



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-15-22

Оганнисян Г.А.
(Ф. И.О. студента)

Преподаватель

Железняк Л.М.
(Ф.И.О. преподавателя)

Москва 2024

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 МЕТОД ПАРЕТО.....	4
1.1 Выбор Парето-оптимального множества.....	5
1.2 Указание верхних/нижних границ критериев.	7
1.3 Субоптимизация.....	8
1.4 Лексикографическая оптимизация.....	9
1.5 Результаты работы программы.....	10
ЗАКЛЮЧЕНИЕ	11
СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ	12
ПРИЛОЖЕНИЯ.....	13

ВВЕДЕНИЕ

Закон Парето, также известный как принцип Парето или правило 80/20, это эмпирическое наблюдение, сформулированное итальянским экономистом и социологом Вильфредо Парето в конце 19 века и развившееся в начале 20 века. Этот закон описывает неравномерное распределение ресурсов и результатов в различных областях деятельности.

Принцип закона Парето можно сформулировать следующим образом: во многих сферах жизни, примерно 80% результатов обусловлены примерно 20% причин. То есть, малая доля причин порождает большую долю результатов, а большая доля причин — малую долю результатов.

Закон Парето был формально сформулирован и впервые опубликован в работе "Руководство к политической экономии", изданной Парето в 1909 году. В этой работе Парето утверждал, что примерно 80% богатства в Италии принадлежит примерно 20% населения. Принцип Парето был впервые сформулирован Джозефом Джураном в 1950-х годах и стал широко распространен в области менеджмента качества и управления бизнесом.

1 МЕТОД ПАРЕТО

Эффективность по Парето означает, что система находится в таком состоянии, где невозможно улучшить один аспект системы, не ухудшив при этом какой-либо другой аспект.

Предположим, что у нас есть множество операций, обозначенное как A . Каждая операция, обозначенная как a , характеризуется двумя числовыми параметрами: $M(a)$ и $R(a)$, где, например, M представляет собой доход, а R - риск. Отличие между различными операциями заключается, как минимум, в одном из этих параметров. При выборе оптимальной операции предпочтительно, чтобы доход (M) был максимальным, а риск (R) - минимальным.

Определим, что операция a доминирует над операцией b , обозначая a превосходит b ($a \succ b$), если $M(a) > M(b)$ и $R(a) < R(b)$, и хотя бы одно из этих неравенств строгое. При этом операция a считается доминирующей, а операция b - доминируемой.

Таким образом, операция, которая доминируется, не может быть наилучшей. Следовательно, для поиска наилучшей операции необходимо обращаться к недоминируемым операциям. Набор недоминируемых операций называется множеством Парето или множеством оптимальности по Парето.

На множестве Парето каждая из характеристик M , R — однозначная функция другой.

1.1 Выбор Парето-оптимального множества

Задача заключается в выборе концерта группы Twice с использованием Парето-оптимального множества решений. Необходимо применить методы для сужения множества ради выявления исключительного варианта. Для начала работы выбирается 9 альтернатив, имеющие следующие критерии: цена билета, удалённость от моей страны проживания, стоимость поездки на концерт, количество билетов, длительность концерта.

Для работы с методом Парето выбирается предметная область «Выбор Барбершопа». Проанализировав разные барбершопы, выбрали несколько вариантов. Они были сведены в таблицу 1.

№	Вариант решений	Критерии			
		Средний чек (руб.) (-)	Удалённость локации (км) (-)	Количество услуг (+)	Рейтинг (от 1 до 5) (+)
A1	OldBoy	2 300	4,30	16	4,80
A2	Метод	1 800	2,30	17	5
A3	FIDEL	1 700	2,70	20	4,90
A4	Чёрная кость	1 200	1,60	13	4,40
A5	Бритый Ёж	800	9,30	8	3,80
A6	БородаВайб	1 950	2	16	5
A7	Чёлочка	500	11,10	4	2,70
A8	Бритва	2 600	2,30	22	4,90
A9	Baradach	1 400	5,80	16	4,80
A10	BomboKlak	3 500	3,20	19	4,70

Таблица 1.1

Примечание: знаком (-) указывается отрицательное стремление критерия (чем меньше, тем лучше), а знаком (+) – положительное (чем больше, тем лучше).

Было определено, что оптимизация по Парето использует отношение Парето-доминирования, которое отдаёт предпочтение одному объекту перед другим только в том случае, когда первый объект по всем критериям не хуже второго и хотя бы по одному из них лучше. При истинности этого условия первый объект считается доминирующим, а второй – доминируемым. Два

объекта, для которых предпочтение хотя бы по одному критерию расходится, считаются несравнимыми.

Сравним попарно все альтернативы и сведем их в таблицу 1.2

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

Таблица 1.2

Парето-оптимальное множество определено альтернативами {A2, A3, A6, A8}.

1.2 Указание верхних/нижних границ критериев.

Установим для приведенного примера верхнюю границу: средний чек не больше 1700 руб., удалённость локации меньше 4,30 км. Согласно данным условиям табл.1 трансформируется в табл.1.3.

№	Вариант решений	Критерии			
		Средний чек (руб.) (-)	Удалённость локации (км) (-)	Количество услуг (+)	Рейтинг (от 1 до 5) (+)
A2	Метод	1 800	2,30	17	5
A3	FIDEL	1 700	2,70	20	4,90
A6	БородаВайб	1 950	2	16	5
A8	Бритва	2 600	2,30	22	4,90
A10	BomboKlak	3 500	3,20	19	4,70

Таблица 1.3

Варианты, удовлетворяющие этим дополнительным ограничениям: {2, 3, 6, 8, 10}; из них оптимальными по Парето является варианты 2, 3, 6, 8.

Вывод: основной недостаток метода состоит в том, что оптимальное решение становится здесь субъективным или вообще может отсутствовать, так как зависит, во-первых, от величин назначаемых верхних/нижних границ критериев и, во-вторых, от окончательного выбора, совершаемого принимающим решение. Однако данный метод позволяет поставить ограничения по тем критериям, которые непосредственно относятся к интереса субъекта.

1.3 Субоптимизация

Субоптимизацию производят следующим образом: выделяют один из критериев, а по всем остальным критериям назначают нижние границы.

Пусть в примере главным критерием выступает количество услуг; ограничения: средний чек от 1200 руб. и больше; рейтинг больше 4,6; удалённость локации меньше 2,7 км. Отбросим варианты, которые не удовлетворяют данным ограничениям и составим табл. 1.4.

№	Вариант решений	Критерии			
		Средний чек (руб.) (-)	Удалённость локации (км) (-)	Количество услуг (+)	Рейтинг (от 1 до 5) (+)
A2	Метод	1 800	2,30	17	5
A6	БородаВайб	1 950	2	16	5
A8	Бритва	2 600	2,30	22	4,90

Таблица 1.4

Из табл. 4 видно, остаются варианты {2, 6, 8}. Из них минимальный средний чек имеет вариант 2 (Метод). Этот вариант и будет оптимальным.

Вывод: выделение одного из критериев, а также указание нижних границ для остальных критериев основано на дополнительной, получаемой от ЛПР. Следовательно, окончательное решение здесь также имеет субъективный характер. Между тем это все еще непосредственно выражает интересы ЛПР.

1.4.Лексикографическая оптимизация

Упорядочим критерии в примере по относительной важности, например, следующим образом: важнейший критерий – количество услуг, следующий за ним по важности – удалённость локации. Результат по лексикографической оптимизации, показан в таблице 1.5.

№	Вариант решений	Критерии			
		Средний чек (руб.) (-)	Удалённость локации (км) (-)	Количество услуг (+)	Рей-тинг (от 1 до 5) (+)
A8	Бритва	2 600	2,30	22	4,90

Таблица 1.5 -

Вывод: таким образом, наглядно проявляется недостаток лексикографической оптимизации – фактический учет одного (важнейшего) критерия. Например, в последнем случае в качестве оптимального выступает вариант 8, который предлагает всего один сеанс опираясь на критерий выбора – «количество услуг».

1.5 Результаты работы программы

Результат лексикографической оптимизации: (Самая важная: Количество услуг)				
	Средний чек (руб.)	Удалённость локации (км)	Количество услуг	Рейтинг (от 1 до 5)
A8	0.000385	0.434783	22	4.9

Рисунок 1.5.4 – Лексикографическая оптимизация.

ЗАКЛЮЧЕНИЕ

На основе метода Парето был осуществлён выбор концерта группы Twice. Проведено сравнение попарно всех альтернатив, установлена для приведенного примера верхняя граница критериев и осуществлена субоптимизация. Среди преимуществ метода Парето можно выделить способность выявлять наиболее эффективные и важные факторы. Путем анализа и классификации данных по их важности, метод помогает сосредоточить усилия на ключевых аспектах и улучшить результаты, что в свою очередь способствует оптимизации использования ресурсов. Однако следует отметить, что метод Парето упрощает реальность, предполагая равномерное распределение влияния факторов, что не всегда соответствует сложной реальности. Также метод фокусируется на наиболее важных факторах, что может привести к упущению дополнительных аспектов, оказывающих влияние на конечные результаты. В целом, метод Парето полезен как инструмент для выявления ключевых аспектов и принятия решений.

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.1. Реализация Парето.

```
import pandas as pd
import numpy as np

df = pd.DataFrame({
    'Средний чек (руб.):': [2300, 1800, 1700, 1200, 800, 1950, 500, 2600, 1400, 3500],
    'Удалённость локации (км)': [4.30, 2.30, 2.70, 1.60, 9.30, 2, 11.10, 2.30, 5.80, 3.20],
    'Количество услуг': [16, 17, 20, 13, 8, 16, 4, 22, 16, 19],
    'Рейтинг (от 1 до 5)': [4.80, 5, 4.90, 4.40, 3.80, 5, 2.70, 4.90, 4.80, 4.70]
})

df.index = ['A1', 'A2', 'A3', 'A4', 'A5', 'A6', 'A7', 'A8', 'A9', 'A10']
print(df)

df['Средний чек (руб.)'] = 1 / df['Средний чек (руб.)']
df['Удалённость локации (км)'] = 1 / df['Удалённость локации (км)']

# Создаем массив для хранения результатов попарного сравнения
arr1 = np.zeros((10, 10), dtype=object)

# Попарное сравнение альтернатив
for i in range(10):
    for j in range(i + 1, 10):
        arr = df.iloc[i].values >= df.iloc[j].values # Сравнение значений всех столбцов для
альтернативы i и j
        check = all(arr) # Проверка, что все критерии для i лучше или равны j
        arr2 = df.iloc[i].values <= df.iloc[j].values # Сравнение значений всех столбцов для
альтернативы i и j
        check2 = all(arr2) # Проверка, что все критерии для i хуже или равны j
        # Запись результата в массив arr1
        if check:
            arr1[j, i] = 'A' + str(i + 1)
        elif check2:
            arr1[j, i] = 'A' + str(j + 1)
        else:
            arr1[j, i] = 'H'

# Создаем новый DataFrame для результатов попарного сравнения
df_ = pd.DataFrame(arr1, columns=['A1', 'A2', 'A3', 'A4', 'A5', 'A6', 'A7', 'A8', 'A9', 'A10'],
                    index=['A1', 'A2', 'A3', 'A4', 'A5', 'A6', 'A7', 'A8', 'A9', 'A10'])

print("\n Таблица Попарное сравнение альтернатив: ")
print(df_)

print("\n Вывод парето-оптимальных альтернатив")
print(df.iloc[[1, 2, 5, 7]])

df['Средний чек (руб.)'] = 1 / df['Средний чек (руб.)']
df['Удалённость локации (км)'] = 1 / df['Удалённость локации (км)']
```

```
print("\n Результат указания верхней/нижней границы: ('Средний чек (руб.)' >= 1700 ,
'Удалённость локации (км)' < 4.30)")
print(df[(df['Средний чек (руб.)'] >= 1700) & (df['Удалённость локации (км)'] < 4.30)])

print(
    "\n Результат отбора вариантов, удовлетворяющих заданным критериям: главный критерий: \n
    Средний чек (руб.) >= 1200, Рейтинг (от 1 до 5) >= 4.6, Удалённость локации (км) < 2.70")
print(df[(df['Средний чек (руб.)'] >= 1200) & (df['Рейтинг (от 1 до 5)'] >= 4.6) & (df['Удалённость
локации (км)'] < 2.70)])

df['Средний чек (руб.)'] = 1 / df['Средний чек (руб.)']
df['Удалённость локации (км)'] = 1 / df['Удалённость локации (км)']

def lex_optimization(df):
    max_crit = df['Количество услуг'].max() # Нахождение максимального значения в столбце
    'Количество услуг'
    optimal_df = df[df['Количество услуг'] == max_crit] # Фильтрация данных по максимальному
    значению количества услуг

    if len(optimal_df) == 1: # Проверка, если найден только один оптимальный вариант
        return optimal_df

    next_crit = optimal_df['Удалённость локации (км)'].max() # Нахождение следующего критерия -
    максимального значения удалённости локации
    optimal_df = optimal_df[optimal_df['Удалённость локации (км)'] == next_crit] # Дополнительная
    фильтрация данных по максимальному значению удалённости локации

    return optimal_df

result = lex_optimization(df)

print("\n Результат лексикографической оптимизации: (Самая важная: Количество услуг)")
print(result)
```