Review: MLE and GLM

Julia Kempe & David S. Rosenberg

CDS, NYU

March 5, 2019

Contents

Maximum Likelihood

Maximum Likelihood Estimation

- Suppose $\mathcal{D} = (y_1, \ldots, y_n)$ is an i.i.d. sample from some distribution.

Definition

A **maximum likelihood estimator (MLE)** for θ in the model $\{p(y;\theta) \mid \theta \in \Theta\}$ is

$$\begin{aligned}
\hat{\theta} &\in \operatorname*{arg\,max}_{\theta \in \Theta} \log p(\mathcal{D}, \hat{\theta}) \\
&= \operatorname*{arg\,max}_{\theta \in \Theta} \sum_{i=1}^{n} \log p(y_i;\theta).
\end{aligned}$$

Maximum Likelihood Estimation

- Finding the MLE is an **optimization problem**.
- For some model families, calculus gives a closed form for the MLE.
- Can also use numerical methods we know (e.g. SGD).

MLE Existence

- In certain situations, the MLE may not exist.
- But there is usually a good reason for this.
- e.g. Gaussian family $\{\mathcal{N}(\mu, \sigma^2) \mid \mu \in \mathbf{R}, \sigma^2 > 0\}$
- We have a single observation y.
- Is there an MLE?
- Taking $\mu = y$ and $\sigma^2 \to 0$ drives likelihood to infinity.
- MLE doesn't exist.

Example: MLE for Poisson

- Observed counts $\mathcal{D} = (k_1, \ldots, k_n)$ for taxi cab pickups over n weeks.
  - k_i is number of pickups at Penn Station Mon, 7-8pm, for week i.
- We want to fit a Poisson distribution to this data.
- The Poisson log-likelihood for a single count is

$$
\begin{aligned}
\log\left[p(k;\lambda)\right] &= \log\left[\frac{\lambda^k e^{-\lambda}}{k!}\right] \\
&= k\log\lambda - \lambda - \log(k!)
\end{aligned}
$$

- The full log-likelihood is

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

Example: MLE for Poisson

- The full log-likelihood is

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

- First order condition gives

$$\begin{aligned} 0 = \frac{\partial}{\partial\lambda}\left[\log p(\mathcal{D},\lambda)\right] &= \sum_{i=1}^{n}\left[\frac{k_i}{\lambda} - 1\right] \\ \implies \lambda &= \frac{1}{n}\sum_{i=1}^{n} k_i \end{aligned}$$

- So MLE $\hat{\lambda}$ is just the mean of the counts.

Estimating Distributions, Overfitting, and Hypothesis Spaces

- Just as in classification and regression, MLE can overfit!
- Example Probability Models:
  - $\mathcal{F} = \{$Poisson distributions$\}$.
  - $\mathcal{F} = \{$Negative binomial distributions$\}$.
  - $\mathcal{F} = \{$Histogram with 10 bins$\}$
  - $\mathcal{F} = \{$Histogram with bin for every $y \in \mathcal{Y}\}$ [will likely overfit for continuous data]
- How to judge which model works the best?
- Choose the model with the **highest likelihood on validation set**.

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 $[0,1]$:

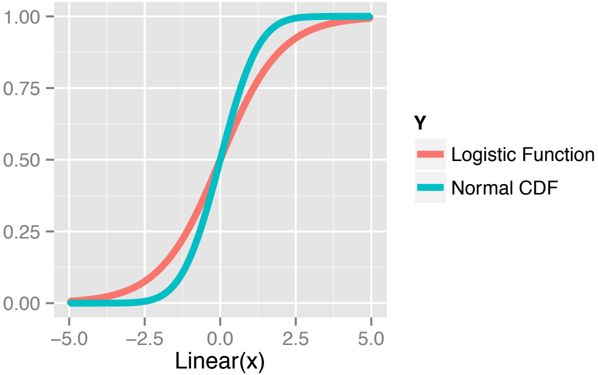
$$\underbrace{x}_{\in \mathbf{R}^d} \mapsto \underbrace{w^T x}_{\in \mathbf{R}} \mapsto \underbrace{f(w^T x)}_{\in [0,1]} = \theta,$$

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

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

Transfer Functions for Bernoulli

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



- 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

Learning

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

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

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

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

- It will be convenient to write this as

$$p(y \mid x; w) = \left[f(w^T x)\right]^y \left[1 - f(w^T x)\right]^{1-y},$$

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

Bernoulli Regression: Likelihood Scoring

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

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

- Easier to work with the log-likelihood:

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

Bernoulli Regression: MLE

- Maximum Likelihood Estimation (MLE) finds w maximizing $\log p(\mathcal{D}, w)$.
- Equivalently, minimize the **negative log-likelihood** objective function

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

- For differentiable f,
  - $J(w)$ is differentiable, and we can use SGD.
  - What guarantees us to find the global minima of $J(w)$ by SGD?
  - Convexity of $J(w)$!

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 Poisson(λ) 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 $f : \mathbf{R} \to (0, \infty)$?

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

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

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

Poisson Regression: Likelihood Scoring

- Suppose we have data $\mathcal{D} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$.
- Recall the log-likelihood for Poisson parameter λ_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 λ_i for every x_i with the model

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

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

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

Poisson Regression: MLE

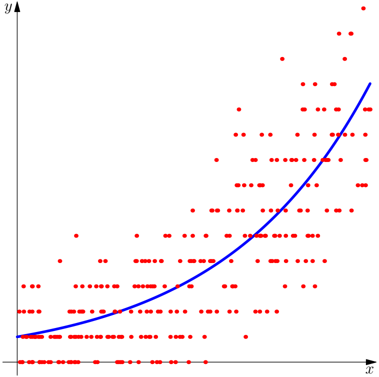
- To get MLE, need to maximize

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

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

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

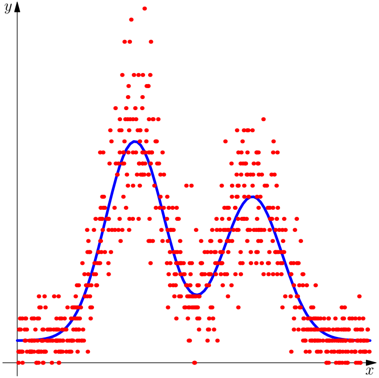
Poisson Regression Example



- 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 σ^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 $\mathcal{D} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$.
- Compute the model likelihood for $\mathcal{D}$:

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

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

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

- Let's start solving this!

Gaussian Regression: MLE

- The conditional log-likelihood is:

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

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

Gaussian Regression: MLE

- The MLE is

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

- 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: $\mathcal{X} = \mathbf{R}^d$, $\mathcal{Y} = \{1, \ldots, k\}$
- 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 $\theta = (\theta_1, \ldots, \theta_k) \in \mathbf{R}^k$:
    - $\sum_{i=1}^{k} \theta_i = 1$ and $\theta_i \geqslant 0$ for $i = 1, \ldots, k$ (i.e. θ represents a **distribution**) and
- So $\forall y \in \{1, \ldots, k\}$, $p(y) = \theta_y$.

Multinomial Logistic Regression

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

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

  where we've introduced parameter vectors $w_1, \ldots, w_k \in \mathbf{R}^d$.
- We need to map this $\mathbf{R}^k$ vector of scores into a probability vector.
- Consider the **softmax function**:

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

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

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

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:

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

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

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

Multinomial Logistic Regression

- Putting this together, we write multinomial logistic regression as

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

- How do we do learning here? What parameters are we estimating?
- Our model is specified once we have $w_1, \ldots, w_k \in \mathbf{R}^d$.
- 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** $\mathcal{A} = \{p(y) \mid p$ is a probability density or mass function on $\mathcal{Y}\}$.
- Hypothesis space $\mathcal{F}$ contains decision functions $f : \mathcal{X} \to \mathcal{A}$.
- Maximum likelihood estimation for dataset $\mathcal{D} = ((x_1, y_1), \ldots, (x_n, y_n))$ is

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

Conditional Probability Modeling as Statistical Learning

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

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

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

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

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

Conditional Probability Modeling as Statistical Learning

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

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

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

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

Review Questions

Maximum Likelihood

1. Suppose we have samples $x_1, \ldots, x_n$ i.i.d drawn from Bernoulli(p). Find the maximum likelihood estimator of p.
2. Suppose we have samples $x_1, \ldots, x_n$ i.i.d drawn from uniform distribution $\mathcal{U}(a, b)$. Find the maximum likelihood estimator of a and b.

Maximum Likelihood

- Suppose we have samples $x_1, \ldots, x_n$ i.i.d drawn from Bernoulli(p). Find the maximum likelihood estimator of p.

  Solution:

  - The likelihood is:

  $$L(p) = \Pi_{i=1}^{n} p^{x_i} (1-p)^{(1-x_i)}.$$

  - The log-likelihood is:

  $$\ell(p) = \log p \sum_{i=1}^{n} x_i + \log(1-p) \sum_{i=1}^{n} (1-x_i).$$

  - Set the derivative of log-likelihood w.r.t. p to zero:

  $$\frac{\partial \ell(p)}{\partial p} = \frac{\sum_{i=1}^{n} x_i}{p} - \frac{\sum_{i=1}^{n} (1-x_i)}{1-p} = 0.$$

Maximum Likelihood

- Solving the equation above, we have:

$$p = \frac{1}{n}\sum_{i=1}^{n} x_i.$$

- The second derivative of log-likelihood w.r.t. p is

$$\frac{\partial^2 \ell(p)}{\partial p^2} = \frac{-\sum_{i=1}^{n} x_i}{p^2} - \frac{\sum_{i=1}^{n}(1-x_i)}{(1-p)^2}.$$

- Since $p \in [0,1]$ and $x_i \in \{0,1\}$, the second derivative is always negative. The log-likelihood is concave. Therefore, $p = \frac{1}{n}\sum_{i=1}^{n} x_i$ gives us the MLE.
- A twice differentiable function of one variable is concave on an interval if and only if its second derivative is non-positive there!
- Why cannot we have the same closed form solution for logistic regression?

Maximum Likelihood

- Suppose we have samples $x_1, \ldots, x_n$ i.i.d drawn from uniform distribution $\mathcal{U}(a, b)$. Find the maximum likelihood estimator of a and b.

  Solution:

  - The likelihood is:
  $$L(a, b) = \Pi_{i=1}^{n} \left(\frac{1}{b-a} \mathbb{1}_{[a,b]}(x_i) \right)$$
  - Let $x_{(1)}, \ldots, x_{(n)}$ be the order statistics.
  - The likelihood is greater than zero if and only if $a < x_{(1)}$ and $b > x_{(n)}$.
  - When $a < x_{(1)}$ and $b > x_{(n)}$, the likelihood is a monotonically decreasing function of $(b-a)$.
  - And the smallest $(b-a)$ will be attained when $b = x_{(n)}$ and $a = x_{(1)}$.
  - Therefore, $b = x_{(n)}$ and $a = x_{(1)}$ give us the MLE.

Maximum Likelihood

1. We want to fit a regression model where $Y|X = x \sim \mathcal{U}([0, e^{w^T x}])$ for some $w \in \mathbf{R}^d$. Given i.i.d. data points $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbf{R}^d \times \mathbf{R}$, give a convex optimization problem that finds the MLE for w.
2. Suppose we have input-output pairs $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, where $x_i \in \mathbb{R}^p$ and $y_i \in N = \{0, 1, 2, 3, \ldots\}$ for $i = 1, .., n$. Our task is to train a Poisson regression to model the data. Assume the linear coefficients in the model is w.
    1. Suppose a test point x^* is orthogonal to the space generated by the training data. What is the prediction ℓ_2 regularized Poisson GLM make on the test point?
    2. Will the solution of the parameters $\hat{w}$ still be sparse when we use ℓ_1 regularization?

Maximum Likelihood

- We want to fit a regression model where $Y|X = x \sim \mathcal{U}([0, e^{w^T x}])$ for some $w \in \mathbf{R}^d$. Given i.i.d. data points $(X_1, Y_1), \dots, (X_n, Y_n) \in \mathbf{R}^d \times \mathbf{R}$, give a convex optimization problem that finds the MLE for w.

  Solution: The likelihood L is given by

$$L(w; x_1, y_1, \dots, x_n, y_n) = \Pi_{i=1}^n \frac{\mathbb{1}(y_i \leqslant e^{w^T x_i})}{e^{w^T x_i}}.$$

  Taking logs we get

$$-\sum_{i=1}^n w^T x_i = -w^T \left(\sum_{i=1}^n x_i \right)$$

  if $y_i \leqslant \exp(w^T x_i)$ for all i, or $-\infty$ otherwise. Thus we obtain the linear program

$$\begin{array}{ll} \text{minimize} & w^T \left(\sum_{i=1}^n x_i \right) \\ \text{subject to} & \log(y_i) \leqslant w^T x_i \quad \text{for } i = 1, \dots, n. \end{array}$$

Maximum Likelihood

- Suppose we have input-output pairs $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, where $x_i \in \mathbb{R}^p$ and $y_i \in N = \{0, 1, 2, 3, \ldots\}$ for $i = 1, .., n$. Our task is to train a Poisson regression to model the data. Assume the linear coefficients in the model is w.
  - Suppose a test point x^* is orthogonal to the space generated by the training data. What is the prediction ℓ_2 regularized Poisson GLM make on the test point?
    Solution: ℓ_2 penalized Poisson regression objective:

$$\hat{J}(w) = -\sum_{i=1}^{n} \left[y_i w^T x_i - \exp\left(w^T x_i \right) - \log\left(y_i! \right) \right] + \lambda \|w\|_2^2$$

From Representer Theorem, the minimizer $\hat{w} = \sum_{i=1}^{n} \alpha_i x_i$. The prediction is

$$\exp(w^T x^*) = \exp(\sum_{i=1}^{n} \alpha_i x_i^T x^*) = \exp(0) = 1$$

Maximum Likelihood

- Suppose we have input-output pairs $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, where $x_i \in \mathbf{R}R^p$ and $y_i \in N = \{0, 1, 2, 3, \ldots\}$ for $i = 1, .., n$. Our task is to train a Poisson regression to model the data. Assume the linear coefficients in the model is w.
  - Will the solution of the parameters $\hat{w}$ still be sparse when we use ℓ_1 regularization? **Solution:** Negative log-likelihood of Poisson regression is a convex function. The sublevel set is a convex set. The level set is the boundary of the sublevel set. When the level set approaches the diamond (level set of the ℓ_1 norm), it is still likely to hit the corner of the diamond.