CSC413 Summary Notes:

Lecture 6:

Optimization and Generalization are two important goals in deep learning, yet there are many difficulties associated with them. In lecture 6 several trickes are introduced to help overcome these difficulties and achieve these two goals for practitioners.

**Tricks for Optimization:**

Table

Description automatically generated

--Mini-batch Gradient Descent:

A trade-off between accuracy and time.

Before GPU/CPU is saturated, it is better to increase the batch size.

When GPU/CPU is saturated, Increasing batch size reduces variance, i.e. produce accurate estimate but consumes more time.

Practical Tips:

Text

Description automatically generated

--Initialization:

Break symmetry by initializing weights to small random values.

Initialize weights to be proportional to or (# of input-nodes)

--Learning Rate:

Adjust learning rate higher or lower according to loss curve

Loss curve oscillates widely 🡪 reduce learning rate

Error falls slowly -> increase leraning rate

Learning rate decay strategy

Tip in practice: Typically, a grid search involves picking values approximately on a logarithmic scale, e.g., a learning rate taken within the set {.1, .01, 10−3, 10−4 , 10−5}

--SGD with momentum

Deals with the ill-conditioned curve problem -> gradient in wide dimension offset each other

Momentum + RMSProp = Adam

Text, letter

Description automatically generated

**Tricks for generalization:**

We want to minimize the test error, not just the training error.

Graphical user interface, text, application

Description automatically generated

--Data Augmentation:

Create more data of similar types, by different types of transformation. Methods are different for different types of data(time series, picutres, languages). See original slides for details.

--Reduce # of Parameters:

Diagram

Description automatically generated

Less expressive but still improve generalization. (Neural network favors long and shallow instead of wide and short models)

--Weight Decay:

Large weight tend to overfit and make model too sensitive to input data 🡪 Regularize the magnitude of weights during training by adding a penalization term.

Text, letter

Description automatically generated

--Early Stopping:

A picture containing line chart

Description automatically generated

--Ensemble:

Change the model and do multiple predictions, average all the predictions.

Can adjust the loss function/ initialization/different subset of training data/different hyperparameters/different models.

Text

Description automatically generated

--Stochastic Regularization:

Inject noise into the computation to prevent overfitting.

Text

Description automatically generated

Dropout at test time: don’t do dropout, instead multiple weights by 1 – p.

Batch Normalization and SGD are also considered stochastic regularization.

--