

# HYPERPARAMETER TUNING

JENS BAETENS

Tunen  
Aanpassen v.d hyperparameters

- manueel
- automatische
- ! leesbaarheid (brute-force)
- efficiënter

The image features a light gray background with a subtle pattern of concentric circles. In the four corners, there are decorative circuit-like lines in a light blue color, consisting of straight lines and small circles, resembling a stylized electronic board.

# HYPERPARAMETERS VS PARAMETERS?

# HYPERPARAMETERS VS PARAMETERS?

## Parameters

- Interne waarden van het model
- Worden geoptimaliseerd bij training

*gewichten / support vectors*

## Hyperparameters

- Configuratie van het model
- Vrij te kiezen

*→  $C$  /  $\gamma$  /  $L_1$  /  $L_2$ -norm*

*→ "graad v.d. hogere orde features"*

*→ type scaling ...*

# HYPERPARAMETER TUNING

Wat is de beste configuratie?

*set hyperparameters*



Hyperparameters



Parameters



Score

Overloop alle mogelijke combinaties

⚙️ n\_layers = 3  
n\_neurons = 512  
learning\_rate = 0.1



Weights optimization



85%

⚙️ n\_layers = 3  
n\_neurons = 1024  
learning\_rate = 0.01



Weights optimization



80%

⚙️ n\_layers = 5  
n\_neurons = 256  
learning\_rate = 0.1



Weights optimization



92%

Kies het model met de hoogste score

*acc.  
precisie  
f1-score  
...*

*...*

WELKE DATA?

Training

Test

# WELKE DATA?

*Hyperparameter Tuning*



Steeds evalueren op ongeziene data

*Volledig afgescheiden*

Daarom zowel validatie als test set nodig (holdout methode)

Veel gebruikte percentages: 70/15/15 tot 98/1/1 bij Big Data

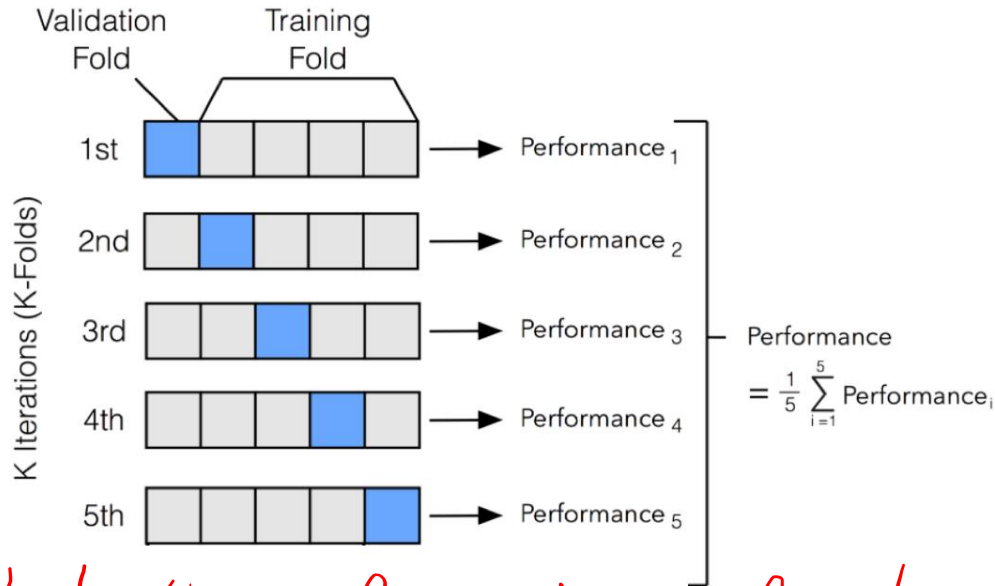
*Weinig Data*

# K-FOLD CROSS VALIDATION

*test data apart*

*→ Determine evaluation v.d. hyperparameters*

*K=5*



*→ Niet ideaal qua efficiëntie → 5 keer trainen i.p.v. 1*





# DATA LEAKAGE

Data van buiten de trainingsdata gebruikt voor het model te trainen

Probleem: bereikte performantie minder betrouwbaar voor ongeziene data

Vooraf risico bij:

- Tijdreeksen → *data uit de toekomst*
- Meerdere lijnen die tot dezelfde persoon/klant behoren  
↳ *duplicaten*

# DATA LEAKAGE

Hoe minimaliseren?

- Geen features die ingevuld worden na de target (behandeld voor ...)
- Maak gebruik van pipelines zodat scalers en imputers werken binnen de folds en geen data van de gehele dataset gebruiken.
- Pas op met oversampling
- Hou de testdata volledig apart! → *niet voor scaling the fitten / null-waarden*

( Voorbeeld: <https://www.kaggle.com/c/the-icml-2013-whale-challenge-right-whale-redux/discussion/4865#25839>

Grootte van de bestanden en de timestamps maakten het mogelijk om de classificatie uit te voeren.