Deep Learning In Python using Keras

Subject: Early Stopping and Other Tips of CNN Models

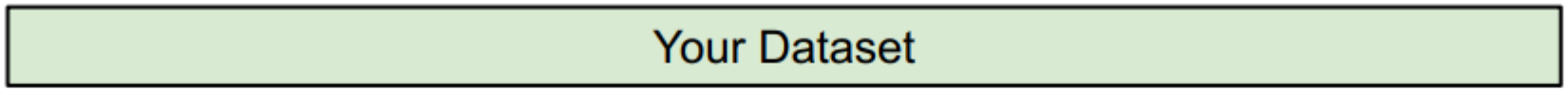
Lecturer : Reza Akbari Movahed

Early Stopping

The best way to splitting dataset for implementing deep learning models

Idea #1: Choose hyperparameters that work best on the data

BAD: K = 1 always works perfectly on training data



Idea #2: Split data into **train** and **test**, choose hyperparameters that work best on test data

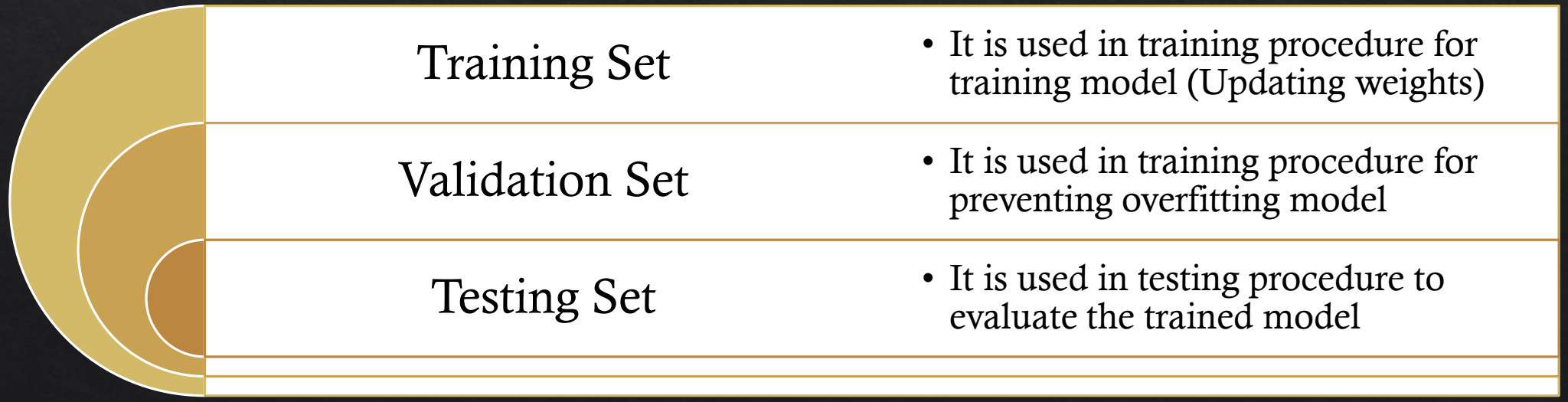
BAD: No idea how algorithm will perform on new data

Idea #3: Split data into **train**, **val**, and **test**; choose hyperparameters on val and evaluate on test

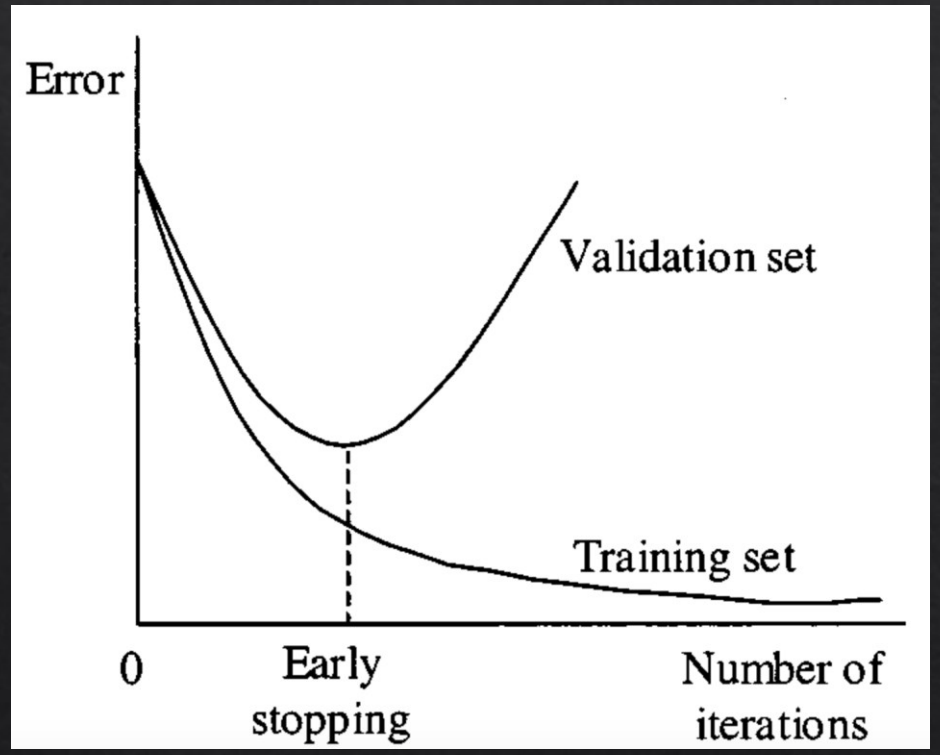
Better!

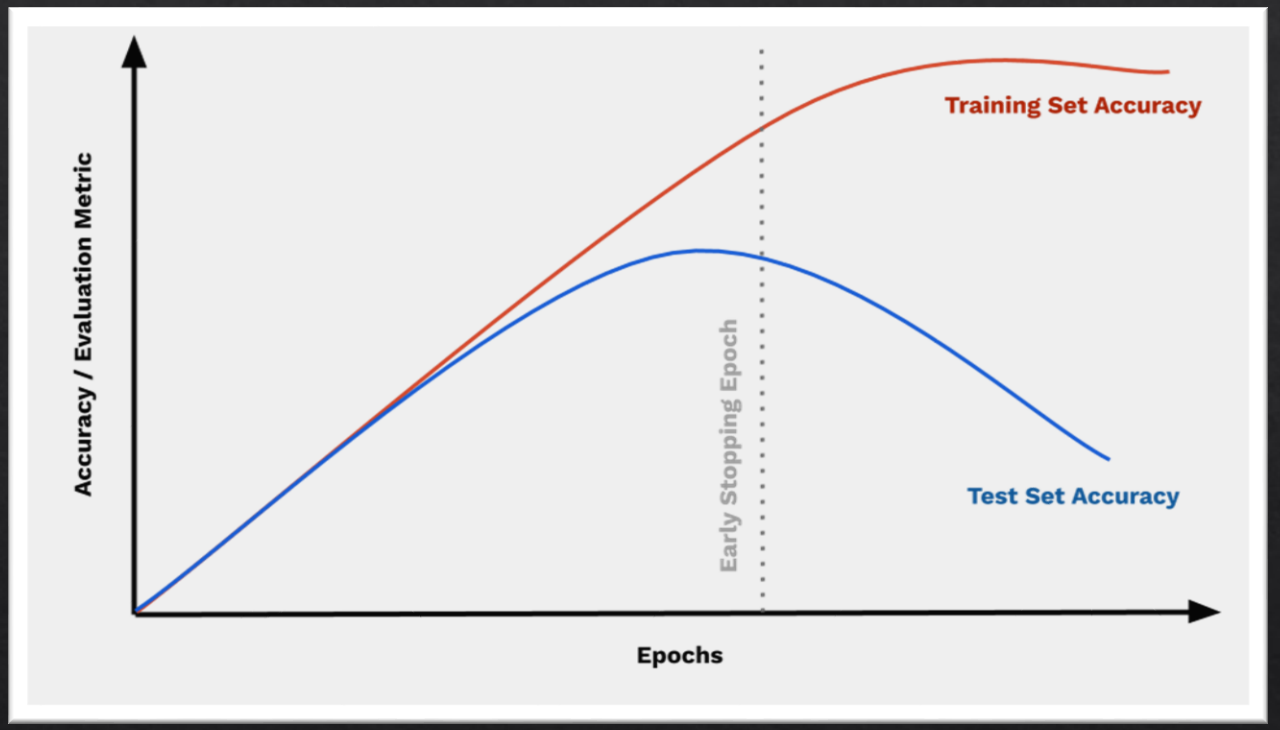
Early Stopping

- Early stopping is a method for preventing overfitting in deep learning models.
- In early stopping, the dataset is categorized in training, validation, and testing subsets.
- In the end of each epoch, the validation set is applied to the model and the validation accuracy (validation loss) is compared with training accuracy (training loss)
- If the training and validation values (accuracy or loss) are such that they are getting away from each other, the model training process will be stopped.

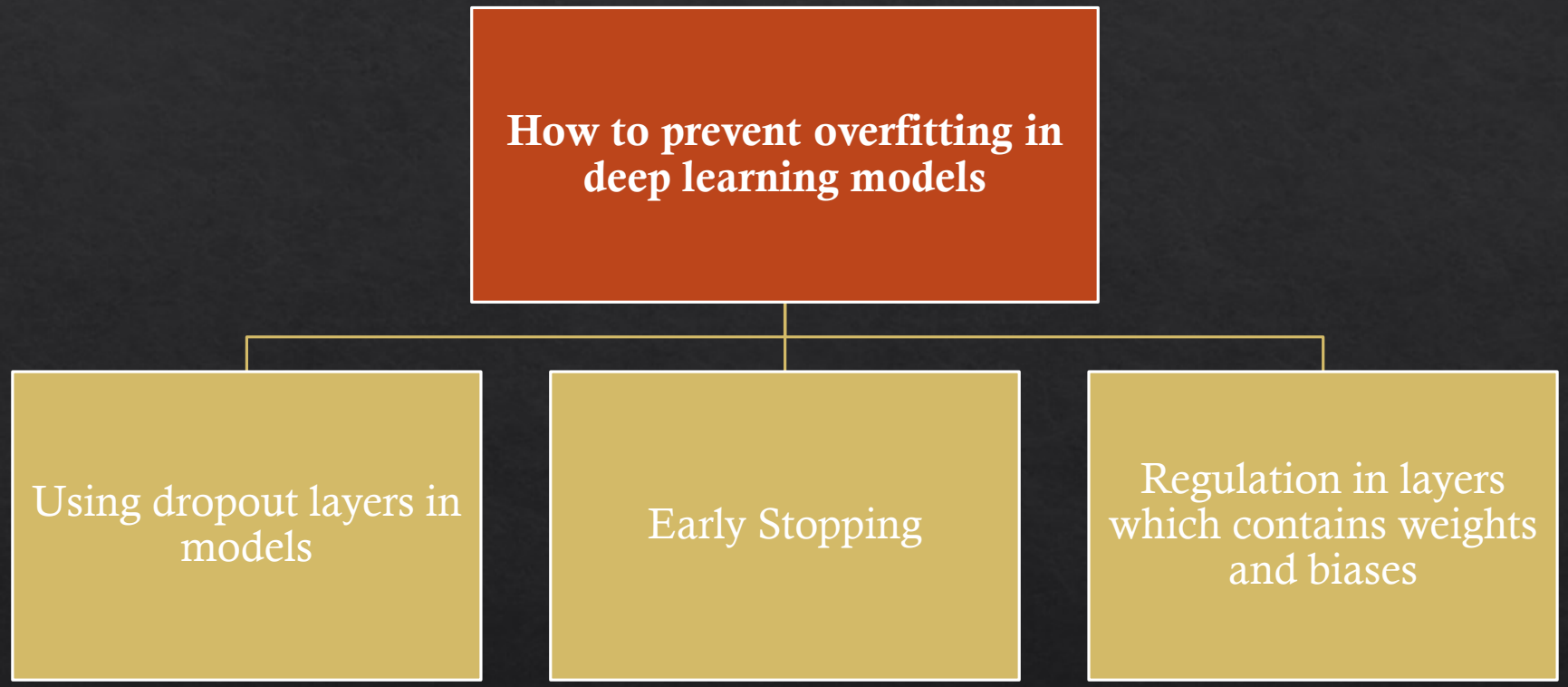


Early Stopping





Other tips


How to prevent overfitting in deep learning models
Using dropout layers in models
Early Stopping
Regulation in layers which contains weights and biases

Normalization Layer

"you want zero-mean unit-variance activations? just make them so."

consider a batch of activations at some layer. To make each dimension zero-mean unit-variance, apply:

$$\widehat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

this is a vanilla differentiable function...

Normalization Layer