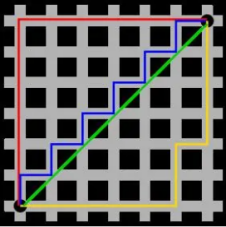
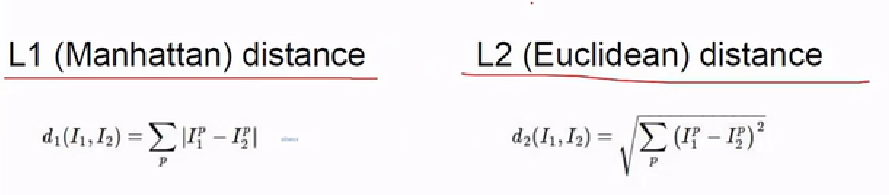
## 1.1K临近与交叉验证

1. 超参数：在机器学习的[上下文](https://baike.baidu.com/item/%E4%B8%8A%E4%B8%8B%E6%96%87/2884376" \t "https://baike.baidu.com/item/%E8%B6%85%E5%8F%82%E6%95%B0/_blank)中，超参数是**在开始学习过程之前设置值的参数**，而不是通过训练得到的参数数据。通常情况下，需要对超参数进行优化，给学习机选择一组最优超参数，以提高学习的性能和效果。

（参数难以理解，曼哈顿距离（红线），欧氏距离/直线距离（绿））参数含义





1. K最近邻(k-Nearest Neighbor，KNN)分类算法，是一个理论上比较成熟的方法，也是最简单的机器学习算法之一。该方法的思路是：在特征空间中，如果一个样本附近的k个最近(即特征空间中最邻近)样本的大多数属于某一个类别，则该样本也属于这个类别。物以类聚
2. 交叉验证

情景：选择超参数时

Test data是非常宝贵的，不能用它来验证参数，否则就没有真实数据测试了

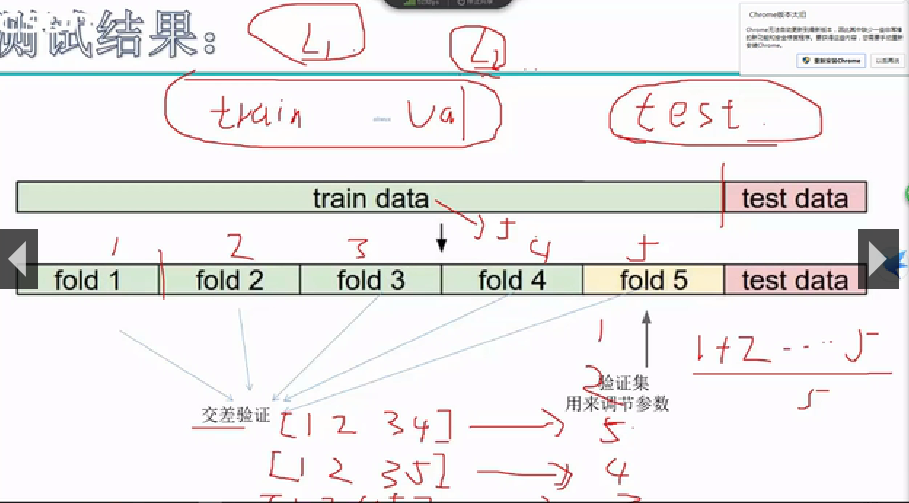
所以将train data划分为多组如下图示意，用fold1~4训练，用fold5验证，又产生问题：

如果fold5里有异常数据怎么办？怎样尽可能提高结果的准确性？

**交叉验证：用fold1235训练用fold4验证、用fold1245训练用fold3验证、用fold1345训练用fold2验证、用fold2345训练用fold1验证，得出结果相加除以五（取平均）**

假设对L1进行交叉验证的精度85%，L2精度95%，就选L2

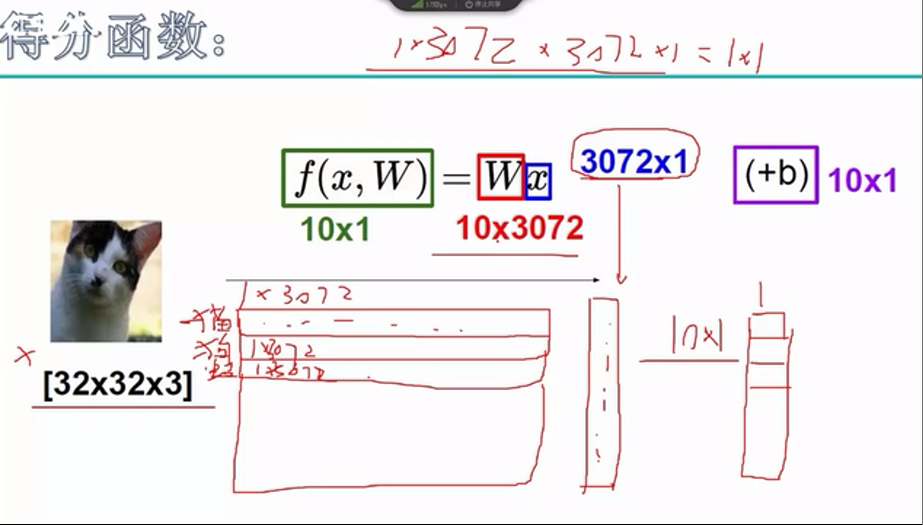
所以选择超参数L2

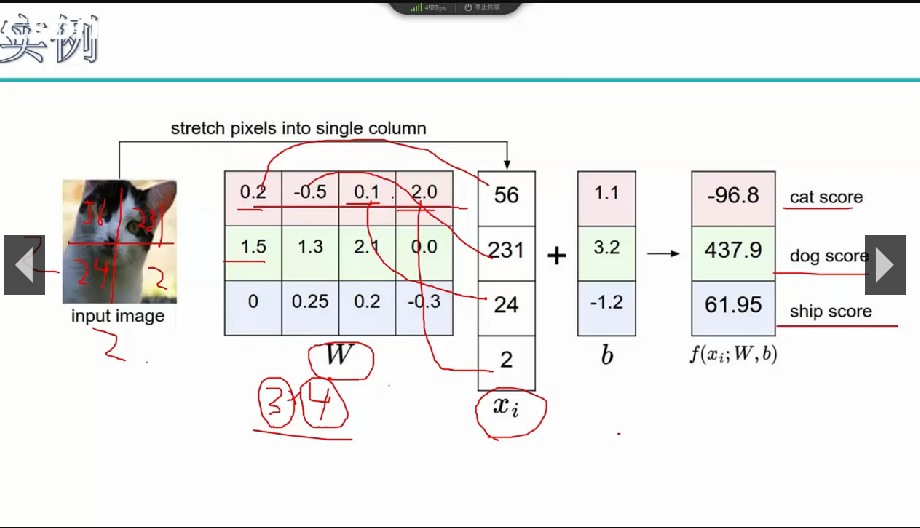


调好参数，用test data进行最后的评估

## 1.2得分函数、损失函数

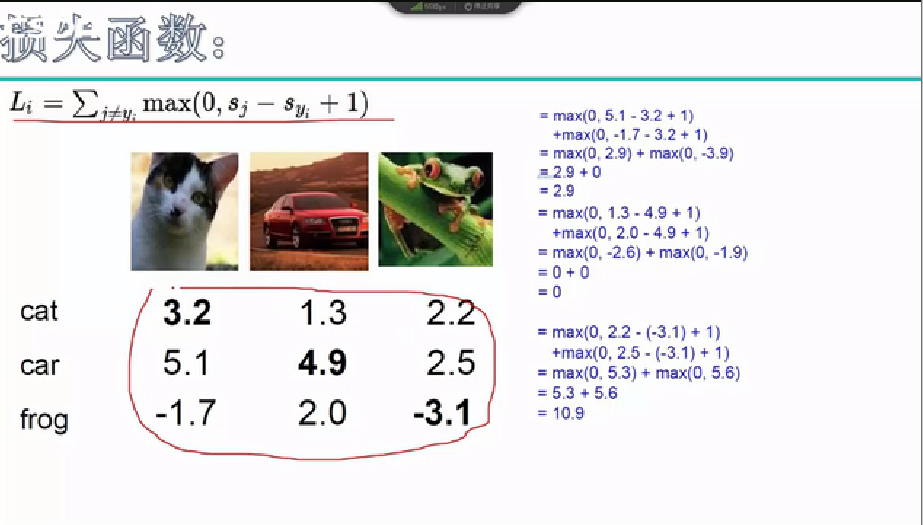
1. 得分函数：我希望针对每个像素块，得到对每个分类都不同的得分值

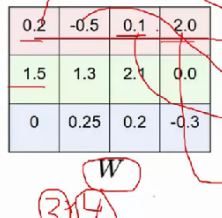




发现狗的得分多最后判断为狗，这明显不对，我们需要告诉这个模型！用什么呢？

2.损失函数（损失越大效果越坏）评估每组权重参数



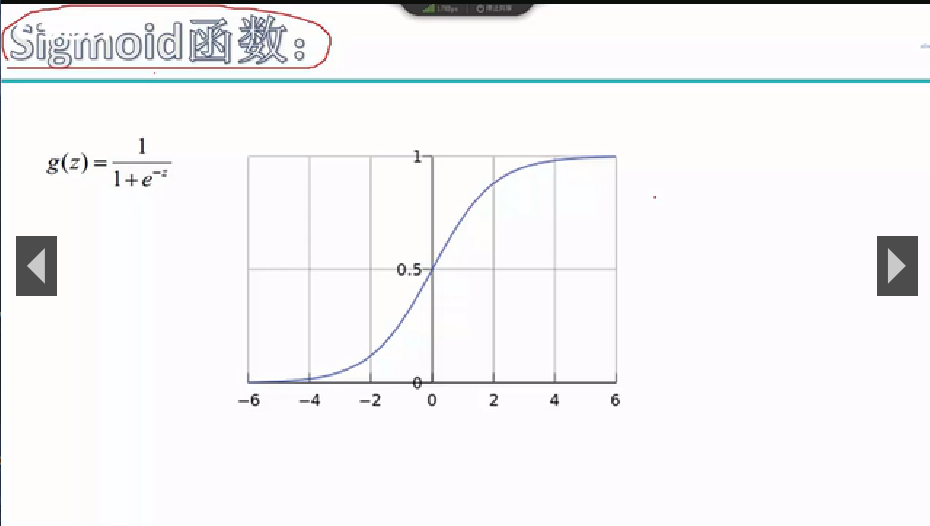
权重W的评估标准loss function，比如有十组W，如图按得分函数计算出十个结果再根据结果计算损失函数的值最小的那个就可以作为权重。

激活函数：线性转非线性

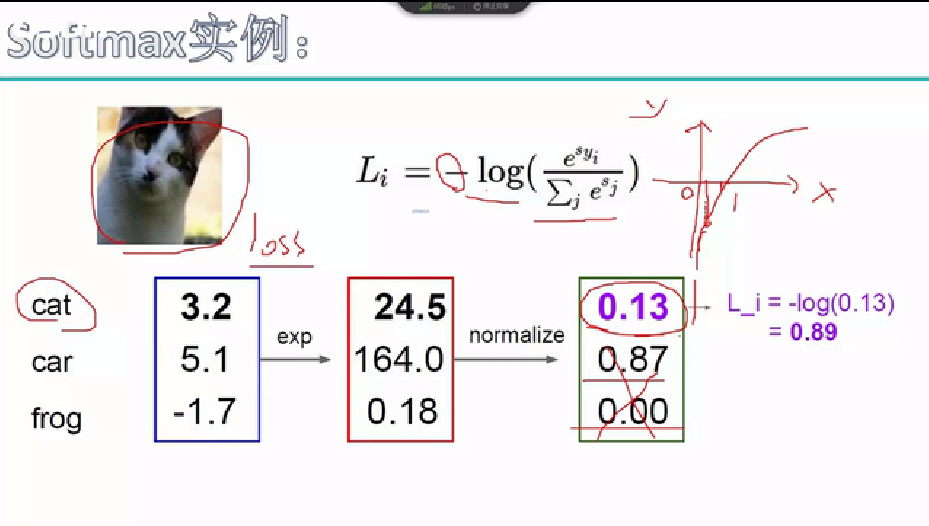
## **1.3Softmax分类器**

输出属于某类的概率

举例：



1. 算得分值x
2. 映射增大差距ex
3. 求得概率



## 2.1梯度下降原理

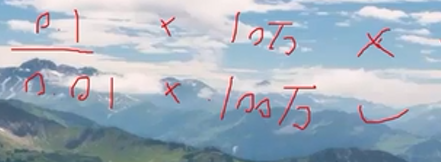
损失函数的值越大，效果越坏，怎么让损失函数的值最快下降？

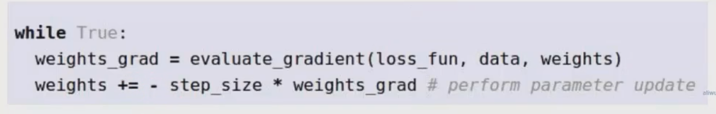


**确定方向**：梯度方向是向上，应该是梯度的反方向

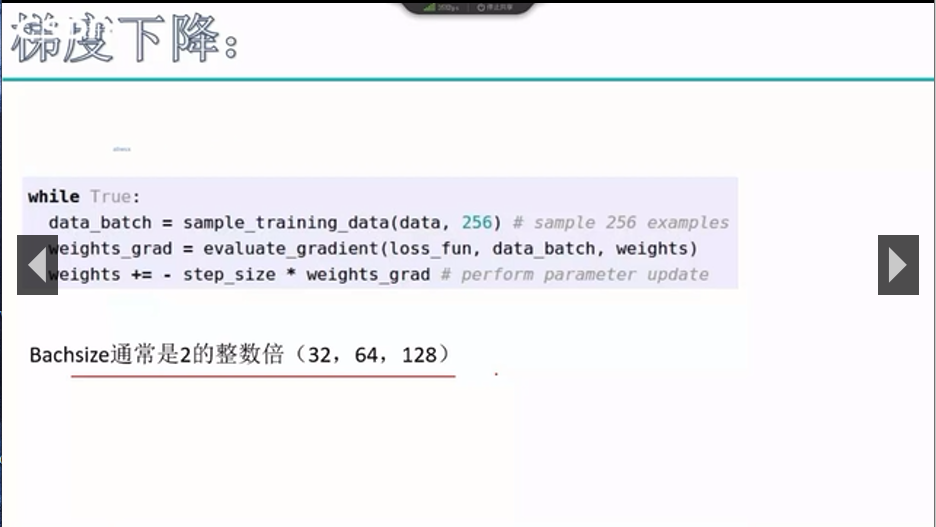


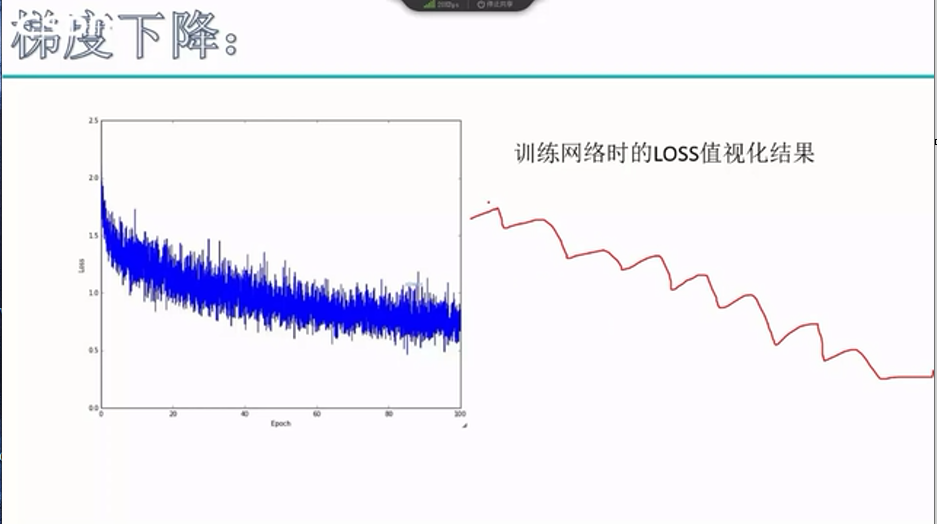
**确定步长**：如上图，步长过大跳过了全局最低点B



伪代码：

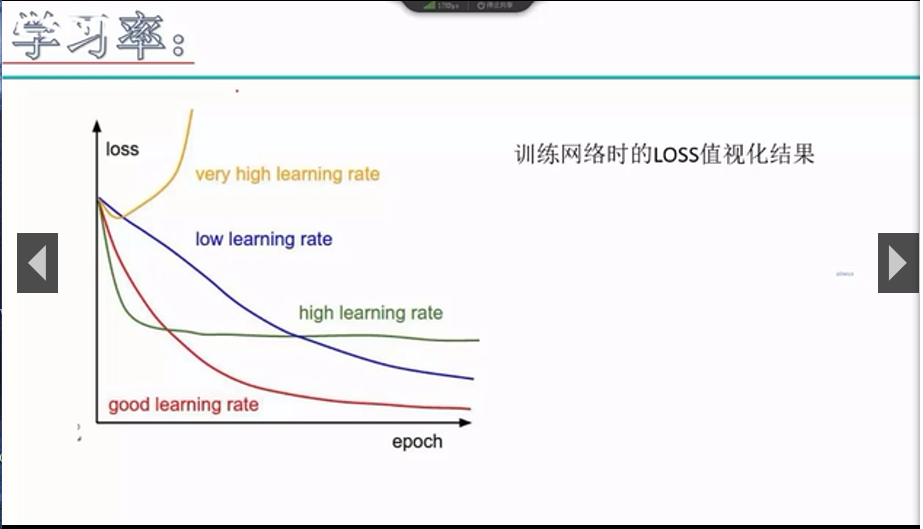
Bachsize图像批处理：if一张图像本身有问题（噪点之类的）那么参数更新计算出的方向可能会出错，所以一次处理多张图片取平均值。





## 2.2学习率的作用

学习率就类似之前的步长：

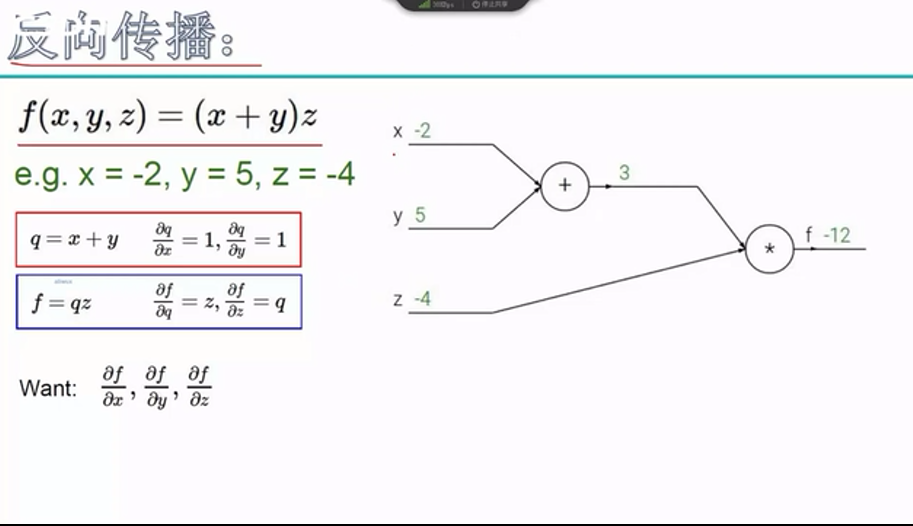


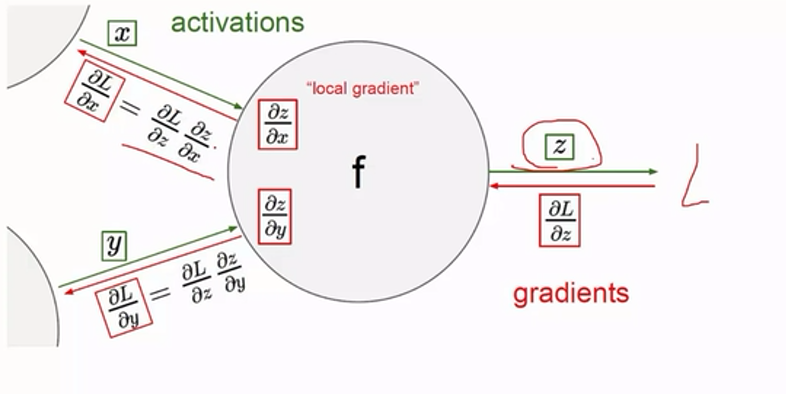
完成了1个Epoch就是迭代完一次一个数据集，两个就是两次

## 2.3反向传播

前向传播（计算loss值）->反向传播（参数怎么调节）->参数更新（调节参数）

Eg:



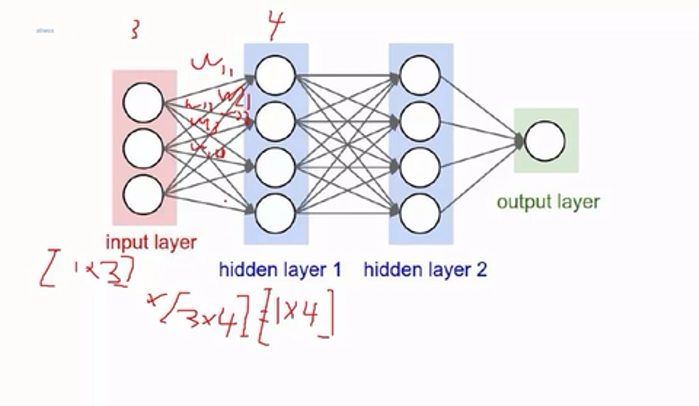


## 2.4神经网络基础架构

前向传播（计算loss值）->反向传播（参数怎么调节）->参数更新（调节参数W）

特征工程

全连接层、inputlayer->hidden layer1输入转为中间的特征如下

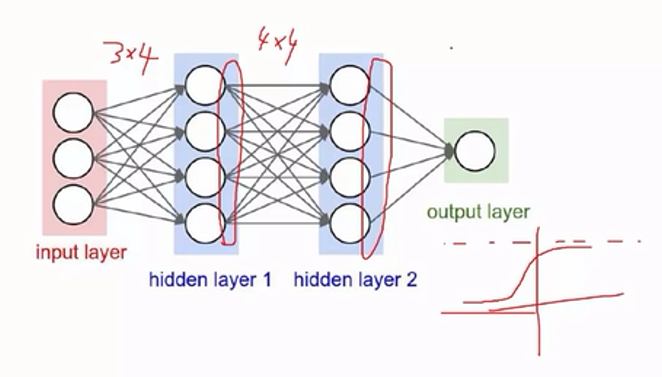


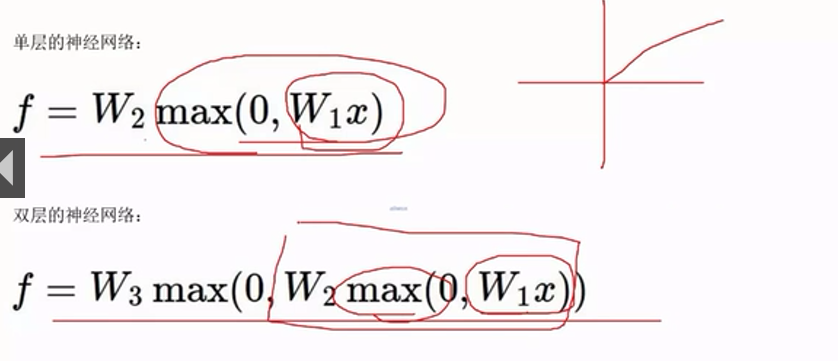
->hidden layer 1->Hidden layer2特征提取

！非线性，所以不能用结合律进行矩阵运算

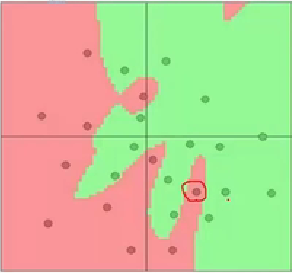
## 2.5正则化与激活函数

各层并不是直接相连而是经过激活函数如下图红圈位置，这样神经网络的效果才会好：



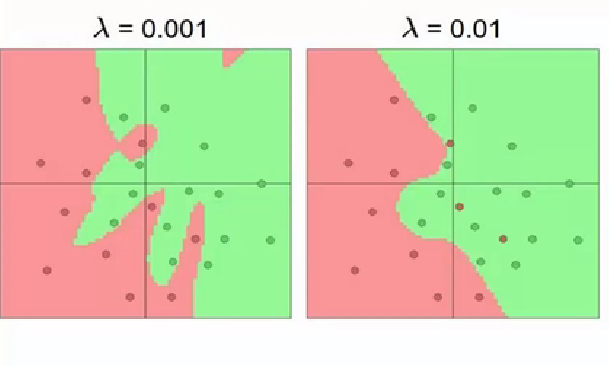


**过拟合**，当训练集毫无逻辑时也可以区分。在实际中，这样的网络效果很一般如图：（例子链接：）



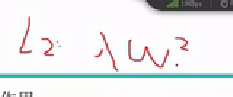
离群点也被分出

模型越平稳越好

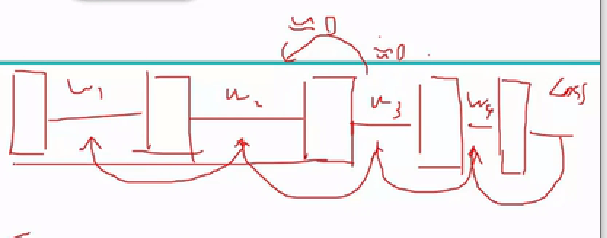


如图我们在实际中更倾向第二个，第一个会产生一会分为红一会分为绿的现象不稳定

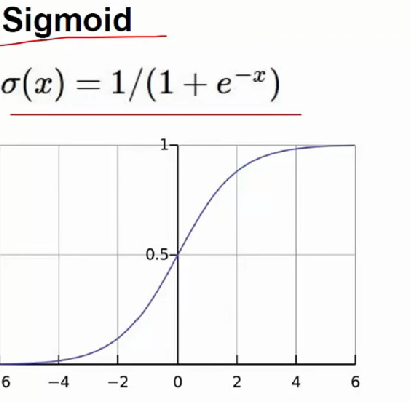
是一个惩罚项（正则化项）

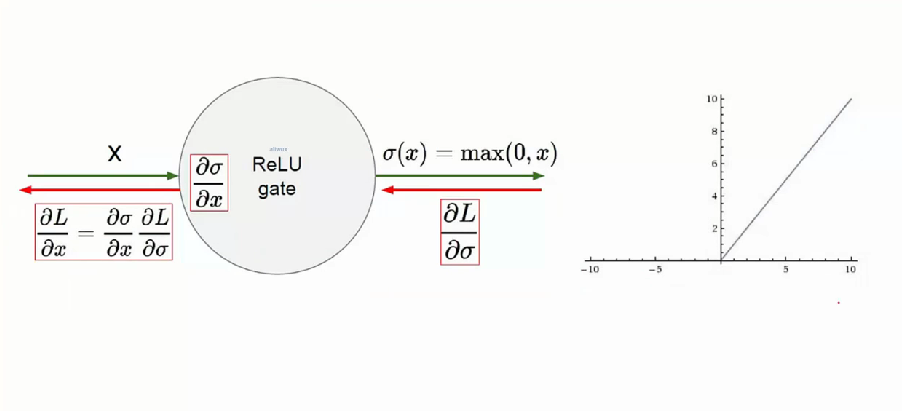


激活函数



一旦某一层梯度约等于零往回传的梯度就会约等于零W+=deltaW=W,参数就不更新了

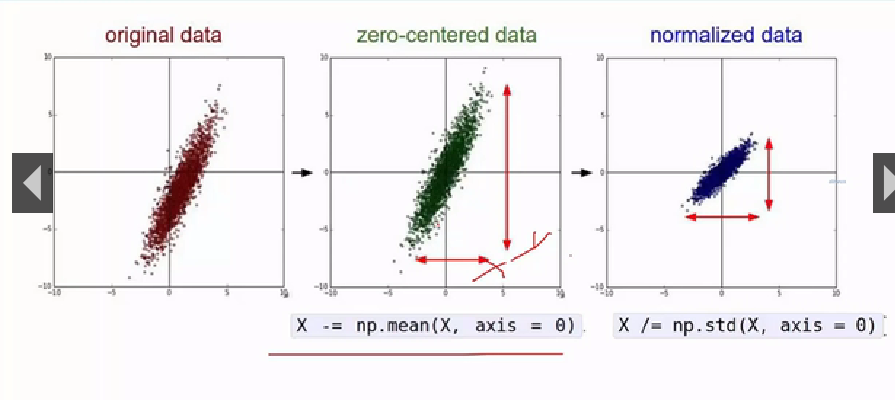
这个也是为什么sigmoid函数效果较差的原因



现在都用ReLUgate激活函数；优点：计算简单、梯度传播不会消失

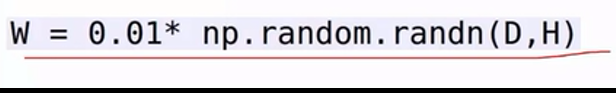
## 2.6drop-out

**数据预处理**



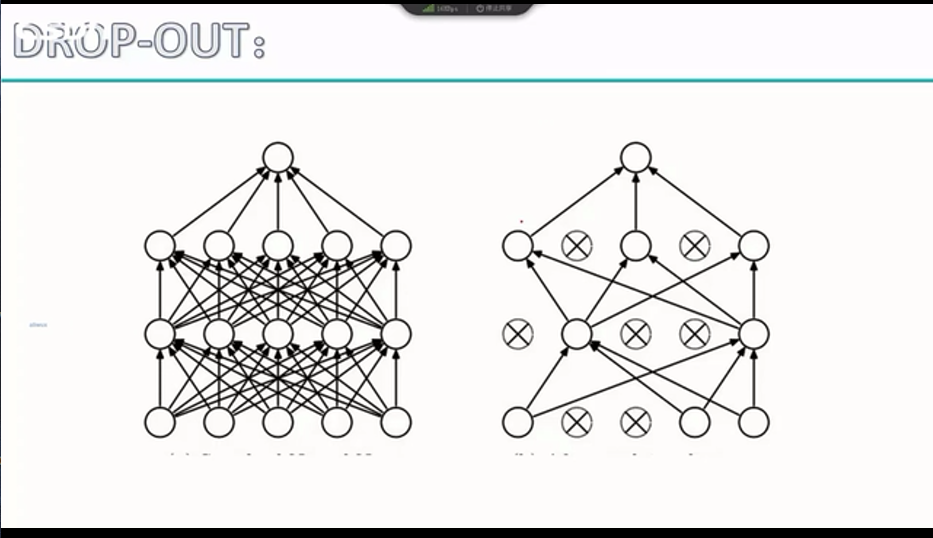
Originaldata不是零点为中心，做调整，减去均值；此时，x、y浮动差异大，除标准差。

**权重初始化**



随机初始化（高斯初始化）

**DROP-OUT**



为了防止过拟合，（原理：参数越少过拟合风险越低）训练的时候可不可以只用其中的几个神经元（随机），只在测试使用全部的。

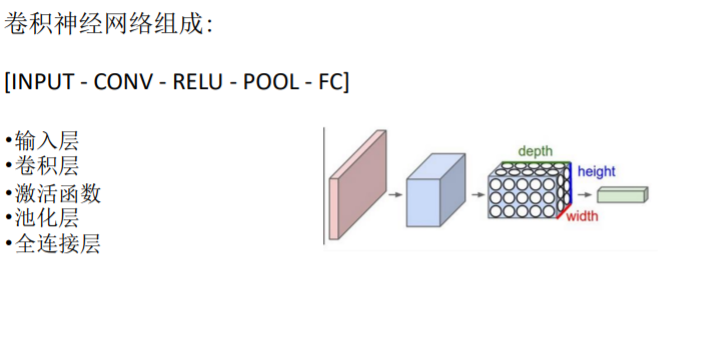
每次训练神经元保证固定比率的神经元留下

## 3.1tensorflow

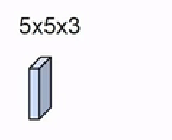
[TensorFlow Core (google.cn)](https://tensorflow.google.cn/tutorials)

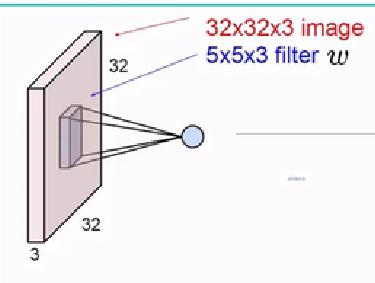
[快速入门教程 | NumPy 中文](https://www.numpy.org.cn/user/quickstart.html" \l "%E5%9F%BA%E6%9C%AC%E6%93%8D%E4%BD%9C)

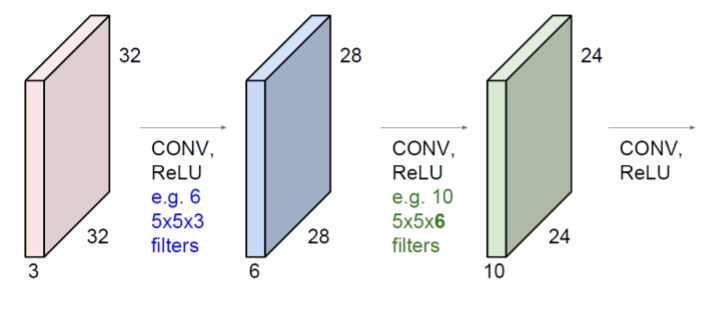
## 4.1卷积特征提取



**卷积层**：特征提取

：filter

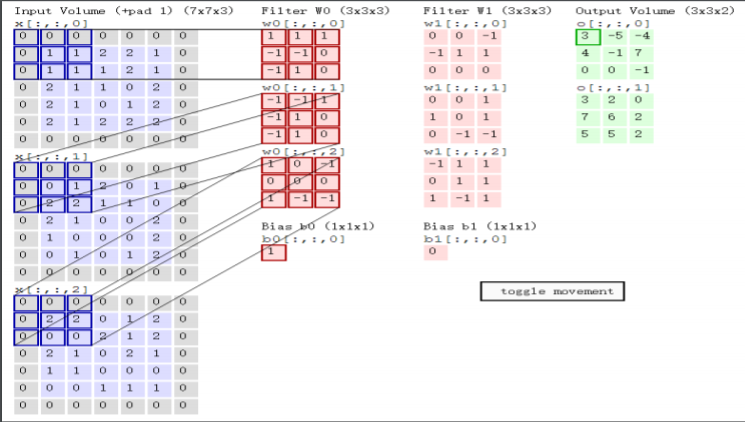
：1个filter提取一个特征



为什么从5\*5\*3->5\*5\*6?因为filter的深度与输入有关

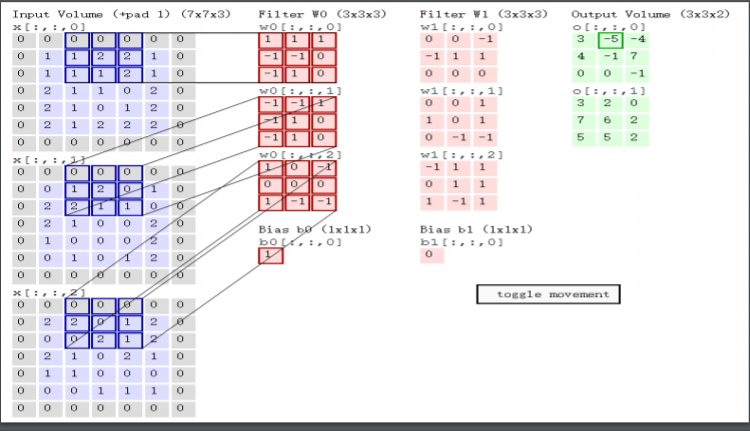
## 4.2卷积计算流程

目的：迭代优化filter



三个通道RGB

下一步



由此可见，步长为2

？为什么这里都乘相同的W0

[机器学习算法之——卷积神经网络（CNN）原理讲解 - InfoQ 写作平台](https://xie.infoq.cn/article/c4d846096c92c7dfcd6539075)

## 4.3卷积层计算参数

[机器学习算法之——卷积神经网络（CNN）原理讲解 - InfoQ 写作平台](https://xie.infoq.cn/article/c4d846096c92c7dfcd6539075)

## 4.4池化层操作

[机器学习算法之——卷积神经网络（CNN）原理讲解 - InfoQ 写作平台](https://xie.infoq.cn/article/c4d846096c92c7dfcd6539075)

## 4.5卷积网络基本架构

[咖啡|深度学习框架 (berkeleyvision.org)](http://caffe.berkeleyvision.org/)

[(26条消息) 机器学习实战 | 卷积神经网络详解(二)——自己手写一个卷积神经网络\_迈微AI研习社 · 号主-CSDN博客\_自己写一个卷积神经网络](https://blog.csdn.net/Charmve/article/details/106076844)

## 4.6经典网络架构

ALEXNET

VGG

## 5.1卷积网络复习

## 5.2CNN训练mnist数

## 5.3卷积与池化操作

## 5.4定义卷积网络计算流程

## 5.5完成迭代训练

## 5.6验证码识别描述

## 6.1自然语言处理与深度学





## 6.2语言模型

<https://blog.csdn.net/sun_brother/article/details/80350062>

<https://www.zhihu.com/tardis/landing/m/360/art/52061158>

https://blog.csdn.net/qq\_36696494/article/details/89028745?spm=1001.2101.3001.6650.2&utm\_medium=distribute.wap\_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7Edefault-2.wap\_blog\_relevant\_default&depth\_1-utm\_source=distribute.wap\_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7Edefault-2.wap\_blog\_relevant\_default

下采样：https://blog.csdn.net/stf1065716904/article/details/78450997

## 6.3神经网络模型

## 6.4CBOW模型

## 6.5参数更新

## 6.6负采样模型

## 7.1基于词袋模型训练分类器

## 7.2准备word2vec输入数据

## 7.3gensim构建word2

## 7.4Tfidf原理

## 7.5对抗生成网络原理概述

## 7.6GAN网络结构定义

## 7.7GAN迭代生成

## 7.8DCGAN网络特性

## 7.9DCGAN网络细节

## 8.1RNN网络架构

<https://blog.csdn.net/qq_39422642/article/details/78676567>

## 8.2LSTM网络架构