# CS229-Cheatsheet

## Supervised Learning

- **Gradient Descent:** to minimize $J(\theta)$, we perform $\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta} J(\theta)$
- $\nabla_A AB = B^T$ , $\nabla_{A^T} f(A) = (\nabla_A f(A))^T$, $\nabla_A trABA^TC = CAB + C^TAB^T$, $\nabla_A |A| = |A|(A^{-1})^T$
- **Normal Equations and Least Squares** $J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^i) - y^i)^2 \rightarrow \nabla_\theta J(\theta) =$ $\nabla_\theta \frac{1}{2}(X\theta - y)^T(X\theta - y) = X^TX\theta - X^Ty = 0 \rightarrow$ $X^TX\theta = X^Ty \rightarrow \theta = (X^TX)^{-1}X^Ty.$
- **Locally Weighted Regression** Fit $\theta$ to minimize $\sum_{i=0}^{m}(y^i - \theta^T x^i)^2$ where $w^i = e^{-\frac{(x^i - x)^2}{2\tau^2}}$
- **Logistic Regression:** $h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$, $g(z) = \frac{1}{1+e^{-z}}$, $g'(z) = \frac{d}{dz}\frac{1}{1+e^{-z}} = g(z)(1 - g(z))$, $p(y|x;\theta) = (h_\theta(x))^y(1 - h_\theta(x))^{1-y}$. $l(\theta) = logL(\theta) = \sum_{i=1}^{m} y^i logh(x^i) + (1 - y^i)log(1 - h(x^i))$, $\frac{\partial}{\partial \theta_j} l(\theta) = (y - h_\theta(x))x_j$
- **Perceptron Learning Algorithm** $\theta_j := \theta_j + \alpha(y^i - h_\theta(x^i))x_j^i$
- **Newton's Method:** $\theta := \theta - \frac{f(\theta)}{f'(\theta)}$, we want the first derivative to be zero, then $\theta := \theta - \frac{l'(\theta)}{l''(\theta)}$, if $\theta$ is a vector then $\theta := \theta - H^{-1}\nabla_\theta l(\theta)$ where $H_{ij} = \frac{\partial^2 l(\theta}{\partial \theta_i \partial \theta_j}$
- **Exponential Family** $p(y;\eta) = b(y)exp(\eta^T T(y) - a(\eta))$
- **General Linear Model Assumptions:** 1. $y|x;\eta \sim ExponentialFamily(\eta)$. 2. Given $x$ our goal is to predict the expected value of $T(y)$ which is usually just $y$, so we would like our hypothesis to satisfy $h(x) = E(y|x)$. 3. The natural parameter $\eta$ and inputs $x$ are related linearly. $\eta = \theta^T x$.