Learning with probabilities as inputs, using kernels

Arthur Gretton

Gatsby Computational Neuroscience Unit

NIPS workshop on Probabilistic Numerics, 2015

Motivating example: Expectation Propagation

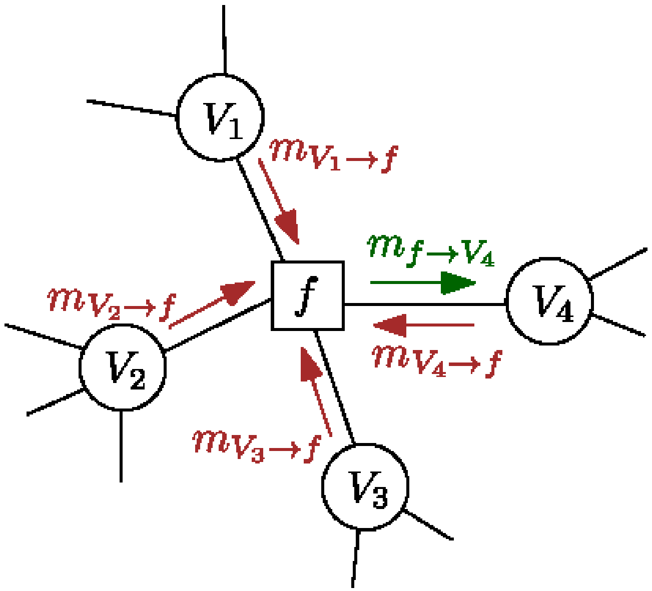
set of c variables connected to f

projected message

$$m_{f\to V_i}(v_i) = \frac{\text{proj}\left[\int d\mathcal{V}\setminus\{v_i\}\, f(\mathcal{V}) \prod_{j=1}^{c} m_{V_j\to f}(v_j)\right]}{m_{V_i\to f}(v_i)} := \frac{q_{f\to V_i}(v_i)}{m_{V_i\to f}(v_i)}$$

proj$[r_{f\to V_i}]$:= arg min$_{q\in\text{ExpFam}}$ KL $[r_{f\to V_i} \,\|\, q]$
(projection onto exponential family)

incoming message from V_j



Motivating example: Expectation Propagation

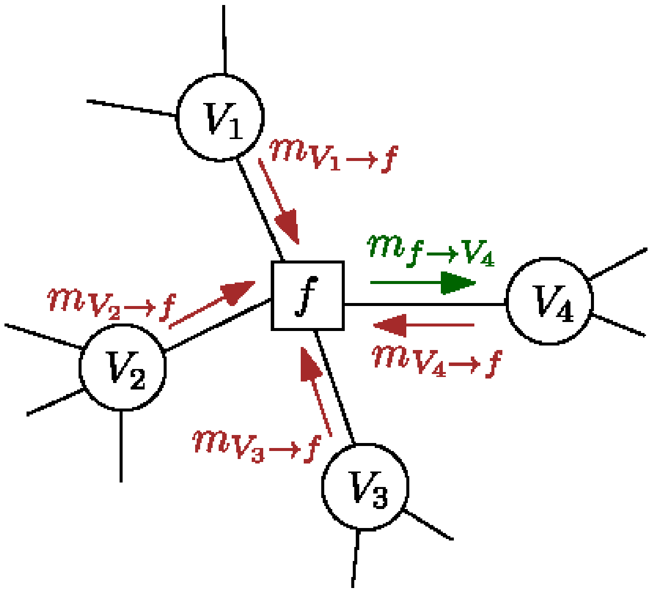
set of c variables connected to f

projected message

$$m_{f\to V_i}(v_i) = \frac{\mathrm{proj}\left[\int d\mathcal{V}\backslash\{v_i\}\, f(\mathcal{V}) \prod_{j=1}^{c} m_{V_j\to f}(v_j)\right]}{m_{V_i\to f}(v_i)} := \frac{q_{f\to V_i}(v_i)}{m_{V_i\to f}(v_i)}$$

$\mathrm{proj}[r_{f\to V_i}] := \arg\min_{q\in \mathrm{ExpFam}} \mathrm{KL}\left[r_{f\to V_i} \,\|\, q\right]$
(projection onto exponential family)

incoming message from V_j



- Expensive integral (besides special cases).
- **Goal:** Learn an *uncertainty aware* message operator (regression function)

$$\left[m_{V_j\to f}\right]_{j=1}^{c} \mapsto q_{f\to V_i}.$$

- **Challenges:** dealing with huge sample size, knowing when to consult expensive oracle.

Overview

- Introduction to reproducing kernel Hilbert spaces
  - Kernels and feature spaces
  - Mapping probabilities to feature space
- Learning with distribution-valued inputs
  - Learning rates achievable when samples from disributions available [AISTATS15, JMLR in revision]
  - Approximate, uncertainty-aware regression with application to EP [UAI15]
  - Learning to predict direction of causality [Lopez-Paz et al., 2015]
- Learning with distribution-valued outputs (not this talk)

Kernels: similarity between features

- We have two objects x and x' from a set $\mathcal{X}$ (documents, images, ...).
  How similar are they?

Kernels: similarity between features

- We have two objects x and x' from a set $\mathcal{X}$ (documents, images, ...). How similar are they?
- Define features of objects:
  - $\varphi_x \in \mathcal{F}$ are features of x,
  - $\varphi_{x'} \in \mathcal{F}$ are features of x'
- A kernel is the dot product between these features:

$$k(x, x') := \langle \varphi_x, \varphi_{x'} \rangle_{\mathcal{F}} = \sum_{j \in J} \varphi_x^{(j)} \varphi_{x'}^{(j)}$$

- A function in the RKHS $\mathcal{F}$ is a linear combination of features,

$$f(x) = \langle f, \varphi_x \rangle_{\mathcal{F}} = \sum_{j \in J} f_j \varphi_x^{(j)} \qquad f \in \ell_2(J)$$

Infinite dimensional feature space

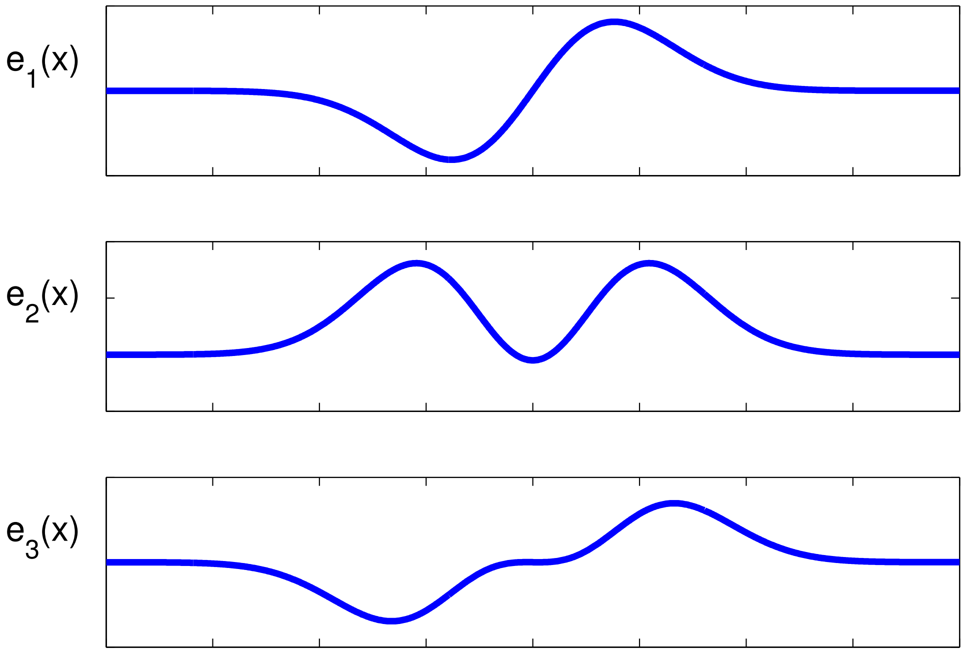
Squared exponential kernel: $k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right)$

Infinite dimensional feature space

Squared exponential kernel: $k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right)$

$$\begin{aligned} \lambda_j &\propto b^j \qquad b < 1 \\ e_j(x) &\propto \exp(-(c-a)x^2)H_j(x\sqrt{2c}), \end{aligned}$$

a, b, c are functions of σ, and H_j is jth order Hermite polynomial.

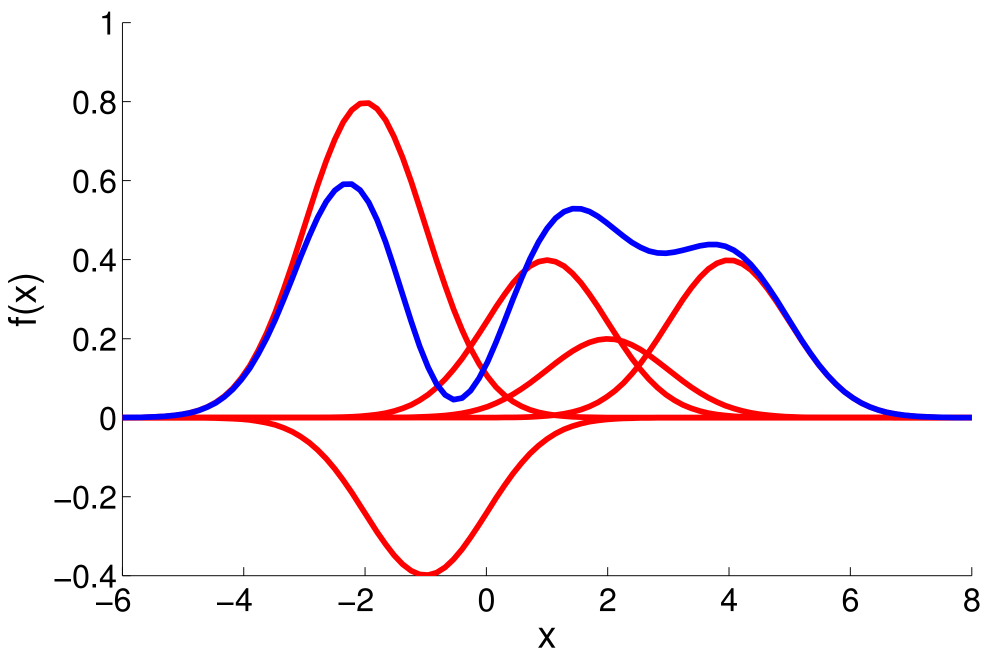


$$\begin{aligned} &k(x, x') \\ &= \sum_{j=1}^{\infty} \lambda_j e_j(x) e_j(x') \\ &= \sum_{j=1}^{\infty} \left(\sqrt{\lambda_j} e_j(x)\right)\left(\sqrt{\lambda_j} e_j(x')\right) \\ &= \sum_{j=1}^{\infty} \varphi_x^{(j)} \varphi_{x'}^{(j)} \end{aligned}$$

The kernel trick

Example RKHS function, squared exponential kernel:

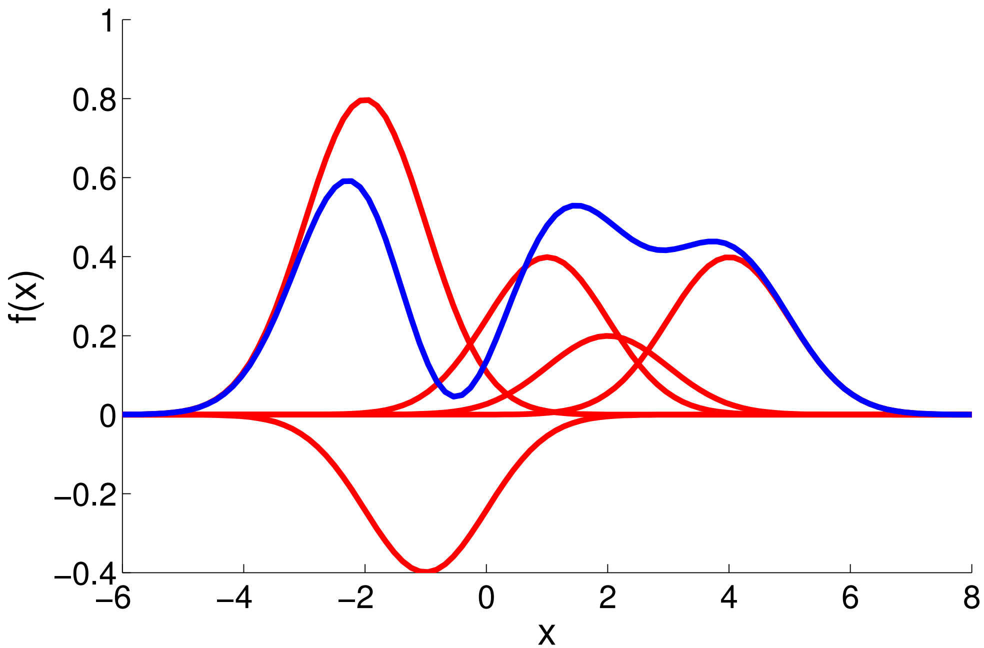
$$f(x) := \sum_{j=1}^{\infty} f_j \varphi_x^{(j)}$$



The kernel trick

Example RKHS function, squared exponential kernel:

$$f(x) := \sum_{i=1}^{m} \alpha_i k(x_i, x)$$

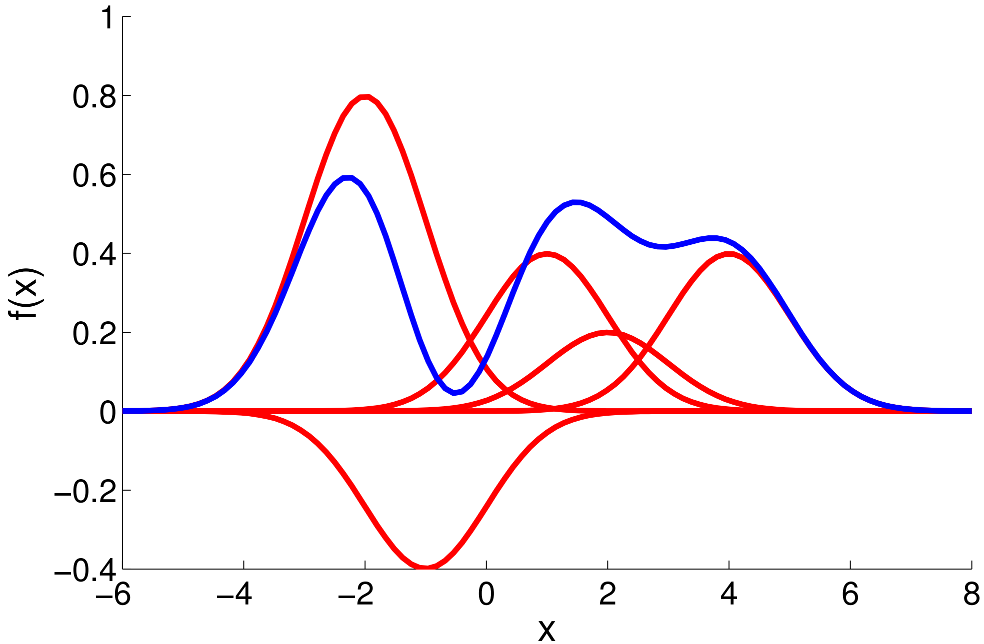


The kernel trick

Example RKHS function, squared exponential kernel:

$$f(x) := \sum_{i=1}^{m} \alpha_i k(x_i, x) = \sum_{i=1}^{m} \alpha_i \left[\sum_{j=1}^{\infty} \varphi_{x_i}^{(j)} \varphi_x^{(j)} \right] = \sum_{j=1}^{\infty} f_j \varphi_x^{(j)}$$

where $f_j = \sum_{i=1}^{m} \alpha_i \varphi_{x_i}^{(j)}$



Probabilities in feature space: the mean trick

The kernel trick

- Given $x \in \mathcal{X}$ for some set $\mathcal{X}$, define feature map $\varphi_x \in \mathcal{F}$,

$$\varphi_x = \left[\ldots \varphi_x^{(j)} \ldots \right] \in \ell_2$$

- For positive definite $k(x, x')$,

$$k(x, x') = \langle \varphi_x, \varphi_{x'} \rangle_{\mathcal{F}}$$

- Function in the RKHS: $\forall f \in \mathcal{F}$,

$$f(x) = \langle f, \varphi_x \rangle_{\mathcal{F}}$$

Probabilities in feature space: the mean trick

The kernel trick

- Given $x \in \mathcal{X}$ for some set $\mathcal{X}$, define feature map $\varphi_x \in \mathcal{F}$,

$$\varphi_x = \left[\ldots \varphi_x^{(j)} \ldots\right] \in \ell_2$$

- For positive definite $k(x, x')$,

$$k(x, x') = \langle \varphi_x, \varphi_{x'} \rangle_{\mathcal{F}}$$

- Function in the RKHS: $\forall f \in \mathcal{F}$,

$$f(x) = \langle f, \varphi_x \rangle_{\mathcal{F}}$$

The mean trick

- Given $\mathbf{P}$ a Borel probability measure on $\mathcal{X}$, define mean embedding $\mu_{\mathbf{P}} \in \mathcal{F}$

$$\mu_{\mathbf{P}} = \left[\ldots \mathbf{E}_{\mathbf{P}}\left[\varphi_X^{(j)}\right] \ldots\right] \in \ell_2(J)$$

- For positive definite $k(x, x')$,

$$\mathbf{E}_{\mathbf{P},\mathbf{Q}} k(X, Y) = \langle \mu_{\mathbf{P}}, \mu_{\mathbf{Q}} \rangle_{\mathcal{F}}$$

for $X \sim \mathbf{P}$ and $Y \sim \mathbf{Q}$.

Need to ensure Bochner integrability of φ_x for $x \sim \mathbf{P}$

- $\mathbf{E}_{\mathbf{P}}(f(X)) =: \langle \mu_{\mathbf{P}}, f \rangle_{\mathcal{F}}$

Kernels on distributions in supervised learning

- Kernels have been very widely used in supervised learning
  - Support vector classification/regression, kernel ridge regression ...

Kernels on distributions in supervised learning

- Kernels have been very widely used in supervised learning
- Simple kernel on distributions (population counterpart of set kernel)

  [Haussler, 1999, Gärtner et al., 2002]

$$K(\mathbf{P}, \mathbf{Q}) = \langle \mu_{\mathbf{P}}, \mu_{\mathbf{Q}} \rangle_{\mathcal{F}}$$

- Squared distance between distribution embeddings (MMD)

$$\mathrm{MMD}^2(\mu_{\mathbf{P}}, \mu_{\mathbf{Q}}) := \|\mu_{\mathbf{P}} - \mu_{\mathbf{Q}}\|^2_{\mathcal{F}} = \mathbf{E}_P k(\mathsf{x}, \mathsf{x}') + \mathbf{E}_Q k(\mathsf{y}, \mathsf{y}') - 2\mathbf{E}_{\mathbf{P},\mathbf{Q}} k(\mathsf{x}, \mathsf{y})$$

Kernels on distributions in supervised learning

- Kernels have been very widely used in supervised learning
- Simple kernel on distributions (population counterpart of set kernel)

  [Haussler, 1999, Gärtner et al., 2002]

$$K(\mathbf{P}, \mathbf{Q}) = \langle \mu_{\mathbf{P}}, \mu_{\mathbf{Q}} \rangle_{\mathcal{F}}$$

- Can define kernels on mean embedding features [Christmann, Steinwart NIPS10],[AISTATS15]

K_G	K_e	K_C	K_t	$\ldots$
$e^{-\frac{\left\Vert \mu_{\mathbf{P}} - \mu_{\mathbf{Q}} \right\Vert_{\mathcal{F}}^2}{2\theta^2}}$	$e^{-\frac{\left\Vert \mu_{\mathbf{P}} - \mu_{\mathbf{Q}} \right\Vert_{\mathcal{F}}}{2\theta^2}}$	$\left(1 + \left\Vert \mu_{\mathbf{P}} - \mu_{\mathbf{Q}} \right\Vert_{\mathcal{F}}^2 / \theta^2\right)^{-1}$	$\left(1 + \left\Vert \mu_{\mathbf{P}} - \mu_{\mathbf{Q}} \right\Vert_{\mathcal{F}}^{\theta}\right)^{-1}, \ \theta \leq 2$	$\ldots$

$$\left\Vert \mu_{\mathbf{P}} - \mu_{\mathbf{Q}} \right\Vert_{\mathcal{F}}^2 = \mathbf{E}_P k(\mathsf{x}, \mathsf{x}') + \mathbf{E}_Q k(\mathsf{y}, \mathsf{y}') - 2\mathbf{E}_{\mathbf{P},\mathbf{Q}} k(\mathsf{x}, \mathsf{y})$$

Expectation Propagation

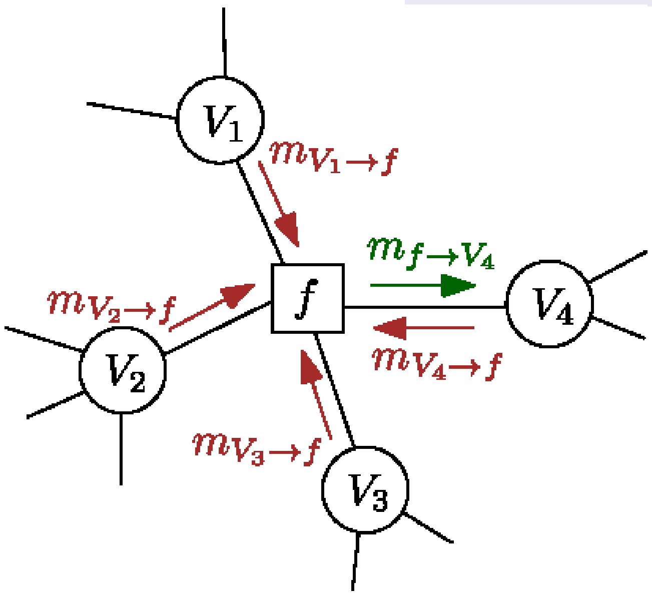
set of c variables connected to f

projected message

$$m_{f\to V_i}(v_i) = \frac{\text{proj}\left[\int d\mathcal{V}\backslash\{v_i\}\, f(\mathcal{V}) \prod_{j=1}^{c} m_{V_j\to f}(v_j)\right]}{m_{V_i\to f}(v_i)} := \frac{q_{f\to V_i}(v_i)}{m_{V_i\to f}(v_i)}$$

$\text{proj}[r_{f\to V_i}] := \arg\min_{q\in \text{ExpFam}} \text{KL}\,[r_{f\to V_i} \,\|\, q]$
(projection onto exponential family)

incoming message from V_j



- Expensive integral (besides special cases)
- **Goal:** Learn an *uncertainty aware* message operator (regression function)

$$\left[m_{V_j\to f}\right]_{j=1}^{c} \mapsto q_{f\to V_i}.$$

- **Challenges:** dealing with huge sample size, knowing when to consult expensive oracle.

Distribution regression using random Fourier features

Kernel representation by random Fourier features [Rahimi and Recht, 2008]

- Bochner's theorem: Continuous, translation-invariant kernel $k(a,b) = k(a-b)$ on $\mathbb{R}^m$ positive definite iff $\exists$ prob. meas. $\mathfrak{K}(\omega)$

$$k(a-b) = \mathbf{E}_{\omega\sim\mathfrak{K}}\mathbf{E}_{c\sim U[0,2\pi]}\left[2\cos(\omega^\top a + c)\cos(\omega^\top b + c)\right]$$

Distribution regression using random Fourier features

Kernel representation by random Fourier features [Rahimi and Recht, 2008]

- **Bochner's theorem:** Continuous, translation-invariant kernel $k(a,b) = k(a-b)$ on $\mathbb{R}^m$ positive definite iff $\exists$ prob. meas. $\mathfrak{K}(\omega)$

$$k(a-b) = \mathbf{E}_{\omega\sim\mathfrak{K}}\mathbf{E}_{c\sim U[0,2\pi]}\left[2\cos(\omega^\top a + c)\cos(\omega^\top b + c)\right]$$

- **Random features:** $\varphi_d(a) \in \mathbb{R}^d$ such that

$$k(a-b) \approx \varphi_d(a)^\top \varphi_d(b)$$

1. Draw i.i.d. $\{\omega_i\}_{i=1}^d \sim \mathfrak{K}(\omega)$.
2. Draw i.i.d. $\{c_i\}_{i=1}^d \sim U[0, 2\pi]$
3. $\varphi_d(a) = \sqrt{\frac{2}{d}}\left[\cos\left(\omega_1^\top a + c_1\right), \dots, \cos\left(\omega_d^\top a + c_d\right)\right]^\top \in \mathbb{R}^d$

Distribution regression using random Fourier features

- Given incoming messages $\mathsf{P} := m_{V_i \to f}$ and $\mathsf{Q} := m_{V_j \to f}$
- Approximate random Fourier mean embeddings:

$$\mu_{\mathsf{P},d} := \mathbf{E}_{\mathsf{x} \sim \mathsf{P}} \left[\varphi_d(\mathsf{x}) \right]$$

Distribution regression using random Fourier features

- Given incoming messages $\mathsf{P} := m_{V_i \to f}$ and $\mathsf{Q} := m_{V_j \to f}$
- Approximate random Fourier mean embeddings:

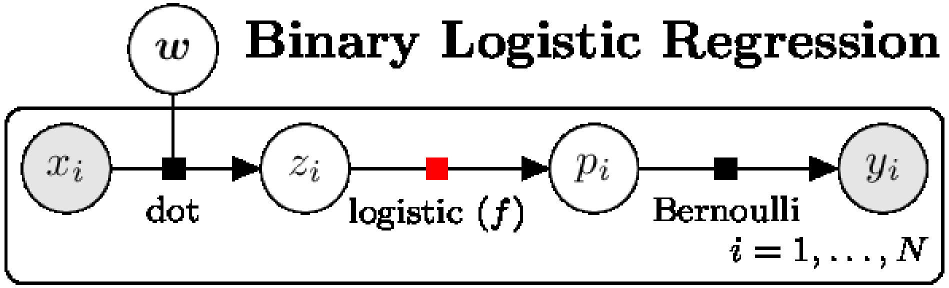
$$\mu_{\mathsf{P},d} := \mathbf{E}_{\mathsf{x} \sim \mathsf{P}} \left[\varphi_d(\mathsf{x}) \right]$$

- Approximate embeddings for kernel K on $\mu_{\mathsf{P}} \in \mathbb{R}^{d'}$:

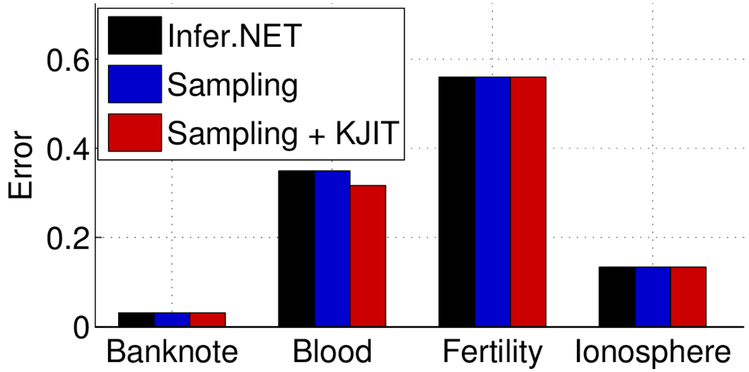
$$K_G(\mu_{\mathsf{P}}, \mu_{\mathsf{Q}}) \overset{1^{st}}{\approx} \underbrace{\exp\left(-\frac{\|\mu_{\mathsf{P},d} - \mu_{\mathsf{Q},d}\|_d^2}{2\gamma^2} \right)}_{\text{finite-dimensional Gaussian kernel}} \overset{2^{nd}}{\approx} \psi_{d'}(\mathsf{P})^\top \psi_{d'}(\mathsf{Q}).$$

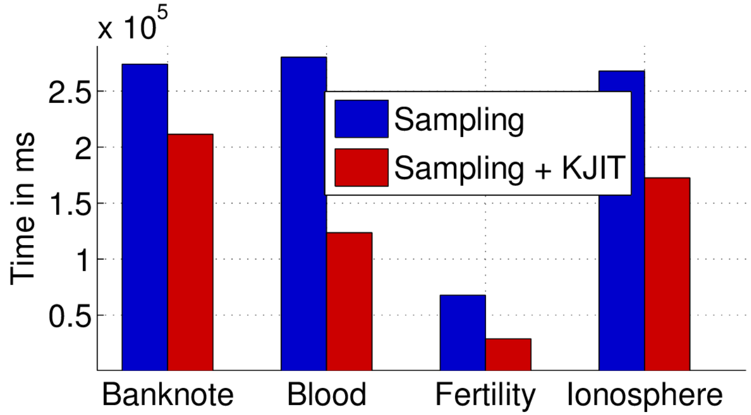
- Gaussian process regression directly on features $\psi_{d'}(\mathsf{P}) \in \mathbb{R}^{d'}$ [UAI15]
  - Bayesian uncertainty estimates tell us when to consult oracle
  - Efficient rank-1 updates, solution size constant as number of samples increases

Expectation Propagation for Classification



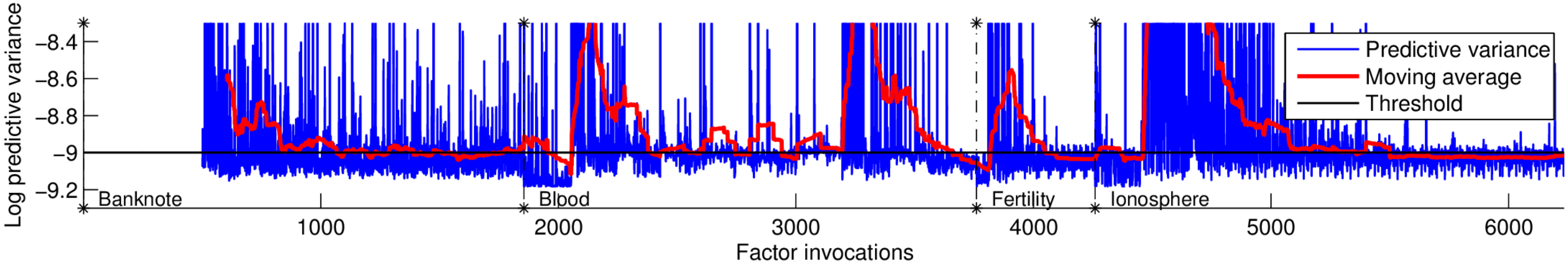
- Sequentially present 4 real datasets to the operator to learn.
- If predictive variance > threshold, ask oracle.



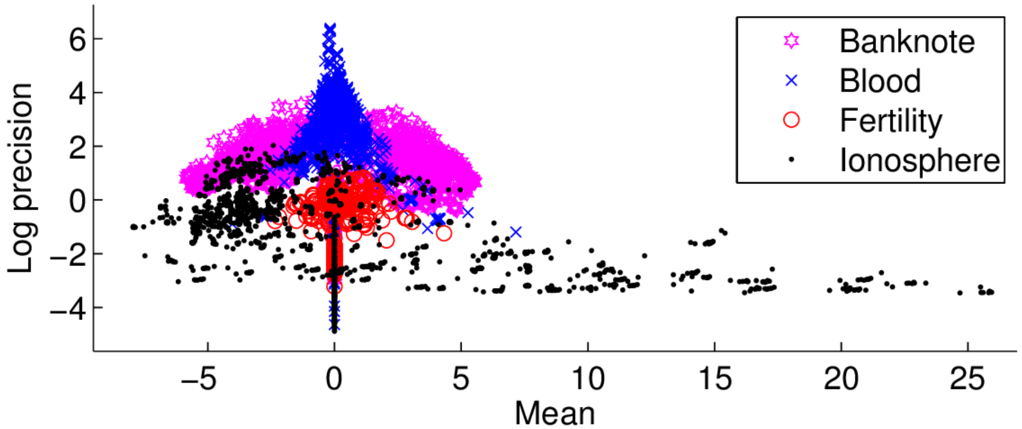


- **Left:** Binary classification error with learned posterior $\boldsymbol{w}$,
  Right: EP runtime.

Expectation Propagation for Classification



- Initial silent period = parameter selection + mini-batch training.
- $*$ = start of a new problem.
- Sharp rises after $*$ indicates ability to detect distribution (problem) change.



Distributions of $m_{z \to f}$ = Gaussian(z).

Regression using *population* mean embeddings

- Samples $\mathbf{z} := \{(\mu_{\mathsf{P}_i}, y_i)\}_{i=1}^{\ell} \overset{\text{i.i.d.}}{\sim} \rho(\mu_{\mathsf{P}}, y) = \rho(y|\mu_{\mathsf{P}})\rho(\mu_{\mathsf{P}})$,

$$\mu_{\mathsf{P}_i} = \mathbf{E}_{\mathsf{P}_i}[\varphi_{\mathsf{x}}]$$

- Regression function

$$f_\rho(\mu_{\mathsf{P}}) = \int_{\mathbb{R}} y \mathrm{d}\rho(y|\mu_{\mathsf{P}}),$$

Regression using *population* mean embeddings

- Samples $\mathbf{z} := \{(\mu_{\mathbf{P}_i}, y_i)\}_{i=1}^{\ell} \overset{\text{i.i.d.}}{\sim} \rho(\mu_{\mathbf{P}}, y) = \rho(y|\mu_{\mathbf{P}})\rho(\mu_{\mathbf{P}})$,

$$\mu_{\mathbf{P}_i} = \mathbf{E}_{\mathbf{P}_i}\left[\varphi_{\mathsf{x}}\right]$$

- Regression function

$$f_\rho(\mu_{\mathbf{P}}) = \int_{\mathbb{R}} y \mathrm{d}\rho(y|\mu_{\mathbf{P}}),$$

- Ridge regression for labelled distributions

$$f_{\mathbf{z}}^{\lambda} = \arg\min_{f \in \mathcal{H}} \frac{1}{\ell} \sum_{i=1}^{\ell} \left(f(\mu_{\mathbf{P}_i}) - y_i\right)^2 + \lambda \|f\|_{\mathcal{H}}^2, \quad (\lambda > 0)$$

- Define RKHS $\mathcal{H}$ with kernel $K(\mu_{\mathbf{P}}, \mu_{\mathbf{Q}}) := \langle \psi_{\mu_{\mathbf{P}}}, \psi_{\mu_{\mathbf{Q}}} \rangle_{\mathcal{H}}$: functions from $F \subset \mathcal{F}$ to $\mathbb{R}$, where

$$F := \{\mu_{\mathbf{P}} : \mathbf{P} \in \mathcal{P}\} \qquad \mathcal{P} \text{ set of prob. meas. on } \mathcal{X}$$

Regression using *population* mean embeddings

- Expected risk, Excess risk

$$\mathcal{R}\left[f\right] = \mathbf{E}_{\rho(\mu_{\mathbf{P}},y)}\left(f\left(\mu_{\mathbf{P}}\right) - y\right)^2 \qquad \mathcal{E}(f_{\mathbf{z}}^{\lambda}, f_\rho) = \mathcal{R}[f_{\mathbf{z}}^{\lambda}] - \mathcal{R}[f_\rho].$$

- Minimax rate [Caponnetto and Vito, 2007]

$$\mathcal{E}(f_{\mathbf{z}}^{\lambda}, f_\rho) = \mathcal{O}_p\left(\ell^{-\frac{bc}{bc+1}}\right) \qquad (1 < b, c \in (1,2]).$$

  - b size of input space, c smoothness of f_ρ

Regression using *population* mean embeddings

- Expected risk, Excess risk

$$\mathcal{R}[f] = \mathbf{E}_{\rho(\mu_{\mathsf{P}}, y)} \left(f(\mu_{\mathsf{P}}) - y\right)^2 \qquad \mathcal{E}(f_{\mathbf{z}}^{\lambda}, f_{\rho}) = \mathcal{R}[f_{\mathbf{z}}^{\lambda}] - \mathcal{R}[f_{\rho}].$$

- Minimax rate [Caponnetto and Vito, 2007]

$$\mathcal{E}(f_{\mathbf{z}}^{\lambda}, f_{\rho}) = \mathcal{O}_p\left(\ell^{-\frac{bc}{bc+1}}\right) \quad (1 < b, c \in (1,2]).$$

  - b size of input space, c smoothness of f_ρ

- Replace μ_{P_i} with $\hat{\mu}_{\mathsf{P}_i} = N^{-1}\sum_{j=1}^{N} \varphi_{x_j} \qquad x_j \overset{\text{i.i.d.}}{\sim} \mathsf{P}_i$

- Given $N = \ell^a \log(\ell)$ and $a = 2$, (and Hölder condition on $\psi : F \to \mathcal{H}$)

$$\mathcal{E}(f_{\hat{\mathbf{z}}}^{\lambda}, f_{\rho}) = \mathcal{O}_p\left(\ell^{-\frac{bc}{bc+1}}\right) \quad (1 < b, c \in (1,2]).$$

Same rate as for population μ_{P_i} embeddings! [AISTATS15, JMLR in revision]

Learning causal direction with mean embeddings

Additive noise model to direct an edge between random variables x and y

[Hoyer et al., 2009]

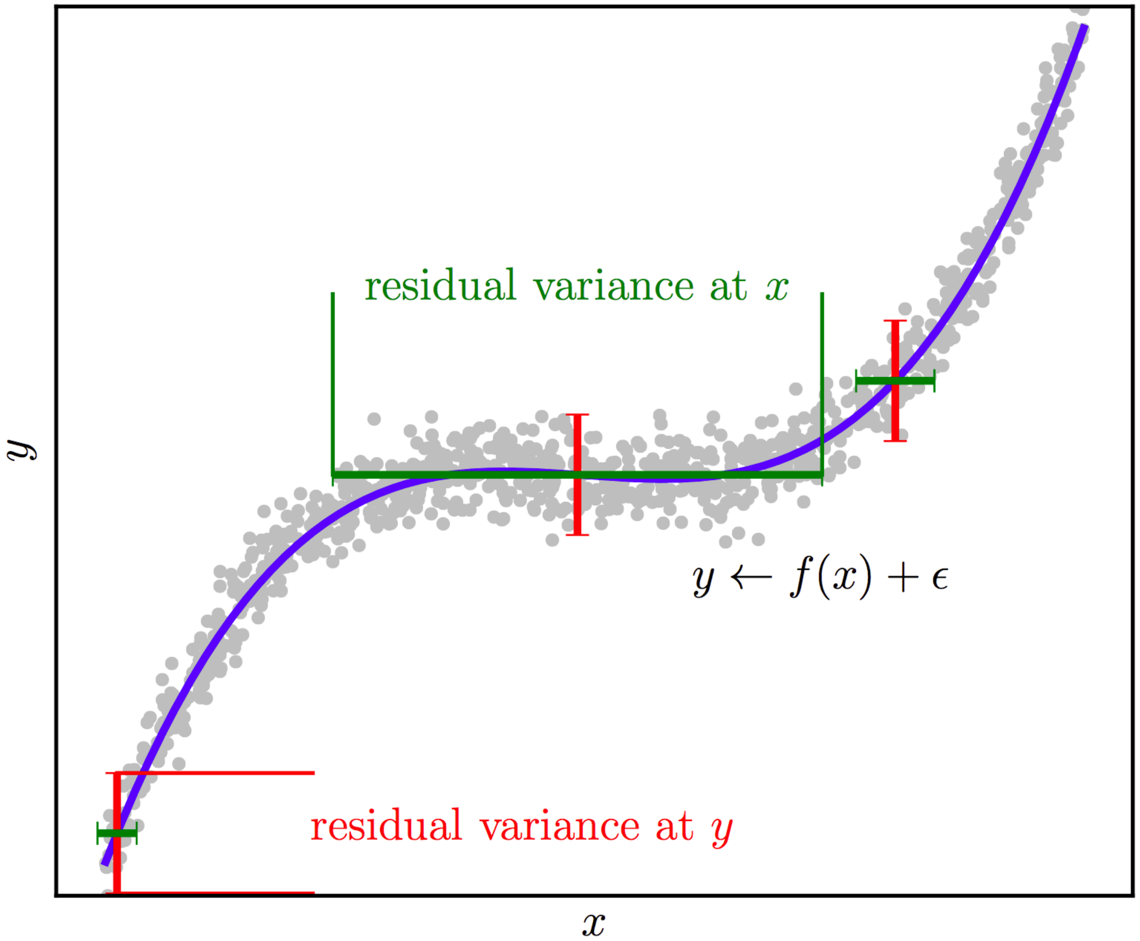


Figure: D. Lopez-Paz

Learning causal direction with mean embeddings

Classification of cause-effect relations [Lopez-Paz et al., 2015]

- Tuebingen cause-effect pairs: 82 scalar real-world examples where causes and effects known [Zscheischler, J., 2014]
- Training data: artificial, random nonlinear functions with additive gaussian noise.
- Features:
  $\hat{\mu}_{\mathbf{P}_x}, \hat{\mu}_{\mathbf{P}_y}, \hat{\mu}_{\mathbf{P}_{xy}}$
  with labels for $x \to y$ and $y \to x$
- Performance
  81% correct

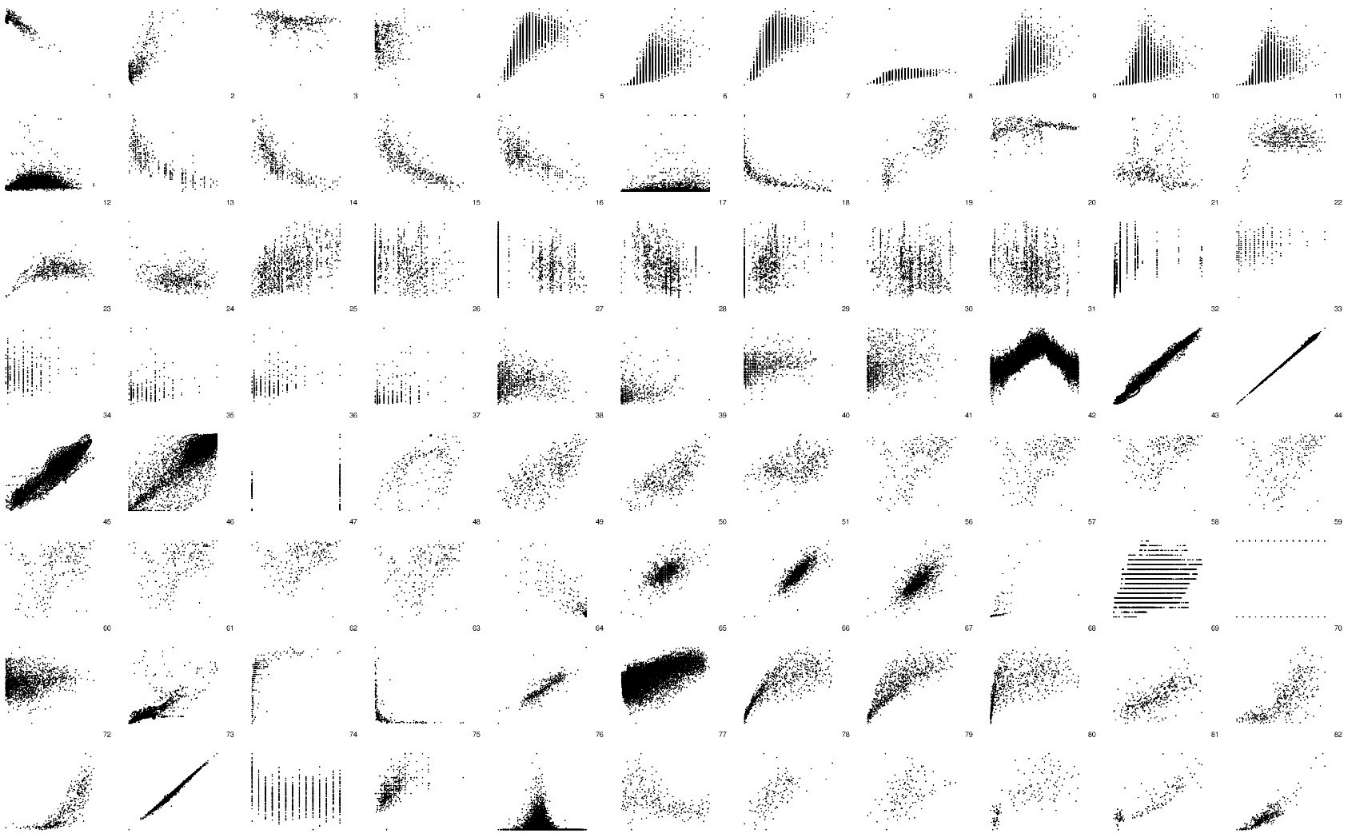


Figure:Mooij et al.(2015)

Overview

- Introduction to reproducing kernel Hilbert spaces
  - Kernels and feature spaces
  - Mapping probabilities to feature space

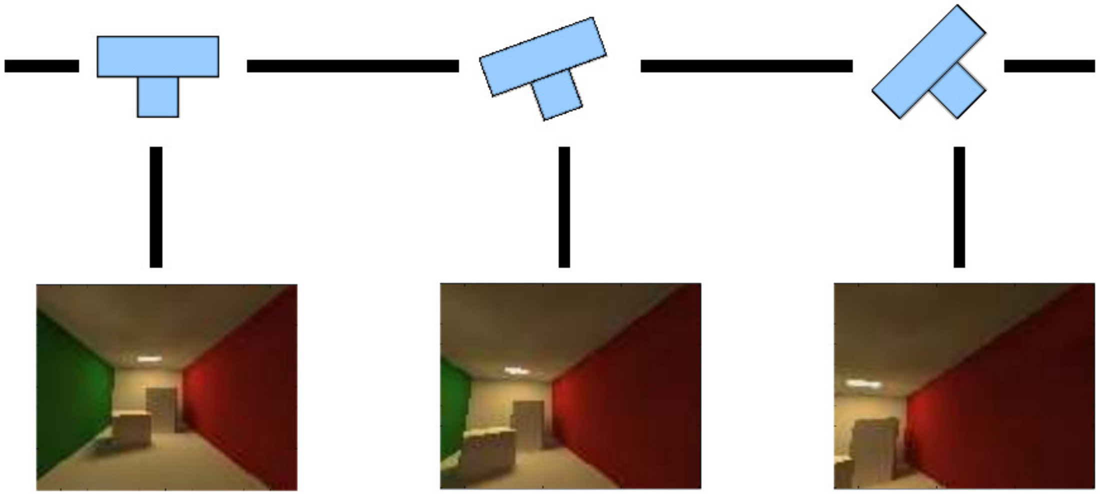
- Learning with distribution-valued inputs
  - Learning rates achievable when samples from disributions available [AISTATS15, JMLR in revision]
  - Approximate, uncertainty-aware regression with application to EP [UAI15]
  - Learning to predict direction of causality [Lopez-Paz et al., 2015]

Co-authors

- **From UCL:**
  - Steffen Grunewalder
  - Wittawat Jitkrittum
  - Guy Lever
  - Zoltan Szabo
- **External:**
  - Ali Eslami, Deepmind
  - Kenji Fukumizu, ISM
  - Nicolas Heess, Deepmind
  - Barnabas Poczos, CMU
  - Bernhard Schoelkopf, MPI
  - Dino Sejdinovic, Oxford
  - Alex Smola, Google/CMU
  - Le Song, Georgia Tech
  - Bharath Sriperumbudur, Penn. State

Learning when the outputs are distributions

Motivating example: Bayesian inference without a model



Challenges:

- No parametric model of camera dynamics (only samples)
- No parametric model of map from camera angle to image (only samples)
- Want to do filtering: Bayesian inference

Conditional distribution embedding

Bayes rule:

$$\mathbf{P}(y|x) = \frac{\mathbf{P}(x|y)\pi(y)}{\int \mathbf{P}(x|y)\pi(y)dy}$$

- $\mathbf{P}(x|y)$ is likelihood
- π is prior

How would this look with kernel embeddings?

Conditional distribution embedding

Bayes rule:

$$\mathbf{P}(y|x) = \frac{\mathbf{P}(x|y)\pi(y)}{\int \mathbf{P}(x|y)\pi(y)dy}$$

- $\mathbf{P}(x|y)$ is likelihood
- π is prior

How would this look with kernel embeddings?

Define RKHS $\mathcal{G}$ on $\mathcal{Y}$ with feature map ψ_y and kernel $l(y, \cdot)$

We need a conditional mean embedding: for all $g \in \mathcal{G}$,

$$\mathbf{E}_{Y|x^*} g(Y) = \langle g, \mu_{\mathbf{P}(y|x^*)} \rangle_{\mathcal{G}}$$

This will be obtained by RKHS-valued ridge regression

Ridge regression and the conditional feature mean

Ridge regression from $\mathcal{X} := \mathbb{R}^d$ to a finite *vector* output $\mathcal{Y} := \mathbb{R}^{d'}$ (these could be d' nonlinear features of y):

Define training data

$$X = \begin{bmatrix} x_1 & \ldots & x_m \end{bmatrix} \in \mathbb{R}^{d \times m} \qquad Y = \begin{bmatrix} y_1 & \ldots & y_m \end{bmatrix} \in \mathbb{R}^{d' \times m}$$

Ridge regression and the conditional feature mean

Ridge regression from $\mathcal{X} := \mathbb{R}^d$ to a finite *vector* output $\mathcal{Y} := \mathbb{R}^{d'}$ (these could be d' nonlinear features of y):

Define training data

$$X = \begin{bmatrix} x_1 & \dots & x_m \end{bmatrix} \in \mathbb{R}^{d \times m} \qquad Y = \begin{bmatrix} y_1 & \dots & y_m \end{bmatrix} \in \mathbb{R}^{d' \times m}$$

Solve

$$\breve{A} = \arg \min_{A \in \mathbb{R}^{d' \times d}} \left(\|Y - AX\|^2 + \lambda \|A\|_{\mathrm{HS}}^2 \right),$$

where

$$\|A\|_{\mathrm{HS}}^2 = \operatorname{tr}(A^\top A) = \sum_{i=1}^{\min\{d,d'\}} \gamma_{A,i}^2$$

Ridge regression and the conditional feature mean

Ridge regression from $\mathcal{X} := \mathbb{R}^d$ to a finite *vector* output $\mathcal{Y} := \mathbb{R}^{d'}$ (these could be d' nonlinear features of y):

Define training data

$$X = \begin{bmatrix} x_1 & \ldots & x_m \end{bmatrix} \in \mathbb{R}^{d \times m} \qquad Y = \begin{bmatrix} y_1 & \ldots & y_m \end{bmatrix} \in \mathbb{R}^{d' \times m}$$

Solve

$$\breve{A} = \arg \min_{A \in \mathbb{R}^{d' \times d}} \left(\|Y - AX\|^2 + \lambda \|A\|_{\mathrm{H}S}^2 \right),$$

where

$$\|A\|_{\mathrm{H}S}^2 = \mathrm{tr}(A^\top A) = \sum_{i=1}^{\min\{d,d'\}} \gamma_{A,i}^2$$

Solution: $\breve{A} = C_{YX} \left(C_{XX} + m\lambda I \right)^{-1}$

Ridge regression and the conditional feature mean

Prediction at new point x:

$$
\begin{aligned}
y^* &= \breve{A}x \\
&= C_{YX}\left(C_{XX} + m\lambda I\right)^{-1} x \\
&= \sum_{i=1}^{m} \beta_i(x) y_i
\end{aligned}
$$

where

$$
\beta_i(x) = (K + \lambda m I)^{-1} \left[\begin{array}{ccc} k(x_1, x) & \ldots & k(x_m, x) \end{array}\right]^\top
$$

and

$$
K := X^\top X \qquad\qquad k(x_1, x) = x_1^\top x
$$

Ridge regression and the conditional feature mean

Prediction at new point x:

$$\begin{aligned} y^* &= \breve{A}x \\ &= C_{YX}\left(C_{XX} + m\lambda I\right)^{-1} x \\ &= \sum_{i=1}^{m} \beta_i(x) y_i \end{aligned}$$

where

$$\beta_i(x) = (K + \lambda m I)^{-1} \left[\begin{array}{ccc} k(x_1, x) & \dots & k(x_m, x) \end{array} \right]^\top$$

and

$$K := X^\top X \qquad\qquad k(x_1, x) = x_1^\top x$$

What if we do everything in kernel space?

Ridge regression and the conditional feature mean

Recall our setup:

- Given training *pairs:*

$$(x_i, y_i) \sim \mathbf{P}_{XY}$$

- $\mathcal{F}$ on $\mathcal{X}$ with feature map φ_x and kernel $k(x, \cdot)$
- $\mathcal{G}$ on $\mathcal{Y}$ with feature map ψ_y and kernel $l(y, \cdot)$

We define the covariance between feature maps:

$$C_{XX} = \mathbf{E}_X \left(\varphi_X \otimes \varphi_X\right) \qquad C_{XY} = \mathbf{E}_{XY} \left(\varphi_X \otimes \psi_Y\right)$$

and matrices of feature mapped training data

$$X = \begin{bmatrix} \varphi_{x_1} & \ldots & \varphi_{x_m} \end{bmatrix} \quad Y := \begin{bmatrix} \psi_{y_1} & \ldots & \psi_{y_m} \end{bmatrix}$$

Ridge regression and the conditional feature mean

Objective: [Weston et al. (2003), Micchelli and Pontil (2005), Caponnetto and De Vito (2007), ICML12, ICML13]

$$\breve{A} = \arg \min_{A \in \mathrm{HS}(\mathcal{F},\mathcal{G})} \left(\mathbf{E}_{XY} \|Y - AX\|_{\mathcal{G}}^2 + \lambda \|A\|_{\mathrm{HS}}^2 \right), \qquad \|A\|_{\mathrm{HS}}^2 = \sum_{i=1}^{\infty} \gamma_{A,i}^2$$

Solution same as vector case:

$$\breve{A} = C_{YX} \left(C_{XX} + m\lambda I \right)^{-1},$$

Prediction at new x using kernels:

$$\begin{aligned} \breve{A}\varphi_x &= \left[\begin{array}{ccc} \psi_{y_1} & \ldots & \psi_{y_m} \end{array} \right] (K + \lambda m I)^{-1} \left[\begin{array}{ccc} k(x_1, x) & \ldots & k(x_m, x) \end{array} \right] \\ &= \sum_{i=1}^{m} \beta_i(x) \psi_{y_i} \end{aligned}$$

where $K_{ij} = k(x_i, x_j)$

Ridge regression and the conditional feature mean

How is loss $\|Y - AX\|_{\mathcal{G}}^2$ relevant to conditional expectation of some $\mathbf{E}_{Y|x} g(Y)$? Define: [Song et al. (2009), Grunewalder et al. (2013)]

$$\mu_{Y|x} := A\varphi_x$$

Ridge regression and the conditional feature mean

How is loss $\|Y - AX\|_{\mathcal{G}}^2$ relevant to conditional expectation of some $\mathbf{E}_{Y|x} g(Y)$? Define: [Song et al. (2009), Grunewalder et al. (2013)]

$$\mu_{Y|x} := A\varphi_x$$

We need A to have the property

$$\begin{aligned}\mathbf{E}_{Y|x} g(Y) &\approx \langle g, \mu_{Y|x} \rangle_{\mathcal{G}} \\ &= \langle g, A\varphi_x \rangle_{\mathcal{G}}\end{aligned}$$

Ridge regression and the conditional feature mean

How is loss $\|Y - AX\|_{\mathcal{G}}^2$ relevant to conditional expectation of some $\mathbf{E}_{Y|x} g(Y)$? Define: [Song et al. (2009), Grunewalder et al. (2013)]

$$\mu_{Y|x} := A\varphi_x$$

We need A to have the property

$$\begin{aligned}\mathbf{E}_{Y|x} g(Y) &\approx \langle g, \mu_{Y|x} \rangle_{\mathcal{G}} \\ &= \langle g, A\varphi_x \rangle_{\mathcal{G}}\end{aligned}$$

Natural risk function for conditional mean

$$\mathcal{R}(A, \mathsf{P}_{XY}) := \sup_{\|g\| \leq 1} \mathbf{E}_X \left[\underbrace{\left(\mathbf{E}_{Y|X} g(Y) \right)}_{\text{Target}} - \underbrace{\langle g, A\varphi_X \rangle_{\mathcal{G}}}_{\text{Estimator}} \right]^2,$$

Ridge regression and the conditional feature mean

The squared loss risk provides an upper bound on the natural risk.

$$\mathcal{R}(A, \mathbf{P}_{XY}) \leq \mathbf{E}_{XY} \|\psi_Y - A\varphi_X\|_{\mathcal{G}}^2$$

Ridge regression and the conditional feature mean

The squared loss risk provides an upper bound on the natural risk.

$$\mathcal{R}(A, \mathbf{P}_{XY}) \leq \mathbf{E}_{XY} \left\| \psi_Y - A\varphi_X \right\|_{\mathcal{G}}^2$$

Proof: Jensen

$$\begin{aligned}
\mathcal{R}(A, \mathbf{P}_{XY}) &:= \sup_{\|g\| \leq 1} \mathbf{E}_X \left[\left(\mathbf{E}_{Y|X} g(Y) \right) - \langle g, A\varphi_X \rangle_{\mathcal{G}} \right]^2, \\
&\leq \mathbf{E}_{XY} \sup_{\|g\| \leq 1} \left[g(Y) - \langle g, A\varphi_X \rangle_{\mathcal{G}} \right]^2
\end{aligned}$$

Ridge regression and the conditional feature mean

The squared loss risk provides an upper bound on the natural risk.

$$\mathcal{R}(A, \mathsf{P}_{XY}) \leq \mathbf{E}_{XY} \|\psi_Y - A\varphi_X\|_{\mathcal{G}}^2$$

Proof: Jensen

$$\begin{aligned}
\mathcal{R}(A, \mathsf{P}_{XY}) &:= \sup_{\|g\| \leq 1} \mathbf{E}_X \left[\left(\mathbf{E}_{Y|X} g(Y) \right) - \langle g, A\varphi_X \rangle_{\mathcal{G}} \right]^2, \\
&\leq \mathbf{E}_{XY} \sup_{\|g\| \leq 1} \left[g(Y) - \langle g, A\varphi_X \rangle_{\mathcal{G}} \right]^2 \\
&= \mathbf{E}_{XY} \sup_{\|g\| \leq 1} \left[\langle g, \psi_Y \rangle_{\mathcal{G}} - \langle g, A\varphi_X \rangle_{\mathcal{G}} \right]^2
\end{aligned}$$

Ridge regression and the conditional feature mean

The squared loss risk provides an upper bound on the natural risk.

$$\mathcal{R}(A, \mathsf{P}_{XY}) \leq \mathbf{E}_{XY} \left\| \psi_Y - A\varphi_X \right\|_{\mathcal{G}}^2$$

Proof: Jensen

$$\begin{aligned}
\mathcal{R}(A, \mathsf{P}_{XY}) &:= \sup_{\|g\| \leq 1} \mathbf{E}_X \left[\left(\mathbf{E}_{Y|X} g(Y) \right) - \langle g, A\varphi_X \rangle_{\mathcal{G}} \right]^2, \\
&\leq \mathbf{E}_{XY} \sup_{\|g\| \leq 1} \left[g(Y) - \langle g, A\varphi_X \rangle_{\mathcal{G}} \right]^2 \\
&= \mathbf{E}_{XY} \sup_{\|g\| \leq 1} \langle g, \psi_Y - A\varphi_X \rangle_{\mathcal{G}}^2
\end{aligned}$$

Ridge regression and the conditional feature mean

The squared loss risk provides an upper bound on the natural risk.

$$\mathcal{R}(A, \mathsf{P}_{XY}) \leq \mathbf{E}_{XY} \left\| \psi_Y - A\varphi_X \right\|_{\mathcal{G}}^2$$

Proof: Jensen

$$\begin{aligned}
\mathcal{R}(A, \mathsf{P}_{XY}) &:= \sup_{\|g\| \leq 1} \mathbf{E}_X \left[\left(\mathbf{E}_{Y|X} g(Y) \right) - \langle g, A\varphi_X \rangle_{\mathcal{G}} \right]^2, \\
&\leq \mathbf{E}_{XY} \sup_{\|g\| \leq 1} \left[g(Y) - \langle g, A\varphi_X \rangle_{\mathcal{G}} \right]^2 \\
&= \mathbf{E}_{XY} \sup_{\|g\| \leq 1} \langle g, \psi_Y - A\varphi_X \rangle_{\mathcal{G}}^2 \\
&= \mathbf{E}_{XY} \left\| \psi_Y - A\varphi_X \right\|_{\mathcal{G}}^2
\end{aligned}$$

Ridge regression and the conditional feature mean

The squared loss risk provides an upper bound on the natural risk.

$$\mathcal{R}(A, \mathsf{P}_{XY}) \leq \mathbf{E}_{XY} \left\|\psi_Y - A\varphi_X\right\|_{\mathcal{G}}^2$$

Proof: Jensen

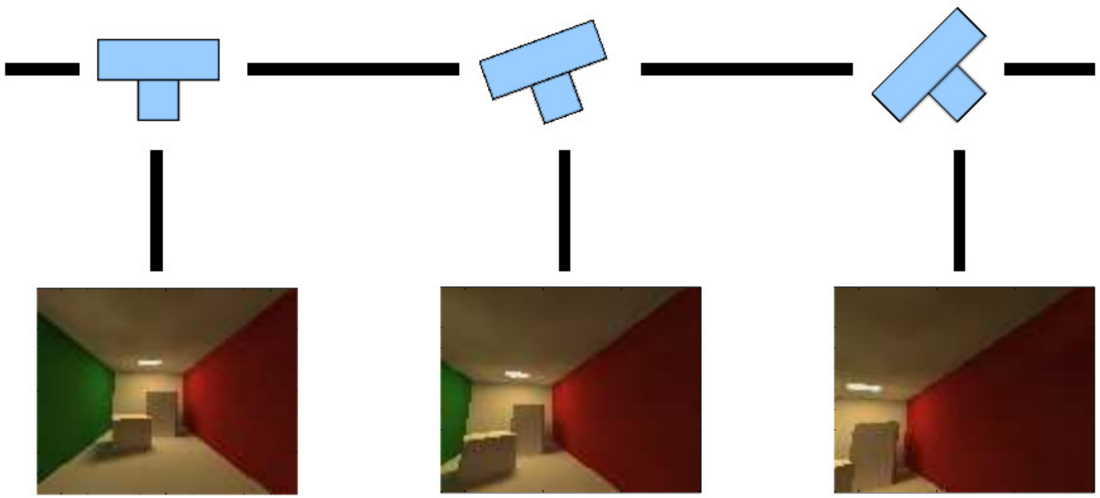
$$\begin{aligned}
\mathcal{R}(A, \mathsf{P}_{XY}) &:= \sup_{\|g\|\leq 1} \mathbf{E}_X \left[\left(\mathbf{E}_{Y|X} g(Y)\right) - \langle g, A\varphi_X\rangle_{\mathcal{G}}\right]^2, \\
&\leq \mathbf{E}_{XY} \sup_{\|g\|\leq 1} \left[g(Y) - \langle g, A\varphi_X\rangle_{\mathcal{G}}\right]^2 \\
&= \mathbf{E}_{XY} \sup_{\|g\|\leq 1} \langle g, \psi_Y - A\varphi_X\rangle_{\mathcal{G}}^2 \\
&= \mathbf{E}_{XY} \left\|\psi_Y - A\varphi_X\right\|_{\mathcal{G}}^2
\end{aligned}$$

If we assume $\mathbf{E}_Y[g(Y)|X = x] \in \mathcal{F}$ then upper bound tight

Kernel Bayes' law

- Prior: $Y \sim \pi(y)$
- Likelihood: $(X|y) \sim \mathbf{P}(x|y)$ from *training* distrib. $\mathbf{P}(x, y)$
- Joint distribution: $\mathbf{Q}(x, y) = \mathbf{P}(x|y)\pi(y)$

Warning: $\mathbf{Q} \neq \mathbf{P}$, *change of measure* from $\mathbf{P}(y)$ to $\pi(y)$



Kernel Bayes' law

- Prior: $Y \sim \pi(y)$
- Likelihood: $(X|y) \sim \mathbf{P}(x|y)$ from *training* distrib. $\mathbf{P}(x,y)$
- Joint distribution: $\mathbf{Q}(x,y) = \mathbf{P}(x|y)\pi(y)$

Warning: $\mathbf{Q} \neq \mathbf{P}$, *change of measure* from $\mathbf{P}(y)$ to $\pi(y)$

- Bayes' law: Want $\mu_{\mathbf{Q}(y|x)}$ with law

$$\mathbf{Q}(y|x) = \frac{\mathbf{P}(x|y)\pi(y)}{\mathbf{Q}(x)}$$

Kernel Bayes' law

- Posterior embedding via the usual conditional update,

$$\mu_{\mathbf{Q}(y|x)} = C_{\mathbf{Q}(y,x)} C_{\mathbf{Q}(x,x)}^{-1} \phi_x.$$

Kernel Bayes' law

- Posterior embedding via the usual conditional update,

$$\mu_{\mathbf{Q}(y|x)} = C_{\mathbf{Q}(y,x)} C^{-1}_{\mathbf{Q}(x,x)} \phi_x.$$

- Given mean embedding of prior: $\mu_\pi(y)$
- Learn marginal covariance by regression:

$$C_{\mathbf{Q}(x,x)} = \int (\varphi_x \otimes \varphi_x)\ \mathbf{P}(x|y)\pi(y)dxdy = C_{(xx)y} C^{-1}_{yy} \mu_{\pi(y)}$$

Kernel Bayes' law

- Posterior embedding via the usual conditional update,

$$\mu_{\mathbf{Q}(y|x)} = C_{\mathbf{Q}(y,x)} C^{-1}_{\mathbf{Q}(x,x)} \phi_x.$$

- Given mean embedding of prior: $\mu_\pi(y)$
- Learn marginal covariance by regression:

$$C_{\mathbf{Q}(x,x)} = \int (\varphi_x \otimes \varphi_x) \; \mathbf{P}(x|y)\pi(y)dxdy = C_{(xx)y} C^{-1}_{yy} \mu_{\pi(y)}$$

- Learn cross-covariance by regression:

$$C_{\mathbf{Q}(y,x)} = \int (\phi_y \otimes \varphi_x) \; \mathbf{P}(x|y)\pi(y)dxdy = C_{(yx)y} C^{-1}_{yy} \mu_{\pi(y)}.$$

Kernel Bayes' law: consistency result

- How to compute posterior expectation from data?
- Given samples: $\{(x_i, y_i)\}_{i=1}^n$ from P_{xy}, $\{(u_j)\}_{j=1}^n$ from prior π.
- Want to compute $\mathbf{E}[g(Y)|X = x]$ for g in $\mathcal{G}$
- For any $x \in \mathcal{X}$,

$$\left|\mathbf{g}_y^T R_{Y|X} \mathbf{k}_X(x) - \mathbf{E}[g(Y)|X = x]\right| = O_p(n^{-\frac{4}{27}}), \quad (n \to \infty),$$

where

  - $\mathbf{g}_y = (g(y_1), \ldots, g(y_n))^T \in \mathbb{R}^n$.
  - $\mathbf{k}_X(x) = (k(x_1, x), \ldots, k(x_n, x))^T \in \mathbb{R}^n$
  - $R_{Y|X}$ learned from the samples, contains the u_j

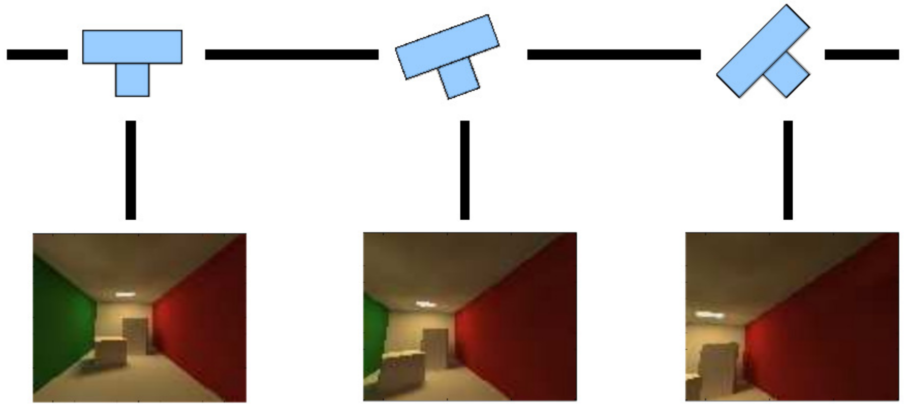
Smoothness assumptions:

- $\pi/p_Y \in \mathcal{R}(C_{YY}^{1/2})$, where p_Y p.d.f. of P_Y,
- $E[g(Y)|X = \cdot] \in \mathcal{R}(C^2_{\mathsf{Q}(xx)})$.

Experiment: Kernel Bayes' law vs EKF

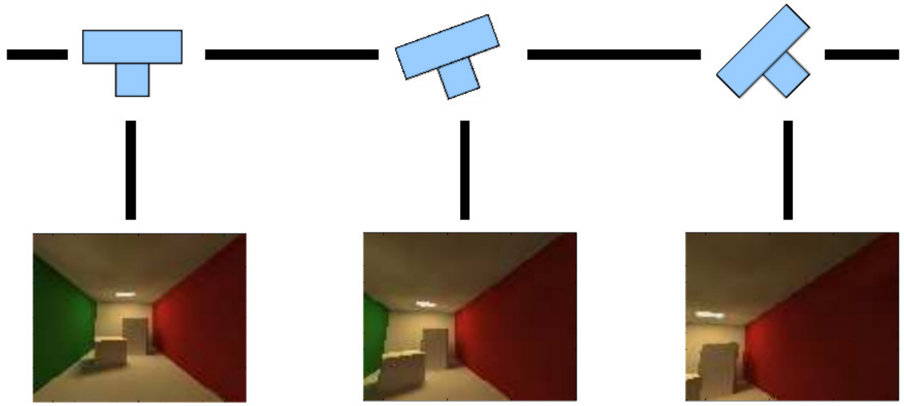
Experiment: Kernel Bayes' law vs EKF

- Compare with extended Kalman filter (EKF) on camera orientation task
- 3600 downsampled frames of 20×20 RGB pixels ($X_t \in [0,1]^{1200}$)
- 1800 training frames, remaining for test.
- Gaussian noise added to X_t.



Experiment: Kernel Bayes' law vs EKF

- Compare with extended Kalman filter (EKF) on camera orientation task
- 3600 downsampled frames of 20×20 RGB pixels ($X_t \in [0,1]^{1200}$)
- 1800 training frames, remaining for test.
- Gaussian noise added to X_t.



Average MSE and standard errors (10 runs)

	KBR (Gauss)	KBR (Tr)	Kalman (9 dim.)	Kalman (Quat.)
$\sigma^2 = 10^{-4}$	0.210 ± 0.015	0.146 ± 0.003	1.980 ± 0.083	0.557 ± 0.023
$\sigma^2 = 10^{-3}$	0.222 ± 0.009	0.210 ± 0.008	1.935 ± 0.064	0.541 ± 0.022

Selected references

Characteristic kernels and mean embeddings:

- Smola, A., Gretton, A., Song, L., Schoelkopf, B. (2007). A hilbert space embedding for distributions. ALT.
- Sriperumbudur, B., Gretton, A., Fukumizu, K., Schoelkopf, B., Lanckriet, G. (2010). Hilbert space embeddings and metrics on probability measures. JMLR.
- Gretton, A., Borgwardt, K., Rasch, M., Schoelkopf, B., Smola, A. (2012). A kernel two- sample test. JMLR.
- Sejdinovic, D., Sriperumbudur, B., Gretton, A., Fukumizu, K. (2013). Equivalence of distance-based and rkhs-based statistics in hypothesis testing. Annals of Statistics.

Two-sample, independence, conditional independence tests:

- Gretton, A., Fukumizu, K., Teo, C., Song, L., Schoelkopf, B., Smola, A. (2008). A kernel statistical test of independence. NIPS
- Fukumizu, K., Gretton, A., Sun, X., Schoelkopf, B. (2008). Kernel measures of conditional dependence.
- Gretton, A., Fukumizu, K., Harchaoui, Z., Sriperumbudur, B. (2009). A fast, consistent kernel two-sample test. NIPS.
- Gretton, A., Borgwardt, K., Rasch, M., Schoelkopf, B., Smola, A. (2012). A kernel two- sample test. JMLR

Selected references (continued)

Conditional mean embedding, RKHS-valued regression:

- Weston, J., Chapelle, O., Elisseeff, A., Schölkopf, B., and Vapnik, V., (2003). Kernel Dependency Estimation, NIPS.
- Micchelli, C., and Pontil, M., (2005). On Learning Vector-Valued Functions. Neural Computation.
- Caponnetto, A., and De Vito, E. (2007). Optimal Rates for the Regularized Least-Squares Algorithm. Foundations of Computational Mathematics.
- Song, L., and Huang, J., and Smola, A., Fukumizu, K., (2009). Hilbert Space Embeddings of Conditional Distributions. ICML.
- Grunewalder, S., Lever, G., Baldassarre, L., Patterson, S., Gretton, A., Pontil, M. (2012). Conditional mean embeddings as regressors. ICML.
- Grunewalder, S., Gretton, A., Shawe-Taylor, J. (2013). Smooth operators. ICML.

Kernel Bayes rule:

- Song, L., Fukumizu, K., Gretton, A. (2013). Kernel embeddings of conditional distributions: A unified kernel framework for nonparametric inference in graphical models. IEEE Signal Processing Magazine.
- Fukumizu, K., Song, L., Gretton, A. (2013). Kernel Bayes rule: Bayesian inference with positive definite kernels, JMLR

Conditions for ridge regression = conditional mean

Conditional mean obtained by ridge regression when $\mathbf{E}_Y[g(Y)|X = x] \in \mathcal{F}$

Given a function $g \in \mathcal{G}$. Assume $E_{Y|X}[g(Y)|X = \cdot] \in \mathcal{F}$. Then

$$\boxed{C_{XX} E_{Y|X}[g(Y)|X = \cdot] = C_{XY} g.}$$

Why this is useful:

$$\begin{aligned} E_{Y|X}[g(Y)|X = x] &= \langle E_{Y|X}[g(Y)|X = \cdot], \varphi_x \rangle_{\mathcal{F}} \\ &= \langle C_{XX}^{-1} C_{XY} g, \varphi_x \rangle_{\mathcal{F}} \\ &= \langle g, \underbrace{C_{YX} C_{XX}^{-1}}_{\text{regression}} \varphi_x \rangle_{\mathcal{G}} \end{aligned}$$

Conditions for ridge regression = conditional mean

Conditional mean obtained by ridge regression when $\mathbf{E}_Y[g(Y)|X = x] \in \mathcal{F}$

Given a function $g \in \mathcal{G}$. Assume $E_{Y|X}[g(Y)|X = \cdot] \in \mathcal{F}$. Then

$$\boxed{C_{XX} E_{Y|X}[g(Y)|X = \cdot] = C_{XY} g.}$$

Proof: [Fukumizu et al., 2004]

For all $f \in \mathcal{F}$, by definition of C_{XX},

$$\begin{aligned}
&\left\langle f, C_{XX} E_{Y|X}[g(Y)|X = \cdot] \right\rangle_{\mathcal{F}} \\
&= \operatorname{cov}\left(f, E_{Y|X}[g(Y)|X = \cdot]\right) \\
&= E_X\left(f(X)\, E_{Y|X}[g(Y)|X]\right) \\
&= E_{XY}(f(X)g(Y)) \\
&= \langle f, C_{XY} g \rangle,
\end{aligned}$$

by definition of C_{XY}.

References

A. Caponnetto and E. De Vito. Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7(3):331–368, 2007.

K. Fukumizu, F. R. Bach, and M. I. Jordan. Dimensionality reduction for supervised learning with reproducing kernel Hilbert spaces. *Journal of Machine Learning Research*, 5:73–99, 2004.

T. Gärtner, P. A. Flach, A. Kowalczyk, and A. J. Smola. Multi-instance kernels. In *Proceedings of the International Conference on Machine Learning*, pages 179–186. Morgan Kaufmann Publishers Inc., 2002.

David Haussler. Convolution kernels on discrete structures. Technical Report UCS-CRL-99-10, UC Santa Cruz, 1999.

P. Hoyer, D. Janzing, J. Mooij, J. Peters, and B. Schölkopf. Nonlinear causal discovery with additive noise models. In *NIPS*, 2009.

D. Lopez-Paz, K. Muandet, B. Schölkopf, and I. Tolstikhin. Towards a learning theory of cause-effect inference. In *ICML*, 2015.

A. Rahimi and B. Recht. Random features for large-scale kernel machines. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*. MIT Press, Cambridge, MA, 2008.