# AssignmentNo. 4

## ProblemStatement:

Use Autoencoder to implement anomaly detection. Build the model by using:

1. Import required libraries
2. Upload / access the dataset
3. Encoder converts it into latent representation
4. Decoder networks convert it back to the original input
5. Compile the models with Optimizer, Loss, and Evaluation Metrics

**Objective:**

1. To be able to apply deep learning algorithms to solve problems of moderate complexity
2. Use Autoencoder to implement anomaly detection.

**Outcomes:**

At the end of the assignment the students should able-

1. Apply Deep Learning techniques like Auto encoders to solve real word Problems.
2. CompilethemodelswithOptimizer,Loss,andEvaluationMetrics
3. Evaluate the performance of the model build using Deep Learning.

**Solution Expected**

AutoEncoders are widely used in anomaly detection. The reconstruction errors are used as the anomaly scores. Let us look at how we can use AutoEncoder for anomaly detection using TensorFlow. Import the required libraries and load the data. Here we are using the ECG data which consists of labels 0 and 1. Label 0 denotes the observation as an anomaly and label 1 denotes the observation as normal.

**Methodology to be used**

* Deep learning
* Autoencoders

**Theory:**

AutoEncoder is a generative unsupervised deep learning algorithm used for reconstructing high-dimensional input data using a neural network with a narrow bottleneck layer in the middle which contains the latent representation of the input data**.** An autoencoder is a type of artificial neural network used to learn data encodings in an unsupervised manner. The aim of an autoencoder is to learn a lower-dimensional representation (encoding) for a higher-dimensional data, typically for dimensionality reduction, by training the network to capture the most important parts of the input image.

**Conclusion:**

Autoencoders can be used as an anomaly detection algorithm when we have an unbalanced dataset where we have a lot of good examples and only a few anomalies. Autoencoders are trained to minimise reconstruction error. When we train the autoencoders on normal data or good data, we can hypothesise that the anomalies will have higher reconstruction errors than the good or normal data.