RUB

RUHR-UNIVERSITÄT BOCHUM

UNCERTAINTY ESTIMATION IN FEDERATED DEEP LEARNING

Florian Linsner

Table of Contents

Chapter
Mathematical Basics

Entropy and Variance

- Shannon Entropy over output classes

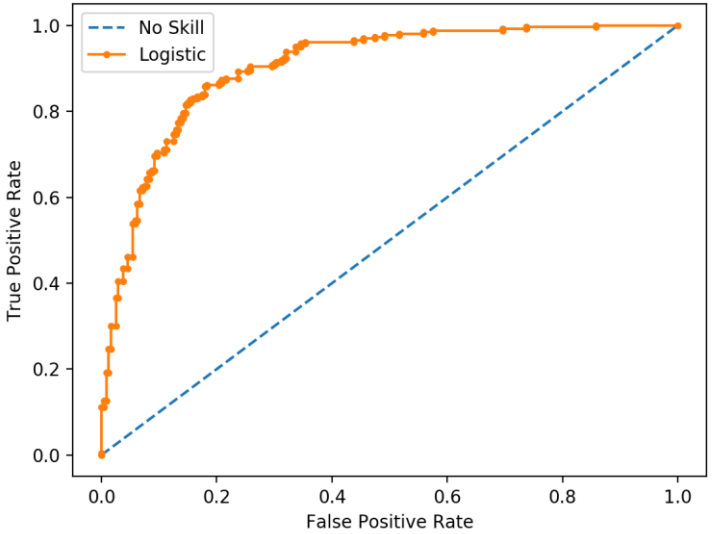
$$H\left(p\left(Y|x^*,\theta\right)\right) = -\sum_{k=1}^{K} p\left(y_k|x^*,\theta\right) \cdot \log p\left(y_k|x^*,\theta\right)$$

- Variance over multiple predictions for a fixed input

$$\sigma^2 = \frac{1}{S}\sum_{s=1}^{S} p\left(y|x^*,\theta_s\right)^2 - p\left(y|x^*\right)^2$$

ROC and AUROC

- Receiver Operating Characterisitc Curve
  - Binary classifier
  - 2 groups of data
  - True Positive vs False Positive
  - Multiple Theshholds
- Area Under ROC
  - Score between 0.5 and 1
  - Want to achieve a score of 1



Uncertainty Estimation in Federated Deep Learning
Image: https://machinelearningmastery.com/

Chapter
Uncertainty Theorie

Importance of Uncertainty

- We use models to make predictions
  - Can we trust these predictions?
- There is no perfect model
  - No-Free-Lunch-Theorem
  - Models make mistakes!
- Many Decisions in critical environments
  - Autonomous Driving
  - Nuclear Powerplants
  - Industrial Settings

Malicious Intentions

- An Attacker can Influence the beaviour of a model
- Data Poisoning
- Aversarial Examples
- Many other

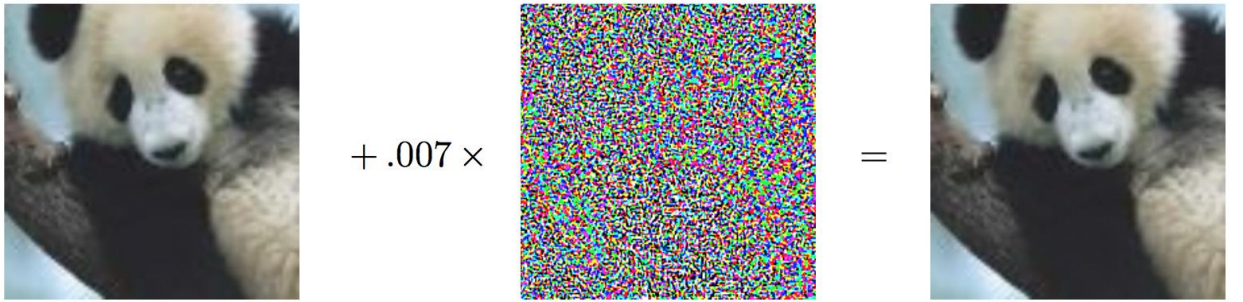


Image: https://pytorch.org/tutorials/_images/fgsm_panda_image.png

Bayesian Uncertainty - Origins

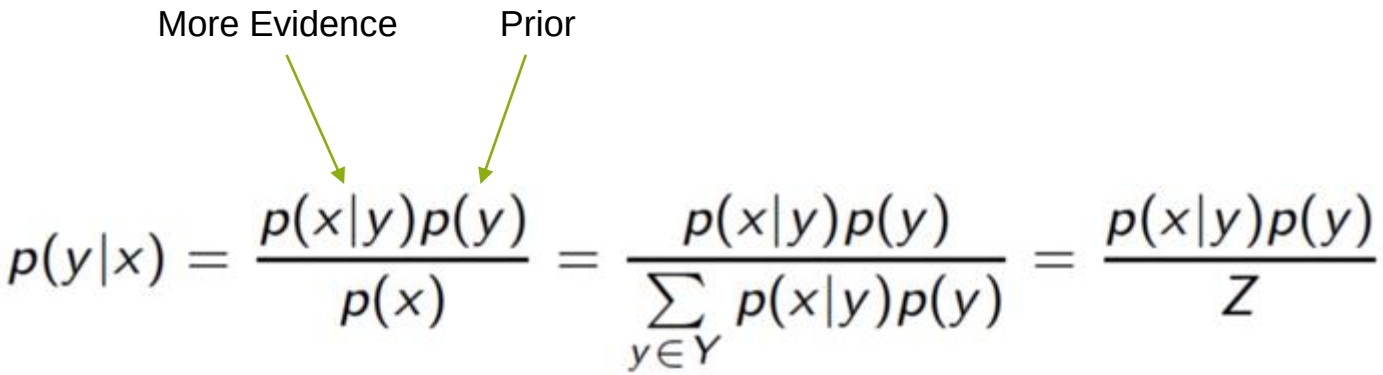
$$p(y|x) = \frac{p(x|y)p(y)}{p(x)} = \frac{p(x|y)p(y)}{\sum_{y \in Y} p(x|y)p(y)} = \frac{p(x|y)p(y)}{Z}$$

Bayesian Uncertainty - Origins

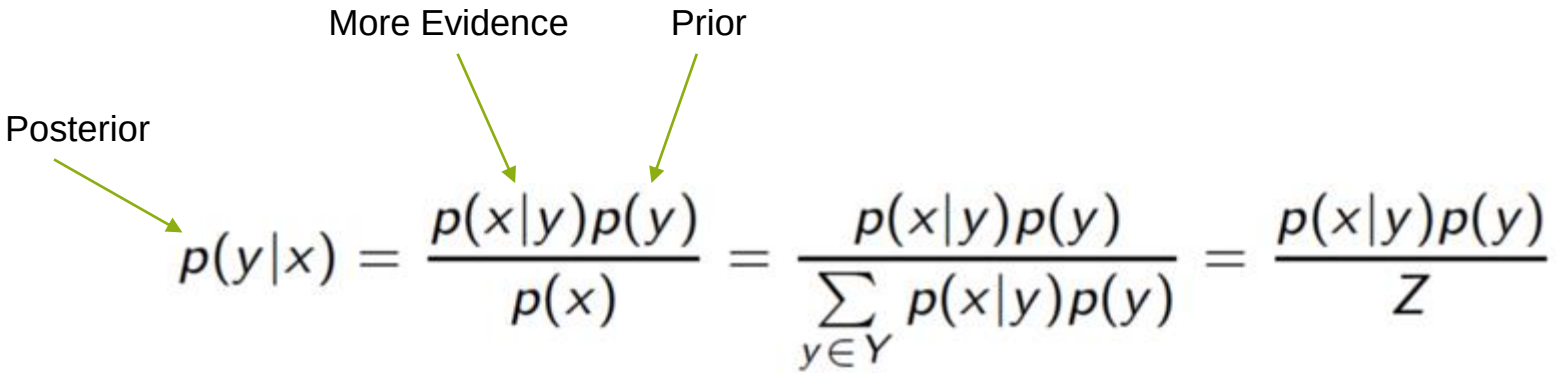
Prior

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)} = \frac{p(x|y)p(y)}{\sum_{y \in Y} p(x|y)p(y)} = \frac{p(x|y)p(y)}{Z}$$

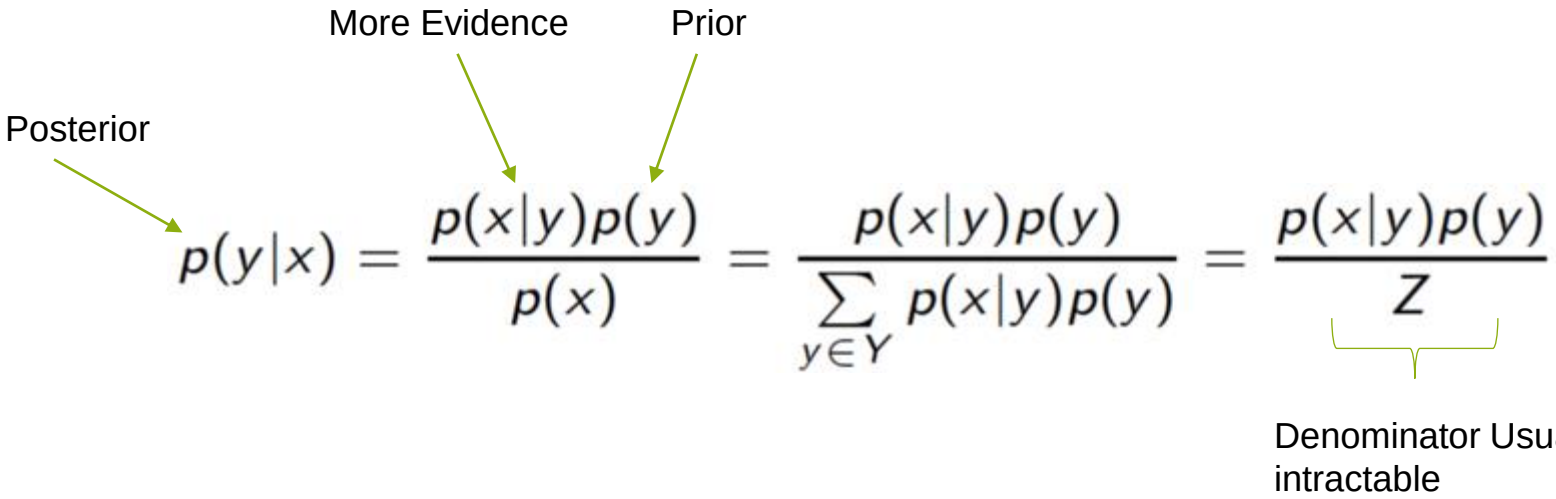
Bayesian Uncertainty - Origins



Bayesian Uncertainty - Origins



Bayesian Uncertainty - Origins



Bayesian Uncertainty - Usage

- Stating priors over model parameters β
- Using training data (X, y)

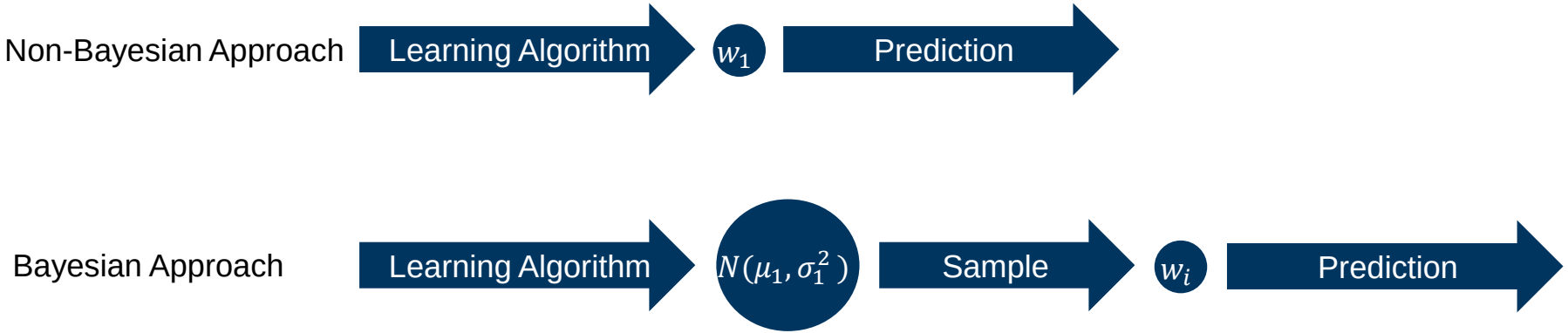
Bayesian Uncertainty - Usage

- Stating priors over model parameters β
- Using training data (X, y)

$$p(\beta|y_t, X_t) = \frac{p(y_t, X_t|\beta)p(\beta)}{p(y_t, X_t)} = \frac{p(y_t, X_t|\beta)p(\beta)}{\int p(y_t, X_t|\beta)p(\beta)d\beta}$$

- Solving analytically is often impossible/impractical
- Solution: Numeric Approximations

Bayesian Uncertainty - Usage

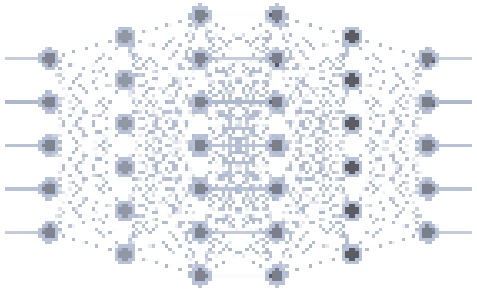


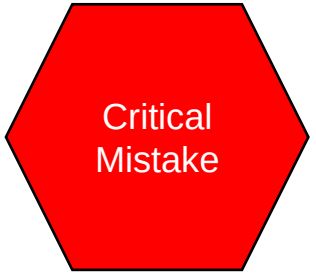
Chapter

Methods for Uncertainty

Ensembles

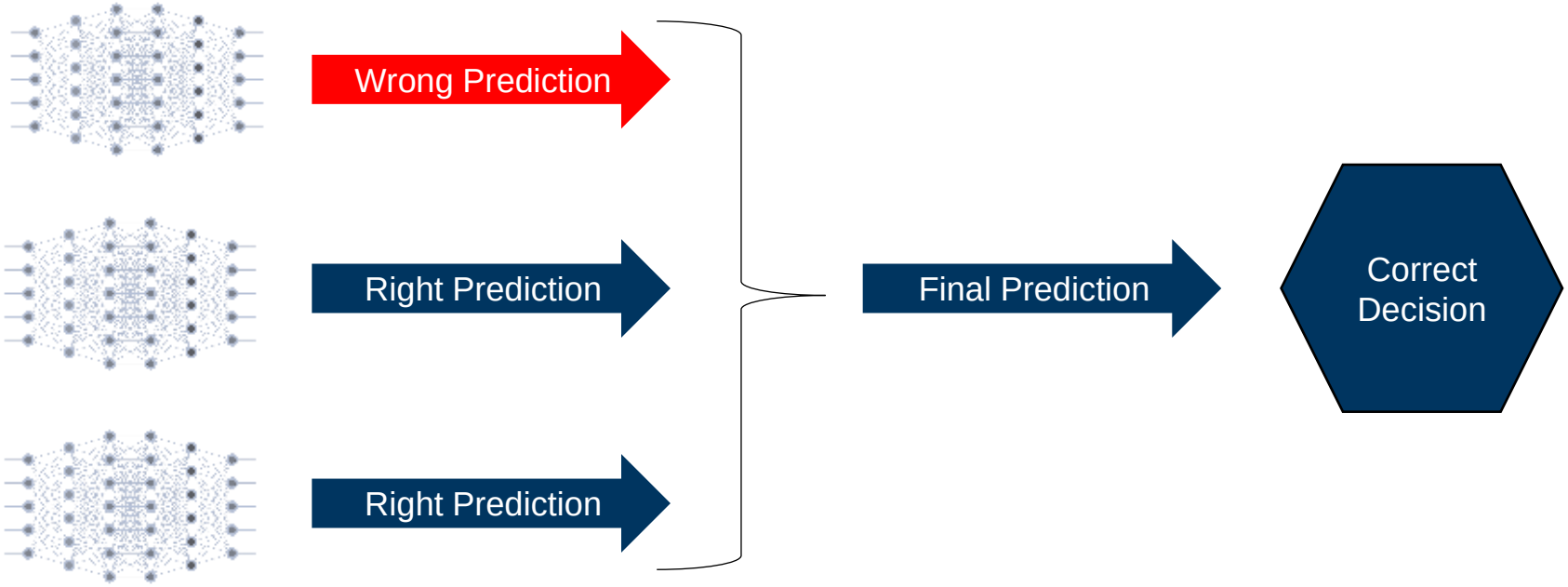
- Originally proposed as non-Bayesian method
- Can be viewed as Bayesian Approximation[1]



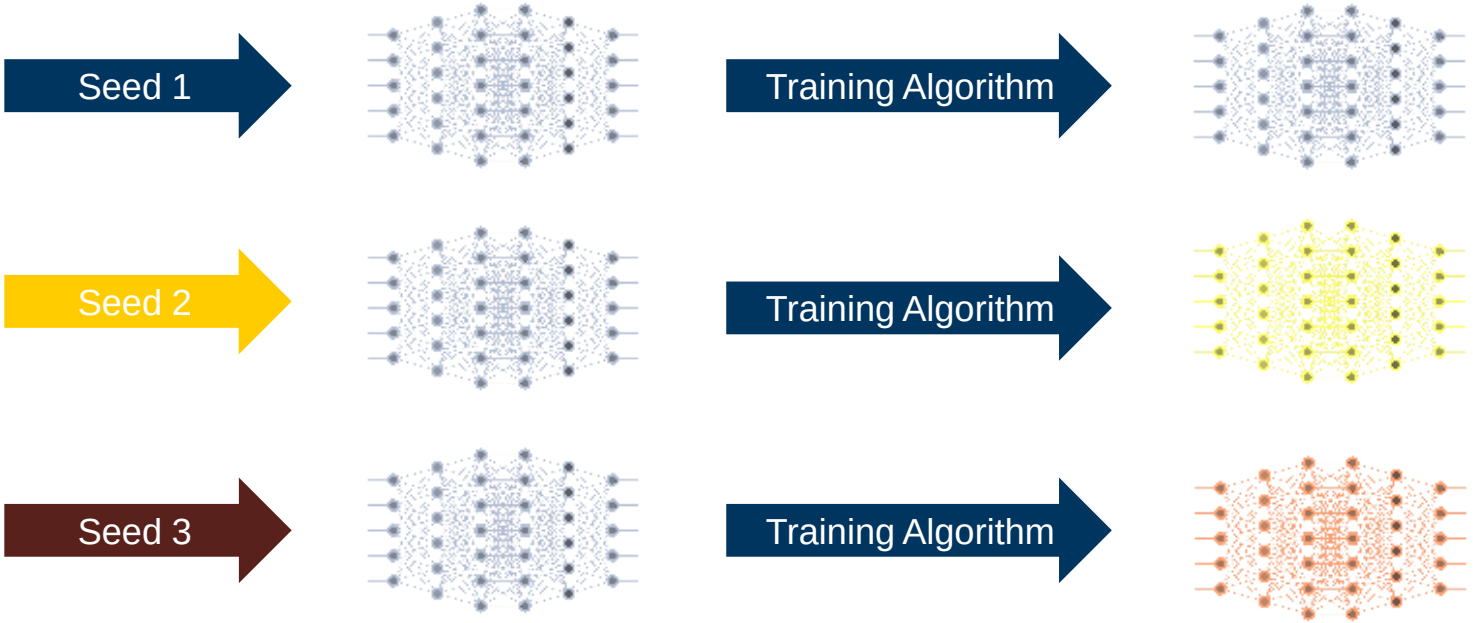


[1]Hoffman et. Al. Deep Ensembles from a Bayesian Perspectiv

Ensembles

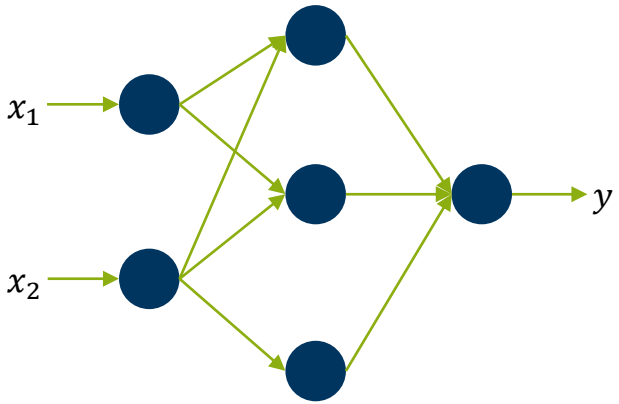

Wrong Prediction
Right Prediction
Right Prediction
Final Prediction
Correct Decision

Ensemble Training



Dropout

- Method for Regularization during Training Time
- Prevents Overfitting

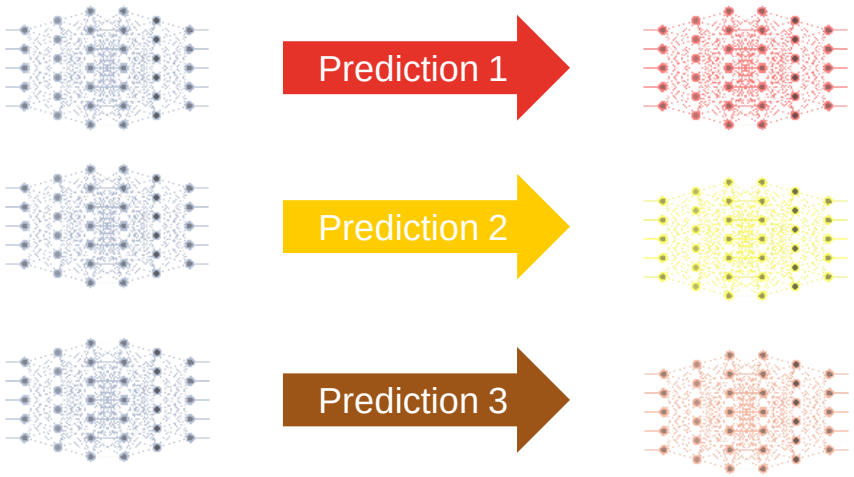


Dropout

- Method for Regularization during Training Time
- Prevents Overfitting

MC-Dropout

- MC-Dropout as Bayesian Approximation[2]
- Dropout During Prediction Time
- Multiple Predictions for same Input



Uncertainty Estimation in Federated Deep Learning

[2] Gal et. Al. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning

SWAG

- Stochastic Weight Averaging Gaussian[3]
- Needs Pre-Trained Solution near a local minima
- Compute a running average

$$\theta_{\text{SWA}} = \frac{1}{T}\sum_{i=1}^{T}\theta_i$$

- Compute squared running average

$$\overline{\theta^2} = \frac{1}{T}\sum_{i=1}^{T}\theta_i^2$$

[3]Maddox et. Al. A Simple Baseline for Bayesian Uncertainty in Deep Learning

SWAG

- Compute Covariance

$$\Sigma_{\text{diag}} = \text{diag}(\overline{\theta^2} - \theta^2_{\text{SWA}})$$

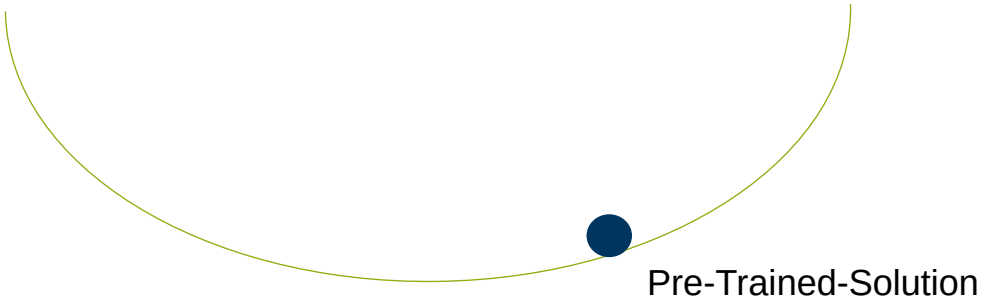
- Sample from Distribution

$$\mathcal{N}(\theta_{\text{SWA}}, \Sigma_{\text{Diag}})$$

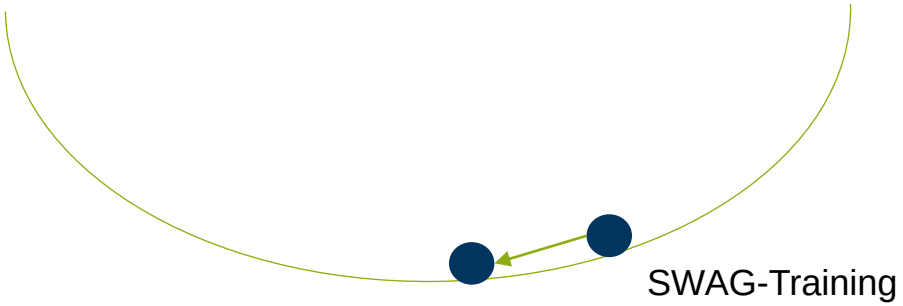
SWAG in Pictures

Loss-Surface

SWAG in Pictures

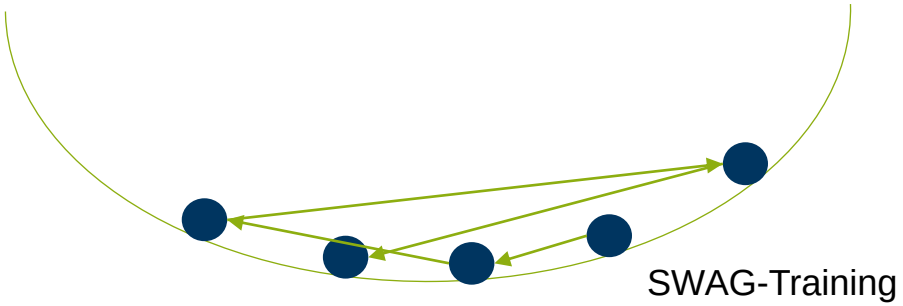


SWAG in Pictures



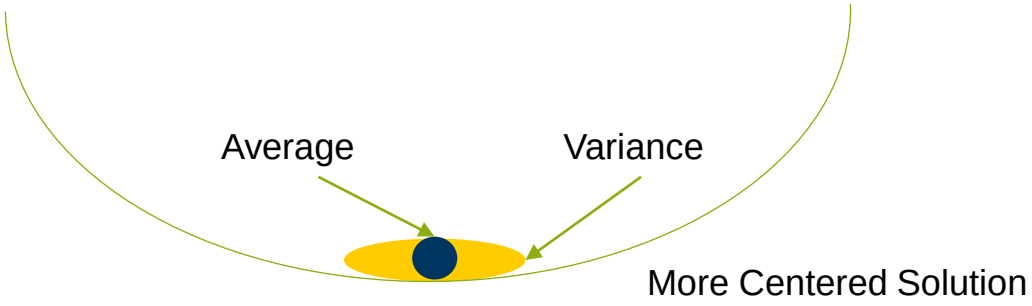
SWAG in Pictures

- Goal: More Centered Solution



SWAG in Pictures

- Goal: More Centered Solution



Chapter

Challenges in Centralized Deep Learning

Data Generating Devices

- $50.1 \cdot 10^9$connected devices by 2020[4]
  - (autonomous) driving and vehicles
  - Smart sensors
  - Mobile phones
  - Airplanes
  - Entertainment Devices
  - Industrial Machines
  - …
- IoT sector is growing exponentially

[4] Nitinder Mohan and Jussi Kangasharju. *Edge-fog cloud: A distributed cloud for internet of things computations*. In Cloudification of the Internet of Things (CIoT), pages 1–6. IEEE, 2016.

Centralized Machine Learning

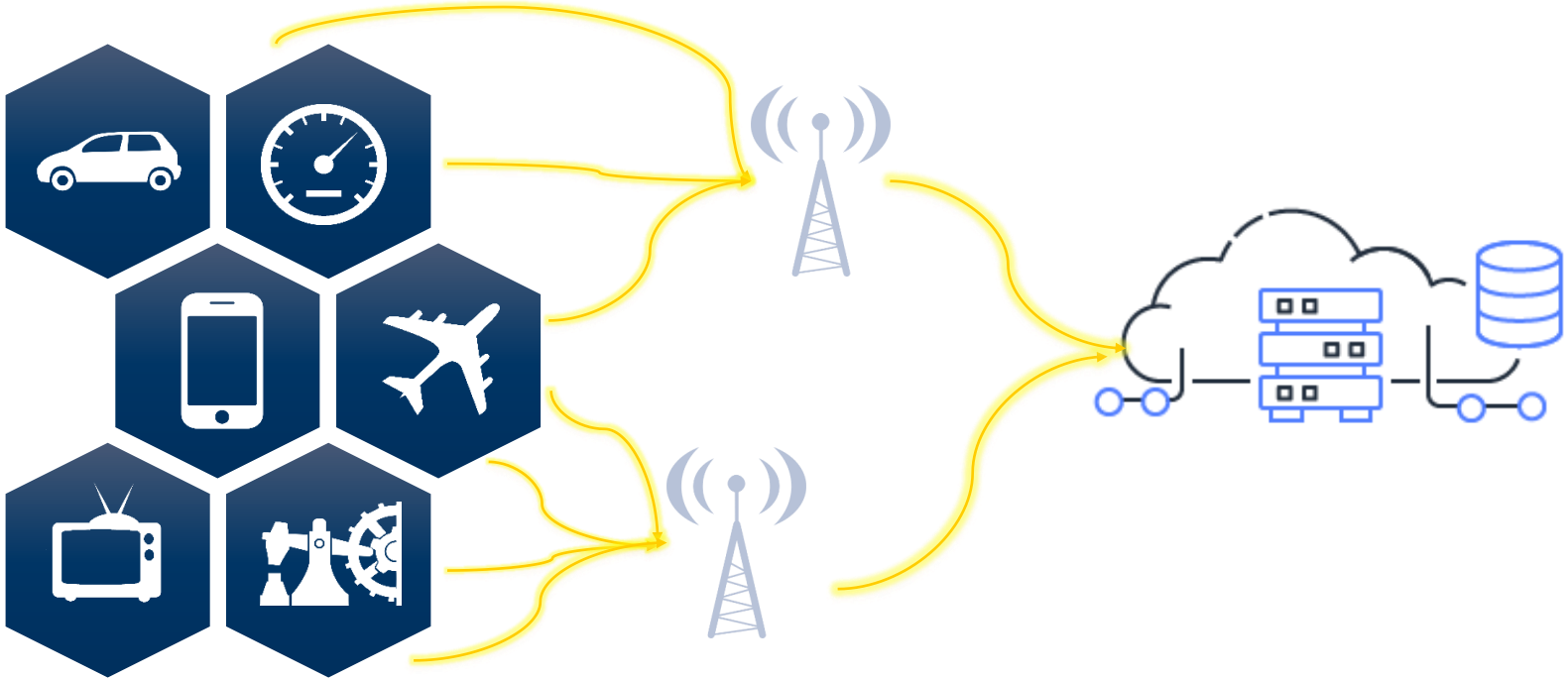
Uncertainty Estimation in Federated Deep Learning

Image: https://d1.awsstatic.com/what-is-aws/AWS-Types-of-Cloud-Computing_Infrastructure-as-a-Service.c2e061a42b68ec6d969880fb8768bcb8e5cc2e69.png

Centralized Machine Learning

Uncertainty Estimation in Federated Deep Learning
Image: https://d1.awsstatic.com/what-is-aws/AWS-Types-of-Cloud-Computing_Infrastructure-as-a-Service.c2e061a42b68ec6d969880fb8768bcb8e5cc2e69.png

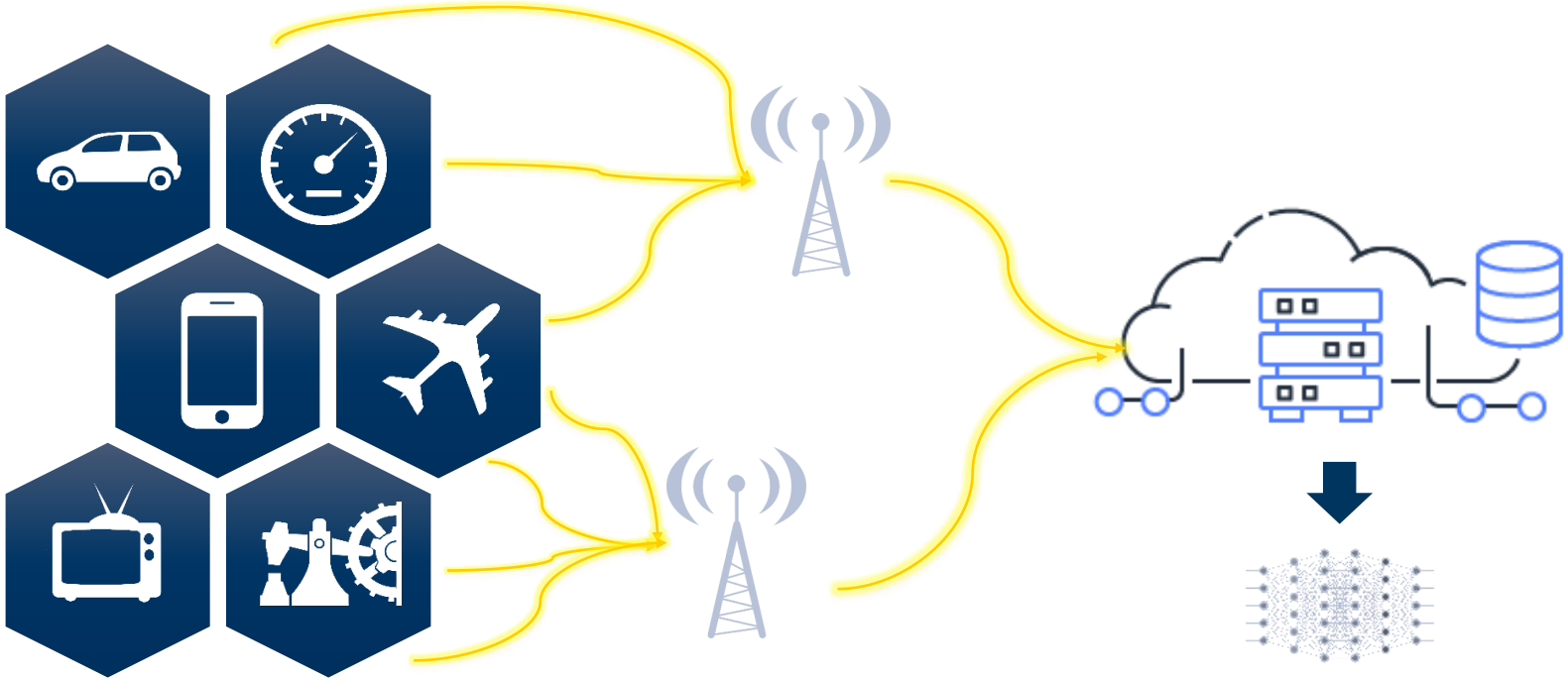
Centralized Machine Learning



Uncertainty Estimation in Federated Deep Learning

Image: https://d1.awsstatic.com/what-is-aws/AWS-Types-of-Cloud-Computing_Infrastructure-as-a-Service.c2e061a42b68ec6d969880fb8768bcb8e5cc2e69.png

Centralized Machine Learning



Uncertainty Estimation in Federated Deep Learning

Image: https://d1.awsstatic.com/what-is-aws/AWS-Types-of-Cloud-Computing_Infrastructure-as-a-Service.c2e061a42b68ec6d969880fb8768bcb8e5cc2e69.png

Limits of Cloud Processing

- Autonomous vehicle generates 1GB per second[5]
  - 4G mobile: 37.5 MB per second
  - VW sold $10.8 \cdot 10^6$ cars in 2018
  - Produce 10.3 PB data per second
  - All CERN experiments process 25 GB per second[6]

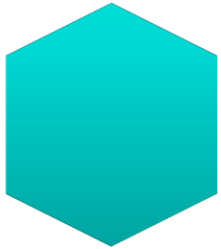
Uncertainty Estimation in Federated Deep Learning
[5]Shi, Weisong et al. *Edge computing: Vision and challenges*. Internet of Things Journal, pages 637–646. IEEE, 2016
[6]CERN: Processing: What to record? (URL: https://home.cern/science/computing/processing-what-record), retrieved 27.3.2019.

Limits of Cloud Processing

- Siemens collects 2EB data per day[7]
  - Sensor data can reveal corporate secrets (e.g., machine settings)
  - Predictive maintenance models usually trained on all data[8]

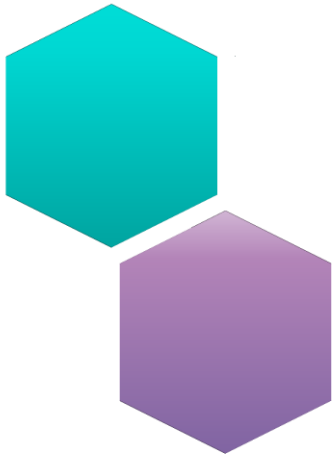
[7]Rüdiger Köhn. *Ringen um die Vorherrschaft über Industrie 4.0*. Blick in die Zukunft: Trends und Szenarien für die Welt von morgen. FAZ, 2014.
[8] Lee, Jay, Hung-An Kao, and Shanhu Yang. *Service innovation and smart analytics for industry 4.0 and big data environment*. In CIRP, pages 3-8, 2014.

Centralized Learning Approach



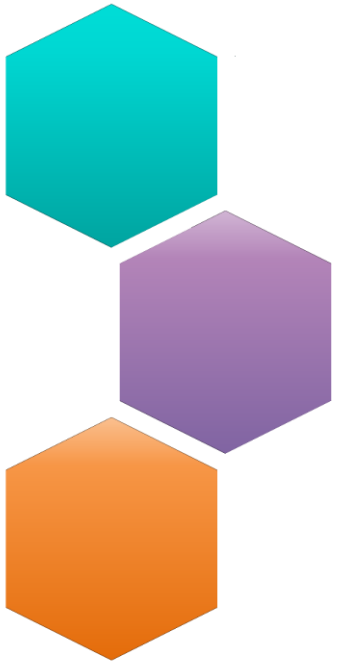
- does not scale with number of devices and neglects their computing power

Centralized Learning Approach



- does not scale with number of devices and neglects their computing power
- prohibitive communication costs

Centralized Learning Approach



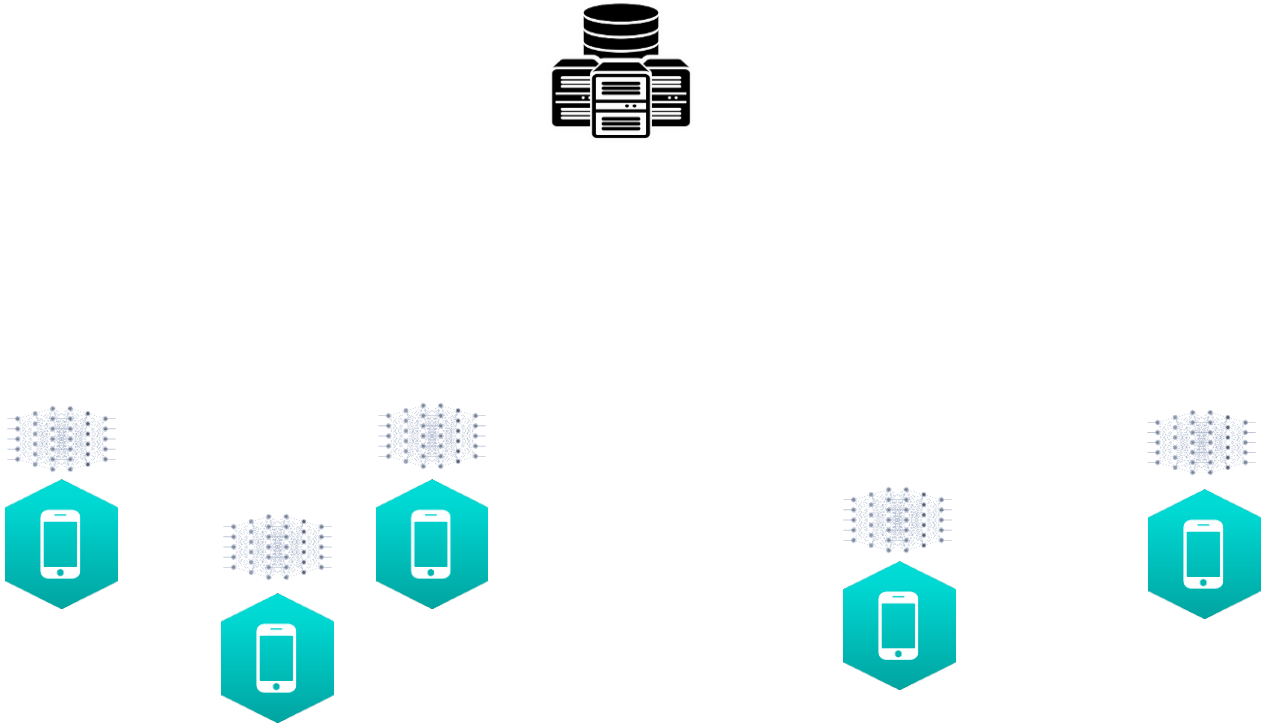
- does not scale with number of devices and neglects their computing power
- prohibitive communication costs
- requires sharing of privacy-sensitive data

Chapter
Federated Deep Learning

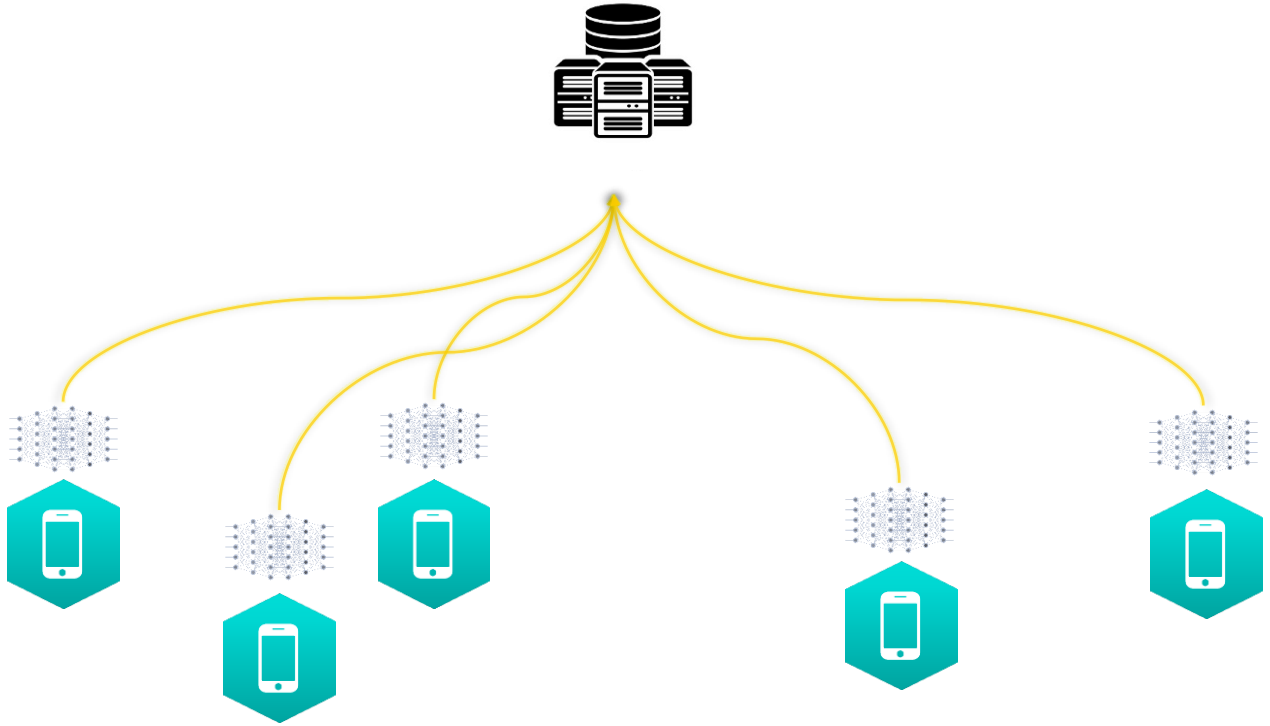
Federated Deep Learning Approach

Federated Deep Learning Approach

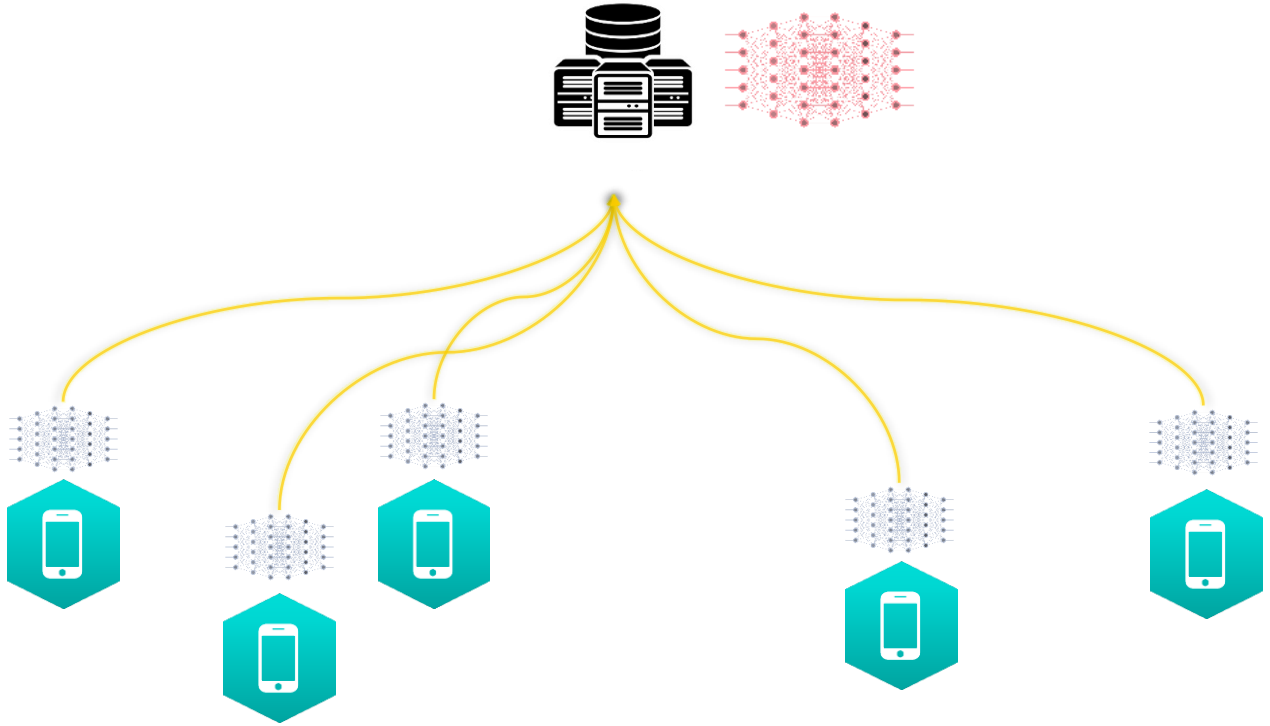
Federated Deep Learning Approach



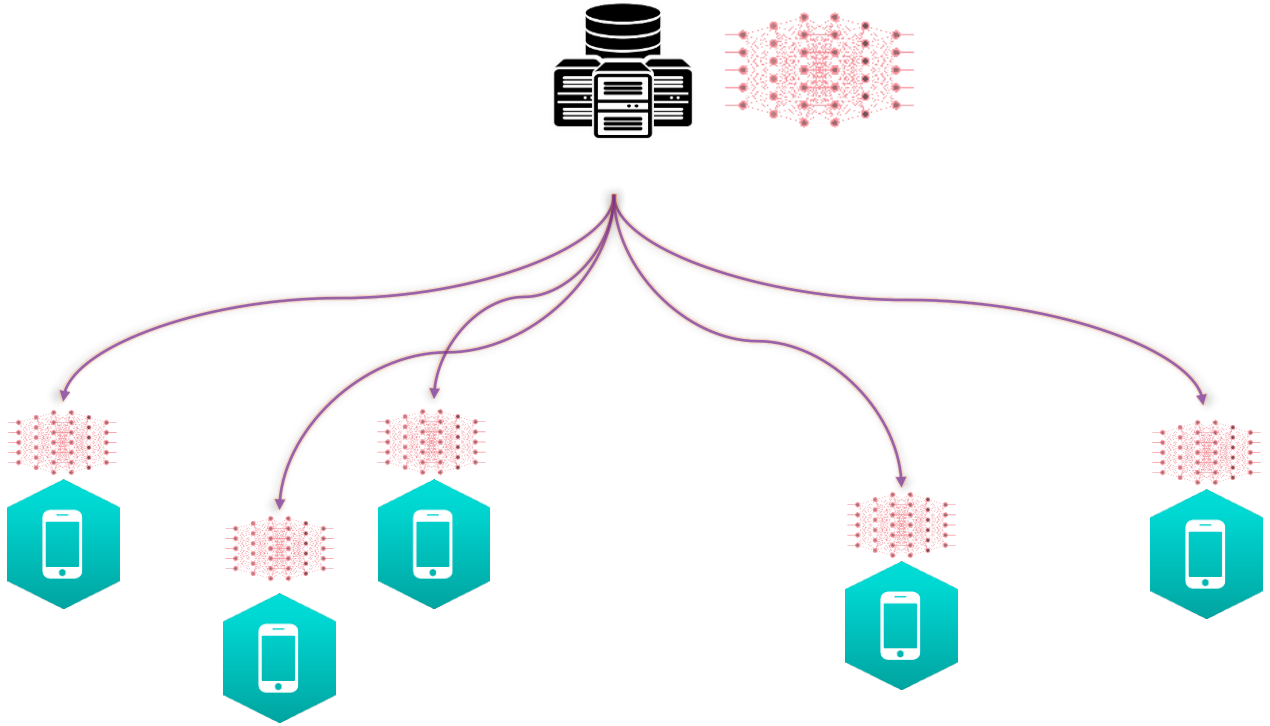
Federated Deep Learning Approach



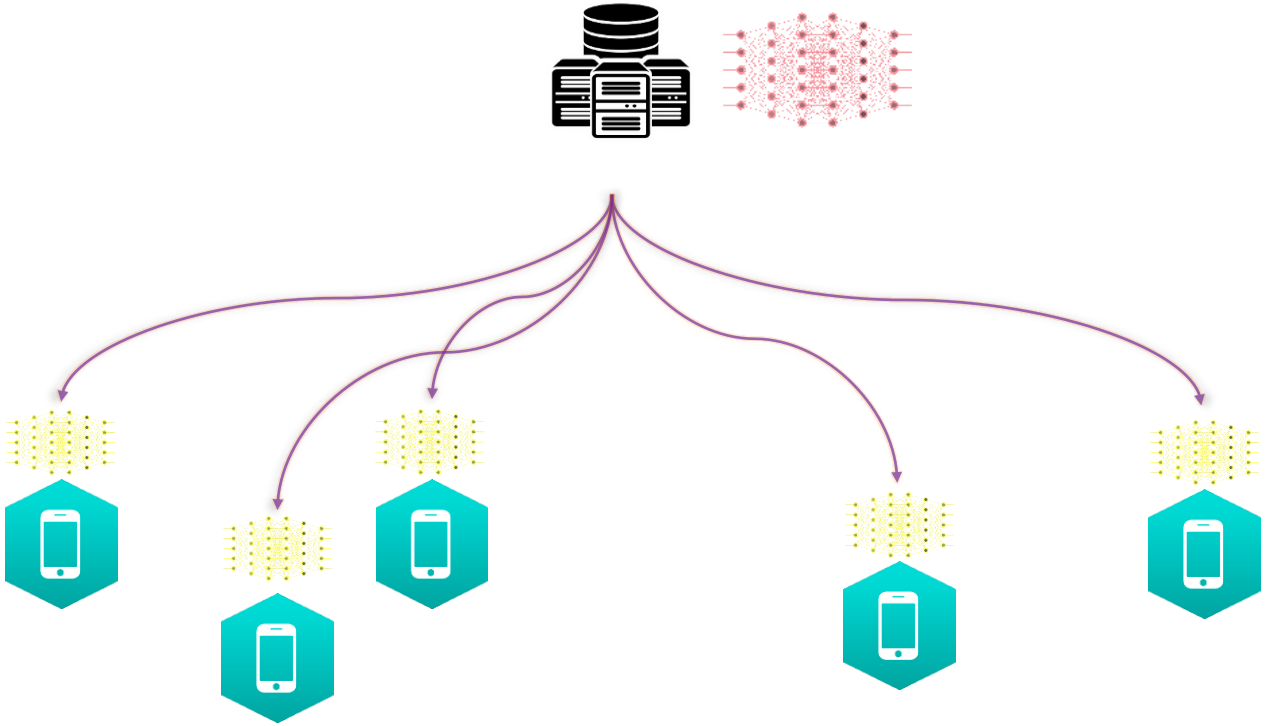
Federated Deep Learning Approach



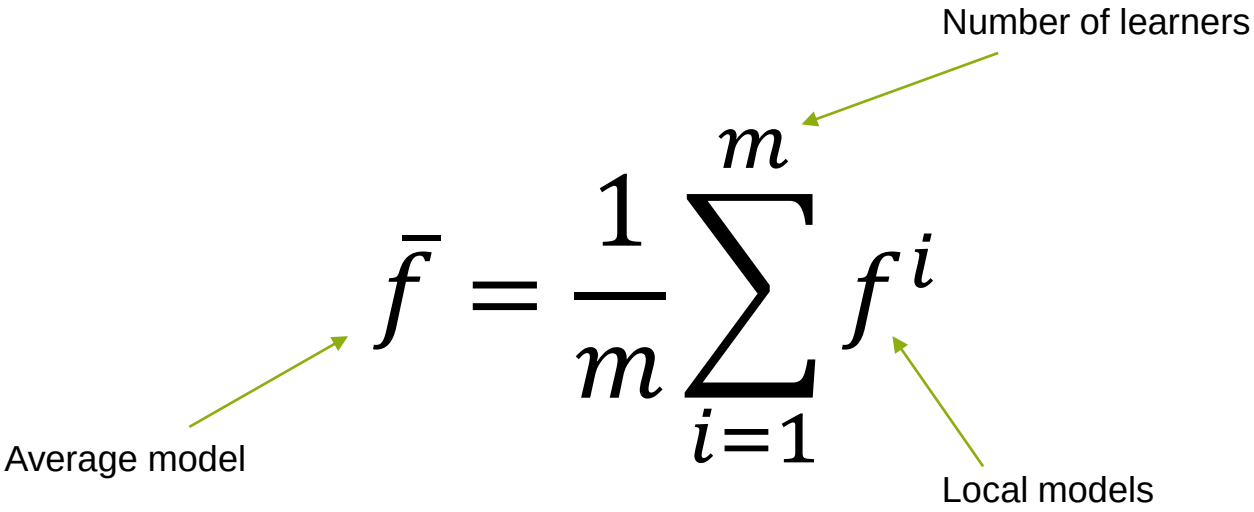
Federated Deep Learning Approach



Federated Deep Learning Approach



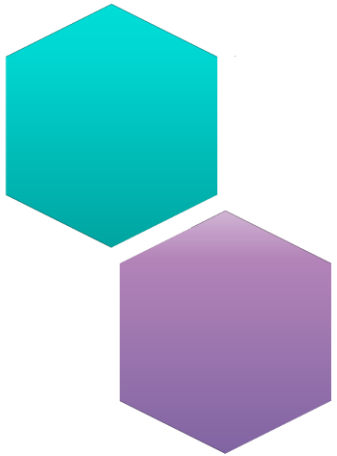
Model Aggregation



Properties

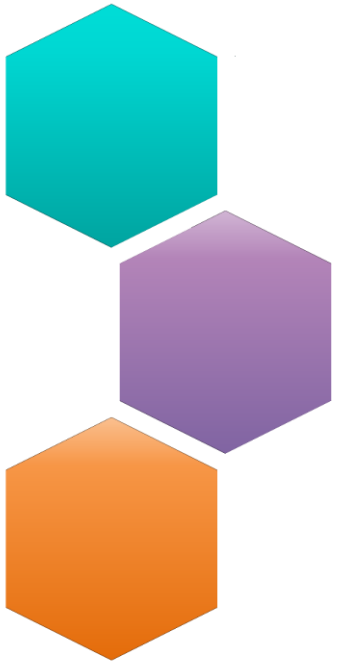
- independent of the learning algorithm (black-box)

Properties



- independent of the learning algorithm (black-box)
- works for large amount of model classes (linear + kernel models, various DNNs)

Properties



- independent of the learning algorithm (black-box)
- works for large amount of model classes (linear + kernel models, various DNNs)
- requires that models can be trained locally, e.g., sufficient local computing power, and labels available

Requirements

1. The Model trained in parallel has a high quality similar to one obtained by a the serial application of the learning algorithm.

Requirements

1. The Model trained in parallel has a high quality similar to one obtained by a the serial application of the learning algorithm.

2. The parallel algorithm achieves a high speedup; ideally it has a polylogarithmic runtime on (quasi-) polynomial many learners.

Requirements

1. The Model trained in parallel has a high quality similar to one obtained by a the serial application of the learning algorithm.

2. The parallel algorithm achieves a high speedup; ideally it has a polylogarithmic runtime on (quasi-) polynomial many learners.

3. The parallel execution requires communication that is almost linear in the number of learners and adaptive to the hardness of the learning problem.

Requirements

1. The Model trained in parallel has a high quality similar to one obtained by a the serial application of the learning algorithm.

2. The parallel algorithm achieves a high speedup; ideally it has a polylogarithmic runtime on (quasi-) polynomial many learners.

3. The parallel execution requires communication that is almost linear in the number of learners and adaptive to the hardness of the learning problem.

Privacy

- Additional inherent Feature
  - No Data is Shared between Parties
  - Only Model Weights are Shared
- Can be proven to be *differential private*[9]
  - By clipping the Gradient for each Weight

[9]Rodriguez-Barroso et. Al. Federated Learning and Differential Privacy: Software tools analysis, the Sherpa.ai FL framework and methodological guidelines for preserving data privacy

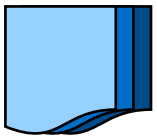
Chapter

Leveraging Uncertainty in Federated Deep Learning

Notation

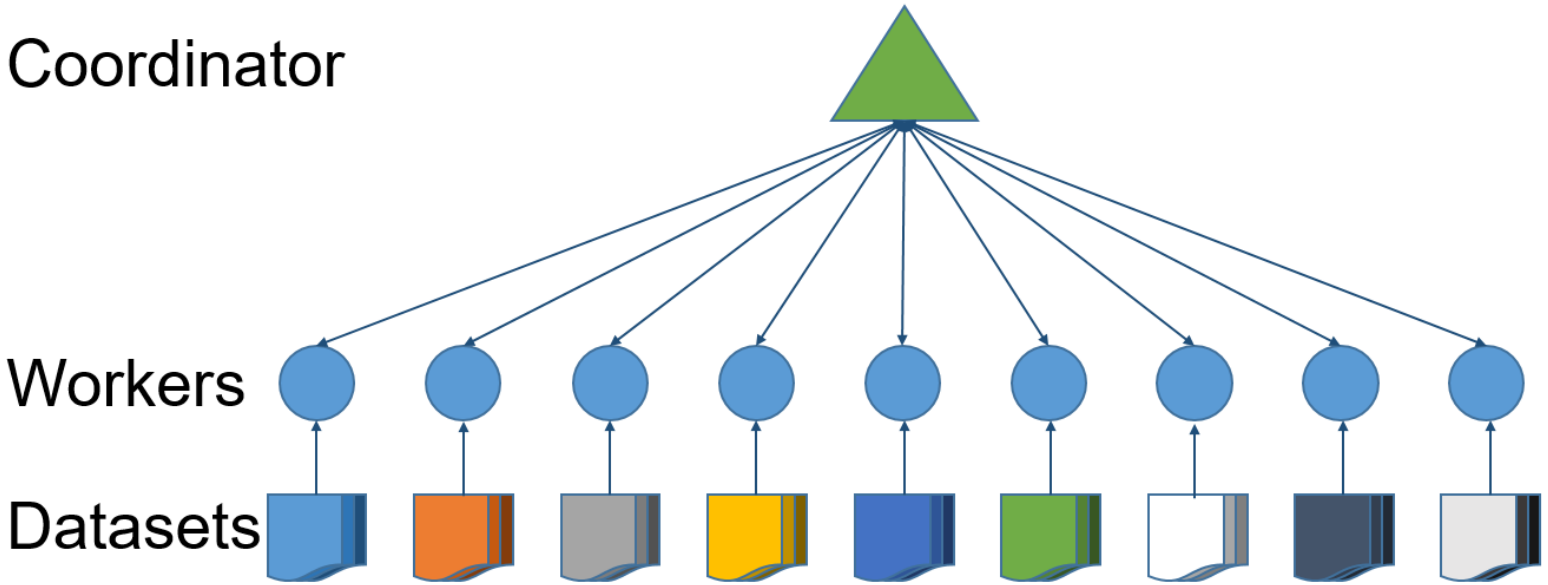
- Communicator: Averages Models

- Worker: Trains Models

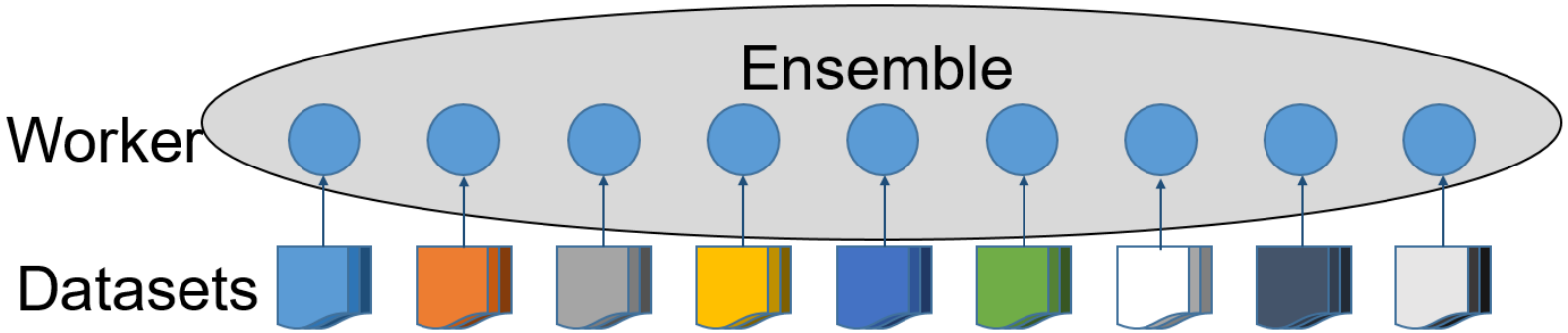


- Dataset: Contains Private Data

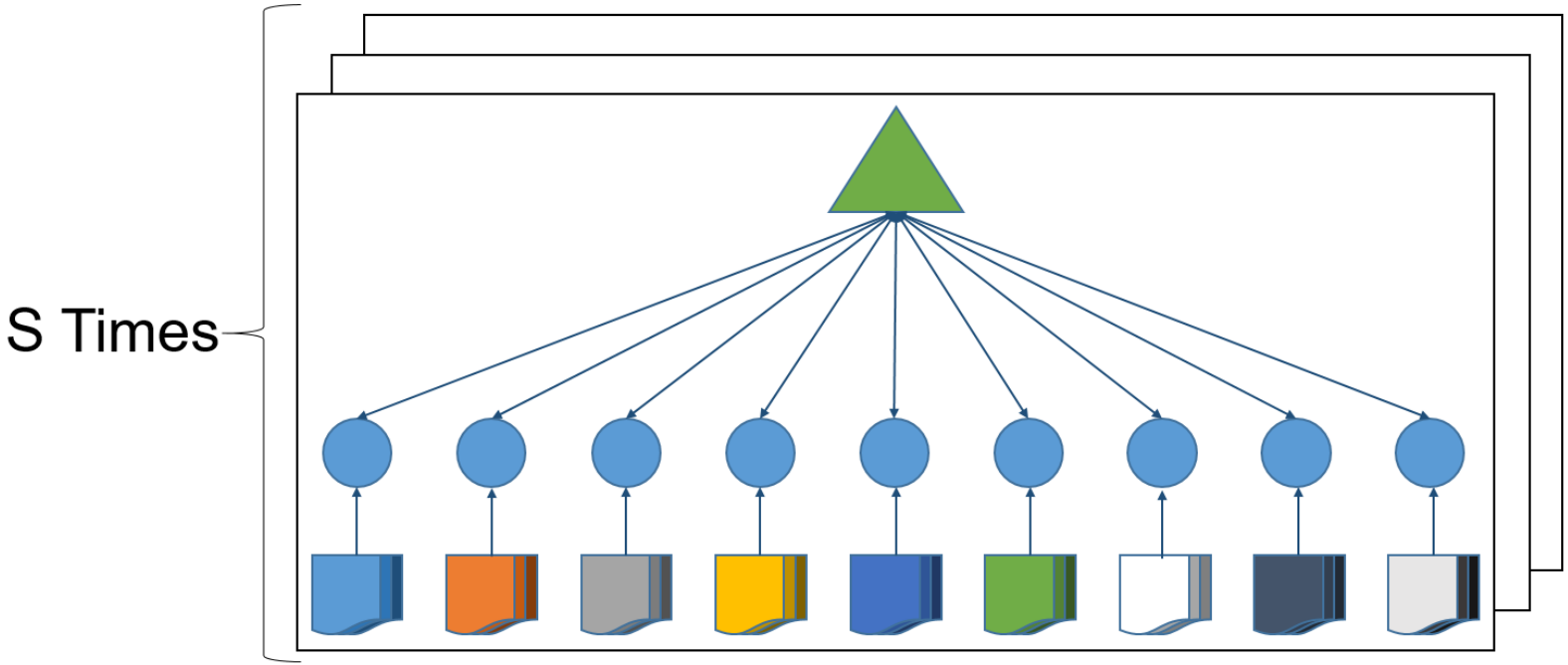
Global Model (Baseline)



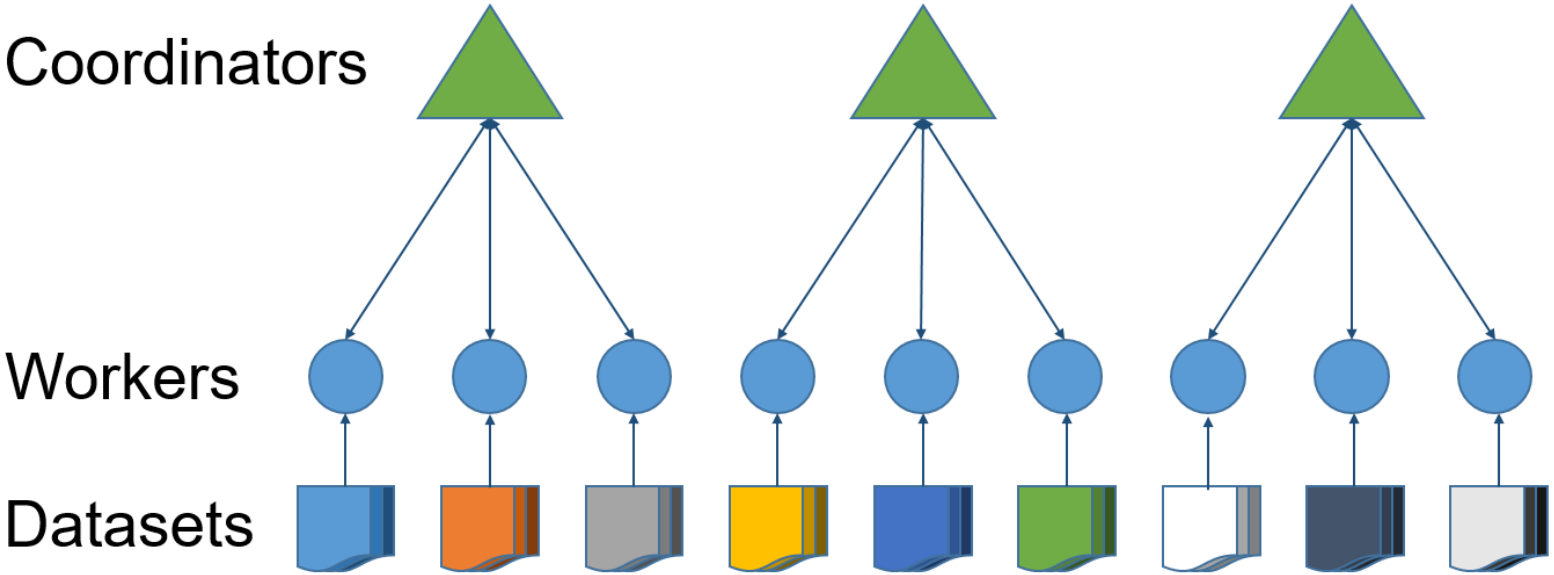
Ensemble of Local Models



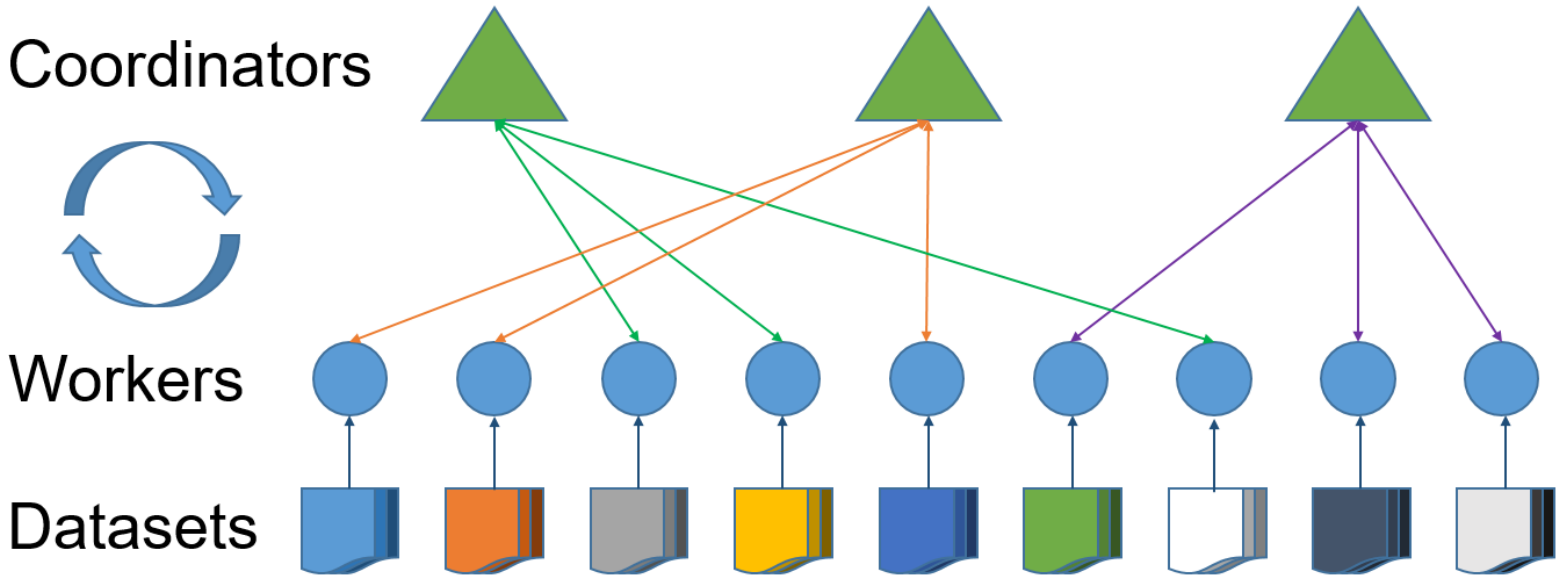
Ensemble of Global Models



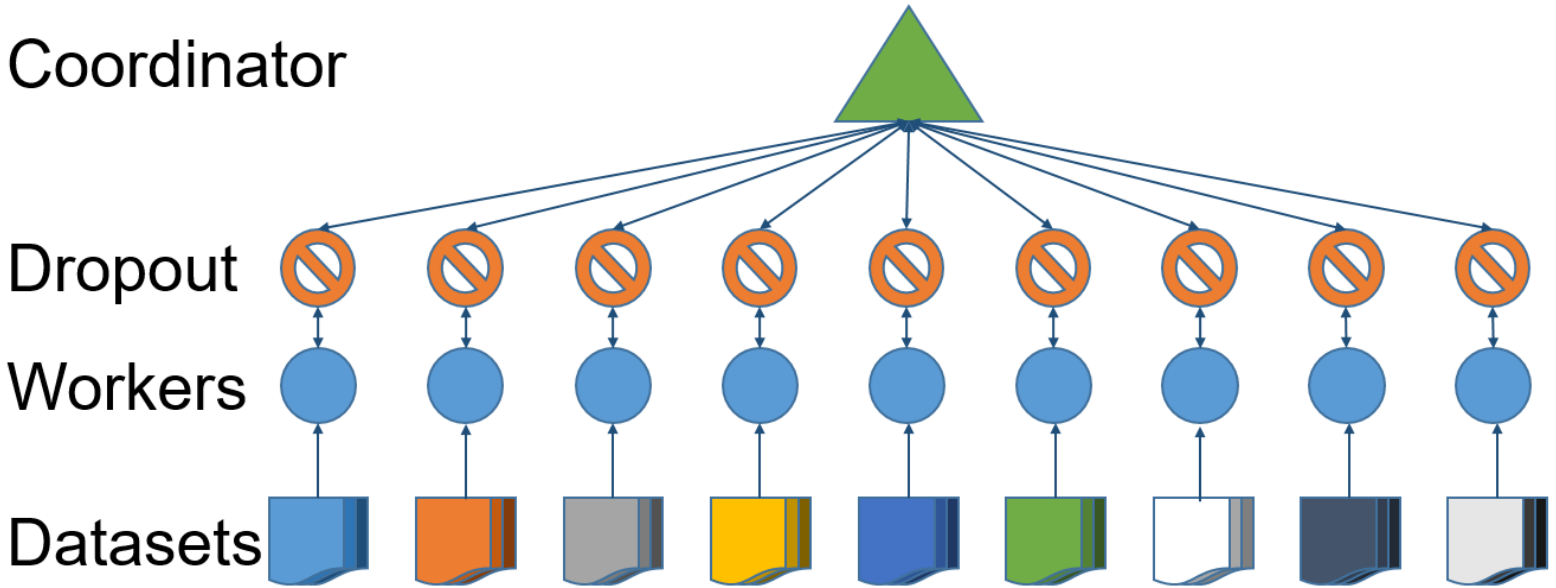
Ensemble w. mult. Coord. Fixed Assignement



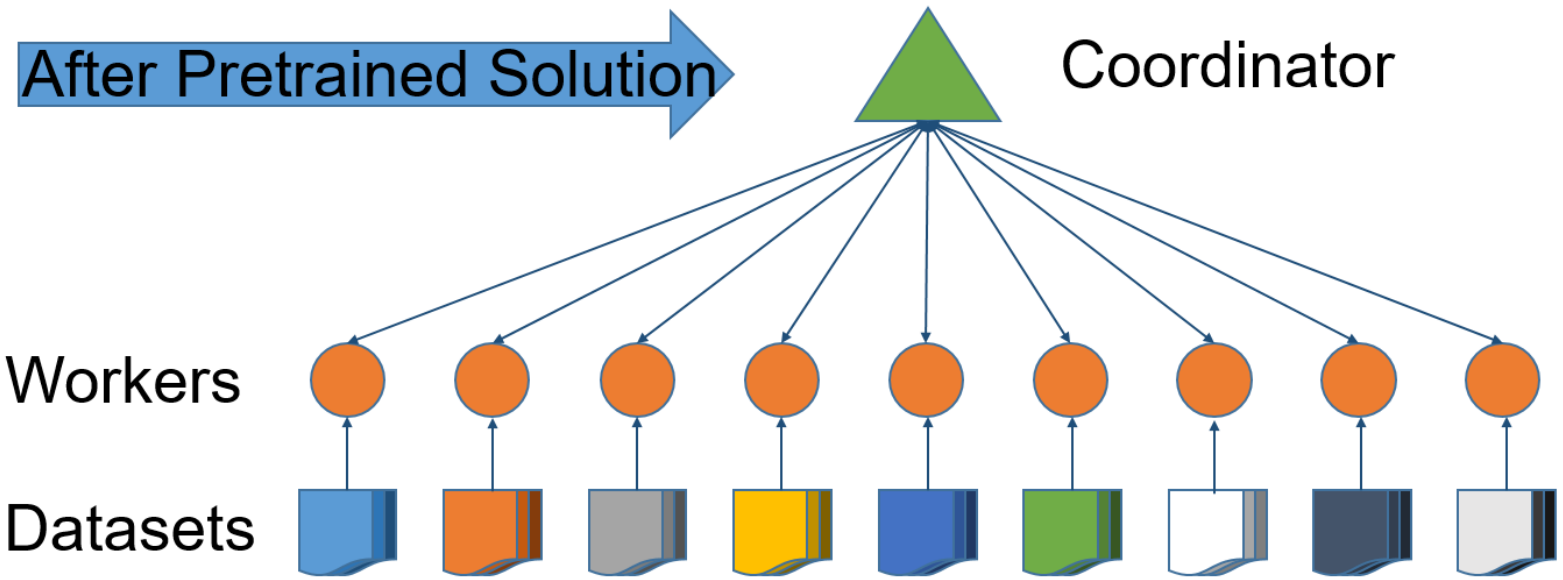
Ensemble w. mult. Coord. Random Assignement



Federated MC-Dropout



Federated SWAG



Chapter
Empirical Evaluation

Empirical Evaluation

Performance of Ensemble Approaches with CIFAR-10

		Detection of OOD (SVHN)		Detection of wrong classified Data	
	Accuracy	**Entropy AUROC**	**Variance AUROC**	**Entropy AUROC**	**Variance AUROC**
Global Model	86.58	0.924	-	0.888	-
Ensemble of local models	72.65	0.679	0.689	0.788	0.687
Ensemble of global models	**89.00**	**0.937**	**0.804**	**0.891**	**0.868**
Ensemble w. mult. Coord.-FA	83.43	0.860	0.740	0.858	0.815
Ensemble w. mult. Coord.-RA	86.72	0.920	0.750	0.887	0.861

Empirical Evaluation

Performance of MC-Dropout with CIFAR-10

		Detection of OOD (SVHN)		Detection of wrong classified Data	
	Accuracy	**Entropy AUROC**	**Variance AUROC**	**Entropy AUROC**	**Variance AUROC**
Global Model	77.88	0.742	-	0.838	-
Global Model with dropout	**86.58**	**0.924**	-	**0.888**	-
Federated MC-dropout	86.32	0.913	**0.714**	0.880	**0.849**

Empirical Evaluation

Performance of SWAG with CIFAR-10

		Detection of OOD (SVHN)		Detection of wrong classified Data	
	Accuracy	**Entropy AUROC**	**Variance AUROC**	**Entropy AUROC**	**Variance AUROC**
Global Model	86.58	0.924	-	0.888	-
Federated SWAG	**87.14**	**0.924**	**0.807**	**0.892**	**0.856**

Chapter
Conclusion

Conclusion

- Best Results in terms of accuracy and AUROC scores
  - Ensemble of Local Models
    - Best Accuracy
    - High Computational Overhead
  - Federated SWAG
    - Easy Integration in Running System
    - Doubled Communication during SWAG Training

Conclusion

- Multiple Coordinators
  - No Performance Increase
  - Might be suitable for moving devices like autonomous driving
- Dropout
  - Improves Performance when used as regularization
  - MC-Dropout does not improve performance

Chapter
Discussion

florian.linsner@rub.de