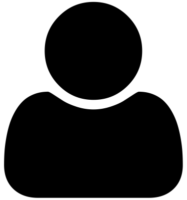
The boundaries between the good and the bad are always blurred

Introduction to classification

What is a classifier?



Movie 1
Movie 2
Movie 3

Binary
Multi-Class
Multi-Label (Special case)

The basic construct - Rows, Columns and the Labels

Day wise rows

Date	Avg Temperature	Previous Day Rain	Humidity	….	Rain or Not
20-Nov	27	Yes	35%		Yes
21-Nov	29	Yes	40%		No
22-Nov	30	No	37%		No
23-Nov	28	No	38%		No

Features

Labels

Given data like average temperature (for the first 3 hours), previous day rain, humidity etc predict the rain for the day

The useful probabilities

Date	Avg Temperature	Previous Day Rain	Humidity	….	P(Rain)
24-Nov	30	No	38%		80%
25-Nov	30	Yes	42%		10%
26-Nov	31	No	39%		30%
27-Nov	27	No	36%		40%

The different probabilities can help us take more granular action →

<10% - Plan for Picnic whole day
10 -30% - Picnic nearby place
30 - 50% - A long walk
>50% - Be indoors, finish some work today

Multiclass Problem

Date	Avg Temperature	Previous Day Rain	Humidity	….	Level of Rain
15-Nov	23	Yes	37%		High
16-Nov	27	Yes	42%		Medium
17-Nov	30	Yes	37%		Low
18-Nov	28	No	38%		Low

Day wise rows

Features

Labels

The useful probabilities

Date	Avg Temperature	Previous Day Rain	Humidity	….	P(Low Rain)	P(Med Rain)	P(High Rain)
19-Nov	32	No	38%		10%	70%	20%
20-Nov	32	Yes	42%		90%	3%	2%
21-Nov	30	No	39%		70%	15%	15%
22-Nov	27	No	36%		60%	20%	20%

Case Study 1

Credit card company wants to pre-approve its customers. It has many relevant details about the customers, we need to decide whether they should approve or not

Rows: Customers

Columns: CIBIL score, user location, age, income, gender, income, device etc

Labels: ‘Good’, ‘Bad’

As we start….

1. Look at the data - Understand the data
2. Pay attention to Y labels and distribution
3. EDA
4. Any quick features that you can think of
5. Train test split
6. Label encoding
7. Model
8. Investigate
9. Explain


Cut over to Code

Look at the data

income	age	experience	bureau_score	married	house_ownership	car_ownership	risk_flag	profession
2514921	31.0	4.0	651.0	single	rented	no	0	Psychologist
7047674	28.0	4.0	526.0	single	rented	yes	0	Economist
2749317	30.0	2.0	526.0	single	rented	no	0	Secretary
7378274	24.0	0.0	764.0	single	rented	no	0	Flight attendant
9574585	27.0	5.0	739.0	single	rented	yes	0	Technician

city	state	current_job_years	current_house_years	device
Chandrapur	Maharashtra	4.0	14.0	Oppo
Ramagundam[27]	Telangana	3.0	13.0	Xiaomi
Ramagundam[27]	Telangana	2.0	14.0	samsung
Adoni	Andhra Pradesh	0.0	11.0	samsung
Imphal	Manipur	5.0	10.0	Vivo

1. Numeric
2. Low level categorical columns
3. Problematic Columns - high granularity categorical columns

Pay attention to Y-Labels

```
0    236567
1     43433
Name: risk_flag, dtype: int64
```

```
0    0.844882
1    0.155118
Name: risk_flag, dtype: float64
```

Slight class imbalance

Expected in real world datasets

Baseline accuracy is at 84% :)

EDA : Exploratory data analysis

1. Attempt to explore and understand data
2. Univariate Distributions
3. Bi-Variate Distributions
4. Abnormalities
5. Key tipping points or categorical values that determine outcome


Cut over to Code

EDA

EDA with Sweetviz

Html : [here]

PDF Annotated: [here]

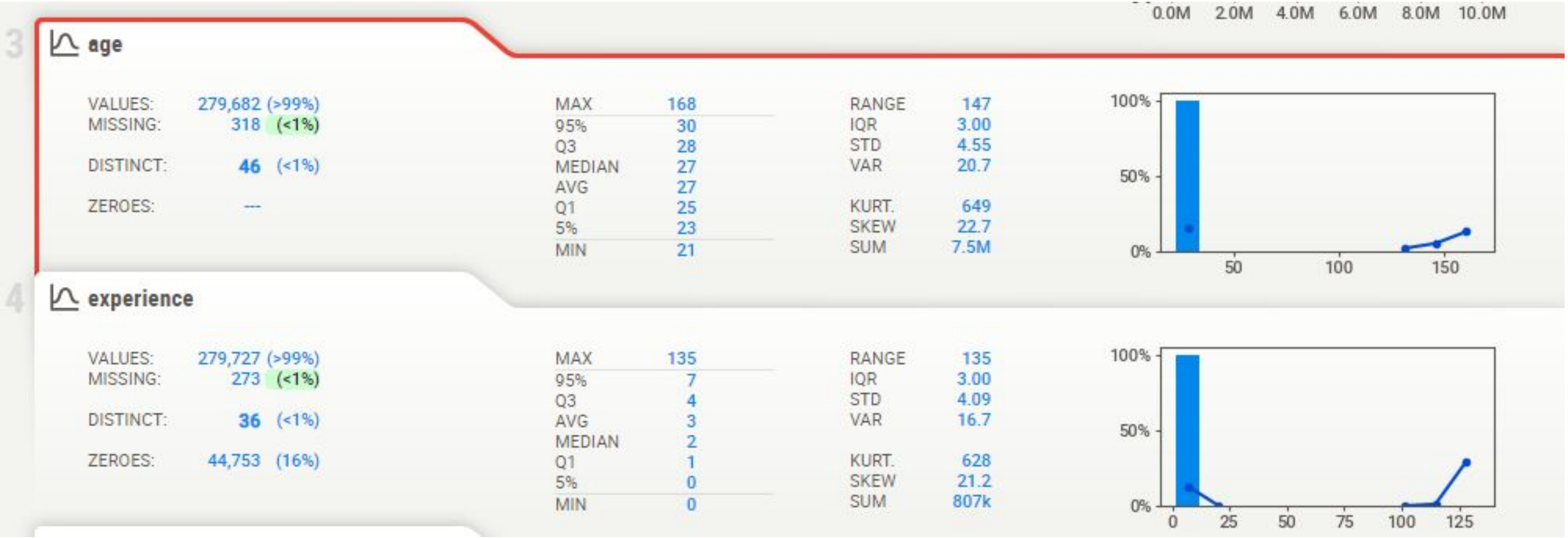
Observations

Key trends

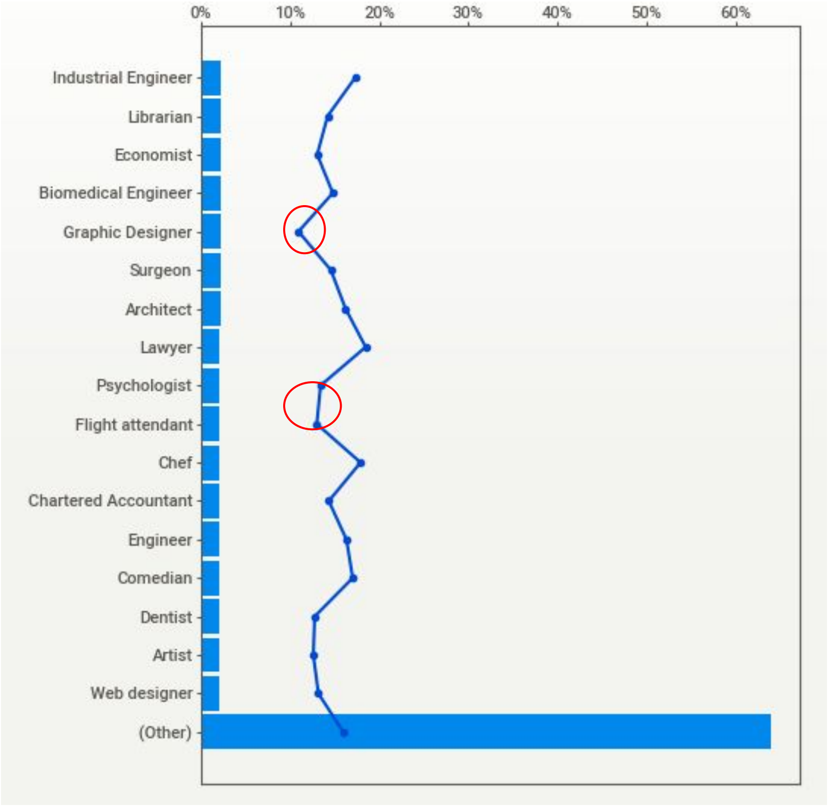
1 . Age (without outlier), Income, Bureau Score, house years

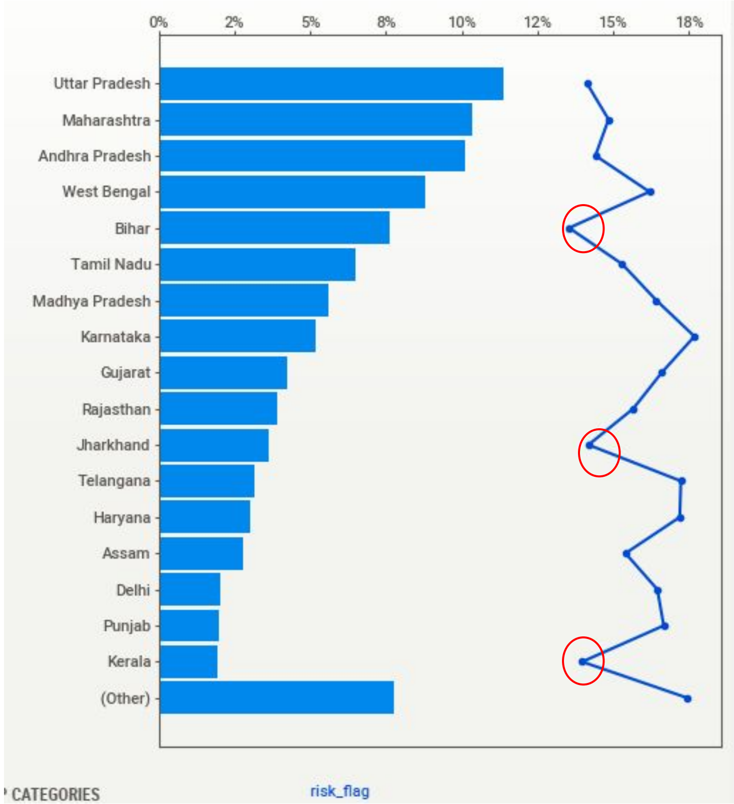
2. Favorable trend for car ownership, married,

EDA - noticing abnormalities

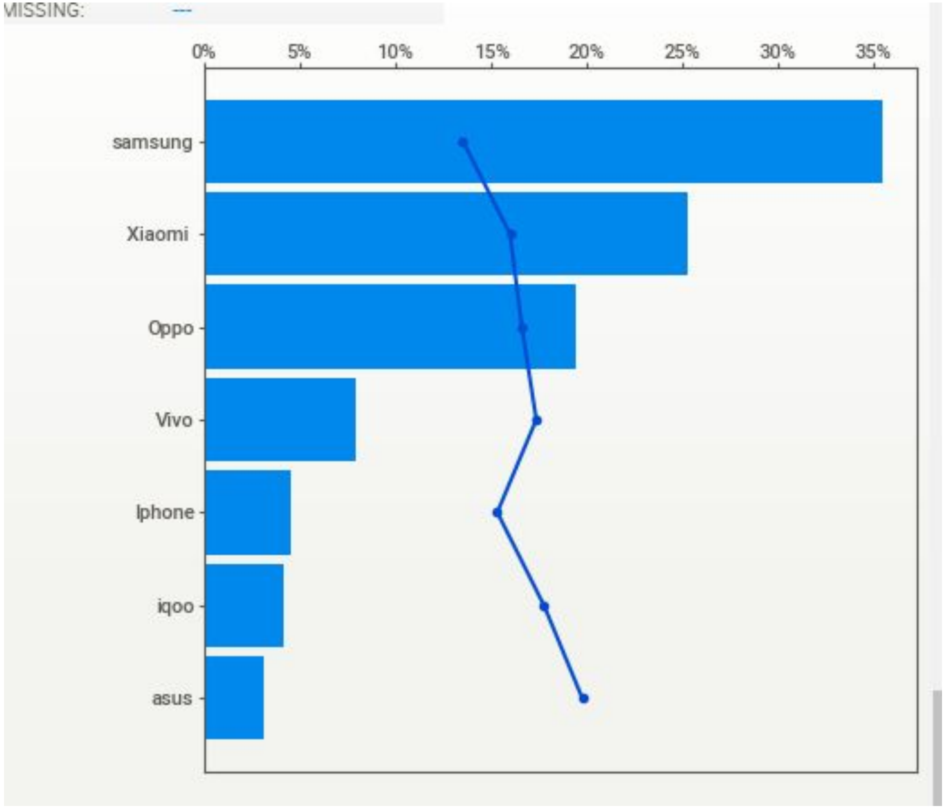


EDA - Granular categories





EDA - Granular Categories



Exercise -

What are the insights you can derive?

Any new features you could think of?

https://docs.google.com/document/d/1YBzdB-crWZLv8t-qbtdZrYc8RBOO27iJf7hngkh_AyA/edit

Plan for features - Categorical

Heuristic way: Based on prior knowledge

Bi-variate plot based:

1. Combine based on frequency and risk propensity (For instance low risk states and high risk states. Low risk professions and high risk professions)

Putting the onus back on the ML model

2. Simple Label encoding (use this only with trees) ✓
3. One hot encoding ✓
4. Vectorizers
5. Combination - (Low granular ones - One hot, High granular ones - Vectorizers)

Example Label Encoding

	profession	married	house_ownership	car_ownership	city	state
0	23	1	2	0	94	25
1	8	1	2	0	219	18
2	41	1	2	0	131	2
3	17	1	2	0	52	13
4	47	0	1	1	14	14

One hot encode all the variables

	x0_air traffic controller	x0_analyst	x0_architect	x0_army officer	x0_artist	x0_aviator	x
0	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	
1	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	
2	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	
3	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	
4	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	

5 rows × 402 columns

Plan for features: Numeric Processing

Missing Values

1. Come up with hard-coded values for each variable (0,1,100,-9999)
2. Mean/Median/PX Impute
3. Using in XGB

Outliers

1. Come up with hard-coded values for each variable
2. Capping
3. Dropping values
4. Using in XGB

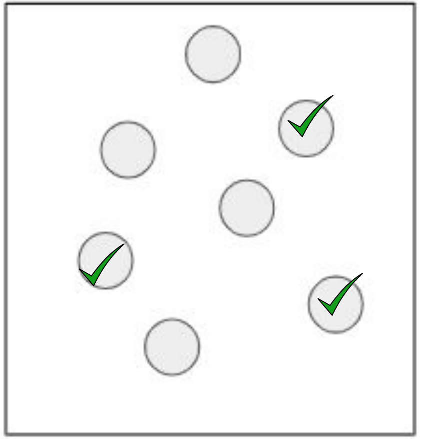
Missing Values Impute Example

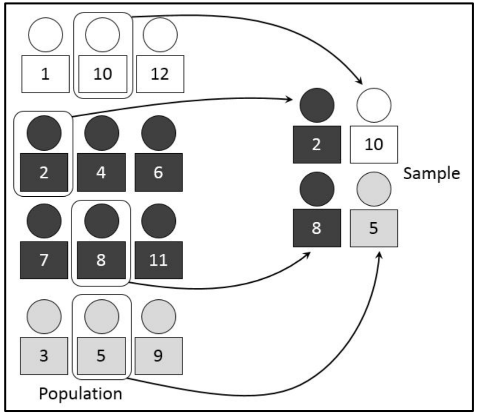
```
income                  0
age                   211
experience            187
current_job_years       0
current_house_years     0
bureau_score           31
dtype: int64
```

	index	Unnamed: 0	income	age	experience
501	212332	99856	373948	nan	nan
1622	38707	99864	1263508	nan	nan
2605	16703	99893	3001270	nan	nan
2831	91802	99808	1189098	nan	3.00000
3065	67802	99808	1189098	nan	3.00000

	income	age
501	373948.00000	26.66658
1622	1263508.00000	26.66658
2605	3001270.00000	26.66658
2831	1189098.00000	26.66658
3065	1189098.00000	26.66658

Split Data





Random Split

Stratified Split

In Sample

Out of Sample

Out of time Sample

Process Features after train-test split to avoid leakage


Cut over to Code

Build Model

Logistic Regression

Random Forest Trees

XGB Trees ✓

SVM

Features : Selected numeric (Age, Income, Experience, etc) + Label Encoded Categorical (One hot encoding, numeric with imputation)

Label encoded variable : risk_flag

Class Weights : For class imbalance. **Risk - 85% vs 15%**

ML models are very sensitive to distribution. Class weights penalize misclassification on poorer classes


Cut over to Code

Investigate : Measure & Measure & Measure

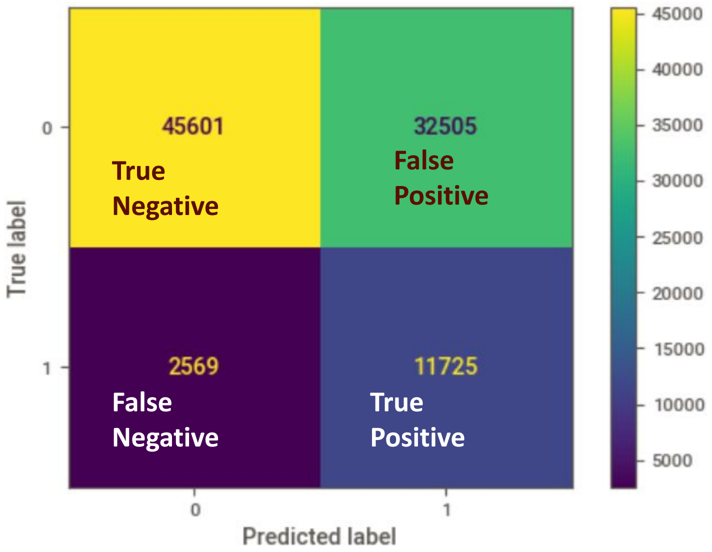
Baseline :

1. We need baseline - current logic or a naive logic.
2. Measurement over baseline

Classification

Confusion Matrix, F1 Score (Macro, Micro)
AUC
AUC PR

Metrics Explained more..



False Positive - We are rejecting customers, when we should not be. Impacts user experience and Growth

False Negative- We are accepting customers, when we should be rejecting.

TPR = TP/(TP+FN)
FPR = FP/(FP+TN)

Capturing the 'bad' customers at the cost of 'good'

Precision = TP/(TP+FP)
Recall = TP/(TP+FN)

F1 = 2*P*R/(P+R)

Higher the better

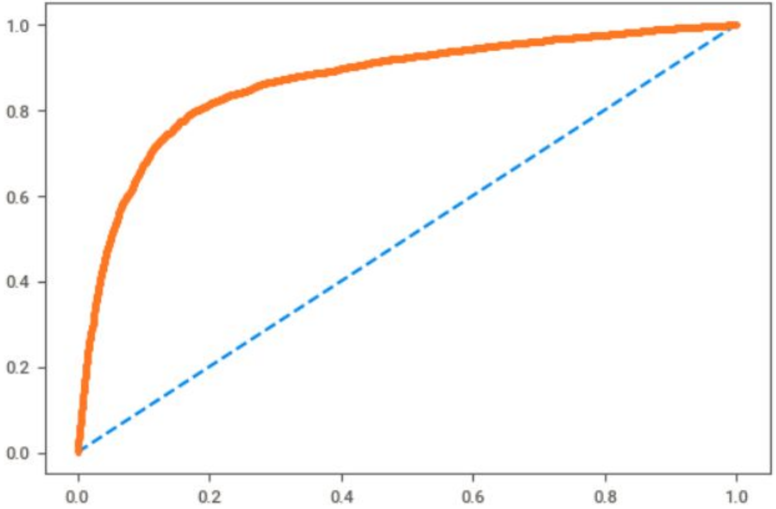
This model is aggressive and if thresholds are not balanced well, it could result in user dissatisfaction

How do we select the model?

Compare the AUCs and AUC PR of different model

AUC = Area under the curve

TP Rate

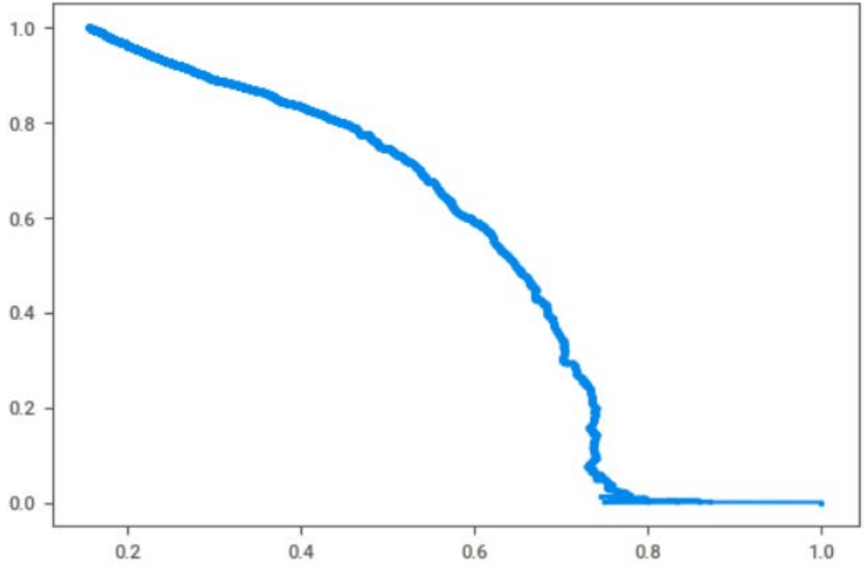


FP Rate

Area under the curve (Random Baseline 0.5).

Signifies how well the model ranks ‘bad’ over ‘good’

How do we select the model? - AUC PR



Recall

Precision

AUC-PR - Single Value to measure > Positive Rate

```
0    0.84530
1    0.15470
Name: risk_flag, dtype: float64
```

Positive Rate


Cut over to Code

Explain the model

1. Get Feature importances - SHAP is proven better to explain
2. Find correlations
3. Think and see if these make sense
4. Are there features that is explaining too much of variation?. What would happen if you drop that variable checkk


Cut over to Code

Metrics comparing with methods

	accuracy	auc	f1	auc_pr	model
0	0.75255	0.86704	0.67524	0.58405	xgb_imp
0	0.56898	0.77328	0.52425	0.41183	xgb_ohe
0	0.75486	0.86746	0.67629	0.58456	xgb

Conclusion

Converting the metrics to business related cost function

1. False Positives - impact user retention?, if so what is the cost.
2. False Negatives - Higher cost of default

The Model is explainable and no leakage is found

Pay attention to interventions - Hard decline, Decline with reason.

Pilot, Deploy and Listen to users