

[About Keras](#)[Getting started](#)[Developer guides](#)[Keras API reference](#)[Models API](#)[Layers API](#)[Callbacks API](#)[Optimizers](#)[Metrics](#)[Losses](#)[Data loading](#)[Built-in small datasets](#)[Keras Applications](#)[Mixed precision](#)[Utilities](#)[KerasTuner](#)[Code examples](#)[Why choose Keras?](#)[Community & governance](#)[Contributing to Keras](#)[KerasTuner](#)[» Keras API reference](#) / [Optimizers](#) / Adagrad

Adagrad

Adagrad class

```
tf.keras.optimizers.Adagrad(  
    learning_rate=0.001,  
    initial_accumulator_value=0.1,  
    epsilon=1e-07,  
    name="Adagrad",  
    **kwargs  
)
```

Optimizer that implements the Adagrad algorithm.

Adagrad is an optimizer with parameter-specific learning rates, which are adapted relative to how frequently a parameter gets updated during training. The more updates a parameter receives, the smaller the updates.

Arguments

- **learning_rate:** Initial value for the learning rate: either a floating point value, or a [tf.keras.optimizers.schedules.LearningRateSchedule](#) instance. Defaults to 0.001. Note that [Adagrad](#) tends to benefit from higher initial learning rate values compared to other optimizers. To match the exact form in the original paper, use 1.0.
- **initial_accumulator_value:** Floating point value. Starting value for the accumulators (per-parameter momentum values). Must be non-negative.
- **epsilon:** Small floating point value used to maintain numerical stability.
- **name:** Optional name prefix for the operations created when applying gradients. Defaults to "Adagrad".
- ****kwargs:** Keyword arguments. Allowed to be one of "clipnorm" or "clipvalue". "clipnorm" (float) clips gradients by norm and represents the maximum L2 norm of each weight variable; "clipvalue" (float) clips gradient by value and represents the maximum absolute value of each weight variable.

Reference

- [Duchi et al., 2011.](#)