SPARSITY IN MULTIPLE KERNEL LEARNING

Ming Yuan

School of Industrial and Systems Engineering

Georgia Institute of Technology

myuan@isye.gatech.edu

http://www.isye.gatech.edu/~myuan

(Joint work with Vladimir Koltchinskii)

Outline

- Multiple kernel learning
  - Finite dimensional dictionaries – linear regression
  - Infinite dimensional dictionaries – additive model, functional ANOVA
- Sparse recovery with ℓ_1 regularization
  - General framework of sparse recovery
  - Excess risk bounds
  - Optimality
- Adaptive learning with multiple kernels
  - Double penalization
  - Adaptive tuning
- Conclusions

Problem of Prediction

- Input/output space: $\mathcal{X}$, $\mathcal{Y}$
- Training samples: $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathcal{X} \times \mathcal{Y}$, i.i.d. copies of $(X, Y) \sim P$
- Prediction: given $\mathbf{x} \in \mathcal{X}$, find a suitable $y \in \mathcal{Y}$

$$f_0 : \mathcal{X} \mapsto \mathcal{Y} : \mathbf{x} \mapsto f_0(\mathbf{x})$$

- Examples:
  - Regression: $f_0(X) = \mathbb{E}(Y|X)$
  - Classification: $f_0(X) = \mathrm{argmax}_y\, \mathbb{P}(Y = y|X)$
  - Generalized regression
  - ......

(Regularized) Empirical Risk Minimization

$$\operatorname*{argmin}_{f\in\mathcal{H}} \left[\mathbb{E}_n \ell(Y; f(X)) + J_\lambda(f)\right]$$

- Loss function: f_0 can be given as

$$\operatorname*{argmin}_{f} \mathbb{E}\ell(Y; f(X))$$

  - Regression – Least squares
  - Support vector machine – Hinge loss
- Model space: $\mathcal{H}$
  - Parametric – $\mathcal{H} = \{X^{\mathsf{T}}\beta\}$
  - Nonparametric – $\mathcal{H} = \mathcal{W}_2^2(X)$
- Penalty $J_\lambda(\cdot)$
  - Dimension too high, e.g., Lasso
  - Functional class too complicated, e.g., smoothing splines

Learning with Multiple RKHS

$$\mathcal{H} := \text{l.s.}\left\{\mathcal{H}_1 \bigcup \mathcal{H}_2 \bigcup \ldots\ldots \bigcup \mathcal{H}_d\right\}$$

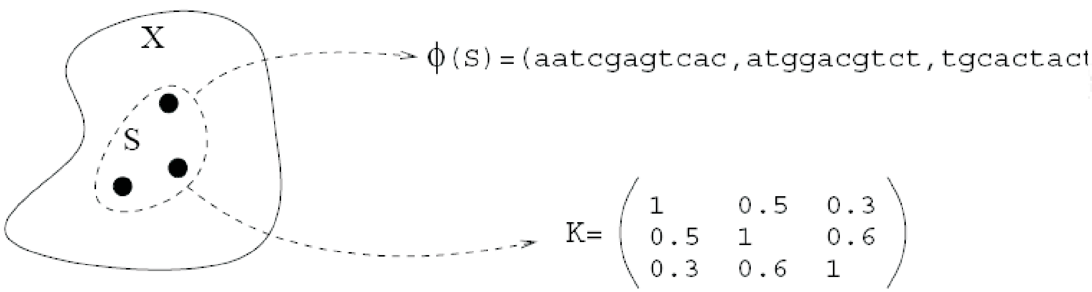
- Each $\mathcal{H}_j$ is a reproducing kernel Hilbert space
  - Normed linear functional space and $\mathcal{H}_j \to \mathbb{R}$: $f_j \mapsto f_j(\mathbf{x})$ is continuous
  - Equipped with a reproducing kernel K_j – $f_j(\mathbf{x}) = \langle f_j(\cdot), K_j(\mathbf{x}, \cdot)\rangle$
- Consists of all functions that have an additive representation

$$f = f_1 + \cdots + f_d, \qquad f_j \in \mathcal{H}_j,\ j = 1, \ldots, d$$

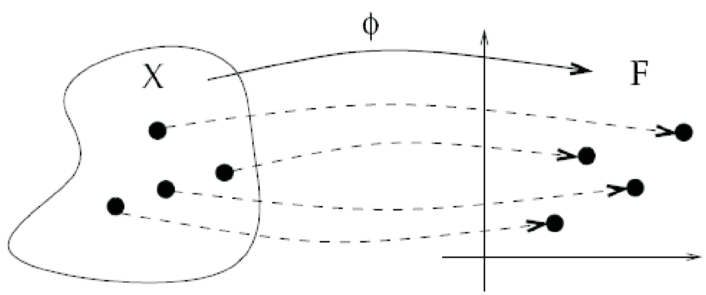
- Examples
  - Finite dimensional dictionaries – Linear regression
  - Infinite dimensional dictionaries – Additive models, Functional ANOVA...

LEARNING WITH MULTIPLE KERNELS

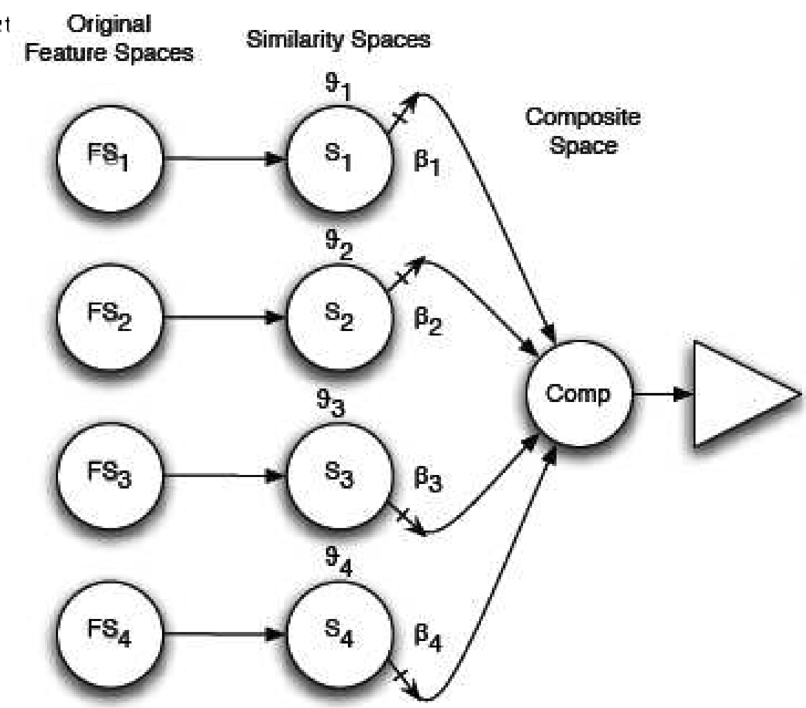
Moore-Aronszajn theorem – one-to-one correspondence between kernel and RKHS



Kernel

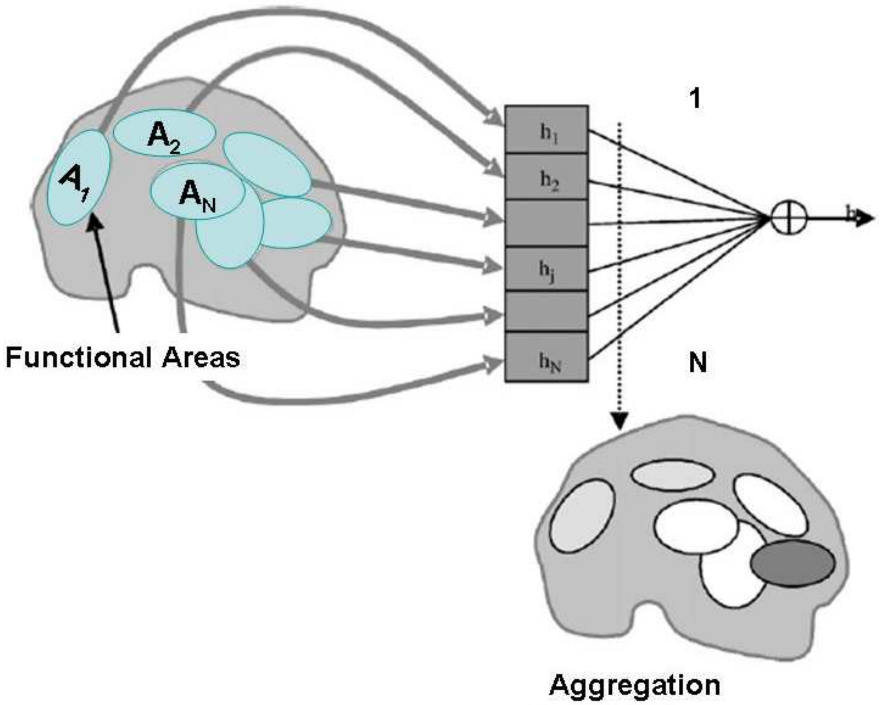


Feature Space

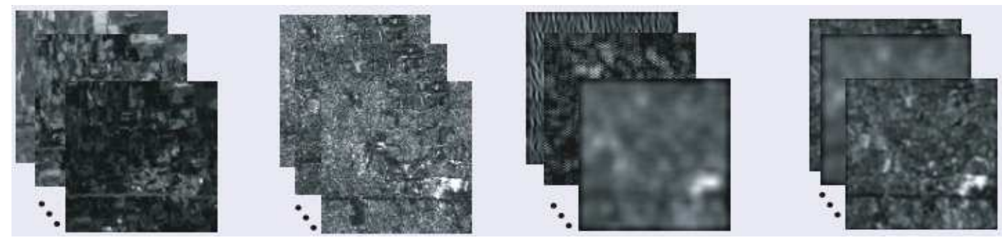


Learning with Multiple Kernels

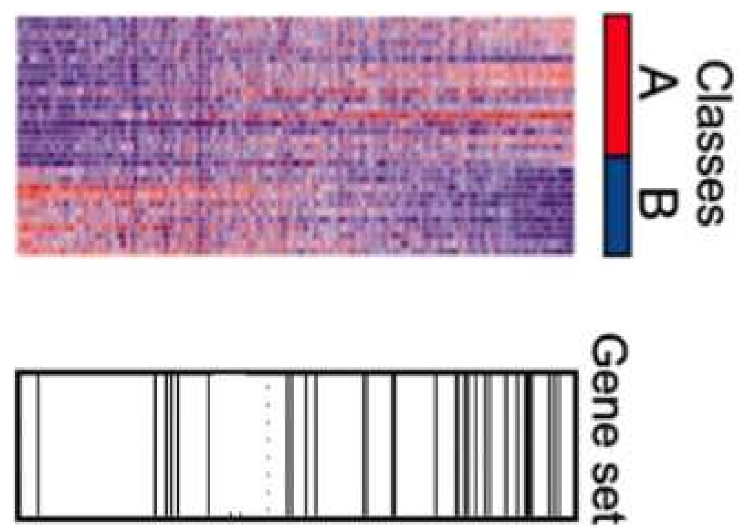
Motivating Examples



Functional MRI



Hyperspectral Imaging



Gene Set Analysis

Outline

- Multiple kernel learning
  - Finite dimensional dictionaries – linear regression
  - Infinite dimensional dictionaries – additive model, functional ANOVA
- Sparse recovery with ℓ_1 regularization
  - General framework of sparse recovery
  - Excess risk bounds
  - Optimality
- Adaptive learning with multiple kernels
  - Double penalization
  - Adaptive tuning
- Conclusions

ℓ_1 Type of Regularization

$$f = f_1 + \cdots + f_d, \qquad f_j \in \mathcal{H}_j,\ j = 1, \ldots, d$$

- $\mathcal{H}$ can be equipped with ℓ_1 type of norm

$$\|f\|_{\ell_1} := \|f\|_{\ell_1(\mathcal{H})} := \inf\left\{\sum_{j=1}^{d} \|f_j\|_{\mathcal{H}_j} : f = \sum_{j=1}^{d} f_j, f_j \in \mathcal{H}_j\right\}$$

- Sparse regularization

$$\hat{f}_\lambda := \operatorname*{argmin}_{f \in \mathcal{H}} \left[\mathbb{E}_n(\ell(Y, f(X)) + \lambda\|f\|_{\ell_1}\right]$$

ℓ_1 REGULARIZATION FOR LINEAR REGRESSION

$$\operatorname*{argmin}_{\beta} \left\{ \mathbb{E}_n \ell(Y, X^\mathsf{T}\beta) + \lambda \|\beta\|_{\ell_1} \right\}$$

- Nature of sparsity in high dimensional linear regression model
  - Apparent dimensionality – d
  - Intrinsic dimensionality (sparsity) – $s = \text{card}\{j : \beta_j \neq 0\}$
  - Sample size – n
- ℓ_1 regularization (Lasso) works in high dimensional setting

$$\text{RIP}(X \text{ is well} - \text{conditioned}) \Longrightarrow \|\hat{\beta} - \beta\|^2 = O_p\left(\frac{s \log d}{n}\right)$$

  - If we know which βs are zero – s/n
  - Additional price pay for not knowing – $\log d$

ℓ_1 Regularization for Additive Models

- COSSO (Lin and Zhang, 2006)

$$\left.\begin{array}{rl} \text{Lasso} & J_\lambda(g) = \lambda \sum_{j=1}^d |\beta_j| \\ \text{Splines} & J_\lambda(g) = \lambda \sum_{j=1}^d \|g_j\|^2_{\mathcal{W}_2^2} \end{array}\right\} \Longrightarrow J_\lambda(g) = \lambda \sum_{j=1}^d \|g_j\|_{\mathcal{W}_2^2}$$

- Spam (Ravikumar, Lafferty, Liu and Wasserman, 2008)

$$\left.\begin{array}{rl} \text{Group Lasso} & J_\lambda(g) = \lambda \sum_{j=1}^d \|\beta_j\| \\ \text{Basis Expansion} & g_j \in \text{ls}\{\phi_{j1}, \ldots, \phi_{jm}\} \end{array}\right\} \Longrightarrow J_\lambda(g) = \lambda \sum_{j=1}^d \|g_j\|_n$$

- Nonnegative Garrote (Yuan, 2008)
- Sparsity smoothness penalty (Meier, van de Geer and Bühlmann, 2009)
- Adaptive group Lasso (Huang, Horowitz and Wei, 2009)
- Screening (Jiang, Fan and Fan, 2010)
- ......

Multiple Kernel Learning

- "Aggregation" of kernels

$$\mathrm{conv}\{K_j : j = 1, \ldots, d\} := \left\{ \sum_{j=1}^{d} \theta_j K_j : c_j \geq 0, \sum_{j=1}^{d} \theta_j = 1 \right\}$$

- Kernel learning (Lanckriet et al., 2004; Micchelli and Pontil, 2005)

$$(\hat{f}_\lambda, \hat{K}_\lambda) := \underset{\substack{K \in \mathrm{conv}(K_j, j=1,\ldots,d) \\ f \in \mathcal{H}_K}}{\mathrm{argmin}} \left[\mathbb{E}_n(\ell(Y, f(X)) + \lambda \|f\|_K \right]$$

- Equivalence

$$\hat{f}_\lambda := \underset{f \in \mathcal{H}}{\mathrm{argmin}} \left[\mathbb{E}_n(L(Y, f(X)) + \lambda \underbrace{\min_{K \in \mathrm{conv}(K_j, j=1,\ldots,d)} \|f\|_K}_{\Downarrow} \right]$$

$$\|f\|_{\ell_1(\mathcal{H})} = \inf \{ \|f\|_K : K \in \mathrm{conv}\{K_j : j = 1, \ldots, d\} \}$$

AND BEYOND . . .

- Partially linear model
  - Linear component space – $\mathcal{H}_j$ univariate linear functions for $j = 1, \ldots, d_1$
  - Nonparametric component space – $\mathcal{H}_j$ infinite dimensional for $j > d_1$
  - ℓ_1 regularization

$$\operatorname*{argmin}_{\substack{\beta \in \mathbb{R}^{d_1} \\ f \in \mathcal{H}_2(X_2)}} \left[\mathbb{E}_n(\ell(Y, X_1^\top \beta + f(X_2)) + \lambda \left(\|f\|_{\ell_1} + \|\beta\|_{\ell_1}\right)\right]$$

- Varying coefficient model
  - Components space – $\mathcal{H}_j = \{f(X)Z_j : f \in \mathcal{H}_j^0\}$
  - ℓ_1 regularization

$$\operatorname*{argmin}_{f \in \mathcal{H}} \left[\mathbb{E}_n(\ell(Y, f(X)) + \lambda \sum_{j=1}^{d} \|f_j\|_{\mathcal{H}_j^0}\right]$$

OUTLINE

- Multiple kernel learning
  - Finite dimensional dictionaries – linear regression
  - Infinite dimensional dictionaries – additive model, functional ANOVA
- Sparse recovery with ℓ_1 regularization
  - General framework of sparse recovery
  - Excess risk bounds
  - Optimality
- Adaptive learning with multiple kernels
  - Double penalization
  - Adaptive tuning
- Conclusions

EXCESS RISK

- Convex loss ℓ such that $f_0 = \text{argmin}_f \, \mathbb{E}\ell(Y, f(X))$
  - Regression: $\mathcal{Y} = \mathbb{R}$, $\ell(y,u) := \phi(y-u)$ – ϕ even and $\phi(0) = 0$
  - Classification: $\mathcal{Y} = \{\pm 1\}$, $\ell(y,u) := \phi(yu)$ – $\phi'(0) < 0$
- Excess risk

$$\begin{aligned} \mathcal{E}(f) &= \mathbb{E}[\ell(Y, f(X))] - \min_f \mathbb{E}[\ell(Y, f(X))] \\ &= \mathbb{E}[\ell(Y, f(X))] - \mathbb{E}[\ell(Y, f_0(X))] \end{aligned}$$

  - Example – squared loss

$$\mathcal{E}(f) = \|f - f_0\|^2_{\mathcal{L}_2(\Pi_X)} := \mathbb{E}[f(X) - f_0(X)]^2$$

Excess risk bounds

- Finite dimensional dictionary (parametric) – $\dim(\mathcal{H}_j) \leq V$

$$\left.\begin{array}{r}\text{Generalized RIP} \\ \lambda \sim (n^{-1}\log d)^{1/2}\end{array}\right\} \Longrightarrow \mathcal{E}(\hat{f}) = O_p\left(\frac{s(V+\log d)}{n}\right)$$

- Infinite dimensional dictionary (nonparametric) – $\dim(\mathcal{H}_j) = \infty$

$$\left.\begin{array}{r}\text{Generalized RIP} \\ \lambda \sim (n^{-1}\log d)^{1/2}\end{array}\right\} \Longrightarrow \mathcal{E}(\hat{f}) = O_p\left(s\sqrt{\frac{\log d}{n}}\right)$$

Example – Group Lasso

- $X = (X_1, \ldots, X_d)^{\mathsf{T}}$ where $X_j \in \mathbb{R}^V$, then

$$\mathcal{E}(\hat{f}^{\text{GroupLasso}}) = O_p\left(\frac{s(V + \log d)}{n}\right)$$

  - s – Group sparsity

- If applying Lasso without group structure

$$\mathcal{E}(\hat{f}^{\text{Lasso}}) = O_p\left(\frac{\tilde{s}\log(dV)}{n}\right)$$

  - $\tilde{s}$ – individual sparsity

- Advantage of Group Lasso
  - No loss in rate – $\tilde{s} \geq s$
  - Could gain substantially – $\tilde{s} = sV$

EXAMPLE – ADDITIVE MODELS

$$\underset{f \in \mathcal{H}}{\operatorname{argmin}} \left\{ \mathbb{E}_n \ell(Y, f(X)) + \lambda \|f\|_{\ell_1} \right\}$$

- Smoothness index α – $\lambda_m(K_j) \sim m^{-2\alpha}$ (e.g., Sobolev space of order α)
- Sparsity s – $\mathrm{card}(\mathrm{supp}(f)) = s$ where $\mathrm{supp}(f) = \{j : f_j \neq 0\}$
- Assume that
  - $\{X_j : j \in \mathrm{supp}(f)\}$ are not too similar
  - $\{X_j : j \in \mathrm{supp}(f)\}$ and $\{X_j : j \notin \mathrm{supp}(f)\}$ are not too similar
- Then

$$\lambda \sim (n^{-1} \log p)^{1/2} \Longrightarrow \mathcal{E}(\hat{f}) = O_p\left(s \sqrt{\frac{\log d}{n}} \right)$$

PARAMETRIC VS NONPARAMETRIC

- If s is finite, consistent estimate with ℓ_1 regularization iff $\log d = o(n)$
  - Parametric – $s \ll n(\log d)^{-1}$
  - Nonparametric – $s \ll n^{1/2}(\log d)^{-1/2}$
- Sample size calculation
  - Parametric – $n \gg s \log d$
  - Nonparametric – $n \gg s^2 \log d$

No effect of smoothness $\Longrightarrow$ Optimality for nonparametric case??

OUTLINE

- Multiple kernel learning
  - Finite dimensional dictionaries – linear regression
  - Infinite dimensional dictionaries – additive model, functional ANOVA
- Sparse recovery with ℓ_1 regularization
  - General framework of sparse recovery
  - Excess risk bounds
  - Optimality
- Adaptive learning with multiple kernels
  - Double penalization
  - Adaptive tuning
- Conclusions

IDEALIZED MODEL

- Additive model but know apriori that
  - X_js are independent
  - Direct observation on each component function

$$dY_j(t) = f_j(t)dt + \sigma dW_j(t)$$

- Optimal rate for ℓ_1 regularization
  - Ultra-high dimensional $d \sim \exp(n^\gamma)$ and s is finite

$$\inf_\lambda \mathcal{E}(\hat{f}) \sim (\log d/n)^{1/2} \quad \text{(rate cannot be improved)}$$

  - High dimensional $d \sim n^\gamma$ and s is finite

$$\inf_\lambda \mathcal{E}(\hat{f}) \sim \begin{cases} n^{-\boxed{\frac{2\alpha}{2\alpha+1}}+\boxed{\frac{\gamma(2\alpha-1)}{2\alpha+1}}} & \text{if } \gamma \le \frac{1}{2} \\ (\log d/n)^{1/2} & \text{if } \gamma > \frac{1}{2} \end{cases} \quad \text{(phase transition)}$$

MINIMAX OPTIMALITY

$$\inf_{\tilde{f}(\cdot;\text{data})} \sup_{f \in \mathcal{H}; \text{supp}(f) \leq s} \mathcal{E}(\tilde{f}) \sim s \left(\underbrace{n^{-\frac{2\alpha}{2\alpha+1}}}_{\text{effect of smoothing}} + \underbrace{n^{-1} \log d}_{\text{effect of high dim}} \right)$$

- When $\log d \ll n^{1/(2\alpha+1)}$

$$\inf_{\tilde{f}(\cdot;\text{data})} \sup_{f \in \mathcal{H}; \text{supp}(f) \leq s} \mathcal{E}(\tilde{f}) \sim s n^{-\frac{2\alpha}{2\alpha+1}}$$

- When $\log d \ll n^{1/(2\alpha+1)}$

$$\inf_{\tilde{f}(\cdot;\text{data})} \sup_{f \in \mathcal{H}; \text{supp}(f) \leq s} \mathcal{E}(\tilde{f}) \sim s n^{-1} \log d$$

OUTLINE

- Multiple kernel learning
  - Finite dimensional dictionaries – linear regression
  - Infinite dimensional dictionaries – additive model, functional ANOVA
- Sparse recovery with ℓ_1 regularization
  - General framework of sparse recovery
  - Excess risk bounds
  - Optimality
- Adaptive learning with multiple kernels
  - Double penalization
  - Adaptive tuning
- Conclusions

Double Penalization

- ℓ_1 regularization serves two purposes simulataneously
  - For smoothing – $\lambda \sim n^{-2\alpha/(2\alpha+1)}$
  - For sparsity – $\lambda \sim (n^{-1}\log d)^{1/2}$
- Minimax optimal approach – double penalization

$$\hat{f}_\lambda := \underset{f\in\mathcal{H}}{\operatorname{argmin}} \left[\mathbb{E}_n(\ell(Y, f(X)) + \underbrace{\lambda_1 \sum_{j=1}^d \|f_j\|^2_{\mathcal{H}_j}}_{\text{for smoothing}} + \underbrace{\lambda_2 \sum_{j=1}^d \|f_j\|_{\mathcal{L}_2(\Pi_n)}}_{\text{for sparsity}}\right]$$

- Tuning

$$\lambda_1 = \lambda_2^2 \sim n^{-2\alpha/(2\alpha+1)} + n^{-1}\log d \Longrightarrow \mathcal{E}(\hat{f}) \sim s\left(n^{-2\alpha/(2\alpha+1)} + n^{-1}\log d\right)$$

LEARNING WITH KERNELS – ADAPTIVITY

- In additive models, α identifies with smoothness – modeling assumption
- In general, α is determined by the decay rate of eigenvalues of a kernel

$$\int K(s,t)\psi_m(s)d\Pi_X(s) = \lambda_m\psi_m(t) \Longrightarrow \lambda_m \sim m^{-2\alpha}$$

- $X \in \mathbb{R}^{d_0}$ and $\mathcal{H}$ is Sobolev space of order β – $K(s,t) = k(s-t)$, where

$$\mathcal{F}(k)_m = (\|m\|^2 + 1)^{-\beta}, \qquad m \in \mathbb{Z}^{d_0}$$

- Then $\alpha = \beta/d_0$, leading to

$$\text{optimal rate of convergence} \qquad n^{-2\beta/(2\beta+d_0)}$$

- $\text{supp}(\Pi_X) \subset \mathbb{R}^{d_1}$ where $d_1 < d_0$, then $\alpha = (\beta - (d_0 - d_1)/2)/d_1$

$$\text{optimal rate of convergence} \qquad n^{-(2\beta-(d_0-d_1))/(2\beta-d_0+2d_1)}$$

α is not known even if K_js are known

Adaptive Tuning

- Gram matrix

$$G_j = \left(n^{-1} K_j(X_i, X_l)\right)_{n\times n}$$

- Eigenvalue decomposition $\hat{\rho}_1 \geq \hat{\rho}_2 \geq \ldots$
- $\lambda_j = c\hat{\eta}(K_j) \sim n^{-2\alpha/(2\alpha+1)}$

$$\hat{\eta}(K_j) = \left\{ \eta \geq (n^{-1}\log p)^{1/2} : \left(\frac{1}{n} \sum_{k\geq 1} \hat{\rho}_k \wedge \delta^2 \right)^{1/2} \leq \eta\delta + \eta^2, \forall \delta \in [0,1] \right\}$$

- Choice motivated by study of Rademacher process (Mendelson, 2002)
- Excess risk bound

$$\mathcal{E}(\hat{f}) \leq Cs \left(n^{-2\alpha/(2\alpha+1)} + \frac{\log d}{n} \right)$$

Summary

- A number of common techniques can be formulated in a unified framework
- The unified framework gives insight to the connection among methods and allows systematic study of different methods
- Sparse recovery is possible with ℓ_1 type regularization if $\log d = o(n)$ for a large class of model
- Similarity and difference between finite and infinite dimensional dictionaries
- More efficient approach with double penalization separating model selection from smoothing