SURF 2019

Jaehoon Cha
Electrical and Electronics Engineering

Content

- What you can do

- Recurrent Neural Networks

- Hierarchical Neural Network Architecture
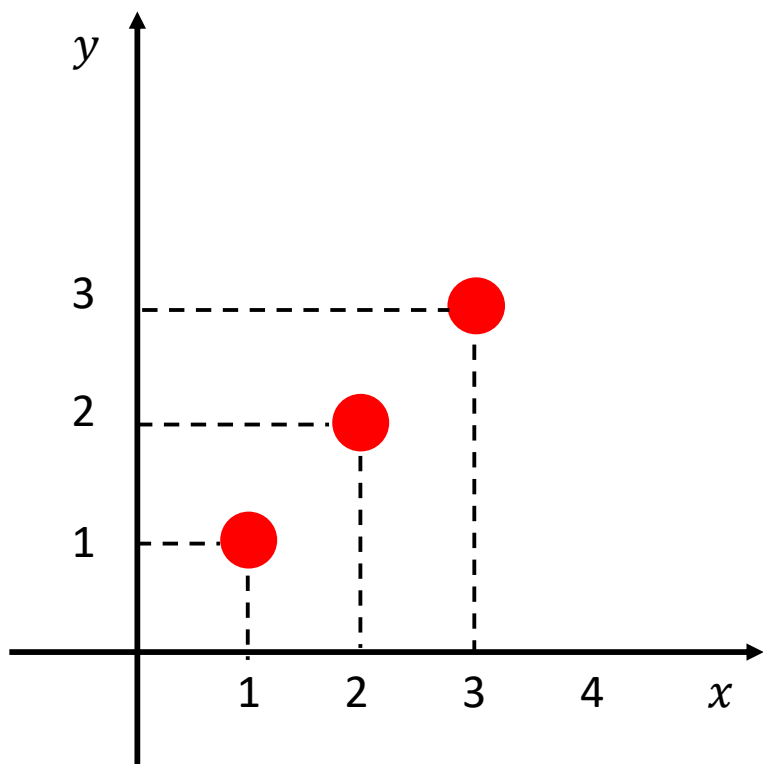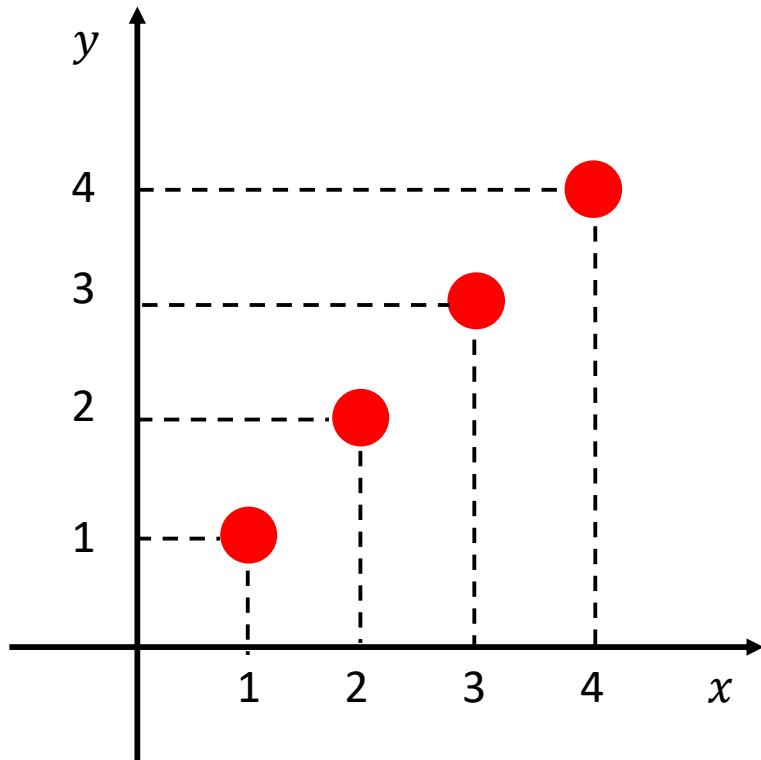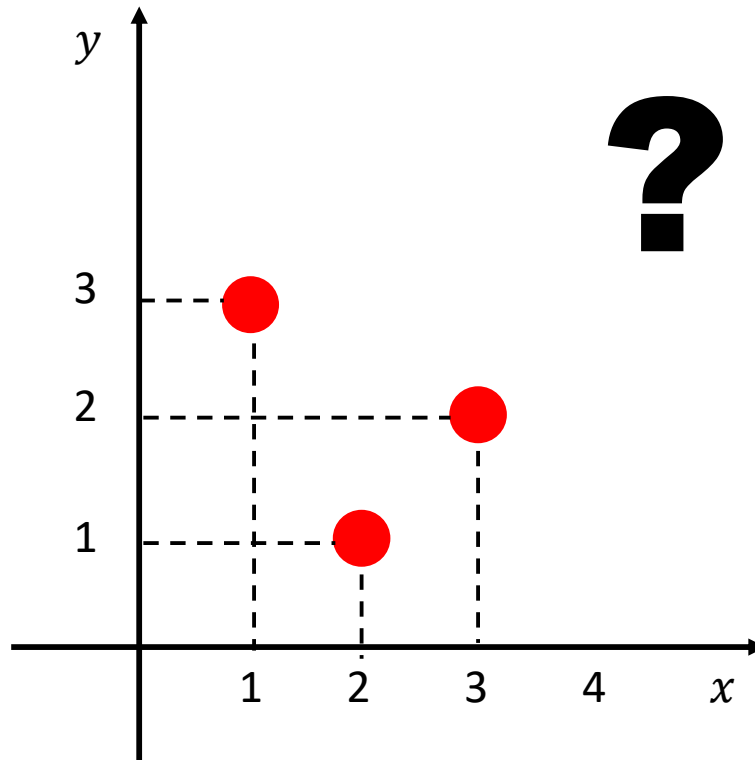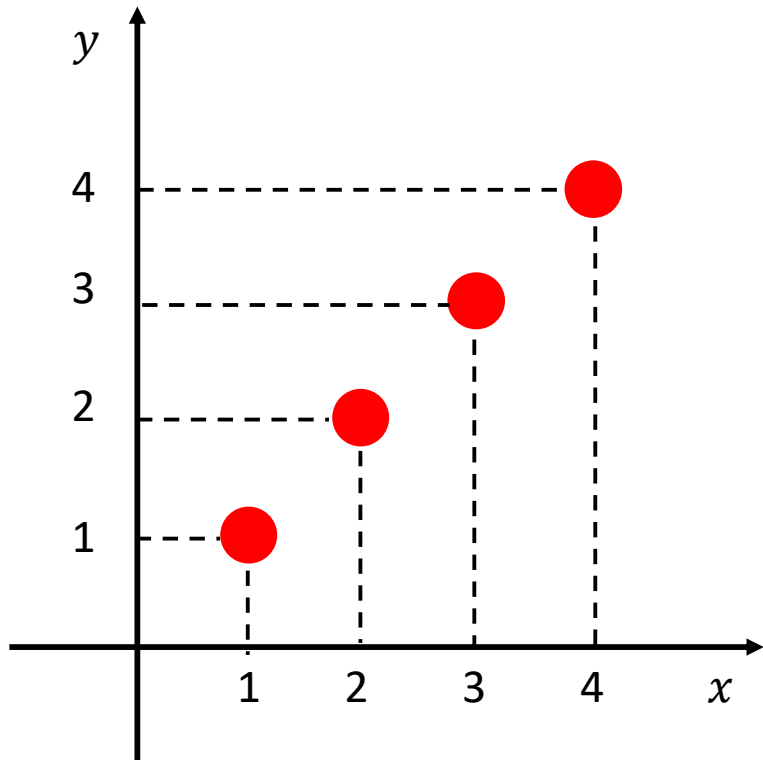
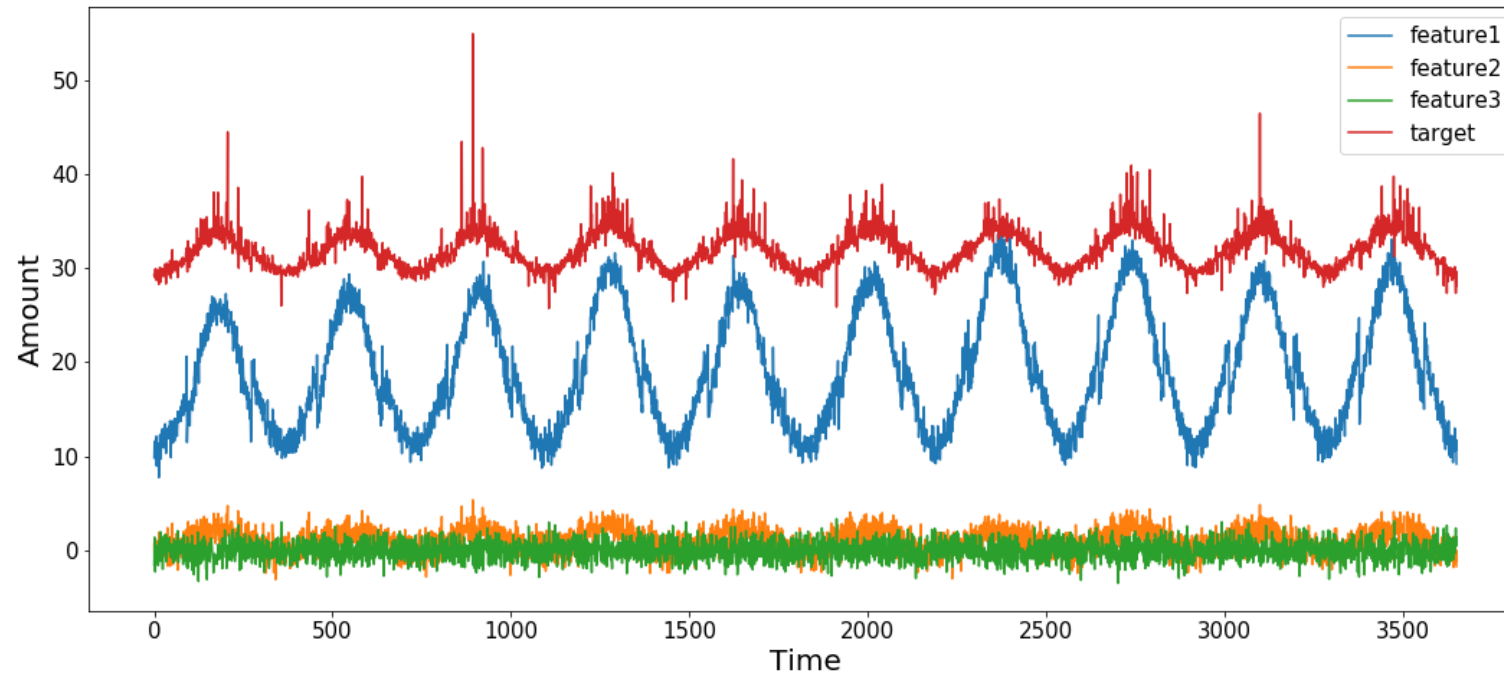# What you can do

What machines do

What machines do

What machines do

What machines do

e.g.,
- temperature, humidity, the amount of cloud
  -> precipitation
- population, the number of shops, the number of new residents
  -> housing price

What is a f here?

$$target = f(feature1, feature2, feature3)$$

What machines do

e.g.,
- temperature, humidity, the amount of cloud
  -> precipitation
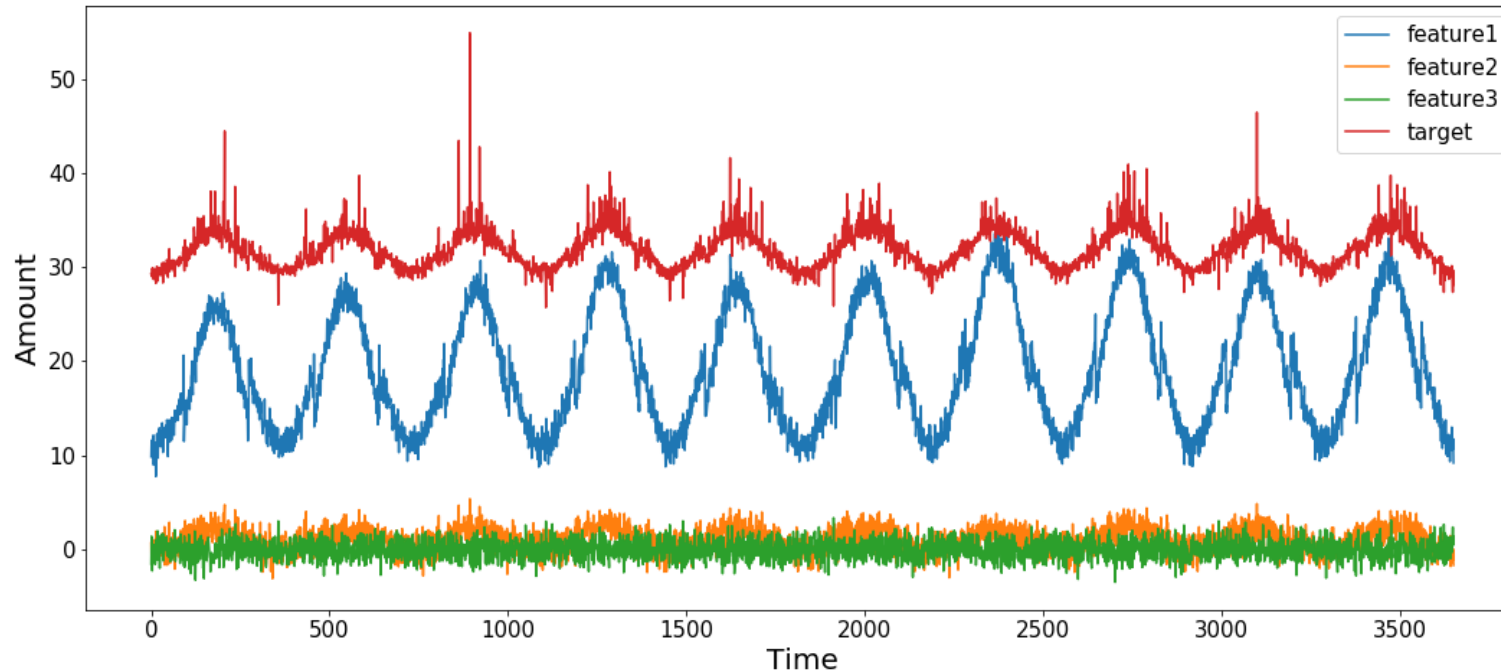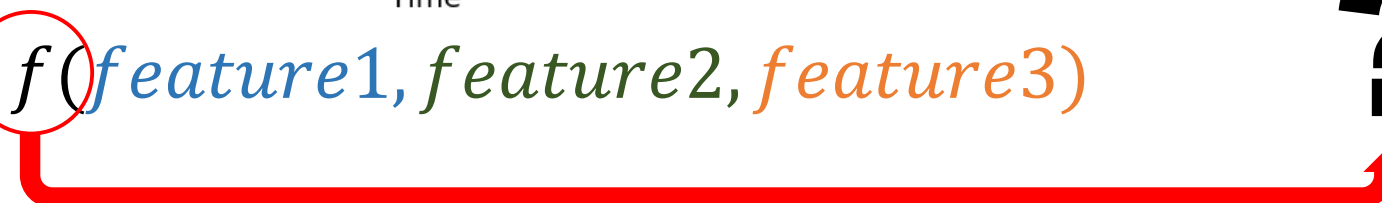- population, the number of shops, the number of new residents
  -> housing price

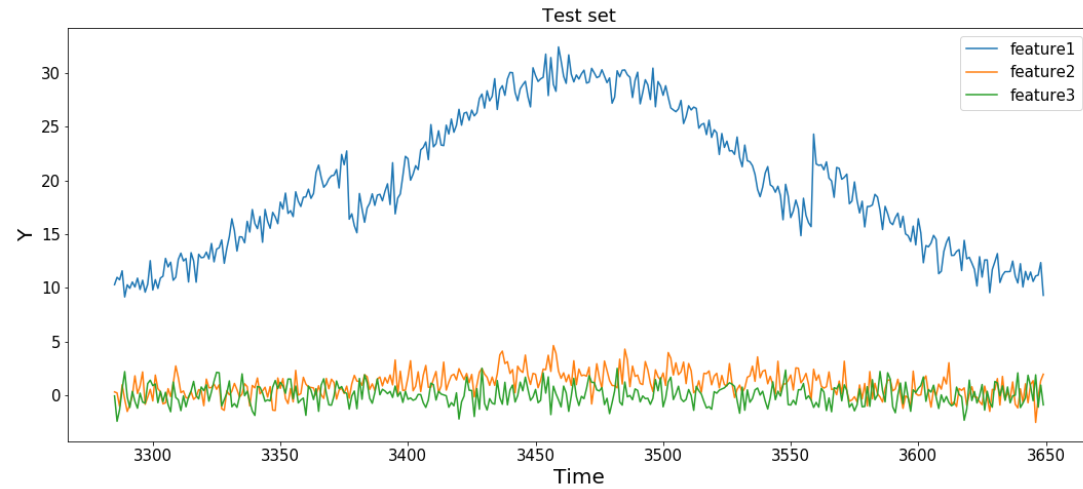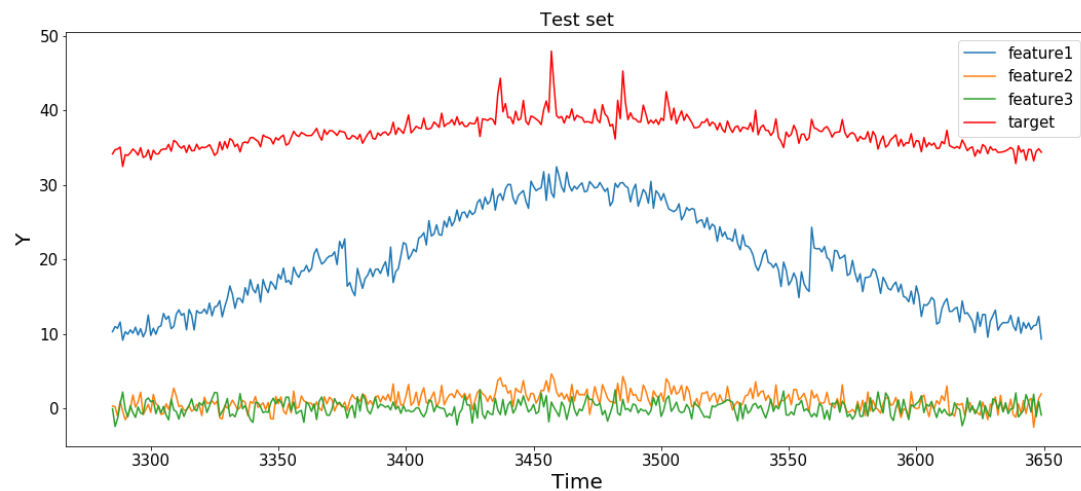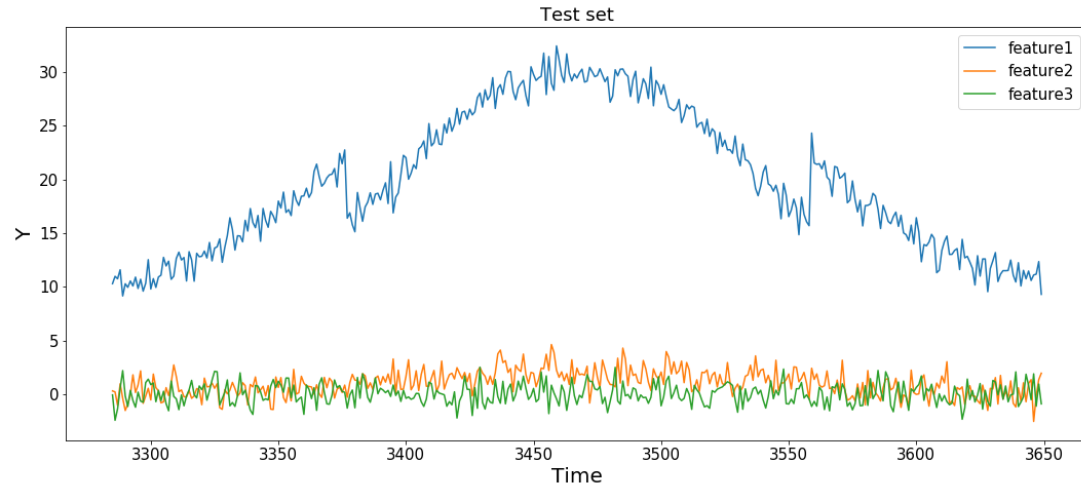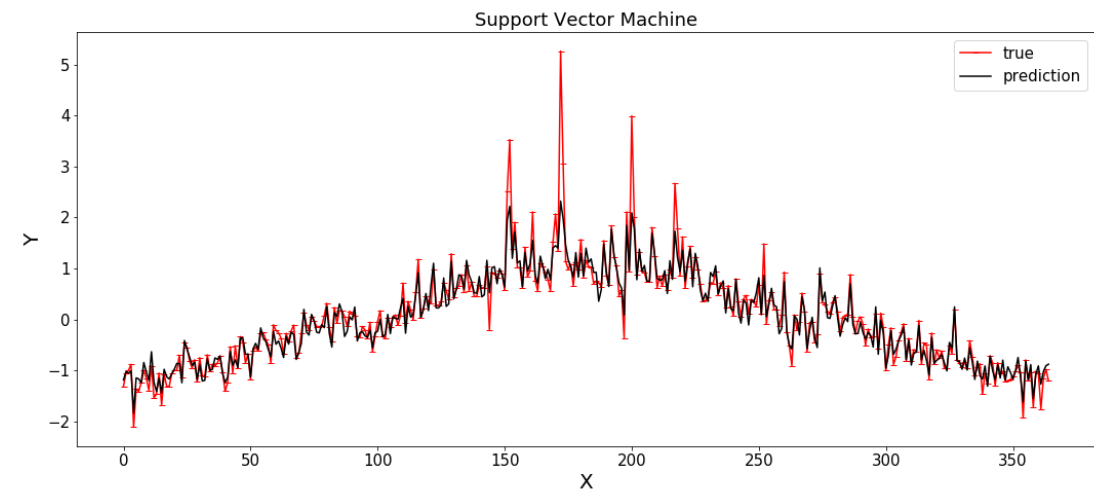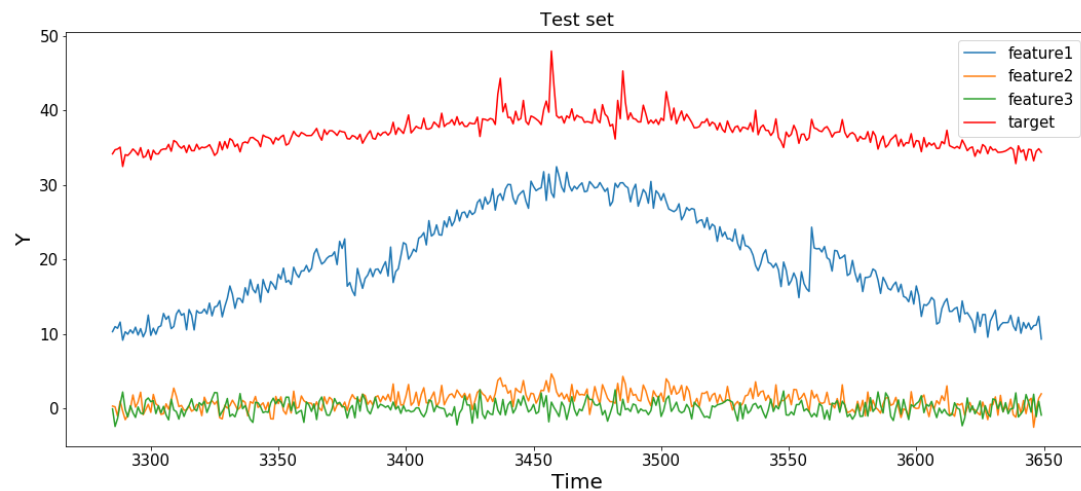It is more difficult to find a function in real life

$$target = f(feature1, feature2, feature3)$$

?

Machines can find the function
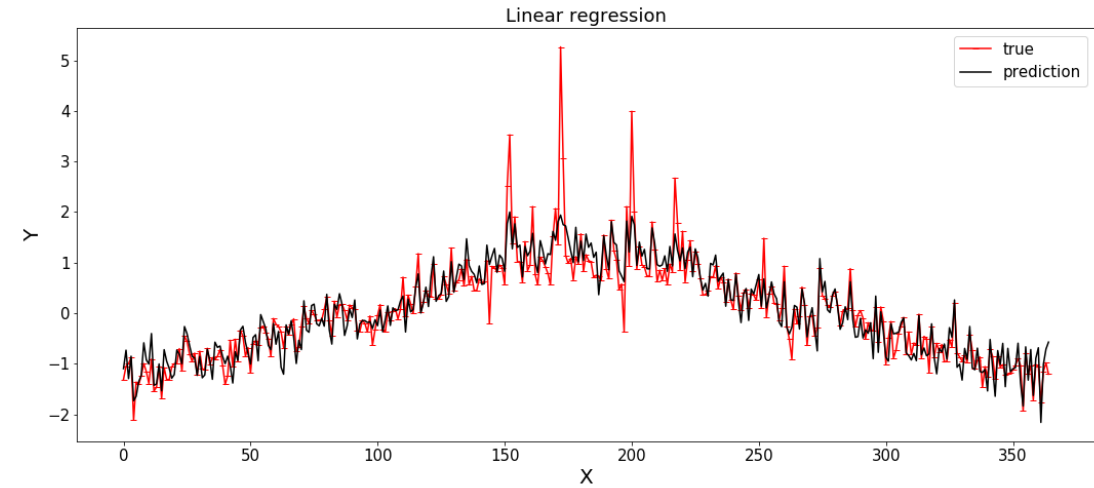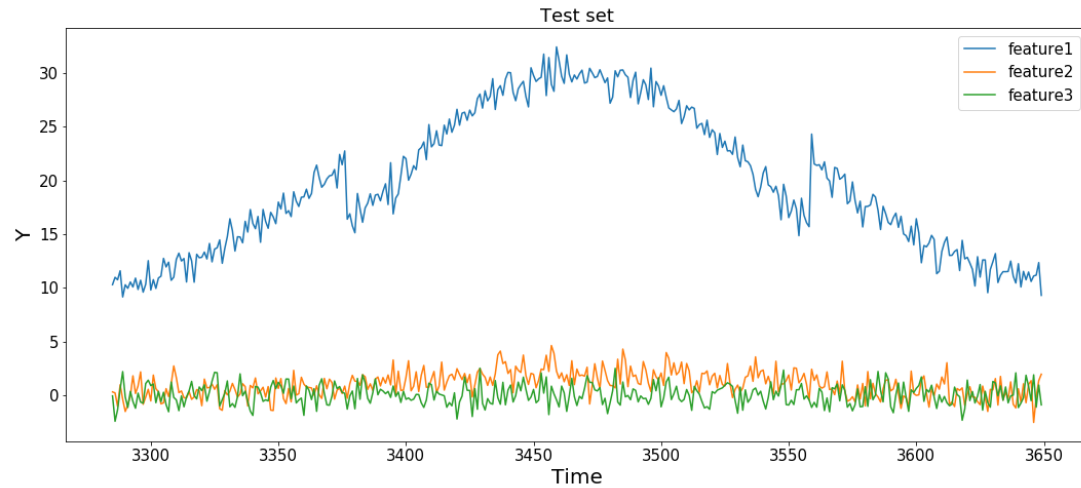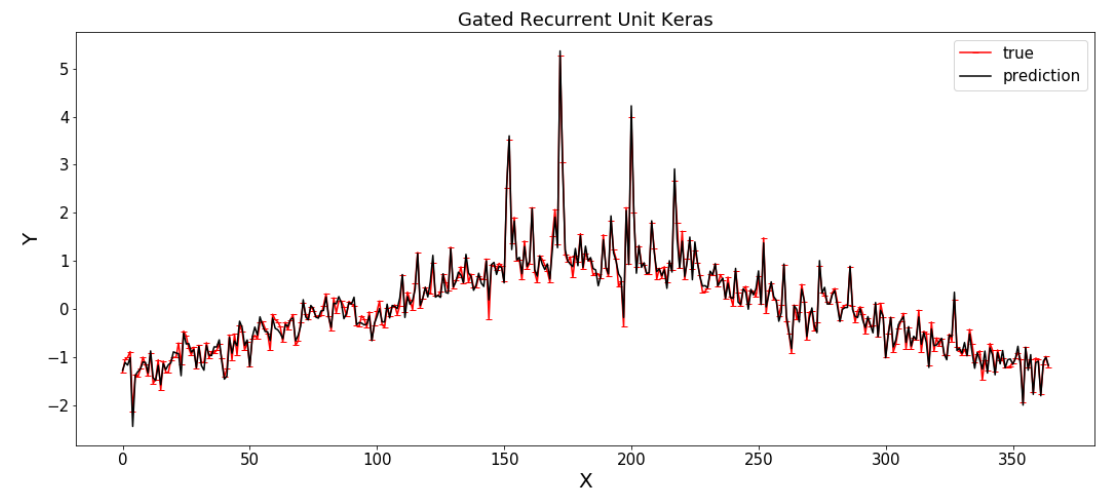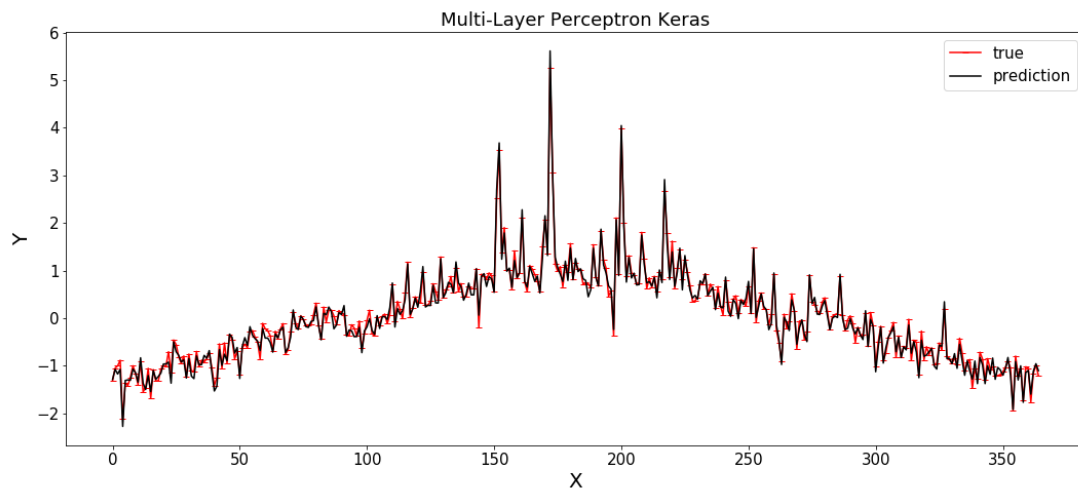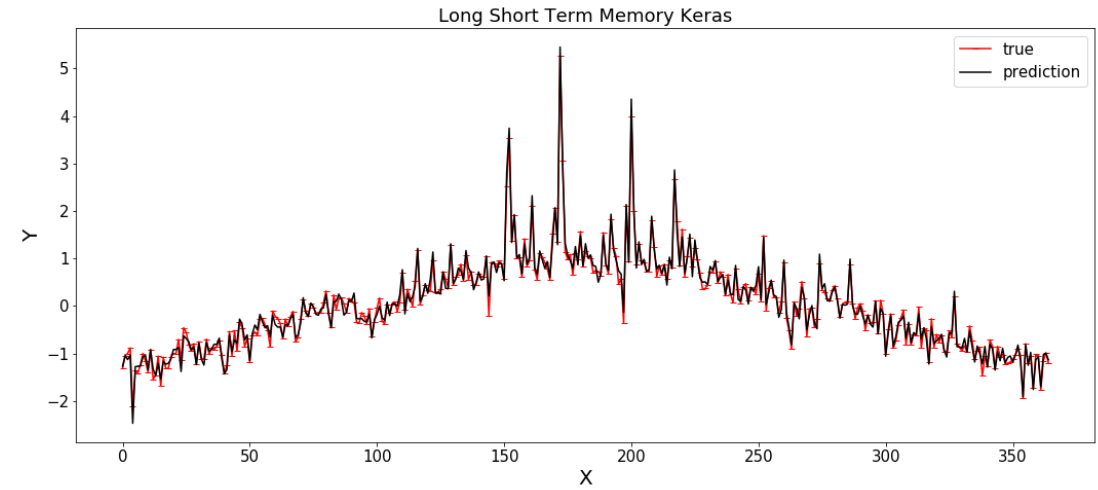
Machines can find the function

Machines can find the function

Machines can find the function
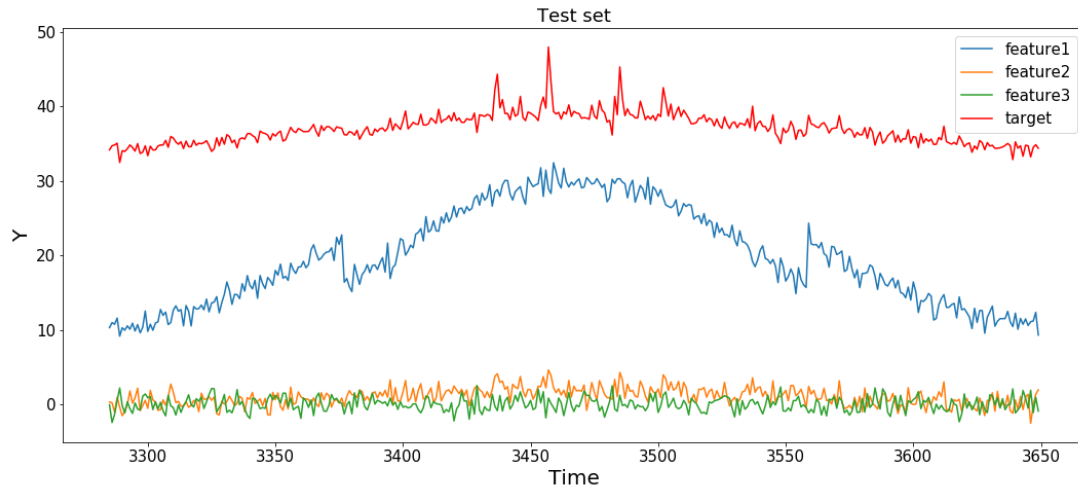
Machine learning can be used to analyze data
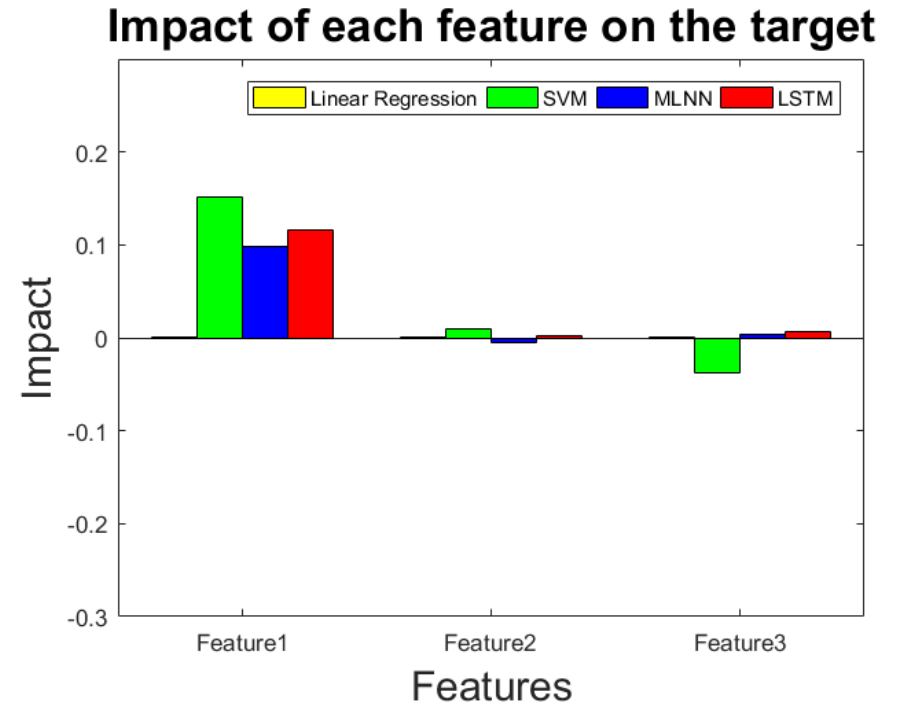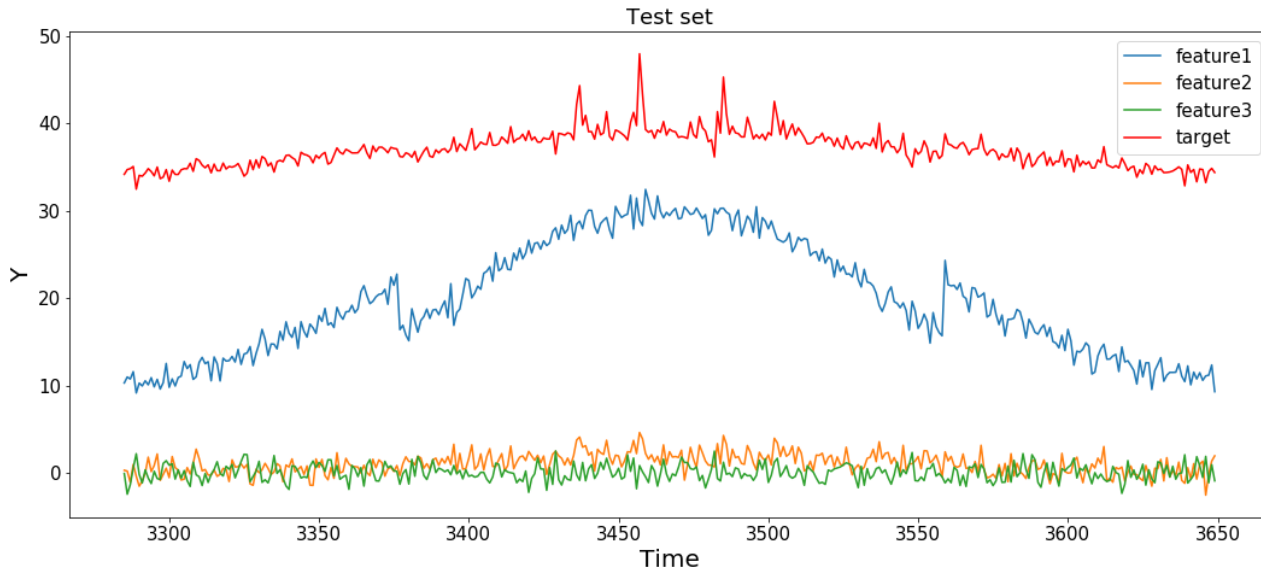
Recurrent Neural Network

Recurrent Neural Networks (RNNs)

- Recurrent Neural Networks (RNNs) is designed for time series data.

- The networks have loops so that they consider time dependencies between elements in the time series data.

Recurrent Neural Networks (RNNs)

Conventional Neural Networks:

hell $\longrightarrow$ $\boxed{f}$ $\longrightarrow$ o

Recurrent Neural Networks:

$\boxed{f}$ $\xrightarrow{s_1}$ $\boxed{f}$ $\xrightarrow{s_2}$ $\boxed{f}$ $\xrightarrow{s_3}$ $\boxed{f}$ $\longrightarrow$ o

h   e   l   l
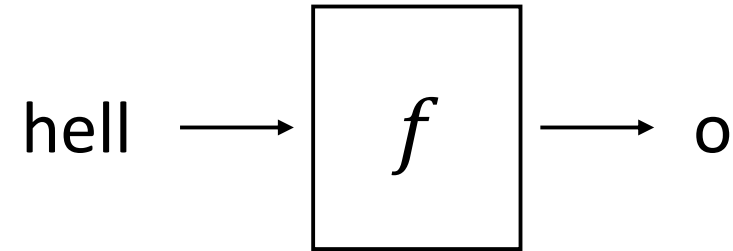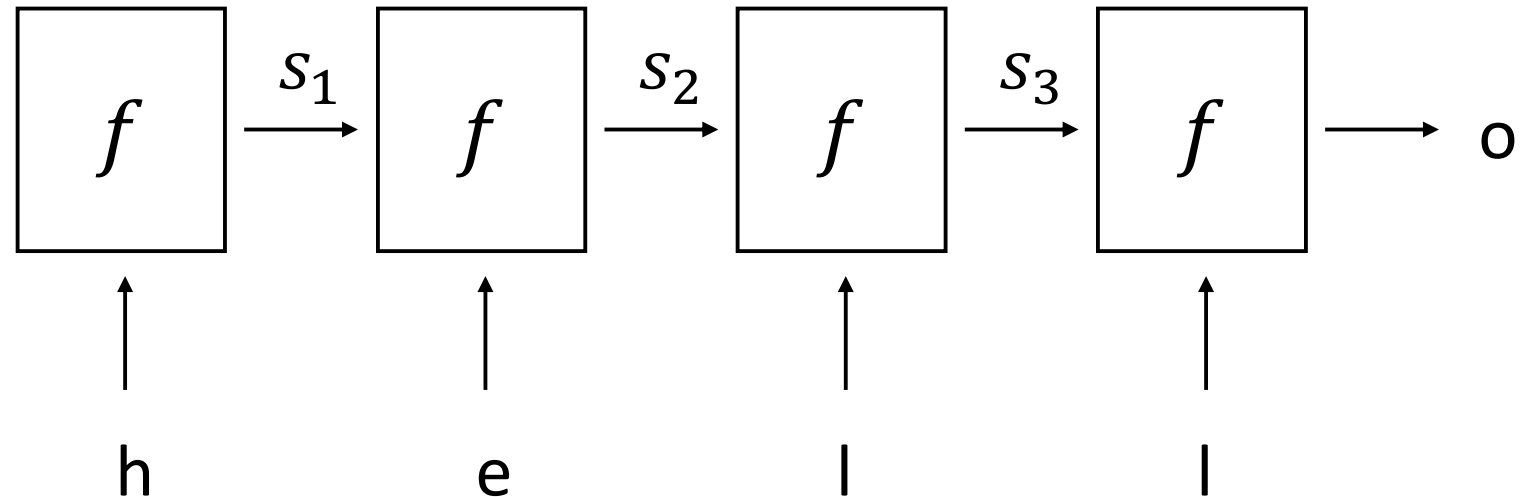
Recurrent Neural Networks (RNNs)

$$s_t = f(Wx_t + Us_{t-1} + b)$$

Recurrent Neural Networks (RNNs)

What if words are so long?
e.g., Pneumonoultramicroscopicsilicovolcanoconiosis: lung disease caused by micro dust

Long Short Term Memory (LSTM)
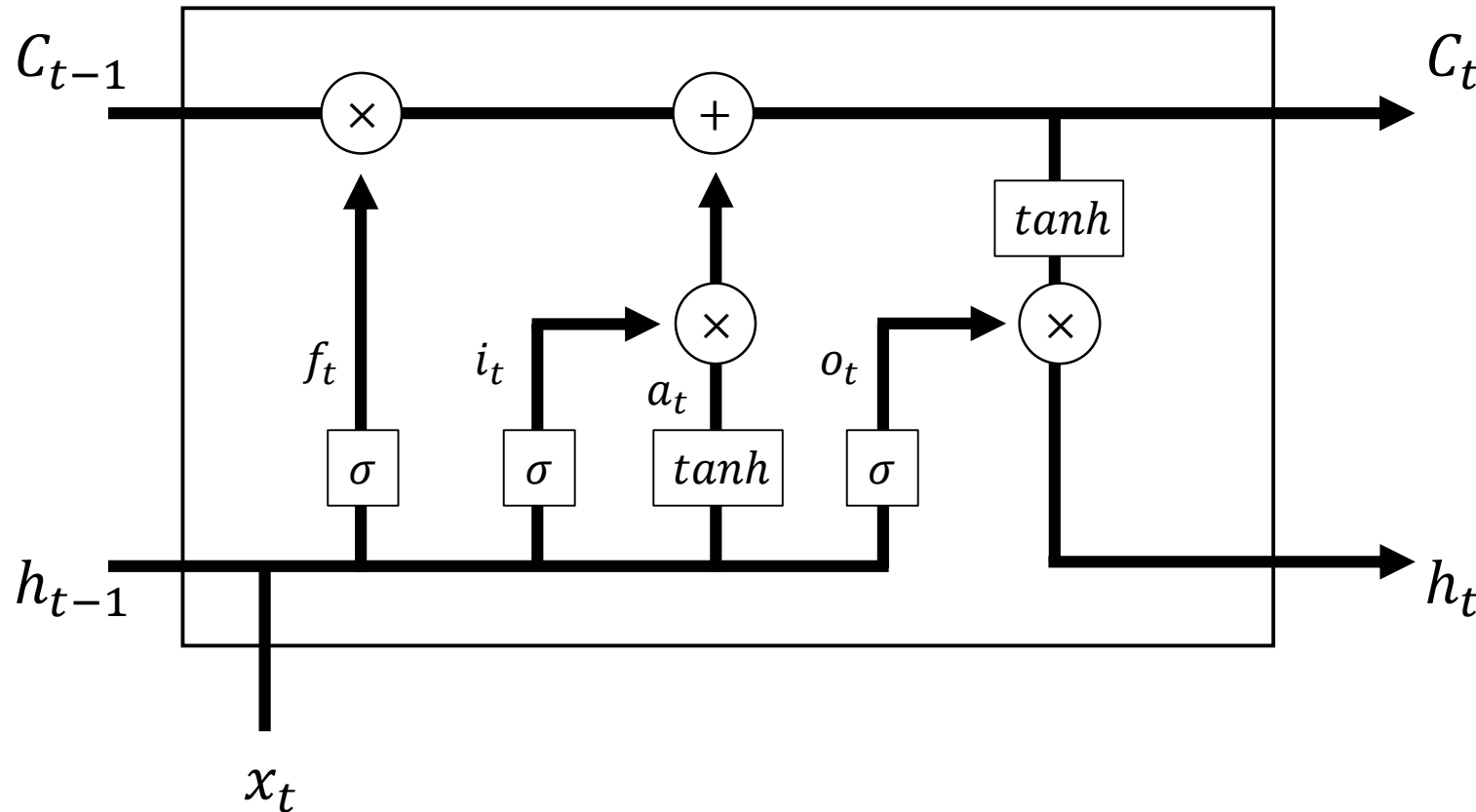
$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$
$$a_t = tanh(W_a x_t + U_a h_{t-1} + b_a)$$

Long Short Term Memory (LSTM)

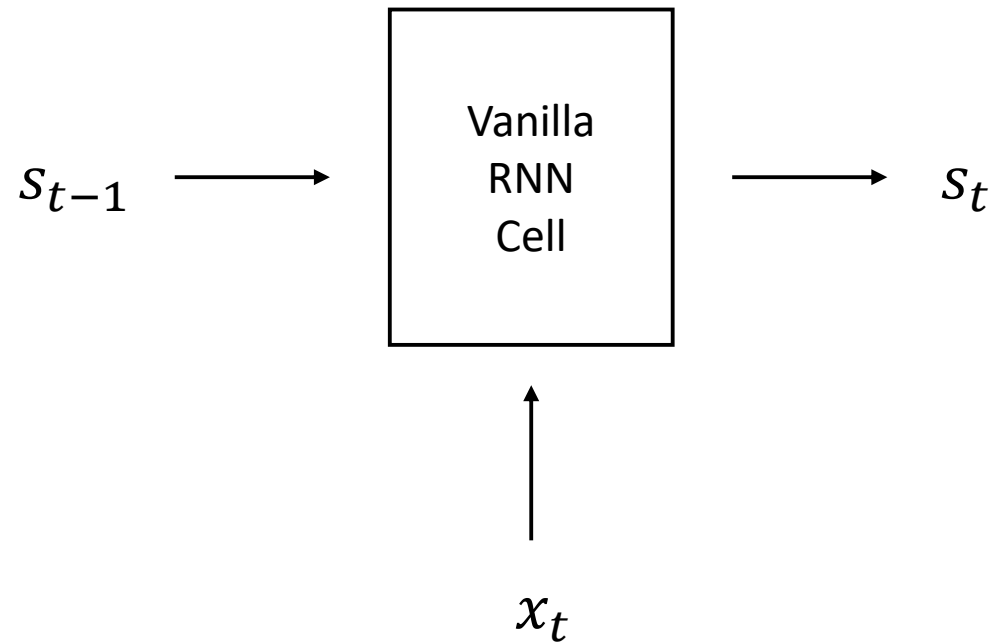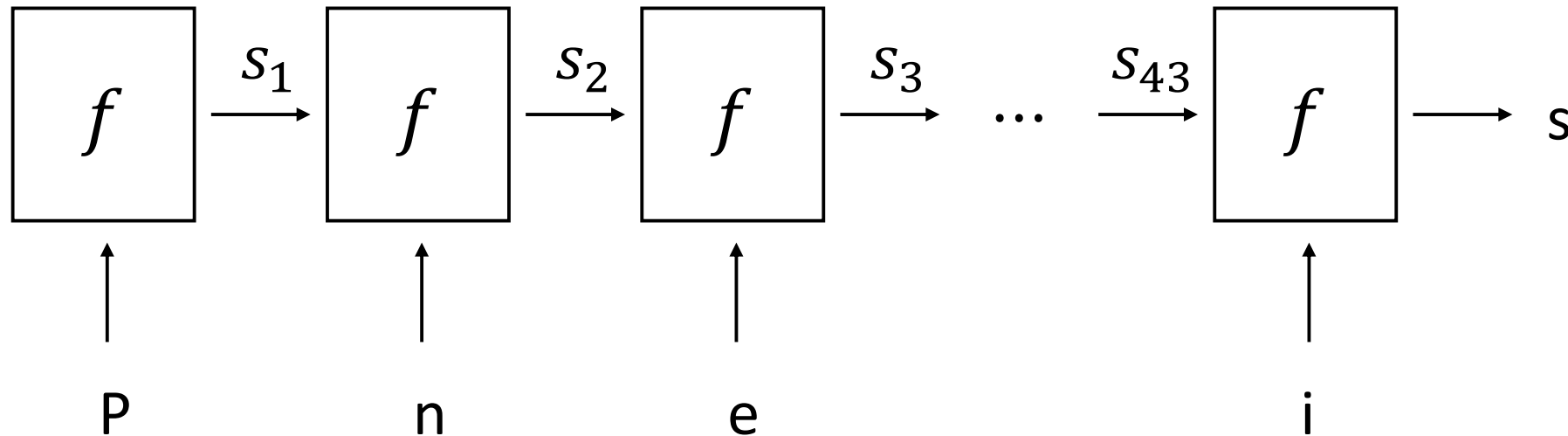$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$
$$a_t = tanh(W_a x_t + U_a h_{t-1} + b_a)$$

$$C_t = a_t \odot i_t + f_t \odot C_{t-1}$$
$$h_t = \tanh(C_t) \odot o_t$$

Long Short Term Memory (LSTM)

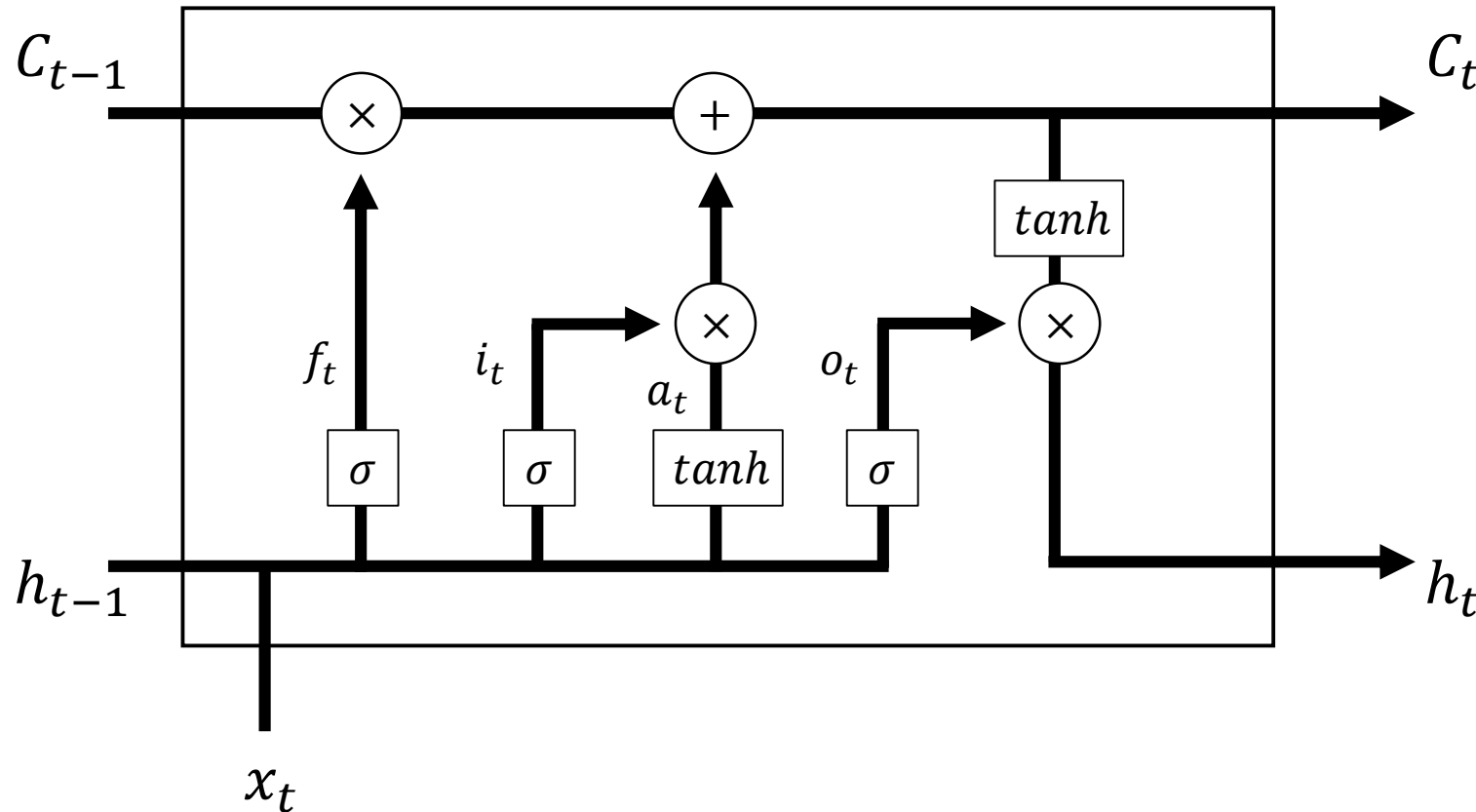$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$
$$a_t = tanh(W_a x_t + U_a h_{t-1} + b_a)$$

$$C_t = a_t \odot i_t + f_t \odot C_{t-1}$$
$$h_t = \tanh(C_t) \odot o_t$$

Q. $0 \leq \sigma(x) \leq 1, -1 \leq tanh(x) \leq 1$
What if σ is used?

Long Short Term Memory (LSTM)
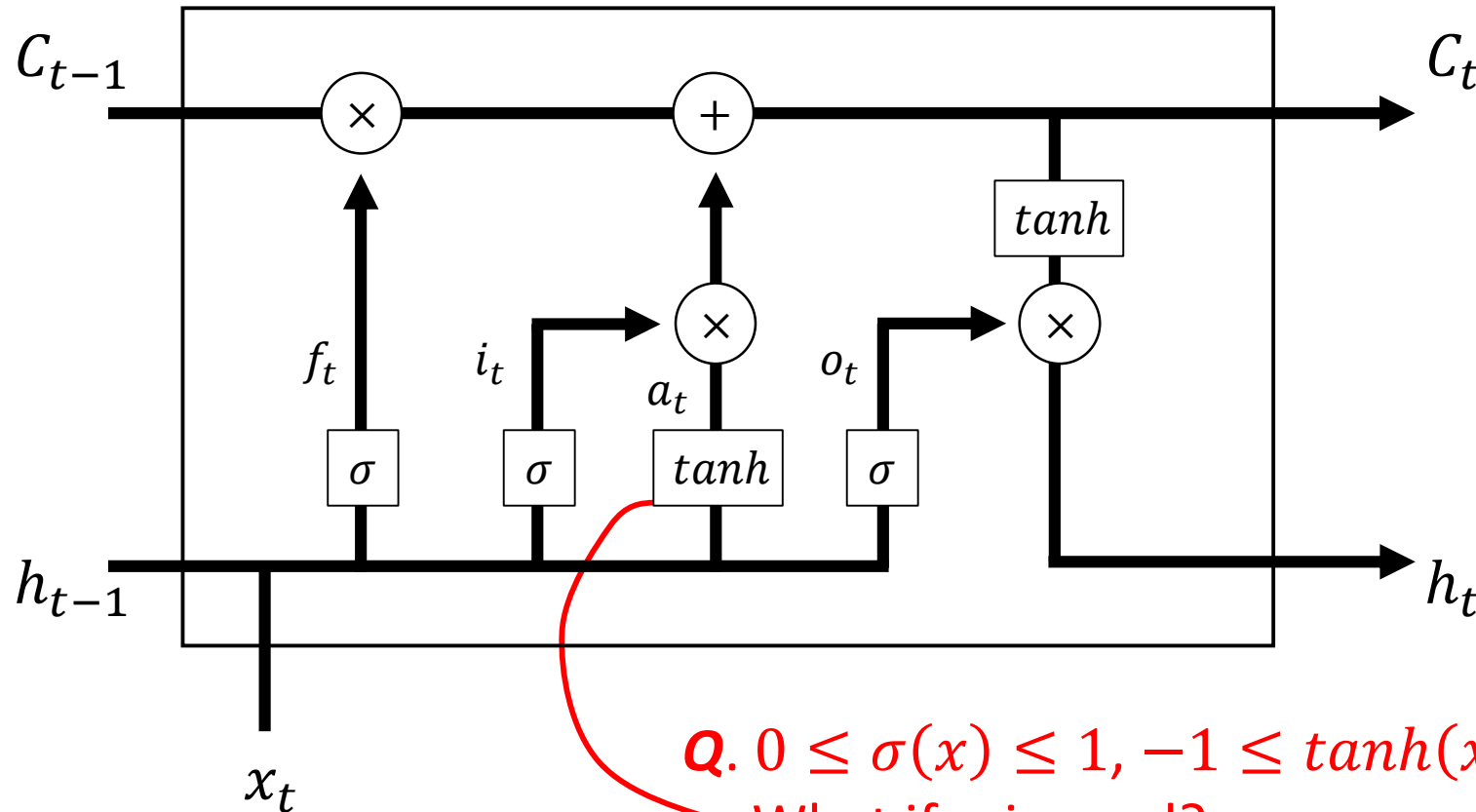
Do not have the previous state information

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$$
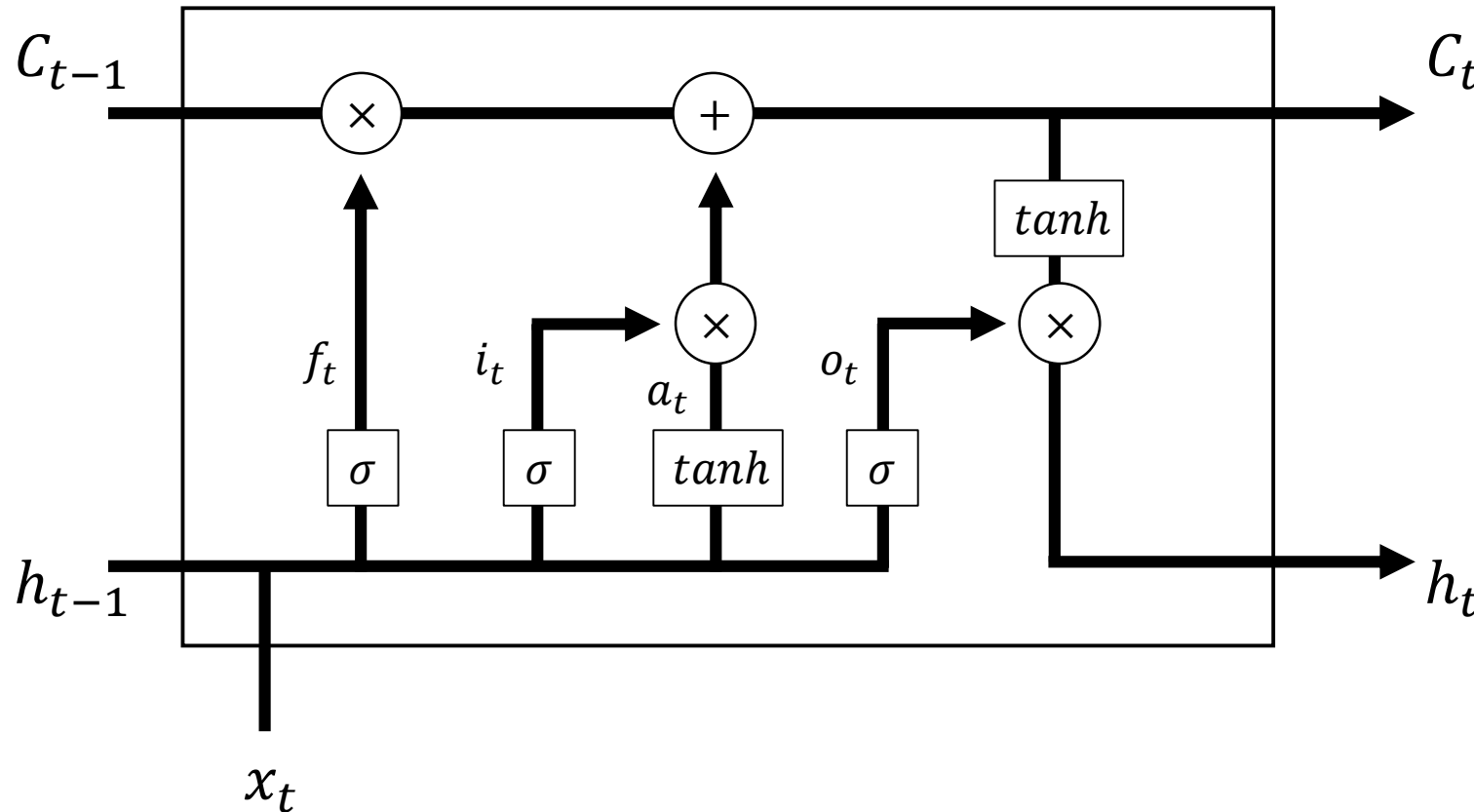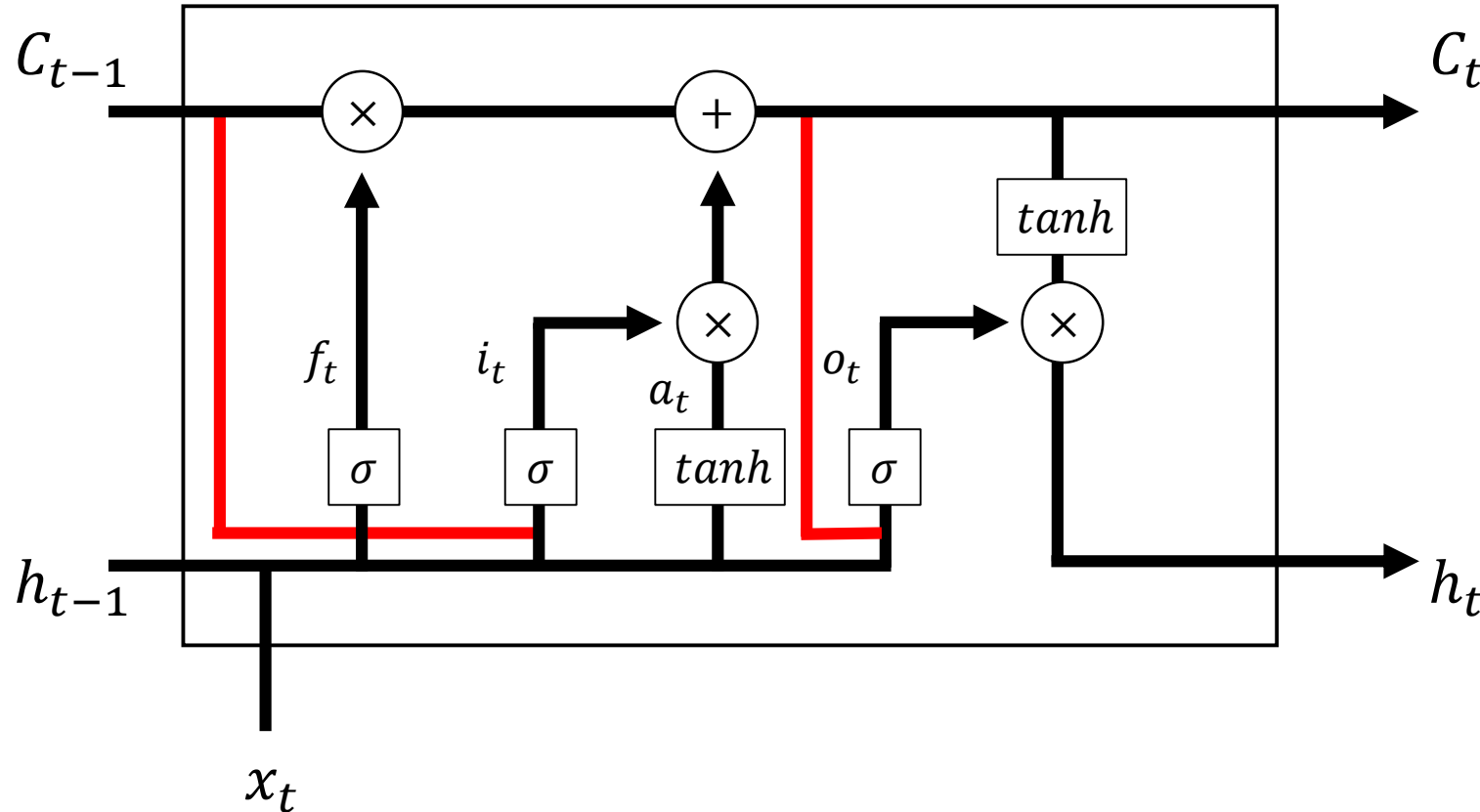
$$a_t = tanh(W_a x_t + U_a h_{t-1} + b_a)$$

$$C_t = a_t \odot i_t + f_t \odot C_{t-1}$$

$$h_t = \tanh(C_t) \odot o_t$$

LSTM with Peephole Connection

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + C_{t-1} P_f + b_f)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + C_{t-1} P_i + b_i)$$

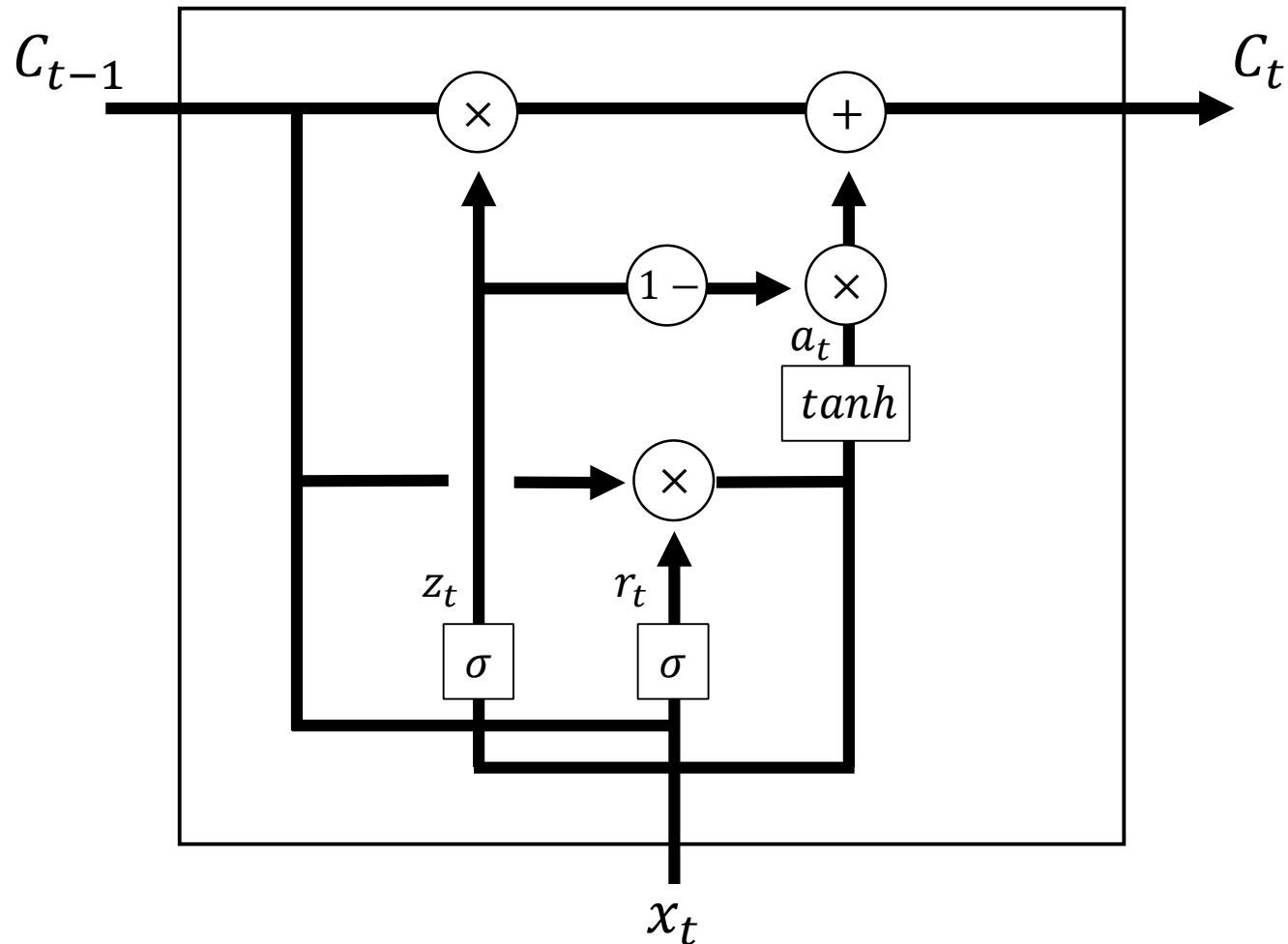$$o_t = \sigma(W_o x_t + U_o h_{t-1} + C_t P_o + b_o)$$

$$a_t = tanh(W_a x_t + U_a h_{t-1} + b_a)$$

$$C_t = a_t \odot i_t + f_t \odot C_{t-1}$$

$$h_t = \tanh(C_t) \odot o_t$$

Gated Recurrent Unit (GRU)

$$z_t = \sigma(W_z x_t + U_z C_{t-1} + b_z)$$

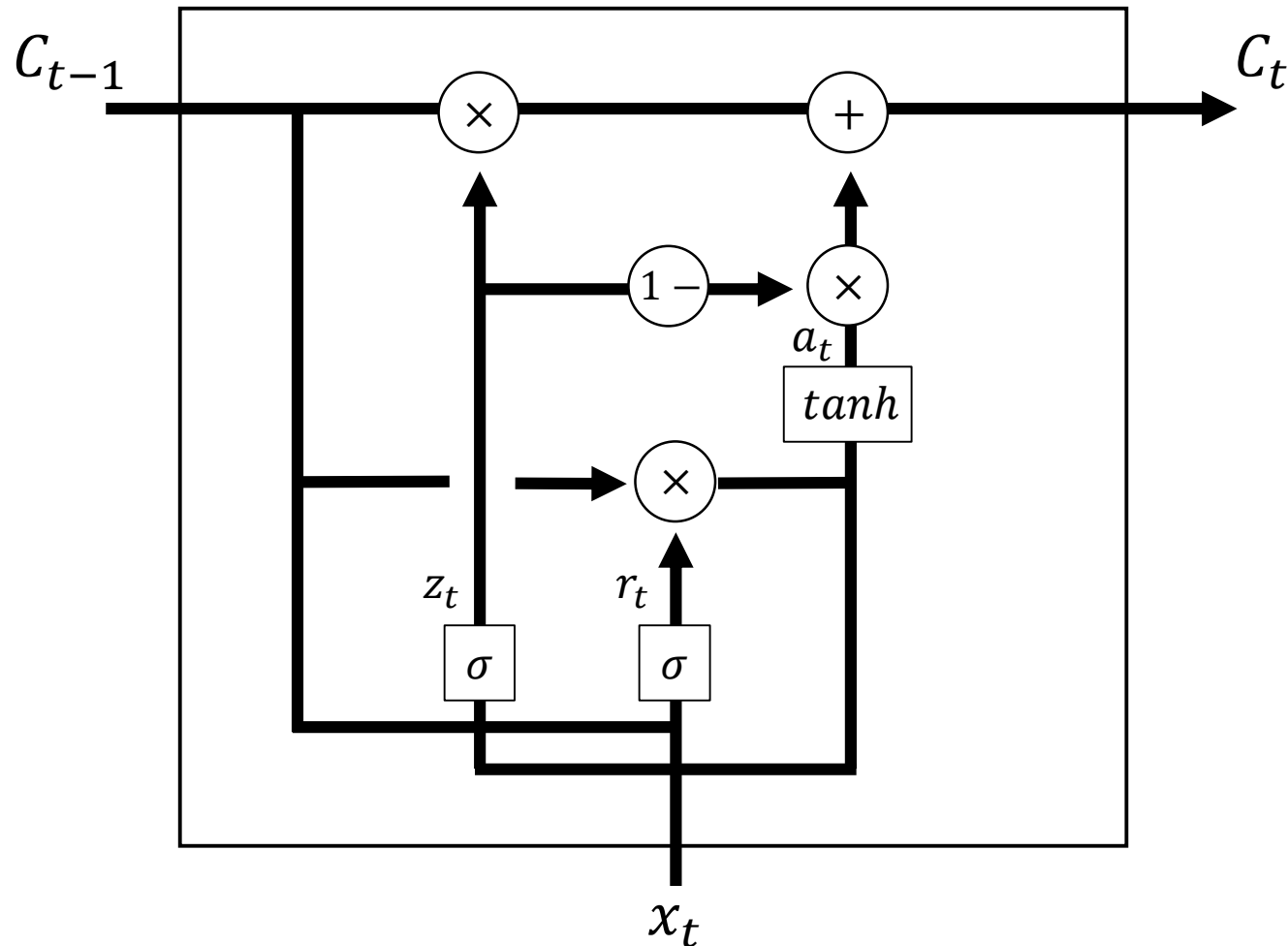$$r_t = \sigma(W_r x_t + U_r C_{t-1} + b_r)$$

$$a_t = tanh(W_a x_t + U_a(r_t \odot C_{t-1}) + b_a)$$

$$C_t = (1 - z_t) \odot a_t + z_t \odot C_{t-1}$$

Gated Recurrent Unit (GRU)

$$z_t = \sigma(W_z x_t + U_z C_{t-1} + b_z)$$
$$r_t = \sigma(W_r x_t + U_r C_{t-1} + b_r)$$
$$a_t = tanh(W_a x_t + U_a(r_t \odot C_{t-1}) + b_a)$$

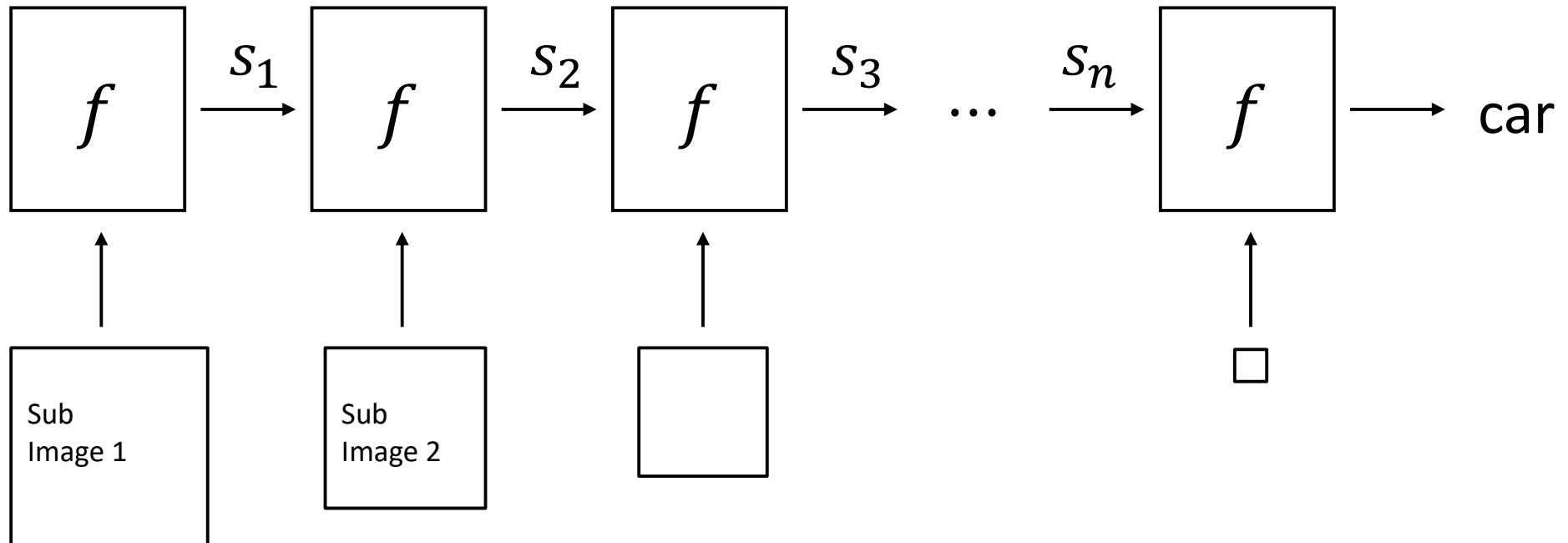$$C_t = (1 - z_t) \odot a_t + z_t \odot C_{t-1}$$

$C_t = i_t \odot a_t + f_t \odot C_{t-1}$ in LSTM

Hierarchical Architecture

Stage-wise Training

- Image classification

- Give subsampling image step by step

$$f \xrightarrow{s_1} f \xrightarrow{s_2} f \xrightarrow{s_3} \cdots \xrightarrow{s_n} f \rightarrow \text{car}$$

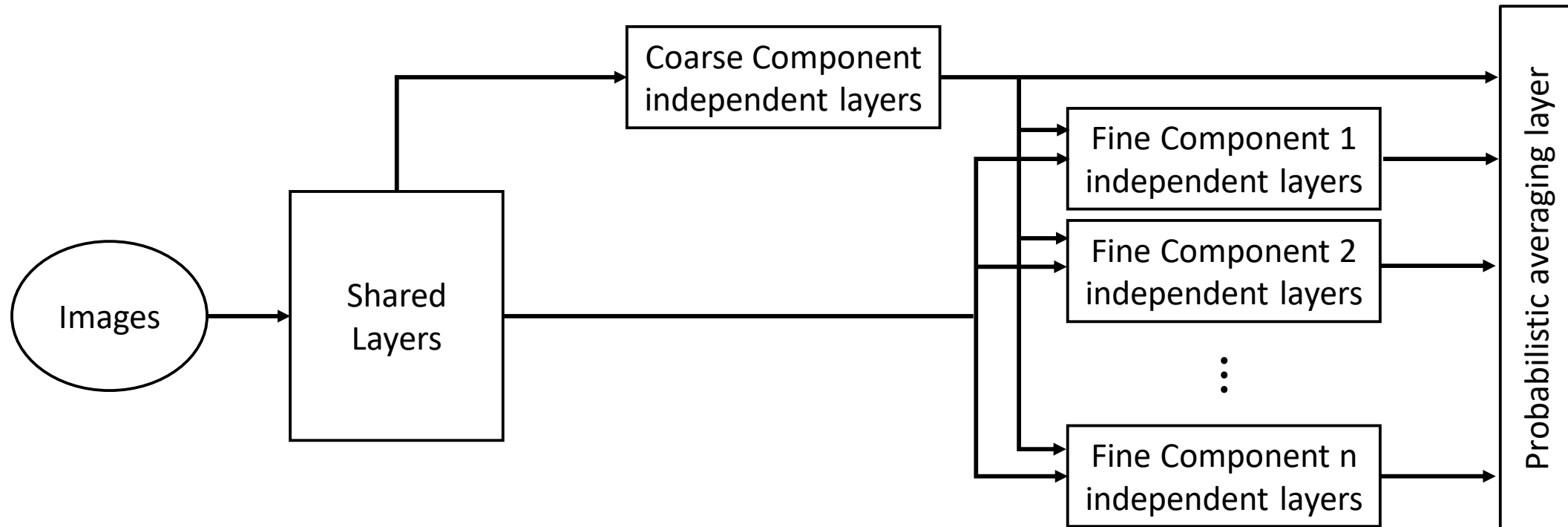Sub
Image 1

Sub
Image 2

Stage-wise Training
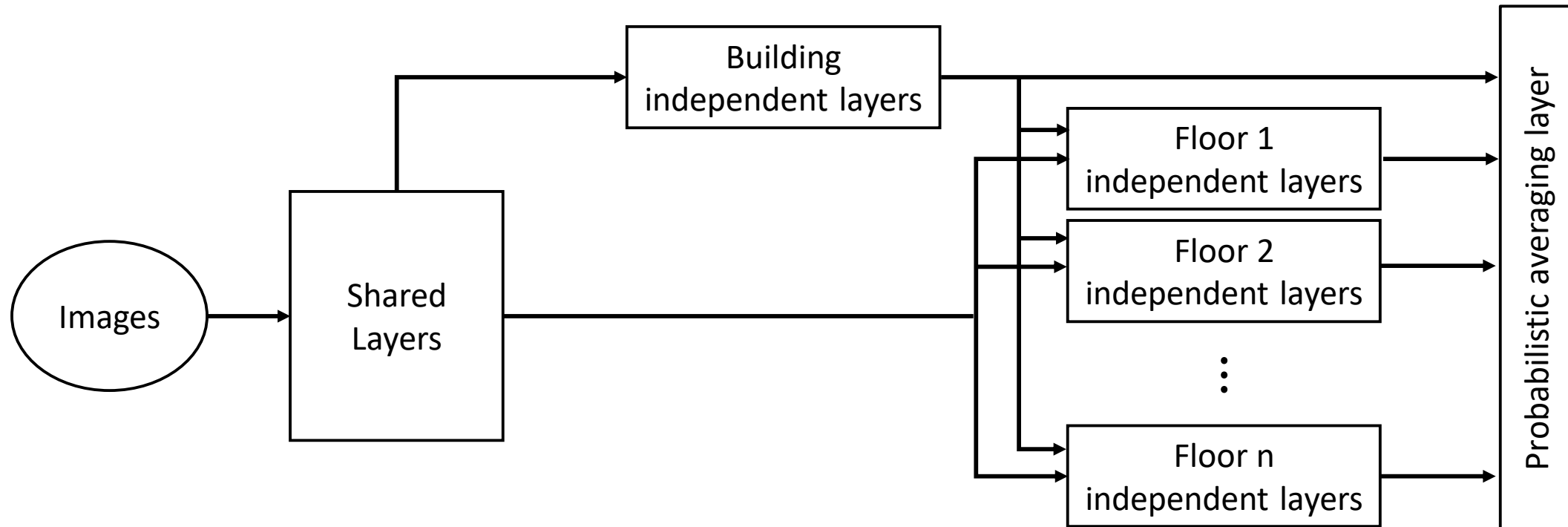
- Image classification

- Give subsampling image step by step
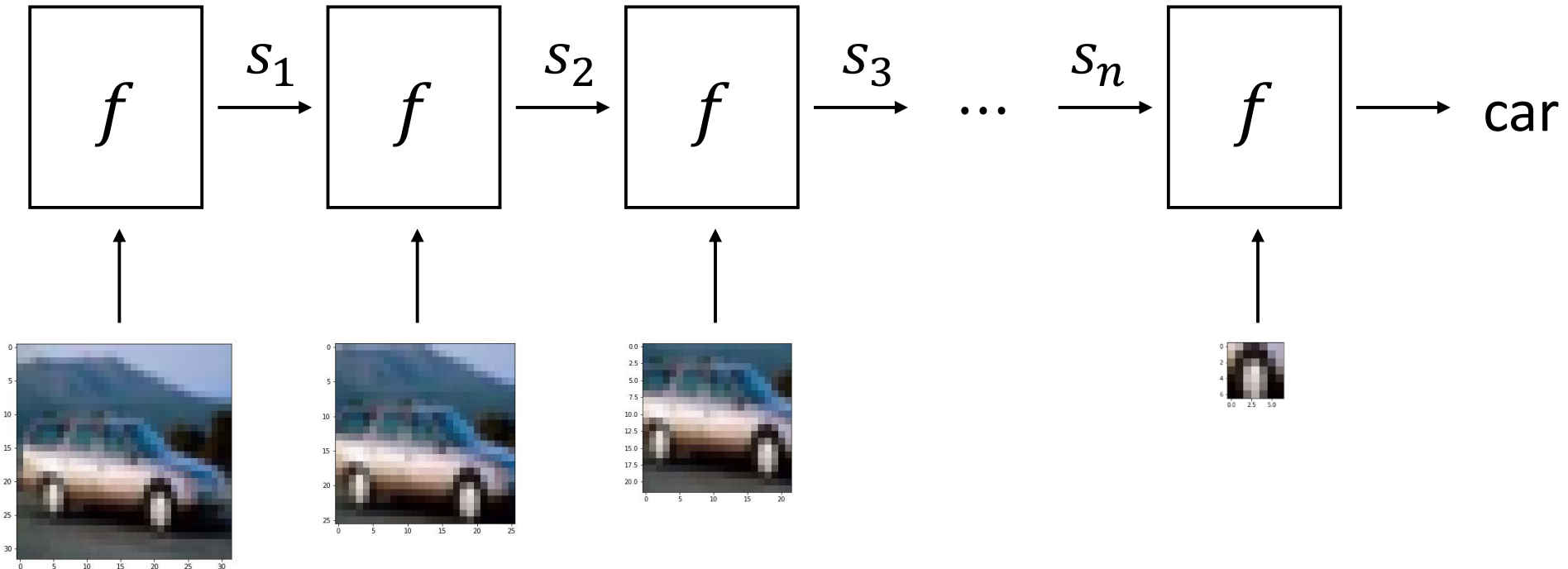
Hierarchical Deep Convolution Neural Network

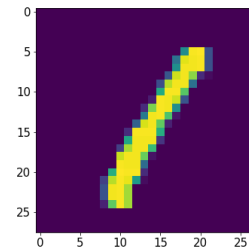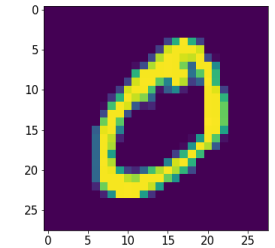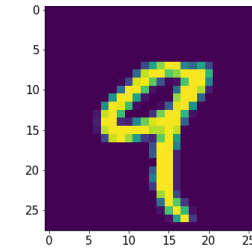- Image classification

- Give subsampling image step by step

Hierarchical Deep Convolution Neural Network

- Image classification

- Give subsampling image step by step

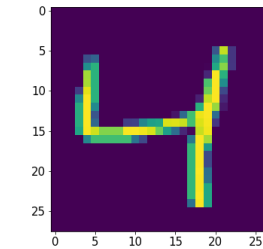# Hierarchical Auxiliary Learning
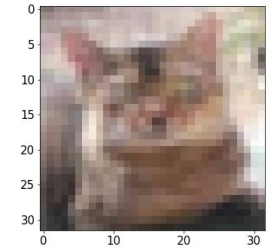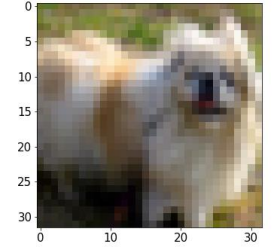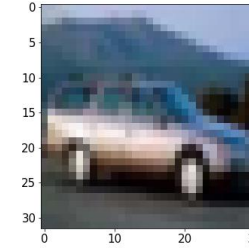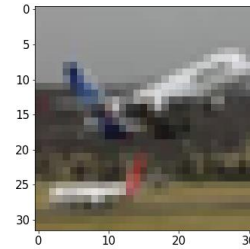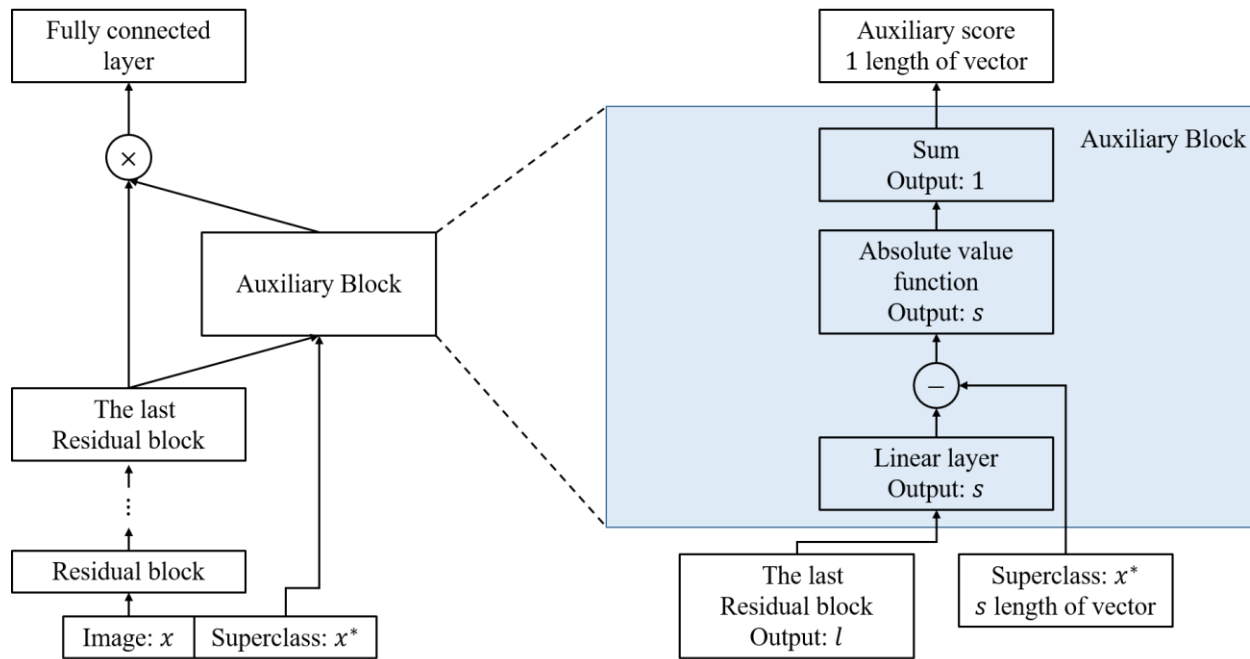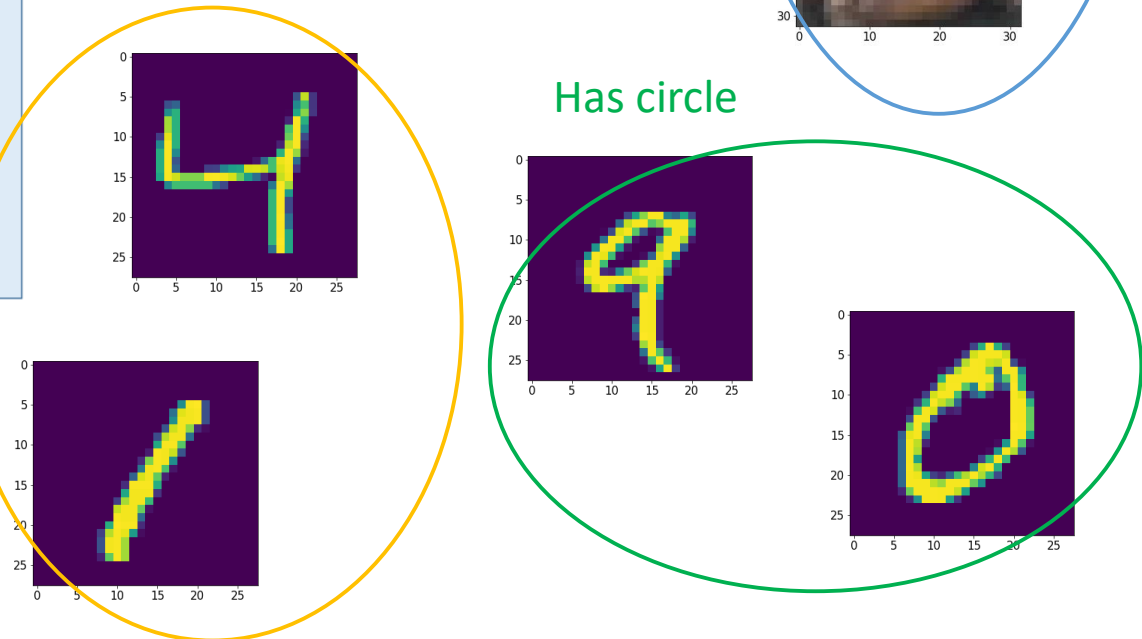
Hierarchical Auxiliary Learning

Transportation

Animal

Has circle

Straight lines
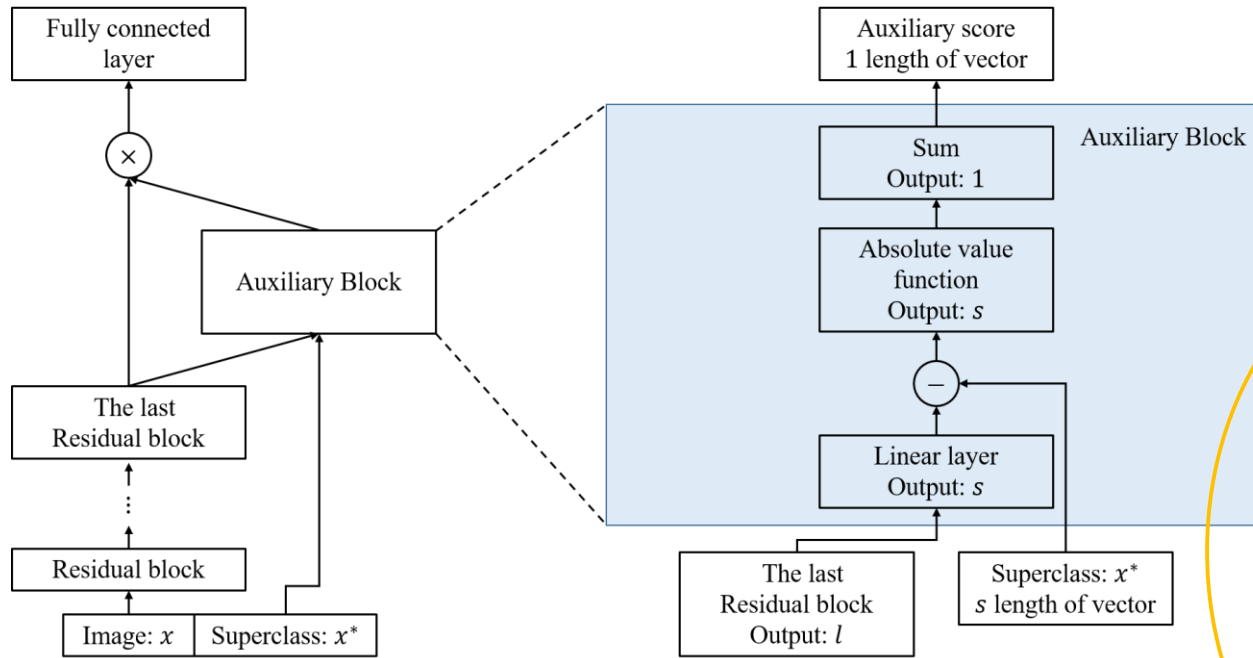
Fully connected layer

×

Auxiliary Block

The last Residual block

Residual block

Image: x Superclass: x^*

Auxiliary score
1 length of vector

Auxiliary Block

Sum
Output: 1

Absolute value function
Output: s

−

Linear layer
Output: s

The last Residual block
Output: l

Superclass: x^*
s length of vector

Hierarchical Auxiliary Learning
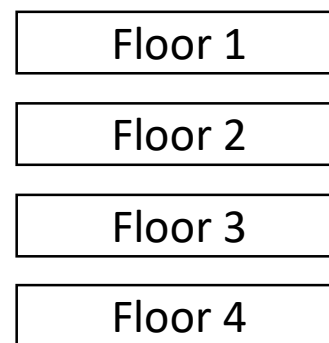
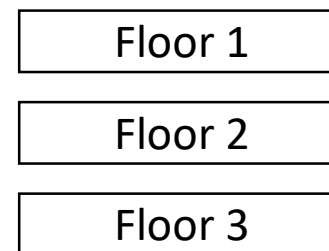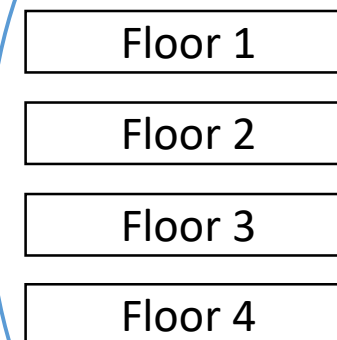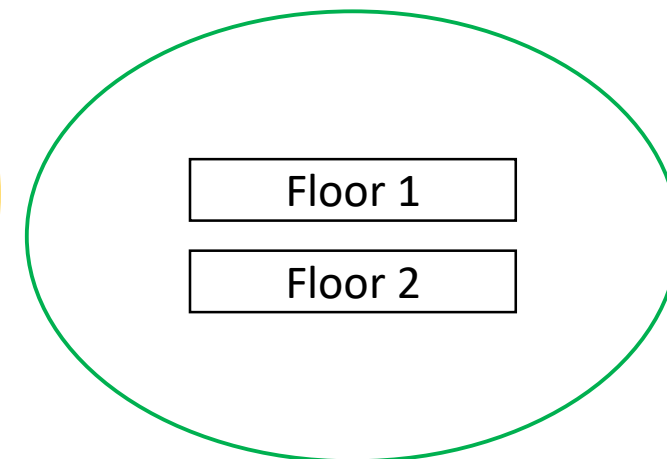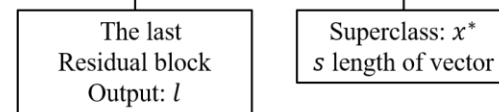| Dataset	Class
MNIST, SVHN	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
CIFAR-10	Airplane, car, bird, cat, deep, dog, frog, horse, ship, truck

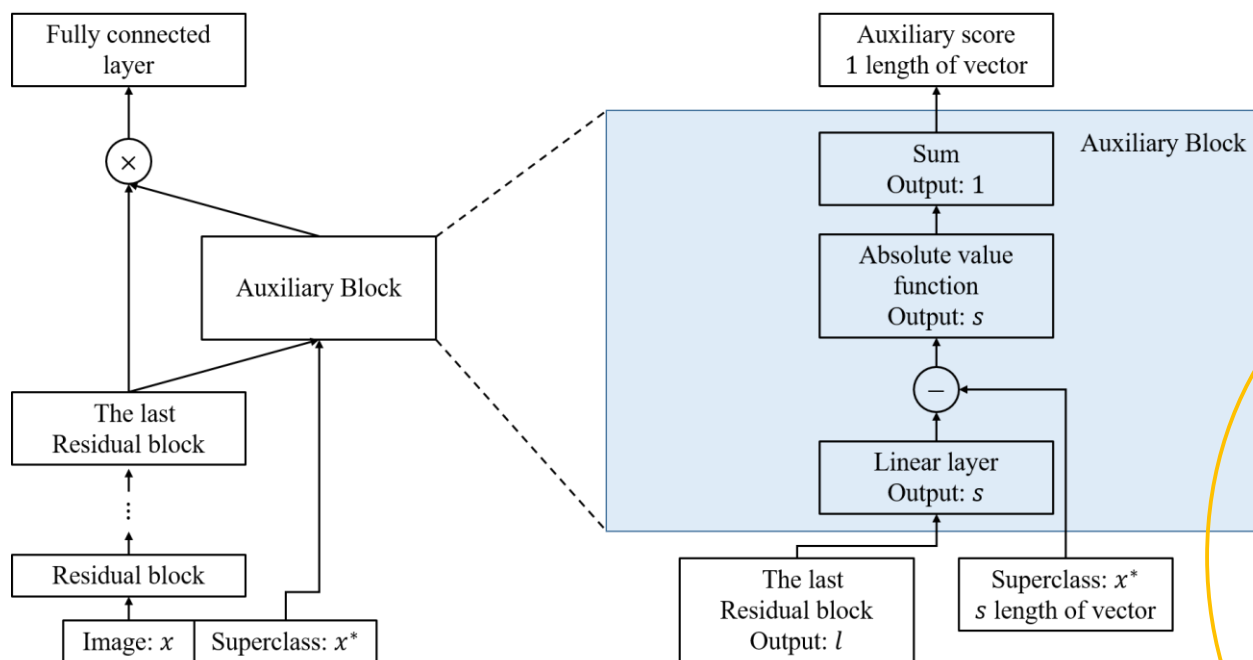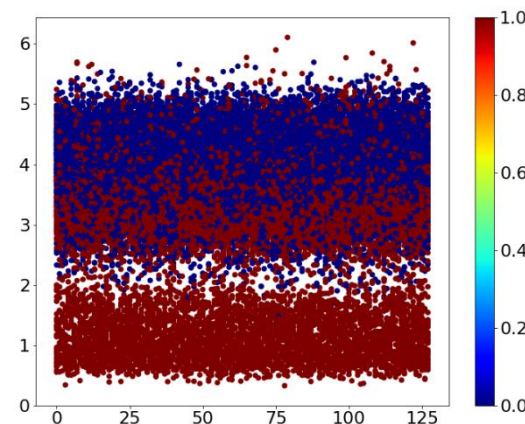| Dataset	Case	Semantics	Superclass
MNIST	Case1	≥ 5	0:{5, 6, 7, 8, 9}, 1:{0, 1 ,2 ,3 ,4}
SVHN	Case2	Mod2	0:{1, 3, 5, 7, 9}, 1:{0, 2, 4, 6, 8}
	Case3	Prime	0:{2, 3, 5, 7}, 1:{0, 1, 4, 6, 8, 9}
	Case4	Circle/ curve/ straight line	0:{0, 6, 8, 9}, 1:{2, 3, 5}, 2:{1, 4, 7}
CIFAR-10	Case1	None	0:{5, 6, 7, 8, 9}, 1:{0, 1 ,2 ,3 ,4}
	Case2	None	0:{1, 3, 5, 7, 9}, 1:{0, 2, 4, 6, 8}
	Case3	Transportation/ animal	0:{2, 3, 4, ,5, 6, 7}, 1:{0, 1, 8, 9}
	Case4	Car/ small animal/ big animal/ craft/ others	0:{1, 9}, 1:{3, 5}, 2:{4, 7}, 3:{0, 8}, 4:{2, 6}

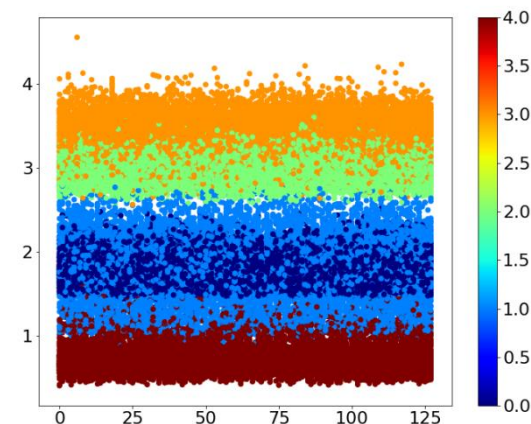| MNIST	baseline	Case1	Case2	Case3	Case4
Error	0.93	0.43±0.03	0.73±0.06	0.70±0.05	0.69±0.00
SVHN	Baseline	Case1	Case2	Case3	Case4
Error	4.05	2.53±0.06	2.64±0.11	2.66±0.07	2.86±0.07
CIFAR-10	baseline	Case1	Case2	Case3	Case4
Error	6.81	3.30±0.06	5.30±0.14	6.46±0.08	5.13±0.09
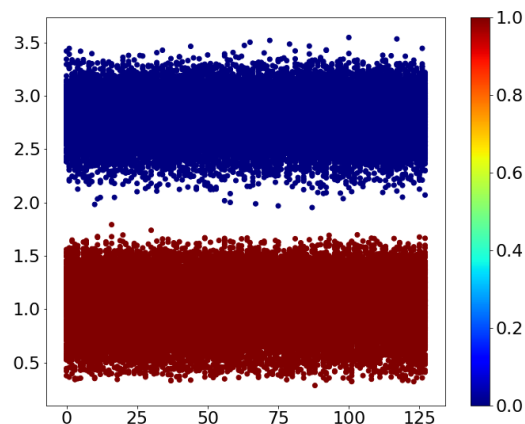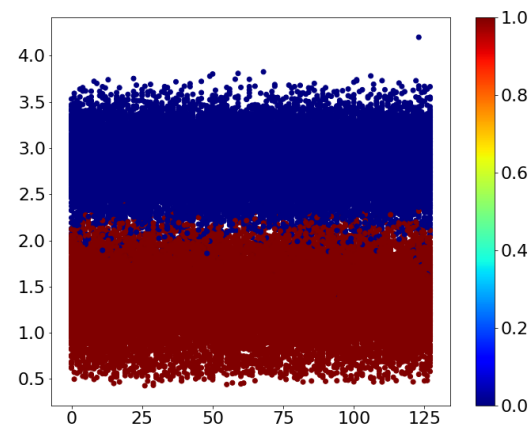
CIFAR-10-Case1   CIFAR-10-Case2   CIFAR-10-Case3   CIFAR-10-Case4

Thank you