



# Weekly Assignment 4

by Fathurrahman Hernanda Khasan

**Scenario:**

You work as a data analytics engineer for a Ritz Jager Bank, analyzing customer behavior and predicting churn. The dataset provides information on existing credit card customers, including demographic data, financial behavior, and interaction history. Your goal is to build a predictive model to identify customers likely to churn (Attrition\_Flag = "Attrited Customer") based on their attributes and behavior. This model will help the bank retain valuable customers and reduce customer churn.

Answer the following questions

1. Create the data quality report of the dataset you received. What is the issue?  
How to solve the issue?
2. Solve the data quality issue on the dataset.
3. Create the exploratory data analysis of the dataset.
4. Explain 5 univariate and bivariate analyses that you found interesting.
5. Your company requests that you create a predictive analytics model based on the above problem. Demonstrate the process and show the model performance result

---

→Dokumentasi Pengerjaan

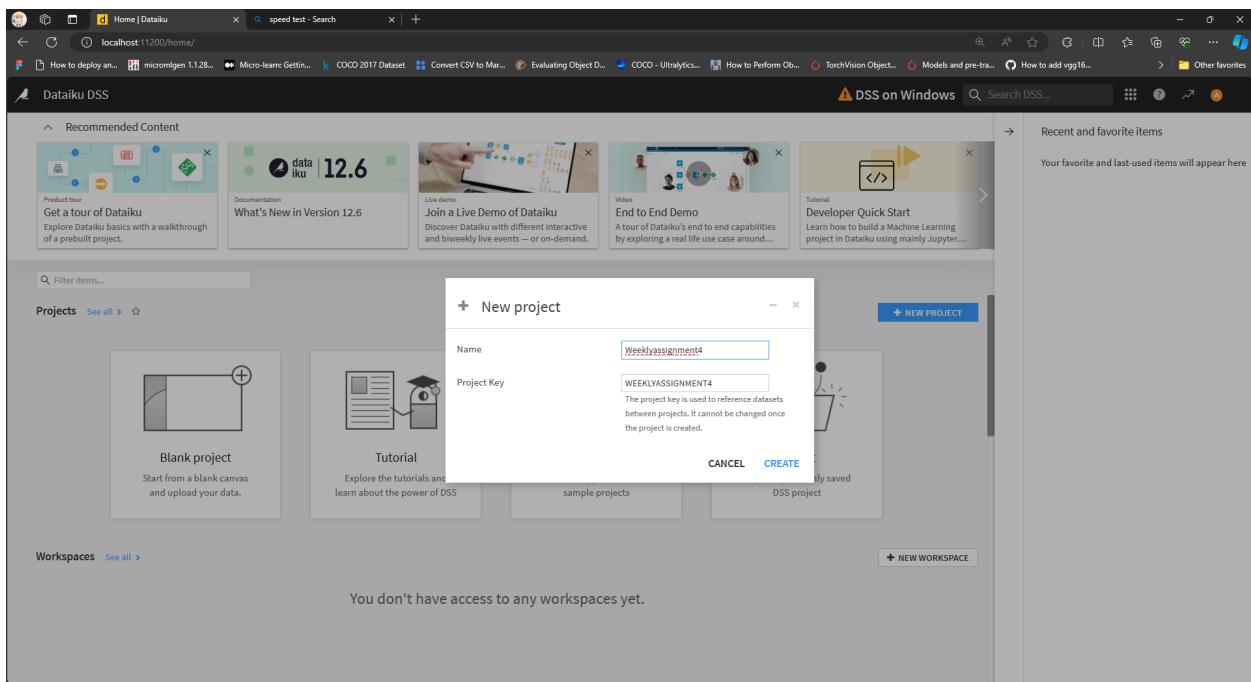
→Penjelasan Analytics

**Riset Data Raw →**

Dokumentasi Pengerjaan(Tools Dataiku):

Pertama-tama membuat project terlebih dahulu di dataiku:

---

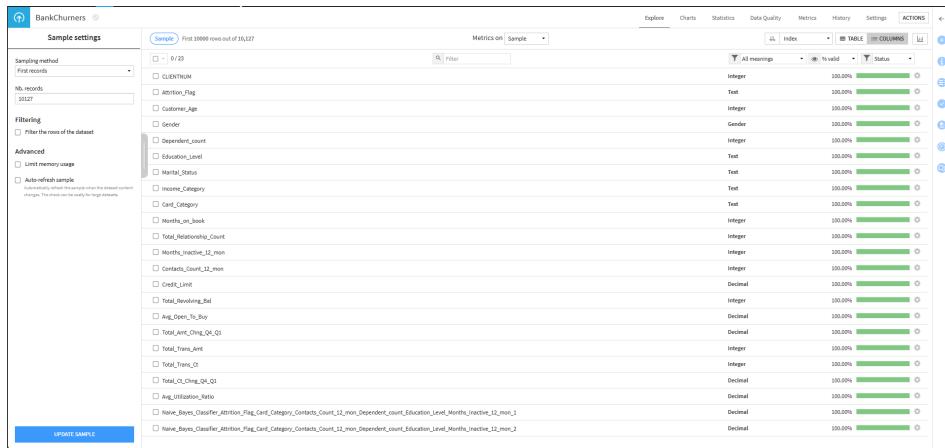


Setelah itu masukan datasetnya ke projek dataiku yg sebelumnya dibuat:

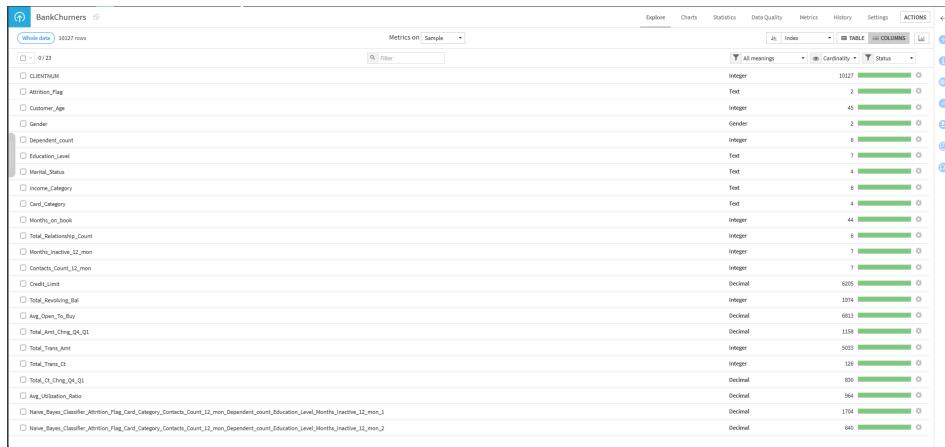
Lalu review jika sudah tekan create:

## Cek data quality Raw Data:

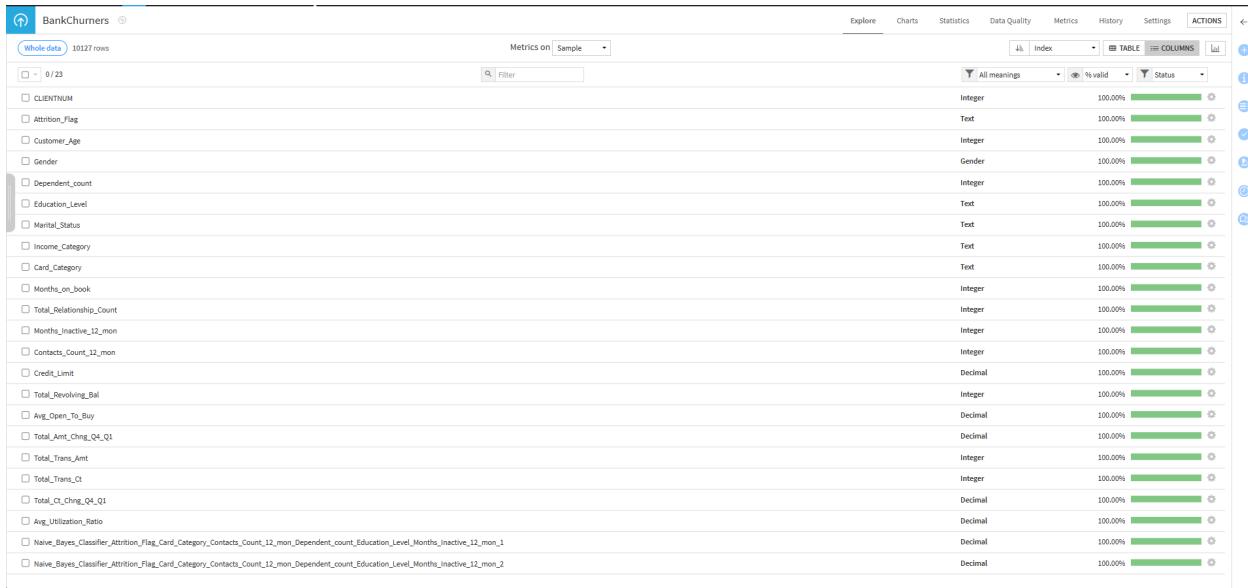
100% valid



## Cek cardinality:



## Cek validity:



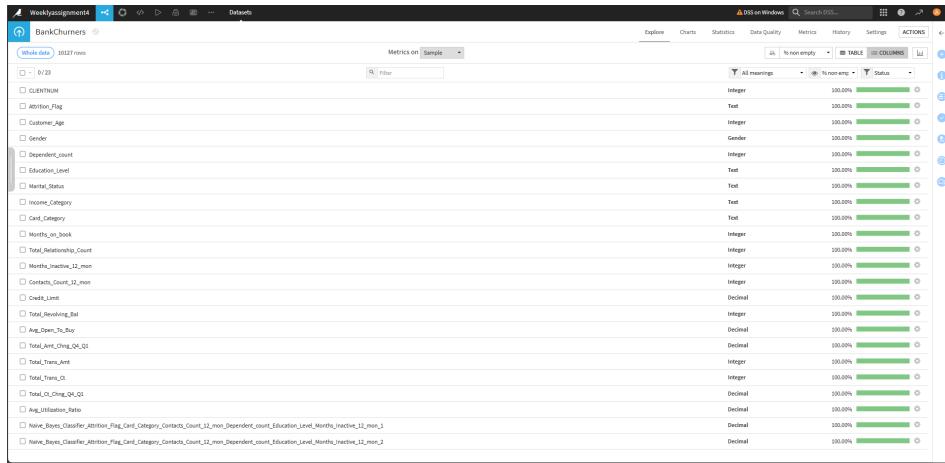
**Cek data yang invalid:**

**0%**

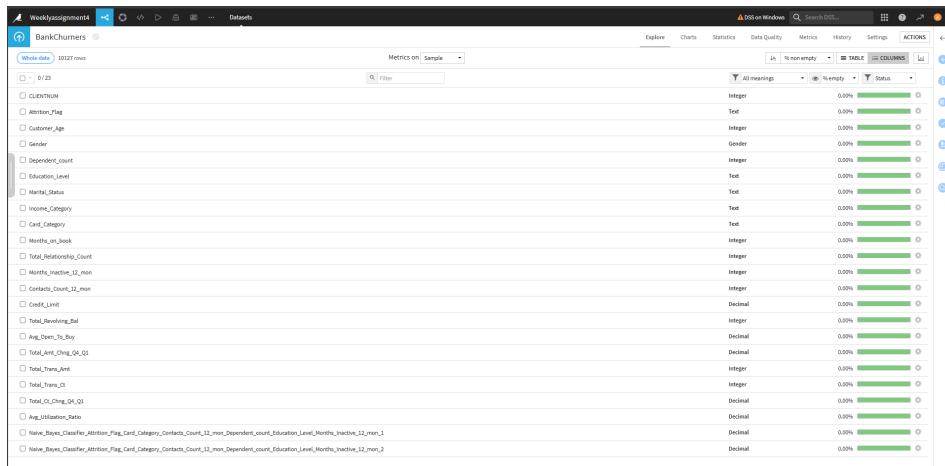


**Cek non empty:**

**100%**



### Cek empty:

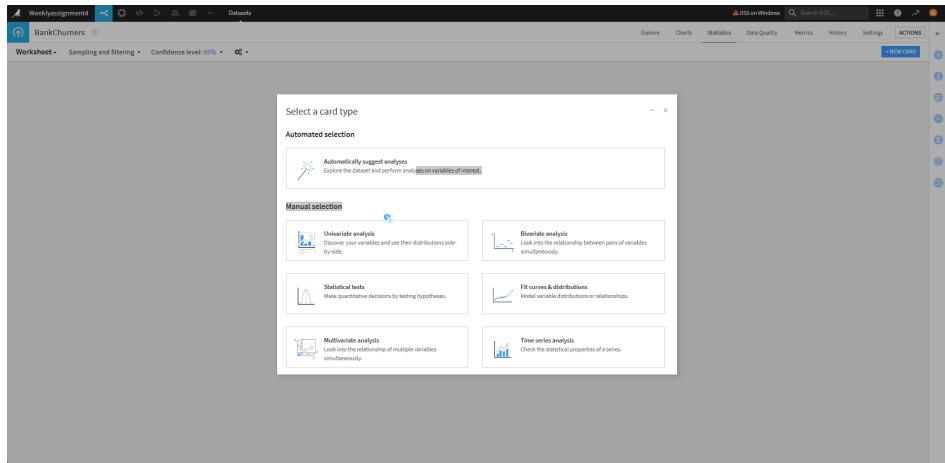


### Cek Data Type:

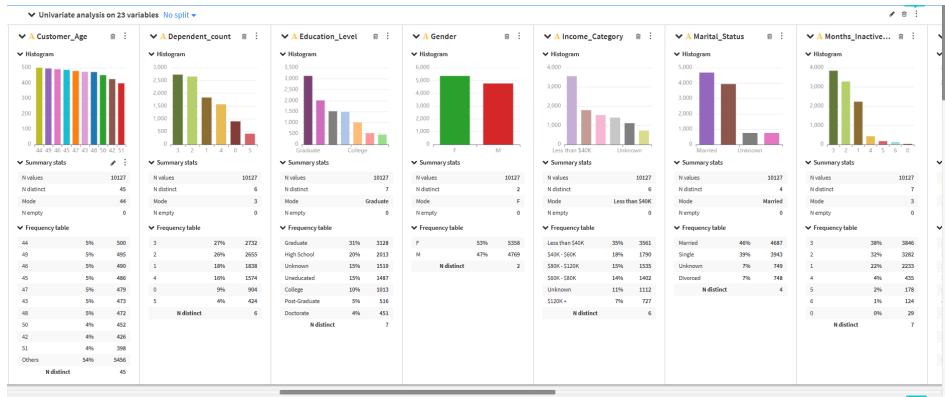
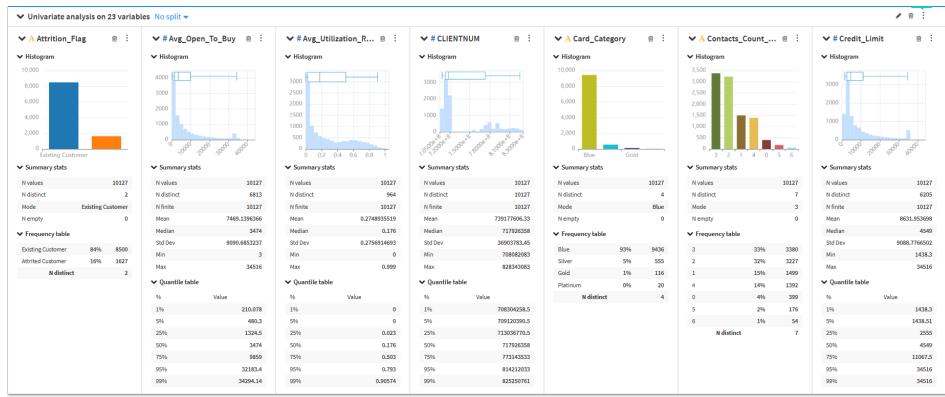


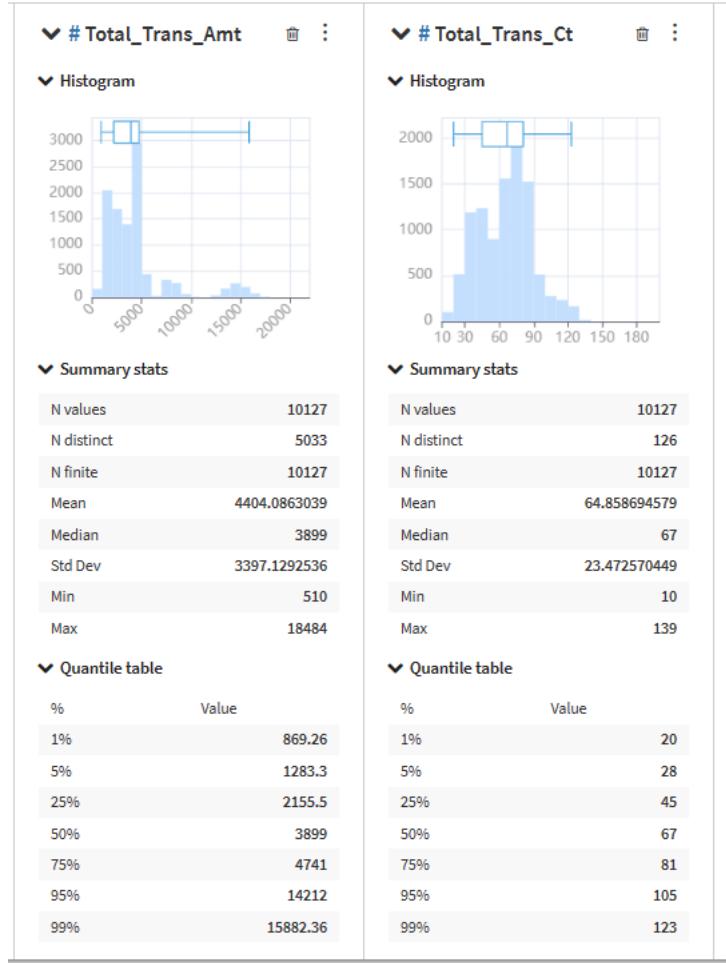
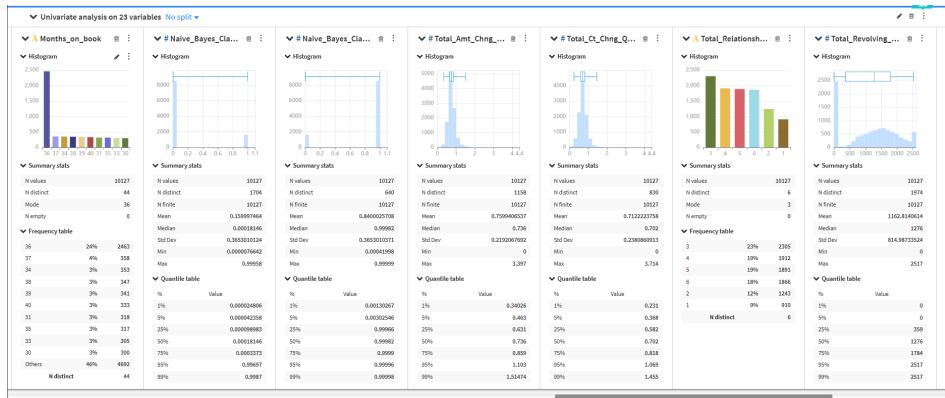
### Cek Distribusi dari Data:

make statistics:



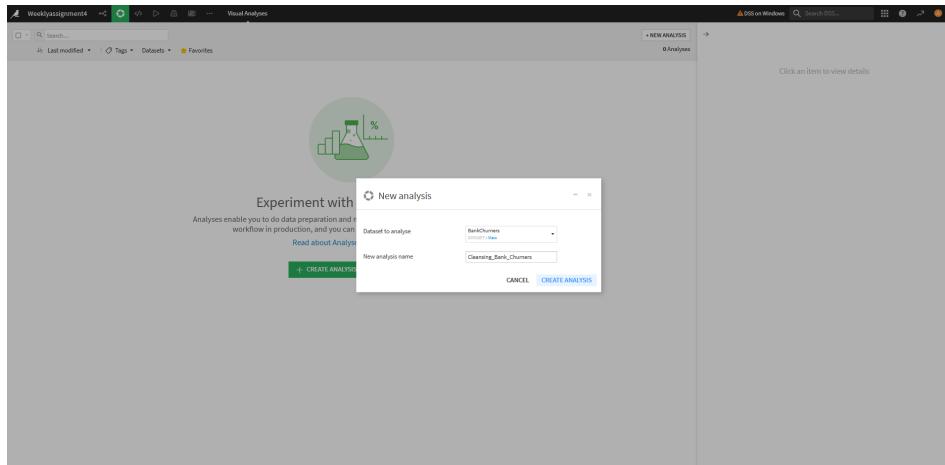
Beikut pengecekan keseluruhan variable dan distribusi datanya:





analyze 1 per 1:

Karena disini masih berfokus pada data rawnya maka jika ingin melakukan Analysis dan feature engineering tinggal menggunakan fungsi analysis namun saya akan menggunakan ini pada tahap selanjutnya:



Processors library

math

17 processors

<input type="checkbox"/> Filter data	4
<input type="checkbox"/> Data cleansing	1
<input type="checkbox"/> Strings	4
<input type="checkbox"/> Math / Numbers	16
<input type="checkbox"/> Split / Extract	1
<input type="checkbox"/> Dates	1
<input type="checkbox"/> Enrich	1
<input type="checkbox"/> Reshaping	1
<input type="checkbox"/> Natural Language	1
<input type="checkbox"/> Code	3

**Formula**

This processor computes new columns using formulas based on other columns (like in a spreadsheet).

The formula language provides:

- Math functions
- String manipulation functions
- Date handling functions
- Boolean and conditional expressions for rules creation

**Usage examples**

- Compute a numerical expression: `(col1 + col2) / 2 * log(col3)`
- Manipulate strings: `tolowercase(substring(strip(MyText), 0, 7))`
- Create a rule-based column: `if (width > height, "wide", "tall")`

**Getting help**

Machine learning development untuk riset machine learning di tahap selanjutnya dengan menggunakan data raw terlebih dahulu:

Weeklyassignment4

ModelsBankChurners\_raw

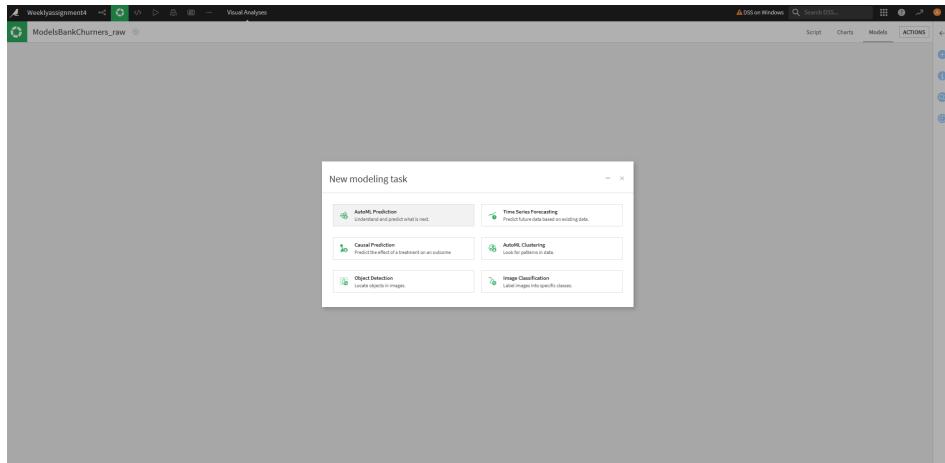
Start modeling from your data

Train prediction and clustering models

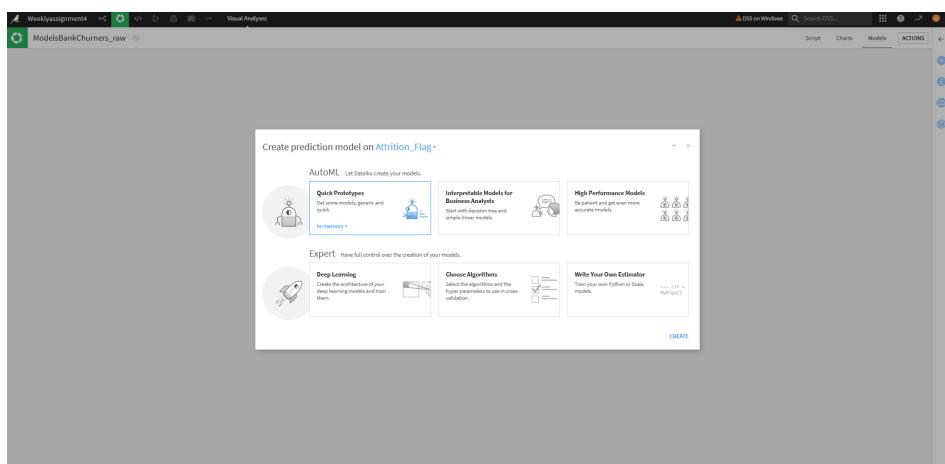
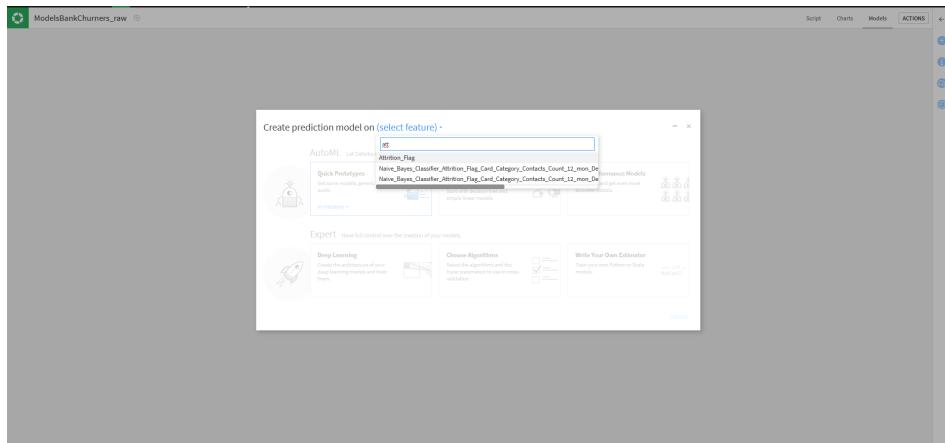
**ACTIONS**

**CREATE FIRST MODEL**

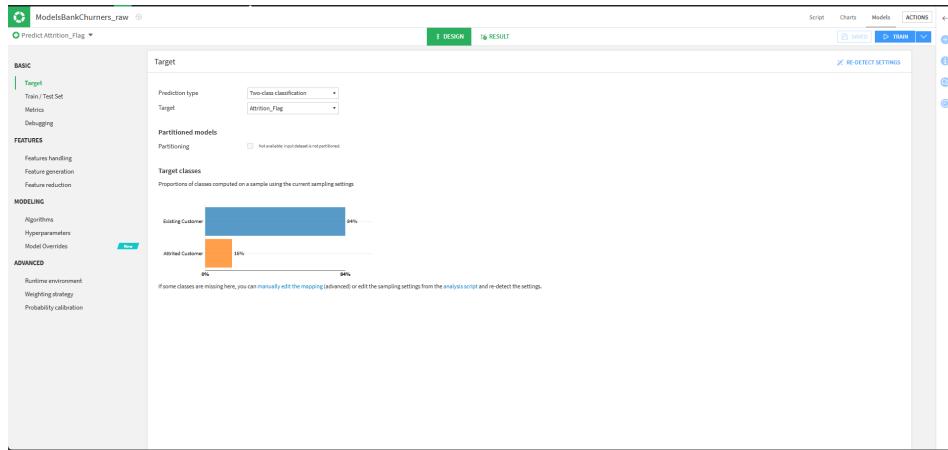
Menggunakan auto ML:



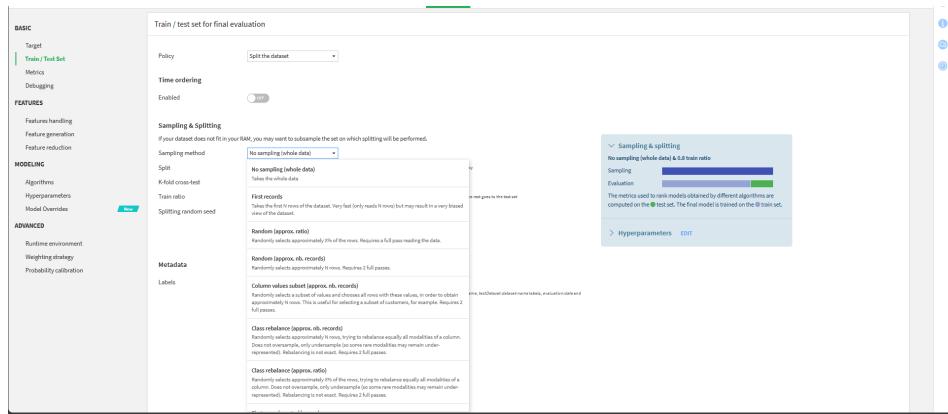
Memilih label dari machine learningnya yakni kolom **Attrition\_Flag**



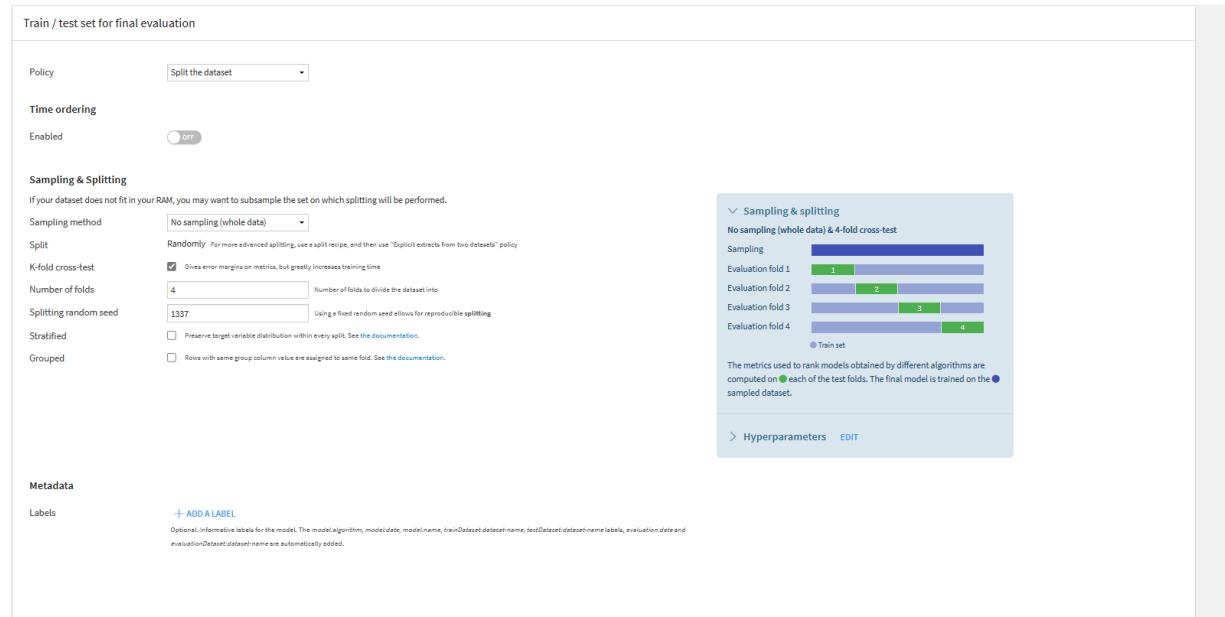
lalu design



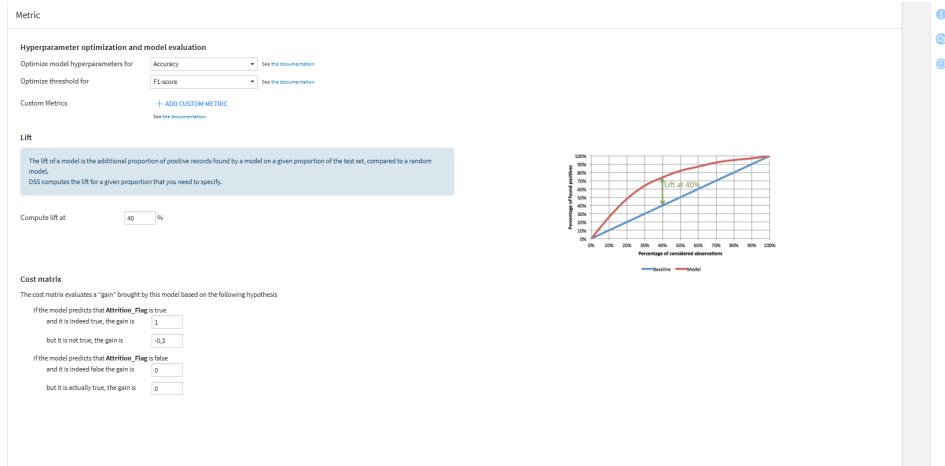
Gunakan whole datanya:



Hyperparameter tuning menggunakan k-fold = 4



matric evaluasinya adalah Accuracy:



Lalu melakukan Feature Handling dan pilih hanya kolom kolok fitur yang relevan saja:

The screenshot shows the 'Features Handling' section in a machine learning tool. It lists various features with checkboxes and status indicators (ON/OFF) for each. The features include:

- Dataset
- CLIENTNUM
- Attrition\_Flag (Target variable)
- Customer\_Age
- Gender
- Dependent\_Count
- Education\_Level
- Marital\_Status
- Income\_Category
- Card\_Category
- Months\_on\_book
- Total\_Relationship\_Count
- Months\_Inactive\_12\_mon
- Contacts\_Count\_12\_mon
- Credit\_Limit
- Total\_Revolving\_Bal
- Avg\_Open\_To\_Buy

Lalu tahap Modeling:

**XGBoost**  
 XGBoost is an advanced gradient boosting algorithm. It has support for parallel processing, regularization and early stopping, which makes it a fast, scalable and accurate algorithm....  
 Show more...

**Booster**  
 Try Gradient Boosted Trees  
 Try DART

DART is a variant of the "Gradient Boosted Tree" estimator where, at each step of the training phase, previous trees are randomly dropped out.

**Tree method**  
 Automatic - CPU only  
 Automatic modes are only available for CPU execution.  
 Automatic will choose one of the other method based on heuristic and shape of the data.  
 DART is a greedy tree-building algorithm.  
 Approximate is a fast tree-building algorithm.  
 Histogram is an optimized fast tree-building algorithm.

**Maximum number of trees**  
 300  
 XGBoost has an early stop mechanism as the exact number of trees will be optimized. High number of actual trees will increase the training and prediction time. Typical values: 100-10000.

**Early stopping**  
  
 Use 10000 built-in early stop mechanism as the exact number of trees will be optimized.  
 The cross-validation scheme defined in the "Hyperparameters" tab will be used.

**Early stopping rounds**  
 4  
 The optimizer stops if the loss never decreases for this consecutive number of iterations. Typical values: 1-100.

**Maximum depth of tree**  
 3  
 Maximum depth of each tree. High values can increase the quality of the prediction, but can lead to overfitting. Typical values: 2-10.

**Learning rate**  
 0.2  
 Lower values slow down convergence and can make the model more robust. Typical values: 0.01-0.3.

**Max delta step**  
 0.5  
 Lower positive values can make the update step more conservative which might help in very imbalanced classifications. 0 means unlimited (default). 0.7 is recommended for Poisson regression.

**L2 regularization**  
 1e-05  
 Lower positive values can make the update step more conservative which might help in very imbalanced classifications. 0 means unlimited (default). 0.7 is recommended for Poisson regression.

**L1 regularization**  
 0.0001  
 Lower positive values can make the update step more conservative which might help in very imbalanced classifications. 0 means unlimited (default). 0.7 is recommended for Poisson regression.

Lalu training model machine learning dan perhatikan seperti apa hasilnya untuk riset kedepannya:

**Previously trained**

- Random forest (vt\_row) 0.953 ( $\pm 0.003$ )
- LightGBM (vt\_row) 0.969 ( $\pm 0.003$ )
- XGBoost (vt\_row) 0.911 ( $\pm 0.008$ )
- Logistic Regression (vt\_row) 0.888 ( $\pm 0.011$ )

**Random forest (vt\_row)** 0.953 ( $\pm 0.003$ ) ✓ Done just now (2024-05-04 19:28:31)

Most important features

Rank	Feature	Value
100	Total_Accruing_Bal	0.0001
14	Total_Trans_Amt	0.0001
1	Total_Trans_Ct	0.0001
2	Total_Ct_Chng_06_08	0.0001
	Total_Relationship_Count	0.0001
	Months_Inactive_12_Mon	0.0001

Train set Train time 10127 rows about 32 seconds

**Logistic Regression (vt\_row)** 0.888 ( $\pm 0.011$ ) ✓ Done just now (2024-05-04 19:28:11)

Most important features

Rank	Feature	Value
1	Total_Trans_Ct	0.0001
12	Total_Trans_Amt	0.0001
5	Total_Revolving_Bal	0.0001
	Total_Relationship_Count	0.0001
	Contacts_Count_12_mon	0.0001
	Total_Ct_Chng_04_01	0.0001

Train set Train time 10127 rows about 12 seconds

**LightGBM (vt\_row)** 0.969 ( $\pm 0.003$ ) ✓ Done just now (2024-05-04 19:28:03)

Most important features

Rank	Feature	Value
100	Total_Trans_Ct	0.0001
14	Total_Trans_Amt	0.0001
1	Total_Revolving_Bal	0.0001
2	Total_Ct_Chng_06_08	0.0001
	Total_Relationship_Count	0.0001

Train set Train time 10127 rows about 32 seconds

Model:

**Random forest (vt\_row)** 0.953 ( $\pm 0.003$ ) ✓ Done 1 minute ago (2024-05-04 19:28:31)

Most important features

Rank	Feature	Value
100	Total_Accruing_Bal	0.0001
14	Total_Trans_Amt	0.0001
1	Total_Trans_Ct	0.0001
2	Total_Ct_Chng_06_08	0.0001
	Total_Relationship_Count	0.0001
	Months_Inactive_12_Mon	0.0001

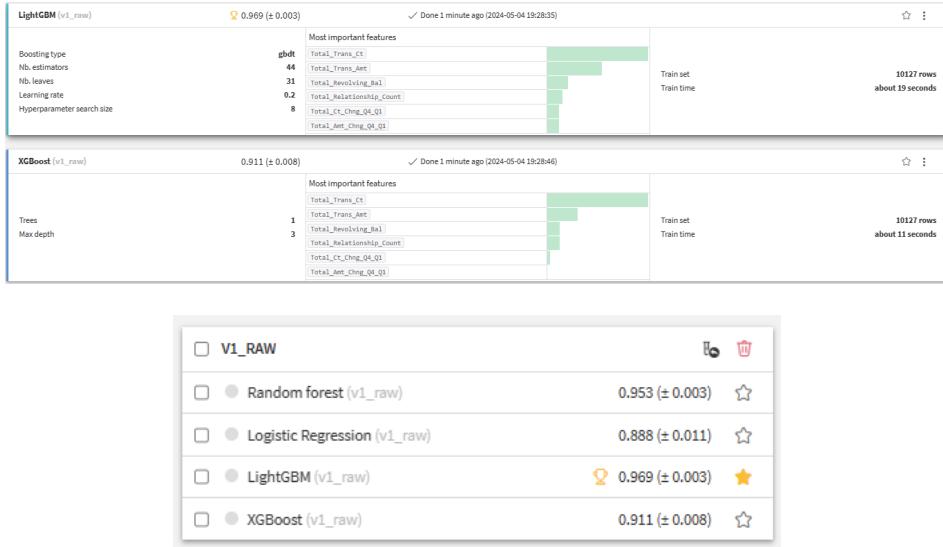
Train set Train time 10127 rows about 32 seconds

**Logistic Regression (vt\_row)** 0.888 ( $\pm 0.011$ ) ✓ Done 1 minute ago (2024-05-04 19:28:11)

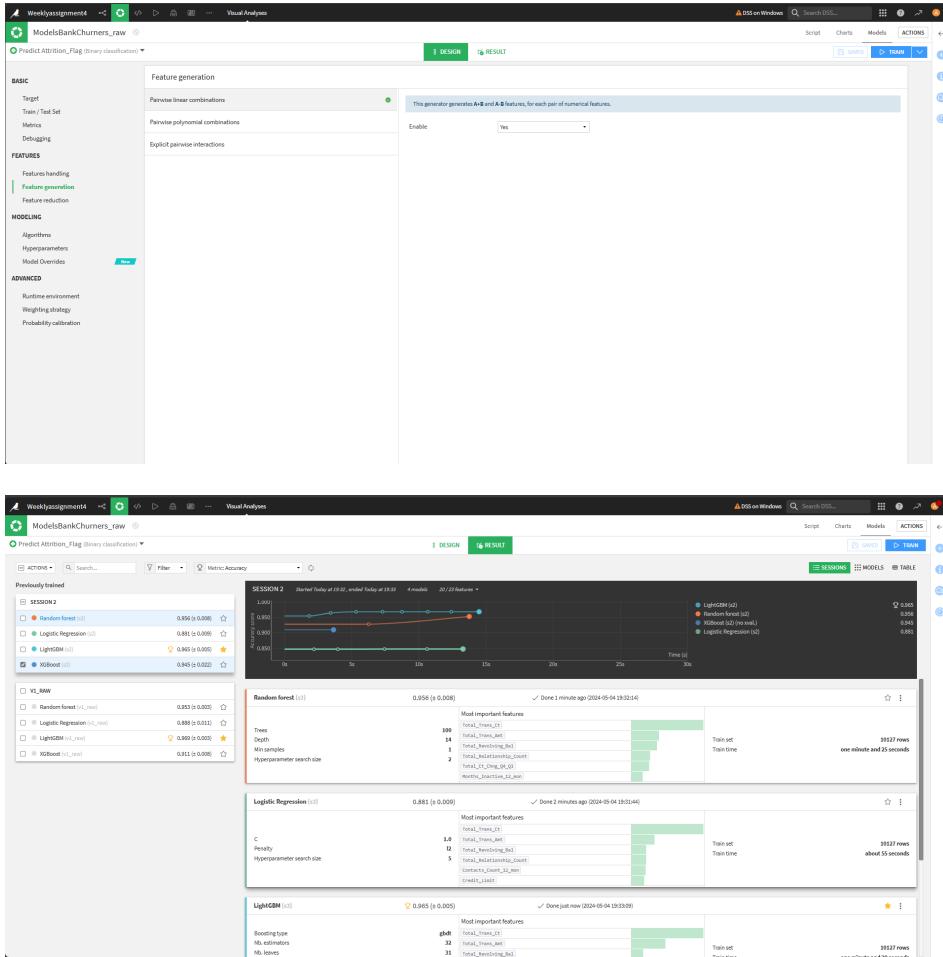
Most important features

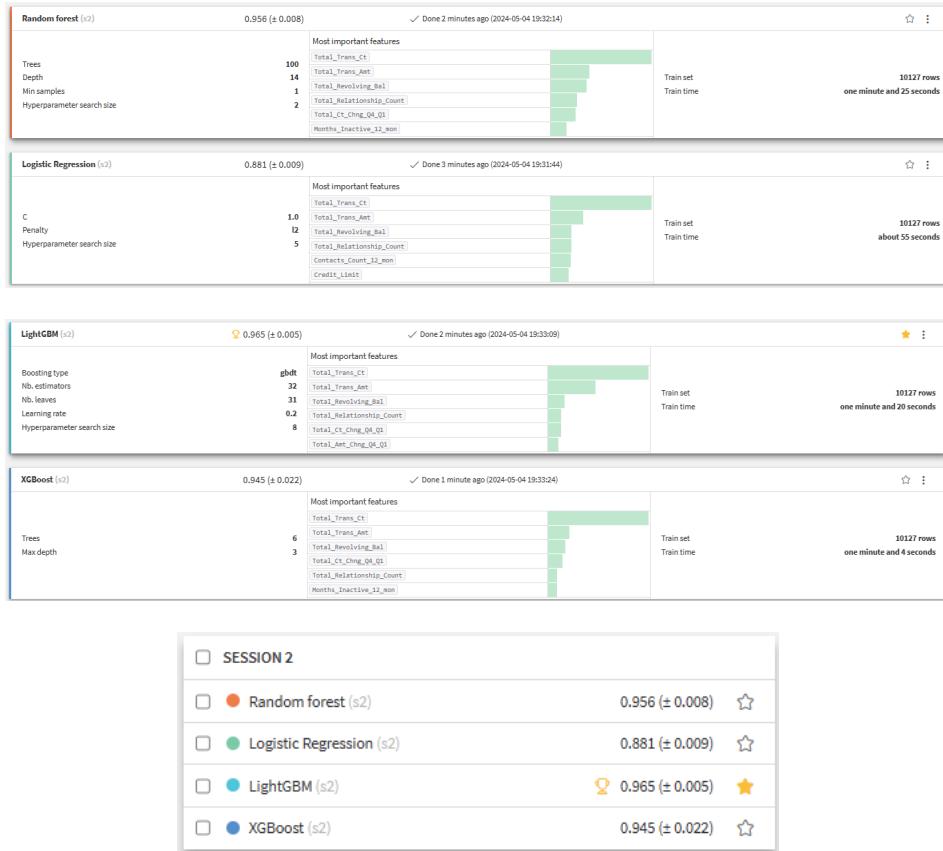
Rank	Feature	Value
1	Total_Trans_Ct	0.0001
12	Total_Trans_Amt	0.0001
5	Total_Revolving_Bal	0.0001
	Total_Relationship_Count	0.0001
	Contacts_Count_12_mon	0.0001
	Total_Ct_Chng_04_01	0.0001

Train set Train time 10127 rows about 12 seconds

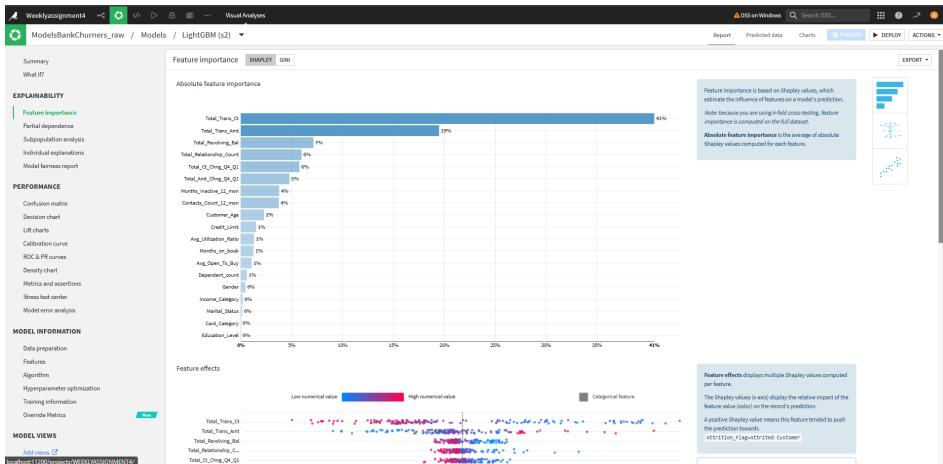


Menggunakan auto feature engineering:

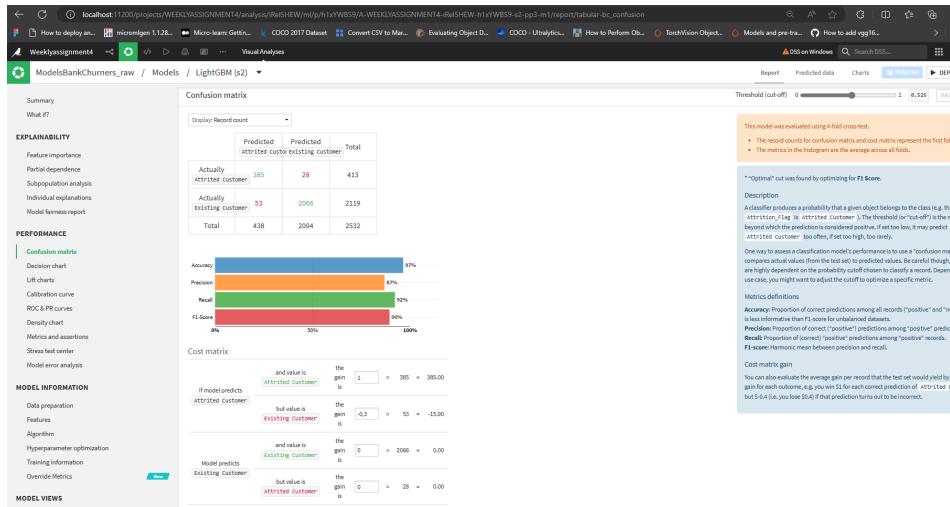




Berikut feature importancenya:



Hasil training dan confussion matrix

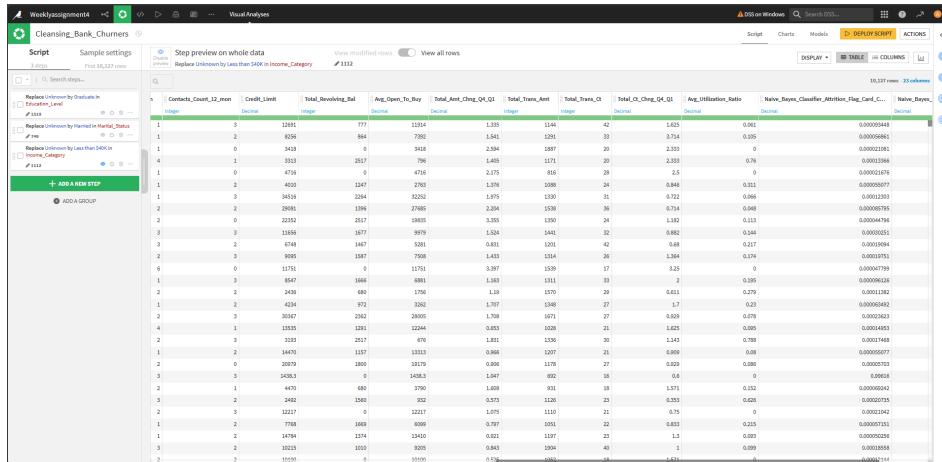


Dari hasil ditas terlihat hanya menggunakan data raw saja dengan minim data preparation mendapatkan hasil yang lumayan oke namun disini kita harus menekan false positive dan false negatifnya di model selanjutnya.

## Riset Analytics →

### Analytics With Treatment:

Menambahkan script untuk mengganti nilai unknown dengan modus di dari kolom Educationa\_level, marital\_status, income\_Categotry :



Lalu deploy Scriptnya:

Script: compute\_bankChurners\_prepared

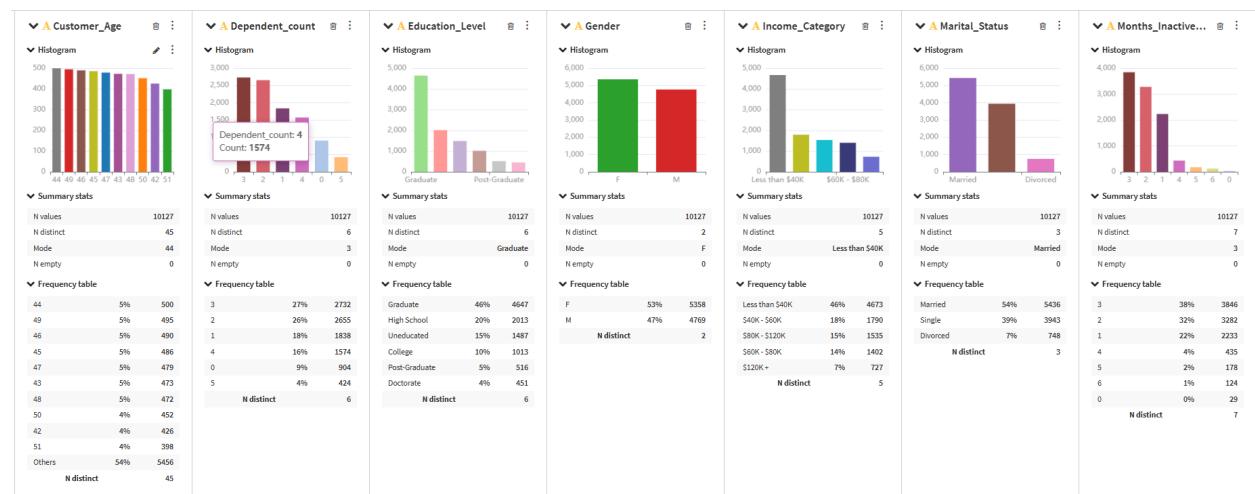
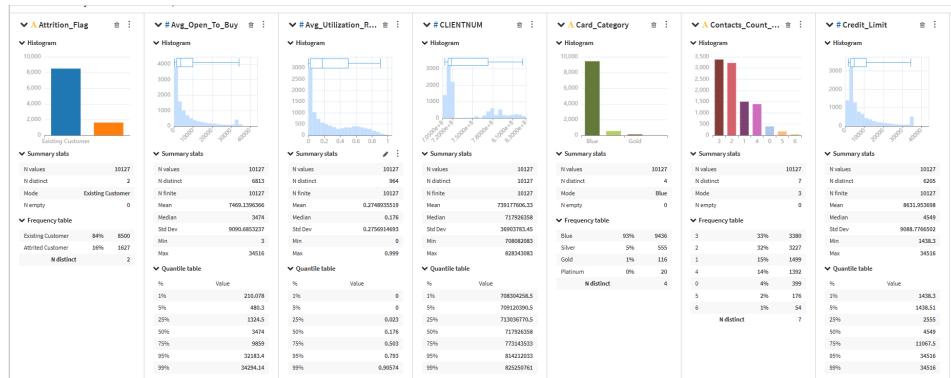
Sample settings: 3 rows

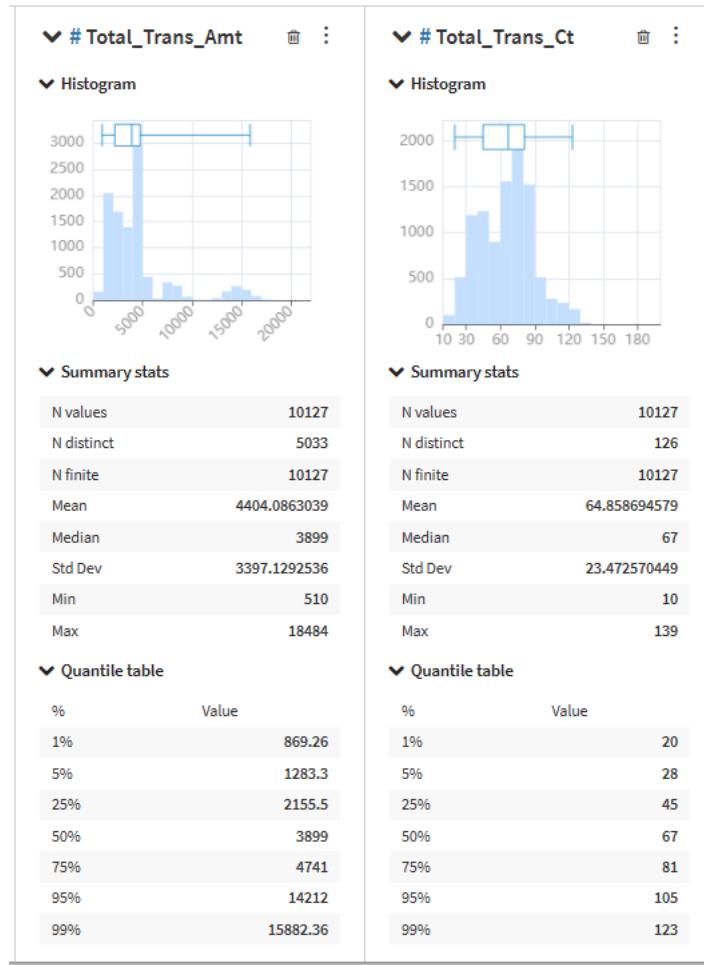
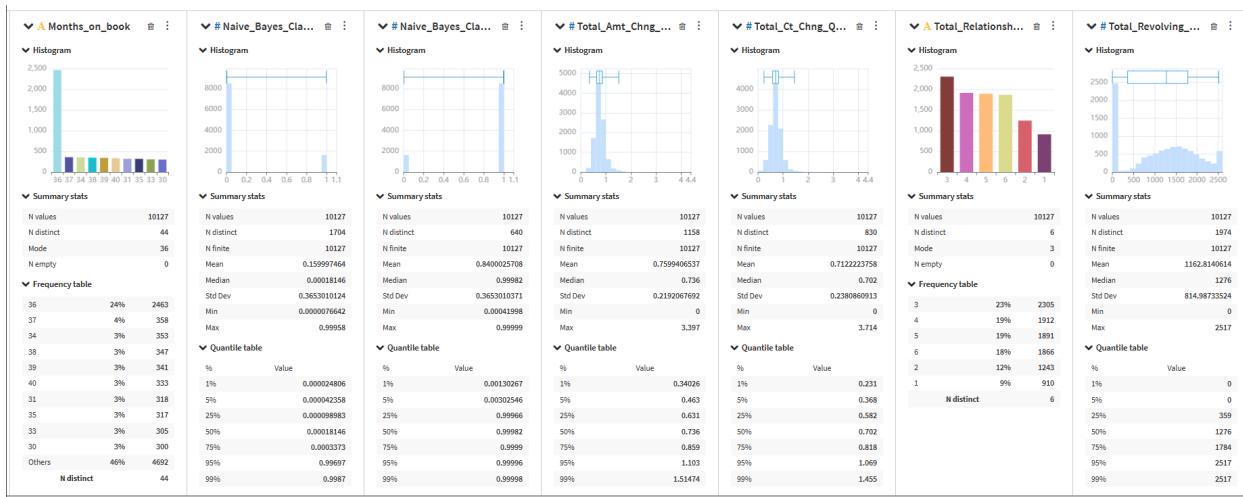
Script output on: Whole data (10,337 rows)

CLIENTNUM Attrition\_Flag Customer\_Age Gender Dependent\_count Education\_Level Marital\_Status Income\_Catagory Card\_Catagory Months\_on\_Bal Total\_Relationship\_Count Monthly\_Inactive\_12\_mos Credit\_Card\_Count\_12\_mos

Job succeeded. Explore dataset bankChurners\_prepared

Didapatkan distribusi datanya setelah script di deploy sebagai berikut:

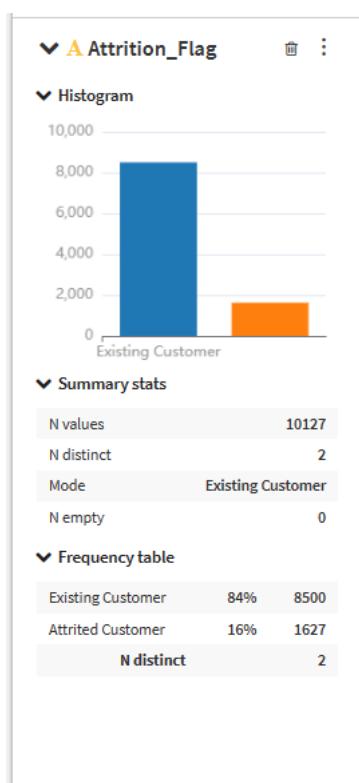




Sekarang saatnya melakukan Univariate Analysis dan bivariate analyses pada fitur fitur yang menarik:

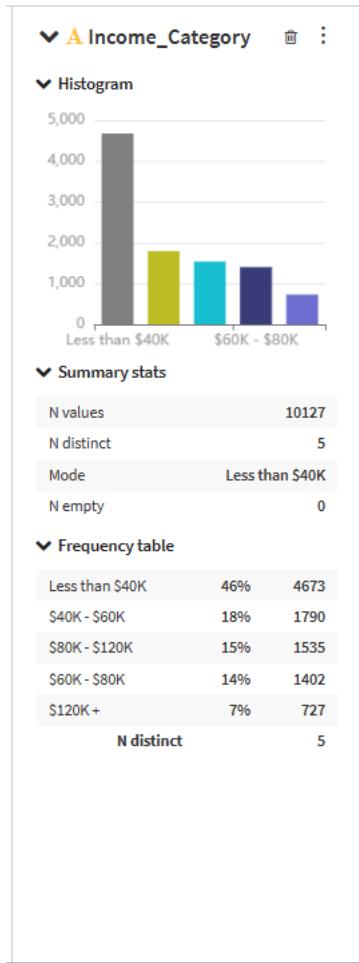
- Attrition\_Flag adalah sebuah indikator atau tanda dalam dataset yang menunjukkan apakah seorang pelanggan telah meninggalkan atau berpindah dari suatu layanan atau produk. "Attrition\_Flag" mungkin menandakan apakah seorang pelanggan telah berpindah dari bank atau tetap sebagai pelanggan yang ada. Biasanya, nilai-nilai yang mungkin dari "Attrition\_Flag" adalah "Attrited Customer" untuk pelanggan yang berpindah dan "Existing Customer" untuk pelanggan yang tetap. Namun pada data ini terdapat imbalance data dimana dalam data ini Existing Customer memiliki jumlah 84% dari total

keseluruan data dan hanya 16% pada kelas Attrited Customer dimana ini akan sedikit berpengaruh pada beberapa algoritma Machine Learning jika saat training berisiko terjadinya overfitting. Maka dari itu saat training nanti disarankan untuk melakukan under sampling atau menggunakan metode resampling lain seperti Metode seperti **SMOTE (Synthetic Minority Over-sampling Technique)** dapat digunakan untuk membuat contoh sintetis dari kelas minoritas, sehingga menciptakan keseimbangan dalam data. Serta menggunakan model machine learning dan matrix evaluasi yang sesuai keadaan sekarang ini.



2. Label atau kolom "Income\_Category" dalam dataset mengacu pada informasi tentang kategori pendapatan pelanggan. Ini adalah variabel yang menunjukkan kategori atau rentang pendapatan dari setiap pelanggan dalam dataset. Misalnya, kategori pendapatan dapat dibagi menjadi beberapa rentang seperti "Less than \$40K", "\$40K - \$60K", "\$60K - \$80K", dan seterusnya, atau bisa saja menggunakan kategori lain sesuai dengan kebutuhan bisnis atau penelitian. Variabel ini membantu dalam pemahaman demografi pelanggan dan dapat digunakan untuk menganalisis perilaku belanja, kebiasaan pengeluaran, dan preferensi produk berdasarkan tingkat pendapatan.

Terlahat disana jumlah less pengguna dengan income category less than 40 dolar memiliki jumlah yang jomplang dan lebih banyak dibanding yang lain.

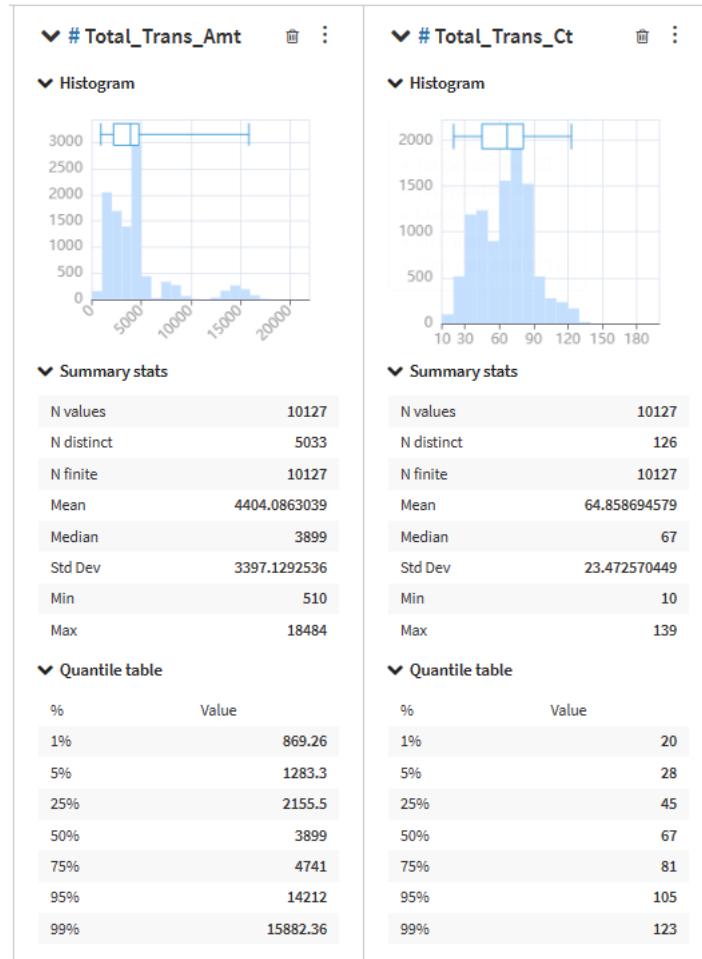


### 3.Total\_Trans\_Amt and Total\_Trans\_Ct

"**Total\_Trans\_Amt**" adalah total jumlah transaksi yang dilakukan oleh pelanggan dalam periode waktu tertentu. Ini mencakup total nilai dari semua transaksi yang dilakukan, termasuk pembelian barang atau layanan, pembayaran tagihan, transfer uang, dan aktivitas transaksi lainnya yang terkait dengan kartu kredit atau produk keuangan lainnya.

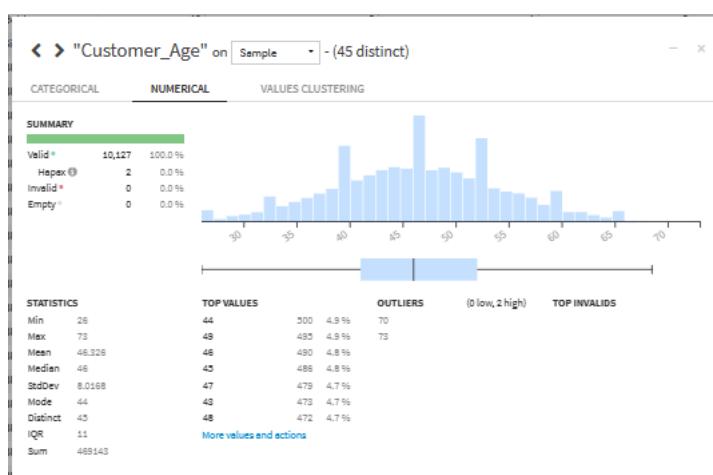
Sementara "**Total\_Trans\_Ct**" mewakili total jumlah transaksi yang dilakukan oleh pelanggan dalam periode waktu yang sama. Ini menghitung jumlah keseluruhan transaksi, tanpa memperhitungkan nilai atau jumlah uang yang terlibat. Jadi, ini mencakup setiap kali kartu kredit digunakan, tanpa memperhatikan seberapa besar atau kecilnya transaksi tersebut.

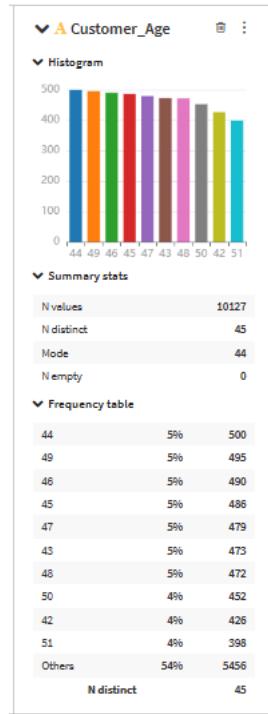
Kedua kolom ini memberikan informasi yang berharga tentang aktivitas penggunaan kartu kredit oleh pelanggan, yang dapat digunakan untuk menganalisis kebiasaan belanja, tingkat keterlibatan, dan potensi kecenderungan churn yang terlihat pada model sebelumnya. 2 variabel ini merupakan feature importance.



#### 4.Customer Age

Untuk melihat berapa minimum dan maximum umur daricustomer  
dan terlihat bahwa range umur customer adalah dari 26 hingga 73 atuhan.

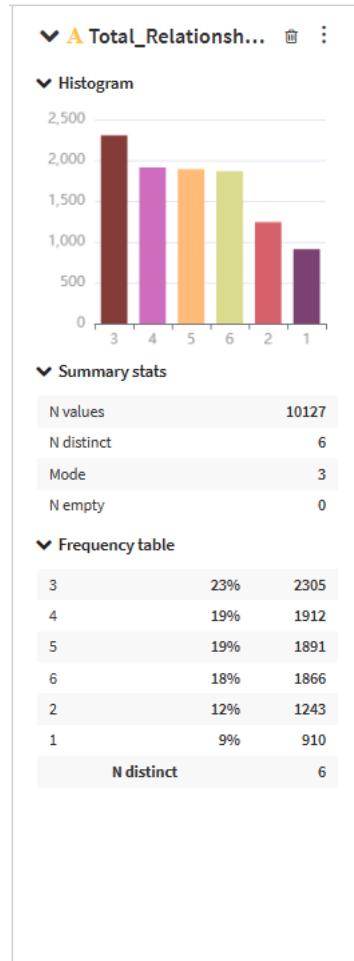




##### 5.Total\_relationship\_count:

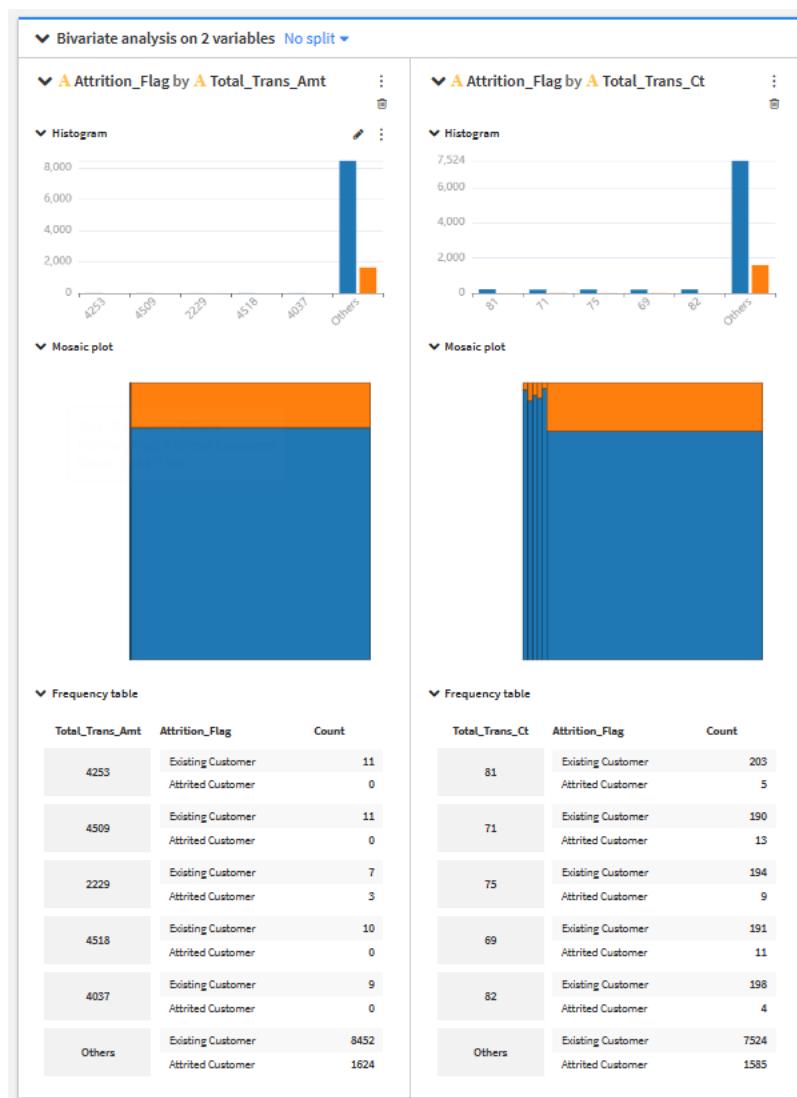
Kolom "Total\_relationship\_count" mengacu pada total jumlah produk atau layanan keuangan yang dimiliki oleh seorang pelanggan di bank atau lembaga keuangan tertentu. Ini mencakup berbagai jenis akun, seperti tabungan, deposito, kartu kredit, pinjaman, dan produk keuangan lainnya yang dimiliki oleh pelanggan.

Kolom ini penting karena mencerminkan seberapa besar hubungan pelanggan dengan bank atau lembaga keuangan tersebut. Semakin banyak produk atau layanan yang dimiliki oleh pelanggan, semakin dalam hubungan mereka dengan bank, dan kemungkinan mereka untuk tetap sebagai pelanggan yang setia meningkat. Analisis kolom ini dapat memberikan wawasan tentang seberapa baik bank dapat mempertahankan dan memperluas hubungan dengan pelanggan, serta potensi risiko churn jika pelanggan mulai mengurangi jumlah produk atau layanan yang mereka miliki.



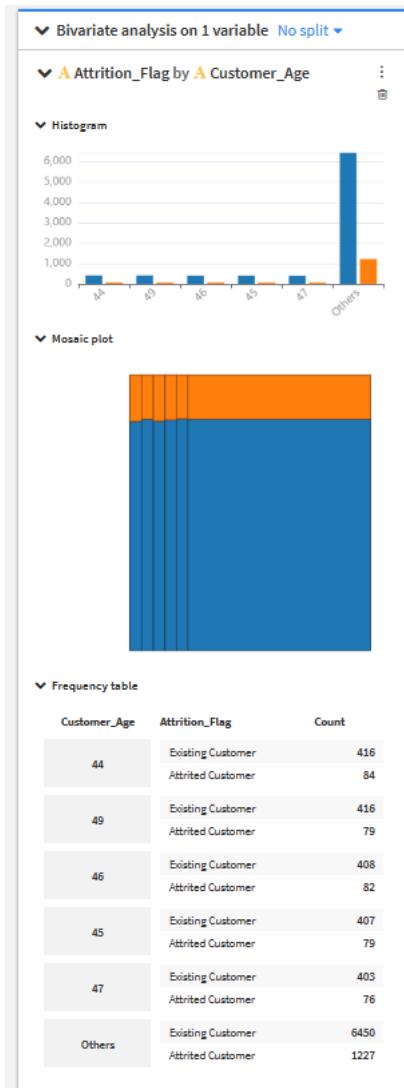
#### 6. Jumlah Transaksi Total dengan Churn:

Menginvestigasi apakah ada korelasi antara jumlah total transaksi yang dilakukan oleh pelanggan dengan kecenderungan churn. Apakah pelanggan yang lebih aktif dalam menggunakan kartu kredit mereka cenderung tetap atau berpindah?



## 7. Hubungan Umur dengan Churn (Attrition):

Mengidentifikasi apakah ada hubungan antara umur pelanggan dan kecenderungan churn (pindah) dari bank. Apakah pelanggan muda lebih cenderung untuk berpindah dibandingkan dengan pelanggan yang lebih tua?



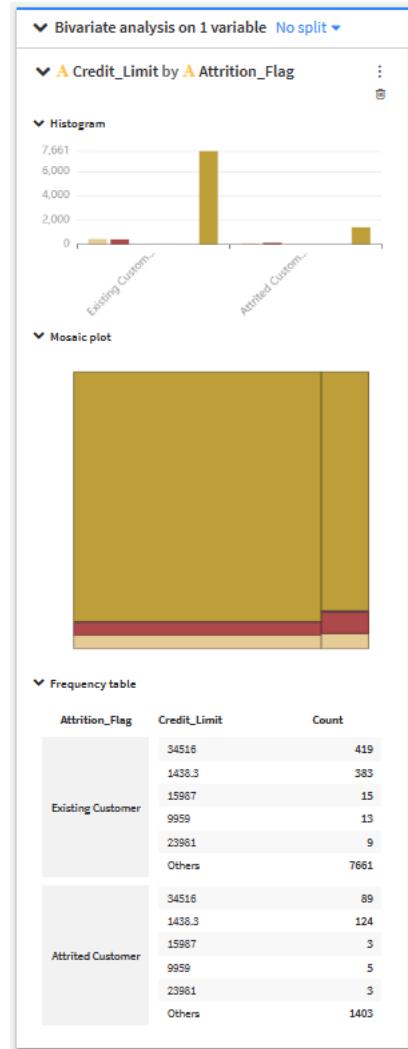
#### 8. Hubungan Limit Kredit dengan Penggunaan Kartu (Card Utilization Ratio):

Melihat apakah ada korelasi antara limit kredit yang disediakan kepada pelanggan dengan seberapa sering mereka menggunakan kartu kredit mereka. Apakah pelanggan dengan limit kredit yang lebih tinggi cenderung menggunakan kartu mereka lebih banyak?



#### 9.Distribusi Limit Kredit antara Pelanggan yang Berpindah dan yang Tetap:

Membandingkan distribusi limit kredit antara pelanggan yang berpindah dan yang tetap dapat memberikan wawasan apakah ada perbedaan dalam profil kredit antara kedua kelompok.



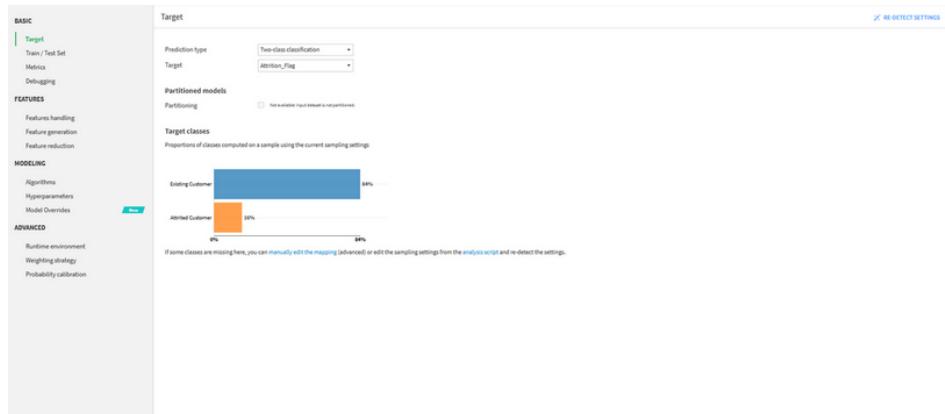
Corelation matrix:



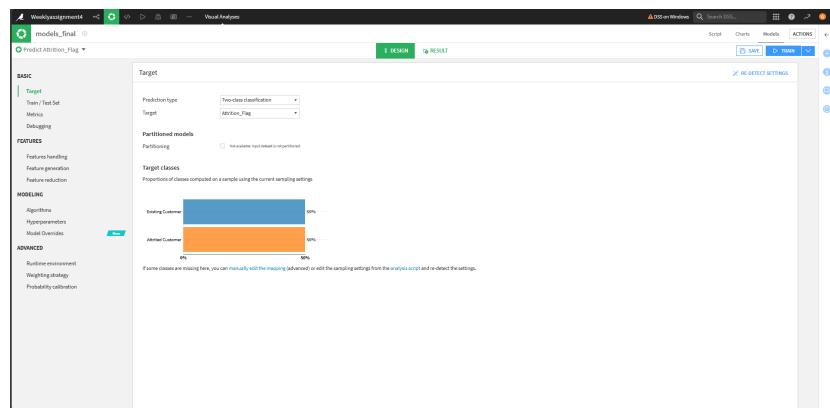
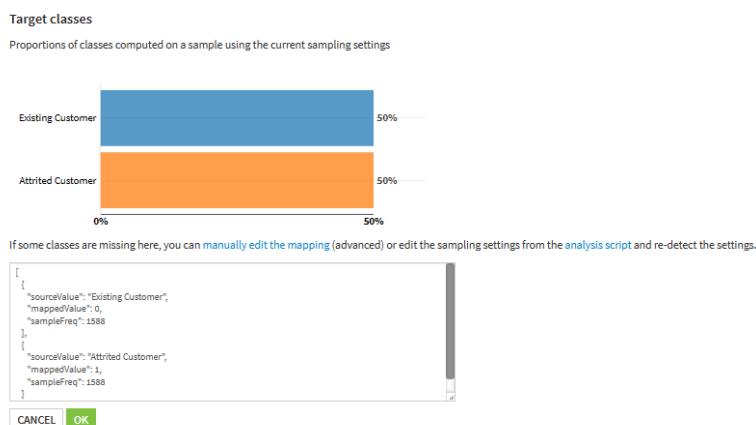
### Modeling final:

Cara 1: Resampling atau under sampling ini jika datanya imbalance seperti pada kasus ini dan buat target proportionnya 50:50

Sebelum:



Sesudah:



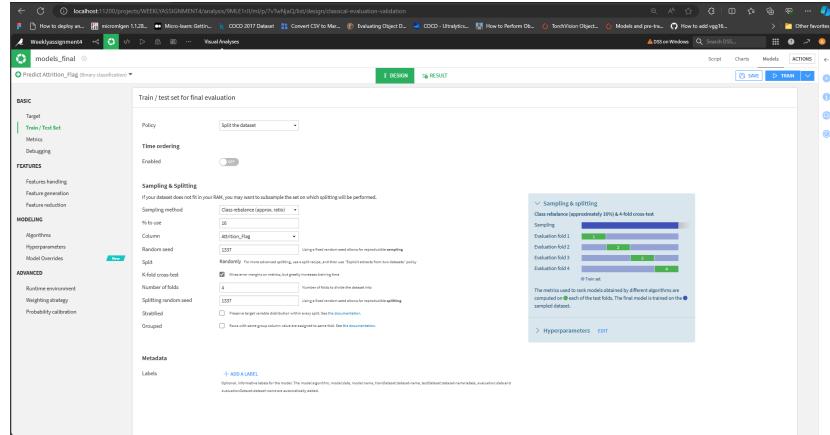
Cara 2:

**Metode Resampling:** Metode seperti SMOTE (Synthetic Minority Over-sampling Technique) dapat digunakan untuk membuat contoh sintetis dari kelas minoritas, sehingga menciptakan keseimbangan dalam data.

Disini saya menggunakan fitur class rebalanceClass rebalance (approx. ratio) yang disediakan oleh dataiku:

Metode ini memungkinkan untuk mencoba mencapai keseimbangan antara kelas tanpa menghasilkan contoh sintetis (oversampling)

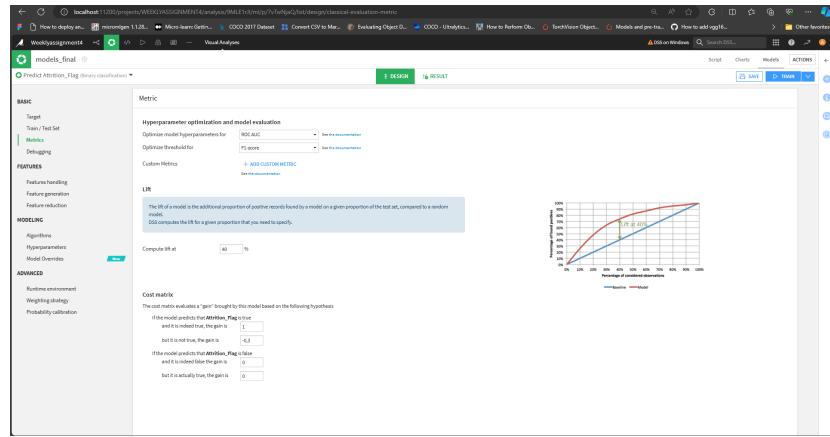
Alasan untuk ini adalah karena data yang dimiliki sangat tidak seimbang, dan menggunakan persentase dari total data (dalam hal ini X%) akan memberi lebih banyak fleksibilitas dalam menentukan seberapa banyak data yang ingin dipilih untuk kelas minoritas. Dengan menggunakan "approx. ratio", dapat mencoba memilih persentase data yang lebih besar dari kelas minoritas untuk menyeimbangkan kelas tanpa harus menghasilkan contoh sintetis. Disini saya menggunakan 18% ration untuk datanya agar balance erikut settingannya:



karena kelasnya imbalance jadi alangkah baiknya disini saya pilih:

ROC AUC:

- ROC AUC adalah metrik yang baik untuk mengevaluasi kinerja model pada data yang tidak seimbang.
- ROC AUC mengukur kemampuan model untuk membedakan antara kelas positif dan negatif.
- Ini mengabaikan ambang batas klasifikasi dan fokus pada urutan peringkat prediksi.



disini saya menonaktifkan beberapa fitur yang tidak berpengaruh untuk proses training:

yaitu:

- 1."Naive\_Bayes\_Classifier\_Attrition\_Flag\_Card\_Category\_Contacts\_Count\_12\_mon\_Dependent\_count\_Education\_Level\_Month: dan
- 2."Naive\_Bayes\_Classifier\_Attrition\_Flag\_Card\_Category\_Contacts\_Count\_12\_mon\_Dependent\_count\_Education\_Level\_Month karena:

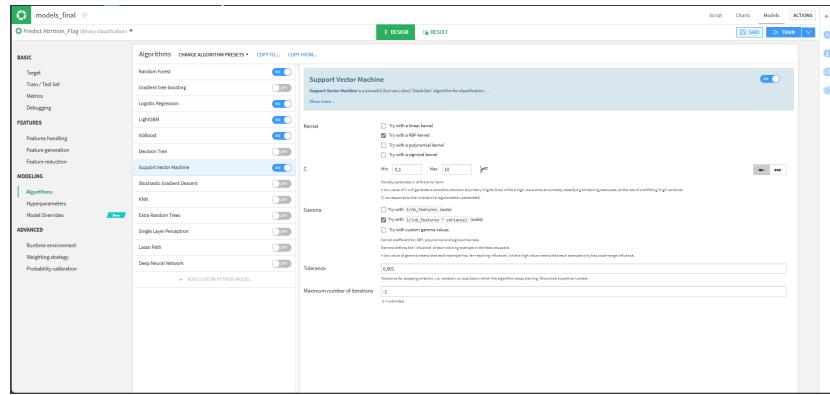
1. **Kolom tersebut mungkin tidak relevan:** Kolom-kolom tersebut mungkin dibuat khusus untuk menyimpan output dari model Naive Bayes. Jika tidak menggunakan metode Naive Bayes lagi, informasi yang disimpan dalam kolom-kolom tersebut mungkin tidak relevan atau tidak berguna untuk model baru yang ingin Anda latih.

2. **Kolom tersebut mungkin tidak independen:** Model Naive Bayes diasumsikan bahwa semua fitur adalah independen satu sama lain. Namun, dalam praktiknya, fitur-fitur yang dihasilkan oleh model Naive Bayes (seperti kolom-kolom tersebut) mungkin tidak benar-benar independen. Memasukkan kolom-kolom ini dalam latihan model baru mungkin melanggar asumsi independensi dan menghasilkan hasil yang tidak akurat.
3. **Potensi pengaruh ganda:** Model Naive Bayes mungkin telah mempertimbangkan kolom-kolom tersebut dalam proses pembuatannya. Jika memasukkan kembali kolom-kolom tersebut dalam latihan model baru, ini dapat menyebabkan pengaruh ganda atau overfitting pada data latih.
4. **Sederhanakan model:** Dengan menonaktifkan kolom-kolom tersebut, dapat menyederhanakan dataset Anda dan fokus pada fitur-fitur yang lebih relevan dan independen untuk model baru yang ingin dilatih. Ini dapat meningkatkan interpretabilitas dan kinerja mod

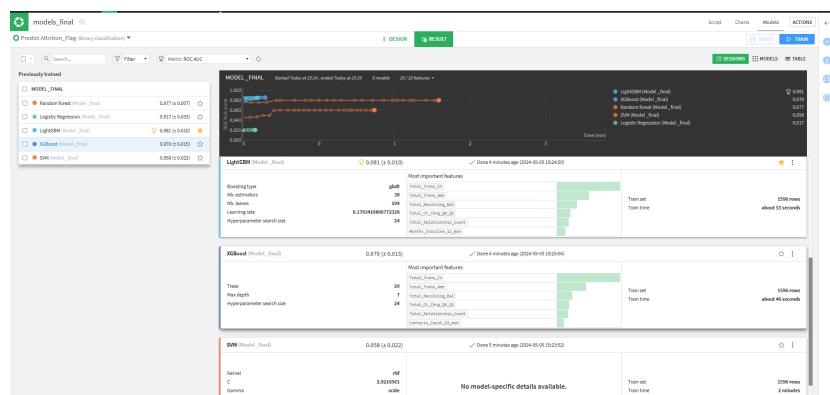
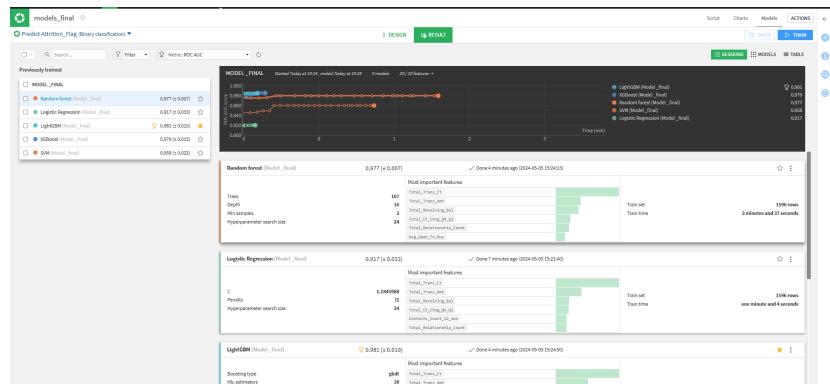
Menambahkan Auto Feature engineering yaitu menggunakan **Pairwise Linear Combination**

karena kemudahan interpretasi dan kompleksitas yang lebih rendah dari model yang dihasilkan. Dengan menggunakan pairwise linear combination, tetap dapat menangkap hubungan antara fitur-fitur dalam data. Selain itu, model dengan fitur-fitur linear sering kali lebih cepat untuk dilatih dan diberi prediksi, dan lebih mudah diinterpretasikan oleh orang lain.

Untuk algoritmanya disini saya menggunakan 4 Algoritma Random Forest, Logistic Regression, LightGBM, dan SVM:



lalu save dan train:



#### Previously trained

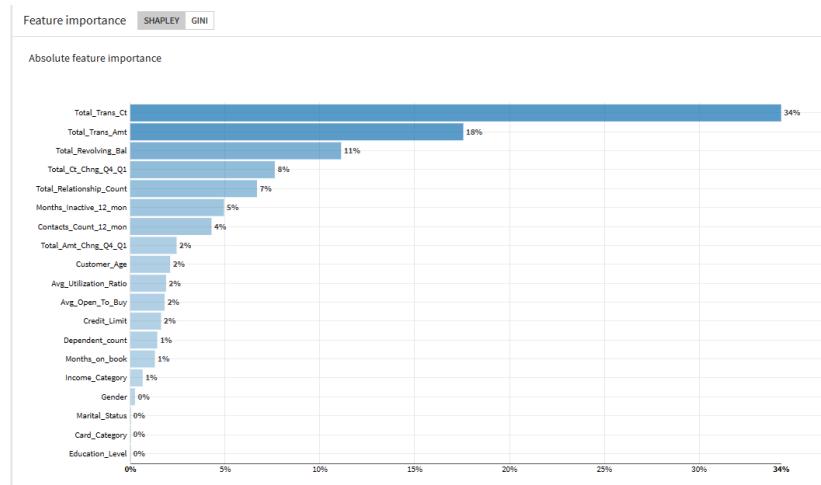
MODEL_FINAL	Random forest (Model_final)	0.977 (± 0.007)	★
Logistic Regression (Model_final)	0.917 (± 0.033)	★	
LightGBM (Model_final)	0.981 (± 0.010)	★	
XGBoost (Model_final)	0.979 (± 0.015)	★	
SVM (Model_final)	0.958 (± 0.022)	★	

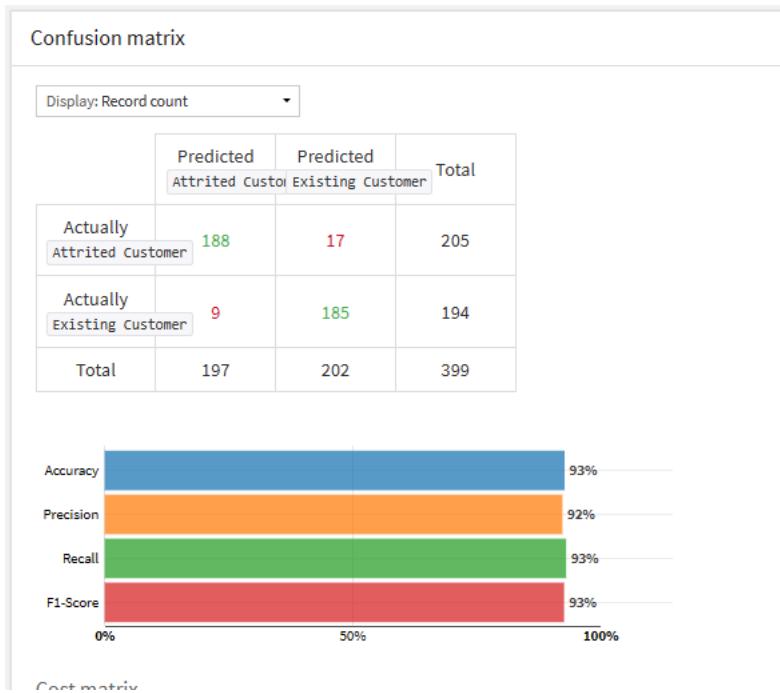
Bisa dilihat Feature Importancenya yang memiliki pengaruh terbesar adalah Total\_Trans\_Ct, Total\_Trans\_Amt, Total\_Revolving\_Bal

- 1. Total\_Trans\_Ct (Total Transaction Count):** Ini adalah jumlah total transaksi yang dilakukan oleh pelanggan dalam periode waktu tertentu. Ini mencakup semua jenis transaksi yang dilakukan dengan kartu kredit atau produk keuangan lainnya. Fitur ini memberikan gambaran tentang tingkat aktivitas penggunaan kartu kredit oleh pelanggan.
- 2. Total\_Trans\_Amt (Total Transaction Amount):** Ini adalah total nilai dari semua transaksi yang dilakukan oleh pelanggan dalam periode waktu tertentu. Ini mencakup jumlah uang yang dihabiskan atau ditransaksikan oleh pelanggan dalam menggunakan kartu kredit atau produk keuangan lainnya. Fitur ini memberikan informasi tentang pola belanja dan kebiasaan pengeluaran pelanggan.
- 3. Total\_Revolving\_Bal (Total Revolving Balance):** Ini adalah saldo total yang harus dibayar oleh pelanggan setelah melakukan pembelian dengan kartu kredit, tetapi belum dibayar sepenuhnya pada akhir periode penagihan. Ini mencakup saldo dari pembelian bulan sebelumnya yang masih harus dibayar, dan bunga yang dikenakan atas saldo tersebut. Fitur ini mencerminkan tingkat utang atau kewajiban finansial pelanggan terhadap bank atau lembaga keuangan.

Mengapa ketiga fitur ini merupakan fitur yang penting dalam model machine learning, karena:

- Informasi penting:** Ketiga fitur ini memberikan informasi penting tentang aktivitas penggunaan kartu kredit pelanggan dan keuangan mereka, yang dapat menjadi faktor penting dalam memprediksi perilaku atau keputusan pelanggan seperti churn atau berpindah.
- Keterkaitan dengan target:** Ketiga fitur ini mungkin memiliki keterkaitan yang kuat dengan variabel target yang ingin Anda prediksi, seperti churn. Misalnya, pelanggan dengan total transaksi yang tinggi atau saldo yang tinggi mungkin cenderung lebih berisiko untuk berpindah.
- Pemilihan fitur:** Saat melatih model machine learning, algoritma mungkin telah memilih fitur-fitur ini sebagai fitur penting berdasarkan analisisnya terhadap data latih dan hubungannya dengan variabel target. Ini mungkin terjadi jika fitur-fitur ini memiliki informasi prediktif yang kuat terhadap variabel target.





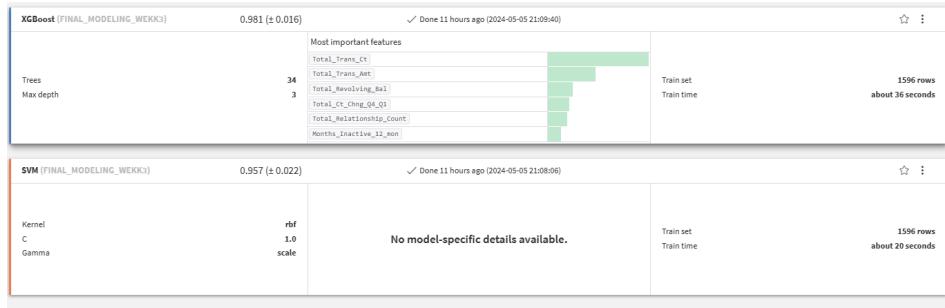
Disini saya masih ingin sedikit tuning machine learning saya karena kurang puas dengan hasilnya lalu saya tuning lagi dengan menambahkan hyper parameter tuning dengan 4 fold cross validation dengan strategi grid search dan menambahkan algoritma Gradient Boosted Trees berikut konfigurasinya:

The screenshot shows the Alteryx interface with the following configuration for a Random Forest model:

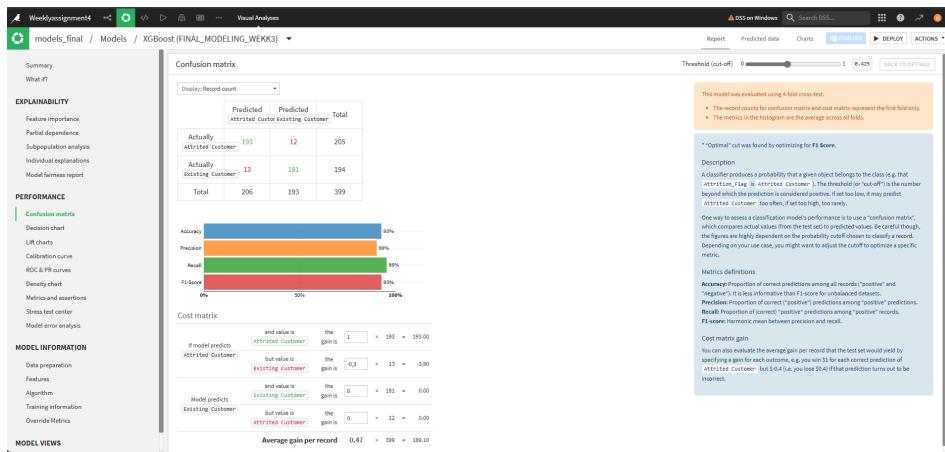
- Algorithms:** Random Forest is selected.
- Number of trees:** Set to 100.
- Feature sampling strategy:** Set to "Fixed proportion".
- Proportion of features to sample:** Set to 0.5.
- Maximum depth of tree:** Set to 10.
- Minimum samples per leaf:** Set to 5.
- Parallelism:** Set to 4.
- Allow sparse matrices:** Unchecked.

Maka dihasilkan Hasil akhir Gradient Boosted trees menghasilkan hasil yang lebih akurat dengan akurasi sekitar 98%:

Model	Akurasi	Statistik
Random forest (FINAL_MODELING_WEEK3)	0.979 ( $\pm 0.009$ )	✓ Done 11 hours ago (2024-05-05 21:08:53)
Gradient Boosted Trees (FINAL_MODELING_WEEK3)	0.982 ( $\pm 0.007$ )	✓ Done 11 hours ago (2024-05-05 21:08:39)
Logistic Regression (FINAL_MODELING_WEEK3)	0.917 ( $\pm 0.033$ )	✓ Done 11 hours ago (2024-05-05 21:07:32)
LightGBM (FINAL_MODELING_WEEK3)	0.981 ( $\pm 0.006$ )	✓ Done 11 hours ago (2024-05-05 21:09:34)



namun setelah diihat 11 algoritma xgboost memiliki kemampuan mengeneralisir yang lebih baik terlihat dari confussion matrix dimana jumlah false positif dan false negativenya lebih minim

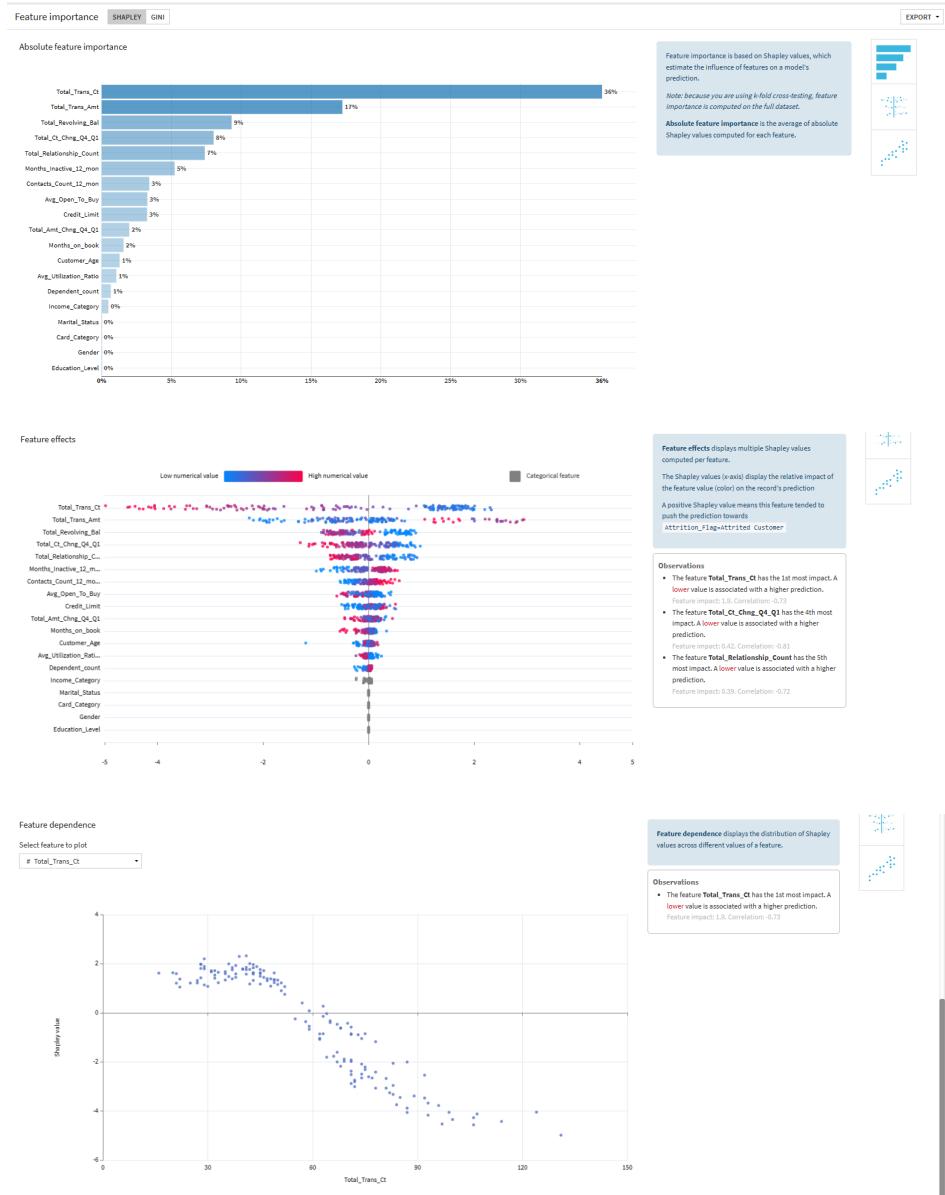


Dan menurut saya ini adalah model terbaik jika ingin digunakan untuk model memprediksi kasus ini. dikarenakan:

- Kinerja yang Tinggi:** XGBoost terkenal karena kinerja yang tinggi dalam banyak kasus, terutama dalam tugas klasifikasi. Ini sering kali menghasilkan model yang akurat dengan waktu pelatihan yang relatif cepat.
- Kemampuan mengatasi Data yang Tidak Seimbang:** Jika data memiliki masalah ketidakseimbangan kelas, di mana kelas yang lebih sedikit memiliki frekuensi yang jauh lebih rendah dari kelas mayoritas, XGBoost memiliki mekanisme built-in untuk menangani masalah ini melalui parameter bobot atau parameter logistic regression.
- Kemampuan untuk Menangani Fitur Non-Linier:** XGBoost mampu menangani hubungan non-linier antara fitur dan variabel target dengan baik. Ini dapat secara otomatis menyesuaikan dengan pola yang kompleks dalam data.
- Regularisasi Terintegrasi:** XGBoost memiliki mekanisme regularisasi yang terintegrasi untuk mencegah overfitting, seperti parameter pembatasan kedalaman pohon, pembatasan berat daun, dan pembatasan berat minimum.
- Optimasi Gradient Boosting:** XGBoost menggunakan teknik optimasi gradien yang canggih, yang memungkinkannya untuk membuat perubahan yang signifikan pada setiap iterasi, menghasilkan peningkatan yang cepat dalam kinerja model.
- Paralelitas dan Skalabilitas:** XGBoost mendukung pelatihan model paralel dan dapat diimplementasikan di platform yang mendukung paralelisasi seperti Apache Spark, memungkinkan pelatihan model dengan dataset besar.

Dengan demikian, XGBoost menjadi pilihan yang baik untuk pemodelan klasifikasi seperti kasus ini, terutama jika menghadapi masalah ketidakseimbangan kelas, membutuhkan kinerja yang tinggi, dan ingin menghindari overfitting yang dibuktikan dengan riset machine learning diatas algoritma XGboost merupakan algoritma terbaik.

Dan sama seperti sebelumnya:



Bisa dilihat Feature Importancenya yang memiliki pengaruh terbesar adalah Total\_Trans\_Ct, Total\_Trans\_Amt, Total\_Revolving\_Bal

- 1. Total\_Trans\_Ct (Total Transaction Count):** Ini adalah jumlah total transaksi yang dilakukan oleh pelanggan dalam periode waktu tertentu. Ini mencakup semua jenis transaksi yang dilakukan dengan kartu kredit atau produk keuangan lainnya. Fitur ini memberikan gambaran tentang tingkat aktivitas penggunaan kartu kredit oleh pelanggan.
- 2. Total\_Trans\_Amt (Total Transaction Amount):** Ini adalah total nilai dari semua transaksi yang dilakukan oleh pelanggan dalam periode waktu tertentu. Ini mencakup jumlah uang yang dihabiskan atau ditransaksikan oleh pelanggan dalam menggunakan kartu kredit atau produk keuangan lainnya. Fitur ini memberikan informasi tentang pola belanja dan kebiasaan pengeluaran pelanggan.
- 3. Total\_Revolving\_Bal (Total Revolving Balance):** Ini adalah saldo total yang harus dibayar oleh pelanggan setelah melakukan pembelian dengan kartu kredit, tetapi belum dibayar sepenuhnya pada akhir periode penagihan. Ini mencakup saldo dari pembelian bulan sebelumnya yang masih harus dibayar, dan bunga yang dikenakan atas saldo

tersebut. Fitur ini mencerminkan tingkat utang atau kewajiban finansial pelanggan terhadap bank atau lembaga keuangan.

Mengapa ketiga fitur ini merupakan fitur yang penting dalam model machine learning, karena:

- **Informasi penting:** Ketiga fitur ini memberikan informasi penting tentang aktivitas penggunaan kartu kredit pelanggan dan keuangan mereka, yang dapat menjadi faktor penting dalam memprediksi perilaku atau keputusan pelanggan seperti churn atau berpindah.
- **Keterkaitan dengan target:** Ketiga fitur ini mungkin memiliki keterkaitan yang kuat dengan variabel target yang ingin Anda prediksi, seperti churn. Misalnya, pelanggan dengan total transaksi yang tinggi atau saldo yang tinggi mungkin cenderung lebih berisiko untuk berpindah.

Mungkin cukup sekian dari saya jika ada salah mohon dimaafkan, terima kasih