

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好。

9 Generalized linear models and the exponential family

### 9.1 Introduction

We have now encountered a wide variety of probability distributions: the Gaussian, the Bernoulli, the Student t, the uniform, the gamma, etc. It turns out that most of these are members of a broader class of distributions known as the exponential family.<sup>1</sup> In this chapter, we discuss various properties of this family. This allows us to derive theorems and algorithms with very broad applicability.

We will see how we can easily use any member of the exponential family as a class-conditional density in order to make a generative classifier. In addition, we will discuss how to build discriminative models, where the response variable has an exponential family distribution, whose mean is a linear function of the inputs; this is known as a generalized linear model, and generalizes the idea of logistic regression to other kinds of response variables.

已思考 21s

下面先给出逐字完整翻译，然后是通俗且详尽的解释（配少量必需公式与例子，尽量不依赖高深数学）。

## 翻译

### 9 广义线性模型与指数族

#### 9.1 引言

我们已经遇到了各种概率分布：高斯分布、伯努利分布、学生 t 分布、均匀分布、伽马分布，等等。事实证明，这些分布中的大多数都属于一个更广泛的分布类别，称为**指数族**。<sup>1</sup> 在本章中，我们将讨论这个家族的各种性质。这使我们能够推导具有非常广泛适用性的定理和算法。

我们将看到如何可以很容易地把指数族中的任何成员用作**类条件密度**，从而构建一个**生成式分类器**。此外，我们还将讨论如何构建**判别式模型**：在这些模型中，**响应变量服从某个指数族分布，并且它的均值是输入的线性函数**；这被称为**广义线性模型**（GLM），它把**逻辑回归**的思想推广到其他类型的响应变量。

## 解释（基础且详尽）

### 1) 这些分布是什么？为什么要放在一起？

- 你已经见过的分布：
  - **高斯 (Gaussian)**：常见的“钟形曲线”，适合连续、近似对称的数据。
  - **伯努利 (Bernoulli)**：只有0/1两种结果，比如“是否点击”。
  - **学生 t (Student t)**：尾部更厚，抗异常值，用于连续数据。
  - **均匀 (Uniform)**：在某个区间内所有数值等可能。
  - **伽马 (Gamma)**：只取正值，经常建模时间/持续时长/正偏数据。
- **指数族**是一个“统一模板”，很多常见分布都能写成同一种数学形式。把它们放在一起，意味着我们可以用**一套通用的方法**来推导性质、做估计、设计算法。

小提示：并不是所有例子都严格属于指数族（例如一般形式的学生 t 分布就不是），所以原文用了“多数 (most) ”。

## 2) 指数族直观长什么样？

指数族分布可以写成下面这种“标准形”（了解直观即可，不必深究）：

$$p(y \mid \eta) = b(y) \exp(\eta^\top T(y) - A(\eta))$$

- $y$ : 随机变量（你要建模的量）。
- $T(y)$ : 把  $y$  做一个统计变换（称为“充分统计量”）。
- $\eta$ : “自然参数”（控制分布形状的参数）。
- $A(\eta)$ 、 $b(y)$ : 保证概率合法的函数。

许多常见分布（高斯、伯努利、二项、泊松、伽马等）都能写进这个框架里。

为什么有用？因为一旦写成这种形式，关于\*\*极大似然估计、凸优化、共轭先验（在贝叶斯里）\*\*等结论可以“一把通吃”。

## 3) “类条件密度”与生成式分类器

- **类条件密度**  $p(x|y=k)$ ：意思是“在类别等于  $k$  的前提下，特征  $x$  的分布”。
- 举例：识别垃圾邮件时，可分别建模

$$p(x \mid y = \text{垃圾}) \quad \text{和} \quad p(x \mid y = \text{正常})$$

这里  $x$  可以是词频、邮件长度等特征。

- \*\*生成式分类器 (generative classifier) \*\*怎么做？
  1. 为每个类  $k$  选一个分布（常从**指数族**里挑，例如多项式/伯努利用于词袋，或高斯用于连续特征），估计出  $p(x \mid y = k)$ 。
  2. 估计类先验  $p(y = k)$ 。
  3. 用贝叶斯法则算后验：

$$p(y = k | x) \propto p(x | y = k) p(y = k)$$

这样就能对新样本  $x$  进行分类。

典型例子：**朴素贝叶斯**（文本常用的多项式朴素贝叶斯、伯努利朴素贝叶斯）和**高斯判别分析**（GDA）。

#### 4) 判别式模型与广义线性模型 (GLM)

- 生成式是先学  $p(x | y)$  再反推  $p(y | x)$ ；  
**判别式是直接学  $p(y | x)$**  或直接学决策边界。
- 原文要讲的判别式思路是：让**响应变量  $y$**  的分布来自**指数族**，并且它的**均值**由输入  $x$  的**线性函数**来决定。  
这就是\*\*广义线性模型 (GLM)\*\* 的核心思想。
- 直观理解 GLM (用最口语化的说法)：
  - 我们有输入特征  $x$  (可以有很多维)，想预测输出  $y$ 。
  - 不同任务下， $y$  的“类型”不同：
    - 连续值 (身高、房价)
    - 二分类 (是/否)
    - 计数 (一天来客数)
  - GLM 规定：选一个**合适的指数族分布**来描述  $y$ ；然后让  $y$  的“平均水平” (均值) 跟  $x$  的**线性组合**相关。
- 实务上通常会用一个**链接函数**  $g$  把均值  $\mu = E[y | x]$  和线性项连起来：

$$g(\mu) = w^\top x$$

这样既能保证“线性可学”，又能让  $\mu$  落在合理范围 (比如概率必须在  $[0, 1]$ )。原文为口径简化成“均值是输入的线性函数”；理解成“(通过恰当链接) 均值线性”就好。

#### 常见 GLM 实例 (一看就懂的对照表)

任务类型	选的分布 (都属于指数族)	常用链接 $g(\mu)$	常见名称
预测连续值 (近似对称)	高斯	恒等 $g(\mu) = \mu$	线性回归
0/1 二分类	伯努利	logit: $g(\mu) = \log \frac{\mu}{1-\mu}$	逻辑回归
计数 (非负整数)	泊松	对数 $g(\mu) = \log \mu$	泊松回归
正值且右偏 (时间/花费)	伽马	常用对数或倒数链接	伽马回归

这也解释了原文最后一句：“GLM 把逻辑回归推广到其他类型的响应变量。”

## 5) 这一节要点小结

1. 指数族统一了许多常见分布 → 可以用一套理论与算法覆盖。
2. 用指数族做类条件密度，能方便地构建生成式分类器（如朴素贝叶斯、GDA）。
3. GLM：让响应变量服从指数族，且其均值（经适当链接）是输入的线性函数 → 逻辑回归只是 GLM 在伯努利分布上的一个特例。

<sup>1</sup> 注：原文脚注号表明“多数而非全部”属于指数族；例如常见形式的学生 t 分布并不属于自然指数族，这也正是“多数”的原因。

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好。

### 9.2 The exponential family

Before defining the exponential family, we mention several reasons why it is important:

- It can be shown that, under certain regularity conditions, the exponential family is the only family of distributions with finite-sized sufficient statistics, meaning that we can compress the data into a fixed-sized summary without loss of information. This is particularly useful for online learning, as we will see later.
- The exponential family is the only family of distributions for which conjugate priors exist, which simplifies the computation of the posterior (see Section 9.2.5).
- The exponential family can be shown to be the family of distributions that makes the least set of assumptions subject to some user-chosen constraints (see Section 9.2.6).
- The exponential family is at the core of generalized linear models, as discussed in Section 9.3.
- The exponential family is at the core of variational inference, as discussed in Section 21.2.

### 9.2.1 Definition

A pdf or pmf  $p(x|\theta)$ , for  $x = (x_1, \dots, x_m) \in X_m$  and  $\theta \in \Theta \subseteq R^d$ , is said to be in the exponential family if it is of the form

$$p(x|\theta) = \frac{1}{Z(\theta)} h(x) \exp[\theta^T \varphi(x)] \quad (9.1)$$

$$= h(x) \exp[\theta^T \varphi(x) - A(\theta)] \quad (9.2)$$

where

$$Z(\theta) = \int_X h(x) \exp[\theta^T \varphi(x)] dx \quad (9.3)$$

$$A(\theta) = \log Z(\theta) \quad (9.4)$$

Here  $\theta$  are called the natural parameters or canonical parameters,  $\varphi(x) \in R^d$  is called a vector of sufficient statistics,  $Z(\theta)$  is called the partition function,  $A(\theta)$  is called the log partition function or cumulant function, and  $h(x)$  is the scaling constant, often 1. If  $\varphi(x) = x$ , we say it is a natural exponential family. Equation 9.2 can be generalized by writing

$$p(x|\theta) = h(x) \exp[\eta(\theta)^T \varphi(x) - A(\eta(\theta))] \quad (9.5)$$

where  $\eta$  is a function that maps the parameters  $\theta$  to the canonical parameters  $\eta = \eta(\theta)$ . If  $\dim(\theta) < \dim(\eta(\theta))$ , it is called a curved exponential family, which means we have more sufficient statistics than parameters. If  $\eta(\theta) = \theta$ , the model is said to be in canonical form. We will assume models are in canonical form unless we state otherwise.

已思考 28s

下面先给出逐字完整翻译，然后提供基础且详尽的解释与示例（力求不跳步、不堆术语）。

## 逐字完整翻译

### 9.2 指数族 (The exponential family)

在给出指数族的定义之前，我们先提到它之所以重要的几个原因：

- 可以证明，在某些正则性条件下，指数族是唯一一类具有**有限维充分统计量**的分布族，这意味着我们可以把数据压缩成一个**固定大小**的摘要而不丢失信息。正如我们稍后将看到的，这对**在线学习**尤其有用。
- 指数族是唯一一类存在**共轭先验**的分布族，这会简化后验分布的计算（见第 9.2.5 节）。
- 可以证明，指数族是在一些由用户选择的约束下，**假设最少**的一类分布（见第 9.2.6 节）。
- 指数族是**广义线性模型**的核心，如第 9.3 节所述。
- 指数族是**变分推断**的核心，如第 21.2 节所述。

### 9.2.1 定义

若一个密度函数 (pdf) 或概率质量函数 (pmf)  $p(x | \theta)$  (其中  $x = (x_1, \dots, x_m) \in X^m$ ,  $\theta \in \Theta \subseteq \mathbb{R}^d$ ) 满足

$$p(x | \theta) = \frac{1}{Z(\theta)} h(x) \exp [\theta^\top \phi(x)] \quad (9.1)$$

$$= h(x) \exp [\theta^\top \phi(x) - A(\theta)] \quad (9.2)$$

则称其属于**指数族**；其中

$$Z(\theta) = \int_{X^m} h(x) \exp [\theta^\top \phi(x)] dx \quad (9.3)$$

$$A(\theta) = \log Z(\theta) \quad (9.4)$$

这里,  $\theta$  称为**自然参数或典范参数** (natural/canonical parameters),  $\phi(x) \in \mathbb{R}^d$  称为**充分统计量向量**,  $Z(\theta)$  称为**配分函数** (partition function),  $A(\theta)$  称为**对数配分函数或累积量函数** (cumulant function), 而  $h(x)$  是一个**缩放常数**, 通常取 1。若  $\phi(x) = x$ , 则称其为**自然指数族** (natural exponential family)。

式 (9.2) 可推广为

$$p(x | \theta) = h(x) \exp [\eta(\theta)^\top \phi(x) - A(\eta(\theta))] \quad (9.5)$$

其中  $\eta$  是把参数  $\theta$  映射到**典范参数**  $\eta = \eta(\theta)$  的函数。若  $\dim(\theta) < \dim(\eta(\theta))$ , 称其为**弯曲指数族** (curved exponential family), 这意味着**充分统计量的维度多于参数的维度**。若  $\eta(\theta) = \theta$ , 则称模型处于**典范形式** (canonical form)。除非另行说明, 我们将默认模型处于典范形式。

## 通俗解释 (配例子, 不跳步)

### 1) 开场白：这段话在讲什么？

- 很多常见分布都能写成统一的“指数—线性—对数配分”的模板 (式 9.1–9.5)。
- 只要写进这个模板, 我们就能共享一整套通用结论: 如何压缩数据 (充分统计量)、如何做贝叶斯 (共轭先验)、如何在给定约束下找“最不偏”的分布 (最大熵), 以及如何构建 GLM 和做变分推断。

### 2) 为什么它重要? (逐条翻译补充)

## (a) 有限维充分统计量（能把数据无损压缩成固定长度）

直观理解：

- 你观察到  $n$  个样本  $x_1, \dots, x_n$ 。如果分布在指数族里，那么你不必记住所有样本，**只需记住某些“总结量”**（比如  $\sum x_i$ 、 $\sum x_i^2$  或每一类的计数等），这些就是**充分统计量**。
- 以后再做推断（估计参数、算后验），只看这些总结量就够了，不会损失关于参数的任何信息。
- 好处：**在线学习时**，你只更新这几个“**汇总量**”，而不是保存全量数据。

### 常见例子

- 伯努利/二项：充分统计量是“成功次数和失败次数”（本质上是  $\sum x_i$ ）。
- 泊松：充分统计量是“总计数  $\sum x_i$ ”。
- 高斯（均值与方差未知）：充分统计量是“ $\sum x_i$ ”和“ $\sum x_i^2$ ”。

## (b) 共轭先验只在指数族里系统存在

- 在贝叶斯里，先验  $p(\theta)$  与似然  $p(x | \theta)$  相乘得后验  $p(\theta | x)$ 。
- 若先验和后验**同属一个分布族**，就叫**共轭**。优点是后验形式简单、参数只需“加数据的贡献”就能更新。
- 经典对：
  - 伯努利/二项的先验—Beta，后验仍是 Beta；
  - 泊松的先验—Gamma，后验仍是 Gamma；
  - 高斯均值（方差已知）的先验—高斯，后验仍是高斯。
- 指数族给了我们这类“可闭合”的先验—后验配对。

## (c) \*\*最小假设（最大熵）\*\*的由来

- 如果你只知道某些约束（如某些期望值  $E[\phi_k(x)]$ ），在不做更多假设的前提下，**熵最大的分布恰好是指数族形式**。
- 这在第 9.2.6 节会展开：给定“我只相信这些约束”，最中立的选择就是指数族。

## (d) GLM 的核心

- GLM 假设： $y$  的分布属于指数族，且其均值与特征  $x$  的线性函数（通过链接函数）相关。
- 所以逻辑回归、泊松回归、Gamma 回归等，都在一个统一框架里。

## (e) 变分推断的核心

- 变分推断常用**指数族**作为近似族，因为对数配分函数  $A(\cdot)$  有漂亮的性质（凸、梯度等式），便于优化与更新。

### 3) 定义细读：式 (9.1) — (9.5) 讲了什么？

#### 元素表

- $x$ : 数据（可能是一维，也可能是向量或样本集）。
- $\theta$ : 参数（实数向量  $R^d$ ）。
- $\phi(x)$ : 充分统计量向量（把数据“提炼”成固定长度的量）。
- $h(x)$ : 基测度/缩放因子（很多分布里它就是 1，或像  $1/x!$  这样的已知因子）。
- $\theta^\top \phi(x)$ : 线性项（这就是“指数族”的“指数里是线性”）。
- $Z(\theta)$ : 配分函数，确保整个分布能积分为 1；
- $A(\theta) = \log Z(\theta)$ : 对数配分函数，把“归一化常数”放到指数外，使表达更简洁。

式 (9.1) 与 (9.2) 是等价写法：

- (9.1) 把“归一化常数”称为  $1/Z(\theta)$ 。
- (9.2) 把它改写成  $\exp[-A(\theta)]$ （因为  $A = \log Z$ ）。

直观比喻： $\theta^\top \phi(x)$  决定了“喜欢哪些  $x$ ”，而  $Z(\theta)$  则是把整体“调准”成一个合法的概率分布。

#### 自然指数族 ( $\phi(x) = x$ )

- 若充分统计量就是数据本身（或简单堆叠），叫“自然指数族”。这时“指数里是  $\theta^\top x$ ”。

#### 典范参数与弯曲指数族

- 有时我们用的参数  $\theta$  不等于“指数里的那个参数”。
- 这时引入映射  $\eta(\theta)$ （式 9.5）， $\eta$  叫典典范数（canonical）。
- 若  $\dim(\theta) < \dim(\eta)$ ，称为弯曲指数族：意思是“指数里的线性项需要更多维的充分统计量，但你实际只用少数几个自由参数来控制它们”，等价于“在高维的自然参数空间里走一条弯曲的低维流形上”。
- 若  $\eta(\theta) = \theta$ ，就叫典典范形式（最干净、最常用），本文默认用它。

### 4) 三个“把常见分布写进指数族”的具体例子（一步步拆）

记法：我们都写成  $p(x | \theta) = h(x) \exp\{\theta^\top \phi(x) - A(\theta)\}$ 。

#### 例 1：伯努利 (0/1)

- 原始： $p(x = 1 | p) = p, p(x = 0 | p) = 1 - p$ 。合起来： $p(x | p) = p^x(1 - p)^{1-x}$ 。
- 改写：

$$p(x | p) = \exp \{x \log p + (1-x) \log(1-p)\}$$

$$= \exp \left\{ x \log \frac{p}{1-p} + \log(1-p) \right\}$$

令  $\theta = \log \frac{p}{1-p}$  (logit),  $\phi(x) = x$ ,  $h(x) = 1$ 。

还需让常数项变成  $-A(\theta)$ :

$$A(\theta) = \log(1 + e^\theta)$$

**总结:**

$$p(x | \theta) = \exp \{\theta x - A(\theta)\}, \quad A(\theta) = \log(1 + e^\theta)$$

这正是逻辑回归里出现的 logit/softplus 结构。

### 例 2: 泊松 (计数 0,1,2,...)

- 原始:  $p(x | \lambda) = \frac{e^{-\lambda} \lambda^x}{x!}$ 。
- 改写:

$$= \frac{1}{x!} \exp \{x \log \lambda - \lambda\}$$

令  $\theta = \log \lambda$ ,  $\phi(x) = x$ ,  $h(x) = \frac{1}{x!}$ , 则

$$A(\theta) = e^\theta \text{ (因为 } Z = \sum_x \frac{e^{\theta x}}{x!} e^{-e^\theta} = e^A\text{)}$$

**总结:**

$$p(x | \theta) = \frac{1}{x!} \exp \{\theta x - e^\theta\}$$

这就是泊松回归中“对数链接”的来源:  $\log \lambda = \theta$ 。

### 例 3: 高斯 (已知方差 $\sigma^2$ )

- 原始:  $N(\mu, \sigma^2)$ :

$$p(x | \mu) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

展开指数:  $-\frac{x^2}{2\sigma^2} + \frac{\mu}{\sigma^2}x - \frac{\mu^2}{2\sigma^2}$ 。

把只含  $x$  的项分组, 可写成

$$p(x | \mu) = h(x) \exp \{\theta \phi(x) - A(\theta)\}$$

其中

$$\phi(x) = x, \quad \theta = \frac{\mu}{\sigma^2}, \quad h(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right), \quad A(\theta) = \frac{\sigma^2\theta^2}{2}$$

如果**均值和方差都未知**，则  $\phi(x)$  需要两维（如  $(x, x^2)$ ），相应地  $\theta$  也有两维（如  $(\mu/\sigma^2, -1/(2\sigma^2))$ ）。

## 5) 配分函数与对数配分函数的直观作用

- $Z(\theta)$  的唯一任务：\*\*把概率“归一化”\*\*到能积分（或求和）为 1。
- 用  $A(\theta) = \log Z(\theta)$  是为了让指数里只出现“ $\theta^\top \phi(x) - A(\theta)$ ”这一个整体，便于推导。
- （超纲但有用的直觉）在指数族中，

$$\nabla_\theta A(\theta) = E_\theta[\phi(x)]$$

即：对数配分函数的梯度就是充分统计量的期望，这让**极大似然/贝叶斯更新/变分优化**都很顺手。

## 6) 典范形式、 $\eta(\theta)$ 与“弯曲”

- **典范形式**： $\eta(\theta) = \theta$ 。最常见、最方便，本文默认如此。
- **一般形式**：有时我们习惯的参数（如  $\mu, \sigma^2$ ）并不是指数里自然出现的那个；这时先做变换  $\eta(\theta)$ （如  $\theta = \mu/\sigma^2$ ）。
- **弯曲指数族** ( $\dim \theta < \dim \eta$ )：想象自然参数需要两维  $(\eta_1, \eta_2)$ ，但我们人为施加了一个关系（例如  $\eta_2 = f(\eta_1)$ ），只用一维  $\theta$  来描述，于是可达的  $(\eta_1, \eta_2)$  只是一条曲线。
  - 例子（直观）：一类高斯分布把方差设为均值的函数（如  $\sigma^2 = \mu^2$ ），这会让“自然参数空间”里的可达集合不再是整个平面，而是一条曲线，从而形成“弯曲”。

## 7) 把“重要性清单”再落地一遍

- **无损压缩**：只存  $\sum x_i, \sum x_i^2 \dots$  就能完成估计，内存友好、在线更新快。
- **共轭先验**：贝叶斯更新像“加法”，例如
  - 二项+Beta： $\alpha' = \alpha + \sum x_i, \beta' = \beta + n - \sum x_i$ ；
  - 泊松+Gamma：形状/率参数加上“总计数/样本量”的贡献。
- **最大熵**：只相信“这些期望值是对的”，那就选指数族；不多加假设。

- **GLM**: 统一回归模型的母体 (逻辑回归、泊松回归、Gamma 回归...)。
- **变分推断**: 指数族的形式使近似后验的优化有“可微、凸、可分解”的好处。

## 8) 极简“心智图”

1. **形式**:  $p(x | \theta) = h(x) \exp\{\theta^\top \phi(x) - A(\theta)\}$ 。
2. **关键词**: 充分统计量  $\phi(x)$ 、(对数) 配分  $A(\theta)$ 、典范参数  $\theta$ 。
3. **收益**: 无损压缩、共轭先验、最大熵、GLM、变分推断。
4. **例子**: 伯努利 ( $\theta = \log \frac{p}{1-p}$ 、 $A = \log(1 + e^\theta)$ )、泊松 ( $\theta = \log \lambda$ 、 $A = e^\theta$ )、高斯 ( $\theta = \mu/\sigma^2$ 、 $A = \sigma^2 \theta^2/2$ )。

到这里，你已经能把绝大多数常见分布认出来、装进指数族框架里，并理解“为什么这很有用”。

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好。

### 9.2.2 Examples

Let us consider some examples to make things clearer.

#### 9.2.2.1 Bernoulli

The Bernoulli for  $x \in \{0, 1\}$  can be written in exponential family form as follows:

$$\text{Ber}(x|\mu) = \mu x (1 - \mu)^{1-x} = \exp[x \log(\mu) + (1 - x) \log(1 - \mu)] = \exp[\varphi(x)^\top \theta] \quad (9.6)$$

where  $\varphi(x) = [I(x=0), I(x=1)]$  and  $\theta = [\log(\mu), \log(1 - \mu)]$ .

However, this representation is over-complete since there is a linear dependence between the features:

$$1^\top \varphi(x) = I(x=0) + I(x=1) = 1 \quad (9.7)$$

Consequently  $\theta$  is not uniquely identifiable. It is common to require that the representation be minimal, which means there is a unique  $\theta$  associated with the distribution. In this case, we can just define

$$\text{Ber}(x|\mu) = (1 - \mu) \exp x \log \mu / 1 - \mu \quad (9.8)$$

Now we have  $\varphi(x) = x$ ,  $\theta = \log \mu / 1 - \mu$ , which is the log-odds ratio, and  $Z = 1/(1 - \mu)$ . We can recover the mean parameter  $\mu$  from the canonical parameter using

$$\mu = \text{sigm}(\theta) = 1 / (1 + e^{-\theta}) \quad (9.9)$$

### 9.2.2.2 Multinoulli

We can represent the multinoulli as a minimal exponential family as follows (where  $x_k = I(x = k)$ ):

$$\text{Cat}(x|\mu) = \prod_{k=1}^K \mu_k^{x_k} = \exp \left( \sum_{k=1}^K x_k \log \mu_k \right) \quad (9.10)$$

$$= \exp \left( \sum_{k=1}^{K-1} x_k \log \mu_k + 1 - \sum_{k=1}^{K-1} x_k \log (1 - \sum_{j=1}^{K-1} \mu_j) \right) \quad (9.11)$$

$$= \exp \left( \sum_{k=1}^{K-1} x_k \log \mu_k + \sum_{j=1}^{K-1} \mu_j + \log (1 - \sum_{k=1}^{K-1} \mu_k) \right) \quad (9.12)$$

$$= \exp \left( \sum_{k=1}^{K-1} x_k \log \mu_k + \log \mu_K \right) \quad (9.13)$$

where  $\mu_K = 1 - \sum_{k=1}^{K-1} \mu_k$ . We can write this in exponential family form as follows:

$$\text{Cat}(x|\theta) = \exp(\theta^\top \varphi(x) - A(\theta)) \quad (9.14)$$

$$\theta = [\log \mu_1, \dots, \log \mu_{K-1}, \log \mu_K] \quad (9.15)$$

$$\varphi(x) = [I(x=1), \dots, I(x=K-1)] \quad (9.16)$$

We can recover the mean parameters from the canonical parameters using

$$\mu_k = e^{\theta_k} / \sum_{j=1}^{K-1} e^{\theta_j} \quad (9.17)$$

From this, we find

$$\mu_K = 1 - \sum_{j=1}^{K-1} e^{\theta_j} / \sum_{j=1}^{K-1} e^{\theta_j} = 1 / \sum_{j=1}^{K-1} e^{\theta_j} \quad (9.18)$$

and hence

$$A(\theta) = \log \sum_{j=1}^{K-1} e^{\theta_j} + \log \mu_K \quad (9.19)$$

If we define  $\theta_K = 0$ , we can write  $\mu = S(\theta)$  and  $A(\theta) = \log \sum_{k=1}^K e^{\theta_k}$ , where  $S$  is the softmax function in Equation 4.39.

已思考 49s

下面先给出逐字完整翻译（公式尽量按照原意排版，内容不缺漏），随后提供基础且详尽的解释（逐步推导+小例子，尽量不跳步）。

(注：原文个别地方排版较紧，我在解释部分会把关键步骤展开，并指出可能的易混点。)

## 逐字完整翻译

### 9.2.2 例子 (Examples)

让我们通过一些例子把问题讲清楚。

#### 9.2.2.1 伯努利 (Bernoulli)

对于  $x \in \{0, 1\}$  的伯努利分布，可以写成指数族形式如下：

$$\text{Ber}(x | \mu) = \mu^x (1 - \mu)^{1-x} = \exp[x \log(\mu) + (1 - x) \log(1 - \mu)] = \exp[\phi(x)^\top \theta] \quad (9.6)$$

其中  $\phi(x) = [\mathbf{I}(x=0), \mathbf{I}(x=1)]$ ,  $\theta = [\log(\mu), \log(1-\mu)]$ 。然而, 这个表示是过完备的 (over-complete), 因为这些特征之间存在线性相关:

$$\mathbf{1}^\top \phi(x) = \mathbf{I}(x=0) + \mathbf{I}(x=1) = 1 \quad (9.7)$$

因此  $\theta$  不是唯一可识别的。通常我们要求表示是极小 (minimal) 的, 这意味着对于该分布有唯一的  $\theta$ 。在这种情况下, 我们可以直接定义

$$\text{Ber}(x | \mu) = (1 - \mu) \exp[x \log \frac{\mu}{1 - \mu}] \quad (9.8)$$

此时我们有  $\phi(x) = x$ 、 $\theta = \log \frac{\mu}{1 - \mu}$  (即对数赔率), 并且  $Z = 1/(1 - \mu)$ 。我们可以从典范参数恢复均值参数  $\mu$ :

$$\mu = \text{sigm}(\theta) = \frac{1}{1 + e^{-\theta}} \quad (9.9)$$

### 9.2.2.2 多项 (Multinoulli / Categorical)

我们可以把多项 (分类) 分布表示为一个极小指数族, 如下 (其中  $x_k = \mathbf{I}(x=k)$ ) :

$$\text{Cat}(x | \mu) = \prod_{k=1}^K \mu_k^{x_k} = \exp\left(\sum_{k=1}^K x_k \log \mu_k\right) \quad (9.10)$$

$$= \exp\left(\sum_{k=1}^{K-1} x_k \log \mu_k + (1 - \sum_{k=1}^{K-1} x_k) \log(1 - \sum_{k=1}^{K-1} \mu_k)\right) \quad (9.11)$$

$$= \exp\left(\sum_{k=1}^{K-1} x_k \log \frac{\mu_k}{1 - \sum_{j=1}^{K-1} \mu_j} + \log(1 - \sum_{k=1}^{K-1} \mu_k)\right) \quad (9.12)$$

$$= \exp\left(\sum_{k=1}^{K-1} x_k \log \frac{\mu_k}{\mu_K} + \log \mu_K\right) \quad (9.13)$$

其中  $\mu_K = 1 - \sum_{k=1}^{K-1} \mu_k$ 。我们可以把它写成指数族形式如下:

$$\text{Cat}(x | \theta) = \exp(\theta^\top \phi(x) - A(\theta)) \quad (9.14)$$

$$\theta = [\log \frac{\mu_1}{\mu_K}, \dots, \log \frac{\mu_{K-1}}{\mu_K}] \quad (9.15)$$

$$\phi(x) = [\mathbf{I}(x=1), \dots, \mathbf{I}(x=K-1)] \quad (9.16)$$

我们可以从典范参数恢复均值参数:

$$\mu_k = \frac{e^{\theta_k}}{1 + \sum_{j=1}^{K-1} e^{\theta_j}} \quad (9.17)$$

由此得到

$$\mu_K = 1 - \frac{\sum_{j=1}^{K-1} e^{\theta_j}}{1 + \sum_{j=1}^{K-1} e^{\theta_j}} = \frac{1}{1 + \sum_{j=1}^{K-1} e^{\theta_j}} \quad (9.18)$$

从而

$$A(\theta) = \log\left(1 + \sum_{k=1}^{K-1} e^{\theta_k}\right) \quad (9.19)$$

如果我们定义  $\theta_K = 0$ , 就可以写成  $\mu = S(\theta)$  且

$$A(\theta) = \log\left(\sum_{k=1}^K e^{\theta_k}\right),$$

其中  $S$  是式 (4.39) 中的 softmax 函数。

---

## 通俗解释 (一步步来)

### A. 伯努利作为指数族 (9.2.2.1)

#### 1) 把熟悉的伯努利改写成“指数里线性”的样子

- 原始形式:  $\Pr(x = 1) = \mu, \Pr(x = 0) = 1 - \mu$ , 合写为  
 $\text{Ber}(x | \mu) = \mu^x(1 - \mu)^{1-x}$ 。
- 取对数并放到指数里 (9.6) :  
 $\log p = x \log \mu + (1 - x) \log(1 - \mu)$ 。  
这说明“指数中的表达式”是关于  $x$  的线性函数 (常数项 +  $x$  的系数)。

#### 2) 为什么说“过完备” (over-complete) ?

- 若取  $\phi(x) = [\mathbf{I}(x = 0), \mathbf{I}(x = 1)]$ , 它们必有  
 $\mathbf{I}(x = 0) + \mathbf{I}(x = 1) = 1$  (式 9.7)。
- 两个“特征”线性相关  $\Rightarrow \theta$  不唯一 (可加同一常数、再整体归一化, 概率不变)。
- 为了唯一性, 我们希望“极小表示” (minimal) : 只保留必要的一维信息。

#### 3) 极小表示怎么写 (式 9.8) ?

- 把“是否为1”作为充分统计量:  $\phi(x) = x$  (只要 0 或 1)。
- 取典范参数 (自然参数) :  $\theta = \log \frac{\mu}{1-\mu}$  (对数赔率, log-odds)。
- 于是

$$p(x) = (1 - \mu) \exp(\theta x), \quad \theta = \log \frac{\mu}{1 - \mu}.$$

验证：

- 若  $x = 0$ :  $p(0) = (1 - \mu)$ ;
- 若  $x = 1$ :  $p(1) = (1 - \mu)e^\theta = (1 - \mu)\frac{\mu}{1-\mu} = \mu$ 。  
正好复原原式。
- 与一般指数族  $p = \frac{1}{Z(\theta)} h(x) \exp(\theta\phi(x))$  对照：此处  $h(x) = 1$ ,

$$Z(\theta) = \frac{1}{1 - \mu} = 1 + e^\theta \quad (\text{因为 } 1 - \mu = \frac{1}{1+e^\theta}).$$

若改写成  $p(x) = \exp\{\theta x - A(\theta)\}$ , 则  $A(\theta) = \log(1 + e^\theta)$ 。

#### 4) 从 $\theta$ 反求 $\mu$ : Sigmoid (式 9.9)

$$\mu = \frac{1}{1 + e^{-\theta}} = \text{sigmoid}(\theta).$$

这就是逻辑回归里链接函数的来历：线性部分给出  $\theta$ , 通过 Sigmoid 转成  $[0, 1]$  的概率。

#### 5) 一个小算例

- 设  $\mu = 0.3$ 。则  $\theta = \log \frac{0.3}{0.7} \approx \log(0.4286) \approx -0.847$  (负数, 表示“更偏向 0”)
  - 用 Sigmoid 反推:  $\text{sigmoid}(-0.847) \approx 0.3$ , 回到原来的  $\mu$ 。
- 

### B. 多项 (分类) 作为指数族 (9.2.2.2)

#### 1) one-hot 表示与“过完备”的由来

- 令  $x \in \{1, \dots, K\}$  表示类别。定义 one-hot:  $x_k = \mathbf{I}(x = k)$ 。  
总有  $\sum_{k=1}^K x_k = 1 \Rightarrow$  向量  $(x_1, \dots, x_K)$  的分量线性相关, 直接用它会“过完备”。
- 为了“极小表示”, 只保留前  $K - 1$  个指示量作为充分统计量。

#### 2) 从原式到极小指数族 (式 9.10 → 9.13)

- 原式 (多项/分类) :

$$p(x | \mu) = \prod_{k=1}^K \mu_k^{x_k} = \exp\left(\sum_{k=1}^K x_k \log \mu_k\right).$$

- 把第  $K$  类单独拿出来:  $\mu_K = 1 - \sum_{k=1}^{K-1} \mu_k$ , 且  $x_K = 1 - \sum_{k=1}^{K-1} x_k$ 。  
代入并整理得到 (9.11) → (9.12) → (9.13) :

$$p(x) = \exp\left(\sum_{k=1}^{K-1} x_k \log \frac{\mu_k}{\mu_K} + \log \mu_K\right).$$

这就只用到了  $K - 1$  个  $x_k$ 。

### 3) 典范参数、充分统计量与对数配分 (式 9.14–9.19)

- 取

$$\phi(x) = [\mathbf{I}(x=1), \dots, \mathbf{I}(x=K-1)], \quad \theta_k = \log \frac{\mu_k}{\mu_K} \ (k = 1, \dots, K-1).$$

- 于是

$$p(x | \theta) = \exp(\theta^\top \phi(x) - A(\theta)),$$

其中

$$A(\theta) = \log\left(1 + \sum_{k=1}^{K-1} e^{\theta_k}\right).$$

- 从  $\theta$  还原  $\mu$  (式 9.17–9.18) :

$$\mu_k = \frac{e^{\theta_k}}{1 + \sum_{j=1}^{K-1} e^{\theta_j}}, \quad \mu_K = \frac{1}{1 + \sum_{j=1}^{K-1} e^{\theta_j}}.$$

- 若再定义  $\theta_K = 0$ , 就把“+1”也写成  $e^{\theta_K}$ 。

此时

$$\mu_k = \frac{e^{\theta_k}}{\sum_{j=1}^K e^{\theta_j}} \quad (\text{对所有 } k = 1, \dots, K),$$

这正是 softmax:  $\mu = \text{softmax}(\theta)$ , 而

$$A(\theta) = \log\left(\sum_{k=1}^K e^{\theta_k}\right).$$

小提示：也有教材选择另一种“去冗余”的方式（例如强制  $\sum_k \theta_k = 0$ ）。本书采用“把最后一类当作基准： $\theta_K = 0$ ”的方式，直观好用。

### 4) 一个 3 类的小算例

- 设  $\mu = (0.1, 0.3, 0.6)$ , 令第 3 类为基准  $\Rightarrow$   
 $\theta_1 = \log(\mu_1/\mu_3) = \log(0.1/0.6) = \log(1/6) \approx -1.7918$ ;  
 $\theta_2 = \log(0.3/0.6) = \log(1/2) \approx -0.6931$ 。
- 反向用 (9.17–9.18) :  
 $\sum e^{\theta_j} = e^{-1.7918} + e^{-0.6931} \approx 0.1667 + 0.5 = 0.6667$ 。

$$\mu_1 = \frac{e^{\theta_1}}{1+0.6667} \approx 0.1667/1.6667 = 0.1;$$

$$\mu_2 = \frac{0.5}{1.6667} = 0.3;$$

$$\mu_3 = \frac{1}{1.6667} = 0.6.$$

完全复原。

---

## C. 关键概念再梳理

### 1. 过完备 vs 极小表示

- 过完备：特征之间线性相关（如伯努利的两项指示、分类的  $K$  个 one-hot），导致参数不唯一。
- 极小表示：去掉冗余，只保留**线性无关**的充分统计量（伯努利保留  $x$ ，分类保留前  $K - 1$  个指标）。

### 2. 从“典范参数”到“均值参数”

- 伯努利： $\mu = \text{sigmoid}(\theta)$ 。
- 多项： $\mu = \text{softmax}(\theta)$ （在把基准类  $\theta_K$  设为 0 的写法下就是标准 softmax）。

### 3. 对数配分函数 $A(\theta)$

- 伯努利： $A(\theta) = \log(1 + e^\theta)$ 。
  - 多项（极小写法）： $A(\theta) = \log(1 + \sum_{k=1}^{K-1} e^{\theta_k})$ ；  
若设  $\theta_K = 0$ ： $A(\theta) = \log(\sum_{k=1}^K e^{\theta_k})$ 。
  - 它保证“归一化为 1”，并在优化/推断里非常好用（例如  $\nabla_\theta A$  给出充分统计量的期望）。
- 

## D. 你需要记住的最少要点

- **伯努利**： $\phi(x) = x$ ,  $\theta = \log \frac{\mu}{1-\mu}$ ,  $\mu = \text{sigmoid}(\theta)$ ,  $A(\theta) = \log(1 + e^\theta)$ 。
- **多项/分类**： $\phi(x) = [\mathbf{I}(x = 1), \dots, \mathbf{I}(x = K - 1)]$ ,  $\theta_k = \log \frac{\mu_k}{\mu_K}$ ,  $\mu = \text{softmax}(\theta)$   
(取  $\theta_K = 0$ )， $A(\theta) = \log \sum_k e^{\theta_k}$ 。
- **极小表示**能让参数**可识别**、推导更干净；Sigmoid/Softmax 就是“从自然参数到概率”的桥梁。

请总结9.2.2的要点

已思考 13s

下面是 9.2.2 的精炼要点 (按两种分布分别总结) :

## 伯努利 (9.2.2.1)

- 原式:  $\text{Ber}(x | \mu) = \mu^x(1 - \mu)^{1-x}$ 。
- 过完备表示:  $\phi(x) = [\mathbf{I}(x = 0), \mathbf{I}(x = 1)]$ 、 $\theta = [\log \mu, \log(1 - \mu)]$ , 但  $\mathbf{I}(x = 0) + \mathbf{I}(x = 1) = 1 \Rightarrow$  参数不唯一。
- 极小表示 (可识别) :**

$$\phi(x) = x, \quad \theta = \log \frac{\mu}{1-\mu} \quad (\text{log-odds}),$$

$$p(x) = \exp\{\theta x - A(\theta)\}, \quad \text{其中 } A(\theta) = \log(1 + e^\theta) \quad (\text{对数配分}) ;$$
等价地  $p(x) = (1 - \mu) \exp\{\theta x\}, \quad Z = 1/(1 - \mu)$ 。
- 从自然参数到均值:**  $\mu = \sigma(\theta) = \frac{1}{1 + e^{-\theta}}$  (Sigmoid) 。
   
 $\Rightarrow$  逻辑回归 = 伯努利 + 指数族 + 线性  $\rightarrow$  Sigmoid。

## 多项/分类 (9.2.2.2)

- 设 one-hot:  $x_k = \mathbf{I}(x = k)$ ,  $\sum_k x_k = 1$ 。原式:
 
$$\text{Cat}(x | \mu) = \prod_{k=1}^K \mu_k^{x_k} = \exp(\sum_k x_k \log \mu_k)$$
- 为去冗余, 取第  $K$  类为基准 ( $\mu_K = 1 - \sum_{k=1}^{K-1} \mu_k$ ) , 只保留前  $K - 1$  个指示:

$$\phi(x) = [\mathbf{I}(x = 1), \dots, \mathbf{I}(x = K - 1)], \quad \theta_k = \log \frac{\mu_k}{\mu_K}.$$

- 指数族形式:  $p(x) = \exp\{\theta^\top \phi(x) - A(\theta)\}$ ,

$$A(\theta) = \log(1 + \sum_{k=1}^{K-1} e^{\theta_k})$$

若再定义  $\theta_K = 0$ , 则

$$A(\theta) = \log \sum_{k=1}^K e^{\theta_k}$$

- 从自然参数到均值:**

$$\mu_k = \frac{e^{\theta_k}}{1 + \sum_{j=1}^{K-1} e^{\theta_j}},$$

$$\mu_K = \frac{1}{1 + \sum_{j=1}^{K-1} e^{\theta_j}};$$

等价地 ( $\theta_K = 0$ )  $\mu = \text{softmax}(\theta)$ 。

$\Rightarrow$  softmax 回归 = 多项分布 + 指数族 + 线性  $\rightarrow$  Softmax。

## 核心共同点

- 极小表示**避免参数不可识别 (去掉线性相关的特征) 。
- 对数配分**  $A(\theta)$  统一负责“归一化”, 其形式直接导出 Sigmoid (伯努利) 与 Softmax (多项) 。

- 这两例展示了“从自然参数到概率”的标准桥梁：Sigmoid/Softmax，是 GLM 与判别模型（逻辑/softmax 回归）的基础。

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好，最后总结要点。

### 9.2.2.3 Univariate Gaussian

The univariate Gaussian can be written in exponential family form as follows:

$$N(x|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp[-\frac{1}{2\sigma^2}(x - \mu)^2] \quad (9.20)$$

$$= \frac{1}{(2\pi\sigma^2)^{1/2}} \exp[-\frac{1}{2\sigma^2}x^2 + \mu\sigma^2x - \frac{1}{2\sigma^2}\mu^2] \quad (9.21)$$

$$= Z(\theta) \exp(\theta T\varphi(x)) \quad (9.22)$$

where

$$\theta = \mu/\sigma^2 - \frac{1}{2\sigma^2} \quad (9.23)$$

$$\varphi(x) = x^2 \quad (9.24)$$

$$Z(\mu, \sigma^2) = \sqrt{2\pi\sigma^2} \exp[-\frac{\mu^2}{2\sigma^2}] \quad (9.25)$$

$$A(\theta) = -\frac{1}{2} \ln(2\pi\sigma^2) - \frac{1}{2} \ln(\theta^2) - \frac{1}{2} \ln(2\pi) \quad (9.26)$$

### 9.2.2.4 Non-examples

Not all distributions of interest belong to the exponential family. For example, the uniform distribution,  $X \sim \text{Unif}(a, b)$ , does not, since the support of the distribution depends on the parameters. Also, the Student T distribution (Section 11.4.5) does not belong, since it does not have the required form.

已思考 43s

下面先给出**逐字完整翻译**（尽量保持原式编号与符号，一字不漏），然后给出**基础且详尽的解释**（把推导步骤摊开、指出易混点），最后给出**要点总结**。

（提示：原文中个别公式有明显排版/笔误，我会在解释里标注“更正/常见写法”，以免误导。）

## 逐字完整翻译

### 9.2.2.3 一元高斯 (Univariate Gaussian)

一元高斯可以写成如下的指数族形式：

$$N(x | \mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp[-\frac{1}{2\sigma^2}(x - \mu)^2] \quad (9.20)$$

$$= \frac{1}{(2\pi\sigma^2)^{\frac{1}{2}}} \exp\left[-\frac{1}{2\sigma^2}x^2 + \frac{\mu}{\sigma^2}x - \frac{1}{2\sigma^2}\mu^2\right] \quad (9.21)$$

$$= \frac{1}{Z(\theta)} \exp(\theta^\top \phi(x)) \quad (9.22)$$

其中

$$\theta = \begin{pmatrix} \mu \\ \sigma^2 \end{pmatrix} \quad (9.23)$$

$$\phi(x) = \begin{pmatrix} x \\ x^2 \end{pmatrix} \quad (9.24)$$

$$Z(\mu, \sigma^2) = \sqrt{2\pi\sigma^2} \exp\left[\frac{\mu^2}{2\sigma^2}\right] \quad (9.25)$$

$$A(\theta) = -\frac{\theta_1^2}{4\theta_2} - \frac{1}{2} \log(-2\theta_2) - \frac{1}{2} \log(2\pi) \quad (9.26)$$

#### 9.2.2.4 非例子 (Non-examples)

并非所有感兴趣的分布都属于指数族。例如，均匀分布  $X \sim \text{Unif}(a, b)$  不属于，因为该分布的**支持集**依赖于参数。另外，学生  $t$  分布（第 11.4.5 节）也不属于，因为它不具有所需的形式。

说明（不改变原文内容）：式 (9.25) 常见且自洽的写法应为

$$Z(\mu, \sigma^2) = \sqrt{2\pi\sigma^2} \exp(\mu^2/(2\sigma^2)).$$

亦即根号下应为  $\sigma^2$ ；上式与 (9.20)–(9.22) 才能严格配平。原文“ $\sqrt{2\pi\sigma}$ ”更像是排版小误。

## 通俗解释（逐步、不跳步）

### A. 为什么高斯能写成指数族？

#### 1. 从标准高斯开始（式 9.20）

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

#### 2. 把平方展开（式 9.21）

$$(x-\mu)^2 = x^2 - 2\mu x + \mu^2 \Rightarrow \log p(x) = -\frac{1}{2\sigma^2}x^2 + \frac{\mu}{\sigma^2}x - \frac{\mu^2}{2\sigma^2} - \frac{1}{2} \log(2\pi\sigma^2).$$

把与  $x$  有关的两项凑成“线性 + 二次”：

$$\underbrace{\frac{\mu}{\sigma^2}x}_{\theta_1} + \underbrace{(-\frac{1}{2\sigma^2})x^2}_{\theta_2}.$$

余下的是常数（与  $x$  无关）。

### 3. 识别指数族结构

指数族的基本形： $p(x) = \frac{1}{Z(\theta)} \exp(\theta^\top \phi(x))$ , 此处

$$\phi(x) = \begin{pmatrix} x \\ x^2 \end{pmatrix}, \quad \theta = \begin{pmatrix} \mu/\sigma^2 \\ -1/(2\sigma^2) \end{pmatrix}.$$

于是  $\theta^\top \phi(x) = \frac{\mu}{\sigma^2}x - \frac{1}{2\sigma^2}x^2$ 。

为了把常数项也吸收进来，我们让“配分函数”  $Z(\theta)$  去承担所有与  $\theta$  有关但不含  $x$  的常数，使得整体仍旧归一化为 1。

### 4. 配分函数 $Z$ 与对数配分函数 $A = \log Z$

- 由式 (9.21) 可看出

$$\frac{1}{Z(\theta)} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\mu^2}{2\sigma^2}\right) \Rightarrow Z(\mu, \sigma^2) = \sqrt{2\pi\sigma^2} \exp\left(-\frac{\mu^2}{2\sigma^2}\right).$$

这就是上面“更正”的版本，与 (9.20) 严格一致。

- 若把  $Z$  改写成  $\theta$  的函数（消去  $\mu, \sigma^2$ ），使用

$$\theta_1 = \mu/\sigma^2, \quad \theta_2 = -1/(2\sigma^2) (< 0),$$

则

$$\sigma^2 = -\frac{1}{2\theta_2}, \quad \mu = -\frac{\theta_1}{2\theta_2}.$$

代回  $\log Z$  得到 (9.26)：

$$A(\theta) = \log Z(\theta) = -\frac{\theta_1^2}{4\theta_2} - \frac{1}{2} \log(-2\theta_2) - \frac{1}{2} \log(2\pi).$$

这里  $\theta_2 < 0$  是必须的（否则积分发散），这与  $\sigma^2 > 0$  等价。

### 5. 为什么充分统计量是 $(x, x^2)$ ？

- 因为密度只通过  $\theta_1 x + \theta_2 x^2$  依赖于数据。
- 这意味着对于  $n$  个样本，只要保存  $\sum x_i$  与  $\sum x_i^2$ （两维）就可以做极大似然或贝叶斯更新——这就是“有限维充分统计量”的具体体现。

### 6. 如何从自然参数找回 $\mu, \sigma^2$ （很关键的“来回通道”）

$$\sigma^2 = -\frac{1}{2\theta_2}, \quad \mu = -\frac{\theta_1}{2\theta_2}.$$

这在推导 GLM、VI (变分推断) 时常用到。

## 7. 一个小算例

取  $\mu = 2, \sigma^2 = 4$ 。

则  $\theta_1 = \mu/\sigma^2 = 0.5, \theta_2 = -1/(2\sigma^2) = -1/8 = -0.125$ 。

反推:  $\sigma^2 = -1/(2\theta_2) = -1/(-0.25) = 4; \mu = -\theta_1/(2\theta_2) = -0.5/(-0.25) = 2$ 。完全一致。

(理解:  $\theta_2$  必须是负数, 才能让  $x^2$  前的系数为负, 从而保证高斯的“钟形”可积。)

---

## B. 为什么“均匀分布”“学生 t 分布”不是指数族? (9.2.2.4)

### 1. 均匀分布 $\text{Unif}(a, b)$

- 密度:  $p(x) = \frac{1}{b-a} \mathbf{I}\{a \leq x \leq b\}$ 。
- **问题点:** 分布的**支持集** (允许取值的区间  $[a, b]$ ) 随参数  $a, b$  改变。
- **指数族 (常规/规则形式) 要求:** 支持集不随参数变化, 这样  $Z(\theta) = \int \exp(\theta^\top \phi(x)) dx$  才能作为单纯的归一化常数; 均匀分布违背了这一点。
- 直观理解: 当你改变  $a, b$  时, 允许的  $x$  区间也变了, 无法用“同一个  $\phi(x)$ 、同一个积分域”把所有参数情况统一表达成指数里的一组线性项。

### 2. 学生 t 分布 (Student t)

- 它是重尾分布, 标准形式不满足“有限维线性充分统计量 + 固定支持 + 指数内线性”的模板, 因此**不属于 (自然) 指数族**。
  - 旁注: 学生 t 可表示为“高斯的尺度混合”(层次模型), 但本身并非单层、有限维指数族。
- 

## 要点总结 (给你一眼抓住的记忆点)

- **高斯的一元情形是指数族:**
  - 充分统计量:  $\phi(x) = (x, x^2)$ 。
  - 自然参数:  $\theta_1 = \mu/\sigma^2, \theta_2 = -1/(2\sigma^2)$  (注意  $\theta_2 < 0$ )。
  - 对数配分:  $A(\theta) = -\frac{\theta_1^2}{4\theta_2} - \frac{1}{2} \log(-2\theta_2) - \frac{1}{2} \log(2\pi)$ 。
  - 来回通道:  $\sigma^2 = -1/(2\theta_2), \mu = -\theta_1/(2\theta_2)$ 。
  - 实务含义: 做估计/更新时只需  $\sum x_i, \sum x_i^2$ 。

- **非例子：**
  - 均匀分布不属于指数族，因为它的支持集随参数变。
  - 学生 t 分布不属于指数族，因为它无法写成所需的指数族标准形（有限维线性充分统计量等条件不满足）。
- **小提示：**很多常见分布都是指数族（伯努利、泊松、Gamma、高斯等），但不是全部；判断要看是否能写成“指数里线性”且支持固定。

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好，最后总结要点。

### 9.2.3 Log partition function

An important property of the exponential family is that derivatives of the log partition function can be used to generate cumulants of the sufficient statistics.<sup>2</sup> For this reason,  $A(\theta)$  is sometimes called a cumulant function. We will prove this for a 1-parameter distribution; this can be generalized to a K-parameter distribution in a straightforward way. For the first derivative we have

$$dA/d\theta = d/d\theta \log \exp(\theta\varphi(x))h(x)dx \quad (9.27)$$

$$= d/d\theta (\exp(\theta\varphi(x))h(x)dx) / \exp(\theta\varphi(x))h(x)dx \quad (9.28)$$

$$= \varphi(x)\exp(\theta\varphi(x))h(x)dx / \exp(A(\theta)) \quad (9.29)$$

$$= \varphi(x)\exp(\theta\varphi(x) - A(\theta))h(x)dx \quad (9.30)$$

$$= \varphi(x)p(x)dx = E[\varphi(x)] \quad (9.31)$$

For the second derivative we have

$$d^2A/d\theta^2 = \varphi(x)\exp(\theta\varphi(x) - A(\theta)) h(x)(\varphi(x) - A'(\theta))dx \quad (9.32)$$

$$= \varphi(x)p(x)(\varphi(x) - A'(\theta))dx \quad (9.33)$$

$$= \varphi^2(x)p(x)dx - A''(\theta) \varphi(x)p(x)dx \quad (9.34)$$

$$= E[\varphi^2(X)] - E[\varphi(X)]^2 = \text{var}[\varphi(X)] \quad (9.35)$$

where we used the fact that  $A''(\theta) = dA/d\theta = E[\varphi'(X)]$ .

In the multivariate case, we have that

$$\partial^2A/\partial\theta_i\partial\theta_j = E[\varphi_i(x)\varphi_j(x)] - E[\varphi_i(x)]E[\varphi_j(x)] \quad (9.36)$$

and hence

$$\nabla^2A(\theta) = \text{cov}[\varphi(X)] \quad (9.37)$$

Since the covariance is positive definite, we see that  $A(\theta)$  is a convex function (see Section 7.3.3).

#### 9.2.3.1 Example: the Bernoulli distribution

For example, consider the Bernoulli distribution. We have

$A(\theta) = \log(1 + e^\theta)$ , so the mean is given by

$$dA/d\theta = e^\theta / (1 + e^\theta) = 1/(1 + e^{-\theta}) = \text{sigm}(\theta) = \mu \quad (9.38)$$

The variance is given by

$$d^2A/d\theta^2 = d/d\theta (1/(1 + e^{-\theta})) = -(1/(1 + e^{-\theta}))^2 \cdot (-e^{-\theta}/(1 + e^{-\theta}))^2 = e^{-\theta} / (1 + e^{-\theta})^2 = e^{-\theta} / (1 + e^{-\theta}) \cdot 1/(1 + e^{-\theta}) = e^{-\theta} / (1 + e^{-\theta}) = (1 - \mu)\mu \quad (9.39)$$

$$= e^{-\theta} / (1 + e^{-\theta}) \cdot 1/(1 + e^{-\theta}) = e^{-\theta} / (1 + e^{-\theta}) = (1 - \mu)\mu \quad (9.40)$$

已思考 29s

下面先给出**逐字完整翻译**（不遗漏原式与编号），接着给出**基础且详尽的解释**（一步步推导、配直觉与小例子），最后给出**要点总结**。

## 逐字完整翻译

### 9.2.3 对数配分函数 (Log partition function)

指数族的一个重要性质是：对数配分函数的导数可以用来生成充分统计量的**累积量** (cumulants)。<sup>2</sup> 因此， $A(\theta)$  有时也称为**累积量函数** (cumulant function)。我们将对**单参数** (1-parameter) 分布证明这一点；将其推广到**多参数** (K-parameter) 分布是直接的。对一阶导数，我们有

$$\frac{dA}{d\theta} = \frac{d}{d\theta} \log \int \exp(\theta \phi(x)) h(x) dx \quad (9.27)$$

$$= \frac{\frac{d}{d\theta} \int \exp(\theta \phi(x)) h(x) dx}{\int \exp(\theta \phi(x)) h(x) dx} \quad (9.28)$$

$$= \frac{\int \phi(x) \exp(\theta \phi(x)) h(x) dx}{\exp(A(\theta))} \quad (9.29)$$

$$= \int \phi(x) \exp(\theta \phi(x) - A(\theta)) h(x) dx \quad (9.30)$$

$$= \int \phi(x) p(x) dx = E[\phi(x)] \quad (9.31)$$

对二阶导数，我们有

$$\frac{d^2A}{d\theta^2} = \int \phi(x) \exp(\theta \phi(x) - A(\theta)) h(x) (\phi(x) - A'(\theta)) dx \quad (9.32)$$

$$= \int \phi(x) p(x) (\phi(x) - A'(\theta)) dx \quad (9.33)$$

$$= \int \phi^2(x) p(x) dx - A'(\theta) \int \phi(x) p(x) dx \quad (9.34)$$

$$= \text{E}[\phi^2(x)] - (\text{E}[\phi(x)])^2 = \text{var}[\phi(x)] \quad (9.35)$$

其中我们用到了  $A'(\theta) = \frac{dA}{d\theta} = \text{E}[\phi(x)]$ 。在多变量情形下，有

$$\frac{\partial^2 A}{\partial \theta_i \partial \theta_j} = \text{E}[\phi_i(x)\phi_j(x)] - \text{E}[\phi_i(x)] \text{E}[\phi_j(x)] \quad (9.36)$$

因此

$$\nabla^2 A(\theta) = \text{cov}[\phi(x)] \quad (9.37)$$

由于协方差是半正定（正定），我们可知  $A(\theta)$  是**凸函数**（见第 7.3.3 节）。

### 9.2.3.1 例子：伯努利分布

例如，考虑伯努利分布。我们有  $A(\theta) = \log(1 + e^\theta)$ ，因此其均值为

$$\frac{dA}{d\theta} = \frac{e^\theta}{1 + e^\theta} = \frac{1}{1 + e^{-\theta}} = \text{sigm}(\theta) = \mu \quad (9.38)$$

方差为

$$\frac{d^2 A}{d\theta^2} = \frac{d}{d\theta} (1 + e^{-\theta})^{-1} = (1 + e^{-\theta})^{-2} \cdot e^{-\theta} \quad (9.39)$$

$$= \frac{e^{-\theta}}{1 + e^{-\theta}} \cdot \frac{1}{1 + e^{-\theta}} = \frac{1}{e^\theta + 1} \cdot \frac{1}{1 + e^{-\theta}} = (1 - \mu)\mu \quad (9.40)$$

## 通俗解释（一步步来）

### 1) 配分函数与“对数配分函数”是什么？

- 指数族的一般形：

$$p_\theta(x) = \frac{1}{Z(\theta)} h(x) \exp(\theta^\top \phi(x)), \quad A(\theta) = \log Z(\theta).$$

- $Z(\theta)$  负责“把概率归一化到 1”；取对数得到  $A(\theta)$ ，便于微分与推导。
- 这段要说明： $A(\theta)$  的**导数=充分统计量  $\phi(x)$  的矩与累积量**。
  - 一阶导数给出  $\text{E}_\theta[\phi(x)]$ （均值/第一累积量）。
  - 二阶导数给出  $\text{var}_\theta[\phi(x)]$ （方差=第二累积量）。
  - 多维情况下，Hessian ( $\nabla^2 A$ ) 就是  $\phi(x)$  的**协方差矩阵**。

### 2) 为什么一阶导数等于期望？

- 先写  $A(\theta) = \log \int h(x) \exp(\theta^\top \phi(x)) dx$ 。

- “对数的导数 = 本体的导数 / 本体”。把导数传进积分号（满足常见正则条件即可）：

$$A'(\theta) = \frac{\int \phi(x) h(x) \exp(\theta \phi(x)) dx}{\int h(x) \exp(\theta \phi(x)) dx}.$$

- 分子分母一看，就是对  $p_\theta(x)$  的期望：

$$A'(\theta) = \int \phi(x) p_\theta(x) dx = E_\theta[\phi(x)].$$

- 直觉：改变  $\theta$  会改变“指数里偏好的程度”，其“斜率”正是对  $\phi(x)$  的平均偏好。

### 3) 为什么二阶导数是方差？

- 再对  $A'(\theta) = E[\phi(x)]$  求导。由于  $p_\theta(x)$  对  $\theta$  的敏感度正比于  $\phi(x) - E[\phi]$ ，展开后得到

$$A''(\theta) = E[(\phi(x) - E[\phi]) \phi(x)] = E[\phi^2] - (E[\phi])^2 = \text{var}[\phi(x)].$$

- 直觉：曲率（Hessian）描述“均值随参数变化的敏感度”，而不确定性越大（方差越大），曲率越大。

### 4) 多维参数时，梯度与 Hessian 是什么？

- 向量情形下，

$$\nabla A(\theta) = E_\theta[\phi(x)], \quad \nabla^2 A(\theta) = \text{cov}_\theta[\phi(x)].$$

- 因为协方差矩阵半正定（PSD）（在极小指数族且非退化时为正定 PD），所以  $A(\theta)$  是凸函数。

这对优化很关键：最大似然通常要最小化  $A(\theta)$  加上线性项等，凸性保证好解、好算法。

### 5) 例子：伯努利（把抽象变成具体）

- 伯努利的对数配分： $A(\theta) = \log(1 + e^\theta)$ 。
- 一阶导数： $\sigma(\theta) = \frac{1}{1 + e^{-\theta}} = \mu$ （就是均值）。
- 二阶导数： $\mu(1 - \mu)$ （0-1 变量的方差）。
- 小算例： $\theta = 0 \Rightarrow \mu = 0.5, \text{ var} = 0.25; \theta \rightarrow +\infty \Rightarrow \mu \rightarrow 1, \text{ var} \rightarrow 0$ 。

小提示（与前面章节呼应）：

- 伯努利的充分统计量是  $\phi(x) = x$ 。
- 因此  $A'(\theta) = E[x] = \mu, A''(\theta) = \text{var}(x)$ 。

### 6) 一眼直觉图（文字版）

- $A(\theta)$ ：归一化常数的对数。

- $\nabla A$ : 充分统计量的均值。
  - $\nabla^2 A$ : 充分统计量的协方差 (曲率 = 不确定性)。
  - 结论:  $A$  凸  $\Rightarrow$  许多学习/推断问题是**凸优化**或至少局部良性。
- 

## 要点总结

- **一阶导:**  $\frac{dA}{d\theta} = E_\theta[\phi(x)]$ 。
- **二阶导:**  $\frac{d^2 A}{d\theta^2} = \text{var}_\theta[\phi(x)]$ 。
- **多维:**  $\nabla A(\theta) = E[\phi(x)]$ ,  $\nabla^2 A(\theta) = \text{cov}[\phi(x)] \Rightarrow A$  **凸**。
- **直觉:** 对数配分的斜率=平均充分统计量, 曲率=不确定性 (方差/协方差)。
- **伯努利示例:**  $A = \log(1 + e^\theta)$ , 均值  $\mu = \sigma(\theta)$ , 方差  $\mu(1 - \mu)$ 。
- **实用意义:** 这些性质使得指数族在**最大似然、贝叶斯、变分推断与 GLM**中推导简洁、优化稳定。

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好，最后总结要点。

### 9.2.4 MLE for the exponential family

The likelihood of an exponential family model has the form

$$p(D|\theta) = N \prod_{i=1}^N h(x_i) g(\theta) \exp \eta(\theta) \sum_{k=1}^K \varphi_k(x_i) \quad (9.41)$$

We see that the sufficient statistics are  $N$  and

$$\varphi(D) = [\sum_{i=1}^N \varphi_1(x_i), \dots, \sum_{i=1}^N \varphi_K(x_i)] \quad (9.42)$$

For example, for the Bernoulli model we have  $\varphi = [I(x_i = 1)]$ , and for the univariate Gaussian, we have  $\varphi = [x_1, x_2, \dots, x_N]$ . (We also need to know the sample size,  $N$ .) The Pitman-Koopman-Darmois theorem states that, under certain regularity conditions, the exponential family is the only family of distributions with finite sufficient statistics. (Here, finite means of a size independent of the size of the data set.) One of the conditions required in this theorem is that the support of the distribution not be dependent on the parameter. For a simple example of such a distribution, consider the uniform distribution

$$p(x|\theta) = U(x|\theta) = \theta I(0 \leq x \leq \theta) \quad (9.43)$$

The likelihood is given by

$$p(D|\theta) = \theta^N \exp(-\sum_{i=1}^N \phi(x_i)) \quad (9.44)$$

So the sufficient statistics are  $N$  and  $s(D) = \sum_{i=1}^N x_i$ . This is finite in size, but the uniform distribution is not in the exponential family because its support set,  $X$ , depends on the parameters. We now describe how to compute the MLE for a canonical exponential family model. Given  $N$  iid data points  $D = (x_1, \dots, x_N)$ , the log-likelihood is

$$\log p(D|\theta) = \theta^T \varphi(D) - N A(\theta) \quad (9.45)$$

Since  $-A(\theta)$  is concave in  $\theta$ , and  $\theta^T \varphi(D)$  is linear in  $\theta$ , we see that the log likelihood is concave, and hence has a unique global maximum. To derive this maximum, we use the fact that the derivative of the log partition function yields the expected value of the sufficient statistic vector (Section 9.2.3):

$$\nabla_\theta \log p(D|\theta) = \varphi(D) - N E[\varphi(X)] \quad (9.46)$$

Setting this gradient to zero, we see that at the MLE, the empirical average of the sufficient statistics must equal the model's theoretical expected sufficient statistics, i.e.,  $\hat{\theta}$  must satisfy

$$E[\varphi(X)] = \frac{1}{N} \sum_{i=1}^N \varphi(x_i) \quad (9.47)$$

This is called moment matching. For example, in the Bernoulli distribution, we have  $\varphi(X) = X$ , so the MLE satisfies

$$E[\varphi(X)] = p(X=1) = \hat{\mu} = \frac{1}{N} \sum_{i=1}^N I(x_i = 1) \quad (9.48)$$

已思考 22s

下面先给出**完整、逐字的中文翻译**（保留原式编号，不遗漏），随后给出**一步步、基础且详尽的解释**，最后以**要点清单收尾**，便于记忆。

## 逐字完整翻译

### 9.2.4 指数族的极大似然估计 (MLE for the exponential family)

指数族模型的似然具有如下形式

$$p(D | \theta) = \prod_{i=1}^N h(x_i) g(\theta)^N \exp(\eta(\theta)^T [\sum_{i=1}^N \phi(x_i)]) \quad (9.41)$$

我们看到，充分统计量是  $N$  以及

$$\phi(D) = \left[ \sum_{i=1}^N \phi_1(x_i), \dots, \sum_{i=1}^N \phi_K(x_i) \right] \quad (9.42)$$

例如，对伯努利模型，我们有  $\phi = [\sum_i \mathbf{I}(x_i = 1)]$ ；对一元高斯，我们有  $\phi = [\sum_i x_i, \sum_i x_i^2]$ 。（我们还需要知道样本量  $N$ 。）Pitman–Koopman–Darmois 定理指出，在某些正则性条件下，指数族是唯一一个具有有限（维）充分统计量的分布族。（这里，“有限”是指其大小与数据集的规模无关。）该定理所需的条件之一是：分布的支持集不依赖于参数。一个简单的“反例”分布是均匀分布：

$$p(x | \theta) = U(x | \theta) = \frac{1}{\theta} \mathbf{I}(0 \leq x \leq \theta) \quad (9.43)$$

其似然为

$$p(D | \theta) = \theta^{-N} \mathbf{I}(0 \leq \max\{x_i\} \leq \theta) \quad (9.44)$$

因此充分统计量是  $N$  和  $s(D) = \max_i x_i$ 。这（个向量）大小是有限的，但**均匀分布并不属于指数族**，因为其支持集  $X$  依赖于参数。

现在我们来描述如何为**典范 (canonical) 指数族**模型计算 MLE。给定  $N$  个独立同分布的数据点  $D = (x_1, \dots, x_N)$ ，对数似然为

$$\log p(D | \theta) = \theta^\top \phi(D) - N A(\theta) \quad (9.45)$$

由于  $-A(\theta)$  关于  $\theta$  是凹的，并且  $\theta^\top \phi(D)$  关于  $\theta$  是线性的，所以对数似然是凹的，因此具有唯一的全局极大值。为了求出该极大值，我们利用“对数配分函数的导数给出充分统计量向量的期望”（见 9.2.3 节）这一事实：

$$\nabla_\theta \log p(D | \theta) = \phi(D) - N E[\phi(X)] \quad (9.46)$$

把该梯度设为 0，我们看到在 MLE 处，**充分统计量的经验平均必须等于模型的理论期望**，也就是说， $\hat{\theta}$  满足

$$E[\phi(X)] = \frac{1}{N} \sum_{i=1}^N \phi(x_i) \quad (9.47)$$

这称为**矩匹配 (moment matching)**。例如，在伯努利分布中，我们有  $\phi(X) = \mathbf{I}(X = 1)$ ，所以 MLE 满足

$$E[\phi(X)] = \Pr(X = 1) = \hat{\mu} = \frac{1}{N} \sum_{i=1}^N \mathbf{I}(x_i = 1) \quad (9.48)$$

## 通俗解释（一步步来，尽量不跳步）

## 1) 式 (9.41): 为什么似然能写成这个样子?

- 单个样本的指数族一般形:

$$p(x | \theta) = h(x) g(\theta) \exp(\eta(\theta)^\top \phi(x)).$$

这里  $h(x)$  只与数据有关,  $\phi(x)$  是充分统计量,  $\eta(\theta)$  是(可能非线性的) **典范参数**, 而  $g(\theta) = \exp(-A(\theta))$  是把概率归一化所需的常数项。

- $N$  个独立样本相乘:

$$\prod_{i=1}^N h(x_i) g(\theta)^N \exp(\eta(\theta)^\top \sum_{i=1}^N \phi(x_i)),$$

这正是 (9.41)。

结论: **数据只通过两种汇总进入似然**: 样本量  $N$  与充分统计量之和  $\sum \phi(x_i)$ 。

## 2) 式 (9.42): 充分统计量的“有限维压缩”

- $\phi(D) = [\sum_i \phi_1(x_i), \dots, \sum_i \phi_K(x_i)]$

表明无论  $N$  多大, \*\*只需存  $K$  个数 (外加  $N$ ) \*\*即可做后续推断。

- 例子:
  - 伯努利: 只需记住“1 的个数”  $\sum_i \mathbf{I}(x_i = 1)$ ;
  - 一元高斯: 记住  $\sum x_i$  与  $\sum x_i^2$  (再配合  $N$ )。

## 3) PKD 定理在说什么? 为什么均匀分布是“反例”?

- Pitman–Koopman–Darmois 定理: 在支持集不依赖参数等正则条件下, **只有指数族** 拥有“维度固定”的充分统计量。
- 均匀分布  $\text{Unif}(0, \theta)$  的密度是  $\frac{1}{\theta} \mathbf{I}(0 \leq x \leq \theta)$ .
  - 它的似然:  $\theta^{-N} \mathbf{I}(0 \leq \max x_i \leq \theta)$ .
  - 虽然充分统计量可以只用两个数  $(N, \max x_i)$  表示, 但**支持集**  $[0, \theta]$  随参数变化, 违背了定理的关键条件, 因此它**不是**指数族。
- 直觉: 指数族需要在一个**固定的**取值域上, 通过指数里的线性项来表达参数影响; 均匀分布是“挪动边界”来表达参数, 不合规。

## 4) 式 (9.45): 对数似然的简洁形态与凸性

- 在**典范形式** ( $\eta(\theta) = \theta$ ) 下,

$$\log p(D | \theta) = \theta^\top \phi(D) - N A(\theta).$$

- 已知  $A(\theta)$  是**凸函数** (见 9.2.3: Hessian 是协方差矩阵, 半正定), 所以  $-A(\theta)$  是**凹**的; 线性项  $\theta^\top \phi(D)$  也是**凹** (也是**凸**), 故整体**凹**。

这意味着**只有一个全局极大值** (没有局部极大坑), 优化更省心。

## 5) 式(9.46)–(9.47): 矩匹配给出 MLE 的封闭条件

- 梯度:

$$\nabla_{\theta} \log p(D | \theta) = \phi(D) - N \nabla_{\theta} A(\theta).$$

- 由 9.2.3,  $\nabla_{\theta} A(\theta) = E_{\theta}[\phi(X)]$ .

令梯度为 0, 得到:

$$E_{\theta}[\phi(X)] = \frac{1}{N} \sum_{i=1}^N \phi(x_i).$$

- 这就是**矩匹配** (moment matching) : 让模型下“充分统计量的期望”=“数据的经验均值”。

- 伯努利:  $\phi(X) = \mathbf{I}(X = 1)$ , 故  $\hat{\mu} = \frac{1}{N} \sum_i \mathbf{I}(x_i = 1)$  (样本中“1”的比例)。

- 一元高斯 (均值与方差未知) :

$$E[X] = \frac{1}{N} \sum x_i \Rightarrow \hat{\mu} = \bar{x};$$

$$E[X^2] = \frac{1}{N} \sum x_i^2 \Rightarrow \hat{\sigma}^2 = \frac{1}{N} \sum (x_i - \bar{x})^2 \text{ (注意是 MLE 版本, 分母 } N \text{ 而非 } N - 1 \text{).}$$

小直觉: 指数族里, 把“充分统计量的样本均值”交给模型去匹配, 就找到了参数的极大似然解。

## 要点总结

- 似然结构:

$$p(D | \theta) = \prod h(x_i) g(\theta)^N \exp(\eta(\theta)^T \sum \phi(x_i)),$$

数据只通过  $N$  与  $\sum \phi(x_i)$  进入 (有限维压缩)。

- PKD 定理: 在“支持不随参数变”等条件下, **只有指数族** 拥有固定维的充分统计量; 均匀分布因支持依赖参数而非指数族。

- 对数似然:  $\log p = \theta^T \phi(D) - NA(\theta)$  (典范形式); 因  $A$  凸  $\Rightarrow$  对数似然凹  $\Rightarrow$  唯一全局 MLE。

- 梯度与矩匹配:

$$\nabla_{\theta} \log p = \phi(D) - N E[\phi(X)].$$

令 0  $\Rightarrow E[\phi(X)] = \frac{1}{N} \sum \phi(x_i)$  (**矩匹配**)。

- 实例: 伯努利的 MLE 就是样本中“1”的比例; 高斯的 MLE 给出样本均值与带分母  $N$  的方差。

一句话记忆: **指数族的 MLE = 让“模型的充分统计量期望”精确贴合“数据的充分统计量样本均值”。**

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好，最后总结要点。

### 9.2.5 Bayes for the exponential family

We have seen that exact Bayesian analysis is considerably simplified if the prior is conjugate to the likelihood.

Informally this means that the prior  $p(\theta|\tau)$  has the same form as the likelihood  $p(D|\theta)$ . For this to make sense, we require that the likelihood have finite sufficient statistics, so that we can write  $p(D|\theta) = p(s(D)|\theta)$ . This suggests that the only family of distributions for which conjugate priors exist is the exponential family. We will derive the form of the prior and posterior below.

#### 9.2.5.1 Likelihood

The likelihood of the exponential family is given by

$$p(D|\theta) \propto g(\theta)^N \exp \eta(\theta)^T s(N) \quad (9.49)$$

where  $s(N) = N \sum_{i=1}^N s(x_i)$ . In terms of the canonical parameters this becomes

$$p(D|\eta) \propto \exp(N\eta^T s - N A(\eta)) \quad (9.50)$$

where  $s = 1/N s(N)$ .

#### 9.2.5.2 Prior

The natural conjugate prior has the form

$$p(\theta|v_0, \tau_0) \propto g(\theta)^{v_0} \exp \eta(\theta)^T \tau_0 \quad (9.51)$$

Let us write  $\tau_0 = v_0 \tau_0$ , to separate out the size of the prior pseudo-data,  $v_0$ , from the mean of the sufficient statistics on this pseudo-data,  $\tau_0$ . In canonical form, the prior becomes

$$p(\eta|v_0, \tau_0) \propto \exp(v_0 \eta^T \tau_0 - v_0 A(\eta)) \quad (9.52)$$

#### 9.2.5.3 Posterior

The posterior is given by

$$p(\theta|D) = p(\theta|vN, \tauN) = p(\theta|v_0 + N, \tau_0 + s(N)) \quad (9.53)$$

So we see that we just update the hyper-parameters by adding. In canonical form, this becomes

$$p(\eta|D) \propto \exp \eta^T (v_0 \tau_0 + s(N)) - (v_0 + N) A(\eta) \quad (9.54)$$

$$= p(\eta|v_0 + N, v_0 \tau_0 + s(N) v_0 + N) \quad (9.55)$$

So we see that the posterior hyper-parameters are a convex combination of the prior mean hyper-parameters and the average of the sufficient statistics.

#### 9.2.5.4 Posterior predictive density

Let us derive a generic expression for the predictive density for future observables  $D = (\tilde{x}_1, \dots, \tilde{x}_N)$  given past data  $D = (x_1, \dots, x_N)$  as follows. For notational brevity, we will combine the sufficient statistics with the size of the data, as follows:  $\tau^0 = (v_0, \tau_0)$ ,  $\tilde{s}(D) = (N, s(D))$ , and  $\tilde{s}(D) = (N, s(D))$ . So the prior becomes

$$p(\theta|\tilde{\tau}^0) = 1 Z(\tilde{\tau}^0) g(\theta)v_0 \exp(\eta(\theta)\tau\tau^0) \quad (9.56)$$

The likelihood and posterior have a similar form. Hence

$$p(D|D) = p(D|\theta)p(\theta|D)d\theta \quad (9.57)$$

$$= \int [ \prod_{i=1}^N h(\tilde{x}_i) ] \prod_{i=1}^N Z(\tilde{\tau}^0 + \tilde{s}(D))^{-1} g(\theta)v_0 + N + N d\theta \quad (9.58)$$

$$\times \exp \left( \int k \eta_k(\theta)(\tau_k + N) \sum_{i=1}^N s_k(x_i) + N \sum_{i=1}^N s_k(\tilde{x}_i) \right) d\theta \quad (9.59)$$

$$= \int [ \prod_{i=1}^N h(\tilde{x}_i) ] Z(\tilde{\tau}^0 + \tilde{s}(D) + \tilde{s}(D)) Z(\tilde{\tau}^0 + \tilde{s}(D)) \quad (9.60)$$

If  $N = 0$ , this becomes the marginal likelihood of  $D$ , which reduces to the familiar form of normalizer of the posterior divided by the normalizer of the prior, multiplied by a constant.

#### 9.2.5.5 Example: Bernoulli distribution

As a simple example, let us revisit the Beta-Bernoulli model in our new notation. The likelihood is given by

$$p(D|\theta) = (1 - \theta)^N \exp \log(\theta) \sum_{i=1}^N x_i \quad (9.61)$$

Hence the conjugate prior is given by

$$p(\theta|v_0, \tau_0) \propto (1 - \theta)^{v_0} \exp \log(\theta) \sum_{i=1}^N x_i \tau_0 \quad (9.62)$$

$$= \theta^{\tau_0} (1 - \theta)^{v_0 - \tau_0} \quad (9.63)$$

If we define  $\alpha = \tau_0 + 1$  and  $\beta = v_0 - \tau_0 + 1$ , we see that this is a beta distribution. We can derive the posterior as follows, where  $s = \sum_{i=1}^N I(x_i = 1)$  is the sufficient statistic:

$$p(\theta|D) \propto \theta^{\tau_0 + s} (1 - \theta)^{v_0 - \tau_0 + n - s} \quad (9.64)$$

$$= \theta^{\tau_0} (1 - \theta)^{v_0 - \tau_0} \quad (9.65)$$

We can derive the posterior predictive distribution as follows. Assume  $p(\theta) = \text{Beta}(\theta|\alpha, \beta)$ , and let  $s = s(D)$  be the number of heads in the past data. We can predict the probability of a given sequence of future heads,  $D = (\tilde{x}_1, \dots, \tilde{x}_m)$ , with sufficient statistic  $s = \sum_{i=1}^m I(\tilde{x}_i = 1)$ , as follows:

$$p(D|D) = \int p(D|\theta) p(\theta|Beta(\theta|\alpha, \beta)) d\theta \quad (9.66)$$

$$= \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)} \frac{\Gamma(\alpha)}{\Gamma(\alpha + m)} \frac{\Gamma(\beta)}{\Gamma(\beta + m)} \theta^{\alpha + m - 1} (1 - \theta)^{\beta + m - 1} d\theta \quad (9.67)$$

$$= \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)} \frac{\Gamma(\alpha)}{\Gamma(\alpha + m)} \frac{\Gamma(\beta)}{\Gamma(\beta + m)} \Gamma(\alpha + m + \beta + m) \quad (9.68)$$

where

$$\alpha + m = \alpha + s = \alpha + s + s \quad (9.69)$$

$$\beta + m = \beta + (m - s) = \beta + (n - s) + (m - s) \quad (9.70)$$

下面先给出**完整直译**（逐字、不遗漏，并保留原式编号）；随后给出**基础且详尽的解释**（一步步、配直觉与小例子）；最后给出**要点总结**。

（说明：原文中少数地方记号/积分上下限存在轻微排版不一致，我会在“解释”部分指出并给出等价的规范写法，便于理解。）

---

## 逐字完整翻译

### 9.2.5 指数族的贝叶斯法 (Bayes for the exponential family)

我们已经看到，如果先验与似然是**共轭的**，那么精确的贝叶斯分析会大大简化。非正式地说，这意味着先验  $p(\theta | \tau)$  与似然  $p(D | \theta)$  具有**相同的形式**。要使这有意义，我们要求似然具有**有限的充分统计量**，从而我们可以把  $p(D | \theta)$  写成  $p(s(D) | \theta)$ 。这表明，存在共轭先验的分布族**只有指数族**。我们将在下面推导先验与后验的形式。

#### 9.2.5.1 似然 (Likelihood)

指数族的似然为

$$p(D | \theta) \propto g(\theta)^N \exp(\eta(\theta)^\top s_N) \quad (9.49)$$

其中  $s_N = \sum_{i=1}^N s(x_i)$ 。用典范参数表示即为

$$p(D | \eta) \propto \exp(N \eta^\top s - N A(\eta)) \quad (9.50)$$

这里  $s = \frac{1}{N} s_N$ 。

#### 9.2.5.2 先验 (Prior)

自然共轭先验具有形式

$$p(\theta | v_0, \tau_0) \propto g(\theta)^{v_0} \exp(\eta(\theta)^\top \tau_0) \quad (9.51)$$

让我们把  $\tau_0$  写成  $\tau_0 = v_0 \tau_0$ ，以便把**先验伪数据** (pseudo-data) 的“规模”  $v_0$  与该伪数据上**充分统计量均值**  $\tau_0$  区分开来。在典范形式下，先验变为

$$p(\eta | v_0, \tau_0) \propto \exp(v_0 \eta^\top \tau_0 - v_0 A(\eta)) \quad (9.52)$$

#### 9.2.5.3 后验 (Posterior)

后验为

$$p(\theta | D) = p(\theta | v_N, \tau_N) = p(\theta | v_0 + N, \tau_0 + s_N) \quad (9.53)$$

因此我们看到，只需通过“相加”来更新超参数。在典范形式下，这变为

$$p(\eta | D) \propto \exp(\eta^\top (v_0 \tau_0 + N s) - (v_0 + N) A(\eta)) \quad (9.54)$$

广义线性模型解释

$$= p(\eta \mid v_0 + N, \frac{v_0 \tau_0 + N s}{v_0 + N}) \quad (9.55)$$

因此我们看到，后验超参数是“先验均值超参数”和“充分统计量样本均值”的**凸组合**。

#### 9.2.5.4 后验预测密度 (Posterior predictive density)

让我们为给定过去数据  $D = (x_1, \dots, x_N)$  的未来观测  $D' = (\tilde{x}_1, \dots, \tilde{x}_{N'})$  推导一个通用的预测密度表达式。为简化记号，我们把充分统计量与数据规模合并如下： $\tilde{\tau}_0 = (v_0, \tau_0)$ ,  $\tilde{s}(D) = (N, s(D))$ ,  $\tilde{s}(D') = (N', s(D'))$ 。于是先验为

$$p(\theta \mid \tilde{\tau}_0) = \frac{1}{Z(\tilde{\tau}_0)} g(\theta)^{v_0} \exp(\eta(\theta)^\top \tau_0) \quad (9.56)$$

似然与后验有相似形式。因此

$$p(D' \mid D) = \int p(D' \mid \theta) p(\theta \mid D) d\theta \quad (9.57)$$

$$= \left[ \prod_{i=1}^{N'} h(\tilde{x}_i) \right] Z(\tilde{\tau}_0 + \tilde{s}(D))^{-1} \int g(\theta)^{v_0 + N + N'} d\theta \quad (9.58)$$

$$\times \exp \left( \sum_k \eta_k(\theta) \left( \tau_k + \sum_{i=1}^N s_k(x_i) + \sum_{i=1}^{N'} s_k(\tilde{x}_i) \right) \right) d\theta \quad (9.59)$$

$$= \left[ \prod_{i=1}^{N'} h(\tilde{x}_i) \right] \frac{Z(\tilde{\tau}_0 + \tilde{s}(D) + \tilde{s}(D'))}{Z(\tilde{\tau}_0 + \tilde{s}(D))} \quad (9.60)$$

若  $N' = 0$ ，这就成为  $D'$  的边际似然，它化成一个熟悉的形式：等于**后验的归一化常数除以先验的归一化常数，再乘以一个常数**。

#### 9.2.5.5 例子：伯努利分布 (Example: Bernoulli distribution)

作为一个简单例子，让我们用新的记号重述 Beta–Bernoulli 模型。似然为

$$p(D \mid \theta) = (1 - \theta)^N \exp\left(\log \frac{\theta}{1 - \theta} \sum_i x_i\right) \quad (9.61)$$

因此共轭先验为

$$p(\theta \mid v_0, \tau_0) \propto (1 - \theta)^{v_0} \exp\left(\log \frac{\theta}{1 - \theta} \tau_0\right) \quad (9.62)$$

$$= \theta^{\tau_0} (1 - \theta)^{v_0 - \tau_0} \quad (9.63)$$

若定义  $\alpha = \tau_0 + 1$  且  $\beta = \nu_0 - \tau_0 + 1$ , 我们可以看到这是一个 Beta 分布。下面推导后验, 其中  $s = \sum_i \mathbf{I}(x_i = 1)$  是充分统计量:

$$p(\theta | D) \propto \theta^{\tau_0+s} (1-\theta)^{\nu_0-\tau_0+n-s} \quad (9.64)$$

$$= \theta^{\tau_n} (1-\theta)^{\nu_n-\tau_n} \quad (9.65)$$

我们可以如下推导后验预测分布。设  $p(\theta) = \text{Beta}(\theta | \alpha, \beta)$ , 令  $s = s(D)$  是过去数据中的正面 (1) 的次数。我们要预测一段未来正面序列  $D' = (\tilde{x}_1, \dots, \tilde{x}_m)$  的概率, 其充分统计量  $\tilde{s} = \sum_{i=1}^m \mathbf{I}(\tilde{x}_i = 1)$ :

$$p(D' | D) = \int_1^0 p(D' | \theta) \text{Beta}(\theta | \alpha_n, \beta_n) d\theta \quad (9.66)$$

$$= \frac{\Gamma(\alpha_n + \beta_n)}{\Gamma(\alpha_n)\Gamma(\beta_n)} \int_1^0 \theta^{\alpha_n+\tilde{s}-1} (1-\theta)^{\beta_n+m-\tilde{s}-1} d\theta \quad (9.67)$$

$$= \frac{\Gamma(\alpha_n + \beta_n)}{\Gamma(\alpha_n)\Gamma(\beta_n)} \frac{\Gamma(\alpha_{n+m})\Gamma(\beta_{n+m})}{\Gamma(\alpha_{n+m} + \beta_{n+m})} \quad (9.68)$$

其中

$$\alpha_{n+m} = \alpha_n + \tilde{s} = \alpha + s + \tilde{s} \quad (9.69)$$

$$\beta_{n+m} = \beta_n + (m - \tilde{s}) = \beta + (n - s) + (m - \tilde{s}) \quad (9.70)$$

## 通俗解释 (逐步、不跳步)

记号小提示: 本节把“充分统计量”记作  $s(\cdot)$  (此前也常用  $\phi(\cdot)$ ) , 把“典范参数”记作  $\eta(\theta)$ , 把  $\exp(-A(\eta))$  写作  $g(\theta)$ 。

### 1) 为什么指数族“天然有”共轭先验?

- 单样本:  $p(x | \theta) = h(x) g(\theta) \exp(\eta(\theta)^\top s(x))$ 。
- $N$  个独立样本相乘后 (式 9.49–9.50) :

$$p(D | \eta) \propto \exp(N \eta^\top s - N A(\eta)),$$

其中  $s = \frac{1}{N} \sum_i s(x_i)$  是**样本均值型**汇总。

- 若令先验也长得像“ $\exp(\text{线性} - \text{规模} \times A)$ ” (式 9.52) :

$$p(\eta | \nu_0, \tau_0) \propto \exp(\nu_0 \eta^\top \tau_0 - \nu_0 A(\eta)),$$

把它与似然相乘, 指数里的项**直接相加**, 仍在同一家族里  $\Rightarrow$  **共轭**。

## 伪数据直觉 ( $v_0, \tau_0$ 的含义)

- $v_0$ : 先验里“相当于”纳入了  $v_0$  个“伪样本”的样本量;
- $\tau_0$ : 这些伪样本的充分统计量的平均值 (如“伪样本的平均成功率”)。
- 因此, 把先验与数据合并后, 更新就是把伪数据和真数据的充分统计量做加法 (式 9.53–9.55) :

$$\underbrace{v_0 \tau_0}_{\text{先验的“总充分统计量”}} + \underbrace{Ns}_{\text{数据的“总充分统计量”}} / (v_0 + N).$$

这给出了后验中的“新均值超参数”, 是二者的加权平均 (凸组合)。

## 2) 后验预测: 为什么会是“归一化常数之比”?

- 由于先验、似然、后验都在同一家族, 积分  $\int p(D' | \theta) p(\theta | D) d\theta$  的结果可以用归一化常数  $Z(\cdot)$  表示 (式 9.60) :

$$p(D' | D) = [\prod h(\tilde{x}_i)] \frac{Z(\tilde{\tau}_0 + \tilde{s}(D) + \tilde{s}(D'))}{Z(\tilde{\tau}_0 + \tilde{s}(D))}.$$

- 直觉: 把“新老数据”的充分统计量合并进超参数, 前后两个“Z”之比就衡量了加入  $D'$  前后, 归一化常数的变化, 从而给出预测概率。

版式纠正: 式 (9.66)–(9.67) 的积分写成  $\int_0^1$  更规范 (原文行文中上下限倒排, 应视为排版小误)。结论不受影响。

## 3) 伯努利–Beta 的具体化 (把抽象落到地)

- 伯努利似然:  $\mu^s(1 - \mu)^{N-s}$ ; 等价写成 (9.61) 那样的  $(1 - \mu)^N \times \exp(\text{logit} \times s)$  只是方便对齐指数族模板。
- 取先验  $\mu \sim \text{Beta}(\alpha, \beta)$ , 其密度  $\propto \mu^{\alpha-1}(1 - \mu)^{\beta-1}$ 。
- 后验:  $\mu | D \sim \text{Beta}(\alpha + s, \beta + N - s)$  (式 9.64–9.65 本质上就是这个)。
- 后验预测 (未来  $m$  次试验正面数  $\tilde{s}$ ) :

$$p(\tilde{s} | D) = \binom{m}{\tilde{s}} \frac{B(\alpha_n + \tilde{s}, \beta_n + m - \tilde{s})}{B(\alpha_n, \beta_n)},$$

这就是Beta–二项分布 (式 9.68 用伽马函数  $\Gamma$  写的就是它)。

- 单次预测 ( $m = 1$ ) 时:  $\Pr(\tilde{x} = 1 | D) = \frac{\alpha_n}{\alpha_n + \beta_n}$ , 即后验均值。

## 4) 与 9.2.3 的联系: 为什么这么“好算”?

- 因为  $\nabla A(\eta) = E[s(X)]$ ,  $\nabla^2 A(\eta) = \text{cov}[s(X)]$ 。
- 这使得:

- MLE 是矩匹配 (9.2.4) ;
  - 贝叶斯里, 共轭先验/后验、预测都能“收口”为  $Z$  的简单组合;
  - 优化里,  $-A$  的凹性带来唯一极值与数值稳定。
- 

## 要点总结

- 指数族的自然共轭先验:

$$p(\eta \mid v_0, \tau_0) \propto \exp(v_0 \eta^\top \tau_0 - v_0 A(\eta)).$$

$v_0$  = 先验“伪样本量”,  $\tau_0$  = 先验中充分统计量的“均值”。

- 后验更新 = 加法:

$$v_N = v_0 + N, \quad \tau_N = \tau_0 + s_N;$$

后验中的“均值超参数” =  $(v_0 \tau_0 + N s_N) / (v_0 + N)$  (凸组合)。

- 后验预测:

$$p(D' \mid D) = [\prod h(\tilde{x}_i)] \frac{Z(\tilde{\tau}_0 + \tilde{s}(D) + \tilde{s}(D'))}{Z(\tilde{\tau}_0 + \tilde{s}(D))} \text{ (归一化常数之比) }.$$

- Bernoulli–Beta 例子:

先验 Beta( $\alpha, \beta$ ), 后验 Beta( $\alpha + s, \beta + N - s$ );

预测  $\tilde{s}$  服从 Beta–二项; 单次预测概率 = 后验均值。

- 大图景: 指数族之所以在贝叶斯里“好用”, 是因为它让先验/似然/后验都保持一族形式, 计算收口为简单的“参数相加”和“ $Z$  之比”。这也是实际建模常优先选指数族的核心原因之一。

翻译并解释以下内容, 翻译不能有遗漏, 解释请基础且详尽, 因为我的数学基础不太好, 最后总结要点。

### 9.2.6 Maximum entropy derivation of the exponential family

Although the exponential family is convenient, is there any deeper justification for its use? It turns out that there is: it is the distribution that makes the least number of assumptions about the data, subject to a specific set of user-specified constraints, as we explain below. In particular, suppose all we know is the expected values of certain features or functions:

$$\mathbf{x}^\top \mathbf{f}(\mathbf{x}) p(\mathbf{x}) = \mathbf{F}(\mathbf{k}) \quad (9.71)$$

where  $F_k$  are known constants, and  $f_k(x)$  is an arbitrary function. The principle of maximum entropy or maxent says we should pick the distribution with maximum entropy (closest to uniform), subject to the constraints that the moments of the distribution match the empirical moments of the specified functions. To maximize entropy subject to the constraints in Equation 9.71, and the constraints that  $p(x) \geq 0$  and  $\sum x_i p(x_i) = 1$ , we need to use Lagrange multipliers. The Lagrangian is given by

$$J(p, \lambda) = -\sum x_i p(x_i) \log p(x_i) + \lambda_0(1 - \sum x_i p(x_i)) + \sum k \lambda_k (F_k - \sum x_i p(x_i) f_k(x_i)) \quad (9.72)$$

We can use the calculus of variations to take derivatives wrt the function  $p$ , but we will adopt a simpler approach and treat  $p$  as a fixed length vector (since we are assuming  $x$  is discrete). Then we have

$$\frac{\partial J}{\partial p(x)} = -1 - \log p(x) - \lambda_0 - k \lambda_k f_k(x) \quad (9.73)$$

Setting  $\frac{\partial J}{\partial p(x)} = 0$  yields

$$p(x) = \frac{1}{Z} \exp(-k \lambda_k f_k(x)) \quad (9.74)$$

where  $Z = e^{-\lambda_0} + \sum k \lambda_k$ . Using the sum to one constraint, we have

$$1 = \sum x_i p(x) = \sum x_i \frac{1}{Z} \exp(-k \lambda_k f_k(x)) \quad (9.75)$$

Hence the normalization constant is given by

$$Z = \frac{1}{\sum x_i \exp(-k \lambda_k f_k(x))} \quad (9.76)$$

Thus the maxent distribution  $p(x)$  has the form of the exponential family (Section 9.2), also known as the Gibbs distribution.

已思考 20s

下面先给出逐字完整翻译（保留原式编号、无遗漏），再给出基础且详尽的解释（一步步推导 + 直觉 + 小例子），最后附上要点总结。

## 逐字完整翻译

### 9.2.6 指数族的最大熵推导 (Maximum entropy derivation of the exponential family)

尽管指数族很方便，但是是否有更深的理由来使用它？事实证明有的：正如下文所解释，它是在一组用户指定的约束条件下，对数据作出假设最少的分布。具体地说，假设我们唯一知道的是某些特征或函数的期望值：

$$\sum_x f_k(x) p(x) = F_k \quad (9.71)$$

其中  $F_k$  是已知常数,  $f_k(x)$  是任意函数。**最大熵原理** (maxent) 说: 在满足这些约束 (即分布的这些矩与指定函数的经验矩相匹配) 的前提下, 我们应当选择**熵最大** (最接近均匀分布) 的分布。为了在式 (9.71) 的约束下最大化熵, 同时还要满足  $p(x) \geq 0$  和  $\sum_x p(x) = 1$ , 我们需要使用拉格朗日乘子。拉格朗日函数为

$$J(p, \lambda) = -\sum_x p(x) \log p(x) + \lambda_0(1 - \sum_x p(x)) + \sum_k \lambda_k (F_k - \sum_x p(x) f_k(x)) \quad (9.72)$$

我们可以用变分法对函数  $p$  求导, 但我们采用更简单的方法, 把  $p$  看成一个**定长向量** (因为我们假设  $x$  是离散的)。于是有

$$\frac{\partial J}{\partial p(x)} = -1 - \log p(x) - \lambda_0 - \sum_k \lambda_k f_k(x) \quad (9.73)$$

令  $\frac{\partial J}{\partial p(x)} = 0$  得

$$p(x) = \frac{1}{Z} \exp(-\sum_k \lambda_k f_k(x)) \quad (9.74)$$

其中  $Z = e^{1+\lambda_0}$ 。利用“概率和为 1”的约束, 有

$$1 = \sum_x p(x) = \frac{1}{Z} \sum_x \exp(-\sum_k \lambda_k f_k(x)) \quad (9.75)$$

因此归一化常数为

$$Z = \sum_x \exp(-\sum_k \lambda_k f_k(x)). \quad (9.76)$$

由此可见, 最大熵分布  $p(x)$  具有**指数族** (第 9.2 节) 的形式, 也称为**吉布斯分布** (Gibbs distribution)。

## 通俗解释 (一步步来)

### 1) “最大熵”在说什么?

- 你只相信**很少的信息**: 某些函数的期望值 (式 9.71)。除此之外不想额外假设。
- 熵  $H(p) = -\sum_x p(x) \log p(x)$  度量“不确定度/均匀程度”。熵越大, 分布越“平”, 越少做额外形状假设。
- **最大熵原理**: 在满足已知约束 (期望值等) 的前提下, 让熵最大  $\rightarrow$  得到“最不偏”的分布。

## 2) 怎么解这个带约束的最大化问题?

- 引入拉格朗日乘子  $\lambda_0, \lambda_1, \dots$  把约束并入目标函数 (式 9.72)。
- 对每个  $x$  的概率分量  $p(x)$  求偏导 (式 9.73)，设为 0 得一阶最优条件。
- 立刻得到 (式 9.74)：

$$p(x) \propto \exp\left(-\sum_k \lambda_k f_k(x)\right).$$

其中  $\lambda_k$  是乘子，待通过配条件 (让期望等于  $F_k$ ) 来确定。

- 用 " $\sum_x p(x) = 1$ " 解出归一化常数  $Z$  (式 9.75–9.76)。

小直觉： $\lambda_k$  像“价格/权重”，调整到恰好让

$$\sum_x f_k(x)p(x) = F_k.$$

如果某个  $f_k(x)$  在某些  $x$  上更大，而  $\lambda_k > 0$ ，指数里就更“惩罚”这些  $x$ ，以降低该函数的期望，直到匹配  $F_k$ 。

## 3) 为什么这是“指数族”？

把 (9.74) 与指数族模板对比：

$$p(x) = \frac{1}{Z} h(x) \exp(\theta^\top \phi(x)) \quad (\text{此处 } h(x) \equiv 1).$$

只需令

$$\phi_k(x) = f_k(x), \quad \theta_k = -\lambda_k, \quad A(\theta) = \log Z(\theta),$$

就完全吻合。也就是说：

“在给定某些期望约束的最大熵解” = “指数族分布”（也称吉布斯/玻尔兹曼分布）。

## 4) 离散/连续的区别

- 文中把  $x$  当作离散 (因此用  $\sum_x$ )。
- 若  $x$  连续，把求和换成积分 ( $\int p(x)dx = 1$ )，推导步骤相同，得到  

$$p(x) \propto \exp(-\sum_k \lambda_k f_k(x));$$
 同样是指数族。

## 5) 两个一眼能懂的小例子

### 例 A：仅知道“均值”

- 设  $x \in \{0, 1\}$ ，约束是  $E[x] = \mu$ 。
- 取  $f_1(x) = x$ 。最大熵解：

$$p(x) \propto \exp(-\lambda x) \Rightarrow p(1) = \frac{e^{-\lambda}}{1 + e^{-\lambda}} = \frac{1}{1 + e^\lambda}.$$

令  $\theta = -\lambda$ , 就得到伯努利的**对数几率形式**:

$p(1) = \sigma(\theta)$ .  $\lambda$  (或  $\theta$ ) 由  $E[X] = \mu$  决定。

这说明: **只给定均值时, 最不偏的 0/1 分布就是伯努利**, 而伯努利正是指数族。

### 例 B: 已知 $E[X]$ 与 $E[X^2]$ (连续实线)

- 取  $f_1(x) = x, f_2(x) = x^2$ , 最大熵解:  
 $p(x) \propto \exp(\alpha x + \beta x^2)$ 。为了可积必须  $\beta < 0$ 。
- 这恰好是高斯分布的指数形式 (对数为二次多项式)。  
 结论: **只给定均值与方差时, 最不偏的实线分布是高斯** (亦属指数族)。

这两个例子强化了本节的核心: **指数族是“最大熵 + 线性矩约束”的通解**。

### 6) “最接近均匀”的含义

- 若没有任何约束 (除了归一化), 最大熵解就是**均匀分布**。
- 加了线性期望约束后, 离开“纯均匀”的最小幅度, 就是把密度改成“指数线性”那样的形状, 正是指数族。

## 要点总结

- 问题:** 只知道一些线性期望约束  $\sum_x f_k(x)p(x) = F_k$ , 选什么分布最“不引入额外假设”?

- 答案 (最大熵):** 在满足约束与归一化下, 最大化熵  $-\sum p \log p$ 。

- 解法:** 拉格朗日乘子  $\rightarrow$

$$p(x) = \frac{1}{Z} \exp(-\sum_k \lambda_k f_k(x)), \quad Z = \sum_x \exp(-\sum_k \lambda_k f_k(x)).$$

- 结论:** 最大熵解 = **指数族/吉布斯分布**; 其中

$$\phi_k(x) = f_k(x), \quad \theta_k = -\lambda_k, \quad A(\theta) = \log Z(\theta).$$

- 直觉:**  $\lambda_k$  调节每个约束的“力度”, 直到  $E[f_k(X)] = F_k$  精确满足; 没有约束就退化为**均匀分布**。

- 示例:**

- 只给  $E[X]$  ( $X \in \{0, 1\}$ )  $\rightarrow$  伯努利 (Sigmoid 形式)。
- 给  $E[X]$  与  $E[X^2]$  (实线)  $\rightarrow$  高斯 (指数二次)。

一句话: **指数族 = 在线性矩约束下的“最不偏” (最大熵) 分布。**



已上传的图片

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好，最后总结要点。

Figure 9.1 A visualization of the various features of a GLM.  
Based on Figure 8.3 of (Jordan 2007).

### 9.3 Generalized linear models (GLMs)

Linear and logistic regression are examples of generalized linear models, or GLMs (McCullagh and Nelder 1989).

These are models in which the output density is in the exponential family (Section 9.2), and in which the mean parameters are a linear combination of the inputs, passed through a possibly nonlinear function, such as the logistic function. We describe GLMs in more detail below. We focus on scalar outputs for notational simplicity. (This excludes multinomial logistic regression, but this is just to simplify the presentation.)

#### 9.3.1 Basics

To understand GLMs, let us first consider the case of an unconditional distribution for a scalar response variable:

$$p(y_i|\theta, \sigma^2) = \exp y_i \theta - A(\theta) \sigma^2 + c(y_i, \sigma^2) \quad (9.77)$$

where  $\sigma^2$  is the dispersion parameter (often set to 1),  $\theta$  is the natural parameter,  $A$  is the partition function, and  $c$  is a normalization constant. For example, in the case of logistic regression,  $\theta$  is the log-odds ratio,  $\theta = \log(\mu / (1-\mu))$ , where  $\mu = E[y] = p(y=1)$  is the mean parameter (see Section 9.2.2.1). To convert from the mean parameter to the natural parameter, we can use a function  $\psi$ , so  $\theta = \psi(\mu)$ . This function is uniquely determined by the form of the exponential family distribution. In fact, this is an invertible mapping, so we have  $\mu = \psi^{-1}(\theta)$ . Furthermore, we know from Section 9.2.3 that the mean is given by the derivative of the partition function, so we have  $\mu = \psi^{-1}(\theta) = A'(\theta)$ . Now let us add inputs/ covariates. We first define a linear function of the inputs:

$$\eta_i = w^T x_i \quad (9.78)$$

We now make the mean of the distribution be some invertible monotonic function of this linear combination. By convention, this function, known as the mean function, is denoted by  $g^{-1}$ , so

$$\mu_i = g^{-1}(\eta_i) = g^{-1}(w^T x_i) \quad (9.79)$$

See Figure 9.1 for a summary of the basic model. The inverse of the mean function, namely  $g()$ , is called the link function. We are free to choose almost any function we like for  $g$ , so long as it is invertible, and so long as  $g^{-1}$  has the appropriate range. For example, in logistic regression, we set  $\mu_i = g^{-1}(\eta_i) = \text{sigm}(\eta_i)$ .

One particularly simple form of link function is to use  $g = \psi$ ; this is called the canonical link function. In this case,  $\theta_i = \eta_i = w^T x_i$ , so the model becomes

$$p(y_i|x_i, w, \sigma^2) = \exp y_i w^T x_i - A(w^T x_i) \sigma^2 + c(y_i, \sigma^2) \quad (9.80)$$

In Table 9.1, we list some distributions and their canonical link functions. We see that for the Bernoulli/ binomial distribution, the canonical link is the logit function,  $g(\mu) = \log(\mu/(1 - \mu))$ , whose inverse is the logistic function,  $\mu = \text{sigm}(\eta)$ . Based on the results in Section 9.2.3, we can show that the mean and variance of the response variable are as follows:

$$E[y|x_i, w, \sigma^2] = \mu_i = A(\theta_i) \quad (9.81)$$

$\text{var}[y|x_i, w, \sigma^2] = \sigma^2 = A(\theta_i)\sigma^2$  To make the notation clearer, let us consider some simple examples.

- For linear regression, we have

$$\log p(y_i|x_i, w, \sigma^2) = y_i \mu_i - \frac{\mu_i^2}{2\sigma^2} - \frac{1}{2} \log(2\pi\sigma^2) \quad (9.83)$$

where  $y_i \in \mathbb{R}$ , and  $\theta_i = \mu_i = w^T x_i$ . Here  $A(\theta) = \theta^2/2$ , so  $E[y_i] = \mu_i$  and  $\text{var}[y_i] = \sigma^2$ .

- For binomial regression, we have

$$\log p(y_i|x_i, w) = y_i \log(\pi_i) + (1 - y_i) \log(1 - \pi_i) + \log N_i \quad (9.84)$$

where  $y_i \in \{0, 1, \dots, N_i\}$ ,  $\pi_i = \text{sigm}(w^T x_i)$ ,  $\theta_i = \log(\pi_i/(1 - \pi_i)) = w^T x_i$ , and  $\sigma^2 = 1$ . Here  $A(\theta) = \theta \log(1 + e^\theta)$ , so  $E[y_i] = N_i \pi_i = \mu_i$ ,  $\text{var}[y_i] = N_i \pi_i (1 - \pi_i)$ .

- For poisson regression, we have

$$\log p(y_i|x_i, w) = y_i \log \mu_i - \mu_i - \log(y_i!) \quad (9.85)$$

where  $y_i \in \{0, 1, 2, \dots\}$ ,  $\mu_i = \exp(w^T x_i)$ ,  $\theta_i = \log(\mu_i) = w^T x_i$ , and  $\sigma^2 = 1$ . Here  $A(\theta) = e^\theta$ , so  $E[y_i] = \text{var}[y_i] = \mu_i$ . Poisson regression is widely used in bio-statistical applications, where  $y_i$  might represent the number of diseases of a given person or place, or the number of reads at a

genomic location in a high-throughput sequencing context (see e.g., (Kuan et al. 2009)).

下面先给出**逐字完整翻译**（保留原式编号、不遗漏），随后提供**基础且详尽的解释**（配直觉与小例子），最后用**要点清单总结**。

## 逐字完整翻译

**图 9.1** GLM 各种要素的可视化。基于 (Jordan 2007) 的图 8.3。

### 9.3 广义线性模型 (GLMs)

线性回归和逻辑回归是广义线性模型 (GLMs; McCullagh and Nelder 1989) 的例子。这类模型的输出密度属于指数族 (第 9.2 节)，而且其**均值参数是输入的线性组合**再通过一个可能**非线性的**函数 (比如 logistic 函数) 得到。我们在下面更详细地描述 GLM。为简化记号，我们只讨论标量输出 (这排除了多项逻辑回归，但只是为了简化表述)。

#### 9.3.1 基础

为了理解 GLM，先考虑**无条件**的标量响应变量的分布：

$$p(y_i \mid \theta, \sigma^2) = \exp\left(\frac{y_i \theta - A(\theta)}{\sigma^2} + c(y_i, \sigma^2)\right) \quad (9.77)$$

其中  $\sigma^2$  是离差参数 (常设为 1)， $\theta$  是自然参数， $A$  是 (对数) 配分函数， $c$  是归一化常数。

例如，在逻辑回归中， $\theta$  是对数赔率  $\theta = \log(\mu/(1 - \mu))$ ，其中  $\mu = E[y] = p(y = 1)$  是**均值参数** (见第 9.2.2.1 节)。为了把均值参数转换为自然参数，我们可以使用一个函数  $\psi$ ，使得  $\theta = \Psi(\mu)$ 。这个函数由指数族分布的形式唯一确定。事实上，这是一个**可逆映射**，因此有  $\mu = \Psi^{-1}(\theta)$ 。此外，我们在第 9.2.3 节知道，均值由配分函数的导数给出，因此

$$\mu = \Psi^{-1}(\theta) = A'(\theta).$$

现在加入**输入/协变量**。先定义一个输入的线性函数：

$$\eta_i = w^\top x_i \quad (9.78)$$

我们现在让分布的**均值**成为这个线性组合再经过某个**可逆单调函数**的结果。按惯例，这个函数称为**均值函数**，记为  $g^{-1}$ ，因此

$$\mu_i = g^{-1}(\eta_i) = g^{-1}(w^\top x_i) \quad (9.79)$$

参见图 9.1 对基本模型的总结。均值函数的**逆函数**  $g(\cdot)$  称为**链接函数** (link)。只要  $g$  可逆、且  $g^{-1}$  的值域合适，我们几乎可以任意选择  $g$ 。例如，在逻辑回归中，我们令  $\mu_i =$

$$g^{-1}(\eta_i) = \text{sigm}(\eta_i)。$$

一种特别简单的链接是取  $g = \psi$ ; 这称为**典范链接函数** (canonical link)。此时  $\theta_i = \eta_i = w^\top x_i$ , 模型变为

$$p(y_i | x_i, w, \sigma^2) = \exp\left(\frac{y_i w^\top x_i - A(w^\top x_i)}{\sigma^2} + c(y_i, \sigma^2)\right) \quad (9.80)$$

在表 9.1 中, 我们列出了一些分布及其典范链接。可以看到, 对伯努利/二项分布, 典范链接是 logit 函数  $g(\mu) = \log(\mu/(1-\mu))$ , 其逆为 logistic 函数  $\mu = \text{sigm}(\eta)$ 。基于第 9.2.3 节的结果, 我们可以得到响应变量的均值与方差如下:

$$\mathbb{E}[y | x_i, w, \sigma^2] = \mu_i = A'(\theta_i) \quad (9.81)$$

$$\text{var}[y | x_i, w, \sigma^2] = \sigma_i^2 = A''(\theta_i) \sigma^2 \quad (9.82)$$

为使记号更清晰, 下面看几个简单例子。

- **线性回归:**

$$\log p(y_i | x_i, w, \sigma^2) = \frac{y_i \mu_i - \mu_i^2/2}{\sigma^2} - \frac{1}{2} \log(2\pi\sigma^2) - \frac{y_i^2}{2\sigma^2} \quad (9.83)$$

其中  $y_i \in \mathbb{R}$ , 且  $\theta_i = \mu_i = w^\top x_i$ 。此时  $A(\theta) = \theta^2/2$ , 所以  $\mathbb{E}[y_i] = \mu_i$ ,  $\text{var}[y_i] = \sigma^2$ 。

- **二项回归:**

$$\log p(y_i | x_i, w) = y_i \log \frac{\pi_i}{1 - \pi_i} + N_i \log(1 - \pi_i) + \log \binom{N_i}{y_i} \quad (9.84)$$

其中  $y_i \in \{0, 1, \dots, N_i\}$ ,  $\pi_i = \text{sigm}(w^\top x_i)$ ,  $\theta_i = \log(\pi_i/(1-\pi_i)) = w^\top x_i$ ,  $\sigma^2 = 1$ 。

此时  $A(\theta) = N_i \log(1 + e^\theta)$ , 故  $\mathbb{E}[y_i] = N_i \pi_i = \mu_i$ ,  $\text{var}[y_i] = N_i \pi_i (1 - \pi_i)$ 。

- **泊松回归:**

$$\log p(y_i | x_i, w) = y_i \log \mu_i - \mu_i - \log(y_i!) \quad (9.85)$$

其中  $y_i \in \{0, 1, 2, \dots\}$ ,  $\mu_i = \exp(w^\top x_i)$ ,  $\theta_i = \log(\mu_i) = w^\top x_i$ ,  $\sigma^2 = 1$ 。此时  $A(\theta) = e^\theta$ , 所以  $\mathbb{E}[y_i] = \text{var}[y_i] = \mu_i$ 。泊松回归在生物统计中广泛使用, 此时  $y_i$  可以表示某个人或地点的疾病数, 或高通量测序环境中某个基因组位置的 reads 计数 (见如 Kuan et al. 2009)。

## 通俗解释 (一步步来)

### 1) 图 9.1 在说什么?

- 左侧：参数向量  $w$  与特征  $x_i$  先做**线性组合**： $\eta_i = w^\top x_i$ （这一步就是“线性”）。
- 中间：把线性结果  $\eta_i$  经过**均值函数**  $g^{-1}$  得到**均值参数**  $\mu_i$ ： $\mu_i = g^{-1}(\eta_i)$ 。
- 右侧： $\mu_i$  再通过指数族内部的固定变换  $\Psi$  与**自然参数**  $\theta_i$  互相可逆： $\theta_i = \Psi(\mu_i)$ ， $\mu_i = \Psi^{-1}(\theta_i) = A'(\theta_i)$ 。
- 合起来：

$$(w, x_i) \xrightarrow{\text{线性}} \eta_i \xleftarrow[g]{\Psi^{-1}} \mu_i \xleftarrow{\Psi} \theta_i \xrightarrow{\text{指数族}} p(y_i | \theta_i, \sigma^2).$$

这就是 GLM 的“管道”。

## 2) GLM 的两块积木

- 输出分布来自指数族**（伯努利、二项、泊松、Gamma、高斯...）。
- 均值通过链接函数**  $g$  与线性预测子  $\eta_i = w^\top x_i$  关联：

$$g(\mu_i) = \eta_i \Leftrightarrow \mu_i = g^{-1}(w^\top x_i).$$

选择  $g$  时要保证  $g^{-1}$  的值域与分布的均值范围一致（如概率要在  $[0, 1]$ ）。

- 典范链接 (canonical link)**：取  $g = \psi$ （指数族自带的从  $\mu$  到  $\theta$  的映射），则  $\theta_i = \eta_i = w^\top x_i$ 。许多推导会更简洁（如梯度、凸性等）。

## 3) 均值与方差从哪里来 ((9.81)–(9.82)) ?

- 对任一指数族： $\mu = A'(\theta)$ ， $\text{var}[\text{充分统计量}] = A''(\theta)$ （见第 9.2.3）。
- GLM 中把  $\theta_i$  与  $x_i$  连接起来后，自然得到

$$\mathbb{E}[y_i | x_i] = \mu_i = A'(\theta_i), \quad \text{var}[y_i | x_i] = A''(\theta_i) \sigma^2,$$

其中  $\sigma^2$  是**离差参数**（有的族固定为 1，如伯努利/泊松；有的可估计，如高斯）。

## 4) 三个经典 GLM 例子 (快速直觉)

- 线性回归 (高斯族, 恒等链接)**：
 
$$g(\mu) = \mu \Rightarrow \mu_i = w^\top x_i.$$

$$A(\theta) = \theta^2/2 \Rightarrow \mathbb{E}[y] = \mu, \text{var}[y] = \sigma^2.$$
 这就是“均值=线性，噪声=常方差”的普通线性回归。
- 二项 (逻辑) 回归 (伯努利/二项族, logit 链接)**：
 
$$g(\mu) = \log(\mu/(1 - \mu)) \Rightarrow \mu_i = \text{sigm}(w^\top x_i).$$

$$\mathbb{E}[y] = N_i \mu_i, \text{var}[y] = N_i \mu_i (1 - \mu_i).$$
 概率来自 Sigmoid，方差随均值变化（异方差）。
- 泊松回归 (泊松族, log 链接)**：
 
$$g(\mu) = \log \mu \Rightarrow \mu_i = \exp(w^\top x_i).$$

$E[y] = \text{var}[y] = \mu_i$  (均值=方差) , 常用于计数数据。

小提示：选择链接函数时，**典范链接**通常带来更简洁的学习与推断；但在实践中，也会为了可解释性或稳健性选择非典范链接（如对 Gamma 族用 log 或 inverse 链接等）。

## 5) 关于式 (9.83) 的读法

- (9.83) 把高斯对数似然按指数族模板拆成若干项；把它合并起来就是熟悉的：

$$\log p(y_i | \mu_i, \sigma^2) = -\frac{(y_i - \mu_i)^2}{2\sigma^2} - \frac{1}{2} \log(2\pi\sigma^2).$$

其中  $\mu_i = w^\top x_i$ 。这与 (9.83) 等价。

## 要点总结

- **GLM 定义**: 输出分布  $\in$  指数族；用**链接函数**  $g$  把**均值**  $\mu_i$  与**线性预测子**  $\eta_i = w^\top x_i$  关联： $g(\mu_i) = \eta_i$ 。
- **典范链接**: 令  $g = \psi$ , 则  $\theta_i = \eta_i$ , 推导最简洁。
- **均值/方差**:  $E[y_i | x_i] = A'(\theta_i)$ ,  $\text{var}[y_i | x_i] = A''(\theta_i)\sigma^2$  ( $\sigma^2$  为离差)。
- **三大实例**:
  - 线性回归：高斯族 + 恒等链接， $\mu = w^\top x$ , 方差常数。
  - 逻辑/二项回归：伯努利/二项族 + logit 链接， $\mu = \text{sigm}(w^\top x)$ , 方差  $N\mu(1 - \mu)$ 。
  - 泊松回归：泊松族 + log 链接， $\mu = \exp(w^\top x)$ ,  $E[y] = \text{var}[y] = \mu$ 。
- **图 9.1 管线**:  $(w, x) \rightarrow \eta \xleftarrow{g^{-1}, g} \mu \xleftarrow{\Psi, \Psi^{-1}} \theta \rightarrow p(y | \theta, \sigma^2)$ 。
- **实践**: 根据任务选择合适的**族** (数据类型) 与**链接** (值域/解释性)，GLM 就把“线性信号”变成“合规的均值”，并继承指数族带来的良好统计与优化性质 (凸性、共轭等)。

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好，最后总结要点。

### 9.3.2 ML and MAP estimation

One of the appealing properties of GLMs is that they can be fit using exactly the same methods that we used to fit

logistic regression. In particular, the log-likelihood has the following form:

$$(w) = \log p(D|w) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^{d_i} \eta_j - A(\theta_j) \quad (9.86)$$

$$\eta_j = \theta_j y_{ij} - A(\theta_j) \quad (9.87)$$

We can compute the gradient vector using the chain rule as follows:

$$\frac{\partial \ell}{\partial \theta_j} = \sum_{i=1}^N \sum_{j=1}^{d_i} \eta_j \frac{\partial \eta_j}{\partial \theta_j} = \sum_{i=1}^N \sum_{j=1}^{d_i} (y_{ij} - \mu_{ij}) \frac{\partial \mu_{ij}}{\partial \theta_j} \quad (9.88)$$

$$= \sum_{i=1}^N (y_{ij} - \mu_{ij}) \frac{\partial \mu_{ij}}{\partial \theta_j} \quad (9.89)$$

$$= \sum_{i=1}^N (y_{ij} - \mu_{ij}) x_{ij} \quad (9.90)$$

If we use a canonical link,  $\theta_j = \eta_j$ , this simplifies to

$$\nabla_w \ell(w) = -\frac{1}{2} \sum_{i=1}^N (y_{ij} - \mu_{ij}) x_{ij} \quad (9.91)$$

which is a sum of the input vectors, weighted by the errors. This can be used inside a (stochastic) gradient descent procedure, discussed in Section 8.5.2. However, for improved efficiency, we should use a second-order method. If we use a canonical link, the Hessian is given by

$$H = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^{d_i} \frac{\partial^2 \mu_{ij}}{\partial \theta_j^2} x_{ij} x_{ij}^T = -\frac{1}{2} \sigma^2 X S X^T \quad (9.92)$$

where  $S = \text{diag}(\frac{\partial \mu_1}{\partial \theta_1}, \dots, \frac{\partial \mu_N}{\partial \theta_N})$  is a diagonal weighting matrix. This can be used inside the IRLS algorithm (Section 8.3.4). Specifically, we have the following Newton update:

$$w^{t+1} = (X^T S X)^{-1} X^T S z^t \quad (9.93)$$

$$z^t = \theta^t + S^{-1} (y - \mu^t) \quad (9.94)$$

where  $\theta^t = X w^t$  and  $\mu^t = g^{-1}(\eta^t)$ . If we extend the derivation to handle non-canonical links, we find that the Hessian has another term. However, it turns out that the expected Hessian is the same as in Equation 9.92; using the expected Hessian (known as the Fisher information matrix) instead of the actual Hessian is known as the Fisher scoring method. It is straightforward to modify the above procedure to perform MAP estimation with a Gaussian prior: we just modify the objective, gradient and Hessian, just as we added L2 regularization to logistic regression in Section 8.3.6.

### 9.3.3 Bayesian inference

Bayesian inference for GLMs is usually conducted using MCMC (Chapter 24). Possible methods include Metropolis Hastings with an IRLS-based proposal (Gamerman 1997), Gibbs sampling using adaptive rejection sampling (ARS) for each full-conditional (Dellaportas and Smith 1993), etc. See e.g., (Dey et al. 2000) for further information. It is also possible to use the Gaussian approximation (Section 8.4.1) or variational inference (Section 21.8.1.1).

已思考 34s

下面先给出逐字完整翻译（不遗漏原式与编号），接着做基础且详尽的解释（一步步推导 + 直觉），最后给出要点总结。

---

## 逐字完整翻译

### 9.3.2 极大似然 (ML) 与最大后验 (MAP) 估计

GLM 的一个吸引人的性质是：它们可以用与拟合逻辑回归完全相同的方法来拟合。具体地，对数似然具有如下形式：

$$\ell(w) = \log p(D | w) = \frac{1}{\sigma^2} \sum_{i=1}^N \ell_i \quad (9.86)$$

$$\ell_i = \theta_i y_i - A(\theta_i) \quad (9.87)$$

我们可以用链式法则来计算梯度向量：

$$\frac{d \ell_i}{d w_j} = \frac{d \ell_i}{d \theta_i} \frac{d \theta_i}{d \mu_i} \frac{d \mu_i}{d \eta_i} \frac{d \eta_i}{d w_j} \quad (9.88)$$

$$= (y_i - A'(\theta_i)) \frac{d \theta_i}{d \mu_i} \frac{d \mu_i}{d \eta_i} x_{ij} \quad (9.89)$$

$$= (y_i - \mu_i) \frac{d \theta_i}{d \mu_i} \frac{d \mu_i}{d \eta_i} x_{ij} \quad (9.90)$$

如果使用典范链接， $\theta_i = \eta_i$ ，上述式子化简为

$$\nabla_w \ell(w) = \frac{1}{\sigma^2} \sum_{i=1}^N (y_i - \mu_i) x_i \quad (9.91)$$

这就是把输入向量按“误差”加权后求和。它可用于（随机）梯度下降（见 8.5.2 节）。不过，为了更高效率，应该使用二阶方法。若用典范链接，Hessian 为

$$H = -\frac{1}{\sigma^2} \sum_{i=1}^N \frac{d \mu_i}{d \theta_i} x_i x_i^\top = -\frac{1}{\sigma^2} X^\top S X \quad (9.92)$$

其中  $S = \text{diag}(\frac{d \mu_1}{d \theta_1}, \dots, \frac{d \mu_N}{d \theta_N})$  是对角权重矩阵。它可以用于 IRLS 算法（见 8.3.4 节）。具体地，有如下牛顿更新：

$$w_{t+1} = (X^\top S_t X)^{-1} X^\top S_t z_t \quad (9.93)$$

$$z_t = \theta_t + S_t^{-1} (y - \mu_t) \quad (9.94)$$

其中  $\theta_t = Xw_t$ ,  $\mu_t = g^{-1}(\eta_t)$ 。

若把推导扩展到**非典范链接**, Hessian 会多出一项。不过事实是：**期望的** Hessian 与式 (9.92) 相同；用期望 Hessian (即 Fisher 信息矩阵) 代替真实 Hessian 的做法称为 **Fisher scoring**。

要做带高斯先验的 MAP 估计也很简单：像在 8.3.6 节把  $l_2$  正则加到逻辑回归里那样，修改目标函数、梯度与 Hessian 即可。

### 9.3.3 贝叶斯推断

GLM 的贝叶斯推断通常用 MCMC (第 24 章)。可选方法包括：用基于 IRLS 的提议分布做 Metropolis–Hastings (Gamerman 1997)、对每个条件后验使用 **自适应拒绝采样 (ARS)** 的 Gibbs 采样 (Dellaportas and Smith 1993) 等。更多信息可见 Dey 等 (2000)。也可以使用**高斯近似** (第 8.4.1 节) 或**变分推断** (第 21.8.1.1 节)。

## 通俗解释（一步一步来）

### 1) GLM 的对数似然为什么这么“像”逻辑回归？

- 指数族下，单样本的对数似然统一长成： $\ell_i = \theta_i y_i - A(\theta_i) + (\text{与 } y_i \text{ 有关的常数})$ 。式 (9.87) 只写了与  $w$  相关的那部分。把所有样本求和并除以  $\sigma^2$  (离差参数) 得到 (9.86)。

### 2) 梯度推导 ((9.88)–(9.91))

- 关键链条： $w \rightarrow \eta_i = w^\top x_i \rightarrow \mu_i = g^{-1}(\eta_i) \rightarrow \theta_i = \Psi(\mu_i) \rightarrow \ell_i(\theta_i)$ 。
- 用链式法则把导数一段段乘起来：

$$\frac{d\ell_i}{dw_j} = \underbrace{(y_i - A'(\theta_i))}_{=y_i-\mu_i} \cdot \frac{d\theta_i}{d\mu_i} \cdot \underbrace{\frac{d\mu_i}{d\eta_i}}_{=x_{ij}} \cdot \underbrace{\frac{d\eta_i}{dw_j}}_{=x_{ij}}.$$

- 若用**典范链接**  $g = \Psi$ , 则  $\theta_i = \eta_i$ , 于是  $\frac{d\theta_i}{d\mu_i} \frac{d\mu_i}{d\eta_i} = 1$  (互为反函数), 梯度简化为 (9.91)：

$$\nabla_w \ell(w) = \frac{1}{\sigma^2} \sum_i (y_i - \mu_i) x_i.$$

直觉：像在线性回归里“残差  $\times$  特征”的叠加；这里只是  $\mu_i$  的计算方式依赖所选的指数族/链接。

### 三个常用族的 $\mu$ 与导数

- 高斯 (线性回归, 恒等链接)** :  $\mu_i = \eta_i$ ,  $\frac{d\mu}{d\theta} = 1$ 。

- **伯努利/二项 (逻辑回归, logit 链接)** :  $\mu_i = \sigma(\eta_i)$ ,  $\frac{d\mu}{d\theta} = \mu_i(1 - \mu_i)$ 。
- **泊松 (log 链接)** :  $\mu_i = e^{\eta_i}$ ,  $\frac{d\mu}{d\theta} = \mu_i$ 。

### 3) 二阶信息、IRLS 与牛顿更新 ((9.92)–(9.94))

- 在典范链接下, Hessian:

$$H = -\frac{1}{\sigma^2} \sum_i \frac{d\mu_i}{d\theta_i} x_i x_i^\top = -\frac{1}{\sigma^2} X^\top S X,$$

其中  $S = \text{diag}(\frac{d\mu_i}{d\theta_i})$  给每个样本一个权重。

- 把牛顿法写成一次**加权最小二乘**, 就得到 IRLS:

$$w_{t+1} = (X^\top S_t X)^{-1} X^\top S_t z_t, \quad z_t = \theta_t + S_t^{-1}(y - \mu_t).$$

这里  $z_t$  叫**工作响应** (working response),  $\mu_t = g^{-1}(X w_t)$ 。

- **非典范链接**: 真实 Hessian 会再多一项, 但用**期望 Hessian** (Fisher 信息) 可回到 (9.92)。用期望 Hessian 的牛顿法就叫 Fisher scoring, 数值上更稳一些。

### 4) 从 ML 到 MAP (带高斯先验)

- 取先验  $w \sim N(0, \lambda^{-1} I)$ 。则

$$\log p(w) = -\frac{\lambda}{2} \|w\|^2 + \text{常数}.$$

- MAP 就是最大化  $\ell(w) + \log p(w)$ ; 相当于在 ML 目标上加一个  $l_2$  正则项 (系数  $\lambda$ ), 梯度多出  $-\lambda w$ , Hessian 多出  $-\lambda I$ 。这与 8.3.6 节对逻辑回归做  $l_2$  正则是同一件事。

### 5) 贝叶斯推断 (9.3.3)

- MCMC:
  - MH + IRLS 提议: 用一次 IRLS 迭代给出一个高质量的高斯提议分布, 再做 Metropolis-Hastings 接受/拒绝。
  - Gibbs + ARS: 若全条件后验对数是凹的, 可用**自适应拒绝采样** 高效地从一维条件分布抽样。
- **近似贝叶斯**:
  - **高斯 (拉普拉斯) 近似**: 在后验众数处二阶展开, 得到近似  $N(\hat{w}, (-H(\hat{w}))^{-1})$ 。
  - **变分推断**: 设定一个可导的近似族 (常为高斯族或均值场), 最小化 KL, 速度快、可扩展。

# 要点总结

- GLM 的 ML 与逻辑回归几乎同款：

$$\ell_i = \theta_i y_i - A(\theta_i),$$

典范链接下  $\nabla_w \ell = \frac{1}{\sigma^2} \sum_i (y_i - \mu_i) x_i$ 。

- Hessian/IRLS：

$$H = -\frac{1}{\sigma^2} X^\top S X, \quad S = \text{diag}\left(\frac{d\mu_i}{d\theta_i}\right);$$

牛顿更新  $w_{t+1} = (X^\top S X)^{-1} X^\top S z, \quad z = \theta + S^{-1}(y - \mu)$ 。

非典范链接时用 Fisher scoring (期望 Hessian)。

- MAP = ML +  $l_2$  正则：高斯先验让目标、梯度、Hessian 分别加上  $-\frac{\lambda}{2} \|w\|^2$ 、 $-\lambda w$ 、 $-\lambda I$ 。
- 贝叶斯 GLM：常用 MCMC (MH+IRLS 提议、Gibbs+ARS)，也可用高斯近似/变分推断，权衡精度与效率。
- 实操记忆：选族+选链接 → 一阶是“误差×特征”，二阶给出加权最小二乘 (IRLS)；加高斯先验即是  $l_2$  正则；做全贝叶斯则用 MCMC 或近似贝叶斯。



已上传的图片

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好，最后总结要点。

Table 9.2 Summary of some possible mean functions for binary regression.

## 9.4 Probit regression

In (binary) logistic regression, we use a model of the form  $p(y=1|x_i, w) = \text{sigm}(w^\top x_i)$ . In general, we can write  $p(y=1|x_i, w) = g^{-1}(w^\top x_i)$ , for any function  $g^{-1}$  that maps  $[-\infty, \infty]$  to  $[0, 1]$ . Several possible mean functions are listed in Table 9.2. In this section, we focus on the case where  $g^{-1}(\eta) = \Phi(\eta)$ , where  $\Phi(\eta)$  is the cdf of the standard normal. This is known as probit regression. The probit function is very similar to the logistic function, as shown in Figure 8.7(b). However, this model has some advantages over logistic regression, as we will see.

### 9.4.1 ML/MAP estimation using gradient-based optimization

We can find the MLE for probit regression using standard gradient methods. Let  $\mu_i = w^T x_i$ , and let  $\tilde{y}_i \in \{-1, +1\}$ .

Then the gradient of the log-likelihood for a specific case is given by

$$g_i = d \frac{dw}{d\mu_i} \log p(\tilde{y}_i | w^T x_i) = d\mu_i \frac{dw}{d\mu_i} \log p(\tilde{y}_i | w^T x_i) = x_i \tilde{y}_i \varphi(\mu_i) \Phi(\tilde{y}_i \mu_i) \quad (9.95)$$

where  $\varphi$  is the standard normal pdf, and  $\Phi$  is its cdf.

Similarly, the Hessian for a single case is given by

$$H_i = d \frac{dw^2}{d\mu_i^2} \log p(\tilde{y}_i | w^T x_i) = -x_i \varphi(\mu_i)^2 \Phi(\tilde{y}_i \mu_i)^2 + \tilde{y}_i \mu_i \varphi(\mu_i) \Phi(\tilde{y}_i \mu_i) x_i^T \quad (9.96)$$

We can modify these expressions to compute the MAP estimate in a straightforward manner. In particular, if we use the prior  $p(w) = N(0, V_0)$ , the gradient and Hessian of the penalized log likelihood have the form  $i g_i + 2V^{-1} 0^T w$  and  $i H_i + 2V^{-1} 0^T$ . These expressions can be passed to any gradient-based optimizer. See `probitRegDemo` for a demo.

已思考 56s

下面先给出**逐字完整翻译**（包含表格与所有公式、编号），再做**基础且详尽的解释**（一步  
步、直觉友好），最后给出**要点总结**。

## 逐字完整翻译

**表 9.2** 一些用于二元回归的可能**均值函数** (mean functions) 汇总。

名称	公式
Logistic (逻辑)	$g^{-1}(\eta) = \text{sigm}(\eta) = \frac{e^\eta}{1 + e^\eta}$
Probit	$g^{-1}(\eta) = \Phi(\eta)$
Log-log	$g^{-1}(\eta) = \exp(-\exp(-\eta))$
Complementary log-log (互补对数-对数, cloglog)	$g^{-1}(\eta) = 1 - \exp(-\exp(\eta))$

## 9.4 Probit 回归

在（二元）逻辑回归中，我们使用形式为  $p(y = 1 | x_i, w) = \text{sigm}(w^\top x_i)$  的模型。更一般地，对于任何把  $[-\infty, \infty]$  映到  $[0, 1]$  的函数  $g^{-1}$ ，我们都可以说成

$$p(y = 1 | x_i, w) = g^{-1}(w^\top x_i)。$$

若干可能的均值函数列在表 9.2 中。本节我们聚焦于  $g^{-1}(\eta) = \Phi(\eta)$  的情况，其中  $\Phi(\eta)$  是标准正态分布的 cdf，这称为 **probit 回归**。如图 8.7(b) 所示，probit 函数与 logistic 函数非常相似。不过，正如我们将看到的，这个模型相对逻辑回归有一些优势。

#### 9.4.1 用基于梯度的优化做 ML/MAP 估计

我们可以用标准的梯度方法求 probit 回归的极大似然估计 (MLE)。令  $\mu_i = w^\top x_i$ ，并令  $\tilde{y}_i \in \{-1, +1\}$ 。则对单个样本，其对数似然的梯度为

$$g_i = \frac{d}{dw} \log p(\tilde{y}_i | w^\top x_i) = \frac{d\mu_i}{dw} \frac{d}{d\mu_i} \log p(\tilde{y}_i | w^\top x_i) = x_i \frac{\tilde{y}_i \phi(\mu_i)}{\Phi(\tilde{y}_i \mu_i)} \quad (9.95)$$

其中  $\phi$  是标准正态的 pdf， $\Phi$  是其 cdf。类似地，单个样本的 Hessian 为

$$H_i = \frac{d^2}{dw^2} \log p(\tilde{y}_i | w^\top x_i) = -x_i \left( \frac{\phi(\mu_i)^2}{\Phi(\tilde{y}_i \mu_i)^2} + \frac{\tilde{y}_i \mu_i \phi(\mu_i)}{\Phi(\tilde{y}_i \mu_i)} \right) x_i^\top \quad (9.96)$$

我们可以直接改写上述表达式来做 MAP 估计。特别地，如果先验  $p(w) = N(0, V_0)$ ，则 **惩罚对数似然**的梯度与 Hessian 分别形如  $\sum_i g_i + 2V_0^{-1}w$  与  $\sum_i H_i + 2V_0^{-1}$ 。这些量可以传给任意基于梯度的优化器。参见 *probitRegDemo* 作为演示。

## 通俗解释（一步步来）

### 1) 表 9.2 里的四种“均值函数”是什么？

GLM 里，用**均值函数**  $g^{-1}$  把线性预测子  $\eta = w^\top x$  映到概率  $[0, 1]$ 。

- **Logistic (逻辑)** :  $\sigma(\eta) = \frac{1}{1 + e^{-\eta}}$ 。S 形、对称、最常用。
- **Probit**:  $\Phi(\eta)$ ，标准正态的累积分布函数。曲线与 logistic 很接近（把  $\eta$  乘约 1.6 后几乎重合），但在贝叶斯推断/潜变量建模里更友好（见下）。
- **Log-log**:  $\exp(-e^{-\eta})$ ; **cloglog**:  $1 - \exp(-e^{\eta})$ 。这两个不对称，常用于“接近 0 (或 1) 且随  $\eta$  变化不对称”的命中率/生存分析/极值型数据。

任选哪一个都行，但要保证  $g^{-1}$  输出在  $[0, 1]$ ，并与任务的物理直觉匹配（是否对称、尾部速度等）。

### 2) 什么是 probit 回归？和逻辑回归差在哪？

- 模型:  $p(y = 1 | x) = \Phi(w^\top x)$ 。
- 优点直觉:
  - 可以写成潜在高斯阈值模型: 令  $z_i \sim N(w^\top x_i, 1)$ , 观察  $y_i = I[z_i > 0]$ 。这让许多贝叶斯方法 (如 Gibbs) 特别顺滑: 在给定  $w$  时,  $z_i$  是截断高斯; 给定  $z_i$  时,  $w$  的条件后验是高斯。
  - 与误差服从正态的“阈值决策”故事天然契合。
- 与 logistic 的性能差异通常很小; 更多是计算与解释偏好。

### 3) 为什么把标签写成 $\tilde{y}_i \in \{-1, +1\}$ ?

这样对单样本的对数似然就是  $\log \Phi(\tilde{y}_i \mu_i)$ 。

- 若  $\tilde{y}_i = +1$ , 就要  $\mu_i$  越大越好;
- 若  $\tilde{y}_i = -1$ , 就要  $\mu_i$  越小越好 (因为  $\Phi(-\mu) = 1 - \Phi(\mu)$ )。

### 4) 梯度 (9.95) 从哪里来? (直观推导)

- 单点目标:  $\ell_i(\mu) = \log \Phi(\tilde{y}_i \mu)$ 。
- 对  $\mu$  求导:  $\frac{d\ell_i}{d\mu} = \frac{1}{\Phi(\tilde{y}_i \mu)} \cdot \phi(\tilde{y}_i \mu) \cdot \tilde{y}_i = \frac{\tilde{y}_i \phi(\mu)}{\Phi(\tilde{y}_i \mu)}$   
(用到  $\phi$  的偶函数性)。
- 再乘链式法则  $\frac{d\mu}{dw} = x_i$ , 得到  

$$g_i = x_i \frac{\tilde{y}_i \phi(\mu_i)}{\Phi(\tilde{y}_i \mu_i)}.$$
- 直觉: 当模型把正例的  $\mu_i$  推得更大时,  $\Phi(\tilde{y}_i \mu_i)$  变大、分母变大、梯度变小  $\rightarrow$  逐渐“满意”; 反之梯度变大, 催促调参。

### 5) Hessian (9.96) 是什么样子? (二阶直觉)

- 先对  $\frac{\tilde{y}_i \phi(\mu)}{\Phi(\tilde{y}_i \mu)}$  对  $\mu$  再求导 (用商法则和  $\phi'(\mu) = -\mu \phi(\mu)$ 、 $\Phi'(\tilde{y}_i \mu) = \tilde{y}_i \phi(\mu)$ ) :

$$\frac{d^2 \ell_i}{d\mu^2} = -\left( \frac{\phi(\mu)^2}{\Phi(\tilde{y}_i \mu)^2} + \frac{\tilde{y}_i \mu \phi(\mu)}{\Phi(\tilde{y}_i \mu)} \right).$$

- 再乘  $x_i x_i^\top$  即得矩阵形式:

$$H_i = \frac{d^2 \ell_i}{dw^2} = \frac{d^2 \ell_i}{d\mu^2} x_i x_i^\top = -x_i (\cdots) x_i^\top.$$

- 因为系数为负,  $\ell$  是严格凹 (单点), 总体目标的二阶信息可用于牛顿/拟牛顿 (更快收敛)。

### 6) 怎么做 MAP?

- 设  $w \sim N(0, V_0)$ 。把  $\log p(w) = -\frac{1}{2}w^\top V_0^{-1}w + \text{常数}$  加进总目标。
  - 于是梯度在  $\sum_i g_i$  的基础上多出一个线性惩罚项（与  $V_0^{-1}w$  成正比），Hessian多出一个常数阵（与  $V_0^{-1}$  成正比）。
  - 不同教材对“是否最大化负损失/是否带 1/2 因子”的符号与系数约定略有差别；实现时保持前后一致即可。核心思想：高斯先验 =  $l_2$  正则。
- 

## 要点总结

- 二元 GLM 可写  $p(y = 1 | x) = g^{-1}(w^\top x)$ 。常见  $g^{-1}$ : Logistic、Probit、Log-log、cloglog (表 9.2)。
- Probit 回归：选择  $g^{-1} = \Phi$ 。与 logistic 曲线近似，但有潜在高斯阈值解释，便于贝叶斯采样与数据增强。
- 梯度 (单样本) :  $g_i = x_i \frac{\tilde{y}_i \phi(\mu_i)}{\Phi(\tilde{y}_i \mu_i)}$ 。
- Hessian (单样本) :  $H_i = -x_i \left( \frac{\phi(\mu_i)^2}{\Phi(\tilde{y}_i \mu_i)^2} + \frac{\tilde{y}_i \mu_i \phi(\mu_i)}{\Phi(\tilde{y}_i \mu_i)} \right) x_i^\top$  (凹性)。
- MAP: 加高斯先验等价于加  $l_2$  正则；梯度与 Hessian 只需再加上与  $V_0^{-1}$  有关的项，即可传给任意梯度型优化器。
- 选哪种均值函数，取决于数据特性 (对称/不对称、尾部速度) 与计算偏好；probit 在贝叶斯框架下常更方便。

翻译并解释以下内容，翻译不能有遗漏，解释请基础且详尽，因为我的数学基础不太好，最后总结要点。

### 9.4.2 Latent variable interpretation

We can interpret the probit (and logistic) model as follows. First, let us associate each item  $x_i$  with two latent utilities,  $u_{0i}$  and  $u_{1i}$ , corresponding to the possible choices of  $y_i = 0$  and  $y_i = 1$ . We then assume that the observed choice is whichever action has larger utility.

More precisely, the model is as follows:

$$u_{0i} = w^\top 0 x_i + \delta_{0i} \quad (9.97)$$

$$u_{1i} = w^\top 1 x_i + \delta_{1i} \quad (9.98)$$

$$y_i = I(u_{1i} > u_{0i}) \quad (9.99)$$

where  $\delta$ 's are error terms, representing all the other

factors that might be relevant in decision making that we have chosen not to (or are unable to) model. This is called a random utility model or RUM (McFadden 1974; Train 2009). Since it is only the difference in utilities that matters, let us define  $z_i = u_{1i} - u_{0i} + \delta_i$ , where  $i = \delta_1 i - \delta_0 i$ . If the  $\delta$ 's have a Gaussian distribution, then so does  $i$ . Thus we can write

$$z_i = w^\top x_i + i \quad (9.100)$$

$$i \sim N(0, 1) \quad (9.101)$$

$$y_i = 1 = I(z_i \geq 0) \quad (9.102)$$

Following (Fruhwirth-Schnatter and Fruhwirth 2010), we call this the difference RUM or dRUM model. When we marginalize out  $z_i$ , we recover the probit model:

$$p(y_i = 1 | x_i, w) = I(z_i \geq 0)N(z_i | w^\top x_i, 1)dz_i \quad (9.103)$$

$$= p(w^\top x_i + i \geq 0) = p(i \geq -w^\top x_i) \quad (9.104)$$

$$= 1 - \Phi(-w^\top x_i) = \Phi(w^\top x_i) \quad (9.105)$$

where we used the symmetry of the Gaussian.<sup>3</sup> This latent variable interpretation provides an alternative way to fit the model, as discussed in Section 11.4.6. Interestingly, if we use a Gumbel distribution for the  $\delta$ 's, we induce a logistic distribution for  $i$ , and the model reduces to logistic regression. See Section 24.5.1 for further details.

已思考 15s

下面先给出**逐字完整翻译**（保留式号、无遗漏），再给出**基础且详尽的解释**（一步步、直觉友好），最后给出**要点总结**。

## 逐字完整翻译

### 9.4.2 潜变量 (latent variable) 解释

我们可以如下解释 probit (以及 logistic) 模型。首先，把每个样本  $x_i$  与两个**潜在效用** (latent utilities) 联系起来，分别是  $u_{0i}$  与  $u_{1i}$ ，对应两种可能的选择  $y_i = 0$  与  $y_i = 1$ 。然后我们假设观察到的选择是**效用更大的**那个动作。更精确地，模型如下：

$$u_{0i} = w_0^\top x_i + \delta_{0i} \quad (9.97)$$

$$u_{1i} = w_1^\top x_i + \delta_{1i} \quad (9.98)$$

$$y_i = I(u_{1i} > u_{0i}) \quad (9.99)$$

其中  $\delta$  是误差项，表示那些在决策中可能相关但我们选择不去（或无法）建模的其他因素。这称为**随机效用模型** (random utility model, RUM; McFadden 1974; Train 2009)。由于只有效用的差值才重要，我们令

$$z_i = u_{1i} - u_{0i} + \varepsilon_i, \text{ 其中 } \varepsilon_i = \delta_{1i} - \delta_{0i}.$$

如果这些  $\delta$  服从高斯分布，那么  $\varepsilon_i$  也服从高斯分布。因此我们可以写成

$$z_i = w^\top x_i + \varepsilon_i \quad (9.100)$$

$$\varepsilon_i \sim N(0, 1) \quad (9.101)$$

$$y_i = 1 \Leftrightarrow I(z_i \geq 0) \quad (9.102)$$

遵循 (Frühwirth-Schnatter and Frühwirth 2010)，我们称其为**差分 RUM (dRUM) 模型**。当我们边缘化掉  $z_i$  时，就得到了 probit 模型：

$$p(y_i = 1 | x_i, w) = \int I(z_i \geq 0) N(z_i | w^\top x_i, 1) dz_i \quad (9.103)$$

$$= \Pr(w^\top x_i + \varepsilon \geq 0) = \Pr(\varepsilon \geq -w^\top x_i) \quad (9.104)$$

$$= 1 - \Phi(-w^\top x_i) = \Phi(w^\top x_i) \quad (9.105)$$

这里我们利用了高斯分布的对称性。<sup>3</sup> 这种潜变量解释提供了拟合该模型的另一种方法，见 11.4.6 节。很有意思的是，如果我们对  $\delta$  使用 Gumbel 分布，那么  $\varepsilon$  的分布就是 logistic 分布，模型便退化为逻辑回归。详见 24.5.1 节。

## 通俗解释（一步步来）

### 1) “潜在效用”怎么理解？

- 想像每个样本  $x_i$  面临两个选择：选 0 或选 1。每个选择都有一个看不见的效用：

$$u_{0i} = w_0^\top x_i + \delta_{0i}, \quad u_{1i} = w_1^\top x_i + \delta_{1i}.$$

- $w_0, w_1$ ：由可观测特征  $x_i$  线性打分形成的**系统性效用**；
- $\delta_{0i}, \delta_{1i}$ ：**随机扰动/遗漏因素**，我们不建模细节，把它们当噪声。
- 规则：哪一边效用更大就选哪一边（式 9.99）。

### 2) 只关心“差值”就够了

- 因为决定是比较  $u_{1i}$  与  $u_{0i}$ ，所以只需考虑它们的差：

$$u_{1i} - u_{0i} = (w_1 - w_0)^\top x_i + (\delta_{1i} - \delta_{0i}).$$

- 令  $w = w_1 - w_0$  (把两套权重合并成一套) , 令  $\varepsilon_i = \delta_{1i} - \delta_{0i}$ , 就得到阈值模型

$$z_i = w^\top x_i + \varepsilon_i, \quad y_i = I(z_i \geq 0).$$

这就是式 (9.100)–(9.102) 的含义。文本直接把  $\varepsilon_i$  设为  $N(0, 1)$  (标准差 1) ; 即使一开始方差是  $\sigma^2$ , 也可以把  $w$  按比例缩放, 使其标准化为 1 (这叫“位置-尺度同一性”)。

### 3) 为什么会得到 probit?

- 如果  $\varepsilon \sim N(0, 1)$ , 那么

$$\Pr(y_i = 1 | x_i) = \Pr(z_i \geq 0) = \Pr(\varepsilon \geq -w^\top x_i) = 1 - \Phi(-w^\top x_i) = \Phi(w^\top x_i).$$

- 这正是 probit 链接:  $p(y = 1 | x) = \Phi(w^\top x)$ 。

直觉: 我们有一个潜在高斯分数  $z_i$ , 只看到它是否过阈值 0。阈值化高斯  $\Rightarrow \Phi$  形的成功概率。

### 4) 为什么 logistic 也能从 RUM 得到?

- 若把  $\delta$  假设为\*\*Gumbel (极值 I 型) \*\*噪声, 那么差值  $\varepsilon = \delta_1 - \delta_0$  服从 \*\*logistic (洛吉斯蒂) \*\*分布。
- Logistic 分布的 cdf 正是  $\sigma(\cdot)$ , 因此得到

$$\Pr(y = 1 | x) = \sigma(w^\top x),$$

也就是逻辑回归。这解释了两种模型都可视为“随机效用 + 阈值决策”的特例, 只是噪声分布不同: 高斯对应 probit, Gumbel 对应 logit。

### 5) 潜变量视角带来的计算好处

- 有了潜在的  $z_i$ , 在贝叶斯推断/EM/数据增强里更容易:
  - 在给定  $w$  与  $y_i$  时,  $z_i$  是截断高斯;
  - 在给定所有  $z_i$  时,  $w$  的条件后验是高斯 (线性回归结构)。
- 因此可以用 Gibbs 采样 或 EM/ECME 之类的方法高效拟合 (教材稍后章节会展开)。

## 要点总结

- RUM 思想:** 每个选择有一个“线性打分 + 噪声”的潜在效用; 选择效用更大的那一项。
- 差分化:** 只需看差值  $z = w^\top x + \varepsilon$ , 再做阈值  $y = I(z \geq 0)$ 。
- probit 的由来:** 噪声  $\varepsilon \sim N(0, 1) \Rightarrow p(y = 1 | x) = \Phi(w^\top x)$ 。
- logistic 的由来:** 若  $\delta \sim \text{Gumbel} \Rightarrow \varepsilon \sim \text{logistic} \Rightarrow p(y = 1 | x) = \sigma(w^\top x)$ 。

- **计算优势**: 潜变量  $z$  让贝叶斯推断与数据增强 (Gibbs/EM) 更顺手: 截断高斯  $\rightarrow$  线性高斯更新。
- **可识别性**: 误差方差可被缩放吸收, 通常标准化为 1; 因此只估计  $w$  即可。
- **直觉**: probit / logit 都是“线性信号 + 随机噪声”的阈值化选择模型; 不同只是噪声分布, 从而得到  $\Phi(\cdot)$  或  $\sigma(\cdot)$  的概率曲线。

### 9.4.2讲的是什么,我不太理解

已思考 17s

没关系! 9.4.2 的核心思想其实很直白: **把二分类看成“隐藏分数 + 阈值决策”的故事**。下面用极少的公式、配上直觉来讲清楚。

---

## 一句话版

每个样本都有一个**潜在分数**  $z$ 。这个分数=“线性信号”+噪声。如果  $z \geq 0$  就判为 1, 否则判为 0。

- 噪声取 **高斯**  $\Rightarrow p(y = 1 | x) = \Phi(w^\top x)$ : **probit 回归**
  - 噪声取 **Logistic** (由两份 Gumbel 误差之差得到)  $\Rightarrow p(y = 1 | x) = \sigma(w^\top x)$ : **逻辑回归**
- 

## 分步直觉

### 1) 两个选择 $\rightarrow$ 两个“效用”

对每个样本  $x_i$ , 想像有两个方案: 选 0 或选 1。各自有一个看不见的“效用”(好处) :

- $u_{0i} = w_0^\top x_i + \delta_{0i}$  (选 0 的好处)
- $u_{1i} = w_1^\top x_i + \delta_{1i}$  (选 1 的好处)

$\delta$  是**没建模的影响** (心情、测量误差、遗漏变量...) , 当成随机噪声。

### 2) 只比较差值就够了

决定规则: 谁的效用大就选谁。因此只看差值

$$z_i = u_{1i} - u_{0i} = (w_1 - w_0)^\top x_i + (\delta_{1i} - \delta_{0i}).$$

把  $w = w_1 - w_0$ 、 $\varepsilon_i = \delta_{1i} - \delta_{0i}$ , 得到**潜在分数模型**:

$$z_i = w^\top x_i + \varepsilon_i, \quad y_i = \mathbf{1}[z_i \geq 0].$$

### 3) 概率从哪里来?

真正看到的是 0/1, 而不是  $z$ 。所以

$$p(y = 1 | x) = p(z \geq 0) = p(\varepsilon \geq -w^\top x).$$

现在关键在噪声分布  $\varepsilon$ 。

- 若  $\varepsilon \sim N(0, 1)$  (高斯, 对称) :
$$p(y = 1 | x) = \Phi(w^\top x) \rightarrow \text{probit}.$$
- 若  $\varepsilon$  服从 logistic 分布 (来源:  $\delta$  取 Gumbel, 差值就是 logistic) :
$$p(y = 1 | x) = \sigma(w^\top x) \rightarrow \text{logistic}.$$

直觉: 这是“**线性信号 + 随机扰动**, 超过阈值就成功”的模型。选噪声类型, 就等于选概率曲线的形状 ( $\Phi$  或  $\sigma$ ), 两者很像。

### 4) 一个数字小例子

设  $w^\top x = 1$ 。

- Probit:  $p = \Phi(1) \approx 0.8413$
- Logistic:  $p = \sigma(1) \approx 0.731$

形状接近, 但尾部略有差异 (probit 的尾更薄一点)。

### 5) 为什么方差常写成 1?

如果  $\varepsilon \sim N(0, \sigma^2)$ , 把  $w$  同时除以  $\sigma$  就能等价地写回标准差 1 (阈值也可一起缩放)。

$\Rightarrow$  误差方差与  $w$  的尺度不“可识别”, 习惯直接定为 1。

### 6) 这个潜变量视角有什么用?

- **统一理解:** probit/logit 都是 RUM (随机效用模型) 的特例, 只是噪声分布不同。
- **好算** (尤其是 probit) :
  - 给定  $w$ 、已知  $y$  时, 潜在  $z$  是**截断高斯**, 易采样;
  - 给定所有  $z$  时, 对  $w$  就是**线性高斯模型**。

$\Rightarrow$  贝叶斯推断/Gibbs 采样/数据增强都很顺手 (教材后文会讲)。

## 要点总结

- 9.4.2 讲的是 probit / logistic 的潜变量 (随机效用) 解释。
- **核心公式:**  $z = w^\top x + \varepsilon$ ,  $y = \mathbf{1}[z \geq 0]$ 。
- **噪声决定链接:**

- $\varepsilon \sim N(0, 1) \Rightarrow p = \Phi(w^\top x)$  (**probit**) ;
- $\varepsilon$  为 logistic  $\Rightarrow p = \sigma(w^\top x)$  (**logistic**) 。
- **尺度约定**: 噪声方差可被  $w$  吸收, 通常设为 1。
- **实用意义**: 提供统一视觉与更方便的推断 (截断高斯 / Gibbs / 数据增强) 。

把它当作: **分数 + 噪声 + 过线就算 1**, 就掌握了 9.4.2 的全部精华。

翻译并解释以下内容, 翻译不能有遗漏, 解释请通俗易懂且详尽, 因为我的数学基础不太好, 最后总结要点。

#### 9.4.3 Ordinal probit regression

One advantage of the latent variable interpretation of probit regression is that it is easy to extend to the case where the response variable is ordinal, that is, it can take on  $C$  discrete values which can be ordered in some way, such as low, medium and high. This is called ordinal regression. The basic idea is as follows. We introduce  $C + 1$  thresholds  $\gamma_j$  and set

$y_i = j$  if  $\gamma_{j-1} < z_i \leq \gamma_j$  (9.106) where  $\gamma_0 \leq \dots \leq \gamma_C$ . For identifiability reasons, we set  $\gamma_0 = -\infty$ ,  $\gamma_1 = 0$  and  $\gamma_C = \infty$ . For example, if  $C = 2$ , this reduces to the standard binary probit model, whereby  $z_i < 0$  produces  $y_i = 0$  and  $z_i \geq 0$  produces  $y_i = 1$ . If  $C = 3$ , we partition the real line into 3 intervals:  $(-\infty, 0]$ ,  $(0, \gamma_2]$ ,  $(\gamma_2, \infty)$ . We can vary the parameter  $\gamma_2$  to ensure the right relative amount of probability mass falls in each interval, so as to match the empirical frequencies of each class label. Finding the MLEs for this model is a bit trickier than for binary probit regression, since we need to optimize for  $w$  and  $\gamma$ , and the latter must obey an ordering constraint. See e.g., (Kawakatsu and Largey 2009) for an approach based on EM. It is also possible to derive a simple Gibbs sampling algorithm for this model (see e.g., (Hoff 2009, p216)).

#### 9.4.4 Multinomial probit models

Now consider the case where the response variable can take on  $C$  unordered categorical values,  $y_i \in \{1, \dots, C\}$ . The multinomial probit model is defined as follows:

$$z_{ic} = w^\top x_{ic} + i_c \quad (9.107) \sim N(0, R) \quad (9.108)$$

$$y_i = \text{argmax } c \ z_{ic} \quad (9.109)$$

See e.g., (Dow and Endersby 2004; Scott 2009; Fruhwirth-Schnatter and Fruhwirth 2010) for more details on the model and its connection to multinomial logistic regression. (By defining  $w = [w_1, \dots, w_C]$ , and  $x_{ic} = [0, \dots, 0, x_i, 0, \dots, 0]$ , we can recover the more familiar formulation  $z_{ic} = x^T w + \epsilon_{ic}$ .) Since only relative utilities matter, we constrain  $R$  to be a correlation matrix. If instead of setting  $y_i = \text{argmax}_c z_{ic}$  we use  $y_{ic} = I(z_{ic} > 0)$ , we get a model known as multivariate probit, which is one way to model  $C$  correlated binary outcomes (see e.g., (Talhouk et al. 2011)).

已思考 31s

下面先给出**逐字完整翻译**（不遗漏原式与编号），再做**通俗且详尽的解释**，最后给出**要点总结**。

---

## 逐字完整翻译

### 9.4.3 有序 probit 回归 (Ordinal probit regression)

probit 回归的潜变量解释有一个优势：它很容易扩展到**有序响应**的情形，也就是响应变量可以取  $C$  个离散值，并且这些值**有顺序**（例如 low、medium、high）。这称为**有序回归**。基本思路如下。我们引入  $C + 1$  个阈值  $\gamma_j$ ，并规定

$$y_i = j \quad \text{当且仅当} \quad \gamma_{j-1} < z_i \leq \gamma_j \quad (9.106)$$

其中  $\gamma_0 \leq \dots \leq \gamma_C$ 。出于**可识别性**的原因，我们设  $\gamma_0 = -\infty$ ,  $\gamma_1 = 0$ ,  $\gamma_C = +\infty$ 。例如，若  $C = 2$ ，就退化为标准的**二元 probit**:  $z_i < 0$  得  $y_i = 0$ ,  $z_i \geq 0$  得  $y_i = 1$ 。若  $C = 3$ ，我们把实数轴分成三个区间:  $(-\infty, 0]$ ,  $(0, \gamma_2]$ ,  $(\gamma_2, \infty)$ 。我们可以通过调节参数  $\gamma_2$ ，让落在每个区间的**概率质量比例**合适，从而匹配各类别标签的经验频率。

与二元 probit 相比，寻找该模型的 MLE 会稍微棘手一些，因为需要同时对  $w$  与  $\gamma$  做优化，而且  $\gamma$  还必须满足**有序约束**。参见如 (Kawakatsu and Largey 2009) 给出的基于 EM 的一种方法。也可以为该模型推导一个简单的 Gibbs 采样算法（见如 (Hoff 2009, p216)）。

### 9.4.4 多项 probit 模型 (Multinomial probit models)

现在考虑响应变量可以取  $C$  个**无序**类别值的情形， $y_i \in \{1, \dots, C\}$ 。**多项 probit 模型**定义如下：

$$z_{ic} = w^T x_{ic} + \epsilon_{ic} \quad (9.107)$$

$$\epsilon_i \sim N(0, R) \quad (9.108)$$

$$y_i = \arg \max_c z_{ic} \quad (9.109)$$

关于该模型及其与**多项逻辑回归**的联系，详情见 (Dow and Endersby 2004; Scott 2009; Frühwirth-Schnatter and Frühwirth 2010)。 (通过定义  $w = [w_1, \dots, w_C]$ , 以及  $x_{ic} = [0, \dots, 0, x_i, 0, \dots, 0]$ , 我们可以得到更熟悉的写法  $z_{ic} = x_i^\top w_c$ ) ) 由于**只相对效用**有意义，我们把  $R$  约束为**相关矩阵**。如果不采用  $y_i = \arg \max_c z_{ic}$ , 而用  $y_{ic} = I(z_{ic} > 0)$ , 就得到所谓的**多元 probit**, 它是刻画  $C$  个**相关的二元结果**的一种方式 (见如 Talhouk et al. 2011) 。

---

## 通俗解释 (一步步来)

### 9.4.3 有序 probit: 用“多个阈值”切分潜在分数

- 还记得二元 probit 的潜变量故事吗？有一个**隐藏分数**  $z_i = w^\top x_i + \varepsilon_i$  ( $\varepsilon_i \sim N(0, 1)$ ) 。二分类时只看它是否超过 0。
- 有序多分类** (比如 1星/2星/3星……) 的做法：用一串**阈值**把数轴切成  $C$  段,

$$y_i = j \Leftrightarrow \gamma_{j-1} < z_i \leq \gamma_j.$$

- 可识别性**为什么要定  $\gamma_0 = -\infty$ ,  $\gamma_1 = 0$ ,  $\gamma_C = +\infty$ ？  
因为整体平移/缩放  $z$  会与阈值一起移动而不改变分类结果。固定首尾，并把第一个有限阈值定成 0，就锁住了“原点与尺度”，避免“同一模型多套参数”的歧义。
- 概率长什么样？** (有助理解)  
若  $\eta_i = w^\top x_i$ , 则

$$\Pr(y_i = j | x_i) = \Phi(\gamma_j - \eta_i) - \Phi(\gamma_{j-1} - \eta_i).$$

这就是“落在第  $j$  个区间的高斯概率”。调节阈值  $\gamma$  就能控制各类的概率份额，从而贴合数据里的类别频率。

- 训练难点：**要同时学  $w$  与有序的  $\gamma$ 。常见技巧：
  - EM**: 把潜在  $z_i$  当隐变量；
  - Gibbs**: 交替抽样  $z_i$  (截断高斯)、 $w$  (线性高斯)、 $\gamma$  (受顺序约束的更新)。  
也可用优化法，并通过**重新参数化** (如设  $\gamma_2 = \tilde{\gamma}_2$ ,  $\gamma_3 = \gamma_2 + e^{\delta_3}$ , ...) 自动满足有序约束。

### 9.4.4 多项 probit: 多个“竞争的效用”，最大者获胜

- 现在类别**无序** (车/公交/步行...), 每个类别  $c$  有自己的潜在效用

$$z_{ic} = x_i^\top w_c + \varepsilon_{ic}.$$

把  $\varepsilon_i = (\varepsilon_{i1}, \dots, \varepsilon_{iC})$  看作联合高斯，协方差（更准确：相关）为  $R$ 。

- **选择规则：**谁的效用最大就选谁

$$y_i = \arg \max_c z_{ic}.$$

- **为什么把  $R$  约束成“相关矩阵”而不是任意协方差？**

因为只**相对效用**重要：给所有  $z_{ic}$  加同一个常数或乘一个正数， $\arg \max$  不变  $\Rightarrow$  总体位置和尺度不可识别。把各边际方差定为 1（即相关矩阵）就固定了尺度。

- **与多项逻辑回归 (softmax) 的关系：**

- 如果误差  $\varepsilon_{ic}$  取 Gumbel 分布且独立，就得到**多项 logit (softmax)**。
- 这里取的是**联合高斯**，得到**多项 probit**；允许类别间的**误差相关**（例如天气同时影响“步行”和“骑车”的好坏），这是它的一大优点。

- **多元 probit：**如果把“选最大”换成对每个  $c$  做独立阈值化  $y_{ic} = I(z_{ic} > 0)$ ，就得到**一组相关的二元标签模型**（适合同一对象有多个是/否结果且彼此相关的场景）。

## 要点总结

- **有序 probit (ordinal) :**

- 仍用潜在分数  $z \sim N(\eta, 1)$ ，但用一串阈值  $\gamma$  把数轴切成  $C$  段：  
 $\Pr(y = j) = \Phi(\gamma_j - \eta) - \Phi(\gamma_{j-1} - \eta)$ 。
- 为可识别性固定  $\gamma_0 = -\infty$ ,  $\gamma_1 = 0$ ,  $\gamma_C = +\infty$ 。
- 训练需同时学  $w, \gamma$  且  $\gamma$  有序；常用 EM / Gibbs / 受约束优化。

- **多项 probit (multinomial) :**

- 每类一个效用  $z_{ic} = x_i^\top w_c + \varepsilon_{ic}$ ,  $\varepsilon_i \sim N(0, R)$ 。
- 选择最大效用的类别： $y = \arg \max_c z_{ic}$ ；只相对效用重要  $\Rightarrow$  把  $R$  设为**相关矩阵**以定尺度。
- 与 softmax 的关系：Gumbel 噪声  $\Rightarrow$  multinomial logit；高斯噪声  $\Rightarrow$  multinomial probit（可建模类别间相关）。
- 若改成  $y_{ic} = \mathbf{1}[z_{ic} > 0]$   $\Rightarrow$  **多元 probit**，适合多个相关的二元结果。

- **统一直觉：**都是“线性信号 + 随机噪声”的**潜变量阈值模型**；二元/有序/多项只是阈值的布置与是否取最大的问题。

翻译并解释以下内容，翻译不能有遗漏，解释请通俗易懂且详尽，因为我的数学基础不太好，最后总结要点。

## 9.5 Multi-task learning

Sometimes we want to fit many related classification or regression models. It is often reasonable to assume the input-output mapping is similar across these different models, so we can get better performance by fitting all the parameters at the same time. In machine learning, this setup is often called multi-task learning (Caruana 1998), transfer learning (e.g., (Raina et al. 2005)), or learning to learn (Thrun and Pratt 1997). In statistics, this is usually tackled using hierarchical Bayesian models (Bakker and Heskes 2003), as we discuss below, although there are other possible methods (see e.g., (Chai 2010)).

### 9.5.1 Hierarchical Bayes for multi-task learning

Let  $y_{ij}$  be the response of the  $i$ 'th item in group  $j$ , for  $i = 1: N_j$  and  $j = 1: J$ . For example,  $j$  might index schools,  $i$  might index students within a school, and  $y_{ij}$  might be the test score, as in Section 5.6.2. Or  $j$  might index people, and  $i$  might index purchases, and  $y_{ij}$  might be the identity of the item that was purchased (this is known as discrete choice modeling (Train 2009)). Let  $x_{ij}$  be a feature vector associated with  $y_{ij}$ . The goal is to fit the models  $p(y_{ij}|x_{ij})$  for all  $j$ . Although some groups may have lots of data, there is often a long tail, where the majority of groups have little data. Thus we can't reliably fit each model separately, but we don't want to use the same model for all groups. As a compromise, we can fit a separate model for each group, but encourage the model parameters to be similar across groups. More precisely, suppose  $E[y_{ij}|x_{ij}] = g(x_{ij}^\top \beta_j)$ , where  $g$  is the link function for the GLM. Furthermore, suppose  $\beta_j \sim N(\beta^*, \sigma^2_j)$ , and that  $\beta^* \sim N(\mu, \sigma^2_\beta)$ . In this model, groups with small sample size borrow statistical strength from the groups with larger sample size, because the  $\beta_j$ 's are correlated via the latent common parents  $\beta^*$  (see Section 5.5 for further discussion of this point). The term  $\sigma^2_j$  controls how much group  $j$  depends on the common parents and the  $\sigma^2_\beta$  term controls the strength of the overall prior. Suppose, for simplicity, that  $\mu = 0$ , and that  $\sigma^2_j$  and  $\sigma^2_\beta$  are all known (e.g., they could be set by

cross validation). The overall log probability has the form  
 $\log p(D|\beta) + \log p(\beta) = j \log p(D_j|\beta_j) - \|\beta_j - \beta^*\|_2^2 2\sigma_j^2 - \|\beta^*\|_2^2 2\sigma^2 * (9.110)$

We can perform MAP estimation of  $\beta = (\beta_1, \dots, \beta_J)$  using standard gradient methods. Alternatively, we can perform an iterative optimization scheme, alternating between optimizing the  $\beta_j$  and the  $\beta^*$ ; since the likelihood and prior are convex, this is guaranteed to converge to the global optimum. Note that once the models are trained, we can discard  $\beta^*$ , and use each model separately.

### 9.5.2 Application to personalized email spam filtering

An interesting application of multi-task learning is personalized spam filtering. Suppose we want to fit one classifier per user,  $\beta_j$ . Since most users do not label their email as spam or not, it will be hard to estimate these models independently. So we will let the  $\beta_j$  have a common prior  $\beta^*$ , representing the parameters of a generic user.

In this case, we can emulate the behavior of the above model with a simple trick (Daume 2007b; Attenberg et al. 2009; Weinberger et al. 2009): we make two copies of each feature  $x_i$ , one concatenated with the user id, and one not. The effect will be to learn a predictor of the form  
 $E[y_i|x_i, u] = (\beta^*, w_1, \dots, w_J)^T [x_i, I(u=1)x_i, \dots, I(u=J)x_i] (9.111)$

where  $u$  is the user id. In other words,

$$E[y_i|x_i, u=j] = (\beta^* + w_j)x_i (9.112)$$

Thus  $\beta^*$  will be estimated from everyone's email, whereas  $w_j$  will just be estimated from user  $j$ 's email. To see the correspondence with the above hierarchical Bayesian model, define  $w_j = \beta_j - \beta^*$ . Then the log probability of the original model can be rewritten as

$$j \log p(D_j|\beta^* + w_j) - \|w_j\|_2^2 2\sigma_j^2 - \|\beta^*\|_2^2 2\sigma^2 * (9.113)$$

If we assume  $\sigma_j^2 = \sigma^2$ , the effect is the same as using the augmented feature trick, with the same regularizer strength for both  $w_j$  and  $\beta^*$ . However, one typically gets better performance by not requiring that  $\sigma_j^2$  be equal to  $\sigma^2$  (Finkel and Manning 2009).

### 9.5.3 Application to domain adaptation

Domain adaptation is the problem of training a set of classifiers on data drawn from different distributions, such as email and newswire text. This problem is obviously a special case of multi-task learning, where the tasks are the same. (Finkel and Manning 2009) used the above

hierarchical Bayesian model to perform domain adaptation for two NLP tasks, namely named entity recognition and parsing. They report reasonably large improvements over fitting separate models to each dataset, and small improvements over the approach of pooling all the data and fitting a single model.

#### 9.5.4 Other kinds of prior

In multi-task learning, it is common to assume that the prior is Gaussian. However, sometimes other priors are more suitable. For example, consider the task of conjoint analysis, which requires figuring out which features of a product customers like best. This can be modelled using the same hierarchical Bayesian setup as above, but where we use a sparsity-promoting prior on  $\beta_j$ , rather than a Gaussian prior. This is called multi-task feature selection. See e.g., (Lenk et al. 1996; Argyriou et al. 2008) for some possible approaches. It is not always reasonable to assume that all tasks are all equally similar. If we pool the parameters across tasks that are qualitatively different, the performance will be worse than not using pooling, because the inductive bias of our prior is wrong. Indeed, it has been found experimentally that sometimes multi-task learning does worse than solving each task separately (this is called negative transfer).

One way around this problem is to use a more flexible prior, such as a mixture of Gaussians. Such flexible priors can provide robustness against prior mis-specification. See e.g., (Xue et al. 2007; Jacob et al. 2008) for details.

One can of course combine mixtures with sparsity-promoting priors (Ji et al. 2009). Many other variants are possible.

已思考 20s

下面先给出**逐字完整翻译**（保留原式与编号、不遗漏），再做**通俗且详尽的解释**（一步步、配直觉与小例子），最后给出**要点总结**。

## 逐字完整翻译

### 9.5 多任务学习 (Multi-task learning)

有时我们希望拟合许多**相关的**分类或回归模型。常见且合理的假设是：这些不同模型的输入→输出映射彼此相似，因此把所有参数**一起**拟合能获得更好的性能。在机器学习中，这

样的设定常被称为**多任务学习** (Caruana 1998)、**迁移学习** (例如 Raina et al. 2005)，或**学习去学习** (Thrun and Pratt 1997)。在统计学中，这通常用**层次贝叶斯模型** (Bakker and Heskes 2003) 来处理 (如下所述)，尽管也有其它方法 (见 Chai 2010)。

### 9.5.1 用层次贝叶斯做多任务学习

令  $y_{ij}$  表示第  $j$  组中第  $i$  个样本的响应， $i = 1:N_j$ ,  $j = 1:J$ 。例如， $j$  可以索引学校， $i$  可以索引某学校内的学生，而  $y_{ij}$  可能是考试分数 (见 5.6.2 节)。或者， $j$  可以索引人， $i$  可以索引购买行为， $y_{ij}$  可以是被购买物品的身份 (这称为**离散选择建模**，Train 2009)。令  $x_{ij}$  是与  $y_{ij}$  关联的特征向量。目标是为所有  $j$  拟合模型  $p(y_{ij} | x_{ij})$ 。尽管某些组可能有大量数据，但常常存在**长尾**：多数组的数据很少。因此我们既不能可靠地为每个组**单独拟合模型**，也不想对所有组使用**同一个模型**。折中做法是：为每组拟合一个**独立模型**，但**鼓励各组的模型参数彼此相似**。更精确地，假设

$$E[y_{ij} | x_{ij}] = g(x_{ij}^\top \beta_j),$$

其中  $g$  是该 GLM 的链接函数。进一步假设

$$\beta_j \sim N(\beta^{**}, \sigma_j^2 I), \quad \beta^{**} \sim N(\mu, \sigma_{**}^2 I).$$

在该模型中，小样本组会**从大样本组借力**，因为各组的  $\beta_j$  通过潜在的公共父结点  $\beta^{**}$  相关 (见 5.5 节)。 $\sigma_j^2$  控制第  $j$  组**依赖公共父结点的强度**； $\sigma_{**}^2$  则控制整体先验的强度。为简单起见，令  $\mu = 0$ ，并假设  $\sigma_j^2$  与  $\sigma_{**}^2$  都已知 (例如用交叉验证设定)。整体的对数概率为

$$\log p(D | \beta) + \log p(\beta) = \sum_j [\log p(D_j | \beta_j) - \frac{\|\beta_j - \beta^{**}\|^2}{2\sigma_j^2}] - \frac{\|\beta^{**}\|^2}{2\sigma_{**}^2} \quad (9.110)$$

我们可以用标准的梯度方法对  $\beta = (\beta_{1:J}, \beta^{**})$  做 MAP 估计。或者，采用**交替优化**：在  $\beta_j$  与  $\beta^{**}$  之间交替优化；由于似然与先验都是凸的，保证收敛到全局最优。注意：训练完成后可以丢弃  $\beta^{**}$ ，分别使用各组模型。

### 9.5.2 个性化邮件垃圾过滤的应用

多任务学习的一个有趣应用是**个性化垃圾邮件过滤**。假设我们为每个用户拟合一个分类器  $\beta_j$ 。由于多数用户并不会标注“垃圾/非垃圾”，独立估计这些模型会很难。因此令  $\beta_j$  共享一个共同先验  $\beta^{**}$ ，代表**通用用户的参数**。

在这种情况下，可以用一个**简单技巧**来模拟上述模型 (Daumé 2007b; Attenberg et al. 2009; Weinberger et al. 2009)：把每个特征  $x_i$ **复制两份**——一份与用户 ID 连接，一份不连接。其效果是学习如下形式的预测器

$$E[y_i | x_i, u] = (\beta^{**}, w_1, \dots, w_J)^\top [x_i, I(u=1)x_i, \dots, I(u=J)x_i] \quad (9.111)$$

其中  $u$  是用户 ID。也就是说，

$$E[y_i | x_i, u = j] = (\beta^{*\top} + w_j^\top) x_i \quad (9.112)$$

因此  $\beta^{*}$  用所有人的邮件来估计，而  $w_j$  只用用户  $j$  的邮件来估计。为了与上面的层次贝叶斯模型对应，设  $w_j = \beta_j - \beta^{*}$ 。则原模型的对数概率可改写为

$$\sum_j \log p(D_j | \beta^{*} + w_j) - \frac{\|w_j\|^2}{2\sigma_j^2} - \frac{\|\beta^{*}\|^2}{2\sigma_{*}^2} \quad (9.113)$$

若假设  $\sigma_j^2 = \sigma_{*}^2$ ，其效果等价于使用**特征扩增技巧**，并对  $w_j$  与  $\beta^{*}$  使用同样强度的正则化。不过，通常**不要强行令**  $\sigma_j^2 = \sigma_{*}^2$  会得到更好性能 (Finkel and Manning 2009)。

### 9.5.3 领域自适应的应用

**领域自适应 (domain adaptation)** 问题是：在来自不同分布的数据上训练一组分类器，例如电子邮件与新闻文本。这显然是多任务学习的**特例**（任务相同）。Finkel 与 Manning (2009) 用上述层次贝叶斯模型在两个 NLP 任务（命名实体识别与句法分析）上做了领域自适应：相较于为每个数据集单独拟合模型，他们报告了显著提升；相较于把所有数据**简单汇合后拟合单一模型**，也有小幅提升。

### 9.5.4 其它类型的先验

在多任务学习中，常假设先验为**高斯**。但有时其它先验更合适。比如**联合分析 (conjoint analysis)** 任务，需要判断客户最喜欢产品的哪些特征。它也可用上述层次贝叶斯框架来建模，但把  $\beta_j$  的先验换成**促稀疏**的先验，而不是高斯——这称为**多任务特征选择**。可参考 Lenk et al. 1996; Argyriou et al. 2008 等方法。

并非总是合理地假设**所有任务同样相似**。如果把**本质不同的**任务的参数强行“合并”，性能可能比不合并还差，因为先验的归纳偏差错了。事实上，实验证明多任务学习有时会比单任务更差（称为**负迁移**）。

解决这一问题的一条路是使用**更灵活的先验**，如**高斯混合**。这种灵活先验对先验错误更**稳健**。详见 Xue et al. 2007; Jacob et al. 2008。也可以把**混合与稀疏先验结合** (Ji et al. 2009)。还有许多其它变种可能。

## 通俗且详尽的解释

### 1) 为什么需要多任务学习？

- 你有很多相似的小问题（任务）：不同学校的成绩预测、不同用户的垃圾邮件过滤、不同领域的文本分类……
- 每个小问题的数据量**参差不齐**，多数很少。单独训练会**过拟合**；全体混在一起又会**丢掉差异**。

- 层次贝叶斯给你一个折中：每个组有自己的参数  $\beta_j$ ，又通过一个“公共父参数”  $\beta^{(*)}$  把它们温柔地拉到一起（称为收缩/shrinkage或借力/borrow strength）。

直觉图：

单独训练：各自乱跑；

完全合并：绑得太紧；

层次贝叶斯：松紧可调 ( $\sigma_j^2, \sigma_{(*)}^2$  控制)。

## 2) $\sigma_j^2$ 与 $\sigma_{(*)}^2$ 的作用

- $\sigma_j^2$  小  $\Rightarrow$  第  $j$  组的  $\beta_j$  被强力拉向  $\beta^{(*)}$ （数据少的组通常设得更小，防过拟合）。
- $\sigma_{(*)}^2$  小  $\Rightarrow$  公共父参数  $\beta^{(*)}$  本身也被更强正则，避免它被少数大组“带偏”。

## 3) 怎么训练？

- MAP：最大化“各组对数似然之和 – 正则项”（式 9.110），可用梯度法或交替优化（先固定  $\beta^{(*)}$  调  $\beta_j$ ，再固定  $\beta_j$  调  $\beta^{(*)}$ ）。该目标是凸的，因此有全局极值。
- 训练完，可只保留  $\beta_j$ ，在组内单独使用； $\beta^{(*)}$  像一个“训练时的牵引点”。

## 4) 个性化垃圾邮件：特征扩增为什么等价？

- 把通用权重记为  $\beta^{(*)}$ ，用户偏差记为  $w_j$ 。预测是

$$(\beta^{(*)} + w_j)^\top x.$$

- 现在把特征扩成  $[x, I(u=1)x, \dots, I(u=J)x]$ ，权重扩成  $(\beta^{(*)}, w_1, \dots, w_J)$ 。这自动实现了“通用部分 + 个性化微调”。
- 若对  $\beta^{(*)}$  与各  $w_j$  用同强度的  $l_2$  正则，就等价于  $\sigma_j^2 = \sigma_{(*)}^2$  的层次贝叶斯（式 9.113）。实务上往往分开调强度更好（不同用户的数据量差很多）。

## 5) 领域自适应：为何比“合并训练”略好？

- 不同领域（邮件 vs 新闻）分布不同，但也有共性。层次贝叶斯让你既共享（通过  $\beta^{(*)}$ ），又保留差异（各  $\beta_j$ ）。因此通常优于“各练各的”，也略优于“胡乱混在一起”。

## 6) 为何要考虑非高斯先验与负迁移？

- 有些任务的真实参数是稀疏的（只少数特征重要），用稀疏先验（如 Laplace、组 Lasso、层次稀疏）更合适——这叫多任务特征选择。
- 并非所有任务都“同一类”。若把本质不同的任务硬绑定，会出现负迁移（学得更差）。
- 办法：用更灵活的先验（如高斯混合）让模型自己把任务分簇；或把“混合 + 稀疏”结合，提升稳健性。

## 要点总结

- 动机**: 多任务=很多相似的小问题、数据稀疏；层次贝叶斯通过共享父参数  $\beta^{(*)}$  实现借力与收缩。
- 模型**:  $\beta_j \sim N(\beta^{(*)}, \sigma_j^2 I)$ ,  $\beta^{(*)} \sim N(0, \sigma_{(*)}^2 I)$ ; 目标如式 (9.110)。
- 训练**: MAP/梯度/交替优化；训练后可丢弃  $\beta^{(*)}$ , 各组单独用  $\beta_j$ 。
- 技巧**: 特征扩增可等价实现“通用 + 个性化”；正则强度对应  $\sigma_j^2, \sigma_{(*)}^2$ 。
- 应用**: 个性化垃圾过滤、领域自适应（通常优于各自训练，也优于简单合并）。
- 先验选择**: 可用稀疏先验做多任务特征选择；注意负迁移；用混合先验（及其与稀疏结合）提升稳健性。
- 一句话**: 多任务学习 = “每个任务有自己的模型，但都围着一个共享的‘中心’学习”，以此在共享与个性化之间取得最佳平衡。

请总结一下层次贝叶斯

已思考 27s

下面把\*\*层次贝叶斯 (Hierarchical Bayes, HB) \*\*用最抓重点、通俗的方式总结给你（结合我们前面讲到的多任务/分组场景）。

## 它是什么

- 一句话: 在多个相似任务/群体之间“共享信息、又保留差异”的贝叶斯建模方式。
- 做法: 给每个组 (或任务) 各自一个参数  $\beta_j$ , 再让这些  $\beta_j$  围绕一个共享的“父参数”  $\beta^{(*)}$  摆动:

$$\beta_j \sim N(\beta^{(*)}, \sigma_j^2 I), \quad \beta^{(*)} \sim N(0, \sigma_{(*)}^2 I).$$

这叫**层次结构**: 数据层 → 组参数层 → 超参数层。

## 为什么要用

- 小数据组“借力”**: 大多数组样本少，单独训练容易过拟合；HB 让小组“向大组/整体经验靠拢”。
- 比“一锅炖”更细致**: 把所有数据混在一起会丢掉组间差异；HB 既共享、又保留个性。

- 自动“部分合并 (partial pooling) ”：效果介于“完全独立 (不合并) ”与“完全合并”之间，通常更稳。
- 

## 它怎么工作 (直觉)

- 关键是两个方差：
    - $\sigma_j^2$  小  $\Rightarrow$  第  $j$  组的  $\beta_j$  被更强拉向  $\beta^{(*)}$  (更“借力”)。
    - $\sigma_{(*)}^2$  小  $\Rightarrow \beta^{(*)}$  本身被更强正则，不被少数大组“带偏”。
  - 结果：**收缩 (shrinkage)** ——每个  $\beta_j$  被“温和地”拉向整体水平  $\beta^{(*)}$ ，拉多少由数据量与方差控制。
- 

## 怎么估计

- MAP (类正则化) : 最大化

$$\sum_j \log p(D_j | \beta_j) - \frac{\|\beta_j - \beta^{(*)}\|^2}{2\sigma_j^2} - \frac{\|\beta^{(*)}\|^2}{2\sigma_{(*)}^2}$$

可用梯度/牛顿/交替优化；思路上和加  $L_2$  正则很像。

- 全贝叶斯：用 MCMC (或变分) 同时对  $\beta_j, \beta^{(*)}$ , 甚至  $\sigma$  做后验推断 (不止给点估计，还给不确定性)。
  - 超参数设定： $\sigma_j^2, \sigma_{(*)}^2$  可用交叉验证、经验贝叶斯 (最大边际似然) 或再加超先验 (如半柯西) 自动学习。
- 

## 什么时候特别好用

- 多任务学习：每个任务数据少但相似 (个性化垃圾邮件过滤、不同学校成绩预测、不同门店销量预测)。
  - 领域自适应：同一任务、不同域 (邮件 vs 新闻)，既共享又分域。
  - 个性化建模： $\beta_j = \beta^{(*)} + w_j$  (通用+个性化偏置)；还可用“特征扩增”在普通正则化框架中近似实现。
- 

## 变体与注意事项

- 稀疏先验 (代替高斯)  $\rightarrow$  多任务特征选择：适合“少数特征真有用”的场景。

- **任务不全相似会导致负迁移**: 强行把不相干的组拉在一起会变差。
    - 对策: 用**更灵活先验** (如**高斯混合**, 让任务自动“分簇”), 或“混合 + 稀疏”的组合, 提升稳健性。
  - **可识别性与标准化**: 常把输入/响应做标准化, 便于方差解释与收敛。
- 

## 小范式 (落地配方)

1. 先搭 GLM/回归的似然  $p(D_j \mid \beta_j)$ 。
  2. 放层次先验:  $\beta_j \sim N(\beta^{(j)}, \sigma_j^2 I)$ ,  $\beta^{(j)} \sim N(0, \sigma_{(j)}^2 I)$ 。
  3. 选估计法: 先用 MAP (快)、需要时再上 MCMC/VI (全贝)。
  4. 调  $\sigma_j^2, \sigma_{(j)}^2$ : 交叉验证/经验贝叶斯/超先验。
  5. 训练后, 可只保留各组  $\beta_j$  做预测;  $\beta^{(j)}$  是“训练时的锚”。
- 

## 要点速记

- **定义**: 参数分两层 (组层围着共享中心) → 共享信息 + 保留差异。
- **效果**: 自动**部分合并/收缩**, 小数据组向整体靠拢, 稳健抗过拟合。
- **估计**: MAP≈带  $L_2$  正则; 全贝叶斯给不确定性; 超参数控制共享强度。
- **风险**: 任务真不相似会**负迁移**; 用混合/稀疏等更灵活先验缓解。
- **应用**: 多任务、个性化、领域自适应、离散选择、门店/地区分组建模等。

**一句话**: 层次贝叶斯 = “每组有自己的模型, 但都被一个共享的‘家长’温柔牵着手”, 既不各自为战, 也不一刀切。



翻译并解释以下内容, 翻译不能有遗漏, 解释请通俗易懂且详尽, 因为我的数学基础不太好, 最后总结要点。

Figure 9.2 (a) Directed graphical model for generalized

linear mixed effects model with  $J$  groups. (b) Spinal bone mineral density vs age for four different ethnic groups. Raw data is shown in the light gray lines. Fitted model shown in black (solid is the posterior predicted mean, dotted is the posterior predictive variance). From Figure 9 of (Wand 2009). Used with kind permission of Matt Wand

## 9.6 Generalized linear mixed models

Suppose we generalize the multi-task learning scenario to allow the response to include information at the group level,  $x_j$ , as well as at the item level,  $x_{ij}$ . Similarly, we can allow the parameters to vary across groups,  $\beta_j$ , or to be tied across groups,  $\alpha$ . This gives rise to the following model:

$$E[y_{ij}|x_{ij}, x_j] = g \varphi_1(x_{ij})^T \beta_j + \varphi_2(x_j)^T \beta_j + \varphi_3(x_{ij})^T \alpha + \varphi_4(x_j)^T \alpha \quad (9.114)$$

where the  $\varphi_k$  are basis functions. This model can be represented pictorially as shown in Figure 9.2(a). (Such figures will be explained in Chapter 10.) Note that the number of  $\beta_j$  parameters grows with the number of groups, whereas the size of  $\alpha$  is fixed. Frequentists call the terms  $\beta_j$  random effects, since they vary randomly across groups, but they call  $\alpha$  a fixed effect, since it is viewed as a fixed but unknown constant. A model with both fixed and random effects is called a mixed model. If  $p(y|x)$  is a GLM, the overall model is called a generalized linear mixed effects model or GLMM. Such models are widely used in statistics.

**9.6.1 Example: semi-parametric GLMMs for medical data**  
Consider the following example from (Wand 2009).

Suppose  $y_{ij}$  is the amount of spinal bone mineral density (SBMD) for person  $j$  at measurement  $i$ . Let  $x_{ij}$  be the age of person, and let  $x_j$  be their ethnicity, which can be one of: White, Asian, Black, or Hispanic. The primary goal is to determine if there are significant differences in the mean SBMD among the four ethnic groups, after accounting for age. The data is shown in the light gray lines in Figure 9.2(b). We see that there is a nonlinear effect of SBMD vs age, so we will use a semi-parametric model which combines linear regression with non-parametric regression (Ruppert et al. 2003). We also see that there is variation across individuals within each group, so we will use a mixed effects model. Specifically, we will use

$\varphi_1(x_{ij})=1$  to account for the random effect of each person;  $\varphi_2(x_{ij})=0$  since no other coefficients are person-specific;  $\varphi_3(x_{ij})=[bk(x_{ij})]$ , where  $b_k$  is the  $k$ 'th spline basis functions (see Section 15.4.6.2), to account for the nonlinear effect of age; and  $\varphi_4(x_j)=[I(x_j = w), I(x_j = a), I(x_j = b), I(x_j = h)]$  to account for the effect of the different ethnicities.

Furthermore, we use a linear link function. The overall model is therefore

$$\begin{aligned} E[y_{ij}|x_{ij}, x_j] &= \beta_j + \alpha T_b(x_{ij}) + i_j \quad (9.115) \\ &+ \alpha_w I(x_j = w) + \alpha_a I(x_j = a) + \alpha_b I(x_j = b) + \alpha_h I(x_j = h) \end{aligned} \quad (9.116)$$

where  $i_j \sim N(0, \sigma^2_y)$ .  $\alpha$  contains the non-parametric part of the model related to age,  $\alpha$  contains the parametric part of the model related to ethnicity, and  $\beta_j$  is a random offset for person  $j$ . We endow all of these regression coefficients with separate Gaussian priors. We can then perform posterior inference to compute  $p(\alpha, \alpha_w, \alpha_a, \alpha_b, \alpha_h | D)$  (see Section 9.6.2 for computational details). After fitting the model, we can compute the prediction for each group. See Figure 9.2(b) for the results. We can also perform significance testing, by computing  $p(\alpha_g - \alpha_w | D)$  for each ethnic group  $g$  relative to some baseline (say, White), as we did in Section 5.2.3.

### 9.6.2 Computational issues

The principle problem with GLMMs is that they can be difficult to fit, for two reasons. First,  $p(y_{ij}|\theta)$  may not be conjugate to the prior  $p(\theta)$  where  $\theta = (\alpha, \beta)$ . Second, there are two levels of unknowns in the model, namely the regression coefficients  $\theta$  and the means and variances of the priors  $\eta = (\mu, \sigma)$ . One approach is to adopt fully Bayesian inference methods, such as variational Bayes (Hall et al. 2011) or MCMC (Gelman and Hill 2007). We discuss VB in Section 21.5, and MCMC in Section 24.1. An alternative approach is to use empirical Bayes, which we discuss in general terms in Section 5.6. In the context of a GLMM, we can use the EM algorithm (Section 11.4), where in the E step we compute  $p(\theta|\eta, D)$ , and in the M step we optimize  $\eta$ . If the linear regression setting, the E step can be performed exactly, but in general we need to use approximations. Traditional methods use numerical quadrature or Monte Carlo (see e.g., (Breslow and Clayton 1993)). A faster approach is to use variational EM; see (Braun and McAuliffe 2010) for an application of

variational EM to a multi-level discrete choice modeling problem. In frequentist statistics, there is a popular method for fitting GLMMs called generalized estimating equations or GEE (Hardin and Hilbe 2003). However, we do not recommend this approach, since it is not as statistically efficient as likelihood-based methods (see Section 6.4.3). In addition, it can only provide estimates of the population parameters  $\alpha$ , but not the random effects  $\beta_j$ , which are sometimes of interest in themselves.

已思考 35s

下面先给出逐字完整翻译（保留原式与编号、不遗漏），随后给出通俗且详尽的解释（把图与公式讲清楚），最后用要点清单总结。

## 逐字完整翻译

### 图 9.2

- (a) 具有  $J$  个组的广义线性混合效应模型 (GLMM) 的有向图模型。
- (b) 四个不同族裔的脊柱骨矿物质密度 (SBMD) 随年龄的关系。浅灰色线是原始数据。黑色曲线为拟合模型（实线是后验预测均值，虚线是后验预测方差）。摘自 (Wand, 2009) 的图 9，并经 Matt Wand 许可使用。

## 9.6 广义线性混合模型 (Generalized linear mixed models)

设想把多任务学习场景推广，使响应同时包含组层面的信息  $x_j$  和个体层面的信息  $x_{ij}$ 。同样，我们允许参数在组间变化  $\beta_j$ ，或在组间共享  $\alpha$ 。这得到如下模型：

$$E[y_{ij} | x_{ij}, x_j] = g^{-1}(\phi_1(x_{ij})^\top \beta_j + \phi_2(x_j)^\top \beta_j + \phi_3(x_{ij})^\top \alpha + \phi_4(x_j)^\top \alpha) \quad (9.114)$$

其中  $\phi_k$  为基函数。该模型可如图 9.2(a) 所示地图示化（第 10 章会解释这类图）。注意： $\beta_j$  的参数个数随组数增加而增加，而  $\alpha$  的维度是固定的。频率学派把  $\beta_j$  称为**随机效应**（在组间随机变化），把  $\alpha$  称为**固定效应**（视作一个固定但未知的常量）。同时含固定效应与随机效应的模型称为**混合模型**。若  $p(y | x)$  是一个 GLM，整体模型称为**广义线性混合效应模型**或 GLMM。这类模型在统计学中被广泛使用。

### 9.6.1 例子：医学数据的半参数 GLMM

看一个来自 (Wand, 2009) 的例子。设  $y_{ij}$  是第  $j$  个人在第  $i$  次测量的 SBMD；令  $x_{ij}$  为年

龄，令  $x_j$  为其族裔，取值为 White、Asian、Black 或 Hispanic。主要目标：在考虑年龄影响后，判断四个族裔的平均 SBMD 是否有显著差异。图 9.2(b) 的浅灰色线给出了数据。可见 SBMD 随年龄呈**非线性效应**，因此我们采用把**线性回归与非参数回归结合的半参数模型**（Ruppert et al., 2003）。也能看到同一族裔内部的个体间存在差异，因此使用**混合效应模型**。具体地，

- 取  $\phi_1(x_{ij}) = 1$  用于刻画每个人的**随机效应**；
- 取  $\phi_2(x_{ij}) = 0$  (没有其他“个人特有”的系数)；
- 取  $\phi_3(x_{ij}) = [b_k(x_{ij})]$ ，其中  $b_k$  为样条的第  $k$  个基函数（见 15.4.6.2 节），用于表示年龄的**非线性效应**；
- 取  $\phi_4(x_j) = [\mathbf{I}(x_j = w), \mathbf{I}(x_j = a), \mathbf{I}(x_j = b), \mathbf{I}(x_j = h)]$  用于表示不同族裔的效应。

此外使用**线性链接函数**。于是整体模型为

$$\mathbb{E}[y_{ij} \mid x_{ij}, x_j] = \beta_j + \alpha^\top b(x_{ij}) + \varepsilon_{ij} \quad (9.115)$$

$$+ \alpha_w \mathbf{I}(x_j = w) + \alpha_a \mathbf{I}(x_j = a) + \alpha_b \mathbf{I}(x_j = b) + \alpha_h \mathbf{I}(x_j = h) \quad (9.116)$$

其中  $\varepsilon_{ij} \sim N(0, \sigma_y^2)$ 。向量  $\alpha$  包含与**年龄**有关的**非参数部分**； $(\alpha_w, \alpha_a, \alpha_b, \alpha_h)$  包含与**族裔**有关的**参数化部分**； $\beta_j$  是第  $j$  个人的随机偏置。我们给所有这些回归系数各自施加**高斯先验**。随后进行后验推断以计算  $p(\alpha, \alpha_{\text{eth}}, \beta, \sigma^2 \mid D)$ （计算细节见 9.6.2）。模型拟合后即可对各族裔做预测，结果见图 9.2(b)。也可像 5.2.3 节那样做显著性检验，例如对每个族裔  $g$ ，计算相对某个基线（如 White）的  $p(\alpha_g - \alpha_w \mid D)$ 。

## 9.6.2 计算问题

GLMM 的主要难点在于**拟合不易**，原因有二：

- (1)  $p(y_{ij} \mid \theta)$  可能**不与先验  $p(\theta)$  共轭**，其中  $\theta = (\alpha, \beta)$ ；
- (2) 模型中有两层未知量：回归系数  $\theta$  以及先验的均值、方差  $\eta = (\mu, \sigma)$ 。

一种做法是采用**完全贝叶斯方法**，如**变分贝叶斯**（Hall et al., 2011）或 MCMC (Gelman and Hill, 2007)（第 21.5 与 24.1 节讨论）。另一种是**经验贝叶斯**（5.6 节），在 GLMM 中可用**EM 算法**（11.4 节）：E 步计算  $p(\theta \mid \eta, D)$ ，M 步优化  $\eta$ 。在线性回归情形 E 步可精确计算，但一般需近似。传统方法用**数值积分或蒙特卡罗**（Breslow and Clayton, 1993）。更快的是**变分 EM**；例如 Braun and McAuliffe (2010) 将其用于多层次离散选择问题。

在频率学派中，有一种流行的 GLMM 拟合方法称为**广义估计方程**（GEE；Hardin and Hilbe, 2003）。但我们**不推荐**该方法，因为与基于似然的方法相比，它的统计效率较低（见 6.4.3 节）；此外它只能给出总体参数  $\alpha$  的估计，不能给出有时本身就很重要的随机效应  $\beta_j$ 。

# 通俗解释（把图和式子讲人话）

## 1) GLMM 是什么？

- 仍是 GLM（回归/分类），但把**两类效应**混在一起：
  - **固定效应**  $\alpha$ : 所有组共享的一套系数（例如“年龄对 SBMD 的平均影响”“各族裔的平均差异”）。
  - **随机效应**  $\beta_j$ : 每个组/个人独有的偏置或系数（例如“第  $j$  个人整体偏高/偏低一点”）。
- 这样既能**共享信息**（靠  $\alpha$ ），又能**个性化**（靠  $\beta_j$ ）——对应图 9.2(a) 里灰点（观测  $y_{ij}$ ）同时受到  $x_{ij}$ 、 $x_j$ 、 $\beta_j$ 、 $\alpha$  的影响。

## 2) 式 (9.114) 怎么读？

- 先把输入通过若干“基函数” $\phi_k(\cdot)$  做变换（能容纳非线性、分段样条、类别指示等），再线性组合出**线性预测子**；最后用  $g^{-1}$ （链接函数的反函数）把它变成**期望/概率**。
- $\phi_1, \phi_2$  搭配**随机效应**  $\beta_j$ ,  $\phi_3, \phi_4$  搭配**固定效应**  $\alpha$ 。

## 3) 医学例子里各项的意思

- $\beta_j$ : 每个人的**随机截距**（谁天生略高/略低）。
- $b(x_{ij})$ : 样条基（多段平滑曲线），用于表示**年龄—SBMD的非线性关系**；其系数由  $\alpha$  给出（固定效应，大家共享）。
- $\alpha_w, \alpha_a, \alpha_b, \alpha_h$ : 四个族裔的**固定效应**（相对基线的平均差异）。
- $\varepsilon_{ij}$ : 测量噪声（均值 0、方差  $\sigma_y^2$ ）。
- 图 9.2(b) 的实线/虚线是模型对“不同族裔”的预测均值/不确定性区间；浅灰是每个人的原始轨迹，能看出个体差异被随机效应吸纳，整体趋势由固定效应（样条+族裔）决定。

小提醒：式 (9.115) 中把  $\varepsilon_{ij}$  写在  $E[\cdot]$  右边只是书写习惯；从严格意义讲， $E[y_{ij} | \cdot] = \beta_j + \alpha^\top b(x_{ij}) + \text{族裔项}$ ,  $\varepsilon_{ij}$  的期望为 0。

## 4) 为什么 GLMM 拟合难？

- **不共轭**: GLM 的似然与我们给  $(\alpha, \beta)$  设的先验往往不是“共轭好算”的搭配。
- **两层未知**: 既要估  $\alpha, \beta$ （一大堆），又要估先验的  $\mu, \sigma$ （超参数）。
- **解决路子**：
  - **全贝叶斯**: MCMC（精确但慢）或变分贝叶斯（快但近似）。

- **经验贝叶斯 + EM**: 把  $\theta = (\alpha, \beta)$  当隐变量, E 步算它的后验, M 步调  $\eta = (\mu, \sigma)$ ; 线性高斯时 E 步可闭式, 其余要近似 (数值积分/蒙特卡罗/变分 EM)。
- **GEE (频率学派)**: 实现方便, 但只能估总体参数  $\alpha$ , 统计效率不如似然法, 也不给  $\beta_j$  的估计——所以本文不推荐。

## 要点总结

- **GLMM = GLM + (固定效应  $\alpha$ ) + (随机效应  $\beta_j$ )**: 既共享规律又保留个体/组差异。
- **模型形态**: 用基函数把  $x_{ij}, x_j$  变换后线性组合, 再过链接函数 (式 9.114)。 $\beta_j$  随组数增长,  $\alpha$  维度固定。
- **例子**: SBMD~年龄 (非线性样条) +族裔 (指示变量) +个人随机截距。拟合后能画出各族裔随年龄的平均曲线与不确定性 (图 9.2b), 并做族裔差异的显著性检验。
- **拟合方法**: 全贝叶斯 (MCMC/VB)、经验贝叶斯的 EM、或近似 (数值积分/蒙特卡罗/变分 EM); GEE 较低效且不给随机效应。
- **直觉**: 固定效应 = 大家共同规律; 随机效应 = 每个组/人自己的偏差; 把两者合在一起, 就能稳健又个性化地建模复杂数据。

翻译并解释以下内容, 翻译不能有遗漏, 解释请通俗易懂且详尽, 因为我的数学基础不太好, 最后总结要点。

### 9.7 Learning to rank

In this section, we discuss the learning to rank or LETOR problem. That is, we want to learn a function that can rank order a set of items (we will be more precise below). The most common application is to information retrieval. Specifically, suppose we have a query  $q$  and a set of documents  $d_1, \dots, d_m$  that might be relevant to  $q$  (e.g., all documents that contain the string  $q$ ). We would like to sort these documents in decreasing order of relevance and show the top  $k$  to the user. Similar problems arise in other areas, such as collaborative filtering. (Ranking players in a game or tournament setting is a slightly different kind of problem; see Section 22.5.5.) Below we summarize some methods for solving this problem, following the presentation of (Liu 2009). This material is not based on GLMs, but we include it in this chapter

anyway for lack of a better place. A standard way to measure the relevance of a document  $d$  to a query  $q$  is to use a probabilistic language model based on a bag of words model. That is, we define  $\text{sim}(q, d) = p(q|d) = \sum_{i=1}^n p(q_i|d)$ , where  $q_i$  is the  $i$ 'th word or term, and  $p(q_i|d)$  is a multinoulli distribution estimated from document  $d$ . In practice, we need to smooth the estimated distribution, for example by using a Dirichlet prior, representing the overall frequency of each word. This can be estimated from all documents in the system. More precisely, we can use

$$p(t|d) = (1 - \lambda) \text{TF}(t, d) / \text{LEN}(d) + \lambda p(t|\text{background}) \quad (9.117)$$

where  $\text{TF}(t, d)$  is the frequency of term  $t$  in document  $d$ ,  $\text{LEN}(d)$  is the number of words in  $d$ , and  $0 < \lambda < 1$  is a smoothing parameter (see e.g., Zhai and Lafferty (2004) for details). However, there might be many other signals that we can use to measure relevance. For example, the PageRank of a web document is a measure of its authoritativeness, derived from the web's link structure (see Section 17.2.4 for details). We can also compute how often and where the query occurs in the document. Below we discuss how to learn how to combine all these signals.

### 9.7.1 The pointwise approach

Suppose we collect some training data representing the relevance of a set of documents for each query. Specifically, for each query  $q$ , suppose that we retrieve  $m$  possibly relevant documents  $d_j$  for  $j = 1: m$ . For each query-document pair, we define a feature vector,  $x(q, d)$ . For example, this might contain the query-document similarity score and the page rank score of the document. Furthermore, suppose we have a set of labels  $y_j$  representing the degree of relevance of document  $d_j$  to query  $q$ . Such labels might be binary (e.g., relevant or irrelevant), or they may represent a degree of relevance (e.g., very relevant, somewhat relevant, irrelevant). Such labels can be obtained from query logs, by thresholding the number of times a document was clicked on for a given query. If we have binary relevance labels, we can solve the problem using a standard binary classification scheme to estimate,  $p(y = 1|x(q, d))$ . If we have ordered relevancy labels, we can use ordinal regression to predict the rating,  $p(y = r|x(q, d))$ . In either case, we can then sort the documents by this scoring metric. This is called the

pointwise approach to LETOR, and is widely used because of its simplicity. However, this method does not take into account the location of each document in the list. Thus it penalizes errors at the end of the list just as much as errors at the beginning, which is often not the desired behavior. In addition, each decision about relevance is made very myopically.

### 9.7.2 The pairwise approach

There is evidence (e.g., (Carterette et al. 2008)) that people are better at judging the relative relevance of two items rather than absolute relevance. Consequently, the data might tell us that  $d_j$  is more relevant than  $d_k$  for a given query, or vice versa. We can model this kind of data using a binary classifier of the form  $p(y_{jk}|x(q, d_j), x(q, d_k))$ , where we set  $y_{jk} = 1$  if  $\text{rel}(d_j, q) > \text{rel}(d_k, q)$  and  $y_{jk} = 0$  otherwise. One way to model such a function is as follows:

$$p(y_{jk} = 1|x_j, x_k) = \text{sigm}(f(x_j) - f(x_k)) \quad (9.118)$$

where  $f(x)$  is a scoring function, often taken to be linear,  $f(x) = w^T x$ . This is a special kind of neural network known as RankNet (Burges et al. 2005) (see Section 16.5 for a general discussion of neural networks). We can find the MLE of  $w$  by maximizing the log likelihood, or equivalently, by minimizing the cross entropy loss, given by

$$L = N \sum_{i=1}^N \sum_{j=1}^{m_i} \sum_{k=j+1}^{m_i} L_{ijk} \quad (9.119)$$

$$-L_{ijk} = I(y_{ijk} = 1) \log p(y_{ijk} = 1|x_{ij}, x_{ik}, w) + I(y_{ijk} = 0) \log p(y_{ijk} = 0|x_{ij}, x_{ik}, w) \quad (9.120)$$

This can be optimized using gradient descent. A variant of RankNet is used by Microsoft's Bing search engine.

已思考 22s

下面先给出**逐字完整翻译**（保留公式与编号、不遗漏），随后用**通俗且详尽的解释**把概念与推导讲清楚，最后给出**要点总结**。

## 逐字完整翻译

### 9.7 排序学习 (Learning to rank)

本节我们讨论**排序学习** (learning to rank, 简称 LETOR) 问题。也就是，我们要学习一个函数，能把一组项目按**相关性排序** (下面会更精确地表述)。最常见的应用是**信息检索**。具体地，设有一个查询  $q$  和一组可能与  $q$  相关的文档  $d_1, \dots, d_m$  (例如，所有包含字符串  $q$  的文档)。我们希望按**相关性从高到低**对这些文档排序，并把前  $k$  个展示给用户。类

似的问题在其他领域也会出现，例如协同过滤。（在游戏或锦标赛环境中对选手进行排名是稍有不同的问题；见 22.5.5 节。）下面我们按 (Liu, 2009) 的介绍，概述一些解决该问题的方法。本节内容并不建立在 GLM 之上，但由于缺乏更合适的章节，我们仍把它放在本章。

一种**标准的衡量**文档  $d$  对查询  $q$  相关性的方式，是使用基于**词袋模型的概率语言模型**。也就是，定义

$$\text{sim}(q, d) = p(q \mid d) = \prod_{i=1}^n p(q_i \mid d),$$

其中  $q_i$  是第  $i$  个词/项，而  $p(q_i \mid d)$  是从文档  $d$  估计出的一个多项分布。在实践中，我们需要**平滑**这个估计分布，例如使用 **Dirichlet 先验**，它表示每个词在整体上的频率，这可以从系统中的所有文档估计。更精确地，我们可以使用

$$p(t \mid d) = (1 - \lambda) \frac{\text{TF}(t, d)}{\text{LEN}(d)} + \lambda p(t \mid \text{background}) \quad (9.117)$$

其中  $\text{TF}(t, d)$  是词项  $t$  在文档  $d$  中出现的次数， $\text{LEN}(d)$  是  $d$  的词数，且  $0 < \lambda < 1$  是一个平滑参数（细节见 Zhai & Lafferty, 2004）。不过，衡量相关性还可以使用许多**其他信号**。例如，网页的 **PageRank** 衡量它的“权威度”，由万维网的链接结构推导而来（细节见 17.2.4 节）。我们也可以计算查询在文档中**出现的频率与位置**。下面讨论如何**学习**把这些信号组合起来。

### 9.7.1 点式 (pointwise) 方法

设我们为每个查询收集了一些训练数据，表示该查询下一组文档的相关性。具体地，对每个查询  $q$ ，设我们取回  $m$  个可能相关的文档  $d_j$  ( $j = 1:m$ )。对每个“查询–文档”二元组，我们定义一个**特征向量**  $x(q, d)$ 。例如，它可以包含“查询–文档相似度分数”和该文档的 PageRank 分数。此外，假设我们还有一组标签  $y_j$ ，表示文档  $d_j$  对查询  $q$  的**相关程度**。这类标签可以是**二元**（如相关/不相关），也可以是**多级**（如非常相关/一般相关/不相关）。这类标签可以从查询点击日志得到，例如对“某查询下该文档的点击次数”设置阈值后得到。

若相关性标签是二元的，我们可以用**标准二分类**来估计  $p(y = 1 \mid x(q, d))$ 。若相关性标签是**有序等级**，可以用**有序回归**来预测评分  $p(y = r \mid x(q, d))$ 。在这两种情况下，我们都可用该评分对文档排序。这称为 LETOR 的**点式方法**，因其**简单**而被广泛使用。然而，该方法**没有考虑文档在列表中的位置**：它把列表末尾的错误与开头的错误惩罚得**同样多**，而这常常不是我们想要的。此外，它对每个文档是否相关的判断常常是**短视且独立的**。

### 9.7.2 对式 (pairwise) 方法

有证据（如 Carterette et al., 2008）表明，人们更擅长判断**两个项目的相对相关性**，而不

是各自的**绝对相关性**。因此，数据可能告诉我们：对给定查询， $d_j$  比  $d_k$  更相关，或相反。我们可以用如下形式的**二分类器**来建模这类数据： $p(y_{jk} \mid x(q, d_j), x(q, d_k))$ ，其中若  $\text{rel}(d_j, q) > \text{rel}(d_k, q)$  则设  $y_{jk} = 1$ ，否则设  $y_{jk} = 0$ 。对此类函数的一种建模方式是：

$$p(y_{jk} = 1 \mid x_j, x_k) = \text{sigm}(f(x_j) - f(x_k)) \quad (9.118)$$

其中  $f(x)$  是一个**打分函数**，常取为**线性**形式  $f(x) = w^\top x$ 。这是一种被称为 RankNet 的特殊神经网络 (Burges et al., 2005；通用的神经网络讨论见 16.5 节)。我们可以通过**最大化对数似然**（或等价地**最小化交叉熵损失**）来求  $w$  的 MLE，其损失为

$$L = \sum_{i=1}^N \sum_{j=1}^{m_i} \sum_{k=j+1}^{m_i} L_{ijk} \quad (9.119)$$

$$-L_{ijk} = I(y_{ijk} = 1) \log p(y_{ijk} = 1 \mid x_{ij}, x_{ik}, w) + I(y_{ijk} = 0) \log p(y_{ijk} = 0 \mid x_{ij}, x_{ik}, w)$$

该目标可用**梯度下降**优化。微软的 Bing 搜索引擎使用了 RankNet 的一种变体。

## 通俗且详尽的解释

### 1) LETOR 是啥？为什么不是“普通分类”？

- 目标不是“文档相关/不相关”，而是**对一组文档排序**，把前  $k$  个给用户。
- 你要整合很多**信号**：文本匹配、PageRank、是否在标题出现、锚文本、点击率、发布时间……
- 训练时，标注可以来自**人工打分**或**点击日志**（点击次数高/早位点击等）。

### 2) 语言模型打分 (式 9.117) 在干嘛？

- 直觉：如果文档  $d$  真“谈论”查询  $q$ ，那从“以  $d$  为生成器的语言模型”里生成  $q$  的概率  $p(q \mid d)$  就应高。
- $p(q \mid d) = \prod_i p(q_i \mid d)$ ：把查询词独立相乘（词袋）。
- **平滑**：文档里没出现的词不能给概率 0，于是用  $(1 - \lambda) \times \text{“文档内频率”} + \lambda \times \text{“背景词频”}$ （全库统计）。
- 这是一个**简单且强力的基线**，但远不够（忽略位置、结构、外链等），所以要**学习融合多信号**。

### 3) 点式 (pointwise) 方法：把排序变成“逐条打分”

- 思路：每个（查询，文档）样本一个特征向量  $x(q, d)$ 。
  - 标签若是**二元**  $\Rightarrow$  训练二分类器，输出  $p(\text{相关} \mid x)$ 。
  - 标签若是**多级/有序**  $\Rightarrow$  训练有序回归，输出等级概率。

- 排序时：按预测分数从高到低排。
- **优点：**实现简单、可直接复用分类模型（LR、SVM、GBDT、NN 等）。
- **缺点：**
  - **不看位置：**把排在第 100 名的错误与第 1 名的错误同等对待；而用户只看前几名。
  - **短视：**每个文档独立判断，不直接考虑“相互之间谁应在谁前面”。

#### 4) 对式 (pairwise) 方法：学“谁应排在谁前面”

- 思路：把排序转化为**成对比较**。对每个查询，把文档两两配对，形成训练样本  $(x_j, x_k, y_{jk})$ ；  
 $y_{jk} = 1$  表示  $d_j$  应该“战胜”  $d_k$ 。
- **RankNet 公式 (9.118) :**  
 $p(y_{jk} = 1) = \sigma(f(x_j) - f(x_k))$ 。
  - 如果  $f(x_j)$  远大于  $f(x_k)$ ，概率接近 1；反之接近 0。
  - 线性  $f(x) = w^\top x$  就得到一个**可解释的线性排序器**；也可用深网做非线性  $f$ 。
- **优化：**用交叉熵 (9.119–9.120) 做梯度下降。
- **优点：**更贴合“相对顺序”的目标；对“把更相关的往上提”更敏感。
- **注意：**一个查询有  $m$  篇文档，成对样本是  $O(m^2)$ ，需要**采样或挖难例** (hard-negative) 来控算力。

##### 简单对比

- **Pointwise：**学“这个文档相关分数是多少？”
- **Pairwise：**学“A 应该排在 B 前面吗？”  
 Pairwise 往往比 Pointwise 更符合排序评估（如 NDCG/Top-k），因此工业界很常用（例如 RankNet 家族）。

## 要点总结

- **任务：**给定查询  $q$  与候选文档  $d_1, \dots, d_m$ ，学习一个打分/排序函数，输出前  $k$ 。
- **文本相关性的基线：**语言模型  $p(q | d) = \prod p(q_i | d)$ ，用式 (9.117) 做平滑（文档内频率与背景词频插值）。
- **点式方法：**把排序化为逐样本打分（二分类/有序回归），**简单但不看位置、决策短视**。
- **对式方法 (RankNet) :** 学**成对偏好**，  
 $p(y_{jk} = 1) = \sigma(f(x_j) - f(x_k))$ ，用交叉熵训练；更符合“谁排谁前”的目标，工业界常

用。

- **工程提示：**对式的样本量随  $m^2$  增长，需做**负例采样/难例挖掘**；点式上手快、强基  
线，常作为出发点。

一句话：**排序学习**的核心是在“文本匹配、权威度、位置、行为”等多信号上~~学一个排序器~~；点式简单、对式更贴合“相对顺序”。

翻译并解释以下内容，翻译不能有遗漏，解释请通俗易懂且详尽，因为我的数学基础不太好，最后总结要点。

9.7.3 The listwise approach The pairwise approach suffers from the problem that decisions about relevance are made just based on a pair of items (documents), rather than considering the full context. We now consider methods that look at the entire list of items at the same time. We can define a total order on a list by specifying a permutation of its indices,  $\pi$ . To model our uncertainty about  $\pi$ , we can use the Plackett-Luce distribution, which derives its name from independent work by (Plackett 1975) and (Luce 1959). This has the following form:

$$p(\pi|s) = \prod_{j=1}^m s_j \prod_{u=j+1}^m s_u \quad (9.121)$$

where  $s_j = s(\pi^{-1}(j))$  is the score of the document ranked at the  $j$ 'th position. To understand Equation 9.121, let us consider a simple example. Suppose  $\pi = (A, B, C)$ . Then we have that  $p(\pi)$  is the probability of A being ranked first, times the probability of B being ranked second given that A is ranked first, times the probability of C being ranked third given that A and B are ranked first and second. In other words,

$$p(\pi|s) = s_A s_B + s_C \times s_B s_C + s_C \times s_A s_C \quad (9.122)$$

To incorporate features, we can define  $s(d) = f(x(q, d))$ , where we often take  $f$  to be a linear function,  $f(x) = w^T x$ . This is known as the ListNet model (Cao et al. 2007). To train this model, let  $y_i$  be the relevance scores of the documents for query  $i$ . We then minimize the cross entropy term

$$-\sum_i p(\pi|y_i) \log p(\pi|s_i) \quad (9.123)$$

Of course, as stated, this is intractable, since the  $i$ 'th term needs to sum over  $m!$  permutations. To make this

tractable, we can consider permutations over the top k positions only:

$$p(\pi_{1:k}|s_{1:m}) = \prod_{j=1}^k s_j / \sum_{u=1}^m s_u \quad (9.124)$$

There are only  $m!/(m - k)!$  such permutations. If we set k = 1, we can evaluate each cross entropy term (and its derivative) in O(m) time. In the special case where only one document from the presented list is deemed relevant, say  $y_i = c$ , we can instead use multinomial logistic regression:

$$p(y_i = c|x) = \frac{\exp(s_c)}{\sum_{c=1}^C \exp(s_c)} \quad (9.125)$$

This often performs at least as well as ranking methods, at least in the context of collaborative filtering (Yang et al. 2011).

已思考 14s

下面先给出**逐字完整翻译**（保留公式与编号，不遗漏），随后用**通俗且详尽的解释**把概念讲清楚，最后给出**要点总结**。

## 逐字完整翻译

### 9.7.3 列表式 (listwise) 方法

对式 (pairwise) 方法的问题在于：它关于相关性的决策只基于**一对项目**（文档），而不是**考虑完整的上下文**。现在我们讨论**同时查看整个项目列表**的方法。我们可以通过为索引指定一个**排列**  $\pi$  来对列表定义一个**全序**。为了刻画我们对  $\pi$  的不确定性，可以使**用 Plackett–Luce 分布**，其名称来自 (Plackett, 1975) 和 (Luce, 1959) 的独立工作。其形式为

$$p(\pi | s) = \prod_{j=1}^m \frac{s_j}{\sum_{u=j}^m s_u} \quad (9.121)$$

其中  $s_j = s(\pi^{-1}(j))$  是排在第  $j$  位的文档的**分数**。为了理解式 (9.121)，考虑一个简单例子：设  $\pi = (A, B, C)$ 。那么  $p(\pi)$  等于“**A 排第一**”的概率，乘以“在 A 已经排第一的条件下 **B 排第二**”的概率，再乘以“在 A、B 分别排一、二的条件下 **C 排第三**”的概率。也就是

$$p(\pi | s) = \frac{s_A}{s_A + s_B + s_C} \times \frac{s_B}{s_B + s_C} \times \frac{s_C}{s_C} \quad (9.122)$$

为了纳入特征，我们可以定义  $s(d) = f(x(q, d))$ ，其中  $f$  常取线性函数  $f(x) = w^\top x$ 。这称为 **ListNet 模型** (Cao et al., 2007)。训练该模型时，令  $y_i$  为查询  $i$  的各文档的**相关性分数**。然后最小化**交叉熵项**

$$-\sum_i \sum_{\pi} p(\pi | y_i) \log p(\pi | s_i) \quad (9.123)$$

当然，按字面这样做是不可 tractable (难以计算) 的，因为第  $i$  项需要对  $m_i!$  个排列求和。为了让计算可行，我们只考虑前  $k$  位的排列：

$$p(\pi_{1:k} | s_{1:m}) = \prod_{j=1}^k \frac{s_j}{\sum_{u=1}^m s_u} \quad (9.124)$$

这样的排列数只有  $m!/(m-k)!$ 。如果设  $k=1$ ，则每个交叉熵项（及其导数）可以在  $O(m)$  时间内计算。一个特殊情形是：在呈现的列表里只有一个文档被认为是相关的，记为  $y_i = c$ 。这时可以改用多项逻辑回归：

$$p(y_i = c | x) = \frac{\exp(s_c)}{\sum_{c'=1}^m \exp(s_{c'})} \quad (9.125)$$

在协同过滤的背景下，这常常至少与排序方法表现相当 (Yang et al., 2011)。

---

## 通俗且详尽的解释

### 1) 为什么要“列表式”而不是“对式”？

- 对式只关心成对的胜负 (A 比 B 相关吗?)，没有直接关注“整个前几名的整体质量”。
- 列表式直接把整个候选集一起考虑，更贴近最终目标 (如 NDCG、MAP、Top-k 点击等都看一条完整列表)。

### 2) Plackett–Luce (PL) 分布直觉

- 把“排一个列表”的过程看成不放回的连续抽取：  
先从全部候选里按分数  $s$  的比例抽出“第 1 名”；  
再在剩下的里按比例抽出“第 2 名”；……
- 所以式 (9.121) 就是连乘：每一步的“被选中概率 = 自己分数 / 当前池子分数和”。
- 例子 ( $A, B, C$ )：  
先从  $\{A, B, C\}$  里选第 1:  $\frac{s_A}{s_A + s_B + s_C}$ ；  
再从  $\{B, C\}$  里选第 2:  $\frac{s_B}{s_B + s_C}$ ；  
最后只剩  $C$ ，概率 1；连乘得到式 (9.122)。

### 3) 怎么把“特征”变成“分数”？

- 用一个打分函数  $s(d) = f(x(q, d))$ 。
  - 线性： $f(x) = w^\top x$  (ListNet 原始形式)；

- 也可以是深度网络等更复杂的 $f$ 。

#### 4) 训练目标 (式 9.123) 在做什么?

- $p(\pi | y_i)$ : 用**标注的相关性分数**  $y_i$  构造“**理想列表分布**”(分数高的排列概率更大)。
- $p(\pi | s_i)$ : 用模型分数  $s_i = f(x)$  得到的“**预测列表分布**”。
- **交叉熵最小化**: 让预测分布尽量贴近理想分布  $\Rightarrow$  学会把高相关性文档排在前面。
- 难点: 对**所有排列求和要  $m!$  项**, 计算爆炸。

#### 5) 如何把计算压下来?

- 只对**前  $k$  位建模** (式 9.124)。这与实际目标吻合——用户通常只看**前几条**。
- 复杂度: 排列数从  $m!$  降到  $m!/(m - k)!$ 。
- **极端简化  $k = 1$** : 只关心“谁是第 1 名”, 每个查询的损失/梯度都能在  $O(m)$  算完, 相当高效。

#### 6) 只要选出“唯一一个相关”的特殊情形

- 如果每次展示的列表里**只有 1 个**是相关 (比如“点击了哪一条”), 训练就退化为**多类分类**: 对候选文档做 softmax (式 9.125)。
- 在协同过滤里, 这个做法常与复杂的排序训练**不相上下**, 实现也更简单。

### 要点总结

- **列表式 (listwise) 排序**一次性考虑整列, 更贴合排序指标与用户体验。
- 用 Plackett–Luce 把“**排列表**”建模为“按分数比例、不放回的连续抽取”(式 9.121/9.122)。
- **ListNet**: 用  $s(d) = f(x(q, d))$  (常为线性) 建分数; 最小化“**理想列表分布 vs 预测列表分布**”的**交叉熵** (式 9.123)。
- **计算削减**: 只建模Top-k (式 9.124);  $k = 1$  时每个查询  $O(m)$  可解。
- **特殊情形**: 若仅一个文档相关, 问题退化为**多项逻辑回归/softmax** (式 9.125), 常有不错表现。

一句话: **listwise** 方法把“整份候选列表”当成一个整体来学, 借助 PL 分布与交叉熵, 直接优化“把最该在前面的放到前面”。

请对比这三种排序学习方法

下面把 Pointwise / Pairwise / Listwise 三种排序学习方法放在一起对比。先给一张总览表，再补充要点与选型建议。

## 一张表看懂

维度	Pointwise (点式)	Pairwise (对式)	Listwise (列表式)
基本思路	把每个 $(q, d)$ 当作一个样本，学“这个文档有多相关/打几分”。	把同一查询下的文档两两成对，学“谁排在谁前”。	一次看整列候选，直接学“这一列的最优排列分布”。
典型目标/损失	二分类/回归/有序回归（如 logistic/MSE/有序回归损失）。	对差值做分类：如 RankNet 的 $\sigma(f(x_j) - f(x_k))$ ，或 RankSVM 的 hinge。	基于排列分布（如 Plackett-Luce）的交叉熵：ListNet/ListMLE；常做 Top-k 近似。
是否考虑“位置”	否（默认每个文档独立，列表头尾错同等惩罚）。	间接：可给成对样本加 <b>位置权重</b> （如 $\Delta NDCG \rightarrow \text{LambdaRank/LambdaMART}$ ）。	是：天然以 <b>整列</b> 为目标，可直接面向 NDCG/MAP/Top-k。
标签需求	二元或分级相关性，最容易从点击/人工分得来。	只需相对偏好（j 比 k 好），人更容易判断；也可由点击生成偏好对。	需要分级分数或能构造“理想排列”的信号；对标注质量更敏感。
计算/样本规模	最省：样本数≈候选量。	对同一查询是 $O(m^2)$ 对，需 <b>采样/难例挖掘</b> 。	最重：全排列不可行，常做 Top-k 或 k=1 近似；实现更复杂。
与指标的贴合	弱（更像在做分类/回归）。	中（优化“相对次序”，可用 $\Delta NDCG$ 权重贴近指标）。	强（直接对列表建模，天然面向 Top-k/NDCG）。
代表算法	LR/GBDT/NN 的点式打分器。	RankNet、RankSVM、LambdaRank/LambdaMART（工业常用）。	ListNet、ListMLE、Top-k PL/Softmax。

维度	Pointwise (点式)	Pairwise (对式)	Listwise (列表式)
优点	简单、稳健、可校准，冷启动友好。	更关注“把更相关的排前面”；和排序指标更一致；偏好标注容易。	直接优化“整列质量”；最契合用户只看前几条的事实。
缺点	不看位置与相互关系，头部错误惩罚不够。	样本多、训练重；需要好策略采样与权重。	训练复杂、需要较好分级标签；实现与调参成本高。
适用场景	快速可用的强基线；标签简单/数据少。	主流线上排序（如 LambdaMART）；有偏好/点击对。	强调 Top-k 质量、分级标签可靠、候选量适中时。

## 进一步说明与小公式

- Pointwise

训练  $\text{score} = f(x(q,d))$ ，用分类/回归损失；排序时按分数降序。简单，但默认“第 1 名的错”≈“第 100 名的错”。

- Pairwise

学  $P(j>k)=\sigma(f(x_j)-f(x_k))$  (RankNet)，或  $1-\text{hinge}(f(x_j)-f(x_k))$  (RankSVM)。可用  $\Delta NDCG$  给样本加权 (LambdaRank/LambdaMART)，让“影响榜首的对”权重大 → 与指标对齐。

- Listwise

用 Plackett–Luce 给排列建模， $p(\pi|s)=\prod s_{\pi_j}/\sum_{\{\text{剩余}\}} s$  (ListNet)，最小交叉熵；常只对 Top-k 做 (如  $k=1$  时是 softmax over docs,  $O(m)$ )。

## 选型建议 (实用角度)

1. **从 Pointwise 起步**：上手快、可复用分类/回归堆栈，给你一个强基线。

2. **上线主力 → Pairwise (LambdaMART) :**

- 有点击/偏好对；
- 需要对“把更相关的往上提”更敏感；
- 结合  $\Delta NDCG$  权重，效果稳定、工业界常用。

3. **指标/业务强依赖 Top-k 且有可靠分级标签 → Listwise :**

- 候选量适中 (或做 Top-k 近似)；
- 需要最大化 NDCG/MAP 时更合适。

4. **点击偏置/展示位偏置**: 三类方法都应配**倾向性校正** (IPW/Propensity weighting) , 但 Pairwise/Listwise 更容易把“位置权重”揉进目标。
5. **算力与数据**:

- Pairwise 注意  $O(m^2)$  的样本爆炸 → 负例采样、难例采样;
  - Listwise 关注 Top-k 近似与稳定训练;
  - Pointwise 最省算力, 适合快速迭代与冷启动。
- 

## 要点速记

- **点式**: 独立打分, 简单但不看位置。
- **对式**: 学成对偏好, 可用  $\Delta$ NDCG 加权, 工业常用 (LambdaMART) 。
- **列表式**: 整体看整列, 用 PL/Softmax 等直接对齐 Top-k/NDCG, 但训练更重。

一句话: **越往右越贴近真实排序目标、也越复杂**——基线用 Pointwise, 产线多用 Pairwise (Lambda 系列), 追求极致 Top-k 质量且标签好时用 Listwise。



已上传的图片

翻译并解释以下内容, 翻译不能有遗漏, 解释请通俗易懂且详尽, 因为我的数学基础不太好, 最后总结要点。

Table 9.3 Illustration of how to compute NDCG, from [http://en.wikipedia.org/wiki/Discounted\\_cumulative\\_gain](http://en.wikipedia.org/wiki/Discounted_cumulative_gain). The value  $r_i$  is the relevance score of the item in position  $i$ . From this, we see that  $DCG@6 = 3 + (2 + 1.887 + 0 + 0.431 + 0.772) = 8.09$ . The maximum DCG is obtained using the ordering with scores 3, 3, 2, 2, 1, 0. Hence the ideal DCG is 8.693, and so the normalized DCG is  $8.09 / 8.693 = 0.9306$ .

9.7.4 Loss functions for ranking There are a variety of ways to measure the performance of a ranking system, which we summarize below.

- Mean reciprocal rank (MRR). For a query  $q$ , let the rank

position of its first relevant document be denoted by  $r(q)$ .

Then we define the mean reciprocal rank to be  $1/r(q)$ . This is a very simple performance measure.

- Mean average precision (MAP). In the case of binary relevance labels, we can define the precision at  $k$  of some ordering as follows:

$P@k(\pi) = \text{num. relevant documents in the top } k \text{ positions of } \pi$  (9.126)

We then define the average precision as follows:

$$AP(\pi) = \frac{1}{k} \sum_{i=1}^k P@k(\pi) \cdot I_k \text{ num. relevant documents} \quad (9.127)$$

where  $I_k$  is 1 iff document  $k$  is relevant. For example, if we have the relevancy labels  $y = (1, 0, 1, 0, 1)$ , then the AP is  $\frac{1}{3} (1 + 0 + 1) / 3 \approx 0.67$ . Finally, we define the mean average precision as the AP averaged over all queries.

- Normalized discounted cumulative gain (NDCG).

Suppose the relevance labels have multiple levels. We can define the discounted cumulative gain of the first  $k$  items in an ordering as follows:

$$DCG@k(r) = \sum_{i=1}^k r_i \log_2 \left( \frac{1}{r_i} \right) \quad (9.128)$$

where  $r_i$  is the relevance of item  $i$  and the  $\log_2$  term is used to discount items later in the list. Table 9.3 gives a simple numerical example. An alternative definition, that places stronger emphasis on retrieving relevant documents, uses

$$DCG@k(r) = \sum_{i=1}^k \frac{r_i}{\log_2(i+1)} \quad (9.129)$$

The trouble with DCG is that it varies in magnitude just because the length of a returned list may vary. It is therefore common to normalize this measure by the ideal DCG, which is the DCG obtained by using the optimal ordering:  $IDCG@k(r) = \max_{\pi} DCG@k(r)$ . This can be easily computed by sorting  $r_1:m$  and then computing  $DCG@k$ . Finally, we define the normalized discounted cumulative gain or NDCG as  $DCG/IDCG$ . Table 9.3 gives a simple numerical example. The NDCG can be averaged over queries to give a measure of performance.

- Rank correlation. We can measure the correlation between the ranked list,  $\pi$ , and the relevance judgment,  $\pi^*$ , using a variety of methods. One approach, known as the (weighted) Kendall's  $\tau$  statistics, is defined in terms of the weighted pairwise inconsistency between the two lists:
- $$\tau(\pi, \pi^*) = \frac{\sum_{u < v} w_{uv} [1 + \text{sgn}(\pi_u - \pi_v)\text{sgn}(\pi^*_u - \pi^*_v)]}{2 \sum_{u < v} w_{uv}} \quad (9.130)$$

A variety of other measures are commonly used.

These loss functions can be used in different ways. In the Bayesian approach, we first fit the model using posterior inference; this depends on the likelihood and prior, but not the loss. We then choose our actions at test time to minimize the expected future loss. One way to do this is to sample parameters from the posterior,  $\theta_s \sim p(\theta|D)$ , and then evaluate, say, the precision@k for different thresholds, averaging over  $\theta_s$ . See (Zhang et al. 2010) for an example of such an approach. In the frequentist approach, we try to minimize the empirical loss on the training set. The problem is that these loss functions are not differentiable functions of the model parameters. We can either use gradient-free optimization methods, or we can minimize a surrogate loss function instead. Cross entropy loss (i.e., negative log likelihood) is an example of a widely used surrogate loss function. Another loss, known as weighted approximate-rank pairwise or WARP loss, proposed in (Usunier et al. 2009) and extended in (Weston et al. 2010), provides a better approximation to the precision@k loss. WARP is defined as follows:

$$\text{WARP}(f(x, :), y) = L(\text{rank}(f(x, :), y)) \quad (9.131)$$

$$\text{rank}(f(x, :), y) = y = y I(f(x, y) \geq f(x, j)) \quad (9.132)$$

$$L(k) = \sum_{j=1}^k \alpha_j, \text{ with } \alpha_1 \geq \alpha_2 \geq \dots \geq 0 \quad (9.133)$$

Here  $f(x, :) = [f(x, 1), \dots, f(x, |y|)]$  is the vector of scores for each possible output label, or, in IR terms, for each possible document corresponding to input query  $x$ . The expression  $\text{rank}(f(x, :), y)$  measures the rank of the true label  $y$  assigned by this scoring function. Finally,  $L$  transforms the integer rank into a real-valued penalty. Using  $\alpha_1 = 1$  and  $\alpha_j > 1 = 0$  would optimize the proportion of top-ranked correct labels. Setting  $\alpha_1:k$  to be non-zero values would optimize the top  $k$  in the ranked list, which will induce good performance as measured by MAP or precision@k. As it stands, WARP loss is still hard to optimize, but it can be further approximated by Monte Carlo sampling, and then optimized by gradient descent, as described in (Weston et al. 2010).

下面先给出**逐字完整翻译**（含表格含义与所有公式编号，不遗漏），接着做**通俗且详尽的解释**（一步步算 NDCG、讲明各评价指标与损失），最后给出**要点总结**。

## 逐字完整翻译

**表 9.3 说明如何计算 NDCG** (来自 Wikipedia“Discounted cumulative gain”词条)。 $r_i$  是排在第  $i$  个位置的条目的**相关性分数** (relevance)。由此可得  $\text{DCG}@6 = 3 + (2 + 1.887 + 0 + 0.431 + 0.772) = 8.09$ 。把相关性分数按 3,3,2,2,1,0 的顺序排列可得到最大的 DCG。因此理想 DCG (IDCG) 为 8.693，于是归一化 DCG 为  $8.09/8.693 = 0.9306$ 。

---

### 9.7.4 排序的损失函数 (Loss functions for ranking)

衡量排序系统表现的方法有多种，概述如下。

- **平均倒数排名 (MRR)**。对查询  $q$ ，设其第一个相关文档出现在排名位置  $r(q)$ 。则  $\text{MRR} = 1/r(q)$ 。这是非常简单的性能度量。
- **平均准确率 (MAP)**。在二元相关性标签下，我们把某个排序  $\pi$  的“前  $k$  名的准确率”定义为

$$\text{P}@k(\pi) = \frac{\text{前 } k \text{ 个位置里的相关文档数}}{k} \quad (9.126)$$

然后定义**平均准确率 (AP)**：

$$\text{AP}(\pi) = \frac{\sum_k \text{P}@k(\pi) \cdot I_k}{\text{相关文档总数}} \quad (9.127)$$

其中  $I_k = 1$  当且仅当第  $k$  个文档是相关的。例：若相关性标签  $y = (1, 0, 1, 0, 1)$ ，则  $\text{AP} = \frac{1}{3} (\frac{1}{1} + \frac{2}{3} + \frac{3}{5}) \approx 0.76$ 。最后，MAP 是把 AP 在所有查询上取平均。

- **归一化折扣累积增益 (NDCG)**。若相关性有多等级，可将一个排序前  $k$  个的**折扣累积增益**定义为

$$\text{DCG}@k(r) = r_1 + \sum_{i=2}^k \frac{r_i}{\log_2 i} \quad (9.128)$$

其中  $r_i$  是第  $i$  个位置的相关性； $\log_2$  项对**较靠后的**条目给折扣。表 9.3 给出简单数值例子。另一种更强调“把高相关文档找出来”的定义是

$$\text{DCG}@k(r) = \sum_{i=1}^k \frac{2^{r_i} - 1}{\log_2(1 + i)} \quad (9.129)$$

DCG 的问题在于随着返回列表长度不同其量纲会变化。因此通常用**理想 DCG** (采用最优排序得到的 DCG) 来**归一化**：

$\text{IDCG}@k(r) = \max_{\pi} \text{DCG}@k(r)$ 。做法：把  $r_{1:m}$  **降序排序**，再按上式计算

### DCG@k。最终定义

$\text{NDCG} = \text{DCG}/\text{IDCG}$ 。表 9.3 给出简单例子。NDCG 可在查询上取平均得到整体性能。

- **秩相关 (Rank correlation)**。可用多种方法衡量排序列表  $\pi$  与“相关性判断” $\pi^{*}$  的相关性。一种叫\*\* (加权) Kendall's  $\tau^{**}$  的统计量，用两列表之间的成对不一致的加权形式定义：

$$\tau(\pi, \pi^{*}) = \frac{\sum_{u < v} w_{uv} \frac{1 + \text{sgn}(\pi_u - \pi_v) \text{sgn}(\pi_u^{*} - \pi_v^{*})}{2}}{\sum_{u < v} w_{uv}} \quad (9.130)$$

实务中还会用其他多种度量。

这些损失可以有不同用法。**贝叶斯路线**：先用后验推断拟合模型（取决于似然和先验，但与损失无关），然后在测试时选择动作以最小化期望未来损失。一种做法是：从后验采样参数  $\theta_s \sim p(\theta | D)$ ，对不同阈值计算如 precision@k，并在  $\theta_s$  上求平均（例见 Zhang et al., 2010）。

**频率学派路线**：尝试最小化训练集上的**经验损失**。问题是这些损失对模型参数通常**不可导**。可用**无梯度优化**，或最小化一个**替代损失 (surrogate)**。**交叉熵损失**（负对数似然）是广用的替代损失。另一种损失是 WARP (weighted approximate-rank pairwise, Usunier et al., 2009; Weston et al., 2010 扩展)，它对 precision@k 有更好的近似。WARP 定义如下：

$$\text{WARP}(f(x, :), y) = L(\text{rank}(f(x, :), y)) \quad (9.131)$$

$$\text{rank}(f(x, :), y) = \sum_{y' \in y} \mathbf{1}(f(x, y') \geq f(x, y)) \quad (9.132)$$

$$L(k) = \sum_{j=1}^k \alpha_j, \quad \alpha_1 \geq \alpha_2 \geq \dots \geq 0 \quad (9.133)$$

其中  $f(x, :) = [f(x, 1), \dots, f(x, |y|)]$  是各可能输出标签（在检索里是各候选文档）的**打分向量**。 $\text{rank}(\cdot)$  度量模型把**真实标签**  $y$  排在第几名； $L$  把整数名次转成实值惩罚。取  $\alpha_1 = 1$ ,  $\alpha_{j>1} = 0$  会优化“把正确标签放在第一名”的比例；让  $\alpha_{1:k}$  取非零则针对 Top-k，通常会带来 MAP 或 P@k 的好表现。WARP 本身仍然不易直接优化，但可用**蒙特卡罗采样**进一步近似，然后用**梯度下降**训练 (Weston et al., 2010)。

## 通俗解释与手把手例子

### 1) MRR / MAP / NDCG 各看什么？

- MRR：只看第一个相关文档的位置，越靠前越好（ $1/1=1$  最好； $1/10=0.1$  较差）。适合“只要第一个合适答案”的场景（如问答）。
- MAP（二元相关）：平均“每次命中时之前位置的准确率”。能综合考虑所有相关文档的位置，而且越早出现贡献越大。
- NDCG（多等级相关）：对高分文档奖励更大，对靠后的条目对数折扣，并用理想排序做归一化，便于不同查询/列表长度比较。

## 2) 表 9.3：一步步算 NDCG

表中（从左到右）给了：位置  $i = 1..6$ ，对应相关性  $r_i = [3, 2, 3, 0, 1, 2]$ ，折扣  $\log_2 i = [0, 1, 1.59, 2.0, 2.32, 2.59]$ ；于是（按公式 9.128）

- $DCG@6 = r_1 + \frac{r_2}{\log_2 2} + \frac{r_3}{\log_2 3} + \dots$
- 具体为  $3 + 2/1 + 3/1.59 + 0/2.0 + 1/2.32 + 2/2.59$
- 即  $3 + 2 + 1.887 + 0 + 0.431 + 0.772 = 8.09$ 。
- $IDCG$ ：把得分按降序排成  $3, 3, 2, 2, 1, 0$ ，再套同一公式算，得  $8.693$ 。
- $NDCG = 8.09/8.693 \approx 0.9306$ 。

直觉：和理想排序比，当前列表质量达到了 93%。

## 3) 秩相关 (Kendall's $\tau$ ) 直觉

- 看所有成对元素  $(u, v)$ 。如果两份排序  $\pi$  与  $\pi^*$  对这对元素的先后顺序一致，就加分；不一致就不得分（或扣分），还可以用权重  $w_{uv}$  强调重要对（如靠前位置）。 $\tau$  越高说明排序越接近理想。

## 4) 贝叶斯 vs 频率学派：怎么用这些损失？

- 贝叶斯：先按似然+先验做后验推断；预测时用期望风险最小化——例如从后验采样多个参数、对每个计算  $P@k$ ，再求平均来挑模型/阈值。
- 频率学派：直接在训练集最小化损失。但 MRR/MAP/NDCG 对参数不可导  $\Rightarrow$ 
  - 用无梯度方法，或
  - 用可导的替代损失（如交叉熵、WARP）近似真实目标来训练。

## 5) WARP 的直觉训练法（为何接近 $P@k$ ）

- 目标：让正确项的名次尽量小（越靠前越好），而且重点优化Top-k。
- 训练时常用随机负采样：对一个正样本  $y$ ，随机抽负样本  $y'$  直到找到“违例” $f(x, y') \geq f(x, y)$ ，用抽到的次数去近似名次，再按  $L(\text{rank})$  给出梯度更新。这样把注意力放在“真正把正样本挤下去的负样本”上，更贴近  $P@k$ /NDCG 的意图。

# 要点总结

- MRR：只看第一个命中位置；MAP：对每个命中累积**先前准确率**，关注所有相关文档；  
NDCG：多等级、对后位折扣、再与**理想排序**归一化（更可比）。
- Kendall's  $\tau$ ：比较两份排序的一致性，可加权强调重要成对关系。
- **训练两大路线：**
  - **贝叶斯**：先做后验，再最小化期望损失（可用采样估计）。
  - **频率学派**：直接最小化经验损失，但需用**可导替代损失**（交叉熵、WARP）或无梯度方法。
- WARP：用“近似名次 + Top-k 权重”来逼近 P@k/NDCG，配合负采样与梯度下降可高效训练。
- **计算 NDCG**：先算 DCG，再算 IDCG（**理想排序**），最后  $NDCG = DCG / IDCG$ ；表 9.3 的例子得到 0.9306。

基于本对话里我贴过的整章原文，输出：  
①**知识点总结**  
(分点+关键词+用途) ②**子章节关联** (依赖/对比/递进+关系句，附Mermaid可选) ③**本章总结**。

好的！下面基于你贴过的整章原文，给出三块内容：  
①**知识点总结** (分点 + 关键词 + 用途) ②**子章节关联** (依赖/对比/递进 + 关系句，并附可选 Mermaid 图) ③**本章总结**。

## ① 知识点总结 (分点 + 关键词 + 用途)

- **指数族分布 (9.2)**
  - 关键词： $p(x|\theta) = h(x) \exp(\theta^\top \phi(x) - A(\theta))$ 、充分统计量  $\phi(x)$ 、(对数) 配分函数  $A(\theta)$ 、规范参数  $\theta$
  - 核心性质：有限维充分统计量；共轭先验存在；最大熵推导； $A'(\theta) = E[\phi]$ 、 $A''(\theta) = \text{cov}(\phi) \Rightarrow$  凸性
  - 用途：统一刻画 Bernoulli/多项/Gaussian 等；支撑 GLM、变分推断；带来**矩匹配的 MLE**与**共轭贝叶斯**计算
- **配分函数与矩 (9.2.3)**
  - 关键词：均值/方差由  $A(\theta)$  的一二阶导给出、海森矩阵=协方差

- 用途：快速得出期望与方差、证明对数似然的**凸性**（利于全局最优）
- **指数族的极大似然与贝叶斯 (9.2.4–9.2.5)**
  - 关键词：**矩匹配**  $E[\phi] = \frac{1}{N} \sum \phi(x_i)$ ; 共轭先验  $p(\eta) \propto \exp(v_0 \eta^\top \tau_0 - v_0 A(\eta))$ ; 后验“加法”更新；预测分布比值形式
  - 用途：通用、稳定的参数估计与闭式更新（如 Beta–Bernoulli）
- **最大熵视角 (9.2.6)**
  - 关键词：在约束  $\sum p(x)f_k(x) = F_k$  下最大化熵  $\Rightarrow$  指数族 (Gibbs)
  - 用途：给指数族以“**假设最少**”的理论正当性
- **广义线性模型 GLM (9.3)**
  - 关键词： $y|x$  服从指数族；**均值函数**  $g^{-1}$ :  $\mu = g^{-1}(w^\top x)$ ; **链接函数**  $g$  (规范链接  $g = \psi$ )
  - 例子：线性回归（高斯）、逻辑回归 (Bernoulli + logit)、Poisson 回归（计数）
  - 训练：一阶梯度、IRLS / 牛顿、Fisher scoring; MAP=加  $L_2$  正则；贝叶斯 = MCMC/VI
  - 用途：把不同响应类型统一到一套回归框架中
- **Probit 回归与潜变量 (9.4)**
  - 关键词： $\mu = \Phi(w^\top x)$ ; 梯度/海森可解析；RUM 随机效用： $z = w^\top x + \varepsilon, y = [z > 0]; \varepsilon \sim N \Rightarrow$  probit,  $\varepsilon$  由 Gumbel 差  $\Rightarrow$  logistic
  - 扩展：**有序 probit** (多阈值)、**多项/多元 probit** (每类效用、相关误差)
  - 用途：便于贝叶斯采样（截断高斯）、自然建模“阈值决策”
- **多任务学习 & 层次贝叶斯 (9.5)**
  - 关键词： $\beta_j \sim N(\beta^{**}, \sigma_j^2 I), \beta^{**} \sim N(0, \sigma_{**}^2 I)$ ; **部分合并/收缩**；特征扩增 trick；领域自适应；**负迁移**；混合/稀疏先验
  - 用途：在“多组小样本、又相似”时**借力**并保留差异（个性化/跨域）
- **GLMM 混合效应模型 (9.6)**
  - 关键词：**固定效应**  $\alpha$  (共享)、**随机效应**  $\beta_j$  (组特异)；基函数  $\phi_k$ ；半参数样条；计算：EM/变分/MCMC；GEE (不推荐)
  - 用途：同时建模组层与个体层变异（医疗/教育/纵向数据）
- **排序学习 LETOR (9.7)**
  - 关键词：Pointwise/Pairwise/Listwise；语言模型基线 + Dirichlet/插值平滑  $\lambda$ ；RankNet ( $\sigma(f_j - f_k)$ )、ListNet (Plackett–Luce)；评估：MRR、MAP、NDCG、Kendall  $\tau$ ；WARP 近似 P@k

- 用途：搜索/推荐的Top-k 排序，把多信号学成评分与顺序

## ② 子章节关联（依赖 / 对比 / 递进）

### 关系句（按知识脉络）

1. 9.2  $\Rightarrow$  9.3：指数族定义与  $A(\theta)$  的导数提供 GLM 的均值/方差公式与凸性保证，使 IRLS / Fisher scoring 可行。
2. 9.2.4/9.2.5  $\Rightarrow$  9.3.2/9.3.3：指数族的 MLE（矩匹配）与共轭贝叶斯直接用于 GLM 的频率学派与贝叶斯拟合。
3. 9.4  $\perp$  逻辑回归：probit 与 logit 功能相近，但通过 RUM 潜变量把二者统一；在贝叶斯采样上 probit 更友好。
4. 9.4.2  $\Rightarrow$  9.4.3/9.4.4：潜变量视角自然推广到有序/多项选择（多阈值或多效用）。
5. 9.5  $\Rightarrow$  9.6：层次贝叶斯的“共享 + 个性化”思路，在线性/GLM 框架下落地即 GLMM 的固定效应 + 随机效应。
6. 9.6  $\rightarrow$  计算方法：由于非共轭与双层未知，需 EM / VB / MCMC；与 9.2 的共轭情形形成对比。
7. 9.7  $\perp$  GLM：排序学习不一定基于 GLM，但大量模型（RankNet/ListNet）用到逻辑/softmax/PL 等与前文相关的概率建模与损失。
8. 9.7.4  $\rightarrow$  训练策略：真实指标（MAP/NDCG）不可导，因此借助交叉熵/WARP 等替代损失，与 9.3 的对数似然训练形成方法学呼应。

### （可选）Mermaid 总览

```

graph TD
    A[9.2 指数族  
·充分统计量] --> B[9.3 GLM  
·链接/IRLS/Fisher]
    A --> C[9.2.4/5 MLE/共轭Bayes]
    C --> B
    B --> D[9.4 Probit/Logit  
·RUM潜变量]
    D --> E[9.4.3 有序Probit]
    E --> F[9.4.4 多项/多元Probit]
    B --> G[9.5 多任务  
·层次Bayes  
 $\beta_j \sim N(\beta^*, \cdot)$ ]
    G --> H[9.6 GLMM  
·固定+随机效应]
    H -.非共轭/双层未知.-> I[EM / VB / MCMC]
    I -.方法学呼应.-> J[9.7 排序学习  
·Point/Pair>List]
    J --> K[评估:  
MRR/MAP/NDCG/ $\tau$ ]
    J --> L[损失: 交叉熵/WARP]
  
```

## ③ 本章总结

- **统一的概率底座**：指数族把常见分布装进同一范式，带来**有限充分统计、共轭先验、凸对数似然与最大熵等强性质**，是后续建模与推断的基础。
- **从分布到模型**：GLM 用**链接函数**把输入线性组合映到不同类型的响应（实值/二元/计数），在计算上可用**梯度 + 二阶**（IRLS/Fisher），在贝叶斯上可用 MCMC/VI。

- **二分类的两条路**: logit 与 probit 在性能相近, 但 RUM 潜变量给予 probit 更好的贝叶斯可计算性, 并自然扩展到**有序与多项选择**。
- **跨组共享与个性化**: 多任务的**层次贝叶斯**通过“ $\beta^{**}$  + 收缩”在共享与差异间取得平衡; 其在线性/GLM 框架的工程化形态是 GLMM (固定 + 随机效应), 可处理**组层/个体层**的复杂数据结构。
- **排序问题的专章**: 虽然不完全依赖 GLM, 但排名学习通过 Pointwise/Pairwise/Listwise 三路线, 把文本匹配、权威度等信号学成 Top-k 排序; 评估用 MRR/MAP/NDCG/ $\tau$ , 训练常用**交叉熵**或 WARP 近似真实目标。

**一句话**: 本章从指数族的数学根基出发, 建起 GLM 的通用建模与推断框架, 进而处理潜变量选择、多任务/混合效应等结构化问题, 并以排序学习收束到检索/推荐的实战应用——**理论统一、算法可解、应用广泛**。