Chapter XIX - Learning Probabilistic Models

19.1 Statistical Learning

Statistical learning refers to the process of learning from data in which statistical methods are used to model the underlying distributions or relationships between variables. In this approach, the goal is to use data to estimate the probability distribution of certain variables or predict future outcomes based on observed patterns.

Key Concepts:

- **Supervised Learning**: The system learns from labeled data, with input-output pairs used to build a model that maps inputs to outputs. Common methods include regression and classification.
- **Unsupervised Learning**: The system learns from data that lacks explicit labels. Here, the goal is to uncover hidden patterns or groupings in the data, using techniques like clustering or dimensionality reduction.
- **Probability and Likelihood**: Statistical models estimate the likelihood of different outcomes given observed data, helping to make predictions or decisions. Maximum likelihood estimation (MLE) is often used to find parameters that maximize the probability of observed data.
- **Bayesian Learning**: A form of statistical learning where probability distributions are used to model uncertainty. Bayesian learning updates beliefs (or models) based on new data, incorporating prior knowledge and evidence.

Statistical learning is foundational in machine learning and provides a rigorous framework for making predictions, understanding uncertainties, and modeling complex relationships in data.

19.2 Learning with Complete Data

Learning with complete data assumes that all variables and observations are fully observed. In such cases, the goal is to build probabilistic models that can directly use this complete data to make predictions, estimate parameters, or classify instances.

Key Concepts:

- **Maximum Likelihood Estimation (MLE)**: In complete data scenarios, MLE is used to estimate the parameters of a probabilistic model. The likelihood function is derived from the

data, and the parameters are chosen to maximize the likelihood of observing the data.

- **Bayesian Inference**: In addition to MLE, Bayesian methods can be used to estimate model parameters by considering prior distributions over parameters and updating them with observed data. This approach accounts for uncertainty in the parameter estimates.
- **Gaussian Mixture Models (GMM)**: A probabilistic model often used for clustering, where the data is assumed to come from a mixture of several Gaussian distributions. Learning a GMM involves estimating the parameters of the Gaussian distributions using the complete data.
- **Multivariate Distributions**: In learning with complete data, multivariate distributions (such as multivariate normal distributions) may be used to model the relationship between multiple variables, capturing complex dependencies.

Learning with complete data is typically easier because there is no need to handle missing values or hidden variables. However, real-world data often contains missing or incomplete information, which requires more advanced techniques.

19.3 Learning with Hidden Variables: The EM Algorithm

When dealing with **incomplete data** or **hidden variables**, the system cannot directly observe all the relevant variables or outcomes. In these cases, the **Expectation-Maximization (EM) Algorithm** is widely used to learn probabilistic models.

Key Concepts:

- **Hidden Variables**: These are unobserved variables that influence the observed data. For example, in clustering, the cluster assignment of each data point is often treated as a hidden variable, as we do not directly observe which cluster a point belongs to.
- **Expectation-Maximization (EM) Algorithm**: The EM algorithm is an iterative method for finding maximum likelihood estimates of parameters in probabilistic models, especially when the model involves hidden or latent variables.
    1. **E-Step (Expectation Step)**: In this step, the algorithm computes the expected value of the hidden variables, given the current estimates of the parameters. This is usually done by using the posterior distribution of the hidden variables.
    2. **M-Step (Maximization Step)**: The algorithm then maximizes the likelihood function with respect to the parameters, using the expected values of the hidden variables computed in the E-step.
    3. **Iterate**: The E-step and M-step are alternated until convergence is reached, meaning that the estimates of the parameters stabilize.

The EM algorithm is particularly useful when the data is incomplete or when we need to infer the structure of the data, such as in **Gaussian Mixture Models** or **Hidden Markov Models (HMMs)**.

Example:

Consider a **Gaussian Mixture Model (GMM)** where each data point is assumed to come from one of several Gaussian distributions, but the cluster assignments (i.e., which Gaussian distribution each point comes from) are hidden. The EM algorithm can be used to estimate the parameters of the Gaussian distributions and the cluster assignments.

Summary

- **Statistical Learning**: Involves using statistical methods to model the relationships between variables, making predictions and decisions based on observed data. It includes both supervised and unsupervised learning techniques.
- **Learning with Complete Data**: Refers to situations where all relevant variables are observed, making it easier to estimate models using techniques like **Maximum Likelihood Estimation** and **Bayesian Inference**.
- **Learning with Hidden Variables**: In cases where not all variables are observable, the **Expectation-Maximization (EM) Algorithm** is used to estimate the parameters of a model by iterating between inferring hidden variables and maximizing the likelihood of the data.

Exercises

1. **Statistical Learning Exercise**:
   - Implement a supervised learning model (e.g., linear regression) to predict the price of a house based on its features. Use a training dataset to estimate the parameters using Maximum Likelihood Estimation.
2. **Learning with Complete Data Exercise**:
   - Use a Gaussian Mixture Model to fit a set of complete data points and estimate the parameters of the Gaussian distributions. Visualize the resulting clusters.
3. **EM Algorithm Exercise**:
   - Implement the **Expectation-Maximization** algorithm for a simple **Gaussian Mixture Model (GMM)**. Generate synthetic data with missing labels (hidden variables), and use EM to estimate the model parameters and assign cluster labels.
4. **Application Exercise**:

- Given a dataset with missing values, describe how you would apply the **EM algorithm** to handle the missing data and learn the parameters of a **Hidden Markov Model (HMM)** or **Gaussian Mixture Model (GMM)**.