Conditional Probability Models

David Rosenberg

New York University

April 5, 2017
Maximum Likelihood Recap
Maximum Likelihood Estimation

Suppose we have a parametric model \( \{ p(y; \theta) \mid \theta \in \Theta \} \) and a sample \( \mathcal{D} = \{ y_1, \ldots, y_n \} \).

Definition

The maximum likelihood estimator (MLE) for \( \theta \) in the model \( \{ p(y, \theta) \mid \theta \in \Theta \} \) is

\[
\hat{\theta} = \arg \max_{\theta \in \Theta} L_\mathcal{D}(\theta) = \arg \max_{\theta \in \Theta} \prod_{i=1}^{n} p(y_i; \theta).
\]

In practice, we prefer to work with the log likelihood. Same maximum but

\[
\log p(\mathcal{D}; \theta) = \sum_{i=1}^{n} \log p(y_i; \theta),
\]

and sums are easier to work with than products.
Maximum Likelihood Estimation

- Finding the MLE is an optimization problem.
- For some model families, calculus gives closed form for MLE.
- Can also use numerical methods we know (e.g. SGD).
- Note: In certain situations, the MLE may not exist.
  - But there is usually a good reason for this.
- e.g. Gaussian family $\{N(\mu, \sigma^2 \mid \mu \in \mathbb{R}, \sigma^2 > 0]\}$, Single observation $y$.
  - Take $\mu = y$ and $\sigma^2 \to 0$ drives likelihood to infinity. MLE doesn’t exist.
Bernoulli Regression
Bernoulli Regression

Probabilistic Binary Classifiers

- Setting: $X = \mathbb{R}^d$, $Y = \{0, 1\}$
- For each $x$, need to predict a distribution on $Y = \{0, 1\}$.
- What kind of parametric distribution could be supported on $\{0, 1\}$?
- Not a lot of choices....
- Bernoulli!
- For each $x$, 
  - predict the Bernoulli parameter $\theta = p(y = 1 \mid x)$.
## Linear Probabilistic Classifiers

- **Setting**: $X = \mathbb{R}^d$, $Y = \{0, 1\}$
- **Want prediction function** $x \mapsto \theta = p(y = 1 \mid x)$.
- **We need** $\theta \in [0, 1]$.
- **For a “linear method”**, we can write this in two steps:

$$
\begin{align*}
  x &\in \mathbb{R}^d \\
  w^T x &\in \mathbb{R} \\
  f(w^T x) &\in [0, 1]
\end{align*}
$$

where $f : \mathbb{R} \rightarrow [0, 1]$ is called the **transfer** or **inverse link** function.
- **Probability model** is then

$$
p(y = 1 \mid x) = f(w^T x)
$$
Inverse Link Functions

- Two commonly used “inverse link” functions to map from $w^T x$ to $\theta$:

- Logistic function $\Rightarrow$ Logistic Regression
- Normal CDF $\Rightarrow$ Probit Regression
Learning

- $\mathcal{X} = \mathbb{R}^d$
- $\mathcal{Y} = \{0, 1\}$
- $\mathcal{A} = \{0, 1\}$ (Representing Bernoulli($\theta$) distributions by $\theta \in [0, 1]$)
- $\mathcal{H} = \{ x \mapsto f(w^T x) | w \in \mathbb{R}^d \}$
- We can choose $w$ using maximum likelihood...
Bernoulli Regression: Likelihood Scoring

- Suppose we have data \( D = \{(x_1, y_1), \ldots, (x_n, y_n)\}, \text{ iid.} \)
- Compute the model likelihood for \( D \):

\[
p_w(D) = \prod_{i=1}^{n} p_w(y_i \mid x_i) \quad [\text{by independence}]
= \prod_{i=1}^{n} \left[ f(w^T x_i) \right]^{y_i} \left[ 1 - f(w^T x_i) \right]^{1-y_i}.
\]

- Huh? Remember \( y_i \in \{0, 1\} \).
- Easier to work with the log-likelihood:

\[
\log p_w(D) = \sum_{i=1}^{n} y_i \log f(w^T x_i) + (1 - y_i) \log \left[ 1 - f(w^T x_i) \right]
\]
Bernoulli Regression: MLE

- Maximum Likelihood Estimation (MLE) finds $w$ maximizing $\log p_w(D)$.
- Equivalently, minimize the objective function

$$J(w) = -\left[ \sum_{i=1}^{n} y_i \log f(w^T x_i) + (1 - y_i) \log [1 - f(w^T x_i)] \right]$$

- For differentiable $f$,
  - $J(w)$ is differentiable, and we can use our standard tools.
- Homework: Derive the SGD step directions for logistic regression.
Multinomial Logistic Regression
Multinomial Logistic Regression

- Setting: $\mathcal{X} = \mathbb{R}^d$, $\mathcal{Y} = \{1, \ldots, k\}$
- The numbers $(\theta_1, \ldots, \theta_k)$ where $\sum_{c=1}^k \theta_c = 1$ represent a “multinoulli” or “categorical” distribution.
- For each $x$, we want to produce a distribution on the $k$ classes.
- That is, for each $x$ and each $y \in \{1, \ldots, y\}$, we want to produce a probability

$$p(y \mid x) = \theta_y,$$

where $\sum_{y=1}^K \theta_y = 1$. 
Multinomial Logistic Regression: Classic Setup

- From each $x$, we compute a linear score function for each class:
  $$x \mapsto (\langle w_1, x \rangle, \ldots, \langle w_k, x \rangle) \in \mathbb{R}^k$$

- We need to map this $\mathbb{R}^k$ vector into a probability vector.

- Use the **softmax function**:
  $$(\langle w_1, x \rangle, \ldots, \langle w_k, x \rangle) \mapsto \left( \frac{\exp(w_1^T x)}{\sum_{c=1}^{K} \exp(w_c^T x)}, \ldots, \frac{\exp(w_k^T x)}{\sum_{c=1}^{K} \exp(w_c^T x)} \right)$$

- If $\theta \in \mathbb{R}^k$ is the output of the softmax, note that
  $$\theta_i > 0$$
  $$\sum_{i=1}^{k} \theta_i = 1$$
Multinomial Logistic Regression: Classic Setup

- Putting this together, we write multinomial logistic regression as

\[
p(y \mid x) = \frac{\exp(w_y^T x)}{\sum_{c=1}^{K} \exp(w_c^T x)},
\]

where we’ve introduced parameter vectors \( w_1, \ldots, w_k \in \mathbb{R}^d \).

- Can view \( x \mapsto w_y^T x \) as the score for class \( y \), for \( y \in \{1, \ldots, k\} \).

- We can also “flatten” this as we did for multiclass classification.
  - Introduce a class-sensitive feature vector \( \Psi(x, y) \in \mathbb{R}^{dk} \)
  - Parameter vector \( w \in \mathbb{R}^{dk} \).

- The log of this likelihood is concave and straightforward to optimize.
Poisson Regression
Poisson Regression: Setup

- Input space $\mathcal{X} = \mathbb{R}^d$, Output space $\mathcal{Y} = \{0, 1, 2, 3, 4, \ldots\}$

- Hypothesis space consists of functions $f : x \mapsto \text{Poisson}(\lambda(x))$.
  - That is, for each $x$, $f(x)$ returns a Poisson with mean $\lambda(x) \in (0, \infty)$.
  - What function?

- Recall $\lambda > 0$.

- In Poisson regression, $x$ enters \textbf{linearly}: $x \mapsto w^T x \mapsto \lambda = f(w^T x)$.

- Standard approach is to take
  \[ \lambda(x) = \exp(w^T x), \]
  for some parameter vector $w$.

- Note that range of $\lambda(x) = (0, \infty)$, (appropriate for the Poisson parameter).
Suppose we have data \( D = \{(x_1, y_1), \ldots, (x_n, y_n)\} \).

Recall the log-likelihood for Poisson is:

\[
\log p(D, \lambda) = \sum_{i=1}^{n} \left[ y_i \log \lambda - \lambda - \log(y_i!) \right]
\]

Plugging in \( \lambda(x) = \exp(w^T x) \), we get

\[
\log p(D, \lambda) = \sum_{i=1}^{n} \left[ y_i \log \left(\exp(w^T x_i)\right) - \exp(w^T x_i) - \log(y_i!) \right]
\]

\[
= \sum_{i=1}^{n} \left[ y_i w^T x_i - \exp(w^T x_i) - \log(y_i!) \right]
\]

Maximize this w.r.t. \( w \) to find the Poisson regression.

No closed form for optimum, but it’s concave, so easy to optimize.
Poisson Regression Example

e.g. Phone call counts per day for a startup company over 300 days.
What About Nonlinear Score Functions
Poisson Count Example

\[ y = f(x) \]

David Rosenberg  (New York University)
Let’s Use Gradient Boosting

- Recall the log-likelihood for Poisson regression

\[ \log p(\mathcal{D}, \lambda) = \sum_{i=1}^{n} [y_i w^T x_i - \exp(w^T x_i) - \log(y_i!)] \]

- Let’s replace \( w^T x \) by a general function \( f(x) \):

\[ J(f) = \sum_{i=1}^{n} [y_i f(x_i) - \exp(f(x_i)) - \log(y_i!)] \]
Generalized Regression
Generalized Regression as Statistical Learning

- **Input space** $\mathcal{X}$
- **Output space** $\mathcal{Y}$
- All pairs $(x, y)$ are independent with distribution $P_{\mathcal{X} \times \mathcal{Y}}$.
- **Action space** $\mathcal{A} = \{p(y) \mid p \text{ is a probability density or mass function on } \mathcal{Y}\}$.
- Hypothesis spaces contain decision functions $f : \mathcal{X} \rightarrow \mathcal{A}$.
  - Given an $x \in \mathcal{X}$, predict a probability distribution $p(y)$ on $\mathcal{Y}$. 

Hypothesis spaces contain decision functions $f : X \rightarrow A$.
- Given an $x \in X$, predict a probability distribution $p(y)$ on $Y$.

Let $f$ be a decision function.
- In regression, $f(x) \in \mathbb{R}$
- In hard classification, $f(x) \in \{-1, 1\}$
- For generalized regression, $f(x) \in \mathbb{R}$?

$f(x)$ is a PDF or PMF on $Y$.
- If $p = f(x)$, can evaluate $p(y)$ for predicted probability of $y$.
- Or just write $[f(x)](y)$ or even $f(x)(y)$. 

---

**A Note on Notation**
Generalized Regression as Statistical Learning

- The risk of decision function \( f : \mathcal{X} \rightarrow \mathcal{A} \)

\[
R(f) = -\mathbb{E}_{x,y} \log[f(x)](y),
\]

where \( f(x) \) is a PDF or PMF on \( \mathcal{Y} \), and we’re evaluating it on \( \mathcal{Y} \).

- The empirical risk of \( f \) for a sample \( D = \{y_1, \ldots, y_n\} \in \mathcal{Y} \) is

\[
\hat{R}(f) = -\sum_{i=1}^{n} \log[f(x_i)](y_i).
\]

This is called the negative conditional log-likelihood.
How General A Distribution Can We Use?
Can't use it in GBM: likelihood not differentiable (not continuous).

Uniform Example?