**基于sigmoid函数的多元线性回归和**

**基于K-近邻算法的鸟类分类器**

姓名：史文翰

班内序号：04

班级：2014211304

学号：2014211218

**基于多元线性回归和K-近邻算法的鸟类分类器**

**摘要：**

本模型应用多元线性回归与sigmoid函数输出相结合的方法，根据鸟类的翅展和身长完成对鸟类种类（A类、B类）的预测。首先对数据进行可视化处理，近似地估计自变量之间的关系，利用MATLAB中的regress命令完成初步的多元线性回归，利用rcoplot命令做残差分析，尝试相关项、二次项，不断改进模型。由于待解问题是个二分类器的问题，因此在多元线性回归的基础上用sigmoid函数层输出，将输出归一化到{0,1}区间内，完成对鸟类的分类任务。

在模型的改进方面，由于问题的二分类性，尝试了除回归分析以外的分类方法，如基于K-近邻算法（KNN）的最临近分类方法，也可以获得较好的分类效果。

针对问题一，对于鸟类的分类方法在这里给出了两种尝试：多元线性回归和KNN算法。在多元线性回归中，将鸟类的翅展和身长作为特征，通过数据可视化和回归分析计算出拟合直线方程，由于增加了sigmoid输出层，回归结果可被抽象成二维平面上的一条直线，使得在直线的一侧为分类A，直线的另一侧为分类B。对于KNN算法，取部分数据构建模型，其余数据用作检验。将翅展和身长作为样本特征，分类结果作为标签。计算待预测点与数据点的所有特征的欧氏距离，取前k短的距离，在这k个选择数据中，属于哪一类的数据更多，待预测数据就被归类到哪一类上。

针对问题二，需要对给出的新的数据做预测。对于多元线性回归，直接带入回归表达式，用sigmoid输出后判断其值是0还是1即可，也可直观地从数据点与回归直线的位置判断。对于基于KNN的分类器，直接调用相关的函数，计算前k个数据点并分析即可。

针对问题三，分析异常数据点对于模型构建的影响。我们将数据按照要求修改后，重新输入到模型的输入模块（回归）和训练集（KNN模型）中，相当在不改变原理的情况下修改构建模型的数据集，我们通过表格和图形来对比因为数据不同而造成的结果上的差异。

对于多元线性回归模型，最终选用了两个一次项做形如y = β1x1+β2x2的线性回归，剔除异常点后，模型准确率可达92.85%（只“分错”一个数据点）。对于数据对（14.50,19.00）预测为B类，（12.40,18.00）预测为B类，（14.00,20.40）预测为B类。修正错误数据（问题三）后，模型的准确率为85.71%（分错“两个点”），三个待预测样本均分为A类。

对于基于KNN的二分类器，利用前9个数据，取k=5做模型检验，正确率可以达到100%（剩余的五个样本均预测正确）。对于数据对（14.50,19.00）预测为B类，（12.40,18.00）预测为A类，（14.00,20.40）预测为A类。噪声数据对模型的影响是不明显的，修正错误数据后，模型的准确率为80%，三个待预测样本均分为A类（与多元线性回归模型一致）。

**关键字：**多元线性回归，二分类， 鸟类种属识别， KNN算法

**一、问题重述**

题目中给出了某森林中生活得两种鸟的特征数据，分为翅展特征和身长特征。鸟类有自己的种属标签，这里只考虑它们是A类鸟或者是B类鸟。

问题一需要我们给出一个鸟类分类的具体方法。分类模型的输入为一只鸟的翅展和身长，输出为它的分类属于A类还是B类。

问题二需要检验分类器的效果，对于新的数据直接输入到分类器中，按照分类器的输出判定它属于那一种类。

问题三需处理错误数据点对模型构建带来的影响。可先从宏观上给出一般的错误的数据点（噪声数据）对本模型的影响，再改正题中给出的错误数据点并重新构建模型并验证。

**二、符号说明**

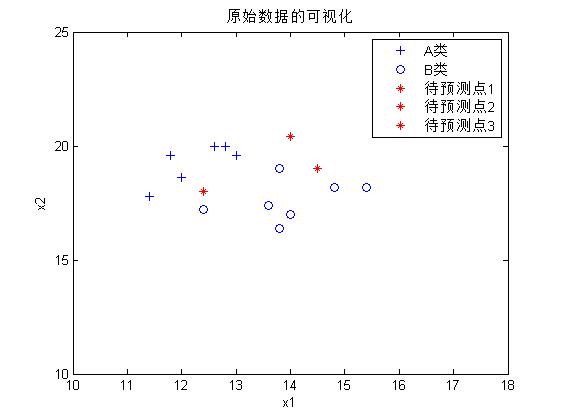
|  |  |  |
| --- | --- | --- |
| 符号名称 | 符号定义 | 类型 |
| bird\_fit | 鸟类的特征矩阵，即数据集 | 矩阵 |
| labels | 鸟类标签 | 标称向量 |
| x1 | 第一自变量：翅展（cm） | 数值向量 |
| x2 | 第二自变量：身长（cm） | 数值向量 |
| x1x2 | 相关乘积项 | 数值向量 |
| b | 回归系数 | 数值向量 |
| bint | 回归系数的回归估计 | 矩阵 |
| r | 残差向量 | 数值向量 |
| rint | 置信区间 | 矩阵 |
| wl | 翅展数据集 | 数值向量 |
| bl | 身长数据集 | 数值向量 |
| crr\_rate | 模型正确率 | 数值 |
| sig\_y | sigmoid函数输出向量 | 数值向量 |
| β | 回归方程参数 | 数值向量 |

**三、模型建立**

**1、基本模型**

**a）基于sigmoid函数的多元线性回归模型**

本模型采用一般的多元线性回归模型[1]作为主体，尝试以常数项、x1项、x2项和x1x2项作为回归模型的自变量。

为了大致分析分类与x1、x2的关系，首先利用MATLAB的作图功能将数据可视化。

可以看出除待预测点之外，A类和B类大体分布在某一条直线或曲线的两侧。

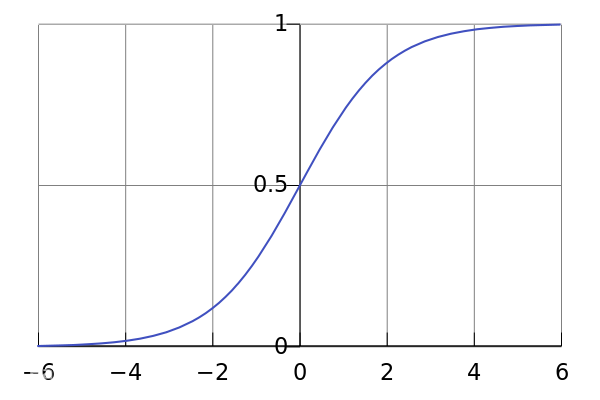
不妨设如下回归方程

y = β0 + β1x1 + β2x2 + ε

或 y = β0 + β1x1 + β2x2 + β3x1x2 +ε

其中向量β为回归方程的参数，ε为服从正态分布的随机误差。由于并不明确x1x2是否存在明显的相关性，不排除使用相关项将其变为非线性回归模型的假设。这里的β向量可以通过线性最小二乘法算得，本模型直接用MATLAB的regress[2]指令实现对β向量的求值。

由于实际的输出y是一个属于{0,1}的变量（其中可将0视为分类‘B’，将1视为分类‘A’），因此需要引入一个归一化函数对结果进行归一化处理。这里我们选择引入sigmoid函数[3]

http://d.hiphotos.baidu.com/baike/s%3D99/sign=a46bd6f1dd33c895a27e9472d01340df/0df3d7ca7bcb0a4659502a5f6f63f6246b60af62.jpg

根据sigmoid函数的性质可知，它类似于阶跃函数，将数轴上的实数值通过sigmoid变换映射到0和1，这里我们取值大于0.5时被认为输出1，否则输出0。这有了这样的输出映射，可以将y归一化至{0,1}集合上。

**b）基于KNN的二分类器模型**

在数据可视化中注意到，A类点与B类点有较高的“聚合度”，这使得简单而有效的KNN临近算法变得可靠起来，可以当作对于多元线性回归模型的一种改进。

KNN的实现基于如下的步骤[4]

1. 计算待预测点的每一个特征值到每一个数据点（不包括自身）的欧氏距离。可以简单地表示为

其中点对（x1，x2）表示待预测鸟的翅展和身长，（y1，y2）取遍已有的用于模型构建的数据点。dhttp://s5.sinaimg.cn/large/52510b1dtd762860813e4&69012为待预测点距某一数据点的欧氏距离。

1. 将计算出的所有距离排序，取前k个（一般是奇数，不超过15）数据点称之为选中的数据点。每个选中的数据点都带有种类标签A或B，若k个数据点中A标签多，则预测该鸟类属于A类，否则预测属于B类。

KNN的优点在于，精度高、对异常值不明感（方便对于问题三的求解），无数据输入假定，且极其易于实现。其缺点在于，计算复杂度和空间复杂度均较高。

**2、简化模型**

**a）基于sigmoid函数的多元线性回归模型简化**

对算法的流程框架，可简单地用下面流程图表示。

样本的可视化处理：对样本进行可视化处理，可从大体上看出其可被一直线或曲线划开，分类特性十分明显。

样本输入：全部的14个样本的两个特征作为自变量，标签作为因变量，输入到regress中。

regress指令回归：调用MATLAB中的regress方法做多元线性回归。

sigmoid输出：对多元线性回归模型做sigmoid输出嵌套，即对回归模型本身，增加sigmoid层使得输出被映射到{0,1}集合上。

残差分析：由stats参数和rcoplot方法[5]等对模型的统计参数做分析和处理。

检验分类器性能：从直观上的图形显示中判断分类器“错分”了几个点，用以对分类器模型本身的性能评价。

**b）基于KNN的二分类器模型简化**

对于算法的流程框架，可简单地用下面流程图表示。

样本的可视化处理：同多元线性回归模型。

模型构建样本输入：取现有样本集中的一部分（多于50%）做训练样本以构建二分类器模型。

距离向量排序：对于待预测的点，直接输入到算法中进行分类预测。

分类输出：根据排序后的点所属类的数量比较，输出预测值，即分类。

剩余样本检验：对于剩余的数据样本，可将它们全部输入到分类器中，根据分类的准确率来评价分类器性能。

**四、模型求解**

**1、基于sigmoid函数的多元线性回归模型求解**

**a）带有相关项的多元线性回归模型**

考虑到鸟类翼展和身长本身可能具有相关性，首先以

y = β0 + β1x1 + β2x2 + β3x1x2 +ε

作为回归方程，其中β向量可以由MATLAB的regress指令求得。调用语句如下：

[b,bint,r,rint,stats] = regress(bi\_labels,bird\_fit\_x1x2,0.05)

其中bi\_labels是全部14个鸟类数据的标签信息（即分类信息），bird\_fit\_x1x2是鸟类样本的特征矩阵。

得到相应的参数后，输出用sigmoid层嵌套即可得到最终的分类结果。定义sigmoid函数如下：

function [ y ] = sig( x )

y = 1/(1+exp(-x));

end

调用sig函数即可输出样本的分类值，如

pre1 = sig(beta1\*14.50+beta2\*19.00)

即是对翼展14.50cm，身长19.00cm的鸟类样本进行分类预测。输出的值还需经过以0.50为界限的sigmoid分类（由函数性质决定）。

语句

if(sig\_y(i,:)>0.5 && labels(i) == 'A')

检验这样的一个事实：判断经过模型输出后，它的分类与原来的分类是否一致。

**b）不带相关项的多元线性回归模型求解**

对于不相关项的回归方程，变化为

y = β0 + β1x1 + β2x2 + ε

其中删除了相关项x1x2，这是因为基于这样的考虑：身长和翅展未必有足够强的相关性。该模型的其他求解过程与模型a）完全一致。

**2、基于KNN的二分类器模型的求解**

本模型用纯C语言编写，考虑到KNN的空间和时间复杂度均较大，因此采用效率较高的C语言和配套的GCC编译器来完成模型的构建和求解。

首先可将鸟类样本归纳为如下的结构

struct bird{

double wl;

double bl;

char label;

};

typedef struct bird BIRD;

其中wl表示了鸟类样本的翅展，bl表示身长，label是分类。由于wl和bl的数据集具有相同的单位，因此不需要做额外的数据预处理的工作。

将数据录入后，可调用如下的方法

char knn(double nwl ,double nbl,int k,int num)

knn方法返回一个分类标识，它要求给出这样的输入：nwl是待预测鸟类的翅展，nbl是待预测鸟类的身长，k值是knn的向量选择阈值，num是究竟用多少样本来做分类器的训练。

对于分类器的评价，我们可以简单地编写一个评价函数

double check\_knn(int num)

评价时要求给出这个模型的剩余样本总数，分类器对于每一个剩余样本均做一次预测，将预测结果与真值对比，方法返回一个准确率，即预测正确的数量与剩余样本总数之比。

**五、模型结果分析和改进**

**1、基于sigmoid的多元线性回归模型检验**

在MATLAB中调用库函数regress，可求得带有相关项的回归模型β向量的预测结果，与回归模型相关的统计变量输出如下

|  |  |  |
| --- | --- | --- |
| **参数** | **参数估计值** | **参数置信区间** |
| β0 | -15.2328 | -38.7871 8.3215 |
| β1 | 0.8712 | -0.8430 2.5854 |
| β2 | 1.0447 | -0.2001 2.2895 |
| β3 | -0.0620 | -0.1525 0.0285 |
| **R^2 = 0.8503 F = 18.9408 p<0.0002 s^2 = 0.0513** | | |

即回归方程可表示为：

y = -15.2328 + 0.8712 x1 + 1.0447 x2 – 0.0620 x1x2

经过简单的置信区间分析可以发现，参数β0的置信区间范围很大，且包含了0点，这表明常数回归变量是很不著的，可以直接从回归方程中删去。进一步观察可以发现，几乎所有的β参数都包含了0点，这表明这个模型可靠性很低，因此可以直接排除带有相关项的模型。

注：与带有相关项模型类似的，如完全二次多项式模型，也通过类似的方法检验，可得到大体相近的结果（限于篇幅，在此不再列出），即置信区间均包含零点且常数回归变量很不显著。

经过上述的模型尝试，我们接着验证不带相关项的线性回归模型。regress方法调用语句为

[b,bint,r,rint,stats] = regress(bi\_labels,bird\_fit,0.05)

可得到如下的结果表格

|  |  |  |
| --- | --- | --- |
| **参数** | **参数估计值** | **参数置信区间** |
| β0 | 0.8096 | -1.7092 3.3273 |
| β1 | -0.3001 | -0.4122 -0.1880 |
| β2 | 0.1955 | 0.0846 0.3064 |
| **R^2 = 0.8155 F = 24.3142 p<0.0001 s^2 = 0.0575** | | |

可以看出在删去了相关项之后，虽然模型中R^2值（因变量有多大的比率可以被模型确定）有略微的下降，但F值有所提升且远大于F-检测阈值。最重要的是，β0置信区间大大缩小，且β1和β2的参数置信区间不再包含0点，这表明回归变量x1和x2在此模型下对于因变量y的影响显著[6]。由于β0的置信区间仍包含0点，不妨剔除常数项，最后得到的回归方程如下

y = -0.3001 x1 + 0.1955 x2

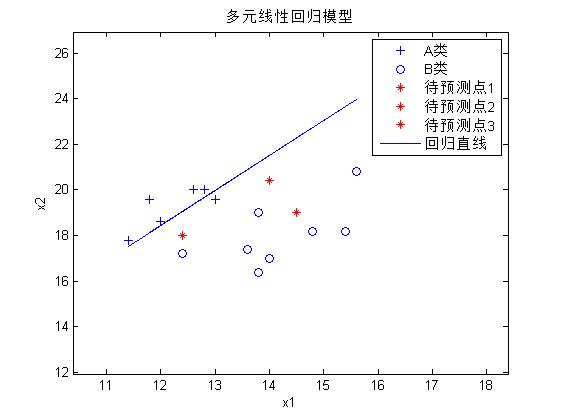
将回归直线可视化到二维数据图中，可建立如下的直线方程

0 = β1 x1 + β2 x2

也即 x2 = - β1/β2 \* x1

这是由于sigmoid的分类依据是根据输入值得正负而定的（参照sigmoid函数性质）。

做出回归直线图像如下



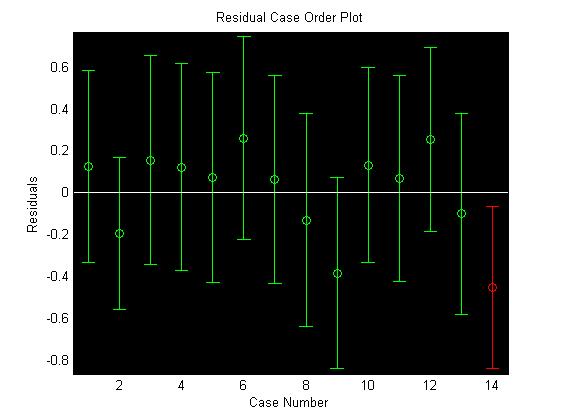
可见，从直观上来看，回归直线将相平面一分为二，sigmoid的性质告诉我们，回归直线的上方代表预测A类，下方代表预测B类。也就是说，在该模型下，分类器只分错了一个点。

对于问题二，三个待预测点在该模型下均被预测为B类。

可以看出分类器的效果较好，附加模型的残差分析，调用rcoplot方法如下

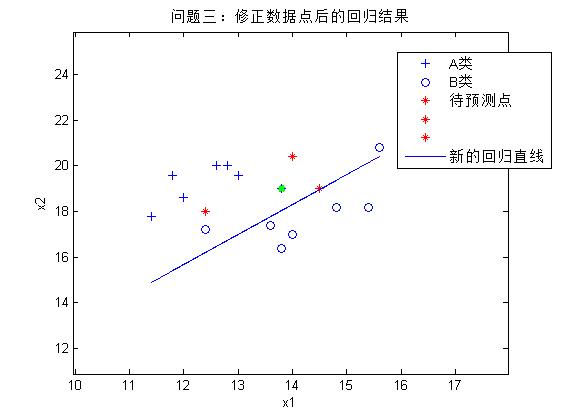
rcoplot(r,rint)

得到如下的残差图



从残差分析中可见，第14个数据对于该模型来说是一个偏离残差范围的异常点，应该从数据集中剔除，但经过尝试后，发现剔除后会引入新的残差偏离问题，考虑到本数据的偏离并不明显，因此将此数据保留。

对于问题三，我们改变原有的数据集（将（13.80,19.00）的label置为‘A’），并重新对该模型以相同的方式建模，可得到如下的结果。



**修改后的数据点**数据

|  |  |  |
| --- | --- | --- |
| **参数** | **参数估计值** | **参数置信区间** |
| β0 | 0.2667 | -2.5289 3.0623 |
| β1 | -0.2807 | -0.4051 -0.1562 |
| β2 | 0.2146 | -0.4051 -0.1562 |
| **R^2 = 0.7773 F = 19.1953 p<0.0003 s^2 = 0.0709** | | |

从图中可以直观地看到，由于改动的点位于原曲线的下方，因此它的分类变化势必造成回归直线的向下移动。

从表中可见，更改了这个错误之后，R^2值和F值均有下降，方差上升，这说明修改后的模型变得没有比原始模型可靠，这是由于，被改变的点影响了模型的分类性能，因为它使得原来比较明显的分类界限变得模糊起来。

可以看出，在改动后的模型下，有两个点被“分错”，且待预测的三个点全部被分为A类，这与原始模型的差别相当大。

**2、基于KNN的二分类器模型检验**

从上述描述中可以看出，由于分类问题是二分类的，对于多元线性回归的适应性并不好，得到的效果不能令人满意，因此考虑用另外一种方法，KNN的方法来构造分类器模型。

我们选择前9个数据作为模型的训练样本，后5个数据用来检验分类器的可靠性。在主函数中直接调用方法knn

crate = check\_knn(MAX\_USE);

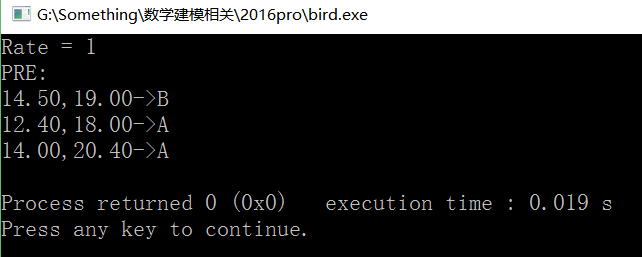
我们同时求解问题二的预测问题，调用如下语句完成预测

ch = knn(14.50,19.00,MAX\_K,MAX\_USE);

ch = knn(12.40,18.00,MAX\_K,MAX\_USE);

ch = knn(14.00,20.40,MAX\_K,MAX\_USE);

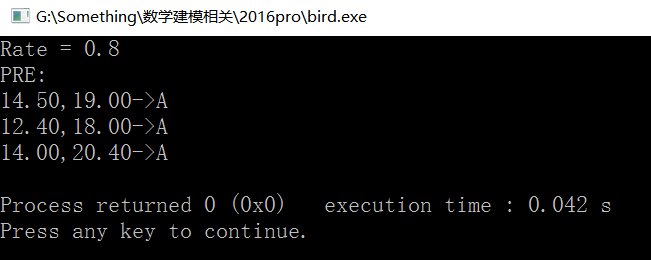
这里的宏变量MAX\_USE是使用的最大训练样本数量，这里取值为9，MAX\_K是k值，这里取值为5。进入控制台界面，可得到如下的结果输出



Rate表示对于剩余样本预测的准确率，可以看到该模型对于剩余样本预测的准确率为1，也就是剩余的5个样本全部预测正确，这是个令人满意的结果。在这样的模型下，对于三个待预测的点的预测结果分别为：B类，A类，A类。

对于问题三，我们同样将相应的点的label从B类修改为A类，请注意，我们刻意让此点作为模型的训练样本而非剩余样本，这样便于我们了解样本的变化带来的对模型的影响。

修改后对同样的剩余样本、同样的待预测点进行预测，有如下的结果



经过对比之后不难看出，这个点的改动使得分类器的准确率下降为0.8（下降明显的很大一部分原因是因为训练样本过少，导致对差异十分敏感）。且对于三个待预测点的预测结果也有变化，第一个待预测点的预测结果有B变为了A，这说明在计算该K邻近向量的时候，变化的数据点起到了关键作用。

六、模型评价

由于本题是个二分类及预测问题，多元线性回归的效果往往不是很好，在考虑删去了相关项、常数项后，模型整体效果较好，但受噪声数据的影响仍然较大，当一个原始数据不可靠时，会对模型整体的效果造成很大的影响。此外，对模型的准确性评价指标还有待完善，若只有“分对分错”的指标来看，即使在最差的情况下，也总有一类的点被完全“分对”，因为无论如何，他们都在回归直线的同一侧。

对于KNN的二分类器，很适用于简单的，分类倾向明显的数据集（从数据可视化中可以看到）。KNN 算法有精度高、对异常值不明感（可从对于问题三的分析中看出）、无数据输入假定的优点。同时，利用独立的剩余样本对分类器性能做检验并以此为准确率指标是合适的。但是，它本身需要计算欧氏距离，且需要经过数据排序，因此计算复杂度很高（因此选择C语言实现），在面对大量的数据样本时可能会付出相当大的时间代价。

七、总结

在建模的开始阶段，对数据的可视化分析是选择模型的基础。在可视化图形中，注意到A类和B类的鸟类可以被某一条明显的“界限”分为两类，且他们本身类别的聚合度很高，因此采用基于sigmoid的多元线性回归和KNN算法都是可行的。

本模型是对多元线性回归的具体事例的实现，由于问题最终被归为二分类问题，仅仅依靠多元线性回归方法是不够的，不过通过类似阶跃函数的sigmoid函数可以很容易将数轴上的值映射到集合{0,1}中，恰好解决了二分类问题。

对于多元线性回归，效果并不是太理想，对于异常数据的敏感性太高，且得到的回归直线很难“一步到位”，将所有的数据点正确分类。但考虑到相关项和高阶项的引入是没有必要的（如对相关项模型的分析那样），因此这个方法是有明显的局限性的。

注意到问题的本质是二分类，因此考虑到用一个简单实用的机器学习的基础算法，即K-近邻算法，简称KNN，它有对异常值不敏感、精度高等特性，由于本题的数据集较小，十分适合用KNN做二分类器，事实也证明其效果要好于多元线性回归模型。

**参 考 文 献：**

[1] 姜启源等，《数学模型》，北京：高等教育出版社，2011年1月第4版.327 。

[2] 谷震平. 常用的回归分析Matlab命令.

http://blog.csdn.net/guzhenping/article/details/43314333, 2016.06.14。

[3] cowboy\_wz. S形函数：Sigmoid函数.

http://blog.csdn.net/chl033/article/details/4851541, 2016.06.14。

[4] Peter Harrington ，《机器学习实战》，北京：人民邮电出版社，2013年6月第1版.15 。

[5] 小爷永远不死.matlab多元线性回归.

http://blog.csdn.net/fobdddf/article/details/9834849, 2016.06.13。

[6] 姜启源等，《数学模型》，北京：高等教育出版社，2011年1月第4版.328 。

附录：

1. 源程序索引：

bird.cpp

（用C语言实现的基于KNN算法的二分类器，给出了完整的程序，解决了问题一，问题二和问题三）

bird.m

（MATLAB实现多元线性回归，给出了完整的程序和一些参数上的尝试，解决了问题一和问题二，问题三可通过手工调整程序参数实现）

sig.m

（sigmoid函数在MATLAB中的实现）

1. 函数方法及数据结构索引（C语言）：

BIRD arr\_bird[BIRD\_NUM]; // 鸟类特征集

char labels[BIRD\_NUM]； // 鸟类标签集

void init\_arr(void) // 初始化鸟类结构数组，数据导入

char knn(double nwl ,double nbl,int k,int num) // KNN驱动函数

double check\_knn(int num) // 计算KNN分类器的准确率