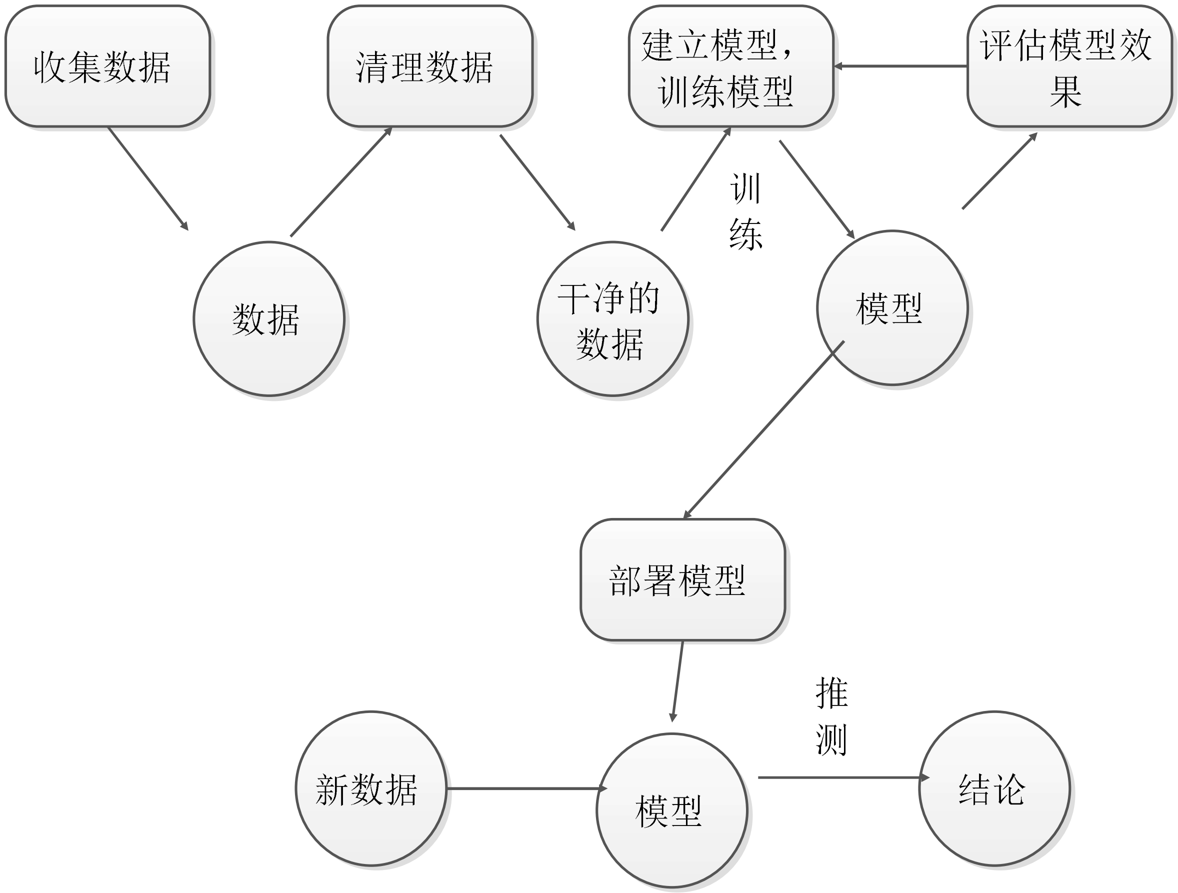
数据科学入门2.1：
机器学习介绍，以及更多回归

Introduction to Data Science
Part 2.1: Intro

□ Goals

- More in-depth stuff of machine learning
- Train-validation-test split
- Classification
- Clustering
- Neural networks
- Finally a glance of the tensorflow2 (python of course)

□ The routine of data science



$$\hat{\boldsymbol{Y}} = \boldsymbol{f}(\boldsymbol{X}, \boldsymbol{\beta})$$

◻ Types of Machine Learning

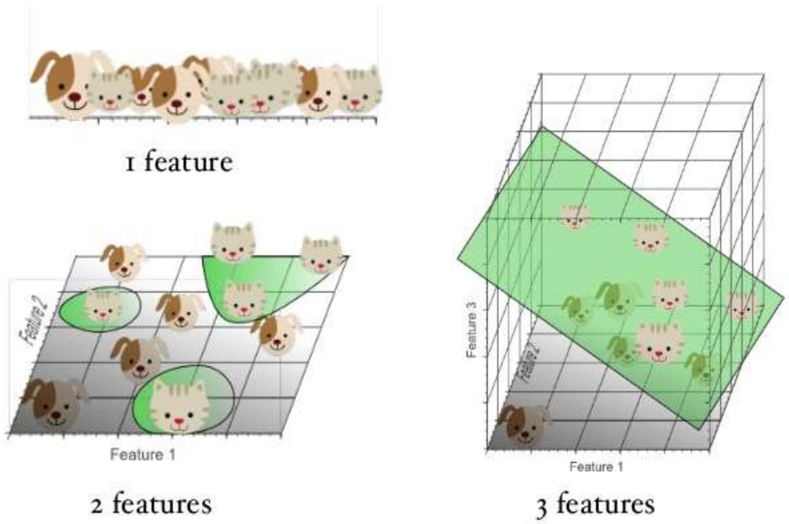
■ Supervised learning

- Trained with labeled data
- Regression
- Classification

■ Unsupervised learning

- Trained with unlabeled data
- Clustering
- Anomaly detection

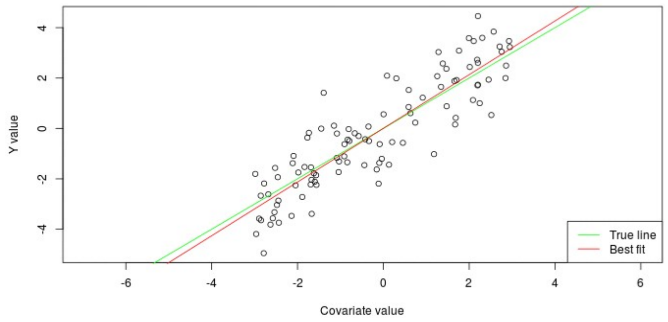
Supervised learning: with label



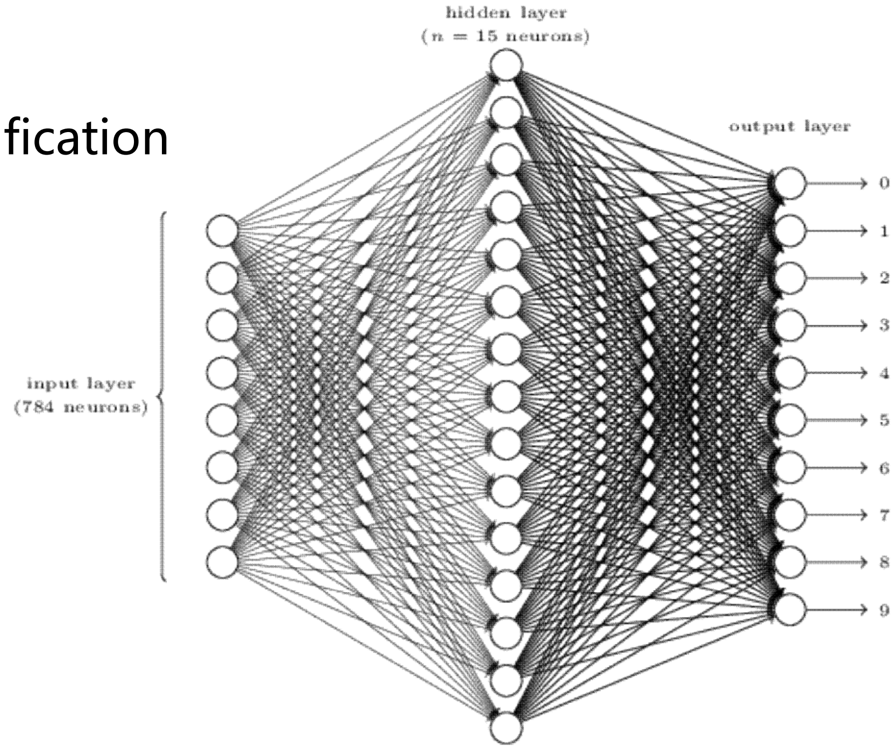
Unsupervised learning： no label

□ Supervised learning

Regression

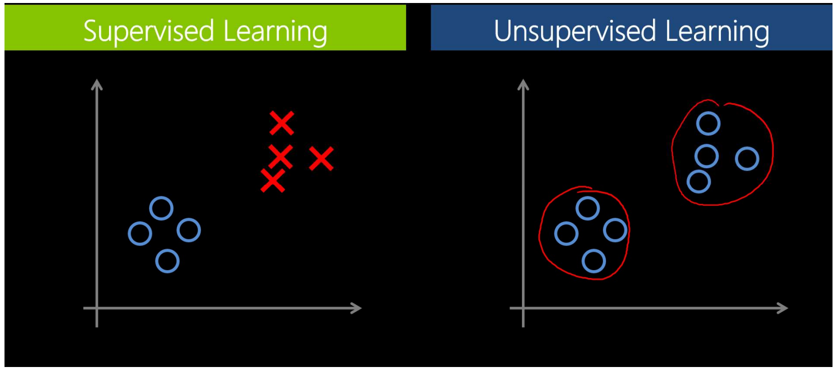


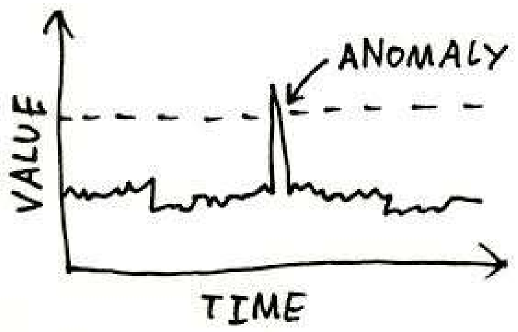
classification

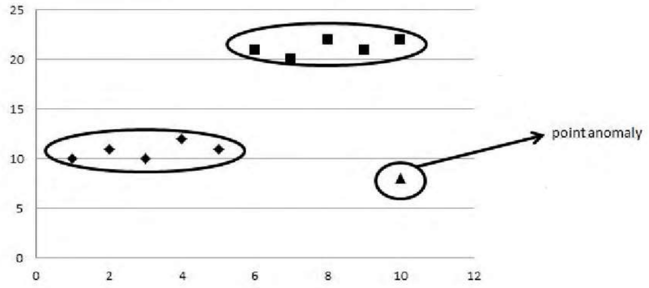


◻ Unsupervised Learning

- **Data has no label**
  - Clustering
  - Anomaly detection





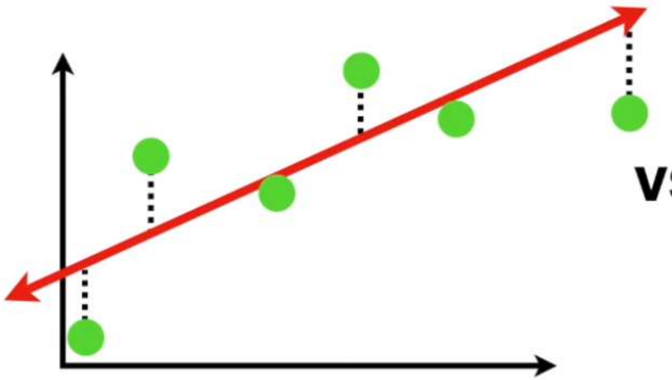


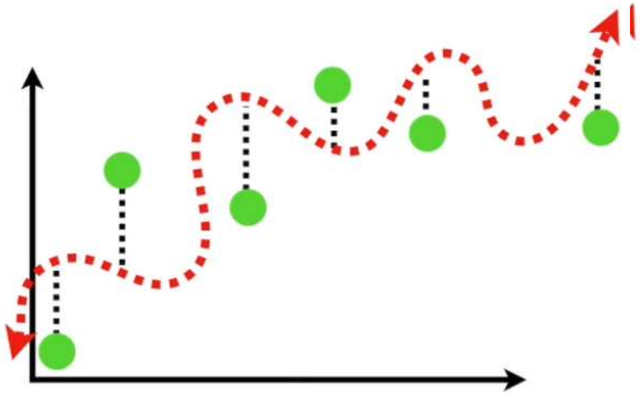
◻ More Regression

- Case 1 – 房屋价格预测
- Statistics and Machine Learning Toolbox

◻ Overfitting

- The bias variance tradeoff

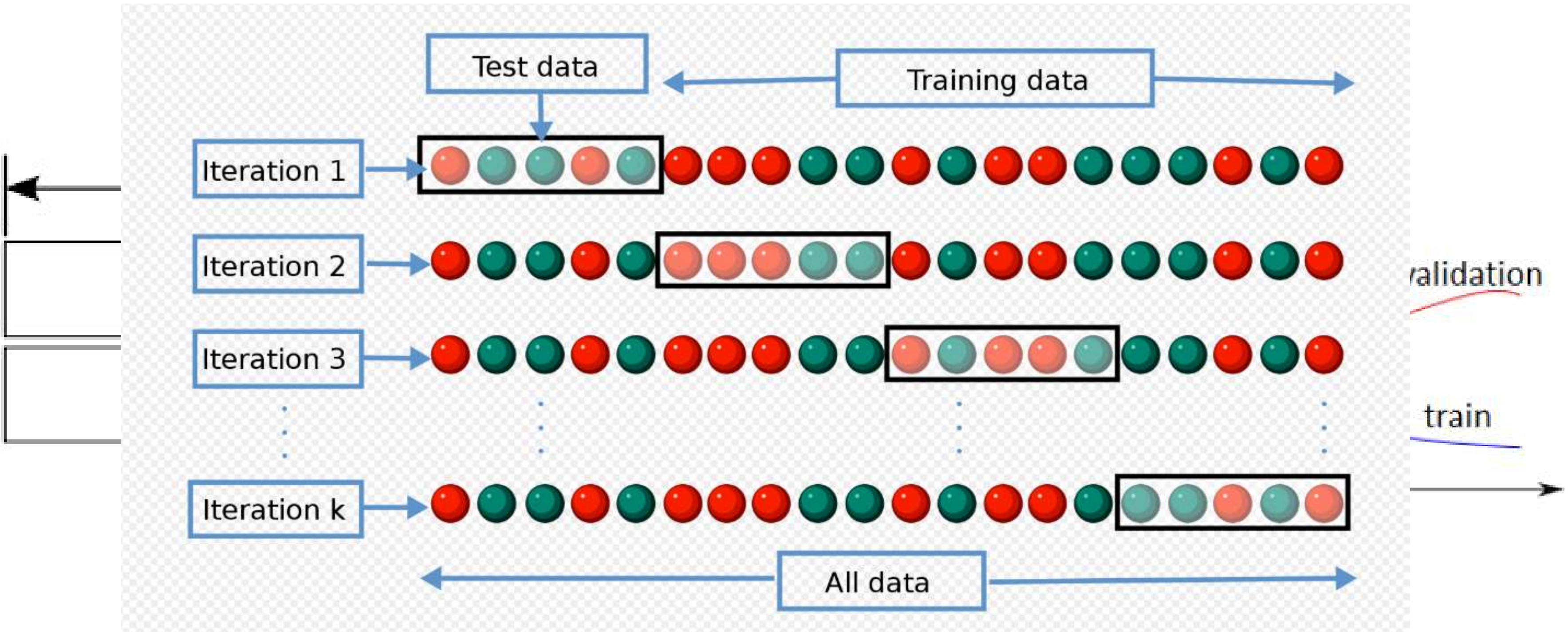




More bias less variance

Less bias more variance

□ Train-(validation)-test split



□ Selecting models and tuning hyper parameters

- **Most ML model are just black boxes**
- **More on this later**
- **Most hyper parameters defines the complexity**
  - adjust it to the complexity of you data (feature size)
  - adjust it to the size of you data (sample count)

Some regression models

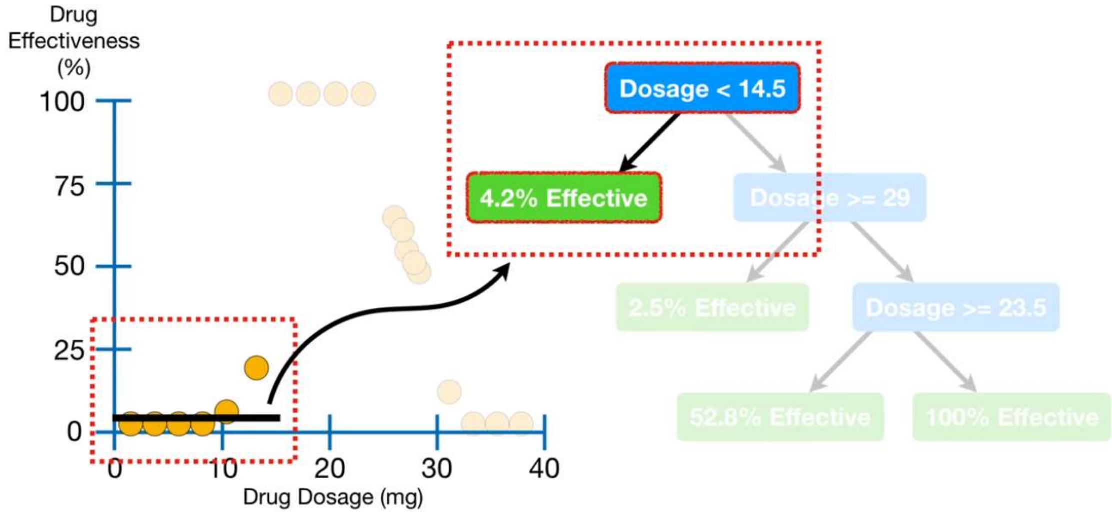
Linear models

- Simple, learned before
- Very high bias so very low variance

☐ Some regression models

■ regression tree

- Kind of decision tree
- Can be very non-linear and can have very low bias
- Can be seriously over fitting
- Ensemble is a good way to avoid over fitting
- Hypermeters:
  - ✓The levels limit
  - ✓The leaf nodes limit



□ Some regression models

- Support Vector Machine regression
  - To find the linear function

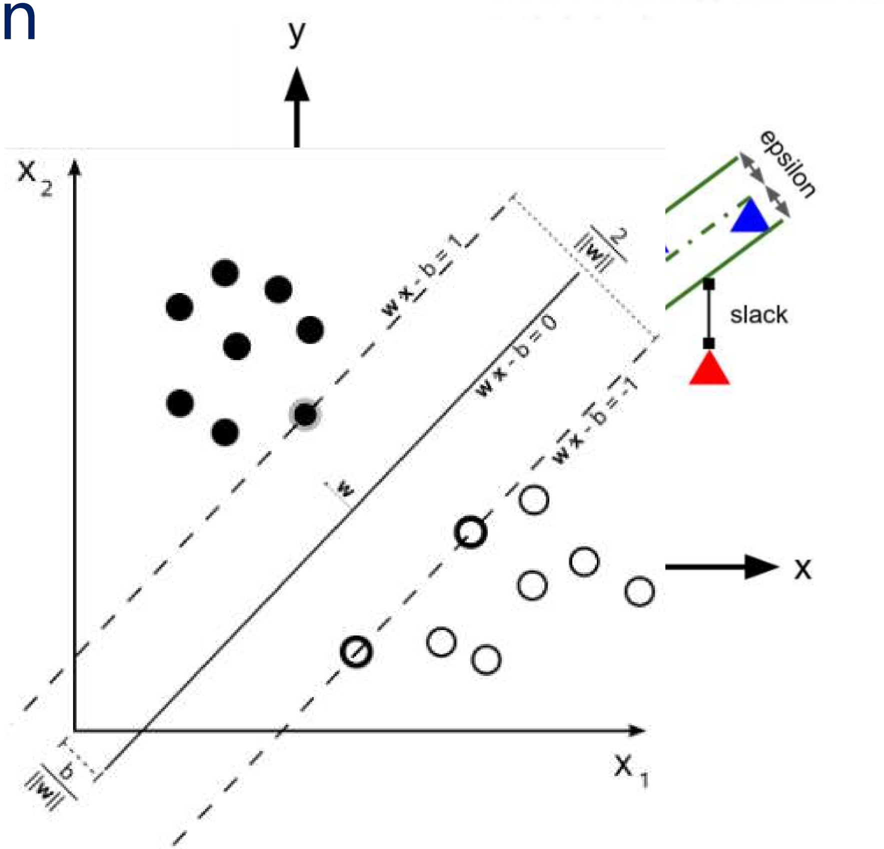
    F (X)=WX+b,
  - The object is to minimize:

$$MIN\ \frac{1}{2}||\boldsymbol{w}||^2$$

  - The constrain is:

$$|y_i - w_i x_i| \leq \varepsilon$$

  - The math is just like SVM classifier
  - Hyperparameter:
    - ✓C
    - ✓Kernals



□ Evaluate your models

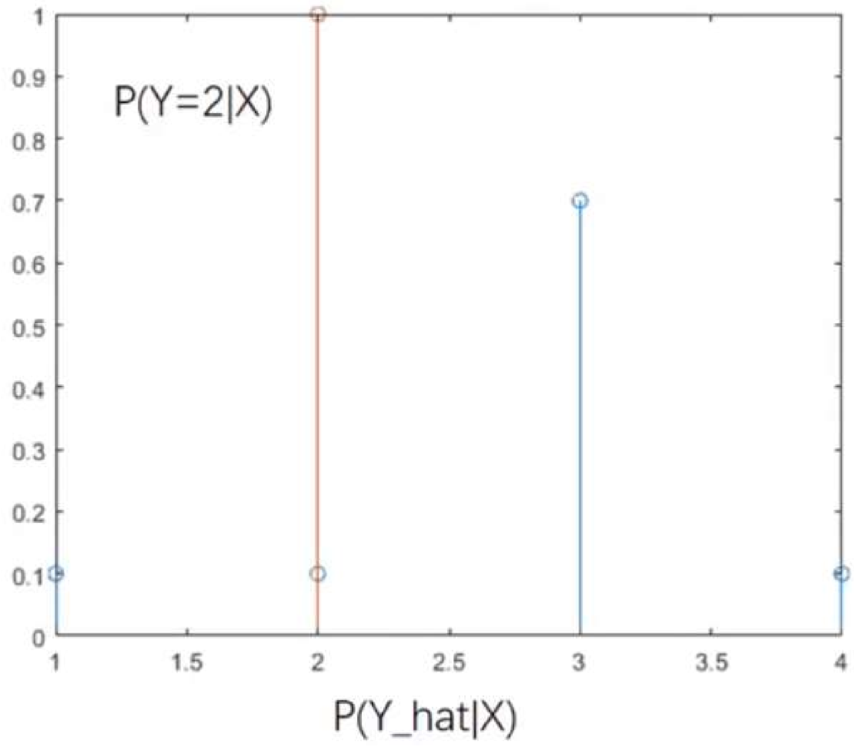
- Same as we learned before
- Adjusted-r-square (over test set)
- Response plot
- Residual plot

数据科学入门2.2：简单的分类

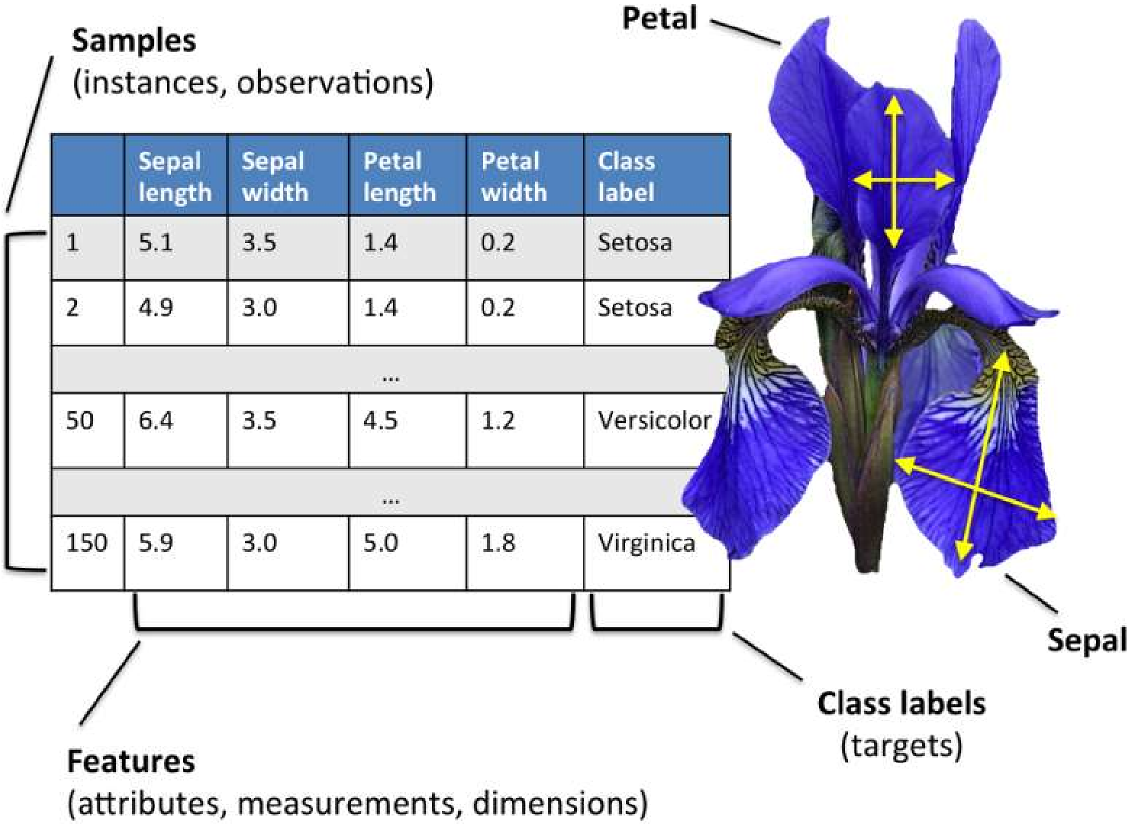
Introduction to Data Science Part 2.2: Classification

◻ Classification

- Classification is supervised learning
- Maximum likelihood is used
- Under the framework maximum likelihood, the error between two probability distributions is measured using cross-entropy.



◻ Case 2 - Iris classification



	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

◻ Evaluate a classification model

■ Confusion matrix

- true positive (TP)
- true negative (TN)
- false positive (FP)
- false negative (FN)

		Actual class	
		Cat	Dog
Predicted class	Cat	**5**	2
	Dog	3	**3**

		Actual class	
		Cat	Non-cat
Predicted class	Cat	5 True Positives	2 False Positives
	Non-cat	3 False Negatives	3 True Negatives

- **True positive rate (TPR)** or sensitivity, recall, hit rate
- **False negative rate (FNR)** or miss rate or

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = 1 - \mathrm{FNR}$$

- **False positive rate (FPR)** or fall-out or
- **True negative rate (TNR)** or specificity, selectivity

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{N}} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} = 1 - \mathrm{TNR}$$

balanced accuracy (BA)

$$\mathrm{BA} = \frac{TPR + TNR}{2}$$

数据科学入门2.3：

分类模型得选择

Introduction to Data Science

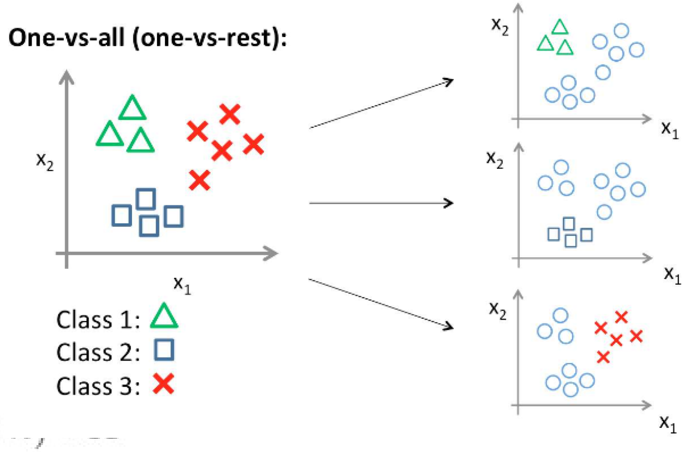
Part 2.3: Select classifiers

□ classifier的基本输出

■ Binary classifiers

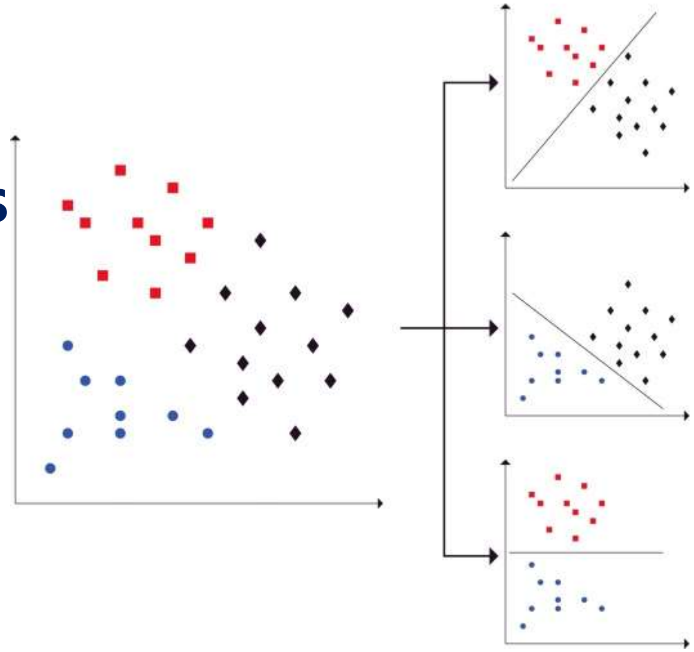
- Logistics regression
- SVM， Y>0,Y<0

$$\ln\left(\frac{P-Class1}{P-Class2}\right) = B1 + B(2: \ldots$$



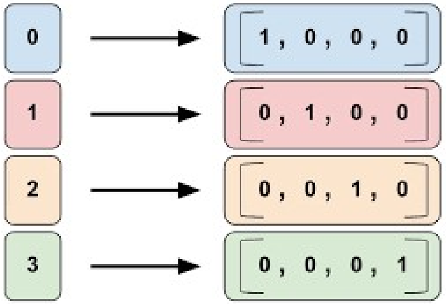
■ Binary classifiers as multi-class classifiers

- one-vs-rest, need n classifiers
- one-vs-one, need C(n,2) classifiers



□ classifier的基本输出

■ One-hot encoding



$$class = argmax(\hat{Y})$$

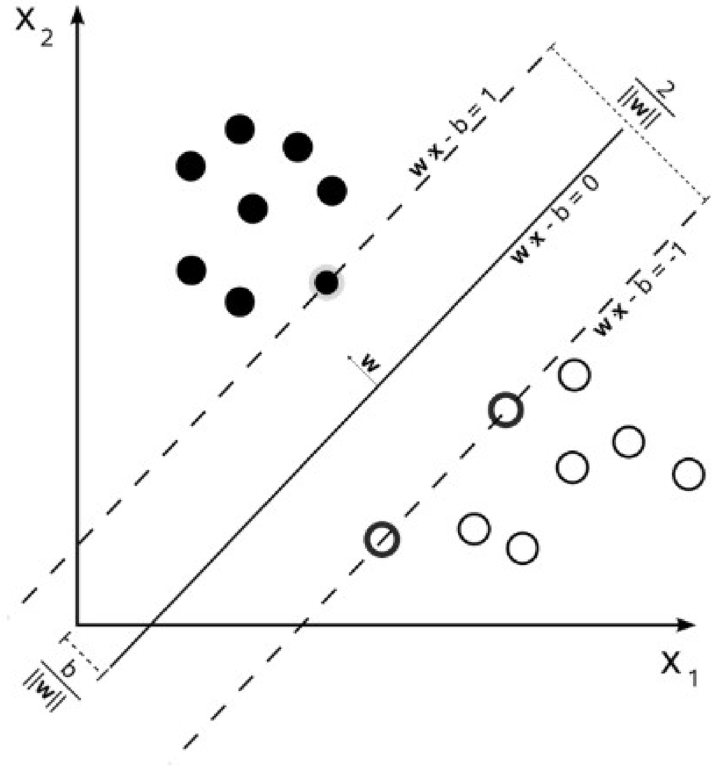
argmax ([0.1， 0.05， 0.8， 0.05])=2

◻ logistic regression

- You already know it
- Kinda like linear regression
- High bias
- Best for:
  - Simple data, low sample count, small feature size
  - Binary classification
  - Most features are continuous variable

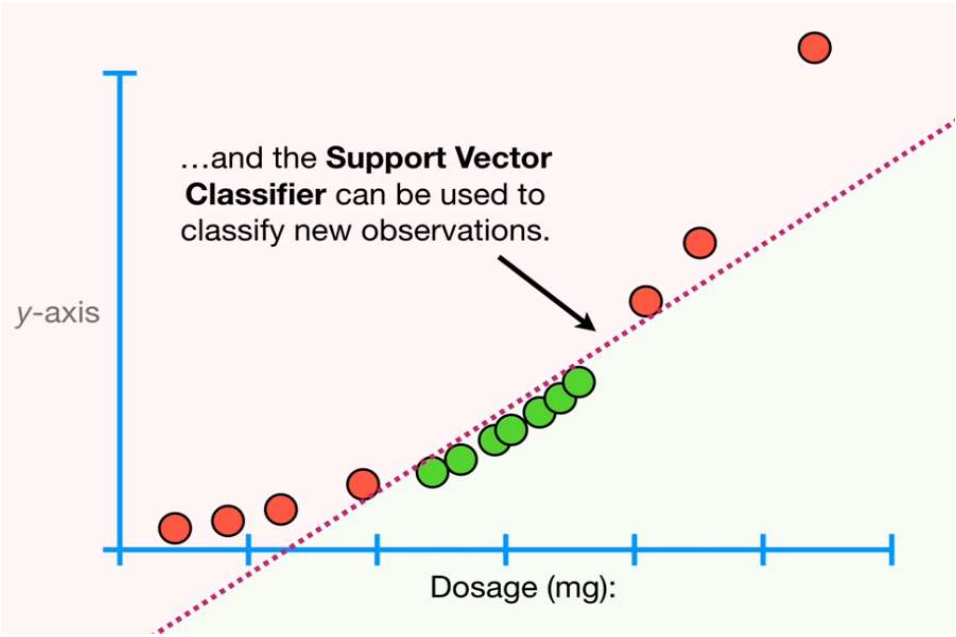
◻SVM，Support Vector Machine

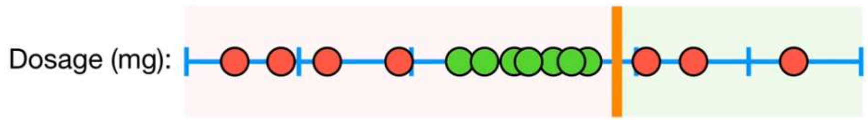
- Separate sample by maximum the margined
- Achieved by Lagrangian optimization



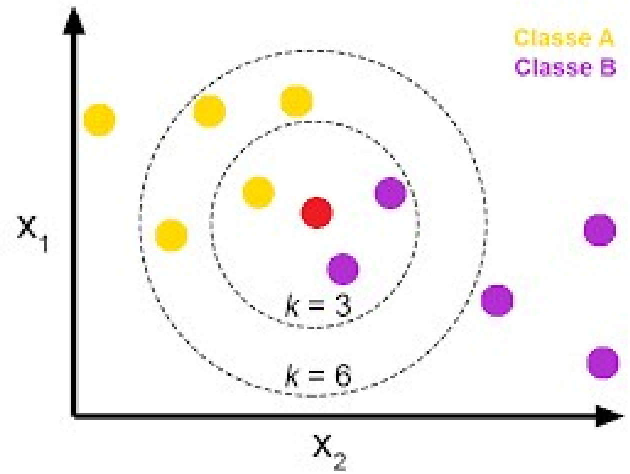
□ SVM，Support Vector Machine

- **Kernals**
- **Hyper parameters**
  - C
  - Kernals
- **Very flexible mode**
  - Low bias
  - good for many types of data
  - Good if the feature size is large





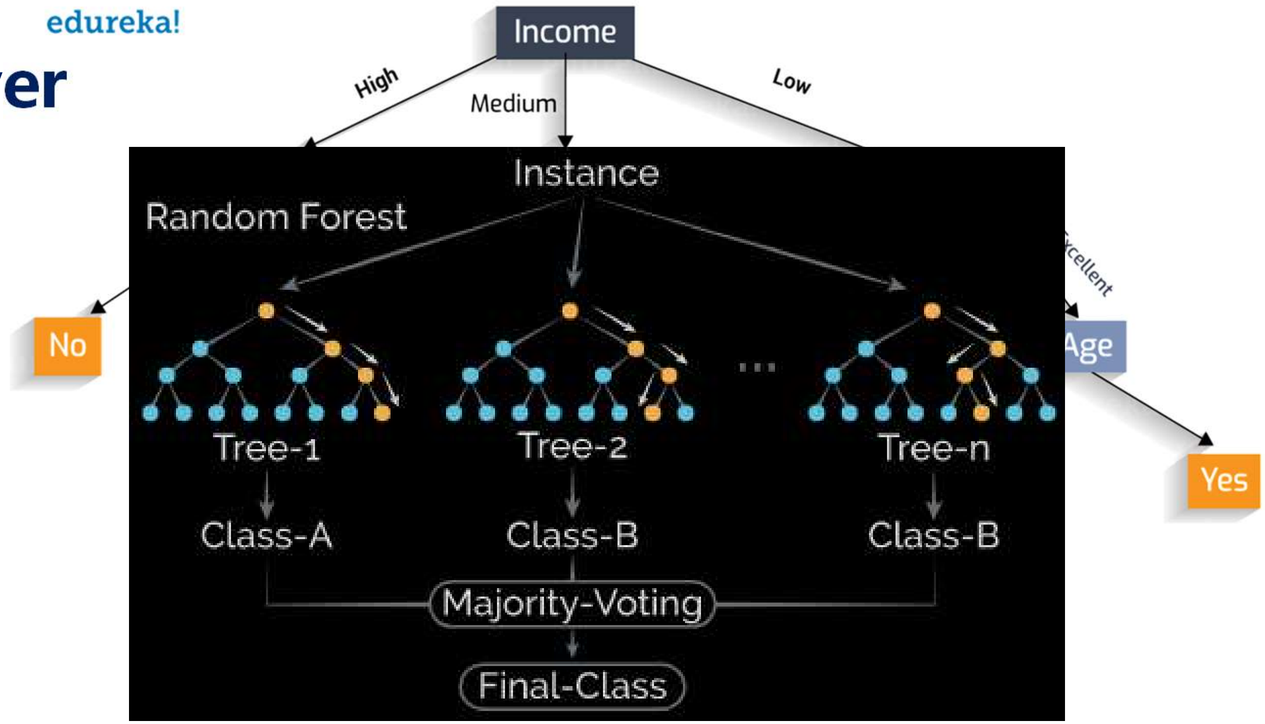
◻ K-nearest neighbors



- ■ **Count nearest neighbors**
- ■ **Larger count wins**
- ■ **Some variants use weighted vote upon distance**
- ■ **Hyper parameters**
  - K
  - Distance calculation
- ■ **High bias model**
  - Best when sample size is large but feature size is small

□ Decision Tree，ensemble of trees

- ■ **Many variants**
- ■ **Basic trees tend to over fitting easily**
- ■ **Use ensemble**
- ■ **Hyper parameters**
  - Criterion-mostly gini
  - Max depth
  - Max leaf nodes
  - min_samples_split



□ Model selection?

- Just try many models and pick the overall best one

#数据科学入门2.4：

特征的选取和特征提取

Introduction to Data Science 4

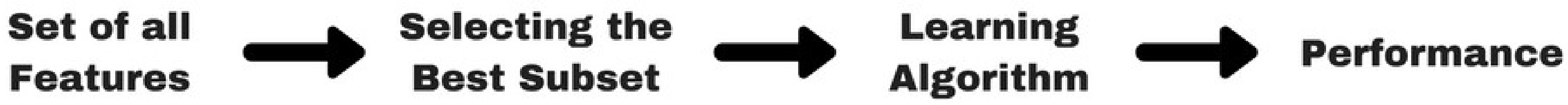
Part2.4: Simple Feature Engineering

◻ Why

- Less computational heavy, faster training
- Less over fitting, better generalization
- Use our domain knowledge to create better features
- Less noise better accuracy
- Simpler model could be interpreted

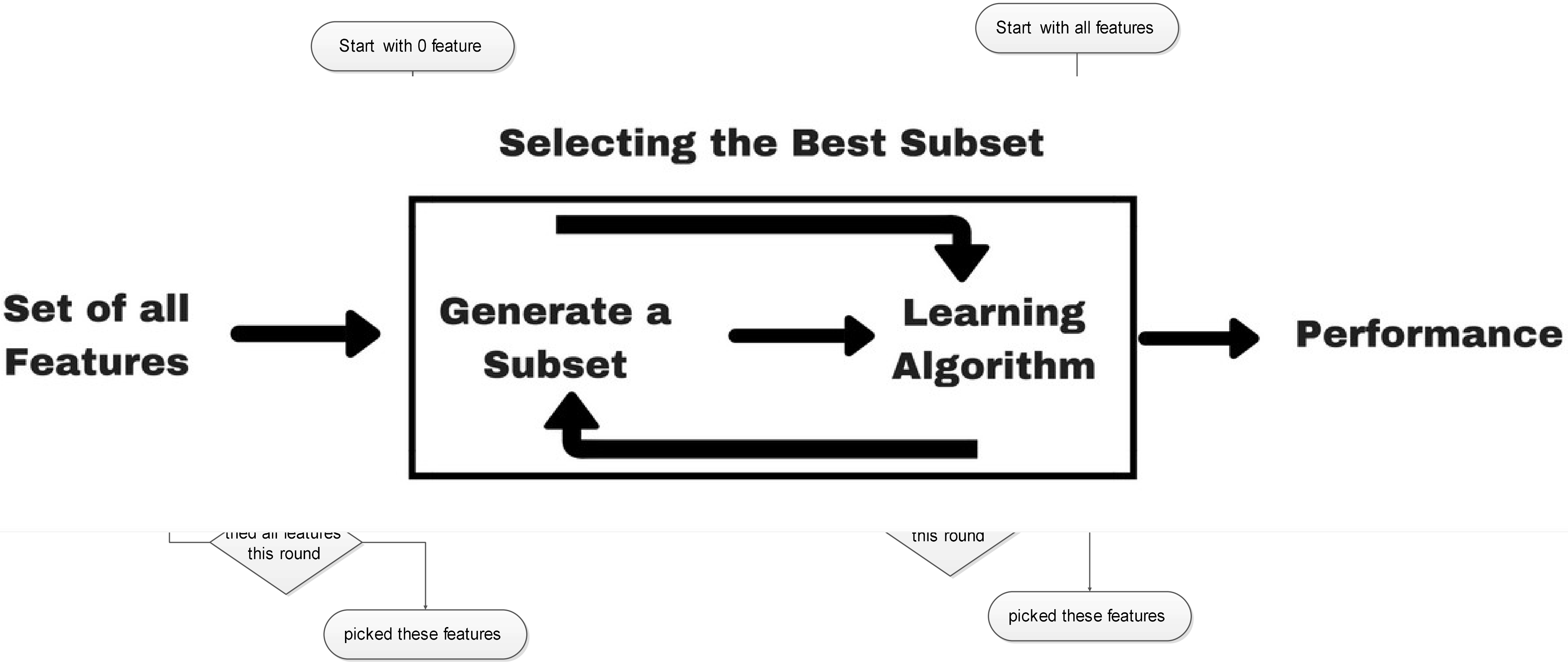
□ Feature selection

■ Filter method



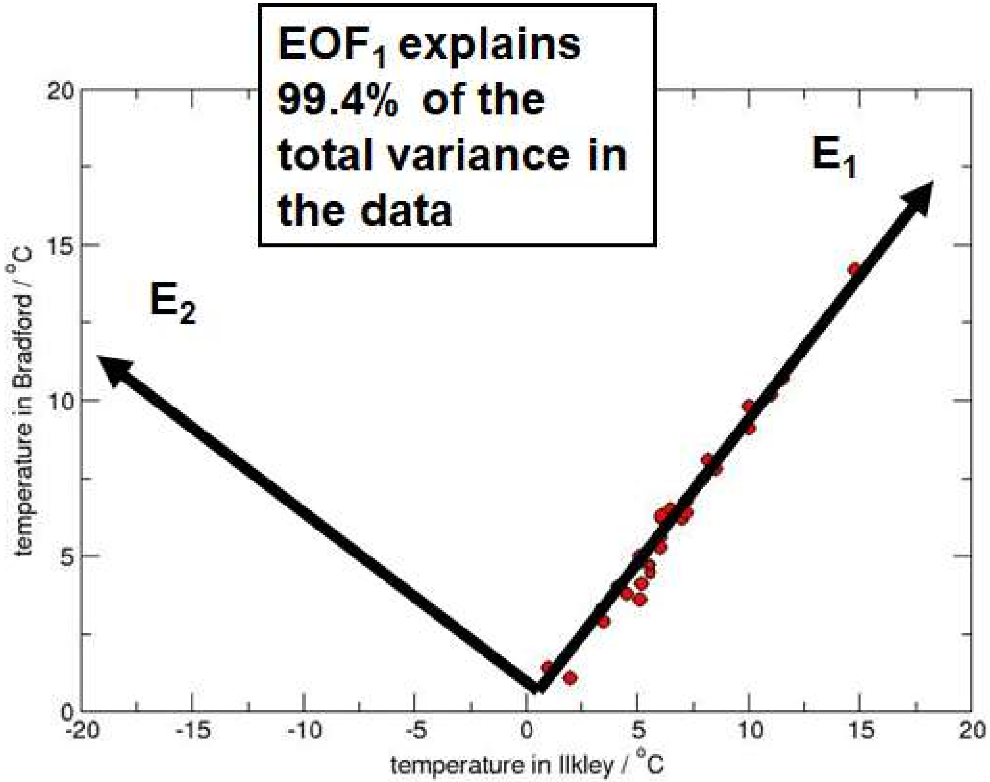
☐ Feature selection

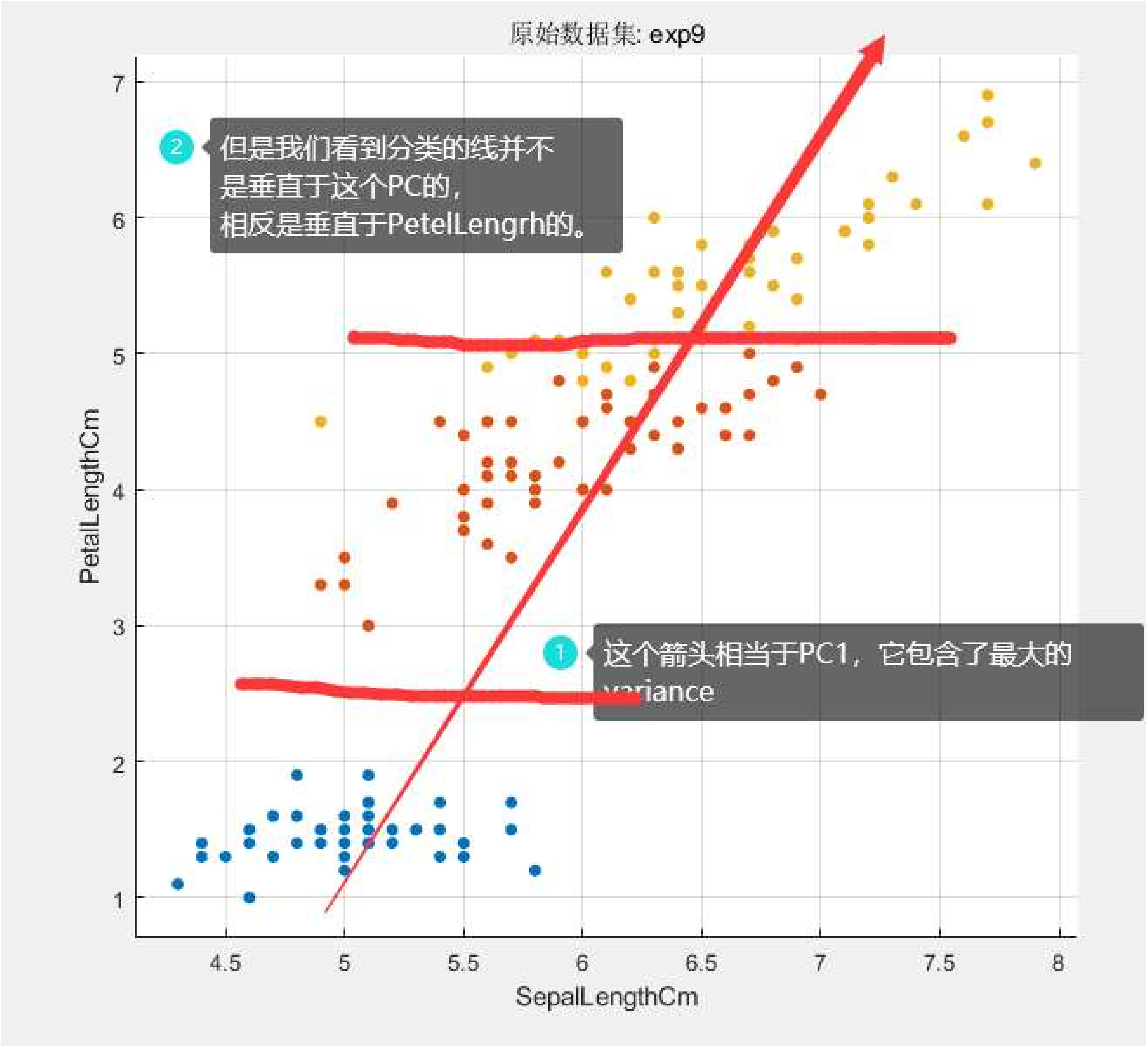
■ Wrapper methods



□ Feature extraction (feature creation)

■ PCA (Principal Component Analysis)




原始数据集: exp9
PetalLengthCm
SepalLengthCm
7
6
5
4
3
2
1
4.5
5
5.5
6
6.5
7
7.5
8
2
但是我们看到分类的线并不
是垂直于这个PC的，
相反是垂直于PetelLengrh的。
1
这个箭头相当于PC1，它包含了最大的
variance

◻ Feature extraction (feature creation)

- Use your domain knowledge
- More on this later

#数据科学入门2.5：

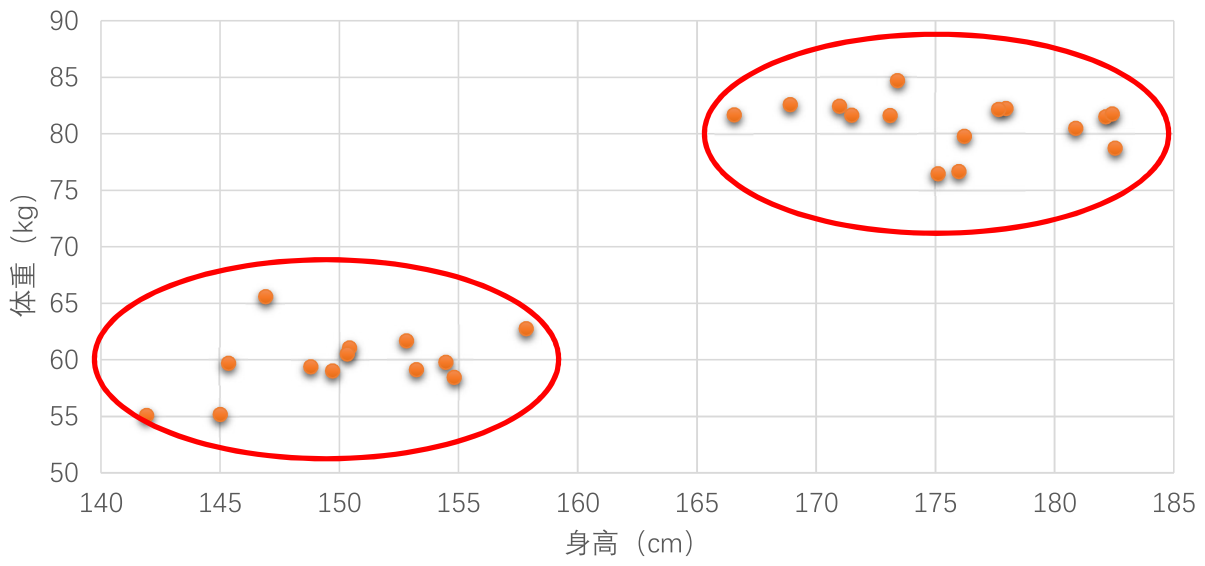
简单的聚类

Introduction to Data Science

Part2.5: Simple Clustering

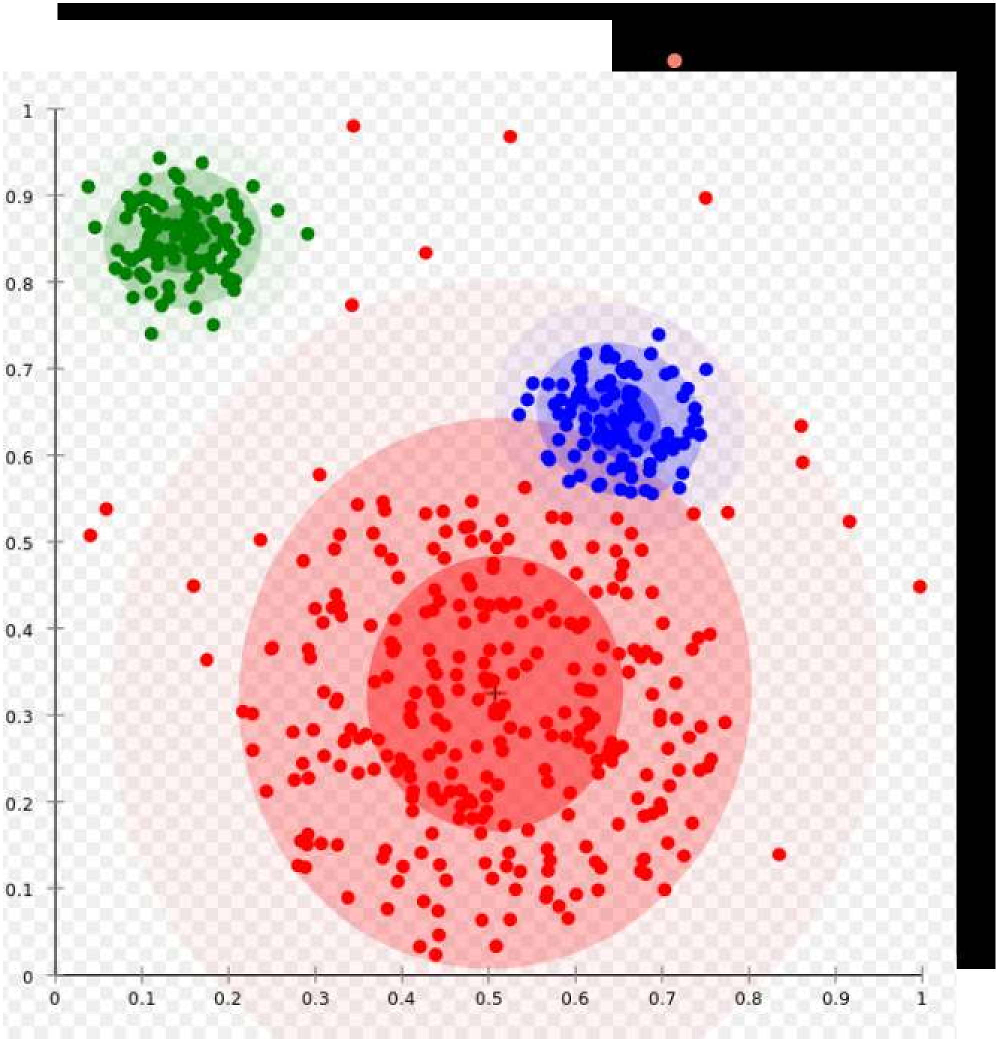
◻ What is clustering

- Clustering is unsupervised learning
- Samples in their feature space, ones that are close to each other are cluster in to one cluster.

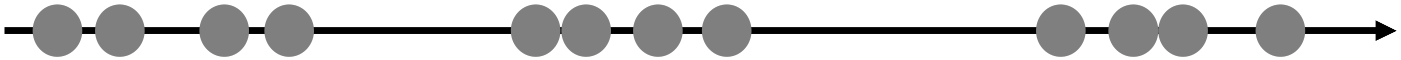


□ Types of clustering

- Connectivity based
- Density based
- Distribution based
- Centroid based

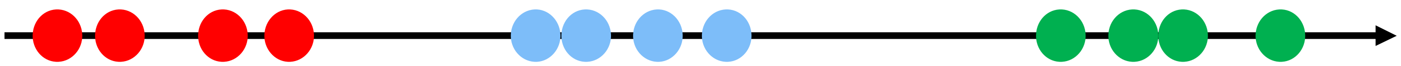


□ K-Means原理



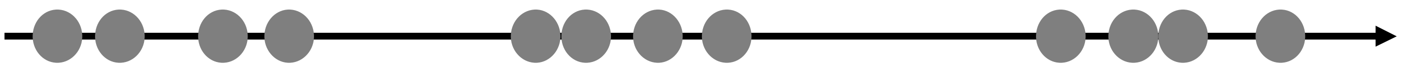
K-Means原理

- 这个我一眼就看出来了
- 怎么让程序完成这个工作呢?



K-Means原理

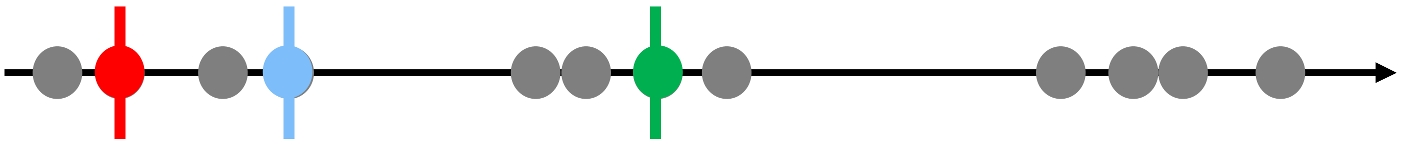
- 第1步：确定要分为几类，K=?
  - 我们分为3类，K=3



□ K-Means原理

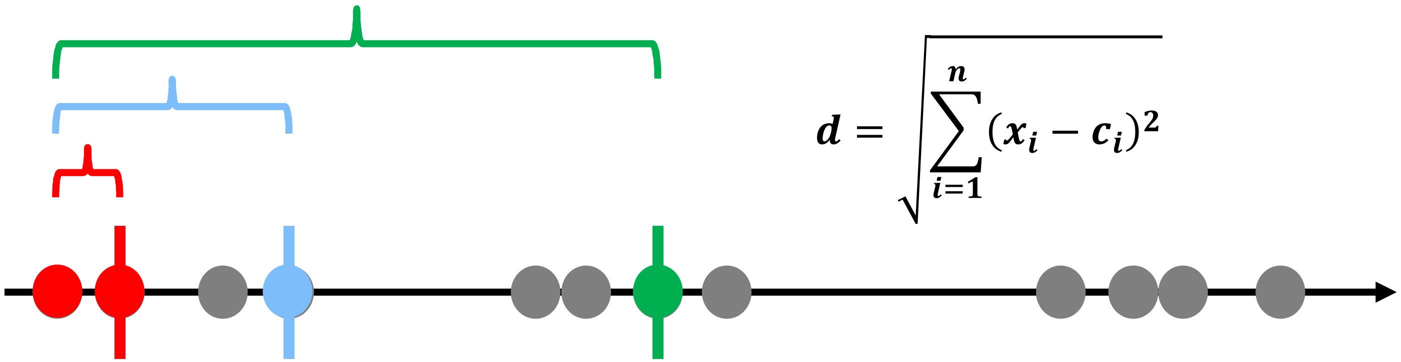
■ 第2步：随便选择3个类型的中心（重心）

- 我们就随便选3个点，让他们作为3个类型的中心
- 这3个点就分别属于对应的类型



□ K-Means原理

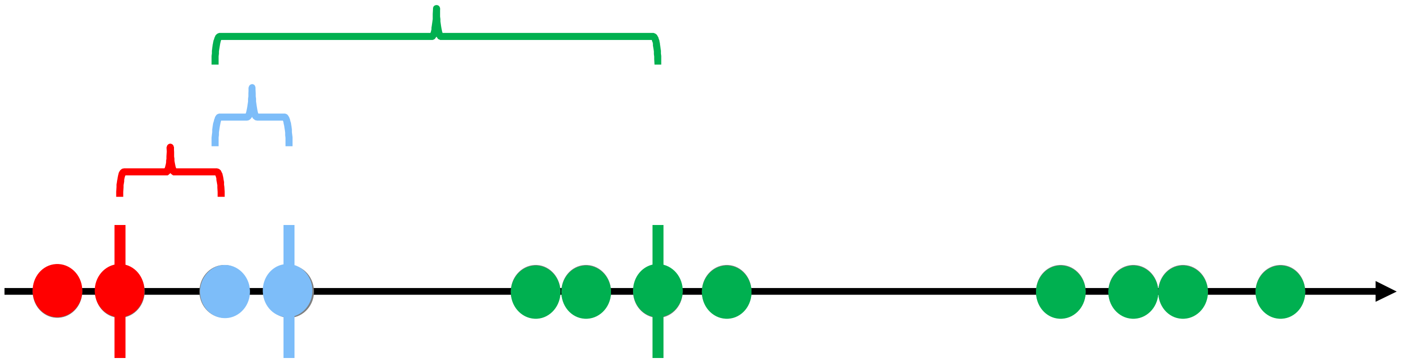
- 第3步：计算第一个点到三个中心的距离
- 第4步：把第一个点分配个距离最近的类型



$$d = \sqrt{\sum_{i=1}^{n} (x_i - c_i)^2}$$

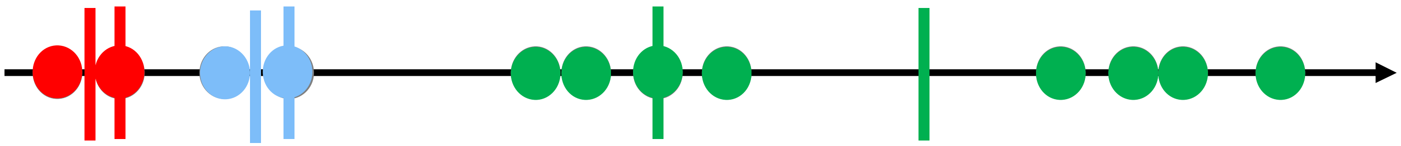
□ K-Means原理

- 第5步：对剩下的点做3，4步



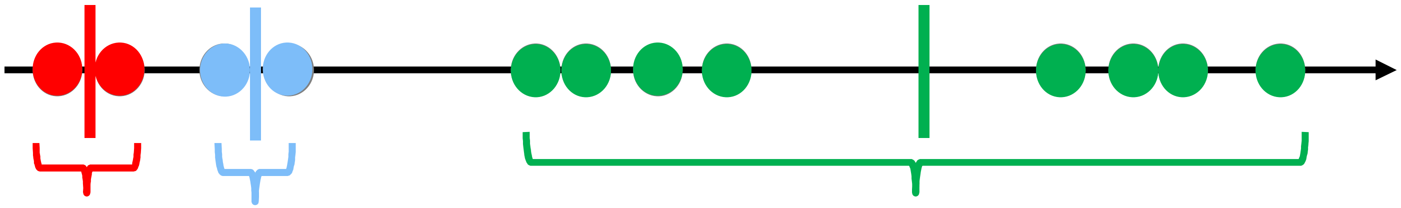
□ K-Means原理

- **第6步：重新计算每个几个的中心（Mean）**
- 第7步：3-6步，直到每个点所属的类型不在变化为止



□ K-Means原理

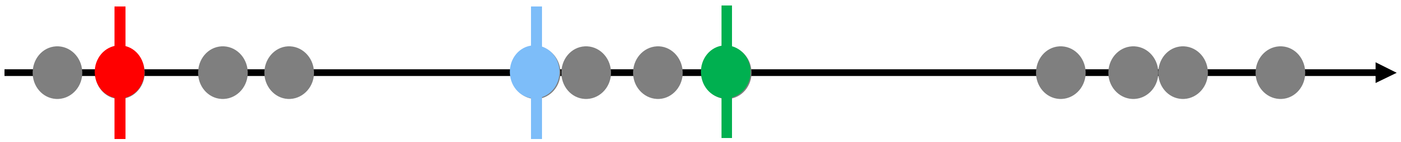
- **分好了！但是结果好像不太好 ☹**
- 怎么评价我们的结果好坏?



$$SE = \sum_{k=0}^{Cluster} \sum_{i=0}^{ALlPoints} D_{ki}$$

□ K-Means原理

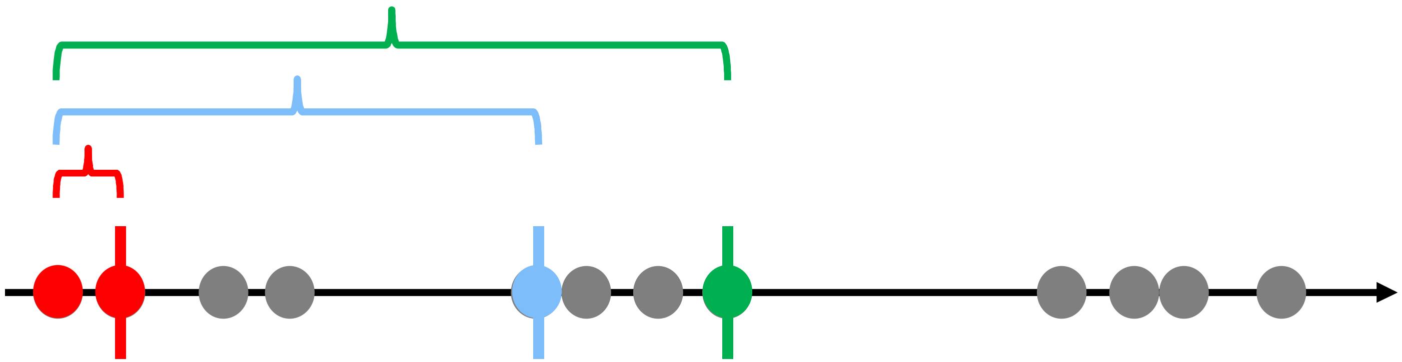
- 重新开始，第1步，K=3
- 第2步：随便选择3个类型的中心（重心）
- 我们就随便选3个点，让他们作为3个类型的中心
- 这3个点就分别属于对应的类型



□ K-Means原理

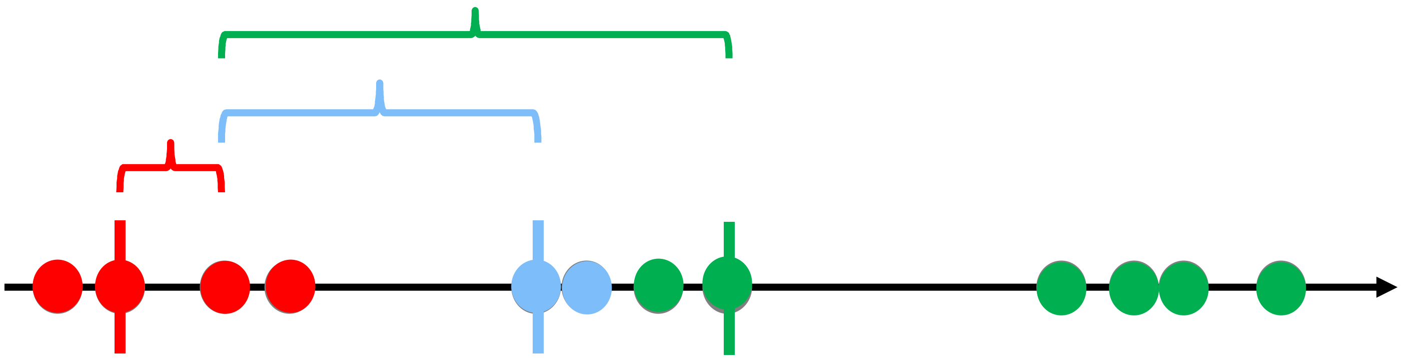
- 第3步：计算第一个点到三个中心的距离

- 第4步：把第一个点分配个距离最近的类型



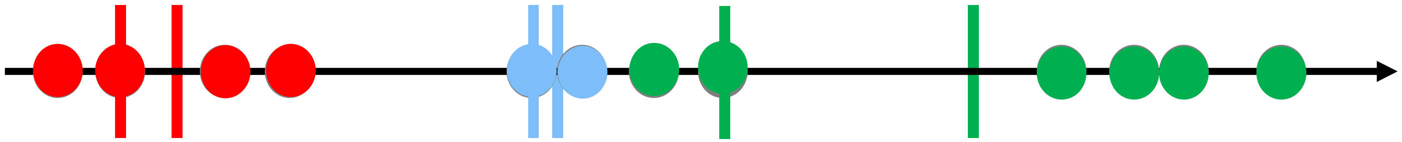
□ K-Means原理

■ 第5步：对剩下的点做3，4步



□ K-Means原理

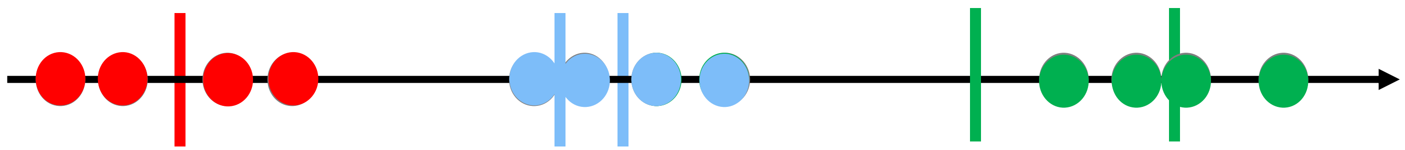
- **第6步：重新计算每个几个的中心（Mean）**
- 第7步：3-6步，直到每个点所属的类型不在变化为止



□ K-Means原理

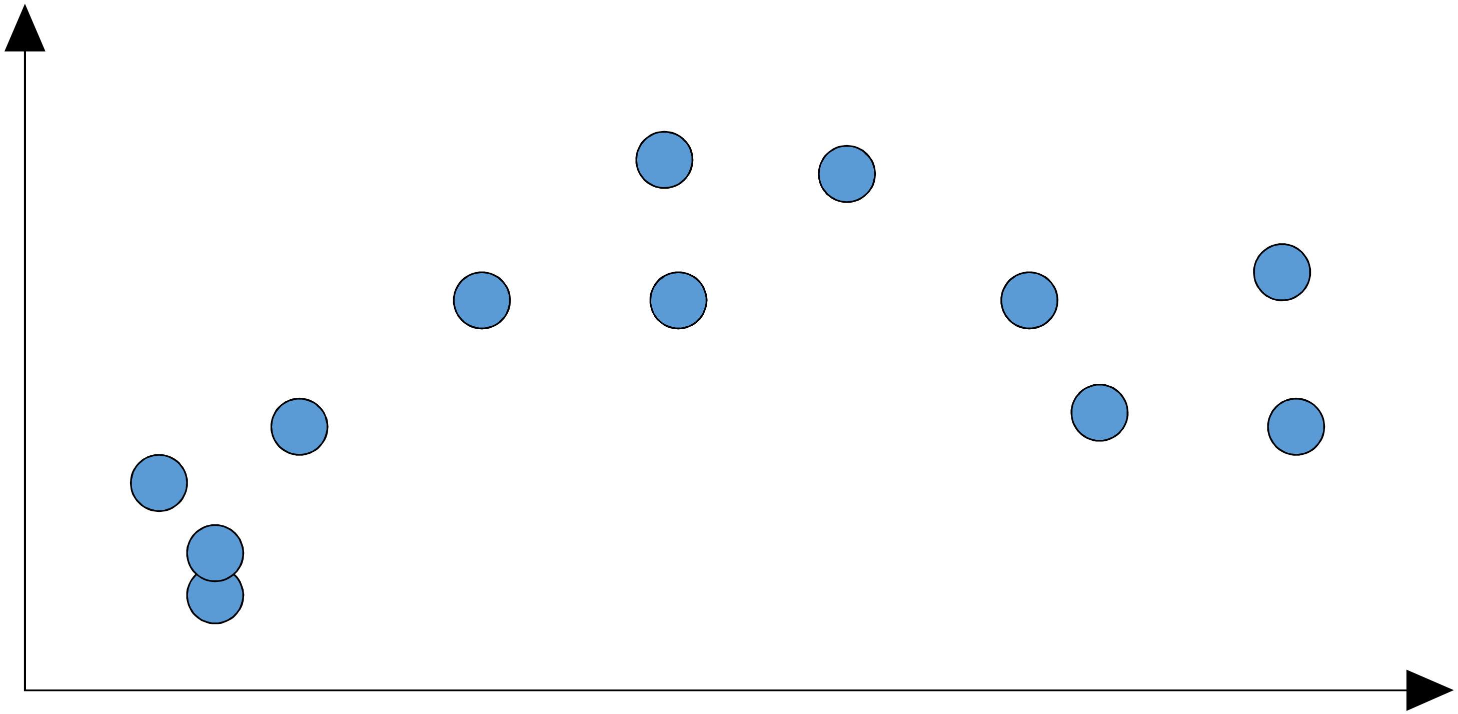
- **第6步：重新计算每个几个的中心（Mean）**
- 第7步：3-6步，直到每个点所属的类型不在变化为止

$$SE = \sum_{k=0}^{Cluster} \sum_{i=0}^{ALlPoints} D_{ki}$$



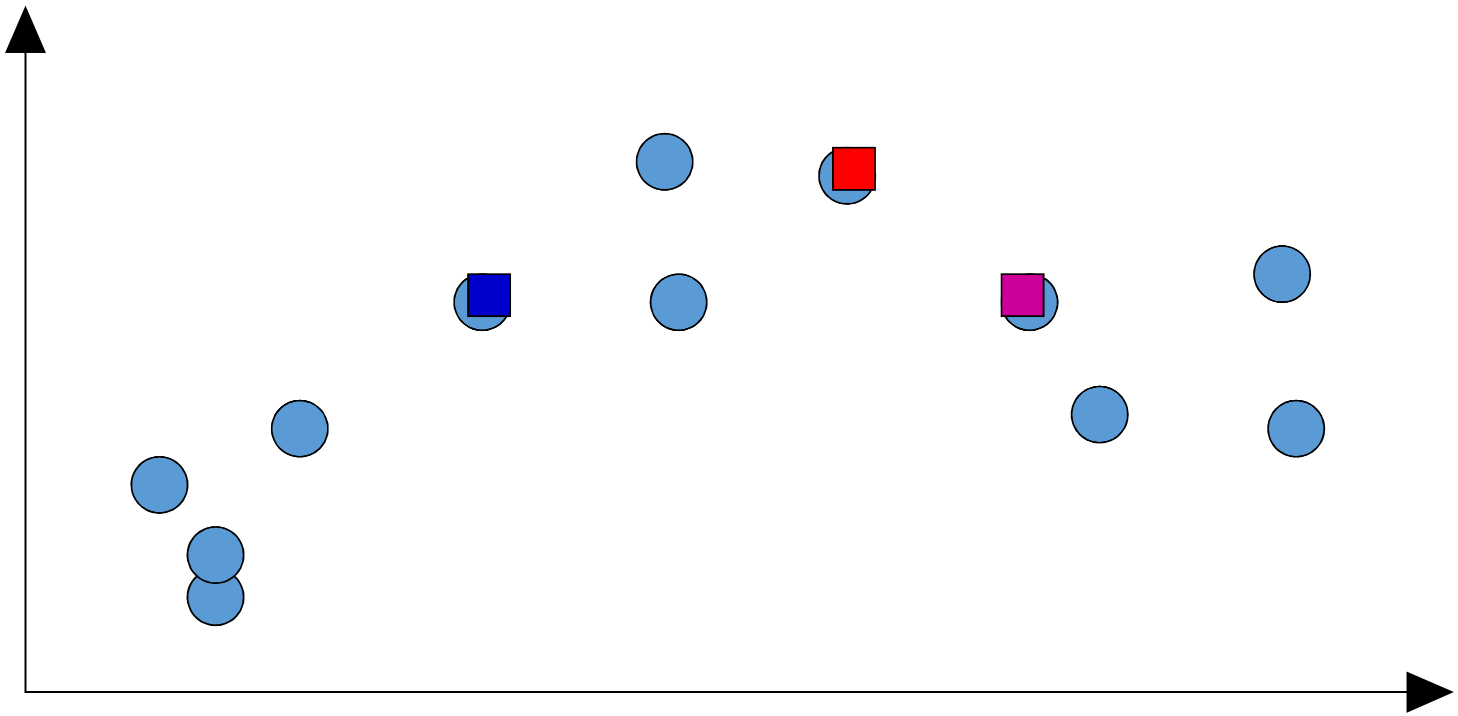
电脑并不知道这个是最好的结果，于是他再重复以上步骤多次，选取最好的结果

□ K-Means原理——2维数据



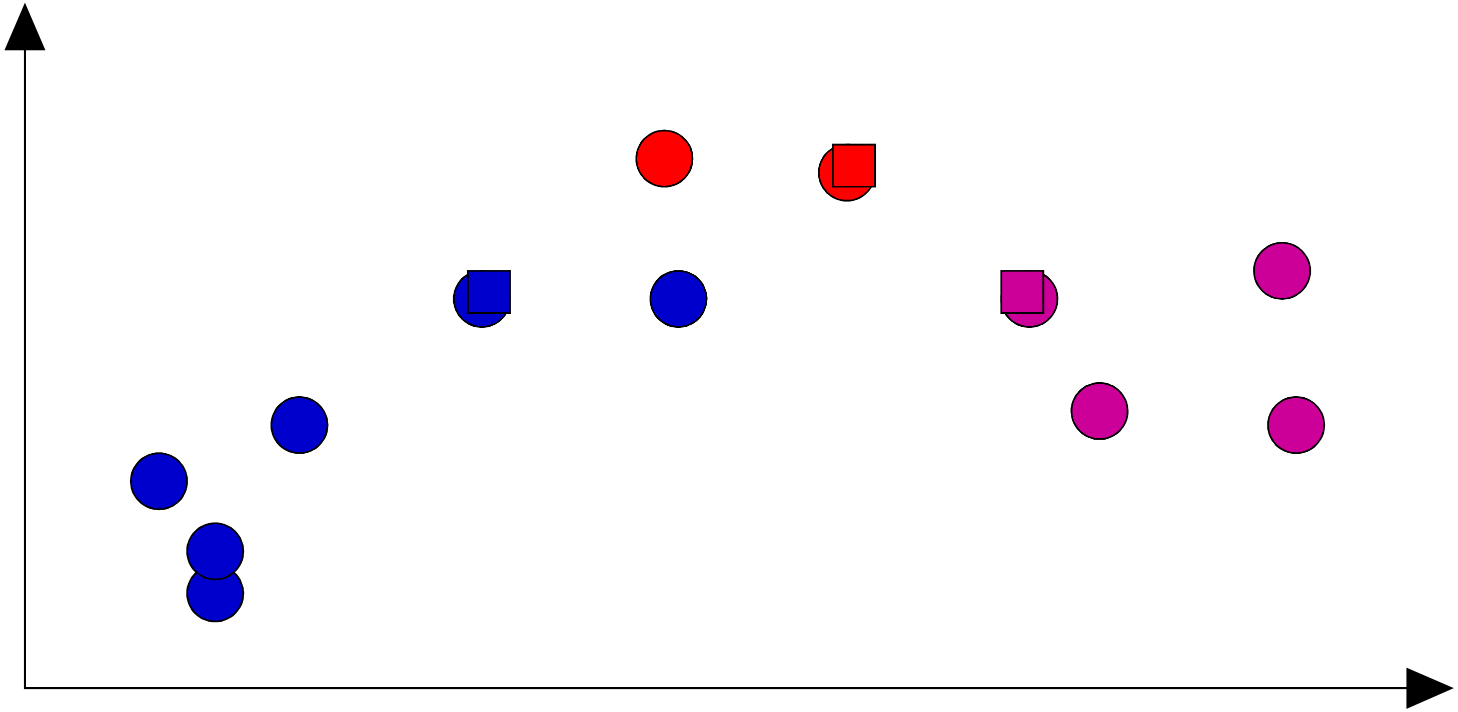
□ K-Means原理——2维数据

- 第1步：确定要分为几类，K=?
- 第2步：随便选择K个类型的中心



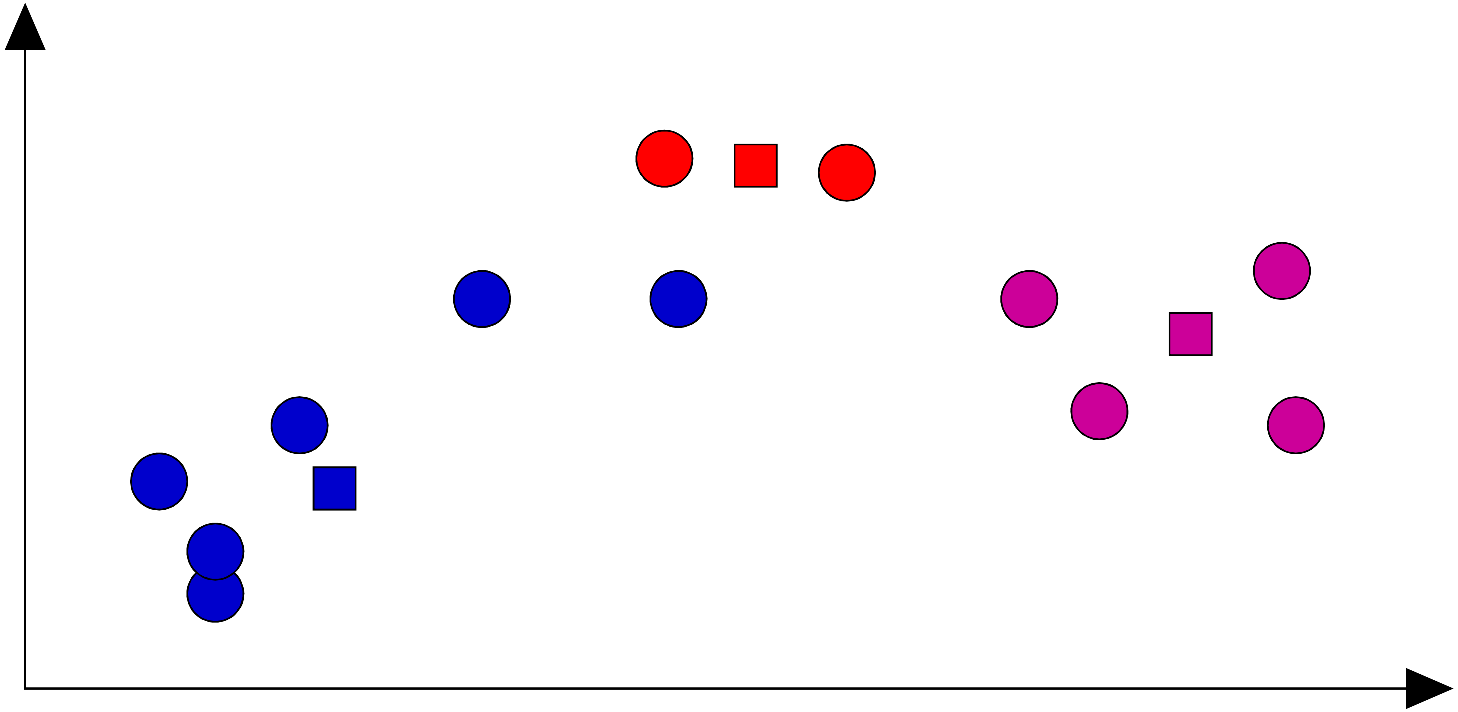
□ K-Means原理——2维数据

- 第3步：计算每个点到三个中心的距离
- 第4步：把第一个点分配个距离最近的类型



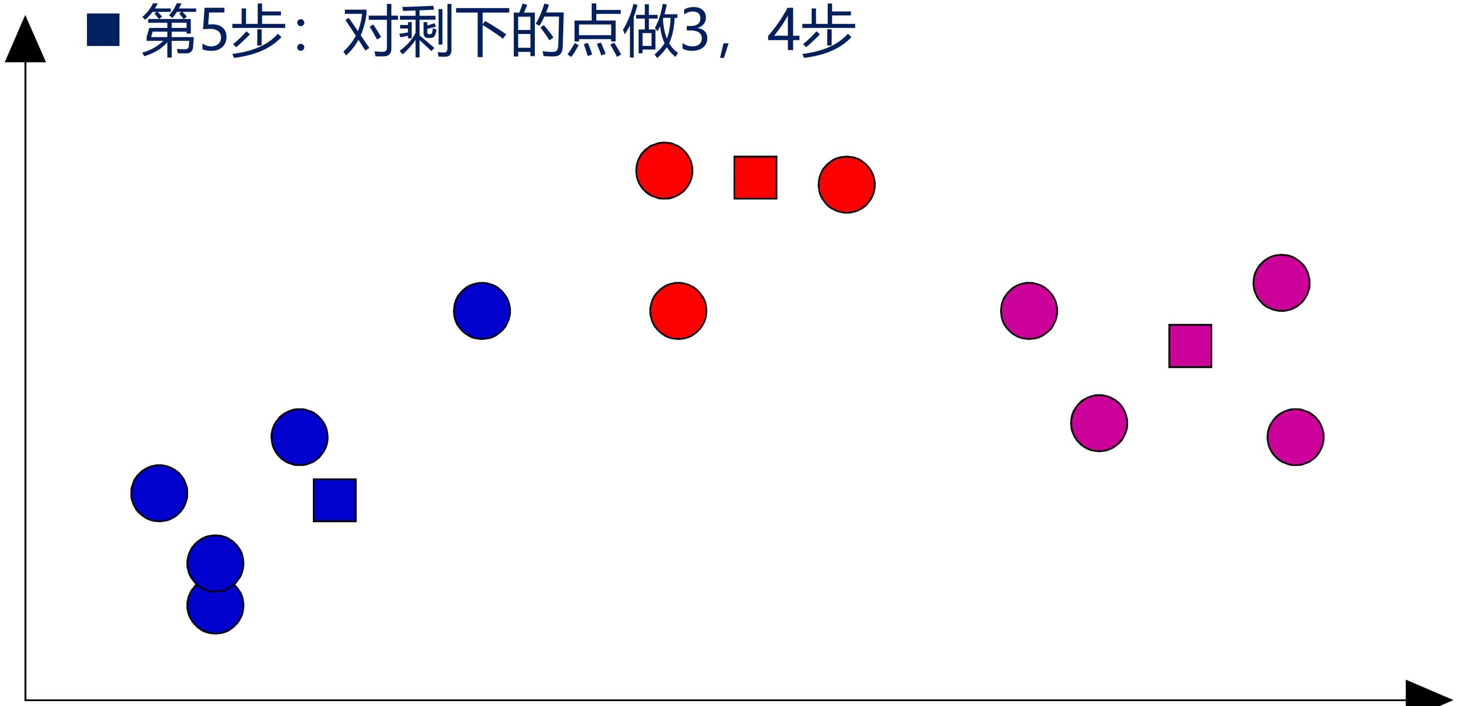
□ K-Means原理——2维数据

■ 第5步：重新计算每个类型的中心



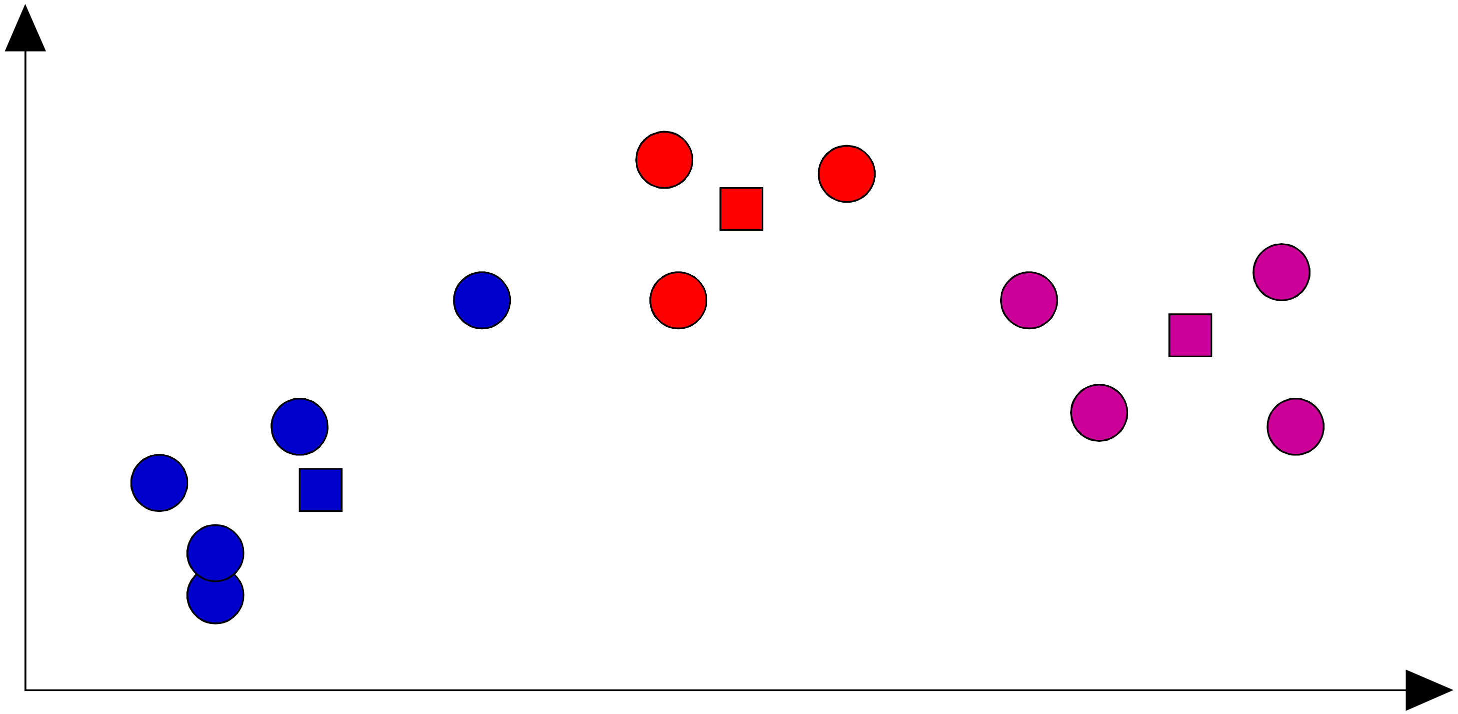
□ K-Means原理——2维数据

- 第7步：3-6步，直到每个点所属的类型不在变化为止
- 第3步：计算每个点到三个中心的距离
- 第4步：把第一个点分配个距离最近的类型
- 第5步：对剩下的点做3，4步



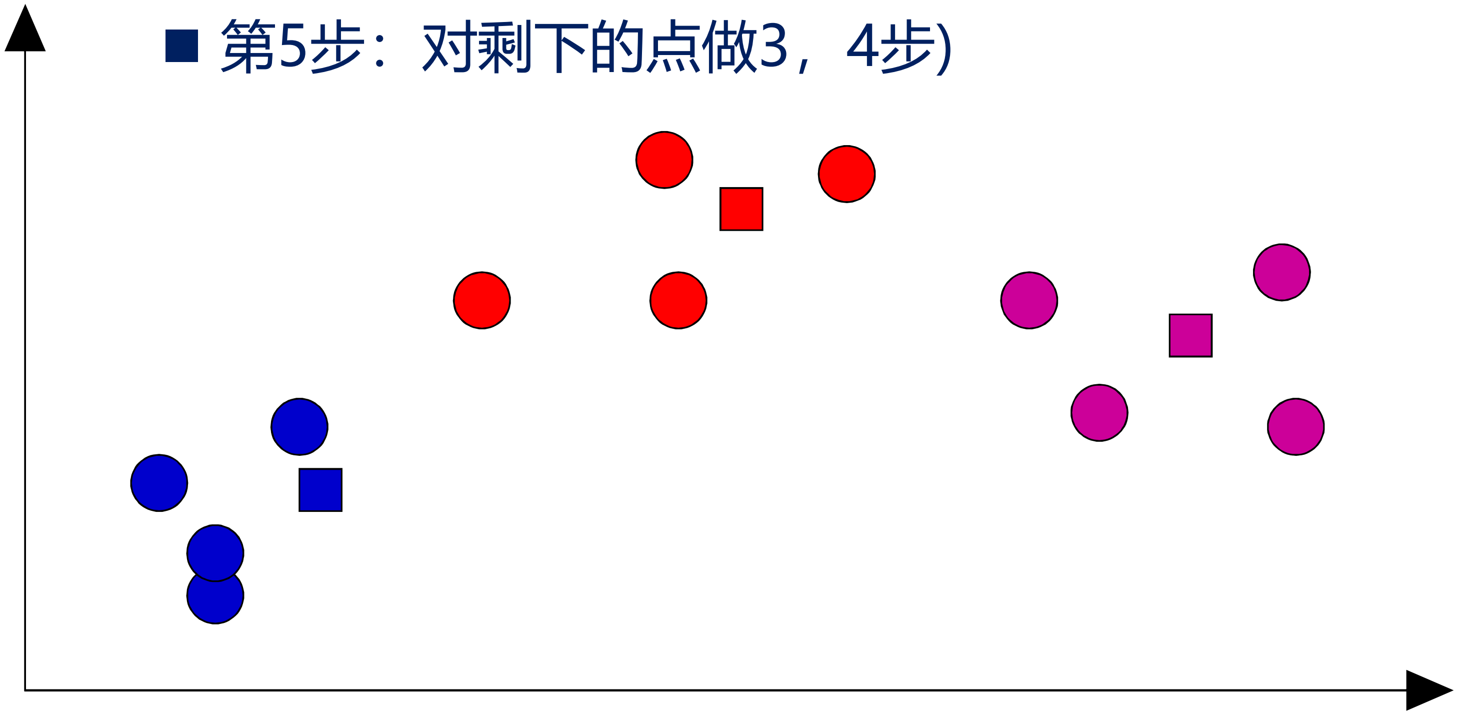
□ K-Means原理——2维数据

- 第7步：3-6步，直到每个点所属的类型不在变化为止
- (第6步：重新计算每个类型的中心)



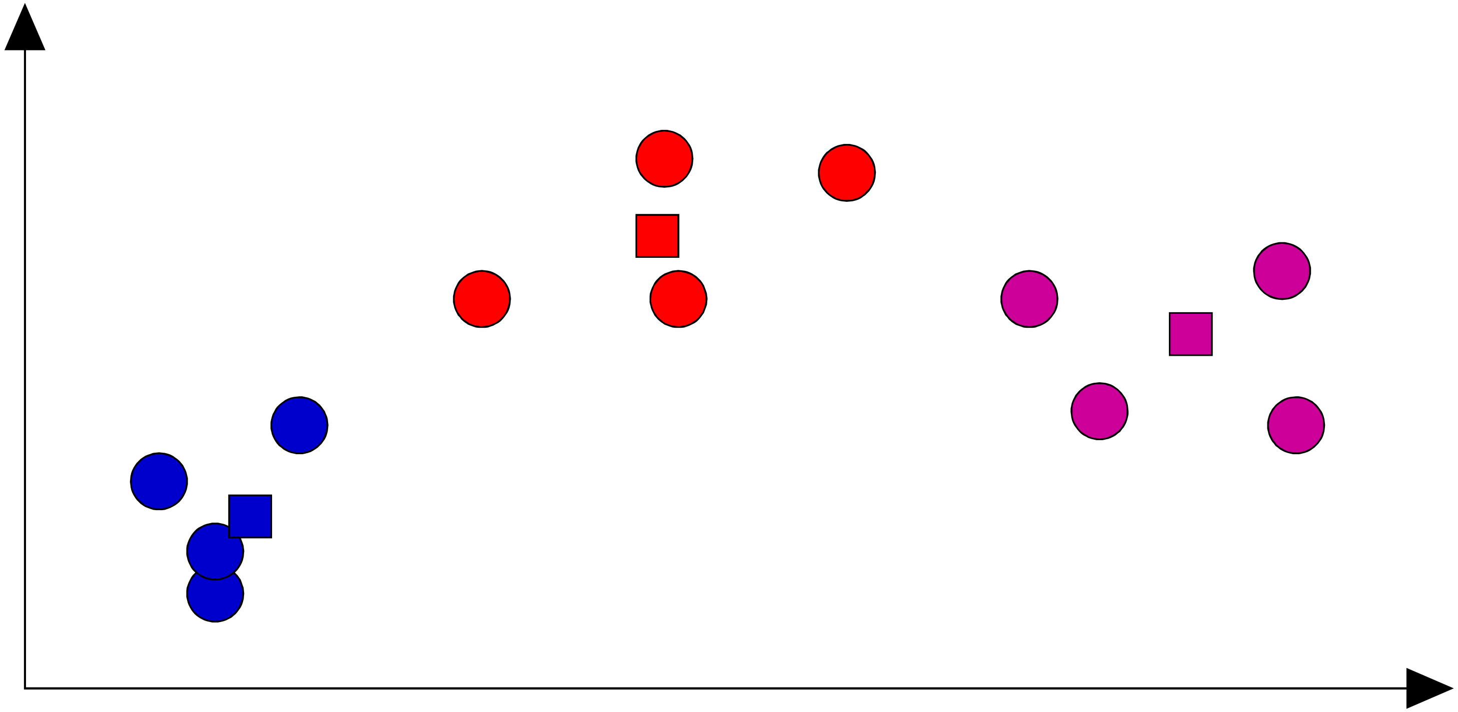
◻ K-Means原理——2维数据

- 第7步：3-6步，直到每个点所属的类型不在变化为止
- (第3步：计算每个点到三个中心的距离
- 第4步：把第一个点分配个距离最近的类型
- 第5步：对剩下的点做3，4步)

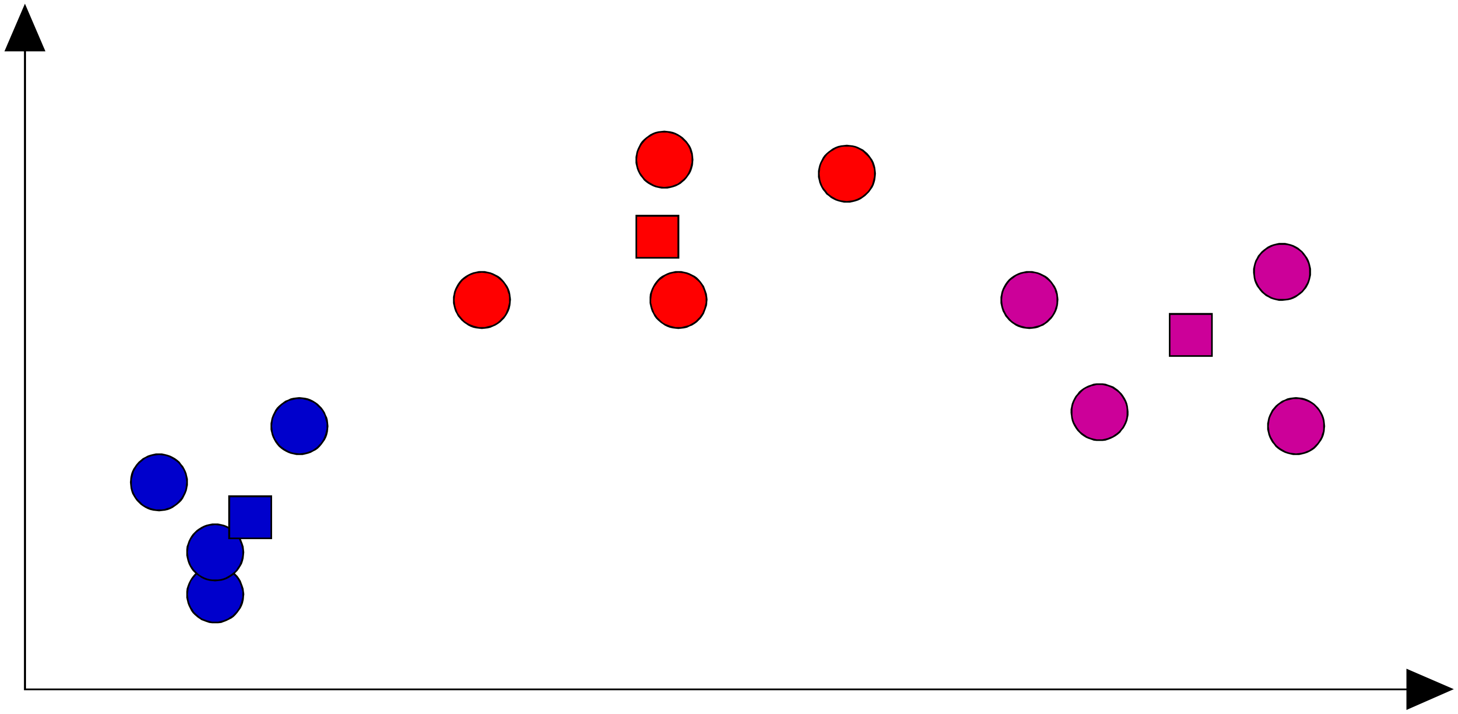


□ K-Means原理——2维数据

- 第7步：3-6步，直到每个点所属的类型不在变化为止
- (第6步：重新计算每个类型的中心)

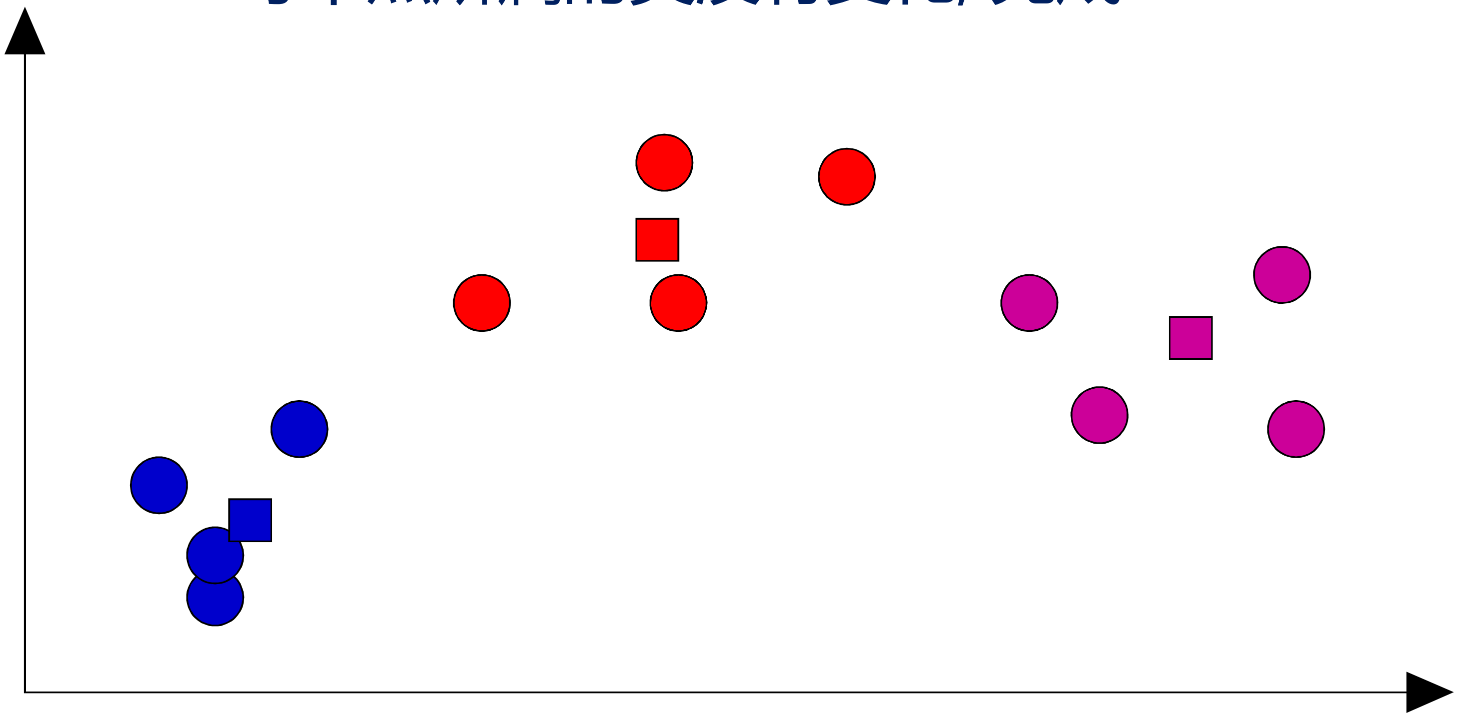


□ K-Means原理——2维数据



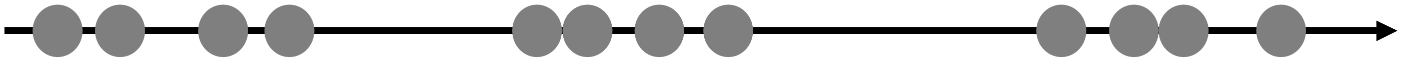
□ K-Means原理——2维数据

- 每个点所属的类没有变化，完成!



□ K-Means原理

■ **问题：那怎么确定K呢?**

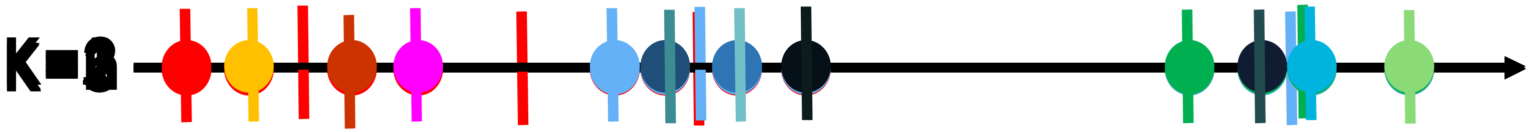


■ **方法一：你希望分成几类就让K=几。**

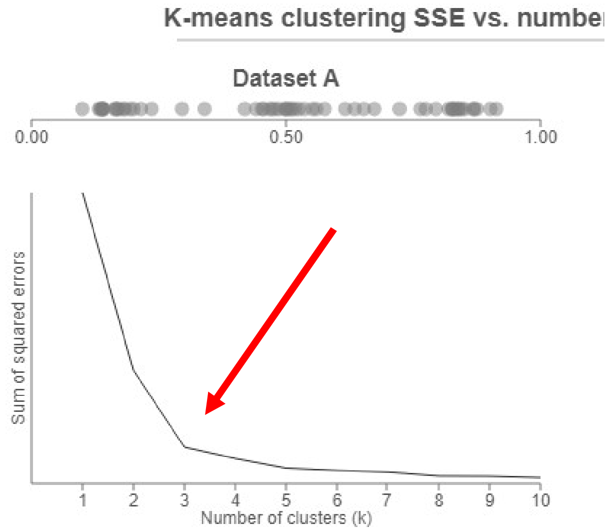
- 大人，小孩
- 男生，女生
- 好，中，差

□ K-Means原理

■ Elbow Method



$$SE = \sum_{k=0}^{Cluster} \sum_{i=0}^{ALlPoints} D_{ki}$$



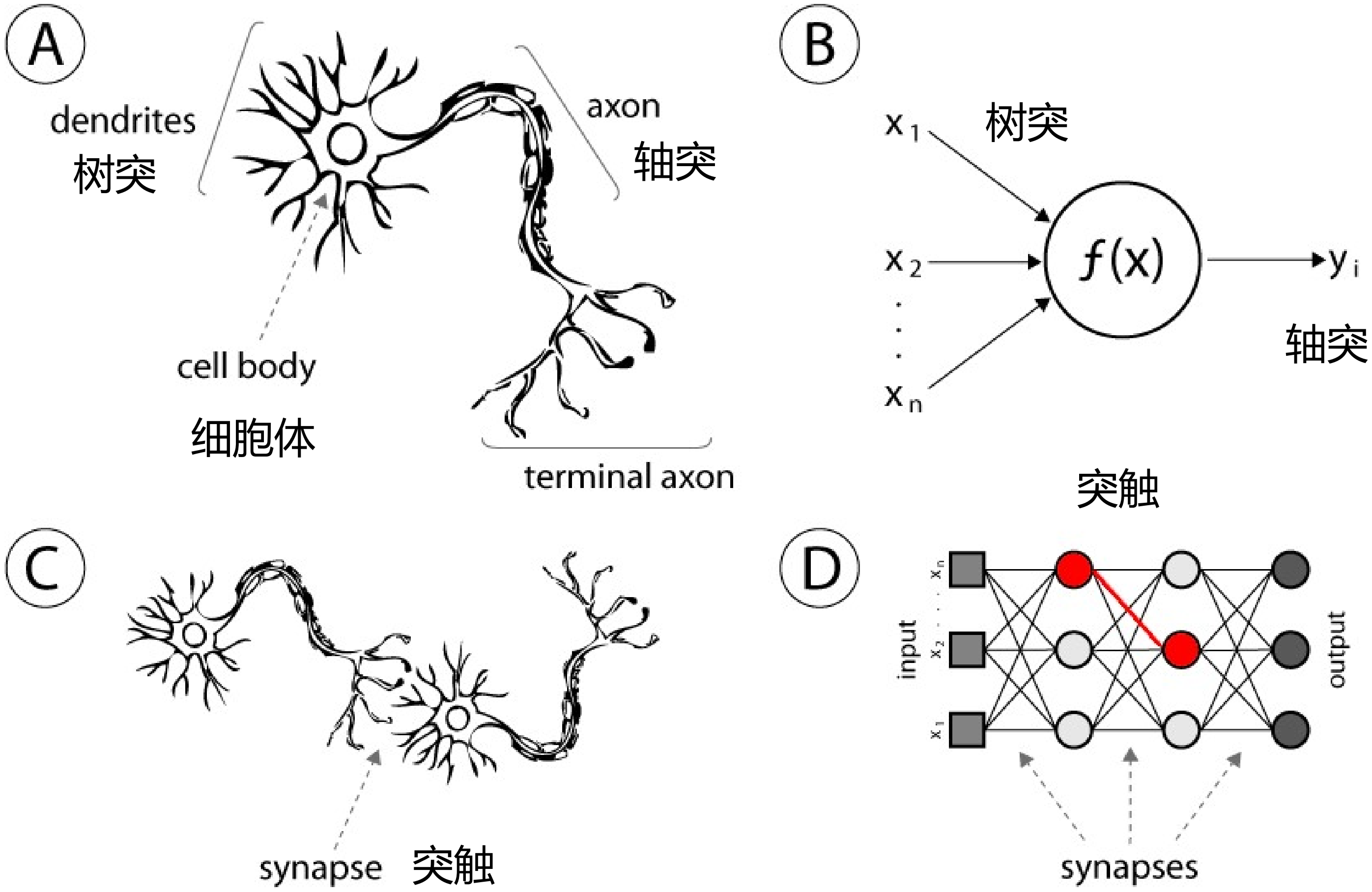
#数据科学入门2.5.9：

人工神经网络入门

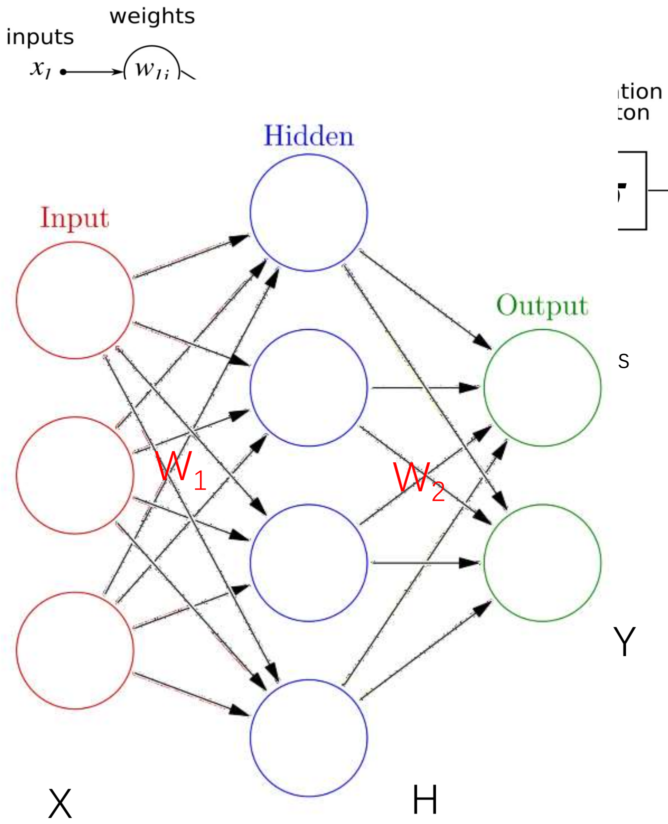
Introduction to Data Science

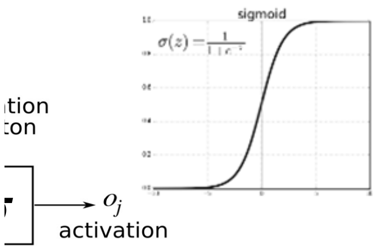
Part2.5.9: Demystify the artificial neural network

□ Artificial neural network



□ How it works





$$o = \sigma(\sum_{i=1}^{n} x_i * w_{ij} + b)$$

$$o = \sigma(WX + b)$$

$$H = \sigma(W_1 X + B_1)$$

W_1 [4*3]
X [3*1]
B_1 [4*1]
H [4*1]

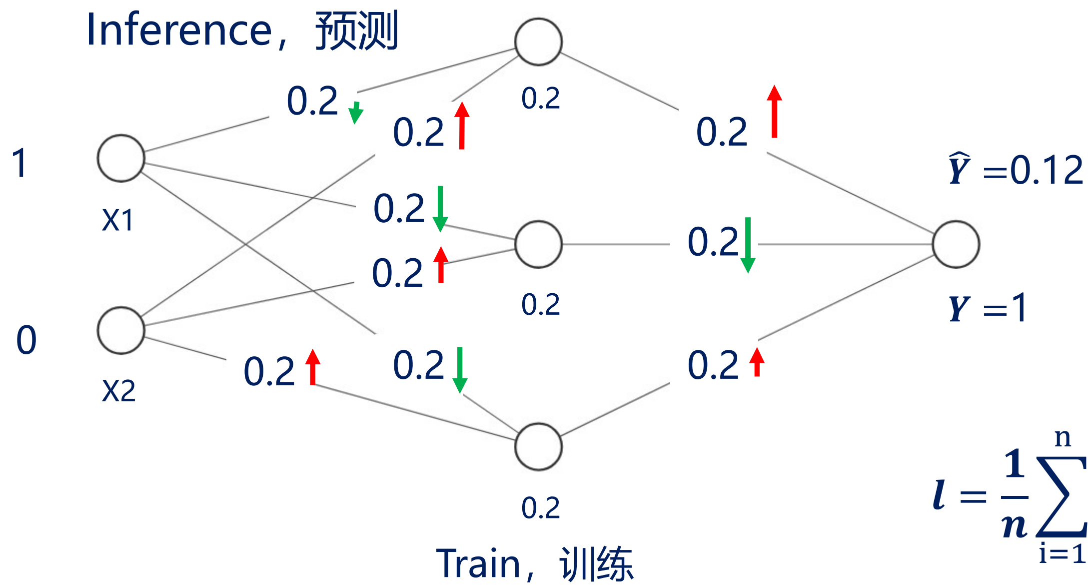
$$Y = \sigma(W_2 H + B_2)$$

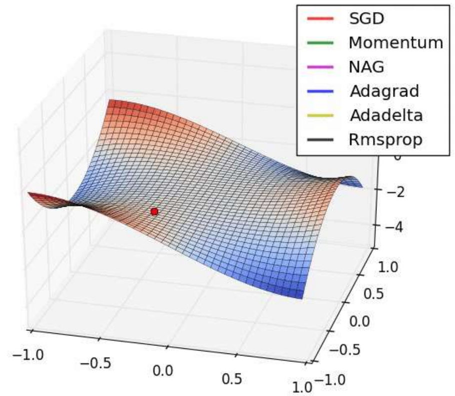
W_2 [2*4]
H [4*1]
B_2 [2*1]
Y [2*1]

$$\hat{Y} = f(X, W, B)$$

4+2=6个神经元 3*4+4*2=20个weight，4+2=6个bias，26个可优化参数

□ How it is trained?





$$l = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$ Loss Function

$$Min(l)$$

X1	X2	Y
1	0	1
0	1	1
1	1	0
0	0	0

$$Y = X_1 \oplus X_2$$

$$\theta = [w_1, w_2, , b_1, b_2, , ,]$$

$$\nabla_\theta l(\theta) = [\frac{\partial l}{w_1}, \frac{\partial l}{w_2}, , , , , \frac{\partial l}{b_1}, \frac{\partial l}{b_2}, , , , ,]$$

☐ Matlab sallow neural network

- Simple regression
- Time series
- Just for fun, not serious application

#数据科学入门2.6：
一点点深度学习

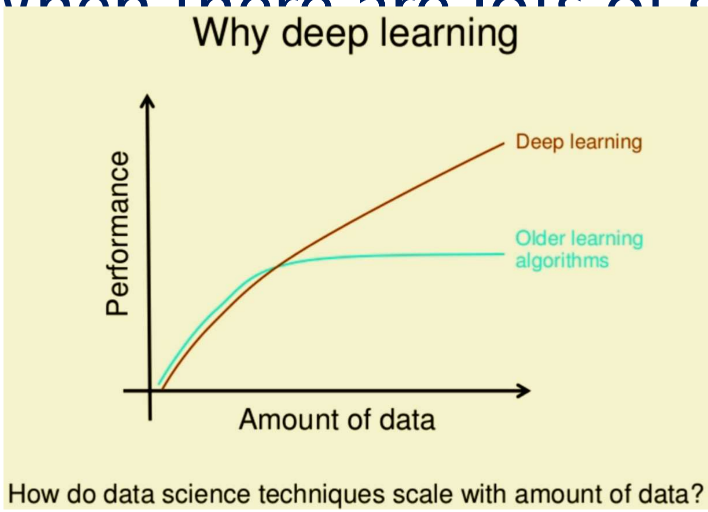
Introduction to Data Science
Part2.6: a glance of deep learning

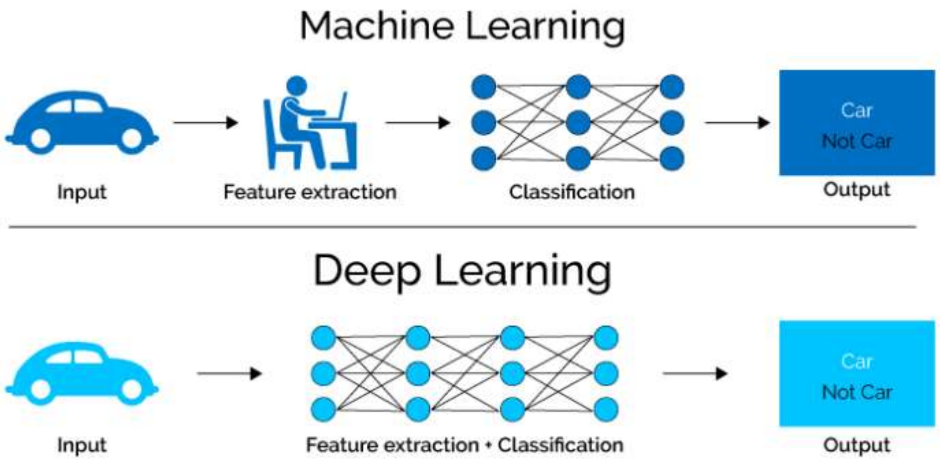
□ Before we start

- Why we want deep learning
- First we look a traditional learning on complex data

Why deep learning (below is only partly true)

- Remember the feature engineering we did in previous class?
- When it’s hard to extract meaningful low dimension features.
- When there are lots of features
- When there are lots of samples





◻ Assignment

- Redo the examples in this class
- Try Titanic survival data using models and evaluation methods learned in this class

If you want to know more