CS3264
Machine Learning
github.com/securespider

01. Intro
What is
1. Supervised Learning - Classification/Regression
2. Unsupervised - Kmeans, PCA
3. Reinforcement - Giving rewards and punishment

Linear regression
- Data - tuples of points $\{(x_n, t_n)\}_{0 \to N}$
- Model - $y = w_0 + xw_1$
- Loss - Mean square error $\frac{1}{2}\sum_{n=1}^{N}(y_n - t_n)^2$
- Derivative to find lowest point to find w
- $t_n = [x_n][w_n]^T$

Non-Linear regression
- Basis factor expansion $x \to [x, x^2, x^3.., x^P]$
- Find weights for each polynomial/exponential/..
- Note that minimization of error function has a unique solution
  - Unique weight that optimises the model

Regularization
Ridge regression solution
- Add regularizer ($\frac{\lambda}{2}||w||^2$) to error function
  - Lambda is a constant
  - smaller = error matters more
  - larger = penalise weights more

02. Regression II
- $t_n = y(x_n, w) + \epsilon$
  - ϵ is random "noise" modelled via some distribution
- Assumption is that each point is IID
  - Independently identically distributed variables
- MLE is a generalised model estimator given a set of data and predicted distribution

Examples
- MSE is a loss function for MLE assuming data is normally distributed
- Variance of gaussian known but mean is unknown
- Model used to derive mean and minimise loss/maximum likelihood

Bayesian Linear Regression
- $posterior = \frac{likelihood \times prior}{evidence} = \frac{p(y|w)p(w)}{p(y)}$
- Repeatedly updating model based on additional data points
- Initial estimator (prior) that can be modelled in different distributions
  - By CLM, posterior distribution becomes independent to prior
  - Prior models normal distribution
- Maximum-a-Posteriori estimation (MAP)
  - For given prior, after observing data, how to update distr of params?
  - Adds "regularisation" component to normally distributed datasets

03. Linear classification
- Binary splitting of points to classify them (separation line is linear)

Bernoulli Distribution
- Binary possibilities with a probability of $p(t = 1) = \mu$
- $E[t] = \mu, var[t] = \mu(1 - \mu)$

Activation function
- Squashing function that constraints output to distinct classes

Logistic sigmoid
- formula: $\sigma(z) = \frac{1}{1 + exp(-Z)}$
  1. $\sigma(-z) = 1 - \sigma(z)$
  2. $z = lg(\frac{\sigma(z)}{1 - \sigma(z)}$ (log odds)
  3. $\frac{d\sigma(z)}{dz} = \sigma(z)(1 - \sigma(z))$

Approach
Data Logistic sigmoid on $w^T\phi_n$
- Note that $T_n \in \{0, 1\}$ and $y_n = \sigma(w^T\phi_n)$

Model Bayesian model with Bernoulli likelihood and Normal prior
- Likelihood: $y_n^{t_n}(1 - y_n)^{1-t_n}$
- Prior: $p(w) \sim N(w|0, \frac{1}{\alpha}I)$

Loss MAP estimation $lg(p(w\|D))$
- $argmin_w - [\sum_{n=1}^{N} t_n lg y_n + (1 - t_n)lg(1 - y_n)] + \frac{\alpha}{2}w^Tw$
- Cross entropy error encourages y_n to match t_n
- Regulariser continues to prevent overfitting by minimising weights

Solving
- No close form solution so Gradient Descent used on loss function
- $w_{k+1} = w_k - \eta\nabla_w\mathcal{L}(w)$
- $\nabla_w\mathcal{L}(w) = \sum_{n=0}^{N}(y_n - t_n)\phi_n + \alpha w$