Final Project Report

Machine Learning
ITE 5310 – Winter 2023 Humber College

Trainify : Machine learning Toolkit

By
Shivam Arora (N01586480)
Dhrumil Patel (N01586994)

- ## Introduction :

Trainify is a web based machine learning application which has data preprocessing, data analysis, and machine learning model training capabilities.

Trainify can be used :
https://shivamaroraa-machine-learning-streamlit-app-cgttdx.streamlit.app/

- ## Steps to use Trainify:

1. Upload dataset which has to be trained
2. Perform Data Preprocessing Techniques (Encoder, drop columns)
3. Perform Data Scaling Techniques(Standard/ MinMax Scaling)
4. Handle Missing values(Fill values using Mean/ Median/ Mode)
5. Choose a ML Model or Benchmark all models
   - KNN
   - SVM
   - Logistic Regression
   - Decision Tree
   - Random Forest
   - Naive Bayes
   - MLP classifier Neural Network

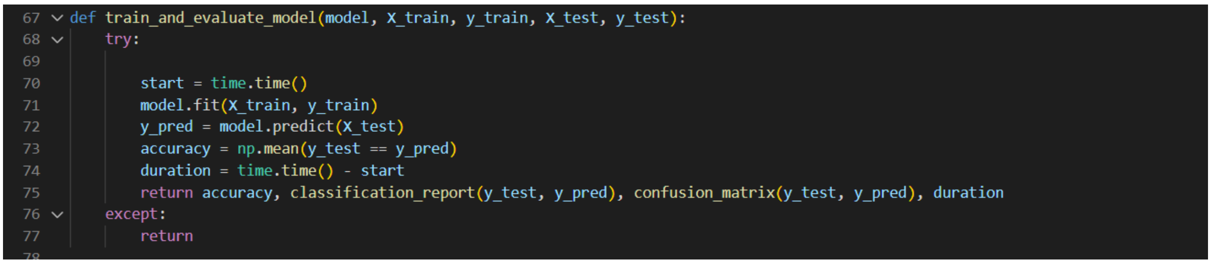
How Trainify Is built ?

Trainify is built completely in Python and uses the following libraries and frameworks:

- Sklearn
- Matplotlib
- Seaborn
- Pandas and Numpy
- Streamlit

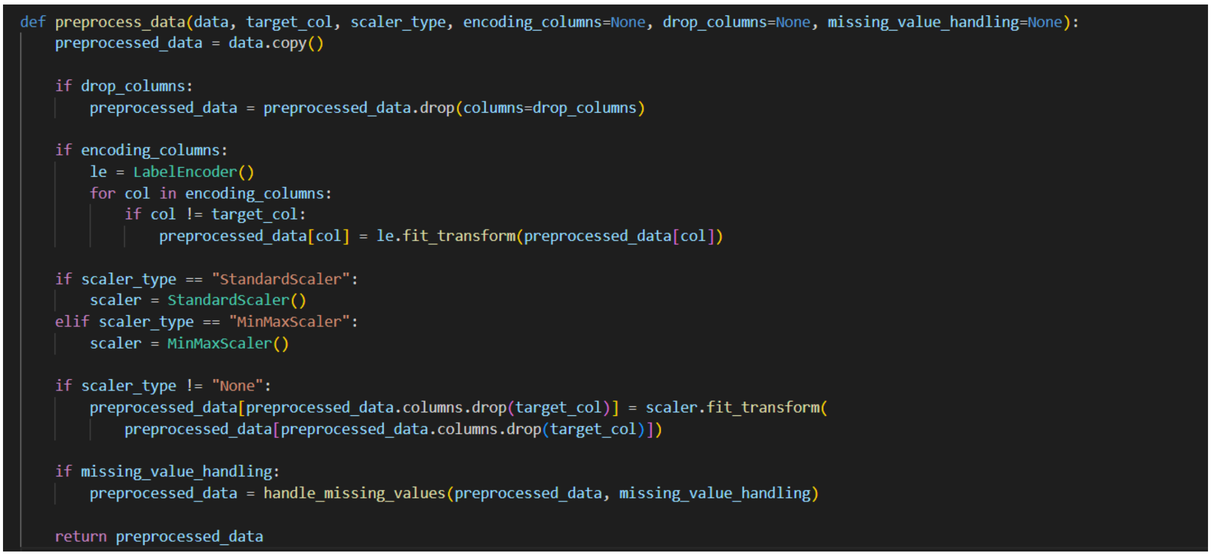
In Python, for every type of functionality such as Data preprocessing,model training, we have designed functions which will run and complete assigned tasks.

For example, if a user wants to train a model and he/she clicks on the train button, this function will be called. It will take the machine learning model name and x_train, y_train, x_text,y_test as input parameters.And if benchmark all models are called. Then one list will be created containing all models and then for each model, this function will run.



```python
def train_and_evaluate_model(model, X_train, y_train, X_test, y_test):
    try:

        start = time.time()
        model.fit(X_train, y_train)
        y_pred = model.predict(X_test)
        accuracy = np.mean(y_test == y_pred)
        duration = time.time() - start
        return accuracy, classification_report(y_test, y_pred), confusion_matrix(y_test, y_pred), duration
    except:
        return
```

Similarly, If any pre-processing step has been initialised, then the preprocess_data function will be called. And Depending on which preprocessing method is called, that part of the code will run and will update dataset. And these selected preprocessing method take dataset which has been uploaded by user and then it modify the dataset and update dataset as preprocessed dataset.

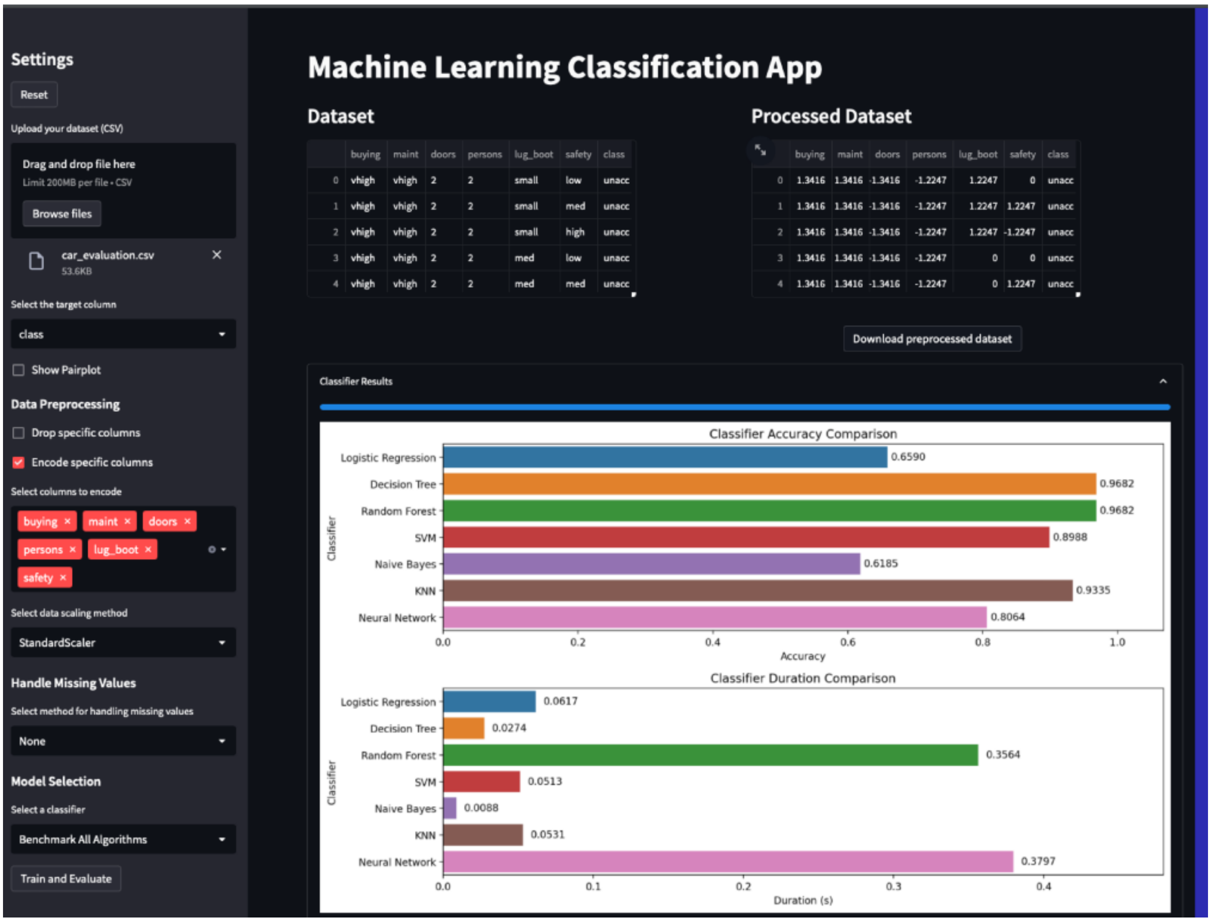


```python
def preprocess_data(data, target_col, scaler_type, encoding_columns=None, drop_columns=None, missing_value_handling=None):
    preprocessed_data = data.copy()

    if drop_columns:
        preprocessed_data = preprocessed_data.drop(columns=drop_columns)

    if encoding_columns:
        le = LabelEncoder()
        for col in encoding_columns:
            if col != target_col:
                preprocessed_data[col] = le.fit_transform(preprocessed_data[col])

    if scaler_type == "StandardScaler":
        scaler = StandardScaler()
    elif scaler_type == "MinMaxScaler":
        scaler = MinMaxScaler()

    if scaler_type != "None":
        preprocessed_data[preprocessed_data.columns.drop(target_col)] = scaler.fit_transform(
            preprocessed_data[preprocessed_data.columns.drop(target_col)])

    if missing_value_handling:
        preprocessed_data = handle_missing_values(preprocessed_data, missing_value_handling)

    return preprocessed_data
```

Steamlit is an open-source Python library that enables easy and efficient creation of interactive data applications. It allows to quickly build and visualise interactive graphs, tables, and charts. By using steamlit, the web application of the toolkit has been created.

Results:

Car-evaluation dataset has been used for testing this published website. After applying encoding for preprocessing and scaling , we have trained this dataset for all machine learning models. And as a result, We can see accuracy for all machine learning models, not only that, we can also check training time comparison between all machine learning models.



Settings

Reset

Upload your dataset (CSV)

Drag and drop file here
Limit 200MB per file • CSV

Browse files

car_evaluation.csv
53.6KB

Select the target column

class

☐ Show Pairplot

Data Preprocessing

☐ Drop specific columns

☑ Encode specific columns

Select columns to encode

buying × maint × doors × persons × lug_boot × safety ×

Select data scaling method

StandardScaler

Handle Missing Values

Select method for handling missing values

None

Model Selection

Select a classifier

Benchmark All Algorithms

Train and Evaluate

Machine Learning Classification App

Dataset

	buying	maint	doors	persons	lug_boot	safety	class
0	vhigh	vhigh	2	2	small	low	unacc
1	vhigh	vhigh	2	2	small	med	unacc
2	vhigh	vhigh	2	2	small	high	unacc
3	vhigh	vhigh	2	2	med	low	unacc
4	vhigh	vhigh	2	2	med	med	unacc

Processed Dataset

	buying	maint	doors	persons	lug_boot	safety	class
0	1.3416	1.3416	-1.3416	-1.2247	1.2247	0	unacc
1	1.3416	1.3416	-1.3416	-1.2247	1.2247	1.2247	unacc
2	1.3416	1.3416	-1.3416	-1.2247	1.2247	-1.2247	unacc
3	1.3416	1.3416	-1.3416	-1.2247	0	0	unacc
4	1.3416	1.3416	-1.3416	-1.2247	0	1.2247	unacc

Download preprocessed dataset

Classifier Results

Future Work :

- Adding more preprocessing steps
- Adding support for performing regression
- Adding steps such as choosing the number of layers, number of neurons, choosing activation functions in Neural Networks
- Deployment on a better cloud VPC and support for GPU for Neural Networks.