**SGD**

Here, we are using different momentum and learning rate values for optimizer.

sgd = optimizers.SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=**True**)

model.compile(loss='mean\_squared\_error', optimizer=sgd)

**RMSprob**

* Another optimizer method
* In general, to prefer for recurrent neural network ( can not be effective)
* Suggest to just change learning rate value. Keep other parameter in default value.
* Typical value of decay\_rate is 0.9, 0.99 …

keras.optimizers.RMSprop(lr=0.001, rho=0.9, epsilon=**None**, decay=0.0)

**Adagrad**

* an optimizer with parameter-specific learning rate ( there are different learning rates)
* suggest to not change default value.
* To adapts learning rate based on a moving window of gradient updates instead of accumulating all past gradients.

keras.optimizers.Adadelta(lr=1.0, rho=0.95, epsilon=**None**, decay=0.0)