Neural network-based seismic event detector:

an application to swarm-like earthquake in West Bohemia and South-west Iceland

J. Doubravová and J. Horálek

Workshop CzechGeo/EPOS, 22.11.2017

EUROPEAN UNION
European Structural and Investment Funds
Operational Programme Research,
Development and Education

MINISTRY OF EDUCATION,
YOUTH AND SPORTS

Outline

Motivation

- continual data are not suitable for direct manual processing
- high quality detection needed for
  - manual processing, i.e. we need minimum number of false alarms
  - automatic processing - detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast

Motivation

- continual data are not suitable for direct manual processing
- high quality detection needed for
    - manual processing, i.e. we need minimum number of false alarms
    - automatic processing - detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast

Motivation

- continual data are not suitable for direct manual processing
- high quality detection needed for
  - manual processing, i.e. we need minimum number of false alarms
  - automatic processing - detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast

Motivation

- continual data are not suitable for direct manual processing
- high quality detection needed for
  - manual processing, i.e. we need minimum number of false alarms
  - automatic processing - detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast

Motivation

- continual data are not suitable for direct manual processing
- high quality detection needed for
  - manual processing, i.e. we need minimum number of false alarms
  - automatic processing - detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast

Motivation

- continual data are not suitable for direct manual processing
- high quality detection needed for
    - manual processing, i.e. we need minimum number of false alarms
    - automatic processing - detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast

Neural network

- artificial neural networks has been inspired by biological neural networks
- acts as an interface between the organism and environment, reacts to inner and outer stimuli
- sensors (=receptors), information is spread through the network to effectors (muscles, glands)
- in cerebral cortex 15-33 billion neurons, each neuron connected to up to 5000 other neurons

Neural network

- artificial neural networks has been inspired by biological neural networks
- acts as an interface between the organism and environment, reacts to inner and outer stimuli
- sensors (=receptors), information is spread through the network to effectors (muscles, glands)
- in cerebral cortex 15-33 billion neurons, each neuron connected to up to 5000 other neurons

Neural network

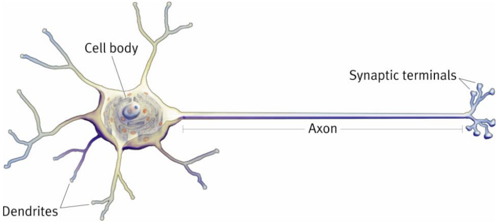
- artificial neural networks has been inspired by biological neural networks
- acts as an interface between the organism and environment, reacts to inner and outer stimuli
- sensors (=receptors), information is spread through the network to effectors (muscles, glands)
- in cerebral cortex 15-33 billion neurons, each neuron connected to up to 5000 other neurons

Neural network

- artificial neural networks has been inspired by biological neural networks
- acts as an interface between the organism and environment, reacts to inner and outer stimuli
- sensors (=receptors), information is spread through the network to effectors (muscles, glands)
- in cerebral cortex 15-33 billion neurons, each neuron connected to up to 5000 other neurons

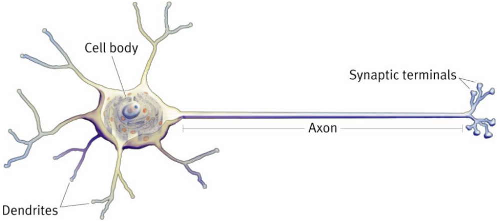
Anatomy of a neuron

- body, dendrites (inputs), axon and synaptic terminals (outputs)
- synaptic weight between dendrite and axon (inhibition or excitation)
- synapsis can be build (learning) or disconnected (forgetting)
- neuron generates an electrical impulse if the activity of dendrites is strong enough (information propagates)



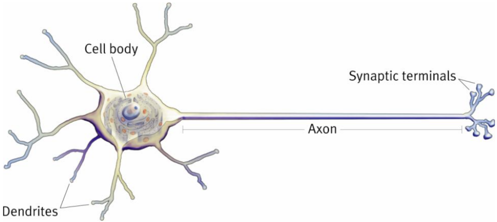
Anatomy of a neuron

- body, dendrites (inputs), axon and synaptic terminals (outputs)
- synaptic weight between dendrite and axon (inhibition or excitation)
- synapsis can be build (learning) or disconnected (forgetting)
- neuron generates an electrical impulse if the activity of dendrites is strong enough (information propagates)



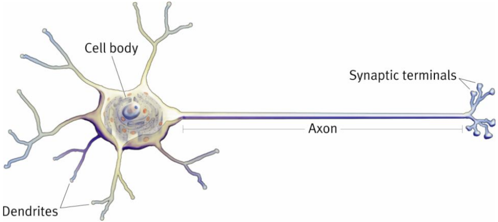
Anatomy of a neuron

- body, dendrites (inputs), axon and synaptic terminals (outputs)
- synaptic weight between dendrite and axon (inhibition or excitation)
- synapsis can be build (learning) or disconnected (forgetting)
- neuron generates an electrical impulse if the activity of dendrites is strong enough (information propagates)

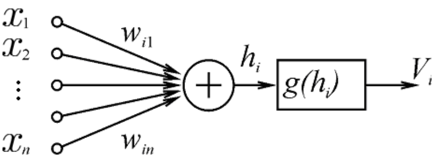


Anatomy of a neuron

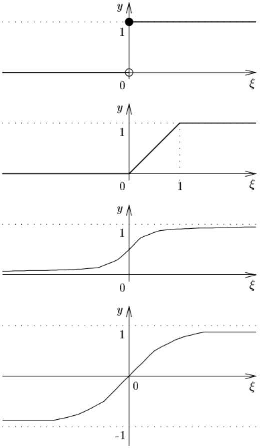
- body, dendrites (inputs), axon and synaptic terminals (outputs)
- synaptic weight between dendrite and axon (inhibition or excitation)
- synapsis can be build (learning) or disconnected (forgetting)
- neuron generates an electrical impulse if the activity of dendrites is strong enough (information propagates)



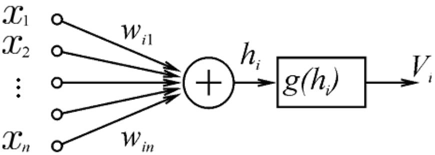
Artificial neuron

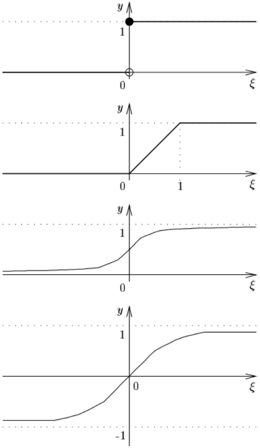


- n real inputs x = dendrites, threshold input $x_0 = 1$
- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$



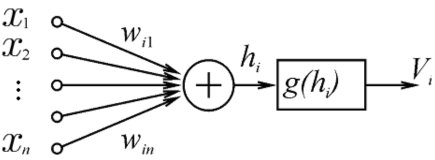
Artificial neuron

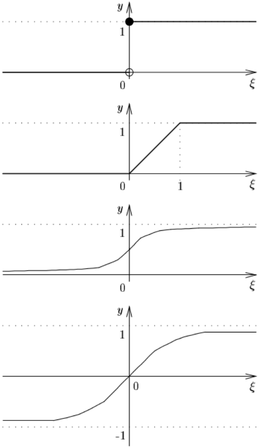




- n real inputs x = dendrites, threshold input $x_0 = 1$
- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$

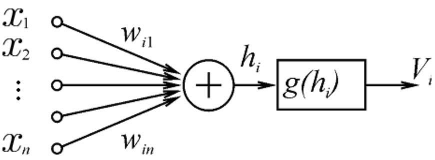
Artificial neuron

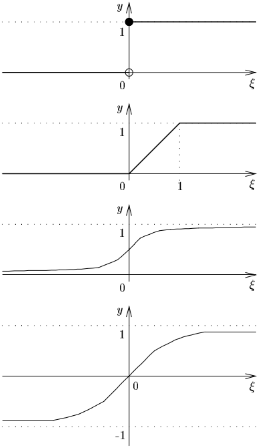




- n real inputs x = dendrites, threshold input $x_0 = 1$
- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$

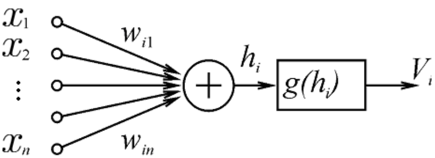
Artificial neuron

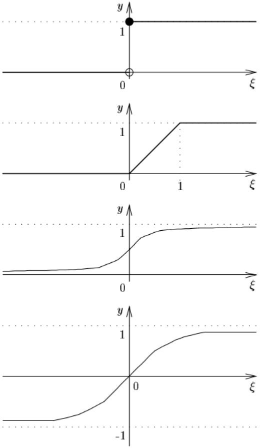




- n real inputs x = dendrites, threshold input $x_0 = 1$
- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$

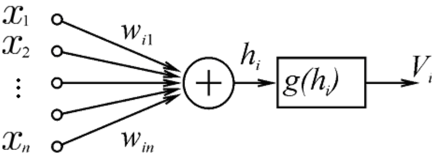
Artificial neuron

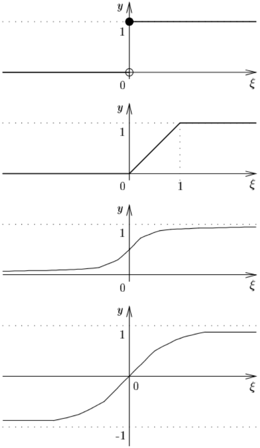




- n real inputs x = dendrites, threshold input $x_0 = 1$
- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$

Artificial neuron

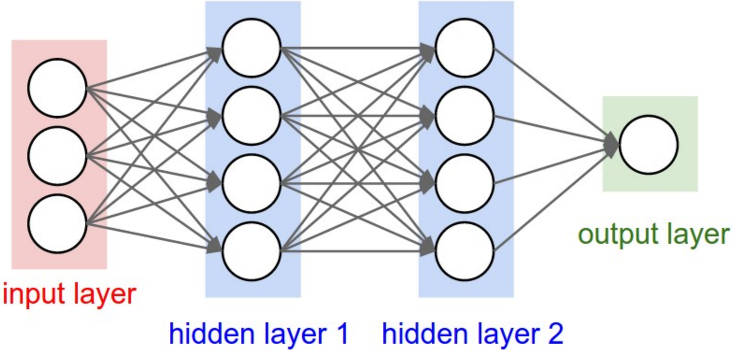




- n real inputs x = dendrites, threshold input $x_0 = 1$
- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$

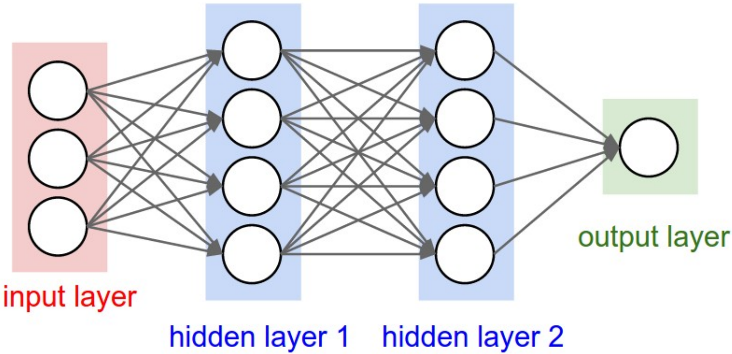
Artificial neural network

- for more complex problems
- typical application: classification, pattern recognition, regression



Artificial neural network

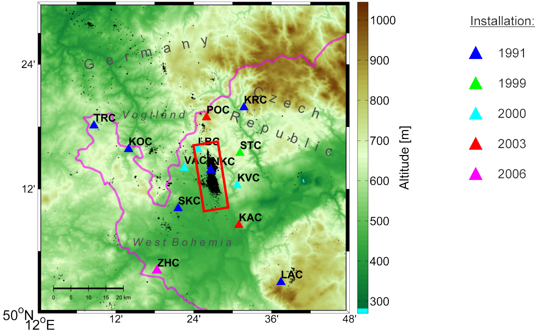
- for more complex problems
- typical application: classification, pattern recognition, regression



Outline

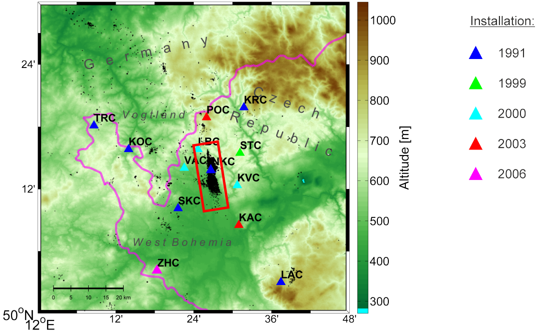
1. Artificial neural networks - basics
2. Data and method
   - WEBNET data
   - Single Layer Recurrent Neural Network
   - Training of the SLRNN
3. Training results
4. Application to Reykjanet

WEBNET data



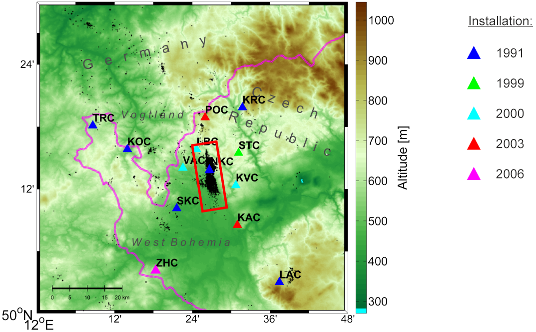
- 23 stations
- 15 on-line stations
- 250Hz, 3-component velocity records

WEBNET data



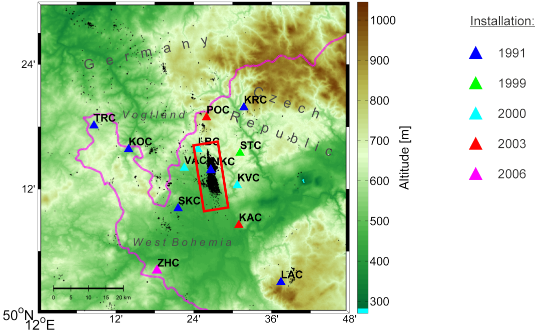
- 23 stations
- 15 on-line stations
- 250Hz, 3-component velocity records

WEBNET data



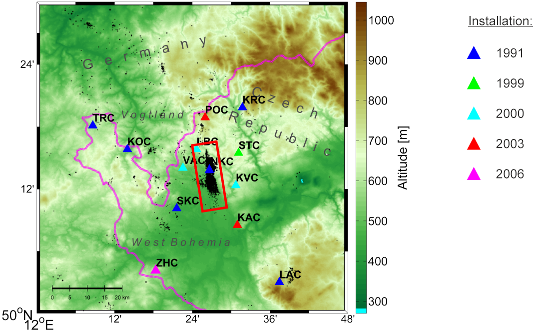
- 23 stations
- 15 on-line stations
- 250Hz, 3-component velocity records

WEBNET data



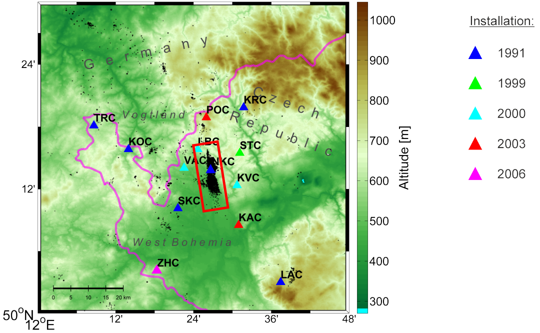
- 23 stations
- 15 on-line stations
- 250Hz, 3-component velocity records

WEBNET data



- 23 stations
- 15 on-line stations
- 250Hz, 3-component velocity records

WEBNET data



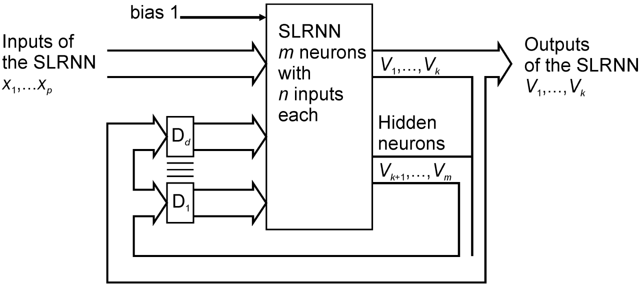
- 23 stations
- 15 on-line stations
- 250Hz, 3-component velocity records

Outline

1. Artificial neural networks - basics
2. Data and method
    - WEBNET data
    - Single Layer Recurrent Neural Network
    - Training of the SLRNN
3. Training results
4. Application to Reykjanet

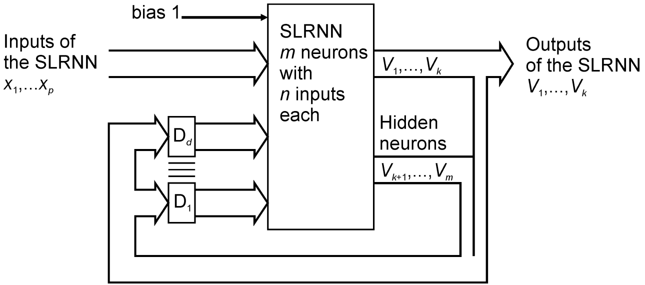
SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays $D_1..D_d$



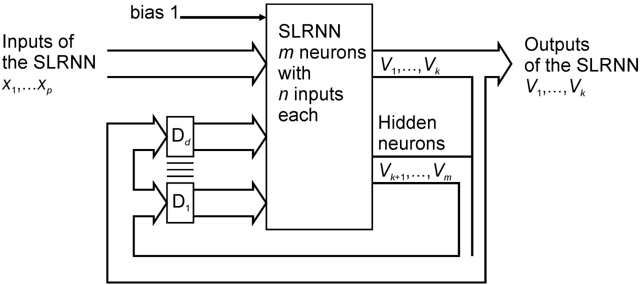
SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays $D_1..D_d$



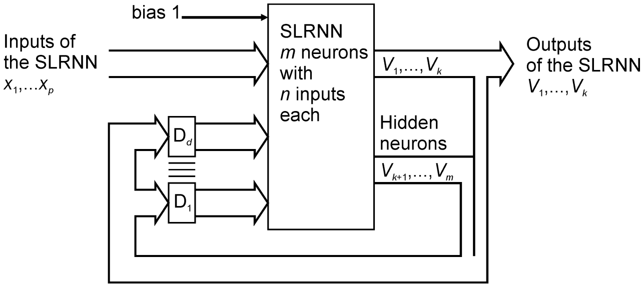
SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays $D_1..D_d$



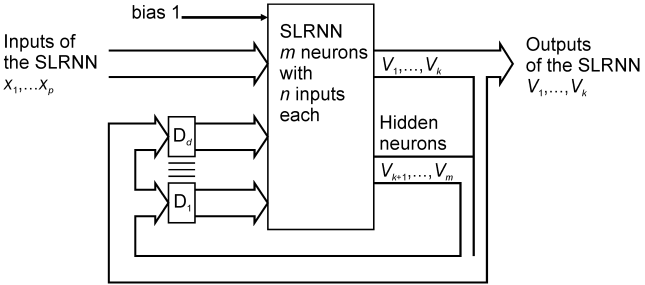
SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays $D_1..D_d$



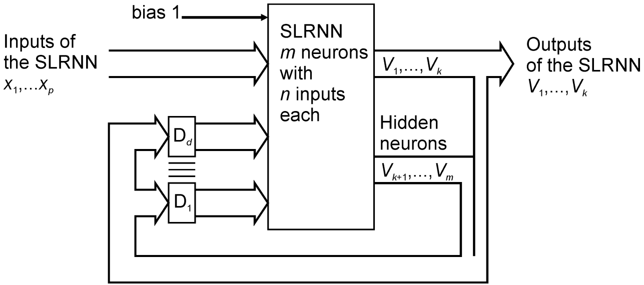
SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays $D_1..D_d$



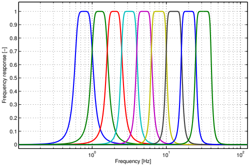
SLRNN - architecture

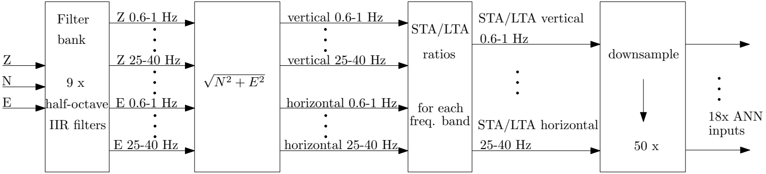
- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays $D_1..D_d$



SLRNN - input signals

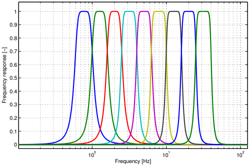
- vertical and horizontal component
- STA/LTA in 9 half-octave frequency bands

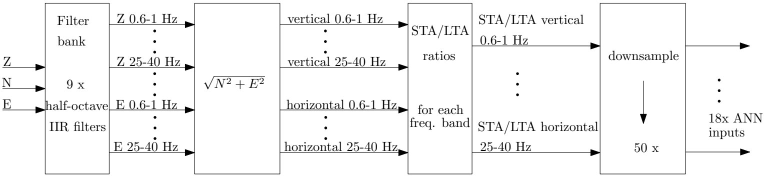




SLRNN - input signals

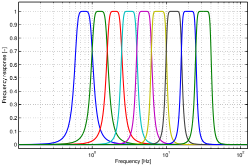
- vertical and horizontal component
- STA/LTA in 9 half-octave frequency bands

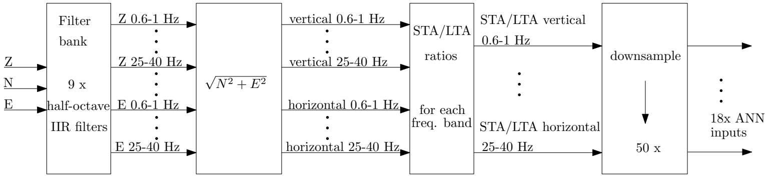




SLRNN - input signals

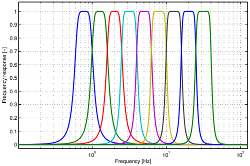
- vertical and horizontal component
- STA/LTA in 9 half-octave frequency bands

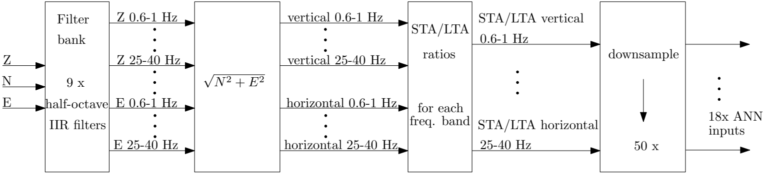




SLRNN - input signals

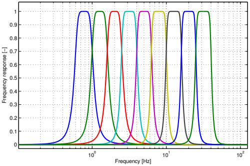
- vertical and horizontal component
- STA/LTA in 9 half-octave frequency bands

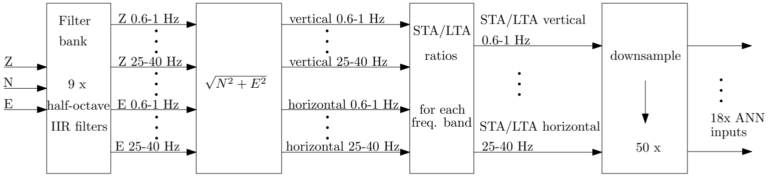




SLRNN - input signals

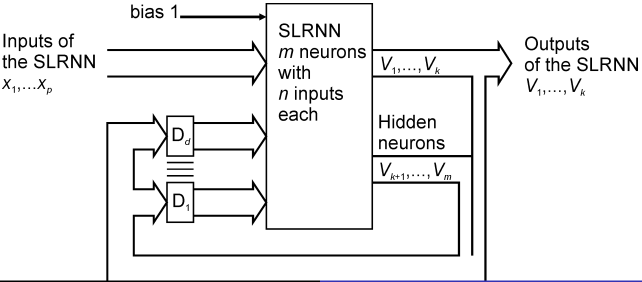
- vertical and horizontal component
- STA/LTA in 9 half-octave frequency bands





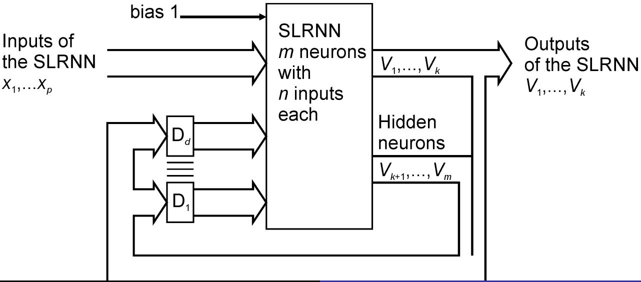
SLRNN - architecture

- 8 neurons, 18 inputs, 3 outputs (**event**, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust



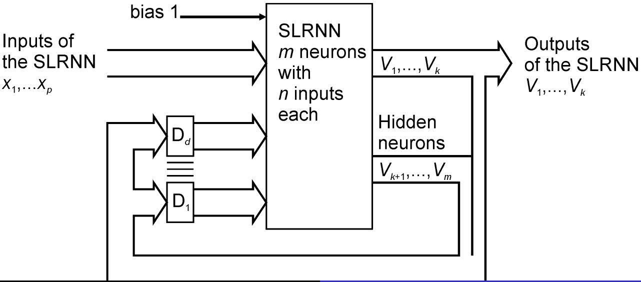
SLRNN - architecture

- 8 neurons, 18 inputs, 3 outputs (**event**, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust



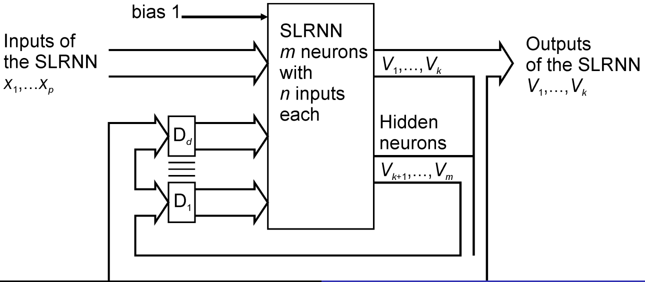
SLRNN - architecture

- 8 neurons, 18 inputs, 3 outputs (**event**, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust



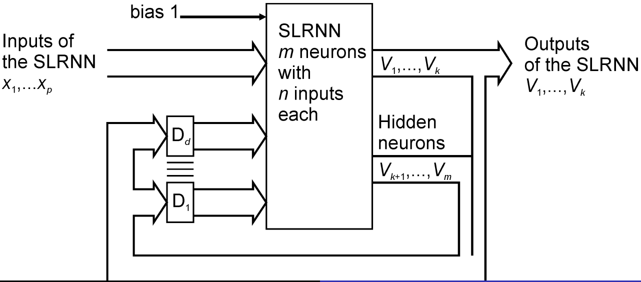
SLRNN - architecture

- 8 neurons, 18 inputs, 3 outputs (**event**, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust



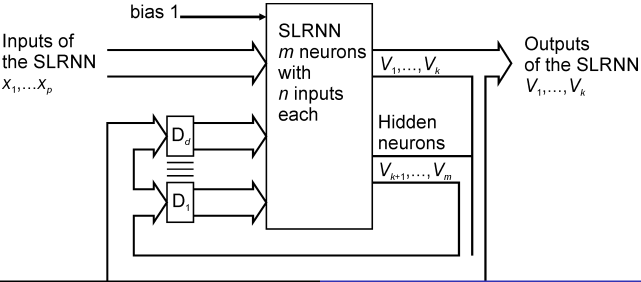
SLRNN - architecture

- 8 neurons, 18 inputs, 3 outputs (**event**, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust



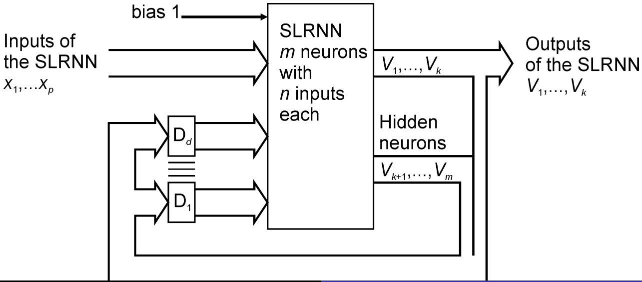
SLRNN - architecture

- 8 neurons, 18 inputs, 3 outputs (**event**, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust



SLRNN - architecture

- 8 neurons, 18 inputs, 3 outputs (**event**, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust

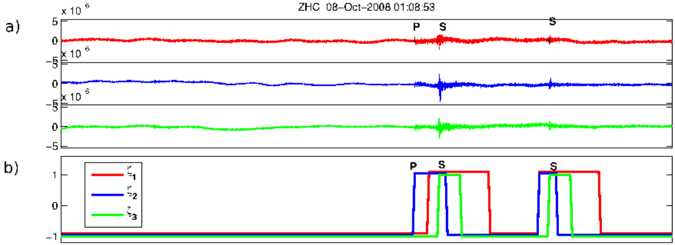


Outline

1. Artificial neural networks - basics
2. Data and method
   - WEBNET data
   - Single Layer Recurrent Neural Network
   - Training of the SLRNN
3. Training results
4. Application to Reykjanet

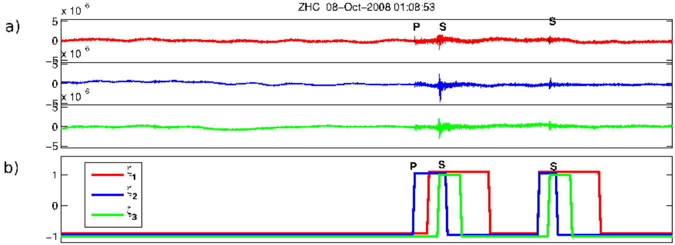
Training

- supervised learning: adjusting weights w_{ij} to get the best fit with desired output
- we define the desired outputs for training data



Training

- supervised learning: adjusting weights w_{ij} to get the best fit with desired output
- we define the desired outputs for training data



Training data

- earthquake swarm 2008 (events-positive examples) and year 2010 (disturbances-negative examples)
- events: different magnitudes, locations, focal mechanisms
- disturbances: quarry blasts, regional or teleseismic events, wind, storms

Training data

- earthquake swarm 2008 (events-positive examples) and year 2010 (disturbances-negative examples)
- events: different magnitudes, locations, focal mechanisms
- disturbances: quarry blasts, regional or teleseismic events, wind, storms

Training data

- earthquake swarm 2008 (events-positive examples) and year 2010 (disturbances-negative examples)
- events: different magnitudes, locations, focal mechanisms
- disturbances: quarry blasts, regional or teleseismic events, wind, storms

Sensitivity and specificity

- sensitivity (true positive rate TPR) - TPR=$\frac{TP}{TP+FN}$ =identified events / all events
- specificity (true negative rate TNR) - TNR=$\frac{TN}{TN+FP}$ =rejected disturbances / all disturbances
- ROC (Receiver operation characteristic) - a relation between the sensitivity and the specificity

Sensitivity and specificity

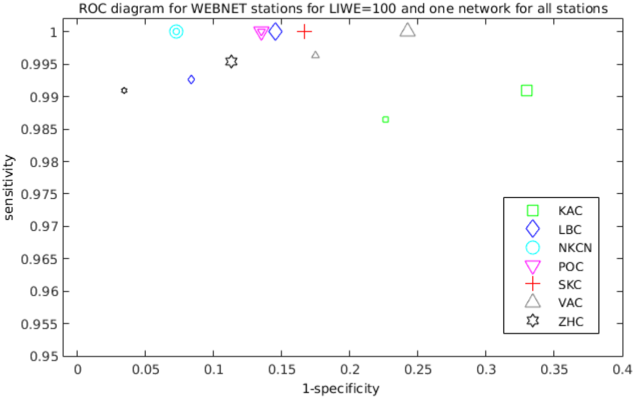
- sensitivity (true positive rate TPR) - TPR=$\frac{TP}{TP+FN}$ =identified events / all events
- specificity (true negative rate TNR) - TNR=$\frac{TN}{TN+FP}$ =rejected disturbances / all disturbances
- ROC (Receiver operation characteristic) - a relation between the sensitivity and the specificity

Sensitivity and specificity

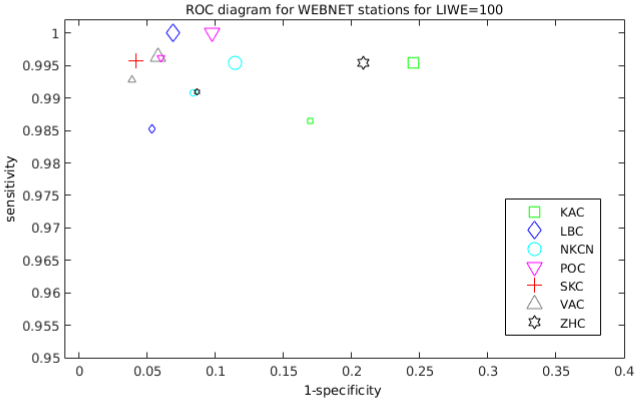
- sensitivity (true positive rate TPR) - TPR=$\frac{TP}{TP+FN}$ =identified events / all events
- specificity (true negative rate TNR) - TNR=$\frac{TN}{TN+FP}$ =rejected disturbances / all disturbances
- ROC (Receiver operation characteristic) - a relation between the sensitivity and the specificity

ROC diagram

joint training



ROC diagram
individual training



False detection

- blind test on data from 2011, single station detection only
- false detections are often weak events
- but many detections could not be verified

False detection

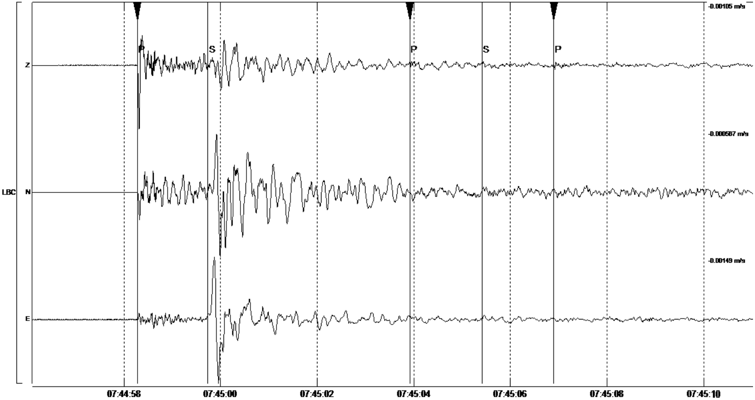
- blind test on data from 2011, single station detection only
- false detections are often weak events
- but many detections could not be verified

False detection

- blind test on data from 2011, single station detection only
- false detections are often weak events
- but many detections could not be verified

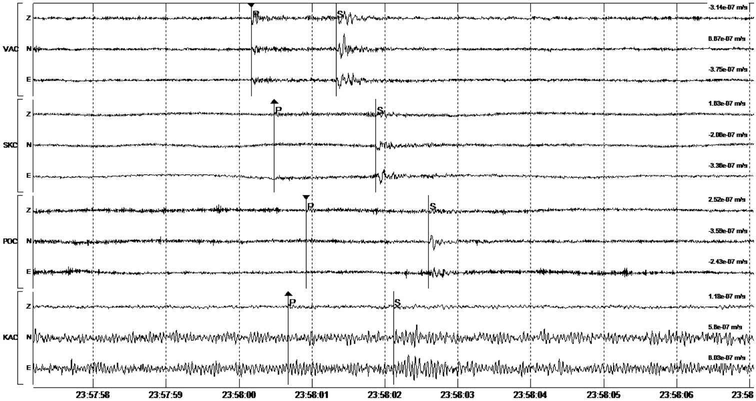
Undetected events - examples

events $M_L = 2,3$ and $M_L = 2,2$ hidden in the coda of $M_L = 3,8$



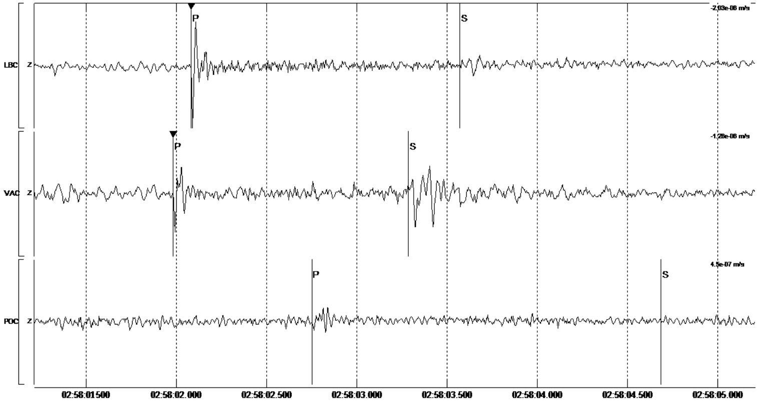
Undetected events - examples

undetected event $M_L = -0,3$ on station with high noise



Undetected events - examples

missing detection of event $M_L = 0,2$ on station with low P- and S-wave amplitudes



Training results

- SLRNN architecture is suitable for local event detection
- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
  - reject false detections
  - eliminate undetected events

Training results

- SLRNN architecture is suitable for local event detection
- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
  - reject false detections
  - eliminate undetected events

Training results

- SLRNN architecture is suitable for local event detection
- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
    - reject false detections
    - eliminate undetected events

Training results

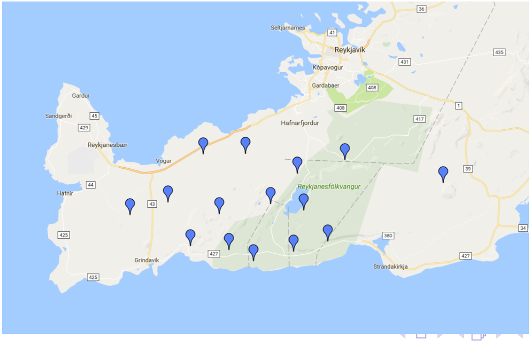
- SLRNN architecture is suitable for local event detection
- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
  - reject false detections
  - eliminate undetected events

Training results

- SLRNN architecture is suitable for local event detection
- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
  - reject false detections
  - eliminate undetected events

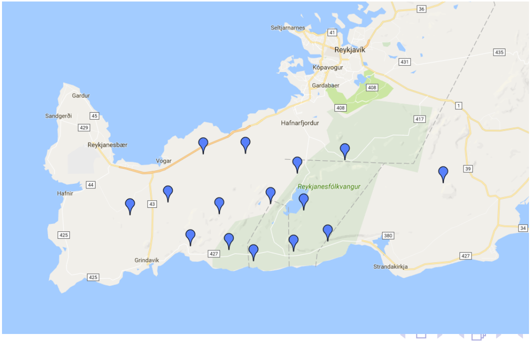
Reykjanet network

- southwest Iceland, Reykjanes peninsula
- 15 off-line broadband stations
- network configuration, number of stations, swarm activity - similar to WB



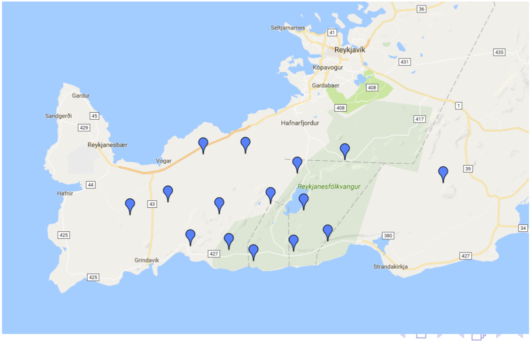
Reykjanet network

- southwest Iceland, Reykjanes peninsula
- 15 off-line broadband stations
- network configuration, number of stations, swarm activity - similar to WB



Reykjanet network

- southwest Iceland, Reykjanes peninsula
- 15 off-line broadband stations
- network configuration, number of stations, swarm activity - similar to WB



Neural network

- the best network of joint training for WEBNET
- coincidence implemented
- detection on at least 6 stations in a time window (0,8s) required to define an EVENT

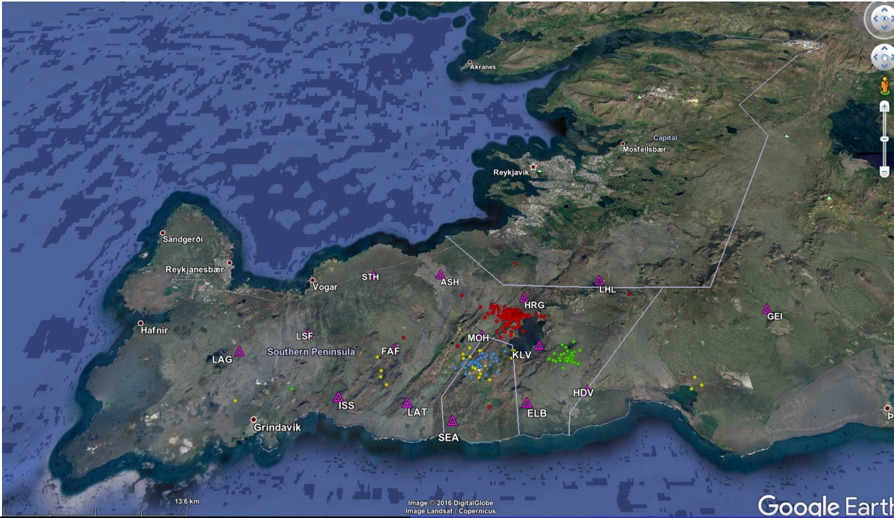
Neural network

- the best network of joint training for WEBNET
- coincidence implemented
- detection on at least 6 stations in a time window (0,8s) required to define an EVENT

Neural network

- the best network of joint training for WEBNET
- coincidence implemented
- detection on at least 6 stations in a time window (0,8s) required to define an EVENT

Selected data: 4 swarms



Catalogs

- SIL - IMO catalog - manually revised automatic locations Icelandic network
- Antelope - automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN - automatic locations by PePiN (T. Fischer)
- ANN - event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude $M_L = 2{,}3$

Catalogs

- SIL - IMO catalog - manually revised automatic locations Icelandic network
- Antelope - automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN - automatic locations by PePiN (T. Fischer)
- ANN - event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude $M_L = 2{,}3$

Catalogs

- SIL - IMO catalog - manually revised automatic locations Icelandic network
- Antelope - automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN - automatic locations by PePiN (T. Fischer)
- ANN - event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude $M_L = 2,3$

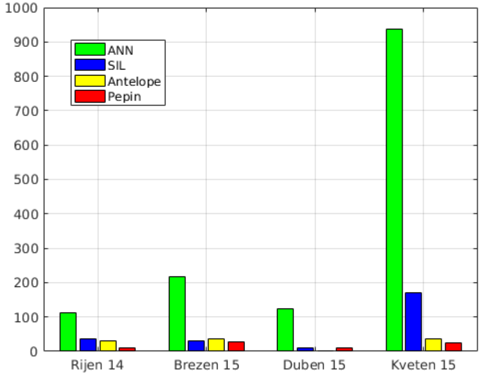
Catalogs

- SIL - IMO catalog - manually revised automatic locations Icelandic network
- Antelope - automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN - automatic locations by PePiN (T. Fischer)
- ANN - event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude $M_L = 2,3$

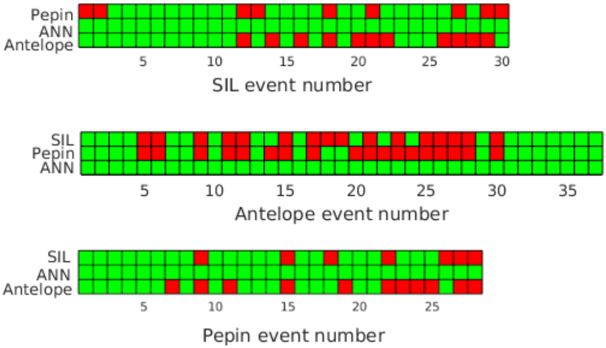
Catalogs

- SIL - IMO catalog - manually revised automatic locations Icelandic network
- Antelope - automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN - automatic locations by PePiN (T. Fischer)
- ANN - event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude $M_L = 2,3$

Number of events

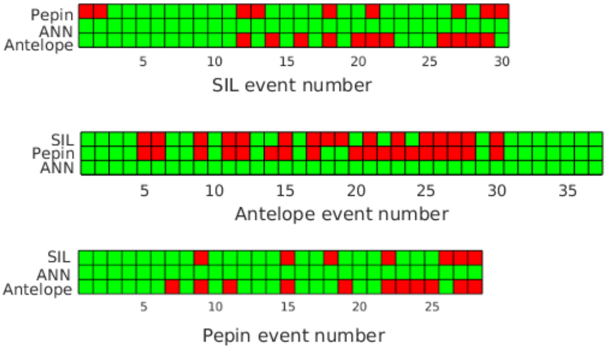


Comparison of individual events - March 2015



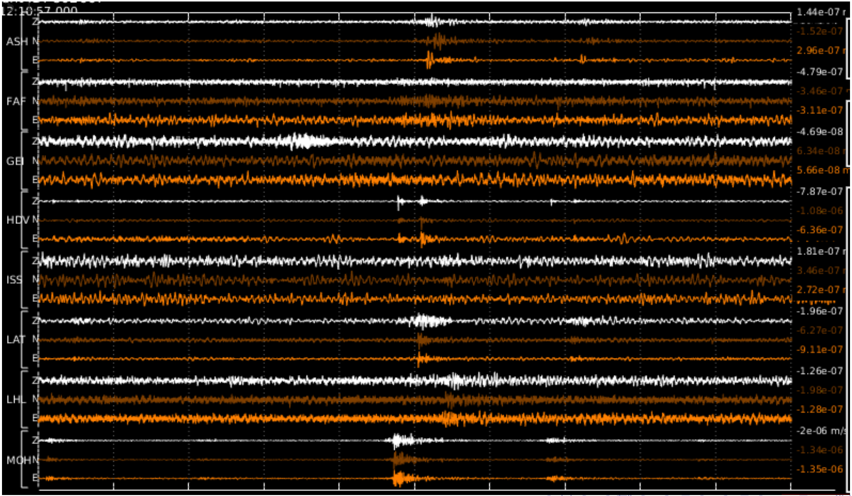
- Pepin and SIL sorted by the magnitude
- Antelope without magnitudes, sorted in time

Comparison of individual events - March 2015



- Pepin and SIL sorted by the magnitude
- Antelope without magnitudes, sorted in time

Smallest events



Conclusion

- SLRNN is fast and efficient algorithm for event detection
- in case of consistent readings of P and S phases - individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes

Conclusion

- SLRNN is fast and efficient algorithm for event detection
- in case of consistent readings of P and S phases - individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes

Conclusion

- SLRNN is fast and efficient algorithm for event detection
- in case of consistent readings of P and S phases - individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes

Conclusion

- SLRNN is fast and efficient algorithm for event detection
- in case of consistent readings of P and S phases - individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes

Conclusion

- SLRNN is fast and efficient algorithm for event detection
- in case of consistent readings of P and S phases - individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes

.

Thank you for your attention !

Sincere thanks to members of Webnet group for providing data, and to CzechGeo/EPOS for supporting our networks