



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

"МИРЭА - Российский технологический университет"

РТУ МИРЭА

Институт Информационных Технологий

Кафедра Вычислительной Техники

ПРАКТИЧЕСКАЯ РАБОТА №1

по дисциплине

«Теория принятия решений»

Метод Парето

Студент группы: ИКБО-04-22

Заковряшин Н.М.
(Ф. И.О. студента)

Преподаватель

Железняк Л.М.
(Ф.И.О. преподавателя)

Москва 2024

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 МЕТОД ПАРЕТО	3
1.1 Выбор Парето-оптимального множества	4
1.2 Указание верхних/нижних границ критериев.	6
1.3 Субоптимизация	6
1.4 Лексикографическая оптимизация	7
1.5 Результаты работы программы.....	8
ЗАКЛЮЧЕНИЕ	9
СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ	10

ВВЕДЕНИЕ

Целью данной практической работы является ознакомление с методом многокритериальной оптимизации Парето и изучение методов сужения оптимального множества альтернатив, включая метод верхних/нижних границ, метод субоптимизации и лексикографический метод оптимизации. Важные понятия в этой области включают: лицо, принимающее решение (ЛПР), альтернативы, критерии и исходы управленческих решений. Оптимальность по Парето определяется как состояние, при котором невозможно улучшить один критерий без ухудшения другого, формируя множество Парето из неулучшаемых или эффективных точек. Подходы к многокритериальной оптимизации включают выделение множества Парето и выбор наиболее предпочтительного варианта на основе дополнительных критериев или логического рассуждения.

1 МЕТОД ПАРЕТО

Алгоритм Парето начинается с определения набора альтернатив и критериев, по которым эти альтернативы будут оцениваться. Суть метода заключается в поиске таких решений, которые невозможно улучшить по одному критерию, не ухудшив при этом по другому. Для этого, изначально идентифицируется множество всех возможных решений. Затем, среди этих решений выделяются те, которые являются оптимальными по Парето, то есть такие, для которых не найдется другого решения, превосходящего их одновременно по всем критериям. Эти точки формируют Парето— множество всех эффективных решений задачи. После определения множества Парето, задача лица, принимающего решение, заключается в выборе наиболее подходящего решения из этого множества на основе дополнительных предпочтений или критериев, которые могут быть сформулированы уже после анализа полученного множества оптимальных решений. Таким образом, метод Парето позволяет сузить область поиска оптимального решения, исключив

заведомо неэффективные варианты, и сосредоточиться на анализе только потенциально наилучших решений.

1.1 Выбор Парето-оптимального множества

Поставить задачу и расписать ее: альтернативы, критерии и стремления, рассчитать множество Парето

Определение альтернатив: Каждый вариант завтрака является альтернативой для выбора.

Критерии оценки: Питательная ценность, калорийность на порцию, время приготовления, доступность и цена

Стремления: Цель состоит в том, чтобы найти завтрак, который имеет высокую питательную ценность и калорийность, низкое время приготовления, высокую доступность и низкую цену.

Таблица 1. Альтернативы и критерии

№	Варианты завтрака	Критерии				
		Калорийность на порцию (+)	Питательная ценность (1-5) (+)	Время приготовления (мин) (-)	Доступность (1-3) (+)	Цена (1-5) (-)
1	Овсянка с фруктами и орехами	3	5	25	3	2
2	Гречневая каша с овощами	3	3	20	3	2
3	Яичница с овощами	3	4	15	3	2
4	Смузи из зелени и фруктов	2	3	5	3	1
5	Творожная запеканка с ягодами	3	3	30	2	3
6	Тосты с авокадо и яйцом	4	4	10	2	4
7	Кускус с овощами и фетой	3	4	15	2	3
8	Йогурт с мюсли и свежими фруктами	2	3	4	3	2
9	Бутерброды с лососем	4	4	7	2	5

10	Бутерброд с сыром на козьем молоке	2	2	7	1	4
----	------------------------------------	---	---	---	---	---

Таблица 2. Сравнение альтернатив

	1	2	3	4	5	6	7	8	9	10
1	X	X	X	X	X	X	X	X	X	X
2	X	X	X	X	X	X	X	X	X	X
3	X	A3	X	X	X	X	X	X	X	X
4	н	н	н	X	X	X	X	X	X	X
5	A1	A2	A3	н	X	X	X	X	X	X
6	н	н	н	н	н	X	X	X	X	X
7	н	н	A3	н	A7	н	X	X	X	X
8	н	н	н	н	н	н	н	X	X	X
9	н	н	н	н	н	н	н	н	X	X
10	н	н	н	A4	н	н	н	A8	н	X

Таблица 3. Парето-оптимальное множество

№	Варианты завтрака	Критерии				
		Калорийность на порцию (+)	Питательная ценность (1-5) (+)	Время приготовления (мин) (-)	Доступность (1-3) (+)	Цена (1-5) (-)
1	Овсянка с фруктами и орехами	3	5	25	3	2
2	Гречневая каша с овощами	3	3	20	3	2
3	Яичница с овощами	3	4	15	3	2
4	Смузи из зелени и фруктов	2	3	5	3	1

7	Кускус с овощами и фетой	3	4	15	2	3
8	Йогурт с мюсли и свежими фруктами	2	3	4	3	2

1.2 Указание верхних/нижних границ критериев.

Установим для приведенного примера верхнюю и нижнюю границу. Калорийность порции не менее 3, питательная ценность не менее 4, время приготовления не более 15 минут

Таблица 4. Результат верхних и нижних границ критериев

№	Варианты завтрака	Критерии				
		Калорийность на порцию (+)	Питательная ценность (1-5) (+)	Время приготовления (мин) (-)	Доступность (1-3) (+)	Цена (1-5) (-)
3	Яичница с овощами	3	4	15	3	2
6	Тосты с авокадо и яйцом	4	4	10	2	4
7	Кускус с овощами и фетой	3	4	15	2	3
9	Бутерброды с лососем	4	4	7	2	5

1.3 Субоптимизация

Субоптимизацию производят следующим образом: выделяют один из критериев, а по всем остальным критериям назначают нижние границы. Оптимальным при этом считается исход, максимизирующий выделенный критерий на множестве исходов, оценки которых по остальным критериям не ниже назначенных.

Выберем главный критерий: **Время приготовления.**

Установим верхние/нижние границы для остальных критериев:

Калорийность не менее 3, питательная ценность не менее 4, цена не более 2

Таблица 1.5. Результат субоптимизации

№	Варианты завтрака	Критерии				
		Калорийность на порцию (+)	Питательная ценность (1-5) (+)	Время приготовления (мин) (-)	Доступность (1-3) (+)	Цена (1-5) (-)
3	Яичница с овощами	3	4	15	3	2
1	Овсянка с фруктами и орехами	3	5	25	3	2

1.4 Лексикографическая оптимизация

Упорядочим критерии по их относительной важности:

- 1) Время приготовления
- 2) Калорийность на порцию
- 3) Доступность
- 4) Цена
- 5) Питательная ценность

Таблица 1.6. Результат лексикографического

№	Варианты завтрака	Критерии				
		Калорийность на порцию (+)	Питательная ценность (1-5) (+)	Время приготовления (мин) (-)	Доступность (1-3) (+)	Цена (1-5) (-)
8	Йогурт с мюсли и свежими фруктами	2	3	4	3	2

1.5 Результаты работы программы

Таблица Парное сравнение альтернатив:

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
A1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
A2	н	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
A3	н	A3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
A4	н	н	н	0.0	0.0	0.0	0.0	0.0	0.0	0.0
A5	A1	A2	A3	н	0.0	0.0	0.0	0.0	0.0	0.0
A6	н	н	н	н	н	0.0	0.0	0.0	0.0	0.0
A7	н	н	A3	н	A7	н	0.0	0.0	0.0	0.0
A8	н	н	н	н	н	н	н	0.0	0.0	0.0
A9	н	н	н	н	н	н	н	н	0.0	0.0
A10	н	н	н	A4	н	н	н	A8	н	0.0

Рисунок 1. Таблица попарного сравнения альтернатив

Результат указания верхней/нижней границы: (Калорийность ≥ 3 , Питательная ценность ≥ 4), Время приготовления ≤ 15

	Калорийность	Питательная ценность	Время приготовления	Доступность	Цена
A3	3	4	15	3	2
A6	4	4	10	2	4
A7	3	4	15	2	3

Рисунок 2. Таблица указания верхней и нижней границы

Результат отбора вариантов, удовлетворяющих заданным критериям: (главный критерий: Время приготовления, Питательная ценность ≥ 4 , Цена ≤ 2)

	Калорийность	Питательная ценность	Время приготовления	Доступность	Цена
A3	3	4	15	3	2
A1	3	5	25	3	2

Рисунок 3. Таблица субоптимизации

Результат лексикографической оптимизации: (Самая важная: Цена)

	Калорийность	Питательная ценность	Время приготовления	Доступность	Цена
A8	2	3	4	3	2

Рисунок 4. Таблица лексикографической оптимизации

ЗАКЛЮЧЕНИЕ

В ходе практической работы №1 по "Теории принятия решений" мы изучили и использовали метод Парето для решения задачи выбора лучшего завтрака, учитывая разные критерии. Мы показали, что метод Парето помогает уменьшить количество вариантов, выбирая только те, которые оптимальны, что облегчает процесс выбора.

Также мы рассмотрели дополнительные методы для уточнения выбора, включая установление верхних и нижних границ для критериев, субоптимизацию и лексикографическую оптимизацию. Эти методы позволили нам более точно отфильтровать и упорядочить варианты, делая процесс выбора еще проще.

В итоге, мы нашли идеальный вариант завтрака, который соответствует всем нашим критериям и предпочтениям. Это подтверждает, что метод Парето и связанные с ним подходы к выбору из множества вариантов действительно работают эффективно.

СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ

1. Болотова Л. С. Многокритериальная оптимизация. Болотова Л. С., Сорокин А. Б. [Электронный ресурс] / Метод. указания по вып. курсовой работы — М.: МИРЭА, 2015.
2. Сорокин А. Б. Методы оптимизации: гибридные генетические алгоритмы. Сорокин А. Б. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2016.
3. Сорокин А. Б. Линейное программирование: практикум. Сорокин А. Б., Бражникова Е. В., Платонова О. В. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2017.

ПРИЛОЖЕНИЯ

Приложение А – Код реализации метода Парето на языке Python.

Приложение А

Код реализации метода Парето на языке Python.

Листинг А – Реализация Парето.

```
import pandas as pd
import numpy as np

data = {
    'Варианты завтрака': [
        'Овсянка с фруктами и орехами', 'Гречневая каша с овощами',
        'Яичница с овощами', 'Смузи из зелени и фруктов',
        'Творожная запеканка с логодами', 'Тосты с авокадо и лимон',
        'Кускус с овощами и фетой', 'Йогурт с мясли и свежими
фруктами',
        'Бутерброд с лососем', 'Бутерброд с сыром на козьем молоке'
    ],
    'Калорийность на порцию': [3, 3, 3, 2, 3, 4, 3, 2, 4, 2],
    'Питательная ценность': [5, 3, 4, 3, 3, 4, 4, 3, 4, 2],
    'Время приготовления': [25, 20, 15, 5, 30, 10, 15, 4, 7, 7],
    'Доступность': [3, 3, 3, 3, 2, 2, 2, 3, 2, 1],
    'Цена': [2, 2, 2, 1, 3, 4, 3, 2, 5, 4]
}

pd.set_option('display.max_columns', None)
pd.set_option('display.width', 200)

df = pd.DataFrame(data)
save_df = df
print(df)

for i in range(20): print()

def alternatives(alt1, alt2, directions):
```

```

        for i in range(len(directions)):
            if (directions[i] and alt1.iloc[i] < alt2.iloc[i]) or (not
directions[i] and alt1.iloc[i] > alt2.iloc[i]):
                return False
        return True

def generate_comparison_df(altered_df, criteria_directions):
    comparison_matrix = np.full((10, 10), None)
    for i in range(10):
        for j in range(i):
            if alternatives(altered_df.iloc[i, 1:], altered_df.iloc[j,
1:], criteria_directions):
                comparison_matrix[i, j] = 'A' + str(i + 1)
            elif alternatives(altered_df.iloc[j, 1:],
altered_df.iloc[i, 1:], criteria_directions):
                comparison_matrix[i, j] = 'A' + str(j + 1)
            else:
                comparison_matrix[i, j] = 'H'
    comparison_df = pd.DataFrame(comparison_matrix,
                                columns=['A1', 'A2', 'A3', 'A4',
'A5', 'A6', 'A7', 'A8', 'A9', 'A10'])
    comparison_df.index = ['A1', 'A2', 'A3', 'A4', 'A5', 'A6', 'A7',
'A8', 'A9', 'A10']

    return comparison_df

criteria_directions = [True, True, False, True, False]
comparison_df = generate_comparison_df(df, criteria_directions)
print(comparison_df)

for i in range(20): print()

print(df[(df['Калорийность на порцию'] >= 3) & (df['Питательная
ценность'] >= 4) & (df['Время приготовления'] <= 15)])

for i in range(20): print()

```

```
df = df[(df['Калорийность на порцию'] >= 3) & (df['Питательная
ценность'] >= 4) & (df['Цена'] <= 2)]

df = df.sort_values(['Время приготовления'])

print(df)

for i in range(20): print()

df = save_df.sort_values(['Время приготовления', 'Калорийность на
порцию', 'Доступность', 'Цена', 'Питательная ценность'],
                        ascending=[False, True, True, False, True])

print(df.iloc[::-1])

for i in range(20): print()
```