Statistical Foundations of Data Science

Jianqing Fan

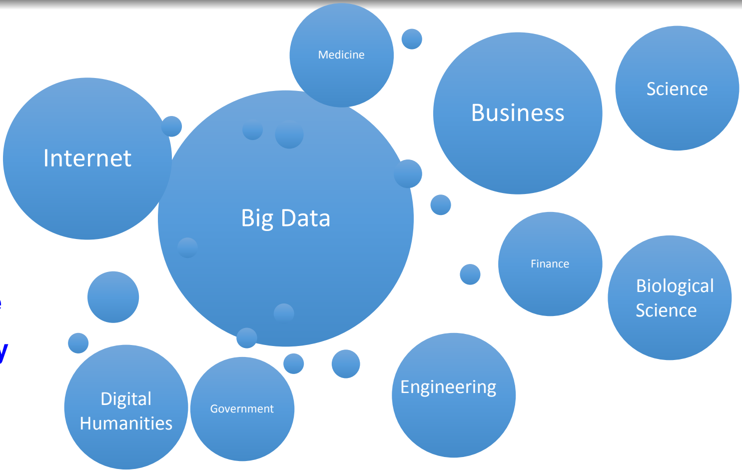
Princeton University

https://fan.princeton.edu

ZOOM ID Lectures: **970 4936 8998** Office Hours: **996 4030 7631**

Annotated Lecture Notes: web view

Big Data are ubiquitous



★**Volume**

★**Velocity**

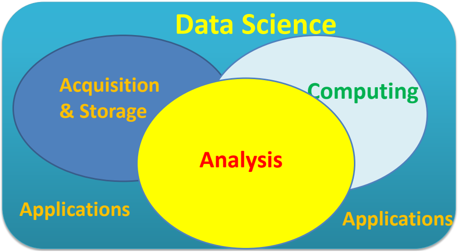
★**Variety**

"There were 5 exabytes of information created between the dawn of civilization through 2003, but that much information is now created every 2 days" — Eric Schmidt, CEO of Google

Deep Impact of Data Tsunami

System: storage, communication, computation architectures

Analysis: statistics, computation, optimization, privacy



Big Data $\Longrightarrow$ Smart Data

What can big data do?

Hold great promises for understanding

- ★ **Heterogeneity**: personalized medicine or services
- ★ **Commonality**: in presence of large variations (noises)

from large pools of variables, factors, genes, environments and their interactions as well as **latent factors**.

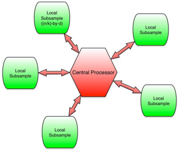
Aims of Data Science:

- **Prediction**: To construct as effective a method as possible to predict future observations.(**correlation**)
- **Inference and Prediction**: To gain insight into relationship between features and response for scientific purposes and to construct an improved prediction method. (**causation**)

Common Features and Techniques

Common Features of Big Data:

- ★ Dependence, heavy tails, endogeneity, spurious corr, heterogeneity,
- ♠ Missing data, measurement errors, survivor, sampling biases
- ♣ Computation, communication, privacy, ownership



Common Techniques for Data Science:

- ★ Statistical Techniques: Least-Squares, MLE, M-estimation
- ♠ Regression: Parametric, Nonparametric, Sparse, Factor(PCR)
- ♣ Principal Component Analysis: Supervised, unsupervised.

1. Multiple and Nonparametric Regression

1.1. Multiple Regression

■ Read materials and R-implementations here

https://fan.princeton.edu/fan/classes/245/chap11.pdf

Purpose of Multiple regression

- ★ Study assocations between dependent & independent variables
- ★ Screen irrelevant and select useful variables
- ★ Prediction

Example: Zillow is an online real estate database company founded in 2006. An important task for Zillow is to predict the house price. (Training data: 15129 cases, testing data: 6484 cases)

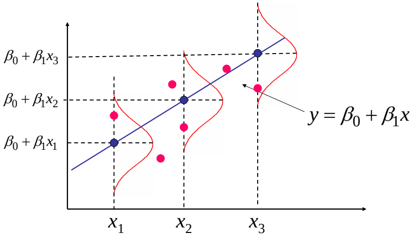
Interest: Associations between **housing** and its **attributes**.

- **Response** Y = Housing prices
- **Covariates**
  - No. of bathrooms X_1; No. of bedrooms X_2
  - sqft-living room X_3; sqft-lot X_4
  - zipcode X_5 (70 zipcodes); view X_6 (5 categories)
  - $\dots$

Multiple linear regression model

$$Y = \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \varepsilon$$

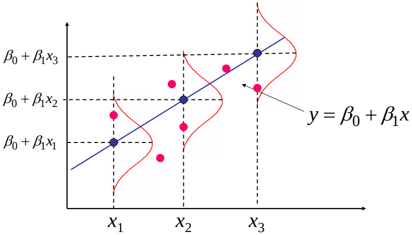
- Y: response / dependent variable
- X_j's: explanatory / independent variables or covariates
- ε: random error not explained / predicted by covariates
- include intercept by setting $X_1 = 1$



Method of least-squares

Data: $\left\{\left(x_{i1}, x_{i2}, \cdots x_{ip}, y_i\right)\right\}_{1\le i\le n}$

Model: $y_i = \sum_{j=1}^{p} \beta_j x_{ij} + \varepsilon_i$



Method of Least-Squares:

$$\text{minimize}_{\beta\in\mathbb{R}^p} \qquad \text{RSS}\left(\beta\right) \triangleq \sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2$$

- RSS stands for **residual sum-of-squares**
- When $\varepsilon_i \overset{\text{i.i.d}}{\sim} \mathcal{N}(0,\sigma^2)$, least-squares estimator is MLE

Regression in matrix notation

$$\mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \mathbf{X} = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ldots & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix}, \beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

Model becomes

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon$$

RSS becomes

$$\mathrm{RSS}(\beta) = \|\mathbf{y} - \mathbf{X}\beta\|^2$$

Closed-form solution

Least-squares: Minimize wrt $\beta \in \mathbb{R}^p$

$$\|\mathbf{y} - \mathbf{X}\beta\|^2 = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta)$$

Setting gradients to zero yields **normal equations**:

$$\mathbf{X}^T\mathbf{y} = \mathbf{X}^T\mathbf{X}\beta$$

Least-squares estimator: (assume $\mathbf{X}$ has full column rank)

$$\widehat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

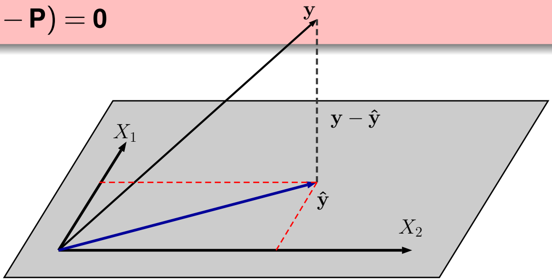
Geometric interpretation of least-squares

Fitted value: $\widehat{\mathbf{y}} = \mathbf{X}\widehat{\beta} = \underbrace{\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T}_{\triangleq \mathbf{P} \in \mathbb{R}^{n\times n}}\mathbf{y}$

Theorem 1.1 [Property of projection matrix]

★ $\mathbf{P}\mathbf{x}_j = \mathbf{x}_j, \quad j = 1, 2, \cdots, p$

★ $\mathbf{P}^2 = \mathbf{P}$ or $\mathbf{P}(\mathbf{I}_n - \mathbf{P}) = \mathbf{0}$



■ project response vector **y** onto linear space spanned by **X**

Statistical properties of least-squares estimator

Assumption:

- **Exogeneity**: $\mathbb{E}(\varepsilon|\mathbf{X}) = 0$;
- **Homoscedasticity**: $\text{var}(\varepsilon|\mathbf{X}) = \sigma^2$.

Statistical Properties:

★ **bias**: $\mathbb{E}(\widehat{\beta}|\mathbf{X}) = \beta$

★ **variance**: $\text{var}(\widehat{\beta}|\mathbf{X}) = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}$

often dropped

■ Recall $\text{cov}(\mathbf{U},\mathbf{V}) = E(\mathbf{U}-\mu_u)(\mathbf{V}-\mu_v)^T$ and $\text{var}(\mathbf{U}) = \text{cov}(\mathbf{U},\mathbf{U})$

$$\text{cov}(\mathbf{AU},\mathbf{BV}) = \mathbf{A}\,\text{cov}(\mathbf{U},\mathbf{V})\mathbf{B}^T, \qquad \text{var}(\mathbf{a}^T\mathbf{U}) = \mathbf{a}^T\,\text{var}(\mathbf{U})\mathbf{a};$$

Gauss-Markov Theorem

How large is variance?

Compared with other estimators?

Theorem 1.2 [Gauss-Markov Theorem]

LSE $\widehat{\beta}$ is best linear unbiased estimator (BLUE):

- $\mathbf{a}^T\widehat{\beta}$ is a linear unbiased estimator of parameter $\theta = \mathbf{a}^T\beta$
- for any linear unbiased estimator $\mathbf{b}^T\mathbf{y}$ of θ,

$$\operatorname{var}(\mathbf{b}^T\mathbf{y}|\mathbf{X}) \geq \operatorname{var}(\mathbf{a}^T\widehat{\beta}|\mathbf{X})$$

Estimation of σ^2: $\quad \widehat{\sigma}^2 = \frac{\text{RSS}}{\mathbf{n-p}} = \frac{\|\mathbf{y}-\mathbf{X}\widehat{\beta}\|^2}{n-p}$

$\widehat{\sigma}^2$ is is an unbiased estimator of σ^2

Statistical inference

Additional assumption: $\varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_n)$

Under fixed design or conditioning on $\mathbf{X}$,

$$\widehat{\beta} = \beta + (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\varepsilon \quad \Longrightarrow \quad \widehat{\beta} \sim \mathcal{N}(\beta, (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2)$$

- ★ $\widehat{\beta}_j \sim \mathcal{N}(\beta_j, v_j\sigma^2)$ where v_j is jth diag of $(\mathbf{X}^T\mathbf{X})^{-1}$
- ★ $(n-p)\widehat{\sigma}^2 \sim \sigma^2\chi^2_{n-p}$ and $\widehat{\sigma}^2$ is indep. of $\widehat{\beta}$.
- ★ **$1-\alpha$ CI for β_j**: $\widehat{\beta}_j \pm t_{n-p}(1-\alpha/2)\sqrt{v_j}\widehat{\sigma}$ (homework)
- ★ $H_0 : \beta_j = 0$: test statistics $t_j = \frac{\widehat{\beta}_j}{\sqrt{v_j}\widehat{\sigma}} \sim_{H_0} t_{n-p}$.

Non-normal error

Appeal to asymptotic theory:

$$\sqrt{n}(\widehat{\beta} - \beta) = \underbrace{(n^{-1}\mathbf{X}^T\mathbf{X})^{-1}}_{n^{-1}\sum_{i=1}^n \mathbf{x}_i\mathbf{x}_i^T} \underbrace{n^{-1/2}\mathbf{X}^T\varepsilon}_{n^{-1/2}\sum_{i=1}^n \mathbf{x}_i\varepsilon_i}$$

LLN **CLT**

Using Slutsky's theorem, (homework)

$$\sqrt{n}(\widehat{\beta} - \beta) \xrightarrow{d} N(0, \mathbf{\Sigma}^{-1}) \qquad \text{or} \qquad \widehat{\beta} \xrightarrow{d} \mathcal{N}(\beta, (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2)$$

Holds approx. for large n

Correlated errors

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon, \quad \text{where} \quad \mathrm{var}(\varepsilon|\mathbf{X}) = \sigma^2 \mathbf{W}$$

Transform data: $\mathbf{y}^* = \mathbf{W}^{-1/2}\mathbf{y}$, $\mathbf{X}^* = \mathbf{W}^{-1/2}\mathbf{X}$, $\varepsilon^* = \mathbf{W}^{-1/2}\varepsilon$. Then

$$\mathbf{y}^* = \mathbf{X}^*\beta + \varepsilon^*, \quad \text{with} \quad \mathrm{var}(\varepsilon^*|\mathbf{X}) = \sigma^2 \mathbf{I}.$$

General Least-Squares:

$$\min_{\beta \in \mathbb{R}^p} \quad ||\mathbf{y}^* - \mathbf{X}^*\beta||^2 = (\mathbf{y} - \mathbf{X}\beta)^T \mathbf{W}^{-1} (\mathbf{y} - \mathbf{X}\beta)$$

Heteroscedastic errors: $\mathbf{W}_i = \sigma^2 \operatorname{diag}(v_1, \cdots, v_n)$

Weighted Least-squares: $\min_\beta \sum_{i=1}^n (y_i - \mathbf{X}_i^T \beta)^2 / v_i$.

1.2. Model Building

Nonlinear and nonparametric regression

Nonlinear regression

Univariate: $Y = f(X) + \varepsilon$,

■ $f(\cdot)$ has structural property: smooth, monotone, convex ...

Weierstrass theorem: any continuous $f(X)$ on $[0,1]$ can be uniformly approximated by a polynomial function.

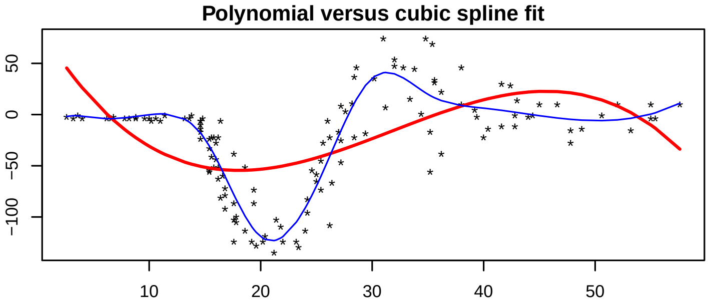
Polynomial regression:

$$Y = \overbrace{\beta_0 + \beta_1 X + \cdots + \beta_d X^d}^{\approx f(X)} + \varepsilon$$

★multiple regression with $X_1 = X, \cdots, X_d = X^d$

Drawback: not suitable for functions with **varying** degrees of smoothness

Polynomial versus cubic spline regressions



- ★**Red**: polynomial regression with degree 3
- **Blue**: spline regression with with degree 3

Spline regression

- ★ piecewise polynomials with degree d, with continuous derivatives up to order $d-1$.
- ★ **Knots**: $\{\tau_j\}_{j=1}^{K}$ where discontinuity occurs.

Example: Linear splines on $[0,1]$ with knots $\tau_1 < \tau_2$

- Linearity on $[0,\tau_1]$ yields $l(x) = \beta_0 + \beta_1 x,\ x \in [0,\tau_1]$.
- Linearity on $[\tau_1,\tau_2]+$ continuity at τ_1 gives

$$l(x) = \beta_0 + \beta_1 x + \beta_2 (x-\tau_1)_+,\ x \in [\tau_1,\tau_2]$$

- Linearity on $[\tau_1,\tau_2]+$ continuity at τ_1 gives $[\tau_2,1]$

$$l(x) = \beta_0 + \beta_1 x + \beta_2 (x-\tau_1)_+ + \beta_3 (x-\tau_2)_+,\ x \in [\tau_2,1] \text{ and}$$

Basis functions for Linear Splines

Basis functions for the linear splines:

$$B_0(x) = 1, B_1(x) = x, B_2(x) = (x - \tau_1)_+, B_3(x) = (x - \tau_2)_+$$

Spline regression:

$$Y = \underbrace{\beta_0 B_0(X) + \beta_1 B_1(X) + \beta_2 B_2(X) + \beta_3 B_3(X)}_{\approx f(X)} + \varepsilon$$

- Multiple regression with $X_0 = B_0(X), X_1 = B_1(X), X_2 = B_2(X), X_3 = B_3(X)$

General case: $\{1, x, (x - \tau_j)_+, \quad j = 1, \cdots, K\}$

Nonparametric: When K is large, $K_n \to \infty$

Cubic splines

Piecewise cubic polynomial with cont. 1^{st} and 2^{nd} derivatives:

$$c(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3, \quad x \le \tau_1,$$
$$c(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 (x-\tau_1)_+^3, \; x \in [\tau_1, \tau_2],$$
$$c(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 (x-\tau_1)_+^3 + \beta_5 (x-\tau_2)_+^3, \quad x \in [\tau_2, 1].$$

Basis functions:

$$B_0(x) = 1, B_1(x) = x, B_2(x) = x^2, B_3(x) = x^3$$
$$B_4(x) = (x-\tau_1)_+^3, \quad B_5(x) = (x-\tau_2)_+^3.$$

★widely used; ★multiple regression

Extension to multiple covariates

- Bivariate quadratic regression model:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1^2 + \beta_4 \underbrace{X_1 X_2}_{\text{interaction}} + \beta_5 X_2^2 + \varepsilon$$

- Multivariate quadratic regression:

$$Y = \sum_{j=1}^{p} \beta_j X_j + \sum_{\mathbf{j \le k}} \beta_{jk} X_j X_k + \varepsilon$$

- Multivariate quadratic regression with main effect and interactions

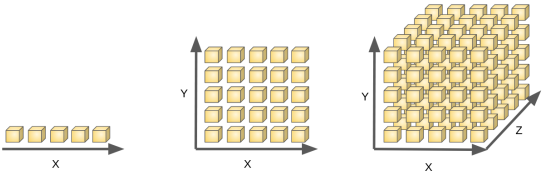
$$Y = \sum_{j=1}^{p} \beta_j X_j + \sum_{\mathbf{j < k}} \beta_{jk} X_j X_k + \varepsilon$$

Multivariate spline regression

Idea: Tensor products of univariate basis functions

$$\{B_{i_1}(x_1)B_{i_2}(x_2)\cdots B_{i_p}(x_p)\}_{i_1=1}^{b_1}\cdots{}_{i_p=1}^{b_p}$$

Drawbacks: **curse of dimensionality**, namely, number of basis functions scales exponentially with p



d	n^d	accuracy $n^{-\frac{2}{d+4}}$
1	**100**	**100**
2	10^4	250
5	10^{10}	4000
10	10^{20}	40000
100	10^{200}	$4 * 10^{41}$

Structured multivariate regressions

Remedy: Add additional structure to $f(\cdot)$

Example: Additive model

$$Y = f_1(X_1) + \cdots + f_p(X_p) + \varepsilon$$

- Number of basis functions scales **linearly** with p

Example: Bivariate interaction models:

$$Y = \sum_{1 \le i \le j \le p} f_{ij}(X_i, X_j) + \varepsilon$$

- Number of basis functions scales **quadratically** with p
- Implementation: Bivariate tensors

Best predictor and nonparametric regression

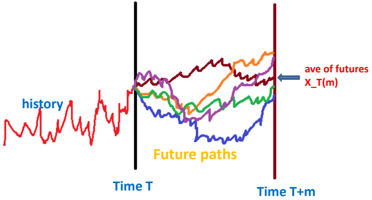
Double Expectation: $EZ = E\{E(Z|\mathbf{X})\}$, for any $\mathbf{X}$

Bias-var in prediction: Letting $f^*(\mathbf{X}) = E(Y|\mathbf{X})$, then

$$E(Y - f(\mathbf{X}))^2 = \underbrace{E(Y - f^*(\mathbf{X}))^2}_{\text{var}=E\sigma^2(\mathbf{X})} + E(\underbrace{f^*(\mathbf{X}) - f(\mathbf{X})}_{\text{bias}})^2.$$

Best prediction: $E(Y|\mathbf{X}) = \text{argmin}_f E(Y - f(\mathbf{X}))^2$

Nonparametric reg.: Estimating $f^*(\cdot)$ directly



Bias variance decomposition

Bias-var in estimation: letting $\bar{f}(\mathbf{x}) = E\widehat{f}_n(\mathbf{x})$, then

$$E(\widehat{f}_n(\mathbf{X}) - f(\mathbf{X}))^2 = \underbrace{E(\widehat{f}_n(\mathbf{X}) - \bar{f}(\mathbf{X}))^2}_{\textbf{var}} + E(\underbrace{\bar{f}(\mathbf{X}) - f(\mathbf{X})}_{\textbf{bias}})^2.$$

Role of Modeling:

★variance is small when n large, big when no. of parameters is big

★biases are small when model is complex (no. of parameters is big)

1.3. Ridge Regression

Ridge Regression

Drawbacks of OLS: ★$n > p$; ★large variance when collinearity

Remedy: Ridge regression (Hoerl and Kennard, 1970)

$$\widehat{\beta}_\lambda = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$$

★$\lambda > 0$ is a regularization parameter.

Interpretation: Penalized LS $\|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda\|\beta\|^2$.

— Setting the gradient to zero, we get $\mathbf{X}^T(\mathbf{X}\beta - \mathbf{y}) + \lambda\beta = \mathbf{0}$.

Bias-Variance Tradeoff

Smaller variances:

$$\mathrm{Var}(\widehat{\beta}_\lambda) = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\sigma^2 \prec \mathrm{Var}(\widehat{\beta}).$$

Larger biases:

$$E(\widehat{\beta}_\lambda) - \beta = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{X}\beta - \beta = -\lambda(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\beta.$$

Overall error: $\mathrm{MSE}(\widehat{\beta}_\lambda) =$

$$E\|\widehat{\beta}_\lambda - \beta\|^2 = \mathrm{tr}\{(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-2}[\lambda^2\beta\beta^T + \sigma^2\mathbf{X}^T\mathbf{X}]\}.$$

$\frac{d}{d\lambda}\mathrm{MSE}(\widehat{\beta}_\lambda)|_{\lambda=0} < 0 \;\Rightarrow \exists$ **a** $\lambda > 0$ **outperforms OLS.**

Generalization: ℓ_q Penalized Least Squares

ℓ_q penalized least-squares estimate:

$$\min_\beta = \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda\|\beta\|_q^q, \quad q \geq 0.$$

- λ tuning parameter, $\|\beta\|_q^q = |\beta_1|^q + \cdots + |\beta_p|^q$
- $q = 0$ is the best subset selection $\qquad \|\beta\|_0 = \#\{j : \beta_j \neq 0\}$
- Only $q = 2$ admits a closed-form solution.
- Known as Bridge estimator (Frank and Friedman, 1993);
- When $q = 1$, called Lasso estimator (Tibshirani, 1996);
- Folded concave when $0 < q < 1$ and convex when $q > 1$;

Prediction by similarity

Theorem 1.3. Alternative expression $\widehat{\beta}_\lambda = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \lambda \mathbf{I})^{-1}\mathbf{y}$

Prediction at $\mathbf{x}$ is $\widehat{y} = \mathbf{x}^T \widehat{\beta}_\lambda = \mathbf{x}^T \mathbf{X}^T (\mathbf{X}\mathbf{X}^T + \lambda \mathbf{I})^{-1}\mathbf{y}$.

Note that $(\mathbf{X}\mathbf{X}^T)_{ij} = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$ and $\mathbf{x}^T\mathbf{X}^T = (\langle \mathbf{x}, \mathbf{x}_1 \rangle, \cdots, \langle \mathbf{x}, \mathbf{x}_n \rangle)$.

- Prediction depends only **pairwise inner products**; **similarity**
- Generalize to other **similarity measures** $K(\cdot, \cdot)$, called **kernel** trick.

$K\Big($  ,  $\Big) = +10$ $\mathcal{K}\Big($  ,  $\Big) = -10$

Kernel regression

Kernel: $\mathbf{K} = (K(\mathbf{x}_i, \mathbf{x}_j))_{n \times n}$ is PSD, for any $\{\mathbf{x}_i\}_{i=1}^n$.

Commonly used kernels: $K(\mathbf{u}, \mathbf{v})$

★linear $\langle \mathbf{u}, \mathbf{v} \rangle$ ★polynomial $(1 + \langle \mathbf{u}, \mathbf{v} \rangle)^{\mathbf{d}}$, $d = 2, 3, \cdots$;

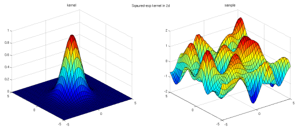
★Gaussian $e^{-\gamma \|\mathbf{u} - \mathbf{v}\|^2}$ ★Laplacian $e^{-\gamma \|\mathbf{u} - \mathbf{v}\|}$

Basis: $\{K(\cdot, \mathbf{x}_j)\}_{j=1}^n$ and express $f(\mathbf{x}) = \sum_{j=1}^n \alpha_j K(\mathbf{x}, \mathbf{x}_j)$

- Fit the model $y_i = f(\mathbf{x}_i) + \varepsilon_i$ by

$$\min_{\alpha \in \mathbb{R}^n} \left\{ \|\mathbf{y} - \mathbf{K}\alpha\|^2 + \lambda \alpha^T \mathbf{K} \alpha \right\}$$

- No curse-of-dim in implementation!



Kernel ridge regression

With $\mathbf{K} = (K(\mathbf{x}_i, \mathbf{x}_j)) \in \mathbb{R}^{n \times n}$, prediction at $\mathbf{x}$ is

$$\widehat{y} = (K(\mathbf{x}, \mathbf{x}_1), \cdots, K(\mathbf{x}, \mathbf{x}_n))(\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{y},$$

★ $\underset{\text{testing}}{\widehat{y}} = \overbrace{\widehat{f}(\mathbf{x})}^{\text{pred}} = \sum_{i=1}^{n} \overbrace{\alpha_i}^{\text{weight}} K(\underset{\text{testing}}{\mathbf{x}}, \underset{\text{training}}{\mathbf{x}_i}),$ $\quad \widehat{\alpha} = (\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{y};$

★ tune the parameter λ to minimize prediction errors.

1.4 Reproducing Kernel Hilbert Spaces

Justification of Kernel Tricks by Representer Theorem

Hilbert Space

Hilbert space: a space endowed with an inner product.

■ $\mathcal{X}$ = set, $\mathcal{H}$ = a space of functions on $\mathcal{X}$ with inner product $\langle \cdot, \cdot \rangle$.

Kernel function $K(\cdot, \cdot)$: Matrix $(K(\mathbf{x}_i, \mathbf{x}_j))_{n \times n}$ is PSD, for all $\{\mathbf{x}_i\}_{i=1}^n$,

Eigen-decomposition:

$$K(\mathbf{x}, \mathbf{x}') = \sum_{j=1}^{\infty} \gamma_j \psi_j(\mathbf{x}) \psi_j(\mathbf{x}'), \qquad \sum_{j=1}^{\infty} \gamma_j^2 < \infty$$

— $\{\gamma_j\}_{j=1}^{\infty}$ are **eigenvalues**, and $\{\psi_j\}_{j=1}^{\infty}$ are **eigen-functions**.

Reproducing Hilbert Space

Hilbert space: $\mathcal{H}_K = \{g = \sum_{j=1}^{\infty} \beta_j \psi_j\}$, endowed with inner product

$$\langle g, g' \rangle_{\mathcal{H}_K} = \sum_{j=1}^{\infty} \gamma_j^{-1} \beta_j \beta_j'; \qquad \|g\|_{\mathcal{H}_K} = \sqrt{\langle g, g \rangle_{\mathcal{H}_K}},$$

for any $g, g' \in \mathcal{H}_K$ with $g = \sum_{j=1}^{\infty} \beta_j \psi_j, g' = \sum_{j=1}^{\infty} \beta_j' \psi_j$.

Reproducibility: $\langle K(\cdot, x'), g \rangle_{\mathcal{H}_K} = \sum_j \gamma_j^{-1} \{\boldsymbol{\gamma_j \psi_j(x')}\} \boldsymbol{\beta_j} = g(\mathbf{x}')$.

Representer Theorem

Theorem 1.4

For a loss $L(y, f(\mathbf{x}))$ and increasing function $P_\lambda(\cdot)$, let

$$\widehat{f} = \text{argmin}_{f \in \mathcal{H}_K} \left\{ \sum_{i=1}^{n} L(y_i, f(\mathbf{x}_i)) + P_\lambda(\|f\|_{\mathcal{H}_K}) \right\}, \quad \lambda > 0,$$

Then (homework)

$$\widehat{f}(\cdot) = \sum_{j=1}^{n} \widehat{\alpha}_j K(\cdot, \mathbf{x}_j),$$

where $\widehat{\alpha} = (\widehat{\alpha}_1, \cdots, \widehat{\alpha}_n)^T$ solves

$$\min_{\alpha} \left\{ \sum_{i=1}^{n} L\Big(y_i, \sum_{j=1}^{n} \alpha_j K(\mathbf{x}_i, \mathbf{x}_j)\Big) + P_\lambda\left(\sqrt{\alpha^T \mathbf{K} \alpha}\right) \right\}.$$

★ **Infinite-dimensional** regression problem;

★ **Finite-dimensional** representation for the solution.

Outline of Proof

1. Any f can be written as $f = f_K + r$, where $f_K(\cdot) = \sum_{j=1}^n \alpha_j K(\cdot, \mathbf{x}_j)$ (projection) and r is in its orthogonal complement.

2. Orthogonality entails $0 = \langle K(\cdot, x_j), r \rangle_{\mathcal{H}_K} = r(x_j)$ by reproducibility. Hence, $f(x_i) = f_K(x_i)$ (the same loss).

3. But $\|f\|^2_{\mathcal{H}_K} = \|f\|^2_{\mathcal{H}_K} + \|r\|^2_{\mathcal{H}_K} \geq \|f\|^2_{\mathcal{H}_K}$.

4. Optimality reaches only if $r = 0$.

Applications of Representer Theorem

Apply representer theorem to **kernel ridge regression**

$$\widehat{f} = \mathrm{argmin}_{f \in \mathcal{H}_K} \left\{ \sum_{i=1}^{n} \big(y_i - f(\mathbf{x}_i)\big)^2 + \lambda \|f\|^2_{\mathcal{H}_K} \right\}.$$

We must have $\widehat{f} = \sum_{i=1}^{n} \widehat{\alpha}_i K(\cdot, \mathbf{x}_i)$ with $\widehat{\alpha} \in \mathbb{R}^n$ solving

$$\min_{\alpha \in \mathbb{R}^n} \left\{ \|\mathbf{y} - \mathbf{K}\alpha\|^2 + \lambda \alpha^T \mathbf{K} \alpha \right\}.$$

It is easily seen that

$$\widehat{\alpha} = (\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{y}.$$

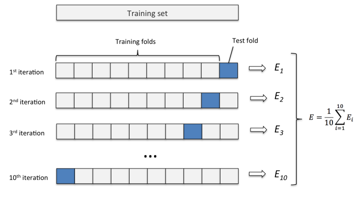
1.5 Cross-Validation

k-fold Cross-Validation

Purpose: To estimate **Prediction Error** for a procedure; to select tuning parameters and compare multiple methods

k-fold Cross-Validation (CV)

- ★ Divide data randomly and evenly into k subsets;
- ★ Use one fold as **testing set** and remaining as **training set** to compute testing errors;
- ★ Repeat for each of k subsets and average testing errors.



Choice of k: $k = n$ *(best, but expensive; leave-one out)*, 10 or 5 (5-fold).

Leave-one-out: $\mathrm{CV} = \frac{1}{n}\sum_{i=1}^{n}[y_i - \widehat{f}^{-i}(\mathbf{x}_i)]^2$, $\widehat{f}^{-i}(\mathbf{x}_i) =$ predicted value based on $\{(\mathbf{x}_j, y_j)\}_{j\neq i}$

Linear smoother

■ $\widehat{\mathbf{y}} = \mathbf{S}\mathbf{y}$ for data $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, **S** depends only on **X**.

Self-stable if $\bar{f}(\mathbf{x}) = \widehat{f}(\mathbf{x})$, where $\bar{f}$ is estimated function based on data $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ and $(\mathbf{x}, \widehat{f}(\mathbf{x}))$, and $\widehat{f}$ based on $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$

Theorem 1.5. For a self-stable linear smoother $\widehat{\mathbf{y}} = \mathbf{S}\mathbf{y}$,

$$y_i - \widehat{f}^{-i}(\mathbf{x}_i) = \frac{y_i - \widehat{y}_i}{1 - S_{ii}}, \quad \forall i \in [n], \qquad \mathrm{CV} = \frac{1}{n}\sum_{i=1}^n \left(\frac{y_i - \widehat{y}_i}{1 - S_{ii}}\right)^2.$$

Proof: By self-stability, $\{(\mathbf{x}_j, y_j), j \neq i\}$ and $\{(\mathbf{x}_j, y_j), j \neq i, (\mathbf{x}_i, \widehat{\mathbf{f}}^{(-i)}(\mathbf{x}_i))\}$ have the same fit: $\widehat{f}^{(-i)}(\mathbf{x}_i) = S_{ii}\widehat{f}^{(-i)}(x_i) + \sum_{j\neq i} S_{ij} y_j$

Generalized Cross-Validation

GCV (Golub et al., 1979): $\mathrm{GCV} = \frac{\frac{1}{n}\sum_{i=1}^n (y_i - \widehat{y}_i)^2}{[1 - \mathrm{tr}(\mathbf{S})/n]^2}$.

■ $\mathrm{tr}(\mathbf{S})$ is called **effective degrees of freedom**.

GCV chooses λ by minimizing

$$\mathrm{GCV}(\lambda) = \frac{\frac{1}{n}\mathbf{y}^T(\mathbf{I} - \mathbf{S}_\lambda)\mathbf{y}}{[1 - \mathrm{tr}(\mathbf{S}_\lambda)/n]^2}.$$

Self-stable Method	$\mathbf{S}$	$\mathrm{tr}(\mathbf{S})$
Multiple Linear Regression	$\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$	p
Ridge Regression	$\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T$	$\sum_{j=1}^p \frac{d_j^2}{d_j^2 + \lambda}$
Kernel Ridge Regression in RKHS	$\mathbf{K}(\mathbf{K} + \lambda\mathbf{I})^{-1}$	$\sum_{j=1}^n \frac{\gamma_j}{\gamma_j + \lambda}$

★ $\{d_j\}$ and $\{\gamma_j\}$ are singular values of $\mathbf{X}$ and $\mathbf{K}$.