dataset processing : parameters

|  |  |
| --- | --- |
| parameter | function |
| r | duration of each epochs |

Logistic regression algorithm : parameters

|  |  |
| --- | --- |
| function | parameter Explain |

|  |  |
| --- | --- |
| 1. GridSearchCV(pipe,param\_grid,cv,n\_job) | **- pipe** :List of (name, transform) tuples (implementing fit/transform) that are chained in sequential orde, it have two paramter :-  scaler to scale the features and the type of classifer which is the logistic resgressor  **- param\_grid :** Dictionary with parameters names it have C parameter which Inverse of regularization strength  **-cv :**Determines the cross-validation splitting strategy  **-n\_jobs :**Number of jobs to run in parallel |
| 1. fit(X,y,groups) | X : Training vector (number of sample and number of features)  Y : Target value to predict  groups: Group labels for the samples used while splitting the dataset into train/test set |

Voting regression algorithm : parameters

|  |  |
| --- | --- |
| paramters | Explain |

|  |  |
| --- | --- |
| 1. **random\_state** | Controls the random seed given to each Tree estimator at each boosting iteration , It also controls the random splitting of the training data to obtain a validation set |
| 1. VotingRegressor(estimators   = [('gb', r1), ('rf', r2), ('lr', r3)] | estimators : list of (str, estimator) tuples  here we use :-  -gb:gradient boosting regressor  -rf : random forest regressor  -lr : linear regressor |
| 1. fit(X,y) | X : Training vector (number of sample and number of features)  Y : Target value to predict |