

Occupant-Centric Grid- Interactive Buildings

4. Machine Learning III

CE397
Spring 2024

Prof. Dr. Zoltan Nagy

Tentative Course Outline / Schedule

Week	Class	Topic	Guest Lecture
1	01/17	Introduction / Overview / Python	
2	01/24	Machine Learning I	
3	01/31	Machine Learning II	
4	02/07	Machine Learning III	Justin Hill (Southern)
5	02/14	Occupant Behavior Modeling	
6	02/21	Occupant Behavior Modeling	Tanya Barham (CEL)
7	02/28	Occupant Behavior Modeling	Jessica Granderson (LBNL)
8	03/06	Occupant Behavior Modeling	Hussain Kazmi (KU Leuven)
9	03/13	Spring Break	
10	03/20	Advanced Control & Calibration	Ankush Chakrabarty (MERL)
11	04/27	Calibration	Donghun Kim (LBNL)
12	04/03	Introduction to CityLearn	
13	04/10	Project Work	Siva Sankaranarayanan (EPRI)
14	04/17	Project work	
15	04/24	Project work	

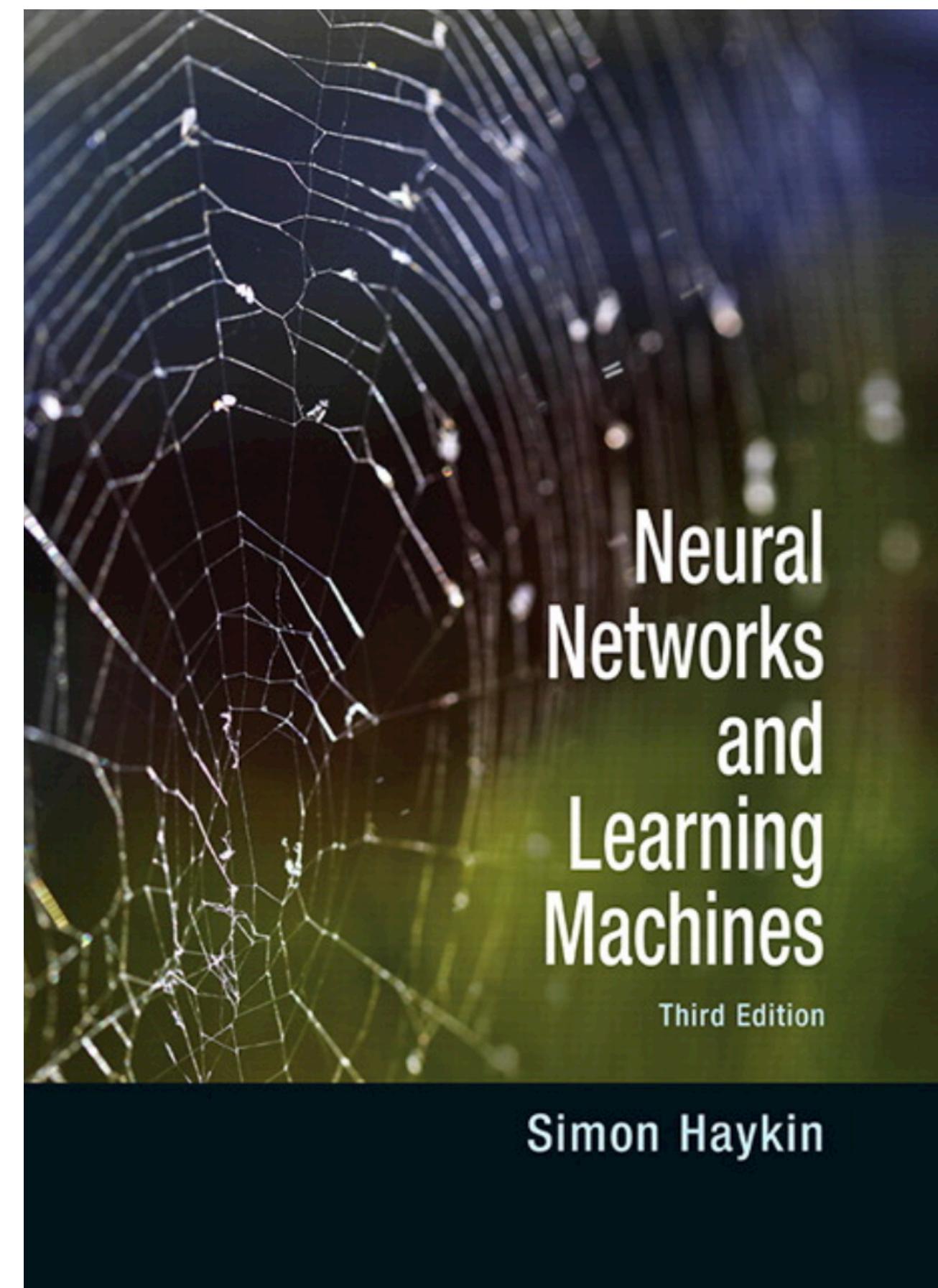
The Plan for Today

- Neural Networks
- Clustering case study
- Homework
- Guest Lecture @10:30 (different zoom link)

Artificial Neural Networks

A neural network is a **massively parallel** distributed processor made up of **simple processing units** that has a natural propensity for storing experimental knowledge and making it available for use. It resembles the brain in two respects:

1. Knowledge is acquired by the network from its environment through a **learning process**
2. Interneuron connection strengths, known as **synaptic weights**, are used to store the acquired knowledge

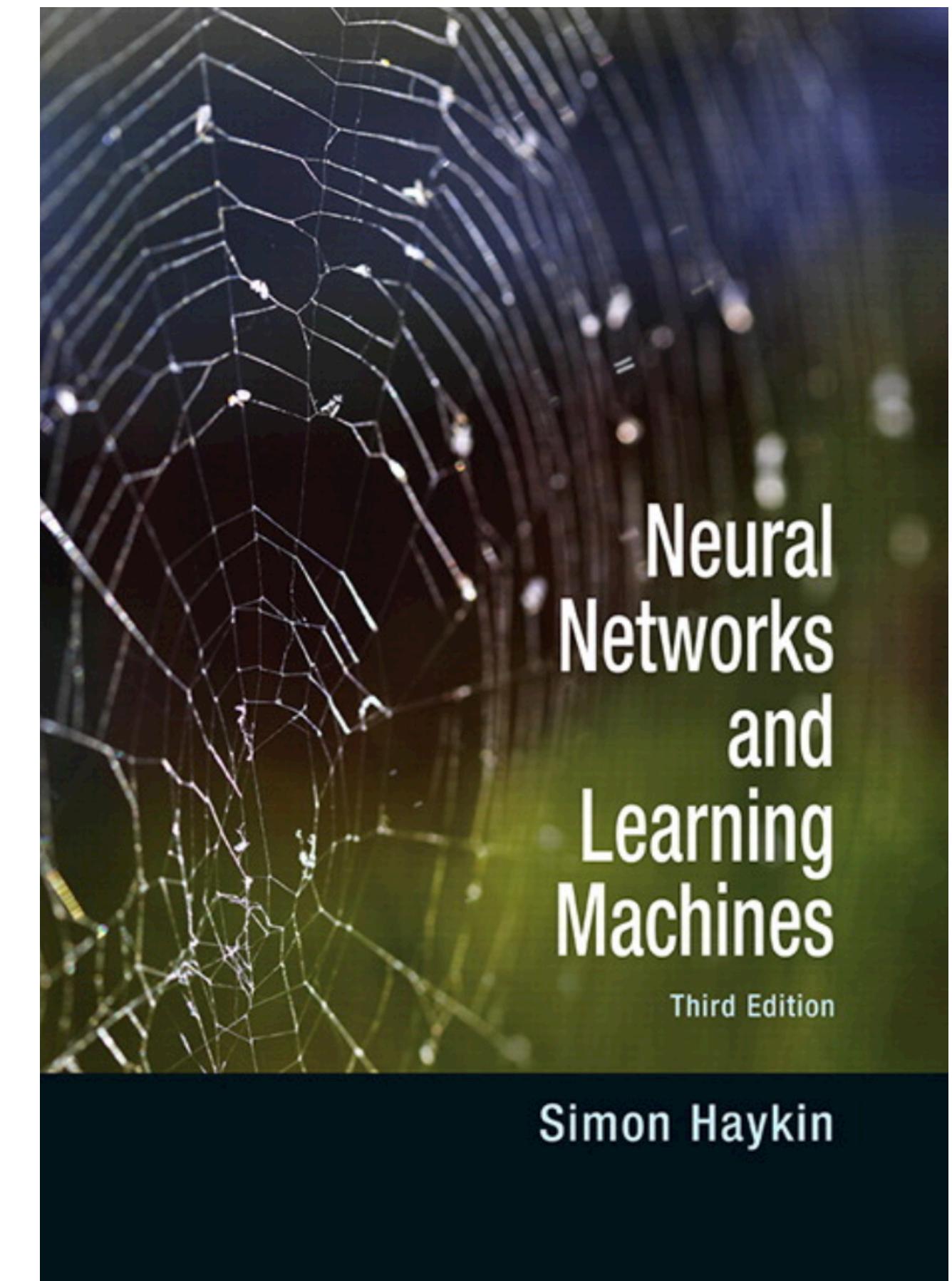


Artificial Neural Networks (ANN)

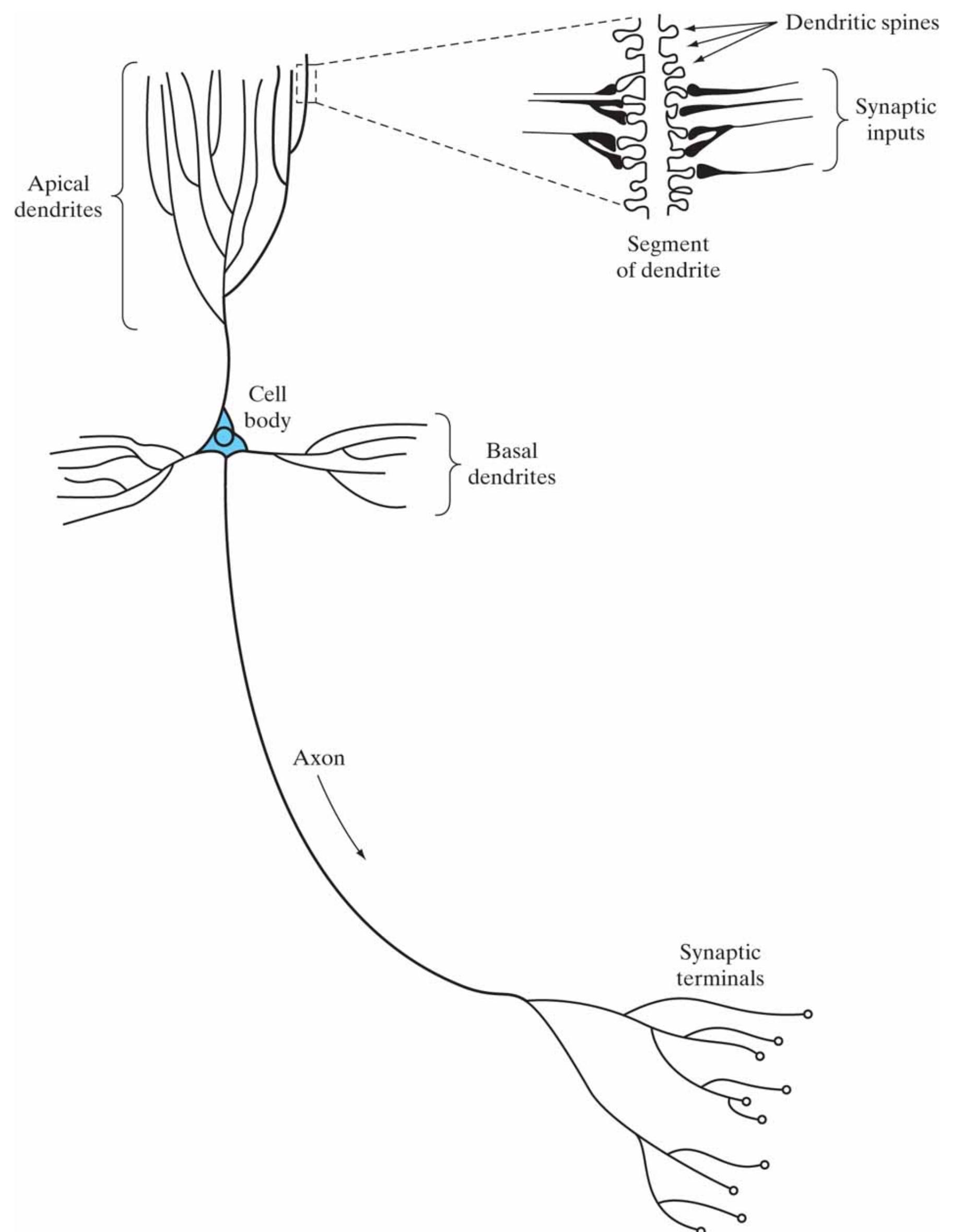
A neural network is a **massively parallel** distributed processor made up of **simple processing units** that has a natural propensity for storing experimental knowledge and making it available for use. It resembles the brain in two respects:

1. Knowledge is acquired by the network from its environment through a **learning process**
2. Interneuron connection strengths, known as **synaptic weights**, are used to store the acquired knowledge

*BUT: an ANN is **NOT** a model for a brain and vice versa.*

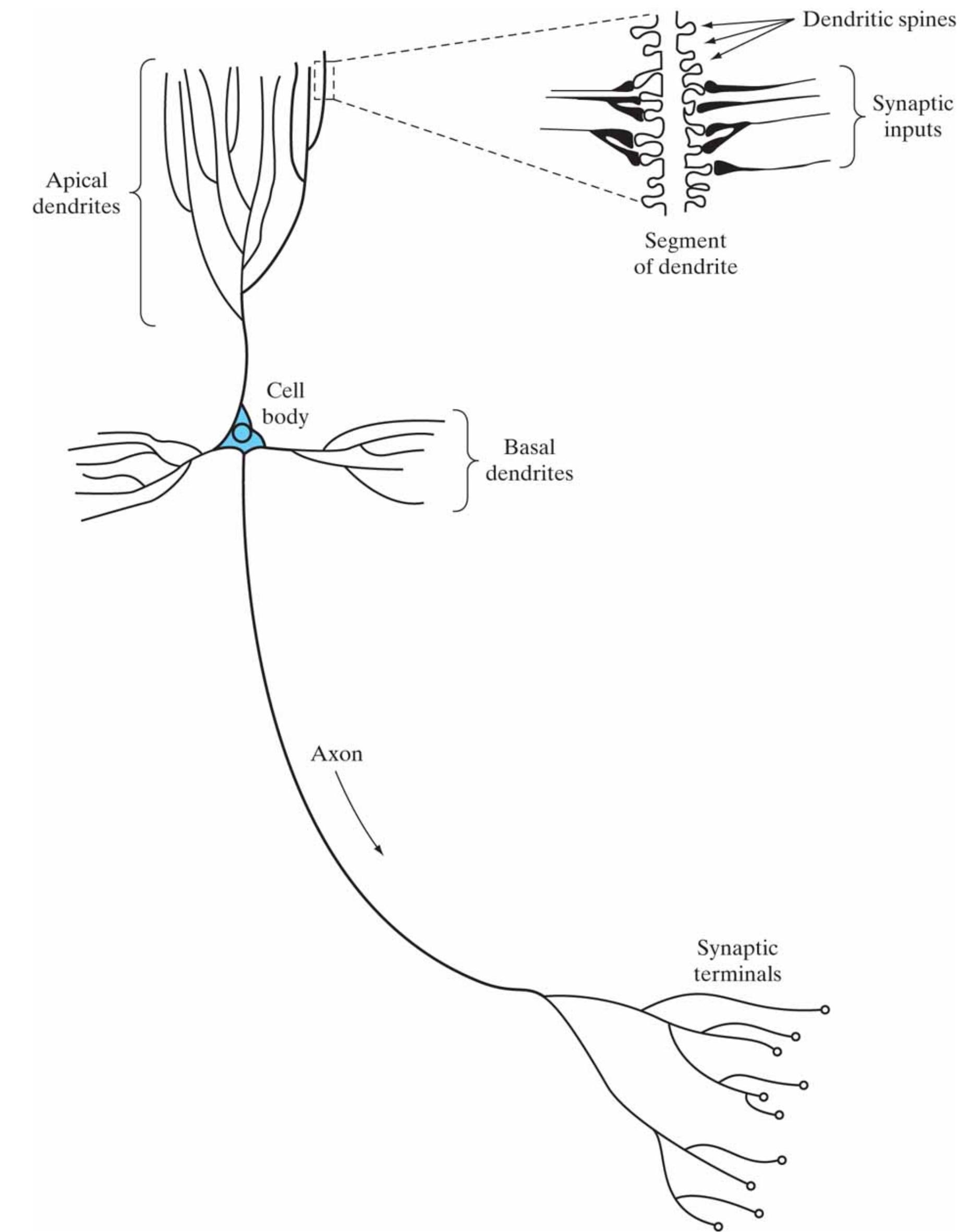
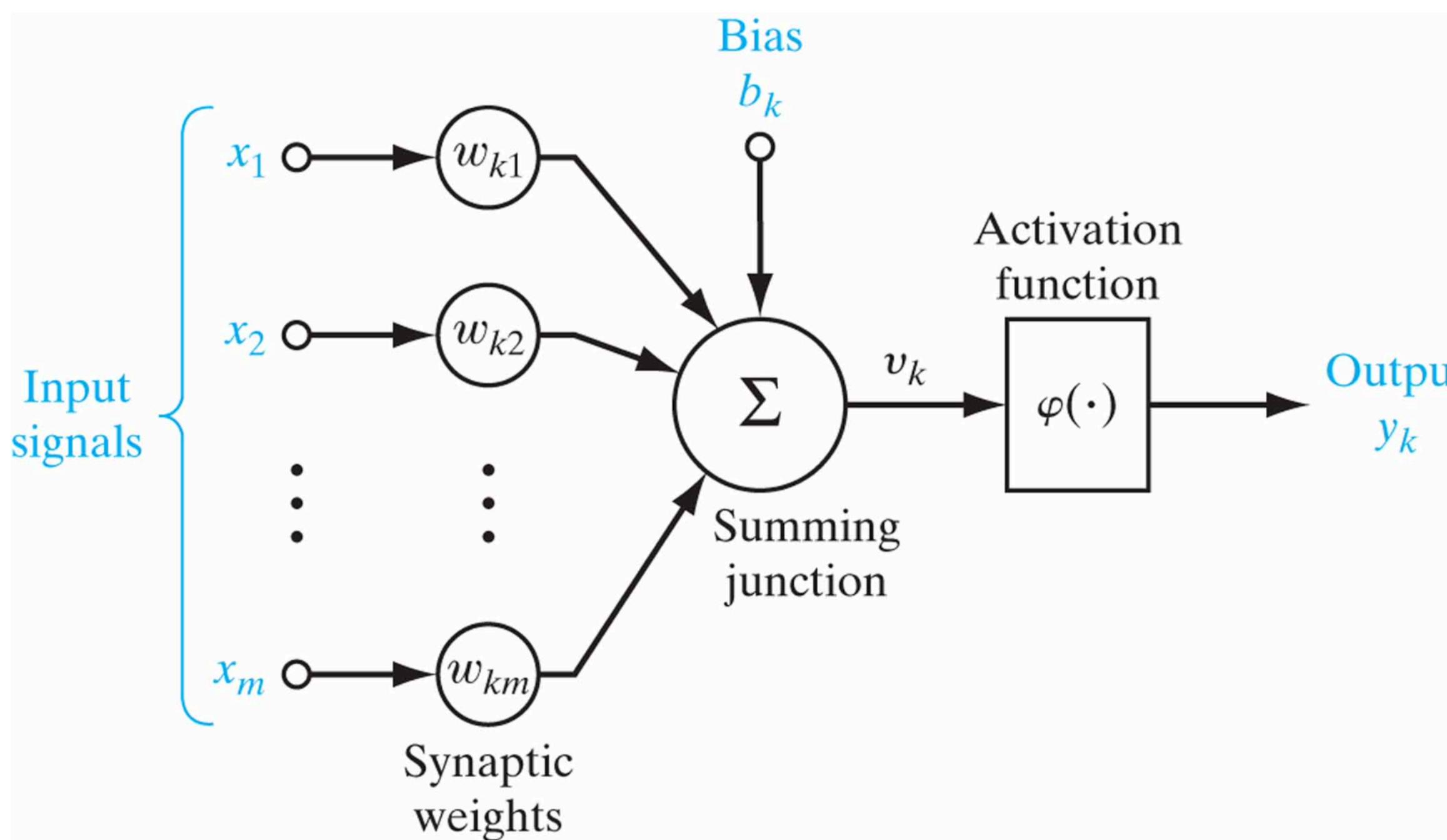


Biological Neural as Inspiration



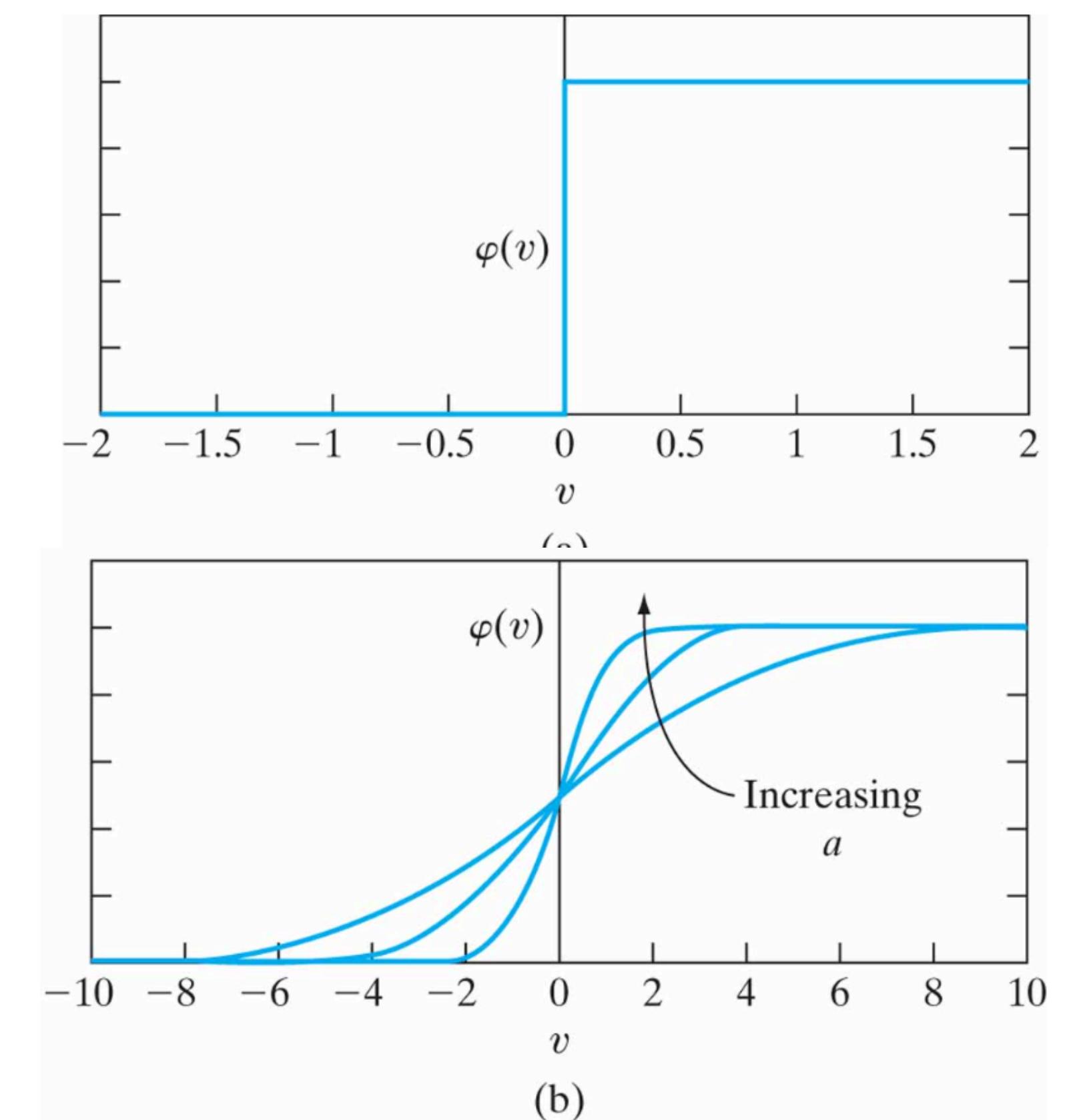
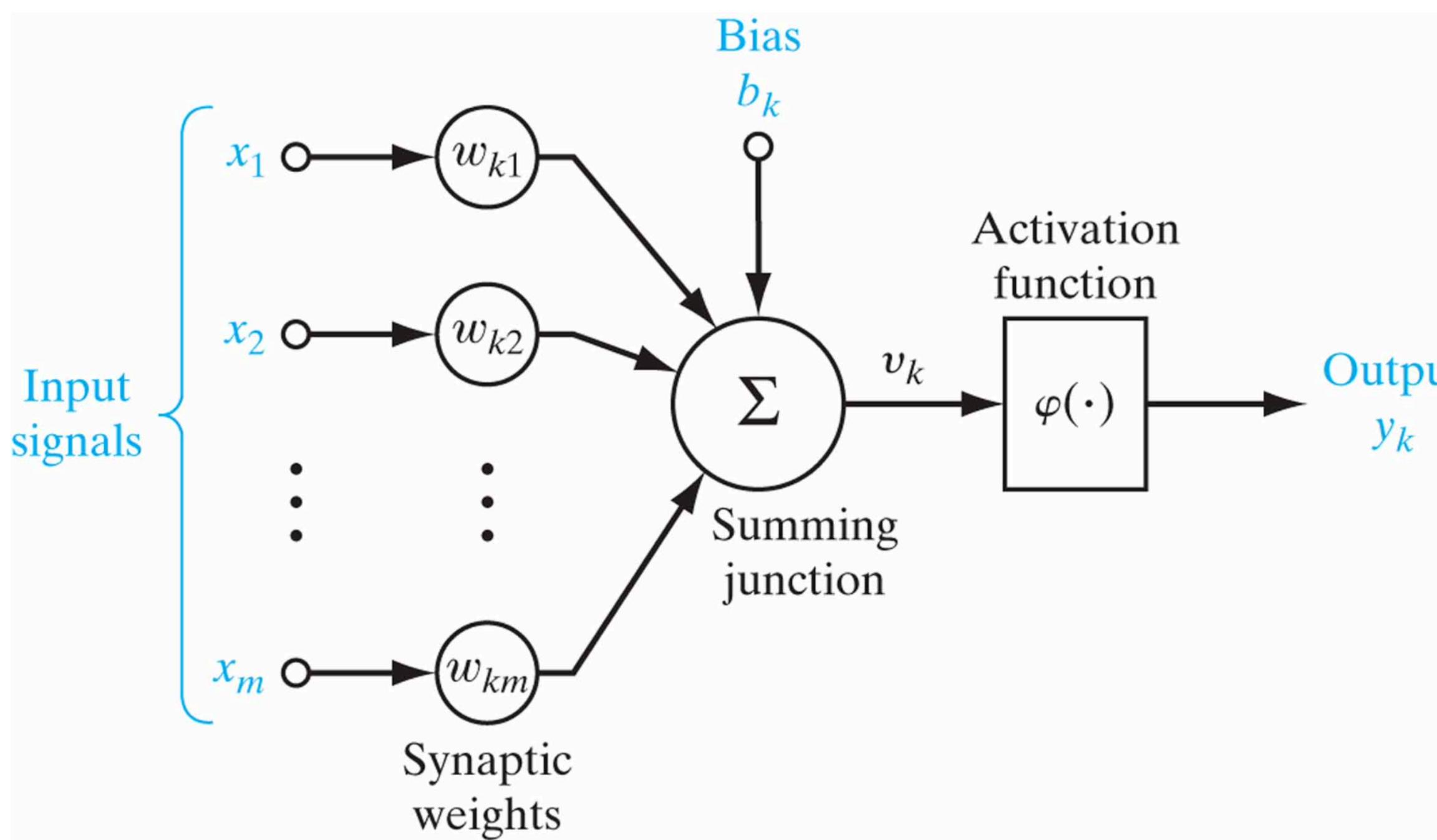
Artificial Neuron

- Received 1 or more inputs (dendrites) and sums inputs to form output (axon)
- Sums are weighted
- Sum is passed through a non-linear (activation) function



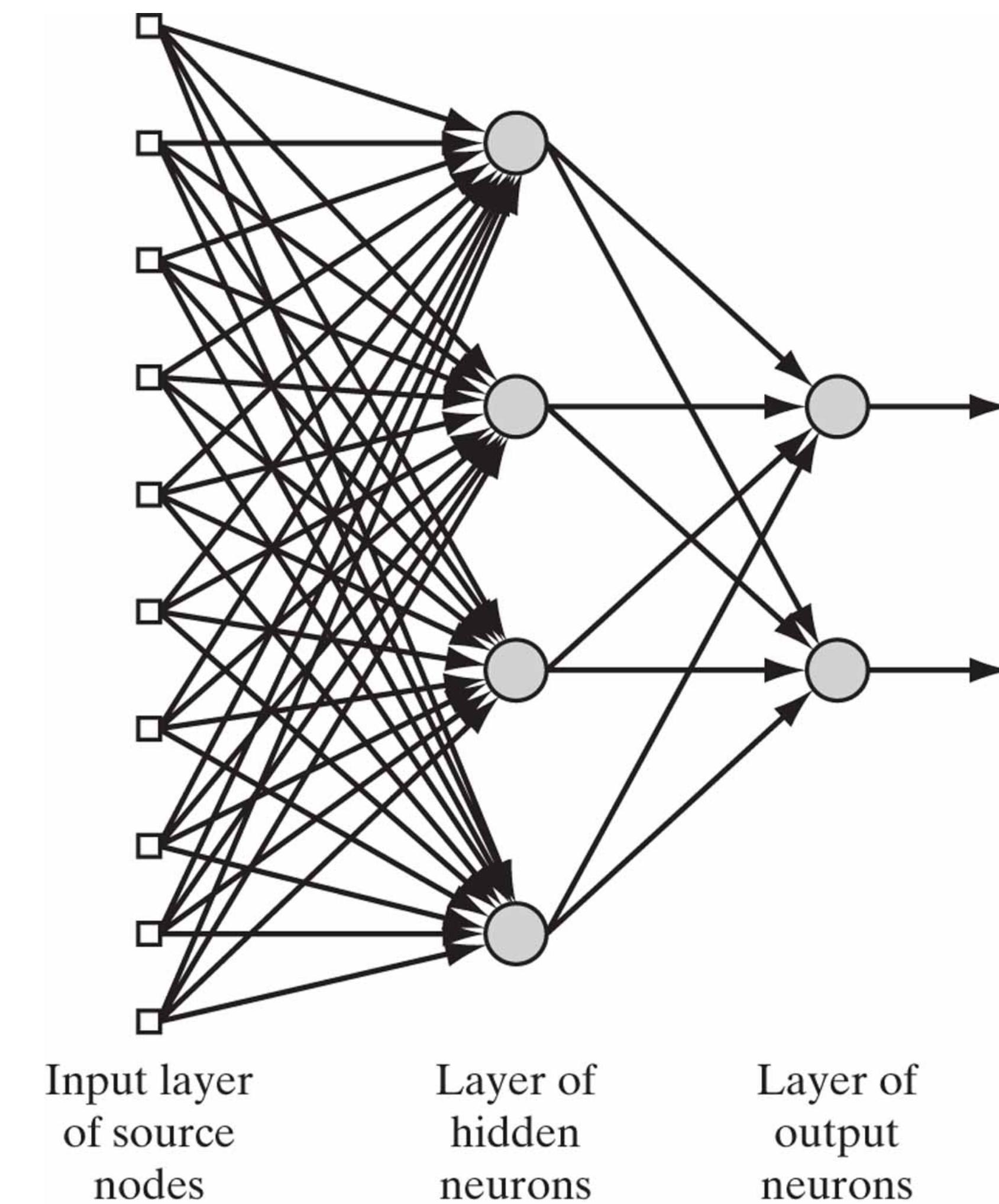
Artificial Neuron

- Received 1 or more inputs (dendrites) and sums inputs to form output (axon)
- Sums are weighted
- Sum is passed through a non-linear (activation) function



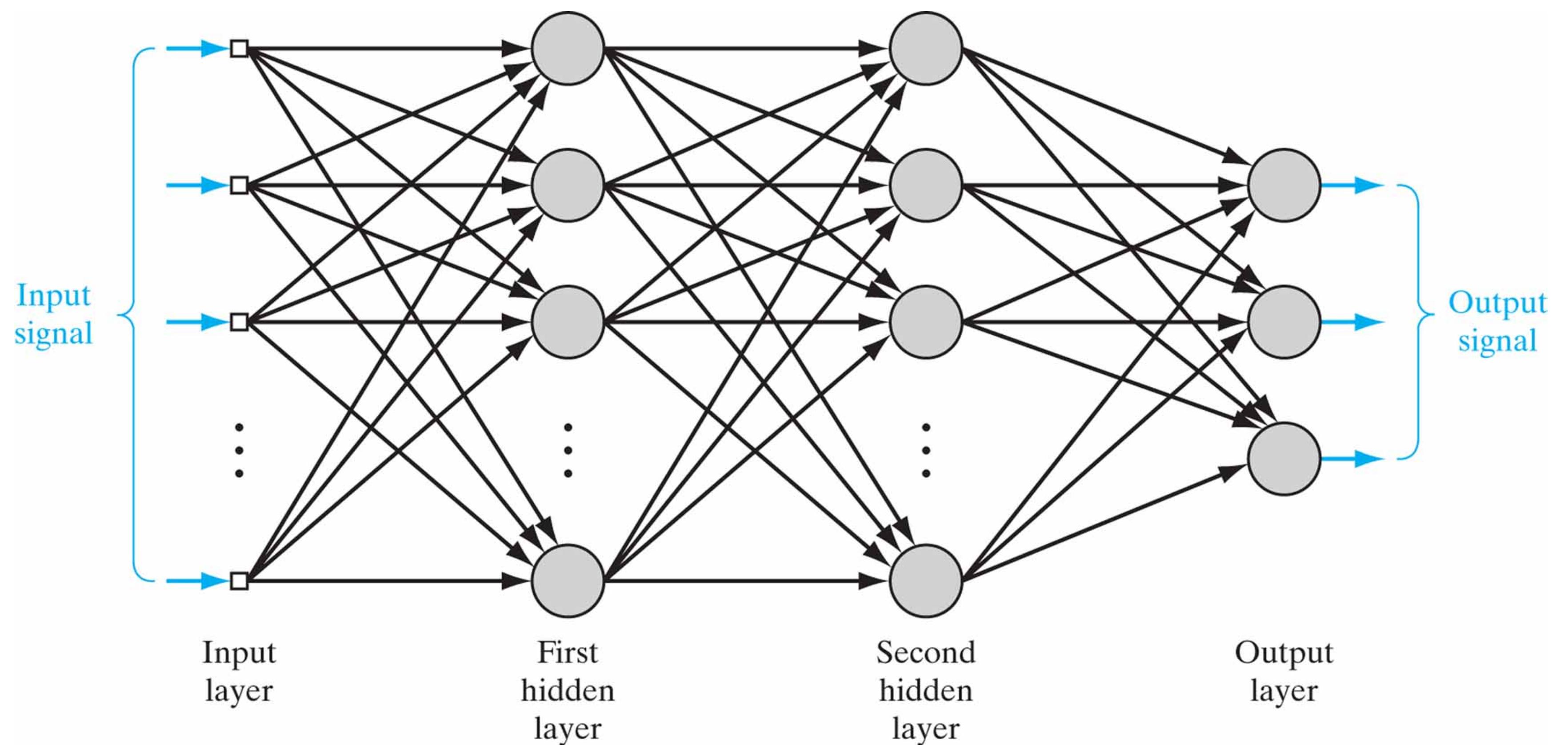
Putting it together: Feed Forward Neural Network

- Input Layer
- Hidden Layer (s)
- Output Layer



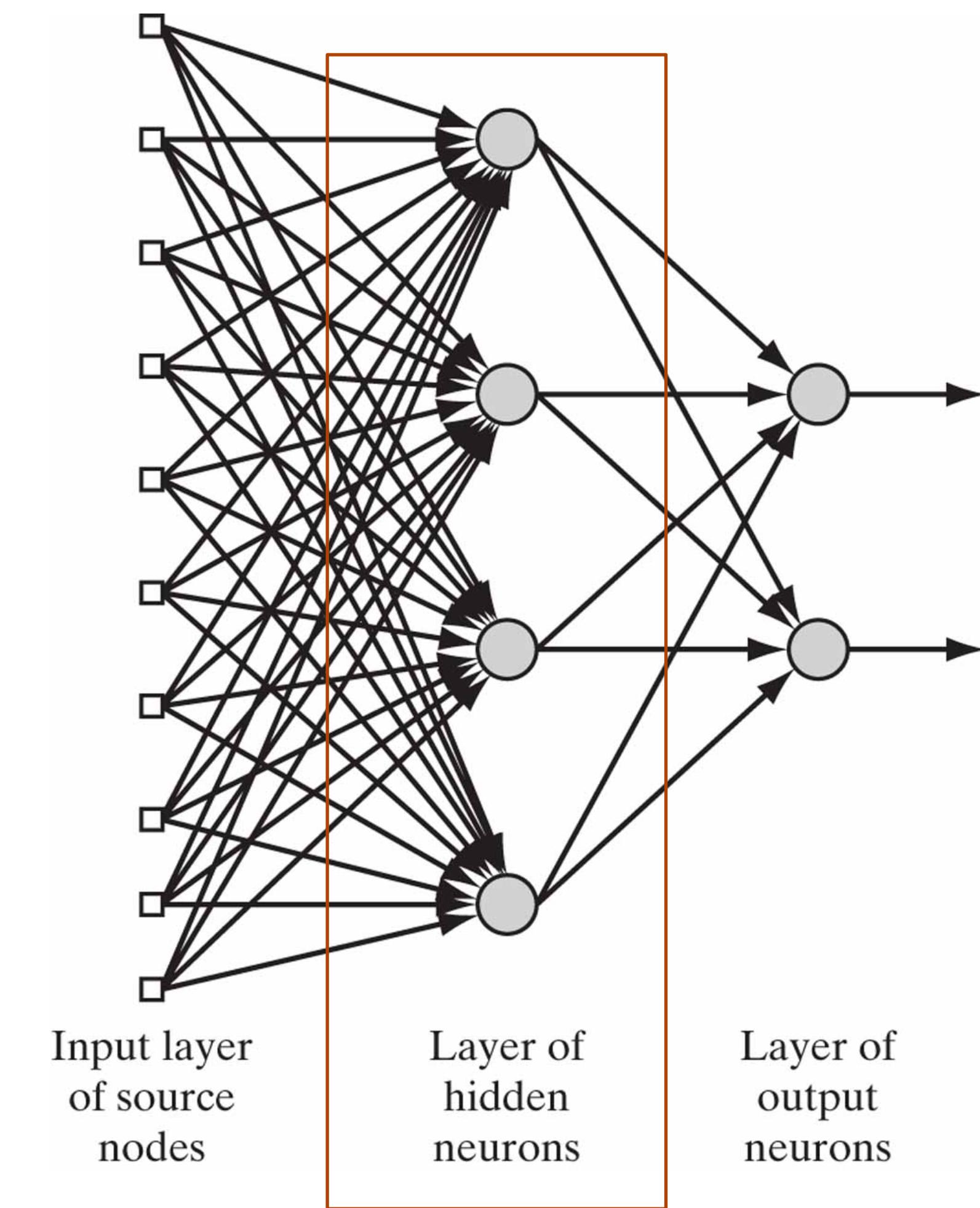
Putting it together: Feed Forward Neural Network

- Input Layer
- Hidden Layer (s)
- Output Layer

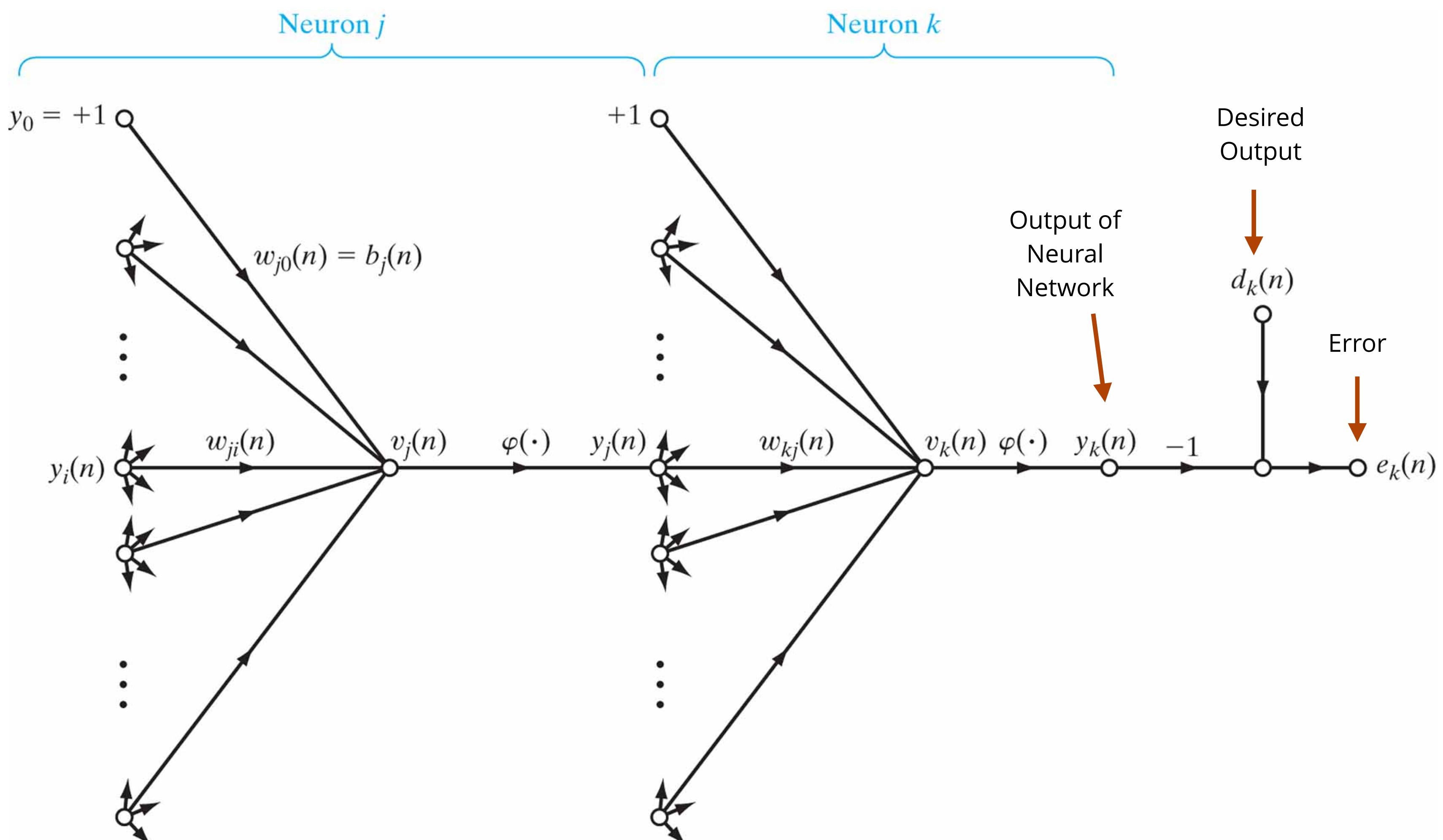


Hidden Layer = Feature Detectors

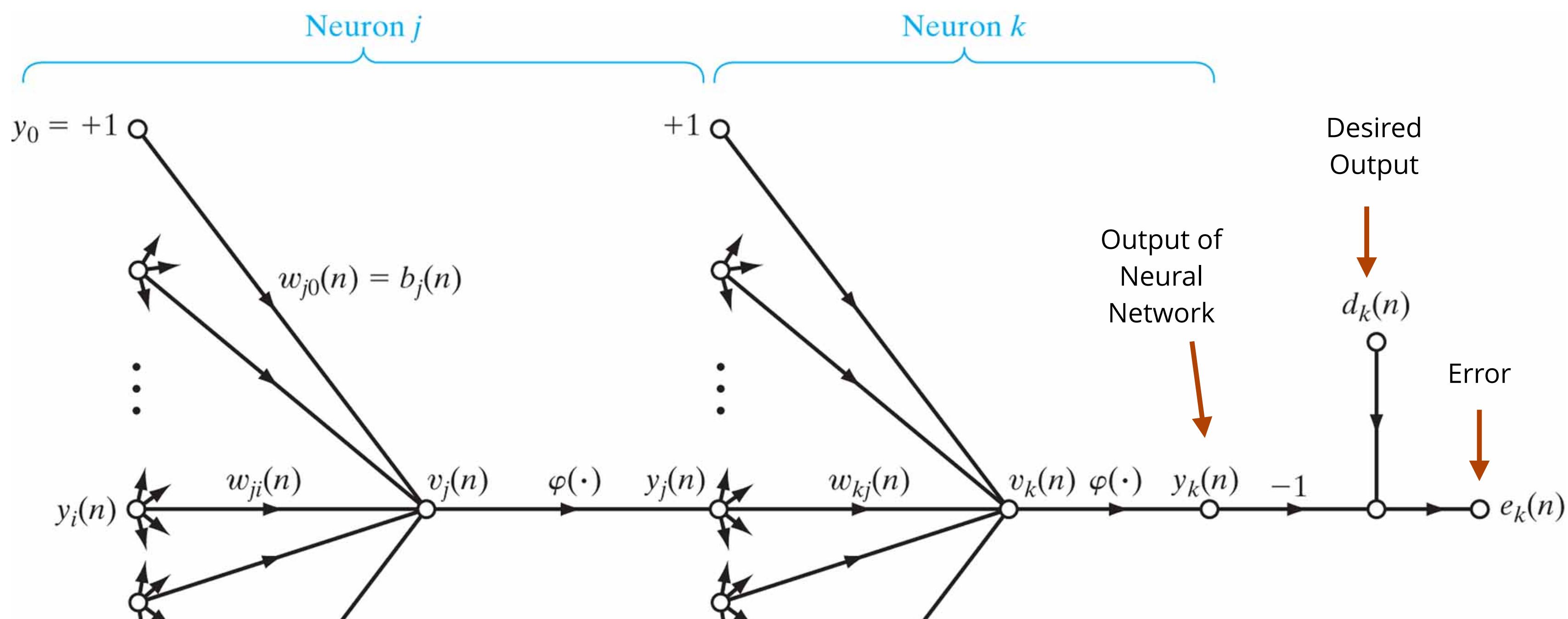
- Non-linear transformation of input data to feature space
- In feature space, classes of interest may be easier separable from each other



Backpropagation Algorithm



Backpropagation Algorithm

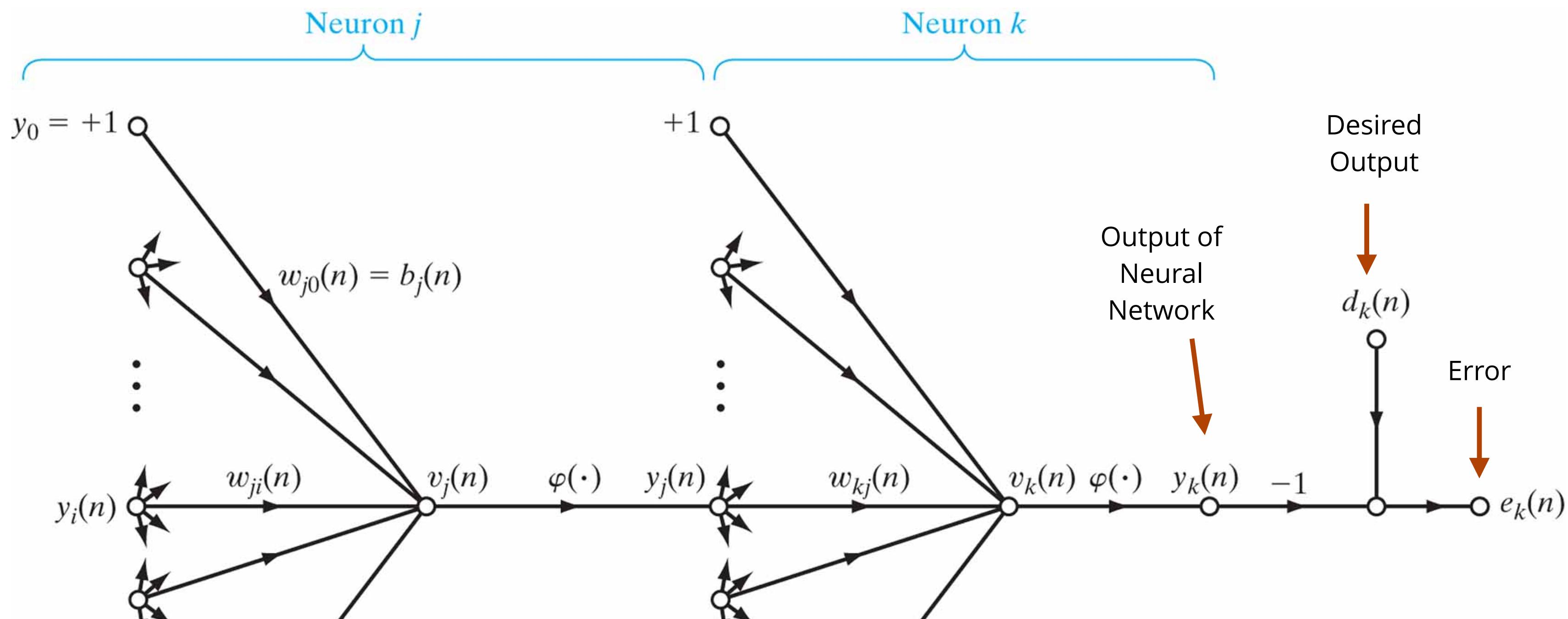


Compute Local Gradients

$$\delta_j^{(l)}(n) = \begin{cases} e_j^{(L)}(n)\varphi'_j(v_j^{(L)}(n)) & \text{for neuron } j \text{ in output layer } L \\ \varphi'_j(v_j^{(l)}(n)) \sum_k \delta_k^{(l+1)}(n)w_{kj}^{(l+1)}(n) & \text{for neuron } j \text{ in hidden layer } l \end{cases}$$



Backpropagation Algorithm



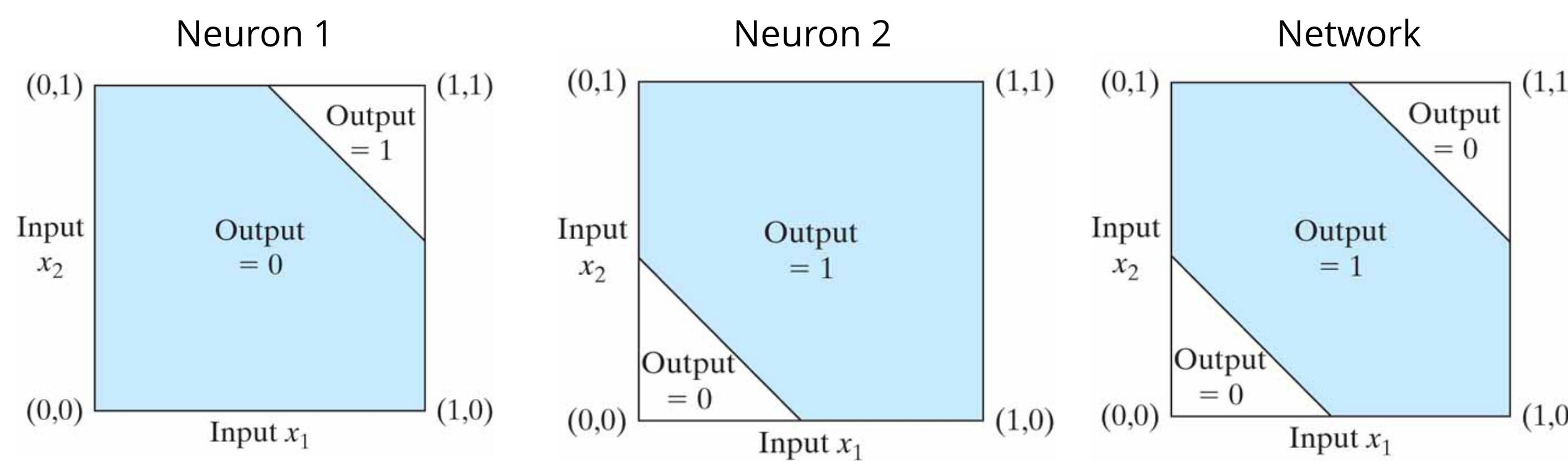
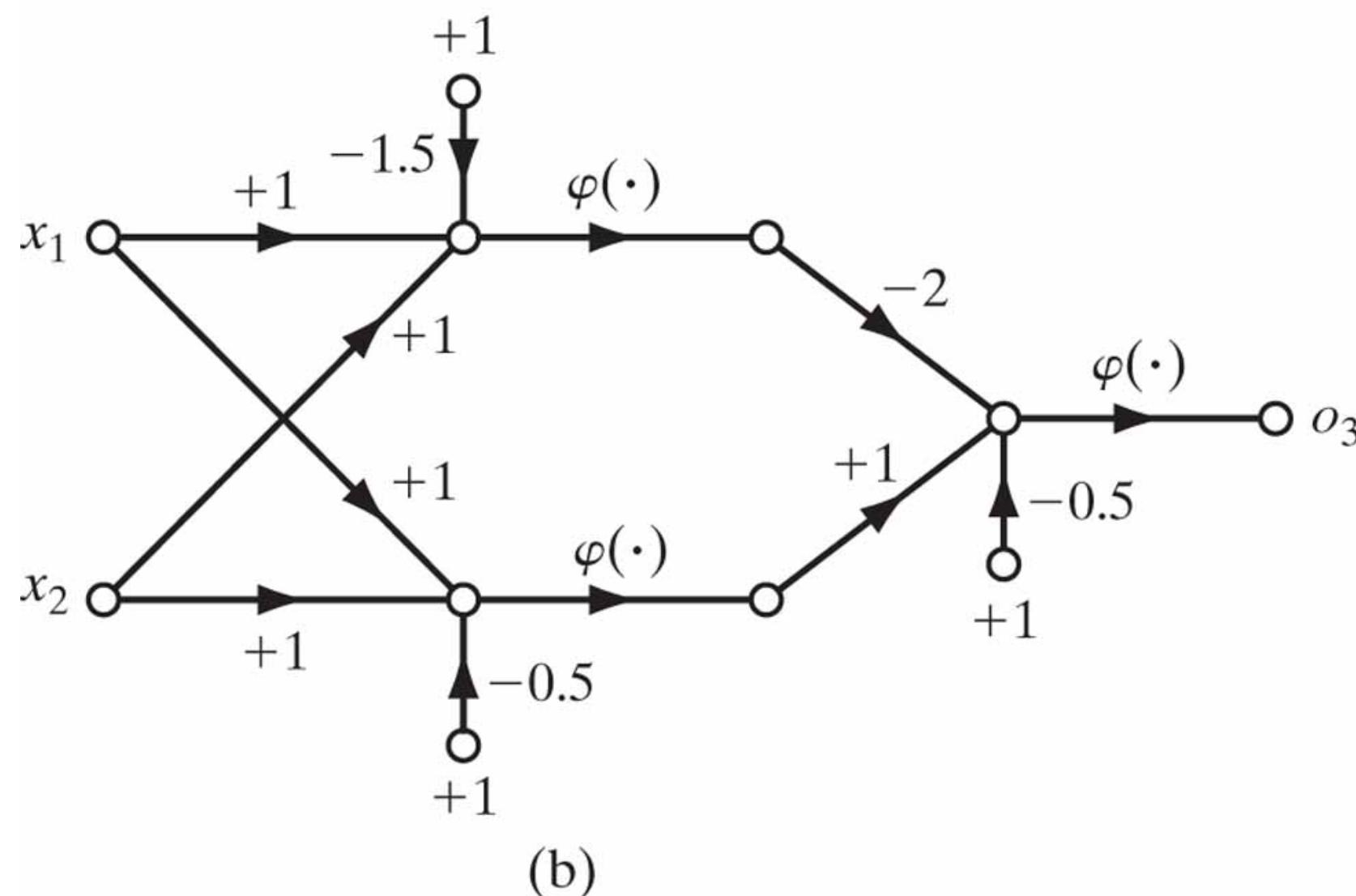
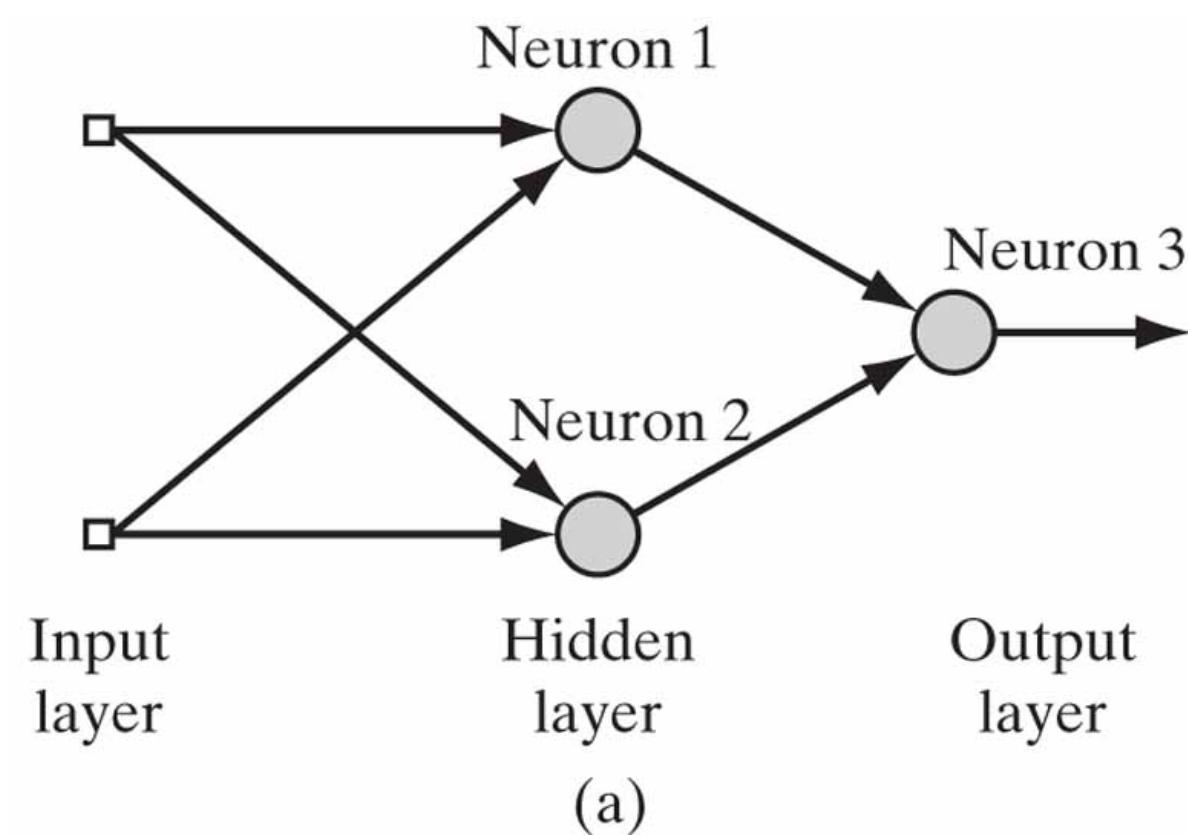
Compute
Local
Gradients

$$\delta_j^{(l)}(n) = \begin{cases} e_j^{(L)}(n)\varphi'_j(v_j^{(L)}(n)) & \text{for neuron } j \text{ in output layer } L \\ \varphi'_j(v_j^{(l)}(n)) \sum_k \delta_k^{(l+1)}(n)w_{kj}^{(l+1)}(n) & \text{for neuron } j \text{ in hidden layer } l \end{cases}$$

Update
Weights

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha[\Delta w_{ji}^{(l)}(n-1)] + \eta \delta_j^{(l)}(n) y_i^{(l-1)}(n)$$

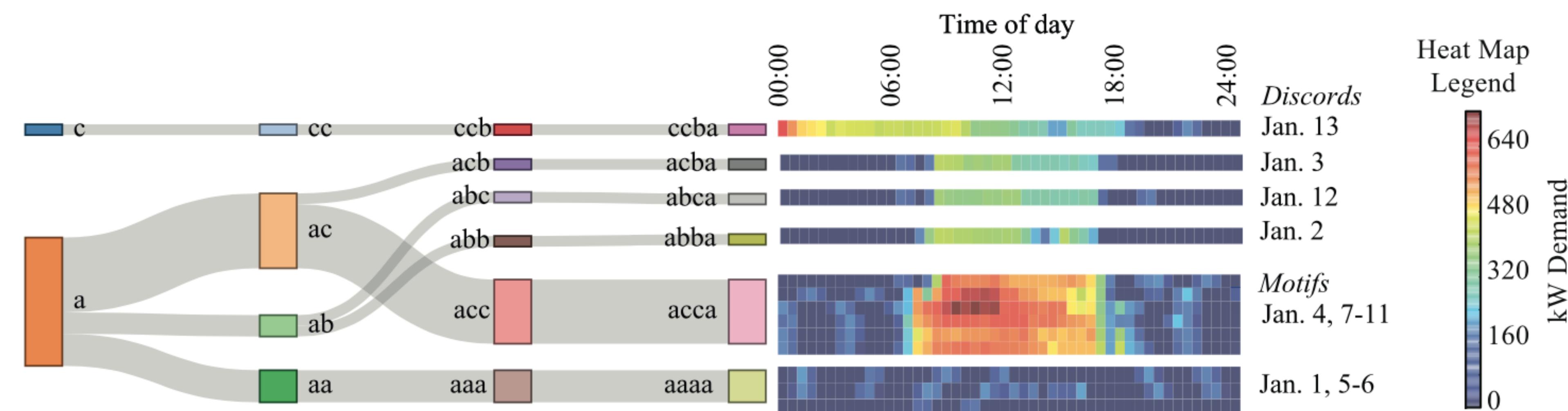
Example (XOR)



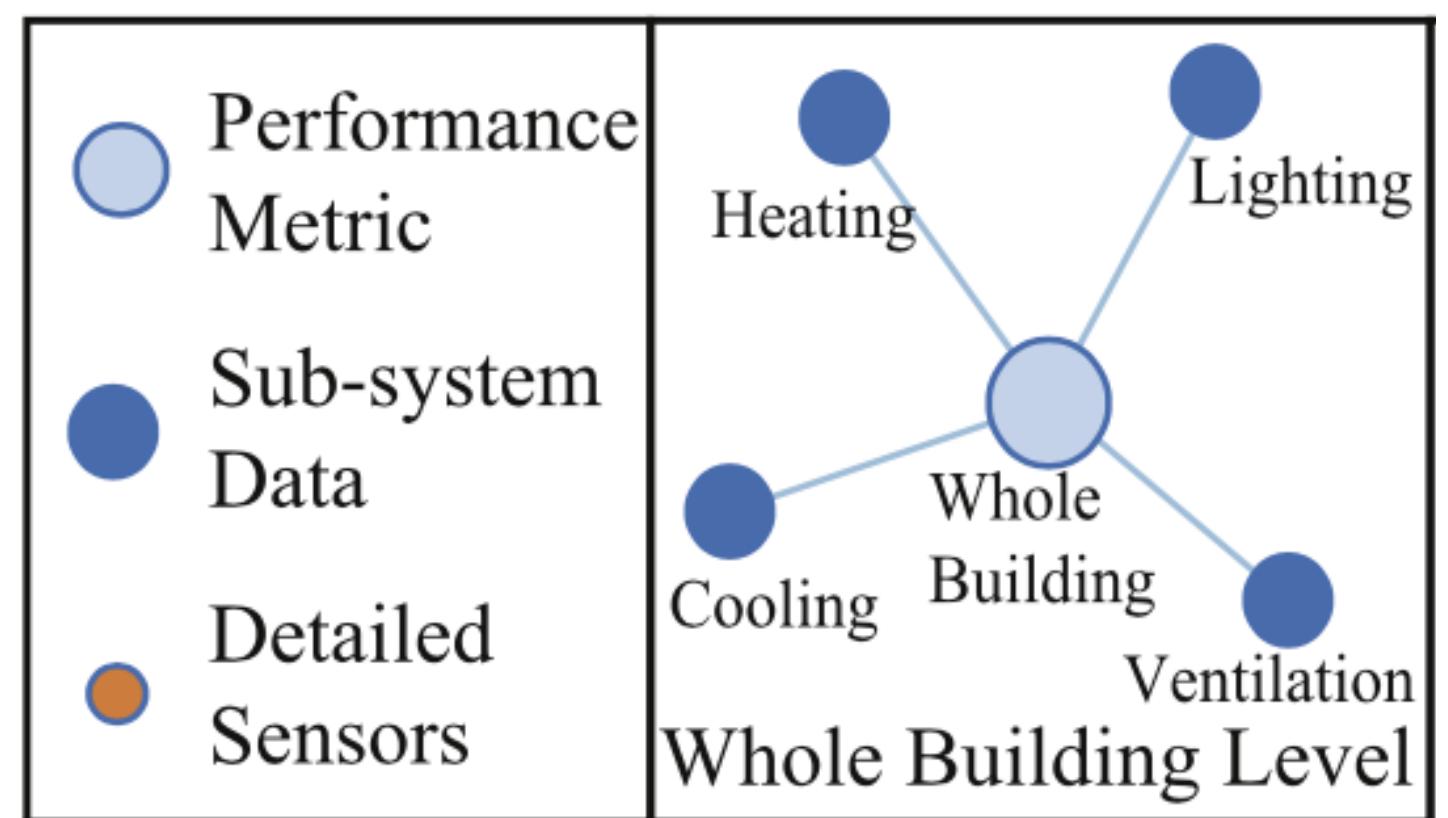
- Assume possible inputs for x_1 and x_2 are only 0 & 1
- Assume *threshold* activation function
- What is the output of the network for each of the possible input combinations (0,0), (0,1), (1,0), (1,1) ?

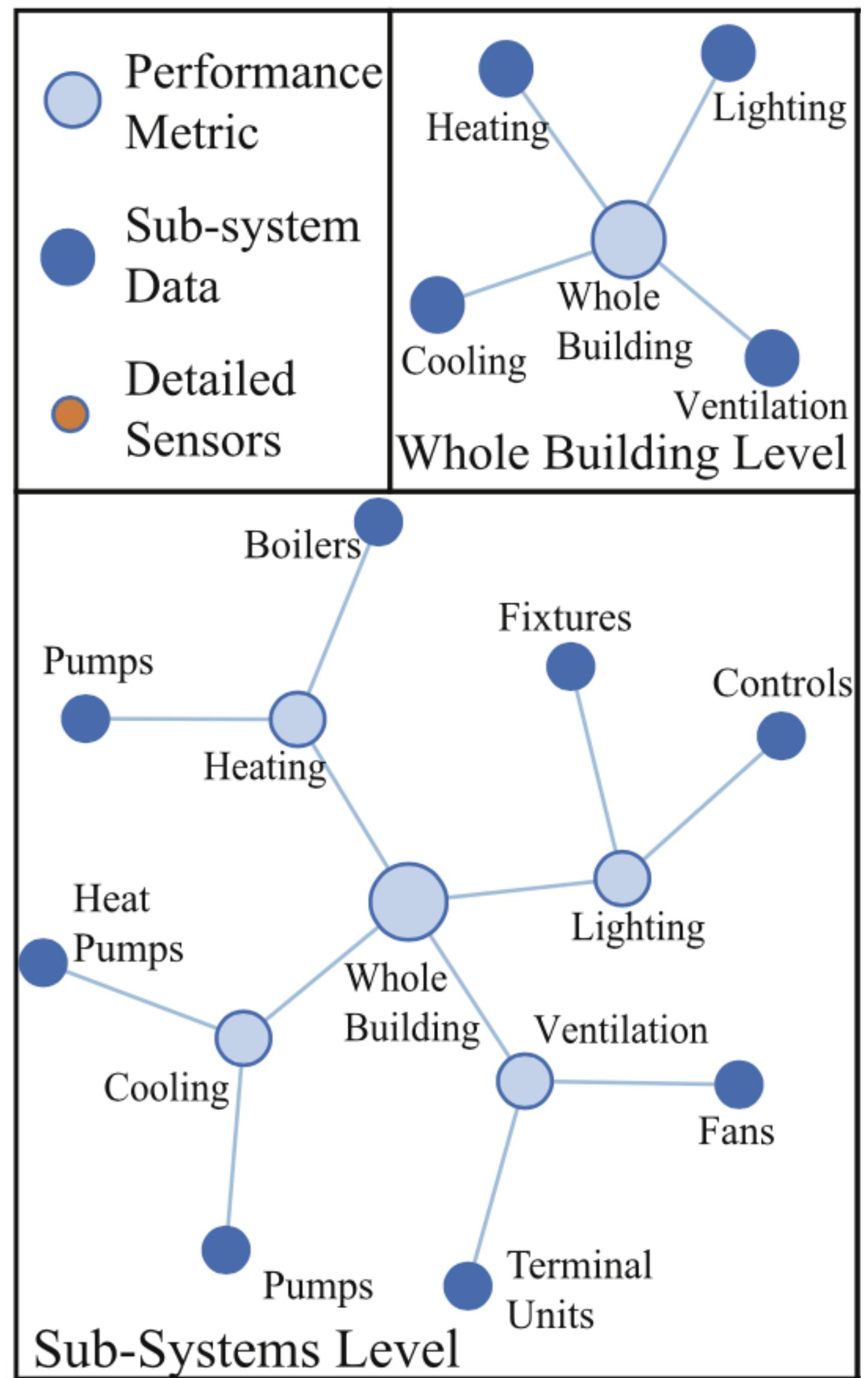
**Radical switch of
topics :)**

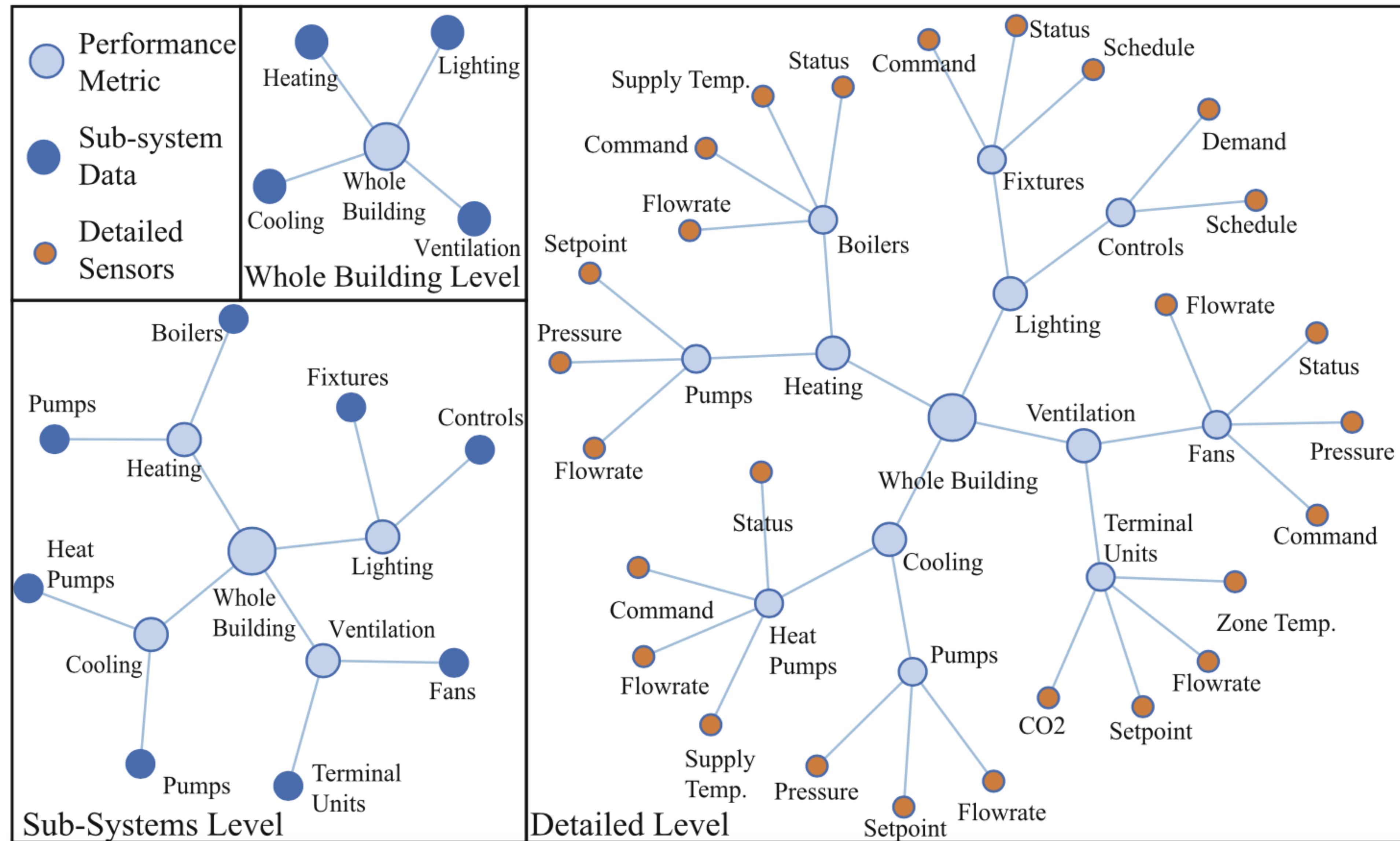
Automated daily pattern filtering of measured building performance data

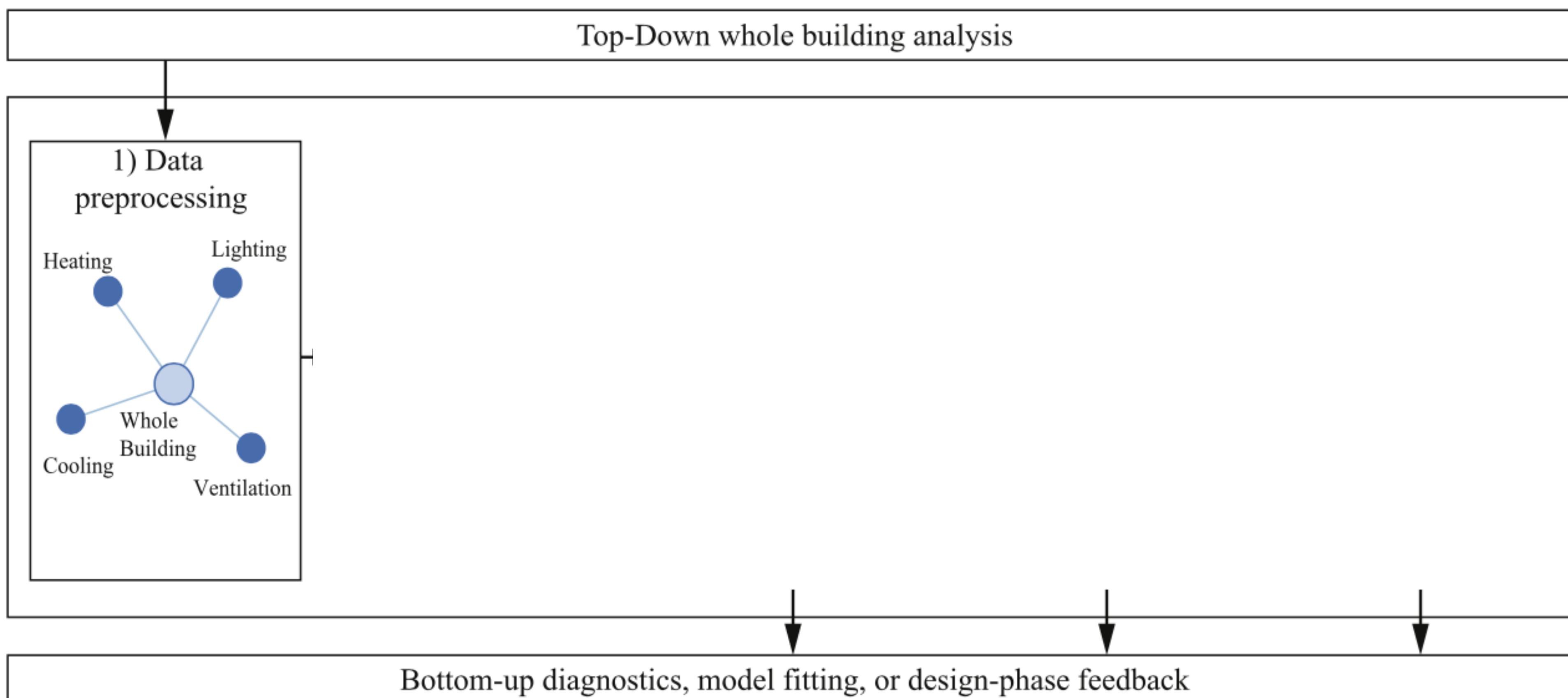


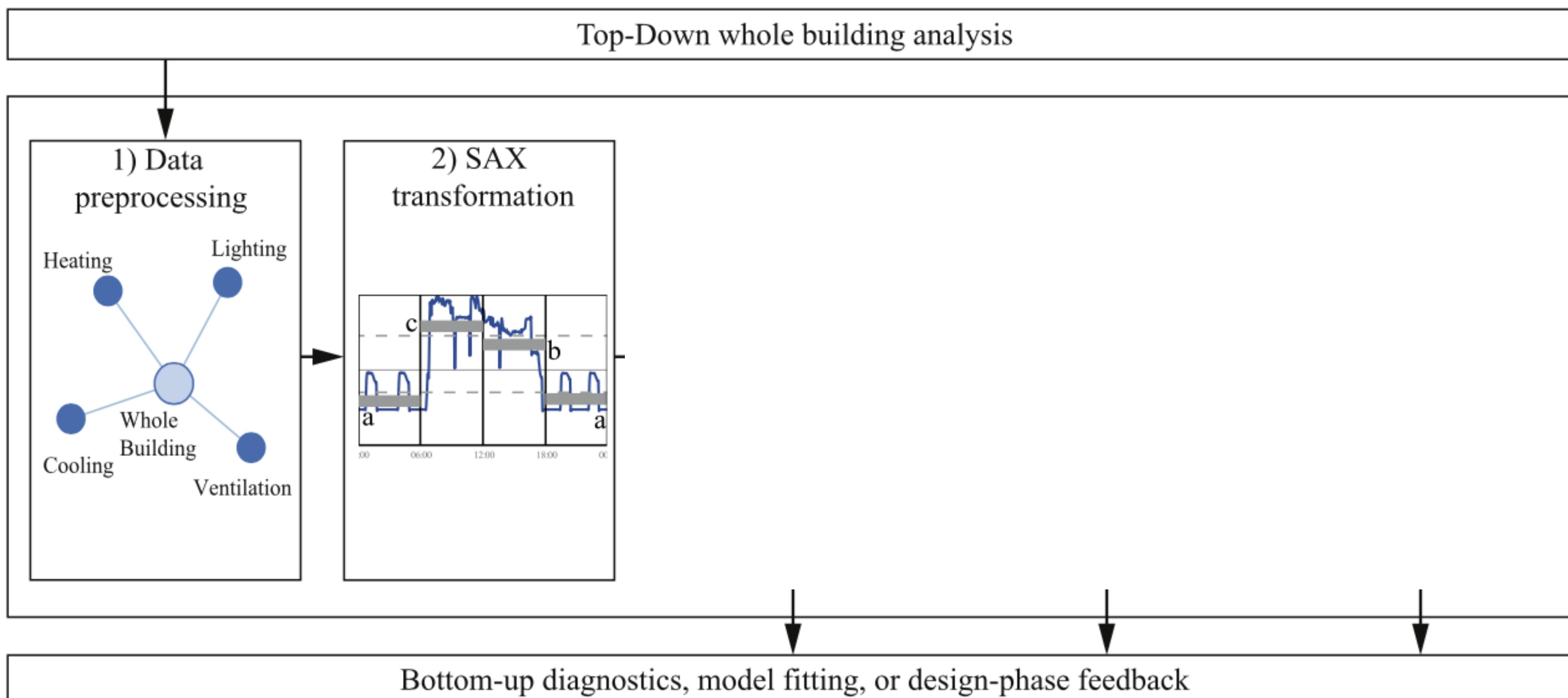
C. Miller, Z. Nagy, and A. Schlueter, *Automated Daily Pattern Filtering of Measured Building Performance Data*, Automation in Construction, vol. 49, no. A, pp. 1–17, Jan. 2014.

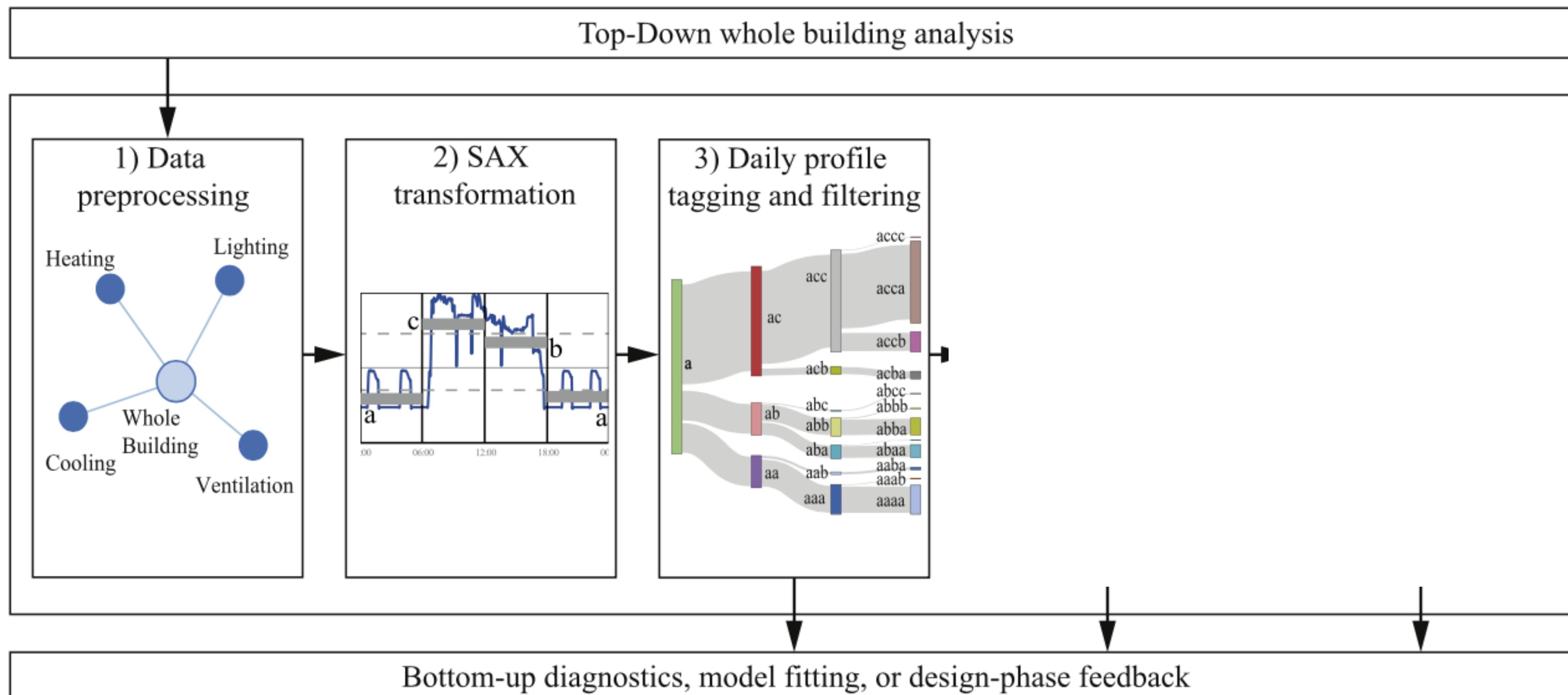


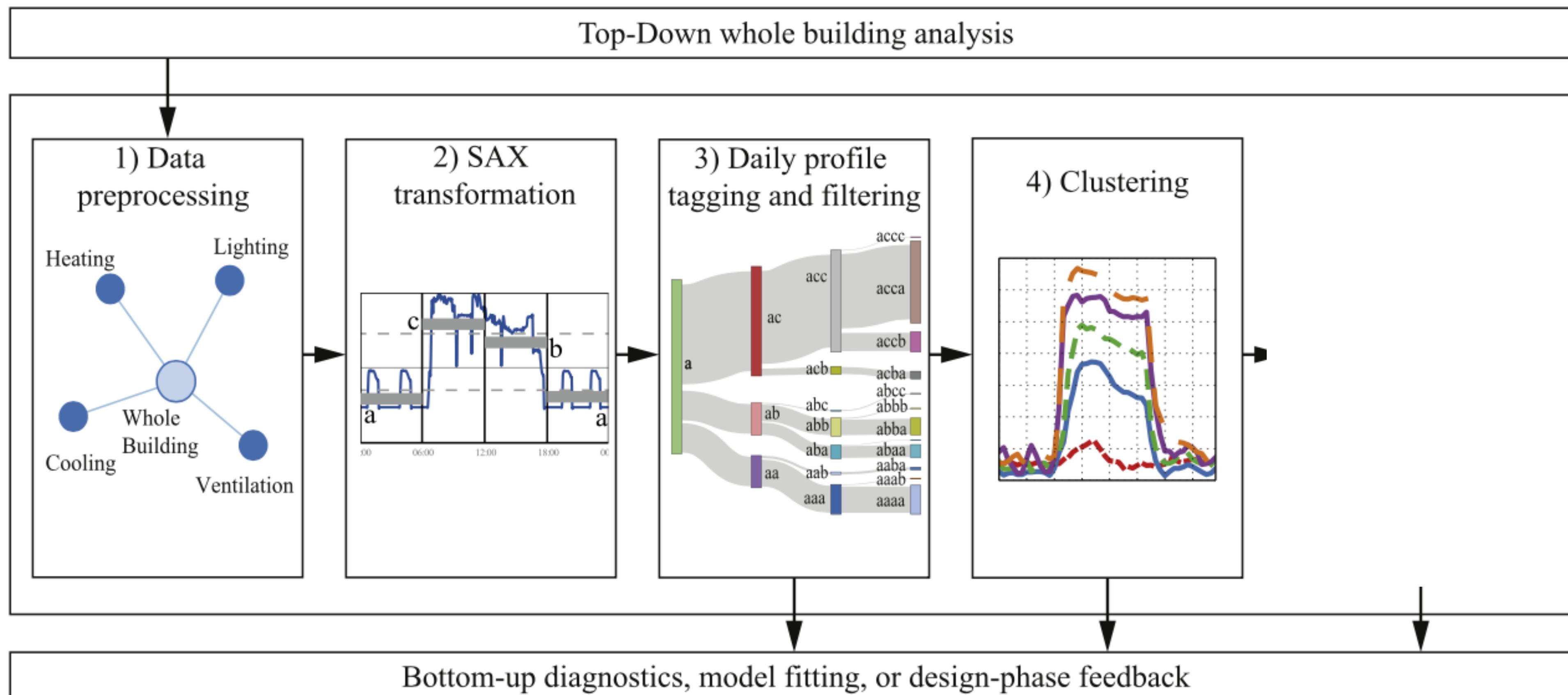


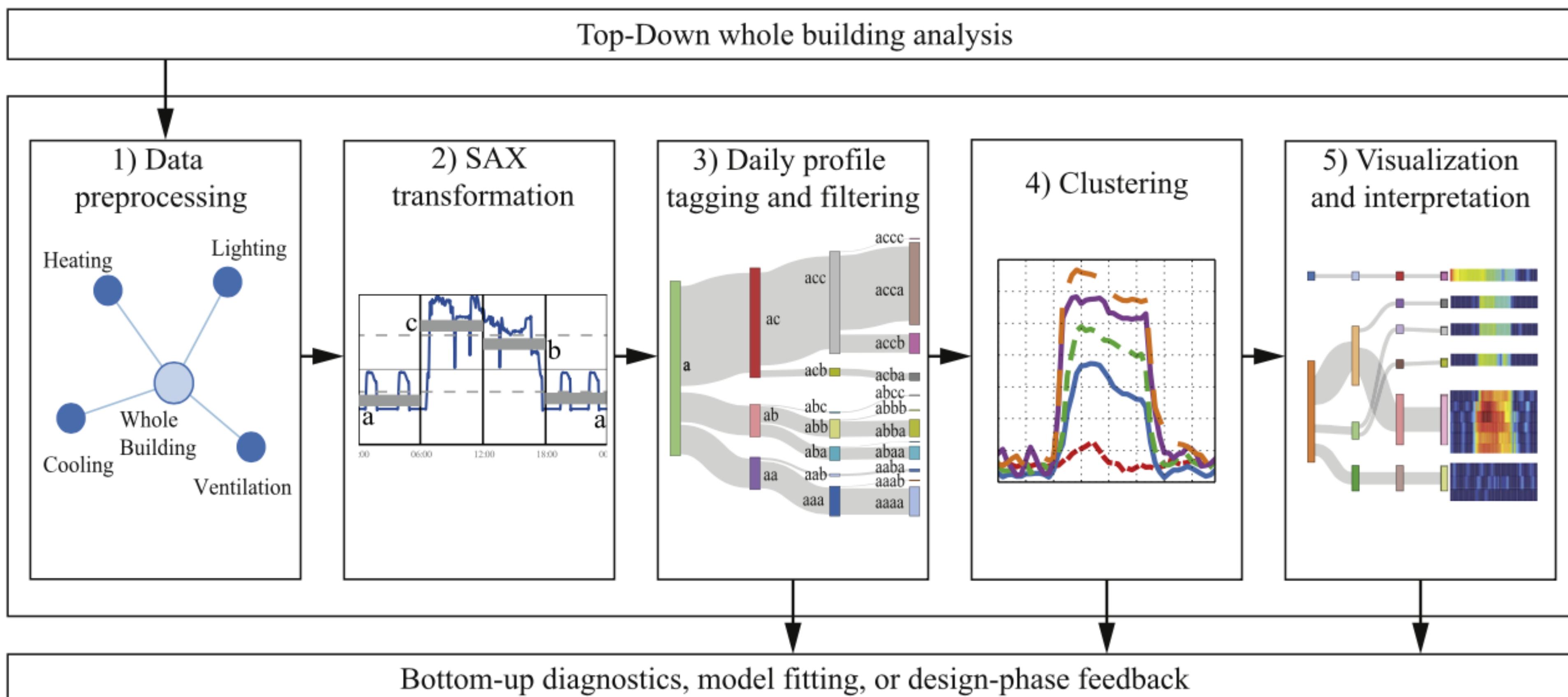






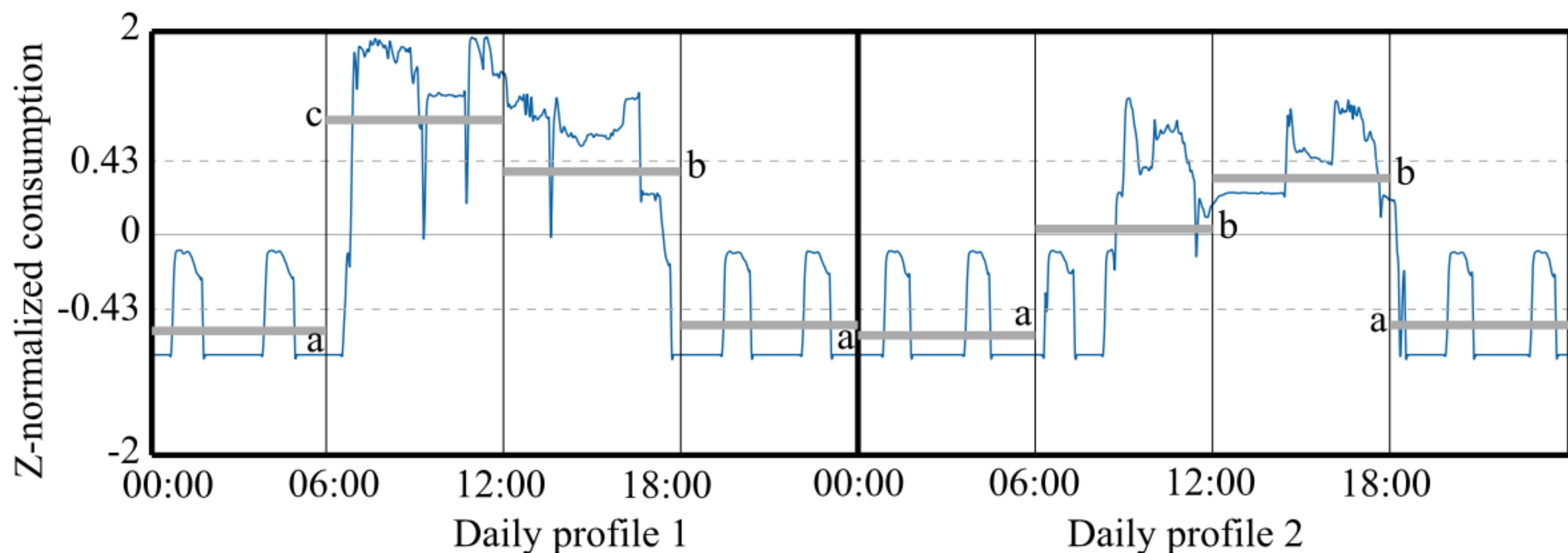






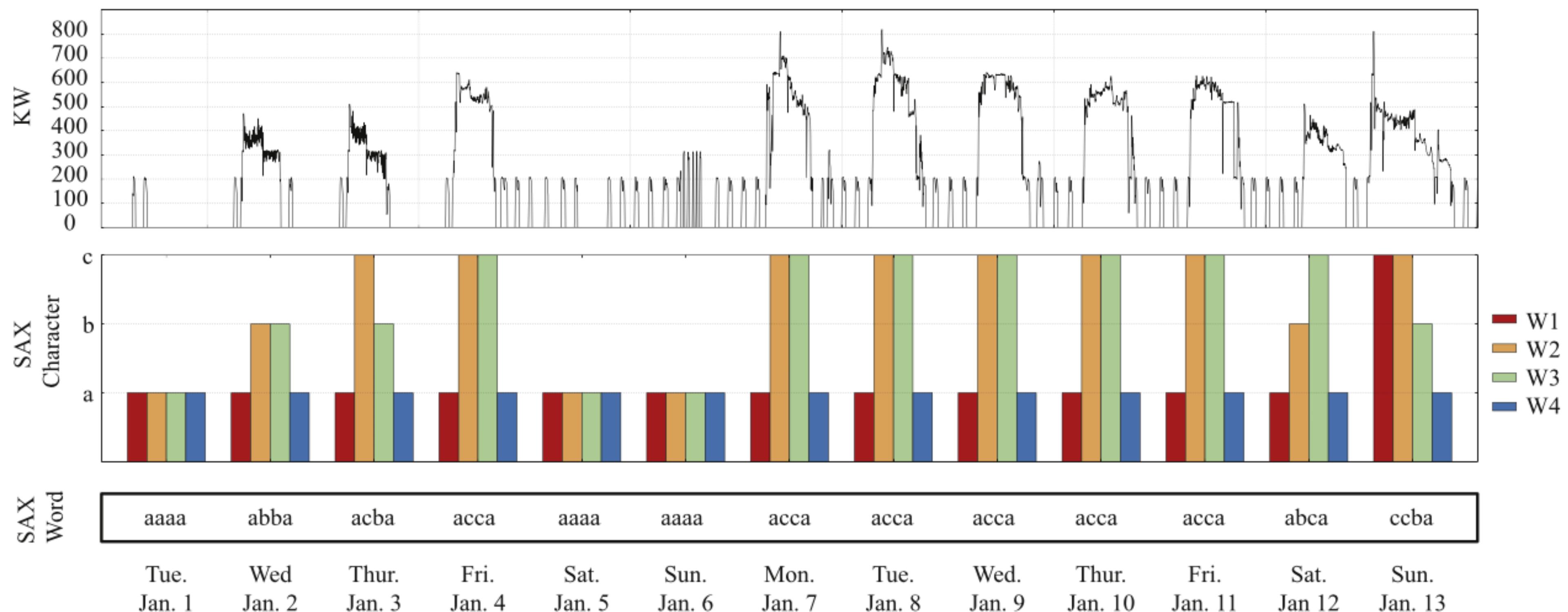
SAX Transformation

- Symbolic Aggregate approXimation

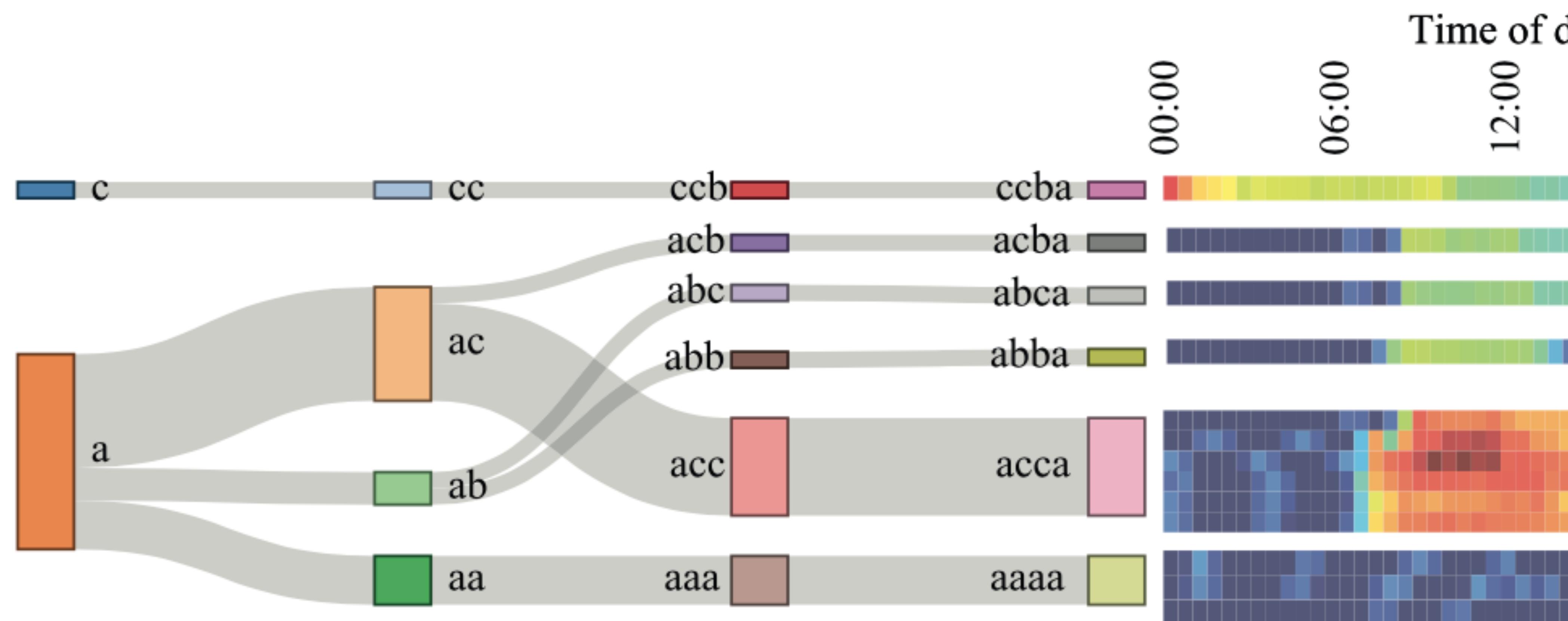


- Why is this a useful idea?
- Why is this a bad idea?

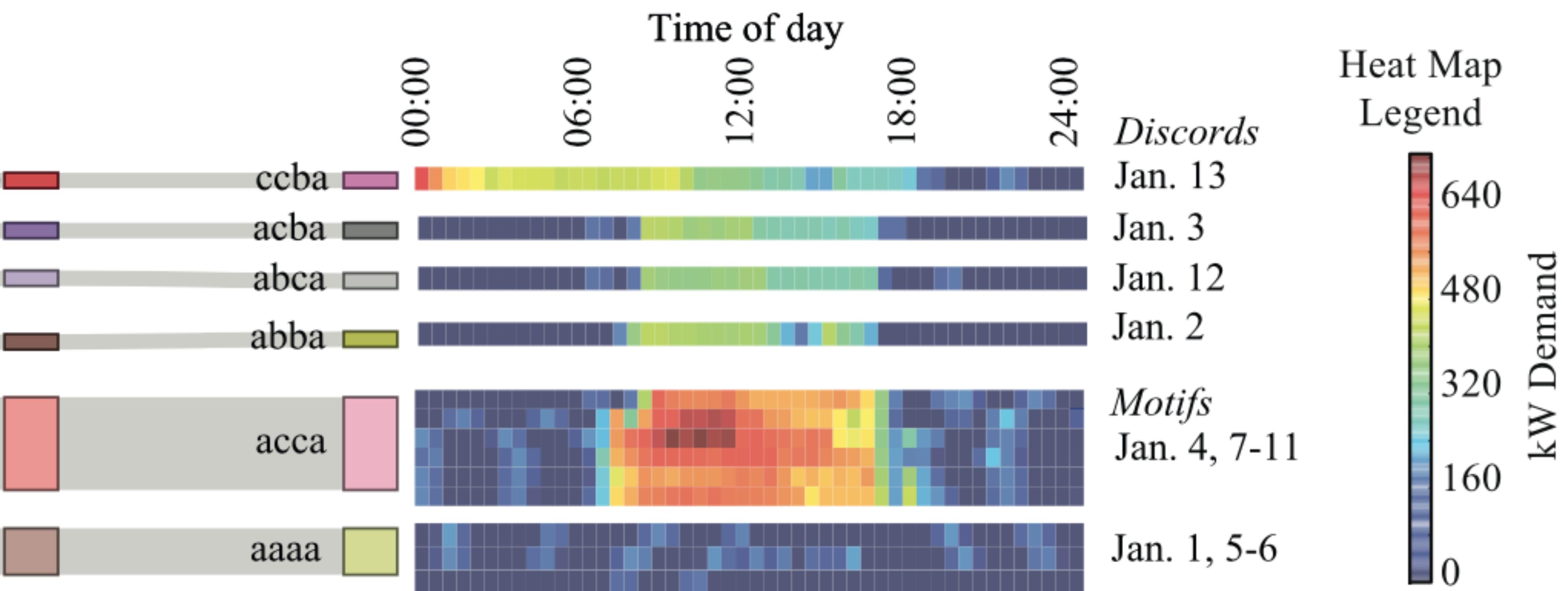
SAX Transformation



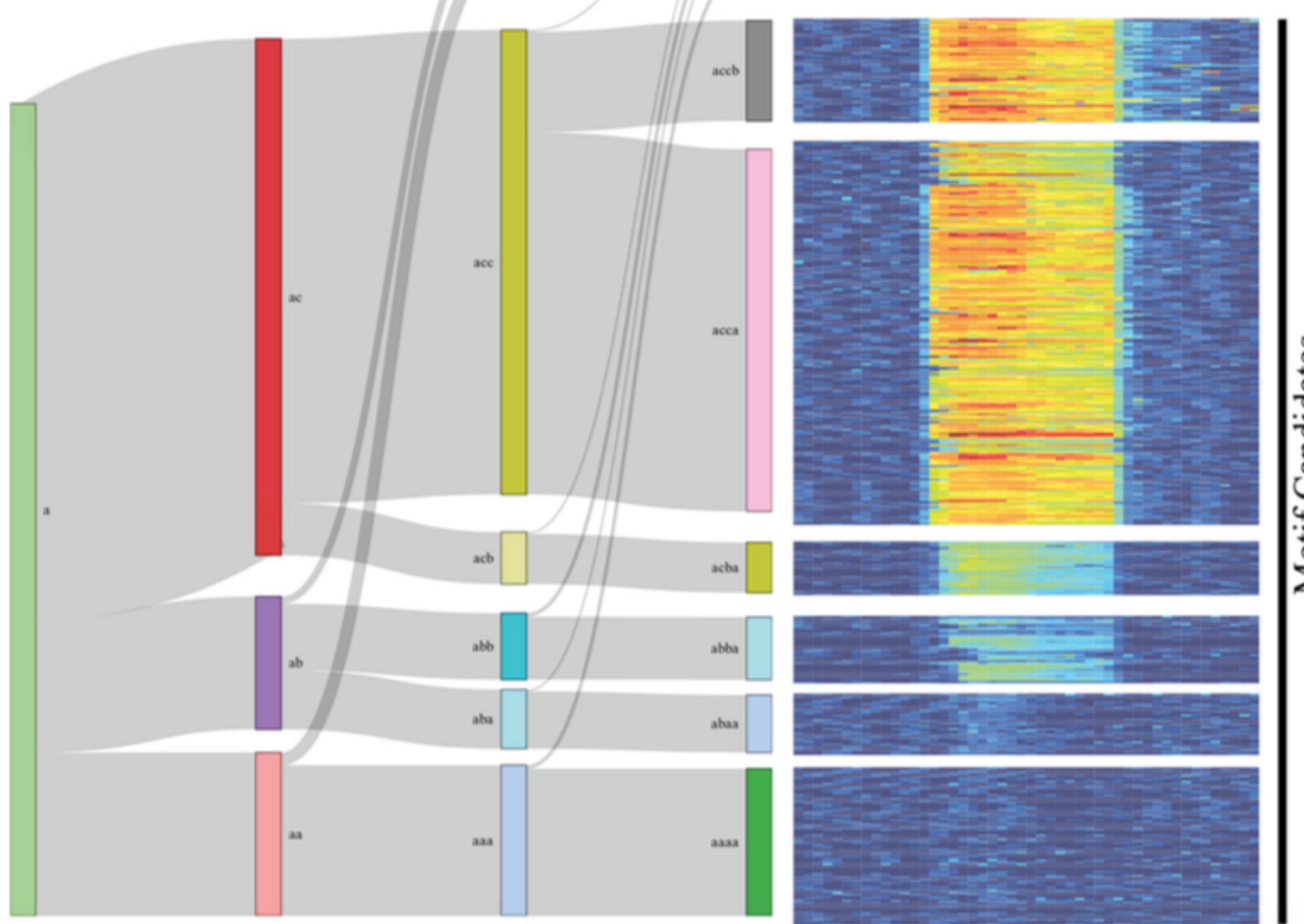
Motifs & Discords



Motifs & Discords



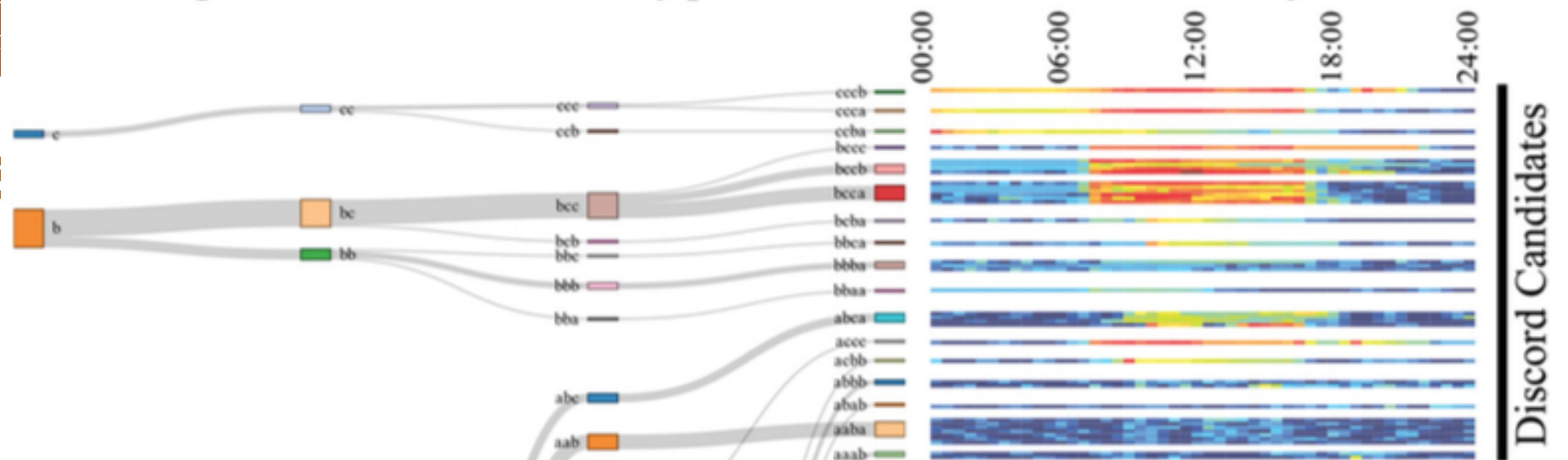
Motifs



Motif Candidates

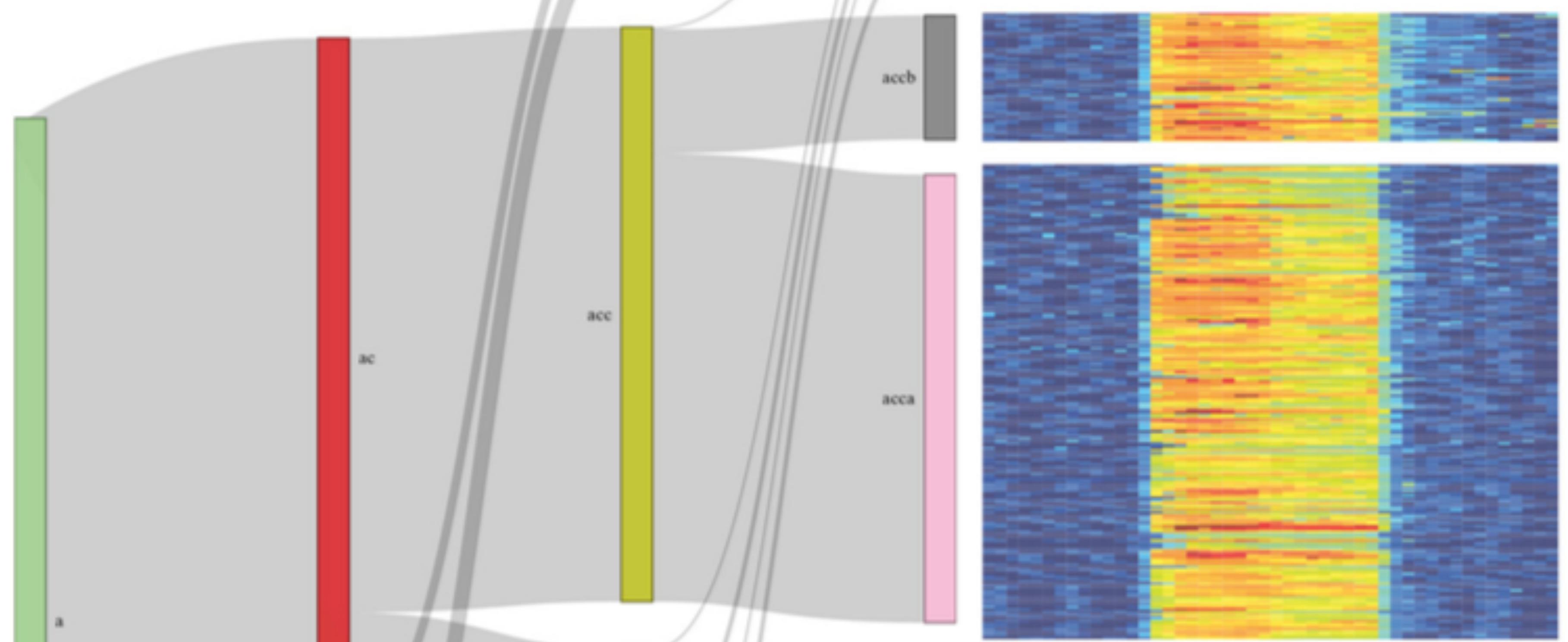
Motifs

Augmented suffix tree of daily patterns

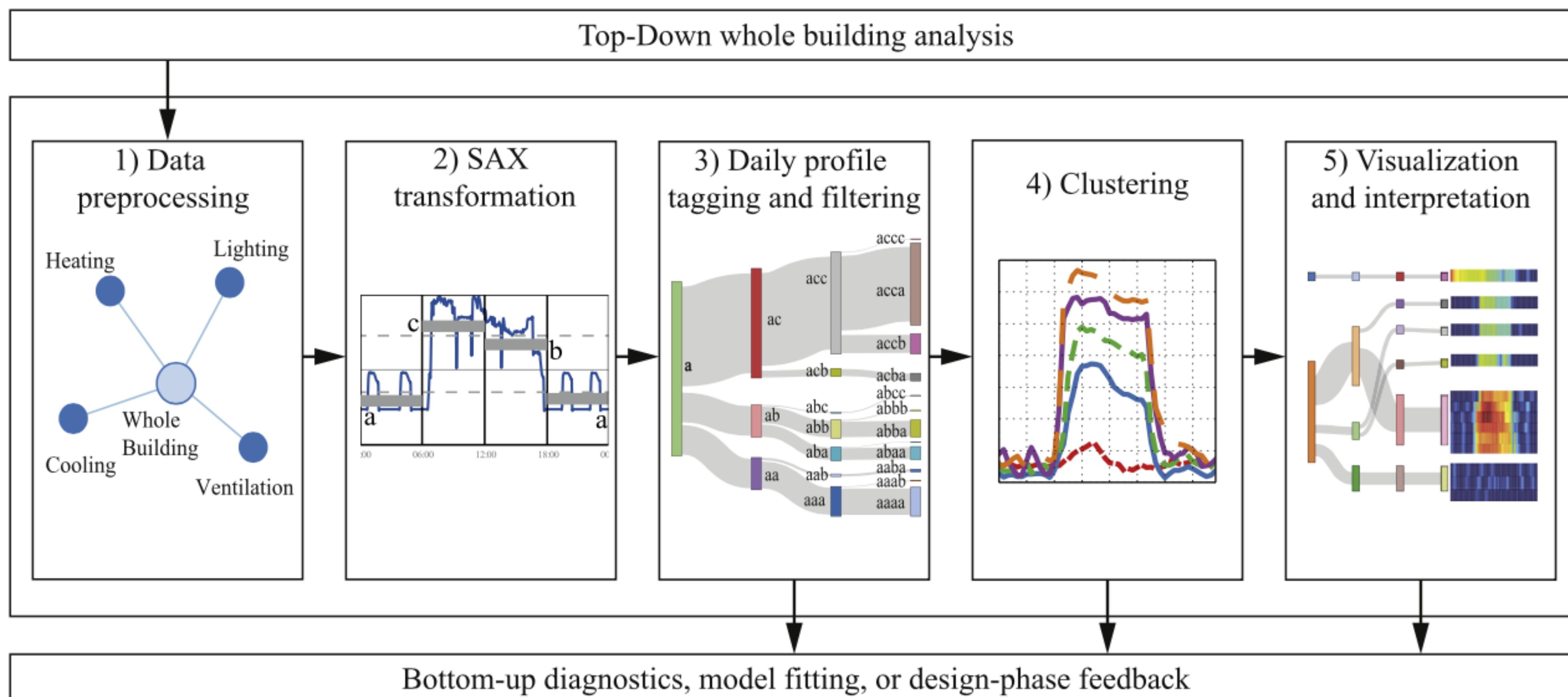


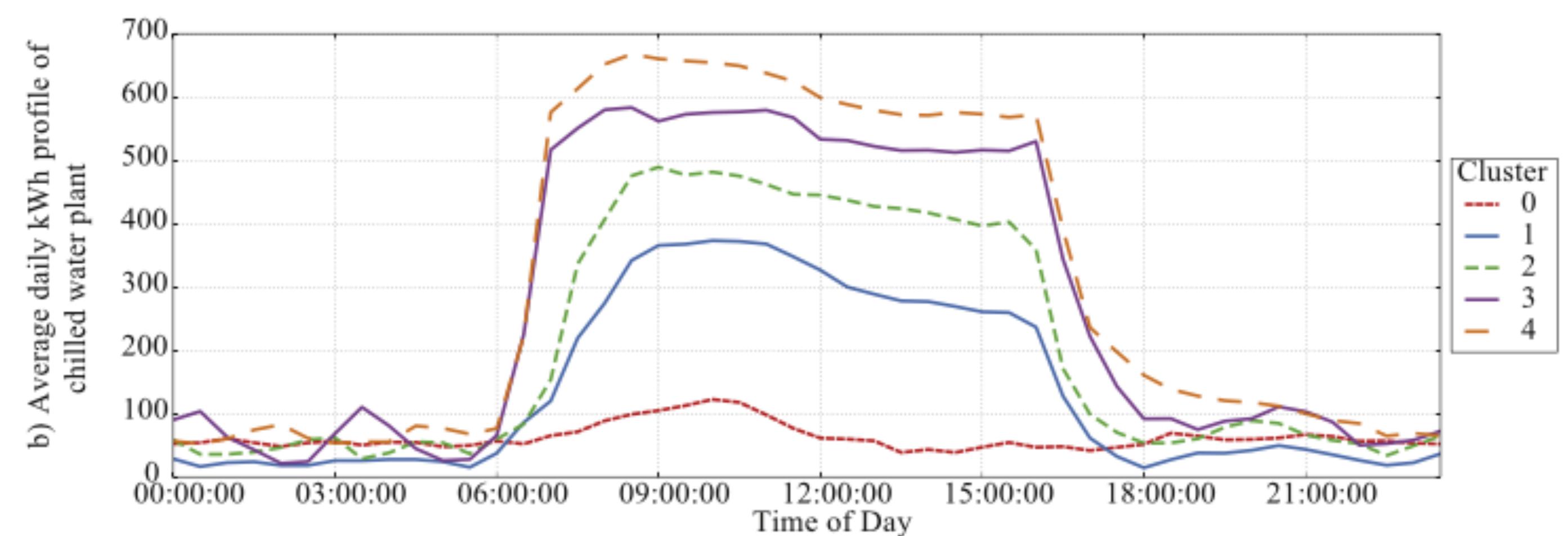
Time of day

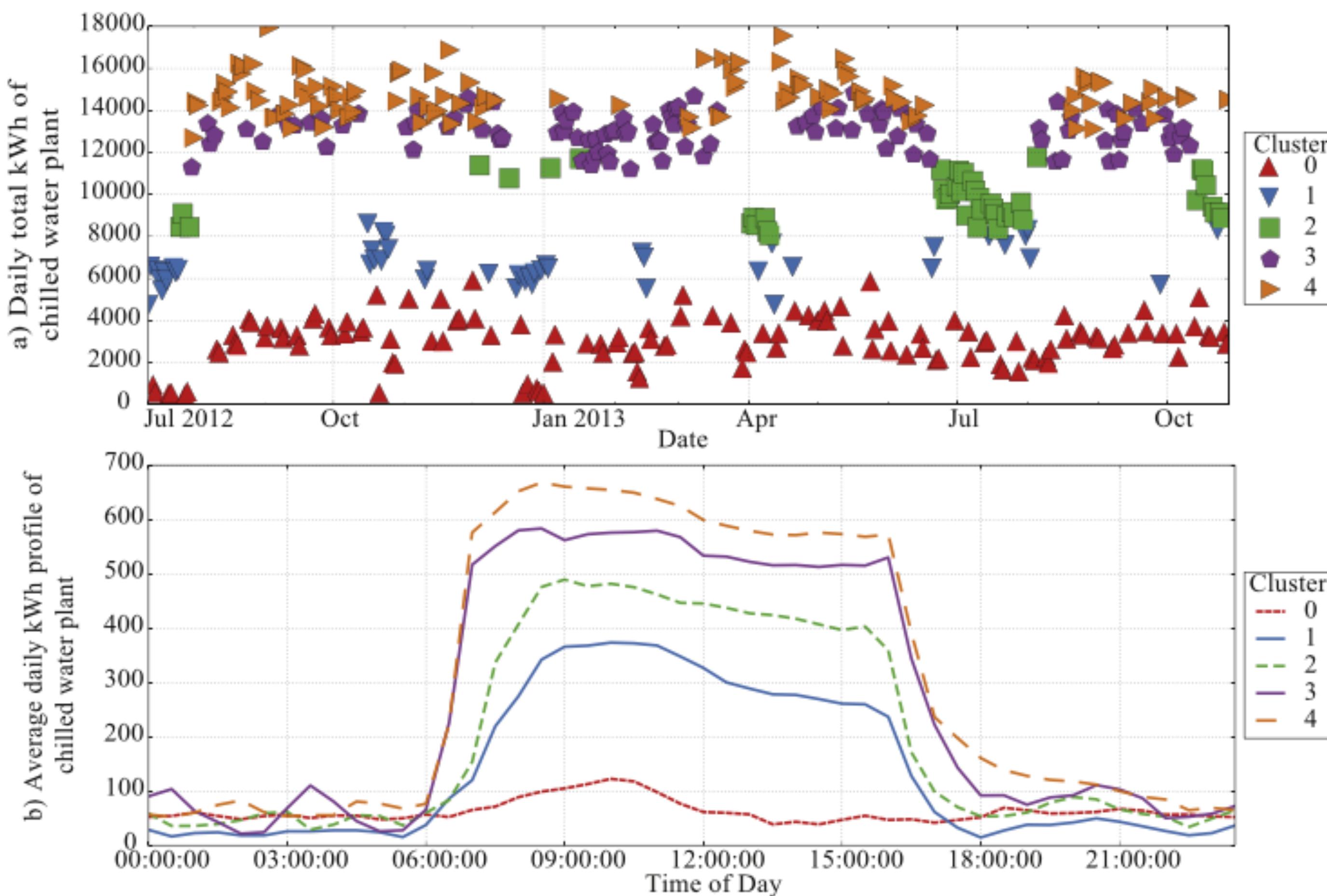
Discord Candidates

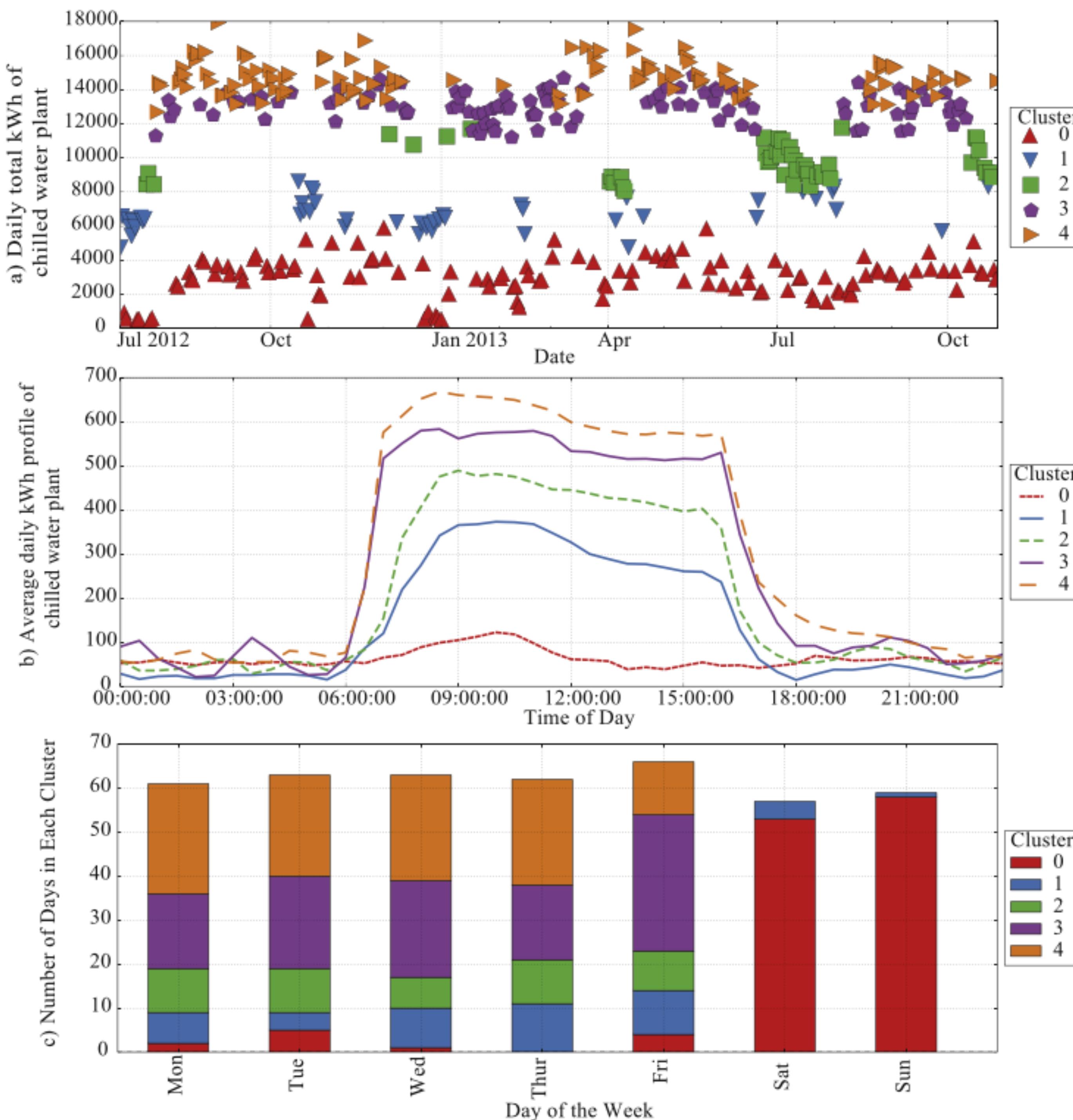


if Candidates









Python activity

- Notebooks for DayFilter
 - Part 1 SAX for motif & discord candidates
 - Part 2 Clustering motif candidates
- Homework 3: clustering of smart meter data

Questions?