Data Mining, Spring 2018

Problem Set #2: Supervised Learning II

(Due on May 4 Friday, 2018 at 11:59pm)

Submission Instructions

These questions require thought but do not require long answers. Please be as concise as possible. You should submit your answers as a write-up in PDF format to DataMining_2018@126.com. The email title is formatted as “hwk2_学号_姓名”.

Questions

1. 模型的性能度量

我们需要比较两个分类模型M_1和M_2。他们在 10 个二类（+或-）样本所组成的测试集上的分类结果如下表格中所示。假设我们更关心正样本是否能被正确检测。

Instance	True Class	Scores from M_1	Scores from M_2
1	+	0.73	0.61
2	+	0.69	0.03
3	-	0.44	0.68
4	-	0.55	0.31
5	-	0.67	0.45
6	+	0.47	0.09
7	-	0.08	0.38
8	-	0.15	0.05
9	+	0.45	0.01
10	-	0.35	0.04

（1）对于分类模型M_1，取阈值为 0.5，分别计算分类准确率（accuracy）、查准率（precision）、查全率（recall，又称真正例率，true positive rate，TPR）、假正例率（false positive rate，FPR）和 F-measure；

答：基于分类模型 M_1，以及阈值为 0.5，可将上述样本集合统计后画成如下表格：

	Predicted Class	
Actual Class	+	-
+	2	2
-	2	4

基于分类模型 M_1，阈值 0.5

因此：

准确率$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+FN+FP+TN}} = \frac{2+4}{2+2+2+4} = 0.6$；

查准率Precision $= \frac{TP}{TP+FP} = \frac{2}{2+2} = 0.5$；

查全率Recall $= \frac{TP}{TP+FN} = \frac{2}{2+2} = 0.5$；

假正例率FPR $= \frac{FP}{FP+TN} = \frac{2}{2+4} = 0.33$；

$F - mearsure = \frac{2rp}{r+p} = \frac{2*0.5*0.5}{0.5+0.5} = 0.5.$

（2）对于分类模型M_2，取阈值为 0.5，分别计算分类准确率（accuracy）、查准率（precision）、查全率（recall，又称真正例率，true positive rate，TPR）、假正例率（false positive rate，FPR）和 F-measure；并与分类模型M_1比较，分析哪个分类模型在这个测试集上表现更好；

答：基于分类模型 M_2，以及阈值为 0.5，可将上述样本集合统计后画成如下表格：

	Predicted Class		
Actual Class		+	-
	+	**1**	**3**
	-	**1**	**5**

基于分类模型 M_2，阈值 0.5

因此：

准确率Accuracy $= \frac{TP+TN}{TP+FN+FP+TN} = \frac{1+5}{1+3+1+5} = 0.6$；

查准率Precision $= \frac{TP}{TP+FP} = \frac{1}{1+1} = 0.5$；

查全率Recall $= \frac{TP}{TP+FN} = \frac{1}{1+3} = 0.25$；

假正例率FPR $= \frac{FP}{FP+TN} = \frac{1}{1+5} = 0.17$；

$F - mearsure = \frac{2rp}{r+p} = \frac{2*0.25*0.5}{0.25+0.5} = 0.33.$

可以看出分类模型 M_1 的 F-measure 值比 M_2 的大，所以分类模型 M_1 在这个测试集上表现更好。

（3）对于分类模型M_1，取阈值为 0.2，分别计算分类准确率（accuracy）、查准率（precision）、查全率（recall，又称真正例率，true positive rate，TPR）、假正例率（false positive rate，FPR）和 F-measure；并讨论当阈值为 0.2 或 0.5 时，哪个分类模型M_1的分类结果哪个更好；

答：基于分类模型 M_1，以及阈值为 0.2，可将上述样本集合统计后画成如下表格：

	Predicted Class		
Actual Class		+	-
	+	**4**	**0**
	-	**4**	**2**

基于分类模型 M_1，阈值 0.2

因此：

准确率Accuracy $= \frac{TP+TN}{TP+FN+FP+TN} = \frac{4+2}{4+0+4+2} = 0.6$；

查准率Precision $= \frac{TP}{TP+FP} = \frac{4}{4+4} = 0.5$；

查全率$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} = \frac{4}{4+0} = 1$；

假正例率$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP+TN}} = \frac{2}{2+4} = 0.33$；

$\mathrm{F-mearsure} = \frac{2rp}{r+p} = \frac{2*0.5*1}{0.5+1} = 0.67.$

可以看出分类模型 M_1 在阈值为 0.2 时，F-measure 值比阈值为 0.5 的大，且正样本更能被准确分类，所以在这个测试集上阈值为 0.2 时表现更好。

（4）试讨论是否存在更好的阈值；若存在，请求出最优阈值并说明原因。

答： 编写程序求解取不同阈值时，比较对应的 F-measure 值。当阈值为 0.44 时，假设把样本集中负样本 0.44 归为 TN，即是正确分布的，分类模型 M_1 的表现更好，此时有：

	Predicted Class		
Actual Class		+	-
	+	**4**	**0**
	-	**2**	**4**

基于分类模型 M_1，阈值 0.44

因此：

准确率$\mathrm{Accuracy} = \frac{\mathrm{TP+TN}}{\mathrm{TP+FN+FP+TN}} = \frac{4+2}{4+0+2+4} = 0.6$；

查准率$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP+FP}} = \frac{4}{4+2} = 0.67$；

查全率$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} = \frac{4}{4+0} = 1$；

假正例率$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP+TN}} = \frac{2}{2+4} = 0.33$；

$\mathrm{F-mearsure} = \frac{2rp}{r+p} = \frac{2*1*2/3}{1+2/3} = 0.8.$

可以看出分类模型 M_1 在阈值为 0.44 时，F-measure 值比阈值为 0.44 的大，且正样本全被准确分类，假正例率降为 1/3，所以最优阈值为 0.44。

2. 神经网络

考虑以下的二类训练样本集

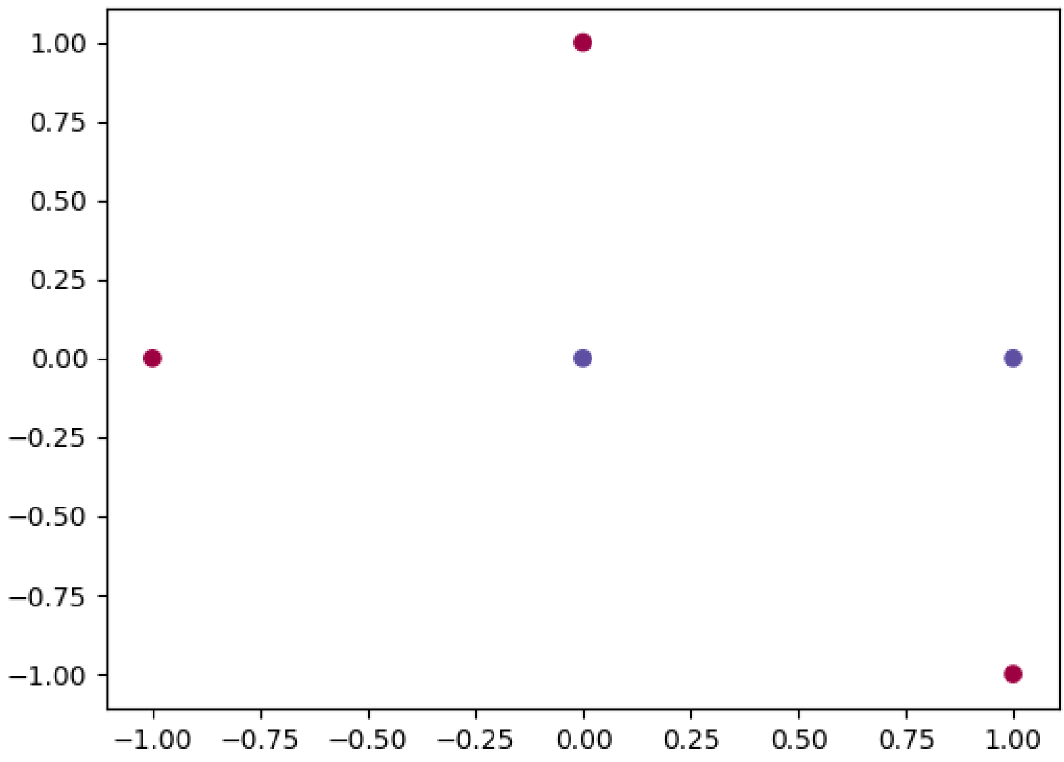
Instance	Feature vector $\boldsymbol{x}$	Output label y
1	(0, 0)	+
2	(1, 0)	+
3	(0, 1)	-
4	(-1, 0)	-
5	(1, -1)	-

对此训练样本集，我们需要训练一个三层神经网络（输入层、单隐层、输出层），其中单隐层的

单元（神经元）数目设为 2，激活函数（activation function）为 Sigmoid 函数：

（1）在二维坐标系中画出这 5 个训练样本点，并讨论此训练样本集是否线性可分；

答：如下图画出这 5 个训练样本点：从图中可观察出该训练样本集非线性可分，无法找到一条直线可以完全准确地把这 5 个训练样本点分类。



（2）试分析将 Sigmoid 激活函数换成线性函数的缺陷；

答：如果激活函数换成线性函数，那么无论神经网络有多少层，输出都是输入的线性组合，与没有隐藏层的效果相当，就成了最原始的感知器了，与不使用激活函数、直接使用逻辑回归没有区别。

（3）令初始化参数全部为 0，试运用前馈（feedforward）算法计算在初始化参数下此三层神经网络的输出；然后运用反向传播（backpropagation）算法，计算代价函数对所有参数的偏导数，并讨论将初始化参数全部设为 0 所带来的问题；

答：当初始化参数全部为 0 时，激活函数为 sigmoid 函数，**运用前馈算法计算得到在隐藏层的输出都为 0.5，在输出层的输出也都为 0.5.具体计算过程如下表：以第一个样本点为例：**

$a_0^{(1)}=x_0=1$		$a_0^{(2)}=1$		
$a_1^{(1)}=x_1=0$	$z_1^{(2)}=0$	$a_1^{(2)}=0.5$	$z_1^{(3)}=0$	$a^{(3)}=0.5$
$a_2^{(1)}=x_2=0$	$z_2^{(2)}=0$	$a_2^{(2)}=0.5$	$z_2^{(3)}=0$	

将 $a_1^{(1)}=x_1$，$a_2^{(1)}=x_2$ 代为其他不同的样本点，可以得出同样的结果 0.5.

然后运用反向传播算法，可以得出代价函数对所有参数的偏导数为：

对$\theta^1 \in R^{(2*3)}$的偏导数如下：

$$\frac{\partial J(\theta)}{\partial \theta^1} = [[0,0,0],[0,0,0]]$$

对$\theta^2 \in R^{(1*3)}$的偏导数如下：

$$\frac{\partial \mathrm{J}(\theta)}{\partial \theta^2} = [0.1, 0.05, 0.05]$$

具体计算过程如下：以第一个样本点为例：

初始化$\Delta^1 = 0, \Delta^2 = 0$;

从前馈算法可得：

$a^{(1)}$=[1, 0, 0];

$z^{(2)}$=[0, 0];

$a^{(2)}$=[1, 0.5, 0.5];

$a^{(3)}$=[0.5];

因此：

$$\delta^{(3)} = a^{(3)} - y = 0.5 - 1 = -0.5$$

$$\delta^{(2)} = \theta^{(2)}.T * \delta^{(3)}.* g'\left(z^{(2)}\right) = [0, 0, 0]$$

$$\Delta^{(1)} = \Delta^{(1)} + a^{(1)} * \delta^{(2)} = [[0, 0, 0], [0, 0, 0]]$$

$$\Delta^{(2)} = \Delta^{(2)} + a^{(2)} * \delta^{(3)} = [-0.5, -0.25, -0.25]$$

接下来代入其他样本点，不断重复该过程，最后可得：

$$\Delta^{(1)} = [[0, 0, 0], [0, 0, 0]]$$

$$\Delta^{(2)} = [0.5, 0.25, 0.25]$$

除以样本数 m=5，**所以偏导数为**：

$$\frac{\partial \mathrm{J}(\theta)}{\partial \theta^1} = D^{(1)} = \frac{1}{m}\Delta^{(1)} = [[0, 0, 0], [0, 0, 0]]$$

$$\frac{\partial \mathrm{J}(\theta)}{\partial \theta^2} = D^{(2)} = \Delta^{(2)} = \frac{1}{m}\Delta^{(2)} = [0.1, 0.05, 0.05]$$

初始化参数不能全部设为 0 的原因：如果我们令所有的初始参数都为 0，这将意味着我们第二层的所有激活单元都会有相同的值，那么隐藏神经元对输出单元的影响也是相同的，通过反向传播梯度下降法进行计算时，会得到同样的梯度大小，所以无论设置多少个隐藏单元，其最终的影响都是相同的。同理，也不能初始化所有的参数都为同一个非 0 的数。因此，要随机化初始参数，以打破对称性。

（4）试给出一个神经网络（画出架构图，并写出激活函数及其对应的参数），使此训练样本集的 5 个训练样本点都可以被正确分类。

答：采用如下的神经网络，输入层到隐藏层采用 tanh 激活函数，隐藏层到输出层采用 sigmoid 激活函数。

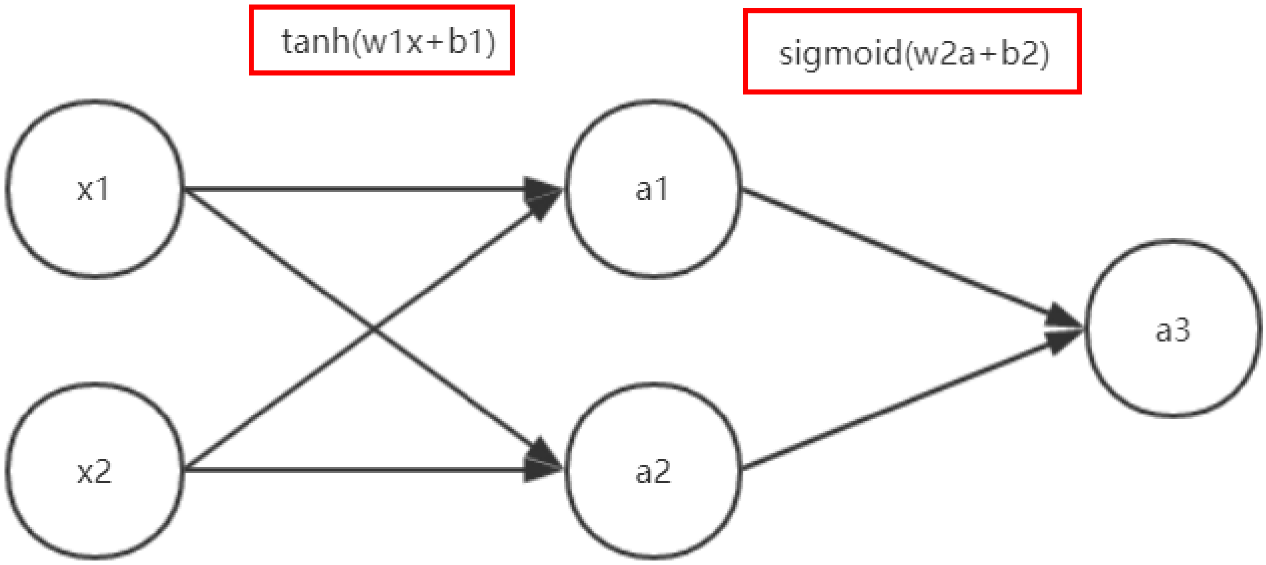
训练后如下参数可以使样本集都可以被正确分类：

w1=[[-1.05787697, -4.99155074], [-3.17844781, 4.27784881]],;

b1=[[-1.65421961], [-1.88599495]];

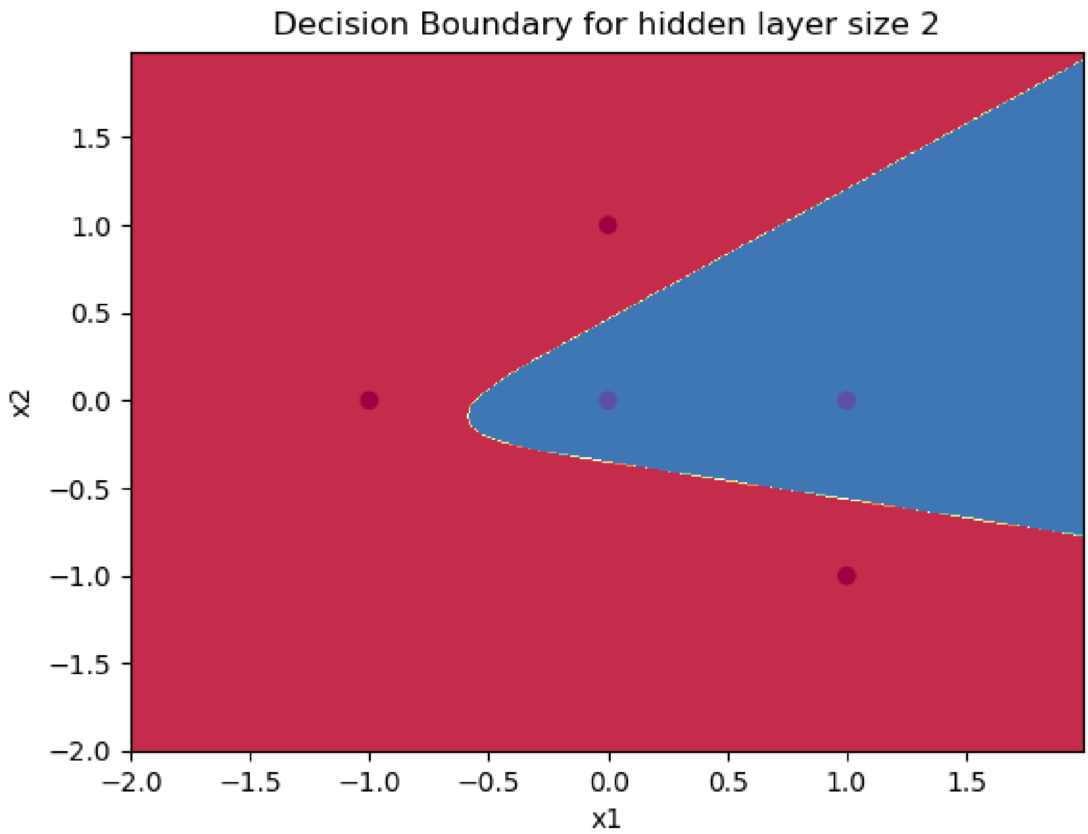
w2=[[-7.78571771, -7.80167614]];

b2=[[-7.26055828]].



程序结果运行如下：

```
{'W1': array([[-1.05787697, -4.99155074],
       [-3.17844781,  4.27784881]]), 'b1': array([[-1.65421961],
       [-1.88599495]]), 'W2': array([[-7.78571771, -7.80167614]]), 'b2': array([[-7.26055828]])}
Accuracy: 100%
```



3. 决策树

考虑以下的二类训练样本集

Instance	A	B	Class Label
1	T	F	+
2	T	T	+
3	T	T	+
4	T	F	-
5	T	T	+
6	F	F	-
7	F	F	-
8	F	F	-
9	T	T	-
10	T	F	-

（1）计算以属性 A 或 B 为划分的信息熵（Entropy）增益，并说明决策树学习算法选择哪个属性进行划分；

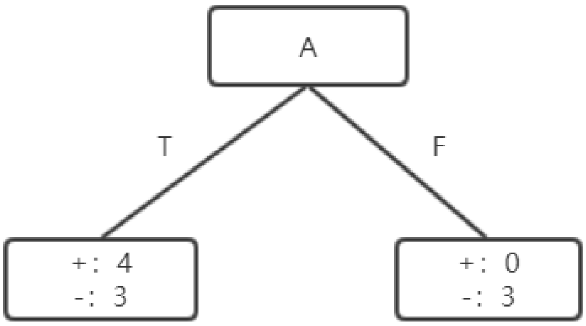
答：

对 A 来说：信息熵$\text{Entropy}(A) = -\frac{4}{10}\log_2\frac{4}{10} - \frac{6}{10}\log_2\frac{6}{10} = 0.9709505944546686$，

条件熵 $\text{Entropy}(T) = -\frac{4}{7}\log_2\frac{4}{7} - \frac{3}{7}\log_2\frac{3}{7} = 0.9852281360342516$， $\text{Entropy}(F) = -\frac{0}{3}\log_2\frac{0}{3} - \frac{3}{3}\log_2\frac{3}{3} = 0.$

因此，信息熵增益 Gain= $\text{Entropy}(A) - \frac{7}{10}\text{Entropy}(T) - \frac{3}{10}\text{Entropy}(F) = 0.2812908992306924$

如图：

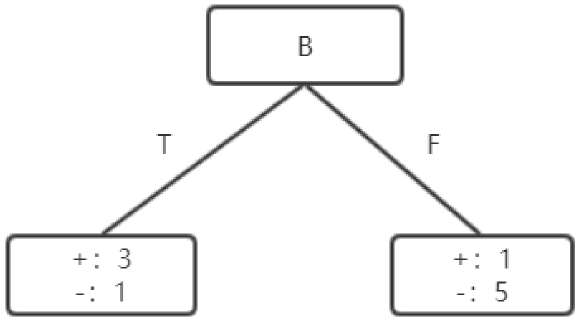


对 B 来说：信息熵$\text{Entropy}(B) = -\frac{4}{10}\log_2\frac{4}{10} - \frac{6}{10}\log_2\frac{6}{10} = 0.9709505944546686$，

条件熵 $\text{Entropy}(T) = -\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4} = 0.8112781244591328$， $\text{Entropy}(F) = -\frac{1}{6}\log_2\frac{1}{6} - \frac{5}{6}\log_2\frac{5}{6} = 0.6500224216483541.$

因此，信息熵增益 Gain= $\text{Entropy}(B) - \frac{4}{10}\text{Entropy}(T) - \frac{6}{10}\text{Entropy}(F) = 0.256425891682003$

如图：



因为属性 A 的信息熵增益比属性 B 的信息熵增益大，所以应该选择属性 A 进行划分。

（2）计算以属性 A 或 B 为划分的 Gini 增益，并说明决策树学习算法选择哪个属性进行划分；

答：

对 A 来说：

$$\text{Gini}(\text{A}) = 1 - \frac{4}{10} * \frac{4}{10} - \frac{6}{10} * \frac{6}{10} = 0.48，$$

$$\text{Gini}(\text{T}) = 1 - \frac{4}{7} * \frac{4}{7} - \frac{3}{7} * \frac{3}{7} = 0.48979591836734704，$$

$$\text{Gini}(\text{F}) = 1 - \frac{0}{3} * \frac{0}{3} - \frac{3}{3} * \frac{3}{3} = 0$$

因此，Gini 增益为 Gain= $\text{Gini}(\text{A}) - \frac{7}{10} Gini(\text{T}) - \frac{3}{10} Gini(\text{F}) = 0.13714285714285707$。

对 B 来说：

$$\text{Gini}(\text{B}) = 1 - \frac{4}{10} * \frac{4}{10} - \frac{6}{10} * \frac{6}{10} = 0.48，$$

$$\text{Gini}(\text{T}) = 1 - \frac{3}{4} * \frac{3}{4} - \frac{1}{4} * \frac{1}{4} = 0.375，$$

$$\text{Gini}(\text{F}) = 1 - \frac{1}{6} * \frac{1}{6} - \frac{5}{6} * \frac{5}{6} = 0.2777777777777778$$

因此，Gini 增益为 Gain= $\text{Gini}(\text{B}) - \frac{4}{10} Gini(\text{T}) - \frac{6}{10} Gini(\text{F}) = 0.16333333333333333$。

因为属性 B 的 Gini 增益比属性 A 的 Gini 增益大，所以应该选择属性 B 进行划分。

（3）计算以属性 A 或 B 为划分的分类误差（Classification Error）增益，并说明决策树学习算法选择哪个属性进行划分；

答：

对 A 来说：

$$\text{Error}(\text{A}) = 1 - \frac{6}{10} = 0.4，$$

$$\text{Error}(\text{T}) = 1 - \frac{4}{7} = \frac{3}{7}，$$

$$\text{Error}(\text{F}) = 1 - \frac{3}{3} = 0。$$

因此，分类误差增益为 Gain= $\text{Error}(\text{A}) - \frac{7}{10} Error(\text{T}) - \frac{3}{10} Error(\text{F}) = 0.1$。

对 B 来说：

$\mathrm{Error(B)} = 1 - \frac{6}{10} = 0.4$，

$\mathrm{Error(T)} = 1 - \frac{3}{4} = \frac{1}{4}$，

$\mathrm{Error(F)} = 1 - \frac{5}{6} = \frac{1}{6}$。

因此，分类误差增益为 Gain= $\mathrm{Error(B)} - \frac{4}{10}Error(\mathrm{T}) - \frac{6}{10}Error(\mathrm{F}) = 0.2$。

因为属性 B 的分类误差增益比属性 A 的分类误差增益大，所以应该选择属性 B 进行划分。

（4）说明信息熵增益、Gini 增益和分类误差增益对属性选择有不一样的偏好。

答：

信息熵增益：当子结点的加权平均熵越小，表示再往下分支越容易，或者说当前特征提供的信息量越多。信息熵针对分类中的属性。然而，在各个特征的可能取值不同时，比如有些特征只有 0/1 取值，而有些特征可以有几十种取值，信息熵容易选择一个取值很多的特征，导致过拟合。除此之外，在多分类问题中，信息熵增益存在大量的 log 计算，因此计算复杂度倍增。在二分类问题中表现突出。

Gini 增益：如果这个结点是个叶子结点，从中随机取一个数据，并按该结点中各类数据的分布随机地预测一个类别，预测错误的概率。Gini 系数是针对较为连续的属性，最小化错分率。Gini 不像信息熵计算复杂，因此效率方面很高。

分类误差增益：和 Gini 增益大同小异。

对于二分类问题：