# Report and Results

Neeladri Das {18CS10026}
Avijit Mandal {18CS30010}

## 1.Introduction

In the data set Train_F.csv , we are provided with **countyname**, **countyfips**, **state_name** and **predicted_deaths** from october 6 to 12 and **severity_count**.

Here we have to predict the **severity_count** (1 or 2 or 3) using a naive bayes classifier and assuming each feature is independent of the other.

The column **countyname** and **countyfips** are mostly **unique** for all the samples, so it won't give much details about training. So, we are **dropping** it in the first place. Then we are encoding state_name from categorical to numerical value using Label Encoder

## 2.Algorithm Description

So, now we have a 2D matrix with all numerical values and we proceed for applying gaussian normal density function to calculate the probability

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$$

f(x) = the probability density
σ = standard deviation
μ = Mean of that feature

**Bayes theorem** states that,

$$P(A|B) = \frac{P(B|A)*P(A)}{P(B)}$$

P(A|B): Posterior Probability
P(A) : Prior Probability
P(B|A): Likelihood
P(B) : Evidence
So, we get,

**Posterior = Likelihood*Prior/Evidence**
From the class and data point of view we say that

**P(Class|Data) = (P(Data|Class)*P(Class))/P(Data)**

So, for a instance <**x1,x2,...xn**>, probability of particular class **c** is
**P(c|x1,x2,..xn) = P(x1|c)*P(x2|c)*...*P(xn|c)*P(c)**

**Note**: *We drop the denominator (the probability of observing the data in this instance) as it is a constant for all calculations.*

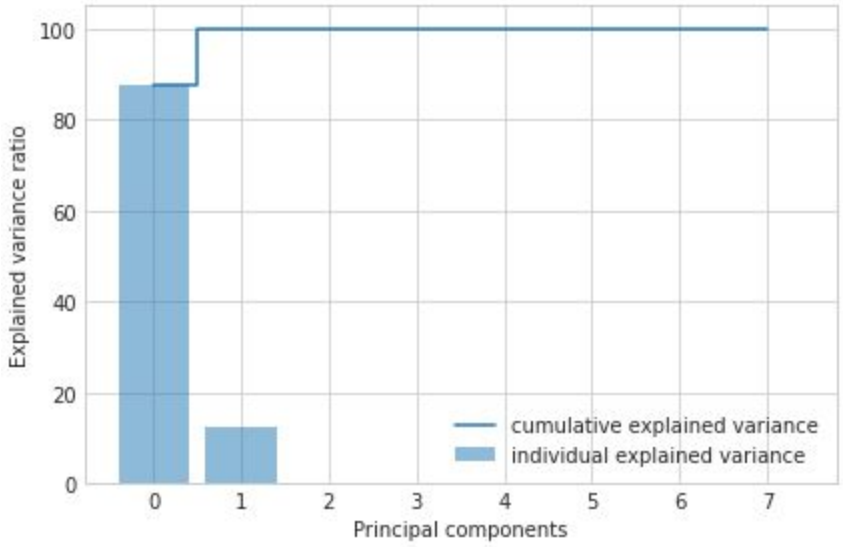
So, to predict the severity_count for a particular instance we will calculate for c=1 , c=2, c=3 and return prediction for which Maximum probability is received.

Here P(x1|c) is calculated by gaussian density function f(x) mentioned above.

**3.Results:**
With 5-fold cross validation we obtained a **accuracy 46.8%**

**Principal Component Analysis:**

Here we calculated the explained variance using **eigen values** of standardised **covariance matrix.** The graph obtained between **Explained variance** artion and **Principal Components** is as follows:



The plot above clearly shows that most of the variance (**87.51%** of the variance to be precise) can be explained by the first principal component alone. The second principal component still bears some information (**12.48%**) while the rest principal components can safely be dropped without losing too much information. Together, the first two principal components contain **99.99%** of the information.

With 5-fold cross validation after PCA transformation we obtained a **accuracy 45.5%**

**Sequential Backward Selection**

With 5-Fold cross validation with Sequential Backward Selection we obtained the results as follows

```
arrSBS = KFoldCrossValSBS(X_std, y)
print("\nMean Accuracy is :" + str(np.mean(arrSBS)))
```

```
feature Set obtained and accuracy: [0 1 2 3 4 5 6] 0.6096774193548387
feature Set obtained and accuracy: [0 1 2 3 4 5 6] 0.4403225806451613
feature Set obtained and accuracy: [0 1 2 3 4 6 7] 0.3
feature Set obtained and accuracy: [0 1 2 3 4 5 6] 0.1935483870967742
feature Set obtained and accuracy: [1 2 3 4 5 6 7] 0.8419354838709677

Mean Accuracy is :0.4829032258064515
```