Module Title	**Machine Learning**
Level	**7**
Credit	**10**
Module Type	**Optional**
Delivery Mode	**Lectures, hands-on labs, group work, user testing sessions, online resources**
Assessment Method	**Practical Assignment, Individual project, Project Presentation and Demonstration**
Module Aim	
This module introduces the foundational and advanced concepts of Machine Learning. Students will explore data preprocessing, model development, evaluation metrics, and real-world application scenarios, with an emphasis on ethical and legal practices in the use of AI models.	

Project – 60%	**This component involves an in-depth, end-to-end machine learning project, where students apply learned techniques to a real-world dataset:** ● **Dataset Selection: Choose a relevant, real-world dataset based on a specific application area (e.g., healthcare, finance, marketing)** ● **Data Exploration and Preprocessing: Perform exploratory data analysis (EDA), identifying patterns, relationships, and preparing the data through cleaning, transformation, and feature engineering** ● **Model Development and Training: Implement machine learning models (classification, regression, clustering, etc.), select appropriate algorithms, and fine-tune using cross-validation, hyperparameter optimization, and regularization techniques** ● **Evaluation and Refinement: Evaluate models using suitable metrics (e.g., ROC curve, accuracy, F1-score) and refine the models by adjusting hyperparameters or applying ensemble methods**

	● **Final Report and Presentation: Document findings in a comprehensive technical report, including methodology, results, and discussions. Conclude with recommendations for further research or potential improvements** ● **Project Presentation: Deliver a final presentation summarizing the problem, methodology, findings, and insights from the model to the class, simulating a professional setting** **Learning Outcomes Mapped: LO2, LO3, LO4**

Libraries covered:

Numpy, Pandas, Matplotlib, Seaborn, Scikit Learn

Case Study1

Predicting Store Sales using Linear Regression and Decision Tree

Objective

You are provided with historical sales data of 1,115 Rossmann stores. Your task is to build predictive models to forecast sales based on a few relevant features using:

- **Linear Regression**
- **Decision Tree Regressor**

Dataset

Rossmann operates over 3,000 drug stores in 7 European countries. Currently, Rossmann store managers are tasked with predicting their daily sales for up to six weeks in advance. Store sales are influenced by many factors, including promotions, competition, school and state holidays, seasonality, and locality. With thousands of individual managers predicting sales based on their unique circumstances, the accuracy of results can be quite varied.

In their first Kaggle competition, Rossmann is challenging you to predict 6 weeks of daily sales for 1,115 stores located across Germany. Reliable sales forecasts enable store managers to create effective staff schedules that increase productivity and motivation. By helping Rossmann create a robust prediction model, you will help store managers stay focused on what's most important to them: their customers and their teams!

Use the **"Rossmann Store Sales"** dataset from Kaggle:
https://www.kaggle.com/competitions/rossmann-store-sales/data

Use only the RossmannSalesData.csv file for this case study.

Features

Most of the fields are self-explanatory. The following are descriptions for those that aren't.

- **Id** - an Id that represents a (Store, Date) duple within the test set
- **Store** - a unique Id for each store
- **Sales** - the turnover for any given day (this is what you are predicting)
- **Customers** - the number of customers on a given day

- **Open** - an indicator for whether the store was open: 0 = closed, 1 = open
- **StateHoliday** - indicates a state holiday. Normally all stores, with few exceptions, are closed on state holidays. Note that all schools are closed on public holidays and weekends. a = public holiday, b = Easter holiday, c = Christmas, 0 = None
- **SchoolHoliday** - indicates if the (Store, Date) was affected by the closure of public schools
- **StoreType** - differentiates between 4 different store models: a, b, c, d
- **Assortment** - describes an assortment level: a = basic, b = extra, c = extended
- **CompetitionDistance** - distance in meters to the nearest competitor store
- **CompetitionOpenSince[Month/Year]** - gives the approximate year and month of the time the nearest competitor was opened
- **Promo** - indicates whether a store is running a promo on that day
- **Promo2** - Promo2 is a continuing and consecutive promotion for some stores: 0 = store is not participating, 1 = store is participating
- **Promo2Since[Year/Week]** - describes the year and calendar week when the store started participating in Promo2
- **PromoInterval** - describes the consecutive intervals Promo2 is started, naming the months the promotion is started anew. E.g. "Feb,May,Aug,Nov" means each round starts in February, May, August, November of any given year for that store

==========

Understanding the Data

Q1. Load the train.csv dataset and display the first 5 rows.
Q2. What is the shape of the dataset?
Q3. Which columns contain missing values? How will you handle them?

Data Filtering and Feature Selection

Q4. Filter out rows where the store was closed (Open == 0) or sales are 0. Why is this step important?

Q5. From the available features, select the following for modeling:

- Store
- DayOfWeek
- Promo

- SchoolHoliday

Why are these features relevant?

Q6. Encode any categorical features (e.g., DayOfWeek) appropriately.

Preparing Data for Modeling

Q7. Split the dataset into:

- Feature matrix X (selected features)
- Target variable y (Sales column)

Q8. Split the data into training and testing sets (e.g., 80/20 split).

Q9. Scale the features using StandardScaler for Linear Regression. Why is scaling important?

Model Building and Evaluation

Q10. Train a **Linear Regression** model using the scaled training data.

Q11. Train a **Decision Tree Regressor** with a max depth of 6 (no scaling needed here).

Q12. Predict on the test set using both models.

Q13. Calculate the **Mean Squared Error (MSE)** and **R^2 Score** for both models.
Which model performs better, and why?

Visualization and Interpretation

Q14. Plot **Actual vs Predicted Sales** for both models. What do these plots tell you?

Q15. Plot the **residuals** (actual - predicted) for Linear Regression. Do they appear normally distributed?

Q16. Which model generalizes better? Which is more interpretable?

Final Questions

Q17. What business insights can you draw from this analysis?

Q18. What limitations exist in this modeling approach?

Q19. How could you improve the model further? (e.g., more features, time series models, etc.)

Case Study 2:

Predicting Employee Attrition using Logistic Regression & Decision Tree

Objective

You are an HR analyst. Your goal is to predict whether an employee is likely to leave the company using historical HR data. You'll compare two classification models:

- **Logistic Regression**
- **Decision Tree Classifier**

Dataset

IBM HR Analytics Employee Attrition & Performance

Download IBMHRData.csv

Understanding the Data

Q1. Load the dataset and display the first 5 rows.
Q2. What are the key features in this dataset?
Q3. What is the target variable? What are its possible values?

Q4. How many employees left the company vs stayed? (Use value counts on the target variable.)

Data Preprocessing

Q5. Check for missing values. Are there any?
Q6. Convert categorical columns (like Attrition, Gender, Department, etc.) into numeric form. What encoding method will you use?

Q7. Drop irrelevant columns like EmployeeNumber, Over18, and EmployeeCount. Why?

Q8. Select meaningful features for prediction (at least 8–10). Which ones did you choose and why?

Splitting and Scaling

Q9. Split the dataset into feature matrix X and target variable y.

Q10. Perform train-test split (e.g., 80% training, 20% testing).

Q11. Apply feature scaling using StandardScaler. Why is this important for Logistic Regression?

Modeling & Evaluation

Q12. Train a **Logistic Regression** model and make predictions.
Q13. Train a **Decision Tree Classifier** with a max_depth=4 and make predictions.

Q14. Evaluate both models using:

- Accuracy
- Confusion Matrix
- Precision, Recall, F1 Score
- ROC AUC Score

Which model performs better overall?

Visual Analysis

Q15. Plot the **confusion matrix** for both models. What do they tell you?
Q16. Plot the **ROC Curve** for both models. Which has a better AUC?

Interpretation

Q17. What are the most important features in predicting attrition (use .coef_ for Logistic, .feature_importances_ for Decision Tree)?

Q18. What does the Logistic Regression model tell you about the odds of attrition for certain features like OverTime or JobSatisfaction?

Q19. What business insights can HR managers gain from this model?

Q20. Suggest 2-3 ways the company could reduce attrition based on your findings.

Case Study 3:

Predicting Credit Default Risk using Logistic Regression & Decision Tree

Objective

As a financial analyst at a lending institution, your job is to predict whether a loan applicant is likely to **default on credit** based on historical data. You will compare two classification algorithms:

- **Logistic Regression**
- **Decision Tree Classifier**

Dataset

Banks play a crucial role in market economies. They decide who can get finance and on what terms and can make or break investment decisions. For markets and society to function, individuals and companies need access to credit.

Variable Name	Description	Data Type
SeriousDlqin2yrs	Person experienced 90 days past due delinquency or worse.	Y/N
RevolvingUtilizationOfUnsecuredLines	Total balance on credit cards and personal lines of credit except real estate and no installment debt like car loans divided by the sum of credit limits.	percentage
Age	Age of borrower in years.	integer
NumberOfTime30-59DaysPastDueNotWorse	Number of times borrower has been 30-59 days past due but no worse in the last 2 years.	integer
DebtRatio	Monthly debt payments, alimony,living costs divided by monthy gross income.	percentage
MonthlyIncome	Monthly income.	real
NumberOfOpenCreditLinesAndLoans	Number of Open loans (installment like car loan or	integer

Variable Name	Description	Data Type
	mortgage) and Lines of credit (e.g. credit cards).	
NumberOfTimes90DaysLate	Number of times borrower has been 90 days or more past due.	integer
NumberRealEstateLoansOrLines	Number of mortgage and real estate loans including home equity lines of credit.	integer
NumberOfTime60-89DaysPastDueNotWorse	Number of times borrower has been 60-89 days past due but no worse in the last 2 years.	integer
NumberOfDependents	Number of dependents in family excluding themselves (spouse, children etc.).	integer

Since the class labels are provided for the sole response variable **SeriousDlqin2yrs** taking values 0 (NO default) or 1 (default).

Credit scoring algorithms, which make a guess at the probability of default, are the method banks use to determine whether or not a loan should be granted.

File to use: GiveMeSomeCredit-training.csv

Understanding the Dataset

Q1. Load the dataset and display the first few rows.
Q2. What does each column represent? Which one is the **target variable**?

Q3. How many default and non-default cases are there? (Use value_counts())

Data Cleaning and Preprocessing

Q4. Are there any missing values? Which features have the most?
Q5. What strategy will you use to handle missing values? Apply it.

Q6. Drop any redundant columns. Which ones did you drop and why?

Q7. Check for class imbalance in the target variable. How will you address it?

Feature Selection and Encoding

Q8. Select key features that could impact credit default (e.g., RevolvingUtilizationOfUnsecuredLines, DebtRatio, MonthlyIncome, etc.)

Q9. If any features are categorical, encode them accordingly.
Q10. Scale the features using StandardScaler. Why is this important?

Train-Test Split

Q11. Split the dataset into train and test sets (80/20 split).
Q12. Print the shape of the train and test datasets.

Modeling and Evaluation

Q13. Train a **Logistic Regression** model.
Q14. Train a **Decision Tree Classifier** with max_depth=5.

Q15. Evaluate both models using:

- Accuracy
- Confusion Matrix
- Precision, Recall, F1-Score
- ROC AUC Score

Which model performs better for identifying **default risk**?

Visualization and Interpretation

Q16. Plot the **Confusion Matrix** for both models. What does it indicate?
Q17. Plot the **ROC Curve** for both models. Which has the higher AUC?

Q18. Print the **feature importances** from the Decision Tree. Which features most influence default risk?

Business Interpretation

Q19. What kind of borrowers are most likely to default, based on the analysis?
Q20. How can the lending institution reduce default rates based on these insights?