数据挖掘大作业三：分类与聚类

姓名：孙澈
学号：2120171054

数据源

从以下 3 个数据集中任选一个

- [https://www.kaggle.com/c/predict-west-nile-virus/data]
- [https://www.kaggle.com/c/predict-wordpress-likes/data]
- [https://www.kaggle.com/c/titanic/data]

我在本次作业中选择了第三个数据库：泰坦尼克号上的生还率和各因素的关系数据库。包括性别、年龄、是否生还等信息。

要求：

1. 使用分类模型（朴素贝叶斯和支持向量机）对数据集进行挖掘；
2. 对挖掘结果进行可视化，并解释其意义（见下文）；
3. 使用聚类方法（k-means 和层次聚类）对数据集进行分析；
4. 对挖掘结果进行可视化，并解释其意义（见下文）。

提交的内容

- 对数据集进行挖掘的源程序

分类源程序：当前目录\classify\classify.m
聚类源程序：当前目录\cluster\ cluster1.m 和当前目录\cluster\ cluster2.m

- 挖掘过程和挖掘的结果

一、 分类

1 数据预处理

数据集提供了 PassengerID、Survived、Pclass 等多达 12 条属性，但是像 PassengerID 等属性显然对分类结果没有帮助，因此我选择了 Sex、Age、SibSP、Parch、Fare 和 Pclass 六条属性用于分类，而 Survived 作为分类标签，这样构成了“是否存活”的二分类任务。为了便于分类，需要将所有的属性变成数值，例如 Sex 变为{0,1}等，同时用每个属性的均值填充该属性的缺失值。最后要将每个属性归一化，我采用的归一化方式是

$$\mathrm{Nattribute} = \frac{\mathrm{attribute} - \min(\mathrm{attribute})}{\max(\mathrm{attribute}) - \min(\mathrm{attribute})},$$

这里，attribute表示原始属性，Nattribute示归一化后的结果。这种归一化的方式能够比较好地保留原始信息。

2 朴素贝叶斯分类器

2.1 实验方法

朴素贝叶斯分类器是基于贝叶斯公式的一种简单的分类器，它地一个假设是给定的属性之间相互条件独立。假定用 S={S_0,S_1}代表类别“存活”和“死亡”，剩下的六个属性用$a_1, a_2, \dots a_6$表示，那么朴素贝叶斯模型可以写成

$$S^* = \arg\max P(S_i|a_1, a_2, \dots a_6),$$

S^*表示在当前给定的属性值情况下最有可能所属的类别（概率值最大）。应用贝叶斯公式可得

$$S^* = \arg\max \frac{P(a_1, a_2, \dots a_6|S_i) * P(S_i)}{P(a_1, a_2, \dots a_6)}.$$

因为对于每个类别的概率值而言分母$P(a_1, a_2, \dots a_6)$没有用，因此可得

$$S^* = \arg\max \ \ P(a_1, a_2, \dots a_6|S_i) * P(S_i).$$

因为每个属性之间相互独立，那么$P(a_1, a_2, \dots a_6|S_i) = \prod_j P(a_j|S_i)$,每个属性的每个类条件概率都$P(a_j|S_i)$都可以通过大数定理统计出来，而每个类先验概率$P(S_i)$也可以统计出来，因此可以计算得到$S^*$，完成分类任务。

2.1 实验结果

我们的评价方法是正确率，即：正确分类的样本数/所有的样本数。下表是我们在训练集训练的模型在训练集和测试集分别实验得到的结果。

	Train_set	Test_set
accuracy	0.792368	0.928230

由表格可以看到，在测试集的正确率比训练集还高，这和我们一般的理解是不一样的，这是因为测试集的分布的类间差异明显比训练集大，易于区分，尤其是当模型十分简单的时候，在这种情况下的结果尤为明显，在支持向量机分类器的实验结果中将可视化这种情况。

3 支持向量机分类器

3.1 实验方法

给定线性可分训练数据集，通过间隔最大化或等价地求解相应的凸二次规划问题学习得到的分离超平面为

$$w * x + b = 0,$$

相应的分类决策函数为

$$f(x) = \text{sign}(w * x + b).$$

根据不同应用的需要，SVM 可以分为 linear SVM、Quadratic SVM、RBF SVM、Fine Gaussian SVM 等。我们通过 matlab 自带的 SVM，并分别选用了 linear SVM 和 RBF SVM 完成实验。

3.1 实验结果

Linear SVM 结果如下：

	Train_set	Test_set
accuracy	0.636364	0.616162

可以看出分类结果很差，这是因为有一些属性对于分类结果起到了负面作用或不起作用，对于 linear SVM 而言不能取得很好的分类超平面。

Matlab 默认 RBF SVM 采用高斯核函数，对于高斯核函数参数 sigma 的影响很大，具体数值如下表所示。

	Train_set	Test_set
Sigma=0.01	0.953984	0.593301
Sigma=1.01	0.829405	0.877990
Sigma=2.01	0.835017	0.968900
Sigma=3.01	0.828283	0.973684
Sigma=4.01	0.815937	0.980861
Sigma=5.01	0.802469	0.988038

可以看出 RBF SVM 的分类结果很好，并且参数 sigma 越大，在测试集上效果越好，反而在训练集上结果越差，这是因为 sigma 反映了 RBF 函数从最大值点向周围函数值下降的速度，sigma 越大，下降速度越慢，对应 RBF 函数越平缓；sigma 越小，下降速度越快，对应 RBF 函数越陡峭。换句话说，sigma 越小，分类曲线越复杂，事实也确实如此。因为 sigma 越小，RBF 函数越陡峭，下降速度越大，预测过程容易发生过拟合问题，使分类模型对训练数据过分拟合，而对测试数据预测效果不佳。所以根据奥卡姆剃刀原则，sigma 选择相对较大。

下面展示可视化结果，因为在高维空间展示比较困难，我选取了有代表性的两个属性 Sex（性别）和 Age（年龄），如图所示（选取 sigma 的值为 2.01），横坐标表示性别，纵坐标表示年龄（归一化后的）。

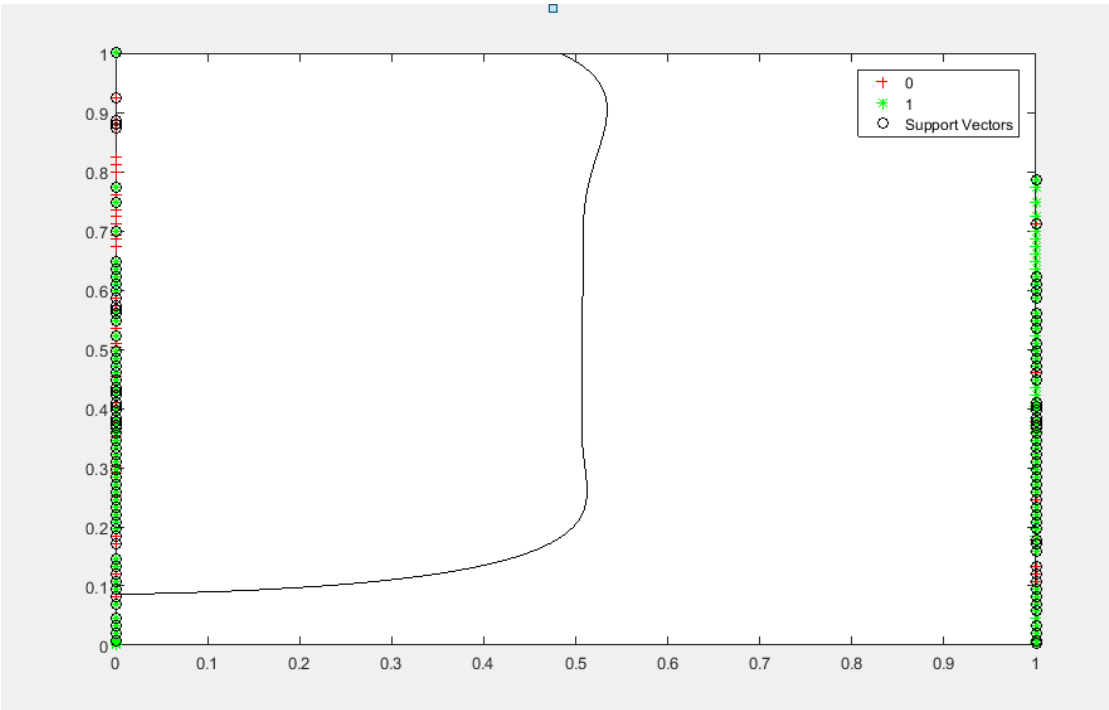

Figure 1 **训练集**

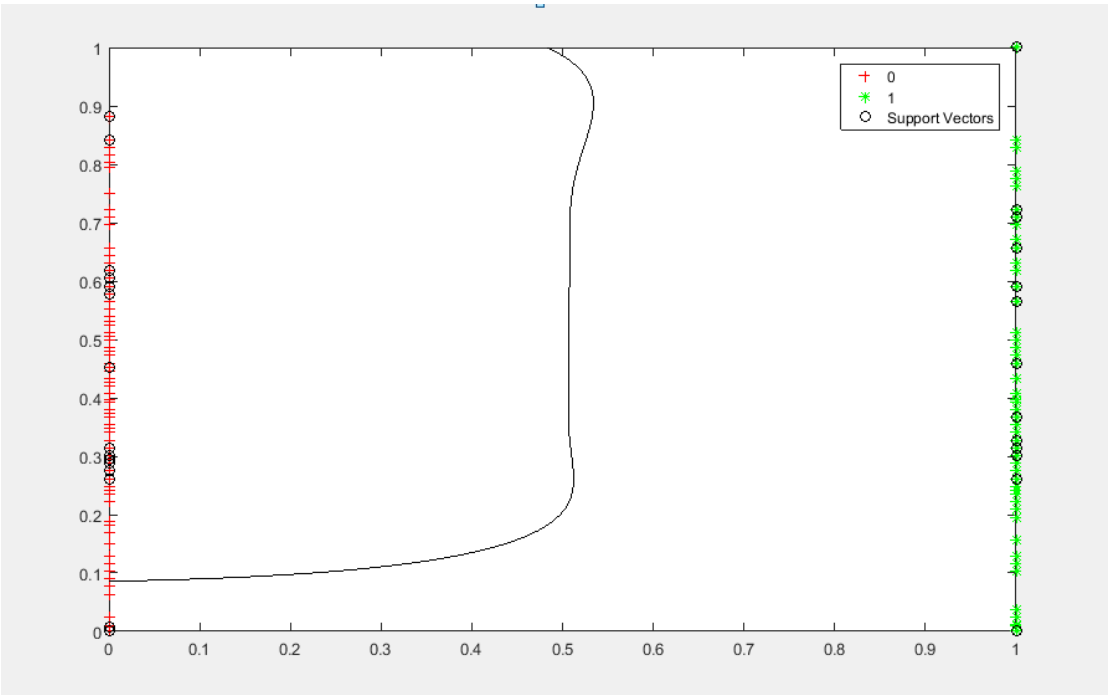

Figure 2 **测试集**

Figure1 是在训练集上的 svm 分类器，可以看出，在训练集上这两个属性不足以将所有的数据分开，但是在 Figure2 测试集上结果却可以看出，能够很好地分开。这是因为测试集属性的分布类间的差异很大，即在测试集上的性别属性就可以将所有的类别分开，这说明测试集分布存在不合理性，这也是为什么朴素贝叶斯的测试集结果要远好于训练集。

二、 聚类

1 数据预处理

聚类方法的数据预处理和分类方法相同，因为聚类方法不需要标签和测试集，因此只在训练集上聚类，将分类的六个属性和类别标签一起组成七个属性进行聚类。

2 k-means 聚类

2.1 实验方法

k-means 算法需要事先确定常数 k，常数 k 表示的聚类类别数。将事先输入的 n 个数据对象划分为 k 个类，得所获得的聚类满足：同一聚类中的对象相似度较高；而不同聚类中的对象相似度较小。聚类相似度是利用各聚类中对象的均值所获得一个“中心对象”来进行计算的。

我在实验中采用了 matlab 的内置函数 kmeans(sample,k)，其中 sample 是输入样本，k 是聚类中心数。

2.2 实验结果

在分类中我们可以得知，数据主要有两类构成，因此在实验中我设置k=2，同样的为了便于可视化，我挑选了两个属性进行聚类，分别是年龄Age和票价Fare，这是因为在我选取的归一化方式中，这两个属性比较有代表性，结果如图，横坐标是年龄，纵坐标是票价。

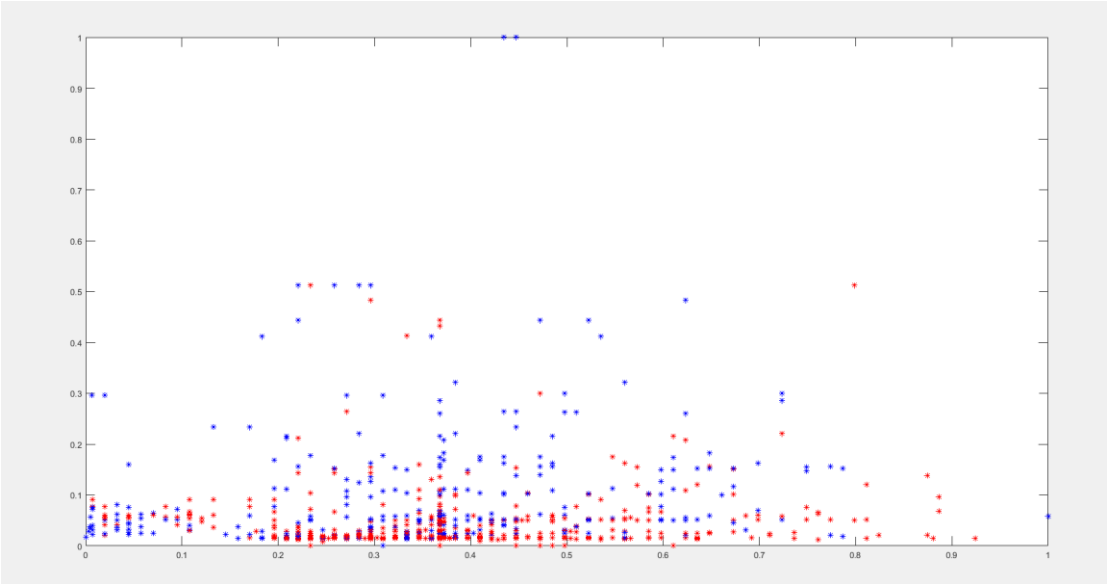

Figure 3

从图中可以看出，票价这一属性可以较好地将两类分开。

3 层次聚类

3.1 实验方法

层次聚类方法的基本思想是：通过某种相似性测度计算节点之间的相似性，并按相似度由高到低排序，逐步重新连接个节点。该方法的优点是可随时停止划分，主要步骤如下：

- 移除网络中的所有边，得到有n个孤立节点的初始状态；
- 计算网络中每对节点的相似度；
- 根据相似度从强到弱连接相应节点对，形成树状图；
- 根据实际需求横切树状图，获得社区结构。

同样的，需要利用matlab的一些内置函数协助完成聚类任务。

3.2 实验结果

这里主要展示层次聚类方法产生的树形结构，可以比较清晰地观察到数据地分布情况，明显地看出聚成了两类，如图所示。

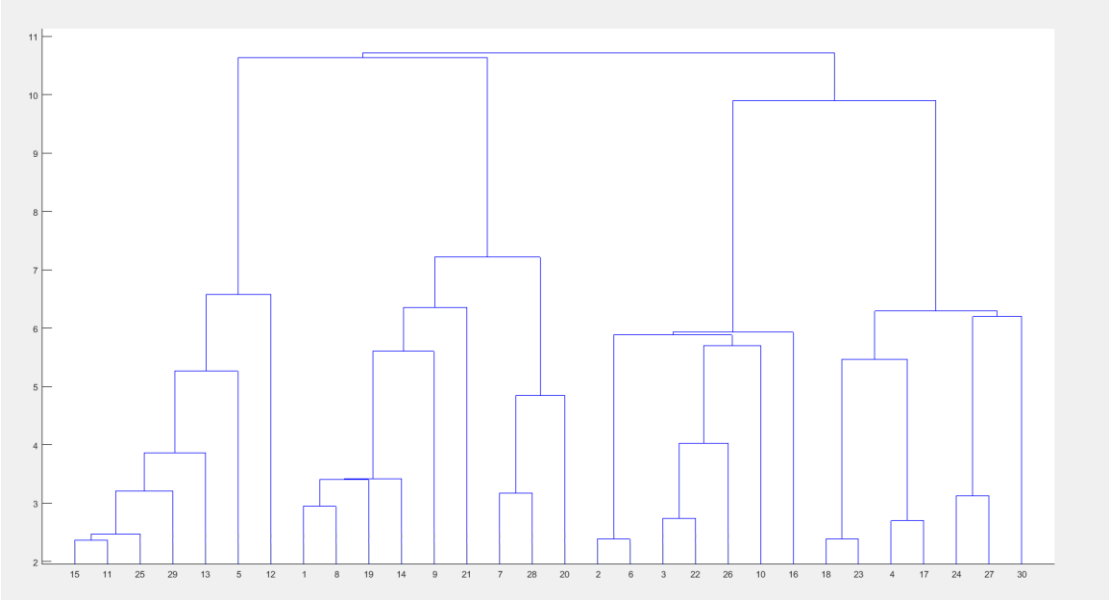

Figure 4