

# Intellegend Ecommerce Churn Prediction

Dokumen  
Laporan Final  
Project

(dipresentasikan setiap sesi mentoring)



# Modeling

**Split Data Train & Test:** Split data train dan test sudah dilakukan pada tahap preprocessing untuk menghindari terjadinya data leak, karena data test adalah unseen data.

**Modeling:** Model yang digunakan adalah

- Logistic Regression (Baseline)
- Decision Tree
- KNN
- SVM
- **Random Forest (best model)** Model ini untuk sementara dipilih karena memiliki score precision dan ROC\_AUC yang paling tinggi dibanding model lainnya.
- Gradient Boosting
- XGBoost
- AdaBoost
- CatBoost

# Modeling

## Model Evaluation:

Pada kasus Ecommerce churn prediction ini, kami menggunakan Recall sebagai metrics utama karena tujuan utama dari prediksi model ini adalah mencegah customer untuk churn jika ia terdeteksi churn, maka dari itu dengan metrics recall prediksi kami dapat berfokus sebanyak-banyaknya customer yang berpotensi untuk churn untuk mencegah mereka untuk churn.

Customer yang diprediksi akan churn akan diberikan kupon atau penawaran spesial agar mereka tidak churn, namun karena dataset ini mempunyai target yang imbalance, untuk mencegah memberikan terlalu banyak kupon kepada customer yang tidak berpotensi churn maka kami akan memakai metrics ROC\_AUC sebagai tambahan.



# Modeling

## **Model Evaluation:**

Walaupun model yang sudah di train sudah melewati baseline model yang berarti model sudah cukup baik, namun Model belum best-fit karena masih dapat ditemukan model yang lebih baik lagi dengan mencoba menggunakan feature-feature lain pada data yang terkena drop pada tahap preprocess (stage 2), serta dapat mentuning hyperparameter dengan lebih baik lagi

# Modeling

**Hyperparameter Tuning:** Terdapat beberapa hyperparameter yang digunakan untuk mentuning masing-masing model, namun hyperparameter yang digunakan untuk mentuning best model adalah:

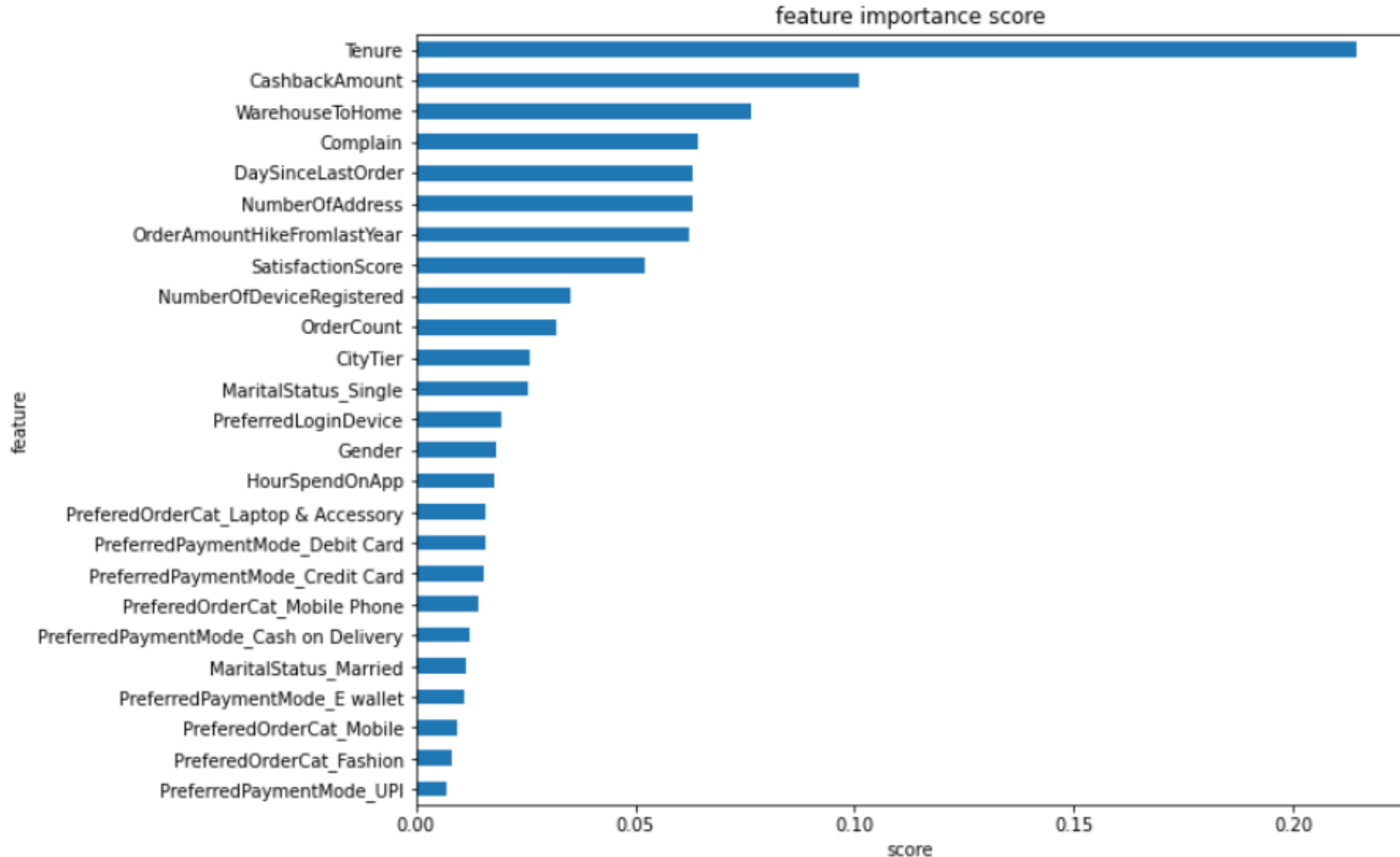
- `n_estimators` : untuk mengatur jumlah tree dalam randomForest model
- `min_samples_split` : minimum sample yang diperlukan untuk melakukan split pada suatu node
- `min_samples_leaf`: minimum sample yang diperlukan dalam suatu leaf, jika kurang dari `min_samples_leaf` maka leaf akan di prune
- `max_features`: maximum feature yang digunakan untuk splitting node pada tree
- `max_depth`: maximum kedalaman yang dapat dibentuk oleh tree
- `bootstrap`: jika true maka tiap tree akan dibuat dengan sample data, jika false semua data akan digunakan untuk semua tree

# Modeling

Eksperimen yang telah dilakukan adalah mencoba train semua model yang belum di tuning hyperparameternya lalu membandingkan score mereka, lalu mencoba train semua model sambil mentuning hyperparameternya lalu membandingkan score mereka. Selain itu mencoba feature scaling dari feature importances yang telah didapat. Dari semua percobaan, untuk saat ini model yang terbaik adalah model RandomForest.

Selain itu setelah mencoba melakukan segala cara pada dataset yang ada, namun score model masih belum memuaskan, kami memutuskan untuk kembali ke stage 1 dan merubah feature selection dan remove outlier, dan setelah melakukan hal tersebut, kami mencoba untuk modelling Kembali dan tuning hyperparameter. Hasilnya adalah score model yang kami dapatkan lebih baik dari dataset yang sebelumnya.

# Feature Importances



# Feature Importances

**Feature yang mempunyai pengaruh besar adalah:**

- Tenure
- CashbackAmount
- WarehouseToHome
- Complain
- DaySinceLastOrder
- NumberOfAddress
- OrderAmountHikeFromlastYear
- SatisfactionScore
- NumberOfDeviceRegistered
- OrderCount
- CityTier
- MaritalStatus\_Single



# Feature Importances

Melihat dari feature-feature yang penting diatas dapat ditarik kesimpulan bahwa, untuk mencegah terjadinya customer churn, pelanggan butuh diberikan Cashback yang cukup besar sampai setidaknya 10 bulan (10 bulan disini merupakan median dimana customer diprediksi tidak churn, customer diprediksi churn jika tenure dibawah 1 bulan). Hal ini dapat kita tarik sebagai kesimpulan karena terlihat CashbackAmount dan Tenure merupakan feature yang paling berpengaruh pada churn customer.

Selain itu jumlah warehouse juga perlu diperbanyak di tempat yang memiliki banyak customer, karena warehouse yang terlalu jauh menyebabkan customer churn. Dan yang terakhir adalah harus menjaga DaySinceLastOrder yang stabil, jika customer terlalu sering atau terlalu jarang menggunakan Ecommerce ini, maka customer tersebut akan berpotensi churn karena itulah customer perlu diberikan voucher mingguan yang membuat belanja mereka teratur setiap minggunya. Pembagian voucher ini juga dapat menaikkan OrderAmountHikeFromlastYear yang dapat mencegah terjadinya churn. Dan yang terakhir adalah Complain, jika customer memiliki complain harus diselesaikan dengan segera, karena complain yang tidak diselesaikan dapat menyebabkan customer churn dan menurunkan SatisfactionScore yang juga berdampak pada meningkatnya kecenderungan customer untuk churn.

# Feature Importances

Sebelum feature selection

```
model = RandomForestClassifier(bootstrap = False, max_depth = None, max_features = 'sqrt', min_samples_leaf = 1, min_samples_split = 2)
model.fit(X_train, y_train)
ypred = model.predict(X_test)
print(f'Recall Score = {recall_score(y_test, ypred)}')
print(f'ROC_AUC Score = {roc_auc_score(y_test, ypred)}')
```

Recall Score = 0.8876404494382022  
ROC\_AUC Score = 0.9399313358302123

Setelah feature selection

```
model = RandomForestClassifier(bootstrap = False, max_depth = None, max_features = 'sqrt', min_samples_leaf = 1, min_samples_split = 2)
model.fit(X_train_selection, y_train)
ypred = model.predict(X_test_selection)
print(f'Recall Score = {recall_score(y_test, ypred)}')
print(f'ROC_AUC Score = {roc_auc_score(y_test, ypred)}')
```

Recall Score = 0.8876404494382022  
ROC\_AUC Score = 0.9399313358302123

# Feature Importances

Dapat dilihat bahwa feature selection tidak berefek kepada score model, namun di satu sisi dimensi menjadi berkurang yang menyebabkan waktu train model menjadi lebih singkat.

# GIT

Git repository: <https://github.com/BryanT05/Intellegend-Final-Project>