MLAPP 读书笔记 - 09 广义线性模型(Generalized linear models)和指数族分布(exponential family)

> A Chinese Notes of MLAPP,MLAPP 中文笔记项目
> https://zhuanlan.zhihu.com/python-kivy

记笔记的人：cycleuser

2018年6月27日09:26:15

9.1 概论

之前已经见到过很多概率分布了:正态(高斯)分布,伯努利分布(Bernoulli),学生T分布,均匀分布,γ分布等等.这些大多数都属于指数族分布(exponential family).本章就要讲这类分布的各种特点.然后我们就能用来推出很多广泛应用的定力和算法.

接下来我们会看到要构建一个生成分类器如何简单地用指数族分布中的某个成员来作为类条件密度.另外还会讲到如何构建判别模型,其中响应变量服从指数族分布,均值是输入特征的线性函数;这就叫做广义线性模型(Generalized linear models),将逻辑回归上的思想扩展到了其他类别的响应变量上去.

9.2 指数族分布

在定义指数族分布之前,先要说一下这东西重要的几个原因:
* 在特定的规范化条件下(regularity conditions),指数族分布是唯一有限规模充分统计量(finite-sized sufficient statistics)的分布族,这意味着可以将数据压缩称固定规模的浓缩概括而不损失信息.这在在线学习情况下特别有用,后面会看到.
* 指数族分布是唯一有共轭先验的分布族,这就简化了后验的计算,参考本书9.2.5.
* 指数族分布对于用户选择的某些约束下有最小假设集合,参考本书9.2.6.
* 指数族分布是广义线性模型(generalized linear models)的核心,参考本书9.3.
* 指数族分布也是变分推理(variational inference)的核心,参考本书21.2.

9.2.1 定义

概率密度函数(pdf)或者概率质量函数(pmf)$p(x|\theta)$,对$x = (x_1, \ldots, x_m) \in X^m, \theta \in \Theta \subseteq R^d$,如果满足下面的形式,就称其属于指数族分布(exponential family):

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

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

其中:

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

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

因此θ也叫作自然参数(natural parameters)或者规范参数(canonical parameters),$\phi(x) \in R^d$叫做充分统计向量(vector of sufficient statistics),$Z(\theta)$叫做配分函数(partition function),$A(\theta)$叫做对数配分函数(log partition function)或者累积函数(cumulant function),$h(x)$是一个缩放常数,通常为1.如果$\phi(x) = x$,就说这个是一个自然指数族(natural exponential family).

等式9.2可以泛华扩展,写成下面这种方式:
$p(x|\theta) = h(x) \exp[\eta(\theta)^T \phi(x) - A(\eta(\theta))]$(9.5)

其中的η是一个函数,将参数θ映射到规范参数(canonical parameters)$\eta = \eta(\theta)$,如果$dim(\theta) < dim(\eta(\theta))$,就成了弯曲指数族(curved exponential family),意味着充分统计(sufficient statistics)比参数(parameters)更多.如果$\eta(\theta) = \theta$,就说这个模型是规范形式的(canonical form).如果不加额外声明,都默认假设模型都是规范形式的.

9.2.2 举例

接下来举几个例子以便理解.

9.2.2.1 伯努利分布(Bernoulli)

$x \in \{0, 1\}$的伯努利分布写成指数族形式如下所示:
$Ber(x|\mu) = \mu^x (1-\mu)^{1-x} = \exp[x \log(\mu) + (1-x) \log(1-\mu)] = \exp[\phi(x)^T \theta]$(9.6)

其中的$\phi(x) = [I(x0), I(x=1), \theta = [\log(\mu) + (1-x) \log(1-\mu)]]$.不过这种表示是过完备的(over-complete),因为在特征(features)之间有一个线性依赖关系:

$1^T \backslash [hi(x) = I(x=0) + I(x=1) = 1$(9.7)

结果θ就不再是唯一可识别的(uniquely identifiable).通常都要求表述最简化(minimal),也就意味着关于这个分布要有为一个的θ.可以定义如下:

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

现在就有了$\phi(x) = x, \log(\frac{\mu}{1-\mu}$就是对数比值比(log-odds ratio),$Z = 1/(1-u)$.然后可以从规范参数里面恢复出均值参数$\mu$:

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

9.2.2.2 多重伯努利(Multinoulli)

多重伯努利分布表述成最小指数族如下所示(其中的$x_k \backslash \mathrm{I}(x=k)$):

$$\begin{aligned} Cat(x|\mu) &= \prod_{k=1}^{K} \mu_k^{x_k} = \exp[\sum_{k=1}^{K} x_k \log \mu_k] && (9.10)\\ &= \exp[\sum_{k=1}^{K-1} x_k \log \mu_k + (1-\sum_{k=1}^{K-1} x_k)\log(1-\sum_{k=1}^{K-1}\mu_k)] && (9.11)\\ &= \exp[\sum_{k=1}^{K-1} x_k \log(\frac{\mu_k}{1-\sum_{j=1}^{K-1}\mu_j}) + \log(1-\sum_{k=1}^{K-1}\mu_k)] && (9.12)\\ &= \exp[\sum_{k=1}^{K-1} x_k \log(\frac{\mu_k}{\mu_K}) + \log \mu_k] && (9.13) \end{aligned}$$

其中$\mu_K = 1-\sum_{k=1}^{K-1}\mu_k$.也可以写成下面的指数族形式

$$\begin{aligned} Cat(x|\theta) &= \exp(\theta^T \phi(x) - A(\theta)) && (9.14)\\ \theta &= [\log\frac{\mu_1}{\mu_K}, \ldots, \log\frac{\mu_{K-1}}{\mu_K}] && (9.15)\\ \phi(x) &= [I(x=1), \ldots, I(x=K-1)] && (9.16) \end{aligned}$$

从规范参数里面恢复出均值参数μ:

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

然后就能发现:

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

因此:

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

如果定义$\theta_K = 0$,就可以卸除$\mu = S(\theta), A(\theta) = \log \sum_{k=1}^{K} e^{\theta_k}$,其中的S是等式4.39中的柔性最大函数(softmax function).

9.2.2.3 单变量高斯分布(Univariate Gaussians)

单变量高斯分布写成指数族形式如下所示:

```
\begin{aligned}
  N(x|\mu,\sigma^2) &= \frac{1}{(2\pi\sigma^2)^{1/2}} \exp[-\frac{1}{2\sigma^2}(x-\mu)^2]    &\text{(9.20)}\\
  &=  \frac{1}{(2\pi\sigma^2)^{1/2}} \exp[-\frac{1}{2\sigma^2}x^2 +\frac{\mu}{\sigma^2}x  -\frac{1}{2\sigma^2}\r
  &= \frac{1}{Z(\theta}\exp(\theta^T\phi(x)))}    &\text{(9.22)}\\
\end{aligned}
```

其中

$$\theta = \begin{pmatrix} \mu/\sigma^2 \\ \frac{-1}{2\sigma^2} \end{pmatrix} \tag{9.23}$$

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

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

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

9.2.2.4 反面样本(Non-examples)

肯定不可能所有分布都属于指数族啊.比如均匀分布$X \sim Unif(a, b)$就不属于指数族,因为这个分布的支撑(support,其实大概也就是定义域的意思)是一来参数的.另外本书11.4.5所示的学生T分布也不属于指数族,因为不含有要求的形式.

9.2.3 对数配分函数(log partition function)

指数族的一个重要性质就是对数配分函数的导数(derivatives)可以用来生成充分统计的累积量(cumulants).由于这个原因才使得$A(\theta)$有时候也被叫做累积函数(cumulant function).先对一个单参数分布来证明一下;然后可以直接泛化扩展到K个参数的分布上.首先是求一阶导数得到了:

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

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

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

$$= \int \phi(x)\exp(\theta\phi(x) - A(\theta))h(x)dx \tag{9.30}$$

$$= \int \phi(x)p(x)dx = \mathrm{E}[\phi(x)] \tag{9.31}$$

然后求二阶导数就得到了:

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

$$= \int \phi(x)p(x)(\phi(x) - A'(\theta))dx \tag{9.33}$$

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

$$= \mathrm{E}[\phi^2(X)] - \mathrm{E}[\phi(x)]^2 = var[\phi(x)] \tag{9.35}$$

其中利用了$A'(\theta) = \frac{dA}{d\theta} = \mathrm{E}[\phi(x)]$.

在多变量的情况下,有:

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

然后就有:

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

因此协方差矩阵就是正定的,会发现$A(\theta)$是一个凸函数(参考本书7.3.3).

9.2.3.1 样例:伯努利分布

以伯努利分布为例.$A(\theta) = \log(1 + e^\theta)$,这样就有均值为:
$$\frac{dA}{d\theta} = \frac{e^\theta}{1+e^\theta} = \frac{1}{1+e^{-\theta}} = sigm(\theta) = \mu \tag{9.38}$$

方差为:

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

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

9.2.4 指数族的最大似然估计(MLE)

指数族模型的似然函数形式如下:

$p(D|\theta) = [\prod_{i=1}^N h(x_i)]g(\theta)^N \exp(\eta(\theta)^T[\sum_{i=1}^N \phi(x_i)])$(9.41)
然后会发现充分统计量(sufficient statistics)为N以及:

$\phi(D) = [\sum_{i=1}^N \phi_1(x_i), \ldots, \sum_{i=1}^N \phi_k(x_i)]$(9.42)

对伯努利模型为$\phi = [\sum_i I(x_i = 1)]$,单变量高斯模型则有$\phi = [\sum_i x_i, \sum_i x_i^2]$.(还需要知道样本规模N.)

Pitman-Koopman-Darmois定理表面,在特定规范化条件(regularity conditions)下,指数族分布是唯一有有限充分统计量的分布.(这里的有限(finite)是与数据集规模无关的.)

这个定理需要的一个条件是分布的支撑(support,就当做定义域理解了)不能独立于参数.比如下面这个均匀分布为例:

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

似然函数就是:

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

所以充分统计量就是N和$s(D) = \max_i x_i$.这个均匀分布规模有限,但并不是指数族分布,因为其支撑集合(support set)X依赖参数.

接下来就要讲下如何对一个规范指数族分布模型(canonical exponential family model)计算最大似然估计(MLE).给定了N个独立同分布的数据点,对数似然函数(log-likelihood） 为:

$\log p(D|\theta) = \theta^T\phi(D) - NA(\theta)$(9.45)

由于$-A(\theta)$在θ上是凹的,而$\theta^T\phi(D)$在θ上是线性的,而对数似然函数是凹的,所以这就会有唯一的一个全局最大值.要推导这个最大值,要用到对数配分函数(log partition function)的导数生成充分统计量向量的期望值(参考本书9.2.3):

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

设置梯度为零,就可以得到最大似然估计(MLE)了,充分统计量的经验均值必须等于模型的理论期望

充分统计量,也就是$\hat{\theta}$必须满足下面的条件:

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

这就叫做矩捕获(moment matching).比如在伯努利分布中,就有$\phi(X)=I(X=1)$,所以最大似然估计(MLE)就满足:

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

9.2.5 指数族的贝叶斯方法*

我们已经看见了,如果先验和似然函数是共轭的,那么确定的贝叶斯分析就相当简单了.粗略地理解也可以认为这意味着先验$p(\theta|\tau)$和似然函数$p(D|\theta)$形式一样.为了好理解,就需要似然函数you有限的充分统计量,所以就可以写成$p(D|\theta)=p(s(D)|\theta)$.这表明只有指数族分布才有共轭先验.接下来推导先验和后验的形式.

9.2.5.1 似然函数

指数族似然函数为:

$$p(D|\theta)\propto g(\theta)^N\exp(\eta(\theta)^Ts_N) \tag{9.49}$$

其中$s_N=\sum_{i=1}^{N}s(x_i)$.以规范参数的形式就成了:

$$p(D|\eta)\propto\exp(N\eta^T\bar{s}-NA(\eta)) \tag{9.50}$$

其中$\bar{s}=\frac{1}{N}s_N$.

9.2.5.2 先验

自然共轭先验形式如下:

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

然后写成$\tau_0=v_0\bar{\tau}_0$来区分先验中伪数据的规模v_0和在这个伪数据上充分统计的均值$\bar{\tau}_0$.写成规范形式(canonical form),先验就成了:

$$p(\eta|v_0,\bar{\tau}_0)\propto\exp(v_0\eta^T\bar{\tau}_0-v_0A(\eta)) \tag{9.52}$$

9.2.5.3 后验

后验形式为:
$$p(\theta|D)=p(\theta|v_N,\tau_N)=p(\theta|v_0+N,\tau_0+s_N) \tag{9.53}$$

可见就可以通过假发来更新超参数(hyper-parameters).用规范性就是:

$$p(\theta|D) \propto \exp(\eta^T(v_0\bar{\tau}_0 + N\bar{s}) - (v_0+N)A(\eta)) \tag{9.54}$$

$$= p(\eta|v_0+N, \frac{v_0\bar{\tau}_0 + N\bar{s}}{v_0+N}) \tag{9.55}$$

所以就会发现后验超参数是先验均值超参数和充分统计量均值的凸组合(convex combination).

9.2.5.4 后验预测密度

给定已有数据$D = (x_1, \ldots, x_N)$,对未来观测量$D' = (\tilde{x}_1, \ldots, \tilde{x}_N)$的预测密度的通用表达式进行推测,过程如下所述.为了记号简单,将充分统计量和数据规模结合起来,即:
$\tilde{\tau}_0 = (v_0, \tau_0), \tilde{s}(D) = (N, s(D)), \tilde{s}(D') = (N', s(D'))$.先验就是:

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

似然函数和后验形式相似.因此有:

$$p(D'|D) = \int p(D'|\theta)p(\theta|D)d\theta \tag{9.57}$$

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

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

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

如果$N = 0$,这就成了D'的边缘似然函数,降低(reduce)到了通过先验的归一化项乘以一个常数而得到的后验归一化项(normalizer)的类似形式.

9.2.5.5 样例:伯努利分布

举个简单例子,这回用新形式回顾一下β伯努利分布(Beta-Bernoulli model).

似然函数为:

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

共轭先验为:

$$p(\theta|v_0,\tau_0) \propto (1-\theta)^{v_0}\exp(\log(\frac{\theta}{1-\theta})\tau_0) \tag{9.62}$$
$$= \theta^{\tau_0}(1-\theta)^{v_0-\tau_0} \tag{9.63}$$

如果定义$\alpha = \tau_0 + 1, \beta = v_0 - \tau_0 + 1$,就会发现这是一个$\beta$分布.

然后可以推导后验了,如下面所示,其中$s = \sum_i I(x_i = 1)$是充分统计量:

$$p(\theta|D) \propto \theta^{\tau_0+s}(1-\theta)^{v_0-\tau_0+n-s} \tag{9.64}$$
$$= \theta^{\tau_n}(1-\theta)^{v_n-\tau_n} \tag{9.65}$$

后验预测分布的推测过程如下所示.设$p(\theta) = Beta(\theta|\alpha,\beta)$,$s = s(D)$是过去数据(past data)的人头数.就可以预测里一系列未来人头朝上$D' = (\tilde{x}_1,\ldots,\tilde{x}_m)$的概率,充分统计量为$s' = \sum_{i=1}^m I(\tilde{x}_i = 1)$,则后验预测分布为:

$$p(D'|D) = \int_0^1 p(D'|\theta|Beta(\theta|\alpha_n,\beta_n)d\theta =$$
$$= \frac{\Gamma(\alpha_n+\beta_n)}{\Gamma(\alpha_n)\Gamma(\beta_n)}\int_0^1 \theta^{\alpha_n+t'-1}(1-\theta)^{\beta_n+m-t'-1}$$
$$= \frac{\Gamma(\alpha_n+\beta_n)\Gamma(\alpha_{n+m})\Gamma(\beta_{n+m})}{\Gamma(\alpha_n)\Gamma(\beta_n)\Gamma(\alpha_{n+m}+\beta_{n+m})}$$

其中

$$\alpha_{n+m} = \alpha_n + s' = \alpha_n + s + s' \tag{9.69}$$
$$\beta_{n+m} = \beta_n + (m-s') = \beta + (n-s) + (m-s') \tag{9.70}$$

9.2.6 指数族分布最大熵的推导*

指数族分布很方便,可对这种分布的使用有什么更深层的评判方法么?还真有:就是关于数据做出最小假设的分布,受限于用户指定约束条件,后面会详细解释.加入我们想知道的是某些特定函数或者特征的期望值:

$$\sum_x \int_k(x)p(x) = F_k \tag{9.71}$$

其中的F_k是已知常数,而$f_k(x)$可以是任意函数.最大熵(maximum entropy,缩写为maxent)原则说的是应该选择有最大熵的分布(最接近均匀分布),在分布的矩(moments)陪指定函数的经验矩(empirical moments)的约束条件下.

要最大化对应等式9.71中约束条件下的熵,兼顾约束条件$p(x) \geq 0, \sum_x p(x) = 1$,就需要使用拉格朗日乘数(Lagrange multipliers).拉格朗日函数(Lagrangian)为:

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

(9.72)

可以使用方差的积分来关于函数p求导,不过需要发展一个更简单的方法,然后将p当做一个固定长度向量(因为这里假设x是离散的).然后就有:

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

设$\frac{\partial J}{\partial p(x)} = 0$,就得到了:

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

此处参考原书图9.1

其中的 $Z = e^{1+\lambda_0}$.利用概率累加总和等于1的约束条件,就有了:

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

因此归一化常数(normalization constant)就是:
$Z = \sum_x \exp(-\sum_k \lambda_k f_k(x))$(9.76)

所以最大熵分布$p(x)$就有了指数族分布中的形式(本书9.2),也叫作吉布斯分布(Gibbs distribution)

9.3 广义线性模型(Generalized linear models,缩写为GLMs)

线性回归和逻辑回归都属于广义线性模型的特例(McCullagh and Nelder 1989).
这些模型中输出密度都是指数族分布(参考本书9.2),而均值参数都是输入的线性组合,经过可能是非线性的函数,比如逻辑函数等等.下面就要详细讲一下广义线性模型(GLMs).为了记号简单,线看标量输出的情况.(这就排除了多远逻辑回归,不过这只是为表述简单而已.)

9.3.1 基础知识

要理解广义线性模型,首先要考虑一个标量响应变量的无条件分布(unconditional dstribution)的情况:

$p(y_i|\theta, \sigma^2) = \exp[\frac{y_i\theta - A(\theta)}{\sigma^2} + c(y_i, \sigma^2)]$(9.77)

上式中的σ^2叫做色散参数(dispersion parameter),通常设为1.θ是自然参数,A是配分函数,c是归一化常数.例如,在逻辑回归的情况下,θ就是对数比值比(log-odds ratio),$\theta = \log(\frac{\mu}{1-\mu})$,其中$\mu = \mathrm{E}[y] = p(y=1)$是均值参数(mean parameter),参考本书9.2.2.1.要从均值参数转成自然参数(natural parameter),可以使用一个函数ϕ,也就是$\theta = \Psi(\mu)$.这个函数由指数族分布的形式唯一

确定(uniquely determined).实际上这是一个可逆映射(invertible mapping),所以也就有$\mu = \Psi^{-1}(\theta)$.另外通过本书9.2.3可以知道这个均值可以通过对配分函数(partition function)求导而得到,也就是有$\mu = \Psi^{-1}(\theta) = A'(\theta)$.

然后加上输入/协变量(covariates).先定义一个输入特征的线性函数:

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

分布	Link $g(\mu)$	$\theta = \psi(\mu)$	$\mu = \psi^{-1}(\theta) = \matmrm{E}[y]$
$N(\mu, \sigma^2)$	indentity	$\theta = \mu$	$\mu = \theta$
$Bin(N, \mu)$	logit	$\theta = \log \frac{\mu}{1-\mu}$	$\mu = sigm(\theta)$
$Poi(\mu)$	log	$\theta = \log(\mu)$	$\mu = e^{\theta}$

表 9.1 常见广义线性模型(GLMs)的连接函数(link function)ψ.

然后使这个分布的均值为这个线性组合的某个可逆单调函数.通过转换,得到这个函数就叫做均值函数(mean function),记作g^{-1},所以:
$\mu_i = g^{-1}(\eta_i) = g^{-1}(w^T x_i)$(9.79)

如图9.1所示为这个简单模型的总结.

均值函数(mean function)的逆函数,记作$g()$,就叫做连接函数(link function).我们可以随意选择任意函数来作为连接函数,只要是可逆的,以及均值函数$g()$有适当的范围.例如在逻辑回归里面,就设置$\mu_i = g^{-1}(\eta_i) = sigm(\eta_i)$.

连接函数有一个特别简单的形式,就是$g = \phi$,这也叫做规范连接函数(canonical link function).这种情况下则有$\theta_i = \eta_i = w^T x_i$,所以模型就成了:

$p(y_i|x_i, w, \sigma^2) = \exp[\frac{y_i w^T x_i - A(w^T x_i)}{\sigma^2} + c(y_i, \sigma^2)]$(9.80)

表格9.1中所示的是一些分布和规范连接函数.可见伯努利分布或者二项分布的规范连接函数是$g(\mu) = \log(\eta/(1-\eta))$,而你函数是逻辑函数(logistic function)$\mu = sigm(\eta)$.

基于本书9.2.3的结果,可以得到响应变量的均值和方差:

$$\mathrm{E}[y|x_i, w, \sigma^2] = \mu_i = A'(\theta_i) \qquad (9.81)$$
$$var[y|x_i, w, \sigma^2] = \sigma_i^2 = A''(\theta_i)\sigma^2 \qquad (9.82)$$

为了记好清楚,接下来就看一些简单样例.

对于线性回归,则有:

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

其中$y_i\in R,\theta_i=\mu_i=w^Tx_i$,而$A(\theta)=\theta^2/2$,所以$\mathrm{E}[y_i]=\mu_i,var[y_i]=\sigma^2$.

对于二项回归(binomial regression),则有:

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

其中$y_i\in\{0,1,\dots,N_i\},\pi_i=sigm(w^Tx_i),\theta_i=\log(\pi_i/(1-\pi_i))=w^Tx_i,\sigma^2=1.$
$A(\theta)=N_i\log(1+e^\theta)$,所以$\mathrm{E}[y_i]=N_i\pi_i=\mu_i,var[y_i]=N_i\pi_i(1-\pi_i)$

对于泊松分布(poisson regression),则有:
$\log p(y_i|x_i,w)=y_i\log\mu_i-\mu_i-\log(y_i!)$(9.85)

其中$y_i\in\{0,1,2,\dots\},\mu_i=\exp(w^Tx_i),\theta=\log(\mu_i)=w^Tx_i,\sigma^2=1.$而$A(\theta)=e^\theta$,所以$\mathrm{E}[y_i]=var[y_i]=\mu_i$.泊松回归在生物统计中应用很广,其中的$y_i$可能代表着给定人群或者地点的病患数目,或者高通量测序背景下基因组位置的读数数量,参考(Kuan et al. 2009).

9.3.2 最大似然估计(MLE)和最大后验估计(MAP)

广义线性模型的最重要的一个性质就是可以用和逻辑回归拟合的同样方法来进行拟合.对数似然函数形式如下所示:

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

$$l_i\triangleq\theta_iy_i-A(\theta_i) \quad (9.87)$$

$$\frac{dl_i}{dw_j}=\frac{dl_i}{d\theta_i}\frac{d\theta_i}{d\mu_i}\frac{d\mu_i}{d\eta_i}\frac{d\eta_i}{\mathrm{d}w_j} \quad (9.88)$$

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

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

名称	公式
Logistic	$g^{-1}9\eta)sigm(\eta)=\frac{e^\eta}{1+e^\eta}$
Probit	$g^{-1}9\eta)=\Phi(\eta)$
Log-log	$g^{-1}9\eta)=\exp(-\exp(-\eta))$

Complementary log-log	$g^{-1}9\eta) = 1 - \exp(-\exp(\eta))$

表格9.2 二值回归(binary regression)的一些可能的均值函数的总结.

如果使用规范连接函数$\theta_i = \eta_i$,就简化成了:
$\nabla_w l(w) = \frac{1}{\sigma^2}[\sum_{i=1}^N (y_i - \mu_i)x_i]$(9.91)

这就成了输入向量以误差为权重的求和.这可以在一个(很复杂的)梯度下降过程中用到,具体参考本书8.5.2.不过为了提高效率,还是应该用一个二阶方法.如果使用一个规范连接函数,海森矩阵(Hessian)就是:
$H = -\frac{1}{\sigma^2}\sum_{i=1}^N \frac{d\mu_i}{d\theta_i}x_i x_i^T = -\frac{1}{\sigma^2}X^T S X$(9.92)

其中$S = diag(\frac{d\mu_1}{d\theta_1}, \ldots, \frac{d\mu_N}{d\theta_N})$是对角权重矩阵(diagonal weighting matrix).这可以用在一个迭代重加权最小二乘法(Iteratively reweighted least squares,缩写为IRLS)算法内部(参考本书8.3.4).然后可以有下面所示的牛顿更新:

$$w_{t+1} = (X^T S_t X)^{-1} X^T S_t z_t \qquad (9.93)$$
$$z_t = \theta_t + S_t^{-1}(y - \mu_t) \qquad (9.94)$$

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

如果我们扩展求导(extend derivation)来处理非规范连接函数(non-canonical links),就会发现海森矩阵(Hessian)有另外一项.不过最终期望海森矩阵(expected Hessian)和等式9.92一模一样,利用这个期望海森矩阵(也就是费舍尔信息矩阵(Fisher information matrix))来替代真实的海森矩阵就叫做费舍尔排序方法(Fisher scoring method).

修改上面的过程来进行高斯先验的最大后验估计(MAP estimation)就很简单了:只需要调整目标函数(objective),梯度*gradient)和海森矩阵(Hessian),就像是在本书8.3.6里面对逻辑回归加上l_2规范化(regularization)一样.

9.3.3 贝叶斯推导

广义线性模型(GLMs)的贝叶斯推导通常都用马尔科夫链蒙特卡罗方法(Markov chain Monte Carlo,缩写为 MCMC),参考本书24章.可用基于迭代重加权最小二乘法(IRLS-based proposal)来的Metropolis Hastings方法(Gamerman 1997),或者每个全条件(full conditional)使用自适应拒绝采样(adaptive rejection sampling,缩写为ARS)的吉布斯采样(Gibbs sampling)(Dellaportas and Smith 1993).更多信息参考(Dey et al. 2000).另外也可以用高斯近似(Gaussian approximation,本书8.4.1)或者变分推理(variational inference,本书21.8.11).

9.4 概率单位回归(Probit regression)

在(二值化)洛基回归中,用到了形式为$p(y=1|x_i,w)=sigm(w^Tx_i)$的模型.对任何从区间$[-\infty,\infty]$到$[0,1]$进行映射的函数$g^{-1}$,一般都可以写出$p(y=1|x_i,w)=g^{-1}(w^Tx_i)$.表格9.2所示为若干可能的均值函数.
在本节,要讲的情况是$g^{-1}(\eta)=\Phi(\eta)$,其中的$\Phi(\eta)$是标准正态分布(standard normal)的累积密度函数(cdf).这样就叫概率单位回归(probit regression).概率函数(probit function)和逻辑函数(logistic function)很相似,如图8.7(b)所示.不过这个模型比逻辑回归有一些优势.

9.4.1 使用基于梯度优化的最大似然估计(MLE)和最大后验估计(MAP)

我们可以对概率单位回归使用标准梯度方法来得到最大似然估计(MLE).设
$\mu_i=w^Tx_i,\tilde{y}_i\in\{-1,+1\}$.然后对于一个特定情况的对数似然函数的梯度就是:
$g_i\triangleq\frac{d}{dw}\log p(\tilde{y}_i|w^Tx_i)=\frac{d\mu_i}{dw}\frac{d}{d\mu_i}\log p(\tilde{y}_i)|w^Tx_i)=x_i\frac{\tilde{y}_i\phi(\mu_i)}{\Phi(\tilde{y}_i\mu_i)}$(9.95)

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

$p(w)=N(0,V_0)\sum_i g_i+2V_0^{-1}w\sum_i H_i+2V_0^{-1}$

9.4.2 潜在变量解释(latent variable interpretation)

$$u_{0i}\triangleq w_0^Tx_i+\delta_{0i} \quad (9.97)$$
$$u_{1i}\triangleq w_1^Tx_i+\delta_{1i} \quad (9.98)$$
$$y_i=I(u_{1i}\geq u_{10}) \quad (9.99)$$

随机效用模型(random utility model,缩写为RUM,McFadden 1974; Train 2009).

$\epsilon_i=\delta_{1i}-\delta_{0i}$

$$z_i\triangleq w^Tx_i+\epsilon_i \quad (9.100)$$
$$\epsilon_i\sim N(0,1) \quad (9.101)$$
$$y_i=1=I(z_i\geq 0) \quad (9.102)$$

根据(Fruhwirth-Schnatter and Fruhwirth 2010),将这叫做差别随机效用模型(difference random utility model,缩写为dRUM).

当边缘化去掉(marginalize out)z_i之后,就恢复(recover)了概率单位模型(probit model):

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

$$p(w^T x_i + \epsilon \geq 0) = p(\epsilon \geq -w^T x_i) \quad (9.104)$$

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

上面用到了高斯分布的对称性(symmetry).这个潜在变量解释提供了你和模型的另外一种方法,具体参考本书11.4.6.

如果我们队δ使用一个冈贝尔分布(Gumbel distribution),得到的就是对ϵ_i的逻辑分布(logistic distribution),模型也降低到了(reduces to)逻辑回归(logistic regression).更多细节参考本书24.5.1.

9.4.3 有序概率回归(Ordinal probit regression)*

概率回归的潜在变量解释的一个优势就是很容易扩展到响应变量有序的情况下,也就是取C个离散值,以某种方式排序,比如从小到大等.这样就成了有序回归(ordinal regression).基本思想如下所述.引入$C+1$个阈值γ_j以及集合:

$$y_i = j \text{ if } \gamma_{j-1} \leq z_i \leq \gamma_j \quad (9.106)$$

其中的$\gamma_0 \leq \ldots \leq \gamma_C$.为了好辨认,就设$\gamma_0 = -\infty, \gamma_1 = 0, \gamma_C = \infty$.例如,如果$C = 2$,这就降低到了标准二值化概率模型(standard binary probit model),其中$z_i < 0$导致$y_i = 0$,而$z_i \geq 0$对应就是$y_i = 1$.如果C=3,就把这个实线(real line)分成了三个区间:$(-\infty, 0], (0, \gamma_2], (\gamma_2, \infty)$.这时候通过调节参数$\gamma_2$就可以确保每个区间有正确相关规模的概率质量,也能符合每个类标签的经验频率(empirical frequencies).

对这个模型找最大似然估计(MLE)就可能比二值化概率回归要更麻烦点了,因为需要优化w和γ,而后面的那个必须服从一个有序约束(ordering constraint).(Kawakatsu and Largey 2009)提出了一个基于期望最大化(EM)的方法.另外从这个模型也可以推导出一个简单的吉布斯采样算法(Gibbs sampling algorithm),参考(Hoff 2009, p216).

9.4.4 多项概率模型(Multinomial probit models)*

然后考虑响应变量可以去C个无排序分类值的情况,也就是$y_i \in \{1, \ldots, C\}$.多项概率模型定义如下所示:

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

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

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

关于这个模型的更多细节以及和多项逻辑回归(multinomial logistic regression)之间的关系可以去参考(Dow and Endersby 2004; Scott 2009; Fruhwirth-Schnatter and Fruhwirth 2010).(通过定义

$w = [w_1, \ldots, w_C], x_{ic} = [0, \ldots, 0, x_i, 0, \ldots, 0]$,就可以恢复到更熟悉的函数方程$z_{ic} = x_i^T w_C$.)由于只有相对效用(relative utilities)有影响,可以将R限制为一个相关矩阵(correlation matrix).如果不去设$y_i = \arg\max_C z_{ic}$而是使用$y_{ic} = I(z_{ic} > 0)$,就得到了多元概率模型(multivariate probit model),也是对于二值化输出相关的C建模的一种方法,参考(Talhouk et al. 2011).

9.5 多任务学习(multi-task learning)

有时候要对多个相关的分类或者拟合模型进行你和.通常都可能会假设不同模型之间的输入输出映射是相似的,所以可以在同时拟合所有参数来得到更好的性能.在机器学习里面,这一般叫做多任务学习 (multi-task learning,Caruana 1998), 转换学习 (transfer learning,Raina et al.2005), 或者 (learning to learn,Thrun and Pratt 1997).在统计学上,通常的解决方法是分层贝叶斯模型(hierarchical Bayesian models,Bakker and Heskes 2003),当然也有一些其他方法(Chai 2010),后文会讲到.

9.5.1 多任务学习的分层贝叶斯

设y_{ij}是第j群(group)当中的第i项(item)的响应变量,其中$i = 1 : N_j, j = 1 : J$.例如,j可以对学校进行索引,然后i就是检索该学校中的学生,然后y_{ij}就是这个学生的测试分数,如本书5.6.2所示.或者也可以用j来对人进行索引,然后i代表队是购买次数,这样y_{ij}就只带被购买的特定商品(这就叫做离散选择模型(discrete choice modeling, Train 2009).设x_{ij}是对应y_[ij]的特征向量.目标就是要对于所有的j拟合出模型$p(y_j|x_j)$.

虽然有的组可能有很多数据,通常都是长尾的(long tail),其中大多数组都只有很少的数据.因此不能很有把握地去分开拟合各个模型,可又不想对所有组使用同一个模型.所以就折中一下,可以对每个组拟合一个离散的模型,但设置在不同的组织间模型参数具有相似性.更具体来说就是设$\mathrm{E}[y_{ij}|x_{ij}] = g(x_i^T \Beta_j)$,其中的g是广义线性模型(GLM)的连接函数(link function).另外设$\Beta_j \sim N(\Beta_*, \sigma_j^2 I), \Beta_* \sim N(\mu, \sigma_*^2 I)$.在这个模型里面,小样本规模的组就从更多样本的组借用统计强度,因为$\Beta_j$是和潜在通用母变量(latent common parents)$\Beta_*$相关的(这一点相关内容参考本书5.5).σ_j^2这一项控制了第j组对通用母变量(common parents)的依赖程度,而σ_*^2这一项控制了全局先验的强度.

为了简单起见,假设$\mu = 0$,这样σ_j^2和σ_*^2就都是一直的了(可以通过交叉验证来设置).全局对数概率函数(overall log probability)形式为:

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

可以使用标准梯度方法进行对$\beta = (\Beta_{1:j}, \Beta_*)$的最大后验估计(MAP).或者也可以使用迭代优化的策略,在$\Beta_j$和$\Beta_*$之间进行优化;因为似然函数和先验都是凸函数,这就保证了会收敛到全局最优解.要记住一旦一个模型训练出来了,就可以不用理会$\Beta_*$,分别使用每个模型.

9.5.2 应用:个性化垃圾邮件过滤

多任务学习的一个有趣应用就是个性化垃圾邮件过滤(personalized email spam filtering).假如要针对每个用户来拟合一个分类器$\Beta_j$.由于大部分用户都不会来标记他们的邮件是不是垃圾邮件,这就很难分开来对他们各自的模型进行估计.所以要设$\Beta_j$有一个通用的先验$\Beta_*$,表示了一个通用用户的参数.

这时候就可以利用上面的模型来估计这行为,这需要一点小技巧(Daume 2007b; Attenberg et al. 2009; Weinberger et al. 2009):将每个特征x_i都只做两个分本,一个联接(concatenated)到用户id,另外的一份则不做联接.这样要学习的预测器(predictor)的形式就为:

$$\mathrm{E}[y_i|x_i, u] = (\Beta_*, w_1, \ldots, w_J)^T [x_i, I(u=1)x_i, \ldots, I(u=J)x_i] \quad (9.111)$$

其中的u是用户id.也就是说:

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

因此$\Beta_*$就可以从每个人的邮件中来进行估计,而w_j则只从第j个用户的邮件中来估计.

这和上面的分层贝叶斯模型具有对应关系(correspondence),定义$w_j = \Beta_j - \Beta_*$.然后原始模型的对数概率函数就可以重新写成下面的形式:

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

如果我们假设$\sigma_j^2 = \sigma_*^2$,效果就和使用增强特征技巧(augmented feature trick)一样了,对w_i和$\Beta_*$都有一样的规范化强度(regularizer strength).不过如果让这两个不相等通常能得到更好的性能(Finkel and Manning 2009).

9.5.3 应用:域自适应(Domain adaptation)

域自适应问题是要训练从不同分布取样来的数据,比如邮件和新闻报道的文本.这个问题很明显是一个多任务学习的特例,其中各个任务都相同.

(Finkel and Manning 2009)使用了上面的分层贝叶斯模型来使用域自适应来实现两个自然语言处理任务(NLP tasks),命名实体识别(namely named entity recognition)和解译(parsing).他们的研究成果比区分每个数据集来拟合分散模型有更大的进步,而对于将所有数据放到一个简单模型来拟合

相比就提升较小了.

9.5.4 其他类别的先验

在多任务学习里面,通常都假设先验是高斯分布的.不过有时候也可能选择别的先验更适合.比如对于联合分析(conjoint analysis)的任务,这个任务需要想办法弄清用户最喜欢产品的哪个特征.这可以使用跟前文同样的分层贝叶斯模型设置,不过要对$\Beta_j$使用更加稀疏的先验(sparsity-promoting prior),而不是高斯先验.这就叫做多任务特征选择(multi-task feature selection).更多可用方法等等参考(Lenk et al. 1996; Argyriou et al. 2008).

总去假设所有任务都一样想死并不总是很合理的.如果把不同质量(qualitatively different)的任务的参数汇集到一起,计算性能回避不汇聚要更差,因为咱们先验中的归纳偏见(inductive bias)是错误的.实际上已经有研究发现有点时候多任务学习要比分开解决每个任务效果更差,这也叫做负转换(negative transfer).

这类问题的另外一个方法就是使用更灵活的先验,比如混合高斯模型.这类灵活先验可以提供健壮性,应对先验误判(prior mis-specification).更多细节参考(Xue et al.2007; Jacob et al. 2008).当然也可以把稀疏提升先验(sparsity-promoting priors)混合起来使用(Ji et al. 2009).当然也有很多其他变体方法.

9.6 广义线性混合模型(Generalized linear mixed models)*

假如将多任务学习场景扩展来允许响应变量包含分组水平(group level)x_j,和项目水平(item level)x_{ij}.然后也允许参数$\Beta_j$在组间改变,或者参数α在组间固定.这样就得到了下面的模型:

$$\mathrm{E}[y_{ij}|x_{ij},x_j] = g(\phi(x_{ij})^T\Beta_j + \phi_2(x_j)^T\Beta_j' + \phi_3(x_{ij})^T\alpha + \phi_t(x_j)^T\alpha') \quad (9.114)$$

其中的ϕ_k是基函数(basis functions).这个函数如图9.2(a)所示.(这种图在本书第十章会解释.)参数$\Beta_j$的个数随着组个数增长而增长,而参数α则是固定的.

频率论里面将$\Beta_j$这一项叫做随机效应(random effects),因为它们在各组中随机变化;称α为固定效应(fixed effect),看做是一个固定但未知的敞亮.如果一个模型同时有固定效应和随机效应,就称之为混合模型(mixed model).如果$p(y|x)$是一个广义线性模型(GLM),整个模型就叫做广义线性混合模型(generalized linear mixed effects model,缩写为GLMM).这种模型在统计学里面用的很普遍.

9.6.1 样例:针对医疗数据的半参数化广义线性混合模型(semi-parametric GLMMs for medical data)

下面的例子来自(Wand 2009).假如y_{ij}是第j个病人在第i次测量的脊椎骨矿物密度(spinal bone mineral density,缩写为SBMD).设x_{ij}为这个人的年龄,设x_j为此人的种族(ethnicity),可以是白种人/亚裔/黑种人/拉丁裔.主要目标就是要判断四个种族的平均脊椎骨矿物密度是否有显著区别.数据如图9.2(b)中浅灰色线所示.其中可见脊椎骨矿物密度(SBMD)和年龄有一个非线性关系,所以可以使用一个半参数化模型(semi-parametric model)综合了线性回归和非参数化回归(Ruppert et al. 2003).另外还可以发现每个组内不同个体之间也有变化,所以要用一个混合效应模型.具体来说就是使用$\phi_1(x_{ij})=1$来计量每个人的随机效应;$\phi_2(x_{ij})=0$,因为没有其他的按照人来变化的量;$\phi_3(x_{ij})=[b_k(x_{ij})]$,其中的$b_k$是第k个样条基函数(spline basis functions)(参考本书15.4.6.2),这是用来及计数年龄的非线性效应(nonlinear effect);另外还有个计算不同种族效应的$\phi_4(x_j)=[I(x_j=w),I(x_j=a),I(x_j=b),I(x_j=h)]$.然后使用一个线性连接函数(linear link function).整个模型就是:

$$\mathrm{E}[y_{ij}|x_{ij},x_j]=\Beta_j+\alpha^T b(x_{ij})+\epsilon_{ij} \tag{9.115}$$

$$+\alpha'_w I(x_j=w)+\alpha'_a I(x_j=a)+\alpha'_b I(x_j=b)+\alpha'_h I(x_j=h) \tag{9.116}$$

其中$\epsion_{ij}\sim N(0,\sigma_y^2)$. α包含了模型中和年龄相关的非参数部分,α'包含了和种族相关的参数部分,而$\Beta_j$则包含了随着每个人j变化的随机偏移量.将这些回归系数(regression coefficients)都赋予各自的高斯先验.然后可以进行后验推导来计算$p(\alpha,\alpha',\Beta,\sigma^2|D)$(计算细节参考本书9.6.2).你喝了模型之后,可以对每个组计算预测.结果如图9.2(b)所示.还可以进行显著性检验(significance testing),以某个种群作为基准值(比如白人),来对每个种群g计算$p(\alpha_g-\alpha_w|D)$,这就跟本书5.2.3里面讲的一样了.

此处参考原书图9.2

9.6.2 计算问题

广义线性混合模型(GLMMs)的主要问题是不太好拟合,这有两个原因.首先是$p(y_{ij}|\theta)$可能和先验$p(\theta)$未必共轭,其中$\theta=\(\alpha,\Beta)$.另外一个原因是这个模型里面有两个未知层次,回归系数\theta#以及先验\eta=(\mu,\sigma)$的均值和方差.

一个方法是采用全贝叶斯方法(fully Bayesian inference methods),比如变分贝叶斯(variational Bayes,Hall et al.2011),或者马尔科夫链蒙特卡罗方法(MCMC,Gelman and Hill 2007).变分贝叶斯(VB)在本书21.5会讲到,而马尔科夫链蒙特卡罗方法(MCMC)在本书24.1.

另一种方法就是使用经验贝叶斯(empirical Bayes),在本书5.6大概讲过.在广义线性混合模型的语境下,可以使用期望最大化算法(EM algorithm,本书11.4).
其中的E步骤计算$p(\tehta|\eta,D$,而M这一步骤优化η.如果是线性回归的情况下,E步骤就可以确切进行,但一般都不用确切计算出来,使用个近似值就行了.传统方法是使用数值正交(numerical quadrature)或者蒙特卡罗方法(Breslow and Clayton 1993).更快的方法是使用变分期望最大化(variational EM),参考Braun and McAuliffe 2010提供了对多水平离散选择模型问题使用变分期望

最大化的应用案例.

在频率论统计学中,有一个拟合广义线性混合模型的流行方法叫做广义估计方程(generalized estimating equations,缩写为GEE,Hardin and Hilbe 2003).不过不推荐这个方法,因为在统计学上这个方法效率不如似然函数方法(参考本书6.4.3).另外这个方法也只能对人口参数α进行估计,而不能对随机效应$\Beta_j$进行估计,而后者可能是更需要的.

$p(y_{ij}|\theta)\ p(\theta)\ \theta = (\alpha, \Beta)\ \eta = (\mu, \sigma)$

$p(\theta|\eta, D)$

9.7 学习排序(Learning to rank)*

在本节降低是学习排序(learning to rank,缩写为LETOR)问题.也就是要学习一个函数,可以将一系列项目进行排序(后面会详细说具体内容).最常见的用途是信息检索(information retrieval).假如有一个检索查询(query)q,以及一系列文档$d^1, \ldots, d^m$,这些文档和q可能相关(比如所有文档都包含字符串q).然后我们想要对文档按照和q的相关性来降序排列,展示出其中前面k个给用户.类似问题也出现在其他领域,比如协作过滤(collaborative filtering)(在一个游戏里面对玩家进行排名或者类似的问题,具体参考本书22.5.5).

接下来简单总结一些解决这个问题的方法,这部分内容参考了(Liu 2009).这部分内容其实并不是基于广义线性模型(GLM)的,但这本书其他地方也不适合放这些内容,就放这里了,就是这么任性.

衡量文档d和查询q之间相关性的标准方法是使用一个概率语言模型(probabilistic language model),基于词汇袋模型(bag of words model).也就是定义$sim(q,d) \triangleq p(q|d) = \prod_{i=1}^{n} p(q_i|d)$,其中的$q_i$是第i个查询项目或者词汇,而$p(q_i|d)$是从文档d里面估计的一个多重伯努利分布(multinoulli distribution).实际使用中需要将这个估计分布进行光滑处理,比如使用狄利克雷先验(Dirichlet prior),来表示每个词汇的全局频率(overall frequency).这可以从系统中的全部文档来估计.具体来说就是使用下面这个表达式:

$$p(t|d) = (1-\lambda)\frac{TF(td)}{LEN(d)} + \lambda p(t|background) \quad (9.117)$$

其中$TF(t|d)$是文档d中第t项的频率,而$LEN(d)$是文档d中词汇长度,而$0 < \lambda < 1$是光滑参数(smoothing parameter,更多细节参考 Zhai and Lafferty 2004).

不过也可能还有很多其他的信号也能用来衡量相关性.比如网络文本的页面评级(PageRank)就是对其权威性(authoritativeness)的衡量,是从网页链接结构来推导的(具体参考本书17.2.4).可以计算在一个文档中某个查询项目出现的次数和位置.接下来就讲一下如何将这些信号集中起来学习使用.

9.7.1 单点法(pointwise approach)

假如我们要收集代表一系列文档对每个查询项的相关度的训练数据.具体来说就是对每个查询q,设找到了m个可能的相关文档 $d_j, \text{for} j = 1:m$.对每个查询文档对,定义一个特征向量$x(q,d)$.比如可能包含了查询项和文档的相似度排序,以及文档的页面评级分数(page rank score).然后假设有一系列的标签y_i代表的是文档d_j和查询项q之间的相关程度.这样的类标签可以十二指华东(比如是相关或不想管),或者也可以使用离散的相关程度来描述(比如很相关,有点相关,不相关).这样的类标签可以从查询项日志(query logs)获得,通过限制一个文档对于一个给定的查询项被点击的次数.

如果使用二值化相关标签,就可以使用标准二值化分类来估计$p(y=1|x(q,d))$,来解决这个问题你.如果使用排序相关标签,就可以使用有序回归(ordinal regression)$p(y=r|x(q,d))$来预测排序.这两种情况下都可以通过排序矩阵(scoring metric)来整理文档.这就叫做学习排序(LETOR)的单点法(pointwise approach),用的很广泛,因为特别简单.不过这个方法没有将文档在列表中的各自位置考虑进去.因此对列表中最末项目和最首位项目有一样的错误惩罚,这通常就不符合预期了.另外每次对相关性的判断也都非常短视(myopically).

9.7.2 成对法(pairwise approach)

Carterette et al. 2008 的研究证明人类要更擅长判断两个对象之间的相对相性,而不是绝对相关性.结果就导致数据可能告诉我们d_j比d_k和给定的查询更相关,或者反过来.可以对这种数据使用二值化分类器来进行建模,形式为$p(y_{jk}|x(q,d_j),x(q,d_k))$,当$rel(d_j,q) > rel(d_k,q)$时设$y_{jk}=1$,反之则设置$y_{jk}=0$.

对这个函数建模的一种方法如下所示:

$$p(y_{jk}=1|x_j,x_k) = sigm(f(x_j)-f(x_k)) \quad (9.118)$$

其中的$f(x)$是排序函数,一般都设置为线性函数$f(x)=w^Tx$.这是一类特殊的神经网络,叫做排序网络(RankNet,Burges et al. 2005,关于神经网络的讨论参考本书16.5).可以通过最大化对数似然函数来找到w的最大似然估计(MLE),或者也可以等价地对交叉熵损失函数(cross entropy loss)最小化,也就是:

$$L = sum_{i=1}^{N} \sum_{j=1}^{m_i} \sum_{k=j+1}^{m_i} L_{ijk} \quad (9.119)$$

$$\begin{aligned} -L_{ijk} &= I(y_{ijk}=1) \log p(y_{ijk}=1|x_{ij},x_{ik},w) \\ &+ I(y_{ijk}=0) \log p(y_{ijk}=0|x_{ij},x_{ik},w) \end{aligned} \quad (9.120)$$

这就可以使用梯度下降法来优化了.微软的Bing搜索引擎就使用了排序网络的一个变种.

9.7.3 列表法(listwise approach)

承兑发的问题就是关于相关性的决策判断只是基于一对项目或者一堆文档,而不是考虑完整的语境(context).接下来要讲的方法就要同时查看整个项目列表.

在列表上可以制定一个索引排序来定义一个全排列,π.要对关于π的不确定性建模,就可以使用一个Plackett-Luce分布,这个分布的命名就是基于两个独立推导该公式的人(Plackett 1975)和(Luce 1959).这个分布形式如下所示:

$$p(\pi|s) = \prod_{j=1}^{m} \frac{s_j}{\sum_{u=j}^{m} s_u} \text{(9.121)}$$

其中的$s_j = s(\pi^{-1}(j))$是对排在第j个位置的文档的排序.

要理解等式9.121,可以举个简单例子.设$\pi = (A, B, C)$.然后就得到了$p(\pi)$是A排第一的概率,乘以A排第一的条件下B排第二的概率,再乘以AB分别排第一第二之后C排第三未知的概率,也就是说:

$$p(\pi|s) = \frac{s_A}{s_A+s_B+s_C} \times \frac{s_B}{s_B+s_C} \times \frac{s_C}{s_C} \text{(9.122)}$$

要整合特征,可以定义$s(d) = f(x(q,d))$,通常都设f为线性函数$f(x) = w^T x$.这就叫做列表网络模型(ListNet model,Cao et al. 2007).要训练这个模型,设y_i是文档对于查询项目i的相关性分数.然后就要最小化交叉熵项(cross entropy term):

$$-\sum_i \sum_p ip(\pi|y_i) \log p(\pi|s_i) \text{(9.123)}$$

当然了,这也很困难,因为第i项目需要累加总共超过$m_i!$次排列才行.要让这个好处理,可以考虑只在前面的k个位置上进行排列:

$$p(\pi_{1:k}|s_{1:m}) = \prod_{j=1}^{k} \frac{s_j}{\sum_{u=1}^{m} s_u} \text{(9.124)}$$

这样就只需要$m!/(m-k)!$次这样的排列了.如果设置$k = 1$,那就可以在$O(m)$时间内计算每个交叉熵以及其导数了.

在列表中只有一个文档被认为相关的特例下,比如设$y_i = c$,就可以使用多项逻辑回归(multinomial logistic regressi)了:

$$p(y_i = c|x) = \frac{\exp(s_C)}{\sum_{(}^{m} c'=1) \exp(s_{c'})} \text{(9.125)}$$

这个方法至少能达到排序方法的性能水平,至少在协作过滤(collaborative filtering)的情况下如此(Yang et al. 2011).

9.7.4 排序的损失函数

对一个排序系统的性能衡量,有好几种方法,大概如下所述.

- 平均排序倒数(Mean reciprocal rank,缩写为MRR).对于一个查询项q,设其第一个相关文档的排序位置记作$r(q)$.然后定义一个平均排序倒数为$1/r(q)$.这是很简单的性能衡量.
- 均值平均准确率(Mean average precision,缩写为MAP,注意要和最大后验分布 MAP相区分).在二值化相关标签的情况下,可以定义某个排序在k位置上的精度如下:

$$P@k(\pi) \triangleq \frac{\text{num. relevant documents in the top k positions of}\pi}{k} \tag{9.126}$$

然后可以定义平均精度(average precision)如下:

$$AP(\pi) \triangleq \frac{\sum_k P@k(\pi) \times I_k}{\text{num relevant documents}} \tag{9.127}$$

其中当且仅当文档k为相关的时候I_k才等于1.例如,如果有相关标签$y = (1, 0, 1, 0, 1)$,然后AP就是 $\frac{1}{3}(\frac{1}{1} + \frac{2}{3} + \frac{3}{5}) \approx 0.76$最终就定义了均值平均精度(mean average precision)为对所有查询上的AP求平均值.

- 归一化折扣累积增益(Normalized discounted cumulative gain,缩写为NDCG).假如香瓜鸟枪有多种层次.就可以定义对前面以一定次序排列的k个项目的折扣累积增益(discounted cumulative gain)如下所示:

$$DCG@k(r) = r_1 + \sum_{i=2}^{k} \frac{r_i}{\log_2 i} \tag{9.128}$$

其中的r_i是第i项的相关性,而$\log_2$项目是用来稍后在列表中扣除项目的.表9.3展示了一个简单的数值样本.另一重定义就是强调了检索到相关文档(retrieving relevant documents),使用的是:

$$DCG@k(r) = \sum_{i=1}^{k} \frac{2^{r_i} - 1}{\log_2(1+i)} \tag{9.129}$$

折扣累积增益(DCG)的一个问题就是只要返回列表的长度变化,这个增益的数量级就会有变化.因此通常都要用理想折扣累积增益(ideal DCG)来将其归一化,理想折扣累积增益(ideal DCG)是指通过使用最优排序来得到的DCG,即$IDCG@k(r) = \arg\max_\pi DCG@k(r)$.最终就定义出来了归一化折扣累积增益(Normalized discounted cumulative gain,缩写为NDCG),定义为$DCG/IDCG$.表9.4给出了一个简单数值样本.NDCG方法可以对查询项目进行平均然后来衡量性能.

- 排序相关性(Rank correlation).可以在排序列表π和相关性判断π^*之间衡量相关性,使用的方法就很多了.比如可以使用加权肯德尔τ统计(weighted Kendall's τ statistics),这个统计定义形式为两个列表间不连续的加权值对:

$$\tau(\pi, \pi^*) = \frac{\sum_{u<v} w_{uv}[1 + sgn(\pi_u - \pi_v)sgn(\pi_u^* - \pi_v^*)]}{2\sum_{u<v} w_{uv}} \tag{9.130}$$

其他方法也都很常用.

这些损失函数可以有不同用法.在贝叶斯方法中,首先使用后验推断来拟合模型;这就一来似然函数

和先验,但不用管损失函数.然后选在测试的时候选择行为来最小化期望未来损失(expected future loss).一种方法就是从后严重对参数取样,即$\theta^s \sim p(\theta|D)$,然后评估,比如对在k精确度不同的阈值,在$\theta^s$上取平均值.这种方法的具体样例参考(Zhang et al. 2010).

在频率论方法中,在训练集上就要试图最小化经验损失函数.问题就是这些损失函数并不是模型参数的可微函数(differentiable functions).要么就要用非梯度的优化方法,要么就要使用代理损失函数来替代进行最小化.交叉熵损失函数(比如负对数似然函数)是一个广泛使用的代理损失函数.

另外一种损失函数,也叫作加权估计-排序成对损失函数(weighted approximate-rank pairwise loss,缩写为WARP loss),由(Usunier et al. 2009)提出,然后由(Weston et al. 2010)对其进行了扩展,提供了对在k精度上的损失函数的更好的估计.这个损失函数的定义如下所示:

$$WARP(f(x,:),y) \triangleq L(rank(f(x,:),y) \tag{9.131}$$

$$rank(f(x,:),y) = \sum_{y' \neq y} I(fx,y') \geq f(x,y)) \tag{9.132}$$

$$L(k) \triangleq \sum_{j=1}^{k} \alpha_j, \text{with} \alpha_1 \geq \alpha_{\geq} \ldots \geq 0 \tag{9.133}$$

上式中的$f(x,:) = [f(x,1),\ldots,f(x,|y|)]$是对每个可能输出标签的评分向量,或者在迭代重加权(IR)项目中,就是对每个对应输入查询x的每个可能文章的评分向量.表达式$rank(f(x,:),y)$衡量由该评分函数分配真实标签y的评分.最后的L是讲整数值的评分转化成实数值的惩罚项(real-valued penalty).使用$\alpha_1 = 1, \alpha_{j>1} = 0$可以优化前部位置分类标签正确的比例.设置\alpha_1:k}为非零值可以优化排序列表中的前k个项目,因为使用了均值平均准确率(MAP)或k精确度来衡量,所以性能不错.即便这样,加权估计-排序成对损失函数(weighted approximate-rank pairwise loss,缩写为WARP loss)也还是很难去优化的,但是可以使用蒙特卡罗取样方法来近似,然后再用梯度下降法去优化,这部分参考(Weston et al. 2010).

练习略