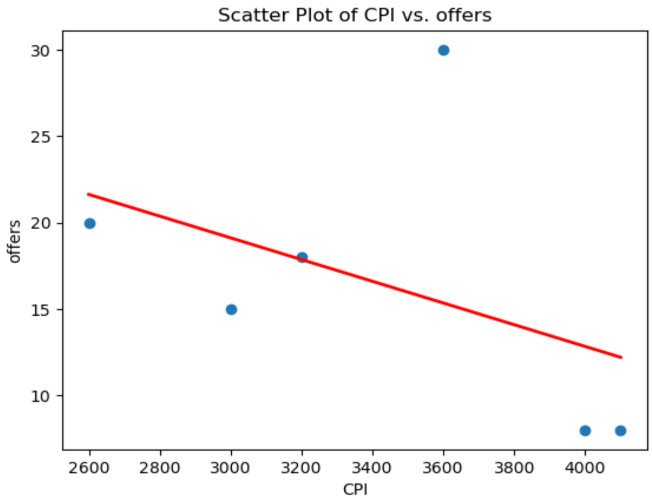
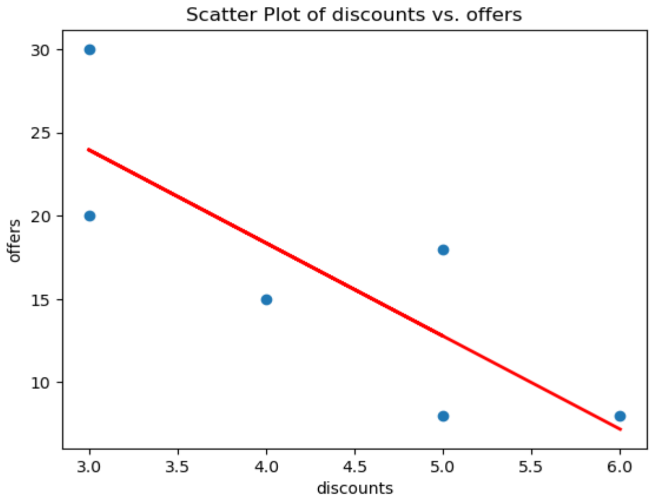
MACHINE LEARNING

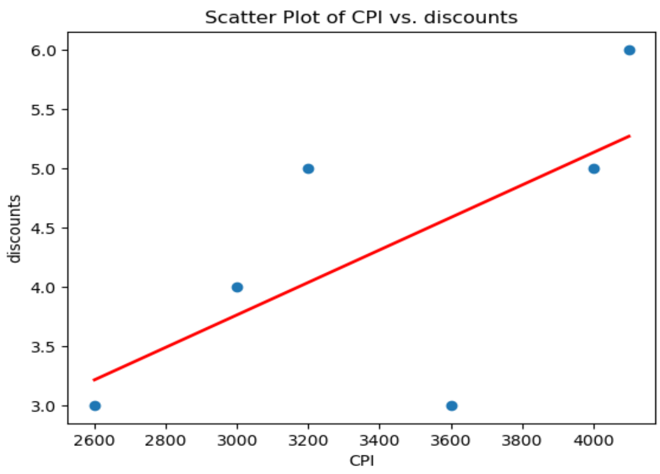
PROBLEM 1: Sales data

Multiple Linear Regression is the best suitable model for this problem because it is satisfying the below assumptions:

1) Linear relationship between variables- since data points are less though we have linear relationship among the variables we find bit difficult to plot it in scatter







2) No multi-collinearity:
   When correlation is less than 0.5% then we can choose those variables for model

	CPI	DISCOUNTS	OFFERS
CPI	1.000000	0.664772	-0.445300
DISCOUNTS	0.664772	1.000000	-0.816902
OFFERS	-0.445300	-0.816902	1.000000

Here though discounts and offers have high correlation our model is performing good. When we leave anyone variable between the two our model accuracy is decreasing.

RESULT SUMMARY:

OLS Regression Results

Dep. Variable:	Sales	**R-squared:**	0.952
Model:	OLS	**Adj. R-squared:**	0.879
Method:	Least Squares	**F-statistic:**	13.14
Date:	Mon, 22 Jan 2024	**Prob (F-statistic):**	0.0716
Time:	15:27:51	**Log-Likelihood:**	-68.476
No. Observations:	6	**AIC:**	145.0
Df Residuals:	2	**BIC:**	144.1
Df Model:	3		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	2.648e+05	1.64e+05	1.613	0.248	-4.41e+05	9.71e+05
CPI	128.4351	39.639	3.240	0.083	-42.120	298.990
discounts	5913.5196	2.99e+04	0.198	0.861	-1.23e+05	1.34e+05
offers	-4902.5460	3641.815	-1.346	0.311	-2.06e+04	1.08e+04

Omnibus:	nan	**Durbin-Watson:**	2.185
Prob(Omnibus):	nan	**Jarque-Bera (JB):**	0.238
Skew:	-0.031	**Prob(JB):**	0.888
Kurtosis:	2.026	**Cond. No.**	3.69e+04

Given below information find out the Sales that has

1. 5000 cpi , 3 percentage discounts, 20 rewards offers
2. 4000 cpi , 8 percentage discounts, 19 rewards offers

Sales for I is 826645.348382
Sales for II is 732680.364860

PROBLEM 2: Loan data

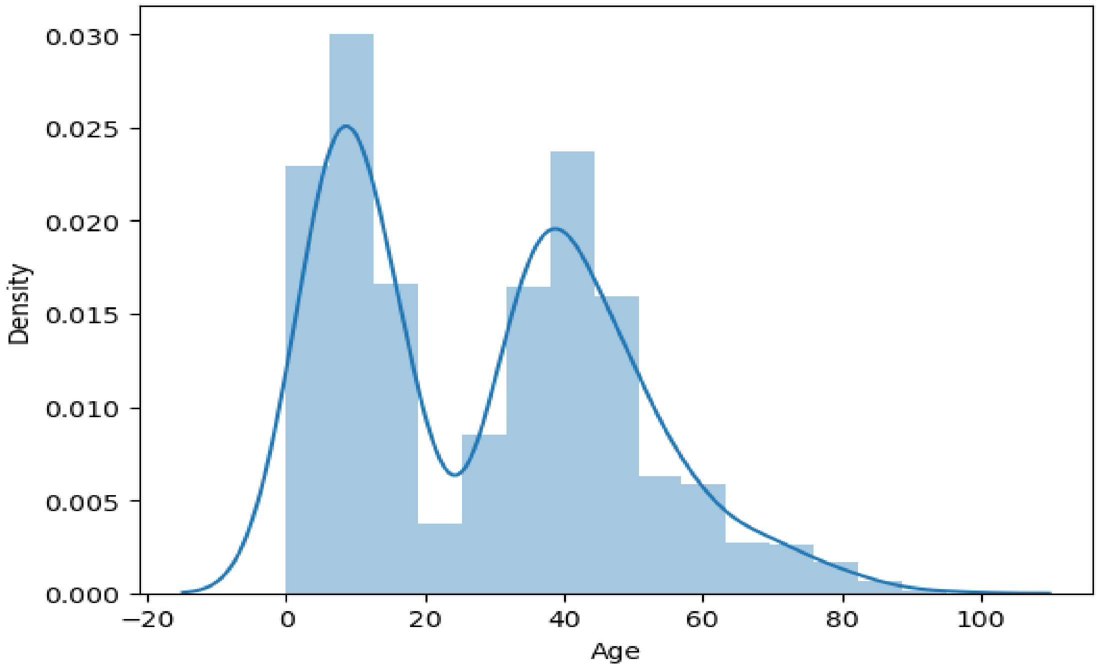
Logistic Regression is the best suitable to solve this problem

This data has the null values:

Customer id	0
Cards	12
Debit card	41
Insurance	48
Age	189
Cibil Score	0
Loan offer	0

NULL VALUE INPUTATION:

Since age is right skewed its better to go with median replacemet



Since Debit card, Insurance, Cards are binary in nature its good to choose Mode Replacement or bfill or ffill

After null imputation this is the result:

Cutomer id	0
Cards	0
Debit card	0
Insurance	0
Age	0
Cibil Score	0
Loan offer	0

TRAIN TEST SPLIT:

MODEL FIT:

```
Model=sm.Logit(y_train,x_train)
```

```
res=Model.fit()
res.summary()
```

```
Optimization terminated successfully.
         Current function value: 0.617064
         Iterations 7
```

Logit Regression Results

Dep. Variable:	Loan offer	**No. Observations:**	1072
Model:	Logit	**Df Residuals:**	1066
Method:	MLE	**Df Model:**	5
Date:	Fri, 26 Jan 2024	**Pseudo R-squ.:**	0.1097
Time:	13:37:07	**Log-Likelihood:**	-661.49
converged:	True	**LL-Null:**	-743.04
Covariance Type:	nonrobust	**LLR p-value:**	2.173e-33

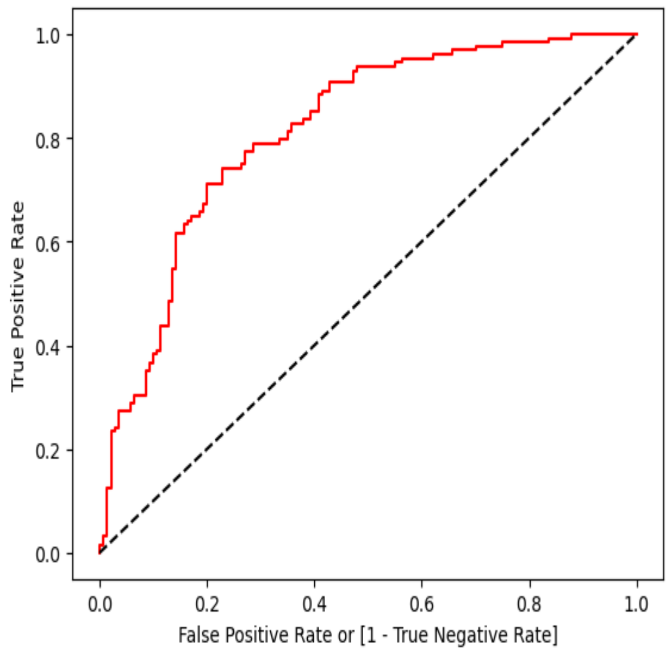
	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
const	-0.1664	0.247	-0.672	0.501	-0.651	0.319
Cards	0.2629	0.133	1.983	0.047	0.003	0.523
Debit card	0.5825	0.229	2.549	0.011	0.135	1.030
Insurance	-0.5287	0.585	-0.903	0.366	-1.676	0.619
Age	0.0059	0.004	1.670	0.095	-0.001	0.013
Cibil Score	-0.2972	0.032	-9.319	0.000	-0.360	-0.235

Accuracy score: 73%

Accuracy metrics

1. ROC curve
2. Confusion matrix

1) ROC curve



2) Confusion matrix:
array([[96, 44],
[27, 101]], dtype=int64)

PROBLEM 3:

Customer data has many null values which has to be treated

age	0
workclass	963
fnlwgt	0
education	0
education-num	0
marital-status	0
occupation	966
relationship	0
race	0
sex	0
capital-gain	0
capital-loss	0
hours-per-week	0
native-country	274
income	0

since null values are in categorical type we have imputed using Bfill,Ffill

Variables in the data set are categorical in nature treated using

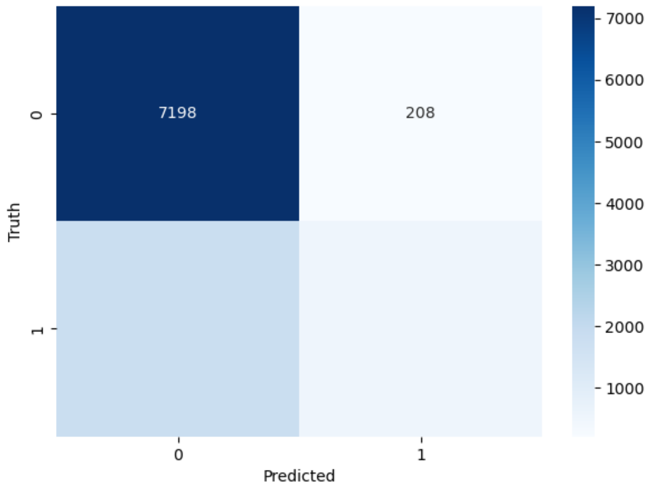
- One-hot encoding →pd.get_dummies
- Label encoding → from sklearn.preprocessing import LabelEncoder

KNN MODEL

Accuracy score →79%

Confusion matrix:

array([[7198, 208],
       [1815, 548]]

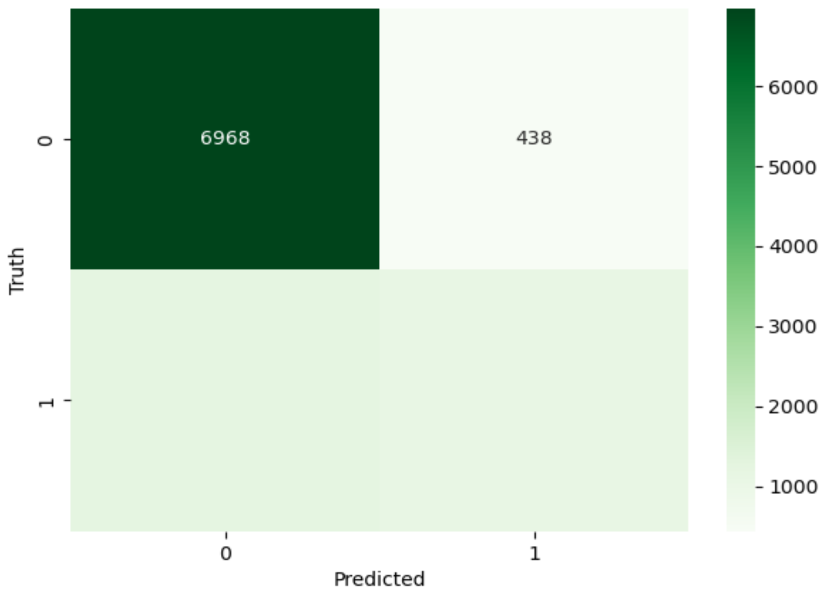


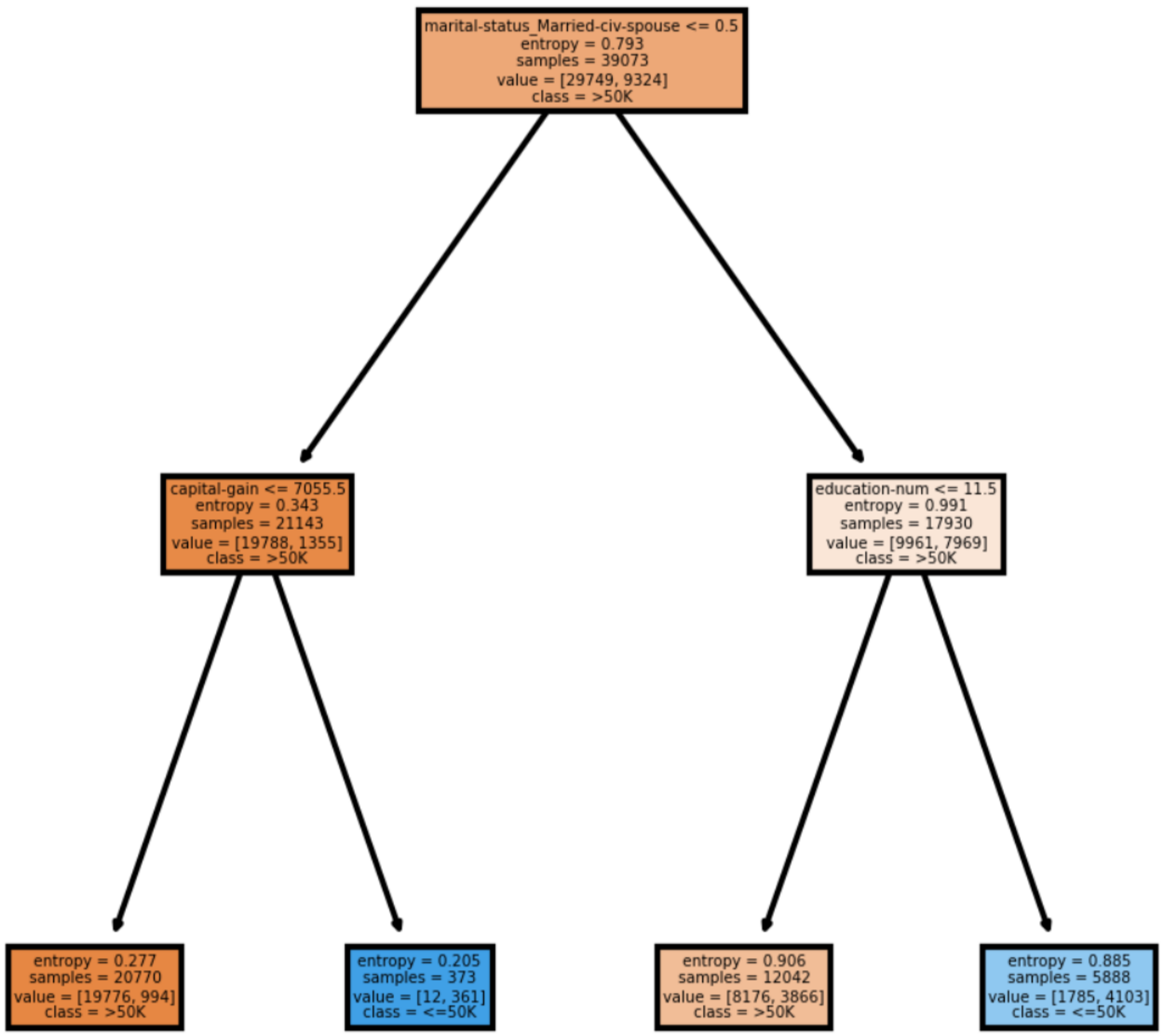
DECISION TREE

Accuracy score →82%

Confusion matrix:

array([[6968, 438],
       [1240, 1123]]

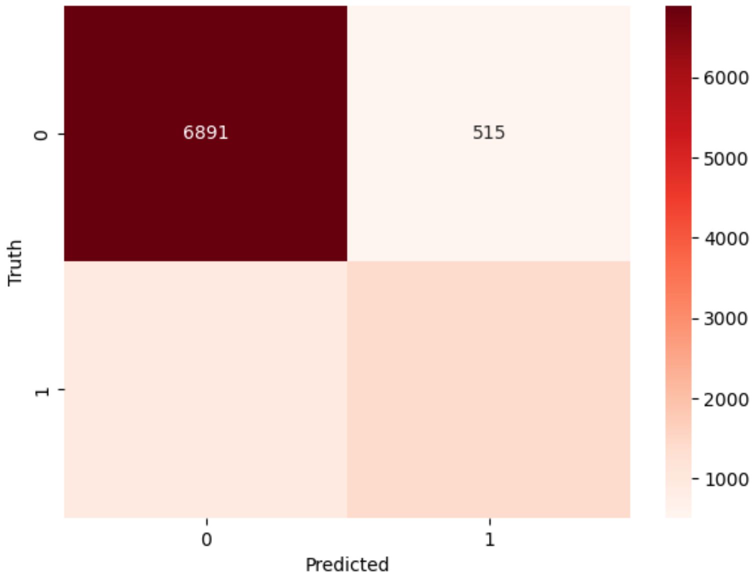

6968
438
0
1
Truth
0
1
Predicted
6000
5000
4000
3000
2000
1000


marital-status_Married-civ-spouse <= 0.5
entropy = 0.793
samples = 39073
value = [29749, 9324]
class = >50K
capital-gain <= 7055.5
entropy = 0.343
samples = 21143
value = [19788, 1355]
class = >50K
education-num <= 11.5
entropy = 0.991
samples = 17930
value = [9961, 7969]
class = >50K
entropy = 0.277
samples = 20770
value = [19776, 994]
class = >50K
entropy = 0.205
samples = 373
value = [12, 361]
class = <=50K
entropy = 0.906
samples = 12042
value = [8176, 3866]
class = >50K
entropy = 0.885
samples = 5888
value = [1785, 4103]
class = <=50K

RANDOM FOREST

Accuracy score →84%

Confusion matrix:
Array ([[6891, 515],
[960, 1403]]

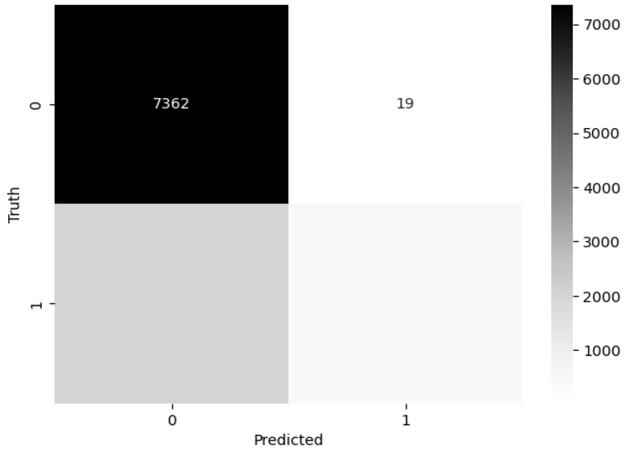


Support Vector Machine

Accuracy score →79%

Confusion matrix:

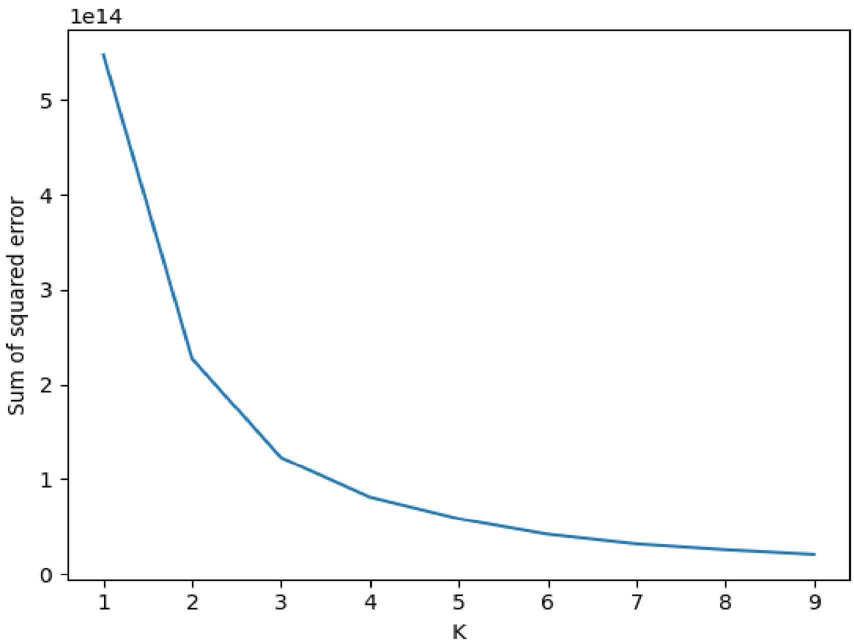
array([[7362,   19],
[2002, 386]],



K-MEANS

Accuracy score →40%

Elbow plot:
To decide how many clusters we must have:



It gives us an idea to opt 2 or 3

MODELS	ACCURACY SCORE
KNN	79%
Decision tree	82%
Random forest	84%
K-means	40%
SVM	79%

By seeing the above table we conclude that RANDOM FOREST is the suitable model for customer data because it has high accuracy score