编程练习 2：逻辑回归

机器学习

# 介绍

在本练习中，您将实现逻辑回归并将其应用于两个不同的数据集。在开始编程练习之前，我们强烈建议观看 the视频讲座并完成相关主题的复习问题。

要开始练习，you 您需要下载初学者代码，然后解压缩其内容到you 您希望完成练习的目录中。如果需要，在开始此练习之前，请使用Octave/MATLAB中的cd 命令更改为此目录。

find instructions您还可以 also在课程网站的 the “"en- virave/MATLAB"中找到安装 installing八度/MATLAB的说明。

## 本练习中包含的文件

ex2.m - Octave/MATLAB script that steps you through the exercise ex2 reg.m - Octave/MATLAB script for the later parts of the exercise ex2data1.txt - Training set for the first half of the exercise ex2data2.txt - Training set for the second half of the exercise submit.m - Submission script that sends your solutions to our servers mapFeature.m - Function to generate polynomial features plotDecisionBoundary.m - Function to plot classifier’s decision bound- ary

[*\**] plotData.m - 绘制 2D 分类数据的函数

[*\**] sigmoid.m - 西格莫伊德功能

[*\**] 成本函数.m - 逻辑回归成本函数

[*\**] 预测.m - 逻辑回归预测函数

[*\**] 成本功能注册 - 定期逻辑回归成本

*•*指示您需要完成的文件

Throughout the exercise, you will be using the scripts ex2.m and ex2 reg.m. These scripts set up the dataset for the problems and make calls to functions that you will write. You do not need to modify either of them. You are only required to modify functions in other files, by following the instructions in this assignment.

## 在哪里获得帮助

本课程中的练习使用八角形 1[1](#_bookmark0)或 MATLAB，这是一种高级程序明语言，非常适合数值计算。如果您you 没有have 安装八度或MATLAB，请参阅课程网站的"环境设置说明"中的安装说明。

在 Octave/MATLAB 命令行中，键入帮助，然后键入func-

tion名称显示内置函数的文档。例如，帮助绘图将显示用于绘图的帮助信息。有关八度扫描函数的进一步文档，请参阅found at the [八度文档页面](http://www.gnu.org/software/octave/doc/interpreter/)。MAT-LAB 文档可在be [MATLAB 文档页](http://www.mathworks.com/help/matlab/?refresh=true)中找到。

我们还强烈建议 encourage使用 the在线**讨论** to with与其他学生讨论前讨论。但是， do不要查看 at其他人编写by 的任何源代码 code或与他人共享您的源代码。

# 逻辑回归

在本练习的这一部分，您将构建一个逻辑回归模型，以预测学生是否被大学录取。

假设您是大学系的管理员，you 您希望根据每位申请人在两次考试中的结果确定他们获得广告任务的机会。您拥有以前申请人的历史数据，您可以将这些数据用作逻辑回归培训集。对于每个培训示例，您都有申请人在两次考试和录取决定上的分数。

Your task is to build a classification model that estimates an applicant’s probability of admission based the scores from those two exams. This outline and the framework code in ex2.m will guide you through the exercise.

1八度是 isMATLAB的 a免费替代品。对于 the编程练习，您可以 are自由使用八度或 MATLAB。

## 可视化数据

在开始实现任何学习算法之前，good 如果可能，最好可视化数据。在ex2.m的第一部分，代码将加载数据，并通过调用函数图Data将其显示在二维图上。现在，您将now 在plotData 中完成代码，以便它显示如图 [1](#_bookmark1)所示，其中轴是两个考试分数，以及正和

负例用不同的标记显示。

100

Admitted Not admitted

90

80

70

考试 2 分

60

50

40

30

30 40 50 60 70 80 90 100

考试 1 分

图 1：训练数据的散射图

为了帮助您更熟悉绘图，我们将 plotData.m 留空，以便您可以you 尝试自己实现它。 但是，这是一个*可选的（未分级）练习*。We 我们还在下面提供我们的实现，you 以便您可以复制或 or引用 to它。 If如果您选择 to复制我们的示例， make请确保 you通过con复制八度/MATLAB文档来了解其每个命令正在执行的操作。

百分比 查找正和负示例的索引位置 = 查找（y=1）;否定 = 查找（y[ 0）;

% 绘图示例

图（X（位置，1），X（位置，2），'k+'，"'k+','线宽'",2,... ...

'标记大小',, 7);

情节（X（1、1）、X（2）、"Ko"，"科,'标记面颜色',"，"和",..., '标记大小'', 7);

## 实现

### 热身练习：sigmoid功能

在开始当前成本函数之前，请记住，物流回归- zion 假设定义 s：s:

*h-（x）* - ( *g*(*（T*  *x）*)*，*

其中函数*g* 是 sigmoid 函数。sigmoid 函数定义为：

1

*g*（*z*）|

*.*

1 + *e−z*

您的第一步是在sigmoid.m中实现此函数，以便程序的其余部分可以调用它。完成后，请尝试在Octave/MATLAB 命令行上调用sigmoid（x）来测试几个值。对于 x 的大正值x，sigmoid 应接近 1，而对于较大的负值，sigmoid 应接近 0。评估 sigmoid（0） 应该给你正好 0.5。代码还应使用矢量和矩阵。**对于矩阵，函数应对每个元素**执行 sigmoid 函数。

您可以通过在Oc- tave /MATLAB命令行键入提交来提交用于 grading的解决方案。提交脚本将提示you 您输入您的登录电子邮件和提交令牌，并询问要提交哪些文件you want。可以从网页上获取分配的提交令牌。

*您现在应该提交您的解决方案。*

### 成本函数和梯度

for现在 you，您将实现 the逻辑回归的成本函数和梯度。在 costSins.m 中完成代码以返回 the成本和 and梯度。

回想一下，逻辑回归中的成本函数是

1

*米*

*J* (*θ*) =

*米*

区［*−y*(*i*) log(*h* (*x*(*i*))) *−* (1 *− y*(*i*)) log(1 *− h* (*x*(*i*)))－ *,*

*i*=1

成本的梯度是长度与*Ⅰ*其中 *j*th

*Ⅰ*

*Ⅰ*

元素（对于*j*  = 0*，*  1*，*  *. . . . n* ） 的定义如下：

*J* *（）* 1

=

*2.1.1.2.*

*米*

*j*

区(*h* (*x*(*i*)) *− y*(*i*))*x*(*i*)

*i*=1

请注意，虽然此渐变看起来与线性回归相同游戏-服务，公式实际上不同，因为线性和逻辑回归有不同定义 *hθ*(*x*).

*米*

*Ⅰ*

*J*

you 完成后，ex2.m将使用*\**的初始参数调用您的 cost 函数。您应该看到成本约为 0.693。

*您现在应该提交您的解决方案。*

* + 1. **使用**fminunc的学习参数

在上一个赋值中，通过实现分级下降，您找到了线性再入回归模型by 的最佳参数。 您编写了成本函数并计算其梯度，然后相应地执行渐变下降步骤。这一次，您将will使用一个称为fminunc的Octave/- MATLAB 内置函数，而不是采取梯度下降步骤。

八度/MATLAB 的fminunc 是一个优化解算器，用于查找 the无约束[2](#_bookmark2)函数的最小值。对于逻辑回归，您希望want 使用参数 *\**. 优化成本函数 *J*  （*\**）。

具体来说，您将使用 fminunc查找逻辑回归成本函数的最佳参数 *\*，*给定一个固定数据集（X 和 *y* 值）。You 您将传递给 fminunc 以下 inputs：

* + - * we我们尝试 to优化的 the参数的初始值。
      * for the一个函数，当给定 the训练集和特定 particular( *\**时，计算 to with and与*数据集的 \** （*X*，  *y*） 的逻辑回归成本和梯度

In ex2.m, we already have code written to call fminunc with the correct arguments.

*≤*

2优化中的约束通常指 to constraints参数的约束 the， for例如，绑定可能的值 *\** 可以采用的约束（例如， *=*  1）。逻辑回归确实有 not have这样的约束， allowed to因为*允许 \**  is 获取任何实际值。

% Set options for fminunc

options = optimset('GradObj', 'on', 'MaxIter', 400);

% Run fminunc to obtain the optimal theta

% This function will return theta and the cost [theta, cost] = ...

fminunc(@(t)(costFunction(t, X, y)), initial theta, options);

In this code snippet, we first defined the options to be used with fminunc. Specifically, we set the GradObj option to on, which tells fminunc that our function returns both the cost and the gradient. This allows fminunc to use the gradient when minimizing the function. Furthermore, we set the MaxIter option to 400, so that fminunc will run for at most 400 steps before it terminates.

为了指定 the我们最小化的实际函数 we a，我们使用 use"速记"来指定具有#（t） （成本函数（t、 X、 y） 的函数。这将创建一个函数，该函数带有参数 t，调用您的 cost函数。这使我们能够包装成本函数与fminunc一起使用。

如果 have成本 the函数完成正确，fminunc将收敛到正确的优化参数上，并返回成本和 *\**的最终值。请注意，通过使用using fminunc，, 您不必自己编写任何循环，也不必像you 设置梯度下降一样设置学习速率。这一切都由fminunc完成：您只需要提供一个计算成本和梯度的函数。

一旦fminunc完成，ex2.m将使用*\**的最佳参数调用您的 cost函数函数。您应该看到成本约为 0.203。

然后，此最终*=* 值将用于be 绘制训练数据的决策边界，从而产生类似于图 [2](#_bookmark3)的数字。 我们还also鼓励您查看图图"决定边界.m"中的代码，了解如何how使用*\**值绘制此类边界。

### 评估逻辑回归

学习参数后you ，您可以使用模型来预测特定学生是否将被录取。 For 对于考试 1 分为 45 且考试 2 得分为 85 的学生，您应该should会看到0.776 的录取概率。

to评估我们发现的参数质量we have found的另一种方法是 to了解 the所学模型对我们的训练集的预测 on情况 well。在此 this

100

Admitted Not admitted

90

80

70

考试 2 分

60

50

40

30

30 40 50 60 70 80 90 100

考试 1 分

图 2：具有决策边界的训练数据

部分，你的任务是在预测.m完成代码。预测predict 函数将生成给定数据集和学习参数向量 \* 的"1"或"0"预测*θ*。

在you have 预测.m中完成代码后by ，ex2.m脚本将通过计算分类器正确示例的百分比来报告分类器的训练准确性。

*您现在应该提交您的解决方案。*

# 定期逻辑回归

在练习的这 part一部分 the中，您将 will实施定期zed 逻辑回归，以预测制造工厂的微芯片是否通过质量 assur- ance （QA）。在QA 期间，每个微芯片都会经过 through各种测试，以确保其正常运行。

假设您是工厂的产品经理，并且在两个不同的测试中，您有一些微芯片的测试结果。通过这两个测试，您希望确定微芯片是应该接受还是拒绝。 为了帮助您 you 做出决策，您u拥有过去微芯片上的测试结果数据集，从中可以 can构建 a逻辑回归模型。

You will use another script, ex2 reg.m to complete this portion of the exercise.

## 可视化数据

与本练习的前一部分类似，plotData 用于生成如图 [3](#_bookmark4)所示的数字，其中轴是两个测试分数，正（y = 1，已接受）和负（y = 0，拒绝）示例显示不同的标记。

1.2

和 1

和 0

1

0.8

0.6

0.4

微芯片测试2

0.2

0

•0.2

•0.4

•0.6

•0.8

•1 ±0.5 0 0.5 1 1.5

微芯片测试 1

图 3：培训数据图

图[3](#_bookmark4) 显示，我们的数据集无法通过be by 绘图的直线分隔成正和负示例。因此，逻辑回归的直向应用 of logistic regression在此数据集中be will不会很好地运行 well，因为逻辑回归只能找到线性决策边界。

## 功能映射

way 更好地拟合数据的一种方法是从每个数据点创建更多要素。 In在提供的函数 mapFeature.m中，我们将 will将 the要素映射到 of*x*1 和*x*2 的所有多项用语中，最多到 the第六个电源。

1 

*x*1

·

*x*2

2

·

*Ⅹ*

·

·

·

1

·

·

*Ⅹ*

2

*x*1*x*2

地图功能(*x*) =

*Ⅹ*

2

3

·

·

2

·

·

1

.

× 1*x*5 ×

6

*Ⅹ*

2

a由于 of此映射，我们的矢量有两个要素（在 on

两个QA测试）已转换为28维矢量。物流 logistic

训练于此高维要素矢量的回归分类器具有更复杂的决策边界，并在我们的二维图中绘制时显示为非线性。

虽然 the特征映射 allows允许我们 to构建更具 more表现力的分类器，但它也更容易过度拟合。在演练的下一部分，you 您将实施规范化逻辑回归以适合数据，并亲自了解how 正规化如何帮助解决过度拟合问题。

## 成本函数和梯度

the现在，您将 will实现代码来计算重新实现逻辑回归的成本函数和梯度。在成本功能 Reg.m中完成代码以返回成本和梯度。

回想一下，逻辑回归中的正规化成本函数是

*N*

1

*米*

*J* (*θ*) =

*米*

*i*±1

*j*'1

区［*−y*(*i*) log(*h* (*x*(*i*))) *−* (1 *− y*(*i*)) log(1 *− h* (*x*(*i*)))－ + *λ*

区*θ*2*.*

注意 你 应该 不 规范 参数 *θ*0. 在 **奥克塔维安/MAT-实验室**, 记得 索引 开始 从 1, 因此 你 应该 不 是 使塔（1)参数（对应于*θ*0)在代码中。成本函数的梯度是*j*th 元素定义为 遵循：

*Ⅰ*

*Ⅰ*

2*米*

*J*

*Ⅰ*

*J*

*J* *（）* 1

=

*2.1.1.2.*

*米*

0

区(*h* (*x*(*i*)) *− y*(*i*))*x*(*i*)

*i*=1

对于*J*= 0

*米*

*J* *（）*

=

*θj*我不得不

1

*m*

*米*

区

*i*=1

(*hθ* (*x*(*i*)) *− y*(*i*))*x*(*i*)

*λ*

• *m=j* j j *=*  *≥* 1

Once you are done, ex2 reg.m will call your costFunctionReg function using the initial value of *θ* (initialized to all zeros). You should see that the cost is about 0.693.

*j*

*您现在应该提交您的解决方案。*

* + 1. **使用**fminunc的学习参数

Similar to the previous parts, you will use fminunc to learn the optimal parameters *θ*. If you have completed the cost and gradient for regularized logistic regression (costFunctionReg.m) correctly, you should be able to step through the next part of ex2 reg.m to learn the parameters *θ* using fminunc.

## 绘制决策边界

为了帮助您you 可视化by 此分类器所学的模型，我们亲pro-化了函数图"决策边界.m"，该函数绘制了分隔正和负示例的（非线性）决策边界。在图"决定边界.m"中，我们通过将 com- the分类器的预测 predictions放在均匀间隔的网格上 an来绘制 the非线性决策边界，然后 and绘制了预测从 y 变化的轮廓图*y* = 0 到 *y*  = 1。

After learning the parameters *θ*, the next step in ex reg.m will plot a decision boundary similar to Figure [4](#_bookmark5).

## 可选（未分级）练习

在练习的这一部分中，您将尝试数据集的不同规范化参数，以了解规范化如何防止过度拟合。

请注意决策边界中的更改，因为您的变化*\**。与小

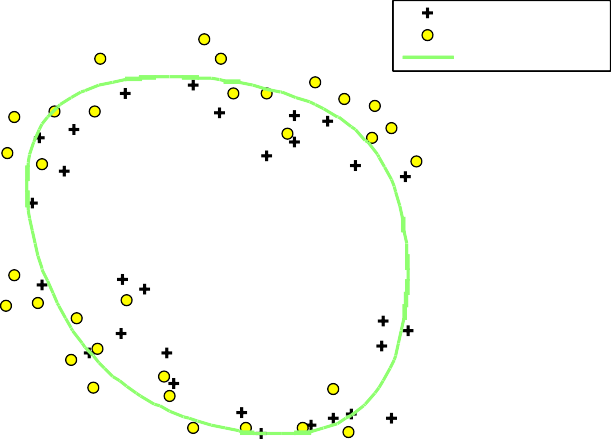
*λ*,你应发现分类器几乎使每个训练示例都正确，但绘制了非常复杂的边界，从而过度拟合数据（图）[5](#_bookmark6)).这不是好决策边界：例如，它预ts 的 点*Ⅹ*=( 0*.*25*,*1*.*5） 被接受 （*和*= 1），这似乎是给定培训时的错误决策 设置。

*•*

较大的*\*，*, 您应该看到一个图，显示一个更简单的决策边界，它仍然很好地分隔正数和负数。如何- 如果 *\** 设置为 to too过高 a的值，您将 will not得不到 a良好的拟合，决策 the边界不会很好地跟踪数据，从而不足数据（图 [6](#_bookmark7)）。

*您无需为这些可选（未分级）练习提交任何解决方案。*

1.2



和 1

和 0

决策边界

lambda = 1

1

0.8

0.6

0.4

微芯片测试2

0.2

0

•0.2

•0.4

•0.6

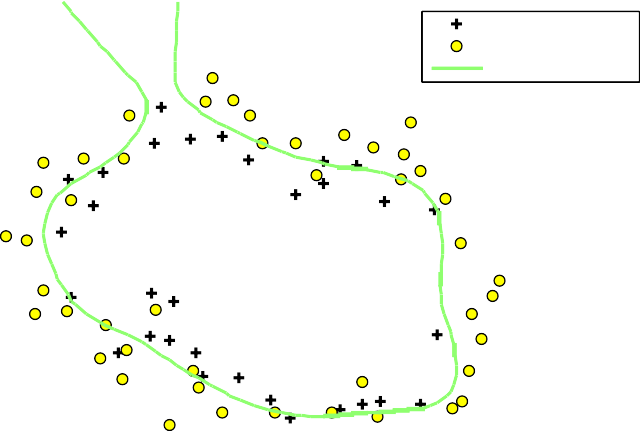
•0.8

•1 ±0.5 0 0.5 1 1.5

微芯片测试 1

图 4：具有决策边界的训练数据 （=*=*  1）

1.5



和 1

和 0

决策边界

lambda = 0

1

0.5

微芯片测试2

0

•0.5

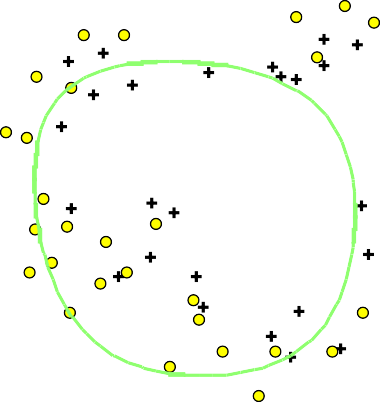
•1

•1 ±0.5 0 0.5 1 1.5

微芯片测试 1

图 5：无正化（过度拟合） （=*=*  0）

1.2



和 1

和 0

决策边界

lambda = 100

1

0.8

0.6

0.4

微芯片测试2

0.2

0

•0.2

•0.4

•0.6

•0.8

•1 ±0.5 0 0.5 1 1.5

微芯片测试 1

图 6： Too 多正态化 （拟合） （=*=*  100）

# 提交和分级

完成分配的各个部分后，请务必使用提交功能系统将您的解决方案提交到我们的服务器。以下是本练习的每个部分的评分的细目。

|  |  |  |
| --- | --- | --- |
| **部分** | **已提交文件** | **点** |
| 西格莫德函数 | 西格莫伊德.m | 5 分 |
| 逻辑回归梯度的计算成本  预测函数 | 成本函数.m  成本函数.m预测.m | 30分  30分  5 分 |
| 正规化 LR 的计算成本  正规化 LR 的渐变 | 成本 功能 注册  成本 功能 注册 | 15分  15分 |
| 总积分 |  | 100 点 |

您可以多次提交您的解决方案，我们将只考虑最高分。