



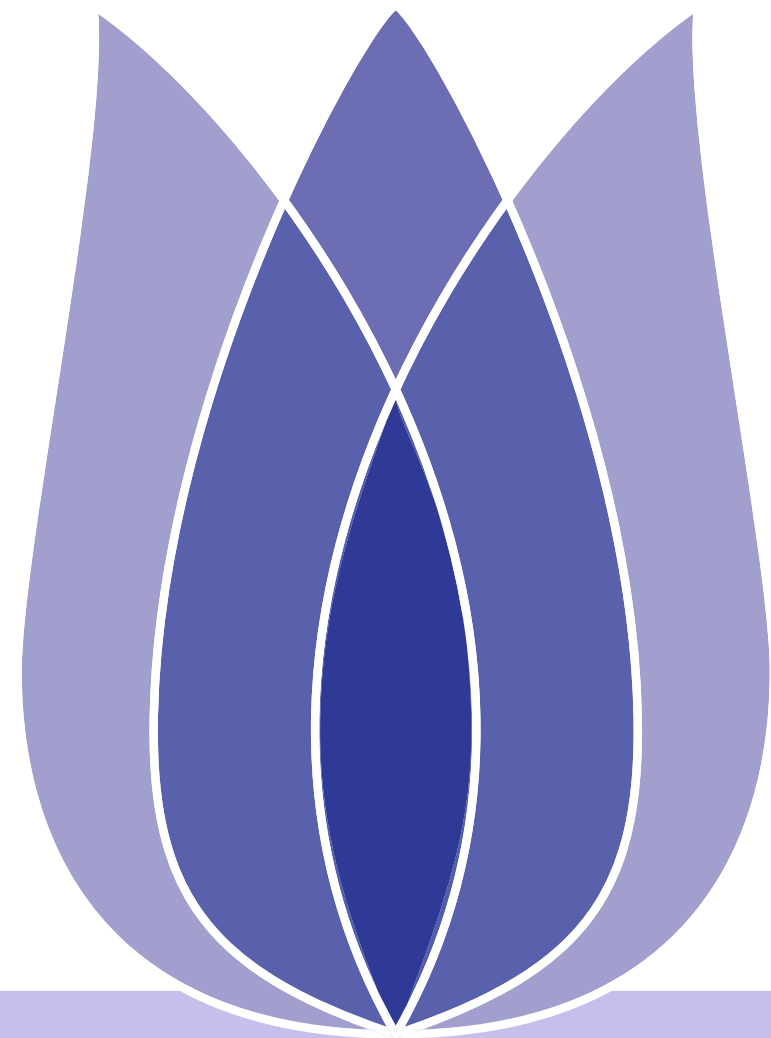
# The First Report

Wang Mingxi

Jilin University

College of Computer Science and Technology

(None)





# Overview

[Learning content](#)

[Hands on practice](#)

[Plan for the following week](#)

## Learning content

Decision tree

Ensemble Learning

Neural Network

## Hands on practice

Generative Adversarial Network GAN

## Plan for the following week

Plan for the following week



Learning content

- Decision tree
- Ensemble Learning
- Neural Network

Hands on practice

Plan for the following week

# Learning content



**Decision tree:** A decision tree is a prediction model used to predict the category of samples. In the structure of these trees, leaf nodes give categories and inner nodes represent attributes.

### Plan for the following week



### Plan for the following week

下午 2:51 5月10日周一

< 98 Q 周 1

机器学习

统计学习方法

草稿

机器学习

统计学习方法

第8章 集成学习  
Boosting 的框架：AdaBoost  
(一种弱分类器)

Boosting 是一种可解释学习提升为强学习的算法。这算法的工作机制是：先求弱的决策函数——一个弱学习，再根据新学习到的决策对训练样本分布进行调整，使得那些被弱学习判断错误的样本分布权重增加多少倍，然后基于调整后的样本分布来训练下一个基学习器，如此重复进行，直至基学习器的形式达到预先设定的值，最后将若干个基学习器进行加权组合。

集成学习的基本公式： $G(x) = \alpha_1 g_1(x) + \alpha_2 g_2(x) + \dots + \alpha_n g_n(x)$

$\alpha_j g_j(x)$  称为第  $j$  个基学习器。 $\alpha_j > 0$  且  $\sum_{j=1}^n \alpha_j = 1$

$G(x)$  称为集成学习模型。 $\alpha_j$  表示该个基学习器的重要性

定义：一个投票规则如下：

$$\text{sign}(f_m(x)) = \begin{cases} 1 & x < 2.5 \\ -1 & 2.5 < x < 5.5 \\ 1 & 5.5 < x < 8.5 \\ 1 & x > 8.5 \end{cases}$$

弱分类器

x	1	2	3	4	5	6	7	8	9
f <sub>1</sub> (x)	-1	-1	-1	-1	1	1	1	1	1
f <sub>2</sub> (x)	-1	-1	-1	1	1	1	1	1	1

注意：上图有3个错误

Bagging (一种弱分类器)

Bagging [Breiman, 1996a] 是并行式集成学习方法最著名的代表。从名字即可看出，它直接源于我们在 2.2.3 节介绍的自助采样法 (bootstrap sampling)。给定包含  $m$  个样本的数据集，我们先随机取出一个样本放入训练集，再把这个样本放回数据集做替换，使得下次采样时该样本有可能被采到。这样，我们每次做  $n$  次随机采样操作，我们将得到  $m$  个样本的采集集。若训练集中的所有样本在采样集中多次出现，有的则从未出现。(由式(2.1)可知，初始训练集中的某些数据点的频率会大于1)

因此说，我们可以把训练了  $n$  个含  $n$  个训练样本的采样集，然后基于每个采样集训练出一个弱分类器，再将这些基学习器进行结合，这就是 Bagging 的基本思想。

与之相对的，adaboost 是用一个（这里重点）数据集训练，每次训练出一个基学习器，与之前训练的 adaboost 使用一个（这里重点）数据集训练，每次训练出一个基学习器。

Bagging 的算法描述如图 8.5 所示。

输入：训练集  $D = [(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)]$ ；  
基学习算法  $L$ ；  
训练次数  $T$ ；  
过程：  
1. for  $t = 1, 2, \dots, T$  do  
2.  $S_t \sim D$ ;  $D_t \leftarrow S_t$   
3. end for  
输出： $H(x) = \arg \max_y \sum_{t=1}^T h_t(x) = p(y)$  (即类标)  
注：这可以就是投票法，哪个  $y$  (即类标) 得到的基分类器  $p(y)$  的票数总和最多，那个  $y$  作为集成分类器的输出。

图 8.5 Bagging 算法

AdaBoost	只适用于二分类问题	单一弱分类器
Bagging	可用于多类、回归等	多个弱分类器并行工作

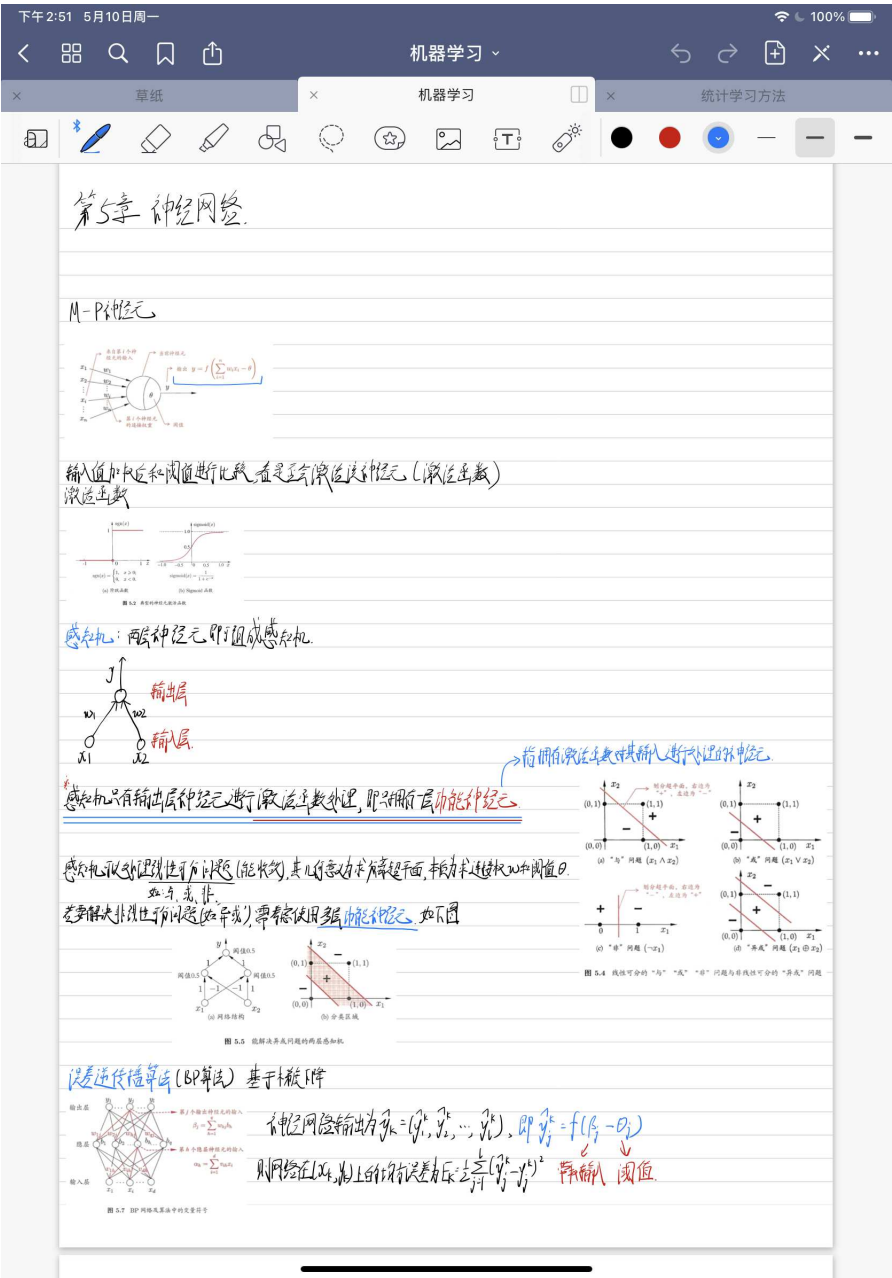
7 / 13



# Neural Network

Learning content
Decision tree
Ensemble Learning
Neural Network
Hands on practice
Plan for the following week

Neural Network:A mathematical or computational model that imitates the structure and function of a biological neural network and is used for estimating or approximating functions.







[Learning content](#)

**Hands on practice**

Generative Adversarial Network GAN

[Plan for the following week](#)

# Hands on practice



# Generative Adversarial Network GAN

Learning content  
Hands on practice  
Generative Adversarial Network GAN  
Plan for the following week

- Generating Adversarial networks (GAN) is a method of unsupervised learning in which two neural networks play each other against each other.

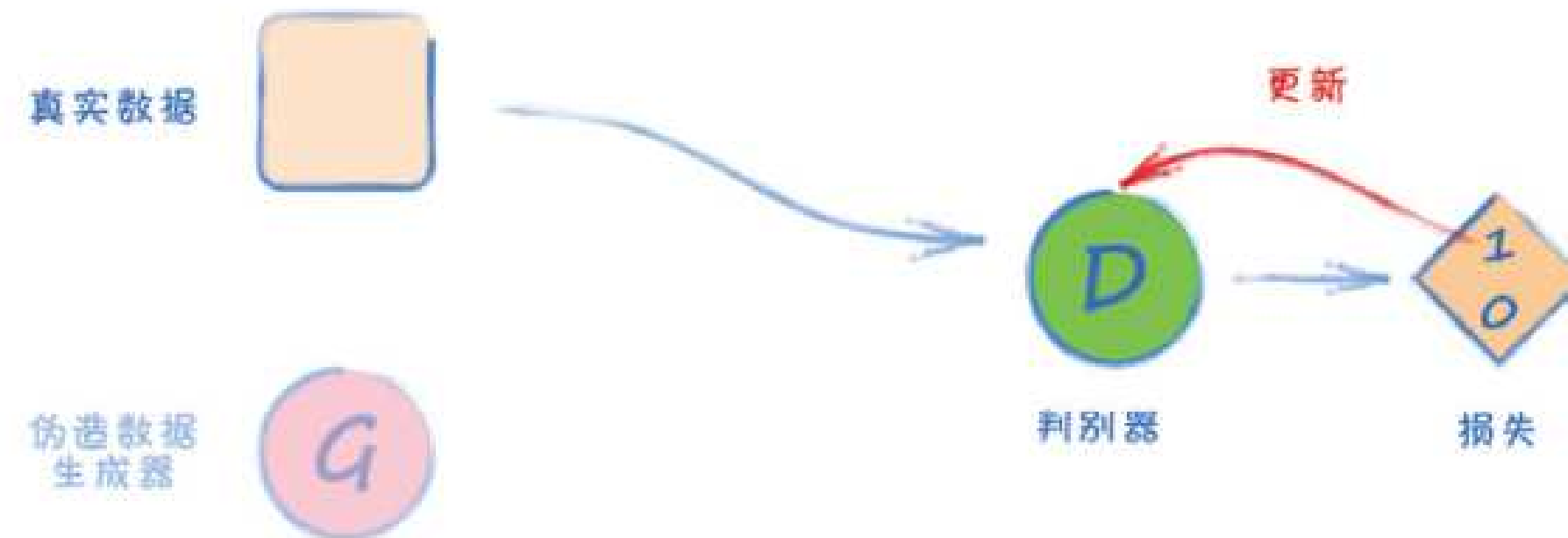


Figure 1



# Generative Adversarial Network GAN

- Learning content
- Hands on practice
- Generative Adversarial Network GAN
- Plan for the following week

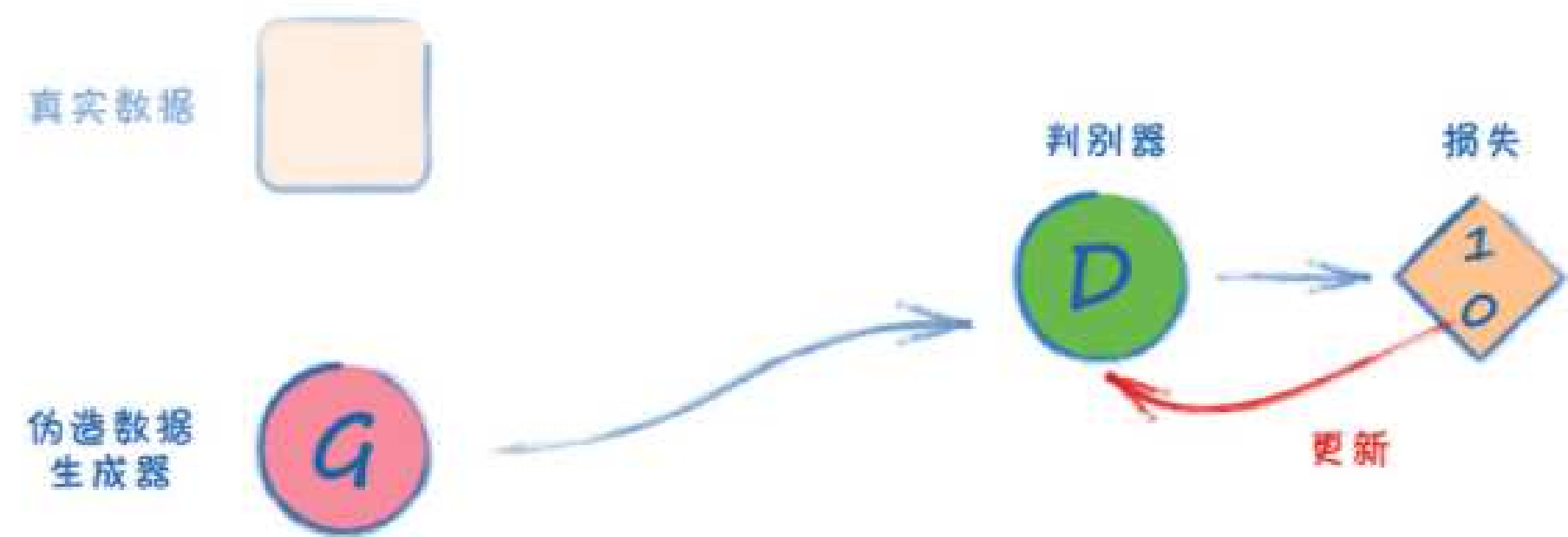


Figure 2: preview

# Generative Adversarial Network GAN

- Learning content
- Hands on practice
- Generative Adversarial Network GAN
- Plan for the following week

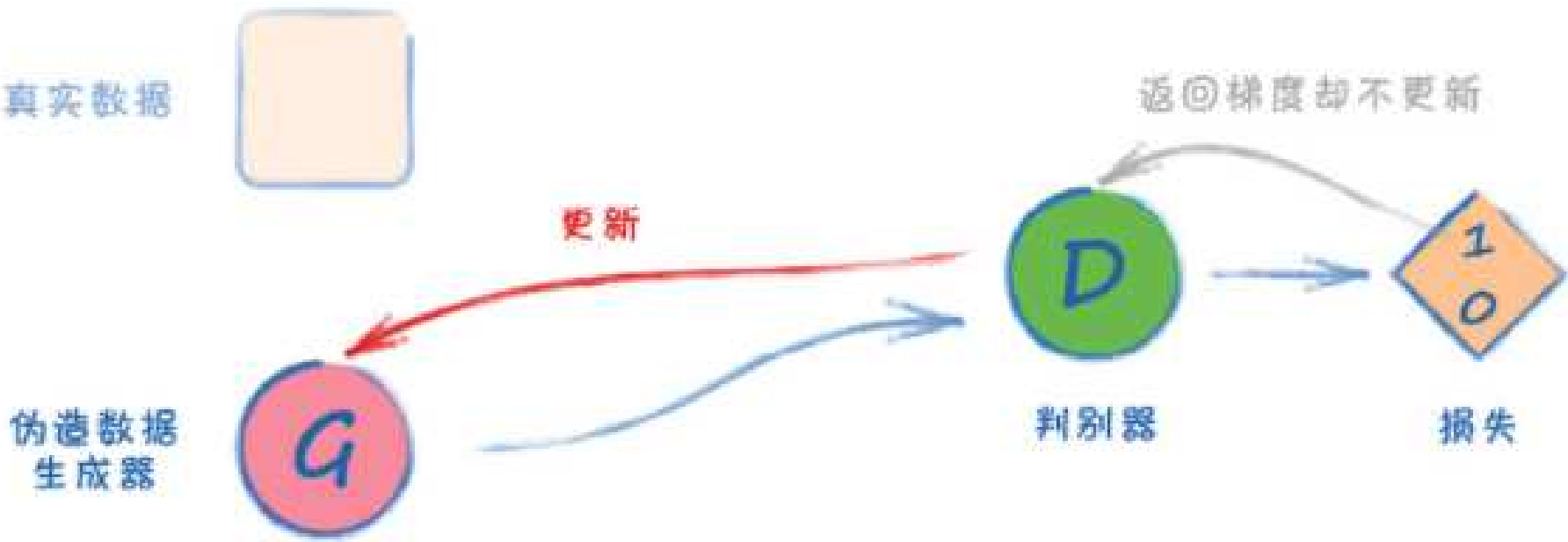


Figure 3: preview



[Learning content](#)

[Hands on practice](#)

[Plan for the following week](#)

[Plan for the following week](#)

# Plan for the following week



# Plan for the following week

- [Learning content](#)
- [Hands on practice](#)
- [Plan for the following week](#)
- [Plan for the following week](#)

- I’ll continue to learn the basics next week.
- I’m going to go ahead and program this stuff out.



# Contact Information

Wang Mingxi  
College of Computer Science and Technology  
Jilin University, China

-  MXWANG@TULIP.ACADEMY
-  TEAM FOR UNIVERSAL LEARNING AND INTELLIGENT PROCESSING

