Statistical Learning III

15-11-2019

1. Probability Theory Remarks
2. Bayesian Statistics
   a) Intro
   b) Laplace's Rule of Succession
   c) MAP Interpretation of Ridge, Lasso
3. Plug-in Estimators
4. Naive Bayes

3. Probability Theory Remarks

Bayes' rule:

$$Pr(A|B) = \frac{Pr(B|A) \cdot Pr(A)}{Pr(B)}$$

Often used in Bayesian ~~stu~~ and standard frequentist statistics

Densities are slippery:

* Location of maximum depends on choice of parametrization
* Uniform density in one parametrization is not uniform in another parametrization

(see slides)

2. Bayesian Statistics

a] Intro

Probability model: $\mathcal{P} = \{ P_\theta(x, y) \mid \theta \in \Theta \}$ or $\mathcal{P} = \{ P_\theta(y) \mid \theta \in \Theta \}$

Frequentist statistics (standard):
- optimal parameter θ^* is fixed, but unknown
- diversity of methods
- need proofs/experiments to justify methods

Bayesian statistics:
- pretend that true parameter θ^* is a random variable distributed according to prior distribution $\pi(\theta)$ that we know.

$T = Y_1, \ldots, Y_N$ (no features for simplicity)

$$\Pr(T, \theta) = P_\theta(T) \cdot \pi(\theta)$$ is joint distribution of data and parameters

Since we know the full joint distribution, can simply use probability theory to compute any probability we are interested in. ← single method

↳ if we estimate different probabilities this way, they are beautifully consistent.

Bayesian premise is too strong:
- prior π is chosen for computational or information theoretic properties in practice, so cannot blindly assume θ^* is random sample from π.
- need proofs/experiments to justify Bayesian methods

Modern frequentist motivation:

- If we choose π right, then often works really well (both in theory and in practice).
- Learn faster if true θ^* has high prior probability, slower if true θ^* has small prior probability, so can use π to express prior knowledge about our data.

Posterior Distribution:

$$\pi(\theta|T) := Pr(\theta|T) = \frac{Pr(\theta,T)}{Pr(T)} = \frac{P_\theta(T)\cdot\pi(\theta)}{Pr(T)}$$

- Often puts its probability mass closer and closer to θ^* as $N \to \infty$. (vd Vaart et al.)
- Expresses uncertainty about θ

Predictive Distribution:

$$Pr(Y|T) = \int_\Theta P_\theta(Y)\cdot\pi(\theta|T)\,d\theta = \frac{Pr(Y,T)}{Pr(T)}$$

new sample outside of training set

often better predictions than frequentist $P_{\hat\theta}(Y)$ because $\pi(\theta|T)$ keeps track of uncertainty better than fixed single choice $\hat\theta$.

b] Example: Laplace Rule of Succession

Bernoulli model: $P_\theta(Y) = \begin{cases}\theta & \text{for } Y=1\\ 1-\theta & \text{for } Y=0\end{cases}$ $\quad \theta\in[0,1]$

Maximum likelihood: $\hat\theta = \frac{n_1}{N} \to P_{\hat\theta}(Y=1) = \hat\theta = \frac{n_1}{N}$ ← dangerous for prediction if $n_1=0$

Suppose $\pi(\theta)=1$ is uniform prior density, ⇐ not the same as "no prior knowledge" because depends on parametrisation

Then $Pr(Y=1|T) = \frac{n_1+1}{N+2}$

(Laplace, 1814)

$Pr(Y=0|T) = \frac{n_0+1}{N+2}$

beta function

Proof: $$\frac{Pr(Y,T)}{Pr(T)} = \frac{\int P_\theta(Y,T)\cdot\pi(\theta)\,d\theta}{\int P_\theta(T)\cdot\pi(\theta)\,d\theta} = \frac{\int \theta^{n_1+Y}(1-\theta)^{n_0+1-Y}\,d\theta}{\int \theta^{n_1}(1-\theta)^{n_0}\,d\theta} = \frac{n_Y+1}{N+2} \quad \square$$

c] MAP interpretation of Ridge and Lasso

Maximum a Posteriori (MAP) parameters:

$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}}\ \pi(\theta|T) = \underset{\theta}{\operatorname{argmax}}\ \frac{P_\theta(T)\cdot\pi(\theta)}{Pr(T)}$$

$$= \underset{\theta}{\operatorname{argmax}}\ P_\theta(T)\cdot\pi(\theta)$$

- maximizes posterior <u>density</u>, so for continuous parameters depends on parametrisation!
- "real" Bayesians prefer prediction with predictive distribution.

Ridge / Lasso:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}\ RSS(\beta) + \lambda\, pen(\beta)$$

ridge: $pen(\beta) = \sum_{j=2}^{p} \beta_j^2$ lasso: $pen(\beta) = \sum_{j=2}^{p} |\beta_j|$

Suppose Gaussian noise:

$$Y = x^T\beta + \varepsilon \qquad \varepsilon \sim \mathcal{N}(0,\sigma^2)$$

$$P_\beta(y_1,\dots,y_N|x_1,\dots,x_N) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(y_i - x_i^T\beta)^2}{2\sigma^2}}$$

$$\hat{\beta}_{MAP} = \underset{\beta}{\operatorname{argmax}}\ P_\beta(y_1,\dots,y_N|x_1,\dots,x_N)\cdot\pi(\beta)$$

$$= \underset{\beta}{\operatorname{argmin}}\ -\log P_\beta(y_1,\dots,y_N|x_1,\dots,x_N) - \log\pi(\beta)$$

$$= \underset{\beta}{\operatorname{argmin}}\ N\cdot\left(-\log\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \frac{1}{2\sigma^2}RSS(\beta) - \log\pi(\beta)$$

$\log(x^a) = a\log(x)$
$\log(ab) = \log(a) + \log(b)$

$$= \underset{\beta}{\operatorname{argmin}}\ RSS(\beta) - 2\sigma^2\log\pi(\beta)$$

Suppose data have been pre-processed such that we can assume that the intercept $\hat{\beta}_1 = 0$.

Ridge: Choose π s.t.

$$\beta_1 = 0 \quad \text{with prob. } 1$$
$$(\beta_2, \dots, \beta_p) \sim \mathcal{N}(0, \tau^2 I)$$

$$-\log \pi(\beta) = \sum_{j=2}^{p} -\log\left(\frac{1}{\sqrt{2\pi\tau^2}} \cdot e^{-\frac{(\beta_j - 0)^2}{2\tau^2}}\right)$$

$$= (p-1)\left(-\log \frac{1}{\sqrt{2\pi\tau^2}}\right) + \sum_{j=2}^{p} \frac{\beta_j^2}{2\tau^2}$$

$$\hat{\beta}_{MAP} = \underset{\beta}{\operatorname{argmin}}\ RSS(\beta) + \frac{\sigma^2}{\tau^2} \sum_{j=2}^{p} \beta_j^2$$

is ridge with $\lambda = \frac{\sigma^2}{\tau^2}$

Is also posterior mean $\mathbb{E}_{\pi(\beta|T)}[\beta]$

Lasso: Choose π s.t.

$$\beta_1 = 0 \quad \text{with prob. } 1.$$
$$(\beta_2, \dots, \beta_p) \sim \prod_{j=2}^{p} \frac{1}{2\tau} e^{-\frac{|\beta_j|}{\tau}}$$

$$-\log \pi(\beta) = \sum_{j=2}^{p} -\log\left(\frac{1}{2\tau} \cdot e^{-\frac{|\beta_j|}{\tau}}\right)$$

$$= (p-1)\left(-\log \frac{1}{2\tau}\right) + \sum_{j=2}^{p} \frac{|\beta_j|}{\tau}$$

$$\hat{\beta}_{MAP} = \underset{\beta}{\operatorname{argmin}}\ RSS(\beta) + \frac{2\sigma^2}{\tau} \sum_{j=2}^{p} |\beta_j|$$

is Lasso with $\lambda = \frac{2\sigma^2}{\tau}$.

Remarks:

- "real" Bayesian prefers predicting with predictive distribution
- MAP + CV to determine λ is very unBayesian.

1. Plug-in Estimators

Bayes' rule gives: $p^*(y=k|x) = \dfrac{p^*(x|y=k)\,p^*(y=k)}{p^*(x)}$

Notation:

$f_k(x) := p^*(x|y=k)$

$\pi_k := p^*(y=k)$

Plug-in estimators:

- Estimate $p^*(y=k|x)$ by $\hat{p}(y=k|x)$ and predict with $\underset{k}{\operatorname{argmax}}\ \hat{p}(y=k|x)$

- By Bayes' rule, it is suffient to estimate f_k and π_k for all classes k:

$$\hat{f}(x) = \underset{k}{\operatorname{argmax}}\ \frac{\hat{f}_k(x)\,\hat{\pi}_k}{p^*(x)} = \underset{k}{\operatorname{argmax}}\ \hat{f}_k(x)\cdot\hat{\pi}_k$$

N.B. Although we have used Bayes' rule, there is nothing inherently Bayesian about this approach. (Can estimate $\hat{f}_k$ by Bayesian or frequentist methods.)

3. Naive Bayes

Model: features are <u>independent</u> given their class k

Remark: This implies that $\Sigma_k = \begin{pmatrix} \sigma^2_{k,1} & & 0 \\ & \ddots & \\ 0 & & \sigma^2_{k,p} \end{pmatrix}$

is a diagonal matrix with p parameters, because (by assumption) all covariances are 0.

For each class k and feature vector x_i:

$$f_k(x_i) = \prod_{j=1}^{p} f_{kj}(x_{ij})$$

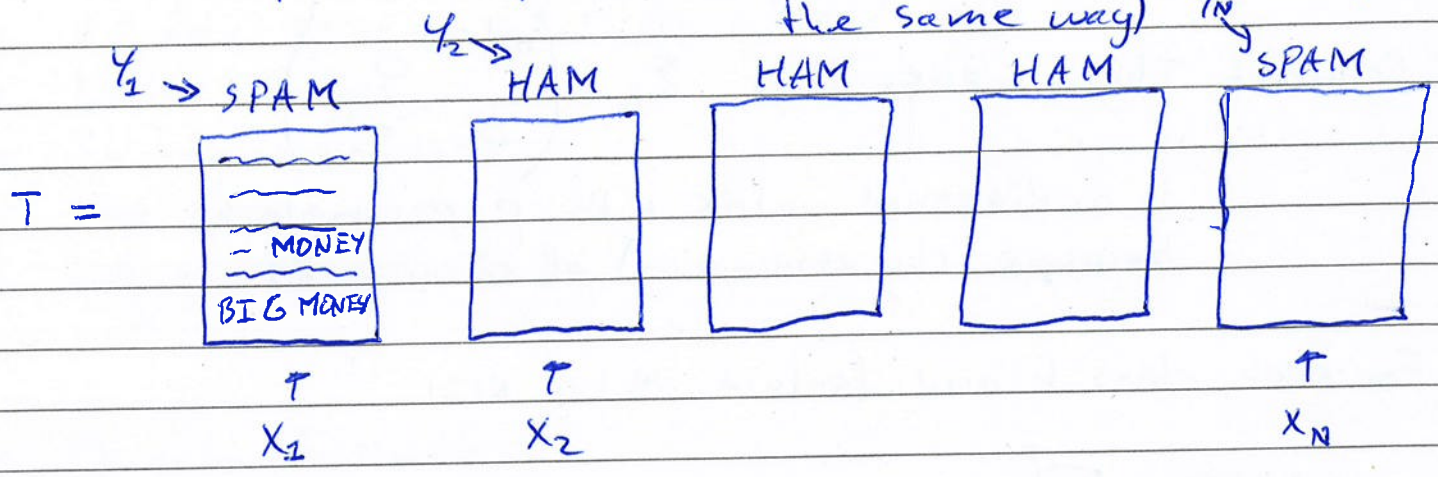
← independent features; very useful simplification if p is very large, because then we have no hope of learning the true distribution exactly anyway.

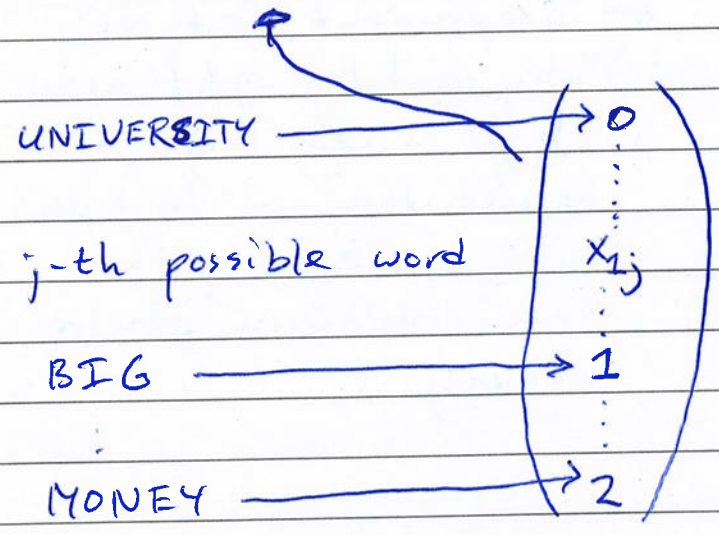
<u>Continuous features: $x_{ij} \in \mathbb{R}$</u>

For each feature j and every class k:
estimate f_{kj} by fitting a univariate Gaussian (similar to LDA).

Discrete features:

Consider spam classification. (other text classification goes the same way)





θ_{k1} $\quad$ $\theta_{k,j} \geq 0$

$\vdots$ $\quad$ $\sum_{j=1}^{p} \theta_{kj} = 1$

$\theta_{k,BIG}$

$\theta_{k,p}$

all the words in my dictionary of possible words

(1) Forget position of each word in text, and represent each e-mail by counts of how many times words occur.

(2) Use separate multinomial model per class k:

$$f_k(x_i) = \prod_{j=1}^{p} \theta_{kj}^{n_{ij}}$$

, where n_{ij} is the number of times the j-th word occurs in the i-th e-mail.

$$\pi_k = \frac{\#\text{e-mails in class } k}{N}$$

$$\hat{\theta}_{k,MONEY} = \frac{\#\text{MONEY in e-mails from class } k + \text{small num}}{\#\text{words in all e-mails from class } k + p \cdot \text{small num}}$$

Without adding the "small numbers": if a spam e-mail contains one word that we have never seen in a spam e-mail before, then $f_{SPAM}(x_i) = 0$, and the e-mail will be classified as HAM. So a single word can throw off classification.

Too simple/naive?

- Only need $\hat{P}(y=k_1|x) > \hat{P}(y=k_2|x)$ whenever $p^*(y=k_1|x) > p^*(y=k_2|x)$
- Possible even if $\hat{P}$ is poor estimate of p^*!