Probabilistic Wind Park Power Prediction using Bayesian Deep Learning and Generative Adversarial Networks

Lars Ødegaard Bentsen[1], Narada Dilp Warakagoda[1], Roy Stenbro[2] and Paal Engelstad[1]

EERA DeepWind Conference, 19 - 21 January, 2022

1. Department of Technology Systems, University of Oslo.
2. Institute for Energy Technology (IFE).

ELOGOW (Electrification of Oil and Gas Installation by Offshore Wind) – Funded by The Research Council of Norway (project nr: 308838) under the PETROMAKS2 program with financial support of Equinor Energy AS, ConocoPhillips Skandinavia AS and Aibel AS.

UNIVERSITY OF OSLO

Motivation and Methodology

- **Dataset**: Historical data collected at Germany's first off-shore wind farm, **Alpha Ventus***.

- **Real datasets** are characterised by **uncertainty**.

  ⟶ We **propose three models** for predicting **power probability distributions**, which should be more informative than point estimators.

  ⟶ Inputs to the models are the wind speed, turbine type and location of upstream neighbours.

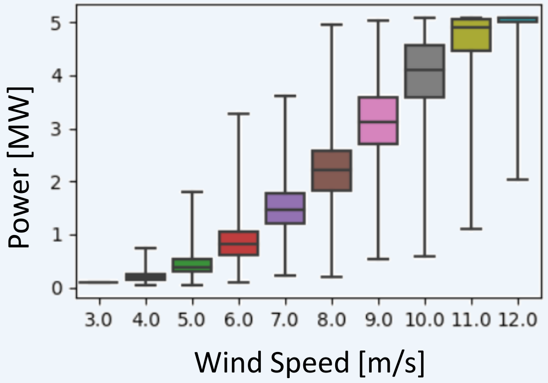

Figure 1: Example showing some of the uncertainties associated with wind power prediction.

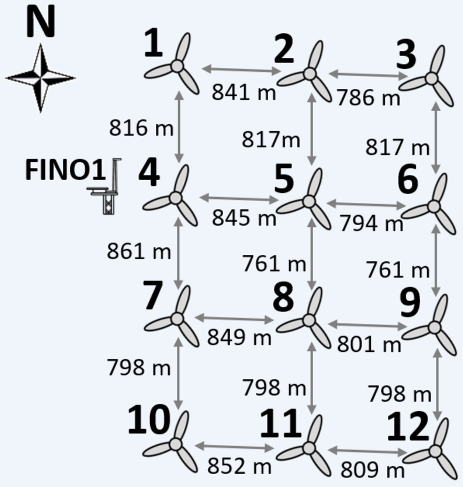

Figure 3: Alpha Ventus

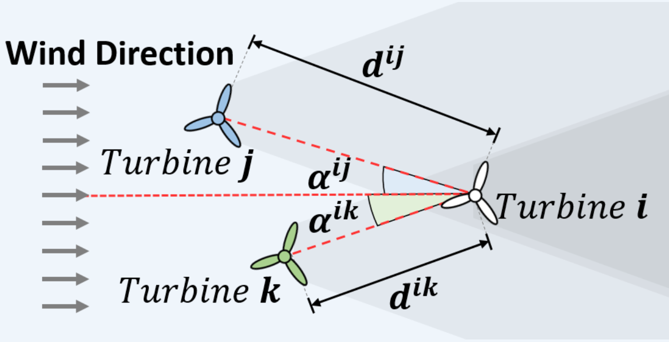

Figure 2: Illustration showing a turbine, i, being downstream of two turbines j and k.

* *The dataset was made available by the RAVE initiative, which was funded by the German Federal Ministry of Economic Affairs and Energy (see: www.rave-offshore.de).*

UiO : **Department of Technology Systems**
University of Oslo
For any queries: lars.bentsen@its.uio.no

Bayesian Models

- **Flipout** (Wen *et* al.) was employed for **Bayesian Variational Inference.**
- **BMLP** – A multilayer perceptron formed by stacked DenseFlipout Layers.
  - $\hat{p}_t^{(i)} = f\left(ws_t, l_i, d^{ik}, \sin\left(\alpha_t^{ik}\right), \cos\left(\alpha_t^{ik}\right), l_k, d^{ij}, \sin\left(\alpha_t^{ij}\right), \cos\left(\alpha_t^{ij}\right), l_j\right),$

    where: ws_t - Wind Speed, l – Turbine type, d – Distance, α – Angle.
- **BGNN**: - A GNN, where the update functions, $\phi^{(\cdot)}$, are MLPs formed by stacked DenseFlipout Layers.

  Inputs to the network is a **graph tuple, $\boldsymbol{G(u, V, E)}$**:
  - $\boldsymbol{u}$ – Global feature shared by the graph: $\boldsymbol{u = ws_t}$
  - $\boldsymbol{V}$ – The node set containing node specific features: $\boldsymbol{v_i = l_i}$
  - $\boldsymbol{E}$ – The edge set containing edge features: $\boldsymbol{e_{ij} = [d^{ij}, \sin\left(\alpha_t^{ij}\right), \cos\left(\alpha_t^{(ij)}\right)]}$

  Outputs an updated graph tuple, $\boldsymbol{G(\_, \widehat{P}, \_)}$, where **individual turbine power predictions** are embedded in the updated node features.

What is a Graph?

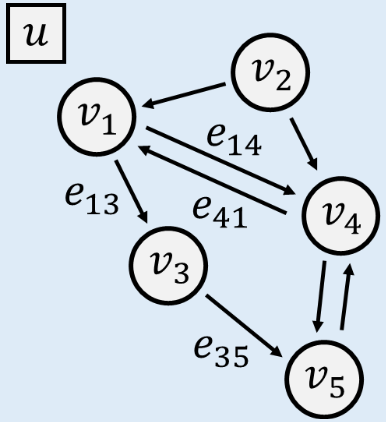

Figure 4: Graph Representation.

What is a GNN?

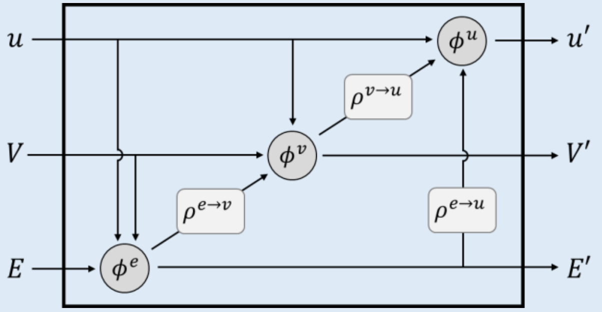

Figure 5: Graph Block in a GNN.

UiO : Department of Technology Systems
University of Oslo
For any queries: lars.bentsen@its.uio.no

(Wen *et al.*): Wen, Y., Vicol, P., Ba, J., Tran, D. and Grosse, R., Flipout: Efficient pseudo-independent weight perturbations on mini-batches. *International Conference on Learning Representations*, 2018 (https://arxiv.org/abs/1803.04386).

cStackWGAN - *Conditional Stack Wasserstein Generative Adversarial Network*

- **WGAN** – Generator and Discriminator play a minimax two-player game:

$$\min_{G}\max_{D} V(G, D) = \mathbb{E}_{x \sim p_{data}(x)}[D(x\,;\theta_D)] - \mathbb{E}_{z \sim p_z(z)}[D(G(z\,;\theta_G)\,;\theta_D)]$$

- Each **Generator and Discriminator** are represented by **GNNs**, with MLPs as update functions.
- Conditioned on the wind speed, turbine type and location of upstream neighbours.

  → *i.e. same inputs as for the BGNN model.*

- The motivation behind a **stacked** architecture is to divide a complex task into **multiple simpler tasks**, each solved by individual Discriminator – Generator pairs.

  Stage1:
  - **G1** – Predicts mean and standard deviation of entire farm power.
  - **D1_mean** – Classify mean samples as real or fake.
  - **D2_std** – Classify standard deviation samples as real or fake.

  Stage2:
  - **G2** – Predict individual turbine powers, given G1 prediction and condition.
  - **D2** – Classify individual turbine powers as real or fake.

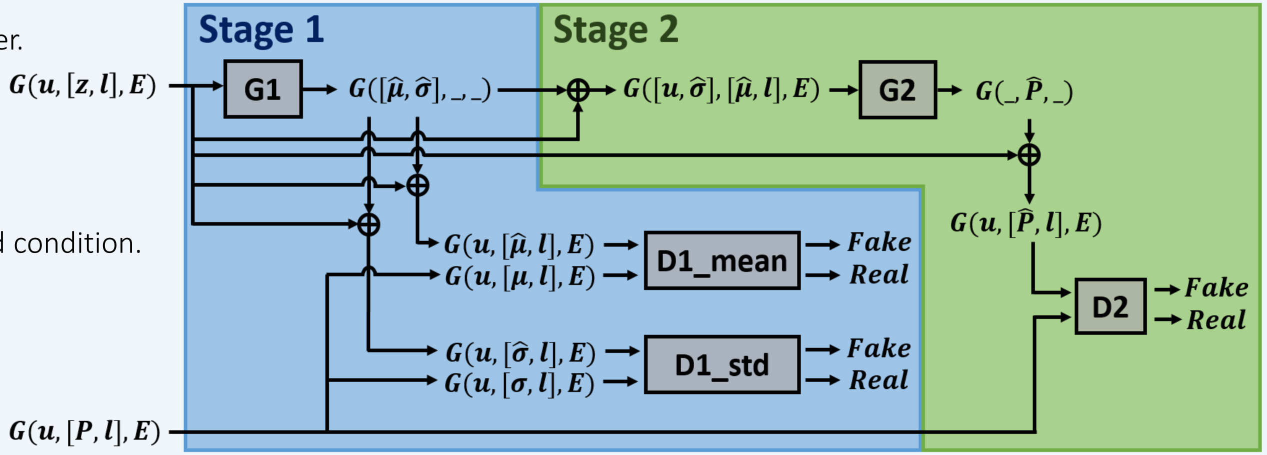

Figure 6: Proposed cStackWGAN model.

UiO : **Department of Technology Systems**
University of Oslo
For any queries: lars.bentsen@its.uio.no

Results

1. *Bayesian models performed the best with regards to* ***MAEs****, with* ***the BGNN performing the best.*** *However, as more points were used to construct the true* ***distributions****, the* ***cStackWGAN performed the best****.*

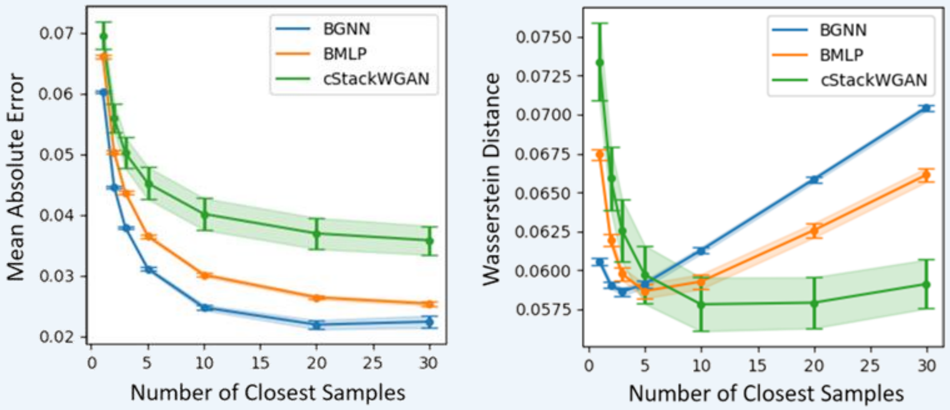

Figure 7: MAEs and mean Wasserstein distance against the number of closest samples, based on the wind condition, in the test set used to estimate the true distributions.

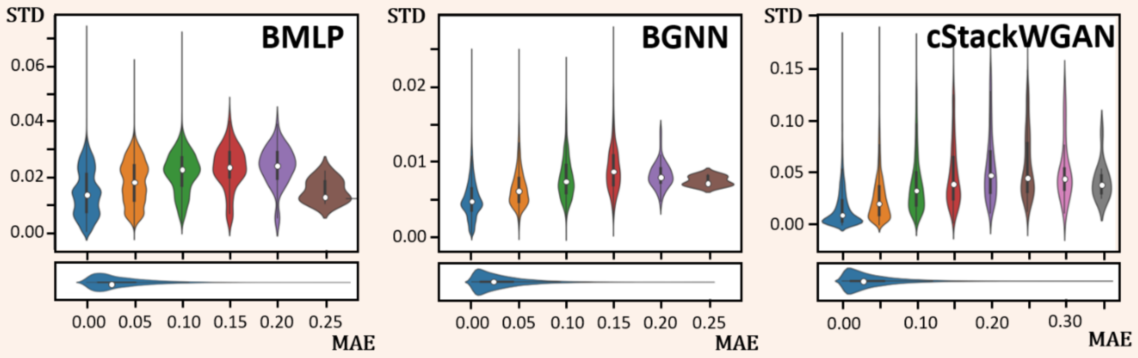

Figure 8: Standard deviation of predicted distributions against MAEs.

2. *A* ***positive correlation*** *between predicted standard deviations and MAEs was observed for all models, showing that the models were able to give an indication of the associated uncertainties with particular predictions.*

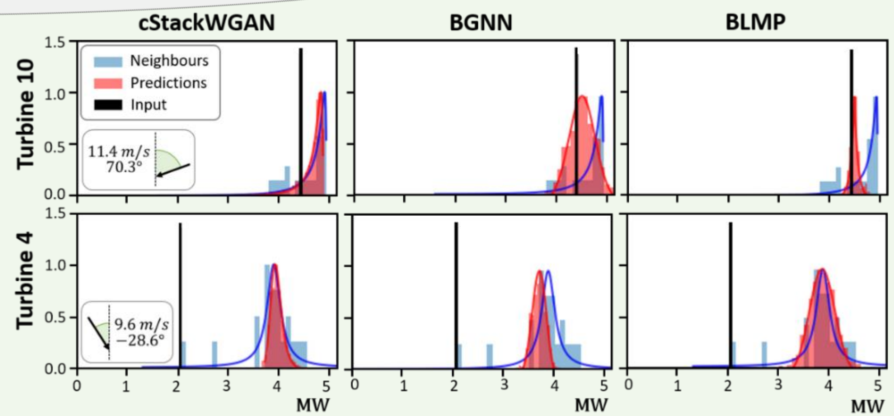

Figure 9: Predicted (red) and true (blue) distributions for turbines 4 and 10 for two randomly selected wind conditions. Histograms were scaled in (0, 1), while the powers for the inputs used to compute the predicted distributions is shown by a single black bin, with a maximum of 1.5. True distributions were taken as the 20 closest neighbours to the inputs, determined based on the wind condition, while the predicted were constructed by sampling 1000 times from the models for the same input condition.

3. *All models were able to predict the probability distributions for turbine powers reasonably well, with the* ***cStackWGAN*** *seeming the best at capturing standard deviations and generating* ***more complex distributions****.*

UiO : Department of Technology Systems
University of Oslo
For any queries: lars.bentsen@its.uio.no