Final Mini Project Report

STA393 SPECIAL TOPICS I: STATISTICAL LEARNING FOR DATA SCIENTISTS

Members:

Mr.Keerathap Ploysri ID student: 61070502404 (ENE)

Mr.Sorayut Meeyim ID student: 61070502468 (ENE)

Topic:

Loan default prediction with Berka Dataset

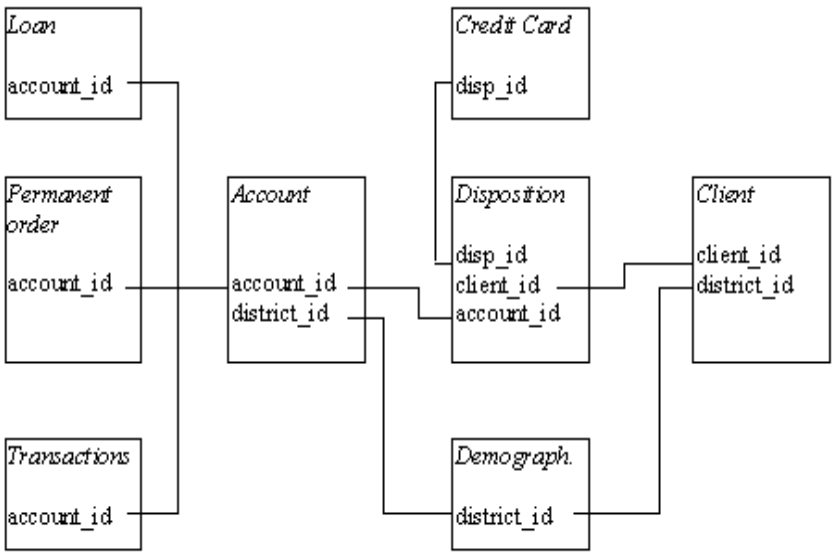
Goal:

To provides mechanisms in determining which consumers should receive loans and to benefit banks in increasing profits.

Procedures

1. Data preparation

Berka Dataset ถูกเก็บไว้ในรูปแบบ Database โดยมี ER diagram เป็นไปตามรูป



รูปที่ 1 ER Diagram แสดงความสัมพันธ์ของข้อมูล

จากรูปที่ 1 พบว่าในแต่ละตารางจะแสดงถึงข้อมูลที่แตกต่างกันจึงต้องทำการรวบรวมสรุปผลข้อมูลร่วมกับตารางหลัก (loan table) โดยมีขั้นตอนการทำดังนี้

- Loan table, Account table

  ทำการ loan table LEFT JOIN account table และสร้าง feature ที่ชื่อว่า day_before_loan ซึ่งเป็นจำนวนวันตั้งแต่เปิดบัญชีถึงวันที่ขอกู้ยืมเงิน

- Order table

  หลังจากการ Explore order table พบว่าข้อมูลที่ได้จาก order table สามารถสร้างมาได้จาก transaction table จึงไม่สนใจ table นี้

- Transaction table
  - เนื่องจากโจทย์ คือ สร้างเครื่องมือสำหรับการออกเงินกู้ ดังนั้น Dataset ที่ใช้ได้ถูกทำให้เป็นเฉพาะก่อนการกู้ยืมเท่านั้น
  - ทำการแบ่งชนิดจำนวนเงินเป็น เงินเข้า และเงินออก
  - สร้าง features ที่ได้จาก Monthly payments (e.g., Household, Insurrance) เป็นผลรวมของจำนวนเงินทั้งหมดในแต่ละชนิดการใช้จ่าย เช่น Sum monthly payment for household จำนวนครั้งที่ทำการจ่ายเงิน
    และจำนวนเงินที่ต้องใช้จ่ายแยกตามแต่ละชนิดในเดือนล่าสุด เช่น Current monthly payment for household
  - สร้าง features ที่จาก transactions อื่น ๆ ที่ไม่ใช่ monthly payments เป็นผลรวมจำนวนเงินและจำนวนครั้งที่ทำธุรกรรม
  - สร้าง features เกี่ยวกับ amount (จำนวนเงิน), balance (ยอดเงินคงเหลือ) เช่น latest, min, max, mean ของทั้งหมด รวมถึงพิจารณาเฉพาะ 1 เดือนล่าสุด และ 3 เดือนล่าสุด
  - นำ features ที่ได้จาก amount, balance มาหารด้วย loan payment (จำนวนเงินที่ต้องจ่ายต่อเดือน สำหรับผ่อนชำระเงินกู้) เพื่อสร้าง features เพิ่ม
  - สร้าง feature ที่ขื่อว่า growth_balance
    ซึ่งแสดงถึงอัตราการเพิ่มขึ้นหรือลดลงของยอดเงินคงเหลือเทียบกับเมื่อ 3 เดือนที่แล้ว

- Disp table, Client table, Card table

  ทำการ Merge ทั้ง 3 tables โดยใช้ client_id และ disp_id จากนั้นได้สร้าง features: age, all_client_mean_age, all_client_count และ card type ต่าง ๆ

- District table

  เป็นข้อมูลเกี่ยวกับที่อยู่หรือสภาพแวดล้อมของลูกค้า ได้สร้าง features num_inhabitants, urban_rate, avg_salary, num_enterpreneurs_per1000inhabitants

ซึ่งจากการทำ Data manipulation และ Feature extraction ทำให้ได้ features ทั้งหมดจำนวน 72 features และมี 682 records

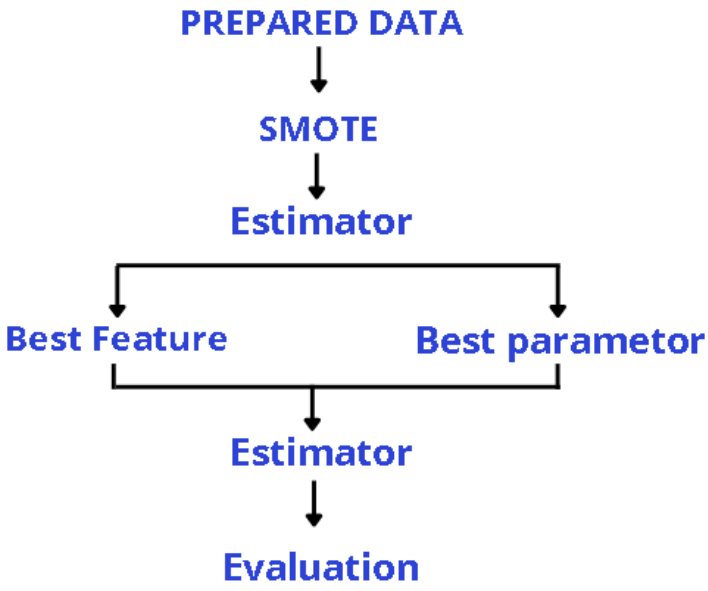
2. Model

ในการเริ่มต้น การสร้างโมเดลเราเริ่มต้นด้วยการใช้ lazypredict & orange canvas เป็นการทำ proof of concept ซึ่งจากผลลัพธ์ดังรูปที่ 2 เราจึงเลือกใช้เป็น XGBClassifier

Model	Accuracy	Balanced Accuracy	ROC AUC	F1 Score	Time Taken
BaggingClassifier	0.94	0.74	0.74	0.93	0.09
XGBClassifier	0.92	0.73	0.73	0.92	0.20
AdaBoostClassifier	0.90	0.72	0.72	0.90	0.23
RandomForestClassifier	0.93	0.69	0.69	0.92	0.39
ExtraTreesClassifier	0.94	0.68	0.68	0.93	0.23
LinearDiscriminantAnalysis	0.93	0.68	0.68	0.91	0.04
BernoulliNB	0.92	0.67	0.67	0.91	0.03
RidgeClassifier	0.92	0.62	0.62	0.90	0.02
RidgeClassifierCV	0.92	0.62	0.62	0.90	0.03

รูปที่ 2 แสดงผลลัพธ์ที่ได้จาก Lazypredict

***Note: lazypredict เป็น AutoMl ซึ่งทำการเลเบลคลาสตรงข้ามกับผู้จัดทำ ฉนั้น F1 Score ที่ได้จึงได้ค่าสูงกว่าปกติ*



รูปที่ 3 ภาพรวมขั้นตอนการทำงาน

เมื่อเราเลือกโมเดลได้แล้วและเตรียมข้อมูลเรียบร้อย จากนั้นเป็นขั้นตอนการหา Best Features เนื่องจากโมเดลประเภท Tree สามารถที่จะคำนวณเป็น importances ได้ ฉนั้นเราจะสามารถเรียงลำดับฟีเจอร์ที่สำคัญได้ จากนั้นจึงทำการ cross validation กับ model โดยใช้ฟีเจอร์จำนวนจากน้อยไปมากตามลำดับฟีเจอร์ที่เรียงด้วย importances ซึ่งเงื่อนไขในการเลือกก็คือใช้ฟีเจอร์เท่าไหรแล้วได้ F1 score สูงสุด จากเงื่อนไขนี้ได้ฟีเจอร์ที่สำคัญทั้งหมด 21 ฟีเจอร์ คือ

No.	Name	Describtion
1	acc_frequency_after_transaction	account issuance after transaction
2	acc_frequency_weekly	account weekly issuance
3	count_out_sanction_interest_neg_bal	จำนวนครั้งที่ถูกธนาคารปรับเนื่องจากเงินในบัญชีติดลบ
4	sum_monthly_payment_amount	ผลรวมยอดเงินทั้งหมดที่ใช้จ่ายไปกับค่าใช้จ่ายรายเดือน
5	balance_min	ยอดเงินคงเหลือต่ำสุด
6	amount_last_month_per_loan_payments	ผลรวมของจำนวนเงินที่ใช้จ่ายเดือนล่าสุดหารโดยจำนวนเงินที่ต้องชำระเงินกู้ต่อเดือน
7	balance_latest	ยอดเงินคงเหลือล่าสุด
8	current_monthly_payment_out_payment_for_statement	จำนวนเงินที่จ่ายให้กับค่า statement เดือนล่าสุด
9	balance_min_last_3months	ยอดเงินคงเหลือต่ำสุดในรอบ 3 เดทือน
10	amount_last_3months_avg	ผลรวมของจำนวนเงินทั้งหมดที่ทำธุรกรรมใน 3 เดือนล่าสุดเฉลี่ยแต่ละเดือน
11	balance_mean_per_loan_payment	ยอดเงินคงเหลือเฉลี่ยหารโดยจำน

		วนเงินที่ต้องชำระเงินกู้ต่อเดือน
12	all_client_count	จำนวนลูกค้าต่อ 1 account
13	amount_last_3months_out_avg	ผลรวมของจำนวนเงินรายจ่ายที่ทำธุรกรรมใน 3 เดือนล่าสุด เฉลี่ยแต่ละเดือน
14	client_gender_True	เพศของลูกค้า
15	day_before_loan	จำนวนวันตั้งแต่ที่เปิดบัญชีจนทำการกู้ยืม
16	cat__card_type_classic_True	ลูกค้าที่มีบัตรเครดิตประเภทคลาสสิก
17	balance_max_last_month	ยอดเงินคงเหลือสูงสุดของเดือนล่าสุด
18	avg_salary	เงินเดือนเฉลี่ย
19	loan_payments	ยอดจ่ายเงินกู้
20	current_monthly_payment_amount_out	จำนวนเงินทั้งหมดที่จ่ายสำหรับค่าใช้จ่ายรายเดือนของเดือนล่าสุด
21	balance_min_last_month	ยอดเงินคงเหลือต่ำสุดของเดือนล่าสุด

ส่วนการหา Best parametors ใช้ฟังก์ชั่น gridsearchcv ของ sklearn ซึ่งกำหนดพารามิเตอร์โดยเน้นการควบคุม over-fitting ได้แก่ max_depth, min_child_weight etc.

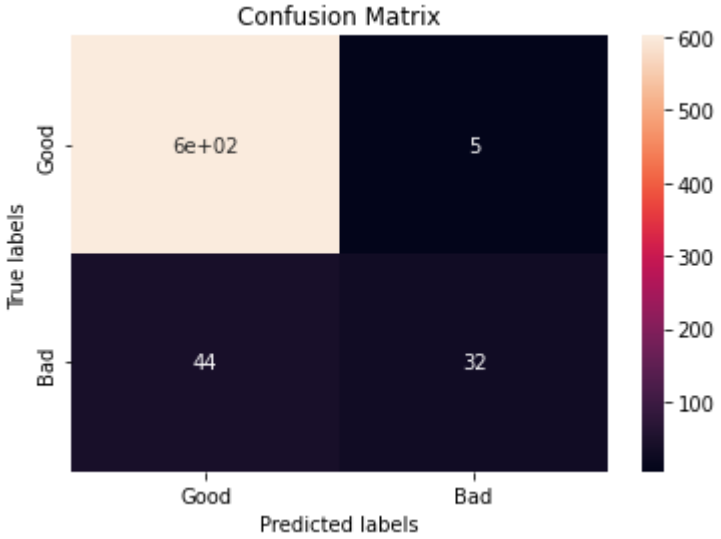
3. **Evaluation**

Model	inital model performance			Performance after using best params & best feature		
	Accuracy	F1 score	Roc auc	Accuracy	F1 score	Roc auc
LGBMClassifier	0.925	0.553	0.743	0.919	0.572	0.731
RandomForestClassifier	0.924	0.544	0.764	0.912	0.616	0.791
XGBClassifier	0.923	0.596	0.738	0.927	0.645	0.784

ซึ่งจากผลลัพธ์ตามตารางข้างต้น Accuracy ของทั้ง 3 โมเดลได้เกิน 0.9 ทั้งหมด ดังนั้นจึงต้องพิจารณาเมทริก F1 score และ ROC AUC ร่วมด้วย ซึ่งข้อมูลชุดนี้เป็นข้อมูลประเภท Unbalanced โดยเรามีการใช้เทคนิค SMOTE มาใช้ บวกกับการคัดเลือกฟีเจอร์และพารามิเตอร์สำหรับ binary classification นี้ ดังนั้นเรามองว่าโมเดลที่ให้ค่า F1 score สูงสุด จึงเหมาะกับงานนี้

Confusion matrix

เนื่องจากโจทย์นี้เป็น Binary classification ดังนั้นเราจึงทำ confusion matrix เกิดได้เป็น 4 กรณี คือ



รูปที่ 4 ผลลัพธ์ที่ได้ในรูป Confusion matrix

ซึ่งแสดงความหมายในเชิงธุรกิจได้ว่า

TN: สามารถได้บุคคลที่ตรงคุณลักษณะ ซึ่งการันตีสามารถจ่ายเงินกู้คืนได้

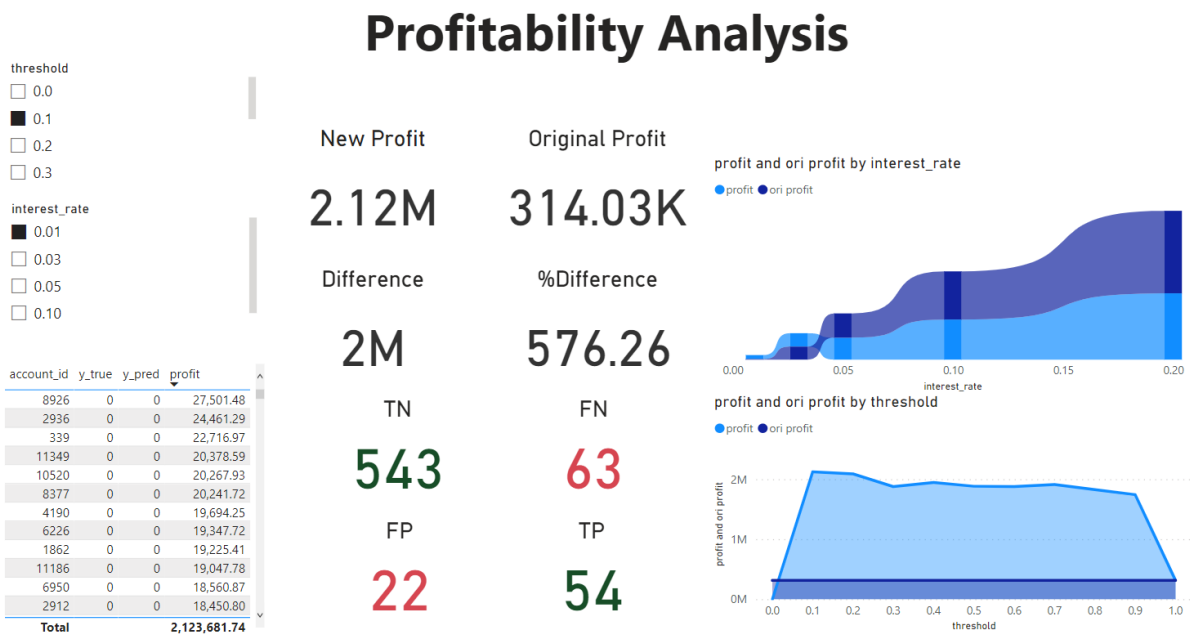
FN: ธนาคารเสียโอกาสทำรายได้จากการปล่อยกู้

FP: สร้างปัญหาหนักแก่ธนาคาร เพราะเค้าอาจจะชักดาบได้ (ฉนั้นสนใจกรณีนี้ที่สุด)

TP: สามารถรู้ได้ว่าไม่ควรปล่อยกู้คนนี้แน่ๆ เพราะน่าจะจ่ายคืนให้ไม่ได้

4. Profit calculation

หลังจากทำการทำนายจาก model เราสามารถนำผลลัพธ์ซึ่งเป็นค่า probs ที่ได้จากโมเดลมาวิเคราะห์ในเชิงของภาคธุรกิจเพิ่มเติม โดย Cost จะพิจารณาจากจำนวนเงินที่ไม่ถูกชำระ (กรณีที่ยังไม่ครบกำหนดสัญญาจะพิจารณาว่าไม่จ่าย 20% ของจำนวนเงินที่ยืมไป ซึ่งเป็นค่าเฉลี่ยที่ได้จากกรณีที่ครบสัญญา) และ Revenue จะได้จากดอกเบี้ยตามอัตราดอกเบี้ยที่กำหนดขึ้น



รูปที่ 5 Dashboard แสดงผลกำไร

จากรูปที่ 5 คือ Dashboard ที่แสดงผลกำไรเปรียบเทียบระหว่าง Original profit (Profit ที่ได้ก่อนใช้โมเดล) และ New profit (Profit ที่ได้หลังจากใช้โมเดลไปทำนายผล) โดยสามารถกำหนดอัตราดอกเบี้ยรายปี และค่า threshold probability พบว่าเราสามารถได้กำไรเพิ่มขึ้นสูงถึงถึง 576.26% จากเดิม ที่อัตราดอกเบี้ย 1% ต่อปี และ threshold เท่ากับ 0.01 แต่เมื่อเราเพิ่มค่าอัตราดอกเบี้ย threshold ที่ดีที่สุดต้องเพิ่มด้วย ซึ่งเมื่ออัตราดอกเบี้ยมากกว่า 10% ต่อปี พบว่าโมเดลเราจะสนใจกรณี False negative มาก จึงตัดสินใจปล่อยเงินกู้ให้ทุกคน

Conclusion

จากโปรเจค Loan default prediction with Berka dataset พบว่า Berka dataset ซึ่งเป็น Real-world dataset ที่ยังไม่ได้ทำการ Extract feature ออกมาเท่าที่ควร รวมทั้งข้อมูลที่เป็นแบบ Unbalanced data และจากโจทย์ที่ตั้งไว้ว่าเป็นการพิจารณาอนุมัติเงินกู้ ทำให้ใช้ข้อได้เฉพาะก่อนการกู้ยืมเงินเท่านั้น

ดังนั้น ทางผู้จัดทำได้ทำการเตรียมข้อมูล รวมถึง Extract features ออกมาเพิ่ม จากนั้นจัดการกับ Unbalanced data ด้วย SMOTE technique จากนั้นทำการทดลองโมเดลหลายชนิด และพบว่า XGboost classifier ได้ผลลัพธ์ดีที่สุด โดยทำการ Feature selection และ Hyperparameter tuning พบว่าได้ Accuracy = 92.7%, F1-score = 0.645, ROC_AUC = 0.784 ที่ threshold เท่ากับ 0.5

จากนั้นได้สร้าง Dashboard ที่สรุปผลกำไรที่ได้หลังจากนำโมเดลมาใช้ พบว่าสามารถทำให้ได้ผลกำไรมากขึ้นสูงสุดที่ 576.26% จากเดิม โดยใช้อัตราดอกเบี้ย 1% ต่อปี และ threshold เท่ากับ 0.01

Future work

- **Customers tracking system** เพื่อแจ้งเตือนธนาคาร ถ้าในเดือนหน้าลูกค้าอาจจะผิดนัดเงินกู้
- **Funding management system** ในกรณีที่ต้องการจัดสรรทรัพยากรของธนาคารที่มีอย่างจำกัด จึงใช้เพื่อจัดลำดับ/คัดกรองลูกค้าที่เหมาะสม
- **Maximize the interest rate** จัดสรรอัตราดอกเบี้ยให้กับแต่ละบุคคล

Miscellaneous

Power BI Dashboard

https://app.powerbi.com/view?r=eyJrIjoiZjAzNzBiODItMjFiMC00N2RhLWJlNzQtOTRhNTUzZDliNDkzIiwidCI6IjZmNDQzMmRjLTIwZDItNDQxZC1iMWRiLWFjMzM4MGJhNjMzZCIsImMiOjEwfQ%3D%3D

Github

https://github.com/sorayutmild/loan-default-prediction