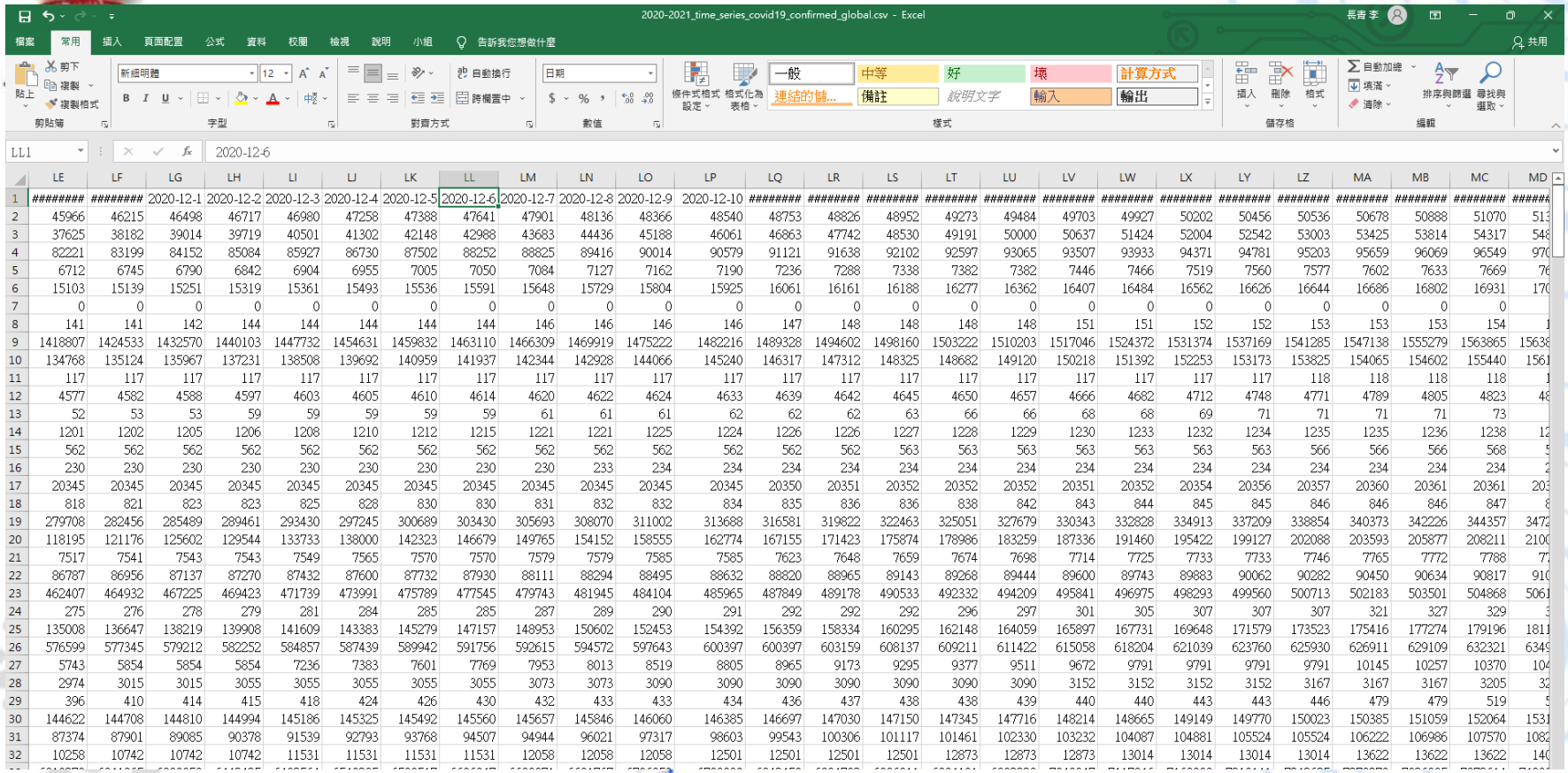
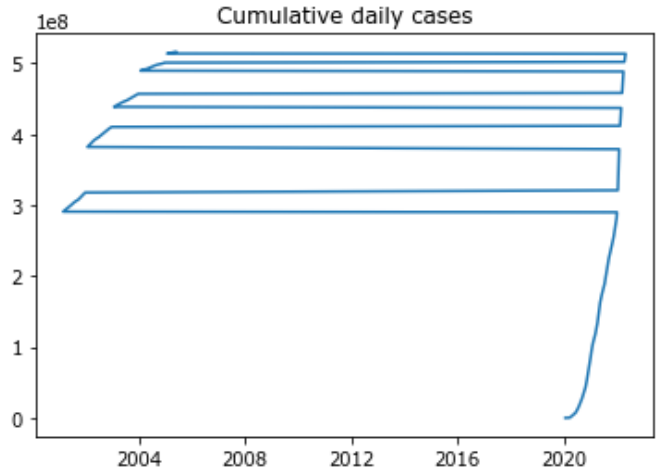
data



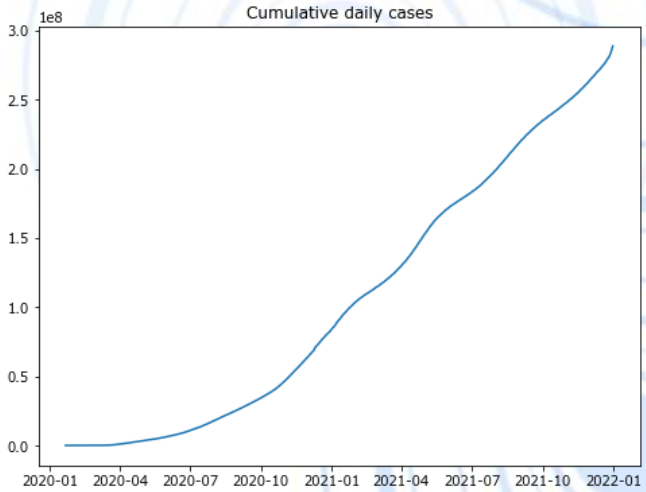
	LE	LF	LG	LH	LI	LJ	LK	LL	LM	LN	LO	LP	LQ	LR	LS	LT	LU	LV	LW	LX	LY	LZ	MA	MB	MC	MD
1	########	########	2020-12-1	2020-12-2	2020-12-3	2020-12-4	2020-12-5	2020-12-6	2020-12-7	2020-12-8	2020-12-9	2020-12-10	########	########	########	########	########	########	########	########	########	########	########	########	########	########
2	45966	46215	46498	46717	46980	47258	47388	47641	47901	48136	48366	48540	48753	48826	48952	49273	49484	49703	49927	50202	50456	50536	50678	50888	51070	513
3	37625	38182	39014	39719	40501	41302	42148	42988	43683	44436	45188	46061	46863	47742	48530	49191	50000	50637	51424	52004	52542	53003	53425	53814	54317	548
4	82221	83199	84152	85084	85927	86730	87502	88252	88825	89416	90014	90579	91121	91638	92102	92597	93065	93507	93933	94371	94781	95203	95659	96069	96549	970
5	6712	6745	6790	6842	6904	6955	7005	7050	7084	7127	7162	7190	7236	7288	7338	7382	7382	7446	7466	7519	7560	7577	7602	7633	7669	76
6	15103	15139	15251	15319	15361	15493	15536	15591	15648	15729	15804	15925	16061	16161	16188	16277	16362	16407	16484	16562	16626	16644	16686	16802	16931	170
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	141	141	142	144	144	144	144	144	146	146	146	146	147	148	148	148	148	151	151	152	152	153	153	153	154	1
9	1418807	1424533	1432570	1440103	1447732	1454631	1459832	1463110	1466309	1469919	1475222	1482216	1489328	1494602	1498160	1503222	1510203	1517046	1524372	1531374	1537169	1541285	1547138	1555279	1563865	15638
10	134768	135124	135967	137231	138508	139692	140959	141937	142344	142928	144066	145240	146317	147312	148325	148682	149120	150218	151392	152253	153173	153825	154065	154602	155440	1561
11	117	117	117	117	117	117	117	117	117	117	117	117	117	117	117	117	117	117	117	117	117	118	118	118	118	1
12	4577	4582	4588	4597	4603	4605	4610	4614	4620	4622	4624	4633	4639	4642	4645	4650	4657	4666	4682	4712	4748	4771	4789	4805	4823	48
13	52	53	53	59	59	59	59	59	61	61	61	62	62	62	63	66	66	68	68	69	71	71	71	71	73	
14	1201	1202	1205	1206	1208	1210	1212	1215	1221	1221	1225	1224	1226	1226	1227	1228	1229	1230	1233	1232	1234	1235	1235	1236	1238	12
15	562	562	562	562	562	562	562	562	562	562	562	562	562	562	563	563	563	563	563	563	563	566	566	566	568	5
16	230	230	230	230	230	230	230	230	230	233	234	234	234	234	234	234	234	234	234	234	234	234	234	234	234	2
17	20345	20345	20345	20345	20345	20345	20345	20345	20345	20345	20345	20345	20350	20351	20352	20352	20352	20351	20352	20354	20356	20357	20360	20361	20361	203
18	818	821	823	823	825	828	830	830	831	832	832	834	835	836	836	838	842	843	844	845	845	846	846	846	847	8
19	279708	282456	285489	289461	293430	297245	300689	303430	305693	308070	311002	313688	316581	319822	322463	325051	327679	330343	332828	334913	337209	338854	340373	342226	344357	3472
20	118195	121176	125602	129544	133733	138000	142323	146679	149765	154152	158555	162774	167155	171423	175874	178986	183259	187336	191460	195422	199127	202088	203593	205877	208211	2100
21	7517	7541	7543	7543	7549	7565	7570	7570	7579	7579	7585	7585	7623	7648	7659	7674	7698	7714	7725	7733	7733	7746	7765	7772	7788	77
22	86787	86956	87137	87270	87432	87600	87732	87930	88111	88294	88495	88632	88820	88965	89143	89268	89444	89600	89743	89883	90062	90282	90450	90634	90817	910
23	462407	464932	467225	469423	471739	473991	475789	477545	479743	481945	484104	485965	487849	489178	490533	492332	494209	495841	496975	498293	499560	500713	502183	503501	504868	5061
24	275	276	278	279	281	284	285	285	287	289	290	291	292	292	292	296	297	301	305	307	307	307	321	327	329	3
25	135008	136647	138219	139908	141609	143383	145279	147157	148953	150602	152453	154392	156359	158334	160295	162148	164059	165897	167731	169648	171579	173523	175416	177274	179196	1811
26	576599	577345	579212	582252	584857	587439	589942	591756	592615	594572	597643	600397	600397	603159	608137	609211	611422	615058	618204	621039	623760	625930	626911	629109	632321	6349
27	5743	5854	5854	5854	7236	7383	7601	7769	7953	8013	8519	8805	8965	9173	9295	9377	9511	9672	9791	9791	9791	9791	10145	10257	10370	104
28	2974	3015	3015	3055	3055	3055	3055	3055	3073	3073	3090	3090	3090	3090	3090	3090	3090	3152	3152	3152	3152	3167	3167	3167	3205	32
29	396	410	414	415	418	424	426	430	432	433	433	434	436	437	438	438	439	440	440	443	443	446	479	479	519	5
30	144622	144708	144810	144994	145186	145325	145492	145560	145657	145846	146060	146385	146697	147030	147150	147345	147716	148214	148665	149149	149770	150023	150385	151059	152064	1531
31	87374	87901	89085	90378	91539	92793	93768	94507	94944	96021	97317	98603	99543	100306	101117	101461	102330	103232	104087	104881	105524	105524	106222	106986	107570	1082
32	10258	10742	10742	10742	11531	11531	11531	11531	12058	12058	12058	12501	12501	12501	12501	12873	12873	12873	13014	13014	13014	13014	13622	13622	13622	140

原本covid-19的data格式不一致:



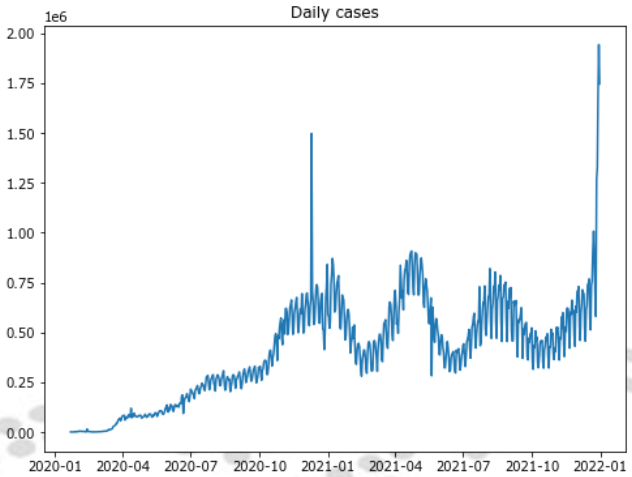
資料誤判

整理後: 累積分布



單位: 確診人數

整理後: 每日值變化



單位: 確診人數

數據歸一化

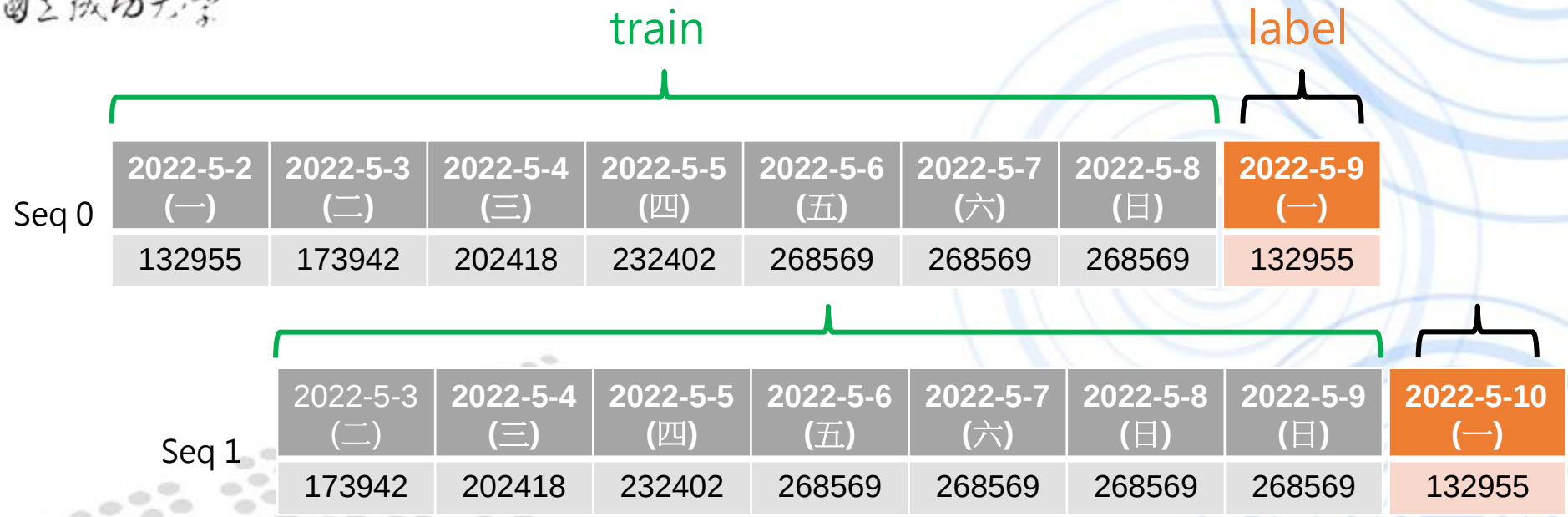
轉換後:
[[0.00030536]
[0.]
[0.00012495]
[0.00026259]
[0.00038955]

最小值

預處理: (data包成一組)
- 一組含7個序列數據->搭配1個label

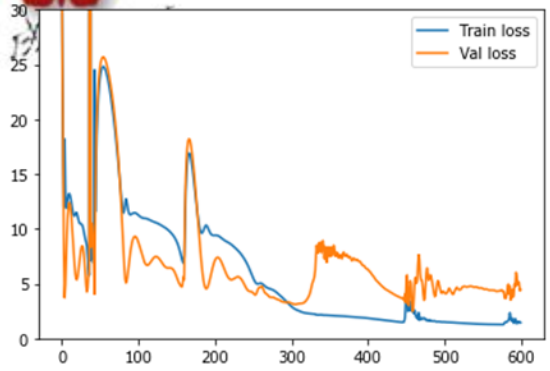
train model

單位: 全球確診人數(這裡以台灣為例)

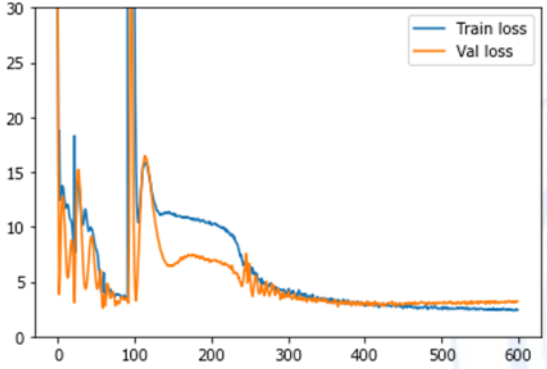


先設定每7天一循環，第8天當label值
之後會測試，不同循環天數 所帶來的影響，找出最佳表現

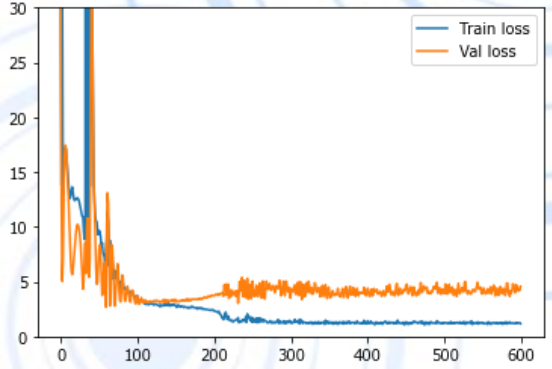
單一LSTM



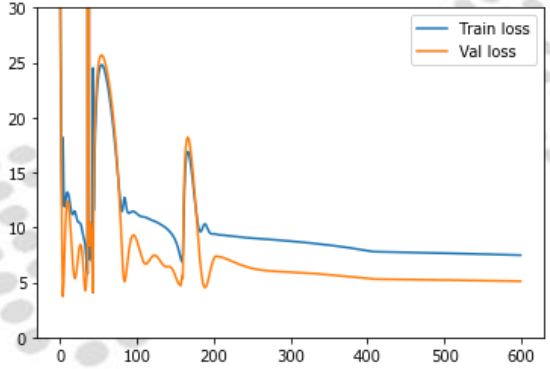
雙LSTM



三層LSTM



根據val
動態learning rate



模型評估指標

MSE, MAE算法:

```
if performance_eval == 'MAE':  # https://pytorch.org/docs/stable/generc
    evaluation_func = torch.nn.L1Loss(reduction='mean')  # (平均絕對誤差
    best_evaluation = 1.0  # 初始最高，越低越好
elif performance_eval == 'MSE':  # https://pytorch.org/docs/stable/gene
    evaluation_func = torch.nn.MSELoss(reduction='mean')  # (均方誤差)
    best_evaluation = 1.0  # 初始最高，越低越好
```

R^2算法:

$$R^2 = 1 - \frac{SSE}{SST}$$

$$SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$SST = SSR + SSE = \sum_{i=1}^{n}(y_i - \bar{y})^2$$

```
evaluation = evaluation_func(val_label, y
evaluation = evaluation.item() / np.sum( np.square( val_label.cpu().numpy() - np.mean(val_label.cpu().numpy()
evaluation = 1 - evaluation
```

MAPE算法:

```
evaluation = np.mean(np.abs(val_label.cpu().numpy() - y_pred_val.cpu().numpy()) / np.abs(val_label.cpu().numpy()))*100
```

(Epoch = 500回合)

歸納01

(初始化權重已固定，方便研究比較)

模型評估指標	MSE	MAE	R^2	MAPE
LSTM layers = 1	0.0105	0.0733	0.3572	18.88
LSTM layers = 2	0.0110	0.0708	0.3233	17.96
LSTM layers = 3	0.0114	0.0713	0.3005	18.24

歸納02

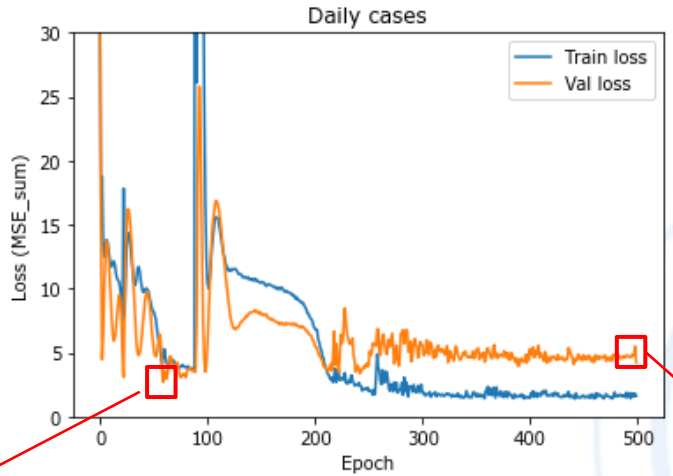
模型評估指標	MSE	MAE	R^2	MAPE
LSTM layers = 2 Hidden size = 256	0.0111	0.0721	0.3224	18.43
LSTM layers = 2 Hidden size = 512	0.0110	0.0708	0.3233	17.96
LSTM layers = 2 Hidden size = 768	0.0129	0.0744	0.2089	18.98

(Epoch = 500回合)

歸納03 (初始化權重已固定，方便研究比較)

模型評估指標	MSE	MAE	R^2	MAPE
LSTM layers = 2 Hidden size = 512 Seq_length = 3	0.0169	0.0849	-0.0208	22.31
LSTM layers = 2 Hidden size = 512 Seq_length = 7	0.0110	0.0708	0.3233	17.96
LSTM layers = 2 Hidden size = 512 Seq_length = 14	0.0146	0.0789	0.1131	20.85

Best Model



訓練loss值

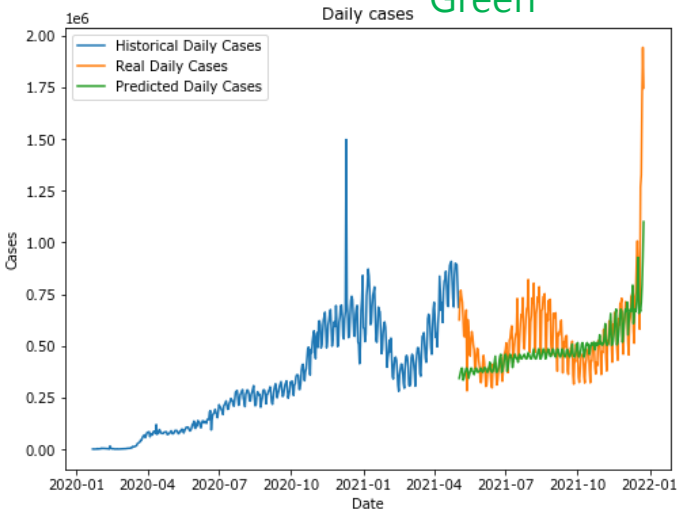
橘色=真實值 (確診人數)
綠色/紅色=預測值 (確診人數)

Save model weights

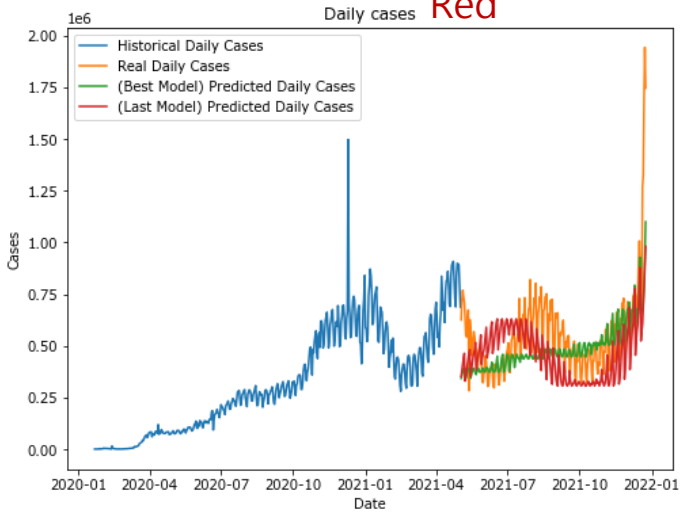
val loss: 2.6999

val loss: 4.2398

Green

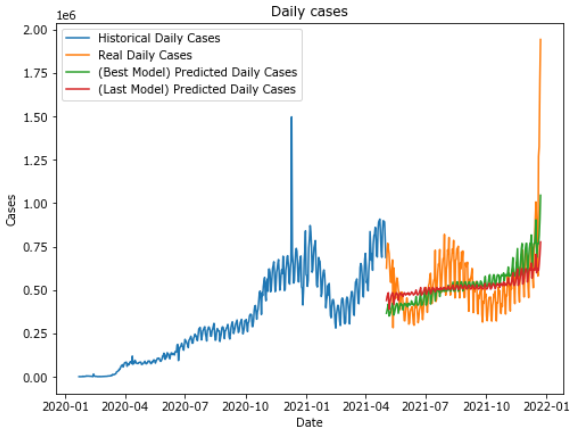


Red

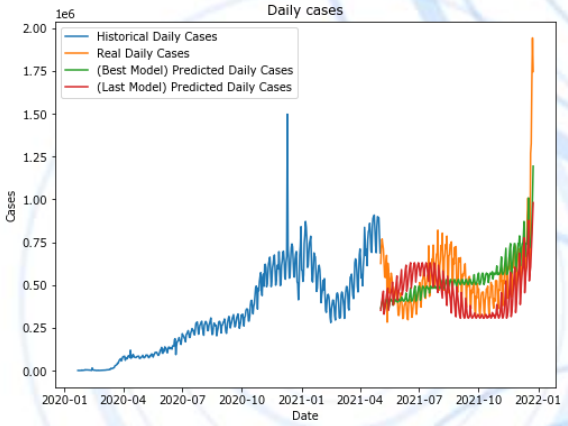


預測結果

資料前處理(ori)



資料前處理(after)



在跑其它資料集時，發現之前寫的data前處理code算錯天數(已修正)

```python
def create_sequence(data, seq_length):
    x_list = []
    y_list = []
    for i in range(len(data)-seq_length-1):
        x = data[i:i+seq_length]  # 0~6
        y = data[i+seq_length]  # 7
        x_list.append(x)
        y_list.append(y)

    return np.array(x_list), np.array(y_list)
```

```python
def create_sequence(data, seq_length):
    x_list = []
    y_list = []
    for i in range(len(data)-seq_length):
        x = data[i:i+seq_length]  # 0~6
        y = data[i+seq_length]  # 7
        x_list.append(x)
        y_list.append(y)

    return np.array(x_list), np.array(y_list)
```