GJØVIK UNIVERSITY COLLEGE

Data-stream Mining for Rule-based Access Control

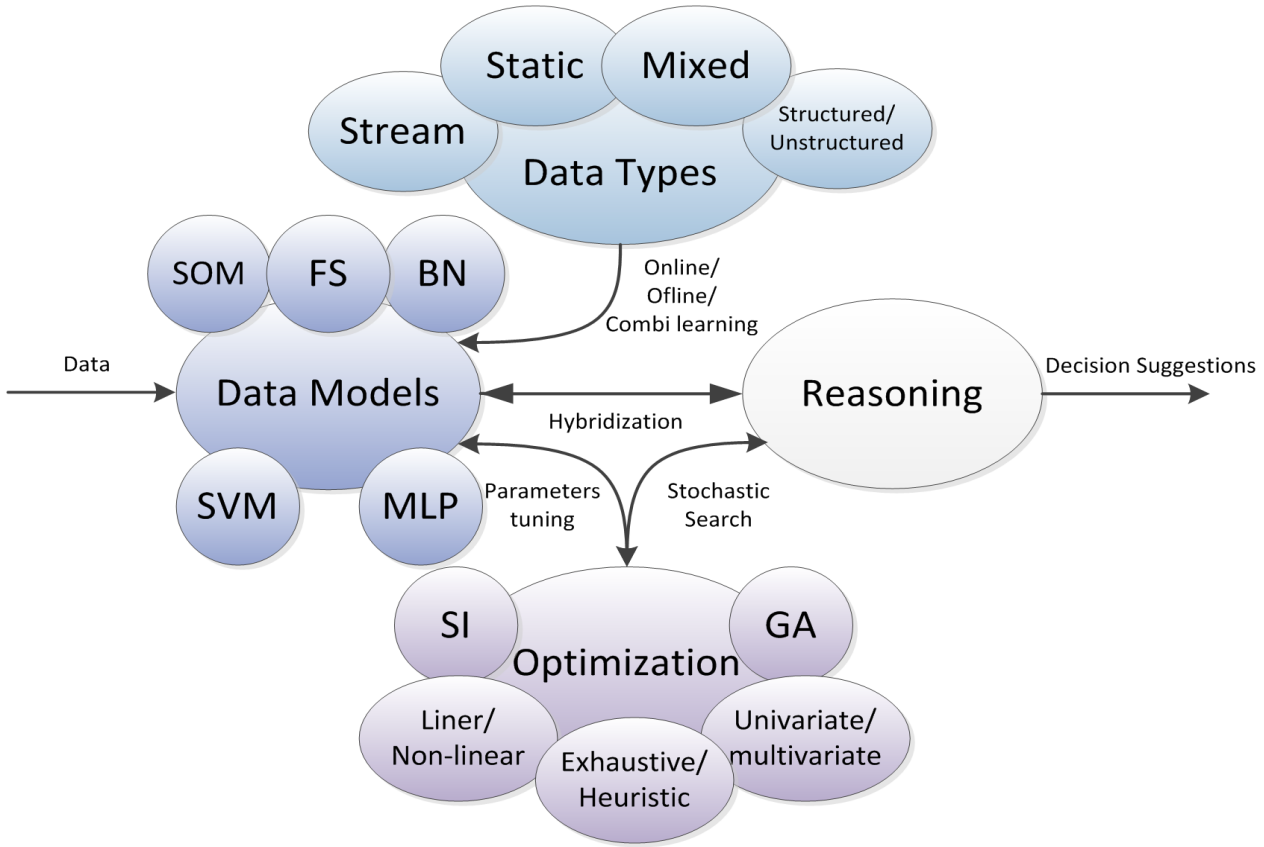
Andrii Shalaginov, andrii.shalaginov@hig.no

13th of October 2014

COINS PhD seminar

OVERALL PICTURE OF PHD

- Work towards Data-driven Reasoning for Information Security and Forensics using Hybrid, Computational Intelligence.



WHY DO WE NEED ADVANCED ML TECHNIQUES?

Obstacles in Digital Forensics:

- 4V of Big Data: Volume, Velocity, Variety, Veracity

- Sequential off-line methods became less reliable and no more efficient

- Decision time time is important

- *"Information fusion"* in the models is needed

However the ways to approach:

+ Almost no limitations in computational power

+ Therefore possibilities to Hybridized computational methods

+ Massive parallel optimization

TYPES OF ML-TRAINING BASED ON THE HISTORICAL DATA

Methods	On-line	Off-line
Pros	- Fast convergence - Short re-training time - Flexibility - Concept drift	- Easy to train - Lower error - Better perception of non-linearity
Cons	- Unsteady for random changes - Over/Under fitting	- Long re-training time - Model is hard to change
Training data availability	- *Low*, each sample is processed once or at most once	- *High*, all data samples are available for re-training
Preferred training	- Single-step - Mini-batch	- Batch

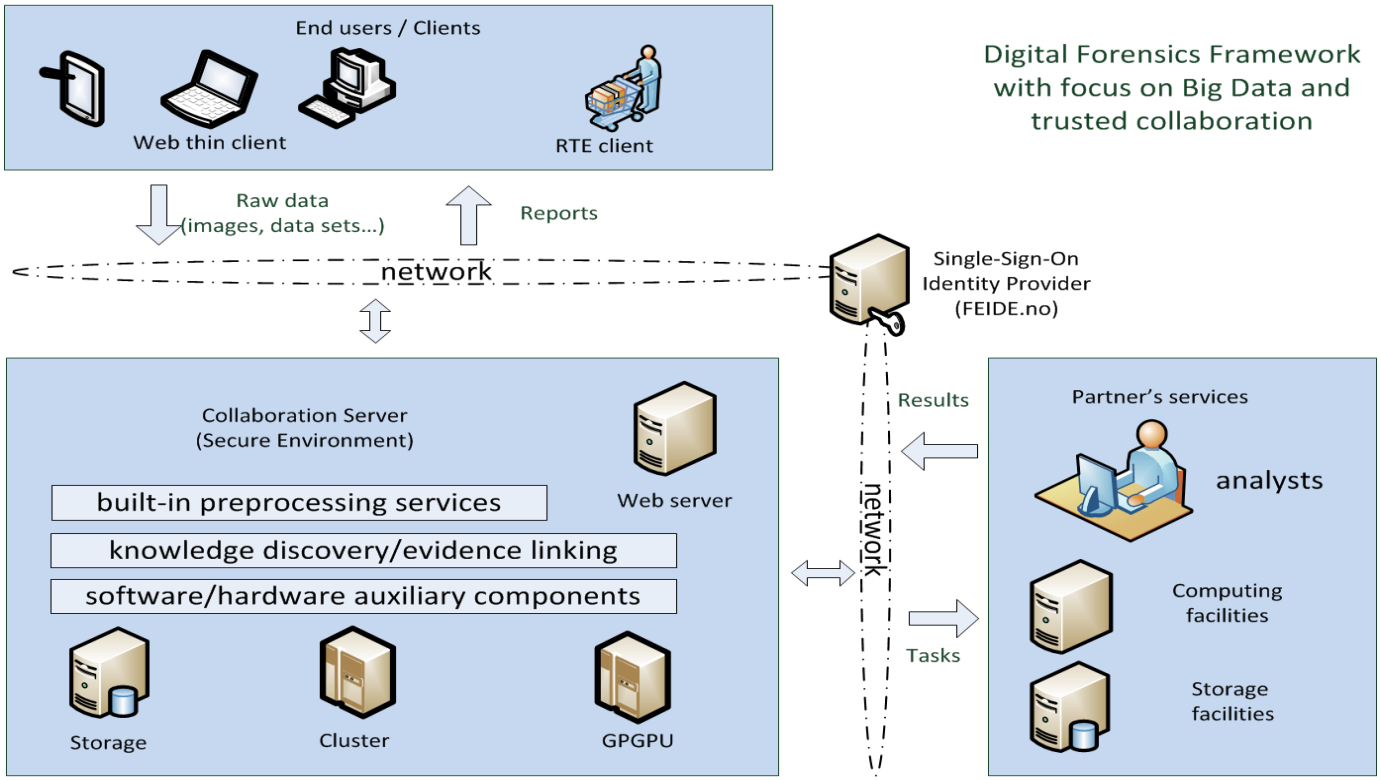
DATA STREAMS MINING

- All computers are interconnected -> information flows
- Besides the network traffic we can consider **Access Control** as a sequence of the <subject, action, object>
- **Hard to control** across networks, organizations, applications since employees are moving, access to similar users/resources
- Off-line ML methods are not applicable/reliable
- Incident response challenges (if the false positive rate is too high)
- Constant re-training is preferred
- AlgoSec developed firewall rules tuning based on traffic [1]
- Identity and Access Management based on the Risk Value [2]:

2.http://www.mnemonic.no/Global/Vedlegg/Frokost%202014%20september/AlgoSec%20Suite%20Overview%202014-06.pdf
1. http://msdn.microsoft.com/en-us/library/cc836395.aspx

COLLABORATION FRAMEWORK FOR DIGITAL FORENSICS

It becomes relevant to have an access control in the computing infrastructure. Possible example:



EXISTING APPROACHES

- **Traditional methods:**
- Discretionary access control (DAC)
- Mandatory access control (MAC)
- Role-based access control (RBAC)
- Attribute-based access control (ABAC)

- **Machine Learning-based methods**
- *"The heuristics include a historical record of access control decisions and machine learning. This means that a RAdAC system will use previous decisions as one input when determining whether access will be granted to a resource in the future."* NIST
- Consider multiple features (resource attributes, user profile, etc)
- On-line learning system (high availability, no need for full retrain)
- Consider historical data for the decision making

KAGGLE: AMAZON.COM - EMPLOYEE ACCESS CHALLENGE

- Historical numerical data collected from 2010 & 2011
- Employees are manually allowed or denied access to resources over time.

Column Descriptions

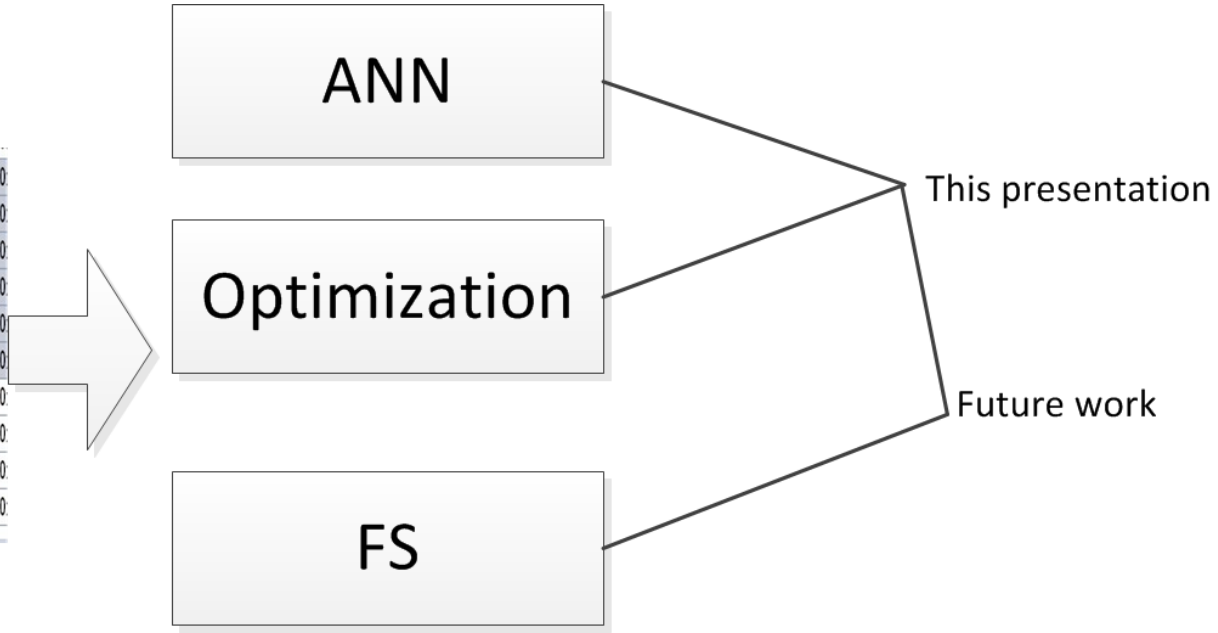
Column Name	Description
ACTION	ACTION is 1 if the resource was approved, 0 if the resource was not
RESOURCE	An ID for each resource
MGR_ID	The EMPLOYEE ID of the manager of the current EMPLOYEE ID record; an employee may have only one manager at a time
ROLE_ROLLUP_1	Company role grouping category id 1 (e.g. US Engineering)
ROLE_ROLLUP_2	Company role grouping category id 2 (e.g. US Retail)
ROLE_DEPTNAME	Company role department description (e.g. Retail)
ROLE_TITLE	Company role business title description (e.g. Senior Engineering Retail Manager)
ROLE_FAMILY_DESC	Company role family extended description (e.g. Retail Manager, Software Engineering)
ROLE_FAMILY	Company role family description (e.g. Retail Manager)
ROLE_CODE	Company role code; this code is unique to each role (e.g. Manager)

https://www.kaggle.com/c/amazon-employee-access-challenge

EXPERIMENT DESIGN

Data Stream

242	0000003c:0000006a:0001	LOP_BEGIN_XACT	LCX_NULL	0000:000004b4
243	0000003c:0000006a:0002	LOP_DELETE_ROWS	LCX_MARK_AS_GHOST	0000:000004b4
244	0000003c:0000006a:0003	LOP_SET_BITS	LCX_PFS	0000:00000000
245	0000003c:0000006a:0004	LOP_INSERT_ROWS	LCX_CLUSTERED	0000:000004b4
246	0000003c:0000006a:0005	LOP_SET_BITS	LCX_PFS	0000:00000000
247	0000003c:0000006a:0006	LOP_ABORT_XACT	LCX_NULL	0000:000004b4
248	0000003c:0000006a:0007	LOP_BEGIN_XACT	LCX_NULL	0000:000004b5
249	0000003c:0000006a:0008	LOP_DELETE_ROWS	LCX_HEAP	0000:000004b5
250	0000003c:0000006a:0009	LOP_INSERT_ROWS	LCX_HEAP	0000:000004b5
251	0000003c:0000006a:000a	LOP_ABORT_XACT	LCX_NULL	0000:000004b5

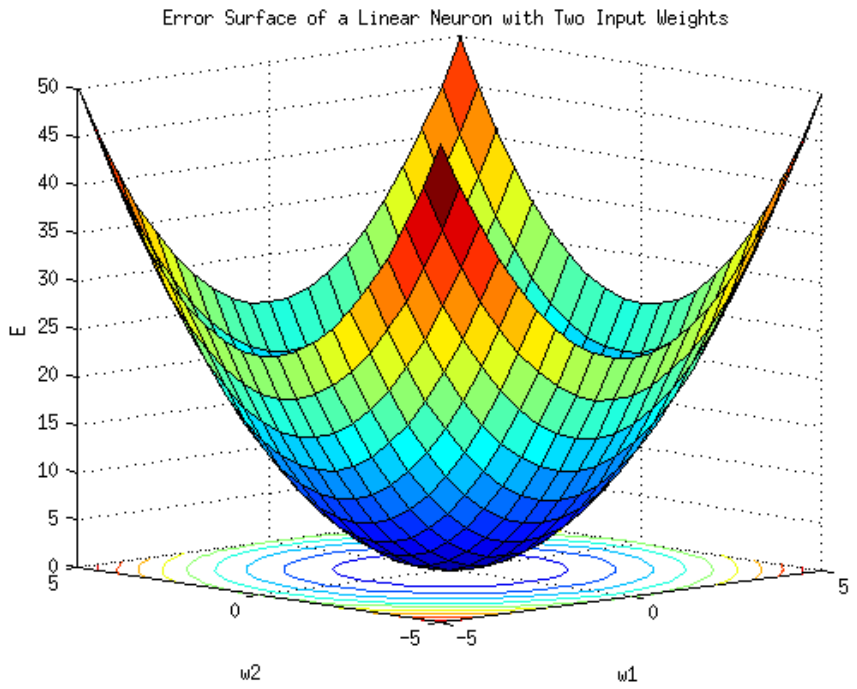


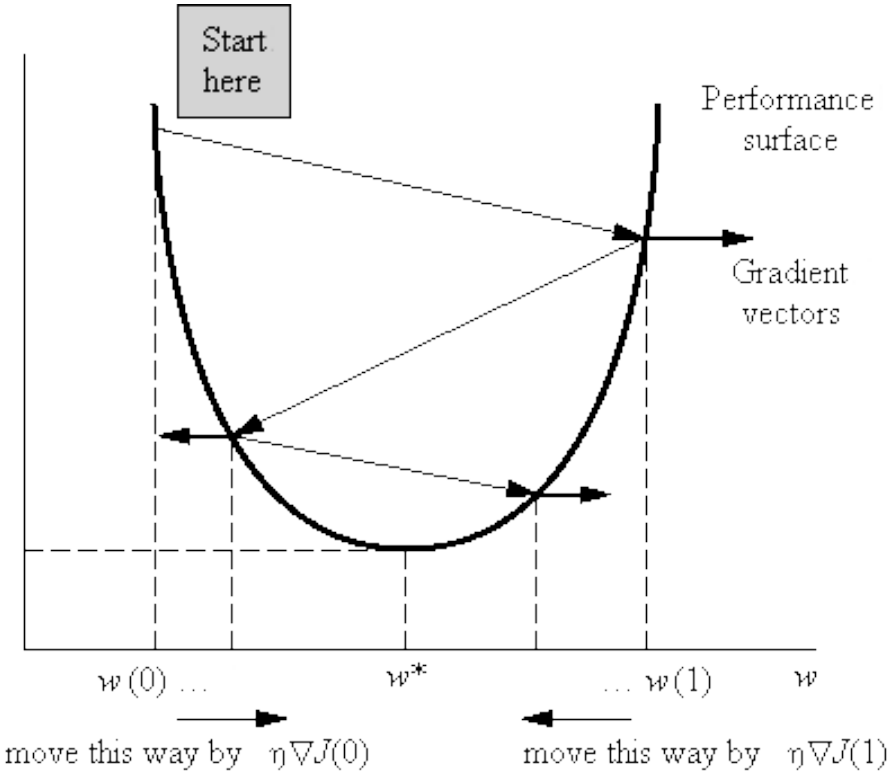
NEURAL NETWORK IN THE ACCESS CONTROL SCENARIO

- **Why?**
- Non-linear model based on the historical data
- Able to differentiate complex patterns
- Automated parameters tuning in the model

- **Cons with respect to data streams mining**
- Significant time for sequential learning with large dimensionality
- Requires availability of the multiple data for re-training
- The training done using a multiple iterative process
- **Learning rate** has to be chosen carefully to be sub-optimal/optimal

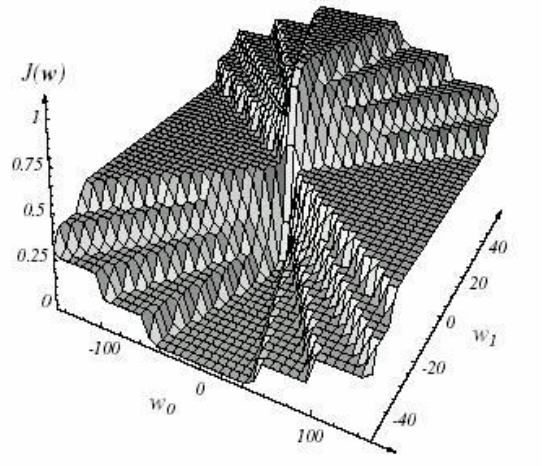
THE LEARNING RATE & ERROR FUNCTION SURFACE

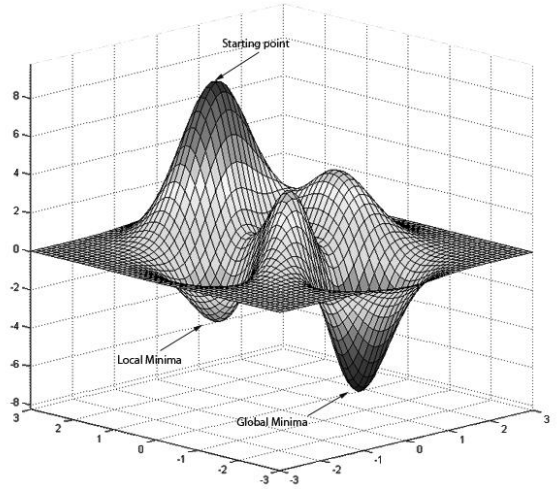


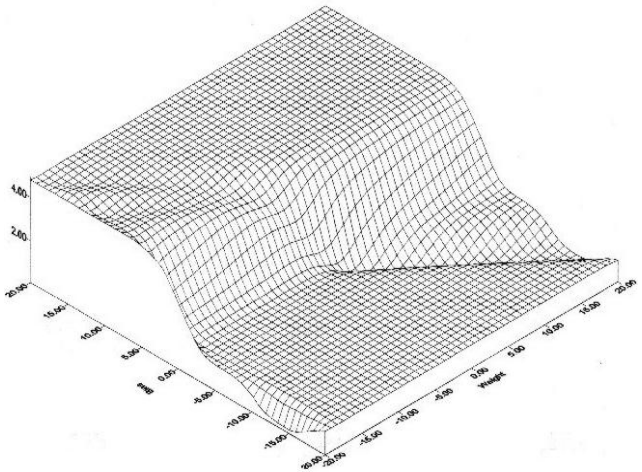


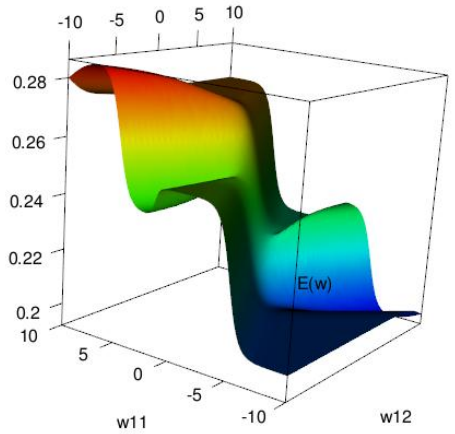
http://www.nd.com/NSBook/NEURAL%20AND%20ADAPTIVE%20SYSTEMS14_Adaptive_Linear_Systems.html

IN FACT, THE REAL-WORLD ANN ERROR FUNCTIONS ARE:







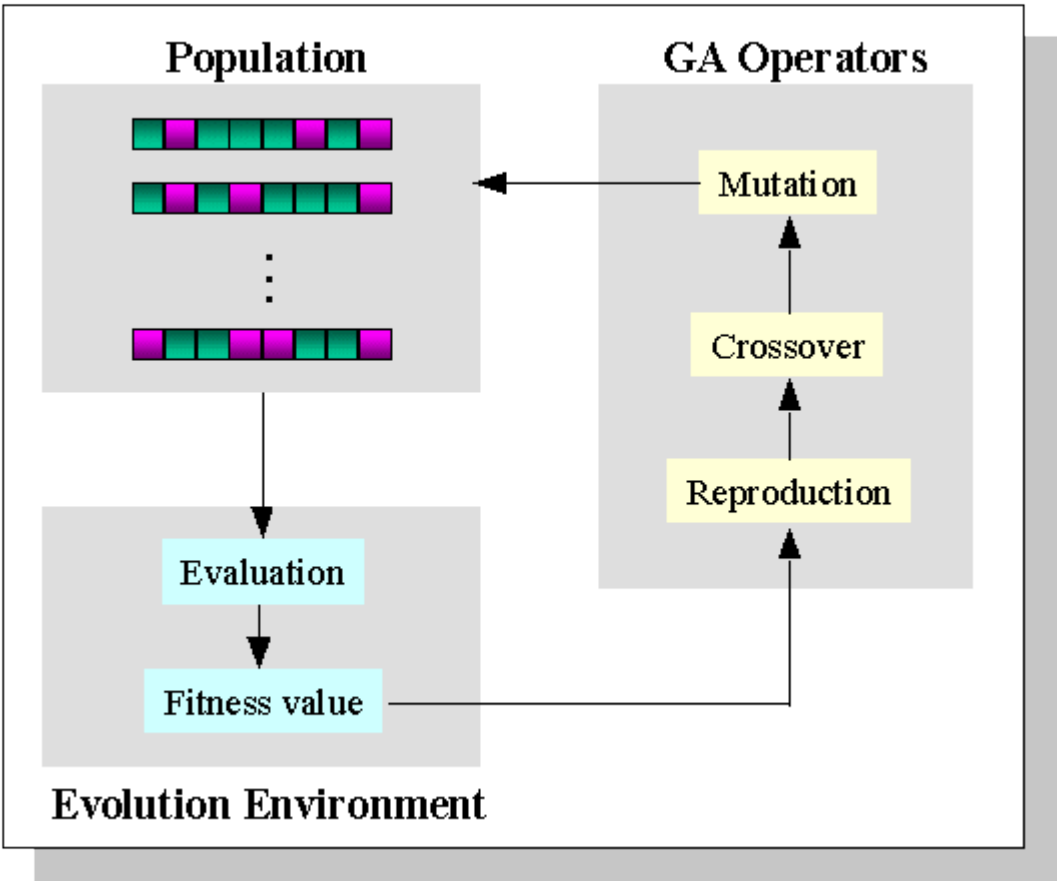


http://www.webpages.ttu.edu/dleverin/neural_network/neural_networks.html

http://www.codeproject.com/Articles/175777/Financial-predictor-via-neural-network

https://www.byclb.com/TR/Tutorials/neural_networks/ch10_1.htm

GENETIC ALGORITHM



function MUTATION($prob_{mutation}$,α)

$d \leftarrow random(0,1)$ (generation of a real number)

$\alpha_{mutated} \leftarrow \alpha_{mutate} \pm d$

return $\alpha_{mutated}$

end function

function CROSSOVER($prob_{crossover}$,$\alpha 1$,$\alpha 2$)

$d \leftarrow random(0,1)$ (generation of a real number)

$offspring1 \leftarrow d \cdot y_i + (1-d) \cdot x_i$

$offspring2 \leftarrow d \cdot x_i + (1-d) \cdot y_i$

return $\alpha_{off1}, \alpha_{off2}$

end function

function SELECTION(α)

$Fit \leftarrow E(w)|_{\alpha}$

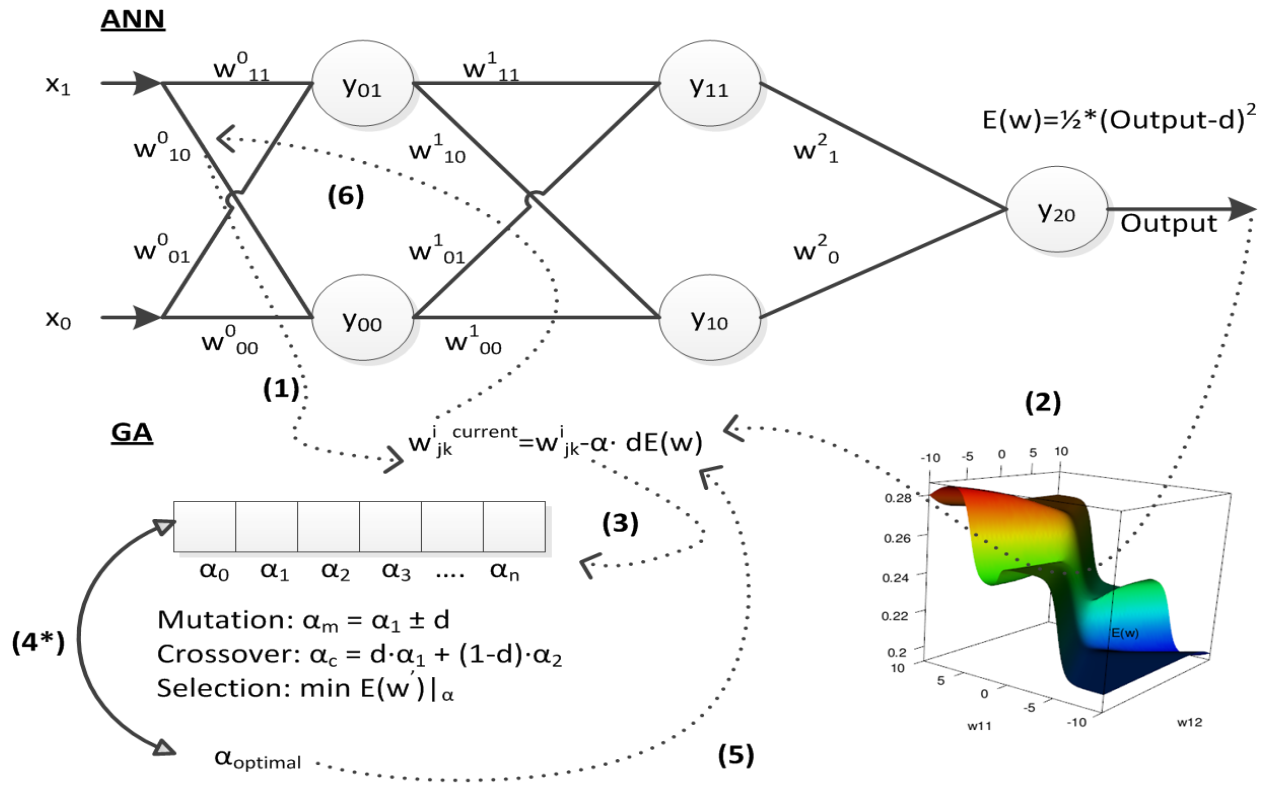
return $\alpha_{optimal}$

end function

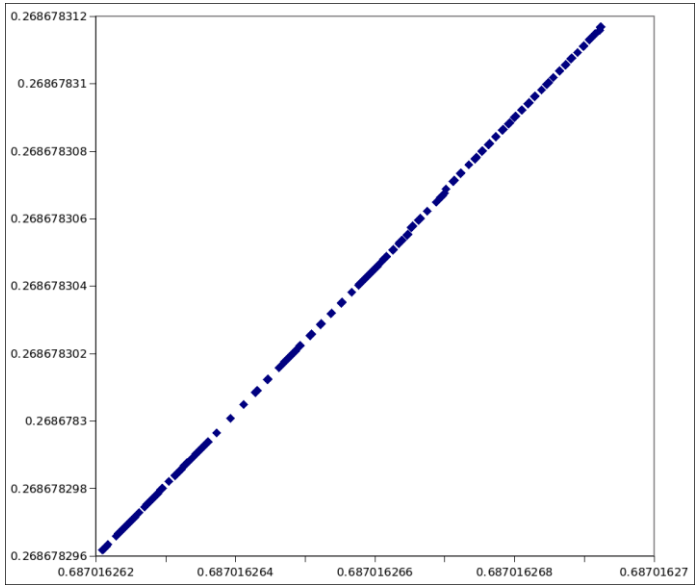
 http://www.ewh.ieee.org/soc/es/May2001/14/Begin.htm

PROPOSED METHOD - GA FOR LEARNING RATE

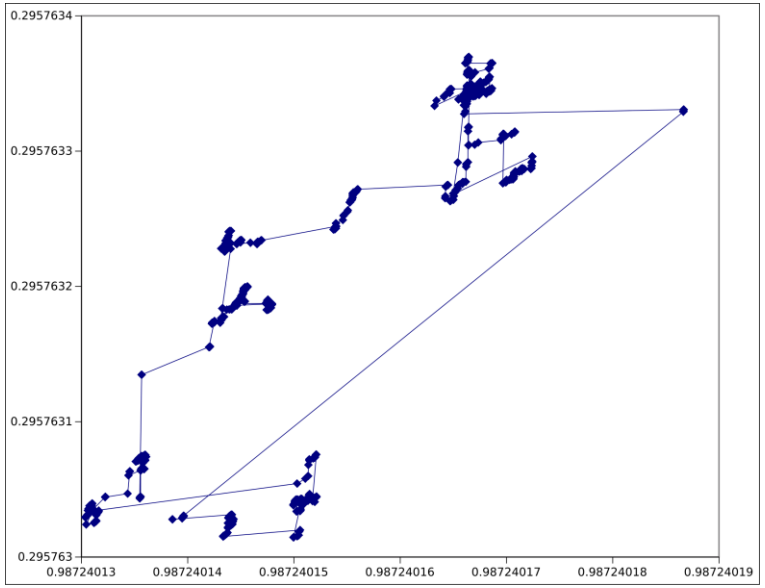
- Individual learning rate reduces the overall error function E(w)
- Allow a single step on-line incremental training, since every data sample is available during short period of time



WEIGHTS SEARCH AREA COVERAGE BY ANN-GA

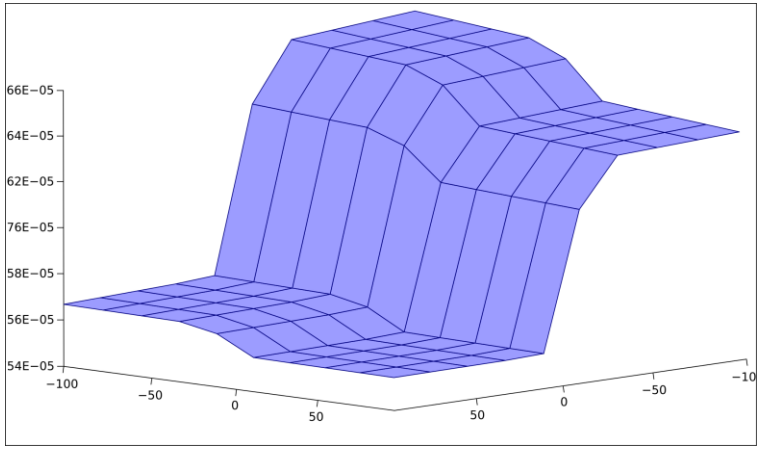


ANN training, weights w_{13}, w_{19}



ANN-GA training, weights w_{13}, w_{19}

- Stopping criteria matters
- Steepness is not high
- Learning rate -> towards MIN
- Convergence optima



Error function surface $E(w_{13}, w_{19})$

USE CASES & PERFORMANCE METRICS

Case 1. "Experiment on the static data set". The batch learning with results compared to *Weka* and *RapidMiner*. (next slide Table 1)
Case 2. "Experiment on the data stream". First the single-step MLP is trained with 100 samples. Then, the stream of 100 samples is classifying sequentially while the MLP is constantly trained. (next slide Table 2)

Performance metrics:

MAE (Mean absolute error , how close forecasts or predictions are to the eventual outcomes)

$$MAE = \frac{1}{n} \cdot \sum |y_i - d_i|$$

RMSE (Root mean squared error, differences between values predicted by a model or an estimator and the values actually observed)

$$RMSE = \sqrt{\frac{1}{n} \cdot \sum (y_i - d_i)^2}$$

RRSE (Root relative squared error)

$$RRSE = \sqrt{\frac{\sum (y_i - d_i)^2}{\sum (y_i - \bar{d})^2}}$$

PRELIMINARY RESULTS & DISCUSSION

- GA gives a slackness within the learning rate range
- Learning can be done in parallel with single step
- Robustness against randomness
- Good performance on the Amazon Kaggle Challenge

	MAE	**RMSE**	**RRSE**
MLP impl.	0.061161	0.140322	100.849277%
MLP impl. + GA	0.054920	0.142849	102.665093%
Weka	0.061	0.1497	107.5543%
RapidMiner	0.059	0.151	108.70%

Table 1: Performance comparison between optimized method and corresponding implementations in Weka and RapidMiner.

	MAE	**RMSE**	**RRSE**
Single-step MLP (with GA)	0.002004	0.020041	96.874085%
Single-step MLP (without GA)	0.002541	0.025417	122.859120%

Table 2: Performance comparison in online incremental learning using optimized and non-optimized techniques

Thank you for the attention!

Any questions? Comments?
Mail andrii.shalaginov@hig.no