Insurance Claim Status Prediction using ML Classification Models

Machine Learning 1, DS&BA, UW

Agenda

1. Project Objective
2. Exploratory Data Analysis (EDA)
3. Feature Selection
4. Feature Engineering
5. Models Consideration
6. Prediction

Project Objective

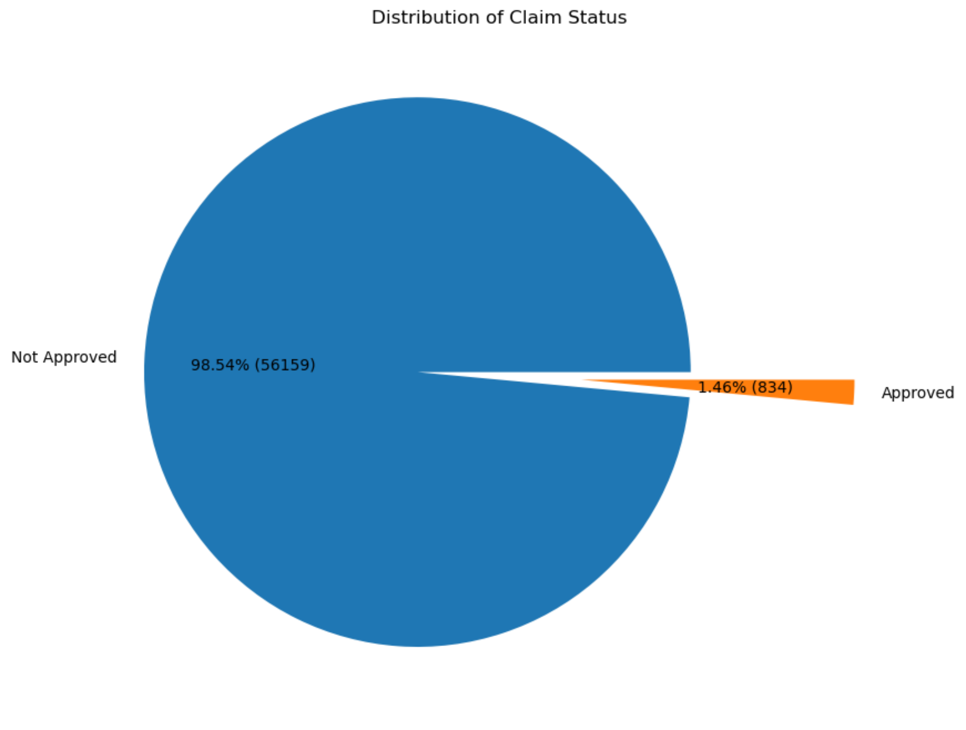
By training on the given sample, best algorithm should be found for predicting the travel insurance claim outcome (target variable `claim_status`: 1 – Approved / 0 – Not approved) on unseen data.

- Training Sample: 56993 observations including target variable
- Test Sample: 6333 observations without target variable
- 14 features in total including target variable

Exploratory Data Analysis

- No missing values and duplicates
- 3 numeric variables (`revenue`, `reward`, `customer_score`)
- 3 discrete variables (`trip_length`, `person_age`, `support_interactions`)
- 7 nominal variables (`person_gender`, `entity_type`, `channel`, `agent_id`, `entity_a`, `location`, `product_id`)
- 1 binary variable (`claim_status`)

The data is heavily imbalanced.

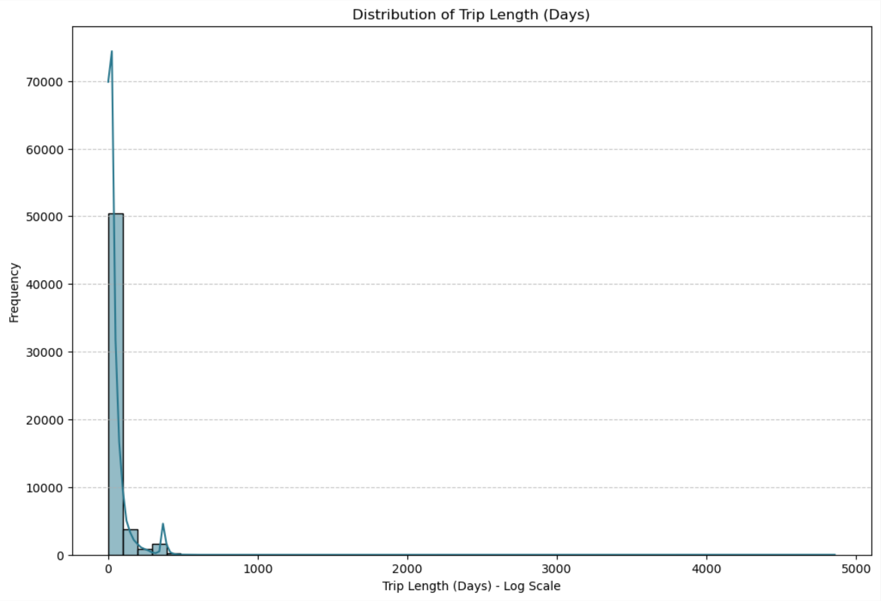


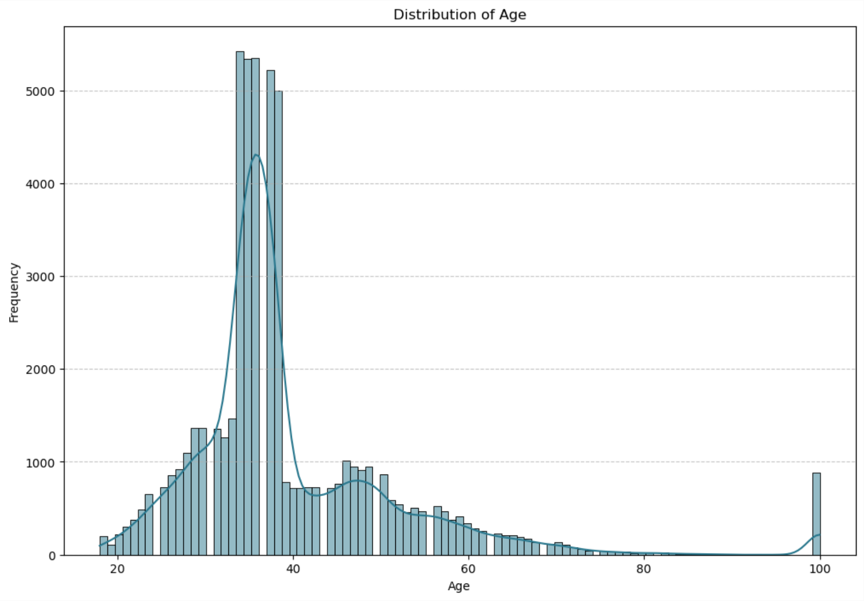
Exploratory Data Analysis

For trip length and person's age:

- Strongly right-skewed, especially for trip length
- Extreme outliers (e.g. max 4856 and min 1 for trip length).

Further binning and transformation have to be considered for modeling.



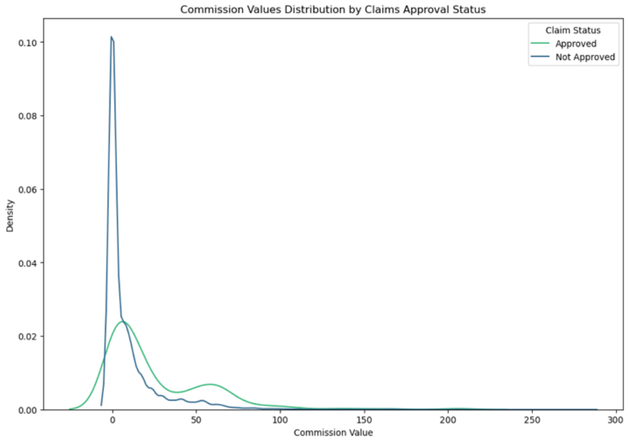


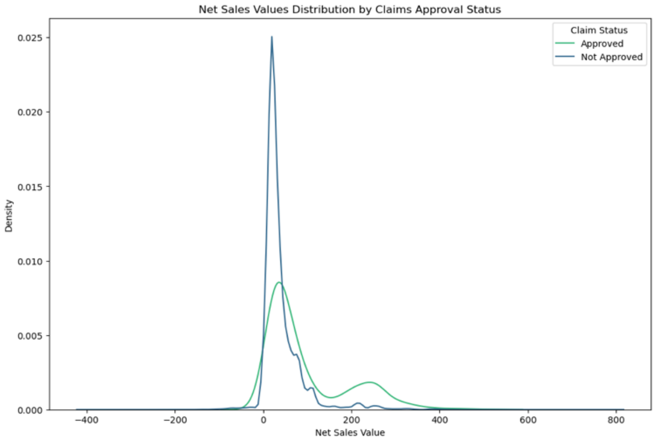
Exploratory Data Analysis

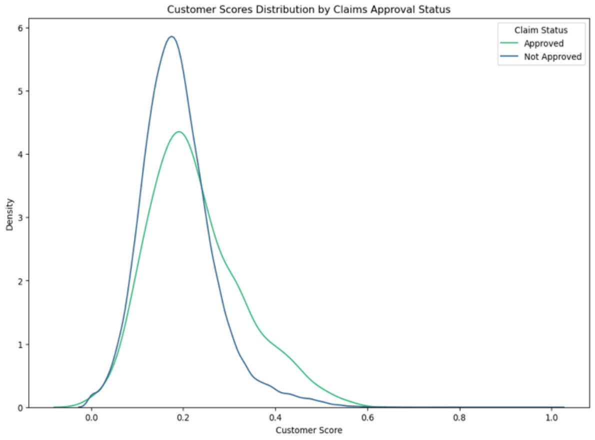
For commission values, net sales values and customer scores:

- Unevenly distributed referring to claim status

Further transformation have to be considered for modeling.

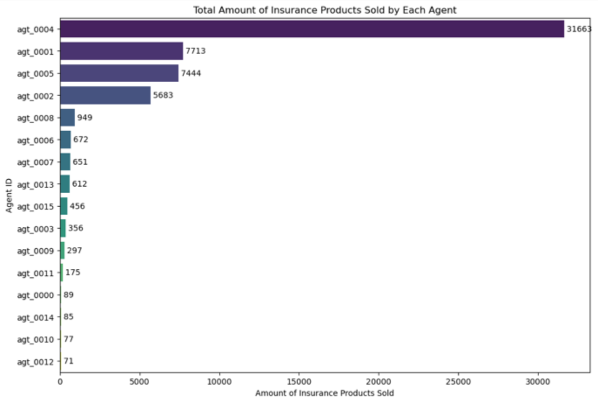


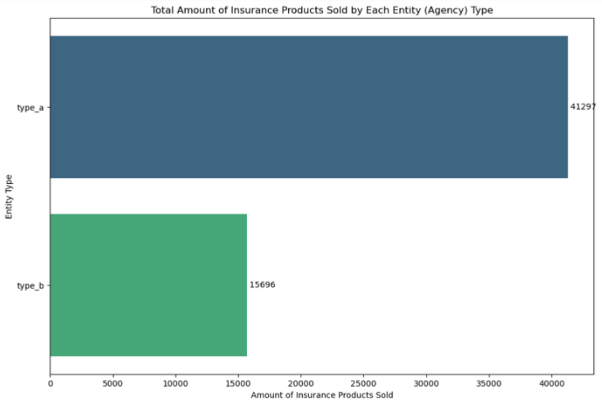


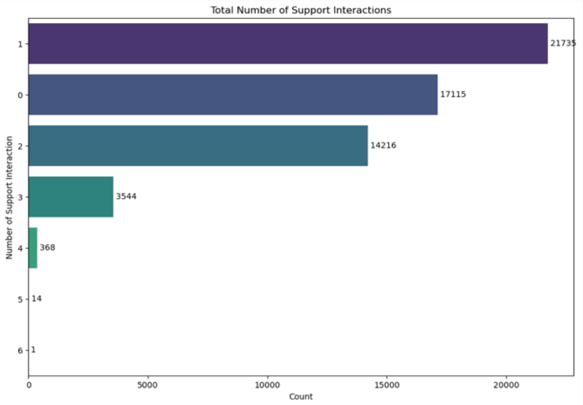


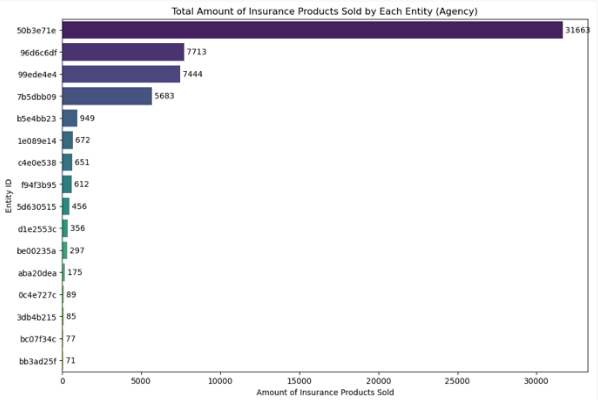
Exploratory Data Analysis

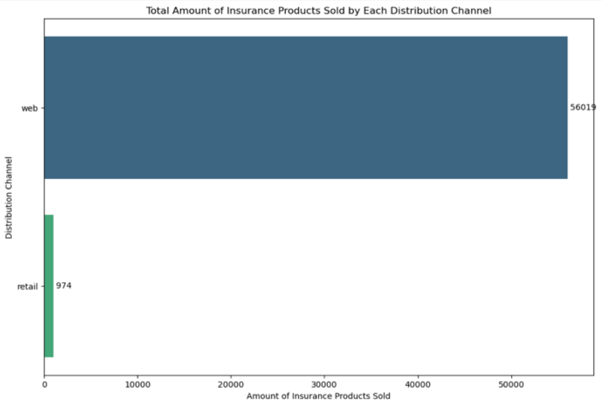
For agent ID, entity type, entity ID, support interactions, gender and channel, it seems there are duplicated, meaningless and unimpactful variables to target.

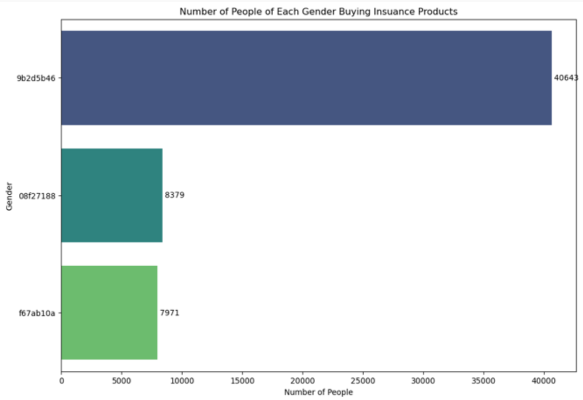












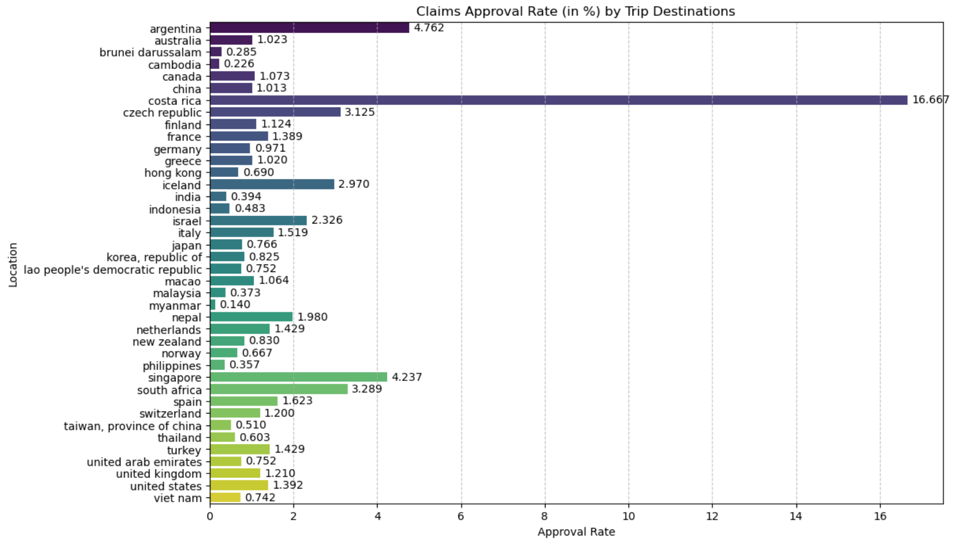
Exploratory Data Analysis

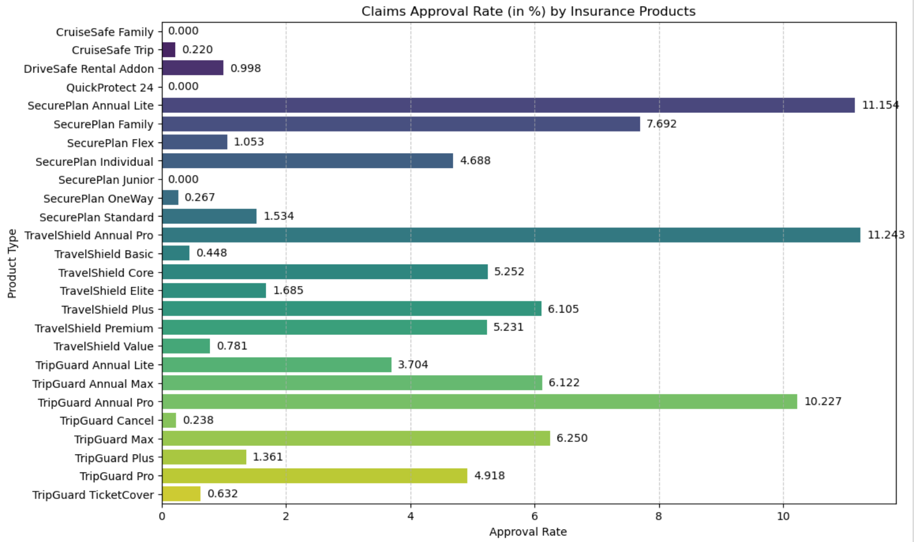
For location:

- Over 140+ countries in observations
- Only 20+ being approved for claims
- Rare distributions (e.g. 11942 observed for Singapore and more than half are below 100 observed)

For insurance product:

- Around 20+ types in observations
- Rare distributions (e.g. 16795 for most popular choices but some plans are only few observed)
- Approval ratio is random



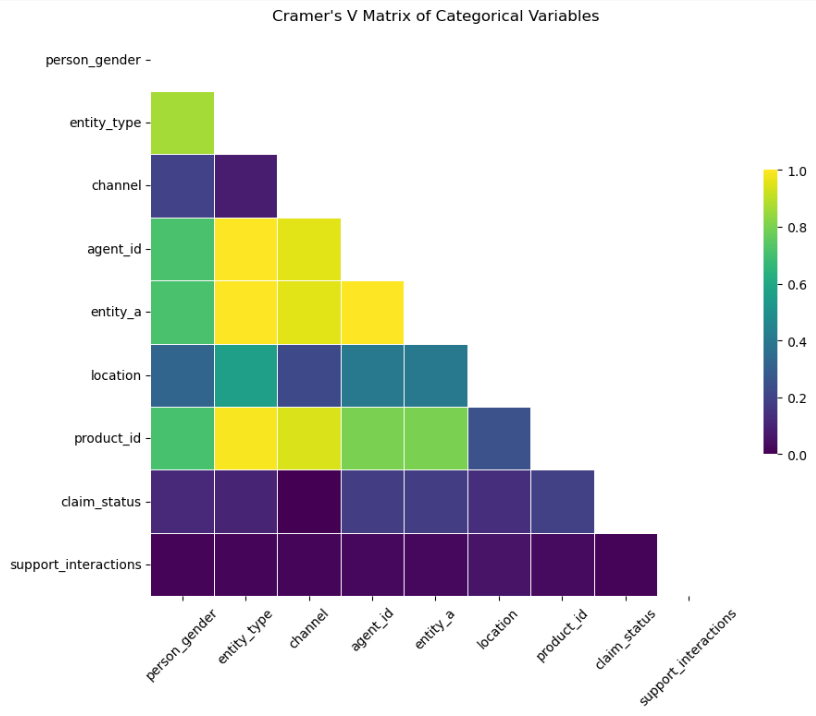


Train / Validation / Test split on training sample (insurance_train.csv)

Before feature selection & engineering:

- Train Set (60%), Validation Set (20%), and Internal Test Set (20%) split
- Avoid data leakage while feature selected and engineered based on train split only
- stratify = insurance['claim_status'] used to ensure all splits maintain the same proportion of 0s and 1s
- Validation and interval testing helping better estimate of generalization and stable / reliable model on minority class

Feature Selection



Cramer's V for categorical target and categorical inputs:

- support_interactions -> nearly 0 importance to all
- entity_a, entity_type and agent_id -> identical, perfectly one-to-one and consistent (coefficient = 1)
- channel -> highly associated to product_id and entity (coefficient > 0.95) and 0 association to target

Feature Selection

Variable	p-value
revenue	2.986475e-140
reward	4.529682e-71
trip_length	6.592646e-42
customer_score	4.212666e-23
person_age	3.990284e-04

ANOVA for categorical target
and continuous inputs:

All features are statistically significant.

In short, following variables will be removed:

entity_a
entity_type
channel
support_interactions

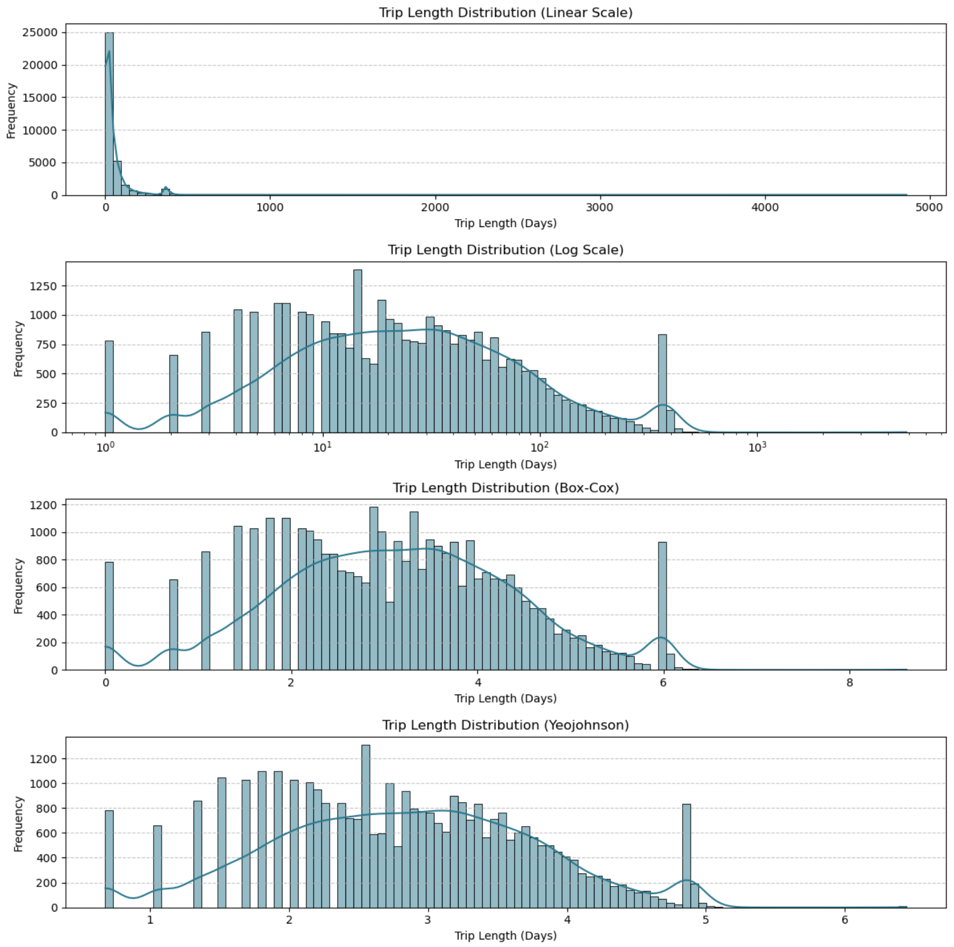
Feature Engineering

No zero values

Original skewness of trip_length: 23.84

Log Skewness: -0.01
Box-Cox Skewness (λ = 0.00): -0.00
Yeo-Johnson Skewness (λ = -0.07): 0.01

Log transformation is chosen.



Feature Engineering

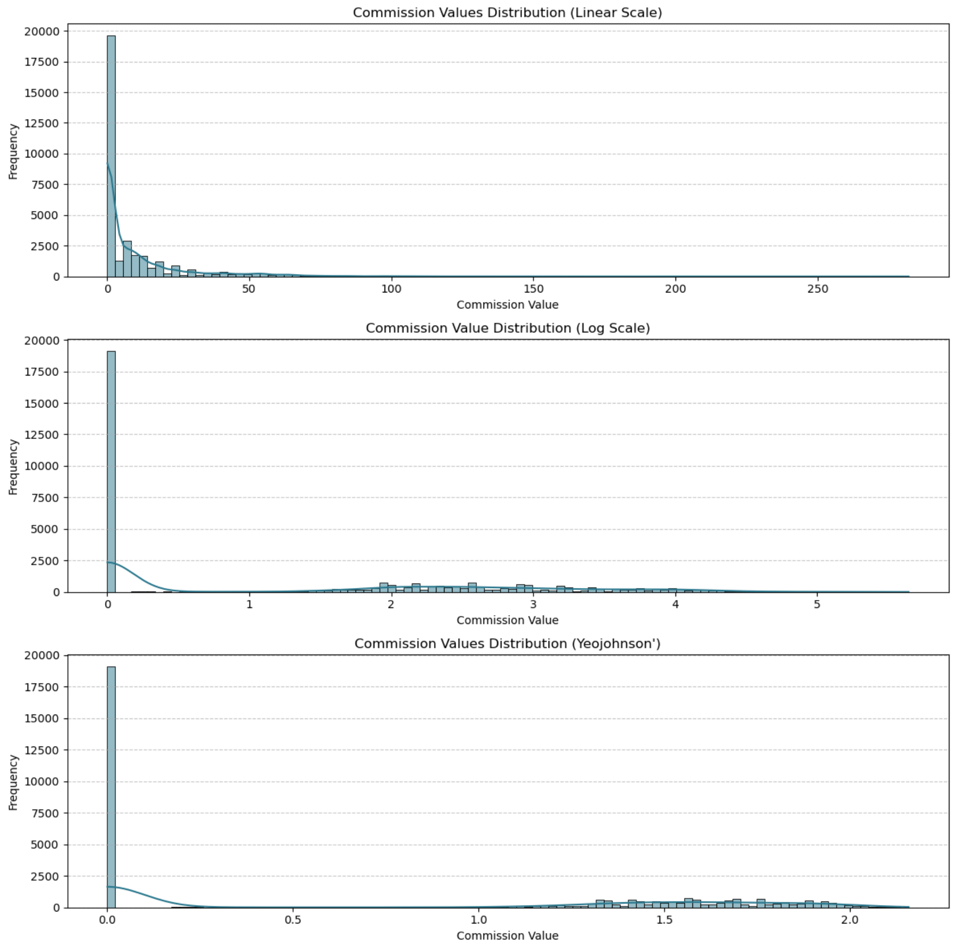
Including zero values

Original skewness of reward: 3.97

Log Skewness: 0.67

Yeo-Johnson Skewness (λ = -0.42): 0.39

Yeo-Johnson transformation is chosen.



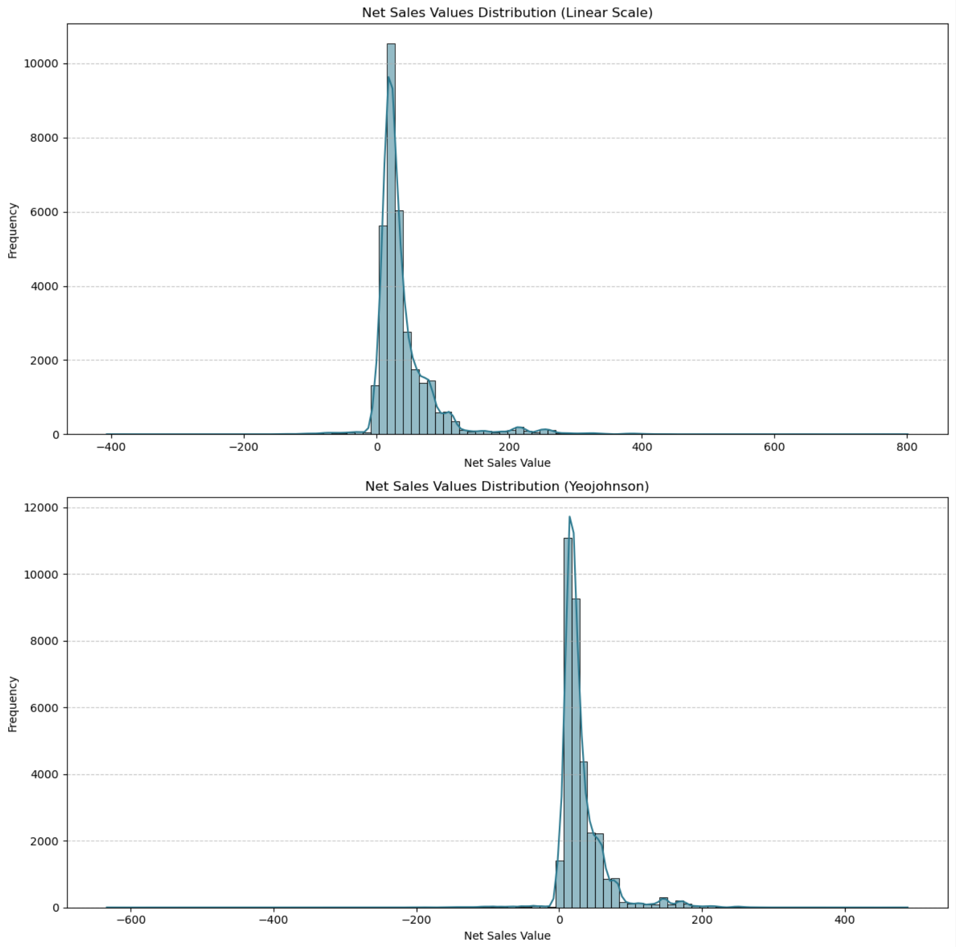
Feature Engineering

Including zero and negative values

Original skewness of revenue: 3.21

Yeo-Johnson Skewness (λ = 0.91): 1.33

Yeo-Johnson transformation is chosen.



Feature Engineering

Based on EDA, **Jenk's natural break** for person_age to 8 classes as ordinal (last group is merged with previous group due to small size)

Age binning	Counts	Encoded as
17.999, 26.0	2301	1
26.0, 32.0	4457	2
32.0, 36.0	10600	3
36.0, 42.0	7845	4
42.0, 51.0	4494	5
51.0, 62.0	2783	6
62.0, 100.0	1715	7

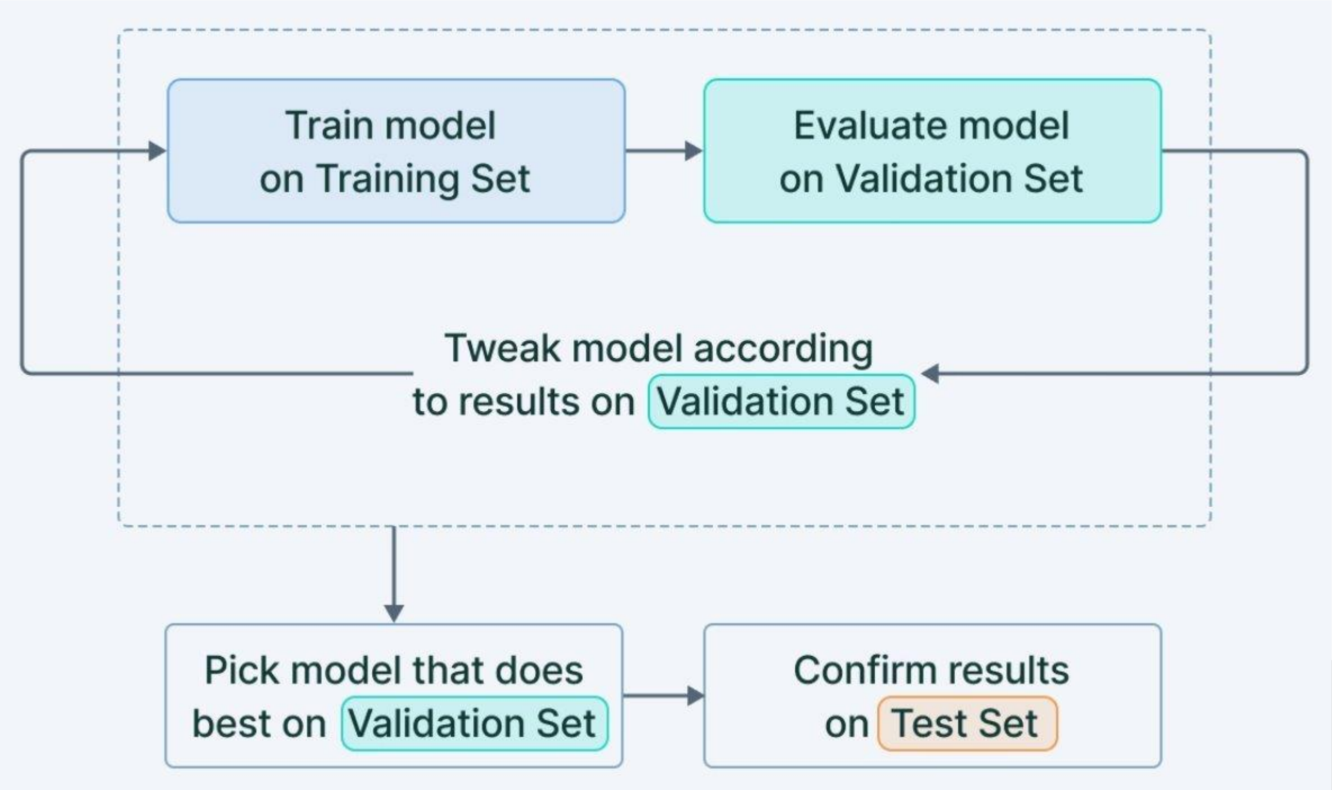
Based on EDA, **frequency binning** as ordinal for location to 4 groups of popularity

Location binning	Counts	Encoded as
Most popular	>= 1000	4
Moderate popular	>= 100	3
Less popular	>= 10	2
Least popular	< 10	1

One-hot Encoding as Dummy

person_gender, agent_id, product_id (names with "Family" grouped as "Other" due to small size)

Models Training



Model Consideration

4 continuous variable, 2 ordinal variables, and 41 dummy variables

SVM	KNN	Logistic Reg. (Elastic Net)
• RBF • Polynomial • Linear		• L1 Lasso (1) • L2 Ridge (0)
StandardScaler() - perform better if 0 mean and unit variance	**MinMaxScaler()** - Deal with the sensitivity of distance in the same range [0, 1]	**StandardScaler()** - help with magnitude

Model Consideration

To handle imbalance dataset, four sampling techniques are considered and only fit to training split.

Each model will try with each sampling techniques to compare the differences.

SMOTE	**SMOTETomek**	**Over Sampling**	**Under Sampling**
• Create synthetic samples for class 1 • Keep class 0 • Better generalization	• Same functions as SMOTE • Clean overlapping samples from both classes • Reduce noise in the decision boundary area	• Undersample class 0 which will case severe info. Loss • Not good to SVM/KNN if requires sufficient data	• Risk of overfitting due to simple oversampling existing class 1

Not preferred but still add for comparison

Model Consideration

1. Always apply the sampling method before scaling

Sampler

↓

Preprocessor

Scaler + Remainder

↓

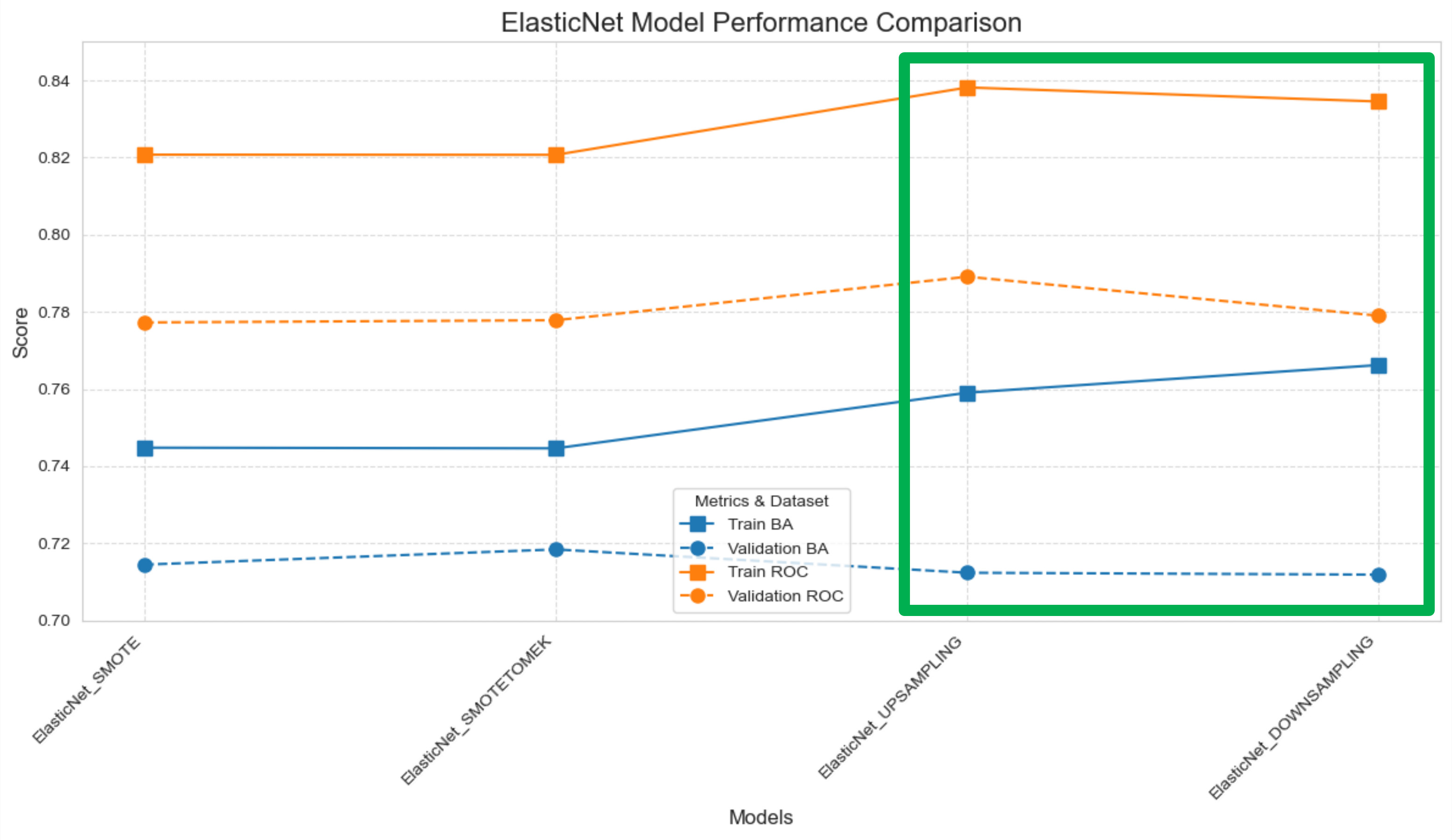
Model

2. Add a preprocessor to ensure only the numeric features being scaled

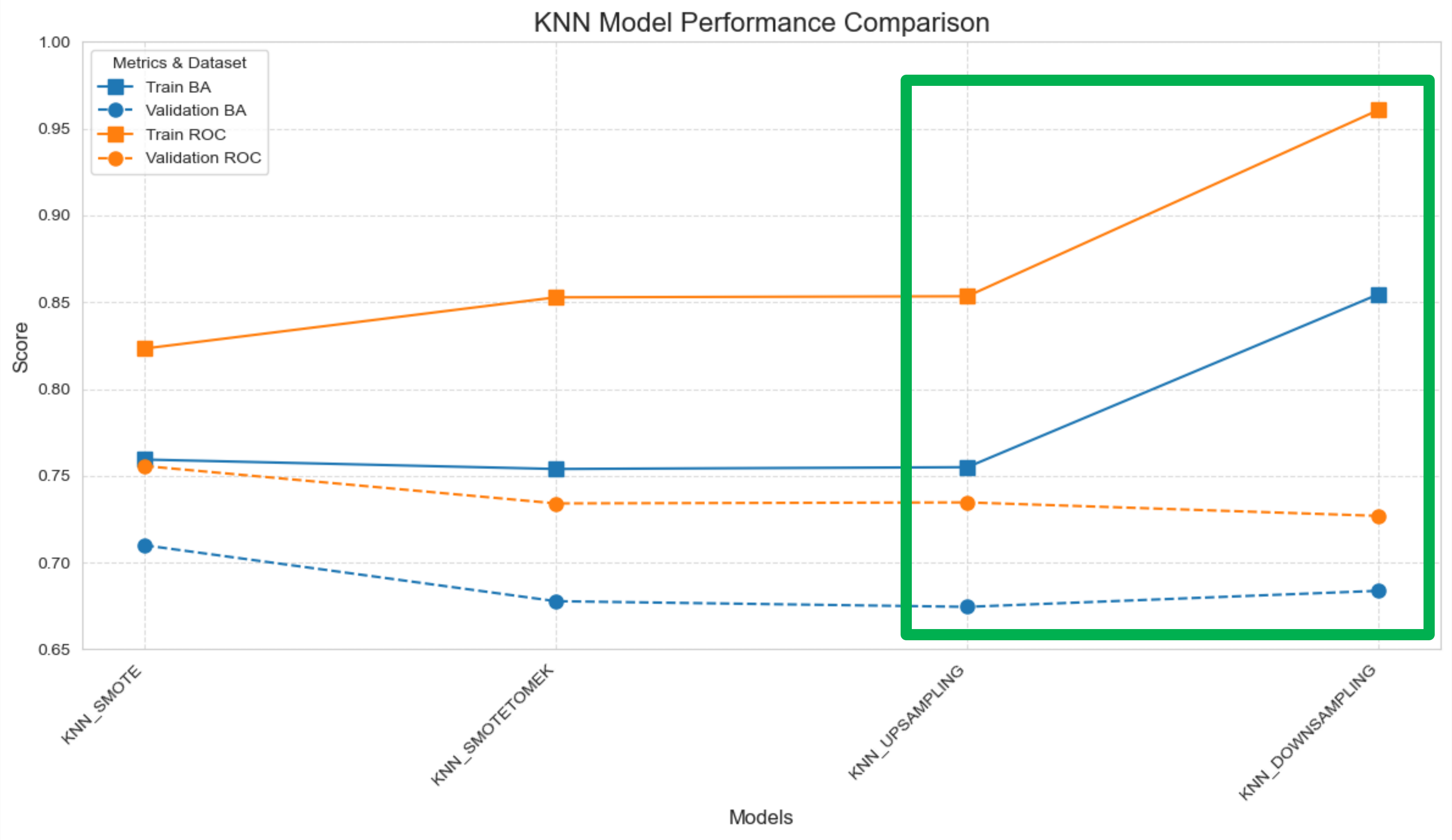
3. Fit the models with modified grids

- RandomizedSearchCV(): reduce computational time
- StratifiedKFold(n = 5): ensure each fold has the same proportion of observations with target
- Scoring = balanced accuracy (prioritized)

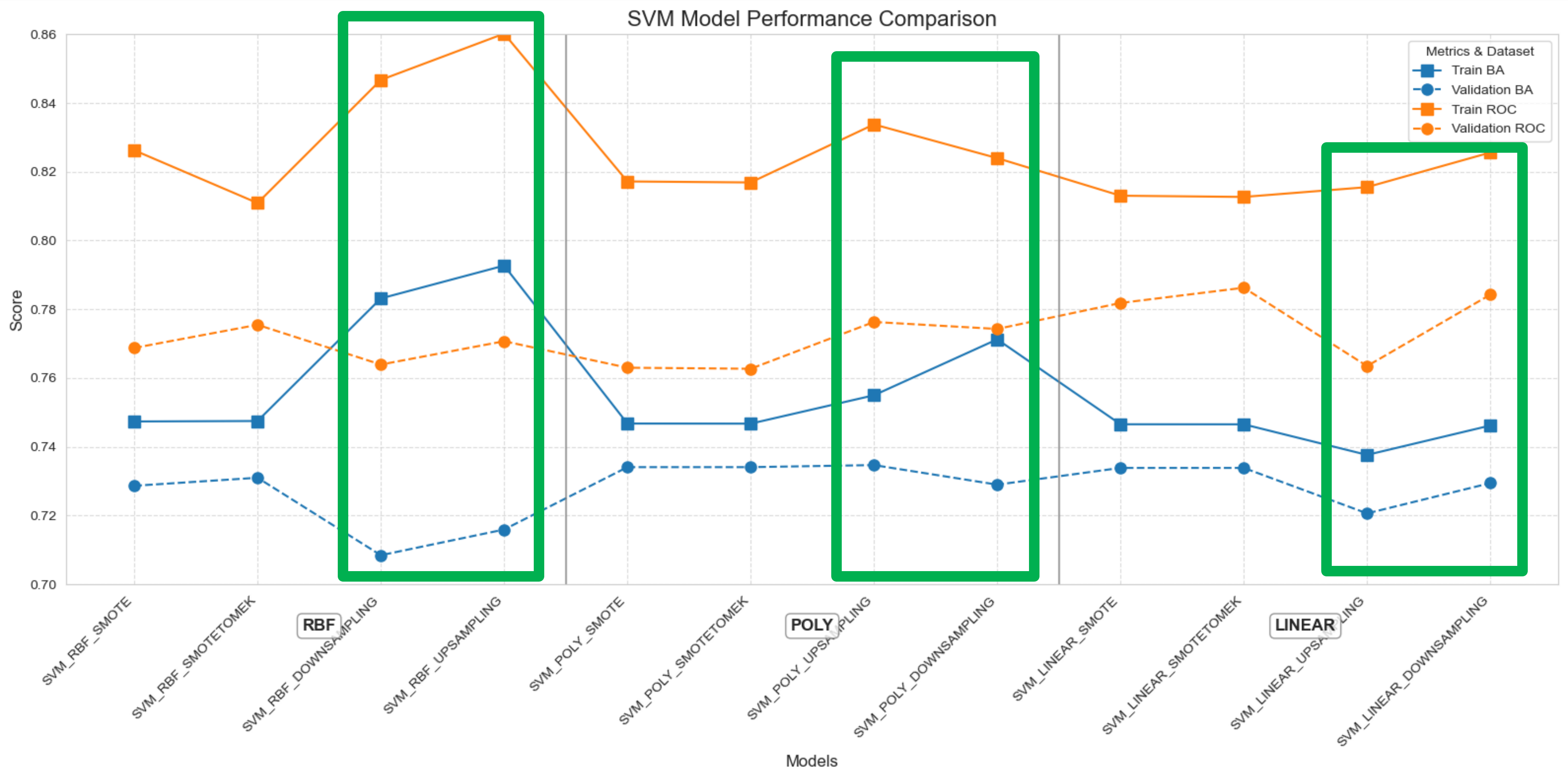
Model Consideration



Model Consideration



Model Consideration



Model Consideration

Using SMOTETomek:

Logistic Regression (Elastic Net)	**KNN** *(SMOTE)*	**SVM(Poly)**	**SVM(Linear)**	**SVM (RBF)**
Train: - AUC: 0.82 Validation: - BA: 0.72 - AUC: 0.78	Train: - AUC: 0.82 Validation - BA: 0.71 - AUC: 0.76	Train: - AUC: 0.82 Validation: - BA: 0.73 - AUC: 0.76	Train: - AUC: 0.81 Validation: - BA: 0.73 - AUC: 0.79	Train: - AUC: 0.81 Validation: - BA: 0.73 - AUC: 0.78
Expected Value for each model (mean cross-validated BA scores)				
0.74	0.75	0.75	0.75	0.75

Model Consideration

By considering:

- Model overall stability
- Computational efficiency
- Scalability (data with heavy dummies and less numeric features)
- Interpretability
- Comparison of other metrics (Recall and F1-score for class 1)
- Risk of overfitting (prone to overfit minority class)

<u>Final expected value / BA : 0.75</u>

Selected model: **<u>SVM (Linear)</u>**

On Interval Test split:

- Train BA: 0.75
- Internal Test BA: 0.75
- Train AUC: 0.81
- Internal Test AUC: 0.80

Classification Report based on Test:

	precision	recall	f1-score	support
0	0.99	0.84	0.91	11232
1	0.06	0.65	0.11	167
accuracy			0.84	11399
macro avg	0.53	0.75	0.51	11399
weighted avg	0.98	0.84	0.90	11399

Prediction

Based on same feature engineering methods and pipeline:

Predicted Claim Status	**Counts**	**Ratio**
Class 0	5266	83.15%
Class 1	1067	16.85%

THANK YOU