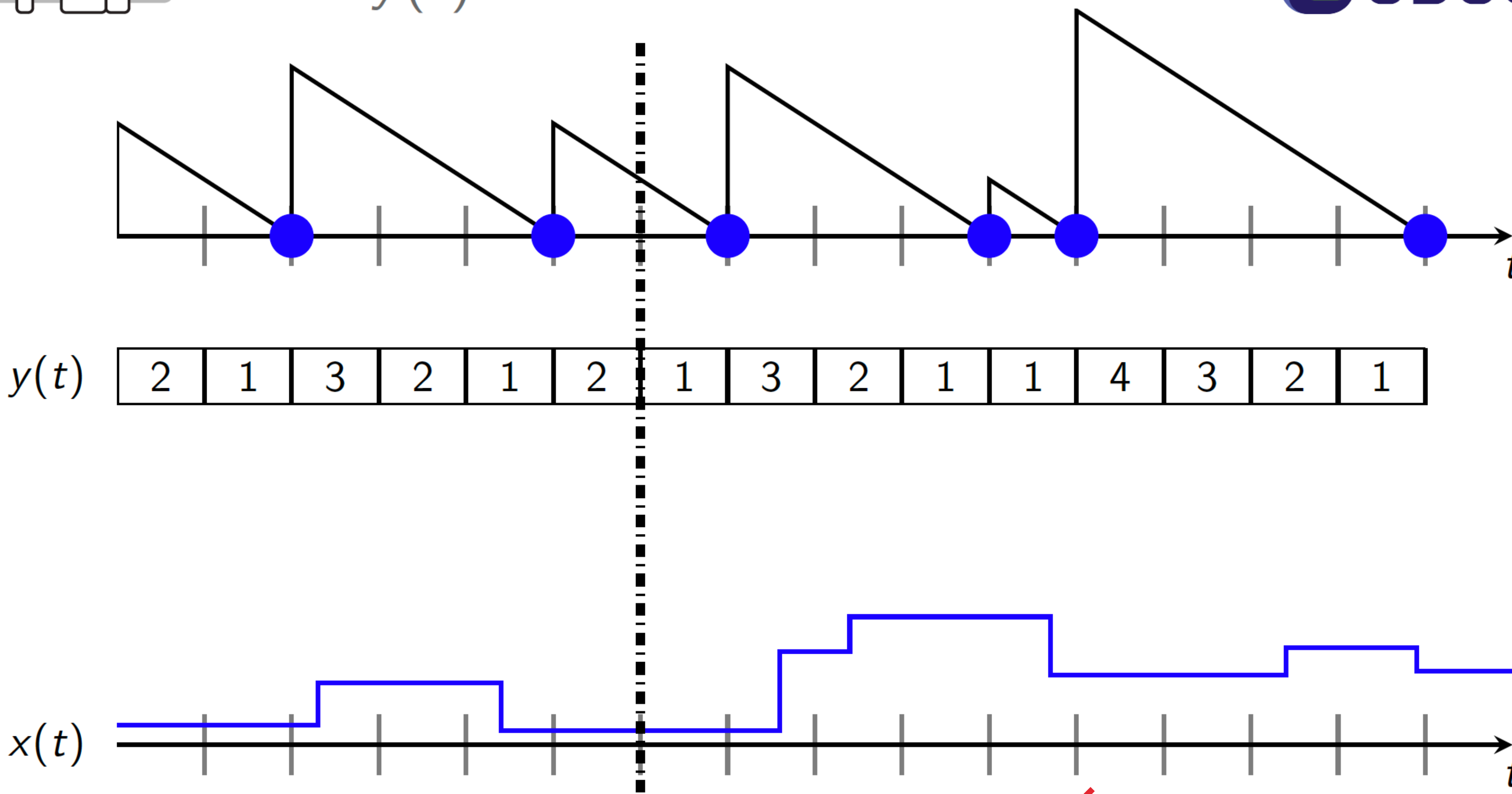
2020.2.24 SICHEN LI

PREDICTING INTERLOCK BY SURVIVAL MODELLING — STATUS UPDATE

GOAL

Define $y(t)$ to be "Time To Interlock"



An example channel (total ~~450~~)

DROPNA: 363

METHOD

Predict a distribution over the future: Weibull distribution

$$f(y; \alpha, \beta) = \frac{\beta}{\alpha} \left(\frac{y}{\alpha}\right)^{\beta - 1} e^{-(y/\alpha)^{\beta}}, y \geq 0 \tag{1}$$

- $\alpha > 0$ is the scale parameter: larger $\rightarrow$ distribution more spread out
- $\beta > 0$ is the shape parameter: affect shape, not simple shifting or stretching

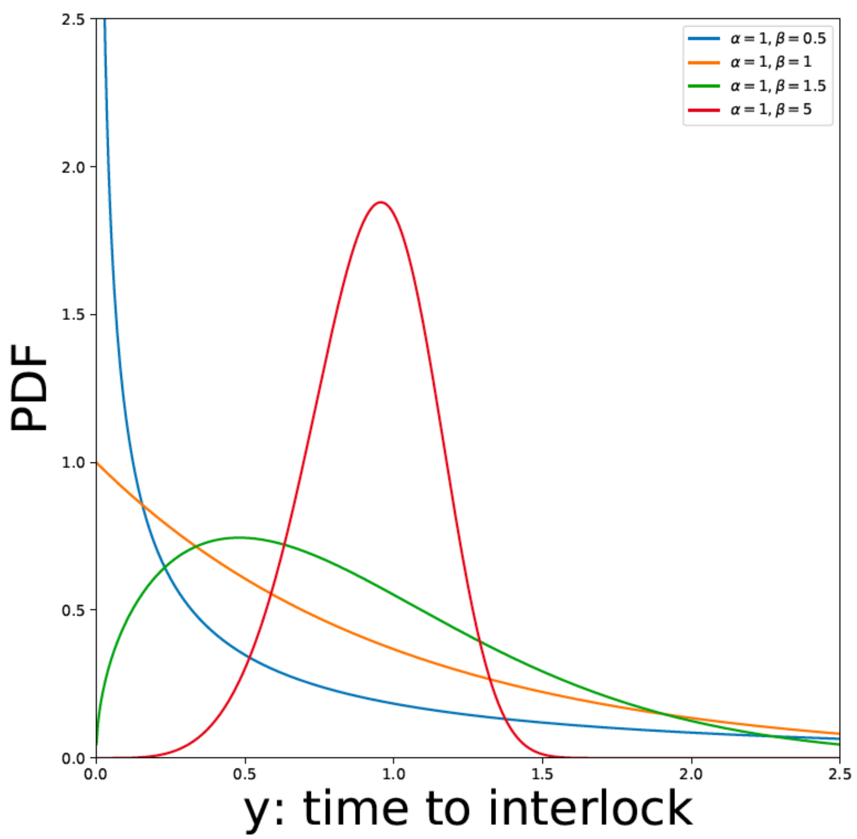


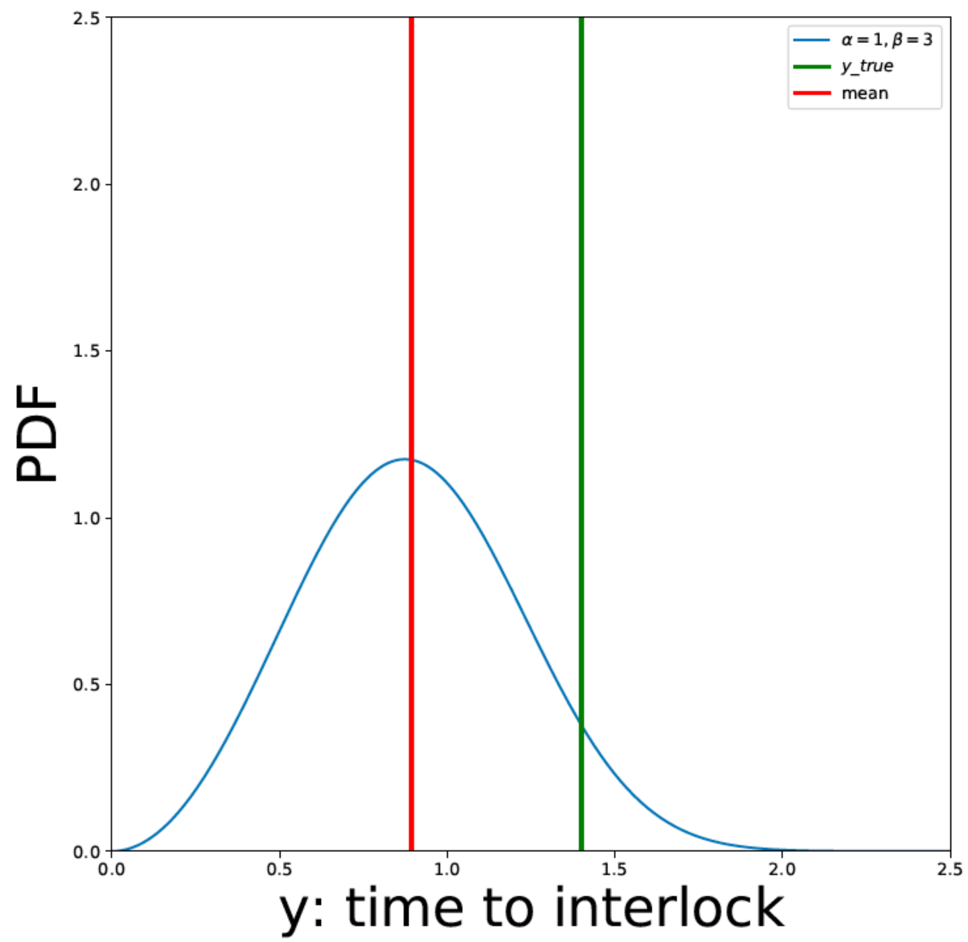
Figure: Weibull distribution PDF

LOSS

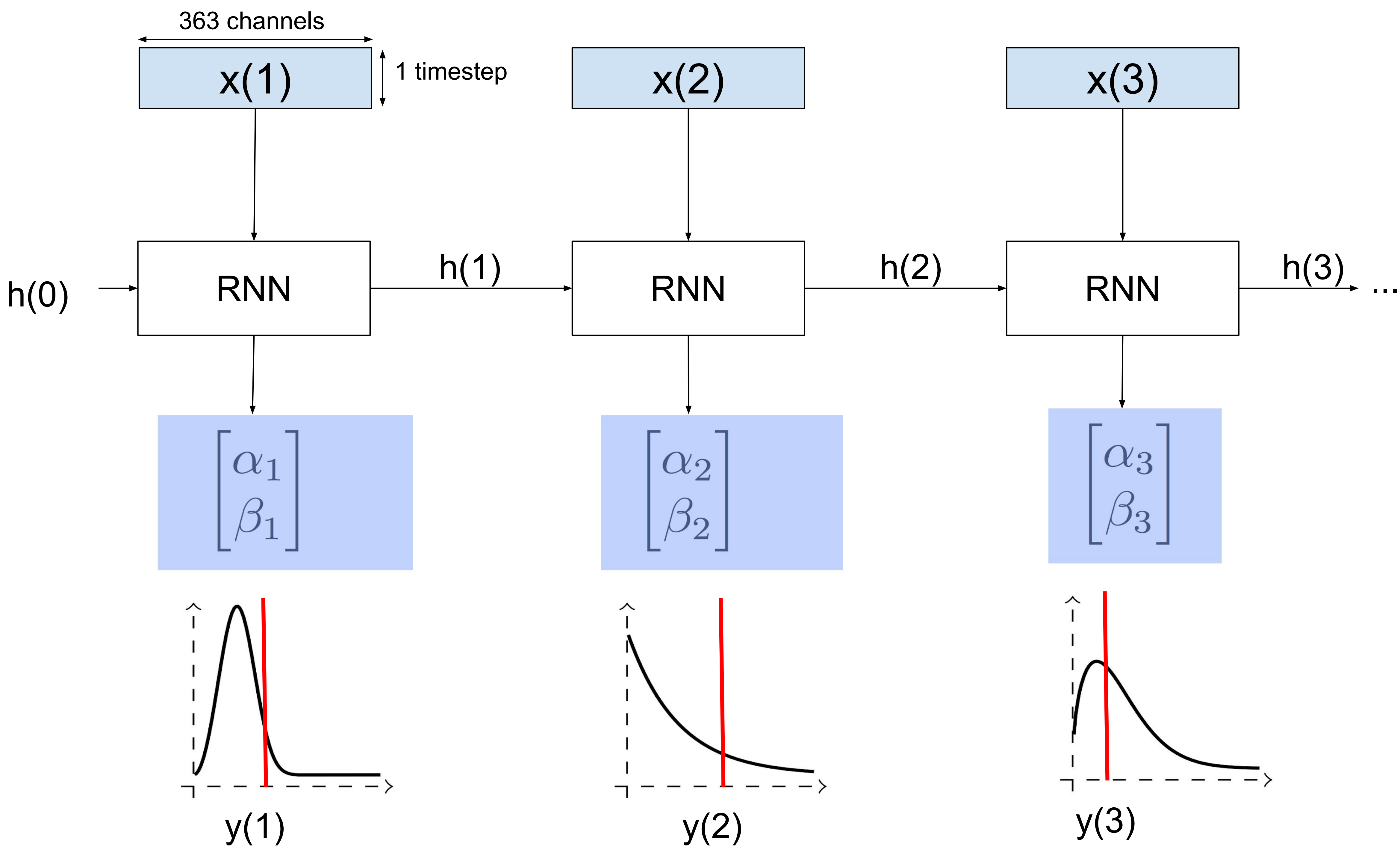
$$\max_{\alpha,\beta} f(y|\alpha,\beta)$$

- Loss = -log likelihood

Maximise the Weibull probability of y_true(the true time to interlock)



MODEL SKETCH

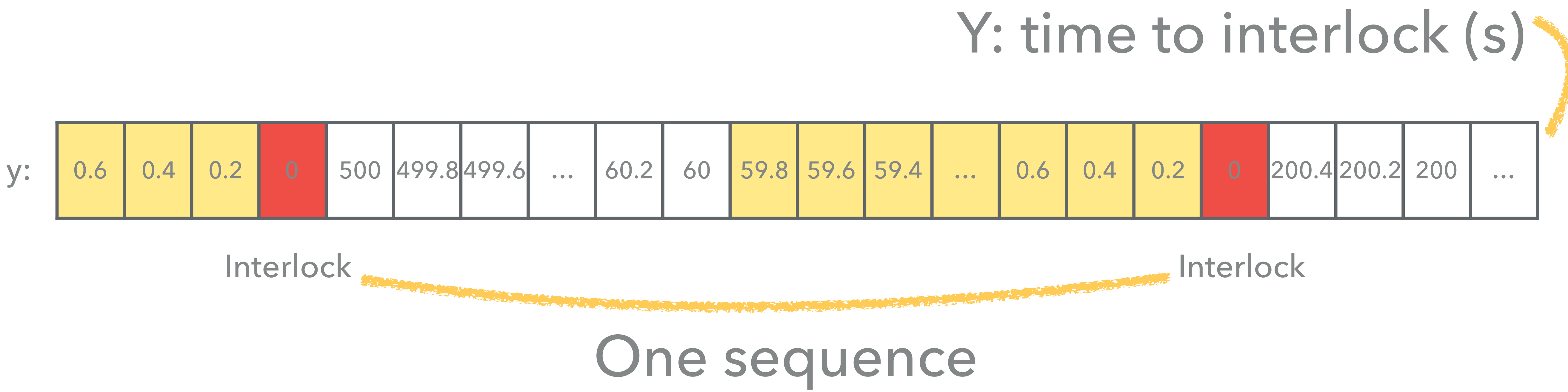


DATA STRUCTURE

Timesteps	0	1	2	3	4	5	6	...
Sample 0	x(0)	y(1)						
Sample 1		x(1)	y(2)					
Sample 2			x(2)	y(3)				
Sample 3				x(3)	y(4)			
...								

One X sample: (1 timestep, 363 channels) -> One y value: time to interlock

DATA STRUCTURE



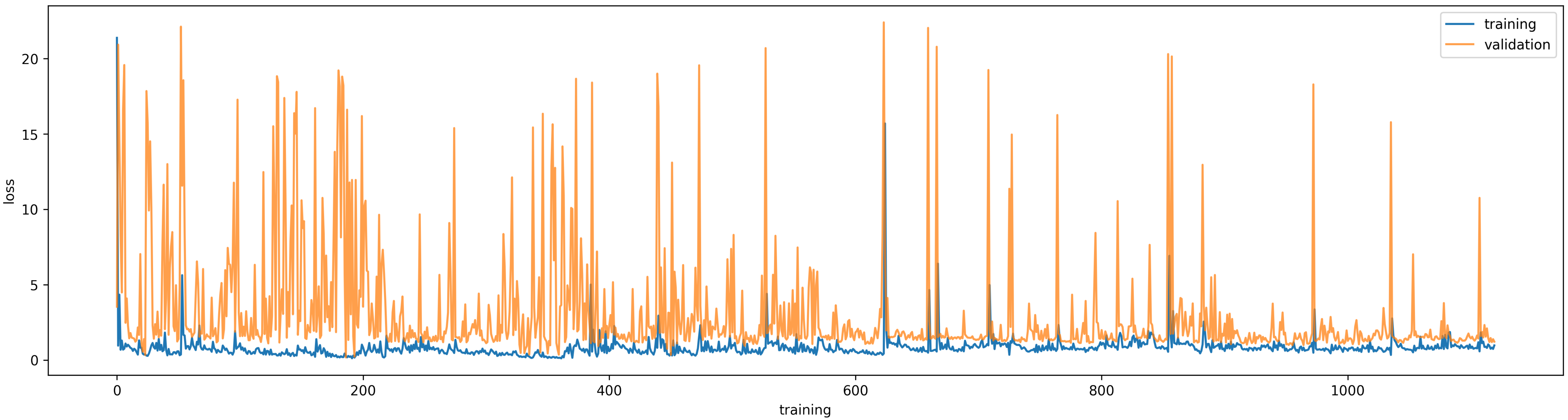
- Sequence: time series from last to this interlock
- Remove sequences shorter than 5 min in total (unstable)
- Only use the last 1min before each interlock

DATA & MODEL STRUCTURE

Data range	2019.9.18 - 9.27
Training set	74949 (249 sequences)
Validation set	16200 (54 sequences)
Test set	16200 (54 sequences)

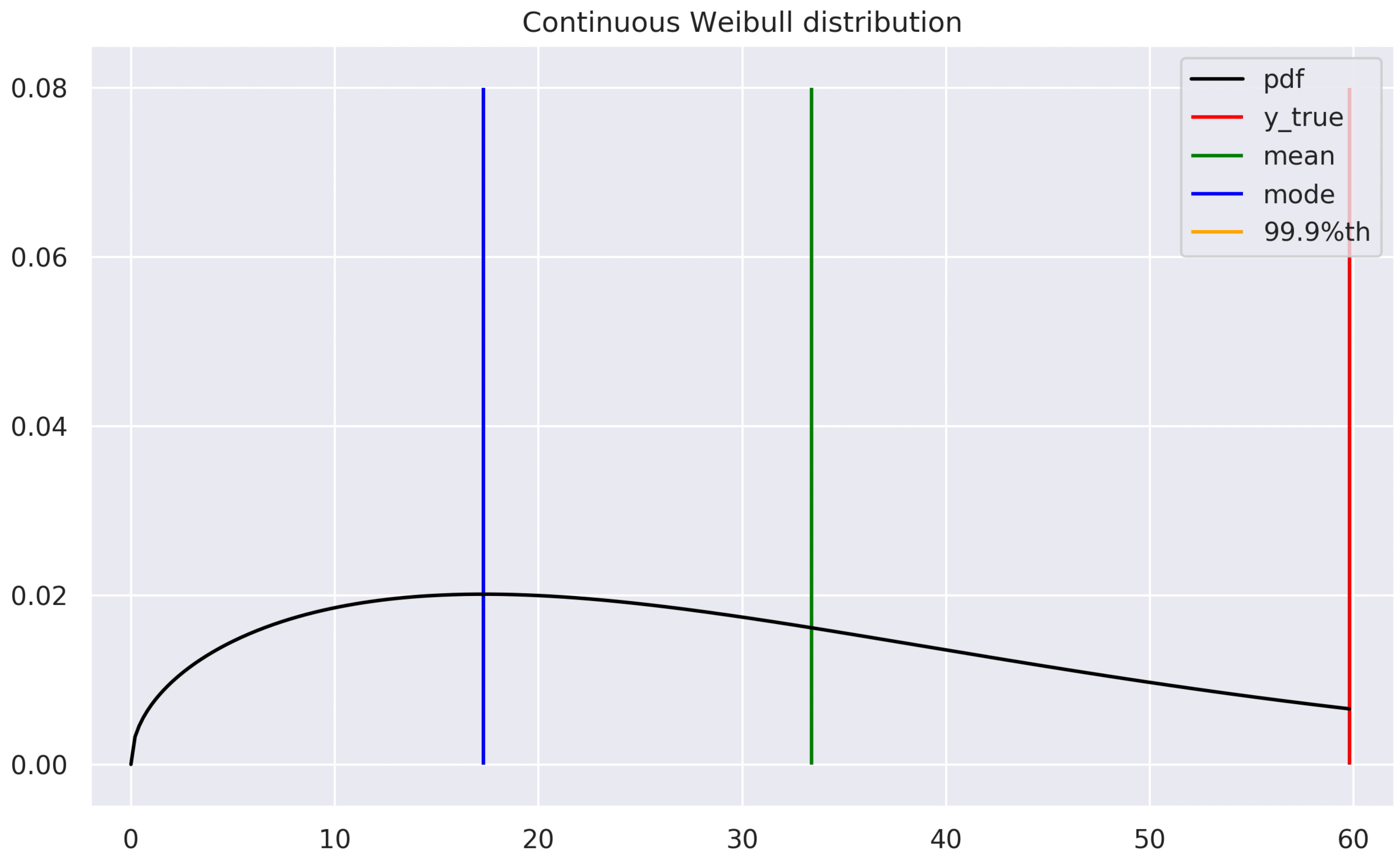
```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_1 (LSTM)                (1, 1, 32)                50688
_________________________________________________________________
dropout_1 (Dropout)          (1, 1, 32)                0
_________________________________________________________________
lstm_2 (LSTM)                (1, 32)                   8320
_________________________________________________________________
dropout_2 (Dropout)          (1, 32)                   0
_________________________________________________________________
dense_1 (Dense)              (1, 2)                    66
_________________________________________________________________
lambda_1 (Lambda)            (1, 2)                    0
=================================================================
Total params: 59,074
Trainable params: 59,074
Non-trainable params: 0
_________________________________________________________________
```

LOSS DURING TRAINING



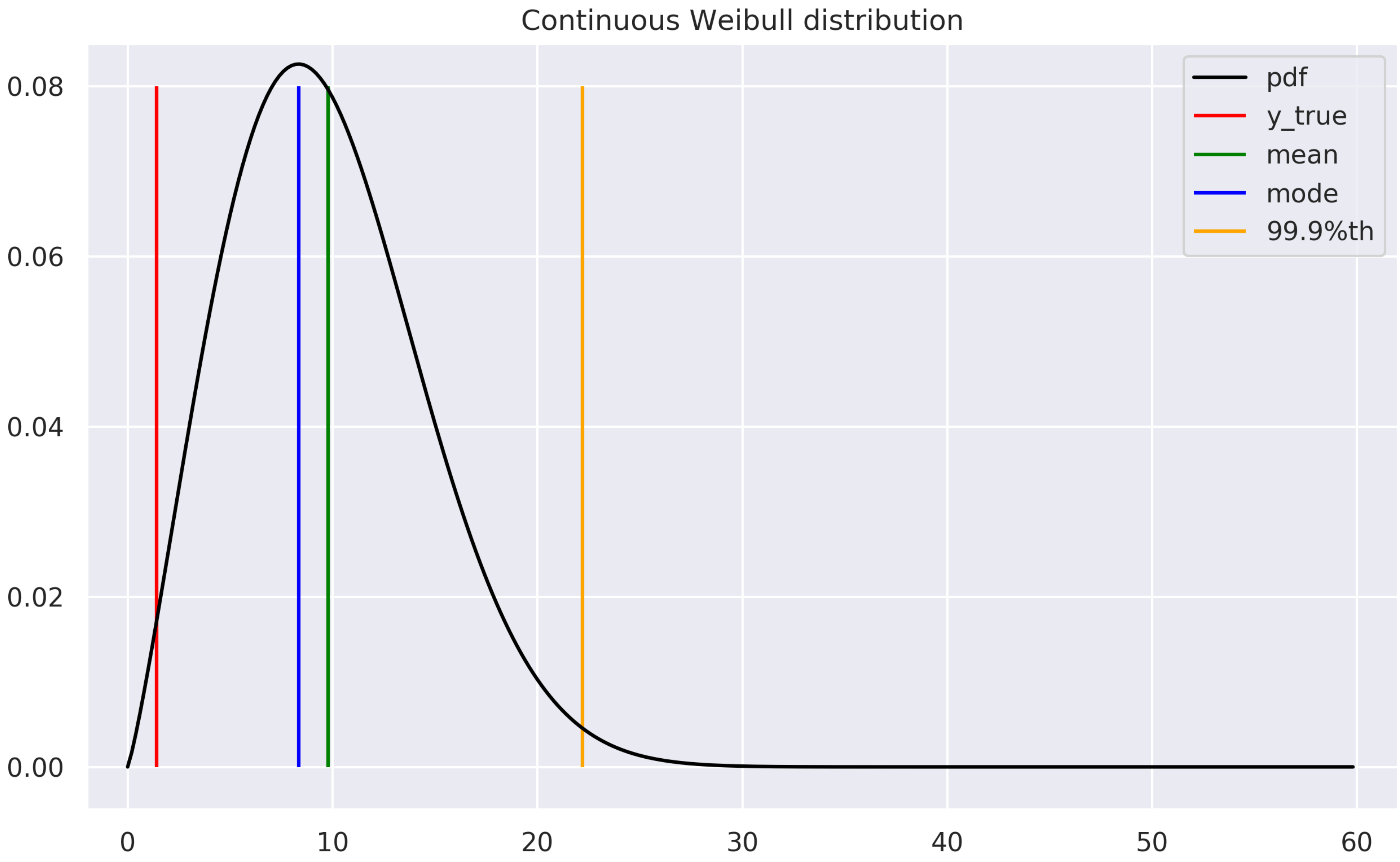
RESULT OF A TEST SEQUENCE

Video!



RESULT OF A TEST SEQUENCE

y_true: 1.4s to interlock



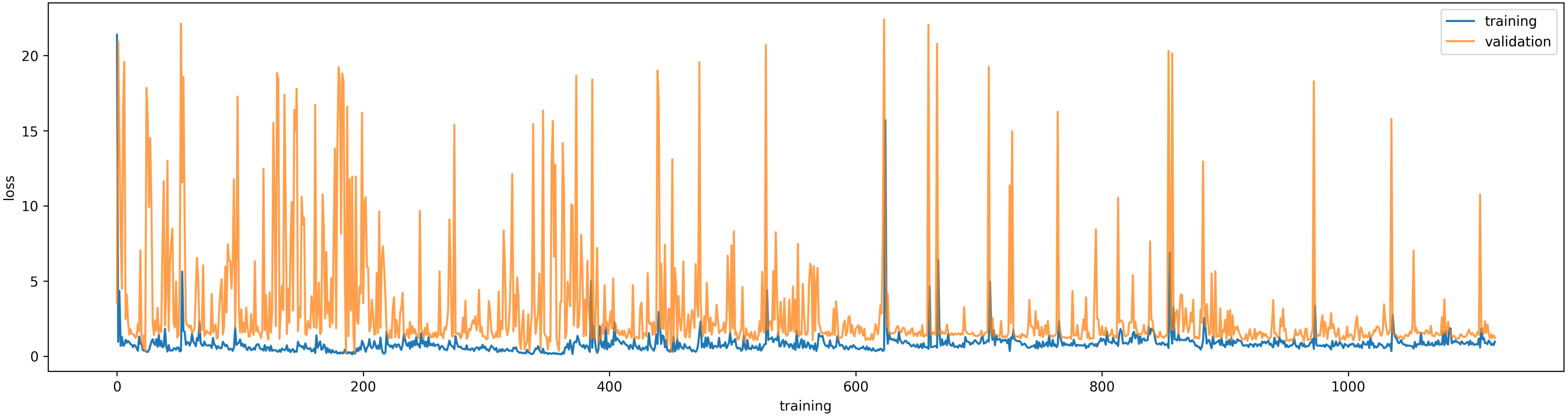
NEXT & QUESTIONS

- Most of the time, the output alpha and beta are at the same value
  - Significant only 5 seconds before interlock?
  - Use a better loss
- “Adaptive slicing”

REFERENCES

- WTTE-RNN package: https://github.com/ragulpr/wtte-rnn

LOSS DURING TRAINING



- (Train-sequences, val-sequences, test-sequences): (70%, 15%, 15%) in all sequences
- Train with 1 train-sequence for several epochs until no improvement can be gained from one random val-sequence -> go on with the next train-sequence