From 电信学院数理提升班2021班杨筠松

试题 1

1. 机器学习，也被称为统计机器学习，是人工智能领域的一个分支，其基本思想是基于数据构建统计模型，并利用模型对数据进行分析和预测的一门学科。机器学习是一种让计算机利用数据而不是指令来进行各种工作的方法。

监督学习 (Supervised Learning)

监督学习使用带有标签的训练数据集进行训练，输入的训练数据由物体的特征向量（输入）和物体的标签（输出）两部分构成，其中，若输出的标签是一个连续的值，则称为回归监督学习；若输出标签是一个离散的值，则称为分类监督学习。

通俗来讲，训练就是先考虑二元分类问题。你告诉小孩这个动物是狗，那个也是狗。但突然一只猫跑过来，你告诉他，这个不是狗。久而久之，小孩就会产生认知模式。这个学习过程，就叫做训练。

训练集（train set）——用于模型拟合的数据样本。在训练过程中对训练误差进行梯度下降，进行学习，可训练的权重参数。

验证集（validation set）—— 是模型训练过程中单独留出的样本集，它可以用于调整模型的超参数和用于对模型的能力进行初步评估。

测试集 —— 用来评估模最终模型的泛化能力。但不能作为调参、选择特征等算法相关的选择的依据。

3.

```cpp
#include<bits/stdc++.h>
using namespace std;
const int N=1e3+7;
#define sqr(x) x*x
int n;//请输入n组数据
int a[N],b[N];
double k,m;
void solve(){
    double avea=0,aveb=0;int suma=0,sumb=0;
    for (int i=1;i<=n;i++) {suma+=a[i];sumb+=b[i];}
    avea=1.0*suma/(n);aveb=1.0*sumb/n;
    int sum1=0,sum2=0;for (int i=1;i<=n;i++) sum1+=a[i]*b[i];
    for (int i=1;i<=n;i++) sum2+=sqr(a[i]);
    k=(1.0*(sum1-n*avea*aveb))/(sum2-n*sqr(avea));
    m=aveb*1.0-k*avea;
}
int main(){
    scanf("%d",&n);
    for (int i=1;i<=n;i++) scanf("%d%d",&a[i],&b[i]);
    solve();
    printf("y=%lf*x+%lf",k,m);
    return 0;
}
```

4.梯度下降法是一种求取函数的局部极小值的迭代算法， 每一步沿当前点的负梯度的方向按一定比例下降。我们试图最小化 J(w)，采用梯度下降法：

$$\frac{\partial J(w)}{\partial \omega_j} = \frac{1}{2}\sum_{i=1}^{m}\frac{\partial\left(y^{(i)} - \sum_{j=0}^{n}\omega_j x_j^{(i)}\right)^2}{\partial \omega_j}$$

$$= -\sum_{i=1}^{m}\left(y^{(i)} - \sum_{j=0}^{n}\omega_j x_j^{(i)}\right)x_j^{(i)} \qquad \omega := \omega - \alpha\nabla_w J$$

数学知识不足，看不懂式子具体在向哪个方向处理。

牛顿法就是求解函数零值位置的一种优化算法。

对于定义在实数域的函数 f(x)，求导并且选取初始点 x0，求出该点的导数，新的 x 有：

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

至于求解函数极值的过程中，我们经常转化为求解其导数为零的问题。

神经网络则在感知机的模型上做了扩展，总结下主要有三点：

1)加入了隐藏层，隐藏层可以有多层，增强模型的表达能力，如下图实例，当然增加了这么多隐藏层模型的复杂度也增加了好多。

2）输出层的神经元也可以不止一个输出，可以有多个输出，这样模型可以灵活的应用于分类回归，以及其他的机器学习领域比如降维和聚类等。

3） 对激活函数做扩展，感知机的激活函数是 sign(z),虽然简单但是处理能力有限，因此神经网络中一般使用的其他的激活函数，比如我们在逻辑回归里面使用过的 Sigmoid 函数，即：

$$f(z) = \frac{1}{1+e^{-z}}$$

DNN 前向传播算法原理

假设选择的激活函数是 σ (z)，隐藏层和输出层的输出值为 a，则对于 DNN,利用和感知机一样的思路，便利用上一层的输出计算下一层的输出,具体数学原理需要用到矩阵，通过矩阵乘法转置等线代知识，得到最终结果（线代没学过，真的不会呀）

DNN 反向传播算法

目的：找到合适的所有隐藏层和输出层对应的线性系数矩阵 W，偏倚向量 b,让所有驯良样本输入能够得到尽可能误差小的的输出

实现：需要用到大量高等数学和线代知识，看不懂呀。。。。

5. CNN 的卷积操作代替的是人移动眼球扫描外界视觉信号的过程。在当前集中在静态图片处理的情况下，这种把扫视放入到网络里面来进行比较有优势；随着计算性能的提高，自动驾驶对于处理视频数据的要求,我们可能需要更多的借鉴人眼在处理视频数据中的特点，到时候 CNN 也许会被抛弃或改进。但是 CNN 中使用的各种技巧还是可以借鉴的，可能会抛弃卷积核，以注视中心为基准，采用"池化+全连接"联合注意力机制，辅以足够的基于数据扩增（data augmentation，包括平移、镜像、旋转和缩放）的监督训练，我个人对此很有信心。目前对视频处理的研究主要采用基于 3D 的 CNN，卷积操作仍然是其核心操作，这种方法很可能只是过渡性的方法。

试题二

1. 目标检测（Object Detection）的任务是找出图像中所有感兴趣的目标（物体），确定它们的类别和位置，是计算机视觉领域的核心问题之一。由于各类物体有不同的外观，形状，姿态，加上成像时光照，遮挡等因素的干扰，目标检测一直是计算机视觉领域最具有挑战性的问题。

目标检测的位置信息一般由两种格式（以图片左上角为原点(0,0)）：

极坐标表示：(xmin, ymin, xmax, ymax)

xmin,ymin:x,y 坐标的最小值

xmin,ymin:x,y 坐标的最大值

中心点坐标：(x_center, y_center, w, h)

x_center, y_center:目标检测框的中心点坐标

w,h:目标检测框的宽、高

2.TP（True Positives）意思就是被分为了正样本，而且分对了。

TN（True Negatives）意思就是被分为了负样本，而且分对了，

FP（False Positives）意思就是被分为了正样本，但是分错了（事实上这个样本是负样本）。

FN（False Negatives）意思就是被分为了负样本，但是分错了（事实上这个样本是正样本）。

Intersection over Union，是一种测量在特定数据集中检测相应物体准确度的一个标准，这个标准用于测量真实和预测之间的相关度，相关度越高，该值越高。

IoU 相当于两个区域重叠的部分除以两个区域的集合部分得出的结果。

一般来说，这个 score ＞ 0.5 就可以被认为一个不错的结果了。

在训练的时候，有正样本和负样本之分，正样本是与 gt 的 iou 值高于 iou 阈值(一般为 0.5)的图像区域/框；负样本是与 gt 的 iou 值低于 iou 阈值的图像区域/框。

在测试时，所有预测出来的框都是 P（机器认为的正样本），与 gt 的 iou 高于 iou 阈

值的为 TP，反之为 FP。没有预测出来的框都是 N，目标检测一般不区分 TN 和 FN。因为负样本根本没有显示出来，也不存在区分真假的问题。因此，目标检测中，TN 和 FN 无意义。

3. **精确率、精度（Precision）**

$$P = \frac{TP}{TP + FP}$$

召回率（recall）

$$recall = \frac{TP}{TP + FN}$$

PR 曲线：

P-R 曲线用来衡量分类器性能的优劣，横轴为 recall ，纵轴为 precision，P-R 曲线是如何衡量分类器性能的呢？

如图，有三个分类器,A,B，C，若一个学习器的 P-R 曲线完全被另外一个学习器完全”包住”，则说后则性能优于前者。 如学习器 A 优于 C，但是有交叉时，就要用平衡点（BEP）来衡量。平衡点即 precision 等于 recall 时的值。那么可以认为 A 优于 B。

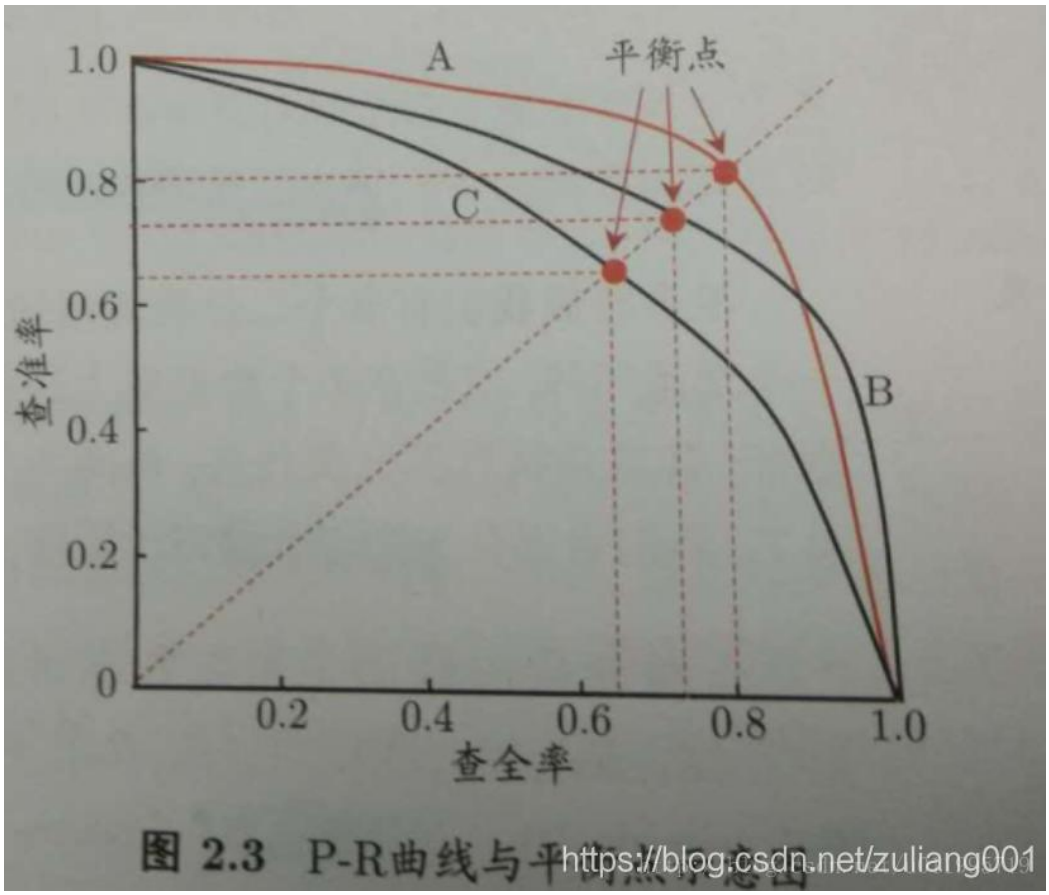

图 2.3 P-R曲线与平衡点示意图

学习资源来自知乎 https://www.zhihu.com/people/brianzhang-37/posts

博客园：https://www.cnblogs.com/pinard/
CSDN：https://blog.csdn.net/u014546828/article/details/106332939
https://blog.csdn.net/pangxing6491/article/details/118391477
https://blog.csdn.net/zuliang001/article/details/83448599