Description:

To fit an AR or MA model by given parameters; to compute the log-likelihood and gradients given an AR or MA model and the data.

Usage:

Before using the functions, please “from model import AR,MA”

AR\_model = AR(lag=3, phi=np.array([[1],[1],[1]]), sigma=1, intercept=0.1)

AR\_llh, AR\_grads= AR\_model.loss(data)

AR\_future=AR\_model.predict(X=1,nstep=5)

MA\_model = MA(lag=3, phi=np.array([[1],[1],[1]]), sigma=1, intercept=0.1)

MA\_llh, MA\_grads= MA\_model.loss(data)

MA\_future=MA\_model.predict(X=1,nstep=5)

Argument:

lag: the lag parameter for the fitted model

phi: the coefficients for the fitted model

sigma: the variance of noise for the fitted model

intercept: the intercept of noise for the fitted model

data: the input time series data

X: the time series before the first prediction. The number of columns (the length of the time series) should exceed the given lag for a reasonable prediction.

nstep: number of predictions

Functions:

AR: return a fitted AR model object

AR\_model.loss: return the value of log-likelihood and the tuple consists of gradients

AR\_model.predict: return the prediction of given steps

MA: return a fitted MA model object

MA\_model.loss: return the value of log-likelihood and the tuple consists of gradients

MA\_model.predict: return the prediction of given steps