**云南大学数学与统计学院上机实验报告 （ 6 ）**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **实验课程名** | | | 运筹学实验 | | | | | | | **成绩** | |  | |
| **学号** |  | | **姓名** | **枫叶** | | **专业** | | **统计学** | **年级** | | **2021级** | | |
| **实验项目名称** | | | 动态规划—生产与存储问题R软件求解 | | | **日期** | | **2023.06.5** | **实验时间** | | | | **2023.06.5** |
| **指导教师** | | 潘东东 | | | **实验地址（室）** | | **格物楼3507** | | | | | | |
| **教师评语** | |  | | | | | | | | | | | |

**一、目的**

使用动态规划的思想和方法编程求解如下的生产与存储问题。

某工厂要对一种产品制定今后四个时期的生产计划，据估计在今后四个时期内，市场对于该产品的需求量如下表所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 时期(k) | 1 | 2 | 3 | 3 |
| 需求量(dk) | 2 | 3 | 2 | 4 |

假定该厂生产每批产品的固定成本为3千元，若不生产就为0；每单位产品成本为1千元；每个时期生产能力所允许的最大生产量不超过6个单位；每个时期末未能售出的产品，每单位需付存储费0.5千元。还假定在第一个时期的初始库存量为0，第四个是期末的库存量也为0。试问该厂应如何安排各个时期的生产与库存，才能在满足市场需要的条件下，使总成本最小。

**二、实验内容**

在R软件下编写程序求解动态规划-生产与存储问题。

**三、使用环境**

R

**四、实验步骤**

**1、数据准备：见题目**

**2、程序实验：**

# 每阶段单位数量产品的储存费为c

C=0.5

# 每阶段开工的固定成本费为a

A = 3;

# 生产单位数量产品的成本费为b

B = 1;

# 市场对该产品的需求量数组

ds <- c(2, 3, 2, 4);

# 市场对该产品的需求量函数

d <- function(k) {

  return (ds[k])};

# 生产目标,所有初始化为NA，在每次遍历的时候，需要更新这个值

us <- rep(NA, times=length(ds));

u <- function(k) {

  return (us[k]);

}

# 每阶段开始时的储存量x（x)

x <- function(k) {

  if(k <= 1) {

    result <- 0;

  } else {

    result <- x(k-1) + u(k-1) - d(k-1);

  }

  return (result);

}

# 阶段的生产成本和储存费之和

vs <- c();

v <- function(k) {

  if(u(k) == 0) {

    vs[k] <- C\*x(k);

  }

  if(u(k)>0) {

    vs[k] <- C\*x(k) + (A + B\*u(k));

  }

  return (vs[k]);

}

f <- function(k) {

  # 设置终止条件，如果为第k+1季度，则返回

  #f(x(k))=min(v(x(k), u(k)) + f(x(k+1)))

  if(k == (length(ds)+1)) {

    result <- 0;

  } else {

    result <- v(k) + f(k+1);

  }

  return (result);

}

uss <- expand.grid(s1=c(0:sum(ds)), s2=c(0:sum(ds)), s3=c(0:sum(ds)), s4=c(0:sum(ds)));

uss <- data.matrix(uss[which(rowSums(uss)==sum(ds)),]);

#i<- 2

#uss[which(rowSums(matrix(uss[,1:i], ncol=i))>=sum(ds[1:i])),];

for(i in 1:length(ds)) {

  uss <- uss[which(rowSums(matrix(uss[,1:i], ncol=i)) >= sum(ds[1:i])),];

}

rs = c();

for(i in 1:nrow(uss)) {

  # 重新设置us值

  us <- uss[i,];

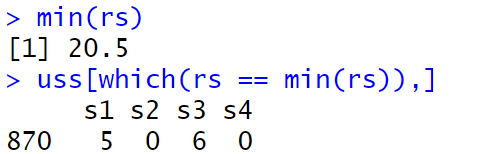
  rs[i] <- f(1);

}

min(rs)

uss[which(rs == min(rs)),]

**3、程序结果：**



1. **实验结果及分析**

最小成本为20.5，各时期分别生产5,0,6,0单位