6.036 Official Cheat Sheet

- A (hyper-)plane is a set of points $x \in \mathcal{R}^d$ such that $\theta \cdot x + \theta_0 = 0$. Vector θ is normal to the plane. The signed distance of any point x from the plane is $\frac{\theta \cdot x + \theta_0}{\|\theta\|}$. The value of distance is positive on the side where θ points to, and negative on the other side.

- A linear classifier with offset:
  $h(x; \theta) = \text{sign}(\theta \cdot x + \theta_0)$

- Training error (classification error):
  $\epsilon_n(h) = \frac{1}{n} \sum_{i=1}^n [[y^{(i)} \neq h(x^{(i)})]]$

- Distance functions:

$$D_{L1}(x, x') = \sum_{i=1}^d |x_i - x'_i|$$

$$D_{L2}(x, x') = \sum_{i=1}^d (x_i - x'_i)^2$$

- Loss Functions: $z = y(\theta \cdot x + \theta_0)$ (agreement)

$$\text{Loss}_{0,1}(z) = [[z \leq 0]]$$

$$\text{Loss}_{\text{hinge}}(z) = \max\{1 - z, 0\}$$

- Passive-aggressive algorithm without offset.
  At step k, in response to (x, y), find $\theta^{(k+1)}$ that minimizes $\lambda \|\theta - \theta^{(k)}\|^2/2 + Loss_{hinge}(y\theta \cdot x)$

- Linear regression: predict value $\theta \cdot x + \theta_0$
  minimize $\frac{\lambda}{2}||\theta||^2 + \sum_{i=1}^n (y^{(i)} - \theta \cdot x^{(i)} - \theta_0)^2/2$

- Kernels: $K(x, x') = \phi(x) \cdot \phi(x')$

Kernel	form
Linear	$x \cdot x'$
Quadratic	$x \cdot x' + (x \cdot x')^2$
radial basis	$\exp(-\|x - x'\|^2/2)$

- Kernel Perceptron: Cycles through $t = 1, \ldots, n$ and checks if $y^{(t)} \sum_{i=1}^n \alpha_i y^{(i)} K(x^{(i)}, x^{(t)}) \leq 0$. If true, $\alpha_t = \alpha_t + 1$. (without offset)

- kernelized classifier (without offset):
  $h(x) = \text{sign}(\sum_{i=1}^n \alpha_i y^{(i)} k(x^{(i)}, x))$

- Boosting: Adaboost formulas
  $\epsilon_m = \sum_{t=1}^n W_{m-1}(t)[[y_t \neq h(x_t; \theta_m)]]$

  $\alpha_m = 0.5 \log(\frac{1 - \epsilon_m}{\epsilon_m})$

  $W_m(t) = c_m W_{m-1}(t) \exp(-y_t \alpha_m h(x_t; \theta_m))$

- Neural Nets: output of a single hidden layer network with activation function f is $F(x; \theta) = \sum_{j=1}^m f(z_j) V_j + V_0$, where $z_j = \sum_{i=1}^d x_i W_{ij} + W_{0j}$.

- Stochastic Gradient Descent
  $\theta \leftarrow \theta - \eta_k \nabla_\theta \text{Loss}(y^{(t)} F(x^{(t)}; \theta))$
  where η_k is the learning rate after k steps

generalization error:

In the realizable case, $\epsilon(\hat{h}) \leq \epsilon_n(h) + \frac{\log|H| + \log(\frac{1}{\delta})}{n}$ where $\epsilon_n(h)$ is the training error and H is a finite set of classifiers.

In the non-realizable case, we obtain a weaker bound: $\epsilon(\hat{h}) \leq \epsilon_n(h) + \sqrt{\frac{\log|H| + \log(\frac{1}{\sigma})}{2n}}$.
When H is not finite, $\log|H|$ will be roughly speaking replaced by the growth function $\log N_H(n)$ which relates to the VC-dimension.

BIC

$BIC(D; \theta) = l(D; \theta) - \frac{\text{number of params}}{2} \log(n)$
where D is the data containing n examples.

HMM

- $P(X_1, ..., X_T) = P(X_1) \Pi_{t=2}^T P(X_t | X_{t-1})$

- $P(Y_{1:T}, X_{1:T}) =$
  $P(Y_1) P(X_1 | Y_1|) \Pi_{t=2}^T P(Y_t | Y_{t-1}) P(X_t | Y_t)$

Q-Value Iteration Algorithm:

1. $Q_0(s, a) = 0$

2. $Q_{i+1}(s, a) = \sum_{s'} T(s, a, s')[R(s, a, s') + \gamma \max_{a'} Q_i(s', a')]$

Value Iteration Algorithm:

1. $V_0(s) = 0$

2. $V_{i+1}(s) = \max_a \left[\sum_{s'} T(s, a, s')[R(s, a, s') + \gamma V_i(s')\right]$

Note that $V_i(s') = \max_{a'} Q_i(s', a')$

Q-learning

Model-free estimation (to avoid explicitly computing T,R) $Q(s,a) \leftarrow Q(s,a) + \alpha[R(s,a,s') + \gamma \max_{a'} Q(s',a') - Q(s,a)]$

K-Means
$\text{cost}(\mu^{(1)}, \ldots, \mu^{(k)}) = \sum_{i=1}^{n} \min_{j=1,\ldots,k} \|x^{(i)} - \mu^{(j)}\|^2$

1. initialize $\mu^{(1)}, ..., \mu^{(k)}$
2. $\delta(j|i) = [[\, j = \text{argmin}_l \|x^{(i)} - \mu^{(l)}\|^2 \,]]$
3. $\hat{\mu}^{(j)} = \frac{1}{\sum_{i=1}^{n} \delta(j|i)} \sum_{i=1}^{n} \delta(j|i) x^{(i)}$

EM for Gaussians:

1. initialize $\theta = \{p_1, ..., p_k, \mu^{(1)}, ..., \mu^{(k)}, \sigma_1^2, ..., \sigma_k^2\}$
2. E-Step: $p(j|i) = \frac{p_j N(x^{(i)}; \mu^{(j)}, \sigma_j^2 I)}{\sum_{z=1}^{k} p_z N(x^{(i)}; \mu^{(z)}, \sigma_z^2 I)}$
3. M-step: $\max_\theta \sum_{i=1}^{n} \sum_{j=1}^{k} p(j|i) \log[p_j N(x^{(i)}; \mu^{(j)}, \sigma_j^2 I)]$, giving
    - $p_j = \frac{\sum_{i=1}^{n} p(j|i)}{n}$
    - $\hat{\mu}^{(j)} = \frac{1}{\sum_{i=1}^{n} p(j|i)} \sum_{i=1}^{n} p(j|i) x^{(i)}$
    - $\hat{\sigma}_j^2 = \frac{1}{d \sum_{i=1}^{n} p(j|i)} \sum_{i=1}^{n} p(j|i) \|x^{(i)} - \hat{\mu}^{(j)}\|^2$

max-likelihood estimates for $N(x; \mu, \sigma^2 I)$

- If $x \in R$ (1-dimensional):
  $\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} x^{(i)},\ \hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^{n} (x^{(i)} - \hat{\mu})^2$
- If $x \in R^d$ (d-dimensional):
  $\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} x^{(i)},\ \hat{\sigma}^2 = \frac{1}{dn} \sum_{i=1}^{n} \|x^{(i)} - \hat{\mu}\|^2$

log-likelihood
$\ell(S_n; \theta) = \sum_{i=1}^{n} \log P(x^{(i)}; \theta)$