Probabilistic Deep Learning on Spheres for Weather/Climate Applications

Yann Yasser Haddad
Wentao Feng

Supervisors : **Michaël Defferrard** and **Gionata Ghiggi**

Outline

1. Why go probabilistic?
2. Methods
3. Results
4. Conclusion and future work

Why go probabilistic?

Why go probabilistic?

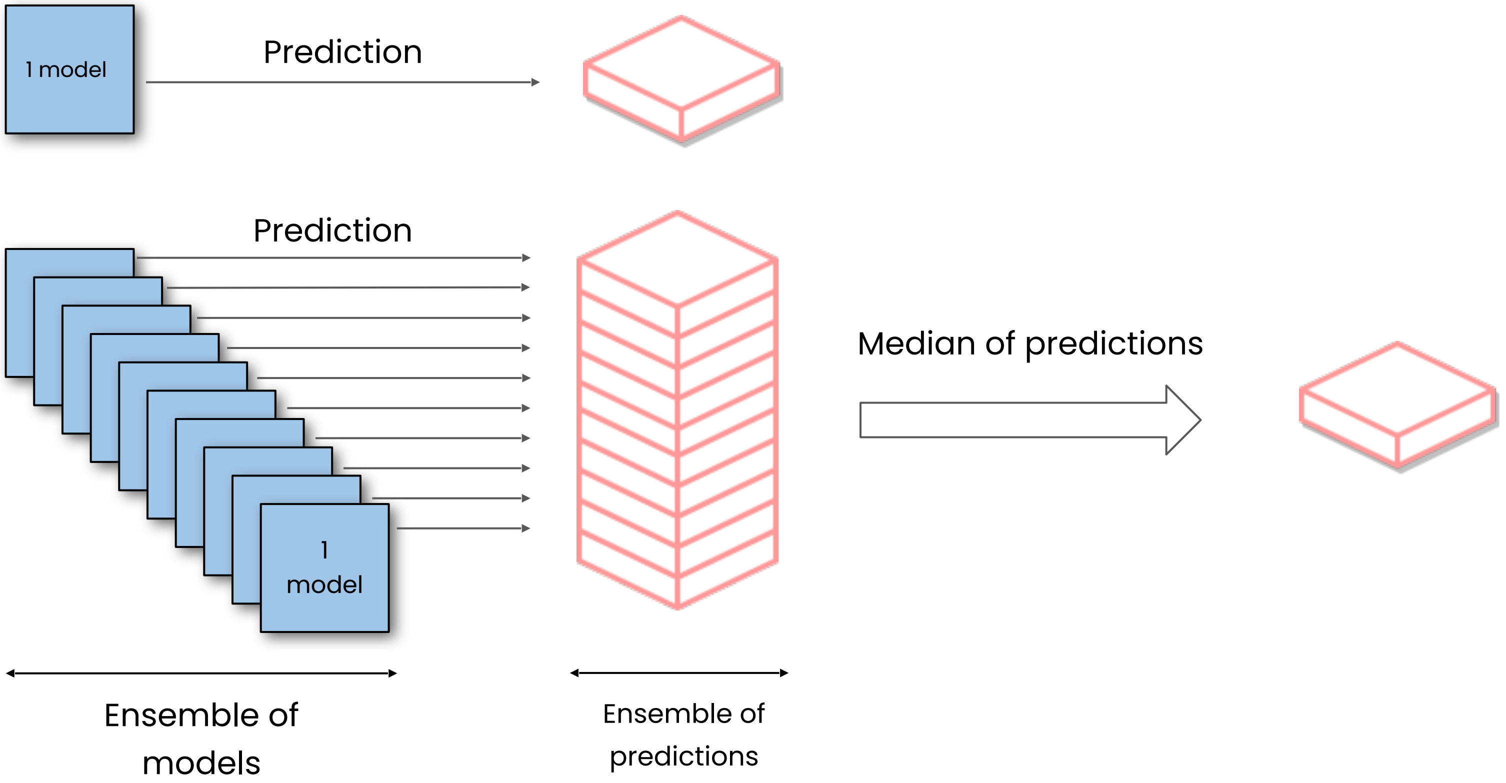
- Address uncertainties in data and model
- Improve deterministic results
- Explore probabilistic metrics

Uncertainties

- Data uncertainty
  - Observations given as input not accurate, contain error
  - Data representativity : we don’t have all the variables
- Model uncertainty
  - Random weight initialization
  - Stochasticity of the network (data and weights)
  - Model architecture (capacity/flexibility)

Models

Deep Ensemble



Stochastic Weight Averaging (SWA)

- Addresses weights uncertainty in a model by recording the weights during training and then taking their average.
  - Leads to better generalization

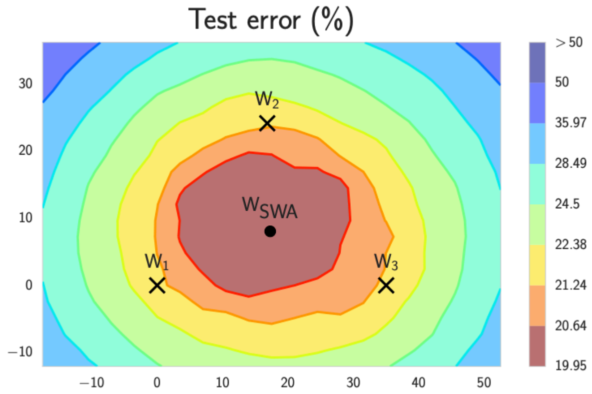


Figure taken from ***Averaging Weights Leads to Wider Optima and Better Generalization, Izmailov et al., 2018***

Stochastic Weight Averaging (SWA)

Model Training

Model after epoch 1 ← Weights : θ_1

Model after epoch 2 ← Weights : θ_2

....

Model after epoch i, SWA start ← Weights : θ_i

....

Model at end of training ← Weights : θ_{end}

Constant learning rate schedule

SWA Training

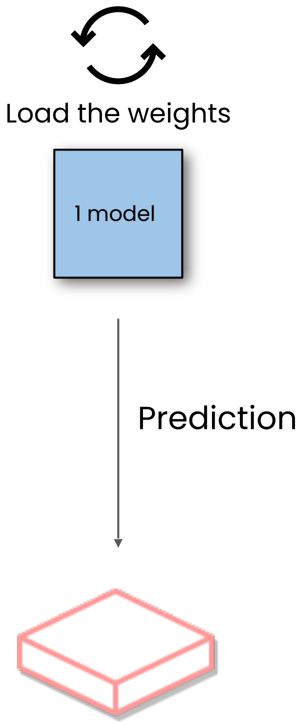
Start collecting weights at every epoch and averaging them at each collection point

$$\bar{\theta} = \frac{n\bar{\theta} + \theta_i}{n+1}$$

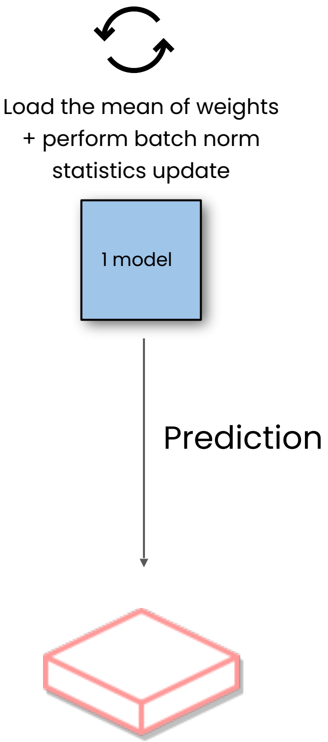
n : number of collections

Stochastic Weight Averaging (SWA)

Normal Testing



SWA Testing



Stochastic Weight Averaging Gaussian (SWAG)

- Similar to SWA, but aims to fit a Gaussian distribution over the weights :
    - using the SWA solution as mean
    - and a low rank + diagonal covariance derived from the weights
    - Sample weights from distribution to create a new model

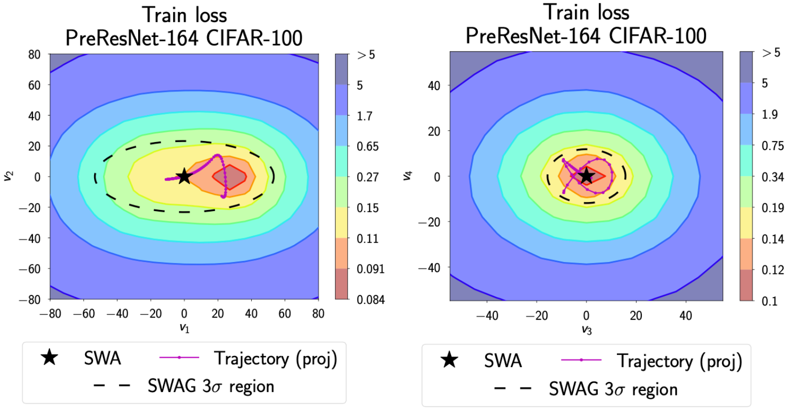


Figure taken from ***A Simple Baseline for Bayesian Uncertainty in Deep Learning, Maddox et al., 2019***

Stochastic Weight Averaging Gaussian (SWAG)

Model Training

Model after epoch 1 ← Weights : θ_1

Model after epoch 2 ← Weights : θ_2

....

Model after epoch i, SWA start ← Weights : θ_i

....

Model at end of training ← Weights : θ_{end}

Constant learning rate schedule

SWAG

Start collecting weights at every epoch

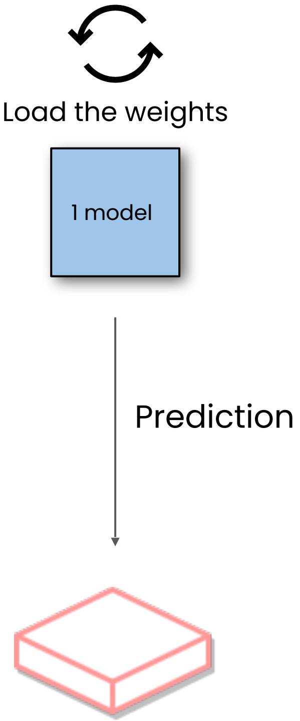
$$\overline{\theta} = \frac{n\overline{\theta} + \theta_i}{n+1} \qquad \overline{\theta^2} = \frac{n\overline{\theta^2} + \theta_i^2}{n+1}$$

Compute a K-rank deviation matrix $\hat{D} = \theta - \overline{\theta}$ by appending at each collection point j $\theta_j - \overline{\theta}$ and removing columns when rank is attained

Compute at the end of training $\Sigma_{diag} = \overline{\theta^2} - \overline{\theta}^2$

Stochastic Weight Averaging Gaussian (SWAG)

Normal Testing

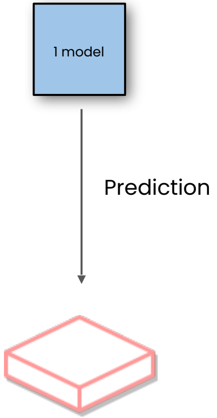


SWAG Testing

Sample weights with scale s from

$$\tilde{\theta}_i \sim \mathcal{N}\left(\theta_{\text{SWA}}, s \times \left(\frac{1}{2}\Sigma_{\text{diag}} + \frac{\widehat{D}\widehat{D}^\top}{2(K-1)}\right)\right)$$

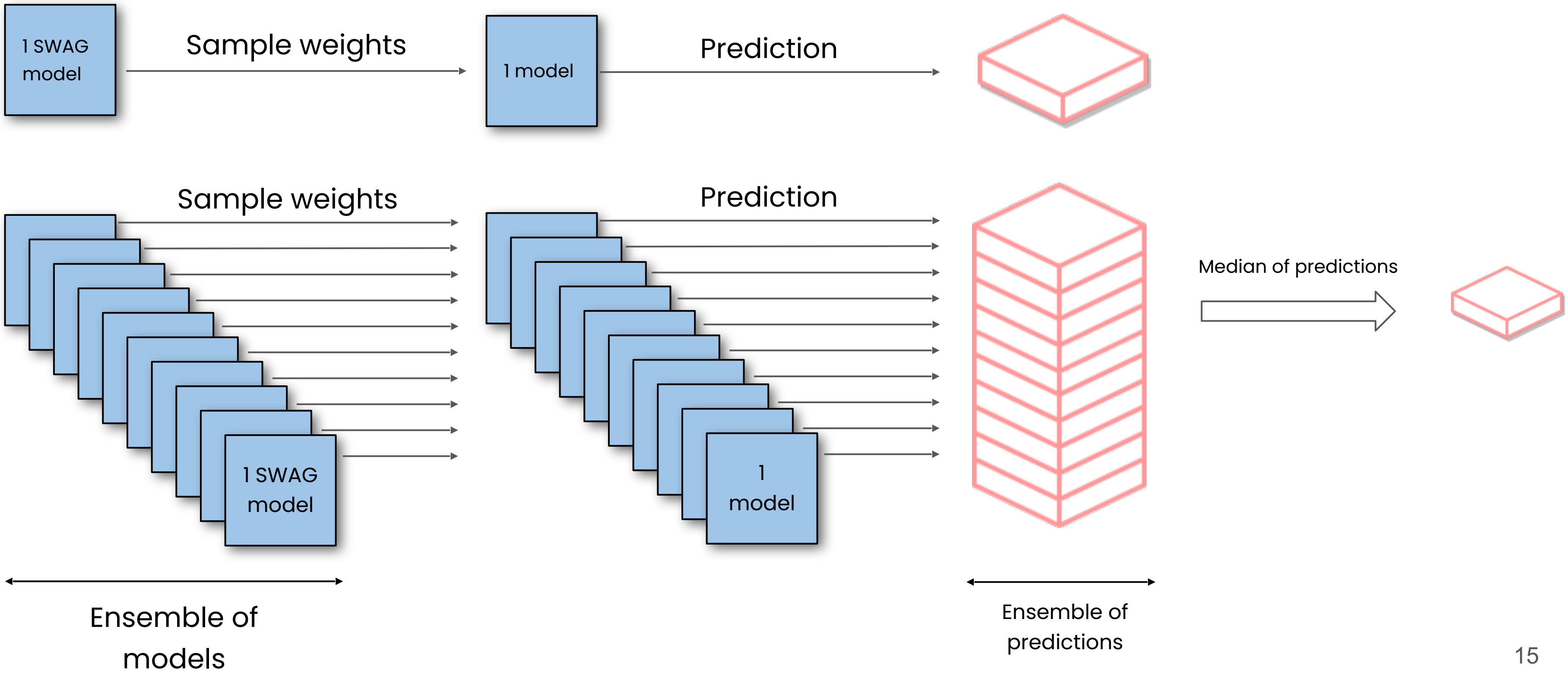
+ perform batch norm statistics update



MultiSWAG

Deep Ensemble + SWAG

MultiSWAG


1 SWAG model
Sample weights
1 model
Prediction
Sample weights
Prediction
1 SWAG model
1 model
Median of predictions
Ensemble of models
Ensemble of predictions

Experiments

General Training Configuration

- Train years : 2010-2015
- Validation year : 2016
- Test years : 2017-2018
- Epochs : 12
- Number of steps ahead : 2 (instead of 8)

SWA/SWAG

- **Training**

Hyperparameter	Value
SWA/SWAG start epoch	9
Rank **K** of deviation matrix	20
Weight Collections	40 (10/epoch)

- **Testing**

Model	Scale	Number of realizations
SWA	0.0	1
SWAG	0.01	10
SWAG	0.1	10
SWAG	0.3	10

Deep Ensemble

- **Training:**

Models	Number of models	Random train/val split	Number of train/val years
Deep Ensemble	10	Yes	6/1
Deep Ensemble with fixed input	10	No	6/1

MultiSWA/SWAG

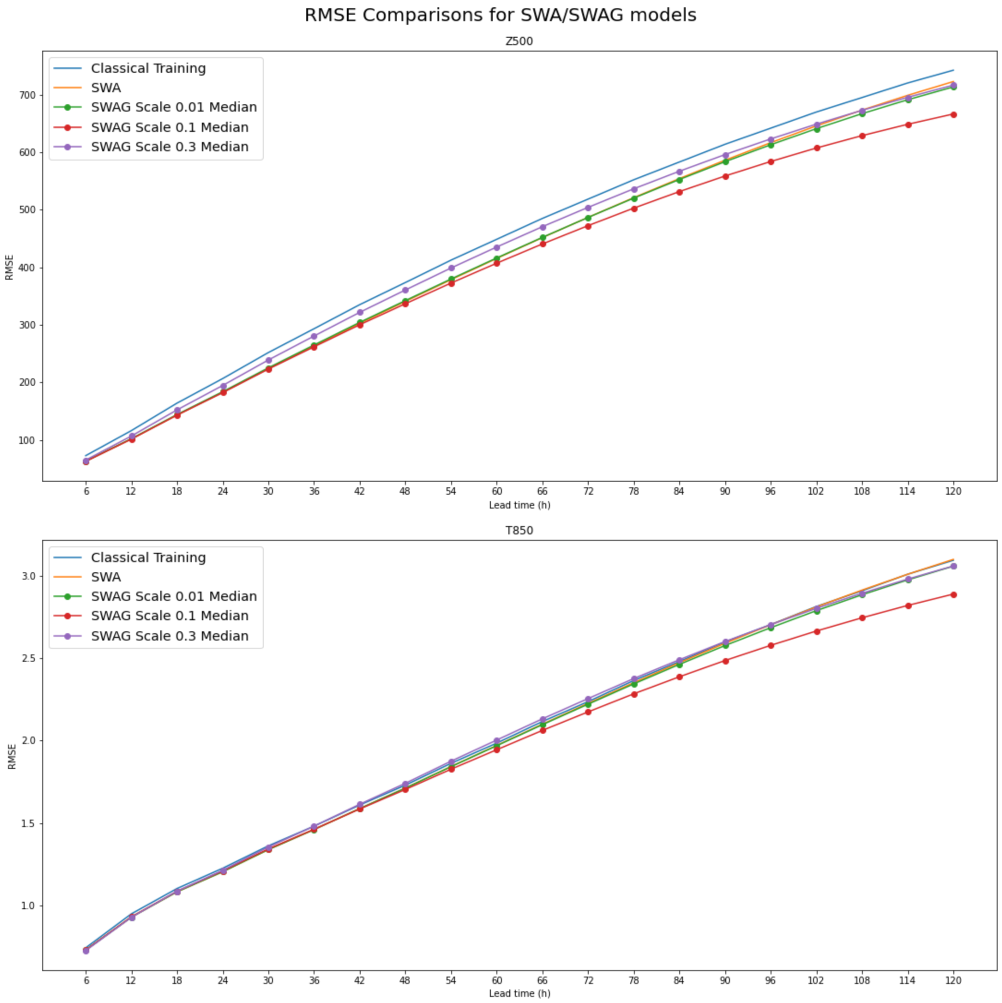
- **Training:**

Hyperparameter	Value
Number of models	10
SWA/SWAG start epoch	9
Rank **K** of deviation matrix	20
Weights Collection	40 (10/epoch)

- **Testing**

Model	Scale	Number of realizations	Take median of realizations/ model
MultiSWA	0.0	1 per model	No
MultiSWAG	0.1	5 per model	No
MultiSWAG	0.1	5 per model	Yes

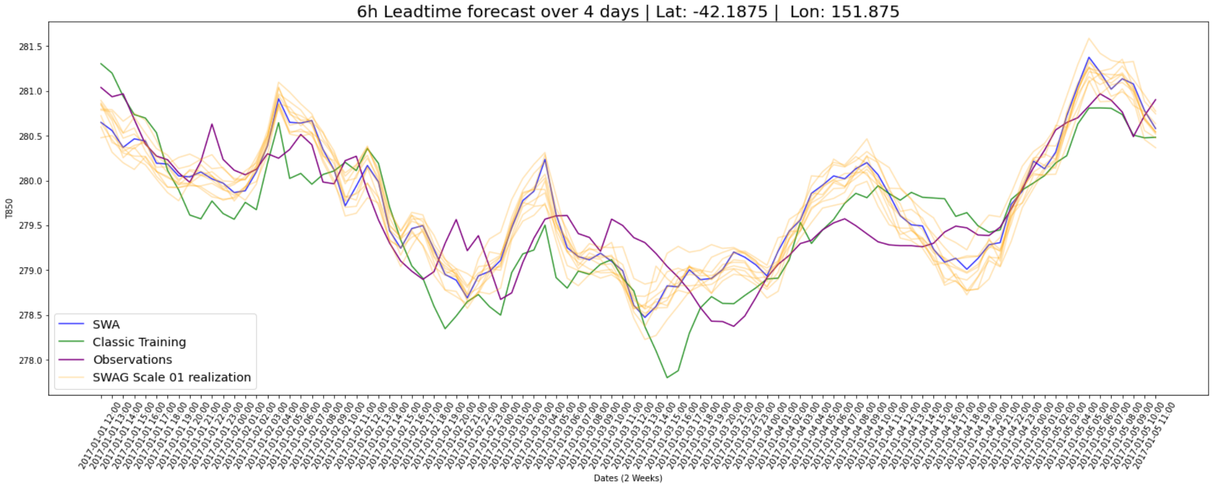
Results

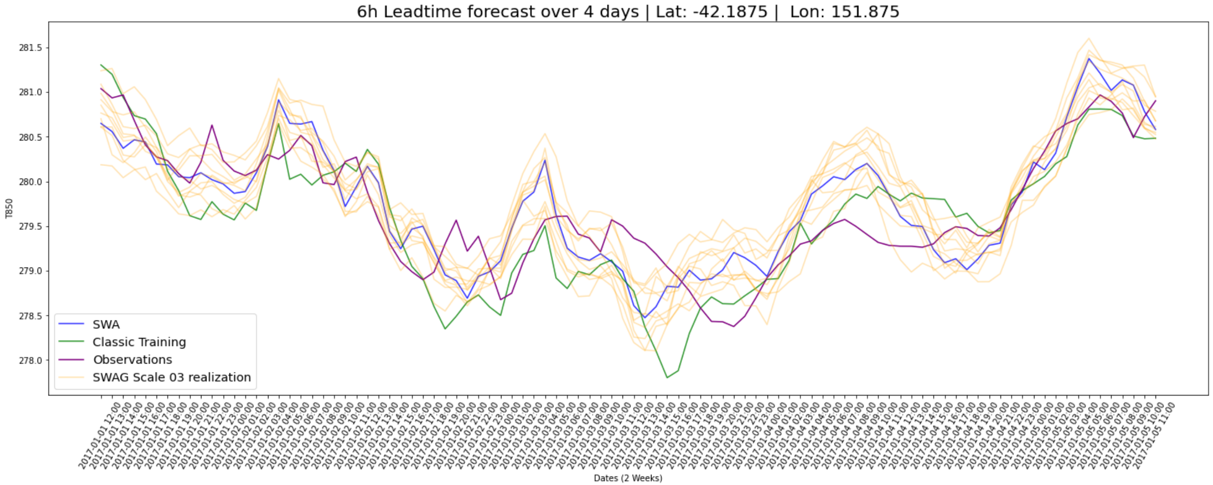


EPFL

Root Mean Squared Error

- **SWA** is already better than Classical Training for Z500
- **The median of SWAG realizations with Scale 0.1** is better than classical training and all other experiments on SWA/SWAG
- Scale of 0.1 seems to be a sweet spot for this model
- Other scales converge to SWA

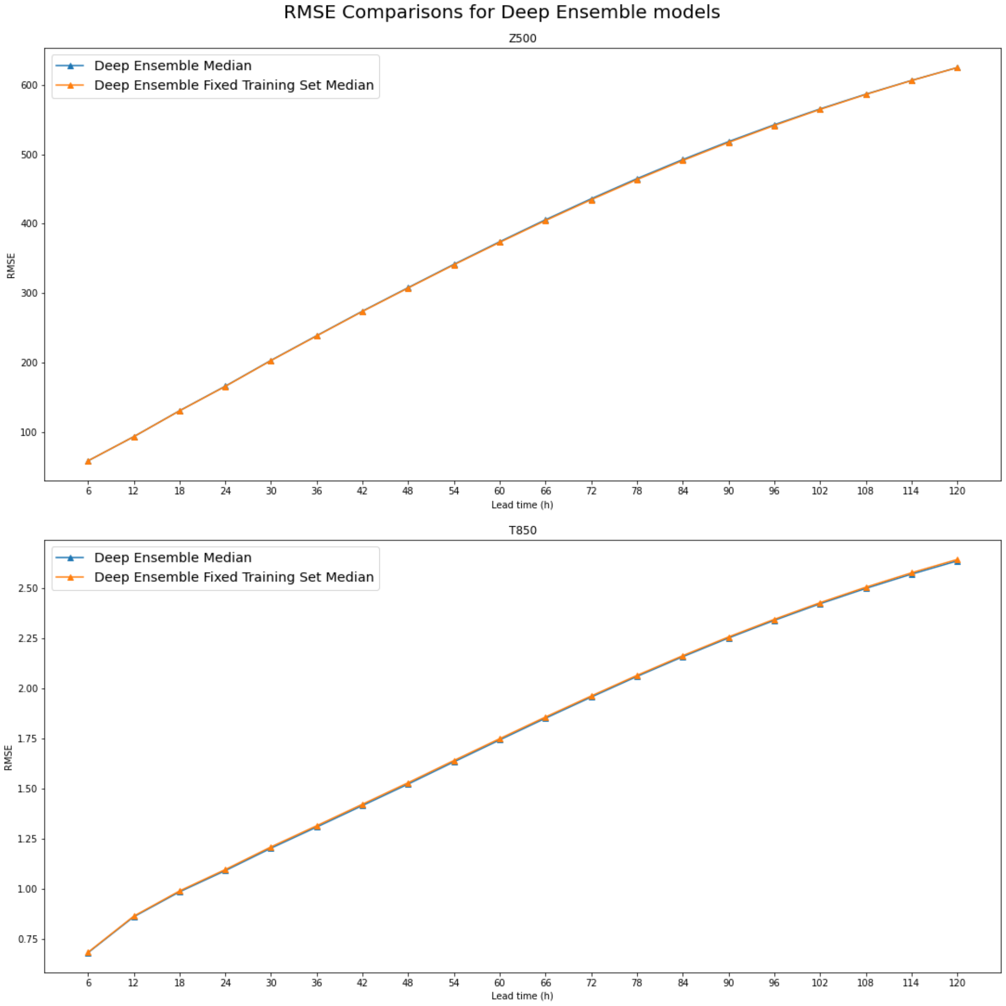

6h Leadtime forecast over 4 days | Lat: -42.1875 | Lon: 151.875
T850
281.5
281.0
280.5
280.0
279.5
279.0
278.5
278.0
SWA
Classic Training
Observations
SWAG Scale 01 realization
Dates (2 Weeks)


6h Leadtime forecast over 4 days | Lat: -42.1875 | Lon: 151.875
T850
281.5
281.0
280.5
280.0
279.5
279.0
278.5
278.0
SWA
Classic Training
Observations
SWAG Scale 03 realization
Dates (2 Weeks)

Root Mean Squared Error

Model	Z500 6H	Z500 120H	T850 6H	T850 120H
Classical Training	72.780	742.754	0.743	3.093
SWA	63.004	723.077	0.730	3.099
SWAG Scale 0.01 Median	63.246	713.748	0.729	3.058
SWAG Scale 0.1 Median	**62.845**	**666.662**	0.729	**2.888**
SWAG Scale 0.3 Median	65.080	716.906	**0.727**	3.059

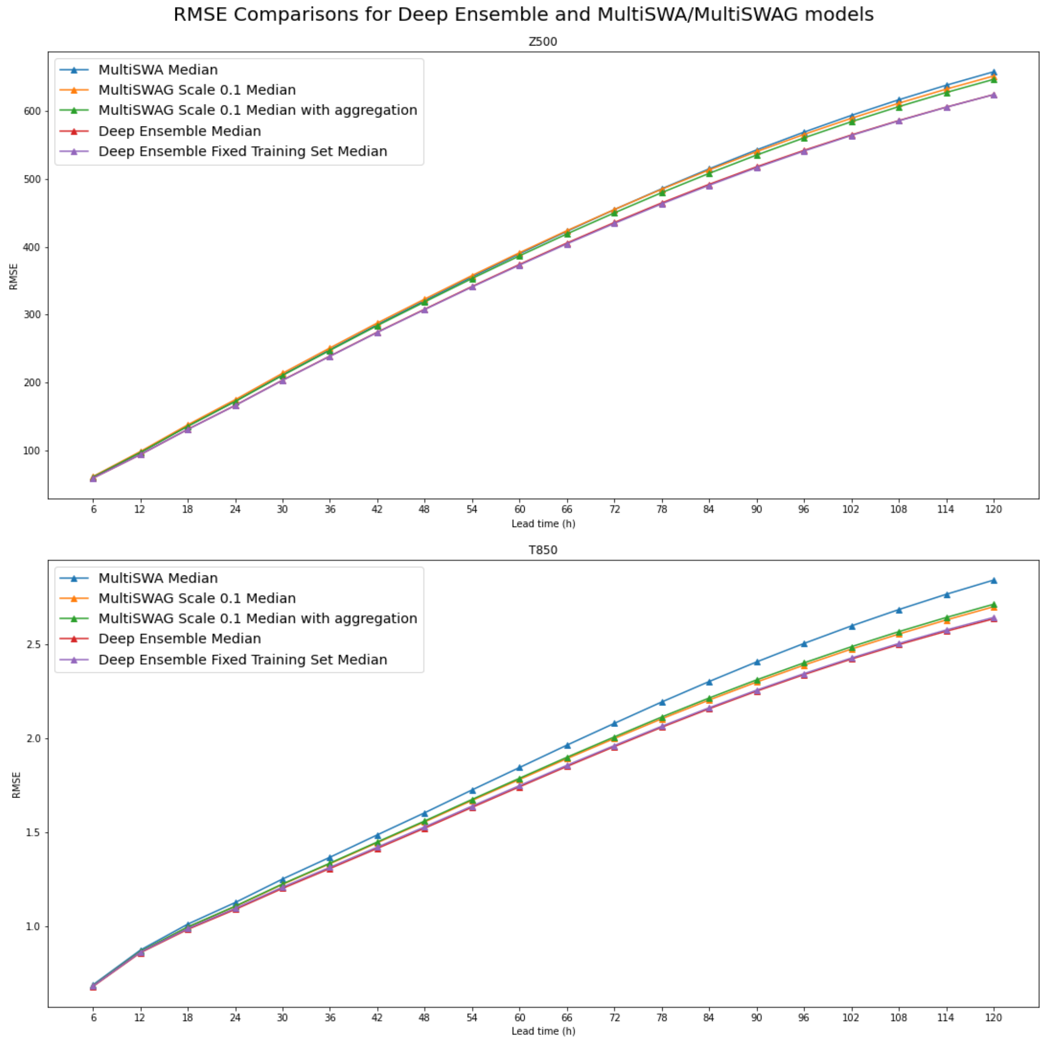
- **SWA** is already better than Classical Training for Z500
- **The median of SWAG realizations with Scale 0.1** is better than classical training and all other experiments on SWA/SWAG
- Scale of 0.1 seems to be a sweet spot for this model
- Other scales converge to SWA



Root Mean Squared Error

- Fixing the training set for Deep Ensemble does not have an impact on deterministic metrics

Model	Z500 6H	Z500 120H	T850 6H	T850 120H
Deep Ensemble Median	**58.567**	624.798	**0.682**	**2.634**
Deep Ensemble Fixed Training Set Median	58.613	624.734	0.684	2.642



EPFL

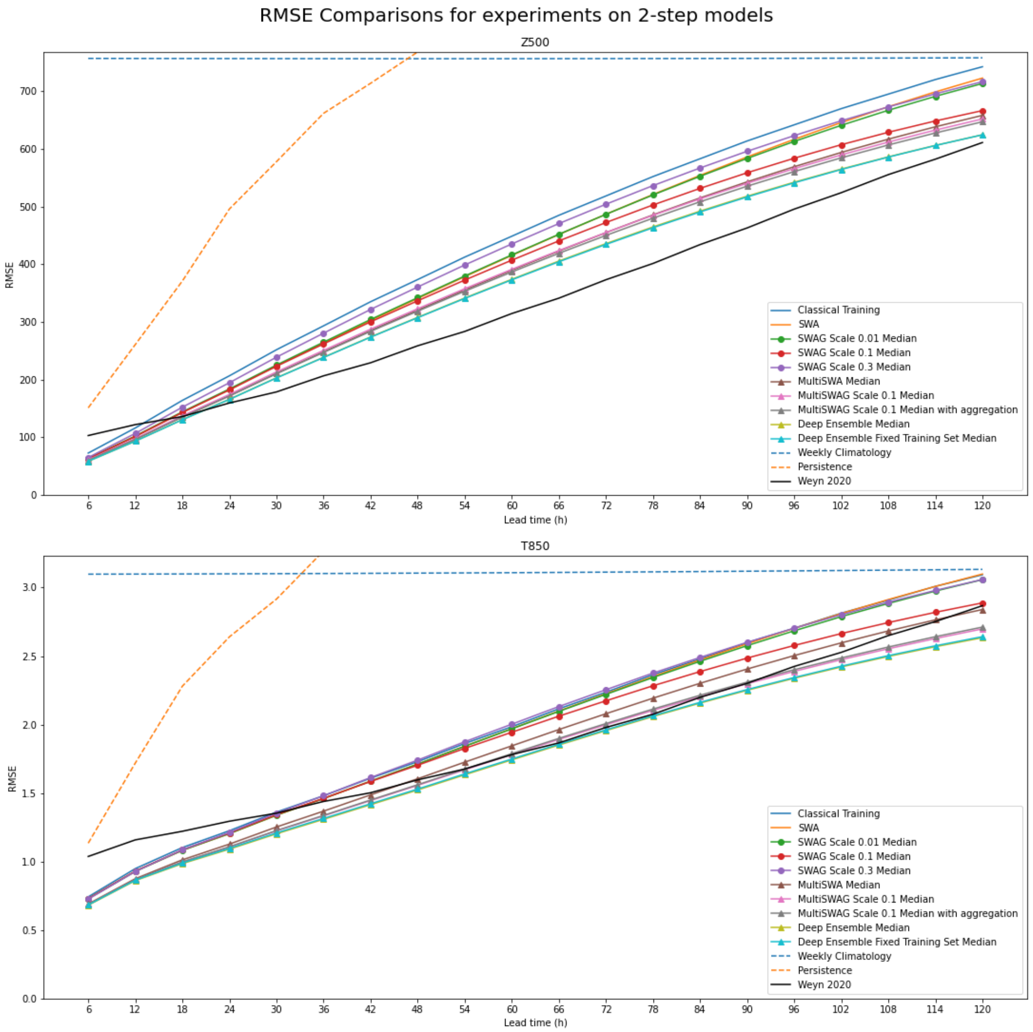
Root Mean Squared Error

- MultiSWAG gives a better estimate than MultiSWA
- **MultiSWAG :** Taking the median of the realizations per model has very little impact on the deterministic performances
- Surprisingly, Deep Ensembling performs better than MultiSWA and MultiSWAG

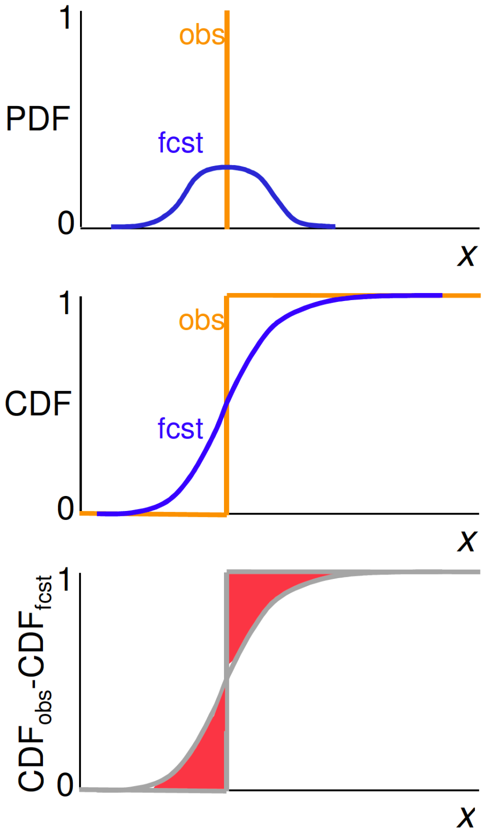
Root Mean Squared Error

Model	Z500 6H	Z500 120H	T850 6H	T850 120H
Deep Ensemble Median	**58.567**	624.798	**0.682**	**2.634**
Deep Ensemble Fixed Training Set Median	58.613	624.734	0.684	2.642
MultiSWA Median	60.102	658.468	0.691	2.84
MultiSWAG Scale 0.1 Median	60.984	652.228	0.685	2.698
MultiSWAG Scale 0.1 Median with aggregation	60.112	647.285	0.686	2.711

- MultiSWAG gives a better estimate than MultiSWA
- **MultiSWAG :** Taking the median of the realizations per model has very little impact on the deterministic performances
- Surprisingly, Deep Ensembling performs better than MultiSWA and MultiSWAG



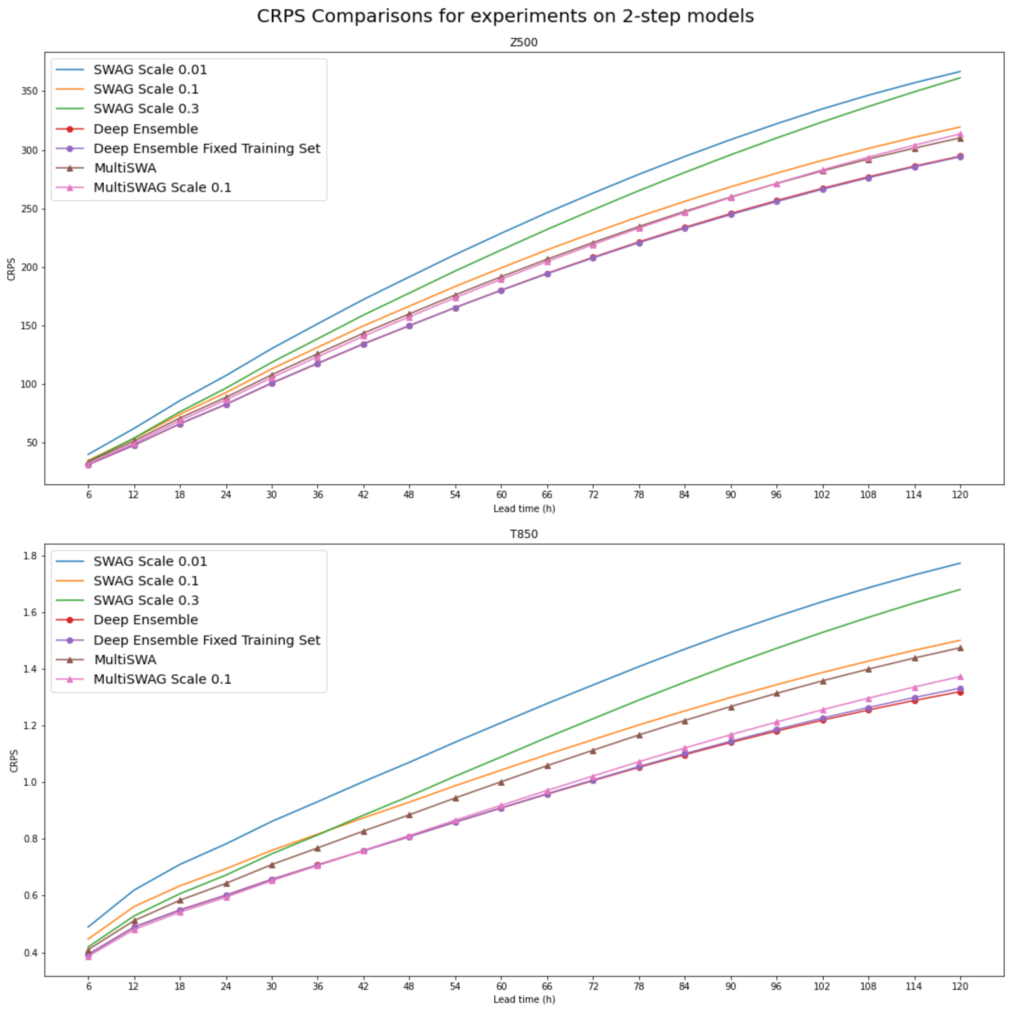
Model	Z500 6H	Z500 120H	T850 6H	T850 120H
Classical Training	72.780	742.754	0.743	3.093
SWAG Scale 0.1 Median	62.845	666.662	0.729	2.888
Deep Ensemble Median	**58.567**	624.798	**0.682**	**2.634**
MultiSWAG Scale 0.1 Median	60.984	652.228	0.685	2.698
Weekly Climatology	757.200	758.276	3.098	3.133
Persistence	151.205	992.632	1.135	4.311



Ensemble Continuous Ranked Probability Score (CRPS)

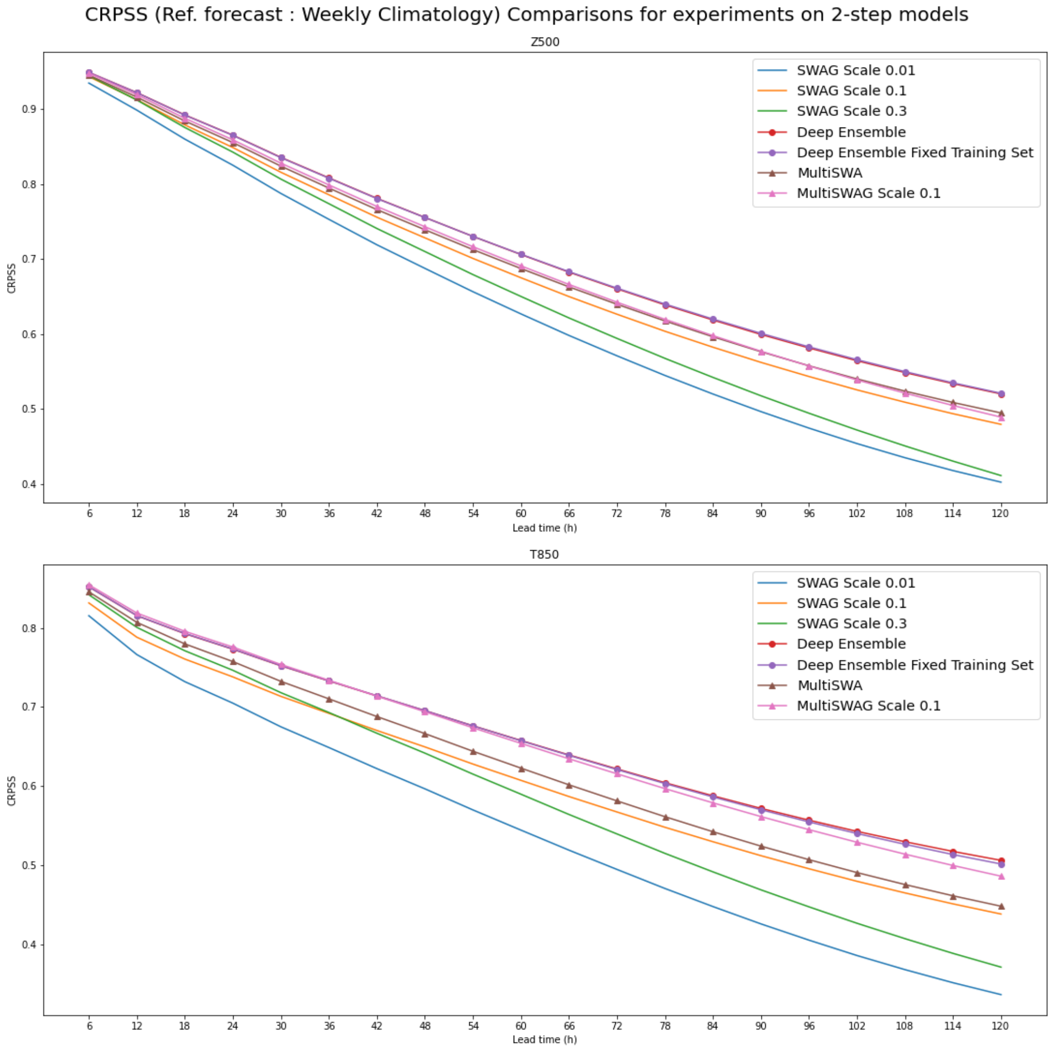
- Evaluates the integrated error between the forecast cumulative distribution function and the observation
- Same as Mean Absolute Error (MAE) for deterministic forecasts
- Best score : 0 —> lower is better

Figures taken from ***Ensemble Verification Metrics, Debbie Hudson 2017***



Ensemble CRPS

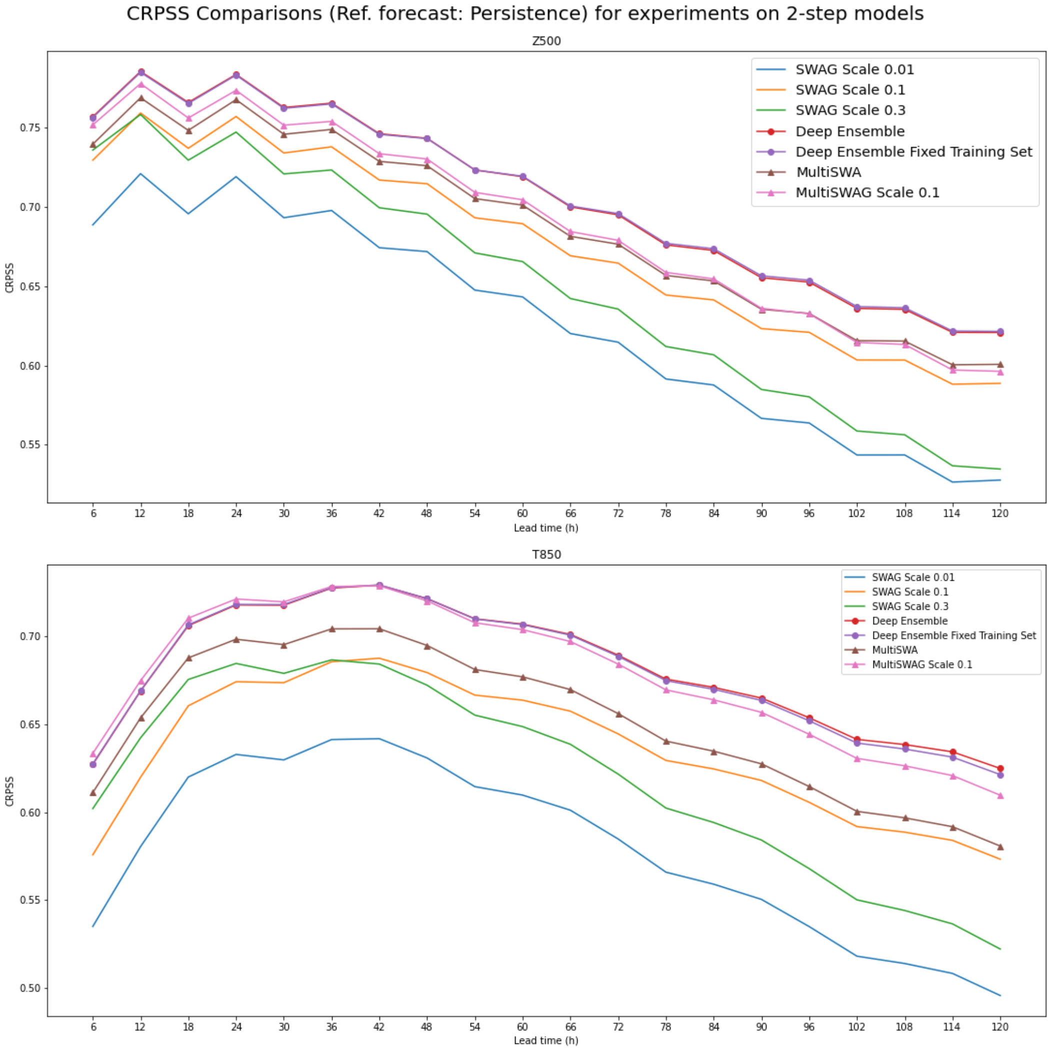
- Evaluates the integrated error between the forecast cumulative distribution function and the observation
- Same as Mean Absolute Error (MAE) for deterministic forecasts
- Best score : 0 —> lower is better



CRPSS wrt Weekly Climatology

- $$CRPSS = 1 - \frac{CRPS_{forecast}}{CRPS_{ref}}$$

  where ***ref*** is a reference forecast
- 2 reference forecasts:
  - Weekly Climatology
  - Persistence



CRPSS wrt Persistence

- $$CRPSS = 1 - \frac{CRPS_{forecast}}{CRPS_{ref}}$$
  where ***ref*** is a reference forecast
- 2 reference forecasts:
  - Weekly Climatology
  - Persistence

Conclusion and future work

Conclusion

- The methods explored during this project all improve deterministic metrics compared to regular training.
- The same conclusion apply to probabilistic metrics.

Conclusion

- We observe some key differences in the methods :
  - **SWA/SWAG :**
    - Little additional training time compared to classic training
    - Already better performances than classic Training
  - **SWAG :**
    - Diversity for free : create many realizations from a single model training
  - **Deep Ensemble :**
    - More models to train -> more time spent on training
    - Captures well the uncertainty and the median of the ensemble gives us the best results
  - **MultiSWA/SWAG :**
    - Same training time as Deep Ensemble
    - Offers flexibility for the different members of the ensemble

Future Work

- Deep Ensemble with less data (data sampling) and perturbed initial conditions
  - Faster computation and hopefully better spread
- Look into the influence of the rank and the number of collections on the performances of the SWAG/MultiSWAG models
- Look into the selection of the optimal scale, or scale range for SWAG and MultiSWAG
- Combine the different models in an ensemble
- Combine different scales in an ensemble of SWAG/MultiSWAG realizations

Thank you for listening!