决策树算法实现实验报告

201250070 郁博文

一、 数据预处理

首先将数据里，含有垃圾值的记录删除。

例如如下情况：

73644	Ethinyl estradiol / norethindrone	<null>	"I was on Loestrin 24 Fe for about 3 months. I just stopped taking it didn't even finish the p
68868	Medroxyprogesterone	<null>	"i was offered this by my doctor due to excessively heavy periods(i am 51 and possibly starting mer
200952	Alcaftadine	<null>	"The product sample worked great. However, when I went to fill my prescription, even WITH insurance
169862	Amitriptyline	<null>	"I used to get migraines 3-5 times a week, one of which would usually be severe.↵↵Elavil reduced my
161656	Diphenhydramine	<null>	"I've been taking nytol for a week now and they do work for trouble sleeping as I work night s
73649	Ethinyl estradiol / norethindrone	<null>	"I was on Junel FE 1/20 and randomly got switched to this without any notice. Since it was a bit ch
123077	Ethinyl estradiol / norgestrel	<null>	"I really like this pill. At first I noticed my skin become more oily, but then that went away. I h
105	Medroxyprogesterone	<null>	"I have liked depo, but today I started bleeding for the first time in a while, I was told I probab
20015	Pregabalin	<null>	"it was awful i used it for one day because of the pain i am in i was horribly sick i was throwing
224609	Levothyroxine	<null>	"Synthroid was horrible. I had severe fatigue, hair loss, dry skin, constipation, weight gain, hig
7196	Aluminum chloride hexahydrate	<null>	"I have been sweating for 3 years, even if it was 30 degrees outside and I wasn't wearing a ja
133232	Tri-Sprintec	0 users found this comment helpful.	"I have been off of birth control for 2 years. Started back on Tri-Sprintec. I have not stopped hav
221271	Drysol	0 users found this comment helpful.	"I'm 27 and I have been using Drysol since I was 21. I found deodorants and antiperspirants c
133358	Tri-Sprintec	10 users found this comment helpfu…	"Second birth control I've been on, and I've been on it for over a year. It gives me the
33560	Microgestin Fe 1 / 20	12 users found this comment helpfu…	"I was really worried at first because of all the negative comments. I just finished my first month
74783	Keppra	13 users found this comment helpfu…	"I started seizures at 55. I am Bipolar, difficult to treat, rapid cycling I take a low dosage of K
147675	Aviane	15 users found this comment helpfu…	"I started using birth control when I was 13 years old and am now 19 1/2. Aviane was the first thir
201734	Zoloft	15 users found this comment helpfu…	"After having a baby my anxiety increased dramatically. Anxiety attacks all the time. Didn't w
42886	Xulane	1 users found this comment helpful.	"I went on birth control due to irregular heavy bleeding and excruciating cramps. I have no prior
28848	Lexapro	1 users found this comment helpful.	"16 female. I've been under the care of a psychiatric adolescent ward for about 5 months now,
66762	Seroquel	24 users found this comment helpfu…	"I've been taking this since January 2013 for depression and insomnia it has worked wonders. E
169450	Vivarin	25 users found this comment helpfu…	"Never got used to drinking coffee, my stomach could not really stand it and energy drinks did only
98290	Nexplanon	2 users found this comment helpful.	"Well I have had the implant for a little over 2 months. I have been bleeding 6 out of the 9 weeks
28954	Lexapro	2 users found this comment helpful.	"Of all the SSRI's, Lexapro works best for my OCD at 30mg/day. I take Wellbutrin SR 100mg 3x
78255	Lasix	33 users found this comment helpfu…	"This has helped swelling, but becomes tiring. "
159772	Bactrim	3 users found this comment helpful.	"Given a 7 day course. After 2 days, insomnia, anxiety. Complete loss of appetite, eating was a cho
33106	Seasonique	3 users found this comment helpful.	"I am on my final week of active pills on my first package, I have had more bleeding in these three
159222	TriNessa	3 users found this comment helpful.	"When I read reviews of Trinessa after I got prescribed to it, I almost decide not even to try it.
230621	Depo-Provera	4 users found this comment helpful.	"I was on Depo for 9 months, and overall I would rate it poorly. After my first shot, things seemed

删除明显与 rating 没有关系的列 recordID,reviewComment 和 date

之后，为了使数据更好处理，对 drugName, condition, sideEffects 进行编码，

具体编码策略如下：对于每个不同的值，用数字代替

```python
dict_drug = {}
count = 0
for i in sorted(set(drug_name)):
    dict_drug[i] = count
    count+=1
```

之后为上述三个属性和 usefulCount 分别计算训练集中它们与 rating 的皮尔逊

相关系数，结果如下：

```
C:\Users\dell\anaconda3\python.exe C:/Users/dell/Desktop/机器学习/homework03/test.py
drugname:(0.009750756805287624, 0.4169258341295182)
condition:(0.0445341297135339, 0.000207841408653436)
sideeffect:(0.01313751786735927, 0.2740687498421191)
usefulcount:(0.21809318186080934, 1.9774580652887071e-75)

Process finished with exit code 0
```

发现四者中只有 condition 和 usefulCount 的 p 值小于 0.05，故舍弃其他两个

最终数据集如下:

condition	usefulCount	rating
108	22	5
101	17	4
401	3	5
417	35	5
54	4	5
217	13	2
54	1	3
246	32	5
280	21	4
54	3	1
183	17	1
407	7	3
417	57	1
351	19	5
205	44	1
257	14	5
368	26	5
3	1	2
53	24	3
401	9	5

二、 决策树算法

1、信息熵

一个节点信息熵，p_k是每个属性出现的概率。

$$\mathrm{Ent}_D = -\sum_{k=1}^{K} p_k * log_2 p_k$$

每种特征中每个属性的信息熵Ent_{D_v}

每个特征的信息熵

$$\sum_{1}^{V}\frac{D_v}{D}\mathrm{Ent}_{D_v}$$

信息增益，a 是属性

$$Gain_{(D,a)} = \mathrm{Ent}_D - \sum_{1}^{V}\frac{D_v}{D}\mathrm{Ent}_{D_v}$$

算法过程

1. 将所有的特征看成一个一个的节点。创建根节点。

2. 遍历所有特征。遍历到其中某一个特征时，遍历当前特征的所有分割方式，找到最好的分割点，将数据划分为不同的子节点，计算划分后子节点的信息熵。

3. 在遍历的所有特征中，比较寻找最优的特征以及最优特征的最优划分方式。选择信息增益最高的特征，根据特征则对当前数据集进行分割操作，产生子树。

4. 对新的子节点继续执行 2 - 3 步，直到下面的停止条件退出循环。

停止条件:

1. 当子节点中只有一种类型或为空的时候停止构建(会导致过拟合)

2. 当前节点种样本数小于某个值，同时迭代次数达到指定值，停止构建，此时使用该节点中出现最多的类别样本数据作为对应值(比较常用)

核心代码:

1、建树

```python
def createTree(dataSet, labels, labelProperty):
    class_list = [example[-1] for example in dataSet] #取出标签

    #如果集合中样本属于同一类别，则将node标记为该类叶节点
    if class_list.count(class_list[0]) == len(class_list):
        return class_list[0]

    #如果属性集为空或者样本在属性集上的取值相同，则将node标记为叶节点
    if len(dataSet[0]) == 1:
        return majorCnt(class_list)

    #选择最佳标签，返回标签索引
    best_feat, best_part_value = chooseBestFeatureToSplit_c(dataSet, labelProperty)

    #如果无法选出最优分类特征，返回次数最多的类别
    if best_feat == -1:
        return majorCnt(class_list)

    if(labelProperty[best_feat] == 0):
        best_feat_label = labels[best_feat]
        myTree = {best_feat_label: {}}
        labels_new = copy.copy(labels)
        labelPropertyNew = copy.copy(labelProperty)
        del(labels_new[best_feat])
        del(labelPropertyNew[best_feat])

```

```python
        for value in unique_value:
            subLabels = labels_new[:]
            subLabelProperty = labelPropertyNew[:]
            myTree[best_feat_label][value] = createTree(splitDataSet(dataSet, best_feat, value), subLabels, subLabelProperty)

    else: #对于连续值的处理，不删除特征，分别建立左右子树
        best_feat_label = labels[best_feat] + '<' + str(best_part_value)
        myTree = {best_feat_label:{}}
        subLabels = labels[:]
        subLabelsProperty = labelProperty[:]

        value_left = 'yes'
        myTree[best_feat_label][value_left] = createTree(splitDataSet_c(dataSet, best_feat, best_part_value, 'L'), subLabels, subLabelsProperty)

        value_right = 'no'
        myTree[best_feat_label][value_right] = createTree(splitDataSet_c(dataSet, best_feat, best_part_value, 'R'), subLabels, subLabelsProperty)
    return myTree
```

2、计算信息熵并寻找最佳信息增益

```python
def calcShannonEnt(dataSet):
    #样本数量
    sample_size = len(dataSet)
    #计算每个label出现次数的字典
    labelCount = {}
    for sample in dataSet:
        current_label = sample[-1]
        if current_label not in labelCount.keys():
            labelCount[current_label] = 0
        labelCount[current_label] += 1

    shannonEnt = 0.0
    for key in labelCount.keys():
        prob = float(labelCount[key]) / sample_size
        shannonEnt -= prob*math.log(prob, 2)

    return shannonEnt
```

```python
def chooseBestFeatureToSplit_c(dataSet, labelProperty):
    feature_number = len(labelProperty)
    base_entropy = calcShannonEnt(dataSet)
    best_info_gain = 0.0
    best_feature = -1 #初始化最佳特征的索引值
    best_part_value = None

    for i in range(feature_number):
        features = [example[i] for example in dataSet]
        unique_feature = set(features)

        entropy = 0.0
        best_part_value_i = None
        if labelProperty[i] == 0: #处理离散化
            for value in unique_feature:
                subDataSet = splitDataSet(dataSet, i, value)
                prob = len(subDataSet) / float(len(dataSet))
                entropy += prob * calcShannonEnt(subDataSet)
```

```python
        else:  #处理连续的特征
            sorted_unique_feature = list(unique_feature)
            sorted_unique_feature.sort()
            # min_entropy = float("-inf")
            min_entropy = 999999999
            for j in range(len(sorted_unique_feature) - 1):  #计算划分点
                part_value = (float(sorted_unique_feature[j]) + float(sorted_unique_feature[j + 1])) / 2
                dataSetLeft = splitDataSet_c(dataSet, i, part_value, 'L')
                dataSetRight = splitDataSet_c(dataSet, i, part_value, 'R')
                prob_left = len(dataSetLeft) / float(len(dataSet))
                prob_right = len(dataSetRight) / float(len(dataSet))
                entropy_temp = prob_left * calcShannonEnt(dataSetLeft) + prob_right * calcShannonEnt(dataSetRight)
                if entropy_temp < min_entropy:
                    min_entropy = entropy_temp
                    best_part_value_i = part_value
            entropy = min_entropy
        info_gain = base_entropy - entropy
        if info_gain > best_info_gain:
            best_info_gain = info_gain
            best_feature = i
            best_part_value = best_part_value_i
    return best_feature, best_part_value
```

三、 实验结果

```
ount<118.5': {'yes': 5, 'no': 4}}}}}}, 'no': 5}}, 'no': {'usefulCount<132.5': {'yes': {'usefulCount<126.5': {'yes': 1, 'no': 4}}, 'no': {'condition<292.0': {'yes': 5, 'n
': {'usefulCount<141.0': {'yes': 5, 'no': {'usefulCount<152.0': {'yes': 4, 'no': 5}}}}}}}}}}}, 'no': {'usefulCount<115.0': {'yes': {'condition<349.0': {'yes': 1, 'no': 5}
, 'no': 5}}}}}, 'no': {'condition<317.5': {'yes': 1, 'no': 4}}}}, 'no': {'usefulCount<119.5': {'yes': 3, 'no': {'usefulCount<134.5': {'yes': 4, 'no': 5}}}}}}}, 'no': {'con
ition<290.0': {'yes': {'condition<283.5': {'yes': {'condition<276.0': {'yes': {'usefulCount<176.5': {'yes': 5, 'no': 4}}, 'no': 5}}, 'no': 4}}, 'no': {'condition<300.0':
{'yes': 5, 'no': {'condition<311.5': {'yes': 3, 'no': 5}}}}}}}}}}}}}, 'no': {'usefulCount<113.5': {'yes': {'condition<420.0': {'yes': 5, 'no': {'usefulCount<112.5': {'yes
: 5, 'no': 3}}}}}, 'no': {'usefulCount<121.5': {'yes': {'usefulCount<119.5': {'yes': 5, 'no': {'condition<413.0': {'yes': 5, 'no': {'condition<420.0': {'yes': 4, 'no': 5}
}}}}, 'no': 5}}}}}}}}}}}}}}}}
100%|██████████████████████████████████████████████████████████████████████| 1186/1186 [32:39<00:00,  1.65s/it
Micro F1:  0.3844856661045531
Macro F1:  0.22777483939168724
```

Micro F1: 0.3844856661045531

Macro F1: 0.22777483939168724

测试集预测结果部分展示:

condition	usefulCount	rating
54	1	5
84	0	4
133	13	5
368	7	3
257	11	5
169	21	5
293	86	5
94	76	5
115	46	5
211	21	4
26	1	5
1	8	2
54	0	5
234	32	5
368	2	5
293	72	5
179	29	5
54	1	5
133	24	5
133	4	5
54	31	4
133	2	5
30	23	5
54	6	5
115	15	5
115	31	1
191	10	5
363	37	5
417	5	5
368	22	5
242	28	5
108	35	4
250	2	5
54	2	1
160	6	5
0	12	5
417	42	5