

Credit Card Approval Analysis and Predictive Modeling

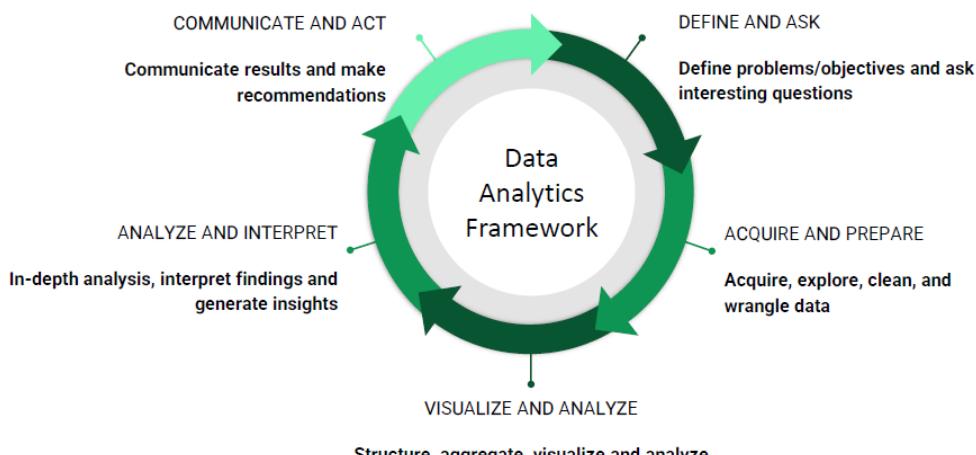
การวิเคราะห์และการสร้างแบบจำลองคาดการณ์การอนุมัติบัตรเครดิต

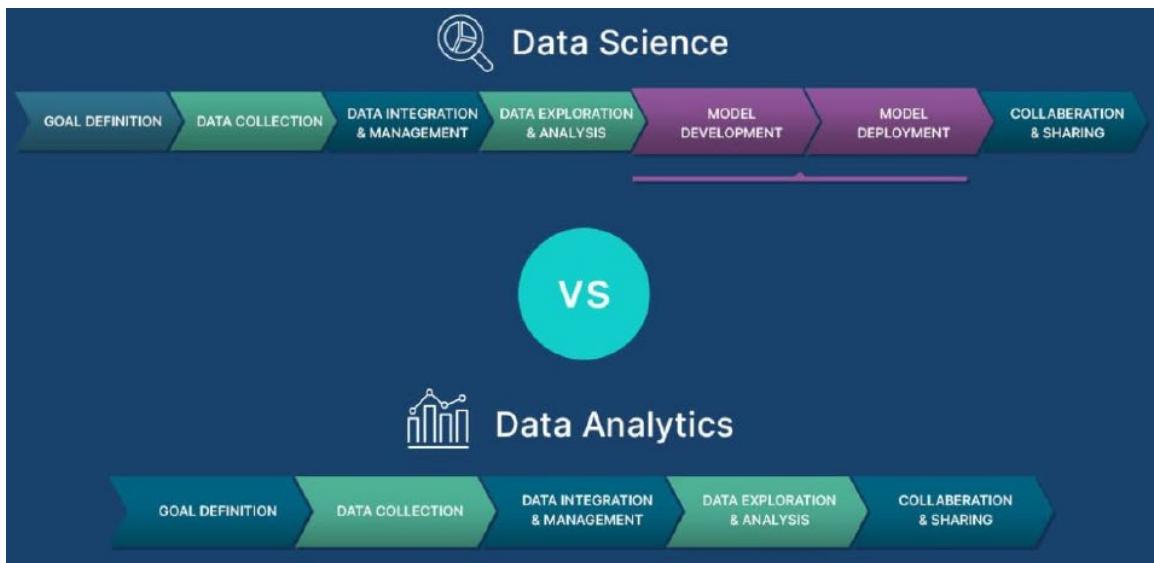
1. Introduction

สถาบันการเงินอนุมัติ/ปฏิเสธการสมัครบัตรเครดิต โดยพิจารณาจากข้อมูลผู้สมัคร โดยกระบวนการพิจารณาอาจใช้เวลานานและขึ้นอยู่กับดุลยพินิจของนักวิเคราะห์สินเชื่อ ซึ่งอาจนำไปสู่ความเสี่ยงหรือการสูญเสียโอกาสทางธุรกิจ การสร้างแบบจำลองเพื่อทำนายผลการอนุมัติบัตรเครดิตโดยอัตโนมัติ ทำให้เพิ่มความรวดเร็วและความแม่นยำในการตัดสินใจ และสามารถลดความเสี่ยงทางการเงิน (อัตราการผิดนัดชำระหนี้ - Default Rate) รวมถึงเพิ่มประสิทธิภาพในการดำเนินงาน (Operational Efficiency) ของฝ่ายอนุมัติสินเชื่อของธนาคารและสถาบันการเงิน (ฝ่ายบริหารความเสี่ยง, ฝ่ายการตลาด, ฝ่ายปฏิบัติการ)

2. Problem Statement/Background

การทำนายผลการอนุมัติบัตรเครดิต (Approved หรือ Rejected) จากข้อมูลของผู้สมัครที่ให้มาด้วยความแม่นยำสูง เพื่อช่วยในการตัดสินใจอัตโนมัติ ความสามารถลดความเสี่ยงทางการเงิน (อัตราการผิดนัดชำระหนี้ - Default Rate) โดยแนวทางการวิเคราะห์จะดำเนินตาม Data Analysis & Data Science Framework ต่อไปนี้





2. Data Analytics Methodology

2.1 Analytical Questions

เพื่อค้นหาคุณลักษณะของผู้สมัคร (เช่น อายุ, รายได้, ประเภทงาน) ที่มีความสัมพันธ์กับการอนุมัติมากที่สุด และหากลุ่มลูกค้า (Segments) ที่มีความเสี่ยงสูง/ต่ำ ที่แตกต่างกันขนาดไหน

2.2. Data Sources/Attributes

Kaggle Dataset: "Credit Card Approvals (Clean Data)" (ชีว์จำลองมาจากข้อมูลการสมัครบัตรเครดิต)

ข้อมูลในทางปฏิบัติ: ฐานข้อมูลภายในของธนาคาร/สถาบันการเงิน (ข้อมูลการสมัคร, ประวัติเดินเรื่อง, ข้อมูลเครดิตบุคคล)

Gender	Age	Debt	Married	BankCustomer	Industry	Ethnicity	YearsEmployed	PriorDefault	Employed	CreditScore	DriversLicense	Citizen	ZipCode	Income	Approved
1	30.83	0	1	1	Industrials	White	1.25	1	1	1	0	ByBirth	202	0	1
0	58.67	4.46	1	1	Materials	Black	3.04	1	1	6	0	ByBirth	43	560	1
0	24.5	0.5	1	1	Materials	Black	1.5	1	0	0	0	ByBirth	280	824	1
1	27.83	1.54	1	1	Industrials	White	3.75	1	1	5	1	ByBirth	100	3	1
1	20.17	5.625	1	1	Industrials	White	1.71	1	0	0	0	ByOtherMeans	120	0	1
1	32.08	4	1	1	CommunicationServices	White	2.5	1	0	0	1	ByBirth	360	0	1
1	33.17	1.04	1	1	Transport	Black	6.5	1	0	0	1	ByBirth	164	31285	1
0	22.92	11.585	1	1	InformationTechnology	White	0.04	1	0	0	0	ByBirth	80	1349	1
1	54.42	0.5	0	0	Financials	Black	3.96	1	0	0	0	ByBirth	180	314	1
1	42.5	4.915	0	0	Industrials	White	3.165	1	0	0	1	ByBirth	52	1442	1
1	22.08	0.83	1	1	Energy	Black	2.165	0	0	0	1	ByBirth	128	0	1
1	29.92	1.835	1	1	Energy	Black	4.335	1	0	0	0	ByBirth	260	200	1
0	38.25	6	1	1	Financials	White	1	1	0	0	1	ByBirth	0	0	1
1	48.08	6.04	1	1	Financials	White	0.04	0	0	0	0	ByBirth	0	2690	1
0	45.83	10.5	1	1	Materials	White	5	1	1	7	1	ByBirth	0	0	1
1	36.67	4.415	0	0	Financials	White	0.25	1	1	10	1	ByBirth	320	0	1
1	28.25	0.875	1	1	CommunicationServices	White	0.96	1	1	3	1	ByBirth	396	0	1
0	23.25	5.875	1	1	Materials	White	3.17	1	1	10	0	ByBirth	120	245	1
1	21.83	0.25	1	1	Real Estate	Black	0.665	1	0	0	1	ByBirth	0	0	1

ภาพที่ 1 ตัวอย่าง Data Set

Attribute	Description	Data Type	Valid Range/Example
Gender	Gender	Nominal	0=Female, 1=Male
Age	Age (Years)	Interval	[20, 80]
Debt	Outstanding debt (Feature has been scaled)	Ratio (Continuous)	[0, Infinity)
Married	Married Status	Nominal	Married, 0=Single/Divorced/etc, 1=Married
BankCustomer	Bank Customer Status	Nominal	0=Does not have a bank account, 1=Has a bank account)
Industry	Industry (Job sector of current or most recent job)	Nominal	[0, 14] [CommunicationServices, ConsumerDiscretionary, ConsumerStaples, Education, Energy, Financials, Healthcare, Industrials, InformationTechnology, Materials, Real Estate, Research, Transport, Utilities]
Industry_Name	Industry Name	Nominal	
Ethnicity	Ethnicity	Nominal	White, Black, Asian, Latino, Other
YearsEmployed	Years employed	Ratio (Discrete)	[0, Infinity)
PriorDefault	Prior default	Asymmetric Binary	0=No prior defaults, 1=Prior default
Employed	Employed Status	Asymmetric Binary	0=Not employed, 1=Employed
CreditScore	Credit score (Feature has been scaled)	Interval	[0, 100]
DriversLicense	Drivers license	Symmetric Binary	0=No license, 1=Has license
Citizen	Citizenship	Nominal	ByBirth, ByOtherMeans, Temporary
ZipCode	ZipCode	Nominal	5 digit number
StateName	StateName	Nominal	[100 = Michigan, etc.]
Income	Income (Feature has been scaled)	Ratio (Continuous)	[0, Infinity)
Approved	Approved Status	Asymmetric Binary	0=Not approved, 1=Approved

ภาพที่ 2 ตัวอย่าง Data Dictionary

2.3 Data cleaning and preprocessing

เนื่องจากเป็น "Clean Data" จะเน้นการตรวจสอบข้อบกพร่องที่สำคัญ การจัดการกับค่าที่หายไป (Missing Values) ที่มีอยู่

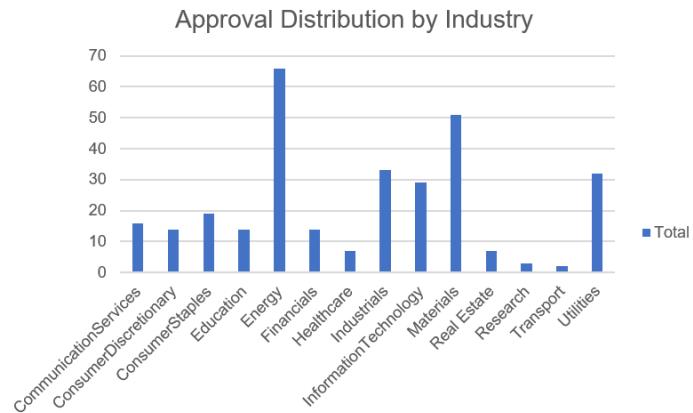
3. Analytics Methodology

3.1 EDA techniques: การวิเคราะห์ความถี่และค่าสถิติเชิงพรรณนา เพื่อดูความลับเบื้องต้นของแต่ละคุณลักษณะกับผลการอนุมัติ

	Gender	Age	Debt	Married	BankCustomer	Industry	YearsEmployed	PriorDefault	Employed	CreditScore	DriversLicense	ZipCode	Income	Approved
count	690.000000	690.000000	690.000000	690.000000	690.000000	690.000000	690.000000	690.000000	690.000000	690.000000	690.000000	690.000000	690.000000	690.000000
mean	0.695652	31.514116	4.758725	0.760870	0.763768	6.960870	2.223406	0.523188	0.427536	2.40000	0.457971	180.547826	1017.385507	0.444928
std	0.460464	11.860245	4.978163	0.426862	0.425074	3.802822	3.346513	0.499824	0.495080	4.86294	0.498592	173.970323	5210.102598	0.497318
min	0.000000	13.750000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	22.670000	1.000000	1.000000	1.000000	3.000000	0.165000	0.000000	0.000000	0.000000	0.000000	60.000000	0.000000	0.000000
50%	1.000000	28.460000	2.750000	1.000000	1.000000	7.000000	1.000000	1.000000	0.000000	0.000000	0.000000	160.000000	5.000000	0.000000
75%	1.000000	37.707500	7.207500	1.000000	1.000000	10.000000	2.625000	1.000000	1.000000	3.000000	1.000000	272.000000	395.500000	1.000000
max	1.000000	80.250000	28.000000	1.000000	1.000000	14.000000	28.500000	1.000000	1.000000	67.000000	1.000000	2000.000000	100000.000000	1.000000

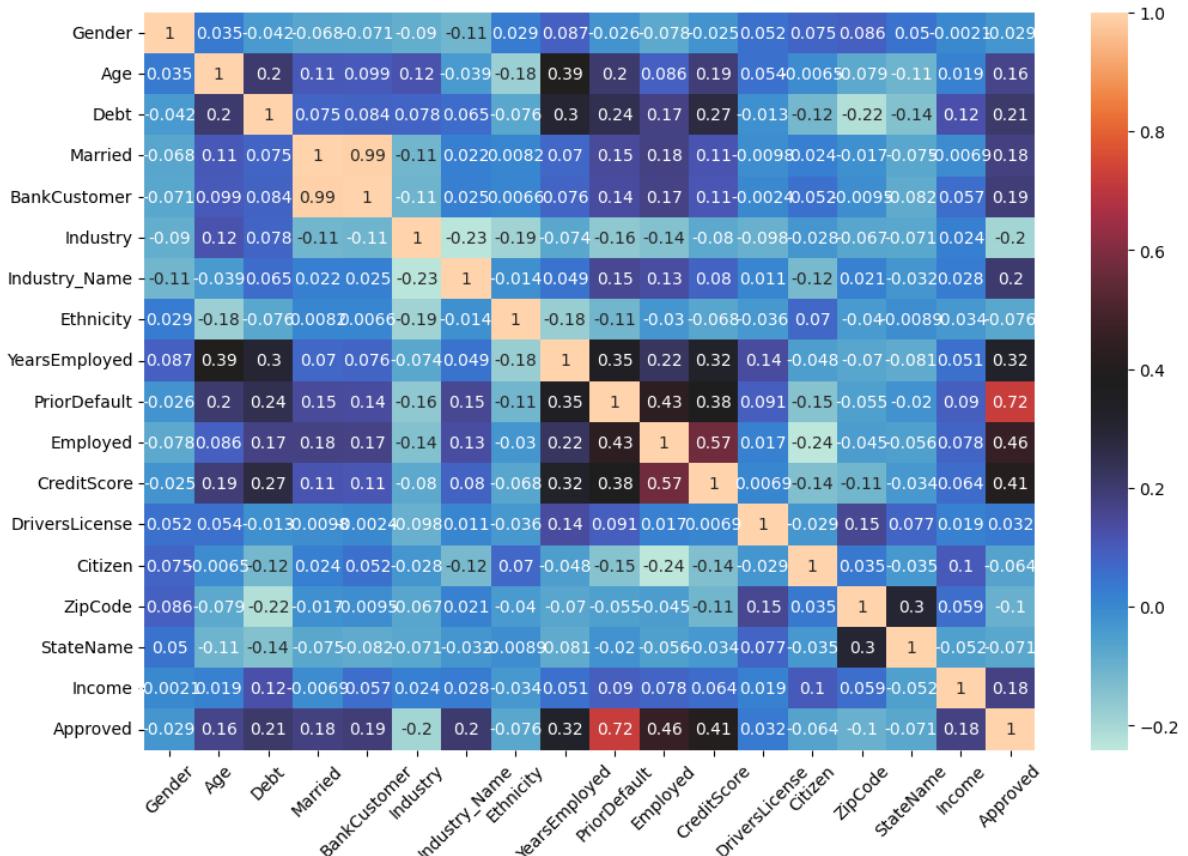
ภาพที่ 4 สถิติเชิงพรรณนา (Descriptive Statistics)

Industry Name	Number of Approved
CommunicationServices	16
ConsumerDiscretionary	14
ConsumerStaples	19
Education	14
Energy	66
Financials	14
Healthcare	7
Industrials	33
InformationTechnology	29
Materials	51
Real Estate	7
Research	3
Transport	2
Utilities	32
Grand Total	307



ภาพที่ 5 Pivot Table and Graph

3.2 Visualization strategy: Heatmap สำหรับ Correlation Matrix



ภาพที่ 6 Correlation Heatmap (แผนที่ความร้อนสหสัมพันธ์)

ค่าสัมประสิทธิ์สหสัมพันธ์แบบเพียร์สัน (Pearson Correlation) ที่มีค่ามากกว่า 0.3 (ทั้งค่าบวกและลบ)

หมายความว่ามีความสัมพันธ์เชิงเส้นระหว่างตัวแปรทั้งสองในระดับ ปานกลางถึงค่อนข้างมาก โดยมีค่าดังนี้

Married and BankCustomer = 0.992
PriorDefault and Approved = 0.720
Employed and CreditScore = 0.571
Employed and Approved = 0.458
PriorDefault and Employed = 0.432
CreditScore and Approved = 0.406
Age and YearsEmployed = 0.391
PriorDefault and CreditScore = 0.380
YearsEmployed and PriorDefault = 0.346
YearsEmployed and Approved = 0.322
YearsEmployed and CreditScore = 0.322
ZipCode and StateName = 0.302

ภาพที่ 7 ค่าสัมประสิทธิ์สหสัมพันธ์แบบเพียร์สัน (Pearson Correlation) ที่มีค่ามากกว่า 0.3

ความสัมพันธ์ที่โดดเด่น	ค่า Correlation	ความหมายเชิงธุรกิจ
PriorDefault and Approved	0.720	สำคัญที่สุด: ผู้สมัครที่มีประวัติ ไม่เคยผิดนัดชำระหนี้ (PriorDefault=0) มีโอกาสได้รับการอนุมัติสูงมาก (ความเสี่ยงต่ำ)
Employed and Approved	0.458	ผู้สมัครที่มี งานทำ (Employed=1) มีแนวโน้มได้รับการอนุมัติสูงกว่า
CreditScore and Approved	0.406	คะแนนเครดิตสูง เป็นปัจจัยหลักในการอนุมัติ
YearsEmployed and Approved	0.322	ระยะเวลาการทำงานที่ยาวนาน มีความสัมพันธ์กับการอนุมัติ ซึ่งบ่งชี้ถึง ความมั่นคงทางการเงิน

ภาพที่ 8 ค่าสัมประสิทธิ์สหสัมพันธ์แบบเพียร์สัน (Pearson Correlation) ที่มีสัมพันธ์กับการอนุมัติสินเชื่อ

บทสรุปเชิงวิเคราะห์: ปัจจัยด้าน ประวัติความเสี่ยง (PriorDefault) และ ความมั่นคง (Employed, CreditScore, YearsEmployed) เป็นคุณลักษณะหลักที่ขับเคลื่อนผลการตัดสินใจอนุมัติบัตรเครดิต ซึ่งตอบคำถามเชิงวิเคราะห์ที่ว่า "คุณลักษณะใดของผู้สมัครมีความสัมพันธ์กับการอนุมัติมากที่สุด"

3.3 Segmentation approach: การจัดกลุ่มลูกค้าตามโปรไฟล์ความเสี่ยง (Risk Profiles)

การเปรียบเทียบ Credit Score (CreditScore) ระหว่างผู้ที่ได้รับการอนุมัติ (Approved=1) และผู้ที่ไม่ได้รับการอนุมัติ (Approved=0) สามารถทำ t-test ได้ เนื่องจาก CreditScore เป็นตัวแปรเชิงปริมาณ(Numerical/ Continuous) และเราต้องการเปรียบเทียบค่าเฉลี่ยของตัวแปรนี้ระหว่างสองกลุ่มอิสระ(Accredited=1 และ Approved=0) การใช้ Independent Samples T-test (การทดสอบที่สำหรับกลุ่มตัวอย่างอิสระ) จึงเป็นวิธีที่เหมาะสมที่สุด

Independent Samples T-test

สมมติฐาน

H_0 (สมมติฐานว่าง): ค่าเฉลี่ย CreditScore ของผู้ที่ได้รับการอนุมัติ ไม่แตกต่าง จากผู้ที่ไม่ได้รับการอนุมัติ

H_a (สมมติฐานทางเลือก): ค่าเฉลี่ย CreditScore ของผู้ที่ได้รับการอนุมัติ แตกต่าง จากผู้ที่ไม่ได้รับการอนุมัติ

ค่าเฉลี่ย CreditScore กลุ่ม Approved (1): 4.6059

ค่าเฉลี่ย CreditScore กลุ่ม Not Approved (0): 0.6319

ค่า T-statistic: 10.6384

ค่า P-value: 0.0000

P-value < 0.05: ปฏิเสธ H_0 สรุปได้ว่า Credit Score เฉลี่ยแตกต่างกันอย่างมีนัยสำคัญ

ภาพที่ 9 Independent Samples T-test

ผลจากการวิเคราะห์ความสัมพันธ์ของฟีเจอร์ (Correlation Analysis) เป็นไปตามสมมติฐานที่ว่า มีกลุ่มลูกค้า (Segments) ที่มีความเสี่ยงสูง/ต่ำ ที่แตกต่างกันขึ้นหรือไม่ กล่าวโดยสรุปคือ ค่าเฉลี่ย Credit Score ของผู้ที่ได้รับการอนุมัติ แตกต่าง จากผู้ที่ไม่ได้รับการอนุมัติ

2. ข้อบกพร่องในเกณฑ์การตัดสินใจแบบเดิม (Potential Flaws in Traditional Criteria)

คำตอบ: หากเกณฑ์การตัดสินใจเพียงพa CreditScore เพียงอย่างเดียวหรือใช้ค่า Cutoff ที่ไม่ยืดหยุ่น อาจเกิดข้อบกพร่องดังต่อไปนี้

ข้อบกพร่องที่อาจเกิดขึ้น

นัยยะทางธุรกิจ

การพึ่งพาตัวแปรเดียวมากเกินไป (Over-reliance on CreditScore)

อาจละเลยศักยภาพของลูกค้าที่มี CreditScore ปานกลาง (Gray Zone) แต่มีปัจจัยชุดเดียวกัน ๆ ที่แข็งแกร่ง (เช่น Income สูงมาก หรือ YearsEmployed ยาวนาน) ซึ่งอาจกลยับเป็นลูกค้าที่ดีในอนาคต

การใช้ค่า Cutoff แบบแข็งตัว

การใช้ค่า Cutoff แบบตายตัวอาจทำให้เสียโอกาส (Lost Opportunities) ในการอนุมัติลูกค้าที่เพิ่งผ่านเกณฑ์มาเพียงเล็กน้อย หรือปฏิเสธลูกค้าที่อยู่ต่ำกว่าเกณฑ์นิดเดียวแต่มีความเสี่ยงต่ำเมื่อรวมปัจจัยอื่น

การไม่พิจารณา Interaction Effects

เกณฑ์แบบเดิมอาจไม่ได้ให้ความสำคัญกับการทำงานร่วมกันของปัจจัย (เช่น ลูกค้าที่มี CreditScore ต่ำ แต่มีประวัติการจ้างงานมั่นคงยาวนาน) ทำให้การตัดสินใจขาดความแม่นยำในกลุ่มลูกค้าที่ซับซ้อน

3.4 Modeling Methodology

3.4.1 Model Development

- Algorithm selection: Classification Algorithms: Logistic Regression

- Training, hyperparameter tuning, evaluation metric:

Training: แบ่งข้อมูลเป็น Training, Validation และ Test Sets (Cross-Validation) Tuning: Grid Search เพื่อหา Hyperparameters ที่เหมาะสมที่สุด

- Evaluation Metric: Accuracy, Precision, Recall, F1-Score

3.4.2 Workflow– Model Development

```
# train_test_split จากไลบรารี scikit-learn เพื่อแบ่งข้อมูลทั้งหมด (x คือฟีเจอร์, y คือลิบลารีเป้าหมาย) ออกเป็น 4 ส่วน สำหรับใช้ในการวนการตัดกันและประเมินผลในเดลเมชันแล้วนั่นเอง
from sklearn.model_selection import train_test_split
```

```
# นำไฟล์ข้อมูลมาพลิก (x_train) และรดข้อมูลการทดสอบ (x_test) ผ่านการลอกเกจ Scale
from sklearn.preprocessing import StandardScaler

sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.transform(x_test)
```

```
# Build Machine Learning Model - Model name: Logistic Regression
from sklearn.linear_model import LogisticRegression

log_reg = LogisticRegression(random_state = 0)
log_reg.fit(x_train, y_train)
```

```
+ LogisticRegression ••
LogisticRegression(random_state=0)
```

```
y_pred = log_reg.predict(x_test)
print("Train Score: {:.5f}".format(log_reg.score(x_train, y_train)))
print("Test Score: {:.5f}".format(log_reg.score(x_test, y_test)))
```

```
Train Score: 0.87041
Test Score: 0.86705
```

3.4.3 Workflow– Model Development (Parameter Tuning / Hyperparameter Tuning)

```
# ในเดล Logistic Regression กำหนดค่าพารามิเตอร์ C = 0.1 เป็นความผูกพันของค่าความแปรของกราฟรูปให้เป็นระเรียง (Regularization)
# กำหนดประเภทของ Regularization เป็น L1 Regularization L1 Regularization มากเรียกว่า Lasso
# กำหนดอัลกอริทึมที่ใช้ในการหาค่าที่ดีที่สุด (Optimization Algorithm)
```

```
log_reg1 = LogisticRegression(C=0.1, penalty='l1', solver='liblinear', random_state = 0)
log_reg1.fit(x_train, y_train)
```

```
+ LogisticRegression ••
LogisticRegression(C=0.1, penalty='l1', random_state=0, solver='liblinear')
```

```
y_pred1 = log_reg1.predict(x_test)
print("Train Score: {:.5f}".format(log_reg1.score(x_train, y_train)))
print("Test Score: {:.5f}".format(log_reg1.score(x_test, y_test)))
```

```
Train Score: 0.85880
Test Score: 0.84971
```

```

# กำหนดค่าร่วมกันเดอร์
paramgrid = { 'C': [0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000], 'penalty':['l1', 'l2'], 'solver':['liblinear']}

from sklearn.model_selection import GridSearchCV
grid_search = GridSearchCV(LogisticRegression(), param_grid=paramgrid, cv=5, scoring='accuracy')

grid_search.fit(x_train, y_train)

# อุปาระนิเดอร์ที่ดีที่สุด
print("Best Parameters:", grid_search.best_params_)
# คะแนนความแม่นยำ (Accuracy Score) เฉลี่ยสูงสุด ที่ได้จากการประเมินโมเดลในกระบวนการ Cross-Validation (CV)
print("Best Score:{:.5f}".format(grid_search.best_score_))

# คะแนนความแม่นยำ (Accuracy Score) ที่ได้จากการประเมินโมเดลสุดท้าย (โมเดลที่ดีที่สุดจากการท่า Grid Search) บนชุดข้อมูล x_train ทั้งหมด
print("Grid Search - Train Score: {:.5f}".format(grid_search.score(x_train, y_train)))
print("Grid Search - Test Score: {:.5f}".format(grid_search.score(x_test, y_test)))

Best Parameters: {'C': 10, 'penalty': 'l1', 'solver': 'liblinear'}
Best Score: 0.87018
Grid Search - Train Score: 0.87621
Grid Search - Test Score: 0.87861

```

3.4.4 Workflow– Model Development (Parameter Tuning / Hyperparameter Tuning with Pipeline and GridsearchCV)

```

import pandas as pd
from sklearn.model_selection import GridSearchCV
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression

# 1. กำหนดขั้นตอนใน Pipeline
# -----
# ใช้ StandardScaler และ LogisticRegression
steps = [
    ('scaler', StandardScaler()), # ขั้นตอนที่ 1: Data Preprocessing (Standardization)
    ('model', LogisticRegression(random_state=0)) # ขั้นตอนที่ 2: Model (Logistic Regression)
]

pipeline = Pipeline(steps)
# -----


# 2. กำหนด Hyperparameters สำหรับ Grid Search ในรูปแบบ Pipeline
# -----
# กำหนดค่าร่วมกันเดอร์ใน Pipeline
paramgrid = {
    'model_C': [0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000],
    'model_penalty':['l1', 'l2'],
    'model_solver':['liblinear']
}
# -----

```

```

# 3. สร้าง Grid Search และทำการฝึกฝน (Fit)
grid_search = GridSearchCV(
    pipeline,           # ใช้ Pipeline
    param_grid=paragrid,
    cv=5,
    scoring='accuracy',
    n_jobs=-1
)

grid_search.fit(x_train, y_train)

# 4. แสดงผลลัพธ์
print("Best Parameters:", grid_search.best_params_)
print(f"Best Score: {grid_search.best_score_:.5f}")

# ตรวจสอบประสิทธิภาพบนชุดข้อมูลริทิ้ง
print(f"Grid Search - Train Score: {grid_search.score(x_train, y_train):.5f}")
print(f"Grid Search - Test Score: {grid_search.score(x_test, y_test):.5f}")

Best Parameters: {'model_C': 10, 'model_penalty': 'l1', 'model_solver': 'liblinear'}
Best Score: 0.87018
Grid Search - Train Score: 0.87427
Grid Search - Test Score: 0.87861

```

3.4 Model Evaluation

```

from sklearn.metrics import confusion_matrix, classification_report

# 1. โมเดลที่ดีที่สุดจาก Grid Search
best_model = grid_search.best_estimator_

# ใช้ get_params() เพื่อดู Hyperparameters ทั้งหมดของโมเดล
print("Hyperparameters ทั้งหมดของโมเดลที่ดีที่สุด:")
print(best_model.get_params())

# 2. ท่านายผลบนชุดข้อมูลทดสอบ
y_pred = best_model.predict(x_test)

# 3. สร้าง Confusion Matrix
cm = confusion_matrix(y_test, y_pred)
print("\nConfusion Matrix:\n", cm)

# 4. แสดง Classification Report
report = classification_report(y_test, y_pred)
print("\nClassification Report:\n", report)

```

Confusion Matrix:

```
[[83 15]
 [ 6 69]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.93	0.85	0.89	98
1	0.82	0.92	0.87	75
accuracy			0.88	173
macro avg	0.88	0.88	0.88	173
weighted avg	0.88	0.88	0.88	173

ประสิทธิภาพและการทำงานของแบบจำลอง (Model Performance and Discussion)

ผลลัพธ์จากโมเดลการพยากรณ์นี้แสดงให้เห็นว่าโมเดลมีความสามารถในการจำแนกผู้ขอสินเชื่อบัตรเครดิตได้อย่างแม่นยำและสมดุล โดยมี Accuracy รวมอยู่ที่ 88% จาก 173 ตัวอย่างในชุดทดสอบ ซึ่งสูงกว่าเกณฑ์ที่ยอมรับได้สำหรับงานจำแนกประเภท

ผลลัพธ์	ค่า	ความหมายเชิงธุรกิจ (Business Problem: ลดความเสี่ยง)
True Negative (TN)	83	โมเดลกำหนดถูกว่า 'ปฏิเสธ' และค่าจริงคือ 'ปฏิเสธ' (ดีมาก: ป้องกันการอนุมัติที่ไม่ควรเกิด)
True Positive (TP)	69	โมเดลกำหนดถูกว่า 'อนุมัติ' และค่าจริงคือ 'อนุมัติ' (ดีมาก: เพิ่มโอกาสทางธุรกิจ)
False Positive (FP)	15	ความเสี่ยงของการเสี่ยง: โมเดลกำหนดผิดว่า 'อนุมัติ' ทั้งที่ค่าจริงคือ 'ปฏิเสธ' (นำไปสู่ พนักงานโดยตรง)
False Negative (FN)	6	การสูญเสียโอกาส: โมเดลกำหนดผิดว่า 'ปฏิเสธ' ทั้งที่ค่าจริงคือ 'อนุมัติ' (ทำให้พลาดลูกค้าดีๆ)

Metric	Class 0 (ปฏิเสธ)	Class 1 (อนุมัติ)	อภิปรายเชิงธุรกิจ
Precision	0.93	0.82	เมื่อโมเดลกำหนด 'ปฏิเสธ' ความแม่นยำสูงมาก (93%) แต่เมื่อกำหนด 'อนุมัติ' ความแม่นยำลดลงเหลือ 82% (สะกดคำ FP = 15)
Recall	0.85	0.92	โมเดลสามารถระบุลูกค้าที่ควร 'อนุมัติ' ได้ถึง 92% (FN ตัว: 6) แต่การระบุลูกค้าที่ควร 'ปฏิเสธ' อยู่ที่ 85% (FP/FN: 15/6)
F1-Score	0.89	0.87	ค่าสมดุลที่ดี แต่ประสิทธิภาพในการกำหนดคลาส 'ปฏิเสธ' (Class 0) ยังคงสูงกว่าเล็กน้อย

โมเดลนี้มีประสิทธิภาพดีและสามารถนำไป Deploy เพื่อใช้ในการตัดสินใจอนุมัติเบื้องต้นโดยอัตโนมัติได้ตาม Value Proposition ของโครงการ อย่างไรก็ตาม ควรเน้นการปรับปรุง Precision ของ Class 1 ให้สูงขึ้นเพื่อลดความเสี่ยงที่จะสูญเสีย (FP) ให้ลดลงถ้วนหน้า

4. Findings and Business Insights

คุณลักษณะหลักที่ขับเคลื่อนผลการอนุมัติบัตรเครดิต (Feature Importance) คือ ปัจจัยด้าน ประวัติความเสี่ยง (PriorDefault) และ ความมั่นคง (Employed, CreditScore, YearsEmployed) โดยสามารถ ระบุกลุ่มลูกค้าที่มีความเสี่ยงสูง/ต่ำ ที่ชัดเจนได้โดย Credit Score ซึ่งเป็นปัจจัยขับเคลื่อนหลัก (Primary Driver) ในกระบวนการขออนุมัติ ทำให้เกิดการแบ่งกลุ่มลูกค้าเป็นสองกลุ่ม ได้อย่างมีประสิทธิภาพ

ดังนั้นแนวโน้มและรูปแบบคุณลักษณะของผู้สมัครที่ถูกอนุมัติ/ปฏิเสธ คือ ประวัติความเสี่ยง (PriorDefault) และ ความมั่นคง (Employed, CreditScore, YearsEmployed) เป็นคุณลักษณะหลักที่ขับเคลื่อนผลการตัดสินใจ อนุมัติบัตรเครดิต ซึ่งตอบคำถามเชิงวิเคราะห์ว่า "คุณลักษณะใดของผู้สมัครมีความสัมพันธ์กับการขออนุมัติมากที่สุด"

จากการเปรียบเทียบ Credit Score (CreditScore) ระหว่างผู้ที่ได้รับการอนุมัติ (Approved=1) และผู้ที่ไม่ได้รับการอนุมัติ (Approved=0) ค่าเฉลี่ย Credit Score ของผู้ที่ได้รับการอนุมัติ แตกต่าง จากผู้ที่ไม่ได้รับการอนุมัติ ซึ่งตอบคำถามเชิงวิเคราะห์ว่า "มีกลุ่มลูกค้า (Segments) ที่มีความเสี่ยงสูง/ต่ำ ที่แตกต่างกันชัดเจนหรือไม่"

อย่างไรก็ตาม หากเกณฑ์การตัดสินใจเพียงพa CreditScore เพียงอย่างเดียวหรือใช้ค่า Cutoff ที่ไม่ยืดหยุ่น อาจเกิดข้อบกพร่องดังต่อไปนี้ 1. การพึ่งพาตัวแปรเดียวมากเกินไป (Over-reliance on CreditScore) 2. การใช้ค่า Cutoff แบบแข็งตัว 3. การไม่พิจารณา Interaction Effects

การพัฒนาโมเดลทำนายผลการอนุมัติบัตรเครดิต (Approved หรือ Rejected) จากข้อมูลของผู้สมัครที่ให้มา ด้วยความแม่นยำสูง เพื่อช่วยในการตัดสินใจตัดโน้มติ โดยใช้ปัจจัยอื่นๆ ร่วมด้วย ทำให้โมเดลการพยากรณ์มี ความสามารถในการจำแนกผู้ขอสินเชื่อบัตรเครดิตได้อย่างแม่นยำและสมดุล โดยมี Accuracy รวมอยู่ที่ 88% จาก 173 ตัวอย่างในชุดทดสอบ ซึ่งสูงกว่าเกณฑ์ที่ยอมรับได้สำหรับงานจำแนกประเภท

5. Conclusion and Recommendation

จากการวิเคราะห์ที่กล่าวมา จะทำให้เราทราบถึงปัจจัยสำคัญที่นำมารวิเคราะห์การขออนุมัติสินเชื่อที่เหมาะสม รวมถึงการสร้างแบบจำลองเพื่อทำนายผลการอนุมัติบัตรเครดิตโดยอัตโนมัติ เพื่อเพิ่มความรวดเร็วและความแม่นยำใน การตัดสินใจ และการลดความเสี่ยงทางการเงิน (อัตราการผิดนัดชำระหนี้ - Default Rate) รวมถึงเพิ่มประสิทธิภาพใน การดำเนินงาน (Operational Efficiency) ของฝ่ายอนุมัติสินเชื่อของธนาคารและสถาบันการเงิน (ฝ่ายบริหารความเสี่ยง , ฝ่ายการตลาด, ฝ่ายปฏิบัติการ) โดยที่ไม่เดลนี้สามารถให้ผลการตัดสินใจเบื้องต้นที่ รวดเร็ว (Operational Impact) และ เป็นกลาง ด้วย Accuracy 88% และทำให้ลดการเกิดหนี้เสีย (ลดความเสี่ยง) เนื่องจากมีความแม่นยำในการคัดกรอง มากขึ้น และเพิ่มโอกาสในการอนุมัติลูกค้าที่มีคุณภาพเร็วขึ้น (เพิ่มกำไร) และผู้สมัครได้รับผลการตัดสินใจรวดเร็วขึ้น ทำให้ประสบการณ์การใช้บริการดีขึ้น