

机器学习的数学笔记

X

目录

Notation	ii
1 逻辑回归	1
1.1 二项逻辑回归模型	1
1.2 Softmax回归模型	3
2 主成分分析	6
2.1 主成分分析的算法	6
2.2 主成分分析的数学原理	7
3 附录：信息熵	10
3.1 相关概念	10
3.2 熵的性质	12
参考文献	17
索引	18

Notation

This section provides a concise reference describing notation used throughout this document. If you are unfamiliar with any of the corresponding mathematical concepts, Goodfellow *et al.* (2016) describe most of these ideas in chapters 2–4.

Numbers and Arrays

a	A scalar (integer or real)
\mathbf{a}	A vector
\mathbf{A}	A matrix
\mathbf{A}	A tensor
\mathbf{I}_n	Identity matrix with n rows and n columns
\mathbf{I}	Identity matrix with dimensionality implied by context
$\mathbf{e}^{(i)}$	Standard basis vector $[0, \dots, 0, 1, 0, \dots, 0]$ with a 1 at position i
$\text{diag}(\mathbf{a})$	A square, diagonal matrix with diagonal entries given by \mathbf{a}
a	A scalar random variable
\mathbf{a}	A vector-valued random variable
\mathbf{A}	A matrix-valued random variable

Sets and Graphs

\mathbb{A}	A set
\mathbb{R}	The set of real numbers
$\{0, 1\}$	The set containing 0 and 1
$\{0, 1, \dots, n\}$	The set of all integers between 0 and n
$[a, b]$	The real interval including a and b
$(a, b]$	The real interval excluding a but including b
$\mathbb{A} \setminus \mathbb{B}$	Set subtraction, i.e., the set containing the elements of \mathbb{A} that are not in \mathbb{B}
\mathcal{G}	A graph
$Pa_{\mathcal{G}}(\mathbf{x}_i)$	The parents of \mathbf{x}_i in \mathcal{G}

Indexing

a_i	Element i of vector \mathbf{a} , with indexing starting at 1
a_{-i}	All elements of vector \mathbf{a} except for element i
$A_{i,j}$	Element i, j of matrix \mathbf{A}
$\mathbf{A}_{i,:}$	Row i of matrix \mathbf{A}
$\mathbf{A}_{:,i}$	Column i of matrix \mathbf{A}
$A_{i,j,k}$	Element (i, j, k) of a 3-D tensor \mathbf{A}
$\mathbf{A}_{:,:,i}$	2-D slice of a 3-D tensor
\mathbf{a}_i	Element i of the random vector \mathbf{a}

Linear Algebra Operations

\mathbf{A}^\top	Transpose of matrix \mathbf{A}
\mathbf{A}^+	Moore-Penrose pseudoinverse of \mathbf{A}
$\mathbf{A} \odot \mathbf{B}$	Element-wise (Hadamard) product of \mathbf{A} and \mathbf{B}
$\det(\mathbf{A})$	Determinant of \mathbf{A}

Calculus

$\frac{dy}{dx}$	Derivative of y with respect to x
$\frac{\partial y}{\partial x}$	Partial derivative of y with respect to x
$\nabla_{\mathbf{x}} y$	Gradient of y with respect to \mathbf{x}
$\nabla_{\mathbf{X}} y$	Matrix derivatives of y with respect to \mathbf{X}
$\nabla_{\mathbf{X}} y$	Tensor containing derivatives of y with respect to \mathbf{X}
$\frac{\partial f}{\partial \mathbf{x}}$	Jacobian matrix $\mathbf{J} \in \mathbb{R}^{m \times n}$ of $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$
$\nabla_{\mathbf{x}}^2 f(\mathbf{x})$ or $\mathbf{H}(f)(\mathbf{x})$	The Hessian matrix of f at input point \mathbf{x}
$\int f(\mathbf{x}) d\mathbf{x}$	Definite integral over the entire domain of \mathbf{x}
$\int_{\mathbb{S}} f(\mathbf{x}) d\mathbf{x}$	Definite integral with respect to \mathbf{x} over the set \mathbb{S}

Probability and Information Theory

$a \perp b$	The random variables a and b are independent
$a \perp b \mid c$	They are conditionally independent given c
$P(a)$	A probability distribution over a discrete variable
$p(a)$	A probability distribution over a continuous variable, or over a variable whose type has not been specified
$a \sim P$	Random variable a has distribution P
$\mathbb{E}_{\mathbf{x} \sim P}[f(\mathbf{x})]$ or $\mathbb{E}f(\mathbf{x})$	Expectation of $f(\mathbf{x})$ with respect to $P(\mathbf{x})$
$\text{Var}(f(\mathbf{x}))$	Variance of $f(\mathbf{x})$ under $P(\mathbf{x})$
$\text{Cov}(f(\mathbf{x}), g(\mathbf{x}))$	Covariance of $f(\mathbf{x})$ and $g(\mathbf{x})$ under $P(\mathbf{x})$
$H(\mathbf{x})$	Shannon entropy of the random variable \mathbf{x}
$D_{\text{KL}}(P \ Q)$	Kullback-Leibler divergence of P and Q
$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$	Gaussian distribution over \mathbf{x} with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$

Functions

$f : \mathbb{A} \rightarrow \mathbb{B}$	The function f with domain \mathbb{A} and range \mathbb{B}
$f \circ g$	Composition of the functions f and g
$f(\mathbf{x}; \boldsymbol{\theta})$	A function of \mathbf{x} parametrized by $\boldsymbol{\theta}$. (Sometimes we write $f(\mathbf{x})$ and omit the argument $\boldsymbol{\theta}$ to lighten notation)
$\log x$	Natural logarithm of x
$\sigma(x)$	Logistic sigmoid, $\frac{1}{1 + \exp(-x)}$
$\zeta(x)$	Softplus, $\log(1 + \exp(x))$
$\ \mathbf{x}\ _p$	L^p norm of \mathbf{x}
$\ \mathbf{x}\ $	L^2 norm of \mathbf{x}
x^+	Positive part of x , i.e., $\max(0, x)$
$\mathbf{1}_{\text{condition}}$	is 1 if the condition is true, 0 otherwise

Sometimes we use a function f whose argument is a scalar but apply it to a vector, matrix, or tensor: $f(\mathbf{x})$, $f(\mathbf{X})$, or $f(\mathbf{X})$. This denotes the application of f to the array element-wise. For example, if $\mathbf{C} = \sigma(\mathbf{X})$, then $C_{i,j,k} = \sigma(X_{i,j,k})$ for all valid values of i , j and k .

Datasets and Distributions

p_{data}	The data generating distribution
\hat{p}_{data}	The empirical distribution defined by the training set
\mathbb{X}	A set of training examples
$\mathbf{x}^{(i)}$	The i -th example (input) from a dataset
$\mathbf{y}^{(i)}$ or $\mathbf{y}^{(i)}$	The target associated with $\mathbf{x}^{(i)}$ for supervised learning
\mathbf{X}	The $m \times n$ matrix with input example $\mathbf{x}^{(i)}$ in row $\mathbf{X}_{i,:}$

Chapter 1

逻辑回归

1.1 二项逻辑回归模型

二项逻辑回归模型是如下的条件概率分布

$$P(Y = 1|\mathbf{x}) = \frac{\exp(\boldsymbol{\theta}^T \mathbf{x} + b)}{1 + \exp(\boldsymbol{\theta}^T \mathbf{x} + b)}$$
$$P(Y = 0|\mathbf{x}) = \frac{1}{1 + \exp(\boldsymbol{\theta}^T \mathbf{x} + b)}$$

其中 $\mathbf{x} \in \mathbb{R}^n$ 是输入变量， $Y \in \{0, 1\}$ 是输出变量， $\boldsymbol{\theta} \in \mathbb{R}^n$ 和 $b \in \mathbb{R}$ 是参数。 \mathbf{x} 和 $\boldsymbol{\theta}$ 为 n 维列向量。

若令 $\boldsymbol{\theta} = (\theta^{(1)}, \dots, \theta^{(n)}, b)^T$ ， $\mathbf{x} = (x^{(1)}, \dots, x^{(n)}, 1)^T$ ，那么条件概率可以表示为

$$P(Y = 1|\mathbf{x}) = \frac{\exp(\boldsymbol{\theta}^T \mathbf{x})}{1 + \exp(\boldsymbol{\theta}^T \mathbf{x})}$$
$$P(Y = 0|\mathbf{x}) = \frac{1}{1 + \exp(\boldsymbol{\theta}^T \mathbf{x})} \quad (1.1)$$

1.1.1 模型的参数估计

对于给定的训练集 $\mathbb{X} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$ ，可应用极大似然估计法估计模型参数。

为表示方便，令 $P(Y = 1|\mathbf{x}) = \pi(\mathbf{x})$ ， $P(Y = 0|\mathbf{x}) = 1 - \pi(\mathbf{x})$ ，似然函数为

$$L(\boldsymbol{\theta}) = \prod_{i=1}^N (\pi(\mathbf{x}_i))^{y_i} (1 - \pi(\mathbf{x}_i))^{1-y_i}$$

那么对数似然函数为

$$\begin{aligned}
 \log L(\boldsymbol{\theta}) &= \sum_{i=1}^N (y_i \log \pi(\mathbf{x}_i) + (1 - y_i) \log(1 - \pi(\mathbf{x}_i))) \\
 &= \sum_{i=1}^N \left(y_i \log \frac{\pi(\mathbf{x}_i)}{1 - \pi(\mathbf{x}_i)} + \log(1 - \pi(\mathbf{x}_i)) \right) \\
 &= \sum_{i=1}^N (y_i (\boldsymbol{\theta}^T \mathbf{x}_i) - \log(1 + \exp(\boldsymbol{\theta}^T \mathbf{x}_i)))
 \end{aligned} \tag{1.2}$$

1.1.1.1 参数估计：梯度下降法

根据公式 (1.2)，对数似然函数对 $\boldsymbol{\theta}$ 的偏导为

$$\begin{aligned}
 \nabla_{\boldsymbol{\theta}} \log L(\boldsymbol{\theta}) &= \sum_{i=1}^N \left(y_i \mathbf{x}_i - \frac{\exp(\boldsymbol{\theta}^T \mathbf{x}_i) \mathbf{x}_i}{1 + \exp(\boldsymbol{\theta}^T \mathbf{x}_i)} \right) \\
 &= \sum_{i=1}^N (y_i - \pi(\mathbf{x}_i)) \mathbf{x}_i
 \end{aligned}$$

由此处求对数似然函数的最大值，故需要沿着梯度上升的方向进行迭代，迭代公式为

$$\begin{aligned}
 \boldsymbol{\theta} &:= \boldsymbol{\theta} + \alpha \frac{\partial}{\partial \boldsymbol{\theta}} \log L(\boldsymbol{\theta}) \\
 &= \boldsymbol{\theta} + \alpha \sum_{i=1}^N (y_i - \pi(\mathbf{x}_i)) \mathbf{x}_i
 \end{aligned} \tag{1.3}$$

其中 α 称为学习率，是一个正常数。

公式 (1.3)可以用矩阵表示

$$\boldsymbol{\theta} := \boldsymbol{\theta} + \alpha X^T \boldsymbol{\Lambda} \tag{1.4}$$

其中 $\boldsymbol{\Lambda} = \begin{pmatrix} y_1 - \pi(\mathbf{x}_1) \\ y_2 - \pi(\mathbf{x}_2) \\ \dots \\ y_N - \pi(\mathbf{x}_N) \end{pmatrix}_{N \times 1}$ ， X 是由训练数据构成的 $N \times (n + 1)$ 矩阵(每一行对应一个样本，每一列对应样本的一个维度，其中还包括一维常数项)。

1.1.1.2 参数估计：随机梯度下降法

梯度下降算法在每次更新回归系数时需要遍历整个数据集，当数据集数量庞大或者

特征过多时，该方法的计算复杂度太高。改进方法是每次迭代仅用一个样本来更新回归系数，称为随机梯度下降法。

具体而言，对于训练集中的每一个样本 (x_i, y_i) ，计算该样本梯度，并依据迭代公式：

$$\boldsymbol{\theta} := \boldsymbol{\theta} + \alpha (y_i - \pi(\mathbf{x}_i)) \mathbf{x}_i \quad (1.5)$$

与公式 (1.3) 相比，随机梯度下降的迭代公式 (1.5) 中

- 误差变量是数值，而不是向量
- 不再有矩阵变换的过程

所以随机梯度下降算法的计算效率较高，缺点是存在解的不稳定性(如解存在周期性波动)的问题。为了解决这一问题，并进一步加快收敛速度，可以通过随机选取样本来更新回归系数。

1.2 Softmax回归模型

Softmax模型是二项回归模型在多分类问题上的推广，在多分类问题中，类标签 Y 可以取两个以上的值。

假设 Y 的取值集合是 $\{1, 2, \dots, K\}$ ，Softmax模型是如下的条件概率分布

$$P(Y = k | \mathbf{x}) = \frac{\exp(\boldsymbol{\theta}_k^T \mathbf{x})}{\sum_{j=1}^K \exp(\boldsymbol{\theta}_j^T \mathbf{x})} \quad (1.6)$$

其中 $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K \in \mathbb{R}^{n+1}$ 是模型的参数。

为方便起见，下文用矩阵 $\boldsymbol{\Theta}_{K \times (n+1)}$ 表示全部的模型参数

$$\boldsymbol{\Theta} = \begin{bmatrix} \boldsymbol{\theta}_1^T \\ \vdots \\ \boldsymbol{\theta}_K^T \end{bmatrix}$$

1.2.1 模型的参数估计

令 $P(Y = k | \mathbf{x}) = \pi_k(\mathbf{x})$ ，与二项逻辑回归类似，Softmax的似然函数可以表示为

$$L(\boldsymbol{\Theta}) = \prod_{i=1}^N \prod_{k=1}^K (\pi_k(\mathbf{x}_i))^{1_{y_i=k}}$$

对数似然函数为

$$\log L(\Theta) = \sum_{i=1}^N \sum_{k=1}^K \mathbf{1}_{y_i=k} \log \pi_k(\mathbf{x}_i) \quad (1.7)$$

1.2.1.1 参数估计：梯度下降法

首先求

$$\frac{\partial \pi_k(\mathbf{x}_i)}{\partial \theta_k} = \frac{\mathbf{x}_i \exp(\theta_k^T \mathbf{x}_i) \left(\sum_{j=1}^K \exp(\theta_j^T \mathbf{x}) - \exp(\theta_k^T \mathbf{x}_i) \right)}{\left(\sum_{j=1}^K \exp(\theta_j^T \mathbf{x}) \right)^2} \quad (1.8)$$

故根据公式 (1.7)，得到Softmax模型的对数似然函数的梯度

$$\begin{aligned} \nabla_{\theta_k} \log L(\Theta) &= \sum_{i=1}^N \mathbf{1}_{y_i=k} \frac{1}{\pi_k(\mathbf{x}_i)} \frac{\partial \pi_k(\mathbf{x}_i)}{\partial \theta_k} \\ &= \sum_{i=1}^N \mathbf{1}_{y_i=k} \frac{1}{\pi_k(\mathbf{x}_i)} \frac{\mathbf{x}_i \exp(\theta_k^T \mathbf{x}_i) \left(\sum_{j=1}^K \exp(\theta_j^T \mathbf{x}_i) - \exp(\theta_k^T \mathbf{x}_i) \right)}{\left(\sum_{j=1}^K \exp(\theta_j^T \mathbf{x}_i) \right)^2} \\ &= \sum_{i=1}^N \mathbf{1}_{y_i=k} \frac{\mathbf{x}_i \left(\sum_{j=1}^K \exp(\theta_j^T \mathbf{x}_i) - \exp(\theta_k^T \mathbf{x}_i) \right)}{\sum_{j=1}^K \exp(\theta_j^T \mathbf{x}_i)} \\ &= \sum_{i=1}^N \mathbf{1}_{y_i=k} \mathbf{x}_i (1 - \pi_k(\mathbf{x}_i)) \end{aligned} \quad (1.9)$$

对于任意第 k 个分类的参数 θ_k ，可沿着梯度上升的方向进行迭代

$$\theta_k := \theta_k + \alpha \sum_{i=1}^N \mathbf{1}_{y_i=k} \mathbf{x}_i (1 - \pi_k(\mathbf{x}_i)) \quad (1.10)$$

公式 (1.10)的迭代关系用矩阵可以表示为

$$\theta_k := \theta_k + \alpha X^T \Lambda \quad (1.11)$$

其中 $\mathbf{\Lambda} = \begin{pmatrix} \mathbf{1}_{y_1=k}(1 - \pi_k(\mathbf{x}_1)) \\ \mathbf{1}_{y_2=k}(1 - \pi_k(\mathbf{x}_2)) \\ \dots \\ \mathbf{1}_{y_N=k}(1 - \pi_k(\mathbf{x}_N)) \end{pmatrix}_{N \times 1}$, X 是由训练数据构成的 $N \times (n + 1)$ 矩阵(每一行对应一个样本, 每一列对应样本的一个维度, 其中还包括一维常数项)。

Chapter 2

主成分分析

主成分分析（Principal Component Analysis, PCA）是一种常见的**数据降维**方法，其目的是在信息量损失较小的前提下，将高维的数据转换到低维，从而减小计算量。实质就是找到一些投影方向，使得数据在这些投影方向上包含的信息量最大，而且这些投影方向是相互正交的。选择其中一部分包含最多信息量的投影方向作为新的数据空间，同时忽略包含较小信息量的投影方向，从而达到降维的目的。

样本的**信息量**可以理解为是样本在特征方向上投影的方差。方差越大，则样本在该特征上的差异就越大，因此该特征就越重要。参见《机器学习实战》上的图，在分类问题里，样本的方差越大，越容易将不同类别的样本区分开。

PCA的数学原理，就是对原始的空间中顺序地找一组相互正交的坐标轴，第一个轴是使得方差最大的，第二个轴是在与第一个轴正交的平面中使得方差最大的，第三个轴是在与第1、2个轴正交的平面中方差最大的，这样假设在N维空间中，可以找到N个这样的坐标轴，取前r个去近似这个空间，这样就从一个N维的空间压缩到r维的空间了，但是最终选择的r个坐标轴能够使得数据的损失最小。

2.1 主成分分析的算法

假设

- 存在n个原始数据，每个数据有p个特征，用矩阵表示为 $\mathbf{Z}_{n \times p} = (\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n)^T$ ，其中 \mathbf{z}_i 为p维列向量。

1. 去除平均值，即中心化，将数据**中心化**变换为 $\mathbf{X}_{n \times p} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)^T$ ，其中 $\mathbf{X} =$

$Z - \mathbb{E}Z$ (具体而言 $\mathbf{x}_i = \mathbf{z}_i - \boldsymbol{\mu}$, $\boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^n \mathbf{z}_i$)。

2. 计算 X 的协方差矩阵, 用 $\boldsymbol{\Sigma}_{p \times p}$ 表示

$$\text{Var} X = \text{Var}(Z - \mathbb{E}Z) = \text{Var} Z = \boldsymbol{\Sigma}$$

实际上 X 的协方差矩阵就是原始数据 Z 的协方差矩阵。

3. 计算协方差矩阵 $\boldsymbol{\Sigma}$ 的特征向量 $\{\boldsymbol{\xi}_j\}$ 和特征值 $\{\lambda_j\}$, $j = 1..p$ 。
4. 将特征值从小到大排序。
5. 保留前若干个特征值对应的特征向量, 假设保留的特征值为 $\{\lambda_j^*\}$, $j = 1..q$, 对应的特征向量构成的矩阵为 $\boldsymbol{\Xi}_{p \times q} = (\boldsymbol{\xi}_1^*, \boldsymbol{\xi}_2^*, \dots, \boldsymbol{\xi}_q^*)$
6. 将数据集 X 转换到上述 q 个特征向量构建的新的空间中, 得到新的数据集 $\mathbf{X}_{n \times q}^* =$

$$\mathbf{X}\boldsymbol{\Xi} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \dots \\ \mathbf{x}_i \\ \dots \\ \mathbf{x}_n \end{pmatrix} (\boldsymbol{\xi}_1^*, \boldsymbol{\xi}_2^*, \dots, \boldsymbol{\xi}_q^*)$$

2.2 主成分分析的数学原理

2.2.1 几个重要的定理

Theorem 2.2.1. $\boldsymbol{\Sigma}$ 为对称矩阵, 如下优化问题的解 \mathbf{u}^* 是 $\boldsymbol{\Sigma}$ 的最大特征值对应的特征向量。

$$\mathbf{u}^* = \arg \max_{\|\mathbf{u}\|=1} (\mathbf{u}^T \boldsymbol{\Sigma} \mathbf{u})$$

证明. 实际上约束条件 $\|\mathbf{u}\| = 1$ 等价于 $\mathbf{u}^T \mathbf{u} = 1$

利用拉格朗日乘子法, 得到

$$G(\mathbf{u}; \lambda) = \mathbf{u}^T \boldsymbol{\Sigma} \mathbf{u} + \lambda(\mathbf{u}^T \mathbf{u} - 1)$$

对 G 求 \mathbf{u} 的偏导得到

$$\nabla_{\mathbf{u}} G(\mathbf{u}; \lambda) = 2\boldsymbol{\Sigma} \mathbf{u} + 2\lambda \mathbf{u}$$

如果 \mathbf{u}^* 是优化问题的解，那么 \mathbf{u}^* 满足

$$\begin{aligned}\nabla_{\mathbf{u}} G(\mathbf{u}; \lambda) |_{\mathbf{u}=\mathbf{u}^*} &= 0 \\ \Rightarrow \Sigma \mathbf{u}^* &= -\lambda \mathbf{u}^* \\ \Rightarrow \Sigma \mathbf{u}^* &= \lambda^* \mathbf{u}^* \quad (\text{令 } \lambda^* = -\lambda)\end{aligned}$$

所以 \mathbf{u}^* 是矩阵 Σ 的特征向量，对应的特征值为 λ^* 。

当 $\mathbf{u} = \mathbf{u}^*$ 时，目标函数

$$\mathbf{u}^{*T} \Sigma \mathbf{u}^* = \lambda^* \quad (2.1)$$

为使得公式 (2.1) 最大，必须使得 λ^* 最大，即 λ^* 等于矩阵 Σ 最大的特征值，那么对应的特征向量便是真正的解 \mathbf{u}^* 。

□

2.2.2 最大方差投影

用 $\mathbf{u}_{p \times 1}$ 表示某投影方向上的单位向量，那么 \mathbf{x}_i 在 \mathbf{u} 上的投影可以表示为

$$\langle \mathbf{x}_i, \mathbf{u} \rangle = \mathbf{x}_i^T \mathbf{u}$$

那么数据集 \mathbf{X} 在 \mathbf{u} 上的投影向量为 $\mathbf{Y} = \mathbf{X}\mathbf{u}$ ，可知 \mathbf{Y} 的均值和方差为

$$\begin{aligned}\mathbb{E} \mathbf{Y} &= \mathbb{E} \mathbf{X} \mathbf{u} \\ \text{Var} \mathbf{Y} &= \text{Var} \mathbf{X} \mathbf{u} = \mathbf{u}^T (\text{Var} \mathbf{X}) \mathbf{u} = \mathbf{u}^T \Sigma \mathbf{u}\end{aligned}$$

主成分分析就是要到一个方向，使得数据集 \mathbf{X} 在该方向上投影方差最大。如果用单位向量 \mathbf{u}_1 来表示这个方向，那么 \mathbf{u}_1 是如下优化问题的解

$$\arg \max_{\|\mathbf{u}\|=1} \mathbf{u}^T \Sigma \mathbf{u}$$

根据定理 2.2.1， \mathbf{u}_1 就是 Σ 的最大特征值对应的特征向量，也是第一个主成分的单位向量。

随后要求第二个主成分，用单位向量 \mathbf{u}_2 表示这个方向。根据原理，第二个主成分依然是要最大化投影的方差，但是约束条件要多一个，即 \mathbf{u}_2 与 \mathbf{u}_1 正交。所以 \mathbf{u}_2 是如下优化问题的解

$$\begin{aligned}\arg \max_{\|\mathbf{u}\|=1} \quad & \mathbf{u}^T \Sigma \mathbf{u} \\ \text{s.t.} \quad & \mathbf{u}^T \mathbf{u}_1 = 0\end{aligned}$$

为求解上述优化问题，继续使用拉格朗日乘子法，得到

$$G(\mathbf{u}; \mathbf{u}_1, \lambda, \gamma) = \mathbf{u}^T \Sigma \mathbf{u} + (\mathbf{u}^T \mathbf{u} - 1) + \gamma \mathbf{u}^T \mathbf{u}_1$$

对 G 求 \mathbf{u} 的偏导得到，

$$\nabla_{\mathbf{u}} G(\mathbf{u}; \mathbf{u}_1, \lambda, \gamma) = 2\Sigma \mathbf{u} + 2\lambda \mathbf{u} + \gamma \mathbf{u}_1$$

如果 \mathbf{u}^* 是优化问题的解，那么 \mathbf{u}^* 满足

$$2\Sigma \mathbf{u}^* + 2\lambda \mathbf{u}^* + \gamma \mathbf{u}_1 = 0 \tag{2.2}$$

对公式 (2.2) 两边同乘以 \mathbf{u}_1^T ，得到

$$\begin{aligned} 2\Sigma \mathbf{u}^* \mathbf{u}_1^T + 2\lambda \mathbf{u}^* \mathbf{u}_1^T + \gamma \mathbf{u}_1 \mathbf{u}_1^T &= 0 \\ \Rightarrow \gamma \mathbf{u}_1 \mathbf{u}_1^T &= 0 \quad (\text{因为 } \mathbf{u}_1, \mathbf{u}^* \text{ 正交}) \\ \Rightarrow \gamma &= 0 \quad (\text{因为 } \mathbf{u}_1 \mathbf{u}_1^T = 1) \end{aligned}$$

于是公式 (2.2) 等于

$$\begin{aligned} \Sigma \mathbf{u}^* + \lambda \mathbf{u}^* &= 0 \\ \Rightarrow \Sigma \mathbf{u}^* &= -\lambda \mathbf{u}^* \\ \Rightarrow \Sigma \mathbf{u}^* &= \lambda^* \mathbf{u}^* \quad (\text{令 } \lambda^* = -\lambda) \end{aligned} \tag{2.3}$$

\mathbf{u}^* 也是矩阵 Σ 的特征值，与定理 2.2.1 的证明类似，优化问题的目标函数等于 λ^* ，为了使目标函数最大，要使 λ^* 尽量大，而 λ^* 最大可取所有特征值中第二大的，对应的特征向量就是优化问题的解，即**第二主成分**的方向向量。

后面的主成分算法同理类推。

Chapter 3

附录：信息熵

假设¹ X 是一个取有限值的离散随机变量(本文只考虑离散情况)，概率分布为 P 。

那么 $I(X = x_i) = -\log P(X = x_i)$ 称为事件 x_i 的自信息量，随机变量 X 的熵定义为 X 的自信息量的数学期望，即

$$H(X) = \mathbb{E}(I(X)) = - \sum_x P(x) \log P(x)$$

熵反映的是随机变量不确定程度的大小：熵的值越大，不确定程度越高。

3.1 相关概念

3.1.1 条件熵

条件熵是指在联合概率空间上熵的条件自信息的数学期望。在已知 X 时， Y 的条件熵为

$$H(Y|X) = \mathbb{E}_{x,y} I(y_j|x_i) = - \sum_x \sum_y P(x, y) \log P(y|x) \quad (3.1)$$

Lemma 3.1.1. 与公式 (3.1)等价的定义为给定 X 条件下 Y 的条件分布概率的熵的数学期望

$$H(Y|X) = \mathbb{E}_x H(Y|X = x) = \sum_x P(x) H(Y|X = x)$$

¹本章参考了信息论与编码(<http://www.docin.com/p-957983839-f6.html>)和信息论基础(<https://wenku.baidu.com/view/5319fed3b9f3f90f76c61b1a.html>)

证明.

$$\begin{aligned}
 H(Y|X) &= - \sum_x \sum_y P(x, y) \log P(y|x) \\
 &= - \sum_x \sum_y P(x) P(y|x) \log P(y|x) \\
 &= - \sum_x P(x) \sum_y P(y|x) \log P(y|x) \quad (P(x) \text{ 与 } y \text{ 无关}) \\
 &= \sum_x P(x) \left[- \sum_y P(y|x) \log P(y|x) \right] \\
 &= \sum_x P(x) H(Y|X = x)
 \end{aligned}$$

□

$H(Y|X)$ 的含义是已知在 X 发生的前提下, Y 发生新带来的熵。

3.1.2 相对熵

相对熵, 也称**KL散度**, 交叉熵等, 定义为两个概率分布之比的数学期望。

设 $Q(x), P(x)$ 是随机变量 X 中取值的两个概率分布, 则 P 对 Q 的相对熵是

$$D_{\text{KL}}(P\|Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)} = \mathbb{E}_x \log \frac{P(x)}{Q(x)} \quad (3.2)$$

相对熵可以用来度量两个随机变量的“距离”。

Lemma 3.1.2. 相对熵恒大于等于零。

证明. 对于任意分布 P, Q , 根据公式 (3.2), 可知

$$\begin{aligned}
 D_{\text{KL}}(P\|Q) &= \sum_x P(x) \log \frac{P(x)}{Q(x)} \\
 &= - \sum_x P(x) \log \frac{Q(x)}{P(x)} \\
 &\geq - \log \left(\sum_x P(x) \frac{Q(x)}{P(x)} \right) \quad (\text{对 } -\log x \text{ 应用 Jensen 不等式}) \\
 &= - \log \sum_x Q(x) \\
 &= - \log 1 \\
 &= 0
 \end{aligned}$$

□

3.1.3 互信息

两个随机变量 X, Y 的互信息，定义为 X, Y 的联合分布和独立分布乘积的相对熵

$$I(X, Y) = D_{\text{KL}}(P(X, Y) \| P(X)P(Y)) \quad (3.3)$$

Lemma 3.1.3. 互信息与条件熵满足如下关系

$$H(X|Y) = H(X) - I(X, Y) \quad (3.4)$$

证明. 根据公式 (3.2) 以及互信息的定义可知

$$I(X, Y) = \sum_{x,y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)}$$

那么

$$\begin{aligned} H(X) - I(X, Y) &= - \sum_x P(x) \log P(x) - \sum_{x,y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)} \\ &= - \sum_x \left(\sum_y P(x, y) \right) \log P(x) - \sum_{x,y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)} \\ &= - \sum_{x,y} P(x, y) \log P(x) - \sum_{x,y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)} \\ &= - \sum_{x,y} P(x, y) \left(\log P(x) + \log \frac{P(x, y)}{P(x)P(y)} \right) \\ &= - \sum_{x,y} P(x, y) \log \frac{P(x, y)}{P(y)} \\ &= - \sum_{x,y} P(x, y) \log P(x | y) \\ &= H(X|Y) \quad (\text{根据公式 (3.1)}) \end{aligned}$$

□

3.2 熵的性质

X 的熵具有如下几个性质

- 非负性： $H(X) \geq 0$ 。²
- 对称性：当随机变量的概率取值任意互换时，熵不变。

$$H(p_1, p_2 \dots p_n) = H(p_2, p_1 \dots p_n) = H(p_3, p_1 \dots p_n) = \dots$$

- 可加性：如果随机变量 X, Y 相互独立，则 $H(X, Y) = H(X) + H(Y)$ 。
- 极值性：对于任意概率分布 $P(X = x_i) = p_i$ 和 $P(Y = y_i) = q_i$ ， $i = 1 \dots n$ ，都有

$$H(X) = - \sum_{i=1}^n p_i \log p_i \leq - \sum_{i=1}^n p_i \log q_i \quad (3.5)$$

当 X 和 Y 的概率分布相同时，公式 (3.5) 取等号。

该性质表明，任意概率分布，它对其他概率分布的自信息取数学期望时，必大于它本身的熵。

- 凸性：对于任意概率分布 $P(X = x_i) = p_i$ 和 $P(Y = y_i) = q_i$ ， $i = 1 \dots n$ ，假设随机变量 Z 的分布为 $P(Z = z_i) = \gamma_i = \alpha p_i + (1 - \alpha) q_i$ ， $\alpha \in [0, 1]$ ，那么 Z 的熵满足

$$H(Z) \geq \alpha H(X) + (1 - \alpha) H(Y) \quad (3.6)$$

Theorem 3.2.1 (最大熵定理). 离散随机变量 X 的概率分布为 $P(X = x_i) = p_i$ ， $i = 1 \dots n$ ，那么

$$H(X) \leq \log n \quad (3.7)$$

当 $p_1 = p_2 = \dots = \frac{1}{n}$ 时，等号成立。

证明. 求熵的最大值等价于以下优化问题

$$\begin{aligned} \max \quad & H(x) = - \sum_{i=1}^n p_i \log p_i \\ \text{s.t.} \quad & \sum_{i=1}^n p_i = 1 \end{aligned}$$

利用拉格朗日乘子法构造函数

$$G(p, \lambda) = - \sum_{i=1}^n p_i \log p_i + \lambda \left(\sum_{i=1}^n p_i - 1 \right) \quad (3.8)$$

²实际上这种非负性对于离散随机变量 X 成立，对连续随机变量 X 不一定成立。这是本文只考虑离散情况的原因。

公式 (3.8) 中分别对 p_i 和 λ 求导，令其为零，得到

$$\begin{aligned} \frac{\partial G(p, \lambda)}{\partial p_i} &= -\log p_i - 1 + \lambda = 0 \\ \sum_{i=1}^n p_i - 1 &= 0 \end{aligned} \tag{3.9}$$

由 $-\log p_i - 1 + \lambda = 0$ 可得到 $p_i = e^{\lambda-1}, i = 1, 2, \dots, n$, 由此可知 $p_1 = p_2 = \dots = \frac{1}{n}$

□

Lemma 3.2.1 (熵的强可加性). 当随机变量 X, Y 相关的情况下，联合熵满足强可加性，即

$$\begin{aligned} H(X, Y) &= H(Y) + H(X|Y) \\ H(X, Y) &= H(X) + H(Y|X) \end{aligned} \tag{3.10}$$

证明.

$$\begin{aligned} H(Y) + H(X|Y) &= - \sum_y P(y) \log P(y) - \sum_x \sum_y P(x, y) \log P(x|y) \\ &= - \sum_x \sum_y P(x, y) \log P(y) - \sum_x \sum_y P(x, y) \log P(x|y) \\ &= - \sum_x \sum_y P(x, y) \log P(x, y) \\ &= H(X, Y) \end{aligned}$$

同理可证

$$H(X, Y) = H(X) + H(Y|X)$$

□

Lemma 3.2.2 (熵的凸性). 证明公式 (3.6)

证明.

$$\begin{aligned}
 H(Z) &= - \sum_{i=1}^n \gamma_i \log \gamma_i \\
 &= - \sum_{i=1}^n \alpha p_i \log \gamma_i - \sum_{i=1}^n (1 - \alpha) q_i \log \gamma_i \\
 &= - \sum_{i=1}^n \alpha p_i \log \left(\gamma_i \frac{p_i}{p_i} \right) - \sum_{i=1}^n (1 - \alpha) q_i \log \left(\gamma_i \frac{q_i}{q_i} \right) \\
 &= - \alpha \sum_{i=1}^n p_i \log p_i - (1 - \alpha) \sum_{i=1}^n q_i \log q_i - \alpha \sum_{i=1}^n p_i \log \frac{\gamma_i}{p_i} - (1 - \alpha) \sum_{i=1}^n q_i \log \frac{\gamma_i}{q_i} \\
 &= \alpha H(X) + (1 - \alpha) H(Y) - \alpha \sum_{i=1}^n p_i \log \frac{\gamma_i}{p_i} - (1 - \alpha) \sum_{i=1}^n q_i \log \frac{\gamma_i}{q_i}
 \end{aligned} \tag{3.11}$$

其中公式 (3.11) 的倒数第二项

$$\begin{aligned}
 -\alpha \sum_{i=1}^n p_i \log \frac{\gamma_i}{p_i} &= \alpha \left(- \sum_{i=1}^n p_i \log \gamma_i + \sum_{i=1}^n p_i \log p_i \right) \\
 &\geq 0 \quad (\text{根据公式 (3.5)})
 \end{aligned}$$

同理可知公式 (3.11) 的倒数第一项

$$-(1 - \alpha) \sum_{i=1}^n q_i \log \frac{\gamma_i}{q_i} \geq 0$$

所以得到

$$H(Z) \geq \alpha H(X) + (1 - \alpha) H(Y)$$

□

Theorem 3.2.2. 条件熵小于无条件熵，即 $H(X|Y) \leq H(X)$

证明.

$$\begin{aligned}
H(X|Y) - H(X) &= - \sum_{x,y} P(x,y) \log P(x|y) + \sum_x P(x) \log P(x) \\
&= - \sum_{x,y} P(x,y) \log P(x|y) + \sum_x \left(\sum_y P(x,y) \right) \log P(x) \\
&= - \sum_{x,y} P(x,y) \log P(x|y) + \sum_{x,y} P(x,y) \log P(x) \\
&= - \sum_{x,y} P(x,y) (\log P(x|y) - \log P(x)) \\
&= - \sum_{x,y} P(x,y) \log \frac{P(x,y)}{P(x)P(y)} \\
&= - \sum_{x,y} P(x,y) \log P(x,y) - \left(- \sum_{x,y} P(x,y) \log P(x)P(y) \right) \\
&\leq 0 \quad (\text{根据熵的极值性})
\end{aligned} \tag{3.12}$$

□

3.2.1 整理得到的公式

根据本节内容整理得到的重要公式

- 根据条件熵定义可得

$$H(X|Y) = H(X, Y) - H(Y) \tag{3.13}$$

- 根据互信息定义展开可得

$$H(X|Y) = H(X) - I(X, Y) \tag{3.14}$$

- 根据公式 (3.13) 和公式 (3.14) 得到的对偶形式

$$H(Y|X) = H(X, Y) - H(X)$$

$$H(Y|X) = H(Y) - I(X, Y)$$

- 多数文献将下式作为互信息的定义公式

$$I(X, Y) = H(X) + H(Y) - H(X, Y)$$

- $H(X|Y) \leq H(X)$

参考文献

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. ii

索引

Conditional independence, iv
Covariance, iv
Derivative, iv
Determinant, iii
Element-wise product, *see* Hadamard product
Graph, iii
Hadamard product, iii
Hessian matrix, iv
Independence, iv
Integral, iv
Jacobian matrix, iv
Kullback-Leibler divergence, iv
Matrix, ii, iii
Norm, v
Scalar, ii, iii
Set, iii
Shannon entropy, iv
Sigmoid, v
Softplus, v
Tensor, ii, iii
Transpose, iii
Variance, iv
Vector, ii, iii