模式识别第一次作业

201300066 麻超 人工智能学院

Q1

a

为了让这个公式有意义,必须要满足$\sqrt{\frac{8a-1}{3}} \geq 0$,即$a \geq \frac{1}{8}$.

b

当$a = \frac{1}{8}$时,求解原公式可得$f(a) = 1$.

c

提出想法:与三次方程的求根公式有关(当然这里是我看到了最后一小问的内容),该公式的值恒为1,代表了三次方程恒有一个为1的根.
取一些其他的特殊样例:
$a = 1/2$,得到$f(a) = 1$.
$a = 1/4$,得到$f(a) = 1$.

d

编写matlab程序:

```
syms a;
f=(a+(a+1)/3*sqrt((8*a-1)/3))^(1/3)+(a-(a+1)/3*sqrt((8*a-1)/3))^(1/3);
g=matlabFunction(f)
b=g(3/4)
```

结果:$b = 1.2182 + 0.1260i$

e

在matlab 的官方文档内可以知道,对于负底数 A 和非整数 B，power 函数返回复数结果。同时可以使用 nthroot 函数可获取实数根。因此考虑问题出现在两个^(1/3)幂次的问题上.进行改良:

```
syms a;
f1=a+(a+1)/3*sqrt((8*a-1)/3);
f2=a-(a+1)/3*sqrt((8*a-1)/3);
g1=matlabFunction(f1);g2=matlabFunction(f2);
b1=g1(3/4);b2=g2(3/4);
s1=nthroot(b1,3);s2=nthroot(b2,3);
s=s1+s2
```

得到结果为$s = 1$.

f

现在需要证明$f(a)=\sqrt[3]{a+\frac{a+1}{3}\sqrt{\frac{8a-1}{3}}}+\sqrt[3]{a-\frac{a+1}{3}\sqrt{\frac{8a-1}{3}}}$恒为1.

令$x=\sqrt[3]{a+\frac{a+1}{3}\sqrt{\frac{8a-1}{3}}}, y=\sqrt[3]{a-\frac{a+1}{3}\sqrt{\frac{8a-1}{3}}}, f(a)=x+y.$

$$
\begin{aligned}
(x+y)^3 &= a+\frac{a+1}{3}\sqrt{\frac{8a-1}{3}}+a-\frac{a+1}{3}\sqrt{\frac{8a-1}{3}}+3xy(x+y)\\
(x+y)^3 &= 2a+3xy(x+y)\\
(x+y)^2 &= 2a/(x+y)+3xy \qquad \text{since x,y} \neq 0\\
(x+y)^2 &= 2a/(x+y)+\sqrt[3]{-8a^3+12a^2-6a+1} \qquad \text{Expand}\\
(x+y)^2 &= 2a/(x+y)-2a+1\\
(x+y-1)((x+y)^2&+(x+y)+2a)=0\\
&\because (x+y)\geq 0, a\geq \frac{1}{8},\\
&\therefore x+y-1=0\\
&\therefore x+y=1.\\
&\therefore f(a)=1.
\end{aligned}
$$

g

我们可以令$a=2$,此时同样代入可以得到$\frac{a+1}{3}\sqrt{\frac{8a-1}{3}}=\sqrt{5}$.符合该公式,所以有
$\sqrt[3]{2+\sqrt{5}}+\sqrt[3]{2-\sqrt{5}}=1.$

h

了解到一元三次方程的解中,第一个解的求解公式是:$t=\sqrt[3]{-\frac{q}{2}+\sqrt{\frac{q^2}{4}+\frac{p^3}{27}}}+\sqrt[3]{-\frac{q}{2}-\sqrt{\frac{q^2}{4}+\frac{p^3}{27}}}$
与原$f(a)$对比,可以发现:$f(a)$正是一个特例,当$p^3=-1-3q-3q^2-q^3=-(q+1)^3$时,也即
$p=-(q+1)$时,原一元三次方程x^3+px+q必然有一个根为1.

Q2

1

Prove:由于$X\sim\mathcal{N}(0,1)$,所以其概率密度函数$f(x)=\frac{1}{\sqrt{2\pi}}e^{-x^2/2}$.于是有:

$$
\begin{aligned}
P(X \geq \epsilon) &= \int_{\epsilon}^{\infty} f(x)dx \\
&= \int_{\epsilon}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2}dx \\
&= \int_{0}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-(x+\epsilon)^2/2}dx \\
&\leq \int_{0}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2}e^{-\epsilon^2/2}dx \qquad \text{since } e^{-(x+\epsilon)^2/2} \leq e^{-\epsilon^2/2}e^{x^2/2} \\
&= e^{-\epsilon^2/2}\int_{0}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2}dx \\
&= e^{-\epsilon^2/2}P(X > 0) \\
&= \frac{1}{2}e^{-\epsilon^2/2}
\end{aligned}
$$

2

Prove:
该证明可以用Markov's 不等式证明,过程如下:

$$
\begin{aligned}
&\int_{0}^{\infty} xf(x)dx \geq \epsilon\int_{\epsilon}^{\infty} f(x)dx = \epsilon P(x > \epsilon) \\
&\because X \sim \mathcal{N}(0,1) \\
&\therefore \frac{1}{\sqrt{2\pi}}\int_{\epsilon}^{\infty} xe^{-x^2/2}dx \geq \frac{\epsilon}{\sqrt{2\pi}}\int_{\epsilon}^{\infty} e^{-x^2/2}dx = \epsilon P(X > \epsilon) = \frac{\epsilon}{2}P(|X| > \epsilon) \\
&\therefore \frac{1}{\sqrt{2\pi}}\int_{\epsilon}^{\infty} xe^{-x^2/2}dx = \frac{1}{\sqrt{2\pi}}[-e^{-x^2/2}]_{\epsilon}^{\infty} = \frac{1}{\sqrt{2\pi}}e^{-\epsilon^2/2} \\
&\therefore \frac{t}{2}P(|X| > \epsilon) \leq \frac{1}{\sqrt{2\pi}}e^{-\epsilon^2/2} \\
&\therefore P(|X| \geq \epsilon) \leq \sqrt{\frac{2}{\pi}}\frac{e^{-\epsilon^2/2}}{\epsilon} \\
&\therefore P(|X| \geq \epsilon) \leq \min\{1, \sqrt{\frac{2}{\pi}}\frac{e^{-\epsilon^2/2}}{\epsilon}\}
\end{aligned}
$$

Q3

1

由于共轭函数的定义为$f^*(y) = \sup\limits_{x \in dom(f)} (y^T x - f(x))$
所以其等价于 $\inf\limits_{x \in dom(f)} (f(x) - x^{'} y) = -f^*(y)$
当$y = 0$时,得到:$\inf_x f(x) = -f^*(0)$.

2

根据共轭函数的定义,一定有$f^*(y) \geq (y^T x - f(x))$
所以:$f^*(y) + f(x) \geq y^T x$.
标量的转置等于自身,所以有$f^*(y) + f(x) \geq x^T y$

3

根据第二问的结论:$f(x) \geq y^T x - f^*(y)$
同理,有:$f(z) \geq y^T z - f^*(y)$
于是可以得到$f^{**}(z) = \sup\limits_{y \in dom\ f^*} (z^T y - f^*(y))$
故$f^{**}(z) \leq f(z)$.

Q4

a

方法一:调整图片分辨率大小,类似于马赛克的计数原理,将原来400×400的矩阵中每一个2×2的小矩阵用它们的平均值代替,这样可以将原来的400×400矩阵压缩至100×100,实现了预处理,同时也可以利用warppafine等图像处理工具,给原矩阵加以 $\mathrm{scale}(0.25, 0.25)$ 的变换,实现处理.当然这种方法比较粗糙,实际上也可以用每个2×2小矩阵中出现次数最多的数字来代替,或者在摒弃与其他数值相差过大的数字后再计算均值.
方法二:对原400×400和100×100的矩阵同时进行特征提取,例如PCA,SVD等方法,比较两个矩阵的提取后的特征值即可.

b

我们可以用每个2×2小矩阵的平均值来代替原来的矩阵,如此其存储开销会由$10000n$降低至$2500n$,降低了4倍.

c

$acc_{train} = 99\%, acc_{test} = 50\%.$

d

micro方法通过计算所有类别的真正例、假正例和假负例的总数来计算总体指标,然后根据这些分别的指标来计算总指标.micro方法对数据集中的每个样本给予同等的权重,而不管它属于哪个类别.它适用于存在类别不平衡的数据集.

另一方面,macro方法通过首先计算每个单独类别的指标,然后取所有类别的平均值来计算总体指标.macro方法对每个类别的处理是平等的,不管每个类别中的样本数量.它适用于每个类别都同样重要的数据集.

总之,micro方法对每个样本给予同等权重,而macro方法对每个类别给予同等权重。选择使用哪种方法取决于具体问题和数据集的特点。

在(c)中,我们分别计算了训练集中的准确率和测试集中的准确率,所以用到的是micro方法.

e

对于长尾识别问题,由于其数据集的样本数量不平衡,使用micro方法容易被短尾样本(即数量较多的样本)所主导,而无法反映长尾样本的情况。而macro方法会平等地对待每个类别，计算每个类别的准确率并取平均值，因此更适用于类别不平衡的情况。

在这种情况下,我们可以采取重采样（re-sampling）等方法.

重采样就是在已有数据不均衡的情况下,通过类别均衡等策略,人为的让模型学习时接触到的训练样本是类别均衡的,从而一定程度上减少对头部数据的过拟合.类别均衡的概念主要是区别于传统学习过程中的样本均衡（instance-balanced sampling）,也就是每个图片都有相同的概率被选中,不论其类别.而类别均衡采样的核心就是根据不同类别的样本数量,对每个图片的采样频率进行加权.当然尾部类别的图片可能会被反复重复采样,所以一般也会做一些简单的数据增强,例如反转,随机剪裁等.

除此之外,针对长尾识别问题还有代价敏感学习(包含重加权等),及迁移学习,数据增强,表征学习,分类器设计,集成学习等方法.

Q5

a

为了对z_1和z_2进行分类,需要计算z_1和z_2对$x_1 ... x_8$分别的欧氏距离,得到如下结果:

$$d(z_1,x_1)=\sqrt{(0-0)^2+(0+2)^2}=2$$

$$d(z_1,x_2)=\sqrt{(0-0)^2+(1+2)^2}=3$$

$$d(z_1,x_3)=\sqrt{(0-0)^2+(-1+2)^2}=1$$

$$d(z_1,x_4)=\sqrt{(-1-0)^2+(0+2)^2}=\sqrt{5}$$

$$d(z_1,x_5)=\sqrt{(1-0)^2+(0+2)^2}=\sqrt{5}$$

$$d(z_1,x_6)=\sqrt{(8-0)^2+(0+2)^2}=\sqrt{68}$$

$$d(z_1,x_7)=\sqrt{(8-0)^2+(1+2)^2}=\sqrt{73}$$

$$d(z_1,x_8)=\sqrt{(9-0)^2+(0+2)^2}=\sqrt{85}$$

由于x_3的距离与样本z_1最近,所以z_1被分类为A类.
对z_2计算欧氏距离:

$$d(z_2,x_1)=\sqrt{(0-8)^2+(0-2)^2}=\sqrt{68}$$

$$d(z_2,x_2)=\sqrt{(0-8)^2+(1-2)^2}=\sqrt{65}$$

$$d(z_2,x_3)=\sqrt{(0-8)^2+(-1-2)^2}=\sqrt{73}$$

$$d(z_2,x_4)=\sqrt{(-1-8)^2+(0-2)^2}=\sqrt{85}$$

$$d(z_2,x_5)=\sqrt{(1-8)^2+(0-2)^2}=\sqrt{53}$$

$$d(z_2,x_6)=\sqrt{(8-8)^2+(0-2)^2}=2$$

$$d(z_2,x_7)=\sqrt{(8-8)^2+(1-2)^2}=1$$

$$d(z_2,x_8)=\sqrt{(9-8)^2+(0-2)^2}=\sqrt{5}$$

可以发现样本x_7与z_2距离最近,所以z_2被分类为A类.

b

当k=3时,z_1与$x_1,x_3,x_4(x_5)$最近,其都为A类,所以z_1被分类为A类.z_2与x_6,x_7,x_8最近,其中x_6,x_8为B类,x_7为A类,所以z_2被分为B类.

c

原因:当采用最近邻算法时,由于只考虑与它距离最近的样本,所以x_7被考虑,但是当$K=3$时,z_2样本最近的3个有2个被分为B类,所以其根据算法原则被分为B类.

d

我并不理解这个题的问题在哪里,x_7在分类时已经被分为了A类,是要考虑分类错误的情况吗?如果是的话,那确实.与x_7最近的几个值都属于B类,而同属A类的其他值却与x_7距离相差甚远,有可能出现分类错误的情况.所以在这种情况下,当采取最近邻算法时,很可能出现被少有的误差误导的情况.而KNN算法可以在一定程度上弥补这种误差.

Q6

本次实验完成了一个KNN分类器模型的机器学习系统，主要包括了分类器模型的编写以及评估的过程，在这个过程中也加深了对python的掌握，最后针对该系统，使用验证集得到的最好的超参数`k=7`，此时最好的测试性能约在96.7%左右。基于5折交叉验证法确定的验证集上最好的超参数`k=5`,此时最好的测试准确率约为96.6%.最后，针对不均衡数据集也完成了对`precision`,`recall`,`f1_score`的计算，利用交叉验证法得到的最好的超参数`k=5`进行计算，计算结果为(0.911452184179457, 0.965, 0.9374620522161505).

In [6]:
```
######### 这里需要实现计算准确率的函数, 注意我们期望的输出是百分制, 如准确率是0.95, 我们期望的输出是95
def cal_accuracy(y_pred, y_gt):
    correct = np.sum(y_pred == y_gt)
    total = y_gt.shape[0]
    return int(correct/total*100)
```

在前面的实验中，得到的在不同的超参数k下进行验证的准确率为{96%,96%,96%,96%,96%,95%}，无法区分，是由于自己在计算准确率时转为了int型，这里int值无法做出更进一步的比较，所以需要转为浮点数，改正。