Aalto University
School of Science
Master's Programme in Life Science Technologies

Bent Ivan Oliver Harnist

Probabilistic Precipitation Nowcasting using Bayesian Convolutional Neural Networks

Master's Thesis
Espoo, July 20th, 2022

DRAFT! — June 19, 2022 — DRAFT!

Supervisor: Professor Arno Solin
Advisor: Terhi Mäkinen D.Sc.
Seppo Pulkkinen D.Sc.

Aalto University
School of Science
Master's Programme in Life Science Technologies

ABSTRACT OF
MASTER'S THESIS

Author:	Bent Ivan Oliver Harnist		
Title: Probabilistic Precipitation Nowcasting using Bayesian Convolutional Neural Networks			
Date:	July 20th, 2022	**Pages:**	27
Major:	Complex Systems	**Code:**	SCI3060
Supervisor:	Professor Arno Solin		
Advisor:	Terhi Mäkinen D.Sc. Seppo Pulkkinen D.Sc.		
!FIXME **This is an example how to use fixme: add your abstract here.** FIXME!			
Keywords:	Precipitation nowcasting,		
Language:	English		

Aalto-yliopisto
Perustieteiden korkeakoulu
Tieto-, tietoliikenne- ja informaatiotekniikan maisteriohjelma

DIPLOMITYÖN TIIVISTELMÄ

Tekijä:	Bent Ivan Oliver Harnist		
Työn nimi: Probabilistinen Sateen Nowcasting käyttäen Bayesilaisia Konvolutiivisia Neuroverkkoja			
Päiväys:	20. heinäkuuta 2022	**Sivumäärä:**	27
Pääaine:	Complex Systems	**Koodi:**	SCI3060
Valvoja:	Professori Arno Solin		
Ohjaaja:	Terhi Mäkinen D.Sc. Seppo Pulkkinen D.Sc.		
joku			
Asiasanat:	Sateen ennustaminen,		
Kieli:	Englanti		

Acknowledgements

!Fixme **My acknowledgements** Fixme!

Espoo, July 20th, 2022

Bent Ivan Oliver Harnist

Abbreviations and Acronyms

DL	Deep learning
NWP	Numerical Weather Prediction
BNN	Bayesian Neural Network
STEPS	Short-Term Ensemble Prediction System
LINDA	Lagrangian Integro-Difference equation model with Autoregression
Z	Reflectivity factor
dBZ	Decibel relative to Z

Contents

Chapter 1

Introduction

!FIXME **Add rest of references, by 22.6, 45min** FIXME!

!FIXME **remove "we" passive** FIXME!

Nowcasting is defined as weather forecasting at very short timescales, from minutes ahead up to several hours. The ability to provide an accurate precipitation nowcast has grown during this century into a meteorologically and societally significant challenge. This significance emanates from the fact that early warnings of severe precipitation enable authorities and other actors to make early decisions enabling for example disaster damage control, traffic safety, flash flood prevention, and mitigation of economic loss incurred by sudden and heavy precipitation. High-density urbanization has exacerbated these issues, making it evermore vital to discover reliable and skillful methods for precipitation nowcasting.

Numerical weather prediction (NWP) solves the partial differential equations governing the physical processes of weather and climate to produce forecasts. However, such method is not applicable to predicting at timescales as short as 0 to 6 hours. This is mainly due to imperfect initialization making simulations unable to reach numerical stability in such short time timescales. Other problems with NWP include their computational expensiveness and the low spatial resolution of large-scale NWP, making those systems unfit to predict for example heavy localized rainfall.

!FIXME **add a tiny little bit about det classical nowcast methods as tongue teaser, by 22.6, 45min** FIXME!

To compensate for the shortcomings of NWP in the realm of precipitation nowcasting, many different dedicated models and algorithms have been developed. Many of those systems are based on the estimation of the precipitation advection field from radar echo image sequences and the further extrapolation of these sequences along the advection field. Other methods have been developed that track and try to nowcast the evolution of individual rain cells, instead of the whole field. Dedicated nowcasting methods have had great success in forecasting the immediate future, but their performance typically degrades quickly, becoming unreliable often in the range of an hour. This is related to the fact that basic extrapolation-based methods have limitations such as failing to capture nonlinear patterns like convective initiation and life-cycles. A lot of work has been done in nowcasting in trying to incorporate modeling of higher-order and multiscale phenomena into advection-extrapolation based models. Recently, Deep-learning (DL) based precipitation nowcasting approaches have started showing promising results delving into the issues of traditional methods thanks to the high volume of training data available, their expressive power and the relative cheapness of inference.

1.1 Problem statement

While the above introduction has cast the problem of nowcasting as finding a single optimal point estimate for future precipitation, it is useful to think about it in a probabilistic way. Because in nowcasting the phenomena we are most interested in are extreme events, and these are the events that we want to prepare for, it makes intuitive sense that accurate and reliable probabilistic nowcasts are primordial for operational decision-making in meteorological crises.

Indeed, the interest in probabilistic nowcasts has grown lately and many new probabilistic models have been introduced in the last decades, notably the Short-Term Ensemble Prediction System (STEPS) [4] and more recently the Lagrangian Integro-Difference equation model with Autoregression (LINDA) [8]. These methods predict ensembles, that are used to estimate underlying distributions of data and precipitation probabilities. They perform reasonably well, but have computationally expensive inference and suffer from the same limitations other advection-extrapolation based methods.

So far, only little work has been done on using DL to produce probabilistic precipitation nowcasts, with the biggest breaktrough perhaps being Ravuri et al. [10] using adversarially trained deep generative models to produce ensemble nowcasts. There exists many possible ways of making probabilistic nowcasts using deep learning, with most of them focusing on directly modeling probability distributions of data. Another approach, that we will focus on from now on is instead to model the uncertainty of model parameters rather than that of the data. This is done by using Bayesian Neural Networks and has an advantage of providing implicit regularization of model parameters thus reducing overfitting, while outputting an ensemble of predictions whose variability in part reflects network parameter uncertainty given the training data. Bayesian neural networks have been proposed for use in other risk-averse application such as biomedical image segmention [6] and autonomous vehicles [7].

In this work, we will approach the problem of making skillful and reliable probabilistic precipitation nowcasts by building an uncertainty-aware Bayesian Neural Networks. A baseline Convolutional Neural Network (CNN) will be turned into a Bayesian Convolutional Network (BCN) with optimization performed using stochastic variational inference (SVI) over model parameters. The model is trained, validated and tested using Finnish Meteorological Institute (FMI) radar reflectivity composites. For verification, both deterministic and probabilistic metrics are calculated in order to assess prediction skill against other probabilistic models and deterministic counterparts.

1.2 Structure of the Thesis

The present work is organized as follows. Chapter 2 contains background and a literature review on precipitation nowcasting and bayesian deep learning, aiming to familiarize the reader with essential concepts regarding the subject. Chapter 3 describes the experimental details of the work performed, including the datasets used for training and verification, the models implemented, as well as verification methods and baselines.

Chapter 4 presents the results of the experiments performed, including nowcasting examples of meteorologically interesting events, prediction skill measured with both deterministic and probabilistic metrics against multiple baseline models, as well as an assessment of predictive uncertainty. Chapter 5 discusses these results, their impact, and validity in details. Finally, Chapter 6 closes the thesis by summarizing the most important findings and takeaways.

Chapter 2

Background

!FIXME **Estimate at start: 12-15 pages** FIXME!

2.1 Precipitation Nowcasting

2.1.1 Weather radars and radar products

!FIXME **7-9.6 Some radar reflectivity image, pic of some radar??, 2h, 1.5-2 pages** FIXME!

- !FIXME **7-9.6, 2h** FIXME!
- physical functioning and paragraph on history of weather radar
- reflectivity products, 2D and 3D
- !FIXME **10-12.6, 2h** FIXME!
- from reflectivity to RR, limitations of reflectivity
- double-polarization weather radars and Polarimetric products: info on microphysical processes

$$z = AR^b \tag{2.1}$$

where z is reflectivity, R is rainrate, and A as well as b are empirically determined parameters. Reflectivity is calculated

2.1.2 Overview of weather forecast methods

!Fixme **17-18.6 using review paper 3h, 1 page** Fixme!

- Taking a broad look into the NWP process: Data collection, data assimilation, numerical prediction, postprocessing into forecast products
- NWP in the context of precipitation forecasts

2.1.3 Precipitation nowcasting : classical deterministic methods

!Fixme **15.6-16.6 4h, 2 pages** Fixme!

- Basic principles of advection equations.
- Chronological advancing, with +/- of each method

2.1.4 Precipitation nowcasting : classical probabilistic methods

!Fixme **19-20.6 4h, 1 page** Fixme!

- approach types, ensemble etc

2.1.5 Machine learning approaches to precipitation nowcasting

!Fixme **21-22.6 5h, 1.5-2 pages** Fixme!

-

2.2 Bayesian deep learning

2.2.1 Learning probability distributions

!Fixme **15-17.6, 1-2 pages on basic principles, history** Fixme!

- What motivation, first old studies
- overview on Intractable integrals and ways to deal with them
- MCMC, etc...
- finish with switch to VI

2.2.2 Variational inference

!FIXME **18-21.6, 3-4 pages** FIXME!

- deriving ELBO
- Bayes by Backprop and SVI
- Local reparametrization
- Monte-Carlo dropout as VI
- where else is VI used
- problems with VI

2.2.3 Predictive uncertainty estimation and decomposition

!FIXME **9-14.6, 3h, main recent literature, find good ref on sources, 1 page** FIXME!

- Where does uncertainty come from
- division into epistemic, aleatoric uncertainty in ML
- Division for classification
- Division for regression

2.3 Related work

!FIXME **Section to include if 2.1.4 and 2.1.5 become convoluted** FIXME!

Chapter 3

Materials and Methods

!FIXME **estimate : 10-12 pages** FIXME!

3.1 Dataset and data selection

As input data we use lowest elevation angle radar reflectivity composites with 1km spatial resolution and 5 minute temporal resolution from the Finnish Meteorological Institute, cropped into a 512x512 km region covering southern Finland.

Because lowest elevation angle horizontal reflectivity is a good estimator of precipitation at ground level, it is a natural data choice for building a nowcasting model. Also, composites made of these products are readily available at FMI which facilitated their retrieval for this work. Additionally, the bounding box covering southern Finland did not have any major data quality issues as opposed to other candidate datasets such as TAASRAD19, and missing data was labeled. All of these factors contributed to the choice of the dataset.

The dataset was chosen so that at first, a selection of rainy days were chosen as to correspond to the 100 days with the most pixels exceeding a 35 dBZ reflectivity threshold during the summer period spanning from May to September during years 2019, 2020, and 2021.

This dataset is then cleaned and filtered, after which it is divided into training, validation, and test sets. Cleaning the data involves first going through all timestamps and removing those with either partially or completely missing data. The filtering part consists of removing timestamps with less than 1% of pixels containing reflectivity values exceeding 20 dBZ. Splitting of data into training, validation, and testing sets is performed by using a block sampling strategy [12] with 6 hour long blocks to prevent autocorrelation between consecutive radar images from invalidating independence between splits. The final split sizes were 15840 radar images for the training split, 2664 for the validation split, and 2448 for the test split.

Radar composites are stored as gzip-compressed PGM composite files holding uint8 values, which are converted to reflectivity (dBZ) by applying the relation $\text{pixel\_value}_{\text{dBZ}} = -32 + 0.5 * \text{pixel\_value}_{\text{uint8}}$. For being fed into neural networks, the composites are read from HDF5 files as this improves reading speed on distributed storage systems.

From radar reflectivities, rainrate estimates were calculated using Eq. 2.1 solved for R, with empirically determined parameters $A = 223$ and $b = 1.53$, and $z = 10^{Z/10}$, giving us

$$R = (10^{Z/10}/223)^{1/1.53} \tag{3.1}$$

!FIXME **15.6 Add domain presenting radar image, 2h** FIXME!
!FIXME **1.5-2 pages** FIXME!

3.2 Model

3.2.1 The baseline: RainNet

!FIXME **13.6, around 3h + (2-10) some hours for diagrams; take your time for them, 1 page** FIXME!

For the implementation of a Bayesian Convolutional Network, we use as our base model RainNet [2], which is itself heavily inspired by UNet and SegNet model families.

!FIXME **Add RainNet training and/or architecture diagrams** FIXME!

- Describe loss functions and training procedure
- describe mods from UNet, SegNet, its problems, good points
- describe modifications made for this work

3.2.2 Our model: a bayesian extension to RainNet

!FIXME **14.6, describe main points, fix as you go, 2h, 1 page** FIXME!

- Converting weights into distributions
- Gaussian prior, Gaussian posterior enabling closed-form KL-divergence
- Data likelihood modeling: Homeoscedastic Gaussian likelihood
- Training procedure and the Local Reparametrization Trick used.

3.3 verification methods

3.3.1 Evaluation of nowcast predictive uncertainty

!FIXME **First version 12.6, fix as you go;** $\sim$ **3h, 1 page** FIXME!

- Exceedence probabilities
- Describe mean and 2*std plots
- Describe compoundedness of aleatoric, epistemic uncertainty in our data
- describe ensemble sizes, their effect on results as motivation

3.3.2 Baseline Deterministic Models

!FIXME **10.6 Write models, inspire yourself from paper explainations, 30min per model, 1 page** FIXME!

- Lagrangian persistence (advection-extrapolation)
- S-PROG
- ANVIL
- LINDA-D
- RainNet (c.f. above)

3.3.3 Baseline Probabilistic Models

!FIXME **11.6 1h per model, 1 page** FIXME!

- STEPS
- LINDA-P

3.3.4 Deterministic Prediction skill evaluation metrics

Deterministic prediction skill scores were calculated in order to compare the raw predictive skill of ensemble nowcasts to the skill of deterministic models. Such comparison is an important facet of verification as low skill in deterministic scores would even for an otherwise competent model mean that it would benefit from being complemented by a stronger deterministic model. In the case of ensemble nowcasting, Deterministic scores were calculated for ensemble means. Implemented deterministic metrics are divided into four different categories, roughly complementing each others. These are continuous, categorical, and spatial scores, as well as radially-averaged power spectral density.

Continuous scores scores are as their name suggests distance metrics used for the evaluation of continuously valued predictions, i.e. regression tasks. The continuous scores used are Mean Error (ME) and Mean Absolute Error (MAE). ME is defined as

$$\mathrm{ME} = \frac{1}{N}\sum_{i=1}^{N} y_i - \hat{y}_i, \tag{3.2}$$

where we sum over the $i = 1, \ldots, N$ pixels in the radar image, y denotes ground truth and $\hat{y}$ the prediction made. The principal utility of Mean Error is detection whether predictions are biased towards too low or too high rainrates at a certain point in time. On the other hand,

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i|, \tag{3.3}$$

only cares about the magnitude of the errors in predictions by taking the absolute value, so it gives an idea of the prediction skill over the images *on average*. Nevertheless, this is not enough to accurately assess the skill of a nowcast method, because not all pixels are equally important. Additionally, A poor forecast may have good MAE and vice versa. To illustrate this, a forecast failing to predict localized intense rainfall, but otherwise accurately capturing light rainfall over large areas will usually have a small MAE but will have low operational usefulness.

!FIXME **small picture showing contingency table to help vizualization** FIXME!

This limited utility of continuous scores serves as a motivation to introduce categorical scores. These are based on the principle of comparing the presence or absence of a rain event in observations and predictions as defined by having a pixel exceeding a threshold value R_{THR}. The categorical scores are defined by dividing events into four categories, that are true positives (TP), i.e rain events that were correctly predicted, true negatives (TN), i.e. lack of rain event that was correctly predicted, false negatives (FN), i.e. rain that wasn't successfully predicted, and false positives (FP), i.e rain that was erroneously predicted. Scores derived from those quantities that were used are probability of detection (POD) defined as

$$\mathrm{POD} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{3.4}$$

, which simply tells the probability that an event really occuring is correctly predicted. Next, false alarm rate (FAR) is calculated as

$$\mathrm{FAR} = \frac{\mathrm{FP}}{\mathrm{TP} + \mathrm{FP}} \tag{3.5}$$

which reversely indicates the percentage of positively predicted events not actually happening. Critical success index (CSI), which is defined as

$$\mathrm{CSI} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP}} \tag{3.6}$$

is computed and aims to generally assess the performance of the forecast by taking the proportion of correct positive event predictions out of critically important cases, that is those excluding true negatives but including both false alarms (FP) and misses (FN). Lastly, the equitable threat score (ETS) defined as

$$\begin{aligned} \mathrm{ETS} &= \frac{\mathrm{TP} - rnd}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP} - rnd}, \\ \text{where } rnd &= \frac{(\mathrm{TP} + \mathrm{FN})(\mathrm{TP} + \mathrm{FP})}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP} + \mathrm{TN}} \end{aligned} \tag{3.7}$$

was computed. ETS aims to improve CSI assessment of forecast skill, by attempting to estimate the amount of random TP among the prediction using the term rnd, and remove that number of data points from the calculations.

The R_{thr} thresholds chosen for the performing verification are 0.5, 1.0, 5.0, 10.0, 20.0, and 30.0 mm/h. 0.5 mm/h corresponds to very light rainfall and should be easy to predict, while higher thresholds like 20.0 and 30.0 mm/h correspond to very heavy rainfall, which are very difficult to predict even for short leadtimes.

- Spatial scores: FSS, Intensity scale using FSS
- Radially averaved PSD

In addition to evaluating prediction skill above rainrate thresholds, it is also important to be able to perform evaluation at multiple scales. The reason for this is that bigger scales have more predictability, and so predicting smaller scales is more difficult while also being of high importance, especially in the context of heavy localized rainfall.

As such, spatial verification scores, namely Fraction Skill Score (FSS) and Intensity-scale verification using FSS are added to the panoply of metrics used.

3.3.5 Probabilistic Prediction skill evaluation metrics

!FIXME **8.6, write CRPS 45min, 1.5 pages** FIXME!

!FIXME **When implementing write the rest in** $\sim$ **3h** FIXME!

- Continuous rank probability score (CRPS) as an extension to MAE
- Rank histogram
- Reliability diagram
- ROC curve

3.3.6 Practical points regarding verification experiments

All of the scores described in the previous sections complement each others, and forecast skill is thus estimated as a combination of them, as no score is able to capture all of the facets of a great nowcast.

In practice, predictions are calculated for all models for all the timestamps contained in the test set described in section 3.1. Predictions are computed for 36 timesteps, which is equivalent to 3 hours into the future with an interval of 5 minutes. After that, verification metrics are computed using the predictions of each model and they are averaged over the set for each leadtime of interest.

In order to make sure that results are valid, all timestamps having any (even a single) observation missing are discarded, and similarly timestamps having any prediction from any model missing are also removed from calculations. Additionally, only pixels where data is present in the predictions of all models are counted in metric calculations. This is accomplished by calculating a common NaN mask using the logical OR operation over NaN values of each model, and subsequently applying that common mask to each prediction.

3.4 Software and resources

!FIXME **Some diagram for visualizing grand scheme of workflow** FIXME!

The deep learning models were all implemented using the PyTorch framework and the PyTorch Lightning wrapper [5]. These libraries were chosen because of the combined ease of prototyping brought by PyTorch Lightning, and the maturity and flexibility of the parent framework PyTorch. RainNet was ported to PyTorch following the original Tensorflow implementation by Ayzel et al. [1].

As for the implementation of the probabilistic inference mechanisms for Bayesian Neural Networks, choice was made not to implement them by hand, but to rely on the machinery contained in the Probabilistic Programming Language (PPL) Pyro [3], which is itself built on top of PyTorch and includes a fully-featured implementation of Stochastical Variational Inference (SVI). In order to facilitate the implementation task, the TyXe package [11] was used. TyXe is a library designed to provide an interface simplifying the implementation of Bayesian Neural Networks using PyTorch and Pyro. A few of the reasons why TyXe was chosen are that it permits easily turning existing neural networks into BNNs without having to use hard-coded bayesian layers, and dynamically switching on-and-off the local reparametrization trick in layers. Some problems encountered include components of TyXe having difficulties working together with PyTorch Lightning abstractions.

Verification experiments and non-deep-learning baseline models were ran and implemented using the open-source library Pysteps [9]. It provides implementations for all non-deep-learning models described as well as implementation of verification metric primitives used in this work, and tools for their visualization.

The computational resources from the Finnish IT Center for Science (CSC) were used for GPU intensive tasks such as training and calculating predictions with deep-learning models, and one of FMI's computational servers was used for performing other, more CPU-intensive operations such as predicting with baseline non-deep-learning models and calculating verification metrics. We trained the models on the CSC Puhti supercomputer, using one Nvidia V100 GPU with 32GB of VRAM, 64GB of RAM, and 10 cores from a 2.1 GHz Intel Xeon Gold 6230 CPU. As for the FMI server, it contains 2 Intel Xeon Gold 6138 2.0 GHz CPUs with each 20 cores and 2 threads by core, with 192GB of RAM.

Chapter 4

Results

4.1 Case studies for nowcasts

4.1.1 Case study 1 : Large-scale Stratiform rain event

4.1.2 Case study 2 : Rapidly evolving convective rain event

4.2 Deterministic prediction skill (Metrics)

4.3 Probabilistic prediction skill (Metrics)

4.4 Uncertainty estimation

4.4.1 Uncertainties against leadtime

4.4.2 Uncertainties against rainfall intensity

4.4.3 Epistemic uncertainty against training parameters

Chapter 5

Discussion

5.1 Goodness of results

5.2 Validity of results

5.3 What could we learn from uncertainty

5.4 What would have to be improved, potential problems in the study?

5.5 Directions for further work

Chapter 6

Conclusions

Chapter 7

references

[1] Ayzel, G. Rainnet: a convolutional neural network for radar-based precipitation nowcasting. `https://github.com/hydrogo/rainnet`, 2020.

[2] Ayzel, G., Scheffer, T., and Heistermann, M. RainNet v1.0: a convolutional neural network for radar-based precipitation nowcasting. 21.

[3] Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. Pyro: Deep Universal Probabilistic Programming. *Journal of Machine Learning Research* (2018).

[4] Bowler, N. E., Pierce, C. E., and Seed, A. W. STEPS: A probabilistic precipitation forecasting scheme which merges an extrapolation nowcast with downscaled NWP. *Quarterly Journal of the Royal Meteorological Society 132*, 620 (2006), 2127–2155. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1256/qj.04.100.

[5] Falcon, W., and The PyTorch Lightning team. PyTorch Lightning, 3 2019.

[6] Kwon, Y., Won, J.-H., Kim, B. J., and Paik, M. C. Uncertainty quantification using Bayesian neural networks in classification: Application to biomedical image segmentation. *Computational Statistics & Data Analysis 142* (Feb. 2020), 106816.

[7] McAllister, R., Gal, Y., Kendall, A., Wilk, M. v. d., Shah, A., Cipolla, R., and Weller, A. Concrete Problems for Autonomous Vehicle Safety: Advantages of Bayesian Deep Learning. 4745–4753.

[8] Pulkkinen, S., Chandrasekar, V., and Niemi, T. Lagrangian Integro-Difference Equation Model for Precipitation Nowcasting. *Journal of Atmospheric and Oceanic Technology 38*, 12 (Dec. 2021), 2125–2145. Publisher: American Meteorological Society Section: Journal of Atmospheric and Oceanic Technology.

[9] Pulkkinen, S., Nerini, D., PÃ©rez Hortal, A. A., Velasco-Forero, C., Seed, A., Germann, U., and Foresti, L. Pysteps: an open-source Python library for probabilistic precipitation nowcasting (v1.0). *Geoscientific Model Development 12*, 10 (Oct. 2019), 4185–4219. Publisher: Copernicus GmbH.

[10] Ravuri, S., Lenc, K., Willson, M., Kangin, D., Lam, R., Mirowski, P., Fitzsimons, M., Athanassiadou, M., Kashem, S., Madge, S., Prudden, R., Mandhane, A., Clark, A., Brock, A., Simonyan, K., Hadsell, R., Robinson, N., Clancy, E., Arribas, A., and Mohamed, S. Skilful precipitation nowcasting using deep generative models of radar. *Nature 597*, 7878 (Sept. 2021), 672–677. Number: 7878 Publisher: Nature Publishing Group.

[11] Ritter, H., and Karaletsos, T. TyXe: Pyro-based Bayesian neural nets for Pytorch. *arXiv preprint arXiv:2110.00276* (2021).

[12] Schultz, M. G., Betancourt, C., Gong, B., Kleinert, F., Langguth, M., Leufen, L. H., Mozaffari, A., and Stadtler, S. Can deep learning beat numerical weather prediction? *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 379*, 2194 (Apr. 2021), 20200097. Publisher: Royal Society.

Appendix A

First appendix

This is the first appendix. You could put some test images or verbose data in an appendix, if there is too much data to fit in the actual text nicely.

For now, the Aalto logo variants are shown in Figure A.1.

(a) In English

(b) Suomeksi

(c) På svenska

Figure A.1: Aalto logo variants