

机器学习笔记

Yuxuan Yuan & Lulu Cao

2022.2.21

目录

1 Lecture 1	4
2 Lecture 2	5
2.1 课堂回顾	5
2.2 投影矩阵 & Normal Equation	6
2.3 增加新的数据 (样本数 or 特征维度)	6
2.4 人工智能的流派	6
2.5 人脸识别	6
3 Lecture 3	8
4 Lecture 4	9
4.1 Example 1. Bernoulli	9
4.2 Example 2. Gaussian	10
4.3 Example 3. Linear Regression	10
4.4 Example 4. Logistic Regression	10
5 Lecture 5	11
5.1 Review: MLE / MAP / Bayesian	11
5.2 Logistic Regression(逻辑斯蒂回归)	12
5.3 Perceptron(感知机)	15
5.4 Generative Classification Model	16
5.5 作业	17
6 Lecture 6	18
6.1 朴素贝叶斯 NB	18
6.1.1 原理与模型	18
6.1.2 算法	18
6.1.3 小结: 产生式 $p(y x)$ vs. 判别式 $p(x,y)$	19
6.2 贝叶斯学习	19
7 Lecture 7	20
7.1 狄利克雷分布	20
7.1.1 二项分布	20
7.1.2 多项分布	20
7.1.3 贝塔分布	20
7.1.4 狄利克雷分布	21
7.1.5 共轭先验	21
7.2 MLE/MAP/Bayesian Learning: 多分类	21
7.3 Bernoulli Mixture Model	22
7.3.1 混合模型的应用	23

7.3.2	二维伯努利分布的混合模型	23
7.3.3	混合系数 π_k 的求解	24
7.4	EM 算法	25
8	Lecture 8	26
8.1	EM-principle	26
8.1.1	生成过程	26
8.1.2	混合模型似然函数	26
8.2	EM 算法	27
8.2.1	更新 π_k	28
8.2.2	更新 μ_k	28
8.2.3	更新 Σ_k	29
8.2.4	E 步——更新 q_{nk}	30
9	Lecture 9	32
9.1	EM-ELBO	32
9.2	EM for unsupervised task——Naive Bayes	33
9.3	EM+Bayesian	33
9.4	SeqModel+HMM	34
9.4.1	概率计算问题	35
10	Lecture 12	35
10.1	作业	35

1 Lecture 1

2022.2.21 第一节课是在 C103 上的，啥也没带 oh lambda 含义

2 Lecture 2

2022.2.28 yyx 忘了记录板书 oh

2.1 课堂回顾

机器学习 = lambda : 机器 = 函数; 学习 = 拟合

Machine Learning = LAMBDA. Loss, Algorithm, Model, BigData, Application.

BigData $D = \{(x_n, y_n)\}_{n=1}^N$, 其中 $x_n \in R^N$, $y \in R$

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_N^T \end{bmatrix}$$

Model

$$y = \mathbf{w}^T \mathbf{x}$$

Loss

$$\mathcal{L}_2(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^N (y_n - \mathbf{w}^T \mathbf{x}_n)^2$$

将上述式子矩阵化,

$$\begin{aligned} \mathcal{L}_2(\mathbf{w}) &= \frac{1}{N} (\mathbf{X}\mathbf{w} - \mathbf{y})^T (\mathbf{X}\mathbf{w} - \mathbf{y}) \\ &= \frac{1}{N} ((\mathbf{X}\mathbf{w})^T - \mathbf{y}^T) (\mathbf{X}\mathbf{w} - \mathbf{y}) \\ &= \frac{1}{N} (\mathbf{X}\mathbf{w})^T \mathbf{X}\mathbf{w} - \frac{1}{N} \mathbf{y}^T \mathbf{X}\mathbf{w} - \frac{1}{N} (\mathbf{X}\mathbf{w})^T \mathbf{y} + \frac{1}{N} \mathbf{y}^T \mathbf{y} \\ &= \frac{1}{N} \mathbf{w}^T \mathbf{X}^T \mathbf{X} \mathbf{w} - \frac{2}{N} \mathbf{w}^T \mathbf{X}^T \mathbf{y} + \frac{1}{N} \mathbf{y}^T \mathbf{y} \end{aligned}$$

Algorithm 对损失函数求导, 并令其等于 0

$$\begin{aligned} \frac{\partial \mathcal{L}_2}{\partial \mathbf{w}} &= \frac{2}{N} \mathbf{X}^T \mathbf{X} \mathbf{w} - \frac{2}{N} \mathbf{X}^T \mathbf{y} = 0 \\ \mathbf{X}^T \mathbf{X} \mathbf{w} &= \mathbf{X}^T \mathbf{y} \end{aligned}$$

矩阵求导相关

$f(w)$	$\frac{\partial f}{\partial w}$
$w^T \mathbf{x}$	\mathbf{x}
$\mathbf{x}^T w$	\mathbf{x}
$w^T w$	$2w$
$w^T \mathbf{C} w$	$2\mathbf{C} w$

从而,

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

2.2 投影矩阵 & Normal Equation

2.3 增加新的数据 (样本数 or 特征维度)

作为作业, 当 \mathbf{X} 增加一行 (样本数增加) 或者增加一列 (特征维度增加) 时, \mathbf{W} 如何变化, 写出更新后的 \mathbf{W} 和更新前的 \mathbf{W} 之间的增量表达式。

相关公式

- Sherman-Morrison 公式

$$(\mathbf{A} + \mathbf{u}\mathbf{v}^T)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^T\mathbf{A}^{-1}}{1 + \mathbf{v}^T\mathbf{A}^{-1}\mathbf{u}}$$

- 分块矩阵求逆 设 \mathbf{A} 是 $m \times m$ 可逆矩阵, \mathbf{B} 是 $m \times n$ 矩阵, \mathbf{C} 是 $n \times m$ 矩阵, \mathbf{D} 是 $n \times n$ 矩阵, $\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}$ 是 $n \times n$ 可逆矩阵, 则有

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{B}(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \\ -(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1} & (\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} \end{bmatrix}$$

2.4 人工智能的流派

- 类比主义: 核方法、SVM
- 连接主义: 神经网络
- 贝叶斯主义
- 符号主义: 决策树、专家系统
- 演化主义 (优化算法): 遗传算法
- 行为主义: 强化学习

2.5 人脸识别

设 D 为人脸的数据, $D = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$, 其中 $\mathbf{x}_n \in \mathbb{R}^N$, $y \in [1, 100]$, $y_i = \mathbf{w}^T \mathbf{x}_i + \epsilon$, $\epsilon \sim W(0, \sigma^2)$, $\sigma^2 = 1$, 则 $y_i \sim W(\mathbf{w}^T \mathbf{x}_i, 1)$

方差一样, 所以一样胖; 均值不一样, 所以 location 不同

找到一个 w , 使得 y_i 出现的概率最大, 即 $P(y_i | \mathbf{w}^T \mathbf{x}_i, 1) = \frac{1}{\sqrt{2\pi}} \exp -\frac{1}{2} \left(\frac{y_i - \mathbf{w}^T \mathbf{x}_i}{1} \right)^2$

$$L = p(\mathbf{y} | \mathbf{X}, \mathbf{w}, \sigma^2) = \prod_{n=1}^N p(y_n | \mathbf{x}_n, \mathbf{w}, \sigma^2) = \prod_{n=1}^N \mathcal{N}(\mathbf{w}^T \mathbf{x}_n, \sigma^2)$$

取对数, 有

$$\begin{aligned} \log L &= \sum_{n=1}^N \log \left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{1}{2\sigma^2} (y_n - f(\mathbf{x}_n; \mathbf{w}))^2 \right\} \right) \\ &= \sum_{n=1}^N \left(-\frac{1}{2} \log(2\pi) - \log \sigma - \frac{1}{2\sigma^2} (y_n - f(\mathbf{x}_n; \mathbf{w}))^2 \right) \\ &= -\frac{N}{2} \log 2\pi - N \log \sigma - \frac{1}{2\sigma^2} \sum_{n=1}^N (y_n - f(\mathbf{x}_n; \mathbf{w}))^2 \end{aligned}$$

$w^T \mathbf{x}_n$ 替换模型中的决定性部分, 对数似然表达式就呈现如下的形式: $\log L = -\frac{N}{2} \log 2\pi - N \log \sigma - \frac{1}{2\sigma^2} \sum_{n=1}^N (y_n - w^T \mathbf{x}_n)^2$ 得到的最小二乘解, 通过求导数、使其等于零以及求解拐点的方法, 类似于 1.1.4 节所述的方式。对于 w (注意, $w^T \mathbf{x}_n = \mathbf{x}_n^T w$),

最小二乘法 and 极大似然估计是等价的

3 Lecture 3

2022.3.7

Outline

1. Least Square: MLE
 - 1.1 SGD
 - 1.2 Probabilistic Graph Representation
 - 1.3 $\mathbb{E}[\hat{w}] / cov[\hat{w}]$
 - 1.4 bias / variance
2. Revisted LS: Curve Fitting
 - 2.1 Model Selection
 - 2.2 Overfitting
3. How to solve overfitting
 - 3.1 Regularization (MAP)
 - 3.2 Bayesian Learning

4 Lecture 4

2022.3.14

使用 MLE/MAP/Bayes Learning 三种方法来求解参数，通过 4 个例子来加深。Example 4 没讲完。其中 Example 3 的三种解法需要自己课后补充。

tips: to be added 三个图对应生成式模型、判别式模型、分布计算；一些前置 discrete continuous 的概率表变量分布；有积分的地方和是求谁的期望的地方，

Outline

- MLE / MAP / Bayesian
- Example 1. Bernoulli
- Example 2. Gaussian
- Example 3. Linear Regression
- Example 4. Logistic Regression

问题：

- 共轭分布是什么意思
- β 分布

4.1 Example 1. Bernoulli

Given dataset $D = \{< x_i > \}_{i=1}^n, x_i \in \{0, 1\}$ x_i 服从 Bernoulli 分布， $P(x_i = 1) = \theta$ ；再假设 n_1 表示 D 中 $x_i = 1$ 的个数；

Method

1. MLE

$$\begin{aligned} P(D|\theta) &= \prod_{i=1}^n P(x_i; \theta) \\ &= \prod_{i=1}^n \theta^{x_i} (1 - \theta)^{1-x_i} \\ &= \theta^{\sum_i x_i} (1 - \theta)^{n - \sum_i x_i} \\ &= \theta^{n_1} (1 - \theta)^{n - n_1} \end{aligned}$$

取对数以便计算，从而

$$\arg \max_{\theta} \ln P(D|\theta) = \arg \max_{\theta} n_1 \ln \theta + (n - n_1) \ln(1 - \theta)$$

对 θ 求导

$$\frac{n_1}{\theta} - \frac{n - n_1}{1 - \theta} = 0$$

得到 $\theta = \frac{n_1}{n}$

2. MAP

Beta 分布 $Beta(\theta | a, b)$ $P(\theta|D) \propto P()$

贝塔分布 (Beta Distribution) 是一个作为伯努利分布和二项式分布的共轭先验分布的密度函数。在概率论中, 贝塔分布, 也称 B 分布, 是指一组定义在 (0,1) 区间的连续概率分布。

3. Bayesian

4.2 Example 2. Gaussian

Method

1. MLE
2. MAP
3. Bayesian

4.3 Example 3. Linear Regression

Given dataset $D = \{< x_i, y_i > \}_{i=1}^n, y_i \in \mathbb{R}$, 假设 $X \sim \mathcal{N}(\theta, \sigma^2)$, 则有 $P(x; \theta) = \frac{1}{\sigma\sqrt{2\pi}} \exp\{-\frac{(x-\theta)^2}{2\sigma^2}\}$

Method

1. MLE
2. MAP
3. Bayesian

4.4 Example 4. Logistic Regression

Method

1. MLE
2. MAP 3. Bayesian

计算步骤 (流程) 总结

1. $P(D|\theta)$
2. $P(\theta)$
3. $P(\theta|D) \propto P(D|\theta)P(\theta)$
4. $\theta_{bayes} = \mathbb{E}[\theta|D]$
5. inference: $P(y_{new}|D, x_{new}) = \int p(y_{new}|x_{new}, \theta)p(\theta|D)$

其中 $D = (X, Y)$

5 Lecture 5

2022.3.21

Outline

1. Review: MLE / MAP / Bayesian Estimation
2. Logistic Regression
3. Perceptron
4. Generative Classification Model

x is continuous (GDA)

x is discrete (NB)

.....
Data: $D = \langle x_i, y_i \rangle_{i=1}^n, x_i \in \mathbb{R}^d, y_i \in \{0, 1\}, D = (X, Y)$

Model: $P(x, y; \Theta)$ (生成式) ; $P(y|x; \Theta)$ (判别式) ; $P(x_i; \Theta)$ 无监督

Inference: Given x, D ; Output $y, P(y|x, D) = ?$

Learning: Given D ; Output $\Theta, P(D|\Theta), \Theta_{Bayes} = \mathbb{E}[\Theta|D]$

5.1 Review: MLE / MAP / Bayesian

三种方法，即 Learning 的三种方法

Learning

1. MLE

$$\begin{aligned}\hat{\Theta}_{MLE} &= \arg \max_{\Theta} P(D|\Theta) \\ &= \arg \max_{\Theta} \sum_{i=1}^n \ln P(blank; \Theta)\end{aligned}$$

其中, *blank* 可以填入 1) x_i, y_i ; 2) $y_i|x_i$; 3) x_i ; 即三种模型都可用 MLE 求解

2. MAP

$$\begin{aligned}\hat{\Theta}_{MAP} &= \arg \max_{\Theta} P(\Theta|D) \propto P(D|\Theta)P(\Theta) \\ &= \arg \max_{\Theta} \ln P(blank; \Theta) + \ln P(\Theta)\end{aligned}$$

其中, $P(\Theta)$ 是先验, $P(D|\Theta)$ 是似然, 等式的最后 $\ln P(blank$ 为数据项 (Data Term), $\ln P(\Theta)$ 为正则项 (Regularization Term, 或平滑项 (Smooth Term))。

* 当 $P(\Theta)$ 是均匀分布时, $MLE = MAP$ 。

3. Bayesian Estimation

(前提是 $P(\Theta|D)$ 即后验分布已知)

$$\hat{\Theta}_{Bayes} = \mathbb{E}[\Theta|D] = E_{\theta \sim P(\cdot|D)}[\Theta] = \int \Theta p(\Theta|D) d\Theta$$

Inference

利用上述三种方法做参数估计后的推理方法

1. MLE

Given $\hat{\Theta}_{MLE}, x, D$, output $P(y|x; \hat{\Theta}_{MLE})$.

举例，在 Logistic Regression 中， $P(y=1|x; \hat{\Theta}_{MLE}) = \sigma(\hat{\Theta}_{MLE}^T x)$

2. MAP

Given $\hat{\Theta}_{MAP}, x, D$, output $P(y|x; \hat{\Theta}_{MAP})$.

举例，在 Logistic Regression 中， $P(y=1|x; \hat{\Theta}_{MAP}) = \sigma(\hat{\Theta}_{MAP}^T x)$

3. Bayes Estimation

Given $x, D, P(\Theta|D)$, output $P(y|x; \hat{\Theta}_{MAP})$.

$$\begin{aligned} P(y|x; D) &= \int p(y, \Theta|x; D) d\Theta \\ &= \int p(\Theta|x; D) p(y|\Theta, x; D) d\Theta \end{aligned}$$

等式最后的 $p(\Theta|x; D)$ 为后验分布， $p(y|\Theta, x; D)$ 为模型。能这么做的前提是假设 x_{new} 与 Θ 无关。

如果求 $y=1$ ，有

$$\begin{aligned} p(y|\Theta, x; D) &= \int p(\Theta|D) \sigma(\Theta^T x) dx (= \mathbb{E}_{\Theta \sim P(\cdot|D)} [\sigma(\Theta^T x)]) \\ &\approx \frac{1}{\mathcal{L}} \sum_{i=1}^{\mathcal{L}} \sigma((\Theta^{(i)})^T x) \end{aligned}$$

其中有假设 x_{new} 与 Θ 相互独立； $\Theta^{(i)} \sim P(\Theta|D)$ ， $i=1, \dots, \mathcal{L}$ ，上面公式使用了抽样技术 (Sampling) 来求期望。

5.2 Logistic Regression(逻辑斯蒂回归)

补点图概率图模型（可观测量用灰色阴影）

Learning

1. MLE

$$P(y_i|x_i; \Theta) = \sigma(\Theta^T x_i)^{y_i} (1 - \sigma(\Theta^T x_i))^{1-y_i}$$

$$\begin{aligned} \ln P(D|\Theta) &= \sum_{i=1}^n \{y_i \ln \sigma_i + (1 - y_i) \ln(1 - \sigma_i)\} \\ &= \mathcal{L}_D(\Theta) \end{aligned}$$

其中 $\sigma_i = \sigma(a_i) = \sigma(\Theta^T x_i) = \frac{1}{1+e^{-\Theta^T x_i}}$ ，从而

$$\Theta^* = \arg \max_{\Theta} \mathcal{L}_D(\Theta)$$

到这里，求解 Θ^* 方法有求偏导数并等于 0 来计算解析解，但无法做到，原因如下，利用链式法则算一下偏导数

$$\begin{aligned}\nabla_{\Theta} \mathcal{L}_D(\Theta) &= \frac{\partial \mathcal{L}_D(\Theta)}{\partial \Theta} \\ &= \frac{\partial \mathcal{L}_D(\Theta)}{\partial \sigma_i} \frac{\partial \sigma_i}{\partial a_i} \frac{\partial a_i}{\partial \Theta} \\ &= \sum_{i=1}^n \left(\frac{y_i}{\sigma_i} - \frac{1-y_i}{1-\sigma_i} \right) \sigma_i (1-\sigma_i) x_i \\ &= \sum_{i=1}^n (y_i - \sigma_i) x_i\end{aligned}$$

试试，几乎无法求得解析解。

第二种方法，尝试二阶导数 $\nabla_{\Theta}(\nabla_{\Theta} \mathcal{L}_D(\Theta))$ ，令 $\nabla_{\Theta} \mathcal{L}_D(\Theta) = g$,

Algorithm 1 A1: Gradient Ascent for Logistic Regression

Input: X, Y ;

Output: Θ ;

1: **init:** $\Theta \sim \mathcal{N}(0, \alpha^{-1}I), \epsilon$

2: **Loop:**

3: $g = X^T(Y - \sigma)$

4: $\Theta^{t+1} := \Theta^t + \eta g$

5: **until** $\|\Theta^{t+1} - \Theta^t\| \leq \epsilon$

6: **return** Θ ;

Hessian Matrix of the Loss Function $\mathcal{L}_D(\Theta)$

Newton $\Theta^{t+1} := \Theta^t + H^{-1}g$ todo

2.MAP

$$P(\Theta) \sim \mathcal{N}(0, \alpha^{-1}I)$$

$$P(D|\Theta) = \prod_{i=1}^n \sigma_i^{y_i} (1 - \sigma_i)^{1-y_i}$$

$$P(\Theta|D) \propto \mathcal{N}(0, \alpha^{-1}I) \prod_{i=1}^n \sigma_i^{y_i} (1 - \sigma_i)^{1-y_i}$$

根据 MAP，有

$$\begin{aligned}\Theta^* &= \arg \max_{\Theta} \ln P(\Theta|D) \\ &= \arg \max_{\Theta} \ln P(D|\Theta) + \ln P(\Theta) \\ &= \arg \max_{\Theta} \sum_{i=1}^n y_i \ln(\sigma_i) + (1 - y_i) \ln(1 - \sigma_i) - \frac{1}{2}(\Theta^{-1} \Sigma^{-1} \Theta)\end{aligned}$$

则

$$\begin{aligned}\nabla_{\Theta} \mathcal{L}_D(\Theta) &= \frac{\partial \mathcal{L}_D(\Theta)}{\partial \Theta} \\ &= \sum_{i=1}^n (y_i - \sigma_i) x_i - \Theta \Sigma^{-1}\end{aligned}$$

这里定 $\alpha^{-1} \mathbf{I} = \Sigma$, 不影响结果。

类似算法A1, 写个A2

Algorithm 2 A2: (MAP) Gradient Ascent for Logistic Regression

Input: X, Y;

Output: Θ ;

1: init: $\Theta \sim \mathcal{N}(0, \alpha^{-1} \mathbf{I}), \epsilon$

2: **Loop:**

3: $g = \mathbf{X}^T(\mathbf{Y} - \sigma) - \Theta^t \Sigma^{-1}$

4: $\Theta^{t+1} := \Theta^t + \eta g$

5: **until** $\|\Theta^{t+1} - \Theta^t\| \leq \epsilon$

6: **return** Θ ;

3. Bayesian Estimation

贝叶斯估计: $\mathbb{E}[\Theta|D]$

$P(y = 1|x, D) =$ 类似算法A1, 写个A3用于贝叶斯推理

Inference 后验预测分布为

$$p(y = 1|x_{new}, D) = \int p(y = 1|x_{new}, \Theta) p(\Theta|x, D) d\Theta$$

但是积分难以处理, 采用近似处理, 同时还有假设 x_{new} 与 Θ 无关, 有

$$\begin{aligned}p(y = 1|x_{new}, D) &= \int p(y = 1|x_{new}, \Theta, D) p(\Theta|D) d\Theta \\ &= \int \sigma(\Theta^T x_{new}) p(\Theta|D) d\Theta \\ &\approx \frac{1}{\mathcal{L}} \sum_{i=1}^{\mathcal{L}} \sigma((\Theta^{(i)})^T x_{new})\end{aligned}$$

其中, $\Theta^{(i)} \propto p(\Theta|D) (i = 1, \dots, \mathcal{L})$

Algorithm 3 A3: Inference after Bayesian Estimation

Input: x_{new}, D ;

Output: y ;

1: init: $\Theta^{(i)} \propto p(\Theta|D) (i = 1, \dots, \mathcal{L})$

2: $P(y = 1|x_{new}, D) = \frac{1}{\mathcal{L}} \sum_{i=1}^{\mathcal{L}} \sigma((\Theta^{(i)})^T x_{new})$

3: $P(y = 0|x_{new}, D) = \frac{1}{\mathcal{L}} \sum_{i=1}^{\mathcal{L}} (1 - \sigma((\Theta^{(i)})^T x_{new}))$

4: **return** 1 if $P(y = 1|x_{new}, D) > P(y = 0|x_{new}, D)$ else 0;

5.3 Perceptron(感知机)

写个历史表

- 1943 M-P —神经元模型
- 1957 Rosenblatt —Perceptron;
- 1982 BP 算法 (Computational Graph 计算图)
- 2006 Hinton —预训练 pretrain、DNN
- 2012 AlexNet.... 图像不太懂...

Model

计算方法: $y = \text{sgn}(\mathbf{W}^T x + b)$

其中;

$$\begin{cases} y = +1, \mathbf{W}^T x + b > 0, \\ y = -1, \mathbf{W}^T x + b < 0 \end{cases}$$

Loss 计算

$$\mathcal{L}_D(W) = \sum_{x_i, y_i \in M} \frac{|y_i(\mathbf{W}^T x_i + b)|}{\|\mathbf{W}\|}$$

其中 $M = \{(x_i, y_i) | y_i(\mathbf{W}^T x_i + b) < 0\}$, 可令 $\|\mathbf{W}\| = 1$, 上式可写成,

$$\mathcal{L}_D(W) = \sum_{i=1}^n \max\{0, -y_i(\mathbf{W}^T x_i + b)\}$$

* 当 \max 中的第二项变为 $1 - y_i(\mathbf{W}^T x_i + b)$ 时, 损失函数变成 SVM 的损失函数。

这里求一下偏导数, 对于单个样本 (x_i, y_i)

$$\frac{\partial(-y_i(\mathbf{W}^T x_i + b))}{\partial W} = -y_i x_i$$

也可以写成一个算法形式

Algorithm 4 A4: Perceptron GD

Input: X, Y ;

Output: $\Theta = \{W, b\}$;

1: **init:** W, b

2: **Loop:**

3: **if** $y_i(\mathbf{W}^T x_i + b) < 0$ **then**

4: $W^{t+1} \leftarrow W^t - \eta \nabla_W \mathcal{L}_D(W)$

5: 等价于 $W^{t+1} \leftarrow W^t + \eta y_i x_i$

6: **end if**

7: **until** 训练集中没有误分类点

8: **return** Θ ;

5.4 Generative Classification Model

y 是离散的

x 是连续的

Given $\{(x_i, y_i)\}_{i=1}^n = D = (X, Y), x \in \mathbb{R}^d$ 。假设 $y_i = 0, 1$ ，即 y_i 服从 Bernoulli 分布， x_i 服从高斯分布 $x_i|y_i \sim \mathcal{N}(\mu_k, \Sigma_k)$ ，model 描述如下

$$\begin{aligned} P(x_i, y_i; \Theta) &= P(y_i; \Theta)P(x_i|y_i; \Theta) \\ &= \text{Bernoulli}(y_i|p) \prod_{y=0}^1 \mathcal{N}(x_i|\mu_k, \Sigma_k)^{\mathbf{1}\{k=y\}} \end{aligned}$$

关于 $P(x_i|y_i; \Theta)$ 部分细述如下，

$$\begin{aligned} P(x|y=1; \Theta) &= \mathcal{N}(x|\mu_1, \Sigma_1) \\ P(x|y=0; \Theta) &= \mathcal{N}(x|\mu_0, \Sigma_0) \end{aligned}$$

从而有

$$P(x|y; \Theta) = \mathcal{N}(x|\mu_1, \Sigma_1)^{\mathbf{1}\{y=1\}} \mathcal{N}(x|\mu_0, \Sigma_0)^{\mathbf{1}\{y=0\}}$$

记 $\Theta = (p, \mu_0, \Sigma_0, \mu_1, \Sigma_1)$ ，则有

$$\begin{aligned} P(D; \Theta) &= \prod_{i=1}^n P(x_i, y_i; \Theta) \\ &= \prod_{i=1}^n (p^{y_i} (1-p)^{1-y_i}) \mathcal{N}(x_i|\mu_1, \Sigma_1)^{\mathbf{1}\{y_i=1\}} \mathcal{N}(x_i|\mu_0, \Sigma_0)^{\mathbf{1}\{y_i=0\}} \end{aligned}$$

取对数，有

$$\ln P(D; \Theta) = \sum_{i=1}^n y_i \ln p + (1-y_i) \ln(1-p) + \sum_{i=1}^n \mathbf{1}\{y=1\} \ln \mathcal{N}(x_i|\mu_1, \Sigma_1) + \sum_{i=1}^n \mathbf{1}\{y=0\} \ln \mathcal{N}(x_i|\mu_0, \Sigma_0)$$

Learning time !!!!! **MLE**

根据 $\hat{\Theta}_{MLE} = \arg \max_{\Theta} \ln P(D; \Theta)$ ，假设 Σ_0, Σ_1 已知，设 $M_1 = \{(x_i, y_i)|y_i = 1\}, M_0 = \{(x_i, y_i)|y_i = 0\}$ ，求 p, μ_0, μ_1 。

$$\begin{aligned} \mathcal{L}_D(\Theta) &= \ln P(D; \Theta) \\ \ln \mathcal{N}(x|\mu, \Sigma) &= -\frac{d}{2} \ln(2\pi) - \frac{1}{2} \ln |\Sigma| - \frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \\ \frac{\partial \mathcal{L}_D(\Theta)}{\partial p} &= \sum_{i=1}^n w^n \left(\frac{y_i}{p} - \frac{1-y_i}{1-p} \right) = 0 \\ \frac{\partial \mathcal{L}_D(\Theta)}{\partial \mu_0} &= \sum_{(x_i, y_i) \in M_0} \Sigma^{-1} (x_i - \mu_0) = 0 \\ \frac{\partial \mathcal{L}_D(\Theta)}{\partial \mu_1} &= \sum_{(x_i, y_i) \in M_1} \Sigma^{-1} (x_i - \mu_1) = 0 \end{aligned}$$

解得

$$p = \frac{\sum_{i=1}^n y_i}{n}$$

$$\mu_0 = \frac{\sum_{(x_i, y_i) \in M_0} x_i}{|M_0|}$$

$$\mu_1 = \frac{\sum_{(x_i, y_i) \in M_1} x_i}{|M_1|}$$

Infering time !!!

Given $x, y = ?$, $p, \mu_0, \mu_1, \Sigma_0, \Sigma_1$ are known.

$$P(y = 1|x) \propto P(y = 1)P(x|y = 1)$$

$$= p\mathcal{N}(x|\mu_1, \Sigma_1)$$

$$P(y = 0|x) \propto P(y = 0)P(x|y = 0)$$

$$= (1 - p)\mathcal{N}(x|\mu_0, \Sigma_0)$$

若 $P(y = 1|x) > P(y = 0|x)$, 则 $y = 1$ 。

5.5 作业

$$g(x) = \ln \frac{P(y = 1|x)}{P(y = 0|x)}$$

$$= \ln \frac{p\mathcal{N}(x|\mu_1, \Sigma_1)}{(1 - p)\mathcal{N}(x|\mu_0, \Sigma_0)}$$

设 $\Sigma_1 = \Sigma_0$, 则 $g(x)$ 是线性平面, 可写作 $g(x) = w^T x + b$, 即 Gaussian Discriminative Analysis(GDA), 求 w, b

$$g(x) = \ln \frac{p}{1 - p} + \left(\frac{1}{2} ((x - \mu_0)^T \Sigma^{-1} (x - \mu_0) - (x - \mu_1)^T \Sigma^{-1} (x - \mu_1)) \right)$$

$$= \ln \frac{p}{1 - p} + \frac{1}{2} ((x^T - \mu_0^T) \Sigma^{-1} (x - \mu_0) - (x^T - \mu_1^T) \Sigma^{-1} (x - \mu_1))$$

$$= \ln \frac{p}{1 - p} + \frac{1}{2} (x^T \Sigma^{-1} x - x^T \Sigma^{-1} \mu_0 - \mu_0^T \Sigma^{-1} x + \mu_0^T \Sigma^{-1} \mu_0 + x^T \Sigma^{-1} x + x^T \Sigma^{-1} \mu_1 + \mu_1^T \Sigma^{-1} x - \mu_1^T \Sigma^{-1} \mu_1)$$

$$= \ln \frac{p}{1 - p} + \frac{1}{2} (2(\mu_1^T \Sigma^{-1} x - \mu_0^T \Sigma^{-1} x) + \mu_0^T \Sigma^{-1} \mu_0 - \mu_1^T \Sigma^{-1} \mu_1)$$

$$= (\mu_1 - \mu_0)^T \Sigma^{-1} x + \frac{1}{2} \mu_0^T \Sigma^{-1} \mu_0 - \frac{1}{2} \mu_1^T \Sigma^{-1} \mu_1 + \ln p - \ln(1 - p)$$

即 (结合: w 记得算一下转置, Σ 是对称矩阵等内容)

$$w = \Sigma^{-1}(\mu_1 - \mu_0)$$

$$b = \frac{1}{2} \mu_0^T \Sigma^{-1} \mu_0 - \frac{1}{2} \mu_1^T \Sigma^{-1} \mu_1 + \ln p - \ln(1 - p)$$

x 是离散的

6 Lecture 6

2022.3.28

Outline

1. Generative Classification Model : $p(x, y)$, y is discrete

x is continuous (高斯判别分析 GDA)

x is discrete (朴素贝叶斯 NB)

2. 贝叶斯学习

6.1 朴素贝叶斯 NB

Problem: $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, 其中 $x_i \in R^d$, $y_i \in \{0, 1\}$, $\langle X, Y \rangle \sim P$

6.1.1 原理与模型

假设: 各个维度独立; 每个维度的取值为 0 或 1

$$P(X_1, \dots, X_d | Y) = \prod_{j=1}^d P(X_j | Y)$$

条件独立: $P(X|Y, Z) = P(X|Z) = X \perp Y | Z$

$$P(X|Y) = P(X_1, X_2 | Y) = P(X_1 | X_2, Y) P(X_2 | Y) = P(X_1 | Y) P(X_2 | Y)$$

模型:

$$\begin{aligned} y^* &= \arg \max_{y_k} \frac{P(y_k) P(X | y_k)}{P(X)} \\ &= \arg \max_{y_k} P(Y = y_k) \prod_{j=1}^d P(X_j = x_j | Y = y_k) \end{aligned}$$

6.1.2 算法

1. MLE

其实就是计算频率

2. 零频率问题

解决方案: 拉普拉斯平滑

$$P(w | c) = \frac{\text{num}(w, c) + \varepsilon}{\text{num}(c) + k\varepsilon}$$

PS: 属性缺失不用管

6.1.3 小结：产生式 $p(y|x)$ vs. 判别式 $p(x,y)$

1. 产生式模型有其对应的判别式模型，这样的组合叫做“产生式-判别式”对；(e.g. 根据 $p(x,y)$ 得到 GDA 的分界面)；
2. 判别式模型不一定有其对应的产生式模型；
3. 训练样本多的情况下，采用判别式；
4. 训练样本少的情况下，选择产生式。

6.2 贝叶斯学习

1. MLE/MAP/贝叶斯估计
2. 贝叶斯估计：将先验信息集成到模型推理中 $p(x, X, \theta)$

作业

朴素贝叶斯的贝叶斯估计

输入：训练集 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ ，其中 $x_i = (x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(n)})^T$ ，第 i 个样本的第 j 个特征 $x_i^{(j)}$ 可能的取值集合为 $\{a_{j1}, a_{j2}, \dots, a_{jS_j}\}$ ，即某一特征的取值为有限集合。 $y_i \in \{c_1, c_2, \dots, c_K\}$ ，即分类结果为有限集合（共 K 个）。

先验概率的贝叶斯估计为：

$$P(Y = c_k) = \frac{\sum_{i=1}^N I(y_i = c_k) + \lambda}{N + K\lambda}, k = 1, 2, \dots, K$$

条件概率的贝叶斯估计为：

$$P(X^{(j)} = a_{jl} | Y = c_k) = \frac{\sum_{i=1}^N I(x_i^{(j)} = a_{jl}, y_i = c_k) + \lambda}{\sum_{i=1}^N I(y_i = c_k) + S_j \lambda}$$

朴素贝叶斯的贝叶斯估计为：

$$y = \arg \max_{c_k} P(Y = c_k) \prod_{j=1}^n P(X^{(j)} = x_j | Y = c_k)$$

7 Lecture 7

2022.4.4

Outline

1. 狄利克雷分布
2. MLE/MAP/Bayesian Learning: 多分类
3. Bernoulli Mixture Model
4. EM 算法
 - EM-principle
 - EM-convergence
 - EM for GMM
 - EM for Topic Model

7.1 狄利克雷分布

7.1.1 二项分布

二项分布用以描述 n 次独立的伯努利实验中有 m 次成功的概率:

$$P(X = m) = \binom{n}{m} p^m (1-p)^{n-m}$$

当试验的次数 n 为 1 时, 二项分布变成伯努利分布。

7.1.2 多项分布

多项分布是一种多元离散随机变量的概率分布, 是二项分布的扩展。假设重复进行 n 次独立随机试验, 每次试验可能出现的结果有 k 种, 第 i 种结果出现的概率为 p_i , 第 i 种结果出现的次数为 n_i 。如果用随机变量 $X = (X_1, X_2, \dots, X_k)$ 表示试验所有可能结果的次数, 其中 X_i 表示第 i 种结果出现的次数, 那么随机变量 X 服从多项分布。

$$\begin{aligned} P(X_1 = n_1, \dots, X_k = n_k) &= \frac{n!}{n_1! \dots n_k!} p_1^{n_1} \dots p_k^{n_k}, \sum_{i=1}^k n_i = n \\ &= n! \prod_{i=1}^k \frac{p_i^{n_i}}{n_i!}, \sum_{i=1}^k n_i = n \end{aligned}$$

当试验的次数 n 为 1 时, 多项分布变成类别分布。类别分布表示试验可能出现的 k 种结果的概率。

7.1.3 贝塔分布

贝塔分布是关于连续变量 $x \in [0, 1]$ 的概率分布, 它由两个参数 $a > 0$ 和 $b > 0$ 确定:

$$\text{Beta}(x | a, b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} x^{a-1} (1-x)^{b-1} = \frac{1}{B(a, b)} x^{a-1} (1-x)^{b-1}$$

其中, $\Gamma(s)$ 为 Gamma 函数:

$$\Gamma(s) = \int_0^{+\infty} t^{s-1} e^{-t} dt$$

$B(a, b)$ 为 Beta 函数, 用来做归一化:

$$B(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$$

7.1.4 狄利克雷分布

狄利克雷分布是一种多元连续随机变量的概率分布, 是贝塔分布的扩展。在贝叶斯学习中, 狄利克雷分布常作为多项分布的先验分布使用。狄利克雷分布是关于一组 d 个连续变量 $x_i \in [0, 1]$ 的概率分布, $\sum_i x_i = 1_0$ 。令 $\mu = (\mu_1, \mu_2, \dots, \mu_d)$, 参数 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_d)$, 其中 $\alpha_i > 0$ 且 $\hat{\alpha} = \sum_i \alpha_i$ 。

$$\text{Dir}(x | \alpha) = \frac{\Gamma(\sum_{i=1}^d \alpha_i)}{\prod_{i=1}^d \Gamma(\alpha_i)} \prod_{i=1}^d x_i^{\alpha_i-1}$$

当 $d = 2$ 时, 狄利克雷分布退化为贝塔分布。

7.1.5 共轭先验

贝叶斯学习中常使用共轭分布。如果后验分布与先验分布属于同类, 则先验分布与后验分布称为 **共轭分布**, 先验分布称为共轭先验。

如果多项分布的先验分布是狄利克雷分布, 则其后验分布也为狄利克雷分布, 两者构成共轭分布。作为先验分布的狄利克雷分布的参数又称为超参数。使用共轭分布的好处是便于从先验分布计算后验分布。

7.2 MLE/MAP/Bayesian Learning: 多分类

1. 二分类 $X \in \{0, 1\}$, $P(X = 0) = \prod_{k=0}^1 \theta_k^{I(x=k)}$
2. 多分类 $X \in \{1, \dots, K\}$, $P(X = x) = \prod_{k=1}^K \theta_k^{I(x=k)}$ 查表法

Learning: 给定 $D = \{x_i\}_{i=1}^n$ 求 θ

- 有目标 goal: 损失函数

MLE

$$P(D; \theta) = \prod_{i=1}^n \prod_{k=1}^6 \theta_k^{I(x=k)} = \prod_{k=1}^6 \theta_k^{\sum_{i=1}^n I(x=k)}$$

$$\ln P(D; \theta) = \sum_{k=1}^6 n_k \ln \theta_k$$

求该数据集出现的概率最大时, θ 的取值

$$\frac{\partial \ln P(D; \theta)}{\partial \theta_k}$$

MAP: $\theta \sim P(\theta | \alpha)$

$$\theta \sim P(\theta|D) = P(\theta)P(D|\theta)$$

$$\theta^* = \arg \max_{\theta} P(\theta|D)$$

贝叶斯学习

MAP 得到 $P(\theta|D)$

$$\forall x \ P(x|D) = \int P(x, \theta|D) d\theta = \int P(\theta|D) P(x|\theta) d\theta = \int \theta_k P(\theta|D)$$

- 有方法 Algorithm: 使损失函数达到最小的算法

Inference: 已知 θ , 根据 \mathbf{x} , 求 \mathbf{y}

7.3 Bernoulli Mixture Model

有限混合模型 (Finite Mixture Model) 是一个可以用来表示在总体分布中含有 K 个子分布的概率模型。子分部可以是各种经典的分布模型, 包括高斯模型, 伯努利模型, 多项式模型等。

首先在 D 维空间中定义由 K 个子分布组成的混合模型通用式:

$$P(\mathbf{x} | \Theta, \boldsymbol{\pi}) = \sum_{k=1}^K \pi_k P(\mathbf{x} | \boldsymbol{\theta}_k)$$

其中 π_k 为第 k 个子分布的混合系数, 也称权重, 且满足 $\sum_{k=1}^K \pi_k = 1$; $\Theta = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_K\}$ 。

$\Omega = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ 为一个包含 N 个样本的数据集, 且所有样本满足独立同分布。

对于混合模型, 通常采用极大似然法 (Maximum likelihood) 来估算参数 Θ , 因此我们建立如下混合模型的似然函数, 并希望将其最大化,

$$L(\Omega | \Theta, \boldsymbol{\pi}) = \prod_{n=1}^N \sum_{k=1}^K \pi_k P(\mathbf{x}_n | \boldsymbol{\theta}_k)$$

为便于计算, 将上式转化为对数似然函数:

$$\log L(\Omega | \Theta, \boldsymbol{\pi}) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k P(\mathbf{x}_n | \boldsymbol{\theta}_k)$$

一点预备知识: 利用贝叶斯定理推导出后验概率 (posterior probability) $\gamma(z_{nk})$:

$$\begin{aligned} \gamma(z_{nk}) = P(z_k = 1 | \mathbf{x}_n) &= \frac{P(\mathbf{x}_n | z_k = 1) P(z_k = 1)}{\sum_{k=1}^K P(\mathbf{x}_n | z_k = 1) P(z_k = 1)} \\ &= \frac{\pi_k P(\mathbf{x}_n | \boldsymbol{\theta}_k)}{\sum_{k=1}^K \pi_k P(\mathbf{x}_n | \boldsymbol{\theta}_k)} \end{aligned}$$

$\gamma(z_{nk})$ 表示在已知观测样本 \mathbf{x}_n 的条件下, 该样本来自于第 k 个子分布 ($z_k = 1$) 的概率。

7.3.1 混合模型的应用

文档分类

$$\mathbf{X}_{n \times d} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_n^T \end{bmatrix}$$

设每一行，即每个样本代表一个文档；每一列代表一个单词出现的次数

$$P(y = k) = \prod_{k=1}^K \theta_k^{I(y=k)}$$

$$P(x_j | y_i) = \prod_{i=1}^K \prod_{j=1}^V \Psi_{vk}^{I(y_i=k \text{ and } x_j=v)}$$

$$P(D; \psi) = \prod P(x_i, y_i)$$

$$= \prod P(y_i) P(x_i | y_i)$$

$$= \prod_{i=1}^n \left(\prod_{k=1}^K \theta_k^{I(y=k)} \right) \left(\prod_{i=1}^K \prod_{j=1}^V \Psi_{vk}^{I(y_i=k \text{ and } x_j=v)} \right)$$

$$\begin{aligned} \log P(D | \psi) &= \sum_{i=1}^n \left(\sum_{k=1}^K I(y = k) \log \theta_k \right) + \sum_{i=1}^n \left(\sum_{i=1}^K \sum_{j=1}^V I(y_i = k \text{ and } x_j = v) \log \Psi_{vk} \right) \\ &= \sum_{k=1}^K c_k \log \theta_k + \sum_{i=1}^K \sum_{j=1}^V c_{k,v} \log \Psi_{vk} \end{aligned}$$

求导即可得 θ_k, ψ_{vk}

.....

抛硬币

y	A	A	B	B	B	A	A	A	B	B
x	1	0	0	1	1	0	0	1	0	1

$$P(A|) = \frac{P(A)P(|A)}{P()} = \frac{P(A)P(|A)}{P(A)P(|A) + P(B)P(|B)}$$

7.3.2 二维伯努利分布的混合模型

二维伯努利分布是关于二维布尔向量 $\mathbf{x} = [x_1, x_2]$ 的概率分布，其中 $x_1, x_2 \in \{0, 1\}$ ，且满足 $x_1 + x_2 = 1$ ，即 x_1, x_2 中只有一个为 1；设参数向量 $\boldsymbol{\theta} = [\theta_1, \theta_2]$ ， $\theta_1, \theta_2 \in [0, 1]$ ，分别表示 $x_1 = 1$ 以及 $x_2 = 1$ 的概率，且满足 $\theta_1 + \theta_2 = 1$ 。其概率分布函数为：

$$\begin{aligned} P(\mathbf{x} | \boldsymbol{\theta}) &= \theta_1^{x_1} \theta_2^{x_2} \\ &= \theta_1^{x_1} (1 - \theta_1)^{1-x_1} \end{aligned}$$

由此可见，二维伯努利就是我们熟知的伯努利分布，这里称其为二维是为了与后面介绍的多维伯努利相一致。二维伯努利最简单的应用就是抛硬币问题，一个硬币共有两个面，每次实验只会出现在一个面朝上。

二维伯努利混合模型的对数似然函数如下所示：

$$\log L(\Omega | \Theta, \pi) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \theta_{k,1}^{x_{n,1}} (1 - \theta_{k,1})^{1-x_{n,1}} \quad (8)$$

其中， $\theta_{k,1}$ 表示第 k 个子分布中参数向量 θ_k 的第 1 维参数， $x_{n,1}$ 表示第 n 个样本的第 1 维变量值。然后让 (8) 对参数 $\theta_{k,1}$ 求导并令其导数为 0：

$$\begin{aligned} \frac{\log L(\Omega | \Theta, \pi)}{\partial \theta_{k,1}} &= \sum_{n=1}^N \frac{\partial}{\partial \theta_{k,1}} \log \sum_{k=1}^K \pi_k \theta_{k,1}^{x_{n,1}} (1 - \theta_{k,1})^{1-x_{n,1}} \\ &= \sum_{n=1}^N \frac{\pi_k \theta_{k,1}^{x_{n,1}} (1 - \theta_{k,1})^{1-x_{n,1}}}{\sum_{k=1}^K \pi_k \theta_{k,1}^{x_{n,1}} (1 - \theta_{k,1})^{1-x_{n,1}}} \frac{\frac{\partial}{\partial \theta_{k,1}} \theta_{k,1}^{x_{n,1}} (1 - \theta_{k,1})^{1-x_{n,1}}}{\theta_{k,1}^{x_{n,1}} (1 - \theta_{k,1})^{1-x_{n,1}}} \\ &= \sum_{n=1}^N \gamma(z_{nk}) \frac{\partial}{\partial \theta_{k,1}} \log \left(\theta_{k,1}^{x_{n,1}} (1 - \theta_{k,1})^{1-x_{n,1}} \right) \\ &= \sum_{n=1}^N \gamma(z_{nk}) \frac{\partial}{\partial \theta_{k,1}} (x_{n,1} \log \theta_{k,1} + (1 - x_{n,1}) \log (1 - \theta_{k,1})) \\ &= \sum_{n=1}^N \gamma(z_{nk}) \left(\frac{x_{n,1}}{\theta_{k,1}} - \frac{1 - x_{n,1}}{1 - \theta_{k,1}} \right) \\ &= 0 \end{aligned}$$

由此可得

$$\begin{aligned} \theta_{k,1} &= \frac{\sum_{n=1}^N \gamma(z_{nk}) x_{n,1}}{\sum_{n=1}^N \gamma(z_{nk})} \\ \theta_{k,2} &= 1 - \theta_{k,1} \end{aligned}$$

7.3.3 混合系数 π_k 的求解

对于混合系数 π_k 的求解，由于 π_k 满足约束 $\sum_{k=1}^K \pi_k = 1$ ，因此在对数似然函数通用表达式末尾添加添加惩罚项 $\lambda \left(1 - \sum_{k=1}^K \pi_k \right)$ ，因此得到加了惩罚项的对数似然函数 $p \log L(\Omega | \Theta, \pi)$

$$p \log L(\Omega | \Theta, \pi) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k P(\mathbf{x}_n | \theta_k) + \lambda \left(1 - \sum_{k=1}^K \pi_k \right)$$

让上式对 π_k 求导得：

$$\begin{aligned} \frac{p \log L(\Omega | \Theta, \pi)}{\partial \pi_k} &= \sum_{n=1}^N \frac{\partial}{\partial \pi_k} \log \sum_{k=1}^K \pi_k P(\mathbf{x}_n | \theta_k) - \lambda \\ &= \sum_{n=1}^N \frac{\gamma(z_{nk})}{\pi_k} - \lambda = 0 \end{aligned}$$

对上式最后一行两边同乘 π_k ，并对 k 求和得：

$$\sum_{k=1}^K \sum_{n=1}^N \gamma(z_{nk}) - \lambda \sum_{k=1}^K \pi_k = 0$$

因此可得:

$$\lambda = N$$

$$\pi_k = \frac{\sum_{n=1}^N \gamma(z_{nk})}{N}$$

7.4 EM 算法

EM 算法是一种迭代算法，用于含有隐变量 (Hidden variable) 的概率模型参数的最大似然估计。该算法流程如下：

1. 确定混合数 K ，并初始化参数。
2. E-step: 依据当前参数，计算每个数据 x_n 来自子模型 k 的后验概率 $\gamma(z_{nk})$ 。
3. M-step: 根据不同分布，计算模型参数 Θ 以及混合系数 π 。
4. 重复计算 E-step 和 M-step 直至收敛 ($\|\Theta_{i+1} - \Theta_i\| < \epsilon$ 是一个很小的正数)。

嘻嘻”... —. —... —. —... —... —... ..— ..— ..—”

8 Lecture 8

2022.4.11

Outline

1. EM-principle

生成过程

混合模型似然函数

2. EM 算法

8.1 EM-principle

苏老师：再过五年，所有的数学课都需要编程。

.....
Data $D = \{(x_i)\}_{i=1}^n$

Model

8.1.1 生成过程

假设，我们将从 2 个 ($k = 2$) 高斯分布中取样数据。我们使用 z_{nk} 作为指示变量。如果我们选择第 k 个组分 (component) 作为第 n 个对象的来源，那么我们设置 $z_{nk} = 1$ ，并且对其他的 $j \neq k$ ，设置 $z_{nj} = 0$ 。 μ_k 和 Σ_k 表示第 k 个高斯分布的参数。

如果 \mathbf{x}_n 是从第 k 个组分中产生的，那么它的密度函数为一个均值和协方差分别为 μ_k 和 Σ_k 的高斯分布：

$$p(\mathbf{x}_n | z_{nk} = 1, \mu_k, \Sigma_k) = \mathcal{N}(\mu_k, \Sigma_k) \quad (8.1)$$

假设，对这 2 个组分我们采用如下的均值和协方差，

$$\mu_1 = [3, 3]^T, \quad \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}, \quad \mu_2 = [1, -3]^T, \quad \Sigma_2 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \quad (8.2)$$

最后，我们需要定义 π_k 。如果第一个组分比第二个组分更可能，那么我们使用 $\pi_1 = 0.7$ 、 $\pi_2 = 0.3$ 。以上就是混合模型的生成程序。

8.1.2 混合模型似然函数

为了进行 EM 算法，我们需要首先给出似然函数的表达式。为了尽可能地通用化，我们用 $p(\mathbf{x}_n | z_{nk} = 1, \Delta_k)$ 表示第 k 个类的密度函数（不一定为高斯分布），其中 Δ_k 为其中的参数。另外，我们用 $\Delta = \{\Delta_1, \dots, \Delta_k\}$ 来表示各个组分的参数集合，并把所有的 π_k 整合为一个向量 $\pi = \{\pi_1, \dots, \pi_k\}$ 。

我们需要在整个模型下数据 \mathbf{x}_n 的似然函数 $p(\mathbf{x}_n | \Delta, \pi)$ 。为了得到这个表达式，我们从 $z_{nk} = 1$ 的特定数据对象的似然函数开始：

$$p(\mathbf{x}_n | z_{nk} = 1, \Delta) = p(\mathbf{x}_n | \Delta_k) \quad (8.3)$$

也就是我们之前定义的 π_k 。那么有

$$p(\mathbf{x}_n | z_{nk} = 1, \Delta) p(z_{nk} = 1) = p(\mathbf{x}_n | \Delta_k) p(z_{nk} = 1)$$

$$p(\mathbf{x}_n, z_{nk} = 1 | \Delta, \pi) = p(\mathbf{x}_n | \Delta_k) \pi_k$$

等式两边对所有的 k 个组分进行求和, 得到似然函数

$$\sum_{k=1}^K p(\mathbf{x}_n, z_{nk} = 1 | \Delta, \pi) = \sum_{k=1}^K p(\mathbf{x}_n | \Delta_k) \pi_k$$

$$p(\mathbf{x}_n | \Delta, \pi) = \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \Delta_k)$$

根据基本的样本独立性假设, 我们可以得到 N 个数据对象的似然函数:

$$p(\mathbf{X} | \Delta, \pi) = \prod_{n=1}^N \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \Delta_k) \quad (8.4)$$

8.2 EM 算法

我们现在需要说明使用 EM 算法求似然函数的最大值。首先对似然函数取自然对数, 即,

$$L = \log p(\mathbf{X} | \Delta, \pi) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (8.5)$$

对数内的求和使我们直接寻找最优的 $\boldsymbol{\mu}_k$ 、 $\boldsymbol{\Sigma}_k$ 、 π 参数值比较困难。而 EM 算法通过计算似然函数的一个下界来解决这个问题。我们不再直接对 L 进行最大化, 而转为对它的下界最大化。

为了得到 L 的下界, 我们可以使用下面期望的对数和期望的对数的期望关系, 也就是著名的 **詹森 (Jensen) 不等式**:

$$\log \mathbf{E}_{p(z)} \{f(z)\} \geq \mathbf{E}_{p(z)} \{\log f(z)\} \quad (8.6)$$

也就是说, $f(z)$ 期望值的对数总是大于等于 $\log f(z)$ 的期望值。

为了能够应用詹森不等式来求似然函数的下界, 我们将对 k 求和的公式内的表达式先乘后除以一个新的变量 q_{nk} 。

$$L = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \frac{q_{nk}}{q_{nk}} \quad (8.7)$$

如果我们约束 q_{nk} 是正的且满足求和约束条件 $\sum_{k=1}^K q_{nk} = 1$ (也就是说, q_{nk} 表示第 n 个个体在这个 k 个组分中的概率分布), 那么我们可以重新整理公式为基于 q_{nk} 的期望, 即,

$$L = \sum_{n=1}^N \log \sum_{k=1}^K q_{nk} \frac{\pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{q_{nk}}$$

$$= \sum_{n=1}^N \log \mathbf{E}_{q_{\mu_k}} \left\{ \frac{\pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{q_{nk}} \right\}$$

利用詹森不等式, 我们可以得到 L 的下界,

$$L = \sum_{n=1}^N \log \mathbf{E}_{q_{\mu_k}} \left\{ \frac{\pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{q_{nk}} \right\} \geq \sum_{n=1}^N \mathbf{E}_{q_{\tau_k}} \left\{ \log \frac{\pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{q_{nk}} \right\}$$

不等式的右侧部分就是我们需要优化的表达式的下界 (记为 \mathcal{B})。把表达式展开, 我们将更容易操作。

$$\begin{aligned}\mathcal{B} &= \sum_{n=1}^N \mathbf{E}_{q_*} \left\{ \log \frac{\pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{q_{nk}} \right\} \\ &= \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log \left(\frac{\pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{q_{nk}} \right) \\ &= \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log \pi_k + \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) - \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log q_{nk}\end{aligned}$$

使得这个下界达到局部最大值的 $q_{nk} \pi_k \boldsymbol{\mu}_k \boldsymbol{\Sigma}_k$ 参数值也会使对数似然函数 L 达到最大。就像我们前面提到的, EM 算法是一个迭代算法。这就需要我们不断地重复更新模型中的数值直到收敛。为了每次更新, 我们需要计算 \mathcal{B} 针对某个参数的偏导数, 并令其等于 0, 然后求解。下面我们将对各个参数依次求解。

8.2.1 更新 π_k

只有 \mathcal{B} 的第一部分包含 π_k (其他部分对 π_k 的偏导数为 0)。 π_k 是一个概率, 所以有 $\sum_k \pi_k = 1$ 。因此, 对 π_k 进行优化是有条件约束的。使用拉格朗日算法将约束条件整合进目标函数:

$$\mathcal{B} = \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log \pi_k - \lambda \left(\sum_{k=1}^K \pi_k - 1 \right) + \dots \quad (8.8)$$

对上式求 π_k 的偏导数, 并使其等于 0, 然后整理, 得到,

$$\begin{aligned}\frac{\partial \mathcal{B}}{\partial \pi_k} &= \frac{\sum_{n=1}^N q_{nk}}{\pi_k} - \lambda = 0 \\ \sum_{n=1}^N q_{nk} &= \lambda \pi_k\end{aligned}$$

最后我们需要计算 λ , 等式两边对 k 求和, 可得:

$$\begin{aligned}\sum_{k=1}^K \sum_{n=1}^N q_{nk} &= \lambda \sum_{k=1}^K \pi_k \\ \sum_{n=1}^N 1 &= \lambda \\ \lambda &= N\end{aligned}$$

其中我们用到了 $\sum_{k=1}^K q_{nk} = 1$ 和 $\sum_{k=1}^K \pi_k = 1$ 的事实。将 $\lambda = N$ 代入, 可以得到 π_k 的表达式为:

$$\pi_k = \frac{1}{N} \sum_{n=1}^N q_{nk} \quad (8.9)$$

8.2.2 更新 $\boldsymbol{\mu}_k$

接下来, 我们考虑 $\boldsymbol{\mu}_k, \mathcal{B}$ 中只有第 2 部分包含 $\boldsymbol{\mu}_k$ 。如果我们将 $p(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ 作为多变量高斯分布的密度函数, 并展开, 可得:

$$\begin{aligned}\mathcal{B} &= \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log \left(\frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_k|^{1/2}} \exp \left(-\frac{1}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right) \right) + \dots \\ &= -\frac{1}{2} \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log ((2\pi)^d |\boldsymbol{\Sigma}_k|) - \frac{1}{2} \sum_{n=1}^N \sum_{k=1}^K q_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) + \dots\end{aligned}$$

第一部分不包含 μ_k , 因此可以忽略。利用下面的性质,

$$f(w) = w^T C w, \quad \frac{\partial f(w)}{\partial w} = 2Cw$$

和链式法则, 我们可以求 B 对 μ_k 的偏导数,

$$\begin{aligned} \frac{\partial B}{\partial \mu_k} &= -\frac{1}{2} \sum_{n=1}^N q_{nk} \times \frac{\partial (\mathbf{x}_n - \mu_k)^T \Sigma_k^{-1} (\mathbf{x}_n - \mu_k)}{\partial (\mathbf{x}_n - \mu_k)} \times \frac{\partial (\mathbf{x}_n - \mu_k)}{\partial \mu_k} \\ &= \sum_{n=1}^N q_{nk} \Sigma_k^{-1} (\mathbf{x}_n - \mu_k) \end{aligned}$$

令其等于 0 并整理, 可以得到 μ_k 的表达式,

$$\begin{aligned} \sum_{n=1}^N q_{nk} \Sigma_k^{-1} (\mathbf{x}_n - \mu_k) &= 0 \\ \sum_{n=1}^N q_{nk} \Sigma_k^{-1} \mathbf{x}_n &= \sum_{n=1}^N q_{nk} \Sigma_k^{-1} \mu_k \\ \sum_{n=1}^N q_{nk} \mathbf{x}_n &= \mu_k \sum_{n=1}^N q_{nk} \\ \mu_k &= \frac{\sum_{n=1}^N q_{nk} \mathbf{x}_n}{\sum_{n=1}^N q_{nk}} \end{aligned}$$

8.2.3 更新 Σ_k

与 μ_k 一样, 我们只需要考虑 B 中的项 $p(\mathbf{x}_n | \mu_k, \Sigma_k)$ 。我们将该项展开:

$$B = -\frac{1}{2} \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log((2\pi)^d |\Sigma_k|) - \frac{1}{2} \sum_{n=1}^N \sum_{k=1}^K q_{nk} (\mathbf{x}_n - \mu_k)^T \Sigma_k^{-1} (\mathbf{x}_n - \mu_k) + \dots \quad (8.10)$$

忽略第一项中的常数部分 (2π) , 我们得到,

$$B = -\frac{1}{2} \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log(|\Sigma_k|) - \frac{1}{2} \sum_{n=1}^N \sum_{k=1}^K q_{nk} (\mathbf{x}_n - \mu_k)^T \Sigma_k^{-1} (\mathbf{x}_n - \mu_k) + \dots \quad (8.11)$$

为了对 Σ_k 求偏导数, 我们需要以下的性质:

$$\frac{\partial \log |C|}{\partial C} = (C^T)^{-1} \quad (8.12)$$

和

$$\frac{\partial \mathbf{a}^T C^{-1} \mathbf{b}}{\partial C} = - (C^T)^{-1} \mathbf{a} \mathbf{b}^T (C^T)^{-1} \quad (8.13)$$

利用这两个性质, 我们可以求 B 对 Σ_k 的偏导数。

$$\frac{\partial B}{\partial \Sigma_k} = -\frac{1}{2} \sum_{n=1}^N q_{nk} \Sigma_k^{-1} + \frac{1}{2} \sum_{n=1}^N q_{nk} \Sigma_k^{-1} (\mathbf{x}_n - \mu_k) (\mathbf{x}_n - \mu_k)^T \Sigma_k^{-1} \quad (8.14)$$

注意, Σ_k 是一个协方差矩阵, 是对称的, 因此 $\Sigma_k^T = \Sigma_k$ 。令该式等于 0 并整理, 得到

$$\begin{aligned} -\frac{1}{2} \sum_{n=1}^N q_{nk} \Sigma_k^{-1} + \frac{1}{2} \sum_{n=1}^N q_{nk} \Sigma_k^{-1} (\mathbf{x}_n - \mu_k) (\mathbf{x}_n - \mu_k)^T \Sigma_k^{-1} &= 0 \\ \frac{1}{2} \sum_{n=1}^N q_{nk} \Sigma_k^{-1} &= \frac{1}{2} \sum_{n=1}^N q_{nk} \Sigma_k^{-1} (\mathbf{x}_n - \mu_k) (\mathbf{x}_n - \mu_k)^T \Sigma_k^{-1} \end{aligned}$$

在等式两侧分别左乘和右乘 Σ_k , 可以使我们消掉 Σ_k^{-1} :

$$\begin{aligned}\Sigma_k \sum_{n=1}^N q_{nk} \Sigma_k^{-1} \Sigma_k &= \Sigma_k \Sigma_k^{-1} \sum_{n=1}^N q_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k) (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} \Sigma_k \\ \Sigma_k \sum_{n=1}^N q_{nk} &= \sum_{n=1}^N q_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k) (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \\ \Sigma_k &= \frac{\sum_{n=1}^N q_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k) (\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N q_{nk}}\end{aligned}$$

8.2.4 E 步——更新 q_{nk}

最后, 我们更新 q_{nk} , 它在 \mathcal{B} 的三项中都出现。另外, 它受条件 $\sum_{k=1}^K q_{nk} = 1$ 的约束, 因此类似于更新 π_k 。我们使用拉格朗日项。下界 \mathcal{B} 和拉格朗日项为:

$$\begin{aligned}\mathcal{B} &= \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log \pi_k + \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log p(\mathbf{x}_n | \boldsymbol{\mu}_k, \Sigma_k) - \sum_{n=1}^N \sum_{k=1}^K q_{nk} \log q_{nk} \\ &\quad - \lambda \left(\sum_{k=1}^K q_{nk} - 1 \right)\end{aligned}$$

对 q_{nk} 求偏导数, 得到,

$$\frac{\partial \mathcal{B}}{\partial q_{nk}} = \log \pi_k + \log p(\mathbf{x}_n | \boldsymbol{\mu}_k, \Sigma_k) - (1 + \log q_{nk}) - \lambda$$

令其等于 0, 整理并求指数, 得到了 q_{nk} 的表达式:

$$\begin{aligned}1 + \log q_{nk} + \lambda &= \log \pi_k + \log p(\mathbf{x}_n | \boldsymbol{\mu}_k, \Sigma_k) \\ \exp(\log q_{nk} + (\lambda + 1)) &= \exp(\log \pi_k + \log p(\mathbf{x}_n | \boldsymbol{\mu}_k, \Sigma_k)) \\ q_{nk} \exp(\lambda + 1) &= \pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \Sigma_k)\end{aligned}$$

与更新 π_k 一样, 为了得到常数项 (此时为 $\exp(\lambda + 1)$), 我们对等式两边的 k 项求和, 得到:

$$\begin{aligned}\exp(\lambda + 1) \sum_{k=1}^K q_{nk} &= \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \Sigma_k) \\ \exp(\lambda + 1) &= \sum_{k=1}^K \pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \Sigma_k)\end{aligned}$$

从而得到 q_{nk} 的表达式:

$$q_{nk} = \frac{\pi_k p(\mathbf{x}_n | \boldsymbol{\mu}_k, \Sigma_k)}{\sum_{j=1}^K \pi_j p(\mathbf{x}_n | \boldsymbol{\mu}_j, \Sigma_j)}$$

作业：写一个 GMM 函数

GMM 模型：

$$L = \log p(\mathbf{X} \mid \Delta, \pi) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (8.15)$$

E 步

$$q_{nk} = \frac{\pi_k p(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j p(\mathbf{x}_n \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

M 步

$$\pi_k = \frac{1}{N} \sum_{n=1}^N q_{nk} \quad (8.16)$$

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N q_{nk} \mathbf{x}_n}{\sum_{n=1}^N q_{nk}}$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N q_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k) (\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_{n=1}^N q_{nk}}$$

9 Lecture 9

2022.4.18

Outline

1. EM-ELBO
2. EM for unsupervised task Naive Bayes
3. EM+Bayesian
4. SeqModel+HMM

9.1 EM-ELBO

EM 算法的 E-step 需要算出，在已知观测样本 x_n 的条件下，该样本来自于第 k 个子分布 ($z_k = 1$) 的概率。

$$\begin{aligned}\gamma(z_{nk}) &= P(z_k = 1 | \mathbf{x}_n) = \frac{P(\mathbf{x}_n | z_k = 1) P(z_k = 1)}{\sum_{k=1}^K P(\mathbf{x}_n | z_k = 1) P(z_k = 1)} \\ &= \frac{\pi_k P(\mathbf{x}_n | \boldsymbol{\theta}_k)}{\sum_{k=1}^K \pi_k P(\mathbf{x}_n | \boldsymbol{\theta}_k)}\end{aligned}$$

Variational inference (变分推论) 是为了解决当真实后验分布 $p_\theta(z | x) = \frac{p_\theta(x|z)p_\theta(z)}{p_\theta(x)}$ 难以计算的情况下，用一个方便学习的分布 $q_\phi(z | x)$ 来近似真实后验分布 $p_\theta(z | x)$ 。因此对 x 的 log-likelihood function 可以写成 $\log p_\theta(x) = D_{KL}(q_\phi(z | x) || p_\theta(z | x)) + \mathcal{L}(\theta, \phi; x)$ 。其中 $D_{KL}(q_\phi(z | x) || p_\theta(z | x))$ 是让学到的分布 $q_\phi(z | x)$ 去接近真实后验分布 $p_\theta(z | x)$ ， $\mathcal{L}(\theta, \phi; x)$ 是 variational lower bound, 也叫 evidence lower bound (ELBO)。

下面来计算 $\mathcal{L}(\theta, \phi; x)$ ：

$$\begin{aligned}\mathcal{L}(\theta, \phi; x) &= \log p_\theta(x) - D_{KL}(q_\phi(z | x) || p_\theta(z | x)) \\ &= \int q_\phi(z | x) \log p_\theta(x) dz - \int q_\phi(z | x) \log \frac{q_\phi(z | x)}{p_\theta(z | x)} dz \\ &= \int q_\phi(z | x) \log p_\theta(x) - q_\phi(z | x) \log q_\phi(z | x) + q_\phi(z | x) \log p_\theta(z | x) dz \\ &= \int -q_\phi(z | x) \log q_\phi(z | x) + q_\phi(z | x) (\log p_\theta(x) + \log p_\theta(z | x)) dz \\ &= \int -q_\phi(z | x) \log q_\phi(z | x) + q_\phi(z | x) (\log p_\theta(x, z)) dz \\ &= \mathbb{E}_{q_\phi(z|x)} [-\log q_\phi(z | x) + \log p_\theta(x, z)]\end{aligned}$$

以上为一个简单的 ELBO 形式。再继续往下推：

$$\begin{aligned}\mathcal{L}(\theta, \phi; x) &= \mathbb{E}_{q_\phi(z|x)} [-\log q_\phi(z | x) + \log (p_\theta(x | z)p_\theta(z))] \\ \mathcal{L}(\theta, \phi; x) &= \mathbb{E}_{q_\phi(z|x)} \left[\log \frac{p_\theta(z)}{q_\phi(z | x)} + \log p_\theta(x | z) \right] \\ \mathcal{L}(\theta, \phi; x) &= \int q_\phi(z | x) \log \frac{p_\theta(z)}{q_\phi(z | x)} dz + \mathbb{E}_{q_\phi(z|x)} [\log p_\theta(x | z)] \\ \mathcal{L}(\theta, \phi; x) &= -\text{KL}[p_\theta(z) || q_\phi(z | x)] + \mathbb{E}_{q_\phi(z|x)} [\log p_\theta(x | z)]\end{aligned}$$

以上就是我们熟知的 ELBO 的形式,等式右边第一项是预设的 prior 与学到的 z 的分布的负 KL-divergence, 第二项是对 x 的 log-likelihood, 这些我们都是知道的。

ELBO 是 x 的 log-likelihood 函数的下界

$$\begin{aligned}\log p(\mathbf{x}) &= \text{ELBO}(q) + \text{KL}(q(\mathbf{z}) \| p(\mathbf{z} | \mathbf{x})) \\ &\geq \text{ELBO}(q)\end{aligned}$$

最小化 $\text{KL}(q(\mathbf{z}) \| p(\mathbf{z} | \mathbf{x}))$ 等价于最大化 $\text{ELBO}(q)$

$$\begin{aligned}q^*(\mathbf{z}) &= \underset{q(\mathbf{z}) \in Q}{\operatorname{argmin}} \text{KL}(q(\mathbf{z}) \| \overbrace{p(\mathbf{z} | \mathbf{x})}^{\text{unknown}}) \\ &= \underset{q(\mathbf{z}) \in Q}{\operatorname{argmax}} \text{ELBO}(q)\end{aligned}$$

9.2 EM for unsupervised task——Naive Bayes

Data: $\{x_i\}_{i=1}^N$, x_i 代表一个文档, x_{ij} 代表一个词向量, $x_{ij} \in \{1, \dots, d, \dots, D\}$

Model:

1. 对类别建模, $P(y_i) = \prod_{k=1}^K \theta_k^{I(y_i=k)}$
2. $P(x_{ij}|y_i)$ 类别为 y_i 的文档中, 单词 x_{ij} 出现的概率。记该矩阵为 B , 矩阵中的每一个元素记为 b_{kd} 。
则 $P(x_i|y_i) = \prod_{j=1}^{t_i} P(x_{ij}|y_i)$
3. 极大似然估计: $P(x_i, y_i; \theta) = P(y_i)P(x_i|y_i)$ (注: θ 的个数为 $(K-1) + K \times (D-1)$)
4. 后验分布: $P(y_i|x_i; \theta) = \gamma_{ik}$
5. 在 EM 算法中求 $\sum_{i=1}^N \sum_{k=1}^K \gamma_{ik} \log P(x_i, y_i = k; \theta)$

Algorithm 5 EM for U-NB

Input: X ;

Output: Θ, B ;

1: **Loop:**

2: E-step: γ_{ik}

3: M-step: Θ, B

4: **until** $\| \Theta^{t+1} - \Theta^t \| \leq \epsilon$

5: **return** Θ, B ;

可以看作 $X_{N \times D} = \gamma_{N \times K} B_{K \times D}$

9.3 EM+Bayesian

如图1所示, y_i 的分布 θ 由以 α 为参数的分布决定。具体的算法流程如算法6所示。

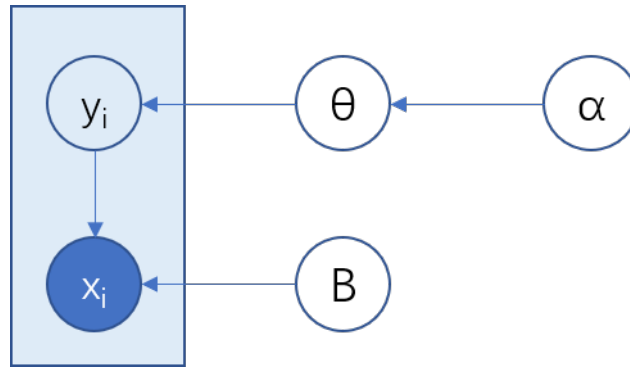


图 1: EM+Bayesian 概率图

Algorithm 6 EM+Bayesian**Input:** X ;**Output:** α, B ;1: **Loop:**2: E-step: $\gamma_{ik} = P(y_i, \theta | x_i; \alpha, B)$ 3: M-step: $\sum_{i=1}^N \int_{\theta} P(y_i, \theta) \log P(x_i, y_i, \theta | \alpha, B) d\theta$ 4: **until**5: **return** α, B ;**9.4 SeqModel+HMM**

隐马尔可夫模型：隐藏的马尔可夫链随机生成的状态的序列，称为**状态序列**；每个状态生成一个观测，而由此产生的观测的随机序列，称为**观测序列**。序列的每一个位置又可以看作是一个时刻。

例如， N 个袋子，每个袋子中有 M 种不同颜色的球。选择一个袋子，取出一个球，得到球的颜色。

1. 状态数为 N (袋子的数量)
2. 每个状态可能的符号数 M (不同颜色球的数目)
3. 状态转移概率矩阵 $A = a_{ij}$ (从一只袋子 (状态 S_i) 转向另一只袋子 (状态 S_j) 取球的概率)
4. 从状态 S_j 观察到某一特定符号 v_k 的概率分布矩阵为: $B = b_j(k)$ (从第 j 个袋子中取出第 k 种颜色的球的概率)
5. 初始状态的概率分布为: $\pi = \pi_i$

一般将一个隐马尔可夫模型记为: $\lambda = [\pi, A, B]$ 需要确定以下三方面内容 (三要素):

1. 初始状态概率 π : 模型在初始时刻各状态出现的概率，通常记为 $\pi = (\pi_1, \pi_2, \dots, \pi_N)$, π_i 表示模型的初始状态为 S_i 的概率。
2. 状态转移概率 A : 模型在各个状态间转换的概率，通常记为矩阵 $A[a_{ij}]$ ，其中 a_{ij} 表示在任意时刻 t ，若状态为 S_i ，则在下一时刻状态为 S_j 的概率。
3. 输出观测概率 B : 模型根据当前状态获得各个观测值的概率通常记为矩阵 $B = [(b_{ij})]$ 。其中， b_{ij} 表示在任意时刻 t ，若状态为 S_i ，则观测值 O_j 被获取的概率。

相对于马尔可夫模型,隐马尔可夫只是多了一个各状态的观测概率给定隐马尔可夫模型 $\lambda = [A, B, \pi]$, 它按如下过程产生观测序列 $\{X_1, X_2, \dots, X_n\}$:

1. 设置 $t = 1$, 并根据初始状态概率 π 选择初始状态 Y_1 ;
2. 根据状态值和输出观测概率 B 选择观测变量取值 X_t ;
3. 根据状态值和状态转移矩阵 A 转移模型状态, 即确定 Y_{t+1} ;

9.4.1 概率计算问题

给定模型 $\lambda = [\pi, A, B]$ 和观测序列 $O = (o_1, \dots, o_T)$, 计算在模型 λ 下观测序列 O 出现的概率 $P(O|\lambda)$

1. 直接算法: 对所有可能的状态序列 I 求和

$$\begin{aligned} P(O|\lambda) &= \sum_I P(O|I, \lambda) P(I|\lambda) \\ &= \sum \pi_i a_{i_1 i_2} \dots a_{i_{T-1} i_T} b_{i_1}(O_1) \dots b_{i_T}(O_T) \end{aligned}$$

10 Lecture 12

Outline

- Undirected Graphical Model
- Revisit: λ
- $\lambda++$: Reinforcement Learning: An Introduction

10.1 作业

分别用 MLE 和 EM 算法求解含有隐变量 h 的无向图模型。

MLE

首先,

$$\nabla_{\theta} \sum_{i=1}^n \ln P(v_i; \theta) = \sum_{i=1}^n \nabla_{\theta} \ln \sum_{h_i} P(v_i, h_i; \theta) \quad (10.1)$$

对于 $\nabla_{\theta} \ln \sum_{h_i} P(v_i, h_i; \theta)$ 进行计算, 有

$$\frac{1}{\sum_{h_i} P(v_i, h_i; \theta)} \sum_{h_i} \nabla_{\theta} P(v_i, h_i; \theta) \quad (10.2)$$

我们可以先看 $\ln P(v, h; \theta)$, 有

$$\ln P(v, h; \theta) = -E(v, h; \theta) - \ln Z(\theta)$$

关于 θ 求导, 有

$$\frac{1}{P(v, h; \theta)} \nabla_{\theta} P(v, h; \theta) = -\nabla_{\theta} E(v, h; \theta) - \frac{1}{Z(\theta)} \nabla_{\theta} Z(\theta)$$

从而得到

$$\nabla_{\theta} P(v, h; \theta) = P(v, h; \theta) * (-\nabla_{\theta} E(v, h; \theta) - \frac{1}{Z(\theta)} \nabla_{\theta} Z(\theta))$$

可以将其代入公式 10.2, 从而有

$$\frac{1}{\sum_{h_i} P(v_i, h_i; \theta)} \sum_{h_i} \nabla_{\theta} P(v_i, h_i; \theta) = \frac{1}{\sum_{h_i} P(v_i, h_i; \theta)} \sum_{h_i} P(v_i, h_i; \theta) (-\nabla_{\theta} E(v_i, h_i; \theta) - \frac{1}{Z(\theta)} \nabla_{\theta} Z(\theta)) \quad (10.3)$$

其中 $\frac{1}{Z(\theta)} \nabla_{\theta} Z(\theta)$ 在第一个无向图模型问题中求过, 如下

$$\begin{aligned} \frac{\nabla_{\theta} Z(\theta)}{Z(\theta)} &= \iint \frac{\exp\{-E(v, h; \theta)\}}{Z(\theta)} (-\nabla_{\theta} E(v, h; \theta)) dv dh \\ &= -\mathbb{E}_{(v, h) \sim P(v, h; \theta)} [\nabla_{\theta} E(v, h; \theta)] \end{aligned}$$

且它与 \sum_{h_i} 无关, 同时, $\sum_{h_i} P(v_i, h_i; \theta)$ 可视为一个值, 公式10.3可写作

$$\begin{aligned} & - \frac{1}{\sum_{h_i} P(v_i, h_i; \theta)} \mathbb{E}_{(v, h) \sim P(v, h; \theta)} [\nabla_{\theta} E(v, h; \theta)] - \sum_{h_i} \frac{P(v_i, h_i; \theta)}{P(v_i; \theta)} \nabla_{\theta} E(v_i, h_i; \theta) \\ &= - \frac{1}{\sum_{h_i} P(v_i, h_i; \theta)} \mathbb{E}_{(v, h) \sim P(v, h; \theta)} [\nabla_{\theta} E(v, h; \theta)] - \mathbb{E}_{P(h_i | v_i)} [\nabla_{\theta} E(v_i, h_i; \theta)] \end{aligned} \quad (10.4)$$

对于单个样本 v_i ,

$$\begin{aligned} \nabla_a E(v, h; \theta) &= -v \\ \nabla_b E(v, h; \theta) &= -h \\ \nabla_c E(v, h; \theta) &= -v^T h \end{aligned}$$

则,

$$\begin{aligned} \mathbb{E}[\nabla_a E(v, h; \theta)] &= -E[V] \\ \mathbb{E}[\nabla_b E(v, h; \theta)] &= -E[H] \\ \mathbb{E}[\nabla_c E(v, h; \theta)] &= -E[V^T H] \end{aligned}$$

同样,

$$\begin{aligned} \mathbb{E}[\nabla_a E(v_i, h_i; \theta)] &= -v_i \\ \mathbb{E}[\nabla_b E(v_i, h_i; \theta)] &= -h_i \\ \mathbb{E}[\nabla_c E(v_i, h_i; \theta)] &= -v_i^T h_i \end{aligned}$$

$\theta = (a, b, c)$ 的更新公式为:

$$\begin{aligned} a^{t+1} &\leftarrow a^t - \alpha(E[V] - v_i) \\ b^{t+1} &\leftarrow b^t - \alpha(E[H] - h_i) \\ c^{t+1} &\leftarrow c^t - \alpha(E[V^T H] - v_i^T h_i) \end{aligned} \quad (10.5)$$

Algorithm 7 MLE: SGD for Undirected Graphical Model**Input:** $D = \{v_i\}_{i=1}^n$;**Output:** $\Theta = (a, b, C)$;1: **init:** a, b, C 2: **Loop:**3: $a^{t+1} \leftarrow a^t - \alpha(E[V] - v_i)$ 4: $b^{t+1} \leftarrow b^t - \alpha(E[H] - h_i)$ 5: $c^{t+1} \leftarrow c^t - \alpha(E[V^T H] - v_i^T h_i)$ 6: **until**7: **return** Θ ;**EM 算法**

根据公式10.1, 有

$$\ln \sum_{h_i} P(v_i, h_i; \theta) = \ln \sum_{h_i} P(h_i|v_i; \theta) \frac{P(v_i, h_i; \theta)}{P(h_i|v_i; \theta)}$$

其中,

$$\sum_{h_i} P(h_i|v_i; \theta) \frac{P(v_i, h_i; \theta)}{P(h_i|v_i; \theta)} = \mathbb{E}_{h_i \sim P(h_i|v_i; \theta)} [P(v_i|h_i; \theta)]$$

根据 Jensen 不等式, 有

$$\begin{aligned} \ln \mathbb{E}_{h_i \sim P(h_i|v_i; \theta)} [P(v_i|h_i; \theta)] &\leq \mathbb{E}_{h_i \sim P(h_i|v_i; \theta)} [\ln P(v_i|h_i; \theta)] \\ &= \sum_{h_i} P(h_i|v_i; \theta) \ln \frac{P(v_i, h_i; \theta)}{P(h_i|v_i; \theta)} \\ &= \sum_{h_i} P(h_i|v_i; \theta) (\ln P(v_i, h_i; \theta) - \ln P(h_i|v_i; \theta)) \\ &= \sum_{h_i} P(h_i|v_i; \theta) \left(\ln \frac{1}{Z(\theta)} \exp\{-E(v_i, h_i; \theta)\} - \ln P(h_i|v_i; \theta) \right) \\ &= \sum_{h_i} P(h_i|v_i; \theta) (-E(v_i, h_i; \theta) - \ln Z(\theta)) - \sum_{h_i} P(h_i|v_i; \theta) \ln P(h_i|v_i; \theta) \end{aligned} \quad (10.6)$$

E-step对 $P(h_i|v_i; \theta)$ 求导, 构造拉格朗日乘子,

$$\mathcal{L}(P(h_i|v_i; \theta), \lambda) = \sum_{h_i} P(h_i|v_i; \theta) (-E(v_i, h_i; \theta) - \ln Z(\theta)) - \sum_{h_i} P(h_i|v_i; \theta) \ln P(h_i|v_i; \theta) - \lambda \left(\sum_{h_i} P(h_i|v_i; \theta) - 1 \right)$$

对 $P(h_i|v_i; \theta)$ 求导, 有

$$\nabla_{P(h_i|v_i; \theta)} \mathcal{L}(P(h_i|v_i; \theta), \lambda) = -E(v_i, h_i; \theta) - \ln Z(\theta) - \ln P(h_i|v_i; \theta) - 1 - \lambda = 0$$

则,

$$P(h_i|v_i; \theta) * \exp\{1 + \lambda\} = \exp\{-E(v_i, h_i; \theta) - \ln Z(\theta)\}$$

因为 $\sum_{h_i} P(h_i|v_i; \theta) = 1$, 因此

$$\sum_{h_i} P(h_i|v_i; \theta) * \exp\{1 + \lambda\} = \sum_{h_i} \exp\{-E(v_i, h_i; \theta) - \ln Z(\theta)\}$$

从而,

$$\exp\{1 + \lambda\} = \sum_{h_i} \exp\{-E(v_i, h_i; \theta) - \ln Z(\theta)\}$$

所以,

$$P(h_i|v_i; \theta) = \frac{\exp\{-E(v_i, h_i; \theta) - \ln Z(\theta)\}}{\sum_{h_i} \exp\{-E(v_i, h_i; \theta) - \ln Z(\theta)\}}$$

M-step

根据公式10.6, 对 θ 求导, 先分别求各项对 θ 的求导结果

$$\nabla_a E(v_i, h_i; \theta) = -v_i$$

$$\nabla_b E(v_i, h_i; \theta) = -h_i$$

$$\nabla_c E(v_i, h_i; \theta) = -v_i^T h_i$$

$$\begin{aligned} \frac{\nabla_\theta Z(\theta)}{Z(\theta)} &= \iint \frac{\exp\{-E(v, h; \theta)\}}{Z(\theta)} (-\nabla_\theta E(v, h; \theta)) dv dh \\ &= -\mathbb{E}_{(v, h) \sim P(v, h; \theta)} [\nabla_\theta E(v, h; \theta)] \end{aligned}$$

则求导结果为

$$\nabla_a(\cdot) = \sum_{h_i} P(h_i|v_i; \theta) v_i - E[V] = v_i - E[V]$$

$$\nabla_b(\cdot) = \sum_{h_i} P(h_i|v_i; \theta) h_i - E[H] = \mathbb{E}_{h_i \sim P(h_i|v_i; \theta)} [h_i] - E[H]$$

$$\nabla_c(\cdot) = \sum_{h_i} P(h_i|v_i; \theta) v_i^T h_i - E[V^T H] = \mathbb{E}_{h_i \sim P(h_i|v_i; \theta)} [v_i^T h_i] - E[V^T H]$$

Algorithm 8 EM: SGD for Undirected Graphical Model

Input: $D = \{v_i\}_{i=1}^n$;

Output: $\Theta = (a, b, C)$;

1: **init:** a, b, C

2: **Loop:**

3: **E-step**

4: $P(h_i|v_i; \theta) = \frac{\exp\{-E(v_i, h_i; \theta) - \ln Z(\theta)\}}{\sum_{h_i} \exp\{-E(v_i, h_i; \theta) - \ln Z(\theta)\}}$

5: **M-step**

6: $a^{t+1} \leftarrow a^t - \alpha(E[V] - v_i)$

7: $b^{t+1} \leftarrow b^t - \alpha(E[H] - \mathbb{E}_{h_i \sim P(h_i|v_i; \theta)} [h_i])$

8: $c^{t+1} \leftarrow c^t - \alpha(E[V^T H] - \mathbb{E}_{h_i \sim P(h_i|v_i; \theta)} [v_i^T h_i])$

9: **until**

10: **return** Θ ;
