Statistics and Data Science 365 / 565

Data Mining and Machine Learning

February 24

Yale

Outline

- Recap: Some concepts and MLE
- 0/1 Loss rule
- exponential families

Back from break so let's recap some concepts.

Minimize population risk

$$R(f) = \mathbb{E}\ell(f(x), y)$$

Can't do that. Do empirical risk minimization

$$\widehat{R}(f) = \frac{1}{n}\sum_{i=1}^{n} \ell(f(x_i), y_i)$$

Binary classification surrogate losses: Boosting, logistic, hinge

Regression losses: squared, absolute, (there are more)

Choice of losses

$\ell(y', y)$	$\widehat{R}$ name
$(y' - y)^2$	mean-squared error (MSE)
$\lvert y - y' \rvert$	mean absolute deviation (MAD)
$\mathbb{1}(y \neq y')$	Hamming error/0-1 Loss

Training Risk vs. Test Risk

Learning with empirical risk is based on **training risk**: computed on data used in fitting/learning/training/estimating the model.

We are more interested in **test risk** computed on *unseen data*.

Training Risk vs. Test Risk

Learning with empirical risk is based on **training risk**: computed on data used in fitting/learning/training/estimating the model.

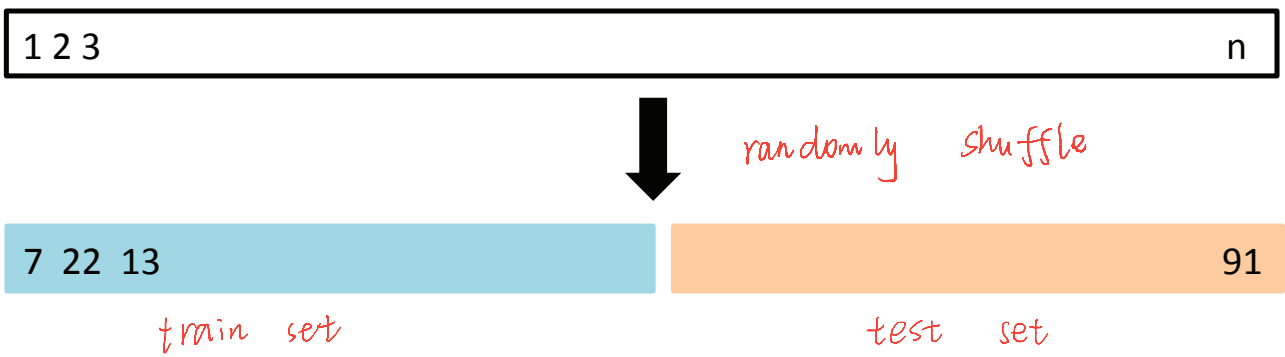
We are more interested in **test risk** computed on *unseen data*.What if we don't have other data?

Training Risk vs. Test Risk

Learning with empirical risk is based on **training risk**: computed on data used in fitting/learning/training/estimating the model.

We are more interested in **test risk** computed on *unseen data*.What if we don't have other data?

We can randomly split our data into a test set and a training set.



be cautious with time series
can't randomly shuffle
eg. train on future data, test on past data

Help avoid over-fitting to training data

data { train set, test set, validation set

Train on training data e.g. find $\widehat{f}$ using ERM

empirical risk minimization

Test on test data e.g. compute MSE on test data using $\widehat{f}$ from training

MLE

Powerful tool to find losses

Parameter space: Θ

True parameter: $\theta^* \in \Theta$

Probability models: $(x, y) \sim \mathbb{P}_{\theta^*}$ assume data (x, y) generate from distribution P_{θ^*}

MLE

Estimation: assume data i.i.d. (don't have to) to build negative log-likelihood (NLL)

negative log likelihood

$$\mathrm{NLL}(\theta) = -\log\left(\prod_{i=1}^{n} \mathbb{P}_\theta(x_i, y_i)\right)$$

$$= -\sum_{i=1}^{n} \log \mathbb{P}_\theta(y_i|x_i) - \sum_{i=1}^{n} \log \mathbb{P}_\theta(x_i)$$

Assume $\mathbb{P}_\theta(x_i)$ does not depend on θ.

$$\widehat{\theta} = \arg\min_\theta \mathrm{NLL}(\theta) = \arg\min_\theta -\sum_{i=1}^{n} \log \mathbb{P}_\theta(y_i|x_i)$$

assume x is fixed, no randomness

negative log-likelihood loss is $-\log \mathbb{P}_\theta(y_i|x_i)$

Logistic Regression Example

$y_i \in \{0, 1\}$

could be $y_i \in \{-1, +1\}$

solution is same

$$\mathbb{P}(y_i|x_i) = \begin{cases} \frac{\exp(x_i^T\beta)}{1+\exp(x_i^T\beta)} & y_i = 1 \\ \frac{1}{1+\exp(x_i^T\beta)} & y_i = 0 \end{cases}$$

concise format

$$\mathbb{P}_\beta(y_i|x_i) = \frac{\exp(y_i x_i^T\beta)}{1+\exp(x_i^T\beta)}$$

drop $P_\beta(x_i)$ part

$$\text{NLL}(\beta) = \sum_{i=1}^{n} \log(1+\exp(x_i^T\beta)) - y_i x_i^T\beta$$

another motivation for logistic loss

More generally $y_i \in \{0, 1\}$

$$\mathbb{P}(y_i|x_i) = p(x_i)^{y_i}(1 - p(x_i))^{1-y_i}$$

an arbitrary function of x_i that estimates probability that label is 1 given x_i

Later $p(x_i)$ will be a neural network. Need to learn $p(x)$

$$\arg\min_{p \in \mathcal{F}} \mathrm{NLL}(p)$$

p in function class

$$\mathrm{NLL}(p) = \sum_{i=1}^{n} -y_i \log(p(x_i)) - (1 - y_i) \log(1 - p(x_i))$$

Linear Regression Example

$x_i^T \beta^*$ can be arbitrary $p(x_i)$

model. $$y_i = x_i^T \beta^* + w_i \quad \text{with } w_i \sim N(0, \sigma^2) \text{ i.i.d.}$$

pdf = $$\mathbb{P}_\beta(y_i | x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i^T \beta - y_i)^2}{2\sigma^2} \right)$$

↑ assume x is fixed

$$\begin{aligned} \text{NLL}(\beta) &= \sum_{i=1}^n \frac{1}{2} \log(2\pi\sigma^2) + \frac{(x_i^T \beta - y_i)^2}{2\sigma^2} \\ &= \sum_{i=1}^n \frac{1}{2} \log(2\pi\sigma^2) + \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i^T \beta - y_i)^2 \end{aligned}$$

Linear Regression Example

$$\mathbb{P}_\beta(y_i|x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i^T\beta - y_i)^2}{2\sigma^2}\right)$$

$$\begin{aligned}\mathrm{NLL}(\beta) &= \sum_{i=1}^{n} \frac{1}{2}\log(2\pi\sigma^2) + \frac{(x_i^T\beta - y_i)^2}{2\sigma^2} \\ &= \sum_{i=1}^{n} \frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i^T\beta - y_i)^2\end{aligned}$$

Verify that

NLL = mean square error (MSE)
= OLS error

$$\widehat{\beta} = \arg\min_\beta \mathrm{NLL}(\beta) = \arg\min_\beta \frac{1}{2n}\sum_{i=1}^{n}(x_i^T\beta - y_i)^2$$

Optimal decision for classification

Suppose we know $\eta(x) = \mathbb{P}(y = 1|x)$

Turns out

$$f^* = \arg\min_{f \in \mathcal{F}} \mathbb{E}\mathbb{1}(f(x) \neq y)$$

satisfies

$$\begin{aligned} f^*(x) &= \mathbb{1}(\eta(x) > 0.5) \\ &= \mathbb{1}\Big[\mathbb{P}(y = 1|x) > \mathbb{P}(y = 0|x)\Big] \end{aligned}$$

Reminder that $\mathbb{E}\mathbb{1}(f(x) \neq y) = \mathbb{P}(f(x) \neq y)$

$$f^* = \arg\min_{f \in \mathcal{F}} \mathbb{E}\mathbb{1}(f(x) \neq y)$$

satisfies

$$\begin{aligned} f^*(x) &= \mathbb{1}(\eta(x) > 0.5) \\ &= \mathbb{1}\Big[\mathbb{P}(y = 1|x) > \mathbb{P}(y = 0|x)\Big] \end{aligned}$$

Why?

$$\mathbb{E}\mathbb{1}(f(x) \neq y) = \mathbb{E}\left[\mathbb{E}[\mathbb{1}(f(x) \neq y)|x]\right]$$

$$= \mathbb{E}\Big[\mathbb{1}(f(x) \neq 1)\mathbb{P}(y = 1|x) + \mathbb{1}(f(x) \neq 0)\mathbb{P}(y = 0|x)\Big]$$

① if $f(x) \neq 0$ and $\neq 1$ have to pay $1 \times P(y=1|x) + 1 \times P(y=0|x) = 1$ penalty $= 1$

② Have to pay $\mathbb{P}(y = 1|x)$ or $\mathbb{P}(y = 0|x)$. Best to pay the smallest.

if $f(x) = 0$ or 1

$$f^*(x) \neq \begin{cases} 1 & \mathbb{P}(y = 1|x) \leq \mathbb{P}(y = 0|x) \\ 0 & \text{otherwise (ow)} \end{cases}$$

Equivalently

$$f^*(x) = \mathbb{1}\Big[\mathbb{P}(y = 1|x) > \mathbb{P}(y = 0|x)\Big]$$

if $=$ penalty $= 0.5$ vs 0.5
don't matter pay which penalty

Recap:

MLE is a powerful tool for building losses

If we know $\mathbb{P}(y = 1|x)$, then we know $f^*(x)$

Estimate from data: for logistic, we have a model

Recap:

Estimate from data: for logistic, we have a model

$$\begin{aligned}\widehat{f}(x) &= \mathbb{1}(\mathbb{P}_{\widehat{\beta}}(y=1|x) > \mathbb{P}_{\widehat{\beta}}(y=0|x)) \\ &= \mathbb{1}\left[\frac{\exp(x^T\widehat{\beta})}{1+\exp(x^T\widehat{\beta})} > \frac{1}{1+\exp(x^T\widehat{\beta})}\right] \\ &= \mathbb{1}[\exp(x^T\widehat{\beta}) > 1] \\ &= \mathbb{1}[x^T\widehat{\beta} > 0]\end{aligned}$$

分母相同

$e^{\cdot}$

Recover linear classifier. (classification function)

Generally easier to just find classifier than to correctly estimate $\mathbb{P}(y=1|x)$ (Called calibration)

Logistic and linear regression are special cases

General framework: generalized linear models

Built from exponential families

Exponential Families

a general way of rewriting some distribution

parameter γ includes χ

- So logistic and linear regression are examples of something bigger

$$p(y;\gamma) = h(y)\ \exp\left[\sum_{i=1}^{s} \gamma_i [T(y)]_i - A(\gamma)\right]$$

probability of y under parameter γ

- γ are the *natural parameters* (canonical params)
- $T(y) \in \mathbb{R}^s$ is the sufficient statistic vector

Exponential Families

$$p(y;\gamma) = h(y)\ \exp\left[\sum_{i=1}^{s} \gamma_i [T(y)]_i - A(\gamma)\right]$$

2 properties (help build function in expenential families)

- Can show $\nabla A(\gamma) = \mathbb{E}T(y)$ (gradient, expectation)
- $H\,A(\gamma) = \mathrm{Cov}(T(y))$ (Hessian is covariance, this means $A(\gamma)$ is convex)

make optimization easy

Exponential Families

$$p(y;\gamma) = h(y)\ \exp\left[\sum_{i=1}^{s} \gamma_i [T(y)]_i - A(\gamma)\right]$$

- Gaussian
- Bernoulli
- Poisson give counts
- Binomial
- Negative binomial give counts

Exponential Families: Gaussian

rewrite Gaussian pdf in this way: one parmeter μ

$$p(y;\gamma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{y^2}{2\sigma^2}\right) \exp\left(y\frac{\mu}{\sigma^2} - \frac{\mu^2}{2\sigma^2}\right)$$

$h(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{y^2}{2\sigma^2}\right)$

- $T(y) = y$
- set $\gamma = \frac{\mu}{\sigma^2}$ unknow is μ, don't care σ
- What is $A(\gamma)$?

Exponential Families: Gaussian

more general : 2 parameters μ, σ^2

$$p(y;\gamma) = \frac{1}{\sqrt{2\pi}} \exp\left(y\frac{\mu}{\sigma^2} - \frac{y^2}{2\sigma^2} - \left(\frac{\mu^2}{2\sigma^2} + \log(\sigma) \right) \right)$$

γ_1 γ_2 $A(\gamma)$

- $T(y) = [y, y^2]^T$
- What are γ and $A(\gamma)$?

$$\Rightarrow \begin{cases} \mu = \frac{\gamma_1 \gamma_2}{2} \\ \sigma^2 = \frac{\gamma_2}{2} \end{cases}$$

Exp fam: Bernoulli

$$p(y;\gamma) = p^y(1-p)^{1-y}$$

$p(y;\gamma) = \mathbb{1}(y \in \{0,1\}) \times \exp[y \log(p) + (1-y)\log(1-p)]$

$= \mathbb{1}(y \in \{0,1\}) \times \exp[y \log \frac{p}{1-p} + \log(1-p)]$

$h(y) = \mathbb{1}(y \in \{0,1\})$

$T(y) = y$, $\boxed{\gamma = \log(\frac{p}{1-p})}$ $A(\gamma) = -\log(1-p)$

log likelihood ratio (log odds)

$\frac{p}{1-p}$: odds

Generalized Linear Models (GLMs)

- Construct GLMs from Exp families

$$p(y;\gamma,x,\beta) = h(y)\ \exp\left[\sum_{i=1}^{s}\gamma_i[T(y)]_i - A(\gamma)\right]$$

The probability of y given γ, x, β

- Simple: case $s = 1$
- $\gamma = x^T\beta$ (modeling choice) now γ depends on x and β
- usually take $T(y) = y$
- $\mathbb{E}[T(y)|x] = A'(x^T\beta)$ 导数
- Can take $s > 1$

Generalized Linear Models

A specific eg

- Construct GLMs from Exp families

$$p(y;\gamma,x,\beta) = h(y)\ \exp\left[\sum_{i=1}^{s}\gamma_i[T(y)]_i - A(\gamma)\right]$$

1. Linear Regression associated with Normal
   - $\mathbb{E}[y|x] = x^T\beta$
2. Logistic with Bernoulli
   - $\mathbb{E}[y|x] = \mathrm{sigm}(x^T\beta)$ no linear function
3. Can do counts (arrivals, phone calls into call center) with Poisson

Linear Regression

eg

$$\mathbb{P}_\beta(y_i|x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i^T\beta - y_i)^2}{2\sigma^2}\right)$$

already seen

$$= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{y_i^2}{2\sigma^2}\right) \exp\left(\frac{1}{\sigma^2} y_i x_i^T \beta - \frac{(x_i^T\beta)^2}{2\sigma^2}\right)$$

$$\begin{cases} \gamma = x^T\beta \\ T(y) = \dfrac{y}{\sigma^2} \\ A(x^T\beta) = (x_i^T\beta)^2/(2\sigma^2) \\ h(y) = \dfrac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\dfrac{y_i^2}{2\sigma^2}\right) \end{cases}$$

Conclusion

MLE is a powerful tool for building losses

Conclusion

MLE is a powerful tool for building losses

General way to build conditional probabilities using exponential families

can then build generalized linear models

do MLE on these models

then do prediction/ classification

Conclusion

Q: Is the purpose of GLM just solve for common properties of multiple kinds of models
OR Do we learn sth new about those models from putting them into this framework
A: both
abstract help remove complexity from specific problems
eg: in optimization, dealing with convex function is easier than directly deal with logistic functions

MLE is a powerful tool for building losses

General way to build conditional probabilities using exponential families

How do we optimize these things? Go to notebook

No closed form solutions except linear regression and weighted least square
if $(X^T X)^{-1}$ could be seen as closed form
↑
have algorithm to solve inverse matrix