|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | 理论 | 步骤 | 代价函数 | 优劣 | 备注 |
| Logistic Regression | 假设数据服从伯努利分布,通过极大化似然函数的方法，运用梯度下降来求解参数 |  | J(θ)=-+ |  | [logistic regression](https://github.com/le773/PythonDebug/blob/master/supervised%20learning/06logistic/logistic.md) |
|  |  |
| SVM | 1.拉格朗日乘子法  2.对偶问题  3.二次规划  4.SMO |  | 0.5\*  st. |  |  |
|  |
| KNN | 投票表决 | 1.假设有一个带有标签的样本数据集（训练样本集），其中包含每条数据与所属分类的对应关系。  2.输入没有标签的新数据后，将新数据的每个特征与样本集中数据对应的特征进行比较。  a.计算新数据与样本数据集中每条数据的距离。  b.对求得的所有距离进行从小到大排序  c.取前k（k一般小于等于 20）个样本数据对应的分类标签。  3.求 k 个数据中出现次数最多的分类标签作为新数据的分类。 |  | 1.理论简单 |  |
| 1.计算量大 | KD-Tree |
| KD-Tree |  | (零)、设 L 为一个有 k 个空位的列表，用于保存已搜寻到的最近点。  (一)、根据 p 的坐标值和每个节点的切分向下搜索（也就是说，如果树的节点是按照 x<sub>r</sub>=a 进行切分，并且 p 的 r 坐标小于 a，则向左枝进行搜索；反之则走右枝）。  (二)、当达到一个底部节点时，将其标记为访问过。  如果 L 里不足 k 个点，则将当前节点的特征坐标加入 L；  如果 L 不为空并且当前节点的特征与 p 的距离小于 L 里最长的距离，则用当前特征替换掉 L 中离 p 最远的点。  (三)、如果当前节点不是整棵树最顶端节点，执行 (a)；反之，输出 L，算法完成。  a. 向上爬一个节点。  如果当前（向上爬之后的）节点未曾被访问过，将其标记为被访问过，然后执行 (1) 和 (2)；  如果当前节点被访问过，再次执行 (a)。  (1)此时如果 L 里不足 k 个点，则将节点特征加入 L；  如果 L 中已满 k 个点，且当前节点与 p 的距离小于 L 里最长的距离，则用节点特征替换掉 L 中离最远的点。  (2) 计算 p 和当前节点切分线的距离。  如果该距离大于等于 L 中距离 p 最远的距离并且 L 中已有 k 个点，则在切分线另一边不会有更近的点，执行(三)；  如果该距离小于 L 中最远的距离或者 L 中不足 k 个点，则切分线另一边可能有更近的点，因此在当前节点的另一个枝从(一)开始执行。 |  |  |  |
|  |

= =

多元GBDT分类算法

样本k负梯度误差

=

=