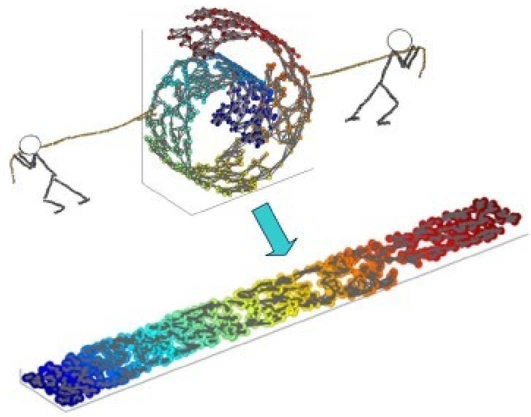
局部线性嵌入(LLE)原理总结

局部线性嵌入(Locally Linear Embedding，以下简称LLE)也是非常重要的降维方法。和传统的PCA，LDA等关注样本方差的降维方法相比，LLE关注于降维时保持样本局部的线性特征，由于LLE在降维时保持了样本的局部特征，它广泛的用于图像图像识别，高维数据可视化等领域。下面我们就对LLE的原理做一个总结。

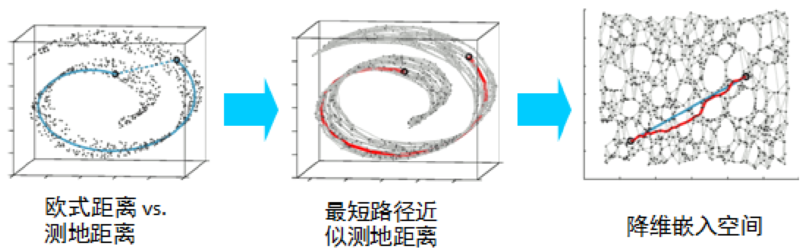
1. 流形学习概述

LLE属于流形学习(Manifold Learning)的一种。因此我们首先看看什么是流形学习。流形学习是一大类基于流形的框架。数学意义上的流形比较抽象，不过我们可以认为LLE中的流形是一个不闭合的曲面。这个流形曲面有数据分布比较均匀，且比较稠密的特征，有点像流水的味道。基于流行的降维算法就是将流形从高维到低维的降维过程，在降维的过程中我们希望流形在高维的一些特征可以得到保留。

一个形象的流形降维过程如下图。我们有一块卷起来的布，我们希望将其展开到一个二维平面，我们希望展开后的布能够在局部保持布结构的特征，其实也就是将其展开的过程，就想两个人将其拉开一样。



在局部保持布结构的特征，或者说数据特征的方法有很多种，不同的保持方法对应不同的流形算法。比如等距映射（ISOMAP）算法在降维后希望保持样本之间的测地距离而不是欧式距离，因为测地距离更能反映样本之间在流形中的真实距离。



但是等距映射算法有一个问题就是他要找所有样本全局的最优解，当数据量很大，样本维度很高时，计算非常的耗时，鉴于这个问题，LLE通过放弃所有样本全局最优的降维，只是通过保证局部最优来降维。同时假设样本集在局部是满足线性关系的，进一步减少的降维的计算量。

2. LLE思想

现在我们来看看LLE的算法思想。

LLE首先假设数据在较小的局部是线性的，也就是说，某一个数据可以由它邻域中的几个样本来线性表示。比如我们有一个样本x_1,我们在它的原始高维邻域里用K-近邻思想找到和它最近的三个样本x_2, x_3, x_4. 然后我们假设x_1可以由x_2, x_3, x_4线性表示，即：

$$x_1 = w_{12}x_2 + w_{13}x_3 + w_{14}x_4$$

其中，w_{12}，w_{13}，w_{14}为权重系数。在我们通过LLE降维后，我们希望x_1在低维空间对应的投影x_1'和x_2, x_3, x_4对应的投影x_2', x_3', x_4'也尽量保持同样的线性关系，即

公告

★珠江追梦，饮岭南茶，恋鄂北家★
昵称：刘建平Pinard
园龄：2年
粉丝：2317
关注：15
+加关注

随笔分类(120)

0040. 数学统计学(4)
0081. 机器学习(69)
0082. 深度学习(11)
0083. 自然语言处理(23)
0084. 强化学习(11)
0121. 大数据挖掘(1)
0122. 大数据平台(1)

随笔档案(120)

2018年10月 (3)
2018年9月 (3)
2018年8月 (4)
2018年7月 (3)
2018年6月 (3)
2018年5月 (3)
2017年8月 (1)
2017年7月 (3)
2017年6月 (8)
2017年5月 (7)
2017年4月 (5)
2017年3月 (10)
2017年2月 (7)
2017年1月 (13)
2016年12月 (17)
2016年11月 (22)
2016年10月 (8)

常去的机器学习网站

52 NLP
Analytics Vidhya
机器学习库
机器学习路线图
强化学习入门书
深度学习进阶书
深度学习入门书

积分与排名

积分 - 351047
排名 - 546

阅读排行榜

$$x'_1 \approx w_{12}x'_2 + w_{13}x'_3 + w_{14}x'_4$$

也就是说，投影前后线性关系的权重系数w_{12}，w_{13}，w_{14}是尽量不变或者最小改变的。

从上面可以看出，线性关系只在样本的附近起作用，离样本远的样本对局部的线性关系没有影响，因此降维的复杂度降低了很多。

下面我们推导LLE算法的过程。

3. LLE算法推导

对于LLE算法，我们首先要确定邻域大小的选择，即我们需要多少个邻域样本来线性表示某个样本。假设这个值为k。我们可以通过和KNN一样的思想通过距离度量比如欧式距离来选择某样本的k个最近邻。

在寻找到某个样本的x_i的k个最近邻之后我们就需要找到找到x_i和这k个最近邻之间的线性关系，也就是要找到线性关系的权重系数。找线性关系，这显然是一个回归问题。假设我们有m个n维样本$\{x_1, x_2, \ldots, x_m\}$,我们可以用均方差作为回归问题的损失函数：即：

$$J(w) = \sum_{i=1}^{m} ||x_i - \sum_{j \in Q(i)} w_{ij}x_j||_2^2$$

其中，$Q(i)$表示i的k个近邻样本集合。一般我们也会对权重系数w_{ij}做归一化的限制，即权重系数需要满足

$$\sum_{j \in Q(i)} w_{ij} = 1$$

对于不在样本x_i邻域内的样本x_j，我们令对应的$w_{ij} = 0$

也就是我们需要通过上面两个式子求出我们的权重系数。一般我们可以通过矩阵和拉格朗日子乘法来求解这个最优化问题。

对于第一个式子，我们先将其矩阵化：

$$\begin{aligned} J(W) &= \sum_{i=1}^{m} ||x_i - \sum_{j \in Q(i)} w_{ij}x_j||_2^2 &(1)\\ &= \sum_{i=1}^{m} ||\sum_{j \in Q(i)} w_{ij}x_i - \sum_{j \in Q(i)} w_{ij}x_j||_2^2 &(2)\\ &= \sum_{i=1}^{m} ||\sum_{j \in Q(i)} w_{ij}(x_i - x_j)||_2^2 &(3)\\ &= \sum_{i=1}^{m} W_i^T(x_i-x_j)(x_i-x_j)^TW_i &(4) \end{aligned}$$

其中$W_i = (w_{i1}, w_{i2}, \ldots w_{ik})^T$。

我们令矩阵$Z_i = (x_i-x_j)(x_i-x_j)^T, j \in Q(i)$,则第一个式子进一步简化为$J(W) = \sum\limits_{i=1}^{m} W_i^TZ_iW_i$.对于第二个式子，我们可以矩阵化为：

$$\sum_{j \in Q(i)} w_{ij} = W_i^T1_k = 1$$

其中1_k为k维全1向量。

现在我们将矩阵化的两个式子用拉格朗日子乘法合为一个优化目标：

$$L(W) = \sum_{i=1}^{m} W_i^TZ_iW_i + \lambda(W_i^T1_k - 1)$$

对W求导并令其值为0，我们得到

$$2Z_iW_i + \lambda 1_k = 0$$

即我们的

$$W_i = \lambda' Z_i^{-1}1_k$$

其中 $\lambda' = -\frac{1}{2}\lambda$为一个常数。利用 $W_i^T1_k = 1$,对W_i归一化，那么最终我们的权重系数W_i为：

$$W_i = \frac{Z_i^{-1}1_k}{1_k^TZ_i^{-1}1_k}$$

现在我们得到了高维的权重系数，那么我们希望这些权重系数对应的线性关系在降维后的低维一样得到保持。假设我们的n维样本集$\{x_1, x_2, \ldots, x_m\}$在低维的d维度对应投影为$\{y_1, y_2, \ldots, y_m\}$, 则我们希望保持线性关系，也就是希望对应的均方差损失函数最小，即最小化损失函数$J(Y)$如下：

$$J(y) = \sum_{i=1}^{m} ||y_i - \sum_{j=1}^{k} w_{ij} y_j||_2^2$$

可以看到这个式子和我们在高维的损失函数几乎相同，唯一的区别是高维的式子中，高维数据已知，目标是求最小值对应的权重系数W,而我们在低维是权重系数W已知，求对应的低维数据。

为了得到标准化的低维数据，一般我们也会加入约束条件如下：

$$\sum_{i=1}^{m} y_i = 0; \;\; \frac{1}{m}\sum_{i=1}^{m} y_i y_i^T = I$$

首先我们将目标损失函数矩阵化：

$$\begin{aligned} J(Y) &= \sum_{i=1}^{m} ||y_i - \sum_{j=1}^{k} w_{ij} y_j||_2^2 & (5) \\ &= \sum_{i=1}^{m} ||YI_i - YW_i||_2^2 & (6) \\ &= tr(Y^T (I-W)^T (I-W) Y) & (7) \end{aligned}$$

如果我们令$M = (I-W)^T (I-W)$,则优化函数转变为最小化下式：$J(Y) = tr(Y^T M Y)$,tr为迹函数。约束函数矩阵化为：$Y^T Y = mI$

如果大家熟悉谱聚类和PCA的优化，就会发现这里的优化过程几乎一样。其实最小化J(Y)对应的Y就是M的最小的d个特征值所对应的d个特征向量组成的矩阵。当然我们也可以通过拉格朗日函数来得到这个：

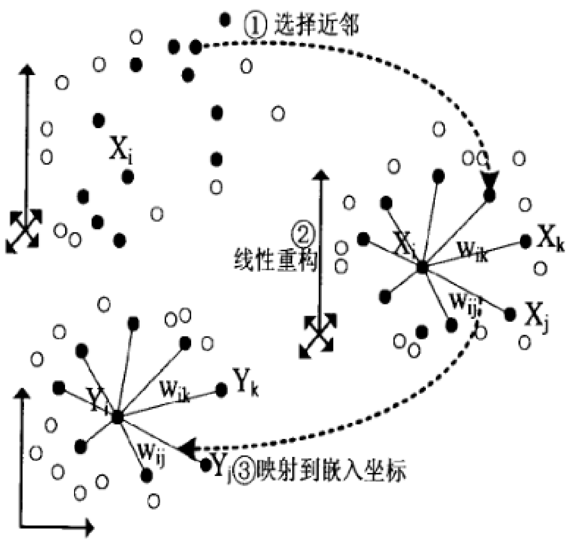
$$L(Y) = tr(Y^T M Y) + \lambda (Y^T Y - mI)$$

对Y求导并令其为0，我们得到$2MY + 2\lambda Y = 0$,即$MY = \lambda Y$,这样我们就很清楚了，要得到最小的d维数据集，我们需要求出矩阵M最小的d个特征值所对应的d个特征向量组成的矩阵$Y = (y_1, y_2, ... y_d)$即可。

一般的，由于M的最小特征值为0不能反应数据特征，此时对应的特征向量为全1。我们通常选择M的第2个到第d+1个最小的特征值对应的特征向量$M = (y_2, y_3, ... y_{d+1})$来得到最终的Y。为什么M的最小特征值为0呢？这是因为$W^T e = e$,得到$|W^T - I|e = 0$,由于$e \neq 0$,所以只有$W^T - I = 0$,即 $(I-W)^T = 0$,两边同时右乘$I - W$，即可得到$(I-W)^T (I-W)e = 0e$,即M的最小特征值为0.

4. LLE算法流程

在上一节我们已经基本推导了LLE降维的整个流程，现在我们对算法过程做一个总结。整个LLE算法用一张图可以表示如下：



从图中可以看出，LLE算法主要分为三步，第一步是求K近邻的过程，这个过程使用了和KNN算法一样的求最近邻的方法。第二步，就是对每个样本求它在邻域里的K个近邻的线性关系，得到线性关系权重系数W，具体过程在第三节第一部分。第三步就是利用权重系数来在低维里重构样本数据，具体过程在第三节第二部分。

具体过程如下：

输入：样本集$D = \{x_1, x_2, ..., x_m\}$, 最近邻数k，降维到的维数d

输出： 低维样本集矩阵D'

1) for i 1 to m，按欧式距离作为度量，计算和x_i最近的的k个最近邻$(x_{i1}, x_{i2}, ..., x_{ik},)$

2) for i 1 to m, 求出局部协方差矩阵$Z_i = (x_i - x_j)^T (x_i - x_j)$,并求出对应的权重系数向量：

$$W_i = \frac{Z_i^{-1}1_k}{1_k^TZ_i^{-1}1_k}$$

3) 由权重系数向量W_i组成权重系数矩阵W,计算矩阵$M = (I-W)^T(I-W)$

4) 计算矩阵M的前d+1个特征值，并计算这d+1个特征值对应的特征向量$\{y_1, y_2, \ldots y_{d+1}\}$。

5）由第二个特征向量到第d+1个特征向量所张成的矩阵即为输出低维样本集矩阵 $D' = (y_2, y_3, \ldots y_{d+1})$

5. LLE的一些改进算法

LLE算法很简单高效，但是却有一些问题，比如如果近邻数k大于输入数据的维度时，我们的权重系数矩阵不是满秩的。为了解决这样类似的问题，有一些LLE的变种产生出来。比如：Modified Locally Linear Embedding(MLLE)和Hessian Based LLE(HLLE)。对于HLLE，它不是考虑保持局部的线性关系，而是保持局部的Hessian矩阵的二次型的关系。而对于MLLE，它对搜索到的最近邻的权重进行了度量，我们一般都是找距离最近的k个最近邻就可以了，而MLLE在找距离最近的k个最近邻的同时要考虑近邻的分布权重，它希望找到的近邻的分布权重尽量在样本的各个方向，而不是集中在一侧。

另一个比较好的LLE的变种是Local tangent space alignment(LTSA)，它希望保持数据集局部的几何关系，在降维后希望局部的几何关系得以保持，同时利用了局部几何到整体性质过渡的技巧。

这些算法原理都是基于LLE，基本都是在LLE这三步过程中寻求优化的方法。具体这里就不多讲了。

6. LLE总结

LLE是广泛使用的图形图像降维方法，它实现简单，但是对数据的流形分布特征有严格的要求。比如不能是闭合流形，不能是稀疏的数据集，不能是分布不均匀的数据集等等，这限制了它的应用。下面总结下LLE算法的优缺点。

LLE算法的主要优点有：

1）可以学习任意维的局部线性的低维流形

2）算法归结为稀疏矩阵特征分解，计算复杂度相对较小，实现容易。

LLE算法的主要缺点有：

1）算法所学习的流形只能是不闭合的，且样本集是稠密均匀的。

2）算法对最近邻样本数的选择敏感，不同的最近邻数对最后的降维结果有很大影响。

（欢迎转载，转载请注明出处。欢迎沟通交流： liujianping-ok@163.com）

评论列表

#1楼 2017-01-10 12:39 旷视科技/face

广泛的用于图像图像识别，高维数据可视化等领域

支持(0) 反对(0)

#2楼 2017-03-21 20:12 小马过_河

刘老师你好！看了你的几篇博文 ，很是钦佩！学生我也在学习流形降维之类的知识，对LTSA很感兴趣，苦于没有代码，请问您可否将代码分享于我？不胜感激！625774790@qq.com这是我的邮箱，我看您对流形学习了解颇深，还请问老师您从哪获得的资料，我也想学习一下！

支持(0) 反对(0)

#3楼[楼主] 2017-03-21 21:27 刘建平Pinard

@ 小马过_河
你好，LTSA我没有去做过实现，也没有在实际项目中用过，惭愧。如果你需要源代码研究。建议参考github的这个实现：
https://github.com/gionuno/local_tangent_space_alignment
不过代码是用MATLAB实现的，你需要熟悉MATLAB。

至于流形学习的资料，主要是从sklearn的流形学习文档，以及对应的参考文献得到的。
http://scikit-learn.org/stable/modules/manifold.html#manifold

支持(0) 反对(0)

#4楼 2017-03-31 17:41 vincentmobile

我看了一下作者的原论文，感觉你的求解好像有点问题。目标函数是局部最优，而您的目标函数是全局最优。

支持(1) 反对(1)

#5楼[楼主] 2017-03-31 18:45 刘建平Pinard

@ vincentmobile
你好，对于某一个点求解是局部优化，但是整个问题求解是全局的。
如果觉得哪儿不妥的话请指出。：）

支持(0) 反对(1)

#6楼 2017-04-10 02:50 snowkylin

您好，从3式到4式进行矩阵化时，随着连加号的去除，连加时使用的j也不再有意义，然而4式中(x_i - x_j)依然有j，可能需要进行一些修改。

支持(0) 反对(0)

#7楼[楼主] 2017-04-10 10:02 刘建平Pinard

@ snowkylin
你好，这儿使用符号j的确可能让人有些疑惑，其实这里的j已经和三式的j没有任何关系了，就是指任意一个样本。应该换一个符号比较好。
感谢你的指正。

支持(0) 反对(0)

#8楼 2017-05-12 17:12 圣殿的御膳房

@ vincentmobile
我也觉得有问题，求解权值不应该是全局的，思想就是用局部线性，最后才是全局优化

支持(0) 反对(0)

#9楼 2017-06-12 16:23 666666coder

楼主，我觉得你上面的推导过程有问题，你在矩阵化的过程中建设Wi为行向量，但是后面都是按照列向量推导的

支持(0) 反对(0)

#10楼[楼主] 2017-06-15 12:49 刘建平Pinard

@ 666666coder
你好，感谢指出错误，这里本意是W_i为列向量，已经修改原文。

支持(0) 反对(0)

#11楼 2017-07-12 21:36 xx132

老师，您好!
第一个损失函数Σ||xi-Σwijxj||2，j=1,...k,(这里输公式不太方便，有点乱，希望不要介意)，减号后面用的是xj的加权，这个地方容易误解为所有的样本xi，i=1,...,m的领域样本都为x1,...,xk，不知我理解的对不对

支持(0) 反对(0)

#12楼[楼主] 2017-07-12 21:54 刘建平Pinard

@ xx132
你好，你的理解是对的，这儿的意思是每个样本x_i自己的k个最邻近样本。但是由于我用的是x_j这样的表示，容易被误解为所有的样本共有这k个最近邻样本。
可能这样表示较好：

$$J(w) = \sum_{i=1}^{m} ||x_i - \sum_{j \in Q(i)}^{k} w_{ij}x_j||_2^2$$

其中$Q(i)$是x_i的k个最近邻样本。

支持(1) 反对(0)

#13楼 2018-03-20 11:09 luoge

想问下您，如何能够确定是不是封闭的曲面呢？样本集不是稠密均匀的，结果又会如何呢？

支持(0) 反对(0)

#14楼[楼主] 2018-03-20 11:17 刘建平Pinard

@ luoge
你好，由于大部分数据是高维的，所以这个判断一般只能凭经验了。
如果不是稠密均匀的话，那么LLE降维效果就很差了

支持(0) 反对(0)

#15楼 2018-04-17 22:30 DsHale

> 引用
> LLE算法很简单高效，但是却有一些问题，比如如果近邻数k大于低维的维度d时，我们的权重系数矩阵不是满秩的。

能解释下为什么不是满秩的吗?

支持(0) 反对(0)

#16楼[楼主] 2018-04-17 23:36 刘建平Pinard

@ DsHale
你好，其实就是由于维度太低，我们的线性关系不能完全表达。因为我们的线性关系需要k个标量组成的向量对应的权重系数矩表达，但是由于维度d小于k，则部分线性关系表达不出来，此时权重系数矩阵秩最大为d。而原重系数矩阵为mxk，故不满秩。

支持(1) 反对(0)

#17楼 2018-04-18 15:37 DsHale

@ 刘建平Pinard
感谢老师能耐心解答.
看了您的博客后,我又去读了一下相关论文(https://cs.nyu.edu/~roweis/lle/papers/lleintro.pdf) 里面附录A中提到:
If the covariance matrix is singular or nearly singular —as arises, for example, when there are more neighbors than input dimensions (K>D), or when the data points are not in general position—it can be conditioned (before solving the system) by adding a small multiple of the identity matrix.
他这里说道的是如果K>D,原文中的D指的是高维度数,在您的博客中是低维度数d.在此有些疑惑.
当然您也提到不能线性表示证明不满秩,我想从矩阵秩的角度证明一下,不知道正确与否.我们求解每个Wij的时候需要计算每个点与它K个最邻近的节点所组成的协方差矩阵Zi的逆.我们知道Zi = (xi - xj)' *(xi-xj)
xi - xj 是 D*K的,那么Zi 是K*K维的
如果K>D 那么r(xi-xj) <D<K的 ,线性代数中有一个关于秩的等式r(AB)<min(r(A),r(B)) 的 所以此时的R(Zi) < K的肯定不是满秩的所以在求解过程中会遇到问题.原文中提到需要在这个矩阵上面来加上一些很小的数来获得正确的计算.
这是我关于协方差矩阵秩的思考.

支持(1) 反对(0)

#18楼[楼主] 2018-04-18 21:58 刘建平Pinard

@ DsHale
你好，我又去看了一下资料，这里权重系数矩阵是否满秩只与输入维度有关，与降维到的维度无关，这里我写的时候有些想当然了。没有深入理解，原文已经修改。而上面的我的回复里面的d应该也是要是输入维度才是对的。

sklearn文档里面也这么说了:
One well-known issue with LLE is the regularization problem. When the number of neighbors is greater than the number of input dimensions, the matrix defining each local neighborhood is rank-deficient.

http://scikit-learn.org/stable/modules/manifold.html#locally-linear-embedding

支持(0) 反对(0)

#19楼 2018-04-18 22:55 DsHale

@ 刘建平Pinard
感谢您的回复,也要感谢您的博客.学习了很多.

支持(0) 反对(0)

#20楼 2018-05-24 08:20 PingLee

刘老师你好，我最近一直在看您的文章，写的真好，矩阵表示似乎把所有问题都简单化了，更容易处理了，谢谢您的付出。这里我有一个疑问，在公式5到6时，步骤能否再写的详细点呢？公式6的两个Y老觉得哪里不对，写成Y表示的应该是降维后的所有样本构成的矩阵，应该是d*m的吧？而式中第二个Y不应该是第i个样本的k个邻近样本吗，应该是d*k吧？它后面乘的是Wi，Wi是k*1的吧

支持(1) 反对(0)

#21楼[楼主] 2018-05-24 22:49 刘建平Pinard

@ PingLee
这一篇的数学表述有些不严谨。参看12楼的回复。

其实这里的w是要所有的样本一起考虑的，而不是k个，也就是维度不是k，而是样本数m。只能说其中的k个是有权重值的，其余的都是0.这样才能做全局的优化求解，后面有时间我会改一下。

支持(0) 反对(0)

#22楼 2018-05-25 11:21 PingLee

好的，多谢刘老师的回复

支持(0) 反对(0)

#23楼 2018-07-10 20:16 AndyPang

博主你好，感谢精彩分享，第3节有个地方没看明白：
第3节用拉格朗日子乘法合一之后的L(W)求导之后，怎么前面求和符号不用了？

支持(0) 反对(0)

#24楼[楼主] 2018-07-12 17:50 刘建平Pinard

@ AndyPang
你好，这里是针对W的某一个维度W_i求导的，所以对于不是i的求和部分求导就为0了。

支持(0) 反对(0)

#25楼 2018-08-28 21:55 Djongo

老师你好，请问你推出M的最小特征值为0，因为W^Te =e，是怎么来的啊。

支持(0) 反对(0)

#26楼[楼主] 2018-08-29 10:49 刘建平Pinard

@ Djongo
你好，这是因为W矩阵的每一行之和为1(权重和为1).所以自然有$W^Te = e$

支持(1) 反对(0)

#27楼 2018-09-02 21:02 kylin0228

博主您好，
有个疑问一直没弄明白：
xi和xj都是n*1维度的
为什么（xi-xj)是n*k维度的？
也就是为什么Z为K*K维
谢谢博主

支持(0) 反对(0)

#28楼 2018-09-03 09:14 kylin0228

博主你好
对于第六个公式
Y是d'乘m维的
Wi是k乘1维的
所以最后一个YWi乘起来维度不对吧？

支持(0) 反对(0)

#29楼[楼主] 2018-09-03 10:18 刘建平Pinard

@ kylin0228
你好，这篇文章的推导过程有些问题，正确的做法是W都是nxn维的而不是nxk维的。因为我们要找到全局的局部线性损失。有权重值的k个W的位置大于0，没有权重值的剩下的W的位置等于0。

后面有机会我会把这篇文章的推导过程重新改写一下。当然原理部分还是没有问题的。我21楼也说到这个问题。

支持(2) 反对(0)

#30楼 2018-09-22 20:59 不会矩阵化

刘老师，您好，请问一下，第三节的，通过求导把w算出来后，然后到归一化是怎么计算的，有点不明白，能否解释一下？

支持(0) 反对(0)

#31楼 2018-09-24 12:12 不会矩阵化

@
老师，这个地方我明白了，但是我觉得分母不是应该是分子的范数吗？

支持(0) 反对(0)

#32楼[楼主] 2018-09-25 10:37 刘建平Pinard

@ 不会矩阵化
你好，这里用的并不是L1/L2归一化，只是为了让W_i的值唯一而已。同时去除拉格朗日因子的影响。

支持(0) 反对(0)

#33楼 2018-09-25 10:42 不会矩阵化

@
这个地方我是这样理解的，w的大小确定的，但是方向不确定，所以分子那个是确定方向的？

支持(0) 反对(0)

#34楼[楼主] 2018-09-25 11:38 刘建平Pinard

@ 不会矩阵化
你好，你说的方向不确定我觉得有点怪，因此是方向已经确定了，但是大小不确定。之前是有一个拉格朗日系数的，所以大小不定，但是消除后，大小就确定了。毕竟Z_i^{-1}的值是确定的。

支持(0) 反对(0)

#35楼 2018-09-25 13:54 不会矩阵化

@
但是z不一定可逆呀

支持(0) 反对(0)

#36楼[楼主] 2018-09-26 13:57 刘建平Pinard

@ 不会矩阵化
你好，如果Z_i^{-1}不可逆，那么权重系数向量不唯一，可以有多个值和方向选择。就算你归一化了也会不确定。

支持(0) 反对(0)

#37楼 2018-09-26 18:56 不会矩阵化

@ 刘建平Pinard
好的，非常感谢

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 登录 或 注册，访问网站首页。

最新IT新闻:
· 永别了Google+：谷歌社交梦碎背后的故事
· "两面"王冉：我的投资哲学永远有三个假设
· 飞猪宣布成立新旅行联盟 阿里CEO张勇表示没有兴趣做OTA
· 不要和贾跃亭说话
· 谷歌不作恶，亚马逊、微软员工：我们也不要作
» 更多新闻...

最新知识库文章:
· 阿里云的这群疯子
· 为什么说 Java 程序员必须掌握 Spring Boot ?
· 在学习中，有一个比掌握知识更重要的能力
· 如何招到一个靠谱的程序员
· 一个故事看懂"区块链"
» 更多知识库文章...