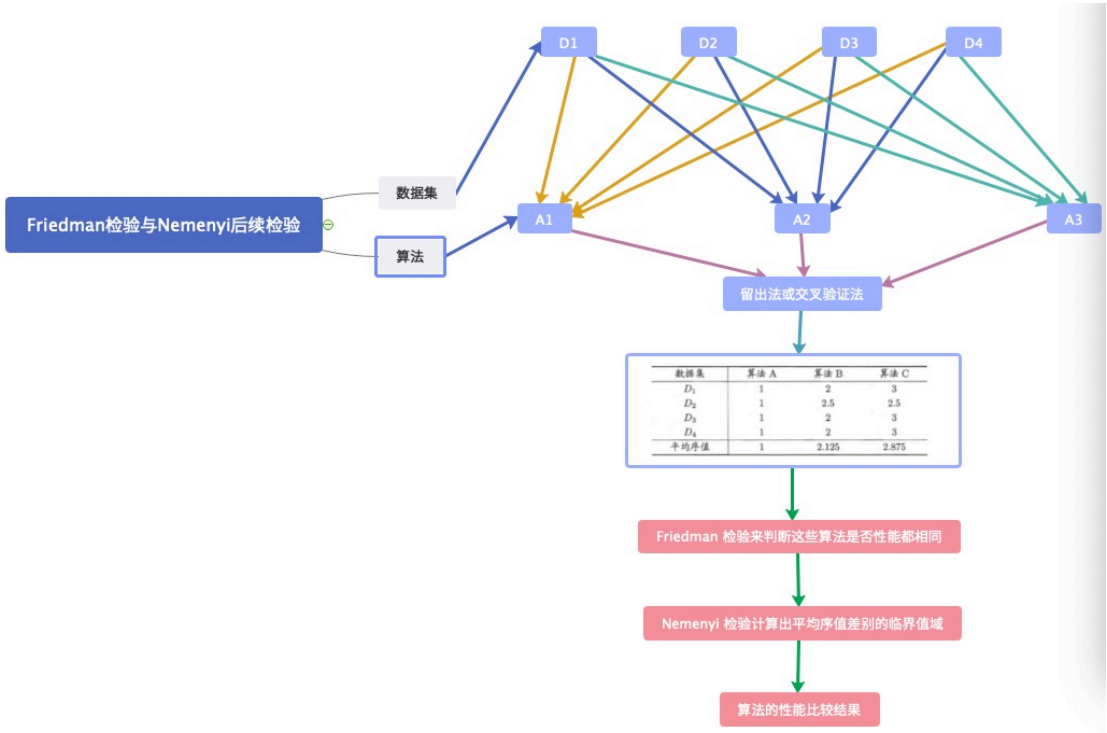
一、前言

> 本文源于《机器学习》–周志华

当有多个算法参与比较时，一种做法是在每个数据集上分别列出两两比较的结果，而在两两比较时可使用前述方法;另一种方法更为直接，即使用基于算法排序的Friedman检验.

算法排序的Friedman检验流程图：



假定我们用 D1 、D2 、D3和D4四个数据集对算法A 、B 、C进行比较.首先，使用留出法或交叉验证法得到每个算法在每个数据集上的测试结果，然后在每个数据集上根据测试性能由好到坏排序，并赋予序值1，2,...，若算法的测试性能相同，则平分序值.例如，在D1和D3上，A最好 、 B其次 、C最差，而在D2上，A最好 、B与C性能相间，.....，则可列出下表 ，其中最后一行通过对每一列的序值求平均，得到平均序值.

表1

数据集	算法 A	算法 B	算法 C
D_1	1	2	3
D_2	1	2.5	2.5
D_3	1	2	3
D_4	1	2	3
平均序值	1	2.125	2.875

然后，使用Friedman 检验来判断这些算法是否性能都相同.若相同，则它们的平均序值应当相同.假定我们在N个数据集上比较k个算法，令 r_i 表示第4个算法的平均序值，为简化讨论，暂不考虑平分序值的情况，则 r_i 服从正态分布,其均值和方差分别为 (k+1)/2和 $\left(k^2-1\right)/12$. 变量

式1 $\tau_{x^2}=\frac{k-1}{k}\cdot\frac{12N}{k^2-1}\sum_{i=1}^{k}\left(r_i-\frac{k+1}{2}\right)^2=\frac{12N}{k(k+1)}\left(\sum_{i=1}^{k}r_i^2-\frac{k(k+1)^2}{4}\right)$

在k和N都较大时，服从自由度为k-1的χ2 分布.

然而,上述这样的"原始Friedman 检验"过于保守，现在通常使用变量

式2 $\tau_F=\frac{(N-1)\tau_{\chi^2}}{N(k-1)-\tau_{\chi^2}}$

其中 τ_{x^2} 由上式得到. τ_F 服从自由度为k-1和(k-l)(N-1)的F分布，下表给出了一些常用临界值.

表2

$\alpha = 0.05$									
数据集	算法个数 k								
个数 N	2	3	4	5	6	7	8	9	10
4	10.128	5.143	3.863	3.259	2.901	2.661	2.488	2.355	2.250
5	7.709	4.459	3.490	3.007	2.711	2.508	2.359	2.244	2.153
8	5.591	3.739	3.072	2.714	2.485	2.324	2.203	2.109	2.032
10	5.117	3.555	2.960	2.634	2.422	2.272	2.159	2.070	1.998
15	4.600	3.340	2.827	2.537	2.346	2.209	2.104	2.022	1.955
20	4.381	3.245	2.766	2.492	2.310	2.179	2.079	2.000	1.935
$\alpha = 0.1$									
数据集	算法个数 k								
个数 N	2	3	4	5	6	7	8	9	10
4	5.538	3.463	2.813	2.480	2.273	2.130	2.023	1.940	1.874
5	4.545	3.113	2.606	2.333	2.158	2.035	1.943	1.870	1.811
8	3.589	2.726	2.365	2.157	2.019	1.919	1.843	1.782	1.733
10	3.360	2.624	2.299	2.108	1.980	1.886	1.814	1.757	1.710
15	3.102	2.503	2.219	2.048	1.931	1.845	1.779	1.726	1.682
20	2.990	2.448	2.182	2.020	1.909	1.826	1.762	1.711	1.668

若"所有算法的性能相同"这个假设被拒绝，则说明算法的性能显著不同.这时需进行"后续检验" (post-hoc test)来进一步区分各算法.常用的有 Nemenyi 后续检验. Nemenyi 检验计算出平均序值差别的临界值域

式3 $CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}}$ 小表给出了 α= 0.05和 0.1 时常用的如值.若两个算法的平均序值之差超出了临界值域 CD，则以相应的置信度拒绝"两个算法性能相同"这一假设。

表3

α	算法个数 k								
	2	3	4	5	6	7	8	9	10
0.05	1.960	2.344	2.569	2.728	2.850	2.949	3.031	3.102	3.164
0.1	1.645	2.052	2.291	2.459	2.589	2.693	2.780	2.855	2.920

以表1中的数据为例，先根据式1和2计算出 τ_F =24.429，由表2可知，它大于α=0.05时的F检验临界值5.143，因此拒绝"所有算法性能相同"这个假设.然后使用Nemenyi 后续检验，在表3中找到k=3时 $q_{0.05}$ =2.344，根据式3计算出临界值域CD=1.657，由表1中的平均序值可知,算法A与B的差距，以及算法B与C的差距均未超过临界值域，而算法A与C的差距超过临界值域，因此检验结果认为算法 A 与 C 的性能显著不 同，而算法 A 与 B 、以及算法 B 与 C 的性能没有显著差别.

上述检验比较可以直观地用Friedman 检验图显示.例如根据表1的序值结果可绘制出下图，图中纵轴显示各个算法，横轴是平均序值.对每个算法，用一个圆点显示其平均序值，以圆点为中心的横线段表示临界值域的大小.然后就可从图中观察，若两个算法的横线段有交叠，则说明这两个算法没有显著差别，否则即说明有显著差别.从下图中可容易地看出,算法A与B没有显著差别，因为它们的横线段有交叠区域，而算法A显著优于算法C因为它们的 横线段没有交叠区域.