



## รายงาน

เรื่อง การพยากรณ์ราคาดัชนีหุ้นเพื่อคาดการณ์ผลตอบแทนจากกลยุทธ์

## จัดทำโดย

นางสาวณัฐชา	สุภาพจันทร์	62090500406
นางสาวธัญนิชา	บวรวิวัฒน์ชัย	62090500411
นางสาวสุพิชชา	จำปาทอง	62090500424
นายสหัสวรรษ	ประคอง	62090500440
นางสาวเกวรินทร์	เจดีย์สถาน	62090500444
นายณพคุณ	อนันตกิจถาวร	62090500447

## เสนอ

รศ.ชูเกียรติ วรรณสุชีพ

ภาคเรียนที่ 1 ปีการศึกษา 2564

รายวิชา CSS 341 Introduction to Data Science

คณะวิทยาศาสตร์ ภาควิชาคณิตศาสตร์ สาขาวิทยาการคอมพิวเตอร์ประยุกต์

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

## บทคัดย่อ

การศึกษานี้มีวัตถุประสงค์เพื่อพยากรณ์ราคาดัชนีหุ้น ได้แก่ Dow Jones Industrial Average, Nikkei 225 Stock Average, Heng Seng Index และ SSE Composite Index และคำนวณผลตอบแทนจากการลงทุนด้วยกลยุทธ์การลงทุนถัวเฉลี่ยต้นทุนอย่างต่อเนื่องเป็นเวลา 60 เดือน ตั้งแต่เดือนมกราคม 2016 ถึง ธันวาคม 2021 โดยใช้เทคนิค Classification และ Regression ในการพยากรณ์ความเป็นไปได้ของราคาดัชนีหุ้นแล้วคำนวณเป็นอัตราผลตอบแทนของดัชนีหุ้น ผลการศึกษา การพยากรณ์ราคาดัชนีหุ้นพบว่า เทคนิค Classification มีความสะดวกและเหมาะสมต่อการพยากรณ์ราคาดัชนีหุ้นมากที่สุดเนื่องจากมีการประเมินความแม่นยำที่สะดวกและดีที่สุด และการคาดการณ์ผลตอบแทนของดัชนีหุ้น โดย ณ วันที่ 6 ธันวาคม 2021 ราคา Dow Jones Industrial Average และราคา Heng Seng Index มีแนวโน้มที่จะปรับตัวลดลง และราคา Nikkei 225 Stock Average และราคา SSE Composite Index มีแนวโน้มการปรับตัวขึ้น ซึ่งพบว่าดัชนีหุ้นที่ควรลงทุนในวันนี้คือ Dow Jones Industrial Average และ Heng Seng Index ส่วนดัชนีหุ้นที่ควรขายในวันนี้คือ Nikkei 225 Stock Average และ SSE Composite Index

**คำสำคัญ :** การพยากรณ์ดัชนีของหุ้น, ผลตอบแทนของการลงทุน



## บทนำ

การลงทุน หรือ การเล่นหุ้น คือการซื้อหุ้นของบริษัทที่จดทะเบียนในตลาดหลักทรัพย์ การเทรดหุ้นมีอยู่ด้วยกัน 2 ตลาด คือ ตลาดแรก การซื้อขายในตลาดแรก หรือการเทรดหุ้น IPO (Initial Public Offering) โดยราคาหุ้นจะถูกกำหนดไว้ให้นักลงทุนมาจับจอง ในการซื้อหุ้น IPO นั้นจะต้องจองซื้อผ่านผู้จัดจำหน่ายเท่านั้น และตลาดรอง ซึ่งเป็นการซื้อขายหุ้นในตลาดหลักทรัพย์ ราคาของหุ้นในตลาดหลักทรัพย์จะเปลี่ยนแปลงตามผลการดำเนินการของบริษัท และสถานะตลาดตามหลักของ demand supply

ปัจจุบันการลงทุนเป็นการเพิ่มมูลค่าของเงินในอีกช่องทางหนึ่ง ผู้คนส่วนใหญ่นิยมการลงทุนหลากหลายรูปแบบ ซึ่งการลงทุนการซื้อ-ขาย หุ้น เป็นช่องทางที่ได้รับความนิยม จากเงินทุนเป็นผลกำไรหรือผลตอบแทน ในการลงทุนซื้อขายหุ้น จำเป็นที่จะต้องมีความรู้ด้านการเงินการลงทุน ความพร้อมทางด้านการเงินของตนเอง ศึกษารายละเอียดของหุ้นที่จะซื้อ ติดตามข่าวสารหุ้นอยู่เสมอ เพื่อลดความเสี่ยงที่จะเกิดขึ้น ในระหว่างการลงทุน อีกทั้งยังต้องรู้จักการคำนวณและหาสาเหตุที่ทำให้ราคาหุ้นมีการเคลื่อนไหวขึ้นหรือลง

การศึกษานี้จึงได้มีการนำเอาเทคนิค Classification และ Regression ในการพยากรณ์ราคาดัชนีหุ้น Dow Jones Industrial Average, Nikkei 225 Stock Average, Heng Seng Index และ SSE Composite Index เพื่อคำนวณความเป็นไปได้ของราคาดัชนีหุ้น โดยคำนวณผลตอบแทนจากกลยุทธ์การลงทุนถัวเฉลี่ยต้นทุนอย่างต่อเนื่องเป็นเวลา 60 เดือน ตั้งแต่เดือนมกราคม 2016 ถึง ธันวาคม 2021 แล้วนำมาเปรียบเทียบผลตอบแทนจากราคาหุ้นที่เกิดขึ้นจริง เพื่อแสดงให้เห็นเป็นแนวทางในการตัดสินใจของนักลงทุน หรือผู้ที่สนใจลงทุนต่อไป

## วัตถุประสงค์

1. เพื่อพยากรณ์ราคาดัชนีหุ้น ได้แก่ Dow Jones Industrial Average, Nikkei 225 Stock Average, Heng Seng Index และ SSE Composite Index
2. คำนวณผลตอบแทนจากการลงทุนด้วยกลยุทธ์การลงทุนถัวเฉลี่ยต้นทุนอย่างต่อเนื่องเป็นเวลา 60 เดือน ตั้งแต่เดือนมกราคม 2016 ถึง ธันวาคม 2021

## วิธีดำเนินการ

### ข้อมูลและตัวแปร

การศึกษานี้จะใช้ข้อมูลตั้งแต่เดือนมกราคม 2016 ถึง ธันวาคม 2021 ของดัชนีหุ้นจำนวน 4 ตัว จากเว็บไซต์ของ Yahoo finance เพื่อสร้างตัวแบบพยากรณ์ แล้วนำไปคำนวณผลตอบแทนการลงทุนในอนาคตของดัชนีหุ้นแต่ละตัว โดยหุ้นที่ใช้ในการวิเคราะห์ ประกอบด้วย (1) Dow Jones Industrial Average หรือ DJI (2) Nikkei 225 Stock Average หรือ N225 (3) Heng Seng Index หรือ HSI และ (4) SSE Composite Index หรือ SSE

## การวิเคราะห์ข้อมูล

### 1. Data Preparation

- 1) เริ่มด้วยการดึงข้อมูลดัชนีหุ้นมาจาก Yahoo โดยใช้ API โดยดึงข้อมูลตั้งแต่วันที่ 2016-01-01 ถึง 2021-12-31 เป็นเวลา 5 ปี
- 2) หา Technical Indicator โดยนำข้อมูลมาประมวลผลโดยใช้ Library ta-Lib

### 2. Feature Selection

- 1) นำเอาราคาปิดของวันนี้และวันต่อมามาหาเปอร์เซ็นต์ความแตกต่างว่าหุ้นขึ้นหรือลง เพื่อใช้เป็น Target สำหรับ Classification  
สูตรในการหาเปอร์เซ็นต์ความต่างของหุ้น

$$\left( \frac{\text{Today's close price} - \text{Yesterday's close price}}{\text{Yesterday's close price}} \right) \times 100 \quad (1)$$

- 2) นำเอาราคาปิดของวันพรุ่งนี้มาใช้เป็น Target สำหรับ Regression
- 3) ใช้การ Feature Selection แบบ Recursive Feature Elimination หรือ RFE มาวิเคราะห์หา Feature สำหรับทั้ง Classification และ Regression โดยใช้ Target คือหาว่า ราคาหุ้นเพิ่มขึ้นหรือลดลงจากเมื่อวานและ Re Target (Regression Target) คือราคาปิดของวันถัดมา เป็นตัว Target ในการวิเคราะห์ข้อมูล โดย RFE จะเลือก Feature ที่ควรจะใช้มาเองไม่ได้จำกัดว่าต้องมีกี่ตัวแต่อย่างใด โดยหลังจากการทำ Feature Selection ได้ผลลัพธ์ว่าเหลือ Feature ทั้งหมด 17 ตัว

### 3. Evaluation

#### Classification

- 1) Accuracy

$$\frac{\text{correct predictions}}{\text{total predictions}} \times 100 \quad (2)$$

- 2) Precision

$$\frac{TP}{(TP + FP)} \quad (3)$$

True Positive (TP) = สิ่งที่ทำนายตรงกับสิ่งที่เกิดขึ้นจริง ในกรณี ทำนายว่าจริง และสิ่งที่เกิดขึ้น ก็คือ จริง

True Negative (TN) = สิ่งที่ทำนายตรงกับสิ่งที่เกิดขึ้น ในกรณี ทำนายว่า ไม่จริง และสิ่งที่เกิดขึ้น ก็คือ ไม่จริง

False Positive (FP) = สิ่งที่ทำนายไม่ตรงกับสิ่งที่เกิดขึ้น คือทำนายว่า จริง แต่สิ่งที่เกิดขึ้น คือ ไม่จริง

False Negative (FN) = สิ่งที่ทำนายไม่ตรงกับที่ที่เกิดขึ้นจริง คือทำนายว่าไม่จริง แต่สิ่งที่เกิดขึ้น คือ จริง

### 3) Recall

$$\frac{TP}{(TP + FN)} \quad (4)$$

True Positive (TP) = สิ่งที่ทำนายตรงกับสิ่งที่เกิดขึ้นจริง ในกรณี ทำนายว่าจริง และสิ่งที่เกิดขึ้น ก็คือ จริง

True Negative (TN) = สิ่งที่ทำนายตรงกับสิ่งที่เกิดขึ้น ในกรณี ทำนายว่า ไม่จริง และสิ่งที่เกิดขึ้น ก็คือ ไม่จริง

False Positive (FP) = สิ่งที่ทำนายไม่ตรงกับสิ่งที่เกิดขึ้น คือทำนายว่า จริง แต่สิ่งที่เกิดขึ้น คือ ไม่จริง

False Negative (FN) = สิ่งที่ทำนายไม่ตรงกับที่ที่เกิดขึ้นจริง คือทำนายว่าไม่จริง แต่สิ่งที่เกิดขึ้น คือ จริง

### 4) F1-score

$$2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \quad (5)$$

## Regression

### 5) Root Mean Squared Error

$$\sqrt{((predictions - targets) \times 2).mean()} \quad (6)$$

### 6) R-Squared

$$1 - \frac{(total\ sum\ of\ residuals)}{(total\ sum\ of\ squares)} \quad (7)$$

## 4. สร้างโมเดลในการพยากรณ์ข้อมูล

1) การพยากรณ์ด้วยวิธี Decision Tree เป็น model แบบ rule-based คือ สร้างกฎ if-else จากค่าของแต่ละ feature โดยไม่มีสมการมากำกับความสัมพันธ์ระหว่าง feature & target สิ่งที่สำคัญในการสร้าง Decision Tree คือ การเลือก split ค่า feature แต่ละครั้ง จะต้อง minimize ค่าของ cost functionให้น้อยที่สุด

วิธีการทำ Decision Tree คือ การค่อยๆ แบ่งข้อมูลออกทีละ 2 ส่วน (recursive binary split) จาก node ล่างสุดของ tree เรียกว่า root node และไล่ขึ้นมาเรื่อย ๆ จนถึง leaf และทำ prediction ค่า target variable ด้วยวิธีการง่ายๆ คือ ใช้ค่า mean ของ target variable node โดยการ split ข้อมูลจาก root node จนถึง leaf node จะทำจนกว่าจะได้ condition ที่กำหนด

หลักการในการแบ่งข้อมูลในแต่ละ node สำหรับข้อมูลที่มี k feature และ n observation มีดังนี้ เลือก 1 feature จาก k feature มาทำ sorting ข้อมูล ด้วยค่าของ feature ที่เลือกมา หาจุดแบ่งข้อมูล (split point) ที่เป็นไปได้ทั้งหมด จากข้อมูล n observation สามารถหาจุดแบ่งข้อมูลที่เป็นไปได้ n-1 จุด สำหรับการแบ่งข้อมูลแต่ละแบบที่เป็นไปได้คำนวณค่า Gini impurity ซึ่งเป็นการวัดความไม่บริสุทธิ์ หรือความไม่เพียวของ class ในแต่ละกลุ่มข้อมูลที่แบ่งตามแต่ละ split point สำหรับปัญหา classification แบบ binary ที่มี target variable เป็น 0 หรือ 1 การ split ที่ดี ควรจะได้กลุ่มข้อมูลออกมา 2 กลุ่มที่สามารถแยก class 0 กับ class 1 ออกมาได้ชัดเจนในแต่ละกลุ่มยังสามารถแบ่งแยก class ของ target variable ออกมาได้ดี ค่า Gini impurity ก็จะมีค่าต่ำ เมื่อสิ้นสุดการ split แล้ว จะ predict ค่า target

$$G = \sum_{k=1}^K \hat{p}_{mk}(1 - \hat{p}_{mk}) \quad (8)$$

ซึ่งมีปรับและเซตค่าพารามิเตอร์ ดังนี้

```
dtc = DecisionTreeClassifier(criterion="entropy")
dtc.fit(X_train, y_train)
y_pred = dtc.predict(X_test)
```

Figure 1. ปรับและเซตค่าพารามิเตอร์ Decision Tree

2) การพยากรณ์ด้วยวิธี Random Forest คือ แนวคิดของ Random Forest นี้คือการสร้างโมเดล ด้วยวิธีการ Decision Tree ขึ้นมาหลายๆ โมเดล โดยวิธีการสุ่มตัวแปร แล้วนำผลที่ได้แต่ละโมเดลมารวมกัน พร้อมนับจำนวนผลที่มีจำนวนซ้ำกันมากที่สุด สกัดออกมาเป็นผลลัพธ์สุดท้ายด้วยวิธีการ ของ Decision Tree คือเทคนิคที่ให้ผลลัพธ์ในลักษณะเป็นโครงสร้างของต้นไม้ภายในต้นไม้จะประกอบไปด้วยโหนด (node) ซึ่งแต่ละโหนดจะมีเงื่อนไขของคุณลักษณะเป็นตัว ทดสอบกิ่งของต้นไม้ (branch) แสดงถึงค่าที่เป็นไปได้ของคุณลักษณะที่ถูกเลือกทดสอบ และใบ (leaf) เป็นสิ่งที่อยู่ล่างสุดของต้นไม้แสดงถึงกลุ่มของข้อมูล (class) ก็คือผลลัพธ์ที่ได้จากการพยากรณ์ ซึ่งข้อดีของวิธีการนี้คือให้ผลการพยากรณ์ที่ แม่นยำและเกิดปัญหา overfitting น้อย ซึ่งมีการปรับค่าพารามิเตอร์ ดังนี้

```

forest = RandomForestClassifier(n_estimators = 1000,
random_state = 42, max_features=9)

forest.fit(X_train, y_train)

y_pred = forest.predict(X_test)

```

Figure 2. ปรับและเซตค่าพารามิเตอร์ Random Forest

3) การพยากรณ์ด้วยวิธี Logistic Regression คือ เป็นเทคนิคทางสถิติภายใต้การดูแลเพื่อค้นหาความน่าจะเป็นของตัวแปรตาม (คลาสที่มีอยู่ในตัวแปร) และสร้างสมการคณิตศาสตร์เพื่อแบ่งแยก (classify) ข้อมูลออกเป็น 2 กลุ่มคำตอบ

$$h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T x}} \quad (9)$$

ซึ่งจะมีการปรับและเซตค่าพารามิเตอร์ ดังนี้

```

logistic = LogisticRegression()

```

Figure 3. ปรับและเซตค่าพารามิเตอร์ Logistic Regression

4) การพยากรณ์ด้วยวิธี XGBoost เป็น model ที่นำเอา Decision Tree มา train ต่อ ๆ กันหลาย ๆ tree โดยที่แต่ละ decision tree จะเรียนรู้จาก error ของ tree ก่อนหน้าทำให้ความแม่นยำในการทำ prediction จะแม่นยำมากขึ้นเรื่อยๆ เมื่อมีการเรียนรู้ของ tree ต่อเนื่องกันจนมีความลึกมากพอ และ model จะหยุดเรียนรู้เมื่อไม่เหลือ pattern ของ error จาก tree ก่อนหน้าให้เรียนรู้แล้ว ทั้ง Random Forrest และ XGBoost เป็น model แบบ ensemble คือ ใช้ model หลายๆ model มาประกอบกันเป็น model ที่ซับซ้อน ซึ่งจะมีการปรับและเซตค่าพารามิเตอร์ ดังนี้

```

xgb = XGBClassifier()

```

Figure 4. ปรับและเซตค่าพารามิเตอร์ Logistic Regression

5) การพยากรณ์ด้วยวิธี Linear Regression ก็คือ การ Fit ข้อมูลด้วย “เส้นตรง หรือ Linear” ในการหาเส้นตรงที่จะใช้ในการสร้างโมเดลทำนายนี้จะต้องมีการคำนวณเพื่อหาฟังก์ชันเส้นตรงที่จะฟิต(พอดี)ไปกับข้อมูลได้ดีที่สุด ฟังก์ชันเส้นตรงพื้นฐาน ก็คือ

$$y = b_0 + b_1 \times x_1 \quad (10)$$



ซึ่งจะมีการปรับและเซตค่าพารามิเตอร์ ดังนี้

```
ls = LinearRegression(fit_intercept=True)
lr.fit(X_train, y_train)
y_pred = lr.predict(X_test)
```

Figure 5. ปรับและเซตค่าพารามิเตอร์ Linear Regression

6) การพยากรณ์ด้วยวิธี Polynomial Regression คือ เป็นเทคนิคการพยากรณ์ที่พยายามอธิบายพฤติกรรมของข้อมูล โดยเรามีสมมติฐานที่ว่า ข้อมูลไม่ได้สัมพันธ์กันเป็นเส้นตรง ในการหาความสัมพันธ์เส้นตรง ของสมการ  $y = ax + b$ . สิ่งที่เราสนใจจริงๆ คือ การหาค่าสัมประสิทธิ์  $a$  ที่เหมาะสม

$$y = \alpha + \beta_1 x + \beta_1 x^2 \quad (11)$$

ซึ่งจะมีการปรับและเซตค่าพารามิเตอร์ ดังนี้

```
Pr = make_pipeline(PolynomialFeatures(degree),
LinearRegression())
pr.fit(X_train,y_train)
```

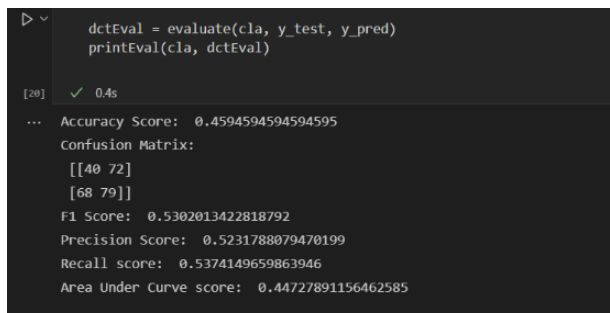
Figure 6. ปรับและเซตค่าพารามิเตอร์ Polynomial Regression

ผลจากการทดลอง

1. Dow Jones Industrial Average.

### Classification

- Decision Tree



```
dctEval = evaluate(c1a, y_test, y_pred)
printEval(c1a, dctEval)
```

[20] ✓ 0.45

... Accuracy Score: 0.4594594594594595  
Confusion Matrix:  
[[40 72]  
[68 79]]  
F1 Score: 0.5302013422818792  
Precision Score: 0.5231788079470199  
Recall score: 0.5374149659863946  
Area Under Curve score: 0.44727891156462585

Figure 7. ผลการทดลอง Decision Tree จากดัชนีหุ้น Dow Jones Industrial Average

- Random Forest

```

▷ forestEval = evaluate(cla, y_test, y_pred)
  printEval(cla, forestEval)
[24] ✓ 0.3s
... Accuracy Score: 0.5675675675675675
  Confusion Matrix:
    [[ 0 112]
     [ 0 147]]
  F1 Score: 0.7241379310344828
  Precision Score: 0.5675675675675675
  Recall score: 1.0
  Area Under Curve score: 0.5

```

Figure 8. ผลการทดลอง Random Forest จากดัชนีหุ้น Dow Jones Industrial Average

- XGBoost

```

▷ xgbEval = evaluate(cla, y_test, y_pred)
  printEval(cla, xgbEval)
[32] ✓ 0.4s
... Accuracy Score: 0.5019305019305019
  Confusion Matrix:
    [[36 76]
     [53 94]]
  F1 Score: 0.5930599369085174
  Precision Score: 0.5529411764705883
  Recall score: 0.6394557823129252
  Area Under Curve score: 0.4804421768707483

```

Figure 9. ผลการทดลอง XGBoost จากดัชนีหุ้น Dow Jones Industrial Average

- Logistic Regression

```

▷ logisticEval = evaluate(cla, y_test, y_pred)
  printEval(cla, logisticEval)
[28] ✓ 0.3s
... Accuracy Score: 0.5444015444015444
  Confusion Matrix:
    [[ 18 94]
     [ 24 123]]
  F1 Score: 0.6758241758241758
  Precision Score: 0.5668202764976958
  Recall score: 0.8367346938775511
  Area Under Curve score: 0.4987244897959184

```

Figure 10. ผลการทดลอง Logistic Regression จากดัชนีหุ้น Dow Jones Industrial Average

## Regression

- Linear Regression



Figure 11. ผลการทดลอง Linear Regression จากดัชนีหุ้น Dow Jones Industrial Average

- Polynomial Regression



Figure 12. ผลการทดลอง Polynomial Regression จากดัชนีหุ้น Dow Jones Industrial Average

## 2. Nikkei 225 Stock Average.

Classification

- Decision Tree

```
dctEval = evaluate(c1a, y_test, y_pred)
printEval(c1a, dctEval)
```

[65] ✓ 0.3s

... Accuracy Score: 0.504  
 Confusion Matrix:  
 [[60 47]  
 [77 66]]  
 F1 Score: 0.5156249999999999  
 Precision Score: 0.584070796460177  
 Recall score: 0.46153846153846156  
 Area Under Curve score: 0.5111430625449318

Figure 13. ผลการทดลอง Decision Tree จากดัชนีหุ้น Nikkei 225 Stock Average.

- Random Forest

```
forestEval = evaluate(c1a, y_test, y_pred)
printEval(c1a, forestEval)
```

[69] ✓ 0.3s

... Accuracy Score: 0.48  
 Confusion Matrix:  
 [[57 50]  
 [80 63]]  
 F1 Score: 0.49218749999999994  
 Precision Score: 0.5575221238938053  
 Recall score: 0.4405594405594406  
 Area Under Curve score: 0.4866348604666362

Figure 14. ผลการทดลอง Random Forest จากดัชนีหุ้น Nikkei 225 Stock Average.

- XGBoost

```
xgbEval = evaluate(c1a, y_test, y_pred)
printEval(c1a, xgbEval)
```

[77] ✓ 0.3s

... Accuracy Score: 0.48  
 Confusion Matrix:  
 [[57 50]  
 [80 63]]  
 F1 Score: 0.49218749999999994  
 Precision Score: 0.5575221238938053  
 Recall score: 0.4405594405594406  
 Area Under Curve score: 0.4866348604666362

Figure 15. ผลการทดลอง XGBoost จากดัชนีหุ้น Nikkei 225 Stock Average.

- Logistic Regression

```

logisticEval = evaluate(c1a, y_test, y_pred)
printEval(c1a, logisticEval)

[73] ✓ 0.3s

... Accuracy Score: 0.472
Confusion Matrix:
[[46 61]
 [71 72]]
F1 Score: 0.5217391304347826
Precision Score: 0.5413533834586466
Recall score: 0.5034965034965035
Area Under Curve score: 0.4667015227762892

```

Figure 16. ผลการทดลอง Logistic Regression จากดัชนีหุ้น Nikkei 225 Stock Average.

## Regression

- Linear Regression



Figure 17. ผลการทดลอง Linear Regression จากดัชนีหุ้น Nikkei 225 Stock Average.

- Polynomial Regression

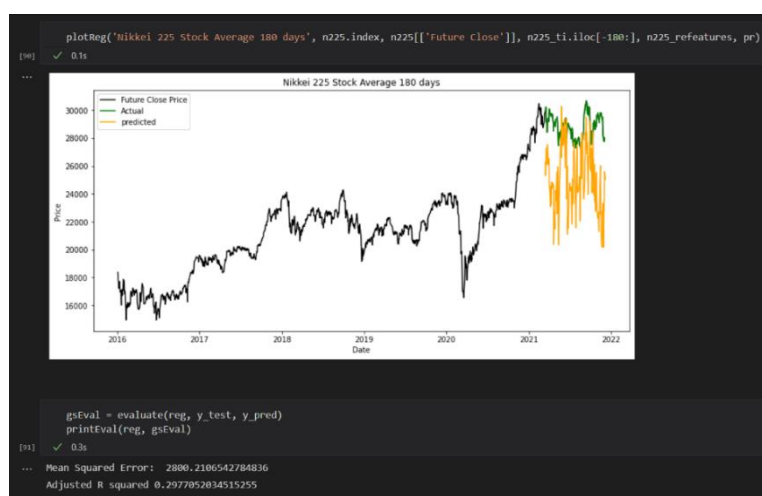


Figure 18. ผลการทดลอง Polynomial Regression จากดัชนีหุ้น Nikkei 225 Stock Average.

## 3. Heng Seng Index.

Classification

- Decision Tree

```
dctEval = evaluate(cla, y_test, y_pred)
printEval(cla, dctEval)
```

[104] ✓ 0.3s

... Accuracy Score: 0.44841269841269843

Confusion Matrix:

```
[[45 73]
 [66 68]]
```

F1 Score: 0.4945454545454546

Precision Score: 0.48226950354609927

Recall score: 0.5074626865671642

Area Under Curve score: 0.444409309385277

Figure 19. ผลการทดลอง Decision Tree จากดัชนีหุ้น Heng Seng Index.

- Random Forest

```
forestEval = evaluate(cla, y_test, y_pred)
printEval(cla, forestEval)
```

[107] ✓ 0.3s

... Accuracy Score: 0.5198412698412699

Confusion Matrix:

```
[[ 7 111]
 [ 10 124]]
```

F1 Score: 0.6720867208672087

Precision Score: 0.5276595744680851

Recall score: 0.9253731343283582

Area Under Curve score: 0.4923475841133316

Figure 20. ผลการทดลอง Random Forest จากดัชนีหุ้น Heng Seng Index.

- XGBoost

```
xgbEval = evaluate(cla, y_test, y_pred)
printEval(cla, xgbEval)
```

[115] ✓ 0.3s

... Accuracy Score: 0.5119047619047619

Confusion Matrix:

```
[[64 54]
 [69 65]]
```

F1 Score: 0.5138339920948617

Precision Score: 0.5462184873949579

Recall score: 0.48507462686567165

Area Under Curve score: 0.513723754110802

Figure 21. ผลการทดลอง XGBoost จากดัชนีหุ้น Heng Seng Index.

- Logistic Regression

```

logisticEval = evaluate(cia, y_test, y_pred)
printEval(cia, logisticEval)

[111] ✓ 0.3s

... Accuracy Score: 0.44841269841269843
Confusion Matrix:
[[45 73]
 [66 68]]
F1 Score: 0.49454545454545454
Precision Score: 0.48226950354609927
Recall score: 0.5074626865671642
Area Under Curve score: 0.444409309385277

```

Figure 22. ผลการทดลอง Logistic Regression จากดัชนีหุ้น Heng Seng Index.

## Regression

- Linear Regression



Figure 23. ผลการทดลอง Linear Regression จากดัชนีหุ้น Heng Seng Index.

- Polynomial Regression



Figure 24. ผลการทดลอง Polynomial Regression จากดัชนีหุ้น Heng Seng Index.

## 4. SSE Composite Index .

Classification

- Decision Tree

```

▷ v
    dctEval = evaluate(cla, y_test, y_pred)
    printEval(cla, dctEval)
[142] ✓ 0.3s
... Accuracy Score: 0.4717741935483871
    Confusion Matrix:
        [[51 65]
         [66 66]]
    F1 Score: 0.5019011406844106
    Precision Score: 0.5038167938931297
    Recall score: 0.5
    Area Under Curve score: 0.4698275862068966

```

Figure 25. ผลการทดลอง Decision Tree จากดัชนีหุ่น SSE Composite Index.

- Random Forest

```

▷ v
    forestEval = evaluate(cla, y_test, y_pred)
    printEval(cla, forestEval)
[145] ✓ 0.3s
... Accuracy Score: 0.4959677419354839
    Confusion Matrix:
        [[47 69]
         [56 76]]
    F1 Score: 0.5487364620938628
    Precision Score: 0.5241379310344828
    Recall score: 0.5757575757575758
    Area Under Curve score: 0.49046499477533956

```

Figure 26. ผลการทดลอง Random Forest จากดัชนีหุ่น SSE Composite Index.

- XGBoost

```

▷ v
    xgbEval = evaluate(cla, y_test, y_pred)
    printEval(cla, xgbEval)
[153] ✓ 0.3s
... Accuracy Score: 0.4959677419354839
    Confusion Matrix:
        [[50 66]
         [59 73]]
    F1 Score: 0.5387453874538745
    Precision Score: 0.5251798561151079
    Recall score: 0.553030303030303
    Area Under Curve score: 0.4920323928944619

```

Figure 27. ผลการทดลอง XGBoost จากดัชนีหุ่น SSE Composite Index.



- Logistic Regression

```

logisticEval = evaluate(c1a, y_test, y_pred)
printEval(c1a, logisticEval)
[149] ✓ 0.3s
... Accuracy Score: 0.5161290322580645
Confusion Matrix:
[[39 77]
 [43 89]]
F1 Score: 0.5973154362416107
Precision Score: 0.536144578313253
Recall score: 0.6742424242424242
Area Under Curve score: 0.5052246603970743

```

Figure 28. ผลการทดลอง Logistic Regression จากดัชนีหุ้น SSE Composite Index.

## Regression

- Linear Regression

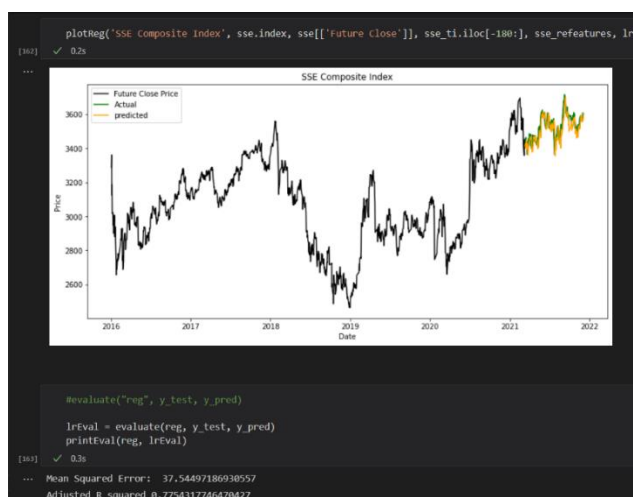


Figure 29. ผลการทดลอง Linear Regression จากดัชนีหุ้น SSE Composite Index.

- Polynomial Regression



Figure 30. ผลการทดลอง Polynomial Regression จากดัชนีหุ้น SSE Composite Index.

## อภิปรายผลการวิจัย

การศึกษานี้มีวัตถุประสงค์หลักเพื่อทำนายราคาดัชนีหุ้นและคำนวณผลตอบแทนจากการลงทุนด้วยกลยุทธ์การลงทุนถัวเฉลี่ยต้นทุนอย่างต่อเนื่องเป็นเวลา 60 เดือน ตั้งแต่เดือนมกราคม 2016 ถึง ธันวาคม 2021 โดยทำการดึงข้อมูลดัชนีมาจากรีเว็บไซต์ Yahoo finance หลังจากนั้นพยากรณ์ราคาหุ้นรายตัว ได้แก่ Dow Jones Industrial Average, Nikkei 225 Stock Average, Heng Seng Index และ SSE Composite Index ด้วยเทคนิค Classification และ Regression แล้วจึงแล้วคำนวณเป็นอัตราผลตอบแทนของดัชนีหุ้น สำหรับ Classification จะนำเอาราคาปิดของวันปัจจุบันและวันต่อไปมาหาเปอร์เซ็นต์ความแตกต่างว่าหุ้นขึ้นหรือลง เพื่อใช้เป็น Tagget ส่วน Regression จะนำเอาราคาปิดของวันถัดไป 1 วันมาใช้เป็น Target หลังจากนั้นใช้ Feature Selection แบบ Recursive Feature Elimination หรือ RFE มาวิเคราะห์หา Feature สำหรับทั้ง Classification และ Regression โดยใช้ Target (สำหรับ Classification) และ Re Target (สำหรับ Regression) เป็นตัว Target ในการวิเคราะห์ข้อมูล โดย RFE จะเลือก Feature ที่ควรจะใช้มาเอง ไม่มีการจำกัดว่าต้องมีกี่ตัวแต่อย่างใด เมื่อทำการเลือก Feature ได้แล้วจะทำการสร้างโมเดลในการพยากรณ์ข้อมูล ซึ่งจะมี 6 วิธี (1) การพยากรณ์ด้วยวิธี Decision Tree อยู่ในกลุ่มของ Classification (2) การพยากรณ์ด้วยวิธี Logistic Regression อยู่ในกลุ่มของ Classification (3) การพยากรณ์ด้วยวิธี Random Forest อยู่ในกลุ่มของ Classification (4) การพยากรณ์ด้วยวิธี XGBoost อยู่ในกลุ่มของ Classification (5) การพยากรณ์ด้วยวิธี Linear Regression อยู่ในกลุ่มของ Regression และ (6) การพยากรณ์ด้วยวิธี Polynomial Regression อยู่ในกลุ่มของ Regression โดยผลการศึกษาพบว่าเทคนิคที่เหมาะสมต่อการพยากรณ์มากที่สุด คือ เทคนิคของ Classification เนื่องจากมีการประเมินความแม่นยำที่ สะดวกและดีที่สุด โดยผลจากการศึกษาและคาดการณ์พบว่า ณ วันที่ 6 ธันวาคม 2021 ราคา Dow Jones Industrial Average และราคา Heng Seng Index มีแนวโน้มที่จะปรับตัวลดลง ส่วนราคา Nikkei 225 Stock Average และราคา SSE Composite Index มีแนวโน้มการปรับตัวขึ้น ซึ่งพบว่าดัชนีหุ้นที่ควรลงทุนในวันนี้คือ Dow Jones Industrial Average และ Heng Seng Index ส่วนดัชนีหุ้นที่ควรขายในวันนี้คือ Nikkei 225 Stock Average และ SSE Composite Index ณ วันนั้น ถ้าหากเกิดเหตุการณ์ที่การคาดการณ์ของเทคนิคทั้งหมดเท่ากันไม่สามารถบอกได้ว่าราคาจะขึ้นหรือลงทางระบบจะทำการประเมินผลความแม่นยำ ซึ่งเทคนิคที่มีความแม่นยำน้อยที่สุดจะถูกตัดออกและต่อมาจะทำการคาดการณ์ครั้งใหม่เพื่อให้ได้คำตอบว่า ดัชนีหุ้นทั้ง 4 ตัวนี้ควรลงทุนหรือไม่ในวันนี้

## สรุปผลการวิจัย

การศึกษานี้ได้ใช้เทคนิค Classification และ Regression เพื่อพยากรณ์ราคาดัชนีหุ้นและคำนวณความเหมาะสมในการลงทุน โดยมีรายละเอียดดังนี้การพยากรณ์ด้วยเทคนิค Classification 4 เทคนิค ได้แก่ Decision Tree, Random Forest, Logistic Regression และ XGBoost และเทคนิค Regression 2 เทคนิค ได้แก่ Linear Regression และ Polynomial Regression โดยใช้ข้อมูลตั้งแต่เดือนมกราคม 2016 ถึง ธันวาคม 2021 ของดัชนีหุ้นจำนวน 4 ตัว ได้แก่ (1) Dow Jones Industrial Average หรือ DJI (2) Nikkei

225 Stock Average หรือ N225 (3) Heng Seng Index หรือ HSI และ (4) SSE Composite Index หรือ SSE จากเว็บไซต์ของ Yahoo finance โดยศึกษาและคาดการณ์ราคาดัชนีหุ้น ได้ผลดังนี้

```
finalPredict(dji_target, dji_features,dji_refeatures, dctEval, forestEval, logisticEval, xgbEval, lrEval, prEval)
[53] ✓ 0.9s
... [0, 0, 1, 1, 0, 0]
The model suggests that you should buy
```

Figure 31. Final Predict Dow Jones Industrial Average

```
finalPredict(dji_target, dji_features,dji_refeatures, dctEval, forestEval, logisticEval, xgbEval, lrEval, prEval)
[92] ✓ 0.9s
... [1, 1, 1, 1, 1, 1]
The model suggests that you should sell
```

Figure 32. Final Predict Nikkei 225 Stock Average

```
finalPredict(dji_target, dji_features,dji_refeatures, dctEval, forestEval, logisticEval, xgbEval, lrEval, prEval)
[130] ✓ 0.9s
... [1, 1, 0, 0, 0, 1]
[1, 0, 0, 0, 0, 1]
The model suggests that you should buy
```

Figure 33. Final Predict Heng Seng Index

```
finalPredict(dji_target, dji_features,dji_refeatures, dctEval, forestEval, logisticEval, xgbEval, lrEval, prEval)
[168] ✓ 0.9s
... [0, 0, 1, 1, 0, 1]
[0, 1, 1, 0, 0, 1]
The model suggests that you should sell
```

Figure 34. Final Predict SSE Composite Index

จากการศึกษาและคาดการณ์พบว่า ณ วันที่ 6 ธันวาคม 2021 ราคา Dow Jones Industrial Average และราคา Heng Seng Index มีแนวโน้มที่จะปรับตัวลดลง ส่วนราคา Nikkei 225 Stock Average และราคา SSE Composite Index มีแนวโน้มการปรับตัวขึ้น ซึ่งพบว่าดัชนีหุ้นที่ควรลงทุนในวันนี้คือ Dow Jones Industrial Average และ Heng Seng Index ส่วนดัชนีหุ้นที่ควรขายในวันนี้คือ Nikkei 225 Stock Average และ SSE Composite Index โดยมีเงื่อนไขว่า หากการคาดการณ์ของเทคนิคทั้งหมดเท่ากันไม่สามารถบอกได้ว่าราคาจะขึ้นหรือลง ทางระบบจะทำการประเมินผลความแม่นยำ ซึ่งเทคนิคที่มีความแม่นยำน้อยที่สุดจะถูกตัดออกและต่อมาจะทำการคาดการณ์ครั้งใหม่เพื่อให้ได้คำตอบว่า ดัชนีหุ้นทั้ง 4 ตัวนี้ควรลงทุนหรือไม่ในวันนี้

### ข้อเสนอแนะ

ข้อเสนอแนะสำหรับการศึกษาค้างต่อไป คือ ควรมีการเปรียบเทียบเทคนิคพยากรณ์ด้วยวิธีการอื่น ๆ เช่น AdaBoost regression เป็นต้น เพื่อให้สามารถเปรียบเทียบผลตอบแทนจากการลงทุนในหุ้นอย่างเหมาะสมและมีความแม่นยำสูง

## เอกสารอ้างอิง

- Bex T. **Powerful Feature Selection with Recursive Feature Elimination (RFE) of Sklearn.** [Online]. 2018. Available from: <https://towardsdatascience.com/powerful-feature-selection-with-recursive-feature-elimination-rfe-of-sklearn-23efb2cdb54e> [27 November 2021]
- Scikit learn. **Sklearn.ensemble.RandomForestClassifier.** [Online]. 2017. Available from: <https://scikitlearn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html> [28 November 2021]
- Prashanth Saravanan. **Understanding Loss Functions in Machine Learning.** [Online]. 2021. Available from : <https://www.section.io/engineering-education/understanding-loss-functions-in-machine-learning/#loss-functions-for-regression> [28 November 2021]
- Avinash Navlani. **Understanding Logistic Regression in Python.** [Online]. 2019. Available from : <https://www.datacamp.com/community/tutorials/understanding-logistic-regression-python> [30 November 2021]
- Saishruthi Swaminathan. **Logistic Regression — Detailed Overview.** [Online]. 2018. Available from : <https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc> [3 December 2021]
- Witchapong Daroontham. **รู้จัก Decision Tree, Random Forest, และ XGBoost!!! — PART 1.** [ออนไลน์]. 2018. แหล่งที่มา : <https://medium.com/@witchapongdaroontham/รู้จัก-decision-tree-random-forrest-และ-xgboost-part-1-cb49c4ac1315> [30 พฤศจิกายน 2021]
- Scikit learn. **Metrics and scoring: quantifying the quality of predictions.** [Online]. 2017. Available from : [https://scikit-learn.org/stable/modules/model\\_evaluation.html](https://scikit-learn.org/stable/modules/model_evaluation.html) [1 December 2021]
- Boom626. **Confusion Matrix.** [ออนไลน์]. 2019. แหล่งที่มา : [https://medium.com/@mirthful\\_sunset\\_cattle\\_231/confusion-matrix-48cc396b1b58](https://medium.com/@mirthful_sunset_cattle_231/confusion-matrix-48cc396b1b58) [3 ธันวาคม 2021]
- ICHI PRO. **การเลือกคุณสมบัติสำหรับ Machine Learning ใน Python - Wrapper Methods.** [Online]. 2017. Available from : <https://ichi.pro/th/kar-leuxk-khunsmbati-sahrab-machine-learning-ni-python-wrapper-methods-47683395050289> [December 2021]