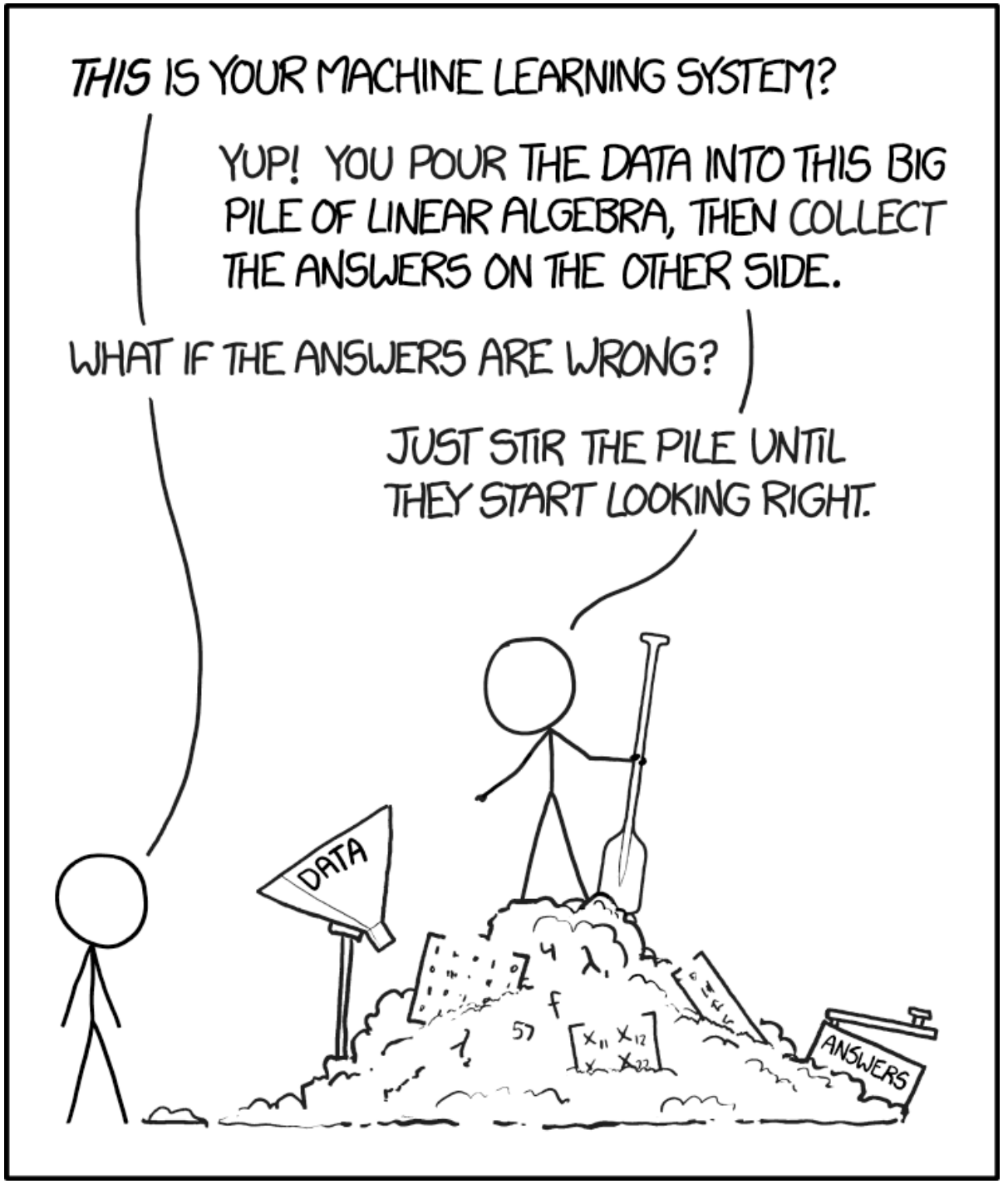
Machine Learning



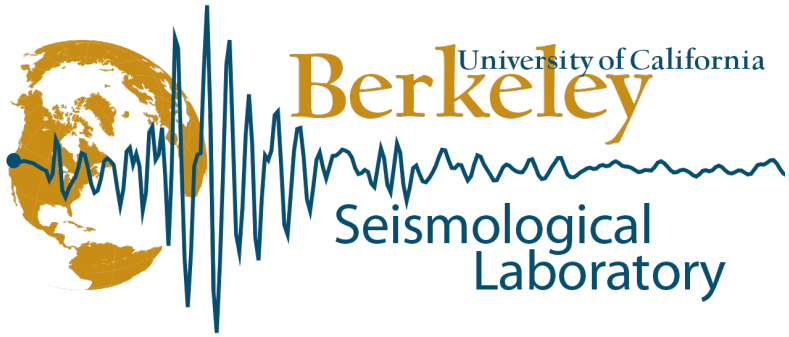
From [xkcd]

UC Berkeley Doctoral CDIPS Career Development Initiative for the Physical Sciences

Machine learning - using scikit-learn

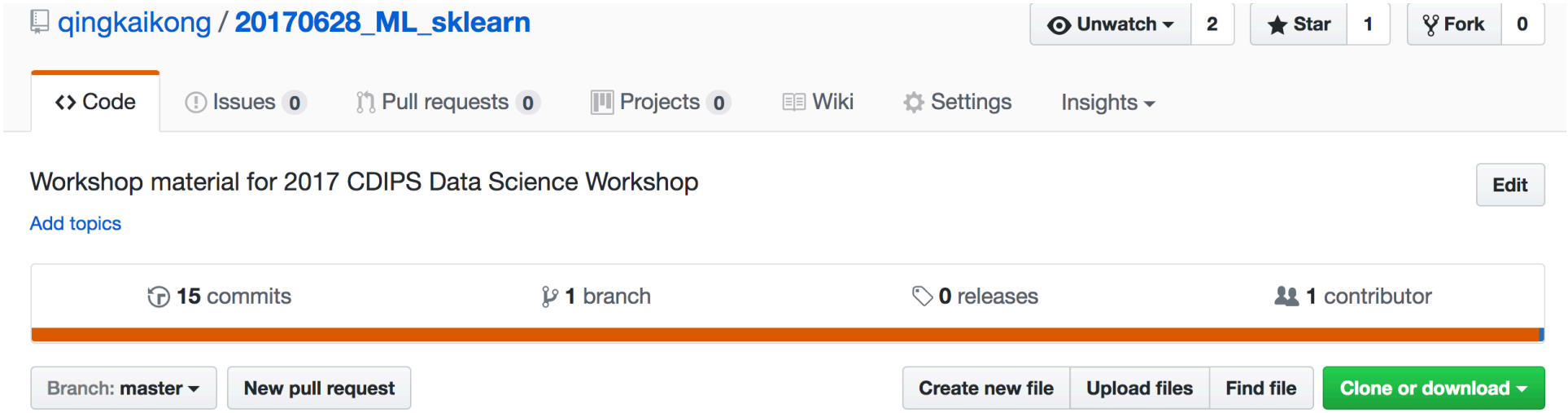
Qingkai Kong

2017-06-28

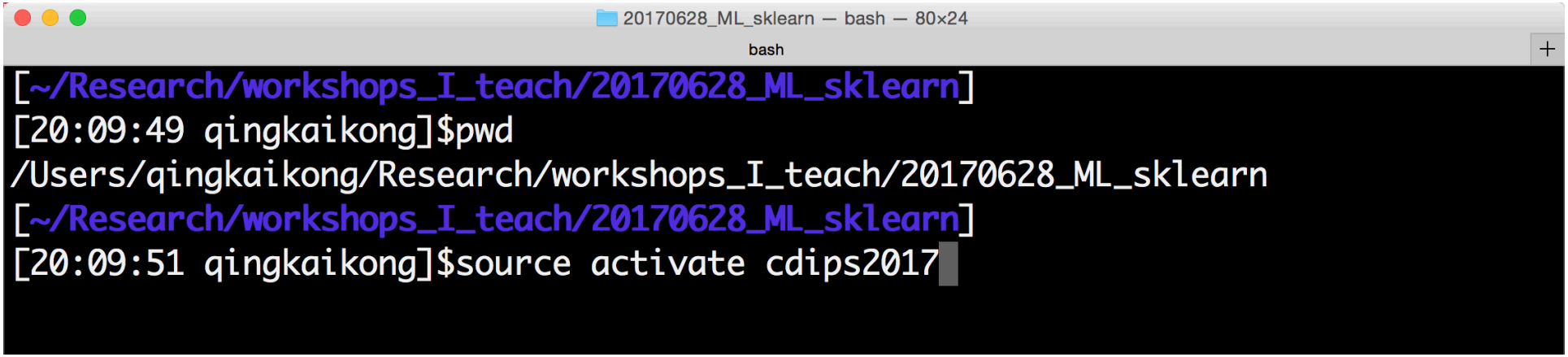


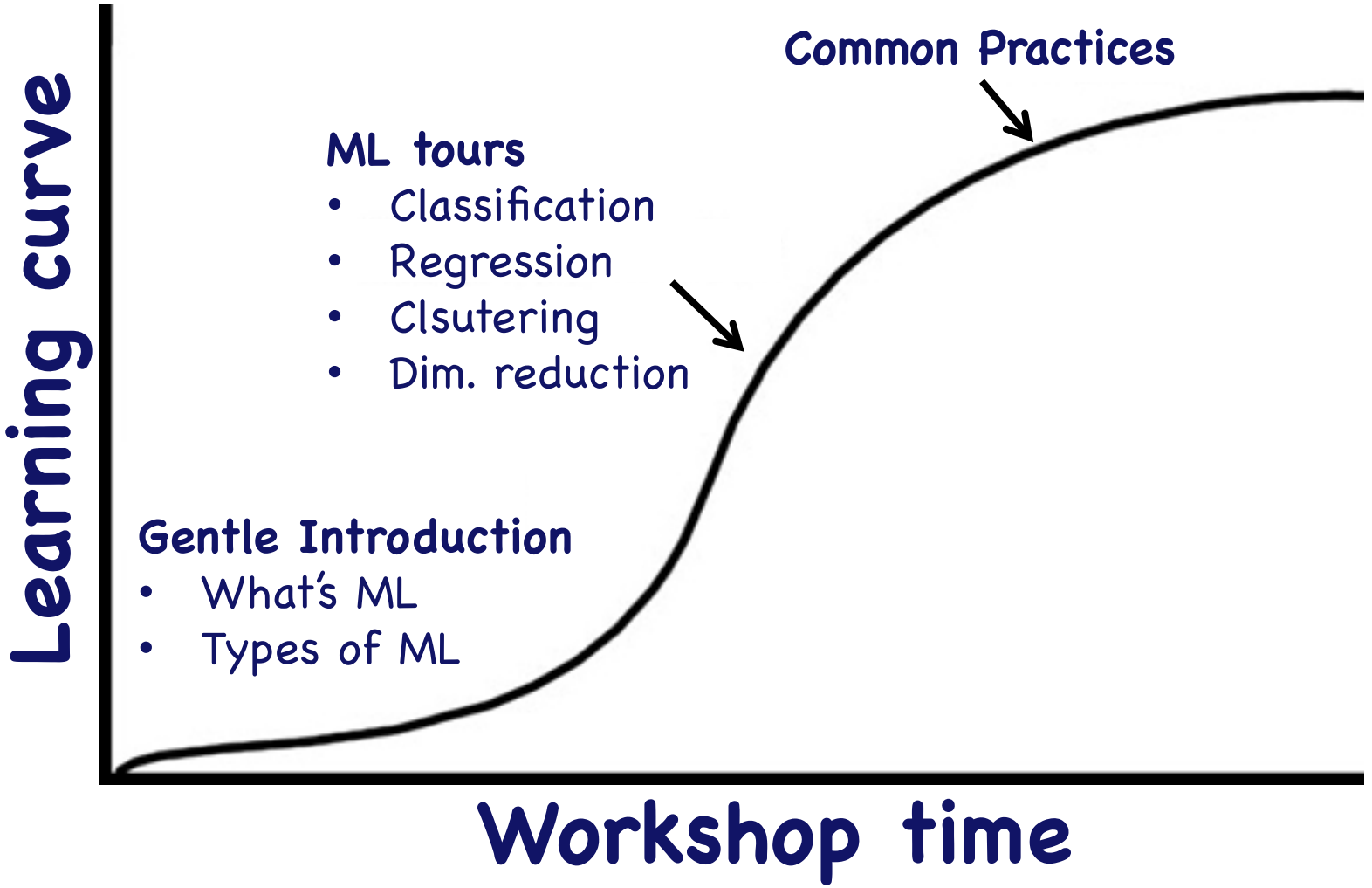
http://seismo.berkeley.edu/qingkaikong/

https://github.com/qingkaikong/20170628_ML_sklearn



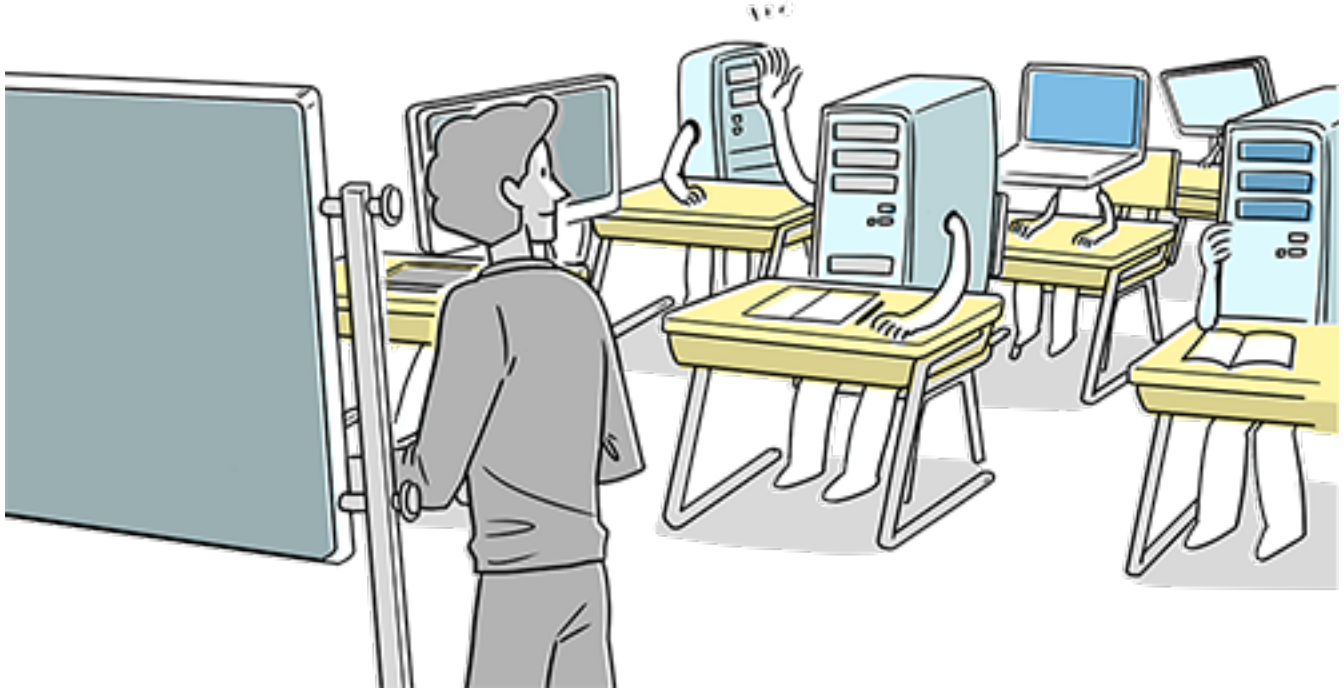
After downloads





https://github.com/qingkaikong/20170628_ML_sklearn

What is machine learning?



https://github.com/qingkaikong/20170628_ML_sklearn

Pipeline of training a machine learning model.

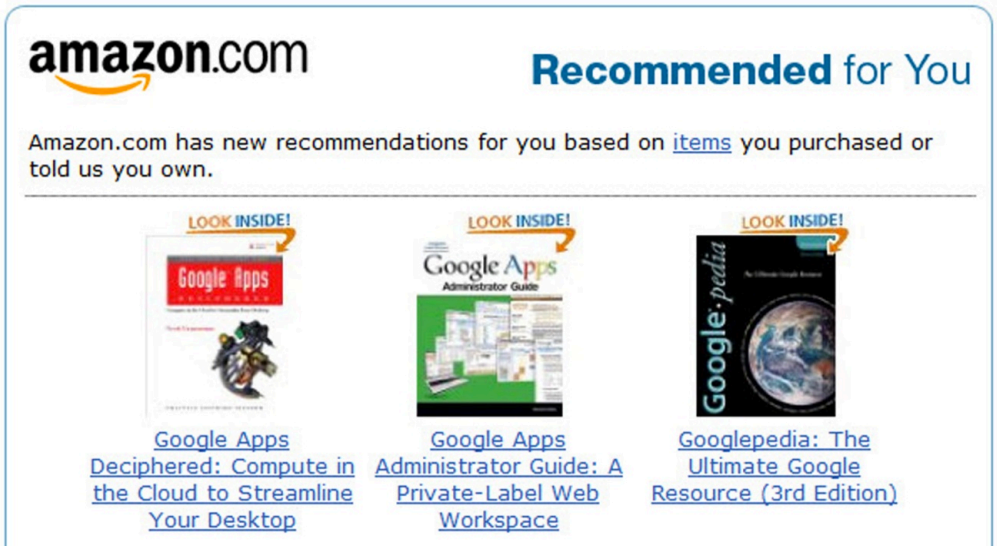
Data examples

01100
10110
11110

Tunable Model

Optimization algorithm

Trained Model



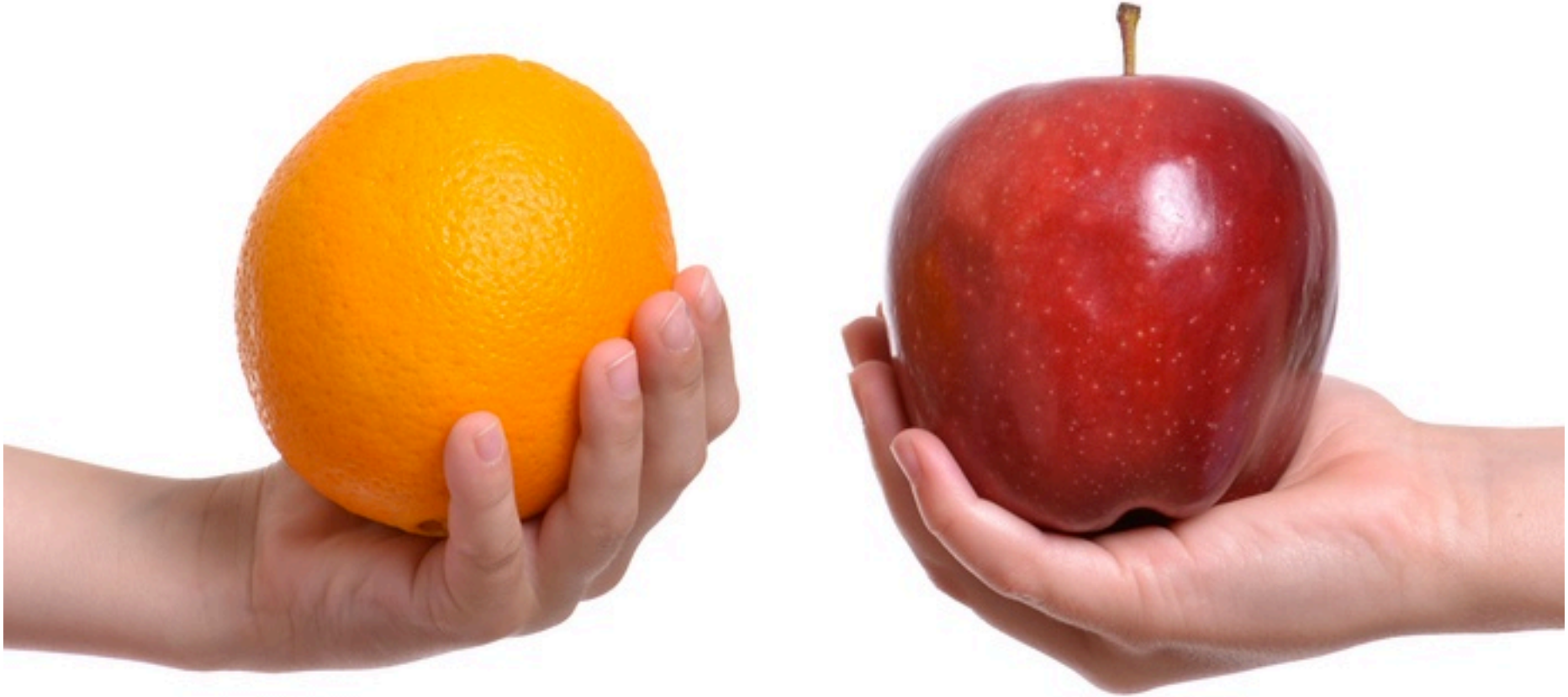
Self-driving car
Voice recognition
…

https://github.com/qingkaikong/20170628_ML_sklearn

Not always working

Common types of machine learning

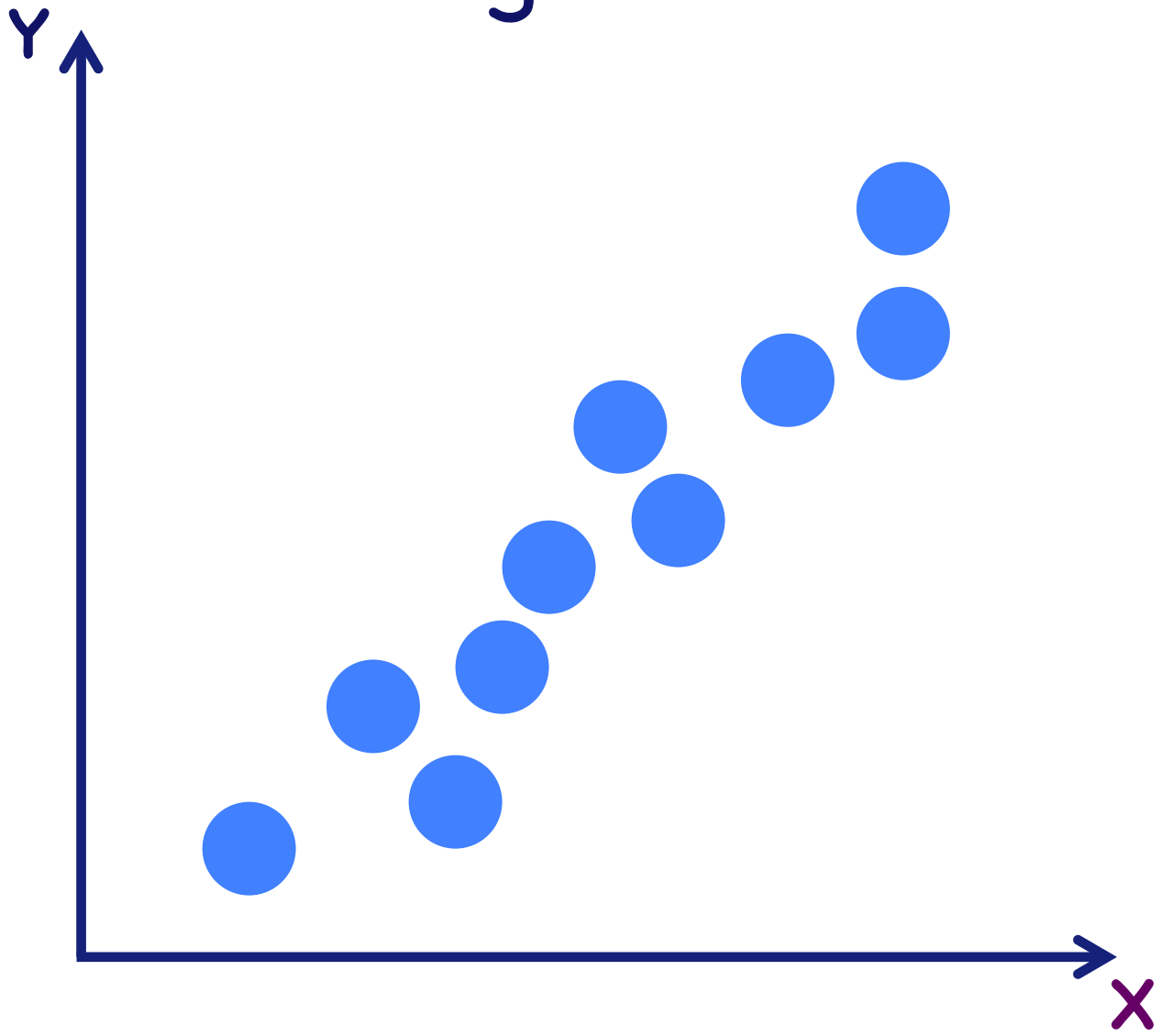
Supervised learning



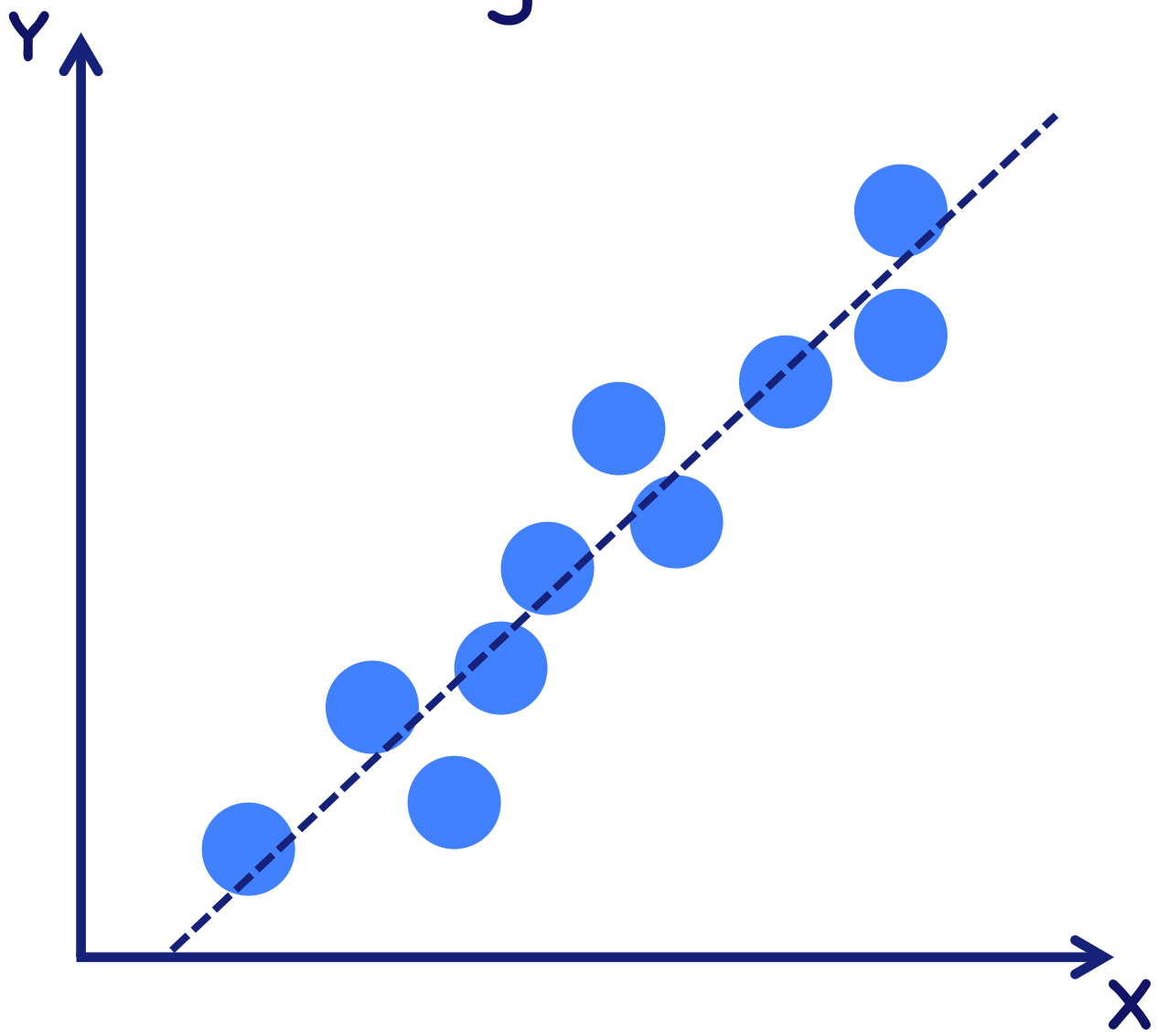
Unsupervised learning

Regression Classification

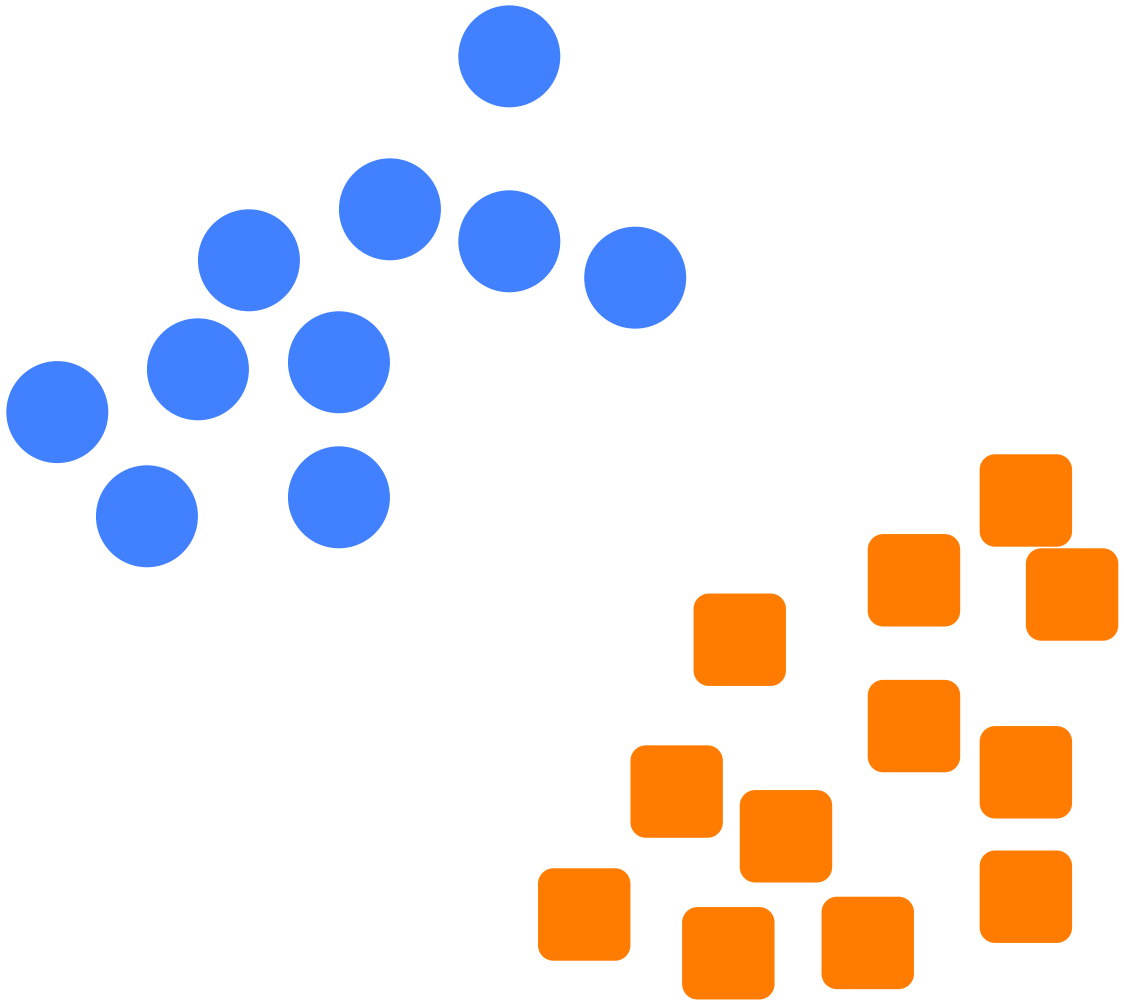
Regression



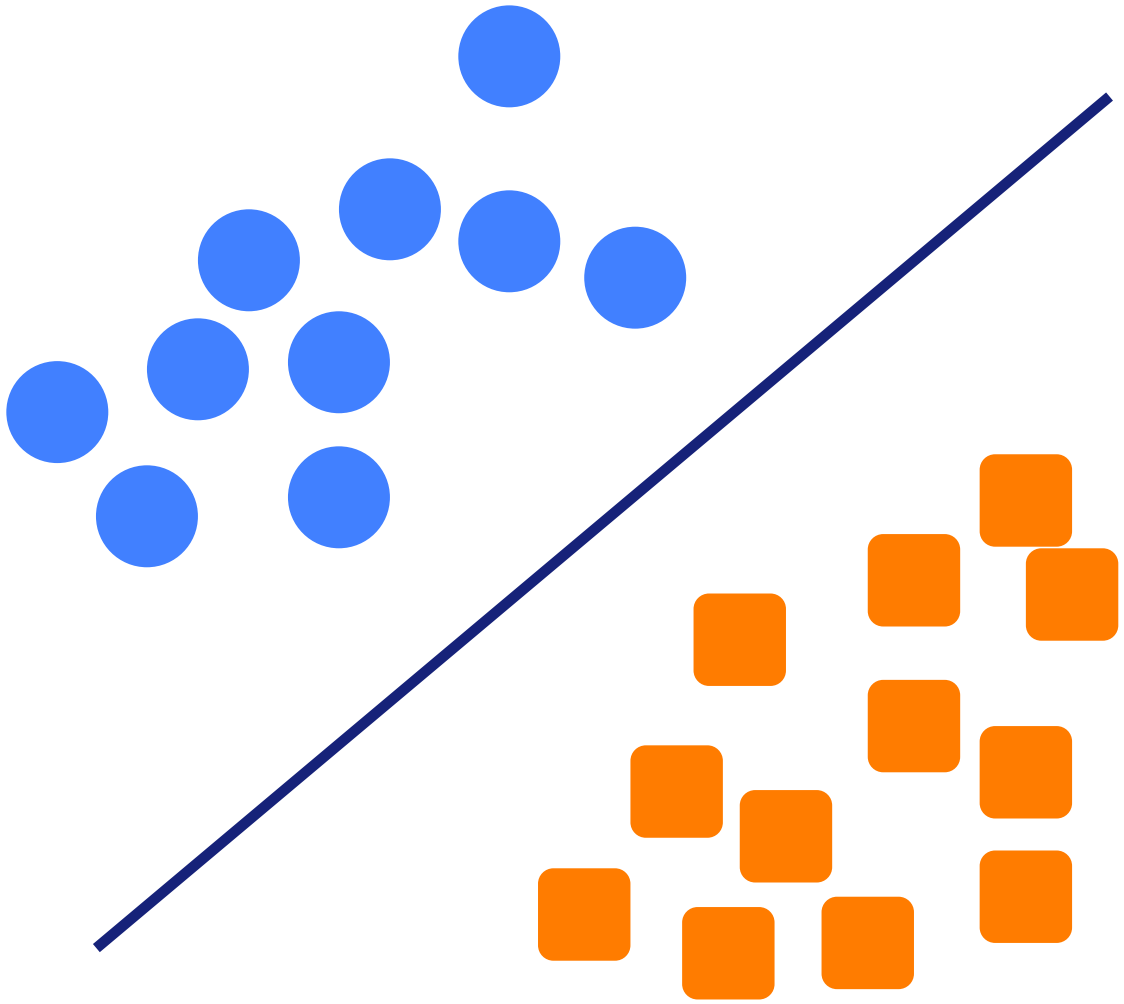
Regression



Classification



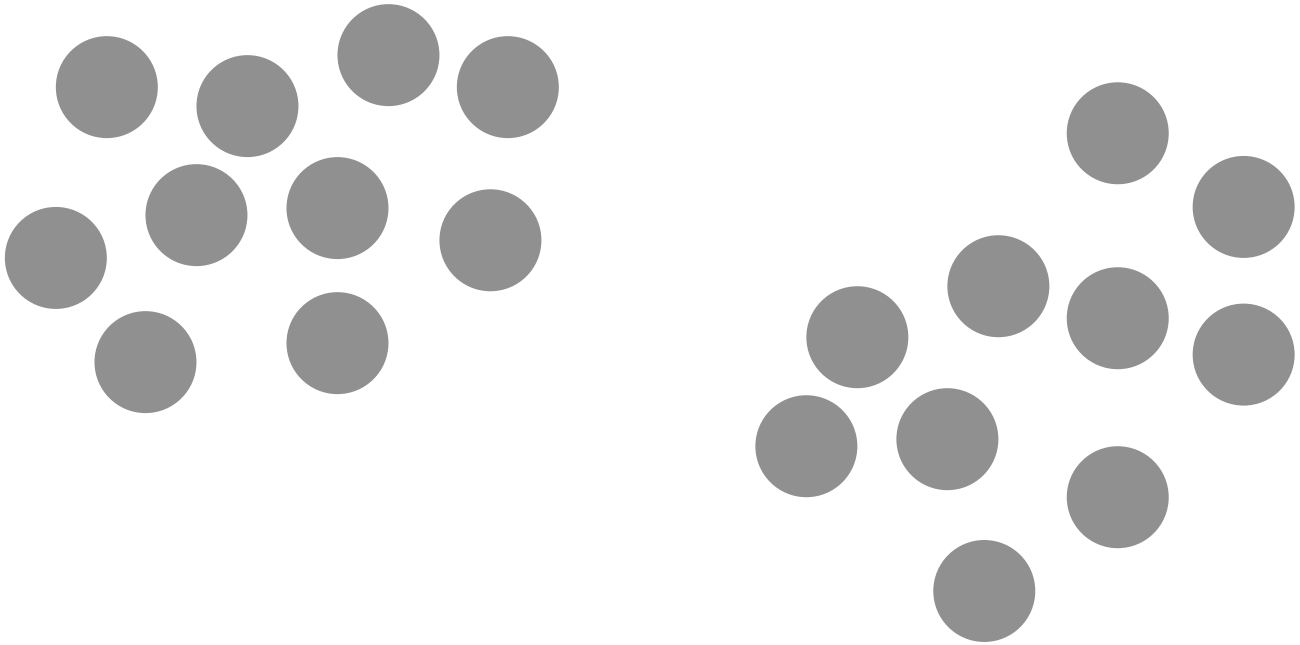
Classification



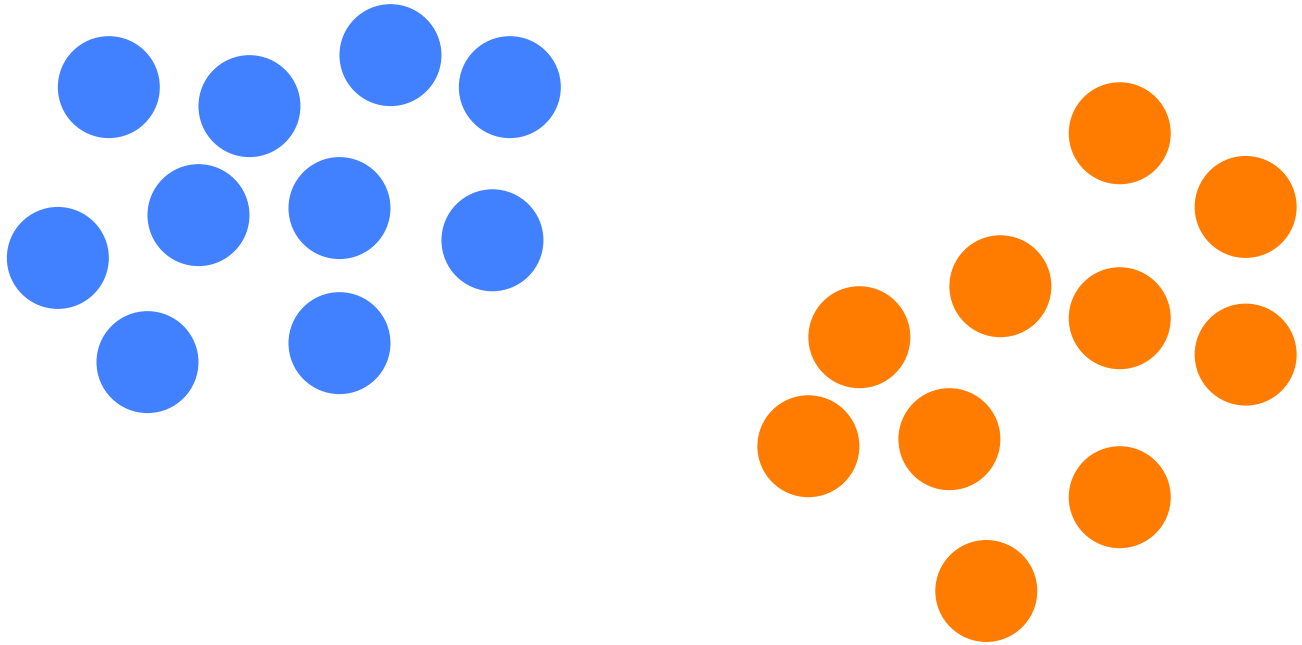
Clustering

Dimensionality
reduction

Clustering



Clustering



Dimensionality reduction

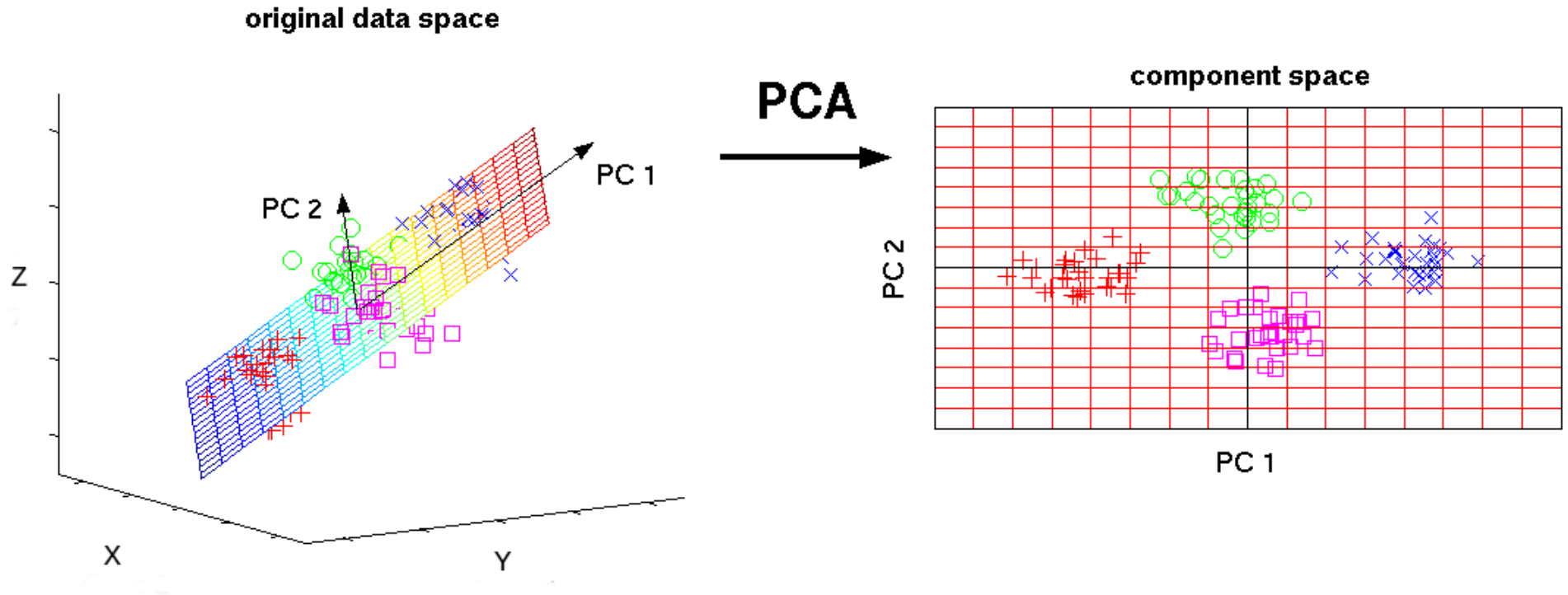


Figure from https://stats.stackexchange.com/questions/183236/what-is-the-relation-between-k-means-clustering-and-pca

Pipeline of training a machine learning model.

Data examples

01100
10110
11110

Tunable Model

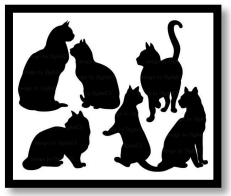
Optimization algorithm

Trained Model

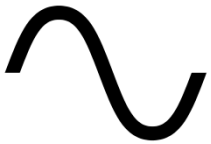
Representation of data

Raw data

Documents



Images



Numbers

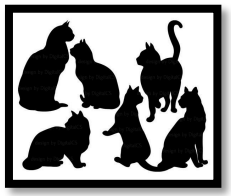
Sounds

●●●

Representation of data

Raw data

Documents



Images

Numbers

Sounds

●●●

Feature matrix (X)

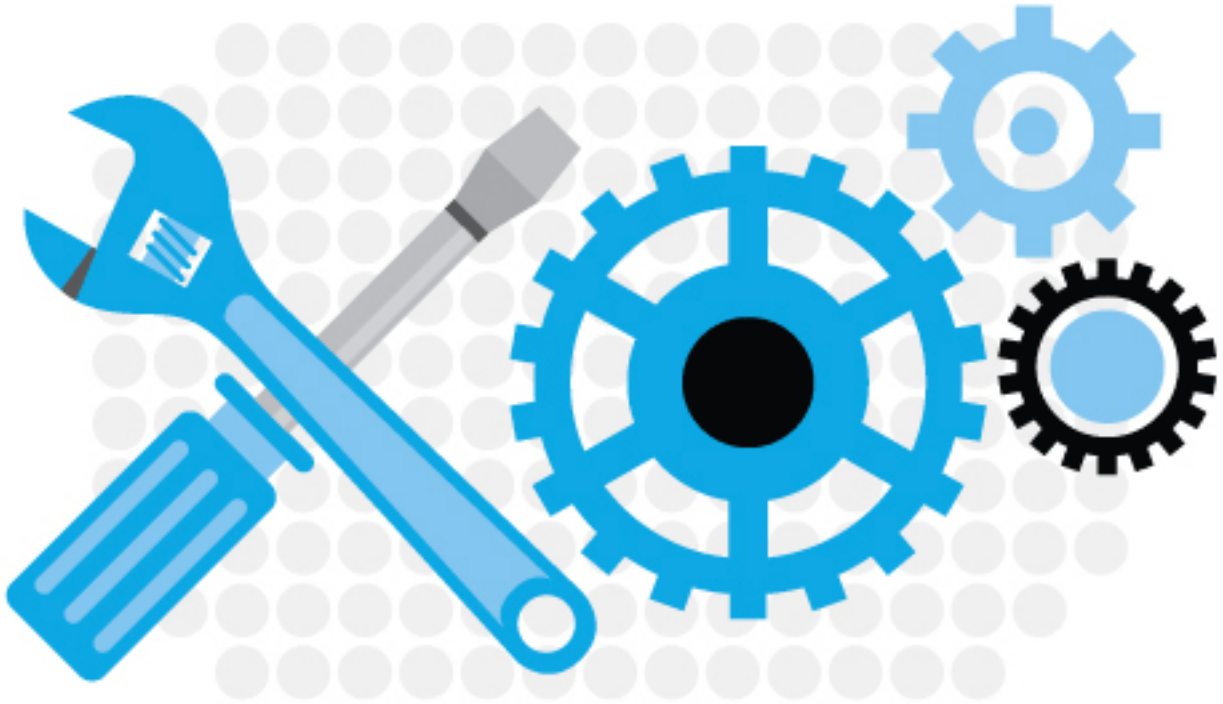
n_features ⟶

n_samples ⟶

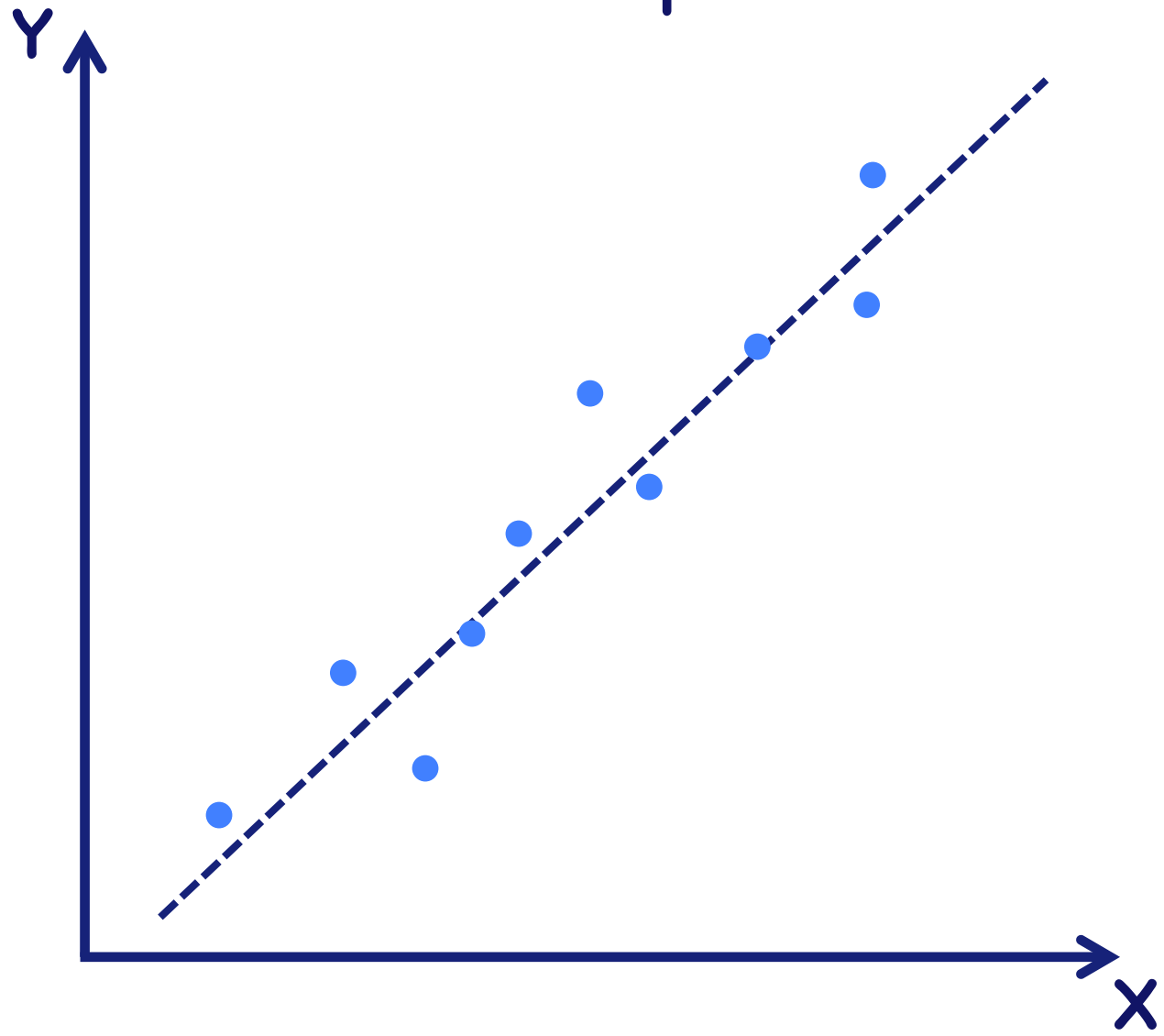
Target (y)

n_samples ⟶

Optimization

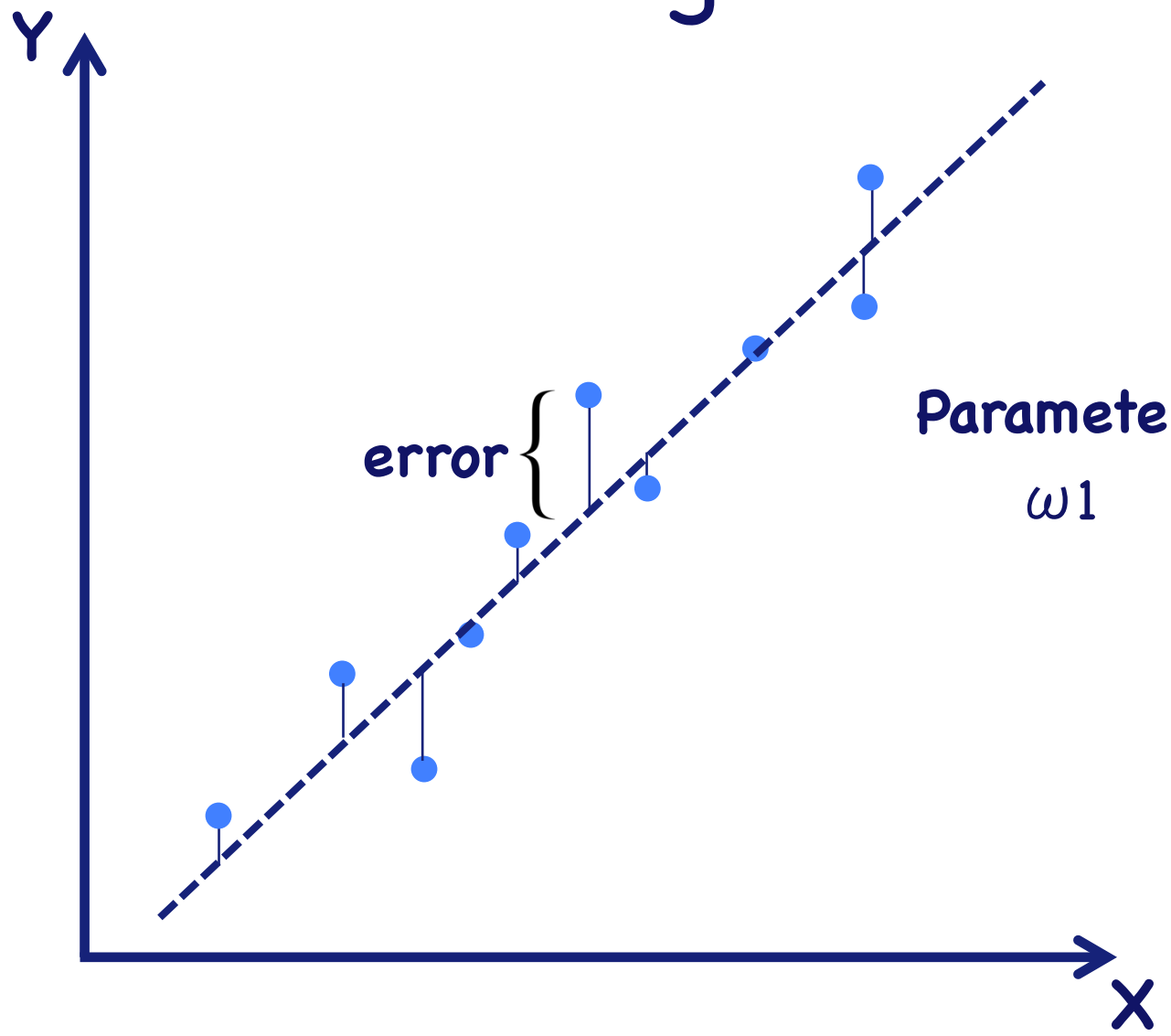


How we optimize?



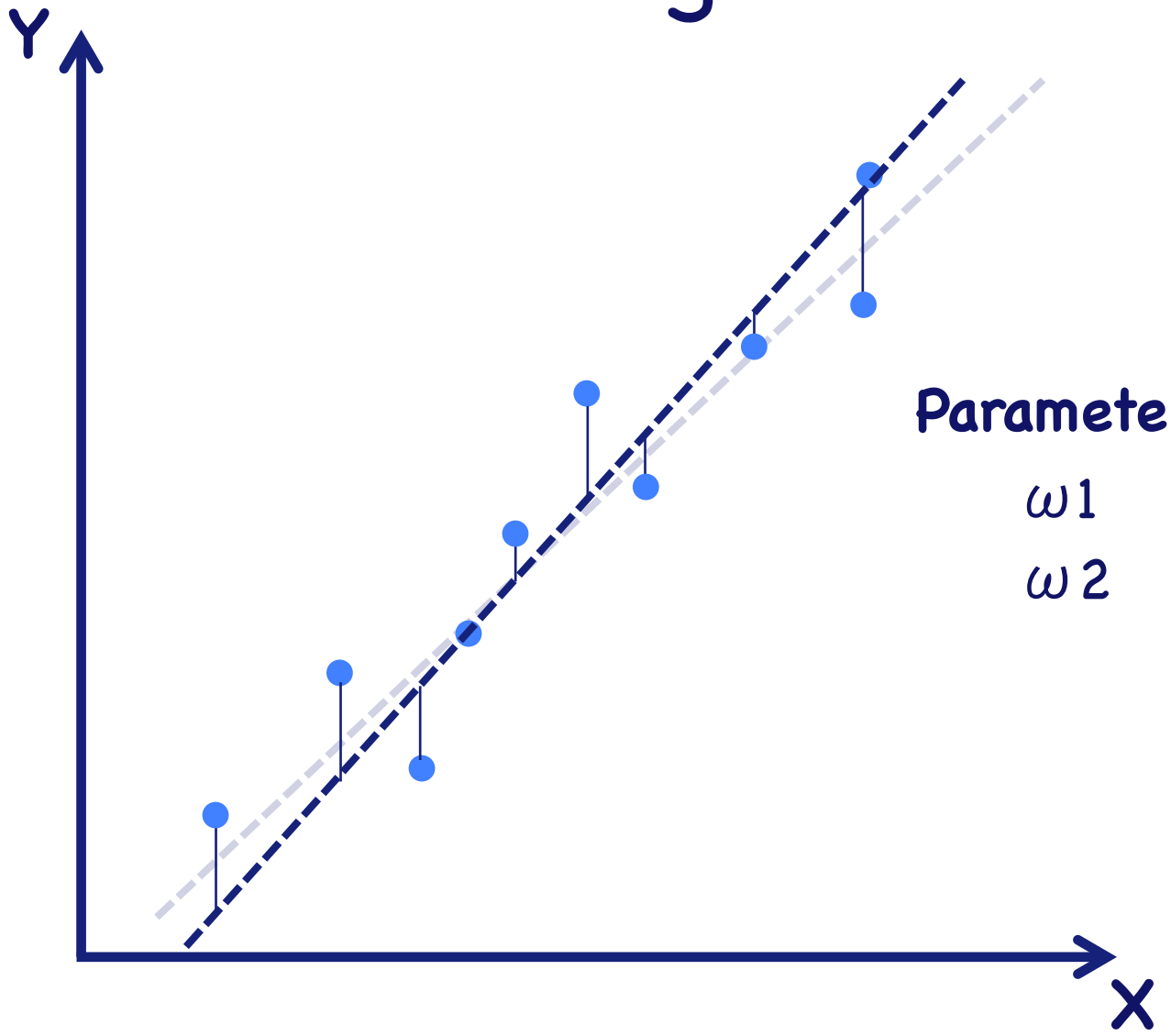
A simple example

Measuring error



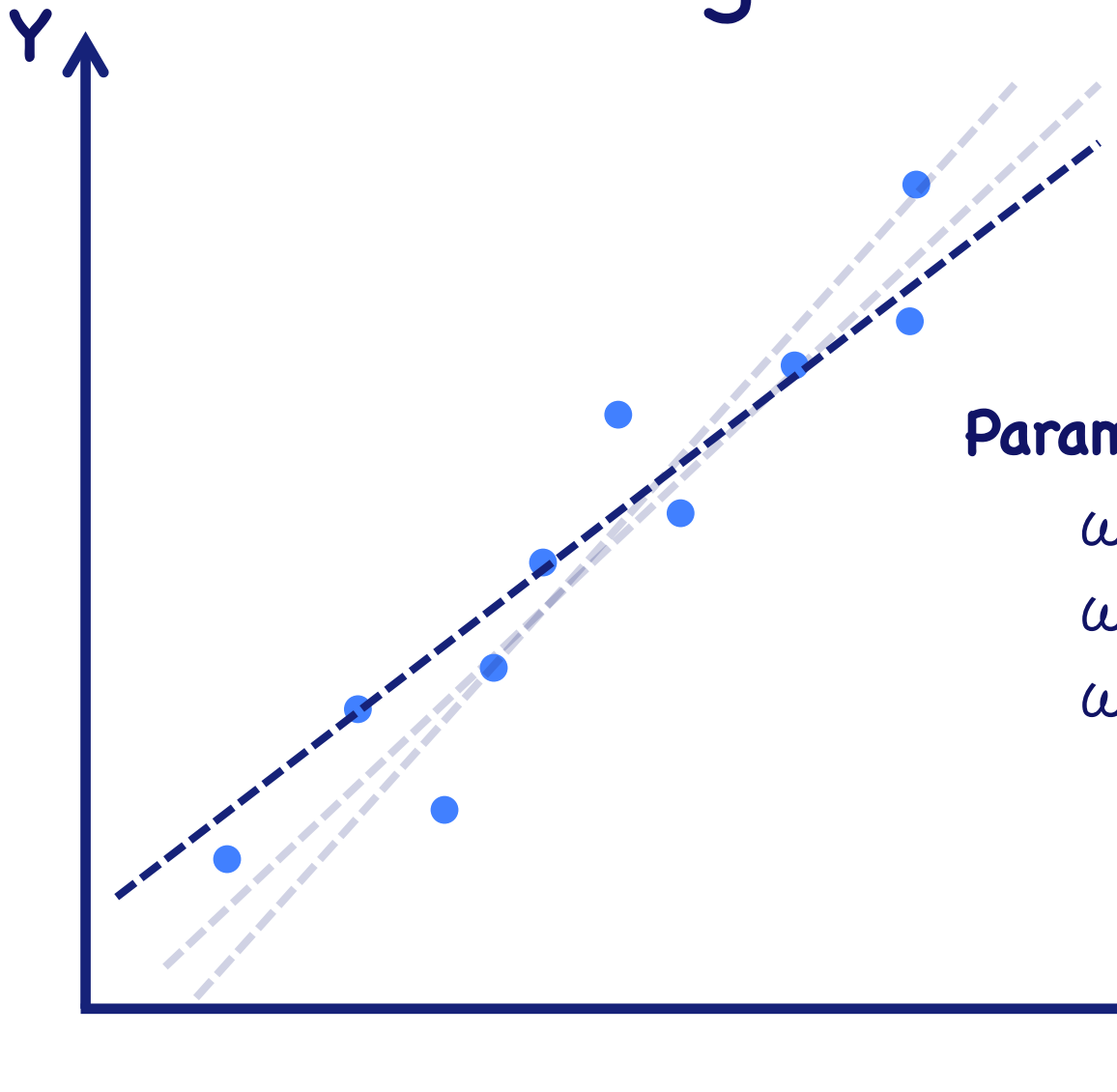
Parameter	Error
ω1	e1

Measuring error



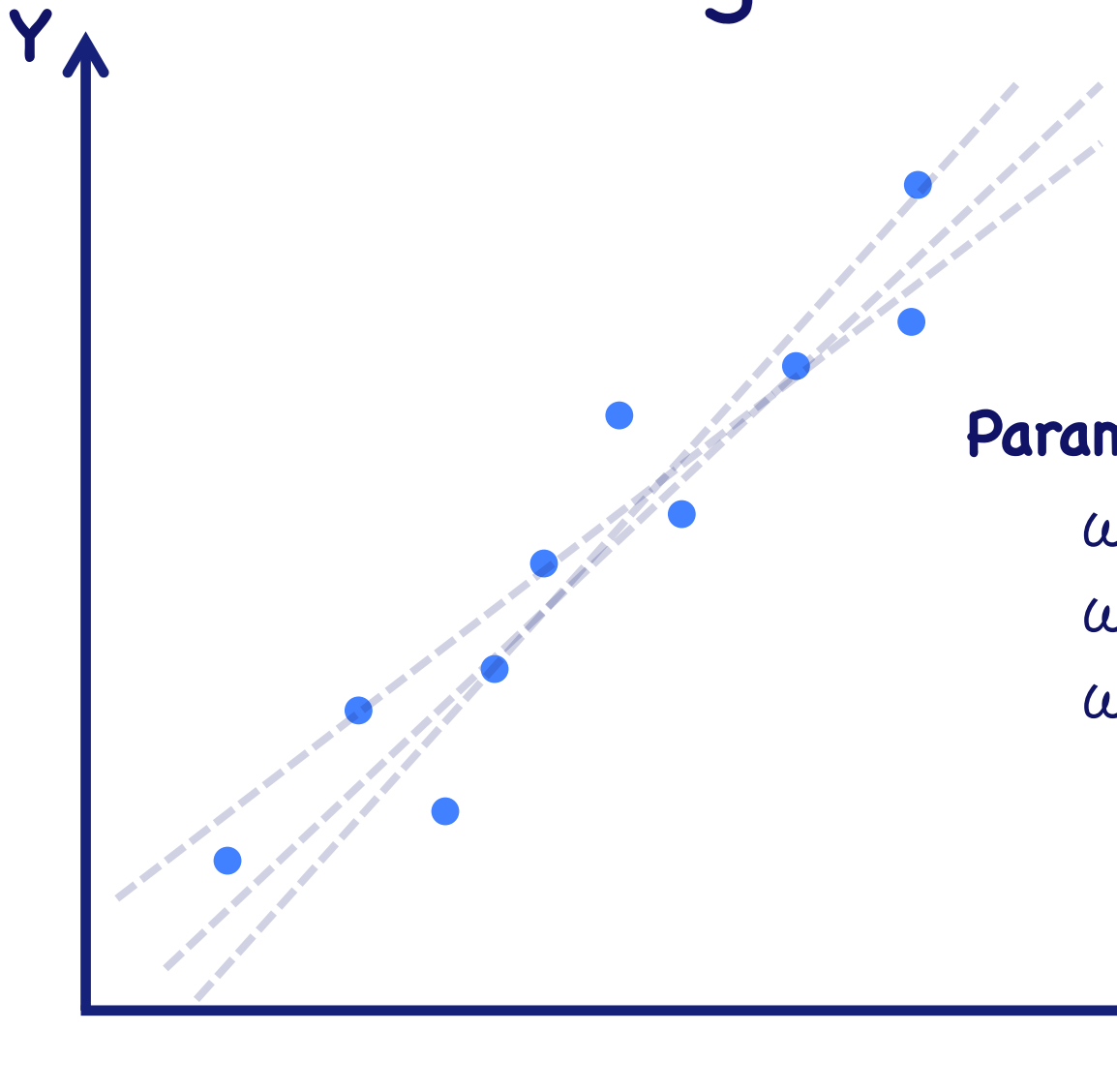
Parameter	Error
ω1	e1
ω2	e2

Measuring error



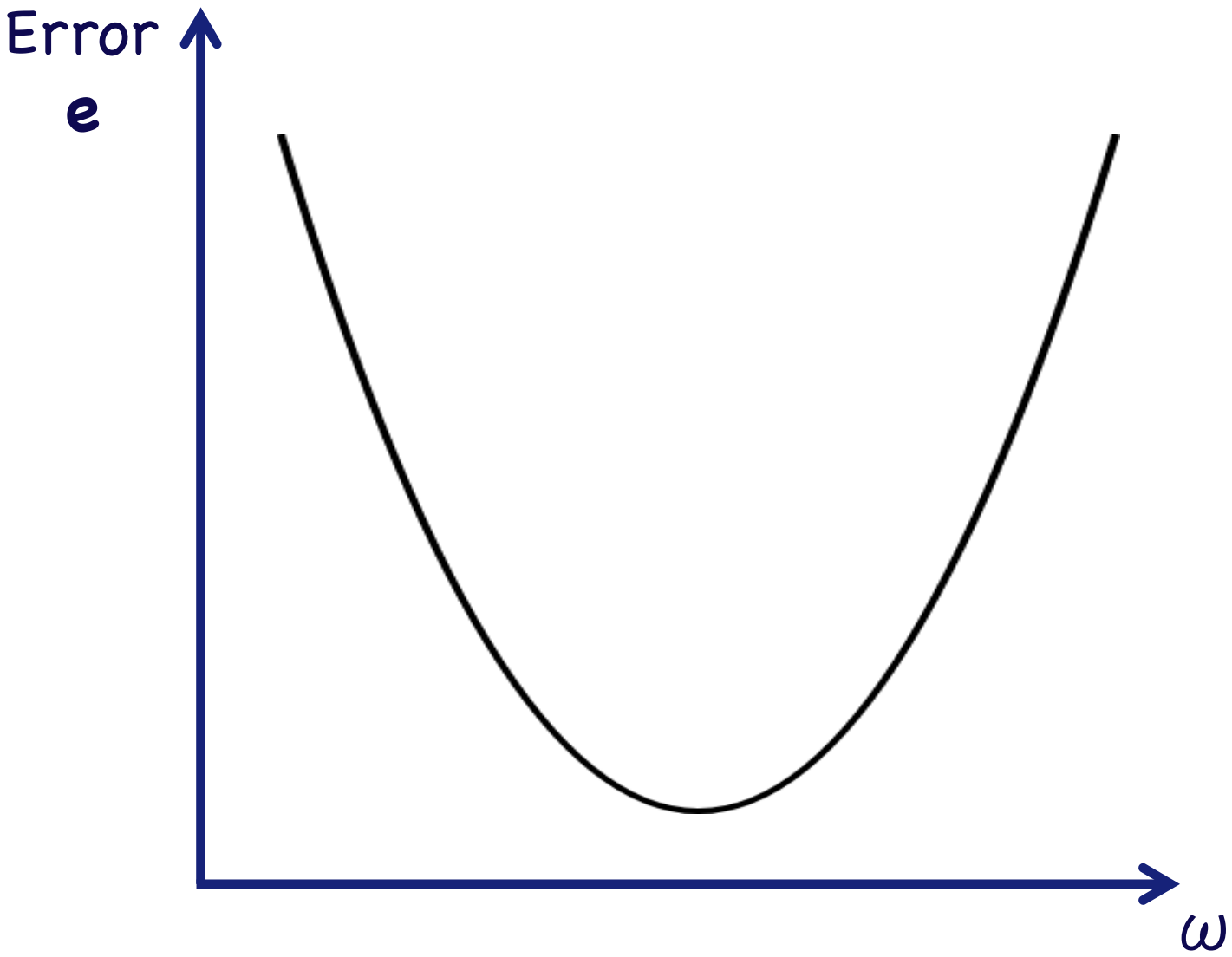
Parameter	Error
$\omega 1$	e1
$\omega 2$	e2
$\omega 3$	e3

Measuring error

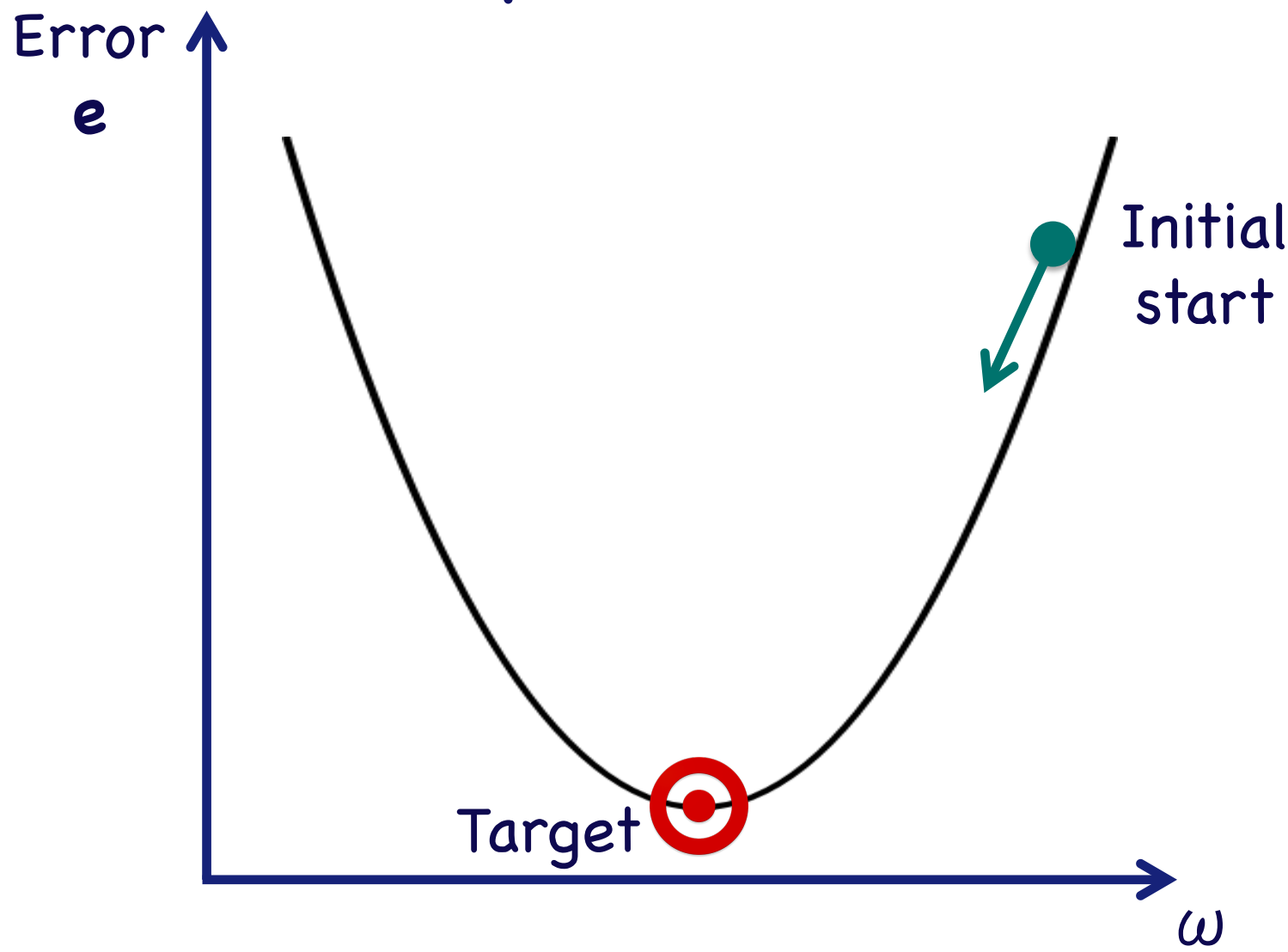


Parameter	Error
ω1	e1
ω2	e2
ω3	e3

Cost function

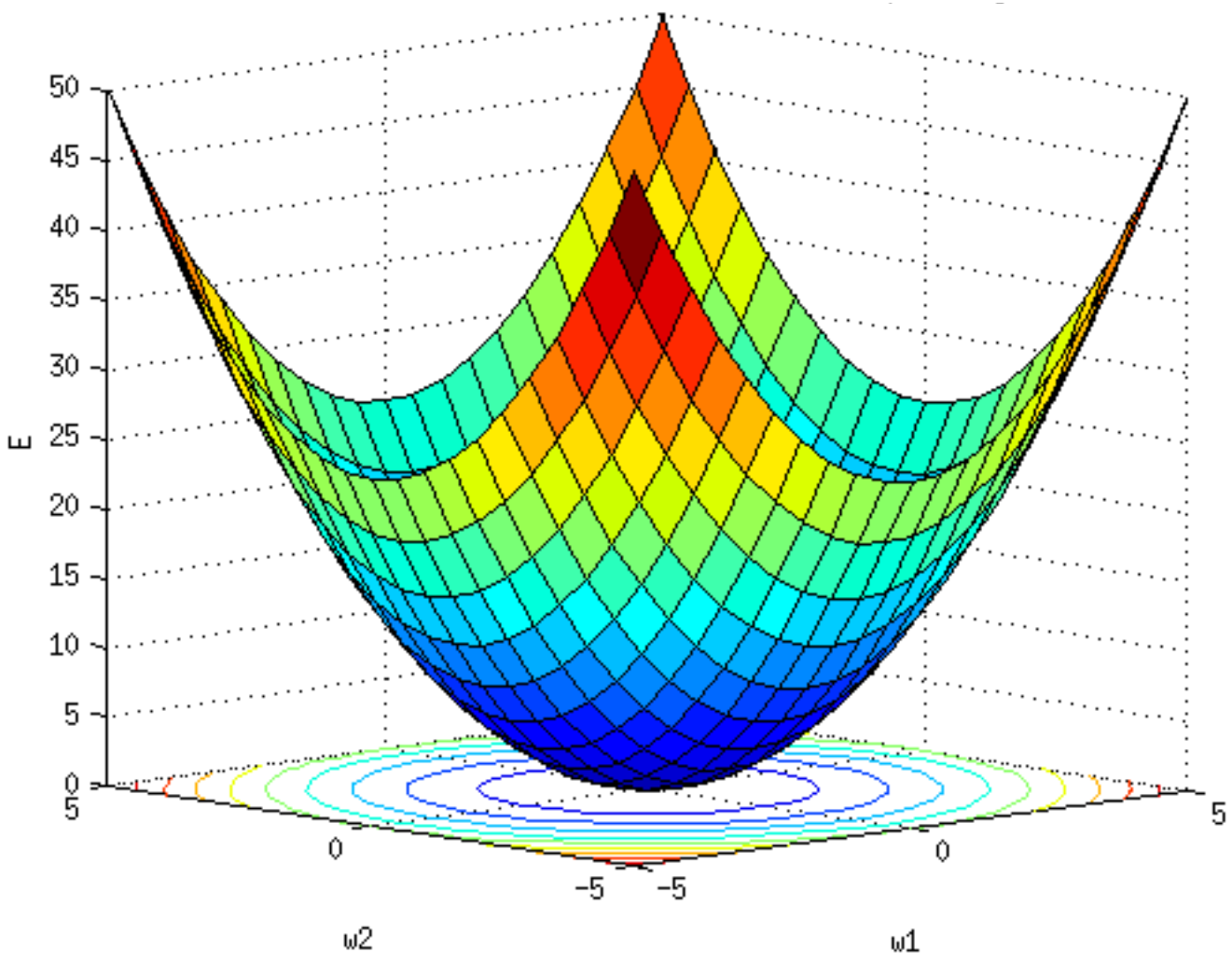


Optimization



Of course, there are many different types of cost functions and optimization algorithms, but the general idea is very similar.

Ideal Cost Function



Real-world Cost Function

Convolutional Neural Network

Quadratic Discriminant Analysis

Generative Adversarial Networks

Hierarchical clustering

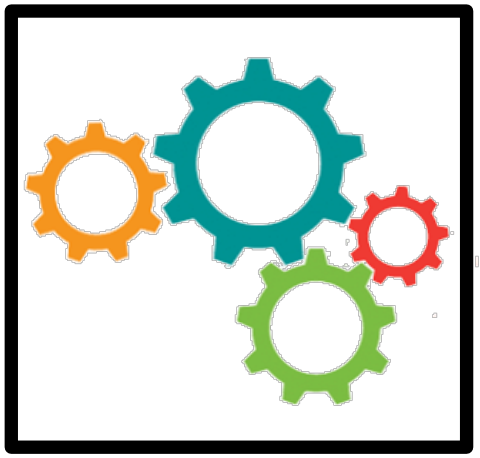
Nearest Neighbor

Radial Basis Function Network

Support Vector Machine

Artificial Neural Network

K-means



Linear Regression

Naïve Bayes

Logistic Regression

AdaBoost

Boosting

Decision Tree

Self-Organizing Map

Stepwise Regression

Ridge Regression

Recurrent Neural Network

Random Forest

LASSO

Principle Component Analysis

Bayesian Network

Ordinary Least Squares Regression

Linear Discriminant Analysis

Gaussian Mixture Model

Convolutional Neural Network

Quadratic Discriminant Analysis

Generative Adversarial Networks

Hierarchical clustering

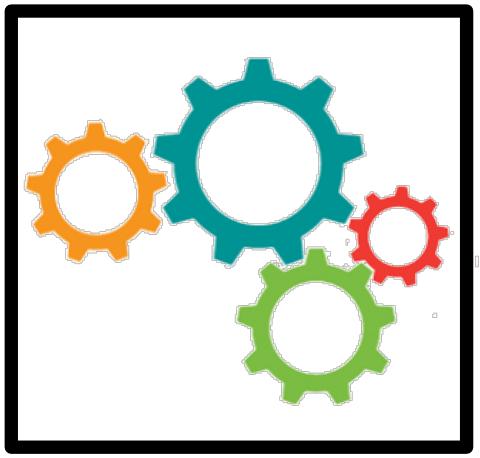
Nearest Neighbor

Radial Basis Function Network

Support Vector Machine

Artificial Neural Network

K-means



Linear Regression

Naïve Bayes

Logistic Regression

AdaBoost

Boosting

Decision Tree

Self-Organizing Map

Stepwise Regression

Ridge Regression

Recurrent Neural Network

Random Forest

LASSO

Principle Component Analysis

Bayesian Network

Ordinary Least Squares Regression

Linear Discriminant Analysis

Gaussian Mixture Model

http://scikit-learn.org/stable/

Machine Learning

what society thinks I do

what my friends think I do

what my parents think I do

$$L_P = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i$$

$$\alpha_i \geq 0, \forall i$$

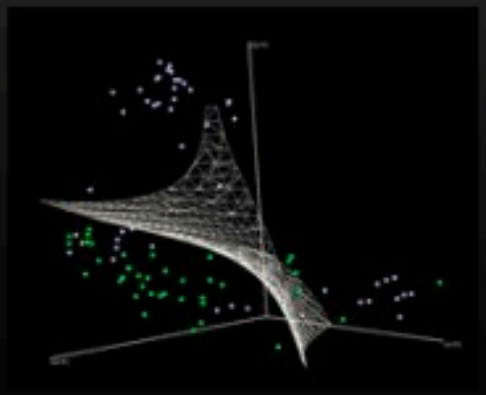
$$\mathbf{w} = \sum_{i=1}^{l} \alpha_i y_i \mathbf{x}_i, \sum_{i=1}^{l} \alpha_i y_i = 0$$

$$\nabla \hat{g}(\theta_t) = \frac{1}{n} \sum_{i=1}^{n} \nabla \ell(x_i, y_i; \theta_t) + \nabla r(\theta_t).$$

$$\theta_{t+1} = \theta_t - \eta_t \nabla \ell(x_{i(t)}, y_{i(t)}; \theta_t) - \eta_t \cdot \nabla r(\theta_t)$$

$$\mathbb{E}_{i(t)}[\ell(x_{i(t)}, y_{i(t)}; \theta_t)] = \frac{1}{n} \sum_i \ell(x_i, y_i; \theta_t).$$

what other programmers think I do



what I think I do

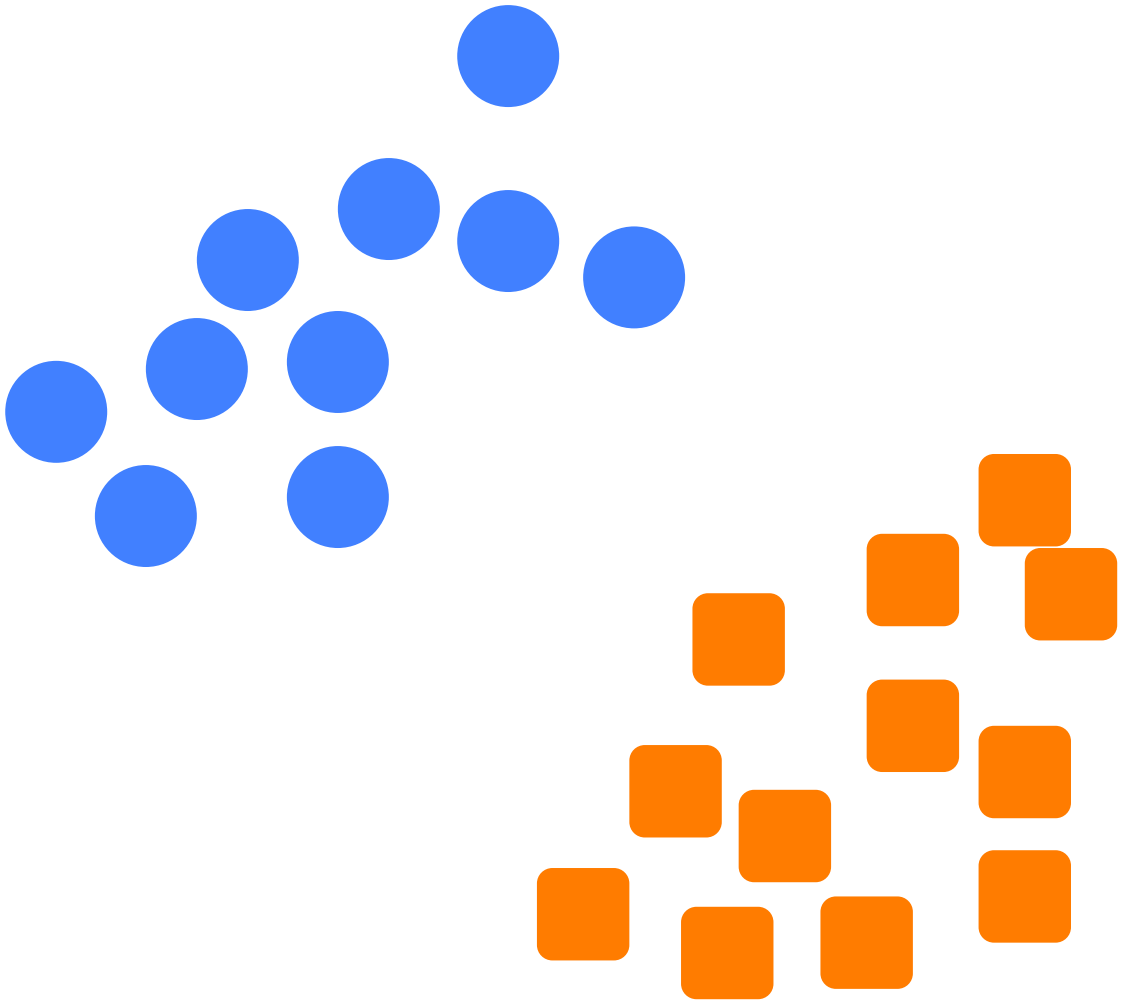
```
>>> from sklearn import svm
```

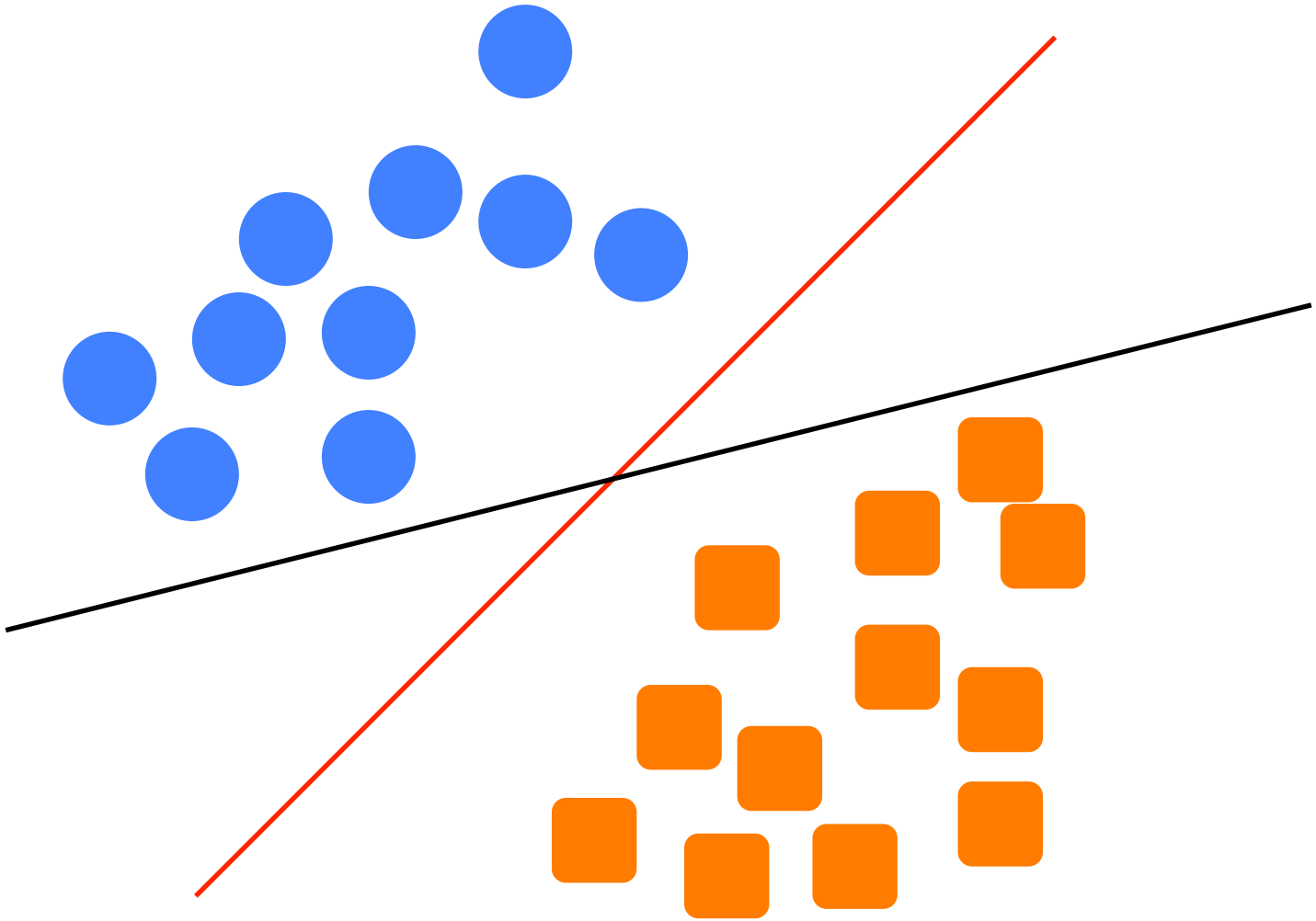
what I really do

Go to notebook 01

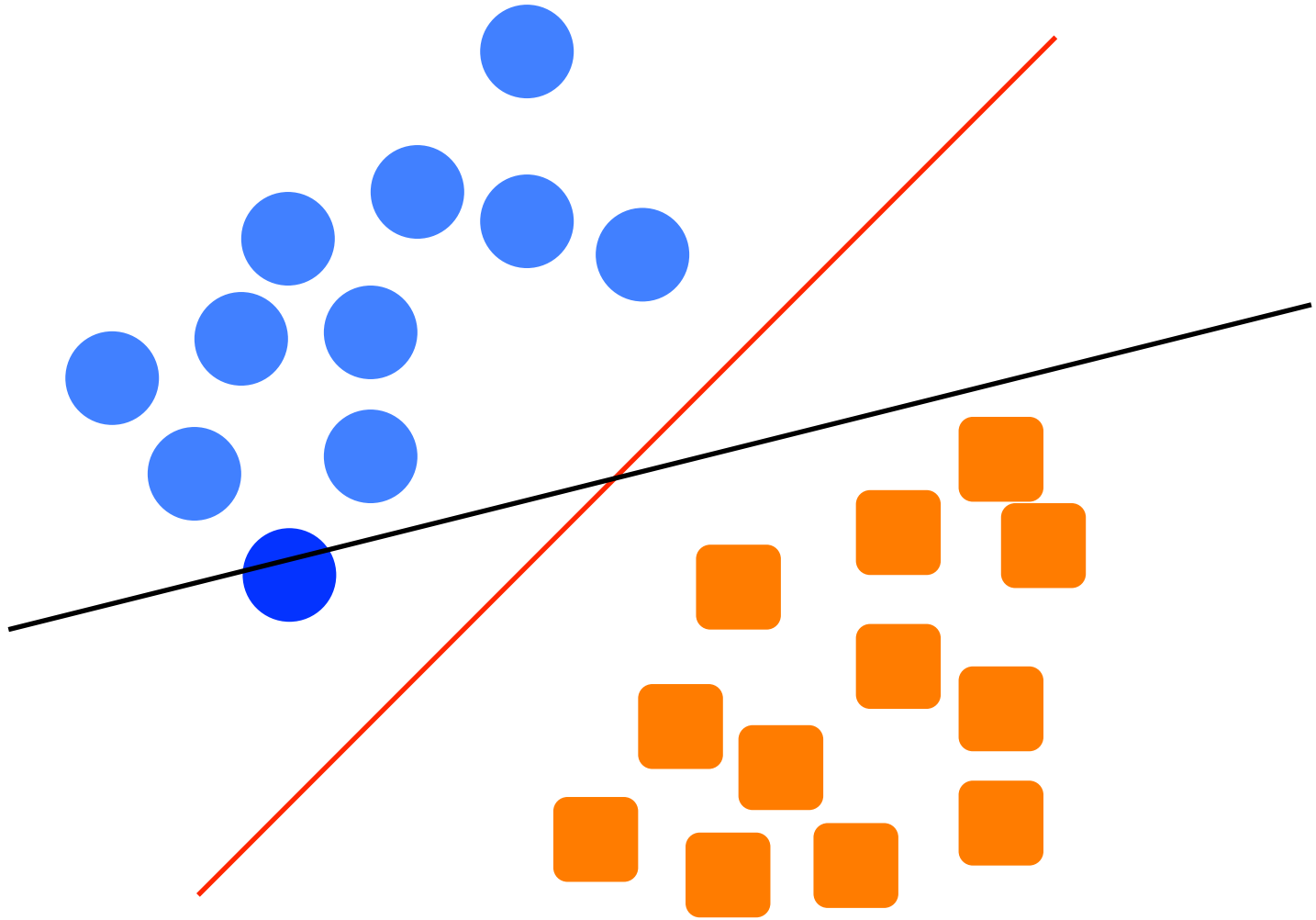
Classification

Support Vector Machine
Artificial Neural Network

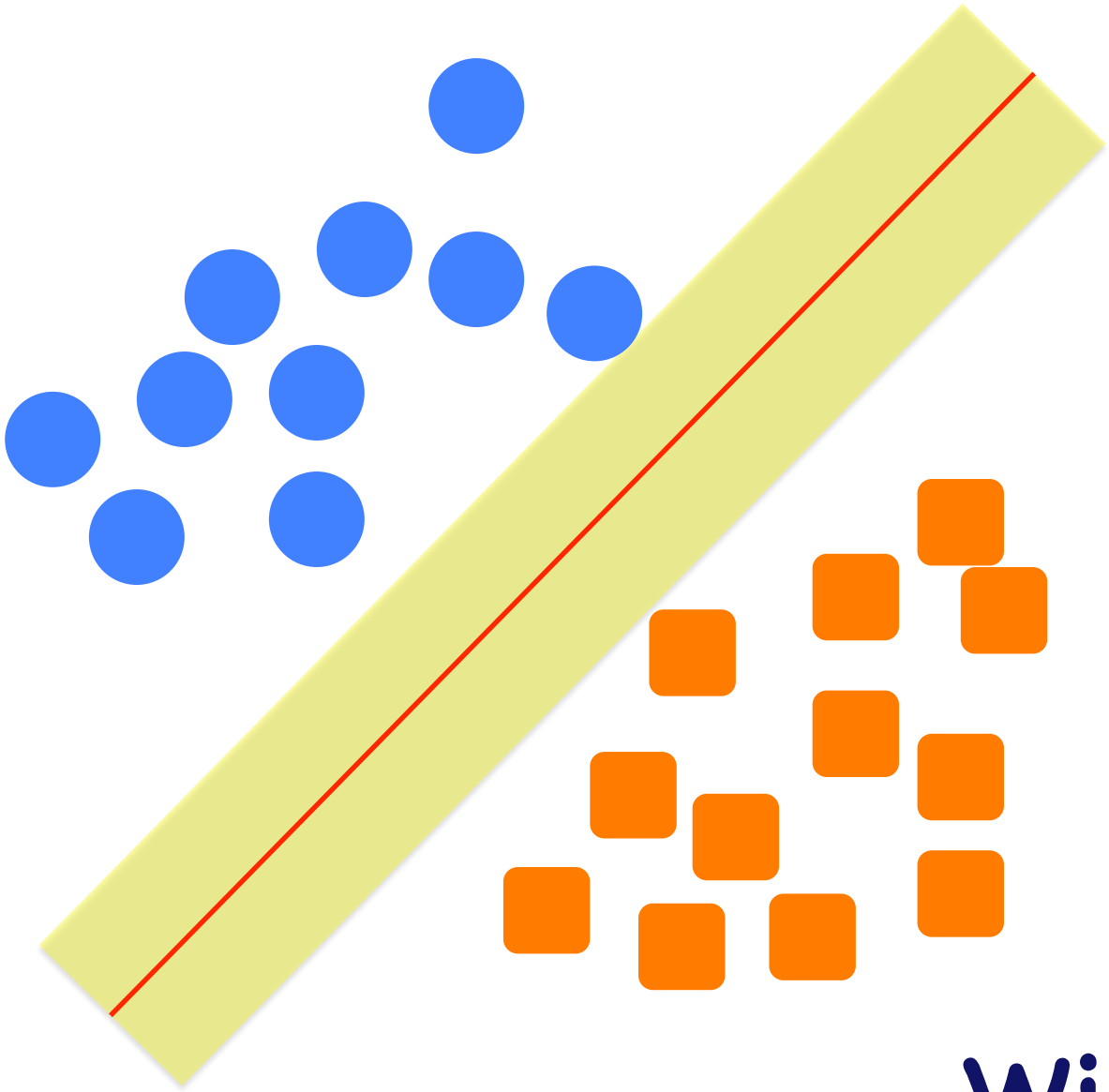




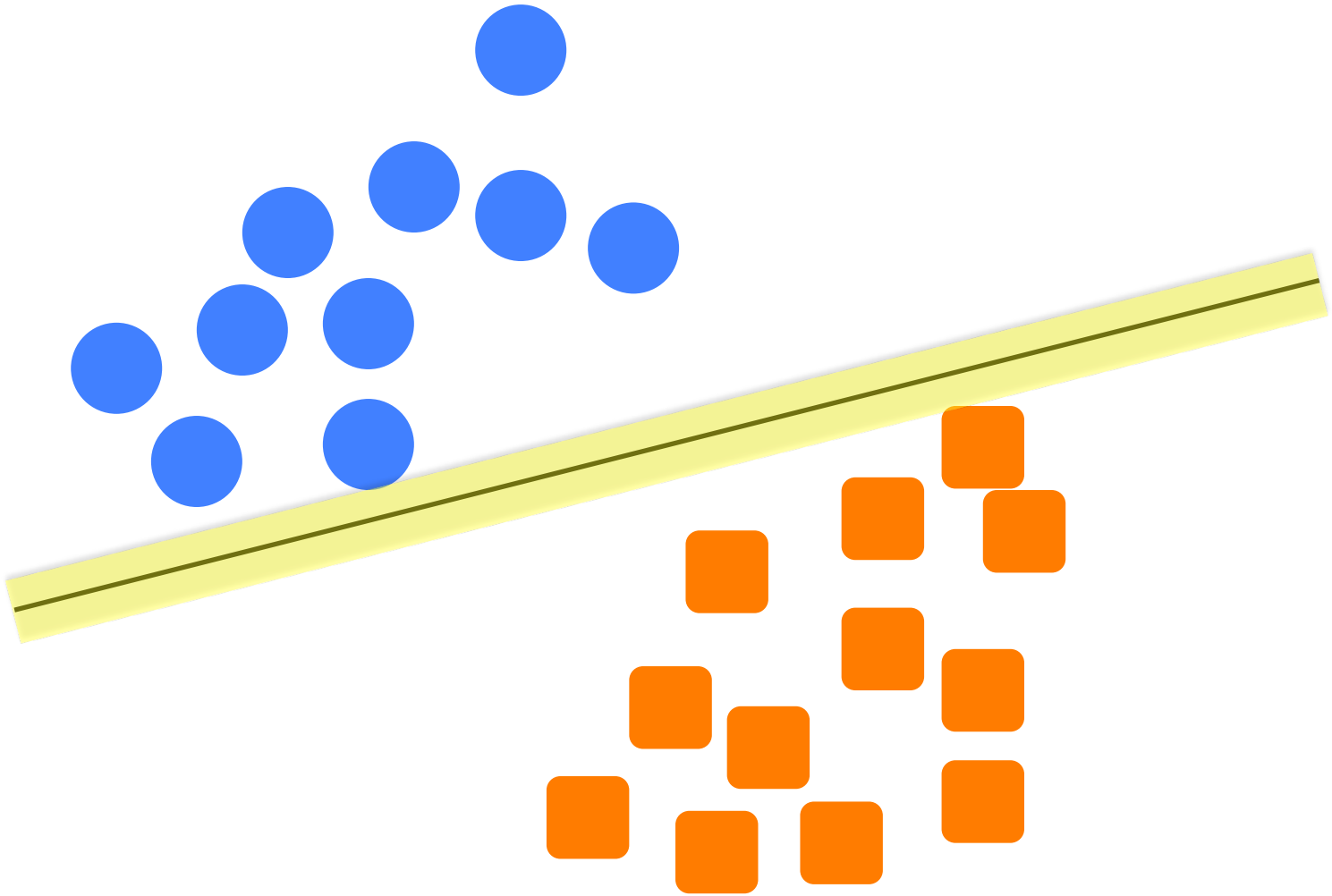
Which one is better?



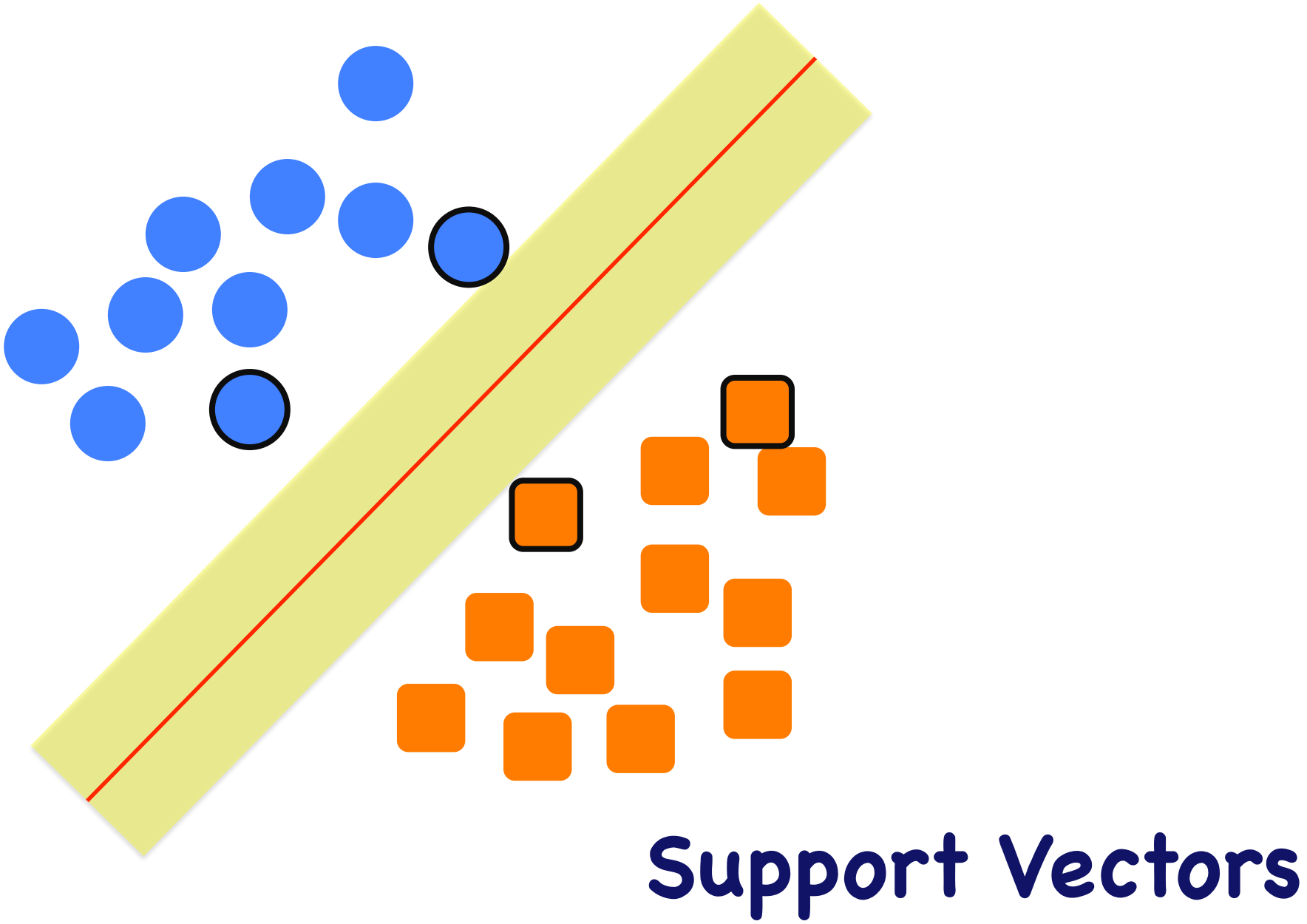
A new data point

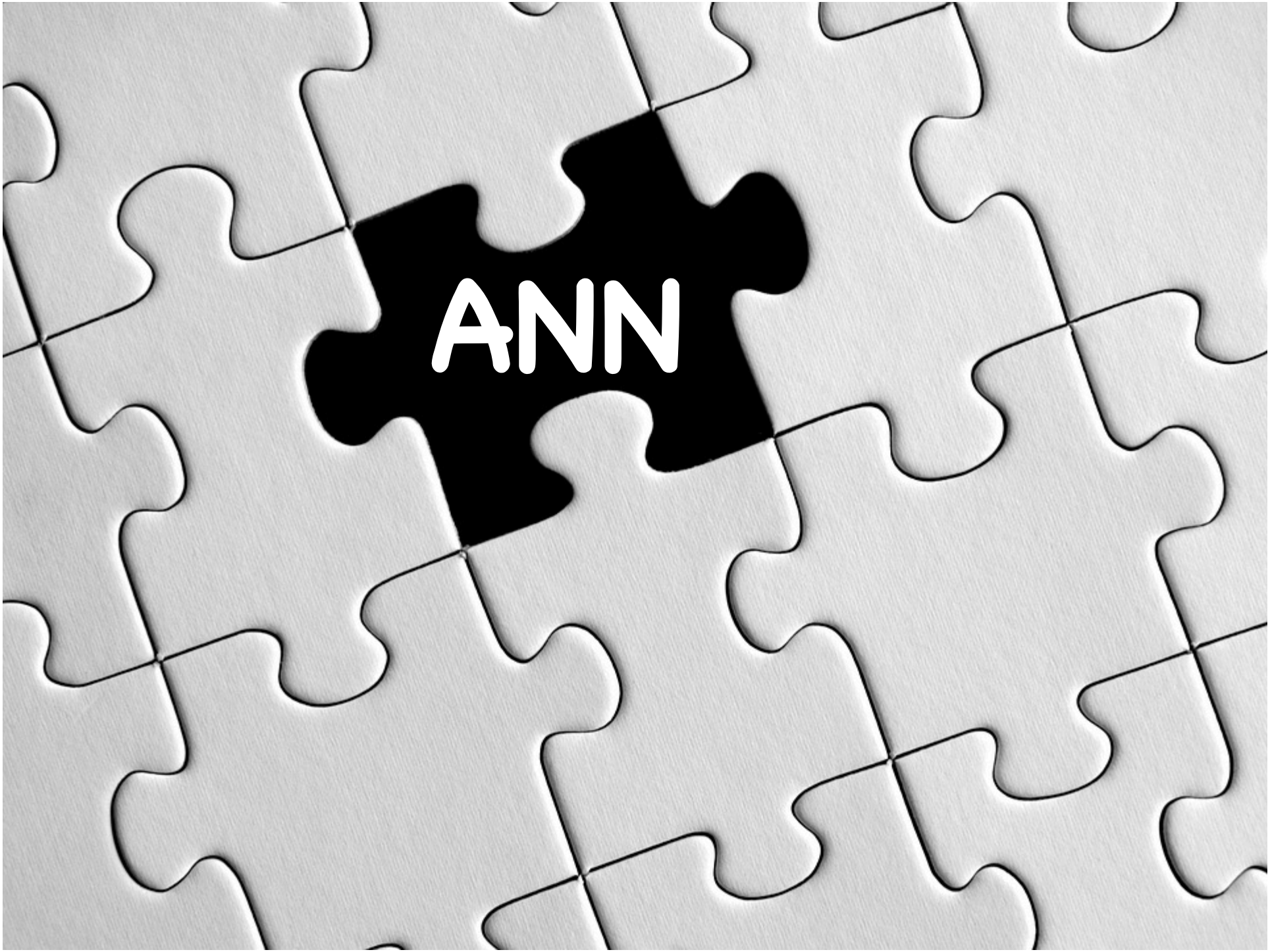


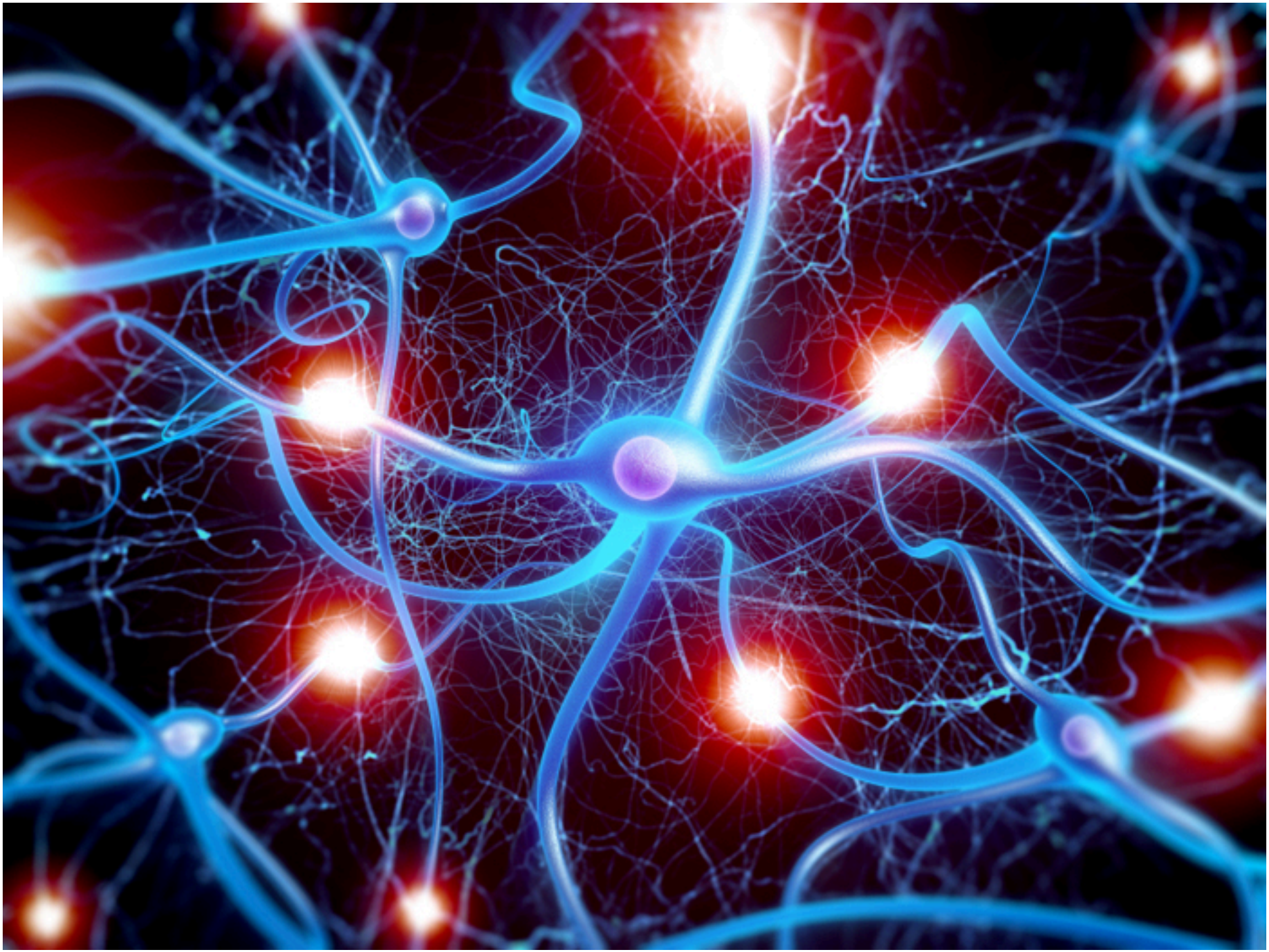
Wide margin

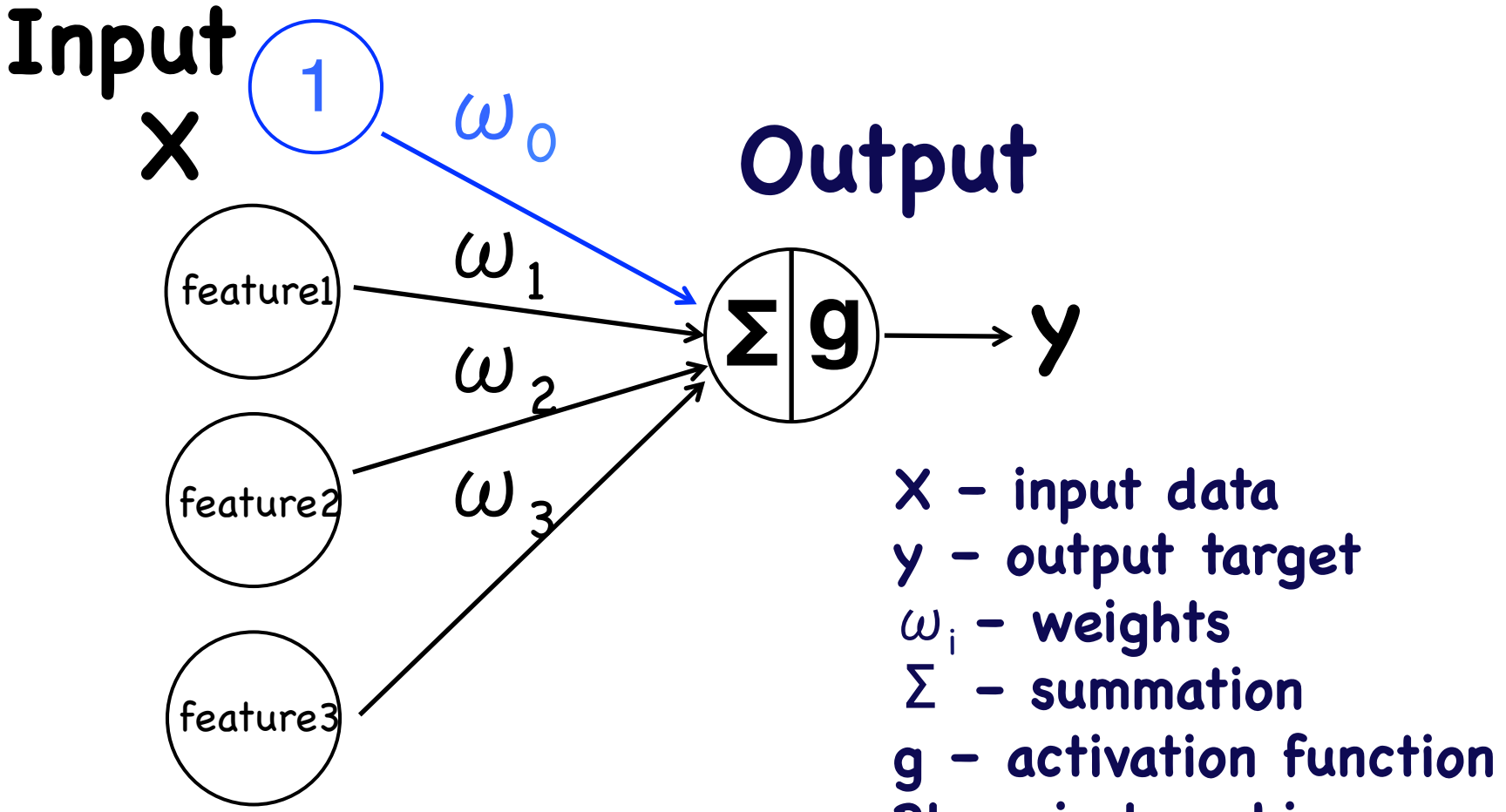


Narrow margin


Support Vectors

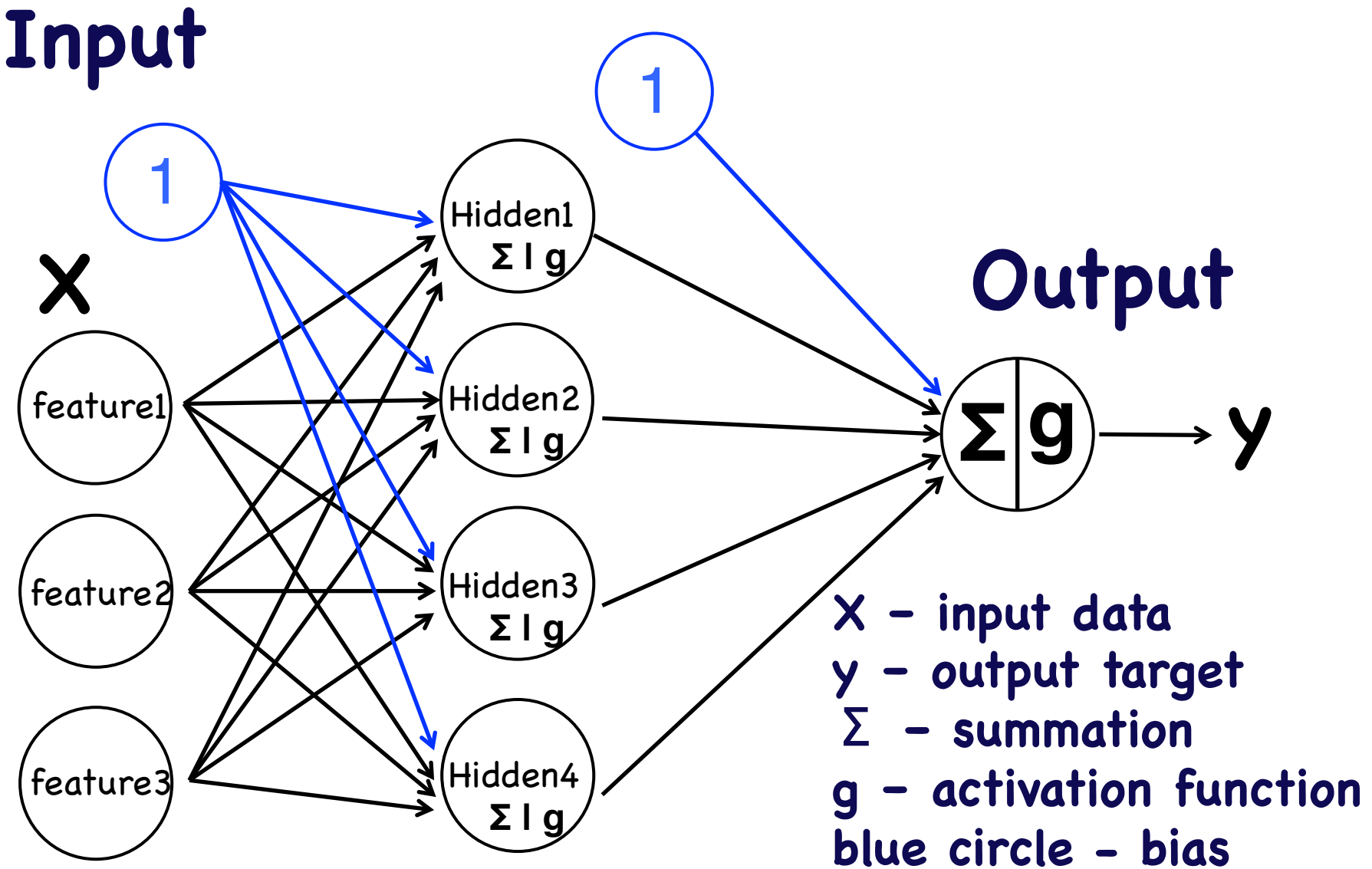

ANN





$$z = \Sigma = \omega_0 x_0 + \omega_1 x_1 + \omega_2 x_2 + \omega_3 x_3 + \ldots + \omega_n x_n$$

$$y = g(\omega_0 x_0 + \omega_1 x_1 + \omega_2 x_2 + \omega_3 x_3 + \ldots + \omega_n x_n)$$


Input
1
1
X
feature1
feature2
feature3
Hidden1
Σ I g
Hidden2
Σ I g
Hidden3
Σ I g
Hidden4
Σ I g
Output
Σ
g
y
X - input data
y - output target
Σ - summation
g - activation function
blue circle - bias


YOU'RE IN MY SPOT
GRAPHICS GARAGE

Intuitive
Artificial Neural Network

Input

.
.
.

w_1

w_2

w_n

$F(eye \times w_1 + nose \times w_2 + \ldots + mouth \times w_n)$

Output

Intuitive
Artificial Neural Network

Input

.
.
.

w_1

w_2

w_n

$F(eye \times w_1 + nose \times w_2 + ... + mouth \times w_n)$

Output

error feedback

Intuitive
Artificial Neural Network

Input

.
.
.

w_1

w_2

w_n

F(eye × w_1+nose × w_2+...+mouth × w_n)

Output

error
feedback

Intuitive
Artificial Neural Network

Input

.
.
.

w_1

w_2

w_n

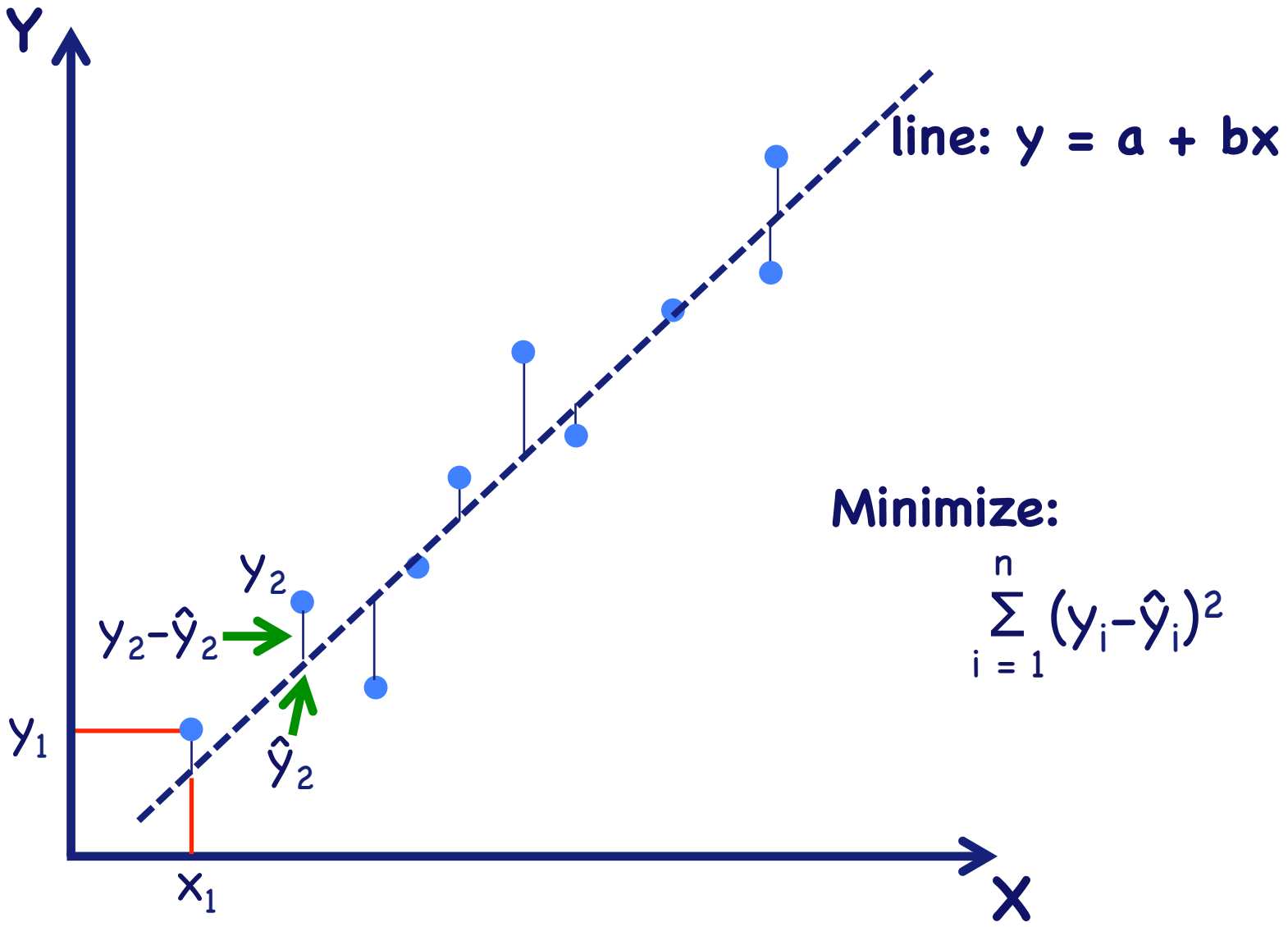
$F(eye \times w_1 + nose \times w_2 + \ldots + mouth \times w_n)$

Output

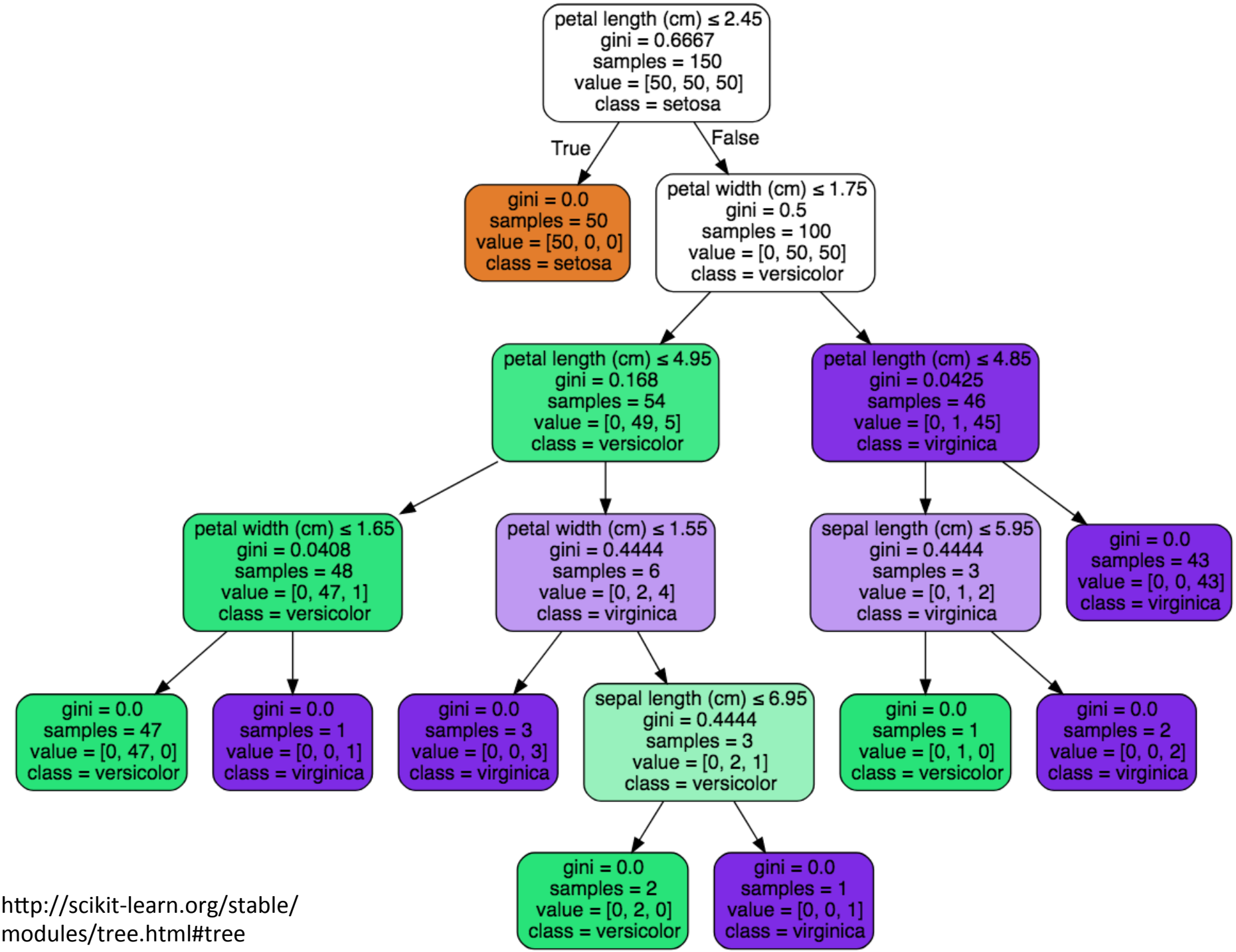
Go to notebook 02

Regression

Simple linear regression

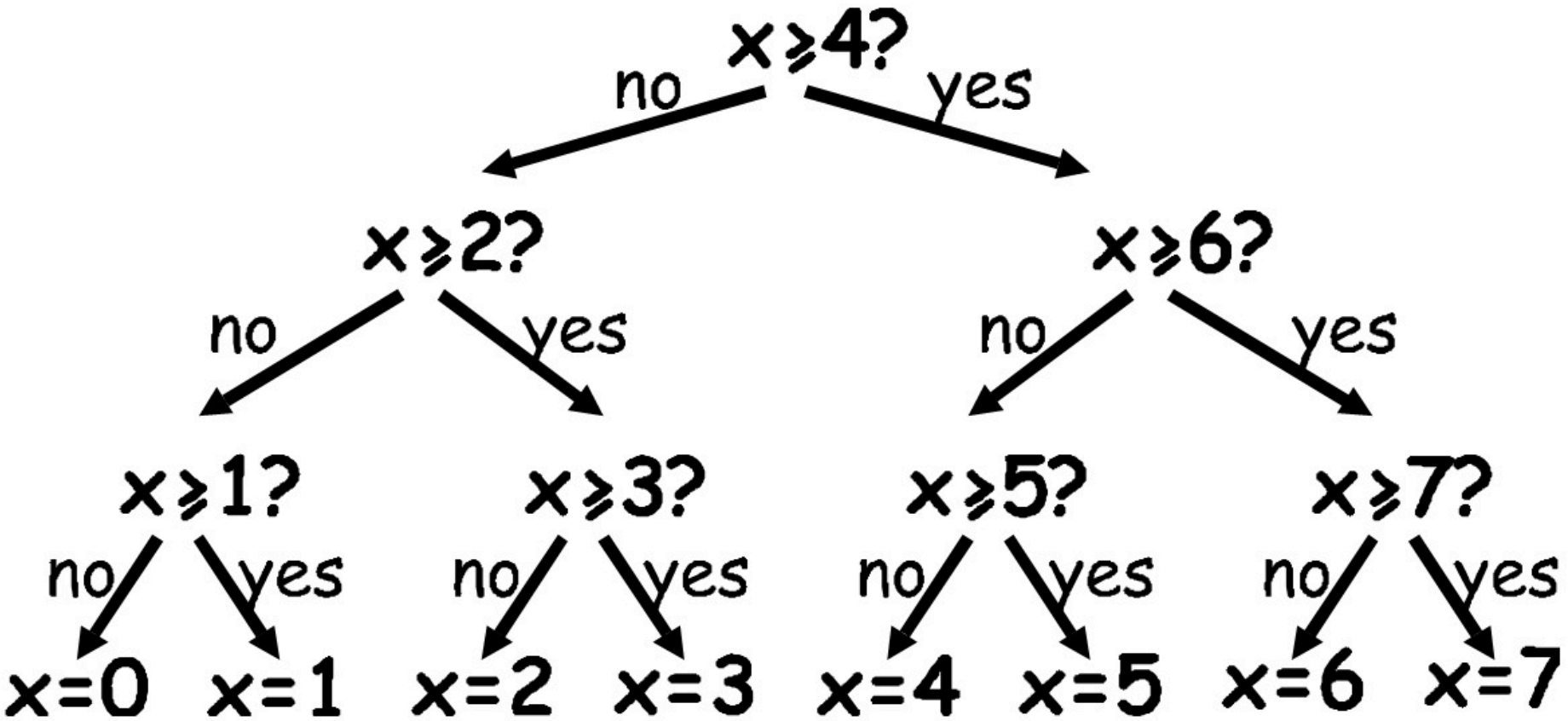


Random Forest

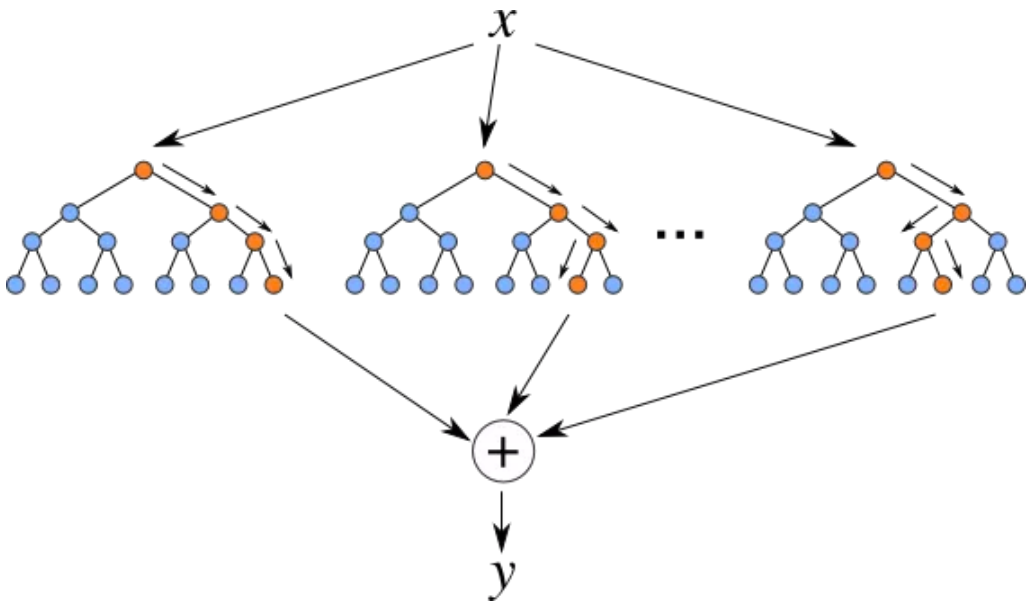


http://scikit-learn.org/stable/modules/tree.html#tree

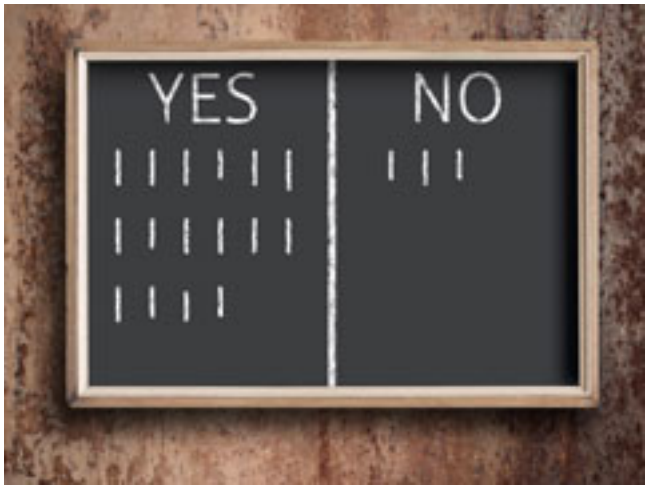
Decision Trees



Random Forest



Majority vote



Go to notebook 03

Unsupervised

Principle component analysis
K-means

PCA

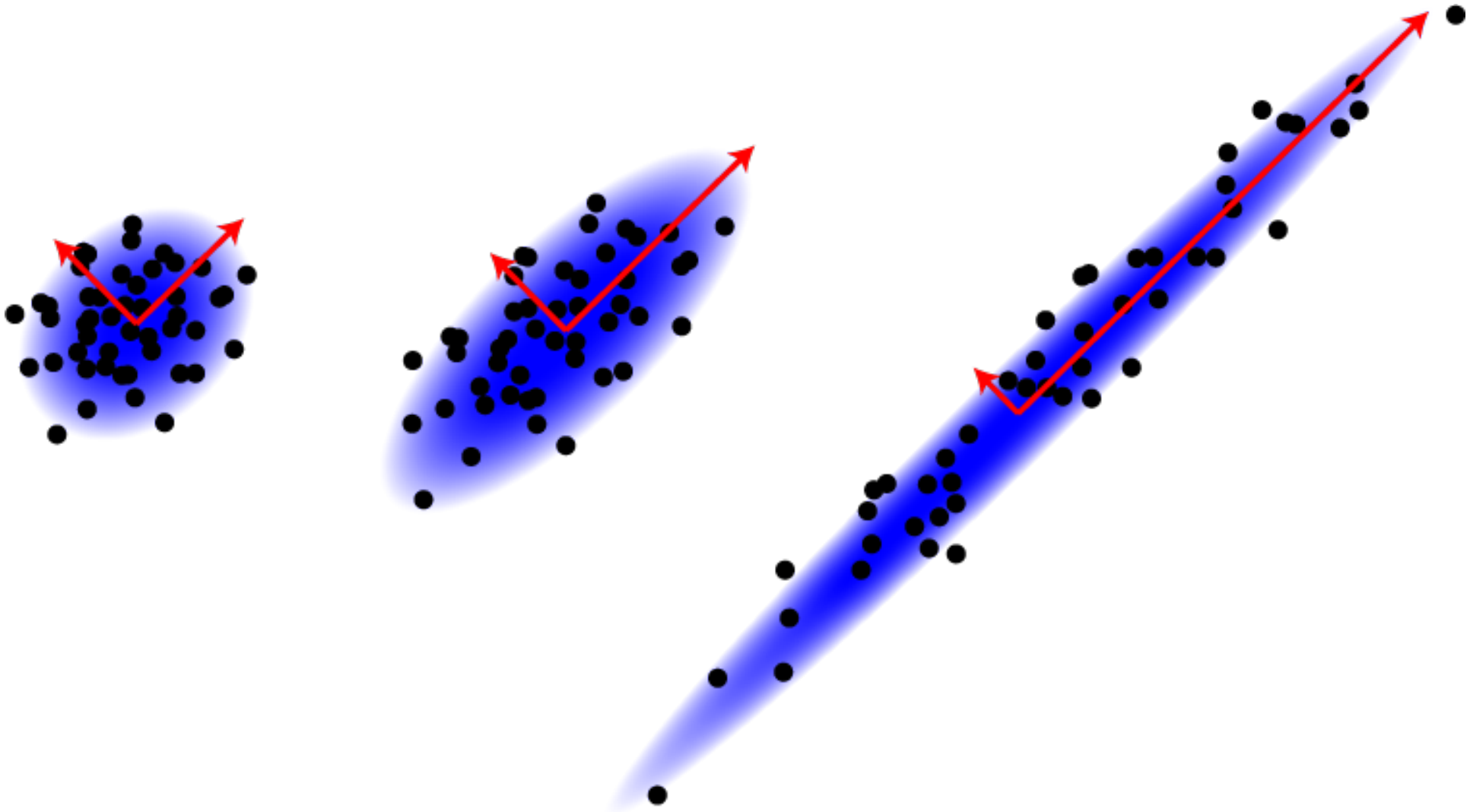
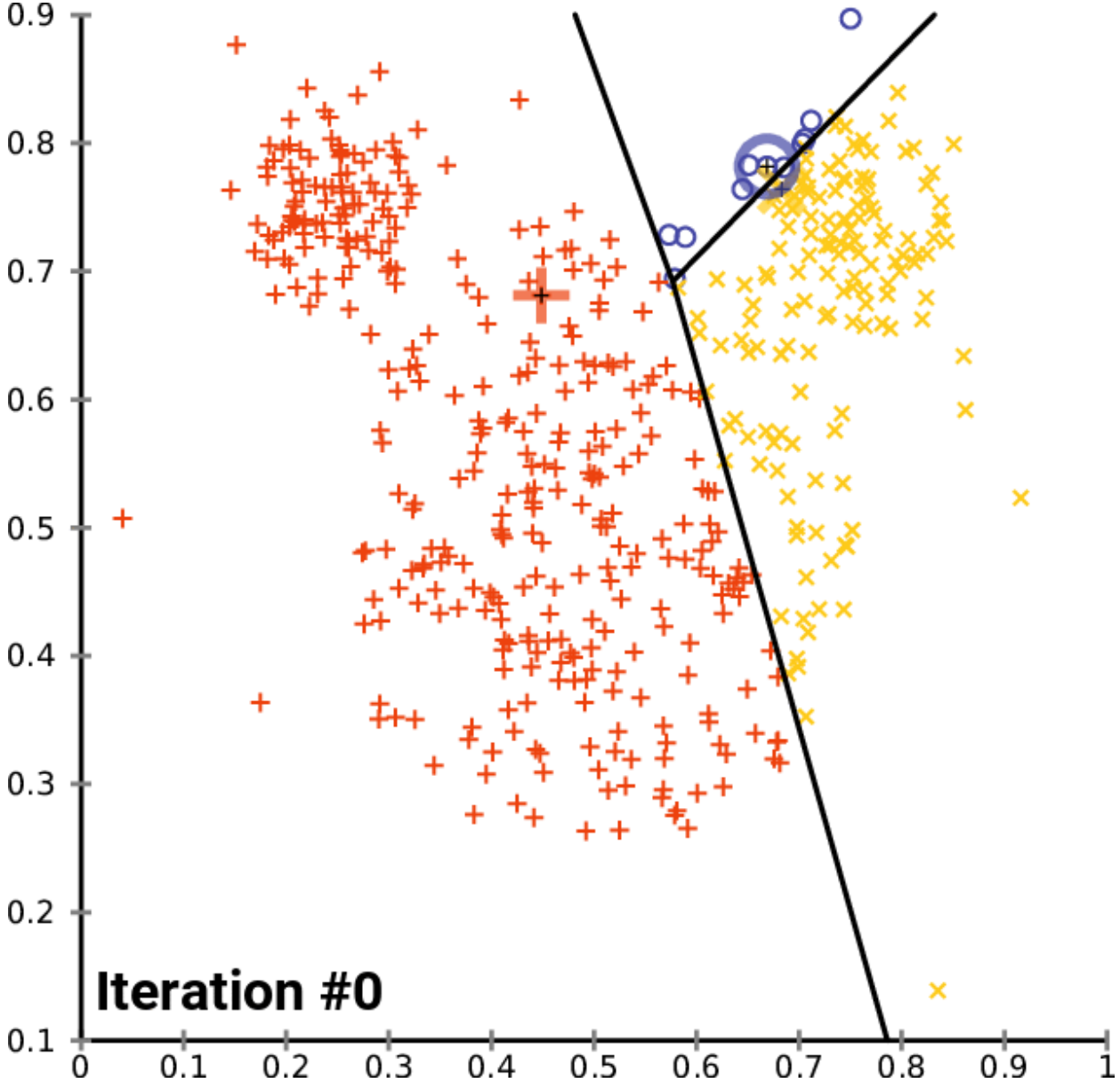


Figure from: https://shapeofdata.wordpress.com/2013/04/09/principle-component-analysis/

Kmeans

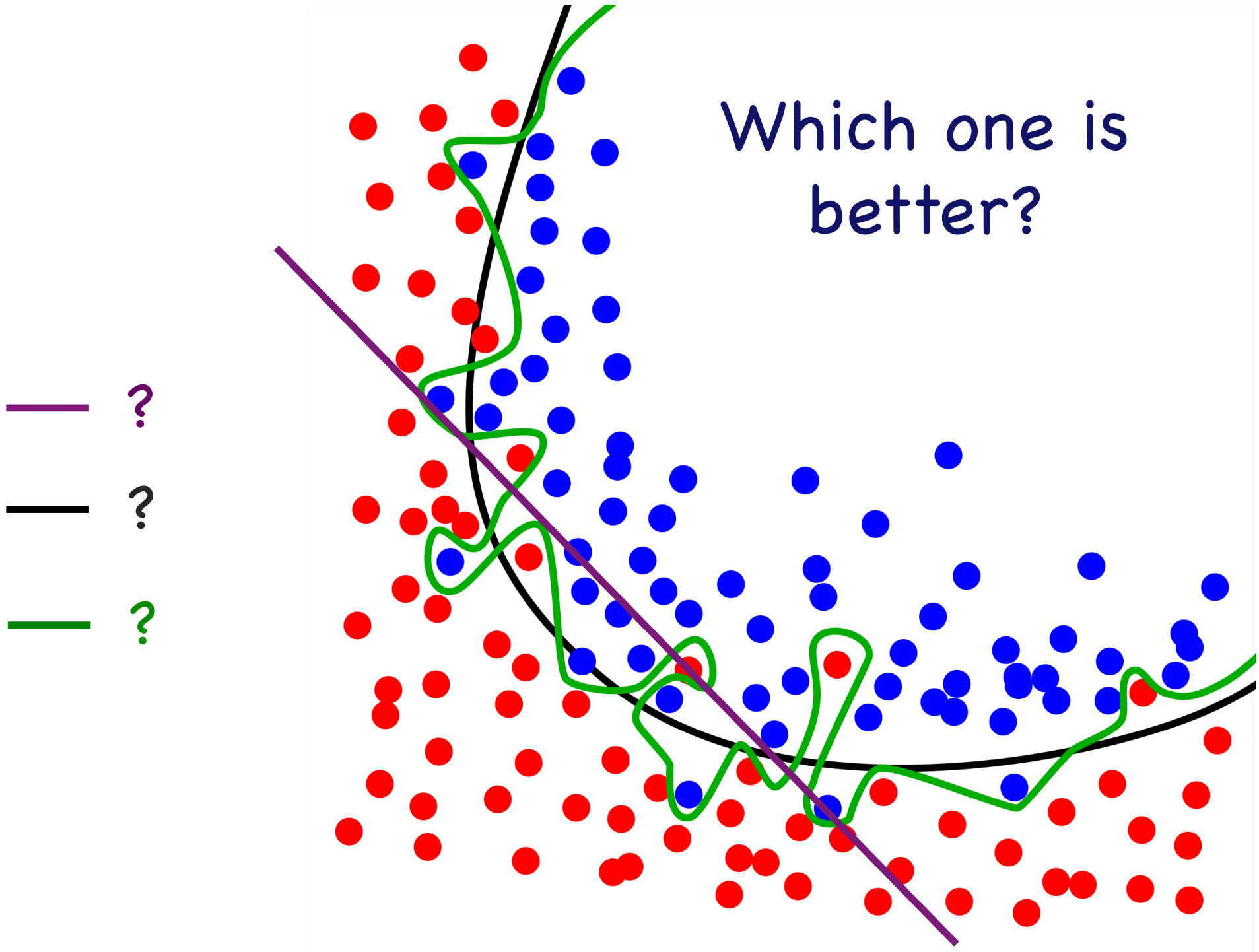


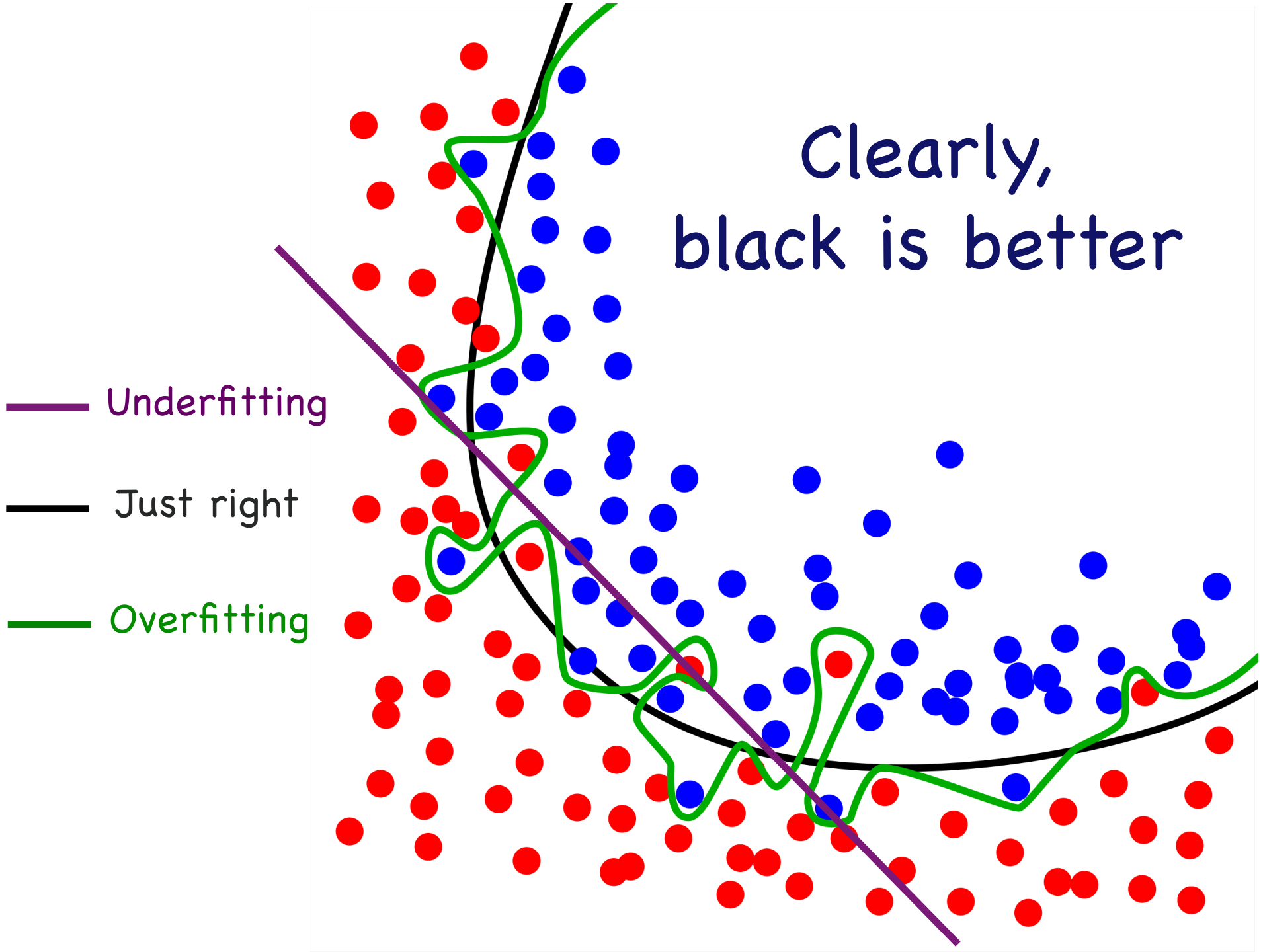
Go to notebook 04

More on common practices

Machine learning is all about

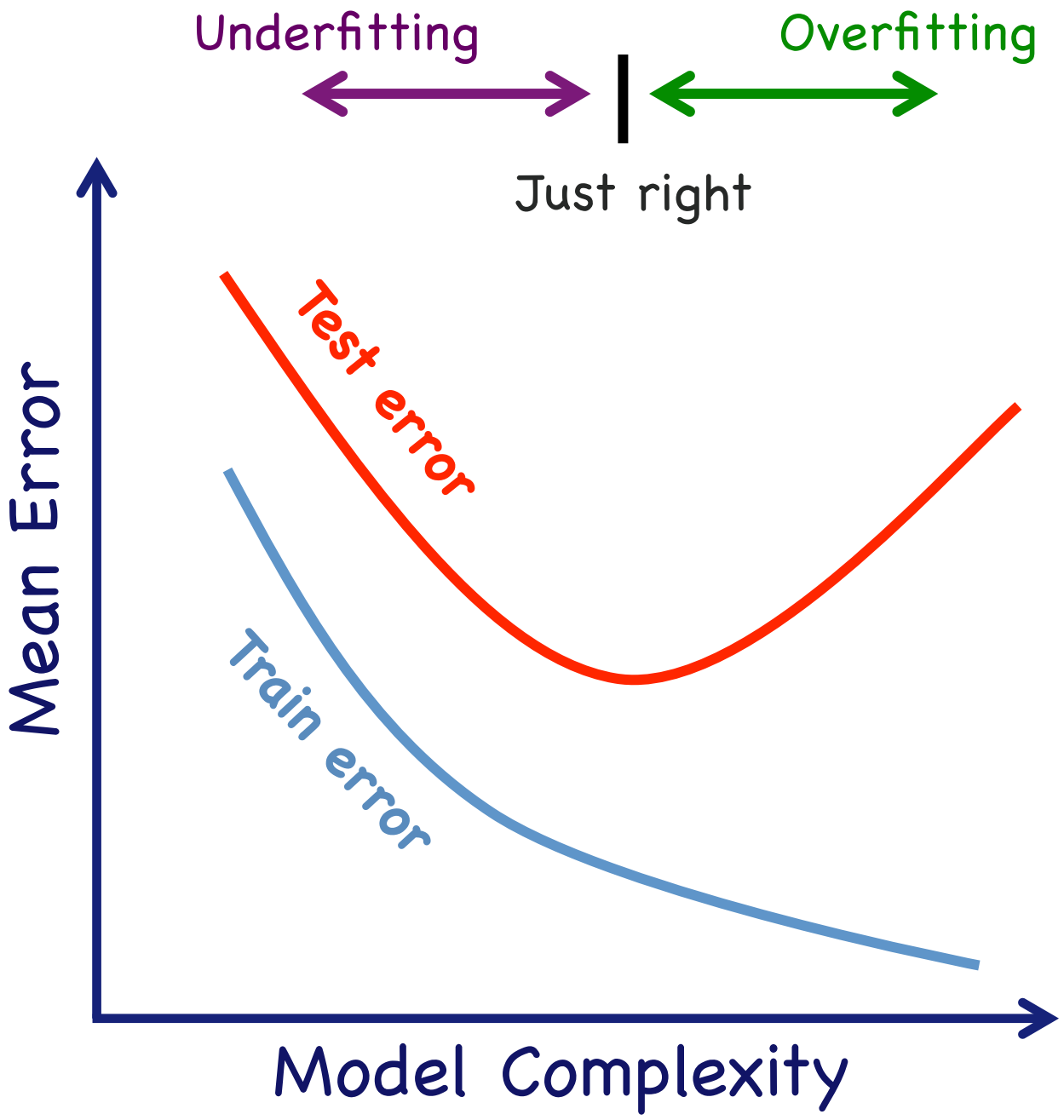
Generalization


Which one is better?
?
?
?

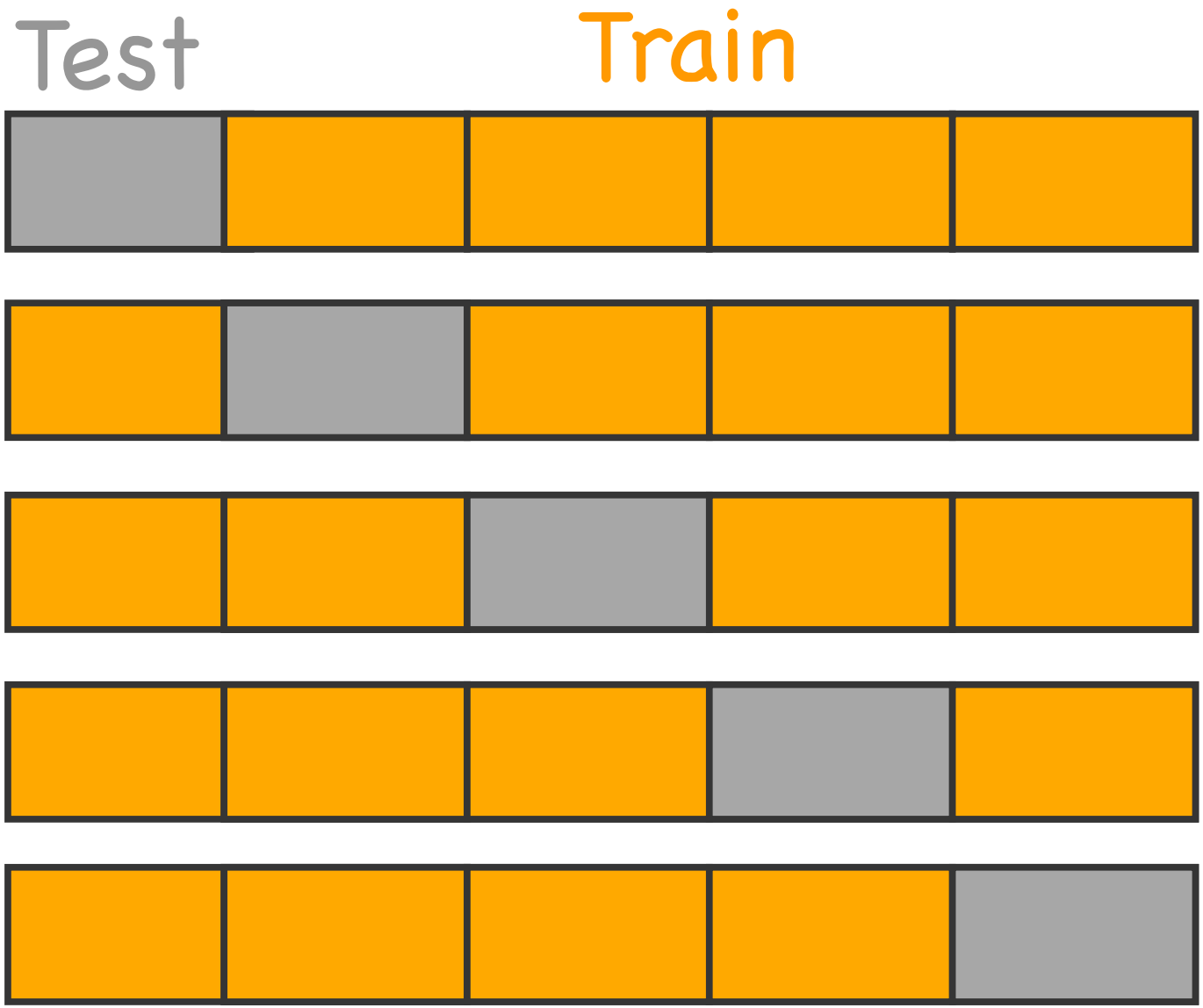

Clearly,
black is better
Underfitting
Just right
Overfitting

Train/test dataset split

Train/test dataset split


Underfitting
Overfitting
Just right
Test error
Train error
Mean Error
Model Complexity

5-fold cross-validation




Data
processing
Shift

Why do we need?

Age	Salary
24	$110,000
36	$130,000
38	$80,000
44	$420,000
27	$420,000
43	$12,000,000
…	…

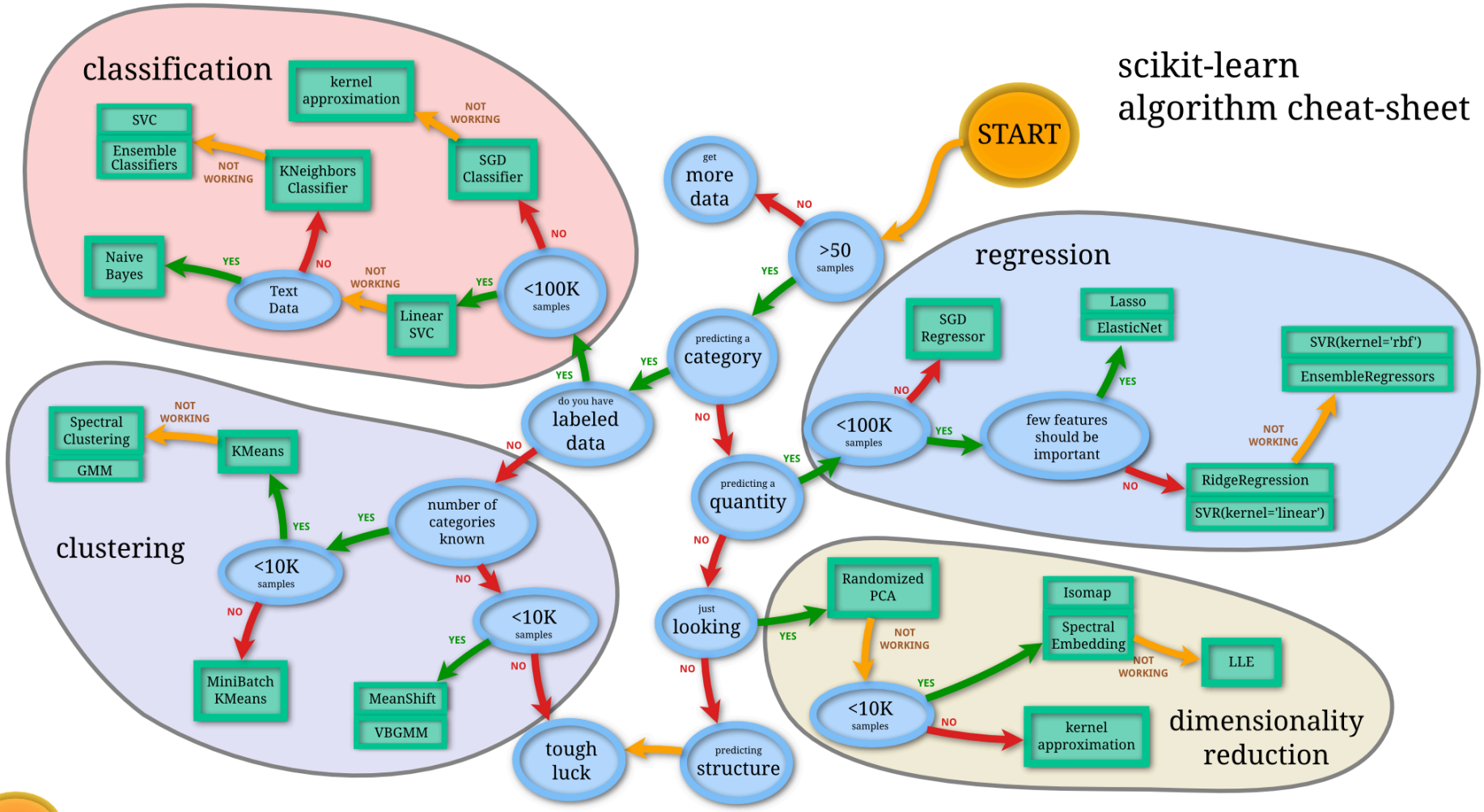
Many ways

- **Standardization**
  - Zero mean and unit variance
- **Scale to a range**
  - i.e. (0, 1)
- **Normalization**
  - Unit norm
- …

Check out: http://scikit-learn.org/stable/modules/preprocessing.html

Go to notebook 05

More resources



Interactive version: http://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

Conclude with notebook 06