Probabilistic Model in Machine Learning

- A Probabilistic model in machine learning is a mathematical representation of a real-world process that incorporates uncertain or random variables.
- The goal of probabilistic modelling is to estimate the probabilities of the possible outcomes of a system based on data or prior knowledge.
- Probabilistic models are used in a variety of machine learning tasks such as classification, regression, clustering, and dimensionality reduction.

Some popular probabilistic models include:

- Gaussian Mixture Models (GMMs)
- Hidden Markov Models (HMMs)
- Bayesian Networks
- Markov Random Fields (MRFs)

Probabilistic models allow for the expression of uncertainty, making them particularly well-suited for real-world applications where data is often noisy or incomplete. Additionally, these models can often be updated as new data becomes available, which is useful in many dynamic and evolving systems.

For better understanding, we will implement the probabilistic model on the OSIC Pulmonary Fibrosis problem on the kaggle.

Problem Statement: "In this competition, you'll predict a patient's severity of decline in lung function based on a CT scan of their lungs. You'll determine lung function based on output from a spirometer, which measures the volume of air inhaled and exhaled.

The challenge is to use machine learning techniques to make a prediction with the image, metadata, and baseline FVC as input."

Importing Libraries

```
1. import pandas as pd
2. import numpy as np
```

3. import seaborn as sns

4. import matplotlib.pyplot as plt

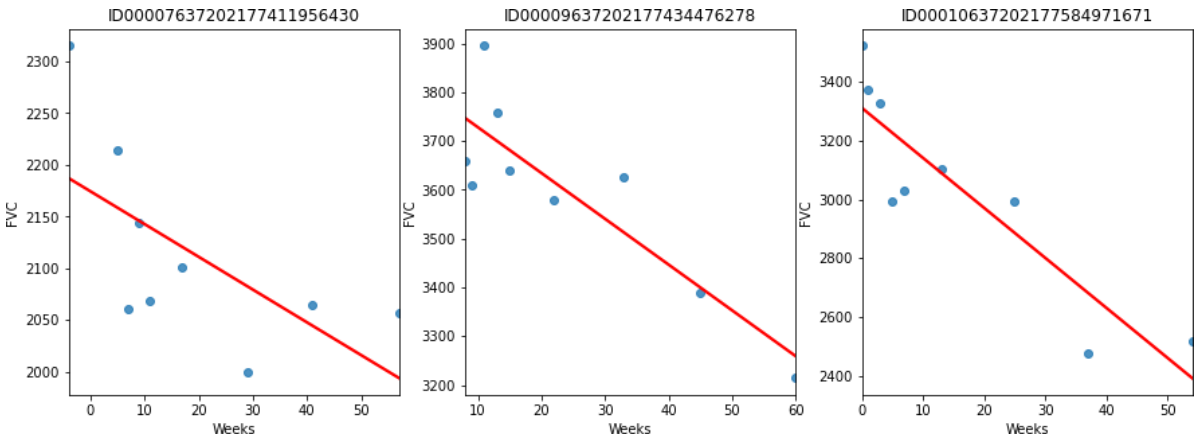
EDA

1. train_Datafame = pd.read_csv('train.csv')
2. test_Dataframe = pd.read_csv('test.csv')

Let's see this decline in lung function for three different patients.

```
1.  def chart_builder(patient_id, ax):
2.      d = train_Datafame[train_Datafame['Patient'] == patient_id]
3.      x = d['Weeks']
4.      y = d['FVC']
5.      ax.set_title(patient_id)
6.      ax = sns.regplot(x, y, ax=ax, ci=None, line_kws={'color':'red'})
7.
8.
9.  f, axes = plt.subplots(1, 3, figsize=(15, 5))
10. chart_builder('ID00007637202177411956430', axes[0])
11. chart_builder('ID00009637202177434476278', axes[1])
12. chart_builder('ID00010637202177584971671', axes[2])
```

Output:



$$FVC_{ij} \sim \mathcal{N}(\alpha_i + j\beta_i, \sigma_i)$$
$$\sigma_i \sim |\mathcal{N}(0, 200)|$$
$$\alpha_i \sim \mathcal{N}(FVC_i^b + w_i^b \beta^{int}, \sigma^{int})$$
$$\beta_i \sim \mathcal{N}(\alpha^s + A_i \beta_c^s, \sigma^s)$$
$$\beta^{int} \sim \mathcal{N}(0, 100)$$
$$\sigma^{int} \sim |\mathcal{N}(0, 100)|$$
$$\beta_c^s \sim \mathcal{N}(0, 100)$$
$$\alpha^s \sim \mathcal{N}(0, 100)$$
$$\sigma^s \sim |\mathcal{N}(0, 100)|$$