

# Pattern Recognition Homework 4 announcement

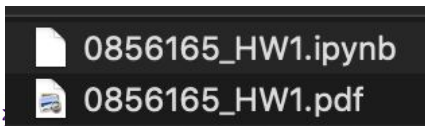
TA: 楊証琨, Jimmy

Ph.D. student at National Taiwan University

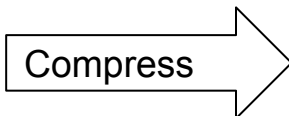
[d08922002@csie.ntu.edu.tw](mailto:d08922002@csie.ntu.edu.tw)

# Homework 4

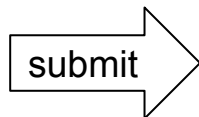
- **Deadline: May. 25, Wed at 23:59.**
  1. Code assignment (50%): Implement cross-validation and hyperparameter searching for SVM model training
  2. Short answer questions (50%)
- Submit your **1) code (.py/.ipynb)** and **2) reports (.pdf)** on [E3](#)
  - [Sample Code](#)
  - [HW4 questions](#)
- Please follow the **file naming rules <STUDENT ID>\_HW4.pdf**, otherwise, you will get penalty of your scores



Compress



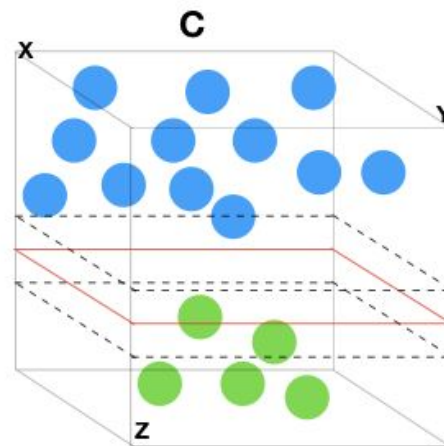
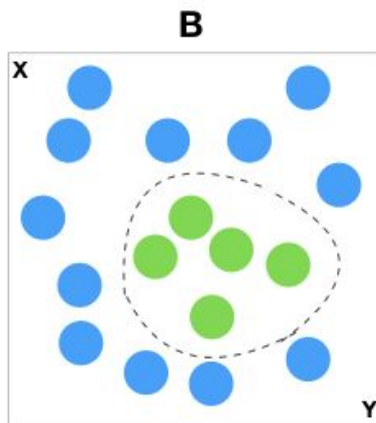
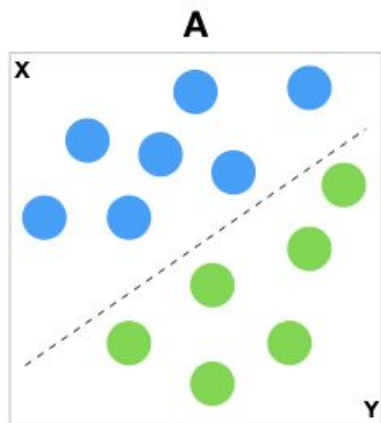
submit



[E3](#)

# Support vector machines

- Support Vectors Classifier tries to find the best hyperplane to separate the different classes by maximizing the distance between sample points and the hyperplane



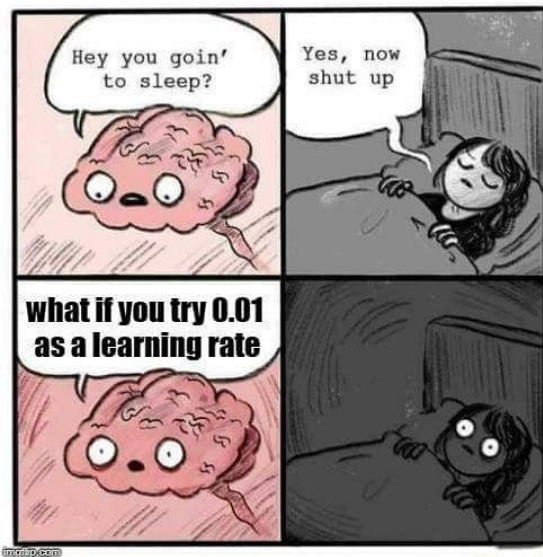
# No need to implment SVM!

- Since SVM requires lots of difficult mathematical operations, we will not ask you to implement SVM in homework 4 :)



# Grid search and cross-validation

- There are lots of hyperparameters in SVM. In this homework, you will need to implement **grid search** and **cross-validation** to find the best hyperparameters of the SVM on the provided dataset

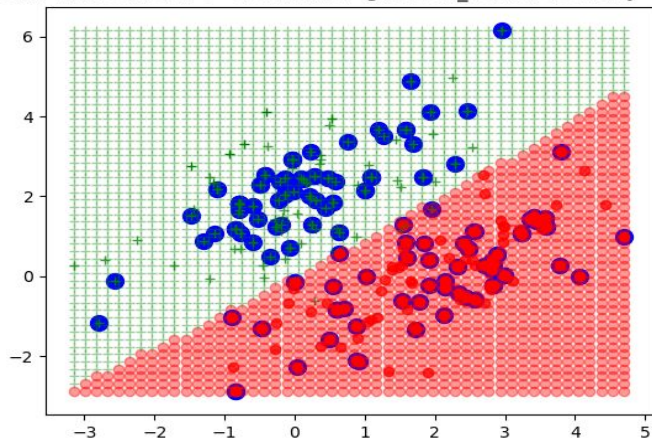


# Hyperparameter searching

- Suppose we want to find the best values of two hyperparameters for an RBF kernel SVM, namely **C** and **gamma**
  - [Interactive demo](#)
  - [Explanation of C and gamma](#)
- There are many combinations to be considered!

`sample_size (1000)`

SVM with CVXOPT, C=0.01 kernel=gaussian\_kernel: accuracy=0.97



# Hyperparameter searching: Grid search

- Grid search exhaustively considers all hyperparameter combinations and picks the best one based on the model that gives the best performance

```
C = [0.1, 1, 10] #3 values
```

```
gamma = [0.01, 0.1, 1, 10] #4 values
```

```
# There are totally 12 combinations for tuning
```

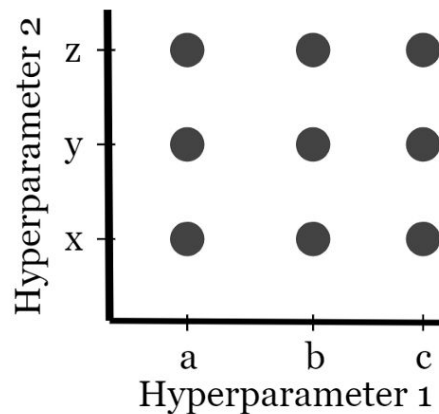
C\gamma	0.01	0.1	1	10
0.1	[0.1, 0.01]	[0.1, 0.1]	[0.1, 1]	[0.1, 10]
1	[1, 0.01]	[1, 0.1]	[1, 1]	[1, 10]
10	[10, 0.01]	[10, 0.1]	[10, 1]	[10, 10]

## Grid Search

Pseudocode

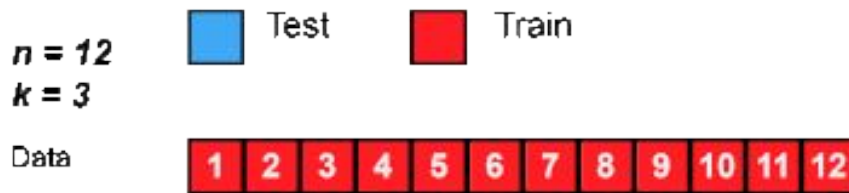
```
Hyperparameter_One = [a, b, c]
```

```
Hyperparameter_Two = [x, y, z]
```



# K-fold Cross-validation

- The main idea behind cross-validation is that **each data point** in the dataset has the opportunity of being tested
- Illustration of K-fold cross-validation when  $n=12$  observations and  $K=3$ . After data is shuffled, a total of 3 models will be trained and tested.





# K-fold Cross-validation

- We split the dataset into  $K$  parts: one part is used for validation, and the remaining  $K-1$  parts are merged into a training subset. This process repeats  $K$  times, with each part used exactly once as the validation data

$n = 12$

$k = 3$

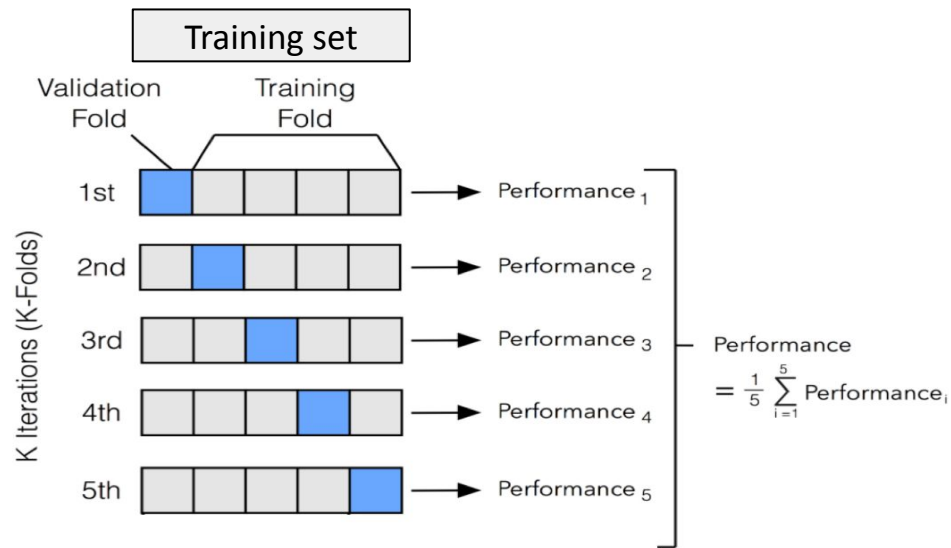
Data



Test



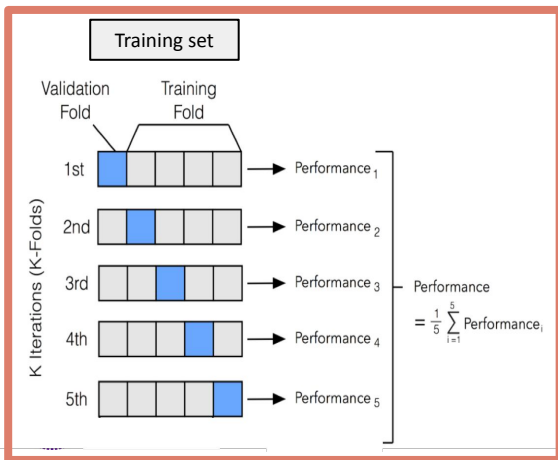
Train



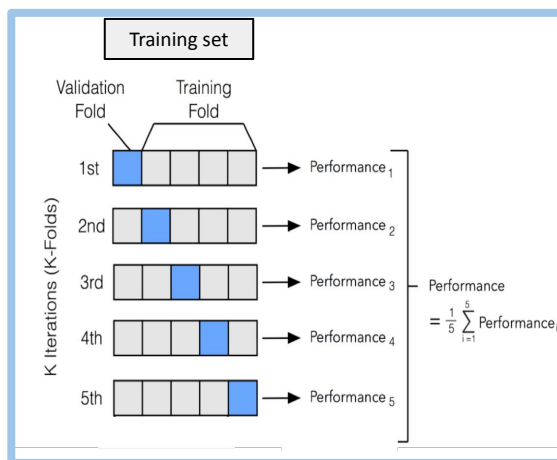
# K-fold Cross-validation for hyperparameter searching

- We can experiment with 12 combinations of hyperparameters defined in page 7. For each combination, we apply the K-fold cross-validation and get the average performance
- Find the best combination which yield best performance

Combination **1**: [0.1, 0.01]  
average score=0.8

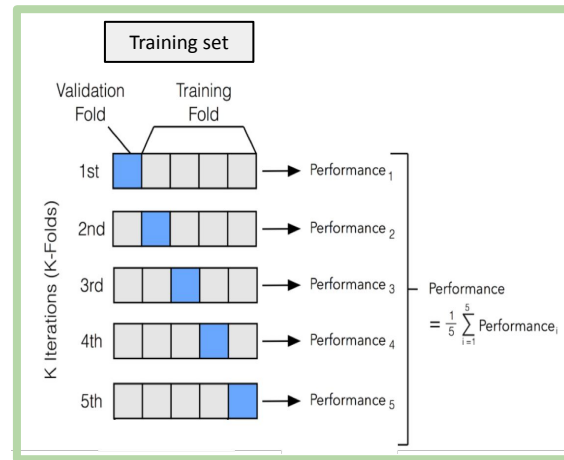


Combination **2**: [0.1, 0.1]  
average score=0.91



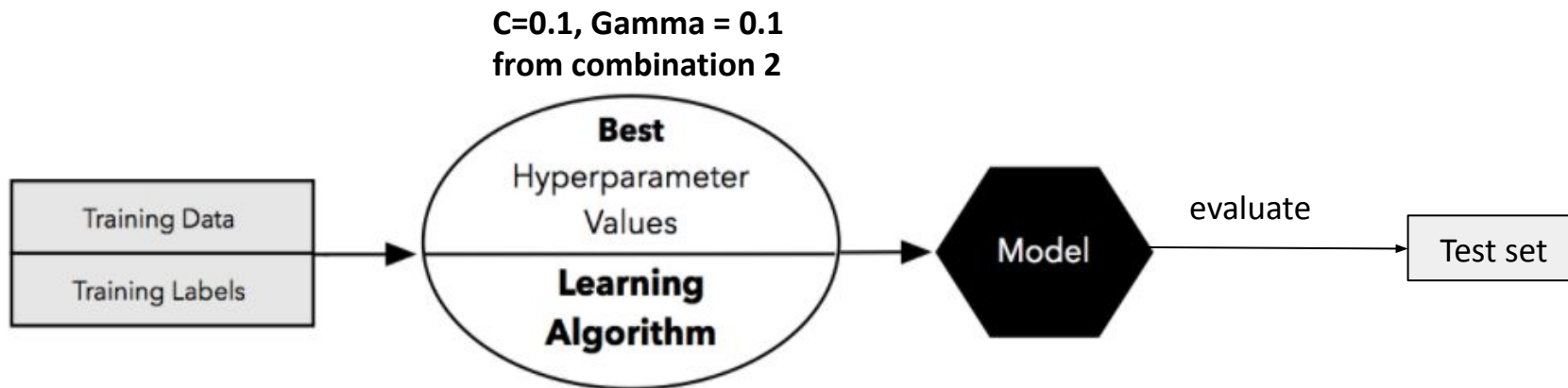
...

Combination **12**: [10, 10]  
average score=0.75



# K-fold Cross-validation for hyperparameter searching

- Finally, train your model **on the whole training set** with the best hyperparameters and evaluate on the test set



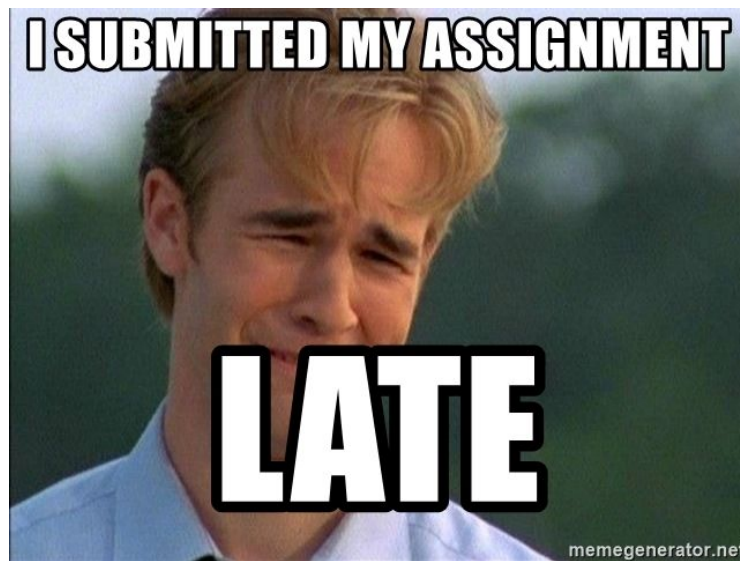
# Reference

- K-fold Cross-Validation & Grid Search
- SVM hyperparameter tuning



# Late Policy

- We will deduct a late penalty of 20 points per additional late day
- For example, If you get 90 points of this HW but delay for **two days**, your will get only  $90 - (20 \times 2) = 50$  points!



# Notice

- Submit your homework on [E3-system](#) !
- Check your email regularly, we will mail you if there are any updates or problems of the homework
- If you have any questions or comments for the homework, please mail TAs and cc Prof. Lin
  - Prof. Lin, [lin@cs.nctu.edu.tw](mailto:lin@cs.nctu.edu.tw)
  - TA Jimmy, [d08922002@csie.ntu.edu.tw](mailto:d08922002@csie.ntu.edu.tw)
  - TA 晨軒, [derekt.cs06@nctu.edu.tw](mailto:derekt.cs06@nctu.edu.tw)
  - TA 政儒, [ace52751208@gmail.com](mailto:ace52751208@gmail.com)



# Have fun!



Neural  
Network



HOG+SVM

