# Predict Clicked Ads Customer Classification by using Machine Learning

**Created by:**
**Muhammad Cikal Merdeka**
**Email :** mcikalmerdeka@gmail.com
**LinkedIn :** linkedin.com/in/mcikalmerdeka
**Github :** github.com/mcikalmerdeka

Dedicated entry-level data scientist with analytical and experimental background of Physics. My graduation 2023, a pivotal year marked by significant advancements in artificial intelligence with the introduction of GPT-4 and other generative AI models, has fueled my curiosity and excitement to delve into the field of data. I have comprehensive grasp of data science methodology from business understanding to modelling process with proficiency in **Python, SQL, Tableau, Power BI, Looker Studio and other tools** related to data analytics workflow from several coursework and bootcamps.

**Supported by:**
**Rakamin Academy**
Career Acceleration School
www.rakamin.com

# Data Modeling

## Modelling Procedure Explanation

❑ In this case after we splitting the data 70-30 (train-test) for both **case A (non-scaled data)** and **case B (scaled data)**, we try modelling with several algorithms which are :

- Logistic Regression
- K-Nearest Neighbors
- Decision Tree
- Random Forest
- Gradient Boosting
- XGBoost

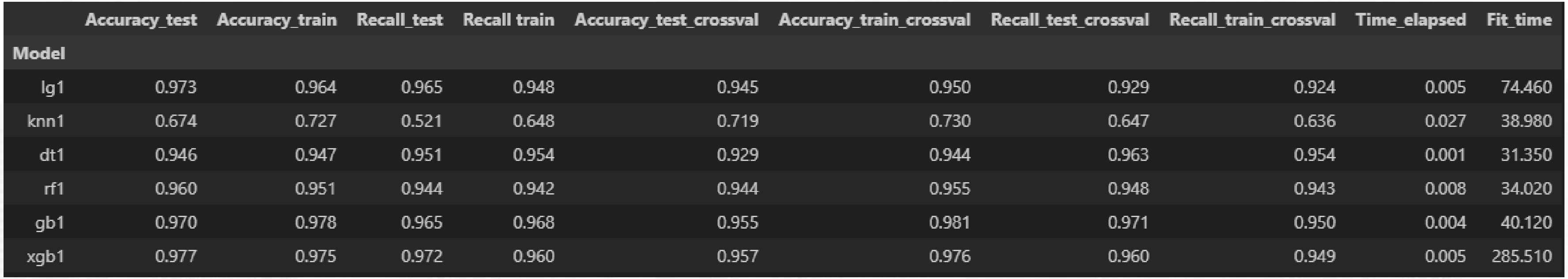
❑ In both case A and case B, after initial train with default hyperparameter (vanilla models), the result shows overfitting on most of the models. To overcome this problem, hyperparameter tuning in conducted and the overfitting has been significantly reduced. **The result that will be shown on the next slides are tuned models performance**.

❑ The main evaluation metric we will use is **Accuracy** because of balanced target. As for the secondary evaluation metric we will be using **Recall** because in business context we want to reduce the number of customers/users that is predicted not clicked on the ad while actually he/she wanted to clicked on the ad (False Negative) in order to maximize profit.

For more details, you can view the Jupyter notebook here.

# Data Modeling

## Models Performance Before Feature Scaling



| | Accuracy_test | Accuracy_train | Recall_test | Recall train | Accuracy_test_crossval | Accuracy_train_crossval | Recall_test_crossval | Recall_train_crossval | Time_elapsed | Fit_time |
|---|---|---|---|---|---|---|---|---|---|---|
| **Model** | | | | | | | | | | |
| lg1 | 0.973 | 0.964 | 0.965 | 0.948 | 0.945 | 0.950 | 0.929 | 0.924 | 0.005 | 74.460 |
| knn1 | 0.674 | 0.727 | 0.521 | 0.648 | 0.719 | 0.730 | 0.647 | 0.636 | 0.027 | 38.980 |
| dt1 | 0.946 | 0.947 | 0.951 | 0.954 | 0.929 | 0.944 | 0.963 | 0.954 | 0.001 | 31.350 |
| rf1 | 0.960 | 0.951 | 0.944 | 0.942 | 0.944 | 0.955 | 0.948 | 0.943 | 0.008 | 34.020 |
| gb1 | 0.970 | 0.978 | 0.965 | 0.968 | 0.955 | 0.981 | 0.971 | 0.950 | 0.004 | 40.120 |
| xgb1 | 0.977 | 0.975 | 0.972 | 0.960 | 0.957 | 0.976 | 0.960 | 0.949 | 0.005 | 285.510 |

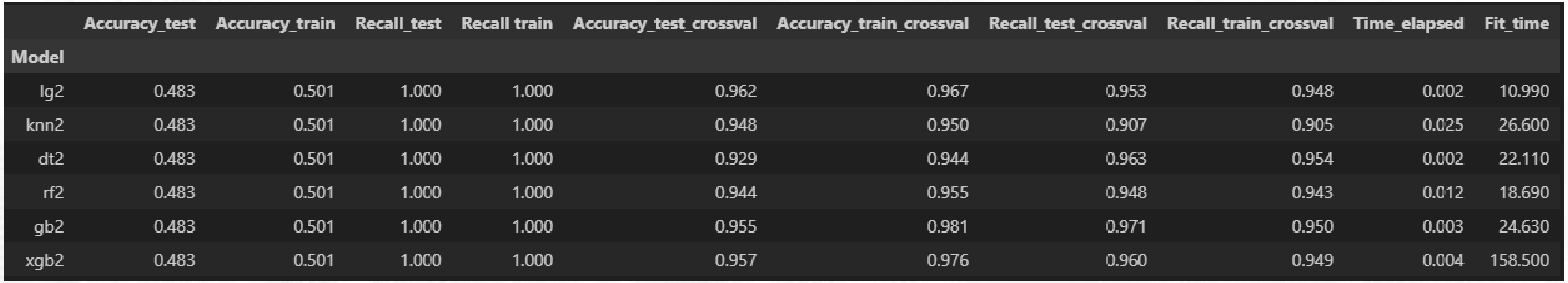
- Decision Tree model had the lowest fit time of all the models but the accuracy is second lowest accuracy overall.

- XGBoost model had the highest cross validated accuracy but Gradient Boosting has the highest cross validated recall. In the case of non-scaled data the 2 algorithm showed the best performance.

- Due to the non-scaled data, distance based algorithms like K-Nearest Neighbours suffered heavily as the accuracy and recall scores is the lowest of all models tested. Linear algorithms like Logistic Regression also suffered from this case in which the fitted time became the second longest.

# Data Modeling

## Models Performance After Feature Scaling



| | Accuracy_test | Accuracy_train | Recall_test | Recall train | Accuracy_test_crossval | Accuracy_train_crossval | Recall_test_crossval | Recall_train_crossval | Time_elapsed | Fit_time |
|---|---|---|---|---|---|---|---|---|---|---|
| Model | | | | | | | | | | |
| lg2 | 0.483 | 0.501 | 1.000 | 1.000 | 0.962 | 0.967 | 0.953 | 0.948 | 0.002 | 10.990 |
| knn2 | 0.483 | 0.501 | 1.000 | 1.000 | 0.948 | 0.950 | 0.907 | 0.905 | 0.025 | 26.600 |
| dt2 | 0.483 | 0.501 | 1.000 | 1.000 | 0.929 | 0.944 | 0.963 | 0.954 | 0.002 | 22.110 |
| rf2 | 0.483 | 0.501 | 1.000 | 1.000 | 0.944 | 0.955 | 0.948 | 0.943 | 0.012 | 18.690 |
| gb2 | 0.483 | 0.501 | 1.000 | 1.000 | 0.955 | 0.981 | 0.971 | 0.950 | 0.003 | 24.630 |
| xgb2 | 0.483 | 0.501 | 1.000 | 1.000 | 0.957 | 0.976 | 0.960 | 0.949 | 0.004 | 158.500 |

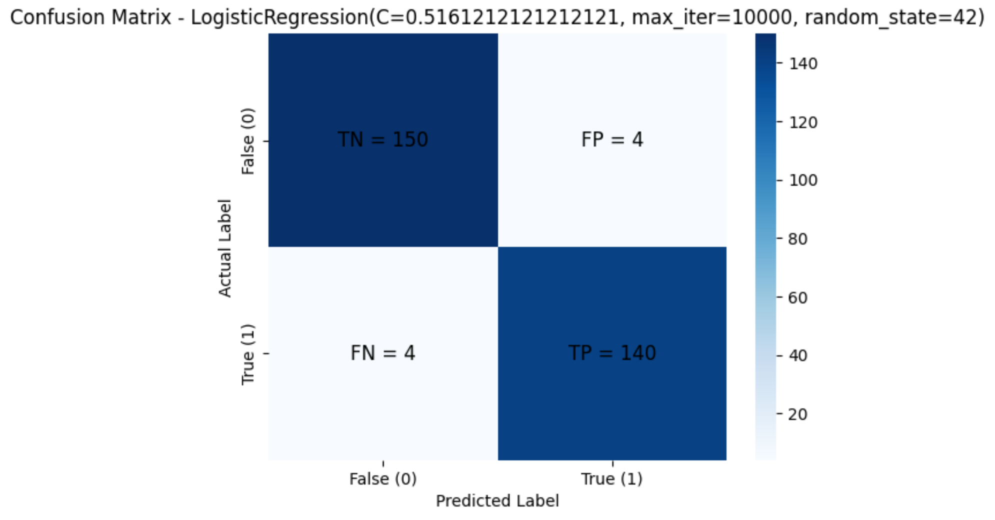
- The overall fit time of all models decrease after scalling is applied, this is because it's now easier for the algorithm to do the calcuation. Logistic Regression model now had the highest cross validated accuracy, while Gradient Boosting model still has the highest cross validated recall.

- With normalized data, the previously poor performing distance based and linear models have shone through. The performance of K-Nearest Neighbours improved really significant while for the Logistic Regression not only the performance in accuracy and recall increased but also the fit time became much faster.

- **By taking consideration of not only the above evaluation metrics but also the simplicity, explainability and fit and elapsed times, the model that will be chosen is the Logistic Regression model with scaled data.**

# Data Modeling

## Selected Model – Confusion Matrix

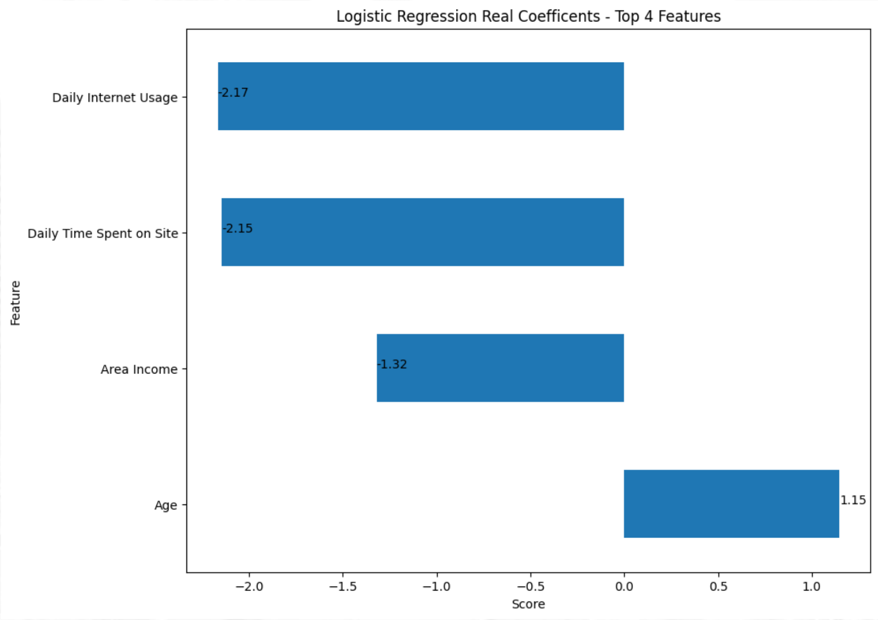


- From the test set confusion matrix above, from 144 people that clicked on an ad the algorithm correctly classified 140 of them and incorrectly classified 4 of them.

- Similarly, out of 154 people that did not click on an ad the algorithm correctly classified 150 of them and only incorrectly classified only 4.

# Data Modeling

## Selected Model – Feature Importance Score (Top 4)



- Feature importance of logistic regression model can simply be analyzed from its coefficient. The coefficients represent the change in the log odds for a one-unit change in the feature variable.

- The lower the Daily Internet Usage, Daily Time Spent on Site, and Area Income, the higher the chances that the customer will click on an ad and vice versa.

- The higher the Age, the bigger the odds that the customer will click on an ad and vice versa.