1. Exercises 3.4, 3.5 from the Bishop's Textbook

3.4 $E_D(w) = \frac{1}{2}\sum_{n=1}^{N}\left(\left[w_0 + \sum_{i=1}^{D} w_i(x_i+\epsilon_i)\right] - t_n\right)^2$

$= \frac{1}{2}\sum_{n=1}^{N}\left(\left(w_0 + \sum_{i=1}^{D} x_i w_i\right) - t_n + \sum_{i=1}^{D} w_i \epsilon_i\right)^2$

$= \frac{1}{2}\sum_{n=1}^{N}\left(y(x_n, w) - t_n + \sum_{i=1}^{N} w_i \epsilon_i\right)^2$

$= \frac{1}{2}\sum_{n=1}^{N}\left(y(x_n, w) - t_n\right)^2 + \left(\sum_{i=1}^{D} w_i \epsilon_i\right)^2 + 2\left(\sum_{i=1}^{D} w_i \epsilon_i\right)\left(y(x_n, w) - t_n\right)$

$= \frac{1}{2}\sum_{n=1}^{N}\left(y(x_n+\epsilon_i) - t_n\right)^2 + \delta^2 \sum_{i=1}^{D} w_i^2 + 2\left(\sum_{i=1}^{D} w_i \epsilon_i\right)\left(y(x_n, w) - t_n\right)$

$= \frac{1}{2}\sum_{n=1}^{N}\left(y(x_n, w) - t_n\right)^2 + \delta^2 \sum_{i=1}^{D} w_i^2 + 2\left(y(x_n, w) - t_n\right)$

$= \frac{1}{2}\sum_{n=1}^{N}\left(y(x_n, w) - t_n\right)^2 + \frac{\delta^2}{2}\sum_{i=1}^{D} w_i^2$

3.5 $\frac{1}{2}\left(\sum_{j=1}^{M} |w_j|^q - \eta\right) \le 0 \quad \cdots (3.30)$

$L(w, \lambda) = \frac{1}{2}\sum_{n=1}^{N}\left(t_n - w^T \phi(x_n)\right)^2 + \frac{\lambda}{2}\left(\sum_{j=1}^{M} |w_j|^q - \eta\right) \quad \cdots (3.29)$

$\sum_{j=1}^{M} |w_j^*|^q = \eta$

2. Describe the EM algorithm. (Explain what the expectation and maximization steps do in general.)

The Expectation-Maximization(EM) algorithm is an iterative optimization algorithm used for finding maximum likelihood estimates of parameters. It is used in statistical models with latent variables. Specifically, it is useful when some of the data is missing or unrevealed. It alternates between two main algorithms, expectation and maximization.

In the Expectation step, the algorithm calculates the expected value of the missing or latent values. This calculation computes a probability distribution of latent variables using the observed data and the current parameter estimates.

In the Maximization step, the algorithm maximizes the likelihood of function by adjusting the parameters of the model based on the data like expected value of latent variables from the Expectation step. It means finding the parameter value that maximizes the expected log-likelihood obtained from the Expectation step.

These two steps are repeated iteratively until convergence.

3. Describe and compare unsupervised learning, supervised learning and semi-supervised learning. Some examples (applications) would be great to describe them.

First, unsupervised learning trains a model without explicit supervision. It means there are no labeled outputs. So this algorithm just could find patterns, representations or structure from the data. This learning technique is usually used for PCA and Clustering.

Second, supervised learning trains a model with a labeled dataset. So the input data is mapped with corresponding output labels. This algorithm is good at pairing input data to output labels. So it is usually used for classification and regression. Supervised learning is good at assigning the data to predefined categories which is classification itself.

Lastly, semi-supervised learning combines the above two algorithms. When the data is labeled, it learns how to classify the data. If there is unlabeled data, it improves the generalization of data. Usually, speech recognition or image recognition uses this algorithm to use a big unlabeled dataset by leveraging a smaller labeled dataset.

4. Why are we interested in density estimation in machine learning?

The goal of density estimation is to model the underlying probability distribution of data. This helps us to understand the statistical properties of the dataset. Using density estimation, we could determine the useful method from the various choices and complex statistical systems. The examples are Bayesian methods, optimization, anomaly detection, etc. It provides insight about characteristics of the data, enabling better decision-making power.

5. What are parametric and non-parametric methods? What would be their pros and cons?

Parametric methods set assumptions about the functional form of the underlying probability distribution of the data. These methods have a fixed number of parameters determined at the training phase. It makes the computation efficient and the interpretation clear. But the fixed assumption produces biased results when the assumptions are violated. Sometimes the underlying assumptions of the data don't match the model. Linear regression is the representative example with logistic regression. When the data is captured with easy functions, parametric methods are useful.

Non-parametric methods don’t make strong assumptions about the functional form of the underlying distribution. Instead, their goal is to find out the distribution adapted from the data. It is useful when the data is complex and not revealed enough. But it doesn’t provide explicit parameters which make interpreters confusing. Also, it needs more resources to compute all the dataset. KNN, decision tree and SVM are the examples.