**关于学习排名的思考**

美国密歇根大学ECSS统计部2015年1月13日，奥斯丁

**大纲**

**1学习按照有监督的ML功能进行排名标签和输出空间性能度量排名功能**

**2排名方法简介缩减方法提升方法大幅度方法优化方法**

**3学习排名理论概括误差范围一致性**

**4指向高级主题在线学习排名超越相关性：多样性和新鲜度大规模学习排名**

**大纲1**

**学习作为受监督的ML特征进行排名标签和输出空间性能度量排名函数**

网络搜索：PageRank和热门体育：FIFA排名（国际足球）、ICC排名（国际板球）、FIDE排名（国际象棋）、BCS评级（大学足球）学术界：美国新闻与世界报道、《泰晤士报》高等教育民主与发展：投票系统、联合国人类发展指数推荐系统：电影（Netflix、IMDB）、音乐（潘多拉）、消费品（亚马逊)。

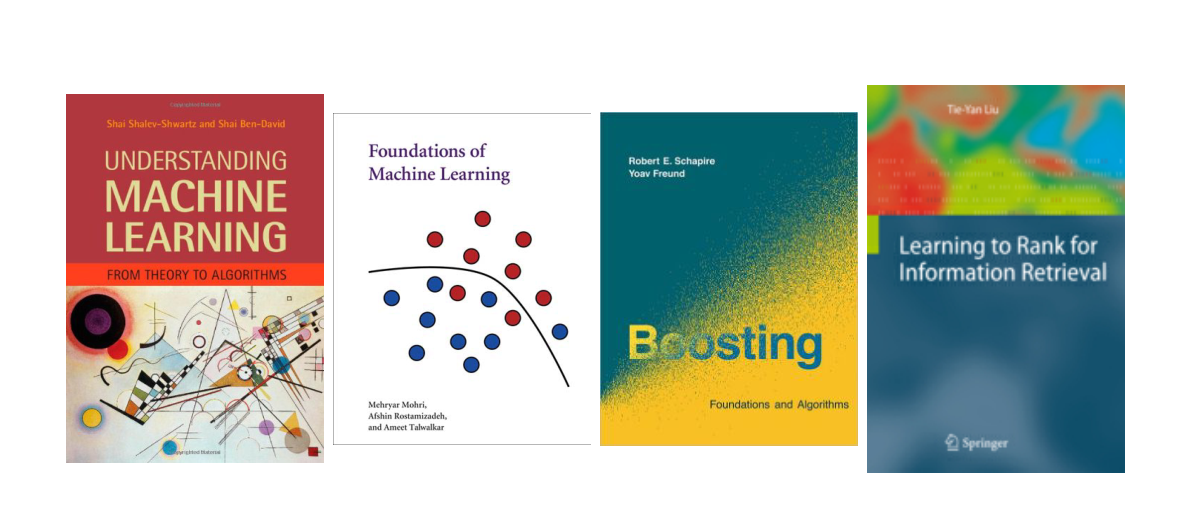
**学习对物体进行排名**

创建一个排名需要很多努力

我们能让计算机学习对物体进行排名吗？

仍然需要一些人的监督，才能将学习列为受监督的ML问题 。

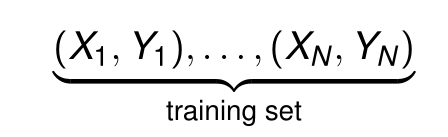
**排名是一个典型的学习问题**

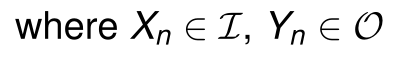


监督机器学习：构建映射



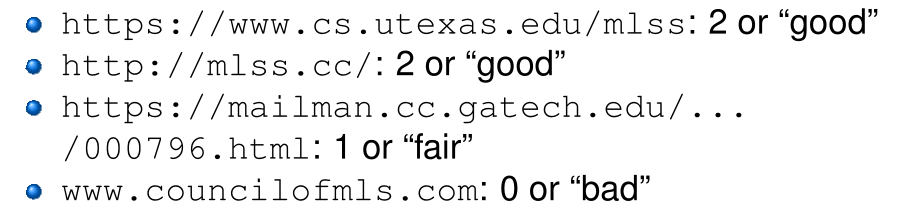
基于输入、输出对（培训示例）





想要f（X）成为排序问题中看不见的X的Y的良好预测因子，O是给定对象集（网页、电影等）的所有可能排序。

查询：“MLSS奥斯汀”候选人网页和相关性分数



思考：示例X=（查询，（URL1，URL2，…，URL4）），标签R=（2，2，1，0）

功能构建：找到一种将（查询，网页）映射到Rp的方法，每个示例（查询，m网页）都映射到Xn∈ I=Rm×p 。

获取标签：获取Rn∈ {0,1,2，…，Rmax}m来自人类的判断Train ranker:Get f:I→ O=Sm，其中Sm是一组m-置换T训练，通常通过解决训练集上的一些优化问题来完成。4评估性能：使用一些性能度量，在“测试集”上评估ranker 。

文档中出现的查询术语数量文档长度总和\/min\/max\/mean\/术语频率方差总和\/min\/max\/mean\/tf idf的方差查询和文档之间的余弦相似性任何模型的P（R=1 | Q，D）都提供了一个特性（例如，BM25）URL长度、Pagerank、站点级Pagerank查询URL单击计数、用户URL单击计数中的斜杠数

对于未来的查询和网页列表，想要对网页进行排名不一定想要预测相关性分数，让f（排名函数）在排名空间中取值，所以现在，训练示例（X，Y）∈ I×L，其中L=待学习的标签空间函数f将I映射到O（注L不等于O）。

1. g. (1, 2, 2, 0, 0) ∈ L and (d1 → 3, d2 → 1, d3 → 2, d4 → 4, d5 → 5) ∈ O 。

**排名符号**

我们会认为m（文档的数量）是固定的，但实际上它因示例而异。

我们将使用排列σ来表示排名∈ Sm（所有m！置换的集合）

σ（i）是文档i

σ的秩\位置−1（j）是排名/第j位的文件。

假设我们有三个文档d1，d2，d3，我们希望它们的排列是这样的：d2，d3，d1，d3:2，这意味着σ（1）=3，σ（2）=1，σ（3）=2，σ−1(1) = 2, σ−1(2) = 3, σ−1(3) = 1 。

**相关性分数的符号/标签**

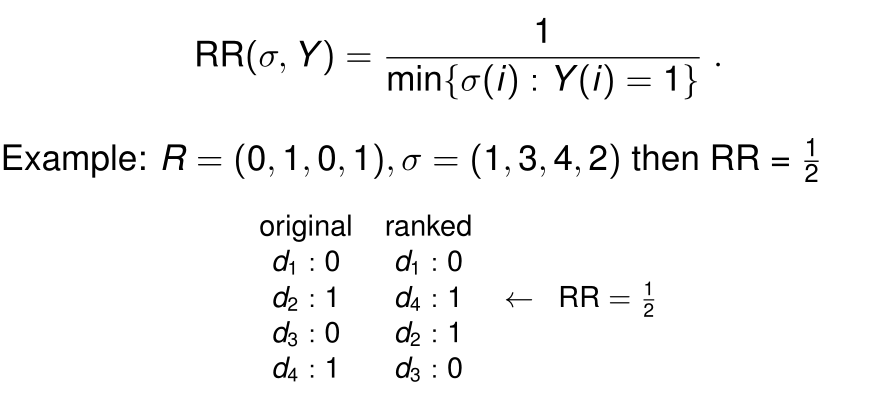
在二进制情况下，相关标签可以是二进制的或多级的，文档在多级情况下是相关的或不相关的（1或0），相关标签通常来自{0,1,2，…，Rmax}T，Rmax≤ 4（最多5级相关性）相关性得分/标签向量R作为m向量R（i）给出文档i的相关性。

**排名中的绩效衡量**

性能指标可以是增益（最大化）或损失（最小化）。它们以相关向量R和排序σ为参数，产生非负增益/损失。一些性能指标仅定义为二元相关性，而其他性能指标可以处理更一般的多级相关性。

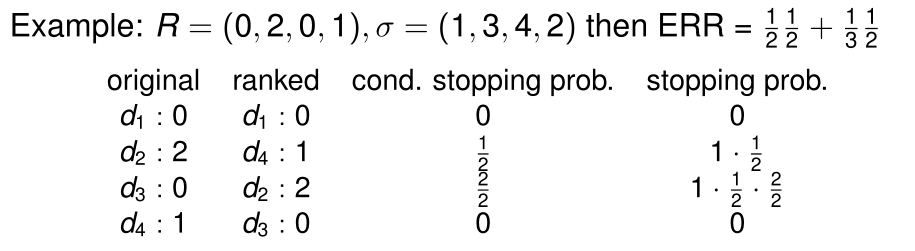
**绩效指标：RR**

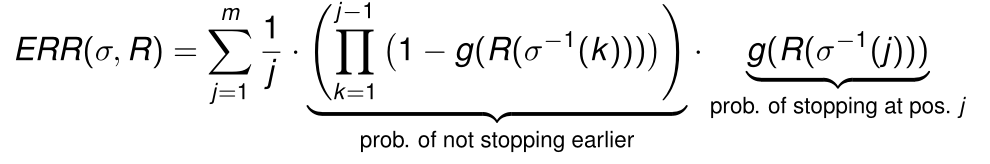
互易等级（RR）是为二元相关性定义的，只有RR是根据等级的第一个相关文档的互易等级。



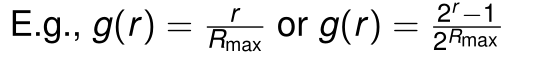
期望倒数排名（ERR）将RR概括为多级相关性将相关性得分转换为概率（例如，除以Rmax）想象一个用户在排名列表中，以相关性得分给出的概率停止。ERR是用户停止的文档的期望倒数排名。

**绩效衡量标准：ERR**



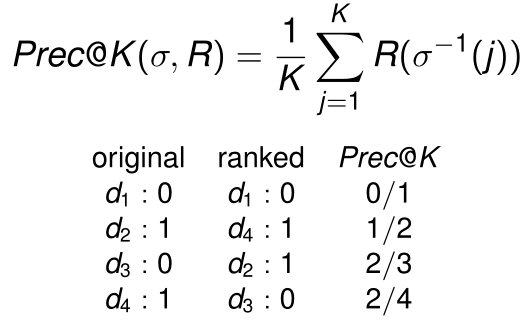


回归：R（σ）−1（j））是位置j处文档的相关性g是用于将相关性得分转换为停止概率的任何函数。



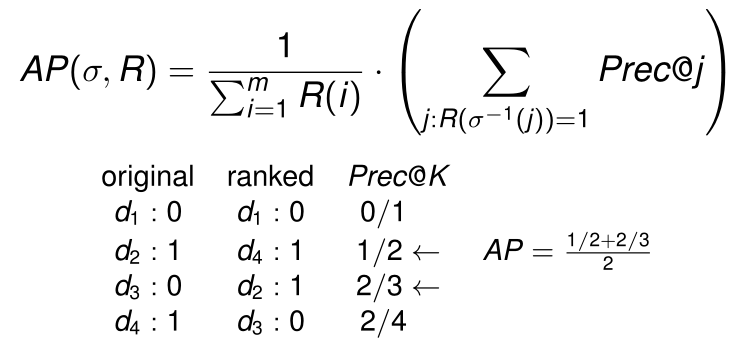
Precision@K仅用于二进制关联

Precision@K是top-k中相关文档的分数



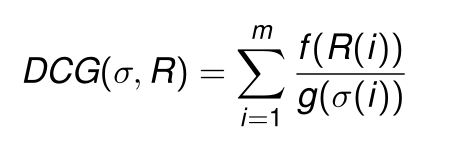
**绩效指标：AP**

平均精度（AP）也适用于二进制相关性，它是Precision@K仅限于相关文件的位置。



**绩效指标：DCG**

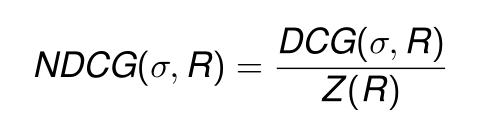
折扣累积收益（DCG）是指相关文档的多级相关性贡献，排名越低，折扣越大。



f，g是递增函数

标准选择：f（r）=2r− 1和g（j）=log2（1+j）。

规范化DCG（NDCG）和DCG一样，顾名思义，NDCG用于多级相关性，NDCG对DCG进行规范化，使增益保持在1的范围内。

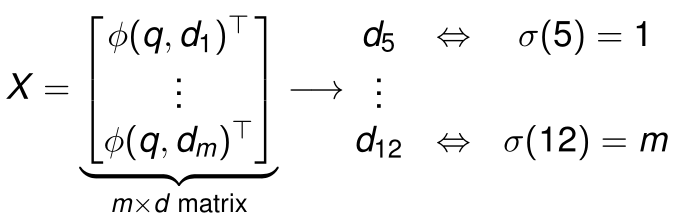


归一化常数

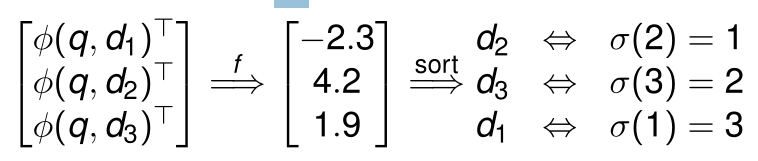


可以通过对相关向量排序快速计算

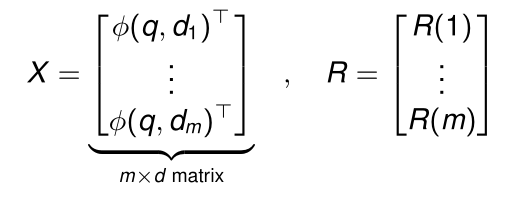
排序函数将m个查询文档的p维特征映射为一个排列



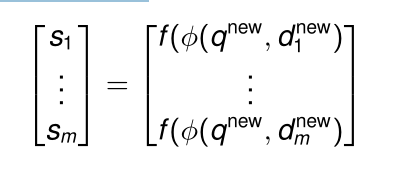
从数学上讲，它是从Rm×p到Sm的映射

然而，在二元分类中，映射到组合空间的函数很难学习，我们通常学习实值函数并对其设置阈值，以便在排序时输出一个类标签（正\/负），我们也这样做，但通过排序来代替阈值，例如，让f:Rp→ R是一个评分函数。

培训数据包括查询文档特征和相关性得分



将使用培训数据学习评分函数f，该函数将用于为新查询对文档进行排序。



可以对分数进行排序，以获得一个排名，该排名将使用ERR、AP、NDCG等指标对其绩效进行评估。

许多不错的创新方法已经被开发出来，包括以下部分：

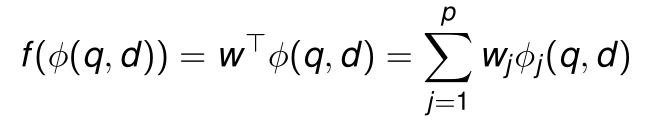
降低：将排名降低到一个更简单的问题（例如，分类或回归）提高：将较弱的排名结合起来，并将其提升为较强的排名大幅度方法：将支持向量机背后的“大幅度”故事扩展到排名案例直接\间接优化：根据希望优化的性能度量选择适当的优化问题。

降低：将排名降低到一个更简单的问题（例如，分类或回归）提高：将较弱的排名结合起来，并将其提升为较强的排名大幅度方法：将支持向量机背后的“大幅度”故事扩展到排名案例直接\间接优化：根据希望优化的性能指标选择适当的优化问题，而不是整齐地优化所有排名方法属于这些类别之一！

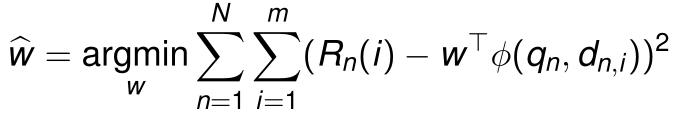
**回归**

也许最简单的方法是：将查询文档功能上的相关性分数（φ（q，di），R（i））反馈给你最喜欢的回归算法优点：可以使用已经构建的回归工具缺点：我们没有真正的理由预测相关性分数，我们只是在构建一个排名函数。

最简单的评分函数是线性函数：



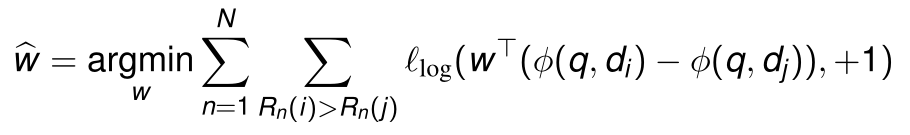
在回归方法中使用线性评分函数将排名降低为线性回归：

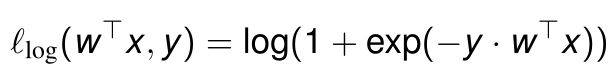


对于一个新的查询文档示例，通过排序（bw>φ（q，di））mi=1输出排名。

如果对于查询q，di的相关性比dj高，那么输入表单的示例（φ（q，di）− φ（q，dj），+1）到你最喜欢的分类算法优点：可以使用已经构建的分类工具缺点：不清楚如何使用成对分类来创建排名。

逻辑回归方法将排名降低到：

逻辑损失



也可以使用其他损失

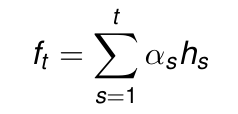


在一个新的三元组（q，di，dj）上，经过训练的分类器将能够告诉您，对于查询q，di是否比dj更相关，如何解决冲突（如循环），并输出d1，dm？快跑！随机选择一个透视文档di使用分类器将其他文档划分为：比di更相关的D>和不相关的D<返回快速排序（D>）、di、快速排序（D<）。

简单的经验法则通常能比随机猜测更好地预测主题行中的“$”⇒ 电子邮件就是垃圾邮件如何将简单的经验法则组合成一个强大的分类器（比如准确率为90%）？Adaboost：也许是最流行的boosting算法 。

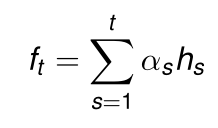
训练数据（X1，Y1），（XN，YN）从均匀分布P1（i）=1\/N开始≤ N≤ N在迭代t=1，2，T:

T使用来自Pt的权重在训练数据T上训练弱分类器ht选择步长αT并设置



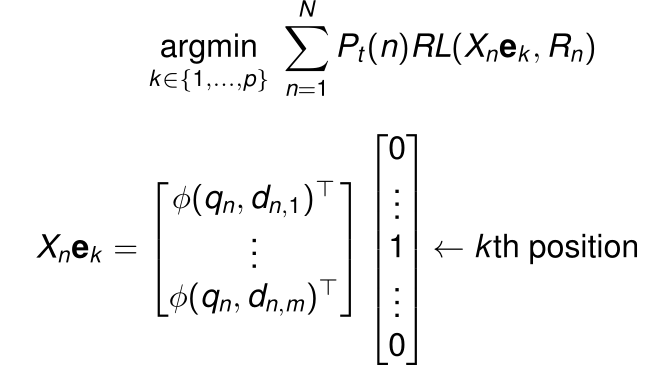
将Pt更新为Pt+1，以便Pt+1更加重视ht表现不佳的示例。

训练数据（X1，R1），（XN，RN）从均匀分布P1（n）=1\/n开始≤ N≤ N在迭代t=1，2，T:T使用来自Pt的权重在训练数据T上训练弱ranker ht选择步长αT并设置。

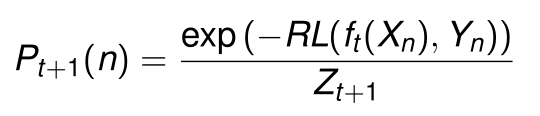


将Pt更新为Pt+1，以便Pt+1更加重视ft表现不佳的示例。

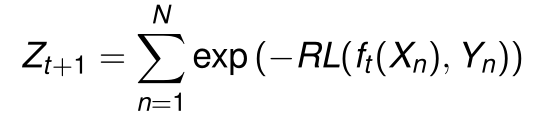
假设RL（排名损失）是我们希望最小化的目标损失（例如，1-NDCG）。弱排名可以只是一个最小化损失的单一功能（例如，Pagerank），即：



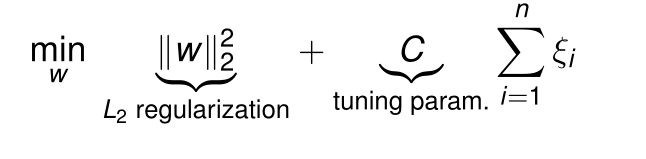
下一个关于训练的分布示例使用以下公式获得：



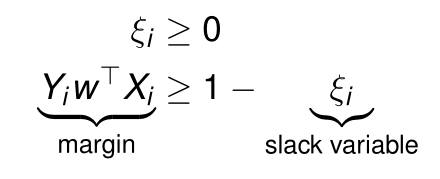
标准化Zt+1确保我们有一个概率分布



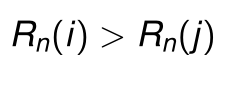
支持向量机解决以下问题：

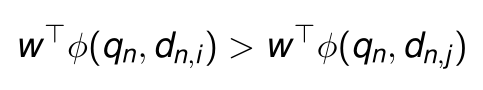


从属于

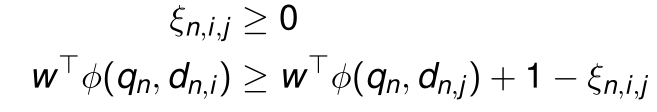


考虑查询QN和文档di、n和dj，n ：

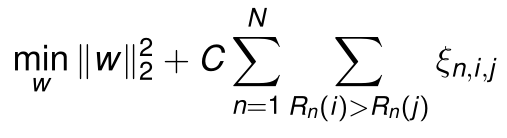




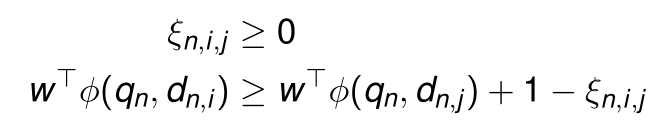
和支持向量机一样，我们可以引入一个带有约束的松弛变量ξn，i，j



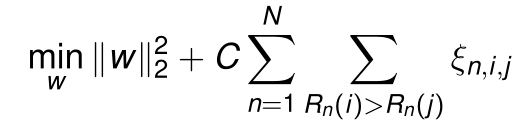
排序支持向量机解决了以下问题：



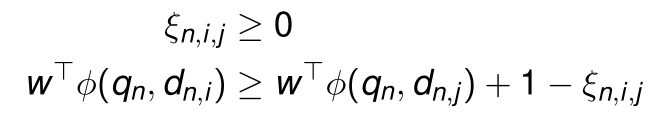
受限制



排序支持向量机解决了以下问题

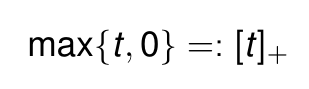


受限制

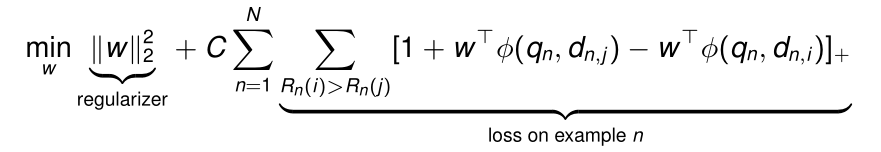


这实际上也是一种缩减方法！

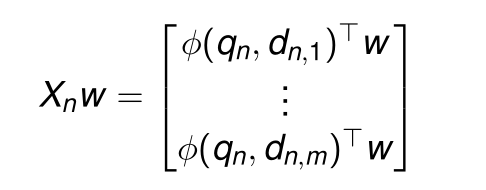
注意，ξ的最小可能值取决于ξ≥ 0和ξ≥ t是



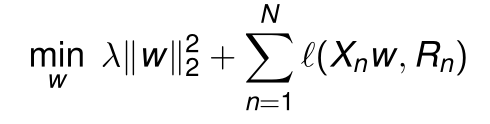
这个观察结果让我们将排名SVM优化写为：



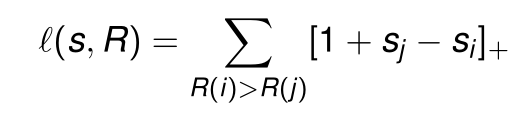
注意 ：



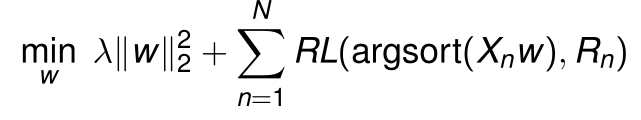
排序SVM问题（其中λ=1\/C）：



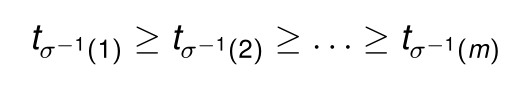
排名上的损失是



假设RL是你想要最小化的实际损失（例如，1-NDCG），那么为什么不解决以下问题呢？

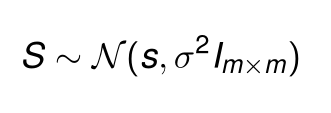


argsort（t）是一个排序σ，它将t的元素按（递减）顺序排列：

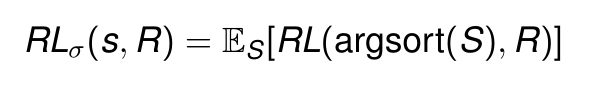


函数w7→ RL（argsort（Xw），R）不是一个好的（从优化的角度来看）！在任何一点附近，它要么是平坦的，要么是不连续的，它最多只能取一个m！将这样的函数之和最小化的不同值在计算上是困难的。

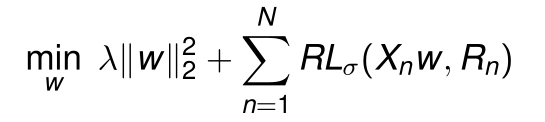
不要考虑确定性分数s=Xw，而是考虑平滑分数分布：



这给了我们一个平滑的排名损失：



然后我们可以优化（使用梯度下降法）：



随着σ的增长，优化问题变得更容易，并且随着σ的收缩，我们不太容易过度拟合，平滑的损失近似于潜在的损失。更好的SmoothRank退火方法：初始化w0（例如，使用线性回归）为t=1，2从wt开始−1.使用（某些版本的）梯度下降法求解正则化RLσt极小化——让新解为wt，并设置σt+1=σt/2 。

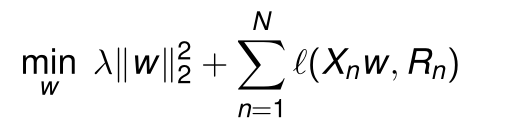
排序支持向量机问题是凸的，因此易于优化，但与性能度量没有直接关系。平滑方法从性能度量开始解决平滑但非凸的问题（可能陷入局部最优）。给定难以优化的性能度量，我们能通过凸代理间接最小化它吗？

我们可以尝试将排名降低到另一个ML问题：例如。，回归或分类我们可以尝试将较弱的等级提升为较强的等级支持向量机的边际最大化方法自然会导致排序支持向量机排序支持向量机解决正则化损失最小化我们可以直接尝试最小化目标性能度量（由于平坦性/不连续性而很难；即使在平滑后也可能陷入局部最小值）我们可以解决凸问题吗（无非全局局部极小值），但仍保留与目标绩效指标的原则联系？

然而，由于训练和测试数据规模较小，有时基准测试不如预期可靠。在这种情况下，需要一种理论来保证算法在无限未知数据上的性能O.Chapelle，Y.Chang和T-Y.Liu《学习排名的未来方向》（2011）

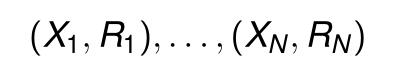
输出空间的组合性质预测空间的大小是m！（m=被排名的事物的数量）没有标准的性能度量许多选择：ERR，MAP，（N）DCG即使有固定的选择，如何提出“好的”凸代理来处理不同类型的反馈模型相关性得分，成对反馈，点击率，偏好DAG大规模挑战并行，分布式，在线\/流式方法的保证。

基于相关性得分的人反馈正则化凸损失最小化排序方法

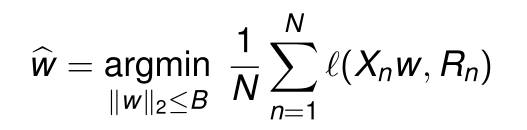


一些常用的性能指标：NDCG、地图综合误差范围和一致性。

理论上的标准假设：例子



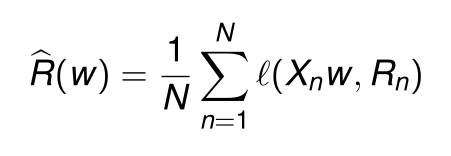
从底层（未知）分布绘制IID考虑损失最小化的约束形式



w的风险：w在“无限不可见数据”上的性能

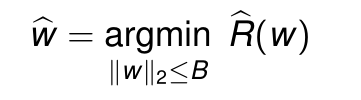


注：在不了解潜在分布的情况下，无法计算w的经验风险：w在训练集上的表现。

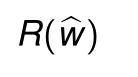


注：仅使用训练数据即可计算

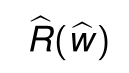
我们说bw是一个经验风险最小化（ERM）



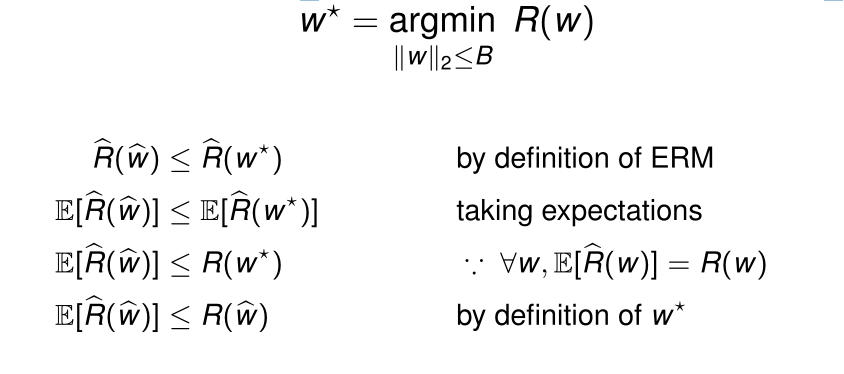
我们通常对泛化误差的界感兴趣



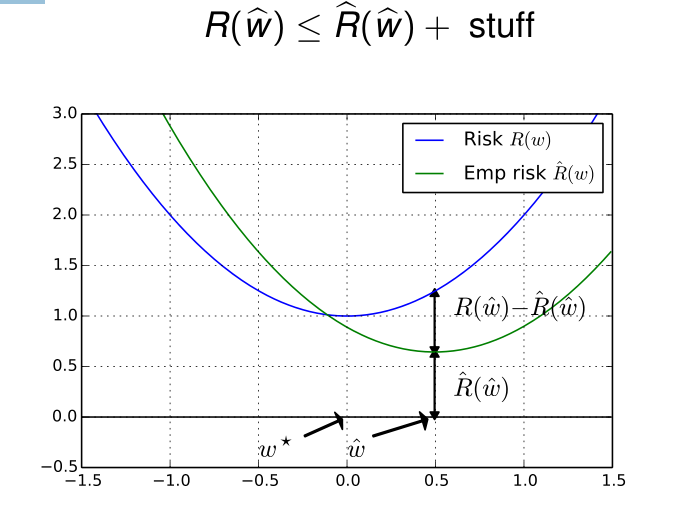
例如，我们期望泛化误差大于



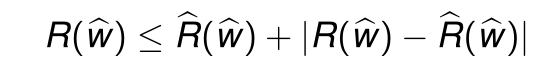
让我们证明bR（bw）在预期中小于R（bw）Let w？成为预期损失最小的评分函数。

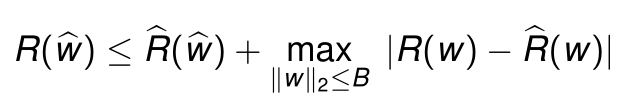


因此，我们可能需要一个形式的泛化误差界

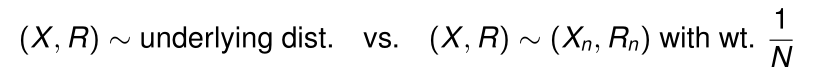


一个无用的泛化错误范围：

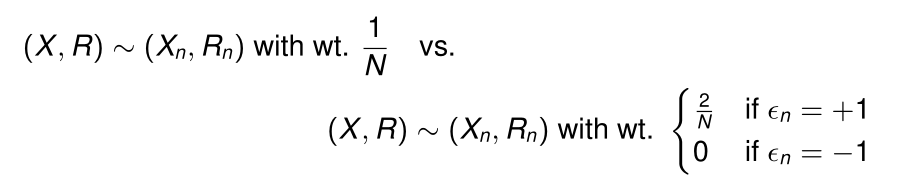




差异R（w）− bR（w）比较了在以下情况下w的预期损失：



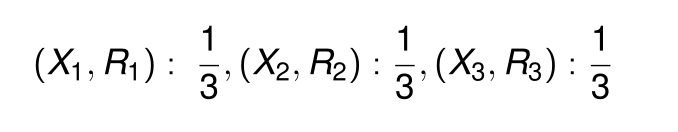
让我们用样本替换基础距离，用（加权）子样本替换样本！



对称有符号伯努利或拉德马赫随机变量。

**加权子样本：一个例子**

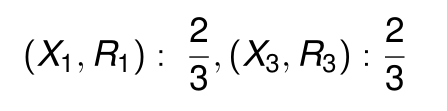
假设带有权重的原始样本为 ：



假设：

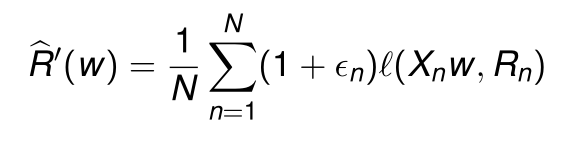


然后加权子样本

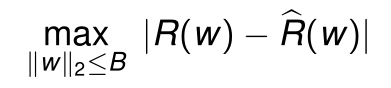


**向广义误差界逼近**

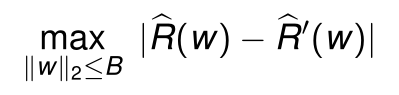
定义子样本下的加权平均值



替代

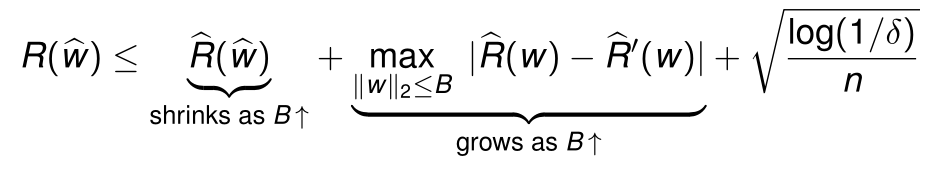


成：



**与数据相关的泛化误差界**

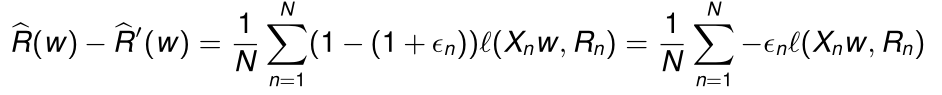
概率至少为1− δ



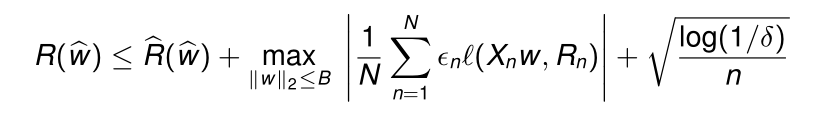
可用于B的数据相关选择

**拉德马赫复杂性**

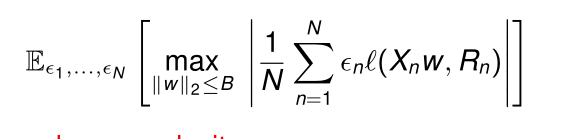
笔记：



因此，很有可能



数量



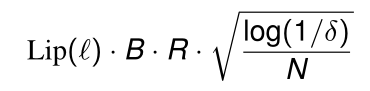
被称为Rademacher复杂性。

**Rademacher的复杂性是如何衡量的？**

Rademacher的复杂性取决于

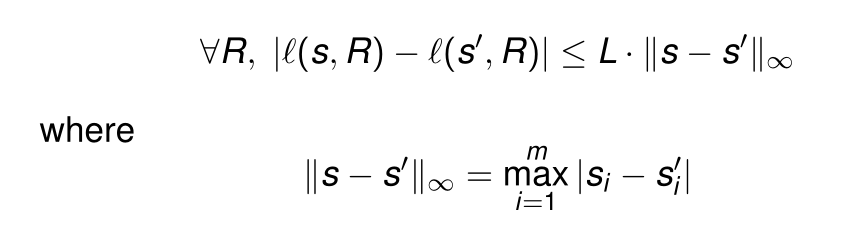
损失函数“更复杂”的损失=⇒ 更大的复杂度w的大小B更大=⇒ 特征的复杂度越大，比如kφ（q，di）k2≤ 更大的=⇒ 更大的复杂性样本量N更大的N=⇒ 较小的复杂性 。

粗略地说，规模就像 ：



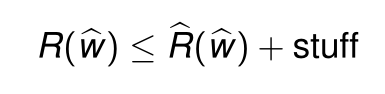
**利普希茨常数**

修正任意相关向量R



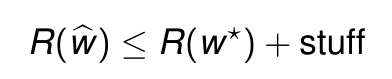
**另一种泛化界**

到目前为止，我们已经看到了形式的界限



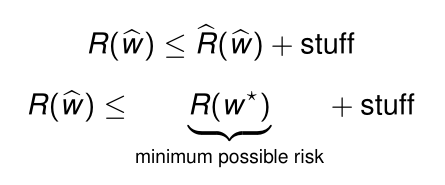
然而，bR（bw）≤ bR（w）和bR（w）集中在R（w ）

因此，也很容易导出形式的边界



**概括界限：重述**

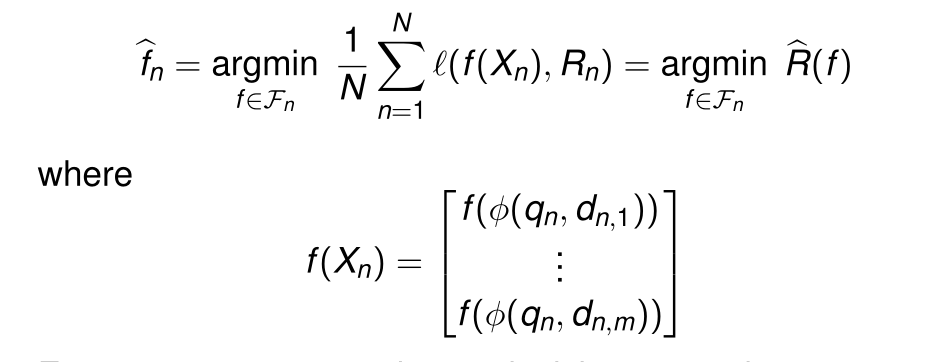
风险R（w）是无限未知数据上的预期损失w经验风险bR（w）是训练数据上的平均损失将训练数据上的损失最小化的bw称为经验风险最小化（ERM），它有两种类型的泛化误差界：



Rademacher复杂度可用于给出ERM的数据相关泛化界限 。

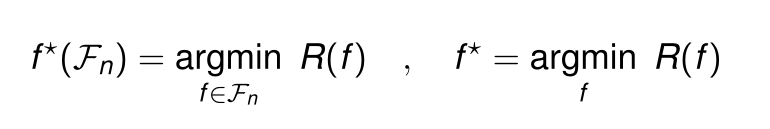
**更丰富的函数类提供更多数据**

随着我们看到更多的数据，自然会扩大计算ERM的函数类

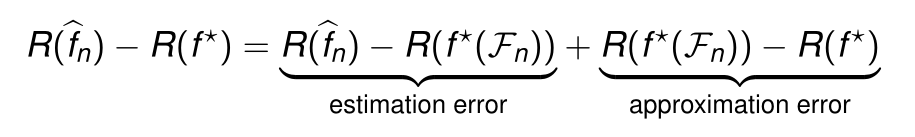


例如，随着n的增长，Fn=更大的决策树、更深的神经网络、正则化程度更低的支持向量机 。

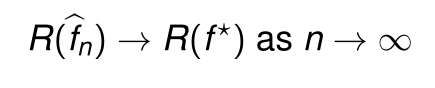
**估计误差与近似误差**



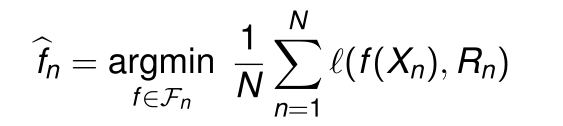
我们可以分解超额风险



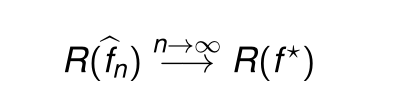
如果Fn增长过快，估计误差不会变为零如果Fn增长过慢，近似误差不会变为零通过明智地增长Fn，我们可以保证两个误差都变为零，这会导致一致性 。



我们说过，我们将使用一个漂亮的（平滑的\/凸的）损失作为替代，并使用

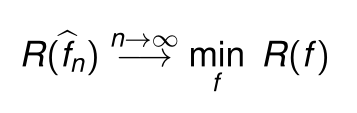


用n放大Fn既不太快也不太慢都会给我们

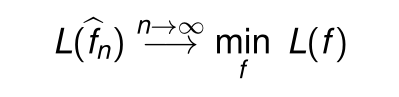


**校准**

修复目标损失RL（如1-NDCG）和替代物（如RankingSVM损失）

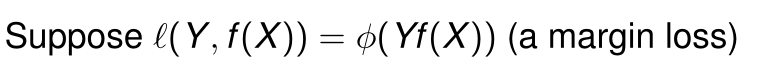


这是否自动保证了以下几点？

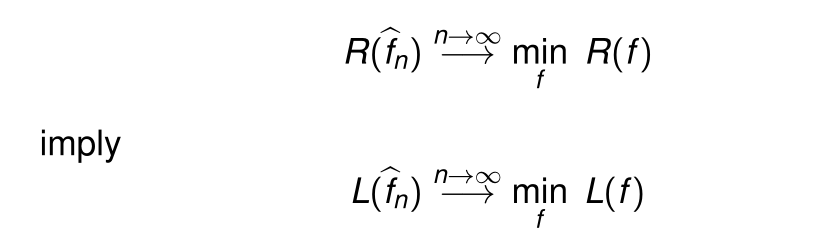


如果是，那么我们说“替代物”是校准的w.r.t.目标损失RL

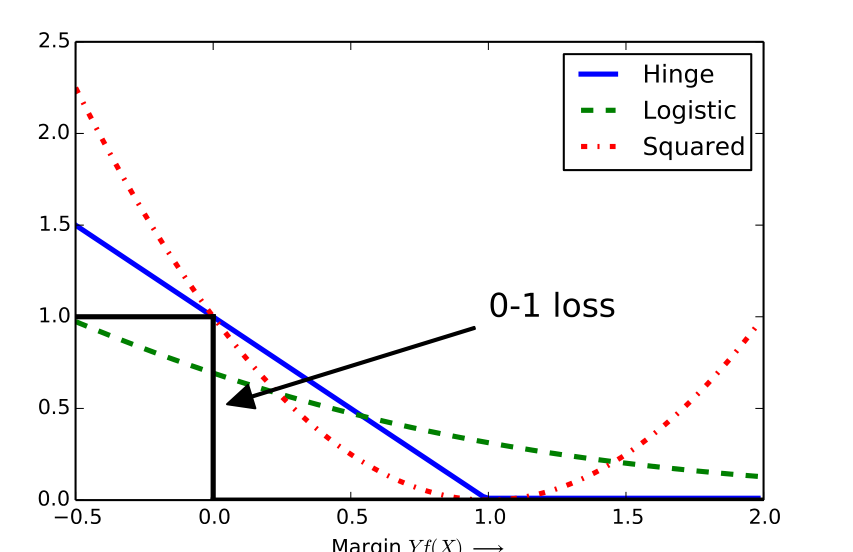
**二元分类中的校正**



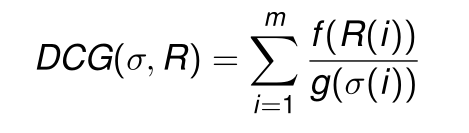
对于基于边距的凸面代理，何时进行

令人惊讶的简单答案：φ0（0）<0

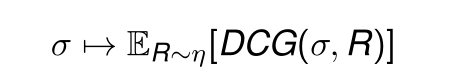
**校准替代物的示例**



校准w.r.t.DCG

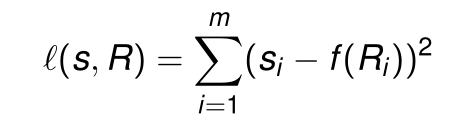


设η是相关向量R上的分布



DCG标定的充分必要条件，s（η）与ER具有相同的排序顺序∼η[f（R）]通过考虑归一化，可以导出NDCG的类似条件 。

考虑基于回归的代理



很容易看出s（η）正是ER∼η[f（R）]因此，基于回归的替代物是DCG校准的 。

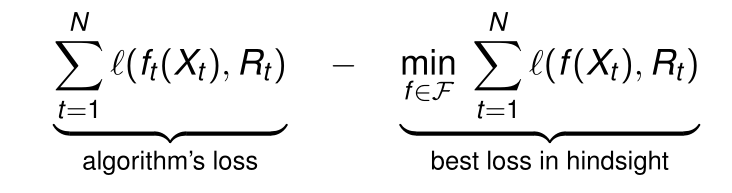
1. DCG案例可能会让人相信凸校准代理总是存在，但事实并非如此！
2. 众所周知，对于ERR和AP警告，不存在任何凸校准替代项：不存在结果假设我们使用argsort从分数移动到排名。

**一致性：回顾**

一致性是一个渐进的（样本量N）→ ∞) 学习算法的特性一致性算法的性能达到“最佳”“ranker当它看到越来越多的数据时，一个最小化替代项的学习算法‘必须平衡估计误差、近似误差权衡’。如果权衡管理得当，它将具有‘如果这也意味着一致性w.r.t’的意义上的一致性。目标损失RL，我们说’是经过校准的w.r.t。RL在排名中的校准比在分类中的更棘手：在某些情况下很容易（例如NDCG），在其他方面不太容易（例如ERR、MAP）

**在线协议**

而不是在固定的批量数据集上进行训练，排名可以在更在线的\/增量设置中进行，t=1，N Ranker看到了特征向量Xt∈ Rm×p Ranker输出评分函数ft:R→ R相关向量Rt显示Ranker遭受损失（ft（Xt），Rt） 。



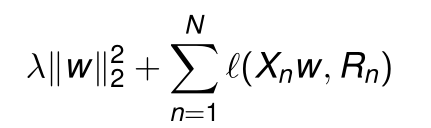
如果l是凸的，我们使用线性评分函数f（x）=w>x，那么我们可以使用在线凸优化（OCO）的工具 。

只关注相关性可能会导致搜索结果冗余一个查询可能会模棱两可：捷豹（汽车vs.动物）简单地为查询的一个方面列出高度相关的结果是不好的，需要鼓励搜索结果的多样性需要新的性能度量\方法\理论框架 。

**大规模学习排名**

考虑对最近发生的事件进行查询的新闻条目或推文需要推广更近期的网页，但只有当查询确实是新近敏感的不同时间尺度：对于年度事件（年）与突发新闻（天/小时）可能的方法：将查询分类为不敏感或不合适，然后路由到适当的秩 。

正则化目标的梯度下降批量优化



可以很容易地在集群集成上并行化基于bagging的方法可以并行地在不同的数据分区上训练RANKER这些想法已经用于有监督的ML，但在排名中没有太多见Apache Spark项目 。

**总结**

排名问题出现在各种各样的环境中学习排名可以是，也已经被视为一个有监督的ML问题我们调查了大量的排名方法我们讨论了排名方法的泛化误差范围和一致性我们看到了一些学习排名的高级主题，这些主题正在积极研究中 。

**建议**

获取LETOR 3.0或4.0数据集，并尝试重现报告的基线下载大型MSLR-WEB30k数据集，查看运行您最喜欢的学习算法需要多长时间来对其进行排名阅读Mohri、Rostamizadeh和T alwalkar的《机器学习基础》第9章（排名）。尽可能多地做练习！阅读Freund和Schapire的《提升：基础和算法》第11章（学习排名）。尽可能多地做练习！

**谢谢大家的关注！**