



Department of Computer Science Gujarat University

M.Sc.(Artificial Intelligence & Machine Learning) – I

Personal Finance

Presented By:

Bhautik Chudasama
Deep Korat

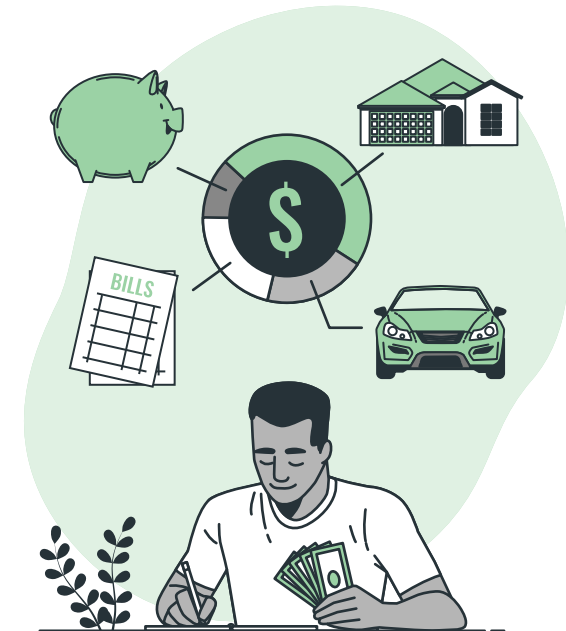
Under the Guidance of :

Ms. Sanskruti Bhatt



Introduction

Personalized financial management involves using ML to analyze an individual's financial data and provide customized recommendations for managing their money.





Objectives of Project

To help people to make better financial decisions, manage their money more effectively, and optimize their financial goals.

Predict monthly expenses, optimize Unusual expense, or assess credit risk, Recommendation of saving & investment.





Data Frame

1. How does the data appear like?:

Dimensionality of the DataFrame:

```
In [5]: df.shape
```

```
Out[5]: (3313, 9)
```

Data type of all features:

```
In [7]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3313 entries, 0 to 3312
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Unnamed: 0   3031 non-null   object
1   Unnamed: 1   3031 non-null   object
2   Unnamed: 2   3304 non-null   object
3   Unnamed: 3    0 non-null      float64
4   Unnamed: 4    0 non-null      float64
5   Unnamed: 5   3029 non-null   object
6   Unnamed: 6   3029 non-null   object
7   Unnamed: 7    831 non-null    object
8   Unnamed: 8   2475 non-null   object
dtypes: float64(2), object(7)
memory usage: 233.1+ KB
```



In [7]: `df.head(20)`

Out[7]:

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	Unnamed: 8
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
5	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
6	1-Apr-15 to 7-Nov-23	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
7	Date	Particulars	NaN	NaN	NaN	Vch Type	Vch No.	Debit	Credit
8	01-Apr-15	To	Opening Balance	NaN	NaN	NaN	NaN	29000.00	NaN
9	25-Mar-19	To	Upad	NaN	NaN	Receipt	1	1980.00	NaN
10	25-Mar-19	To	Chirag	NaN	NaN	Receipt	2	15000.00	NaN
11	25-Mar-19	To	Rajnik	NaN	NaN	Receipt	3	10000.00	NaN
12	26-Mar-19	By	Party With Friends	NaN	NaN	Payment	1	NaN	200.00
13	26-Mar-19	By	Tour Traveling Expenses	NaN	NaN	Payment	2	NaN	5.00
14	26-Mar-19	By	Jayesh Vekariya	NaN	NaN	Payment	3	NaN	25000.00
15	26-Mar-19	By	Mandal Date Advocate Group 02	NaN	NaN	Payment	4	NaN	19000.00
16	26-Mar-19	By	Axis Midcap Fund	NaN	NaN	Payment	5	NaN	10000.00
17	26-Mar-19	By	Food	NaN	NaN	Payment	6	NaN	20.00
18	26-Mar-19	By	Home Cash Paid	NaN	NaN	Payment	7	NaN	1000.00
19	26-Mar-19	By	Hair Cut	NaN	NaN	Payment	8	NaN	100.00



Data Preprocessing

Intended approach

Drop Rows which contains Null value in each column

```
In [12]: df1 = df.dropna(how='all')  
df1 = df1.reset_index(drop=True)
```

Extract Relevant Features

	Date	Description	Debit	Credit
0	2015-04-01	Opening Balance	29000.0	NaN
1	2019-03-25	Upad	1980.0	NaN
2	2019-03-25	Chirag	15000.0	NaN
3	2019-03-25	Rajnik	10000.0	NaN
4	2019-03-26	Party With Friends	NaN	200.0

Date
Description
Debit
Credit



Data Preprocessing

Change the data type of debit, credit and date.

```
In [47]: df1['Debit'] = pd.to_numeric(df1['Debit'], errors='coerce')
df1['Credit'] = pd.to_numeric(df1['Credit'], errors='coerce')
df1['Date'] = pd.to_datetime(df1['Date'])
```

Replace all NaN Value with 0

	Date	Description	Debit	Credit
0	2015-04-01	Opening Balance	29000.0	0.0
1	2019-03-25	Upad	1980.0	0.0
2	2019-03-25	Chirag	15000.0	0.0
3	2019-03-25	Rajnik	10000.0	0.0
4	2019-03-26	Party With Friends	0.0	200.0
5	2019-03-26	Tour Traveling Expenses	0.0	5.0
6	2019-03-26	Jayesh Vekariya	0.0	25000.0



Data Preprocessing

**For Expense Prediction,
we considered only Debit column.**

	Date	Description	Debit
0	2015-04-01	Opening Balance	29000.0
1	2019-03-25	Upad	1980.0
2	2019-03-25	Chirag	15000.0
3	2019-03-25	Rajnik	10000.0
4	2019-03-28	Alpesh Vithani	2000.0
5	2019-03-28	Misc.Income	5.0
6	2019-03-29	Upad	350.0
7	2019-03-31	Office Income	4000.0
8	2019-04-01	IDBI Bank	15000.0
9	2019-04-01	Office Expense	1210.0



Data Preprocessing

We extracted 'Month' feature from Date

	Date	Description	Debit
0	2015-04-01	Opening Balance	29000.0
1	2019-03-25	Upad	1980.0
2	2019-03-25	Chirag	15000.0
3	2019-03-25	Rajnik	10000.0
4	2019-03-28	Alpesh Vithani	2000.0
5	2019-03-28	Misc.Income	5.0
6	2019-03-29	Upad	350.0
7	2019-03-31	Office Income	4000.0
8	2019-04-01	IDBI Bank	15000.0
9	2019-04-01	Office Expense	1210.0

	Date	Description	Debit	Month
5	2019-03-28	Misc.Income	5.0	3
286	2021-01-15	Misc.Income	3800.0	1
592	2023-05-12	DBS Bank	15000.0	5
242	2020-12-05	Piyush Vaghasiya	555.0	12
568	2023-02-26	Bhavesb Sabhadiya (Loan)-1	122000.0	2
334	2021-03-22	Office Income	8600.0	3
192	2020-09-10	Office Income	190.0	9
586	2023-05-06	Sanskari Mandal	150.0	5
647	2023-10-16	Consultancy	1700.0	10
107	2019-12-23	DBS Bank	10000.0	12



Regression Model

We extracted 'Month' and 'Day of week' feature from Date

	Date	Description	Debit
0	2015-04-01	Opening Balance	29000.0
1	2019-03-25	Upad	1980.0
2	2019-03-25	Chirag	15000.0
3	2019-03-25	Rajnik	10000.0
4	2019-03-28	Alpesh Vithani	2000.0
5	2019-03-28	Misc.Income	5.0
6	2019-03-29	Upad	350.0
7	2019-03-31	Office Income	4000.0
8	2019-04-01	IDBI Bank	15000.0
9	2019-04-01	Office Expense	1210.0

	Date	Description	Debit	Month	Day_of_week
389	2021-08-06	Ankit Sorathiya	1000.0	8	4
419	2021-11-01	G S Enterprise (Ghanshyambhai)	2700.0	11	0
155	2020-06-07	Bharat Bank	5000.0	6	6
475	2022-03-17	DBS Bank	10000.0	3	3
232	2020-11-24	Rahil	846.0	11	1
341	2021-03-27	Upad	1000.0	3	5
27	2019-04-27	Cash Reserve I	500.0	4	5
359	2021-05-02	Cash Reserve I	100.0	5	6
390	2021-08-10	Chanchad	6200.0	8	1
464	2022-02-25	IDBI Bank	20000.0	2	4



Regression Model

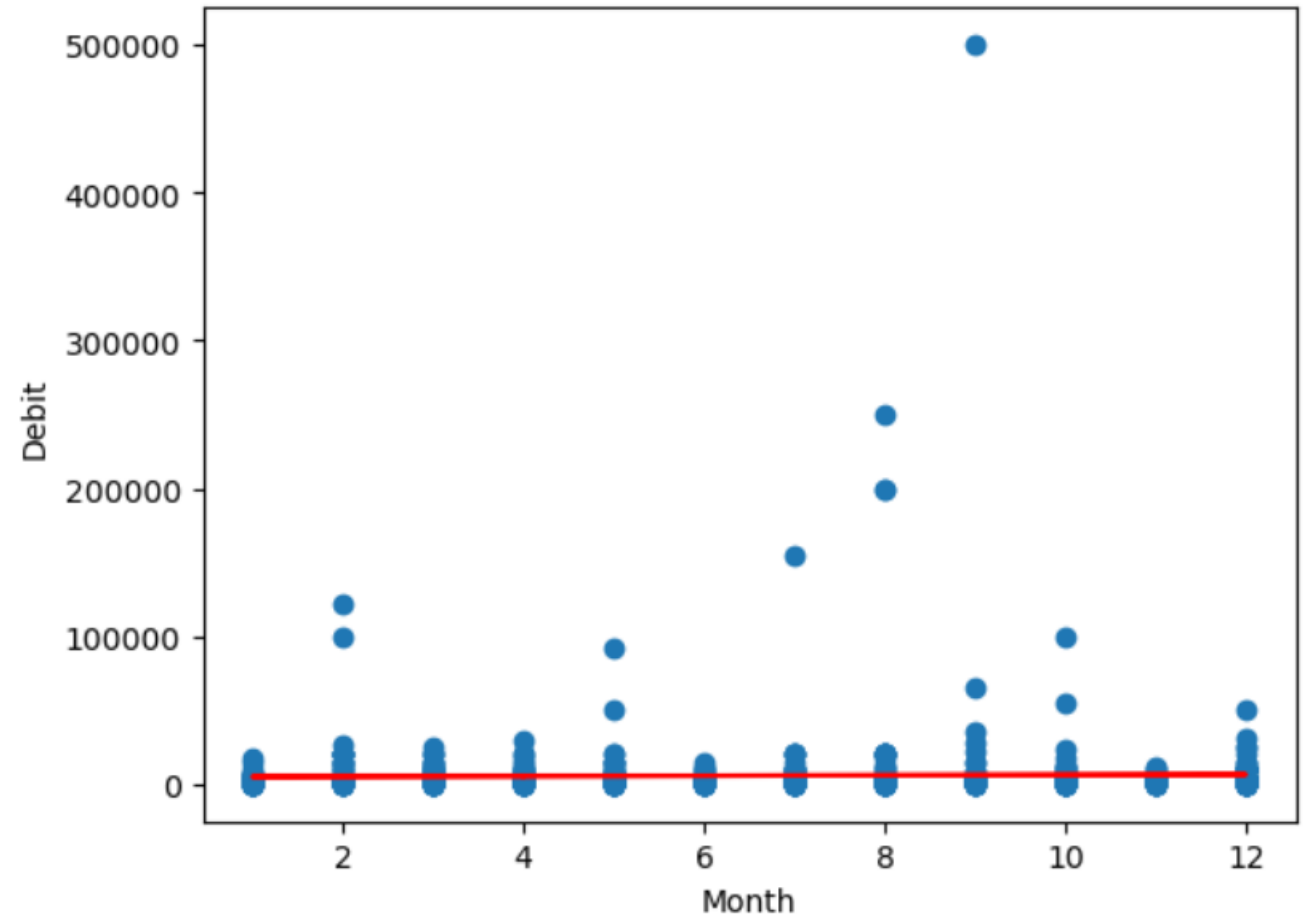
Input feature:- MONTH 

Output Feature :- DEBIT

Score:- -0.1004051

Intercept:- 4965.84

Coefficient:- `array([134.31835994])`





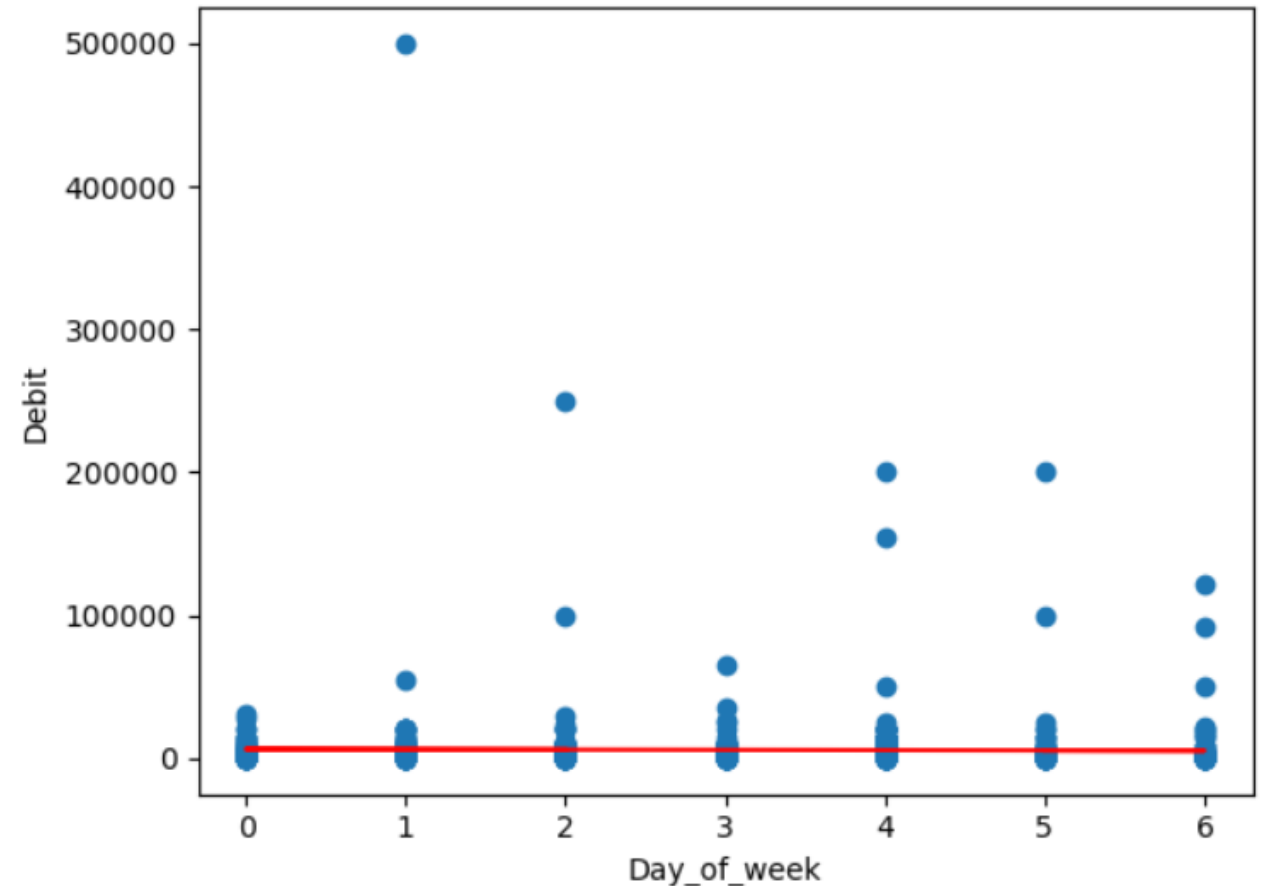
Regression Model

Input feature:- DAY OF WEEK
Output Feature :- DEBIT

Score:- -0.1042310

Intercept:- 6331.28

Coefficient:- `array([-173.81454389])`



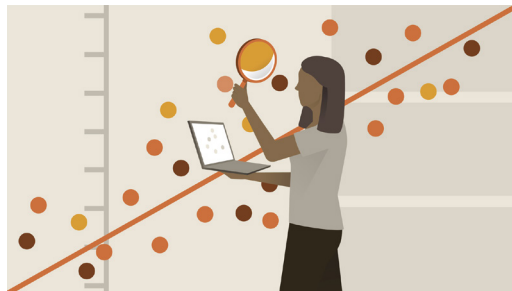


Obstacles and Considerations

We attempted to implement a regression model with 'Month' as the input feature and 'Debit' as the output.

Additionally, we explored using 'Day_of_week' as an input feature with 'Debit' as the output.

However, despite our efforts, we did not achieve satisfactory results or scores with the regression models.

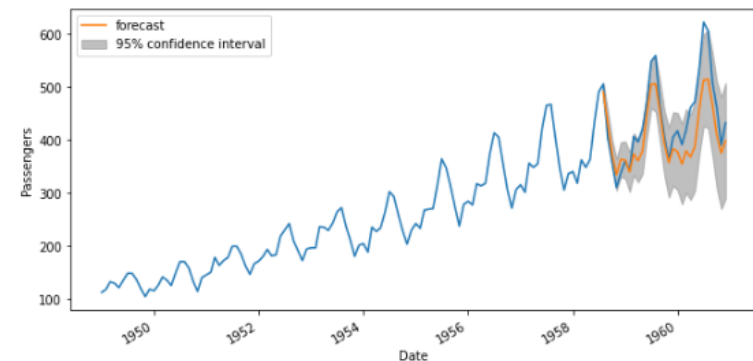




Obstacles and Considerations

In our pursuit of understanding the challenges, we considered the possibility that the time-dependent nature of the data might be better addressed using a time series model.

Hence, we will exploring the potential application of time series models such as ARIMA or other similar models, given that traditional regression models did not yield the desired outcomes.





Thank You
