

Python

NGUYEN
Hong Thinh

Introduction

ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

Machine Learning Basic

NGUYEN Hong Thinh

FET-UET-VNU

Ngày 11 tháng 4 năm 2020

Machine Learning

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

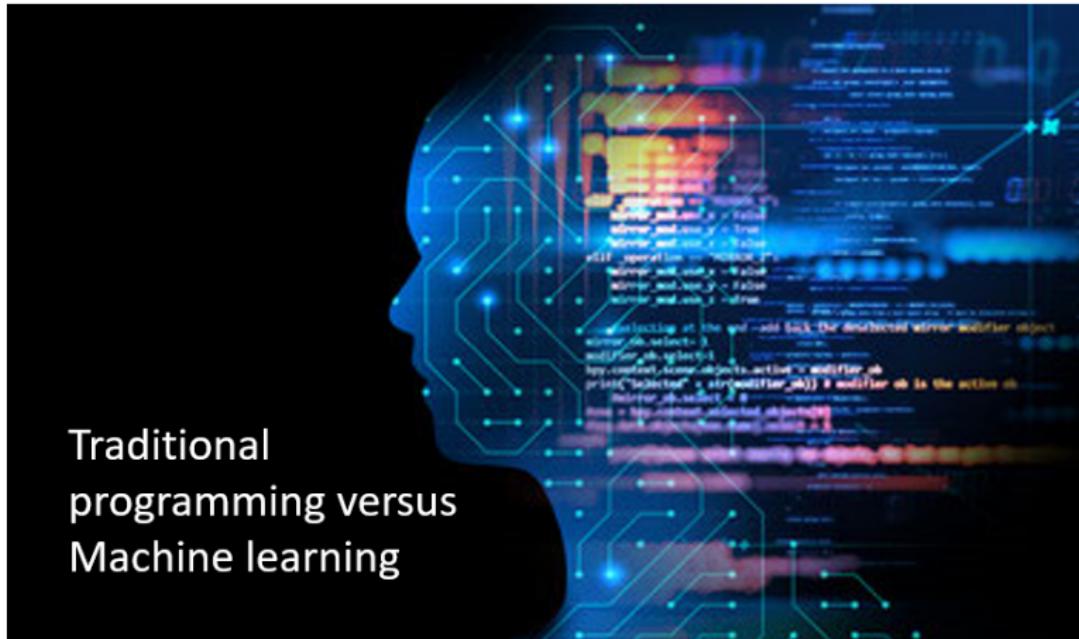
Other methods

Classification

KNN

SVM

Traditional
programming versus
Machine learning



Machine Learning

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

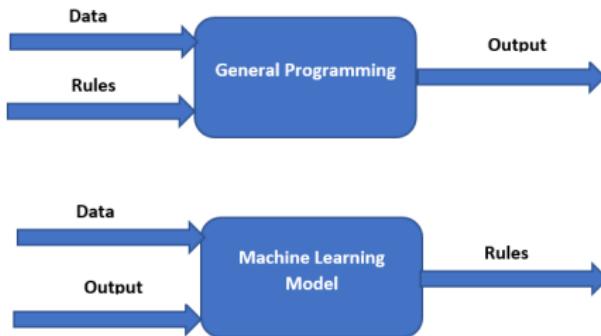
Other methods

Classification

KNN

SVM

Sự khác biệt giữa Programming và Machine Learning:



- **Programming:** Các kỹ sư dựa trên kiến thức của mình, xây dựng chương trình (rules) để từ dữ liệu vào, có được dữ liệu ra như đã có. Sau đó, đem các rules đó áp dụng cho dữ liệu mới tương tự.
- **Machine Learning:** Từ dữ liệu vào và dữ liệu ra (đã có), đưa qua máy tính. Máy tính sẽ tự "học" và tìm ra model (rules) phù hợp nhất, áp dụng cho các trường hợp sau

Machine Learning

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

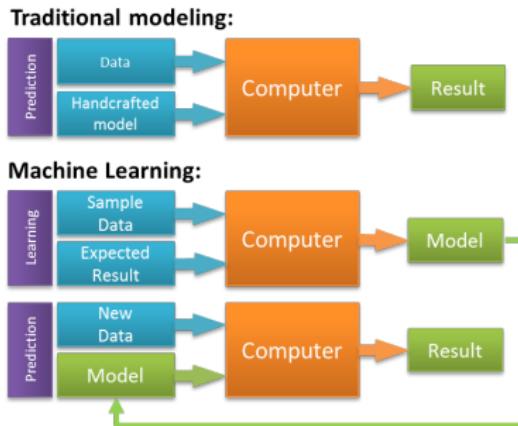
Other methods

Classification

KNN

SVM

Sự khác biệt giữa Programming và Machine Learning:



- **Programming:** Phụ thuộc nhiều vào con người trong việc tìm rules => phù hợp với bài toán dữ liệu nhỏ, đơn giản.
- **Machine Learning:** Phụ thuộc vào dữ liệu. Dữ liệu càng tổng quát, càng đa dạng thì các rules càng chính xác => phù hợp với bài toán dữ liệu lớn phức tạp.

Các bài toán của Machine Learning

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

- **Regression:** Cho 1 tập dữ liệu. Tìm mô hình (hàm số) phù hợp nhất (fit nhất) với tập dữ liệu.
- **Clustering:** Cho 1 tập dữ liệu. Hãy nhóm các phần tử thành các cụm riêng lẻ
- **Classification:** Cho 1 tập dữ liệu, trong đó các phần tử thuộc các loại khác nhau. Tìm cách để phân loại các phần tử đó theo loại nó thuộc về. Áp dụng với các phần tử mới vào, hãy phân loại nó về tập thích hợp.
Hệ quả được đưa ra:
- **Dimensionality reduction:** Giảm bớt số chiều biểu diễn dữ liệu
- **Model selection:** So sánh, đánh giá các phương pháp
- **Preprocessing:** Các phương pháp tiền xử lý dữ liệu để tối ưu kết quả

Định dạng dữ liệu trong ML

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

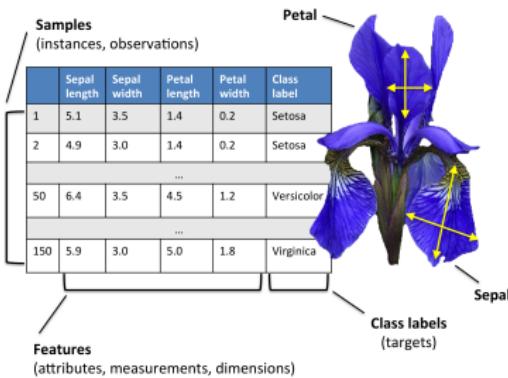
Kmeans

Other methods

Classification

KNN

SVM



- Dữ liệu thường lưu dạng các bảng hoặc ma trận
- Mỗi hàng là thông tin của một đối tượng (samples)
- Mỗi cột là 1 loại đặc trưng (features)
- Các đối tượng có thể thuộc về một lớp ID (label)

Phân loại các phương pháp ML

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

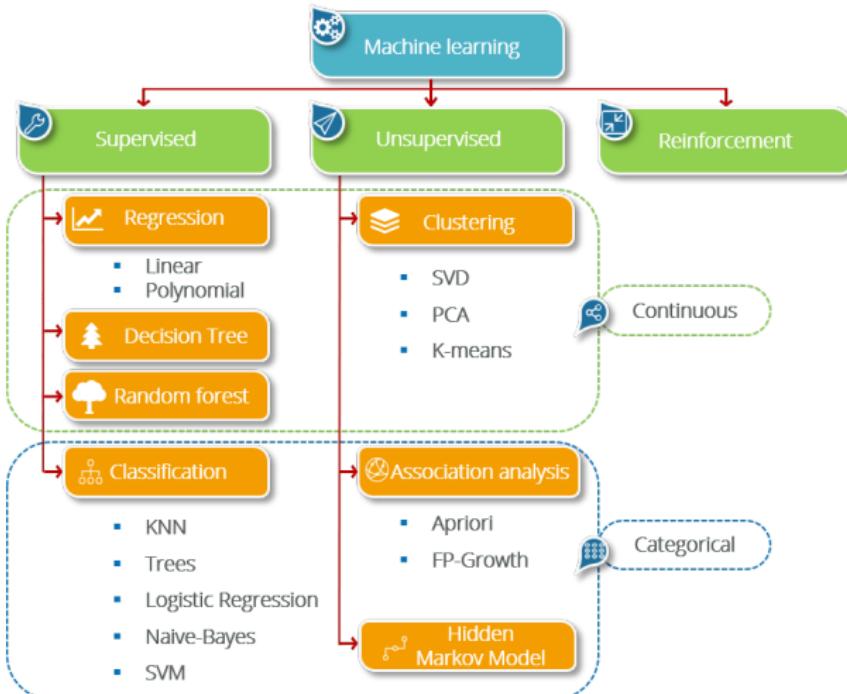
Kmeans

Other methods

Classification

KNN

SVM



Scikit-learn Module

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

- Là thư viện các thuật toán học máy (machine learning) sử dụng Python
- Module gồm các công cụ đơn giản, hiệu quả cho nhiều ứng dụng trong khai phá và phân tích dữ liệu
- Free, liên tục được phát triển, bảo trì, tối ưu.
- Được xây dựng dựa trên NumPy, SciPy và matplotlib. Nếu phải cài đặt lại từ pip, hãy đảm bảo cài đủ 3 thư viện trên trước.
- Có gần như đầy đủ các phương pháp ML phổ biến hiện nay.

```
1 import numpy as np
2 import sklearn
3 from sklearn.cluster import KMeans
4
```

Clustering-Phân cụm

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

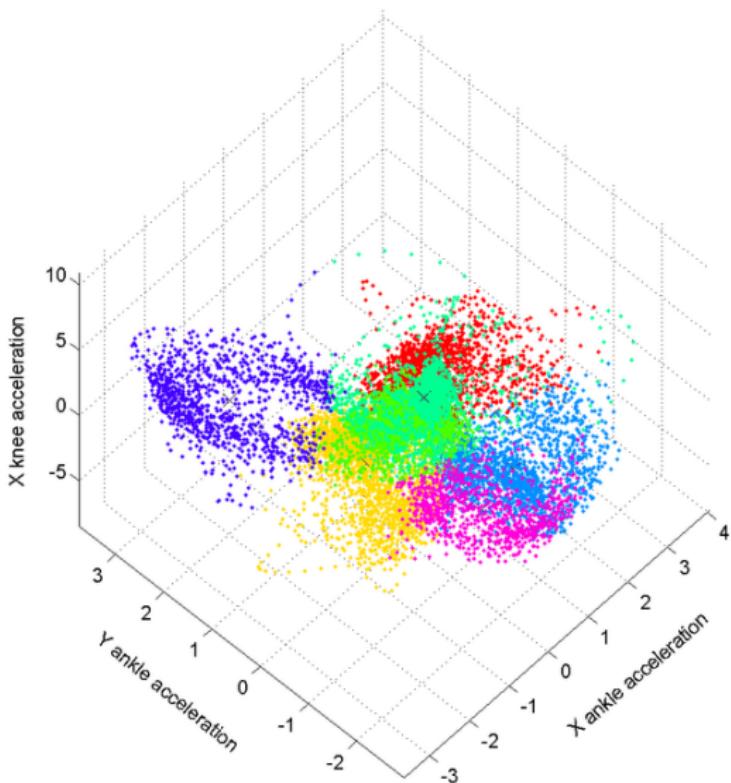
Kmeans

Other methods

Classification

KNN

SVM



Clustering-Phân cụm

Python

NGUYỄN
Hong Thinh

Introduction
ML

Clustering

Kmeans

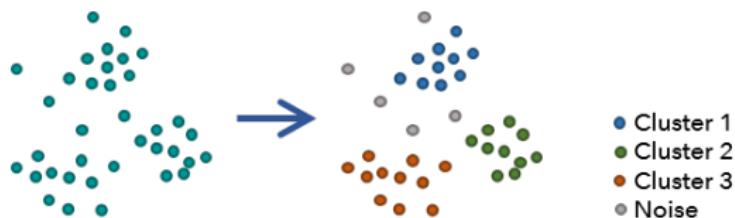
Other methods

Classification

KNN

SVM

- Phân cụm dữ liệu là phương pháp/tác vụ mà chương trình cần nhóm các đối tượng **giống nhau hơn** (theo nghĩa này hay nghĩa khác) với nhau so với các đối tượng khác vào một cụm



Clustering-Phân cụm

Python

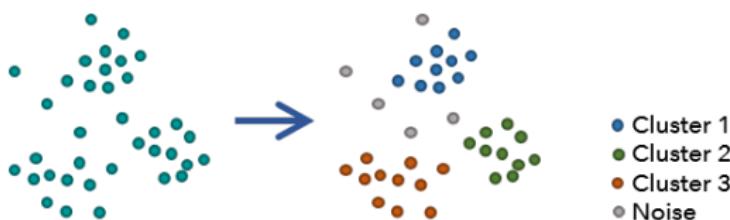
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

- Các đối tượng giống nhau, sẽ có các đặc điểm giống nhau
- => Sử dụng vector biểu diễn thông tin đặc trưng (features vector)
 - Các đối tượng giống nhau: khoảng cách vector đặc trưng nhỏ
 - Các đối tượng khác nhau: khoảng cách vector đặc trưng lớn



Khó khăn

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

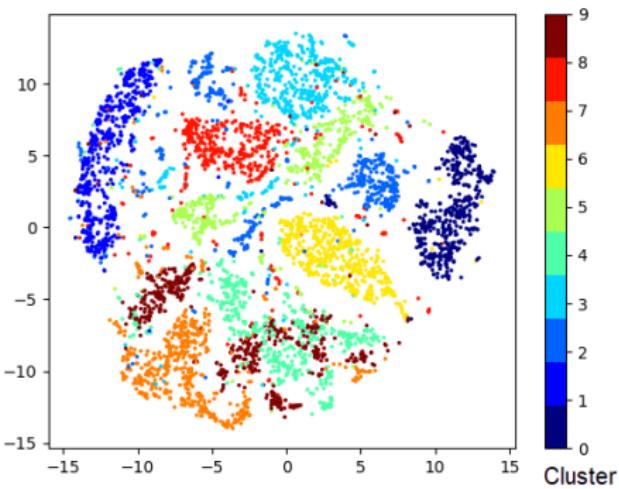
Other methods

Classification

KNN

SVM

- Bài toán phân cụm là bài toán không huấn luyện (unsupervised learning) => Kiểm chứng, đánh giá kết quả phân cụm
- Số cụm không cố định, khó ước lượng



Các thuật toán clustering phổ biến

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

- Kmeans
- Meanshift
- Agglomerative Hierarchical Clustering
- DB Scan
- Spectral clustering
- Chinese Whisper
- GMM
- More..

Kmeans clustering

Python

NGUYEN
Hong Thinh

Introduction
ML

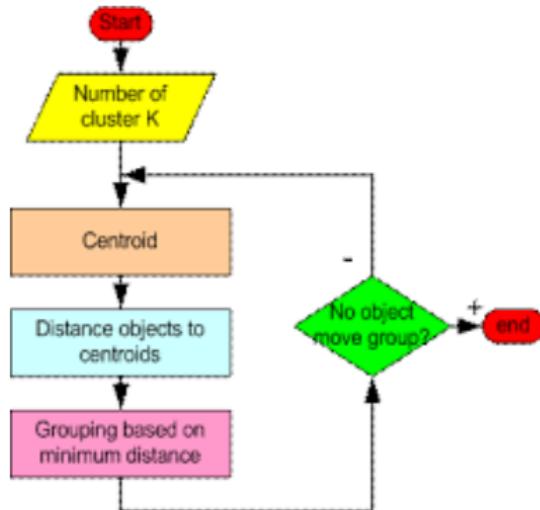
Clustering
Kmeans

Other methods

Classification

KNN
SVM

- Kmeans thực hiện phân nhóm N đối tượng $\{X\}$ thành K cụm. Mỗi cụm đặc trưng bởi centroids C (có thể không thuộc $\{X\}$).
- Phân cụm dùng phương pháp lặp, sao cho khoảng cách từ các điểm đến centroid gần nó nhất nhỏ nhất
- Thuật toán gồm 5 bước:



Kmeans clustering

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans

Other methods

Classification

KNN
SVM

- 1 Chọn số cụm K.
- 2 Khởi tạo K điểm làm centroids.
- 3 Tính khoảng cách từ tất cả các điểm $\{X\}$ đến tất cả các centroids.
- 4 Với mỗi điểm x_i , gán nó thuộc cụm ứng với centroids gần nó nhất (khoảng cách nhỏ nhất)
- 5 Sau khi gán hết các điểm của $\{X\}$, tính toán lại centroids, là trung bình của tất cả các điểm thuộc cụm đó.
- 6 (Điều kiện dừng) Tính toán độ thay đổi giữa center cũ và mới (tổng khoảng cách dịch giữa center cũ=> mới). Nếu giá trị này lớn hơn threshold, quay lại bước 3

Áp dụng

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

- Thực hiện kmeans trên Python
- Kiểm tra và so sánh với sử dụng thư viện có sẵn sklearn
- So sánh trên Iris dataset và Mnist dataset

IRIS dataset:

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

```
1 # Load data from source:  
2 url = 'https://archive.ics.uci.edu/ml/machine-  
learning-databases/iris/iris.data'  
3 iris = np.genfromtxt(url, delimiter=',', dtype='  
object')  
4 #Vector dac trung va nhan:  
5 Features=iris[:,0:4].astype("float")  
6  
7 ### Chuyen flowers-name into number ID [0, 1, 2]  
8 count =0  
9 for i in set(iris[:,4]): ### Set() return uniq  
    item in list  
    iris[iris==i]=count ###  
    count+=1  
12 target_ids = iris[:,4] #label
```

IRIS dataset:

Python

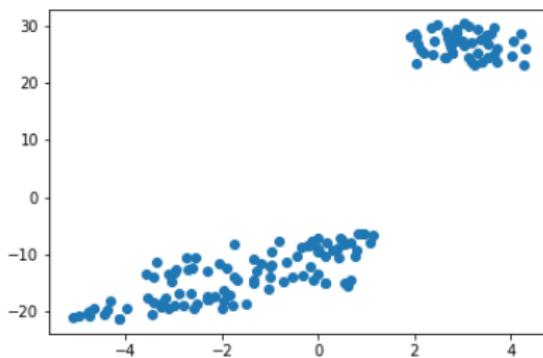
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

```
1 #####Visualization:  
2 from sklearn.manifold import TSNE  
3 import matplotlib.pyplot as plt  
4 import matplotlib.image as mpimg  
5 import matplotlib.cm as cm  
6  
7 tsne = TSNE(n_components=2, random_state=0)  
8 Features_2d = tsne.fit_transform(Features)  
9 plt.scatter(Features_2d[:, 0], Features_2d[:, 1])
```



IRIS dataset:

Python

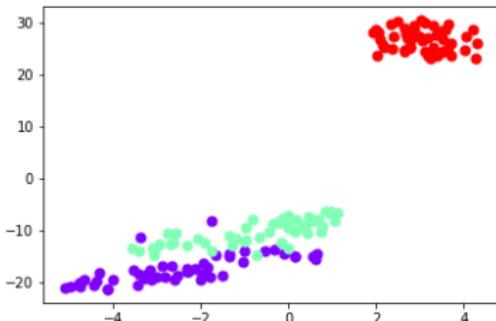
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

```
1 #####Visualization with color:  
2 def get_colors(ids):  
3     x = np.arange(len(ids))  
4     ys = [i+x+(i*x)**2 for i in range(len(ids))]  
5     return cm.rainbow(np.linspace(0, 1, len(ys)))  
6 colors = get_colors(list(set(target_ids)))  
7 for i in set(target_ids):  
8     indexes = [z for z in range(len(Features_2d)) if  
9                 target_ids[z] == i]  
9     plt.scatter(Features_2d[indexes, 0], Features_2d[  
          indexes, 1], s=50, color=colors[i])
```



IRIS dataset:

Python

NGUYEN
Hong Thinh

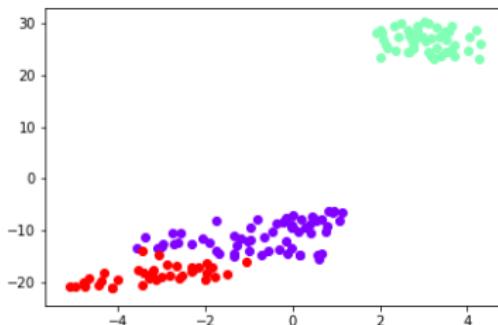
Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Clustering into 3 groups:

```
1 from sklearn.cluster import KMeans
2 kmeans = KMeans(n_clusters=3).fit(Features)
3 colors = get_colors(list(set(kmeans.labels_)))
4 for i in set(kmeans.labels_):
5     indexes = [z for z in range(len(Features_2d)) if
6                 kmeans.labels_[z] == i]
7     plt.scatter(Features_2d[indexes, 0], Features_2d
8                 [indexes, 1], color=colors[i])
```



IRIS dataset:

Python

NGUYEN
Hong Thinh

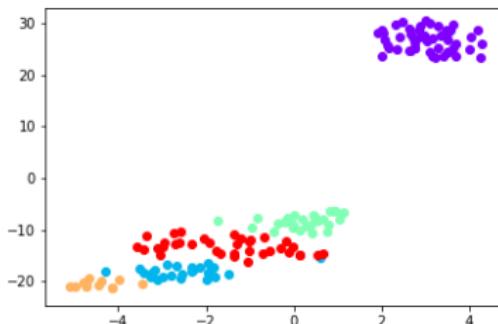
Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Clustering into 5 groups:

```
1 from sklearn.cluster import KMeans
2 kmeans = KMeans(n_clusters=5).fit(Features)
3 colors = get_colors(list(set(kmeans.labels_)))
4 for i in set(kmeans.labels_):
5     indexes = [z for z in range(len(Features_2d))
6                 if kmeans.labels_[z] == i]
7     plt.scatter(Features_2d[indexes, 0],
8                 Features_2d[indexes, 1], color=colors[i])
```



IRIS dataset:

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

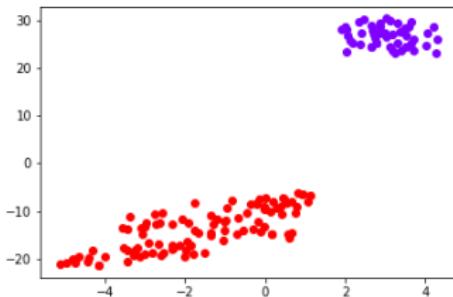
Classification

KNN

SVM

Ngoài ra có thể thử 1 vài thuật toán khác: DBSCAN

```
1 from sklearn.cluster import DBSCAN
2 db = DBSCAN(eps=1, min_samples=5).fit(Features)
3 colors = get_colors(list(set(db.labels_)))
4 for i in set(db.labels_):
5     indexes = [z for z in range(len(Features_2d))
6                 if db.labels_[z] == i]
7     plt.scatter(Features_2d[indexes, 0],
8                 Features_2d[indexes, 1], color=colors[i])
```



IRIS dataset:

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

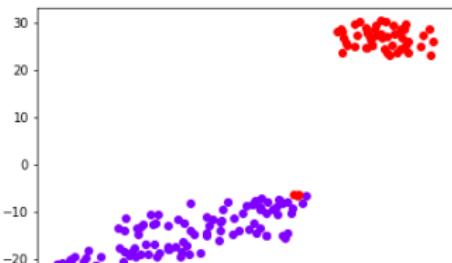
Classification

KNN

SVM

Clustering sử dụng MeanShift

```
1 from sklearn.cluster import MeanShift ,  
    estimate_bandwidth  
2 BW = estimate_bandwidth(Features , quantile=0.25)  
3 ms = MeanShift(bandwidth=BW , bin_seeding=True)  
4 ms.fit(Features)  
5 colors = get_colors(list(set(ms.labels_)))  
6 for i in set(db.labels_):  
7     indexes = [z for z in range(len(Features_2d))  
8                 if ms.labels_[z] == i]  
9     plt.scatter(Features_2d[indexes , 0] ,  
10                Features_2d[indexes , 1] , color=colors[i])
```



Classification

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

- Phân loại là bài toán gán ID hay class-id cho đối tượng

Classification

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

- Phân loại là bài toán gán ID hay class-id cho đối tượng
- Nguyên tắc: So sánh "**sự giống nhau**" với các đối tượng đã biết. 2 đối tượng **giống nhau** sẽ thuộc cùng class

Classification

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

- Phân loại là bài toán gán ID hay class-id cho đối tượng
- Nguyên tắc: So sánh "**sự giống nhau**" với các đối tượng đã biết. 2 đối tượng **giống nhau** sẽ thuộc cùng class
- **Giống nhau:** Các đặc trưng giống nhau.

Classification

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

- Phân loại là bài toán gán ID hay class-id cho đối tượng
- Nguyên tắc: So sánh "**sự giống nhau**" với các đối tượng đã biết. 2 đối tượng **giống nhau** sẽ thuộc cùng class
- **Giống nhau:** Các đặc trưng giống nhau.
- Sử dụng vector đặc trưng để so sánh 2 đối tượng.
Khoảng cách 2 features vector càng bé thì 2 đối tượng càng giống nhau

Các phương pháp Classification phổ biến

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

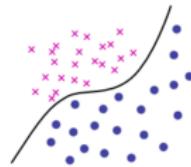
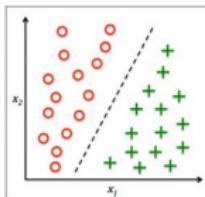
Other methods

Classification

KNN

SVM

- Tuyến tính phi tuyến
- K-láng giềng gần (KNN)
- SVM
- Decision Tree
- Random Forest



Dữ liệu trong Classification

Python

NGUYEN
Hong Thinh

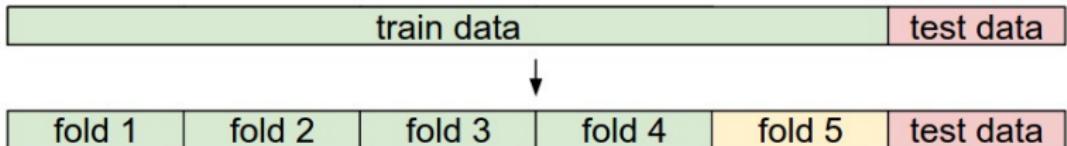
Introduction
ML

Clustering
Kmeans
Other methods

Classification

KNN
SVM

- Dữ liệu được chia làm Training set và Testing set
- Trainning: Tìm ra tiêu chuẩn/quy tắc (**model**) để phân loại dữ liệu thành các class khác nhau
- Testing: Đánh giá hiệu quả của (**model**) vừa tìm được
- Cross-validation: Kỹ thuật sử dụng để đánh giá thuật toán (**model**) một cách "fair" nhất



Cross-validation

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

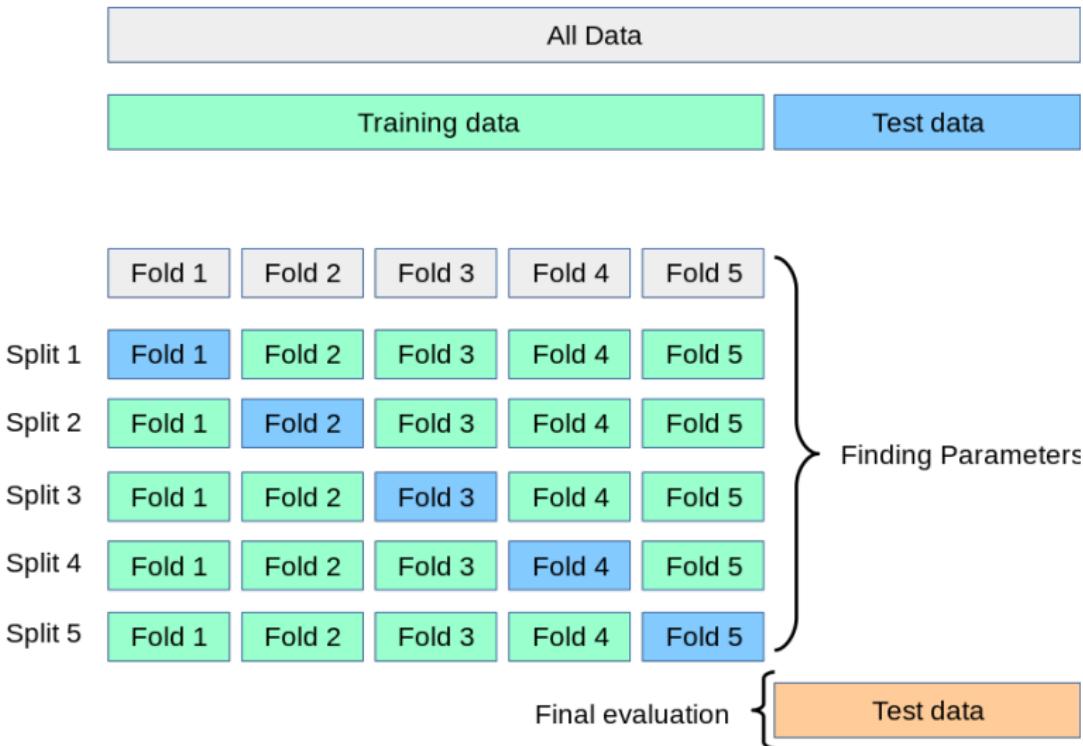
Other methods

Classification

KNN

SVM

5-folds crossvalidation



KNN

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

Ý tưởng: "Tell me who your friends are and I'll tell you who you are"



KNN

Python

NGUYEN
Hong Thinh

Introduction ML

Clustering

Kmeans

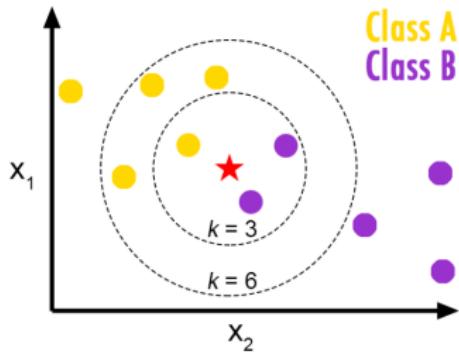
Other methods

Classification

KNN

SVM

- Definition: My neighbors (my friends) are ones **closed** to me
 - Các điểm gần nhau trong không gian (features space) có khoảng cách nhỏ hay các đặc điểm giống nhau
 - K khác nhau có thể cho ra kết quả khác nhau!!!



Sơ đồ thuật toán:

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

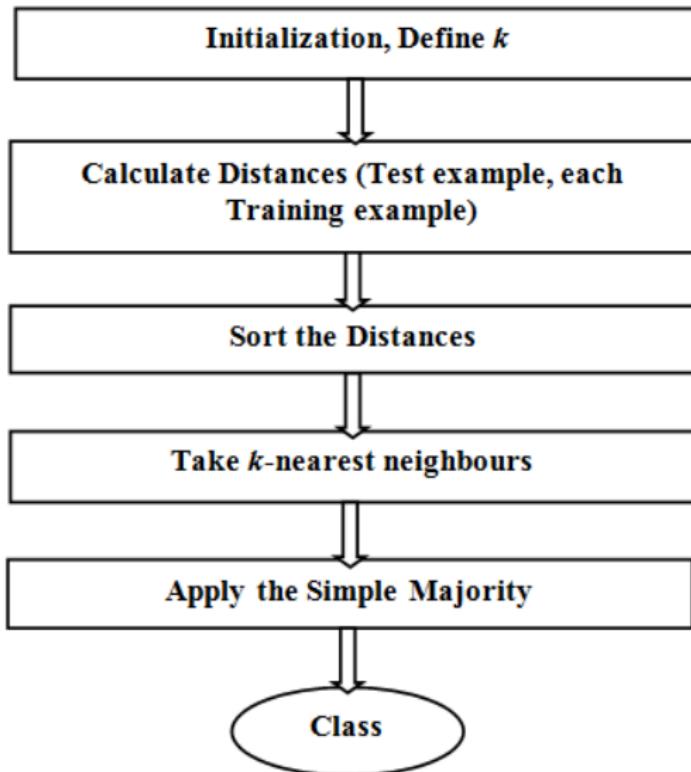
Kmeans

Other methods

Classification

KNN

SVM



Sơ đồ thuật toán:

Python

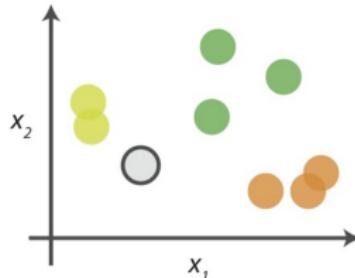
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

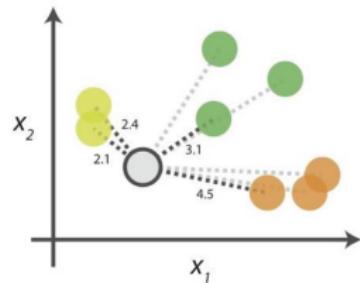
Classification
KNN
SVM

0. Look at the data



Say you want to classify the grey point into a class. Here, there are three potential classes - lime green, green and orange.

1. Calculate distances



Start by calculating the distances between the grey point and all other points.

Sơ đồ thuật toán:

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification

KNN
SVM

2. Find neighbours

Point Distance

		2.1	→ 1st NN
		2.4	→ 2nd NN
		3.1	→ 3rd NN
		4.5	→ 4th NN

Next, find the nearest neighbours by ranking points by increasing distance. The nearest neighbours (NNs) of the grey point are the ones closest in dataspace.

3. Vote on labels

Class	# of votes
	2
	1
	1

Class wins the vote!
Point is therefore predicted to be of class .

Vote on the predicted class labels based on the classes of the k nearest neighbours. Here, the labels were predicted based on the k=3 nearest neighbours.

KNN

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans
Other methods

Classification

KNN
SVM

□ Euclidean: $D = \sqrt{\sum_{i=1}^k (x_i - y_i)^2}$

□ Manhattan: $D = \sum_{i=1}^k |x_i - y_i|$

□ Minkowski: $D = \left(\sum_{i=1}^k (|x_i - y_i|)^q \right)^{1/q}$

□ Hamming: $D = \sum_{i=1}^k |x_i - y_i|;$

- $x = y \rightarrow D = 0$

- $x \neq y \rightarrow D = 1$

Hàm khoảng cách khác nhau cũng có thể ảnh hưởng đến kết quả!!!

KNN

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

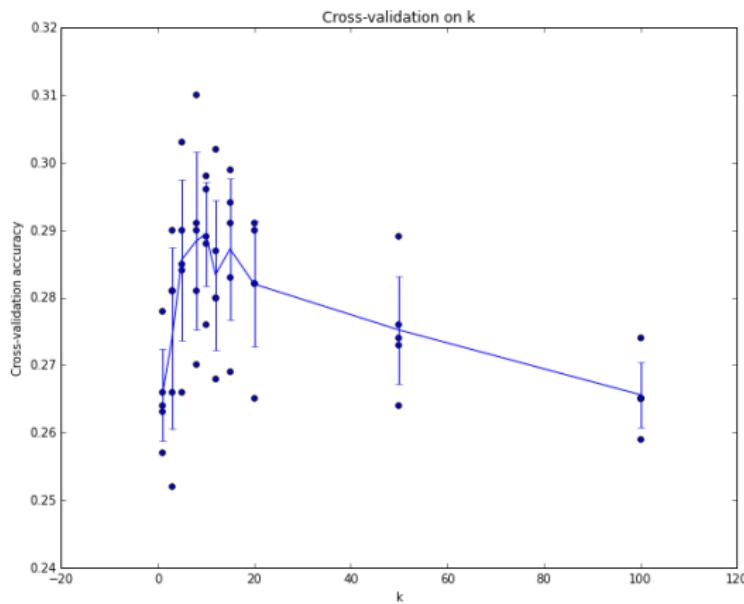
Other methods

Classification

KNN

SVM

Chọn K tốt nhất sử dụng cross-validation:



Bài tập

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Làm việc với dữ liệu Iris dataset:

- Load dữ liệu, chuyển tên hoa -> id
- Chọn ngẫu nhiên 90 đối tượng (30 mỗi loại) làm dữ liệu training. Phần còn lại làm dữ liệu testing.
- Viết chương trình thực hiện thuật toán phân loại KNN với K cho trước (Dùng khoảng cách Euclid (L2))
- Kiểm thử với tập testing. K bằng bao nhiêu cho kết quả tốt nhất.
- Sử dụng tool sklearn, đánh giá lại thuật toán

Bài tập

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Làm việc với dữ liệu MNIST dataset:

- Load dữ liệu
- Chọn ngẫu nhiên 1000 đối tượng làm dữ liệu training và 200 dữ liệu testing.
- Kiểm thử với chương trình viết lúc trước
- Sử dụng tool sklearn, đánh giá lại thuật toán

SVM classification

Python

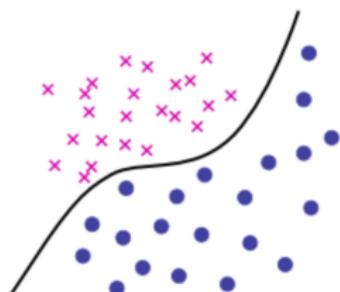
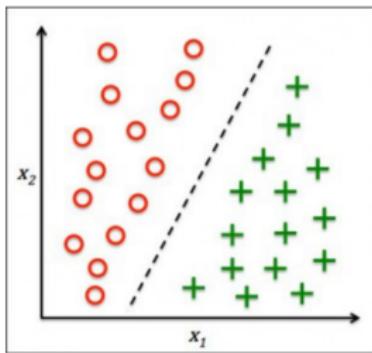
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Làm thế nào để phân chia 2 tập điểm?



SVM

Python

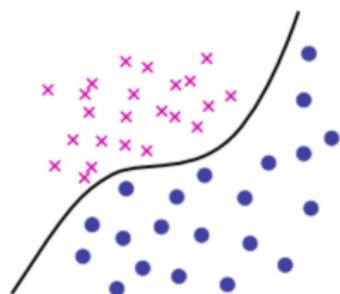
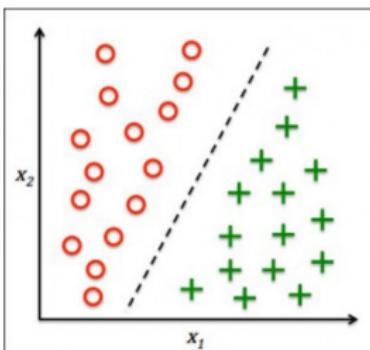
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Làm thế nào để phân chia 2 tập điểm?



- Điểm 1 chiều => chọn ngưỡng:
- Điểm 2 chiều (x_1, y_1), (x_2, y_2)... => Chọn đường (đường thẳng $y=ax+b$, đường cong...)
- Điểm 3 chiều (x_1, y_1, z_1), (x_2, y_2, z_2)...=> Chọn mặt $ax+by+cz=0$ (lý tưởng: mặt phẳng)
- Điểm 4 chiều hoặc >4 chiều ???? => **Siêu phẳng (hyper plane)**

SVM

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

- SVM là phương pháp phân loại có giám sát (Supervised learning)
- Được giới thiệu vào năm 1992 bởi Boser, Guyon và Vapnik
- Mục đích là xác định siêu phẳng **tối ưu** để phân loại 2 tập điểm

SVM

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

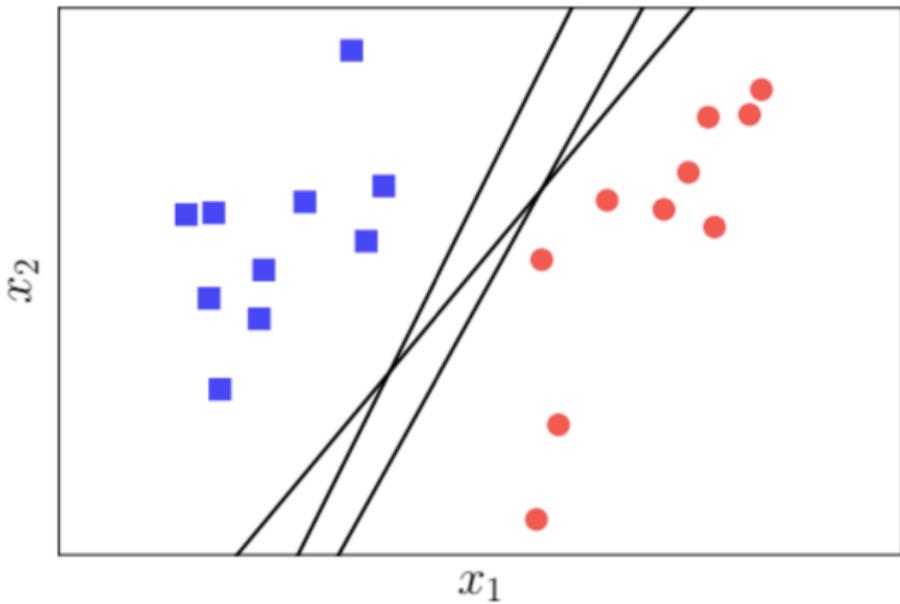
Other methods

Classification

KNN

SVM

Có vô số siêu phẳng thoả mãn. Đáp án tối ưu?



SVM

Python

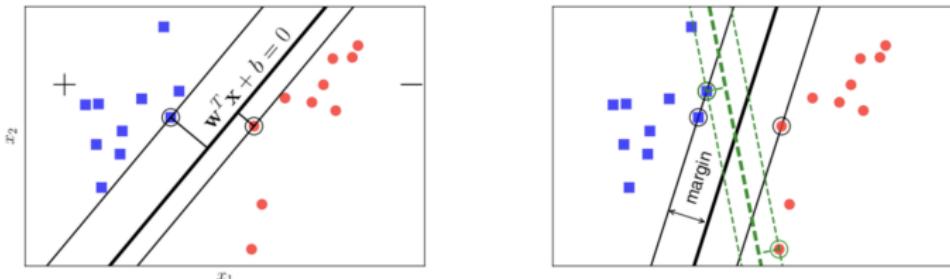
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Có vô số siêu phẳng thoả mãn. Đáp án tối ưu?



Support vectors và margin:

- Margin: là khoảng cách gần nhất từ một điểm dữ liệu (Support vectors) tới mặt phân cách ấy
- Bài toán tối ưu trong SVM là đi tìm đường phân chia sao cho margin giữa hai lớp là lớn nhất
- Mục đích là xác định siêu phẳng **tối ưu** để phân loại 2 tập điểm

Underfitting/Overfitting

Python

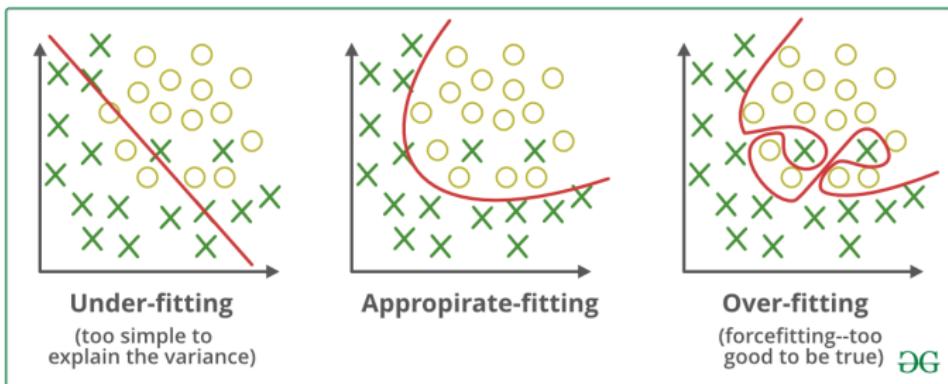
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Là vấn đề thường gặp khi thực hiện các thuật toán học máy nói chung



Underfitting/Overfitting

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

	Underfitting	Just right	Overfitting
Symptoms	- High training error - Training error close to test error - High bias	- Training error slightly lower than test error	- Low training error - Training error much lower than test error - High variance
Regression			
Classification			
Deep learning			
Remedies	- Complexify model - Add more features - Train longer		- Regularize - Get more data

SVM

Python

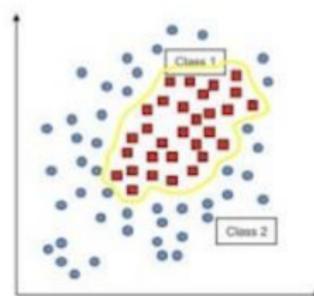
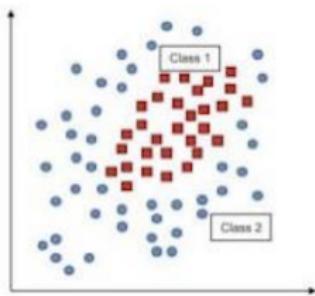
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Giải quyết bài toán classification với trường hợp dữ liệu “non linearly separable”?



Non Linear
Decision
Boundary

Non linear classifier

SVM

Python

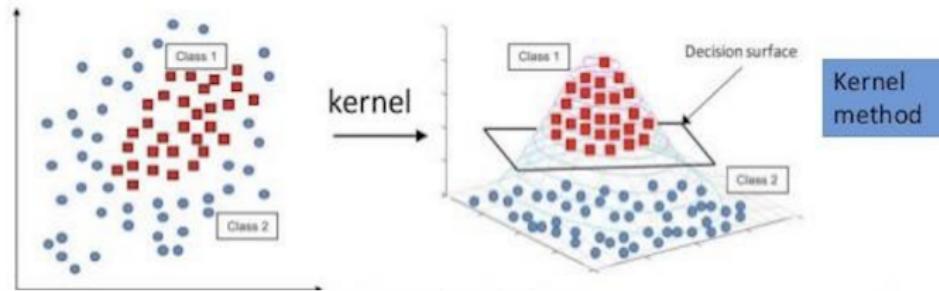
NGUYEN
Hong Thinh

Introduction
ML

Clustering
Kmeans
Other methods

Classification
KNN
SVM

Giải quyết bài toán classification với trường hợp dữ liệu “non linearly separable”?



Non linear classifier => Kernel based method

- Tăng số chiều dữ liệu để chuyển từ non-linear => linear
- Kernel function: Đảm bảo tính khoảng cách giữa các điểm
- Kernel based method: Phép chiếu tỉ lệ

SVM

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

Một vài kernel phổ biến:

Các kernel thông dụng

Tên kernel	Công thức	Thiết lập hệ số
'linear'	$\mathbf{x}^T \mathbf{z}$	không có hệ số
'poly'	$(r + \gamma \mathbf{x}^T \mathbf{z})^d$	d : degree, γ : gamma, r : coef0
'sigmoid'	$\tanh(\gamma \mathbf{x}^T \mathbf{z} + r)$	γ : gamma, r : coef0
'rbf'	$\exp(-\gamma \ \mathbf{x} - \mathbf{z}\ _2^2)$	$\gamma > 0$: gamma

Bài tập

Python

NGUYEN
Hong Thinh

Introduction
ML

Clustering

Kmeans

Other methods

Classification

KNN

SVM

Làm việc với dữ liệu Iris và MNIST dataset:

- Kiểm thử thuật toán SVM với các kernel khác nhau.

```
1 from sklearn import svm
2 #X_training: trainingdata, y_training: label of
   trainingdata
3 clf = svm.SVC(kernel='linear', c=1, gamma=0.3)
4 clf.fit(X_training, y_training)
5
6 clf.predict(X_test)
```