

08.05.2025

Machine Learning With TensorFlow

INTRODUCTION TO TENSORFLOW PART II

- Quiz
- Assignments
- Input
- (Breakout Discussions)
- Projects

Introduction to TensorFlow Part II



QUIZ

<https://forms.office.com/e/S4XDcfBZWv>

ASSIGNMENTS

ASSIGNMENTS: WHO WILL PRESENT NEXT?

INPUT: PERFORMANCE ESTIMATION

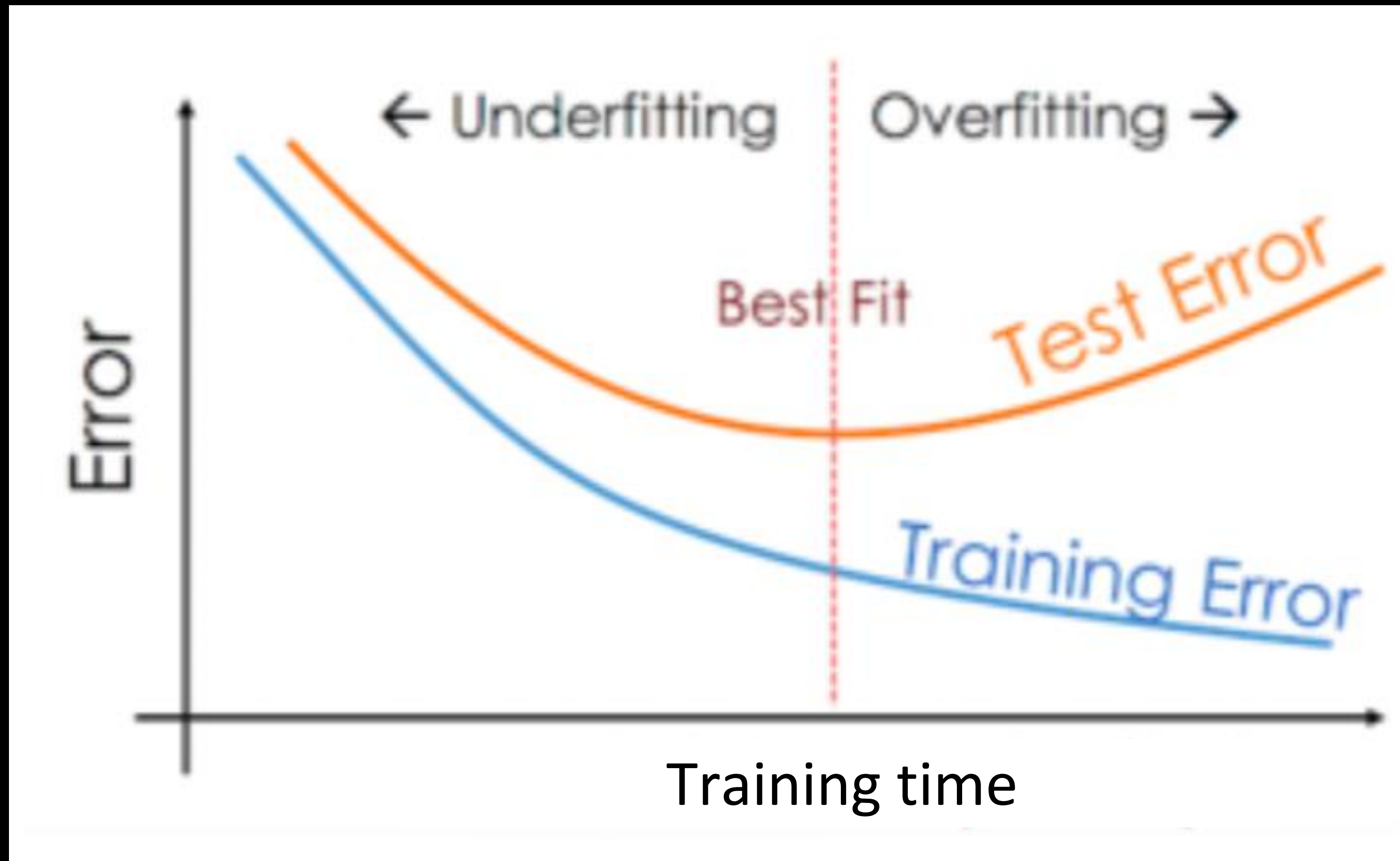
Why do we split the data into training and validation set?

INPUT: PERFORMANCE ESTIMATION

Why do we split the data into training and validation set?

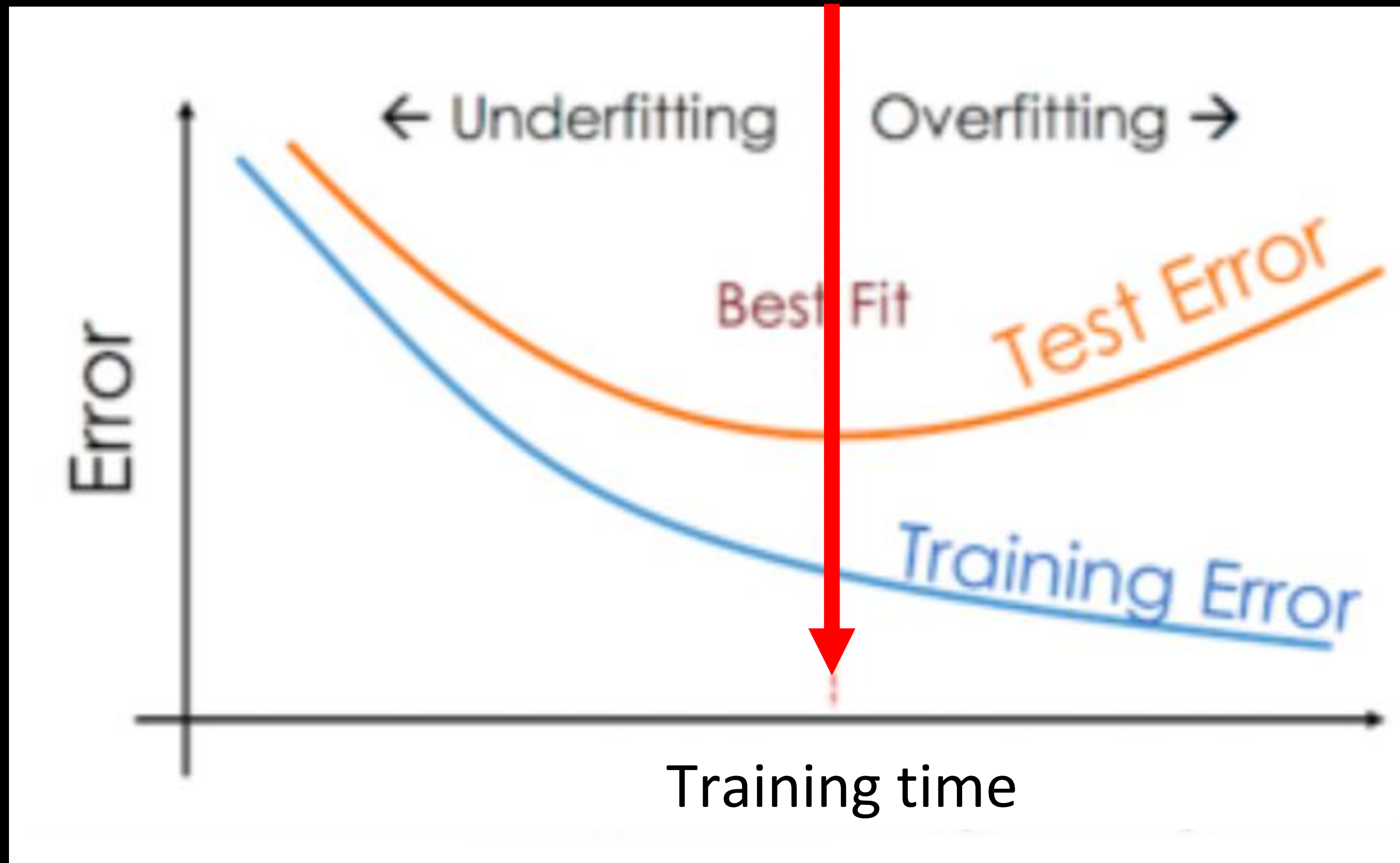
- We want an **unbiased performance estimate**
- If we evaluate on the training set, we will be overconfident
- Example: If the model just learns the solutions by heart, performance on the training data would be perfect, but the model would not generalize well to unseen data

INPUT: TRAINING CURVES & OVERFITTING



INPUT: TRAINING CURVES & OVERFITTING

Early stopping



RECAP: CALLBACKS

TensorFlow

Install

Learn

API

Resources

Community

Why TensorFlow

Search

English

GitHub

Sign in

TensorFlow v2.12.0

Overview

Python

C++

Java

More

Filter

tf.io

tf.keras

Overview

Input

Model

Sequential

activations

applications

backend

callbacks

Overview

BackupAndRestore

BaseLogger

CSVLogger

Callback

CallbackList

EarlyStopping

History

LambdaCallback

LearningRateScheduler

TensorFlow is back at Google I/O! Learn the latest on generative AI, applied ML and more on May 10

Explore program

TensorFlow > API > TensorFlow v2.12.0 > Python

Was this helpful?

Feedback

tf.keras.callbacks.Callback

View source on GitHub

Abstract base class used to build new callbacks.

View aliases

tf.keras.callbacks.Callback()

Used in the notebooks

On this page

Used in the notebooks

Attributes

Methods

on_batch_begin

on_batch_end

on_epoch_begin

on_epoch_end

on_predict_batch_begin

on_predict_batch_end

on_predict_begin

on_predict_end

on_test_batch_begin

on_test_batch_end

on_test_begin

on_test_end

on_train_batch_begin

on_train_batch_end

on_train_begin

on_train_end

set_model

set_params

Methods

- on_batch_begin
- on_batch_end
- on_epoch_begin
- on_epoch_end
- on_predict_batch_begin
- on_predict_batch_end
- on_predict_begin
- on_predict_end
- on_test_batch_begin
- on_test_batch_end
- on_test_begin
- on_test_end
- on_train_batch_begin
- on_train_batch_end
- on_train_begin
- on_train_end

EARLY STOPPING CALLBACK

```
tf.keras.callbacks.EarlyStopping(  
    monitor='val_loss',  
    min_delta=0,  
    patience=0,  
    verbose=0,  
    mode='auto',  
    baseline=None,  
    restore_best_weights=False,  
    start_from_epoch=0  
)
```

INPUT: PERFORMANCE ESTIMATION

Why would we need an additional test set, when we already have a validation set?

What's the downside of a splitting the data only once into training, validation, and test set?

INPUT: PERFORMANCE ESTIMATION

Why would we need an additional test set, when we already have a validation set?

- Hyperparameter tuning is often done based on the validation set
- Validation set thus influences model choice/performance
- **Performance estimate is no longer unbiased** and overfitting may occur

What's the downside of a splitting the data only once into training, validation, and test set?

INPUT: PERFORMANCE ESTIMATION

Why would we need an additional test set, when we already have a validation set?

- Hyperparameter tuning is often done based on the validation set
- Validation set thus influences model choice/performance
- **Performance estimate is no longer unbiased** and overfitting may occur

What's the downside of a splitting the data only once into training, validation, and test set?

- We use only one specific subset of the data to test on
- We could get a **better performance estimate** if we tested on more data

INPUT: k -FOLD CROSS-VALIDATION

4-fold validation ($k=4$)



BREAKOUT DISCUSSIONS

- What layers can typically be found in a computer vision model architecture? What are their purposes? How would you – intuitively – describe how convolutional layers work?
- Suppose you try to distinguish dogs from cats. In your training set, you have 1000 images of dogs and 100 images of cats. You test your trained model on 10 images of dogs and cats. The confusion matrix looks as follows. What's the problem and how can you try to solve it?

| Ground truth | Prediction | |
|--------------|------------|------|
| | dogs | cats |
| | dogs | cats |
| | 10 | 0 |
| | 9 | 1 |

- If you're fast: Recap what the final layer and loss function look like for regression, binary classification, and multiclass classification problems.

PROJECTS MILESTONES

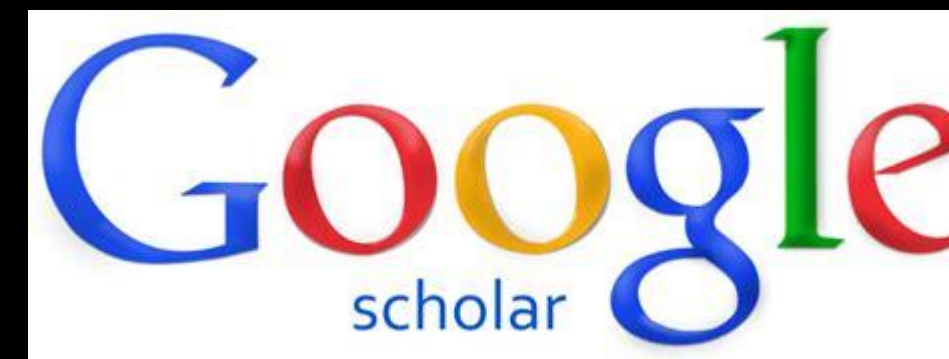
- 24.04. Project pitches
- 01.05. Further Project Proposals and Discussions in Mattermost
- 08.05. Form Groups
- 15.05. *Literature Review (Submission Deadline: 18.05.)*
- 22.05. Dataset Characteristics
- 05.06. Definition of Model Evaluation
- 12.06. Baseline Model Estimation (*Submission Deadline: 15.06.*)
- 22.06. Individual Feedback Sessions
- 03.07. Project Presentations, Part I
- 10.07. Project Presentations, Part II

LITERATURE REVIEW

- What to look for?
 - Datasets that are suitable to solve the task
OR
Tasks that are interesting to solve with a given dataset
 - General advice / best practices in the relevant field (Computer Vision, Natural Language Processing, or Time-Series Prediction)
 - Approaches that have been tried before on similar projects

LITERATURE REVIEW

- Where to look?
 - News articles / blogs
- Academic papers
- Data repositories
- ...



TASKS UNTIL NEXT WEEK

- Setup a GitHub repository for your project based on the provided template (see course handbook)
- Conduct the literature review and upload it to your project repository
- Complete weeks 1 and 2 of the course “Convolutional Neural Networks in TensorFlow”
- Complete the assignment notebook