CASSANDRA-22

Team Name - PAV-BHU-JEE

Team Members -

Parthasarthi Aggarwal

Ayush Agarwal

Vishal U Gosain

Final Rankings :

5/55 on public leaderboard
RMSE : **23.86276**
Best RMSE score: **22.38235**

Public Private

This leaderboard is calculated with approximately 30% of the test data. The final results will be based on the other 70%, so the final standings may be different.

#	Team	Members	Score	Entries	Last	Code
1	Akatsuki		22.38235	44	14h	
2	The Matrix		22.52019	49	15h	
3	HAL		23.61089	59	20h	
4	Har Har Mahadev		23.69602	104	15h	
5	**PAV-BHU-JEE**		23.86276	68	14h	
6	why so jealous		23.94615	32	16h	

Your Best Entry!
Your most recent submission scored 23.81558, which is an improvement of your previous score of 23.86276. Great job!

Tweet this

7/55 on private leaderboard
RMSE : **23.77738**
Best Rmse score :**21.566665**

Public Private

The private leaderboard is calculated with approximately 70% of the test data.
This competition has completed. This leaderboard reflects the final standings.

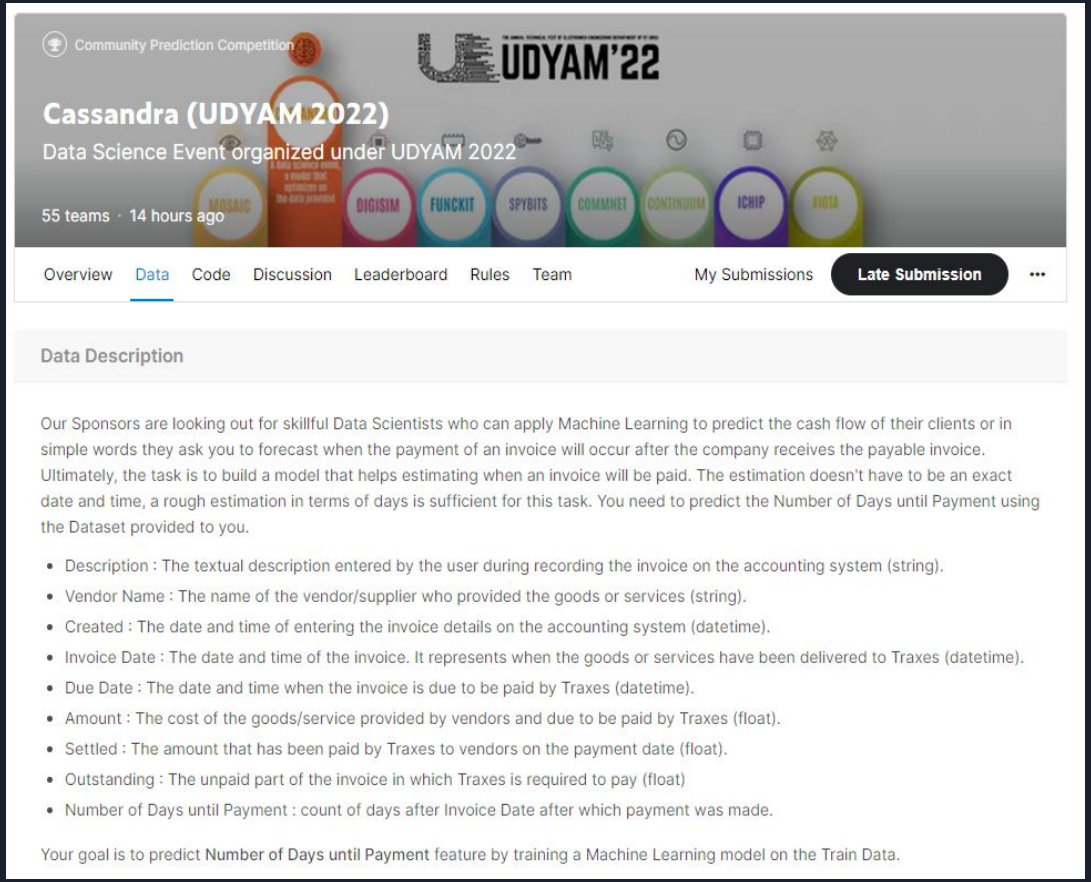
#	△	Team	Members	Score	Entries	Last	Code
1	▲ 1	The Matrix		21.56665	49	15h	
2	▼ 1	Akatsuki		22.26267	44	14h	
3	▲ 1	Har Har Mahadev		23.29969	104	15h	
4	▼ 1	HAL		23.69245	59	20h	
5	▲ 8	GREEKGODS		23.73101	76	14h	
6	▲ 14	submission_limit_full		23.77142	5	3d	
7	▼ 2	**PAV-BHU-JEE**		23.77738	68	14h	
8	▲ 4	Data Warriors		24.24329	47	15h	

Points of interest in our project :

- Difference feature columns
- Vendor_name column
- Fraud detection Algorithm
- Mutual Correlation graph and its impact
- K-Means Clustering

Understanding the Dataset

(Dataset : Cassandra (UDYAM 2022) | Kaggle)



Understanding the dataset terminology (Business Understanding) :

- “INVOICE_DATE” : is when goods are delivered
- “CREATED” : is when the invoice is created (which is usually after the delivery of goods
- “DUE_DATE” : is the date by which invoice is supposed to be paid back ,
- “AMOUNT” : is the amount of the invoice
- “SETTLED” : is the amount of invoice which has been paid before the due date
- “OUTSTANDING” : is the amount which is paid after the due date
- “DESCRIPTION” : is the description of the transaction
- “VENDOR_NAME” : name of vendor

Since the data patterns for this particular data were not be easy to observe , hence we have performed basic feature engineering first .

Feature Engineering :

	Description	Vendor_Name	Created	Invoice_Date	Due_Date	Amount	Settled	Outstanding	Number_of_Days_until_Payment
0	Milk x 7 ltrs	David Taylor	2011-04-26 11:50:00	2011-04-26	2011-05-26	672.78	672.78	0.00	13
1	Office Stationery	Stephen Wright MD	2011-05-24 09:40:00	2011-05-24	2011-06-23	5101.98	5101.98	0.00	38
2	Milk x 10 ltrs	Mark Cordova	2011-05-24 12:56:00	2011-03-24	2011-04-23	7422.78	7422.78	0.00	61
3	Annual Fee	Kimberly White	2011-09-07 10:42:00	2011-05-10	2011-05-24	11.98	11.98	0.00	62
4	NaN	Teresa Marshall	2011-05-09 20:55:00	2011-09-05	2011-09-06	5501.98	5501.98	0.00	2
5	NaN	Christian Ellis	2011-12-10 09:57:00	2011-10-12	2011-10-26	605.51	501.98	103.53	6
6	Office stationery	Courtney Smith	2011-10-26 12:24:00	2011-09-26	2011-10-27	2701.98	2701.98	0.00	35
7	Reverse misposting	Mario Peters	2011-10-26 15:47:00	2011-10-26	2011-11-25	3869.58	3869.58	0.00	26
8	Reverse mispost	Kelly Gray	2011-10-31 14:45:00	2011-10-31	2011-11-30	7801.98	7801.98	0.00	35
9	NaN	Teresa Marshall	2011-12-11 15:21:00	2011-11-12	2011-11-26	1048.28	1048.28	0.00	7

A lot of our data is in datetime format . For better analysis and better predictions by the ML model , we need the difference in number of days between the different columns . Hence we have created such features .

Our Features :

Created Hour	Created Minutes	Invoice Day	Invoice Month	Invoice Year	Due Day	Due Month	Due Year	diff_created	diff_due	Description_encoded
11	50	26	4	2011	26	5	2011	29.506944	30.0	1.0
9	40	24	5	2011	23	6	2011	29.597222	30.0	2.0
12	56	24	3	2011	23	4	2011	-31.538889	30.0	2.0
10	42	10	5	2011	24	5	2011	-106.445833	14.0	43.0
20	55	5	9	2011	6	9	2011	119.128472	1.0	NaN
9	57	12	10	2011	26	10	2011	-45.414583	14.0	NaN
12	24	26	9	2011	27	10	2011	0.483333	31.0	4.0
15	47	26	10	2011	25	11	2011	29.342361	30.0	1.0
14	45	31	10	2011	30	11	2011	29.385417	30.0	2.0
15	21	12	11	2011	26	11	2011	-15.639583	14.0	NaN

Explanation of added features : The difference feature

diff_created	diff_due
29.506944	30.0
29.597222	30.0
-31.538889	30.0
-106.445833	14.0
119.128472	1.0
-45.414583	14.0
0.483333	31.0
29.342361	30.0
29.385417	30.0
-15.639583	14.0

We have calculated the difference (in days) between different dates in a row , as the difference is much much more significant feature than the exact dates .

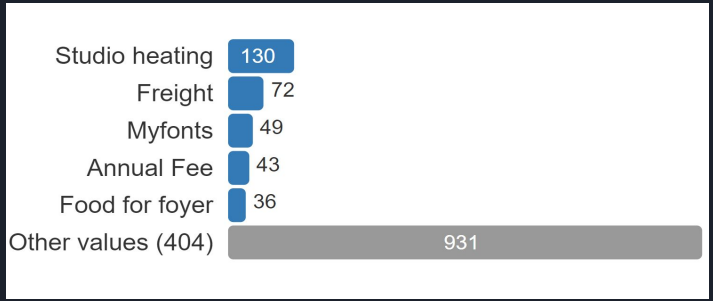
“diff_created” = no. of days it took for invoice to get created after delivery of goods

“diff_due” = difference (in number of days) between delivery of goods and due date

Explanation of added features : Description_encoded

This categorical column has about 86% missing values, 400 distinct classes and none of the class has significantly high frequency. Following these factors, one-hot encoding, ordinal encoding and target encoding were not an option.

Hence, frequency encoding was done as it preserves some information about values distribution, and is helpful for tree based models.



[MSP] Explanation of added features : Vendor_Name

This categorical column has no missing values, but it has about 3200 distinct classes. A few classes have considerable frequency. Hence, target encoding was done for this column, since this column has very important information.

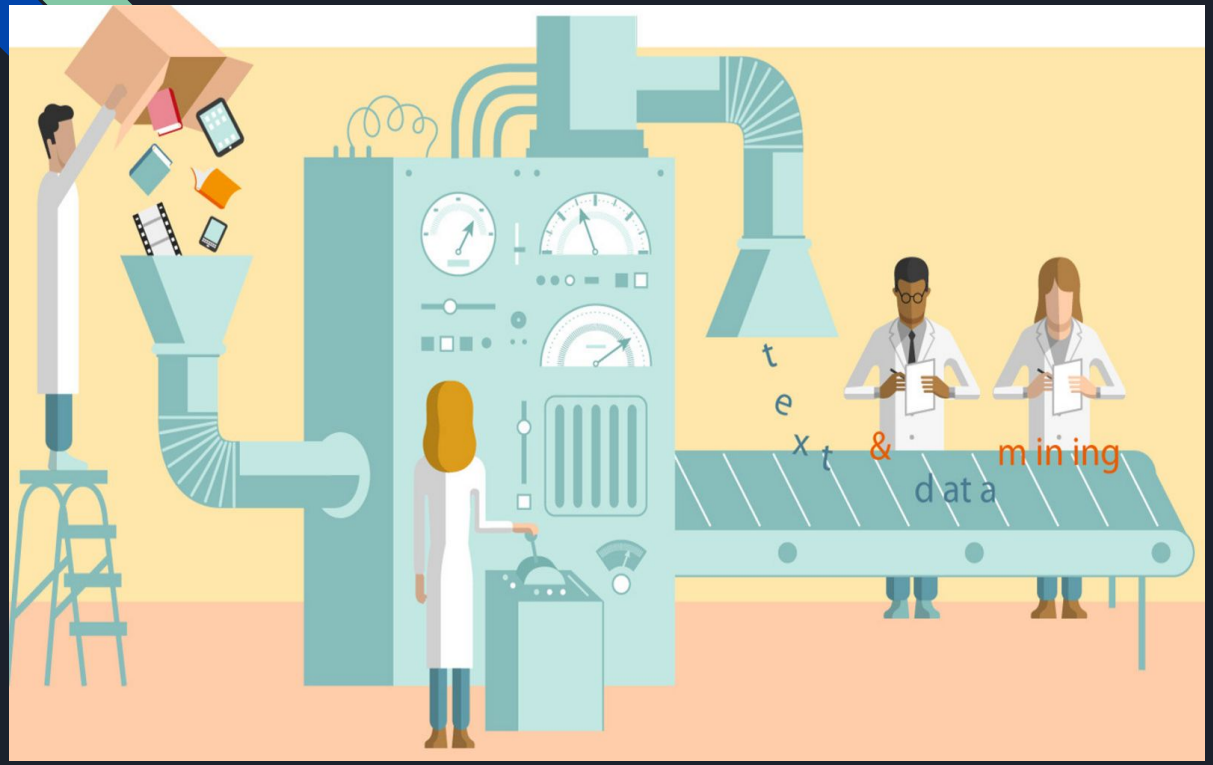
To obtain the encodings a separate split was taken from training data, this was done to prevent overfitting.

Also this column played very important role K-means clustering.

Explanation of added features : hour , minute , day , month , year

Since the datetime format is not very suitable for prediction by the ML model , hence we have extracted the day , month , year , time parts of the datetime data and stored in different columns separately .

Created Day	Created Month	Created Year	Created Hour	Created Minutes	Invoice Day	Invoice Month	Invoice Year	Due Day	Due Month	Due Year
26	4	2011	11	50	26	4	2011	26	5	2011
24	5	2011	9	40	24	5	2011	23	6	2011
24	5	2011	12	56	24	3	2011	23	4	2011
7	9	2011	10	42	10	5	2011	24	5	2011
9	5	2011	20	55	5	9	2011	6	9	2011
10	12	2011	9	57	12	10	2011	26	10	2011
26	10	2011	12	24	26	9	2011	27	10	2011
26	10	2011	15	47	26	10	2011	25	11	2011
31	10	2011	14	45	31	10	2011	30	11	2011
11	12	2011	15	21	12	11	2011	26	11	2011

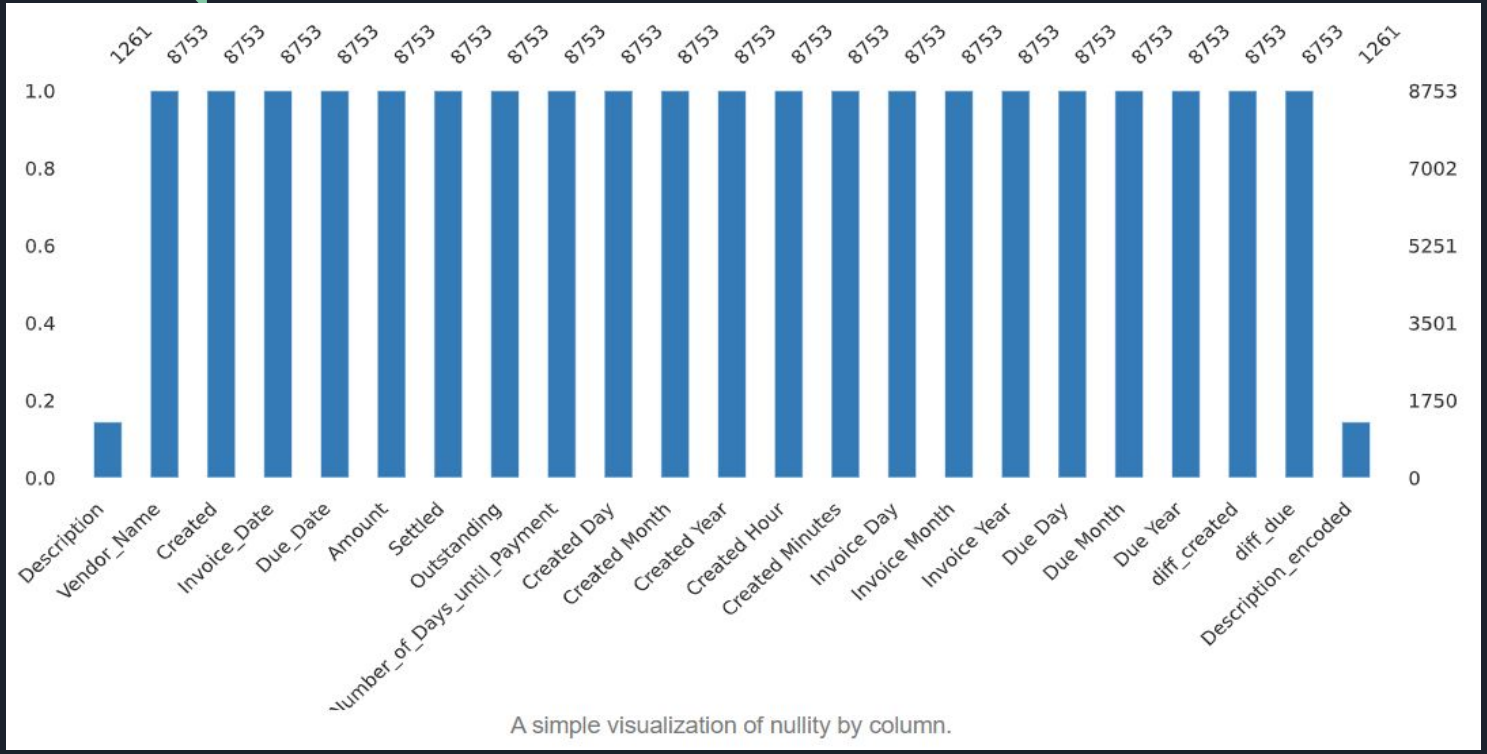


DATA EXPLORATION AND CLEANING

~ Via pandas profiling report

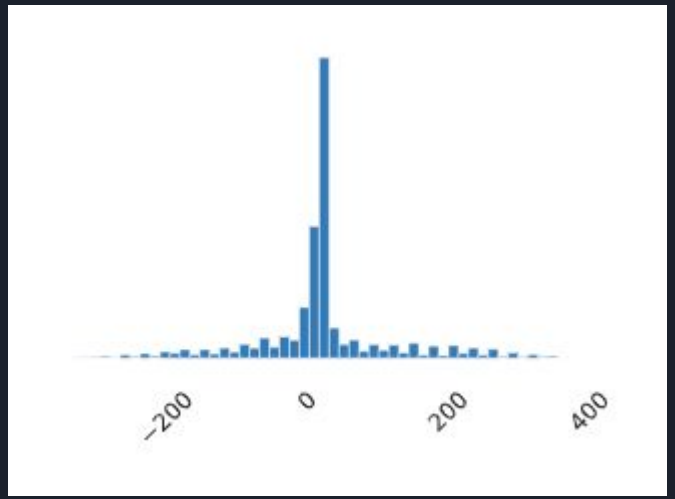
MISSING VALUES :

“Description” and “Description_encoded” are the only columns with missing values , not a lot of treatment required here .

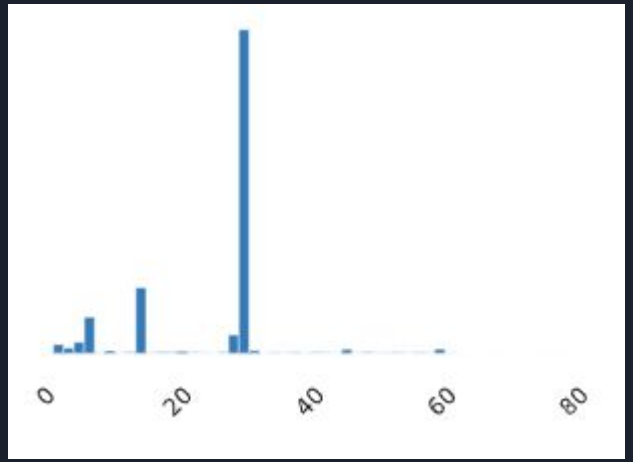


Exploring the data -

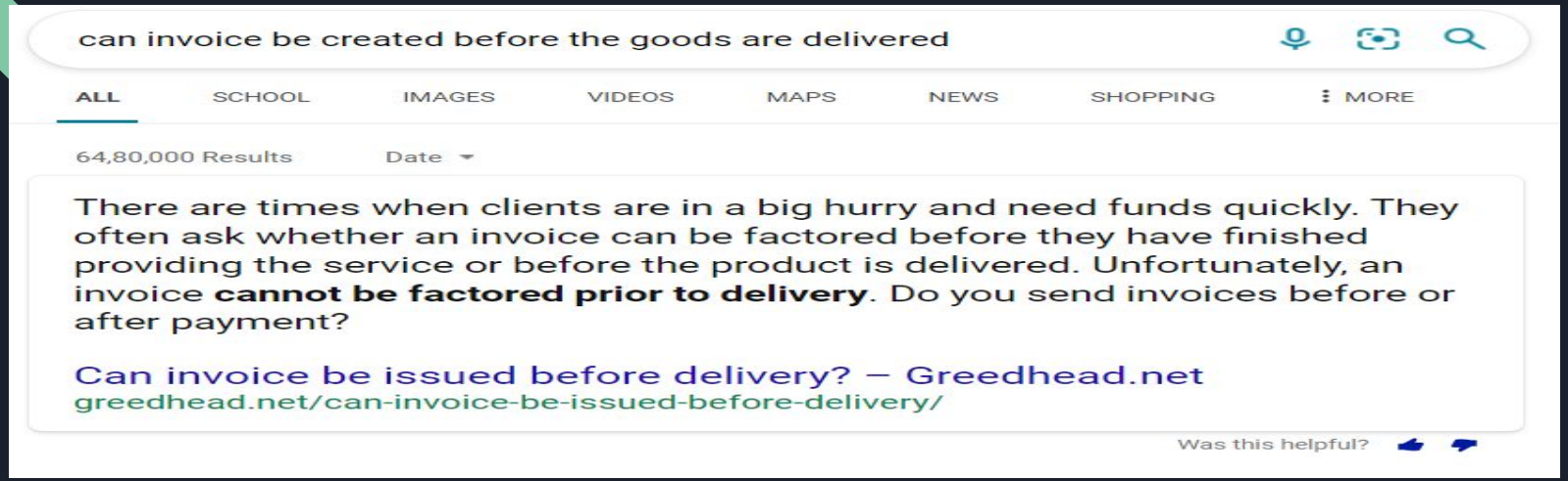
diff_created



diff_due

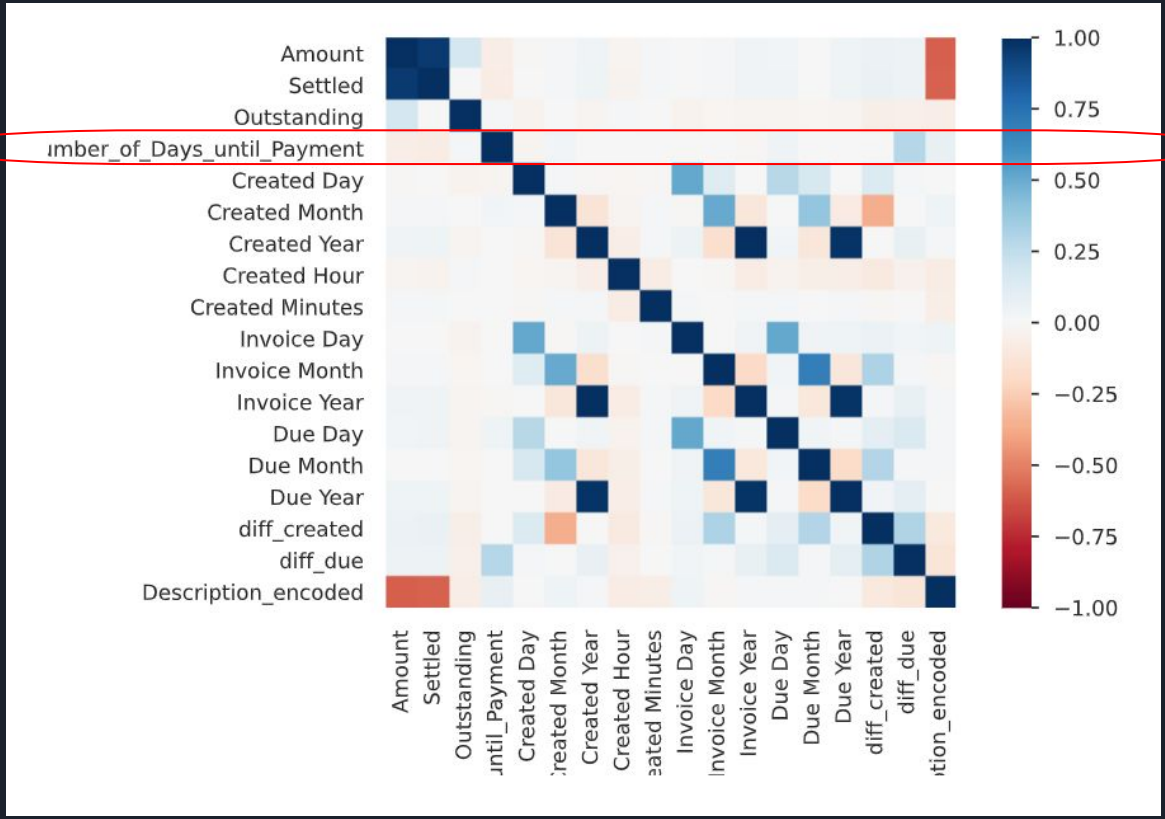


MSP : Fraud Detection Algorithm

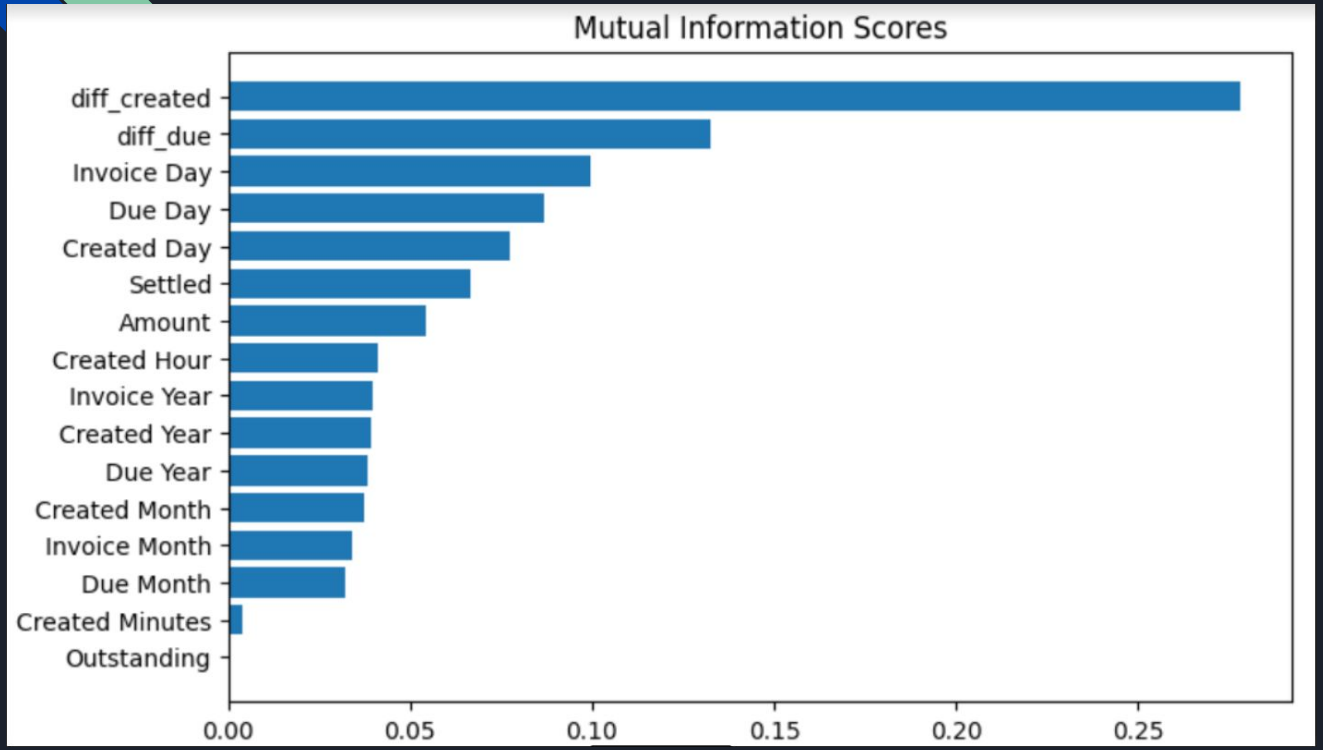


There are few cases in the dataset where the invoice has been created before the delivery of the goods , which is illegal , and can be easily detected. In realistic case , such cases would be outliers and would need to be removed , but for the competition , even the test data has such cases so to increase the accuracy of the model such cases were not removed .

[MSP] Correlations among the columns and target variable:



[MSP] Correlations among the columns and target variable:



The correlation calculator used by the pandas profiling report only captures the linear correlation between the variables , hence not being of much use to us .

However , the mutual_information_regressor from sklearn.feature_selection captures even non linear relations between the columns and the target variables , hence showing us the importance of the columns

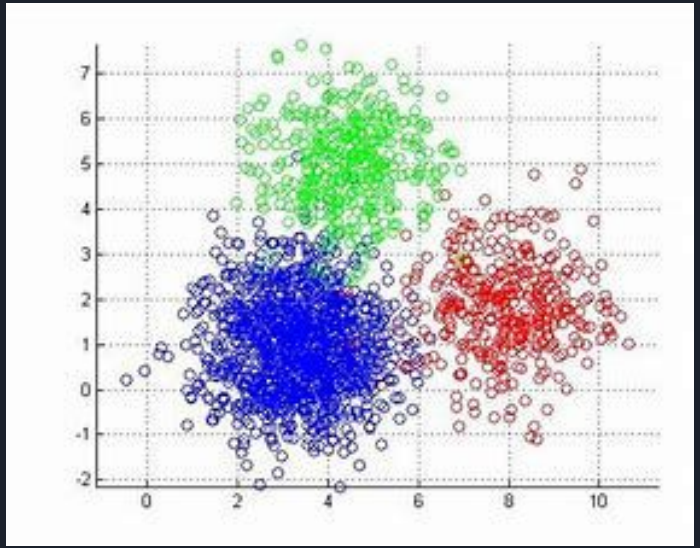
[MSP] Correlations among the columns and target variable:

This gave us conclusions such as :

- The difference features added by us were highly essential
- Created_minutes is not related to the target variable hence dropped later
- Separating the day month year and adding them in different columns was beneficial for the model prediction .

[MSP] K-Means Clustering

- K-Means clustering is an unsupervised learning algorithm that groups together data spread across feature space into separate clusters .
- We have used certain columns as feature space and performed k means clustering on it , and then we have assigned each data point to a cluster , and added a column named cluster in the data .
- Ensembling algorithms has a weakness of not being able to account for inter feature correlations aggregation , using k means combats that well .



The ML Model part :

- We decided to go for gradient boosting and ensembling methods based algorithms since these are considered State of the Art (SOTA) on structured data at present .
- After hit and trial with CatBoost , XGBoost , LightGBM and Sklearn gradient boosting regressor , we achieved maximum performance with CatBoost at n_estimators=600 , learning_rate=0.05 .

THANK YOU !!!!!

Any questions ?