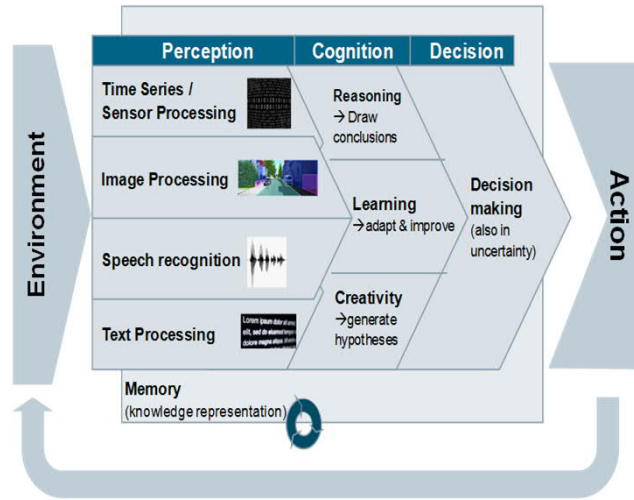
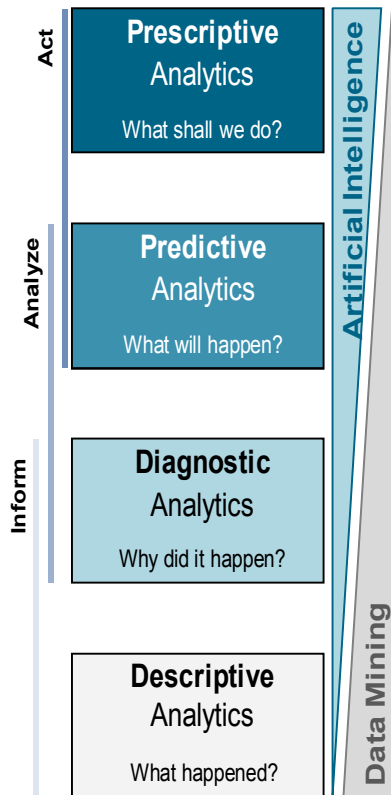


# Forecasting Customer Demand with Deep Neural Networks



# Data Analytics & Artificial Intelligence



Neural Networks      Support Vector Machines  
 Deep Learning      Decision Trees  
 (Non-)linear Regression      Random Forests  
 Dynamic Systems      Reinforcement Learning      Bayesian Networks  
 Gaussian Processes



## Predictive Maintenance

- Analysis of plant data to detect or predict failures
- Suggest measures to ensure uptime.

## Process Surveillance

- Analysis of production processes to ensure product quality
- Diagnose failures in production, avoid scrap

## Digital Twins

- Learn models from data
- Help to parameterize analytical models
- Feature selection / identification

## Planning

- Demand forecasts
- Commodity price forecasts
- Evaluation of business transactions



## Agenda

- Forecasting Customer Demand: Problem Setting & Test Set-up
- Forecast Models: Focus on Neural Networks
- Model comparison and benchmarking
- From forecasting to decision making (planning)



## Forecasting Customer Demand at the Geräte-Werk Erlangen (GWE)

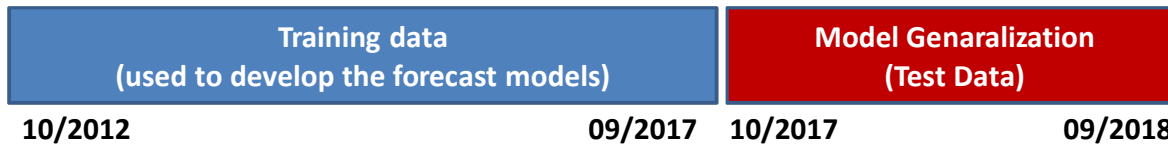
### Problem Setting:

- Predict the customer demand for products for **the next year** in weekly time buckets (i.e. 52 forecasts)
- Forecast objects are so-called MLFBs (fine granularity, e.g. a product with a specific configuration). We have about **1100 active MLFBs**
- Forecasts should be used for production planning and material sourcing
- A **fully automated forecast and planning process** should be established



## Test Set-Up

- We predict the customer demand under “real-world-conditions” (so-called backtest)
- Test horizon:



- We measure the performance in terms of absolute error in pieces and the standard deviation of the error over the forecast objects



## Model Inputs

- Calendar effects
  - WeekOfMonth
  - WeekOfQuarter
  - WeekOfYear
- Work days per week
- German public holidays
- Customer demand on preferred delivery date. Data is available from 2011 to 2018 (daily bookings weeks)
- Macroeconomic data: Business climate of the top 10 customer sectors and top 5 regions (both monthly data).



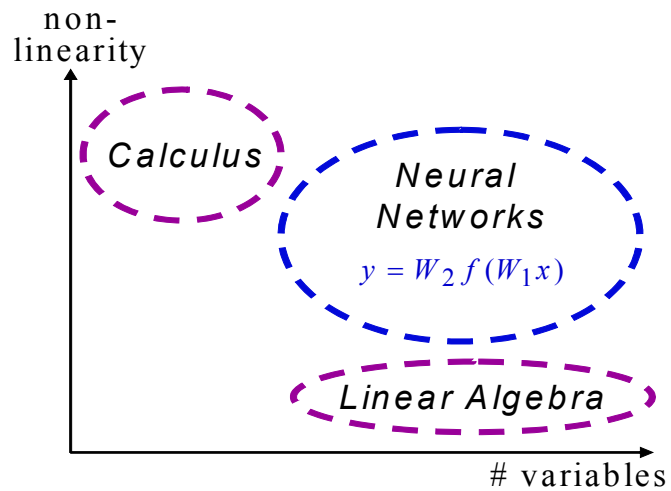
## Agenda

- Forecasting Customer Demand: Problem Setting & Test Set-up
- **Forecast Models: Focus on Neural Networks**
- Model comparison and benchmarking
- From forecasting to decision making (planning)





## Mathematical Neural Networks in Nonlinear Regression



### Existence Theorem:

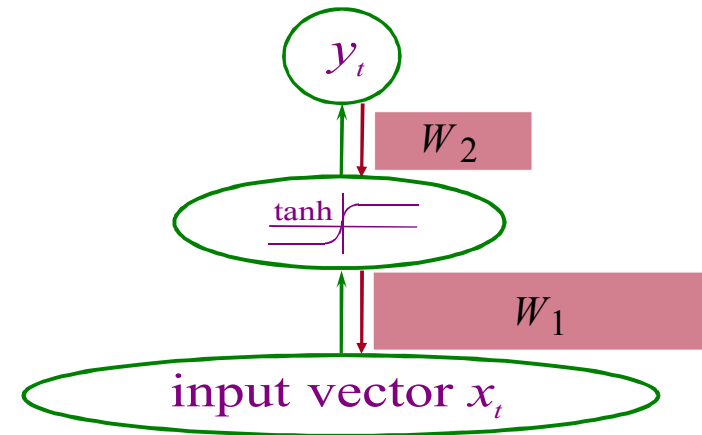
(Hornik, Stinchcombe, White 1989)

3-layer neural networks can approximate any continuous function on a compact domain.

Based on data identify an input-output relation

$$y = W_2 f(W_1 x)$$

$$\sum_{t=1}^T (y_t - y_t^d)^2 \rightarrow \min_{W_1, W_2}$$

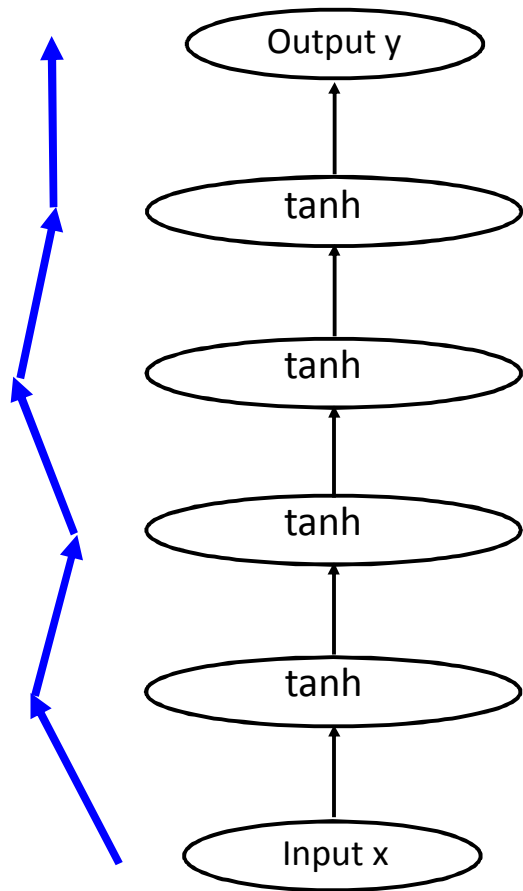


NN imply a **Correspondence** of **Equations**, **Architectures**, **Local Algorithms**.





## Why Deep Feed-Forward Neural Networks



The learning of deep neural networks is difficult, because...

- first layers can do nonsense, while the last hidden layer corrects everything – why to use a deep structure?
- forward path: relevant input information may get lost in the hierarchy of hidden layers,
- backward path: the error signal decays passing through many hidden layers.

Challenge: Define explicit tasks for the intermediate layers!

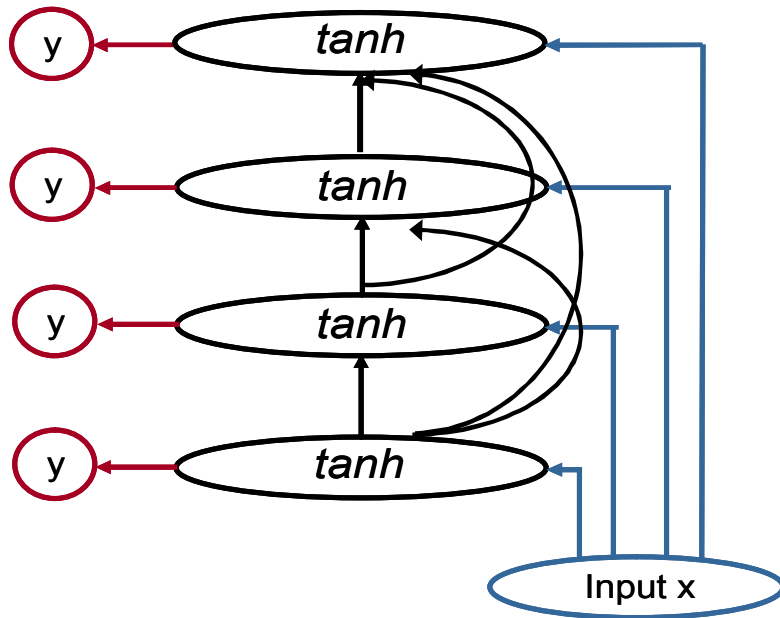
- + to allow a sequential assessment of very complicated input-output relationships,
- + to do a dimensionality reduction for very large input layers,
- + to exploit neighboring relationships in the inputs (images, spectra, time series).





## Deep Feed-Forward Neural Networks

The extended net acts as a hierarchical filter



Forward path: feeding the inputs to all intermediate layers avoids a loss of the input information.

Backward path: learning is not only applied to the final target but to all intermediate layers.

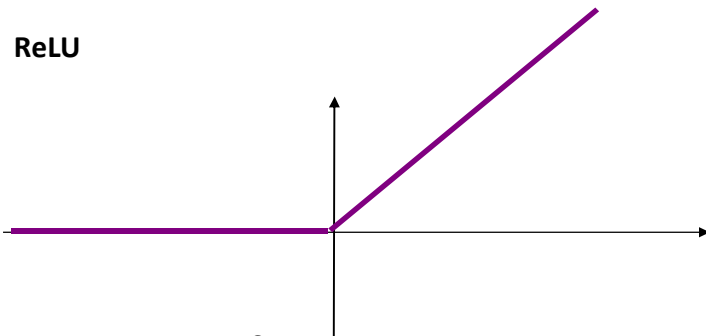
Performance is improved if we manipulate the backward flow in the hidden backbone (hidden  $\rightarrow$  hidden connections learn, but transfer no error flow)

We can add information highways to spread information (faster) across the network

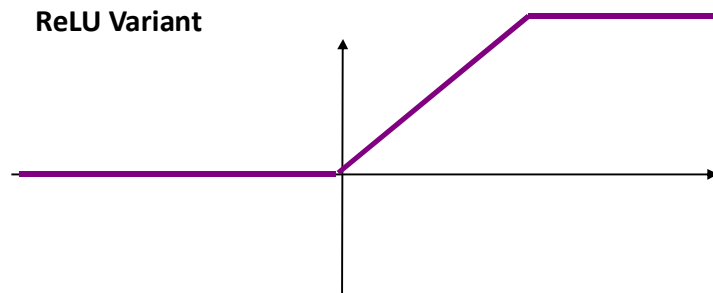
Drop-out creates “internal” ensemble models



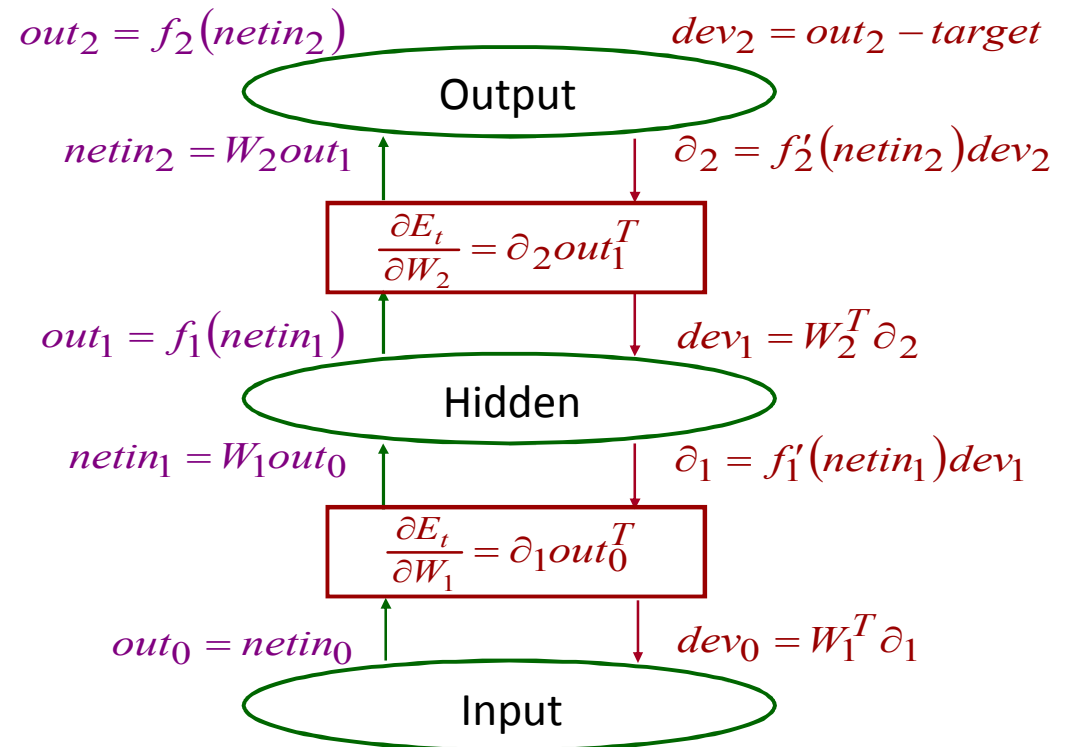
## Rectified Linear Unit (ReLU, Alternative Basis Function)



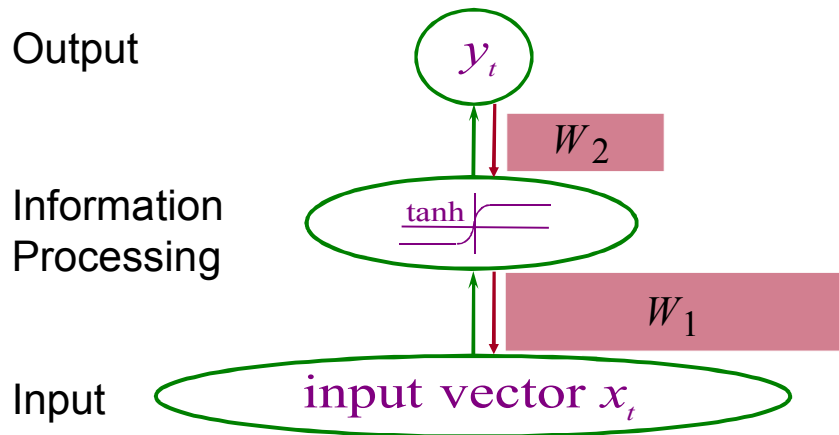
For  $x > 0$ , the error flow is not squeezed  
 For  $x < 0$ , we have no error flow anymore.  
 For  $x = 0$  the function is not differentiable.



Squashing functions differ in their impact on the backpropagation algorithm.

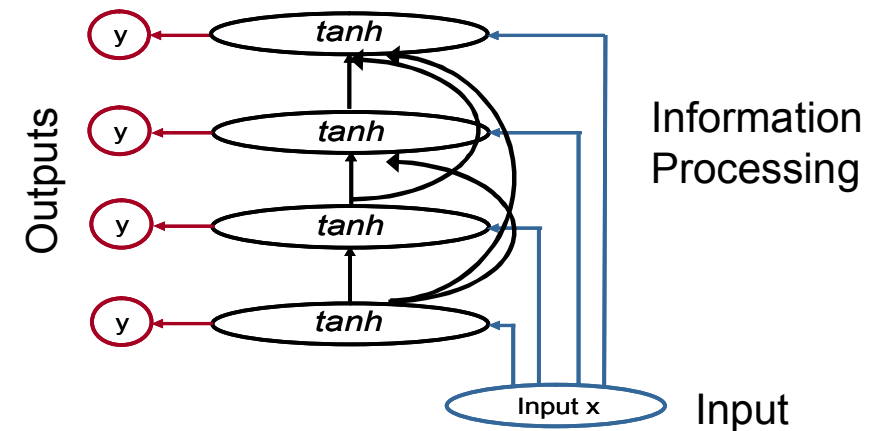


## Standard 3-Layer Neural Networks



- Non-linear regression approach and universal function approximator
- All tasks are solved by a single information processing layer. Tends to overfit the data

## Example of a Deep Neural Network

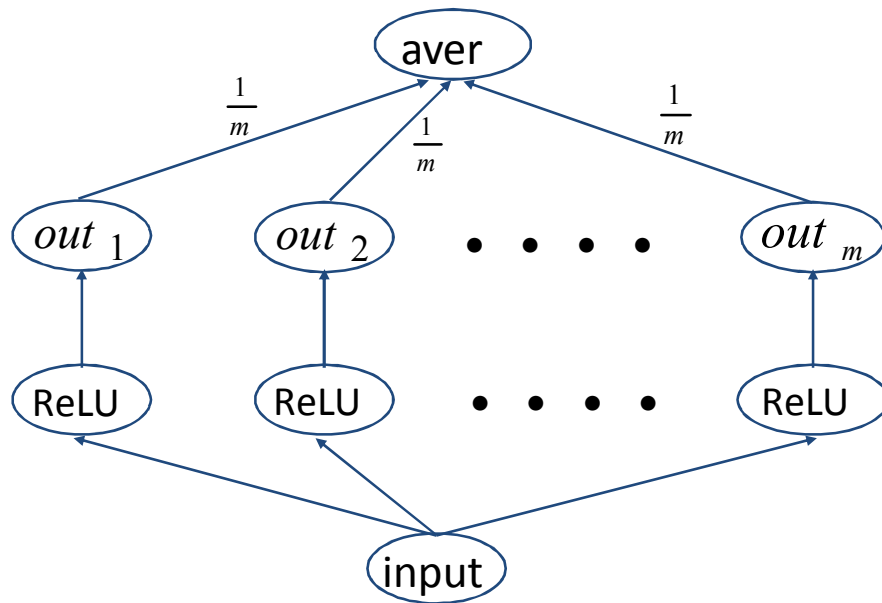


- Multi-level information processing: Enrichment of the information flow by multiple hidden-to-hidden and input-to-hidden connections
- Multiple outputs (same task) enrich the learning





## Decreasing Model Uncertainty by Averaging



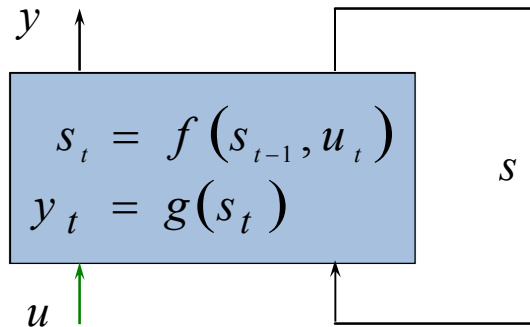
The sub-networks learn different solutions of the same task. In case of large averages ( $m > 20$ ) an equal weighting is superior, in case of small averages it is superior to freeze the subnets and optimize the weighting factors.

$$\begin{aligned}
 E_{aver} &= \frac{1}{T} \sum_T [out_{aver} - tar]^2 \\
 &= \frac{1}{T} \sum_T \left[ \left( \frac{1}{m} \sum_i out_i \right) - tar \right]^2 \\
 &= \frac{1}{T} \sum_T \left[ \frac{1}{m} \sum_i (out_i - tar) \right]^2 \\
 &= \frac{1}{m^2} \frac{1}{T} \sum_T \sum_i (out_i - tar)^2 \\
 &= \frac{1}{m} \frac{1}{m} \sum_i \frac{1}{T} \sum_T (out_i - tar)^2 \\
 &= \frac{1}{m} aver(E_i)
 \end{aligned}$$

$$\begin{aligned}
 & \frac{1}{T} \sum_T (out_i - tar) \cdot (out_j - tar) = 0 \quad \forall i \neq j \\
 & \text{(covariance of the errors of the submodels)}
 \end{aligned}$$



# Dynamical Systems and Recurrent Neural Networks (RNN)



$$s_t = \tanh(As_{t-1} + Bu_t) \quad \text{state transition}$$

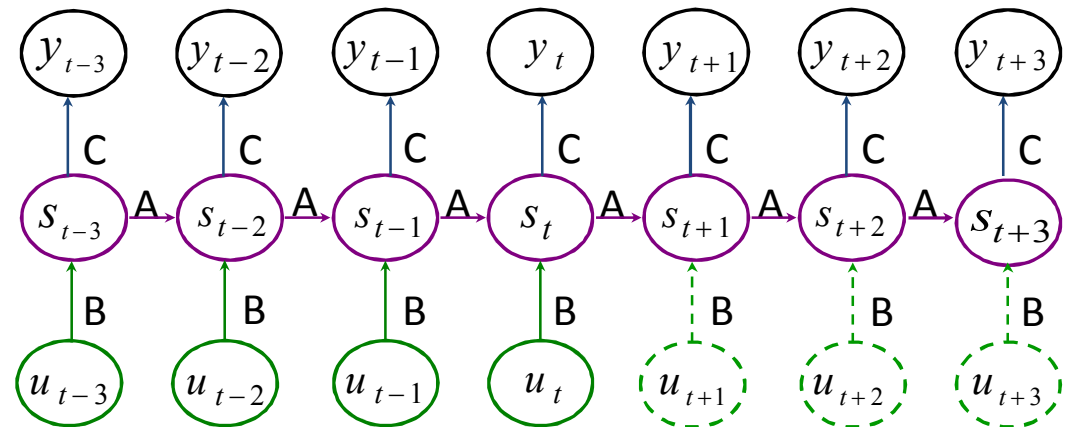
$$y_t = Cs_t \quad \text{output equation}$$

$$\sum_{t=1}^T (y_t - y_t^d)^2 \rightarrow \min_{A, B, C} \quad \text{identification}$$

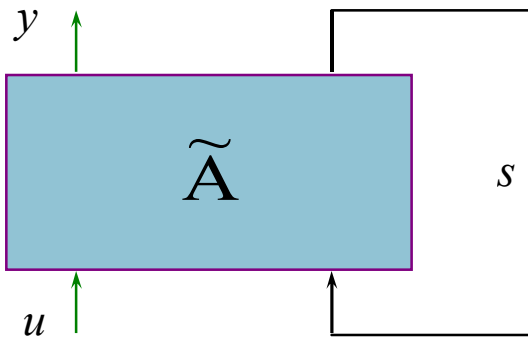
Finite unfolding in time transforms time into a spatial architecture. We assume, that  $x_t = \text{const}$  in the future.

The analysis of open systems with RNNs allows a decomposition of the **autonomous** & **external driven** part.

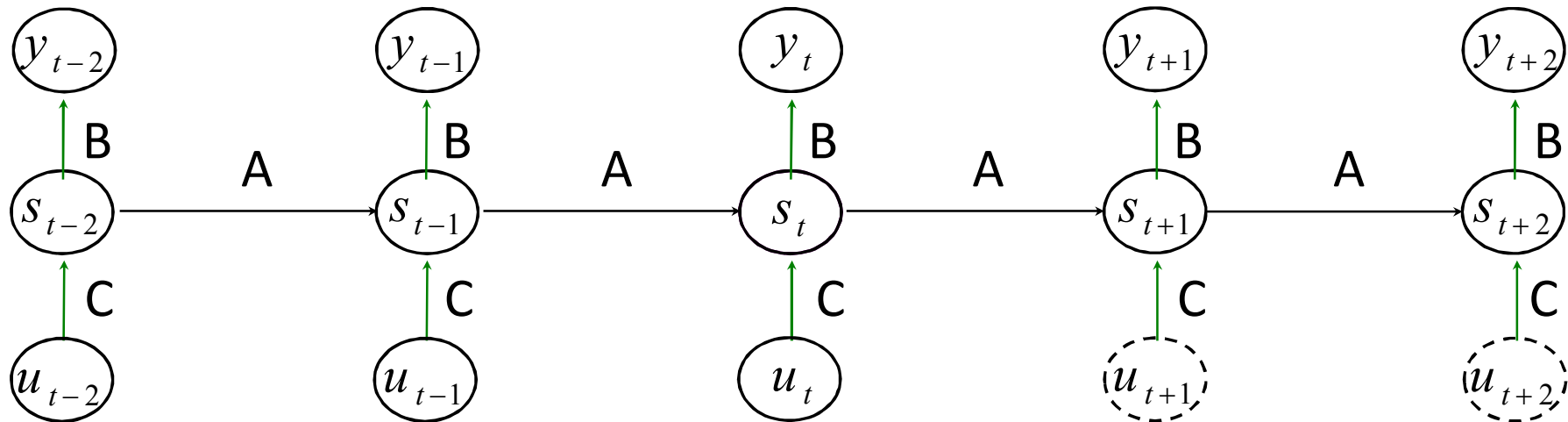
Long-term predictability depends on a strong autonomous subsystem.



## Echo State Neural Networks



In echo state networks we have large state space with a fixed, sparse transition matrix  $A$  and matrix  $C$  for incorporating the inputs. We learn a linear filter  $B$ .



## Causal & Retro-Causal Neural Networks (CRCNN)

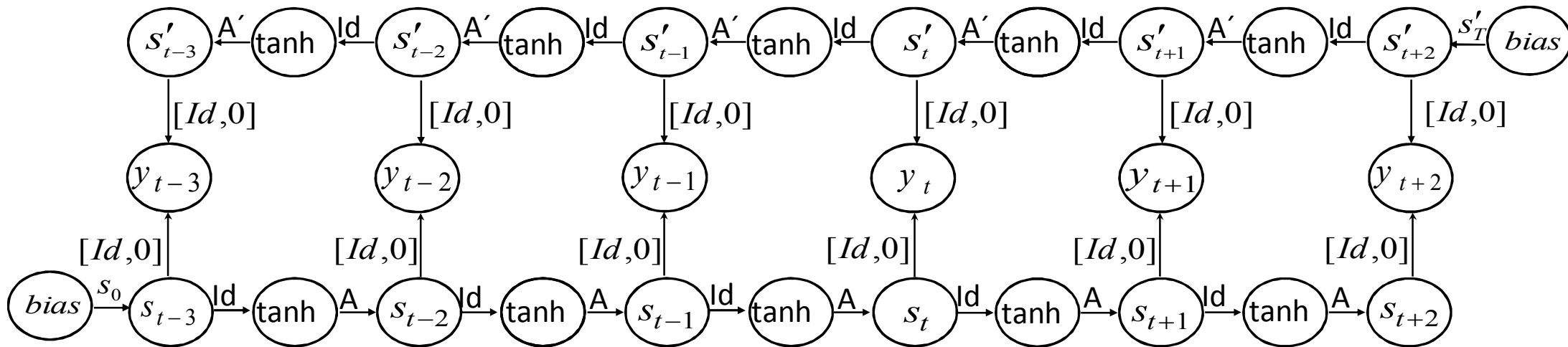
- We explain observations by a symmetric superposition of causal & retro-causal subnets.
- Both subnets are universal learners, but the more appropriate branch learns faster and reduces the error flow of the opposite branch.
- In non-unique optimization, we have to have attention on the path to the optimum!!!

$$s_t = A \tanh(s_{t-1}) \quad , s_0 \quad \text{causal transition}$$

$$s'_t = A' \tanh(s'_{t+1}) \quad , s'_T \quad \text{retro transition}$$

$$y_t = [Id, 0]s_t + [Id, 0]s'_t \quad \text{output equation}$$

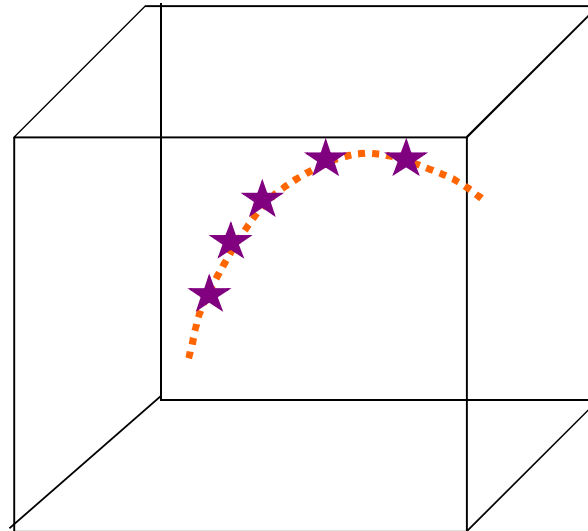
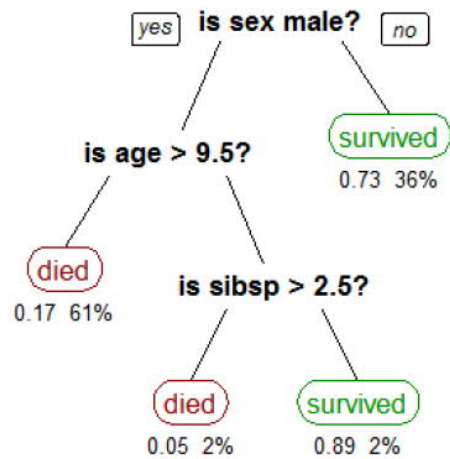
$$\sum_{t=1}^T (y_t - y_t^d)^2 \rightarrow \min_{A, s_0, A', s'_T} \quad \text{identification}$$





## Additional Benchmarks

- Random Forests incorporating gradient boosting
- Support vector machines (SVM)



$$f(x) \approx \sum_{j=1}^m v_j b(x-x_j)$$

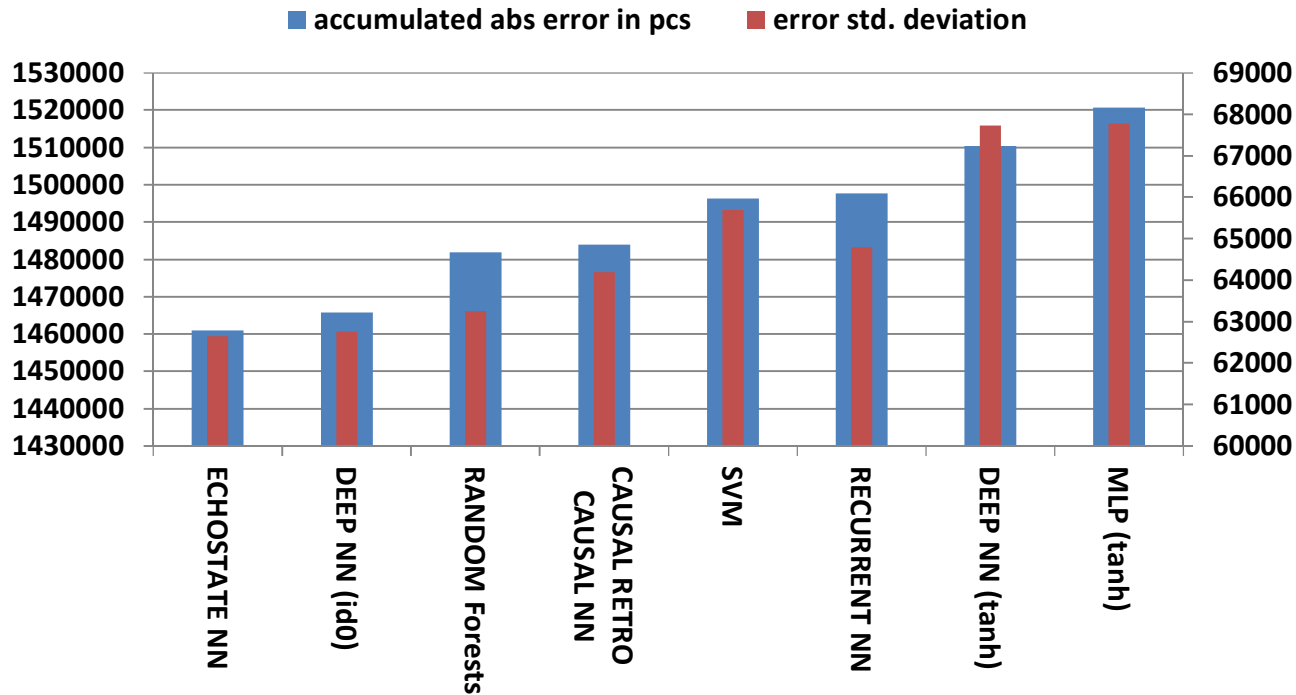


## Agenda

- Forecasting Customer Demand: Problem Setting & Test Set-up
- Forecast Models: Focus on Neural Networks
- **Model comparison and benchmarking**
- From forecasting to decision making (planning)



## Model comparison and benchmarking



Model	accumulated abs error in pcs	error std. deviation
ECHOSTATE NN	1461094	62651
DEEP NN (ido)	1465803	62768
RANDOM Forests	1481960	63263
CAUSAL RETRO CAUSAL NN	1484119	64194
SVM	1496393	65694
RECURRENT NN	1497693	64807
DEEP NN (tanh)	1510528	67733
MLP (tanh)	1520819	67777

Accumulated absolute error in pieces over 52 weeks and all products (MLFBs) on test data.

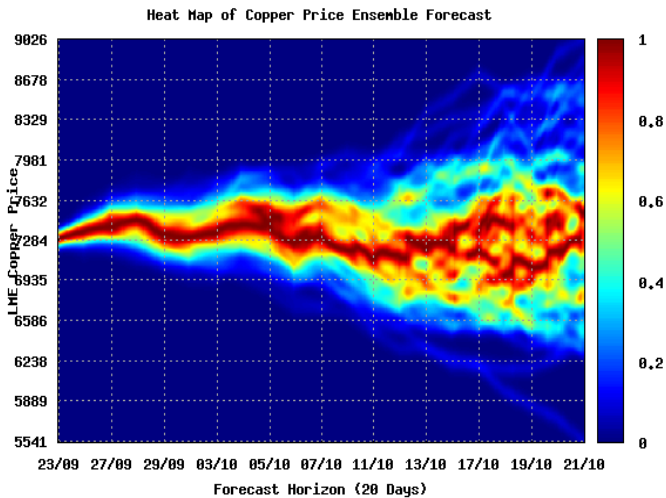
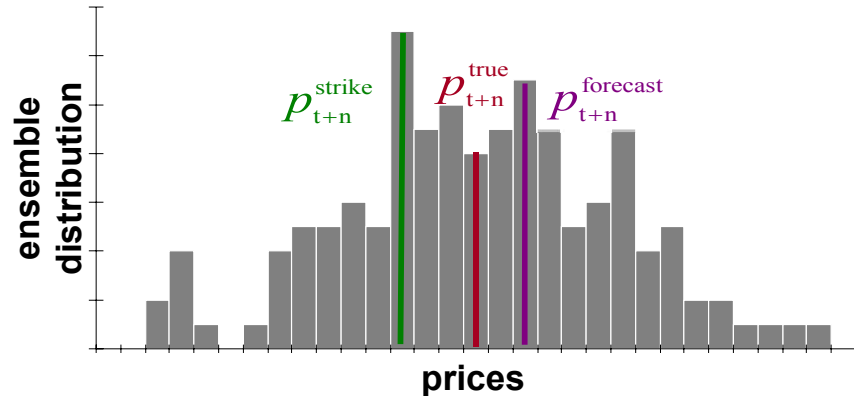


## Agenda

- Forecasting Customer Demand: Problem Setting & Test Set-up
- Forecast Models: Focus on Neural Networks
- Model comparison and benchmarking
- **From forecasting to decision making (planning)**



## On Risk and Forecasting Uncertainty

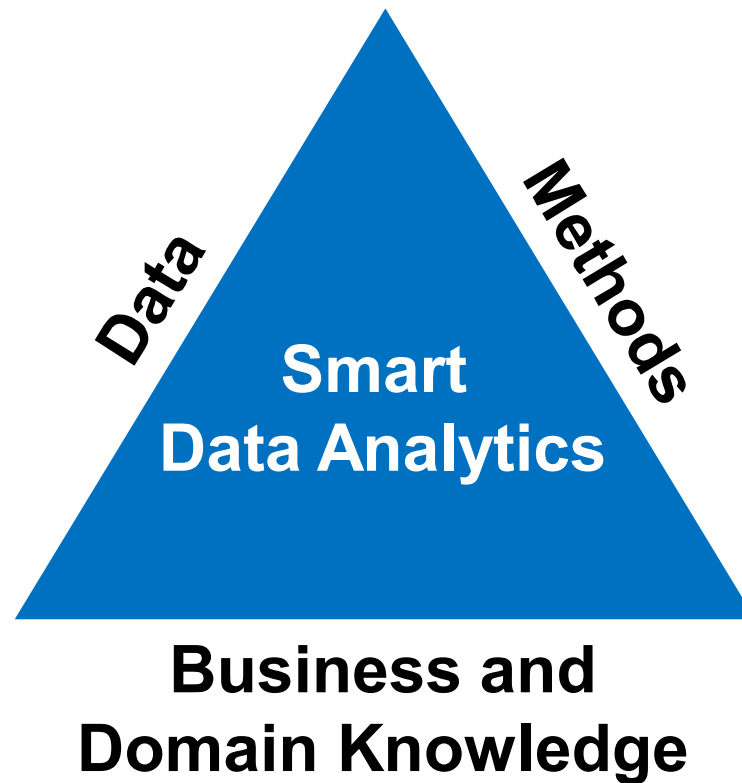


### Paradigms for the Computation of Uncertainty / Risk:

- **Volatility** as a risk measure implies a naive, constant forecast (a high frequency wave has high volatility without uncertainty).
- **Backward risk is the error between model and real-world.** Here distributional features of the risk depends on the chosen model.
- Forward risk can be estimated by an **ensemble forecast**: many models fit the past perfectly and still give diversifying forecasts.
- Risk is often not double-sided within the uncertainty distribution, e.g. unexpected gain or unexpected loss!



## The Magic Triangle in Data Analytics



### Customer benefits

- Performance increase
- Energy saving
- Cost reduction
- Risk avoidance / security

