第二章笔记

小狗

目录

1 引言

在本章中，我们将深入探讨二项式模型，这是统计学中一个非常重要的概念。二项式模型广泛应用于许多实际问题中，特别是在处理只有两种可能结果的情况时。我们将从最基本的伯努利分布开始，逐步深入到更复杂的概念。

2 伯努利分布

2.1 什么是伯努利分布?

伯努利分布是统计学中最简单的离散概率分布之一。它是以瑞士数学家雅各布 · 伯努利（1655-1705）命名的。想象一下抛硬币的情景：你只关心硬币

是正面还是反面，这就是一个典型的伯努利试验。

伯努利分布描述的是只有两种可能结果的随机试验。我们通常用 1 表示” 成功”，0 表示” 失败”。

2.2 伯努利分布的数学定义

假设 Y 是一个随机变量，它只能取两个值：0 或 1。我们用 μ 表示 Y 取值为 1 的概率（即成功的概率）。那么：

$$P(Y = 1) = \mu, \quad P(Y = 0) = 1 - \mu$$

我们将这种分布记作 $Y \sim \text{Bernoulli}(\mu)$，读作”Y 服从参数为 的伯努利分布”。

2.3 概率质量函数

伯努利分布的概率质量函数（PMF）可以写成一个简洁的形式：

$$p(y) = \mu^y (1 - \mu)^{1-y}, \quad y \in \{0, 1\}$$

这个公式看起来可能有点复杂，但它其实很巧妙：- 当 $y = 1$ 时，$p(1) = \mu$ - 当 $y = 0$ 时，$p(0) = 1 - \mu$

2.4 伯努利分布的期望和方差

理解一个分布的关键是知道它的期望（平均值）和方差（离散程度）。

1. 期望：$E(Y) = \mu$

   这很直观，因为 就是成功的概率。

2. 方差：$\mathrm{Var}(Y) = \mu(1-\mu)$

   方差的推导需要一点技巧：

$$
\begin{aligned}
\mathrm{Var}(Y) &= E(Y^2) - [E(Y)]^2 \\
&= E(Y) - [E(Y)]^2 \quad (\text{因为} Y^2 = Y \text{ 当} Y \text{ 只取 } 0 \text{ 或 } 1) \\
&= \mu - \mu^2 \\
&= \mu(1-\mu)
\end{aligned}
$$

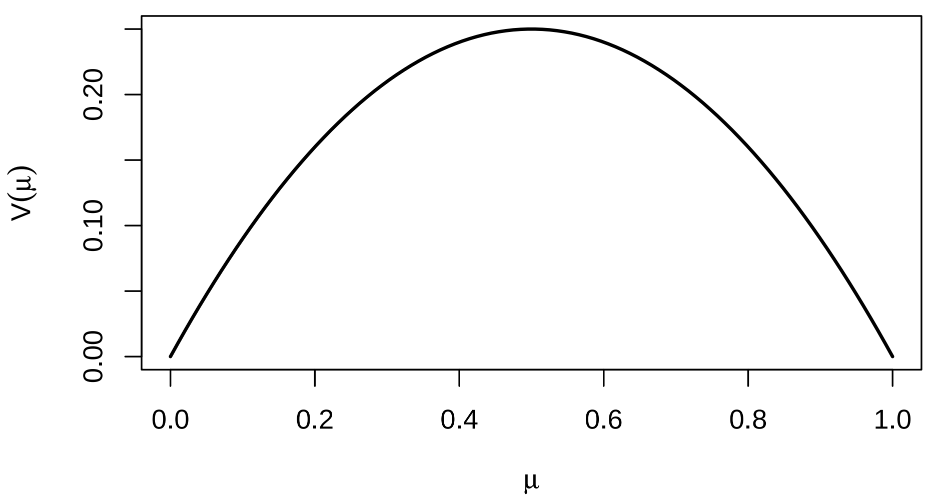
2.5 伯努利分布的方差图像

让我们画出伯努利分布方差随 变化的图像：

```r
mu <- seq(0, 1, length.out = 100)
V_mu <- mu * (1 - mu)

plot(mu, V_mu, type = "l", lwd = 2,
     xlab = expression(mu), ylab = expression(V(mu)),
     main = " 伯努利分布的方差")
```

....



从图中我们可以看出：- 当 = 0.5 时，方差最大，等于 0.25。- 当 接近 0 或 1 时，方差接近 0。- 这种方差随均值变化的特性称为" 异方差性"。

2.6 R 代码模拟伯努利分布

让我们用 R 代码模拟伯努利分布，以加深理解：

```
set.seed(123)  # 设置随机种子，确保结果可重复
n <- 1000      # 样本量
p <- 0.6       # 成功概率

# 生成伯努利随机变量
bernoulli_data <- rbinom(n, size = 1, prob = p)

# 计算样本均值和方差
mean_bernoulli <- mean(bernoulli_data)
var_bernoulli <- var(bernoulli_data)

cat(" 伯努利分布模拟结果：\n")
```

```
## 伯努利分布模拟结果：
```

```
cat(" 理论均值：", p, "\n")
```

```
## 理论均值： 0.6
```

```
cat(" 样本均值：", mean_bernoulli, "\n")
```

```
## 样本均值： 0.607
```

```
cat(" 理论方差：", p * (1 - p), "\n")
```

```
## 理论方差： 0.24
```

```
cat(" 样本方差：", var_bernoulli, "\n")
```

```
## 样本方差： 0.2387898
```

这个模拟帮助我们验证了伯努利分布的理论性质。

3 潜在线性模型

3.1 为什么需要潜在线性模型?

在实际问题中，我们经常遇到二元响应变量（如是否购买、是否患病等）。但这些变量背后可能有连续的潜在因素影响。潜在线性模型就是用来描述这种情况的。

3.2 潜在线性模型的定义

假设有一个看不见的连续变量 Z_i，它遵循正态分布：

$$Z_i \sim N(x_i^T \beta, 1)$$

这里，x_i 是预测变量，β 是系数。

我们观察到的二元变量 Y_i 是根据 Z_i 的值决定的：

$$Y_i = \begin{cases} 1 & \text{如果} Z_i \geq 0, \\ 0 & \text{如果} Z_i < 0. \end{cases}$$

3.3 潜在线性模型的概率解释

现在，我们来计算 $Y_i = 1$ 的概率：

$$
\begin{aligned}
P(Y_i = 1) &= P(Z_i \geq 0) \\
&= P(N(x_i^T\beta, 1) \geq 0) \\
&= P(N(0, 1) \geq -x_i^T\beta) \\
&= P(N(0, 1) \leq x_i^T\beta) \\
&= \Phi(x_i^T\beta)
\end{aligned}
$$

这里，Φ 是标准正态分布的累积分布函数。

我们引入一个新的术语：线性预测器 $\eta_i = x_i^T\beta$。

3.4 R 代码实现潜在线性模型

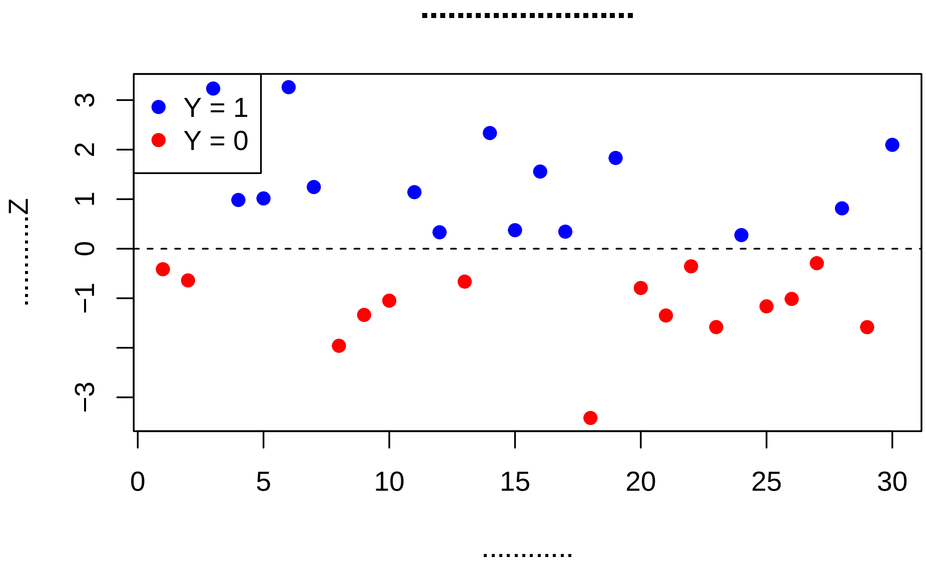
让我们用 R 代码模拟一个简单的潜在线性模型：

```
set.seed(123)
n <- 30          # 样本量
x <- rnorm(n)  # 预测变量
beta <- 1.5     # 系数

# 生成潜在变量 Z
z <- x * beta + rnorm(n)

# 根据 Z 生成观察到的 Y
y <- ifelse(z >= 0, 1, 0)

# 绘图
plot(1:n, z, type = "p", pch = 19, col = ifelse(y == 1, "blue", "red"),
     xlab = " 观察序号", ylab = " 潜在变量 Z",
     main = " 潜在线性模型示例")
abline(h = 0, lty = 2)
legend("topleft", legend = c("Y = 1", "Y = 0"),
       pch = 19, col = c("blue", "red"))
```



在这个图中: - 蓝色点表示 Y = 1 (Z 0) - 红色点表示 Y = 0 (Z < 0) - 虚线是 Z = 0 的分界线

这个模型展示了我们如何从连续的潜在变量得到二元的观察结果。

4 二项分布

4.1 什么是二项分布?

二项分布是伯努利分布的自然扩展。想象你不是抛一次硬币，而是抛了多次。二项分布描述的就是在 n 次独立的伯努利试验中，成功的次数。

4.2 二项分布的数学定义

如果我们进行 m 次独立的伯努利试验，每次成功的概率是 μ，那么成功总次数 Y 就服从二项分布:

$$Y \sim \mathrm{Binomial}(m, \mu)$$

4.3 二项分布的概率质量函数

二项分布的概率质量函数是：

$$p(y) = \binom{m}{y} \mu^y (1-\mu)^{m-y}, \quad y \in \{0, 1, \dots, m\}$$

这里：- $\binom{m}{y}$ 是组合数，表示从 m 个中选 y 个的方式数 - μ^y 是 y 次成功的概率 - $(1-\mu)^{m-y}$ 是 m-y 次失败的概率

4.4 二项分布的期望和方差

1. 期望：$E(Y) = m\mu$

   这很直观，就是试验次数乘以每次成功的概率。

2. 方差：$\mathrm{Var}(Y) = m\mu(1-\mu)$

   这是单次伯努利试验方差的 m 倍。

4.5 缩放二项分布

在实际应用中，特别是在广义线性模型中，我们经常使用缩放二项分布。定义 $Y^* = \frac{Y}{m}$，则：

$$E(Y^*) = \mu, \quad \mathrm{Var}(Y^*) = \frac{1}{m}\mu(1-\mu)$$

这样做的好处是，无论试验次数如何，期望值始终在 0 到 1 之间。

4.6 R 代码实现二项分布

让我们用 R 代码拟合一个二项分布模型。我们将使用 `turbines` 数据集，这个数据集记录了涡轮机的运行时间和出现裂缝的数量。

```r
data(turbines)

# 拟合二项式广义线性模型
glm.out <- glm(Fissures/Turbines ~ Hours, weights = Turbines,
               family = binomial(link = "probit"), data = turbines)

# 查看模型摘要
sumary(glm.out)
```

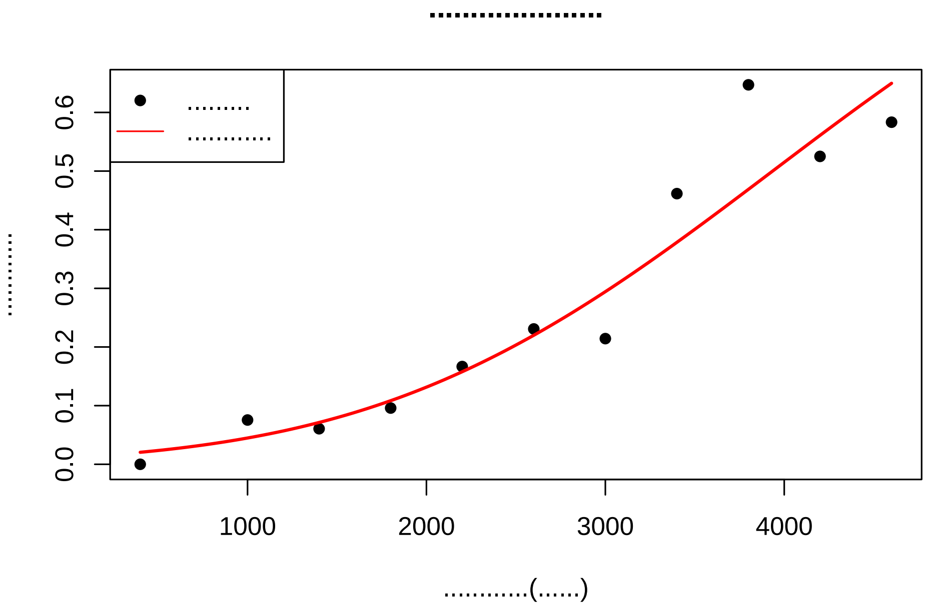
```
##                Estimate  Std. Error  z value  Pr(>|z|)
## (Intercept) -2.2758e+00  1.9742e-01 -11.5278 < 2.2e-16
## Hours        5.7832e-04  6.2597e-05   9.2388 < 2.2e-16
##
## n = 11 p = 2
## Deviance = 9.81484 Null Deviance = 112.67005 (Difference = 102.85521)
```

```r
# 绘制数据和拟合曲线
plot(Fissures/Turbines ~ Hours, data = turbines,
     xlab = " 运行时间（小时）", ylab = " 裂缝比例",
     main = " 涡轮机裂缝模型", pch = 16)

# 生成预测值
new_hours <- seq(min(turbines$Hours), max(turbines$Hours), length.out = 100)
pred <- predict(glm.out, newdata = data.frame(Hours = new_hours), type = "response")

# 添加拟合曲线
lines(new_hours, pred, col = "red", lwd = 2)

legend("topleft", legend = c(" 观察值", " 拟合曲线"),
       pch = c(16, NA), lty = c(NA, 1), col = c("black", "red"))
```



这个图展示了随着涡轮机运行时间的增加，出现裂缝的概率如何变化。红线是我们的模型预测，点是实际观察值。

5 逻辑回归

5.1 什么是逻辑回归?

逻辑回归是处理二元响应变量的一种常用方法。它使用 logistic 函数将线性预测转换为概率。

5.2 logit 链接函数

logit 链接函数定义为：

$$g(\mu) = \log\left(\frac{\mu}{1-\mu}\right) = \eta$$

这里，η 是线性预测器。

对应的均值函数（logistic 函数）是：

$$h(\eta) = \frac{\exp(\eta)}{1+\exp(\eta)} = \frac{1}{1+\exp(-\eta)}$$

5.3 参数解释

在逻辑回归中，系数的解释很重要：

- 对于二元预测变量，$\exp(\beta)$ 是优势比（odds ratio）。
- 对于连续预测变量，β 表示预测变量每增加一个单位，对数优势增加的量。

5.4 R 代码实现逻辑回归

让我们用逻辑回归重新分析涡轮机数据：

```r
# 拟合逻辑回归模型
glm.logit <- glm(Fissures/Turbines ~ Hours, weights = Turbines,
                 family = binomial(link = "logit"), data = turbines)

# 查看模型摘要
sumary(glm.logit)
```

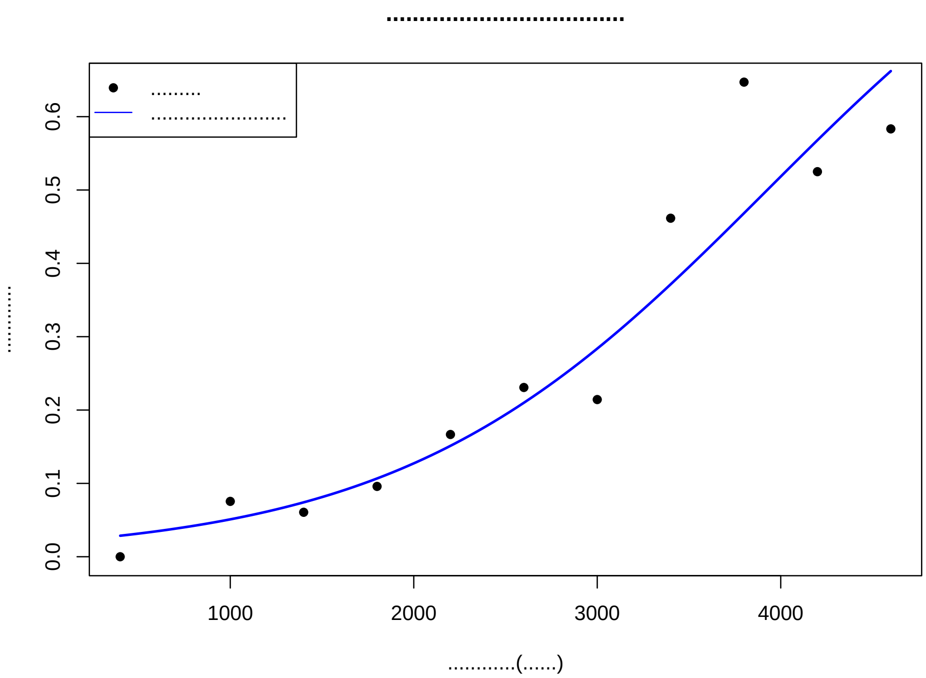
```
##                Estimate  Std. Error  z value  Pr(>|z|)
## (Intercept) -3.92359656  0.37795894 -10.3810 < 2.2e-16
## Hours        0.00099924  0.00011415   8.7537 < 2.2e-16
##
## n = 11 p = 2
## Deviance = 10.33147 Null Deviance = 112.67005 (Difference = 102.33858)
```

```r
# 绘制数据和拟合曲线
plot(Fissures/Turbines ~ Hours, data = turbines,
     xlab = " 运行时间（小时）", ylab = " 裂缝比例",
     main = " 涡轮机裂缝的逻辑回归模型", pch = 16)
```

```r
# 生成预测值
new_hours <- seq(min(turbines$Hours), max(turbines$Hours), length.out = 100)
pred_logit <- predict(glm.logit, newdata = data.frame(Hours = new_hours), type = "respo

# 添加拟合曲线
lines(new_hours, pred_logit, col = "blue", lwd = 2)

legend("topleft", legend = c(" 观察值", " 逻辑回归拟合曲线"),
       pch = c(16, NA), lty = c(NA, 1), col = c("black", "blue"))
```



5.5 解释逻辑回归结果

从上面的输出中，我们可以看到：

1. 截距（Intercept）为 -3.9236，这表示当运行时间为 0 时，出现裂缝的对数几率。

2. Hours 的系数为 0.0009992，这意味着每增加一小时的运行时间，裂缝出现的对数几率增加 0.0009992。

为了更直观地理解这个系数，我们可以计算优势比（odds ratio）：

```r
odds_ratio <- exp(coef(glm.logit)["Hours"])
cat(" 每增加一小时运行时间的优势比：", odds_ratio, "\n")
```

```
## 每增加一小时运行时间的优势比： 1.001
```

这个优势比意味着每增加一小时的运行时间，出现裂缝的几率会增加约 0.1%。

从图中我们可以看到，随着运行时间的增加，出现裂缝的概率呈 S 形曲线增加，这正是逻辑回归的特征。

6 容忍分布

6.1 什么是容忍分布?

容忍分布是一种用于推广潜在线性模型的概念。它允许我们使用不同的概率分布来描述潜在变量，从而得到不同的链接函数。

6.2 一般化潜在线性模型

我们可以将潜在线性模型写成：

$$Z_i = \eta_i + \epsilon_i$$

其中 ϵ_i 服从某个连续的实值分布。根据 ϵ_i 的分布不同，我们可以得到不同的模型。

6.3 常见的容忍分布

1. **正态分布**：导致 probit 链接函数
2. **逻辑分布**：导致 logit 链接函数
3. **Gumbel 分布**：导致互补对数对数（complementary log-log）链接函数
4. **Cauchy 分布**：导致 cauchit 链接函数

6.4 比较不同的链接函数

让我们用不同的链接函数来拟合涡轮机数据，看看它们有什么不同：

```
links <- c(Probit = "probit", Logit = "logit", CLogLog = "cloglog", Cauchit = "cauchit"

fit.turbines <- function(L) {
  glm(Fissures/Turbines ~ Hours, weights = Turbines,
      family = binomial(link = L), data = turbines)
}

glm.out <- lapply(links, fit.turbines)

# 比较系数
coef_comparison <- sapply(glm.out, coef)
print(coef_comparison)
```

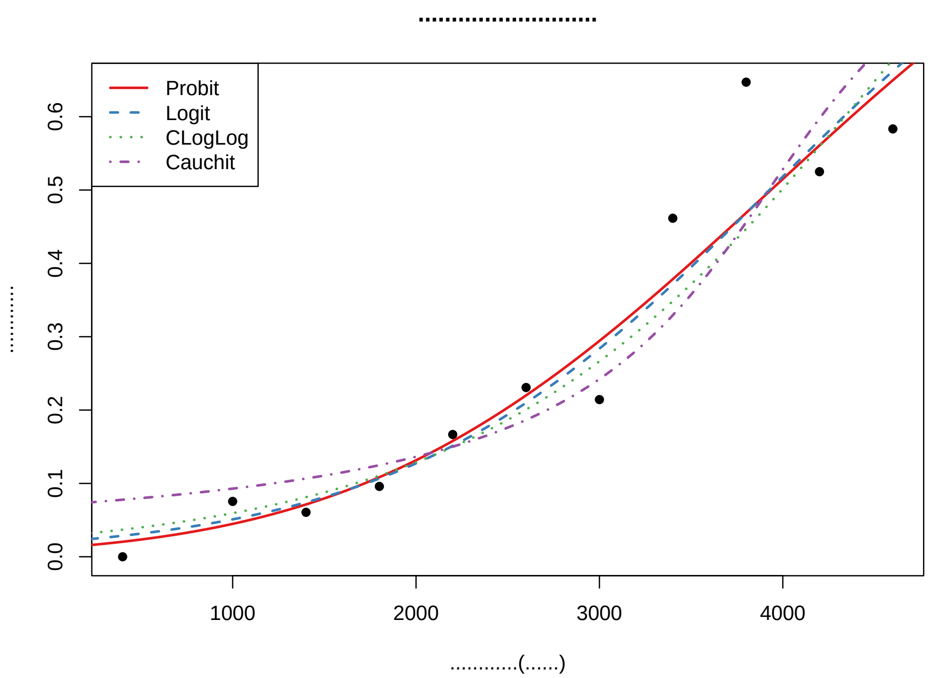
```
##                     Probit         Logit       CLogLog      Cauchit
## (Intercept) -2.2758074623 -3.9235965551 -3.6032798443 -4.467548420
## Hours        0.0005783211  0.0009992372  0.0008104936  0.001139074
```

```
# 绘制拟合曲线
newHours <- seq(0, 5000, length = 100)
predict.turbines <- function(model) {
  predict(model, type = "response", newdata = data.frame(Hours = newHours))
}
```

```
predicted <- sapply(glm.out, predict.turbines)

plot(Fissures/Turbines ~ Hours, data = turbines, pch = 16,
     xlab = " 运行时间 (小时) ", ylab = " 裂缝比例",
     main = " 不同链接函数的比较")

palette <- brewer.pal(4, "Set1")
matlines(newHours, predicted, lty = 1:4, lwd = 2, col = palette)
legend("topleft", lty = 1:4, lwd = 2, col = palette, legend = names(links))
```



6.5 解释不同链接函数的结果

从上面的图中，我们可以观察到：

1. 所有链接函数都产生了相似的 S 形曲线，这是二元响应模型的典型特征。
2. 在数据的中间范围内，所有模型的预测非常接近。

3. 在极端值处（很小或很大的运行时间），不同模型的预测开始出现差异。
4. Probit 和 Logit 链接函数的结果非常相似，这在实践中经常发生。
5. CLogLog（互补对数对数）链接函数在低概率区域上升较慢，但在高概率区域上升较快。
6. Cauchit 链接函数在两个极端都有较大的尾部效应。

选择哪种链接函数通常取决于：- 数据的特性 - 研究的具体背景 - 模型的拟合优度

在许多情况下，Logit 链接函数是首选，因为它的系数可以解释为对数优势比，这在许多领域（如流行病学）中有直观的解释。

7 练习

为了加深理解，请尝试以下练习：

1. 对于每种容忍分布，验证以下性质：
   - $F'(x) = f(x)$（累积分布函数的导数是概率密度函数）
   - $g(h(\eta)) = \eta$ 和 $h(g(\mu)) = \mu$（链接函数是均值函数的逆）
2. 使用 `glm()` 函数，尝试用不同的链接函数拟合你自己找到的二元响应数据。比较结果并解释差异。
3. 模拟一个二项分布的数据集，然后用逻辑回归模型拟合它。比较你的模拟参数和估计的参数。

8 总结

在这一章中，我们学习了：

1. 伯努利分布：最基本的二元概率分布
2. 潜在线性模型：连接连续潜在变量和二元观察结果
3. 二项分布：多次伯努利试验的结果
4. 逻辑回归：处理二元响应数据的强大工具

5. 容忍分布：通过不同的误差分布推广潜在线性模型

这些概念和方法在许多领域都有广泛的应用，如医学研究（预测疾病发生）、市场营销（预测客户行为）、金融（信用评分）等。掌握这些工具将使你能够处理各种涉及二元结果的实际问题。

9 参考文献

1. McCullagh, P., & Nelder, J. A. (1989). *Generalized Linear Models (2nd ed.).* Chapman and Hall/CRC.
2. Agresti, A. (2015). *Foundations of Linear and Generalized Linear Models.* Wiley.
3. Dobson, A. J., & Barnett, A. G. (2018). *An Introduction to Generalized Linear Models (4th ed.).* Chapman and Hall/CRC.