Universität
Bremen

Lifelong Machine Learning - Summer 25

Introduction, Recap Static Datasets & Current Practice

Prof. Dr. Martin Mundt
Open World Lifelong Machine Learning Lab
owl-ml.uni-bremen.de
07.04.2025

Universität Bremen

Course Context

Machine learning studies the design of models and training algorithms in order to learn how to solve tasks from data. Whereas historically machine learning has concentrated primarily on static predefined training datasets and respective test scenarios, recent advances also take into account the fact that the world is constantly evolving.

Course Context

Machine learning studies the design of models and training algorithms in order to learn how to solve tasks from data. Whereas historically machine learning has concentrated primarily on static predefined training datasets and respective test scenarios, recent advances also take into account the fact that the world is constantly evolving.

Course Content

The course traverses the fundamentals of lifelong machine learning, spanning methodology of how to effectively *learn in the present*, *remember the past*, and *anticipate an unknown future.*

Course Context

Machine learning studies the design of models and training algorithms in order to learn how to solve tasks from data. Whereas historically machine learning has concentrated primarily on static predefined training datasets and respective test scenarios, recent advances also take into account the fact that the world is constantly evolving.

Course Content

The course traverses the fundamentals of lifelong machine learning, spanning methodology of how to effectively *learn in the present*, *remember the past*, and *anticipate an unknown future.*

Learning Outcome

- understand the breath of factors relevant to lifelong machine learning and their biological inspiration
- design methods to transfer machine knowledge and mitigate interference in continual training
- go beyond rigid train-validate-test methodology towards assessment of lifecycles
- deal with unknown future inputs and adapt machines to diverse contexts

Course requirements

- Basic understanding of the ideas behind machine learning
- We will revisit some select basics, if they are directly necessary to understand the difference/importance to/for lifelong learning
- In-depth knowledge of algorithms will be beneficial, but is not a requirement. It is recommended you catch up on “standard” algorithms if you realize you do not know them
- Lecture attendance will not be checked - but experience shows that attendance is strongly correlated with positive exam outcomes

Universität Bremen

Course format

- Twice a week: Mon 16:00 - 18:00 & Wed 10:00 - 12:00
- Suggestion: start 15 past (CT), no break, end 15 to the full hour

Course format

- Twice a week: Mon 16:00 - 18:00 & Wed 10:00 - 12:00
- Suggestion: start 15 past (CT), no break, end 15 to the full hour
- We will have tutorials **every second** week: 6 tutorials in total
- The first tutorial is on **April 23rd** (Wed 10-12)
- Discounting the 1st week, we thus have 3 lectures & then 1 tutorial

Course format

- Twice a week: Mon 16:00 - 18:00 & Wed 10:00 - 12:00
- Suggestion: start 15 past (CT), no break, end 15 to the full hour
- We will have tutorials **every second** week: 6 tutorials in total
- The first tutorial is on **April 23rd** (Wed 10-12)
- Discounting the 1st week, we thus have 3 lectures & then 1 tutorial
- We likely need to split the group & find a **2nd tutorial slot**

Tutorials

- Hands-on practical exercises for the lecture topics
- Tutorials require basic knowledge of Python

Tutorials held by OWL-ML member Subarnaduti Paul

Universität Bremen

Tutorials

- Hands-on practical exercises for the lecture topics
- Tutorials require basic knowledge of Python
- Tutorials are designed to use Google Colab Notebooks. Colab's GPU compute is sufficient to solve the tutorials
- Content varies from "fill in the gap" & "code a function" to "experiment with settings" & "discuss results"

Tutorials held by OWL-ML member Subarnaduti Paul

Universität Bremen

Tutorials

- Hands-on practical exercises for the lecture topics
- Tutorials require basic knowledge of Python
- Tutorials are designed to use Google Colab Notebooks. Colab's GPU compute is sufficient to solve the tutorials
- Content varies from "fill in the gap" & "code a function" to "experiment with settings" & "discuss results"
- Each student **must** present a solution (attempt) for one exercise sub-task at least once in one of the tutorials

Tutorials held by OWL-ML member Subarnaduti Paul

Course Materials

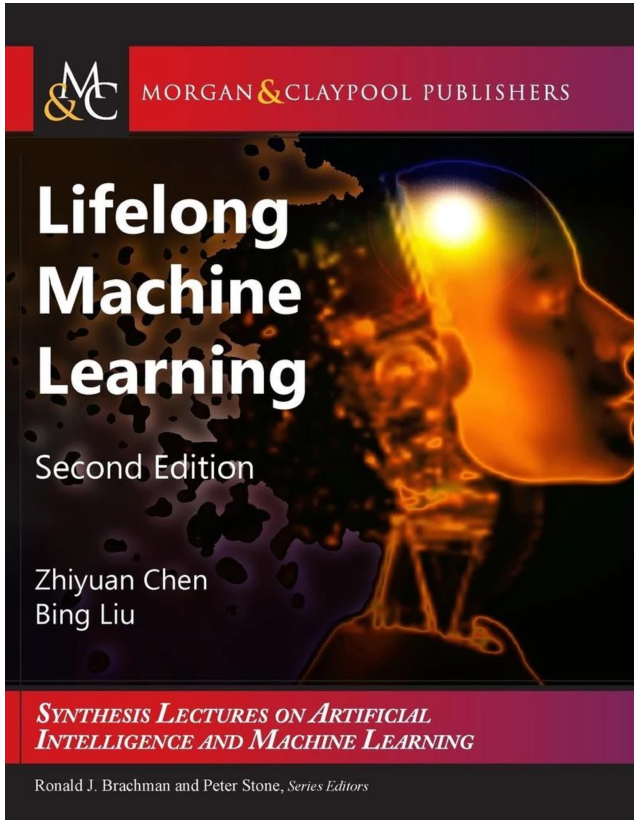
- Slides + recommended materials are sufficient: https://owl-ml.uni-bremen.de/teaching/LLML25/ tutorials: https://github.com/OWL-ML/LLML25-tutorial_notebooks (see also note on StudIP)

Course Materials

- Slides + recommended materials are sufficient: https://owl-ml.uni-bremen.de/teaching/LLML25/ tutorials: https://github.com/OWL-ML/LLML25-tutorial_notebooks (see also note on StudIP)

- It is a rapidly evolving field & consolidation of works is still largely ongoing

- Potentially helpful, but limited & short book: “Lifelong Machine Learning” by Chen & Liu



Course Culture

- This course balances “well-known” foundations with many state-of-the-art frontiers. Expect to not always receive an answer

- Interrupt and ask questions!

Course Culture

- This course balances “well-known” foundations with many state-of-the-art frontiers. Expect to not always receive an answer

- Interrupt and ask questions!

- I will be asking you questions several times in every lecture: please participate actively to make it more fun

- Don’t worry about “wrong” answers or “nonsensical thoughts”. One goal of the course is to understand how challenging ML really is & that we sometimes pretend to have answers

Let’s start with questions right away!

What is machine learning?

Universität Bremen

The conventional, static ML workflow

“A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E”.

Machine Learning,
T. M. Mitchell, McGraw-Hill,1997

Training and test splits

“The result of running the machine learning algorithm can be expressed as a ***function****. The precise form of the function is determined during the* ***training phase****, also known as the* ***learning phase****, on the basis of the* ***training data****.*

Pattern Recognition and Machine Learning,
C. M. Bishop, Springer 2006,
example on image classification: introduction page 2

Training and test splits

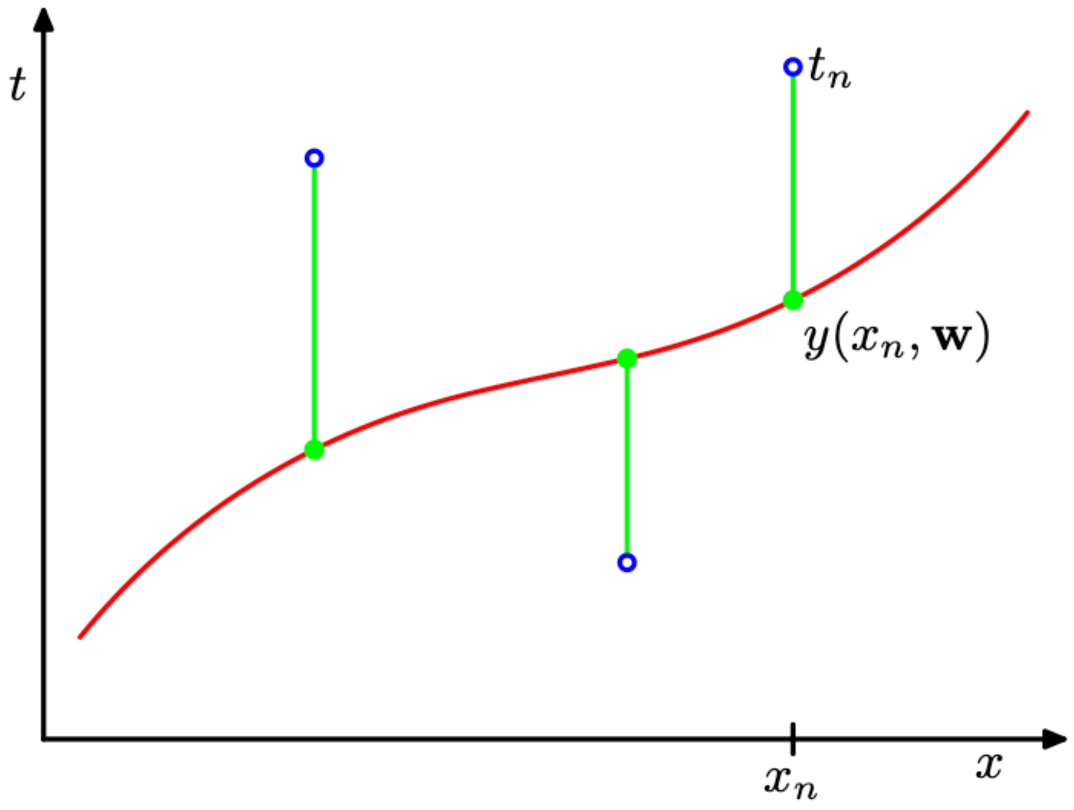
“The result of running the machine learning algorithm can be expressed as a ***function****. The precise form of the function is determined during the* ***training phase****, also known as the* ***learning phase****, on the basis of the* ***training data****.*

Once the model is trained it can then determine the identity of new images, which are said to comprise a ***test set****. The ability to categorize correctly new examples that differ from those used for training us known as* ***generalization****”.*

Pattern Recognition and Machine Learning,
C. M. Bishop, Springer 2006,
example on image classification: introduction page 2

Error functions & learning

Figure 1.3 The error function (1.2) corresponds to (one half of) the sum of the squares of the displacements (shown by the vertical green bars) of each data point from the function $y(x, \mathbf{w})$.



Pattern Recognition and Machine Learning, C. M. Bishop, Springer 2006, example on polynomial curve fitting: intro page 7

Universität
Bremen

Underfitting and overfitting

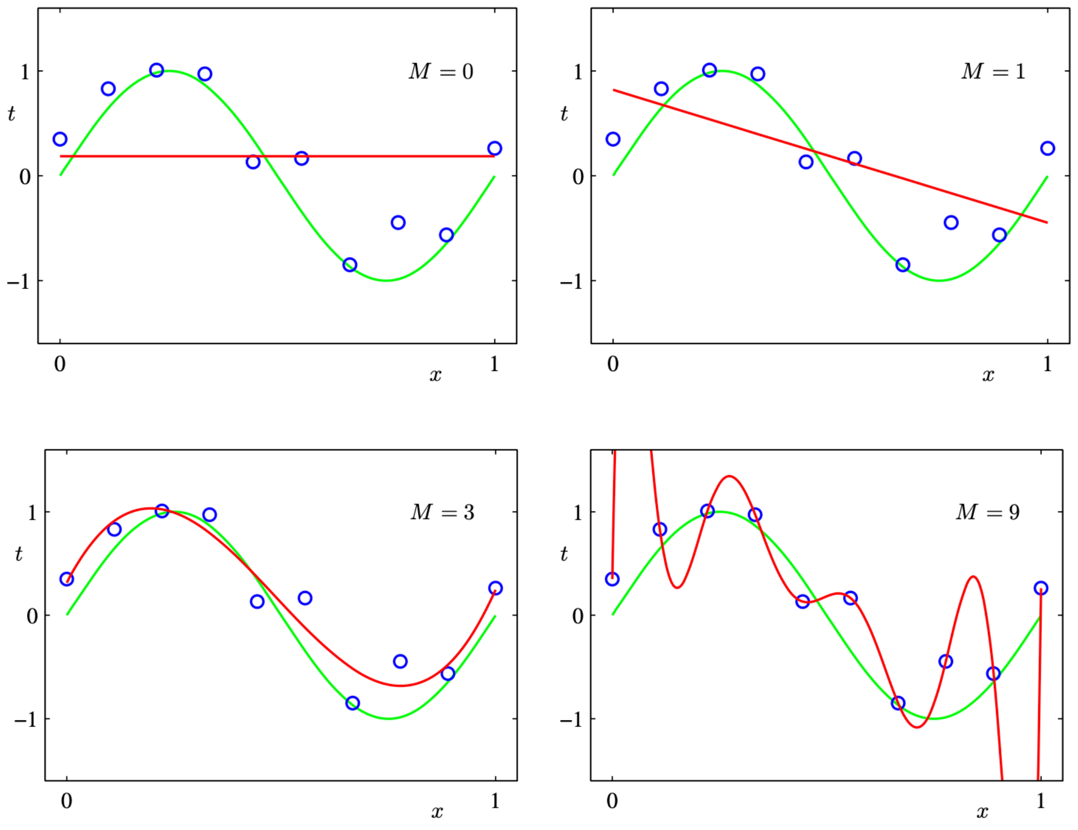


Figure 1.4 Plots of polynomials having various orders M, shown as red curves, fitted to the data set shown in Figure 1.2.

Pattern Recognition and Machine Learning, C. M. Bishop, Springer 2006, example on polynomial curve (over-)fitting: intro page 8

Underfitting and overfitting

“Intuitively, what is happening is that the more flexible polynomials with larger values of M are becoming increasingly tuned to the random noise on the target values”.

Pattern Recognition and Machine Learning, C. M. Bishop, Springer 2006, example on polynomial curve (over-)fitting: intro page 8

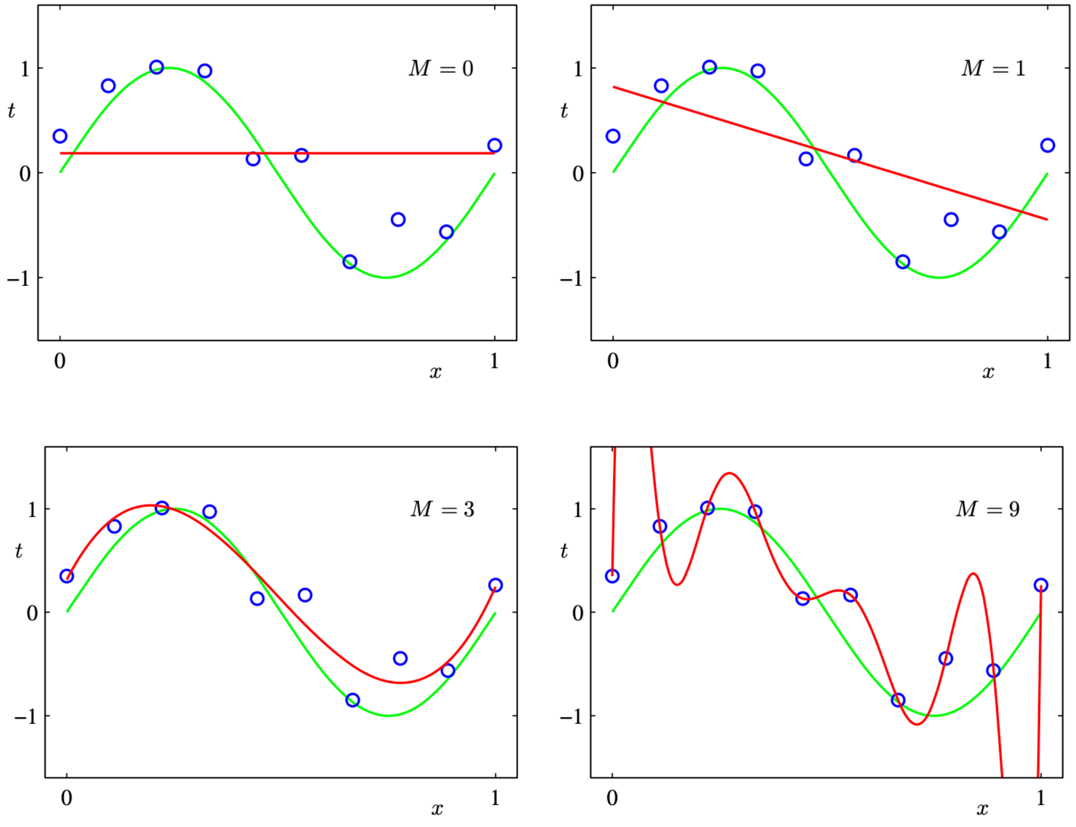


Figure 1.4 Plots of polynomials having various orders M, shown as red curves, fitted to the data set shown in Figure 1.2.

Underfitting and overfitting

"Intuitively, what is happening is that the more flexible polynomials with larger values of M are becoming increasingly tuned to the random noise on the target values".

Pattern Recognition and Machine Learning, C. M. Bishop, Springer 2006, example on polynomial curve (over-)fitting: intro page 8

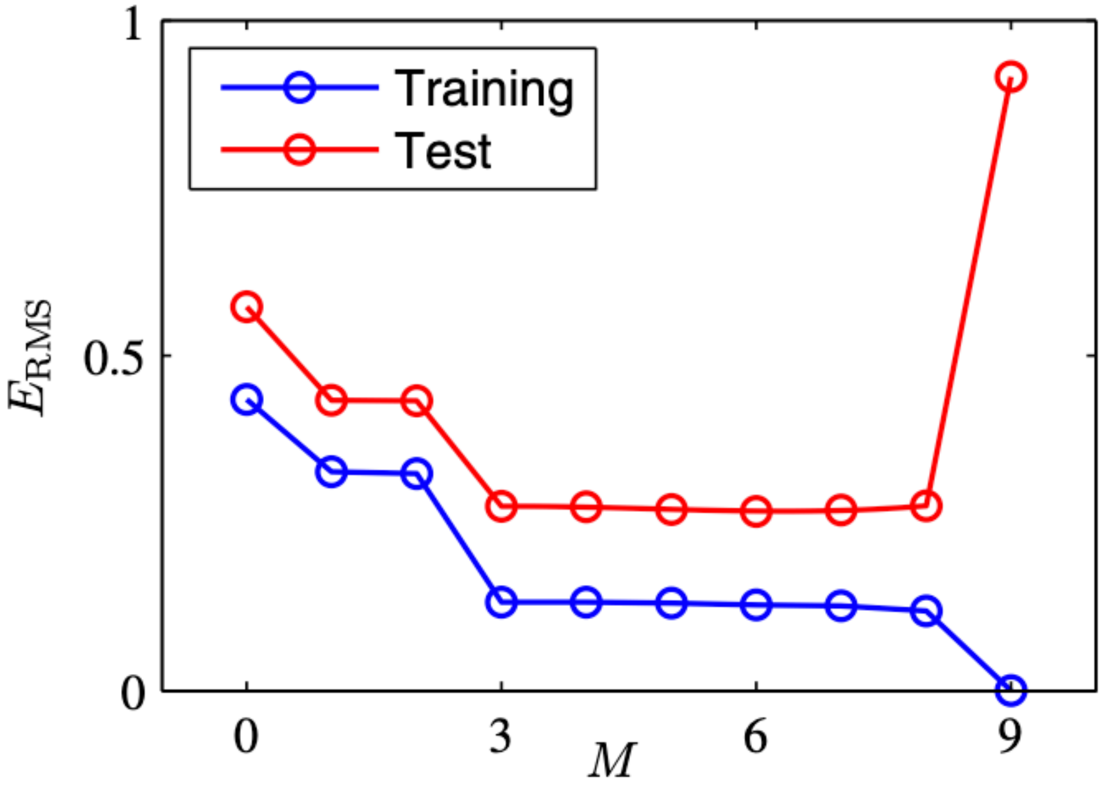


Figure 1.5 Graphs of the root-mean-square error, defined by (1.3), evaluated on the training set and on an independent test set for various values of M.

Universität Bremen

OWL ML

Question Time

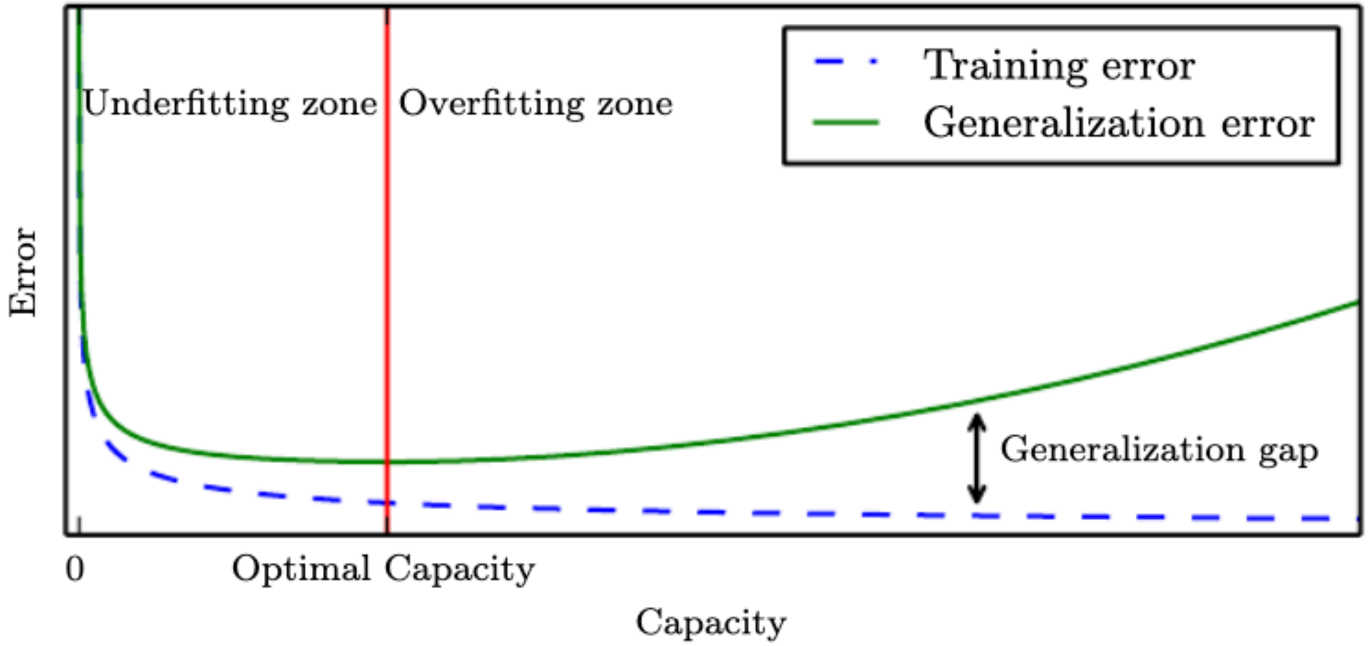
Have these machine learning fundamentals changed in the “deep learning era”?

Underfitting and overfitting

It is still the picture of the "deep learning era"?

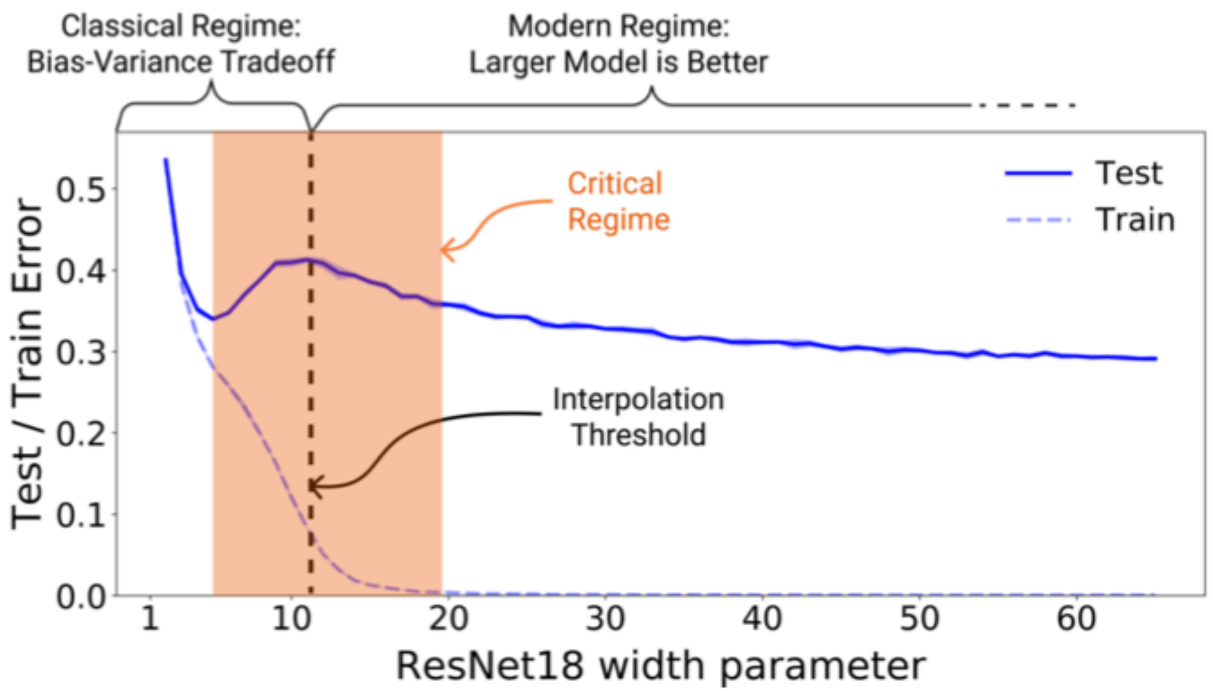
Or have things changed with large data amounts & large compute availability?

Deep Learning, Goodfellow, Bengio, Courville, MIT Press 2016, Machine Learning Basics chapter, page 112.



Deep Double Descent: Overcoming Overfitting?

- With *increased model size*, performance *first gets worse*, *then* gets *better*

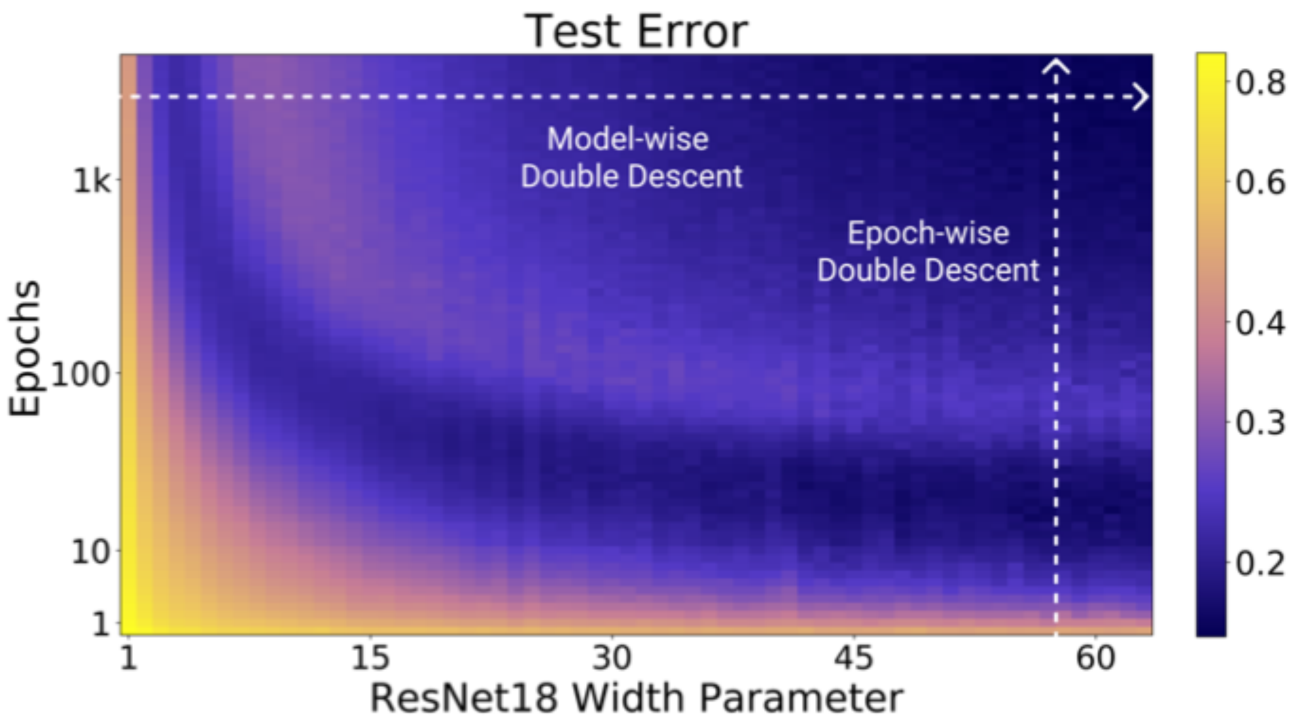


Nakkiran et al, "Deep Double Descent: Where Bigger Models and More Data Hurt", ICLR 2020

Universität Bremen

Deep Double Descent: Overcoming Overfitting?

- With *increased model size*, performance *first gets worse*, *then* gets *better*
- Similar “deep double descent” phenomenon when *increasing training steps*



Nakkiran et al, “Deep Double Descent: Where Bigger Models and More Data Hurt”, ICLR 2020

Universität
Bremen

Question Time

What do you think are the goals of ML?

Goals of the static ML workflow

“Of course, when we use a machine learning algorithm, we ***do not fix the parameters ahead of time****, then sample both datasets. We* ***sample the training set****, then use it to* ***choose the parameters*** *to reduce training set error,* ***then sample the test set****.*

Deep Learning, Goodfellow, Bengio, Courville, MIT Press 2016,
Machine Learning Basics chapter, page 108.

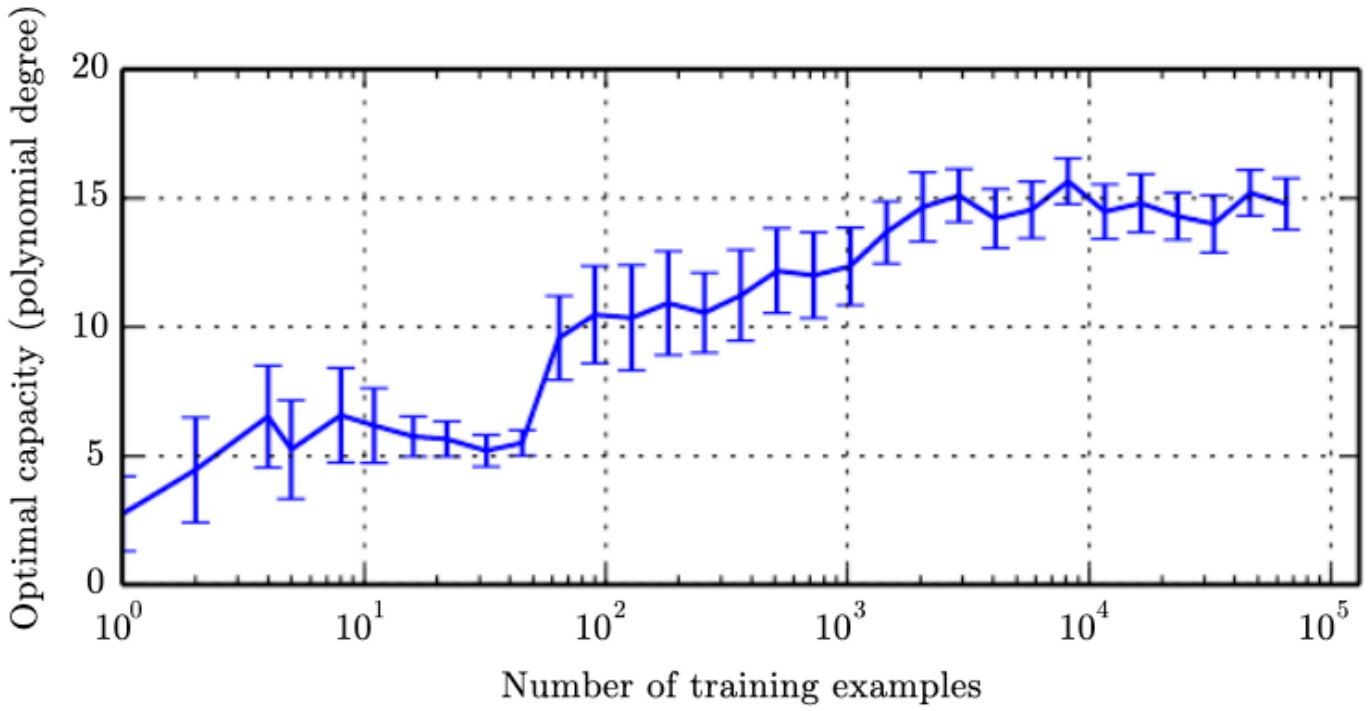
Goals of the static ML workflow

"Of course, when we use a machine learning algorithm, we ***do not fix the parameters ahead of time****, then sample both datasets. We* ***sample the training set****, then use it to* ***choose the parameters*** *to reduce training set error,* ***then sample the test set****.*

The factors determining how well a machine learning algorithm will perform are its ability to:
1. Make the training error small.
2. Make the gap between training and test error small".

Deep Learning, Goodfellow, Bengio, Courville, MIT Press 2016,
Machine Learning Basics chapter, page 108.

ML = finding a large dataset & right model?



Deep Learning, Goodfellow, Bengio, Courville, MIT Press 2016,
Machine Learning Basics chapter, page 114.

Question Time

How are datasets acquired & composed?

Controlled: systematic, but small

Image number	Object pose			Illumination direction		
	Frontal	22.5 ° right	22.5 ° left	Frontal	≈ 45 ° from top	≈ 45 ° from side
1	x			x		
2	x				x	
3	x					x
4		x		x		
5		x			x	
6		x				x
7			x	x		
8			x		x	
9			x			x

Table 3: The labeling of images within each scale in the KTH-TIPS database.


Image #1

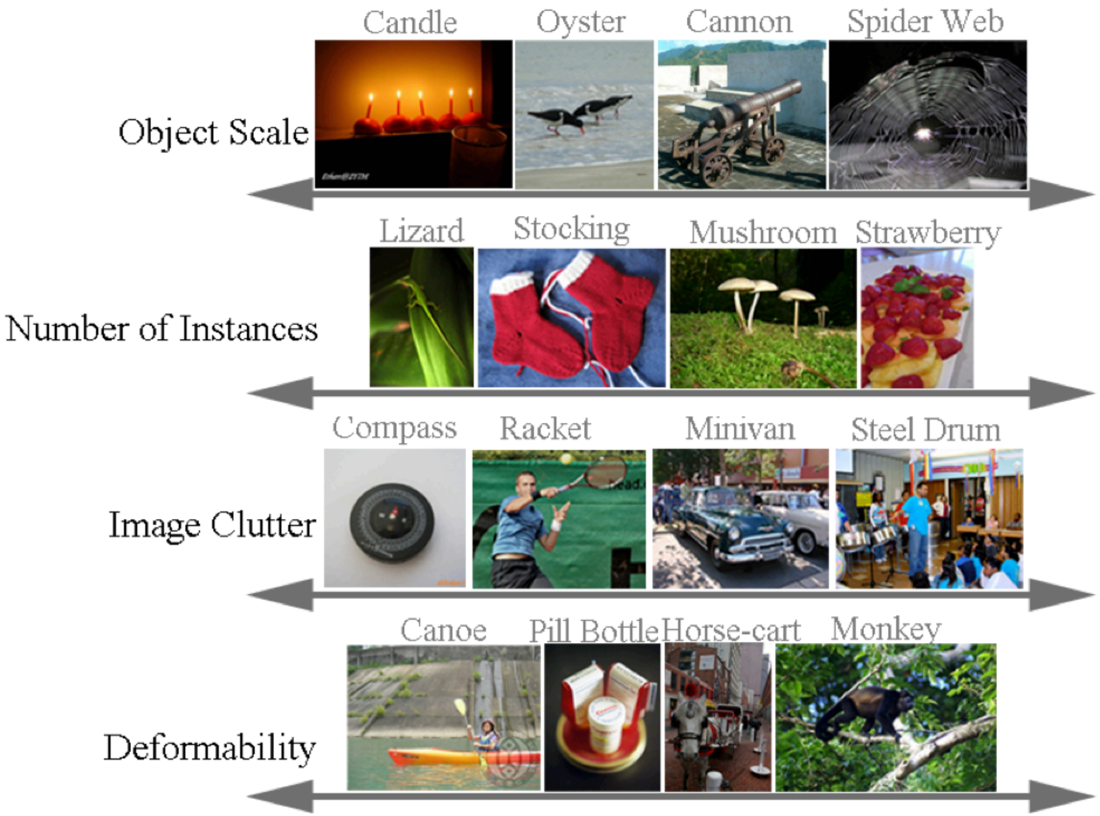

Image #2


Image #4


Image #5

Hayman et al, “On the significance of real-world conditions for material classification”, ECCV 2004 & Fritz, Hayman et al, “The KTH-TIPS database”, technical report 2004

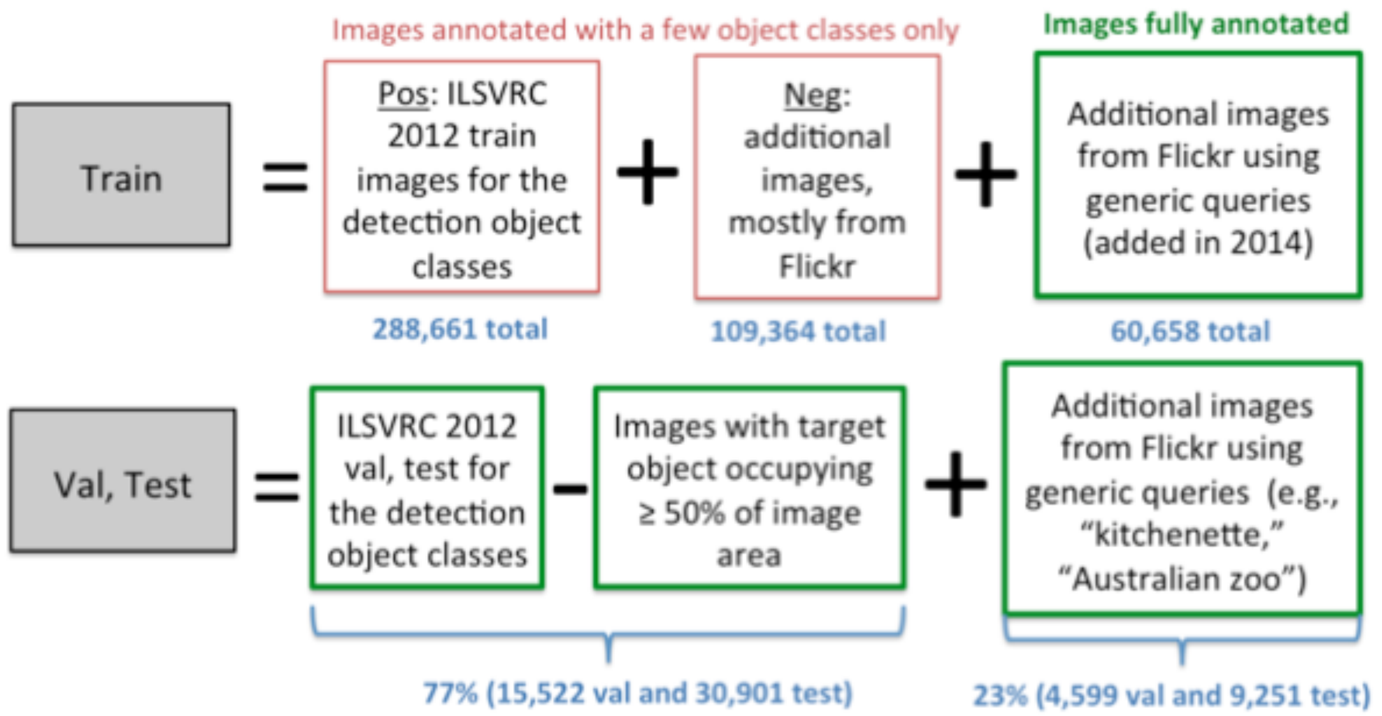
Larger: more diverse, partially uncontrolled



Russakovsky & Deng et al, "ImageNet Large Scale Visual Recognition Challenge, IJCV 2015,
(challenges since 2010)

Universität Bremen

Larger: try to ensure reasonable data splits through complex collection + filtering processes



Russakovsky & Deng et al, "ImageNet Large Scale Visual Recognition Challenge, IJCV 2015, (challenges since 2010)

Larger: filtering often done through crowdsourcing: checking annotator “agreement”

Zhou et al, “Places: An Image Database for Deep Scene Understanding”, J.Vision 17, 2016

“As much as we can get”: crawling (+ filtering?)

Common Crawl maintains a free, open repository of web crawl data that can be used by anyone.

Common Crawl is a 501(c)(3) non–profit founded in 2007.

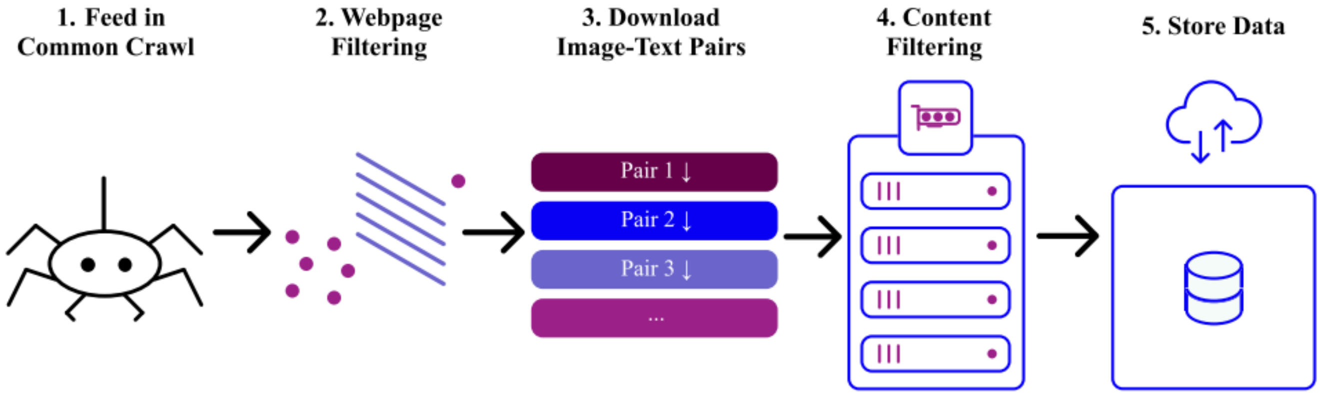


Figure 2: **Overview of the acquisition pipeline:** Files are downloaded, tracked, and undergo distributed inference to determine inclusion. Those above the specified CLIP threshold are saved.

Schuhmann et al, “LAION-5B: An open large-scale dataset for training next generation image-text models”, NeurIPS 2022

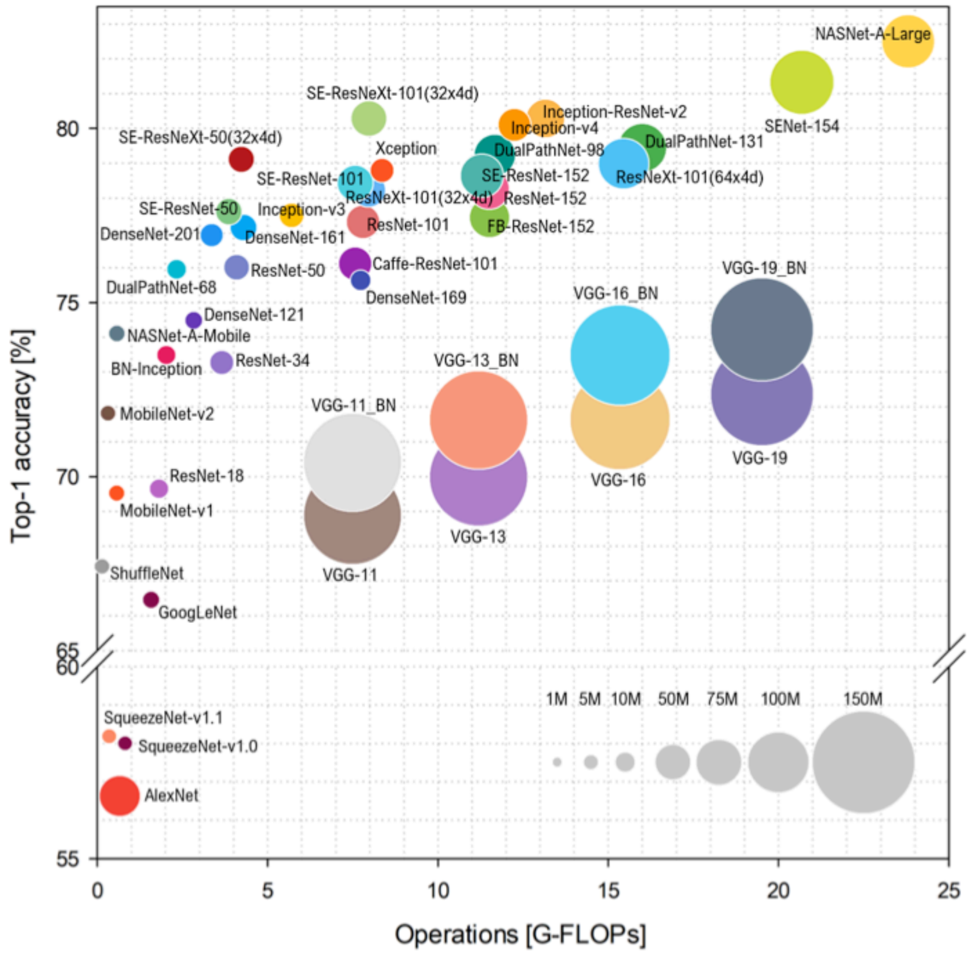
Question Time

Should our primary goal be the solution to such benchmarks? (What are we “solving”?)

“Solving” benchmarks

A very big emphasis has been on getting better numbers on benchmarks

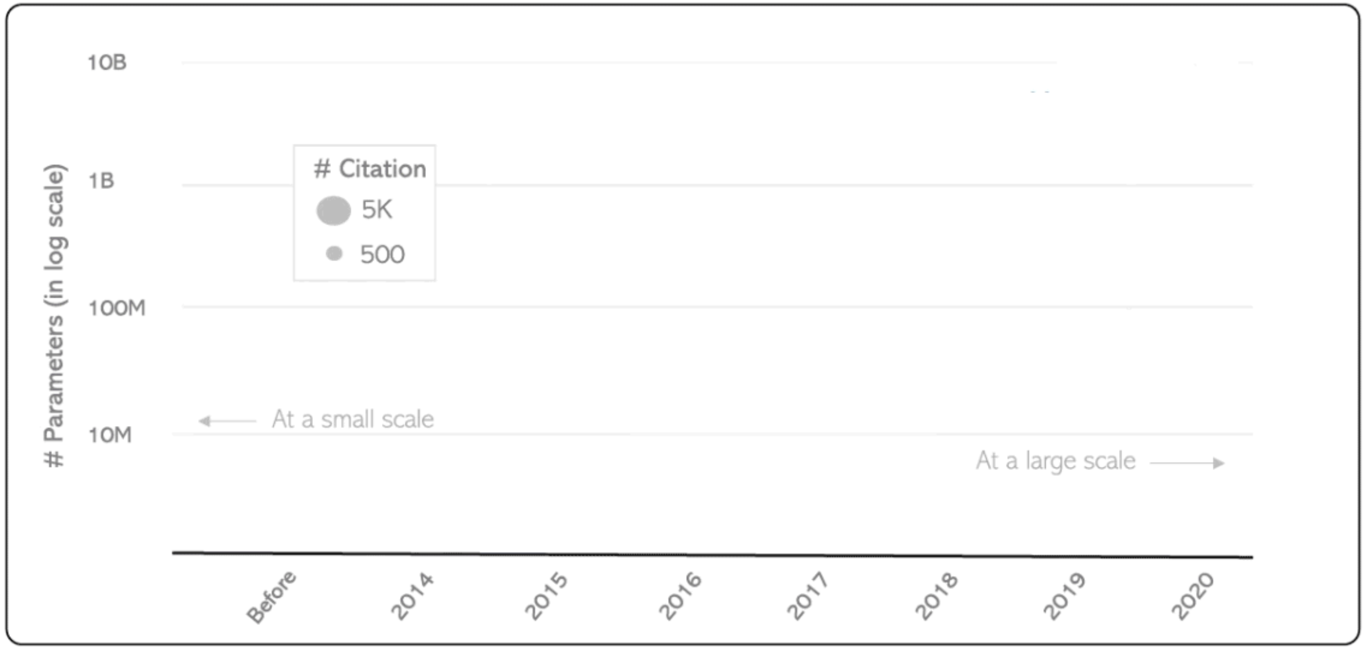
ImageNet is a prime example, where models & compute got bigger and more accurate over time



Bianco et al, “Benchmark Analysis of Representative Deep Neural Network Architectures”, IEEE Access, 2018

Universität
Bremen

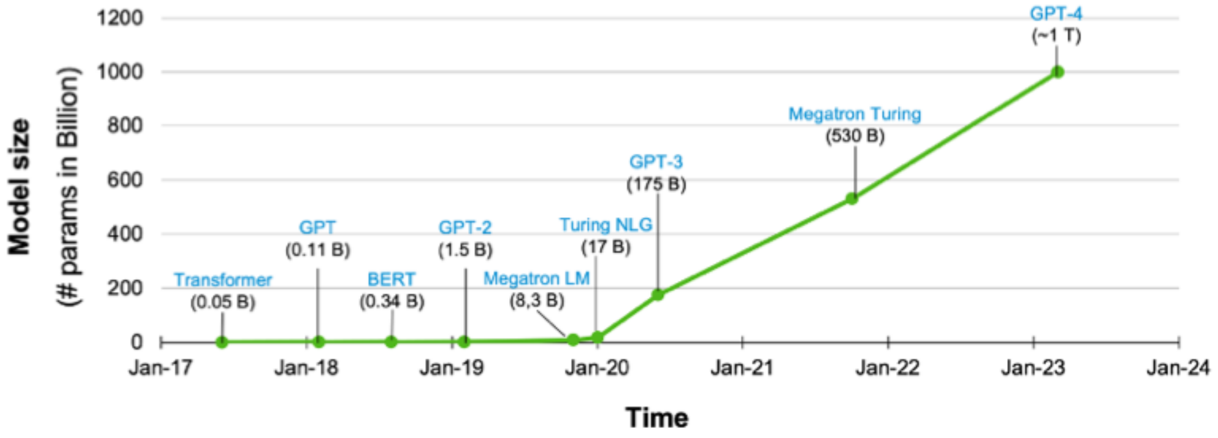
A continued trend: up to 2020



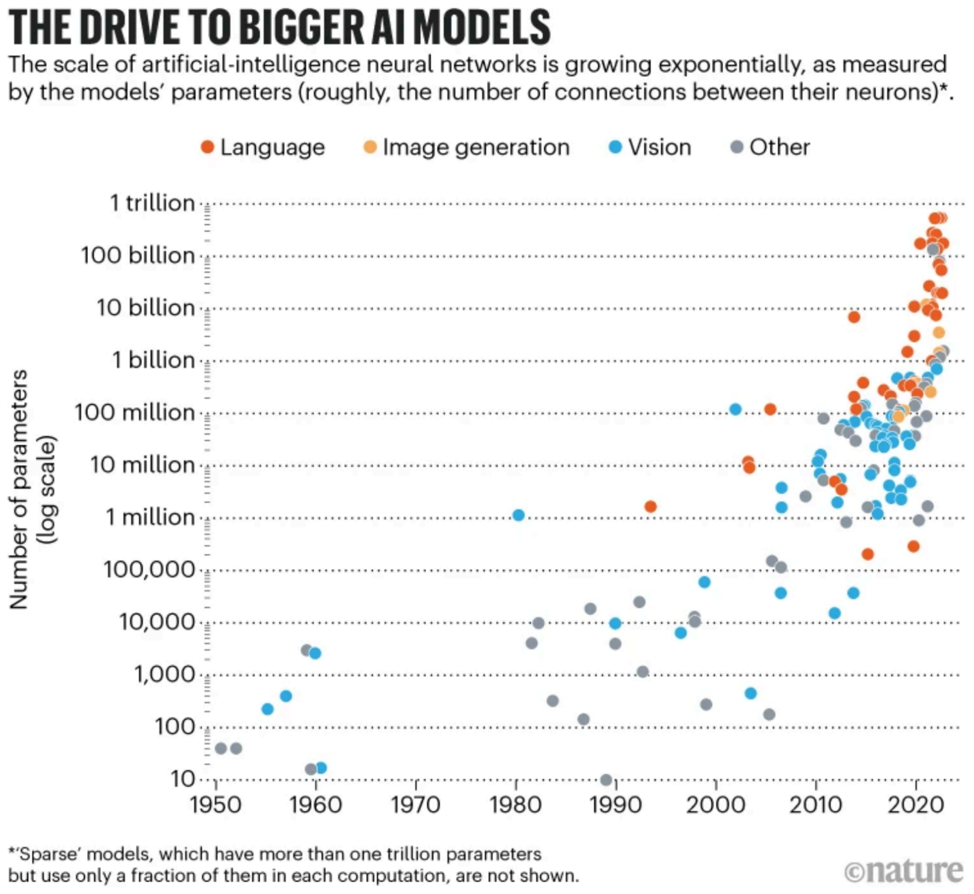
Li & Gao, “A deep generative model trifecta: three advances that work towards harnessing large-scale power, Microsoft Research Blog, 2020: https://www.microsoft.com/en-us/research/blog/a-deep-generative-model-trifecta-three-advances-that-work-towards-harnessing-large-scale-power/

Universität Bremen

A continued trend: "explosion" after 2020



https://medium.com/@gladabhi/optimize-cost-to-host-llm-with-sagemaker-async-endpoints-1a6755e458c5



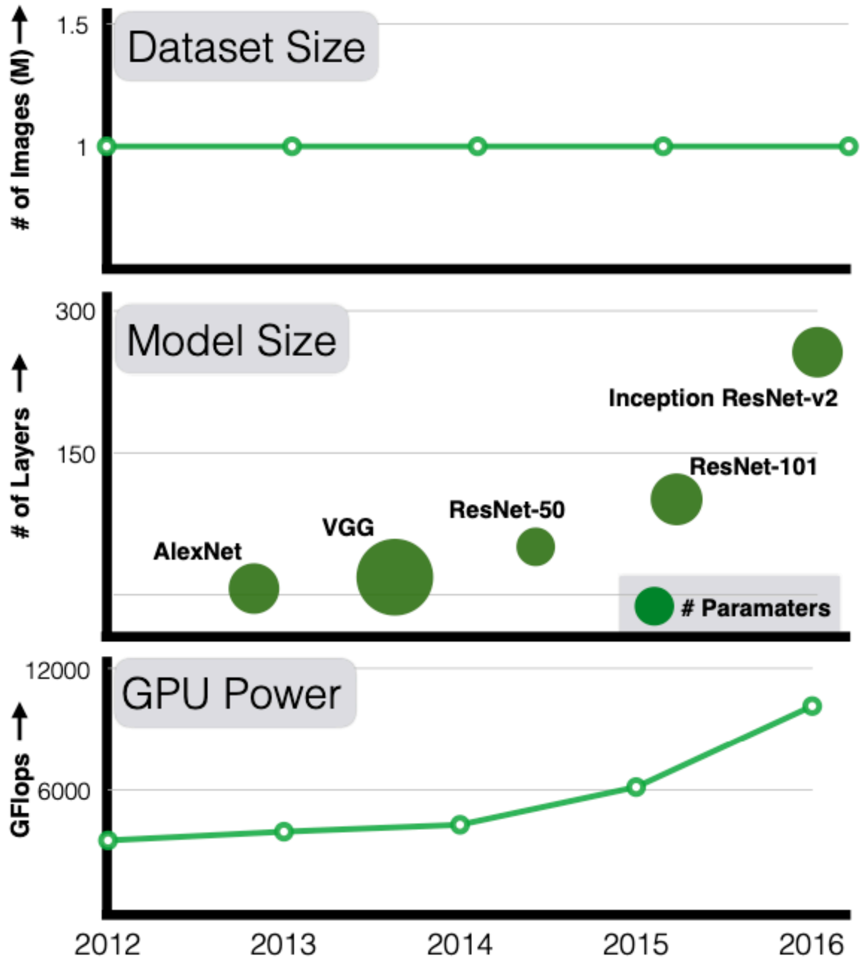
Source: Adapted from Our World in Data, and from J. Sevilla *et al*. Preprint at https://arxiv.org/abs/2202.05924 (2022).

Data and model centrism

It's often "either" models or data

For example, ImageNet has remained largely static* over time

(* excluding some concerns over fair representation and filtering)



Sun et al, "Revisiting Unreasonable Effectiveness of Data in Deep Learning Era", ICCV 2017

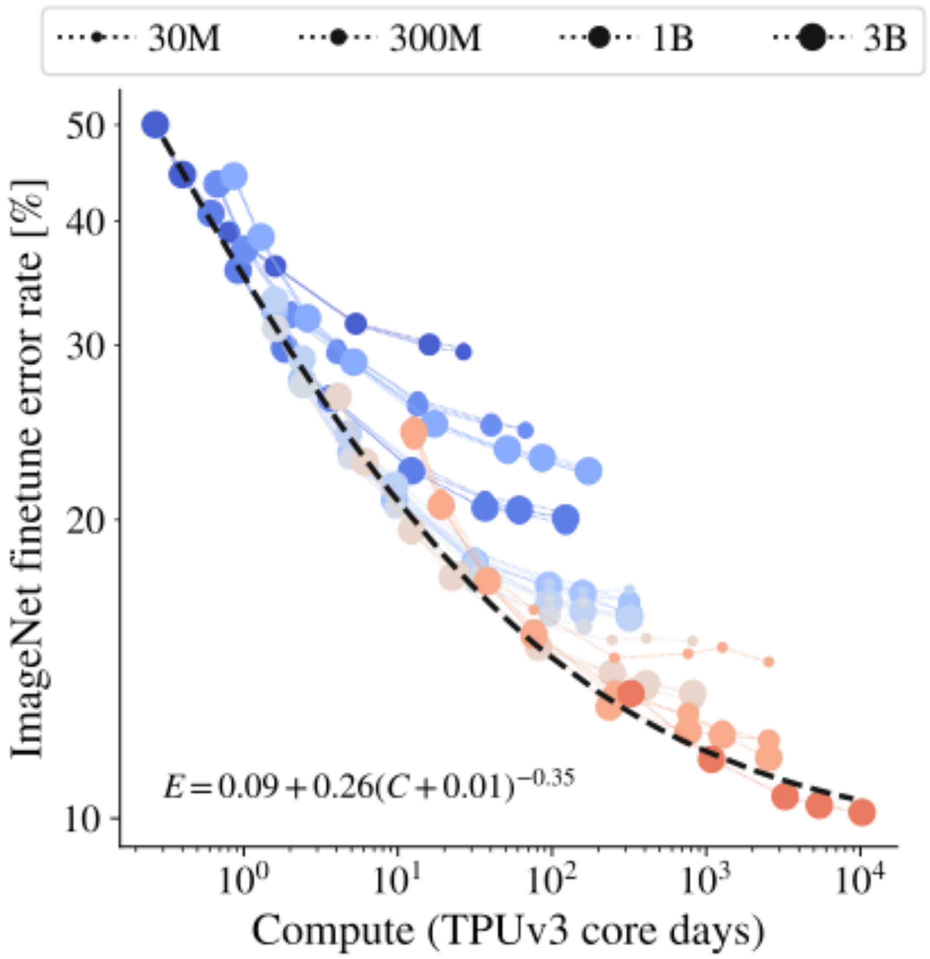
Universität Bremen

Data and model centrism

Or conversely, a model is picked (here a transformer) and datasets are extended

Example from ImageNet to the (non-public) JFT 300M & JFT-3B

There are now many companies and large-scale models that profit primarily from more and larger data sets



Zhai et al, “Scaling Vision Transformers”, CVPR 2022

Universität Bremen

Question Time

How are (the parameters of) machine learning models optimized (learned)?

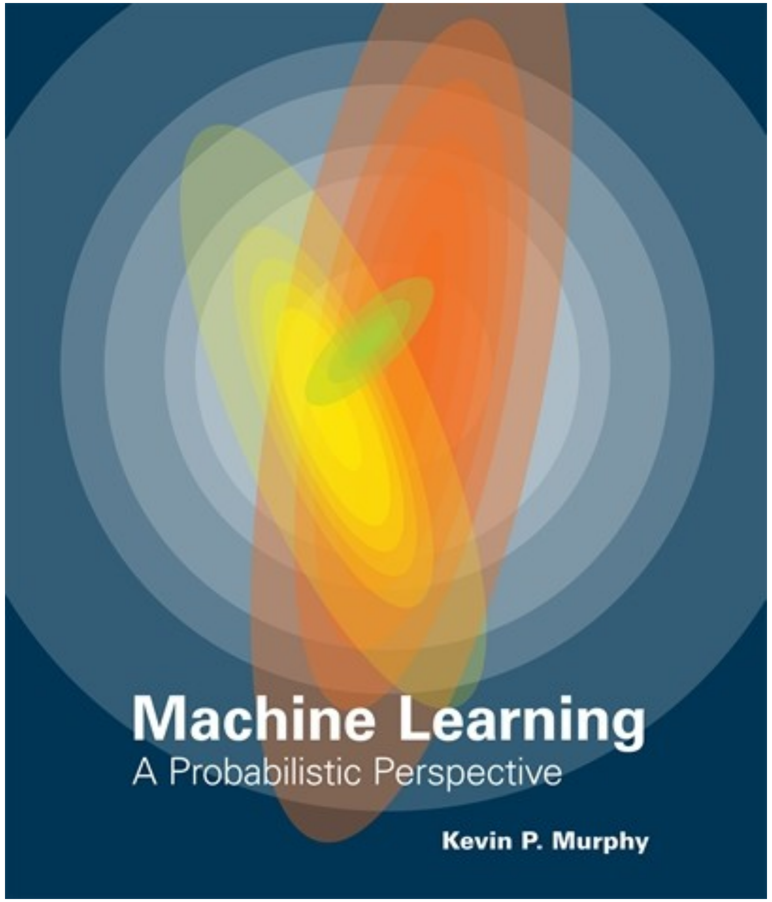
Optimization: risk & loss

What we would like to generally do is the following scenario:

Find a hypothesis or decision procedure:
$\delta : \mathscr{X} \rightarrow \mathscr{A}$

and define the risk or expected loss as:
$R(\theta^*, \delta) = \mathbb{E}_{p(\tilde{D}|\theta^*)} \left[L(\theta^*, \delta(\tilde{D})) \right]$

Where $\tilde{D}$ is data from the true distribution, represented by parameter θ^*



Pages 197-209

Universität Bremen

Question Time

What makes this approach challenging?

Optimization: risk & loss

$$R(\theta^*, \delta) = \mathbb{E}_{p(\tilde{D}|\theta^*)}\left[L(\theta^*, \delta(\tilde{D}))\right]$$

(Some of) the challenges:

- Cannot actually compute above risk (usually don’t know the distribution)
- Besides: if we think of e.g. binary classification, i.e. a 0-1 measure, it can be hard to optimize as it is not smooth

We will learn about some additional challenges later throughout the course, or rather, the consequences of the assumptions we make

Optimization: risk & loss

$$R(\theta^*, \delta) = \mathbb{E}_{p(\tilde{D}|\theta^*)}\left[L(\theta^*, \delta(\tilde{D}))\right]$$

Instead: $R(p^*, \delta) = \mathbb{E}_{(x,y)\sim p^*}\left[L(y, \delta(x))\right]$

-> look at the true but unknown response & predictions $\delta(x)$ given an input x.

Optimization: risk & loss

$R(\theta^*, \delta) = \mathbb{E}_{p(\tilde{D}|\theta^*)}\left[L(\theta^*, \delta(\tilde{D}))\right]$

Instead: $R(p^*, \delta) = \mathbb{E}_{(x,y)\sim p^*}\left[L(y, \delta(x))\right]$

-> look at the true but unknown response & predictions $\delta(x)$ given an input x.

As we still do not know the true distribution, we use empirical estimates: $R_{emp}(D, \delta) = 1/N \sum_{i=1}^{N} L(y_i, \delta(x_i))$

Optimization: risk & loss

$$R_{emp}(D, \delta) = 1/N \sum_{i=1}^{N} L(y_i, \delta(x_i))$$

We then usually chose a loss function, e.g. the mean squared error (supervised):

$$L(y, \delta(x)) = (y - \delta(x))^2$$

or similarly an unsupervised reconstruction:

$$L(y, \delta(x)) = ||x - \delta(x)||_2^2$$

Question Time

Can you explain an algorithm to make use of the loss to tune model parameters?

Optimization: (stochastic) gradient descent

There are various optimization algorithms, the most popular ones are perhaps: (Stochastic) gradient descent - SGD and expectation maximization (EM)

Optimization: (stochastic) gradient descent

There are various optimization algorithms, the most popular ones are perhaps: (Stochastic) gradient descent - SGD and expectation maximization (EM)

Let us consider (S)GD here, as the "workhorse" underlying a lot of deep learning:

- In the simple form, a first order optimization algorithm to find a minimum of a differentiable function
- Achieved by iteratively taking (small) steps in the gradient direction of a function f in the direction of fastest decrease:

$$x_{n+1} = x_n - \lambda \nabla f(x_n) \quad where \quad f(x_0) \geq f(x_1) \geq \ldots \geq f(x_n)$$

Optimization: (stochastic) gradient descent

We can easily transfer this concept to the idea of parameters and losses: $L(\theta) = 1/N \sum_{i=1}^{N} L_\theta(x_i))$

Optimization: (stochastic) gradient descent

We can easily transfer this concept to the idea of parameters and losses: $L(\theta) = 1/N \sum_{i=1}^{N} L_\theta(x_i))$

Then iterative updates become (where in neural nets we backpropagate gradients): $\theta \leftarrow \theta - \lambda \nabla L(\theta) = \theta - \lambda/N \sum_{i}^{N} \nabla L_i(\theta)$

We will (need to) revisit the benefits and limits of this optimization perspective later in the course, but for now assume it as a standard

Universität Bremen

Question Time

Do you see any challenges arising from such an optimization based framing?

Intuitive summary: the static ML workflow

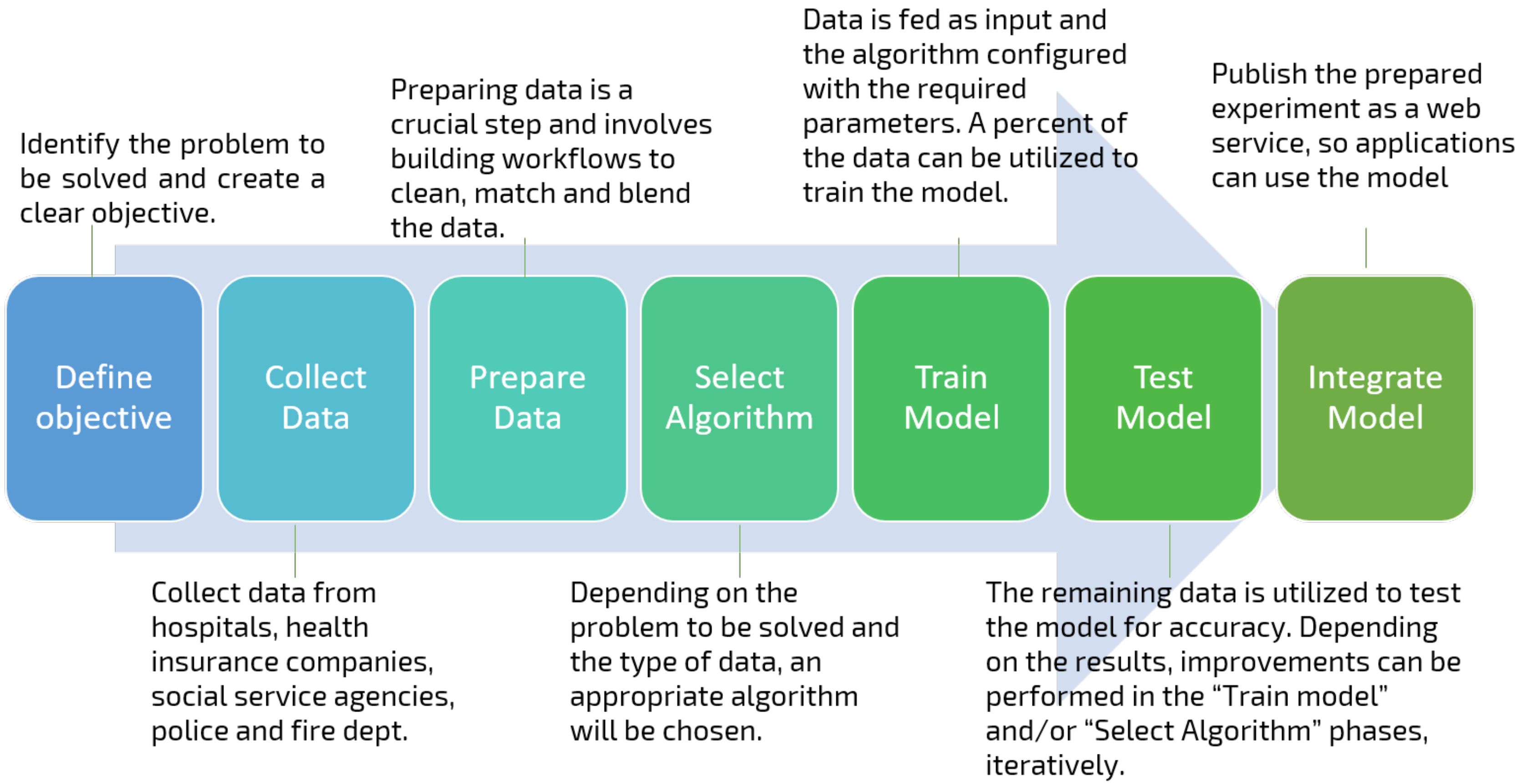
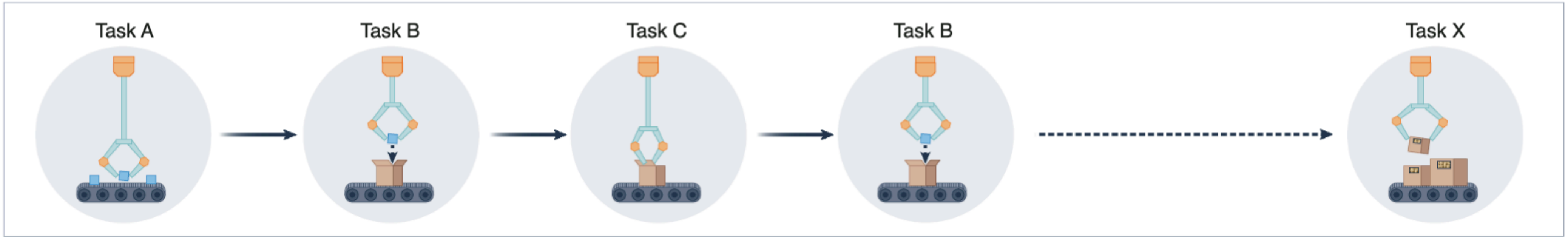


Figure from https://www.congrelate.com/get-workflow-machine-learning-images/

What if we want to continue learning?

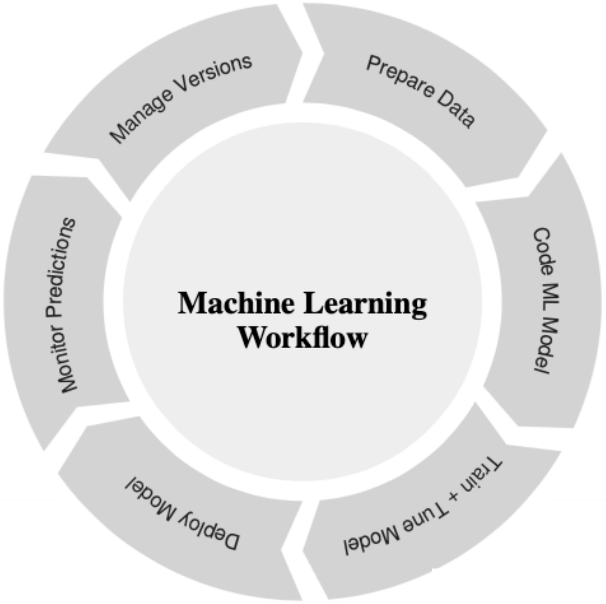


How do we identify new tasks, add more categories, learn multiple tasks, change the model structure, order data, distinguish known from unknown concepts, ensure learning efficiency, maintain knowledge over time … ?

Kudithipudi et al, “Biological underpinnings for lifelong learning machines”, Nature Machine Intelligence (4), 2022

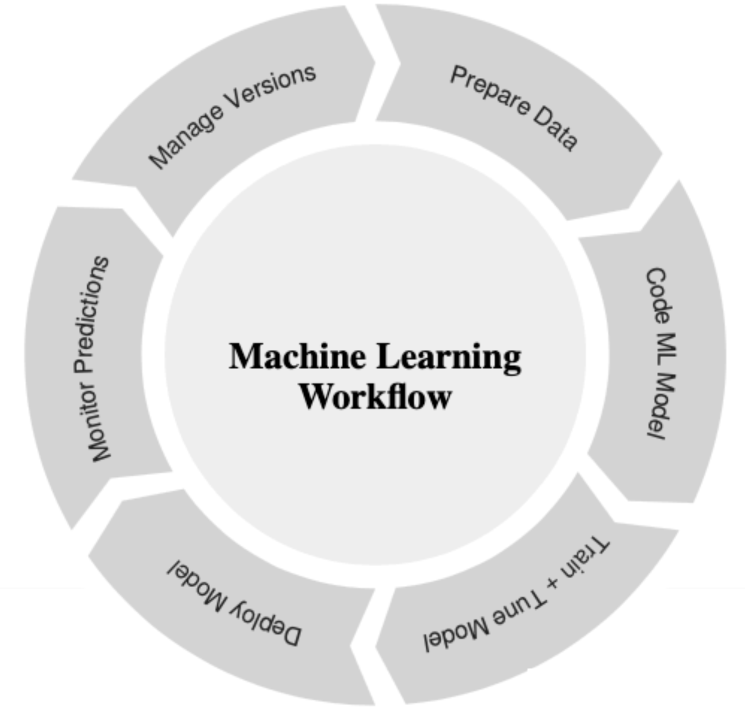
Question Time

Can we turn the static workflow into a “circle”?



Why is a lifelong ML workflow hard?

Yes! And there are many reasons why we should move in that direction!



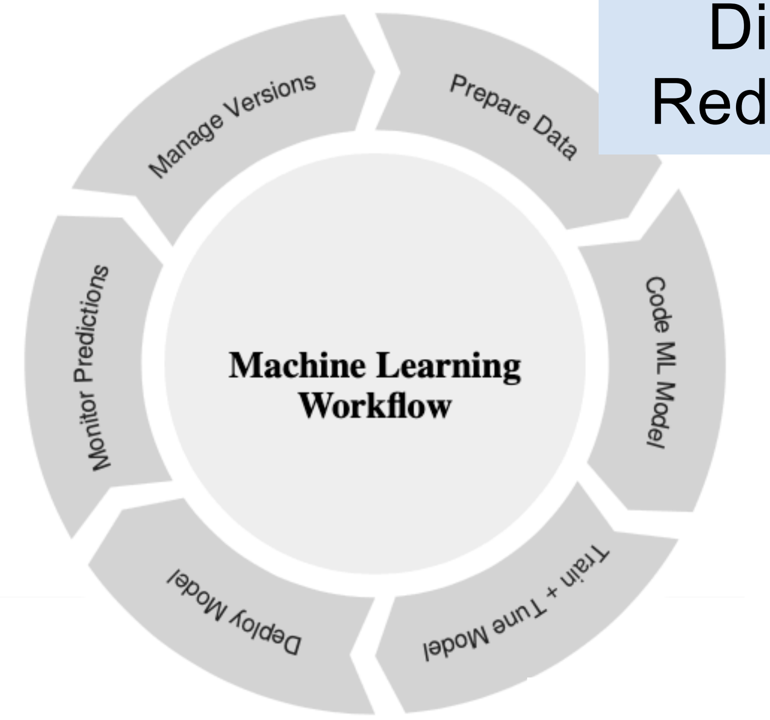
But, it will also turn out that this is MUCH harder than perhaps expected

Mundt et al, "CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability", ICLR 2022

Why is a lifelong ML workflow hard?
Light: “Static” - Dark: “Continual” questions

Data: Amount? Diversity? Redundancy?

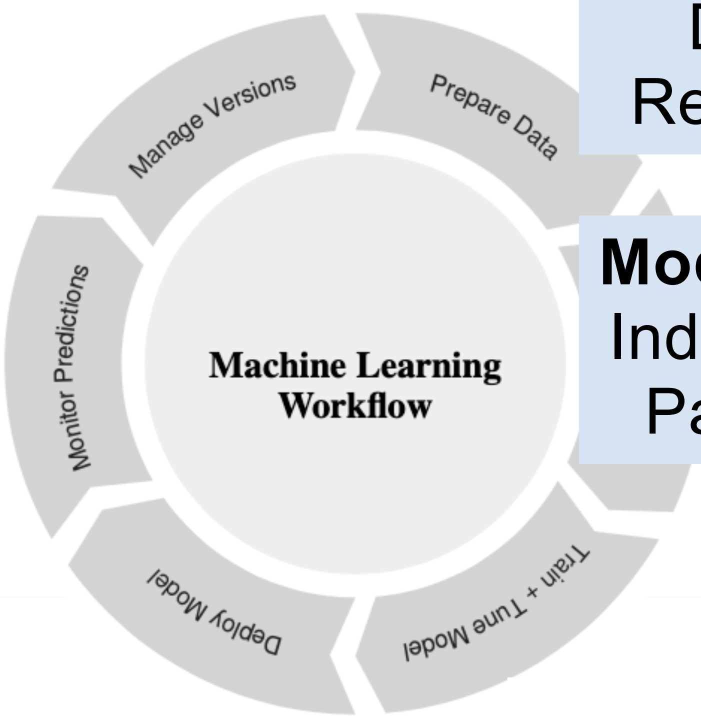
Selection? Ordering? Shift? Noise?



Mundt et al, “CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability”, ICLR 2022

Universität Bremen

Why is a lifelong ML workflow hard?
Light: “Static” - Dark: “Continual” questions

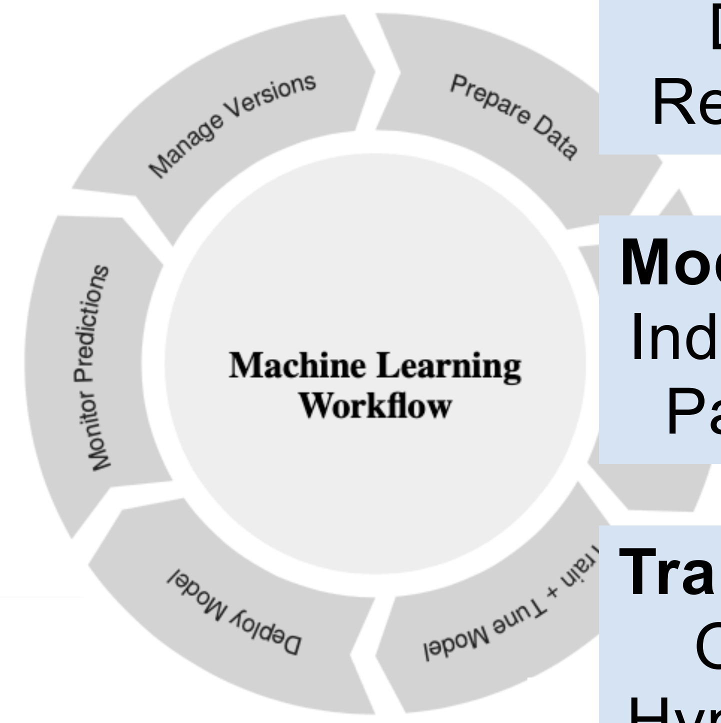


Data: Amount? Diversity? Redundancy?	Selection? Ordering? Shift? Noise?
Model: Choice? Inductive Bias? Parameters?	Extensions? Task-specificity of parameters?

Mundt et al, “CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability”, ICLR 2022

Universität Bremen

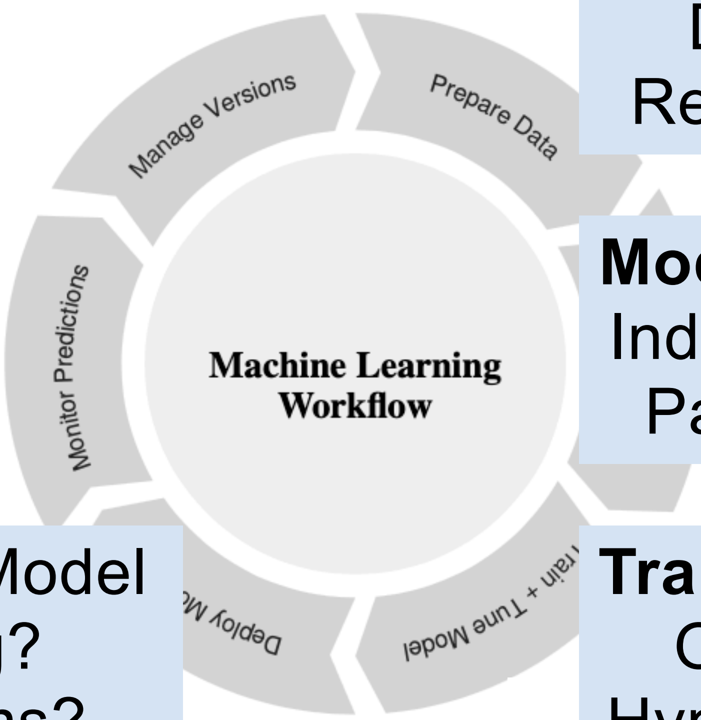
Why is a lifelong ML workflow hard?
Light: “Static” - Dark: “Continual” questions



Data: Amount? Diversity? Redundancy?	Selection? Ordering? Shift? Noise?
Model: Choice? Inductive Bias? Parameters?	Extensions? Task-specificity of parameters?
Training: Loss? Optimizer? Hyper-params?	Forgetting? Transfer? Partial Updates?

Mundt et al, “CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability”, ICLR 2022

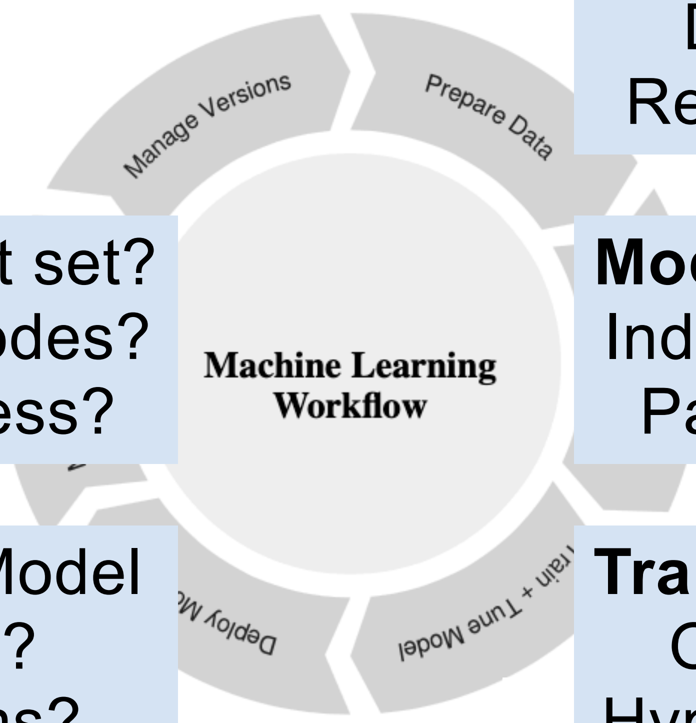
Why is a lifelong ML workflow hard?
Light: “Static” - Dark: “Continual” questions



Static	Continual
Data: Amount? Diversity? Redundancy?	Selection? Ordering? Shift? Noise?
Model: Choice? Inductive Bias? Parameters?	Extensions? Task-specificity of parameters?
Training: Loss? Optimizer? Hyper-params?	Forgetting? Transfer? Partial Updates?
Deploy: Model saving? Platforms?	Optim. States? Distr. updates? Meta-data?

Mundt et al, “CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability”, ICLR 2022

Why is a lifelong ML workflow hard?
Light: “Static” - Dark: “Continual” questions



Data: Amount? Diversity? Redundancy?

Selection? Ordering? Shift? Noise?

Evolving test? Perturbations? Unknown input?

Pred.: Test set? Failure modes? Robustness?

Model: Choice? Inductive Bias? Parameters?

Extensions? Task-specificity of parameters?

Optim. States? Distr. updates? Meta-data?

Deploy: Model saving? Platforms?

Training: Loss? Optimizer? Hyper-params?

Forgetting? Transfer? Partial Updates?

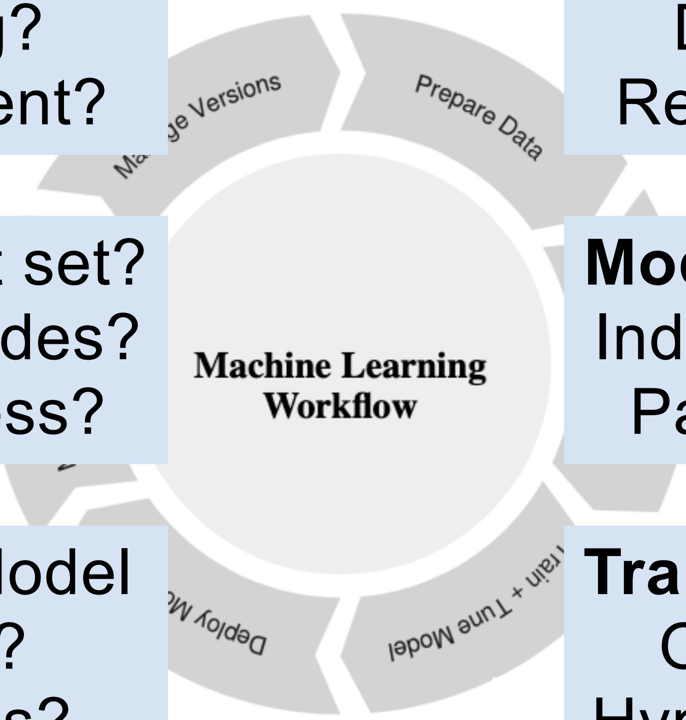
Mundt et al, “CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability”, ICLR 2022

Universität Bremen

Why is a lifelong ML workflow hard?
Light: “Static” - Dark: “Continual” questions

Discretization? Backward compatibility?

Versioning: Staging? Deployment?



Data: Amount? Diversity? Redundancy?

Selection? Ordering? Shift? Noise?

Evolving test? Perturbations? Unknown input?

Pred.: Test set? Failure modes? Robustness?

Model: Choice? Inductive Bias? Parameters?

Extensions? Task-specificity of parameters?

Optim. States? Distr. updates? Meta-data?

Deploy: Model saving? Platforms?

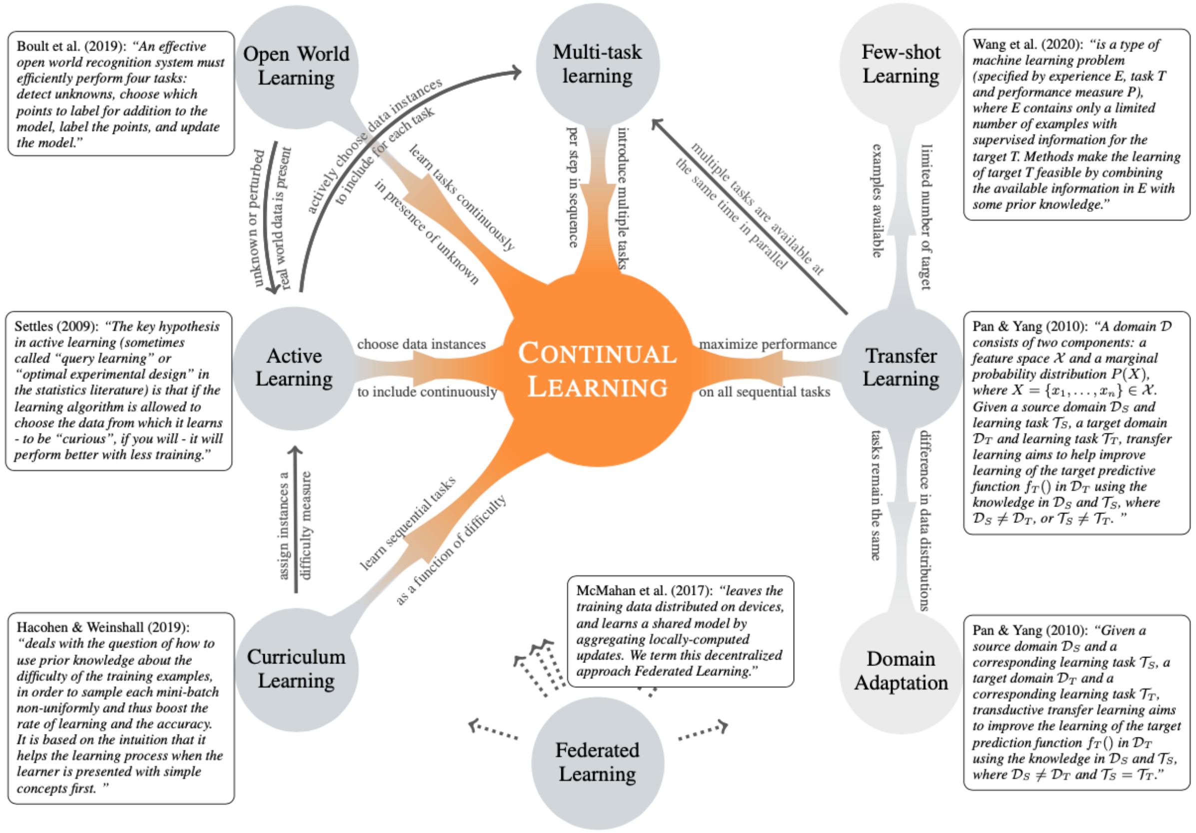
Training: Loss? Optimizer? Hyper-params?

Forgetting? Transfer? Partial Updates?

Mundt et al, “CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability”, ICLR 2022

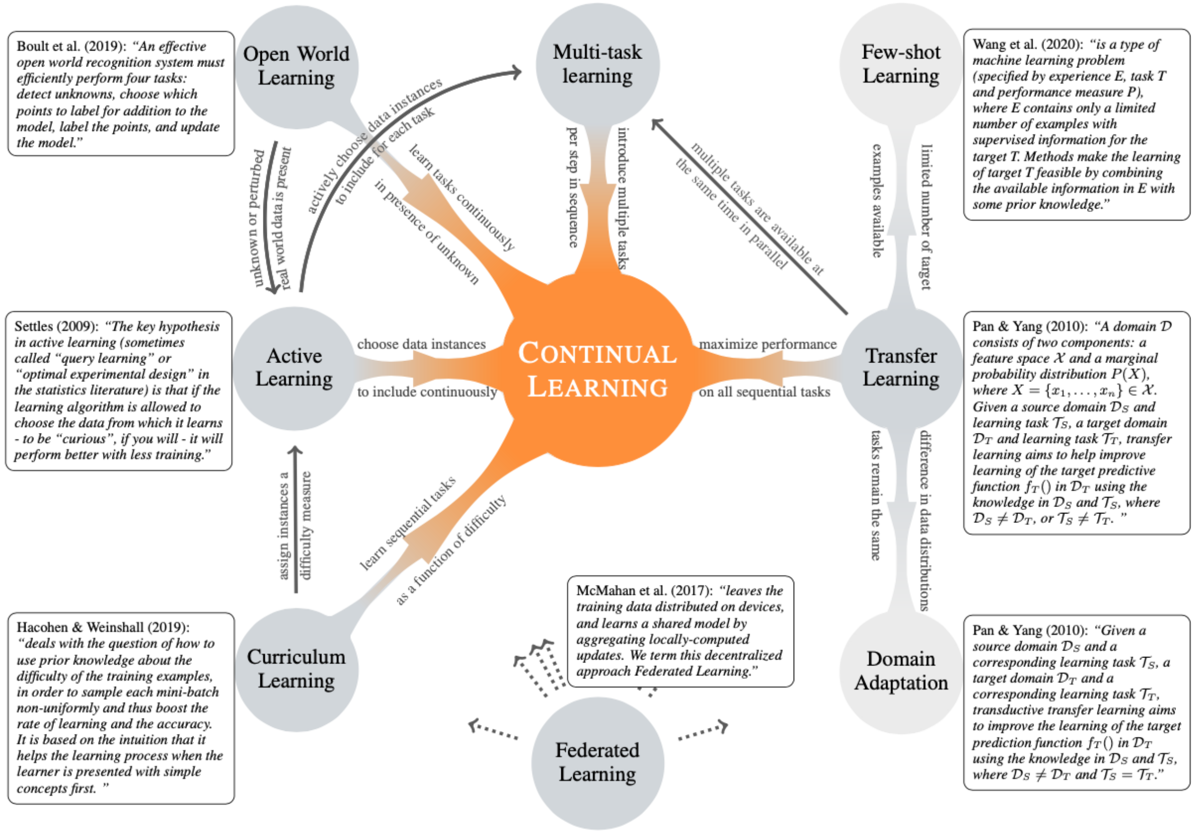
Universität Bremen

Summary of course content



Mundt et al, "CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability", ICLR 2022

Summary of course content



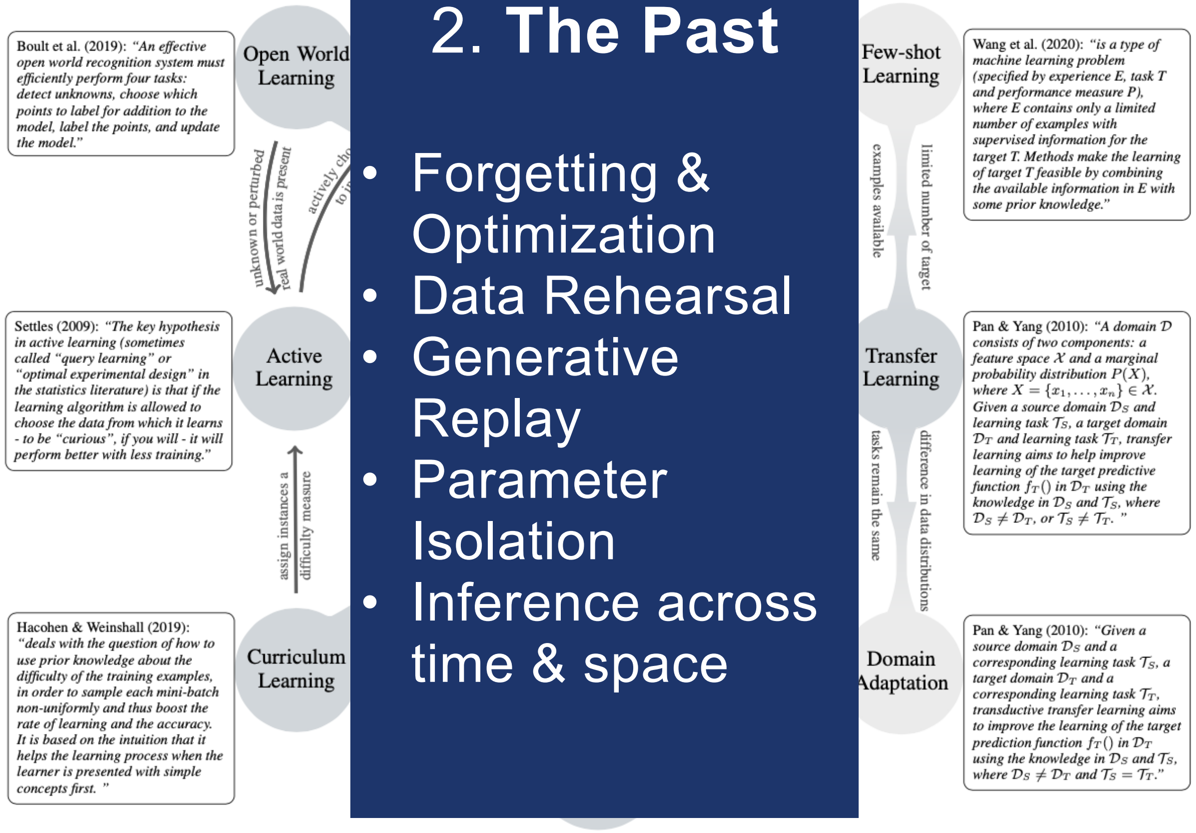
1. The Present

- Data Difficulty & Learning Pace
- Adaptive Curricula
- Transfer Learning & Domain Adapt.
- Transfer in Deep Neural Networks

Mundt et al, "CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability", ICLR 2022

Universität
Bremen

Summary of course content



2. The Past

- Forgetting & Optimization
- Data Rehearsal
- Generative Replay
- Parameter Isolation
- Inference across time & space

1. The Present

- Data Difficulty & Learning Pace
- Adaptive Curricula
- Transfer Learning & Domain Adapt.
- Transfer in Deep Neural Networks

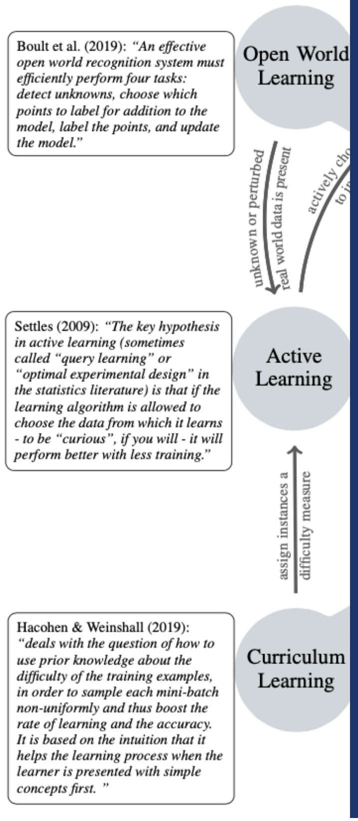
Mundt et al, "CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability", ICLR 2022

Universität Bremen

Summary of course content

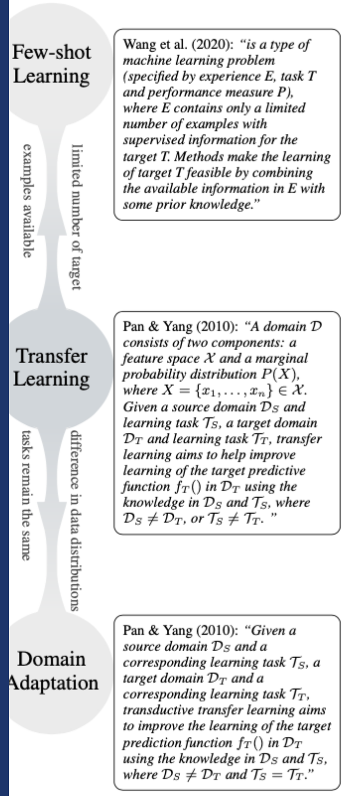
3. The Future

- Active Data Queries
- Gauging Data Informativeness
- Learning & Unknowns
- Open World Learning



2. The Past

- Forgetting & Optimization
- Data Rehearsal
- Generative Replay
- Parameter Isolation
- Inference across time & space



1. The Present

- Data Difficulty & Learning Pace
- Adaptive Curricula
- Transfer Learning & Domain Adapt.
- Transfer in Deep Neural Networks

Mundt et al, "CLEVA-Compass: A Continual Learning Evaluation Assessment Compass to Promote Research Transparency and Comparability", ICLR 2022