**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра математического обеспечения и применения ЭВМ**

отчет

**по практической работе №1**

**по дисциплине «Теория принятия решений»**

Тема: Принятие решений в матричных играх

Вариант 2

|  |  |  |
| --- | --- | --- |
| Студент гр. 8383 |  | Бабенко Н.С. |
| Преподаватель |  | Попова Е.В. |

Санкт-Петербург

2022

**Цель работы**

Изучение различных инструментальных средств для решения задач поддержки принятия решения, а также овладение навыками принятия решения на основе матричных игр.

**Основные теоретические положения**

Рассмотрим простейшую математическую модель конечной конфликтной ситуации, в которой имеется два участника и выигрыш одного равен проигрышу другого. Такая модель называется антагонистической игрой двух лиц с нулевой суммой. Игра состоит из двух ходов: игрок А выбирает одну из возможных стратегий, а игрок Б выбирает одну из возможных стратегий . Каждый выбор производится при полном незнании выбора соперника. В результате выигрыш игроков составит соответственно  и . Цель игрока А – максимизировать величину , а игрока Б – минимизировать эту величину. Критерием принятия решения является функция, выражающая предпочтение лица, принимающего решение, и определяющая правило, по которому выбирается приемлемый или оптимальный вариант решения.

Матрица, составленная из величин .

|  |  |  |
| --- | --- | --- |
|  | . | (1) |

Матрица (1) называется платежной матрицей, или матрицей игры. Каждый элемент платежной матрицы , равен выигрышу А (проигрышу Б), если он выбрал стратегию , а игрок Б выбирал стратегию .

Задача каждого из игроков – найти наилучшую стратегию игры, при этом предполагается, что противники одинаково разумны и каждый из них делает все, чтобы получить наибольший доход.

Найдем наилучшую стратегию первого игрока. Если игрок А выбрал стратегию , то в худшем случае (например, если его ход известен Б) он получит выигрыш . Предвидя такую возможность, игрок А должен выбрать такую стратегию, чтобы максимизировать свой минимальный выигрыш.

|  |  |  |
| --- | --- | --- |
|  |  | (2) |

Представленная в (2) величина  – гарантированный выигрыш игрока А называется нижней ценой игры. Стратегия , обеспечивающая получение выигрыша , называется максиминной. Если первый игрок будет придерживаться своей максиминной стратегии, то у него есть гарантия, что он в любом случае выиграет не меньше . Аналогично определяется наилучшая стратегия второго игрока. Игрок Б при выборе стратегии , в худшем случае получит проигрыш . Он выбирает стратегию  при которой его проигрыш будет минимальным и составит

|  |  |  |
| --- | --- | --- |
|  |  | (3) |

Представленная в (3) величина  – гарантированный проигрыш игрока Б называется верхней ценой игры. Стратегия , обеспечивающая получение проигрыша , называется минимаксной. Если второй игрок будет придерживаться своей минимаксной стратегии, то у него есть гарантия, что он в любом случае проиграет не больше . Фактический выигрыш игрока А (проигрыш игрока Б) при разумных действиях партнеров ограничен верхней и нижней ценой игры. Для матричной игры справедливо неравенство . Если , т.е.

|  |  |  |
| --- | --- | --- |
|  |  | (4) |

то выигрыш игрока А (проигрыш игрока Б) определяется числом . Оно называется ценой игры.

В соответствии с (4), если , то такая игра называется игрой с седловой точкой. Элемент матрицы , соответствующий паре оптимальных стратегий , называется седловой точкой матрицы. Этот элемент является ценой игры.

Седловой точке соответствуют оптимальные стратегии игроков. Их совокупность – решение игры, которое обладает свойством: если один из игроков придерживается оптимальной стратегии, то второму отклонение от своей оптимальной стратегии не может быть выгодным. Если игра имеет седловую точку, то говорят, что она решается в чистых стратегиях.

Если платежная матрица не имеет седловой точки, т.е.  и то поиск решения игры приводит к применению сложной стратегии, состоящей в случайном применении двух и более стратегий с определенными частотами.

Сложная стратегия, состоящая в случайном применении всех стратегий с определенными частотами, называется смешанной.

**Постановка задачи**

Используя инструментальные средства компьютерной алгебры решить матричные задачи.

**Выполнение работы**

* С помощью инструментального средства определить границы выигрыша и наличие седловой точки для матрицы . Матрица представлена в (5).

|  |  |  |
| --- | --- | --- |
|  |  | (5) |

Результат выполнения программы, представленной в приложении А, показан на рис. 1.

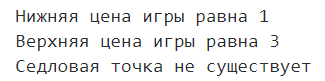
****

Рисунок 1 – Результат выполнения программы для матрицы

* Графически и аналитически решить матричную игру 2×2 для матрицы . Матрица представлена в (6).

|  |  |  |
| --- | --- | --- |
|  |  | (6) |

Решим данную задачу аналитически.

Надо проверить, есть ли седловая точка для исходной платежной матрицы. В случае если платежная матрица имеет седловую точку, необходимо выписать решение игры в чистых стратегиях.

Найдем нижнюю (7) и верхнюю (8) цену игры.

|  |  |  |
| --- | --- | --- |
|  |  | (7) |
|  |  | (8) |

Получаем, что , а значит нет седловой точки и решение в чистых стратегиях не существует.

Найдем цену игры . Известно, что .

В таком случае игрок A должен выбрать свои смешанные стратегии так, чтобы получить максимальный средний выигрыш. Аналогично, игрок Б должен выбрать свои смешанные стратегии так, чтобы минимизировать средний выигрыш игрока А.

Для этого запишем две системы (9) и (10) уравнений и решим их.

Для игрока А:

|  |  |  |
| --- | --- | --- |
|  |  | (9) |

Для игрока Б:

|  |  |  |
| --- | --- | --- |
|  |  | (10) |

Оптимальная стратегия игрока А:

Оптимальная стратегия игрока Б:

Цена игры:

Решим данную задачу графически. Результаты представлены на рис. 2 и 3.

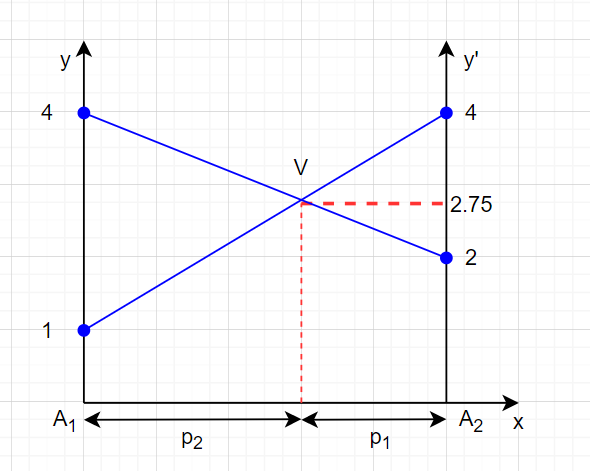


Рисунок 2 – Гарантированный выигрыш первого игрока в матрице платежей

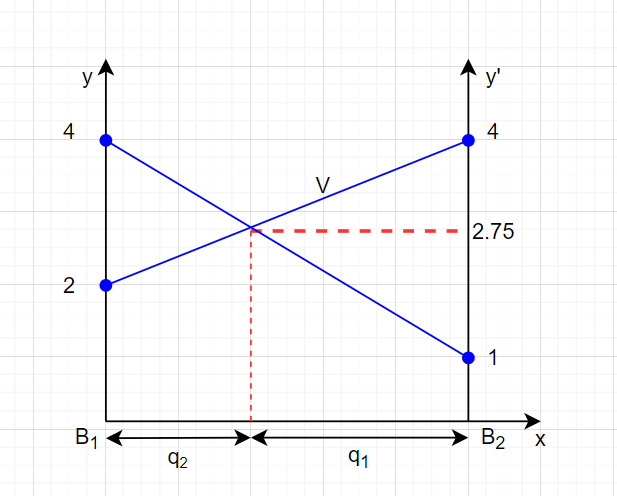


Рисунок 3 – Гарантированный проигрыш второго игрока в матрице платежей

Относительная погрешность равна

По рис. 2 и 3 можно определить, что стратегии игроков А и Б равны , цена игры – , Так как можем сделать вывод о том, что седловой точки не существует.

* Графически и аналитически решить матричную игру 2×N для матрицы . Матрица представлена в (11).

|  |  |  |
| --- | --- | --- |
|  |  | (11) |

Решим данную задачу графически. Результат представлен на рис. 5.

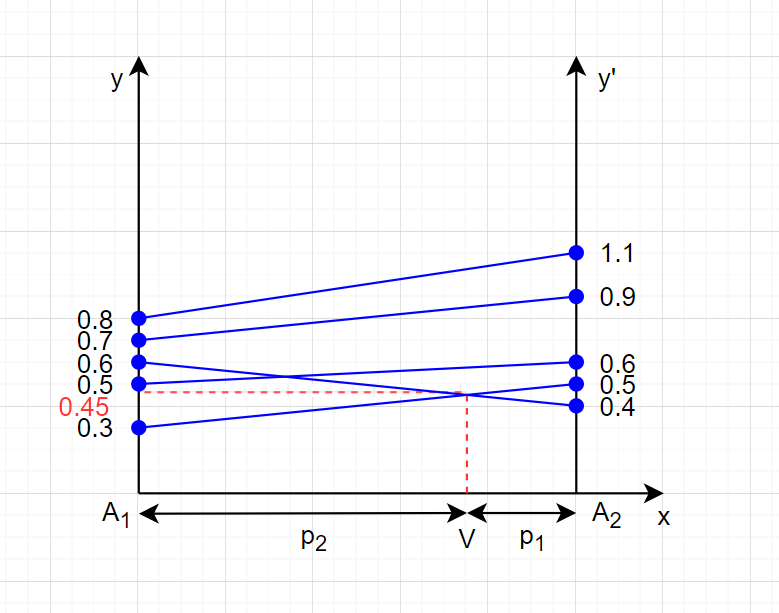


Рисунок 5 – Гарантированный выигрыш первого игрока при различном выборе смешанной стратегии

По рис. 5 можно определить цена игры , Так как можем сделать вывод о том, что седловая точка отсутствует. Первому игроку заведомо невыгоды стратегии 4 и 5.

Решим данную задачу аналитически.

Для аналитического решения выберем прямые, на пересечении которых находится седловая точка и построим из них матрицу 2 × 2:

Надо проверить, есть ли седловая точка для исходной платежной матрицы. В случае если платежная матрица имеет седловую точку, необходимо выписать решение игры в чистых стратегиях.

Найдем нижнюю (12) и верхнюю (13) цену игры.

|  |  |  |
| --- | --- | --- |
|  |  | (12) |
|  |  | (13) |

Получаем, что , а значит нет седловой точки и решение в чистых стратегиях не существует.

Найдем цену игры . Известно, что .

Запишем две системы (14) и (15) уравнений и решим их.

Для игрока А:

|  |  |  |
| --- | --- | --- |
|  |  | (14) |

Для игрока Б:

|  |  |  |
| --- | --- | --- |
|  |  | (15) |

Оптимальная стратегия игрока А:

Оптимальная стратегия игрока Б:

Цена игры:

* Графически и аналитически решить матричную игру *M×*2 для матрицы . Матрица представлена в (16).

|  |  |  |
| --- | --- | --- |
|  |  | (16) |

Решим данную задачу графически. Результат представлен на рис. 9.

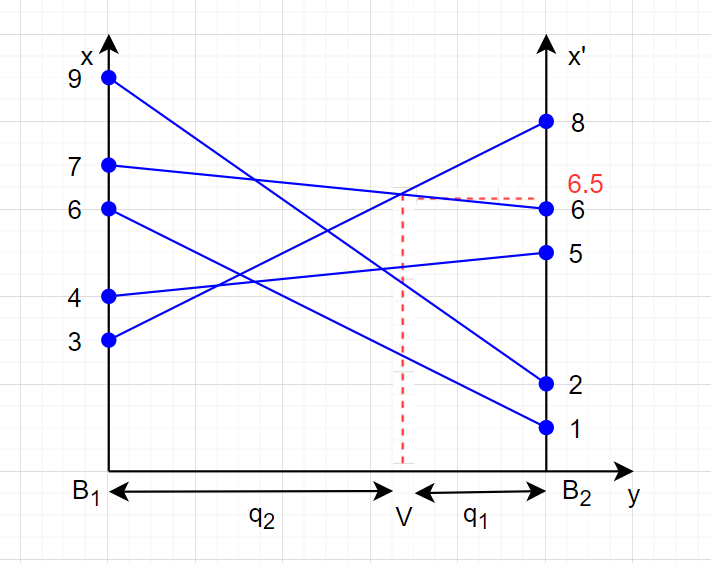


Рисунок 6 – Гарантированный проигрыш второго игрока при различном выборе смешанной стратегии

Исходя из рис. 6 можно определить, что смешанная стратегия игрока Б приблизительно равна цена игры – Так как можем сделать вывод о том, что седловой точки не существует. Второму игроку заведомо невыгодны стратегии 3 и 5.

Решим данную задачу аналитически.

Для аналитического решения выберем прямые, на пересечении которых находится седловая точка и построим из них матрицу 2 × 2:

Требуется проверить, есть ли седловая точка для исходной платежной матрицы. В случае если платежная матрица имеет седловую точку, необходимо выписать решение игры в чистых стратегиях.

Найдем нижнюю (17) и верхнюю (18) цену игры.

|  |  |  |
| --- | --- | --- |
|  |  | (17) |
|  |  | (18) |

Получаем, что , а значит нет седловой точки и решение в чистых стратегиях не существует.

Найдем цену игры . Известно, что .

Запишем две системы (19) и (21) уравнений.

Для игрока А:

|  |  |
| --- | --- |
|  | (19) |
|  | (20) |

Для игрока Б:

|  |  |
| --- | --- |
|  | (21) |
|  | (22) |

Оптимальная стратегия игрока А:

Оптимальная стратегия игрока Б:

Цена игры:

Относительная погрешность равна

* С помощью симплекс-метода решить матричную игру *M×N* для матрицы . Матрица представлена в (23).

|  |  |  |
| --- | --- | --- |
|  |  | (23) |

Требуется проверить, есть ли седловая точка для исходной платежной матрицы. В случае если платежная матрица имеет седловую точку, необходимо выписать решение игры в чистых стратегиях.

Найдем нижнюю (24) и верхнюю (25) цену игры.

|  |  |  |
| --- | --- | --- |
|  |  | (24) |
|  |  | (25) |

Получаем, что , а значит нет седловой точки и решение в чистых стратегиях не существует.

Найдем цену игры . Известно, что

Запишем две системы (26) и (27) уравнений.

Для игрока А:

|  |  |  |
| --- | --- | --- |
|  | | (26) |
|  |  | |

Чтобы найти решение данной задачи относительно первого игрока А надо:

найти минимум функции

при следующих ограничениях:

|  |  |
| --- | --- |
|  | (24) |

С помощью библиотеки SciPy для оптимизации и поиска корней в линейном программировании симплекс-методом вычислен вектор *X* (рис. 7).



Рисунок 7 – Решение вектора симплекс-методом матрицы

Получаем при Цена игры при этом что соотносится с первоначальной оценкой .

|  |  |
| --- | --- |
| Для игрока Б: | (27) |

Чтобы найти решение данной задачи относительно второго игрока Б надо найти максимум функции при следующих ограничениях:

С помощью библиотеки SciPy симплекс-методом вычислен вектор *Y*.

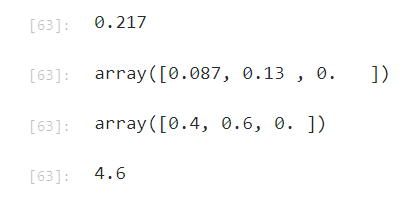


Рисунок 8 – Решение вектора симплекс-методом матрицы

Получаем при Цена игры при этом что соотносится с первоначальной оценкой .

Опираясь на полученные данные, можно найти цену игры (2) и оптимальные смешанные стратегии игроков А (26) и Б (27):

|  |  |  |
| --- | --- | --- |
|  |  | (25) |
|  |  | (26) |
|  |  | (27) |

**Выводы**

В ходе выполнения практической работы было реализовано инструментальное средство для нахождения нижней и верхней цен игры и для нахождения координат седловой точки на языке программирования Python, а также были получены навыки работы с библиотекой SciPy для линейного программирования.

При решении матричных игр были применены графический и аналитический методы нахождения оптимальных смешанных стратегий, а также был применён симплекс-метод. Решения матричных игр обоими способами дали одинаковый результат.

Приложение А

**ПРОГРАММА ПОИСКА СТАЦИОНАРНОЙ ТОЧКИ**

# %%

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from IPython.core.interactiveshell import InteractiveShell

InteractiveShell.ast\_node\_interactivity = "all"

# %%

def default(c):

amin = c.min(axis=1)

bmax = c.max(axis=0)

alpha = max(amin)

beta = min(bmax)

print(f'Нижняя цена игры равна {alpha}')

print(f'Верхняя цена игры равна {beta}')

print(f'Седловая точка существует') if alpha == beta else print(f'Седловая точка не существует')

if alpha == beta:

print(f'Координаты седловой точки равны ({np.argmax(amin)+1}, {np.argmin(bmax)+1})')

p1 = (c[1,1]-c[1,0])/(c[0,0]+c[1,1]-(c[0,1]+c[1,0]))

p2 = 1-p1

q1 = (c[1,1]-c[0,1])/(c[0,0]+c[1,1]-(c[0,1]+c[1,0]))

q2 = 1-q1

v = c[0,0]\*p1+c[1,0]\*p2

print(p1,p2,q1,q2,v)

# %%

c1 = np.array([[2,3,-1,4], [3,2,4,1], [-4,3,-1,-2], [-5,5,-3,-4]])

c2 = np.array([[4,1],[2,4]])

c3 = np.array([[0.5,0.3,0.6,0.7,0.8], [0.6,0.5,0.4,0.9,1.1]])

c31 = np.array([[0.3,0.6],[0.5,0.4]])

c4 = np.array([[3,8],[7,6],[4,5],[9,2],[6,1]])

c41 = np.array([[3,8],[7,6]])

c5 = np.array([[4,5,6], [7,3,2], [2,1,8]])

# %%

default(c5)

# %%

from scipy.optimize import linprog

obj = [1,1,1]

lhs\_ineq = [[-4,-7,-2],[-5,-3,-1],[-6,-2,-8]]

rhs\_ineq = [-1,-1,-1]

bnd = [(0, float("inf")),(0, float("inf")),(0, float("inf"))]

opt = linprog(c=obj, A\_ub=lhs\_ineq, b\_ub=rhs\_ineq, bounds=bnd,method="simplex")

opt.fun.round(3)

opt.x.round(3)

(1/opt.fun) \* opt.x

(1/opt.fun).round(3)

# %%

from scipy.optimize import linprog

obj = [-1,-1,-1]

lhs\_ineq = [[4,5,6],[7,3,2],[2,1,8]]

rhs\_ineq = [1,1,1]

bnd = [(0, float("inf")),(0, float("inf")),(0, float("inf"))]

opt = linprog(c=obj, A\_ub=lhs\_ineq, b\_ub=rhs\_ineq, bounds=bnd,method="simplex")

-opt.fun.round(3)

opt.x.round(3)

(-1/opt.fun) \* opt.x

(-1/opt.fun).round(3)