**Data Science Practice**

**Course Report**

**(数据科学实践课程报告)**

****

|  |  |
| --- | --- |
| **学 期：** | **2024秋** |
| **姓 名：** |  |
| **学 号：** |  |
| **专 业：** |  |
| **班 级：** |  |
| **序 号：** |  |
| **任课教师：** | **胡仕成** |
| **经济管理学院** | |

本实践课程的重要意义

本课程是数据科学基础的配套课程，在于巩固数据科学基础中各种算法的理论学习，并通过多个实际应用案例数据，锻炼运用这些算法用于解决实际问题的能力。同时，所学习的各种算法在今后的数据挖掘、机器学习、深度学习和GPT（generative pre-training transformer）中都会用到，熟练理解和运用这些算法，是今后进一步学习的关键。

本实践课程的主要内容

总的来说，本实践课程的算法包括模型参数的优化算法（连续优化方法）和算法的超参数的优化算法（离散优化方法）。

针对这些算法，本课程设计了20个实践应用案例，每个案例都提供了相应的数据集。学生需要按照每个实践所规定的步骤完成实验。

本实践课程实验分为必做(标记为P)和选做(标记为O)。

每个必做实验分值10分，满分200分，按50%折算为100分。

每个选做实验分值10分，满分130分，按10%折算为13分。

以上两部分之和为最终得分。

本实践课程的考核方式

本实践课程采用考核和抽查相结合的方式。

考核：学生完成电子实践报告，打印后提交纸质报告。

抽查：随机抽查学生，检查其实验程序。检查有问题的适度扣分，每个必做实验扣分最高不超过5分。

报告的格式：报告的字体、大小和格式，请参照哈工大本科生毕业设计论文规范。不合规范的将适度扣分，最高不超过10分。

# Experiments

实践课程打分表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 类型 | 算法 | 案例 | 任务 | 选做 | 必做得分 | 选做得分 |
| 一阶优化方法 | GDA | 基础 | P1 | **O1-4** |  |  |
| BGD | 线性 | P2 | **O2-4** |  |  |
| 非线性 | P3 | **O3-4** |  |  |
| SGD | 线性 | P4 | **O4-4** |  |  |
| 非线性 | P5 | **O5-4** |  |  |
| MBGD | 线性 | P6 | **O6-4** |  |  |
| 非线性 | P7 | **O7-4** |  |  |
| Adam | 线性 | P8 | **O8-4** |  |  |
| 非线性 | P9 | **O9-4** |  |  |
| 有约束的优化方法 | LMM | 等式约束 | P10 |  |  |  |
| 离散优化方法 | GA | TSP | P11 | **O11-4** |  |  |
| KP | P12 | **O12-4** |  |  |
| NSGAII | MV | P13 | **O13-4** |  |  |
| 线性分类器 | LC | 二分类 | P14 | **O14-5** |  |  |
| 多分类 | P15 |  |  |  |
| 图像分类 | P16 |  |  |  |
| 线性判别分析 | LDA | 二分类 | P17 |  |  |  |
| 多分类 | P18 |  |  |  |
| 主成分分析 | PCA | 二分类 | P19 |  |  |  |
| 多分类 | P20 |  |  |  |
| 实践任务得分合计 | | | |  | | |
| 格式扣分(最多扣10分) | | | |  | | |
| 抽查扣分 | | | |  | | |
| 总成绩最终得分 | | | |  | | |

# 一阶优化方法

本组实验是为了巩固各种基于梯度的优化算法。有两组数据：

**Grad-Line**：自变量和变量之间是线性关系，如公式(一)：

**Grad -NonL**：自变量和变量之间是非线性关系，如公式(二)：

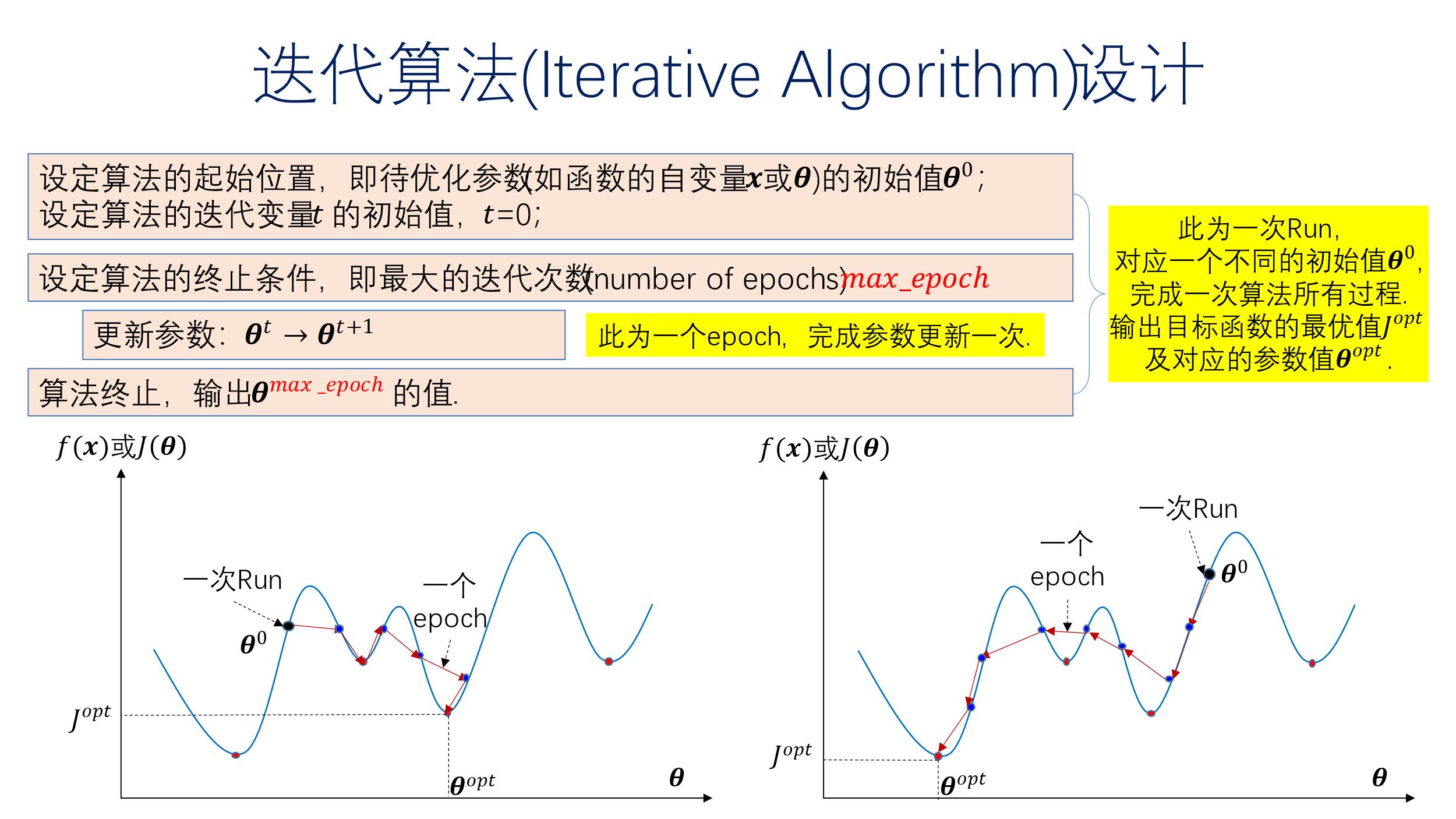
其中是一个非线性函数，常见形式有：

(1)

(2)

(3)

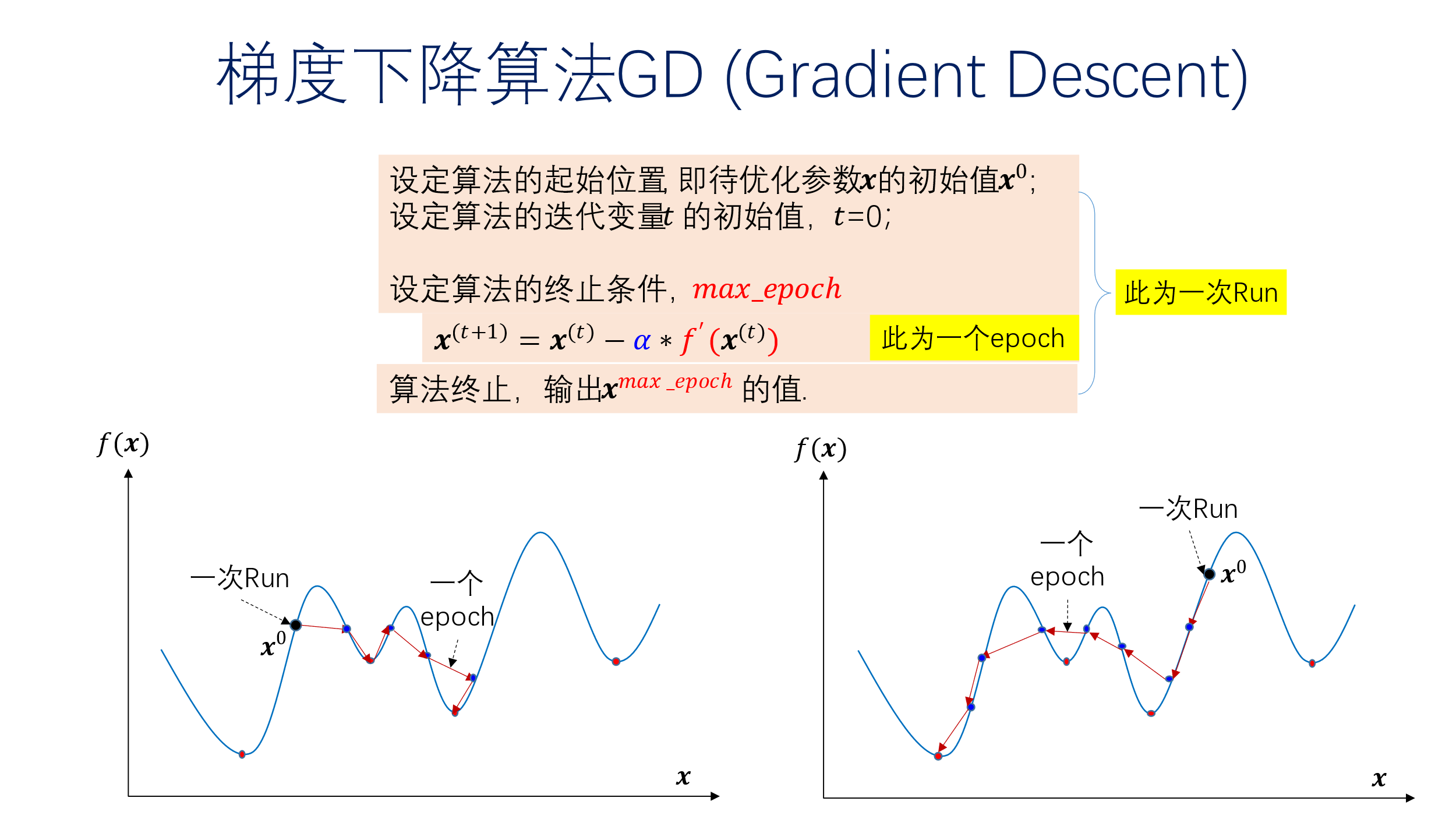
各种基于梯度的优化算法均是迭代算法，如下图。



对于迭代算法，需要分清两个概念：**epoch**和**Run**。

## 梯度下降算法GDA (Gradient Descent Algorithm)

梯度下降算法GDA的算法步骤如下图。



**P1**：设计GDA算法, 求以下函数的最小值, 用python实现。

=

要求：写出GDA算法步骤，并用Python完成如下实验：

**P1-1**求的最小值：设定算法的值，设定学习率值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的变量的最优值。

**P1-2**优化超参数：固定值，Run算法多次，每次Run：随机选定的初始值，根据网格法设定学习率的一个值。给出算法所求得的函数最小值随学习率变化的曲线图，根据该曲线确定的最优值。

**P1-3**求的全局最小值：固定值，设定学习率值为，Run算法多次，每次随机选定的初始值。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的参数的最优值。

**O1-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

## 批量梯度下降算法BGD (Batch Gradient Descent)

批量梯度下降算法是针对所有样本设计一个Loss函数：

然后采用GDA算法求的最小值。

**P2**：数据集为：**Grad-Line**。

设计BGD算法，估计模型参数的最优值，用python实现。

要求：写出BGD算法步骤，并用Python完成如下实验：

**P2-1**求的最小值：设定算法的值，设定学习率值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的参数的最优值。

**P2-2**优化超参数：固定值，Run算法多次，每次Run：随机选定的初始值，根据网格法设定学习率的一个值。给出算法所求得的函数最小值随学习率变化的曲线图，根据该曲线确定的最优值。

**P2-3**求的全局最小值：固定值，设定学习率值为，Run算法多次，每次随机选定的初始值。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的参数的最优值。

**O2-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

**P3**：数据集为：**Grad-NonL**。非线性函数选择第(1)种形式。

设计BGD算法，估计模型参数的最优值，用python实现。

要求：写出BGD算法步骤，并用Python完成如下实验：

**P3-1**求的最小值：设定算法的值，设定学习率值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的参数的最优值。

**P3-2**优化超参数：固定值，Run算法多次，每次Run：随机选定的初始值，根据网格法设定学习率的一个值。给出算法所求得的函数最小值随学习率变化的曲线图，根据该曲线确定的最优值。

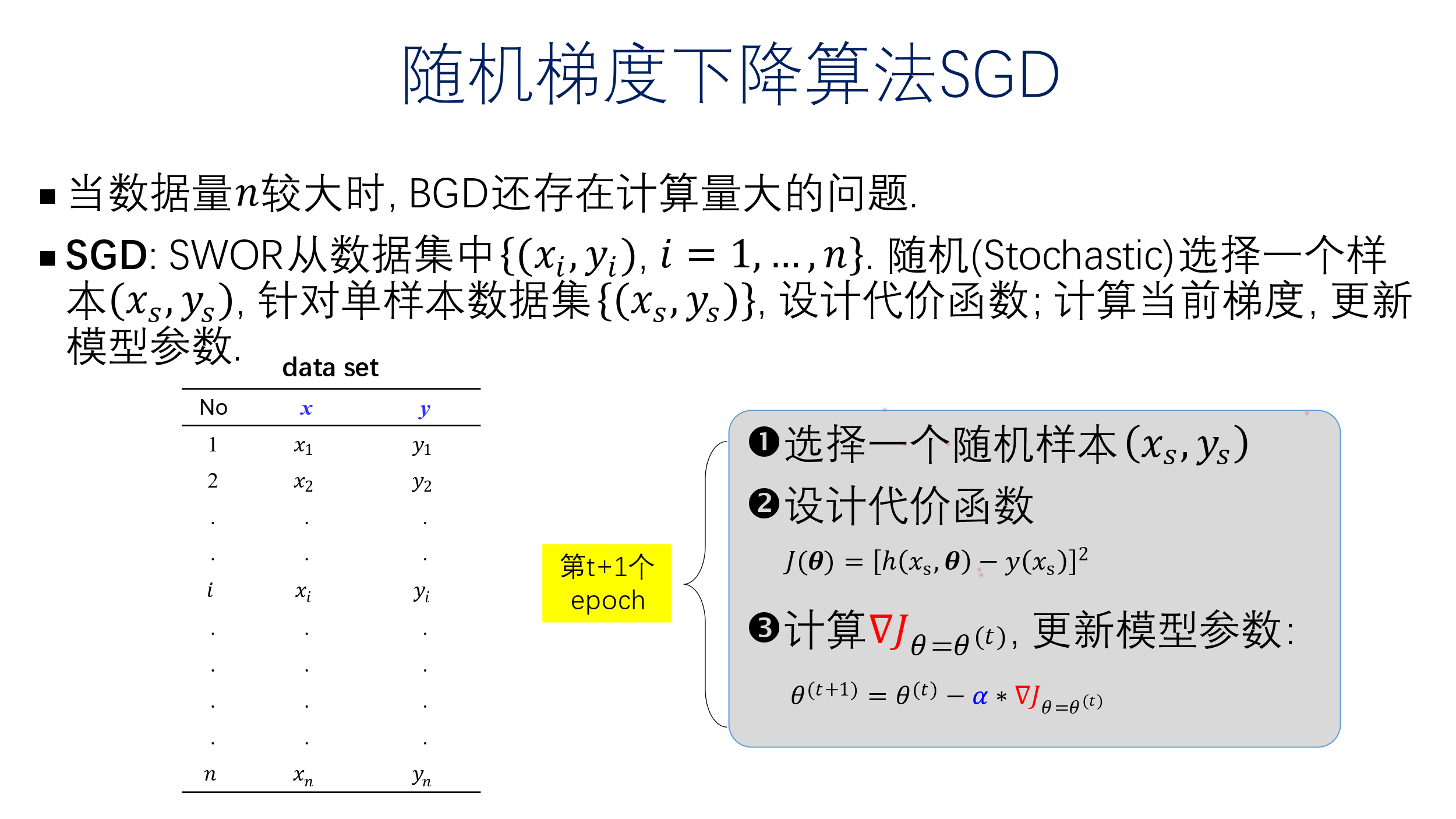
**P3-3**求的全局最小值：固定值，设定学习率值为，Run算法多次，每次随机选定的初始值。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的参数的最优值。

**O3-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

## 随机梯度下降算法SGD (Stochastic Gradient Descent)

随机梯度下降算法是针对随机选择的一个样本设计一个Loss函数：

针对该函数计算当前梯度, 更新模型参数。其第t+1个epoch的算法步骤如下图。



**P4**：数据集为：**Grad-Line**。

设计SGD算法，估计模型参数的最优值，用python实现。

要求：写出SGD算法步骤，并用Python完成如下实验：

**P4-1**求的最小值：设定算法的值，设定学习率值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的参数的最优值。

**P4-2**优化超参数：固定值，Run算法多次，每次Run：随机选定的初始值，根据网格法设定学习率的一个值。给出算法所求得的函数最小值随学习率变化的曲线图，根据该曲线确定的最优值。

**P4-3**求的全局最小值：固定值，设定学习率值为，Run算法多次，每次随机选定的初始值。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的参数的最优值。

**O4-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

**P5**：数据集为：**Grad-NonL**。非线性函数选择第(1)种形式。

设计BGD算法，估计模型参数的最优值，用python实现。

要求：写出BGD算法步骤，并用Python完成如下实验：

**P5-1**求的最小值：设定算法的值，设定学习率值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的参数的最优值。

**P5-2**优化超参数：固定值，Run算法多次，每次Run：随机选定的初始值，根据网格法设定学习率的一个值。给出算法所求得的函数最小值随学习率变化的曲线图，根据该曲线确定的最优值。

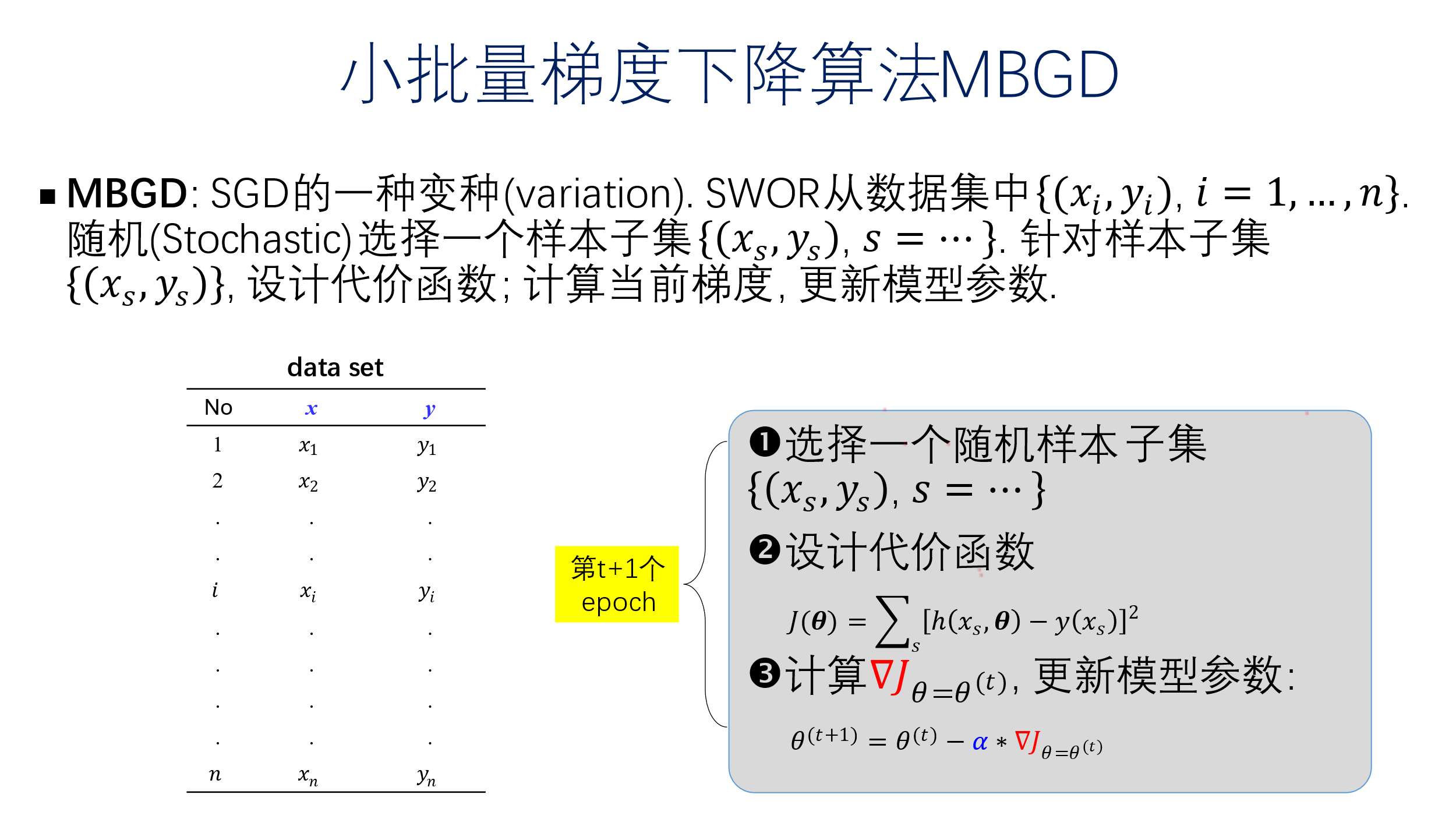
**P5-3**求的全局最小值：固定值，设定学习率值为，Run算法多次，每次随机选定的初始值。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的参数的最优值。

**O5-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

## 小批量梯度下降算法MBGD (Mini-Batch Gradient Descent)

小批量梯度下降算法是针对随机选择的一个样本子集设计一个Loss函数：

针对该函数计算当前梯度, 更新模型参数。其第t+1个epoch的算法步骤如下图。



**P6**：数据集为：**Grad-Line**。

设计MBGD算法，估计模型参数的最优值，用python实现。

要求：写出MBGD算法步骤，并用Python完成如下实验：

**P6-1**求的最小值：设定算法的值，设定学习率值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的参数的最优值。

**P6-2**优化超参数：固定值，Run算法多次，每次Run：随机选定的初始值，根据网格法设定学习率的一个值。给出算法所求得的函数最小值随学习率变化的曲线图，根据该曲线确定的最优值。

**P6-3**求的全局最小值：固定值，设定学习率值为，Run算法多次，每次随机选定的初始值。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的参数的最优值。

**O6-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

**P7**：数据集为：**Grad-NonL**。非线性函数选择第(1)种形式。

设计MBGD算法，估计模型参数的最优值，用python实现。

要求：写出MBGD算法步骤，并用Python完成如下实验：

**P7-1**求的最小值：设定算法的值，设定学习率值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的参数的最优值。

**P7-2**优化超参数：固定值，Run算法多次，每次Run：随机选定的初始值，根据网格法设定学习率的一个值。给出算法所求得的函数最小值随学习率变化的曲线图，根据该曲线确定的最优值。

**P7-3**求的全局最小值：固定值，设定学习率值为，Run算法多次，每次随机选定的初始值。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的参数的最优值。

**O7-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

## 适应性矩估计梯度下降算法Adam (Adaptive Moment)

批量梯度下降算法BGD、随机梯度下降算法SGD、小批量梯度下降算法MBGD对数据的使用方法（即在一个epoch中选用多少个样本用于计算梯度，可以选全部样本、一个随机样本或一个随机样本子集）适应性矩估计梯度下降算法。本实验采用小批量样本。超参数选用原作者推荐的值：= 0.9，= 0.999， = *=*

**P8**：数据集为：**Grad-Line**。

设计Adam算法，估计模型参数的最优值，用python实现。

要求：写出Adam算法步骤，并用Python完成如下实验：

**P8-1**求的最小值：设定算法的值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的参数的最优值。

**P8-2**求的全局最小值：固定值，Run算法多次，每次随机选定的初始值。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的参数的最优值。

**P8-3** 将原数据集按2：1比例分为两部分：训练集和测试集。针对测试集，重复**P8-2**；然后针对测试集计算准确率。

**O8-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

**P9**：数据集为：**Grad-NonL**。非线性函数选择第(1)种形式。

设计Adam算法，估计模型参数的最优值，用python实现。

要求：写出Adam算法步骤，并用Python完成如下实验：

**P9-1**求的最小值：设定算法的值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的参数的最优值。

**P9-2**求的全局最小值：固定值，Run算法多次，每次随机选定的初始值。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的参数的最优值。

**P9-3** 将原数据集按2：1比例分为两部分：训练集和测试集。针对测试集，重复**P9-2**；然后针对测试集计算准确率。

**O9-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

# 有约束的优化方法

## 拉格朗日乘子法LMM（Optimization problem with equal constraints）

**P10**：抛物线上的点集记为，抛物线上的点集记为，确定和上的点，使得二者之间的距离最短。

要求：按照以下步骤完成实验：

**P10-1** 根据以上问题建立优化模型，即确定优化目标和约束条件。

**P10-2** 采用拉格朗日乘子法LMM，将有约束的优化问题转化为无约束的优化问题。

**P10-3** 利用梯度下降算法GDA求解。

# 离散优化方法

## GA to single objective optimization problem (SOP)

**P11**：数据参见文件**GA-TSP**。针对旅行商问题TSP，采用GA求解。

要求：给出TSP的优化模型，设计GA算法步骤，并用Python完成如下实验：

**P11-1** 设置起点和终点，求TSP的最短路径。设定算法的值，设定其它的超参数值：群体规模**POP**、交叉概率**Pcross**、变异概率**Pmutate**、选择操作类型**Stype**。随机生成初始群体，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的变量的最优值。

**P11-2**优化超参数值（每次选择一个超参数进行优化）：固定值，Run算法多次，每次Run：随机生成初始群体，根据网格法设定超参数的一个值。给出算法所求得的函数最小值随超参数值变化的曲线图，根据该曲线确定超参数的最优值。

**P11-3**求TSP的全局最小值。固定值，设定所有超参数为其最优值。Run算法多次，每次随机生成初始群体。给出每次Run对应的目标函数最小值的散点图，输出的全局最小值及对应的变量的最优值。

**O11-4**探索求得函数全局更小值的其他思路和方法，给出实验过程和结果。

**P12**：数据参见文件**GA-KP**。针对背包问题KP，采用GA求解。

要求：给出KP的优化模型，设计GA算法步骤，并用Python完成如下实验：

**P12-1** 设计适应度函数（考虑在原目标函数加上约束不满足的惩罚项目，解决域外解问题），求KP的最大价值。设定算法的值，设定其它的超参数值：群体规模**POP**、交叉概率**Pcross**、变异概率**Pmutate**、选择操作类型**Stype**。随机生成初始群体，Run算法一次，给出目标函数值随变化的曲线图，输出的最大值及对应的变量的最优值。

**P12-2**优化超参数值（每次选择一个超参数进行优化）：固定值，Run算法多次，每次Run：随机生成初始群体，根据网格法设定超参数的一个值。给出算法所求得的函数最大值随超参数值变化的曲线图，根据该曲线确定超参数的最优值。

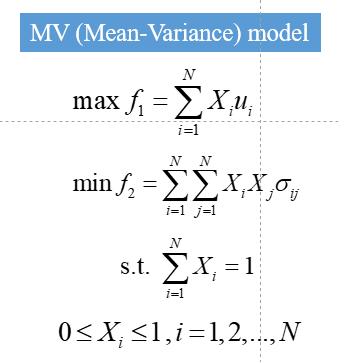
**P12-3**求KP的全局最大值。固定值，设定所有超参数为其最优值。Run算法多次，每次随机生成初始群体。给出每次Run对应的目标函数最大值的散点图，输出的全局最大值及对应的变量的最优值。

**O12-4**探索求得函数全局更大值的其他思路和方法，给出实验过程和结果。

## NSGAII to multiple objective optimization problem (MOP)

**P13**：数据参见文件 **NSGAII-MV**。针对投资组合优化问题的均值方差模型MV，采用NSGAII求解。

投资组合优化问题的均值方差模型MV如下：



指资产种类，指第种资产在一定时期内的平均收益率，指和之间的协方差。目标函数指综合收益，目标函数指综合风险。模型在于确定每种资产的投资比例，使得综合收益最大，风险最小。

要求：设计NSGAII算法步骤，并用Python完成如下实验：

**P13-1**求MV的Pareto最优前沿PF。设定算法的值，设定其它的超参数值：群体规模**POP**、交叉概率**Pcross**、变异概率**Pmutate**、选择操作类型**Stype**。随机生成初始群体，Run算法一次，给出PF随变化的曲线图，输出的最优值及对应的变量的最优值。

**P13-2**优化超参数值（每次选择一个超参数进行优化）：固定值，Run算法多次，每次Run：随机生成初始群体，根据网格法设定超参数的一个值。给出算法所求得的函数最优值（即最优PF）随超参数值变化的曲线图，根据该曲线确定超参数的最优值。

**P13-3**求MV的全局最优值。固定值，设定所有超参数为其最优值。Run算法多次，每次随机生成初始群体。给出每次Run对应的目标函数最优值（即最优PF）的散点图，输出的全局最优值（即最优PF）及对应的变量的最优值。

**O13-4**探索求得函数全局更优值的其他思路和方法，给出实验过程和结果。

# 线性分类器LC

**P14**：二分类问题，数据参见文件**LC-Raisin**。将数据集随机分为5：2：3三部分，分别作为训练集、验证集和测试集。采用线性分类器LC实现二分类，损失函数定义为多类支持向量机(SVM)损失。

要求：采用小批量梯度下降算法MBGD，并用Python完成如下实验：

**P14-1**训练：针对训练集，估计LC模型参数的最优值。设定算法的值，设定学习率值，随机选定的初始值，Run算法一次，给出目标函数值随变化的曲线图，输出的最小值及对应的参数的最优值。

**P14-2**评价：针对验证集，评估LC模型的性能。通常为计算LC的准确率。

**P14-3**超参数优化：对算法的学习率值进行优化。通常采用网格优化法：重新选择一个不同的学习率值，重复**P14-1**训练和**P14-2**评价两个步骤。遍历所有可能的值后，给出不同值对应准确率的散点图，确定学习率的最优值及对应的模型参数的最优值。

**P14-4**测试：针对测试集，计算LC的准确率。

**O14-5**探索线性分类器LC准确率更高的其他思路和方法，给出实验过程和结果。

**P15**：多分类问题，数据参见文件**LC-UKM**。完成**P14**的试验任务。

**P16**：图像分类，数据参见文件**LC-iris**。完成**P14**的试验任务。鸢尾花(**iris**)是常用的图像分类实验数据。一副图像的每个像素点的像素值作为LC的一个输入特征值，如果图像是20\*20的，则需要设计400个输入特征。输出则是图像的类别，**LC-iris**数据集有三个类别（setosa、versicolour、virginica）。

# 线性判别分析LDA

**P17**：二类问题，数据参见文件 **LC-Raisin**（同线性分类器LC中的数据）。

要求：采用LDA进行降维，并用Python完成如下实验：

**P17-1** 采用LDA进行降维，降维后将各类别数据拟合为高斯分布，以中垂线作为分类界线，计算分类的准确率。

**P17-2** 采用LDA进行降维，对降维前后的数据集采用线性判别器LC进行分类，计算分类的准确率和。

**P17-3** 比较以上3个准确率，针对结果进行分析。

**P18**：多类问题，数据参见文件 **LC-UKM**（同线性分类器LC中的数据）。

要求：采用LDA进行降维，并用Python完成如下实验：

**P18-1** 采用LDA进行降维。

**P18-2**对降维前后的数据集采用线性判别器LC进行分类，计算分类的准确率和。

**P18-3** 比较以上2个准确率，针对结果进行分析。

# 主成分分析PCA

**P19**：二类问题，数据参见文件 **LC-Raisin**（去掉其中的类别列）。

要求：采用PCA进行降维，并用Python完成如下实验：

**P19-1** 采用PCA进行降维。

**P19-2**对降维前后的数据集采用线性判别器LC进行分类，计算分类的准确率和。

**P19-3** 比较4个准确率：、、和，针对结果进行分析。

**P20**：多类问题，数据参见文件 **LC-UKM**（去掉其中的类别列）。

要求：采用PCA进行降维，并用Python完成如下实验：

**P20-1** 采用PCA进行降维。

**P20-2**对降维前后的数据集采用线性判别器LC进行分类，计算分类的准确率和。

**P20-3** 比较4个准确率：、、和，针对结果进行分析。