# 法律声明

- 本课件包括演示文稿、示例、代码、题库、视频和声音等内容，小象学院和主讲老师拥有完全知识产权的权利；只限于善意学习者在本课程使用，不得在课程范围外向任何第三方散播。任何其他人或机构不得盗版、复制、仿造其中的创意及内容，我们保留一切通过法律手段追究违反者的权利。

- 课程详情请咨询
  - 微信公众号：小象
  - 新浪微博：ChinaHadoop

# 决策树和随机森林实践

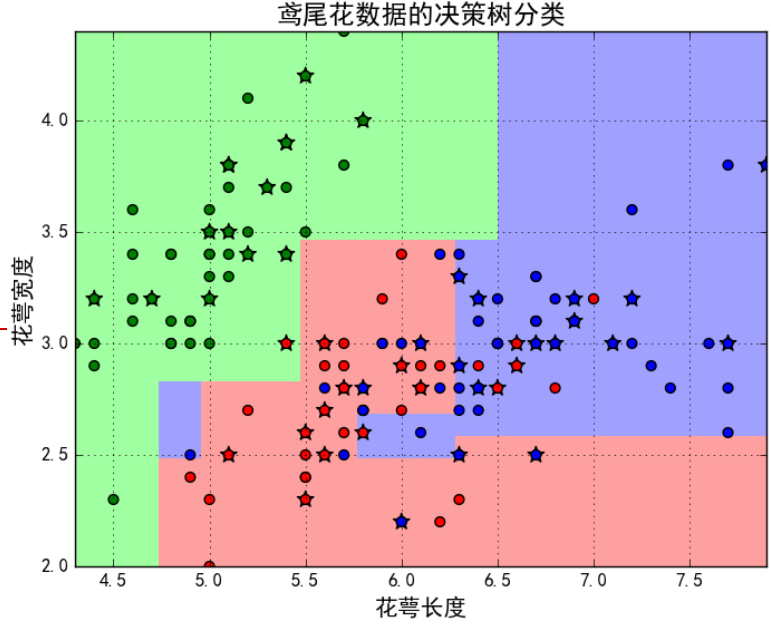

邹博

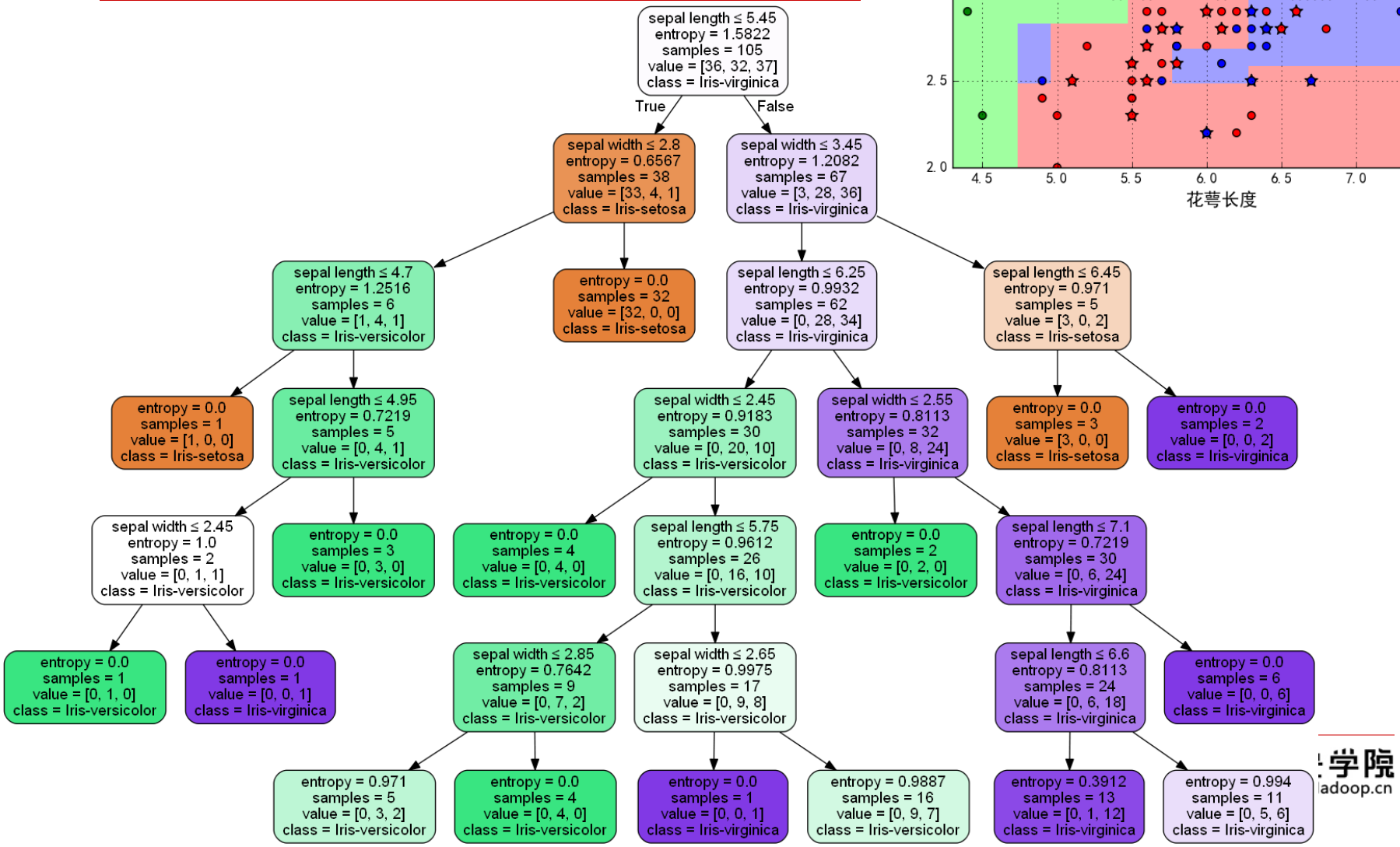
# 主要内容

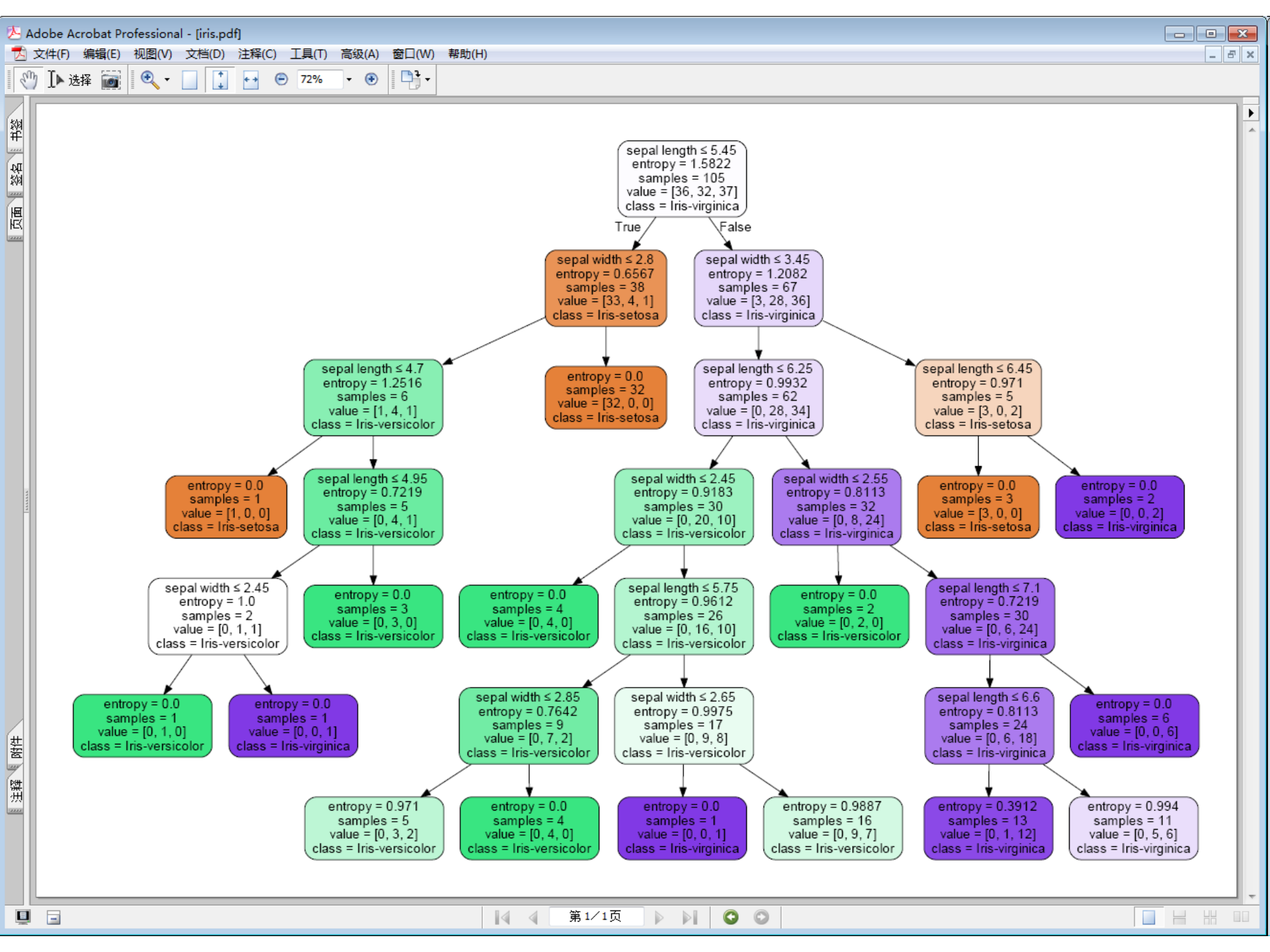
- 决策树分类鸢尾花数据
- 随机森林分类鸢尾花数据
- 决策树的可视化
- 决策树回归
- 多输出决策树

- 思考：树的深度与过拟合

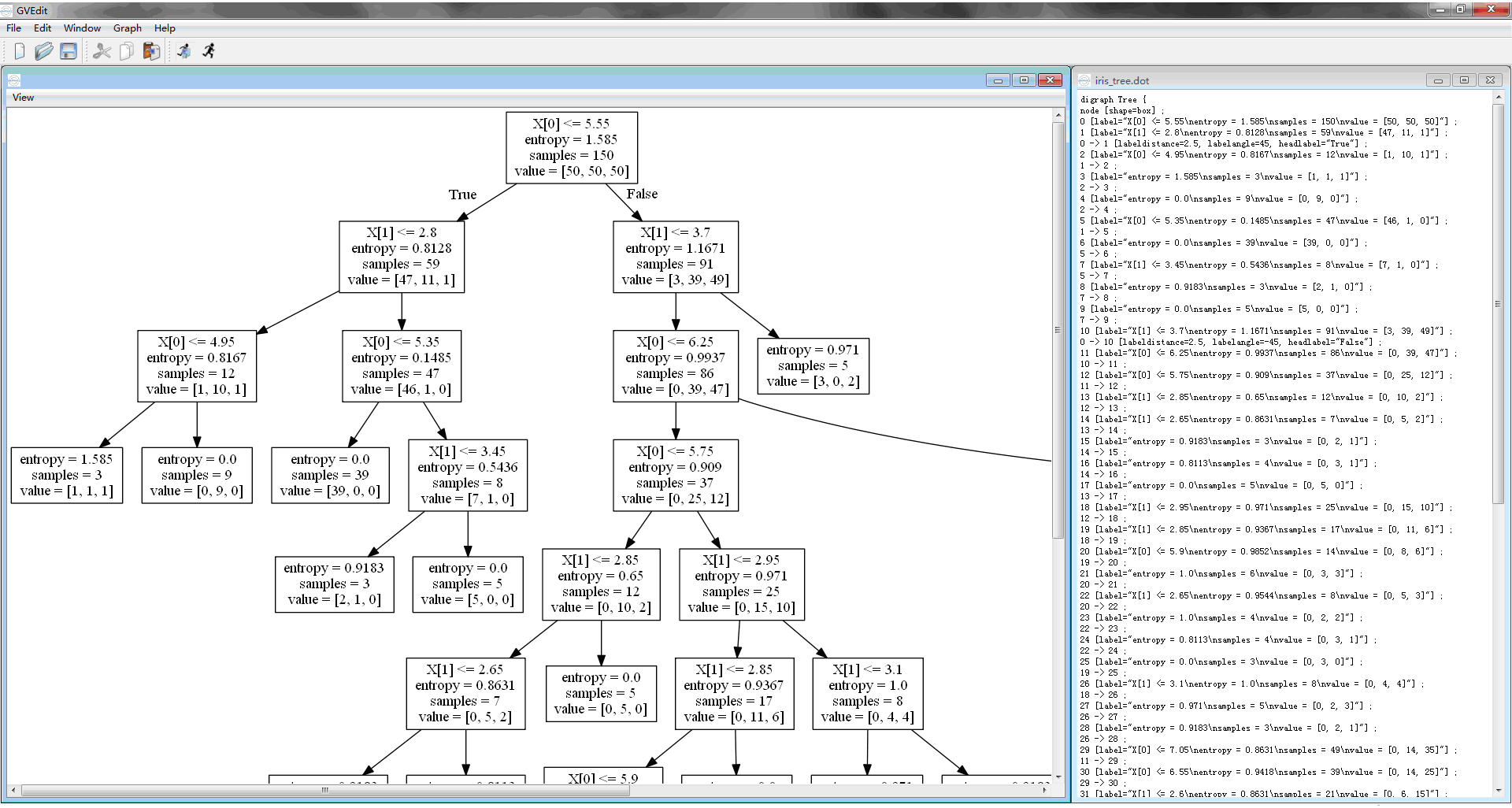
# 鸢尾花数据决策树



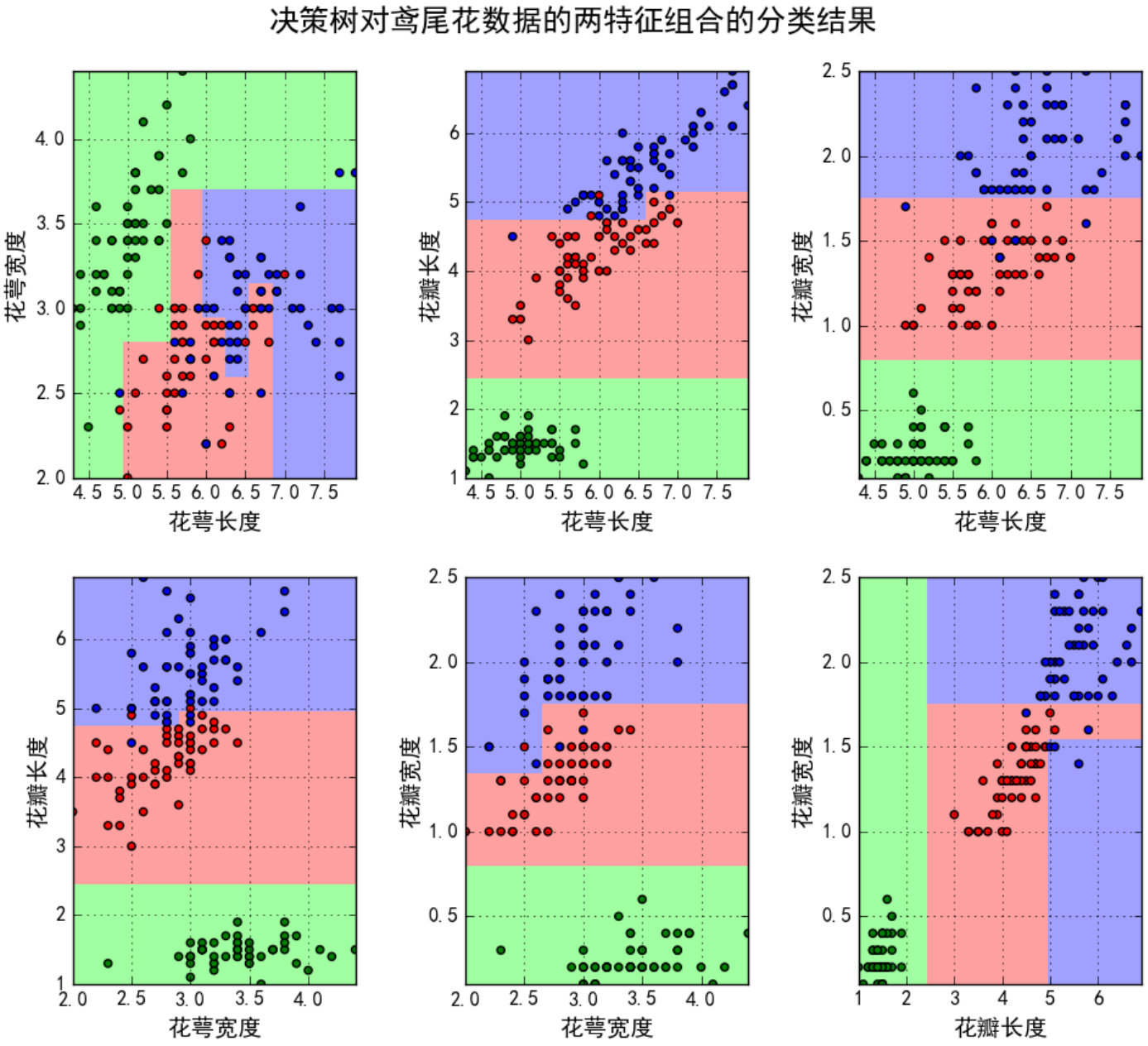




Adobe Acrobat Professional - [iris.pdf]
文件(F) 编辑(E) 视图(V) 文档(D) 注释(C) 工具(T) 高级(A) 窗口(W) 帮助(H)
选择
72%
书签
签名
页面
附件
注释
sepal length ≤ 5.45
entropy = 1.5822
samples = 105
value = [36, 32, 37]
class = Iris-virginica
True
False
sepal width ≤ 2.8
entropy = 0.6567
samples = 38
value = [33, 4, 1]
class = Iris-setosa
sepal width ≤ 3.45
entropy = 1.2082
samples = 67
value = [3, 28, 36]
class = Iris-virginica
sepal length ≤ 4.7
entropy = 1.2516
samples = 6
value = [1, 4, 1]
class = Iris-versicolor
entropy = 0.0
samples = 32
value = [32, 0, 0]
class = Iris-setosa
sepal length ≤ 6.25
entropy = 0.9932
samples = 62
value = [0, 28, 34]
class = Iris-virginica
sepal length ≤ 6.45
entropy = 0.971
samples = 5
value = [3, 0, 2]
class = Iris-setosa
entropy = 0.0
samples = 1
value = [1, 0, 0]
class = Iris-setosa
sepal length ≤ 4.95
entropy = 0.7219
samples = 5
value = [0, 4, 1]
class = Iris-versicolor
sepal width ≤ 2.45
entropy = 0.9183
samples = 30
value = [0, 20, 10]
class = Iris-versicolor
sepal width ≤ 2.55
entropy = 0.8113
samples = 32
value = [0, 8, 24]
class = Iris-virginica
entropy = 0.0
samples = 3
value = [3, 0, 0]
class = Iris-setosa
entropy = 0.0
samples = 2
value = [0, 0, 2]
class = Iris-virginica
sepal width ≤ 2.45
entropy = 1.0
samples = 2
value = [0, 1, 1]
class = Iris-versicolor
entropy = 0.0
samples = 3
value = [0, 3, 0]
class = Iris-versicolor
entropy = 0.0
samples = 4
value = [0, 4, 0]
class = Iris-versicolor
sepal length ≤ 5.75
entropy = 0.9612
samples = 26
value = [0, 16, 10]
class = Iris-versicolor
entropy = 0.0
samples = 2
value = [0, 2, 0]
class = Iris-versicolor
sepal length ≤ 7.1
entropy = 0.7219
samples = 30
value = [0, 6, 24]
class = Iris-virginica
entropy = 0.0
samples = 1
value = [0, 1, 0]
class = Iris-versicolor
entropy = 0.0
samples = 1
value = [0, 0, 1]
class = Iris-virginica
sepal width ≤ 2.85
entropy = 0.7642
samples = 9
value = [0, 7, 2]
class = Iris-versicolor
sepal width ≤ 2.65
entropy = 0.9975
samples = 17
value = [0, 9, 8]
class = Iris-versicolor
sepal length ≤ 6.6
entropy = 0.8113
samples = 24
value = [0, 6, 18]
class = Iris-virginica
entropy = 0.0
samples = 6
value = [0, 0, 6]
class = Iris-virginica
entropy = 0.971
samples = 5
value = [0, 3, 2]
class = Iris-versicolor
entropy = 0.0
samples = 4
value = [0, 4, 0]
class = Iris-versicolor
entropy = 0.0
samples = 1
value = [0, 0, 1]
class = Iris-virginica
entropy = 0.9887
samples = 16
value = [0, 9, 7]
class = Iris-versicolor
entropy = 0.3912
samples = 13
value = [0, 1, 12]
class = Iris-virginica
entropy = 0.994
samples = 11
value = [0, 5, 6]
class = Iris-virginica
第 1／1 页

# Graphviz

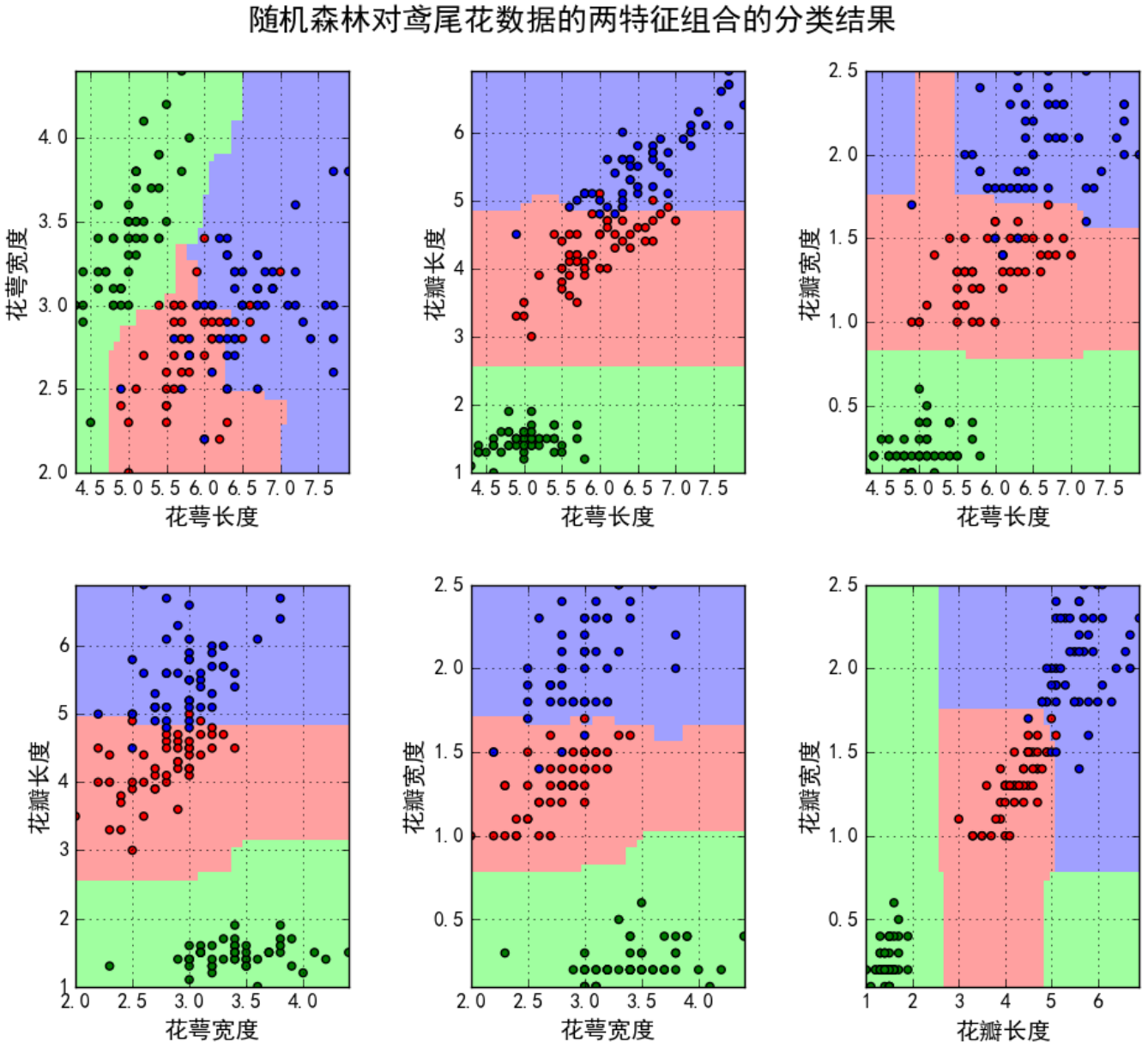

GVEdit
File Edit Window Graph Help
View
X[0] <= 5.55
entropy = 1.585
samples = 150
value = [50, 50, 50]
True
False
X[1] <= 2.8
entropy = 0.8128
samples = 59
value = [47, 11, 1]
X[1] <= 3.7
entropy = 1.1671
samples = 91
value = [3, 39, 49]
X[0] <= 4.95
entropy = 0.8167
samples = 12
value = [1, 10, 1]
X[0] <= 5.35
entropy = 0.1485
samples = 47
value = [46, 1, 0]
X[0] <= 6.25
entropy = 0.9937
samples = 86
value = [0, 39, 47]
entropy = 0.971
samples = 5
value = [3, 0, 2]
entropy = 1.585
samples = 3
value = [1, 1, 1]
entropy = 0.0
samples = 9
value = [0, 9, 0]
entropy = 0.0
samples = 39
value = [39, 0, 0]
X[1] <= 3.45
entropy = 0.5436
samples = 8
value = [7, 1, 0]
X[0] <= 5.75
entropy = 0.909
samples = 37
value = [0, 25, 12]
entropy = 0.9183
samples = 3
value = [2, 1, 0]
entropy = 0.0
samples = 5
value = [5, 0, 0]
X[1] <= 2.85
entropy = 0.65
samples = 12
value = [0, 10, 2]
X[1] <= 2.95
entropy = 0.971
samples = 25
value = [0, 15, 10]
X[1] <= 2.65
entropy = 0.8631
samples = 7
value = [0, 5, 2]
entropy = 0.0
samples = 5
value = [0, 5, 0]
X[1] <= 2.85
entropy = 0.9367
samples = 17
value = [0, 11, 6]
X[1] <= 3.1
entropy = 1.0
samples = 8
value = [0, 4, 4]
iris_tree.dot
digraph Tree {
node [shape=box] ;
0 [label="X[0] <= 5.55\nentropy = 1.585\nsamples = 150\nvalue = [50, 50, 50]"] ;
1 [label="X[1] <= 2.8\nentropy = 0.8128\nsamples = 59\nvalue = [47, 11, 1]"] ;
0 -> 1 [labeldistance=2.5, labelangle=45, headlabel="True"] ;
2 [label="X[0] <= 4.95\nentropy = 0.8167\nsamples = 12\nvalue = [1, 10, 1]"] ;
1 -> 2 ;
3 [label="entropy = 1.585\nsamples = 3\nvalue = [1, 1, 1]"] ;
2 -> 3 ;
4 [label="entropy = 0.0\nsamples = 9\nvalue = [0, 9, 0]"] ;
2 -> 4 ;
5 [label="X[0] <= 5.35\nentropy = 0.1485\nsamples = 47\nvalue = [46, 1, 0]"] ;
1 -> 5 ;
6 [label="entropy = 0.0\nsamples = 39\nvalue = [39, 0, 0]"] ;
5 -> 6 ;
7 [label="X[1] <= 3.45\nentropy = 0.5436\nsamples = 8\nvalue = [7, 1, 0]"] ;
5 -> 7 ;
8 [label="entropy = 0.9183\nsamples = 3\nvalue = [2, 1, 0]"] ;
7 -> 8 ;
9 [label="entropy = 0.0\nsamples = 5\nvalue = [5, 0, 0]"] ;
7 -> 9 ;
10 [label="X[1] <= 3.7\nentropy = 1.1671\nsamples = 91\nvalue = [3, 39, 49]"] ;
0 -> 10 [labeldistance=2.5, labelangle=-45, headlabel="False"] ;
11 [label="X[0] <= 6.25\nentropy = 0.9937\nsamples = 86\nvalue = [0, 39, 47]"] ;
10 -> 11 ;
12 [label="X[0] <= 5.75\nentropy = 0.909\nsamples = 37\nvalue = [0, 25, 12]"] ;
11 -> 12 ;
13 [label="X[1] <= 2.85\nentropy = 0.65\nsamples = 12\nvalue = [0, 10, 2]"] ;
12 -> 13 ;
14 [label="X[1] <= 2.65\nentropy = 0.8631\nsamples = 7\nvalue = [0, 5, 2]"] ;
13 -> 14 ;
15 [label="entropy = 0.9183\nsamples = 3\nvalue = [0, 2, 1]"] ;
14 -> 15 ;
16 [label="entropy = 0.8113\nsamples = 4\nvalue = [0, 3, 1]"] ;
14 -> 16 ;
17 [label="entropy = 0.0\nsamples = 5\nvalue = [0, 5, 0]"] ;
13 -> 17 ;
18 [label="X[1] <= 2.95\nentropy = 0.971\nsamples = 25\nvalue = [0, 15, 10]"] ;
12 -> 18 ;
19 [label="X[1] <= 2.85\nentropy = 0.9367\nsamples = 17\nvalue = [0, 11, 6]"] ;
18 -> 19 ;
20 [label="X[0] <= 5.9\nentropy = 0.9852\nsamples = 14\nvalue = [0, 8, 6]"] ;
19 -> 20 ;
21 [label="entropy = 1.0\nsamples = 6\nvalue = [0, 3, 3]"] ;
20 -> 21 ;
22 [label="X[1] <= 2.65\nentropy = 0.9544\nsamples = 8\nvalue = [0, 5, 3]"] ;
20 -> 22 ;
23 [label="entropy = 1.0\nsamples = 4\nvalue = [0, 2, 2]"] ;
22 -> 23 ;
24 [label="entropy = 0.8113\nsamples = 4\nvalue = [0, 3, 1]"] ;
22 -> 24 ;
25 [label="entropy = 0.0\nsamples = 3\nvalue = [0, 3, 0]"] ;
19 -> 25 ;
26 [label="X[1] <= 3.1\nentropy = 1.0\nsamples = 8\nvalue = [0, 4, 4]"] ;
18 -> 26 ;
27 [label="entropy = 0.971\nsamples = 5\nvalue = [0, 2, 3]"] ;
26 -> 27 ;
28 [label="entropy = 0.9183\nsamples = 3\nvalue = [0, 2, 1]"] ;
26 -> 28 ;
29 [label="X[0] <= 7.05\nentropy = 0.8631\nsamples = 49\nvalue = [0, 14, 35]"] ;
11 -> 29 ;
30 [label="X[0] <= 6.55\nentropy = 0.9418\nsamples = 39\nvalue = [0, 14, 25]"] ;
29 -> 30 ;
31 [label="X[1] <= 2.6\nentropy = 0.8631\nsamples = 21\nvalue = [0, 6, 15]"] ;

# 决策树

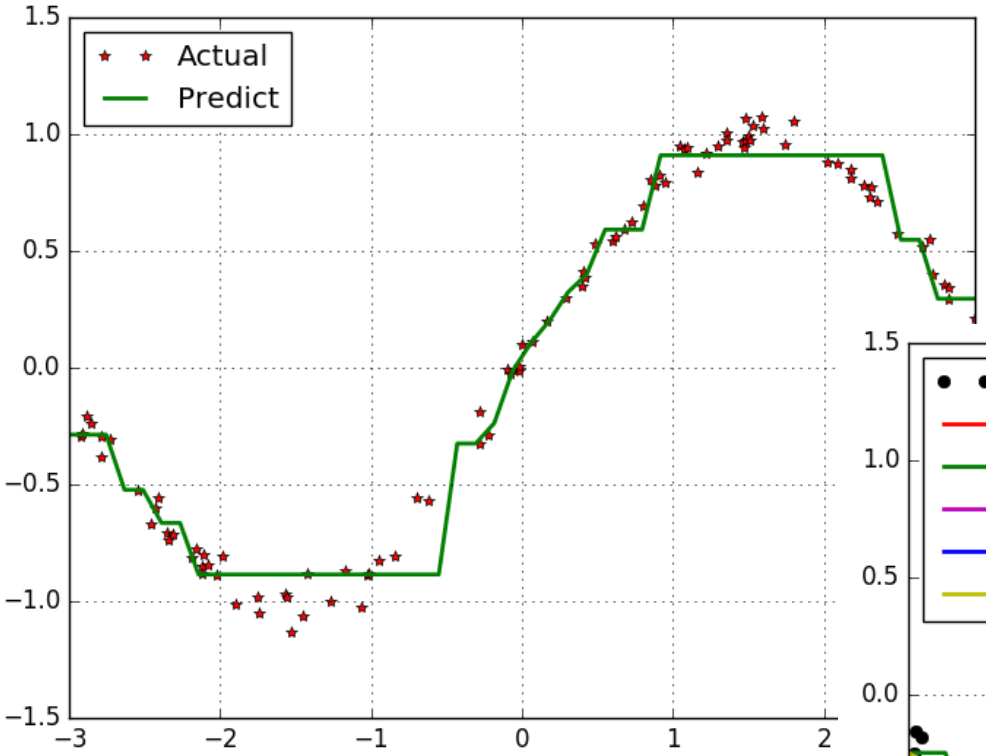

决策树对鸢尾花数据的两特征组合的分类结果
花萼宽度
花萼长度
花瓣长度
花萼长度
花瓣宽度
花萼长度
花瓣长度
花萼宽度
花瓣宽度
花萼宽度
花瓣宽度
花瓣宽度
花瓣长度

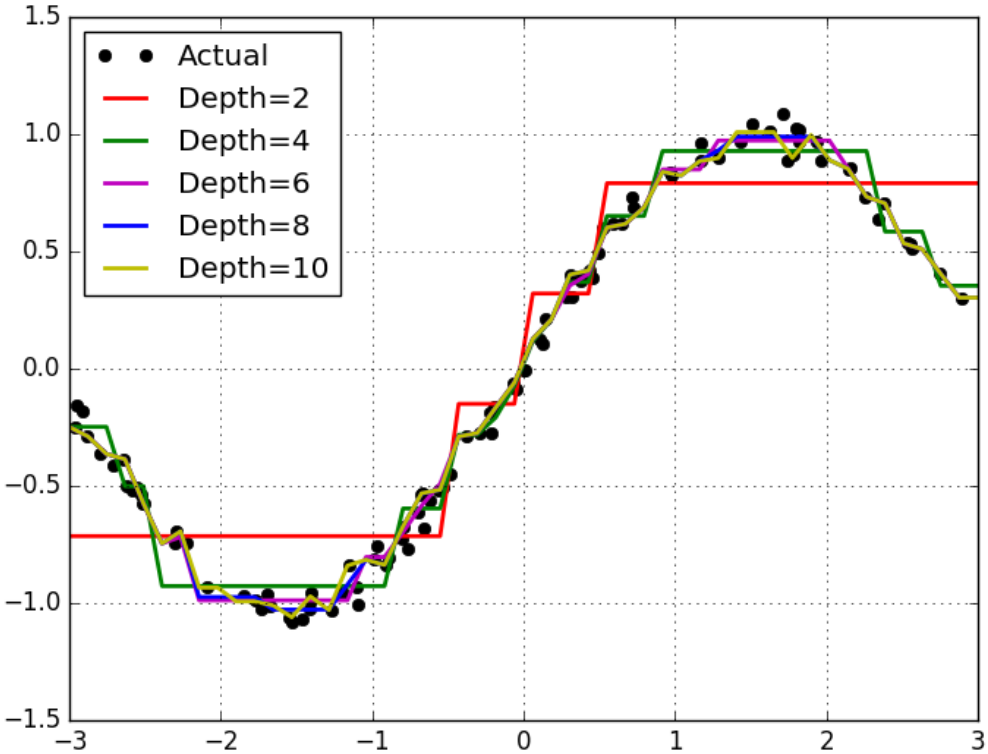
# 随机森林

- 50

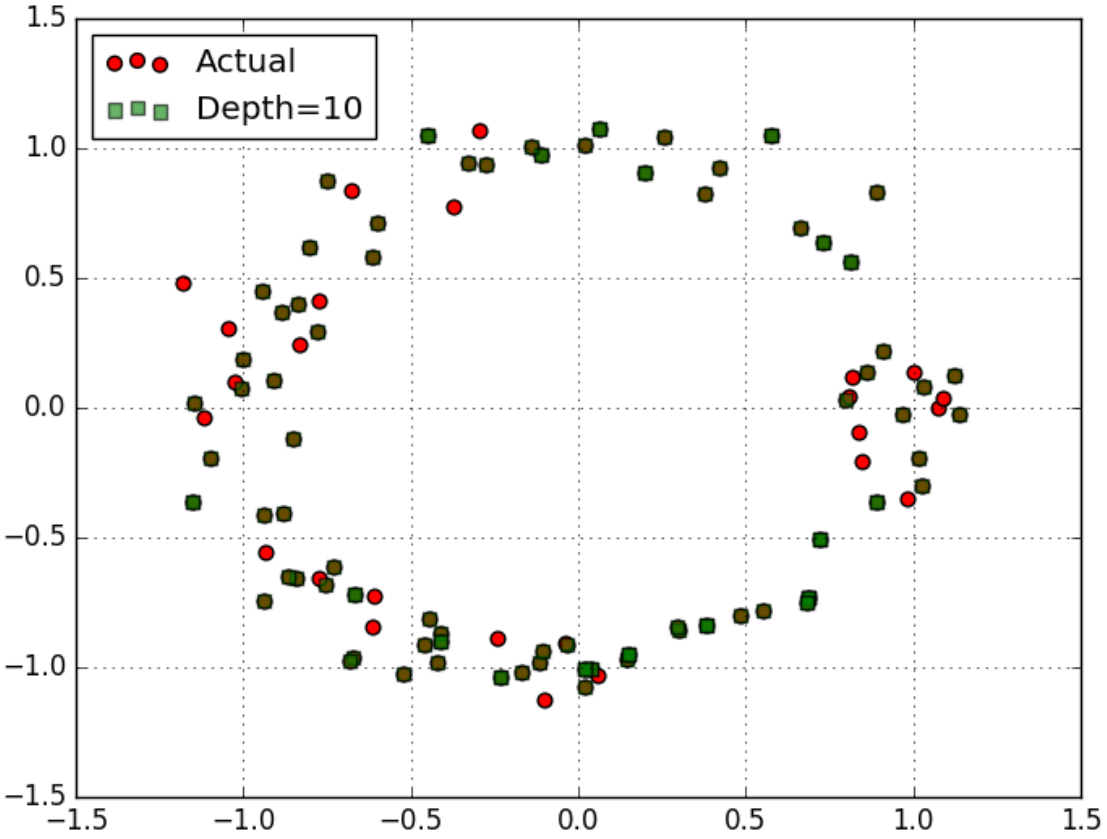

随机森林对鸢尾花数据的两特征组合的分类结果
花萼宽度
花萼长度
花瓣长度
花萼长度
花瓣宽度
花萼长度
花瓣长度
花萼宽度
花瓣宽度
花萼宽度
花瓣宽度
花瓣长度

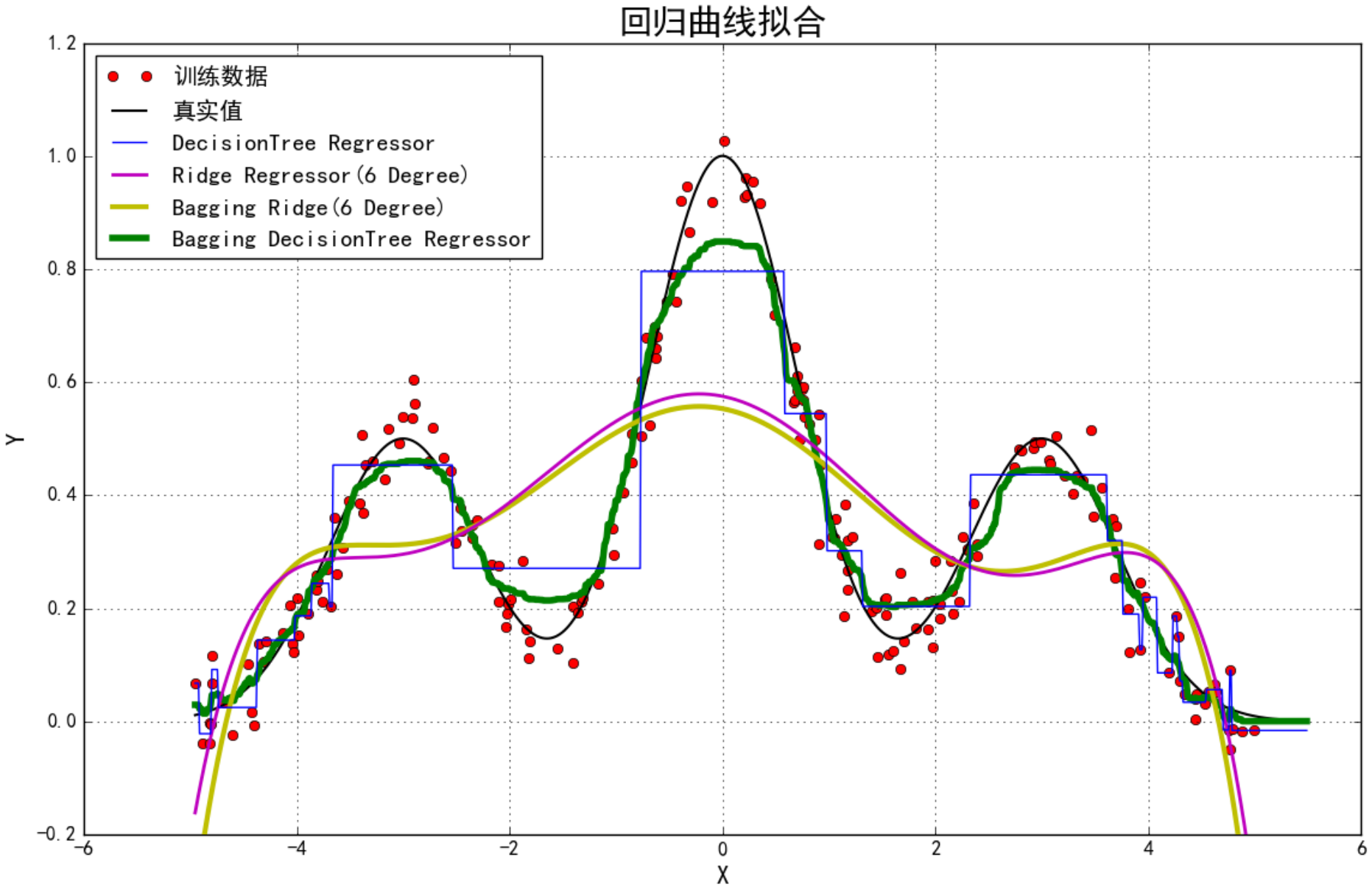
# 决策树用于拟合


Actual
Predict
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−3
−2
−1
0
1
2


Actual
Depth=2
Depth=4
Depth=6
Depth=8
Depth=10
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−3
−2
−1
0
1
2
3

# 多输出的决策树回归




回归曲线拟合
训练数据
真实值
DecisionTree Regressor
Ridge Regressor(6 Degree)
Bagging Ridge(6 Degree)
Bagging DecisionTree Regressor
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
Y
-6
-4
-2
0
2
4
6
X

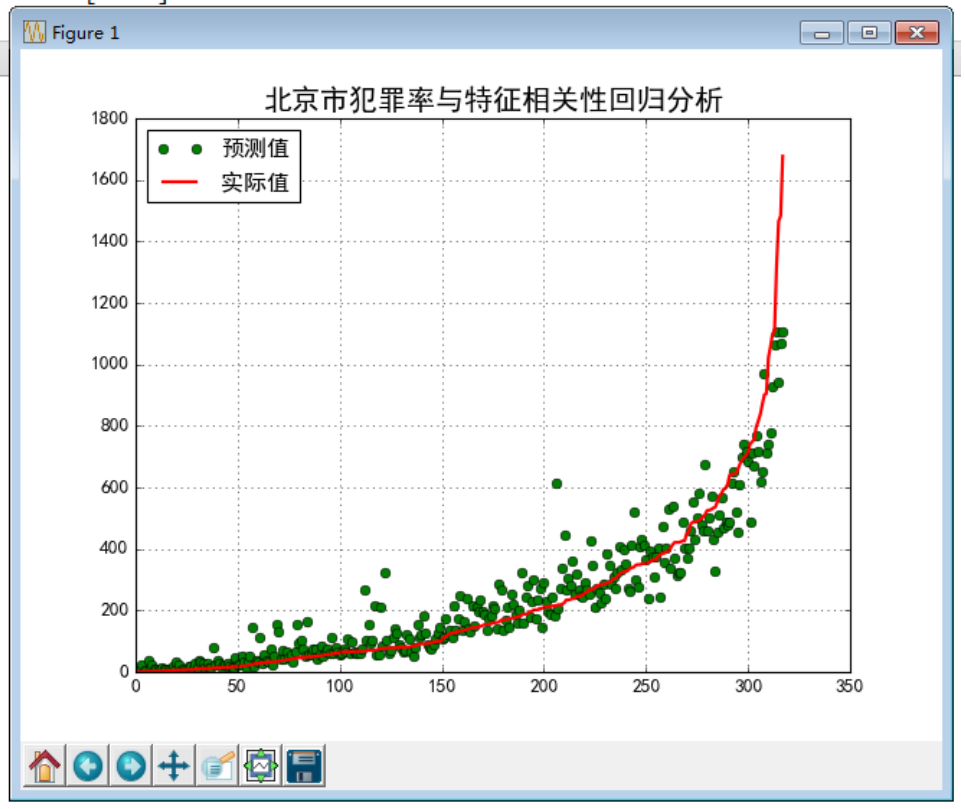
# 再谈北京市区域犯罪率分析

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 地区 | 盗窃案件数 | 批发和零售业数量 | 交通运输仓储邮政业数量 | 房地产业数量 | 住宿和餐饮业数量 | 卫生和社会工作数量 | 居民服务修理服务业数量 | 大型单位数量 | 中型单位数量 | 小微单位数量 | 金融业单位数量 | 液化石油气 | 能源合计 | 从业人员 | 销售费用 | 营业收入 | 营业税及附加 | 总产值 | 利润总额 | 人员支出 |
| 2 | 安定门街道办事处 | | | | | | | | | | | | | | | | | | | | |
| 3 | 安定镇 | | | | | | | | | | | | | | | | | | | | |
| 4 | 安贞街道办事处 | | | | | | | | | | | | | | | | | | | | |
| 5 | 奥运村街道办事处 | | | | | | | | | | | | | | | | | | | | |
| 6 | 八宝山街道办事处 | | | | | | | | | | | | | | | | | | | | |
| 7 | 八达岭镇 | | | | | | | | | | | | | | | | | | | | |
| 8 | 八角街道办事处 | | | | | | | | | | | | | | | | | | | | |
| 9 | 八里庄街道办事处(朝阳) | | | | | | | | | | | | | | | | | | | | |
| 10 | 八里庄街道办事处（海淀） | | | | | | | | | | | | | | | | | | | | |
| 11 | 白纸坊街道办事处 | | | | | | | | | | | | | | | | | | | | |
| 12 | 百泉街道办事处 | | | | | | | | | | | | | | | | | | | | |
| 13 | 百善镇 | | | | | | | | | | | | | | | | | | | | |
| 14 | 宝山镇 | | | | | | | | | | | | | | | | | | | | |
| 15 | 北房镇 | | | | | | | | | | | | | | | | | | | | |
| 16 | 北京经济技术开发区 | | | | | | | | | | | | | | | | | | | | |
| 17 | 北七家镇 | | | | | | | | | | | | | | | | | | | | |
| 18 | 北石槽镇 | | | | | | | | | | | | | | | | | | | | |
| 19 | 北太平庄街道办事处 | | | | | | | | | | | | | | | | | | | | |
| 20 | 北务镇 | | | | | | | | | | | | | | | | | | | | |
| 21 | 北下关街道办事处 | | | | | | | | | | | | | | | | | | | | |
| 22 | 北小营镇 | | | | | | | | | | | | | | | | | | | | |
| 23 | 北新桥街道办事处 | | | | | | | | | | | | | | | | | | | | |

# Code

```
model = RandomForestRegressor(n_estimators=100, criterion='mse', max_depth=10, min_samples_split=5,
                              max_features=0.6, oob_score=True)
model.fit(x, y)
print 'OOB Score = ', model.oob_score_
y_hat = model.predict(x)
rmse = np.sqrt(np.mean((y_hat - y)**2))
print 'RMSE = ', rmse, 'Predict Score = ', rmse / np.mean(y)
feature_importances = np.array(zip(columns, model.feature_importances_))
feature_importances[:, 1] = feature_importances[:, 1].astype(np.float)
feature_importances.sort(axis=0)
feature_importances = feature_importances[::-1]
for fi in feature_importances:
```

```
房地产业数量 0.0240475164437
总产值 0.0196562211564
居民服务修理服务业数量 0.0194228549579
小微单位数量 0.0187625408241
大型单位数量 0.0184318382301
地铁线路 0.0182858907058
地铁站 0.016660182329
卫生和社会工作数量 0.0144664713616
劳务费 0.0137140251708
利润总额 0.0128121358596
公用支出 0.0113859310541
公交线路 0.0113687880457
公交站 0.0112938309575
住宿和餐饮业数量 0.0107283288012
从业人员 0.00870104496645
人员支出 0.00541540683171
交通运输仓储邮政业数量 0.0053253206699
中型单位数量 0.00385674089549
```



# 作业

- 使用决策树做任意数据集的分类。
  - 离散变量
- 使用随机森林做数据回归。
  - 连续变量

# 我们在这里

- http://wenda.ChinaHadoop.cn
  - 视频/课程/社区
- 微博
  - @ChinaHadoop
  - @邹博_机器学习
- 微信公众号
  - 小象
  - 大数据分析挖掘

感谢大家！

恳请大家批评指正！