

Welcome to Covid19 Data Analysis Notebook

Let's Import the modules ¶

```
In [120]: import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

print('Modules are imported.')
```

Modules are imported.

Task 2

Task 2.1: importing covid19 dataset

importing "Covid19_Confirmed_dataset.csv" from "J/Dataset" folder.

```
In [121]: corona_dataset_csv = pd.read_csv('Dataset/covid19_Confirmed_dataset.csv')
corona_dataset_csv.head(10)
```

```
In [121]: corona_dataset_csv = pd.read_csv('Dataset/covid19_Confirmed_dataset.csv')
corona_dataset_csv.head(10)
```

```
Out[121]:
```

	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	...	4/21/20	4/22/20	4/23/20	4/24/20	4/25/20	4/26/20
0	NaN	Afghanistan	33.0000	65.0000	0	0	0	0	0	0	...	1092	1176	1279	1351	1463	1531
1	NaN	Albania	41.1533	20.1683	0	0	0	0	0	0	...	609	634	663	678	712	726
2	NaN	Algeria	28.0339	1.6596	0	0	0	0	0	0	...	2811	2910	3007	3127	3256	3382
3	NaN	Andorra	42.5063	1.5218	0	0	0	0	0	0	...	717	723	723	731	738	738
4	NaN	Angola	-11.2027	17.8739	0	0	0	0	0	0	...	24	25	25	25	25	26
5	NaN	Antigua and Barbuda	17.0608	-61.7964	0	0	0	0	0	0	...	23	24	24	24	24	24
6	NaN	Argentina	-38.4161	-63.6167	0	0	0	0	0	0	...	3031	3144	3435	3607	3780	3892
7	NaN	Armenia	40.0691	45.0382	0	0	0	0	0	0	...	1401	1473	1523	1596	1677	1746
8	Australian Capital Territory	Australia	-35.4735	149.0124	0	0	0	0	0	0	...	104	104	104	105	106	106
9	New South Wales	Australia	-33.8688	151.2093	0	0	0	0	3	4	...	2969	2971	2976	2982	2994	3002

10 rows × 104 columns

Let's check the shape of the dataframe

```
In [122]: corona_dataset_csv.shape
```

Let's check the shape of the dataframe

```
In [122]: corona_dataset_csv.shape
```

```
Out[122]: (266, 104)
```

Task 2.2: Delete the useless columns

```
In [123]: corona_dataset_csv.drop(['Lat', 'Long'], axis=1, inplace=True)
```

```
In [124]: corona_dataset_csv.head(10)
```

```
Out[124]:
```

	Province/State	Country/Region	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20	1/29/20	...	4/21/20	4/22/20	4/23/20	4/24/20	4/25/20	4/26/20	4
0	NaN	Afghanistan	0	0	0	0	0	0	0	0	...	1092	1176	1279	1351	1463	1531	
1	NaN	Albania	0	0	0	0	0	0	0	0	...	609	634	663	678	712	726	
2	NaN	Algeria	0	0	0	0	0	0	0	0	...	2811	2910	3007	3127	3256	3382	
3	NaN	Andorra	0	0	0	0	0	0	0	0	...	717	723	723	731	738	738	
4	NaN	Angola	0	0	0	0	0	0	0	0	...	24	25	25	25	25	26	
5	NaN	Antigua and Barbuda	0	0	0	0	0	0	0	0	...	23	24	24	24	24	24	
6	NaN	Argentina	0	0	0	0	0	0	0	0	...	3031	3144	3435	3607	3780	3892	
7	NaN	Armenia	0	0	0	0	0	0	0	0	...	1401	1473	1523	1596	1677	1746	
8	Australian Capital Territory	Australia	0	0	0	0	0	0	0	0	...	104	104	104	105	106	106	
9	New South Wales	Australia	0	0	0	0	3	4	4	4	...	2969	2971	2976	2982	2994	3002	

10 rows x 102 columns

Task 2.3: Aggregating the rows by the country

```
In [125]: corona_dataset_aggregated = corona_dataset_csv.groupby("Country/Region").sum()
```

```
In [126]: corona_dataset_aggregated.head(10)
```

```
Out[126]:
```

	Country/Region	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20	1/29/20	1/30/20	1/31/20	...	4/21/20	4/22/20	4/23/20	4/24/20	4/25/20	4/26/20	4/27/20
0	Afghanistan	0	0	0	0	0	0	0	0	0	0	...	1092	1176	1279	1351	1463	1531	1603
1	Albania	0	0	0	0	0	0	0	0	0	0	...	609	634	663	678	712	726	750
2	Algeria	0	0	0	0	0	0	0	0	0	0	...	2811	2910	3007	3127	3256	3382	3508
3	Andorra	0	0	0	0	0	0	0	0	0	0	...	717	723	723	731	738	738	745
4	Angola	0	0	0	0	0	0	0	0	0	0	...	24	25	25	25	25	26	27
5	Antigua and Barbuda	0	0	0	0	0	0	0	0	0	0	...	23	24	24	24	24	24	25
6	Argentina	0	0	0	0	0	0	0	0	0	0	...	3031	3144	3435	3607	3780	3892	4004
7	Armenia	0	0	0	0	0	0	0	0	0	0	...	1401	1473	1523	1596	1677	1746	1815
8	Australia	0	0	0	0	4	5	5	6	9	9	...	6645	6652	6662	6677	6694	6714	6734
9	Austria	0	0	0	0	0	0	0	0	0	0	...	14873	14925	15002	15071	15148	15225	15302

10 rows x 100 columns

Course | Online Co | COVID19 Data Analy | notebook 2 | Inbox (104) - krutika | convolutions_text - | covid - Google Docs | +

d3c33hcgivew3.cloudfront.net/6pdyZ-wwTWWXcmfsmC1IsA_9fab6d0704b64f98a6ac0d11a834e855_The-Notebook---Covid19-Data-Analysis.html?Expires=1593648...

Apps YouTube Maps Gmail News Translate Basic Data Types ... DeepLearningBook... Libraries — ML Glo... What is the meanin...

```
In [127]: corona_dataset_aggregated.shape
```

```
Out[127]: (187, 100)
```

Task 2.4: Visualizing data related to a country for example China

visualization always helps for better understanding of our data.

```
In [128]: corona_dataset_aggregated.loc['China'].plot()
corona_dataset_aggregated.loc['Italy'].plot()
corona_dataset_aggregated.loc['Spain'].plot()
plt.legend()
```

```
Out[128]: <matplotlib.legend.Legend at 0x29aab8a2788>
```

Task3: Calculating a good measure

Course | Online Co | COVID19 Data Analy | notebook 2 | Inbox (104) - krutika | convolutions_text - | covid - Google Docs | +

d3c33hcgivew3.cloudfront.net/6pdyZ-wwTWWXcmfsmC1IsA_9fab6d0704b64f98a6ac0d11a834e855_The-Notebook---Covid19-Data-Analysis.html?Expires=1593648...

Apps YouTube Maps Gmail News Translate Basic Data Types ... DeepLearningBook... Libraries — ML Glo... What is the meanin...

Task3: Calculating a good measure

we need to find a good measure represented as a number, describing the spread of the virus in a country.

```
In [129]: corona_dataset_aggregated.loc['China'].plot()
```

```
Out[129]: <matplotlib.axes._subplots.AxesSubplot at 0x29aacac1ec8>
```

task 3.1: calculating the first derivative of the curve

```
In [130]: corona_dataset_aggregated.loc['China'].diff().plot()
```

```
Out[130]: <matplotlib.axes._subplots.AxesSubplot at 0x29aab141908>
```

Course | Online Co | COVID19 Data Analy | notebook 2 | Inbox (104) - krutika | convolutions_text - | covid - Google Docs | +

d3c33hcgivew3.cloudfront.net/6pdyZ-wwTWWXcmfsmC1IsA_9fab6d0704b64f98a6ac0d11a834e855_The-Notebook---Covid19-Data-Analysis.html?Expires=1593648...

Apps YouTube Maps Gmail News Translate Basic Data Types |... DeepLearningBook... Libraries — ML Glo... What is the meanin...

```
In [130]: corona_dataset_aggregated.loc['China'].diff().plot()
Out[130]: <matplotlib.axes._subplots.AxesSubplot at 0x29aab141908>
```

task 3.2: find maximum infection rate for China

```
In [131]: corona_dataset_aggregated.loc['China'].diff().max()
Out[131]: 15136.0
```

```
In [132]: corona_dataset_aggregated.loc['Italy'].diff().max()
Out[132]: 6557.0
```

```
In [133]: corona_dataset_aggregated.loc['Spain'].diff().max()
```

Windows Taskbar: Type here to search | [Taskbar icons] | ENG IN 08:27 PM 30-06-2020

Course | Online Co | COVID19 Data Analy | notebook 2 | Inbox (104) - krutika | convolutions_text - | covid - Google Docs | +

d3c33hcgivew3.cloudfront.net/6pdyZ-wwTWWXcmfsmC1IsA_9fab6d0704b64f98a6ac0d11a834e855_The-Notebook---Covid19-Data-Analysis.html?Expires=1593648...

Apps YouTube Maps Gmail News Translate Basic Data Types |... DeepLearningBook... Libraries — ML Glo... What is the meanin...

```
Out[132]: 6557.0
```

```
In [133]: corona_dataset_aggregated.loc['Spain'].diff().max()
Out[133]: 9630.0
```

Task 3.3: find maximum infection rate for all of the countries.

```
In [134]: countries = list(corona_dataset_aggregated.index)
max_infection_rates = []
for country in countries :
    max_infection_rates.append(corona_dataset_aggregated.loc[country].diff().max())
corona_dataset_aggregated['max infection rate'] = max_infection_rates
```

```
In [135]: corona_dataset_aggregated.head()
Out[135]:
```

	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20	1/29/20	1/30/20	1/31/20	...	4/22/20	4/23/20	4/24/20	4/25/20	4/26/20	4/27/20	4/28/20
Country/Region																		
Afghanistan	0	0	0	0	0	0	0	0	0	0	...	1176	1279	1351	1463	1531	1703	1703
Albania	0	0	0	0	0	0	0	0	0	0	...	634	663	678	712	726	736	736
Algeria	0	0	0	0	0	0	0	0	0	0	...	2910	3007	3127	3256	3382	3517	3517
Andorra	0	0	0	0	0	0	0	0	0	0	...	723	723	731	738	738	743	743
Angola	0	0	0	0	0	0	0	0	0	0	...	25	25	25	25	26	27	27

5 rows x 101 columns

Windows Taskbar: Type here to search | [Taskbar icons] | ENG IN 08:27 PM 30-06-2020

Task 3.4: create a new dataframe with only needed column

```
In [136]: corona_data = pd.DataFrame(corona_dataset_aggregated['max_infection_rate'])
```

```
In [137]: corona_data.head()
```

Out[137]:

Country/Region	max_infection_rate
Afghanistan	232.0
Albania	34.0
Algeria	199.0
Andorra	43.0
Angola	5.0

Task4:

- Importing the WorldHappinessReport.csv dataset
- selecting needed columns for our analysis
- join the datasets
- calculate the correlations as the result of our analysis

Task 4.1 : importing the dataset

Task 4.1 : importing the dataset

```
In [138]: world_happiness_report = pd.read_csv("Dataset/worldwide_happiness_report.csv")
world_happiness_report.head()
```

Out[138]:

	Overall rank	Country or region	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption
0	1	Finland	7.769	1.340	1.587	0.986	0.596	0.153	0.393
1	2	Denmark	7.600	1.383	1.573	0.996	0.592	0.252	0.410
2	3	Norway	7.554	1.488	1.582	1.028	0.603	0.271	0.341
3	4	Iceland	7.494	1.380	1.624	1.026	0.591	0.354	0.118
4	5	Netherlands	7.488	1.396	1.522	0.999	0.557	0.322	0.298

```
In [139]: world_happiness_report.shape
```

Out[139]: (156, 9)

Task 4.2: let's drop the useless columns

```
In [140]: columns_to_dropped = ['Overall rank', 'Score', 'Generosity', 'Perceptions of corruption']
world_happiness_report.drop(columns_to_dropped,axis=1, inplace=True)
```

```
In [141]: world_happiness_report.head()
```

Out[141]:

Course | Online Co | COVID19 Data Analy | notebook 2 | Inbox (104) - krutika | convolutions_text - | covid - Google Docs | +

d3c33hcgiewv3.cloudfront.net/6pdyZ-wwTWWXcmfMC1IsA_9fab6d0704b64f98a6ac0d11a834e855_The-Notebook---Covid19-Data-Analysis.html?Expires=1593648...

Apps YouTube Maps Gmail News Translate Basic Data Types [...] DeepLearningBook... Libraries — ML Glo... What is the meanin...

```
In [141]: world_happiness_report.head()
```

Out[141]:

	Country or region	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices
0	Finland	1.340	1.587	0.986	0.596
1	Denmark	1.383	1.573	0.996	0.592
2	Norway	1.488	1.582	1.028	0.603
3	Iceland	1.380	1.624	1.026	0.591
4	Netherlands	1.396	1.522	0.999	0.557

Task 4.3: changing the indices of the dataframe

```
In [142]: world_happiness_report.set_index(['Country or region'], inplace=True)
world_happiness_report.head()
```

Out[142]:

	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices
Country or region				
Finland	1.340	1.587	0.986	0.596
Denmark	1.383	1.573	0.996	0.592
Norway	1.488	1.582	1.028	0.603
Iceland	1.380	1.624	1.026	0.591
Netherlands	1.396	1.522	0.999	0.557

Type here to search

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000

Course | Online Co | COVID19 Data Analy | notebook 2 | Inbox (104) - krutika | convolutions_text - | covid - Google Docs | +

d3c33hcgiewv3.cloudfront.net/6pdyZ-wwTWWXcmfMC1IsA_9fab6d0704b64f98a6ac0d11a834e855_The-Notebook---Covid19-Data-Analysis.html?Expires=1593648...

Apps YouTube Maps Gmail News Translate Basic Data Types [...] DeepLearningBook... Libraries — ML Glo... What is the meanin...

Task 4.4: now let's join two dataset we have prepared

Corona Dataset :

```
In [143]: corona_data.head()
```

Out[143]:

	max infection rate
Country/Region	
Afghanistan	232.0
Albania	34.0
Algeria	199.0
Andorra	43.0
Angola	5.0

wold happiness report Dataset :

```
In [144]: world_happiness_report.head()
```

Out[144]:

	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices
Country or region				
Finland	1.340	1.587	0.986	0.596
Denmark	1.383	1.573	0.996	0.592

Type here to search

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000

In [144]: `world_happiness_report.head()`

Out[144]:

	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices
Country or region				
Finland	1.340	1.587	0.986	0.596
Denmark	1.383	1.573	0.996	0.592
Norway	1.488	1.582	1.028	0.603
Iceland	1.380	1.624	1.026	0.591
Netherlands	1.396	1.522	0.999	0.557

In [145]: `data = world_happiness_report.join(corona_data).copy()`
`data.head()`

Out[145]:

	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	max infection rate
Country or region					
Finland	1.340	1.587	0.986	0.596	267.0
Denmark	1.383	1.573	0.996	0.592	391.0
Norway	1.488	1.582	1.028	0.603	386.0
Iceland	1.380	1.624	1.026	0.591	99.0
Netherlands	1.396	1.522	0.999	0.557	1346.0

Task 4.5: correlation matrix

In [146]: `data.corr()`
it is representing the correlation between every two columns of our dataset

Out[146]:

	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	max infection rate
GDP per capita	1.000000	0.754906	0.835462	0.379079	0.250118
Social support	0.754906	1.000000	0.719009	0.447333	0.191958
Healthy life expectancy	0.835462	0.719009	1.000000	0.390395	0.289263
Freedom to make life choices	0.379079	0.447333	0.390395	1.000000	0.078196
max infection rate	0.250118	0.191958	0.289263	0.078196	1.000000

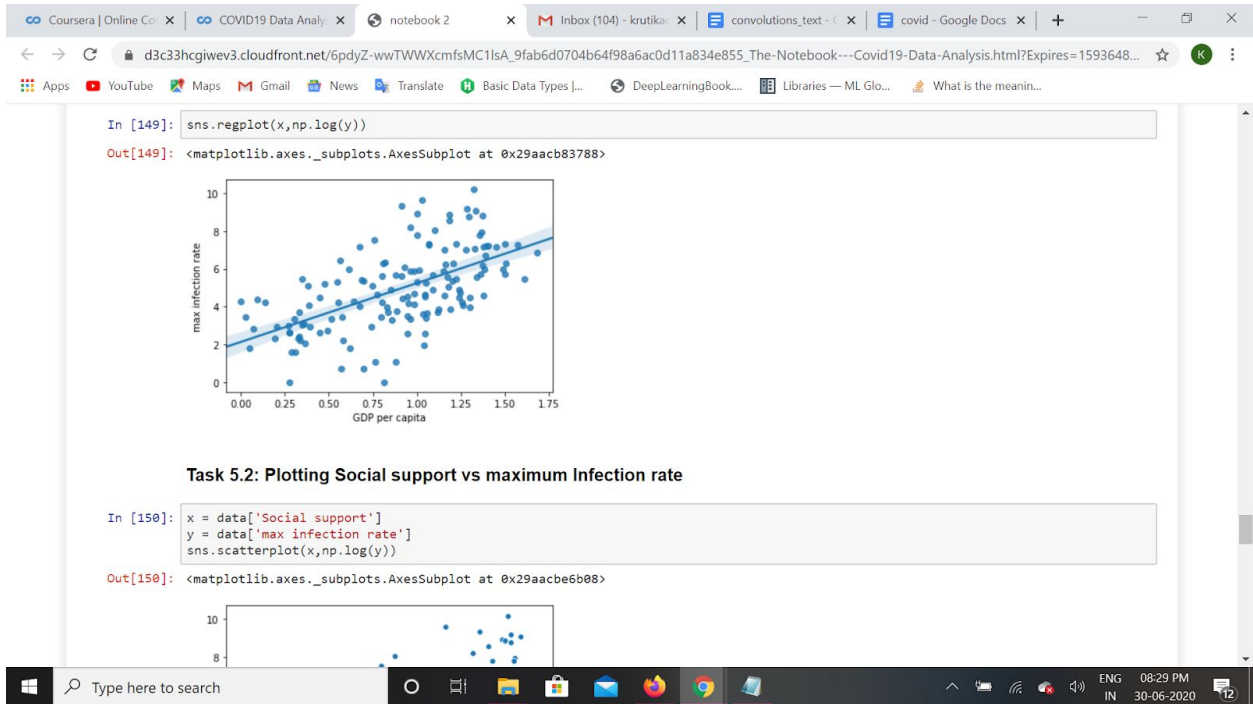
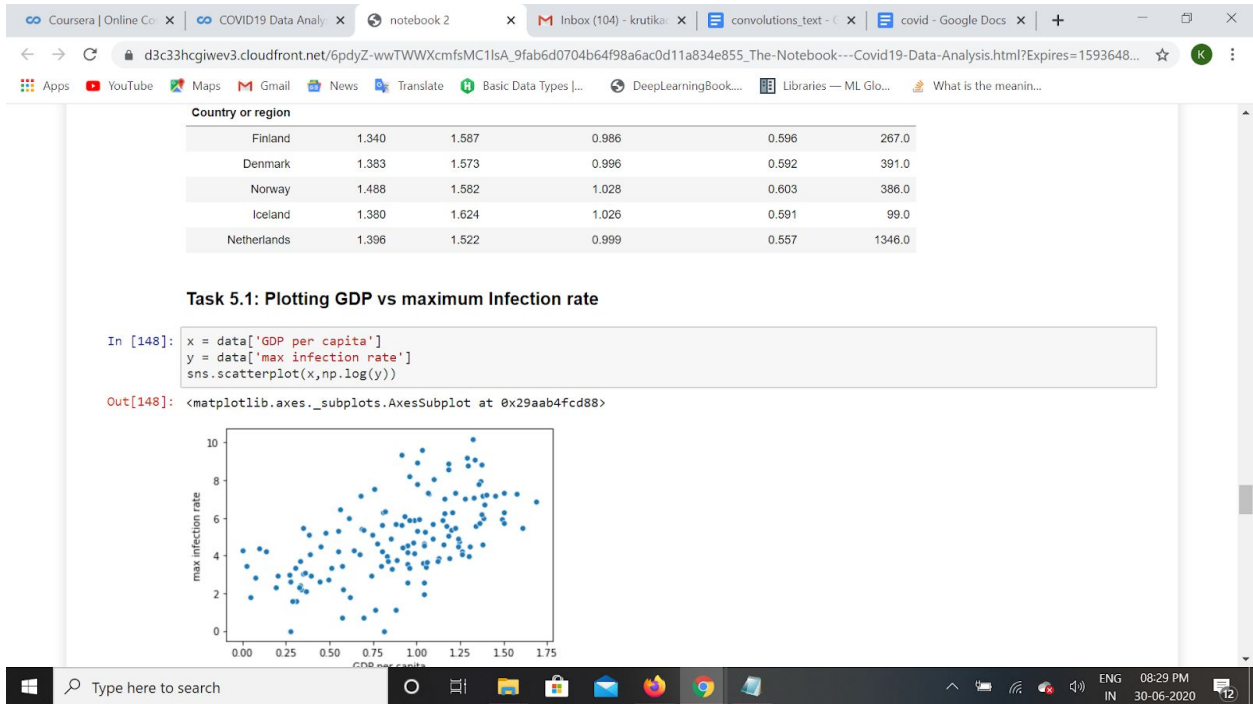
Task 5: Visualization of the results

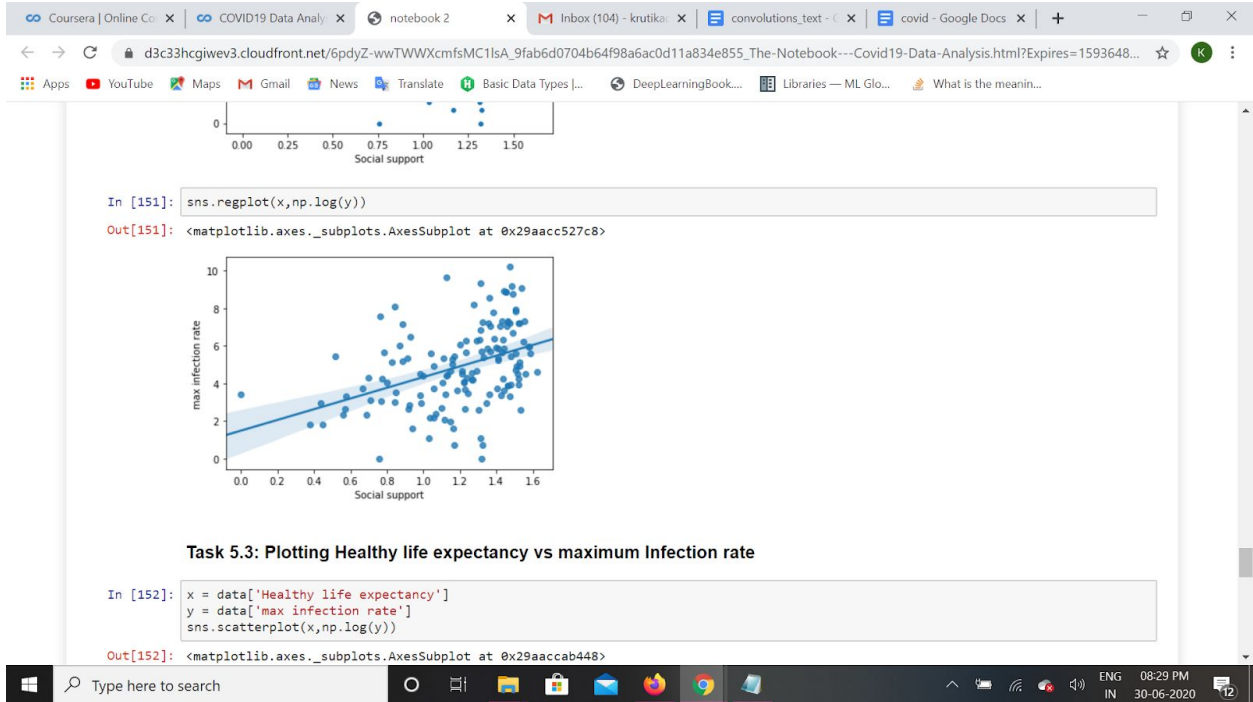
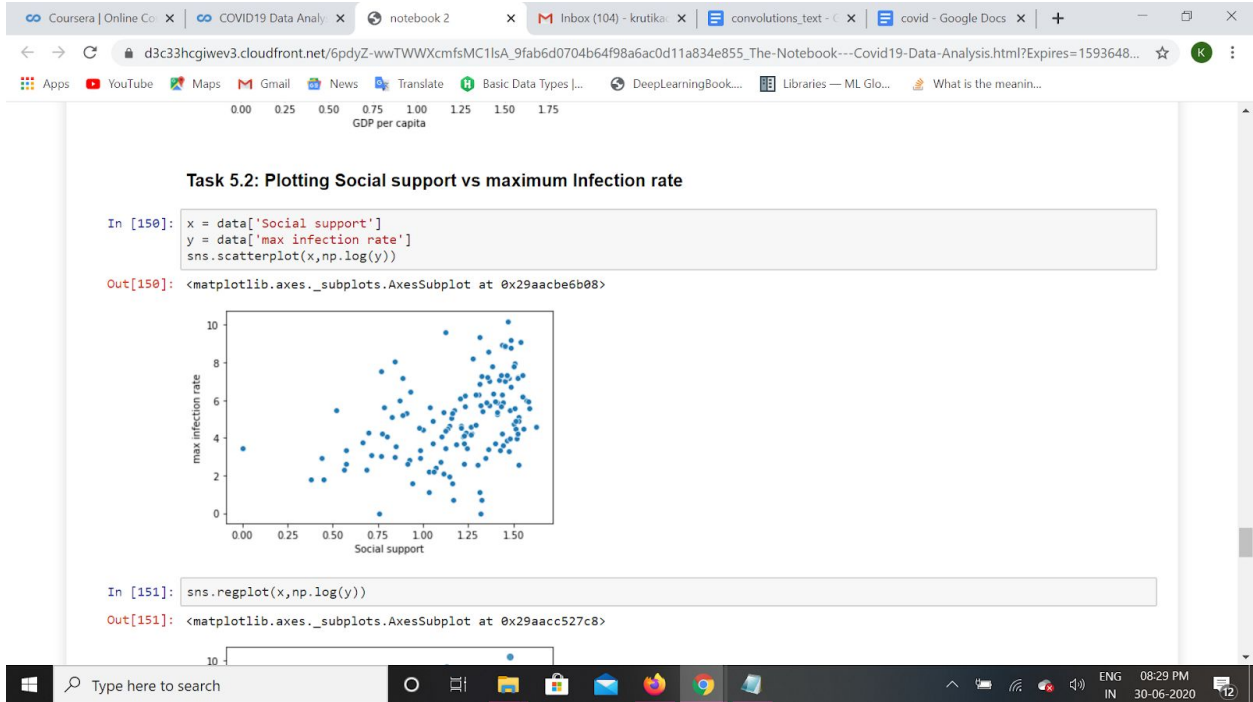
our Analysis is not finished unless we visualize the results in terms figures and graphs so that everyone can understand what you get out of our analysis

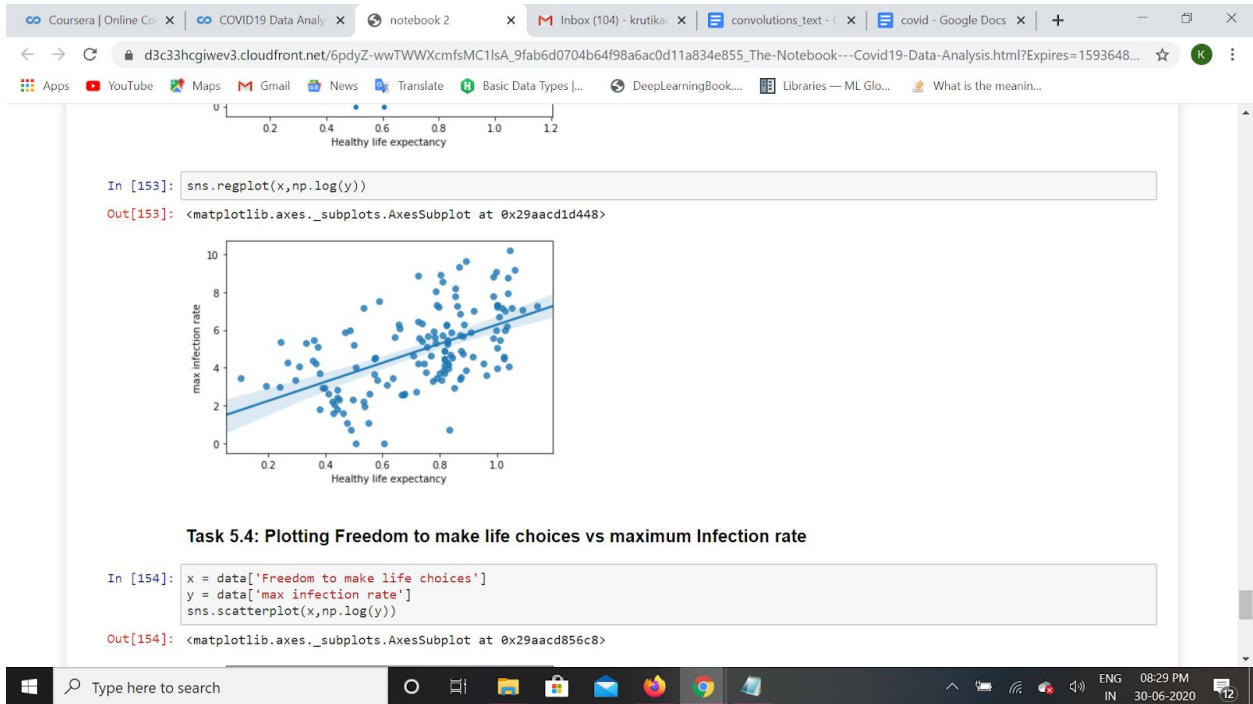
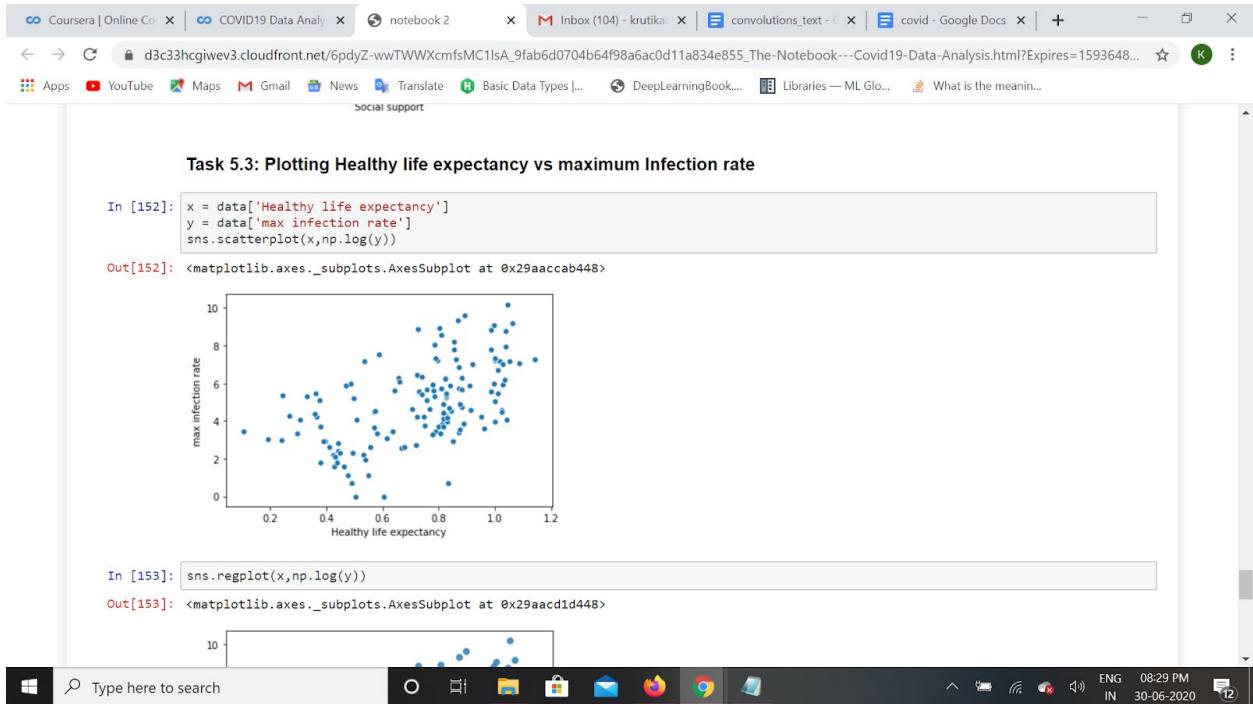
In [147]: `data.head()`

Out[147]:

	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	max infection rate
Country or region					
Finland	1.340	1.587	0.986	0.596	267.0
Denmark	1.383	1.573	0.996	0.592	391.0



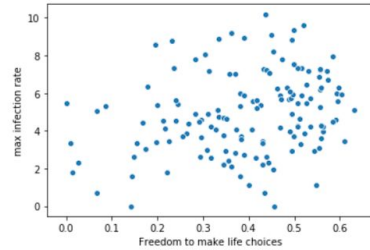




Task 5.4: Plotting Freedom to make life choices vs maximum Infection rate

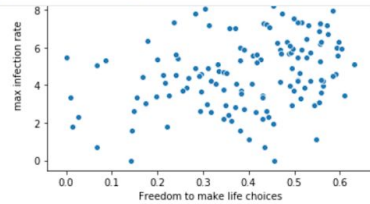
```
In [154]: x = data['Freedom to make life choices']  
y = data['max infection rate']  
sns.scatterplot(x,np.log(y))
```

```
Out[154]: <matplotlib.axes._subplots.AxesSubplot at 0x29aacd856c8>
```



```
In [156]: sns.regplot(x,np.log(y))
```

```
Out[156]: <matplotlib.axes._subplots.AxesSubplot at 0x29aace5e748>
```



```
In [156]: sns.regplot(x,np.log(y))
```

```
Out[156]: <matplotlib.axes._subplots.AxesSubplot at 0x29aace5e748>
```

