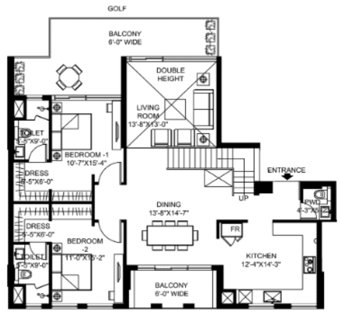
EDA

19 August 2023 19:54

Real estate based analytics website

- → Overview → $\boxed{\text{Data gathering}}$ ←
- → Data cleaning
- → Feature engineering
- → $\boxed{\text{EDA}}$
- → Devops

log(0.7) X log(1+0.7) →



EDA

univariate ✓
- col (distribution
- missing
- outliers

Pandas profiling ✓

multivariate analysis

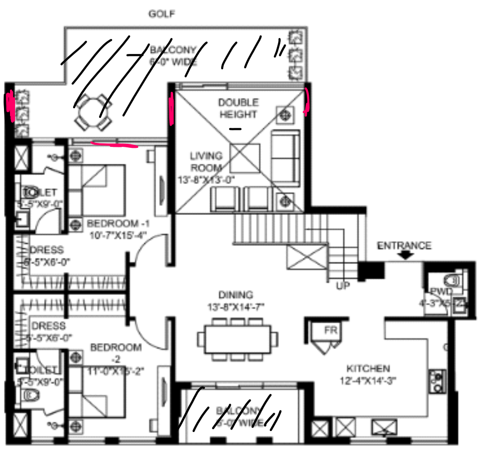
675 app → 3600 flats
↓
75 apps → 1800 flats { 23+22+21 1 }
600 app → 1800 flats

log(2.5) =

23
↓
23 × 22
↓
bivariate

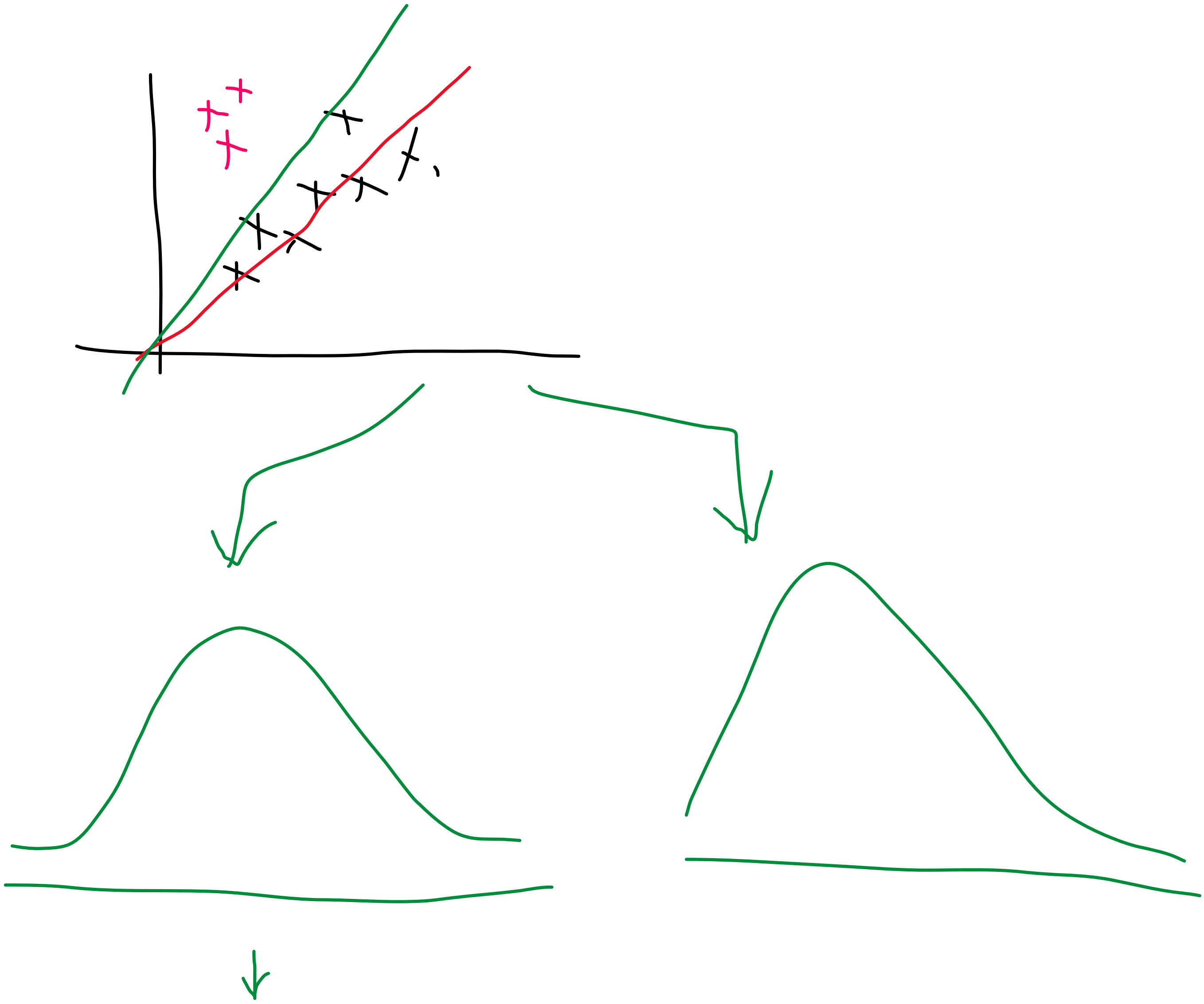
99 acres housing

super built
↓
Built up area
- balcony
- thickness of walls
↓
carpet area



{ outlier detection & removal } → { missing value imputation } → { feature selection } → { reg model }

19 August 2023 21:36



23 August 2023 10:12

Project

1 Discussion
2 Data gathering
3 Data cleaning
4 Feature engineering
5 Outlier detection
6 Missing values → data

1) model selection
   → hy
2) website
3) deploy

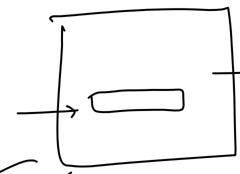
(6) classes
Recom → (1) (2)
Analysis → (1)
Scpt

feature selection →

Baseline model → price predict →

sector —
sqft —
3bhk —
bath —
~~price/sqft~~ X

~~society~~ → 99acres



→ luxury score

52/64/100

Categorical → 3 category
budget, semi luxury, luxury

low rue (1-2), medium 3-10, high (10)

permut import

8 feature selection technique

f1	f2	f3	target
57 ~~23~~			
23 ~~41~~			
41 ~~57~~			

jumble

→ model → 0.82

f1 f2 f3   f1 is import

f1 jumble	f2	f3	target

→ model → $\boxed{0.67}$ →
→ 0.81

city	
mum	0
del	1
hyd	2

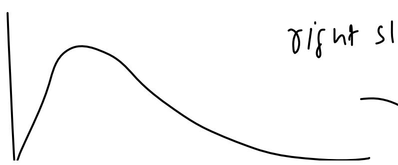
city_mum	city_del	city_hyd
1	0	0
0	1	0
0	0	1

sector → 104 → 104 cols

532

Linear reg

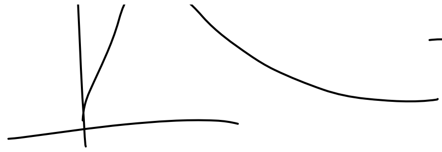
→ OHE



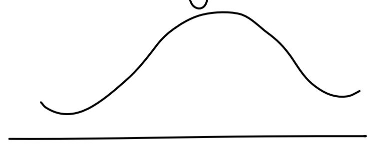
right skew

→ log

→ OHE
→ scaling
→ log transformation



→ log



0.4C

$\boxed{0\ 64\ \text{Cr}}$ → $\boxed{64\ \text{lacs}}$

3 BHK Sector 82 → 1 Cr
. . . → 1.6 Cr

- → diff algo
- → hyperparameter
- → feature engi
- → more data

→ website → deploy

$y \rightarrow \log(y)$

$\downarrow$

$e^{\log(y)}$

$\downarrow$

y

Real estate analysis website

7

→ price prediction → ?
→ analysis
→ recommend
→ insights

gather
↳ preprocess
↳ fe
↳ outlier
↳ missing
↳ feature selection

encoding

cat cols → numbers

exp.

tree — Ordinal encoding → models → r2 score (0.89) ↳ 0.51

linear — OHE → model → r2 score → 0.89 / 0.46 — linear

PCA

tree — Target encoding → 0.90 / 0.45

Ordinal encoding → linear

property type → tree

flat → 0
house → 1
plot → 2
com → 3

0.94 → 0.51

2 cr → 2.9 cr, 1.1 cr

OHE

property type
flat

0

OHE

flat	house	plot
1	0	0
0	1	0

flat → 0 → OHE 0 1 | 1 0
house → 1 0 0 | 0 1
plot 2

OHE → cat (50) → 100 new cols appear

LE →

Pipeline → OHE → PCA

OHE → dim → 120 cols → 3000 cols

1200 cols → 3 L

95% explained

99 cols

dimension red → PCA

sector → sec1 | sec2 | . . . | sec 100

sector (100 cate)

target encoding

sector		
Sec 1	—	2
Sec 49	—	3
sec 1	—	4
sec 102	—	1
sec 49	—	6

sector
3
4.5
1

model selections
- tree based → rf | xgboost
- target encoding (sec)

[0, 90] → 0.93

0.45 → 0.35

→ target encoding (seu)

hyperparameter

rf → default settings

Xgboost → Bayesian hy
↳ Hyperopt

45L mae NO →

25L NO →

→ more data (3000 rows)

45L → mae → Xgboost → hyperopt

→ revisit the entire process

45L (22, 22)

3.22

3 Cr

3.44 Cr

pipeline

pred.

[pred —] → output

$\boxed{\text{Pra} -} \longrightarrow$ Output

Analytics Module

15 September 2023 15:41

- geo map
- word cloud amenities
- scatterplot -> area vs price
- pie chart bhk filter by sector
- side by side boxplot bedroom price
- distplot of price of flat and house

features

price

area

Projects

Price pred

Analysis module

eda

easy sample

Insights module

Recommender

[lat long]

geo map

sec 1 → avg price_per_sqft → geo map

sec 2

.

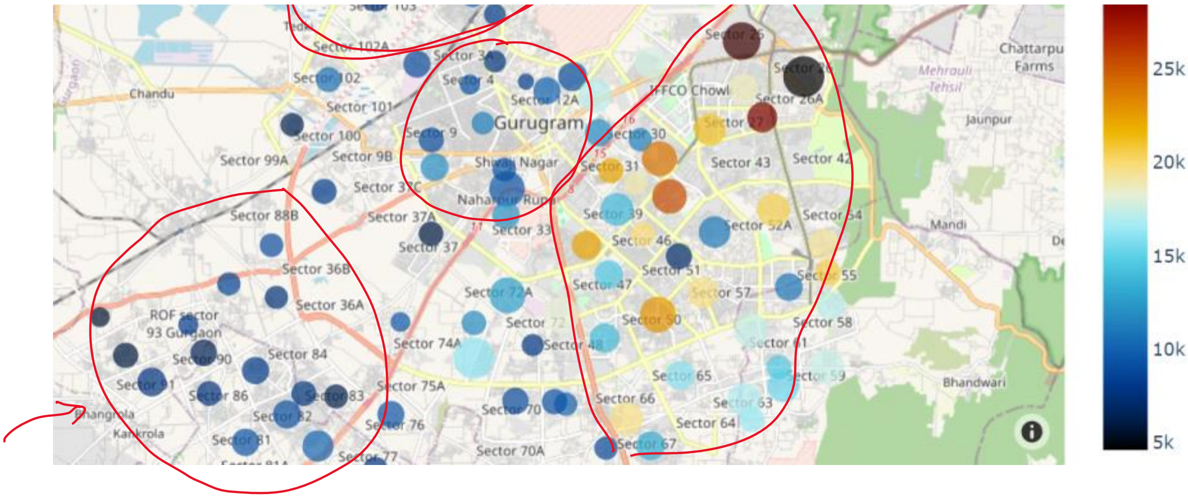
.

.

sec 115

→ plot on geo map

delhi



[1, 2, 3]

[4, 1, 2]

[1, 4, 5] → [1, 2, 3, 4, 1, 2, 1, 4, 5 - - -]

[ ]

String

Recommender System

19 September 2023 19:24

Recommender systems are a subclass of information filtering systems that aim to predict the "rating" or "preference" a user would give to an item. There are several types of recommender systems, each with its own strengths and weaknesses:

1. Collaborative Filtering (CF) Recommender Systems:
   - User-Based Collaborative Filtering (UBCF): This method finds users that are similar to the target user and recommends items that those similar users have liked. It's based on the assumption that users who have agreed in the past tend to agree again in the future.
   - Item-Based Collaborative Filtering (IBCF): Instead of finding user similarities, IBCF finds item similarities. If a user likes a particular item, they will likely also like other items that are similar to it.
2. Content-Based Recommender Systems:
   - These systems recommend items by comparing the content of the items and a user profile. Content can be described in terms of several descriptors or terms that are inherent to the item (e.g., a book might be described by its author, its genre, etc.). If a user has interacted positively with certain content attributes in the past, the system will recommend new items with similar attributes.
3. Hybrid Recommender Systems:
   - These systems combine the strengths of both CF and content-based methods. There are several ways to design hybrid systems, such as by making predictions separately with each approach and combining them, adding collaborative and content-based features into a single model, or unifying the models into a single model.

rustam

olx

→ ranking system → web → static web

amazon

fb/insta feed

2005 → ranking system → google

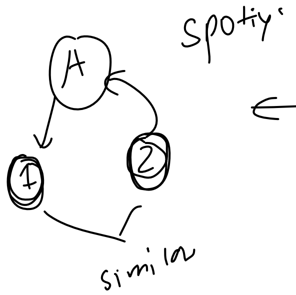
Chatgpt

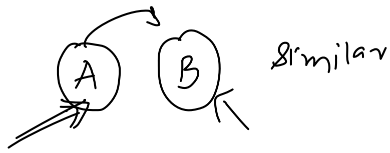
Ola → driver

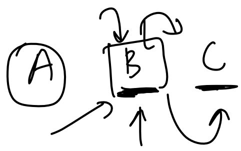
p1 products | user u1

rating

u1 p1 3
u3 p3 4







19 September 2023 19:39

Capstone

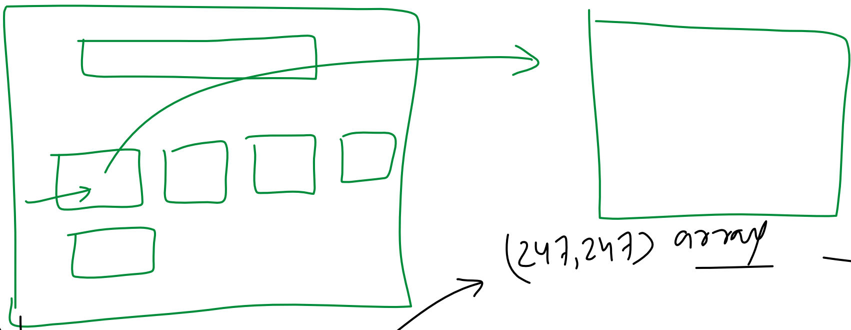
Price pred

Analytics

Recommender

Insights

Deploy



(247,247) array

5 appr

DLF The Arbour

W_1 Loc advan

W_2 Price details

W_3 Top Facility

247,247

array

W

1 2 3

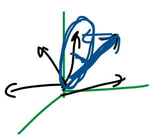
247,...

247 vectors in a 953 dim space

Text vectorization

vector → set of numb

each



BoW

Tf Idf

wordvec

247 appr

6 dim

953

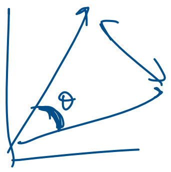
[1 0 3 4 5 1]

[2 0 4 1 3 2]

[3 1 1 1 2 1]

247,247

953 → angular



247

953 → angular

cosine simi

	V1	V2	V3	. .	V247
V1	0.98	0.23	0.4		- -
V2	- -	- -			
V3	-	=	—		—
⋮	-				
V247					

(247,6)

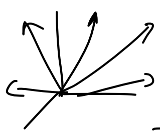
vectorize → vector

247 rows

```
{
'3 BHK': {'building_type': 'Apartment', 'area_type': 'Super Built-up Area', 'area': '1,605 - 2,170 sq.ft.', 'price-range': '₹ 2.2 - 3.03 Cr'},
'4 BHK': {'building_type': 'Apartment', 'area_type': 'Super Built-up Area', 'area': '2,248 - 2,670 sq.ft.', 'price-range': '₹ 3.08 - 3.73 Cr'}
}
```

1	2	3	4	5	6
building_type_3bk	area_3bhk	price_3bk	buil_type 4bhk	area 4bk	price 4bhk
Apartment	1605	2.2	Apart	2248	3.08
—	—	—	—	—	—
house	—	—	—		
villa					

OHE



(247,247) → simi score

vectors

ndim

	L1	L2	L3	. . .	Ln
A1	0.8	2.5	3.1		Null
A2	0.7	Null	—		-
A3	—	—	—		-
⋮	—	-	—		
A247					

1070 dim

246 vector

247.247

246

(246,246)

246

Apparments

Apparment

0.2 facilities 0.5 pricing 0.8 location

simi

245

+ +

245 new sim (5) (10) room

how well the performance

prepro

similar

Indira Gandhi Air

IGIA

NH8

National hwy 8.

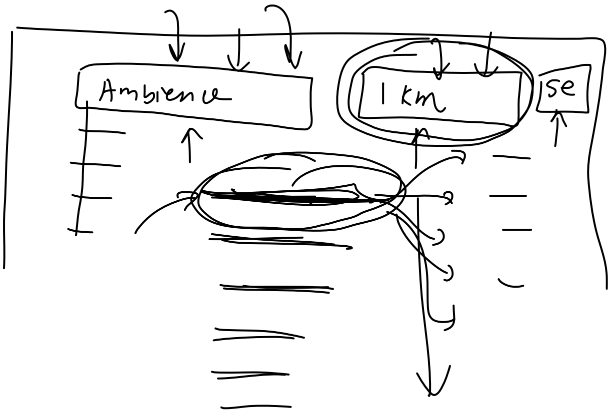
fuzzy

fuzzy

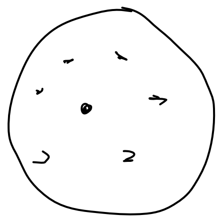
website

Ambience

landmark



location based filtering



Capstone Project

- Prediction
- Anal
- Reco
- Insights module → Deploy → AWS

houses

X Y

understanding

Insights module

supervised ml

- prediction
- Inference

X → y

X Y

new X → new y

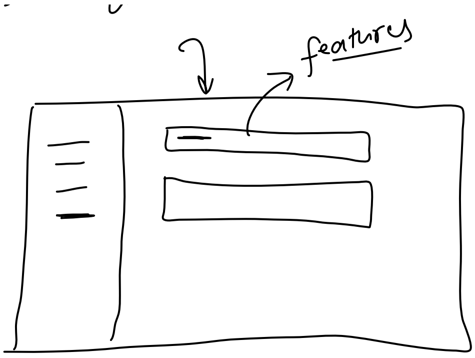
sector → price ↑ (by percentage)

sec 49 → sec 28 → 16 %

area → 100 sqft →

features

1 2 → x %



1-2 → x %

sector → secu9 → price

linear

rf

feature

$x_1 \longrightarrow x_n$

model train → inference

deep learn

↓

Black box model

Boosting / Bagging

↳ feature importance

→ Linear reg →

$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n$

↓

linear reg

data → lr → reg analysis → inference → website

coefs

bedroom price

1 → 2

1 cr

$x_1 \mid x_2 \mid \ldots \mid x_n \mid Y$

$\beta_2 = 0.5$

$Y \rightarrow \log(y)$

$X \rightarrow$ X_scale

$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n x_n$

1 unit

bedRoom	0.054002

From <http://localhost:8888/notebooks/dsmp-capstone-project/insights-module.ipynb#>

1+0.5 1.5+0.5

$x_2 = 1$ $y = 1$ $\beta_2 = 0.5$

↓

2 $y = 1.5$

↓

3 2

$\beta_0 \; \beta_1 \; \beta_2 \ldots \beta_n$

↓

1 → 2 → 3 → 4

1cr 1+0.55

= 1cr 5la

1cr 10la lin..

(a) (b) σ_b

stan - unstan std(bed)

unstandardis bedroom

(a) Stan coef (bed) = unstan coef (bed) (b) × $\frac{\text{std(bed)}}{\text{std}(\log(y))}$ σ_b σ_p

$\boxed{b = 0.024} \rightarrow \boxed{e^b}$

$$\boxed{b = \frac{a}{0.054} \times \frac{\sigma_p}{\sigma_b}} \quad \begin{matrix} \rightarrow 0.557 \\ \rightarrow 1.245 \end{matrix}$$

y → log(y)

y

$e^{\log(y)}$

coef → bed → $\boxed{0.0243}$

bedroom $\boxed{0.024}$

1 → 2

1 — 2 — 3 → $\boxed{1.02\ Cr}$

1 Cr 1.04

(a) (b)

0.210

$$b = 0.210 \times \frac{\sigma_p}{\sigma_{area}} \quad \begin{matrix} \rightarrow 0.557 \\ 1216 \end{matrix}$$

$\log(y) \rightarrow e^{\log(y)} \longrightarrow y$

$$SC = UC \times \frac{\sigma_x}{\sigma_y} \Rightarrow \boxed{UC = SC \frac{\sigma_y}{\sigma_x}}$$

45 mins

Prediction Analytics Recommender $\boxed{\text{Insights}}$ → park

Deploy → AWS → Share

Heroku

Azure

Azure

GCP

AWS → EC2 → virtual server

AWS ↓ cloud service provider

EC2 → own → software → python

SSH login → remote

SSH login ↓ passw

my laptop

8501

ec2

streamlit

TCP

SSH

web scraping
selenium

Xgboost

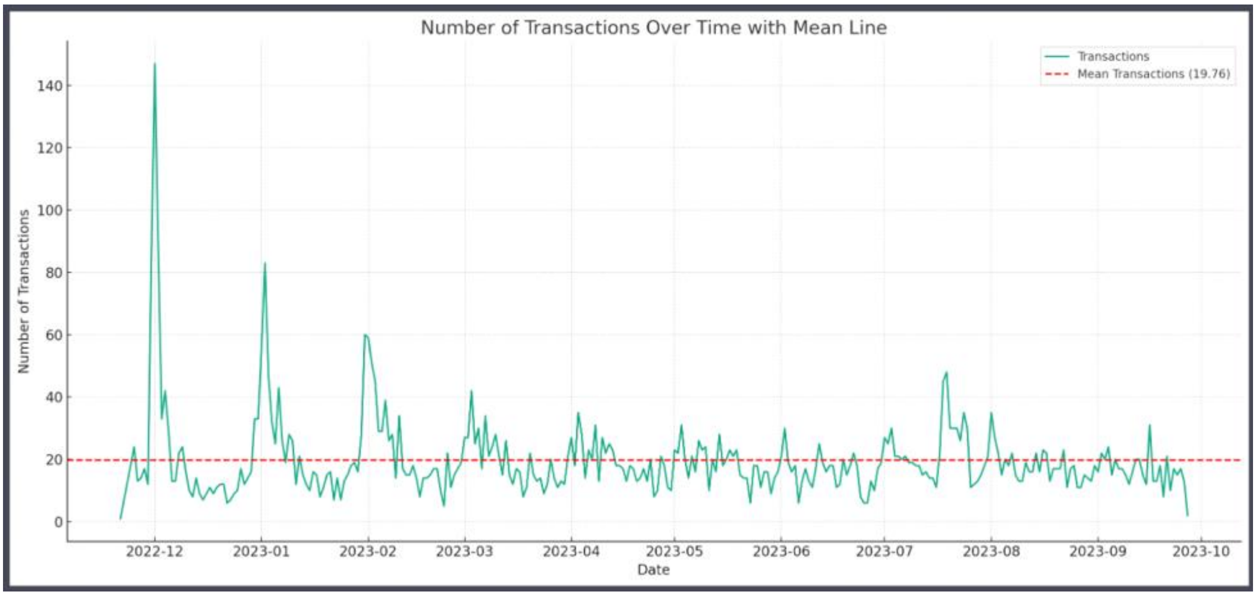
Kmeans

DBScan

Tsne

feature engineering

leftover


Number of Transactions Over Time with Mean Line
Transactions
Mean Transactions (19.76)
Number of Transactions
140
120
100
80
60
40
20
0
2022-12
2023-01
2023-02
2023-03
2023-04
2023-05
2023-06
2023-07
2023-08
2023-09
2023-10
Date