Final Project Presentation

Nomor Kelompok: 8
Nama Mentor: Rachmadio Noval L.
Nama:

- Faldo Fajri Afrinanto
- Annisa Dewi Kusuma Wardani

Machine Learning Class

Program Studi Independen Bersertifikat
Zenius Bersama Kampus Merdeka

Latar Belakang

Latar Belakang Project

Sumber Data: **Customer Churn**
https://www.kaggle.com/datasets/barun2104/telecom-churn?datasetId=567482

Problem: **Classification**

Dengan pesatnya perkembangan industri telekomunikasi, penyedia layanan lebih condong ke arah perluasan basis pelanggan. Dikarenakan bahwa biaya untuk mendapatkan pelanggan baru jauh lebih besar daripada mempertahankan pelanggan yang sudah ada. Oleh karena itu, sangat penting bagi suatu perusahaan melakukan analisis untuk memahami perilaku konsumen dan memprediksi perilaku pelanggan tersebut, apakah mereka akan meninggalkan layanan atau tidak.

Explorasi Data dan Visualisasi

Business Understanding

Apa itu Customer Churn?

Customer churn adalah kehilangan pelanggan dari suatu bisnis. Churn dihitung dari berapa banyak pelanggan meninggalkan bisnis dalam waktu tertentu. Customer churn penting diketahui bisnis karena merupakan gambaran kesuksesan suatu bisnis dalam mempertahankan pelanggan.

Mengapa customer churn harus dihentikan?

Bisnis akan rugi besar jika kehilangan pelanggan. Faktanya, mempertahankan pelanggan lama lebih susah ketimbang mendapatkan pelanggan baru. Perusahaan akan menjadi lebih efisien jika jumlah pelanggan lama lebih banyak dari pelanggan baru. Jadi, diperlukan strategi untuk menghentikan customer churn atau kehilangan pelanggan karena merekalah sumber utama revenue bisnis

Business Understanding

Bagaimana cara menghentikan customer churn?

1. Cari tahu penyebab churn!
2. Tingkatkan ketertarikan customer terhadap layanan
3. Berikan reward pelanggan
4. Berikan keuntungan jangka panjang

Introduction Dataset

Dataset yang digunakan dalam final project diperoleh dari kaggle dengan nama dataset **Customer Churn,** terdiri dari 3333 Baris dan 11 Kolom

	Churn	AccountWeeks	ContractRenewal	DataPlan	DataUsage	CustServCalls	DayMins	DayCalls	MonthlyCharge	OverageFee	RoamMins
0	0	128	1	1	2.70	1	265.1	110	89.0	9.87	10.0
1	0	107	1	1	3.70	1	161.6	123	82.0	9.78	13.7
2	0	137	1	0	0.00	0	243.4	114	52.0	6.06	12.2
3	0	84	0	0	0.00	2	299.4	71	57.0	3.10	6.6
4	0	75	0	0	0.00	3	166.7	113	41.0	7.42	10.1
...	...	...	...	...	...	...	...	...	...	...	...
3328	0	192	1	1	2.67	2	156.2	77	71.7	10.78	9.9
3329	0	68	1	0	0.34	3	231.1	57	56.4	7.67	9.6
3330	0	28	1	0	0.00	2	180.8	109	56.0	14.44	14.1
3331	0	184	0	0	0.00	2	213.8	105	50.0	7.98	5.0
3332	0	74	1	1	3.70	0	234.4	113	100.0	13.30	13.7

Keterangan Kolom :

* Churn - 1 if customer cancelled service, 0 if not
* AccountWeeks - number of weeks customer has had active account
* ContractRenewal - 1 if customer recently renewed contract, 0 if not
* DataPlan - 1 if customer has data plan, 0 if not
* DataUsage - gigabytes of monthly data usage
* CustServCalls - number of calls into customer service
* DayMins - average daytime minutes per month
* DayCalls - average number of daytime calls
* MonthlyCharge - average monthly bill
* OverageFee - largest overage fee in last 12 months
* RoamMins - average number of roaming minutes

Data Cleansing

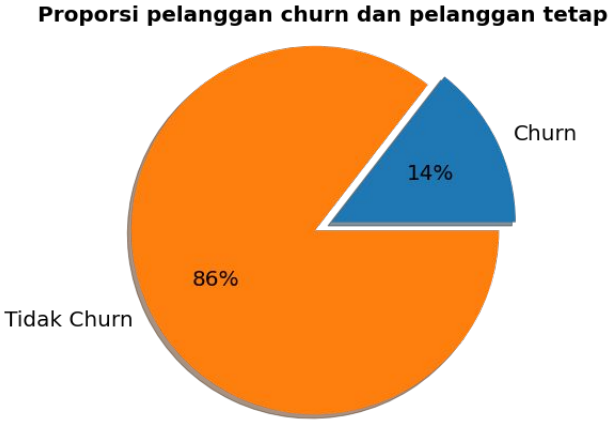
Pada dataset yang kami gunakan yaitu Customer Churn, tidak ditemukan adanya Missing Values dan tipe data pada setiap kolom sudah benar. Oleh karena itu dataset tidak membutuh data cleansing

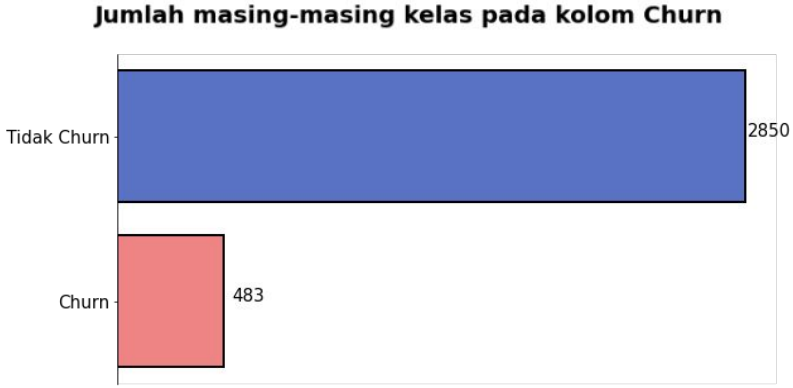
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3333 entries, 0 to 3332
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   Churn            3333 non-null   int64  
 1   AccountWeeks     3333 non-null   int64  
 2   ContractRenewal  3333 non-null   int64  
 3   DataPlan         3333 non-null   int64  
 4   DataUsage        3333 non-null   float64
 5   CustServCalls    3333 non-null   int64  
 6   DayMins          3333 non-null   float64
 7   DayCalls         3333 non-null   int64  
 8   MonthlyCharge    3333 non-null   float64
 9   OverageFee       3333 non-null   float64
 10  RoamMins         3333 non-null   float64
dtypes: float64(5), int64(6)
```

	Total Data Hilang	Persentase Data Hilang
Churn	0	0.0
AccountWeeks	0	0.0
ContractRenewal	0	0.0
DataPlan	0	0.0
DataUsage	0	0.0
CustServCalls	0	0.0
DayMins	0	0.0
DayCalls	0	0.0
MonthlyCharge	0	0.0
OverageFee	0	0.0
RoamMins	0	0.0

Exploratory Data Analysis

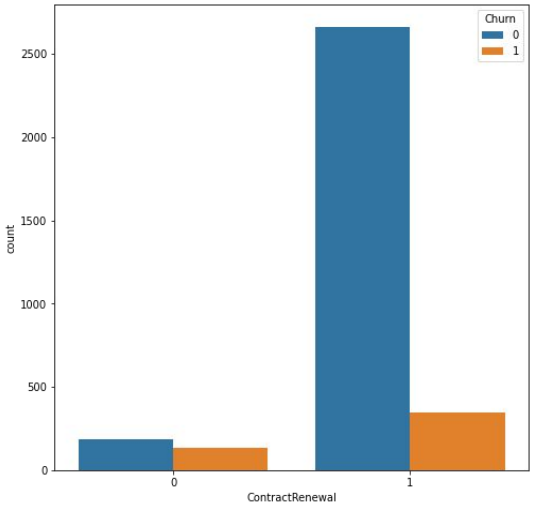
Exploratory Data Analysis memungkinkan untuk memahami isi data yang digunakan, mulai dari distribusi, frekuensi dan korelasi antar variabel. Dalam studi kasus ini kita akan melihat jumlah prosentase dari variabel target dan Persebaran data dari variabel predictor terhadap label (Churn).

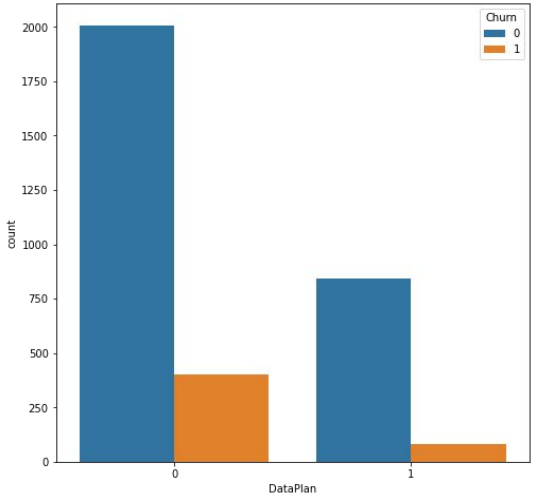


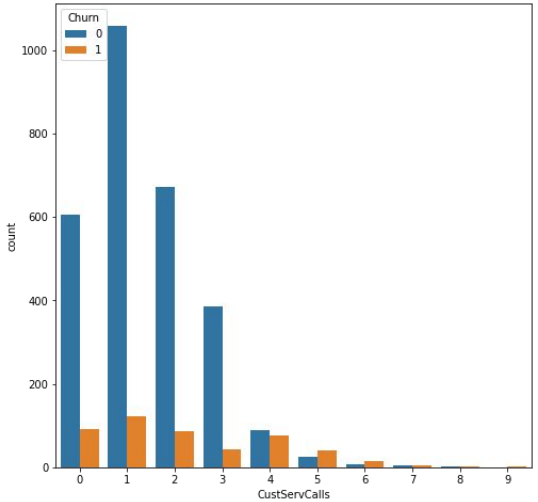


Exploratory Data Analysis

Exploratory Data Analysis (EDA) dengan Variabel Kategori terhadap Variabel Target



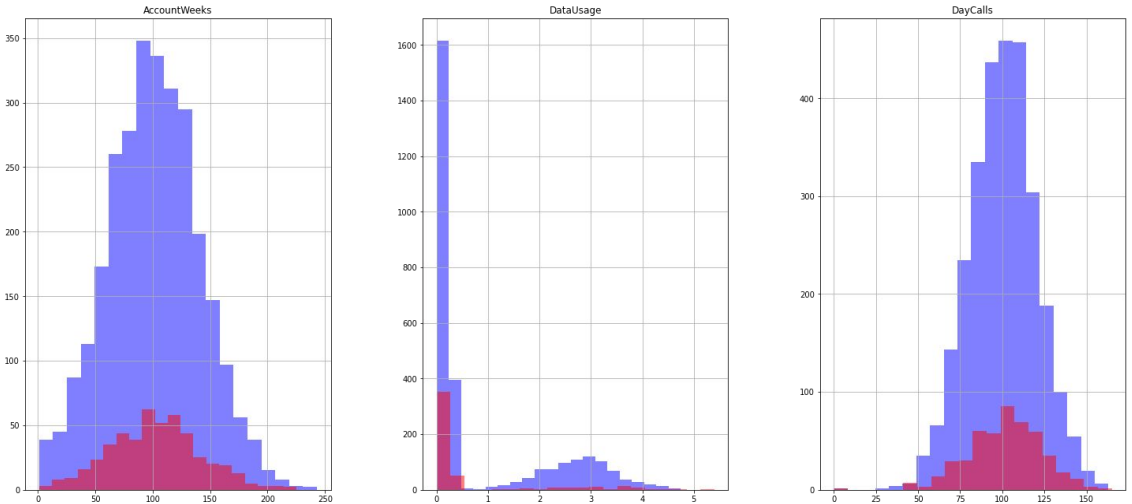




Pada visualisasi diatas dapat diketahui bahwa perpanjangan kontrak justru membuat pelanggan untuk churn, kecenderungan bahwa orang yang melakukan churn adalah orang-orang yang tidak menggunakan paket data, Serta pelanggan hanya menerima 1 kali dari customer service mempengaruhi pelanggan untuk churn.

Exploratory Data Analysis

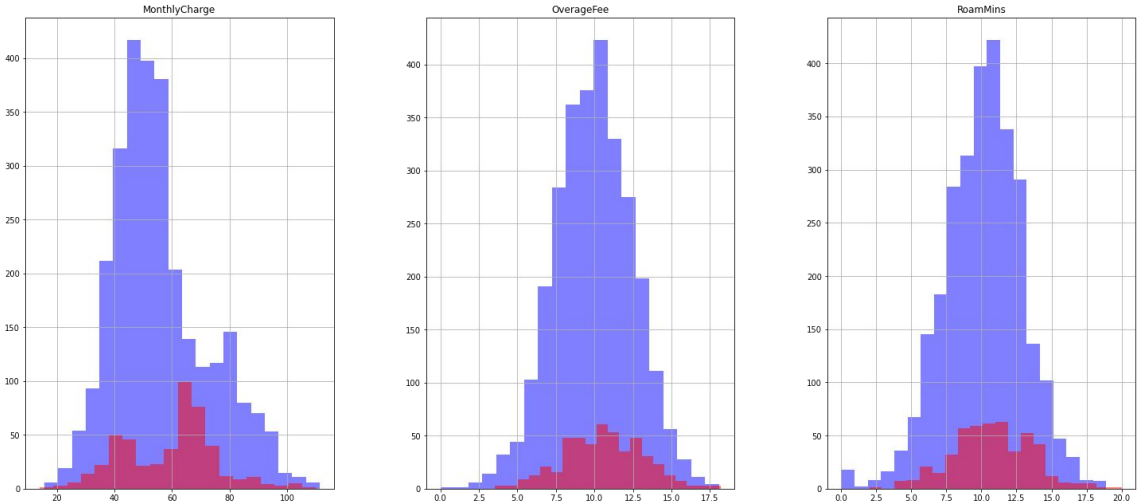
Exploratory Data Analysis (EDA) dengan Variabel Continuous terhadap Variabel Target



kecenderungan bahwa pelanggan yang melakukan churn adalah Pelanggan yang memiliki 100 akun aktif di setiap minggu nya dan pelanggan yang melakukan panggilan dengan rata-rata 100 kali panggilan perbulan, Serta pelanggan dengan penggunaan data sebesar 0 atau dapat diartikan tidak menggunakan paket.

Exploratory Data Analysis

Exploratory Data Analysis (EDA) dengan Variabel Continuous terhadap Variabel Target



Pelanggan yang melakukan churn mayoritas pelanggan dengan rata-rata tagihan bulanan sebesar 70 dollar serta terkena biaya lebih, Pelanggan yang menggunakan layanan roaming selama 8 - 13 menit kecenderungan untuk churn

Modelling

Imbalanced Dataset

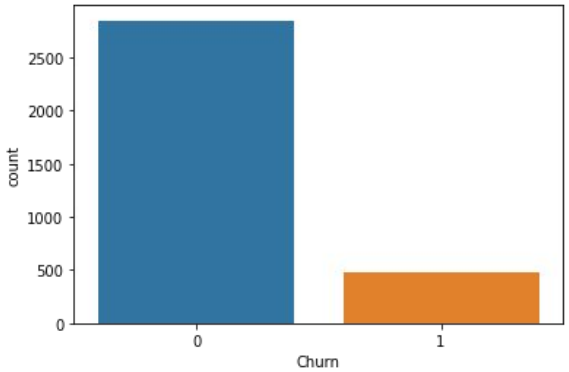
Imbalanced Dataset merupakan ketidakseimbangan data di suatu kelas pada kolom dataset. Pada umumnya suatu kolom memiliki data dengan kelas yang sedikit dan data dengan kelas yang terbanyak.

Apa dampak dari imbalanced dataset? Model akan lebih dominan mengenal kelas dengan jumlah data yang banyak pada saat selesai melakukan pelatihan model, Dampaknya akan mempengaruhi kinerja model pada saat akan melakukan prediksi.

Untuk mengatasi imbalanced dataset pada project ini, kami menggunakan **metode oversampling** yang dimana jumlah data dengan kelas sedikit akan diduplikasi sebanyak jumlah data dengan kelas terbanyak. Pada studi kasus ini kolom target yang digunakan adalah Churn yang terdiri dari 2 kelas yaitu kelas 0 yang berjumlah 2850 data dan kelas 1 yang berjumlah 483 data. Maka dari itu kita kan menyeimbangkan dataset sesuai kolom churn dengan menggunakan teknik oversampling.

Resampling Dataset

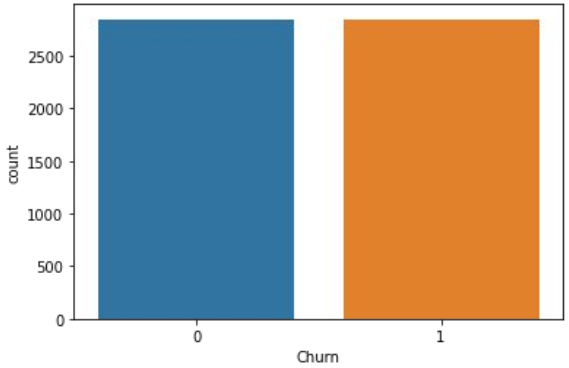
Sebelum menggunakan oversampling



Jumlah dataset awal

```
0    2850
1     483
Name: Churn, dtype: int64
```

Sesudah menggunakan oversampling



Jumlah data sesudah resampling

```
0    2850
1    2850
Name: Churn, dtype: int64
```

Train Test Split

- Train test split adalah salah satu metode yang penting sebelum melakukan proses machine learning.
- Metode ini membagi dataset menjadi dua bagian yakni data train yang digunakan untuk pelatihan model dan data test yang digunakan untuk mengevaluasi kinerja model setelah melalui proses pelatihan.
- Train/test split dapat digunakan untuk permasalahan regresi maupun klasifikasi.

Train Test Split yang digunakan pada final project kali ini sebesar 70% Data Train dan 30% Data test

```
#spliting dataset 70% data train dan 30% data test

from sklearn.model_selection import train_test_split as split
X_train, X_test, y_train, y_test = split(X, y, test_size=0.3, random_state=42)
```

Modelling

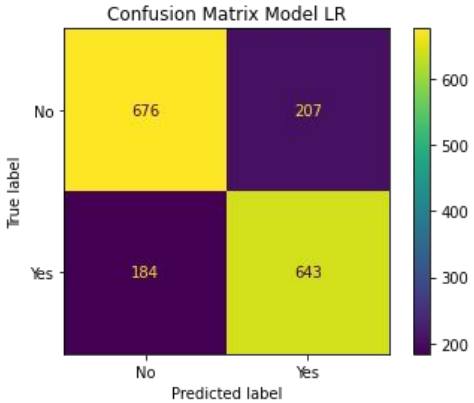
- Setelah melakukan oversampling dan spliting dataset, selanjutnya dilakukan pelatihan model machine learning.
- Pada project kali ini, kami menggunakan beberapa model Machine Learning, yaitu Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, K-Nearest Neighbour, Naive Bayes dan Gradient Boosting.
- Tujuan kami menggunakan beberapa model untuk mencari model terbaik dengan cara membandingkan hasil evaluasi dari setiap model. Evaluasi model yang digunakan yaitu classification report dan confusion matrix.

Logistic Regression

Laporan Klasifikasi Model LR

	precision	recall	f1-score	support
0	0.72	0.77	0.74	813
1	0.77	0.73	0.75	897
accuracy			0.75	1710
macro avg	0.75	0.75	0.75	1710
weighted avg	0.75	0.75	0.75	1710

Hasil Akurasi dengan menggunakan model Logistic Regression sebesar **75%**



Hasil Confusion Matrix dengan menggunakan Model Logistic Regression

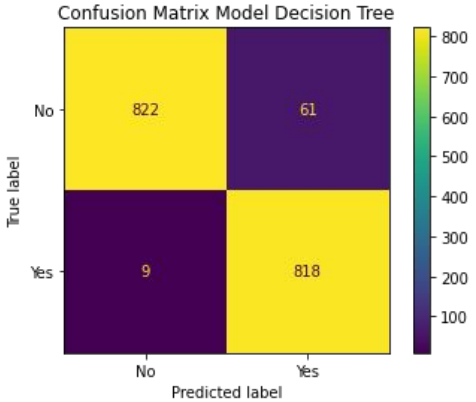
True Positive : 676
True Negative : 643
False Positive : 207
False Negative : 184

Decision Tree

Laporan Klasifikasi Model Decision Tree

	precision	recall	f1-score	support
0	1.00	0.90	0.95	813
1	0.92	1.00	0.96	897
accuracy			0.95	1710
macro avg	0.96	0.95	0.95	1710
weighted avg	0.96	0.95	0.95	1710

Hasil Akurasi dengan menggunakan model Logistic Regression sebesar **95%**



Hasil Confusion Matrix dengan menggunakan Model Logistic Regression

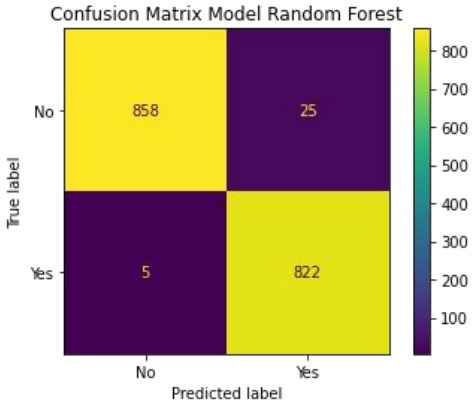
True Positive : 882
True Negative : 818
False Positive : 61
False Negative : 9

Random Forest

Laporan Klasifikasi Model Random Forest

	precision	recall	f1-score	support
0	1.00	0.95	0.97	813
1	0.96	1.00	0.98	897
accuracy			0.98	1710
macro avg	0.98	0.98	0.98	1710
weighted avg	0.98	0.98	0.98	1710

Hasil Akurasi dengan menggunakan model Logistic Regression sebesar **98%**



Hasil Confusion Matrix dengan menggunakan Model Logistic Regression

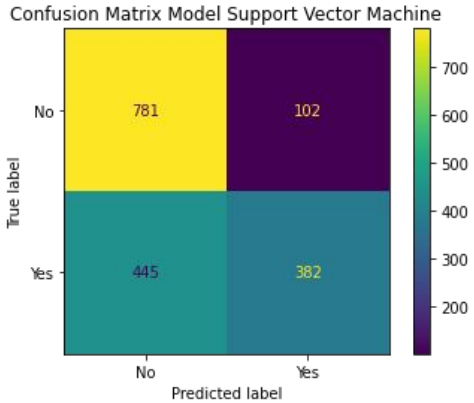
True Positive : 858
True Negative : 822
False Positive : 25
False Negative : 5

Support Vector Machine

Laporan Klasifikasi Model Support Vector Machine

	precision	recall	f1-score	support
0	0.58	0.91	0.71	813
1	0.84	0.41	0.55	897
accuracy			0.65	1710
macro avg	0.71	0.66	0.63	1710
weighted avg	0.72	0.65	0.63	1710

Hasil Akurasi dengan menggunakan model Logistic Regression sebesar **65%**



Hasil Confusion Matrix dengan menggunakan Model Logistic Regression

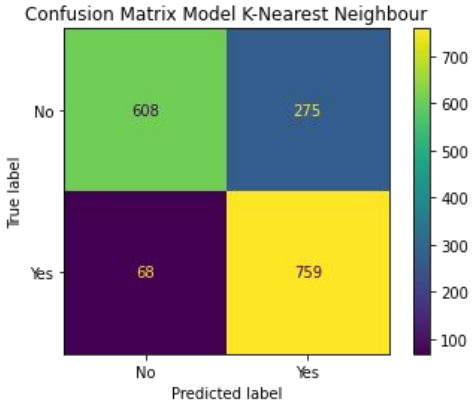
True Positive : 781
True Negative : 382
False Positive : 102
False Negative : 445

K-Nearest Neighbour

Laporan Klasifikasi Model K-Nearest Neighbour

	precision	recall	f1-score	support
0	0.90	0.69	0.78	883
1	0.73	0.92	0.82	827
accuracy			0.80	1710
macro avg	0.82	0.80	0.80	1710
weighted avg	0.82	0.80	0.80	1710

Hasil Akurasi dengan menggunakan model Logistic Regression sebesar **80%**



Hasil Confusion Matrix dengan menggunakan Model Logistic Regression

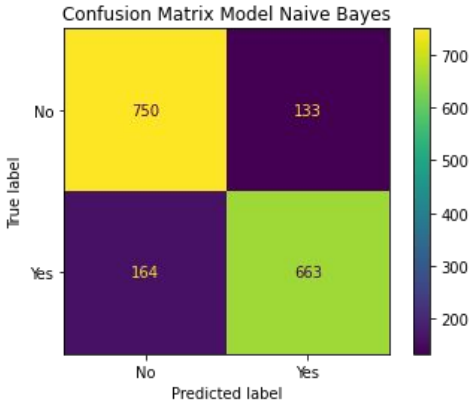
True Positive : 608
True Negative : 759
False Positive : 275
False Negative : 68

Naive Bayes

Laporan Klasifikasi Model Naive Bayes

	precision	recall	f1-score	support
0	0.82	0.85	0.83	883
1	0.83	0.80	0.82	827
accuracy			0.83	1710
macro avg	0.83	0.83	0.83	1710
weighted avg	0.83	0.83	0.83	1710

Hasil Akurasi dengan menggunakan model Logistic Regression sebesar **83%**



Hasil Confusion Matrix dengan menggunakan Model Logistic Regression

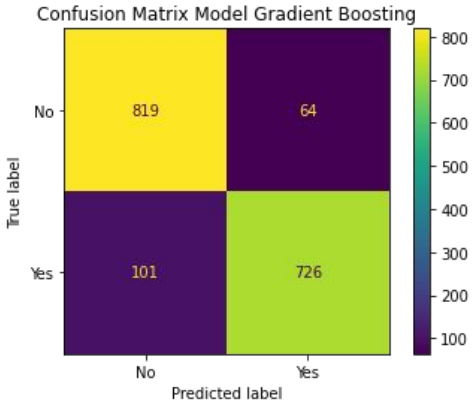
True Positive : 750
True Negative : 663
False Positive : 133
False Negative : 164

Gradient Boosting

Laporan Klasifikasi Model Gradient Boosting

	precision	recall	f1-score	support
0	0.89	0.93	0.91	883
1	0.92	0.88	0.90	827
accuracy			0.90	1710
macro avg	0.90	0.90	0.90	1710
weighted avg	0.90	0.90	0.90	1710

Hasil Akurasi dengan menggunakan model Logistic Regression sebesar **90%**



Hasil Confusion Matrix dengan menggunakan Model Logistic Regression

True Positive : 819
True Negative : 726
False Positive : 64
False Negative : 101

Final Model

Setelah dataset diuji dengan beberapa model , akurasi tertinggi didapat dengan menggunakan model Random Forest sebesar 98% , maka dari itu model akan dilanjutkan dengan penggunaan hyperparameter tuning untuk mengoptimalkan kinerja dari model agar mendapatkan akurasi yang lebih baik dan menggunakan feature selection untuk memilih feature yang berpengaruh dan mengesampingkan feature yang tidak berpengaruh.

Final Model

Feature Selection adalah memilih feature yang berpengaruh dan mengesampingkan feature yang tidak berpengaruh dalam suatu kegiatan pemodelan. Maka dari itu dalam project ini menggunakan teknik feature selection yang bernama feature importance. Feature Importance menghasilkan skor untuk setiap fitur pada dataset, semakin tinggi skor semakin penting fitur tersebut

	Fitur yang Penting
DayMins	0.179849
CustServCalls	0.175095
MonthlyCharge	0.160981
ContractRenewal	0.099004
OverageFee	0.091516
RoamMins	0.074113
DataUsage	0.069022
DayCalls	0.063929
AccountWeeks	0.063006
DataPlan	0.023485

Dapat dilihat pada tabel hasil dari penggunaan fungsi feature importance yang dimana fitur DataPlan mendapatkan skor paling rendah, maka dari itu dapat dikatakan fitur tersebut tidak mempengaruhi model. Dengan ini apakah dengan menghapus fitur tersebut dapat meningkat performa dari model.

Final Model

Mencoba menghapus kolom “DataPlan”, kemudian dataset diuji kembali dengan menggunakan model random forest beserta penggunaan hyperparameter tuning

```
#mendefinisikan feature matrix X dan Y dengan menghapus fitur DataPlan

X_train=X_train.drop('DataPlan', axis=1)
y_train=y_train
X_test=X_test.drop('DataPlan', axis=1)
y_test=y_test
```

Final Model

Hyperparameter Tuning adalah nilai untuk parameter yang digunakan untuk mempengaruhi proses pelatihan model dengan penggunaan hyperparameter tuning dapat meningkatkan performa dari model. Dalam project ini kami menggunakan hyperparameter tuning pada model final yang diusulkan yaitu model random forest, berikut hyperparameter tuning yang digunakan

```
hyperparams = {
    'max_depth':[2,3,5,10,20],
    'min_samples_leaf':[5,10,20,50,100,200],
    'n_estimators':[10,25,30,50,100,200]
}
```

Untuk membantu dalam pemilihan parameter terbaik kami menggunakan fungsi dari **GridSearchCV. GridSearchCV** adalah teknik untuk mencari nilai parameter terbaik dari kumpulan parameter grid yang diberikan. Nilai parameter terbaik diekstraksi dan kemudian digunakan pada model.

```
grid_search.best_params_
```

```
{'max_depth': 20, 'min_samples_leaf': 5, 'n_estimators': 100}
```

Final Model

Setelah penggunaan hyperparameter tuning pada model random forest , akurasi yang didapatkan sebesar 95%

Laporan Klasifikasi Model Random Forest dengan hyperparameter tuning

	precision	recall	f1-score	support
Pelanggan Tetap	0.95	0.95	0.95	846
Pelanggan Churn	0.95	0.95	0.95	864
accuracy			0.95	1710
macro avg	0.95	0.95	0.95	1710
weighted avg	0.95	0.95	0.95	1710

Final Model

Pengujian prediksi dengan menggunakan model terbaik

```
Predict(
    model=rf_model,
    AccountWeeks=128,
    ContractRenewal=1,
    DataUsage=2.70,
    CustServCalls=1,
    DayMins=265.1,
    DayCalls=110,
    MonthlyCharge=89.0,
    OverageFee=9.87,
    RoamMins=10.0,
)
```

```
Prediksi Pelanggan : 0


Keterangan
0 : Pelanggan akan bertahan berlangganan
1 : Pelanggan akan berhenti berlangganan
```

Conclusion

Kesimpulan

Faktor yang dapat mempengaruhi Churn :

- Memberikan biaya lebih terhadap pelanggan dapat menyebabkan pelanggan tidak nyaman dan harus berpindah ke provider lain
- Penggunaan layanan roaming data yang lama berdampak pada biaya bulanan yang membengkak menjadikan pelanggan berpindah ke provider lain

Cara untuk mengatasi pelanggan agar tetap bertahan dengan layanan

- Lebih banyak panggilan customer service kepada pelanggan akan lebih rentan untuk churn
- Mempromosikan paket data kepada pelanggan yang tidak menggunakan paket data, agar pelanggan tersebut bertahan menggunakan layanan provider

Terima kasih!

Ada pertanyaan?