Intro to Recurrent Neural Networks

Tue 13th November 2018 5pm - 6:30pm

K

PyData
Bristol

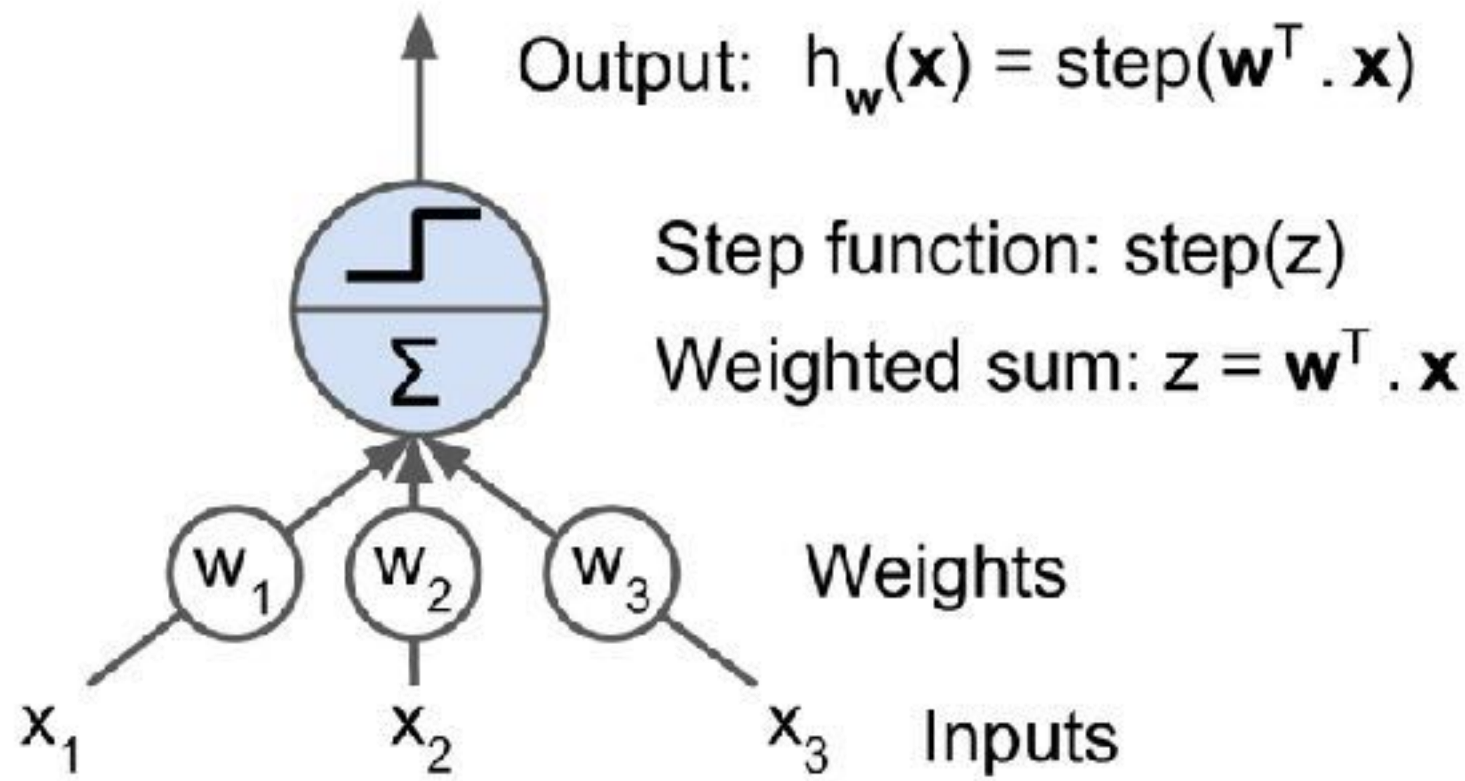
TensorFlow

RECAP: ARTIFICIAL NEURAL NETWORKS

The Linear Threshold Unit

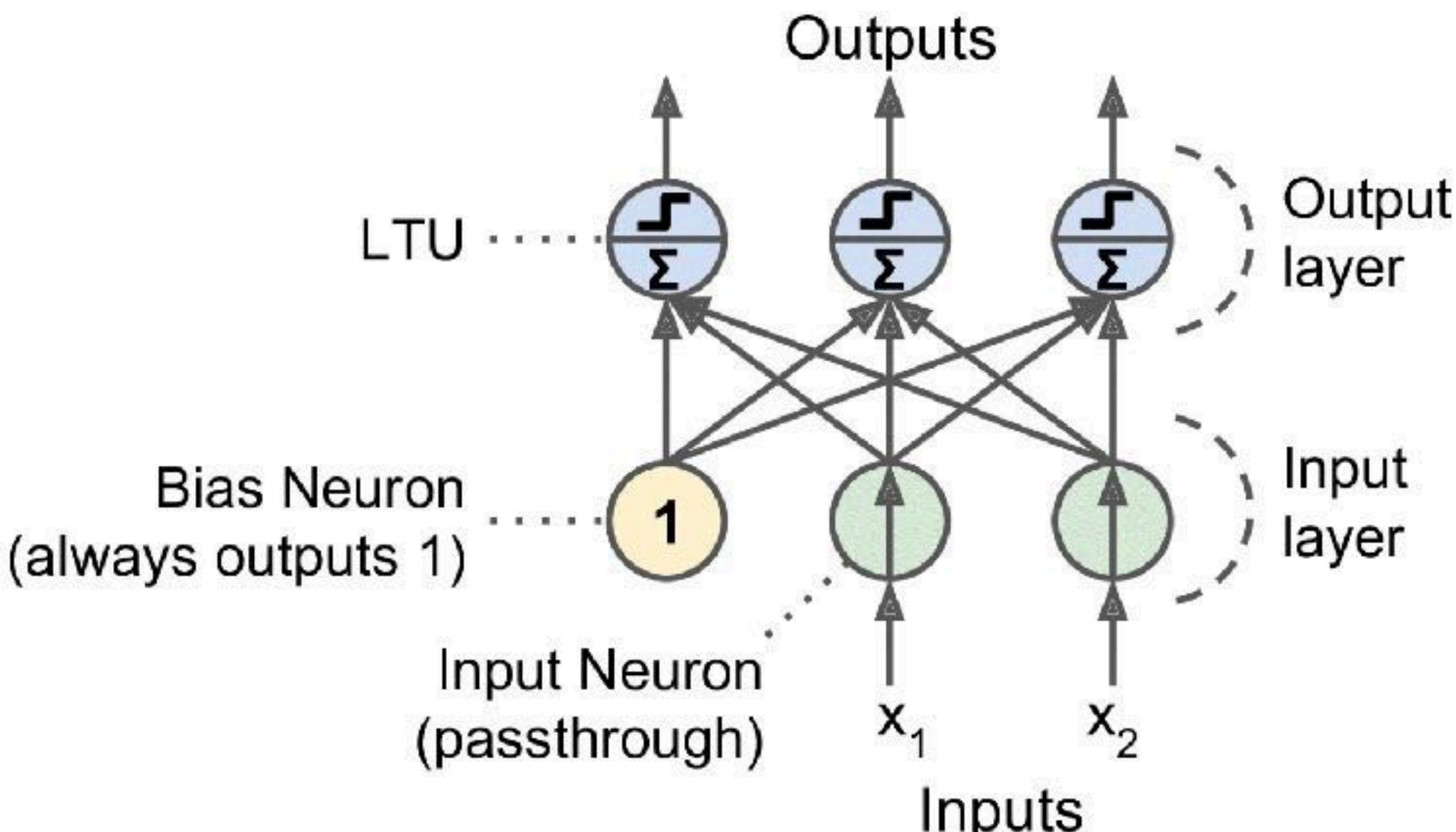
- 1. Compute **weighted sum of inputs** (linear, i.e. dot product)
- 2. Apply **step function** (e.g. *heaviside* or *sign)*

$$\text{heaviside}\,(z) = \begin{cases} 0 & \text{if } z < 0 \\ 1 & \text{if } z \geq 0 \end{cases}$$



The Perceptron

- A **Perceptron** is simply a **layer of LTUs** plus a **bias** term.
- Every input feature flows into every LTU.
- The Perceptron shown below can model three-class binary classification problems.



RECURRENT NEURAL NETWORKS

Recurrent Neural Networks

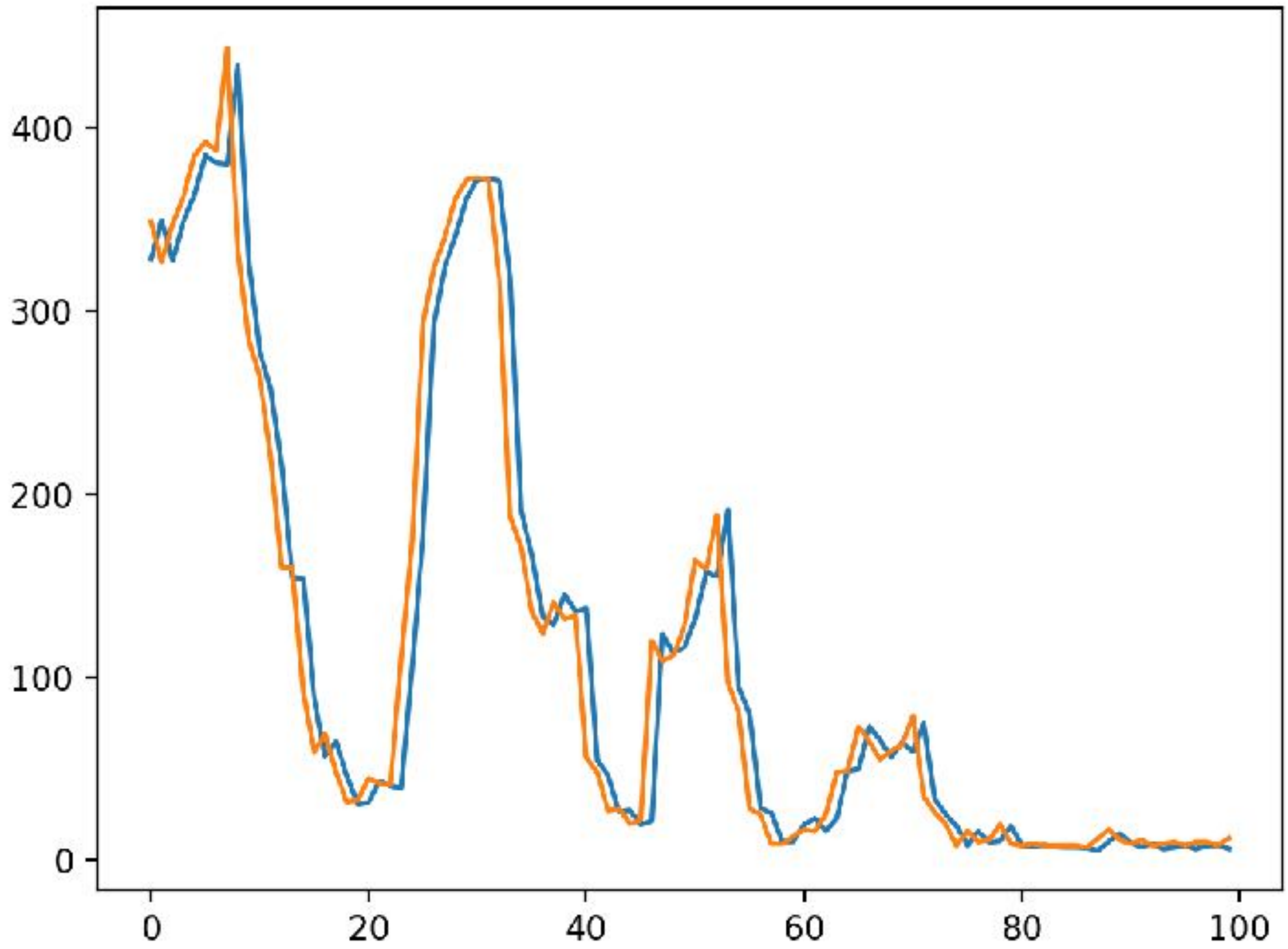
- Predict the future!

- Forecast stock prices, anticipate car trajectories in autonomous vehicles, translate words into other languages.

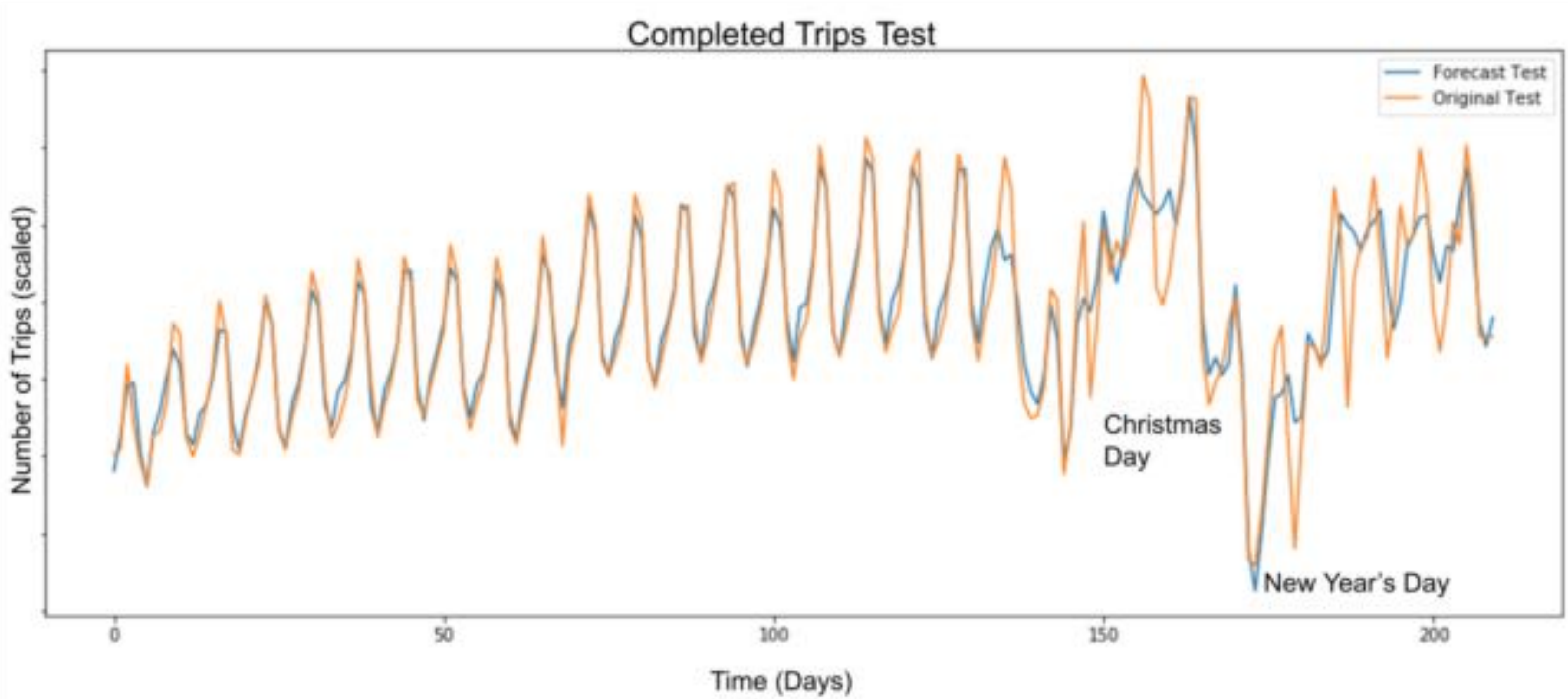
- Many applications in **time series** and **NLP** (natural language processing), e.g. machine translation, speech-to-text, sentiment analysis and more!

- **Creative applications**: generating words, sentences, music, knitting patterns, image captions...

Applications: Time Series Forecasting



Applications: Time Series Forecasting



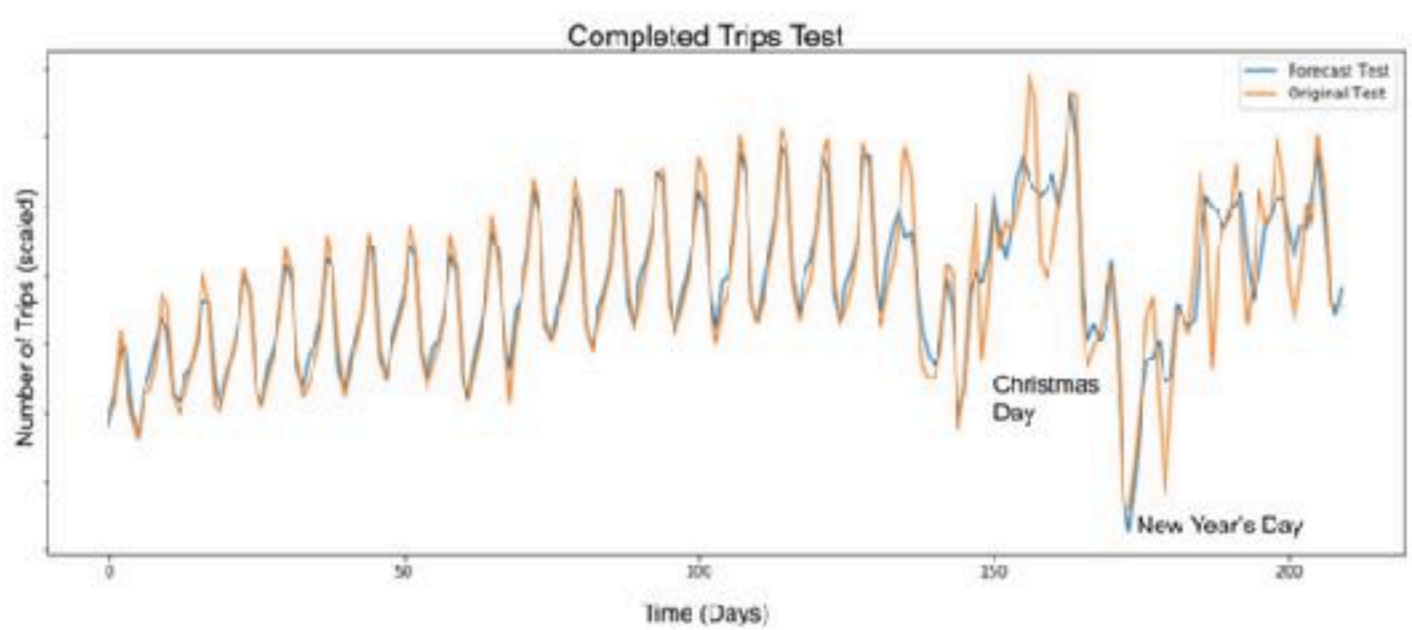
Applications: Time Series Forecasting

Uber Engineering

Uher Data

Engineering Extreme Event Forecasting at Uber with Recurrent Neural Networks

June 9, 2017



Applications: Image Captioning

Applications: Text Summarisation

- Original text
  - `Alice and Bob took the train to visit the zoo. They saw a baby giraffe, a lion, and a flock of colourful tropical birds.`
- Summary
  - `Alice and Bob visited the zoo and saw animals and birds.`

Applications: Generating English Towns

https://medium.com/@hondanhon/i-trained-a-neural-net-to-generate-british-placenames-9460e907e4e9

Applications: Generating Shakespeare

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never
fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.
```

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Applications: Generating Knitting Patterns

Applications: Generating Knitting Patterns

Janelle Shane @JanelleCShane · Feb 17

Nobody attempted to test-knit this one, for some reason.

Waist Row RS: K1. Row 1 RS: P1, k2tog, yo twice more times. 10 sts). Rows 3, 4, 6, 3, 4, 4 times.

Row 2 (RS): K1, work in pattern to end. Row 3 RS: Work in pattern to end.

Row 4 RS: K1, M1, k1, p1, repeat from * to end.

Repeat these 2 rows in pattern.

Rows 3 and 8 WS: Work in pattern as set. Row 8 WS: P1. 6395, 71, 70, 77 sts.

Row 3 WS: K across. Row 4 (WS): K1, p1, work in pattern to end. Rows 4 and 5 WS: K1, (k1, p1] to-end. Row 4 (WS): K2tog, yo, k to last st, k1. 5797, 73. Row 3 WS: P1, k5. Row 3 (WS): K3, p1, k1, p1. 7752, 55, 164, 146 sts.

Applications: Generating Knitting Patterns

Janelle Shane
@JanelleCShane

Follow

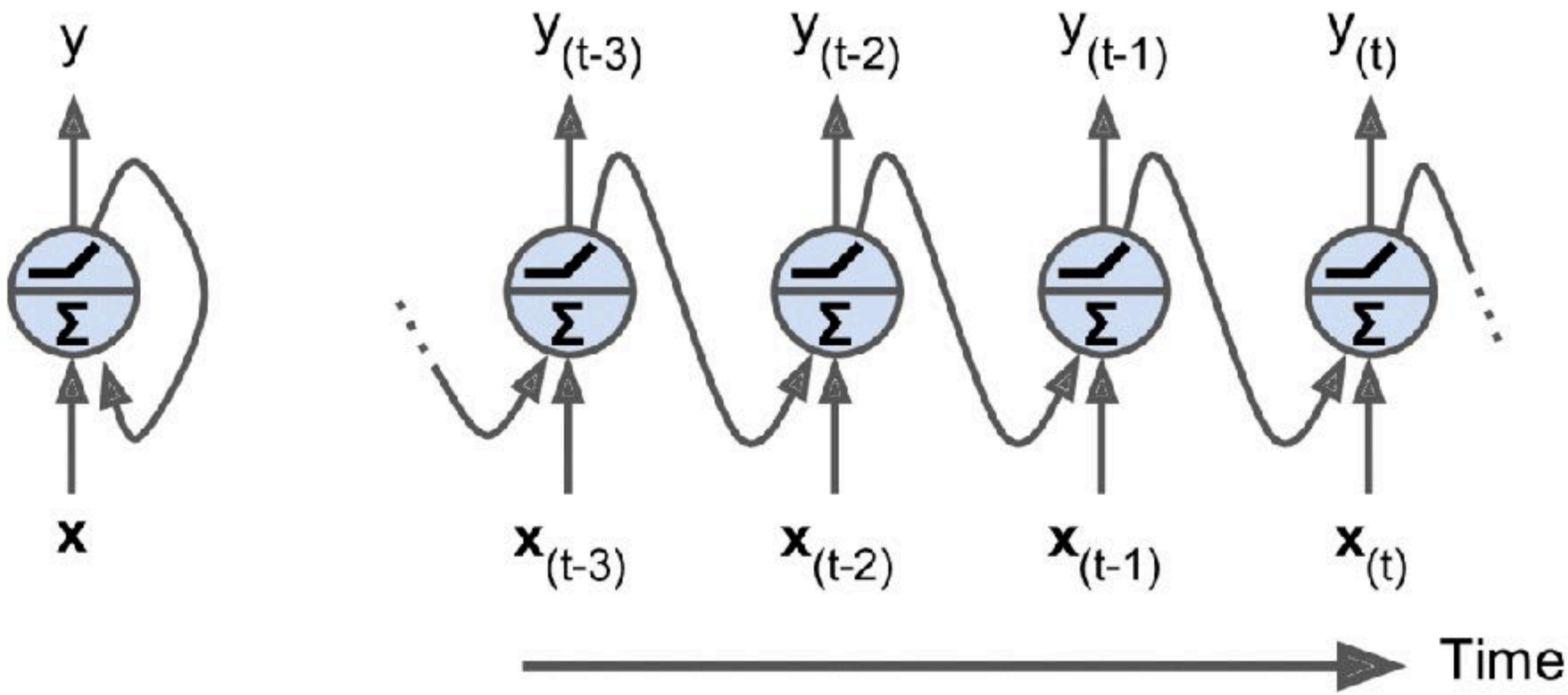
I learned knitters are amazing at debugging, used to fixing all sorts of pattern problems. They took most of the neural net's mistakes in stride.
Test-knitting: CorvusAlatus ravelry.com/discuss/lazy-s ...

7:20 PM - 17 Feb 2018

58 Retweets 313 Likes

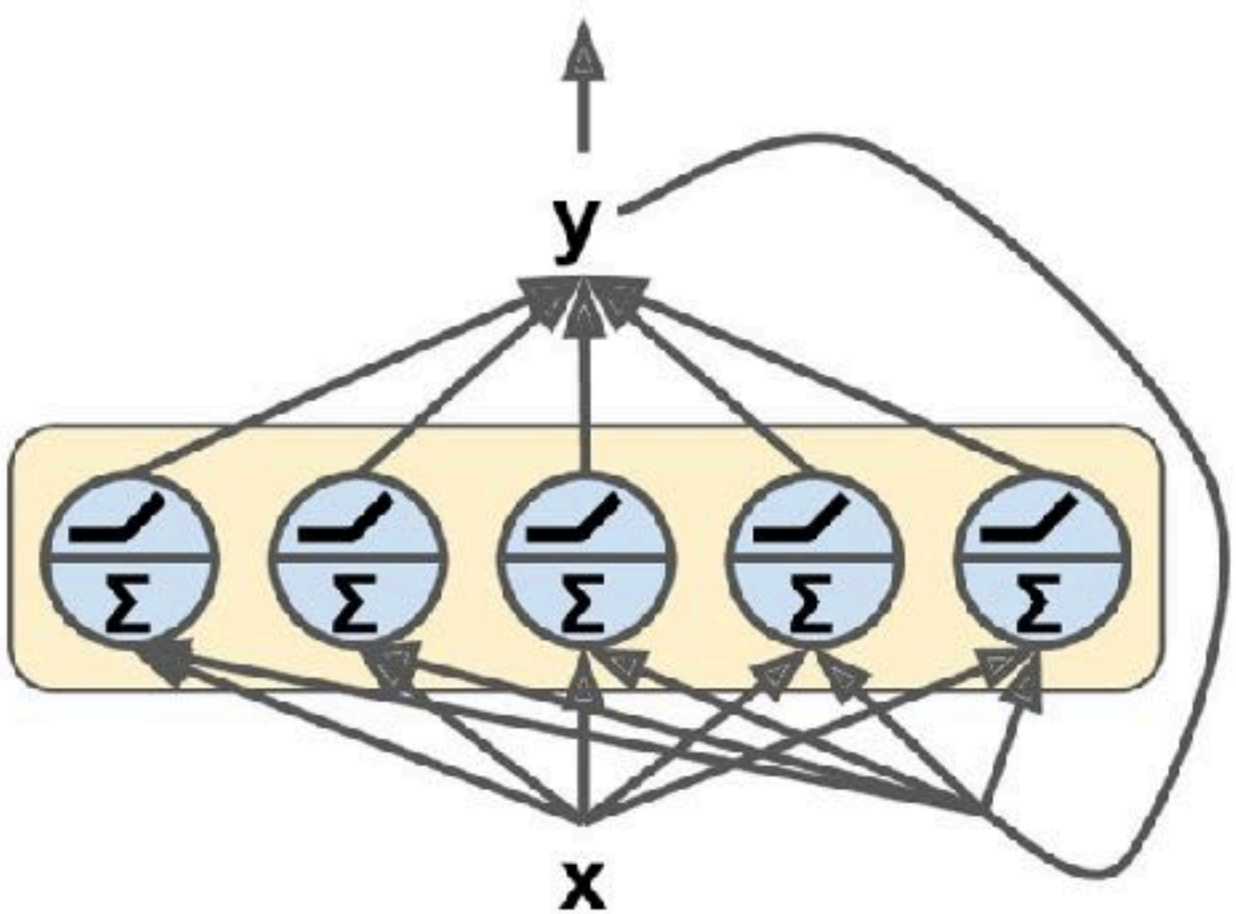
Architecture of RNNs

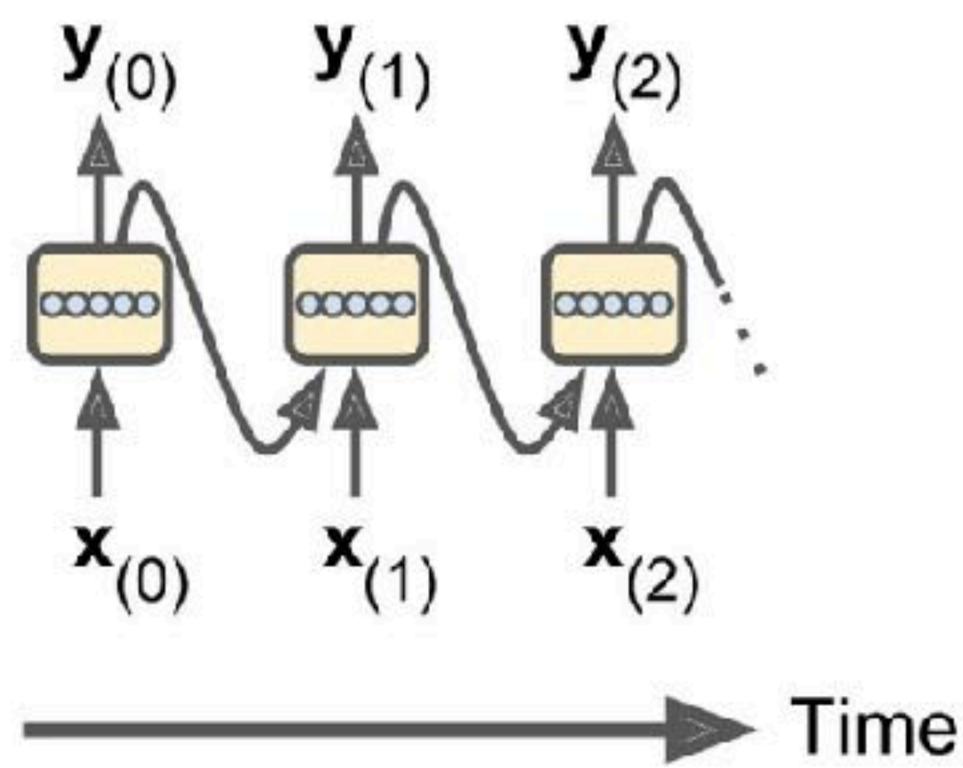
- RNNs are neurons (e.g. LTUs) that feed back into themselves.
- Simple RNN
- For a time series, we have multiple $\boldsymbol{x_t}$ over time, also outputs $\boldsymbol{y_t}$
- Here's what the network looks like "**unrolled through time**"



Cells are layers of recurrent neurons

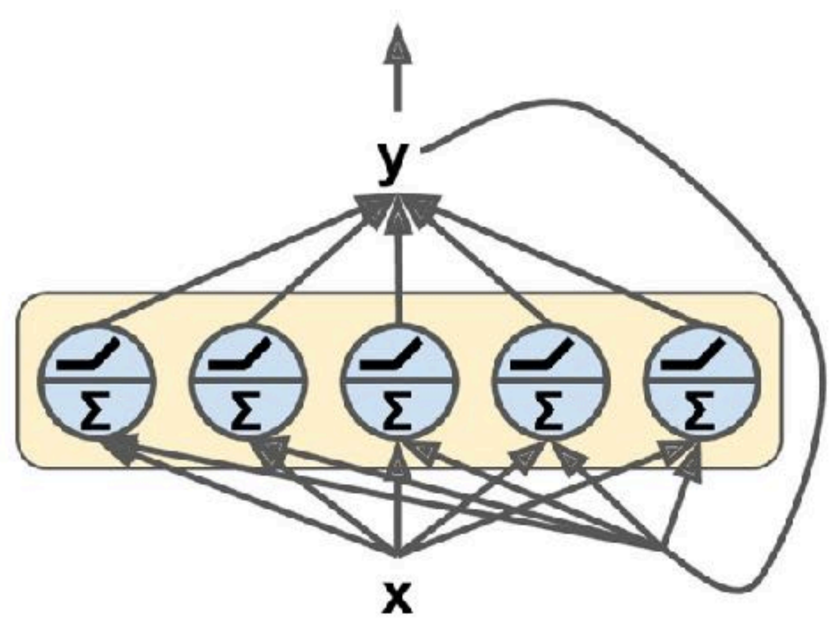
- A **cell** contains multiple recurrent neurons.
- Every neuron receives same input (one $\boldsymbol{x_t}$ and one $\boldsymbol{y_{t-1}}$)
- Every neuron generates an output.
- Combine all the neurons' output to form $\boldsymbol{y_t}$
- We can also unroll the cell through time.





What are the weights?

- Our input $\boldsymbol{x_t}$ can be a vector of multiple features (e.g. using AAPL stock price and S&P500 index gives two features).
- Our output $\boldsymbol{y_t}$ depends on how many neurons are in the cell, e.g. 3 neurons gives a vector of length three.
- Neurons in the cell have following weights:
  - $\boldsymbol{W_x}$ links 2-feature $\boldsymbol{x_t}$ input to 3 neurons, i.e. has shape (2,3)
  - $\boldsymbol{W_y}$ links 3-neuron $\boldsymbol{y_{t-1}}$ input to 3 neurons, i.e. has shape (3,3)
  - Like LTU, the cell also has a bias vector of shape (3,)



LAB:
BUILD A RECURRENT NEURAL NETWORK