Math Review

X and **Y** are **independent** iff $P(\mathbf{X},\mathbf{Y}) = P(\mathbf{X})P(\mathbf{Y})$

X and **Y** are **uncorrelated** iff $\mathbb{E}(\mathbf{X},\mathbf{Y}) = \mathbb{E}(\mathbf{X})\mathbb{E}(\mathbf{Y})$

Expected value of $g(X)$: $E[g(X)] = \int_{-\infty}^{\infty} g(x)f(x)dx$

Variance $\sigma^2 = E[(X-\mu)^2] = E[X^2] - \mu^2$

Determinant of matrix is product of its eigenvalues.

$f(\mathbf{x}) = \mathbf{Ax}+\mathbf{x}'\mathbf{A}+\mathbf{x}'\mathbf{x}+\mathbf{x}'\mathbf{Ax} \Rightarrow \frac{df(\mathbf{x})}{d\mathbf{x}} = \mathbf{A}'+\mathbf{A}+2\mathbf{x}+\mathbf{Ax}+\mathbf{A}'\mathbf{x}$

$$\nabla_{\boldsymbol{x}}(\boldsymbol{y}\cdot\boldsymbol{z}) = (\nabla_{\boldsymbol{x}})\boldsymbol{z} + (\nabla_{\boldsymbol{x}})\boldsymbol{y} \qquad \nabla_{\boldsymbol{x}} f(\boldsymbol{y}) = (\nabla_{\boldsymbol{x}}\boldsymbol{y})(\nabla_{\boldsymbol{y}} f(\boldsymbol{y}))$$

$$\nabla_w w^T A w = (A + A^T)w \qquad \mathbf{H}_{i,j} = \frac{\partial^2 f}{\partial x_i \partial x_j}$$

Perceptron

$f(\mathbf{x}) = \mathbf{w}\cdot\mathbf{x} + \alpha = \sum_{i=1}^d w_i x_i + \alpha$,
Goal: find w s.t all constraints $y_i X_i \cdot w \geq 0$. Define a risk function and optimize it, where the loss is defined as $L(z, y_i) = -y_i z$ if $y_i z < 0$, else 0. Therefore risk $R(w) = \sum_{i\in V} -y_i X_i \cdot w$

Decision boundary, a **hyperplane** in $\mathbb{R}^d$:
$H = \{\mathbf{x}\in\mathbb{R}^d : f(\mathbf{x}) = 0\} = \{\mathbf{x}\in\mathbb{R}^d : \mathbf{w}\cdot\mathbf{x} + \alpha = 0\}$

$\mathbf{w}$ is the **normal** of the hyperplane,
α is the **offset** of the hyperplane from origin,
$\frac{f(\mathbf{x})}{\|\mathbf{w}\|}$ is the **signed distance** from the $\mathbf{x}$ to hyperplane $\mathcal{H}$.

Perceptron algorithm,
Input: $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \in \mathbb{R}^d \times \{\pm 1\}$
while some $y_i \neq \operatorname{sign}(\mathbf{w}\cdot\mathbf{x}_i)$
  pick some misclassified $(\mathbf{x}_i, y_i)$
  $\mathbf{w} \leftarrow \mathbf{w} + y_i \mathbf{x}_i$

Given a **linearly separable data**, perceptron algorithm will take no more than $\frac{R^2}{\gamma^2}$ updates to **converge**, where $R = \max_i \|\mathbf{x}_i\|$ is the radius of the data, $\gamma = \min_i \frac{y_i(\mathbf{w}\cdot\mathbf{x}_i)}{\|\mathbf{w}\|}$ is the margin.
Also, $\frac{\mathbf{w}\cdot\mathbf{x}}{\|\mathbf{w}\|}$ is the signed distance from H to $\mathbf{x}$ in the direction $\mathbf{w}$.

Gradient descent view of perceptron, minimize margin cost function $J(\mathbf{w}) = \sum_i (-y_i(\mathbf{w}\cdot\mathbf{x}_i))_+$ with $\mathbf{w} \leftarrow \mathbf{w} - \eta\nabla J(\mathbf{w})$

Support Vector Machine

Hard margin SVM,
This method makes the margin as wide as possible. The signed distance from the hyperplane to X_i is $\frac{f(\mathbf{x}_i)}{\|\mathbf{w}\|}$ Hence the margin is
$\min_i \frac{1}{\|w\|}|w\cdot X_i + \alpha| \geq \frac{1}{\|w\|} \implies \min_{\mathbf{w}} \|\mathbf{w}\|^2$, such that $y_i \mathbf{w}\cdot\mathbf{x}_i \geq 1 (i = 1, \ldots, n)$
Soft margin SVM,
$\min_{\mathbf{w}} \|\mathbf{w}\|^2 + C\sum_{i=1}^n \xi_i$

Regularization and SVMs: Simulated data with many features $\phi(\mathbf{x})$; C controls trade-off between margin $1/\|\mathbf{w}\|$ and fit to data; Large C: focus on fit to data (small margin is ok). More overfitting. Small C: focus on large margin, less tendency to overfit. Overfitting increases with: less data, more features.

Decision Theory

Bayes Theorem: $\underbrace{P(Y=C|X)}_{\text{Poster. Prob}} = \frac{P(X|Y=C)\overbrace{P(Y=C)}^{\text{Prior Prob.}}}{P(X)}$ Assume

$(\mathbf{X},\mathbf{Y})$ are chosen i.i.d according to some probability distribution on $\mathcal{X}\times\mathcal{Y}$. **Risk** is misclassification probability:
$R(r) = \mathbb{E}(L(r(\mathbf{X}),\mathbf{Y})) = Pr(r(\mathbf{X}) \neq \mathbf{Y}) =$
$\sum_{\mathbf{x}} \left[L(r(\mathbf{x}),1)P(Y=1|x) + L(r(x),-1)P(Y=-1|X=\mathbf{x})\right] \times P(\mathbf{x}$

$= P(Y=1)\sum_x L(r(\mathbf{x}),1)P(\mathbf{x}|Y=1) + P(Y= -1)\sum_x L(r(\mathbf{x}),-1)P(\mathbf{x}|Y=-1)$

Bayes Decision Rule is
$r^*(x) = \begin{cases} 1, & \text{if } L(-1,1)P(\mathbf{Y}=1|x) > L(1,-1)P(\mathbf{Y}=-1|x) \\ -1, & \text{otherwise.} \end{cases}$,
and the optimal risk (Bayes risk) $R^* = \inf_r R(r) = R(r^*)$

Risk in Regression is expected squared error:
$R(f) = \mathbb{E}l(f(\mathbf{X}),\mathbf{Y}) = \mathbb{E}\mathbb{E}[f(\mathbf{X}) - \mathbf{Y}^2|\mathbf{X}]$

Bias-variance decomposition:
$R(f) = \mathbb{E}[\underbrace{(f(\mathbf{X}) - \mathbb{E}[\mathbf{Y}|\mathbf{X}])^2}_{\text{bias}^2}] + \mathbb{E}[\underbrace{(\mathbb{E}[\mathbf{Y}|\mathbf{X}] - \mathbf{Y})^2}_{\text{variance}}]$
$R(f) = \mathbb{E}[(f(\mathbf{X}) - f^*(\mathbf{X}))^2] + \mathbb{E}[(f^*(\mathbf{X}) - \mathbf{Y})^2]$
$R(f) = \mathbb{E}[(f(\mathbf{X}) - f^*(\mathbf{X}))^2] + R(f*)$
$R(f) - R(f*) = \mathbb{E}[(f(\mathbf{X}) - f^*(\mathbf{X}))^2]$, $f^*(\mathbf{X}) = \mathbb{E}[\mathbf{Y}|\mathbf{X}]$

Generative and Discriminative

Discriminative models: $P(\mathbf{X},\mathbf{Y}) = P(\mathbf{X})P(\mathbf{Y}|\mathbf{X})$.
Estimate $P(\mathbf{Y}|\mathbf{X})$, then pretend out estimate $\hat{P}(\mathbf{Y}|\mathbf{X})$ is the actual $P(\mathbf{Y}|\mathbf{X})$ and plug in bayes rule expression.

Generative model: $P(\mathbf{X},\mathbf{Y}) = P(\mathbf{Y})P(\mathbf{X}|\mathbf{Y})$.
Estimate $P(\mathbf{Y})$ and $P(\mathbf{X}|\mathbf{Y})$, then use bayes theorem to calculate $P(\mathbf{Y}|\mathbf{X})$ and use discriminative model.

Gaussian class conditional densities $P(\mathbf{X}|Y=+1)$,$P(\mathbf{X}|Y=-1)$ (with the same variance), the posterior probability is **logistic**:
$P(Y=+1|\mathbf{x}) = \frac{1}{1+\exp(-\mathbf{x}\cdot\mathbf{w}-\beta_0)}$,
$\mathbf{w} = \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$, $\beta_0 = \frac{\boldsymbol{\mu}_0'\Sigma^{-1}\boldsymbol{\mu}0 - \boldsymbol{\mu}_1\Sigma^{-1}\boldsymbol{\mu}1}{2} + \log\frac{P(Y=1)}{P(Y=0)}$

Multivariate Normal Distribution

$\mathbf{x}\in\mathbb{R}^d : p(x) = \frac{1}{(2\pi)^{d/2}|\mathbf{\Sigma}|^{1/2}} e^{(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})'\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}))}$

QDA: Class-conditional densities $X_C \sim \mathcal{N}((,)\mu_C, \Sigma_C)$. Optimal decision rule $r^*(x)$ for 0-1 loss: Choose class **C** that maxes $P(Y=C|X) \propto f_C(x)\pi_C$. Parameters estimated via MLE:
LDA: Assumes equal covariance matrices across classes ($\Sigma_C = \Sigma$), simplifying to linear decision surfaces.

$\mathbf{\Sigma} = \mathbb{E}(\mathbf{X}-\boldsymbol{\mu})(\mathbf{X}-\boldsymbol{\mu})'$
Symmetric: $\mathbf{\Sigma}_{i,j} = \mathbf{\Sigma}_{j_i}$
Non-negative diagonal entries: $\mathbf{\Sigma}i, i \geq 0$
Positive semidefinite: $\forall\mathbf{v}\in\mathbb{R}^d, \mathbf{v}'\mathbf{\Sigma}\mathbf{v} \geq 0$

Given a d-dimensaional Gaussian $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu},\mathbf{\Sigma})$, matrix $\mathbf{A}\in\mathbb{R}^{m\times d}$ and vector $\mathbf{b}\in\mathbb{R}^m$, define $\mathbf{Y} = \mathbf{AX}+\mathbf{b}$.
Then $\mathbf{Y} \sim \mathcal{N}(\mathbf{A}\boldsymbol{\mu}+\mathbf{b}, \mathbf{A\Sigma A}')$

Given a d-dimensaional Gaussian $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu},\mathbf{\Sigma})$, with $\mathbf{\Sigma}$ positive definite,
$\mathbf{Y} = \mathbf{\Sigma}^{-\frac{1}{2}}(\mathbf{X}-\boldsymbol{\mu}) \sim \mathcal{N}(\mathbf{0},\mathbf{I},)$

MLE's

Maximum a posterior probability: the mode of the posterior. If uniform prior, MAP is MLE; if not uniform prior, MAP is Penalized MLE.

Prior: $\hat{\pi}_C = P(Y=C) = \frac{N_C}{n}$

Mean: $\hat{\mu}_C = \mathbb{E}[\mathbf{X}|Y=C] = \frac{1}{N_C}\sum_{i:Y_i=C} X_i$

Covariance: $\hat{\Sigma}_C = \frac{1}{N_C}\sum_{i:Y_i=C}(X_i - \hat{\mu}_C)(X_i - \hat{\mu}_C)^\top$

Pooled Cov: $\hat{\Sigma} = \frac{1}{n}\sum_{C_k}\sum_{i:Y_i=C_k}(X_i - \hat{\mu}_{C_k})(X_i - \hat{\mu}_{C_k})^\top$

Linear Regression

Empirical risk minimization

Empirical risk is the sample average of squared error:
$\hat{R}(r) = \hat{\mathbb{E}}_n L(r(\mathbf{X}), Y) = \frac{1}{n}\sum_{i=1} n(r(\mathbf{X}_i) - Y_i)^2$
Choose $\hat{f} := \arg\, min_{f\in F_{\text{lin}}} \hat{\mathbb{E}}_n L(f(\mathbf{X}), Y)$

Find $\hat{r} : \mathbf{x} \mapsto \mathbf{x}^T\hat{\mathbf{w}}$, such that
$\hat{\mathbf{w}} = \arg\, min_{\mathbf{w}\in\mathbb{R}^p} \sum_{i=1}^n (\mathbf{X}_i'\mathbf{w} - Y_i)^2 = \arg\, min_{\mathbf{w}\in\mathbb{R}^p} \underbrace{\|\mathbf{Xw}-\mathbf{y}\|^2}_{\text{RSS}}$
where **design matrix** $\mathbf{X}\in\mathbb{R}^{n\times p}$ and **response vector** $\mathbf{y}\in\mathbb{R}^n$.

Normal equations: $\mathbf{X}'\mathbf{Xw} = \mathbf{X}'\mathbf{y} \Rightarrow \hat{\mathbf{w}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$

Projection Theorem also leads to normal equations:
$(\mathbf{y}-\hat{\mathbf{y}})^{-1}\mathbf{X} = 0 \Leftrightarrow \mathbf{X}'(\mathbf{y}-\mathbf{Xw}) = 0 \Leftrightarrow \mathbf{X}'\mathbf{y} = \mathbf{X}'\mathbf{Xw}$

Linear model with additive Gaussian noise

Typical model of reality: $y_i = g(X_i) + \epsilon_i : \epsilon \sim \mathcal{N}(\mathbf{0},\sigma^2)$. The goal of regression is to find h that estimates g, the ground truth.
Ideal h: $h(x) = E_Y[Y|X=x] = g(x) + E[\epsilon] = g(x)$

$\implies y_i \sim \mathcal{N}(g(X_i),\sigma^2)$

$\implies P(Y|\mathbf{X}=\mathbf{x}) = \mathcal{N}(\mathbf{x}'\mathbf{w},\sigma^2)$
Equivalently: $Y = \mathbf{x}'\mathbf{w} + \epsilon$, where $\epsilon \sim \mathcal{N}(\mathbf{0},\sigma^2)$
Maximum likelihood is least square, fix **X**. Provided $\mathbb{E}\mathbf{y} = \mathbf{Xw}$ and $\operatorname{Cov}(\mathbf{y}) = \sigma^2\mathbf{I}$

Bayesian analysis: Treat $\mathbf{w}$ as a r.v. with prior distribution $\mathcal{N}(\mathbf{0},\tau^2\mathbf{I})$, then compute posterior distribution $P(\mathbf{w}|\mathbf{X},Y)$.

$P(\mathbf{w}|\mathbf{X}_1, Y_1, \ldots, \mathbf{X}_n, Y_n) \propto P(Y_1, \ldots, Y_n|\mathbf{w}, \mathbf{X}_1, \ldots, \mathbf{X}_n)P(\mathbf{w})$
$P(\mathbf{w}|\mathbf{X}_1, Y_1, \ldots, \mathbf{X}_n, Y_n) \propto exp(-\frac{1}{2}(\sum_{i=1}^n \frac{(Y_i - \mathbf{X}_i'\mathbf{w})^2}{\sigma^2} + \frac{1}{\tau^2}\|\mathbf{w}\|^2))$

Linear Regression Regularization

Trading off bias and variance: some increase in bias can give a big decrease in variance

---

Ridge regression is like $L2$ regularization:
$\hat{\mathbf{w}} = \arg\ \min_{\mathbf{w}}(\sum_{i=1}^{n}(y_i - \mathbf{x}_i'\mathbf{w})^2 + \lambda\sum_{j=1}p\beta_j^2)$
$\hat{\mathbf{w}}^{\text{ridge}} = (\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{x}'\mathbf{y}$

Lasso regression is like $L1$ regularization:
$\hat{\mathbf{w}} = \arg\ \min_{\mathbf{w}}(\sum_{i=1}^{n}(y_i - \mathbf{x}_i'\mathbf{w})^2 + \lambda\sum_{j=1}p|\beta_j|)$ While ridge regression leads to reduced, but rare non-zero values of the weights, Lasso regression forces some weights to be zero.

Bayesian analysis: Ridge regression is equivalent to a MAP estimate with a gaussian prior. Lasso regression is equivalent to a MAP estimate with a Laplace prior.

Logistic Regression

$P(Y=1|\mathbf{x}) = \frac{1}{1+\exp(-\mathbf{w}'\mathbf{x})} = \sigma(\mathbf{w}'\mathbf{x})$
Given data $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n) \in \mathbb{R}^d \times \{0,1\}$, estimate $\mathbf{w}$ with maximum likelihood.

Log likelihood:
$\ell(\mathbf{w}) = \sum_{i=1}^{n} y_i \log s_i) + (1-y_i)\log(1-s_i)),$
where $s_i = P(Y=1|\mathbf{X}=\mathbf{x}_i, \mathbf{w}) = \frac{1}{1+\exp(-\mathbf{w}'\mathbf{x}_i)} = \sigma(\mathbf{w}'\mathbf{x}_i)$

$\nabla_{\mathbf{w}} s_i = s_i(1-s_i)\mathbf{x}_i$
$\nabla_{\mathbf{w}}\ell(\mathbf{w}) = \mathbf{X}'(\mathbf{s}-\mathbf{y})$
$\nabla_{\mathbf{w}}^2\ell(\mathbf{w}) = \mathbf{X}'\text{diag}(\boldsymbol{\mu}(1-\boldsymbol{\mu}))\mathbf{X}$

Gradient ascent:
$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta\nabla_{\mathbf{w}}R(\mathbf{w}^{(t)})$: $O(nd)$ per step
Stochastic gradient ascent:
$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta\nabla R_i(\mathbf{w}^{(t)})$: $O(d)$ per step
Newton's method:
$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - [\nabla_{\mathbf{w}}^2 R(\mathbf{w}^{(t)})]^{-1}\nabla_{\mathbf{w}}R(\mathbf{w}^{(t)})$

Discriminant Analysis

Discriminant Fn (For LDA and QDA):
$Q_C(\mathbf{x}) = \ln\left((2\pi)^{-\frac{d}{2}} f_{\mathbf{X}|Y=C}(\mathbf{x})\pi_C\right) =$
$-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_C)^T\boldsymbol{\Sigma}_C^{-1}(\mathbf{x}-\boldsymbol{\mu}_C) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_C| + \ln\pi_C.$

For Multi-class LDA: choose C that maxes linear Q_C:
$\boldsymbol{\mu}_C^T\boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}\boldsymbol{\mu}_C^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_C + \ln\pi_C$

Linear Decision Function:
$$Q_C(\mathbf{x}) - Q_D(\mathbf{x}) = \underbrace{(\boldsymbol{\mu}_C - \boldsymbol{\mu}_D)^T\boldsymbol{\Sigma}^{-1}\mathbf{x}}_{\mathbf{w}^T\mathbf{x}} \underbrace{-\frac{1}{2}\boldsymbol{\mu}_C^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_C - \frac{1}{2}\boldsymbol{\mu}_D^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_D + \ln\pi_C - \ln\pi_D}_{\alpha}.$$

Misc

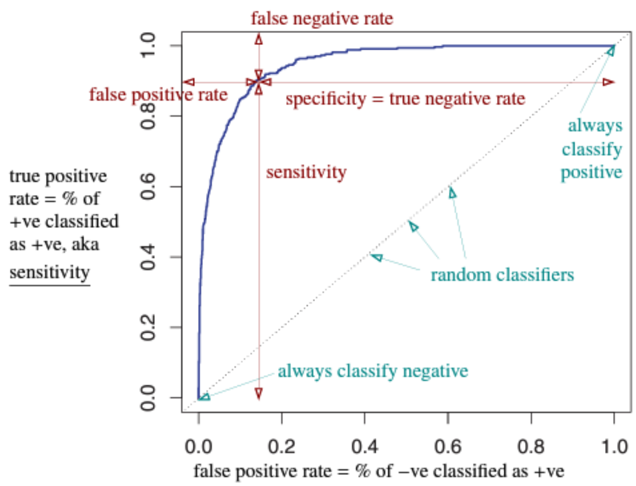
Centering X: This involves subtracting $\boldsymbol{\mu}^T$ from each row of $\mathbf{X}$. Symbolically, $\mathbf{X}$ transforms into $\bar{\mathbf{X}}$.

Decorrelating X: This process applies a rotation $\mathbf{Z} = \bar{\mathbf{X}}\mathbf{V}$, where $\text{Var}(\mathbf{R}) = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T$. This step rotates the sample points to the eigenvector coordinate system.

Sphering: $\bar{\mathbf{X}}$: applying transform $\mathbf{W} = \bar{\mathbf{X}}\text{Var}(\mathbf{R})^{-\frac{1}{2}}$

whitening X: centering + sphering, $\mathbf{X} \to \mathbf{W}$

ROC Curves



Past Exams Q/A

Support Vector Machine

* True: If the data is not linearly separable, there is no solution to hard margin SVM.
* True: Complementarity slackness implies that every training point that is misclassified by a soft margin SVM is a support vector.
* True: When we solve the SVM with the dual problem, we need only the dot product of x_i and x_j for all i, j.

Logistic Regression

* True: Logistic regression can be used for classification.
* Logistic regression can be motivated from log odds equated to an affine function of x and generative models with gaussian class conditionals.

Linear Regression

* L2 regularization is equivalent to imposing a Gaussian prior in linear regression.
* If we have 2 two-dimensional Gaussians, the same covariance matrix for both will result in a linear decision boundary.
* The normal equations can be derived from minimizing empirical risk, assuming normally distributed noise, and assuming $P(Y|X)$ is distributed normally with mean $B^\top X$ and variance σ^2.

Perceptron

* The perceptron algorithm will converge only if the data is linearly separable.

Decision Theory

* True: a discriminative classifier explicitly models $P(Y|X)$.

Optimization

- True: Newton's method is typically more expensive to calculate than gradient descent per iteration.
- True: for quadratic equations, Newton's method typically requires fewer iterations than gradient descent.
- False: Gradient descent can be viewed as iteratively reweighted least squares.
- True: we use Lagrange multipliers in an optimization problem with inequality constraints.

Multivariate Gaussian

* False: For multivariate gaussian, the eigenvalues of the covariance matrix are inversely proportional to the lengths of the ellipsoid axes that determine the isocontours of the density.

General Machine Learning Practices

* True: It is not good machine learning practice to use the test set to help adjust the hyperparameters.