**COMPUTER SCIENCE**

**COM 4138-1**

**MACHINE LEARNING**

**CAT**

**GROUP PRESENTATION**

|  |  |
| --- | --- |
| **JOHN LESHAN** | **SB06/JR/MN/14072/2022** |
| **COLLINS KOSGEI** | **SB06/JR/MN/14066/2022** |
| **NYAMBURA DANIEL MWANGI** | **SB06/JR/MN/16342/2022** |
| **DENNIS MWANZIA** | **SB06/SR/MN/7952/2018** |
| **ALLOYS KENNEDY ODUOR** | **SB06/JR/MN/14085/2022** |

**Question 1: (6 marks)**

1. **Differentiate between supervised and unsupervised learning. Provide examples of each.** (3 marks)

Supervised learning uses labeled datasets to train algorithms to classify data or predict outcomes. As input data or predict outcomes. As input data is inputted into the model, its weight modify until it fits into the model; this process is known as cross validation which ensures the model is not over fitted or under fitted.

Example;

Classification: Email spam detection (classify emails as “spam” or “not spam”).

Unsupervised learning analyses and clusters unlabeled datasets using machine learning methods. The algorithms find hidden patterns or data groupings without human interaction. This method is useful for exploratory data analysis, cross-selling, consumer segmentation, and image and pattern recognition.

Examples:

Clustering: Customer segmentation (grouping customers by purchasing behavior).

**(b) Explain the concept of overfitting in machine learning models and how it can be mitigated.** (3 marks)

Overfitting occurs when a machine learning model is too complex and fits the training data too complex and fits the training data too closely. This can lead to poor performance on new, unseen data because the model is too specialized to the training dataset. To prevent overfitting, it is important to use a validation dataset to evaluate the model’s performance and to use regularization techniques to simplify model.

**Question 2: (8 marks)**

1. Briefly describe the working principles of the following algorithms:  
   **i. Linear Regression** (2 marks)

Linear Regression is a supervised learning algorithm used for predicting a continuous output variable (dependent variable) based on the relationship between one or more independent variables. It assumes a linear relationship between the input features and the output.

**Working Principle:**

1. **Model Representation**:
   * The algorithm models the relationship as: y=β0+β1x1+β2x2+⋯+βnxn+ϵy = \beta\_0 + \beta\_1x\_1 + \beta\_2x\_2 + \dots + \beta\_nx\_n + \epsilony=β0​+β1​x1​+β2​x2​+⋯+βn​xn​+ϵ where yyy is the predicted output, xix\_ixi​ are input features, βi\beta\_iβi​ are coefficients, and ϵ\epsilonϵ is the error term.
2. **Objective**:
   * Find the coefficients (β\betaβ) that minimize the error between the predicted values and the actual values in the training data.
   * The most common error metric is **Mean Squared Error (MSE)**: MSE=1n∑i=1n(yi−y^i)2\text{MSE} = \frac{1}{n} \sum\_{i=1}^{n}(y\_i - \hat{y}\_i)^2MSE=n1​i=1∑n​(yi​−y^​i​)2
3. **Training Process**:
   * Use optimization techniques (e.g., Gradient Descent or the Normal Equation) to compute the best-fit line or hyperplane by minimizing MSE.
4. **Prediction**:
   * Once trained, the model can predict yyy for unseen xxx values using the learned coefficients.

**Applications:**

* Predicting house prices based on features like area, number of rooms, etc.
* Forecasting stock prices or sales.

**ii. K-Nearest Neighbors** (2 marks)

K-Nearest Neighbors (KNN) is a supervised learning algorithm used for both classification and regression. It is a non**-**parametric method, meaning it makes no assumptions about the underlying data distribution.

**Working Principle:**

1. **Data Representation**:
   * Each data point in the training set is plotted in feature space.
   * The algorithm stores all training data and uses it during prediction.
2. **Classification**:
   * For a new data point, the algorithm identifies the kkk closest data points (neighbors) using a distance metric such as:
     + **Euclidean Distance**: d(p,q)=∑i=1n(pi−qi)2d(p, q) = \sqrt{\sum\_{i=1}^{n} (p\_i - q\_i)^2}d(p,q)=i=1∑n​(pi​−qi​)2​
     + Manhattan or Minkowski distances can also be used.
   * It assigns the class label that is most frequent among these kkk neighbors (majority voting).
3. **Regression**:
   * For regression tasks, KNN predicts the output as the average of the outputs of the kkk nearest neighbors.
4. **Key Parameters**:
   * **Value of kkk**: Determines the number of neighbors considered.
   * **Distance Metric**: Determines how neighbors are measured.
5. **Training Process**:
   * No explicit training phase; all computations are deferred to prediction (lazy learning).
6. **Prediction**:
   * The algorithm searches for neighbors and applies the majority vote (classification) or averaging (regression).

**Applications:**

* Handwritten digit recognition (classification).
* Predicting house prices based on nearby property values (regression).

b) Machine learning is used in recommendation systems. Explain how collaborative filtering works in building recommendations. (4 marks)

Collaborative filtering is a machine learning technique employed by recommendation systems to identify the preference of a user based on the historic interactions or preferences of a user group. It assumes that users with similar preferences in the past will share similar preferences in the future also.

**How Collaborative Filtering Works**

There are two major kinds of collaborative filtering: **User-based** and **Item-based.**

**1. User-Based Collaborative Filtering.**

This technique finds similar users and recommends items liked or interacted with by similar users.

The following are theSteps:

1. **Create a User-Item Interaction Matrix:**

A matrix is created in which rows represent users, columns represent items, and entries represent the interaction of a user with an item. Examples include ratings, purchases, clicks, etc.

Example:

Mathematically;

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | User | Item | Item A | Item B | Item C | | User 1 | | 5 | 4 | ? | | User 2 | | 3 | ? | 5 | | User 3 | | 3 | 2 | 4 | |  | | | | | |

Here, "?" denotes unknown preferences.

1. **Similarity Computation:**

It Calculate the similarity between users using metrics such as:

* **Cosine Similarity:**

Sim (u,v) = u.v / ||u|| ||v|| ​

* **Pearson Correlation Coefficient**:

It determines the strength of a linear relationship between the preferences of users.

1. **Recommendation:**

It finds similar users for a target user and recommends items that those users have interacted with but the target user has not.

**2. Item-Based Collaborative Filtering**

It takes into consideration item similarity, rather than user similarity. Items that are similar to what a user liked are recommended.

**The following are the Steps:**

1. **Calculate the Item-Item Similarity Matrix:**

Similarity between two items is obtained by comparing what users have selected/liked/rated about them.

1. **Similarity Computation**:

Compute the similarity between two items using such metrics as cosine similarity.

1. **Recommendation:**

It gives a target user recommendation of items similar to his/her already interacted-with items.

**Advantages**

* **Scalable:** Scales well with large datasets.
* **Domain Agnostic**: It does not require item-specific information.

**Challenges**

* **Cold Start Problem**: It is hard to make recommendations for new users or items with no interaction record available.
* **Data Sparsity**: The user-item interaction matrix usually has a very big sparsity which will be a nightmare while calculating the similarity.

**Applications**

* **Movie Recommendation:** Netflix, IMDB.
* **E-commerce**: Amazon, eBay
* **Music Recommendation**: Spotify, Pandora.

**Question 3: (6 marks)**

a) Discuss the importance of feature scaling in machine learning and compare two common methods used: Min-Max Scaling and Standardization. (3 marks)

**Feature Scaling:** Feature scaling is a crucial preprocessing step in machine learning. It ensures that the features (input variables) in a dataset have the same scale or range, which is essential for many machine learning algorithms. Without scaling, algorithms that use distance metrics (e.g., K-Nearest Neighbors, Support Vector Machines) or gradient-based optimization (e.g., neural networks, linear regression) may not perform well because some features with larger numeric ranges can dominate others, leading to biased results or poor convergence. Scaling helps in speeding up the convergence of algorithms and leads to better model performance.

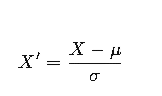
**Why Feature Scaling is Important:**

1. **Improved Model Performance:** Machine learning models, especially those that rely on calculating distances (like KNN or SVM) or gradients (like logistic regression or neural networks), can be sensitive to differences in feature scales. Features with larger ranges can overshadow smaller ones, leading to skewed results.
2. **Faster Convergence:** Many optimization algorithms (such as gradient descent) converge faster when features are scaled, reducing the number of iterations needed to find an optimal solution.
3. **Prevents Numerical Instability:** Models that use mathematical computations may face numerical instability when features are on very different scales, leading to inaccurate results.
4. **Ensures Equal Weight:** Feature scaling ensures that all features contribute equally to the model, preventing features with larger magnitudes from having an outsized influence.

**Common Methods of Feature Scaling:** Two widely used methods of feature scaling are **Min-Max Scaling** and **Standardization**. Both methods are designed to transform the data into a specific range or distribution, but they differ in their approach.

1. **Min-Max Scaling (Normalization):**

* **Explanation:** This method scales each feature to a specific range, often [0, 1]. It works by subtracting the minimum value of the feature and dividing by the range (maximum - minimum) of the feature. The result is that all values are transformed to fit within the desired range.
* **When to Use:** Min-Max scaling is effective when the data is known to follow a specific range, especially when the features are bound within a known minimum and maximum. It is sensitive to outliers, as the scaling is based on the minimum and maximum values of the data.
* **Advantage:**
* Simple and effective.
* Retains the original distribution of the data.
* **Disadvantage:**
* Sensitive to outliers, which can skew the scaling process.
* Not suitable for datasets with a wide range of values.

1. **Standardization (Z-score Normalization):**

**Formula:**

Where μ is the mean of the feature and σ is the standard deviation.

Standardization transforms the data into a distribution with a mean of 0 and a standard deviation of 1. This ensures that the data has no units, and it scales the data based on how far the individual data points are from the mean.

Standardization is ideal when the dataset follows a normal distribution or when there are no specific bounds on the features. It is less sensitive to outliers compared to Min-Max scaling because it uses mean and standard deviation rather than the range.

Advantages

* Less sensitive to outliers.
* Useful when data has an unknown or varying range.

**Disadvantages:**

* Does not guarantee that the scaled data will lie within a specific range (e.g., [0, 1]).
* May not work well with highly skewed data.

**Comparison:**

* **Outliers:** Min-Max scaling is highly affected by outliers because the scale depends on the minimum and maximum values. In contrast, standardization is more robust to outliers since it relies on mean and standard deviation.
* **Range of Scaled Data:** Min-Max scaling forces the data to lie within a specific range, while standardization does not impose a fixed range on the data.
* **Distribution Assumptions:** Standardization works best when the data follows a normal distribution, whereas Min-Max scaling does not make any assumptions about the data distribution.

### b) Explain the role of training, validation, and test datasets in machine learning projects. (3 marks)

* **Training Dataset:** This is where the model learns and adapts, making it essential for building a functioning model. If too small, the model may not learn adequately; if too large, it might overfit, especially if not complemented with validation and test sets.
* **Validation Dataset:** Used for tuning the model's hyperparameters and avoiding overfitting. It allows for model improvement without using the test data, which ensures that the evaluation remains objective.
* **Test Dataset:** The test dataset is crucial for final model evaluation. It ensures that the model's performance is not overly optimistic and provides an accurate measure of its predictive ability in real-world scenarios

**Question 4: (10 marks)**

**Case Study Scenario:**  
An e-commerce company wants to build a machine learning model to predict customer churn. The company has historical customer data, including demographics, purchase history, customer support interactions, and feedback.

**Tasks:**  
**a) Identify the type of machine learning problem the company is solving. Justify your answer. (2 marks)**

This is a supervised learning problem and, more specifically, a classification problem. The goal is to predict whether a customer will churn (leave the company) or not, which is a binary outcome (e.g., "Yes" or "No"). Supervised learning is suitable because the company has historical data with labeled examples (e.g., customers who have churned in the past). By training the model on this labeled data, it can learn patterns and make predictions for new customers  
b) Outline the steps involved in preparing the data for this model. (4 marks)

Preparing data is critical for the success of a machine learning model. Below are the steps:

Understand and Explore the Data:

Begin by analyzing the dataset to understand the types of features available (e.g., demographics, purchase history). Check for imbalances in the target variable (e.g., churn rates) and visualize relationships between features to gain insights.

Clean the Data:

Handle missing values by imputing them (e.g., using the mean for numerical data or the mode for categorical data) or removing rows/columns with excessive missing data.

Remove duplicates and outliers, which could distort model performance.

Encode Categorical Variables:

Since machine learning algorithms work with numerical data, convert categorical features like "customer region" or "feedback" into numerical format. Techniques include one-hot encoding for non-ordinal data or label encoding for ordinal data.

Feature Scaling:

Normalize or scale numerical features (e.g., purchase amounts or number of interactions) to ensure all features contribute equally to the model, especially for algorithms sensitive to magnitude, like logistic regression.

Feature Engineering:

Create new features that may add predictive power, such as the average purchase frequency or the time since the last purchase. Additionally, consider reducing irrelevant features using feature selection techniques.

Split the Data:

Divide the dataset into training and testing subsets (e.g., 80-20 split) to evaluate model performance on unseen data.

**c) Recommend two suitable machine learning algorithms for this task and explain why they would be effective. (4 marks)**

1. **Logistic Regression**:○ Logistic regression is a simple yet effective algorithm for binary classification problems like customer churn.

○ It provides interpretable results, as the model coefficients can indicate the importance of each feature.

○ It performs well when the relationship between features and the target variable is linear and is relatively fast to train.

2. **Random Forest**:○ Random forest is a powerful ensemble method that combines multiple decision trees to make predictions.

○ It handles non-linear relationships and works well with a mix of categorical and numerical data.

○ It is robust to overfitting, especially with a sufficient number of trees, and can also handle feature importance analysis to identify key drivers of churn.

**References**:

1. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning*.
2. Towards Data Science: [Linear Regression Explained](https://towardsdatascience.com/).
3. Alpaydin, E. (2021). *Introduction to Machine Learning*.
4. Scikit-learn Documentation: KNN Algorithm.
5. Analytics Vidhya: [Understanding KNN](https://www.analyticsvidhya.com/).
6. Ricci, F., Rokach, L., & Shapira, B. (2015). *Recommender Systems Handbook*.
7. Koren, Y., Bell, R., & Volinsky, C. (2009). *Matrix Factorization Techniques for Recommender Systems*.
8. TutorialsPoint. *Machine learning tutorial.* TutorialsPoint. Retrieved from https://www.tutorialspoint.com/machine\_learning
9. Towards Data Science: [Collaborative Filtering Explained](https://towardsdatascience.com/).
10. Scikit-learn Documentation: Recommender Systems.