

Building Machine Learning Systems with TensorFlow.js #1

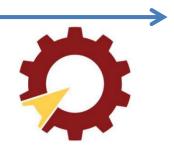
Asst.Prof.Dr.Supachai Vorapojpisut

Thammasat University





Outline



Jan 7, 2020

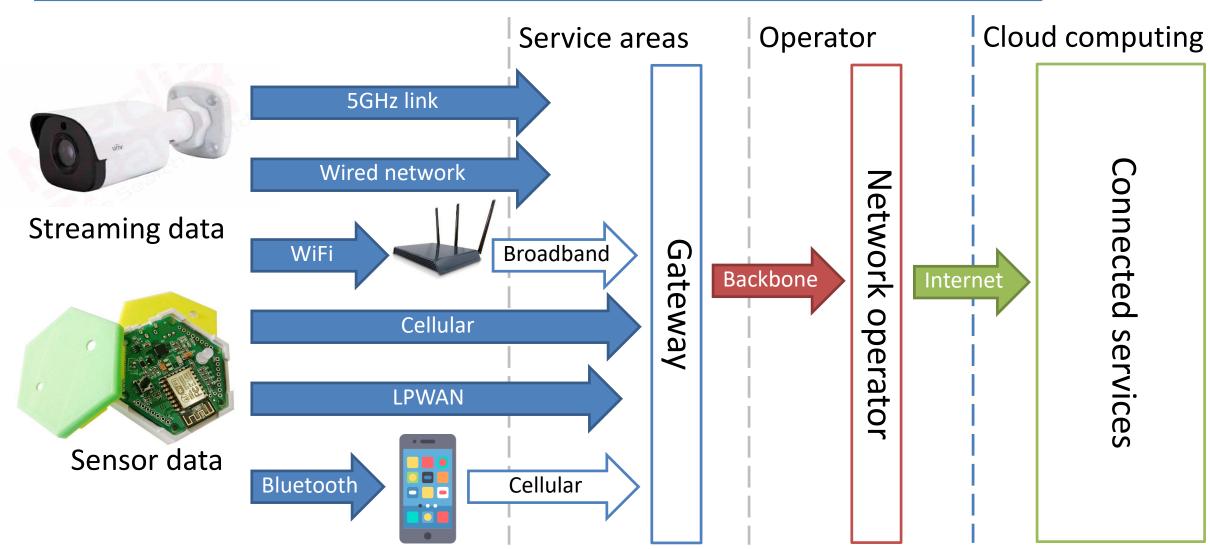
- Working with data
- Machine learning concepts
- Time-series prediction
- ML coding
 - jest
 - TensorFlow.js

Jan 8, 2020

- Machine learning system design
- SpatioTemporal data analytics

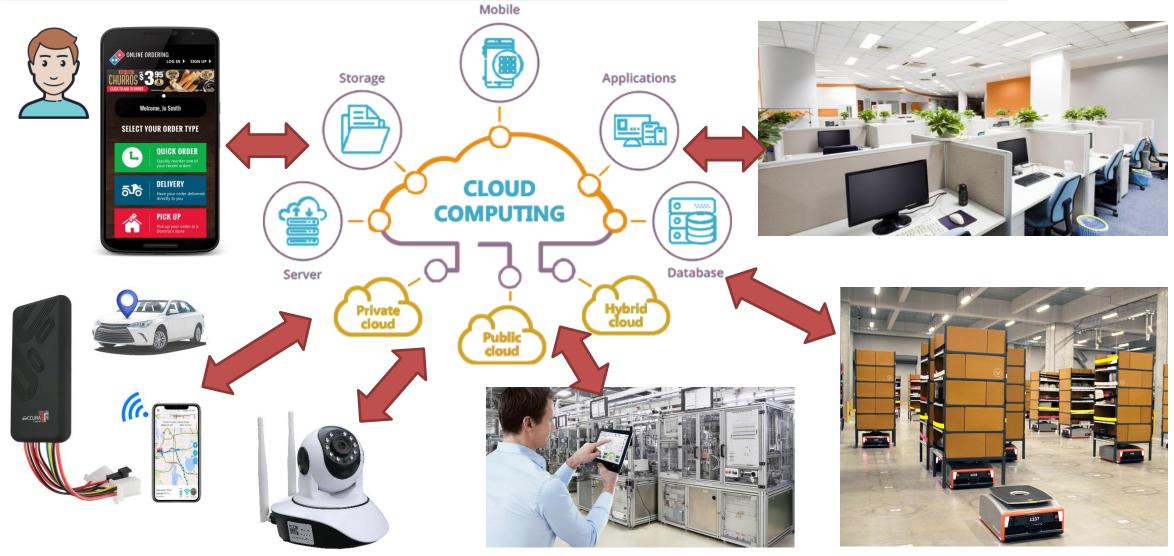
IoT as enabler technology



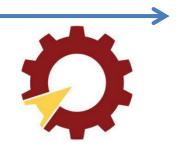


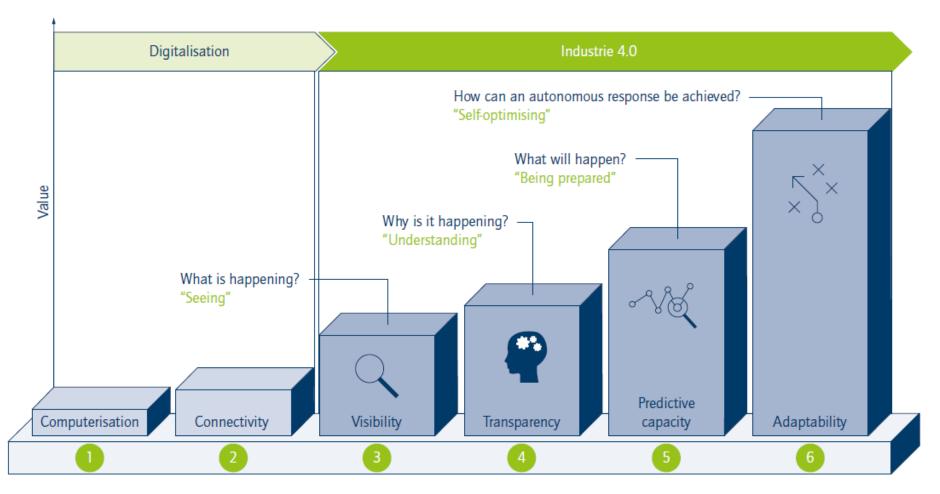
Data in modern world





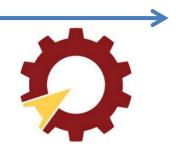
Industry 4.0





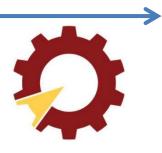
acatech STUDY: Industrie 4.0 Maturity Index, Managing the Digital Transformation of Companies

Working with data



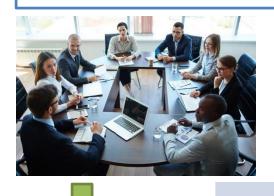
- Data perspective
 - Data is uncertain quantities of interest
 - Data is usually obtained empirically or experimentally
 - Data set usually contains non-numeric parts
- Model perspective
 - Model is obtained statistically
 - Model quality depends on data size, completeness and correctness
- Problem perspective
 - Objectives are usually explained by natural language
 - Decisions are made based on data, models and rules

Machine learning concepts



Scope

The science of getting computers to act without being explicitly programmed.



Data science

Machine learning

Data preparation Exploratory data analysis

Data visualization

Classification & clustering

Predictive analytics

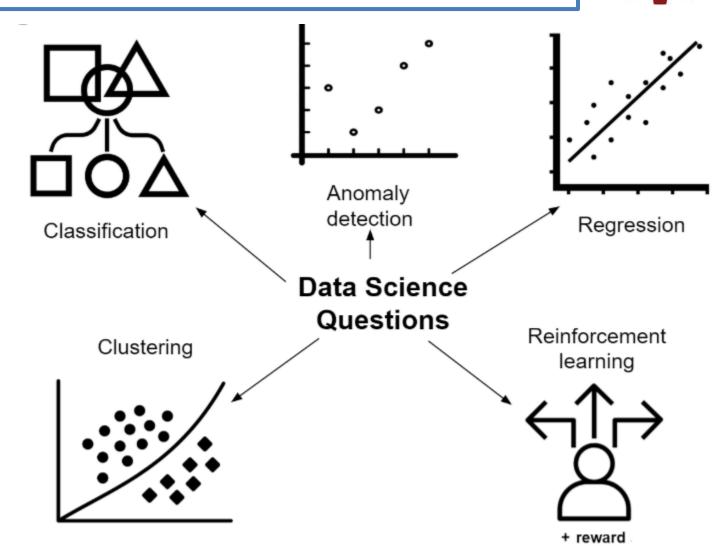
Al

Questions for data analytics

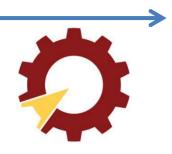


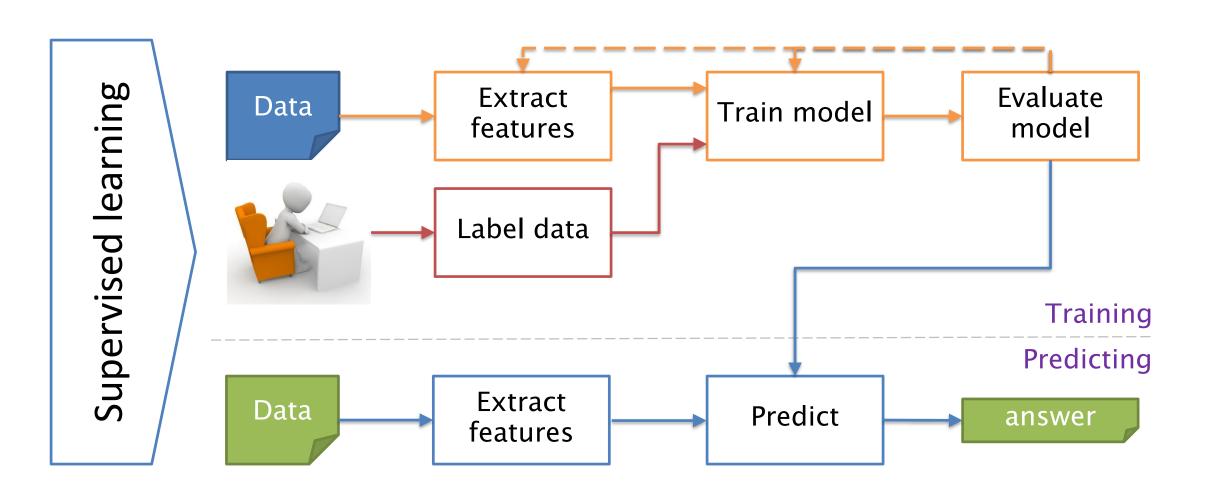
How to make questions?

- 1. How business will use results
- 2. Required analytic results
- 3. Technique to obtain analytic results
- 4. Data source, volume, speed
- 5. Data infrastructure

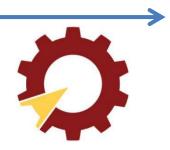


Supervised learning

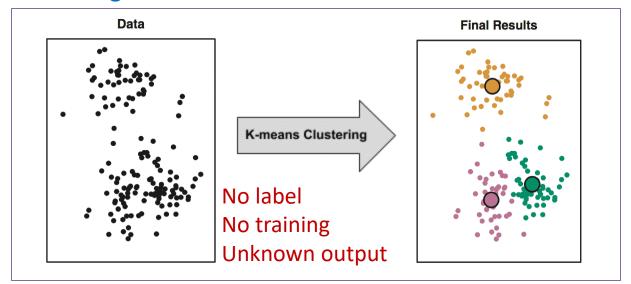




Unsupervised learning

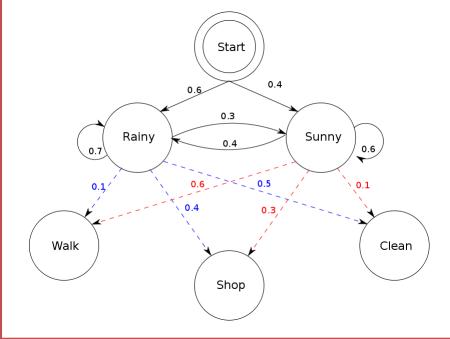


Clustering

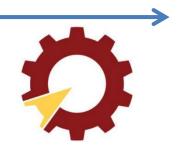


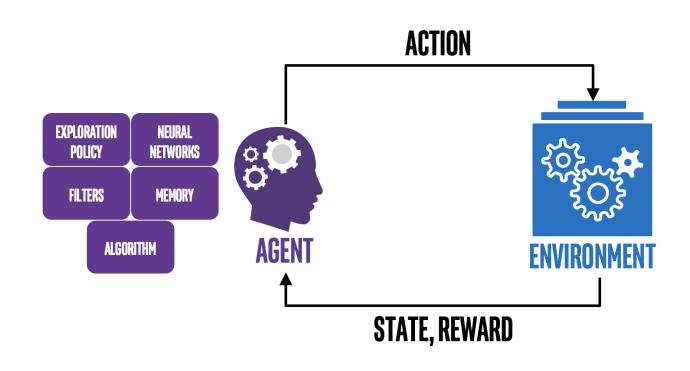


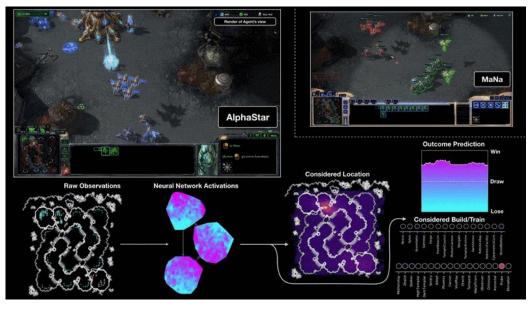
Hidden Markov Model



Reinforcement learning



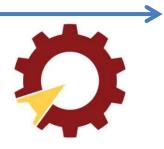






TEST BEFORE DEBUG, OR DEBUG BEFORE TEST

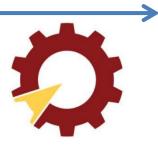
Unit testing



```
> npm init
  test command: jest
> npm run test
app.js
function append(x,y) {
  return x+y;
module.exports.append = append;
```

```
app.test.js
   beforeEach(() => {
     initializeMockupDatabase();
                                               Setup
   });
   const mycode = require(./mycode);
   test('append 1,2 be "12"', () => {
     expect(app.append(1, 2)).toBe('12');
   });
   afterEach(() => {
                                               Teardown
     clearMockupDatabase();
   });
```

Data preprocessing



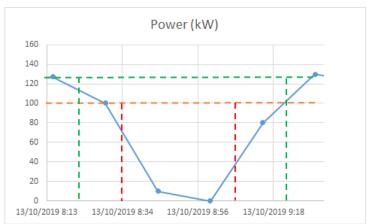
Timestamp	Power (kW)
13/10/2019 8:00	120
13/10/2019 8:15	127
13/10/2019 8:30	100
13/10/2019 8:45	10
13/10/2019 9:15	80
13/10/2019 9:30	130

Data cleansing /

Data editing

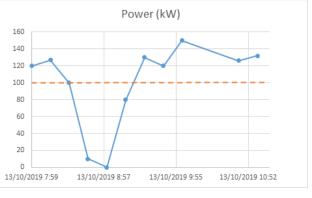
Timestamp	Power (kW)	Event
13/10/2019 8:00	120	NORMAL
13/10/2019 8:15	127	NORMAL
13/10/2019 8:30	100	SHUTDOWN
13/10/2019 8:45	10	SHUTDOWN
13/10/2019 9:00	0	SHUTDOWN
13/10/2019 9:15	80	SHUTDOWN
13/10/2019 9:30	130	NORMAL

ABNORMAL 13/10/2019 8:22 - 9:20 SHUTDOWN 13/10/2019 8:34 - 9:07 Data wrangling

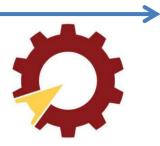


Data reduction

Timestamp	Power (kW)
13/10/2019 8:15	127
13/10/2019 8:30	100
13/10/2019 8:45	10
13/10/2019 9:00	0
13/10/2019 9:15	80
13/10/2019 9:30	130



Data import



Array of

Object

JSON file

```
{
   "firstName": "Supachai",
   "lastName": "Vorapojpisut",
   "isAlive": true
}
```

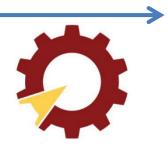
```
const fs = require('fs');
const rawdata = fs.readFileSync('???');
let data = JSON.parse(rawdata);
```

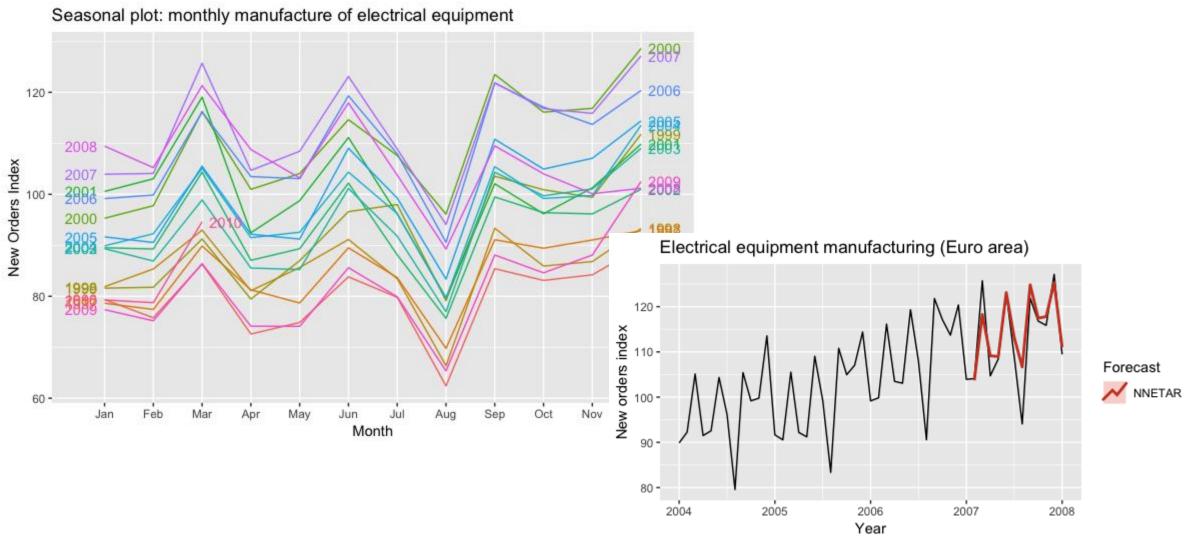
CSV file

```
city, state, population, area seattle, WA, 652405, 83.9 new york, NY, 8405837, 302.6 boston, MA, 645966, 48.3
```

```
const tf = require('@tensorflow/tfjs-node');
async function run() {
   const dataset = tf.data.csv('file://?', {ha
   sHeader: true});
   const v = await dataset.take(2).toArray();
}
run();
```

Time-series prediction



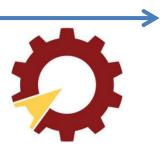


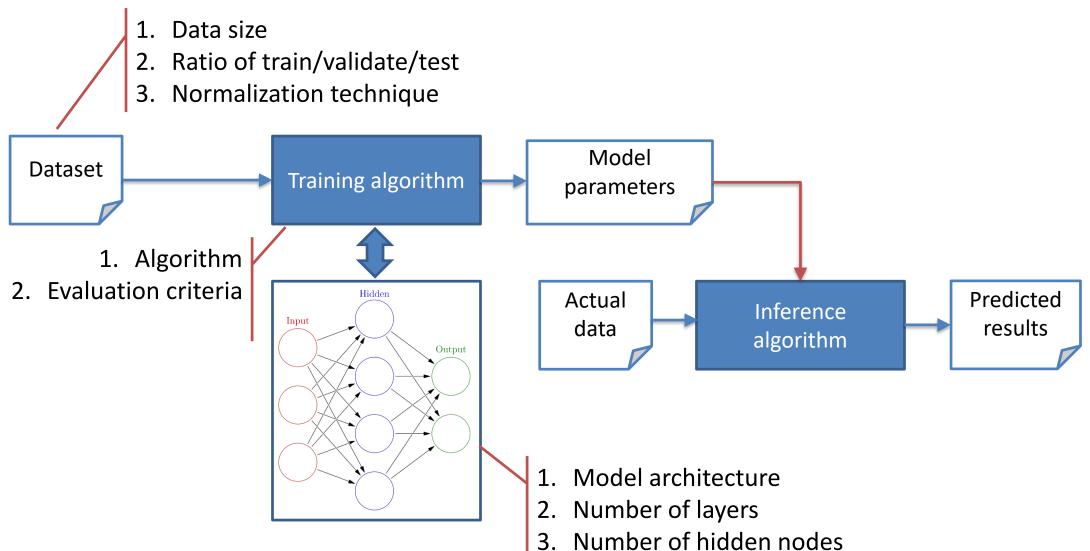
Neural networks



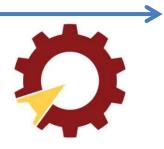
1980S-ERA NEURAL NETWORK DEEP LEARNING NEURAL NETWORK Hidden Multiple hidden layers layer process hierarchical features Input Input Output Output layer layer layer layer Output: 'George' Input Identify Node Identify combinations light/dark or features pixel value Identify Identify Identify Links carry signals edges combinations features from one node of edges to another, boosting or damping them according to each link's 'weight'.

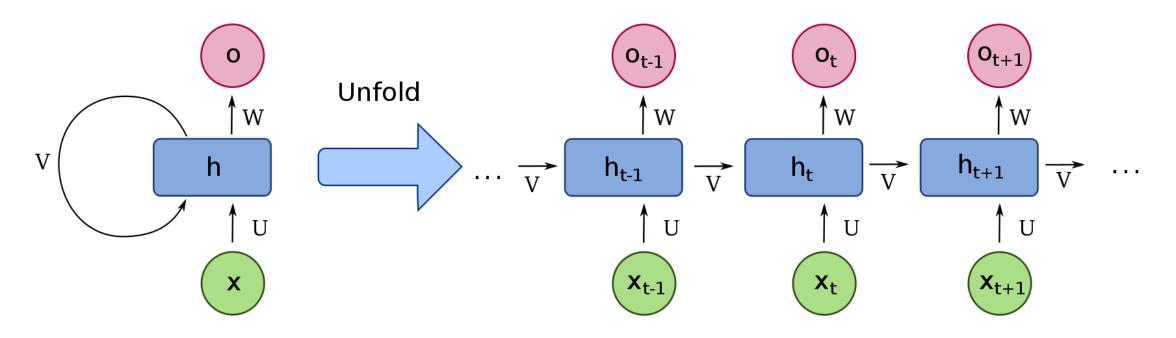
NN tuning → hyperparameters





Recurrent Neural Network



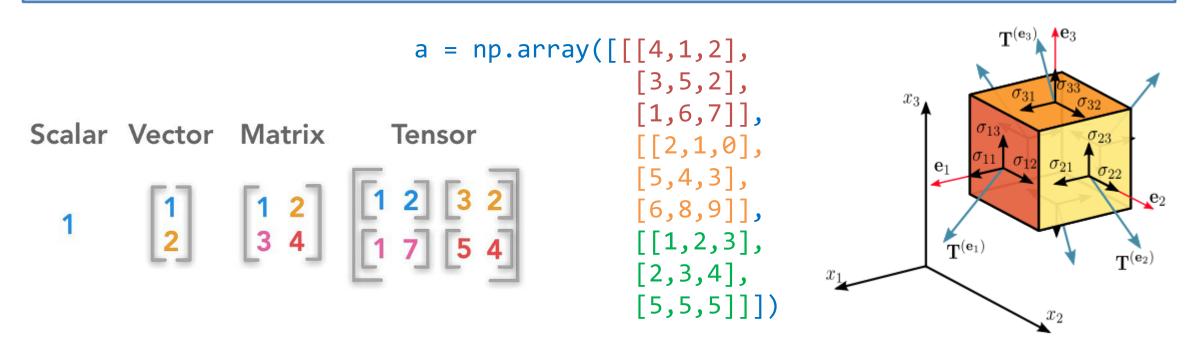


TensorFlow.js



Scope

JavaScript Library for training and deploying machine learning models in the browser and in Node.js



TensorFlow.js



> npm install @tensorflow/tfjs-node

```
const tf = require('@tensorflow/tfjs-node');
const model = tf.sequential();
model.add(tf.layers.dense({units: 100, activation: 'relu', inputShape: [10]}));
model.add(tf.layers.dense({units: 1, activation: 'linear'}));
model.compile({optimizer: 'sgd', loss: 'meanSquaredError'});
                                                                  Build model
const xs = tf.randomNormal([100, 10]);
const ys = tf.randomNormal([100, 1]);
                                          Prepare data
model.fit(xs, ys, {
  epochs: 100,
                         Train model
  callbacks: {
    onEpochEnd: (epoch, log) => console.log(`Epoch ${epoch}: loss = ${log.loss}`)
```