Background

Linear Algebra

$\|\mathbf{x}\|_p = (\sum |x_i|^p)^{1/p}$ $\quad\|\mathbf{x}\|_\infty = \max |x_i|$

$\mathrm{tr}(\mathbf{Axx}^T) = \mathbf{x}^T\mathbf{Ax}$

$|\mathbf{AB}| = |\mathbf{A}||\mathbf{B}|$ $\quad|\mathbf{A}^m| = |\mathbf{A}|^m$

$(\mathbf{A}+\mathbf{UCV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{C}^{-1}+\mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1}$

$(\mathbf{A}+\mathbf{B})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}(\mathbf{A}+\mathbf{B})^{-1}\mathbf{A}^{-1}$

$\mathbf{U}(\mathbf{VU}+\mathbf{I})^{-1} = (\mathbf{UV}+\mathbf{I})^{-1}\mathbf{U}$

$\mathbf{I} - \mathbf{A}(\mathbf{I}+\mathbf{A})^{-1} = (\mathbf{I}+\mathbf{A})^{-1}$

Probability

$\mathrm{Ber}(x|\theta) = \theta^x(1-\theta)^{1-x}$ $\quad 0\le\theta\le1$

$\mathbb{P}[X|Y] = \frac{\mathbb{P}[X,Y]}{\mathbb{P}[Y]} = \frac{\mathbb{P}[Y|X]\mathbb{P}[X]}{\mathbb{P}[Y]}$

$p_Y(y) = p_X(g^{-1}(y)) \left|\det \frac{\partial g^{-1}(y)}{\partial y}\right|$

$\mathbb{E}[X] = \int_\Omega xp(x)\,dx = \int_\omega x\mathbb{P}[X{=}x]\,dx$

$\mathbb{E}_{Y|X}[Y] = \mathbb{E}_Y[Y|X]$ | $\mathbb{E}_Y[\mathbb{E}_X[X|Y]] = \mathbb{E}_X[X]$

$\mathbb{E}_{X,Y}[f(X,Y)] = \mathbb{E}_X\mathbb{E}_{Y|X}[f(X,Y)|X]$

$\mathbb{V}(X) = \mathbb{E}[(X-\mathbb{E}[X])^2] = \mathbb{E}[X^2]-\mathbb{E}[X]^2$

$\mathrm{Var}(X) = \mathbb{E}[\mathrm{Var}(X\mid Y)] + \mathrm{Var}(\mathbb{E}[X\mid Y])$

$\mathbb{V}[X+Y] = \mathrm{Var}[X]+\mathrm{Var}[Y]+2\mathrm{Cov}(X,Y)$

$\mathrm{Cov}(\mathbf{X},\mathbf{Y}) = \mathbb{E}[\mathbf{XY}^T] - \mathbb{E}[\mathbf{X}]\mathbb{E}[\mathbf{Y}]^T$

$\mathrm{Cov}(A\mathbf{X}+c, B\mathbf{Y}+d) = A\mathrm{Cov}(\mathbf{X},\mathbf{Y})B^T$

$\mathcal{N}(x|\mu,\Sigma) = \frac{exp(-\frac12(x-\mu)^T\Sigma^{-1}(x-\mu))}{(2\pi)^{D/2}|\Sigma|^{1/2}}$

$X = \Sigma^{1/2}\mathcal{N}(0,1)+\mu \sim \mathcal{N}(\mu,\Sigma)$

$Y = MX+b \sim \mathcal{N}(M\mu+b, M\Sigma M^T)$

$X,Y \overset{\text{iid}}{\sim} \mathcal{N}:\quad X+Y\sim\mathcal{N}(\mu+\mu', \Sigma+\Sigma')$

Conditional Gaussian

$P(\begin{bmatrix}\mathbf{X}\\\mathbf{Y}\end{bmatrix}) = \mathcal{N}(\begin{bmatrix}\mathbf{X}\\\mathbf{Y}\end{bmatrix}; \begin{bmatrix}\mu_1\\\mu_2\end{bmatrix}, \begin{bmatrix}\Sigma_{11}&\Sigma_{12}\\\Sigma_{21}&\Sigma_{22}\end{bmatrix})$

$p(\mathbf{Y}|\mathbf{X}=\mathbf{x}) = \mathcal{N}(\mu,\Sigma)$

$\cdot\ \mu = \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(\mathbf{x}-\mu_{\mathbf{X}})$

$\cdot\ \Sigma = \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$

Inequalities and Estimators

Jensen: $log(\sum_i \lambda_i^{(\ge0)} x_i) \ge \sum_i \lambda_i log(x_i)$

Chebyshev: $\mathbb{P}[|\hat X - X| \ge \varepsilon] \le \frac{MSE[\hat X]}{\varepsilon^2}$

Estimators: *Unbiased:* $\mathbb{E}[\hat\theta] = \theta^\star$

Consistent: $\mathbb{P}(|\hat\theta-\theta^\star|<\varepsilon)\to 0$ convP

Asymp Normal: $(\hat\theta-\theta^\star)\hat{se}^{-1}\sim\mathcal{N}(0,1)$

Rao-Cra.: $\mathbb{E}_{x|\theta}[(\theta-\hat\theta)^2] \ge \frac{(\frac{\partial}{\partial\theta}b_{\hat\theta}+1)^2}{\mathbb{E}_{x|\theta}[\Lambda^2]} + b_{\hat\theta}^2$

$b_{\hat\theta} = \mathbb{E}_{x|\theta}[\hat\theta]-\theta$ $\quad\Lambda = \frac{\partial}{\partial\theta}\log p(x|\theta)$

$\mathbb{E}_{x|\theta}[\Lambda] = 0 \to \mathbb{E}_{x|\theta}[\Lambda\hat\theta] = \frac{\partial}{\partial\theta}b_{\hat\theta}+1$

$\to \mathrm{Cov}(\Lambda,\hat\theta)\to$ Cauchy

$\mathrm{Var}[\hat\theta] \ge \mathcal{I}_n(\theta)^{-1} = -\mathbb{E}[\frac{\partial^2 \log p[\mathcal{X}_n|\theta]}{\partial\theta^2}]^{-1}$

Efficiency of $\hat\theta$: $e(\theta_n) = \frac{1}{\mathrm{Var}[\hat\theta_n]\mathcal{I}_n(\theta)}$

$\hat\theta_{JS} = \left(1-\frac{(d-2)\sigma^2}{\|y\|^2}\right)y$

Derivatives $\frac{\partial}{\partial\mathbf{x}}(\mathbf{b}^\top\mathbf{x}) = \frac{\partial}{\partial\mathbf{x}}(\mathbf{x}^\top\mathbf{b}) = \mathbf{b}$

$\frac{\partial}{\partial\mathbf{x}}(\mathbf{x}^\top\mathbf{Ax}) = (\mathbf{A}^\top+\mathbf{A})\mathbf{x}$

$\frac{\partial}{\partial\mathbf{X}}(\mathbf{c}^\top\mathbf{Xb}) = \mathbf{cb}^\top$ $\quad\frac{\partial}{\partial\mathbf{X}}(\|\mathbf{X}\|_F^2) = 2\mathbf{X}$

$\frac{\partial}{\partial\mathbf{x}}\|\mathbf{x}\|_2 = \frac{\mathbf{x}}{\|\mathbf{x}\|_2}$ | $\frac{\partial}{\partial\mathbf{x}}\|f(\mathbf{x})\|_1 = \frac{\partial f(\mathbf{x})}{\partial\mathbf{x}}^T\mathrm{sgn}(\mathbf{x})$

$\frac{\partial}{\partial\mathbf{x}}(\|\mathbf{Ax}-\mathbf{b}\|_2^2) = 2\mathbf{A}^\top(\mathbf{Ax}-\mathbf{b})$

$\frac{\partial}{\partial\mathbf{X}}(|\mathbf{X}|) = |\mathbf{X}|\cdot\mathbf{X}^{-1}$, $\quad|\mathbf{X}|^{-1} = |\mathbf{X}^{-1}|$

$\frac{\partial}{\partial\mathbf{X}}f(\mathbf{X})^\top = \frac{\partial f(\mathbf{X})}{\partial\mathbf{X}}^T$ | $\frac{\partial}{\partial\mathbf{X}}\mathrm{tr}\,f(\mathbf{X}) = \mathrm{tr}\,\frac{\partial f(\mathbf{X})}{\partial\mathbf{X}}$

$\frac{\partial}{\partial\mathbf{X}}\det f(\mathbf{X}) = \det f(\mathbf{X})\,\mathrm{tr}(f(\mathbf{X})^{-1}\frac{\partial f(\mathbf{X})}{\partial\mathbf{X}})$

$\frac{\partial}{\partial\mathbf{X}}f(\mathbf{X})^{-1} = -f(\mathbf{X})^{-1}\frac{\partial f(\mathbf{X})}{\partial\mathbf{X}}f(\mathbf{X})^{-1}$

Quadratic Forms

$\mathbf{x}^TA\mathbf{x}+2\mathbf{b}^T\mathbf{x}+c = (\mathbf{x}+A^{-1}\mathbf{b})^TA$

$(\mathbf{x}+A^{-1}\mathbf{b}) - \mathbf{b}^TA^{-1}\mathbf{b}+c,$

$ax^2+bx+c = (x+\frac{b}{2a})^2 - (\frac{b}{2a})^2 + c$

Information Theory

$\mathrm{H}[p] = \mathbb{E}_{\mathbf{x}\sim p}[-\log p(\mathbf{x})]$

$\mathrm{H}[p\|q] = \mathbb{E}_{\mathbf{x}\sim p}[-\log q(\mathbf{x})]$

$\mathrm{KL}[p\|q] = \mathrm{H}[p\|q]-\mathrm{H}[p]$

$\mathrm{KL}[p\|q] = \mathbb{E}_{\theta\sim p}\left[\log\left(\frac{p(\theta)}{q(\theta)}\right)\right]$

$\mathrm{KL}[p\|q]\ne\mathrm{KL}[q\|p]\ge0$

$\mathrm{H}[\mathbf{X}] = \mathbb{E}_{\mathbf{X}\sim p}[-\log p(\mathbf{X})]$

$\mathrm{H}[\mathbf{X}|\mathbf{Y}=y] = \mathbb{E}_{\mathbf{X}\sim p(\cdot|y)}[-\log p(\mathbf{X}|y)]$

$\mathrm{H}[\mathbf{X}|\mathbf{Y}] = \mathbb{E}_y[\mathrm{H}[\mathbf{X}|\mathbf{Y}=y]]$

$\mathrm{H}[\mathbf{X}|\mathbf{Y}] = \mathrm{H}[\mathbf{Y}|\mathbf{X}]+\mathrm{H}[\mathbf{X}]-\mathrm{H}[\mathbf{Y}]$

$\mathrm{H}[\mathbf{X},\mathbf{Y}] = \mathbb{E}_{(\mathbf{X},\mathbf{Y})\sim p(\cdot,\cdot)}[-\log p(\mathbf{X},\mathbf{Y})]$

$\mathrm{I}[\mathbf{X};\mathbf{Y}] = \mathrm{H}[\mathbf{X}]-\mathrm{H}[\mathbf{X}|\mathbf{Y}]\ge0$

$\mathrm{I}[\mathbf{X};\mathbf{Y}|\mathbf{Z}] = \mathrm{I}[\mathbf{X};\mathbf{Y},\mathbf{Z}]-\mathrm{I}[\mathbf{X};\mathbf{Z}]$

$\mathrm{H}(\mathcal{N}(\mu,\Sigma)) = \frac12\ln(\det(2\pi e\Sigma))$

$\mathrm{KL}(\mathcal{N}(a,A)\|\mathcal{N}(b,B)) = \frac12(\mathrm{tr}(B^{-1}A) + (a-b)^TB^{-1}(a-b) - d + \ln(\frac{\det B}{\det A}))$

Risks

Expected Risk: $R(f) = P(f(X)\ne y)$

$\mathcal{R}(f) = \sum_{y\le k}P(y)\mathbb{E}_{P(x|y)}[1_{f(x)\ne y}|Y=y]$

Empirical Risk Minimizer (ERM) $\hat f$:

$\hat f\in\arg\min_{f\in\mathcal{H}}\hat R(\hat f, \mathcal{D}^{train})$

$\hat R(\hat f,\mathcal{D}^{train}) = \frac1n\sum_{i=1}^n\mathcal{L}(Y_i,\hat f(X_i))$

$\hat R(\hat f,\mathcal{D}^{test}) = \frac1m\sum_{i=n+1}^{n+m}\mathcal{L}(Y_i,\hat f(X_i))$

Loss Fcts: $\mathcal{L}(y,z)$ $\quad z=w^\top x$

$\mathcal{L}^{0/i} = \mathbb{I}[\mathrm{sign}(z)\ne y]$

$\mathcal{L}^{\text{hinge}} = \max(0,1-yz)$ for SVM's

$\mathcal{L}^{\text{percep}} = \max(0,-yz)$

$\mathcal{L}^{\text{logistic}} = \log(1+\exp(-yz))$

$\mathcal{L}^{\exp} = \exp(-yz)$ for AdaBoost

$\mathcal{L}^{\text{CE}} = -[y'\log z' + (1-y')\log(1-z')]$

$y' = \frac{1+y}{2}$, $\quad z' = \frac{1+z}{2}$

Optimization *GDM:*

$\theta^{(t+1)}\leftarrow\theta^{(t)}-\eta\nabla_\theta\mathcal{L}+\mu(\theta^{(t)}-\theta^{(t-1)})$

GD: $\theta^{(t+1)}\leftarrow\theta^{(t)}-\eta\nabla_\theta\mathcal{L}$

SGD: $\theta^{(t+1)}\leftarrow\theta^{(t)}-\eta\nabla\mathcal{L}(\theta^{(t)},x_i,y_i)$

NGD: $\theta^{(t+1)}\leftarrow\theta^{(t)}-\eta(\nabla_\theta^2\mathcal{L})^{-1}\nabla_\theta\mathcal{L}$

$\to f(x+t)\approx f(x)+tf'(x)+\frac12f''(x)t^2=0$

Parametric Density Estimation

Assume prior $\mathbb{P}(\theta)$,

Likelihood: $\mathbb{P}[\mathcal{X}|\theta] = \prod_{i\le n}p(x_i|\theta)$

$\hat\theta_{MLE} = \arg\max_\theta\mathbb{P}[\mathcal{X}|\theta]$

$\hat\theta_{MAP} = \arg\max_\theta[P(\theta|\mathcal{X}) = P(\mathcal{X}|\theta)P(\theta)]$

Solve $\nabla_\theta logP(\mathcal{X}|\theta)P(\theta) = 0$

1-D Gaussian Bayesian learning

$X|\theta\sim\mathcal{N}(\theta,\sigma^2)$ $\quad\theta\sim\mathcal{N}(m_0,s_0^2)$

$\theta|X\sim\mathcal{N}(\mu_n,\sigma_n^2)$

$\sigma_n^2 = \frac{\sigma^2s_0^2}{ns_0^2+\sigma^2}$, $\quad\mu_n = \frac{ns_0^2\bar x+m_0\sigma^2}{ns_0^2+\sigma^2}$

Recursive Bayesian density learning

$\mathcal{X}^n = x_{1:n}: p(\theta|\mathcal{X}^n) = \frac{p(x_n|\theta)p(\theta|\mathcal{X}^{n-1})}{\int p(x_n|\theta)p(\theta|\mathcal{X}^{n-1})d\theta}$

Frequentist vs Bayesian

Bayes: priors, distributions, needs efficient integration, adds regularization term.

Frequentist: no priors, point estimate, requires only differentiation methods. MLE are consistent, equivariant, asymptotically normal, asymptotically efficient (no efficient for finite samples).

Data Types

monadic: $X:O\to\mathbb{R}^d$ dyadic: $X:O_1\times O_2\to\mathbb{R}^d$. pairwise: $X:O_1\times O_1\to\mathbb{R}^d$ polyadic data: $X:O_1\times O_2\times O_3\to\mathbb{R}^d$ nominal = qualitative (sweet, sour ...), ordinal = absolute order, quantitative = numbers

Regression

Model of data: $\mathbf{Y} = \mathbf{X}\beta^\star+\varepsilon$

$\mathbf{X}\in\mathbb{R}^{(d+1)\times n}$ $\quad\beta\in\mathbb{R}^{d+1}$ $\quad\varepsilon\sim\mathcal{N}(\mathbf{0},\mathbb{I}\sigma^2)$

$\mathbf{Y}|\mathbf{X},\beta,\sigma^2\sim\mathcal{N}(\mathbf{Y};\mathbf{X}^T\beta,\mathbb{I}_{(d+1)}\sigma^2)$

MLE: Ordinary Least Squares

OLSE is unbiased, orthogonal projection with lowest variance. differentiate wrt β.

$\mathcal{L} = \mathrm{RSS}(\beta) = \sum_{i=1}^n(y_i-x_i^T\beta)^2 = (\mathbf{y}-\mathbf{X}\beta)^2$

Estimator: $\hat\beta^{\text{OLS}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

Prediction: $\hat y = \mathbf{X}\hat\beta = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

MAP: Ridge Regression (L^2 penalty)

Penalize energy in β. *Prior:* $\beta\sim\mathcal{N}(0,\lambda^{-1}\mathbb{I})$

Loss: $\mathcal{L} = (\mathbf{y}-\mathbf{X}\beta)^T(\mathbf{y}-\mathbf{X}\beta)+\lambda\beta^T\beta$

Estimator: $\hat\beta^{\text{ridge}} = (\mathbf{X}^T\mathbf{X}+\lambda\mathbb{I})^{-1}\mathbf{X}^T\mathbf{y}$

MAP: Lasso (L^1 penalty)

Penalize full β. Lasso has no closed form.

$\beta\sim Lapl(0,\lambda^{-1}) = \frac\lambda2 exp(-\lambda|\beta|)$

$\mathcal{L} = \sum_{i=1}^n(y_i-x_i^T\beta)^2+\lambda\sum_{j=1}^d|\beta_j|$

$= (\mathbf{y}-\mathbf{X}\beta)^T(\mathbf{y}-\mathbf{X}\beta)+\lambda\|\beta\|_1$

Bayesian view: $Y|(X,\beta)\sim\mathcal{N}(x^T\beta,\sigma^2I)$

d-Dim Bayesian Linear Regression

Prior: $\beta\sim\mathcal{N}(\mu_0,\Lambda^{-1})$

Likelihood: $Y|\beta,X,\sigma\sim\mathcal{N}(X\beta,\sigma_n^2\mathbb{I})$

Posterior: $\beta|\mathbf{X},\mathbf{y}\sim\mathcal{N}(\mu,\Sigma)$

$\cdot\ \Sigma = (\sigma_n^{-2}\mathbf{X}^T\mathbf{X}+\Lambda)^{-1}$

$\cdot\ \mu = \Sigma(\Lambda\mu_0+\sigma_n^{-2}\mathbf{X}^T\mathbf{y})$

Nonlinear Regression

Idea: Add feature space transformation, kernel to compute inner product. Suppose:

$\beta\sim\mathcal{N}(\mathbf{0},\Lambda^{-1})$ $\quad\varepsilon\sim\mathcal{N}(\mathbf{0},\sigma_n^2\mathbb{I}_d)$

$\mathbf{Y} = \mathbf{X}\beta+\varepsilon\sim\mathcal{N}(\mathbf{0},\mathbf{X}\Lambda^{-1}\mathbf{X}^T+\sigma_n^2\mathbb{I}_d)$

Kernels

Kernel: $k(x_i,x_j) = \phi(x_i)\Lambda^{-1}\phi(x_j)^T$

Similarity based reasoning.

Gram Matrix: $K = k(\mathbf{x}_i,\mathbf{x}_j)$, $\quad 1\le i, j\le n$

$\cdot k(\mathbf{x},\mathbf{x}') = k(\mathbf{x}',\mathbf{x})$ $\cdot\ k(\mathbf{x},\mathbf{x}')$ pos.semi-def.

If k_1,k_2 kernels, $c\in\mathbb{R}_{>0}$, $\mathbf{A}^{psd}$, $p_{\text{pos-coeff}}$:

$k(\mathbf{x},\mathbf{x}') = k_1(\mathbf{x},\mathbf{x}')+k_2(\mathbf{x},\mathbf{x}') = \mathbf{x}^T\mathbf{Ax}'$

$= k_1(\mathbf{x},\mathbf{x}')\cdot k_2(\mathbf{x},\mathbf{x}') = c\cdot k_1(\mathbf{x},\mathbf{x}')$

$= p(k_1(\mathbf{x},\mathbf{x}')) = f(\mathbf{x})k_1(\mathbf{x},\mathbf{x}')f(\mathbf{x}')$

$k(\mathbf{x},\mathbf{x}') = \phi(x)^T\phi(x') = (1+\mathbf{x}^T\mathbf{x}')^m$

$= \tanh(\alpha\mathbf{x}^T\mathbf{x}'+c)$

$= \sigma^2\exp(-\frac{2\sin(p^{-1}\pi\|\mathbf{x}-\mathbf{x}'\|_2^2)}{l^2})$

$= \exp(-\|\mathbf{x}-\mathbf{x}'\|_1\,l^{-1})$

$= \exp(-\|\mathbf{x}-\mathbf{x}'\|_2^2(2l^2)^{-1})$

RBF: $\phi_j(x) = \exp(-\frac{\|x\|_2^2}{2})\prod_{i=0}^d x^{j_i}(j_i!)^{-\frac12}$

$\uparrow$ Lengthscale, smoother fcts.

Gaussian Process Regression

Applying a kernel, we get:

$\mathbf{Y} = \Phi\beta+\varepsilon\sim\mathcal{N}(\mathbf{0},\Phi\Lambda^{-1}\Phi^T+\sigma_n^2\mathbb{I}_d) =$

$\mathcal{N}(\begin{bmatrix}\mathbf{y}\\y_*\end{bmatrix}|\mathbf{0},\begin{bmatrix}\mathbf{K}+\sigma^2\mathbb{I} & \mathbf{k}\\ \mathbf{k^T} & k(x_*,x_*)+\sigma^2\end{bmatrix})$

Gaussian Process Prediction

Given $\mathcal{GP}(\mu,K)$,

$p(y_*|x_*,\mathbf{X},\mathbf{y}) = \mathcal{N}(\tilde\mu,\tilde\sigma^2)$,

$\cdot\ \tilde\mu = \mu(x_*)+\mathbf{k}^T(\mathbf{K}+\sigma_n^2\mathbb{I})^{-1}(\mathbf{y}-\mu(\mathbf{X}))$,

$\cdot\ \tilde\sigma^2 = k(x_*,x_*)-\mathbf{k}^T(\mathbf{K}+\sigma^2\mathbb{I})^{-1}\mathbf{k}$

$\cdot\ \mathbf{k} = k(x_*,\mathbf{X})$ $\quad\mathbf{K}_{ij} = k(x_i,x_j)$

$\cdot\ \tilde\sigma_{ij}^2 = k(x_i,x_j)-\mathbf{k}_i^T(\mathbf{K}+\sigma^2\mathbb{I})^{-1}\mathbf{k}_j$

Causality

Regression models capture correlation (not causality). ie, non-causal features can mislead models (*Spurious Correlations*). Iff Train Test Distribution change (*Domain Shift*). *Counterfactual Invariance:* A function f is invariant if $f(X(w)) = f(X(w'))$ for any w,w', reducing bias from spurious correlations. **Confounding:** A hidden variable influences both W and X, creating a spurious correlation with Y. **Selection Bias:** A hidden variable S filters the training data based on W and X, inducing non-causal associations.

If f is a counterfactually invariant predictor: In the **anti-causal scenario**: $f(X)\perp W|Y$. In the **causal scenario** (no selection but confounded): $f(X)\perp W$. In the **causal scenario** (no confounding but selected): $Y\perp X\mid W, X_{\perp W}$ and $f(X)\perp W\mid Y$.

A set of variables Z **d-separates** X and Y in a DAG $\mathcal{G}$ if all paths between X and Y are blocked by Z: $X\perp Y|Z$. A path is blocked if: **Collider:** $A\to B\leftarrow C$ and neither B nor its descendants are in Z. **Chain:** $A\to B\to C$. **Fork:** $A\leftarrow B\to C$ where $B\in Z$.

Algos

K-Means $J = \sum_{x\in\mathcal{X}}\|x-\mu_{c(x)}\|^2$

PCA proj. maximum variance subspace. top d eigenv. of $S = \frac1n\sum_{i=1}^n(x_i-\bar X)(x_i-\bar X)^T$

EM fit GMMs ($\sum_{k=1}^K\pi_k\mathcal{N}(x|\mu_k,\Sigma_k)$) by max. likelihood. Reaches local optimum.

Latent variable: $M_{xc} = 1_{c\text{ generated }x}$

$P(\mathcal{X},M|\theta) = \prod_x\prod_{c=1}^k(\pi_cP(x|\theta_c))^{M_{xc}}$

$\gamma_{xc} = \mathbb{E}[M_{xc}|\mathcal{X},\theta^{(j)}] = \frac{P(x|c,\theta^{(j)})P(c|\theta^{(j)})}{P(x|\theta^{(j)})}$

$\mu_c^{(j+1)} = \frac{\sum_{c\in\mathcal{X}}\gamma_{xc}x}{\sum_{c\in\mathcal{X}}\gamma_{xc}}$ $\quad\pi_c^{(j+1)} = \frac{1}{|\mathcal{X}|}\sum_{c\in\mathcal{X}}\gamma_{xc}$

$(\sigma_c^2)^{(j+1)} = \frac{\sum_{c\in\mathcal{X}}\gamma_{xc}(x-\mu_c)^2}{\sum_{c\in\mathcal{X}}\gamma_{xc}}$

Bias-Variance tradeoff

Bias$(\hat f) = \mathbb{E}[\hat f]-f$

Var$(\hat f) = \mathbb{E}[(\hat f-\mathbb{E}[\hat f])^2]$

Squared Error Decomposition

$\mathbb{E}_D\mathbb{E}_{X,Y}[(\hat f(X)-Y)^2] =$

$\mathbb{E}_{X,Y}[(\mathbb{E}_{Y|X}[Y]-Y)^2]$ (noise var)

$+\mathbb{E}_X\mathbb{E}_D[(\hat f_D(X)-\mathbb{E}_D[\hat f(X)])^2]$ (var.)

$+\mathbb{E}_X[(\mathbb{E}_D[\hat f_D(X)]-\mathbb{E}_{Y|X}[Y])^2]$ (bias^2)

With $\mathbb{E}_{Y|X}[Y]$ the expected label and $\mathbb{E}_D[\hat f(X)]$ the expected classifier.

Statistical Learning and Validation

Find $f: X\to Y$ to minimize expected risk by approximation with empirical risk.

K-Fold Cross Validation

Partition data Z into K equally sized, disjoint subsets:
$\mathcal{Z} = \mathcal{Z}_1 \bigcup \mathcal{Z}_2 \bigcup \cdots \bigcup \mathcal{Z}_K$, $\mathcal{Z}_\mu \bigcap \mathcal{Z}_\nu = \emptyset$
$|\mathcal{Z}_k| \approx n\frac{K-1}{K}$ # of training samples. Learn:
$\hat{f}^{-\nu}(x) = \arg\min_{f\in\mathcal{F}} \frac{\sum_{i\notin\mathcal{Z}_\nu} \mathcal{L}(y_i, f(x_i))}{|\mathcal{Z}-\mathcal{Z}_\nu|}$
$\hat{R}^{cv}(\mathcal{A}) = \frac{1}{n}\sum_{i\leq n} \mathcal{L}(y_i, \hat{f}^{-\kappa(i)}(x_i))$
Underfits because smaller dataset.
Leave-one-out: $K = n$ (unbiased but var can be large from correlated datasets)

Bootstrapping

Bootstrap samples: $\mathcal{Z}^* = \{\mathcal{Z}_1^*, \cdots \mathcal{Z}_B^*\}$, of same size as original, drawn with replacement. The chance of a sample to have appeared in the bootstrap is:
$1-(1-\frac{1}{n})^n \overset{n\to\infty}{\to} 1-\frac{1}{e} \approx 0.632$. So if we compute the ERM on $\mathcal{Z}$ we could get 63% accuracy by memorization. Over-confident (shows too small bias)!
Leave-one-out/out of bucket error: compensates by computing the ERM where no memorization was for specific sample. E.g., for classification, like cross-validation:
$\hat{\mathcal{R}}(\mathcal{A}) = \frac{1}{B}\sum_{b=1}^B \sum_{z_i\notin\mathcal{Z}^{*b}} \frac{\mathbb{I}_{c(x_i)\neq y_i}}{B-|\mathcal{Z}^{*b}|}$ $\hat{R}_{0.632} = 0.368\hat{R}(A(Z)) + 0.632\hat{R}_{bs}$
Wald Test: $W = \frac{\hat{\theta}-\theta_0}{\text{s.e.}(\hat{\theta})}$

Bayesian Neural Networks (BNN)

NN: no uncertainty quantification, overconfident, adversarial examples, poor generalization for domain shifts.
BNN: Using $p(w)$ and $p(D|w)$, approx. poster. by variational infer. (min rev KL).
$\sigma \leftarrow \sigma - \alpha_t \left(\varepsilon^\top \frac{\partial}{\partial w} F(w,\theta) + \frac{\partial}{\partial\sigma} F(w,\theta)\right)$

Information-based Transductive Lear.

ITL selects x_n that maximizes mutual information of $y_x = f_x + \varepsilon_x$ about f:
$x_n = \arg\max_{x\in S} I(f_A; y_x|D_{n-1})$
If $f \sim \mathrm{GP}(\mu, k)$, then:
$I(f_A; y_x|D_{n-1}) = \frac{1}{2}\log\left(\frac{\mathrm{Var}[y_x|D_{n-1}]}{\mathrm{Var}[y_x|f_A, D_{n-1}]}\right)$

Safe Bayesian Optimization

$x_n = \arg\max_{x\in\hat{S}_n=\{x|u_n^g(x)\geq 0\}} u_n^f(x)$

Batch Active Learning | ProbCover

$G=(X,E)$, $E=\{(x,x') \mid \|x-x'\|\leq\delta\}$
$L\leftarrow\emptyset$ $\forall i = 1, 2, \ldots, b$ $\hat{x} \leftarrow \arg\max_{x\in X} |\{x' \mid (x,x')\in E, x'\in X\}|$
$L \leftarrow L\cup\{\hat{x}\}$ | $E \leftarrow E\setminus(\{\hat{x}\}\times(B_\delta(\hat{x})\cap X))$

Classification

Definitions and Lemmas from Chapter 7

Definition 11 (Constrained Optimization Problem) A constrained optimization problem is of the form:

$$\min_{w\in\mathbb{R}^d} f(w) \quad \text{s.t.} \quad g_i(w) = 0,\ i\leq m, \quad h_j(w) \leq 0,\ j\leq n.$$

A feasible solution satisfies all constraints, and an optimal solution minimizes $f(w)$ over all feasible solutions8203;:contentReference[oaicite:0]index=0.

Definition 12 (Convex Optimization Problem) A constrained optimization problem is convex if $f, g_1, \ldots, g_m, h_1, \ldots, h_n$ are convex and the feasible region is convex8203;:contentReference[oaicite:1]index=1.

Definition 13 (Lagrangian) The Lagrangian of a constrained optimization problem is:

$$L(\lambda,\alpha,w) = f(w) + \sum_{i\leq m}\lambda_i g_i(w) + \sum_{j\leq n}\alpha_j h_j(w),$$

where λ and α are the Lagrange multipliers8203;:contentReference[oaicite:2]index=2.

Lemma 14 (Necessary Conditions for Optimal Solutions) Any optimal solution w^* must satisfy:

$$\nabla_w L(\lambda,\alpha,w) = 0, \quad g_i(w) = 0, \quad h_j(w) \leq 0, \quad \alpha_j \geq 0,$$

for $i\leq m, j\leq n$8203;:contentReference[oaicite:3]index=3.

Lemma 15 (Dual Function and Lower Bound) Given the Lagrangian $L(\lambda,\alpha,w)$, the dual function is:

$$\theta(\lambda,\alpha) = \inf_w L(\lambda,\alpha,w),$$

which satisfies:

$$\max_{\lambda,\alpha\geq 0}\inf_w L(\lambda,\alpha,w) \leq f(w^*).$$

This forms the basis of the dual problem8203;:contentReference[oaicite:4]index=4.

Definition 17 (Strong Duality) A convex optimization problem satisfies strong duality if:

$$\theta(\lambda^*,\alpha^*) = f(w^*),$$

where w^* solves the primal and (λ^*,α^*) solves the dual8203;:contentReference[oaicite:5]index=5.

Definition 18 (Slater's Condition) A convex optimization problem satisfies Slater's condition if there exists a strictly feasible point w_0 such that:

$$h_j(w_0) < 0, \quad \forall j\leq n.$$

This ensures strong duality8203;:contentReference[oaicite:6]index=6.

Lemma 19 (Slater's Condition and Strong Duality) If a convex optimization problem satisfies Slater's condition, then strong duality holds8203;:contentReference[oaicite:7]index=7.

Lemma 20 (Complementary Slackness) If strong duality holds, then for any optimal solution w^*:

$$f(w^*) = L(\lambda^*,\alpha^*,w^*),$$

$$\alpha_j h_j(w^*) = 0, \quad \forall j\leq n.$$

which enforces that inactive constraints have zero dual multipliers8203;:contentReference[oaicite:8]index=8.

Bayes Optimal Classifier

Minimizes total risk for 0-1 Loss
$\hat{c}(x) = \begin{cases} y & \mathbb{P}[y|x] > 1-d, \exists y \\ \mathcal{D} & \mathbb{P}[y|x] < 1-d, \forall y \end{cases}$

Linear Classifier

$g(x) = a^T\tilde{x}$ $\quad a = (w_0, w)^T, \tilde{x} = (1,x)^T$
$a^T\tilde{x}_i > 0 \Rightarrow y_i = 1, a^T\tilde{x}_i < 0 \Rightarrow y_i = 2$
Normalization: $\tilde{x}_i \to -\tilde{x}_i$ if $y_i = 2$
Find a: $a^T\tilde{x}_i > 0, \forall_i$!

Perceptron Criterion

$J_P(a) = \sum_{\tilde{x}\in\mathcal{X}^{\text{msc}}}(-a^T\tilde{x})$,
$\Rightarrow a_{k+1} = a_k + \eta_k \sum_{\tilde{x}\in\mathcal{X}^{\text{miscl}}} class * \tilde{x}$
Converges if data separable.

WINNOW Algorithm

Performs better when many dimensions are irrelevant. Search for 2 weight vectors a^+, a^- (for each class). If a point is misclassified: $a_i^+ \leftarrow \alpha^{+\tilde{x}_i} a_i^+, a_i^- \leftarrow \alpha^{-\tilde{x}_i} a_i^-$
(class 1 err.) $a_i^+ \leftarrow \alpha^{-\tilde{x}_i} a_i^+, a_i^- \leftarrow \alpha^{+\tilde{x}_i} a_i^-$
(class 2 err.) Exponential update.

Support Vector Machine (SVM)

Generalize Perceptron with margin and kernel. Find plane that max. margin m s.t.:
$z_i g(\mathbf{y}) = z_i(\mathbf{w}^T\mathbf{y} + w_0) \geq m, \forall \mathbf{y}_i \in \mathcal{Y}$
$z_i \in \{-1,+1\}$ $\quad \mathbf{y_i} = \phi(\mathbf{x_i})$
Vectors $\mathbf{y}_i$ are the support vectors

Functional Margin Problem:
minimizes $||\mathbf{w}||$ for $m=1$: $L(\mathbf{w}, w_0, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^n \alpha_i[z_i(\mathbf{w}^T\mathbf{y}_i + w_0) - 1]$
where αs are Lagrange multipliers.
$\frac{\partial L}{\partial w}=0$ and $\frac{\partial L}{\partial w_0}=0$ give us constraints
$\mathbf{w} = \sum_{i=1}^n \alpha_i z_i \mathbf{y_i}$ $\quad 0 = \sum_{i=1}^n \alpha_i z_i$
Replacing these in L we get (max α)
$\tilde{L}(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_{i,j=1}^n \alpha_i\alpha_j z_i z_j \mathbf{y_i}^T\mathbf{y_j}$
with $\alpha_i \geq 0$ and $\sum_{i=1}^n \alpha_i z_i = 0$
This is the dual representation. The optimal hyperplane is given by
$\mathbf{w}^* = \sum_{i=1}^n \alpha_i^* z_i \mathbf{y_i}$
$w_0^* = -\frac{1}{2}(\min_{z_i=1} \mathbf{w^*}^T\mathbf{y_i} + \max_{z_i=-1} \mathbf{w^*}^T\mathbf{y_i})$
where α maximize the dual problem.
Only Support Vectors ($\alpha_i \neq 0$) contribute to the evaluation.
Optimal Margin: $\mathbf{w}^T\mathbf{w} = \sum_{i\in SV}\alpha_i^*$
Discrim.: $g^*(x) = \sum_{i\in SV} z_i\alpha_i \mathbf{y_i}^T\mathbf{y_i} + w_0^*$
class $= \mathrm{sign}(\mathbf{y}^T\mathbf{w}^* + w_0^*)$
Kuhn-Tucker Conditions: only then strong duality holds: $\quad \alpha_i^* \geq 0$
$\alpha_i^*(z_i g^*(y_i) - 1) = 0, (z_i g^*(y_i) - 1) \geq 0$

Soft Margin SVM

Introduce slack to relax constraints
minimize $\frac{1}{2}w^T w + C\sum_{i=1}^n \xi_i$ with respect to: $z_i(\mathbf{w}^T\mathbf{y}_i + w_0) \geq m(1-\xi_i)$
Lagrangian: $L(\mathbf{w}, w_0, \xi, \alpha, \beta) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i[z_i(\mathbf{w}^T\mathbf{y}_i + w_0) - 1 + \xi_i] - \sum_{i=1}^n \beta_i\xi_i$
C controls margin maximization vs. constraint violation
Dual Problem same as usual SVM but with supplementary constraint: $\quad C \geq \alpha_i \geq 0$
Kuhn-Tucker Conditions: $\alpha_i^*(z_i(w^T y_i + w_0) - 1 + \xi) = 0, \xi_i(\alpha_i - C) = 0$

Non-Linear SVM

Use kernel in discriminant function:
$g(\mathbf{x}) = \sum_{i,j=1}^n \alpha_i z_i K(\mathbf{x_i}, \mathbf{x})$
E.g solve the XOR Problem with:
$K(x,y) = (1 + x_1y_1 + x_2y_2)^2$

Multiclass SVM

$\forall$class $z \in \{1, 2, \cdots, M\}$ we introduce $\mathbf{w}_z$ and define our problem: $\quad$ ($\mathbf{w}$ is stacked)
$min_w \frac{1}{2}w^T w = min_{\{w_z\}_{n=1}^M} \sum_{z=1}^M w_z^T w_z$
s.t. $(\mathbf{w}_{z_i}^T\mathbf{y}_i + w_{z_i,0}) -$
$\quad \max_{z\neq Z_i}(\mathbf{w}_z^T\mathbf{y}_i + w_{z,0}) \geq 1, \forall \mathbf{y}_i \in \mathcal{Y}$
classification: $\hat{z} = argmax_z(\mathbf{w}_z^T y + w_{z,0})$

Structured SVM

Each sample $\mathbf{y}$ is assigned to a structured output label z
Output Space Representation:
joint feature map: $\psi(z, \mathbf{y})$
Scoring function: $f_\mathbf{w}(z,\mathbf{y}) = \mathbf{w}^T\psi(\mathbf{z},\mathbf{y})$
Classify: $\quad \hat{z} = h(\mathbf{y}) = \arg\max_{z\in\mathbb{K}} f_{\mathbf{w}(z,\mathbf{y})}$
SVM objective:
$w^T\psi(z_i, \mathbf{y_i}) - max_{z_j\neq z} w^T\psi(z, \mathbf{y_i}) \geq m$
with margin rescaling: $min_{w,\xi\geq 0}\frac{1}{2}w^T w + C\sum_{i=1}^n \xi_i$ s.t. $\quad w^T\psi(z_i, \mathbf{y_i}) - \Delta(z, z_i) - w^T\psi(z, \mathbf{y_i}) \geq -\xi_i \forall z \neq z_i \forall i$
Lagrangian: let $\mathbb{K}_i = \mathbb{K}\setminus z_i$
$\frac{1}{2}w^T w + C\sum_{i=1}^n \xi_i -$
$\sum_{i=1}^n \sum_{z_j\in\mathbb{K}_i} \alpha_{i,j}(w^T\psi(z_i, y_i) -$
$\Delta(z_j, z_i) - w^T\psi(z_j, y_i) + \xi_i) -$
$\sum_{i=1}^n \beta_i\xi_i$ with $\alpha_{i,j} \geq 0, \beta_i \geq 0$

Ensemble Methods

Combining Regressors

Set of estimators: $\hat{f}_1(x), \cdots, \hat{f}_B(x)$
simple average: $\quad \hat{f}(x) = \frac{1}{B}\sum_{i=1}^B \hat{f}_i(x)$
$\mathrm{Bias}[\hat{f}(x)] = \frac{1}{B}\sum_{i=1}^B \mathrm{Bias}[f_i(x)]$
$\mathbb{V}[\hat{f}(x)] \approx \frac{\sigma^2}{B}$ if the estimators are uncorrelated.

Combining Classifiers

Input: classifiers $c_1(x), \cdots, c_B(x)$
Infer $\hat{c}_B(x) = \mathrm{sgn}(\sum_{b=1}^B \alpha_b c_b(x))$
with weights $\{\alpha_b\}_{b=1}^B$
Requires diversity of the classifiers.

Bagging

Train on bootstrapped subsets.
Sample: $\mathcal{Z} = \{(x_1, y_1), \cdots (x_n, y_n)\}$
$\mathcal{Z}^*$: chose i.i.d from $\mathcal{Z}$ w. replacement.
Covariance small, variance similar, bias weakly affected.

Random Forest (Bagging strategy)

Collection of uncorr. decision trees. Partition data space recursively. Grow the tree sufficiently deep to reduce bias. (each tree on other bagged set and with random collection of features available at every node) Prediction with voting.

Boosting

Combine uncorr. weak learners in sequence. (Weak to avoid overfitting).
Coeff. of $\hat{c}_{b+1}$ depend on $\hat{c}_b$'s results
AdaBoost (minimizes exp. loss)
Init: $\mathcal{X} = \{(x_1, y_1), \cdots, (x_n, y_n)\}$, $w_i^{(1)} = \frac{1}{n}$
Fit $\hat{c}_b(x)$ to $\mathcal{X}$ weighted by $w^{(b)}$
$\varepsilon_b = \sum_{i=1}^n w_i^{(b)}\mathbb{I}_{\{\hat{c}_b(x_i)\neq y_i\}} / \sum_{i=1}^n w_i^{(b)}$
$\alpha_b = \log\frac{1-\varepsilon_b}{\varepsilon_b} > 0$
$w_i^{(b+1)} = w_i^{(b)}\exp(\alpha_b\mathbb{I}_{\{\hat{c}_b(x_i)\neq y_i\}})$
return $\hat{c}_B(x) = \mathrm{sgn}(\sum_{b=1}^B \alpha_b\hat{c}_b(x))$
Best approx. at log-odds ratio.
Like stagewise-additive modeling.

Difference

(1) Boosting keeps identical training data, bagging potentially varies the training data for each classifier. (2) Boosting weighs the prediction of each classifier according to its accuracy, bagging gives same importance to each.

Notes

AdaBoost gives large weight to samples that are hard to classify: those could be outliers. For bagging, there is a chance that imbalanced data-sets lead to bootstrap samples missing a class alltogether. Fix by making the bootstrap size large enough s.t. at least one point is included.

Logistic Regression

$log \frac{P(y=1|x)}{P(y=-1|x)} = \sum_{b=1}^{B} c_b(x) =: F(x)$

$P(y=1|x) = \frac{exp(F(x))}{1+exp(F(x))}$

PAC learning

Function of interest

The probability of large excess error: $\mathbb{P}[misclassification|C] < \delta$.

But: could be unlucky with C, c^{Bayes} not in hypoth. class.

$\mathbb{P}[\mathcal{R}(\hat{c}) - \inf_{c\in\mathcal{C}} \mathcal{R}(c) > \varepsilon] < 1-\delta$.

Def R.H.S.$=\delta$: $\varepsilon = \sqrt{\frac{\log N - \log(\delta/2)}{2n}}$.

$N=$ size of hypothesis class, $n=$ num. of samples.expected error of c depends on $1/\sqrt{n}$ and $\log N$!

for any class $\mathcal{C}$**:** $\mathcal{R}_n(\hat{c}) - \inf_{c\in\mathcal{C}} \mathcal{R}(c) \leq 2\sup_{c\in\mathcal{C}}|\hat{\mathcal{R}}_n(c) - \mathcal{R}(c)|$

for finite class: $\mathbb{P}[2\sup_{c\in\mathcal{C}}|\hat{\mathcal{R}}_n(c) - \mathcal{R}(c)| > \varepsilon] \leq 2|\mathcal{C}|e^{-\frac{1}{2}n\varepsilon^2}$.

Rectangle learning

Pick tight rectangle. Diff. between picked rectangle $\hat{R}$ and true R with few examples. Rectangles are **efficiently PAC learnable**: runs in polynom. $1/\varepsilon$ (error param.) and $1/\delta$ (confidence val.).

Hyperplane learning

Hypothesis: $\sum_{i=1}^{d} a_i x_i + a_0$ (all possible hyperplanes through d-dim vector) has #-of-possible-classifiers $2\binom{n}{d}$. In class: the classifiers c and $\hat{c}$ differ for no more than d data points on a plane, IF found with ERM: $\forall_{c\in\mathcal{C}} \hat{\mathcal{R}}_n(c) \geq \hat{\mathcal{R}}_n(\hat{c}) - \frac{d}{n}$.

VC dimension

If you can find a set of n points, so that it can be shattered by the classifier (i.e. classify all possible 2^n labelings correctly) and you cannot find any set of n+1 points that can be shattered then the VC dimension is n.

Examples: $(-\infty, a] = 1$ all intervals in R: $V_C = 2$ For unions of k intervals, $V_c = 2k$ half planes in R^2: 3 for unit circles $V_c = 3$ convex polygons in R^2: ∞ convex polygons in R^2 with at most k vertices: $2k+1$

Nonparametric Bayesian methods

$\text{Beta}(x|a,b) = B(a,b)^{-1}x^{a-1}(1-x)^{b-1}$: prob. of Bernoulli proc. after observing $a-1$ success and $b-1$ failures. Expended to multivariate case with Dirichlet distr. That will give multivar. probs, *based on finite* counts! But we don't know exactly which multivar. distribution works. With more data, we update the Dirichlet distribution. Is a conjugate prior.

Stick-breaking Dirichl. proc.

Repeatedly draw from $\text{Beta}(x|1,\alpha)$ with fixed α, but from reducing stick: $\rho_k = \beta_k(1 - \sum_{i=1}^{k-1} \rho_i)$. The prior:

$$\mathbb{P}[z_i = k|z_{-i}, \alpha] = \begin{cases} \frac{N_{k,-i}}{\alpha+N-1} & \text{existing } k \\ \frac{\alpha}{\alpha+N-1} & \text{otherwise} \end{cases}$$

Final Gibbs sampler:

$$\mathbb{P}[z_i = k|z_{-i}, \alpha, \mu] = \begin{cases} \frac{N_{k,-i}}{\alpha+N-1} p(x_i|x_{-i,k}, \mu) & \text{existing } k \\ \frac{\alpha}{\alpha+N-1} p(x_i, \mu) & \text{otherwise} \end{cases}$$

Gibbs sampling

Init: assign all data to a cluster, with prior π_i, with $\sum_{k=1}^{K} \pi_i < 1$ (s.t. new clusters possible). E.g. with stick-breaking.

Then remove x from k and compute new θ_k, then compute Gibbs sampler prob. (CRP), and sample the new cluster assignment $z_i \sim p(z_i|x_{-i}, \theta_k)$. If cluster is empty, remove it and decrease K.