- 监督学习算法
  - 最大熵模型 33
  - 决策树（分类与回归）
    - 特征选择
      - ID3（信息增益最大且大于阈值的特征）
        - 熵的计算公式 $H(D)=-\sum_{k=1}^{K}\frac{|C_k|}{|D|}log_2\frac{|C_k|}{|D|}$
        - 条件熵的计算公式 $H(D|A)=\sum_{i=1}^{n}\frac{|D_i|}{|D|}H(D_i)=-\sum_{i=1}^{n}\frac{|D_i|}{|D|}\sum_{k=1}^{K}\frac{|D_{ik}|}{|D_i|}log_2\frac{|D_{ik}|}{|D_i|}$
        - 信息增益计算公式
          - $g(D,A)=H(D)-H(D|A)$
          - 信息增益等于熵与条件熵的差（互信息）
          - 信息增益越大，代表了以所选特征划分后，训练集的不确定性越小
          - 以信息增益划分训练集的特征，存在偏向于选择取值较多的特征的问题。
      - C4.5（信息增益比最大的作为最优特征）
        - 信息增益比计算公式
          - $g_R(D,A)=\frac{g(D,A)}{H_A(D)}$
          - $H_A(D)=-\sum_{i=1}^{n}\frac{|D_i|}{|D|}log_2\frac{|D_i|}{|D|}$ $H_A(D)$:训练集 D 关于特征 A 的值的熵
      - CART （生成的是二叉树）
        - 分类树：基尼指数最小化准则
          - 基尼指数计算公式
            - $Gini(D)=1-\sum_{k=1}^{K}(\frac{|C_k|}{|D|})^2$ K 是类的个数
            - $Gini(D,A)=\frac{|D_1|}{|D|}Gini(D_1)+\frac{|D_2|}{|D|}Gini(D_2)$
          - 基尼指数越大，不确定性越大（同熵），基尼指数小的作为最优特征
        - 回归树（最小二乘回归树）：平方误差最小化准则
    - 决策树学习的策略
      - 以损失函数为目标函数的最小化。它的损失函数极小化通常等价于正则化的极大似然估计。
        - 决策树的生成只考虑了提高信息增益或信息增益比对训练数据进行更好的拟合，而决策树的剪枝通过优化损失函数还考虑了减小模型复杂度。决策树的生成学习局部的模型，而决策树的剪枝学习整体的模型。
      - $C_\alpha(T)=\sum_{t=1}^{|T|}N_tH_t(T)+\alpha|T|=C(T)+\alpha|T|$
        - $|T|$ 为叶节点个数，代表了模型的复杂度
        - N_t 为该叶节点 t 包含的样本数
        - $H_t(T)$ 叶节点 t 上的经验熵，可近似地代表分类误差率
        - $\alpha\geq 0$,权衡预测误差和模型复杂度，较大的 α 促使选择较简单的模型树
        - $H_t(T)=-\sum_{k}\frac{N_{tk}}{N_t}log\frac{N_{tk}}{N_t}$
        - $C(T)=\sum_{t=1}^{|T|}N_tH_t(T)=-\sum_{t=1}^{|T|}\sum_{k=1}^{K}N_{tk}log\frac{N_{tk}}{N_t}$
        - N_{tk} 为叶节点 t 上 k 类的样本点个数
        - $C(T)$ 代表了预测误差。所以决策树是对数损失函数
    - 树的生成
      - 开始，构建根节点，将所有训练数据都放到根节点上，选择一个最有特征，按照这一特征将训练数据集分割成子集，使得各个子集有一个在当前条件下最好的分类。如果这些子集已经能够被基本正确分类，那么构建叶节点，并将这些子集分到所对应的叶节点去；如果还有子集不能被基本正确分类，那么就对这些子集选择新的最有特征，继续对其进行分割，构建相应的节点。如此递归的进项下去，直至所有训练数据子集被基本正确分类，或者没有合适的特征为止。
    - 树的剪枝
      - 预剪枝
      - 后剪枝
        - 去掉过于细分的节点，使其回退到父节点，甚至更高的节点，然后将父节点或更高的节点改为新的叶节点。
      - 防止决策树过拟合，提高泛化性能