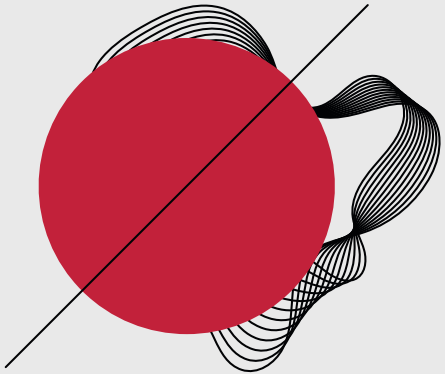
Transitioning into an endemic phase in Singapore



Group A+PLZ!
ZHU BANGYUAN - Logistic Regression Model
LI JINGYUAN - Data Preparation
YANG PEISHI - Background & Exploration
YU LINYAN - Visualisation

TABLE OF CONTENTS

01.

INTRODUCTION

Background

- Record high daily covid cases
- Strain on limited healthcare system
- Rising number of deaths
- Urgent and prolonged threat to Singapore’s health and well-being

Problem

How will Singapore’s Covid-19 cases trend in the near future?

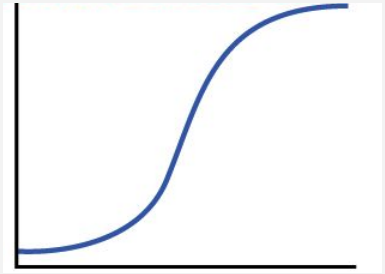
Solution

Daily Confirmed Cases

New Recoveries

Daily Deaths

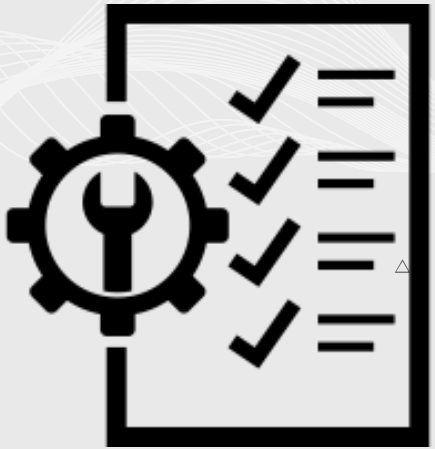
Logistic Regression Model



Predict number of Covid-19 cases in the near future

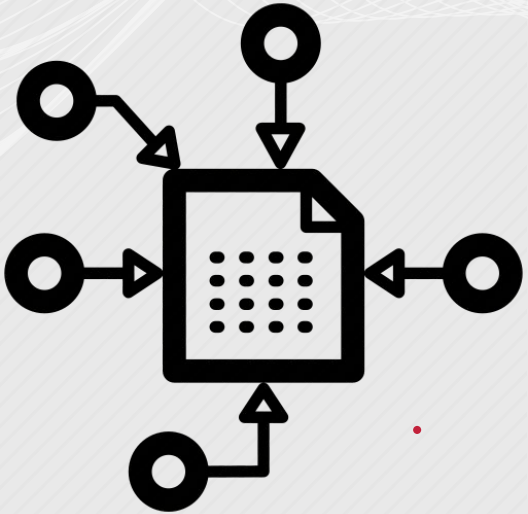
Objective

Aid in the planning and development of measures during the Covid-19 endemic



Data Set

Data Set used: Covid-19 Singapore Statistics
Data Source: data.world



02.

PROGRAMME

To begin the analysis, we extracted our data set from https://data.world/hxchua/covid-19-singapore/workspace/file?filename=Covid-19+SG.xlsx and imported the data into Excel as Covid-19 SG.csv.

Firstly, a read_csv command is performed in Jupyter to breifly look at the dataset

```
In [2]: covidData = pd.read_csv('Covid-19 SG.csv')
        covidData.head()
```

Out[2]:

	Date	DailyConfirmed	NewRecoveries	False Positives Found	Daily Discharged	Passed but not due to COVID	Cumulative Discharged	Discharged to Isolation	Still Hospitalised	DailyDeaths	...	Phase	Cumulative Vaccine Doses	Cumulative Individuals Vaccinated
0	2020-01-23	1	0	NaN	0	0	0	0	1	0	...	NaN	NaN	NaN
1	2020-01-24	2	0	NaN	0	0	0	0	3	0	...	NaN	NaN	NaN
2	2020-01-25	1	0	NaN	0	0	0	0	4	0	...	NaN	NaN	NaN
3	2020-01-26	0	0	NaN	0	0	0	0	4	0	...	NaN	NaN	NaN
4	2020-01-27	1	0	NaN	0	0	0	0	5	0	...	NaN	NaN	NaN

BASIC INFORMATION

```
covidData.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 647 entries, 0 to 646
Data columns (total 33 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   Date                                    647 non-null    object
 1   DailyConfirmed                          647 non-null    int64
 2   NewRecoveries                           647 non-null    int64
 3   False Positives Found                   538 non-null    float64
 4   Daily Discharged                        647 non-null    int64
 5   Passed but not due to COVID             647 non-null    int64
 6   Cumulative Discharged                   647 non-null    int64
 7   Discharged to Isolation                 647 non-null    int64
 8   Still Hospitalised                      647 non-null    int64
 9   DailyDeaths                             647 non-null    int64
 10  Tested positive demise                  647 non-null    int64
 11  Daily Imported                          647 non-null    int64
 12  Daily Local transmission                647 non-null    int64
 13  Local cases residing in dorms MOH report 581 non-null   float64
```

```
covidData.shape
```

```
(647, 33)
```

```
covidData.dtypes
```

```
Date                                      object
DailyConfirmed                             int64
NewRecoveries                              int64
False Positives Found                    float64
Daily Discharged                           int64
Passed but not due to COVID                int64
Cumulative Discharged                      int64
Discharged to Isolation                    int64
Still Hospitalised                         int64
DailyDeaths                                int64
Tested positive demise                     int64
Daily Imported                             int64
Daily Local transmission                   int64
Local cases residing in dorms MOH report  float64
Local cases not residing in doms MOH report float64
Intensive Care Unit (ICU)                  int64
General Wards MOH report                  float64
In Isolation MOH report                   float64
Total Completed Isolation MOH report      float64
```

```
covidData.describe()
```

	DailyConfirmed	NewRecoveries	False Positives Found	Daily Discharged	Passed but not due to COVID	Cumulative Discharged	Discharged to Isolation	Still Hospitalised	DailyDeaths
count	647.000000	647.000000	538.000000	647.000000	647.000000	647.000000	647.000000	647.000000	647.000000
mean	301.840804	251.698609	0.148699	251.735703	0.009274	49004.109737	3408.401855	310.046368	0.610510
std	737.282571	631.793962	2.323110	631.760759	0.095926	28362.766902	6545.023173	527.849517	2.309574
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	-16.000000	0.000000
25%	12.000000	11.000000	0.000000	11.000000	0.000000	39619.000000	90.500000	13.000000	0.000000
50%	31.000000	25.000000	0.000000	25.000000	0.000000	58214.000000	248.000000	83.000000	0.000000
75%	191.000000	175.500000	0.000000	175.500000	0.000000	61281.500000	3552.000000	349.500000	0.000000
max	5324.000000	4348.000000	41.000000	4348.000000	1.000000	162873.000000	31166.000000	4229.000000	18.000000

8 rows × 29 columns

```python
def covidDataNum(covidData):
    covidDataNum = pd.DataFrame(covidData[["Date"]])
    covidDataNum["DailyConfirmed"] = covidData["DailyConfirmed"]
    covidDataNum["NewRecoveries"] = covidData["NewRecoveries"]
    covidDataNum["DailyDeaths"] = covidData["DailyDeaths"]
    for col in ["DailyConfirmed", "NewRecoveries", "DailyDeaths"]:
        covidDataNum.loc[:, col] = covidDataNum[col].astype(float)
    return covidDataNum


covidDataNum = covidDataNum(covidData)
covidDataNum
```

	Date	DailyConfirmed	NewRecoveries	DailyDeaths
0	2020-01-23	1.0	0.0	0.0
1	2020-01-24	2.0	0.0	0.0
2	2020-01-25	1.0	0.0	0.0
3	2020-01-26	0.0	0.0	0.0
4	2020-01-27	1.0	0.0	0.0
...	...	...	...	...
642	2021-10-26	3277.0	2856.0	10.0
643	2021-10-27	5324.0	3172.0	10.0
644	2021-10-28	3432.0	4348.0	15.0
645	2021-10-29	4248.0	3011.0	16.0
646	2021-10-30	3112.0	3912.0	14.0

647 rows × 4 columns

DAILY CONFIRMED

NEW RECOVERIES

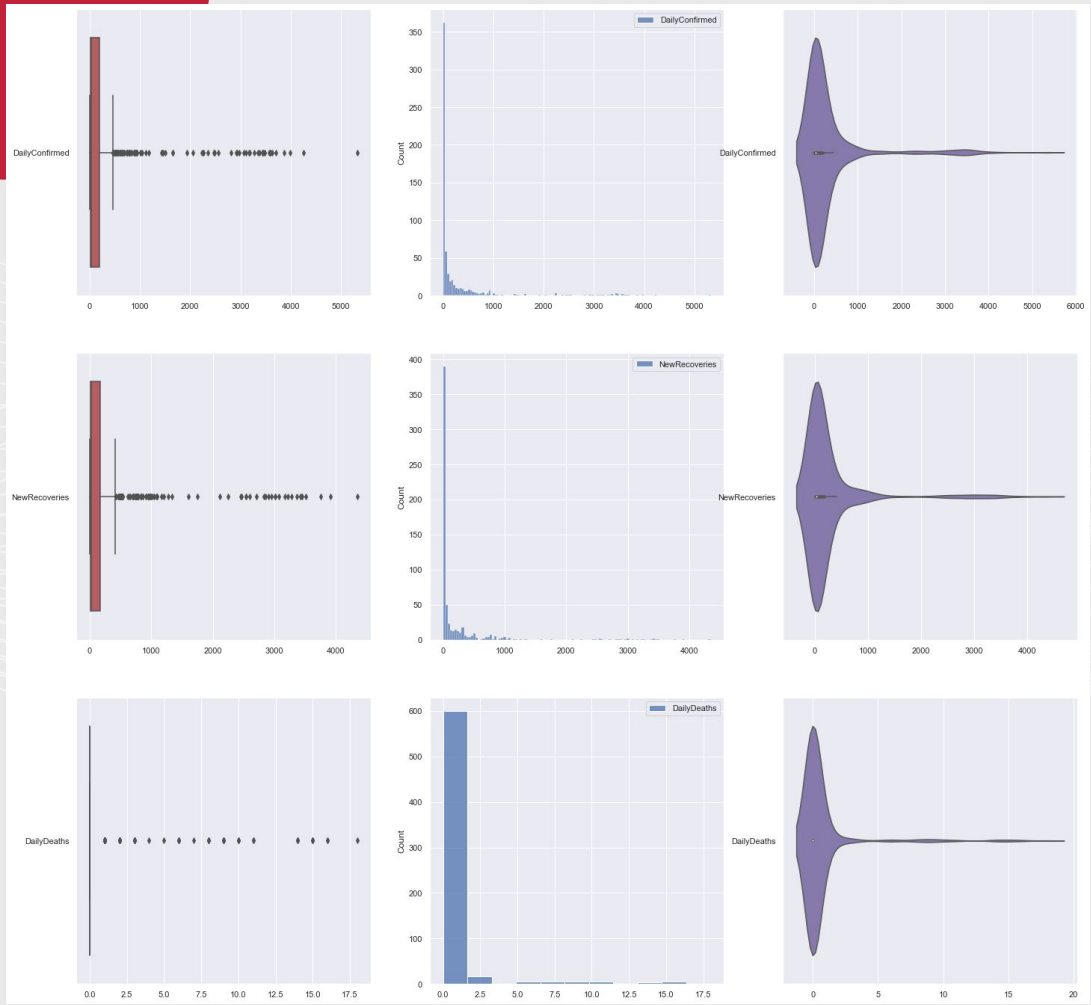
DAILY DEATHS

CUMULATIVE DATA

```
covidData_DailyConfirmed = pd.DataFrame(covidData["DailyConfirmed"])
covidData_NewRecoveries = pd.DataFrame(covidData["NewRecoveries"])
covidData_DailyDeaths= pd.DataFrame(covidData["DailyDeaths"])

covidData_CumConfirmed = pd.DataFrame(covidData_DailyConfirmed["DailyConfirmed"])
covidData_CumConfirmed["CumConfirmed"] = covidData_DailyConfirmed["DailyConfirmed"].cumsum()

covidData_CumRecoveries = pd.DataFrame(covidData_NewRecoveries["NewRecoveries"])
covidData_CumRecoveries["CumRecoveries"] = covidData_NewRecoveries["NewRecoveries"].cumsum()

covidData_CumDeaths = pd.DataFrame(covidData_DailyDeaths["DailyDeaths"])
covidData_CumDeaths["CumDeaths"] = covidData_DailyDeaths["DailyDeaths"].cumsum()

def covidData_Cum(covidData):
    covidData_Cum = pd.DataFrame(covidData[["Date"]])
    covidData_Cum["CumConfirmed"] = covidData_CumConfirmed["CumConfirmed"]
    covidData_Cum["CumRecoveries"] = covidData_CumRecoveries["CumRecoveries"]
    covidData_Cum["CumDeaths"] = covidData_CumDeaths["CumDeaths"]
    for col in ["CumConfirmed", "CumRecoveries", "CumDeaths"]:
        covidData_Cum.loc[:, col] = covidData_Cum[col].astype(float)
    return covidData_Cum
covidData_Cum = covidData_Cum(covidData)
covidData_Cum
```

	Date	CumConfirmed	CumRecoveries	CumDeaths
0	2020-01-23	1.0	0.0	0.0
1	2020-01-24	3.0	0.0	0.0
2	2020-01-25	4.0	0.0	0.0
3	2020-01-26	4.0	0.0	0.0
4	2020-01-27	5.0	0.0	0.0
...	...	...	...	...
642	2021-10-26	179175.0	148406.0	340.0
643	2021-10-27	184499.0	151578.0	350.0
644	2021-10-28	187931.0	155926.0	365.0
645	2021-10-29	192179.0	158937.0	381.0
646	2021-10-30	195291.0	162849.0	395.0

647 rows × 4 columns



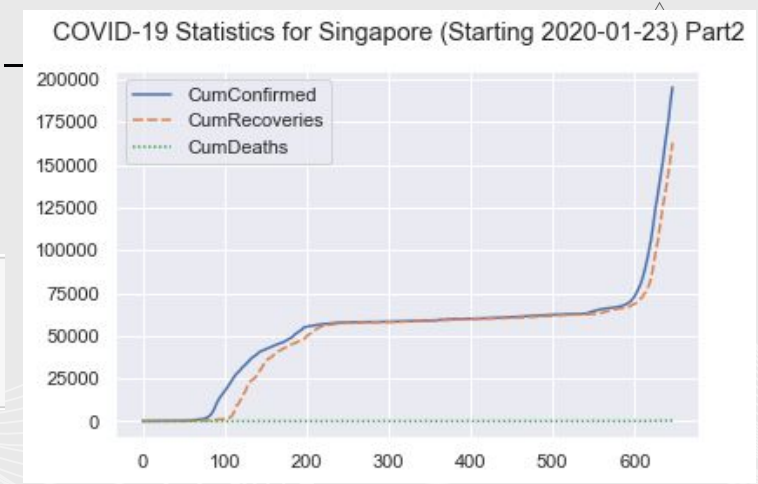
DAILY CONFIRMED

NEW RECOVERIES

DAILY DEATHS

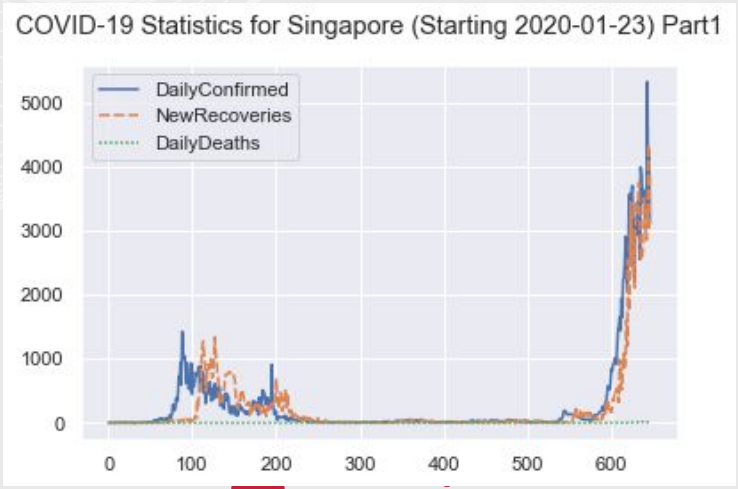
CUMULATIVE

```
fig = plt.figure()
plt.suptitle("COVID-19 Statistics for Singapore (Starting 2020-01-23) Part2")
sb.lineplot(data=covidData_Cum)
plt.show()
```



DAILY

```
fig = plt.figure()
plt.suptitle("COVID-19 Statistics for Singapore (Starting 2020-01-23) Part1")
sb.lineplot(data=covidDataNum)
plt.show()
```



DATA PREPARATION

```
d0 = datetime(2021, 6, 14)
d1 = datetime(2021, 10, 30)
delta = d1 - d0
print(delta.days+1)
```

```
139
```

```
covidData_Cum_part = covidData_Cum.tail(139)
covidData_Cum_part
```

	Date	CumConfirmed	CumRecoveries	CumDeaths
508	2021-06-14	62381.0	61892.0	34.0
509	2021-06-15	62395.0	61909.0	34.0
510	2021-06-16	62419.0	61929.0	34.0
511	2021-06-17	62446.0	61958.0	34.0
512	2021-06-18	62462.0	61985.0	34.0
...	...	...	...	...
642	2021-10-26	179175.0	148406.0	340.0
643	2021-10-27	184499.0	151578.0	350.0
644	2021-10-28	187931.0	155926.0	365.0
645	2021-10-29	192179.0	158937.0	381.0
646	2021-10-30	195291.0	162849.0	395.0

139 rows × 4 columns

```
covidDataNum_part = covidDataNum.tail(139)
covidDataNum_part
```

	Date	DailyConfirmed	NewRecoveries	DailyDeaths
508	2021-06-14	25.0	25.0	0.0
509	2021-06-15	14.0	17.0	0.0
510	2021-06-16	24.0	20.0	0.0
511	2021-06-17	27.0	29.0	0.0
512	2021-06-18	16.0	27.0	0.0
...	...	...	...	...
642	2021-10-26	3277.0	2856.0	10.0
643	2021-10-27	5324.0	3172.0	10.0
644	2021-10-28	3432.0	4348.0	15.0
645	2021-10-29	4248.0	3011.0	16.0
646	2021-10-30	3112.0	3912.0	14.0

139 rows × 4 columns

DATA PREPARATION

```
split_rate = 0.75
demarcation_point = round((646 - 508) * 0.75)
initial_cases = 62381.0
```

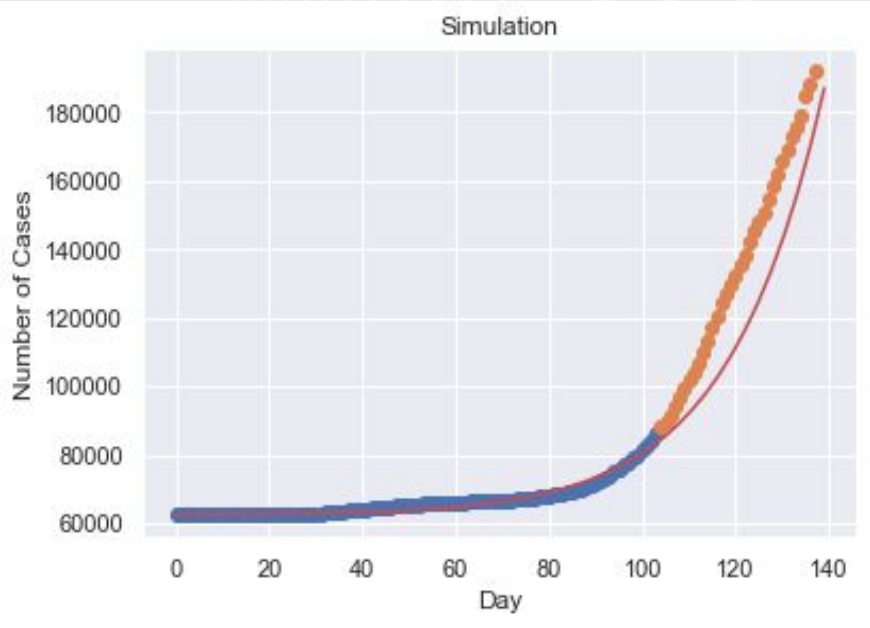
```
covid_cumconfirmed = covidData_Cum_part["CumConfirmed"]
train_cases_number = covid_cumconfirmed[0:demarcation_point]
train_day_time = len(train_cases_number)

train_xdata = np.array([ i for i in range(train_day_time) ])
train_ydata = train_cases_number

test_cases_number = covid_cumconfirmed[demarcation_point:138]
test_day_time = len(test_cases_number)

test_xdata = np.array([ i+104 for i in range(test_day_time) ])
test_ydata = test_cases_number

plt.scatter(train_xdata, train_ydata, label='data',s=5)
plt.scatter(test_xdata, test_ydata, label='data',s=5)
```

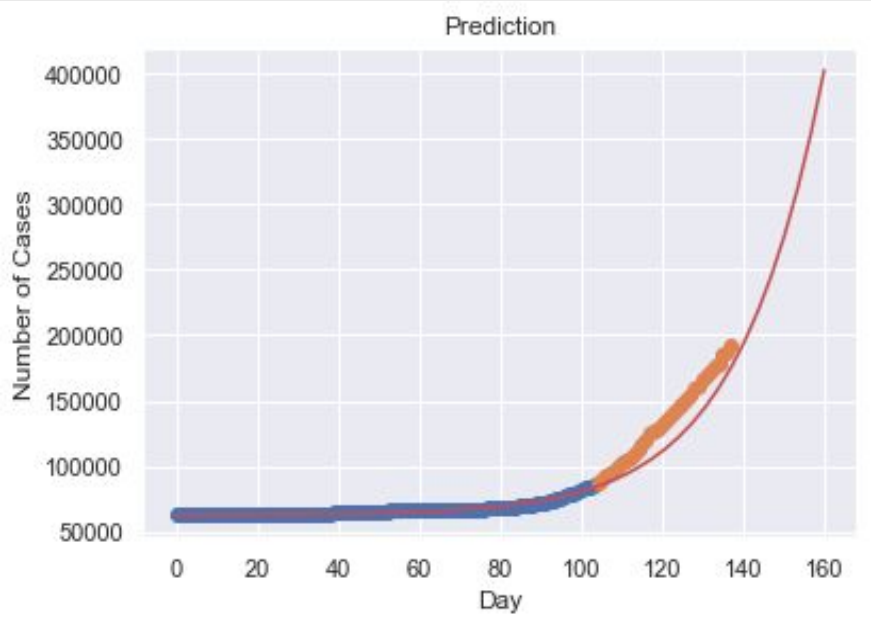
LOGISTIC REGRESSION MODEL

```
from scipy.optimize import curve_fit as curve_fit

def func(x, k, a, b):
 return k/(1+(k/b-1)*np.exp(-a*x))

popt, pcov = curve_fit(func, train_xdata, train_ydata - initial_cases, method='dogbox', \
                       bounds=([1000., 0.01, 10.],[5000000., 1.0, 1000.]))
k = popt[0]
a = popt[1]
b = popt[2]
```

```
plt.scatter(train_xdata, train_ydata, label='data')
plt.scatter(test_xdata, test_ydata, label='data')

x = np.linspace(0, len(train_xdata)+testdays+21)
y = func(x, *popt)

plt.plot(x, y + initial_cases,  color='r', label='fit')
plt.xlabel('Day')
plt.ylabel('Number of Cases')
plt.title('Prediction')
plt.show( )
```

```
plt.scatter(train_xdata, train_ydata, label='data')
plt.scatter(test_xdata, test_ydata, label='data')
testdays = 139 - demarcation_point

x = np.linspace(0, len(train_xdata)+testdays)
y = func(x, *popt)


plt.plot(x, y + initial_cases,  color='r', label='fit')
plt.xlabel('Day')
plt.ylabel('Number of Cases')
plt.title('Simulation')
plt.show( )
```


Simulation
Number of Cases
180000
160000
140000
120000
100000
80000
60000
0
20
40
60
80
100
120
140
Day


Prediction
Number of Cases
400000
350000
300000
250000
200000
150000
100000
50000
0
20
40
60
80
100
120
140
160
Day

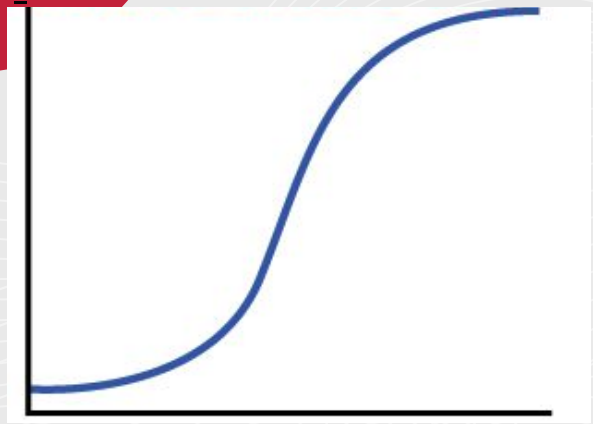
03.

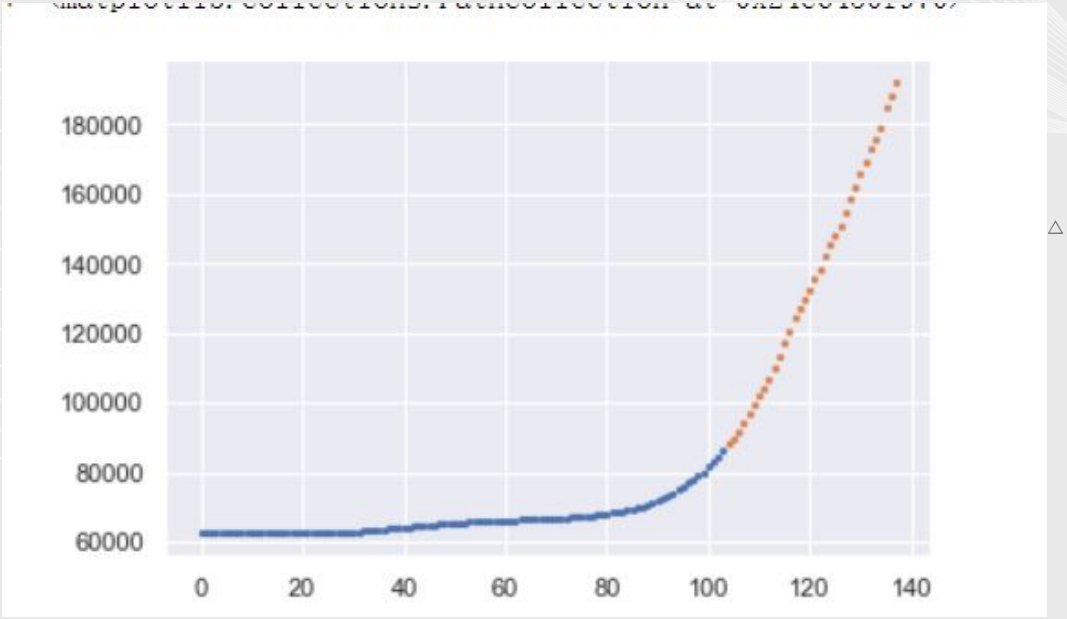
LOGISTIC REGRESSION MODEL

$$P(t) = \frac{K}{1 + (\frac{K}{P_0} - 1)e^{-rt}}$$

P: confirmed cases
K: highest possible confirmed cases
r: initial growth rate
P0: initial confirmed number

```
from scipy.optimize import curve_fit as curve_fit

def func(x, k, a, b):
 return k/(1+(k/b-1)*np.exp(-a*x))

popt, pcov = curve_fit(func, train_xdata, train_ydata - initial_cases, method='dogbox', \
                       bounds=([1000., 0.01, 10.],[5000000., 1.0, 1000.]))
k = popt[0]
a = popt[1]
b = popt[2]
```




180000
160000
140000
120000
100000
80000
60000
0
20
40
60
80
100
120
140

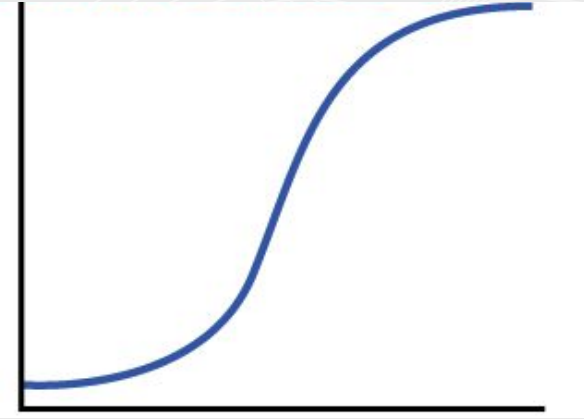
04.

ERROR ANALYSIS & CONCLUSION

200K in 21 days ≈ 10K/DAY

Error Analysis

Methodology: Model

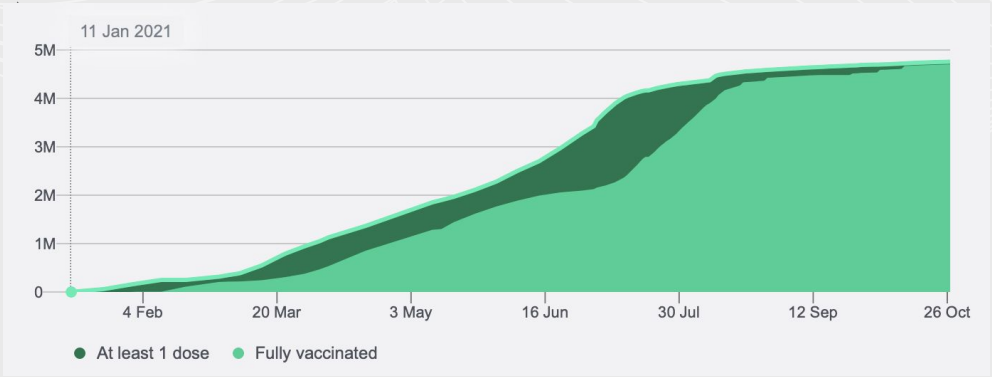


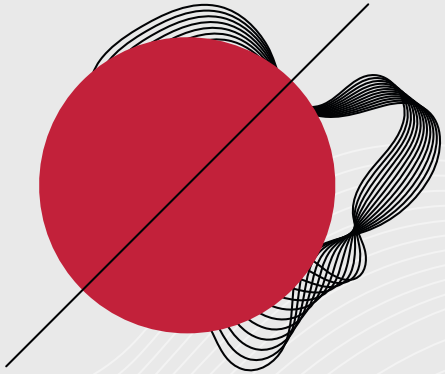
Error Analysis

Endemic itself: Delta variant

Error Analysis

Factual: Vaccination





THANKS!

Time for Q&A

Group A+PLZ!

ZHU BANGYUAN - Logistic Regression Model

LI JINGYUAN - Data Preparation

YANG PEISHI - Background & Exploration

YU LINYAN - Visualisation

REFERENCES

- https://data.world/hxchua/covid-19-singapore/workspace/file?filename=Covid-19+SG.xlsx
- https://www.covid.gov.sg/unwell/overview
- https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc
- https://www.forbes.com/sites/siladityaray/2021/10/21/living-with-covid-strategy-under-threat-as-singapore-uk-face-sharp-rise-in-cases/?sh=76b0c6f73a77
- https://www.straitstimes.com/singapore/politics/longer-waiting-times-as-singapores-healthcare-capacity-impacted-by-recent
- https://www.iconfinder.com/icons/3030663/captures_collection_data_recopilation_sources_icon
- https://thenounproject.com/term/development-policies/
- https://www.iconfinder.com/icons/5664262/analysis_chart_data_inspection_trend_icon
- https://www.moh.gov.sg