泛化理论

第一章 传统统计模型

§1.2.1 广义线性模型

@ 滕佳烨

Recall:

Consistency: when we have infinite training samples ($n \to \infty$), the estimator $\hat{\beta}$ should converge to the true parameter β^*.

- Basic **linear regression**: $y = x^\top \beta^* + \epsilon$
- LSE (least-square estimator): minimize MSE (min square error) $||Y - X\beta||^2$

$$\hat{\beta} = (X^\top X)^{-1} X^\top Y, \qquad \hat{\beta}|X \sim N(\beta^*, \sigma^2 (X^\top X)^{-1})$$

- Variable selection when small signal-to-noise ratio.

Today's topic:

Generalized Linear Models (GLM)

Linear Regression

Consider the linear regimes $y = x^\top \beta^* + \epsilon$, where $\epsilon \sim N(0, \sigma^2)$. Fixed design regime. The estimator can be derived as

$$\hat{\beta} = (X^\top X)^{-1} X^\top Y,$$

where $X \in \mathbb{R}^{n\times d}$ is the design matrix, $Y \in \mathbb{R}^n$ is the response vector (n: training samples, d: dimension).

In other words, we assume that $y|x \sim N(x^\top \beta^*, \sigma^2)$
Strength: simple, easy to estimate and do inference;
Weakness: not flexible.
For example, what if we know $y \in \{0, 1\}$? What if we know $y \geq 0$? Obviously, this violates the assumption $y|x \sim N(x^\top \beta^*, \sigma^2)$.

Can we use some other assumptions on the distribution $y|x$ while preserving the linear property?

Intuition for Generalized Linear Models (GLM)

For example, if we know that $y \in \{0,1\}$ (binary classification), we can assume that $y|x \sim \text{Bernoulli}(p)$, where $p \in [0,1]$.
[That is to say, y $= 1$ with probability p, and $y = 0$ with probability $1 - p$.]

How to set p with linear predictor $x^\top \beta^*$? Since $p \in [0,1]$, we can use $p = \frac{e^{x^\top \beta^*}}{1+e^{x^\top \beta^*}} \in [0,1]$.

Therefore, the total assumption is that:

$$y|x \sim \text{Bernoulli}\left(\frac{e^{x^\top \beta^*}}{1 + e^{x^\top \beta^*}}\right),$$

We can then calculate its MLE (maximum likelihood estimator), for example, in Bernoulli distribution, we maximize $l(\beta)$ to get estimator $\hat{\beta}$:

$$l(\beta) = \sum_i y_i \log(\hat{p}) + (1 - y_i) \log(1 - \hat{p}) = \sum_i y_i \log\left(\frac{e^{x_i^\top \beta}}{1 + e^{x_i^\top \beta}}\right) + (1 - y_i) \log\left(\frac{1}{1 + e^{x_i^\top \beta}}\right).$$

This is so called **Logistic regression**.

But, how to set p?

The Exponential Family

$$p(y;\eta) = b(y)\exp\big(K(\eta)T(y) - a(\eta)\big)$$

- It is a general family, including Gaussian distribution, Bernoulli distribution, and Poisson distribution …

For example, for Bernoulli distribution $p(y) = p^y(1-p)^{1-y}$:

$$p(y;p) = \exp\left(y\log\left(\frac{p}{1-p}\right) + \log(1-p)\right).$$

Therefore, $K(p) = \log\left(\frac{p}{1-p}\right)$ and $T(y) = y$. Setting $K(p) = x^\top\beta^*$ leads to $p = \frac{e^{x^\top\beta^*}}{1+e^{x^\top\beta^*}}$

That is just what we want!

Generalized Linear Models (GLM)

Generally, given a distribution in the exponential family, we can assume that

$$T(y)|x \sim P(\eta) \ \text{ where } \ K(\eta) = x^\top\beta^*.$$

That is how we set the parameter!

Examples in GLM

$$p(y;\eta) = b(\eta)\exp\big(K(\eta)T(y) - a(\eta)\big)$$

We assume $T(y)|x \sim P(\eta)$ where $K(\eta) = x^\top \beta^*$.

Linear regression

- For Gaussian distribution, $K(\eta) = \eta$ and $T(y) = y$.
- Therefore, we assume $y|x \sim N(x^\top \beta^*, \sigma^2)$

Logistic regression

- For Bernoulli distribution, $K(\eta) = \log(\eta/(1-\eta))$ and $T(y) = y$.
- Therefore, we assume $y|x \sim \mathrm{Ber}\left(\frac{e^{x^\top \beta^*}}{1+e^{x^\top \beta^*}}\right)$

Poisson regression

- For Poisson distribution, $K(\eta) = \log \eta$ and $T(y) = y$.
- Therefore, we assume $y|x \sim \mathrm{Poi}(\exp(x^\top \beta^*))$

Softmax regression with multinomial distribution.

…

Take-away messages

(a) Exponential Family: $p(y;\eta) = b(y)\exp\big(K(\eta)T(y) - a(\eta)\big)$.
(b) Generalized linear models (GLM): assume $T(y)|x \sim P(\eta)$ where $K(\eta) = x^\top \beta^*$.
(c) Some examples: linear regression, logistic regression, Poisson regression …

All the slides will be available at www.tengjiaye.com/generalization soon.

@ 滕佳烨

Thanks!