Лекция 4. Задачи нелинейного и динамического программирования

Вопросы лекции:

1. **Модели нелинейного программирования**
2. **Задачи динамического программирования**

Вопрос 1. Модели нелинейного программирования

Задачами нелинейного программирования называются задачи математического программирования, в которых нелинейны и (или) целевая функция, и (или) ограничения в виде неравенств или равенств.

Общая постановка задачи формулируется следующим образом: необходимо найти такое решение $X=(x_1, x_2, ..., x_n)$ системы (являющееся оптимальным решением, или оптимальным планом), обеспечивающие достижение экстремума (максимума или минимума) целевой функции (нелинейной)задачи

$$Z(X) = f(x_1, x_2, ..., x_n) \rightarrow \max\ (\min) \tag{1}$$

и удовлетворяющие системе ограничений (уравнений и неравенств)

$$\begin{cases} \varphi_i(x_1, x_2, ...x_n) = 0, \quad i = 1,2,..,l \\ \varphi_i(x_1, x_2, .., x_n) \le 0, \quad i = l+1, l+2, ..., m \end{cases} \tag{2}$$

Подробнее

Например, задачи о максимизации выпуска продукции

$$P = a_o \times F_1^{a_1} \times F_2^{a_2} \times \ldots \times F_n^{a_n} . \rightarrow \max \tag{3}$$

Где P – производственная функция (объем производства), F_i – расход i–го фактора производства.

Издержки зависят от расхода всех факторов и от цен этих факторов (c_i)

Совокупные издержки выражаются формулой

$$b = c_1F_1 + c_2F_2 + \cdots + c_nF_n \tag{4}$$

(Требуется при данных совокупных издержках определить такое количество факторов производства, которое максимизирует объем продукции

Многие задачи нелинейного программирования могут быть приближены к задачам линейного программирования и найдено близкое к оптимальному решение.

Рассмотрим выпуклые задачи нелинейного программирования, у которых область допустимых ограничений и целевая функция являются выпуклыми или вогнутыми. Это, например, задачи квадратичного программирования, в которых целевая функция и (или) ограничения являются функциями своих аргументов, в степени не выше второй. Для подобных моделей локальный экстремум обязательно является и глобальным экстремумом.

Подробнее

Например, Имеется три вида возможных вложений (например, ценных бумаг или вкладов), каждый из которых характеризуется определенной эффективностью и определенным риском:

Таблица 1 – Характеристики видов инвестирования

Вид инвестирования	*эффективность*	*риск*
1	*4*	*10*
2	*10*	*40*
3	*40*	*80*

Не склонный к риску инвестор считает необходимым составить инвестиционный портфель с минимальным риском (так называемый *портфель Марковица (минимального риска) и доходностью не менее 15 ед. Для этого нужно определить такие удельные веса x_{ij} каждого вида инвестиций в портфеле, чтобы свести к минимуму риск и сохранить доходность на заданном уровне.*

Построим экономико-математическую модель.

Целевая функция (минимум совокупного риска портфеля из трех видов инвестиций) согласно формуле Марковица имеет вид:

$$f(x_1, x_2, x_3) = \sqrt{100x_1^2 + 1600x_2^2 + 6400x_3^2} \to min \qquad (5)$$

Ограничения модели:

Доходность не ниже 15 ед.:

$$4x_1 + 10x_2 + 40x_3 \geq 15 \qquad (6)$$

Сумма долей всех видов инвестиций равна 1:

$$x_1 + x_2 + x_3 = 1 \qquad (7)$$

Условие неотрицательности:

$$x_j \geq 0,\ j = 1, 2, 3. \qquad (8)$$

Решение задач нелинейного программирования принципиально ничем не отличается от решения задач линейного и целочисленного программирования. Единственное отличие заключается в том, что при установке параметров поиска решения в диалоговом окне «Параметры поиска решения», необходимо снять галочку в строке «Линейная модель».

Реализуя приведенную модель средствами пакета Excel, получим оптимальный портфель Марковица:

$$x_1 = 0{,}5213,\ x_2 = 0{,}2078,\ x_3 = 0{,}2709, \qquad (9)$$

т.е. доли ценных бумаг оказались равными 52,13 %; 20,78 % и 27,09 %. При этом минимальный риск - 23,79, доходность портфеля оказалась равной заданной - 15.

Вопрос 2. Задачи динамического программирования

В некоторых экономических задачах необходимо учитывать изменение моделируемого процесса во времени и влияние времени на критерий оптимальности. Для решения указанных задач используется **метод динамического планирования** (программирования).

Данный метод представляет собой способ решения сложных задач путём разбиения их на более простые подзадачи.

Динамическое программирование исходит из следующих свойств задачи:

- перекрывающиеся подзадачи (решаются отдельные части задачи (подзадачи), после чего решения подзадач объединяются в одно общее решение);
- оптимальная подструктура (оптимальное решение подзадач меньшего размера может быть использовано для решения исходной задачи);
- возможность запоминания решения часто встречающихся подзадач (каждая подзадача решается один раз, что сокращает количество вычислений).

В основе динамического программирования лежит **принцип оптимальности Беллмана**, заключающийся в следующем: управление на каждом шаге надо выбрать так, чтобы оптимальной была сумма выигрышей на всех оставшихся до конца процесса шагах, включая выигрыш на данном шаге.

Последовательность (алгоритм) метода динамического программирования можно представить следующим образом:

1. Определяются параметры, характеризующие состояние управляемой системы перед каждым шагом (s_i).

2. Определятся возможный набор шаговых управлений u_i для каждого шага и налагаемые на них ограничения (условия осуществления управления).

3. Определяется выигрыш от применения управления u_i, на i шаге, если перед этим система была в состоянии s, т.е. функция выигрыша:

$$W_i = f_i(s, u_i). \tag{10}$$

4. Определяется, как изменяется состояние s системы S при реализации управления u_i на i шаге, то есть новое состояние системы:

$$s' = \square_i(s, u_i). \tag{11}$$

5. Формулируется основное рекуррентное уравнение динамического программирования, выражающее условный оптимальный выигрыш $W_i(s)$ (начиная с i шага и до конца) через функцию $W_{i+1}(s)$:

$$W_i(s) = \max \{f_i(s,u_i) + W_{i+1}(\square_i(s, u_i))\}. \tag{12}$$

6. Осуществляется условная оптимизация последнего шага (шага m), исходя из ряда состояний s, из которых можно за один шаг дойти до конечного состояния, вычисляя для каждого из них условный оптимальный (максимальный) выигрыш:

$$W_m(s) = \max \{f_m(s,u_i)\}. \tag{13}$$

В процессе вычисления находится то условное оптимальное управление $u_m(s)$, для которого достигается максимум выигрыша.

7. Производится условная оптимизация (m-1) шага согласно формуле основного рекуррентного уравнения динамического программирования, полагая в ней i=(m-1), (m-2).Для каждого из этих шагов указывается условное оптимальное управление $u_i(s)$, при котором максимум выигрыша достигается. На первом шаге состояние системы не изменяется:

$$W^* = W_1(s_o). \tag{14}$$

8. Осуществляется безусловная оптимизация управления: при найденном оптимальном управлении на первом шаге $u_1^* = u_1(s_o)$ изменяют состояние системы согласно основному рекуррентному уравнению динамического программирования; для вновь найденного состояния находят оптимальное управление на втором шаге u_2^* и так далее до конца.

Основное рекуррентное уравнение динамического программирования составляется для каждой конкретной задачи и вид его определяется выбором исходных данных, видом целевой функции и ограничений, числом варьируемых параметров.

Подробнее

Например, данным методом можно решить задачу распределения ресурсов между объектами (мероприятиями, вкладами и т.п.) Пусть имеется определенный размер средств (ресурсов) К, который должен быть распределен между m объектами O_1, O_2, ... O_m. Каждый из объектов O_i при вложении в него каких-то средств в размере x приносит доход, зависящий от x, т. е. представляющий собой функцию $f_i(x)$. Все функции $f_i(x)$ при i=1,…m известны. Необходимо таким образом распределить средства К между объектами, чтобы в сумме они дали максимальный доход.

Эта задача решается методом динамического программирования. Операцию распределения средств следует представить в виде последовательности, считая за первый шаг вложение средств в объект O_1, за второй – в объект O_2 и так далее. Управляемая система S в данном случае – это средства (ресурсы), которые распределяются между объектами. Состояние системы S перед каждым шагом характеризуется одним числом Q - имееющимся запасом еще не вложенных средств. В этой задаче "шаговыми управлениями" являются средства x_1, x_2, ... x_m, выделяемые объектам. Требуется найти оптимальное управление, т. е. такую совокупность значений объема средств x_1, x_2, ... x_m, при которой суммарный доход максимален.

Вопросы для самопроверки:

1. Приведите пример задачи нелинейного программирования

2. Поясните разницу между задачами линейного и нелинейного программирования

3. Что такое динамическое программирование

4. Приведите примеры практических задач, которые можно решить с помощью динамического программирования