Adult Income Analysis

Tan Yue Hang

Data Analysis & Supervised ML

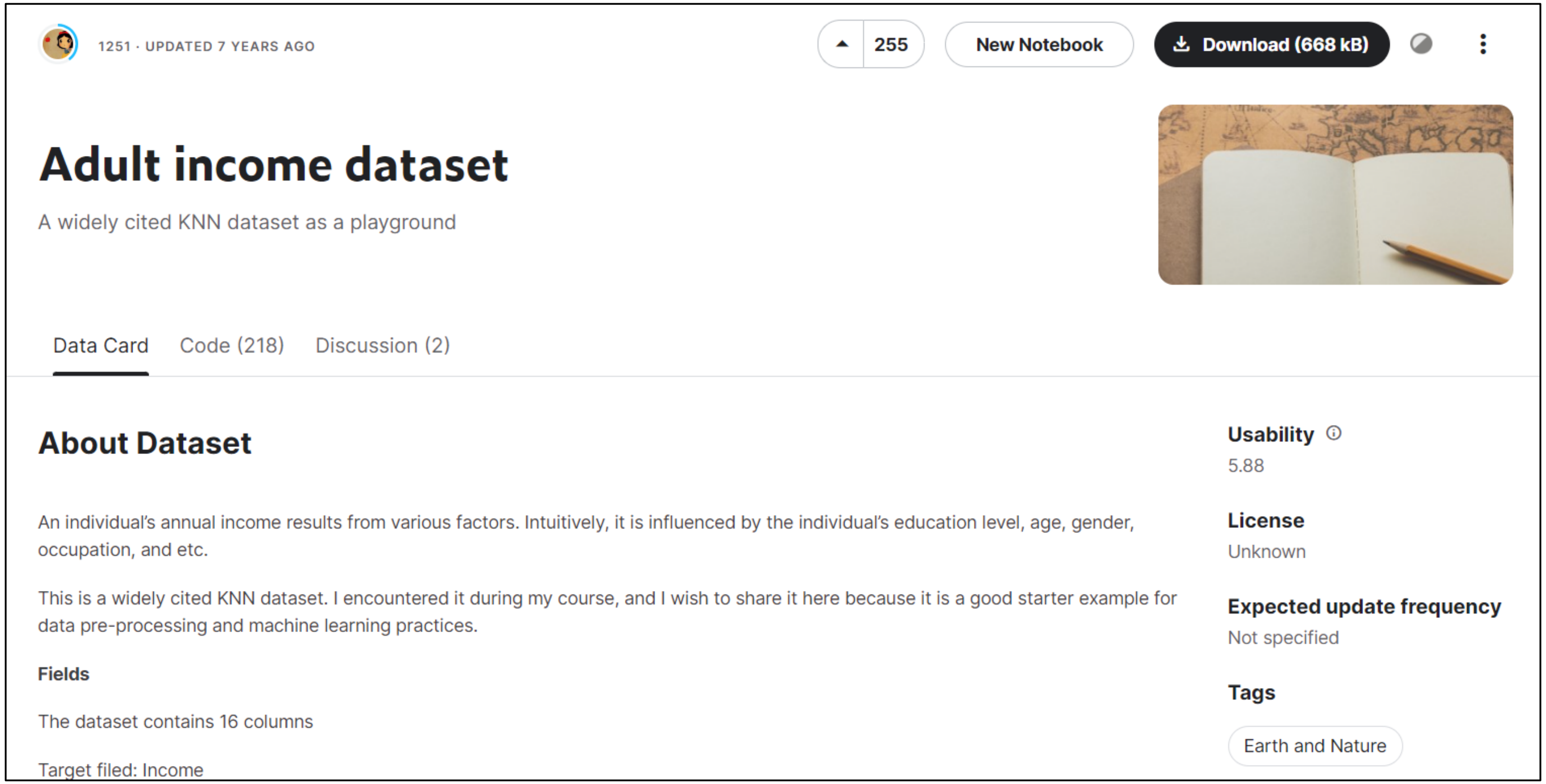
Mini Project 2

Introduction

- This project aims to look at what factors contribute to a working adult's salary, whether it is below or above the $50k threshold.
- A Machine learning model will be built in this project as well based on the selected features, to provide predictions on the income category that an adult falls into based on his age, education, country and many more.

Source of Dataset

- The dataset was acquired from Kaggle, which was also retrieved from UCI depository.
- While this dataset is not large (48,842 rows), it is quite messy that there are different data types and there are missing values which requires some efforts to clean up.



Presentation Contents

Section 1: Data Cleaning & EDA

Section 2: ML Modelling

- *Base models selection*
- *Feature selection*
- *Hyperparameters tuning*
- *Voting (Ensemble)*

Section 1: Data Cleaning & EDA

Dataset Information

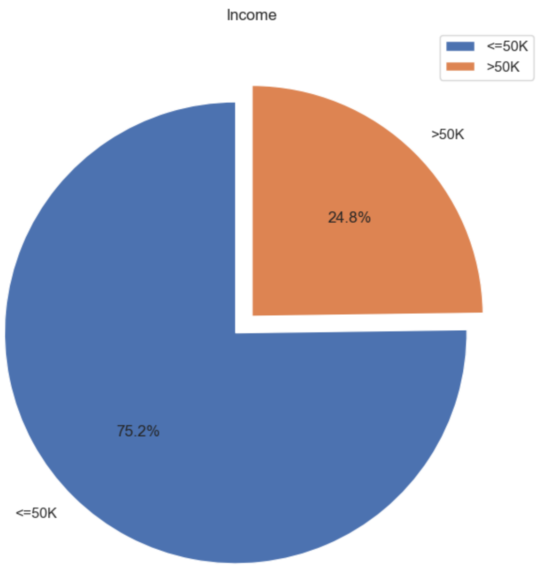
- The dataset comprises of 48,842 data rows and 15 features in total before data cleaning.
- The target (response) of the analysis would be “income”, which indicates two income categories of ≤50k and >50k.
- Six features are continuous variables, while 9 features are nominal categorical variables which needs to be encoded.
- There are missing values denoted as “?” which will be excluded from the dataset dropped. Data loss: 3620 rows (7.4%)

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48842 entries, 0 to 48841
Data columns (total 15 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   age              48842 non-null  int64 
 1   workclass        48842 non-null  object
 2   fnlwgt           48842 non-null  int64 
 3   education        48842 non-null  object
 4   educational-num  48842 non-null  int64 
 5   marital-status   48842 non-null  object
 6   occupation       48842 non-null  object
 7   relationship     48842 non-null  object
 8   race             48842 non-null  object
 9   gender           48842 non-null  object
 10  capital-gain     48842 non-null  int64 
 11  capital-loss     48842 non-null  int64 
 12  hours-per-week   48842 non-null  int64 
 13  native-country   48842 non-null  object
 14  income           48842 non-null  object
dtypes: int64(6), object(9)
memory usage: 5.6+ MB
```

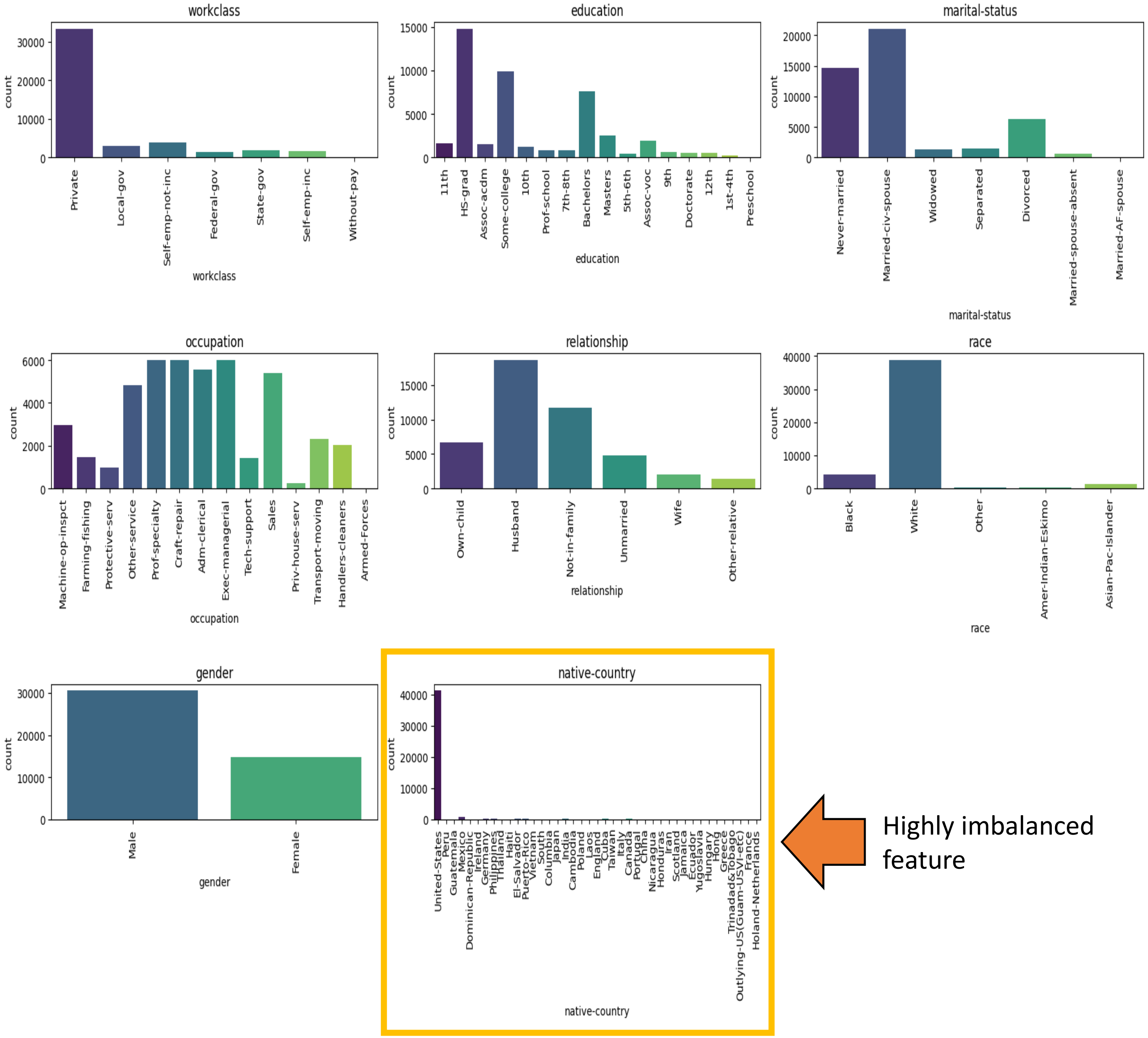
	age	workclass	fnlwgt	education	educational-num	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	native-country	income
0	25	Private	226802	11th	7	Never-married	Machine-op-inspct	Own-child	Black	Male	0	0	40	United-States	<=50K
1	38	Private	89814	HS-grad	9	Married-civ-spouse	Farming-fishing	Husband	White	Male	0	0	50	United-States	<=50K
2	28	Local-gov	336951	Assoc-acdm	12	Married-civ-spouse	Protective-serv	Husband	White	Male	0	0	40	United-States	>50K
3	44	Private	160323	Some-college	10	Married-civ-spouse	Machine-op-inspct	Husband	Black	Male	7688	0	40	United-States	>50K
4	18	?	103497	Some-college	10	Never-married	?	Own-child	White	Female	0	0	30	United-States	<=50K
5	34	Private	198693	10th	6	Never-married	Other-service	Not-in-family	White	Male	0	0	30	United-States	<=50K
6	29	?	227026	HS-grad	9	Never-married	?	Unmarried	Black	Male	0	0	40	United-States	<=50K
7	63	Self-emp-not-inc	104626	Prof-school	15	Married-civ-spouse	Prof-specialty	Husband	White	Male	3103	0	32	United-States	>50K
8	24	Private	369667	Some-college	10	Never-married	Other-service	Unmarried	White	Female	0	0	40	United-States	<=50K
9	55	Private	104996	7th-8th	4	Married-civ-spouse	Craft-repair	Husband	White	Male	0	0	10	United-States	<=50K

Features Exploration
(Categorical, w/o One-Hot Encoding)

- Most people work in private sector.
- High school and college graduates make up most of the samples, while people who have a bachelor's degree seem to make up a fair number as well.
- The attention-catching feature is native country, where the data is highly imbalanced.



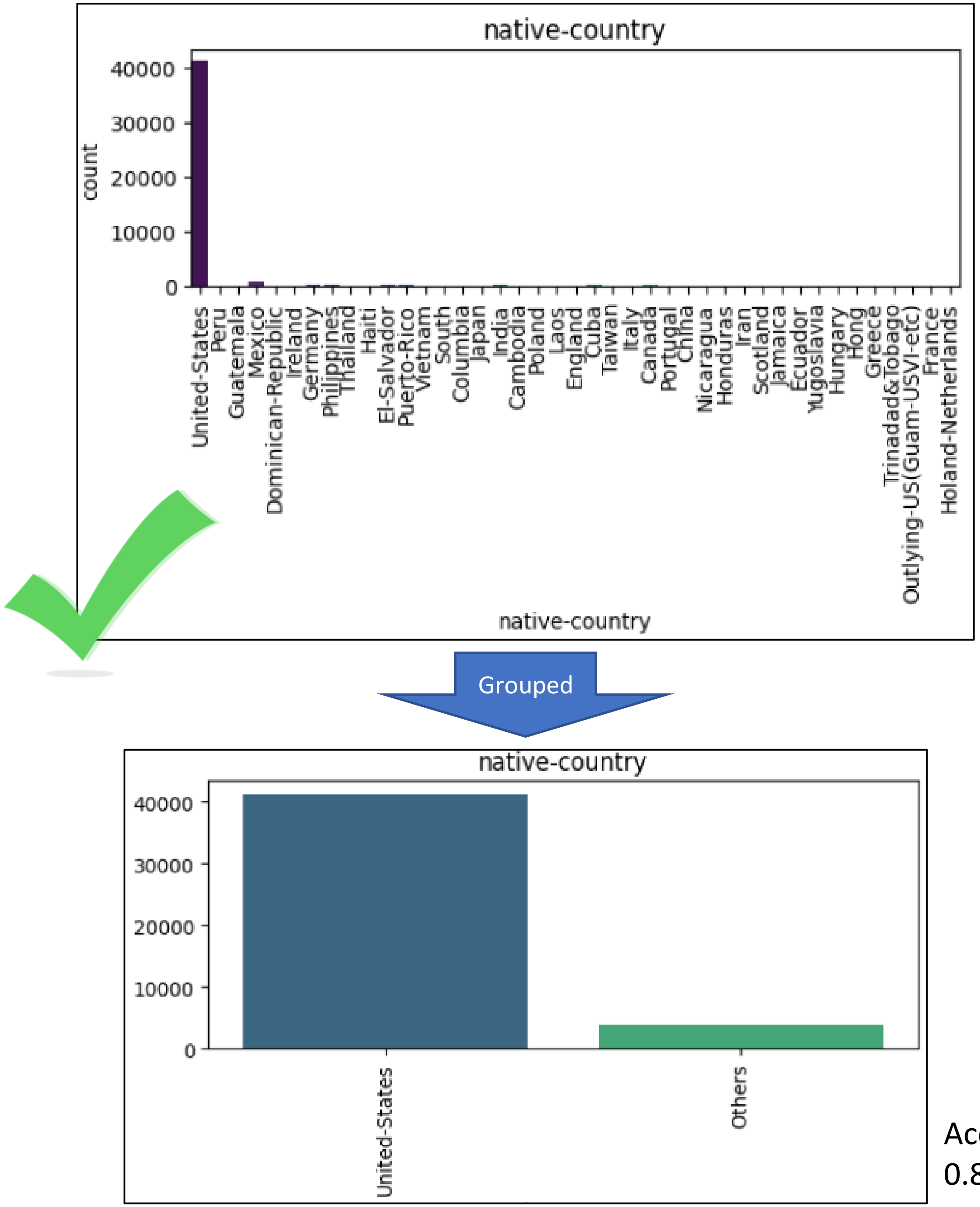
Target



Features

Features Exploration
(Categorical, w/o One-Hot Encoding)

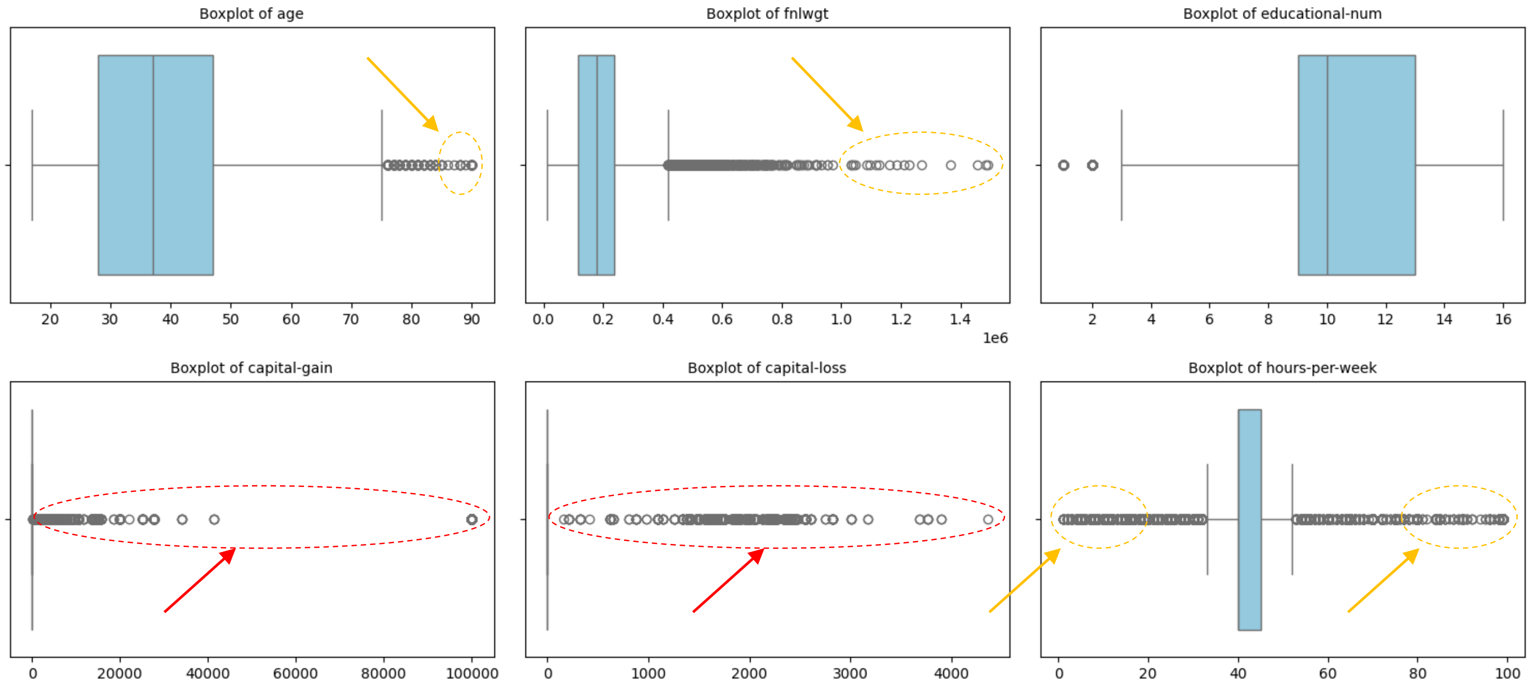
- Tried to fit and evaluate the models several times (not shown here) with all the other countries grouped as one and named as “Others”, but turned out the accuracy score was slightly worse than without grouping.
    - Both were evaluated with training data.
- Therefore will proceed with the original data just as it.



Accuracy: 0.8707

Accuracy: 0.8690

Extreme Outliers Removal
(Continuous)

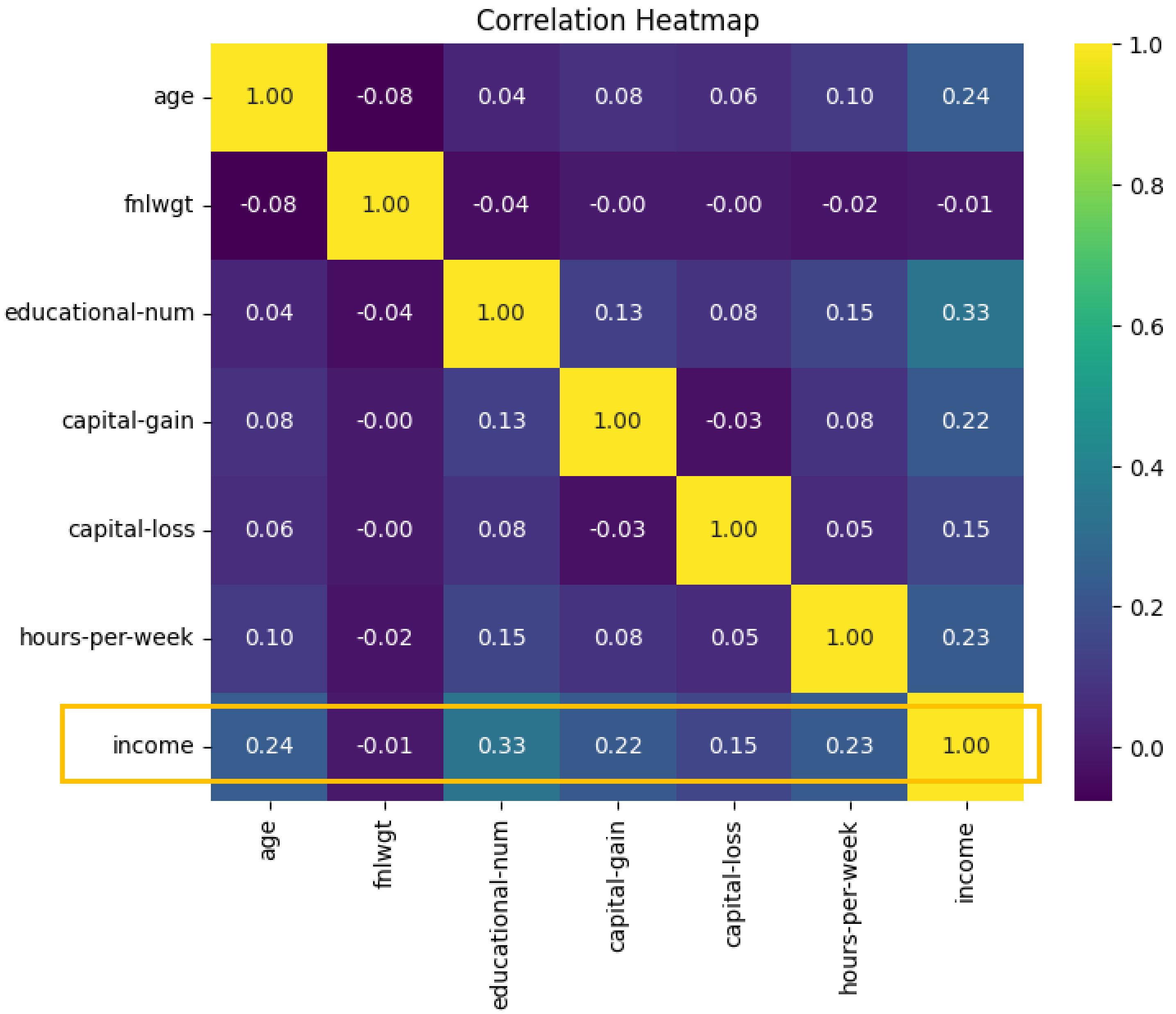


- Extreme outliers removal is not feasible with this dataset, mainly because of the “capital-gain” and “capital-loss” columns.
- Due to the nature of the heavily skewed distribution of these two columns, even if <u>IQR threshold is set as high as 4</u>, these two columns will just disappear while non of the outlying points from “age” and “fnlwgt” will get removed at all.
- On the other hand, if both “capital-gain” and “capital-loss” are excluded entirely, the model’s performance would actually become worse.
- As a results the decision is **not to remove** any outlier of any column at all.

Pearson Correlation Plot

(Continuous Data only)

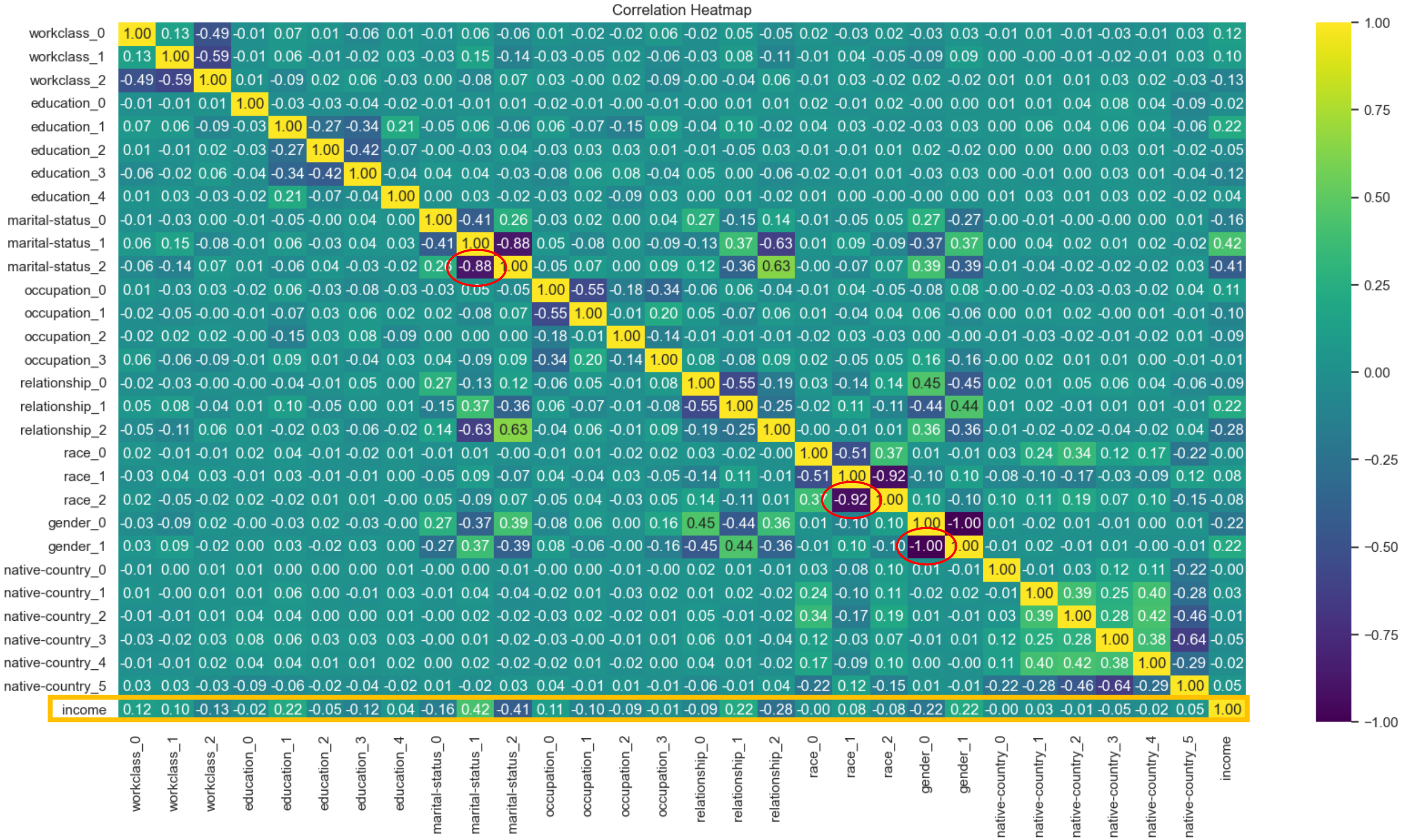
- All the numerical features appear to have fairly significant correlation with the target of income, except for “fnlwgt”.
- It’s not explained in the data source what “fnlwgt” is.



Spearman Correlation Plot

(Categorical Features with Binary Encoding)

- Binary encoding is used for all the categorical features.
    - All the categorical features are nominal, hence ordinal encoding (or label encoding) is not suitable.
    - One-hot encoding will increase the dimensionality by too much.
- Some features are observed to have <u>very high correlation with each other</u>, for example marital status 2 vs. marital status 1, race 2 vs. race 1, gender 1 vs. gender 0 and so on.
- This is not good as it may introduce multicollinearity which could make the model too complex and lead to unstable performance.
- But we will leave it as it, as we will try <u>RFECV</u> and see how will it deal with these.



Scaling
(Continuous Data)

- By reviewing the numerical columns, it appears that the scale of these columns are significantly different.
  - Large variation in scale in between features could potentially lead to misleading weights in the model.
- Apply **StandardScaler** to bring all the numerical columns to the same scale.

```
# Before the columns are scaled.
df_binary[scaled_columns][:5]
```

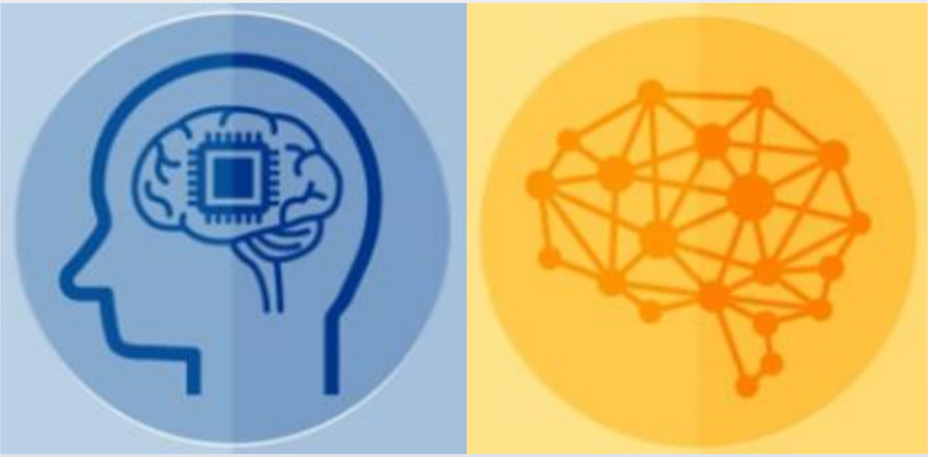
	age	fnlwgt	educational-num	capital-gain	capital-loss	hours-per-week
0	25	226802	7	0	0	40
1	38	89814	9	0	0	50
2	28	336951	12	0	0	40
3	44	160323	10	7688	0	40
5	34	198693	6	0	0	30

After scaling

```
# After the columns are scaled.
df_binary[scaled_columns][:5]
```

	age	fnlwgt	educational-num	capital-gain	capital-loss	hours-per-week
0	-1.024983	0.350889	-1.221559	-0.146733	-0.21878	-0.078120
1	-0.041455	-0.945878	-0.438122	-0.146733	-0.21878	0.754701
2	-0.798015	1.393592	0.737034	-0.146733	-0.21878	-0.078120
3	0.412481	-0.278420	-0.046403	0.877467	-0.21878	-0.078120
5	-0.344079	0.084802	-1.613277	-0.146733	-0.21878	-0.910942

Section 2: ML Modelling



Cartoon source: https://www.futurefundamentals.com/

ML Workflow & Strategy

<u>Step 1: Base Models Selection</u>

- 7 models
- Training data
- Best accuracy score get selected

<u>Step 2: Features Selection</u>

- RFECV
- Training data
- Best accuracy score get selected

<u>Step 3: Hyperparameters Tuning</u>

- RandomizedSearchCV
- Training and/or Validation data
- Best accuracy score get selected

<u>Step 4: Evaluation with Test Data</u>

- Final model trained with training data
- Validation and Test data
- Evaluated with CV and ROC AUC

Define Features and Target

	Dataset	Rows	No. of Features
Training	X_train	28941	35
	y_train	28941	1
Validation	X_val	7236	35
	y_val	7236	1
Test	X_test	9045	35
	y_test	9045	1

- The data has been split into training, validation and test sets, where:
  - Training data: For base model selection, features selection, and hyperparameters tuning.
  - Validation data: Validate the selected model (be it tuned or original model) to check the performance, and if there is any sign of overfitting.
  - Test data: To perform final evaluation of the model.

Base Model Selection

- A few models have been chosen for base model selection.
- Initial selection is based on the Scikit Learn website, as well as some additional info from articles.
- Most of the chosen base models are boosting estimators, which normally have better performance.
- CatBoostClassifier appears to give the best accuracy among all and thus selected.

Scores of the Base Models

Base Models	**Accuracy**	Precision	Recall	F1	Elapsed Time
LogisticRegression	0.8401	0.7258	0.5763	0.6424	6.676651
GaussianNB	0.7493	0.5007	0.8042	0.6158	1.131334
HistGradientBoostingClassifier	0.8659	0.7760	0.6496	0.7072	8.421102
SGDClassifier	0.8366	0.6976	0.6112	0.6502	2.029939
RandomForestClassifier	0.8471	0.7289	0.6156	0.6674	17.759419
XGBClassifier	0.8647	0.7652	0.6599	0.7086	3.753199
CatBoostClassifier	0.8661	0.7763	0.6500	0.7075	94.828684

Features Selection

- Use Recursive Feature Elimination with Cross-Validation (**RFECV**) package from Scikit-Learn to perform features selection.
  - Most of the countries have been excluded by RFECV.
- Training data was used.
- With this,
  - **27** out of 34 features have been **selected**.
  - Accuracy score improved from **0.8661** to **0.867**

Features	Selection
age	True
workclass_0	True
workclass_1	True
workclass_2	True
fnlwgt	True
education_0	False
education_1	True
education_2	True
education_3	False
education_4	True
educational-num	True
marital-status_0	True
marital-status_1	True
marital-status_2	True
occupation_0	True
occupation_1	True
occupation_2	True
occupation_3	True
relationship_0	True
relationship_1	True
relationship_2	True
race_0	True
race_1	False
race_2	True
gender_0	True
gender_1	True
capital-gain	True
capital-loss	True
hours-per-week	True
native-country_0	False
native-country_1	False
native-country_2	False
native-country_3	True
native-country_4	False
native-country_5	False

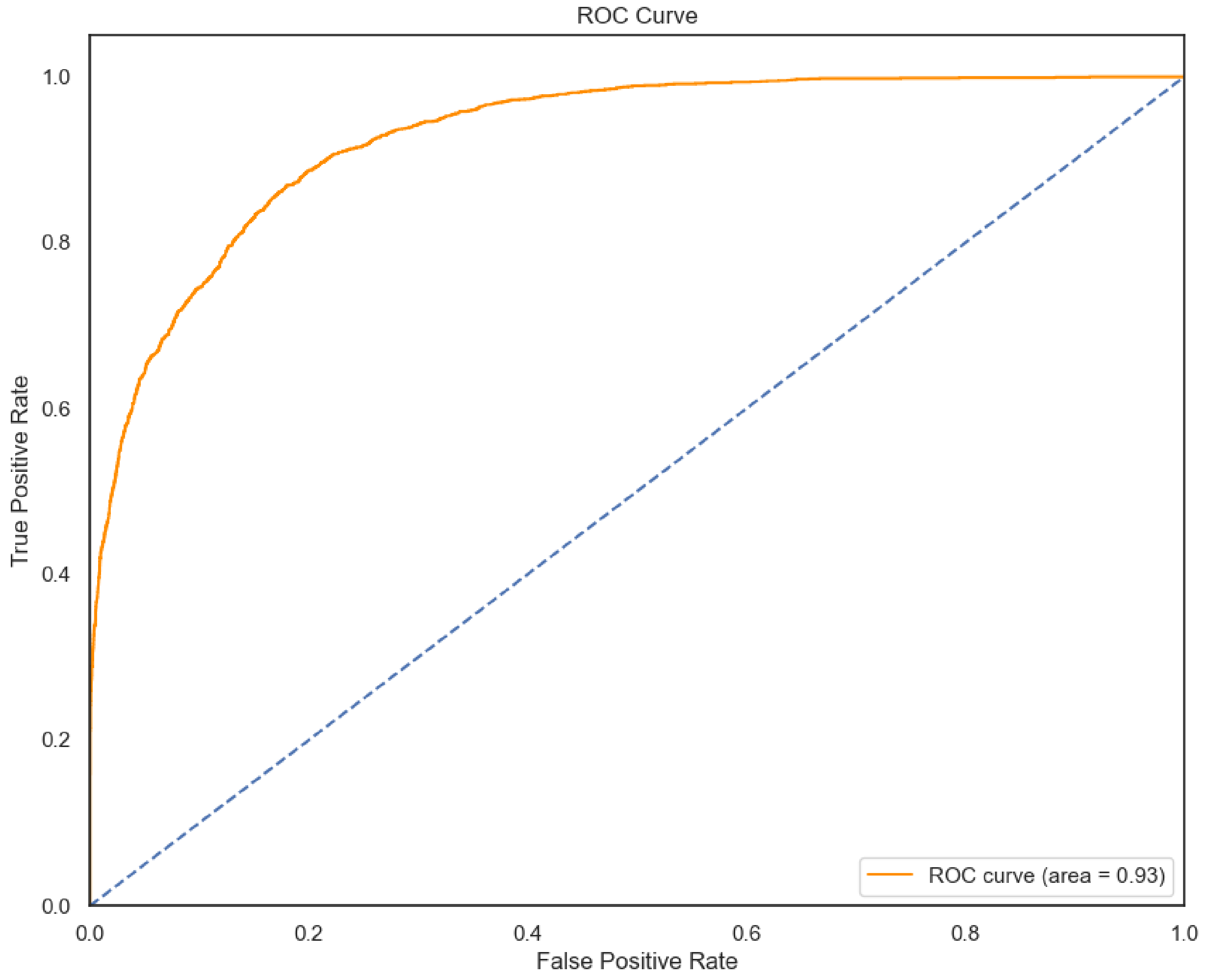
Model	**Accuracy**	Precision	Recall	F1	Elapsed Time
CatBoostClassifier (Baseline)	0.8661	0.7763	0.6500	0.7075	94.828684
CatBoostClassifier (Selected features)	0.8670	0.778	0.6529	0.71	312.726567

Hyperparameters Tuning

- RandomizedSearchCV has been used to save computational resources.
- Parameters like depth, learning rate, l2 leaf reg, iterations and custom loss are to be tuned.
- New training data with selected features is used so that we can make apple-to-apple comparison with the base model accuracy. Validation data can be used as well to confirm the consistency.
- Results: The best accuracy score obtained was **0.8674** which is slightly better than the base model (0.8670).
- <u>Decision: Use the tuned model.</u>

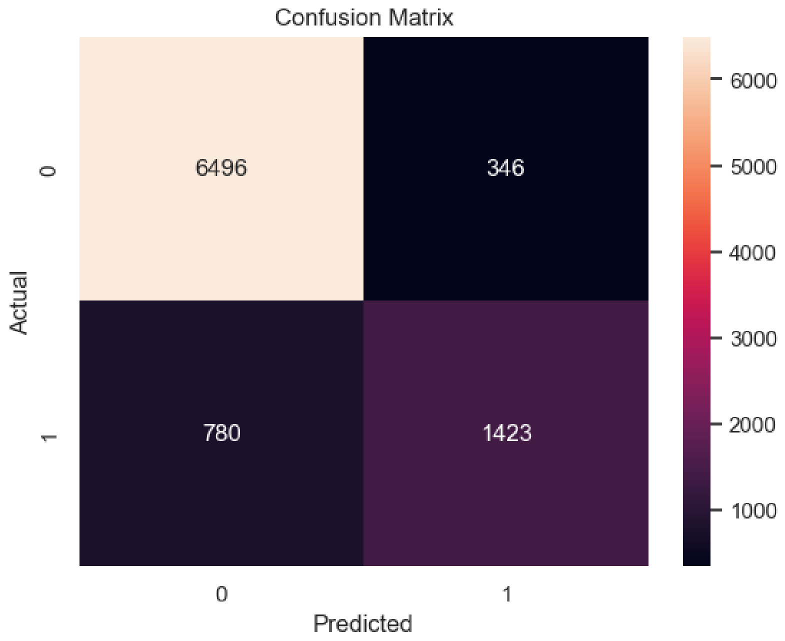
```
param_grid = {
    'depth': [6, 8, 10],
    'learning_rate': [0.01, 0.05, 0.1],
    'l2_leaf_reg': [1, 3, 5, 7, 9],
    'iterations': [100, 150, 200],
    'custom_loss': ['Logloss', 'CrossEntropy']
}
```

Model	**Accuracy**	Precision	Recall	F1	Elapsed Time
CatBoostClassifier (Baseline)	0.8661	0.7763	0.6500	0.7075	94.828684
CatBoostClassifier (Selected features)	0.8670	0.778	0.6529	0.71	312.726567
CatBoostClassifier (Selected features + Tuned model)	0.8674	-	-	-	-



Validation & Final Score

- Model used: CatBoostClassifier base model
- Training data: X_train_new, y_train
- Validation data: X_val, y_val to validate the consistency after hyperparameters tuning
- Final score: X_test, y_test



Model	Data Set	**Accuracy**	Precision	Recall	F1	Elapsed Time
CatBoostClassifier (Tuned)	X_val, y_val	0.8603	0.76	0.6338	NaN	14.598181
	X_test, y_test	**0.864**	0.7774	0.6194	0.6894	9.120915

Summary & Final Thoughts

- Had tried ***VotingClassifier*** with RandomForestClassifier, HistGradientBoostClassifier, and CatBoostClassifier but the results didn't seem to improve at all (not shown here), so this has been excluded from the notebook.
- Had also tried a few SVC, NuSVC, and GaussianNB but the resulting accuracy was quite low, on top of that SVC took much longer time to train compared to other estimators.
  - As a result, these weak learners have already been excluded from the notebook.
- Some Kagglers managed to get higher accuracy scores at around 0.87 or even 0.91, but cross-validation was not used.
  - There are variations and randomness to the resulting scores, and CV allows us to take the mean / median of these variations.
  - Cross-validation can make a more robust evaluation by taking the mean of multiple splits.

Bonus: CatBoostClassifier without Encoding

Model	Data Set	Accuracy	Precision	Recall	F1	Elapsed Time
CatBoostClassifier	0.8703	0.7819	0.665	0.7187	3052.70	0.8703
	0.8596	0.7572	0.6351	0.6905	431.46	0.8596
	0.8668	0.7793	0.6321	0.6979	334.80	0.8668

- CatBoostClassifier capability: Able to process string / text data without encoding.
  - Built-in target encoding capability.
- The results are slightly better than manually encoded categorical data.
- On top of that, no feature selection was done

End of Presentation