FM理论与实践

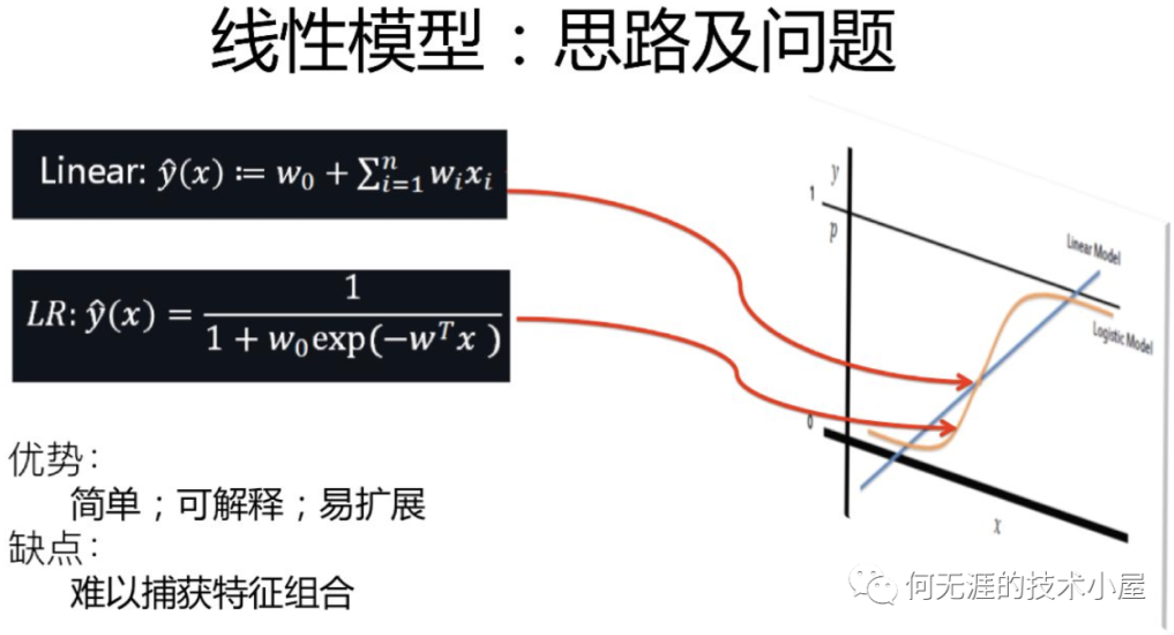
# 背景

FM算法（Factor Machine），又叫因子分解机算法。在推荐系统和点击率预估（CTR预估）中，FM算法有很广泛的应用。这两个场景的实质都是根据所提供的一些特征信息来判断用户是否会有点击行为，或者说点击的概率。在推荐系统和CTR预估任务中，通常把LR作为baseline。

如果直接利用所提供的特征信息，线性模型将是最简单直接的方法。如下图所示，xi 就是某个特征值，线性模型需要为每一个特征值学习一个权重wi，最终的模型预测值就是所有的特征值乘以这个权重，加起来求和。 公式如下：



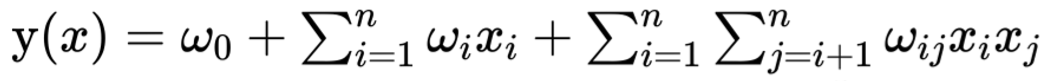
如果是逻辑回归（LR），需要在上面求和的基础上套一个sigmoid函数，也就是图中的黄色曲线。直观上很好理解，使用sigmoid将线性模型的取值范围压缩到0-1，这 样就可以很容易的判断是正面的结果还是负面的结果（对应着点击与不点击）。



在使用LR解决推荐和CTR预估问题中，常常对数据进行大量的特征工程，构造出大量的单特征，编码后送入LR模型，从而预估点击率。这种线性模型的优势在于，运算速度快，可解释性强，在特征挖掘完备且训练数据充分的前提下，能够达到一定的精度。但线性模型的缺点也很多，如：

1. 模型并未考虑到特征之间的组合关系。而特征交叉组合在推荐系统和CTR任务中往往能够更好地提升模型效果。
2. 对于类别型（categorical）特征进行one-hot编码，具有高度稀疏性，一方面会带来维度灾难问题，另一方面会由于特征高度稀疏，导致模型无法充分利用这些稀疏特征。

要使用特征两两交叉组合，最直观的做法就是直接引入两两组合特征融入到模型中，在线性模型的基础上得到，可以使用下面的公式表示：

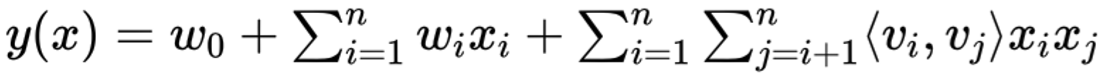


然而这样有两个比较大的问题，第一，这样处理组合特征的参数个数为n(n-1)/2，如果n=1000（对原始数据连续特征离散化，并且one-hot编码，特征很容易达到此量级），那么参数个数将达到50w，参数量巨大；第二，由于对原始数据的one-hot处理，数据将特别稀疏，满足xi，xj都不等于0的个数很少，这样就很难去学习到参数wij，导致组合特征的泛化能力差（详细解释：由于xi和xj大部分为0，所以xi\*xj也会大部分为0，假设X=xi\*xj，则，又因为X=0，所以，梯度为0，参数无法更新）。

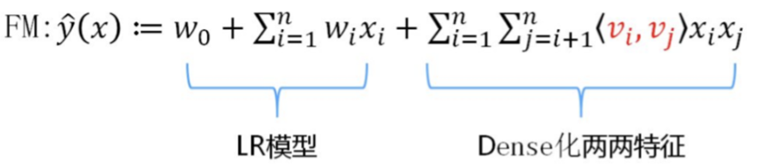
导致这种情况出现的根源在于：特征过于稀疏。需要找到一种方法，使得wij的求解不受特征稀疏性的影响。

# FM算法

FM以特征组合为切入点，在线性模型的基础上引入特征交叉项，弥补了一般线性模型未考虑特征间关系的缺憾。FM（Factorization Machine）模型对每个特征学习一个隐向量，由两个隐向量的内积表示两个特征交叉的权重，公式如下：



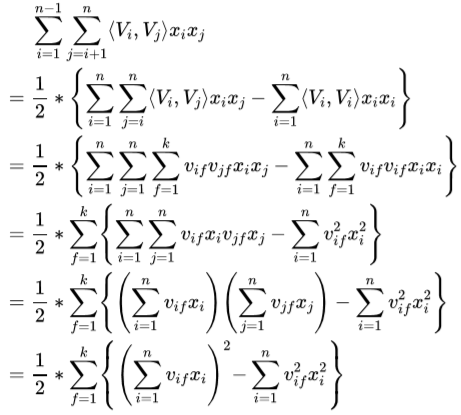
FM算法的本质是利用近似矩阵分解，将参数权重矩阵W分解成两个向量相乘，从而将参数从平方级别减少到线性级别。



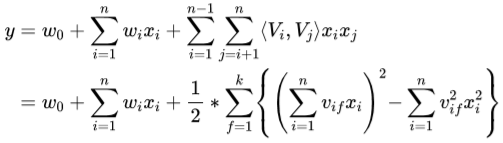


vi和vj分别是对于xi这个特征来说它会学到一个embedding向量，特征组合权重是通过两个单特征各自的embedding的内积呈现的，因为它内积就是个数值，可以代表它的权重，这就是FM算法。

FM算法最核心的就是后面这个交叉项，为了计算方便，对该交叉项进行化简得到：



按照这种化简之后，FM的最终形式如下：



这样化简之后，计算就会简单多了，改写前，计算y的复杂度为，改写后计算y的复杂度变为，大大提高了模型预测的速度。

FM算法自被提出以来，在推荐系统、CTR和广告等相关领域取得了巨大突破，由于FM优越的性能表现，后续出现了一系列FM变种模型（如FFM、FNN、AFM、DeepFM、xDeepFM等），从浅层模型到深度推荐模型中都有FM的影子。

# FM求解

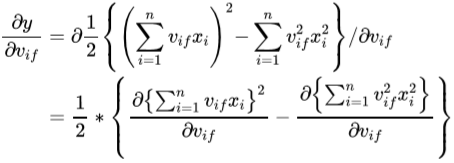
这里阐述如何求解FM的模型参数。

求解FM参数的过程可以采用梯度下降法，为了对参数进行梯度下降更新，需要计算模型各参数的梯度表达式。

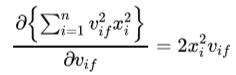
当参数是w0时，

当参数是wi时，

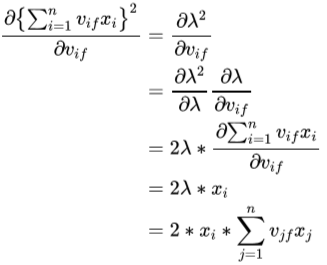
当参数为vif时，只需要关注模型的高阶项，此时，其余无关的参数可以看作常数：



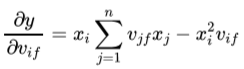
其中：



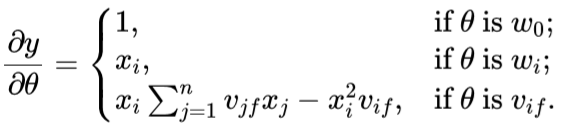
令，则：



因此可以得到：



综上，最终模型各参数的梯度表达式如下：



# 性能分析

FM进行推断的时间复杂度为O(kn)。依据参数的梯度表达式，与i无关，在参数更新时可以首先将所有的计算出来，复杂度为O(kn)，后续更新所有参数的时间复杂度均为O(1)，参数量为1 + k + kn，所以最终训练的时间复杂度同样为O(kn)，其中n为特征数，k为隐向量维数。

FM训练与预测的时间复杂度均为O(kn)，是一种十分高效的模型。

FM缺点

每个特征只引入了一个隐向量，不同类型特征之间交叉没有区分性。FFM模型正是以这一 点作为切入进行改进。

# 实践

FM既可以应用在回归任务，也可以应用在分类任务。在分类任务中只需在上面的公式最外层套上sigmoid函数即可，上述解析都是基于回归任务来进行推导的。

代码请参考：<https://github.com/jpegbert/code_study/tree/master/FM>

注：

1. 对于回归任务，损失函数可以用MSE，对于分类任务损失函数可以用交叉熵（CrossEntry）
2. 虽然FM可以应用于任意数值类型的数据上，但是需要注意对输入特征数值进行预处理。 优先进行特征归一化，其次再进行样本归一化
3. FM不仅可以用于rank阶段，同时可以用于向量召回

参考：

1. <https://mp.weixin.qq.com/s/FkqsLjpqH66nLg2OoDGAgg>
2. <https://mp.weixin.qq.com/s/hKRGD02LumttFsJt_WyILQ>