VERIZON #2

AI STUDIO PROJECT

CUSTOMER CHURN

VERIZON #2

INTRODUCTION

MEET
THE TEAM

Arzy Abliadzhyieva
Wellesley College

Isabella Rasku-Casas
Umass Boston

Sheza Chaudhry
Tufts University

Yi
Tufts University

Xiomary
Salem State

OUR **LEADERS**

Our AI Studio TA, Katie Chen & Our Challenge Advisor, Nathan Jones were thoughtful, helpful, and proactive leaders during our project **

PRESENTATION **AGENDA**

- Project Overview 1
- Data Understanding and Prep 2
- Feature Engineering 3
- Modeling & Evaluation 4
- Customer Churn Analysis 5
- Conclusion 6
- Questions 7

OVERVIEW

- Determine which features lead to customer churn
- Accurately predict which customers are likely to churn
- Determine cost effective strategies to increase customer retention

BUSINESS IMPACT

- Customer churn is vital for telecom companies like Verizon for optimizing costs.
- Insights into churn drivers enable a targeted approach to retain customers and boost competitiveness.
- The project aims to enhance resource allocation efficiency, minimizing unnecessary retention expenses.

VERIZON #2

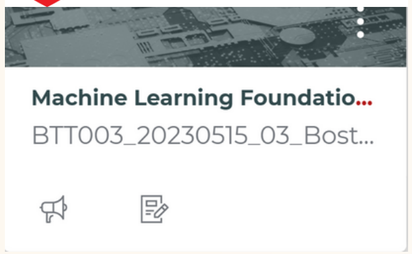
DATA UNDERSTANDING & DATA PREPERATION

RESOURCES
WE LEVERAGED

NumPy

SciPy

eCornell



scikit learn

pandas

matplotlib

seaborn

PEAK AT **DATA**

Initial Observations:

- A lot of features
- Some features are binary but categorical
- Others are numeric (total charges and monthly charges)
- Some features had only three unique values which could be grouped together

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService
0	7590-VHVEG	Female	0	Yes	No	1	No
1	5575-GNVDE	Male	0	No	No	34	Yes
2	3668-QPYBK	Male	0	No	No	2	Yes
3	7795-CFOCW	Male	0	No	No	45	No
4	9237-HQITU	Female	0	No	No	2	Yes
5	9305-CDSKC	Female	0	No	No	8	Yes

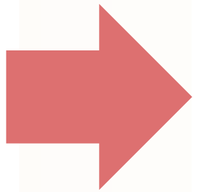
MultipleLines	InternetService	OnlineSecurity	OnlineBackup	DeviceProtection	TechSupport	StreamingTV
No phone service	DSL	No	Yes	No	No	No
No	DSL	Yes	No	Yes	No	No
No	DSL	Yes	Yes	No	No	No
No phone service	DSL	Yes	No	Yes	Yes	No
No	Fiber optic	No	No	No	No	No
Yes	Fiber optic	No	No	Yes	No	Yes

StreamingMovies	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges	Churn
No	Month-to-month	Yes	Electronic check	29.85	29.85	No
No	One year	No	Mailed check	56.95	1889.5	No
No	Month-to-month	Yes	Mailed check	53.85	108.15	Yes
No	One year	No	Bank transfer (automatic)	42.30	1840.75	No
No	Month-to-month	Yes	Electronic check	70.70	151.65	Yes
Yes	Month-to-month	Yes	Electronic check	99.65	820.5	Yes

MISSING **VALUES**

- Found rows containing missing values and removed them.
- Minimal number of missing values so removing them was best for our modeling stage.
- Ended up removing 22 rows out of ~7,000

customerID	0
gender	0
SeniorCitizen	0
Partner	0
Dependents	0
tenure	0
PhoneService	1
MultipleLines	3
InternetService	1
OnlineSecurity	2
OnlineBackup	0
DeviceProtection	2
TechSupport	2
StreamingTV	3
StreamingMovies	0
Contract	2
PaperlessBilling	1
PaymentMethod	2
MonthlyCharges	1
TotalCharges	1
Churn	1
dtype: int64	



customerID	0
gender	0
SeniorCitizen	0
Partner	0
Dependents	0
tenure	0
PhoneService	0
MultipleLines	0
InternetService	0
OnlineSecurity	0
OnlineBackup	0
DeviceProtection	0
TechSupport	0
StreamingTV	0
StreamingMovies	0
Contract	0
PaperlessBilling	0
PaymentMethod	0
MonthlyCharges	0
TotalCharges	0
Churn	0
dtype: int64	

DATA TYPE **CONVERSIONS**

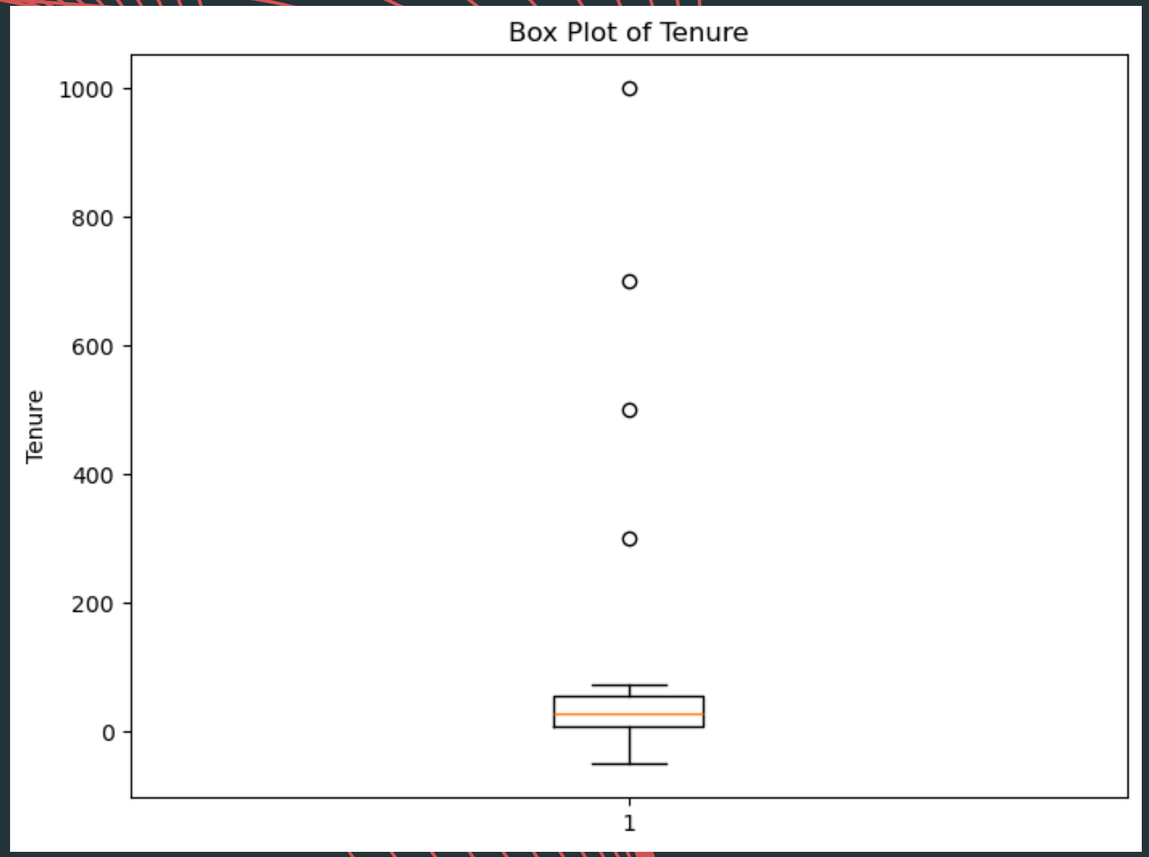
- We converted features with object dtypes to Integers: 1 (yes) and 0 (no)
- Then we took those 1's and 0's, and converted them to booleans for our modeling stage.

customerID	object
gender	object
SeniorCitizen	int64
Partner	object
Dependents	object
tenure	int64
PhoneService	object
MultipleLines	object
InternetService	object
OnlineSecurity	object
OnlineBackup	object
DeviceProtection	object
TechSupport	object
StreamingTV	object
StreamingMovies	object
Contract	object
PaperlessBilling	object
PaymentMethod	object
MonthlyCharges	float64
TotalCharges	object
Churn	object

gender	bool
SeniorCitizen	bool
Partner	bool
Dependents	bool
scaledtenure	float64
PhoneService	bool
MultipleLines	bool
InternetService	bool
OnlineSecurity	bool
OnlineBackup	bool
DeviceProtection	bool
TechSupport	bool
StreamingTV	bool
StreamingMovies	bool
Contract_Month-to-month	bool
Contract_One year	bool
Contract_Two year	bool
PaperlessBilling	bool
AutomaticPayment	bool
winsorized_MonthlyCharges	float64
scaledTotalCharges	float64
Churn	bool

OUTLIERS

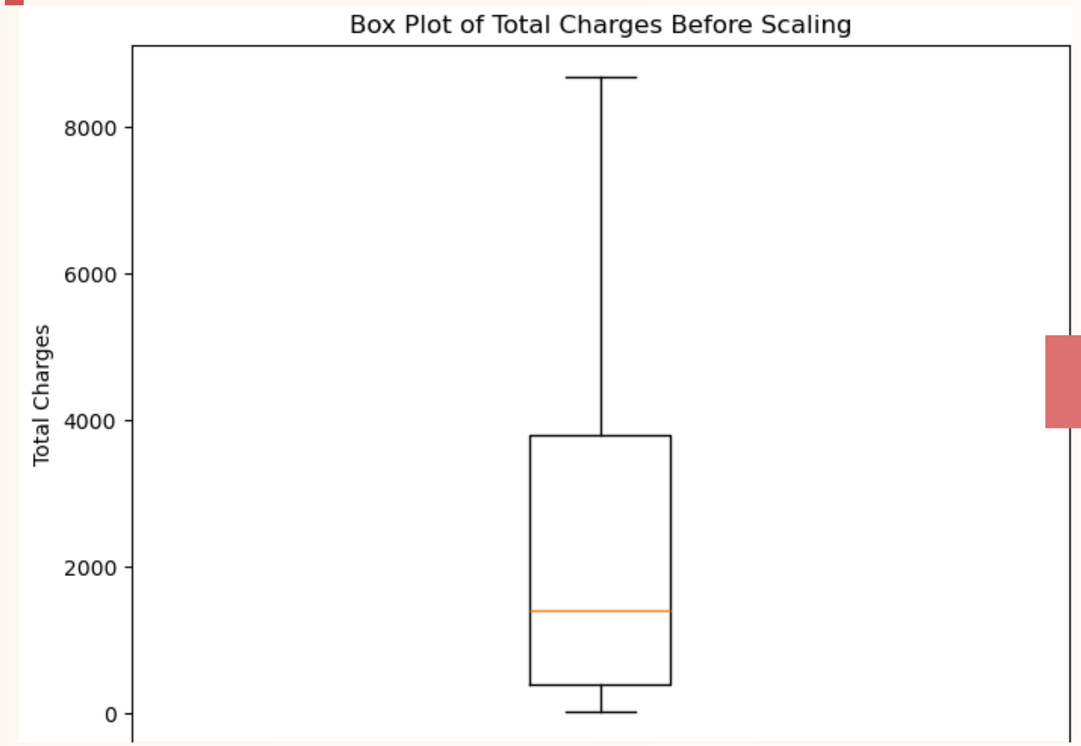
- Used the .describe to see if there were any clear features with outliers.
- Tenure had some obvious outliers so we removed them accordingly.



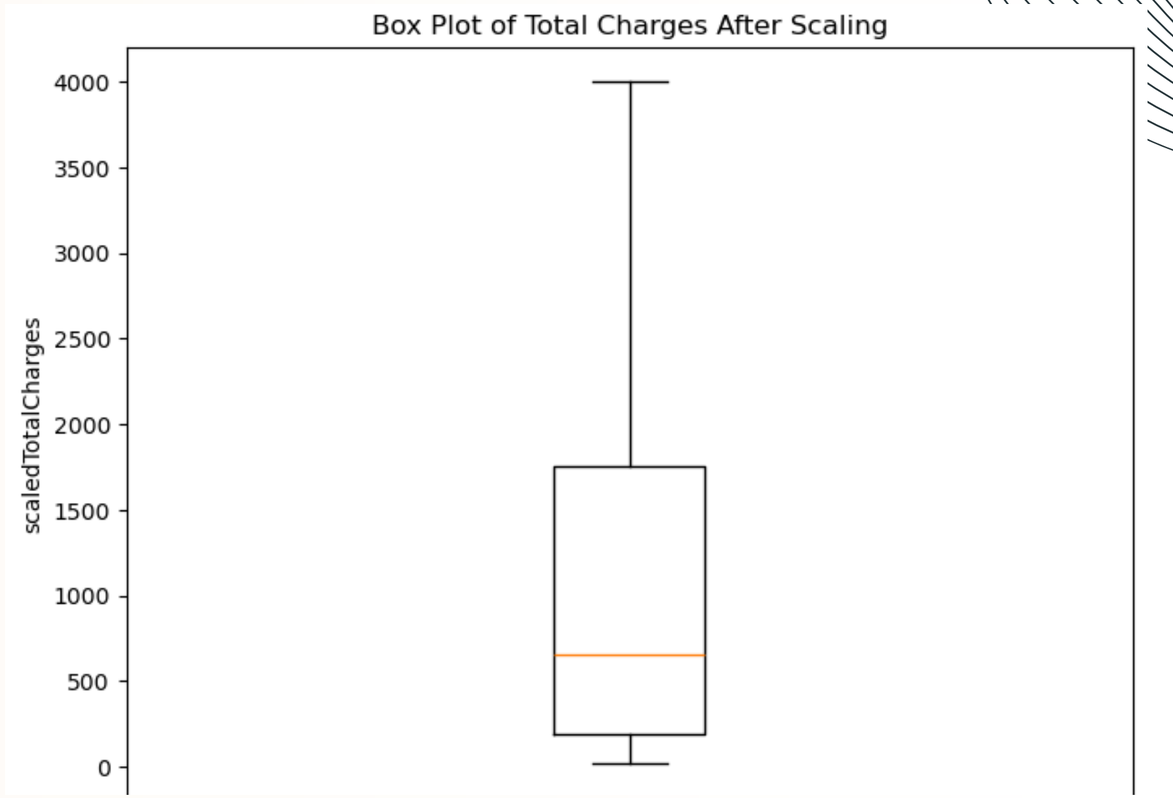
	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection	TechSupport	StreamingTV	StreamingMovies	Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges	Churn
count	7043	7043	7043.000000	7043	7043	7043.000000	7042	7040	7042	7041	...	7041	7041	7040	7043	7041	7042	7041	7042.000000	7042	7042
unique	7043	2	NaN	2	2	NaN	2	3	3	3	...	3	3	3	3	4	2	4	NaN	6530	2
top	7590-VHVEG	Male	NaN	No	No	NaN	Yes	No	Fiber optic	No	...	No	No	No	No	Month-to-month	Yes	Electronic check	NaN		No
freq	1	3555	NaN	3641	4933	NaN	6360	3388	3095	3498	...	3094	3472	2808	2785	3870	4170	2364	NaN	11	5173
mean	NaN	NaN	0.162147	NaN	NaN	32.683516	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	64.756774	NaN	NaN
std	NaN	NaN	0.368612	NaN	NaN	28.998589	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	30.089352	NaN	NaN
min	NaN	NaN	0.000000	NaN	NaN	-50.000000	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	18.250000	NaN	NaN
25%	NaN	NaN	0.000000	NaN	NaN	9.000000	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	35.500000	NaN	NaN
50%	NaN	NaN	0.000000	NaN	NaN	29.000000	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	70.350000	NaN	NaN
75%	NaN	NaN	0.000000	NaN	NaN	55.000000	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	89.850000	NaN	NaN
max	NaN	NaN	1.000000	NaN	NaN	1000.000000	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	118.750000	NaN	NaN

11 rows × 21 columns

SCALING
TOTAL CHARGES



Maximum: 8684
Minimum: 18
Range: 8666
Average: 2283.6
Standard Deviation: 2265



Maximum: 4000
Minimum: 18
Range: 3982
Average: 1059
Standard Deviation: 1041

VERIZON #2

FEATURE ENGINEERING

	Partner	Dependents	FamilyTies
0	True	False	1
1	False	False	0
2	False	False	0
3	False	False	0
4	False	False	0
5	False	False	0
6	False	True	1
7	False	False	0
8	True	False	1
9	False	True	1
10	True	True	1
11	False	False	0
12	True	False	1
13	False	False	0
14	False	False	0
15	True	True	1
16	False	False	0

FAMILY TIES

Combining ‘Partner’ and ‘Dependants’ into single feature that indicates if a customer has family ties.

We wanted to see If families or accounts with dependants had a correlation to churn.

CONTRACT **TYPE**

Transforming the Contract feature with 3 values (‘Month-to-month’, ‘One year,’ and ‘Two years’) into a binary ContractType, which answers the question of whether the customer is locked into a contract or pays monthly.

```
# Replacing the contract names with 0, 1, and 2
df_cleansed['ContractType'].replace({
    'Contract_Month-to-month': 0,
    'Contract_One year': 1,
    'Contract_Two year': 1
}, inplace=True)

print(df_cleansed[['ContractType']])
```

Contract
Month-to-month
One year
Month-to-month
One year
Month-to-month

ContractType
True
False
True
False
True
...
False
True

YES/NO **MAPPING**

Transform features with 3 unique values but little information into binary features by grouping ‘No’ and ‘No phone service’ together, as well as not differentiating between internet type.

```
# Define mapping dictionaries
yes_no_mapping = {'No phone service': 'No', 'No internet service': 'No', 'DSL': 'Yes', 'Fiber optic': 'Yes'}

# Columns to replace
columns_to_replace = ['MultipleLines', 'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
                      'TechSupport', 'StreamingTV', 'StreamingMovies', 'InternetService', 'PaymentMethod']


# Apply mapping to the specified columns
df[columns_to_replace] = df[columns_to_replace].replace(yes_no_mapping)
```

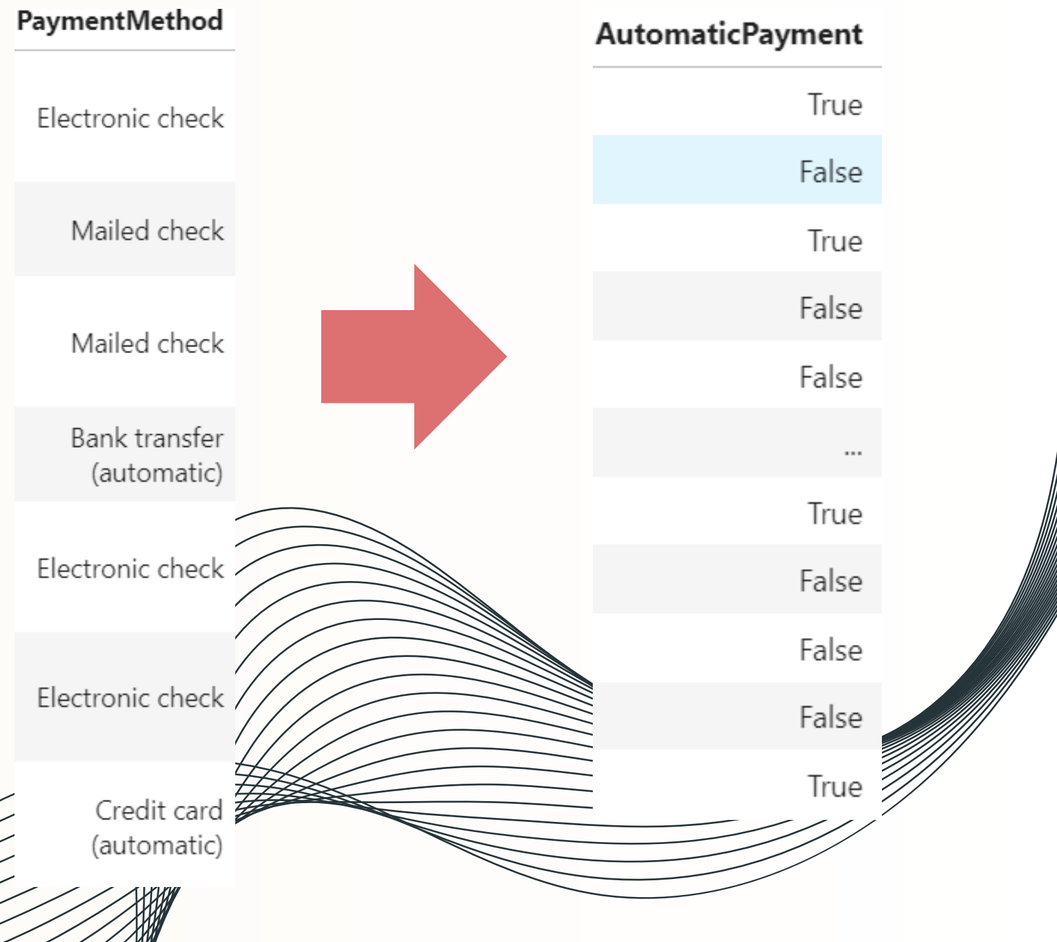
MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection
No phone service	DSL	No	...	No
No	DSL	Yes	...	Yes
No	DSL	Yes	...	No
No phone service	DSL	Yes	...	Yes
No	Fiber optic	No	...	No
Yes	Fiber optic	No	...	Yes
Yes	Fiber optic	No	...	No

MultipleLines	StreamingTV	StreamingMovies
False	True	False
True	True	False
True	True	True
True	False	False
True	False	False
...	...	...
False	False	False
False	False	False
True	False	False
True	True	True
True	False	True

PAYMENT **METHOD**

```
automatic_payment_mapping = {'Bank transfer (automatic)': 'Yes', 'Credit card (automatic)': 'Yes',
                             'Electronic check': 'No', 'Mailed check': 'No'}
df['AutomaticPayment'] = df['PaymentMethod'].replace(automatic_payment_mapping)

# Drop the original 'PaymentMethod' column
df.drop('PaymentMethod', axis=1, inplace=True)
```



Transforming the PaymentMethod feature with 4 unique values into a binary AutomaticPayment feature.

This feature was the opposite of the paperless billing feature. Having this contradiction helped us during feature Importance.

VERIZON #2

MODELING & EVALUATION

EVALUATION METRICS

- ***Accuracy***

  Measures the overall correctness, including both churn and non-churn predictions

$$\frac{\text{True Pos} + \text{True Neg}}{\text{Total Instances}}$$

- ***Precision***

  Measures how many customers actually churned among the instances predicted as churn by the model.

$$\frac{\text{True Pos}}{\text{True Pos} + \text{False Pos}}$$

- ***Recall***

  Measures how many customers the model identified as churn among the ones that actually churned.

$$\frac{\text{True Pos}}{\text{True Pos} + \text{False Neg}}$$

- ***F1-Score***

  Combines precision and recall. Useful when there is an imbalance between the classes, as is often the case in churn prediction.

$$\frac{2 * (\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}}$$

- ***ROC-AUC***

  Evaluates the ability of a model to tell between churn and non-churn instances across different probability thresholds.

LOGISTIC REGRESSION

Pros:

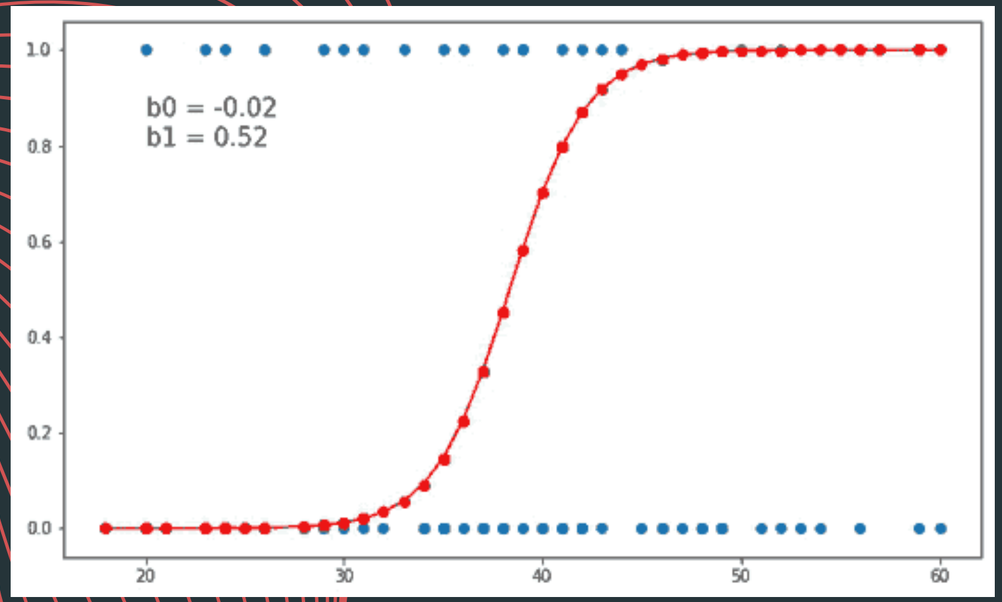
- Interpretable results (feature weights tell us how individual features contribute to the prediction)
- Designed for binary classification problems, such as the one we have

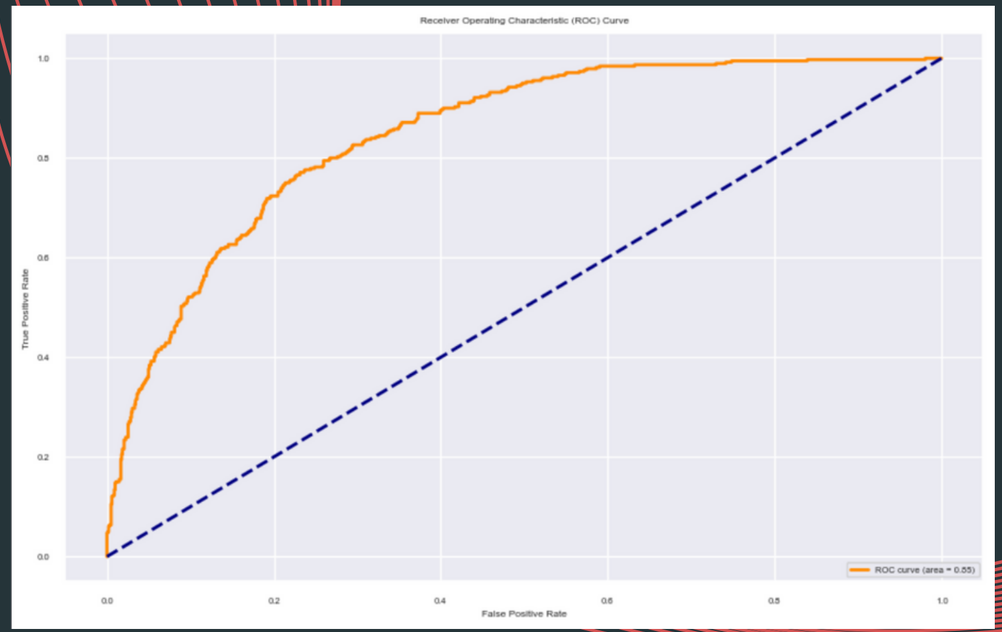
Cons

- Bad at handling categorical data
- Doesn’t capture feature interaction

Results

- Accuracy: 80%
- Precision: 63%
- Recall: 56%
- F1: 60%
- ROC AUC: 0.84





RANDOM FOREST

Pros:

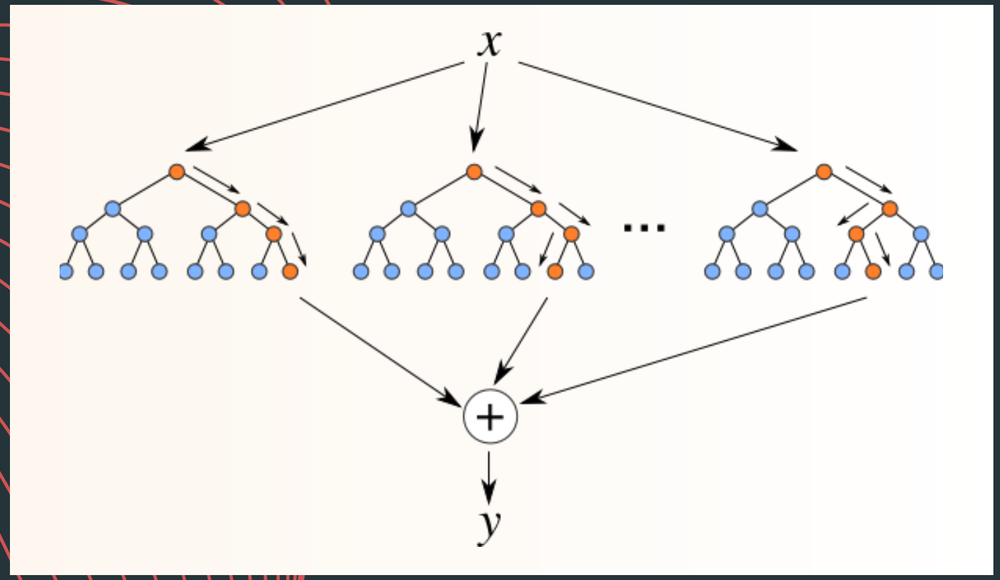
- Well at handling non-linearity and feature interaction
- Suitable for data that has both categorical and numeric features

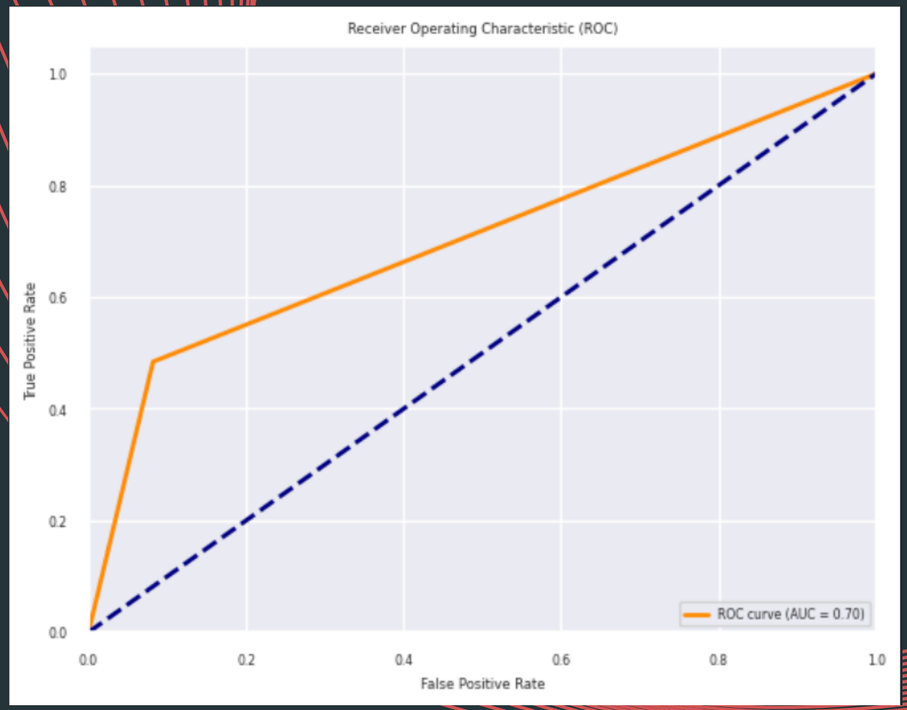
Cons

- Could overfit unless hyperparameters are well-tuned
- Has high computational costs

Results

- Accuracy: 80%
- Precision: 65%
- Recall: 53%
- F1: 58%
- ROC AUC: 0.71





ADABOOST

Pros:

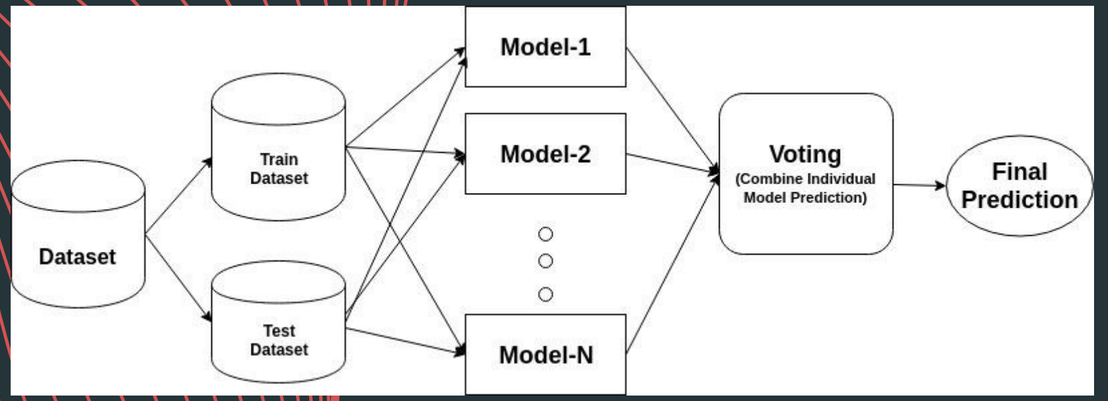
- Good for datasets with many features as it implicitly performs feature selection by assigning weights to features based on their importance.
- Combines multiple weak learners to create a strong model while avoiding overfitting

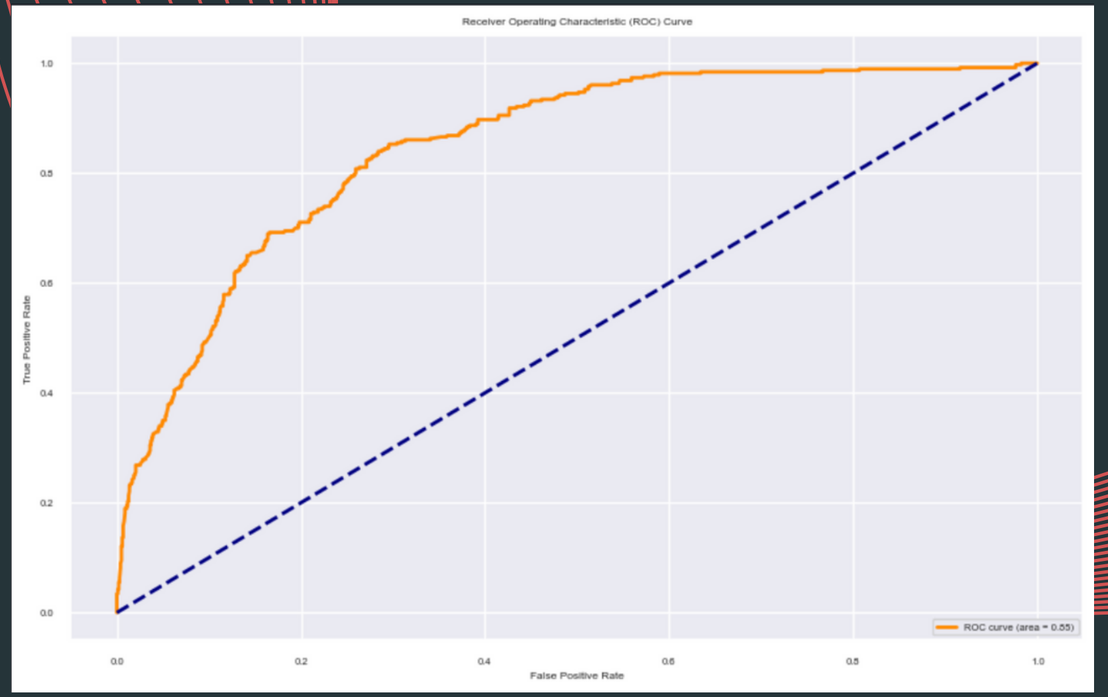
Cons

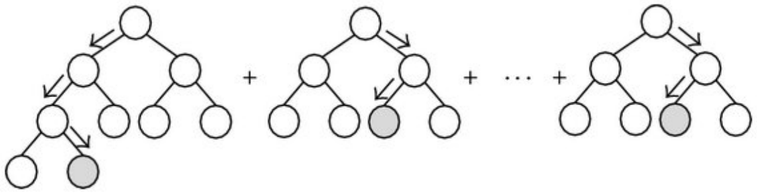
- Sensitive to noise: outliers in the early stages of boosting can have a lingering impact.
- Not as good at capturing complex relationships between features

Results

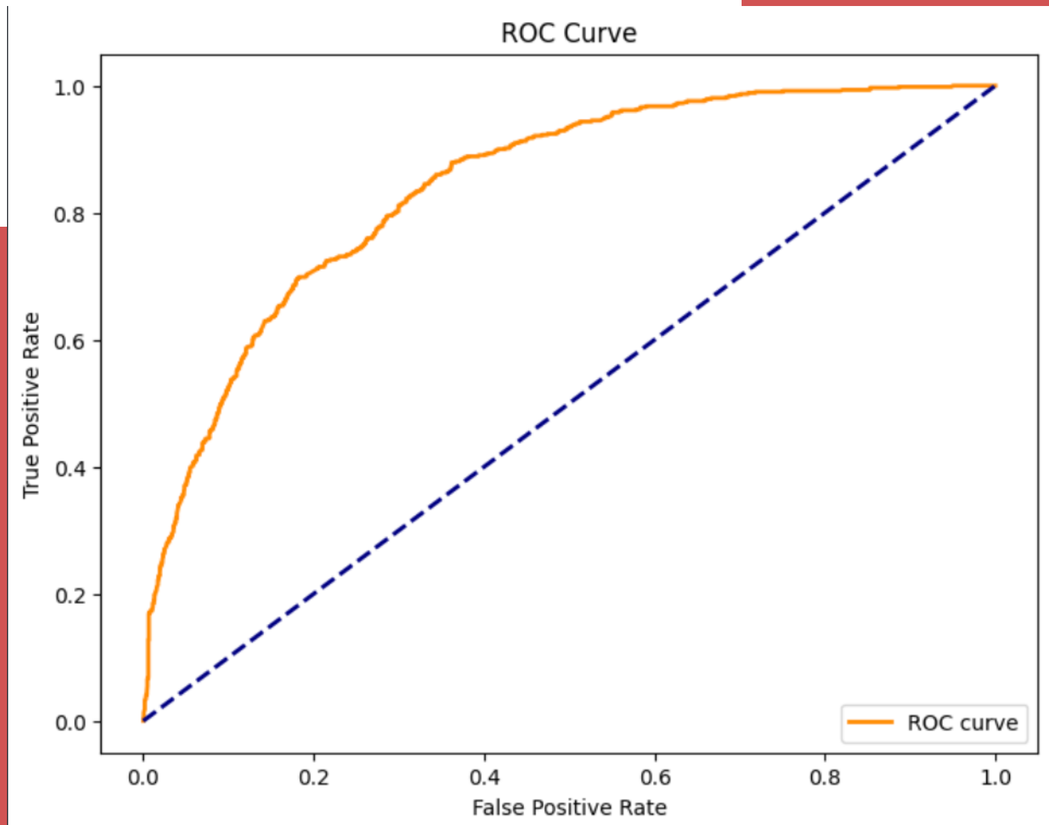
- Accuracy: 80%
- Precision: 63%
- Recall: 55%
- F1: 59%
- ROC AUC: 0.72







GRADIENT BOOSTING



Pros:

- Excels in capturing complex relationships and interactions in the data.
- Robust: can handle data with irregularities without being overly influenced by individual instances.
- Handles imbalanced datasets well by adjusting the weights of misclassified instances.

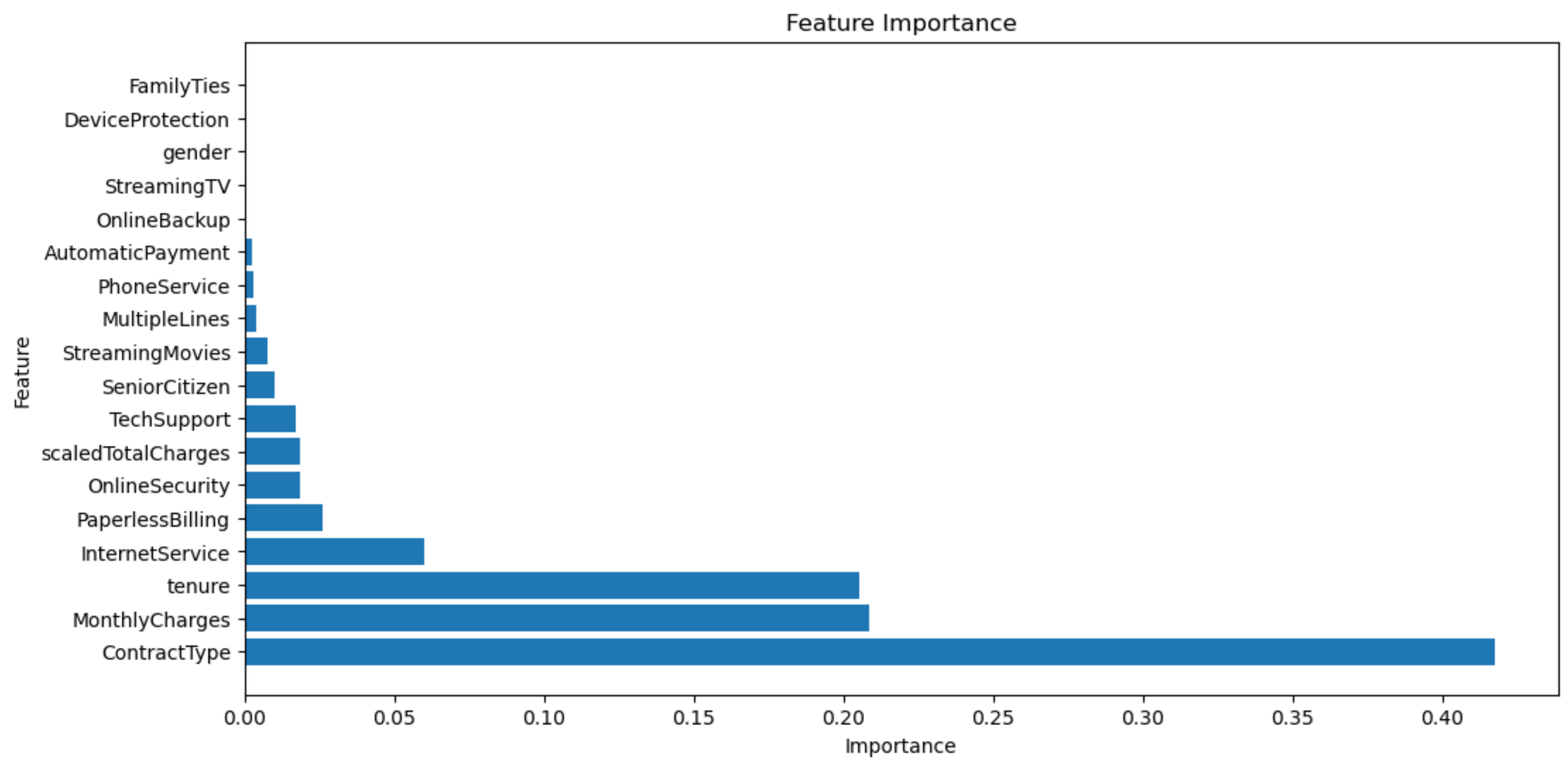
Cons

- Computationally expensive as it requires more hyperparameter tuning

Results

- Accuracy: 80%
- Precision: 65%
- Recall: 53%
- F1: 58%
- ROC-AUC Score: 0.85

FEATURE IMPORTANCE



VERIZON #2

CUSTOMER CHURN ANALYSIS

BUSINESS GOALS

Customers

Want to save customers who bring In at least $66/month

Optimizing $$

How much of revenue are we allocating to saving customers.

Investors

We need to keep Investors happy and customers appreciated.

ASSUMPTIONS

MONTHLY CHARGE

CONSISTENT OVER TIME

80%

SUCCESS RATE IN SAVING ID'D ACCOUNTS

$800

COST OF SAVING AN ACCOUNT AND LOCKING IN TO 2-YEAR CONTRACT

MAKING TRADEOFFS

Conservative Approach

Pros:

+ We have model that predicts churning customers at a high accuracy (+95%)

+ The money we spend Is likely to retain customers.

Cons:

-Missing savable customers

-Wall street prioritizes customers, Investors may leave.

Aggressive Approach

Pros:

+ We have model that predicts churning customers at a low accuracy (~40%)

+ Save 80% of the customers we Identify.

+ Good wall street review

Cons:

-Unsustainable financially

-Will end up saving customers that were not going to churn.

MAKING TRADEOFFS

Conservative Approach

e have model that predicts
g customers at a high accuracy
(+95%)
e money we spend Is likely to
retain customers.

Missing savable customers
street prioritizes customer
Investors may leave.

We need balance!

Our Approach

Reasonable confidence rate
60%-80%

Decide which customers to save based on their add ons and monthly revenue.

Aggressive Approach

+ We have model that predicts
customers at a low accuracy (
+ Save 80% of the customer
Identify.
+ Good wall street review

-Unsustainable financiall
end up saving customers th
not going to churn.

DATA-DRIVEN
APPROACH

DATA-DRIVEN **APPROACH**

Customer Threshold

Cost to save a customer: $800
(& locked Into two year contract)
In order to double our spending, customers should have a monthly charge of at least $66 & should have Internet & Phone Service.

DATA-DRIVEN APPROACH

Customer Threshold

Cost to save a customer: $800
(& locked Into two year contract)
In order to double our spending, customers should have a monthly charge of at least $66 & should have Internet & Phone Service.

Cost to Save

After setting our customer threshold we are saving 284 customers or approximately 60% of churning customers.

Total cost of saving

Customers to save * Cost to save = $227,200

DATA-DRIVEN APPROACH

Customer Threshold

Cost to save a customer: $800
(& locked Into two year contract)
In order to double our spending, customers should have a monthly charge of at least $66 & should have Internet & Phone Service.

Cost to Save

After setting our customer threshold we are saving 284 customers or approximately 60% of churning customers.

Total cost of saving

Customers to save * Cost to save =
$227,200

Cost Analysis

Revenue =
'monthly charges' + 'total charges'
We have a revenue of $7,903,845

Cost Analysis =
Total cost of saving / Revenue =
~2.8%

MODEL ANALYSIS

- Reasonably high accuracy, indicating good overall predictive performance.
- If we choose to give bonuses to a random customer, we will be accurate only 27% of the time.

OUR MODEL IS

80%

ACCURATE

VERIZON #2

FINAL THOUGHTS

WHAT WE **LEARNED**

- Apply classroom-learned data analysis skills to real-world telecom data.
- Learn handling complex, large-scale datasets.
- Understand statistical techniques and machine learning for predicting customer churn.
- Gain experience in feature selection, visualization, and effective communication of findings.
- Recognize the business implications of analysis and the importance of ethical data handling.

WHAT COULD **BE DONE BETTER?**

More insights from the marketing department can guide the choice of evaluation metrics.

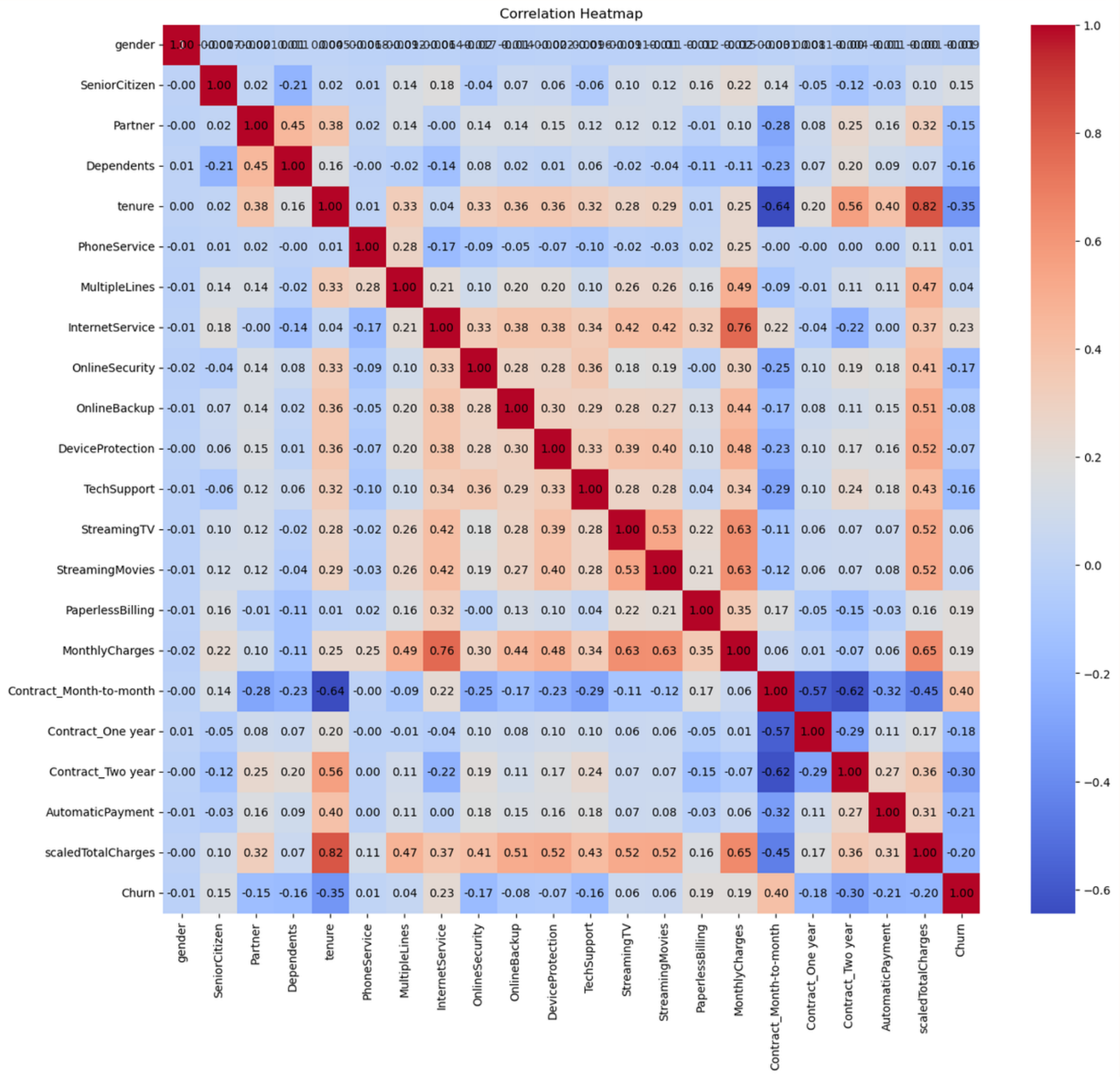
More data is always useful!

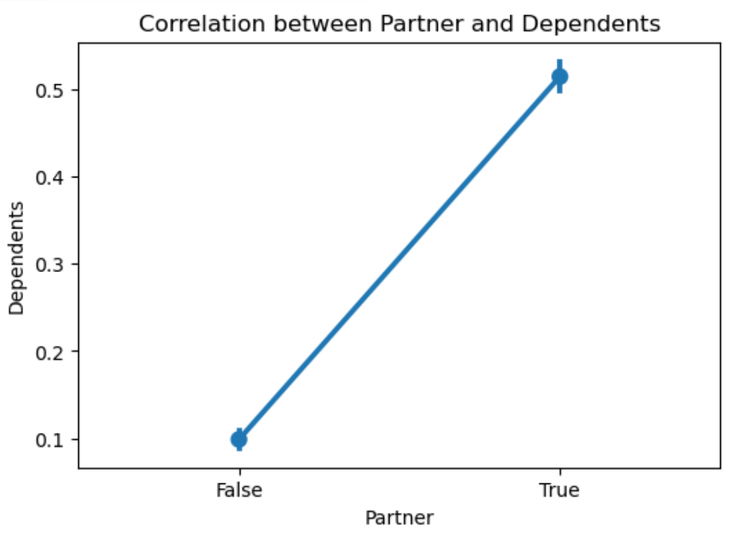
Potential information to collect: usage intensity, product satisfaction, etc.

VERIZON #2

QUESTIONS?

Before data engineering





After data engineering