

Министерство образования Российской Федерации
МОСКВОСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. Н.Э.БАУМАНА

Факультет: Информатика и системы управления (ИУ)
Кафедра: Информационная безопасность (ИУ8)

МЕТОДЫ ОПТИМИЗАЦИИ

Лабораторная работа №5 на тему:
«Игры с природой». Критерии принятия решений.».

Вариант – 1

Преподаватель:
Коннова Н.С.

Студент:
Александров А. Н.

Группа:
ИУ8-34

Москва, 2020

Цель работы:

Ознакомиться с основными понятиями «игр с природой»; получить навыки применения различных критериев (Бернулли, Вальда, максимума, Сэвиджа) для выбора стратегии в условиях полной неопределённости.

Постановка задачи:

Найти стратегии игрока при реализации гипотез недостаточного основания (Бернулли), пессимизма (Вальда), оптимизма, смешанной (Гурвица) при $\alpha = 0,5$, рисков (Сэвиджа). Рекомендовать выбор стратегии согласно принципу большинства. (строки матриц соответствуют стратегиям игрока, столбцы – состояниям природы):

$$\begin{pmatrix} 1 & 11 & 12 & 11 \\ 7 & 5 & 7 & 7 \\ 16 & 6 & 13 & 2 \\ 9 & 9 & 16 & 13 \\ 17 & 18 & 15 & 7 \end{pmatrix}$$

Ход решения:

Для решения задачи, воспользовался собственной программой, написанной на языке программирования **Python**(см. Приложение А):

Файл **best_strategy.py**, содержащий функции для нахождения оптимальных стратегий согласно критериям:

- функция ***BernulliCriteria(matrix: np.array)***: критерий недостаточного основания (Бернулли);
- функция ***ValdCriteria(matrix: np.array)***: критерий пессимизма (Вальда);
- функция ***OptimismCriteria(matrix: np.array)***: критерий авантюры (максимума, оптимизма);
- функция ***GurvitzCriteria(matrix: np.array)***: критерий Гурвица (смешанный);
- функция ***SevigeCriteria(matrix: np.array)***: критерий рисков (Севиджа).

Каждая из функций принимает на вход матрицу стратегий и возвращает оптимальную стратегию, а функция ***ChooseBestStrategy(matrix: np.array)*** отбирает по принципу большинства стратегию, оказавшуюся наилучшей в большем числе критериев.

Итак, матрица стратегий, описывающая «игру с природой», имеет вид:

Таблица 1 Матрица стратегий.

Стратегии	b_1	b_2	b_3	b_4
a_1	1	11	12	11
a_2	7	5	7	7
a_3	16	6	13	2
a_4	9	9	16	13
a_5	17	18	15	7

Критерий Бернулли (принцип недостаточного основания).

Все состояния природы предполагаем равновероятными. Тогда стоит руководствоваться стратегией с максимальной величиной математического ожидания. Для каждой стратегии она определяется как

$$\Psi_i = \frac{1}{4} \sum_{j=1}^5 a_{ij}.$$

```

"Игры с природой". Критерии принятия решений.
Матрица стратегий
+-----+-----+-----+-----+
| Стратегии | b1 | b2 | b3 | b4 |
+-----+-----+-----+-----+
| a1 | 1.0 | 11.0 | 12.0 | 11.0 |
| a2 | 7.0 | 5.0 | 7.0 | 7.0 |
| a3 | 16.0 | 6.0 | 13.0 | 2.0 |
| a4 | 9.0 | 9.0 | 16.0 | 13.0 |
| a5 | 17.0 | 18.0 | 15.0 | 7.0 |
+-----+-----+-----+-----+
1) Критерий недостаточного основания Бернулли.
Если пользоваться критерием Бернулли, то следует руководствоваться стратегией a5.
Соответствующее математическое ожидание выигрыша при этом максимально и равно 14.25.
+-----+-----+-----+-----+
| Стратегии | b1 | b2 | b3 | b4 |  $\Psi_i$  |
+-----+-----+-----+-----+
| a1 | 1.0 | 11.0 | 12.0 | 11.0 | 8.75 |
| a2 | 7.0 | 5.0 | 7.0 | 7.0 | 6.5 |
| a3 | 16.0 | 6.0 | 13.0 | 2.0 | 9.25 |
| a4 | 9.0 | 9.0 | 16.0 | 13.0 | 11.75 |
| a5 | 17.0 | 18.0 | 15.0 | 7.0 | 14.25 |
+-----+-----+-----+-----+

```

Рисунок 1 Критерий Бернулли.

Оптимальной является стратегия a_5 с соответствующим максимальным математическим ожиданием $\Psi_5 = 14,5$.

Критерий Вальда (пессимистический).

В соответствии с этим критерием стоит выбрать стратегию с наименьшей вероятностью проигрыша и доставит максимальную прибыль. То есть критерий есть ни что иное, как нижняя цена игры:

$$\max_i \min_j a_{ij}.$$

2) Критерий пессимизма Вальда.
Пессимистическая стратегия (Вальда) определяет выбор a_4 (нижняя цена игры равна 9.0).

Стратегии	b1	b2	b3	b4	α_i
a1	1.0	11.0	12.0	11.0	1.0
a2	7.0	5.0	7.0	7.0	5.0
a3	16.0	6.0	13.0	2.0	2.0
a4	9.0	9.0	16.0	13.0	9.0
a5	17.0	18.0	15.0	7.0	7.0

Рисунок 2 Критерий Вальда.

Оптимальной является стратегия a_4 . Нижняя цена игры равна 9.

Критерий максимума (оптимистический).

Выбираем наиболее авантюристическую стратегию с максимальным выигрышем:

$$\max_i \max_j a_{ij}.$$

Максимальный выигрыш 18 достигается при стратегии a_5 .

Критерий Гурвица.

Определим вероятность своего «везения» как $\alpha = 0,5$. Тогда критерий определяется из условия равновероятной реализации пессимистической и оптимистической гипотез:

$$\max_i (0,5 \min_i a_{ij} + 0,5 \max_j a_{ij}).$$

4) Критерий Гурвица.
 $\alpha = 0.5$

Стратегии	b1	b2	b3	b4	Ψ
a1	1.0	11.0	12.0	11.0	6.5
a2	7.0	5.0	7.0	7.0	6.0
a3	16.0	6.0	13.0	2.0	9.0
a4	9.0	9.0	16.0	13.0	12.5
a5	17.0	18.0	15.0	7.0	12.5

Наилучшая стратегия a4
Ожидаемый выигрыш: 12.5

Рисунок 3 Критерий Гурвица

Наилучшая стратегия a_4 , ожидаемый выигрыш равен 12,5.

Критерий Сэвиджа (критерий рисков).

Заполним таблицу рисков по формуле $r_{ij} = \max_i a_{ij} - a_{ij}$ (столбец $\alpha_i = \max_j r_{ij}$):

5) Критерий Севиджа.

Составим таблицу рисков стратегий:

Стратегии	b1	b2	b3	b4	α_i
a1	16.0	7.0	4.0	2.0	16.0
a2	10.0	13.0	9.0	6.0	13.0
a3	1.0	12.0	3.0	11.0	12.0
a4	8.0	9.0	0.0	0.0	9.0
a5	0.0	0.0	1.0	6.0	6.0

Таким образом, оптимальная рисковая стратегия – a5.

Рисунок 4 Критерий Сэвиджа.

Тогда оптимальная стратегия соответствует минимальному значению в столбце α_i . Оптимальная рисковая стратегия a_5 .

В заключение, по принципу большинства, следует рекомендовать стратегию a_5 .

В итоге выберем стратегию, которая оказалась оптимальной в большем числе критериев:

a1	a2	a3	a4	a5
0	0	0	2	3

По принципу большинства рекомендуем стратегию a5

Рисунок 5 Выбор рекомендуемой стратегии.

Вывод:

В данной работе я познакомился с основными понятиями «игр с природой»; получил навыки применения критериев стратегии в условиях неопределённости.

По результатам оптимальными стратегиями стали:

- по критерию Бернулли: a_5 ;
- по критерию Вальда: a_4 ;
- по критерию максимума: a_5 ;
- по критерию Гурвица: a_4 ;
- по критерию Сэвиджа: a_5 .

По принципу большинства была рекомендована стратегия a_5 .

Приложение А

Код программы

Файл «main.py»

```
# Copyright 2020 Alexey Alexandrov <sks2311211@yandex.ru>

"""
Лабораторная работа № 6
"Игры с природой". Критерии принятия решений
Цель работы: ознакомление с основными понятиями "игр с природой"; получение навыков применения
различных критериев
(Бернулли, Вальда, максимума, Гурвица, Сэвиджа) для выбора стратегии в условиях полной
неопределённости.
Вариант 1.
"""

from best_strategy import *
STRATEGY_MATRIX = np.array([
    [1, 11, 12, 11],
    [7, 5, 7, 7],
    [16, 6, 13, 2],
    [9, 9, 16, 13],
    [17, 18, 15, 7]
], dtype=float)
if __name__ == '__main__':
    print(f"Игры с природой". Критерии принятия решений.\nМатрица
стратегий\n{OutMatrix(STRATEGY_MATRIX)})
    ChooseBestStrategy(STRATEGY_MATRIX)
```

Файл «best_strategy.py»

```
# Copyright 2020 Alexey Alexandrov <sks2311211@yandex.ru>

import numpy as np
from prettytable import PrettyTable
# Вероятность "везения" для критерия Гурвица.
ALPHA = 0.5

def OutMatrix(matrix: np.array):
    table = PrettyTable()
    table.field_names = ["Стратегии"] + [f"b{j}" for j in range(1, matrix.shape[1] + 1)]
    for i in range(matrix.shape[0]):
        table.add_row([f"a{i + 1}"] + list(matrix[i]))
    return table

def BernulliCriteria(matrix: np.array):
    """
    Критерий недостаточного основания Бернулли.
    :param matrix: матрица стратегий.
    :return полученная оптимальная стратегия.
    """
    table = OutMatrix(matrix)
    col_sum = np.sum(matrix, axis=1)
    table.add_column("Ψi", list(col_sum / matrix.shape[1]))
```

```

optimal_strategy = f"a{col_sum.argmax(axis=0) + 1}"
print(f"1) Критерий недостаточного основания Бернулли.\n"
      f"Если пользоваться критерием Бернулли, то следует руководствоваться стратегией"
      f"{optimal_strategy}. Соответствующее математическое ожидание выигрыша при этом"
      f"максимально и равно {col_sum.max(axis=0) / matrix.shape[1]}. \n{table}")
return optimal_strategy
def ValdCriteria(matrix: np.array):
    """
    Критерий пессимизма Вальда.
    :param matrix: матрица стратегий.
    :return полученная оптимальная стратегия.
    """

    table = OutMatrix(matrix)
    col_min = np.min(matrix, axis=1)
    table.add_column("αi", list(col_min))
    optimal_strategy = f"a{col_min.argmax(axis=0) + 1}"
    print(f"2) Критерий пессимизма Вальда.\n"
          f"Пессимистическая стратегия (Вальда) определяет выбор {optimal_strategy} "
          f"(нижняя цена игры равна {col_min.max(axis=0)}). \n{table}")
    return optimal_strategy
def OptimismCriteria(matrix: np.array):
    """
    Критерий авантюры (максимума, оптимизма).
    :param matrix: матрица стратегий.
    :return полученная оптимальная стратегия.
    """

    optimal_strategy = f"a{np.max(matrix, axis=1).argmax() + 1}"
    print(f"3) Критерий авантюры(максимума, оптимизма).\n "
          f"Оптимистическая стратегия соответствует выбору {optimal_strategy} "
          f"с максимальным выигрышем в матрице - {matrix.max()}.")
    return optimal_strategy
def GurrwitzCriteria(matrix: np.array):
    """
    Критерий Гурвица.
    :param matrix: матрица стратегий.
    :return полученная оптимальная стратегия.
    """

    table = OutMatrix(matrix)
    psi = ALPHA * np.min(matrix, axis=1) + (1 - ALPHA) * np.max(matrix, axis=1)
    table.add_column("Ψ", psi)
    optimal_strategy = f"a{psi.argmax() + 1}"
    print(f"4) Критерий Гурвица.\n"
          f"Критерий Гурвица определяется из условия равновероятной реализации пессимистической и"
          f"оптимистической гипотез (α = {ALPHA}). \n{table}\nНаилучшая стратегия {optimal_strategy}\n"
          f"Ожидаемый выигрыш: {psi.max()}")
    return optimal_strategy
def SeviceCriteria(matrix: np.array):
    """
    Критерий рисков Севиджа.
    :param matrix: матрица стратегий.
    :return полученная оптимальная стратегия.
    """

    risks = np.max(matrix, axis=0) - matrix
    table = OutMatrix(risks)
    max_col = np.max(risks, axis=1)
    table.add_column("αi", max_col)

```

```

optimal_strategy = f"a{ max_col.argmax() + 1 }"
print(f"5) Критерий Севиджа.\n"
      f"Составим таблицу рисков стратегий:\n{table}\n"
      f"Таким образом, оптимальная рискованная стратегия - {optimal_strategy}.")
return optimal_strategy
def ChooseBestStrategy(matrix: np.array):
    """
    Отбирает наилучшую стратегию согласно принципу большинства на основе всех рассмотренных критериев.
    :param matrix: матрица стратегий.
    """
    result_map = {}
    for k in [f"a{i + 1}" for i in range(matrix.shape[0])]:
        result_map[k] = 0
    result_map[BernulliCriteria(matrix)] += 1
    result_map[ValdCriteria(matrix)] += 1
    result_map[OptimismCriteria(matrix)] += 1
    result_map[GurvitzCriteria(matrix)] += 1
    result_map[SevigeCriteria(matrix)] += 1
    strategies = list(result_map.keys())
    ranks = list(result_map.values())
    table = PrettyTable()
    table.field_names = strategies
    table.add_row(ranks)
    print(f"\nВ итоге выберем стратегию, которая оказалась оптимальной в большем числе критериев:\n{table}\n"
          f"По принципу большинства рекомендуем стратегию {strategies[max(enumerate(ranks), key=lambda x: x[1])[0])}]")

```