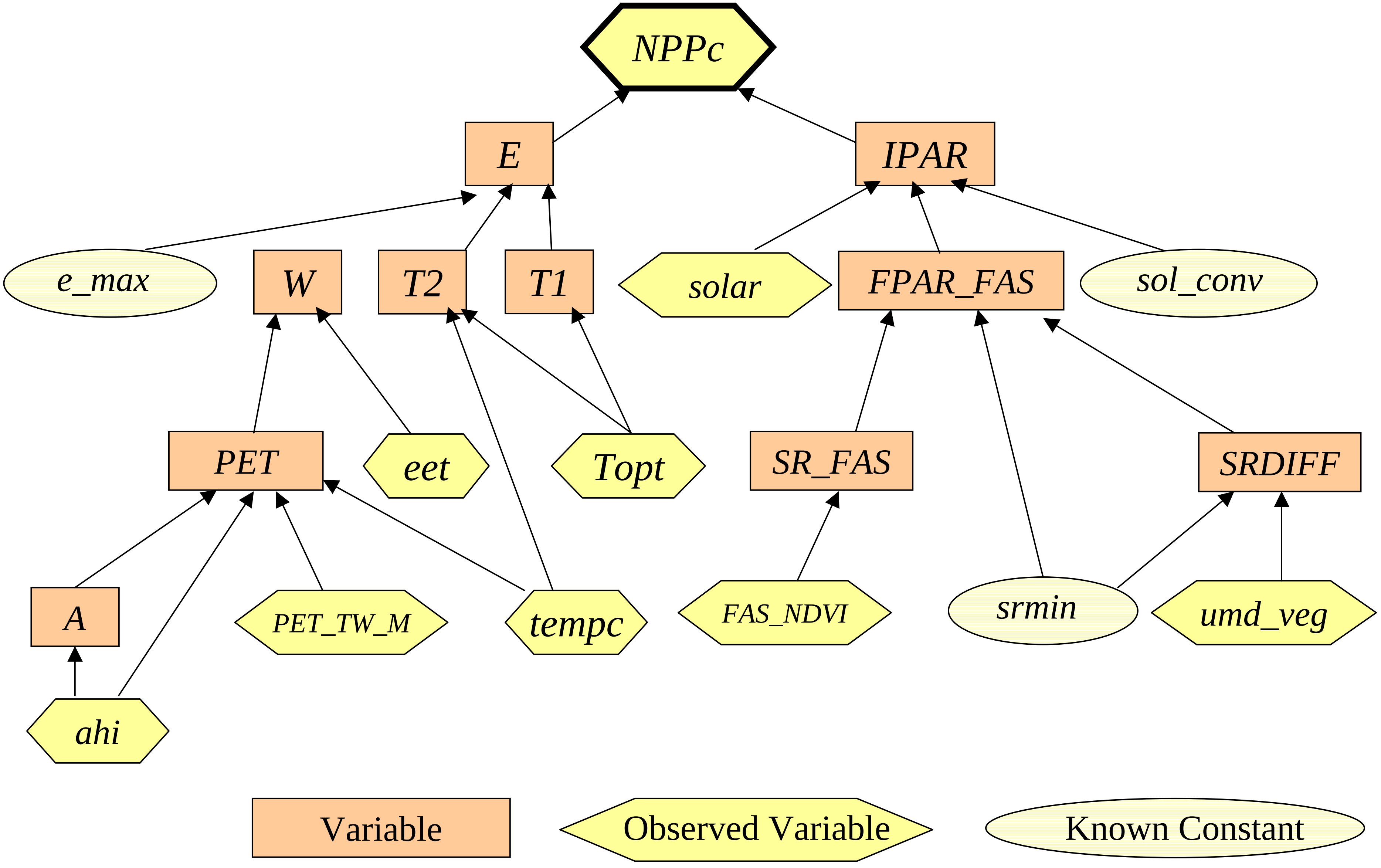
# Improving an Ecosystem Model Using Earth Science Data

Kazumi Saito (NTT Communication Science Laboratories)
Pat Langley (CSLI Stanford University)
Trond E. Grenager (Stanford University)

# Introduction

- Developing computational methods for discovering knowledge in communicable forms.
- Improving CASA using observed data.
- CASA: an existing computational model of aspect of the Earth ecosystem developed by Christopher Potter and his colleagues at NASA Ames.

# Portion of CASA



# Some Equations

*NPPc*: net primary production.

$$NPPc = \max\left(0, E \times IPAR\right)$$

*E*: value of maximum possible photosynthetic efficiency under temperature and moisture stress scalars.

$$E = e\_\max \times T1 \times T2 \times W$$

*IPAR*: converter for intercepted photosynthetically active radiation by the vegetation cover.

$$IPAR = FPAR\_FAS \times Solar \times sol\_conv \times 0.5$$

# General Problem

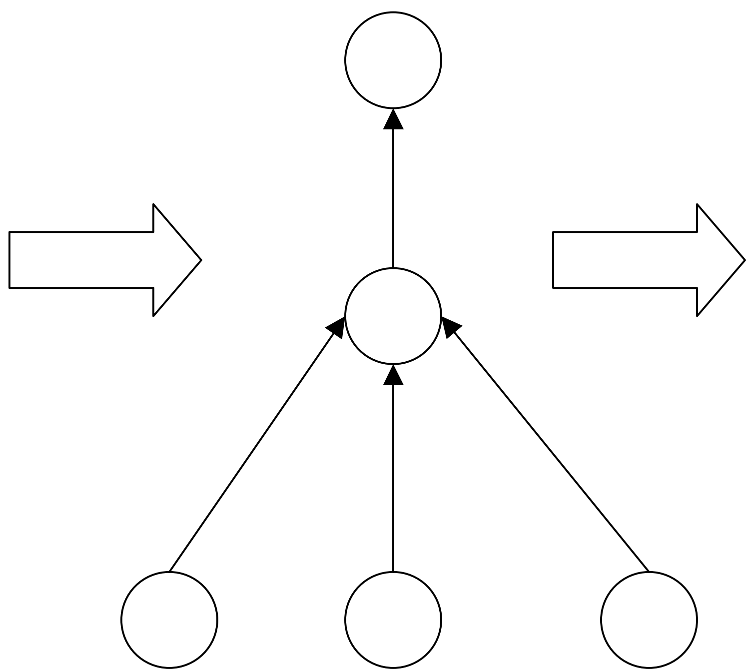
- Revisions to the model must be consistent with existing knowledge of Earth science and, ideally, retain similarity to the current model.
- Our research involves attempting to improve the CASA model's predictive accuracy.

# Outline of Approach

- Transforming the equations into a neural network
- Revising weights in that network
- Transforming the network back into equations

$$\begin{cases} NPPc = f(\cdots) \\ \cdots \\ \cdots \end{cases}$$

original equations



$$\begin{cases} NPPc = g(\cdots) \\ \cdots \\ \cdots \end{cases}$$

revised equations

# Some Types of Neural Networks

Standard (sigma-sigma) net:

$$\sum w_j f_j \left( \sum w_{jk} x_k \right)$$

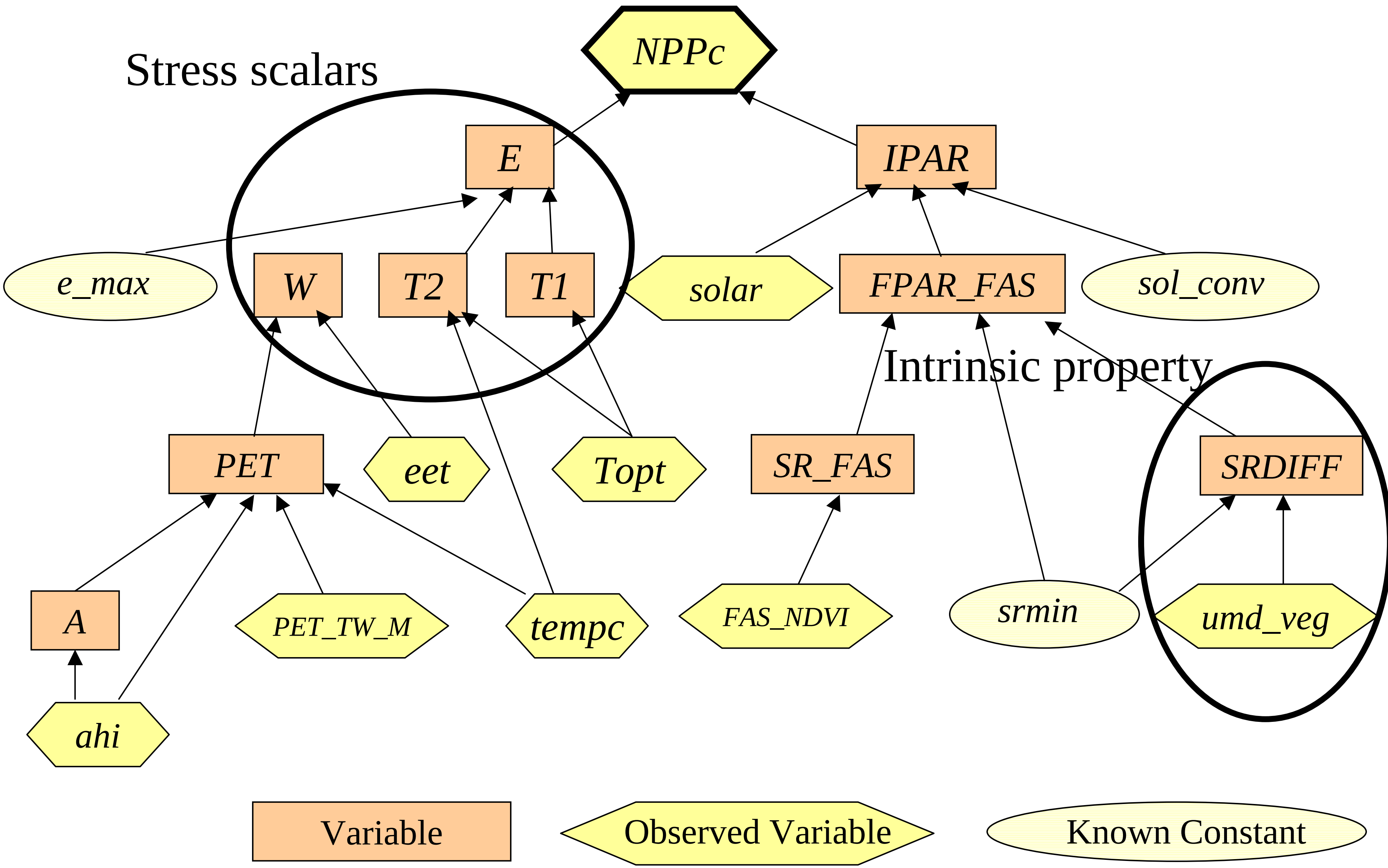
Sigma-pi net (generalized polynomial):

$$\sum w_j \prod x_k^{w_{jk}} = \sum w_j \exp\left( \sum w_{jk} \ln x_k \right)$$

Pi-sigma net (this talk):

$$\prod w_j f_j \left( \sum w_{jk} x_k \right)$$

# Transforming Equations



# Stress Scalars

Original equations:

$$E = e\_\max \times T1 \times T2 \times W$$

$$T1 = 0.8 + 0.02 \times Topt - 0.0005 \times Topt^2$$

$$= 1 - (-0.4472 + 0.0224 \times Topt)^2$$

$$T2 = 1.1814 \times \frac{1}{1 + \exp\left(0.2 \times (-10 + Topt - tempc)\right)}$$

$$\times \frac{1}{1 + \exp\left(0.3 \times (-10 - Topt + tempc)\right)}$$

$$W = 0.5 + 0.5 \times \left( \frac{eet}{PET} \right)$$

# Transformation into Network

$$E = e\_\max \times T1 \times T2 \times W = w_0 \times \prod f_i$$

$$T1 = f_1(x) = 1 - x^2 = f_1(w_{11} + w_{12} \times Topt)$$

$$= 1 - (-0.4472 + 0.0224 \times Topt)^2$$

$$T2 = f_2(x) = f_{21}(x) \times f_{22}(x) = \frac{1}{1+\exp(-x)} \times \frac{1}{1+\exp(-x)}$$

$$f_{21}(x) = f_{21}\left(w_{21} + w_{22}(Topt - tempc)\right)$$

$$= \frac{1}{1+\exp\left(2 - 0.2 \times (Topt - tempc)\right)}$$

$$W = f_3(x) = x = f_3\left(w_{31} + w_{32}\frac{eet}{PET}\right) = 0.5 + 0.5 \times \frac{eet}{PET}$$

# Intrinsic Values for Vegetation Type

*FPAR_FAS*: fraction of absorbed photosynthetically active radiation by the vegetation cover

$$FPAR\_FAS = \min\left( \frac{SR\_FAS - srmin}{SRDIFF}, 0.95 \right)$$

$$\approx \frac{1}{SRDIFF} \times (SR\_FAS \text{ - } srmin)$$

*SRDIFF*: map from the ground cover to an *srmax-srmin* value

# Transformation into Network

$$\frac{SR\_FAS - srmin}{SRDIFF}$$

$$= \exp\left(\log(SR\_FAS - srmin) - \log(SRDIFF)\right)$$

$$-\log\left(SRDIFF\right) = \sum_{i=1}^{13} v\_i \times umd\_veg\_i$$

$v\_i$ : weight in neural network

$$umd\_veg\_i = \begin{cases} 1 & \text{if} \quad umd\_veg = i \\ 0 & \text{othewise} \end{cases}$$

# Revising weights in Networks

| supervised learning | | Step-length | |
| --- | --- | --- | --- |
| | | fixed(constant) | variable |
| search direction | 1st-order | BP, etc. | Silva-Almeida algorithm,etc. |
| | 2nd-order | Newton method | SCG, OSS, BPQ, etc. |

| 2nd-order learning algorithm | applicability to large-scale problems | performance with inaccurate step-length |
| --- | --- | --- |
| Gauss-Newton method | × | ○ |
| quasi-Newton method | △ | △ |
| conjugate gradient method | ○ | × |

# BPQ Algorithm

- The search direction is calculated on the basis of partial BFGS update.

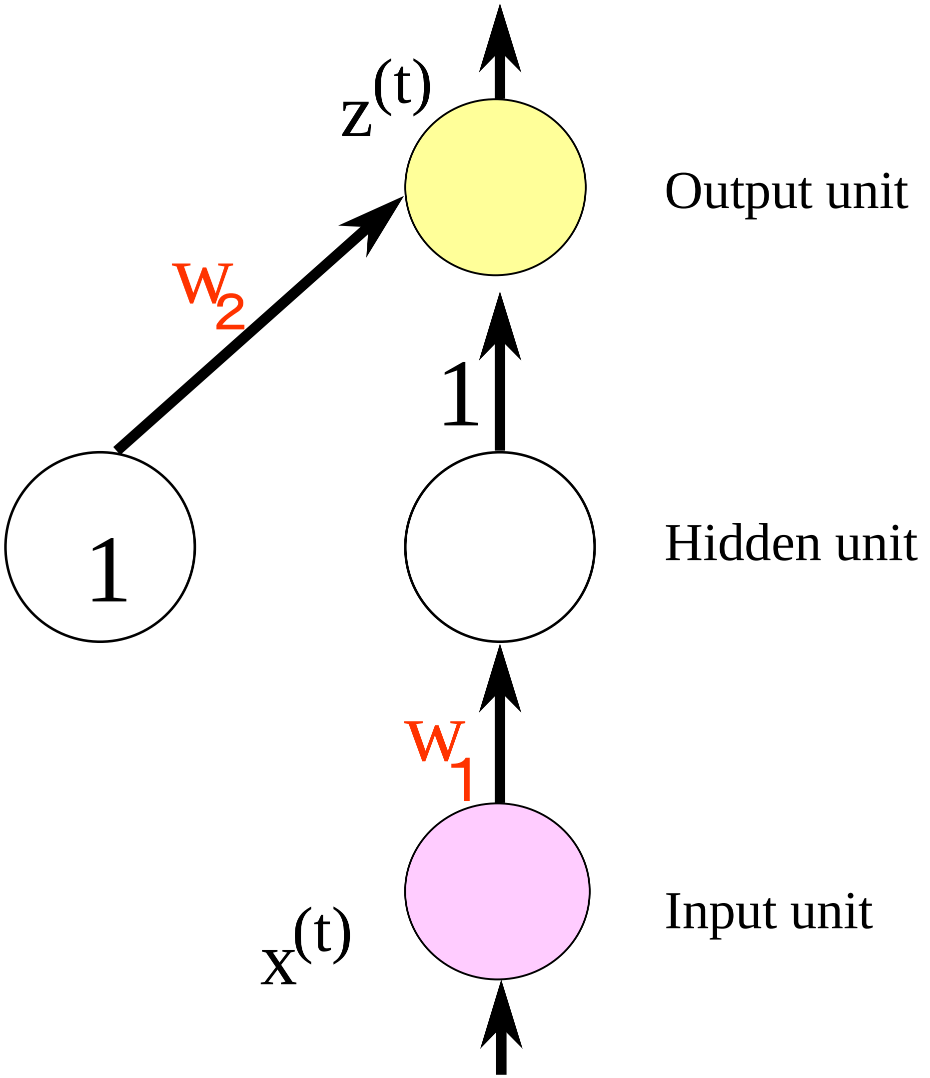
- The step-length is calculated by using a second-order approximation.

# Demonstration Problem

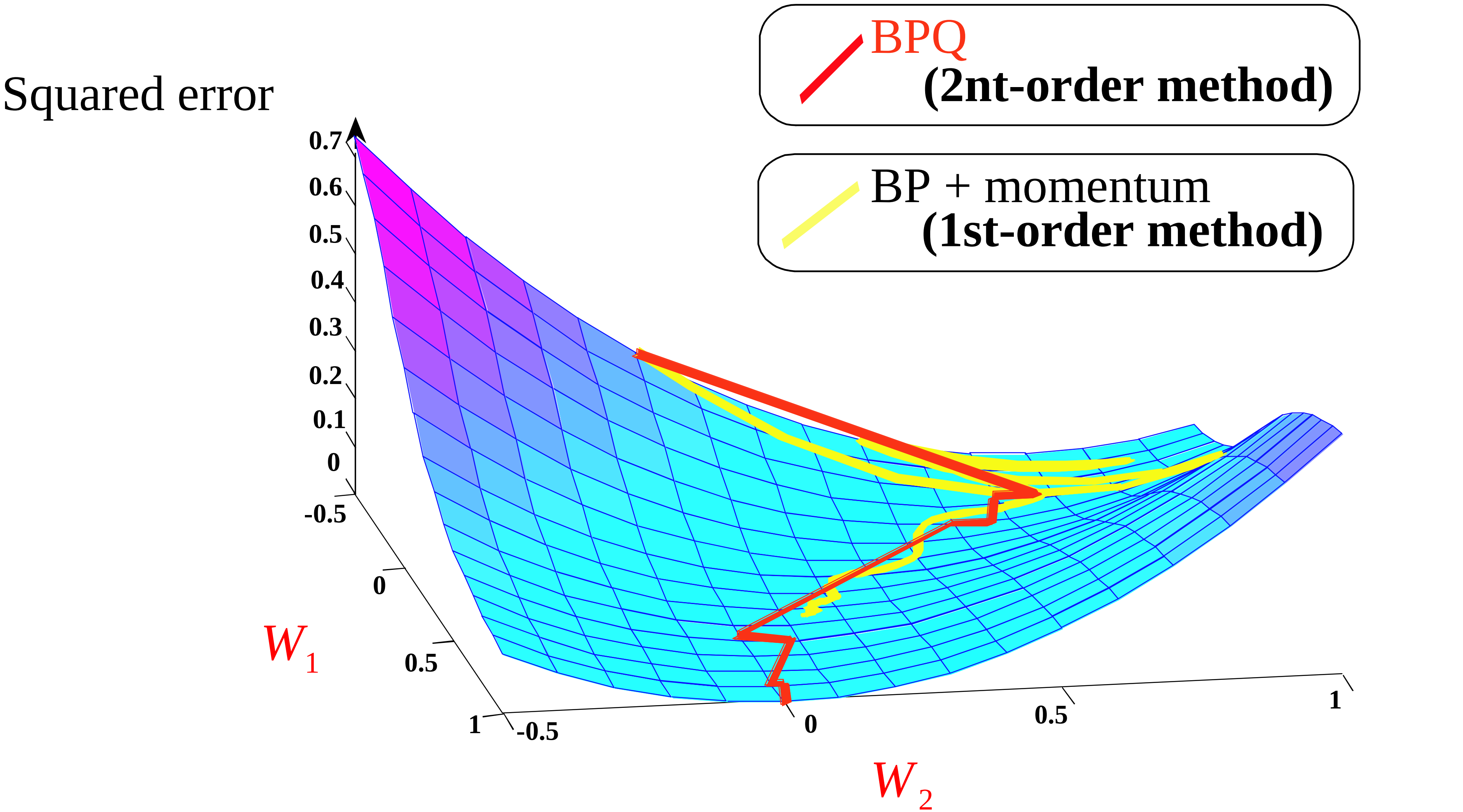
Sample set

Sample

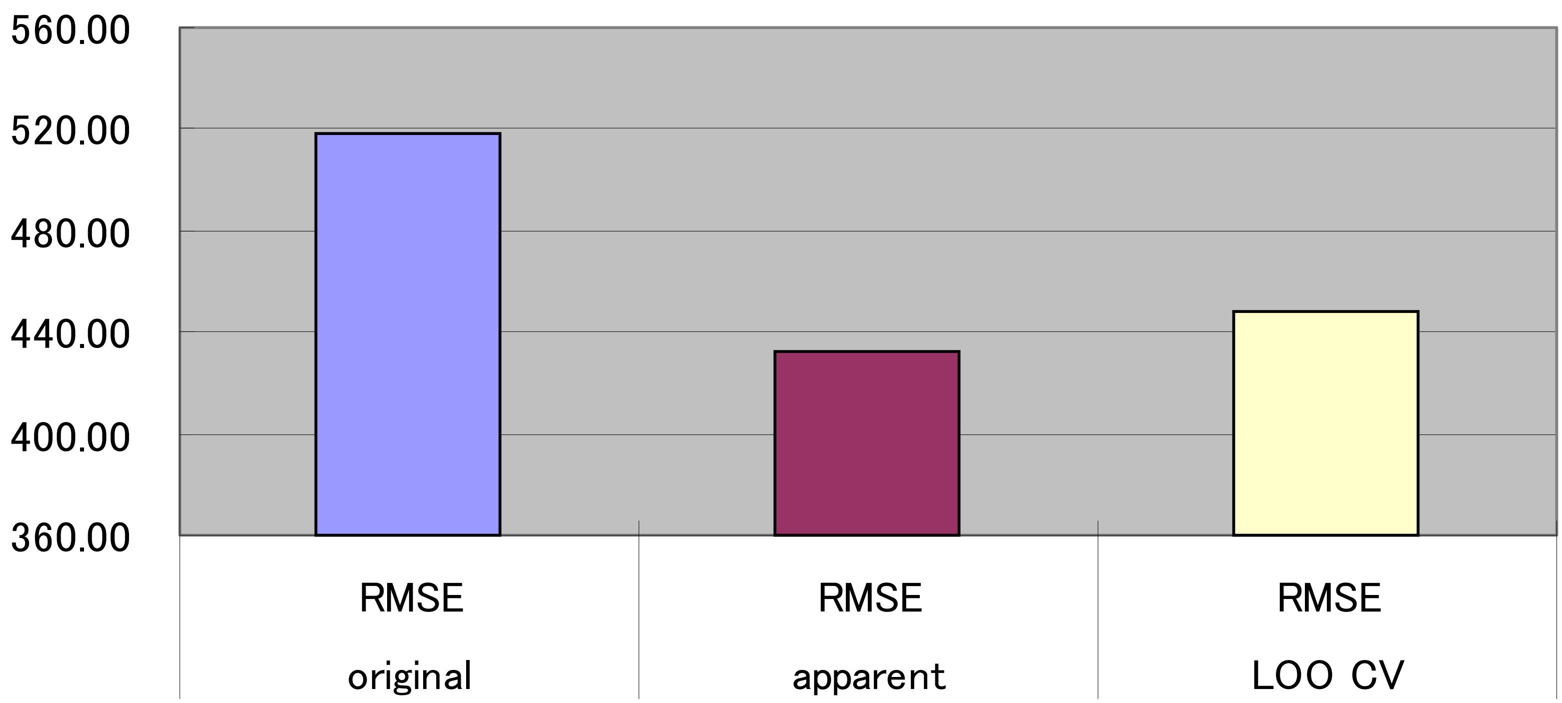
| | y | x |
|---|---|---|
| 1 | 0.73 | 1 |
| 2 | 0.88 | 2 |
| 3 | 0.95 | 3 |
| 4 | 0.98 | 4 |
| 5 | 0.99 | 5 |



# Learning Neural Network: Result



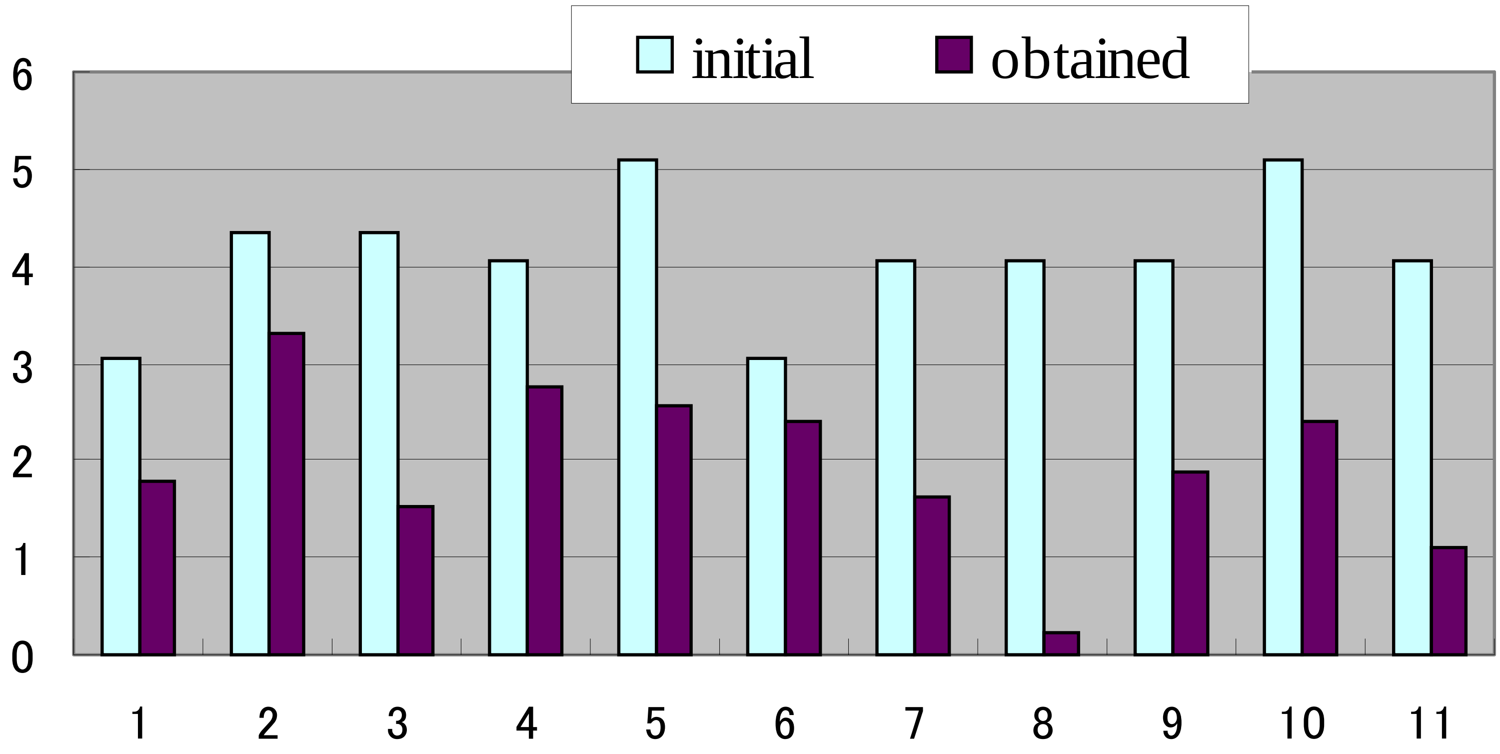
# Experimental Result




The RMSE of the original model was reduced by 15 percent, as measured using cross validation.

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{\mathrm{samples}} (\mathrm{NPP}_{\mathrm{observed}} - \mathrm{NPP}_{\mathrm{predicted}})^2}{\text{number of samples}}}$$

# Intrinsic Values




The intrinsic values associated with vegetation types obtained in this way were consistently lower

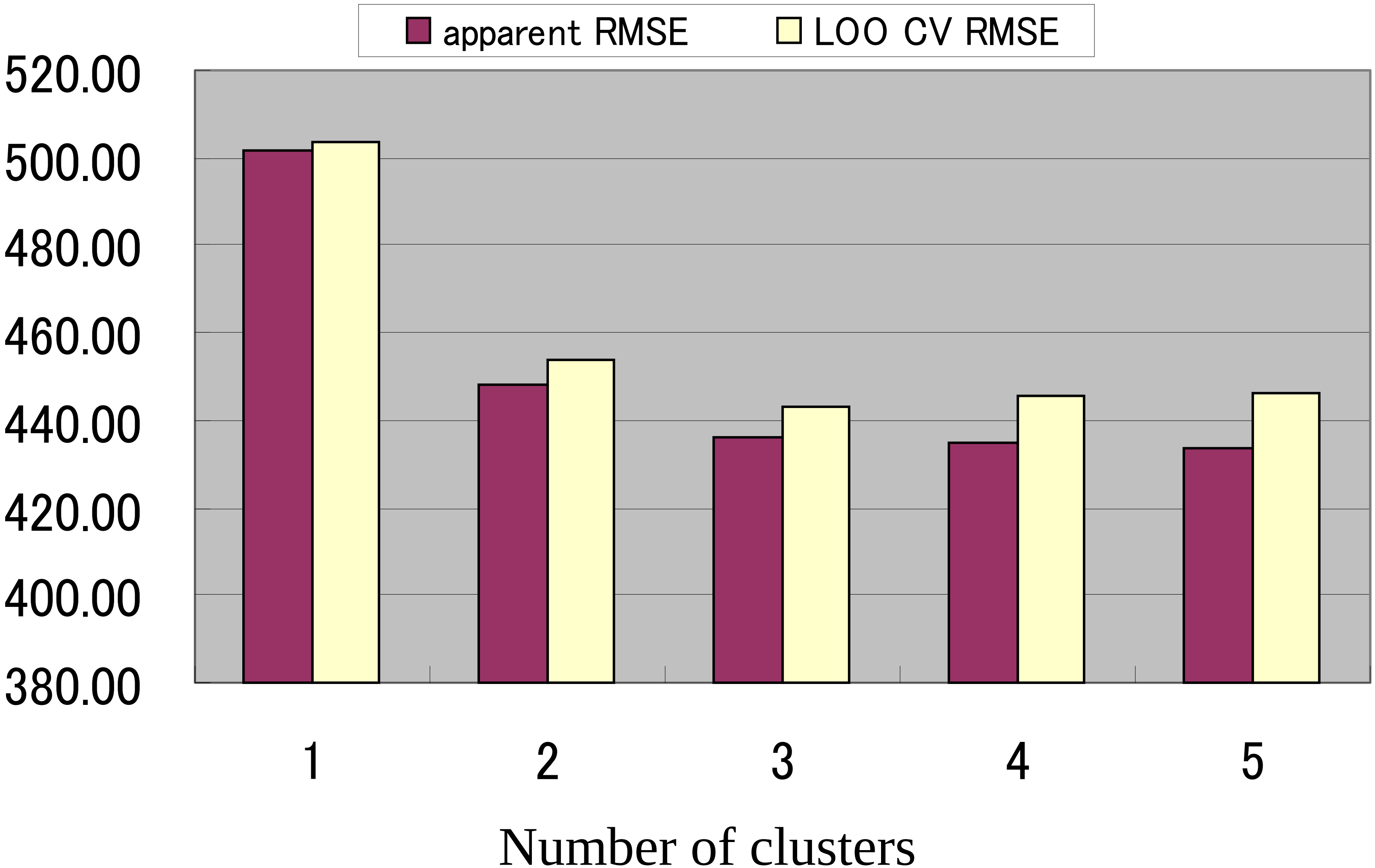
# Transforming Network

Step1. Quantize $\left\{\exp\left(\sum_{kl} v_{kl} q_{kl}^{(n)}\right) : n = 1, \cdots N\right\}$
by using a clustering method.

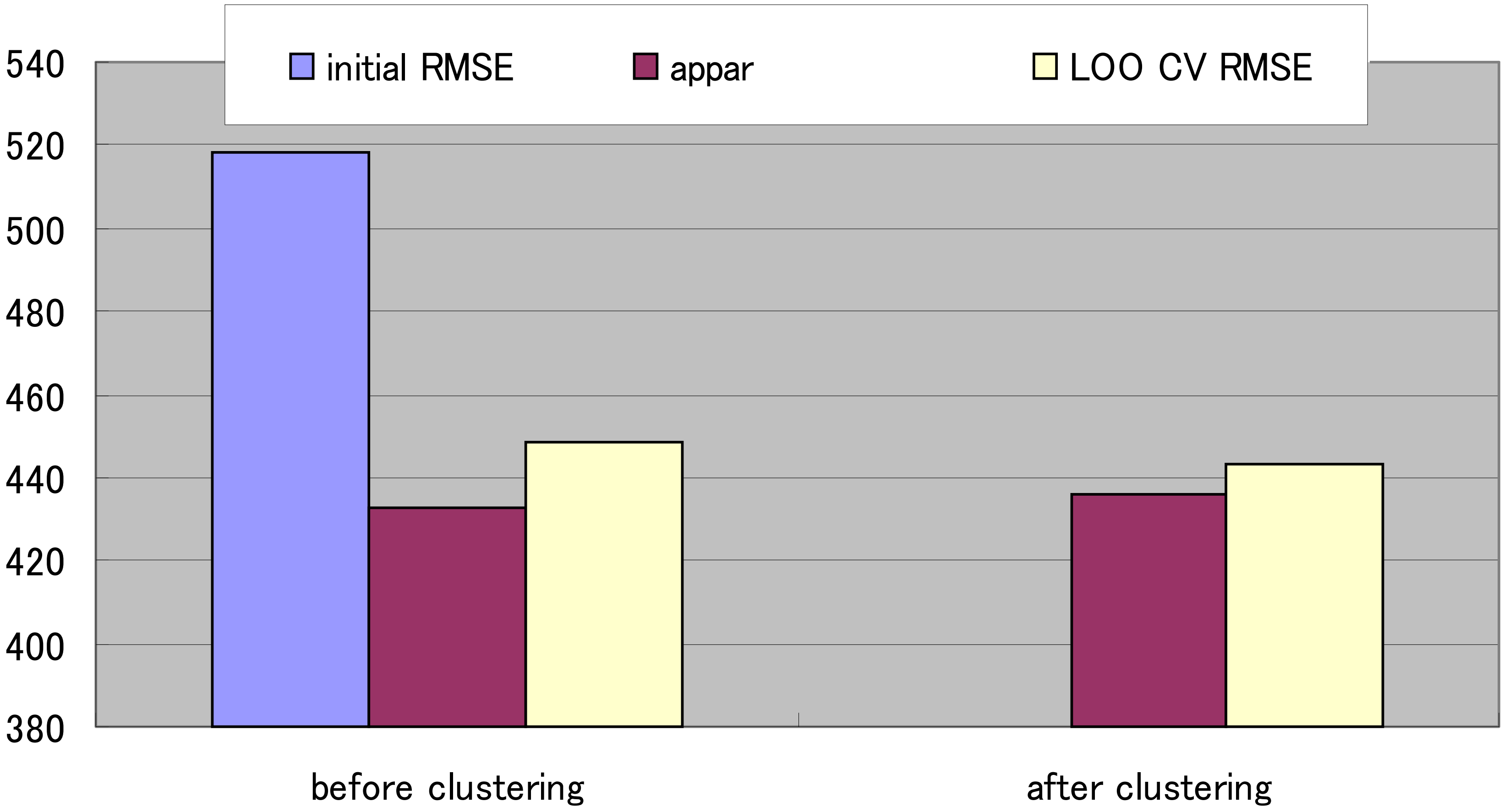
Step2. Determine an adequate number of rules
by using cross-validation.

Step3. Generate nominal condition
by solving a standard classification problem.

# Clustering Analysis



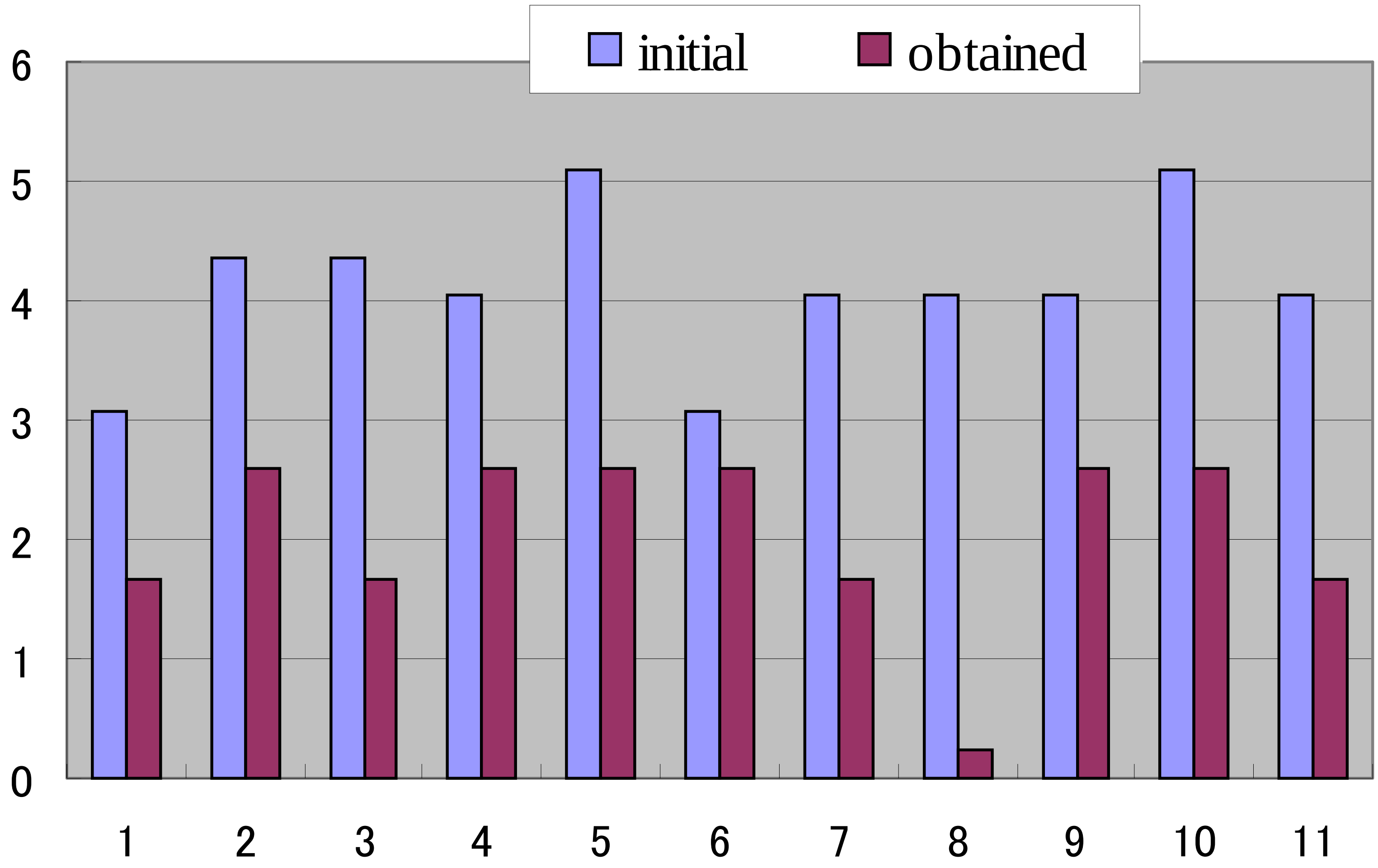
# Evaluating Experimental Result



# Obtained Decision Tree

```
t8 = 1: 0 (10.0)
t8 = 0:
|   t9 = 1: 1 (58.0)
|   t9 = 0:
|   |   t7 = 1: 1 (46.0)
|   |   t7 = 0:
|   |   |   t11 = 1: 1 (11.0)
|   |   |   t11 = 0:
|   |   |   |   t1 = 0: 2 (168.0/1.0)
|   |   |   |   t1 = 1: 1 (10.0)
```

# Clustered Intrinsic Values



# Conclusion

- This talk described an approach to improving the predictive accuracy of the existing ecosystem model.
- In the experiments, we can reduce the mean squared error of the original model by 15 percent, as measured using cross validation
- In the future, we’ll carry out further experiments along this direction