Modeling and inversion using neural networks

Zhan Niu, Jian Sun, Daniel Trad

Jan 25, 2019

NSERC CRSNG
UNIVERSITY OF CALGARY
FACULTY OF SCIENCE
Department of Geoscience

Outline

- Part I: Born modeling using recurrent neural networks (RNN)
- Part II: Recover velocities from RTM image (reflectivity)
  - Fully-connected
  - Convolutional
- Conclusion
- Future works

Motivation in all

- Born modeling using RNN
  - Bring the advantages of NN and provides a new aspect of the problem
- Solving a non-linear mapping between reflectivity and velocity updates.
  - Reflectivity to Δv update – use as FWI gradient

Part I: Born modeling using Recurrent NN

Part I: Born modeling using Recurrent NN

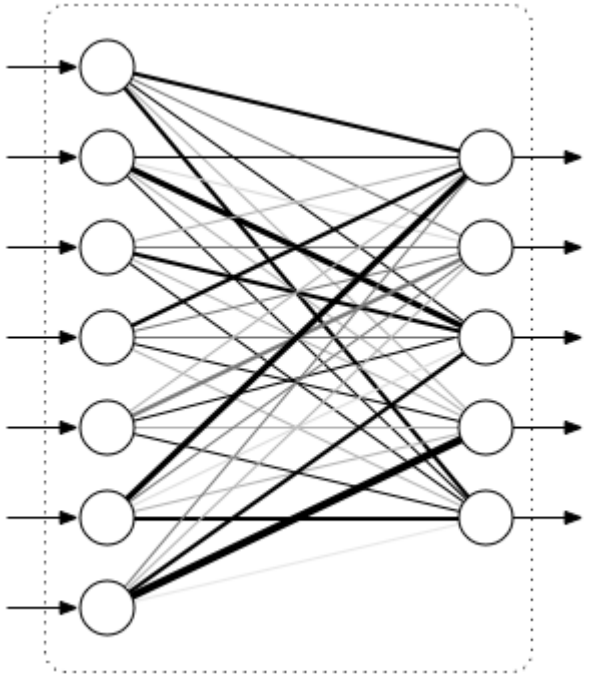
The Born modeling is governed by the following equations

$$\left(\frac{1}{\boldsymbol{v}_0^2}\frac{\partial^2}{\partial t^2} - \nabla^2\right)\boldsymbol{p}_0 = \boldsymbol{f},$$

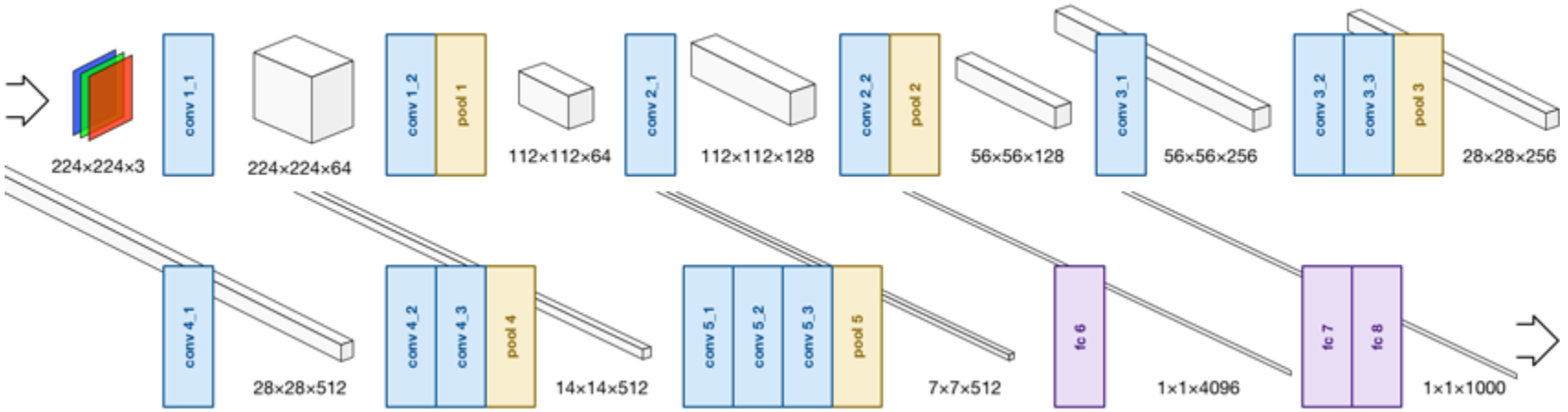
$$\left(\frac{1}{\boldsymbol{v}_0^2}\frac{\partial^2}{\partial t^2} - \nabla^2\right)\delta\boldsymbol{p} = \frac{1}{\boldsymbol{v}_0^2}\cdot\boldsymbol{m}\cdot\frac{\partial^2}{\delta t^2}\boldsymbol{p}_0,$$

Where $\boldsymbol{m} \coloneqq \frac{2\delta\boldsymbol{v}}{\boldsymbol{v}_0}$

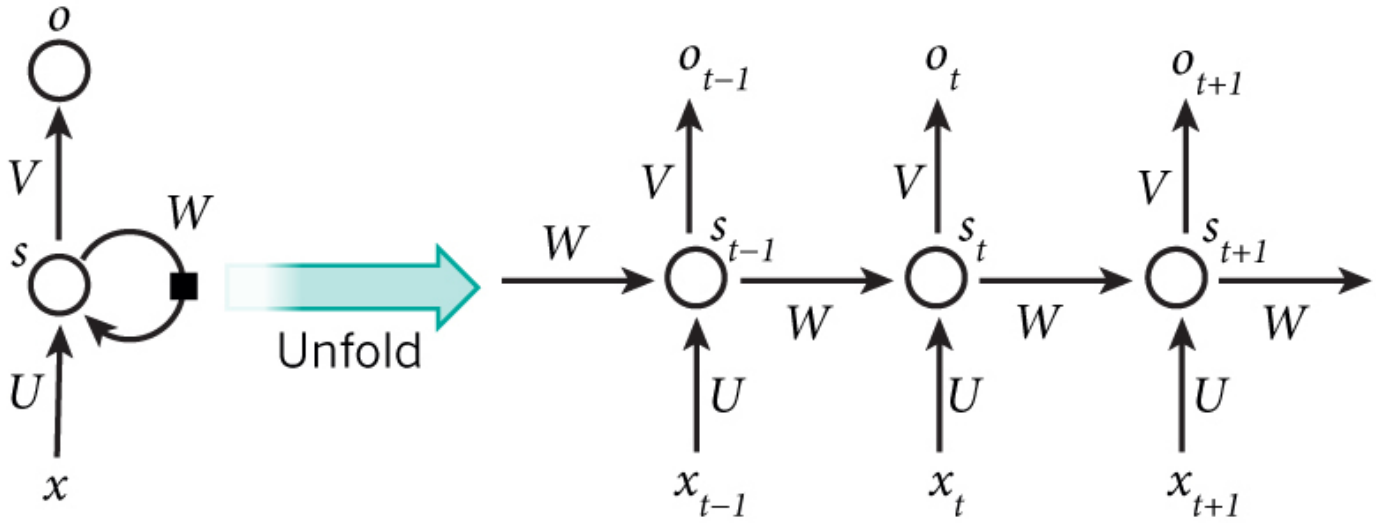
Intro to basic NN types



Fully connected



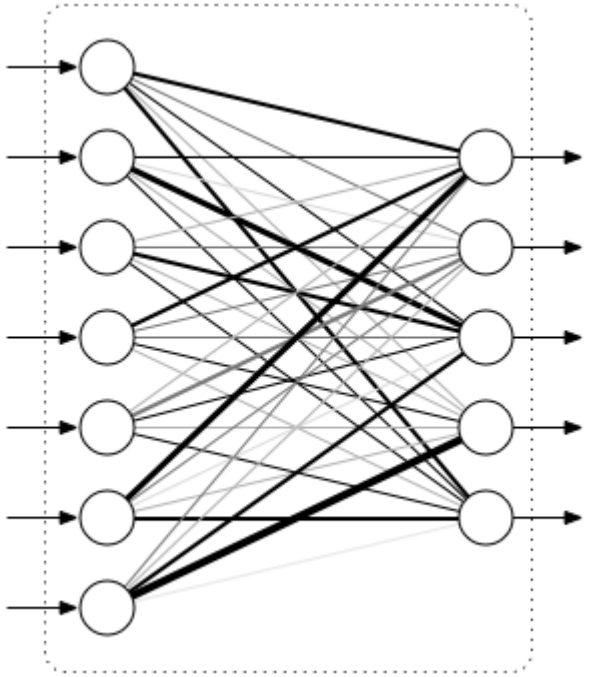
Convolutional



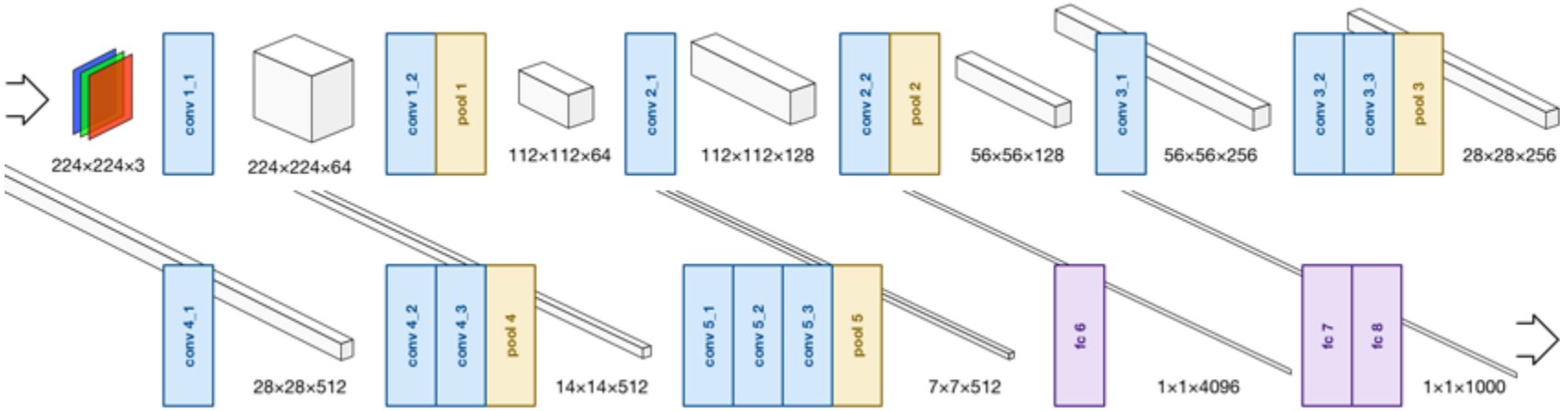
Recurrent

Diagrams from the internet

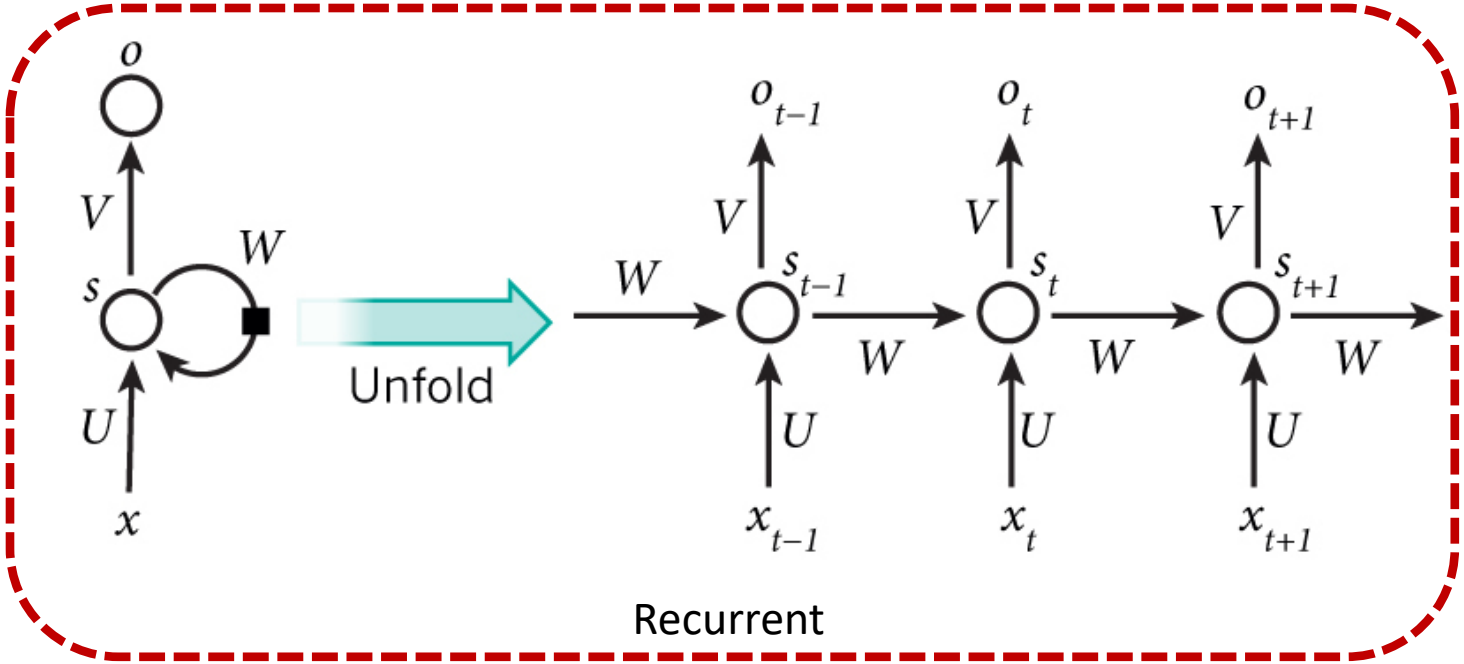
Intro to basic NN types



Fully connected



Convolutional



Recurrent

Diagrams from the internet

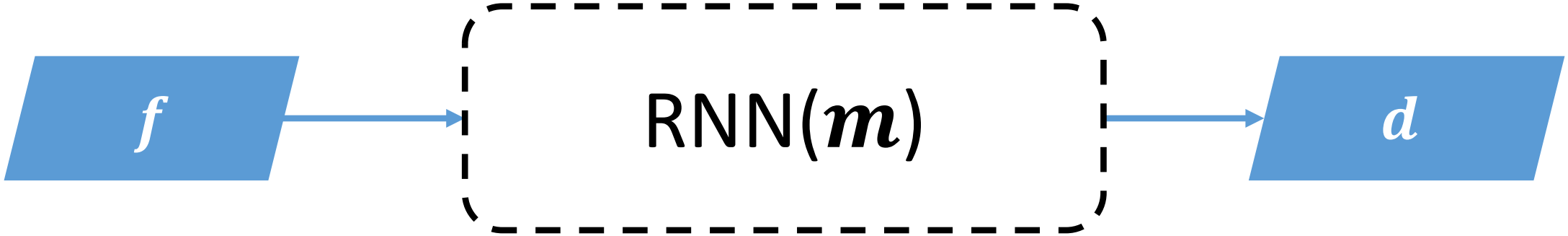
Part I: Born modeling using Recurrent NN

Can we design it better?

Part I: Born modeling using Recurrent NN

Can we design it better?

Part I: Born modeling using Recurrent NN



The forward:

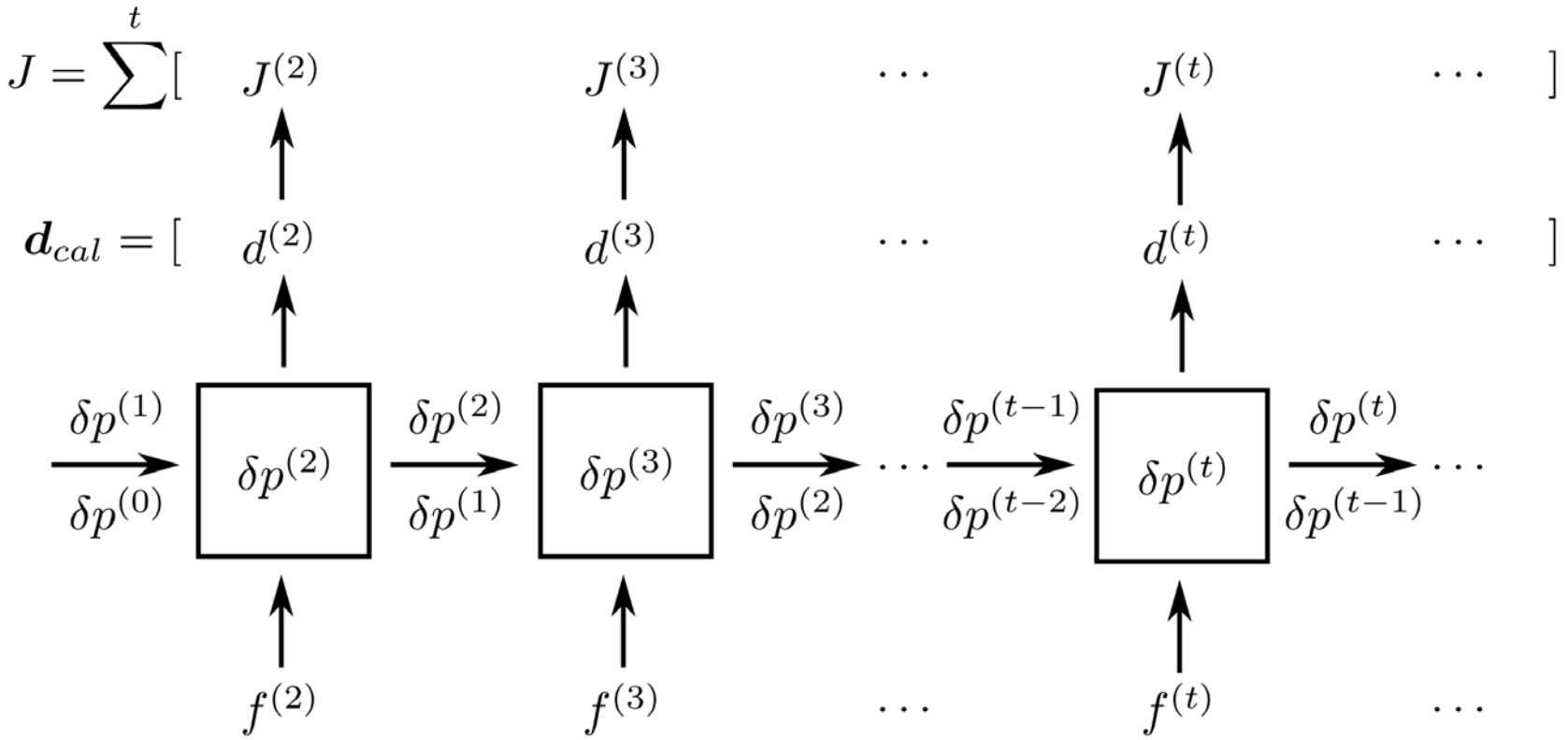
set $\boldsymbol{m}$ to a known value and let the RNN to predict only once to get $\boldsymbol{d}$.

The inverse:

set $\boldsymbol{m}$ to initial value (zeros) and "train" it with $\boldsymbol{d}_{obs}$ for getting updates on $\boldsymbol{m}$

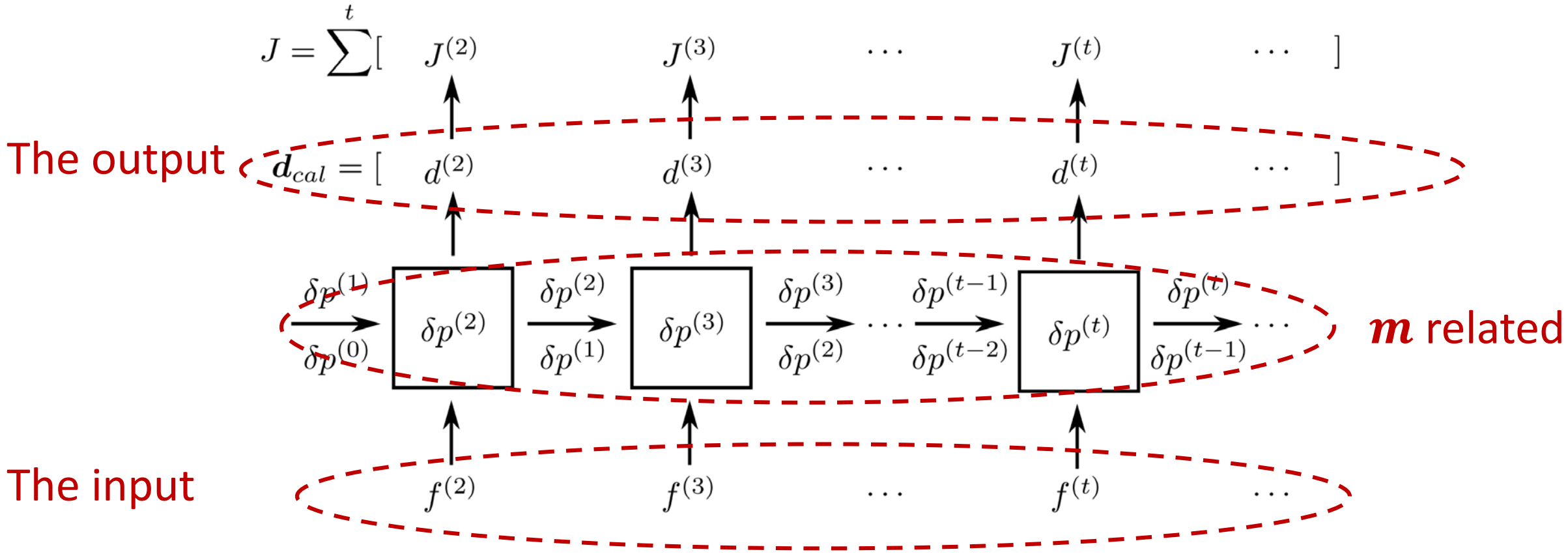
Part I: Born modeling using Recurrent NN

The RNN architecture:

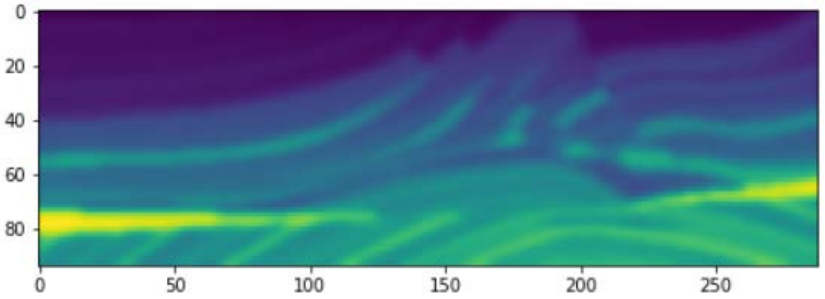


Part I: Born modeling using Recurrent NN

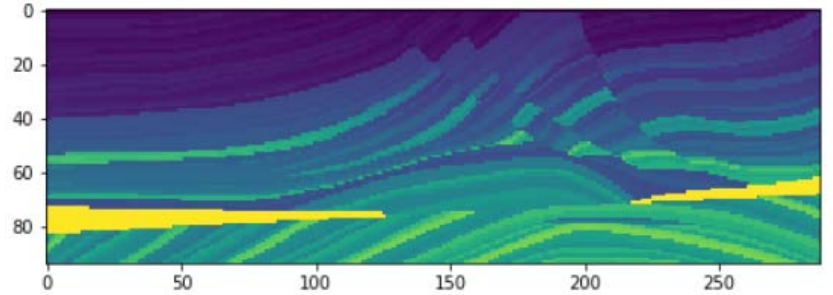
The RNN architecture:



Part I: Results

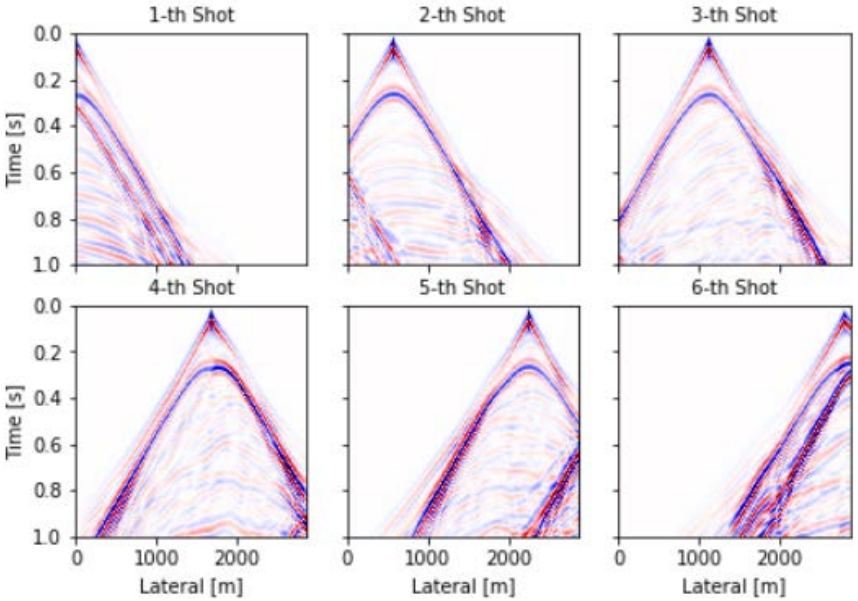


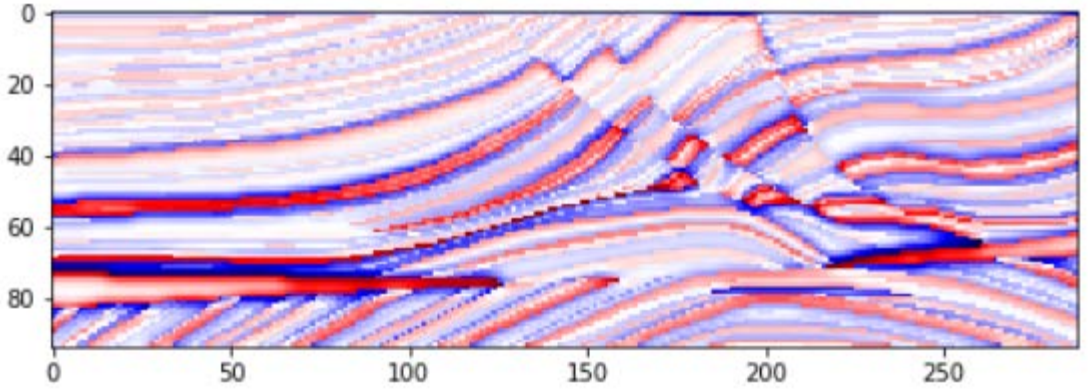
Smoothed



True

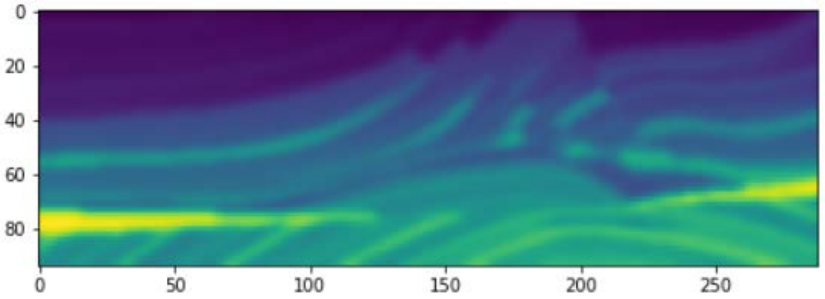
RNN



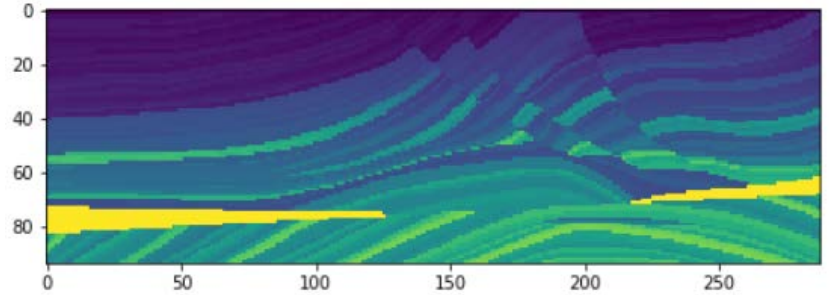


m

Part I: Results

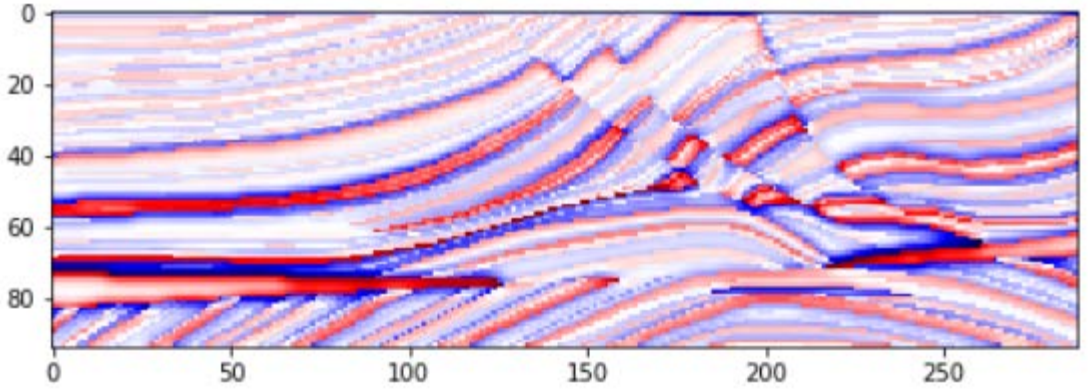


Smoothed



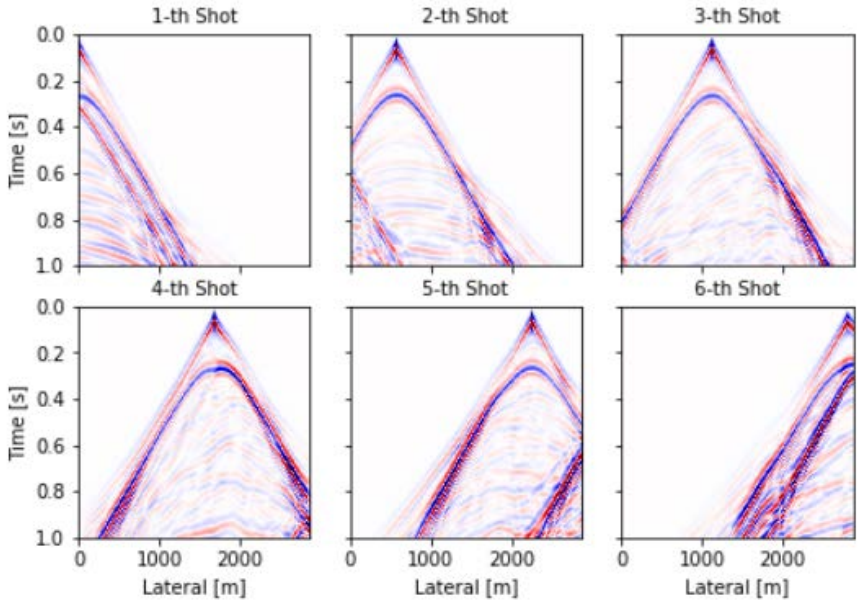
True

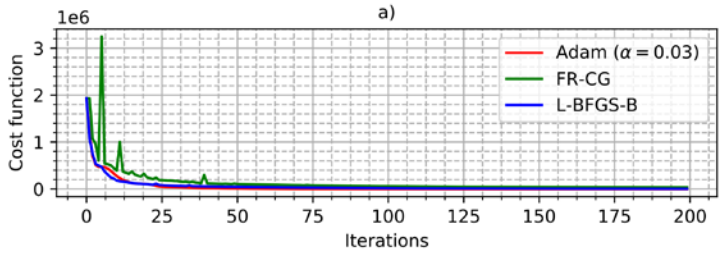
RNN



$\boldsymbol{m}$

200 iterations





Part II: Recover velocities from RTM results

Part II: Define the problem

Input

- RTM image (reflectivity)

n_x 1D array

NN

Output

- Recovered velocity

n_x 1D array

Part II: Fully-connected NN

Model size: $n_x = 1000$

Training: (8:2 split)

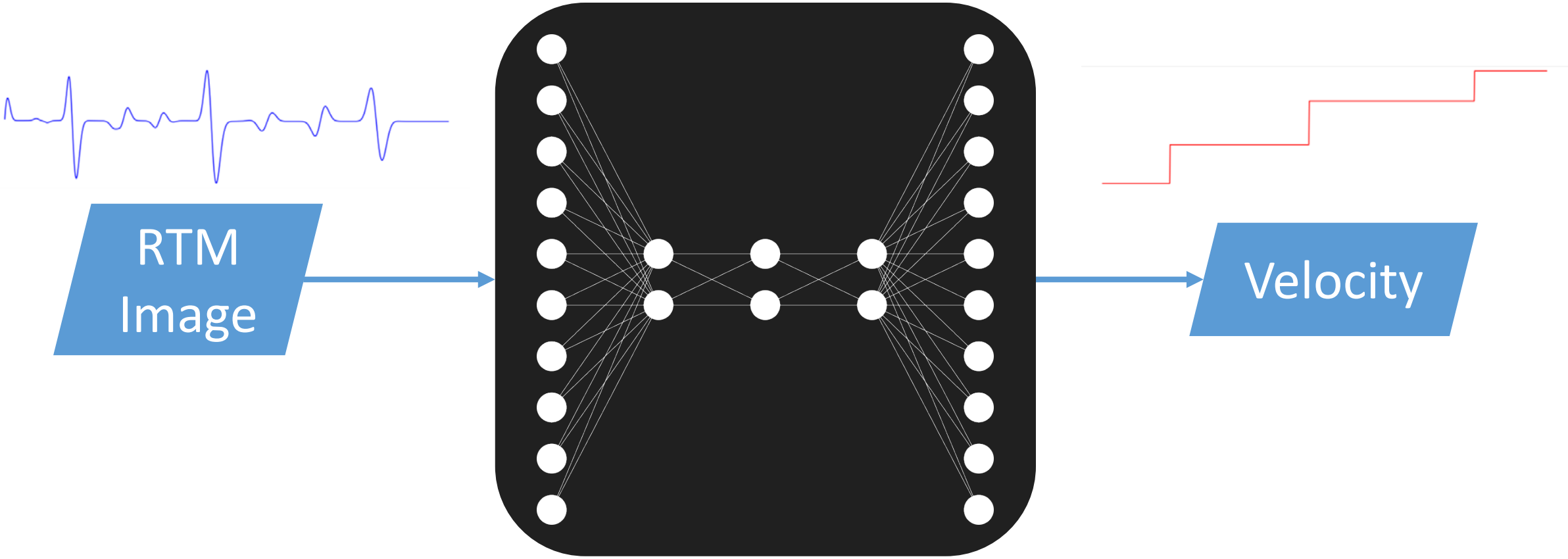
8000 random 1D 4-layer-flat models in the training set

2000 in the cross-validation set

Testing:

Images come from the same model distribution (or not)

Part II: Fully-connected NN



input -> 1000 -> 200 -> 200 -> 200 -> 1000 -> output

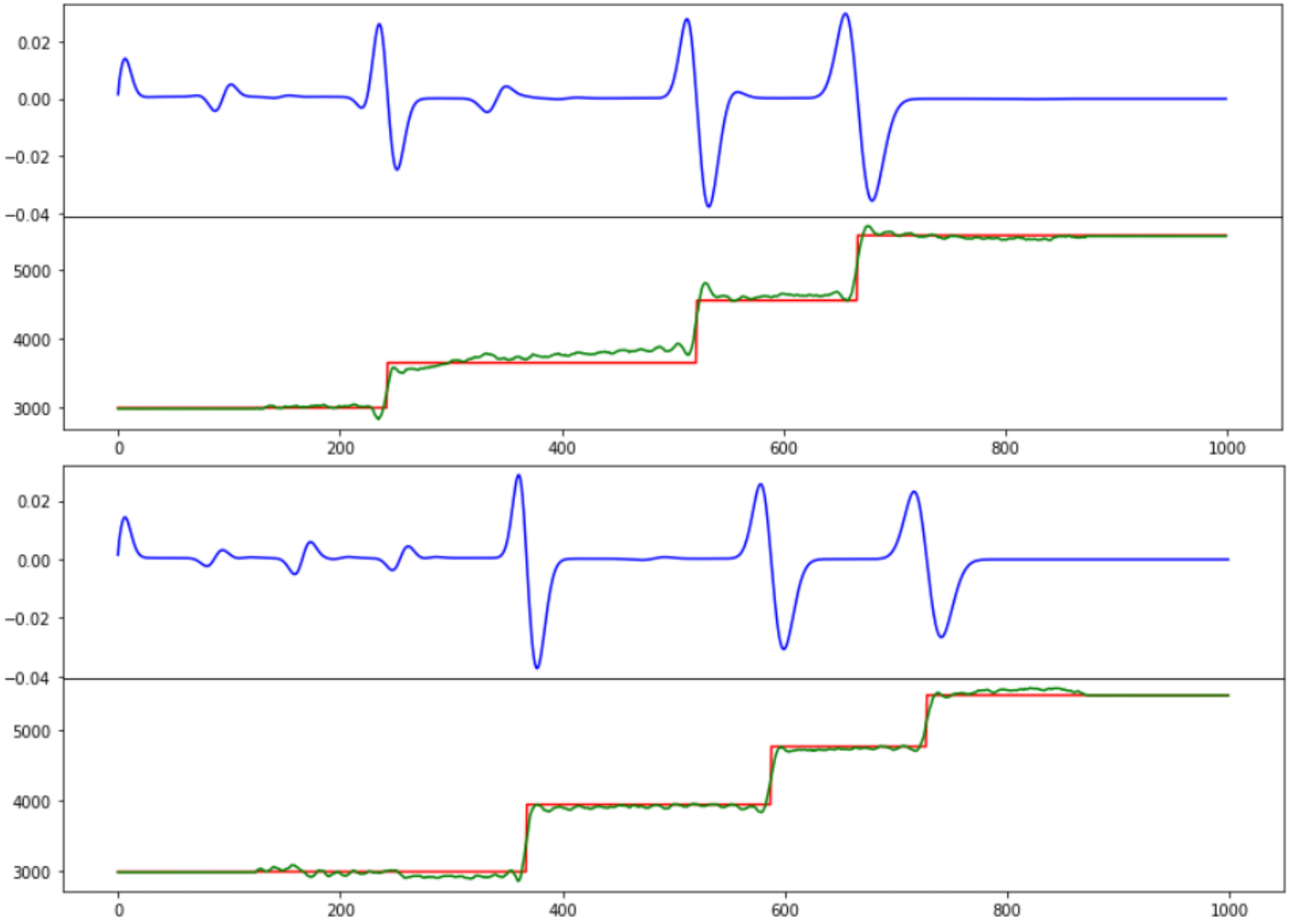
5 layers with 3 hidden

~ 2600 nodes

NN

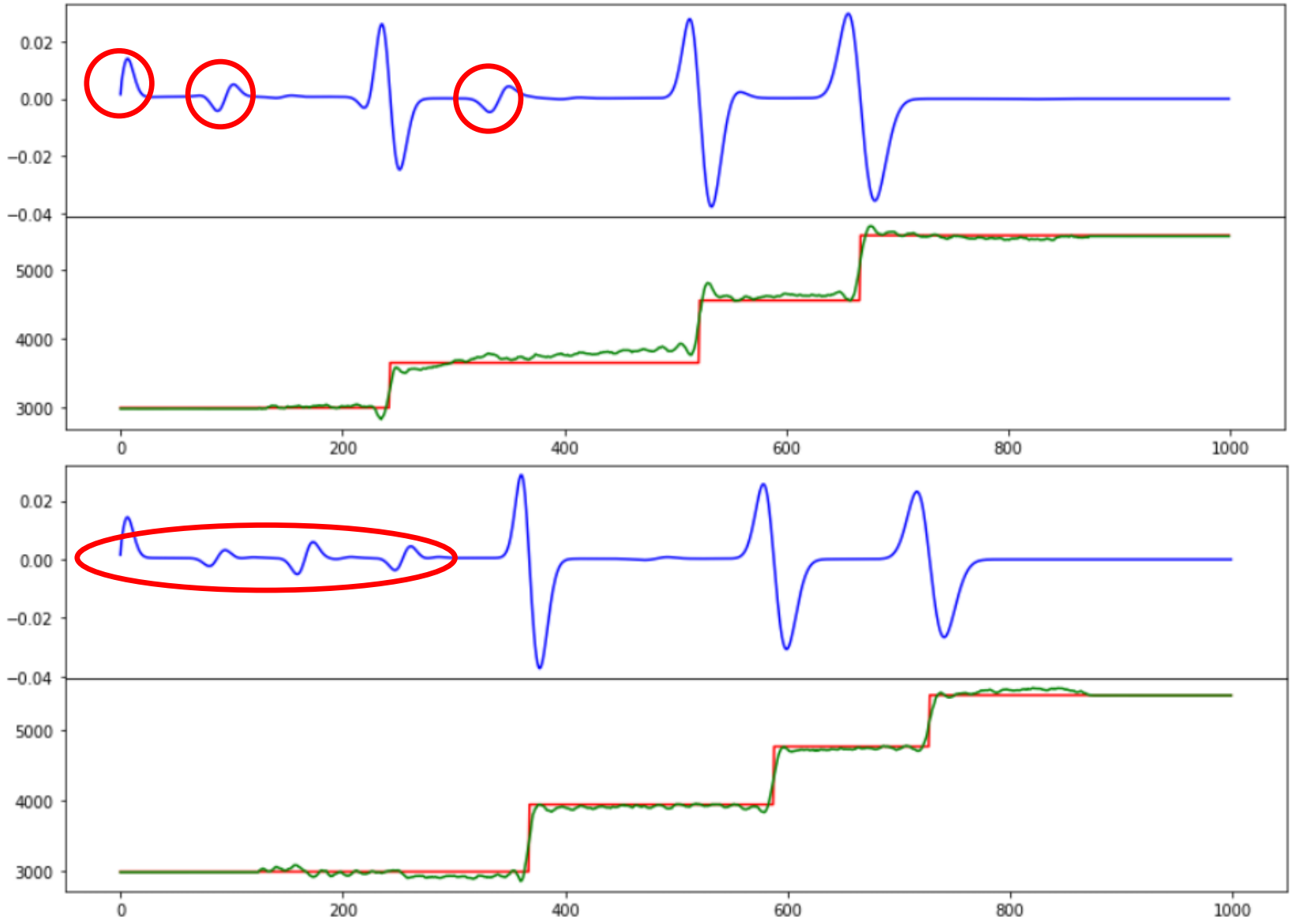
Part II: Fully-connected NN

5 layers, ~2600 weights + 2600 bias



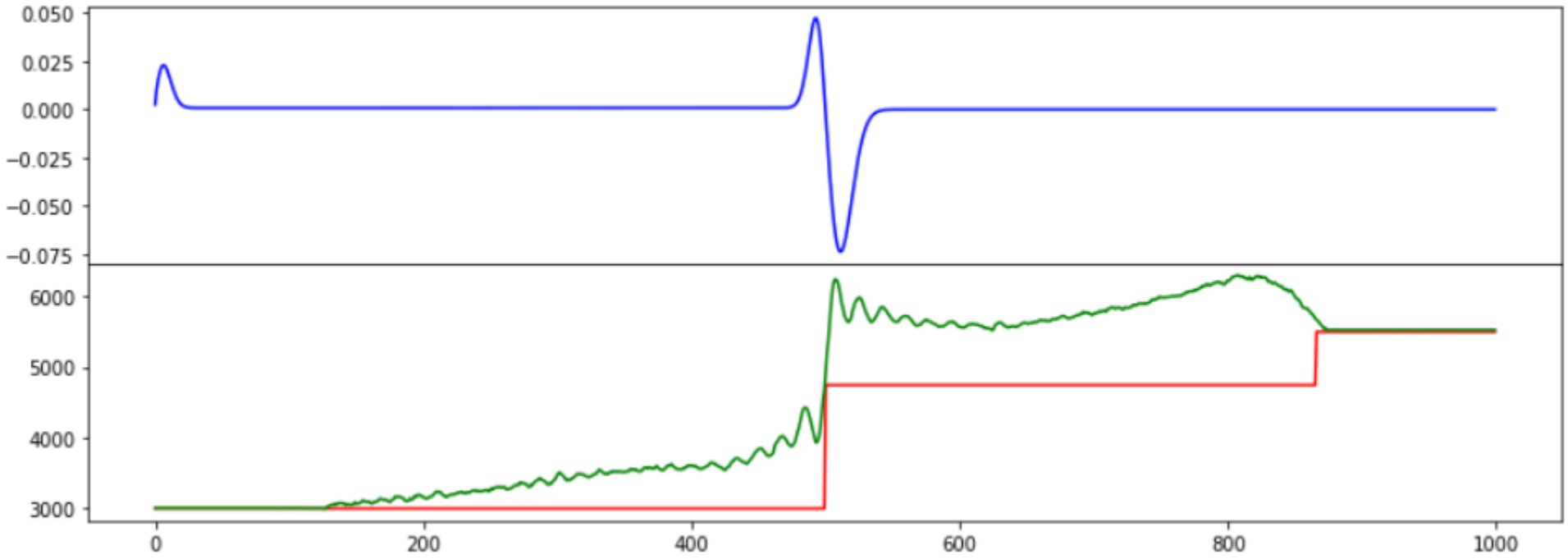
Part II: Fully-connected NN

5 layers, ~ 2600 weights + 2600 bias



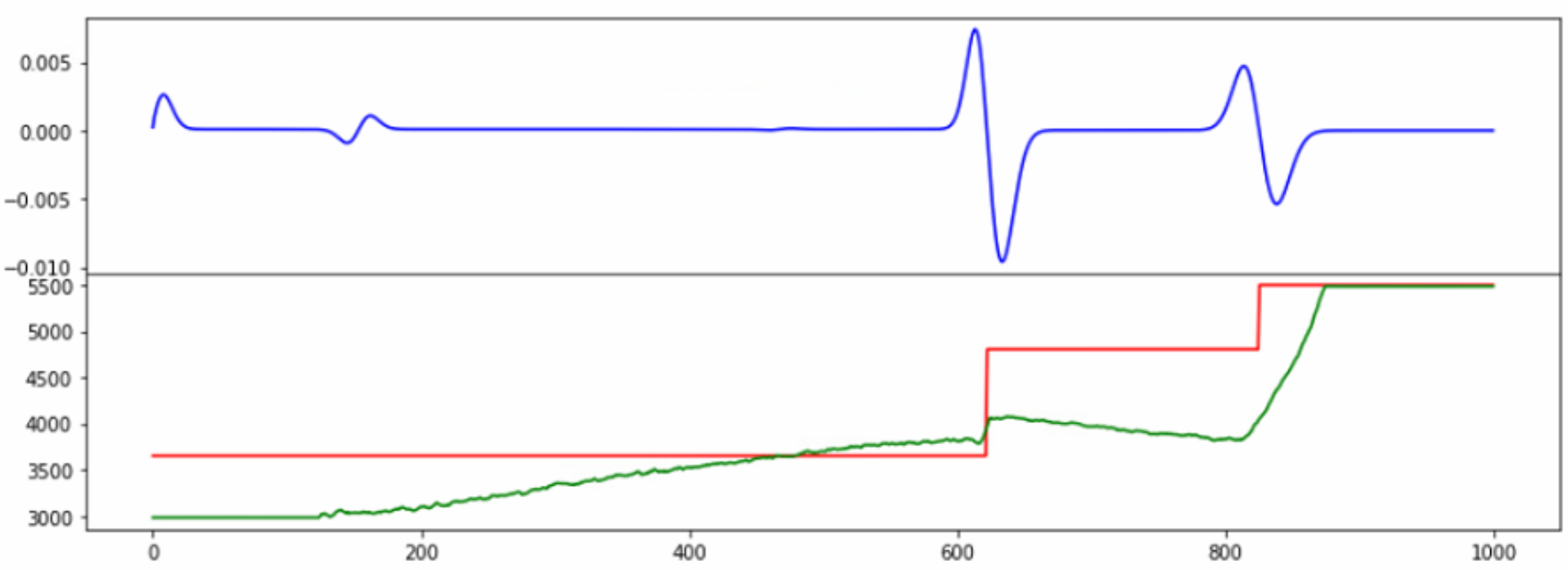
Part II: Fully-connected NN: limitations

On a 3 layer model:



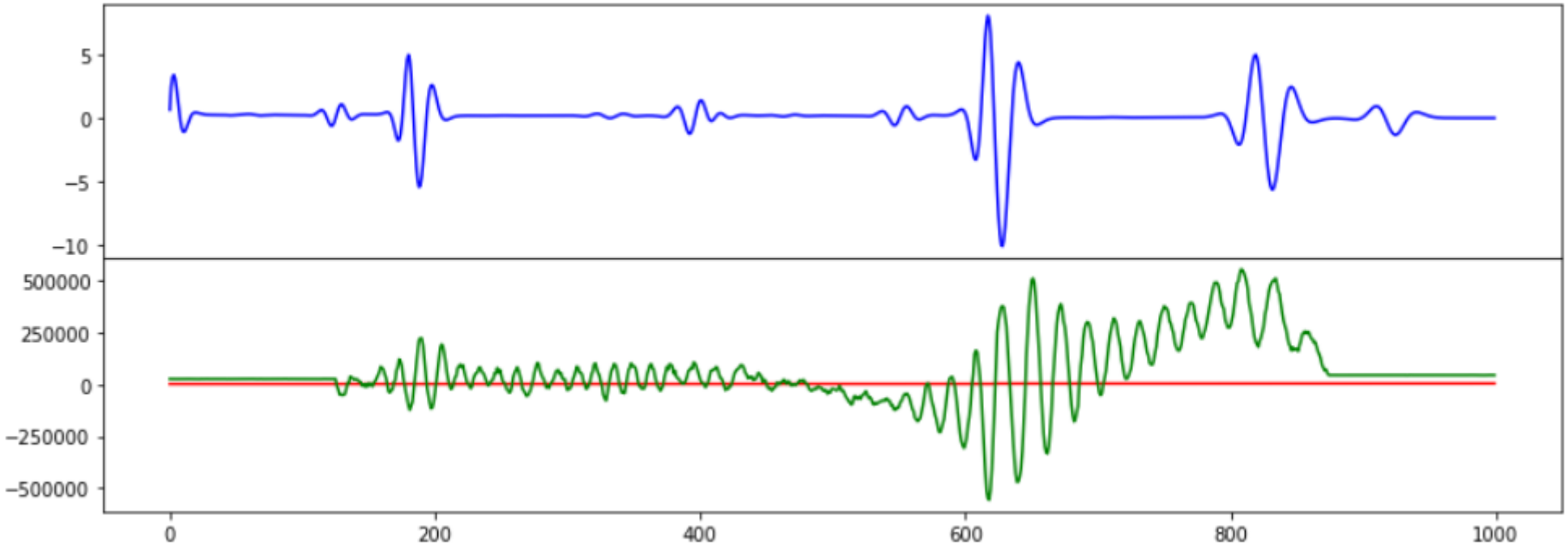
Part II: Fully-connected NN: limitations

Wrong background velocity:

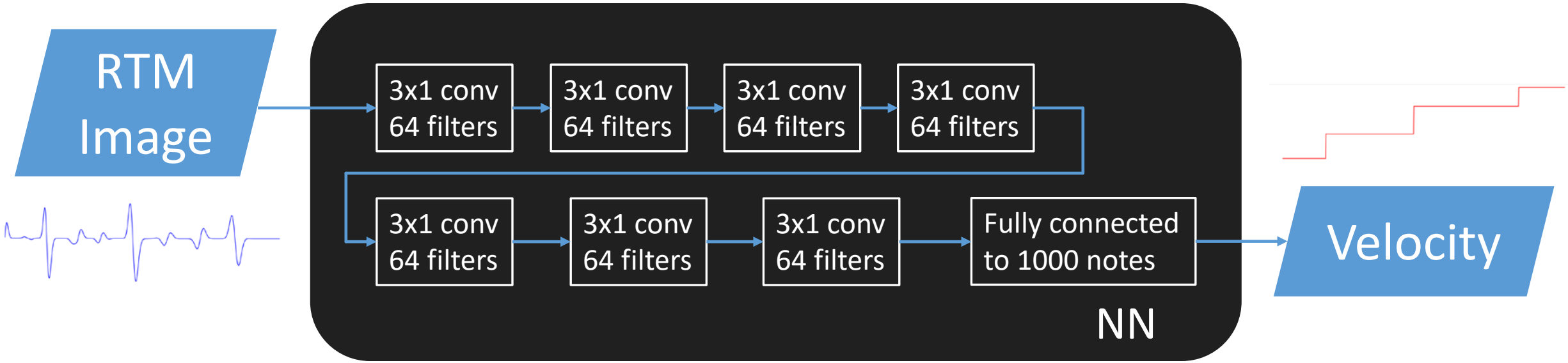


Part II: Fully-connected NN: limitations

Wrong wavelet:



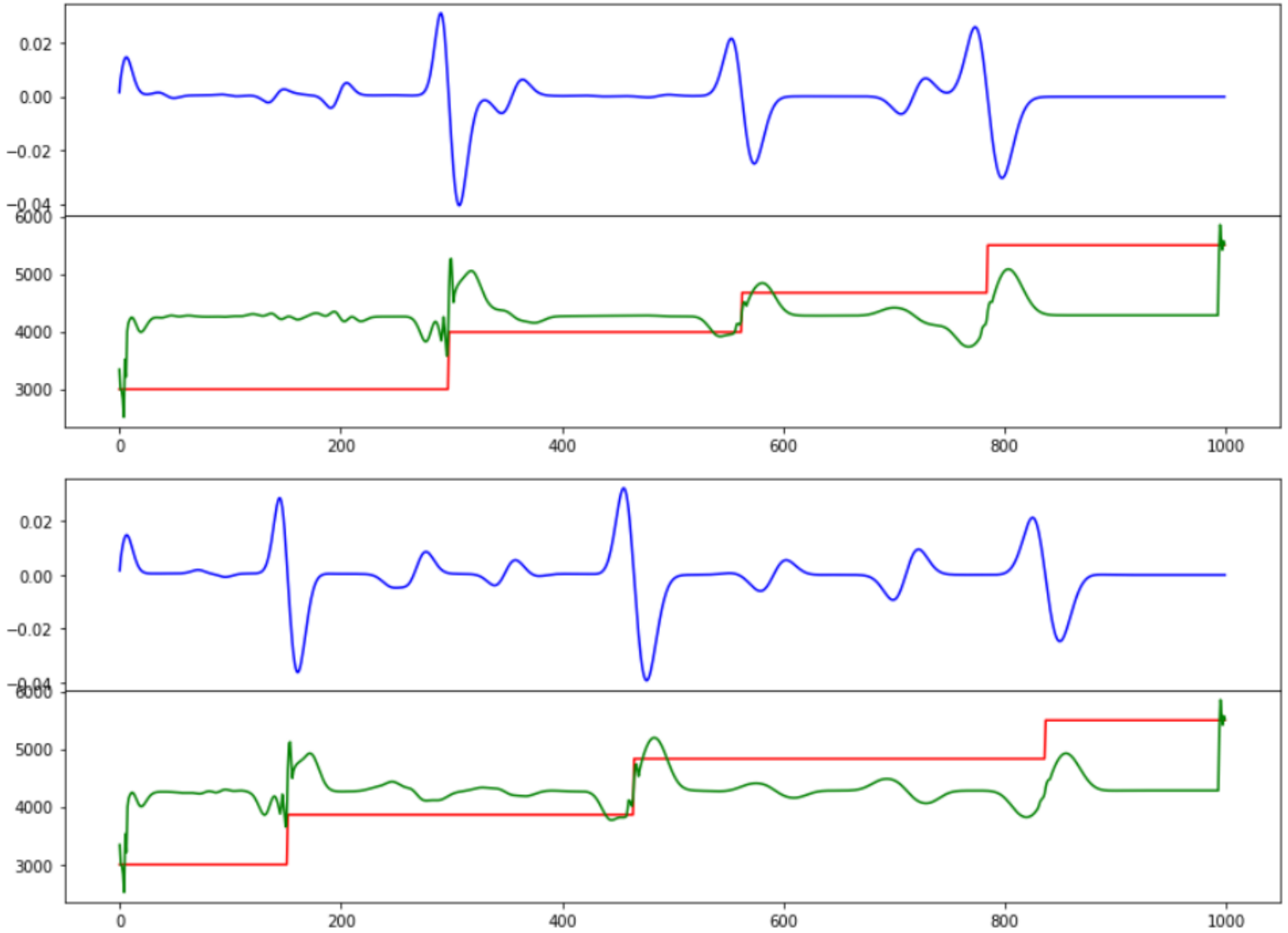
Part II: Convolutional NN – version II



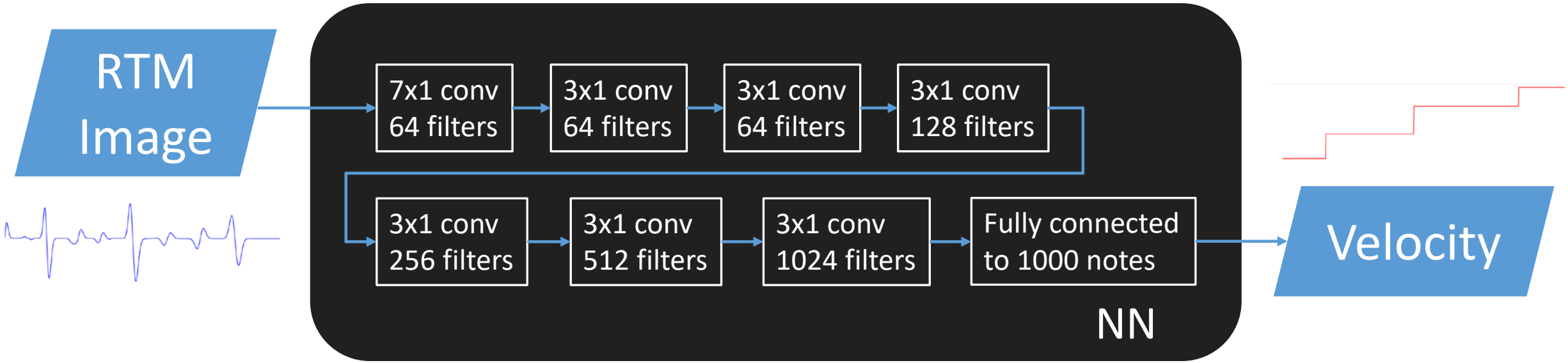
8 hidden layers

~ 2344 nodes

Part II: Convolutional NN



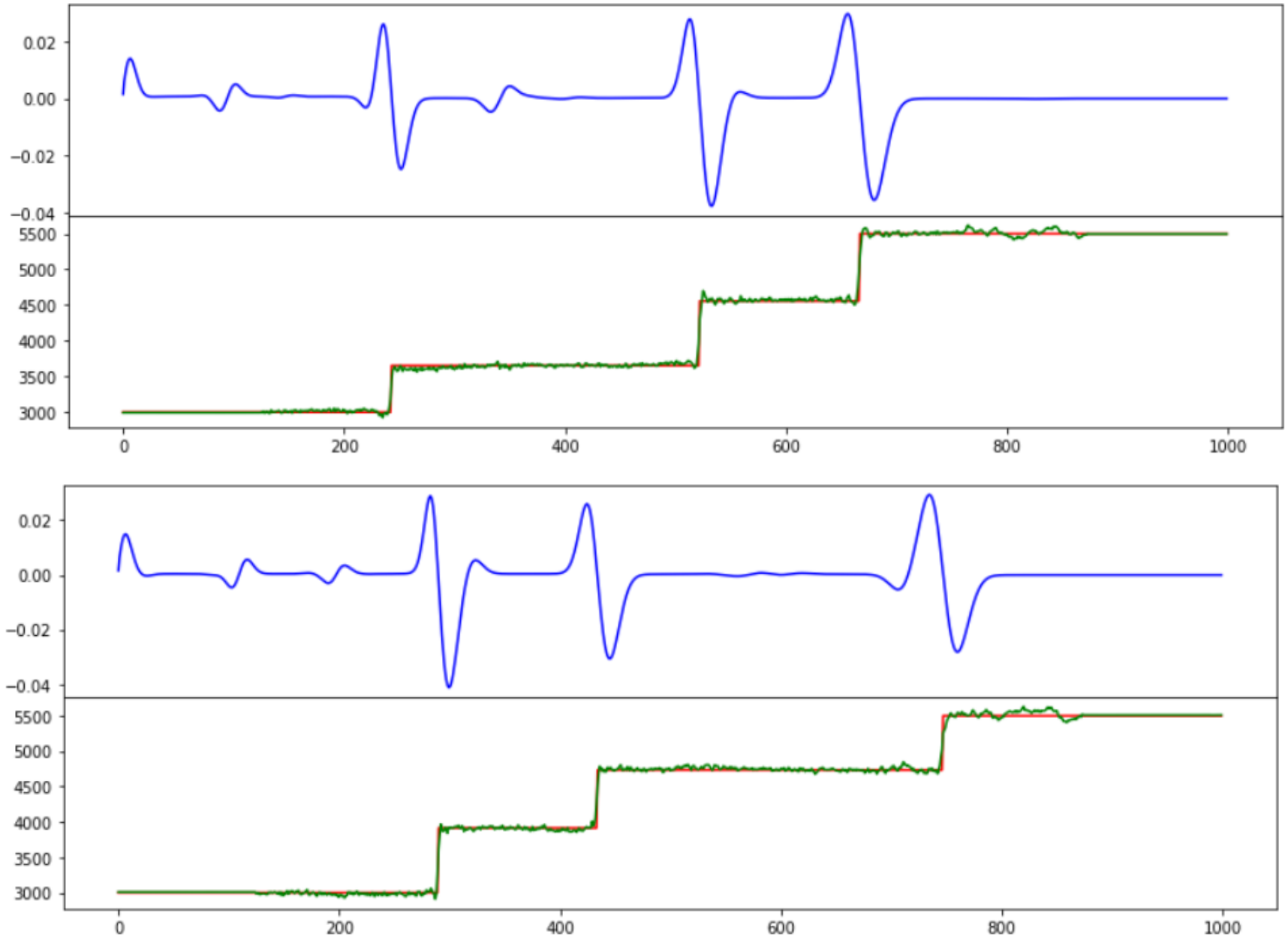
Part II: Convolutional NN – version II



8 hidden layers

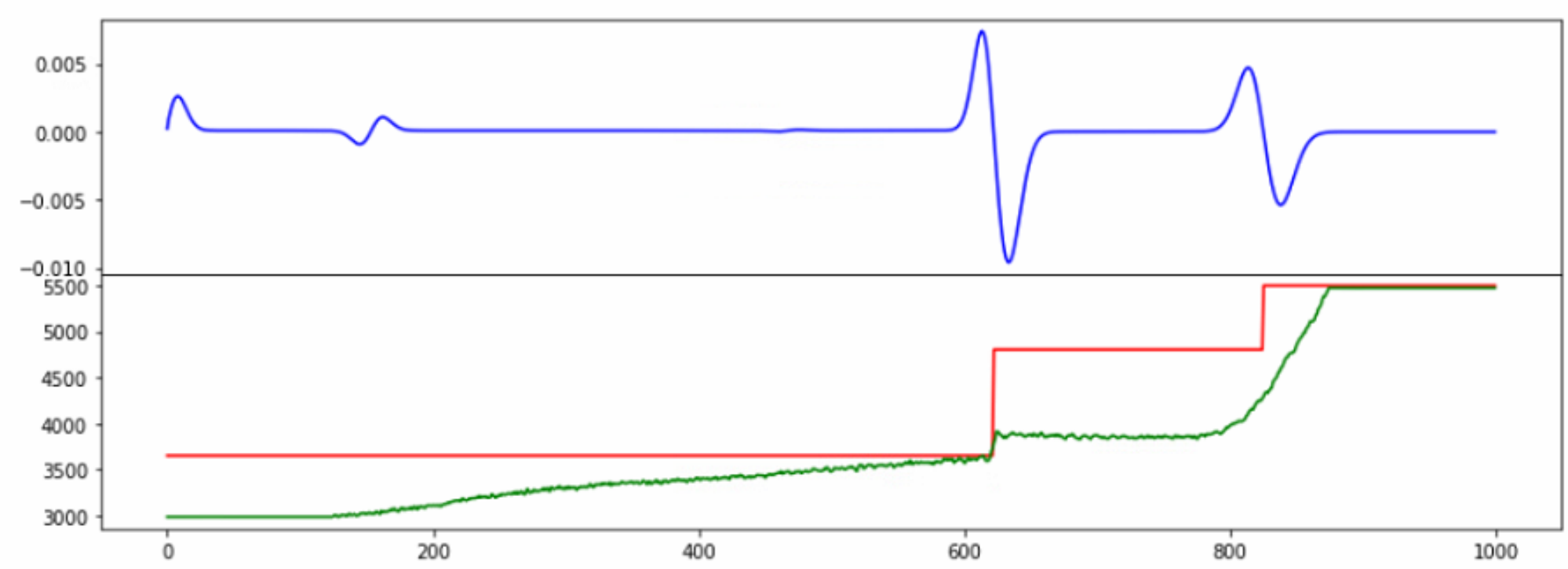
~ 7592 nodes

Part II: Convolutional NN – version II



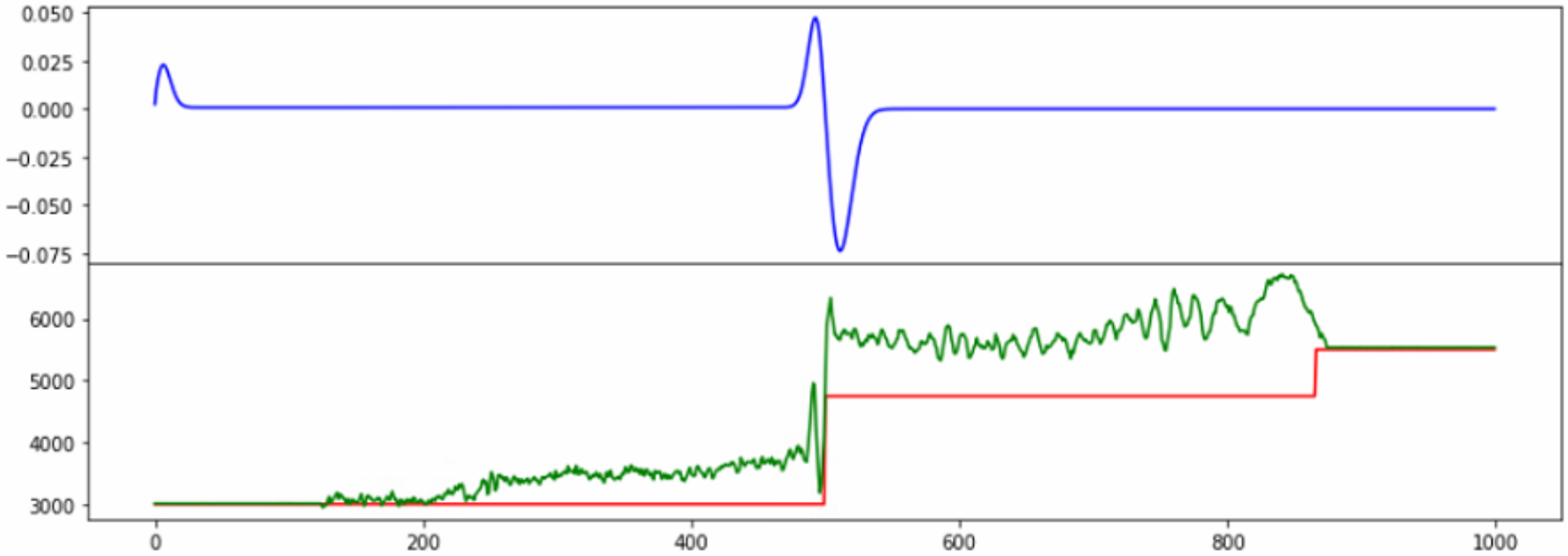
Part II: Convolutional NN: limitations

Wrong background velocity:



Part II: Convolutional NN: limitations

On a 3 layer model:



Conclusions

- NNs is …
  - dependent on model distribution
  - highly dependent on the wavelet
- Fully-connected NN outperforms Convolutional for this problem:
  - Recovered velocities depends not only the nearby features
  - Structures are crucial for solving geophysics problems

Future works:

- Find a more suitable structures/combination of NN
- Integrate small scale NNs into traditional algorithms
- Choose a better cost function that expresses more geophysics

Acknowledgements

- All CREWES sponsors
- NSERC (CRDPJ 46117913)

and thank you

Part I: Born modeling using Recurrent NN

Results: