# Toros University

## Department of Software Engineering

# PYTHON PROGRAMMING LANGUAGE: Final project

**Submitted by:**

Hamza ABDULLAH

215060051

## Dataset Overview:

The dataset consists of 7 independent variables: pH, Temperature, Taste, Odor, Fat, Turbidity, and Color, with the Grade or Quality of the milk being the target variable. The milk can be classified as Low (Bad), Medium (Moderate), or High (Good) based on these parameters. The ultimate goal of this project is to leverage the benefits of machine learning in the dairy industry by building statistical and predictive models to predict the quality of milk.

## Algorithms Overview:

In this project, we utilized three distinct machine learning algorithms to predict the quality of milk based on the provided dataset. Here's a concise overview of each algorithm:

- Random Forest Classifier:
  - Ensemble learning method combining multiple decision trees to make accurate predictions.
  - Utilizes random subsets of features and data samples during training for robustness.
  - Offers high accuracy and handles both numerical and categorical data effectively.
- Decision Tree Classifier:
  - Constructs a classification model in a tree-like structure.
  - Splits the data based on features to maximize homogeneity within resulting subsets.
  - Provides interpretability and handles various data types, including numerical and categorical.
- Multilayer Perceptron (MLP) Classifier:
  - Neural network with multiple interconnected layers.
  - Uses a feedforward mechanism to learn complex feature-target relationships.
  - Well-suited for nonlinear classification tasks and captures intricate patterns in the data.

## Overview of Evaluation Parameters and Results:

**Accuracy**: This parameter measures how often the algorithm correctly predicts the class of the data. It is calculated by dividing the number of correctly predicted instances by the total number of instances.

**Precision**: Precision measures the percentage of positive instances that were correctly predicted by the algorithm. It is calculated by dividing the number of true positive instances by the sum of true positive and false positive instances.

**Recall**: Recall measures the percentage of actual positive instances that were correctly predicted by the algorithm. It is calculated by dividing the number of true positive instances by the sum of true positive and false negative instances.

**F1 score**: F1 score is the harmonic mean of precision and recall. It provides a balanced measure of the two parameters and is calculated by taking the reciprocal of the sum of the reciprocals of precision and recall.

| Algorithm | Accuracy | Precision | Recall | F1 Score |
|---|---|---|---|---|
| Random Forest | 0.9953 | 0.9954 | 0.9953 | 0.9953 |
| Decision Tree | 0.9906 | 0.9906 | 0.9906 | 0.9906 |
| Multilayer Perceptron | 0.9953 | 0.9953 | 0.9953 | 0.9953 |

## Libraries:

| Library | Explanation | Why Used |
|---|---|---|
| pandas (pd) | A data manipulation library in Python that provides useful data structures such as data frames and series, and functions for data input/output (I/O). | Used to read and manipulate the dataset stored in a CSV file. |
| matplotlib.pyplot (plt) | A plotting library in Python that provides functions for creating a variety of visualizations. | Used for data visualization, specifically for creating scatterplots and heatmaps to explore the relationships between different features in the dataset. |
| seaborn (sns) | A visualization library in Python that provides a high-level interface for creating informative and visually appealing statistical graphics. | Used for data visualization, providing additional functionality and customization options not available in matplotlib. The 'sns.scatterplot' and 'sns.heatmap' functions are used to create visually appealing and informative plots. |
| sklearn.model_selection | A module in scikit-learn that provides functions for splitting the data into training and testing sets. | Provides the 'train_test_split' function, used to split the dataset into training and testing sets. |
| sklearn.metrics | A module in scikit-learn that provides functions for evaluating the performance of machine learning models. | Used to compute various evaluation metrics such as accuracy, precision, recall, and F1 score. |
| sklearn.preprocessing | A module in scikit-learn that provides functions for transforming data before applying machine learning algorithms. | Used to standardize the data by scaling it to zero mean and unit variance using the 'StandardScaler' function. |
| sklearn.tree | A module in scikit-learn that provides functions for building decision trees. | Used to build a decision tree classifier using the 'DecisionTreeClassifier' function. |
| sklearn.neural_network | A module in scikit-learn that provides functions for building neural networks. | Used to build a multilayer perceptron classifier using the 'MLPClassifier' function. |
| sklearn.ensemble | A module in scikit-learn that provides functions for building ensemble models, such as random forests. | Used to build a random forest classifier using the 'RandomForestClassifier' function. |

## Project overview:

In this project, we aim to predict the quality or grade of milk using machine learning techniques. To achieve this goal, we will be using the following algorithms:

- Random Forest Classifier
- Decision Tree Classifier
- Multilayer Perceptron (MLP) Classifier

The dataset used in this project contains 7 independent variables: pH, Temperature, Taste, Odor, Fat, Turbidity, and Color. The target variable is the grade of the milk, which can be classified as Low (Bad), Medium (Moderate), and High (Good).

We split the dataset into training and testing sets, and then trained the models using the training set. Next, we evaluated the performance of each model using the following evaluation metrics:

- Accuracy
- Precision
- Recall
- F1 Score

Overall, our project successfully demonstrated the usefulness of machine learning in predicting the quality of milk. With further development and optimization, these models could potentially be used to improve the efficiency and accuracy of milk quality control in the dairy industry.

# Appendix

1. Random Forest Classifier:

```
from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier(random_state=42)
rfc.fit(X_train_scaled, y_train)
```

2. Decision Tree Classifier:

```
from sklearn.tree import DecisionTreeClassifier
dt = DecisionTreeClassifier(random_state=42)
dt.fit(X_train_scaled, y_train)
```

3. Multilayer Perceptron (MLP) Classifier:

```
from sklearn.neural_network import MLPClassifier
mlp = RandomForestClassifier(random_state=42, max_iter=500)
mlp.fit(X_train_scaled, y_train)
```

Algorithms test code:

```
fromd sklearn.metrics import accuracy_score,precision_score, recall_score, f1_score

models = [rfc,dt,mlp]

model_names = ['Random Forest','Decision Tree','Multilayer Perceptron (MLP)']

for i, model in enumerate(models):

    y_pred = model.predict(X_test_scaled)

    acc = accuracy_score(y_test, y_pred)

    precision = precision_score(y_test, y_pred, average='weighted')

    recall = recall_score(y_test, y_pred, average='weighted')

    f1 = f1_score(y_test, y_pred, average='weighted')
```