



Present

การวิเคราะห์ความ  
แม่นยำในการทำนาย  
ของอุกกาบาต

(ฉบับต่อเติม)





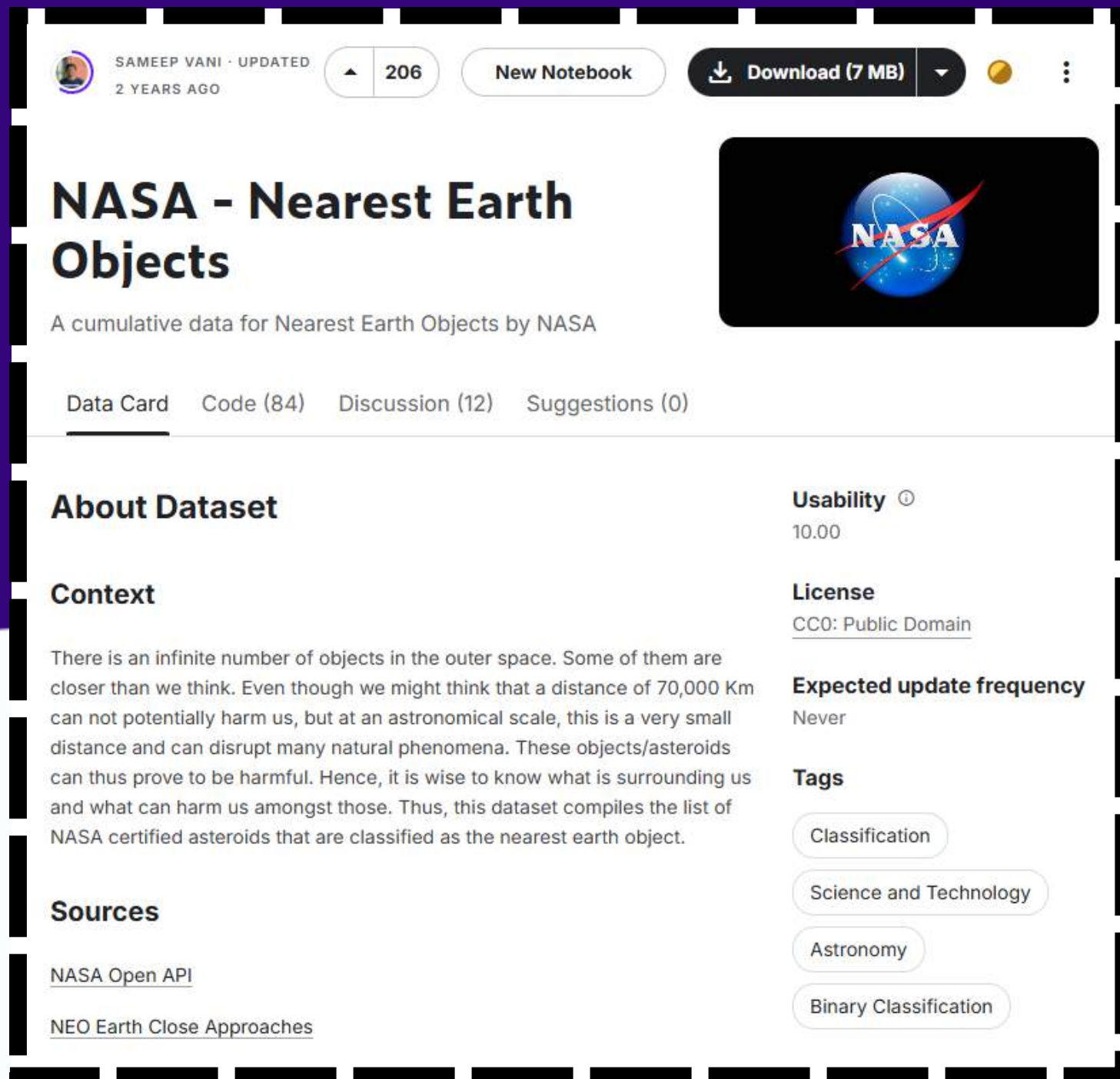
# จุดประสงค์

- นำโมเดลมาต่อเติม ช่วยวางแผนในการรับมือกับอุกกาบาตและข้อมูลใหม่ๆที่เพิ่มมาในอนาคต
- เพิ่มประสิทธิภาพความสามารถในการระบุความแม่นยำในการทำนายความอันตรายของอุกกาบาต
- เพื่อลดจำนวนพีเจอรี่ในการเก็บข้อมูล ทำให้ประหยัดค่าใช้จ่ายมากขึ้น



# DATASET

# FEATURE



The screenshot shows the Kaggle dataset page for "NASA - Nearest Earth Objects". At the top, it indicates the dataset was updated 2 years ago by SAMEEP VANI, has 206 versions, and a download size of 7 MB. The title "NASA - Nearest Earth Objects" is prominently displayed, followed by a subtitle "A cumulative data for Nearest Earth Objects by NASA" and a NASA logo. Below the title, there are tabs for "Data Card", "Code (84)", "Discussion (12)", and "Suggestions (0)". The "About Dataset" section includes a "Context" paragraph explaining the significance of near-Earth objects and their potential hazards. It also lists "Sources" as "NASA Open API" and "NEO Earth Close Approaches". On the right side, there are metadata fields: "Usability" (10.00), "License" (CC0: Public Domain), "Expected update frequency" (Never), and "Tags" (Classification, Science and Technology, Astronomy, Binary Classification).

- **id** : รหัสเฉพาะของอุกกาบาต
- **name** : ชื่อ หรือ หมายเลขที่กำหนดให้กับอุกกาบาต
- **est\_diameter\_min** : เส้นผ่านศูนย์กลางต่ำสุด
- **est\_diameter\_max** : เส้นผ่านศูนย์กลางสูงสุด
- **relative\_velocity** :
  - ความเร็วสัมพัทธ์ของอุกกาบาตเมื่อเทียบกับโลก
- **miss\_distance** :
  - ระยะห่างที่อุกกาบาตจะผ่านใกล้โลกที่สุด
- **orbiting\_body** :
  - วัตถุที่อุกกาบาตโคจรรอบ เช่น โลก (Earth) หรือดาวเคราะห์อื่น ๆ
- **sentry\_object** :
  - boolean (True/False) อุกกาบาตนี้รติดตามอย่างใกล้ชิดของ NASA หรือไม่ เนื่องจากมีโอกาสชนในอนาคต
- **absolute\_magnitude** :
  - ความสว่างสัมบูรณ์ โดยไม่คำนึงถึงระยะทางจากโลก
- **hazardous** :
  - boolean (True/False) จัดว่าอาจเป็นอันตรายต่อโลกหรือไม่



# Hyperparameter Tuning

ใช้เทคนิคการปรับค่าพารามิเตอร์ของ RandomForest เพื่อหาเซตที่ดีที่สุด  
ปรับค่า n\_estimators, max\_depth, min\_samples\_split โดยใช้วิธี GridSearchCV  
เพื่อค้นหาค่าพารามิเตอร์ที่ทำให้ผลลัพธ์ดีที่สุด  
จากนั้นเราก็ใช้ features selection เพื่อลดจำนวน feature ที่ใช้

```
7 from sklearn.model_selection import train_test_split
8 from sklearn.ensemble import RandomForestClassifier
9 from sklearn.metrics import accuracy_score, classification_report
10 from sklearn.feature_selection import RFE
11 from sklearn.model_selection import GridSearchCV
12 data = pd.read_csv('neo.csv')
13
14
15 X = data[['absolute_magnitude', 'est_diameter_min', 'est_diameter_max', 'relative_velocity', 'miss_distance']]
16 y = data['hazardous']
17
18 # Feature names can be extracted from the dataframe itself
19 feature_names = X.columns
20
21 # Split the dataset into training and testing sets
22 X_train, X_test, y_train, y_test = train_test_split(*arrays: X, y, test_size=0.2, random_state=42)
23 param_grid = {
24     'n_estimators': [50, 100, 25],
25     'max_depth': [None, 10, 20, 30],
26     'min_samples_split': [2, 3, 5]
27 }
28
29 grid_search = GridSearchCV(RandomForestClassifier(random_state=42), param_grid, cv=5)
30 grid_search.fit(X_train, y_train)
31
32 print(f'Best parameters: {grid_search.best_params_}')
33 best_model = grid_search.best_estimator_
34 y_pred_best = best_model.predict(X_test)
35 print(classification_report(y_test, y_pred_best))
```

Run

grid x



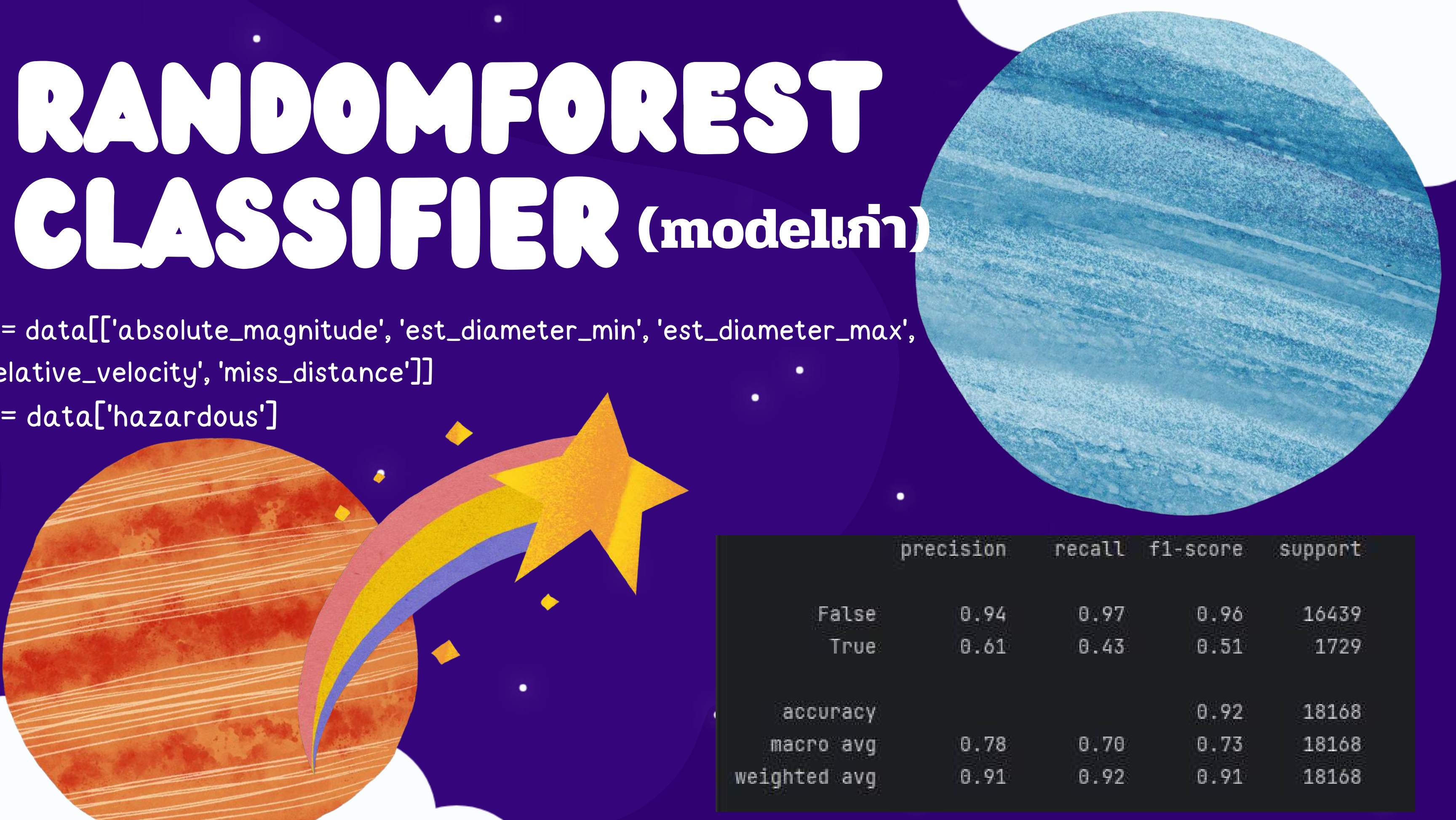
Best parameters: {'max\_depth': 20, 'min\_samples\_split': 3, 'n\_estimators': 50}

	precision	recall	f1-score	support
False	0.93	0.99	0.96	16439
True	0.71	0.31	0.43	1729
accuracy			0.92	18168
macro avg	0.82	0.65	0.69	18168
weighted avg	0.91	0.92	0.91	18168



# RANDOMFOREST CLASSIFIER (model)

```
X = data[['absolute_magnitude', 'est_diameter_min', 'est_diameter_max',  
'relative_velocity', 'miss_distance']]  
y = data['hazardous']
```



	precision	recall	f1-score	support
False	0.94	0.97	0.96	16439
True	0.61	0.43	0.51	1729
accuracy			0.92	18168
macro avg	0.78	0.70	0.73	18168
weighted avg	0.91	0.92	0.91	18168





# SELECTED FEATURES USING ANOVA (modelใหม่)

```
Selected features using ANOVA: Index(['absolute_magnitude', 'est_diameter_min', 'est_diameter_max'], dtype='object')
```

	precision	recall	f1-score	support
False	0.92	0.99	0.95	16439
True	0.64	0.22	0.32	1729
accuracy			0.91	18168
macro avg	0.78	0.60	0.64	18168
weighted avg	0.90	0.91	0.89	18168



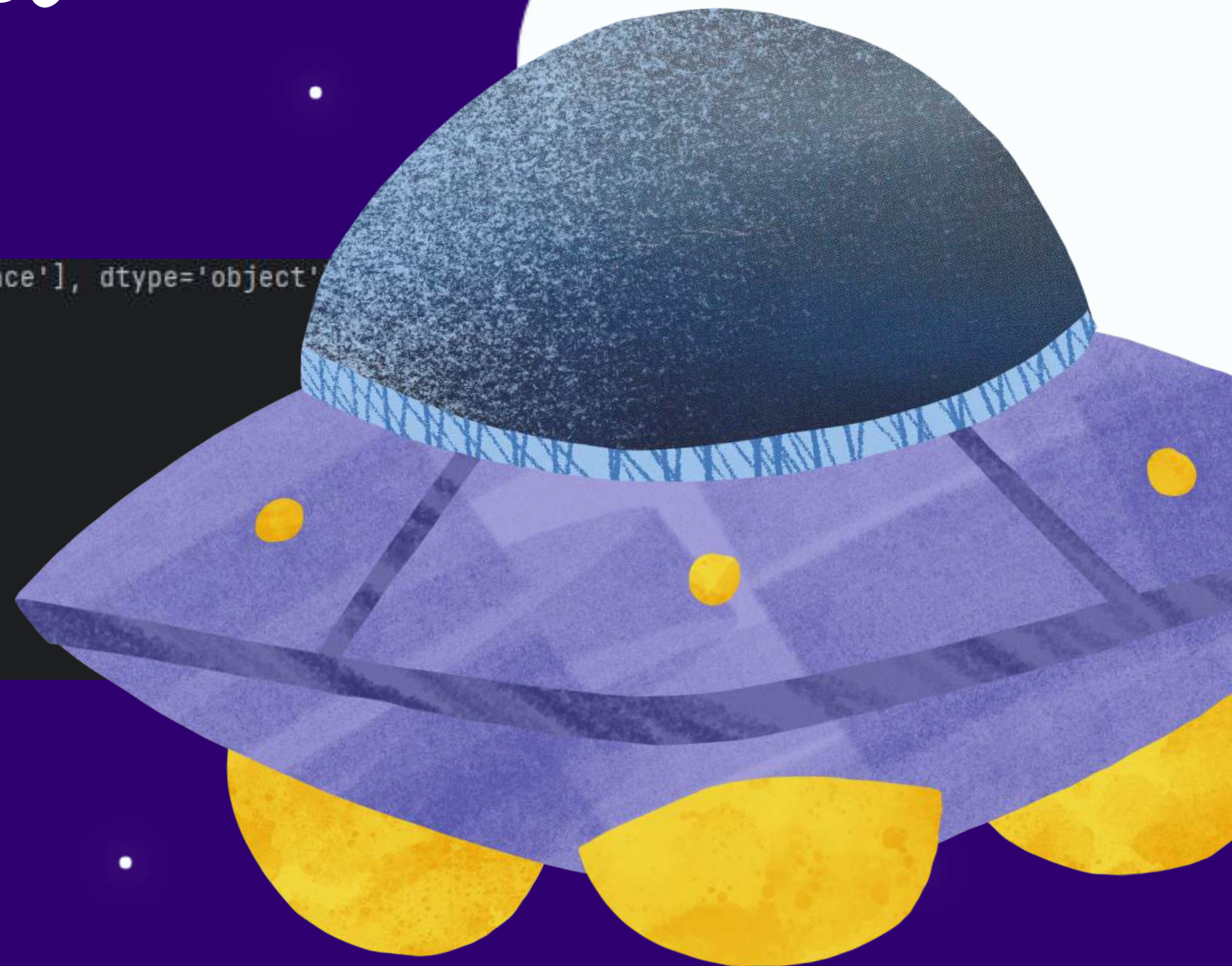
# RECURSIVE FEATURE ELIMINATION (RFE)

(model ใหม่)

Selected features using RFE: Index(['est\_diameter\_min', 'relative\_velocity', 'miss\_distance'], dtype='object')

	precision	recall	f1-score	support
False	0.93	0.98	0.96	16439
True	0.63	0.32	0.43	1729
accuracy			0.92	18168
macro avg	0.78	0.65	0.69	18168
weighted avg	0.90	0.92	0.91	18168

(ผลลัพธ์ที่ดีที่สุด)





## ข้อดี-ข้อเสีย

ของ Feature selection (ตัด feature) แต่ละตัวที่เลือกใช้

### Filter : ANOVA

ดูความสามารถของ Feature รายตัว

**ข้อดี** : ง่ายและรวดเร็ว ไม่ซับซ้อน เหมาะสำหรับปัญหาการจำแนกประเภท (Classification)

**ข้อเสีย** : ไม่เหมาะกับข้อมูลที่มีซับซ้อน

ไม่รองรับฟีเจอร์แบบเชิงหมวดหมู่ ไม่สามารถจัดการกับความสัมพันธ์ระหว่างฟีเจอร์

### Wrapper : Recursive Feature Elimination (RFE)

เทคนิคเลือกฟีเจอร์ที่ใช้การฝึกโมเดลแบบวนซ้ำ สังเกตพลังการทำนายของกลุ่ม Feature หลายๆตัว

**ข้อดี** : จับความสัมพันธ์ระหว่างฟีเจอร์ต่างๆ ได้ดี เหมาะกับโมเดลที่ซับซ้อน รองรับฟีเจอร์หลายประเภททั้งตัวเลขและหมวดหมู่

**ข้อเสีย** : ใช้เวลาประมวลผลมากกว่า ไม่เหมาะกับข้อมูลที่มีฟีเจอร์จำนวนมาก



# ข้อดี-ข้อเสีย ของ Hyperparameter Tuning



**ข้อดี** : ช่วยให้โมเดลมีความแม่นยำสูงขึ้น โดยการหาชุดพารามิเตอร์ที่เหมาะสมที่สุดสำหรับโมเดลนั้นๆ, เพิ่มความสามารถในการทำนาย, ช่วยให้เข้าใจความสำคัญของพารามิเตอร์, ปรับโมเดลให้เข้ากับข้อมูลเฉพาะ, ช่วยให้เข้าใจความสำคัญของพารามิเตอร์

**ข้อเสีย** : ใช้เวลามาก, ใช้ทรัพยากรสูง(oh my PC!!) , อาจเกิด Overfitting ได้หากจูนพารามิเตอร์มากเกินไป, มีความไม่แน่นอนขึ้นอยู่กับลักษณะของข้อมูลที่ใช้







# ผลลัพธ์การเทรนด้วย Selected Feature

หลังจากได้ลองใช้ selected feature  
ได้ผลสรุปว่าได้ความแม่นยำในการทำนายเท่าเดิม  
จึงทดลองลดจำนวนฟีเจอร์ที่ใช้ลงทำให้ความแม่นยำ  
ในการทำนายลดลงเล็กน้อย แต่สามารถลด  
ทรัพยากรที่ใช้งานได้



# ประโยชน์

- ช่วยให้ทำนายอุกกาบาตได้แม่นยำขึ้น  
ทำให้วางแผนรับมือได้ดียิ่งขึ้น
- ช่วยในการคาดการณ์อุกกาบาตลูก  
ใหม่ๆที่เพิ่งค้นพบได้
- ช่วยลดทรัพยากรที่ต้องใช้ได้





# สรุป

หลังจากได้นำโมเดลมาทดลองและต่อยอดด้วยวิธีต่างๆจึงได้  
ข้อสรุปว่า

ถ้าต้องการความแม่นยำที่ดีที่สุดก็จะใช้ feature 5 ตัว แต่ถ้าหากอยากลด  
จำนวน feature ที่ต้องใช้ก็ใช้วิธีการคัดเลือก feature ที่ดีที่สุดมาจาก RFE



# รายชื่อสมาชิก

นาย กัญจน์ จรัสโรจนพร 650710213

นาย จีราวัฒน์ ศิริยศลักษณ์ 650710534

นาย วรณพ ลิ้มปี่ปิตีวรกุล 650710579

นาย วรเมธ อภิวังโสกุล 650710580

