

---

## ML – Project

Your course project is an opportunity for you to explore an interesting machine learning problem of your choice in the context of a real-world data set.

You have to choose your own problem and collect your own dataset. And apply different algorithms (minimum 3 algorithms) for example: naïve bayes, support vector machine, KNN ...ect.

**Max 3 members in team.**

Projects ideas:

### **1-Customer Churn Prediction for Telecom Companies:**

Description: Develop a machine learning model to predict customer churn for telecom companies. By analyzing customer usage patterns, call records, demographics, and billing information, the model can identify customers who are likely to switch to a competitor or cancel their subscriptions. This can help telecom companies implement targeted retention strategies and reduce customer churn rates.

### **2-Movie Genre Classification:**

Description: Build a machine learning model to classify movies into different genres based on their plot summaries, cast information, and user ratings. By analyzing textual features and metadata, the model can automatically assign genres to movies, assisting movie recommendation systems and content categorization.

### **3-Credit Card Fraud Detection:**

Description: Develop a machine learning model to detect fraudulent credit card transactions. By analyzing transactional patterns, spending behavior, geographic locations, and other transaction features, the model can identify potentially fraudulent activities and prevent financial losses for credit card issuers and merchants.

### **4-Employee Attrition Prediction:**

Description: Build a machine learning model to predict employee attrition or turnover for organizations. By analyzing employee demographics, job satisfaction surveys, performance metrics, and tenure, the model can identify factors that contribute to employee churn and help HR departments implement retention strategies.

### **5-Loan Default Prediction:**

Description: Develop a machine learning model to predict loan default risk for banks and financial institutions. By analyzing applicant information, credit scores, financial history, and loan terms, the model can assess the likelihood of borrowers defaulting on their loans and help lenders make informed lending decisions.

---

## 6-Healthcare Diagnosis and Prediction:

Description: Develop a model to diagnose medical conditions or predict patient outcomes based on electronic health records, medical imaging data, or genetic information, assisting healthcare professionals in making accurate diagnoses and treatment decisions.

## 7-Sentiment Analysis for Social Media:

Description: Create a sentiment analysis system to analyze sentiment in social media posts or comments, helping companies understand public opinion, monitor brand sentiment, and identify emerging trends or issues.

## Requirements:

- **Data Collection:** The first step in building a music recommendation system is to collect a dataset of user listening history and music metadata, such as artist, genre, and album information. This data can be obtained from a music streaming platform or other relevant sources.
- **Data Preprocessing:** Once the data is collected, it needs to be preprocessed to remove any noise or irrelevant information, and to normalize the data. This can be done using Python libraries such as pandas or numpy.
- **Feature Extraction:** The next step is to extract relevant features from the music metadata and user listening history, such as play count, skip count, and time of day. This will be used to train the machine learning model.
- **Model Training:** After extracting the features, the machine learning model needs to be trained on the data to learn patterns and make recommendations. This can be done using various machine learning algorithms such as Collaborative Filtering, Content-Based Filtering, or Hybrid Filtering.
- **Testing and Evaluation:** Once the music recommendation model is trained, it needs to be tested and evaluated on a separate dataset to measure its performance and accuracy. This will help determine if the model is producing accurate and relevant recommendations.
- **Deployment:** Finally, the music recommendation system needs to be deployed on a platform where users can input their listening history and receive personalized recommendations. The system should be user-friendly and easy to navigate.

---

**To deploy** an app or desktop application using Java or any other language, you can follow these general steps:

1. **Model Development:** First, you need to develop and train your machine learning model using algorithms like Logistic Regression, Random Forest, or Gradient Boosting Machines. This involves collecting and preprocessing the data, selecting features, training the model, and evaluating its performance.
2. **Model Serialization:** Once you have trained your model and are satisfied with its performance, you need to serialize it to save it to disk. Serialization converts the model object into a byte stream that can be easily stored and retrieved.

---

### 3. Application Development:

- For a web application:
    - Use frameworks like Spring Boot, Flask, or Django to develop a web application in Java, Python, or any other suitable language.
    - Create a user interface where users can input the required data (e.g., customer information, usage patterns).
    - Implement backend logic to preprocess user input, load the serialized model, and make predictions using the model.
    - Display the churn prediction results to the user.
  - For a desktop application:
    - Use JavaFX, Swing, or Tkinter to develop a desktop application with a graphical user interface (GUI).
    - Design the GUI to include input fields for user data and a button to trigger the churn prediction.
    - Implement the backend logic to handle user input, load the serialized model, and make predictions.
    - Display the churn prediction results within the application window.
4. **Integration with Model:** Integrate the serialized machine learning model into your application code. Load the model from disk during application startup or when needed for prediction.
  5. **Testing and Validation:** Test your deployed application to ensure that it works as expected and produces accurate churn predictions. Validate the predictions against known outcomes or ground truth data.
  6. **Maintenance and Updates:** Regularly maintain and update your application to incorporate new features, bug fixes, and improvements. Monitor its performance and user feedback to identify areas for optimization.