Rakamin Academy

Marketing Campaign

Kelompok 2 LASKAR ONLINE

Dokumen
Laporan Final
Project

Nama Anggota Kelompok

Angelus Felix Sihombing

Aisyah Raudhatuzzahra

Saip Ardo Pratama

Edhita Kristasari

Richard Noel

Marha Nur Amalina

Modeling

A. Split Train Test
   - train, test = train_test_split(df, test_size=0.2, random_state=42)
   - X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

Modeling

B. Modeling

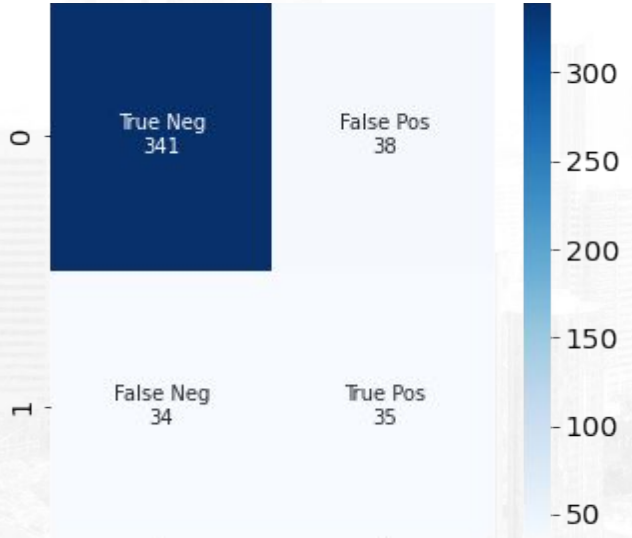
- Logistic Regression
- Decision Tree
- K-Nearest Neighbor
- Random Forest
- SVM
- Naive Bayes
- XGBoost
- AdaBoost

Modeling

C. Model Evaluation

Untuk dataset Marketing_Campaign dilakukan evaluasi dengan tujuan untuk meningkatkan kualitas dari model ML dan meminimalisir kesalahan dari hasil prediksi. Tujuannya adalah dengan meminimalisir nilai False Negative, karena ingin meningkatkan tingkat penerimaan campaign menjadi lebih baik. Oleh karena itu, metrics evaluasi yang digunakan adalah nilai ‘recall’.

Logistic Regression



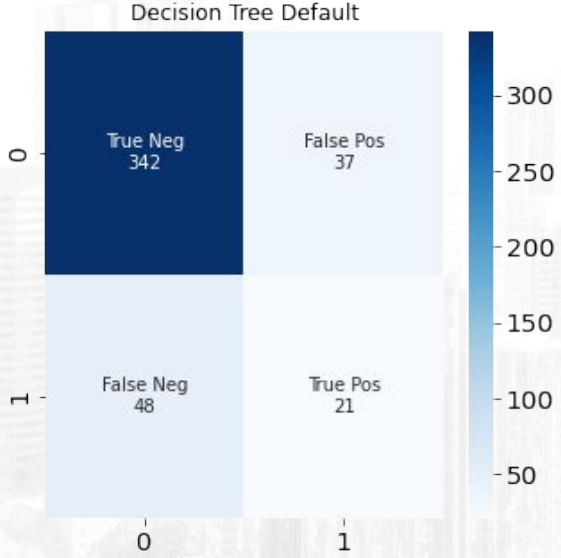
Model Evaluation	
Recall (Train)	0.79
Recall (Test)	0.7

Dalam proses eksperimen, dilakukan pengujian model Logistic Regression. Model Logistic Regression dilakukan untuk solver: lbfgs, lblinear, newton-cg, newton-cholesky, sag, dan saga. Dari beberapa kali eksperimen didapatkan hasil terbaik dengan hasil seperti tabel disamping dengan solver lbfgs, yaitu Recall (Train) : 0.79 dan Recal(Test) : 0.7. Dengan hasil confusion Matrix seperti gambar disamping yaitu False Negatif : 34 dan False Positif : 38. Hasil ini sesuai dengan tujuan model evaluasi diawal, yaitu meminimalisir False Negatif, agar meningkatkan tingkat penerimaan campaign. Didapatkan hasil yang cukup baik

Logistic Regression Hyperparameter Tuning

Pada Hyperparameter Tuning dilakukan dengan dua cara, yaitu secara otomatis dengan metode randomize search, dan tuning secara manual. Dari kedua tuning, diketahui bahwa model Logistic Regression parameter optimalnya yaitu : solver : lbfgs dan C : 1.0

Decision Tree



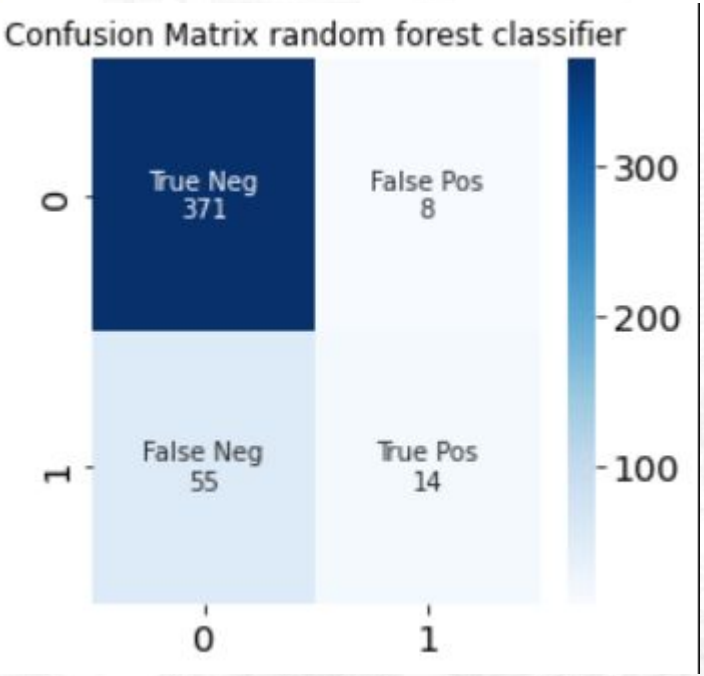
Model Evaluation	
Recall (Train)	1
Recall (Test)	0.6

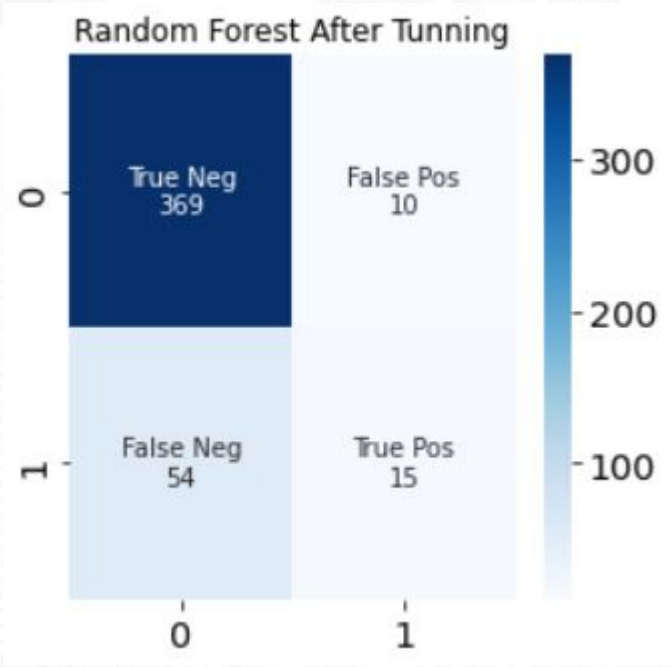
Dalam proses eksperimen, dilakukan pengujian model Decision Tree. Model Decision Tree dilakukan untuk criterion : gini, entropy, dan log_loss. Dari beberapa kali eksperimen didapatkan hasil terbaik dengan hasil seperti tabel disamping dengan solver gini, yaitu Recall (Train) : 1 dan Recal(Test) : 0.6. Dengan hasil confusion Matrix seperti gambar disamping yaitu False Negatif : 49 dan False Positif : 31. Hasil ini sesuai dengan tujuan model evaluasi diawal, yaitu meminimalisir False Negatif, agar meningkatkan tingkat penerimaan campaign. Namun hasil modelnya underfit.

Decision Tree Hyperparameter Tuning

Pada Hyperparameter Tuning dilakukan secara otomatis dengan metode randomize search. Dari tuning, diketahui bahwa model Logistic Regression parameter optimalnya yaitu :
criterion : gini (default parameter)

Random Forest





Random Forest Hyperparameter Tuning

Model Evaluasi dilakukan dengan menggunakan Random Forest . Hasil metrics evaluasi terlihat pada tabel , didapatkan kesimpulan bahwa model ini berada pada keadaan overfit

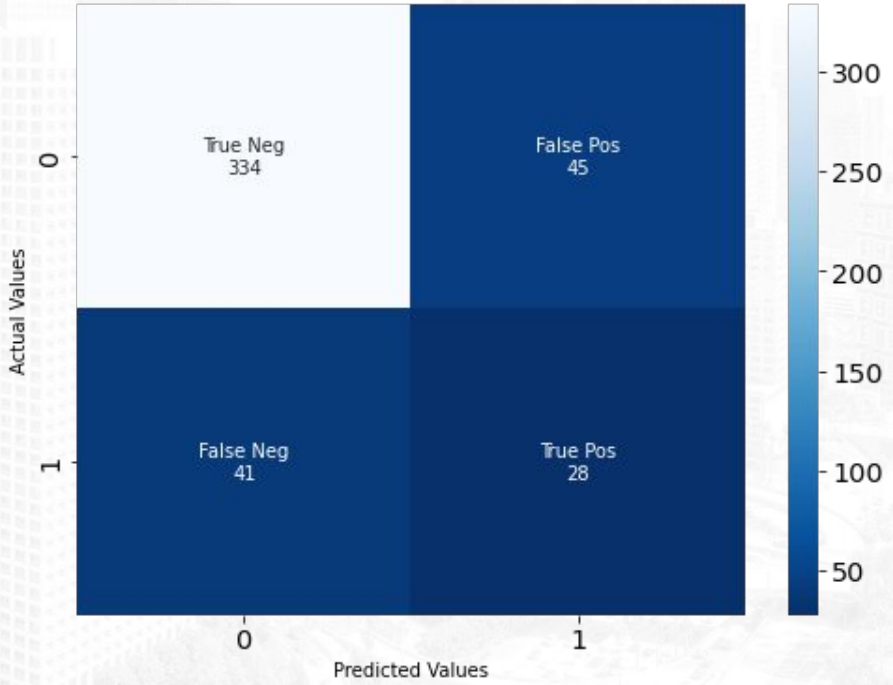
Before		After	
Recall (Train)	Recall (Test)	Recall (Train)	Recall (Test)
1.00	0.59	1.00	0.60

K-Nearest Neighbor

Pada algoritma KNN dengan parameter n_neighbor=3 didapatkan confusion matrix seperti di samping. Dan nilai recall sebagai berikut :

Recall (Train)	Recall (Test)
0.96	0.64

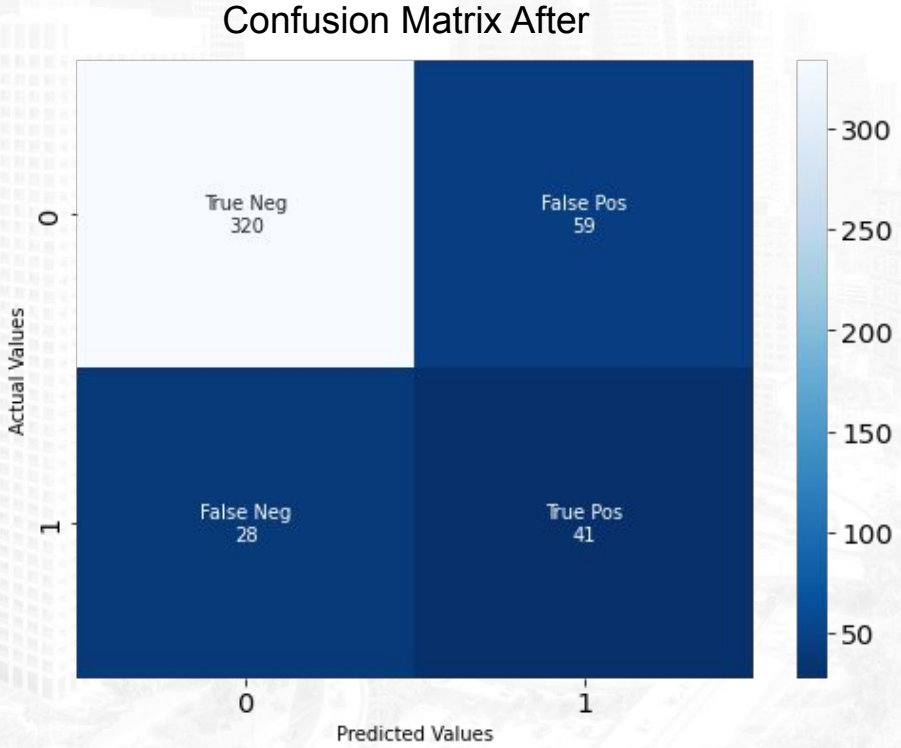
Dari score recall tersebut terdapat gap cukup jauh antara score train & test nya yaitu sekitar 0.32 (overfit)



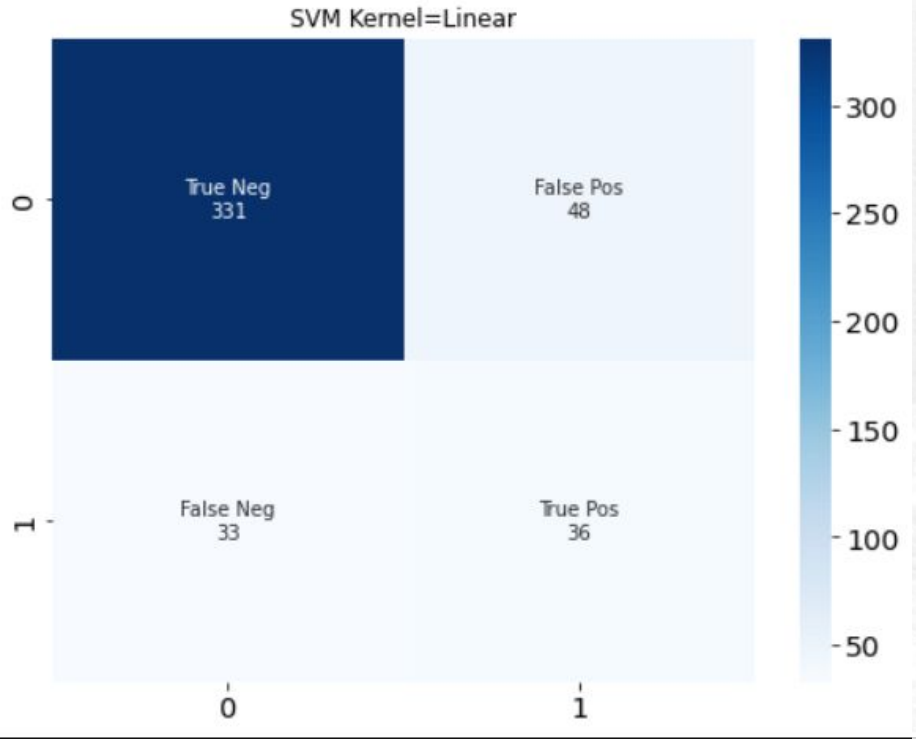
KNN Hyperparameter Tuning

Before		After	
Recall (Train)	**Recall (Test)**	**Recall (Train)**	**Recall (Test)**
0.96	0.64	0.81	0.72

Setelah mencari nilai n_neighbor terbaik antara range 1-30 ditemukain nilai n_neighbor terbaik yaitu 29. Dari tabel diatas dapat dilihat untuk Recall Train mengalami penurunan, sedangkan Recall Test mengalami kenaikan.
GAP nilai Recall train dan Recall test selisih 0.09 (bestfit).



SVM

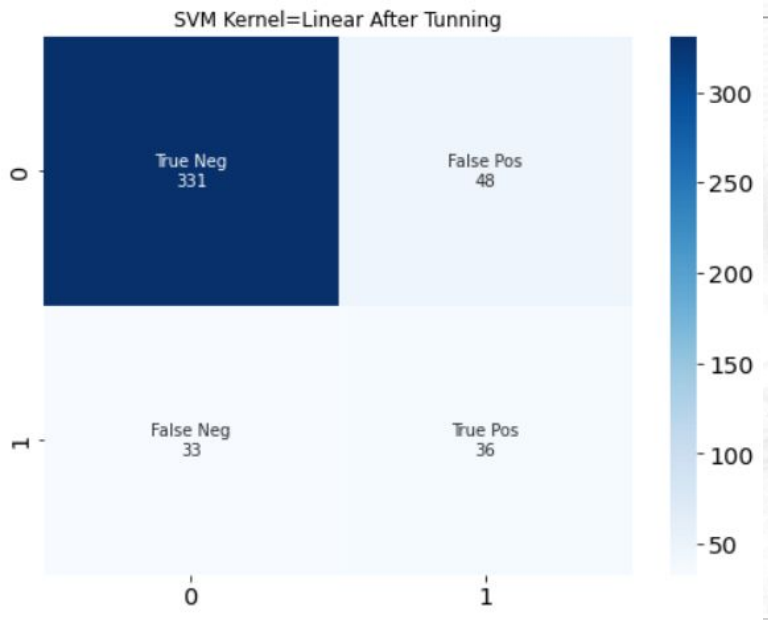


- SVM

Dalam proses eksperimen, telah dilakukan ujicoba SVM dengan beberapa parameter Kernel, ada rbf, linear, poly, sigmoid. Dari 4 parameter ini dipilih kernel = ‘linear’, karena distribusi nilai pada FN dan FP lebih merata dibandingkan 3 parameter lainnya. dilakukan hyperparameter tunning dengan menggunakan RandomSearchCV.

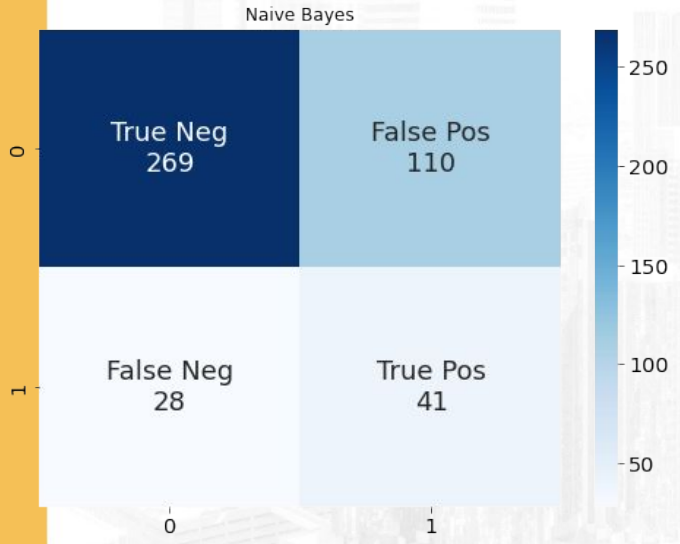
SVM Hyperparameter Tuning

Before		After	
Recall (Train)	Recall (Test)	Recall (Train)	Recall (Test)
0.8	0.7	0.8	0.7



Setelah dilakukan tunning, tidak ada perubahan yang signifikan terlihat. Karena model berada pada kondisi bestfit

Naive Bayes



Recall (Train)	Recall (Test)
0.66	0.65

Algoritma Naive Bayes digunakan sebagai eksperimen modeling untuk mencari model dengan hasil yang paling tinggi. Pada model Naive Bayes, didapatkan confusion matrix seperti di samping.

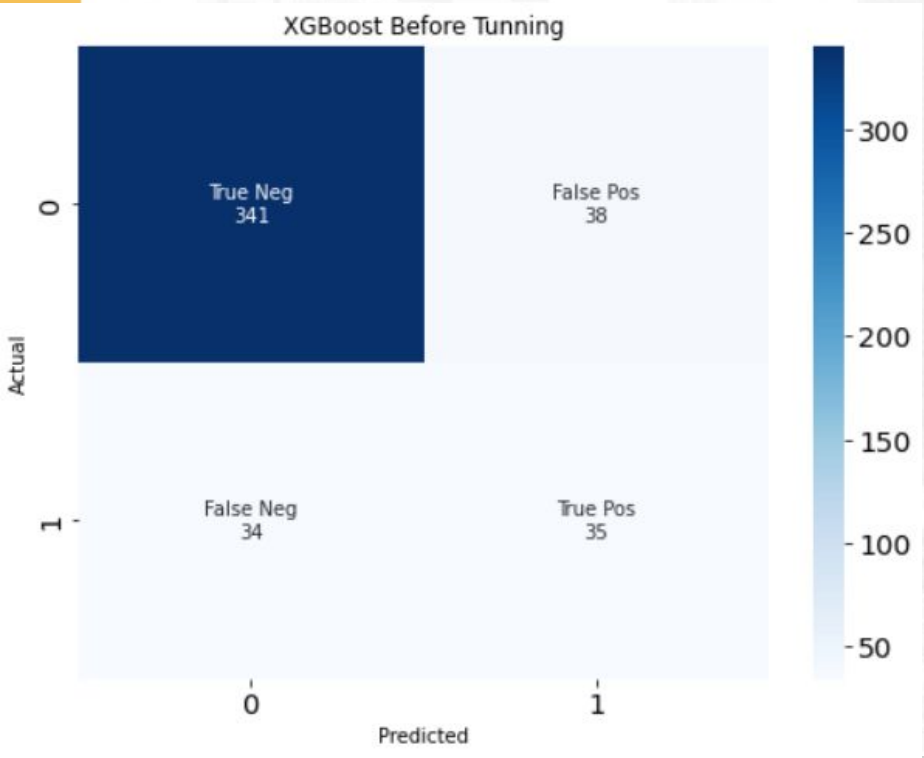
Setelah dilakukan cross-validation nilai recall antara data test dan data train, didapatkan nilai **recall (train) = 0.66** dan **recall (test) = 0.66**. Hasil evaluasi model ini memiliki jarak yang cukup baik masuk dalam kategori **tidak memiliki performa baik**.

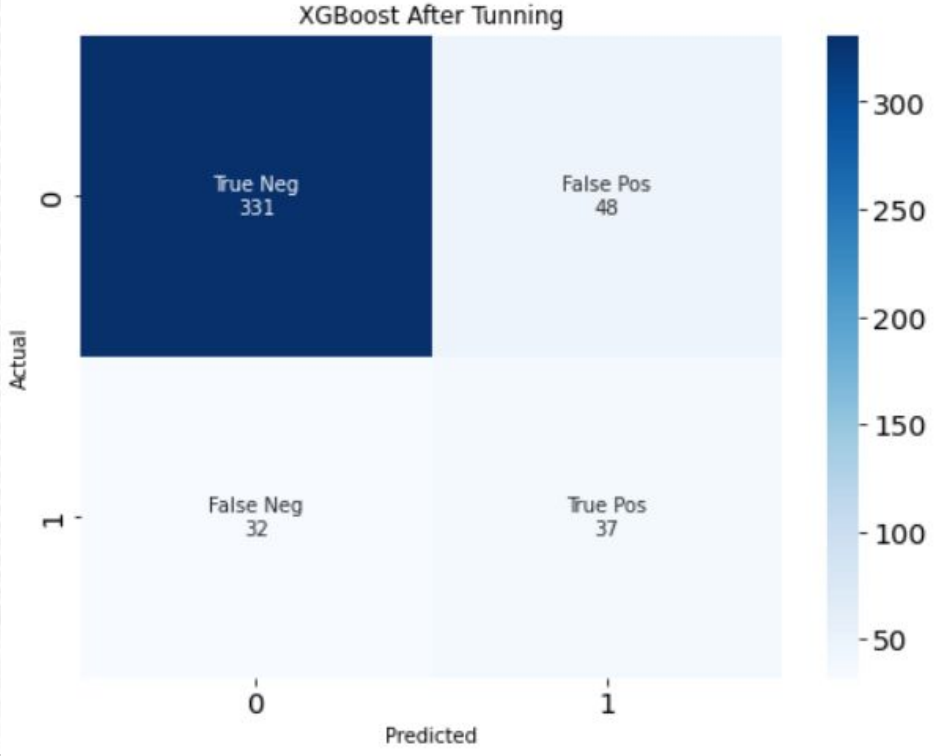
Naive Bayes Hyperparameter Tuning

Before		After	
Recall (Train)	Recall (Test)	Recall (Train)	Recall (Test)
0.66	0.65	0.476	0.449

Untuk meningkatkan performa model Naive Bayes, dilakukan hyperparameter tuning dengan hyperparameter **var_smoothing**. Namun dari hasil eksperimen hyperparameter tuning ini didapatkan nilai **recall (train) dan recall (test) yang semakin menurun** meskipun jarak keduanya menjadi lebih kecil, yang berarti bahwa hyperparameter tuning **tidak** meningkatkan performa model.

XGBoost


XGBoost Before Tunning
True Neg
341
False Pos
38
False Neg
34
True Pos
35
Actual
0
1
Predicted
0
1
300
250
200
150
100
50

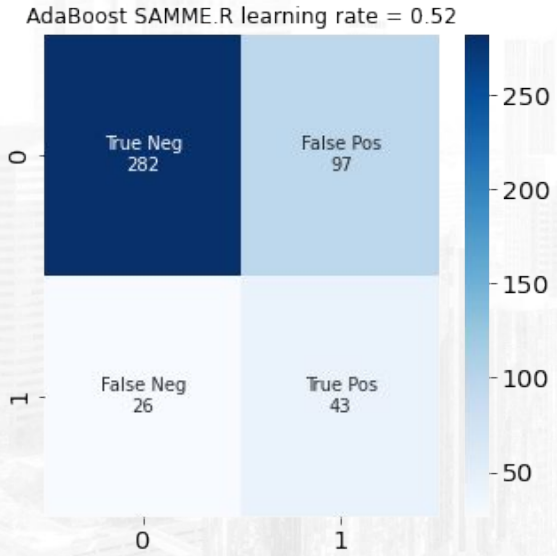

XGBoost After Tunning
True Neg
331
False Pos
48
False Neg
32
True Pos
37
Actual
0
1
Predicted
0
1
300
250
200
150
100
50

XGBoost Hyperparameter Tuning

Before		After	
Recall (Train)	**Recall (Test)**	**Recall (Train)**	**Recall (Test)**
0.79	0.70	0.8	0.7

Xgboost merupakan salah satu algoritma machine learning yang dapat digunakan pada dataset supervised learning. Hasil ujicoba menggunakan Xgboost telah dilakukan dengan menggunakan fitur product, purchase, recency (16 fitur). Dari hasil tersebut didapatkan bahwa algoritma ini menghasilkan model yang bestfit, selanjutnya dilakukan tunning hyperparamater. Namun yang di dapatkan tidak ada perubahan yang signifikan karena model berada pada kondisi Bestfit

AdaBoost



Model Evaluation	
Recall (Train)	0.68
Recall (Test)	0.77

Dalam proses eksperimen, dilakukan pengujian model AdaBoost. Model AdaBoost dilakukan untuk algoritma : Samme dan Samme.R. Dari beberapa kali eksperimen didapatkan hasil terbaik dengan hasil seperti tabel disamping, yaitu Recall (Train) : 0.68 dan Recall (Test) : 0.77. Dengan hasil confusion Matrix seperti gambar disamping yaitu False Negatif : 26 dan False Positif : 97. Hasil ini sesuai dengan tujuan model evaluasi di awal, yaitu meminimalisir False Negatif, agar meningkatkan tingkat penerimaan campaign. Namun hasil yang didapatkan masih kurang maksimal

AdaBoost Hyperparameter Tuning

Pada Hyperparameter Tuning dilakukan secara otomatis dengan metode randomize search. Dari tuning, didapatkan hasil yang terbaik yaitu dengan algoritma Samme.R dengan learning rate : 0.52

Model Evaluation Results

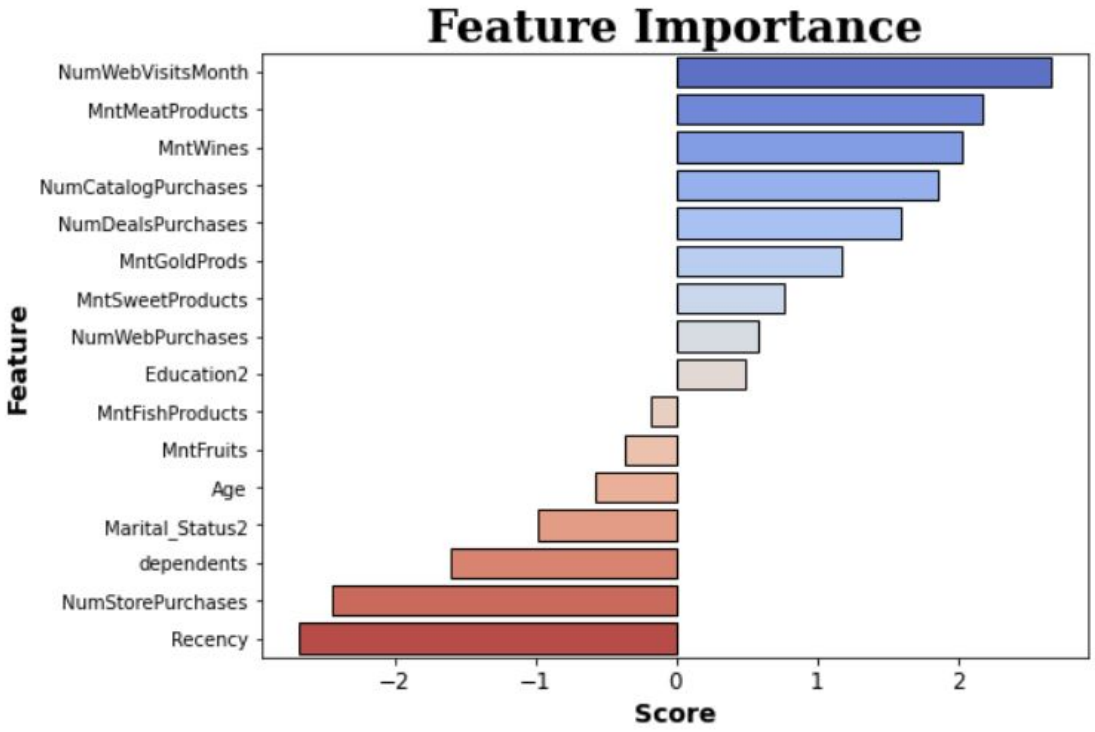
	Recall (Train)	Recall (Test)
Logistic Regression		
Decision Tree		
SVM	0.80	0.70
K-Nearest Neighbor	0.81	0.72
AdaBoost	0.847	0.841
XGBoost	0.79	0.70
Random Forest	1.00	0.60
Naive Bayes	0.66	0.65

Modeling Summary

Dari hasil pemodelan dengan 8 algoritma machine learning, didapatkan hasil yang beragam. Karena model evaluation model yang kami ajukan berfokus pada meningkatkan penerimaan campaign yaitu dengan meningkatkan hasil Recall, maka model yang terbaik dari hasil pemodelan kami yaitu : Logistic Regression

Feature Importance

Berdasarkan hasil eksperimen berbagai algoritma, **Logistic Regression** memiliki performa model terbaik. Sehingga pada data ini akan digunakan model **Logistic Regression**. Berikut adalah feature importance dari model Logistic Regression.



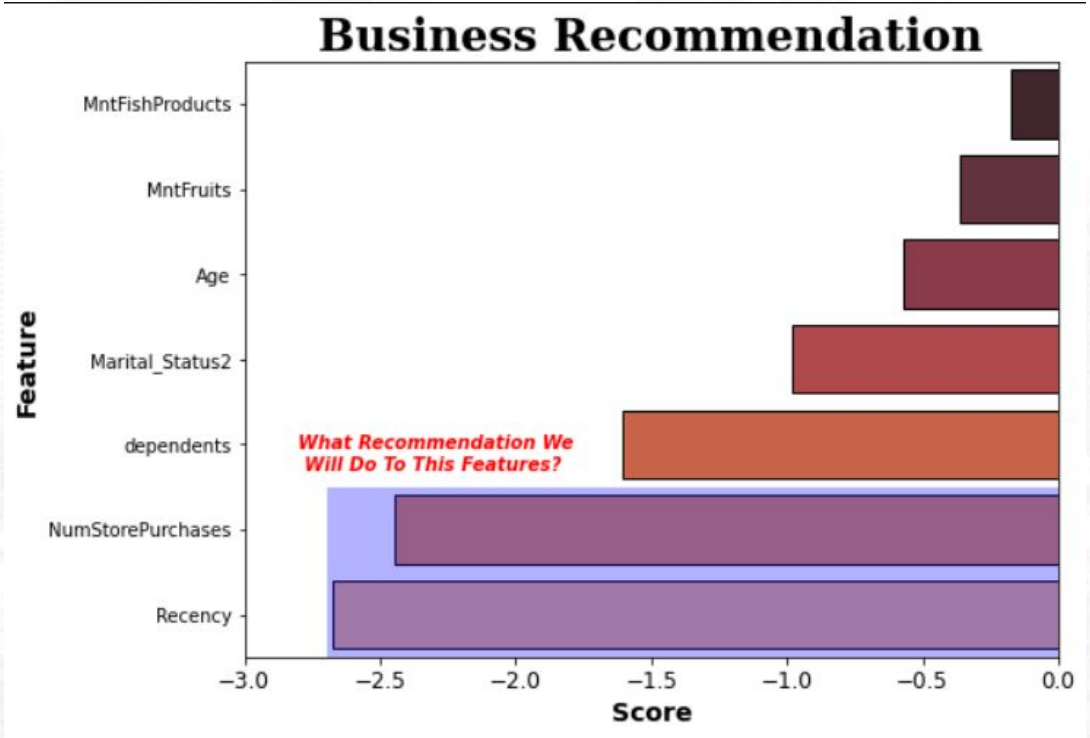
Business Insight

Dari hasil feature importance, insight yang dapat diambil adalah:

- Customer yang lebih sering melakukan pembelanjaan melalui katalog memiliki potensi lebih besar untuk menerima campaign
- Produk yang paling banyak dibeli oleh customer penerima campaign sebelumnya adalah daging
- Semakin sering customer melihat website perusahaan, semakin tinggi potensi customer tersebut untuk menerima campaign
- Semakin tinggi jumlah pembelian customer melalui store mengurangi potensi customer tersebut untuk merespon campaign
- Customer aktif lebih berpotensi untuk merespon campaign yang ditawarkan
- Semakin jauh jarak pembelian terakhir customer, maka akan semakin berkurang minat customer tersebut untuk merespon campaign

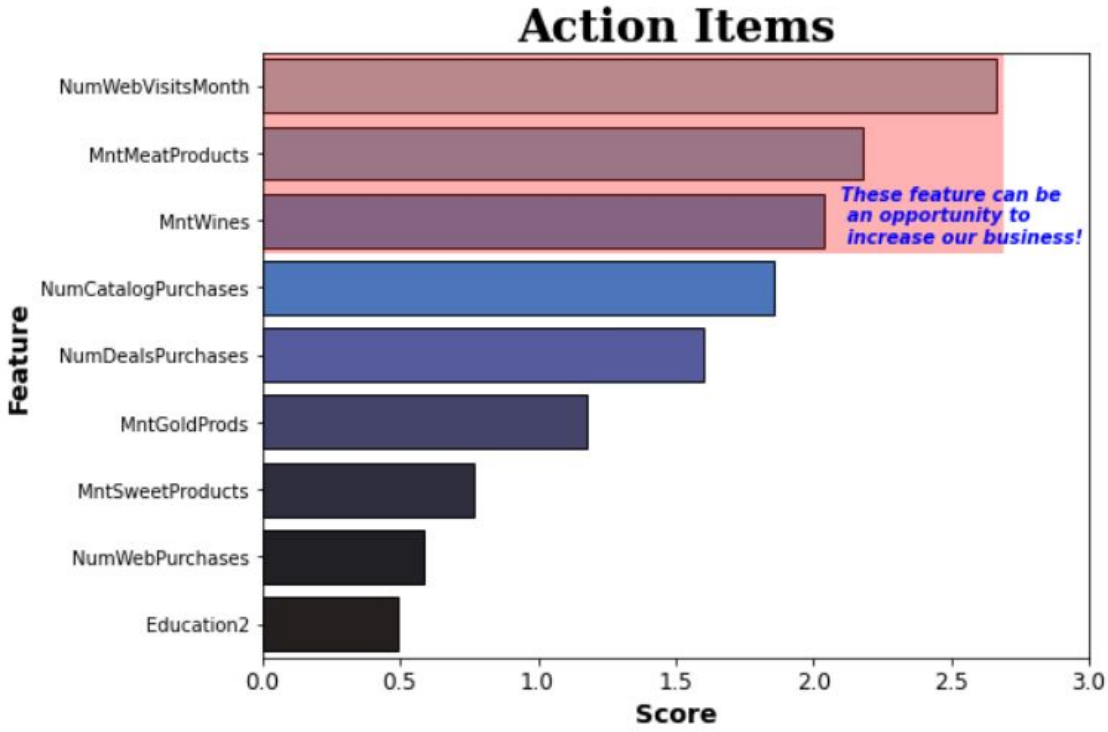
Business Recommendation

- Menawarkan campaign terhadap customer yang sering melihat website perusahaan dan membeli produk daging melalui katalog
- Memberikan penawaran khusus untuk setiap pembelian melalui store untuk meningkatkan penjualan store
- Melakukan remarketing kepada customer yang sudah lama tidak melakukan pembelian



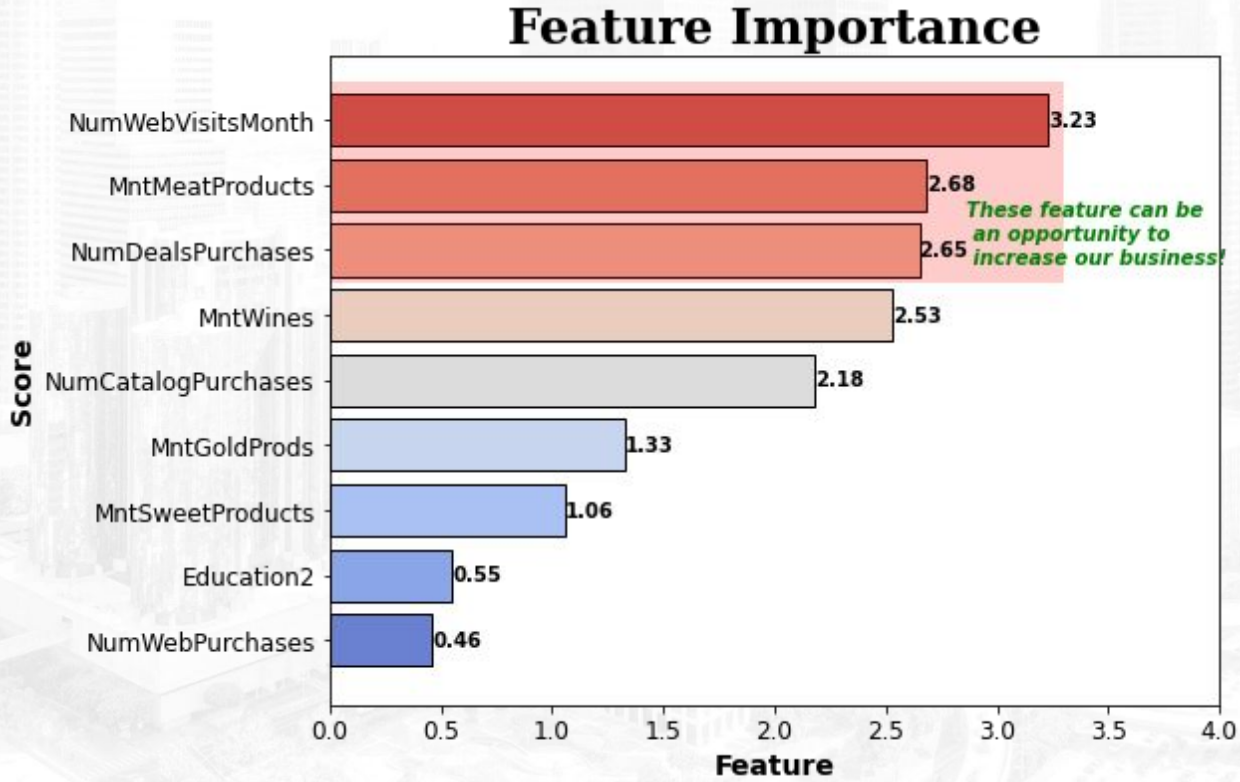
Action Items

- Merekomendasikan untuk melakukan pengubahan tampilan web dengan tujuan meningkatkan '*interest*' pengunjung/ calon customer
- Memberikan saran untuk melakukan pengiklanan terhadap produk daging dan anggur, dan meyakinkan calon pembeli bahwa barang akan sesuai dengan iklan

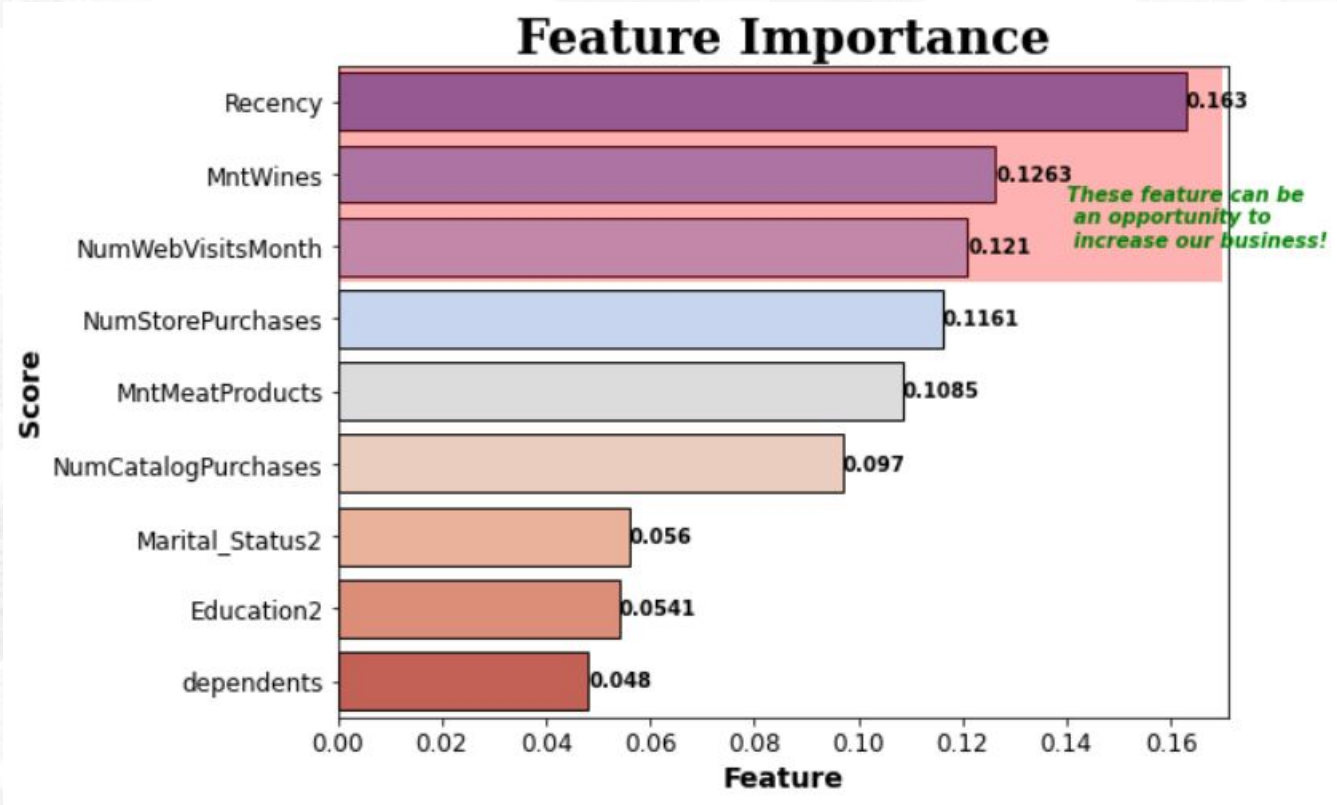


Back-Up

Additional Feature Importance: XGBoost



Additional Feature Importance: SVM



Additional Feature Importance: Random Forest