Faculty of Agricultural and Nutritional Science

Christian-Albrechts-University Kiel

Institute of Animal Breeding and Husbandry

Artificial neural networks for genome-enabled prediction of milk traits in Simmental cattle

Anita Ehret[1],

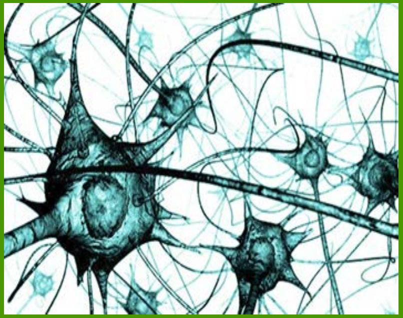
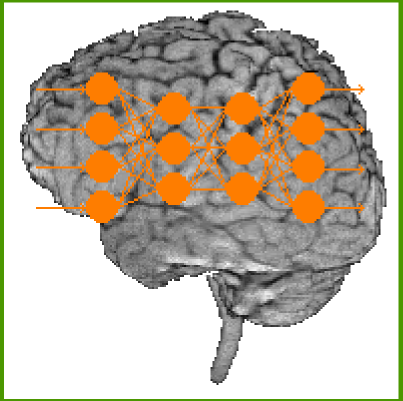
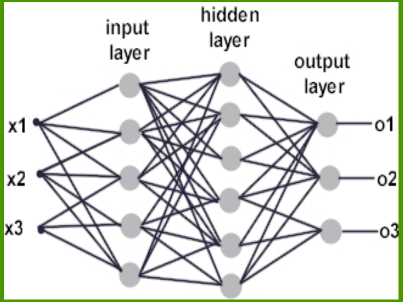
David Hochstuhl[2] and Georg Thaller[1]

[1] *Institute of Animal Breeding and Husbandry, Christian-Albrechts-University, Olshausenstr. 40, 24098 Kiel, Germany*

[2]*Institute for Theoretical Physics and Astrophysics, Christian-Albrechts-University, Leibnizstr. 15, 24098 Kiel, Germany*

64th Annual EAAP Meeting Nantes, France

August 26th to 30th , 2013

Introduction

Application of artificial neural networks (ANNs) to genome-enabled predictions of complex traits

- Massive genomic and phenotypic data is available
- Complex traits may be affected by various gene interactions
  - ✓ e.g. dominance, epistasis
- Methods used in genomic selection field are mostly linear

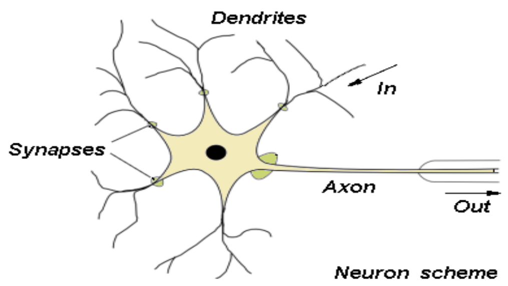
$$y_i = \mu + \sum_{j=1}^{p} x_{ij}\beta_j + e_i$$

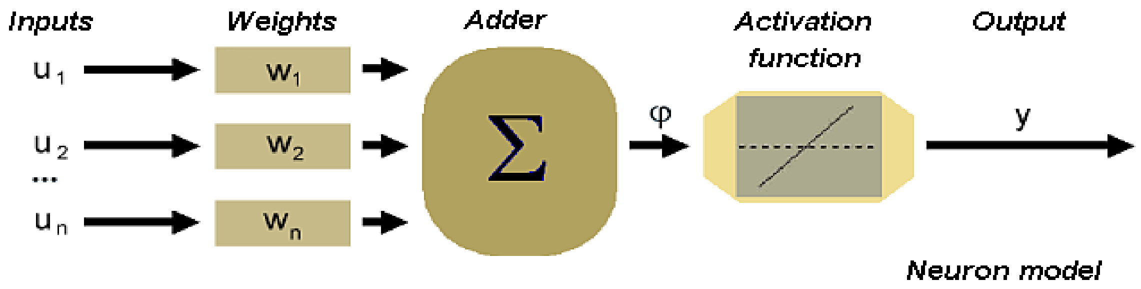
- It may be possible to increase accuracy using more general models
  - e.g. non-parametric models, machine learning methods

Introduction – Artificial neural networks

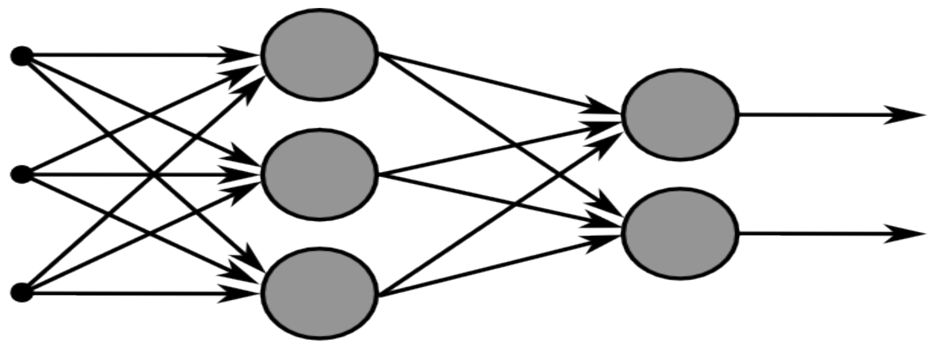
What are artificial neural networks?

- An extremely simplified model of the human brain





- To mimic physical aspects of the human brain these artificial neurons are organized in networks with several layers



Inputs Hidden layer Output layer

Introduction – Artificial neural networks

$$y_i = \mu + f(\mathbf{x}_i) + e_i$$

Any complex continuous function can be exactly represented in the following form → **Kolmogorov's theorem**

$$f(\mathbf{x}_i) = f(x_{i1},\ldots,x_{ip}) = \sum_{q=1}^{2p+1} g\left(\sum_{r=1}^{p} \lambda_r h_q(x_{ir})\right)$$

Linear or nonlinear transformation

Weights

Linear or nonlinear transformation of inputs

Introduction – Artificial neural networks

$$y_i = \mu + f(\mathbf{x}_i) + e_i$$

Any complex continuous function can be exactly represented in the following form → **Kolmogorov's theorem**

$$f(\mathbf{x}_i) = f(x_{i1},\ldots,x_{ip}) = \sum_{q=1}^{2p+1} g\left(\sum_{r=1}^{p} \lambda_r h_q(x_{ir})\right)$$

Linear or nonlinear transformation

Weights

Linear or nonlinear transformation of inputs

- ➢ This theorem can be completely represented as an ANN
  - ✓ ANNs act as general function approximators
  - ✓ ability to capture underlying functions between input and outputs without explicitly defining a fixed model
  - ✓ not limited to linear problems

Introduction – Artificial neural networks

Prediction can be done in two steps:

1. Inputs transformed non linearly in the hidden layer
2. Outputs from the hidden layer are combined to obtain predictions

$$y_i = g(b + \underbrace{\sum_{t=1}^{s}}_{\text{Combine output from hidden layer}} w_{2t} \underbrace{f_t(a_t + \sum_{j=1}^{p} w_{1j}^{[t]} x_{ij})}_{\text{Output from hidden layer}}) + \varepsilon_i$$

!!! To obtain predictions a so called training phase is needed

Introduction – Artificial neural networks

Prediction can be done in two steps:

1. Inputs transformed non linearly in the hidden layer
2. Outputs from the hidden layer are combined to obtain predictions

Aim

Assessing the influence of the network architecture, the training parameters and different genomic covariates as well as the phenotypes on the predictive performance of an ANN

Combine output from hidden layer

Output from hidden layer

!!! To obtain predictions a so called training phase is needed

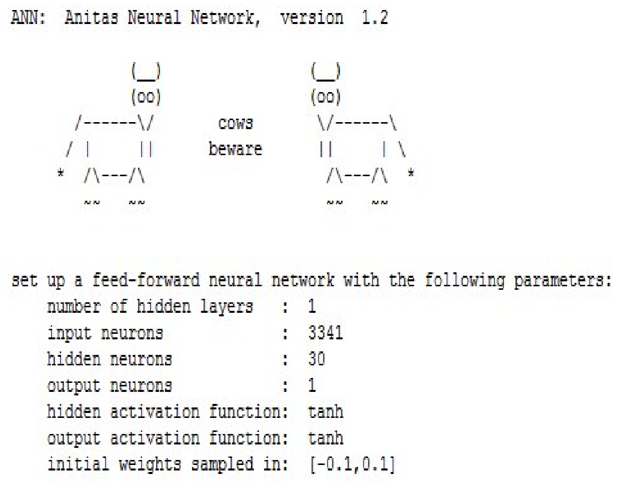
Materials and methods – Data basis

Three cattle data sets

Data set	Animals in analysis	Number of markers after quality control	Type of phenotype records
Simmental cattle bulls	3,341	39,344 SNPs	DYD of milk traits
Holstein Friesian bulls	2,303	41,995 SNPs	DRP of milk traits
Holstein Friesian dams	777	41,718 SNPs	YD of milk traits

Materials and methods – Network design

- ➢ Single Hidden Layer Feed Forward ANN
- ➢ 1 - 20 neurons in the hidden layer
- ➢ Non-linear activation function in the hidden layer
- ➢ Supervised learning rule:
  - ✓ Back-propagation algorithm with early stopping
- ➢ Program written in C++



```
ANN:  Anitas Neural Network,  version  1.2

             (__)                (__)
             (oo)                (oo)
       /------\/        cows      \/------\
      / |     ||       beware     ||      | \
     *  /\---/\                    /\---/\  *
        ~~   ~~                    ~~   ~~

set up a feed-forward neural network with the following parameters:
    number of hidden layers   :  1
    input neurons             :  3341
    hidden neurons            :  30
    output neurons            :  1
    hidden activation function:  tanh
    output activation function:  tanh
    initial weights sampled in:  [-0.1,0.1]
```

Assessing the predictive ability via cross-validation

- ✓ 5-fold (20 random repetitions)
- ✓ average Pearson’s correlation coefficient between predicted and true phenotypic value in the testing sets

Materials and methods – Fitting the network

Target:

DYD, DRP, YD of three milk traits (milk yield, protein yield, fat yield)

Genomic information:

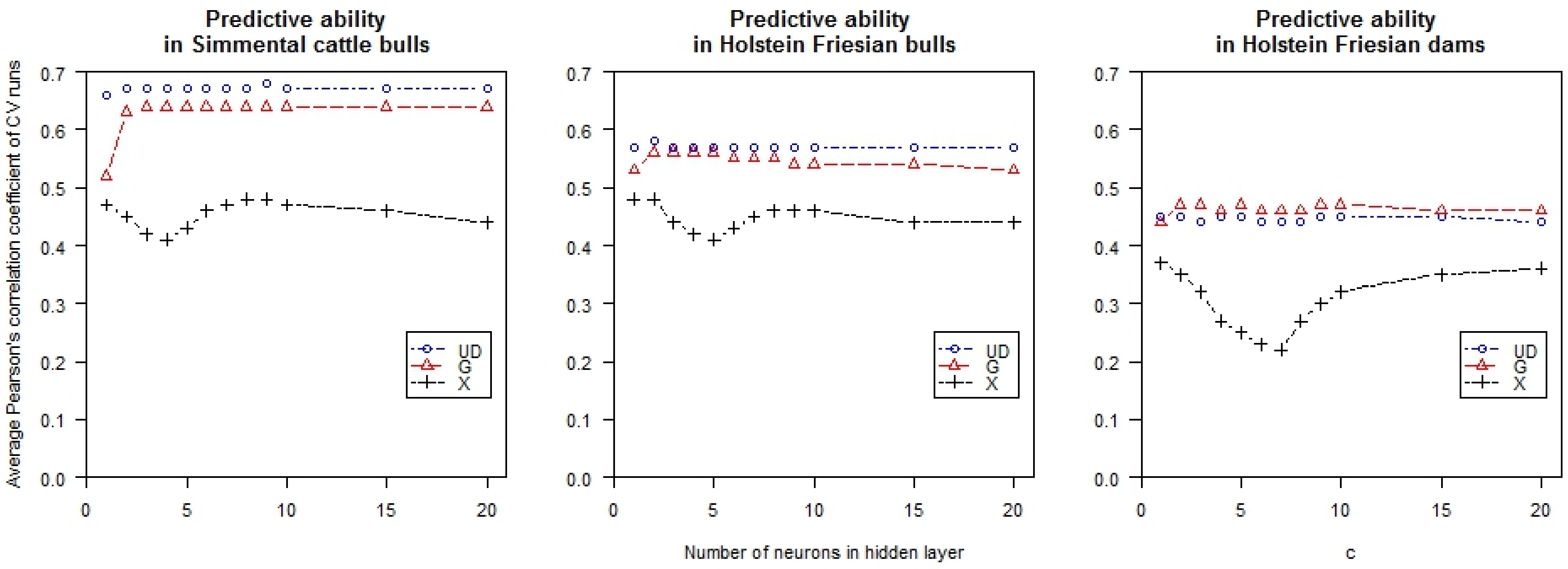
I.	Raw marker genotypes (SNPs)	→	**X**
II.	Genomic relationship matrix	→	**G**
III.	Principle component score matrix	→	**UD**

Feature scaling:

$$y_i^* = \frac{y_i - \mu_y}{Max_y}$$

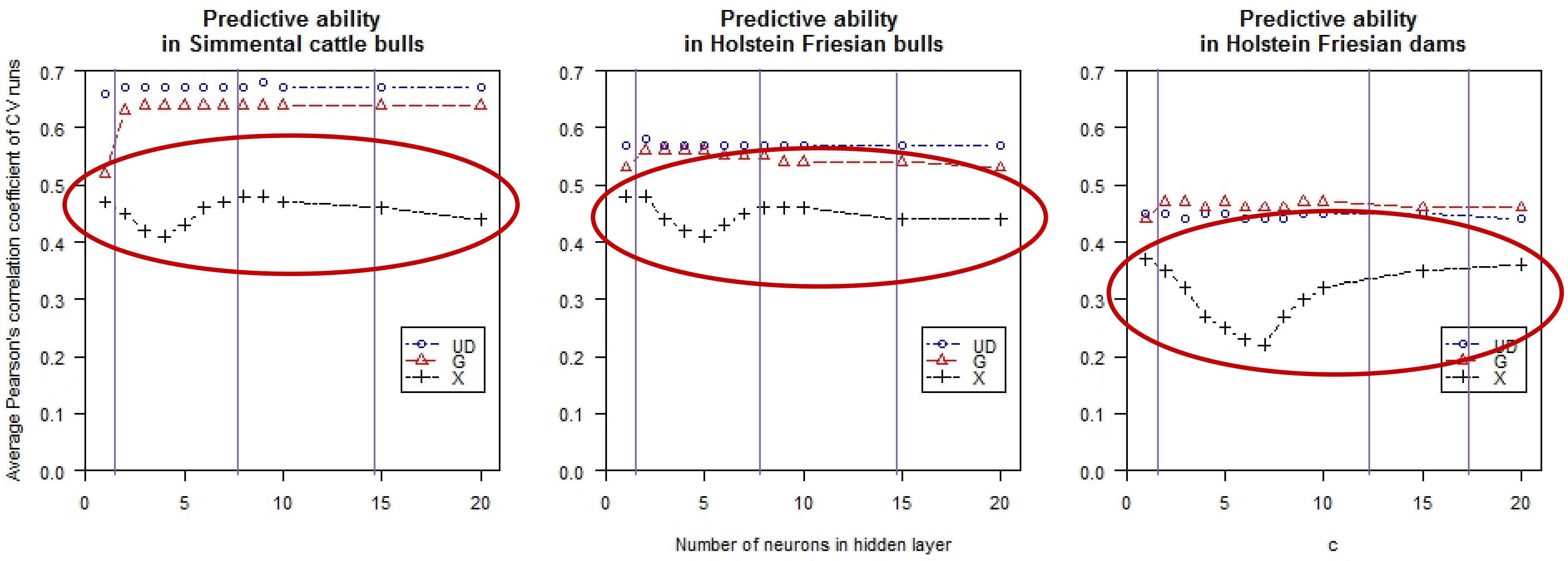
Results – Predicting milk yield

- 100 individual cross-validation runs
- Impact of
  - Network architecture
  - Genomic input
  - Phenotype



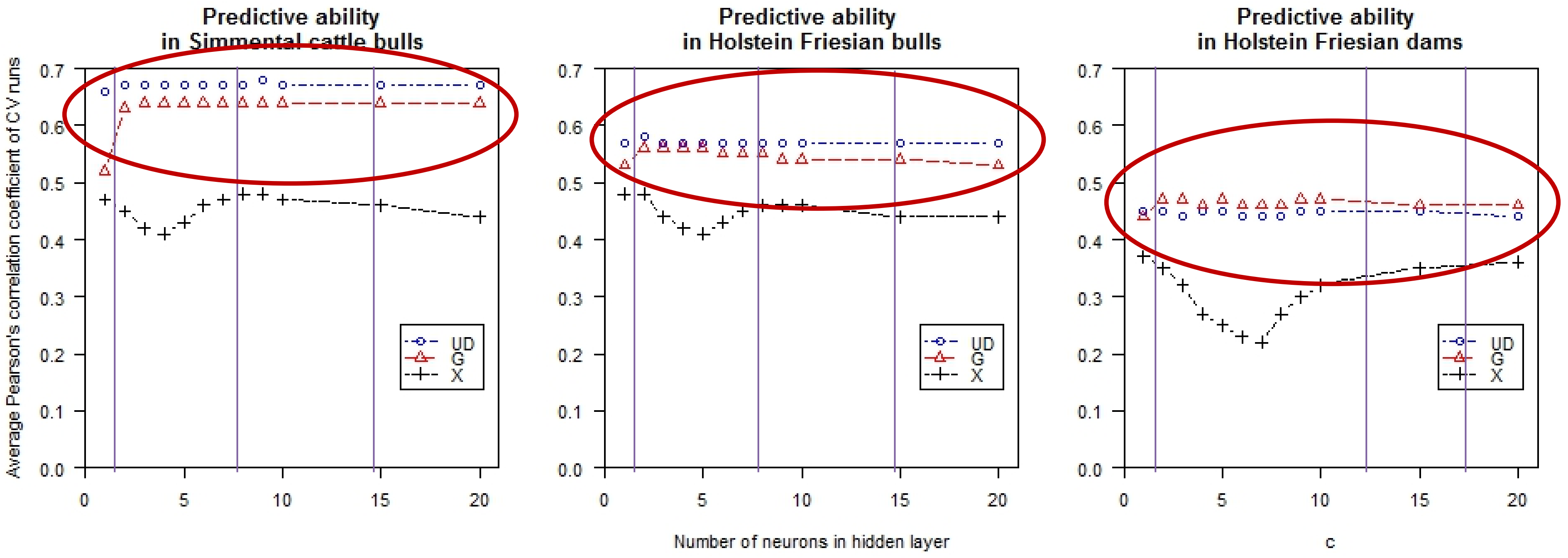
Results – Predicting milk yield

- 100 individual cross-validation runs
- Impact of
  - Network architecture
  - Genomic input
  - Phenotype



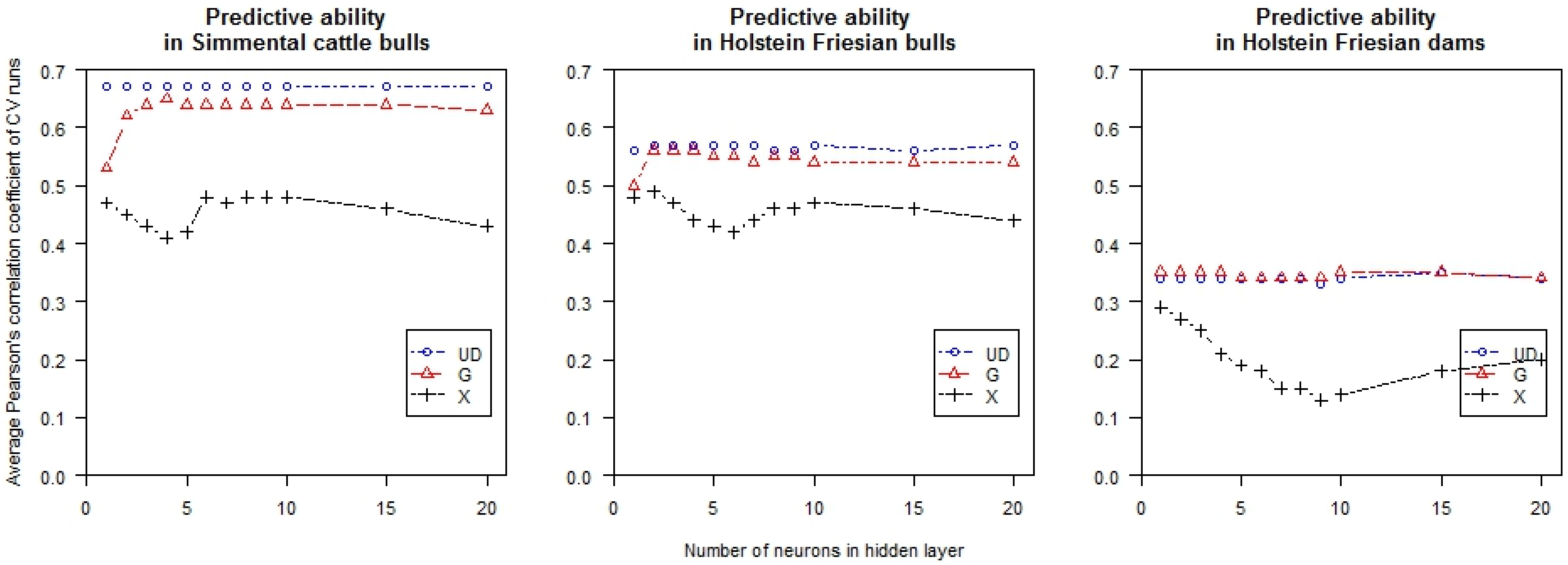
Results – Predicting milk yield

- 100 individual cross-validation runs
- Impact of
  - Network architecture
  - Genomic input
  - Phenotype



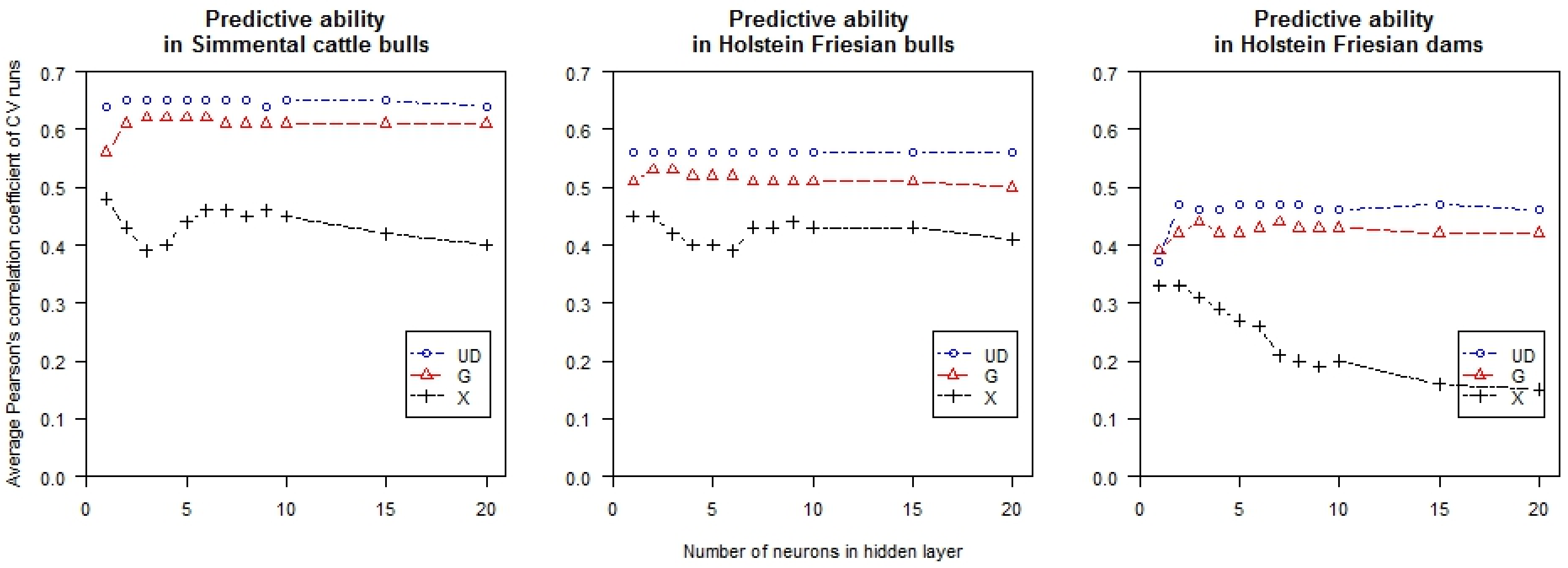
Results – Predicting protein yield

- 100 individual cross-validation runs
- Impact of
  - Network architecture
  - Genomic input
  - Phenotype



Results – Predicting fat yield

- 100 individual cross-validation runs
- Impact of
  - Network architecture
  - Genomic input
  - Phenotype



Conclusion

The network architecture

- ➢ Has only a slight effect on the predictive ability of future outcomes when dimension-reduction inputs are used
- ➢ Has a large effect on the prediction performance when the raw marker matrix is used
  - ✓ maybe a numerical problem?

The type of genomic input

- ➢ Has a large effect on the predictive ability
  - ✓ because of model complexity
- ➢ The principle component score matrix (UD) seems to be a suitable input
  - ✓ not loosing to much information of the original matrix while simultaneously reducing model complexity

Conclusion

The milk traits

- ➢ There is only a slight difference in prediction of future outcomes
    - ✓ because of a similar genetic background of the traits?

The number of animals in the analyses

- ➢ Maybe has an effect on the predictive ability as well as the pre-correction of the target trait
    - ✓ but hard to distinguish between both

In summary

Artificial neural networks can be applied to genome-enabled predictions but feature selection methods are highly recommended

Thank you for your attention!

"Always remember, there is no such thing as a free lunch - will you pick up the bill?"

Synbreed

SPONSORED BY THE

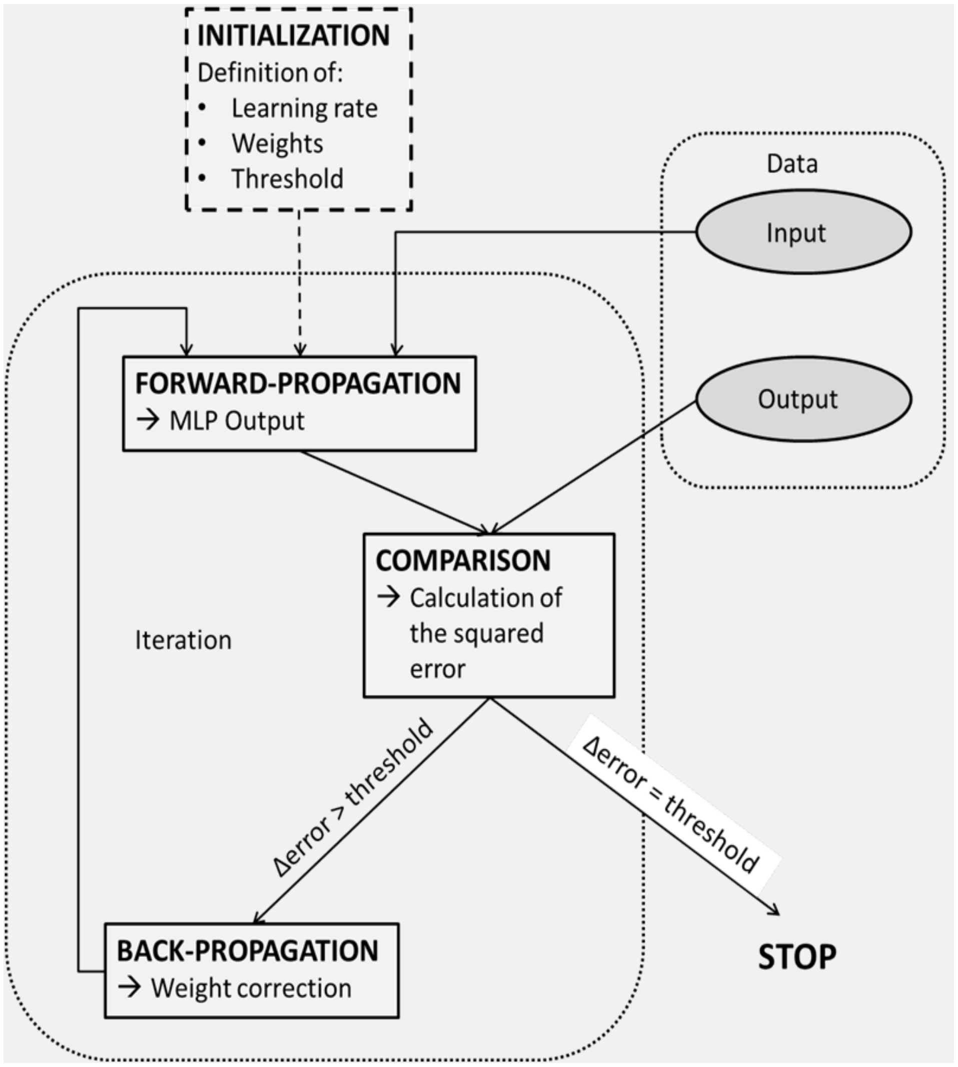
Federal Ministry
of Education
and Research

Results – supervised learning rule

Back-propagation algorithm
with early stopping (generalization)

➢ Minimization of a sum-of-squares error function using a gradient descent optimization

- ✓ Threshold = 10^{-3}
- ✓ Learning rate = 0.002
- ✓ Learn delay = 0.03
- ✓ Weights = random [-0.1;0.1]
- ✓ Feature scaling


INITIALIZATION
Definition of:
• Learning rate
• Weights
• Threshold
Data
Input
Output
FORWARD-PROPAGATION
→ MLP Output
COMPARISON
→ Calculation of the squared error
Iteration
Δerror > threshold
Δerror = threshold
BACK-PROPAGATION
→ Weight correction
STOP