HELP MANUAL

DDCN-2019@CSED, MNNIT Allahabad

Neural Networks in Python

Multi Layer Perceptron Classifier

Neural Network using MLP

- Required packages
  - from sklearn.datasets import load_breast_cancer
  - from sklearn.model_selection import train_test_split
  - from sklearn.preprocessing import StandardScaler
  - from sklearn.neural_network import MLPClassifier
  - from sklearn.metrics import classification_report,confusion_matrix

Neural Network using MLP Contd…

- Load dataset
  - Eg:cancer = load_breast_cancer()
- Creating Feature set and Target set from Data set
  - Eg: X = cancer['data']
  - y = cancer['target']
- Split Feature set and Target Set into Training set and Target Set.
  - X_train, X_test, y_train, y_test = train_test_split(X, y)

Neural Network using MLP Contd…

- Standardizing features by removing the mean and scaling to unit variance
  - scaler = StandardScaler()
- Fitting scaled data only to the training data
  - scaler.fit(X_train)
- Transforming Training and Test data of feature set
  - X_train = scaler.transform(X_train)
  - X_test = scaler.transform(X_test)

Neural Network using MLP Contd…

- Instantiate MLPClassifer with specific hidden layer size
  - MLPClassifier(hidden_layer_size)
  - mlp = MLPClassifier(hidden_layer_sizes=(30,30,30))
- Fit MLP Classifier on Training Data of Feature set.
  - mlp.fit(X_train,y_train)
- Perform Prediction on Testset data of feature set.
  - predictions = mlp.predict(X_test)

Neural Network using MLP Contd…

- Print “Confusion Matrix”
  - print (confusion_matrix(y_test,predictions))
- Print “Classification Report”
  - print(classification_report(y_test,predictions))