Conditional Exponential Distributions: A Worked Example

David S. Rosenberg

1 Conditional Exponential Distributions

Suppose we want to model the amount of time one will have to wait for a taxi pickup based on the location and the time. The exponential distribution is a natural candidate for this situation. The exponential distribution is a continuous distribution supported on $[0, \infty)$. The set of all exponential probability density functions is given by

$$\text{ExpDists} = \left\{ p_\lambda(y) = \lambda e^{-\lambda y} 1(y \in [0, \infty)) \mid \lambda \in (0, \infty) \right\}.$$

Let $x \in \mathbf{R}^d$ represent the input features from which we want to predict an exponential distribution. We can represent an element of ExpDists by the parameter λ.

1.1 GLM Approach

We will start with a "**generalized linear model**" (GLM) approach, in which for a given input x, we predict $\lambda = \psi(w^T x)$ for some function ψ and some parameter vector $w \in \mathbf{R}^d$.

1. In a GLM, the function ψ is chosen by the data scientist as part of the model choice. Suggest a reasonable function ψ to map $w^T x$ to λ. Then write an expression for $p_w(y \mid x)$, the predicted probability density function conditioned on x. Because of subsequent problems, you are encouraged to choose a function that is differentiable.

   Solution: Since $\lambda \in (0, \infty)$, the range of ψ should also be $(0, \infty)$. Functions that are monotonically increasing as a function of the score

$w^T x$ are preferred, as is differentiability. Thus we will choose $\psi(\cdot) = \exp(\cdot)$. The predicted probability density for a given x is

$$p_w(y \mid x) = e^{w^T x} e^{-\exp\left(w^T x\right) y}$$

for $y \geq 0$ and 0 otherwise.

2. Once ψ is chosen, $w \in \mathbf{R}^d$ is determined by maximum likelihood on a training set, say $(x_1, y_1), \ldots, (x_n, y_n)$ sampled i.i.d. $P_{\mathcal{X} \times \mathcal{Y}}$, where $x_i \in \mathbf{R}^d$ and $y_i \in [0, \infty)$ for $i = 1, \ldots, n$. Give the optimization problem you would solve to fit the GLM.

   Solution: By independence, the likelihood for the dataset is

   $$\prod_{i=1}^{n} p_w(y_i \mid x_i) \quad = \quad \prod_{i=1}^{n} e^{w^T x} e^{-\exp\left(w^T x\right) y}$$

   and the log-likelihood is

   $$J(w) = \log \left[\prod_{i=1}^{n} p_w(y_i \mid x_i) \right] \quad = \quad \sum_{i=1}^{n} \left[w^T x_i - y_i \exp\left(w^T x_i\right) \right].$$

   Maximizing the likelihood is equivalent to maximizing the log-likelihood. The optimization problem to solve is

   $$w^* = \underset{w \in \mathbf{R}^d}{\arg\max}\, J(w).$$

   It's a maximum because we want the maximum likelihood. We can also look for $\arg\min_{w \in \mathbf{R}^d} [-J(w)]$, to be back in our usual minimization setting.

3. Is $-J(w)$ convex?

   Solution: Yes. $w^T x_i$ is an affine function of w (in fact, it is linear). $\exp(\cdot)$ is a convex function. The composition of a convex function and and an affine function is convex. [You can also just remember that $\exp(f(x))$ is convex whenever $f(x)$ is.]. $y_i \geq 0$, so $y_i \exp\left(w^T x_i\right)$ is convex, and subtracting off $w^T x_i$ (a linear function) is still convex. [Since $-w^T x_i$ is also convex, we can view $y_i \exp\left(w^T x_i\right) + (-w^T x_i)$ as the sum of two convex functions. Finally, the sum over i is a nonnegative [convex] combination of convex functions, and so it's convex.

4. Give a numerical method for finding w^*. No need to specify a step size plan or a termination plan. Just give the step directions you will use.

   Solution: We'll use SGD. At each step we'll choose a random data point (x_i, y_i) and we'll take the step

$$w \leftarrow w + \eta \left[x_i - y_i \exp\left(w^T x_i\right) x_i\right],$$

   for some step size η.

1.2 GBM Approach

Suppose we are not convinced that $w^T x$ extracts enough information from x to make a good prediction of λ, and we want to use a nonlinear function of x. We can use a gradient boosting approach for this. Rather than predicting $x \mapsto \psi(w^T x)$, where w is learned from the data, we will now predict $x \mapsto \psi(f(x))$, where f is some more general function learned from the data.

1. Write our new objective function $J(f)$, where f is now the function described above.

   Solution:

$$J(f) = \sum_{i=1}^{n} \left[f(x_i) - y_i \exp\left(f(x_i)\right)\right].$$

2. We can find f using gradient boosting. Let $\mathcal{H}$ be our base hypothesis space of real-valued functions. In each step of gradient boosting, we choose a function $h \in \mathcal{H}$ that solves a particular regression problem. Give this regression problem.

   Solution: For gradient boosting, we need to compute the gradient at the datapoints. So

$$\frac{\partial}{\partial f(x_i)} J(f) = 1 - y_i \exp\left[f(x_i)\right].$$

   This is the unconstrained gradient. We want to find the best fit to the negative of this gradient direction among functions in $\mathcal{H}$. This is the following regression problem:

$$\begin{aligned} h^* &= \underset{h \in \mathcal{H}}{\arg\min} \sum_{i=1}^{n} \left[-\frac{\partial}{\partial f(x_i)} J(f) - h(x_i)\right]^2 \\ &= \underset{h \in \mathcal{H}}{\arg\min} \sum_{i=1}^{n} \left[-\left(1 - y_i \exp\left[f(x_i)\right]\right) - h(x_i)\right]^2 \end{aligned}$$

3. Give the full GBM algorithm for finding the maximum likelihood function f. No need to specify a stopping criterion. You may assume that the algorithm takes M steps, if that it makes the algorithm easier to express :

Solution:

(a) Initialize $f_0(x) = 0$.

(b) For $m = 1$ to M:

i. Compute:

$$\mathbf{g}_m = \left(1 - y_i \exp\left[f_{m-1}(x_i)\right]\right)_{i=1}^{n}$$

ii. Fit regression model to $-\mathbf{g}_m$:

$$h_m = \underset{h \in \mathcal{F}}{\arg\min} \sum_{i=1}^{n} \left(-\left(\mathbf{g}_m\right)_i - h(x_i)\right)^2 .$$

A. Choose fixed step size $\nu_m = \nu \in (0, 1]$, or take

$$\nu_m = \underset{\nu > 0}{\arg\max}\, J(f_{m-1} + \nu h_m).$$

B. Take the step:

$$f_m(x) = f_{m-1}(x) + \nu_m h_m(x)$$