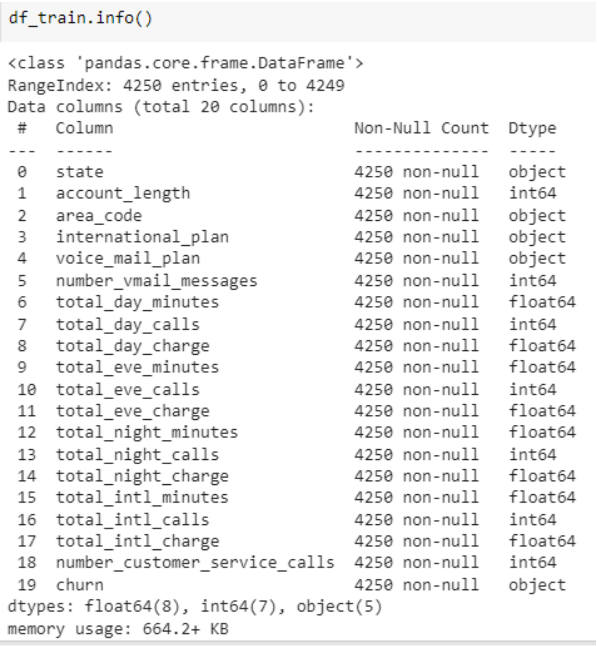
CUSTOMER CHURN PREDICTION

IMPORT DATA

Dataset *customer churn* berisi 4250 sampel dan setiap sampel berisi 19 variabel dengan 1 variabel target.

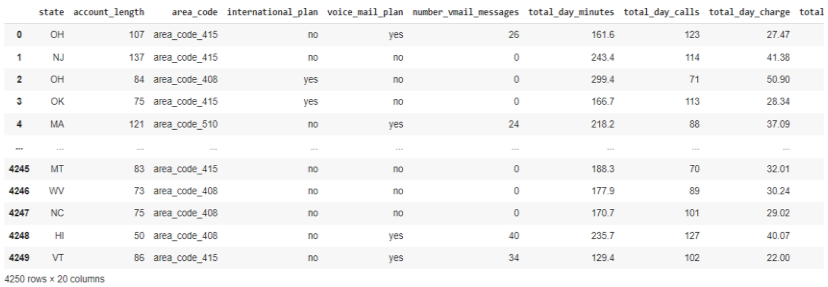


```
df_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4250 entries, 0 to 4249
Data columns (total 20 columns):
 #   Column                         Non-Null Count  Dtype  
---  ------                         --------------  -----  
 0   state                          4250 non-null   object 
 1   account_length                 4250 non-null   int64  
 2   area_code                      4250 non-null   object 
 3   international_plan             4250 non-null   object 
 4   voice_mail_plan                4250 non-null   object 
 5   number_vmail_messages          4250 non-null   int64  
 6   total_day_minutes              4250 non-null   float64
 7   total_day_calls                4250 non-null   int64  
 8   total_day_charge               4250 non-null   float64
 9   total_eve_minutes              4250 non-null   float64
 10  total_eve_calls                4250 non-null   int64  
 11  total_eve_charge               4250 non-null   float64
 12  total_night_minutes            4250 non-null   float64
 13  total_night_calls              4250 non-null   int64  
 14  total_night_charge             4250 non-null   float64
 15  total_intl_minutes             4250 non-null   float64
 16  total_intl_calls               4250 non-null   int64  
 17  total_intl_charge              4250 non-null   float64
 18  number_customer_service_calls  4250 non-null   int64  
 19  churn                          4250 non-null   object 
dtypes: float64(8), int64(7), object(5)
memory usage: 664.2+ KB
```

Melakukan import data *training* dan data *testing* ke dalam *workspace* dan menampilkan bentuk dari *dataset* tersebut.

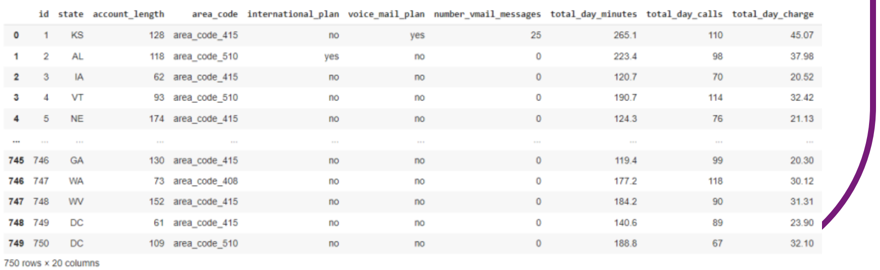
Data Training



	state	account_length	area_code	international_plan	voice_mail_plan	number_vmail_messages	total_day_minutes	total_day_calls	total_day_charge	total
0	OH	107	area_code_415	no	yes	26	161.6	123	27.47	
1	NJ	137	area_code_415	no	no	0	243.4	114	41.38	
2	OH	84	area_code_408	yes	no	0	299.4	71	50.90	
3	OK	75	area_code_415	yes	no	0	166.7	113	28.34	
4	MA	121	area_code_510	no	yes	24	218.2	88	37.09	
...	...	...	...	...	...	...	...	...	...	
4245	MT	83	area_code_415	no	no	0	188.3	70	32.01	
4246	WV	73	area_code_408	no	no	0	177.9	89	30.24	
4247	NC	75	area_code_408	no	no	0	170.7	101	29.02	
4248	HI	50	area_code_408	no	yes	40	235.7	127	40.07	
4249	VT	86	area_code_415	no	yes	34	129.4	102	22.00	

4250 rows × 20 columns

Data Testing

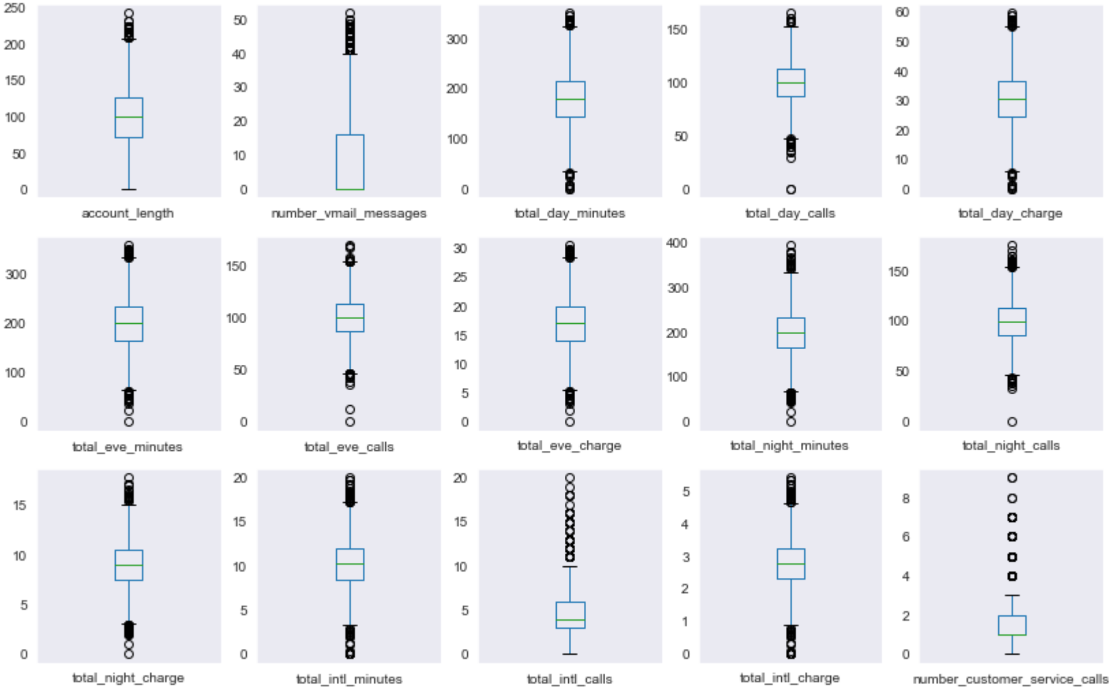


	id	state	account_length	area_code	international_plan	voice_mail_plan	number_vmail_messages	total_day_minutes	total_day_calls	total_day_charge
0	1	KS	128	area_code_415	no	yes	25	265.1	110	45.07
1	2	AL	118	area_code_510	yes	no	0	223.4	98	37.98
2	3	IA	62	area_code_415	no	no	0	120.7	70	20.52
3	4	VT	93	area_code_510	no	no	0	190.7	114	32.42
4	5	NE	174	area_code_415	no	no	0	124.3	76	21.13
...	...	...	...	...	...	...	...	...	...	...
745	746	GA	130	area_code_415	no	no	0	119.4	99	20.30
746	747	WA	73	area_code_408	no	no	0	177.2	118	30.12
747	748	WV	152	area_code_415	no	no	0	184.2	90	31.31
748	749	DC	61	area_code_415	no	no	0	140.6	89	23.90
749	750	DC	109	area_code_510	no	no	0	188.8	67	32.10

750 rows × 20 columns

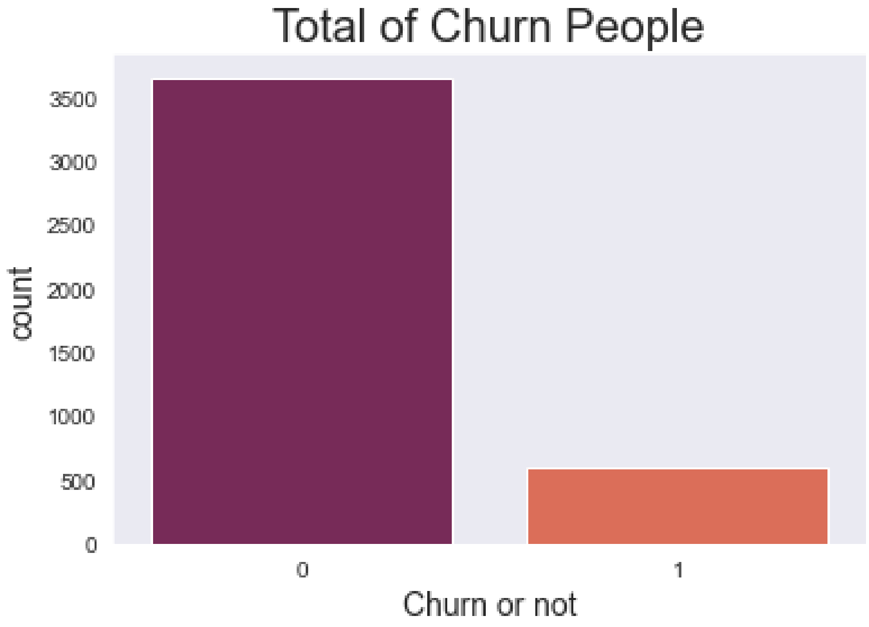
EXPLANATORY DATA ANALYSIS

- **Deteksi dan Visualisasi Outliers**



EXPLANATORY DATA ANALYSIS

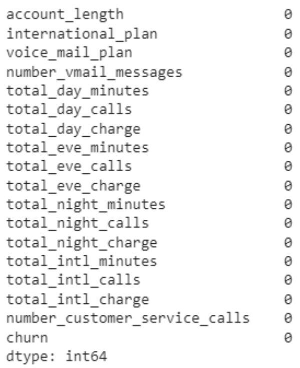
- **Target Variable : Churn**



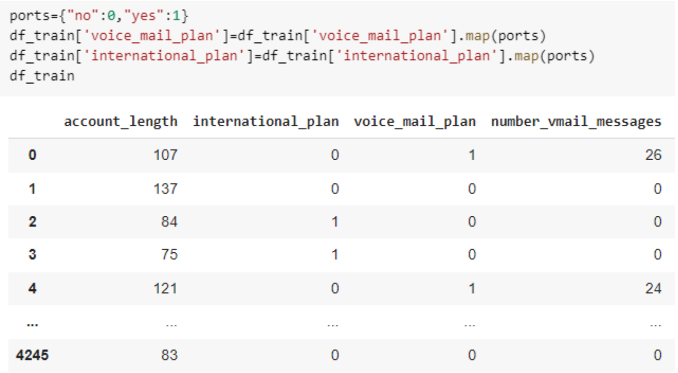
Dapat dilihat bahwa dataset terindikasi memiliki ketidak seimbangan karena jumlah tidak *churn* lebih banyak daripada jumlah *churn*.

DATA PREPROCESSING

- Melakukan pengecekan *missing value*, tidak terdapat *missing value*.
- Menghapus kolom dari data ‘*state*’ dan ‘*area_code*’ karena tidak diikutsertakan dalam pemodelan.
- Mengubah tipe data yang masih berbentuk kategorik dalam bentuk numerik, yaitu data ‘*voice_mail_plan*’ dan ‘*international_plan*’.
- Melakukan korelasi data.
- Melakukan *scaling* data dengan *Min-max Scaler*.

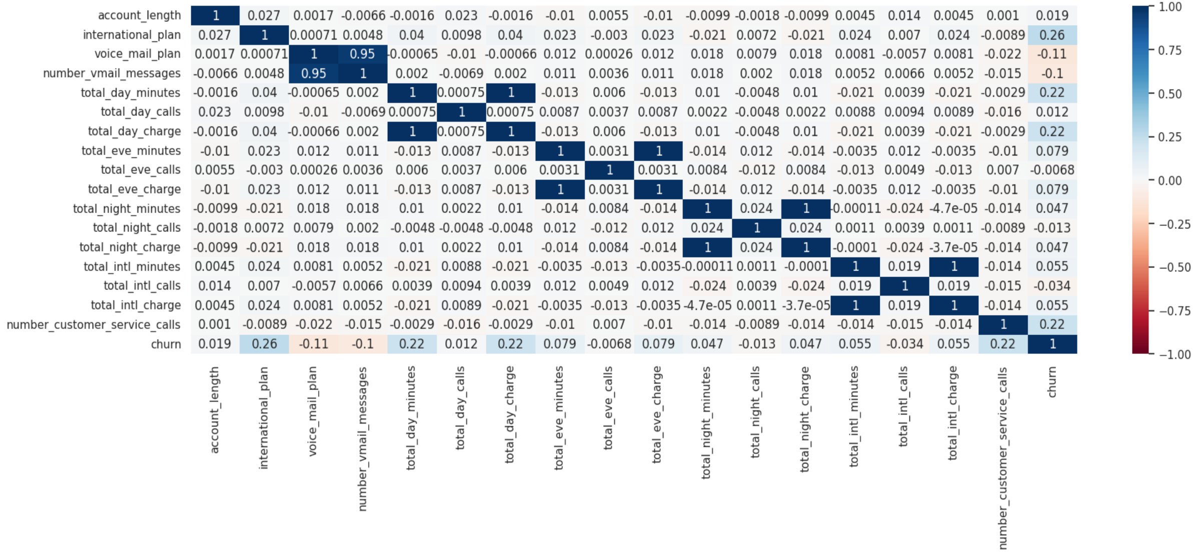


```
account_length                   0
international_plan               0
voice_mail_plan                  0
number_vmail_messages            0
total_day_minutes                0
total_day_calls                  0
total_day_charge                 0
total_eve_minutes                0
total_eve_calls                  0
total_eve_charge                 0
total_night_minutes              0
total_night_calls                0
total_night_charge               0
total_intl_minutes               0
total_intl_calls                 0
total_intl_charge                0
number_customer_service_calls    0
churn                            0
dtype: int64
```



```
ports={"no":0,"yes":1}
df_train['voice_mail_plan']=df_train['voice_mail_plan'].map(ports)
df_train['international_plan']=df_train['international_plan'].map(ports)
df_train
```

	account_length	international_plan	voice_mail_plan	number_vmail_messages
0	107	0	1	26
1	137	0	0	0
2	84	1	0	0
3	75	1	0	0
4	121	0	1	24
...	...	...	...	...
4245	83	0	0	0



	account_length	international_plan	voice_mail_plan	number_vmail_messages	total_day_minutes	total_day_calls	total_day_charge	total_eve_minutes	total_eve_calls	total_eve_charge	total_night_minutes	total_night_calls	total_night_charge	total_intl_minutes	total_intl_calls	total_intl_charge	number_customer_service_calls	churn
account_length	1	0.027	0.0017	-0.0066	-0.0016	0.023	-0.0016	-0.01	0.0055	-0.01	-0.0099	-0.0018	-0.0099	0.0045	0.014	0.0045	0.001	0.019
international_plan	0.027	1	0.00071	0.0048	0.04	0.0098	0.04	0.023	-0.003	0.023	-0.021	0.0072	-0.021	0.024	0.007	0.024	-0.0089	0.26
voice_mail_plan	0.0017	0.00071	1	0.95	-0.00065	-0.01	-0.00066	0.012	0.00026	0.012	0.018	0.0079	0.018	0.0081	-0.0057	0.0081	-0.022	-0.11
number_vmail_messages	-0.0066	0.0048	0.95	1	0.002	-0.0069	0.002	0.011	0.0036	0.011	0.018	0.002	0.018	0.0052	0.0066	0.0052	-0.015	-0.1
total_day_minutes	-0.0016	0.04	-0.00065	0.002	1	0.00075	1	-0.013	0.006	-0.013	0.01	-0.0048	0.01	-0.021	0.0039	-0.021	-0.0029	0.22
total_day_calls	0.023	0.0098	-0.01	-0.0069	0.00075	1	0.00075	0.0087	0.0037	0.0087	0.0022	-0.0048	0.0022	0.0088	0.0094	0.0089	-0.016	0.012
total_day_charge	-0.0016	0.04	-0.00066	0.002	1	0.00075	1	-0.013	0.006	-0.013	0.01	-0.0048	0.01	-0.021	0.0039	-0.021	-0.0029	0.22
total_eve_minutes	-0.01	0.023	0.012	0.011	-0.013	0.0087	-0.013	1	0.0031	1	-0.014	0.012	-0.014	-0.0035	0.012	-0.0035	-0.01	0.079
total_eve_calls	0.0055	-0.003	0.00026	0.0036	0.006	0.0037	0.006	0.0031	1	0.0031	0.0084	-0.012	0.0084	-0.013	0.0049	-0.013	0.007	-0.0068
total_eve_charge	-0.01	0.023	0.012	0.011	-0.013	0.0087	-0.013	1	0.0031	1	-0.014	0.012	-0.014	-0.0035	0.012	-0.0035	-0.01	0.079
total_night_minutes	-0.0099	-0.021	0.018	0.018	0.01	0.0022	0.01	-0.014	0.0084	-0.014	1	0.024	1	-0.00011	-0.024	-4.7e-05	-0.014	0.047
total_night_calls	-0.0018	0.0072	0.0079	0.002	-0.0048	-0.0048	-0.0048	0.012	-0.012	0.012	0.024	1	0.024	0.0011	0.0039	0.0011	-0.0089	-0.013
total_night_charge	-0.0099	-0.021	0.018	0.018	0.01	0.0022	0.01	-0.014	0.0084	-0.014	1	0.024	1	-0.0001	-0.024	-3.7e-05	-0.014	0.047
total_intl_minutes	0.0045	0.024	0.0081	0.0052	-0.021	0.0088	-0.021	-0.0035	-0.013	-0.0035	-0.00011	0.0011	-0.0001	1	0.019	1	-0.014	0.055
total_intl_calls	0.014	0.007	-0.0057	0.0066	0.0039	0.0094	0.0039	0.012	0.0049	0.012	-0.024	0.0039	-0.024	0.019	1	0.019	-0.015	-0.034
total_intl_charge	0.0045	0.024	0.0081	0.0052	-0.021	0.0089	-0.021	-0.0035	-0.013	-0.0035	-4.7e-05	0.0011	-3.7e-05	1	0.019	1	-0.014	0.055
number_customer_service_calls	0.001	-0.0089	-0.022	-0.015	-0.0029	-0.016	-0.0029	-0.01	0.007	-0.01	-0.014	-0.0089	-0.014	-0.014	-0.015	-0.014	1	0.22
churn	0.019	0.26	-0.11	-0.1	0.22	0.012	0.22	0.079	-0.0068	0.079	0.047	-0.013	0.047	0.055	-0.034	0.055	0.22	1

MODELLING

Tahap 1 – Pembuatan Model

1. K-Nearest Neighbor (KNN)
2. Decision Tree
3. Random Forest

Tahap 2 – Meningkatkan Performa Model

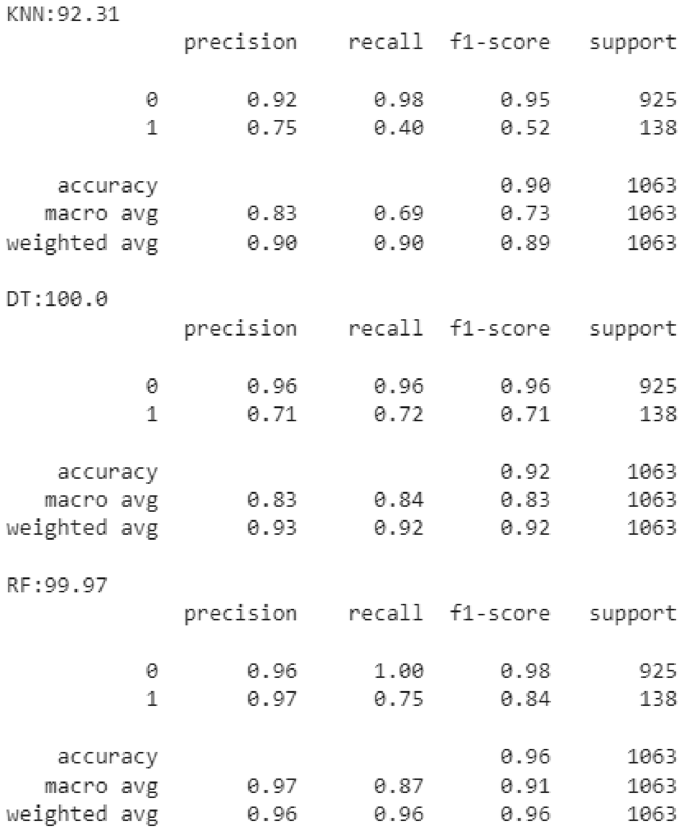
1. Mencari model terbaik dengan *Randomized Search CV*
2. Mencari parameter terbaik dengan *Grid Search CV*

MODELLING RESULT

Tahap 1 – Pembuatan Model

Classifier	Accuracy	Precision	Recall
K-Nearest Neighbor (KNN)	90%	75%	40%
Decision Tree	92%	71%	72%
Random Forest	96%	97%	75%

Pada tahap ini, model yang direkomendasikan adalah *Random Forest* dengan *accuracy* : 0.90, *precision* : 0.97, dan *recall* : 0.75.

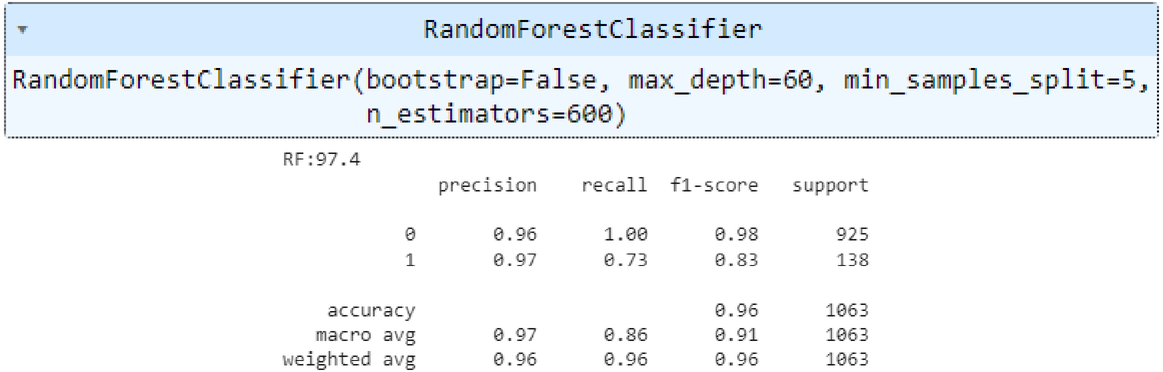


```
KNN:92.31
              precision    recall  f1-score   support

           0       0.92      0.98      0.95       925
           1       0.75      0.40      0.52       138

    accuracy                           0.90      1063
   macro avg       0.83      0.69      0.73      1063
weighted avg       0.90      0.90      0.89      1063

DT:100.0
              precision    recall  f1-score   support

           0       0.96      0.96      0.96       925
           1       0.71      0.72      0.71       138

    accuracy                           0.92      1063
   macro avg       0.83      0.84      0.83      1063
weighted avg       0.93      0.92      0.92      1063

RF:99.97
              precision    recall  f1-score   support

           0       0.96      1.00      0.98       925
           1       0.97      0.75      0.84       138

    accuracy                           0.96      1063
   macro avg       0.97      0.87      0.91      1063
weighted avg       0.96      0.96      0.96      1063
```

MODELLING RESULT

Tahap 2 – Meningkatkan Performa Model

- Pada tahap ini dilakukan pencarian model terbaik menggunakan *Randomized Search CV* *dan Grid Search CV* berdasarkan hasil Tahap 1 dimana model terbaiknya adalah *Random Forest*.



```
▾                    RandomForestClassifier
RandomForestClassifier(bootstrap=False, max_depth=60, min_samples_split=5,
                       n_estimators=600)
```

```
RF:97.4
              precision    recall  f1-score   support

           0       0.96      1.00      0.98       925
           1       0.97      0.73      0.83       138

    accuracy                           0.96      1063
   macro avg       0.97      0.86      0.91      1063
weighted avg       0.96      0.96      0.96      1063
```

- Dari proses pemodelan yang dilakukan, rekomendasi yang digunakan adalah *Random Forest* menggunakan *max_depth* = 60, *min_samples_split* = 5, dan *n_estimator* = 600 dengan *accuracy* : 0.96 , *precision* : 0.97, dan *recall* : 0.73.

FINAL RESULT

- Berdasarkan pemodelan yang telah dilakukan dengan menggunakan KNN, *Decision Tree*, dan *Random Forest*, maka dapat disimpulkan untuk memprediksi *churn* dengan menggunakan dataset ini model terbaiknya adalah menggunakan *Random Forest*.