Machine learning: An Introduction

Vishal Upendran
IUCAA

Menu for today

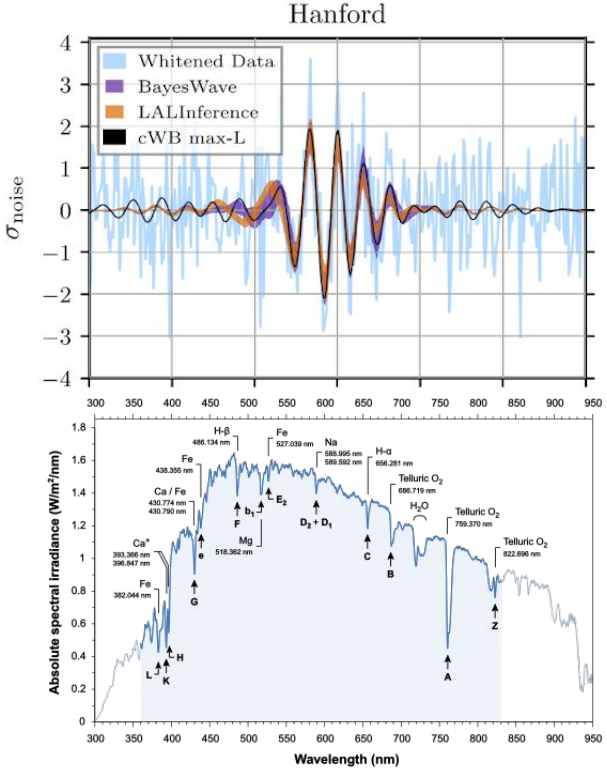
1. What is Machine learning?
2. When to use Machine Learning?
3. Machine learning algorithms.
4. Classification: exercise #1
    a. Using Scikit-learn
5. Regression: exercise #2
    a. Using Pytorch

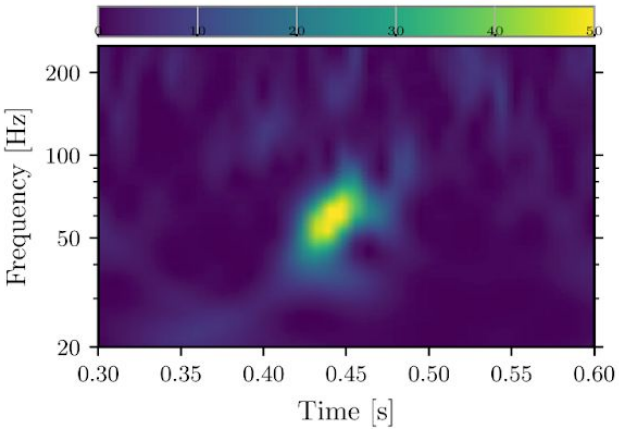
What is Machine learning?

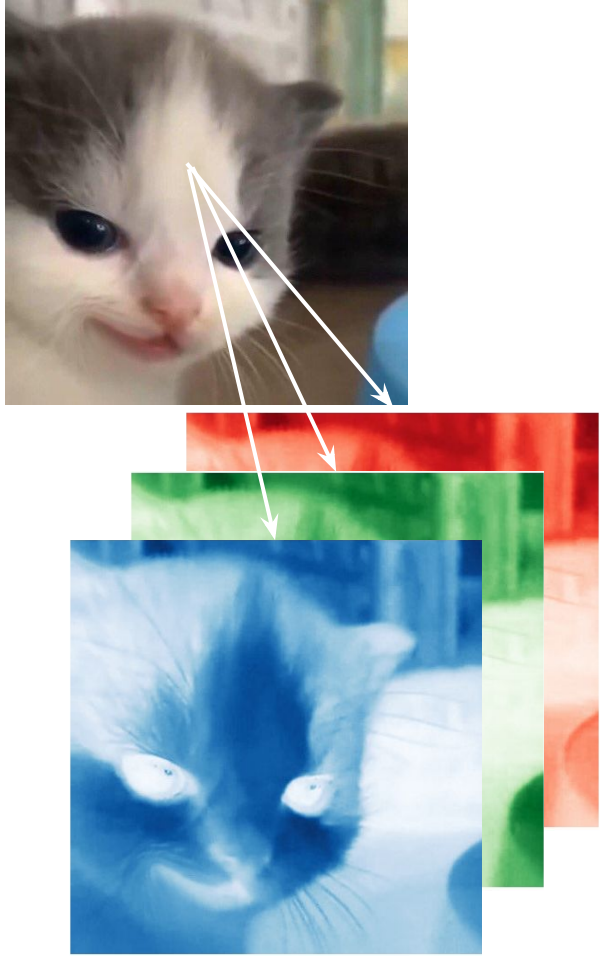
> Machine learning is the study of computer algorithms that improve automatically through experience and by the use of data.
>
> – *Wikipedia*

- Computer algorithms.
- Improve automatically.
- Use data.

Different kinds of data







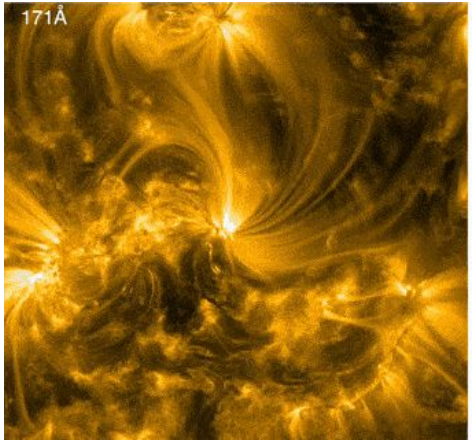
अग्निमीळे पुरोहितं यज्ञस्य देवमृत्विजम् । होतारं रत्नधातमम् ॥

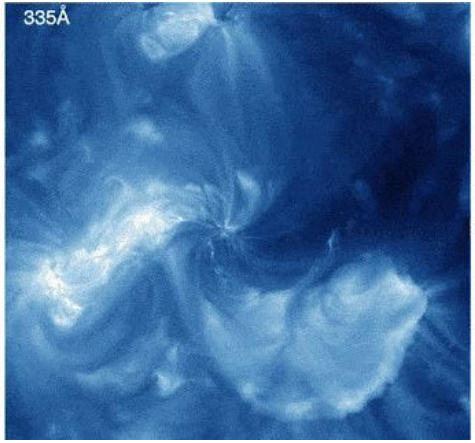
1-D Data

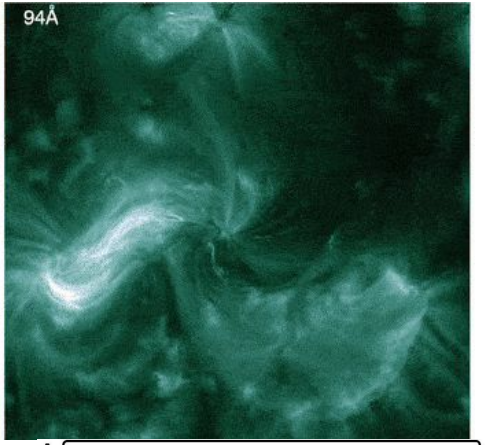
2-D Data

3-D Data

Different kinds of data



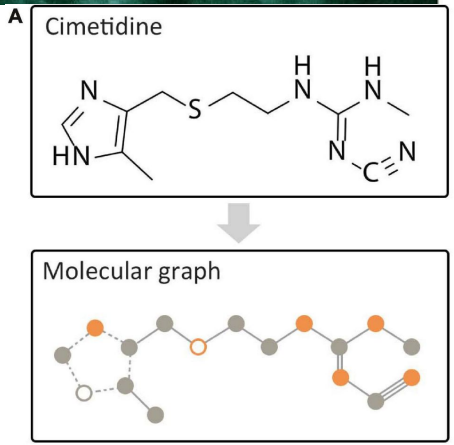




4-D data

ra	dec	u	g	r	i	z	run	rerun
318.951692	9.315146	19.51665	18.50036	17.95667	17.53139	17.32035	7777	301
217.940001	14.608378	19.13548	18.55482	17.95603	17.68272	17.63717	5322	301
129.948221	25.213328	19.54955	18.19434	17.83220	17.51329	17.47054	4335	301
160.357788	3.567886	17.72343	16.65830	16.23667	16.07098	16.02797	2126	301
226.001700	38.619699	16.60500	15.66234	15.39406	15.29443	15.29302	3699	301

Tabular data



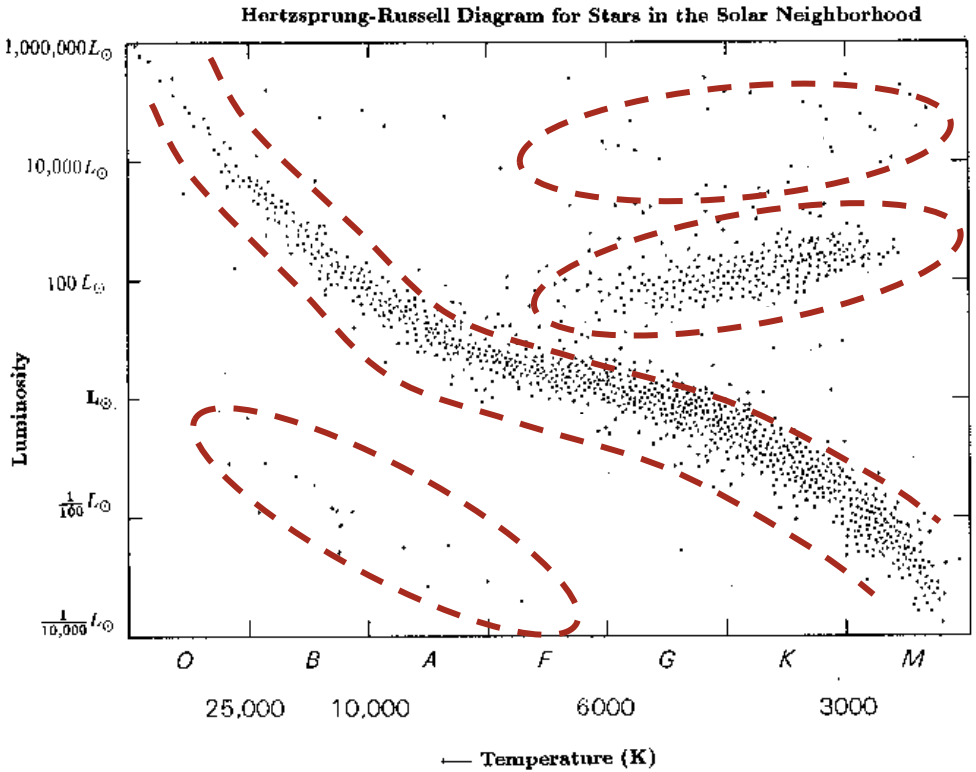
Graph data

- nitrogen atom without explicit hydrogens
- nitrogen atom with one explicit hydrogens
- sulfur atom
- carbon atom
- single bond
- double bond
- triple bond
- aromatic bond

Show of hands exercise, next!

Typical ML problems #1

Easy question: How many clusters are present in this pic?

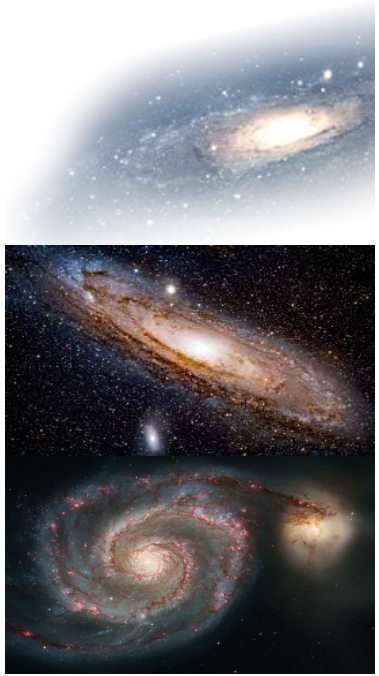


Clustering problem

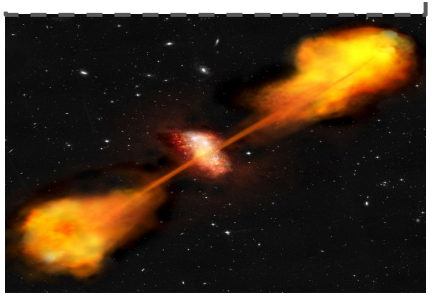
Typical ML problems #2

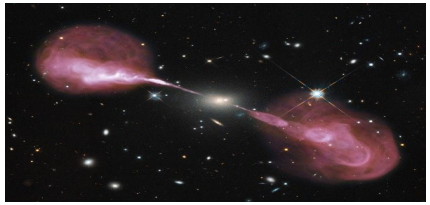
Classification problem

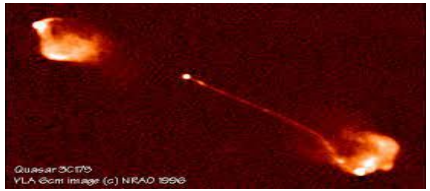
Galaxies (0)



Jets (1)



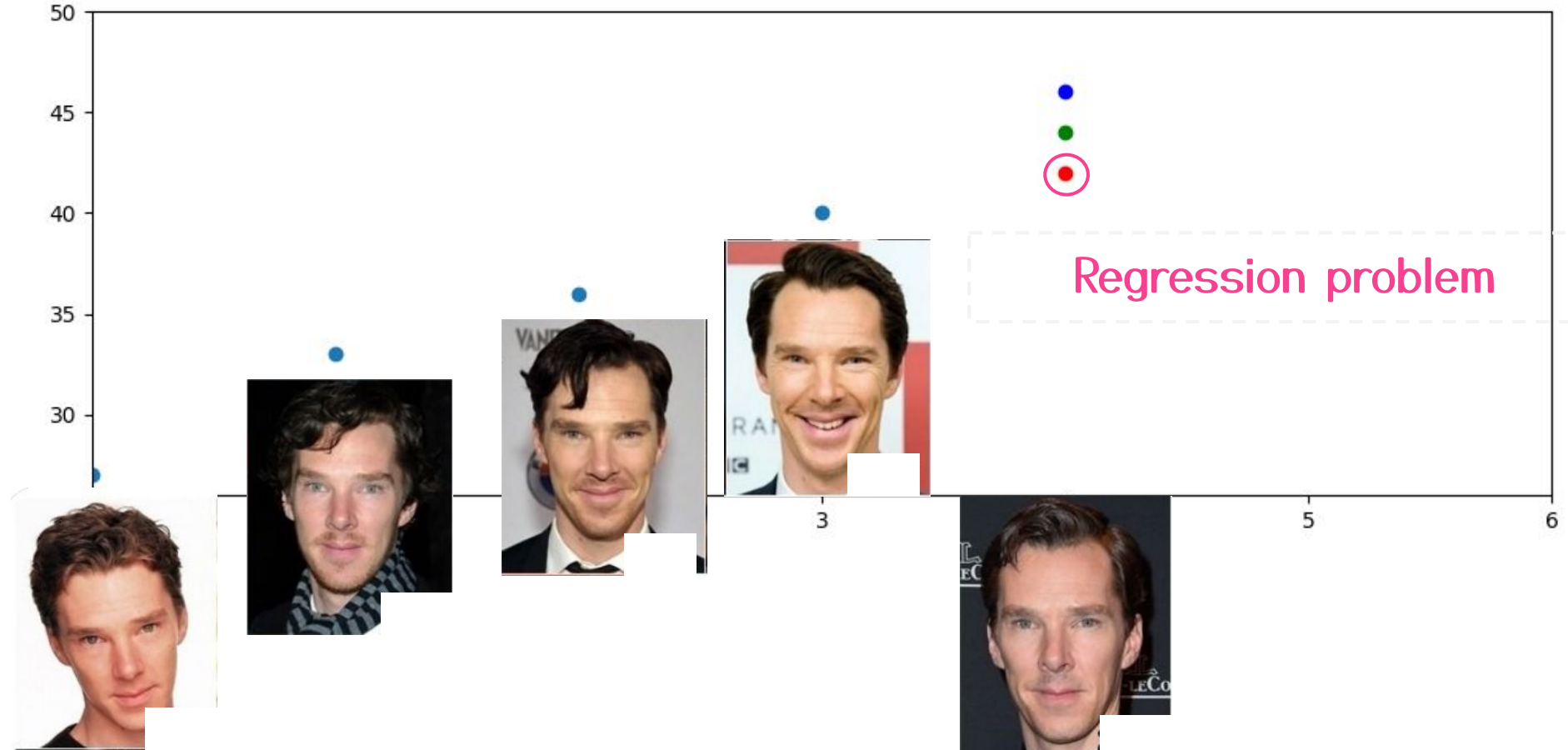




Galaxy or Jet (0 or 1)?

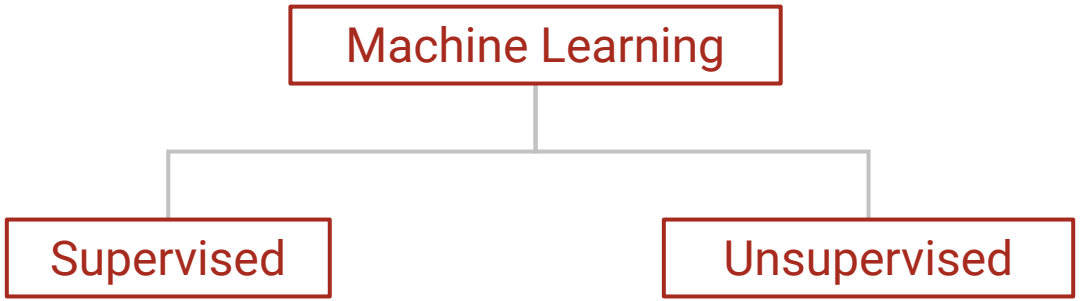
Typical ML problems #3

Which is the correct age of the HBO Sherlock in the pic given below?

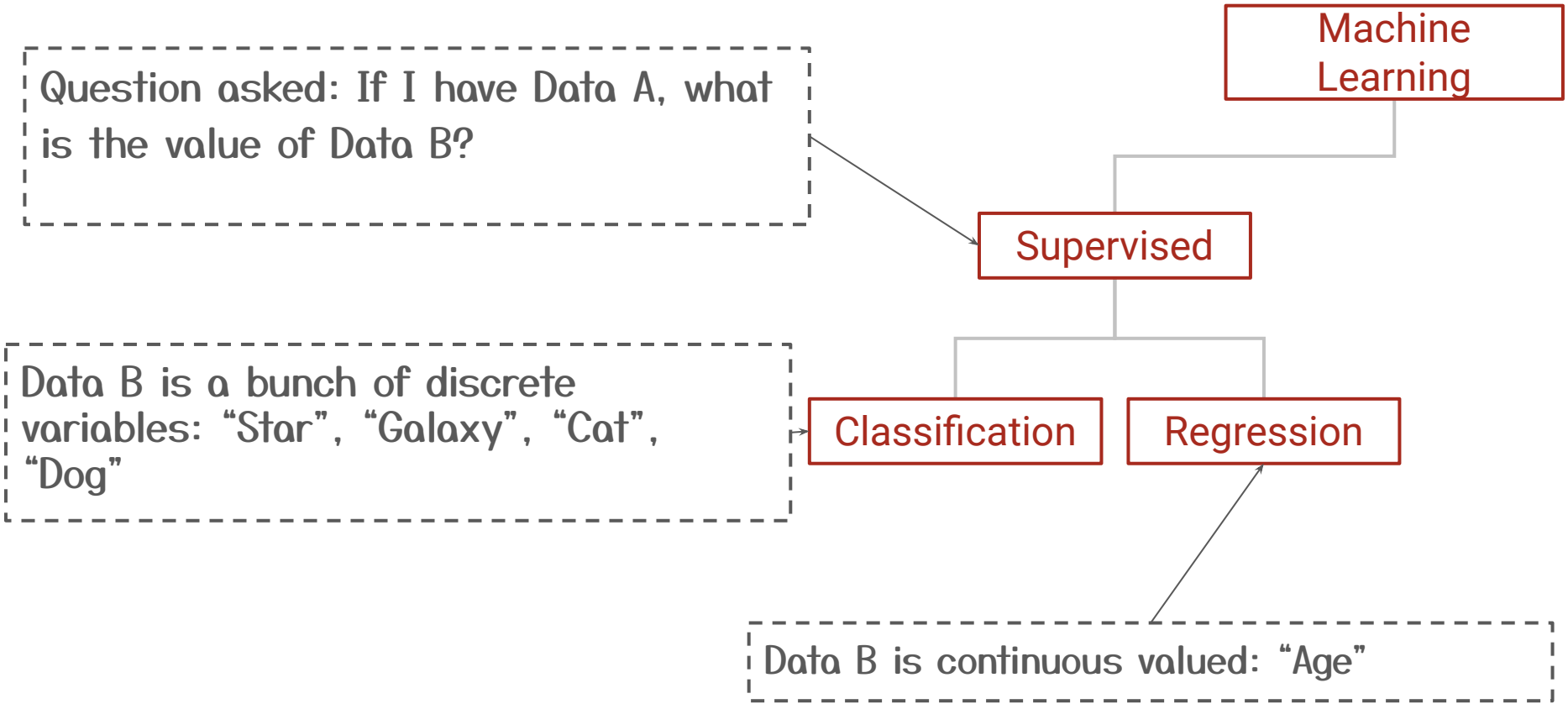


Regression problem

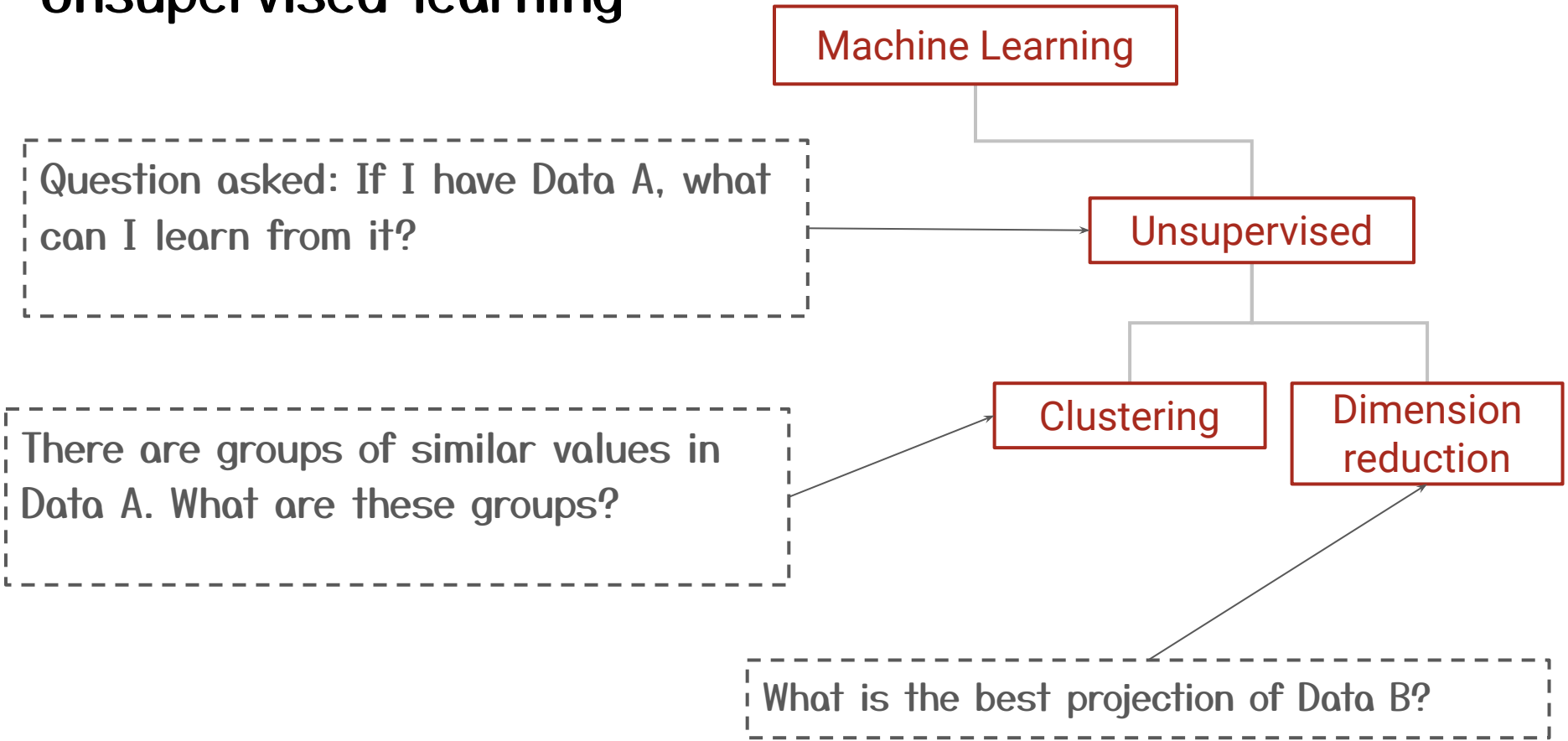
Typical ML problems



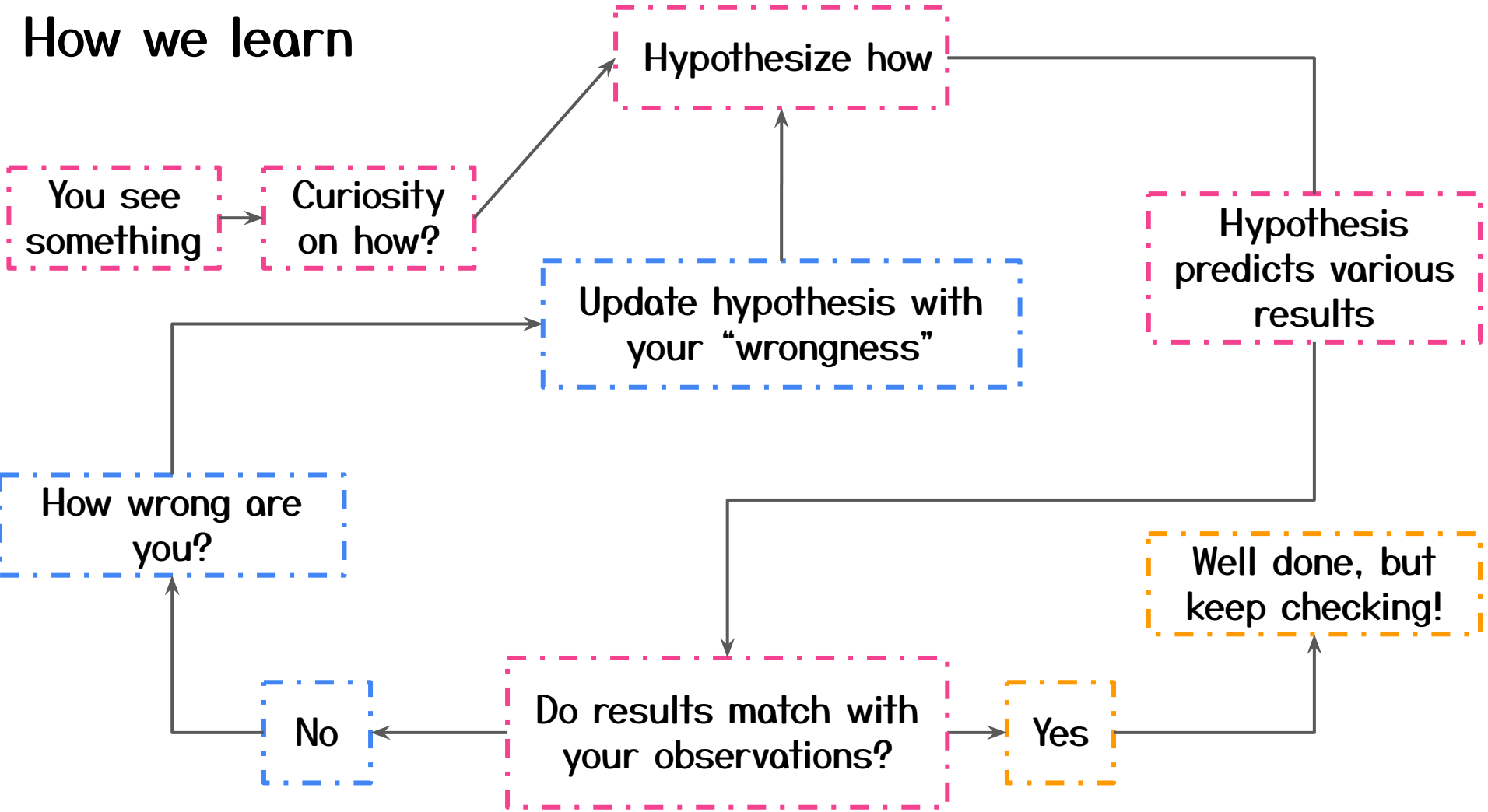
Supervised learning

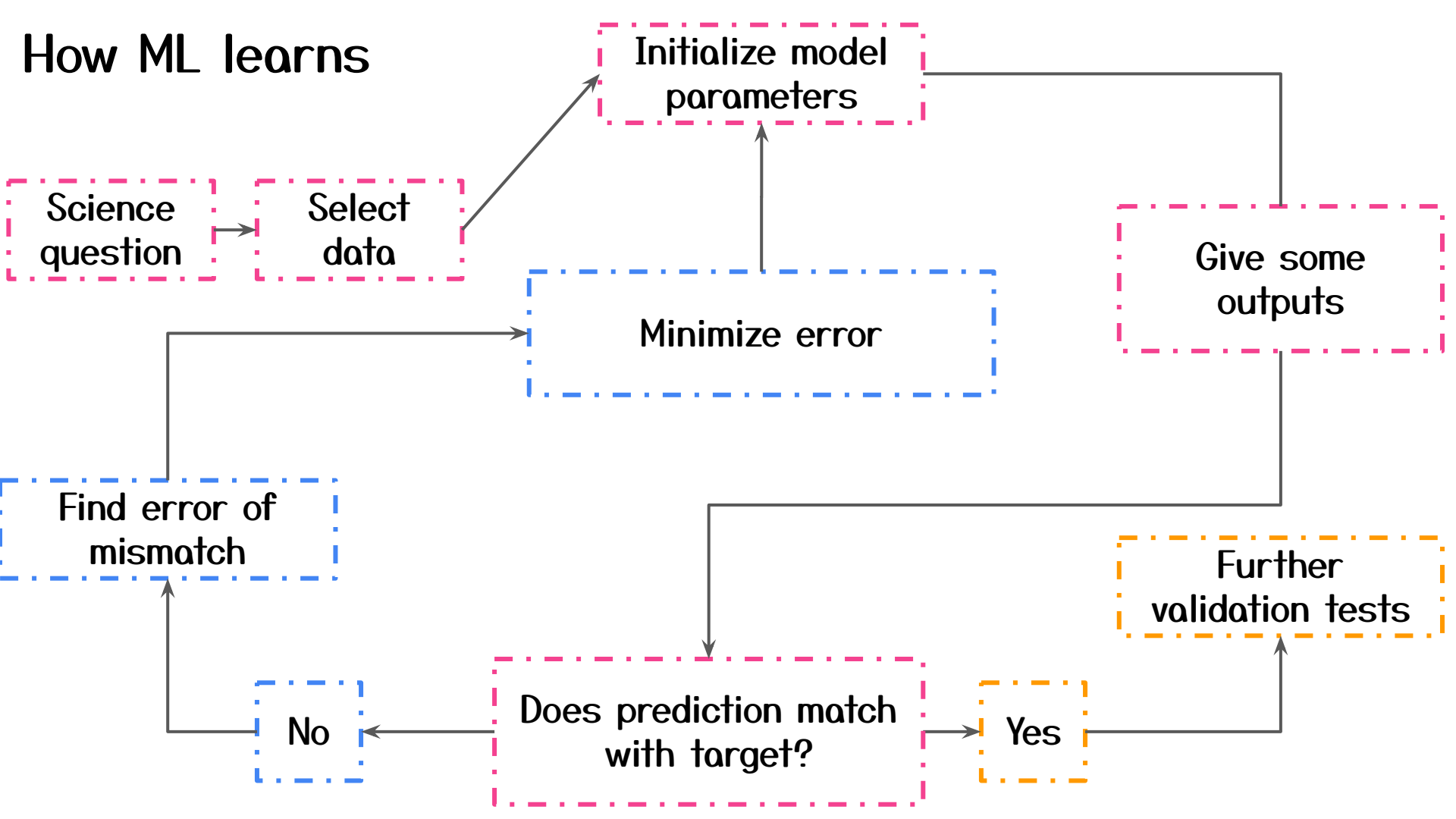


Unsupervised learning

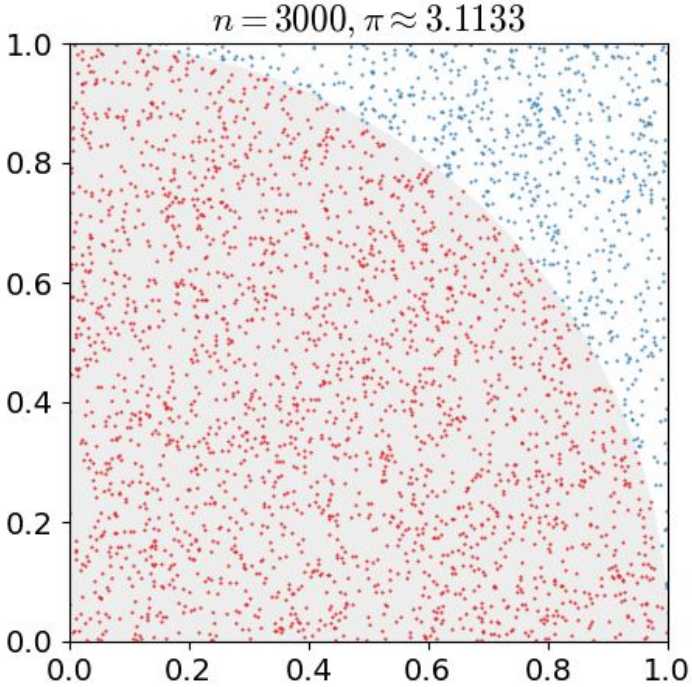


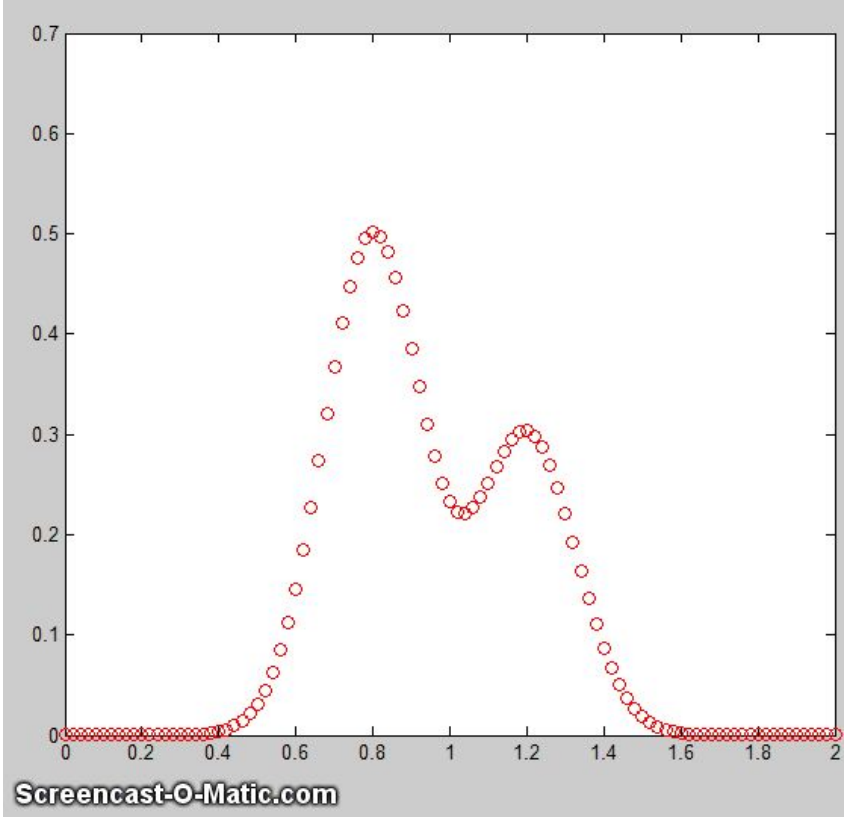
How we learn


Hypothesize how
You see something
Curiosity on how?
Hypothesis predicts various results
Update hypothesis with your "wrongness"
How wrong are you?
Well done, but keep checking!
No
Do results match with your observations?
Yes


How ML learns
Initialize model parameters
Science question
Select data
Minimize error
Give some outputs
Find error of mismatch
Further validation tests
No
Does prediction match with target?
Yes

How to minimize error?





Randomly keep changing your model parameters till you strike gold
⇒ Monte Carlo methods

Iteratively select the best solution depending on the previous solution
⇒ Simplex/Gradient based methods

Gradient-based methods

Select the model parameters (let's say W) for which error is minimum

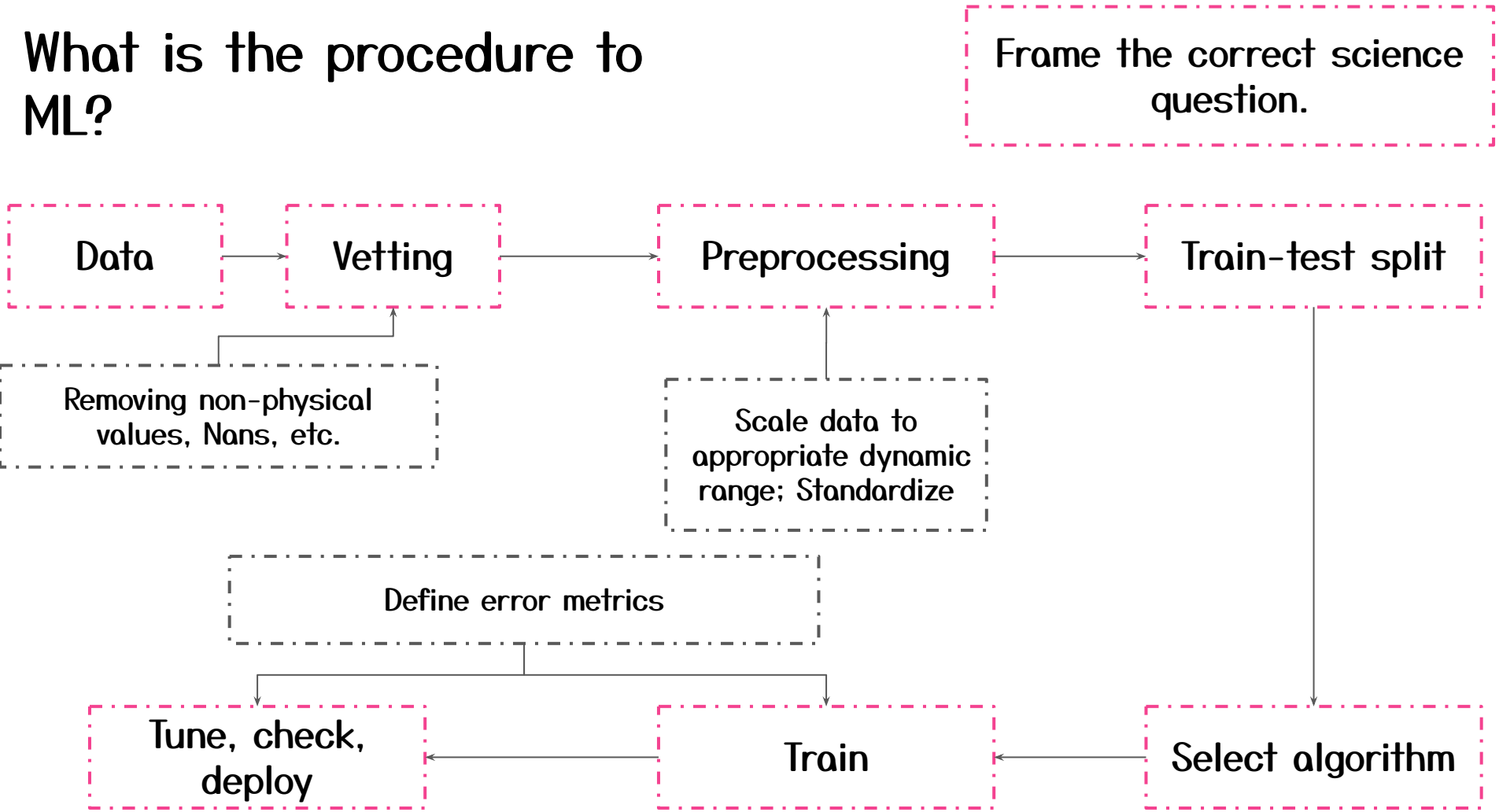
For the best model, we have:

$$\frac{\partial Error}{\partial W} = 0$$

If it is not, then change W!

$$W \leftarrow W - \frac{\partial Error}{\partial W}$$

What is the procedure to ML?

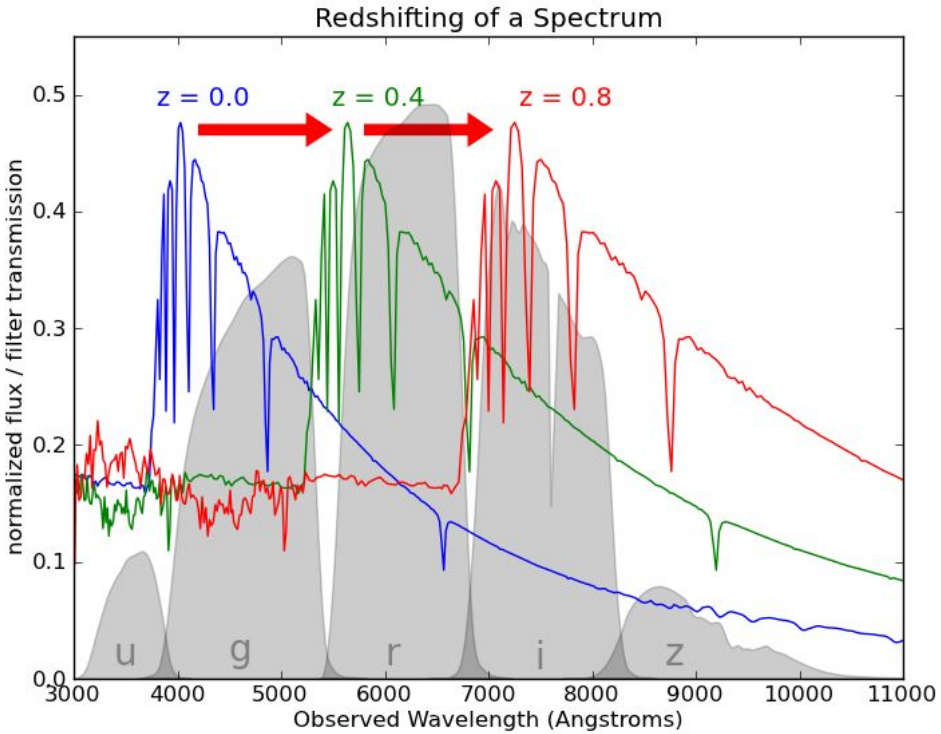


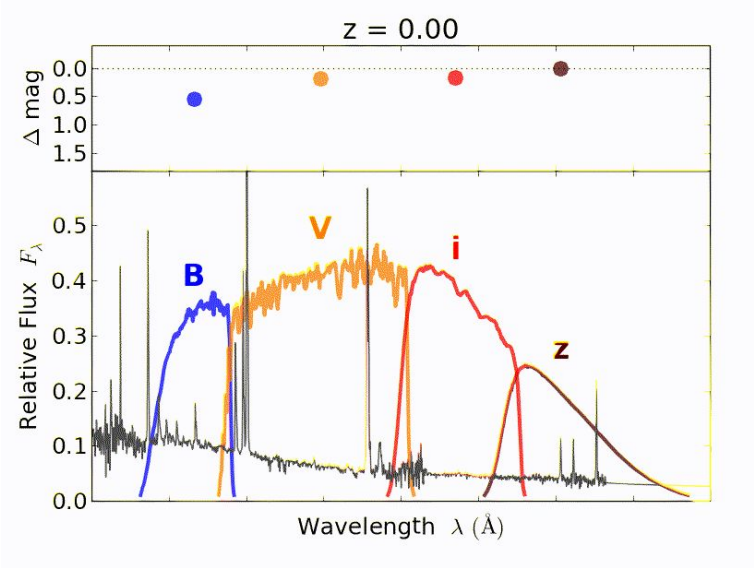
Task for the day

Estimate spectroscopic redshift from photometric colors ⇒

We will use a simple Linear Regression and a Deep neural network.

Why should it work?





From https://www.kaggle.com/c/photometric-redshift-estimation-2019

Metrics

$$\frac{1}{N}\Sigma(z_{pred} - z_{known})^2$$

Mean square error

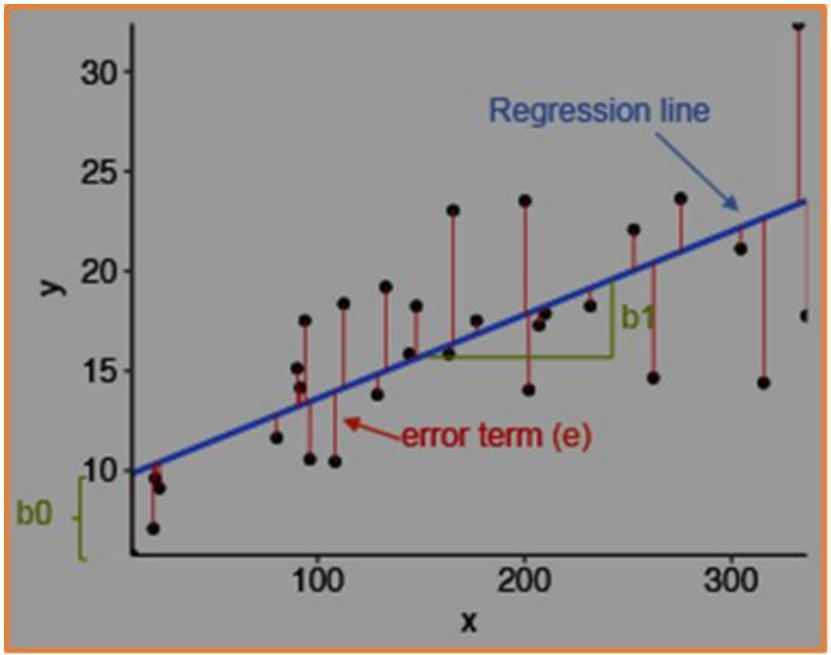
$$\frac{1}{N}\sum_{i=0}^{N}\frac{|z_{known,i} - z_{pred,i}|}{max(\epsilon, |z_{known,i}|)}$$

Mean absolute % error

$$1 - \frac{\Sigma(z_{pred} - z_{known})^2}{\Sigma(z_{known} - \mu(z_{known}))^2}$$

Coefficient of determination

Linear regression



Known z

Set of colors

Estimated (or predicted) y value

Estimate of the regression intercept

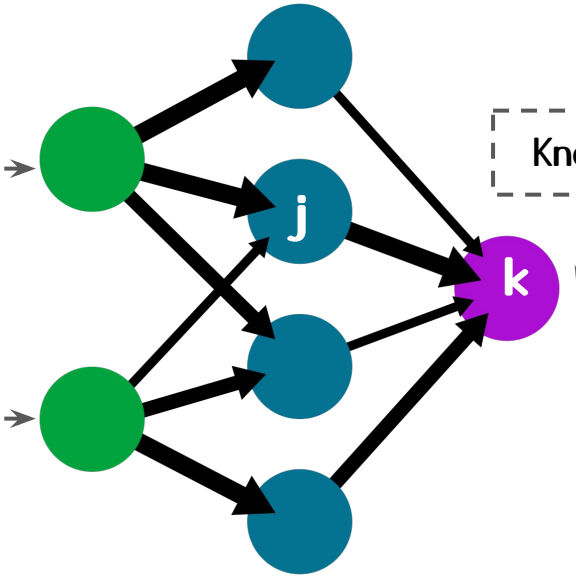
Estimate of the regression slope

Independent variable

$$y_i = b_0 + b_1 x + e$$

Error term

Neural network



Weight variable of neuron j

Bias variable of neuron j

Output from neuron j

Known z

$$y_j = f(\Sigma_i w_{ij} x_i + b_j)$$

Input to neuron j from all neurons i

Non linear activation function

$$y_k = f(\Sigma_j w_{jk} y_j + b_k)$$

Neural network: Training

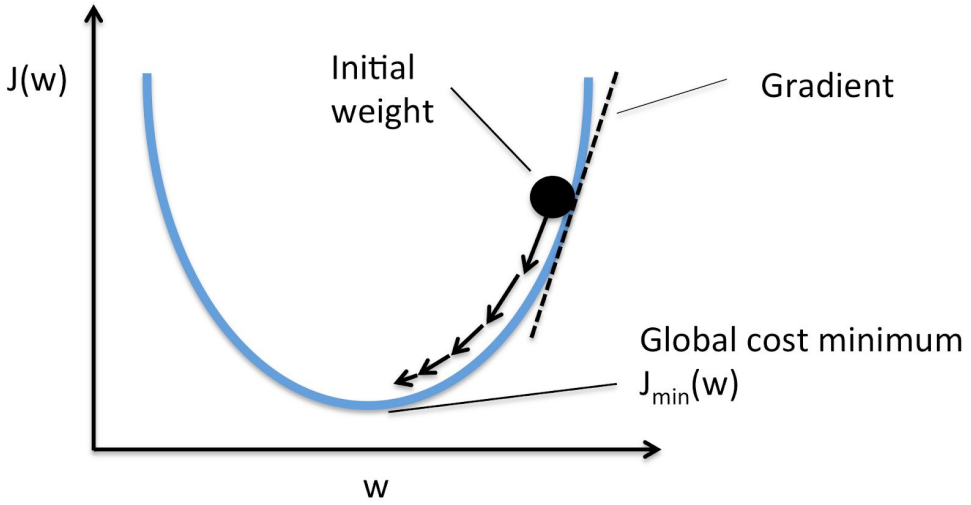
Step 1: Initialize w and b for all layers with some non-zero values.
Step 2: Calculate output from NN for input set:

$$z_{pred} = f(..f(\Sigma_k f(\Sigma_j w_{jk} x_j + b_k) + b_k)...$$

Step 3: z_{pred} will not match z_{known}. So get the error between these two values.
Step 4: Now you update weights and biases using this error:

$$w \rightarrow w - \alpha \frac{\partial loss}{\partial w}$$

Step 5: Repeat till convergence!



Let us move on to Jupyter →

References for further reading

1. Andrew Ng's course on Machine learning in coursera: https://www.coursera.org/learn/machine-learning
2. Fast AI deep learning course: https://www.fast.ai/
3. Analytics vidhya and Towards Data Science are good blogs too: https://www.analyticsvidhya.com/blog/2015/06/machine-learning-basics/, https://towardsdatascience.com/machine-learning-basics-part-1-a36d38c7916 .
4. Advanced: Bishop's book on Pattern recognition and Machine learning; Ian Goodfellow's book on Machine learning.