



Detection of Wind Power Ramp Events under Severe Class Imbalance

Afif Ayoub Supervisors: Prof. Gianluca Bontempi, Dr. Alejandro Morales Hernández

0.713

Best MCC (Balanced Neural Network), vs 0.038 for a probabilistic baseline

0.107

Lowest Balanced Error Rate (Balanced Neural Network), vs 0.653

0.858

Best weighted F1-score (Easy Ensemble), vs 0.614 for the baseline

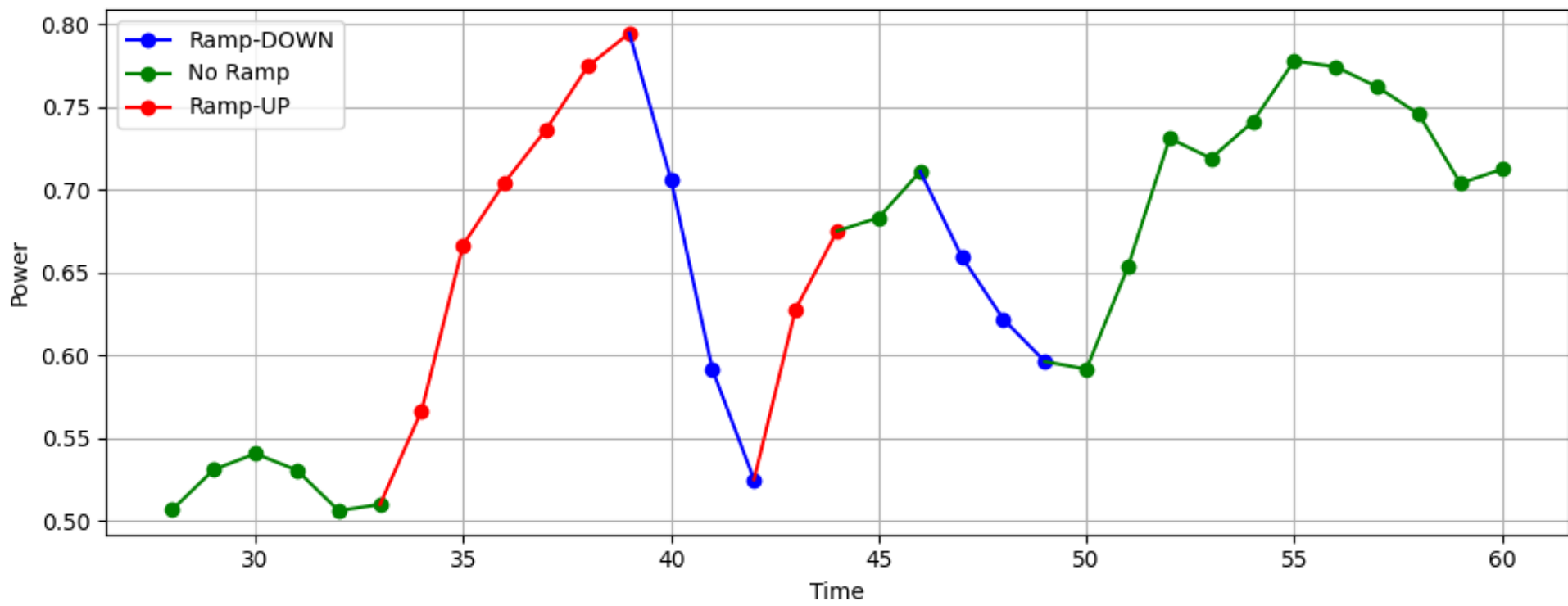
95%

of ramp-up segments detected (recall 0.952, Balanced Neural Network)

1 Motivation

Wind Power Ramp Events (WPRES) are sudden, large increases (ramp-up) or decreases (ramp-down) in wind power output over a short period, typically caused by passing fronts or fast changes in wind direction.

They cause grid frequency and voltage fluctuations, accelerate wear on turbines and storage, and raise operating costs. Operators need to recognise them as they happen.



Ramp-up, ramp-down and no-ramp segments in a wind power time series.

2 Problem and objectives

Late detection

The Swinging Door Algorithm (SDA) and its variants only label a ramp once it has ended.

Class imbalance

Ramps are rare, so standard classifiers drift toward the normal class and miss them.

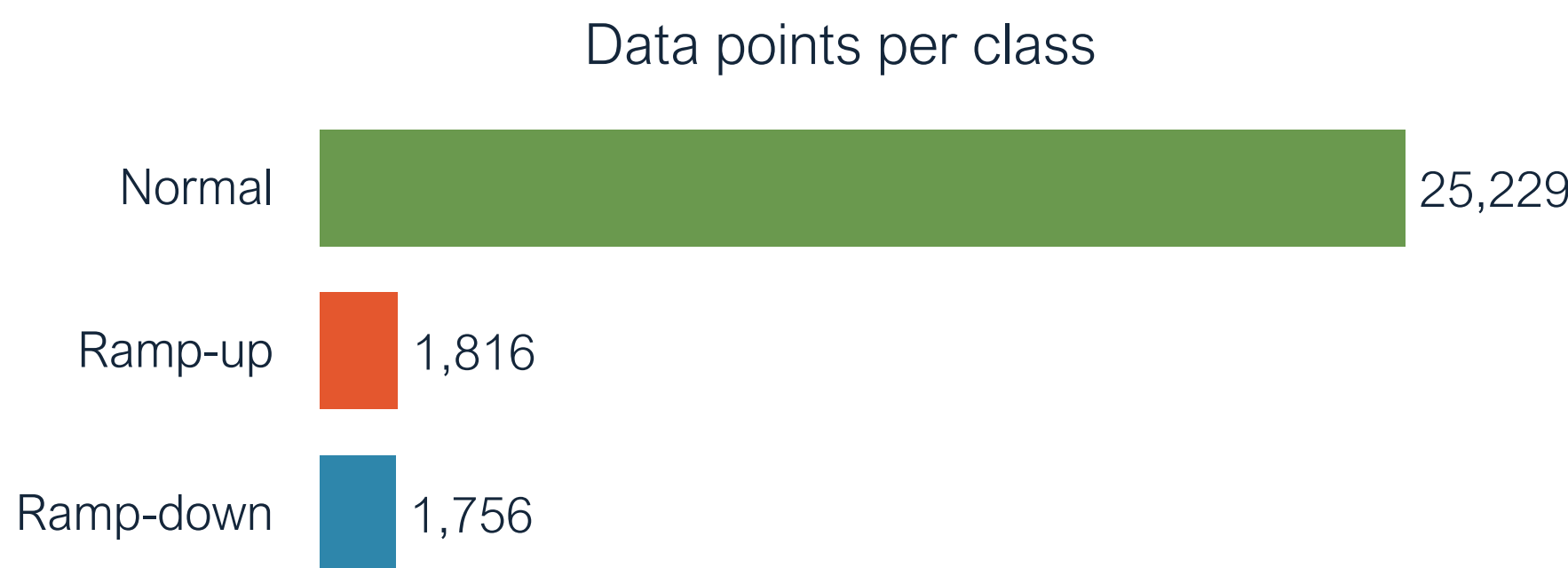
Goal: detect ramp events as early as possible, as a three-class problem (ramp-up, ramp-down, normal), with models built to handle imbalance.

Research questions

- How can ramp events be detected in real time?
- Which strategies best address class imbalance in ramp data?

3 Dataset

Aggregated output of a wind farm, sampled every 15 minutes (about 28,800 points), from a collaboration with the Italian Transmission System Operator and the University of Sannio. SDA labels, five classes merged into three.



4,107 SDA segments. Ramp segments are short (mean 3.1 to 3.3 steps, max 12). Normal segments average 8.4 steps (median 5, max 449), so the window length λ must stay small.

4 Detection pipeline

1 SDA segmentation

Split the power series into linear SDA segments labeled ramp-up, ramp-down or normal.

2 Feature extraction

22 features per segment (TSFEL): statistical, temporal, fractal, plus length and previous class.

3 Fixed-length windows

Cut each segment into windows of λ power values (padded with -1) and append the segment's features.

4 Imbalance-aware classifier

Classify each window independently. In training, windows inherit their segment's label.

5 Majority vote

Aggregate window predictions into one label per segment (ties broken at random).

5 Imbalance-aware models

Easy Ensemble

Balanced subsets of the majority class, each boosted with AdaBoost. Tuned: 92×92 trees, depth 7.

Balanced Random Forest

Each tree grown on a balanced bootstrap sample. Tuned: 72 trees, max depth 42.

Balanced Bagging

Bagging with under-sampling at each bootstrap. Tuned: 7 strongly regularized trees.

Balanced Neural Network

MLPs on under-sampled balanced subsets, majority vote. Tuned: 18 MLPs, 1 layer of 69 neurons.

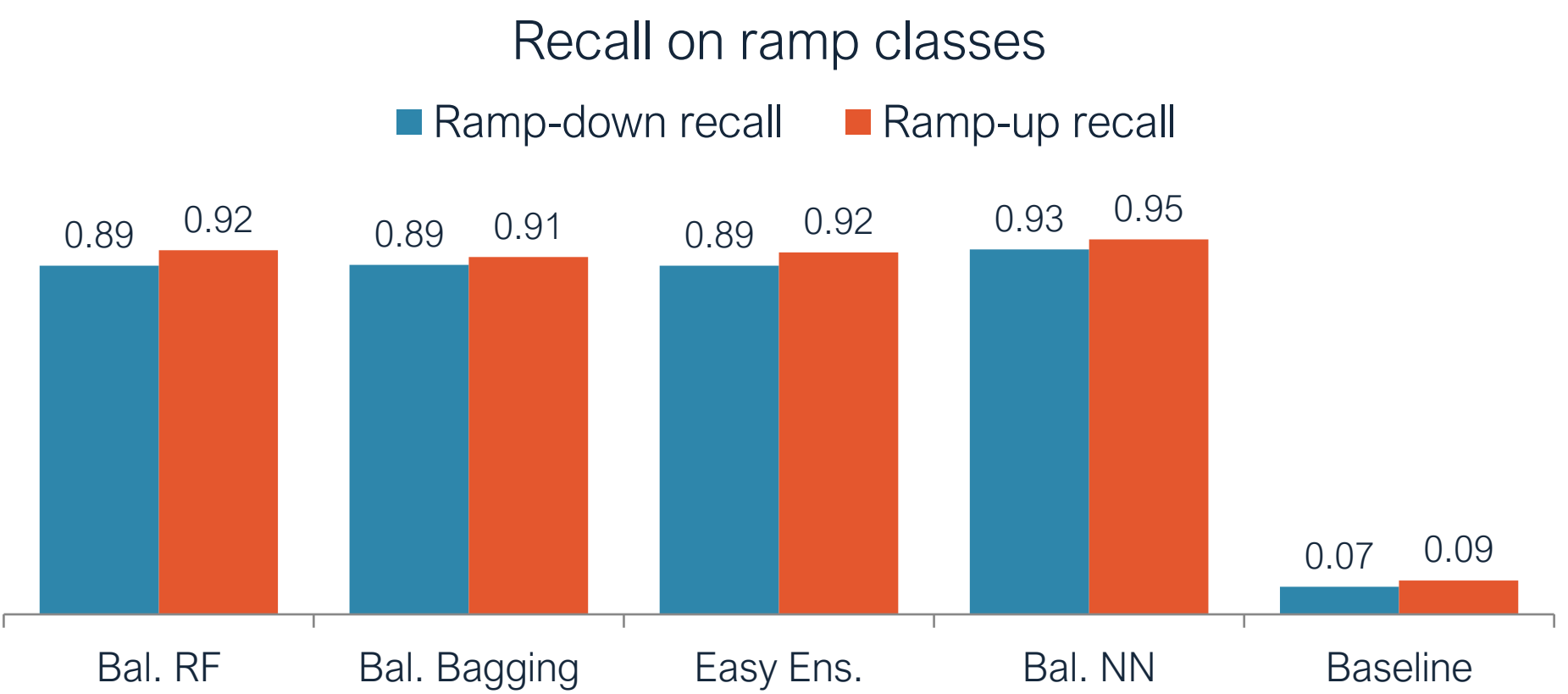
6 Experimental setup

- Baseline: probabilistic classifier sampling the next class from a transition matrix on the previous class.
- 80/20 stratified split. HyperOpt (75 iterations) with stratified 3-fold CV, maximizing weighted F1.
- Metrics robust to imbalance: MCC, Balanced Error Rate (BER), weighted F1, per-class precision, recall, F1, one-vs-rest ROC.
- Gini and permutation importance (100 permutations, balanced accuracy).

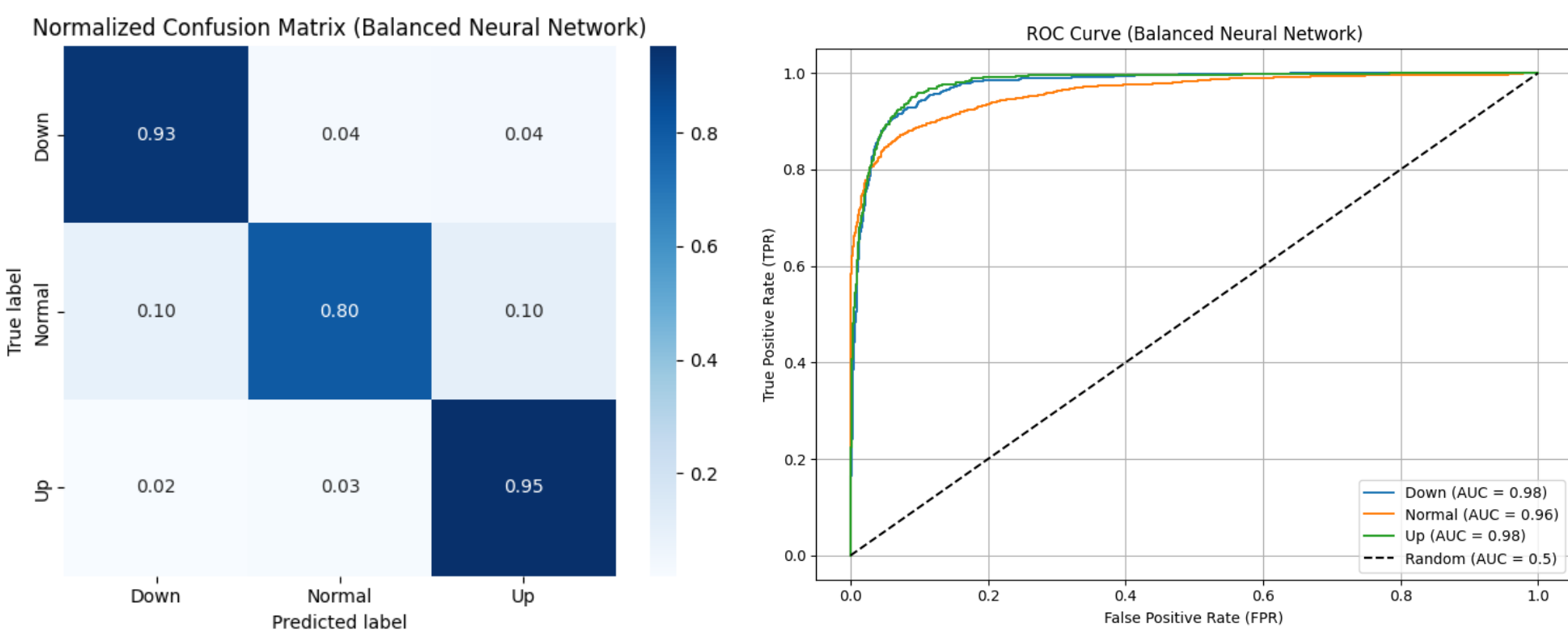
7 Results

Global test-set metrics. Lower BER is better. Best values are highlighted.

Model	Acc.	MCC	BER	W-F1
Balanced RF	0.845	0.705	0.123	0.852
Balanced Bagging	0.850	0.709	0.125	0.857
Easy Ensemble	0.851	0.712	0.121	0.858
Balanced NN	0.839	0.713	0.107	0.848
Baseline	0.665	0.038	0.653	0.614



- All four models reach MCC above 0.70 and BER below 0.13. One-vs-rest AUC on ramp classes is 0.93 to 0.98, vs 0.53 to 0.54 for the baseline.
- Errors mostly confuse a ramp with normal, rarely ramp-up with ramp-down.
- Test-set inference time: 0.007 s (BRF, Bagging), 0.07 s (NN), 1.5 s (Easy Ensemble), far below the 15-minute sampling step.



Confusion matrix, Balanced NN

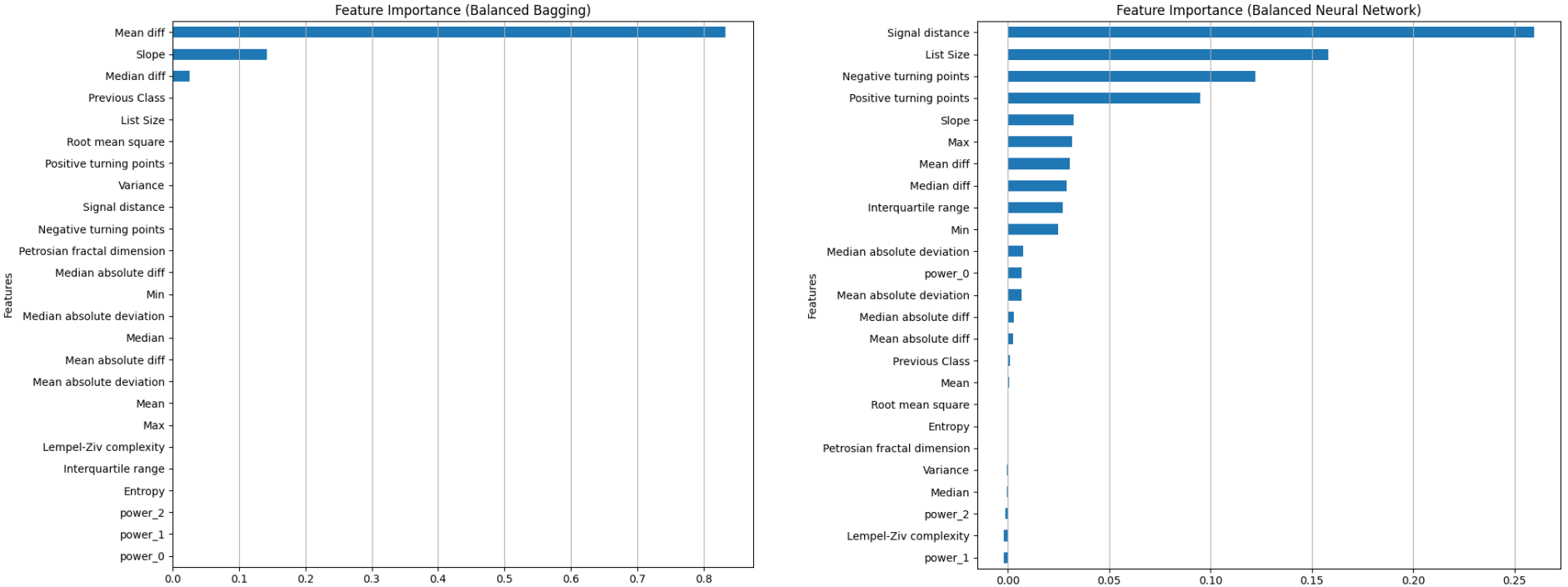
ROC curves, Balanced NN

8 What drives detection?

Tree ensembles rely on local change: mean_diff dominates (over 70% of Gini importance in Easy Ensemble, over 80% in Balanced Bagging), then slope.

The neural network spreads importance across signal distance, segment length and turning points.

Raw power values and previous class add little, so the window length λ barely affects results.



Gini importance, Bal. Bagging

Permutation importance, Bal. NN

Conclusions

- Ramp-up, ramp-down and normal segments can be separated with balanced performance despite severe imbalance (MCC up to 0.713, ramp recall of at least 0.885 for every model).
- No single best model: the Balanced NN misses the fewest ramps, Easy Ensemble gives the best accuracy and weighted F1, Balanced Bagging and BRF come close at the lowest cost.
- Detection is driven by descriptors of local change (mean difference, slope), not by absolute power levels.

Future work

- Sequence models (LSTM, GRU, Transformers) on raw power
- Meteorological inputs: wind speed, direction, pressure
- Validation on other wind farms and climates
- Online deployment with incremental learning