MLOps

Hyperparameters

02457 Machine Learning Operations

Nicki Skafte Detlefsen,
Postdoc
DTU Compute

Why do we optimize hyperparameters

https://paperswithcode.com/

Many reasons

1. Get a working model
2. Get a more robust model
3. Get better performance
4. Something else...

”We are all SOTA suckers”

- Ole Winther

Image Classification

Edit Task

Computer Vision

1483 papers with code **50** benchmarks **104** datasets

About

Edit

Image Classification is a fundamental task that attempts to comprehend an entire image as a whole. The goal is to classify the image by assigning it to a specific label. Typically, Image Classification refers to images in which only one object appears and is analyzed. In contrast, object detection involves both classification and localization tasks, and is used to analyze more realistic cases in which multiple objects may exist in an image.

Source: Metamorphic Testing for Object Detection Systems

Benchmarks

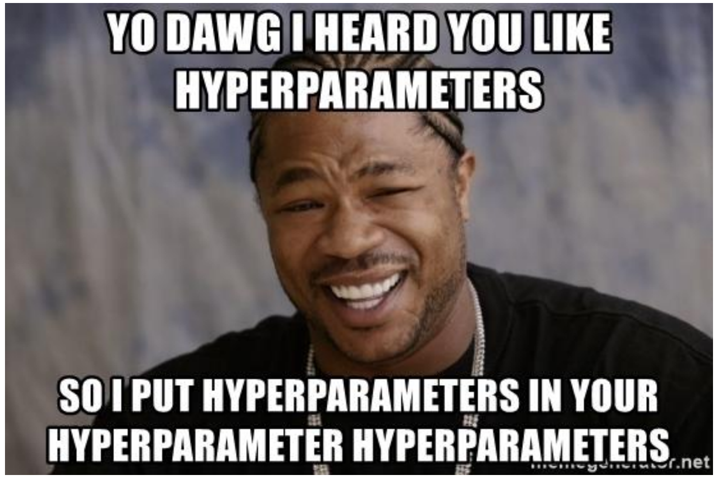
Add a Result

TREND	DATASET	BEST METHOD	PAPER TITLE	PAPER	CODE	COMPARE
	ImageNet	Meta Pseudo Labels (EfficientNet-L2)	Meta Pseudo Labels			See all
	CIFAR-10	EffNet-L2 (SAM)	Sharpness-Aware Minimization for Efficiently Improving Generalization			See all
	CIFAR-100	EffNet-L2 (SAM)	Sharpness-Aware Minimization for Efficiently Improving Generalization			See all
	STL-10	Wide-ResNet-101 (Spinal FC)	SpinalNet: Deep Neural Network with Gradual Input			See all

What is considered a hyperparameter?

- Everything that effects training:
  - Model architechture
  - Optimizer structure

- Question: Is the following a hyperparameter?
  - Dataset
  - Seed



Detour: What should we optimize?

Regardless of chosen model, hyperparameter, search strategy etc. it all depends that we can define some **Metric** for which we can determine if a set of hyperparameters h_1 is better than h_2

In general we will say that

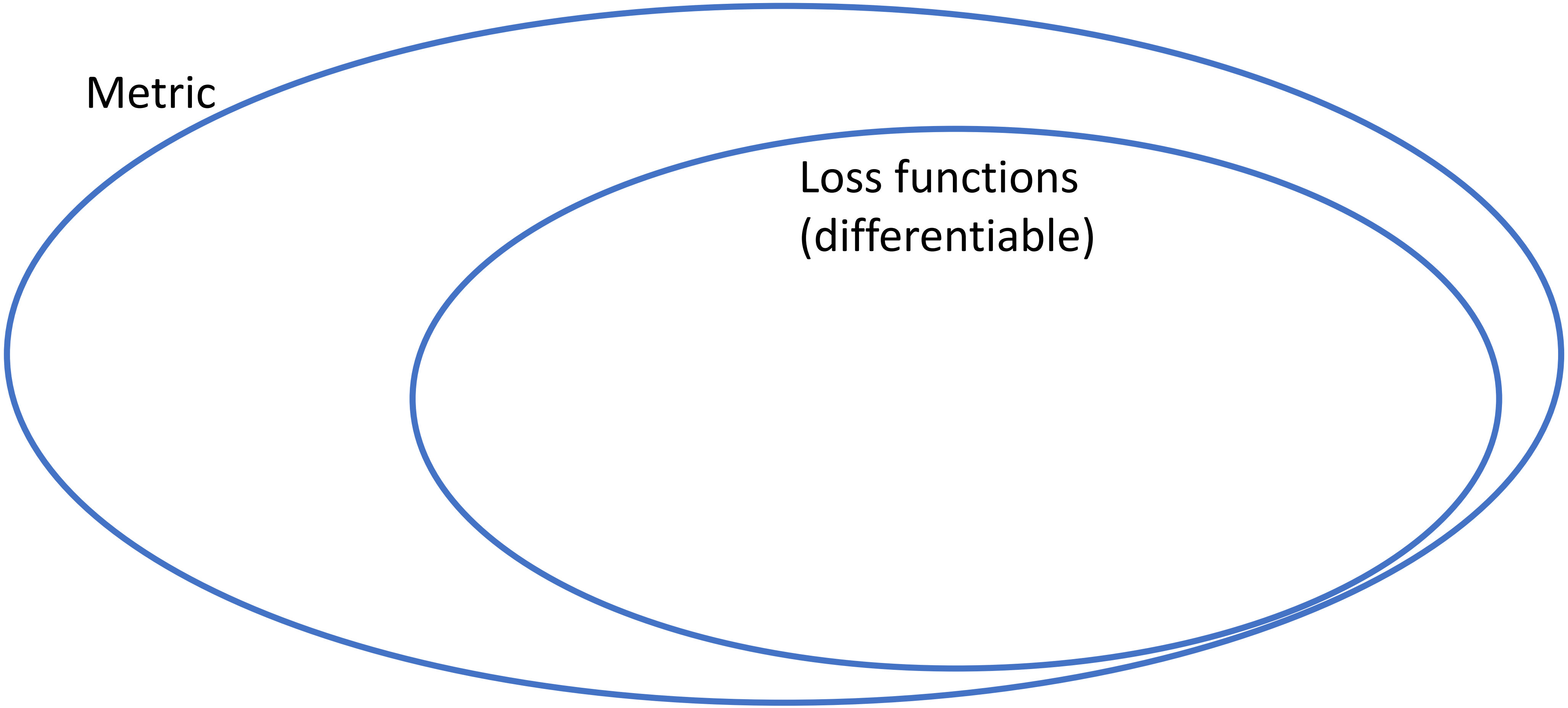
$$h_1 \text{ is better than } h_2$$

If it holds for our **Metric M** that

$$M(h_1) < M(h_2)$$

What is a metric?

Any quantity that measures ”goodness of fit”



Generalization error

The generalization error/expected loss/risk is given by

$$I[f] = \int_{X\times Y} M(f(x), y) p(x, y)\, dxdy$$

where f is some function $f: X \to Y$, M denotes the evaluation Metric and $p(x, y)$ is the joint probablility distribution between x and y

We cannot optimize this directly because $p(x, y)$ is unknown and even if we knew it, the integral would be intractable.

Generalization error

We can calculate the empirical error

$$I_n[f] = \frac{1}{n}\sum_{i=1}^{n} M(f(x_i), y_i)$$

Which measured the "error" that our function f does on n datapoints measured by metric M

An algorithm is said to **generalize** if

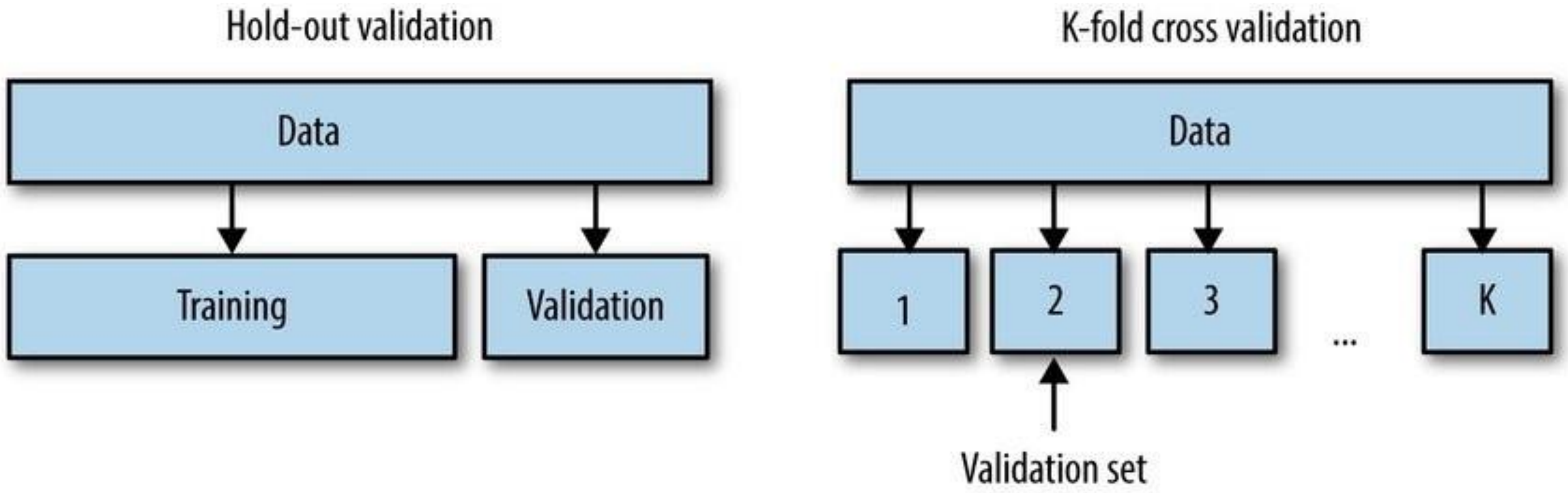
$$\lim_{n\to\infty} I[f] - I_n[f] = 0$$

however, this does not solve our problem because $I[f]$ is still unknown

Cross-validation

Cross-validation in a nutshell:

We take some of our data away which we can use to estimate the true generalized loss $I[f]$



Question: Which one do we use in deep learning and why?

Back to topic

We now assume that we have

Choice of some hyperparameters

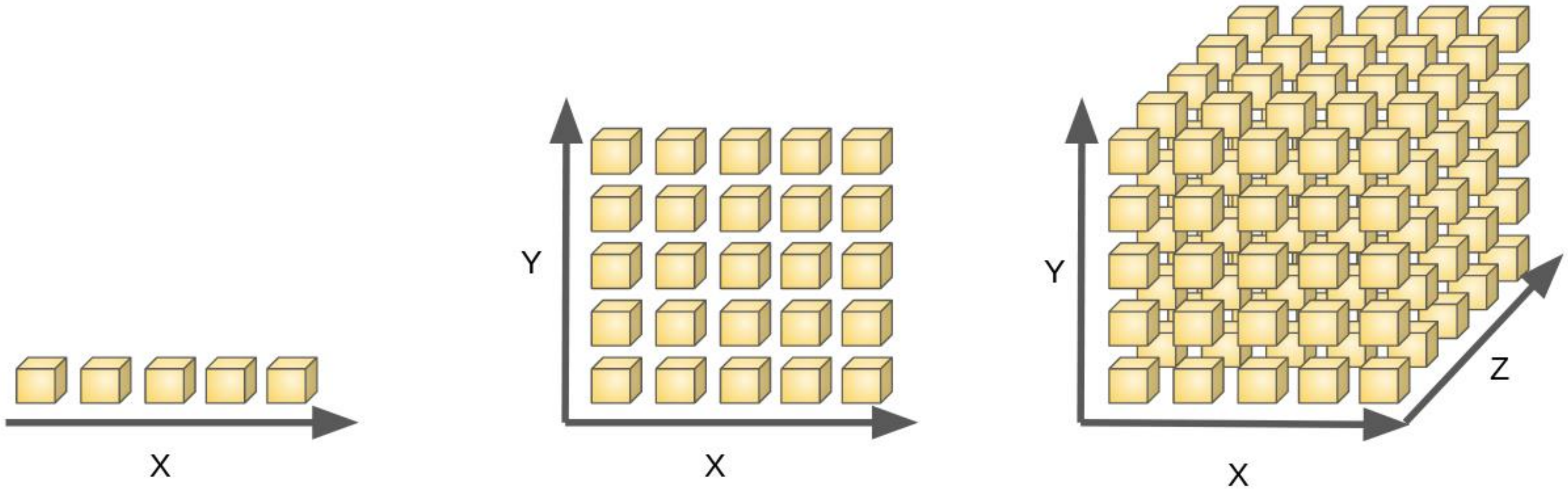
Some metric we trust

Some way to measure the generalization error (validation set)

How should we investigate our hyperparameter space?

What about doing grid search?

Curse of dimensionality?



The number of combinations grows exponentially in the number of parameters we try to optimize.

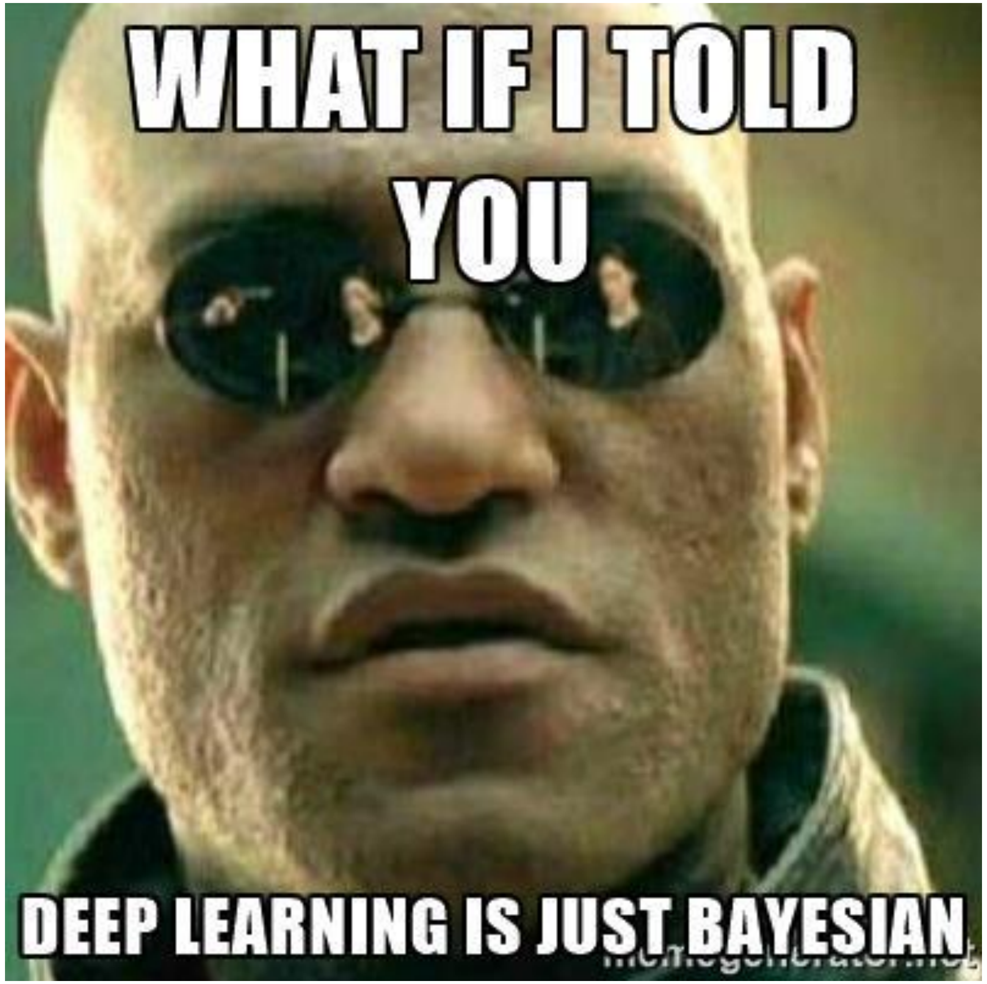
Baysian to the rescue

Baysian optimization tries to find solution to the problems of the form

$$\max_{x \in A} f(x)$$

Where A is known but f may be arbitarily complex

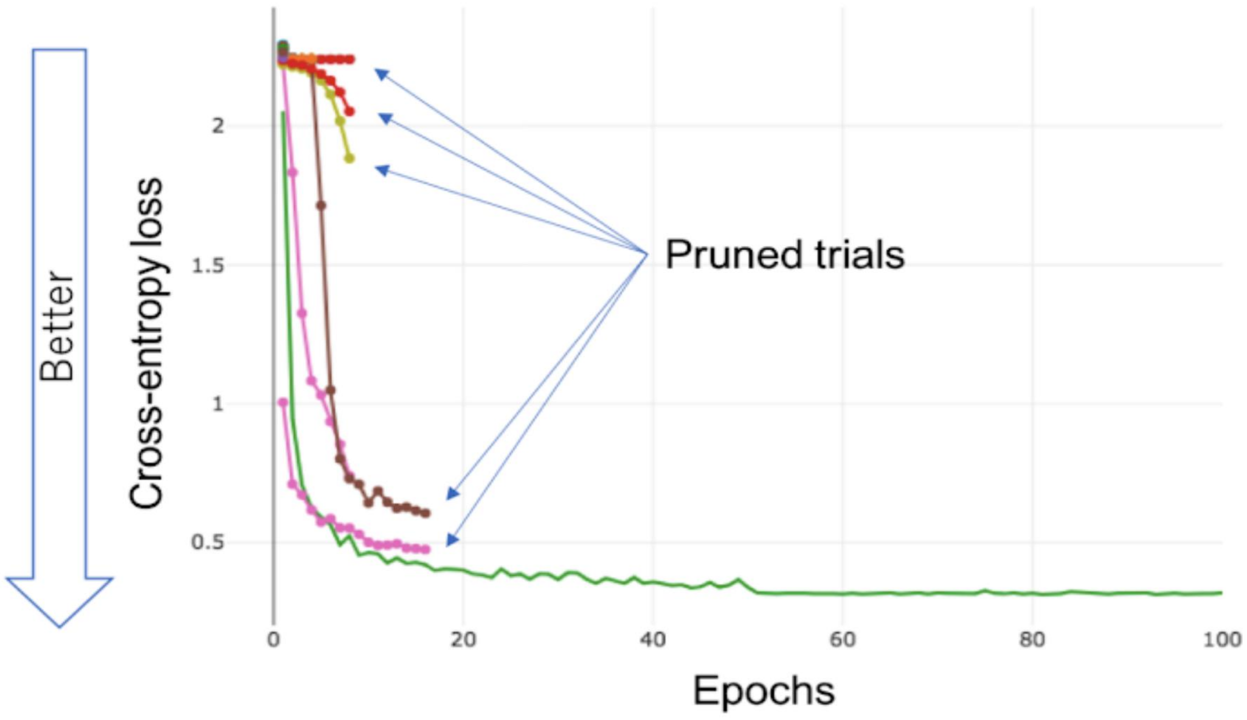
Hyperparameter optimization of the neural networks fit this formulation perfectly meaning that the problem is already solved...



Sampler

- **NSGA-II**
- https://ieeexplore.ieee.org/document/996017
- Evolutionary algorithm

Pruning



Be carefull with over-pruning as you may loose some performance on the floor

Hyperparameter fremework

OPTUNA

https://optuna.readthedocs.io/en/stable/

Meme of the day