

# **CH5019: Mathematical Foundations of Data Science**

## **Course Project**

**Submitted By: Group 35**

**EE17B154 - D Tony Fredrick  
EE17B156 - Dhruvjyoti Bagadthey  
EE17B113 - Om Shri Prasath  
EE17B141 - P Sai Venkat Kushal  
EE17B114 - Pruthvi Raj R G**

# Question 1: SVD-Based Face Recognition

## I. Introduction:

In this problem, we perform facial recognition by using representative images formed from a given set of images for each person. These representative images will be used to identify the person.

## II. Method:

### a) Model without Data Augmentation:

Given N (15) images for a person, we followed this approach to obtain the representative image:

1. We set aside one image as the test image for each person and did not use it anywhere in the process of forming the representative image, in order to be able to properly gauge the performance of our model on an unseen image.
2. Let  $x_i$  represent an image. We find the image  $\bar{x}$ , which is the “*mean image*” for a person as follows, the summation is over all the N-1 images:

$$\bar{x} = \frac{1}{N-1} \sum_i x_i$$

3. We then find the centered images ( $x_c$ ) by subtracting the mean image from each image, as shown:

$$x_c = x_i - \bar{x}$$

4. The centered images are matrices of shape 64X64. We convert them into column vectors of shape 4096X1 and stack the column vectors to create the matrix A.
5. We apply Singular Value Decomposition on matrix A using `numpy.linalg.svd` function. We compute a weighted average of the singular vectors using the singular values as weights to obtain an image( $x_r$ ) with higher weightage given to more important features. Here the importance of a feature is decided by the singular value corresponding to that feature.
6. The mean image  $\bar{x}$  is added to the image obtained above since originally we ‘*centered*’ images by subtracting the mean image  $\bar{x}$ .

$$x_{rep} = x_r + \bar{x}$$

7. We have performed these steps for each person and saved the representative images.
8. In order to measure the performance, we compute the L2 norm between the representative image  $x_{rep}$  and the image to be tested (say x) as follows:

$$\text{Distance} = || x_{rep} - x ||_2$$

9. When the distance is less than a predefined threshold (in our case, the threshold is 2550), we declare that the given image matches with the representative image from which its distance is the least among all the representative images. When the L2 norm distance exceeds the threshold, the image is said to belong to an unknown person who is not in our database.

## **b) Model with Data Augmentation:**

We will now try to improve the model using data augmentation. We will use N-1 training images for each person and augment them to get a larger training set. We will perform data augmentation by:

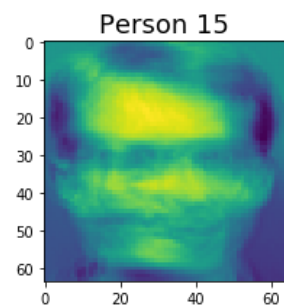
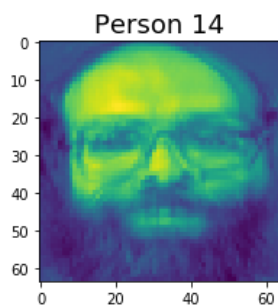
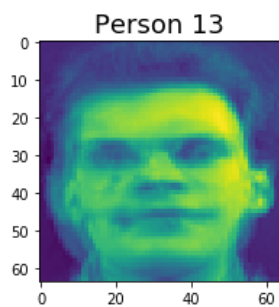
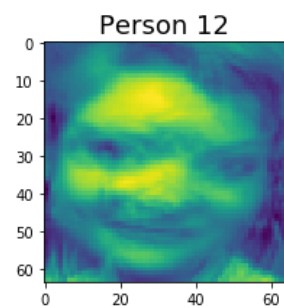
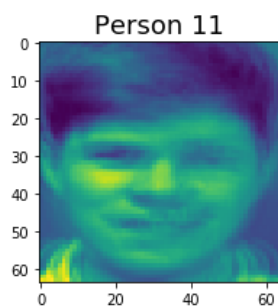
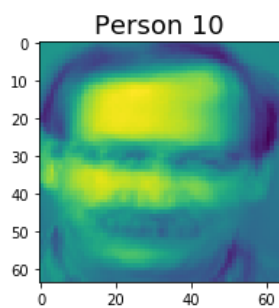
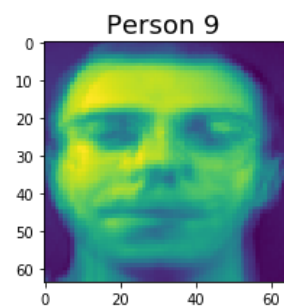
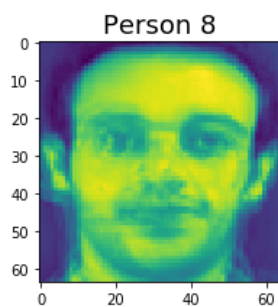
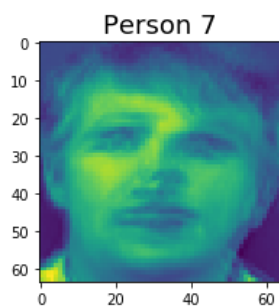
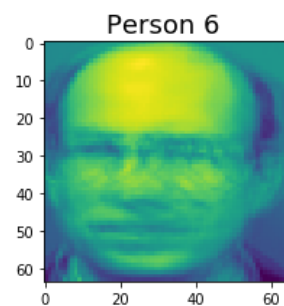
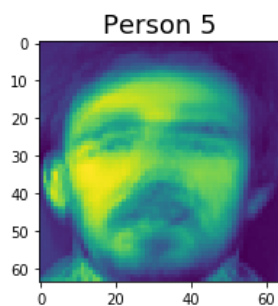
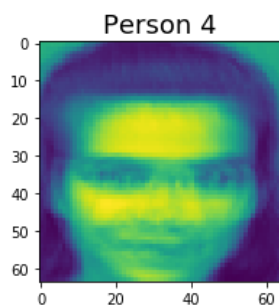
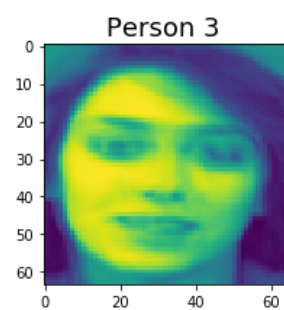
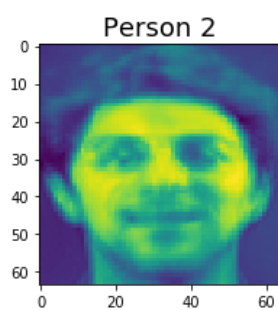
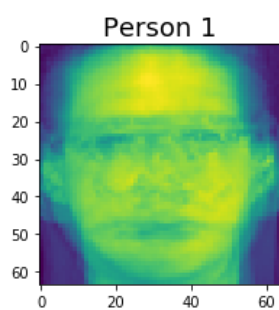
- Horizontal flip: The image is flipped on the horizontal axis.
- Horizontal width shift: The image is shifted slightly to the left and to the right.
- Change in brightness: The image is slightly dimmed and brightened.

The above methods give us 5 new images from 1 training image(so we have a total of 6 images), and we now have 54 images for each person. We perform all the 8 steps from the previous model without data augmentation on the expanded dataset and obtain a new set of representative images.

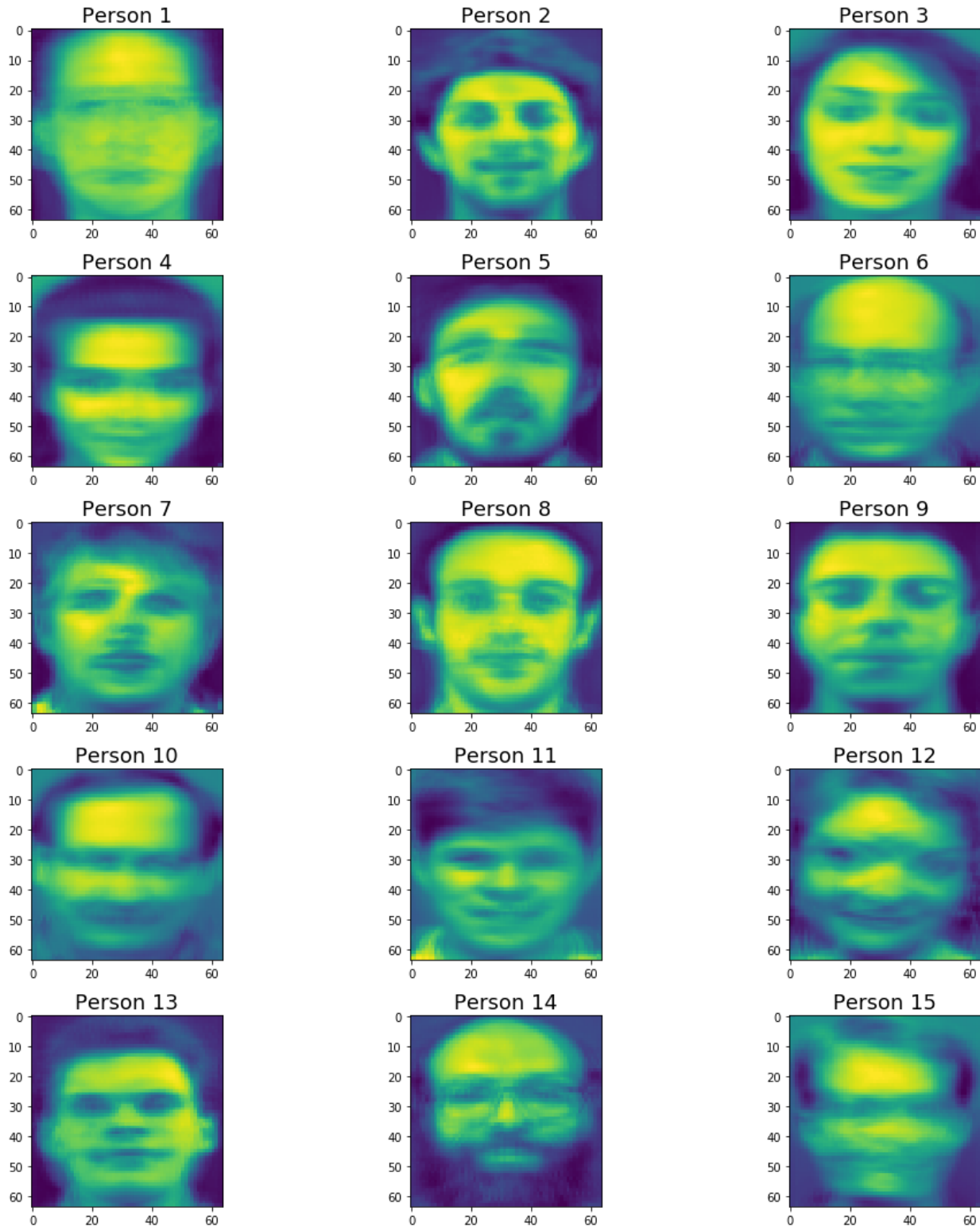
The notebook with the above methods implemented can be found [here](#) (in nbviewer format) and [here](#) (Google Colab).

**NOTE:** In both cases, we have removed one image of each person and created a test set consisting of 15 images in total(one for each person. We have not used these images anywhere during the formation of the representative image in either model. These images have not been used during data augmentation too. We only use them to test our models, to measure how well they perform on previously unseen images. “Test set” in our report and in the program, refers to this set of images. “Training set” refers to all the images used in the process of creating the representative images.

### III. Representative Images(without Data Augmentation)



#### IV. Representative Images with Data Augmentation



## V. Results and Conclusions:

The original dataset had 10 images for each class(person). We used 9 images from each class for training the SVD-based face recognition model. We used the remaining image from each class as our test set. Thus, the training set had 135 images and the test set had 15 images. We got these accuracies:

- Training accuracy = 99.259%
- Test accuracy = 86.667%

We observed that due to the small size of our dataset, **the model was overfitting the data**. This is what caused the **large difference between the training and test accuracy**.

**In order to combat overfitting, we used data augmentation.** We now had 810 training images. The performance with data augmentation is:

- Training accuracy = 90.247%
- Test accuracy = 93.333%

In both cases, the threshold used is 2550. When the L2 norm distance exceeds the threshold, the image is said to belong to an unknown person who is not in our database.

	Without Data Augmentation	With Data Augmentation
Training accuracy	99.259%	90.247%
Test accuracy	86.667%	93.333%

## VI. References

1. Guoliang Zeng, "Facial Recognition with Singular Value Decomposition", Motorola DSP Laboratory, Division of Computing Studies, Arizona State University Polytechnic Campus.
2. Matthew A. Turk, Pentland P. Alex(1991), "Face Recognition using Eigenface method", IEEE Conference on Computer Vision and Pattern Recognition, pp.586-591, 1991.

## Question 2: Logistic Regression Model-Based Classifier

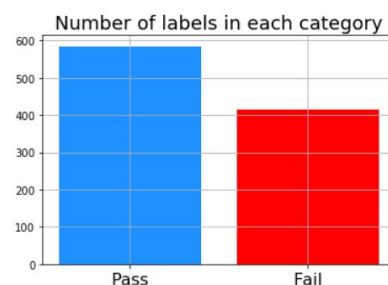
### I. Problem Statement:

The data set describes the operating conditions of a reactor and contains class labels about whether the reactor will operate or fail under those operating conditions. The problem is to construct a logistic regression model to predict the same. The data contains a 1000 X 6 data matrix. The first five columns are the operating conditions of the reactor. The sixth column provides the necessary annotation:

- Temperature: 400-700 K
- Pressure: 1-50 bar
- Feed Flow Rate: 50-200 kmol/hr
- Coolant Flow Rate: 1000-3600 L/hr
- Inlet Reactant Concentration: 0.1-0.5 mole fraction
- Test: fail/pass(Whether the reactor will operate or fail under the corresponding operating conditions)

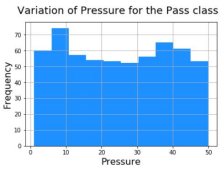
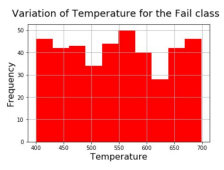
### II. Statistics of the Data:

Out of 1000 data points, 585(58.5%) were labeled Pass(positive samples) and 415(41.5%) were labeled fail(negative Samples).



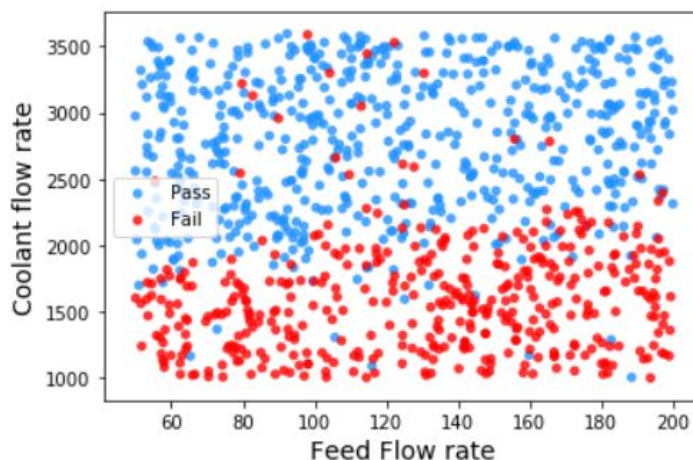
For each class(Pass/Fail) Histograms were plotted for each feature:

Statistics	Class	Temperature(K)	Pressure(bar)	Feed flow rate (kmol/hr)	Coolant flow rate (L/hr)	Inlet Reaction concentration
Mean	Pass	546.150291	24.906256	121.623419	2785.807744	0.303489
	Fail	547.634964	26.320747	129.829783	1605.060819	0.301568
Median	Pass	543.220000	24.710000	116.170000	2828.230000	0.313000
	Fail	547.380000	25.800000	134.650000	1557.140000	0.304400
Variance	Pass	7457.553255	202.473818	1934.354187	279073.388041	0.013339
	Fail	7647.468679	202.397734	1790.651119	194937.258644	0.013621

Minimum	Pass	400.630000	1.060000	50.030000	1006.650000	0.100300
	Fail	400.310000	1.190000	50.110000	1002.530000	0.104700
Maximum	Pass	699.870000	49.890000	199.960000	3595.620000	0.499600
	Fail	698.880000	49.790000	199.320000	3587.690000	0.498800
Histogram	Pass					
	Fail					

Out of the five features given, feature-4 'Coolant Flow Rate' has a distribution that is markedly different for the pass and fail examples. Next, the variances of the 'Feed flow rate' are also different.

The data points on the coolant flow rate - feed flow rate plane are plotted below:



A significant accuracy can be achieved by simply drawing a straight line and classifying all points on the upper half-plane as pass and rest as fail. In the next section, we see that the trained classifier does exactly that.



### III. Model

- The DataSet was partitioned into train\_set and test\_set by using train\_test\_split of sklearn.model\_selection with a test size of 30%.
- For Preprocessing, the sklearn.preprocessing.StandardScaler was used for feature-scaling.

$$X_{norm} = \frac{X - \mu}{\sigma}$$

- **The Logistic Loss function:**

$$\frac{1}{m} \cdot \sum_{i=1}^m \log(1 + e^{-y_i a^T x_{norm_i}})$$

Where the decision boundary is  $a^T x = 0$ ,  $m$  = no. of training samples, 700 in our case.  $y_i$ s are the corresponding boolean labels (1 for Pass and 0 for fail).

The classifier must 'Learn'  $\bar{a}$  to minimize the logistic loss.

- **Optimization objective:**

$$\min_a \frac{1}{m} \cdot \sum_{i=1}^m \log(1 + e^{-y_i a^T x_{norm_i}})$$

- **Algorithm Gradient Descent:**

⇒ randomly initialize  $\bar{a}$  with a random number of the order 10e-5

⇒ Loop num\_iterations times:

$$\Rightarrow \bar{h} = \frac{1}{1 + e^{-X_{norm} \cdot \bar{a}}} \quad \# \text{ h is called hypothesis function}$$

⇒ for i in number of features (in our case 0 to 5):

$$\Rightarrow a_i := a_i - \frac{\alpha}{m} X_{norm_i}' \cdot (\bar{h} - \bar{y})$$

⇒ end

⇒ end

⇒ return  $\bar{a}$

Where  $\alpha$  is the learning\_rate = 0.1

and num\_iterations is taken to be 15000

- To avoid the algorithm getting stuck at some local optima, We randomly initialize  $\bar{a}$ , 10 times, and pick the  $\bar{a}$  which gives minimum loss.

- **Algorithm Logistic Regression:**

⇒ Append a column of ones to  $X_{norm}$

⇒ for i in 1 to 10:

⇒  $\bar{a} = \text{Gradient\_Descent}(X_{norm}, \bar{y}, m, \text{num\_iterations}, \alpha)$

$$\Rightarrow \text{Loss} = \frac{1}{m} \cdot \sum_{i=1}^m \log(1 + e^{-y_i a^T x_{norm_i}})$$

⇒ choose min\_Loss = least Loss ;  $\overline{a_{min}}$  = corresponding  $\bar{a}$

⇒ return  $\overline{a_{min}}$

$$y_{predicted} = \mathbb{I} \left[ \frac{1}{1 + e^{-X_{testnorm} \cdot \overline{a_{min}}}} > 0.5 \right]$$

#### IV. Results and Conclusions

The logistic Regression algorithm was implemented by scratch and as a test for correctness, sklearn.linear\_model.LogisticRegression was also used separately. The notebook and the code can be found [here](#).

- The Test set accuracy for both the classifiers was 93%
- The confusion matrix(on the test set) obtained was:

Predicted → Actual ↓	Fail	Pass
Fail	106	7
Pass	14	173

- The F1-score obtained is **0.9427**
- Visualizing the separating Hyper-plane on a 2d plot :

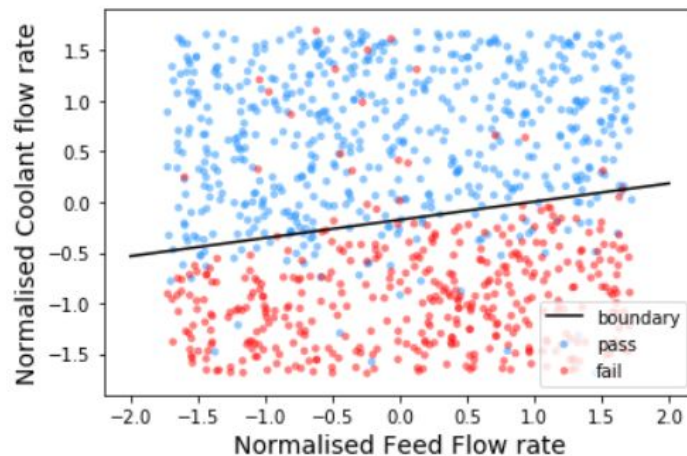
$$\overline{a_{min}} = [1.04618 \quad -0.60667 \quad -0.66327 \quad -1.08025 \quad 6.00661 \quad -0.18957]^T$$

From the values of  $\overline{a_{min}}$ , the weight assigned to  $X_{\text{coolant flow rate}}$  is much larger than the rest, this implies that the coolant flow rate is the feature over which the classification depends on the most. The second highest contender in this list is the Feed flow rate

Considering only feed flow and coolant flow rate, The decision boundary obtained for the 2d plot shown above is

$$X_{\text{Coolant flow rate}} * 6.006 - X_{\text{Feed flow rate}} * 1.08 + 1.046 = 0$$

This boundary is plotted below:



## Question 3: COVID19 Data Analysis

### 1) Introduction :

In this part, we are going to analyze the data of COVID-19 in the context of India with the COVID-19 related data available in Kaggle. The data is available [here](#). The notebook containing the code and the graphs is available [here](#).

### 2) Answers for the posed questions :

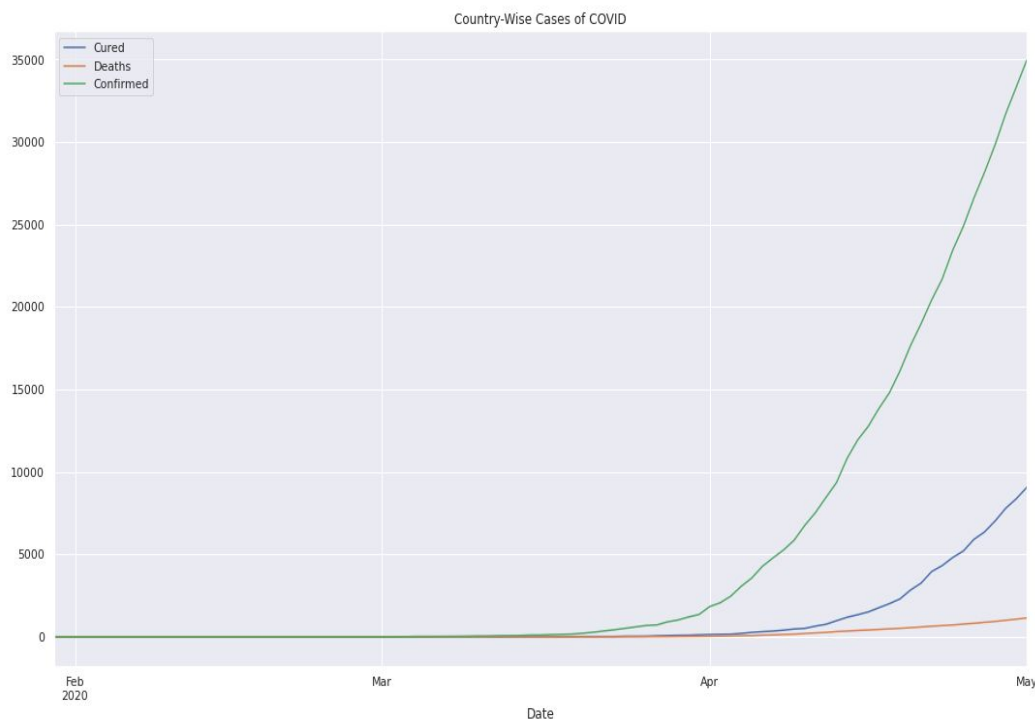
Q1) Which age group is the most infected?

A) From the data in the file AgeGroupDetails.csv, it can be concluded that the age group between **20-29 is most affected**. This result is as of 10/05/2020.

Q2) Plot graphs of the cases observed, recovered, deaths per day country-wise, and statewise.

A) The plot for the country is shown below but since displaying plots for 29 states here is not feasible, they are present in the notebook attached.

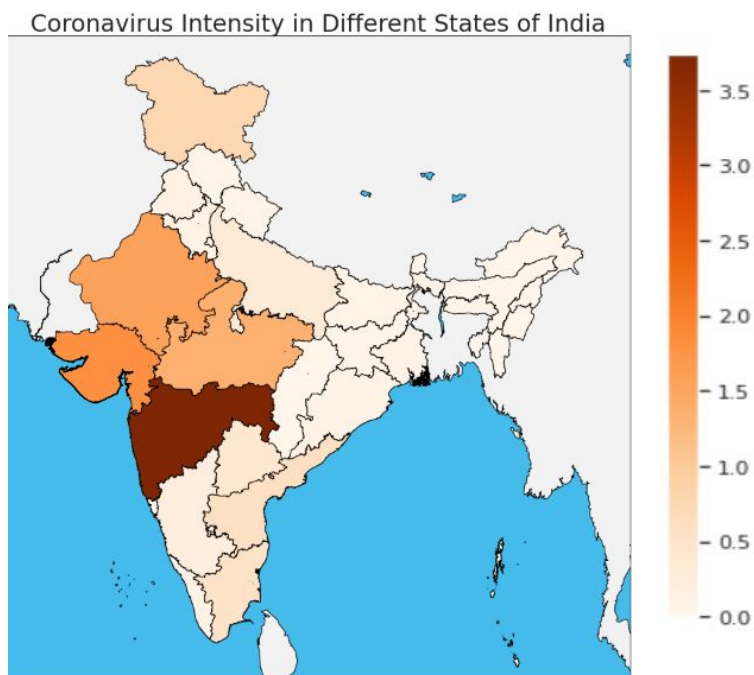
#### The number of cases in India from in the months from February to May



Q3) Identify the positive cases on a state level. Quantify the intensity of the virus spread for each state. (\*Intensity here means the number of positive cases/population density.)

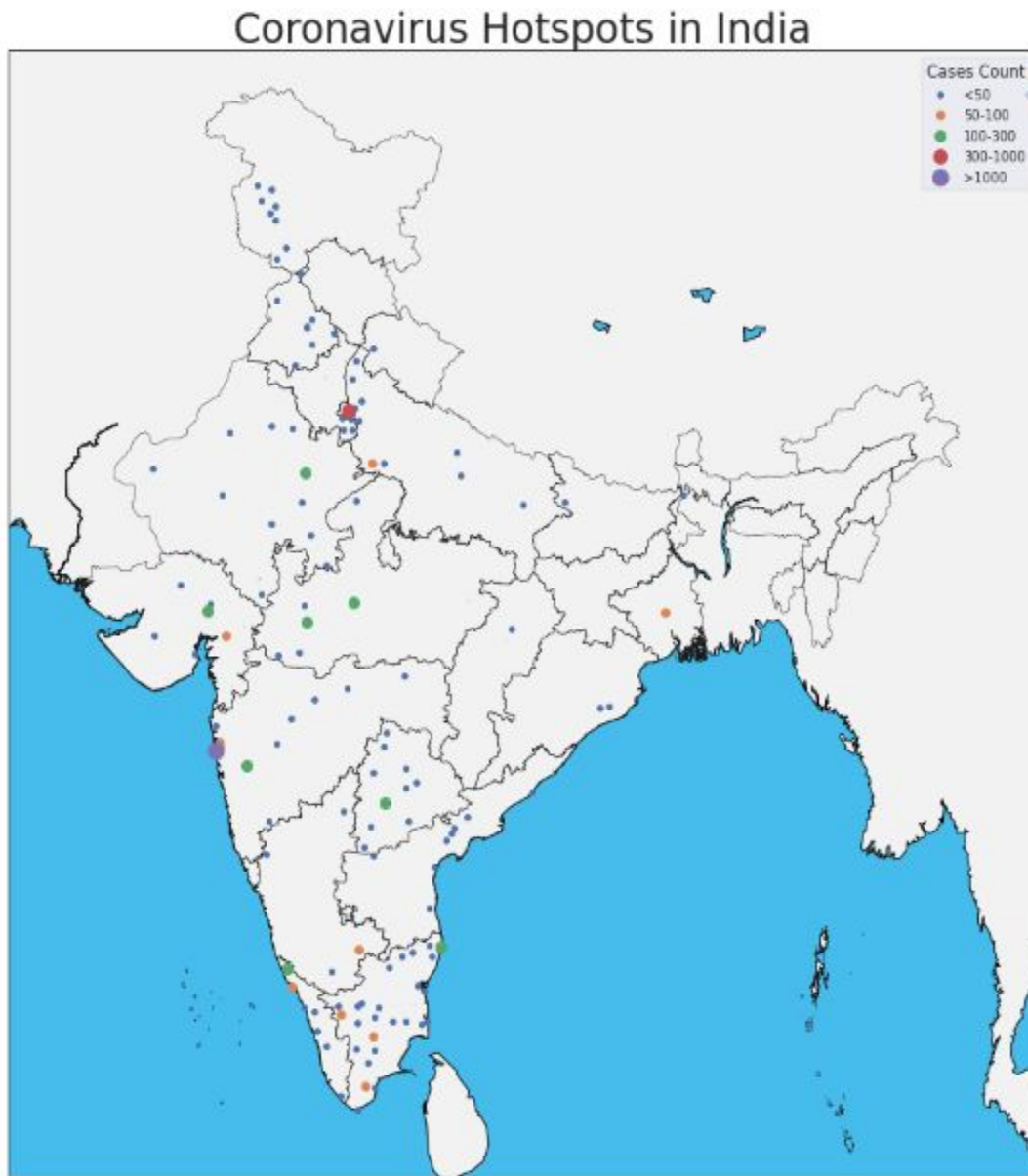
A) Intensity is calculated until the date of 10 / 04 / 20

State	Intensity	State	Intensity
Andaman and Nicobar Islands	0.239	Ladakh	5.357
Andhra Pradesh	1.198	Madhya Pradesh	1.097
Arunachal Pradesh	0.059	Maharashtra	3.737
Assam	0.073	Manipur	0.016
Bihar	0.054	Mizoram	0.019
Chandigarh	0.002	Odisha	0.164
Chhattisgarh	0.053	Puducherry	0.002
Delhi	0.079	Punjab	0.24
Goa	0.018	Rajasthan	2.303
Gujarat	0.782	Tamil Nadu	1.503
Haryana	0.295	Telangana	1.516
Himachal Pradesh	0.228	Tripura	0.003
Jammu and Kashmir	1.878	Uttar Pradesh	0.521
Jharkhand	0.031	Uttarakhand	0.185
Karnataka	0.618	West Bengal	0.113
Kerala	0.416		



Q4) List places in the country which are active hotspots/clusters as on 10.04.2020.\* (A hotspot is defined as an area in a city where 10 or more people have been tested positive.)

A) The list of places which are declared as hotspots are shown in the India map below  
(The data in list format is in the notebook attached)



Q5) Which states have the maximum change (consider increase and decrease separately) in the number of hotspots on a weekly basis from 20.03.2020 to 10.04.2020 (3 weeks).

A) **Assumption** For the calculation of the hotspots for each, we took the number of positive test cases appeared in the week of interest in order to determine whether a particular area is a hotspot or not, though the duration considered here for changing the status of a previously existing hotspot as 'not a hotspot' is small, we believe that for the duration asked to consider in the question, this would be the optimal way for determining the hotspots with given data.

**States with Maximum Increase during the interested week duration**

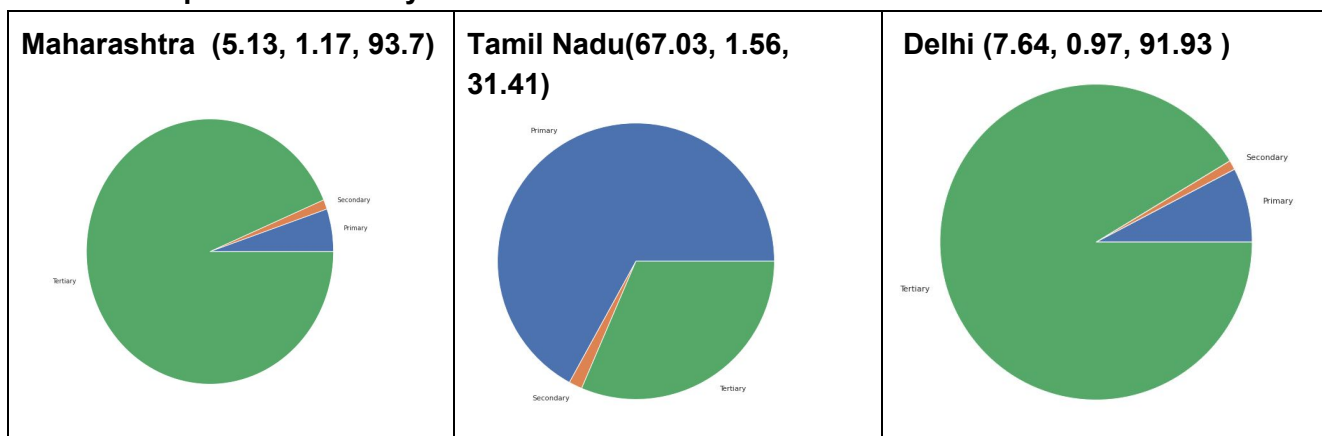
Week duration	State	Amount increase
20/03/20-27/03/20	Kerala	3
27/03/20-03/04/20	Tamil Nadu	12
03/04/20-10/04/20	Telangana	10

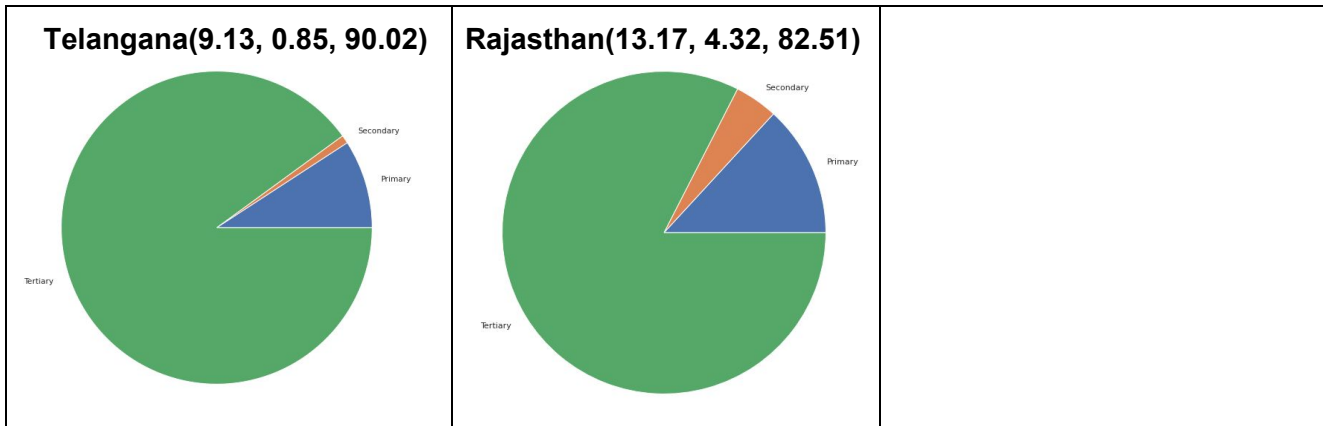
**States with Maximum decrease during the interested week duration**

Week duration	State	Amount decrease
20/03/20-27/03/20	Haryana,Telangana,Uttar Pradesh	1
27/03/20-03/04/20	Kerala	1
03/04/20-10/04/20	Andhra Pradesh	2

Q6) For the given data, identify cases with international travel history (primary case), personal contact with the primary case (secondary case). Cases that do not fall in the primary and secondary fall into the tertiary case. Quantify them based on the percentage for the top 5 states with maximum cases till 10.04.2020

A) **Note:** The ordered pair beside the states are in the order (Primary%, Secondary%, Tertiary%) in the pie chart Blue represents Primary, Orange represents Secondary, and Green represents Tertiary





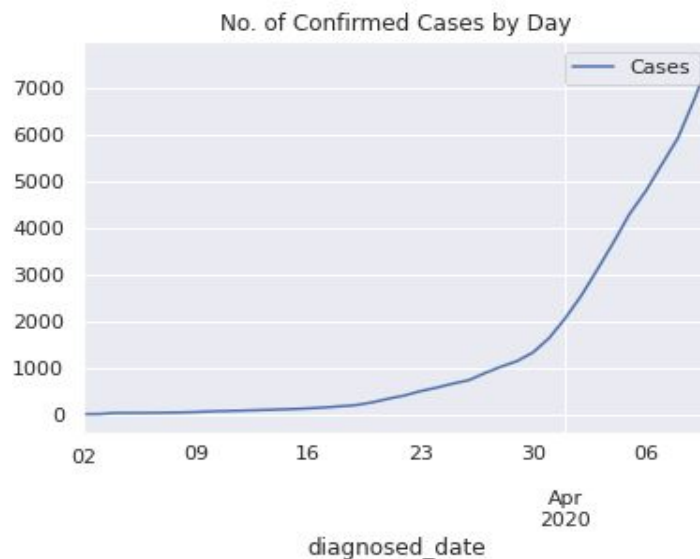
Q7) Find out the number of additional labs needed from the current existing labs (assume 100 tests per day per lab) with an increase in the rate of 10% cases per day from 11.04.2020 - 20.04.2020. List out any further assumptions considered.

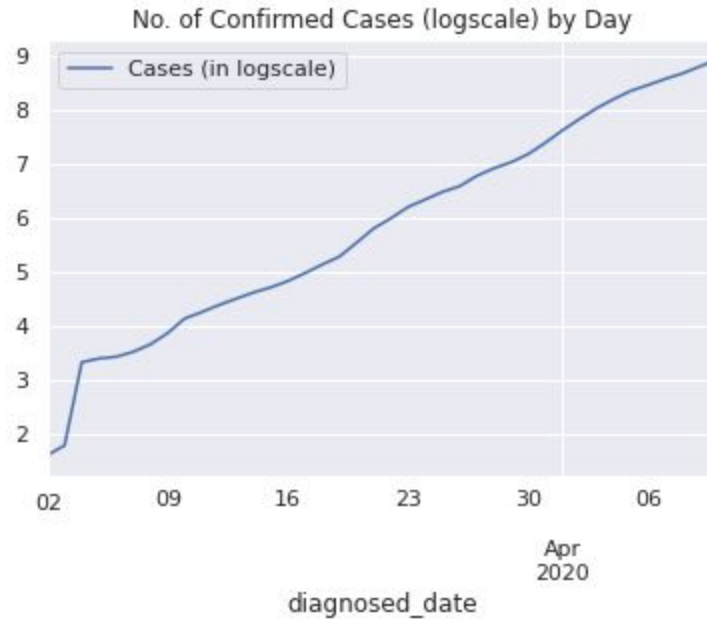
A) Assuming that the maximum number of tests a lab can perform per day remains fixed at 100 tests even when in the future dates the number extra required based on data from .csv files **ICMRTestingLabs.csv** and **ICMRTestingDetails.csv** is **3918**.

(Here we have assumed that it is asked the number of additional labs required after completion of date 20/04/20 and the starting initial value is from 10/04/20 and compounded from 11/04/20)

Q8) Plot the number of cases starting from 1st March - 10th April. Based on this plot can you comment on the popular notion of 'flattening the curve'.

A) From the data from the .csv file IndividualDetails.csv the plotting, the number of cases across dates gives



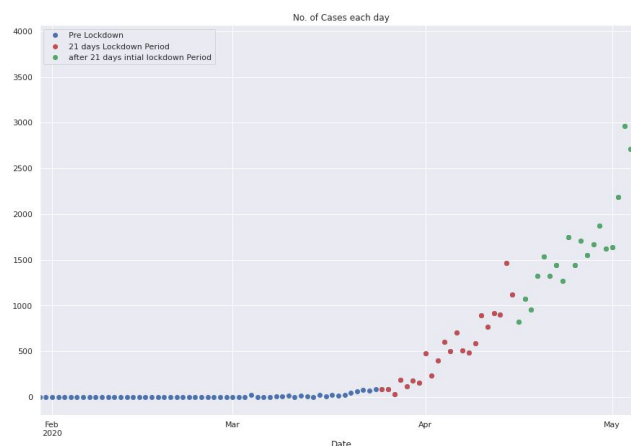


Since COVID-19 is a highly contagious virus, the rate of increase in the number of cases is a preferable parameter for judging the curve compared to judging it from the total number of cases. The above graph shows the number of COVID-19 cases visually, which is increasing at an almost exponential rate. The aim of flattening the curve is to reduce this rate of increase of cases, by making the rate of growth in cases nearly zero. By making the rate of increase zero, we can ensure that the increase in COVID 19 cases can be stopped, and treat each of the individual cases effectively.

Q9) As we know, social distancing is the best option to avoid the spread. Based on the time series data (covid\_19\_india.csv), can you suggest how successful the 21 days lockdown has been?

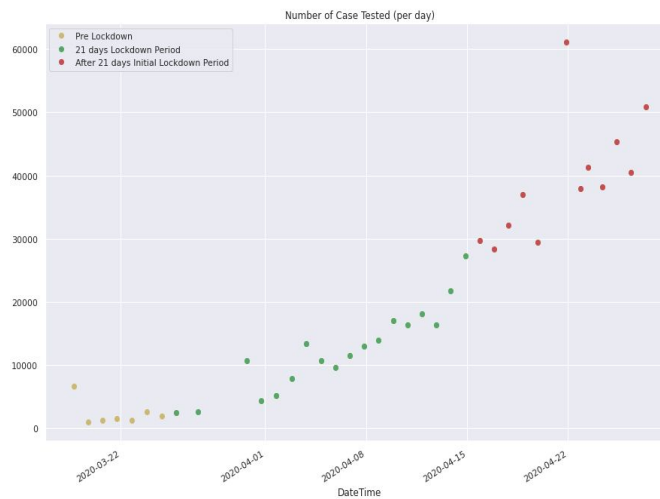
A) The data is plotted using data from **covid\_19\_india.csv** for the first plot, and for the second plot, **ICMRTTestingLabs.csv** and **ICMRTTestingDetails.csv** are used. Even though the question was about the initial lockdown period of 21 days since we had the data for before and after the 21-day lockdown we thought it will give a better insight into the situation if we consider the days outside of the duration interested in the question also.

**Figure 1 Total number of cases per day from the months February to May**

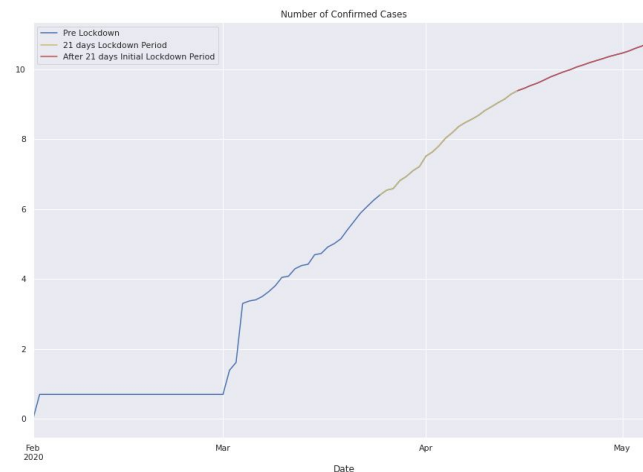




**Figure 2 Total number of cases tested per day from the months February to May**



**Figure 3 Number of confirmed cases of COVID19 in log scale**



After a thorough analysis of the data, visualized in the graphs shown above, we concluded that the lockdown has been partially successful. Figure 3 above shows the variation in the rate of increase in the number of Covid-19 cases in the log scale. In the portion following the 21-day lockdown period, the curve is flatter than the expected linear slope (obtained by extrapolating the data), showing that the 21-day lockdown has been partly successful.

In figures 1 and 2, we plotted the number of cases showing an increase in the number of cases every day, but when augmented with the information from figure 3, it can be concluded that the rate of increase in cases is decreasing. A multitude of reasons contribute towards an increase in the number of cases; one of them is the growth of the number of Covid-19 tests per day. From the given data though not possible to conclusively decide whether the lockdown has been definitively successful, it can be said to be partially successful, and following measures like social distancing and maintaining personal hygiene will go a long way in helping us combat this deadly virus.