Latent Dirichlet kernel & Bayesian kernels

Daichi Mochihashi

daichi.mochihashi@atr.jp

ATR SLT, Department 2 (SLR)

Über SVM 2004

Aug 3, 2004, NAIST

Introduction

- 確率分布の間のカーネル (↔ データ点間のカーネル)
  $K(p(\theta|\mathbf{x}), p(\theta|\mathbf{y})) \leftrightarrow K(x, y)$
- 適切な prior を設定することで，事前知識を自然に取りこめる
- “Latent Dirichlet kernel” for documents
  - 最尤推定の罠を回避することができる
  - 文書の「長さ」を (正規化することなく) 自然にカーネルに反映できる.

Kernel for probability distributions

- Probability Product Kernel (PPK):

$$K_\beta(p, p') = \int p(x)^\beta p'(x)^\beta dx \tag{1}$$

  - パラメータ: β (意味は後で説明)
  - Positive semidefinite.
- $\beta = 1$: Expected Likelihood Kernel

$$K(p, p') = \int p(x)p'(x)dx \tag{2}$$

$$= \langle p(x) \rangle_{p'(x)} = \langle p'(x) \rangle_{p(x)} \tag{3}$$

  - 互いに見た，相手の確率分布の期待値.

Bhattacharyya kernel

- $\beta = 1/2$: Bhattacharyya kernel (Bhattacharyya 1943, Kondor & Jebara 2003)

$$K(p, p') = \int \sqrt{p(x)}\sqrt{p'(x)}dx \tag{4}$$

  - $K(p, p') \equiv 1$.
- Hellinger 距離: bound of KL divergence

$$H(p, p') = \frac{1}{2}\int \left(\sqrt{p(x)} - \sqrt{p'(x)}\right)^2 dx \tag{5}$$

- $H(p, p') = \sqrt{2 - 2K(p, p')}$.

Example: Gaussian PPK

- D 次元の 2 つの Gaussian の PPK:

$$K(N(\mu,\Sigma),N(\mu',\Sigma')) = (2\pi)^{\frac{1-2\beta}{2}D}|\Sigma_0|^{\frac{1}{2}}\left(|\Sigma||\Sigma'|\right)^{-\frac{\beta}{2}} \times \exp\left(-\frac{\beta}{2}(\mu^T\Sigma^{-1}\mu - \mu'^T\Sigma'^{-1}\mu') + \frac{1}{2}\mu_0^T\Sigma_0\mu_0\right) \tag{6}$$

- 特に，共分散行列が diagonal: $\sigma^2 I$ であれば,

$$\begin{aligned} K(N(\mu,\sigma^2 I),N(\mu',\sigma^2 I)) &= (4\pi\sigma^2)^{-\frac{D}{2}}\exp\left(-|\mu-\mu'|^2/4\sigma^2\right) \\ &= (4\pi\sigma^2)^{-\frac{D}{2}}\exp\left(-|\bar{x}-\bar{x'}|^2/4\sigma^2\right) \\ &\quad (\mu=\bar{x},\mu'=\bar{x'}) \end{aligned} \tag{7}$$

$\therefore$ RBF kernel が導かれる.

Example 2 : Multinomial PPK

- 多項分布 $\hat{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$,
  $\hat{\theta}' = (\theta'_1, \theta'_2, \ldots, \theta'_K)$ の PPK は，X 語の 2 つの文書に対して,

$$K(p(x|\hat{\theta}), p(x|\hat{\theta}')) = \left(\sum_{i=1}^{K} (\theta_i \theta'_i)^{\frac{1}{2}} \right)^X . \tag{8}$$

  - = homogeneous polynomial kernel, degree $X/2$.
  - 文書の長さが同じ X 語でないと比べられない
    (x で積分 (和) がとれないため.)

EF as a Gibbs distribution

- Exponential Family

$$p(x|\theta) \propto \exp(\theta \cdot \phi(x) + a(x)) \quad (\theta：自然パラメータ) \tag{9}$$

$$\propto \frac{1}{Z_\theta} \exp(H(x,\theta)). \qquad (H(x,\theta) = \theta \cdot \phi(x) + a(x))$$

$$\therefore\ p(x|\theta)^\beta \propto \exp(\beta(\theta \cdot \phi(x) + a(x))) \tag{10}$$

$$= \exp(\beta H(x,\theta)). \quad (\beta = \frac{1}{T}) \tag{11}$$

- ゆえに，PPK(β)

$$K_\beta(p,p') = \int \{p(x|\theta)^\beta\}\{p'(x|\theta)^\beta\}dx \tag{12}$$

は，温度 $T = 1/\beta$ のもとでの Expected likelihood kernel? (quasi Heat Kernel.)

PLSI for two documents

- 文書 $d = \mathbf{w}$ があったとき，トピック $t = t_1 \dots t_K$ に対して，

$$\begin{cases} p(t|w,d) \propto p(w|t)p(t|d) \\ p(w|t) \quad \propto \sum_d n(w,d)p(t|w,d) \\ p(t|d) \quad \propto \sum_w n(w,d)p(t|w,d) \end{cases} \tag{13}$$

  以下，$\boldsymbol{\lambda} = \{p(t|d)\}$ とおく．

- 文書 d, d' の affinity = トピック混合比 $\boldsymbol{\lambda}, \boldsymbol{\lambda}'$ の近さ
  - KL, Jensen-Shannon
  - Information diffusion kernel
- 問題: $\boldsymbol{\lambda}$ は点推定
  - オーバーフィット (ex. とても短い文書)
  - 長い文書も短い文書も一点 $\boldsymbol{\lambda}$ で代表?

    ↓

    $p(\boldsymbol{\lambda}|d), p(\boldsymbol{\lambda}|d')$ の比較．(Latent Dirichlet kernel)

Latent Dirichlet kernel (1)

- $p(\boldsymbol{\lambda}|d) = \mathrm{Dir}(\boldsymbol{\lambda}|\boldsymbol{\alpha})$ (Dirichlet 分布)

$$\boldsymbol{\alpha}^* = \underset{\boldsymbol{\alpha}}{\mathrm{argmax}} \int p(\mathbf{w}|\boldsymbol{\lambda}) p(\boldsymbol{\lambda}|\boldsymbol{\alpha}) d\boldsymbol{\lambda} \tag{14}$$

$$= \underset{\boldsymbol{\alpha}}{\mathrm{argmax}} \int \prod_{i=1}^{|d|} \sum_{k} p(w_i|z_k) \lambda_k p(\boldsymbol{\lambda}|\boldsymbol{\alpha}) d\boldsymbol{\lambda}. \tag{15}$$

- この $\boldsymbol{\alpha}$ は単純な EM アルゴリズムでは求まらない.
(隠れ変数が階層化されているため)

Latent Dirichlet kernel (2)

- Variational Bayes EM algorithm (Attias 1999)

$$\begin{cases} q(Z) \propto \exp\langle \log p(Y, Z|\theta) \rangle_{q(\theta)} & \textbf{(VB-Estep)} \\ q(\theta) \propto \exp\langle \log p(Y, Z|\theta) \rangle_{q(Z)} \cdot p(\theta) & \textbf{(VB-Mstep)} \end{cases} \tag{16}$$

$$\Longleftrightarrow \begin{cases} q(z_i^t = 1|d) \propto p(w_i|t) \exp(\Psi(\alpha_0 + n_t)) \\ q(\boldsymbol{\lambda}|d) \propto \prod_{t=1}^{K} \lambda_t^{\alpha_0 + n_t - 1} \quad (\Leftrightarrow \alpha = \alpha_0 + n_t) \end{cases} \tag{17}$$

$$\text{where } n_t = \sum_{i=1}^{|d|} q(z_i^t = 1|d) \tag{18}$$

- (17) 式を iterate することで，文書 d に対するハイパーパラメータ $\boldsymbol{\alpha}$ が得られる．

Latent Dirichlet kernel (3)

- 文書 d, d' について，それを生成したトピックの Dirichlet 分布のハイパーパラメータ $\boldsymbol{\alpha}, \boldsymbol{\beta}$ が得られたので，Bhattacharyya kernel の場合，

$$K(d, d') = \int_{\triangle} \sqrt{p(\boldsymbol{\lambda}|\alpha)} \sqrt{p(\boldsymbol{\lambda}|\beta)} d\boldsymbol{\lambda} \tag{19}$$

$$= \int_{\triangle} \left[\frac{\Gamma(\alpha_0)}{\prod_i \Gamma(\alpha_i)} \right]^{1/2} \prod_i \lambda_i^{\alpha_i/2 - 1/2} \cdot \left[\frac{\Gamma(\beta_0)}{\prod_i \Gamma(\beta_i)} \right]^{1/2} \prod_i \lambda_i^{\beta_i/2 - 1/2} d\boldsymbol{\lambda} \tag{20}$$

$$= \left[\frac{\Gamma(\alpha_0)\Gamma(\beta_0)}{\prod_i \Gamma(\alpha_i)\Gamma(\beta_i)} \right]^{1/2} \cdot \frac{\prod_i \Gamma(\frac{1}{2}\{\alpha_i + \beta_i\})}{\Gamma(\frac{1}{2}\sum_i \{\alpha_i + \beta_i\})}. \tag{21}$$

Latent Dirichlet kernel (4)

- $\beta = 1/2$ (Bhattacharyya) でなくともよい ($\beta = 1$, あるいは最適な「温度」は実験的に決まる)
- 文書の「長さ」を自然にカーネルとして取りこむことが可能
  - 二つの Dirichlet 分布の重なり
  - 短い文書はなだらかな推定，長い文書は鋭いピーク (のこともある)

Is LDA perfect?

- No.
- (事前の)LDA によって決まった，topic simplex の内部しか表現できない．(山本 2003/SLP48)
  - 極端な偏りのある文書 (simplex から外れる)
- このことの原因: $p(w|z_t)$ が点推定であること.
  - $p(w|z_t)$ を Dirichlet 分布とする
  - Dirichlet mixture of Dirichlet mixture?

Kernel machine ≠ SVM

- Herbrich & Graepel (1999): “Bayesian Learning in Reproducing Kernel Hilbert Spaces”
  - 最初から二値分類 (SVM) を前提にしている
    ⇓
    萎え萎え．orz
- NLP でのカーネルは，出口がみんな SVM
- 分類問題以外での NLP でのカーネルの使い道は?
  - 詳しい方教えて下さい.
- Bayes Point Machine についてはもっと勉強します‥.

おまけ: SVMsequel

- `http://www.isi.edu/~hdaume/SVMsequel/`
- 常識??
- Linear, Polynomial, RBF, ..., などの普通のカーネルの他に,
  - Tree kernel
  - String kernel (DP ($O(nm)$)/Suffix Tree ($O(n+m)$))
  - Information diffusion kernel
  - Gram 行列直接指定

  などが実装されている.
- カーネルを自分で追加可能
- Ocaml で書かれている (バイナリあり)
  - Ocaml コンパイラが必要な人は, `cl:~daiti-m/arch/linux/bin/` にパスを通して下さい.

SVMsequel サンプル

dot.ml より:

```
let denseKernel k x y =
let xdoty = dotValue x y in
  match k with
      LinearKernel -> xdoty
    | PolynomialKernel (a,b,d) -> (a +. b *. xdoty) ** d
    | RbfKernel g -> exp (g *. (2. *. xdoty -. dotValue x x
    | SigKernel (a,b) -> tanh (a +. b *. xdoty)
    | IdKernel b ->
        let v = acos xdoty in
          (4.*.3.14159265*.b) ** (-0.5 *. (float_of_int l -.
          exp (0. -. v *. v /. b)
```

- 注: let は関数定義 (と変数束縛の両方).