Decision Tree Classifiers

Link to Colab Notebook:

https://colab.research.google.com/drive/14l5QdJ0b7TwWMgv3HVAjdMPzQENHcd7-?usp=sharing

Data Prepocessing

ผมพบว่ามี 3 columns ที่มีค่าแบบเดียวกันทั้ง column คือจากนั้นทำ LabelEncoder บน object column ทั้งหมด

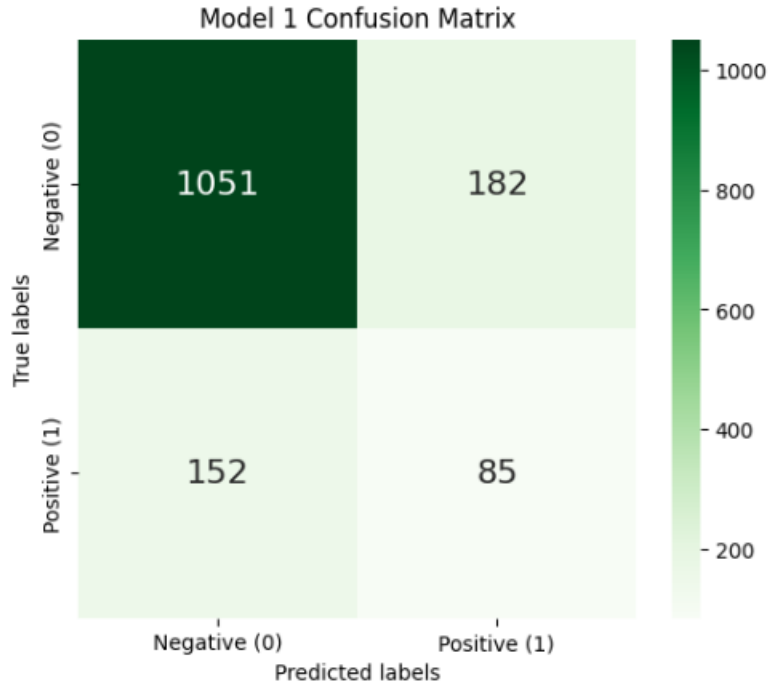
```
# Remove variables that do not change across the observation
# all is 1
df.drop('EmployeeCount', axis=1, inplace=True)
# all is 80
df.drop('StandardHours', axis=1, inplace=True)
# all is Y (everyone age more than 18)
df.drop('Over18', axis=1, inplace=True)
```

Model 1 Default CART

ได้ tree depth = 19

```
Model 1 Tree Depth: 19

Model 1 Confusion Matrix:
[[1051  182]
 [ 152   85]]
```

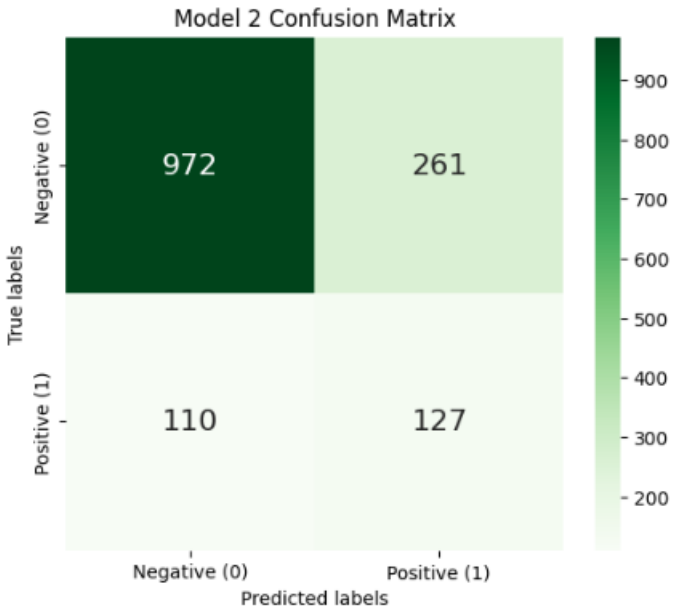


Model 2 Tuned CART

ผมใช้ GridSearchCV ทำให้ได้ max_depth คือ 5, min_samples_split คือ 20

```
Model 2 Best Hyperparameters: {'max_depth': 5, 'min_samples_split': 20}

Model 2 Confusion Matrix:
[[972 261]
 [110 127]]
```

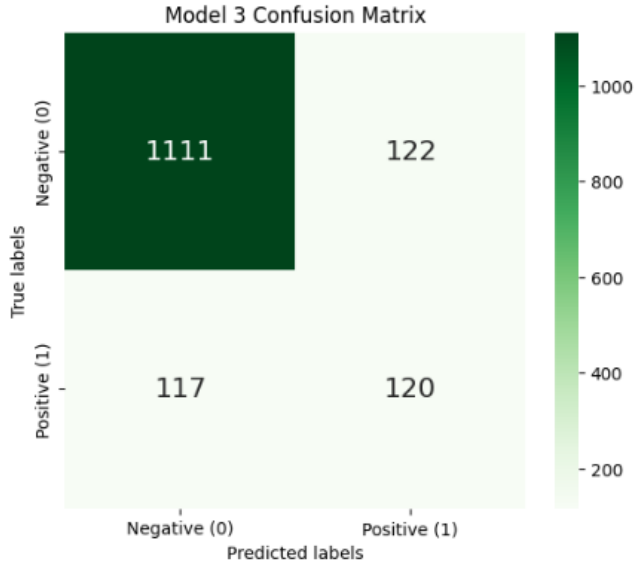


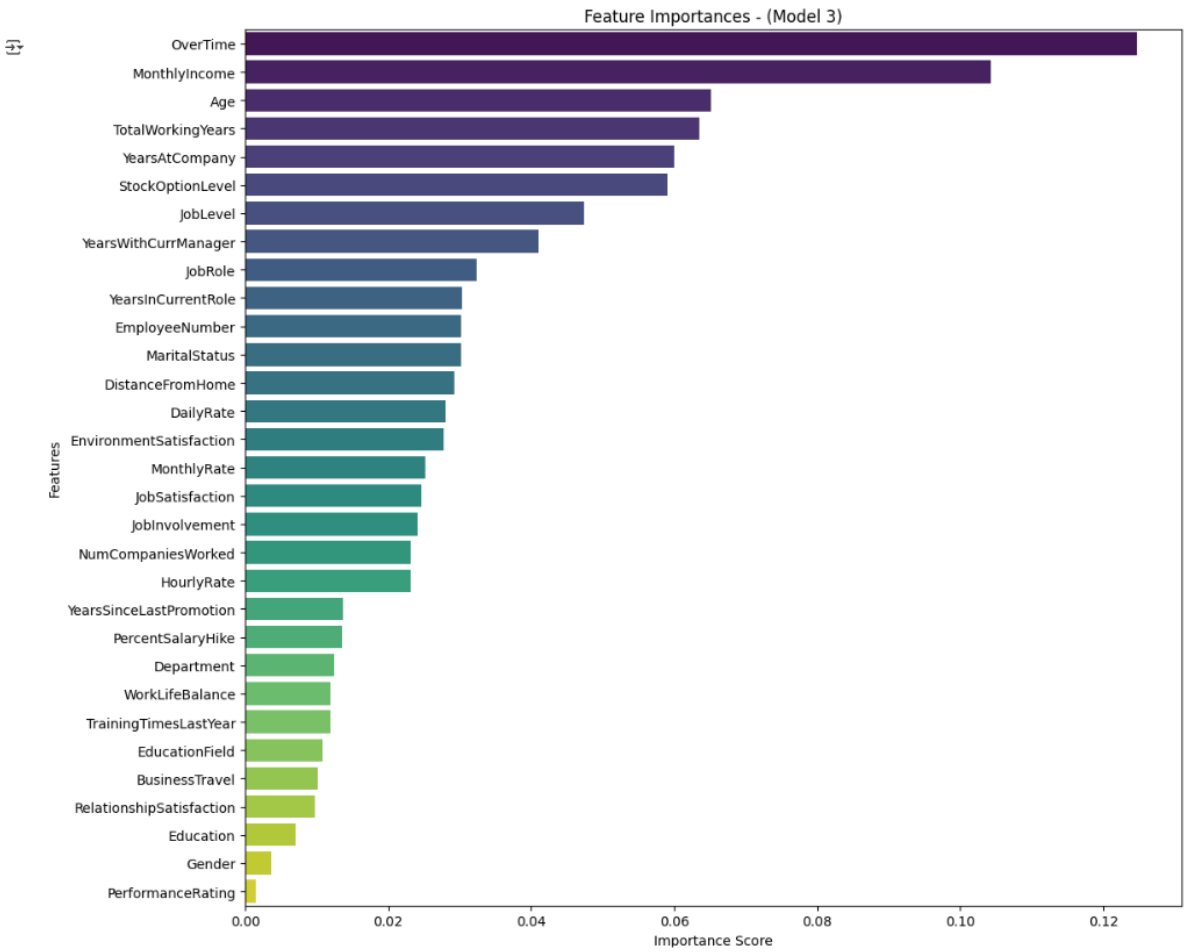
Model 3 Tuned Random Forest

คล้ายกับข้อสองแต่เปลี่ยนตัว estimator เป็น RandomForest

```
Model 3 Best Hyperparameters: {'max_depth': 5, 'max_features': 'sqrt', 'min_samples_split': 10, 'n_estimators': 200}

Model 3 Confusion Matrix:
[[1111  122]
 [ 117  120]]
```





จะเห็นว่า OverTime, MonthlyIncome มีสัดส่วนความสำคัญสูงมาก ส่วน Relation, Education, Gender, PerformanceRating สัดส่วนความสำคัญน้อยมาก หากจะพิจารณาความสำคัญในอนาคต ควรพิจารณาที่ OverTime, MonthlyIncome เป็นอย่างแรก

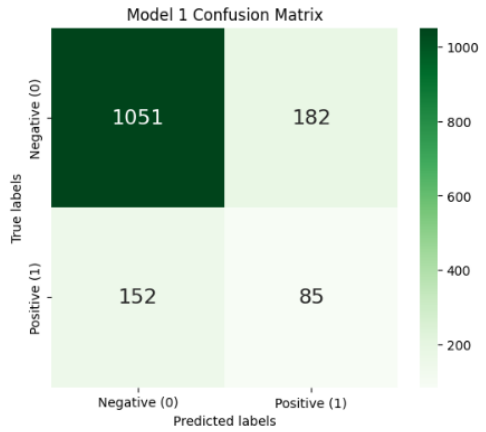
```
OverTime                    0.124710
MonthlyIncome               0.104264
Age                         0.065191
TotalWorkingYears           0.063529
YearsAtCompany              0.060034
StockOptionLevel            0.059032
JobLevel                    0.047376
YearsWithCurrManager        0.041055
JobRole                     0.032307
YearsInCurrentRole          0.030335
EmployeeNumber              0.030200
MaritalStatus               0.030188
DistanceFromHome            0.029230
DailyRate                   0.028019
EnvironmentSatisfaction     0.027811
MonthlyRate                 0.025200
JobSatisfaction             0.024678
JobInvolvement              0.024035
NumCompaniesWorked          0.023165
HourlyRate                  0.023141
YearsSinceLastPromotion     0.013695
PercentSalaryHike           0.013555
Department                  0.012476
WorkLifeBalance             0.011933
TrainingTimesLastYear       0.011901
EducationField              0.010819
BusinessTravel              0.010169
RelationshipSatisfaction    0.009793
Education                   0.007056
Gender                      0.003655
PerformanceRating           0.001448
dtype: float64
```

Analyze the results of the three models

Model 1 Default CART

```
Model 1 Tree Depth: 19

Model 1 Confusion Matrix:
[[1051  182]
 [ 152   85]]
```



เริ่มที่ model แรก ก็จะเห็นว่าข้อมูล มัน imbalance มากๆ ควรทำ sampling ให้มีฝั่ง 1, 0 เท่าๆกัน ไม่งั้นเราทาย 0 ทั้งหมด ก็ accuracy 81.84% แล้ว

Tree depth ที่ได้คือ 19 ชั้นซึ่งสูงมากมีโอกาสที่จะเป็น overfitting

พิจารณา Recall, Precision ของ confusion matrix

Recall = TP / (TP + FN) = 85 / (85 + 152) = 35.86%

ในบรรดาคนที่ลาออกจริง 237 คน model 1 ตรวจจับได้แค่ 85 คน และพลาด 152 คน ซึ่งไม่ดีมากๆ เพราะเราทำเพื่อดูการลาออกของ employee

Precision = TP/ (TP + FP) = 85 / (85 + 182) = 31.84%

ตัว precision ก็เช่นกัน ทายว่าจะลาออกผิดไปถึง 182 ครั้ง ซึ่งก็ไม่ดีมากๆ

Accuracy = (TP+TN)/Total = 1136 / 1470 = 77.28%

Specificity = TN / (TN + FP) = 1051 / 1233 = 85.24%

NPV = TN / (TN + FN) = 1051 / 1203 = 87.37%

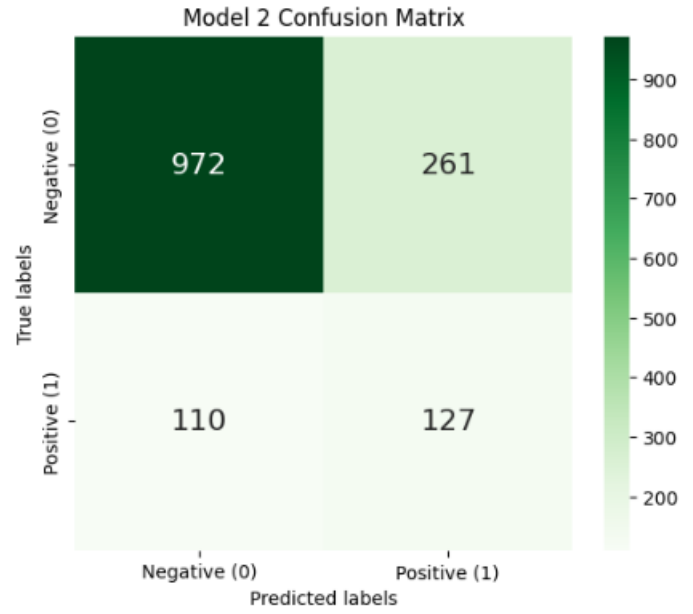
F1 = 2 * (0.3184 * 0.3586) / (0.3184 + 0.3586) = 33.72%

ไม่ได้พิจารณา Accuracy, Specificity, NPV เนื่องจาก data ที่ใช้ train imbalance มากๆ ค่าที่ได้มาจาก 3 ตัวคือ 77.28%, 85.24%, 87.37% ตามลำดับ จึงคิดว่าไม่ควรนำมาพิจารณา ดีหรือแย่ของ model

Model 2 Tuned CART

```
Model 2 Best Hyperparameters: {'max_depth': 5, 'min_samples_split': 20}

Model 2 Confusion Matrix:
[[972 261]
 [110 127]]
```



ได้ค่า Best Hyperparameter คือ max_depth : 5 และ min_samples_split: 20

พิจารณา Recall, Precision ของ confusion matrix

Recall = TP / (TP + FN) = 127 / (127 + 110) = 53.59%

ในบรรดาคนที่ลาออกจริง 237 คน model 2 ตรวจจับได้ 127 คน และพลาด 110 คน ซึ่งดีกว่า Default มาก

Precision = TP/ (TP + FP) = 127 / (127 + 261) = 32.73%

ตัว precision แม้จะดีขึ้นเล็กน้อยแต่ก็ทายว่าจะลาออกผิดไปถึง 261 ครั้ง

Accuracy = (TP+TN)/Total = 1099 / 1470 = 74.76%

Specificity = TN / (TN + FP) = 972 / 1233 = 78.83%

NPV = TN / (TN + FN) = 972 / 1082 = 89.83%

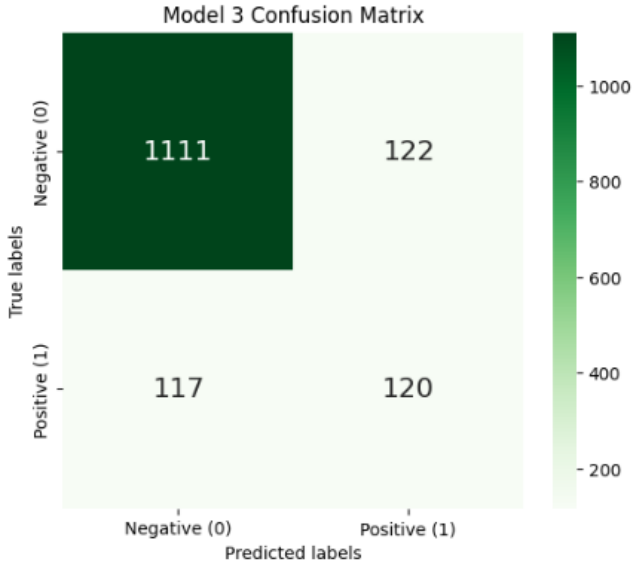
F1 = 2 * (0.3273 * 0.5359) / (0.3273 + 0.5359) = 40.64%

โดยรวมก็ยังพบว่า model 2 ที่ทำบน decision Tree แม้ recall จะดีขึ้นมาก อยู่ที่ครึ่งนึง แต่ก็เพราะว่ามีการเพิ่ม class_weight=”balance” ซึ่งแม้จะ GridSearchCV scoring ที่ recall แต่ precision ก็ไม่ได้ดีขึ้นตาม สักเกตว่า precision อยู่ที่ 32.73% ซึ่งมาจากการเพิ่มการทายว่า 1 มากขึ้น โดยรวม model 2 อาจจะดีขึ้นกว่า model 1 อยู่บ้าง ที่น่าสนใจนอกจาก Precision, Recall, F1 จะเห็นว่า Accuracy ต่ำลง เพราะว่ามี FP มากขึ้นสังเกตจาก 261 ที่โดดขึ้นมา

Model 3 Tuned Random Forest

```
Model 3 Best Hyperparameters: {'max_depth': 5, 'max_features': 'sqrt', 'min_samples_split': 10, 'n_estimators': 200}

Model 3 Confusion Matrix:
[[1111  122]
 [ 117  120]]
```



ได้ค่า Best Hyperparameter คือ max_depth : 5, max_features: sqrt, min_samples_split: 10 และ n_estimators: 200

พิจารณา Recall, Precision ของ confusion matrix

Recall = TP / (TP + FN) = 120 / (120+ 117) = 50.63%
พบว่า recall น้อยกว่า Model 2 แต่มากกว่า Model 1
ในบรรดาคนที่ลาออกจริง 237 คน model 3 ตรวจจับได้ 120 คน และพลาด 117 คน ซึ่งดีกว่า Default มาก
Precision = TP/ (TP + FP) = 120/ (120 +122) = 49.59%
ตัว precision จะเห็นได้ชัดว่าดีกว่าทั้ง model 1 และ model 2 อยู่มาก
Accuracy = (TP+TN)/Total = 1231 / 1470 = 83.74%

Specificity = TN / (TN + FP) = 1111 / 1233 = 90.11%

NPV = TN / (TN + FN) = 1111 / 1228 = 90.47%

F1 = 2 * (0.4959* 0.5063) / (0.4959 + 0.5063) = 50.10%

อีกทั้งใน Model ที่ 3 พบว่าทุกค่านอกเหนือจาก Recall, Precision, F1 ค่าอื่นๆเช่น Accuracy, Specificity, NPV ดีกว่า Model 1 และ Model 2 ทั้งหมด

Summary Table

Metric (ค่าชี้วัด)	Model 1 (Default CART)	Model 2 (Tuned CART)	Model 3 (Tuned RF)
Confusion Matrix	[[1051, 182], [152, 85]]	[[972, 261], [110, 127]]	[[1111, 122], [117, 120]]
Accuracy	77.28%	74.76%	83.74%
Recall	35.86%	53.59%	50.63%
Precision	31.84%	32.73%	49.59%
F1-Score	33.71%	40.64%	50.10%
Specificity	85.24%	78.83%	90.11%
NPV	87.37%	89.83%	90.47%

จาก Table และการสรุป model ที่ทำมาพบว่าข้อมูล Imbalance สูงมาก ควรทำ sampling oversampling class ที่น้อยและ undersampling class ที่มาก จึงทำให้ค่าอื่นๆเช่น Accuracy, Specificity, Negative Predictive Value ทุก model สูงทั้งหมด จึงพิจารณาในส่วนของ Recall, Precision ซึ่งเป็นส่วนหลักของการตัดสินคนลาออกของ model เป็นหลัก ซึ่งจากการทำทั้ง 3 model พบว่า Model 1 ที่เป็นค่า default ให้ผลลัพธ์ต่ำสุด Model Recall ดีขึ้นแต่ว่า Precision ยังไม่ดี และสุดท้าย Model 3 Recall, 50.63%, Precision 49.59% ซึ่งดีกว่าทั้ง 2 Model มาก อีกทั้ง Accuracy, Specificity, NPV ของ Model 3 ยังดีกว่า Model 1, 2 อีกด้วยเพราะเนื่องจากมีการทาย 1 มากขึ้น และที่ทาย 0 ออกตรงมากขึ้น จึงสรุปได้ว่า Model 3 เป็น Model ที่เหมาะที่สุดในการทำนาย Employee Attrition