Introduction to Machine Learning summary

Michael Van Huffel, Dino Colombo

August 6, 2022

Introduction to Machine Learning summary created by *michavan@student.ethz.ch* && *dicolomb@student.ethz.ch*

This summary has been written based on the Lecture 252-0220-00 S Introduction to Machine Learning by Prof. A. Krause (Spring 22s). There is no guarantee for completeness and/or correctness regarding the content of this summary. This summary is a corrected, modified and a more completed version of the summary of Yannick Merkli. Use it at your own discretion

Basics

Bayes: $P(X \mid Y)P(Y) = P(X \cap Y)$
$P(X,Y) = P(X|Y)P(Y) = P(Y|X)P(X)$
$P(X,Y,Z) = P(X|Y,Z)P(Y|Z)P(Z)$
X,Y iid: $P(X,Y|Z) = P(X|Z)P(Y|Z)$
$\sigma_X^2 = Var[X] = \mathbb{E}[(X-\mu_X)^2] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$
$ln(x) \leq x-1, x>0$; $||x||_2 = \sqrt{x^T x}$
Jensen ineq: $g(E[X]) \leq\geq E[g(X)]$, if $g \smile \frown$
Div. $KL(p||q) = \mathbb{E}_p[log(\frac{p(x)}{q(x)})] \neq KL(q||p)$
$JSD(p||q) = \frac{1}{2}[KL(p||\frac{1}{2}(p+q)) + KL(q||\frac{1}{2}(q+p))]$
Cauchy-Sch.: $|\langle x,y\rangle| \leq \|x\| \cdot \|y\|$, *strong*
Orth: $A^{-1} = A^T, AA^T = A^TA = I, \det = \pm 1$
Inv: $A^{-1} = \frac{1}{\det(A)} \begin{bmatrix} A_{22} & -A_{12} \\ -A_{21} & A_{11} \end{bmatrix}$
Der: $\partial_x(x^TAx) = (A^T+A)x$, $\partial_x(b^TAx) = A^Tb, \partial_X(c^TX^Tb) = bc^T, \partial_X a^TX^{-1}b = -X^{-T}ab^TX^{-T}, a^Ta = \mathrm{Tr}(aa^T), \mathrm{Tr}(A) = \sum \lambda_i$
Eigdec: $A = Q\Lambda Q^{-1}$, $\lambda_{1,2}^{2x2} = \frac{\mathrm{Tr}(A) \pm \sqrt{\mathrm{Tr}(A)^2 - 4\det(A)}}{2}$, $\mathbf{v}_{1,2} \propto \begin{bmatrix} A_{12} \\ \lambda_{1,2} - A_{11} \end{bmatrix}$
SVD: $X \in \mathbb{R}^{n\times p}, U \in \mathbb{R}^{n\times n}, S \in \mathbb{R}^{n\times p}, V \in \mathbb{R}^{p\times p}$
$X = USV^T = \sum_{k=1}^{rank(X)} \sigma_{k,k} u_k (v_k)^T$
$X^TX = VS^TU^TUSV^T = VS^TSV^T = V\Sigma V^T$
$\Sigma = diag(\sigma_i^2); \sigma_i^2 = \lambda_i; \forall \lambda_i \geq 0$, $(U^TU = V^TV = I)$

Convex: $\forall x_1, x_2, h(\lambda x_1 + (1-\lambda)x_2) \leq \lambda h(x_1) + (1-\lambda)h(x_2)$, strong if $h(x) - m/2\|x\|^2 \smile$
Operations: $\alpha f + \beta h \smile$ if $\alpha, \beta \geq 0$, $f \smile, h \smile$; $f(h(x))$ if $f \smile$, h affine or f non-decreasing, $g \smile$; $\max\{f(x), h(x)\}$ if f convex, h convex

Gaussian distribution: $\log \prod_i^n f(x_i) = -n \ln \sigma - \frac{n}{2}\ln(2\pi) - \sum_{i=1}^n \frac{1}{2\sigma^2}(x_i - \mu)^2$
$f(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^k |\Sigma|}} \exp(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu}))$

Regression - S

Linear Regression $f(x) = w^Tx, w^* = \min_w \hat{R}$
$\hat{R}(w) = \sum_{i=1}^n (y_i - w^Tx_i)^2 = ||Xw - y||_2^2$
Clf: $\in \mathcal{O}(nd^2 + d^3)$: $w^* = (X^TX)^{-1}X^Ty$, $X \in \mathbb{R}^{n\times d}$, $w \in \mathbb{R}^{d\times 1}$; $\nabla_w \hat{R}(w) = -2\sum_{i=1}^n (y_i - w^Tx_i) \cdot x_i = 2X^T(Xw - y) \in \mathcal{O}(nd)$
Regularized regression (*bias* ↑, *variance* ↓)
Ridge/Lasso: $\min_w \sum_{i=1}^n (y_i - w^Tx_i)^2 + \lambda||w||_{2,1}^2$
Close f.: $w^* = (X^TX + \lambda I)^{-1}X^Ty = \frac{1}{1+\lambda} w_{OLS}^*$
In general: $\|w\|_2 \leq \|w\|_1 \leq \sqrt{||w||_0}||w||_2$
Gradient descent $\mathcal{O}(nd^2 + \tau d^2)$
$\|A\|_{op} = \max\{|\lambda_{\min}|, |\lambda_{\max}|\}$, SVD for $X^\top X$
$\|w^t - w_{\min}\|_2 \leq \left[\|I - \eta X^\top X\|_{op}\right]^t \|w^0 - w_{\min}\|_2$
SGD: 1pt, batch ↓, variance ↑, P(esc. sdl.) ↑

k-CV: k ↑, ↓ bias, ↑ var, $R(gen)$ ↓, $\hat{R}(valid)$ ↑
LOOCV: ↑ $\mathcal{O}(\cdot)$, high var, overfitng, $= n$-fold.

Classification - S ($\ell_{0-1}(f(x), y) = \mathbb{I}_{\{y \neq \mathrm{sign}\,\hat{f}(x)\}}$)

Surrogated losses (0-1, non-convex, non-cont.)
Hinge loss: $l_{\text{hinge}}(y\hat{f}(x)) = \max(0, 1 - y\hat{f}(x))$
Logistic loss: $l_{\log}(y\hat{f}(x)) = \log\left(1 + e^{-y\hat{f}(x)}\right)$
CE: $l_{\text{CE}}(\widehat{f}(x), y) = -y \log p(\widehat{Y} = 1 \mid x) - (1-y)\log(1 - p(\widehat{Y} = 1 \mid x)), Y \in [0,1]$
Softmax: $\mathrm{softmax}_\alpha(\boldsymbol{v})_i = \frac{e^{\alpha v_i}}{\sum_{j=1}^K e^{\alpha v_j}}$

Multi-Class Classification
$\hat{p}_k = \mathrm{softmax}(\hat{f}(x))_k$, $\tilde{f}(x) = \left(\hat{f}_1(x), \ldots, \hat{f}_K(x)\right)$
$\hat{y} = \mathrm{argmax}_{k=1\ldots K}\, \hat{f}_k(x) = \mathrm{argmax}_{k=1\ldots K}\, \hat{p}_k$
OvA: $\hat{y}_i = \mathrm{argmax}_{j\in\{1,..,c\}}\, w_j^T x_i$; C bin. classif
OvR: Train $\frac{c(c-1)}{2}$ bin. classif., one for each pair (i,j). Class with most positive predictions wins (slower, but no confidence needed)
Support Vector Machine (margin $= 1/\|w\|$)
H: $\mathscr{L}(w,b,\alpha) = \frac{1}{2}\|\mathbf{w}\|_2^2 - \sum_{i=1}^n \alpha_i(y_i(\mathbf{x}_i^\top \mathbf{w} + b) - 1)$
S: $\min_w \frac{1}{n}\sum_{i=1}^n (\max\{0, 1 - y_i w^T x_i\} + \lambda||w||_2^2)$

$$\nabla_w l_i(w) = \begin{cases} -y_i x_i + 2\lambda w & \text{, if } y_i w^T x_i < 1 \\ 2\lambda w & \text{, if } y_i w^T x_i \geq 1 \end{cases}$$

Metrics: rand. classifier → $FPR = TPR\ \forall\tau$
Accuracy$= \frac{\#\text{correct predictions}}{\#\text{all predictions}} = \frac{TP+TN}{TP+TN+FP+FN}$,
Precis: $\frac{TP}{\#[\hat{y}=+1]} = \frac{TP}{TP+FP} \to P(y=1 \mid \hat{y}=1)$
Rec $TPR = \frac{TP}{\#[y=1]} = \frac{TP}{TP+FN} \to P(\hat{y}=1|y=1)$
$FPR = \frac{FP}{\#[y=-1]} = \frac{FP}{TN+FP} \to P(\hat{y}=1|y=-1)$
$FDR = \frac{FP}{\#[\hat{y}=+1]} = 1 - Prec. \to P(y=-1|\hat{y}=1)$
F1 score $= \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$, $FNR = 1 - TPR$

Kernels $k : \mathscr{X} \times \mathscr{X} \to \mathbb{R}$; $x_i^T x_j \to k(x_i, x_j)$

Feature explosion ($p = \mathscr{O}\binom{d+m}{m} \approx \frac{(d+m-1)\cdots d}{m\cdots 1}$)
Polynomial $\phi(x)$, m degree polynomial features in dimension d, n features: $\sim \mathcal{O}(n^2 d^m)$ ↑ d, $\sim \mathcal{O}(n^2 m^d)$ ↑ m; with kernel $k(x,z) = \langle\phi(x), \phi(z)\rangle = (1 + \langle x, z\rangle)^m \sim \mathcal{O}(n^2(d+m))$.
Properties of kernel $k(x,y) = \phi(x)^T\phi(y)$
Mercer's Theorem → infinite feature map k must be symmetric: $k(x,y) = k(y,x)$
Kernel matrix must be positive semi-definite.
Kernel engineering
$k_1(x,y) + k_2(x,y)$; $k_1(x,y) \cdot k_2(x,y)$; $c \cdot k_1(x,y)$,$c > 0$; $f(k_1(x,y))$, f polynomial with positive coefficients or the exponential function.
Examples of kernels on $\mathbb{R}^d$
Linear kernel: $k(x,y) = x^Ty$
Polynomial kernel: $k(x,y) = (x^Ty + 1)^d$
Gaussian kernel: $k(x,y) = \exp(-||x-y||_2^2/h^2)$
Laplacian kernel: $k(x,y) = \exp(-||x-y||_1/h)$

h ↑, overft ↓, distr. wider, flatter, B. smoother.
Parametric (finite): LR, linear perceptron.
Non-parametric (∞): ker. perceptron, k-NN
$k(x,y) = \min(x,y), |A \cap B|, \frac{1}{\max(\cdot)} = \min(\frac{1}{x}, \frac{1}{y})$
Kernelized linear regression $\hat{w} = \sum_i \alpha_i \phi(x_i)$
$w^* = \min_{\alpha_{1:n}} \sum_{i=1}^n (\sum_{j=1}^n \alpha_j \langle\phi(x_i), \phi(x_j)\rangle - y_i)^2 + \lambda \sum_{i,j} \alpha_i\alpha_j\langle\phi(x_i), \phi(x_j)\rangle = \min_\alpha ||\alpha^T K - y||_2^2 + \lambda\alpha^T K\alpha$, $\alpha^* = (K + \lambda I)^{-1}y$
Prediction: $y = w^{*T}x = \Sigma_{i=1}^n \alpha_i^* k(x_i, x)$
Kernelized Support Vector Machine
SVM: $k_i = [y_1 k(x_i, x_1), \ldots, y_n k(x_i, x_n)]$
$\min_\alpha \sum_{i=1}^n \max\{0, 1 - y_i\alpha^T k_i\} + \lambda\alpha^T K\alpha$
Prediction: $y = \mathrm{sign}\left(\sum_{j=1}^n \alpha_j y_j k(x_j, x)\right)$
k Nearest Neighbor classifier
No training, classification during test time:
$y = \mathrm{sign}(\sum_{i=1}^n y_i[x_i \text{ among k nn of } x])$

Neural Networks - U/S

$f(x) = \Sigma_{i=1}^k w_i^{(2)} \phi(\Sigma_{j=1}^m w_{ij}^{(1)} x_j) = W^{(2)}\phi(W^{(1)}x)$
$f(x) = \phi^{(L)}(W^{(L)}\phi^{(L-1)}(W^{(L-1)}\ldots(\phi^{(1)}(W^{(1)}x))))$
Learning features (loss non convex)
Parametr. feat. maps & optimize over params:
$w^* = \mathrm{argmin}_{\mathrm{w},\theta} \sum_{i=1}^n l(y_i; f(x))$
One possibility: $\phi(x,\theta) = \varphi(\theta^T x) = \varphi(z)$
Predict $y_j = f_j$ reg. / $y_j = \mathrm{sign}(f_j)$ class.
Backpropagation
Error_{out}: $\delta^{(L)} = l'(f) = [l'(f_1), \ldots, l'(f_p)]$
Grad_{out}: $\nabla_{W^{(L)}} \ell(W; y, x) = \delta^{(L)} v^{(L-1)T}$
Error_{hidden}: $\delta^{(\ell)} = \phi'\left(z^{(\ell)}\right) \odot W^{(\ell+1)T}\delta^{(\ell+1)}$
Grad_{hidden}: $\nabla_{W^{(\ell)}} \ell(W; y, x) = \delta^{(\ell)} v^{(\ell-1)T}$
Activation functions
Sigmoid: $\varphi(z) = \frac{1}{1+exp(-z)}$; $\varphi' = (1 - \varphi(z)) \cdot \varphi(z)$
$\mathrm{Tanh}_{[-1,1]}$: $\varphi(z) = tanh(z) = \frac{exp(z) - exp(-z)}{exp(z) + exp(-z)}$
Initialization (matters) and Var propagation
Vanish-/explod- gradients: keep $\|v_i\|$ bounded
Dropout: neuron pres $p \to \mathbb{E}[I_i] = \mathbb{E}[I_i^2] = p$
$w_i \sim \mathcal{N}(0, \sigma^2) z = \sum_1^d w_i I_i x_i, v = \phi(z), \mathbb{E}[x_i^n] = 1$
$\mathbb{E}[z] = 0, \mathrm{Var}(z) = \sum_1^d \mathbb{E}[x_i^2]\,\mathrm{Var}(w_i), \mathbb{E}[v^2] \overset{!}{=} 1$
ReLu: $\sigma^2 = \frac{2}{n_{in}}$, Tanh: $\sigma^2 = \frac{1}{n_{in}} = \frac{2}{n_{in}+n_{out}}$
Regularization
Ao: monitor validation set, early stopping; dropout: avoid hidden units memorize training samples; weight decay; batchnorm.
Batchnorm: ↓ internal cov. shift, enable ↑ η_t: 1. mini μ_S, σ_S^2, 2. normalize, 3. scale, shift.
CNN $z_i = \sum_{j=\max(1,i-d+1)}^{\min(i,k)} w_j x_{i-j+1} \mathbb{R}^{d+k-1}$
Filter output size: $l_{out} = \frac{w_{in}+2p-f}{s} + 1$
CNN #fts$\#chan_{in} \prod \ker_i \#chan_{out} + \#chan_{out}$
ANN $\#chan_{in} I_{in}^2 \#chan_{out} I_{out}^2 + \#chan_{out} I_{out}^2$
ResN.: skip lay., ↑ deep NN, ↓ vanishing grad.

Clustering - U

k- means
$\hat{\mu} = \arg\min_\mu \hat{R}(\mu) = \sum_{i=1}^n \min_{j\in\{1,..k\}} ||x_i - \mu_j||_2^2$
Non-convex, NP-hard, can be kernelized
Lloyd's heuristic ($\mathcal{O}(nkd), d = dim, n = \#pt$)
Initialize cluster centers $\mu^{(0)} = [\mu_1^{(0)}, ..., \mu_k^{(0)}]$
While still changes in assignments:
$z_i^{(t)} = \min_{j\in\{1..k\}} ||x_i - \mu_j^{(t-1)}||_2^2$; $\mu_j^{(t)} = \frac{1}{n_j}\sum_{i:z_i^{(t)}=j} x_i$

k-mean++ ($\mathcal{O}(\log k)$)
Random $\mu_1^{(0)} := x_i$ for $i \sim \mathrm{Uniform}(\{1 \ldots n\})$
Add centers 2 to k randomly, proportionally to squared distance to closest selected center for $j = 2$ to k: i_j sampled with prob.
$\mu_j^{(0)} \leftarrow x_i$ for $\mathrm{Prob}(i) \propto \min_{l\in\{1\ldots j-1\}} \left\|x_i - \mu_l^{(0)}\right\|_2^2$

Model selection (separated sphere clusters)
Regularization (favor simple model with few param.); information criterium; elbow method

Dimension Reduction - U

PCA ($\approx$ **k-m. but** W **orth.,** $z_i \notin E_k = e_1 \ldots e_k$)
Given: $D = \{x_1, ..., x_n\} \subset \mathbb{R}^d$, $1 \leq k \leq d$
$\Sigma_{d\times d} = \frac{1}{n}\sum_{i=1}^n x_i x_i^T$, $\mu = \frac{1}{n}\sum_{i=1}^n x_i = 0$!!
$(W, z_1, ..., z_n) = \mathrm{argmin} \sum_{i=1}^n ||Wz_i - x_i||_2^2$,
$W \in \mathbb{R}^{d\times k}$ is orthogonal, $W^* = (v_1|...|v_k)$ w/ v_i evec. of Σ and evals $\lambda_1 \geq ... \geq \lambda_d \geq 0$.
Projections $z_1, ..., z_n \in \mathbb{R}^k$ are given by
$z_i = W^T x_i$ with $\Sigma = \sum_{i=1}^d \lambda_i v_i v_i^T, W^\top W = \mathbb{I}$
Kernel PCA
$k \geq 1$, KPCA given by $\alpha^{(1)}, ..., \alpha^{(k)} \in \mathbb{R}^n$, $\alpha^{(i)} = \frac{1}{\sqrt{\lambda_i}} v_i$ from: $K^{n\times n} = \sum_{i=1}^n \lambda_i v_i v_i^T$
Point x proj. as $z \in \mathbb{R}^k$: $z_i = \sum_{j=1}^n \alpha_j^{(i)} k(x, x_j)$
Autoencoders $f_{en} : \mathbb{R}^d \to \mathbb{R}^k, f_{de} : \mathbb{R}^k \to \mathbb{R}^d$
Learn id. fc: $x \approx f(x;\theta)$, if $\varphi(z) = z$, $AE = PCA$
$f(x;\theta) = f_{de}(f_{en}(x;\theta_{en});\theta_{de}), W^{(en/de)} = f_{en/de}$
$W^* = \mathrm{argmin}_{\mathrm{w}} \sum_{i=1}^n ||x_i - f(x;\theta)||_2^2$

Probability Modeling

Risk estimation (iid)
Estim. risk (unfeasable): $\tilde{R}(\hat{f}) = \mathbb{E}_x[l(\hat{f}, f^*)]$
Gen risk: $R(\hat{f}) = \mathbb{E}_{x,y\sim p}[l(\hat{f}, y)] = \tilde{R}(\hat{f}) + \epsilon_{\mathscr{N}}^2$
$EMSE \propto Bias^2 + Variance + Noise$
$\mathbb{E}_{D,\boldsymbol{x},y}[(y - \hat{f}_D(\boldsymbol{x}))^2] = \mathbb{E}_{\boldsymbol{x}}[\mathbb{E}_D[\hat{f}_D(\boldsymbol{x})] - f^*(\boldsymbol{x})]^2 + \mathbb{E}_{\boldsymbol{x}}[\mathrm{Var}_D[\hat{f}_D(\boldsymbol{x})]] + \mathbb{E}_{\boldsymbol{x},y}[y - f^*(\boldsymbol{x})]^2$
Empirical risk: $\hat{R}_D(w) = \frac{1}{|D|}\sum_{(x,y)\in D} l(\hat{f}, y)$
$\mathbb{E}_{D_{val}}\left[\hat{R}_{D_{val}}\left(\hat{f}_{D_{train}}\right)\right] = R\left(\hat{f}_{D_{train}}\right)$
Risk minimization ($f^* = \min_f \mathbb{E}_{x,y\sim p}[l(f, y)]$)
Reg: $f^* = \mathbb{E}_x\left[\min_f \mathbb{E}_y\left[(y - f(x))^2 \mid X = x\right]\right]$
$\int \partial_{\hat{y}}(\hat{y} - y)^2 p(y \mid x) dy = 0$, $f^* = \mathbb{E}[y \mid X = x]$
Predict via: $\hat{y} = \hat{E}[Y \mid X = x] = \int y\hat{p}(y \mid x) dy$

Clas: $f^* = \mathbb{E}_x\left[\min_{\hat{y}} \mathbb{E}_y\left[[y \neq \hat{y}] \mid X = x\right]\right] \rightarrow$
$\sum p(y \mid \cdot)[y \neq \hat{y}] = \sum_{y \neq \hat{y}} p(y \mid \cdot) \rightarrow \max_{\hat{y}} p(\hat{y} \mid x)$

MLE (min var among all unbiased estim.)

Solve $\theta^\star = \text{argmax}_\theta \prod_{i=1}^n \hat{p}(y_i|x_i, \theta)$

$p_{OLS}(y \mid x, w) = \mathcal{N}(w^\top x, \sigma^2)$

$p_C(y \mid x, w) = \text{Ber}(y; \sigma(w^T x)) = \frac{1}{1+exp(-ywTx)}$

MAP (bias through $p(\theta|x,y) = \frac{p(\theta)p(y|x,\theta)}{p(y|x)}$**)**

Gauss prior: $p(\theta \mid \mu, \beta) = \prod_1^d \frac{1}{\beta\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\theta_i - \mu}{\beta}\right)^2}$

Laplace prior: $p(\theta \mid \mu, b) = \prod_i^d \frac{1}{2b} e^{-\frac{|\theta_i - \mu|}{b}}$

Ridge: $p_{gauss}(\theta \mid 0, \beta)$, $\lambda = \frac{\sigma^2}{\beta^2}$, $\|\theta\|_2^2 = \sum_i^d \theta_i^2$

Lasso: $p_{laplace}(\theta \mid 0, b)$, $\lambda = \frac{2\sigma^2}{b}$, $\|\theta\|_1 = \sum |\theta_i|$

MLE for Classification ($\hat{w}$**)**

$\max_w p_C(y_{1:n}|x_{1:n}, w) = \min - \sum \log p(y_i|x_i, w)$
$= \text{argmin}_w \sum_{i=1}^n log(1 + exp(-y_i w^T x_i))$

GD for logistic regression

$p(\text{missclas.}) = \hat{p}(Y = -y|w, x) = \frac{1}{1+exp(ywTx)}$

Update $w \leftarrow w + \eta_t yx\hat{p}(Y = +y|w, x)$

Ridge: $w \leftarrow w(1 - 2\lambda\eta_t) + \eta_t yx\hat{p}(Y = +y|w, x)$

Multiclass Logistic Reg. (unique if $w_c = 0$**)**

Estimate $f_{1-c}(x) = w_1^\top x \ldots, w_c^\top x$ for logisitc

Predict $p(y \mid x) = \text{Cat}(y \mid \text{softmax}(f_{1:c}(x)))$

$= \frac{\exp(w_y^T x)}{\sum_i^c \exp(w_j^T x)} = \frac{1}{Z}\exp(w_y^T x)$ for label $Y = y$

Overflow problem: let $s_i = w_i^\top x$, choose $s_{i^*} = \max_{i \in K} s_i$, subtract s_{i^*} from other weights.

CE: $l(y, x; w_1..w_c) = -\log p(Y = y \mid x, w_1...w_c)$

Kernelized Logistic Regression

$\hat{\alpha} = \min_\alpha \sum_1^n \log\left(1 + e^{-y_i \alpha^T K_i}\right) + \lambda \alpha^T K \alpha$

$\hat{p}(y \mid x, \widehat{\alpha}) = \frac{1}{1+\exp\left(-y\sum_{j=1}^n \alpha_j k(x_j, \boldsymbol{x})\right)}$

Bayesian decision theory

- Conditional distribution over labels $p(y|x)$
- Set of actions A
- Cost function $C : Y \times \mathscr{A} \rightarrow \mathbb{R}$

$a^* = \min_{a \in A} \mathbb{E}_y[C(y, a)|x] = \sum_y p(y|x)C(y, a)$

Symm. cost class. $C(y, a) = [y \neq a], A = \{\pm 1\}$

$a^* = \min_{a \in A} \sum_y p_C(y \mid x)C(y, a) = \text{sign}(w^T x)$

Asymmetric cost classification $A = \{\pm 1\}$

$$C(y, a) = \begin{cases} c_{fp}\ , \text{if } y = -1 \text{ and } a = +1 \\ c_{fn}\ , \text{if } y = +1 \text{ and } a = -1 \\ 0\ , \text{otherwise} \end{cases}$$

$C_+ = \mathbb{E}_y[C(y, 1)|x] = p_C(1|x) \cdot 0 + p_C(-1|x)c_{fp}$

$C_- = \mathbb{E}[C(y, -1)|x] = p_C(1|x)c_{fn} + p_C(-1|x) \cdot 0$

Pred. $+1$ if $C_+ \leq C_- \rightarrow p(y = 1|x) \geq \frac{c_{fp}}{c_{fp}+c_{fn}}$

Doubtful logistic class. $A = \{+1, -1, D\}$

$$C(y, a) = \begin{cases} c_1 \cdot [y \neq a] & \text{if } a \in \{+1, -1\} \\ c_2 & \text{if } a = D \end{cases}$$

$a^* = y$ if $p_C(y|x) \geq 1 - c_2/c_1$, D otherwise

Linear reg. $p_{OLS}(y \mid x), C(y, a) = (y - a)^2$

$A = \mathbb{R}, a^* = \min_a \int C(y, a)p_{OLS}(y|x)dy = w^T x$

Asymm. cost reg. $\Phi(z) = \int_{-\infty}^z \mathcal{N}(z; 0, 1)dz$

$C(y, a) = c_1 \max(y - a, 0) + c_2 \max(a - y, 0)$

$A = \mathbb{R} \rightarrow a^* = w^T x + \sigma\Phi^{-1}(\frac{c_1}{c_1+c_2})$

Active learning (uncertainty sampling, no iid)

Entropy: $\text{H}(X) = -\sum_{i=1}^n p(x_i) \log p(x_i)$

Given $D = D^L \cup D^U$, initially $D^L = \varnothing$.

- Estimate $p(y \mid x)$ given current D^L
- Pick unlabeled $x_{i_t}^U$, $i_t \in \min_{x \in D^U} H(p(y \mid x))$
- Query label y_{i_t} and set $D_L \leftarrow D_L \cup \{(x_{i_t}, y_{i_t})\}$

Discriminative vs. Generative Modeling

D: directly assume $p(y \mid x)$ and estimate parameters from training dataset. $\approx$ always S. SVM, NN, KNN, Rand. Forest, Log. regression $\oplus$ robust, $\oplus$ $\mathscr{O}(cheaper)$ $\oplus$ less overfit

G: estimate $p(y)$ and $p(x \mid y)$ from training data and compute posterior with Bayes's rule:

$$p(y|x) = \frac{p(y)p(x|y)}{p(x)},\ p(x) = \sum_y p(x, y)$$

Models: LDA, Naive Bayes, GMM, GANS

$\oplus$ outlier detection, $\oplus$ generate new data,

$\ominus$ robust to outliers $\ominus$ model x may difficult

Outlier detect $P(x) = \sum_{y=1}^c P(y)P(x|y) \leq \tau$

MLE for Categorical$(y \mid \boldsymbol{\theta}), \boldsymbol{\theta} = \{p_1, p_0\}$

Let $p(y = 1) = p_1, p(y = 0) = p_0 = 1 - p_1$

$L(y \mid \boldsymbol{\theta}) = \prod_{i=1}^n p_1^{[y_i=1]} p_0^{[y_i=0]}(1 - p_1 - p_0)^{1-[y_i=1]-[y_i=0]} = p_1^{n_1} p_0^{n_0}(1 - p_1 - p_0)^{1-n_1-n_0}$

$\nabla_{\boldsymbol{\theta}} \log L(y \mid \boldsymbol{\theta}) \stackrel{!}{=} 0 \Rightarrow p_1 = \frac{n_1}{n_1+n_0}, p_0 = \frac{n_0}{n_1+n_0}$

Gaussian Bayes Classifier (or QDA) - S

MLE class prior: $P(Y = y) = \hat{p}_y = \frac{\text{Count}(Y=y)}{n}$

MLE feature distr: $P(x|y) = \mathcal{N}(x; \hat{\mu}_y, \hat{\Sigma}_y)$

$\hat{\mu}_y = \frac{1}{\text{Count}(Y=y)} \sum_{i:y_i=y} x_i \in \mathbb{R}^d$

$\hat{\Sigma}_y = \frac{1}{\text{Count}(Y=y)} \sum_{i:y_i=y} (x_i - \hat{\mu}_y)(x_i - \hat{\mu}_y)^T \mathbb{R}^{d \times d}$

Naive: iid, $\Sigma_y = \text{diag}(\sigma_1^2 ... \sigma_d^2)$, predict new x:

$y = \max_{y'} P(y'|x) = \max_{y'} P(y') \prod_{i=1}^d P(x_i|y')$

Decision rule binary classif. $c = 2$, $y \in \{\pm 1\}$

$y = \max_{y'} P(y'|x) = \text{sign}(f(x))$, $p_+(x) = \sigma(f(x))$

Discriminant fnc: $f(x) = \log \frac{P(y=1|x)}{P(y=-1|x)} =$

$\log \frac{p_+(x)}{1-p_+(x)} = \log \frac{p}{1-p} + \frac{1}{2}[\log \frac{|\hat{\Sigma}_-|}{|\hat{\Sigma}_+|} + ((\mathbf{x} - \hat{\mu}_-)^T \hat{\Sigma}_-^{-1}(\mathbf{x} - \hat{\mu}_-)) - ((\mathbf{x} - \hat{\mu}_+)^T \hat{\Sigma}_+^{-1}(\mathbf{x} - \hat{\mu}_+))]$

Naive: $p(x \mid y) = \prod_i \mathcal{N}(x_i; \mu_{y,i}, \sigma_{y,i}^2)$

If shared $\sigma_{y,i}^2$: $f(x) = w^\top x + w_0 \rightarrow$ log. regress.

$w_0 = \log \frac{\hat{p}_+}{1-\hat{p}_+} + \sum_{i=1}^d \frac{\hat{\mu}_{-,i}^2 - \hat{\mu}_{+,i}^2}{2\hat{\sigma}_i^2}, w_i = \frac{\mu_{+,i} - \mu_{-,i}}{\sigma_i^2}$

Overcounting: if $x_2 = \ldots = x_d$ then decision plane stop sum w_0/w_i at 2.

Overconfident: conditional independence $\rightarrow$ prediction close to 1 or 0. Fine to pred. most likely class, $\ominus$ making decision (asymm. loss)

Linear discrim. analysis (LDA, $\Sigma_1 = \ldots = \Sigma_c$**)**

Fisher LDA: $c = 2$, $p = 0.5$; $\hat{\Sigma}_- = \hat{\Sigma}_+ = \hat{\Sigma}$

Predict: $y = \text{sign}(f(x)) = \text{sign}(w^T x + w_0)$

$w = \hat{\Sigma}^{-1}(\hat{\mu}_+ - \hat{\mu}_-)$; $w_0 = \frac{1}{2}(\hat{\mu}_-^T \hat{\Sigma}^{-1} \hat{\mu}_- - \hat{\mu}_+^T \hat{\Sigma}^{-1} \hat{\mu}_+)$

Categorical Naive Bayes Classifier

MLE class prior: $P(Y = y) = \hat{p}_y = \frac{\text{Count(Y=y)}}{n}$

MLE feature distr: $P(X_i = x \mid Y = y) = \theta_{x|y}^{(i)}$

$\hat{\theta}_{x|y}^{(i)} = \frac{\text{Count}(X_i=x, Y=y)}{\text{Count}(Y=y)}$. Prediction as GBC

Avoid overfitting

Restrict model class (es only diag cov $\rightarrow$ few parameters), priors $\rightarrow$ smaller param. values

MAP in Generative Modelling

Use MAP to estimate class prior/feature distrb. (≈ 0 comp. cost). Prior to Bernoulli/Bin:

$\text{Beta}(\theta; \alpha_+, \alpha_-) = \frac{1}{B(\alpha_+, \alpha_-)} \theta^{\alpha_+ - 1}(1 - \theta)^{\alpha_- - 1}$

$$MLE_{Beta} = \frac{\alpha_+ - 1}{\alpha_+ + \alpha_- - 2}$$

Categorical/Multinomial (l) $\rightarrow$ Dirichlet (prior)

Gauss. (l) $\rightarrow$ Gauss. /-inverse Wishart (prior)

Gaussian Mixture Models (GMM) - SSL/U

Sel. k via CV or D_{train}/D_{val}, maxim. log-like.

Can choose τ to control estimated FPR

Mixture modeling $\theta = (\mu, \Sigma, w)$ **(iid)**

Convex comb: $P(\mathbf{x} \mid \theta) = \sum_1^k w_j \mathcal{N}(\mathbf{x}; \mu_j, \Sigma_j)$

Minimize: $\theta^* = \min_\theta - \sum_i^n \log \sum_j^k w_j P(x_i|\theta_j)$

Non feasable $\rightarrow$ non-convex, alternatively:

$P(\mathbf{x}|\theta) = \sum_z P(\mathbf{x}, z|\theta) = \sum_z P(z|\theta)P(\mathbf{x}|z, \theta)$

and $P(z_i = j|\theta) = w_j, P(\mathbf{x}, z|\theta) = w_z \mathcal{N}(\mathbf{x}|\mu_z, \Sigma_z)$

Fitting GMM = training GBC without labels

Hard-EM algorithm (most probable z_i**)**

Predict class z_i for each x_i:

E:
$$\begin{aligned} z_i^{(t)} &= \text{argmax}_z\, P(z \mid x_i, \theta^{(t-1)}) \\ &= \text{argmax}_z\, P(z|\theta^{(t-1)})P(x_i|z, \theta^{(t-1)}) \\ &= \text{argmax}_z\, w_z^{(t-1)} \mathcal{N}(x_i|\mu_z^{(t-1)}, \Sigma_z^{(t-1)}) \end{aligned}$$

Now complete: $D^{(t)} = \{(x_1, z_1^{(t)}) ... (x_n, z_n^{(t)})\}$

M: MLE as in GBC $\theta^{(t)} = \text{argmax}_\theta\, P(D^{(t)}|\theta)$

Issues: points have label, $\ominus$ if cluster overlap

Lloyd = H-EM $w_z = \frac{1}{k}, \Sigma_z = \sigma^2 \mathbb{I}$, bal class

$\rightarrow$ **E:** $z_i^{(t)} = \text{argmin}_z\, ||x_i - \mu_z^{(t-1)}||_2^2$

$\rightarrow$ **M:** $\mu_j^{(t)} = \frac{1}{n_j} \Sigma_{i:z_i^{(t)}=j} x_i$

Lloyd h. = S-EM as above and $\sigma^2 \rightarrow 0$

Soft-EM algorithm (weighting average)

($\equiv$ training a GBC with weighted data!)

Let iid. $P(\mathbf{x}_{1:n}, \mathbf{z}_{1:n}|\theta) = \prod_{i=1}^n P(x_i, z_i|\theta)$

E-step: calculate expected complete log-like:

$Q(\theta; \theta^{(t-1)}) = \mathbb{E}_{\mathbf{z}_{1:n}}[\log P(\mathbf{x}_{1:n}, \mathbf{z}_{1:n}|\theta)\,|\mathbf{x}_{1:n}, \theta^{(t-1)}]$

$= \sum_i^n \sum_j^k \underbrace{P(z_i = j|x_i, \theta^{(t-1)})}_{\gamma_{z_i}^{(t)}(x_i)\text{: ex. suf. stat.}} \log \underbrace{(P(x_i, z_i = j|\theta))}_{w_j \mathcal{N}(x_i|\mu_j, \Sigma_j)}$

where (Bayes) $\gamma_j^{(t)}(x_i) = \frac{w_j P(x_i|\Sigma_j, \mu_j)}{\sum_{l=1}^k w_l P(x_i|\Sigma_l, \mu_l)}$

M-step: calculate $\theta^{(t)} = \text{argmax}_\theta\, Q(\theta; \theta^{(t-1)})$ subject to $\sum_{j=1}^k w_j = 1 \rightarrow$ Lagrange multipliers

$\theta^{(t)} = \min_\theta \mathscr{L}(\theta, \lambda) = Q(\theta; \theta^{(t-1)}) + \lambda(\Sigma w_k - 1)$

$w_j^{(t)} \leftarrow \frac{1}{n}\sum_{i=1}^n \gamma_j^{(t)}(x_i)$; $\mu_j^{(t)} \leftarrow \frac{\sum_{i=1}^n \gamma_j^{(t)}(x_i) x_i}{\sum_{i=1}^n \gamma_j^{(t)}(x_i)}$

$\Sigma_j^{(t)} \leftarrow \frac{\sum_{i=1}^n \gamma_j^{(t)}(x_i)(x_i - \mu_j^{(t)})(x_i - \mu_j^{(t)})^T}{\sum_{i=1}^n \gamma_j^{(t)}(x_i)} + \nu^2 \mathbb{I}$

Avoid degen, calculated via MAP, Wishart prior

Convergence of the Soft-EM algorithm

S-EM monotonically increases likelihood: $\log P\left(\mathbf{x}_{1:n} \mid \theta^{(t)}\right) \geq \log P\left(\mathbf{x}_{1:n} \mid \theta^{(t-1)}\right)$

For Gaussian mixtures, guarranteed to conv. local maxim. Quality solution $\leftarrow$ initialization! Re-run multiple times, chose sol biggest likeh.

Semi-S case: $\{(x_1, z_1) ... (x_l, z_l)\}$ **labels given**

For $i > l \rightarrow \gamma_j^{(t)}(\mathbf{x}_i, y_i)$ (usual).

For $i \leq l \rightarrow \gamma_j^{(t)}(\mathbf{x}_i, y_i) = [j = z_i]$

Gaussian-Mixture Bayes classifiers

Labeled $D = \{(x_i, y_i)\}, i = 1...n$, m labels

Estimate class prior $P(y)$

$P(x|y) = \sum_{j=1}^{k_y} w_j^{(y)} \mathcal{N}(x; \mu_j^{(y)}, \Sigma_j^{(y)})$

$P(y|x) = \frac{1}{P(x)} p(y) \sum_{j=1}^{k_y} w_j^{(y)} \mathcal{N}(x; \mu_j^{(y)}, \Sigma_j^{(y)})$

Generative adversarial networks (GANS)

Given sample of unlabeled points $\mathbf{x}_1 ... \mathbf{x}_n$

Goal: learn model $\mathbf{X} = G(\mathbf{Z}; \mathbf{w})$ with Z simple distrib. (low dim Gauss.), $G = NN$

Likelihood hard, optimize w to make sample difficult to distinguish from data sample.

Training as a "game" betw. 2 NN

1. Gen. G tries produce $G : z \mapsto G(z; w_G)$
2. Discr. D find fake $D : x \mapsto D(x; w_D) \in [0, 1]$

$$D(x) = D(x, w_D) = \begin{cases} \approx 1 \text{ if } x \text{ real} \\ \approx 0 \text{ if } x \text{ fake} \end{cases}$$

If G, D enough capacity, guaranteed to conv. to

Saddle: $\min_{w_G} \max_{w_D} M(w_G, w_D)$, $M(w_G, w_D) =$

$\underbrace{\mathbb{E}_{\mathbf{x} \sim \text{dat}} \log D(x; w_D)}_{\text{real images}} + \underbrace{\mathbb{E}_{\mathbf{z} \sim \mathcal{N}} \log[1\text{-}D(G(z; w_G); w_D)]}_{\text{fake images}}$

Simultaneous gradient descent:

$$w_G^{(t+1)} = w_G^{(t)} - \eta_t \nabla_{w_G} M(w_G, w_D^{(t)})$$
$$w_D^{(t+1)} = w_D^{(t)} + \eta_t \nabla_{w_D} M(w_G^{(t)}, w_D)$$

Optimal discriminator

For a fixed generator $G(z; w_G) = x$ the optimal discriminator $D(x)$ is:

$$P(x \text{ from data}) = D_G^*(x) = \frac{p_{data}(x)}{p_G(x) + p_{data}(x)}$$

$R(G) = \max_D M(G, D) = -\log 4 + 2JSD\left(p_{dat} \| p_G\right)$

Duality Gap

$DG(w_G, w_D) = \max_{w_D'} M(w_G, w_D') - \min_{w_G'} M(w_G', w_D)$

$DG = 0$ w_G, w_D pure equilibrium, else $DG > 0$

If G and D sufficient capacity, DG upper-bounds the JSD betw. data distrib. and gen.