BIBDR 北京大数据研究院 | 博雅大数据学院 BOYA BIG DATA COLLEGE | 数据酷客 COOKDATA

北京市大数据人才培训示范基地

第三讲：分类

欧高炎，北京大学博士、博士后

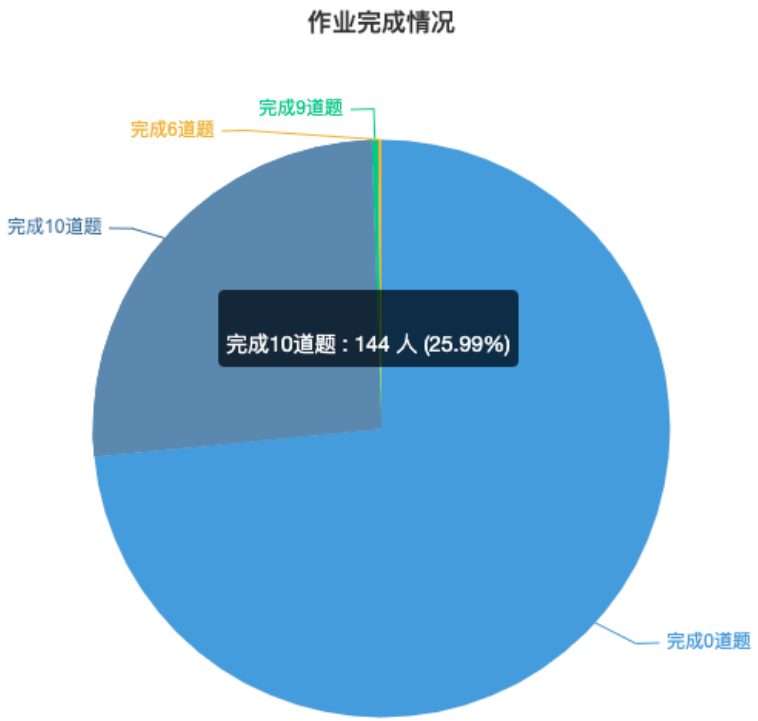
数据酷客创始人，博雅大数据学院院长

“机器学习十讲”直播安排

时间	主题	介绍
5/14	机器学习介绍	机器会学习吗？
5/21	回归	回归初心，方得始终
5/28	分类	分门别类，各得其所
6/4	模型提升	三个臭皮匠，顶个诸葛亮
6/11	聚类	物以类聚，人以群分
6/18	降维	取其精华，去其糟粕
6/23	最优化	摸着石头过河，蒙着眼睛爬山
7/2	维度灾难	来自维数的诅咒
7/9	深度学习	深层次学习的艺术
7/16	强化学习	让机器像人类一样学习

第二讲作业完成情况

- 144 人完成 10 道题，124 人合格



- ”666” 准则：
  - 参加本系列直播 6 次以上（观看回放不计入）
  - 6 次考核合格
  - 60 分以上为合格

内容大纲

1. 数学知识回顾：点到平面距离、梯度下降法、最大似然估计
2. 感知机（Perceptron）
3. 支持向量机（Support Vector Machines）
4. 逻辑回归（Logistic Regression）
5. 分类模型评估与Sklearn分类模块
6. 实践案例：使用感知机、逻辑回归和支持向量机进行中文新闻分类

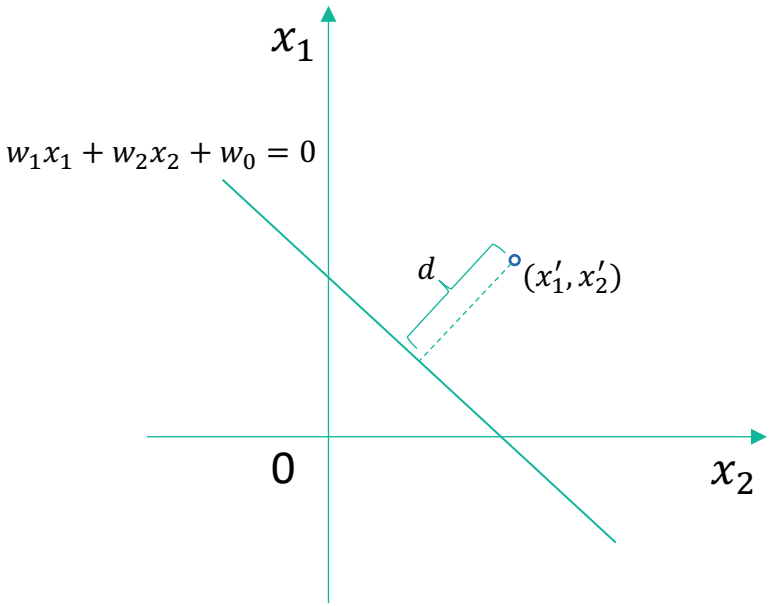
点到平面距离

- 直线方程：$\mathrm{w}_1 x_1 + \mathrm{w}_2 x_2 + \mathrm{w}_0 = 0$
- 点到直线距离

$$d = \frac{|\mathrm{w}_1 x_1' + \mathrm{w}_2 x_2' + \mathrm{w}_0|}{\sqrt{\mathrm{w}_1^2 + \mathrm{w}_2^2}}$$

- 欧式空间超平面： $\mathrm{w}_1 x_1 + \mathrm{w}_2 x_2 + \cdots + \mathrm{w}_d x_d + \mathrm{w}_0 = 0$
- 点到超平面距离:

$$d = \frac{|\mathrm{w}_1 x_1' + \mathrm{w}_2 x_2' + \cdots + \mathrm{w}_d x_d' + \mathrm{w}_0|}{\sqrt{\mathrm{w}_1^2 + \mathrm{w}_2^2 + \cdots + \mathrm{w}_d^2}} = \frac{|\mathbf{w}^{\mathrm{T}} \boldsymbol{x}' + \mathrm{w}_0|}{\| \mathbf{w} \|_2}$$



梯度下降法

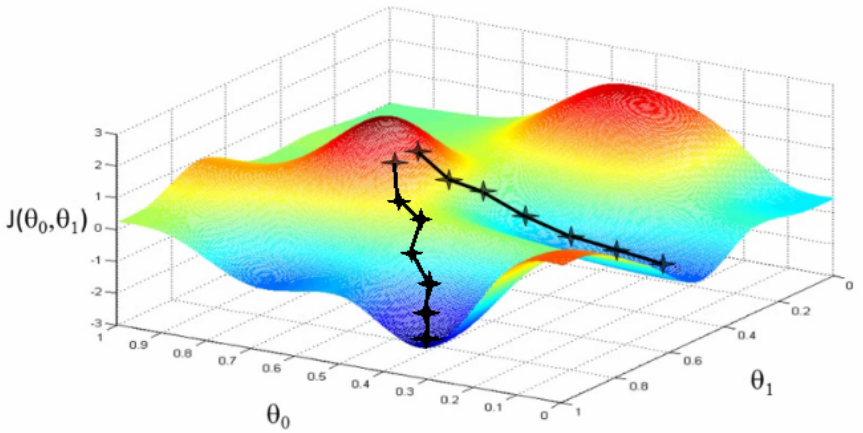
- 求解无约束最优化问题的经典方法，机器学习和深度学习中应用最广泛的模型求解算法
- 如果实值函数 $g(\mathbf{w})$ 在点 $\boldsymbol{a}$ 处可微且有定义，那么函数 $g(\boldsymbol{w})$ 在点 $\boldsymbol{a}$ 处沿着梯度相反的方向$-\nabla g(\boldsymbol{a})$下降最快
- 优化问题：$\min_{\boldsymbol{w}} g(\mathbf{w})$
- 假设初始值为 $\mathbf{w}^{(0)}$ ，梯度下降法用以下迭代公式更新参数：

$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} - \eta_t \nabla g(\mathbf{w}^{(t)})$$

- 其中 η_t 是学习率，取值范围(0,1)

如何走到山谷？

策略：沿着斜坡向下走



随机梯度下降法

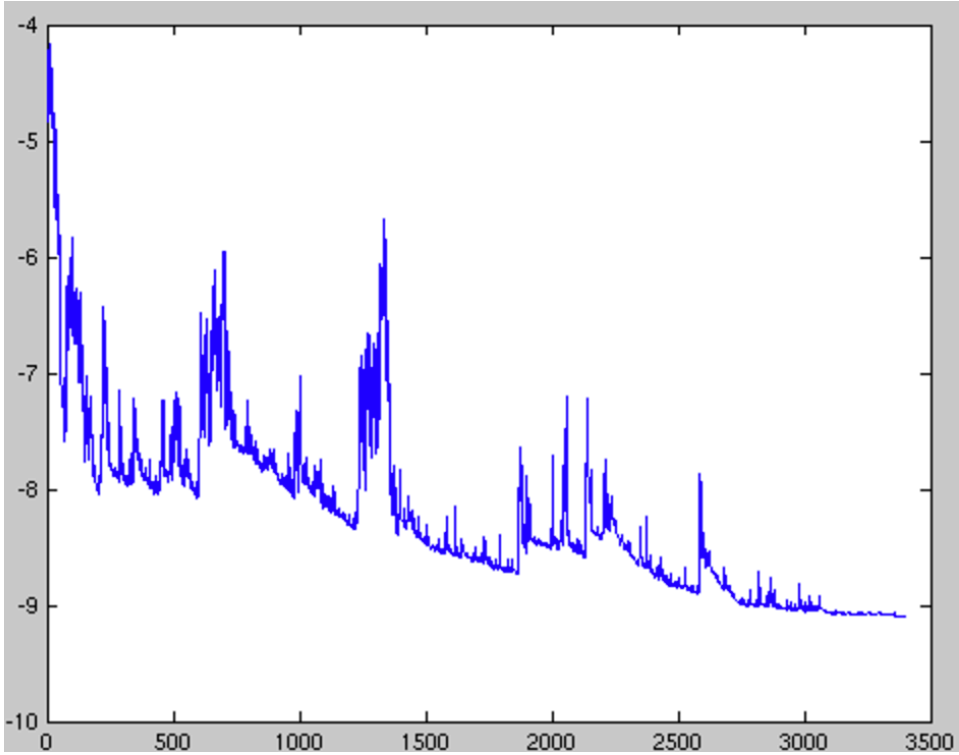
- 机器学习中，优化目标和梯度具有特定结构：

$$\mathrm{L}(\mathbf{w}) = \sum_{i=1}^{n} l(y_i, f(\boldsymbol{x}_i; \mathbf{w})) \qquad \nabla \mathrm{L}(\mathbf{w}) = \sum_{i=1}^{n} \nabla l(y_i, f(\boldsymbol{x}_i; \mathbf{w})) = \sum_{i=1}^{n} \nabla \mathrm{L}_i(\mathbf{w})$$

- 更新参数只用一个样本的梯度，即随机梯度下降法：

$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} - \eta_t \nabla \mathrm{L}_i(\mathbf{w}^{(t)})$$

- 收敛充分条件 $\sum_{t=1}^{\infty} \eta_t = \infty, \sum_{t=1}^{\infty} \eta_t^2 < \infty$
- 需要随着迭代次数的增加降低学习率



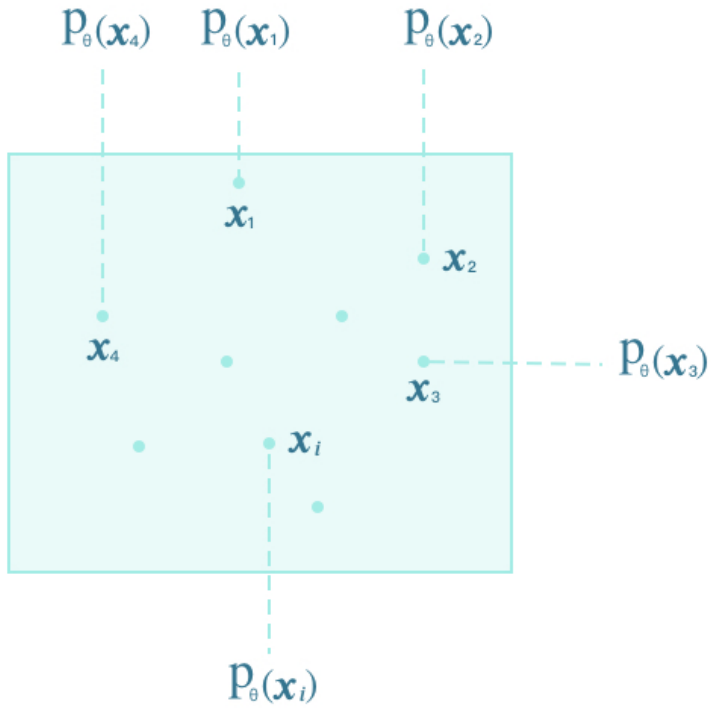
最大似然估计

- "似然"：likelihood 可能性
- 最大似然法，一种求解概率模型参数的方法
- 最早是遗传学家以及统计学家罗纳德·费雪在1912年至1922年间开始使用。

罗纳德·费雪(1890-1962)

最大似然估计

- 假设有 n 个从概率模型 $p_\theta(x)$ 独立生成的样本 $\{x_i\}_{i=1}^n$
- 似然函数 $\mathrm{L}(\theta) = \prod_{i=1}^n p_\theta(x)$
- 通过最大化$\mathrm{L}(\theta)$求解模型参数的方法叫做最大似然法



$$\mathrm{L}(\theta) = \prod \theta^m (1-\theta)^n$$

$$\mathrm{NLL}(\theta) = -m\log\theta - n\log(1-\theta)$$

$$\frac{\mathrm{d}\,\mathrm{NLL}(\theta)}{d\theta} = -\frac{m}{\theta} + \frac{n}{1-\theta} = 0, \text{ 可得 } \theta = \frac{m}{m+n}$$

什么是分类？

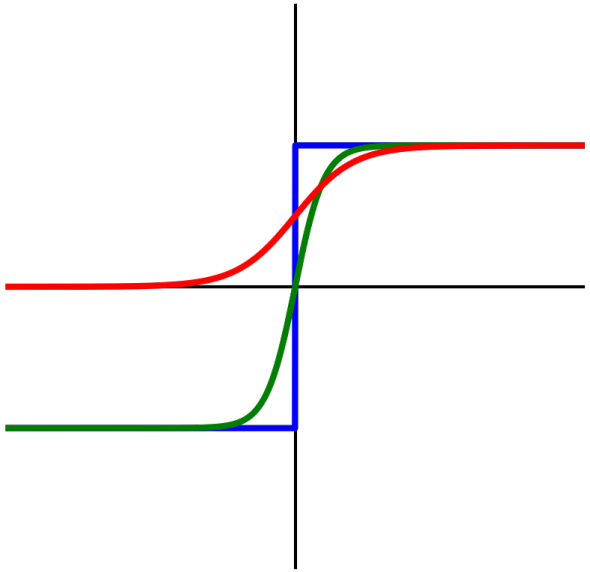
- 另一种典型的有监督学习问题
- 标签（模型预测值）y 为离散值
- 实际应用举例
  - 新闻主题分类：科技、教育、社会、体育？
  - 疾病诊断：根据病人肺部影像图片，诊断是否患 COVID-19 肺炎
  - 市场营销：根据顾客历史购买记录和行为偏好，预测用户是否喜欢新产品
  - 信用评估：根据客户历史信贷记录，预测贷款是否会违约

如何做分类？

- 线性回归: $f(\boldsymbol{x}) = \mathbf{w}^{\mathrm{T}}\boldsymbol{x}, y \in (-\infty, +\infty)$
- 二分类中， $y \in \{-1,1\}$，用回归的方法做分类，在回归结果上添加映射函数 $H(f)$：

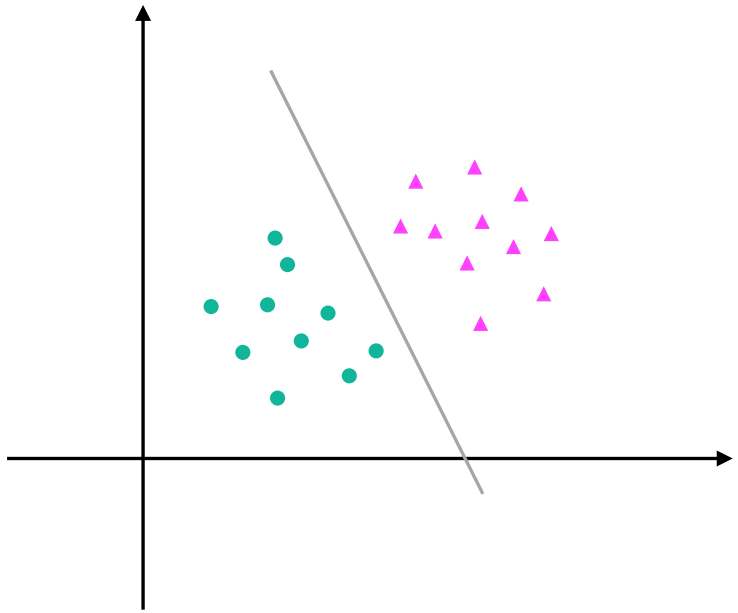
$$H(f) = \begin{cases} +1, f > 0 \\ -1, f \leq 0 \end{cases}$$

- H 的其他选择：
  - $H(f) = \tanh(f)$
  - $H(f) = \sigma(f) = \frac{1}{1+e^{-f}}$



感知机、支持向量机和逻辑回归

- 线性可分训练集 $D = \{\boldsymbol{x}_i, y_i\}_{i=1}^{n}$, $y \in \{-1,1\}$
- 感知机：
  - 找到一条直线，将两类数据分开即可
- 支持向量机：
  - 找到一条直线，不仅将两类数据正确分类，还使得数据离直线尽量远
- 逻辑回归：
  - 找到一条直线使得观察到训练集的“可能性”最大



训练集的矩阵表示

- 假设训练集的特征部分记为 $n\times(d+1)$ 矩阵 $\mathbf{X}$，其中最后一列取值全为 1
- 标签部分记为 $\boldsymbol{y}=(y_1,y_2,\ldots,y_n)^{\mathrm{T}}$，参数记为 $\mathbf{w}=(w_1,w_2,\ldots,w_d,w_0)^{\mathrm{T}}$

$$\hat{\boldsymbol{y}}=\mathbf{X}\mathbf{w}$$

$$\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} & 1 \\ x_{21} & x_{22} & \cdots & x_{2d} & 1 \\ \cdots & \cdots & \cdots & \cdots & 1 \\ x_{i1} & x_{i2} & \cdots & x_{id} & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} & 1 \end{bmatrix} \quad \begin{bmatrix} w_1 \\ w_2 \\ \cdots \\ w_d \\ w_0 \end{bmatrix} \quad \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \cdots \\ \hat{y}_i \\ \cdots \\ \hat{y}_n \end{bmatrix} \quad \begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_i \\ \cdots \\ y_n \end{bmatrix}$$

$\mathbf{X}$ $\mathbf{w}$ $\hat{\boldsymbol{y}}$ $\boldsymbol{y}$

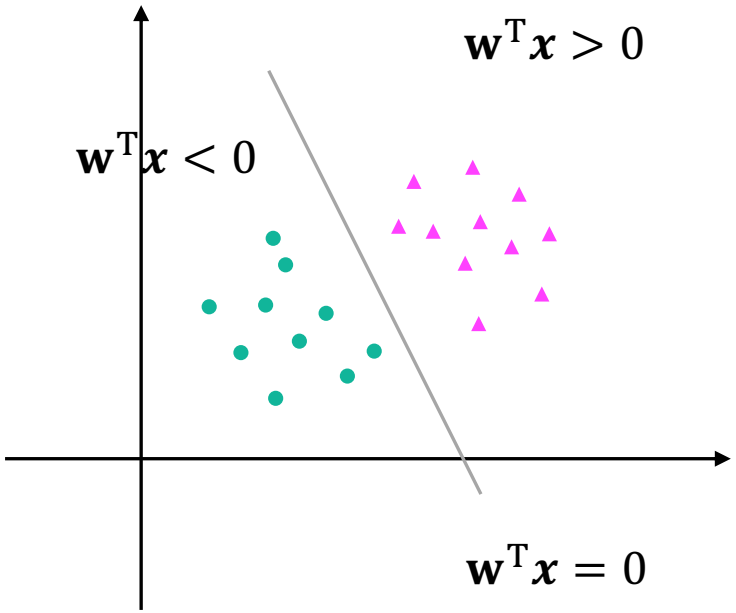
内容大纲

1. 数学知识回顾：点到平面距离、梯度下降法、最大似然估计
2. 感知机（Perceptron）
3. 支持向量机（Support Vector Machines）
4. 逻辑回归（Logistic Regression）
5. 分类模型评估与Sklearn分类模块
6. 实践案例：使用感知机、逻辑回归和支持向量机进行中文新闻分类

感知机

- $f(\boldsymbol{x}) = \mathbf{w}^{\mathrm{T}}\boldsymbol{x}$，$\mathbf{w} = (\mathrm{w}_1, \mathrm{w}_2, \ldots, \mathrm{w}_\mathrm{d}, \mathrm{w}_0)^{\mathrm{T}}$为系数，模型为

$$y = H(f(\boldsymbol{x})) = \begin{cases} +1, & \mathbf{w}^{\mathrm{T}}\boldsymbol{x} > 0 \\ -1, & \mathbf{w}^{\mathrm{T}}\boldsymbol{x} \leq 0 \end{cases}$$



- 决策超平面为：$\mathbf{w}^{\mathrm{T}}\boldsymbol{x} = 0$
- 线性可分训练集 $\mathrm{D} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_\mathrm{n}, y_\mathrm{n})\}$，点$(\boldsymbol{x}_i, y_i)$ 到决策超平面的距离为

$$d_i = \frac{|\mathbf{w}^{\mathrm{T}}\boldsymbol{x}_i|}{\|\mathbf{w}\|_2} = \frac{y_i\mathbf{w}^{\mathrm{T}}\boldsymbol{x}_i}{\|\mathbf{w}\|_2} \rightarrow y_i\mathbf{w}^{\mathrm{T}}\boldsymbol{x}_i$$

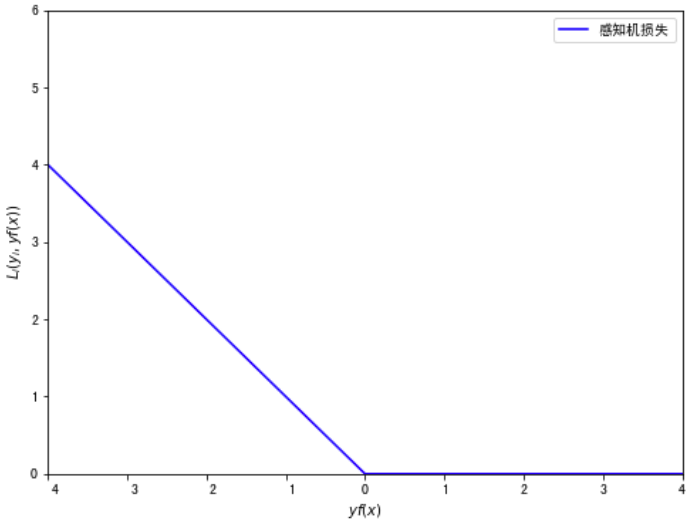
不妨令$\|\mathrm{w}\|_2 = 1$

- 优化目标：误分类样本离超平面距离之和最小

感知机的优化目标

- 目标函数：$\mathrm{L}(\mathbf{w}) = -\sum_{i \in M} y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x_i}$，$M$为误分类样本集合 $\{j | y_j \mathbf{w}^{\mathrm{T}} \boldsymbol{x_j} < 0\}$
- $\mathrm{L}(\mathbf{w}) = \sum_{i=1}^{n} \mathrm{L}(y_i, f(\boldsymbol{x_i}))$
- 梯度：$\nabla \mathrm{L}(\mathbf{w}) = -\sum_{i \in M} y_i \boldsymbol{x_i}$
- 梯度下降法（GD）：

$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} + \eta_t \sum_{i \in M} y_i \boldsymbol{x_i}$$

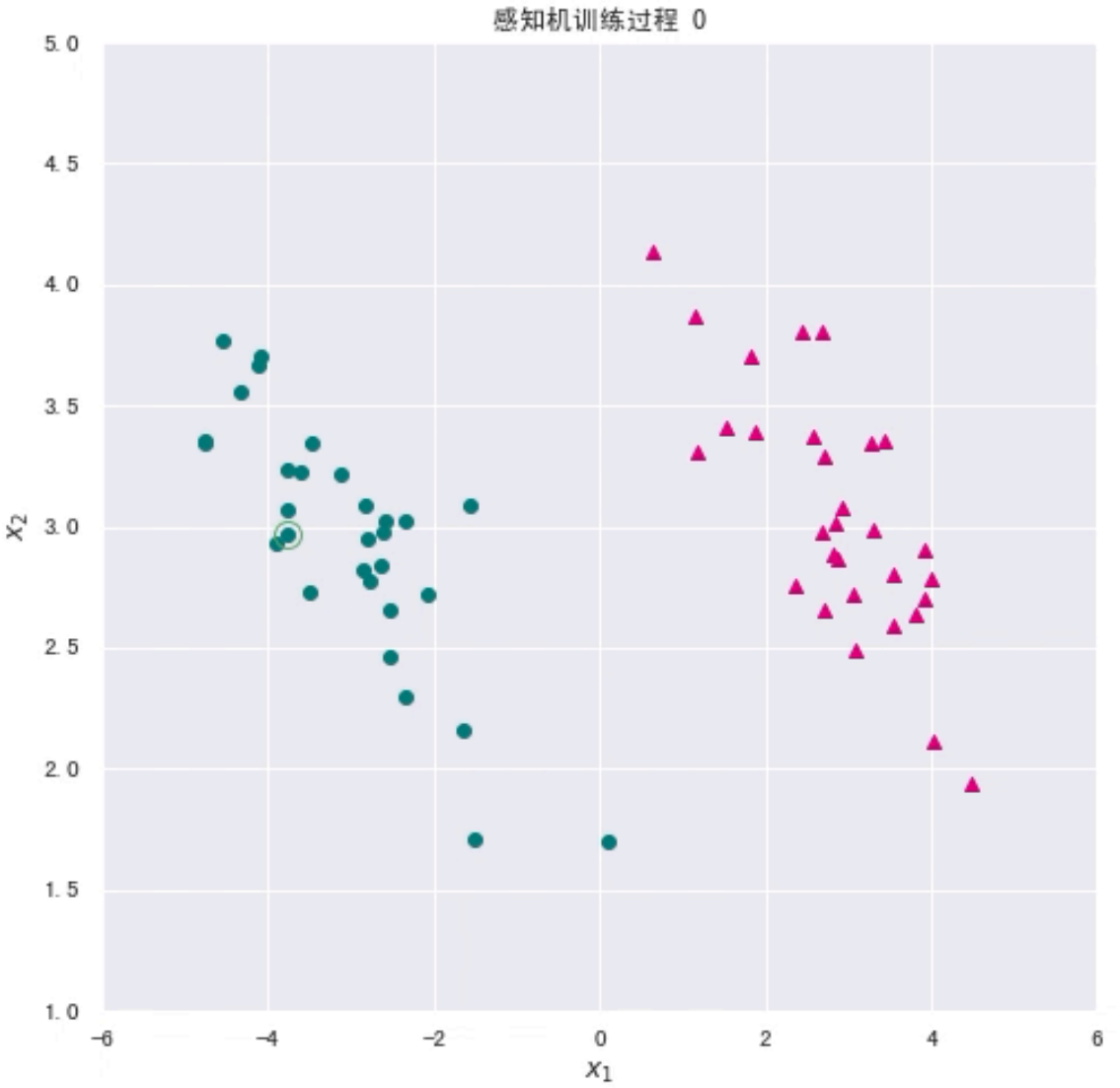


- 随机梯度下降(SGD)：

$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} + \eta_t y_i \boldsymbol{x_i}$$

感知机算法（SGD）

- **输入**：训练数据 $\mathbf{X}, \boldsymbol{y}$，学习率 η，迭代步数 T
- 1 初始化参数 $\mathbf{w}^{(0)}$
- 2 for $t = 1, \ldots, T$
  - 2.1 找出误分类样本集合 M；
  - 2.2 从 M 中随机采样一个样本 i
  - 2.3 更新参数 $\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} + \eta_t y_i \boldsymbol{x_i}$
- **输出**： $\mathbf{w}$

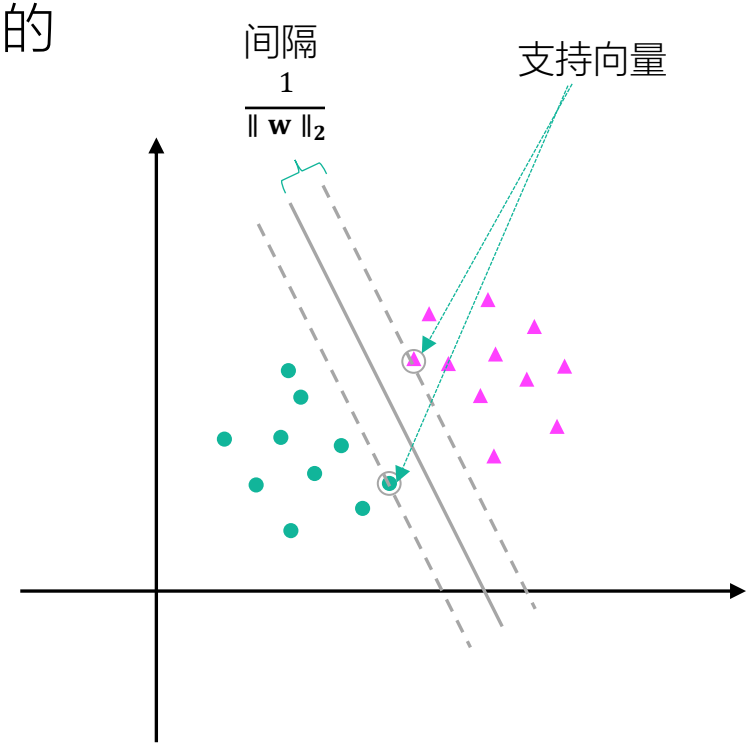


内容大纲

1. 数学知识回顾：点到平面距离、梯度下降法、最大似然估计
2. 感知机（Perceptron）
3. **支持向量机（Support Vector Machines）**
4. 逻辑回归（Logistic Regression）
5. 分类模型评估与Sklearn分类模块
6. 实践案例：使用感知机、逻辑回归和支持向量机进行中文新闻分类

支持向量机：间隔最大化

- 线性可分训练集 $\mathrm{D} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_\mathrm{n}, y_\mathrm{n})\}$，点$(\boldsymbol{x}_i, y_i)$ 到决策超平面的距离为 $d_i = \frac{y_i \mathbf{w}^\mathrm{T} \boldsymbol{x}_i}{\|\mathbf{w}\|_2}$



- 间隔：训练集中离超平面最小的距离 $\min_i \frac{y_i \mathbf{w}^\mathrm{T} \boldsymbol{x}_i}{\|\mathbf{w}\|_2}$
- 间隔最大化：

$$\max_{\mathbf{w}} \min_i \frac{y_i \mathbf{w}^\mathrm{T} \boldsymbol{x}_i}{\| \mathbf{w} \|_2} \Leftrightarrow \max_{\mathbf{w}} \frac{1}{\| \mathbf{w} \|_2} \min_i y_i \mathbf{w}^\mathrm{T} \boldsymbol{x}_i$$

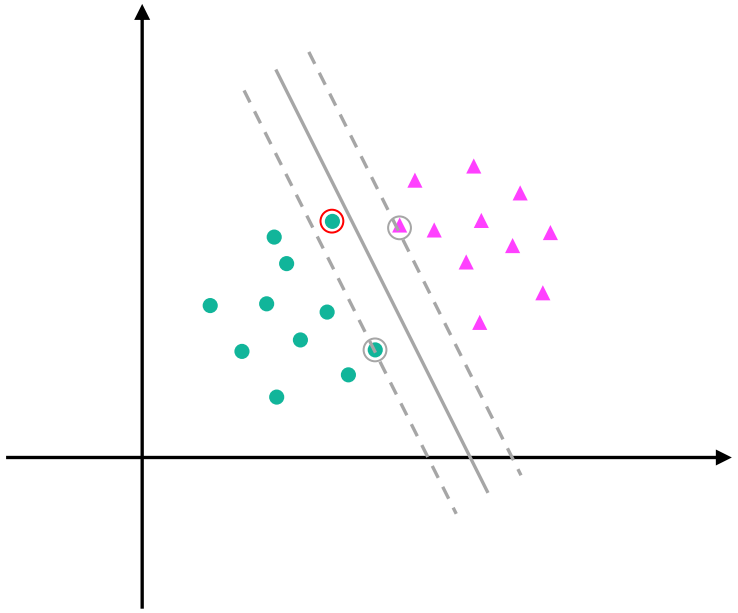
- 不妨令 $\min_i y_i \mathbf{w}^\mathrm{T} \boldsymbol{x}_i = 1$,则上述目标等价于

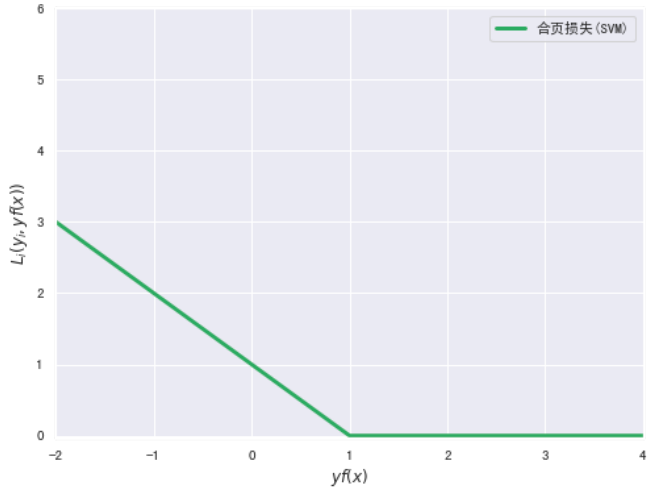
$$\max_{\mathbf{w}} \frac{1}{\|\mathbf{w}\|_2} \Leftrightarrow \min_{\mathbf{w}} \frac{1}{2} \| \mathbf{w} \|_2^2$$

支持向量机：样本损失函数

- 假设 $\min_i y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x_i} = 1$，则对于训练集任意样本需满足 $y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x_i} \geq 1$
- 对于不满足上述条件样本的损失函数定义为 $1 - y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x_i}$
- 则样本损失为合页损失（hinge loss）：

$$\mathrm{L}\big(y_i, f(\boldsymbol{x_i})\big) = \max(0, 1 - y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x_i})$$





优化目标

- 目标函数 $\mathrm{L}(\mathbf{w}) = \frac{\lambda}{2} \| \mathbf{w} \|_2^2 + \sum_{i=1}^{n} \max(0, 1 - y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x}_i)$，其中 λ 超参数
- 记不满足约束的样本集为 $M = \{i | \, y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x}_i < 1\}$，则梯度为

$$\nabla \mathrm{L}(\mathbf{w}) = \lambda \mathbf{w} - \sum_{i \in M} y_i \boldsymbol{x}_i$$

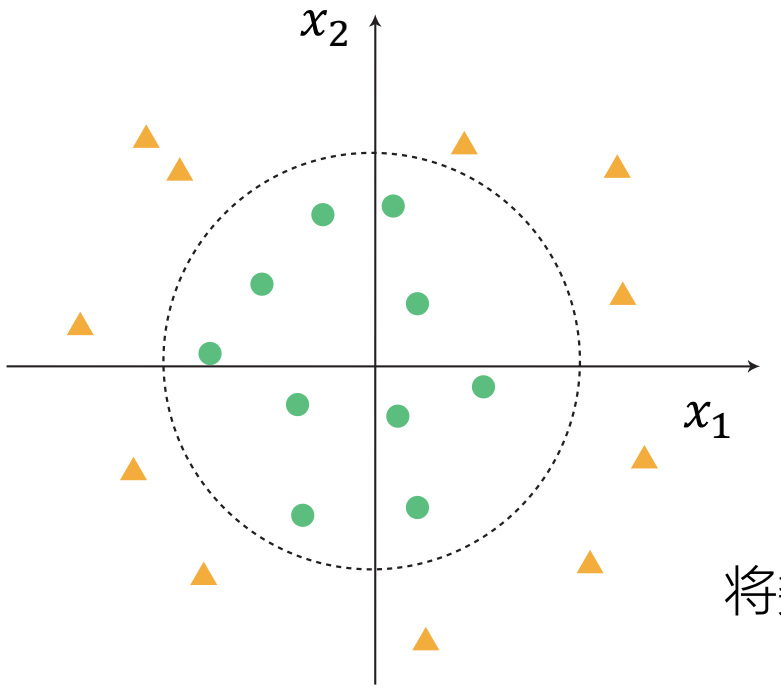
- 梯度下降法：

$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} - \eta_t \left(\lambda \mathbf{w}^{(t)} - \sum_{i \in M} y_i \boldsymbol{x}_i \right)$$

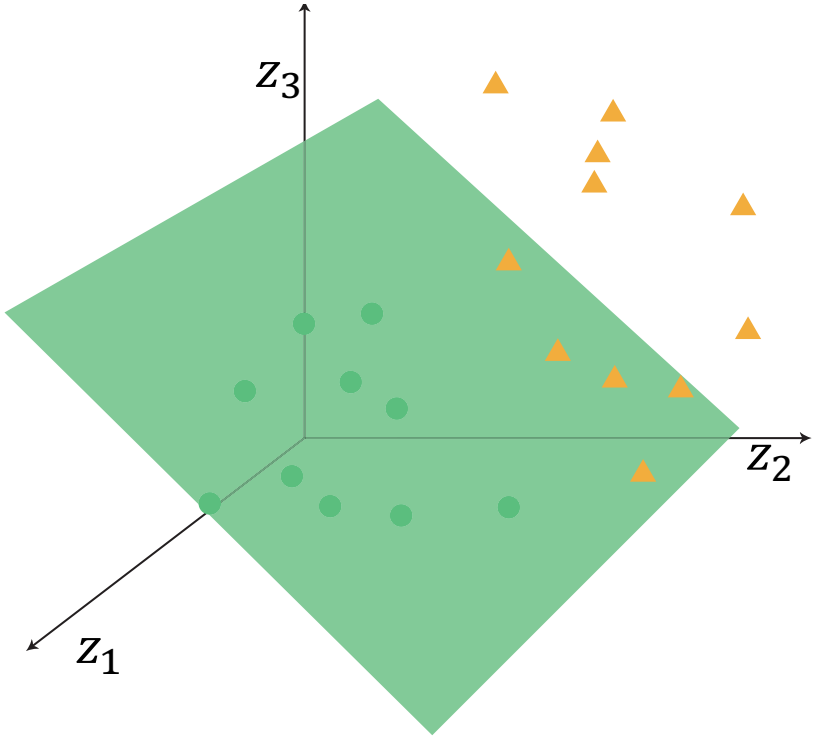
- 随机梯度下降：

$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} - \eta_t \left(\lambda \mathbf{w}^{(t)} - \mathrm{I}(i \in M) y_i \boldsymbol{x}_i \right)$$

非线性：核技巧



映射trick！



将数据点从2维空间映射到3维空间中，使得数据线性可分

内容大纲

1. 数学知识回顾：点到平面距离、梯度下降法、最大似然估计
2. 感知机（Perceptron）
3. 支持向量机（Support Vector Machines）
4. **逻辑回归（Logistic Regression）**
5. 分类模型评估与Sklearn分类模块
6. 实践案例：使用感知机、逻辑回归和支持向量机进行中文新闻分类

逻辑回归：赋予样本概率解释

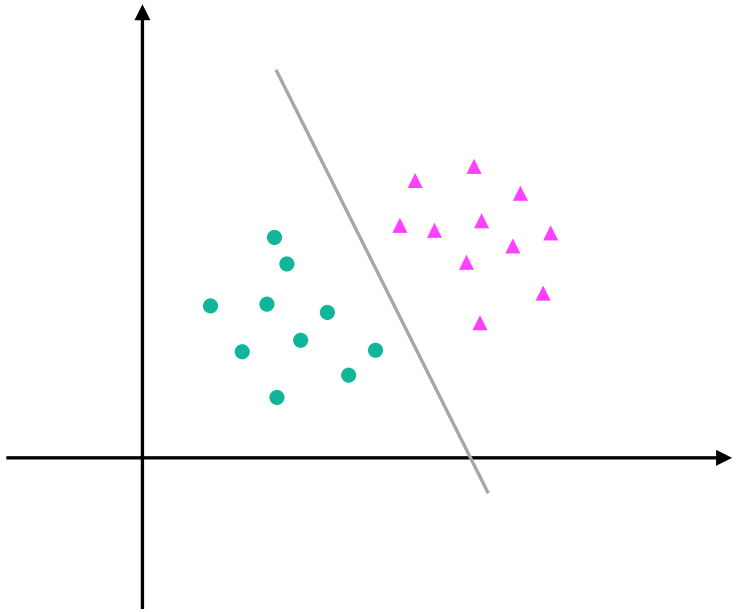
- $f(\boldsymbol{x}) = \mathbf{w}^{\mathrm{T}}\boldsymbol{x}$，$\mathbf{w} = (\mathrm{w}_1, \mathrm{w}_2, \ldots, \mathrm{w}_{\mathrm{d}}, \mathrm{w}_0)^{\mathrm{T}}$为系数

- 训练集 $D = \{\boldsymbol{x}_i, y_i\}_{i=1}^{n}$，$y \in \{-1,1\}$，概率解释：
  - $p(y = 1|\boldsymbol{x}) = \frac{1}{1+e^{-\mathbf{w}^{\mathrm{T}}\boldsymbol{x}}}$
  - $p(y = -1|\boldsymbol{x}) = 1 - p(y = 1|\boldsymbol{x}) = \frac{1}{1+e^{\mathbf{w}^{\mathrm{T}}\boldsymbol{x}}}$

- 考虑到 $y \in \{-1,1\}$，则样本 $(\boldsymbol{x}_i, y_i)$ 概率为：

$$p(y_i|\boldsymbol{x}_i) = \frac{1}{1+e^{-y_i\mathbf{w}^{\mathrm{T}}\boldsymbol{x}_i}}$$

离决策面越远，概率越高



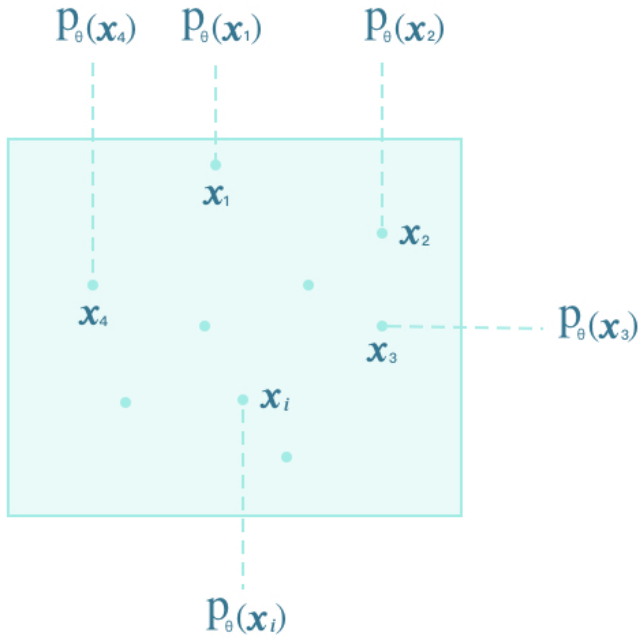
似然函数和负对数似然函数

- 似然函数为 $\mathrm{L}(\mathbf{w}) = \prod_{i=1}^{n} p(y_i|\boldsymbol{x}_i) = \prod_{i=1}^{n} \frac{1}{1+e^{-y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x}_i}}$
- 便于计算，取对数，将连乘转换成求和，负对数似然函数为：
- $\mathrm{NLL}(\mathbf{w}) = \sum_{i=1}^{n} \ln\left(1 + e^{-y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x}_i}\right)$
- 梯度为 $\nabla \mathrm{NLL}(\mathbf{w}) = -\sum_{i=1}^{n} \frac{y_i \boldsymbol{x}_i}{1+e^{y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x}_i}}$
- 梯度下降法：

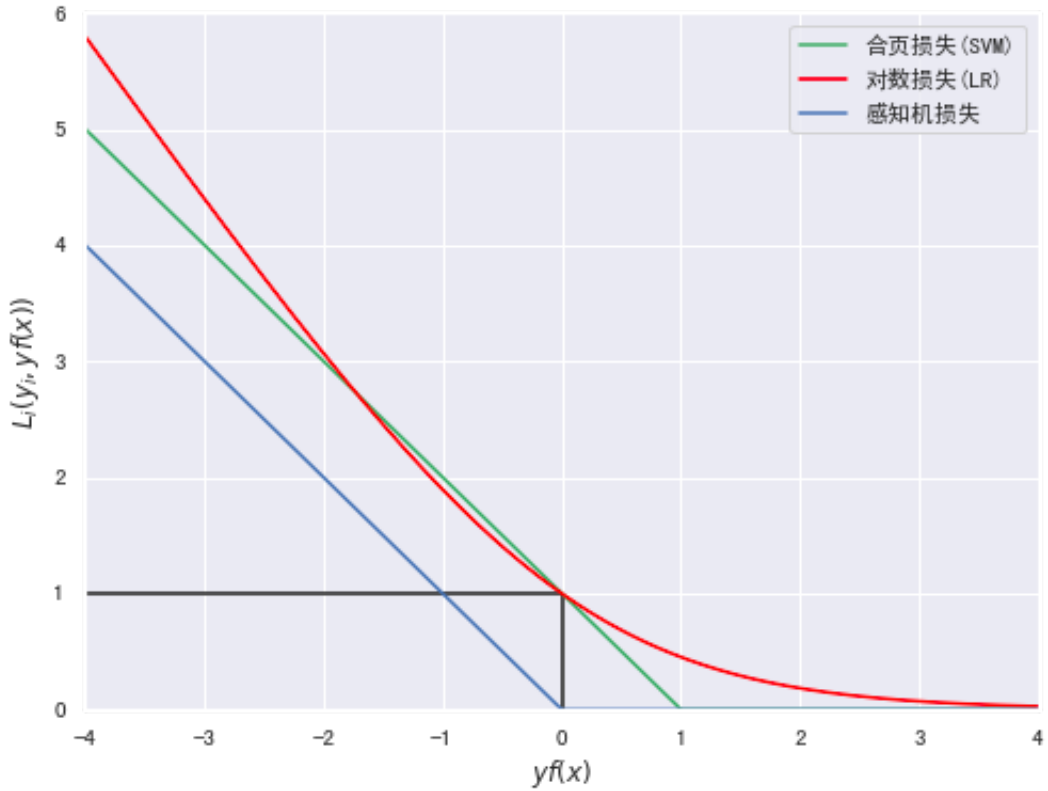
$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} + \eta_t \sum_{i=1}^{n} \frac{y_i \boldsymbol{x}_i}{1 + e^{y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x}_i}}$$

- 随机梯度下降法：

$$\mathbf{w}^{(t+1)} \leftarrow \mathbf{w}^{(t)} + \eta_t \frac{y_i \boldsymbol{x}_i}{1+e^{y_i \mathbf{w}^{\mathrm{T}} \boldsymbol{x}_i}}$$



- 理想的损失函数不能被高效地优化：0-1损失
- 感知机： $\mathrm{L}\big(y, f(\boldsymbol{x})\big) = \max(0, -yf(\boldsymbol{x}))$
- 支持向量机：$\mathrm{L}\big(y, f(\boldsymbol{x})\big) = \max(0,1 - yf(\boldsymbol{x}))$
- 逻辑回归：$\mathrm{L}\big(y, f(\boldsymbol{x})\big) = \ln\left(1 + \exp(-yf(\boldsymbol{x}))\right)$



内容大纲

1. 数学知识回顾：点到平面距离、梯度下降法、最大似然估计
2. 感知机（Perceptron）
3. 支持向量机（Support Vector Machines）
4. 逻辑回归（Logistic Regression）
5. **分类模型评估与 Sklearn 分类模块**
6. 实践案例：使用感知机、逻辑回归和支持向量机进行中文新闻分类

分类问题的评价指标

混淆矩阵 (Confusion Matrix)		预测标签	
		1（正例）	0（负例）
真实标签	1（正例）	TP（真正样本数量）	FN（假负样本数量）
	0（负例）	FP（假正样本数量）	TN（真负样本数量）

$$\text{正确率}(accuracy) = \frac{TP + TN}{TN + FN + FP + TP}$$

$$\text{召回率}(recall) = \frac{TP}{TP + FN}$$

$$\text{精确率}(precision) = \frac{TP}{TP + FP}$$

$$F_1 = \frac{2 \times \text{精确率} \times \text{召回率}}{\text{精确率} + \text{召回率}}$$

实现工具：sklearn.metrics 模块实现了常见的模型评价指标函数，直接调用即可

类	说明
linear_model.SGDClassifier	梯度下降法实现感知机、LR、SVM等
linear_model.LogisticRegression	逻辑回归
neighbors.KNeighborsClassifier	K近邻
tree.DecisionTreeClassifier	决策树
naive_bayes.BernoulliNB	Bernoulli 贝叶斯
naive_bayes.GaussianNB	Gaussian 贝叶斯
naive_bayes.MultinomialNB	多项式贝叶斯
svm.LinearSVC	线性支持向量分类器
svm.SVC	支持向量分类器

方法	说明
fit(X, y)	训练模型
predict(X)	返回X中样本的预测类标签
predict_log_proba(X)	返回X中样本的对数预测类别概率
predict_proba(X)	返回X中样本的预测类别概率
score(X, y)	对X进行预测得到预测标签，再与真实标签y作比对，返回正确率

内容大纲

1. 数学知识回顾：点到平面距离、梯度下降法、最大似然估计
2. 感知机（Perceptron）
3. 支持向量机（Support Vector Machines）
4. 逻辑回归（Logistic Regression）
5. 分类模型评估与 Sklearn 分类模块
6. **实践案例：使用感知机、逻辑回归和支持向量机进行中文新闻分类**

进一步学习

- 数据酷客 “**数据科学导引**”课程 分类模型章节：

http://cookdata.cn/course/course_introduction/1/

- 数据酷客 “**机器学习实践**” 课程 分类章节：

http://cookdata.cn/course/course_introduction/39/

数据科学导引

5602人 163人

机器学习实践

998人 61人

案例：使用感知机、逻辑回归和支持向量机进行中文新闻分类

基于中文新闻数据集，利用三种分类模型建立新闻自动分类器。

1. 使用 Python 和梯度下降法感知机、逻辑回归和支持向量机
2. 借助 Sklearn SGDClassifier类建立文本分类模型
3. 使用混淆矩阵、正确率、精度、召回率等指标进行分类模型评估

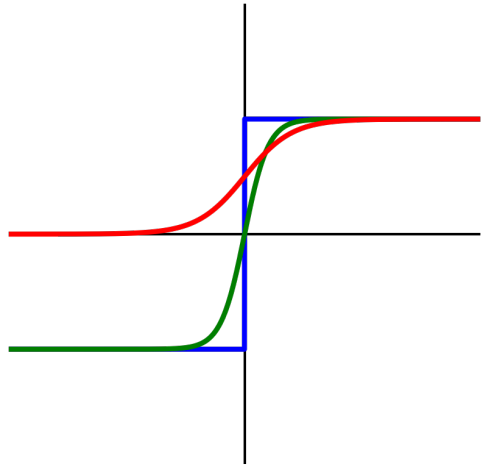
案例地址： http://cookdata.cn/note/view_static_note/83d69a6c88ff3eec2e5867da890b1957/

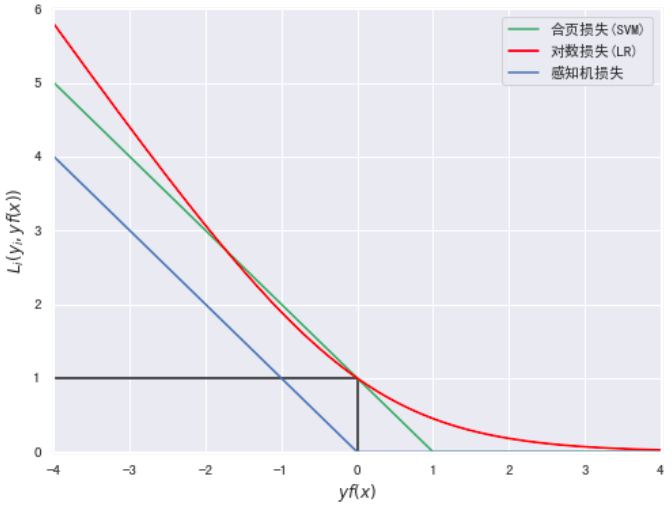
BIBDR 北京大数据研究院 | 博雅大数据学院 BOYA BIG DATA COLLEGE | 数据酷客 COOKDATA

数据酷客

数据科学人工智能

- 介绍了三种用回归办法解决分类问题的模型
- 感知机：
  - 关注误分类样本，将训练集样本分对即可
  - 是支持向量机、神经网络的基本模型
  - 只能应用于线性可分数据集
- 逻辑回归：
  - 使用 Logistic 函数赋予样本概率解释
  - 使用最大似然法求解，是一种线性分类模型
  - 为防止过度拟合，可在优化目标添加正则项
- 支持向量机：
  - 可以使用核技巧将低维数据转换到高维运算，保持低维的计算量
  - 如何选择核函数是一大困难





1. 逻辑回归和支持向量机分别如何处理多分类问题？

2. 如果训练集线性不可分（更常见的场景），应该怎么办？

3. 请尝试实现一个带 l_2 正则项的逻辑回归模型。

4. 除了本次课介绍的分类模型，还有哪些分类问题？你能写出他们的优化目标函数吗？

数据酷客

数据科学人工智能

加入数据酷客交流群