Weronika Jopek, 912400002

Learning From Data

Exercise 1

a) Machine Learning Approaches

Supervised learning involves training a machine from labeled data (the data that has been tagged with a correct answer or classification). The machine learns the relationship between inputs and outputs and then the trained machine can make predictions on new, unlabeled data. Supervise learning is classified into two categories of algorithms: regression (when the output variable is a real value) and classification (when the output variable is a category). Supervised learning is well-suited for tasks where the desired output is known.

Unsupervised learning is a type of machine learning that allows the model to discover patterns and relationships in unlabeled data. The task of the machine is to group unsorted information according to similarities, patterns and differences without any prior training of data. There are two categories of algorithms: clustering (where you want to discover the inherent groupings in the data by purchasing behavior) and association (where you want to discover rules that describe large portions of your data). Unsupervised learning is well-suited for tasks where the desired output is unknown.

Reinforcement Learning is a branch of machine learning focused on making decisions to maximize cumulative rewards in a given situation. It operates on the principle of learning optimal behavior through trial and error. The agent takes actions within the environment, receives rewards or penalties, and adjusts its behavior to maximize the cumulative reward. RL is all about making decisions sequentially, the output depends on the state of the current input and the next input depends on the output of the previous input.

b) The training and testing phases in supervised learning

Training phase involves feeding the algorithm labeled data, where each point is paired with its correct output. The algorithm learns to identify patterns and relationships between the input and output data.

Testing phase involves feeding the algorithm new, unseen data and evaluating its ability to predict the correct output based on the learned patterns.

Exercise 2

1) K-Nearest Neighbour

Let X be the training dataset with n data points, where each data point is represented by a d-dimensional feature vector Xi and Y be the corresponding labels or values for each data point in X. Given a new data point x, the algorithm calculates the distance between x and each data point Xi in X using a distance metric, such as Euclidean distance:

$$distance(x, X_i) = \sqrt{\sum_{j=1}^{d} (x_j - X_{i_j})^2}$$

The algorithm selects the K data points from X that have the shortest distances to x. For classification tasks, the algorithm assigns the label y that is most frequent among the K nearest neighbors to x. For regression tasks, the algorithm calculates the average or weighted average of the values y of the K nearest neighbors and assigns it as the predicted value for x.

The value of k is very crucial in the KNN algorithm to define the number of neighbours in the algorithm. The value of K in the K-Nearest Neighbors algorithm should be chosen based on the input data. If the input data has more outliers or noise, a higher value of k would be better. It is recommended to choose an odd value for k to avoid ties in classification. Cross-validation methods can help in selecting the best k value for the given dataset.

2) Gaussian Naive Bayes

Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of conditional independence between every pair of features given the value of the class variable. Bayes' theorem states the following relationship, given class variable *y* and dependent feature vector x_1 through x_n,:

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n \mid y)}{P(x_1, \ldots, x_n)}$$

Gaussian Naive Bayes is a type of Naive Bayes method where continuous attributes are considered and the data features follow a Gaussian distribution throughout the dataset.

Gaussian Naive Bayes assumes that the likelihood $P(x_i \mid y)$ follows the Gaussian Distribution for each x_i within yk. Therefore,

$$P(x_i \mid y) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

To classify each new data point x the algorithm finds out the maximum value of the posterior probability of each class and assigns the data point to that class.

3) Principal Component Analysis

PCA technique works on the condidtion that while data in a higher dimensional space is mapped to data in a lower dimension space, the variance of the data in the lower dimensional space should be maximum. The main goal of PCA is to reduce the dimensionality of a dataset while preserving the most important patterns or relationships between the variables without any prior knowledge of the target variables.

First we nedd to standardize dataset to ensure that each variable has a mean of 0 and a standard deviation of 1:

$$Z = \frac{X - \mu}{\sigma}$$

where μ is mean of independent features and σ is the standard deviation of indepentent features.

Then we computate Covariance Matrix. Covariance measures the strength of joint variability between two or more variables, indicating how much they change in relation to each other. To find the covariance we can use the formula:

$$cov(x1, x2) = \frac{\sum_{i=1}^{n}(x_{1_i} - \overline{x_1})(x_{2_i} - \overline{x_2})}{n-1}$$

The value of covariance can be positive (as the x1 increases x2 also increases), negative (as the x1 increases x2 also decreases), or zeros (no direct direction).

Then we have to compute Eigenvalues and Eigenvectors of Covariance Matrix.

Let A be a square *n*x*n* matrix, X be a non-zero vector, λ is a matrix of scalar values, I is the identity matrix of the same shape as matrix A.

$$AX = \lambda X$$

$$AX - \lambda X = 0$$

$$(A - \lambda I)X = 0$$

$$|A - \lambda I| = 0$$

From this equation, we can find the eigenvalues, and therefore corresponding eigenvector.

Resources:

https://www.geeksforgeeks.org/supervised-machine-learning/#how-supervised-machine-learning-works

https://www.geeksforgeeks.org/what-is-reinforcement-learning/

https://www.geeksforgeeks.org/supervised-unsupervised-learning/

https://raw.githubusercontent.com/wwkenwong/book/master/Simon%20Rogers%2C%20Mark%20Girolami%20A%20First%20Course%20in%20Machine%20Learning.pdf

https://scikit-learn.org/1.5/modules/naive_bayes.html

https://www.geeksforgeeks.org/gaussian-naive-bayes/

https://www.geeksforgeeks.org/principal-component-analysis-pca/