第一章课堂练习与讨论

1.1. 你在经营一家公司，你想开发机器学习算法来解决以下两个问题：

问题 1：你有很多相同的物品。你想预测这些商品在未来 3 个月内的销量。

问题 2：您想用软件来检查每个客户的帐户，并判断该账户是否被黑客攻击或者中毒泄露。

请判断以上问题是分类问题还是回归问题？

答：问题 1 是回归问题，问题 2 是分类问题。

1.2. 在下面的示例中，哪些示例可以通过使用无监督的机器学习算法解决？

(a)提供已标记为垃圾邮件/非垃圾邮件的电子邮件，通过训练获得垃圾邮件过滤器。

(b)提供一组在 web 上找到的新闻文章，将其中具有相同故事的文章进行集合分组。

(c)提供一个客户数据数据库，将客户分组成不同的区域市场。

(d)提供一个诊断为糖尿病或非糖尿病患者的数据集，通过学习判断新患者患有糖尿病还是没患糖尿病。

答：(b)和(c)可以使用无监督的机器学习算法解决。

1.3 请给出监督学习和无监督学习的例子

答：监督学习：人脸识别，场景分类等等

无监督学习：推荐系统，用户细分等

第二章课堂练习与讨论

下面 2.1-2.5 用 MATLAB 编写程序

2.1 计算

$$\frac{1}{2+3^2}+\frac{4}{5}\times\frac{6}{7}$$

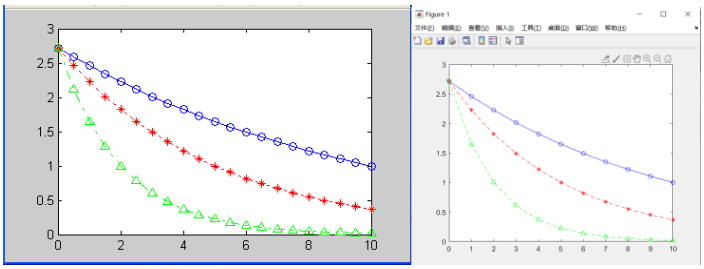
答：x = 1/(2 + 3^2) + (4/5) * (6/7)

```
>> source

x =

    0.7766

fx >>
```

2.2. 有一组测量数据满足 $y = e^{-at+1}$，t 的变化范围为 0~10，用不同的线型和标记点画出 a=0.1、a=0.2 和 a=0.5 三种情况下的下面的曲线。

答：

```
x = 0:10;
y1 = exp(-0.1*x + 1);
y2 = exp(-0.2*x + 1);
y3 = exp(-0.5*x + 1);
figure
plot(x,y1,'-ob',x,y2,'--*r',x,y3,'--^g')
```



2.3 已知：$a = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$，分别计算 a 的行列式，转置矩阵。

答：

```
a = [1,2,3;4,5,6;7,8,9]
A = det(a)
B = transpose(a)
```

```
A =

  -9.5162e-16

B =

     1     4     7
     2     5     8
     3     6     9

fx >>
```

2.4 对于 $AX=B$，如果 $A=\begin{bmatrix}4&9&2\\7&6&4\\3&5&7\end{bmatrix}$，$B=\begin{bmatrix}37\\26\\28\end{bmatrix}$，求解 X。

答：X = inv([4,9,2;7,6,4;3,5,7]) * [37;26;28]

```
X =

   -0.5118
    4.0427
    1.3318

fx >>
```

2.5 请使用 MATLAB 帮助功能了解自带函数 fminsearch，并利用该函数求解以下问题

求函数 $f(x_1,x_2)=100(x_2-x_1^2)^2+(1-x_1^2)^2$ 的最小值.

答：

```
fun = @(x)100*(x(2) - x(1)^2)^2 + (1 - x(1)^2)^2;
x0 = [-1.2,1];
x = fminsearch(fun,x0)
y = fun(x)
```

```
y =

   1.7283e-09

fx >>
```

第三章课堂练习与讨论

3.1 什么是 kernel 思想？Kernel 函数需要满足什么条件？写出三种常用的 kernel 函数。

答：kernel 思想：在低维空间中不能线性分割的点集，通过转化为高维空间中的点集时，很有可能变为线性可分的。

Kernel 函数满足条件：Finitely positive semi-definite functions

常见的 kernel 函数：

Linear Kernel

$$k(x,z)=<x,z>$$

Polynomial Kernel

$$k(x,z)=(<x,z>+1)^r, r\in Z^+$$

RBF(Gaussian) Kernel

$$k(x,z)=\exp(\frac{-\|x-z\|^2}{2\sigma^2}), \sigma\in R-\{0\}$$

3.2* 有 4 个样本（0，1）、（1，0）、（1，1）和（2，1），当采用三种常用的 kernel 函数时求出 kernel 矩阵（可以采用手算，也可以计算机编写程序）

答：

```
x1 = [0;1]
x2 = [1;0]
x3 = [1;1]
x4 = [2;1]

% 线性核
options.KernelType = 'linear'
matrix1 = [
    kernel(x1,x1,options),kernel(x1,x2,options), kernel(x1,x3,options);
    kernel(x2,x1,options),kernel(x2,x2,options), kernel(x2,x3,options);
    kernel(x3,x1,options),kernel(x3,x2,options), kernel(x3,x3,options);
    ]

% 多项式核
options.KernelType = 'poly'
options.KernelPars = 2
matrix2 = [
    kernel(x1,x1,options),kernel(x1,x2,options), kernel(x1,x3,options);
    kernel(x2,x1,options),kernel(x2,x2,options), kernel(x2,x3,options);
    kernel(x3,x1,options),kernel(x3,x2,options), kernel(x3,x3,options);
    ]
% 高斯核
options.KernelType = 'rbf'
options.KernelPars = 5
matrix3 = [
    kernel(x1,x1,options),kernel(x1,x2,options), kernel(x1,x3,options);
    kernel(x2,x1,options),kernel(x2,x2,options), kernel(x2,x3,options);
    kernel(x3,x1,options),kernel(x3,x2,options), kernel(x3,x3,options);
    ]
```

```
matrix1 =

     1     0     1
     0     1     1
     1     1     2
```

线性核

```
matrix2 =

     1     0     1
     0     1     1
     1     1     4
```

多项式核

```
matrix3 =

    1.0000    0.9608    0.9802
    0.9608    1.0000    0.9802
    0.9802    0.9802    1.0000

fx >>
```

高斯核

第四章课堂练习与讨论

4.1 PCA 的物理意义？

答：PCA 本质上是线性变换中的基变换。物理（几何）含义是把样本值投影到方差最大的方向上，从而保留更多信息，消除线性相关信息。

4.2*推导 PCA 公式。

4.3 写出 PCA 算法的步骤。

答：第一步：把样本进行正则化使其均值为 0

第二步：求样本的协方差矩阵

第三步：对协方差矩阵进行特征值分解

第四步：特征值从大到小排列并选择前 k 个特征值

第五步：使用 k 个特征值对应的方向向量投影样本值，从而完成降维

4.4*写出 KPCA 算法的步骤。

答：步骤类似 PCA 只是添加上来一个核函数操作。

第五章课堂练习与讨论

5.1 LDA 的物理意义？

答：LDA 本质上是线性变换中的基变换。物理（几何）含义是把样本值投影到类内距离小，类间距离大的几个方向上，从而实现降维。

5.2 与 PCA 的区别和相似？

答：LDA 和 PCA 本质上都是线性变换中的基变换。

PCA 是把样本值投影到方差最大的方向上。

LDA 是把样本值投影到类内距离小，类间距离大的方向上。充分利用了样本的类别信息。

5.3*推导类间矩阵

5.4 写出 LDA 算法的步骤。

答：第一步求出：S_b^{LDA} 矩阵

第二步求出：S_w^{LDA} 矩阵

第三步求出：$A = S_w^{LDA}$ 逆矩阵*S_b^{LDA} 矩阵

第四步求出：A 进行特征值分解

第四步求出：特征值从大到小排列并选择前 k 个特征值

第五步求出：使用 k 个特征值对应的方向向量投影样本值，从而完成降维

5.5 写出 KLDA 算法的步骤。

答：第一步求出：S_b^{GDA} 矩阵

第二步求出：S_w^{GDA} 矩阵

第三步求出：$A = S_w^{GDA}$ 逆矩阵*S_b^{GDA} 矩阵

第四步求出：A 进行特征值分解

第四步求出：特征值从大到小排列并选择前 k 个特征值

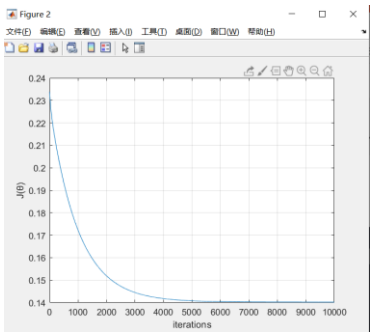
第五步求出：使用 k 个特征值对应的方向向量投影样本值，从而完成降维

5.6*你工作中有与 LDA 相关的问题吗？

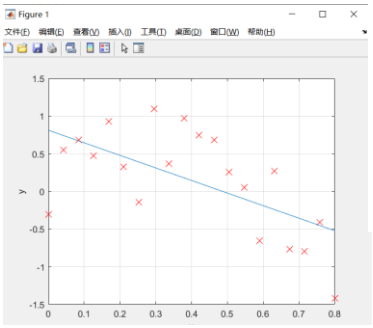
第六章课堂练习与讨论

6.1 用 MATLAB 编写用梯度下降法完成线性回归的程序。

答：1. 实验结果如下图所示，代码在附件中。



Loss 迭代



线性回归

```
theta =

    0.8124
   -1.6683
```

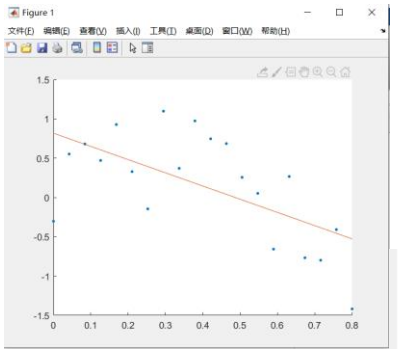
2. 代码结构：

./ LinearRegressionGrad

./main.mat % 直接运行该主文件即可

6.2 用 MATLAB 编写用正则方程完成线性回归的程序。

答：1.实验结果如下图所示，代码在附件中。




```
W0 = 0.8171
W1 = -1.6794
```

2. 代码结构：./ LinearRegressionEquation

./main.mat % 直接运行该主文件即可

6.3 你的工作中有与回归有关的问题吗？

答：有回归的问题，例如，最近研究人行为指标时，需要拟合出指标值和行为表现之间的线性关系。

6.4 写出实现基于 kernel 的线性回归的步骤。

答：第一步：求出 kernel 矩阵

第二步：利用公式求出参数 theta.

$$\alpha = \left(\mathbf{1}_{m\times m} + K\right)^{-1} y$$

$$\theta = X^T \alpha = X^T \left(\mathbf{1}_{m\times m} + K\right)^{-1} y$$

第七章课堂练习与讨论

7.1 逻辑回归与线性回归的相同点和不同点。

答：不同点：线性回归是做回归的，逻辑回归是做分类的。

联系：线性回归+sigmoid 函数=逻辑回归

7.2 写出逻辑回归的损失函数。

答：

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost\left(h_\theta\left(x^{(i)}\right), y^{(i)}\right)$$

$$= -\frac{1}{m}\left[\sum_{i=1}^{m} y^{(i)} \log h_\theta\left(x^{(i)}\right) + \left(1-y^{(i)}\right)\log\left(1-h_\theta\left(x^{(i)}\right)\right)\right]$$

To fit parameters θ :

$$\min_\theta J(\theta)$$

To make a prediction given new x :

Output $h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$

7.3 如何避免过拟合问题？

答：1. 增加训练数据量　　2. 选择复杂度低的拟合模型

3. 设置正则化　　4. 提前结束训练

第八章课堂练习与讨论

8.1.什么是对偶理论。

答：对偶理论（Duality theory）就是研究线性规划中原始问题与对偶问题之间关系的理论。通俗的话说就是，对模型进行等价的对偶变换。

8.2.什么是 KKT 理论。

答：KTT 理论就是对拉格朗日乘子法消除优化问题限制条件的扩充。具体作用是消除优化问题的限制条件。

对于具有等式和不等式约束的一般优化问题

$$\min f(\mathbf{x})$$
$$s.t.\, g_j(\mathbf{x}) \le 0 (j = 1, 2, \cdots, m)$$
$$h_k(\mathbf{x}) = 0 (k = 1, 2, \cdots, l)$$

KKT条件给出了判断 $\mathbf{x}^*$ 是否为最优解的**必要条件**，即：

$$\begin{cases} \dfrac{\partial f}{\partial x_i} + \sum\limits_{j=1}^{m} \mu_j \dfrac{\partial g_j}{\partial x_i} + \sum\limits_{k=1}^{l} \lambda_k \dfrac{\partial h_k}{\partial x_i} = 0, \ (i = 1, 2, \ldots, n) \\ h_k(\mathbf{x}) = 0, (k = 1, 2, \cdots, l) \\ \mu_j g_j(\mathbf{x}) = 0, (j = 1, 2, \cdots, m) \\ \mu_j \ge 0. \end{cases}$$

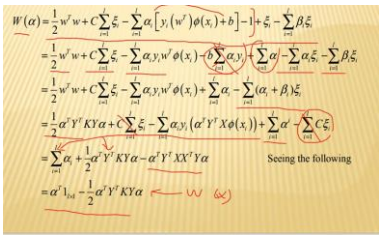
8.3.写出 hard-margin 的公式。

答：超平面的参数计算公式如下。

$$b^* = y_t - w^{*T}\phi(x_t) = y_t - \sum_{i=1}^{\ell} \alpha^*_i y_i \phi(x_i)^T \phi(x_t)$$
$$= y_t - \sum_{i=1}^{\ell} \alpha^*_i y_i \kappa(x_i, x_t). \qquad w = \sum \alpha_i y_i \phi(x_i)$$

8.4 写出 soft-margin 的公式。

答：软超平面的计算公式如下。



$$b^* = y_t - w^{*T}\phi(x_t) = y_t - \sum_{i=1}^{\ell} \alpha_i^* y_i \phi(x_i)^T \phi(x_t)$$
$$= y_t - \sum_{i=1}^{\ell} \alpha_i^* y_i \kappa(x_i, x_t).$$

第九章课堂练习与讨论

9.1.聚类与分类的区别。

答：分类就是向事物分配标签，聚类就是将相似的事物放在一起。

9.2. k-means 实现步骤，优缺点。

答：步骤如下。

1. 从样本中随机选取 k 个点作为聚类中心点
2. 对于任意一个样本点，求其到 k 个聚类中心的距离，然后，将样本点归类到距离最小的聚类中心，直到归类完所有的样本点（聚成 k 类）
3. 对每个聚类求平均值，然后将 k 个均值分别作为各自聚类新的中心点
4. 重复 2、3 步，直到中心点位置不在变化或者中心点的位置变化小于阈值

优点：原理简单，实现容易。

缺点：

1、聚类中心的个数 K 需要事先给定，但在实际中这个 K 值的选定是非常难以估计的，很多时候，事先并不知道给定的数据集应该分成多少个类别才最合适；

2、Kmeans 需要人为地确定初始聚类中心，不同的初始聚类中心可能导致完全不同的聚类结果。

9.3. k-means 与 FCM 的相同点和不同点。

答：相同点：都属于聚类算法

不同点：FCM 是 k-means 的改进型，具体表现为，样本以概率分给不同中心点。也就是软聚类，而 k-means 属于硬聚类。

9.4. FCM 算法的目标函数是什么？隶属度矩阵和聚类中心的公式是什么。

答：目标函数如下

$$J(u_{ij},c_i)=\sum_{i=1}^{I}\sum_{j=1}^{N}u_{ij}^{m}\,\|x_j-c_i\|^2,\ \sum_{i=1}^{L}u_{ij}=1,\ j=1,2,\ldots N$$

隶属于度矩阵和类中心公式如下

$$\begin{cases} u_{ij}=\dfrac{1}{\sum_{k=1}^{L}\dfrac{\|x_j-c_i\|^{\frac{2}{m-1}}}{\|x_j-c_k\|^{\frac{2}{m-1}}}}, & L\times N \\ c_i=\sum_{j=1}^{N}\dfrac{u_{ij}^{m}}{\sum_{s=1}^{N}u_{is}^{m}}x_j & i=1,\ldots,L \end{cases}$$

9.5 我们讲 FCM 算法的计算过程是，先随机给出隶属度矩阵，再计算聚类中心，可否先给出聚类中心，再计算隶属度矩阵？

答：不可以。

第十章课堂练习与讨论

10.1 什么是人工神经网络？

答：人工神经网络（Artificial Neural Network，即 ANN ），是 20 世纪 80 年代以来人工智能领域兴起的研究热点。它从信息处理角度对人脑神经元网络进行抽象， 建立某种简单模型，按不同的连接方式组成不同的网络。

神经网络是一种运算模型，由大量的节点（或称神经元）之间相互联接构成。每个节点代表一种特定的输出函数，称为激励函数（activation function）。每两个节点间的连接都代表一个对于通过该连接信号的加权值，称之为权重，这相当于人工神经网络的记忆。网络的输出则依网络的连接方式，权重值和激励函数的不同而不同。而网络自身通常都是对自然界某种算法或者函数的逼近，也可能是对一种逻辑策略的表达。

10.2* 以三层网络为例，推导 BP 网络训练中输出层和隐藏层的误差函数。

10.3 有哪些激活函数？各自优缺点？

答：1. sigmoid 函数

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

优点：

（a） Sigmoid 函数的输出在(0,1)之间，输出范围有限，优化稳定，可以用作输出层。

（b） 连续函数，便于求导。

缺点：

（a） sigmoid 函数在变量取绝对值非常大的正值或负值时会出现饱和现象，意味着函数会变得很平，并且对输入的微小改变会变得不敏感。

（b） sigmoid 函数的输出不是 0 均值的，会导致后层的神经元的输入是非 0 均值的信号，这会对梯度产生影响。

（c） 计算复杂度高，因为 sigmoid 函数是指数形式

2. Tanh 函数

$$f(x) = \mathrm{Tanh}\, x = \frac{\sinh x}{\cosh x} = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

优缺点：Tanh 函数是 0 均值的，因此实际应用中 Tanh 会比 sigmoid 更好。但是仍然存在梯度饱和与 exp 计算的问题。

3. ReLU 函数

$$f(x) = max(0, x)$$

优点：

（a） 使用 ReLU 的 SGD 算法的收敛速度比 sigmoid 和 tanh 快。

（b） 在 x>0 区域上，不会出现梯度饱和、梯度消失的问题。

（c） 计算复杂度低，不需要进行指数运算，只要一个阈值就可以得到激活值。

缺点：

（a） ReLU 的输出不是 0 均值的。

（b） Dead ReLU Problem(神经元坏死现象): ReLU 在负数区域被 kill 的现象叫做 dead relu。ReLU 在训练的时很“脆弱”。在 x<0 时，梯度为 0。这个神经元及之后的神经元梯度永远为 0，不再对任何数

据有所响应，导致相应参数永远不会被更新。

10.4*学习使用 MATLAB 中的人工神经网络的 nnstart.

10.5 机器学习方法在你的毕业论文中有哪些应用。

答：首先，在图像处理，例如语义分割，图像分类，超分辨等任务中，都会使用到机器学习，特别是深度学习。而最近做的眼动研究中高维数据的降维也会使用到机器学习方法中的 PCA 和 LDA。