Regression

Convex

$g(x)$ is convex $\Leftrightarrow x_1, x_2 \in \mathbb{R}, \lambda \in [0,1] : g''(x) > 0$

$g(\lambda x_1 + (1-\lambda)x_2) \le \lambda g(x_1) + (1-\lambda) g(x_2)$

Jensen: $g(\mathbb{E}[X]) \le \mathbb{E}[g(x)]$

Gaussian/Multivariate Normal

Ass: μ = mean, σ = std., σ^2 = var., Σ = covar.:

$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(x-\mu)^2}{2\sigma^2})$

$f(x) = ((2\pi)^d |\Sigma|)^{-1/2} \exp(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu))$

Standardization

$\forall$ features: $\mu = 0, \sigma^2 = 1$: $\tilde{x}_{i,j} = \frac{(x_{i,j} - \hat\mu_j)}{\hat\sigma_j}$

$\hat\mu_j = \frac{1}{n}\sum_{i=1}^n x_{i,j}$, $\hat\sigma_j^2 = \frac{1}{n}\sum_{i=1}^n (x_{i,j}-\hat\mu_j)^2$

Generalization Error

Ass: data generated iid: (expected / estim)

$R(w) = \int P(x,y)(y-w^Tx)^2 dxdy = \mathbb{E}_{x,y}[(y-w^Tx)^2]$

$\hat R_D(w) = \frac{1}{|D|}\sum_{(x,y)\in D}(y-w^Tx)^2$

Linear Regression

Optim: $w^* = \arg\min_w \hat R(w)$; $y = Xw$

Error: $\hat R(w) = \sum_{i=1}^n (y_i - w^T x_i)^2 = \|Xw - y\|_2^2$

Closed form: $w^* = (X^TX)^{-1}X^Ty$

$\nabla_w \hat R(w) = -2\sum_{i=1}^n (y_i - w^Tx_i)\cdot x_i = 2X^T(Xw-y)$

L2: Ridge Regression

Regularization λ: $\min_w \hat L(w) + \lambda C(w)$

Optim: $\hat R(w) = \|Xw-y\|_2^2 + \lambda\|w\|_2^2$

Closed form: $w^* = (X^TX + \lambda I)^{-1}X^Ty$

$\nabla_w \hat R(w) = 2X^T(Xw-y) + 2\lambda w$

Gradient Descent

1. Start arbitrary $w_o \in \mathbb{R}$
2. For $t = 1,2,\ldots$ do $w_{t+1} = w_t - \eta_t \nabla \hat R(w_t)$

Complexity: $\mathcal{O}(nd)$

Stochastic Gradient Descent (SGD)

1. Start at an arbitrary $w_0 \in \mathbb{R}^d$
2. For $t=1,2,\ldots$ do:

Pick data point $(x',y') \in_{\text{unif.a.r.}} D$

$w_{t+1} = w_t - \eta_t \nabla_w l(w_t; x', y')$

Complexity: $\mathcal{O}(dT)$

Classification

0/1 loss

$l_{0/1}$ is not convex, not differentiable.

$l_{0/1}(w;y_i,x_i) = \begin{cases} 1 \text{ , if } y_i \ne sign(w^Tx_i) \\ 0 \text{ , otherwise}\end{cases}$

Perceptron loss

l_P is convex, not differentiable, gradient inform.

$l_P(w;y_i,x_i) = \max\{0, -y_i w^T x_i\}$

$\nabla_w l_P = \begin{cases} 0 & \text{, if } y_i w^T x_i \ge 0 \\ -y_i x_i & \text{, if } y_iw^Tx_i < 0\end{cases}$

Hinge loss

l_H upper bounds #mistakes, encourages margin

$l_H(w;x,y) = \max\{0, 1 - y_i w^T x_i\}$

$\nabla_w l_H = \begin{cases} -y_ix_i & \text{, if } y_iw^Tx_i<1 \\ 0 & \text{, if } y_iw^Tx_i \ge 1\end{cases}$

SVM - Max Margin

Goal: max the margin around the separator with l_H

Optim: $\hat R(w) = \max\{0, 1 - y^T X w\} + \lambda\|w\|_2^2$

$\nabla_w \hat R(w) = \begin{cases} -X^Ty + 2\lambda w & \text{, if } y_iw^Tx_i<1 \\ 2\lambda w & \text{, if } y_i w^T x_i \ge 1 \end{cases}$

L1: $\|w\|_1$ sends coeff to be zero (only lin. models)

Matrix-Vector Gradient

$\beta \in \mathbb{R}^d$: $\nabla_\beta(\|y - X\beta\|_2^2 + \lambda\|\beta\|_2^2) = 2X^T(y-X\beta) + 2\lambda\beta$

Kernels

Properties of a Kernel

k mb function: $f: X\times X \to R$

k mb symmetric: $k(x,y) = k(y,x)$

k mb inner product: $k(x,y) = \langle \phi(x)^T, \phi(y)\rangle$

Matrix K must be positive semi-definite (psd).

$$K = \begin{bmatrix} k(x_1,x_1) & \ldots & k(x_1,x_n) \\ \vdots & \ddots & \vdots \\ k(x_n,x_1) & \ldots & k(x_n,x_n)\end{bmatrix}$$

positive semi-definite matrices $\Leftrightarrow$ kernels

Definition of PSD

$M \in \mathbb{R}^{n\times n}$ is psd $\Leftrightarrow$

$\forall \alpha \in \mathbb{R}^n: \alpha^T M \alpha \ge 0 \Leftrightarrow \sum_i\sum_j \alpha_i\alpha_j k(x_i,x_j) \ge 0 \Leftrightarrow$

all eigenvalues of M are positive: $\lambda_i \ge 0$

Examples of kernels on $\mathbb{R}^d$

Linear: $k(x,y) = x^Ty$; Constant: $k(x,y) = c, c>0$

Monomial: $k(x,y) = (x^Ty)^d$

Polynomial: $k(x,y) = (x^Ty+1)^d$

Gaussian: $k(x,y) = exp(-\|x-y\|_2^2/h^2)$

Laplacian: $k(x,y) = exp(-\|x-y\|_1/h)$

h = bandwidth $\approx 1\sigma$

Kernel composition

$k_1(x,y) + k_2(x,y)$; $k_1(x,y)\cdot k_2(x,y)$; $c\cdot k_1(x,y), c>0$;

$f(k_1(x,y))$, where f is a polynomial with pos. coeffs. or the exponential function

Parametric vs. Nonparametric

Parametric: have finite set of parameters

$f(x) = w^Tx, w\in\mathbb{R}^d$ (d is independent of # data)

Nonparametric: grow in complexity with size of data

$f(x) = \sum_{i=1}^n \alpha_i y_i k(x_i, x_n)$ (depends on # data)

Kernelized perceptron

Trick: $x^Ty \mapsto \phi(x)^T\phi(y) =: k(x,y)$ s.t. $\exists\phi: X \to \mathbb{R}^d$

Ass: $w \in \text{span}(X) \to w = \sum_{j=1}^n \alpha_j y_j x_j$

Kernel: $k_i = [y_1 k(x_i,x_1), \ldots, y_n k(x_i,x_n)]^T$:

Optim: $\hat R(w) = \min_{w\in\mathbb{R}^d}\sum_i^n \max\{0, -y_i w^T x_i\}$

$\hat R(\alpha) = \min_{\alpha_{1:n}} \sum_{i=1}^n \max\{0, -\sum_{j=1}^n \alpha_j y_i y_j x_i^T x_j\}$

Perceptron: $\min_\alpha \sum_{i=1}^n \max\{0, -y_i\alpha^T k_i\}$

SGD Updates

1. Initialize $\alpha_{1:n} = 0$
2. For $t = 1,2,\ldots$ do: $(x_i, y_i) \in_{u.a.r} D$

Check: $\hat y = sign(\sum_{j=1}^n \alpha_j y_j k(x_j, x_i))$

Update: If $\hat y \ne y_i$ set $\alpha_i = \alpha_i + \eta_t$

Predict new x: $\hat y = \text{sign}(\sum_{j=1}^n \alpha_j y_j k(x_j, x))$

Kernelized SVM

SVM: $\min_\alpha \sum_{i=1}^n \max\{0, 1 - y_i\alpha^T k_i\} + \lambda\alpha^T D_y K D_y \alpha$

Prediction: $y = sign(\sum_{j=1}^n \alpha_j y_j k(x_j, x))$

Kernelized linear regression + ridge

Ass: $w^* = \sum_i \alpha_i x = X^T\alpha$

KLR: $\hat a = \arg\min_\alpha \|\alpha^T K - y\|_2^2 + \lambda\alpha^T K\alpha$

Closed form: $\alpha^* = (K+\lambda I)^{-1} y$

Prediction: $y = w^{*T} x = \sum_{i=1}^n \alpha_i^* k(x_i, x)$

Nearest Neighbor k-NN

$y = sign(\sum_{i=1}^n y_i [x_i \text{ among k nn of } x])$

Imbalance

Cost Sensitive Classification

Replace loss by: $l_{CS}(w;x,y) = c_y l(w;x,y)$

$\hat R(w;c_+,c_-) = \sum_{i:+} c_+ l(w;x_i,y_i) + \sum_{i:-} c_- l(w;x_i,y_i)$

Metrics

Accuracy: $\frac{TP+TN}{TP+TN+FP+FN}$, Precision: $\frac{TP}{TP+FP} = \frac{TP}{p_+}$

TPR (Recall)=: $\frac{TP}{TP+FN} = \frac{TP}{n_+}$, FPR = $\frac{FP}{TN+FP} = \frac{FP}{n_-}$

F1 score: $\frac{2TP}{2TP+FP+FN}$

$\hat y \backslash y$	+	−	
+	TP	FP	p_+
−	FN	TN	p_-
	n_+	n_-	n

Multi-Class Hinge Loss

One vs. One | One vs. All | Maintain $w^{(1)},..,w^{(c)}$

$l_{MC-H}(w^{(1:c)};x,y) = \max(0, 1 + \max_{\forall j\ne y} w^{(j)T}x - w^{(y)T}x)$

Neural Networks

Learning features

Parameterize feature maps, optimize over params

$w^* = \text{argmin}_{w,\theta}\sum_{i=1}^n l(y_i; w\phi(x_i,\theta))$

such that $\phi(x,\theta) = \varphi(\theta^Tx) = \varphi(z)$

Optim: $W^* = \text{argmin}_W \sum_{i=1}^n l(W; y_i, x_i)$

Activation functions

Sigmoid: $\varphi(z) = (1+\exp(-z))^{-1}$, $\varphi'(z) = (1-\varphi(z))\varphi(z)$

Tanh (-1,1): $\varphi(z) = tanh(z) = \frac{exp(z)-exp(-z)}{exp(z)+exp(-z)}$

ReLu: $\varphi(z) = max(z,0)$, $\varphi'(z) = 1$ if $z>0$

Forward propagation

For input layer: $v_j = x_j$ ($v^{(0)} = x$)

For each layer $l = 1 : L-1$:

- For unit j on layer l: $v_j = \varphi(\sum_{i\in(l-1)} w_{j,i} v_i))$

For output layer: $f_j = \sum_{i\in(L-1)} w_{j,i} v_i$

Predict: $y_j = f_j$ for reg. / $y_j = sign(f_j)$ for class

$(z^{(l)} = W^{(l)} v^{(l-1)}; v^{(l)} = \varphi(z^{(l)}; f = W^{(L)} v^{(L-1)})$

Backpropagation

For output layer:

- Error: $\delta_j = \ell_j'(f_j)$
- For each unit i on layer L: $\partial/\partial w_{j,i} = \delta_j v_i$

For hidden layer $l = \{L-1,\ldots,1\}$:

- Error: $\delta_j = \varphi'(z_j)\sum_{i\in(l+1)} w_{i,j}\delta_i$
- For each unit i on layer $l-1$: $\frac{\partial}{\partial w_{j,i}} = \delta_j v_i$

Error: $\delta^{(L)} = l'(f)$; $\delta^{(l)} = \varphi'(z^{(l)}) \odot (W^{(l+1)T}\delta^{(l+1)})$

Gradient: $\nabla_{W^{(l)}} l(W;y,x) = \delta^{(l)} v^{(l-1)T}$

Learning with momentum

$a \leftarrow m\cdot a + \eta_t \nabla_W l(W;y,x)$; $W \leftarrow W - a$

Convolutional

CN $o = (n-f+2p)/s + 1$; #p: $n\cdot n\cdot o\cdot o$

Clustering

k-mean

Optim: $\hat R(\mu) = \hat R(\mu_{1:k}) = \sum_{i=1}^n \min_{j\in\{1,\ldots,k\}} \|x_i - \mu_j\|_2^2$

not convex! → only local optimum!

Algorithm (Lloyd's heuristic):

Initialize cluster centers $\mu^{(0)} = [\mu_1^{(0)},\ldots,\mu_k^{(0)}]$

While not converged

$z_i \leftarrow \arg\min_{j\in\{1\ldots k\}} \|x_i - \mu_j^{(t-1)}\|_2^2$; $\mu_j^{(t)} \leftarrow \frac{1}{n_j}\sum_{i:z_i=j} x_i$

Complexity: $\mathcal{O}(nkd)$ per step

Adaptive seeding k-mean++

- Start with random data point as center
- Add centers randomly, proportionally to squared distance to closest selected center

for $j = 2\ldots k$: i_j sampled with prob.

$P(i_j = i) = \frac{1}{z}\min_{1\le l<j}\|x_i - \mu_l\|_2^2$; $\mu_j \leftarrow x_{i_j}$

Expected cost: $\mathcal{O}(logk) \times$ opt. k-means

Dimension Reduction

Principal component analysis (PCA)

Ass: $\mu = \frac{1}{n}\sum_{i=1}^n x_i = 0$, $\Sigma = \frac{1}{n}\sum_{i=1}^n x_ix_i^T = \frac{1}{n}X^TX$

Optim: $(w,z) = \arg\min_{\|w\|_2=1,z}\sum_{i=1}^n \|wz_i - x_i\|_2^2$

Sol: $w = \arg\max_{\|w\|_2=1} w^T\Sigma w$, $w^* = v_1$, $z_i^* = w^Tx_i$

Optim: $(W, z_{1:n}) = \arg\min\sum_{i=1}^n \|Wz_i - x_i\|_2^2$,

Sol: $W = (v_1|\ldots|v_k) \in \mathbb{R}^{d\times k}$ is orthogonal

$\Sigma = \sum_{i=1}^d \lambda_i v_i v_i^T$ and $z_i = W^Tx_i$

Linear mapping $f(x) = W^Tx$, SVD: $X = USV^T$

Kernel PCA

Ass: $w = \sum_{j=1}^n \alpha_j\phi(x_j)$, $\|w\|_2^2 = \alpha^T K\alpha$

Optim: $\arg\max_{\alpha^TK\alpha=1}\alpha^TK^TK\alpha$

Sol: $\alpha^{(i)} = \frac{1}{\sqrt{\lambda_i}} v_i$, $K = \sum_{i=1}^n \lambda_i v_i v_i^T$

New point x projection as z: $z_i = \sum_{j=1}^n \alpha_j^{(i)} k(x, x_j)$

Autoencoders

Learn identity function: $x \approx f(x;\theta)$

$f(x;\theta) = f_2(f_1(x;\theta_1);\theta_2)$; f_1 : en-, f_2 : de-coder

Optim: $\min_W \sum_{i=1}^n \|x_i - f(x;W)\|_2^2$

Internal representation: $v = \varphi W^{(1)} x$

Probability Modeling

Regression
Ass: (x_i,y_i) iid $\sim P(X,Y)$, Hypothesis: $h: X \to Y$
Min Prediction Error: $R(h) = \mathbb{E}_{x,y}[l(y;h(x))]$
Cond. mean: $h^*(x) = \mathbb{E}[Y|X=x]$ (min for sq. loss)
Estimate: $\hat{P}(Y|X)$, Pred: $\hat{y} = \hat{\mathbb{E}}[Y|X=x]$

Maximum Likelihood Estimation (MLE)
Conditional Likelihood $\hat{P}(Y|X,\theta)$, opt. param. w MLE
$\theta^* = \arg\max_\theta \hat{P}(Y|X,\theta) = \arg\min_\theta -\sum_i^n \log \hat{P}(y_i|x_i,\theta)$

MLE Gaussian
Ass: noise $P(Y=y|X=x,\theta) = \mathcal{N}(y;h(x),\sigma^2)$
MLE: $\hat{h} = \arg\min_{h\in\mathcal{H}} \sum_{i=1}^n (y_i - h(x_i))^2$
Linear: $h(x) = w^T x$, $Y = w^T X + \epsilon, \epsilon \sim \mathcal{N}(0,\sigma^2)$
MLE: $w^* = \arg\min_w \sum_{i=1}^n (y_i - w^T x_i)^2$

Pred.Error = Bias^2 + Variance + Noise
$\mathbb{E}[(Y-\hat{h})^2] = \mathbb{E}[\mathbb{E}[\hat{h}]-h^*]^2 + \mathbb{E}[(\mathbb{E}[\hat{h}]-\hat{h})^2] + \mathbb{E}[Y-h^*]^2$

Maximum a posteriori estimate (MAP)
Prior: $w\sim P(w)$ s.t. $w \perp x$, eg. $w \sim \mathcal{N}(0,\beta^2)$
Posterior: $P(w|x,y) = \frac{P(w)P(y|x,w)}{P(y|x)}$ (Bayes Rule)
MAP: $w^* = \arg\max_w P(w|x,y) =$
$= \arg\min_w -\log P(w) - \log P(y|x,w) + const$

MAP Gaussian
Ass: noise $P(y,x,w)$ iid. $\sim\mathcal{N}$, prior $P(w)\sim\mathcal{N}$
MAP: $w^* = \arg\max_w P(w)\prod_i P(y_i|x_i,w) =$
$= \arg\min_w \frac{1}{2\sigma^2}\sum_{i=1}^n (y_i - w^T x_i)^2 + \frac{1}{2\beta^2}\|w\|_2^2$

Regularization
Optim: $\arg\min_w \sum_i^n l(w^T x_i; x_i, y_i) + C(w)$
Prior: $C(w) = -\log P(w)$
Likelihood: $l(w^T x_i; x_i, y_i) = -\log P(y_i|x_i,w)$

Classification
Min Prediction Error: $R(h) = \mathbb{E}_{x,y}[[Y \neq h(x)]]$
$h^*(x) = \arg\min_{\hat{y}} \mathbb{E}_y[[Y\neq\hat{y}]|X=x] =$
$= \arg\max_{\hat{y}} P(Y=\hat{y}|X=x)$

Logistic Regression
Link function: $\sigma(w^T x) = (1+exp(-w^T x))^{-1}$
Ass: Bernoulli noise $P(y|x,w) = \text{Ber}(y;\sigma(w^T x))$
Cond. dist: $P(y|x,\hat{w}) = (1+\exp(-y\hat{w}^T x))^{-1}$
Pred: $\hat{y} = \arg\max_{\hat{y}} P(\hat{y}|x,\hat{w}) = \text{sign}(w^T x)$
MLE: $w^* = \arg\min_w \sum_{i=1}^n l_{\log}(w)$

Logistic loss
$l_{\log}$ is convex, everywhere diff., reg $z = -yw^T x$
$l_{\log}(z) = \log(1+e^{-z}) \approx z$ for $z \gg 0$, ≈ 0 for $z \ll 0$
$l_{\log}(w) = \log(1+\exp(-y_i w^T x_i)) = P(Y=y|w,x)$
$\nabla_w l_{\log} = \frac{1}{1+\exp(yw^T x)}(-yx) = \hat{P}(Y=-y|w,x)(-yx)$

SGD Logistic Regression
Initialize w, for $t=1,2...$: $(x,y) \in_{\text{unif.a.r}} D$
Missclassif prob: $\hat{P}(Y=-y|w,x) = (1+exp(yw^T x))^{-1}$
Update:$w \leftarrow w + \eta_t y x \hat{P}(Y=-y|w,x)$

Cross-Entropy loss
$l_{CE}(y;x,w_{1:c}) = -\log P(Y=y|x,w_{1:c})$
Softmax: $P(Y=y|x,w_{1:c}) = \frac{\exp(w_i^T x)}{\sum_{j=1}^c \exp(w_j^T x)}$

Bayesian decision theory
- Conditional distribution over labels $P(y|x)$
- Set of actions $\mathcal{A}$
- Cost function $C: Y\times\mathcal{A}\to\mathbb{R}$

Choose action that minimizes the expected cost:
$a^* = \arg\min_{a\in\mathcal{A}} \mathbb{E}_y[C(y,a)|x] = \sum_y P(y|x)\cdot C(y,a)$

Logistic regression
Cond. dist: $\hat{P}(y|x) = Ber(y;\sigma(\hat{w}^T x))$
Actions: $\mathcal{A} = +1,-1$, Cost: $C(y,a) = [y\neq a]$

Asymmetric costs
Cost: $C(y,a) = \begin{cases} c_{FP}\text{ , if } y=-1 \text{ and } a=+1 \\ c_{FN}\text{ , if } y=+1 \text{ and } a=-1 \\ 0\text{ , otherwise}\end{cases}$
$C_+ = \mathbb{E}_y[C(y,+1)|x] = P(y=-1|x)\cdot c_{FP}$
$C_- = \mathbb{E}_y[C(y,-1)|x] = P(y=+1|x)\cdot c_{FN}$
$a^* = +1$ if $C_+ \le C_- \Leftrightarrow P(y=+1|x) \ge \frac{c_{FP}}{c_{FP}+c_{FN}}$

Doubtful
Actions: $\mathcal{A} = \{+1,-1,D\}$
Cost: $C(y,a) = \begin{cases}[y\neq a] & \text{if } a\in\{+1,-1\}\\ c & \text{if } a = D\end{cases}$
$a^* = y$ if $\hat{P}(y|x) \ge 1-c$, D otherwise

Linear regression
Cond. dist: $\hat{P}(y|x,w) = \mathcal{N}(y;w^T x,\sigma^2)$
Actions: $\mathcal{A}=\mathbb{R}$; Cost: $C(y,a) = (y-a)^2$
$a^* = \mathbb{E}_y[y|x] = \int \hat{P}(y|x)\partial y = \hat{w}^T x$

Asymmetric cost
Cost: $C(y,a) = c_1\max(y-a,0) + c_2\max(a-y,0)$
Cost = underest+ overest, if $c_1 > c_2$, then shift down
$a^* = \hat{w}^T x + \sigma\Phi^{-1}(\frac{c_1}{c_1+c_2})$

Discriminative vs. Generative Modeling
Discriminative est conditional$P(y|x)$
Generative esr joint $P(y,x)$
1. Est prior on labels $P(y)$
2. Est cond. dist $P(x|y)$ (for each class y)
3. Predictive dist. using Bayes' rule:

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

Naive Bayes
Classes: $P(Y) = P(Y=y) = p_y$
Features: $P(X|Y) = \prod_i^n P(x_i|y)$
Joint: $P(X,Y) = \prod_i^n P(x_i,y_i) = P(y)\prod_i^n P(x_i|y)$
Ass: $X_{i:n}$ are conditionally independent given Y

MLE for P(y)
Ass: $P(Y=1)=p, P(y=-1) = 1-p$ (Bernoulli)
MLE: $p^* = \prod_{i=1}^n p^{[y_i=+1]}(1-p)^{[y_i=-1]} = \frac{n_+}{n_++n_-}$

MLE for P=(x|y)
Ass: $P(X=x_i|y) = \mathcal{N}(x_i;\mu_{i,y},\sigma_{i,y}^2)$ (Gaussian)
MLE: $\hat{\mu}_{i,y} = \frac{1}{n_y}\sum_{x_i|y} x$; $\hat{\sigma}_{i,y}^2 = \frac{1}{n_y}\sum_{x_i|y}(x-\hat{\mu}_{i,y})^2$

Decision / Classification rule
$P(y|x) = \frac{1}{Z}P(y)P(x|y)$, $Z = \sum_y P(y)P(x|y)$
$y^* = \arg\max_y P(y|x) = \log P(y) + \sum_i^d \log P(x_i|y)$

Gaussian Naive Bayes (different Var)
MLE class prior: $\hat{P}(Y=y) = \hat{p}_y = \frac{n_y}{n}$
MLE feature dist.: $\hat{P}(x_i|y) = \mathcal{N}(x_i;\hat{\mu}_{y,i},\sigma_{y,i}^2)$
$\hat{\mu}_{y,i} = \frac{1}{n_y}\sum_{j:y_j=y} x_{j,i}$, $\sigma_{y,i}^2 = \frac{1}{n_y}\sum_{j:y_j=y}(x_{j,i}-\hat{\mu}_{y,i})^2$
Pred: $y = \arg\max_{y'}\hat{P}(y'|x) = \text{sign}(f(x))$
Discriminant: $f(x) = \log\frac{P(Y=1|x)}{P(Y=-1|x)}$

Gaussian Naive Bayes (common Var, c=2)
Ass: $P(Y=1) = p_+$; $P(x|y) = \prod_i \mathcal{N}(x_i;\mu_{y,i},\sigma_i^2)$
Discriminant: $f(x) = w^T x + w_0$
$w_i = \frac{\mu_{+,i}-\mu_{-,i}}{\sigma_i^2}$; $w_0 = \log\frac{\hat{p}_+}{1-\hat{p}_+} + \sum_{i=1}^d \frac{\hat{\mu}_{-,i}^2 - \hat{\mu}_{+,i}^2}{2\hat{\sigma}_i^2}$
Class dist: $P(Y=1|x) = (1+\exp(-f(x)))^{-1} = \sigma(f(x))$

Quadratic Discriminant Analysis
Classes: $P(Y=y) = p_y$
Features: $P(X|Y) = \mathcal{N}(x;\mu_y,\Sigma_y)$
Ass: features generated by multivariate Normal
MLE class prior: $\hat{P}(Y=y) = \hat{p}_y = \frac{n_y}{n}$
MLE feature dist: $\hat{P}(x|y) = \mathcal{N}(x;\hat{\mu}_y,\hat{\Sigma}_y)$
$\hat{\mu}_y = \frac{1}{n_y}\sum_{i:y_i=y} x_i$; $\hat{\Sigma}_y = \frac{1}{n_y}\sum_{i:y_i=y}(x_i-\hat{\mu}_y)(x_i-\hat{\mu}_y)^2$
Discriminant: $f(x) = \log\frac{p}{1-p} + \frac{1}{2}\Big[\log\frac{|\hat{\Sigma}_-|}{|\hat{\Sigma}_+|} +$
$+(x-\hat{\mu}_-)^T\hat{\Sigma}_-^{-1}(x-\hat{\mu}_-) - (x-\hat{\mu}_+)^T\hat{\Sigma}_+^{-1}(x-\hat{\mu}_+)\Big]$

Linear Discriminant Analysis
Ass: $p=0.5$; $\hat{\Sigma}_- = \hat{\Sigma}_+ = \hat{\Sigma}$
$f(x) = x^T\hat{\Sigma}^{-1}(\hat{\mu}_- + -\hat{\mu}_-) + \frac{1}{2}(\hat{\mu}_-^T\hat{\Sigma}^{-1}\hat{\mu}_- - \hat{\mu}_+^T\hat{\Sigma}^{-1}\hat{\mu}_+)$
Pred: $y = \text{sign}(f(x)) = \text{sign}(w^T x + w_0)$
$w = \hat{\Sigma}^{-1}(\hat{\mu}_+ - \hat{\mu}_-)$; $w_0 = \frac{1}{2}(\hat{\mu}_-^T\hat{\Sigma}^{-1}\hat{\mu}_- - \hat{\mu}_+^T\hat{\Sigma}^{-1}\hat{\mu}_+)$

Outlier Detection
$P(x) = \sum_{y=1}^c P(y)P(x|y) = \sum_y \hat{p}_y\mathcal{N}(x|\hat{\mu}_y,\hat{\Sigma}_y) \le \tau$

Categorical Naive Bayes Classifier
MLE class prior: $\hat{P}(Y=y) = \hat{p}_y = \frac{n_y}{n}$
MLE feature dist: $\hat{P}(X_i=c|Y=y) = \theta_{c|y}^{(i)} = \frac{n_{c,y}}{n_y}$
Pred: $y^* = \arg\max_y \hat{P}(y|x)$

Latent: Missing Data

Mixture modeling
Gaussian mixture: $P(x|\mu,\Sigma,w) = \sum_j^k w_j\mathcal{N}(x;\mu_j,\Sigma_j)$
Model clusters as probability dist: $P(X|\theta_j)$
Likelihood of iid data: $P(D|\theta) = \prod_i^n\sum_j^k w_j P(x_i|\theta_j)$
Choose params: $\theta^* = \arg\min_\theta -\log P(D|\theta)$
$(\mu^*,\Sigma^*,w^*) = \arg\min -\sum_i^n\log\sum_j^k w_j\mathcal{N}(x_i|\mu_j,\Sigma_j)$

Gaussian Mixture Models
Generate cluster index z_i s.t. $P(z_i=j) = w_j$
Generate data point x_i from $\mathcal{N}(x_i|\mu_{z_i},\Sigma_{z_i})$

Hard-EM
Initialize parameters $\theta^{(0)}$; For $t=1,2,..$
E-Step: Predict most likely class for each data point
$z_i^{(t)} = \arg\max_z P(z|x_i,\theta) = P(z|\theta^{(t-1)})P(x_i|z,\theta^{(t-1)})$
Complete data: $D^{(t)} = \{(x_i,z_i^{(t)})\forall i\}$
M-Step: Compute MLE as in Gaussian Bayes
$\theta^{(t)} = \arg\max_\theta P(D^{(t)}|\theta)$

Posterior Probabilities
Given: $P(z|\theta), P(x|z,\theta)$; Posterior dist over clusters:
$\gamma_j(x) = P(z|x,\theta) = \frac{w_jP(x|\Sigma_j,\mu_j)}{\sum_l^k w_lP(x|\Sigma_l,\mu_l)}$

Soft-EM
E-Step: Calculate clusters weights (responsibilities)
$\gamma_j^{(t)}(x_i)\ \forall i,j$ given $\mu^{(t-1)},\Sigma^{(t-1)},w^{(t-1)}$
M-Step: Fit clusters to weighted data points (MLE)
$w_j^{(t)} \leftarrow \frac{1}{n}\sum_{i=1}^n\gamma_j^{(t)}(x_i)$; $\mu_j^{(t)} \leftarrow \frac{\sum_{i=1}^n\gamma_j^{(t)}(x_i)x_i}{\sum_{i=1}^n\gamma_j^{(t)}(x_i)}$
$\Sigma_j^{(t)} \leftarrow \frac{\sum_{i=1}^n\gamma_j^{(t)}(x_i)(x_i-\mu_j^{(t)})(x_i-\mu_j^{(t)})^T}{\sum_{i=1}^n\gamma_j^{(t)}(x_i)}$

Things To Remember
$ln(x) \le x-1, x>0$; $e^x = 1+x+\frac{x^2}{2!}+\frac{x^3}{3!} = \lim(1+\frac{x}{n})^n$
$\frac{\exp(f(x))}{1+\exp(f(x))} = \frac{1}{1+\exp(-f(x))}$
$\|x\|_2 = \sqrt{x^T x}$; $\nabla_x\|x\|_2^2 = 2x$
$f(x) = x^T Ax$; $\nabla_x f(x) = (A+A^T)x$

CDF: $\Phi(x) = \int_{-\infty}^x \phi(t)\partial t$; PDF: $\phi(x) = \frac{1}{\sqrt{2\pi}}e^{-(1/2)x^2}$
$\int x\phi(x) = -\phi(x)+c$; $\int x^2\phi(x)\partial x = \Phi(x) - x\phi(x) + c$

Probabilities
$Var[X] = \mathbb{E}[(X-\mu_X)^2] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$
$P(A|B) = \frac{P(B|A)\cdot P(A)}{P(B)}$; $p(Z|X,\theta) = \frac{p(X,Z|\theta)}{p(X|\theta)}$
Joint: $P(x,y) = P(y|x)\cdot P(x) = P(x|y)\cdot P(y)$
E step: $P(Z|X,\theta) = P(X,Z)/P(X) \sim P(Z,X|\theta)$
M step: $\theta = \arg\max\mathbb{E}_Z\log P(X,Z\theta)$