Home Credit

HCMC 8/2020

Outlook

I. General View

1. Data
2. Missing values
3. Correlation

II. Detail View

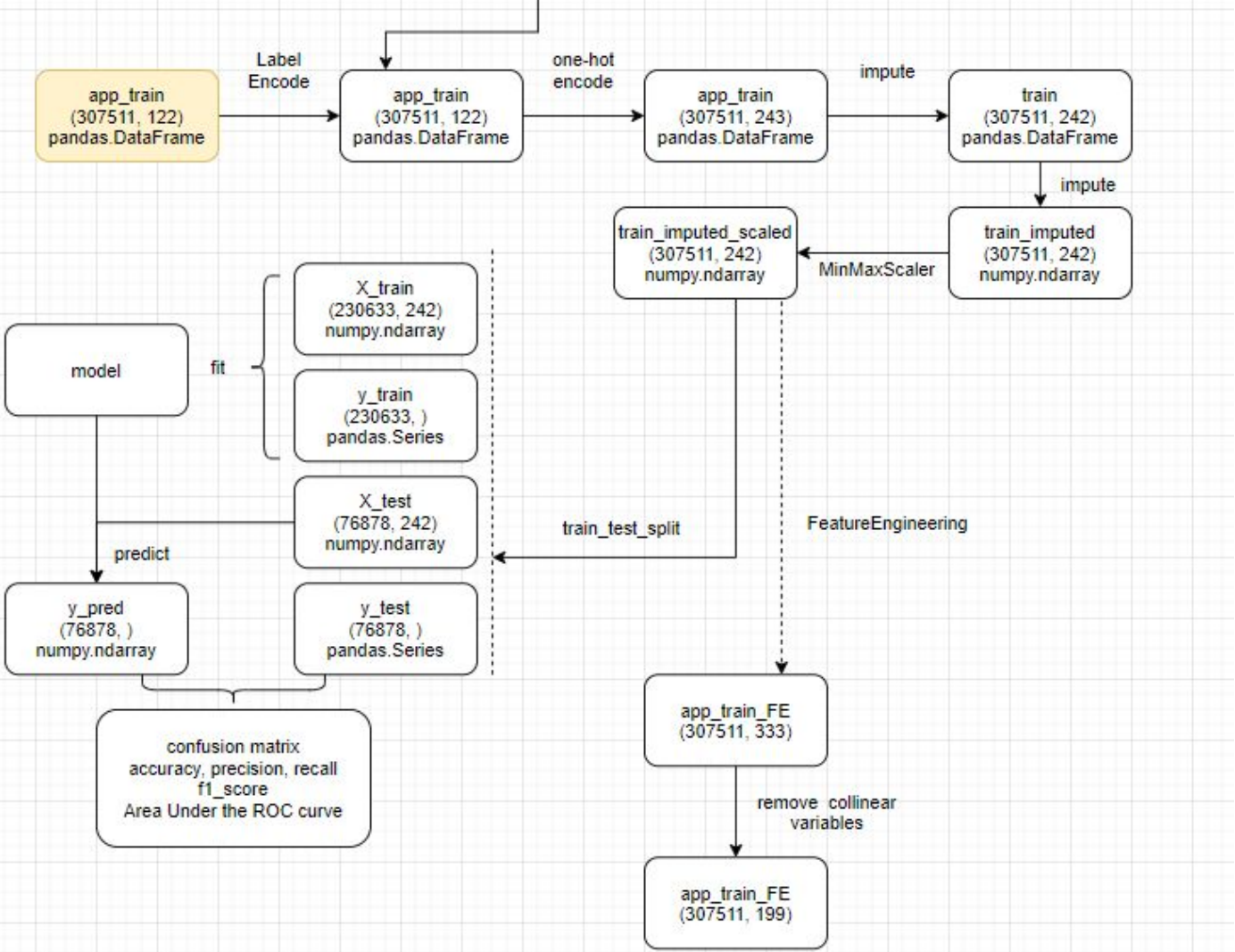
1. Histogram
2. KDE

III. Models

1. Logistic Regression
2. Random Forest
3. Conclusion

IV. Models (12 features)

1. Models


app_train
(307511, 122)
pandas.DataFrame
Label
Encode
app_train
(307511, 122)
pandas.DataFrame
one-hot
encode
app_train
(307511, 243)
pandas.DataFrame
impute
train
(307511, 242)
pandas.DataFrame
impute
train_imputed
(307511, 242)
numpy.ndarray
MinMaxScaler
train_imputed_scaled
(307511, 242)
numpy.ndarray
model
fit
X_train
(230633, 242)
numpy.ndarray
y_train
(230633,)
pandas.Series
X_test
(76878, 242)
numpy.ndarray
train_test_split
FeatureEngineering
predict
y_pred
(76878,)
numpy.ndarray
y_test
(76878,)
pandas.Series
confusion matrix
accuracy, precision, recall
f1_score
Area Under the ROC curve
app_train_FE
(307511, 333)
remove collinear
variables
app_train_FE
(307511, 199)

I. General View

1. Data

a. Dữ liệu

- Home Credit về các thông tin của khách hàng đã trả được khoản vay hoặc chưa
- 307.511 quan sát (là từng khoản vay khác nhau) và 122 đặc trưng bao gồm biến TARGET (dùng để dự báo). Trong đó có 65 biến kiểu dữ liệu số thực, 41 biến kiểu số nguyên, 16 biến kiểu dữ liệu categorical
- Biến phụ thuộc Y là TARGET - biến cần đi dự báo đối với dataset này:

  Y = 0: khoản vay được trả đúng hạn - có 282.686 khoản - chiếm 91,9%

  Y = 1: khoản vay chưa được trả đúng hạn - 24.825 khoản - chiếm 8,07%

  => *Mẫu có hiện tượng mất cân bằng =>có thể sử dụng đến các chỉ số thay thế như F1-Score hoặc AUC*

```
float64    65
int64      41
object     16
dtype: int64
```

```
Training data shape :  (307511, 122)
```

	SK_ID_CURR	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	NAME_TYPE_SUITE	NAME_INCOME_TYPE	NAME_EDUCATION_TYPE	NAME_FAMILY_STATUS	NAM
0	100001	Cash loans	F	N	Y	0	135000.0	568800.0	20560.5	450000.0	Unaccompanied	Working	Higher education	Married	H
1	100005	Cash loans	M	N	Y	0	99000.0	222768.0	17370.0	180000.0	Unaccompanied	Working	Secondary / secondary special	Married	H
2	100013	Cash loans	M	Y	Y	0	202500.0	663264.0	69777.0	630000.0	NaN	Working	Higher education	Married	H
3	100028	Cash loans	F	N	Y	2	315000.0	1575000.0	49018.5	1575000.0	Unaccompanied	Working	Secondary / secondary special	Married	H
4	100038	Cash loans	M	Y	N	1	180000.0	625500.0	32067.0	625500.0	Unaccompanied	Working	Secondary / secondary special	Married	H

5 rows × 121 columns

I. General View

1. Data

b. Định nghĩa một vài biến

- EXT_SOURCE 1,2,3: điểm chuẩn hóa từ các nguồn dữ liệu bên ngoài, nằm trong khoảng 0-1
- DAYS_BIRTH: Tuổi của khách hàng tại thời điểm vay
- REGION_RATING_CLIENT_W_CITY: Xếp hạng khu vực nơi khách hàng sinh sống xét đến thành phố(1,2,3)
- REGION_RATING_CLIENT: Xếp hạng khu vực nơi khách hàng sinh sống (1,2,3)
- NAME_INCOME_TYPE: Loại thu nhập của khách hàng: doanh nhân, nội trợ,...
- DAYS_LAST_PHONE_CHANGE: khách hàng đổi số điện thoại bao nhiêu ngày trước khi vay
- Code_gender: Giới tính
- NAME_EDUCATION_TYPE: Trình độ giáo dục

2. Missing Value

- 67 cột chứa missing values
- Cách xử lý:
  - Biến number: các missing values sẽ thay bằng giá trị trung bình
  - Biến categorical: sử dụng Label encoding và one-hot encoding để quy sang 0,1, rồi làm như biến number

Tệp dữ liệu gồm 122 cột.
Có 67 cột có giá trị bị thiếu.

	Tổng số giá trị thiếu	% giá trị thiếu
COMMONAREA_MEDI	214865	69.9
COMMONAREA_AVG	214865	69.9
COMMONAREA_MODE	214865	69.9
NONLIVINGAPARTMENTS_MEDI	213514	69.4
NONLIVINGAPARTMENTS_MODE	213514	69.4
NONLIVINGAPARTMENTS_AVG	213514	69.4
FONDKAPREMONT_MODE	210295	68.4
LIVINGAPARTMENTS_MODE	210199	68.4
LIVINGAPARTMENTS_MEDI	210199	68.4
LIVINGAPARTMENTS_AVG	210199	68.4

3. Correlation

- Một cách để nắm rõ dữ liệu là tìm mối tương quan giữa các biến X và Y (ở đây là TARGET). Hệ số tương quan không phải là phương pháp tốt nhất để biểu thị "mức độ liên quan" của X, nhưng nó cho ý tưởng về các mối quan hệ có thể có trong dữ liệu

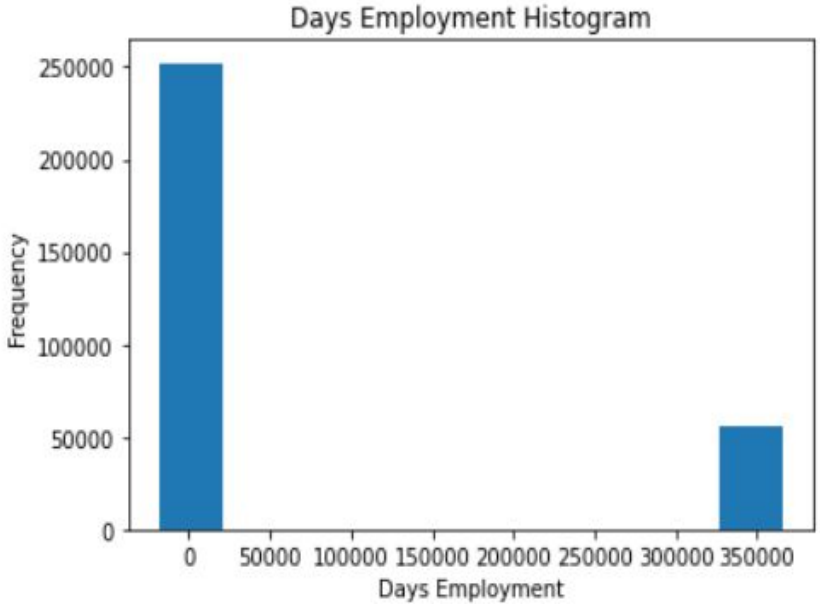
```
Most Positive Correlations:
 NAME_EDUCATION_TYPE_Secondary / secondary special    0.049824
REG_CITY_NOT_WORK_CITY                                0.050994
DAYS_ID_PUBLISH                                       0.051457
CODE_GENDER_M                                         0.054713
DAYS_LAST_PHONE_CHANGE                                0.055218
NAME_INCOME_TYPE_Working                              0.057481
REGION_RATING_CLIENT                                  0.058899
REGION_RATING_CLIENT_W_CITY                           0.060893
DAYS_BIRTH                                            0.078239
TARGET                                                1.000000
Name: TARGET, dtype: float64

Most Negative Correlations:
 EXT_SOURCE_3                                -0.178919
EXT_SOURCE_2                                 -0.160472
EXT_SOURCE_1                                 -0.155317
NAME_EDUCATION_TYPE_Higher education         -0.056593
CODE_GENDER_F                                -0.054704
NAME_INCOME_TYPE_Pensioner                   -0.046209
ORGANIZATION_TYPE_XNA                        -0.045987
DAYS_EMPLOYED                                -0.044932
FLOORSMAX_AVG                                -0.044003
FLOORSMAX_MEDI                               -0.043768
Name: TARGET, dtype: float64
```

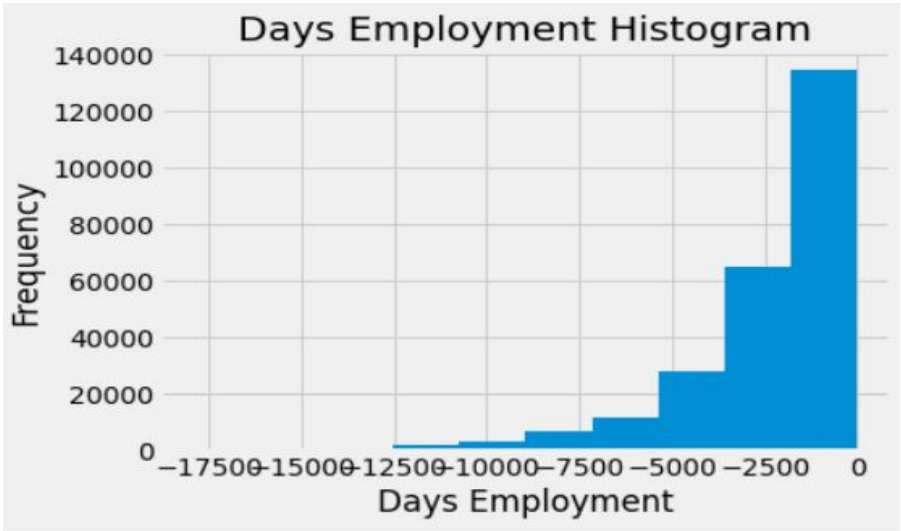
I. Detail View

1. Histogram

- Một dạng biểu đồ thể hiện tần suất dạng cột
- Hình thái phân bố của dữ liệu qua đó thiết lập mục tiêu và xu hướng khắc phục cho từng vấn đề

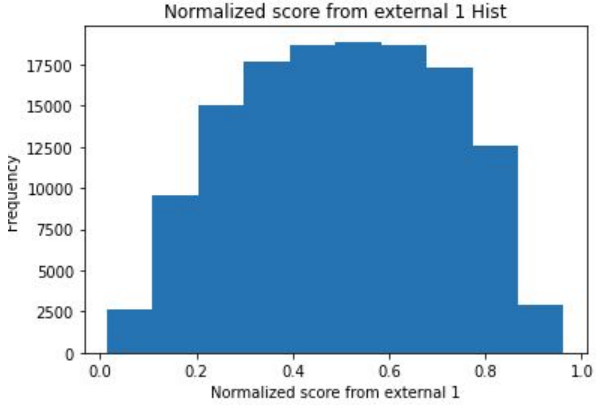
- VD: DAYS_EMPLOYED, nghi ngờ có outlier vì days employment =350000 tương đương gần 1000 năm

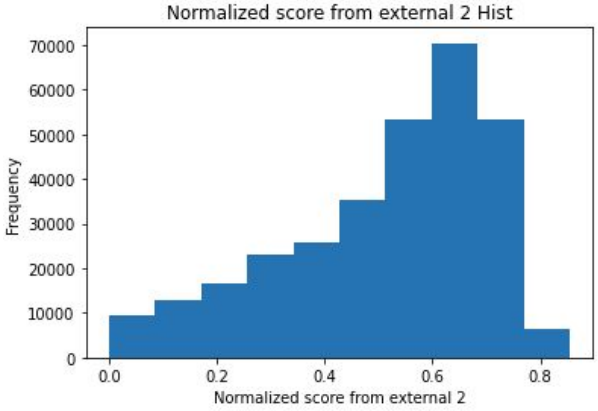


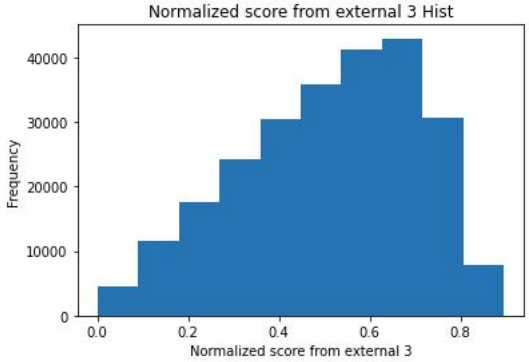


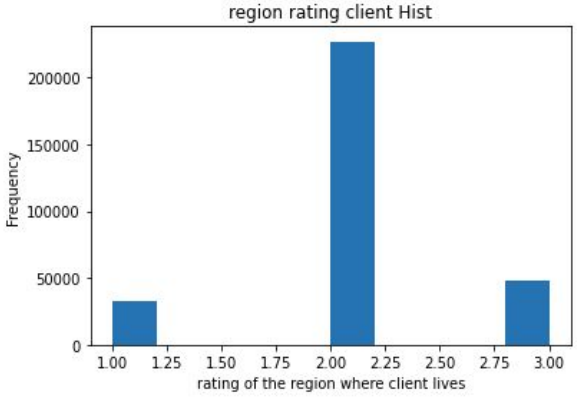
1. Histogram

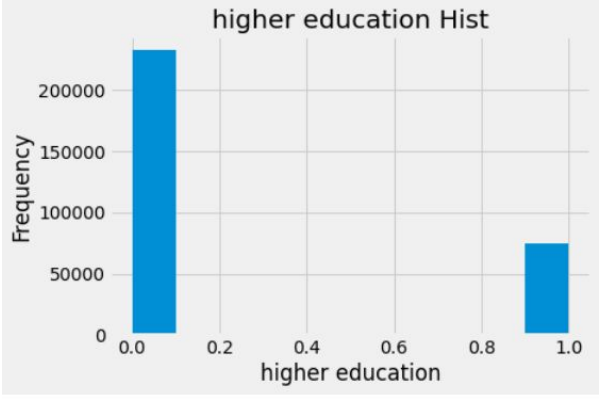
- Histogram của các biến còn lại

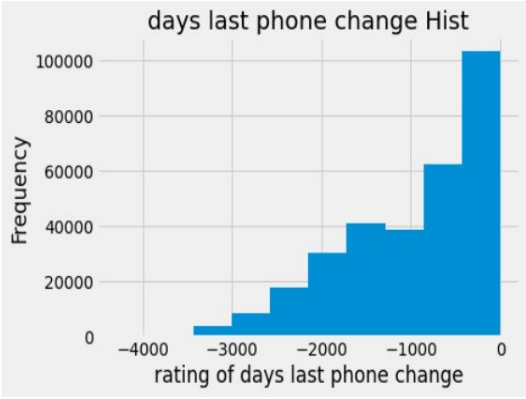






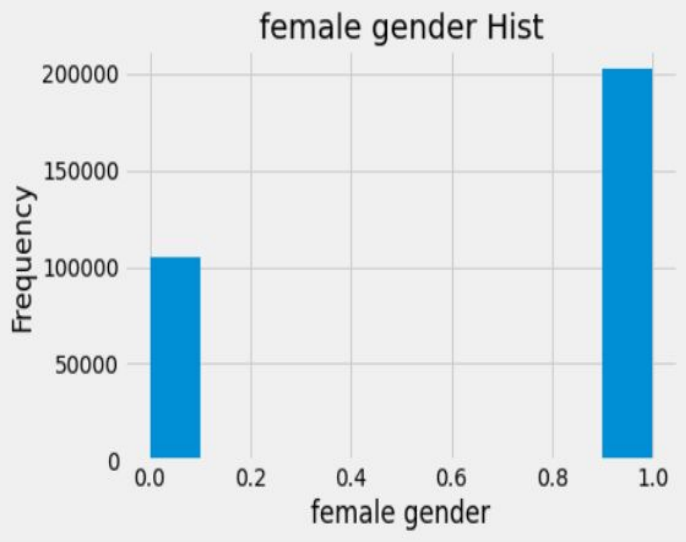






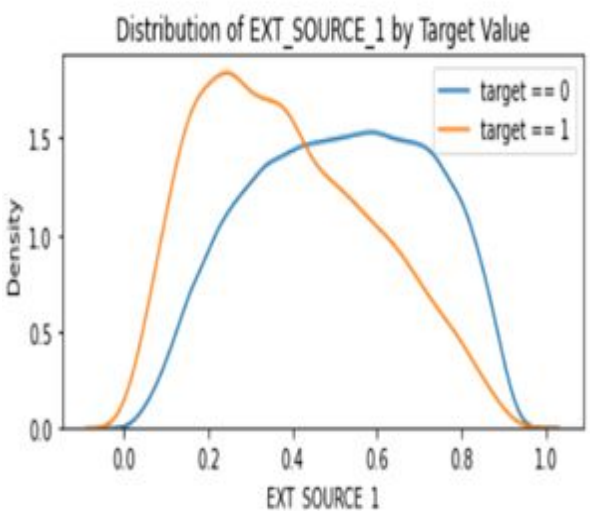
1. Histogram

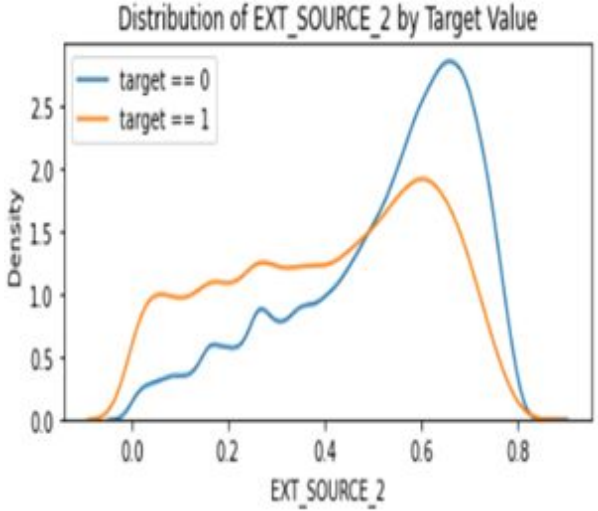
- Histogram của các biến còn lại

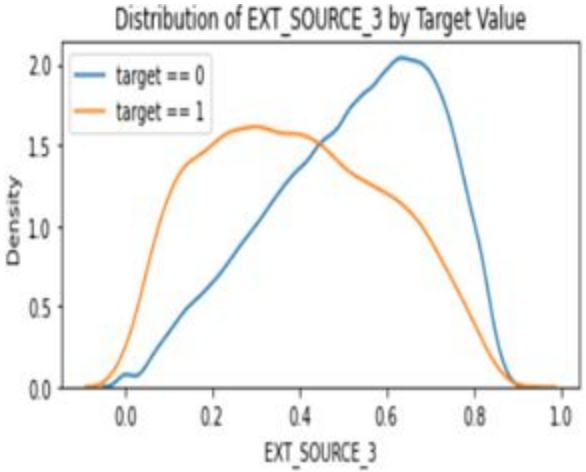


2. KDE

- Kernel density estimate: là một phương pháp không tham số để ước tính hàm mật độ xác suất (PDF) của một biến ngẫu nhiên liên tục



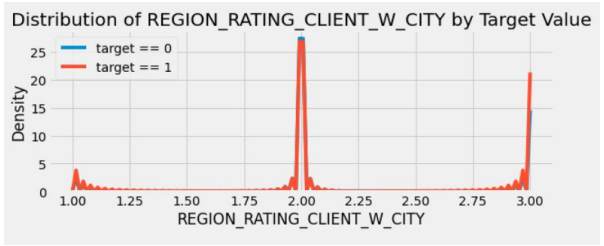


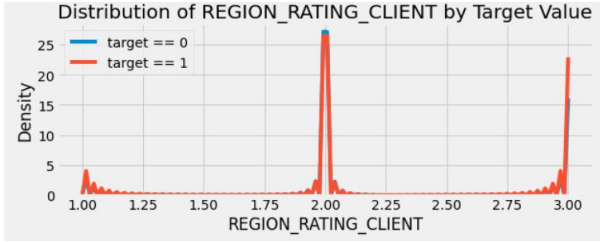


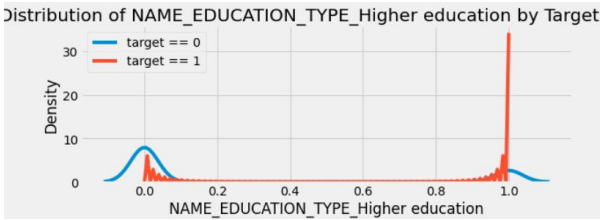
-Trong 3 nhóm EXT_SOURCE thì EXT_SOURCE_1 có tác dụng phân lớp rõ ràng nhất đối với TARGET

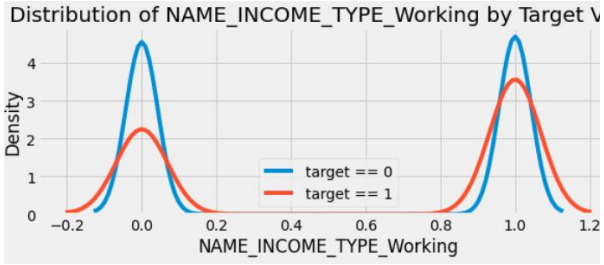
2. KDE

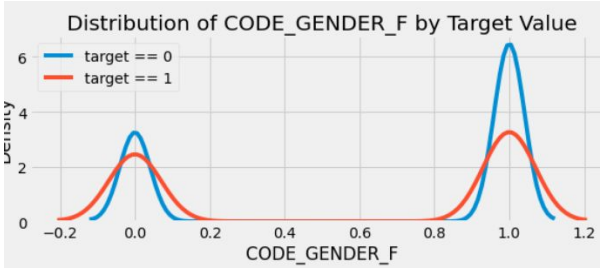
- KDE của một số biến khác

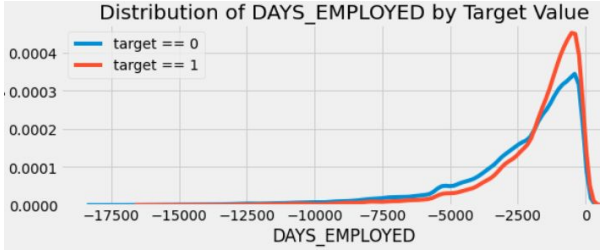


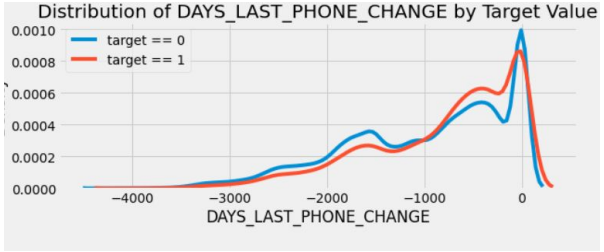


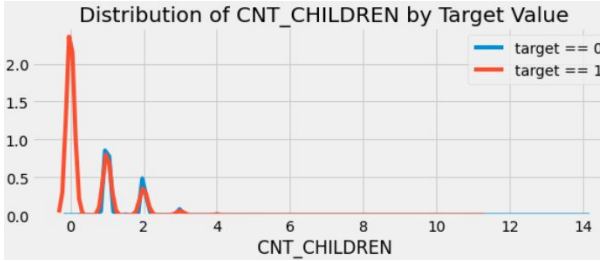












'NAME_INCOME_TYPE_Working' và 'CODE_GENDER_F' có sự phân hóa rõ ràng đối với TARGET trong các biến này.

III. Models

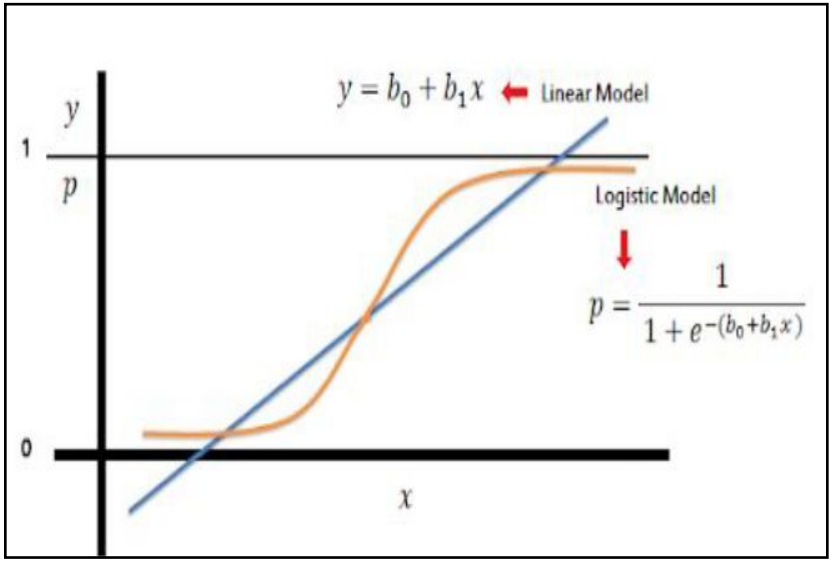
Ý tưởng: Tệp train sẽ được tách thành 2 tệp là X_train và X_test với tỷ lệ 75/25. Sau khi dùng training dữ liệu trên tệp X_train, ta sẽ được một model, áp dụng model này vào X_test để được y_pred. Sau đó tính các score giữa y_test và y_pred để chọn mô hình tốt hơn.

1. Logistic Regression

- Phương pháp hồi quy logistic là một mô hình hồi quy nhằm dự đoán giá trị đầu ra *rời rạc* (*discrete target variable*) y ứng với một véc-tơ đầu vào **x.**
- Để tránh overfitting nhưngi vẫn giữ tính tổng quát => kĩ thuật chính quy hóa (regularization). Giảm độ lớn của các theta đi (C), làm giảm sức ảnh hưởng của một số feature khiến hàm đơn giản hơn, tránh được overfit (vì hàm dự đoán càng phức tạp thì càng dễ rơi vào overfit)

Tham số (C)	1000	1	0,0001
F1_score	0.884517	0.884324	0.882789
AUC score	0.744065	0.743809	0.684231

=> C càng lớn => chỉ số càng cao =>khả năng overfit càng lớn => có thể tính tổng quát sẽ không đạt hiệu quả cao



2. Random Forest

- Mô hình thuộc lớp kết hợp (ensemble model) tức kết quả được đưa ra dựa trên không chỉ một mô hình mà từ nhiều mô hình khác nhau
- Xây dựng một rừng cây ngẫu nhiên dựa trên các node và nhánh. Đại diện cho mỗi node là một câu hỏi mà giá trị trả về là YES hoặc NO. Các nhánh sẽ có tác dụng kết nối các nodes để tạo ra một kịch bản đường đi (routine)
- Node bắt đầu của Random Forest là root node. Từ root node, mô hình sẽ xây dựng một bộ câu hỏi Yes/No dựa trên thông tin được cung cấp từ biến dự báo. Các nhánh YES, NO sẽ rẽ đến các node mới được gọi là internal node.
- Tại phía cuối của các nhánh YES/NO mô hình tiếp tục khởi tạo những internal node ở tầng thấp hơn với các biến khác. Thứ tự các biến được lựa chọn là ngẫu nhiên. Quá trình rẽ nhánh được thực hiện liên tục cho đến khi mô hình đi đến node cuối
- Kết quả từ mô hình Random Forest được kết hợp từ nhiều cây quyết định con và được thử nghiệm trên nhiều bộ dữ liệu con nên sai số dự báo thông thường nhỏ hơn so với những mô hình phân loại tuyến tính như logistic hoặc linear regression

3. Conclusion

	Logistic Regression	Random Forest
F1_score	0.8827899156693721	0.8830325400003962
ROC_AUC	0.685865725539639	0.7102730133651327

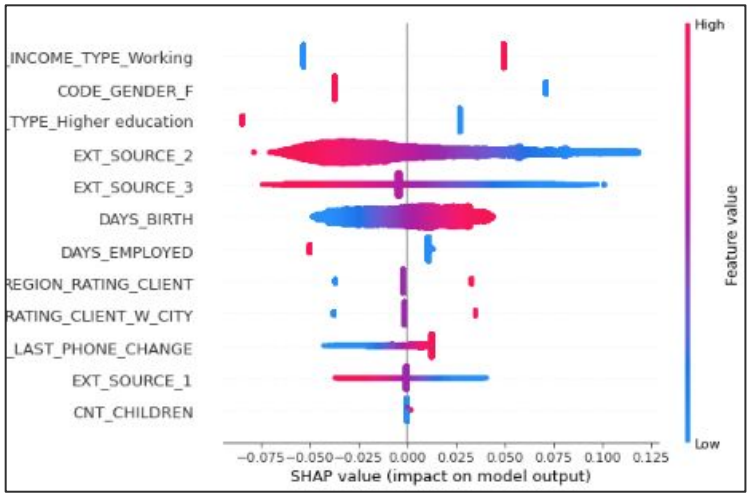
=> Random Forest có F1_score và ROC_AUC tốt hơn =>chọn Random Forest

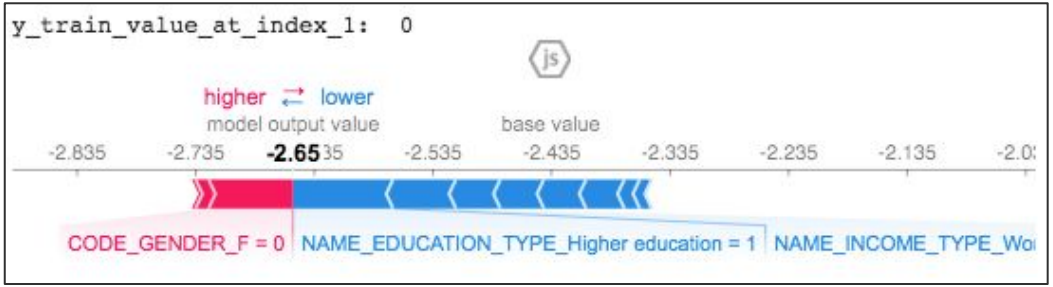
IV. Models (top 12 features)

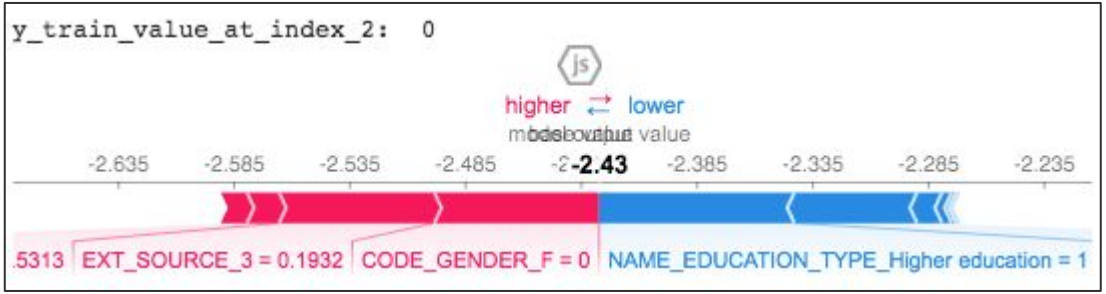
Áp P_value cho 12 biến đa chọn => biến x12 (CNT_CHILDREN) không có ý nghĩa thống kế
=> chỉ áp mô hình cho 11 biến

```
Optimization terminated successfully.
         Current function value: 0.255428
         Iterations 7
                           Logit Regression Results
==============================================================================
Dep. Variable:                 TARGET   No. Observations:                230633
Model:                          Logit   Df Residuals:                    230621
Method:                           MLE   Df Model:                           11
Date:                Sat, 15 Aug 2020   Pseudo R-squ.:                 0.09345
Time:                        03:26:24   Log-Likelihood:                -58910.
converged:                       True   LL-Null:                       -64983.
Covariance Type:            nonrobust   LLR p-value:                     0.000
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
x1             0.1989      0.038      5.251      0.000       0.125       0.273
x2            -2.3788      0.037    -64.344      0.000      -2.451      -2.306
x3            -1.8411      0.031    -58.804      0.000      -1.902      -1.780
x4            -1.1889      0.050    -23.608      0.000      -1.288      -1.090
x5             0.5437      0.106      5.133      0.000       0.336       0.751
x6            -0.2066      0.105     -1.963      0.050      -0.413      -0.000
x7            -0.4257      0.021    -19.896      0.000      -0.468      -0.384
x8             0.1519      0.018      8.361      0.000       0.116       0.188
x9            -0.2611      0.016    -15.950      0.000      -0.293      -0.229
x10           -0.1324      0.032     -4.159      0.000      -0.195      -0.070
x11            0.4224      0.038     11.067      0.000       0.348       0.497
x12           -0.0344      0.209     -0.165      0.869      -0.444       0.375
==============================================================================
```

Prediction explained by SHAP library







IV. Models (top 12 features)

Sử dụng models cho 11 biến có correlation lớn nhất

-

	Logistic Regression	Random Forest
F1_score	0.8827899156693721	0.8849387635563939
ROC_AUC	0.6926136209679306	0.7047801229480826

=> Random Forest có F1_score và ROC_AUC tốt hơn =>chọn Random Forest

Lessions Learnt

- Tầm quan trọng của file chuẩn hóa =>đỡ tốn thời gian debug
- Planning của 1 project
- Tầm quan trọng của phần tiền xử lý dữ liệu (outlier; missing values;...)
- Hiểu rõ data phân tích => lựa chọn MH, phương pháp đánh giá phù hợp

THANK YOU!