武汉大学

模式识别论文

基于 K-means 聚类算法和 KNN 决策判别器的国家经济实力评价

姓　　名：伍丹梅

学　　号：2016301500017

班　　级：计算机科学与技术 1 班

2018 年 4 月 21 日

摘　要

经济实力是衡量国家的综合实力的重要部分，本文通过 K-means 聚类算法可以将国家经济实力分类并评价，其中衡量经济实力的 4 个指标为人均 GDP，国家政府收入，贸易进口额和贸易出口额。再通过 KNN 算法作为决策判别器，可以对分类的结果做训练和预测，通过比较预测和实际的分类结果，证实决策判别器选择的合理性。

关键词：K-means 聚类算法，KNN 算法，国家经济实力评价

Abstract

Economy is an important part of measuring the country's overall performance. This paper can classify and evaluate the national economy through the K-means clustering algorithm. The four indicators for measuring economic strength are GDP per capita, national government revenue, trade import volume and trade export volume. By using the KNN algorithm as a decision classifier, the results of the classification can be trained and predicted, and the rationality of the choice of the decision maker can be confirmed by comparing the predicted and actual classification results.

Key words: K-means clustering algorithm, KNN algorithm, national economy evaluation

目 录

第一章 引言 5

第二章 基于 K-means 的分类统计 5

2．1 K-means 介绍 5

2．2 K-means 聚类在分类国家经济实力中的应用 7

第三章 基于 KNN 的决策判别器和实例分类判决检验 9

3．1 KNN 算法介绍 9

3．2 KNN 算法在国家经济实力判别与预测中的应用 9

3．3 朴素贝叶斯在国家经济实力判别与预测中的应用 10

3．4 随机森林算法在国家经济实力判别与预测中的应用 10

3．5 决策判别器的选择 11

第四章 结论 12

参考文献 13

附录 14

K-means 算法的 matlab 代码 14

KNN，朴素贝叶斯，随机森林算法 matlab 代码 18

第一章 引言

随着全球化的发展，国家间交流不断增强，贸易往来更加频繁，政府的收支收到国家经济实力的支配，那么我可以通过人均 GDP，国家政府收入，贸易进出口额来衡量各个国家的经济实力水平。

为了评价国家的经济水平，我在 IMF——国际货币基金组织中找到了相应的 172 个国家的数据，数据包含 GDP（十亿美元为单位），人口数量（百万位单位），国家进出口额（十亿美元为单位），国家出口额（十亿美元为单位），国家政府收入（占 GDP 的百分比）。数据集中有 172 个国家，包含大多数世界上常见的国家，但是比如索马里等较危险的国家并没有数据显示。通过分析这五个数据集可以将国家分为“超级发达”国家，“比较发达”国家，“发展中”国家和“贫困”国家。再通过设计决策判别器对分类的数据进行训练，测试剩余数据分类的正确率，证实决策判别器设计的可行性。

第二章 基于 K-means 的分类统计

2．1 K-means 介绍

K-means 是一种无监督学习算法，K-means 算法试图使数据分为 n 组，使 n 组集群间的方差相等，适合于大规模的适合的样本，在我选取的数据中十分适合使用。

K-means 聚类计算元素之间的距离是欧氏距离：

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

K-means 算法流程如下：

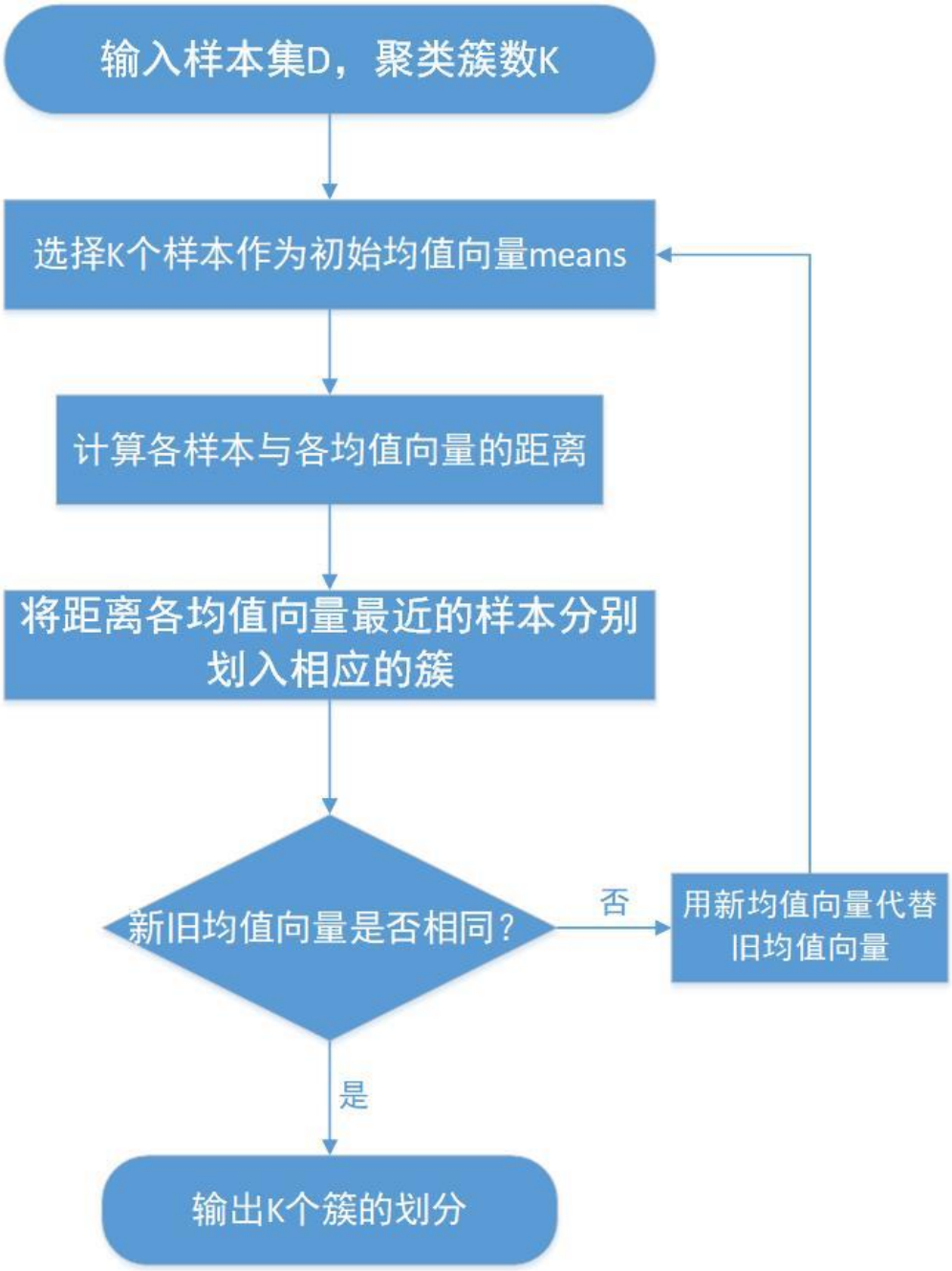

图 2.1 K-means 算法流程

2．2 K-means 聚类在分类国家经济实力中的应用

在 k-means 的实践中，我可以控制的超参数是聚类数量，这是最终的国家分类号码。同时，聚类中心的初始值对最终聚类结果有一定的影响，因此我使用多个随机初始中心来选择最合理的结果组作为聚类分组。

我首先调整了类别中的聚类数量。国际经济实力在国际上有几套指标。最简单的分为发达国家和发展中国家，最复杂的分为六个层次。这个问题的类别数量太少，并不能更准确地说明各国之间的差异程度，所以我分别试图对 3-6 个类别的 172 个国家进行分组。在实践中，我发现 5 个和 6 个集群都未能在有限的尝试中取得令人满意的结果。一些经济水平差异明显的国家出现在同一类别中，反映了由于集群类别太多而造成的不稳定性。然而，当类别 K 的数量是 3 或 4 时，数据开始显示出稳定的特征。大多数同一类别的国家都显示出类似的经济数据。

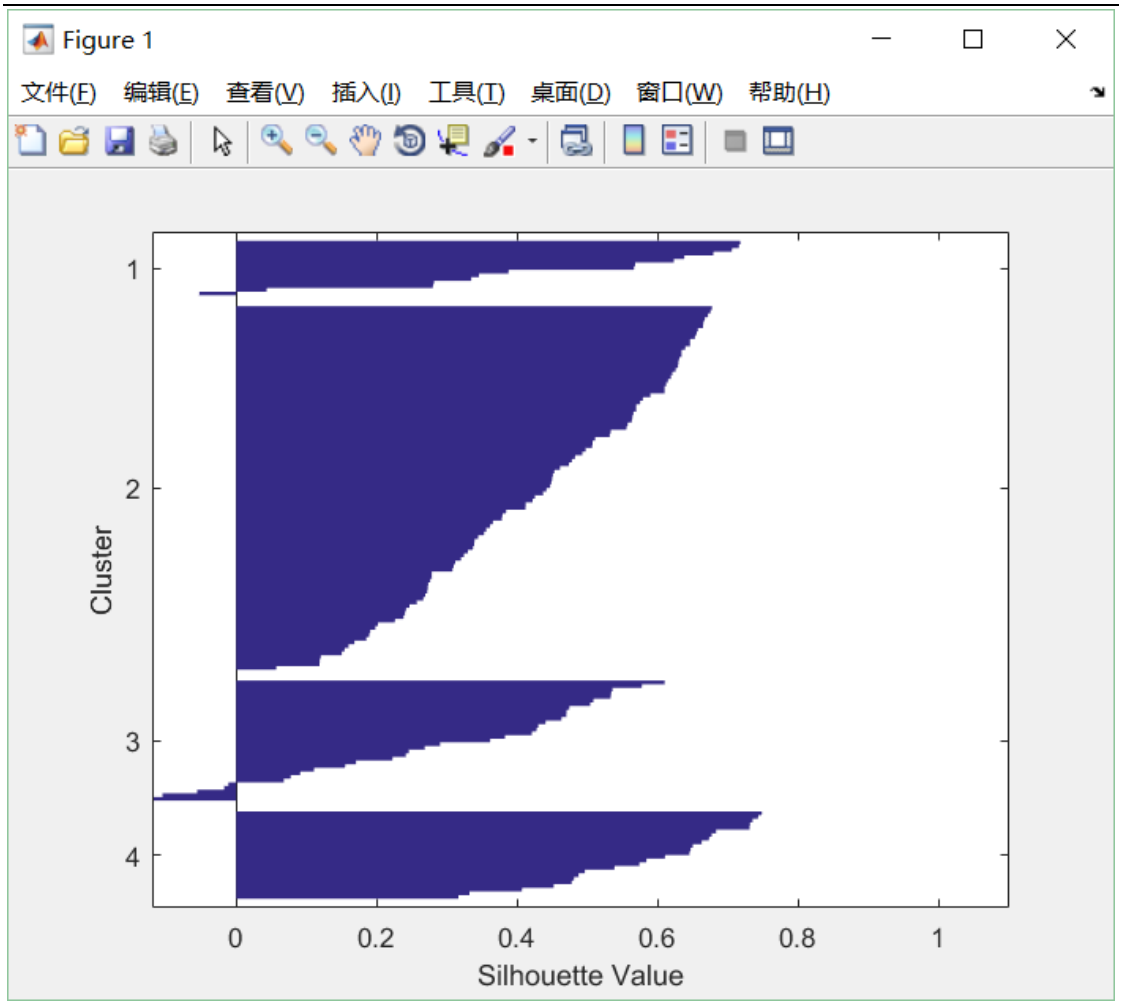

图 2.2 国家经济实力分类情况

在多次改变初始中心之后，我在 K = 4 的一组 K-means 聚类中取得了期望的结果。作为对这种模式的验证，我还寻找了各组中相互比较熟悉的国家，以便验证其经济状况是否合理。结果证实了聚类过程的合理性。例如，阿塞拜疆，利比里亚等国家被分为“贫穷”国家，中国，埃及被分为“发展中”国家，西班牙，土耳其被分为“较为发达”国家，美国，英国被分为“发达”国家。看分类结果，这十分合理。

第三章 基于 KNN 的决策判别器和实例分类判决检验

3．1 KNN 算法介绍

在检验分类决策器的适用性时，我用了三种方法，分别是：KNN，朴素贝叶斯和随机森林分类器，它们的预测正确性分别是 94.1176%，91.1765%，50%。所以最后我选择 KNN 算法作为决策判别器。

KNN 是通过测量不同特征值之间的距离进行分类。它的思路是：如果一个样本在特征空间中的 k 个最相似(即特征空间中最邻近)的样本中的大多数属于某一个类别，则该样本也属于这个类别，其中 K 通常是不大于 20 的整数。KNN 算法中，所选择的邻居都是已经正确分类的对象。该方法在定类决策上只依据最邻近的一个或者几个样本的类别来决定待分样本所属的类别。

KNN 算法如下：

对每一个未知点执行：

1. 计算未知点到所有已知类别点的距离
2. 按距离排序（升序）
3. 选取其中前 k 个与未知点离得最近的点
4. 统计 k 个点中各个类别的个数
5. 在上述 k 个点里类别出现频率最高的作为未知点的类别

3．2 KNN 算法在国家经济实力判别与预测中的应用

我将数据分为 80%和 20%，其中 80%作为训练数据，20%作为测试数据，然

后用 80%的已知分类数据预测剩余 20%的分类数据，将预测结果与实际结果作对比，发现在 KNN 分类算法下正确率高达 94.1176%，说明这个决策判别器选择的十分合理。

3．3 朴素贝叶斯在国家经济实力判别与预测中的应用

除了 KNN，比较常见的判别器还有朴素贝叶斯分类算法，随机森林分类算法，我也将这两个与 KNN 算法作了比较。

在朴素贝叶斯分类算法中，依据公式：

$$p(C_i|X) = \frac{P(X|C_i)p(C_i)}{p(X)}$$

其中，P（C_i | X）为后验概率，P（C_i）为先验概率，P（X | C_i）为条件概率。朴素贝叶斯的两个假设：1、属性之间相互独立。2、每个属性同等重要。通过假设 1 知，条件概率 P（X | C_i）可以简化为：

$$p(X|C_i) = \prod_{k=1}^{K=n} p(X_k|C_i)$$

朴素贝叶斯是选择具有最高后验概率作为确定类别的指标而对数据集过滤的算法。最终得到分类的正确率为 91.1765%。

3．4 随机森林算法在国家经济实力判别与预测中的应用

随机森林就是通过集成学习的思想将多棵树集成的一种算法，它的基本单元是决策树，而它的本质属于机器学习的一大分支——集成学习方法。

算法如下：

1. 给定训练集 S，测试集 T，特征维数 F。确定参数：使用到的 CART（分类回归树）的数量 t，每棵树的深度 d，每个节点使用到的特征数量 f，终止条件：节点上最少样本数 s，节点上最少的信息增益 m。对于第 1-t 棵树，i=1-t
2. 从 S 中有放回的抽取大小和 S 一样的训练集 S(i)，作为根节点的样本，从根节点开始训练
3. 如果当前节点上达到终止条件，则设置当前节点为叶子节点，如果是分类问题，该叶子节点的预测输出为当前节点样本集合中数量最多的那一类 c(j)，概率 p 为 c(j)占当前样本集的比例；如果是回归问题，预测输出为当前节点样本集各个样本值的平均值。然后继续训练其他节点。如果当前节点没有达到终止条件，则从 F 维特征中无放回的随机选取 f 维特征。利用这 f 维特征，寻找分类效果最好的一维特征 k 及其阈值 th，当前节点上样本第 k 维特征小于 th 的样本被划分到左节点，其余的被划分到右节点。继续训练其他节点。有关分类效果的评判标准在后面会讲。
4. 重复 2,3 直到所有节点都训练过了或者被标记为叶子节点。
5. 重复 2,3,4 直到所有 CART 都被训练过。

通过随机森林算法，最终得到的正确率为 50%

3.5 决策判别器的选择

通过对三种分类器的分析，可以看出 KNN 算法是对于此数据集最合适的决策判别器。正确率为 94.1176%。

第四章 结论

通过对各国的经济实力分析，用 K-means 可以将 172 个国家分为 4 类：贫困国家，发展中国家，较为发达国家和发达国家。之后分别用 KNN,朴素贝叶斯和随机森林算法作为决策判别器进行分类和预测正确率的计算，发现 KNN 的效果最好，最终选择 KNN 分类算法作为决策判别器，正确率高达 94.1176%。

参考文献

[1] 李弼程.模式识别原理与应用【M】.西安电子科技大学出版，2008.
[2] 杨淑莹.模式识别与智能计算——Matlab 技术实现【M】.电子工业出版社，2015

附录

K-means 算法的 matlab 代码

```
[x,textdata] = xlsread('E:\recognition\country.xlsx');
countryname = textdata(2:173,1);
GDPperson=x(:,1)./x(:,5);
xlswrite('E:\recognition\GDPperson.xlsx',GDPperson);
X=[x(:,1:2),x(:,4),x(:,6)];
X = zscore(X);

startdata = X([10,23,78,138],:);      % 选取第 10、第 23、第 78 和第 138 个观测为初始凝聚点
idx = kmeans(X,4,'Start',startdata);      % 设置初始凝聚点，进行 K 均值聚类
[S, H] = silhouette(X,idx);

countryname(idx == 1)%贫穷国家
ans =

  15×1 cell 数组

    'Azerbaijan'
    'Barbados'
    'Bhutan'
    'Botswana'
    'Comoros'
    'Republic of Congo'
    'Equatorial Guinea'
    'Liberia'
    'Nigeria'
    'Saudi Arabia'
    'St. Kitts and Nevis'
    'Sudan'
    'Trinidad and Tobago'
    'Turkmenistan'
    'Venezuela'

countryname(idx == 2)%发展中国家
ans =

  100×1 cell 数组

    'Afghanistan'
```

'Albania'
'Algeria'
'Angola'
'Antigua and Barbuda'
'Armenia'
'The Bahamas'
'Bahrain'
'Bangladesh'
'Belize'
'Benin'
'Bolivia'
'Brazil'
'Brunei Darussalam'
'Burkina Faso'
'Burundi'
'Cabo Verde'
'Cambodia'
'Cameroon'
'Central African Republic'
'Chad'
'Chile'
'China'
'Colombia'
'Democratic Republic of the Congo'
'Costa Rica'
'Côte d'Ivoire'
'Djibouti'
'Dominican Republic'
'Egypt'
'El Salvador'
'Eritrea'
'Ethiopia'
'Gabon'
'The Gambia'
'Georgia'
'Ghana'
'Grenada'
'Guatemala'
'Guinea'
'Guinea-Bissau'
'Guyana'
'Haiti'
'Honduras'
'India'

'Indonesia'
'Islamic Republic of Iran'
'Jamaica'
'Jordan'
'Kazakhstan'
'Kenya'
'Korea'
'Kosovo'
'Lao P.D.R.'
'Lebanon'
'FYR Macedonia'
'Madagascar'
'Malawi'
'Malaysia'
'Maldives'
'Mali'
'Mauritania'
'Mauritius'
'Mexico'
'Morocco'
'Mozambique'
'Myanmar'
'Namibia'
'Nicaragua'
'Niger'
'Oman'
'Pakistan'
'Panama'
'Papua New Guinea'
'Paraguay'
'Peru'
'Philippines'
'Romania'
'Rwanda'
'São Tomé and Príncipe'
'Senegal'
'Sierra Leone'
'South Africa'
'Sri Lanka'
'St. Lucia'
'St. Vincent and the Grenadines'
'Suriname'
'Swaziland'
'Taiwan Province of China'

'Tajikistan'
'Tanzania'
'Togo'
'Tunisia'
'Uganda'
'Uruguay'
'Uzbekistan'
'Vietnam'
'Yemen'
'Zambia'
'Zimbabwe'

countryname(idx == 3)%较为发达的国家
ans =

33×1 cell 数组

'Argentina'
'Belarus'
'Bosnia and Herzegovina'
'Bulgaria'
'Croatia'
'Cyprus'
'Czech Republic'
'Dominica'
'Ecuador'
'Estonia'
'Greece'
'Hungary'
'Italy'
'Kuwait'
'Kyrgyz Republic'
'Latvia'
'Lesotho'
'Libya'
'Malta'
'Moldova'
'Mongolia'
'Montenegro'
'Poland'
'Portugal'
'Russia'
'Serbia'
'Seychelles'

```
    'Slovak Republic'
    'Slovenia'
    'Solomon Islands'
    'Spain'
    'Turkey'
    'Ukraine'

countryname(idx == 4)%发达国家
ans =

  24×1 cell 数组

    'Australia'
    'Austria'
    'Belgium'
    'Canada'
    'Denmark'
    'Finland'
    'France'
    'Germany'
    'Hong Kong SAR'
    'Iceland'
    'Ireland'
    'Israel'
    'Japan'
    'Luxembourg'
    'Netherlands'
    'New Zealand'
    'Norway'
    'Qatar'
    'Singapore'
    'Sweden'
    'Switzerland'
    'United Arab Emirates'
    'United Kingdom'
    'United States'
```

KNN，朴素贝叶斯，随机森林算法 matlab 代码

```
xlswrite('E:\recognition\number.xlsx',idx);
label=xlsread('E:\recognition\number.xlsx');
x=[x,label];
```

```
data=[x(:,1:2),x(:,4),x(:,6)];
traindata=data(1:138,:);%训练数据
testdata=data(139:172,:);%测试数据
trainlabel=label(1:138,:);%训练分类
%朴素贝叶斯分类器（Naive Bayes）
testlabel=label(139:172,:);
nb = fitcnb(traindata, trainlabel);
predictlabel     = predict(nb, testdata);
accuracy = length(find(predictlabel == testlabel))/length(testlabel)*100;
accuracy
accuracy =
    91.1765
%K 近邻分类器 （KNN）
mdl = ClassificationKNN.fit(traindata,trainlabel,'NumNeighbors',1);
predictlabel2     = predict(mdl, testdata);
accuracy2    =length(find(predictlabel2 == testlabel))/length(testlabel)*100
accuracy2 =
     94.1176
%随机森林分类器（Random Forest）
B = TreeBagger(4,traindata,trainlabel);
predictlabel3 = predict(B,testdata);
m=0;
n=0;
for i=1:17
        if predictlabel3{i,1}>0
              m=m+1;
        end
        if predictlabel3{i+17,1}<0
             n=n+1;
        end
 end
s=m+n;
accuracy3=s/34*100
accuracy3=
50
```