CAPSTONE Project

E-Commerce Data

(Actual Transactions from UK retailer)

BHAVYA ARORA

OVERVIEW

- INTRODUCTION
- OBJECTIVE
- APPROACH
- MODEL SUMMARY
- RESULTS
- REFERENCES

INTRODUCTION

The E-Commerce dataset used in this project is a comprehensive record of online retail transactions made by a UK-based retailer between December 2010 and December 2011. It comprises over 500,000 rows of real-world transactional data, capturing details such as invoice numbers, stock codes, product descriptions, quantities ordered, invoice dates, unit prices, customer IDs, and the countries from which the orders were placed. The data reflects purchases made by customers across various regions, with the majority originating from the United Kingdom. This rich dataset provides a valuable snapshot of consumer behavior, inventory dynamics, and international sales trends within the online retail sector.

```
In [4]: # datatypes of columns
        df.dtypes

Out[4]: InvoiceNo       object
        StockCode       object
        Description     object
        Quantity         int64
        InvoiceDate     object
        UnitPrice      float64
        CustomerID     float64
        Country         object
        dtype: object
```

OBJECTIVE

The primary objective of this Capstone Project is to explore, model, and analyze real-world eCommerce transaction data using both Machine Learning and Deep Learning techniques. The aim is to develop predictive models that can accurately estimate important business metrics such as total transaction value, understand and forecast customer purchasing behaviour, optimise sales strategies, enhance business intelligence and to compare the performance using both Machine Learning and Deep Learning approaches.

- **Clean and preprocess raw transactional data to ensure quality and consistency.**
- **Perform exploratory data analysis (EDA) to extract meaningful patterns and insights.**
- **Train and evaluate various Machine Learning regression models including XGBoost, Random Forest, LightGBM, and K-Nearest Neighbors.**
- **Build and fine-tune a Deep Neural Network (DNN) model for regression.**
- **Compare and contrast the effectiveness, accuracy, and computational efficiency of ML and DL models on this dataset.**
- **Draw insights into which modeling approach is more suitable for structured business data like eCommerce transactions.**

APPROACH

1. Understanding the Dataset
2. Data Cleaning & Preprocessing
3. Exploratory Data Analysis (EDA)
4. Machine Learning Modeling
5. Deep Learning Modeling
6. Model Comparison & Finalization



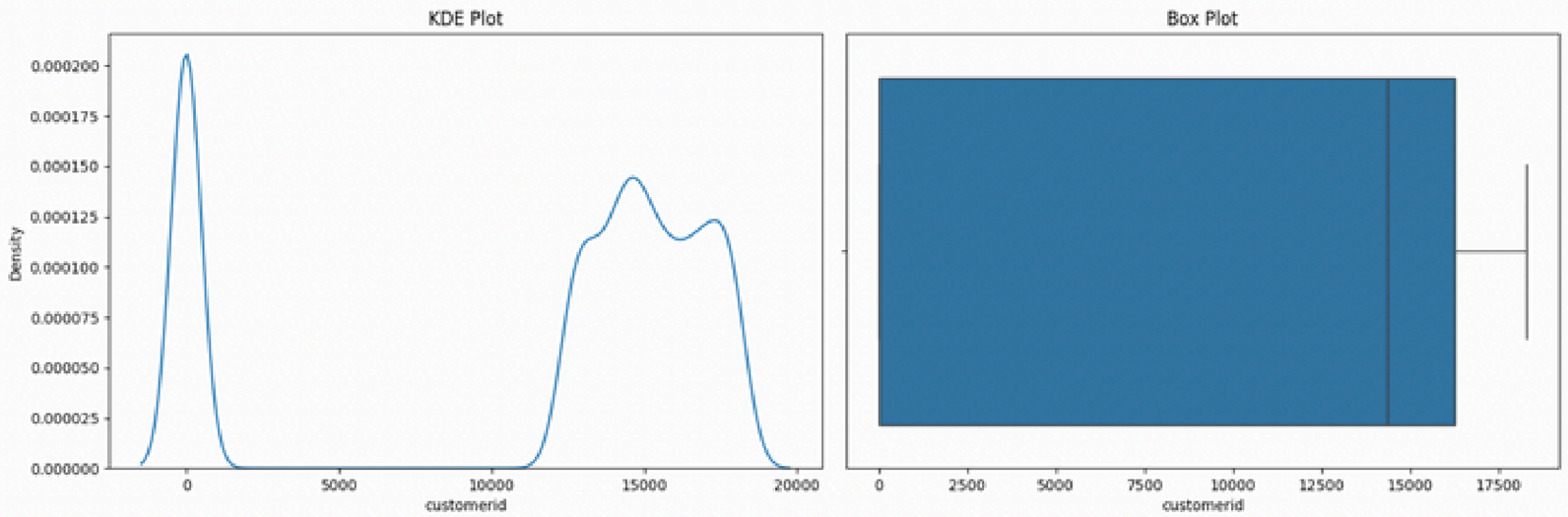


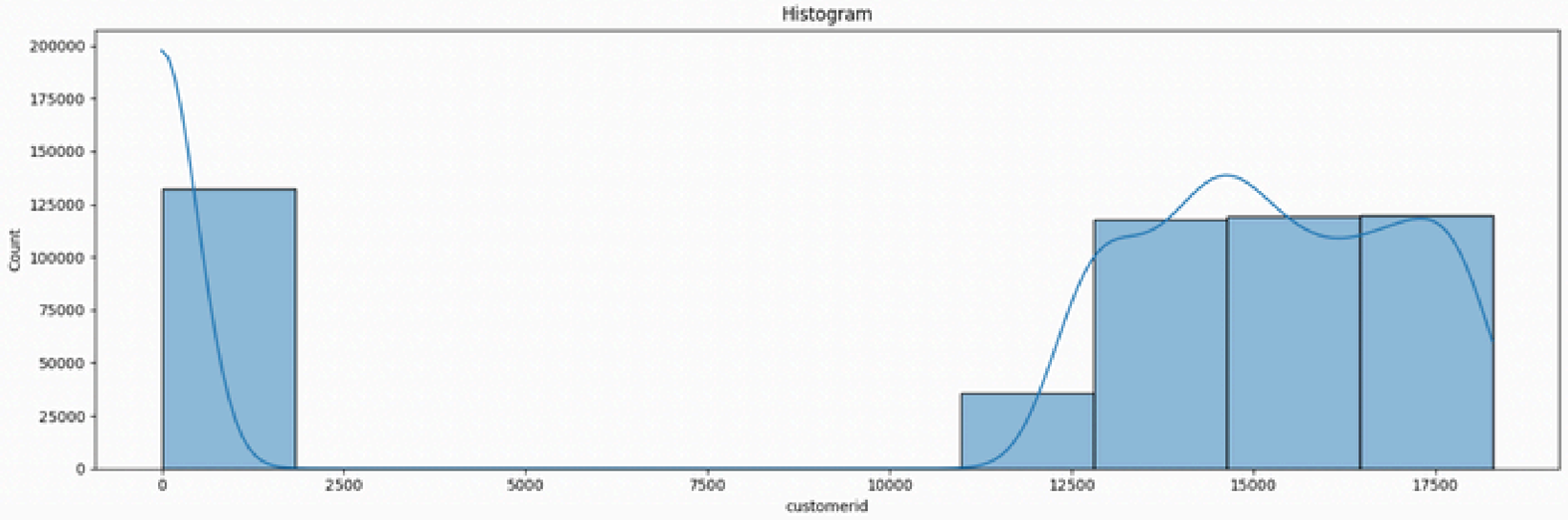
MODEL SUMMARY

MACHINE LEARNING RESULTS

NUMERICAL ANALYSIS

```
In [70]: # numerical analysis
         numerical_analysis(df_final, 'customerid',bins=10)
```





CLASSIFICATION REPORT

- Logistic Regression
- Random Forest
- KNN
- SVM

Logistic Regression Classification Report:

	precision	recall	f1-score	support
0	0.99	0.99	0.99	127784
1	0.71	0.51	0.59	3436
accuracy			0.98	131220
macro avg	0.85	0.75	0.79	131220
weighted avg	0.98	0.98	0.98	131220

Random Forest Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	127784
1	1.00	1.00	1.00	3436
accuracy			1.00	131220
macro avg	1.00	1.00	1.00	131220
weighted avg	1.00	1.00	1.00	131220

KNN Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	127784
1	1.00	1.00	1.00	3436
accuracy			1.00	131220
macro avg	1.00	1.00	1.00	131220
weighted avg	1.00	1.00	1.00	131220

SVM Classification Report:

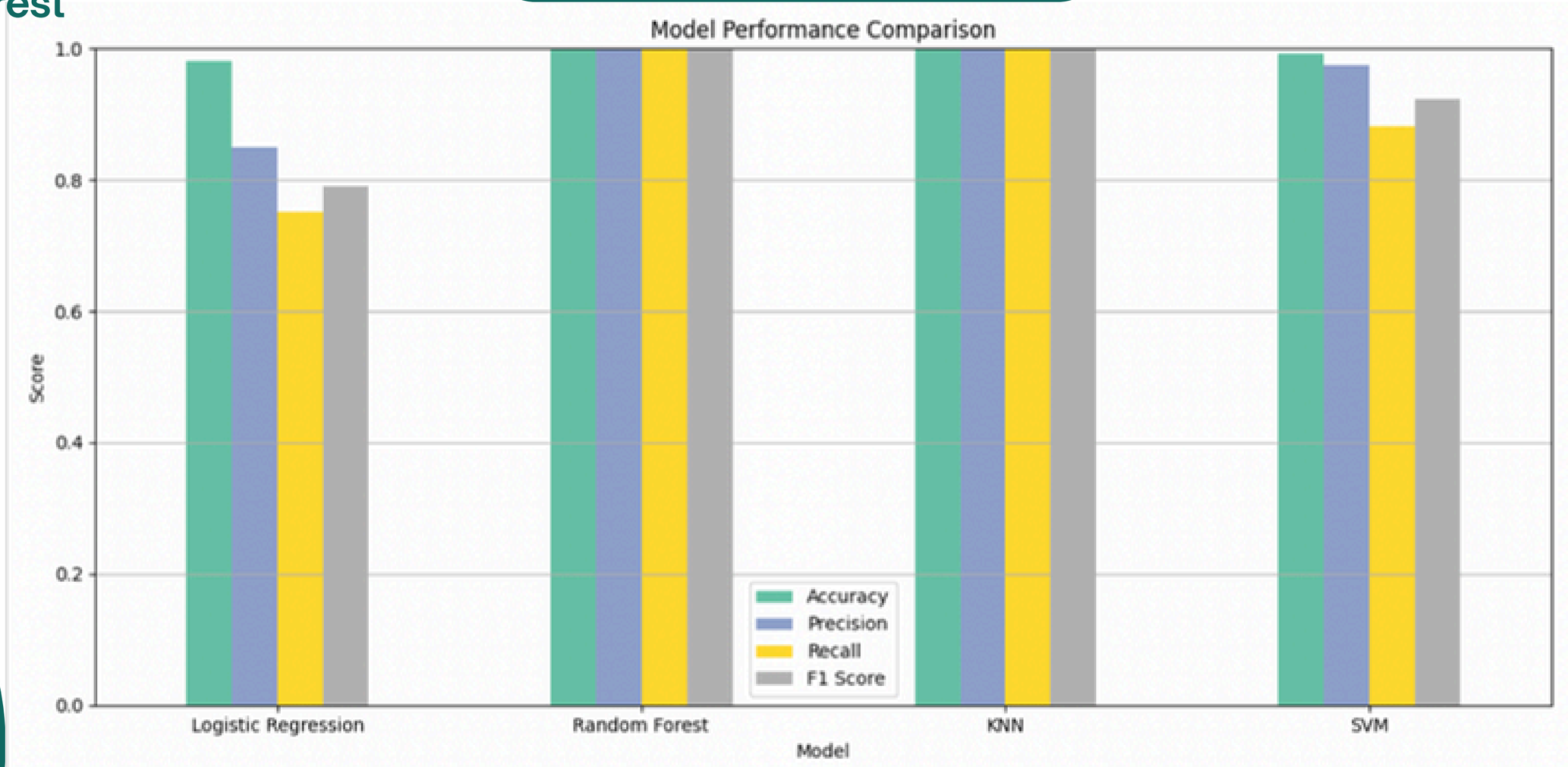
	precision	recall	f1-score	support
0	0.99	1.00	1.00	127784
1	0.95	0.77	0.85	3436
accuracy			0.99	131220
macro avg	0.97	0.88	0.92	131220
weighted avg	0.99	0.99	0.99	131220

MODEL PERFORMANCE COMPARISON

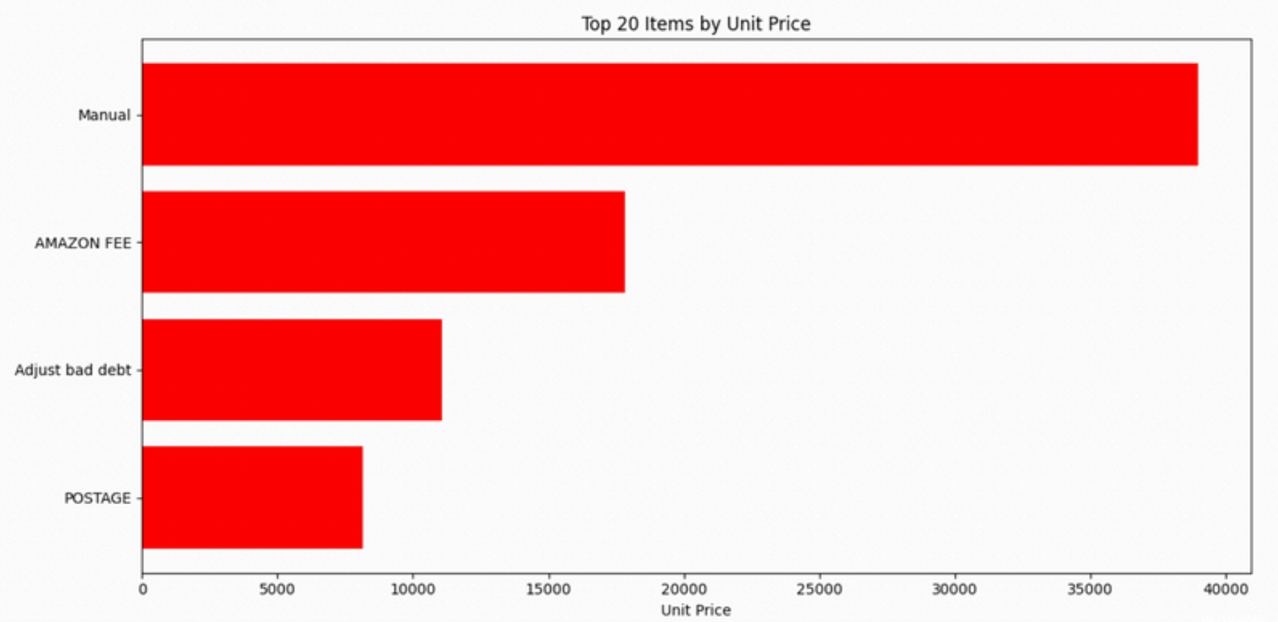
- Logistic Regression
- Random Forest
- KNN
- SVM

📊 Model Performance Comparison:

	Model	Accuracy	Precision	Recall	F1 Score
1	Random Forest	0.999931	0.999540	0.999115	0.999327
2	KNN	0.999779	0.997903	0.997762	0.997833
3	SVM	0.992897	0.974165	0.882078	0.922899
0	Logistic Regression	0.981702	0.848995	0.750868	0.791397

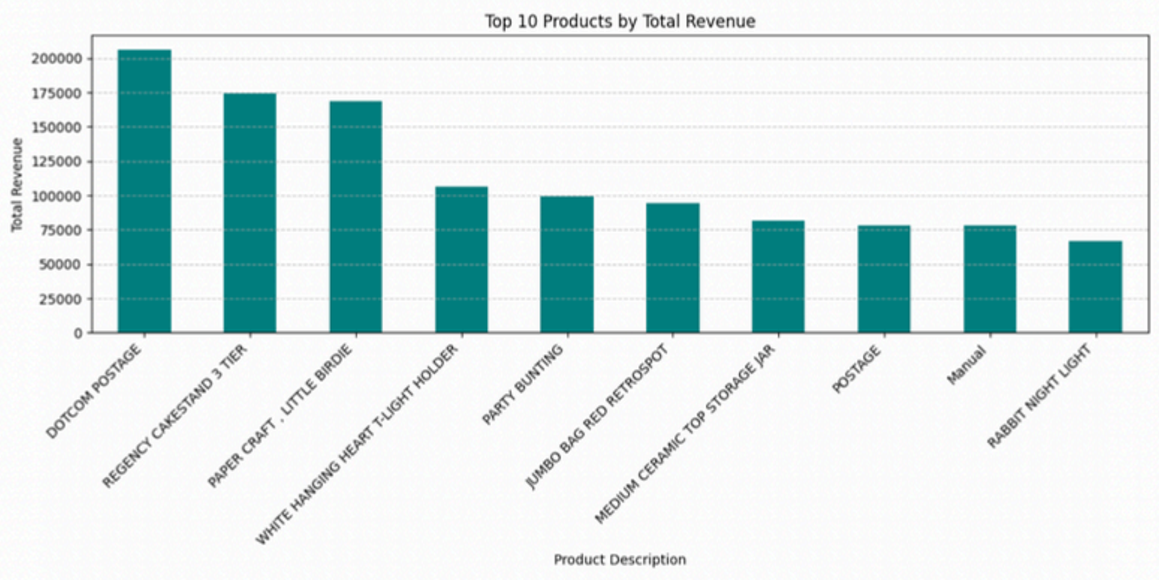


MACHINE LEARNING RESULTS



	quantity	unitprice	customerid	totalprice
count	524878.000000	524878.000000	524878.000000	524878.000000
mean	10.616600	3.922573	11437.732164	20.275399
std	156.280031	36.093028	6799.513627	271.693566
min	1.000000	0.001000	0.000000	0.001000
25%	1.000000	1.250000	0.000000	3.900000
50%	4.000000	2.080000	14350.000000	9.920000
75%	11.000000	4.130000	16245.000000	17.700000
max	80995.000000	13541.330000	18287.000000	168469.600000

	count	unique	top	freq
invoiceno	524878	19962	573585	1114
stockcode	524878	3922	85123A	2253
description	524878	4015	WHITE HANGING HEART T-LIGHT HOLDER	2311
invoicedate	524878	18499	2011-10-31 14:41:00	1114
country	524878	38	United Kingdom	479985



DEEP LEARNING MODEL RESULTS

FeedForward Neural Network

A Feedforward Neural Network often referred to simply as a Deep Neural Network (DNN) is an artificial neural network where the information moves in only one direction forward from the input layer, through hidden layers, to the output layer, without any loops or cycles.

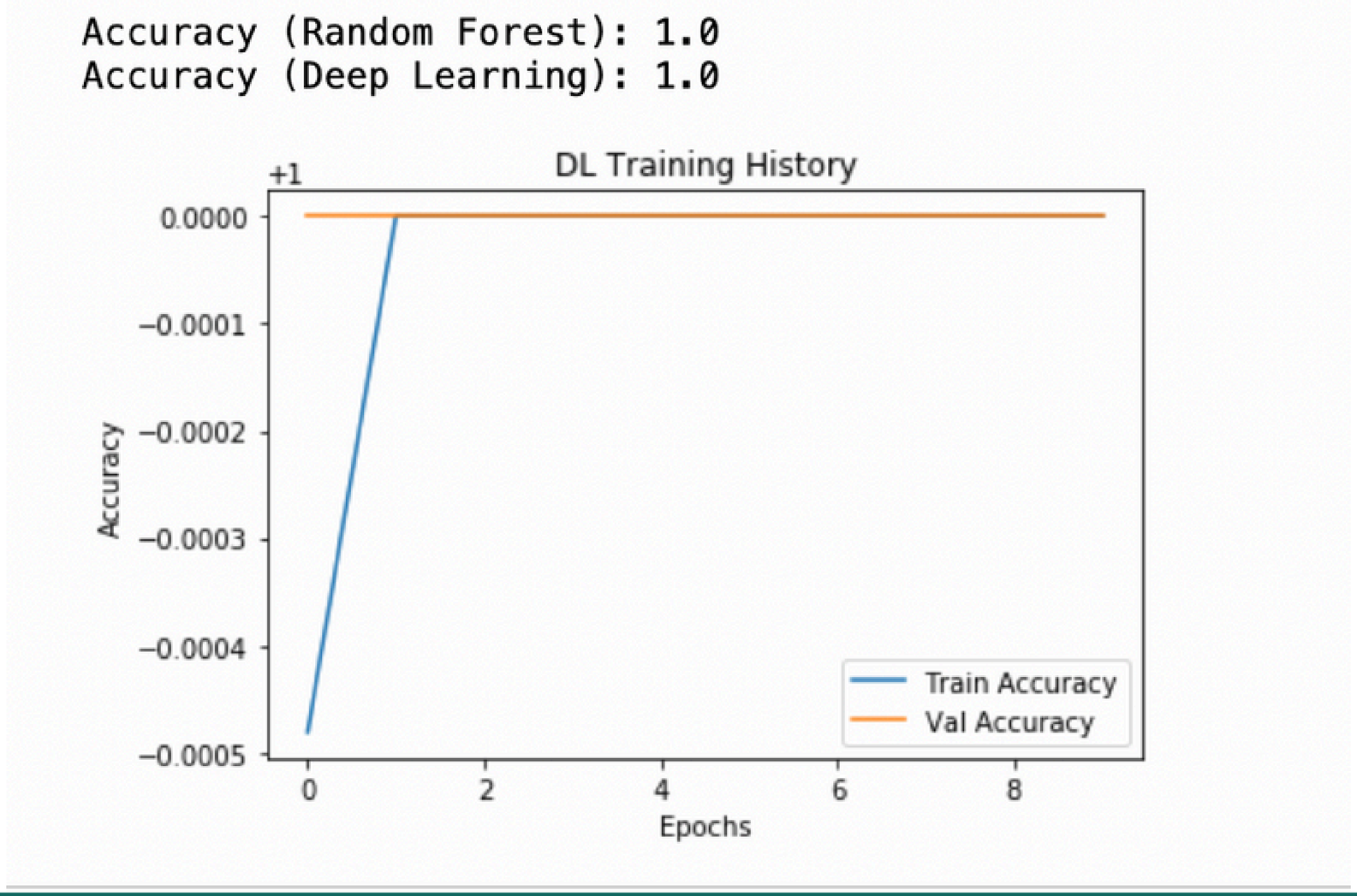
```
Epoch 1/10
9949/9949 [==============================] - 20s 2ms/step - loss: 0.0026 - accuracy: 0.9995 - val_loss: 1.3295e-08
- val_accuracy: 1.0000
Epoch 2/10
9949/9949 [==============================] - 19s 2ms/step - loss: 4.7119e-08 - accuracy: 1.0000 - val_loss: 1.5327e
-11 - val_accuracy: 1.0000
Epoch 3/10
9949/9949 [==============================] - 19s 2ms/step - loss: 5.1035e-10 - accuracy: 1.0000 - val_loss: 7.6754e
-13 - val_accuracy: 1.0000
Epoch 4/10
9949/9949 [==============================] - 17s 2ms/step - loss: 1.3855e-10 - accuracy: 1.0000 - val_loss: 2.3184e
-13 - val_accuracy: 1.0000
Epoch 5/10
9949/9949 [==============================] - 17s 2ms/step - loss: 6.2547e-11 - accuracy: 1.0000 - val_loss: 1.3243e
-13 - val_accuracy: 1.0000
Epoch 6/10
9949/9949 [==============================] - 18s 2ms/step - loss: 4.4833e-11 - accuracy: 1.0000 - val_loss: 8.5954e
-14 - val_accuracy: 1.0000
Epoch 7/10
9949/9949 [==============================] - 23s 2ms/step - loss: 2.2834e-08 - accuracy: 1.0000 - val_loss: 1.0267e
-14 - val_accuracy: 1.0000
Epoch 8/10
9949/9949 [==============================] - 19s 2ms/step - loss: 5.5675e-12 - accuracy: 1.0000 - val_loss: 9.8591e
-15 - val_accuracy: 1.0000
Epoch 9/10
9949/9949 [==============================] - 18s 2ms/step - loss: 6.9708e-12 - accuracy: 1.0000 - val_loss: 9.1762e
-15 - val_accuracy: 1.0000
Epoch 10/10
9949/9949 [==============================] - 17s 2ms/step - loss: 5.7227e-12 - accuracy: 1.0000 - val_loss: 8.8044e
-15 - val_accuracy: 1.0000
2488/2488 [==============================] - 3s 1ms/step - loss: 8.8044e-15 - accuracy: 1.0000
```

Deep Learning Model Results

	precision	recall	f1-score	support
1	1.00	1.00	1.00	79585
accuracy			1.00	79585
macro avg	1.00	1.00	1.00	79585
weighted avg	1.00	1.00	1.00	79585

COMPARISON IN MACHINE LEARNING AND DEEP LEARNING MODELS

```
Accuracy (Random Forest): 1.0
Accuracy (Deep Learning): 1.0
```



REFERENCES

01

DATASET

https://www.kaggle.com/datasets/carrie1/ecommerce-data/data

02

GITHUB

https://github.com/bhavyaarora23/Capstone-project.git

03

Hands-On Machine Learning with Scikit-Learn, Keras, and Tensorflow - Aurelien Geron

04

CloudX Lab Presentation and Resources