

# Conditional Probability Models

David S. Rosenberg

New York University

February 27, 2018

# Contents

- 1 Overview and Disclaimer
- 2 Modeling Conditional Distributions
- 3 Bernoulli Regression
- 4 Poisson Regression
- 5 Conditional Gaussian Regression
- 6 Multinomial Logistic Regression
- 7 Maximum Likelihood as ERM

## Overview and Disclaimer

# Linear Probabilistic Models vs GLMs

- Today we'll be talking about **linear probabilistic models**.
- Most books and software libraries related to this topic are actually about
  - **generalized linear models** (GLMs).
- GLMs are a special case of what we're talking about today.
- They're "special" because
  - they're a restriction of our setting
  - there are theorems for GLMs that we don't have in our more general setting
- However, a full development of GLMs requires a fair bit of additional machinery.
  - In particular, **exponential families** – a topic from intermediate statistics courses.
- Exponential families are wonderful, but I don't believe they're worth the payoff at this level.
- For practical purposes, our development will be more than sufficient (and simpler).

# Modeling Conditional Distributions

# Conditional Distribution Estimation (Generalized Regression)

- Given  $x$ , predict *probability distribution*  $p(y)$
- How do we represent the probability distribution?
- We'll consider *parametric families* of distributions.
  - distribution represented by parameter vector
- Examples:
  - 1 Logistic regression (Bernoulli distribution)
  - 2 Probit regression (Bernoulli distribution)
  - 3 Poisson regression (Poisson distribution)
  - 4 Linear regression (Normal distribution, fixed variance)
  - 5 Generalized Linear Models (GLM) (encompasses all of the above)
  - 6 Generalized Additive Models (GAM) (popular in statistics community)
  - 7 Gradient Boosting Machines (GBM) / AnyBoost [in a few weeks]
  - 8 Almost all neural network models used in practice (though this is not their essential feature)

# Bernoulli Regression

# Probabilistic Binary Classifiers

- Setting:  $\mathcal{X} = \mathbf{R}^d$ ,  $\mathcal{Y} = \{0, 1\}$
- For each  $x$ , need to predict a distribution on  $\mathcal{Y} = \{0, 1\}$ .
- How can we define a distribution supported on  $\{0, 1\}$ ?
- Sufficient to specify the **Bernoulli parameter**  $\theta = p(y = 1)$ .
- We can refer to this distribution as  $\text{Bernoulli}(\theta)$ .



# Linear Probabilistic Classifiers

- Setting:  $\mathcal{X} = \mathbf{R}^d$ ,  $\mathcal{Y} = \{0, 1\}$
- Want prediction function to map each  $x \in \mathbf{R}^d$  to  $\theta \in [0, 1]$ .
- We first **extract information** from  $x \in \mathbf{R}^d$  and summarize in a single number.
  - That number is analogous to the **score** in classification.
- For a **linear method**, this extraction is done with a linear function:

$$\underbrace{x}_{\in \mathbf{R}^d} \mapsto \underbrace{w^T x}_{\in \mathbf{R}}$$

- As usual,  $x \mapsto w^T x$  will include affine functions if we include a constant feature in  $x$ .
- $w^T x$  is called the **linear predictor**.
- Still need to map this to  $[0, 1]$ .

# The Transfer Function

- Need a function to map the linear predictor in  $\mathbf{R}$  to  $[0, 1]$ :

$$\underbrace{x}_{\in \mathbf{R}^d} \mapsto \underbrace{w^T x}_{\in \mathbf{R}} \mapsto \underbrace{f(w^T x)}_{\in [0,1]} = \theta,$$

where  $f: \mathbf{R} \rightarrow [0, 1]$ . We'll call  $f$  the **transfer** function.

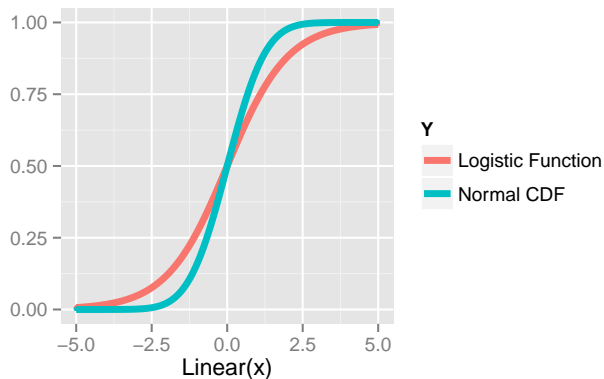
- So prediction function is  $x \mapsto f(w^T x)$ .

## Terminology Alert

In generalized linear models (GLMs), if  $\theta$  is the distribution mean, then  $f$  is called the **response function** or **inverse link** function. We avoid that terminology, since we do not require  $\theta$  to be the distribution mean.

# Transfer Functions for Bernoulli

- Two commonly used transfer functions to map from  $w^T x$  to  $\theta$ :



- Logistic function:  $f(\eta) = \frac{1}{1+e^{-\eta}} \implies$  Logistic Regression
- Normal CDF  $f(\eta) = \int_{-\infty}^{\eta} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \implies$  Probit Regression

- Input space  $\mathcal{X} = \mathbf{R}^d$
- Outcome space  $\mathcal{Y} = \{0, 1\}$
- Action space  $\mathcal{A} = [0, 1]$  (Representing Bernoulli( $\theta$ ) distributions by  $\theta \in [0, 1]$ )
- Hypothesis space  $\mathcal{F} = \{x \mapsto f(w^T x) \mid w \in \mathbf{R}^d\}$
- Parameter space  $\mathbf{R}^d$  (Each prediction function represented by  $w \in \mathbf{R}^d$ .)
- We can choose  $w$  using maximum likelihood...

## Bernoulli Regression: Likelihood Scoring Example

- Suppose we have  $\mathcal{X} = \mathbf{R}$  and data  $\mathcal{D}$ :  $(-3, 0), (0, 0), (1, 1), (2, 0) \in \mathbf{R} \times \{0, 1\}$
- Our model is  $p(y = 1 | x) = f(wx)$ , for some parameter  $w \in \mathbf{R}$ .
- Compute the likelihood for each observation:

$x$	$y$	$wx$	$\theta = f(wx)$	$\hat{p}(y)$
-3	0	$-3w$	$f(-3w)$	$1 - f(-3w)$
0	0	0	$f(0)$	$1 - f(0)$
1	1	$w$	$f(w)$	$f(w)$
2	0	$2w$	$f(2w)$	$1 - f(2w)$

- The likelihood of  $w$  for the data  $\mathcal{D}$  is

$$p(\mathcal{D}; w) = [1 - f(-3w)] \cdot [1 - f(0)] \cdot [f(w)] \cdot [1 - f(2w)]$$

- The MLE  $\hat{w}$  is the  $w \in \mathbf{R}$  maximizing  $p(\mathcal{D}; w)$  for the given  $\mathcal{D}$ .

## A Clever Way To Write $\hat{p}(y | x; w)$

- For a given  $x, w \in \mathbf{R}^d$  and  $y \in \{0, 1\}$ , the likelihood of  $w$  for  $(x, y)$  is

$$p(y | x; w) = \begin{cases} f(w^T x) & y = 1 \\ 1 - f(w^T x) & y = 0 \end{cases}$$

- It will be convenient to write this as

$$p(y | x; w) = [f(w^T x)]^y [1 - f(w^T x)]^{1-y},$$

which is obvious as long as you remember  $y \in \{0, 1\}$ .

# Bernoulli Regression: Likelihood Scoring

- Suppose we have data  $\mathcal{D} : (x_1, y_1), \dots, (x_n, y_n) \in \mathbf{R}^d \times \{0, 1\}$ .
- The likelihood of  $w \in \mathbf{R}^d$  for data  $\mathcal{D}$  is

$$\begin{aligned} p(\mathcal{D}; w) &= \prod_{i=1}^n p(y_i | x_i; w) \text{ [by independence]} \\ &= \prod_{i=1}^n [f(w^T x_i)]^{y_i} [1 - f(w^T x_i)]^{1-y_i}. \end{aligned}$$

- Easier to work with the log-likelihood:

$$\log p(\mathcal{D}; w) = \sum_{i=1}^n (y_i \log f(w^T x_i) + (1 - y_i) \log [1 - f(w^T x_i)])$$

# Bernoulli Regression: MLE

- Maximum Likelihood Estimation (MLE) finds  $w$  maximizing  $\log p(\mathcal{D}, w)$ .
- Equivalently, minimize the **negative log-likelihood** 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.
- Possible Homework: Derive the SGD step directions for logistic regression and [harder] probit regression.



# Poisson Regression

# Poisson Regression: Setup

- Input space  $\mathcal{X} = \mathbf{R}^d$ , Output space  $\mathcal{Y} = \{0, 1, 2, 3, 4, \dots\}$
- In Poisson regression, prediction functions produce a Poisson distribution.
  - Represent  $\text{Poisson}(\lambda)$  distribution by the mean parameter  $\lambda \in (0, \infty)$ .
- Action space  $\mathcal{A} = (0, \infty)$
- In Poisson regression,  $x$  enters **linearly**:  $x \mapsto \underbrace{w^T x}_{\mathbf{R}} \mapsto \lambda = \underbrace{f(w^T x)}_{(0, \infty)}$ .
- What can we use as the transfer function  $f : \mathbf{R} \rightarrow (0, \infty)$ ?

# Poisson Regression: Transfer Function

- In Poisson regression,  $x$  enters **linearly**:

$$x \mapsto \underbrace{w^T x}_{\mathbf{R}} \mapsto \lambda = \underbrace{f(w^T x)}_{(0, \infty)}.$$

- Standard approach is to take

$$f(w^T x) = \exp(w^T x).$$

- Note that range of  $f(w^T x) \in (0, \infty)$ , (appropriate for the Poisson parameter).

# Poisson Regression: Likelihood Scoring

- Suppose we have data  $\mathcal{D} = \{(x_1, y_1), \dots, (x_n, y_n)\}$ .
- Recall the log-likelihood for Poisson parameter  $\lambda_i$  on observation  $y_i$  is:

$$\log p(y_i; \lambda_i) = [y_i \log \lambda_i - \lambda_i - \log(y_i!)]$$

- Now we want to predict a different  $\lambda_i$  for every  $x_i$  with the model

$$\lambda_i = f(w^T x_i) = \exp(w^T x_i).$$

- The likelihood for  $w$  on the full dataset  $\mathcal{D}$  is

$$\begin{aligned} \log p(\mathcal{D}; w) &= \sum_{i=1}^n [y_i \log [\exp(w^T x_i)] - \exp(w^T x_i) - \log(y_i!)] \\ &= \sum_{i=1}^n [y_i w^T x_i - \exp(w^T x_i) - \log(y_i!)] \end{aligned}$$

# Poisson Regression: MLE

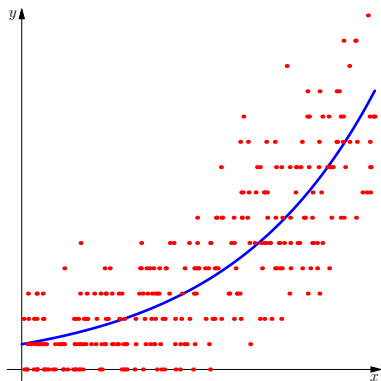
- To get MLE, need to maximize

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

over  $w \in \mathbf{R}^d$ .

- No closed form for optimum, but it's concave, so easy to optimize.

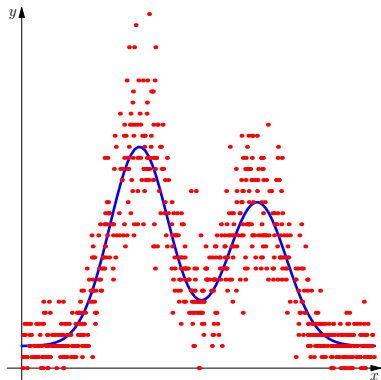
# Poisson Regression Example



- Example application: Phone call counts per day for a startup company, over 300 days.
- Blue line is mean  $\mu(x) = \exp(wx)$ , some  $w \in \mathbf{R}$ . (Only linear part  $x \mapsto wx$  is learned.)
- Samples are  $y_i \sim \text{Poisson}(wx_i)$ .

Plot courtesy of Brett Bernstein.

# Nonlinear Score Function: Sneak Preview



- Blue line is mean  $\mu(x) = \exp(f(x))$ , for some nonlinear  $f$  learned from data.
- Samples are  $y_i \sim \text{Poisson}(\exp(f(x_i)))$ .
- We can do this with gradient boosting and neural networks, coming up in a few weeks.

Plot courtesy of Brett Bernstein.

# Conditional Gaussian Regression



# Gaussian Linear Regression

- Input space  $\mathcal{X} = \mathbf{R}^d$ , Output space  $\mathcal{Y} = \mathbf{R}$
- In Gaussian regression, prediction functions produce a distribution  $\mathcal{N}(\mu, \sigma^2)$ .
  - Assume  $\sigma^2$  is known.
- Represent  $\mathcal{N}(\mu, \sigma^2)$  by the mean parameter  $\mu \in \mathbf{R}$ .
- Action space  $\mathcal{A} = \mathbf{R}$
- In Gaussian linear regression,  $x$  enters **linearly**:  $x \mapsto \underbrace{w^T x}_{\mathbf{R}} \mapsto \mu = \underbrace{f(w^T x)}_{\mathbf{R}}$ .
- Since  $\mu \in \mathbf{R}$ , we can take the identity transfer function:  $f(w^T x) = w^T x$ .

# Gaussian Regression: Likelihood Scoring

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

$$p(\mathcal{D}; w) = \prod_{i=1}^n p(y_i | x_i; w) \text{ [by independence]}$$

- Maximum Likelihood Estimation (MLE) finds  $w$  maximizing  $\hat{p}(\mathcal{D}; w)$ .
- Equivalently, maximize the data log-likelihood:

$$w^* = \arg \max_{w \in \mathbf{R}^d} \sum_{i=1}^n \log p(y_i | x_i; w)$$

- Let's start solving this!

# Gaussian Regression: MLE

- The conditional log-likelihood is:

$$\begin{aligned} & \sum_{i=1}^n \log p(y_i | x_i; w) \\ &= \sum_{i=1}^n \log \left[ \frac{1}{\sigma\sqrt{2\pi}} \exp \left( -\frac{(y_i - w^T x_i)^2}{2\sigma^2} \right) \right] \\ &= \underbrace{\sum_{i=1}^n \log \left[ \frac{1}{\sigma\sqrt{2\pi}} \right]}_{\text{independent of } w} + \sum_{i=1}^n \left( -\frac{(y_i - w^T x_i)^2}{2\sigma^2} \right) \end{aligned}$$

- MLE is the  $w$  where this is maximized.
- Note that  $\sigma^2$  is irrelevant to finding the maximizing  $w$ .
- Can drop the negative sign and make it a minimization problem.

- The MLE is

$$w^* = \arg \min_{w \in \mathbf{R}^d} \sum_{i=1}^n (y_i - w^T x_i)^2$$

- This is exactly the objective function for least squares.
- From here, can use usual approaches to solve for  $w^*$  (SGD, linear algebra, calculus, etc.)

# Multinomial Logistic Regression

---

# Multinomial Logistic Regression

- Setting:  $\mathcal{X} = \mathbf{R}^d$ ,  $\mathcal{Y} = \{1, \dots, k\}$
- For each  $x$ , we want to produce a distribution on  $k$  classes.
- Such a distribution is called a “**multinoulli**” or “**categorical**” distribution.
- Represent categorical distribution by probability vector  $\theta = (\theta_1, \dots, \theta_k) \in \mathbf{R}^k$ :
  - $\sum_{i=1}^k \theta_i = 1$  and  $\theta_i \geq 0$  for  $i = 1, \dots, k$  (i.e.  $\theta$  represents a **distribution**) and
- So  $\forall y \in \{1, \dots, k\}$ ,  $p(y) = \theta_y$ .

# Multinomial Logistic Regression

- From each  $x$ , we compute a linear score function for each class:

$$x \mapsto (\langle w_1, x \rangle, \dots, \langle w_k, x \rangle) \in \mathbf{R}^k,$$

where we've introduced parameter vectors  $w_1, \dots, w_k \in \mathbf{R}^d$ .

- We need to map this  $\mathbf{R}^k$  vector of scores into a probability vector.
- Consider the **softmax function**:

$$(s_1, \dots, s_k) \mapsto \theta = \left( \frac{e^{s_1}}{\sum_{i=1}^k e^{s_i}}, \dots, \frac{e^{s_k}}{\sum_{i=1}^k e^{s_i}} \right).$$

- Note that  $\theta \in \mathbf{R}^k$  and

$$\begin{aligned} \theta_i &> 0 & i = 1, \dots, k \\ \sum_{i=1}^k \theta_i &= 1 \end{aligned}$$

# Multinomial Logistic Regression

- Say we want to get the predicted categorical distribution for a given  $x \in \mathbf{R}^d$ .
- First compute the scores ( $\in \mathbf{R}^k$ ) and then their softmax:

$$x \mapsto (\langle w_1, x \rangle, \dots, \langle w_k, x \rangle) \mapsto \theta = \left( \frac{\exp(w_1^T x)}{\sum_{i=1}^k \exp(w_i^T x)}, \dots, \frac{\exp(w_k^T x)}{\sum_{i=1}^k \exp(w_i^T x)} \right)$$

- We can write the conditional probability for any  $y \in \{1, \dots, k\}$  as

$$p(y | x; w) = \frac{\exp(w_y^T x)}{\sum_{i=1}^k \exp(w_i^T x)}.$$



# Multinomial Logistic Regression

- Putting this together, we write multinomial logistic regression as

$$p(y \mid x; w) = \frac{\exp(w_y^T x)}{\sum_{i=1}^k \exp(w_i^T x)}.$$

- How do we do learning here? What parameters are we estimating?
- Our model is specified once we have  $w_1, \dots, w_k \in \mathbf{R}^d$ .
- Find parameter settings maximizing the log-likelihood of data  $\mathcal{D}$ .
- This objective function is concave in  $w$ 's and straightforward to optimize.

## Maximum Likelihood as ERM

# Conditional Probability Modeling as Statistical Learning

- Input space  $\mathcal{X}$
- Outcome 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 space  $\mathcal{F}$  contains decision functions  $f : \mathcal{X} \rightarrow \mathcal{A}$ .
- Maximum likelihood estimation for dataset  $\mathcal{D} = ((x_1, y_1), \dots, (x_n, y_n))$  is

$$\hat{f}_{\text{MLE}} \in \arg \max_{f \in \mathcal{F}} \sum_{i=1}^n \log [f(x_i)(y_i)]$$

## Exercise

Write the MLE optimization as empirical risk minimization. What's the loss?

# Conditional Probability Modeling as Statistical Learning

- Take loss  $\ell : \mathcal{A} \times \mathcal{Y} \rightarrow \mathbf{R}$  for a predicted PDF or PMF  $p(y)$  and outcome  $y$  to be

$$\ell(p, y) = -\log p(y)$$

- The risk of decision function  $f : \mathcal{X} \rightarrow \mathcal{A}$  is

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

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

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

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

This is called the negative **conditional log-likelihood**.

- Thus for the negative log-likelihood loss, ERM and MLE are equivalent