ทำนายการเสียชีวิตของผู้ติดเชื้อโควิดในประเทศเกาหลีใต้ด้วยเทคนิค Data mining

นำเสนอโดย

นางสาวอนุศรา คำงาม รหัส 645020062-0

วัตถุประสงค์

- 1. เพื่อศึกษาการเสียชีวิตของผู้ติดเชื้อโควิด 19 ในประเทศเกาหลีใต้
- 2. เพื่อทำนายการเสียชีวิตของผู้ติดเชื้อโควิดในประเทศเกาหลีใต้ ด้วย วิธี Classification

เทคนิคที่ใช้

- Classification
  - Decision Tree
  - K Nearest Neighbor (KNN)

ประโยชน์ที่คาดว่าจะได้รับ

- ได้สารสนเทศเกี่ยวกับผู้ที่เสียชีวิตจากการติดเชื้อโควิด 19 ในประเทศเกาหลีใต้
- นำผลการศึกษาไปปรับใช้ในวางแผนงานด้านสาธารณสุข ในการรักษาพยาบาลผู้ป่วยที่ติดเชื้อโควิด 19

ข้อมูลการติดเชื้อโควิด 19 ในประเทศเกาหลีใต้ ปี 2020 ระหว่าง เดือน กุมภาพันธ์ - พฤษภาคม

	patient_id	sex	age	country	province	city	infection_case	infected_by	contact_number	symptom_onset_date	confirmed_date	released_date	deceased_date	state
0	1000000001	male	50s	Korea	Seoul	Gangseo-gu	overseas inflow	NaN	75	2020-01-22	2020-01-23	2020-02-05	NaN	released
1	1000000002	male	30s	Korea	Seoul	Jungnang-gu	overseas inflow	NaN	31	NaN	2020-01-30	2020-03-02	NaN	released
2	1000000003	male	50s	Korea	Seoul	Jongno-gu	contact with patient	2002000001	17	NaN	2020-01-30	2020-02-19	NaN	released
3	1000000004	male	20s	Korea	Seoul	Mapo-gu	overseas inflow	NaN	9	2020-01-26	2020-01-30	2020-02-15	NaN	released
4	1000000005	female	20s	Korea	Seoul	Seongbuk-gu	contact with patient	1000000002	2	NaN	2020-01-31	2020-02-24	NaN	released
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
5160	7000000015	female	30s	Korea	Jeju-do	Jeju-do	overseas inflow	NaN	25	NaN	2020-05-30	2020-06-13	NaN	released
5161	7000000016	NaN	NaN	Korea	Jeju-do	Jeju-do	overseas inflow	NaN	NaN	NaN	2020-06-16	2020-06-24	NaN	released
5162	7000000017	NaN	NaN	Bangladesh	Jeju-do	Jeju-do	overseas inflow	NaN	72	NaN	2020-06-18	NaN	NaN	isolated
5163	7000000018	NaN	NaN	Bangladesh	Jeju-do	Jeju-do	overseas inflow	NaN	NaN	NaN	2020-06-18	NaN	NaN	isolated
5164	7000000019	NaN	NaN	Bangladesh	Jeju-do	Jeju-do	overseas inflow	NaN	NaN	NaN	2020-06-18	NaN	NaN	isolated

5165 rows × 14 columns

วิธีดำเนินการ

- ขั้นตอนที่ 1 เตรียม ข้อมูล

นำข้อมูลทำการตรวจหาค่า Missing ของข้อมูล

ตรวจสอบค่า mising จากตาราง PatientInfo

```
In [34]: Patient.isnull().any()

Out[34]: patient_id           False
         sex                   True
         age                   True
         country              False
         province             False
         city                  True
         infection_case        True
         infected_by           True
         contact_number        True
         symptom_onset_date    True
         confirmed_date        True
         released_date         True
         deceased_date         True
         state                False
         dtype: bool
```

จัดการค่า **Missing** ด้วยค่า **Mode** เนื่องจากเป็นตัวแปรเชิงคุณภาพ

ขั้นตอนที่ 2 ทำการ Visualization

จำนวนผู้เสียชีวิตของผู้ติดเชื้อโควิด 19 ในประเทศเกาหลีใต้ จำแนกดังนี้

1.กลุ่มอายุ

2.เพศ

3.สาเหตุการติดเชื้อ

4.เมือง

ข้อมูลที่ใช้ทำ Visualization

	patient_id	sex	age	city	infection_case	deceased_date
0	1000000001	male	50s	Gangseo-gu	overseas inflow	NaN
1	1000000002	male	30s	Jungnang-gu	overseas inflow	NaN
2	1000000003	male	50s	Jongno-gu	contact with patient	NaN
3	1000000004	male	20s	Mapo-gu	overseas inflow	NaN
4	1000000005	female	20s	Seongbuk-gu	contact with patient	NaN
...	...	...	...	...	...	...
5160	7000000015	female	30s	Jeju-do	overseas inflow	NaN
5161	7000000016	female	20s	Jeju-do	overseas inflow	NaN
5162	7000000017	female	20s	Jeju-do	overseas inflow	NaN
5163	7000000018	female	20s	Jeju-do	overseas inflow	NaN
5164	7000000019	female	20s	Jeju-do	overseas inflow	NaN

5165 rows × 6 columns

1.เพศ

2.อายุ

3.เมืองที่อาศัย

4.สาเหตุของการติดเชื้อ

5.สถานะของการเสียชีวิต

ตารางแสดงจำนวนผู้ที่ติดเชื้อแยกตามสถานการณ์ โดยมีผู้ป่วยติดเชื้อโควิด 19 ทั้งหมด 5165 คน

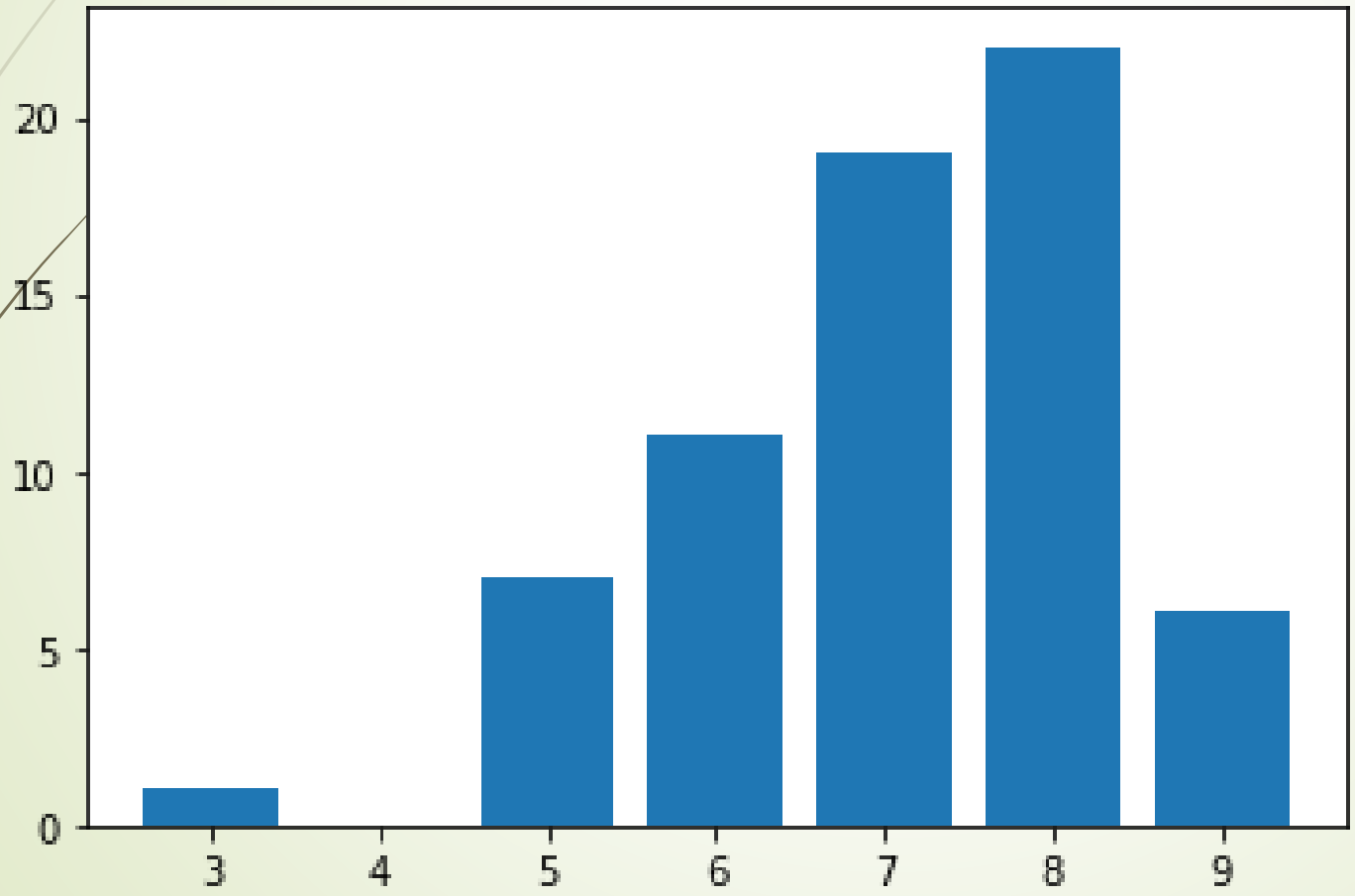
	patient_id	sex	age	city	infection_case
deceased_date					
0.0	5099	5099	5099	5099	5099
1.0	66	66	66	66	66

- **เสียชีวิตทั้งหมด 66 คน**
- **ไม่เสียชีวิต ทั้งหมด 5099 คน**
- โดยกำหนด

  0 คือ จำนวนผู้ติดเชื้อโควิดที่ไม่เสียชีวิด มีจำนวน 5099 คน

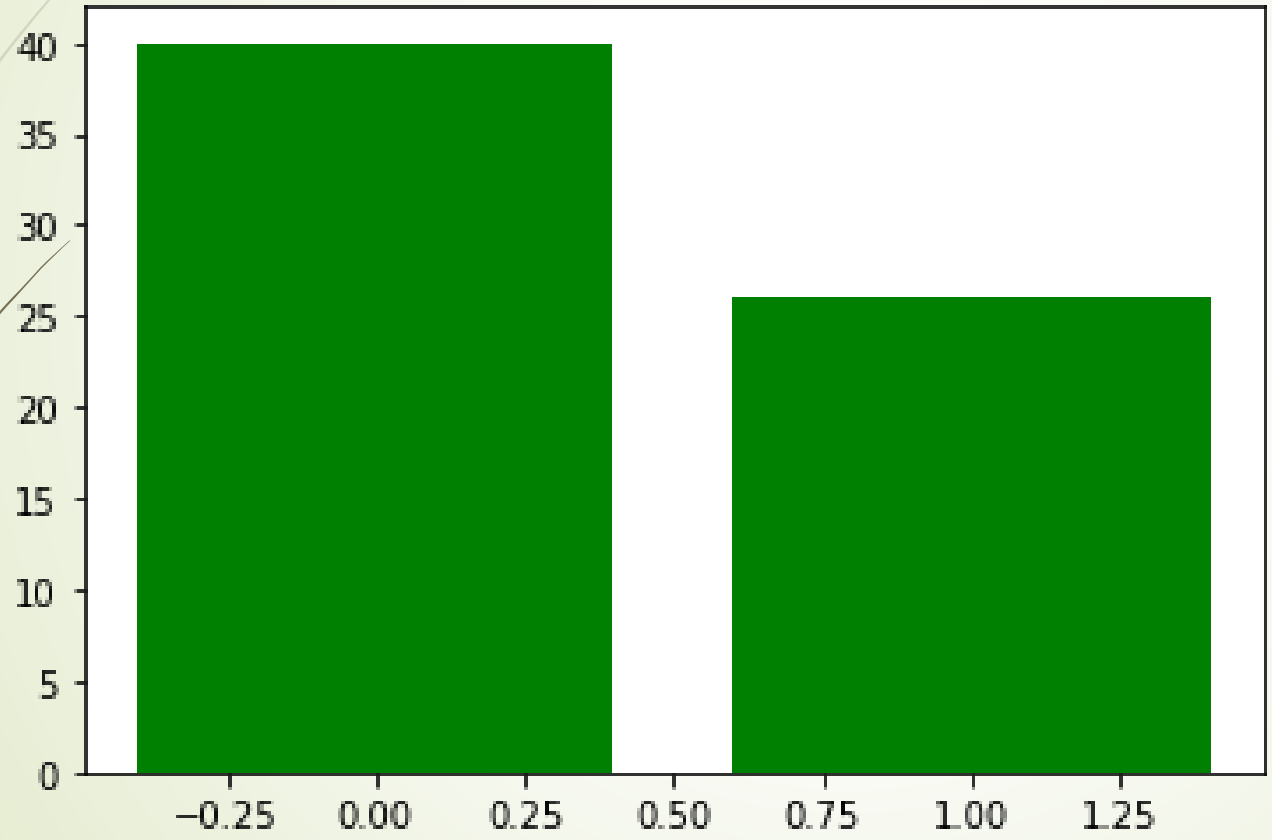
  1 คือ จำนวนผู้ติดเชื้อคิดที่เสียชีวิต มีจำนวน 66 คน

จำนวนผู้ติดเชื้อโควิด 19 ที่เสียชีวิตจำแนกตามกลุ่มอายุ



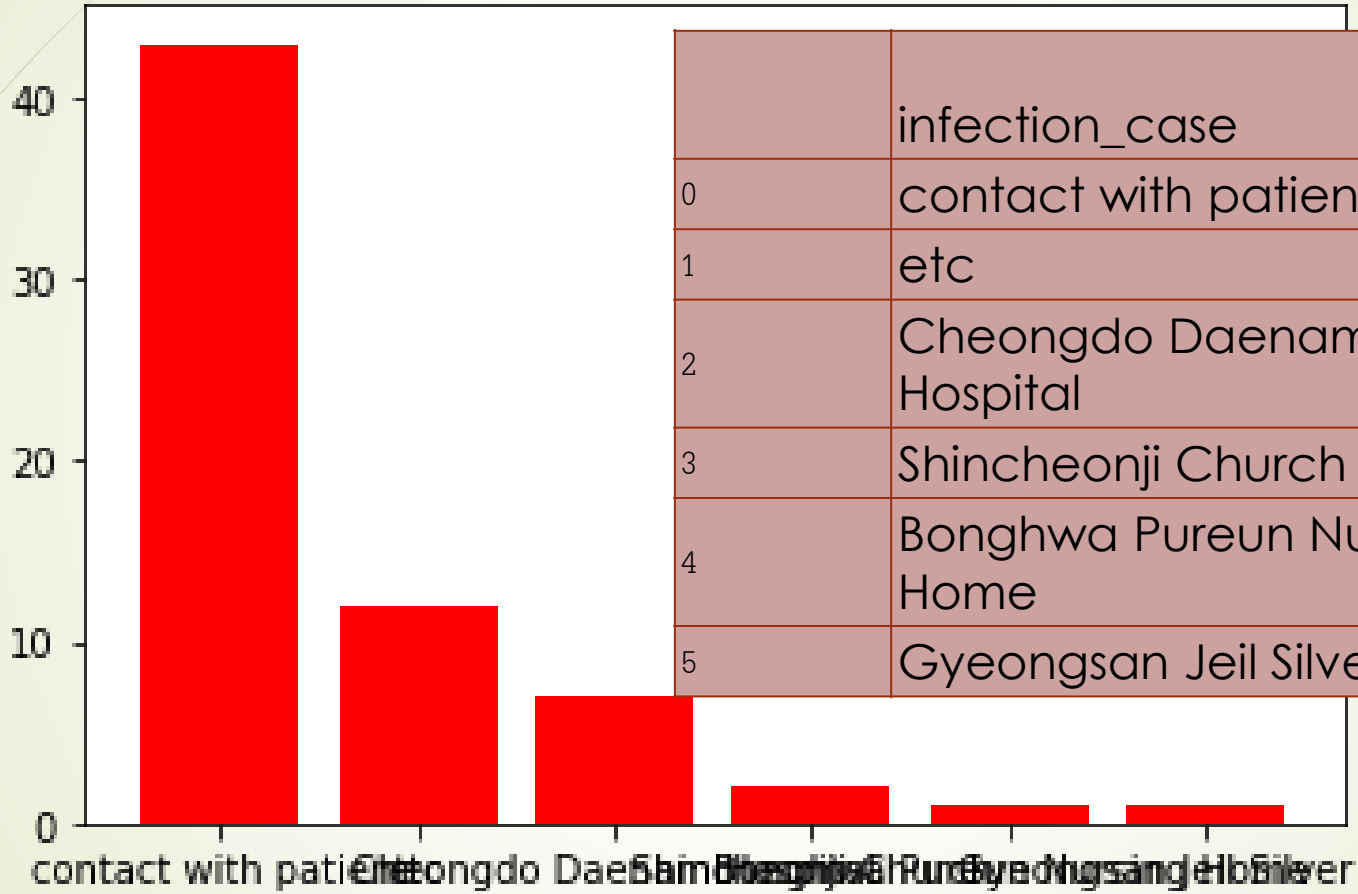
ผู้ติดเชื้อโควิดที่มี
อายุ 80-89 ปี
มีจำนวนผู้เสียชีวิตมากที่สุด

จำนวนผู้ติดเชื้อโควิด 19 ที่เสียชีวิตจำแนกตามเพศ



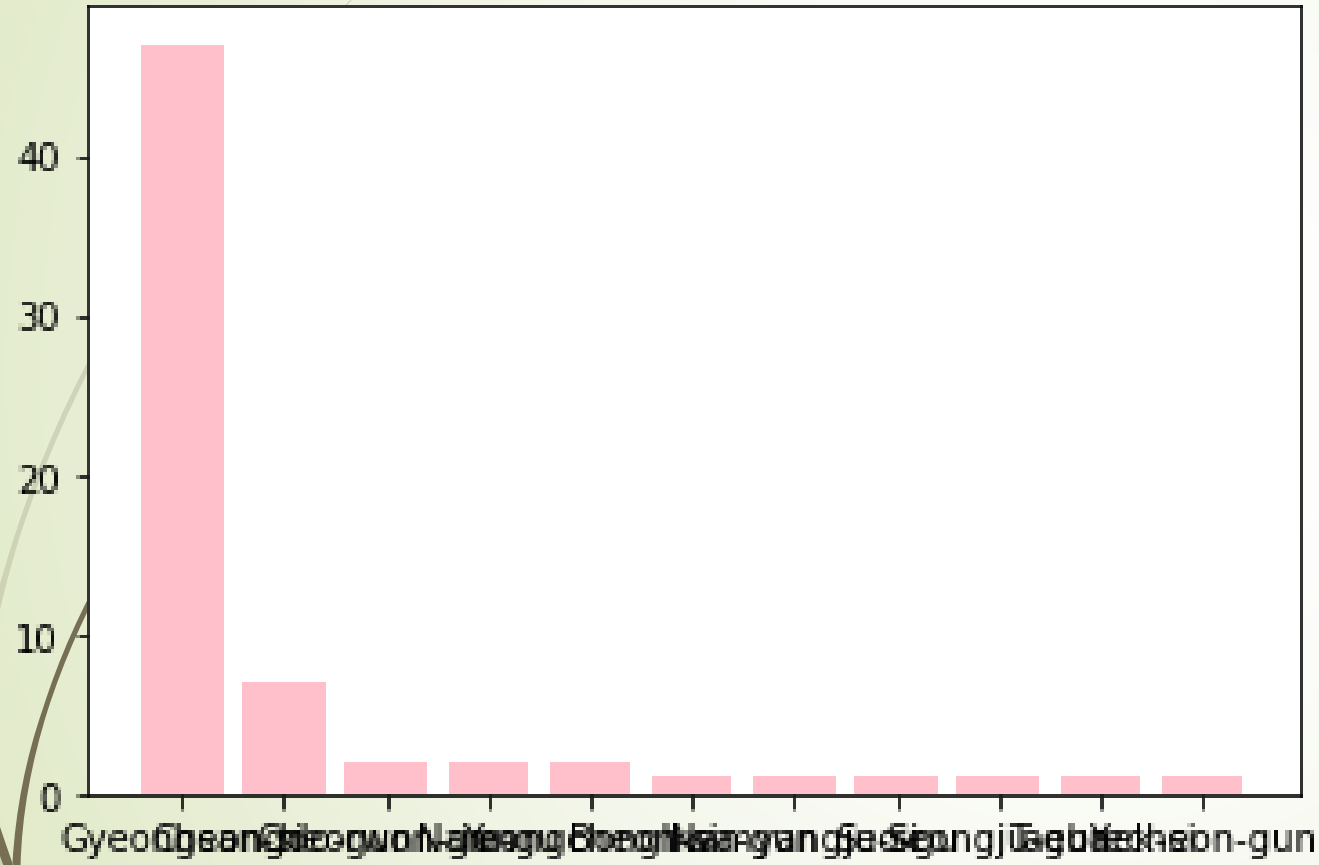
ผู้ติดเชื้อโควิด เพศชาย มีผู้เสียชีวิตมากกว่าเพศหญิง

จำนวนผู้ติดเชื้อโควิด เสียชีวิตจำแนกตามสาเหตุการติดเชื้อ



	infection_case	deceased_date
0	contact with patient	43
1	etc	12
2	Cheongdo Daenam Hospital	7
3	Shincheonji Church	2
4	Bonghwa Pureun Nursing Home	1
5	Gyeongsan Jeil Silver Town	1

ผู้ติดเชื้อโควิดที่เสียชีวิตส่วนใหญ่มีสาเหตุจากการสัมผัสหรือติดต่อกับผู้ติดเชื้อ



ผู้ติดเชื้อโควิดส่วนใหญ่อยู่ที่เมือง

Gyeongsan-si

	city	deceased_date
0	Gyeongsan-si	47
1	Cheongdo-gun	7
2	Cheorwon-gun	2
3	Nam-gu	2
4	Yeongcheon-si	2
5	Bonghwa-gun	1
6	Namyangju-si	1
7	Seo-gu	1
8	Seongju-gun	1
9	Taebaek-si	1
10	Yecheon-gun	1

Data mining

- **Classification**

-Deecision Tree

-K Nearest Neighbor (KNN)

กำหนดการทำนาย

0 คือ ไม่เสียชีวิต

1 คือ เสียชีวิต

```
Patient_Total = pd.merge(Patient_Casedata,city_id,on='city')# เชื่อมตาราง  ด้วย .merge
Patient_Total
```

	patient_id	sex	age	city	infection_case	deceased_date	infection_case_id	city_id
0	1000000001	0	5	Gangseo-gu	overseas inflow	0.0	51	40
1	1000000027	0	5	Gangseo-gu	overseas inflow	0.0	51	40
2	1000000317	1	3	Gangseo-gu	overseas inflow	0.0	51	40
3	1000000327	1	2	Gangseo-gu	overseas inflow	0.0	51	40
4	1000000335	0	3	Gangseo-gu	overseas inflow	0.0	51	40
...	...	...	...	...	...	...	...	...
5160	6014000005	0	6	Yeongju-si	etc	0.0	48	153
5161	6100000089	0	6	Haman-gun	etc	0.0	48	74
5162	6100000104	1	7	Sancheong-gun	etc	0.0	48	117
5163	6100000013	1	7	Goseong-gun	Shincheonji Church	0.0	39	54
5164	6100000063	1	2	Goseong-gun	Shincheonji Church	0.0	39	54

5165 rows × 8 columns

Decision Tree

```
In [101]: from sklearn.model_selection import cross_val_score
          model1 = DecisionTreeClassifier(criterion='entropy',min_samples_leaf=4)

          csv = cross_val_score(model1,X,Y, cv=3) #แบ่งข้อมูลเป็น 3 ส่วน
          print(csv.round(3)) #ทศนิยม 3 ตำแหน่ง
          csv.mean().round(3) # ค่าเฉลี่ย

          [0.987 0.888 0.986]
Out[101]: 0.954

In [103]: model2 = DecisionTreeClassifier(criterion='entropy',max_leaf_nodes=5)

          csv = cross_val_score(model2,X,Y, cv=3) #แบ่งข้อมูลเป็น 3 ส่วน
          print(csv.round(3)) #ทศนิยม 3 ตำแหน่ง
          csv.mean().round(3) # ค่าเฉลี่ย

          [0.987 0.987 0.987]
Out[103]: 0.987

In [104]: model3 = DecisionTreeClassifier(criterion='entropy',max_depth=7)

          csv = cross_val_score(model3,X,Y, cv=3) #แบ่งข้อมูลเป็น 3 ส่วน
          print(csv.round(3)) #ทศนิยม 3 ตำแหน่ง
          csv.mean().round(3) # ค่าเฉลี่ย

          [0.987 0.882 0.987]
Out[104]: 0.952
```

แบ่งข้อมูล โดย Crosvalidation แบ่งข้อมูลออกเป็น 3 ส่วน นำไปใช้กับทุก Model ทั้งหมดจำนน 3 Model

KNN

KNN

In [106]:
```
model4 =  KNeighborsClassifier(n_neighbors=1)

csv = cross_val_score(model4,X,Y, cv=3) #แบ่งข้อมูลเป็น 3 ส่วน
print(csv.round(3)) #ทศนิยม 3 ตำแหน่ง
csv.mean().round(3) # ค่าเฉลี่ย
```

```
[0.99  0.894 0.986]
```

Out[106]:
```
0.956
```

In [107]:
```
model5 =  KNeighborsClassifier(n_neighbors=11,weights='distance') #เชื่อทุกคนเท่ากัน

csv = cross_val_score(model5,X,Y, cv=3) #แบ่งข้อมูลเป็น 3 ส่วน
print(csv.round(3)) #ทศนิยม 3 ตำแหน่ง
csv.mean().round(3) # ค่าเฉลี่ย
```

```
[0.988 0.934 0.986]
```

Out[107]:
```
0.969
```

In [108]:
```
model6 =  KNeighborsClassifier(n_neighbors=5,weights='distance') #เชื่อคนใกล้มากว่าคนไกล

csv = cross_val_score(model6,X,Y, cv=3) #แบ่งข้อมูลเป็น 3 ส่วน
print(csv.round(3)) #ทศนิยม 3 ตำแหน่ง
csv.mean().round(3) # ค่าเฉลี่ย
```

```
[0.987 0.923 0.985]
```

Out[108]:
```
0.965
```

```
model2 = DecisionTreeClassifier(criterion='entropy',max_leaf_nodes=5)

csv = cross_val_score(model2,X,Y, cv=3) #แบ่งข้อมูลเป็น 3 ส่วน
print(csv.round(3)) #ทศนิยม 3 ตำแหน่ง
csv.mean().round(3) # ค่าเฉลี่ย
```

```
[0.987 0.987 0.987]
0.987
```

เลือก Model 2 เพื่อทำการ Train เนื่องจากให้ค่าความถูกต้องมากที่สุด 98.7 %

วัดผล

```
In [117]: from sklearn.metrics import accuracy_score
          final_resule = model2Full.predict(X_test)
          accuracy_score(y_test,final_resule)

Out[117]: 0.9856755710414247
```

วัดค่าความถูกต้องได้ 98.56%

สรุปผล

Evaluation

In [118]:
```
from sklearn.metrics import classification_report, confusion_matrix
```

In [119]:
```
cm1 = confusion_matrix(y_test,model2Full.predict(X_test))
cm1
```

Out[119]:
```
array([[2546,    0],
       [  37,    0]])
```

In [120]:
```
cr1 = classification_report(y_test,model2Full.predict(X_test))
print(cr1)
```

```
              precision    recall  f1-score   support

         0.0       0.99      1.00      0.99      2546
         1.0       0.00      0.00      0.00        37

    accuracy                           0.99      2583
   macro avg       0.49      0.50      0.50      2583
weighted avg       0.97      0.99      0.98      2583
```

Model ทำนายถูก 99% โดยทำนายว่าผู้ติดเชื้อที่ไม่เสียชีวิต เป็น 100% เนื่องจากข้อมูลใน Class มีจำนวนที่ต่างกันมาก โปรแกรมจึงทำนายเฉพาะค่าผู้ที่ไม่เสียชีวิต

```
/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1272: UndefinedMetricWarning: Precision and F-score are ill-def
ined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

In [ ]:

การแก้ไข

1. แบ่ง Class ของข้อมูลออกเป็น 2 ชุด คือ

   - Class 0 คือ ข้อมูลของผู้ติดเชื้อที่ไม่เสียชีวิต จำนวน 3569 คน

   - Class 1 ข้อมูลของผู้ติดเชื้อที่เสียชีวิต จำนวน 46 คน

2. แบ่งข้อมูลเพื่อใช้เป็นข้อมูล Train : Test ทั้ง 2 Class ดังนี้

   - ข้อมูล Train 70%

   - ข้อมูล Test 30%

3.นำข้อมูล Class 1 คือ ผู้ติดเชื้อที่เสียชีวิต มา Random ให้เท่ากับข้อมูล Train และ Test ใน Class 0

4. รวมข้อมูลให้เป็นตารางเดียวกันและนำไปทำ Classification

จบการนำเสนอ