Intro to ML Cheat Sheet

By Junru Shao <junrushao1994@gmail.com>

https://github.com/junrushao1994/Cheat-Sheets

General

- MLE: $P(D \mid \theta)$
- MAP: $P(\theta \mid D)$
- Discrete density estimator: MLE $\Leftrightarrow$ counting
- $p_Y(y) = p_X(h(y)) \left| \frac{dh(y)}{dy} \right|$

KNN

- k: # of nearest neighbors. k_1: # of samples labeled 1 in k.
- n: # of samples. n_1: # of samples labeled 1.
- $p(y) = n_1/n$. $p(x \mid y=1) = k_1/n_1V$. $p(x) = k/nV$.
- $p(y=1 \mid x) = p(x \mid y=1)\, p(y)/p(x) = k_1/K$.

Bayes

- Bayes risk: $R(x) = \min\left\{\frac{P(x \mid y=0)P(y=0)}{P(x)}, \frac{P_0(x \mid y=1)P(y=1)}{P(x)}\right\}$
- Bayes error

$$\begin{aligned}\mathbb{E}[R(x)] &= \int_x R(x)\,P(x)\,dx \\ &= P(y=0)\int_{L_1} P(x \mid y=0)\,dx + P(y=1)\int_{L_0} P(x \mid y=1)\,dx\end{aligned}$$

- Naive Bayes
  - discrete: calculate θ_0 and θ_1 by counting (MLE): $L(X \mid y=1;\,\theta) = \prod_j p(x_j \mid y=1;\,\theta_{1,j})$
  - continuous: assume $y_i \sim \text{Multinomial}\left(p_1, \ldots, p_{N_y}\right)$, $X \sim N(\mu_y, \Sigma_y)$ (Σ is diagonal)

$$P(X \mid y) = \prod_j \frac{1}{(2\pi)^{1/2}\sigma_y^j} \exp\left[-\frac{1}{2}\left(\frac{x_j - \mu_y^j}{\sigma_y^j}\right)^2\right]$$

Decision Tree

- Entropy $H(X) = \sum_{i=1}^n -P(X_i)\log P(X_i)$
- Conditional entropy

$$\begin{aligned}H(Y \mid X) &= \sum_{x\in\mathcal{X}} p(x)\,H(Y \mid X=x) \\ &= \sum_{x\in\mathcal{X}} p(x) \sum_{y\in\mathcal{Y}} -p(y \mid x)\log p(y \mid x) \\ &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} -p(x,\,y)\log\frac{p(x,\,y)}{p(x)}\end{aligned}$$

  - Chain rule: $H(Y \mid X) = H(X,\,Y) - H(X)$
  - Bayes rule: $H(Y \mid X) = H(X \mid Y) - H(X) + H(Y)$
- Mutual information = information gain: symmetric, nonnegative

$$I(X;\,Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X) = \sum_x\sum_y p(x,\,y)\log\frac{p(x,\,y)}{p(x)\,p(y)}$$

- Handling overfitting: remove some subtree $\Rightarrow$ decrease validation error $\Rightarrow$ remove

Linear regression

Linear regression: assume $Y = \theta^T X + \varepsilon$, where $\varepsilon \in N\left(0,\,\sigma^2\right)$.

- Maximize LL $\Leftrightarrow$ minimize MSE

$$\begin{aligned}\mathcal{L}(\theta) &= \prod_{i=1}^m \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{\left(y_i - \theta^T x_i\right)^2}{2\sigma^2}\right) \\ \mathcal{LL}(\theta) &= -m\log\left(\sqrt{2\pi}\sigma\right) - \frac{1}{2\sigma^2}\sum_{i=1}^m\left(y_i - \theta^T x_i\right)^2\end{aligned}$$

- with L_2, add prior $\theta \sim N\left(0,\,\lambda^{-1}\right)$

$$\begin{aligned}\mathcal{L}(\theta) &= \prod_{i=1}^m \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{\left(y_i - \theta^T x_i\right)^2}{2\sigma^2}\right)\exp\left(-\frac{\lambda}{2}\theta^T\theta\right) \\ \mathcal{LL}(\theta) &= -m\log\left(\sqrt{2\pi}\sigma\right) - \frac{1}{2\sigma^2}\sum_{i=1}^m\left(y_i - \theta^T x_i\right)^2 - \frac{\lambda}{2}\theta^T\theta\end{aligned}$$

- General linear regression (should be called, general linear model)
  - ϕ_j transforms the j-th feature
  - loss function $J(w) = \sum_i\left(y_i - w^T\phi(x_i)\right)^2$
- Spline: continuity (first-order derivative), smoothness (second-order derivative)
- Locally weighted models, given a point x, data are weighted by $\Omega_x(x_i)$

Logistic regression

- sigmoid: $\sigma(x) = 1/(1+\exp(-x))$, $d\sigma/dx = \sigma(1-\sigma)$
- hypothesis: $h_\theta(x) = \sigma\left(\theta^T x\right)$
- MLE

$$\begin{aligned}\mathcal{L}(\theta) &= \prod_{i=1}^n h_\theta^{y_i}(x_i)\,(1 - h_\theta(x_i))^{1-y_i} \\ \mathcal{LL}(\theta) &= \sum_{i=1}^n y_i\log h_\theta(x_i) + \sum_{i=1}^n (1-y_i)\log(1 - h_\theta(x_i)) \\ \frac{d\mathcal{LL}}{d\theta} &= \sum_{i=1}^n\left(\frac{y_i}{h_\theta(x_i)} - \frac{1-y_i}{1-h_\theta(x_i)}\right)\frac{d\,h_\theta(x_i)}{d\theta} \\ &= \sum_{i=1}^n \frac{y_i - h_\theta(x_i)}{\sigma(\theta^T x_i)(1-\sigma(\theta^T x_i))}\sigma\left(\theta^T x_i\right)\left(1 - \sigma\left(\theta^T x_i\right)\right)\cdot x_i \\ &= \sum_{i=1}^n\left(y_i - \sigma\left(\theta^T x_i\right)\right)x_i\end{aligned}$$

- Softmax regression $\frac{d\mathcal{LL}}{d\theta_k} = \sum_{i=1}^n(\mathbb{I}(y_i = k) - h_\theta(x_i))\,x_i$
- Cross entropy: $H(p,\,q) = \sum_x -p(x)\log q(x)$

Perceptron

- Update: $\mathbf{v}^{t+1} = \mathbf{v}^t + y\mathbf{x}$, where $y \in \{1,\,-1\}$ if made mistake.
- Margin γ: $\exists$unit vector $\mathbf{u}$, $\mathbf{u}\cdot y_i\mathbf{x} > \gamma$. Radius R: all length $\le R$. $\mathbf{v}_k\cdot\mathbf{u} \ge k\gamma$.
- $\|v_k\|^2 \le kR^2$. $k \le (R/\gamma)^2$.
- Delta trick: $d_i = \max(0,\,\gamma - y_i\mathbf{x}_i\mathbf{u})$, $D = \|d_i\|_2$, $Z = \sqrt{1 + D^2/\Delta^2}$.
- Let $\mathbf{u}' = \frac{1}{Z}(u_1,\,\ldots,\,u_n,\,y_1d_1/\Delta,\,\ldots,\,y_md_m/\Delta)$ where $\Delta = \sqrt{RD}$, then $k \le ((R+D)/\gamma)^2$.

MLP

- Universal function approximator I
  - generalized sigmoid: non-decreasing, limit to $-\infty$ is 0, limit to $+\infty$ is 1.
  - Theorem: if $\delta > 0$, g arbitrary sigmoid function, f is continuous on a closed and bounded set A, then $\forall x \in A$, there exists a neural network $\hat{f}$ with 1 hidden layer such that $\left|f(x) - \hat{f}(x)\right| < \delta$
- Universal function approximator II
  - $\mathrm{signNet}^{(2)}(x, w)$ with two hidden layers and sgn activation function is uniformly dense in L_2.

SVM

- generalized Lagrangian
  - geometric margin $\gamma = \frac{1}{\|\mathbf{w}\|}\left|\mathbf{w}^T x + b\right|$
  - primal optimization problem

$$\begin{aligned} \min_w \quad & f(w) \\ s.t. \quad & g_i(w) \le 0 \quad i = 1, \ldots, k \\ & h_i(w) = 0 \quad i = 1, \ldots, l \end{aligned}$$

  - generalized Lagrangian

$$\begin{aligned} \mathcal{L}(w, \alpha, \beta) &= f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w) \\ \theta_{\mathcal{P}}(w) &= \max_{\alpha_i > 0, \beta} \mathcal{L}(w, \alpha, \beta) \\ &= \begin{cases} f(w) & w \text{ satisfies primal constraints} \\ \infty & \text{otherwise} \end{cases} \\ \min_w \theta_{\mathcal{P}}(w) &= \min_w \max_{\alpha_i > 0, \beta} \mathcal{L}(w, \alpha, \beta) \end{aligned}$$

  - dual

$$\begin{aligned} \theta_{\mathcal{D}}(w) &= \min_w \mathcal{L}(w, \alpha, \beta) \\ \max_{\alpha_i > 0, \beta} \theta_{\mathcal{D}}(w) &= \max_{\alpha_i > 0, \beta} \min_w \mathcal{L}(w, \alpha, \beta) \end{aligned}$$

  - comparing primal and dual

$$\begin{aligned} d^* &= \max_{\alpha_i > 0, \beta} \min_w \mathcal{L}(w, \alpha, \beta) \\ &\le \min_w \max_{\alpha_i > 0, \beta} \mathcal{L}(w, \alpha, \beta) \\ &= p^* \end{aligned}$$

  - proof of maximini $\le$ minimax

$$\begin{aligned} \min_\beta f(\alpha, \beta) &\le f(\alpha, \beta) && \forall \alpha, \beta \\ \max_\alpha \min_\beta f(\alpha, \beta) &\le \max_\alpha f(\alpha, \beta) && \forall \beta \\ \max_\alpha \min_\beta f(\alpha, \beta) &\le \min_\beta \max_\alpha f(\alpha, \beta) \end{aligned}$$

- KKT condition when f, g convex, h_i affine, and g_i strictly feasible
  - w^* is solution to primal
  - α^* and β^* is solution to dual
  - w^*, α^* and β^* satisfy

$$\begin{aligned} \frac{\partial}{\partial w_i} \mathcal{L}(w^*, \alpha^*, \beta^*) &= 0 \\ \frac{\partial}{\partial \beta_i} \mathcal{L}(w^*, \alpha^*, \beta^*) &= 0 \\ \alpha_i^* g_i(w^*) &= 0 \\ g_i(w^*) &\le 0 \\ \alpha^* &\ge 0 \end{aligned}$$

  - then w^*, α^* and β^* satisfy KKT, is also solution to primal and dual, and $p^* = d^*$
- support vectors
  - optimization goal

$$\begin{aligned} \min_w \quad & \frac{1}{2}\|w\|^2 \\ s.t. \quad & y_i\left(w^T x_i + b\right) \ge 1 \end{aligned}$$

  - optimal margin

$$\begin{aligned} w &= \sum_{i=1}^{m} \alpha_i y_i x_i \\ \sum_{i=1}^{m} \alpha_i y_i &= 0 \\ b^* &= -\frac{\max_{i:\, y_i = -1} {w^*}^T x_i + \min_{i:\, y_i = 1} {w^*}^T x_i}{2} \end{aligned}$$

  - objective

$$\begin{aligned} \max_\alpha \quad & W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle \\ s.t. \quad & 1 \le y_i \left[b + \sum_{i=1}^{m} \alpha_j y_j \langle x_j, x_i \rangle \right] \\ & \sum_{i=1}^{m} \alpha_i y_i = 0 \\ & \alpha_i \ge 0 \end{aligned}$$

- Soft margin and regularization
  - 0 / 1 loss: $\min_{w,\, b} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m} \mathbb{I}\left(1 - y_i\left(w^T x_i + b\right) > 0\right)$
  - surrogate loss
    - hinge loss: $l(z) = \max(0, 1 - z) = \max\left(0, 1 - y_i\left(w^T x_i + b\right)\right)$
    - exponential loss: $l(z) = \exp(-z)$
    - logistic loss: $l(z) = \log(1 + \exp(-z))$
  - taking hinge loss, and using $\xi_i = \max\left(0, 1 - y_i\left(w^T x_i + b\right)\right)$

* optimization goal

$$\begin{aligned} \min_{w} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i \\ s.t. \quad & y_i\left(w^T x_i + b\right) \ge 1-\xi_i \\ & \xi_i \ge 0 \end{aligned}$$

* dual form

$$\begin{aligned} \max_{\alpha} \quad & W(\alpha) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j \langle x_i,\, x_j\rangle \\ s.t. \quad & \sum_{i=1}^{m}\alpha_i y_i = 0 \\ & 0 \le \alpha_i \le C \end{aligned}$$

Kernel

- Hilbert space H: inner product space, complete metric space (distance induced by inner product)
  - $\langle y,\, x\rangle = \overline{\langle x,\, y\rangle}$
  - $\langle x,\, x\rangle \ge 0$, norm $\|x\| = \sqrt{\langle x,\, x\rangle}$
  - $\langle ax_1 + bx_2,\, y\rangle = a\langle x_1,\, y\rangle + b\langle x_2,\, y\rangle$
  - $\langle x,\, ay_1 + by_2\rangle = a\langle x,\, y_1\rangle + b\langle x,\, y_2\rangle$
  - $d(x,\, y) = \sqrt{\langle x-y,\, x-y\rangle}$, the triangle inequality holds
  - $|\langle x,\, y\rangle| \le \|x\|\,\|y\|$
- reproducing kernel Hilbert space: WTF
- Mercer theorem: let $K:\mathbb{R}^n\times\mathbb{R}^n \mapsto \mathbb{R}^n$ be given. K is a valid kernel $\Leftrightarrow$ for any data points, kernel matrix $\succeq 0$.
- some kernels
  - linear: $k(x_1,\, x_2) = x_1^T x_2$
  - polynomial : $k(x_1,\, x_2) = \left(x_1^T x_2 + c\right)^d$, when $d=2$
    * $k(x,\, y) = \sum_{i=1}^{n} x_i^2 y_i^2 + \sum_{i=2}^{n}\sum_{j=1}^{i}\left(\sqrt{2}x_i x_j\right)\left(\sqrt{2}y_i y_j\right) + \sum_{i=1}^{n}\left(\sqrt{2c}x_i\right)\left(\sqrt{2c}x_i\right) + c^2$
    * $\phi(x) = \left(x_n^2,\, \ldots,\, x_1^2,\, \sqrt{2}x_n x_{n-1},\, \sqrt{2}x_n x_{n-2}\, \ldots,\, \sqrt{2}x_2 x_1,\, \sqrt{2c}x_n,\, \ldots,\, \sqrt{2c}x_1,\, c\right)$
  - Gaussian (radius basis function): $k(x_1,\, x_2) = \exp\left(-\frac{1}{2\sigma^2}\|x_1 - x_2\|_2^2\right)$
  - Laplace: $k(x_1,\, x_2) = \exp\left(-\frac{1}{\sigma}\|x_1 - x_2\|\right)$
  - Sigmoid: $k(x_1,\, x_2) = \tanh\left(\beta x_1^T x_2 + \theta\right)$
- combination of kernels
  - linear combination: $\gamma_1 k_1 + \gamma k_2$
  - direct product: $(k_1 \otimes k_2) = (k_1(x,\, y))(k_2(x,\, y))$
  - for arbitrary $g(x)$: $g(x)\,k(x,\, z)\,g(z)$

Boosting

- Stacking: learning a classifier using predictions from base classifiers
- Voting: weighted vote / confidence vote (ensemble)
- AdaBoost:
  - weighted data samples with D_t & weighted ensembling using α_t

$$\begin{aligned} D_{t+1}(i) &= \frac{1}{Z}D_t(i)\exp\left(-\alpha_t y_i h_t(x_i)\right) \\ Z_t &= \sum_{i=1}^{m} D_t(i)\exp\left(-\alpha_t y_i h_t(x_i)\right) \\ H_x &= \operatorname{sgn}\left(\sum_{t=1}^{T}\alpha_t h_t(x)\right) \end{aligned}$$

  - training error bound

$$\begin{aligned} \frac{1}{m}\sum_{i=1}^{m}\mathbb{I}\left(\operatorname{sgn}\left(\sum_{t=1}^{T}\alpha_t h_t(x_i)\right) \ne y_i\right) &\le \frac{1}{m}\sum_{i=1}^{m}\exp\left(-y_i\sum_{t=1}^{T}\alpha_t h_t(x_i)\right) \\ &= \prod_t Z_t \end{aligned}$$

  - weighted error (for boolean target function like decision trees), choosing α_t

$$\begin{aligned} \varepsilon_t &= \sum_{i=1}^{m} D_t(i)\,\mathbb{I}\left(h_t(x_i) \ne y_i\right) \\ Z_t &= (1-\varepsilon_t)\exp(-\alpha_t) + \varepsilon_t\exp(\alpha_t) \\ \alpha_t &= \frac{1}{2}\ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right) \end{aligned}$$

Active Learning

- Active SVM: current guess of max-margin separator, request label closest to current separator
- Density-based sampling: centroid of largest unsampled cluster
- Uncertainty sampling: closest to decision boundary
- Maximal diversity sampling: maximally distant from labeled x's
- Ensemble-based sampling: ensemble of some above

EM & k-means & GMM

- EM derivation

$$\begin{aligned} \log P\left(D\,|\,\theta^t\right) &= \int_y \log P\left(D\,|\,\theta^t\right)\, dQ(y) \\ &= \underbrace{\underbrace{\int_y \log P\left(y,\, D\,|\,\theta^t\right)\, dQ(y)}_{\mathbb{E}_{y\sim q(y)}[\log P(y,\, D\,|\,\theta^t)]} - \underbrace{\int_y \log q(y)\, dQ(y)}_{H(q)}}_{\text{Free energy: } F_{\theta^t}(q,\, D)} + \underbrace{\int_y \log\frac{q(y)}{P(y\,|\,D,\,\theta^t)}dQ(y)}_{\mathrm{KL}(q\,\|\,P(\cdot\,|\,D,\,\theta^t))} \end{aligned}$$

  - $\mathbb{E}_{y\sim q(y)}\left[\log P\left(y,\, D\,|\,\theta^t\right)\right]$ is expected log-likelihood of data distribution given θ^t
  - $H(q)$ is the entropy of latent variables
  - KL is the divergence between real and posterior distribution of y
  - E-step: fix parameters θ^t, find latent distribution q^t that maximize the likelihood
    * general EM: let $q^t = P\left(\cdot\,|\,D,\,\theta^t\right)$, so that we have

$$q^t = \arg\max_q F_{\theta^t}\left(q,\, D\,|\,\theta^t\right) = \arg\min_q \mathrm{KL}\left(q,\, P\left(\cdot\,|\,D,\,\theta^t\right)\right)$$

* variational methods: when you cannot get a KL $= 0$ (cannot estimate $P\left(\cdot \mid D, \theta^t\right)$)

- M-step: fix latent distribution q^t, find parameters θ^{t+1} that maximize the likelihood
  * $\theta^{t+1} = \arg\max_\theta F_\theta\left(q^t, D\right) = \arg\max_\theta Q\left(\theta \mid \theta^t\right)$
  * where $Q\left(\theta^{t+1} \mid \theta^t\right) = \mathbb{E}_{y \sim P(y \mid D, \theta^t)}\left[\log P\left(y, D \mid \theta^{t+1}\right)\right]$

- Mixture of K Gaussian:

$$p(x) = \sum_{i=1}^{K} p(x \mid y = i) P(y = i)$$

  - Mixture component: $p(x \mid y = i)$
  - Mixture proportion: $P(y = i)$
  - MLE: find $\arg\max_\theta \prod_{j=1}^n P(x_j \mid \theta)$

$$mle = \arg\max_\theta \prod_{j=1}^{n} \sum_{i=1}^{K} P\left(y_j = i \mid \theta\right) p\left(x_j \mid y_j = i, \theta\right)$$
$$= \begin{cases} \arg\max_\theta \prod_{j=1}^n \sum_{i=1}^K \frac{\pi_i}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-1}{2\sigma^2}\|x_j - \mu_i\|^2\right) \\ \arg\max_\theta \prod_{j=1}^n \sum_{i=1}^K \frac{\pi_i}{\sqrt{|2\pi\Sigma_i|}} \exp\left(-\frac{1}{2}(x_j - \mu_i)^T \Sigma_i^{-1}(x_j - \mu_i)\right) \end{cases}$$

- Spherical, same variance GMMs
  - E-step

$$\begin{aligned} R_{i,j}^{t-1} &= P\left(y_j = i \mid x_j, \theta^{t-1}\right) \\ &\propto \pi_i \exp\left(-\frac{1}{2\sigma^2}\left\|x_j - \mu_i^{t-1}\right\|^2\right) \\ &\quad \text{(normalize over } i \in [1, k]) \end{aligned}$$

  - M-step

$$\begin{aligned} Q\left(\mu_i^t \mid \theta^{t-1}\right) &\propto \sum_{j=1}^{n} R_{i,j}^{t-1}\left(-\frac{1}{2\sigma^2}\left\|x_j - \mu_i^t\right\|^2\right) \\ \frac{\partial}{\partial \mu_i^t} Q\left(\mu_i^t \mid \theta^{t-1}\right) &= \sum_{j=1}^{n} R_{i,j}^{t-1}\left(x_j - \mu_i^t\right) \\ &= 0 \\ \mu_i^t &= \sum_{j=1}^{n} w_j x_j \\ w_j &\propto R_{i,j}^{t-1} \\ &\quad \text{(normalize over } j \in [1, N]) \end{aligned}$$

- General GMM
  - E-step

$$R_{i,j}^{t-1} \propto \exp\left(-\frac{1}{2}\left(x_j - \mu_i^{t-1}\right)^T \Sigma^{-1}\left(x_j - \mu_i^{t-1}\right)\right)\pi_i^{t-1}$$

  - M-step

$$\begin{aligned} \mu_i^t &= \sum_{j=1}^{n} w_j x_j \\ w_j &\propto R_{i,j}^{t-1} \\ \Sigma_i^t &= \sum_{i=1}^{n} w_j\left(x_j - \mu_i^t\right)^T\left(x_j - \mu_i^t\right) \\ \pi_i^t &= \frac{1}{n}\sum_{j=1}^{n} R_{i,j}^{t-1} \end{aligned}$$

PCA

- Attention: sample should be centered $\sum_{i=1}^m \mathbf{x}_i = \mathbf{0}$
- Projection: $z_{ij} = \mathbf{w}_j^T \mathbf{x}_i$ and $\mathbf{z}_i = W^T \mathbf{x}_i$
- Reconstruction: $\mathbf{x}' = \sum_{i=1}^{d'} \mathbf{w}^T \mathbf{x}$
- Orthogonal space: $W^T W = \mathbf{I}$
- Two equivalent objectives
  - minimize error:

$$\begin{aligned} \sum_{i=1}^{m}\left\|\sum_{j=1}^{d'} z_{ij}\mathbf{w}_j - \mathbf{x}_i\right\|_2^2 &= \sum_{i=1}^{m} \mathbf{z}_i^T \mathbf{z}_i - 2\sum_{i=1}^{m} \mathbf{z}_i^T W^T \mathbf{x}_i + \text{const} \\ &\propto -\text{tr}\left(W^T X X^T W\right) \end{aligned}$$

  - maximize variance:

$$\sum_i W^T x_i x_i^T W = \text{tr}\left(W^T X X^T W\right)$$

- Lagrange multipliers: $XX^T \mathbf{w}_i = \lambda_i \mathbf{w}_i$
- Trick: use $L = X^T X$ instead of $\Sigma = XX^T$. If v is eigenvector of L, then Xv is eigenvector of Σ
- SVD: centered data matrix $X \in \mathbb{R}^{N \times M}$ where N is # of features, M is # of samples
  - $X = USV^T$, where $U \in \mathbb{R}^{N \times N}$, $S \in \mathbb{R}^{N \times M}$, $V \in \mathbb{R}^{M \times M}$
  - U, V are unitary, S is diagonal
  - Each column of U is a PC
  - $\Sigma = XX^T = \sum_{i=1}^N \lambda_i p_i p_i^T$
  - $S = \text{diag}\left(\sqrt{\lambda_1}, \sqrt{\lambda_2}, \ldots, \sqrt{\lambda_r}, 0, 0\right)$ where $r = \text{rank}\left(X^T X\right)$
  - $S = \text{diag}\left(\sigma_1, \ldots, \sigma_r, 0, 0\right)$, σ_i^2/n is the variance when X is projected to the corresponding PC
  - Let $P = SV^T$, then P_{ij} is the coordinate of sample j projected to PC i
  - Each PC u is a weighted sum of data points: $u = \sum_{i=1}^m \alpha_i x_i$, where $\alpha_i = \frac{X_i^T u}{\lambda m}$

ICA

- Goal: $X = AS \in \mathbb{R}^{N \times M}$, find $W = A^{-1}$ s.t. $S = WX$, $\mathbb{E}\left[SS^T\right] = I_N$ and $\mathbb{E}[S] = 0$
- Whitening:
  - first center X, i.e. removes mean of X
  - then remove covariance of X
    * let $\Sigma = \text{cov}(X) = \mathbb{E}\left[XX^T\right] = AA^T = UDU^T$ (eigenvalue decomposition $UU^T = I$)
    * let $Q = D^{-1/2}U^T$ be the whitening matrix
    * let $X^* = QX$, then $X^* X^{*T} = I$, $A^* A^{*T} = I$

- Whitening matrix: $Q = D^{-1/2}U^T$, then $A^* = QA$, $A^* A^{*T} = I_m$
  - $\Sigma = \mathrm{cov}(X) = \mathbb{E}\left[XX^T\right] = A\mathbb{E}\left[SS^T\right]A^T = AA^T$
  - SVD: $\Sigma = UDU^T$, where $UU^T = I_M$
- Find an orthogonal matrix W optimizing an objective funcion $J(Y)$, where $Y = WX$
  - an orthogonal matrix is the production of a sequence of rotation $\log|\det W| = 0$
  - minimize the mutual information between $y_1, \ldots, y_n$

$$\begin{aligned} J_{\mathrm{ICA}_1}(w) &= \int p(y_1, \ldots, y_n)\log\frac{p(y_1, \ldots, y_n)}{p(y_1)\cdots p(y_n)}dy \\ &= -H(y_1, \ldots, y_n) + H(y_1) + \cdots + H(y_n) \\ &= -H(x_1, \ldots, x_n) - \log|\det W| + H(y_1) + \cdots + H(y_n) \\ &\propto H(y_1) + \cdots + H(y_n) \end{aligned}$$

  - normal distribution has maximum entropy, we should deviate y_i from normal
  - Kurtosis: $\kappa_4(y) = \mathbb{E}\left[y^4\right] - 3\left(\mathbb{E}\left[y^2\right]\right)^2$
- Objective: $\max f(W) = \mathbb{E}\left[y^4\right] - 3$, subject to $\|W\|^2 - 1 = 0$
  - Newton's method

$$x_{k+1} = x_k - \frac{\phi(x_k)}{\phi'(x_k)}$$

  - Newton's method (multivariate)

$$x_{k+1} = x_k - \left[\nabla F(x_k)\right]^{-1} F(x_k)$$

  - apply Lagrange Multiplier: let w be the first ICA vector: $f'(W) + \lambda\hat{h}(W) = 0$, let

$$\begin{aligned} F(w) &= 4\mathbb{E}\left[\left(w^T z\right)^3 z\right] + 2\lambda w \\ F'(w) &= 12\mathbb{E}\left[\left(w^T z\right)^2 zz^T\right] + 2\lambda I \\ &\sim 12\mathbb{E}\left[\left(w^T z\right)^2\right]\mathbb{E}\left[zz^T\right] + 2\lambda I \\ &= (12 + 2\lambda) I \end{aligned}$$

  - follow the Newton's method

$$\begin{aligned} w_{k+1} &= w_k - \frac{4\mathbb{E}\left[\left(w^T z\right)^3 z\right] + 2\lambda w}{(12 + 2\lambda)} \\ -\frac{12 + 2\lambda}{4} w_{k+1} &= -3w_k + \mathbb{E}\left[\left(w^T z\right)^3 z\right] \\ \widetilde{w}_{k+1} &= \mathbb{E}\left[\left(w^T z\right)^3 z\right] - 3w_k \\ \widetilde{w}_{k+1} &= \frac{\widetilde{w}_{k+1}}{\|\widetilde{w}_{k+1}\|} \end{aligned}$$

  - when we get w_1, calculate w_2 with additional constraint $w \perp w_1$

SSL

- Self training: augment data using a subset of unlabeled data, paired with predicted label
- Generative methods:

$$\begin{aligned} \log p(X_l, Y_l, X_u \mid \theta) &= \sum_{i=1}^{l}\log p(x_i, y_i \mid \theta) + \lambda\sum_{i=l+1}^{l+u}\log p(x_i \mid \theta) \\ &= \sum_{i=1}^{l}\log p(x_i, y_i \mid \theta) + \lambda\sum_{i=l+1}^{l+u}\log\sum_{y} p(x_i, y \mid \theta) \end{aligned}$$

- Graph regularization: k-NN graph, fc graph, ϵ-radius graph,

$$\min_f \left\{\sum_{i\in l}(y_i - f_i)^2 + \lambda\underbrace{\sum_{i, j\in l, u} w_{ij}(f_i - f_j)^2}_{\text{smoothness}}\right\}$$

- Co-training
  - assumption: 1) features can be split into two sets; 2) each sub-feature is sufficient to train a good classifier
  - sach classifier teaches the other classifier with the few unlabeled examples
- Semi-supervised SVMs
  - assumption: unlabeled data are separated with large margin

Learning Theory

- Risk: $R_{L, P}(f) = \mathbb{E}_{(x, y)\sim P(x, y)}\left[L(x, y, f(x))\right]$. $R(f)$ is the abbreviation.
- Bayes risk: $R^*_{L, P} = \inf_{f\in\mathcal{H}} R_{L, P}(f)$. $R^*_{\mathcal{F}}$ is Bayes risk over $\mathcal{F}$.
- Empirical risk: $\hat{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i)) \to R(f)$.
- ERM: $f^*_n = \arg\min_{f\in\mathcal{F}} \hat{R}_n(f)$
- Universally consistent: $R_{L, P}(f_D) \xrightarrow{p} R^*_{L, P}$ as $n = |D| \to \infty$
- No free lunch: for every consistent learning method, any convergence rate a, $\exists P(X, Y)$ s.t. this learning method on P is slower than a
- Approximation (model) error: $R^*_{\mathcal{F}} - R^* \geq 0$
- Estimation error: $R\left(f^*_{n, \mathcal{F}}\right) - R^*_{\mathcal{F}} \geq 0$
- Goal: empirical risk captures true risk: $R\left(f^*_{n, \mathcal{F}}\right) - R^* = \left(R\left(f^*_{n, \mathcal{F}}\right) - R^*_{\mathcal{F}}\right) + \left(R^*_{\mathcal{F}} - R^*\right)$
- PAC framework: find n such that $P\left(R(f^*_n) - \inf_{f\in\mathcal{F}} R(f) > \varepsilon\right) < \delta$

	risk of a given function f	risk of best function f^*	best function f^*
Bayes	$R(f) = P(Y \neq f(X))$	$R^* = R(f^*) = \inf_f R(f)$	$f^* = \arg\min_f R(f)$
$\mathcal{F}$		$R^*_{\mathcal{F}} = R\left(f^*_{\mathcal{F}}\right) = \inf_{f\in\mathcal{F}} R(f)$	$f^*_{\mathcal{F}} = \arg\min_{f\in\mathcal{F}} R(f)$
ER	$\hat{R}_n = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}(Y_i \neq f(X_i))$	$\hat{R}^*_{n, \mathcal{F}} = \inf_{f\in\mathcal{F}} \hat{R}_n(f)$	$f^*_{n, \mathcal{F}} = \arg\min_{f\in\mathcal{F}} \hat{R}_n(f)$

  - EMR minus true risk: $\left|\hat{R}\left(f^*_{n, \mathcal{F}}\right) - R\left(f^*_{n, \mathcal{F}}\right)\right| \leq \sup_{f\in\mathcal{F}}\left|\hat{R}_n(f) - R(f)\right|$
  - Estimation error bound (true risk by EMR, true risk by best f): $\left|R\left(f^*_{n, \mathcal{F}}\right) - R^*_{\mathcal{F}}\right| \leq 2\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f) - R(f)\right|$
  - Using Hoeffding's bound: $P\left(\left|\hat{R}_n(f) - R(f)\right| > \varepsilon\right) \leq 2\exp\left(-2n\varepsilon^2\right)$
  - Union bound (where $N = |\mathcal{F}|$)

$$P\left(\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f) - R(f)\right| \geq \varepsilon\right) \leq 2N\exp\left(-2n\varepsilon^2\right)$$

- Expected deviation:

$$\mathbb{E}\left[\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|\right]\le\sqrt{\frac{\log 2N}{2n}}$$

- Vapnik-Chervonenkis inequality:

$$\mathbb{E}\left[\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|\right]\le 2\sqrt{\frac{\log 2S_{\mathcal{F}}(n)}{n}}$$

- Vapnik-Chervonenkis Theorem:

$$\begin{aligned}P\left(\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|\ge\varepsilon\right) &\le 4S_{\mathcal{F}}(2n)\exp\left(-2n\varepsilon^2/8\right)\\ P\left(\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|\ge\varepsilon\right) &\le 8S_{\mathcal{F}}(n)\exp\left(-2n\varepsilon^2/32\right)\end{aligned}$$

- Bounded difference

$$P\left(\left|\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|-\mathbb{E}\left[\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|\right]\right|\ge\varepsilon\right)\le 2\exp\left(-2\varepsilon^2 n\right)$$

- Growth function, Shatter coefficient: max number of behaviors on n points
  - $S_{\mathcal{F}}(x_1,\ldots,x_n)=|\{f(x_1),\ldots,f(x_n)\};\ f\in\mathcal{F}|$
  - $S_{\mathcal{F}}(n)=\max_{x_1,\ldots,x_n}|\{f(x_1),\ldots,f(x_n)\};\ f\in\mathcal{F}|$
  - $\mathcal{F}$ shatters $x_1\cdots x_n$ iff $\mathcal{F}$ has all 2^n behaviors on the sample
- VC dimension: $\mathrm{VC}_{\mathcal{F}}=\max\{n:\ S_{\mathcal{F}}(n)=2^n\}$
  - you select the best $x_1,\ldots,x_n$
  - adversary assigns label $y_1,\ldots,y_n$
  - if $\mathrm{VC}_{\mathcal{F}}\ge n$, you can find $f\in\mathcal{F}$ that is consistent with the labels
- Sauser's lemma:

$$\begin{aligned}S_{\mathcal{F}}(n) &\le \sum_{k=0}^{\mathrm{VC}_{\mathcal{F}}}\binom{n}{k}\\ S_{\mathcal{F}}(n) &\le \left(\frac{ne}{\mathrm{VC}_{\mathcal{F}}}\right)^{\mathrm{VC}_{\mathcal{F}}}\end{aligned}$$

- VC inequality + Sauser's lemma:

$$\begin{aligned}\mathbb{E}\left[\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|\right] &\le 2\sqrt{\frac{\mathrm{VC}_{\mathcal{F}}\log(n+1)+\log 2}{n}}\\ \mathbb{E}\left[\hat{R}_n(f)-R(f)\right] &\le 4\sqrt{\frac{\mathrm{VC}_{\mathcal{F}}\log(n+1)+\log 2}{n}}\end{aligned}$$

- VC theorem + Sauser's lemma:

$$P\left[\sup_{f\in\mathcal{F}}\left|\hat{R}_n(f)-R(f)\right|\le 8\sqrt{\frac{\log S_{\mathcal{F}}(n)+\log\frac{8}{\delta}}{2n}}\right]\ \ge\ 1-\delta$$

Bayesian Network

- Parent: direct predecessor; Children: direct successor
- Number of parameters: $\sum_{v\in V}2^{|\mathrm{pred}(v)|}$
- Markov blanket: direct predecessors, direct successors, direct successors' predecessors
  - given Markov blanket, a variable is conditionally independent of all other variables
- d-separation: given a set of Z, x and y are independent of each other. $I(x,y\,|\,Z)$
- Collider: if x and y both have a path to this node
- d-connected given Z: variables that are not d-connected are d-separated (path is bidirectional)
  - exists a path between x and y containing no collider or any member of Z (Z can be empty)
  - Z contains a collider or one of its successors, and exists a $x-y$ path that contains this node'
  - *** A version in human language
    - first assume X and Y are independent
    - if there is a bidirectional path between X and Y, we say X and Y are dependent
    - members of Z and all colliders (nodes that have >1 direct predecessors) will block a path
    - $z\in Z$ will unblock a node on the path if it is predecessors (inclusive) of z and it is a collider
- Complete joint distribution: product all parameters in the network
- Stochastic inference: sample free variable, sample other variables based on conditional distribution
  - fix variables that are conditioned on, accumulate the complete joint distribution
- Variable elimination: trivial
- Convert network to a polytree

HMM

- Definition
  - states: $\{s_1,\ldots,s_n\}$
  - Π_i the probability starting at state s_i
  - transition matrix $P(q_t=s_i\,|\,q_{t-1}=s_j)\overset{\mathrm{def}}{=}a_{j,i}$
  - possible outputs Σ
  - emission probability at state s $p(o_t=\sigma\,|\,s)\overset{\mathrm{def}}{=}b_i(o_t)$
- Calculate $P(q_t=A)$: DP. Time complexity $O\left(n^2t\right)$
  - $P_1(i)=\pi_i$ (prior)
  - $P_t(i)=\sum_j p(q_t=s_i;\,q_{t-1}=s_j)\,P_{t-1}(j)=\sum_j a_{j,i}P_{t-1}(j)$
- Calculate $P(Q\,|\,O)=\frac{P(O\,|\,Q)P(Q)}{P(O)}$, $P(O\,|\,Q)$ and $P(Q)$ is easy
  - Let $\alpha_t(i)=P(O\wedge q_t=s_i)$, can be calculated using DP. Time complexity $O\left(n^2t\right)$

$$\begin{aligned}\alpha_1(i) &= P(o_1\wedge q_1=i)\\ &= P(o_1\,|\,q_1=s_i)\,\pi_i\\ \alpha_{t+1}(i) &= P(o_1,\ldots,o_{t+1}\wedge q_{t+1}=s_i)\\ &= \sum_j b_i(o_{t+1})\,a_{j,i}\alpha_t(j)\\ P(O) &= \sum_i\alpha_t(i)\\ P(q_t=s_i\,|\,o_1,\ldots,o_t) &= \frac{\alpha_t(i)}{\sum_j\alpha_t(j)}\end{aligned}$$

- Find the best path that matches observation: $\arg\max_Q P(Q\,|\,O)=\arg\max_Q P(O\,|\,Q)\,P(Q)$

– Prob of the best previous states & observation whose final state is s_t

$$\begin{aligned}
\delta_t(i) &= \max_{q_1,\ldots,q_{t-1}} P(q_1,\ldots,q_{t-1} \wedge q_t = s_i \wedge o_1,\ldots,o_t) \\
\delta_1(i) &= P(q_1 = s_i \wedge o_1) \\
&= P(o_1 \mid q_1 = s_i)\,\pi_i \\
\delta_{t+1}(i) &= \max_{q_1,\ldots,q_t} P(q_1,\ldots,q_{t+1} = s_i \wedge o_1,\ldots,o_{t+1}) \\
&= \max_j \delta_t(j)\, P(q_{t+1} = s_i \mid q_t = s_j)\, P(o_{t+1} \mid q_{t+1} = s_i) \\
&= \max_j \delta_t(j)\, a_{j,i} b_i(o_{t+1})
\end{aligned}$$

– Then, we have

$$\begin{aligned}
Q^* &= \arg\max_Q P(Q \mid O) \\
&= \text{path defined by } \arg\max_j \delta_t(j)
\end{aligned}$$

- Training
  - Forward function: $\alpha_t(i) = P(o_1,\ldots,o_t,\, q_t = s_i) = \sum_j a_{j,i} b_i(o_t)\,\alpha_{t-1}(j)$
  - Backward function: $\beta_t(i) = P(o_{t+1},\ldots,o_T \mid q_t = s_i) = \sum_j b_j(o_{t+1})\, a_{i,j} \beta_{t+1}(j)$

$$\begin{aligned}
\beta_{t-1}(i) &= P(o_t \mid q_{t-1} = s_i) \\
&= \sum_j P(o_t,\, q_t = s_j \mid q_{t-1} = s_i) \\
&= \sum_j P(o_t \mid q_t = s_j,\, q_{t-1} = s_i)\, P(q_t = s_j \mid q_{t-1} = s_i) \\
&= \sum_j b_j(o_t)\, a_{ij}
\end{aligned}$$

  - Prob of a state given all observations

$$s_t(i) = P(q_t = s_i \mid O) = \frac{\alpha_t(i)\,\beta_t(i)}{\sum_i \alpha_t(i)\,\beta_t(i)}$$

  - Transition prob given all observations

$$s_t(i,j) = P(q_t = s_i,\, q_{t+1} = s_j \mid O) = \frac{q_t(i)\, a_{i,j} b_j(o_{t+1})\,\beta_{t+1}(i)}{\sum_i \alpha_t(i)\,\beta_t(i)}$$

  - EM
    * Init: guess initial distribution & emission probs, calculate initial a and b
    * E-step: compute $s_t(i)$ and $s_t(i,j)$ using a and b
    * M-step: update a and b using counting
      · update a

$$\begin{aligned}
\hat{n}(i,j) &= \sum_t s_t(i,j) \\
a_{i,j} &= \frac{\hat{n}(i,j)}{\sum_k \hat{n}(i,k)}
\end{aligned}$$

      · update b

$$\begin{aligned}
B_k(j) &= \sum_{t \,\mid\, o_t = j} s_t(k) \\
b_k(j) &= \frac{B_k(j)}{\sum_i B_k(i)}
\end{aligned}$$