编程练习5：

正则线性回归和偏方差

机器学习

# 介绍

在本练习中，您将实现正则化线性回归，并使用它来研究具有不同偏差方差特性的模型。在开始编程练习之前，我们强烈建议观看视频讲座并完成相关主题的复习问题。

要开始练习，您需要下载起始代码并将其内容解压缩到您希望完成练习的目录中。如果需要，在开始本练习之前，请使用octave/matlab中的cd命令更改到此目录。

您也可以在课程网站的“环境设置说明”中找到安装octave/matlab的说明。

## 本练习中包含的文件

ex5.m-octave/matlab脚本，引导您完成练习

ex5data.mat-dataset

submit.m-将您的解决方案发送到我们的服务器的提交脚本

featurenormalize.m-功能规范化函数

fmincg.m-函数最小化例程（类似于fminunc）

plot fit.m-绘制多项式拟合

trainlineareg.m-使用您的成本函数训练线性回归

[]linearregostfunction.m-正则化线性回归cosT函数*？*

[]learning curve.m-生成学习曲线*？*

[]polyFeatures.m-将数据映射到多项式特征空间*？*

[]validation curve.m-生成交叉验证曲线*？*

*？*指示需要完成的文件

在整个练习中，您将使用脚本ex5.m。这些脚本为问题设置数据集，并调用将要编写的函数。您只需要按照本分配中的说明修改其他文件中的函数。

## 在哪里寻求帮助

本课程的练习使用Octaveor Matlab，这是一种高级编程语言，非常适合数值计算。如果您没有安装八度音阶或matlab，请参考课程网站“环境设置说明”中的安装说明。[〔1〕](" \l "_ftn1" \o ")

在octave/matlab命令行中，键入帮助，然后键入函数名，显示内置函数的文档。例如，帮助图将显示用于绘制的帮助信息。有关八度函数的更多文档，请参见八度文档页面。matlab文档可以在matlab文档页面中找到。

我们还强烈鼓励使用在线讨论与其他学生讨论练习。但是，不要查看其他人编写的任何源代码，也不要与其他人共享源代码。



# 1             正则线性回归

在练习的前半部分，您将使用水库水位的变化来实现正则化线性回归，以预测流出大坝的水量。在下半部分中，您将对调试学习算法进行一些诊断，并检查偏差V.S.方差的影响。

提供的脚本ex5.m将帮助您逐步完成此练习。

## 1.1           可视化数据集

我们将从可视化数据集开始，该数据集包含水位变化的历史记录x和流出大坝的水量y。

此数据集分为三部分：

•您的模型将学习的训练集：x，y

•用于确定正则化参数的交叉验证集：

西瓦尔

•用于评估性能的测试集。这些是您的模型在培训过程中没有看到的“看不见的”示例：xtest，ytest

ex5.m的下一步将绘制培训数据（图1）。在下面的部分中，您将实现线性回归，并使用它来拟合数据的直线并绘制学习曲线。然后，您将实现多项式回归，以找到更好的数据拟合。



图1：数据

## 1.2           正则化线性回归成本函数

回想一下，正则化线性回归具有以下成本函数：

*,*

其中，λ是控制正则化程度的正则化参数（从而有助于防止过度拟合）。正则化项对总成本j施加了惩罚，随着模型参数θj的增加，惩罚也随之增加。注意，你不应该调整θ0项。（在八度/matlab中，θ0项表示为theta（1），因为八度/matlab中的索引从1开始）。

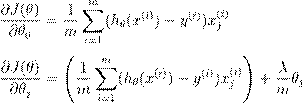
现在您应该完成linearregostfunction.m文件中的代码。您的任务是编写一个函数来计算正则化线性回归成本函数。如果可能，尝试向量化代码并避免编写循环。完成后，ex5.m的下一部分将使用在[1；1]处初始化的theta运行成本函数。您应该看到

303.993.

*现在您应该提交解决方案。*

## 1.3           正则化线性回归梯度

相应地，θj正则化线性回归成本的偏导数定义为：

j＝0

j＝1

在linearregostfunction.m中，添加代码以计算渐变，并返回变量渐变。完成后，ex5.m的下一部分将使用在[1；1]处初始化的theta运行渐变函数。你应该看到一个坡度为[15.30；598.250]。

*现在您应该提交解决方案。*

## 1.4           拟合线性回归

一旦您的成本函数和梯度工作正常，ex5.m的下一部分将运行trainlinereg.m中的代码来计算θ的最佳值。此培训功能使用快速消费品来优化成本功能。

在这部分中，我们将正则化参数λ设置为零。因为我们目前的线性回归的实现是试图拟合二维θ，正则化对于如此低维的θ不会有难以置信的帮助。在练习的后面部分，您将使用具有正则化的多项式回归。

最后，ex5.m脚本还应该绘制出最适合的行，从而生成类似于图2的图像。最佳拟合线告诉我们，模型不适合数据，因为数据具有非线性模式。虽然如图所示可视化最佳匹配是调试学习算法的一种可能方法，但是可视化数据和模型并不总是容易的。在下一节中，您将实现一个函数来生成学习曲线，它可以帮助您调试学习算法，即使数据不容易可视化。

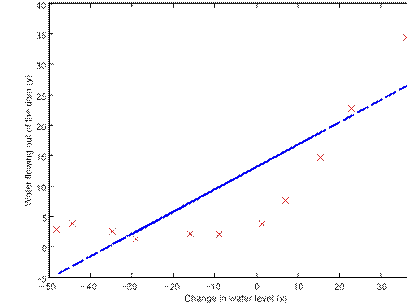


图2：线性拟合

# 2             拜厄斯方差

机器学习中的一个重要概念是偏方差权衡。具有高偏差的模型对于数据来说不够复杂，容易出现下溢现象，而具有高方差的模型则与训练数据过度匹配。

在本部分练习中，您将在学习曲线上绘制培训和测试错误，以诊断偏差方差问题。

## 2.1           学习曲线

现在，您将实现代码来生成在调试学习算法时有用的学习曲线。回想一下，学习曲线将训练和交叉验证错误作为训练集大小的函数绘制出来。你的工作是填写learningcurve.m，以便它返回训练集和交叉验证集的错误向量。

为了绘制学习曲线，我们需要针对不同的训练集大小设置一个训练和交叉验证集误差。要获得不同的训练集大小，应使用原始训练集x的不同子集。具体来说，对于训练集大小为i的情况，应使用第一个i示例（即x（1:i，：）和y（1:i））。

可以使用trainlinereg函数查找θ参数。注意lambda作为参数传递给learningcurve函数。学习θ参数后，应该计算训练集和交叉验证集上的错误。回想一下，数据集的训练错误定义为

*.*

特别注意，培训错误不包括正规化术语。计算培训误差的一种方法是使用现有的成本函数，仅当使用它计算培训误差和交叉验证误差时，才将λ设置为0。在计算训练集错误时，请确保在训练子集（即x（1:n，：）和y（1:n））上计算它（而不是整个训练集）。但是，对于交叉验证错误，应该在整个交叉验证集上计算它。您应该将计算的错误存储在向量错误序列和错误值中。

完成后，ex5.m将打印学习曲线并生成类似于图3的绘图。

*现在您应该提交解决方案。*

在图3中，您可以观察到当培训示例的数量增加时，列车错误和交叉验证错误都很高。这反映了模型中的高偏差问题——线性回归模型是

线性回归学习曲线

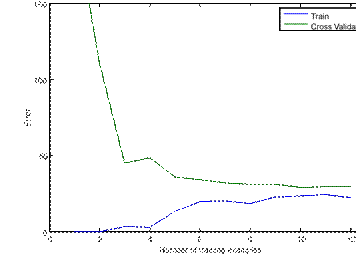


图3：线性回归学习曲线

太简单，无法很好地适应我们的数据集。在下一节中，您将实现多项式回归以适合此数据集的更好的模型。

# 3             多项式回归

我们的线性模型的问题在于，对于数据来说，它太简单了，导致了未匹配（高偏差）。在本部分练习中，您将通过添加更多功能来解决此问题。

对于多项式回归，我们的假设形式如下：

*H时*（x）=θ0+θ1（水位）+θ2（水位）2+·····+θP（水位）P=θ0+θ1x1+θ2x 2+…+TH-PXP。

注意，通过定义x1=（水位），x2=（水位）2，…，xp=（水位）p，我们得到一个线性回归模型，其中特征是原始值（水位）的各种幂次。

现在，您将使用数据集中现有功能X的更高幂次添加更多功能。您在本部分中的任务是完成polyFeatures.m中的代码，以便函数将M×1大小的原始训练集x映射到其更高的幂。具体地说，当一个M×1大小的训练集x传递到函数中时，函数应该返回一个M×P矩阵x多边形，其中第1列保存X的原始值，第2列保存X.^2的值，第3列保存X.^3的值，依此类推。请注意，您不必在这个函数中解释零ETH功率。

现在您有了一个将功能映射到更高维度的函数，ex5.m的第6部分将把它应用到培训集、测试集和交叉验证集（您还没有使用）。

*现在您应该提交解决方案。*

## 3.1           学习多项式回归

完成polyFeatures.m之后，ex5.m脚本将使用线性回归成本函数来训练多项式回归。

请记住，即使我们的特征向量中有多项式项，我们仍然在解决线性回归优化问题。多项式项已经简单地变成了我们可以用于线性回归的特征。我们使用的成本函数和梯度与您在本练习前面部分中编写的相同。

对于练习的这一部分，您将使用8次多项式。结果表明，如果我们直接对预测数据运行培训，将无法很好地工作，而且特性的比例将很差（例如，具有x=40的示例现在将具有特性x8=408=6.5×1012）。因此，您需要使用特性规范化。

在学习多项式回归的参数θ之前，ex5.m将首先调用FeatureNormalize和Normalize训练集的特性，分别存储mu和sigma参数。我们已经为您实现了这个函数，它与第一个练习中的函数相同。

在学习参数θ之后，您应该看到两个图（图4,5），它们是为λ=0的多项式回归生成的。

从图4中，您应该看到多项式拟合能够很好地跟踪数据点，从而获得一个较低的训练误差。然而，多项式拟合是非常复杂的，甚至会在极值点处下降。这表明多项式回归模型对训练数据的拟合度过高，不能很好地推广。

为了更好地理解非规范化（λ=0）模型的问题，您可以看到学习曲线（图5）在低培训误差低但交叉验证误差高的情况下显示出相同的效果。培训和交叉验证错误之间存在差距，表明存在高方差问题。

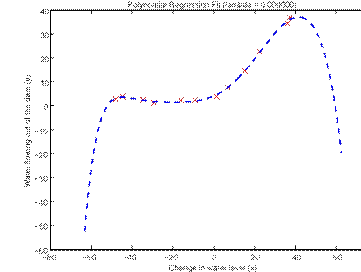


图4：多项式拟合，λ=0

多项式回归学习曲线（lambda=0.000000）

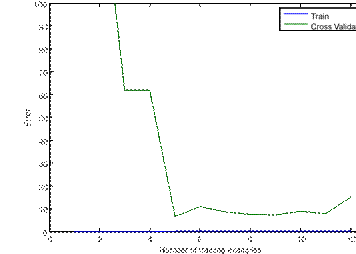


图5：多项式学习曲线，λ=0

克服过拟合（高方差）问题的一种方法是在模型中添加正则化。在下一节中，您将尝试使用不同的λ参数，以了解正则化如何能够得到更好的模型。

## 3.2           可选（未分级）练习：调整正则化参数

在本节中，您将观察正则化参数如何影响正则化多项式回归的偏差方差。现在，您应该修改ex5.m中的lambda参数，并尝试λ=1100。对于这些值中的每一个，脚本都应该生成一个与数据相匹配的多项式，以及一条学习曲线。

对于λ=1，您应该看到一个符合数据趋势的多项式拟合（图6）和一条学习曲线（图7），表明交叉验证和训练误差收敛到一个相对较低的值。这表明λ=1正则多项式回归模型不存在高偏差或高方差问题。实际上，它实现了偏差和方差之间的良好权衡。

对于λ=100，您应该看到多项式拟合（图8）不符合数据。在这种情况下，有太多的正则化，模型无法适应训练数据。

*您不需要为此可选（未分级）练习提交任何解决方案。*

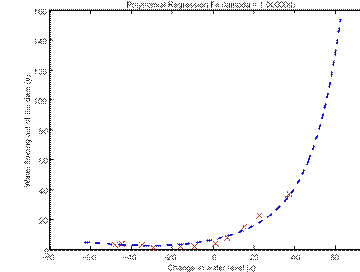


图6：多项式拟合，λ=1

多项式回归学习曲线（lambda=1.000000）

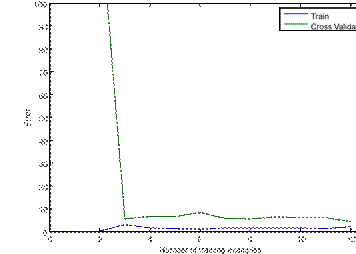


图7：多项式学习曲线，λ=1

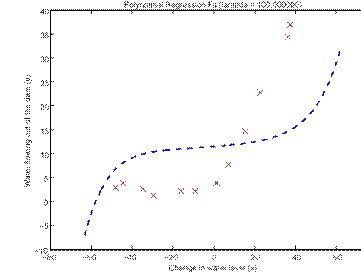


图8：多项式拟合，λ=100

## 3.3           使用交叉验证集选择*拉姆*

从前面的练习部分可以看出，λ的值可以显著影响训练集和交叉验证集上正则多项式回归的结果。特别是，一个没有正则化的模型（λ=0）很好地拟合了训练集，但没有推广。反之，正则化程度太高（λ=100）的模型不适合训练集和测试集。选择好的λ（例如，λ=1）可以很好地拟合数据。

在本节中，您将实现一个自动方法来选择λ参数。具体来说，您将使用交叉验证集来评估每个λ值有多好。在使用交叉验证集选择最佳的λ值后，我们可以在测试集上评估模型，以估计模型在实际未公开数据上的性能。

您的任务是完成validationcurve.m中的代码。具体来说，应该使用trainlinereg函数使用不同的λ值对模型进行培训，并计算培训错误和交叉验证错误。

应在以下范围内尝试λ：0,0.001,0.003,0.01,0.03,0.1,0.3,1,3,10。

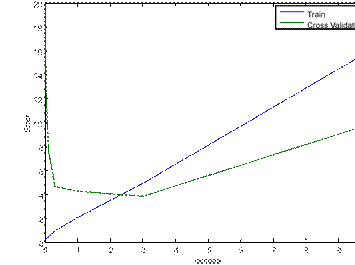


图9：使用交叉验证集选择λ

完成代码后，ex5.m的下一部分将运行您的函数，可以绘制一条错误v.s.λ的交叉验证曲线，允许您选择要使用的λ参数。您应该看到一个类似于图9的图。在这个图中，我们可以看到λ的最佳值在3左右。由于数据集的训练和验证分割的随机性，交叉验证误差有时会低于训练误差。

*现在您应该提交解决方案。*

## 3.4           可选（未分级）练习：计算测试集错误

在练习的前一部分中，您实现了计算正则化参数λ的各种值的交叉验证错误的代码。然而，为了更好地反映模型在现实世界中的性能，重要的是要在一个测试集上评估“最终”模型，该测试集在培训的任何部分都没有使用（也就是说，它既没有用于选择λ参数，也没有用于学习模型参数θ）。

对于此可选（未分级）练习，应使用找到的最佳值λ计算测试误差。在交叉验证中，我们得出了λ=3的测试误差为3.8599。

*您不需要为此可选（未分级）练习提交任何解决方案。*

## 3.5           可选（未分级）练习：用随机选择的示例绘制学习曲线

在实践中，尤其是对于小的训练集，当您绘制学习曲线来调试算法时，通常有助于在多组随机选择的示例中进行平均，以确定训练错误和交叉验证错误。

具体来说，为了确定I示例的训练错误和交叉验证错误，您应该首先从训练集中随机选择I示例，从交叉验证集中随机选择I示例。然后，您将使用随机选择的训练集学习参数θ，并评估随机选择的训练集和交叉验证集上的参数θ。然后，应重复上述步骤多次（例如50），并应使用平均误差来确定训练误差和交叉验证误差。

对于这个可选的（未分级的）练习，您应该实现上面计算学习曲线的策略。作为参考，图10显示了我们为λ=0.01的多项式回归得到的学习曲线。由于示例的随机选择，您的数字可能略有不同。

*您不需要为此可选（未分级）练习提交任何解决方案。*

多项式回归学习曲线（lambda=0.010000）

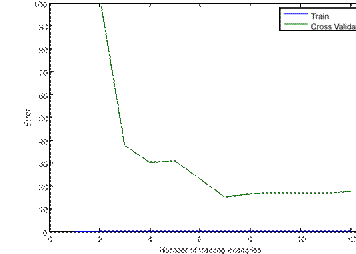


图10：可选（未分级）练习：随机选择示例的学习曲线

# 提交和评分

完成任务的各个部分后，请务必使用提交功能系统将您的解决方案提交给我们的服务器。下面是这个练习的每个部分的评分方式的细目。

|  |  |  |
| --- | --- | --- |
| **部分** | **提交文件** | **点** |
| 正则化线性回归成本  功能 | 线性预测函数.m | 25点 |
| 正则化线性回归梯度 | 线性预测函数.m | 25点 |
| 学习曲线 | 学习曲线 | 20点 |
| 多项式特征映射 | 多特征 | 10点 |
| 交叉验证曲线 | 验证曲线.m | 20点 |
| 总分 |  | 100点 |

您可以多次提交解决方案，我们只考虑最高分数。

[〔1〕](" \l "_ftnref1" \o ")八度是matlab的免费替代品。对于编程练习，您可以自由使用八度或matlab。