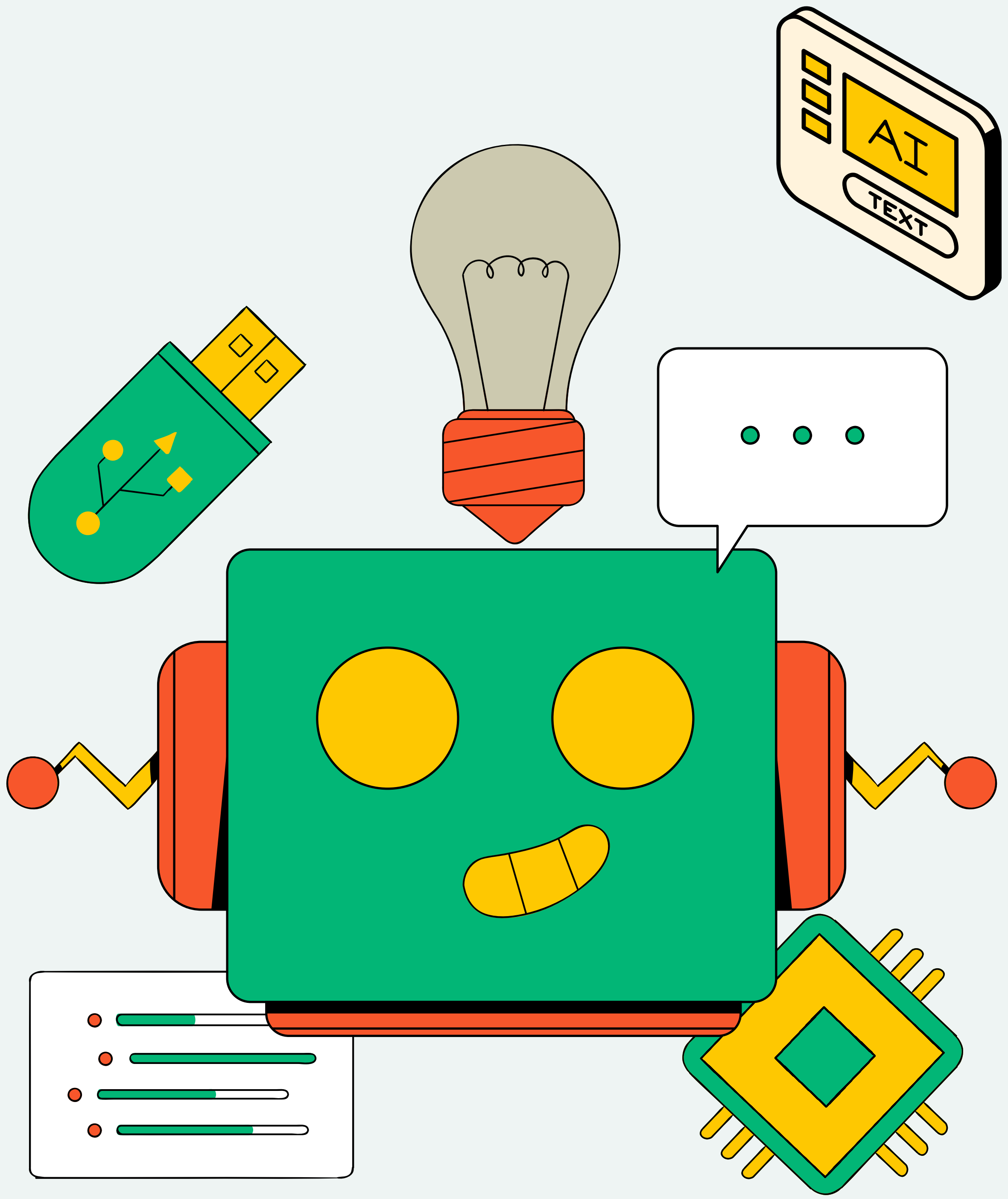
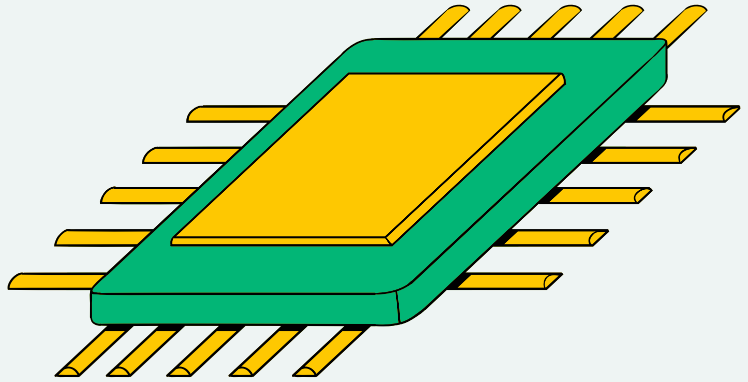
PURWADHIKA
DIGITAL TALENT INCUBATOR (DTI)



TELCO CUSTOMER CHURN

PRESENTATION

PRESENTED BY:
TIO SYAIFUDDIN



PRESENTATION OUTLINE

- Introduction
- What is Churn Rate?
- Traditional VS ML Approach
- Dataset
- Data Cleansing
- Exploratory Data Analysis
- Data Preprocessing
- Modeling
- Results

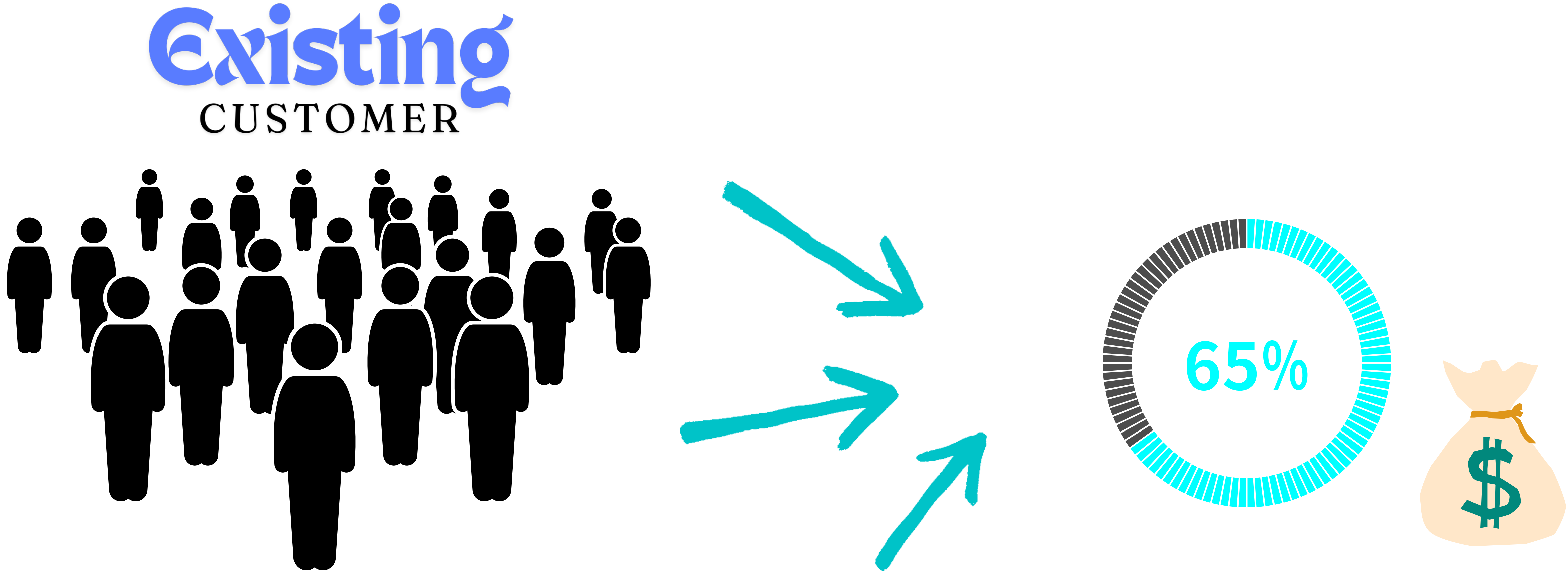
INTRODUCTION

With over 230 million mobile internet users across the country, **Indonesia's telecommunications market** is among the largest in Southeast Asia.

In 2022, the information and communications sector contributed over 800 trillion Indonesian rupiah to the national gross domestic product (GDP).

WHAT IS CHURN RATE?

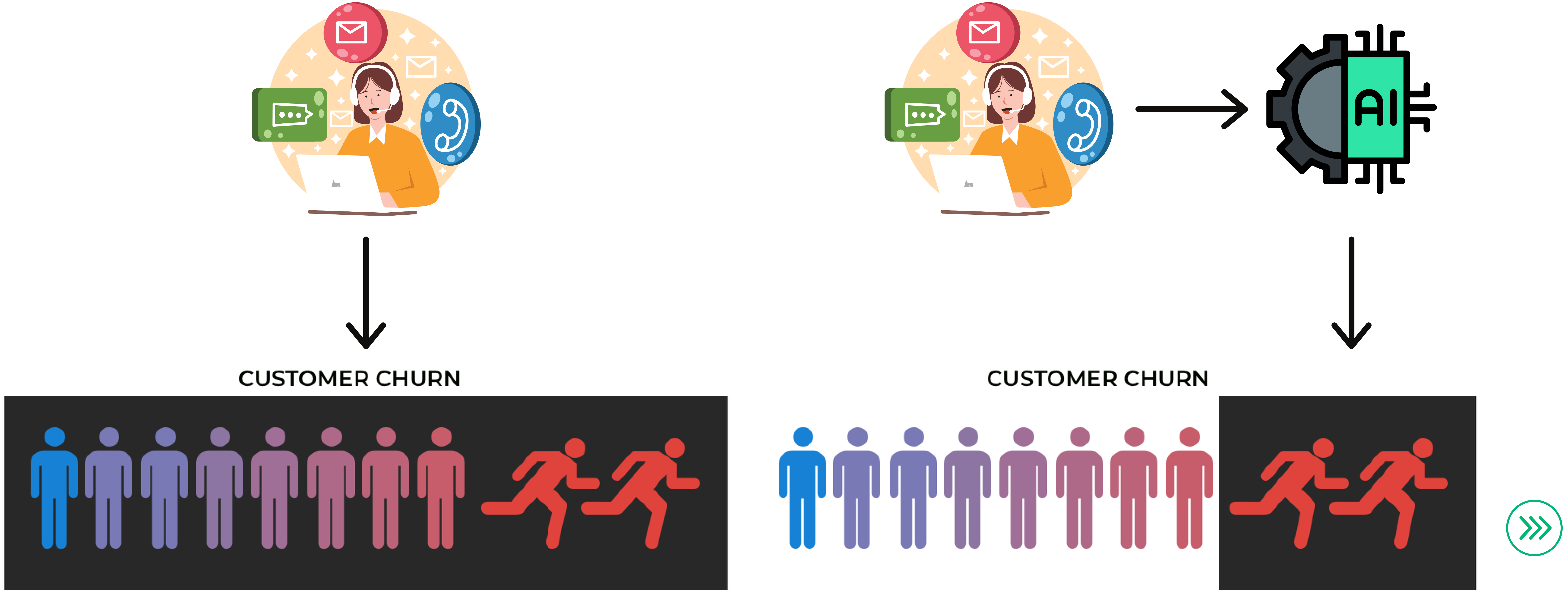
The churn rate is “the percentage of customers that stopped using your company’s product or service during a certain time frame” (Source: Hubspot).



HOW TO KEEP THESE EXISTING CUSTOMERS?

Telco companies usually have plenty of prevention strategies such as personalized offers, customer engagement, feedback mechanisms, and much more. It is crucial to manage the budget, so it gets into the right hands.

TRADITIONAL VS MACHINE LEARNING APPROACH



BENEFIT OF USING MACHINE LEARNING

Cost Efficiency

Personalized Retention Strategies

Enhanced Customer Experience

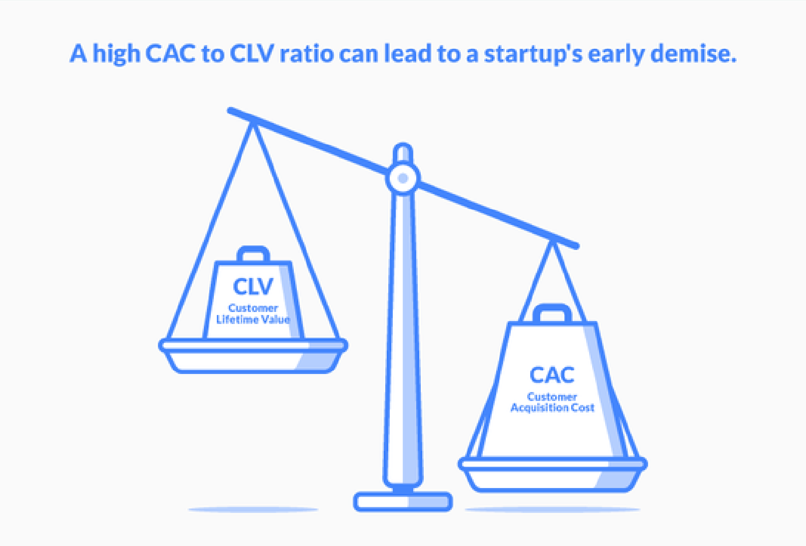
EVALUATION METRIC

Here we define the positive class as a churn customers with this detail.
– False Positive (FP): Predict a customer will churn, while it is not
– False Negative (FN): Predict a customer will not churn, while it is churn

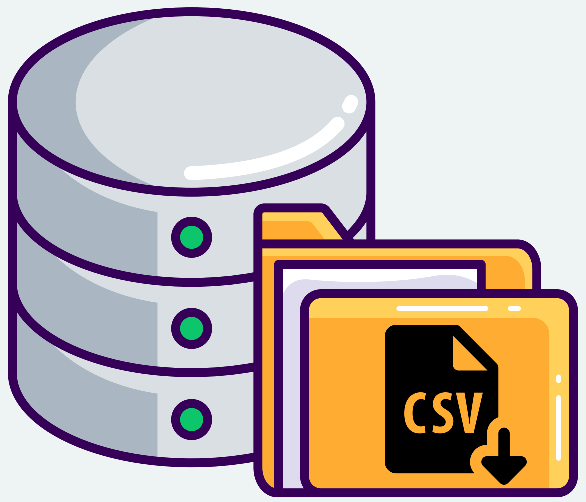
Giving false prediction will result in these costs.
– FP cost: Waste of promotional budget
– FN cost: Lost of customers

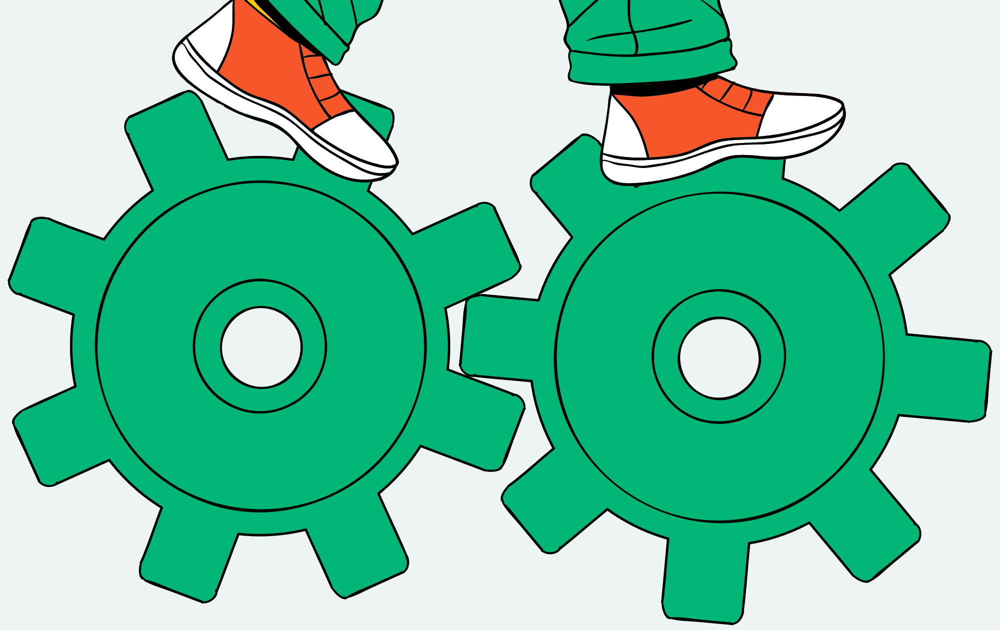
		Predicted	
		No	Yes
Actual	No	True Negatives Predicted Customers Stay Customers Actually Stay	False Positives Predicted Customers Churn Customers Actually Stay
	Yes	False Negative Predicted Customers Stay Customers Actually Churn	True Positives Predicted Customers Churn Customers Actually Churn

$$F_2 = \frac{5}{\frac{4}{Precision} + \frac{1}{Recall}} = \frac{5}{\frac{4TP+4FN}{TP} + \frac{TP+FP}{TP}}$$



DATASET





	feature	data_type	null	negative	n_unique	sample_unique
0	Dependents	object	0.0	False	2	[Yes, No]
1	tenure	int64	0.0	False	73	[9, 14, 64]
2	OnlineSecurity	object	0.0	False	3	[No, Yes, No internet service]
3	OnlineBackup	object	0.0	False	3	[No, Yes, No internet service]
4	InternetService	object	0.0	False	3	[DSL, Fiber optic, No]
5	DeviceProtection	object	0.0	False	3	[Yes, No internet service, No]
6	TechSupport	object	0.0	False	3	[Yes, No, No internet service]
7	Contract	object	0.0	False	3	[Month-to-month, Two year, One year]
8	PaperlessBilling	object	0.0	False	2	[Yes, No]
9	MonthlyCharges	float64	0.0	False	1422	[72.9, 82.65, 47.85]
10	Churn	object	0.0	False	2	[Yes, No]

Num of rows : 4930

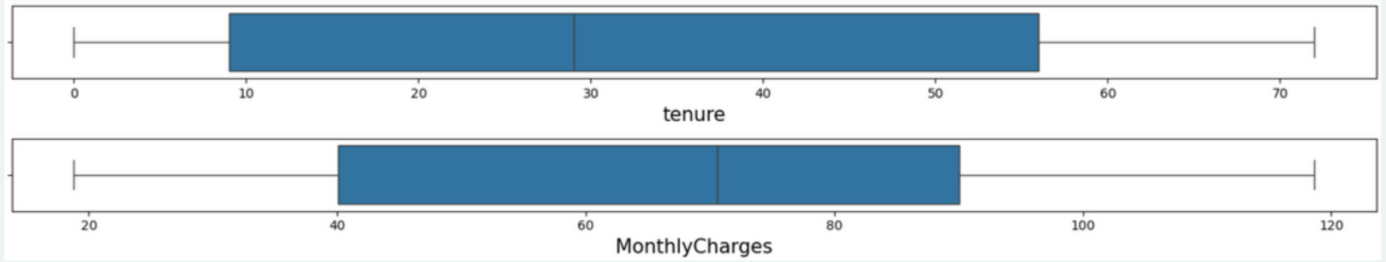
Num of columns : 11

Target variable : Churn

Qualitative feature : 8

quantitative feature: 2

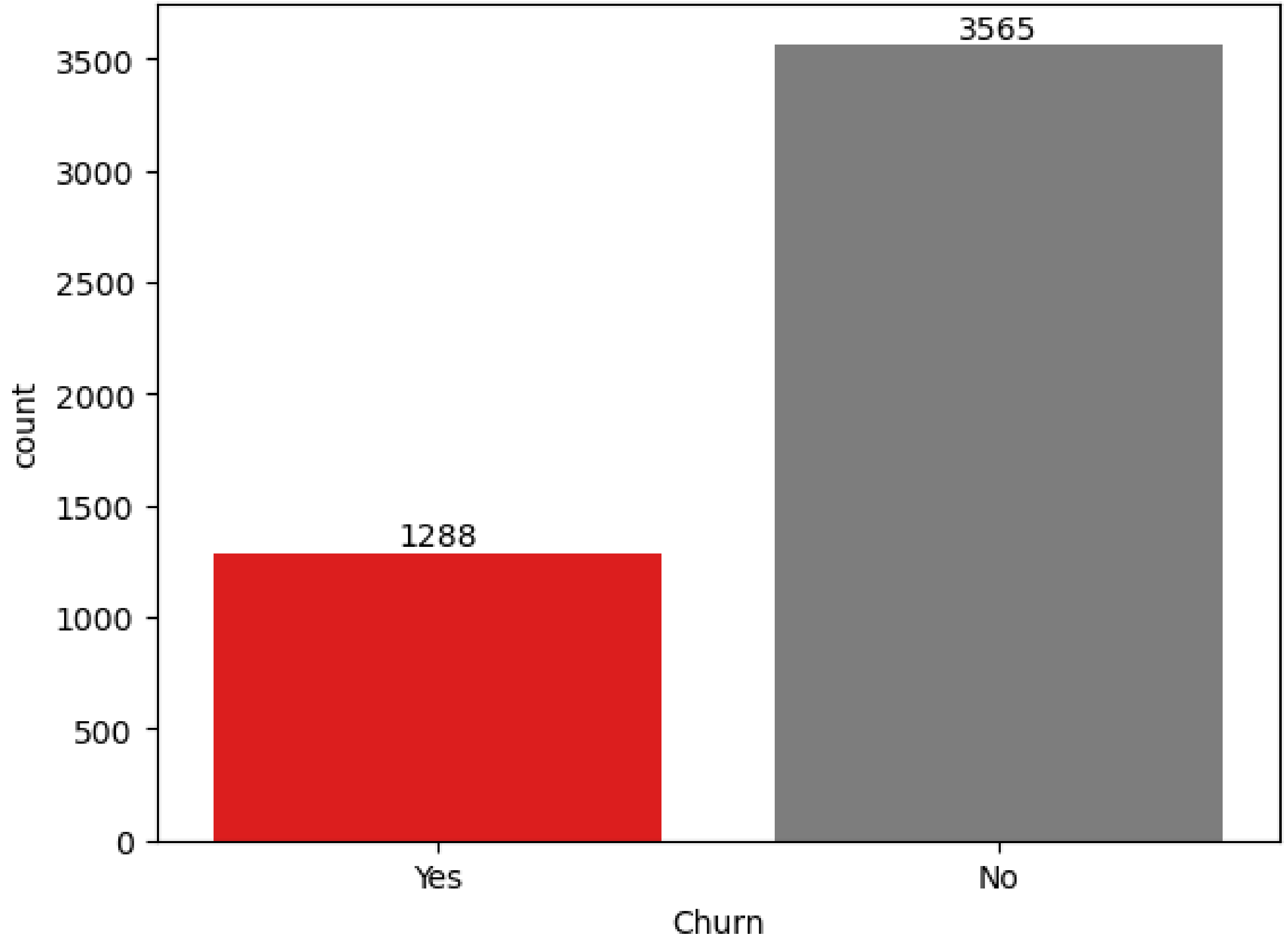
DATA CLEANSING

Cleaning Type	Assessment	Action
Duplication	77 (1.56% of total)	Drop
Missing values	None	None
Negative values	None	None
Outliers		None

EXPLORATORY DATA ANALYSIS

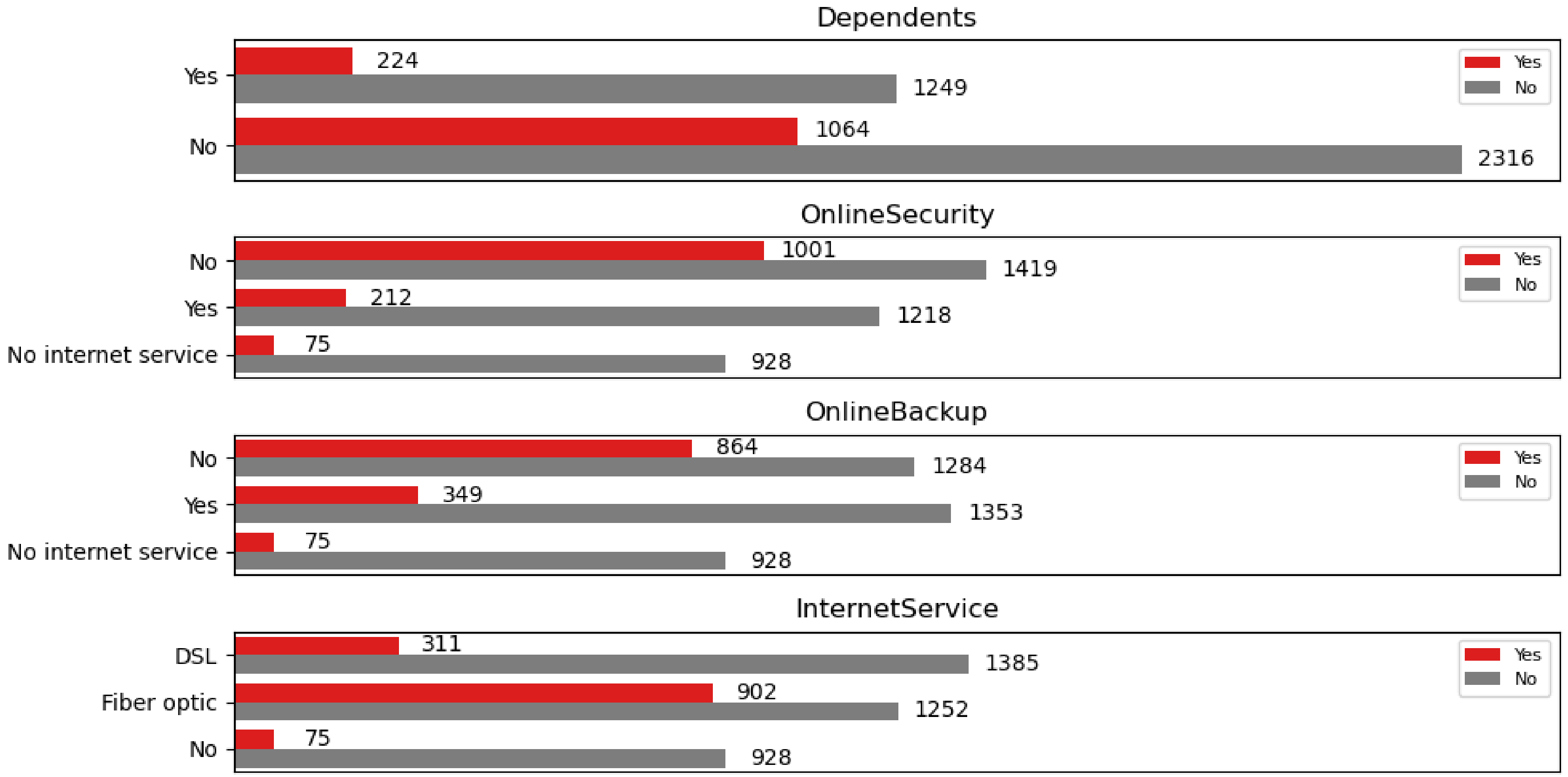
Imbalance target variable

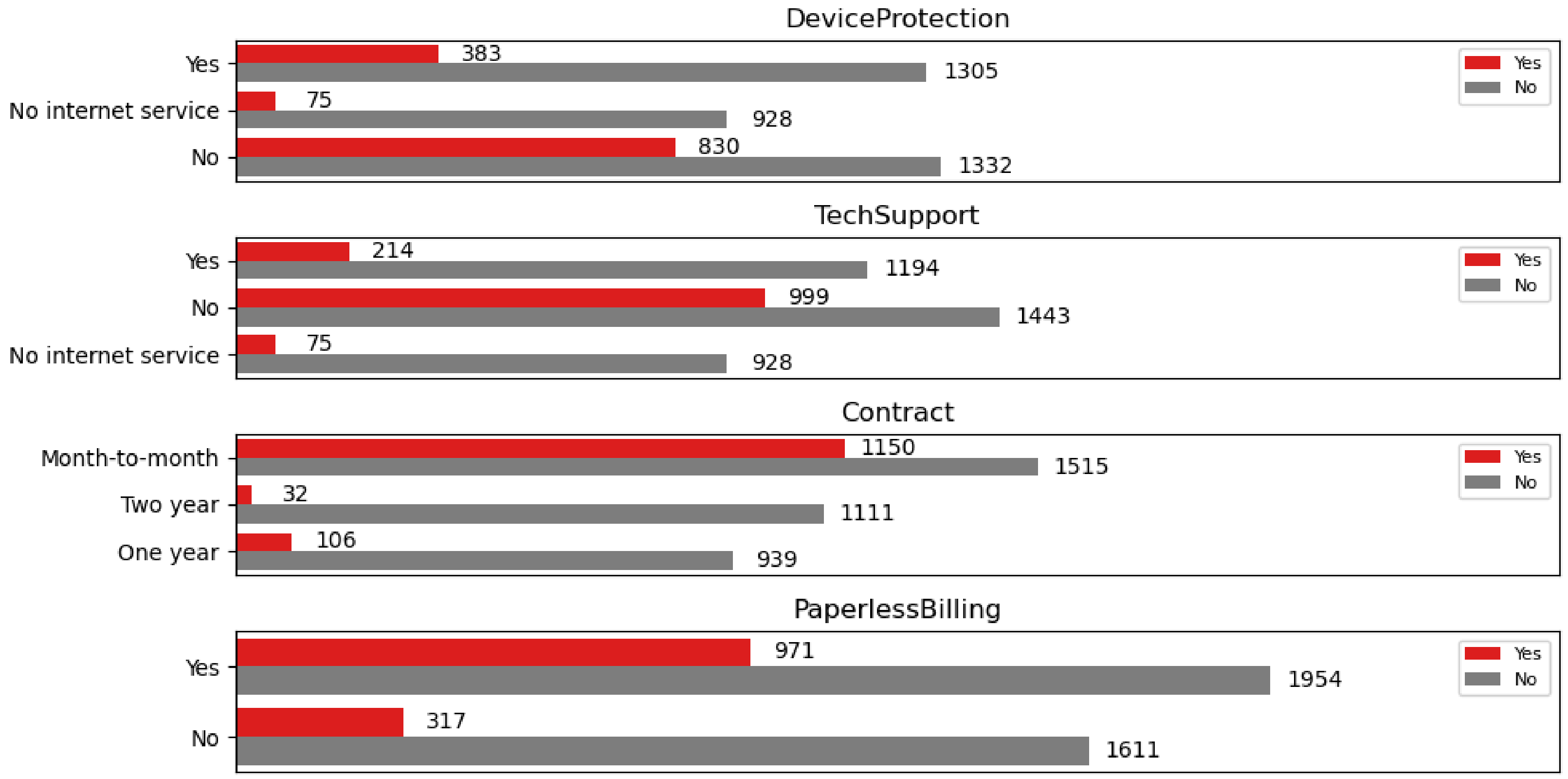
Churn
No : 73%
Yes: 27%



EXPLORATORY DATA ANALYSIS

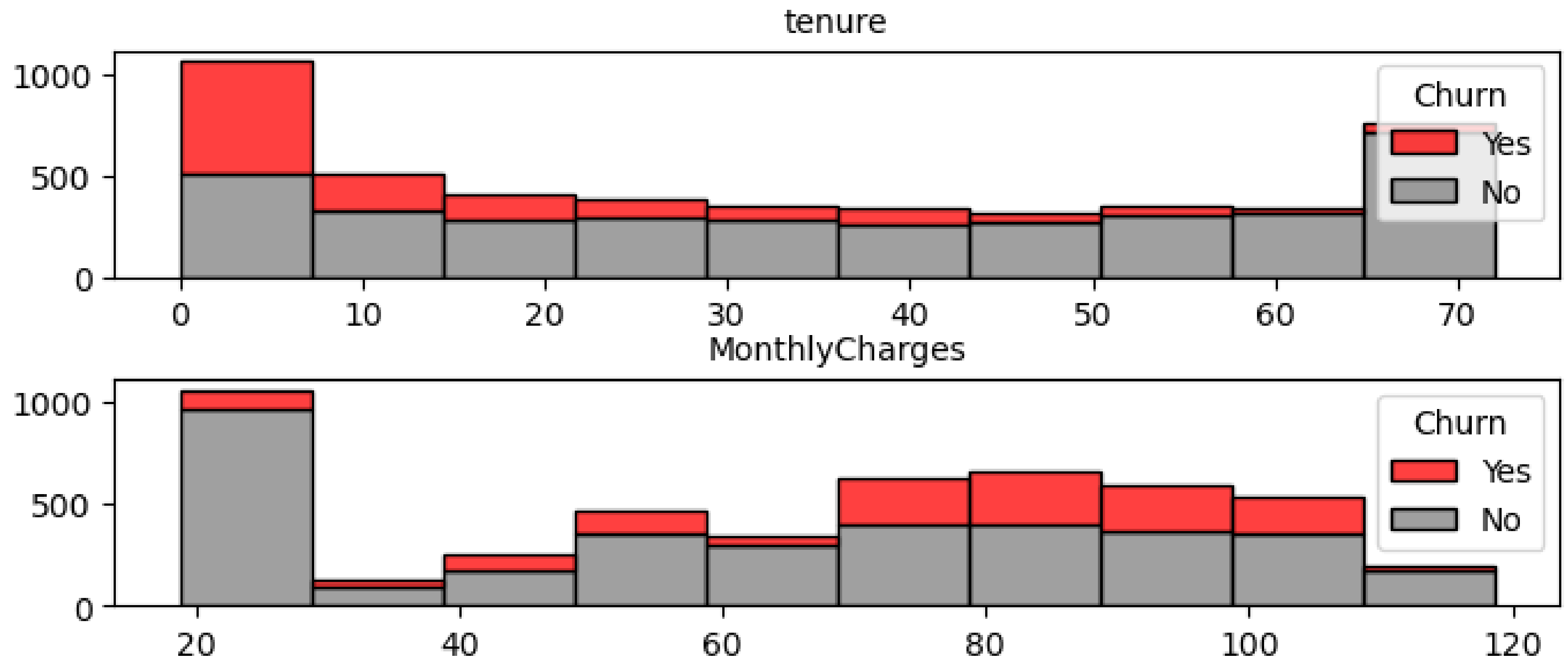
Qualitative feature
vs
target variable

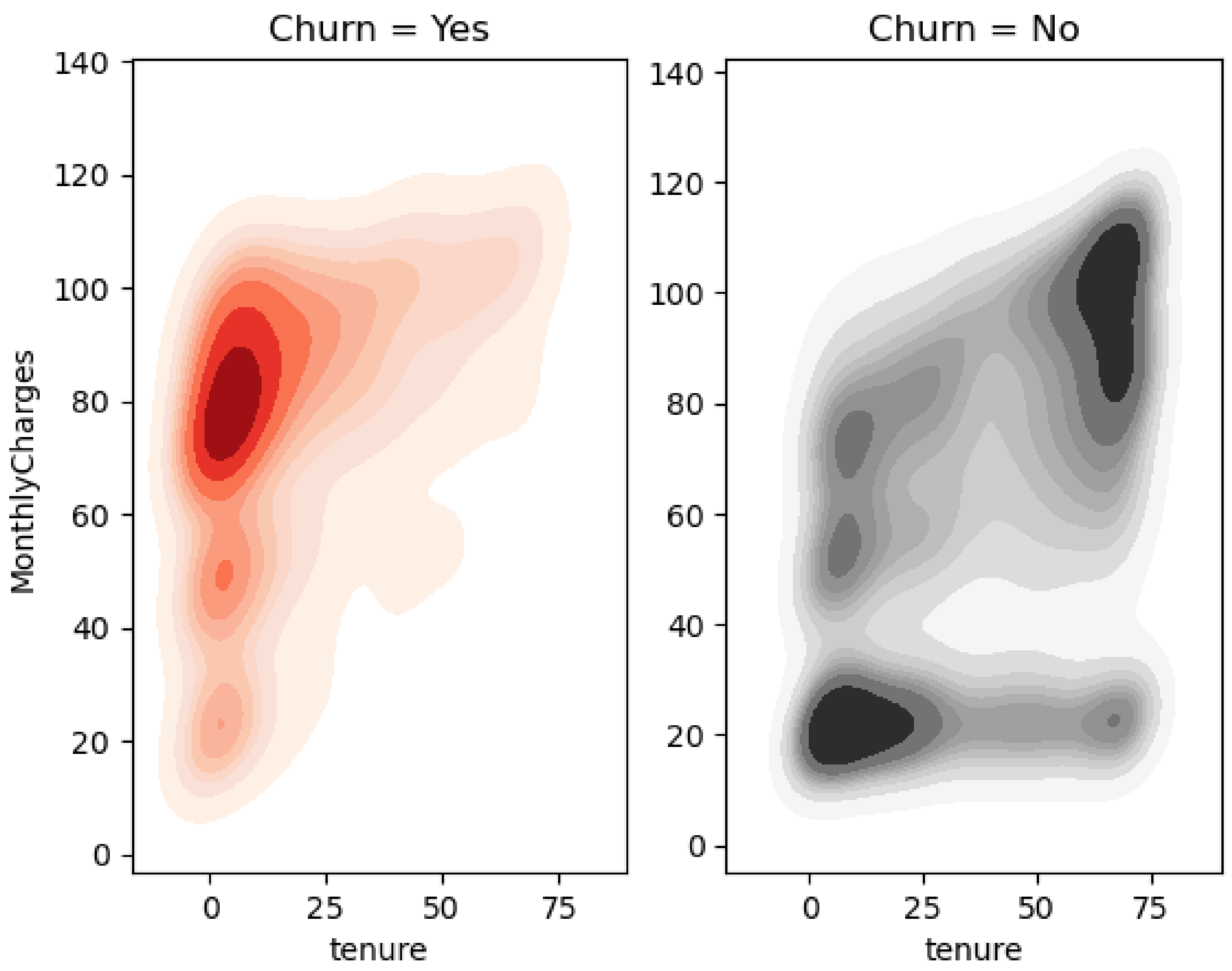




EXPLORATORY DATA ANALYSIS

Quantitative feature
vs
target variable





DATA PREPROCESSING

- Encoding
- Feature Selection
- Train-Test Split
- Resampling
- Scaling

DATA PREPROCESSING

Preprocessing	Assessment	Action
Encoding	All categorical feature contain 2/3 unranked category	All categorical features encoded using One-Hot Encoder
Feature Selection	Univariate Statistic • Categorical : Chi2-test • Numerical : F-test	Drop variable DeviceProtection_Yes which have a very low chi2 stat. Drop other NoInternetService column. Leaving just one out of 4.
Train-Test Split	Imbalance dataset with almost 5k rows	Splitting with portion of 80:20 and stratified
Resampling	Imbalance dataset	Random Under Sampler
Scaling	MinMax Scaler	Standard scaling method as we have only 2 numerical variable without outliers

MODELING

Algorithm Selection

Hyperparameter Tuning

Feature Importance

Model Interpretation

ALGORITHM SELECTION

Top performance model:

1. Stochastic gradient descent
2. Naive bayes gaussian
3. Adaptive boost

	model	train score mean	train score std	test score	score diff
1	sgd	0.7742	0.0388	0.7632	0.0109
6	nb_gaus	0.8078	0.0157	0.7532	0.0546
10	boost_ada	0.7911	0.0125	0.7402	0.0508
5	svc_lin	0.7865	0.0221	0.7382	0.0484
0	logreg	0.7845	0.0248	0.7338	0.0507
15	soft_voting	0.7995	0.0149	0.7213	0.0783
16	stacking	0.7784	0.0145	0.7163	0.0621
13	neural	0.7824	0.0204	0.7113	0.0712
4	gauss	0.7709	0.0184	0.7014	0.0695
7	boost_g	0.7715	0.0237	0.7011	0.0703
14	hard_voting	0.7532	0.0154	0.6851	0.0681
8	boost_hg	0.7558	0.0218	0.6794	0.0764
9	boost_xg	0.7528	0.0114	0.6595	0.0933
2	knn	0.7374	0.0256	0.6447	0.0927
11	rforest	0.7184	0.0193	0.6440	0.0744
12	xtree	0.7110	0.0116	0.5997	0.1113
3	dtree	0.6877	0.0273	0.5544	0.1334

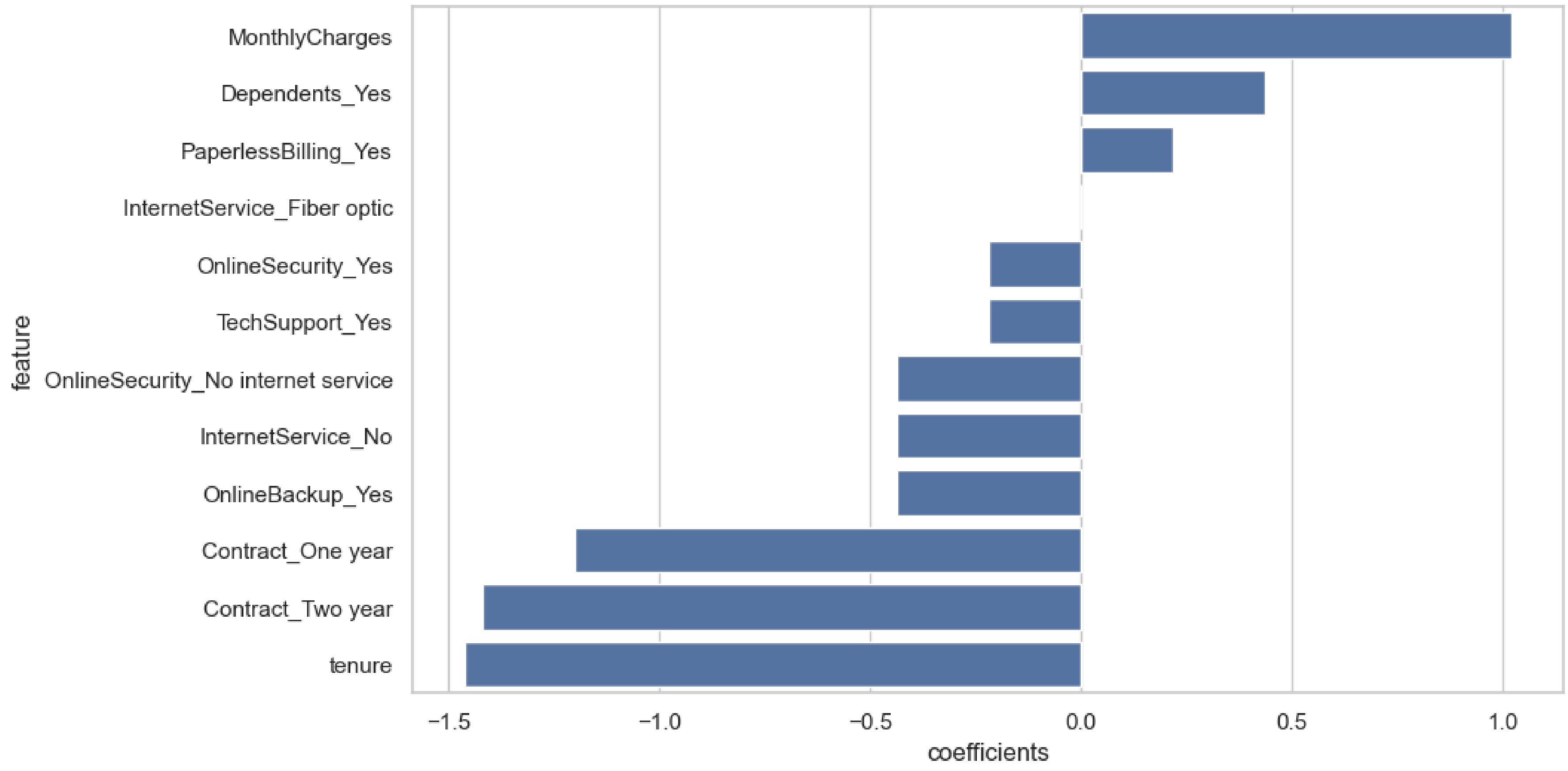
HYPERPARAMETER TUNING

Model	Before Tuning	After Tuning
Stochastic gradient descent	F2-score Train: 0.77 Test : 0.76	F2-score Train: 0.84 Test : 0.76
Naive bayes gaussian	F2-score Train: 0.81 Test : 0.75	F2-score Train: 0.84 Test : 0.74
Adaptive boost	F2-score Train: 0.79 Test : 0.74	F2-score Train: 0.87 Test : 0.70

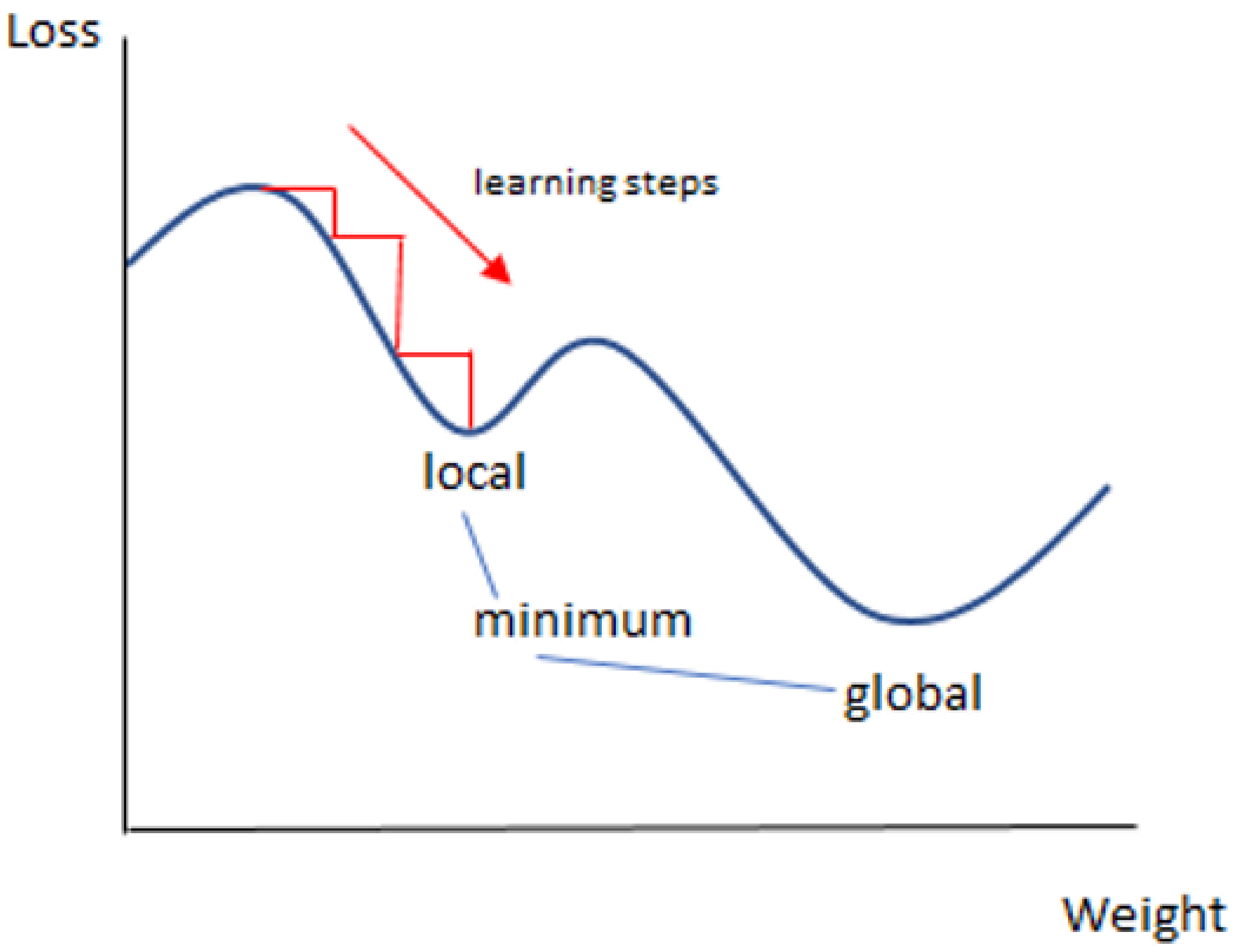
FEATURE IMPORTANCE (SGD)

Coefficients (Weights): These represent the importance and direction of each feature.

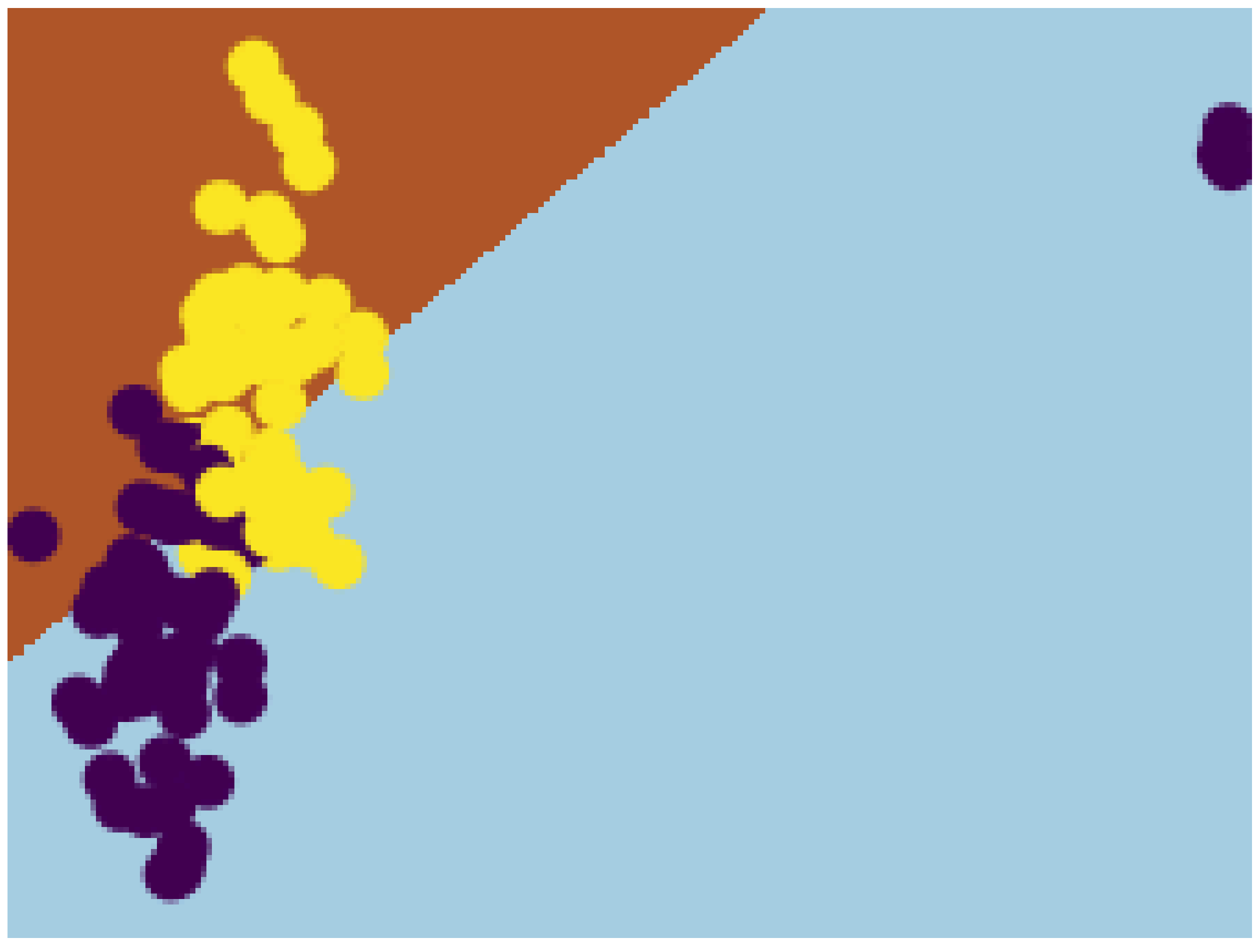
- Positive Coefficient: Indicates that as the feature increases, the probability of the positive class increases.
- Negative Coefficient: Indicates that as the feature increases, the probability of the positive class decreases.
- Magnitude: Larger absolute values indicate more influential features.



MODEL INTERPRETATION (SGD)



SGDClassifier, Hinge loss

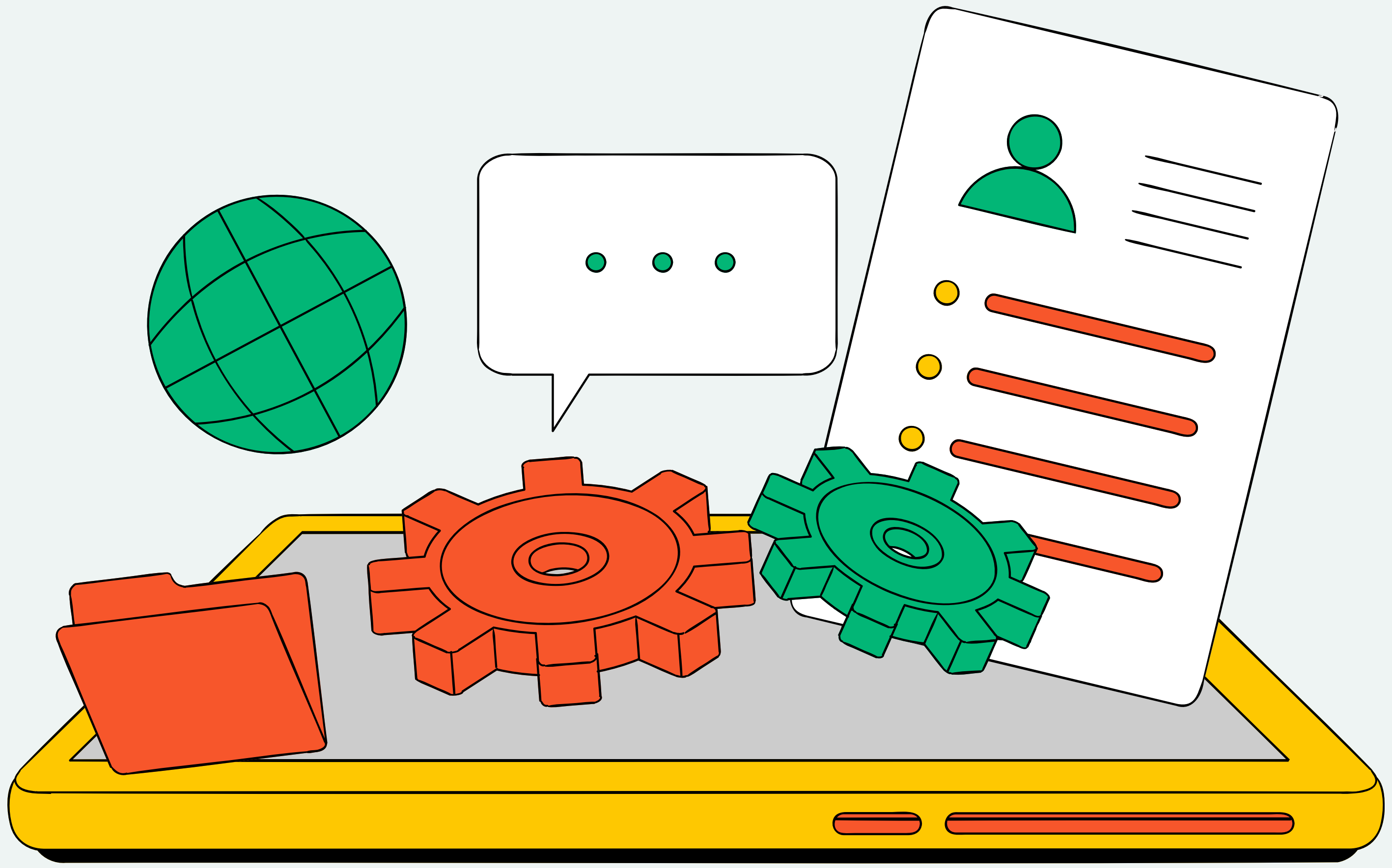


FINAL RESULT

Calculating the Cost Efficiency

Conclussion

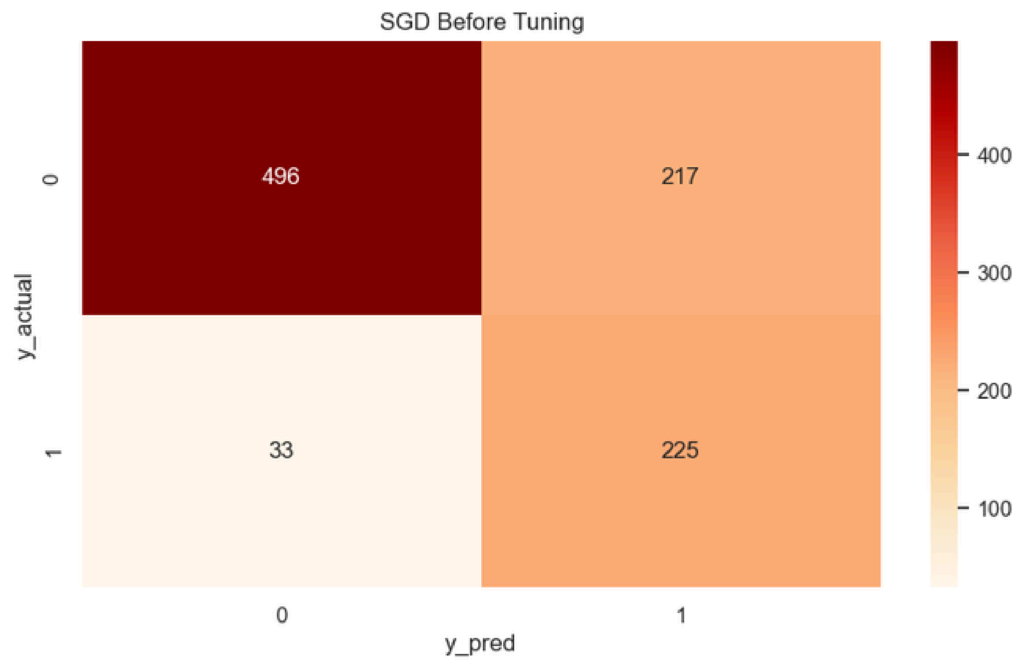
Recommendation



CALCULATING THE COST EFFICIENCY

Before implementing ML:
Cost_before = Median(MonthlyCharges)/5 * n_customer

After implementing ML:
Cost_after = Median(MonthlyCharges)/5 * FP+TP
Loss = Median(MonthlyCharges) * FN



Cost before implementing ML: $13711

Cost after implementing ML : $6241
Revenue loss : $2330

CONCLUSION

1. By implementing the ML prediction model. The company can cut half of their cost. From $13,711 to $6,241. With exchange of $2,330 revenue loss.
2. Customers who doesn't take add-on services (online security, online backup, device protection, tech support) are most likely to churn.
3. Customers who takes month-to-month contract are most likely to churn.
4. Customers whose tenure is low (below 10 month) and monthly charges is high (above 70 month) are most likely to churn.
5. Customers whose tenure is low (below 25 month) and monthly charge is low (below 30) are most likely to not churn. And customers whose tenure is high (above 50 month) and monthly charge is high (above 80) are most likely to not churn.

RECOMMENDATION

1. Recommend the add-on services to the customers.
2. Try to convince the customers to take on One or Two year contract rather than the month-to-month contract.
3. Give a special treatment for new customers (tenure below 1 year). Such as discount or maybe just as simple as good customer services.