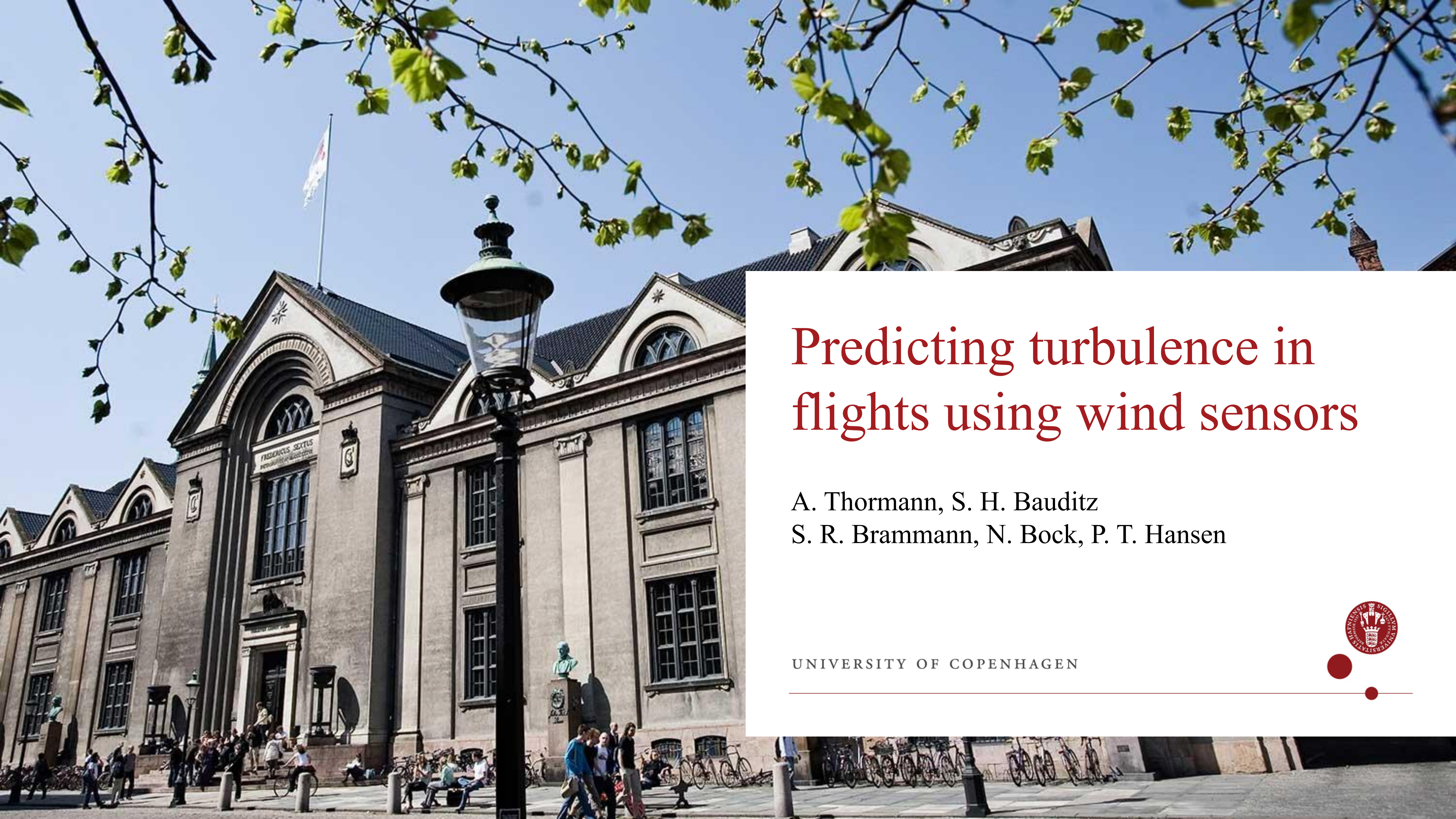


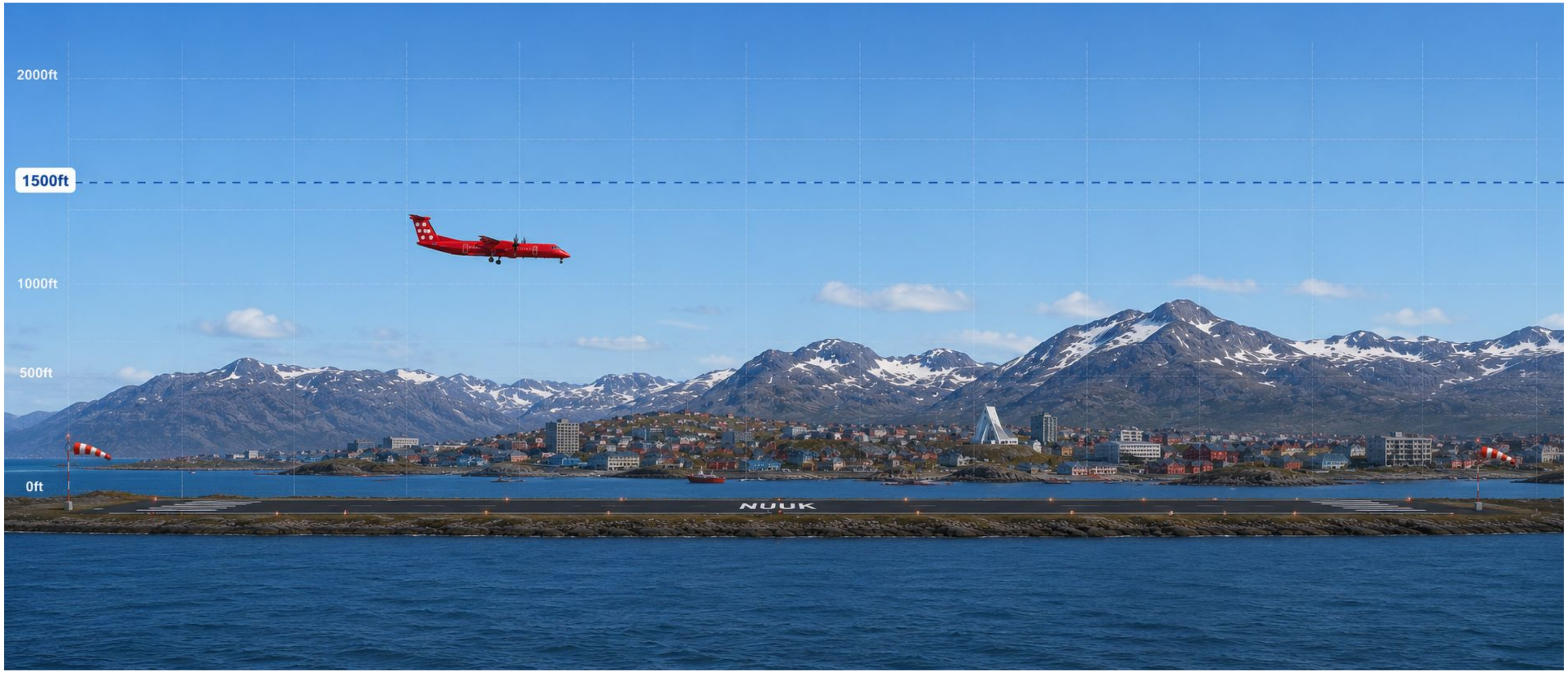
Predicting turbulence in flights using wind sensors

A. Thormann, S. H. Bauditz
S. R. Brammann, N. Bock, P. T. Hansen

UNIVERSITY OF COPENHAGEN

Introduction

Goal:
Estimate the level of turbulence in airplanes from wind sensors at the runway

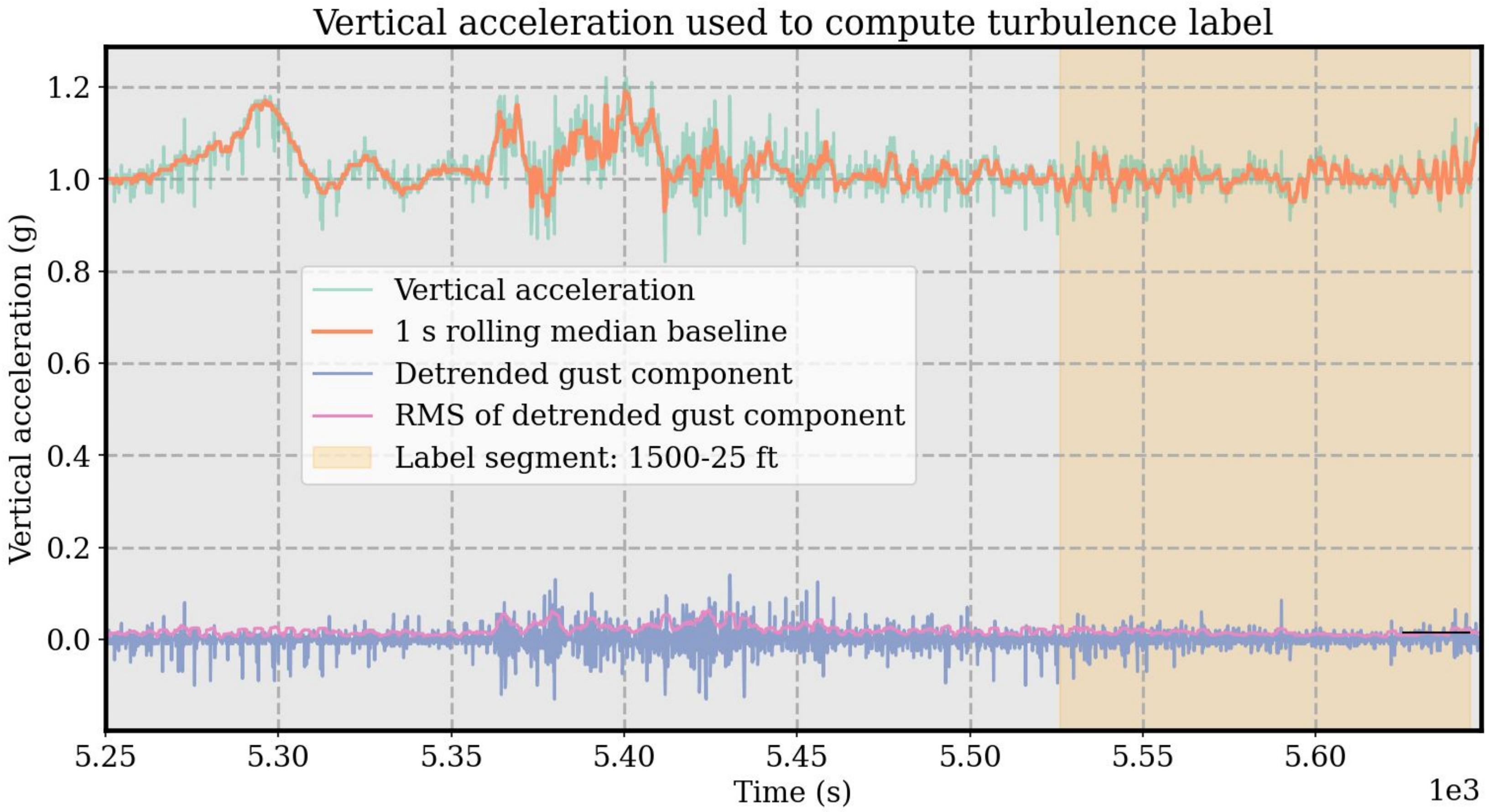


Data and format

Flights:
2099 departures
3139 arrivals

Measure:
Vertical acceleration time series
Features: Static -> runway, arrival/departure, flight type etc.

Metric:
RMS of vertical acceleration in intervals

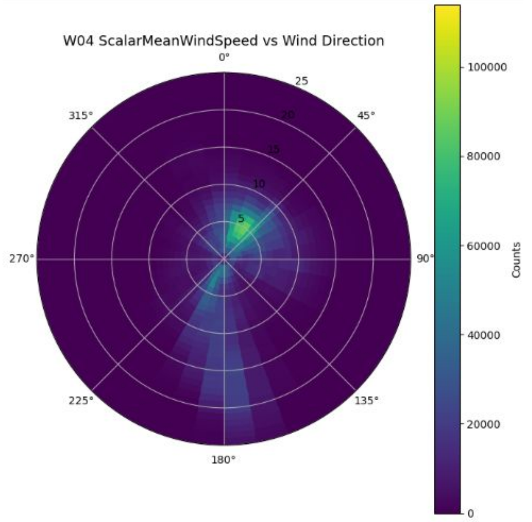


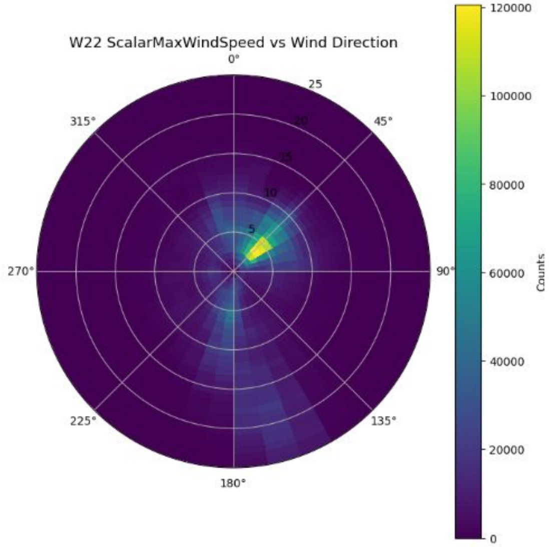
Wind data

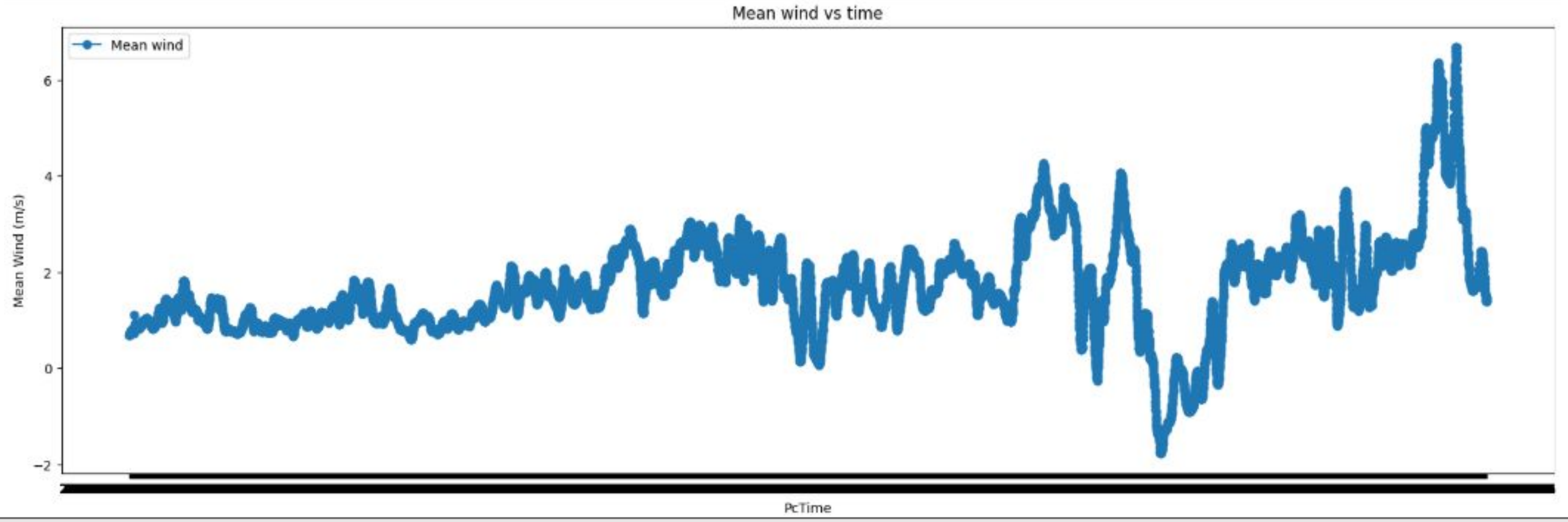
- 540 full days

- Two wind anemometer and different
  wind and direction measurements - ect. W04.ScalarMeanWindSpeed

- Static features -> year, month, day ect.



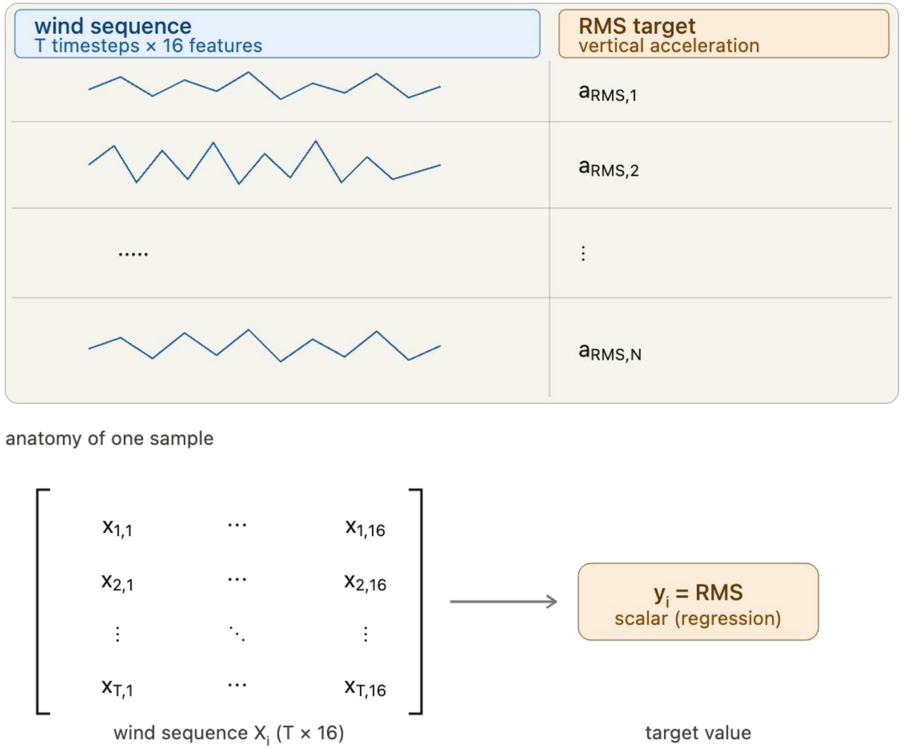




LSTM and GRU data

- Data structure:
  - 16 wind features (6 constructed)
  - 5238 time series
  - 16 static features
  - 1 target value (Regression)

- Target: RMS acceleration



BDT data

- Features constructed from wind series
- Target: RMS acceleration

Number of flights	Mean wind	Start wind	End wind	.....	RMS Wind	Prediction — RMS acceleration
1	..	..	..	.....	..	..
2	..	..	..	.....	..	..
⋮	..	..	..	.....	..	..
N	..	..	..	.....	..	..

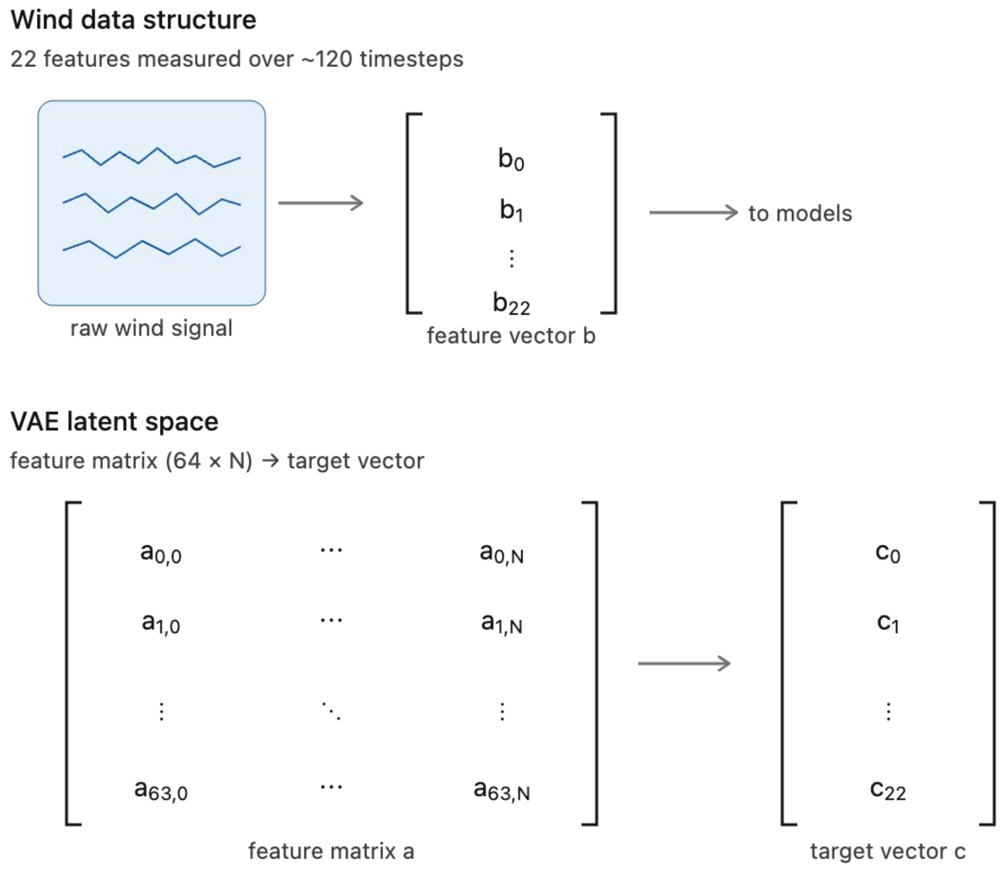
VAE Boosted Decision Tree data

- Wind data structure:
  - Timeframe of 10 minutes ~ 120 indexes ~ until 2 minutes before plane enters 400 ft height
  - 22 wind data features

- Plane data structure
  - Performance of model limited by # of datapoints
  - Turbulence labelled from RMS
    - Binary for classifier, continuous for regressor

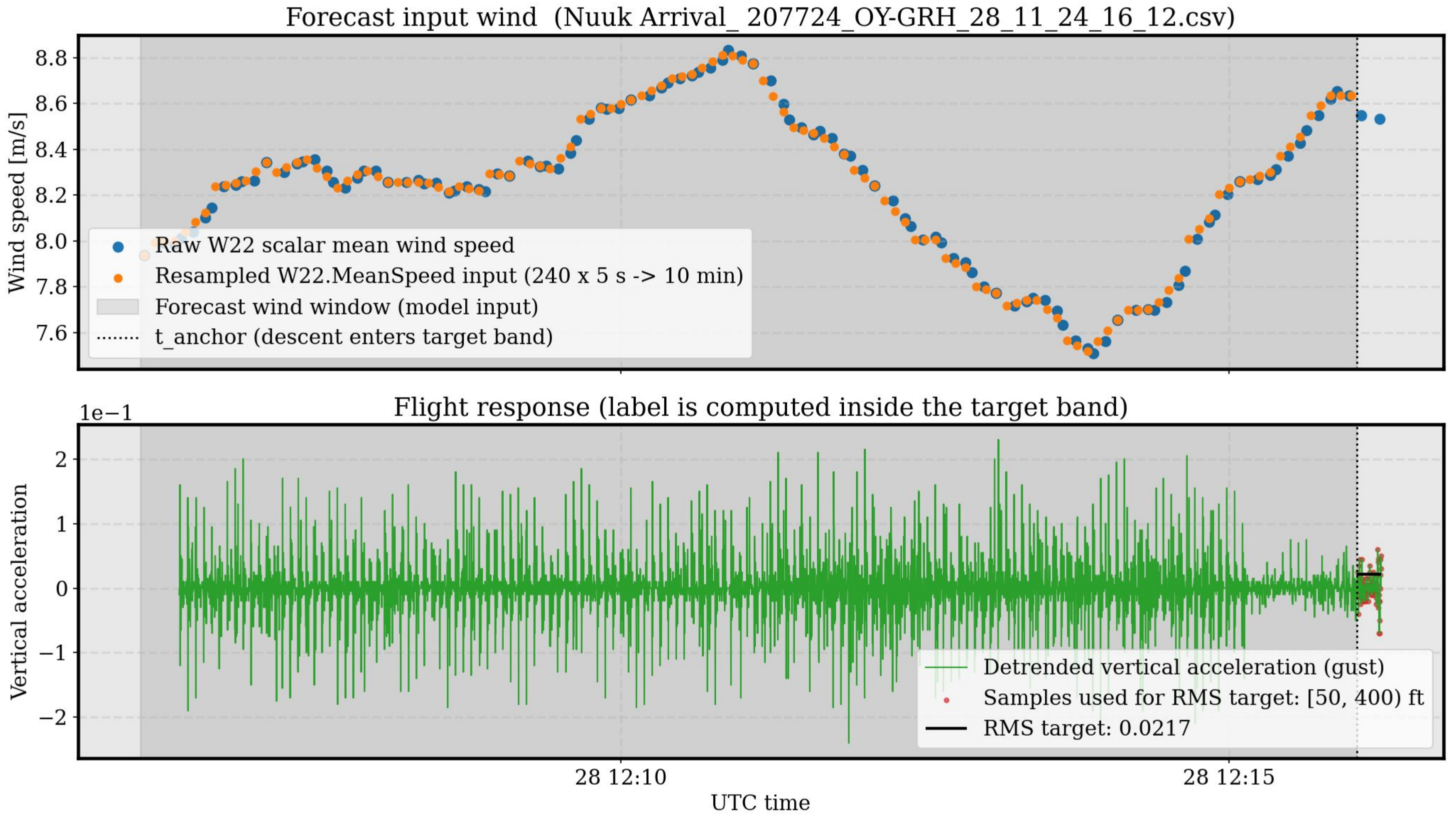
-

- Models
  - VAE: Takes 256 batches consisting of 22 input features and 120 indexes. Reduces each 10 minute interval to a coordinate in 64-dim latent space
  - Data fed into a classifier and a regressor

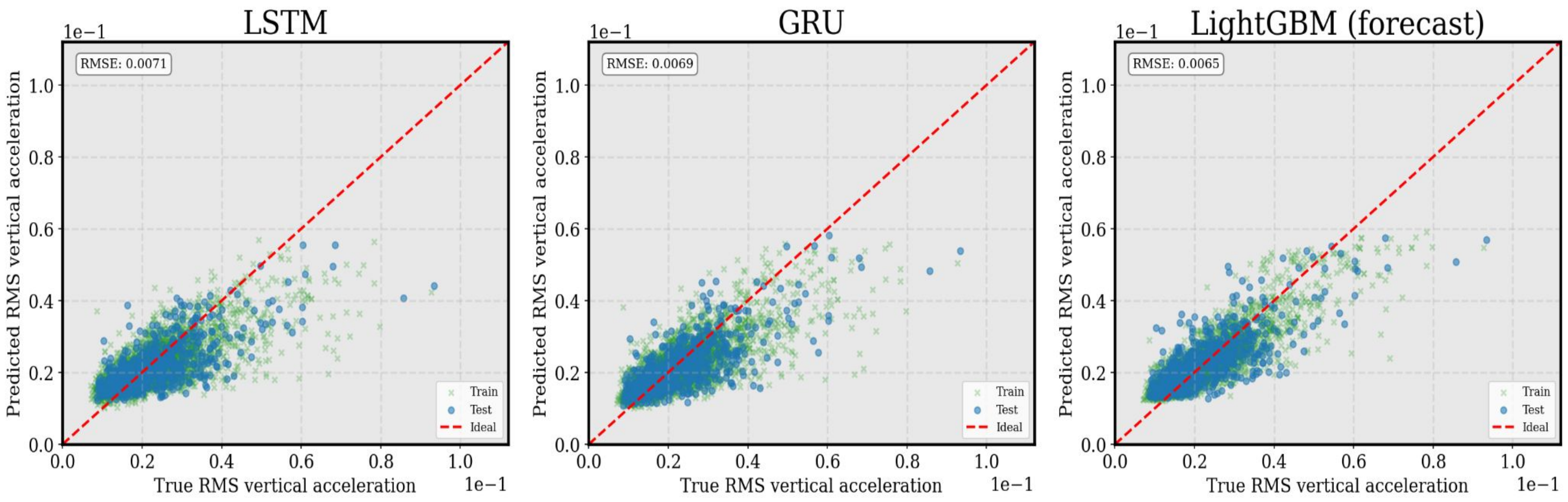


Forecast

- Flight and wind data from same time interval.
- Use wind data from 10 minutes before 400 feet altitude
- Predict RMS in 400-25 ft. altitude.



Results: Forecast



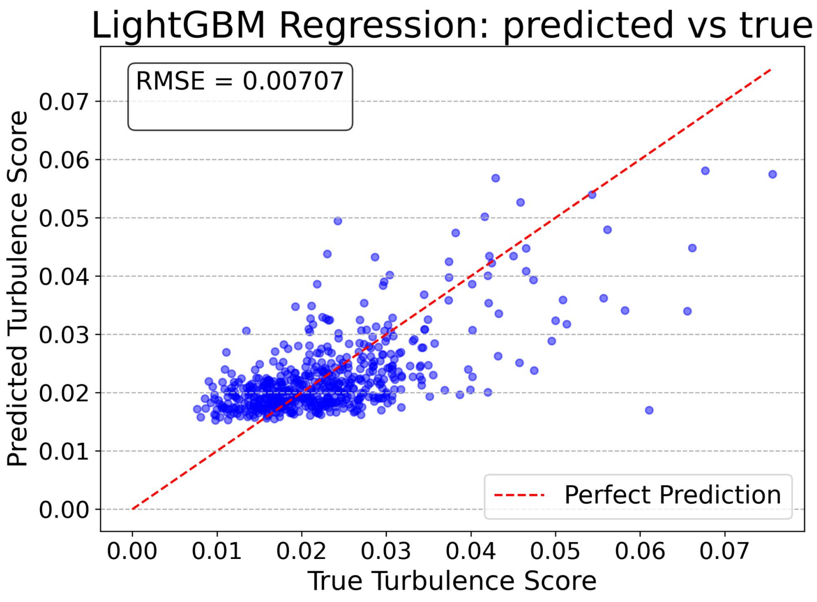
- LightGBM outperforms the models for sequential data
- Performance improves with declining complexity

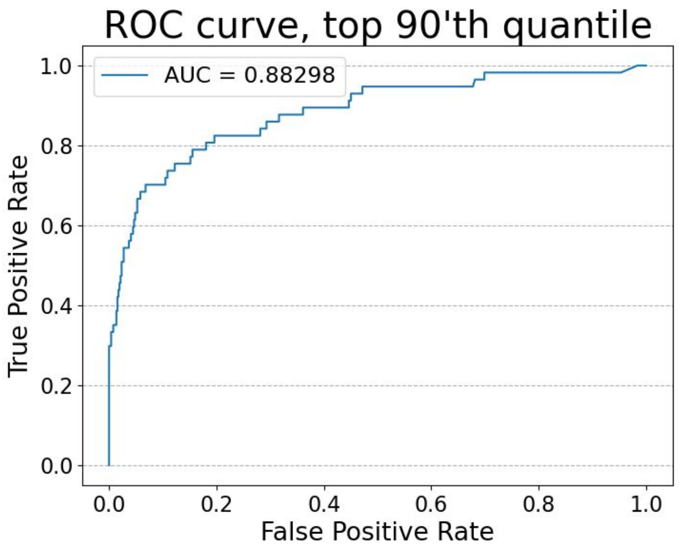
VAE forecast results

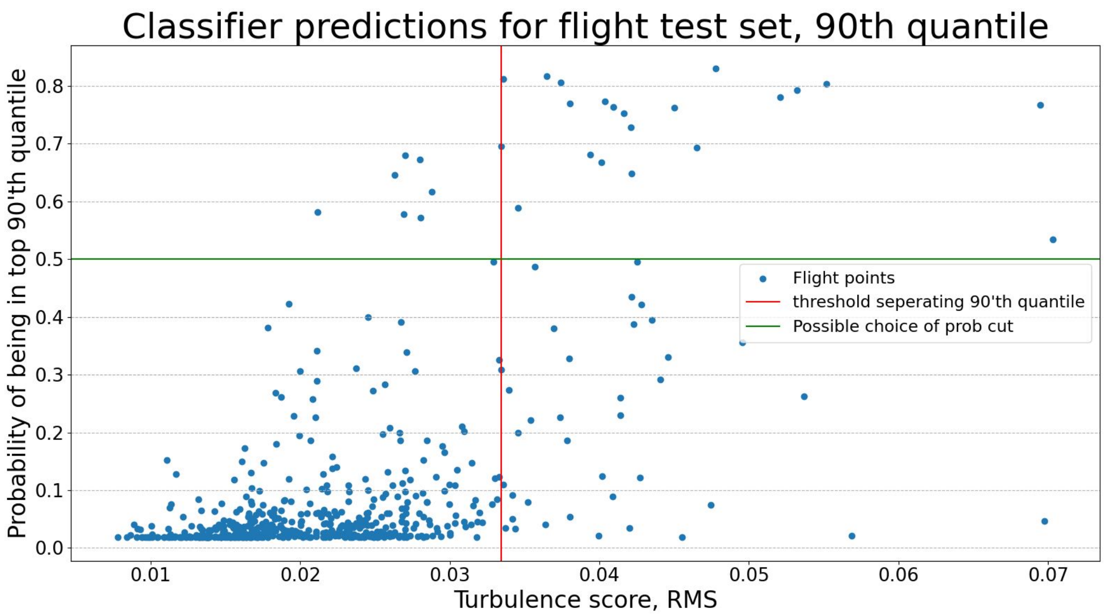
- Regressor at top

- Classifier trained on 90’th quantile of turbulent scores: key estimators:

  - For cut = 0.5, TPR = 0.75
  - We hit top 10 most turbulent flights with TPR = 0.6

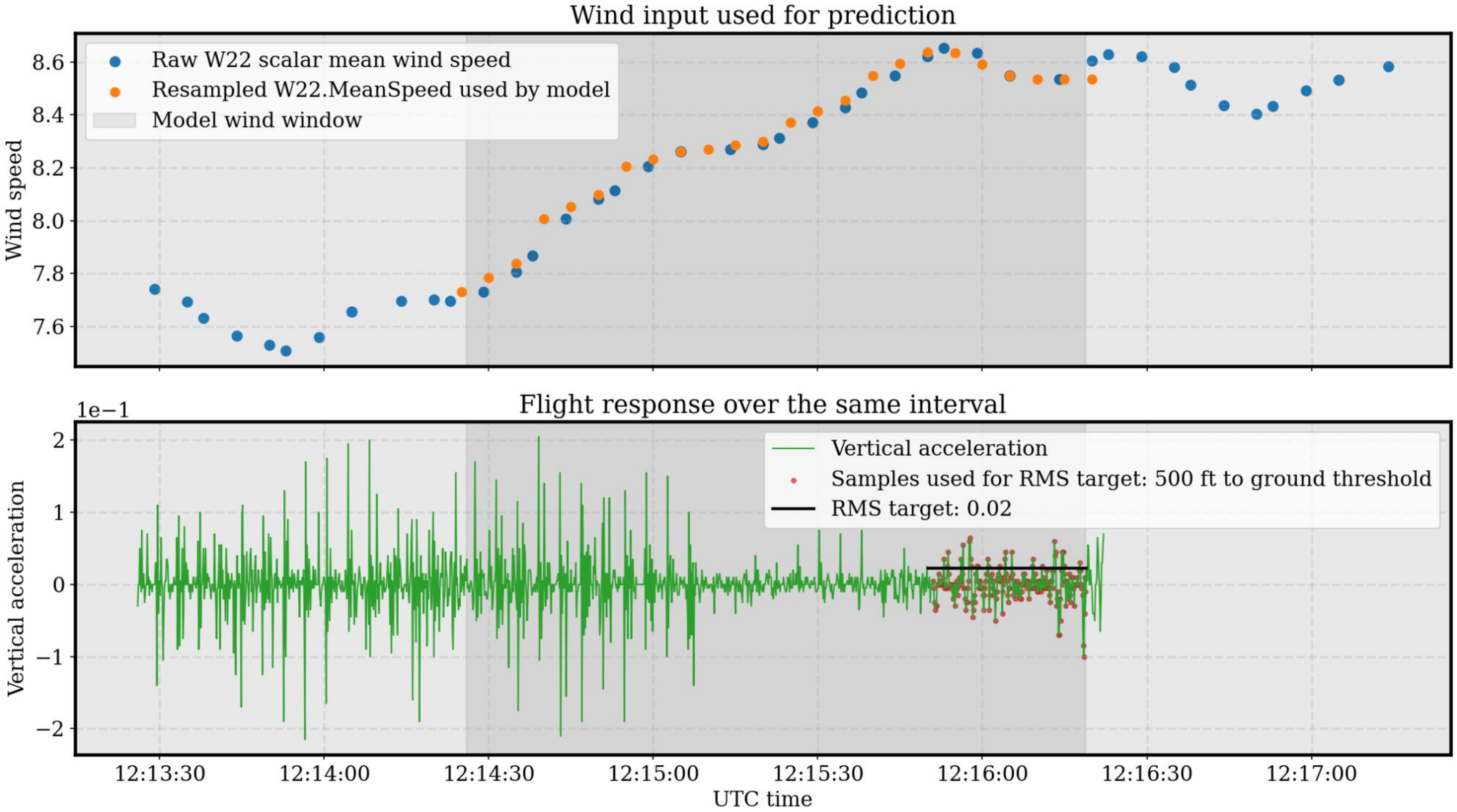




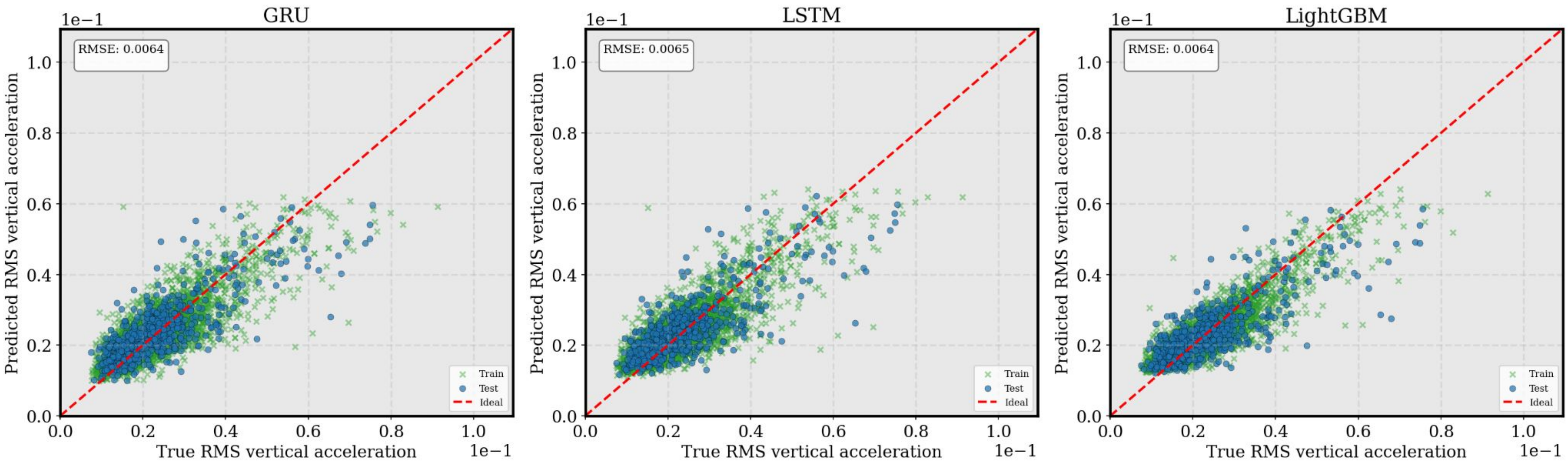


Nowcast

- Flight and wind data from same time interval.
- Use wind data from 1500 - 25 ft. altitude.
- Predict RMS in 400 - 25 ft. altitude.



Results: Nowcast



- Resulting RMSE is about the same for all models
- A simpler model matches performance of more complex models

Comparison forecast vs nowcast

Nowcast	LSTM	GRU	LightGBM
RMSE (g)	0.0065	0.0064	0.0064

VAE Forecast	BDT
RMSE	0.0071

Forecast	LSTM	GRU	LightGBM
RMSE (g)	0.0071	0.0069	0.0065

- Baseline: Predicting the mean RMS on all flights

Conclusion

- Nowcast performs better than forecast when comparing to the baseline guess of the mean RMS of vertical acceleration.

- Best model Forecast: LightGBM (measured on RMSE)

- Best model Nowcast: Models perform equally good

- Simpler models matches performance of more complex models

Why

- More data is probably needed for training the complex models (GRU, LSTM)
  - Low number of flights with high turbulence

- Low correlation between wind and metric used for turbulence

- Local turbulence captured in the time series might not correlate to an aircraft far away. Could be long distance, different wind layer, etc.

- Metric (RMS) might not capture turbulence correctly

Thank you!
Questions?

Appendix

Everyone contributed equally in this process

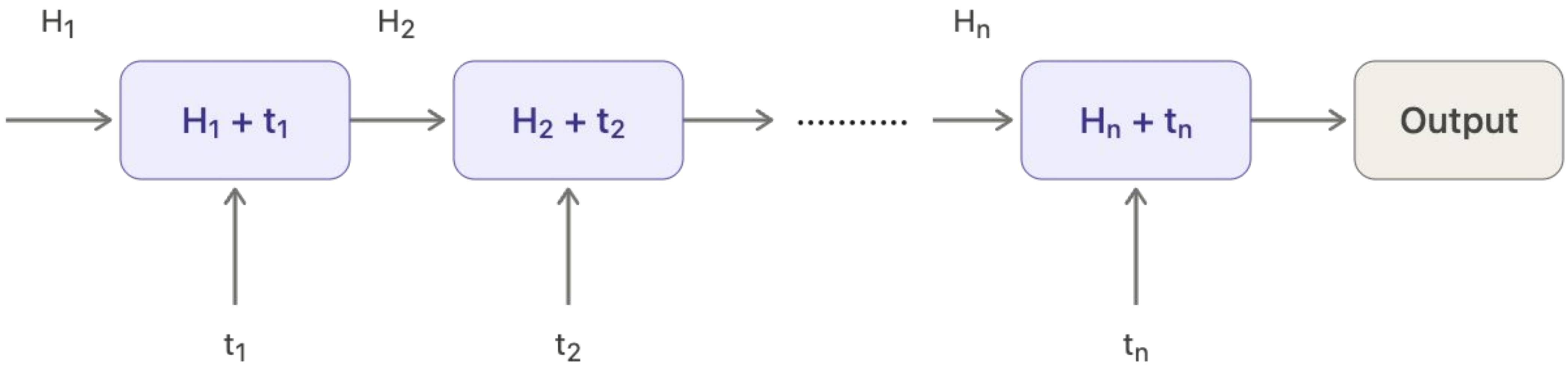
RNN's (Recurrent Neural Networks)

GRU: Neural networks for sequential data

-Uses a hidden state to carry information from previous time steps

-Combines current input with information from the hidden state

-Updates the hidden state and uses it for the next prediction



LSTM (Long short-term memory)

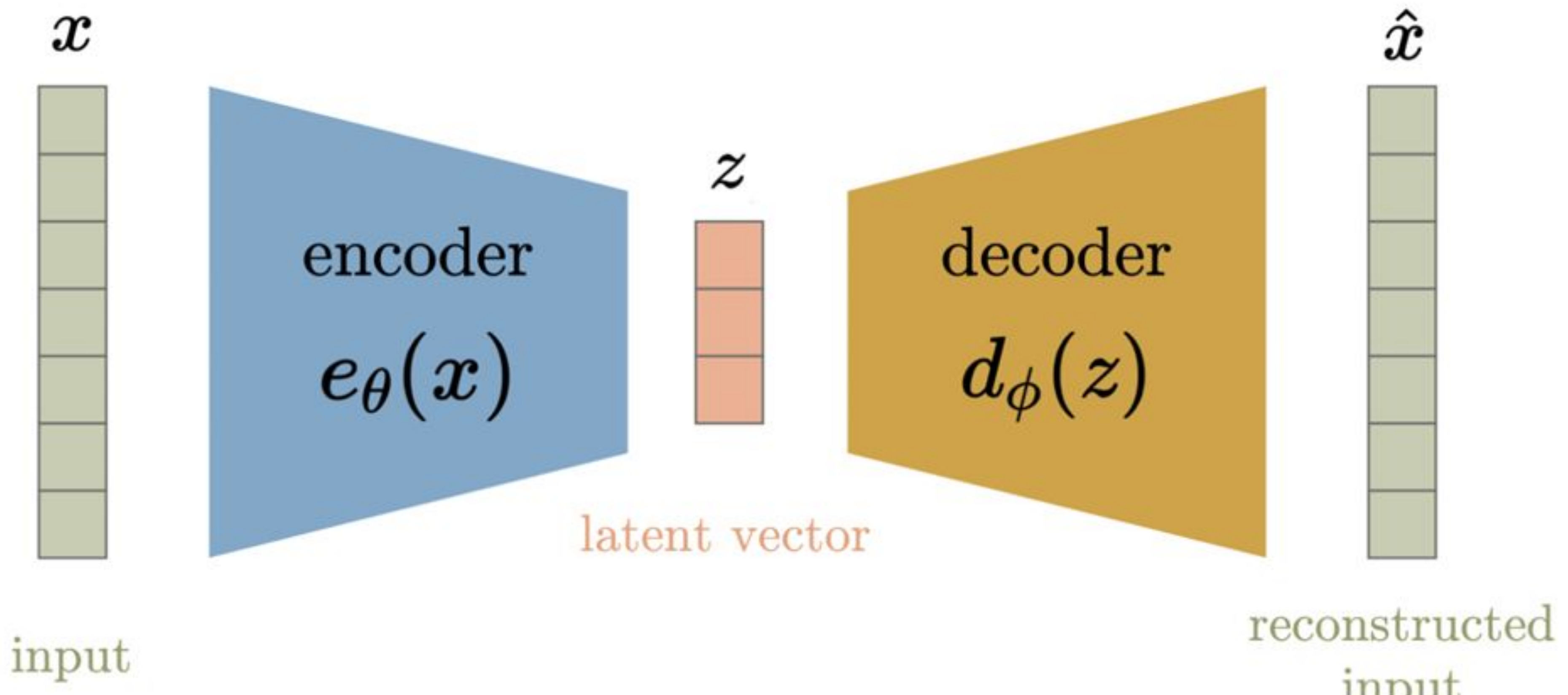
-Maintains both short-term and long-term information

-Current input is combined with information from previous observations

-States are updated and passed to the next time step for prediction

Variational Autoencoder - VAE

- An auto-encoder (AE) is a method (typically based on neural networks) to learn efficient data codings in an unsupervised manner (hence the “auto”).
- Data -> Latent space with smaller dimension



VAE classifier pipeline

- Create sequences function, feeding VAE with 256 batches of size (22, 120)
- VAE with normalizing checkpoints between convolutional layers
- VAE condenses each batch into 1 point in 64-dim latent space
- Classifier passed Latent space from Encoder, optimized with random search, paired with flight data as given in main presentation
  - Dataset was small ~ 2856 entries ~ which was subsequently split for further use
- Classifier trained with early stopping

GRU pipeline

- Create time sequences where wind correspond to flights, with padding so GRU can accept different sizes of data. (or static length for forecast)
- Scale training data set and targets (RobustScaler and log1p())
- Use GRU encoder on sequence and create a summary vector with learned representation of the wind.
- Use mean pool on the GRU output to capture wind history and combine with static features.
- Use in neural network for predicting the RMS of the detrended vertical acceleration.
- Minimize MSE loss on predictions for epoch training.
- Hyperparameter optimization with RandomSearchCV
- Choose optimal model based on final RMSE on validation test set

LightGBM pipeline

- Features is extracted from the wind data to get features instead of a time series
- Scale data and the targets (RobustScaler and log1p())
- Hyperparameter optimization - Random search (Model training)
- Choose best model based on RMSE
- Compare on test data to other models

LSTM pipeline

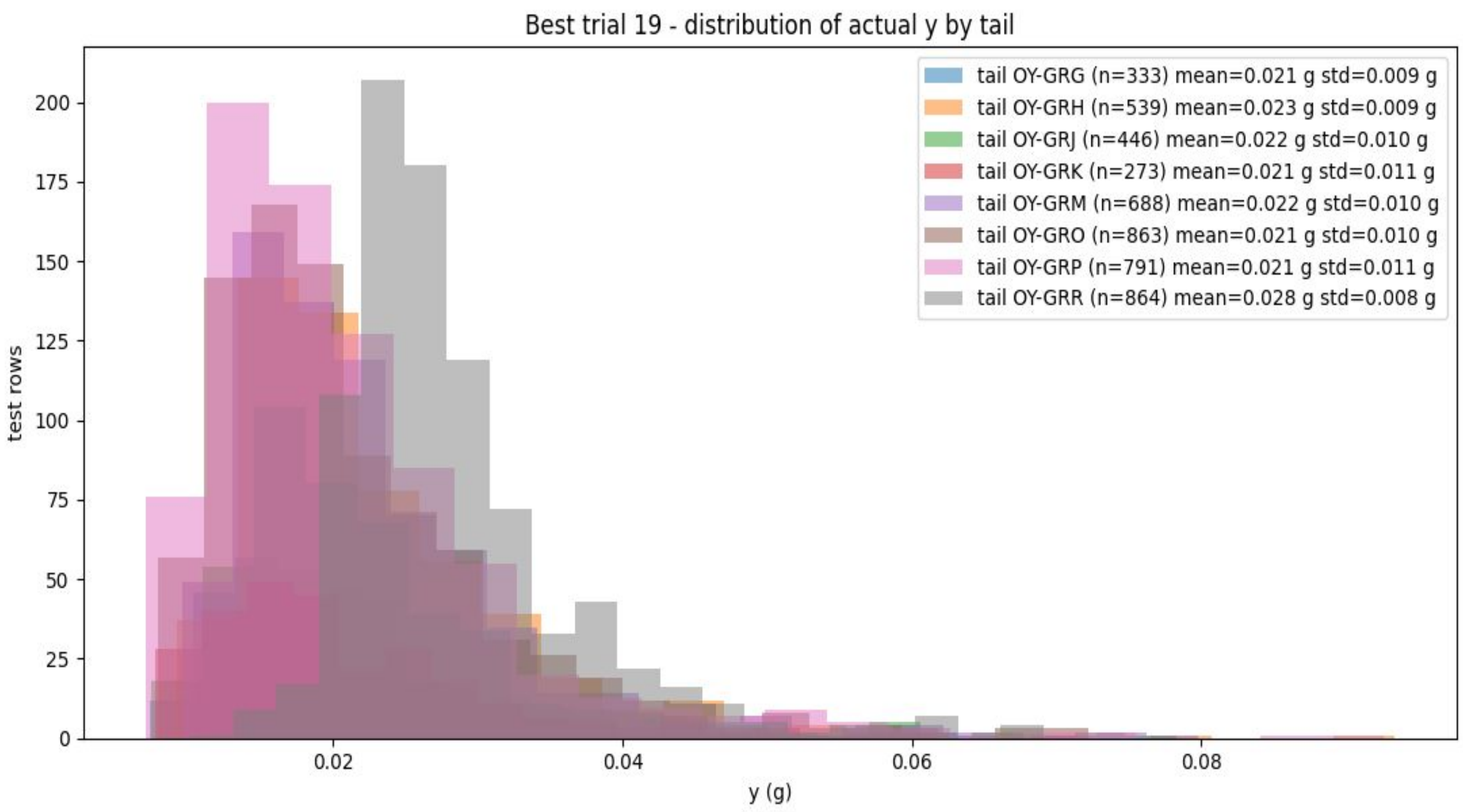
- Create time sequences where wind correspond to flights, with padding so LSTM can accept different sizes of data. (or static length for forecast)
- Scale training data set and targets (RobustScaler and log1p())
- Use LSTM encoder on sequence and create a summary vector with learned representation of the wind.
- Use mean pool on the LSTM output to capture wind history and combine with static features.
- Use in neural network for predicting the RMS of the detrended vertical acceleration.
- Minimize RMES loos on predictions for epoch training.
- Hyperparameter optimization with using random search
- Choose optimal model based on final RMSE on validation test set

Distribution of y-values

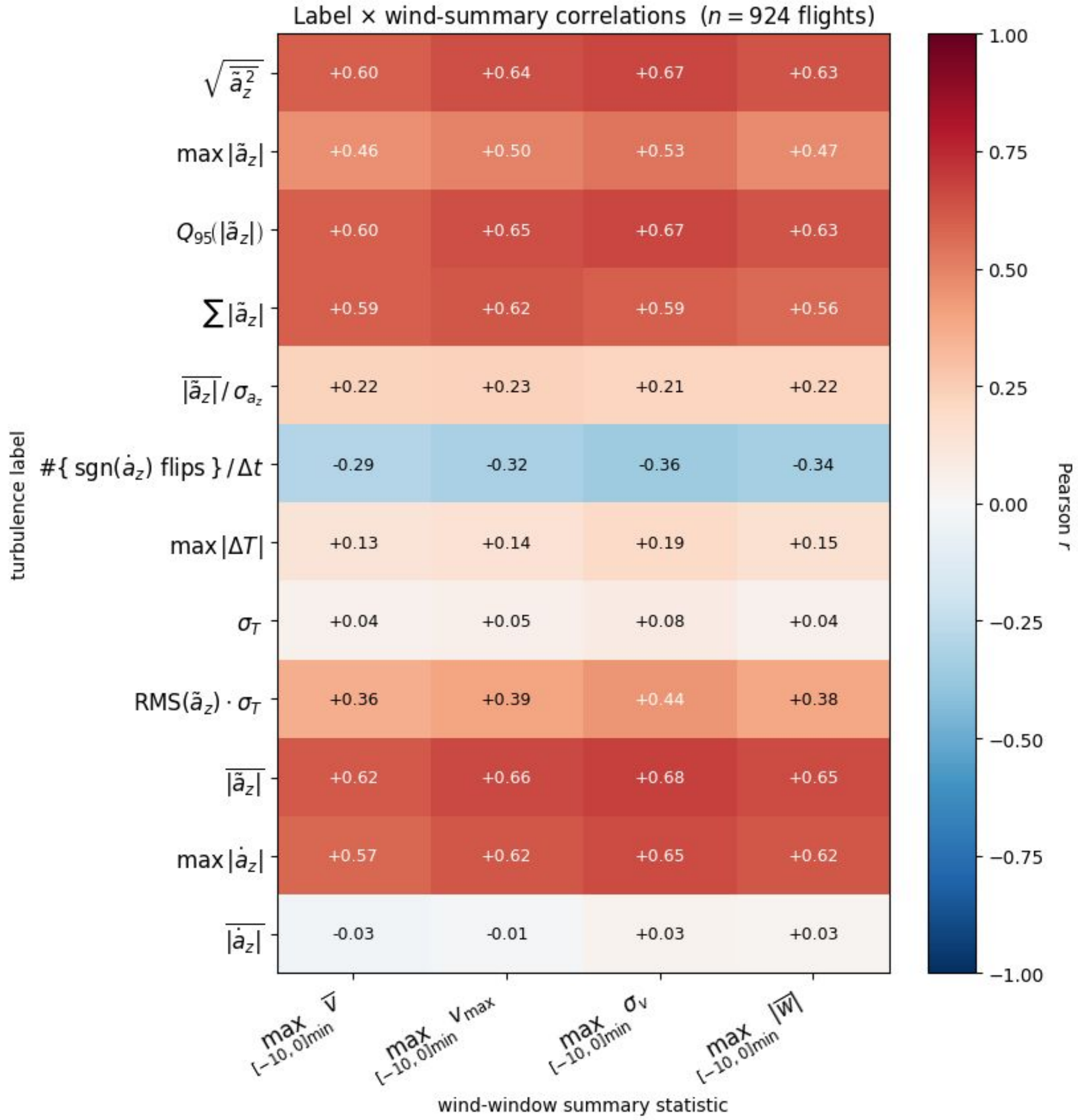
The distribution of the values show a heavy tail toward large values.

Predicting the low tail is hard

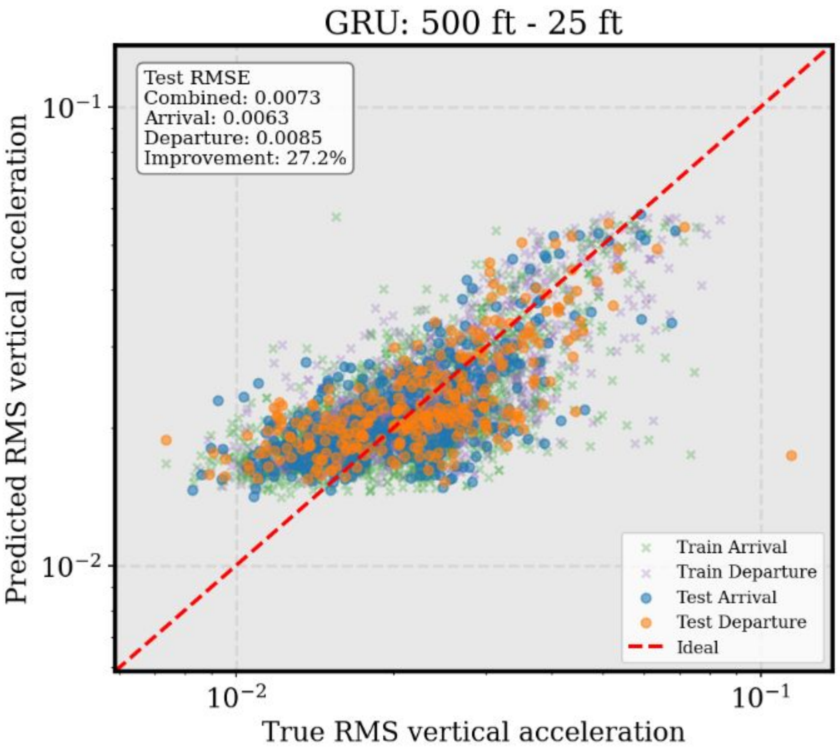
One plane stands out with a different distribution of target values.

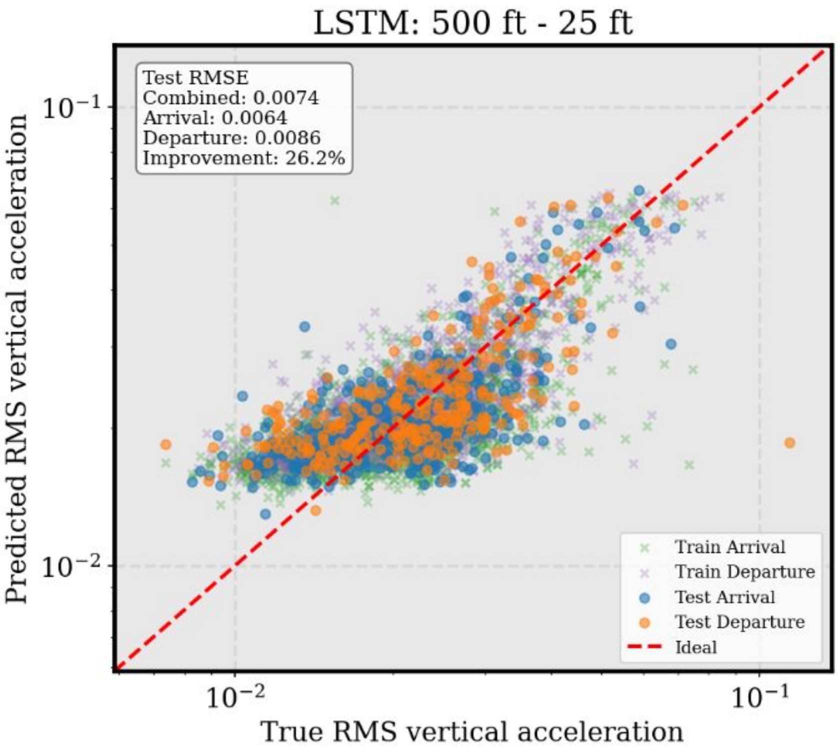


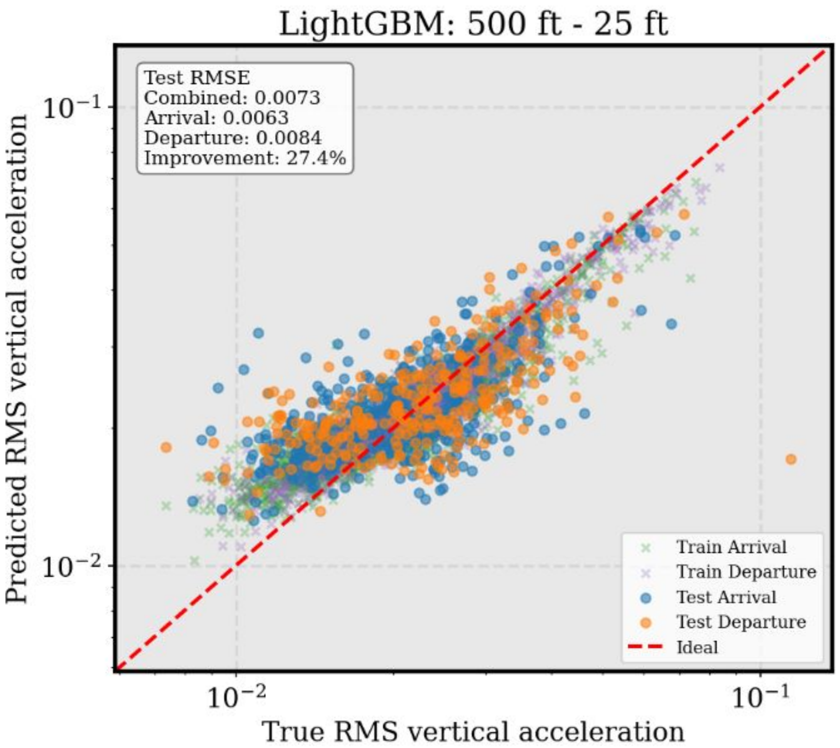
Correlations for Different Metrics



Departure vs arrival with different models

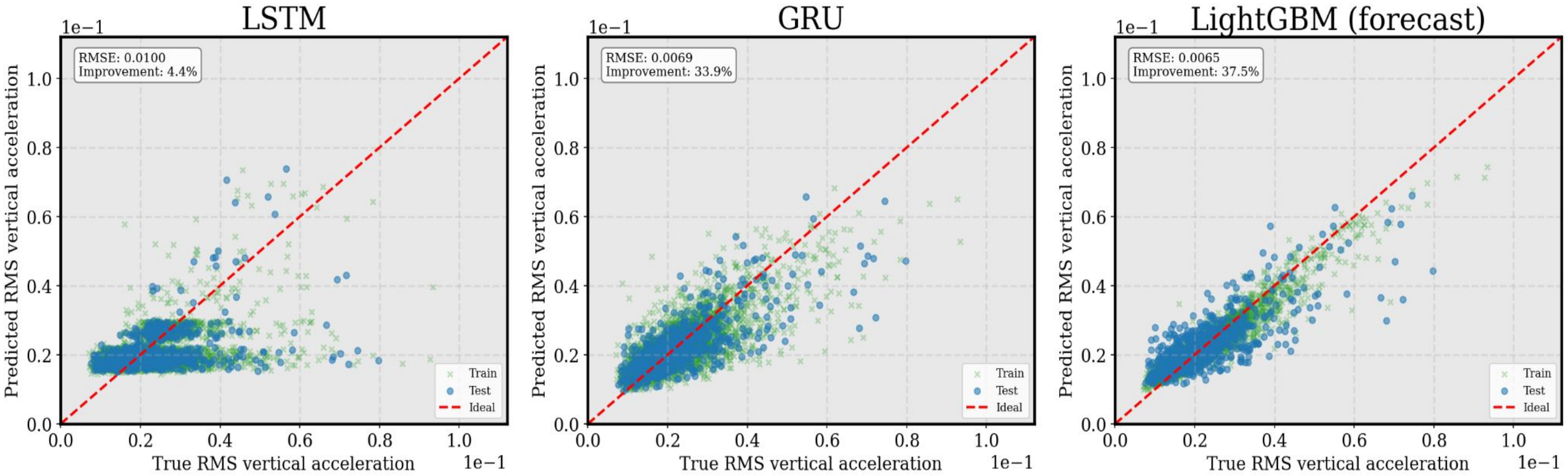






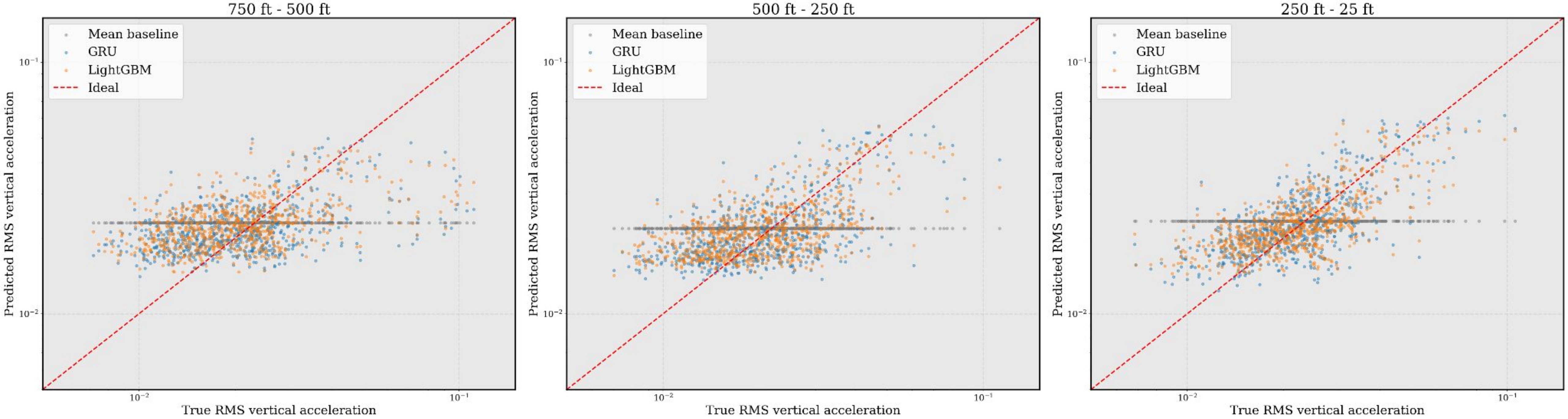
Arrival and departure looks the same: overshooting at low and scattered at high vertical accelerations

Forecast results with flight numbers and 22 minutes back in time



Note: A larger time window as input does not improve the model and is only able to predict the two distinct mean of the data

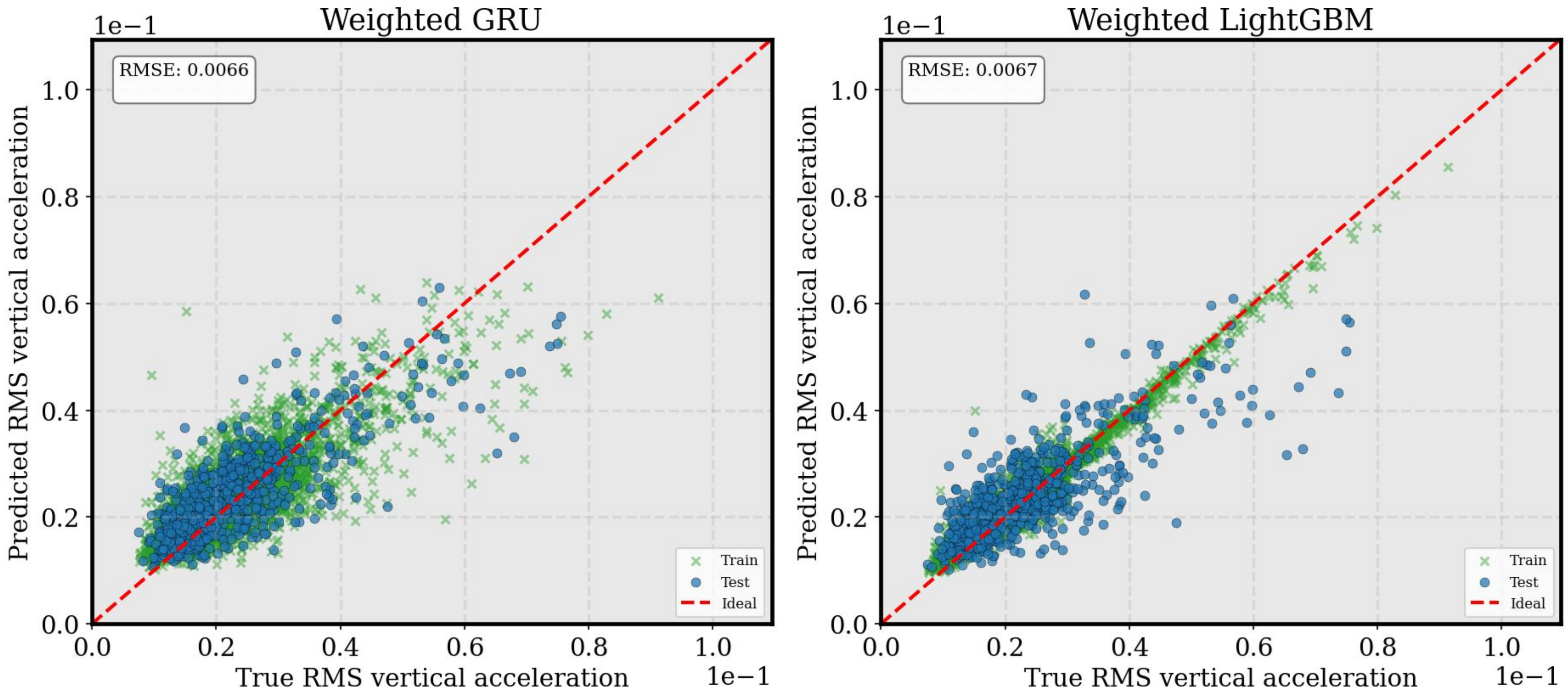
Predicting different intervals



The predictions are best close to the wind measurements.

This plot shows the Nowcast in different intervals.

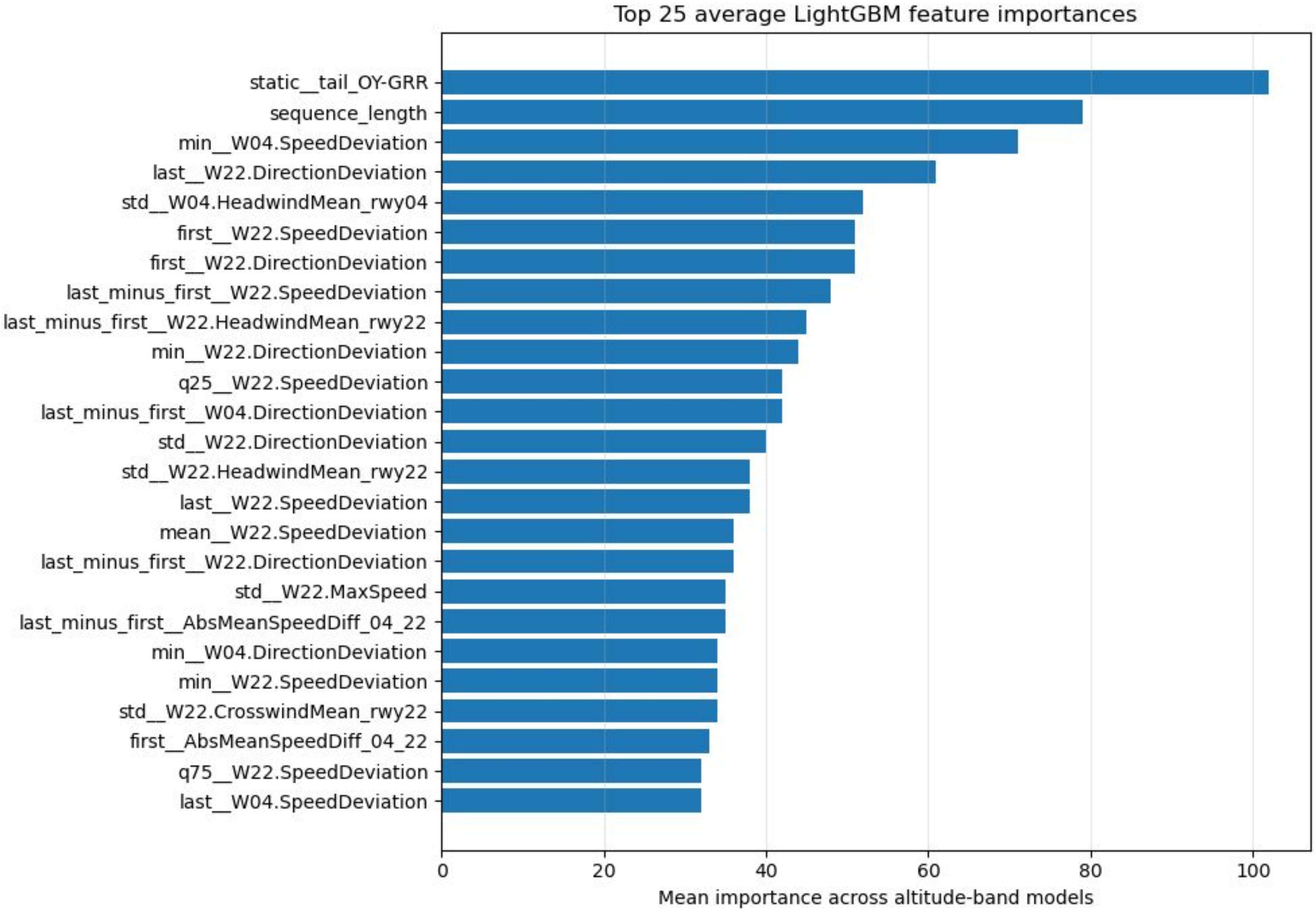
Adding weights



We tried adding weights to the GRU to account for the tails. This is done by sampling the tail 3 times as frequently.

It is seen that the training does not generalize to the test. The weighting does not work for the GRU and the LGBM shows signs of overtraining.

It is seen that there is one specific plane-type that is very important



VAE - statistics for classifier

80’th quantile:

- AUC of model is 0.84
- For cut = 0.5, TPR = 0.66
- We hit top 10 most turbulent flights with TPR = 0.8

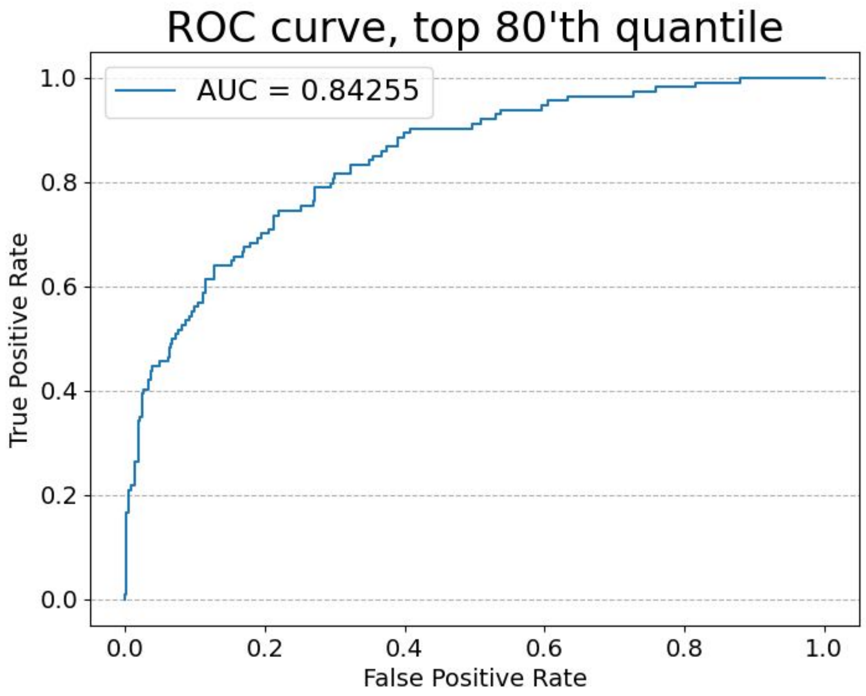
85’th quantile:

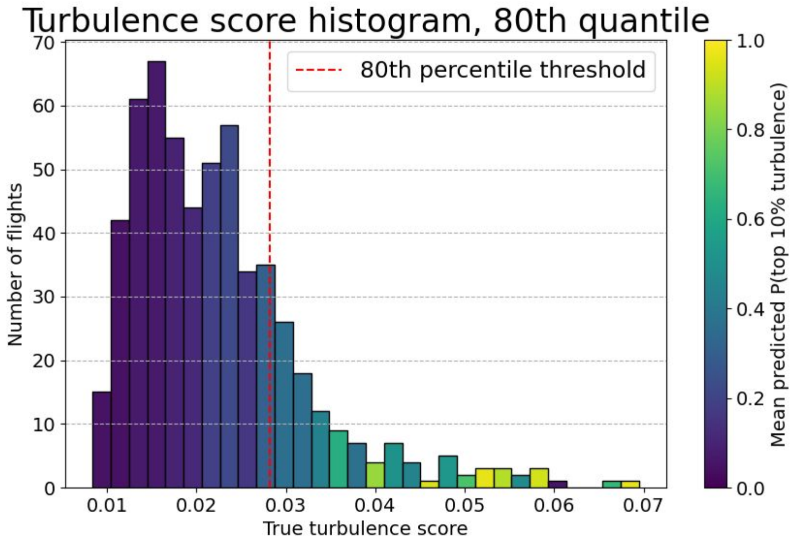
- AUC of model is 0.81
- For cut = 0.5, TPR = 0.75
- We hit top 10 most turbulent flights with TPR = 0.6

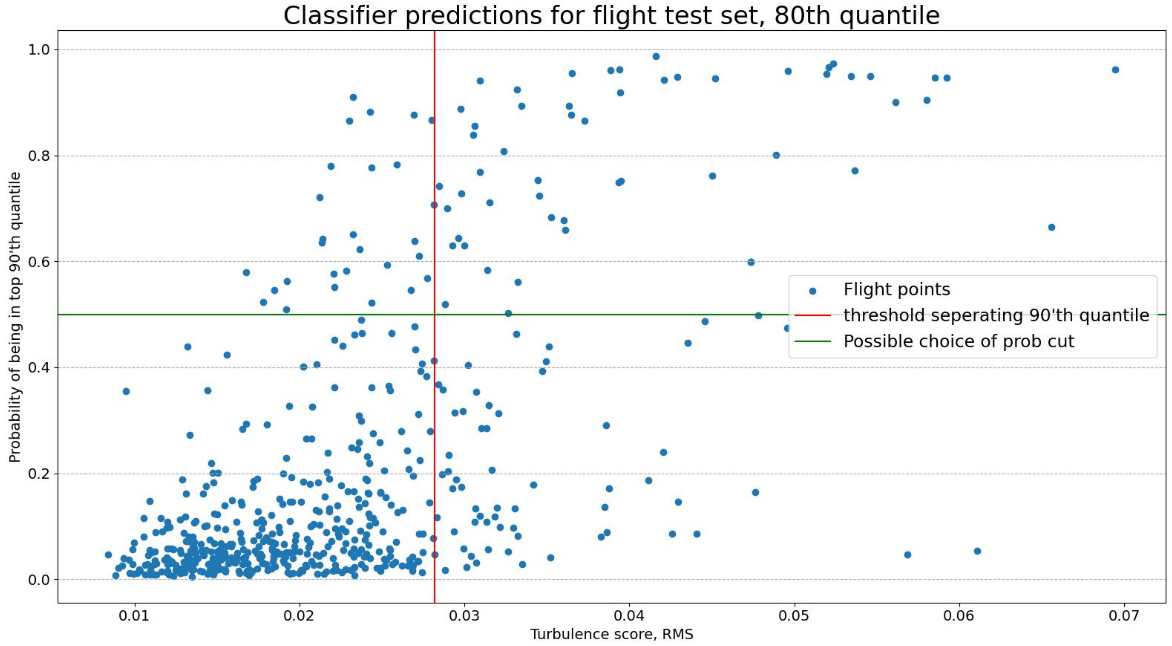
90’th quantile:

- AUC of model is 0.88
- For cut = 0.5, TPR = 0.75
- We hit top 10 most turbulent flights with TPR = 0.6

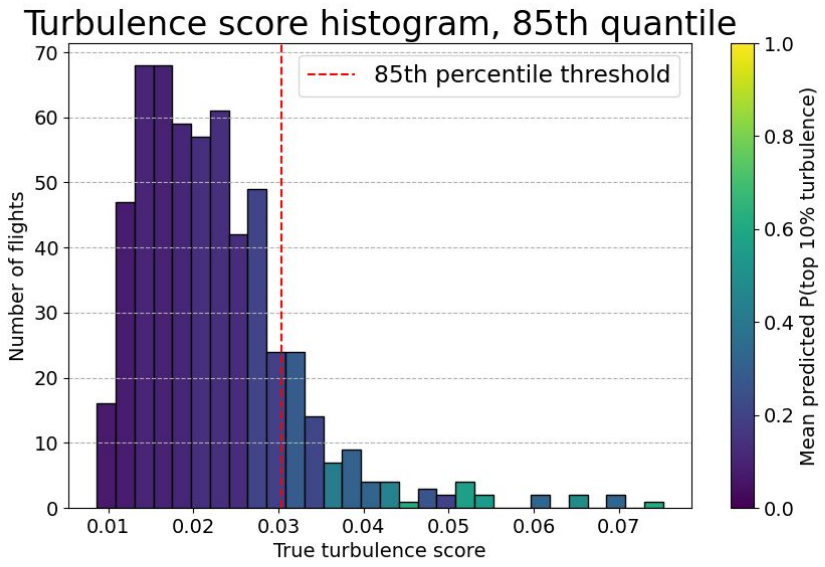
VAE - 80th quantile

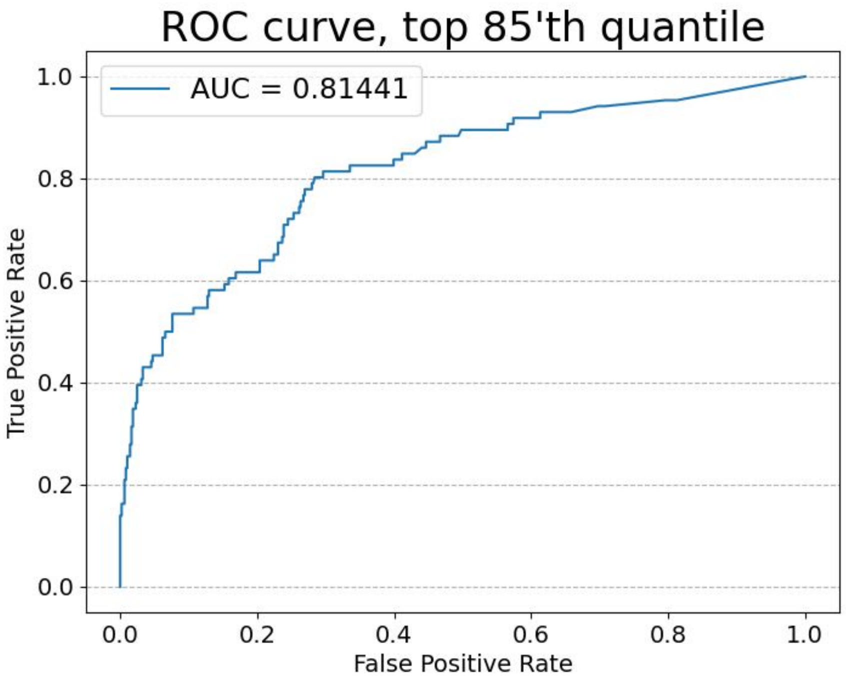


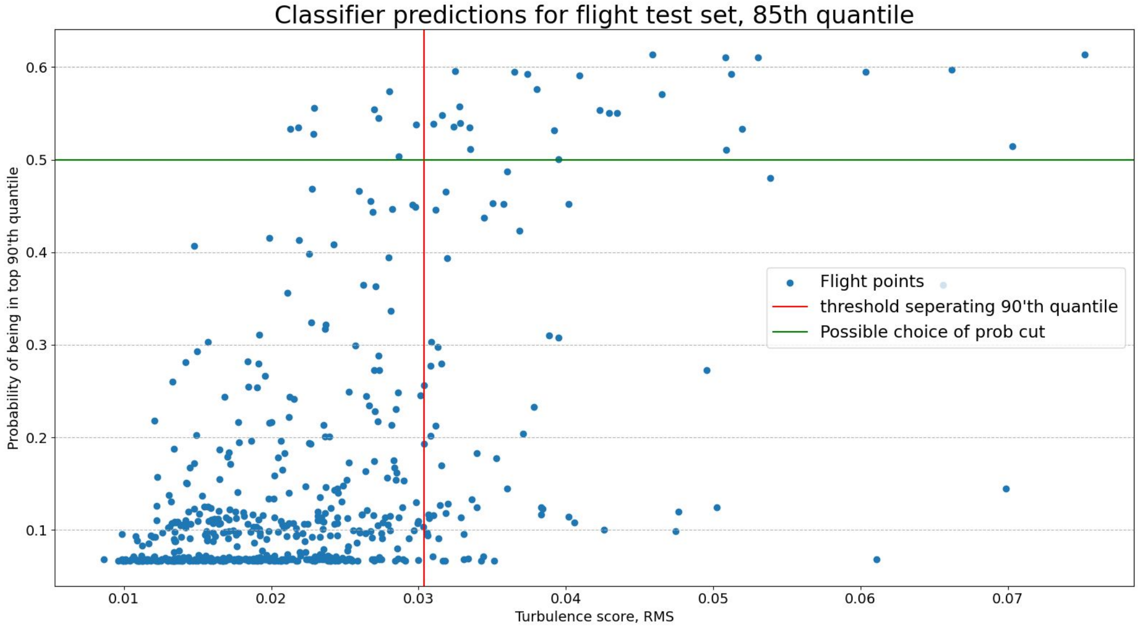




VAE 85th quantile







VAE - 90’th quantile