。其中，第一个新坐标轴选择是原始数据中方差最大的方向，第二个新坐标轴选取是与第一个坐标轴正交的平面中使得方差最大的，第三个轴是与第1,2个轴正交的平面中方差最大的。依次类推，可以得到n个这样的坐标轴。通过这种方式获得的新的坐标轴，我们发现，大部分方差都包含在前面k个坐标轴中，后面的坐标轴所含的方差几乎为0。于是，我们可以忽略余下的坐标轴，只保留前面k个含有绝大部分方差的坐标轴。事实上，这相当于只保留包含绝大部分方差的维度特征，而忽略包含方差几乎为0的特征维度，实现对数据特征的降维处理。

### 对数据集的处理过程：

首先将数据集加载为Instances对象，再用PrincipalComponents类进行PCA降维处理。

package com.example.test;  
  
import weka.attributeSelection.PrincipalComponents;  
import weka.core.Instances;  
import weka.core.converters.ConverterUtils;  
  
***/\*\*  
 \* @title: PCA  
 \* @Author: Stanton JY  
 \* @Date: 2022/10/9 20:37  
 \*/***public class PCA {  
 public static void main(String[] args) throws Exception {  
 PrincipalComponents pca = new PrincipalComponents();  
 //pca.setVarianceCovered(0.6);  
 Instances data = ConverterUtils.DataSource.*read*("src/main/java/com/example/test/diabetes.arff");  
 pca.buildEvaluator(data);  
 System.*out*.println(pca);  
 //System.out.println(pca.transformedData(data));  
 }  
}

### 运行结果：



如图中结果所示，获得从大到小8个特征值，根据降维需要，可选择前k个特征值。

weka的pca默认是降维后的特征对原数据的可解释度达到95%，如果希望降维程度更大，可使用setVarianceCovered方法进行调整。如将上文代码对应行取消注释，运行结果为：

