Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

УНИВЕРСИТЕТ ИТМО

Группа: M33101

Студент: Косовец Роман / Дмитрий Надеждин

Преподаватель: Дорофеева Юлия Александровна

Рабочий протокол и отчет по лабораторной работе № 1

«Линейное программирование»

1. Цель работы:

Реализовать симплекс-метод для решения задачи

2. Задачи, решаемые при выполнении работы:

1. Реализовать возможность ввода файла в формате JSON.
2. Добавить переход от общей постановки задачи к канонической путем добавления балансирующих переменных.
3. Реализовать симплекс-метод
4. Предусмотреть, что задача может не иметь решений

Ссылка на код на GitHub:
https://github.com/RomanKosovets/PriMat/blob/main/Lab_5/SimplexMethod.py

3. Условие задачи:

Постановка задачи

1. Реализуйте возможность ввода данных из файла в формате JSON. Рекомендуемая структура JSON указана ниже.

2. При необходимости добавьте балансирующие переменные для перехода от общей постановки к канонической форме задачи линейного программирования.

3. Реализуйте симплекс-метод для решения задачи.

4. Предусмотрите, что задача как может не иметь решений вообще, так и иметь бесконечное количество решений.

Структура JSON

Задача линейного программирования (общая форма):

$$f(x) = 1 \cdot x_1 + 2 \cdot x_2 + 3 \cdot x_3 \to \max \tag{1}$$

$$\begin{cases} x_1 \leqslant 1 \\ x_1 + x_2 \geqslant 2 \\ x_1 + x_2 + x_3 = 3 \end{cases} \tag{2}$$

$$x_1 \geqslant 0, x_2 \geqslant 0, x_3 \geqslant 0 \tag{3}$$

В формате JSON:

```
{"f": [1, 2, 3],
 "goal": "max",
 "constraints": [{"coefs": [1, 0, 0],
                  "type": "eq",
                  "b": 1},
                 {"coefs": [1, 1, 0],
                  "type": "gte",
                  "b": 2},
                 {"coefs": [1, 1, 1],
                  "type": "lte",
                  "b": 3}]}
```

4. Решение задачи:

Для начала был написан необходимый парсер для того, чтобы облегчить обработку и анализ данных для задачи линейного программирования из JSON.

Таким образом, мы извлекли данные из объекта JSON, получив доступ к ключам и их значениям. То есть мы получили коэффициенты целевой функции и ограничения.

Дальнейшим шагом выполнения лабораторной работы было преобразование общей задачи в каноническую с помощью добавления балансирующих переменных. Для этого были использованы циклы и условные операторы.

Симплекс-метод

Симплекс метод — это подход к решению задачи линейного программирования с использованием дополнительных переменных, таблиц и опорных переменных для нахождения оптимального решения задачи оптимизации. Чтобы решить задачу линейного программирования используя симплекс метод, выполним следующие шаги:

1. Приведём проблему к канонической форме
2. По надобности добавим доп. Переменные
3. Создадим таблицу для решения
4. Определим опорные элементы
5. Создадим новую таблицу с помощью опорного элемента
6. Проверим решение на оптимальность

1. Переход к каноничной форме:

Можно сказать, что каноническая форма – это базовый формат для всех задач линейного программирования. У нас есть три требования:

1. ЦФ(x) -> max
2. Все ограничения должны быть представлены в виде неравенств “меньше либо равно” или “больше либо равно”, чтобы затем преобразовать неравенства в равенства.
3. все переменные должны быть неотрицательными.

Эти требования всегда могут быть выполнены путем элементарных преобразований нашей задачи. Стандартная форма необходима, потому что она создает идеальную начальную точку для решения симплекс методом.

Также надо помнить, чтобы преобразовать проблему минимизации к проблеме максимизации, умножим обе части уравнения на -1. Кроме этого, с помощью умножения на -1 можно изменить знак с ≤ на ≥, благодаря этому мы избавимся от отрицательных дополнительных переменных, которые мы бы добавили в случае ≥.

2. Добавление дополнительных переменных:

Дополнительные переменные — это переменные, вводимые в ограничения задачи для преобразования их из неравенств в равенства. Если модель находится в стандартной форме, коэффициенты перед доп. переменными всегда равны +1. Доп. переменные необходимы в ограничениях для того, чтобы преобразовать их в равенства, которые можно решить. Также нужно помнить, что в целевой функции эти переменные входят с коэффициентом “0”

После введения доп. переменных, можно создать таблицу для проверки оптимальности, как описано в третьем шаге.

3. Создание таблицы:

Таблица используется для выполнения операций над строками, а также для проверки решения на оптимальность. Таблица состоит из коэффициентов, соответствующих переменным ограничения, и коэффициентов целевой функции. В таблице ниже жирным шрифтом выделена верхняя строка, которая указывает, что представляет собой каждая колонка. Следующие две строки представляют коэффициенты переменных ограничения из задачи, и последняя строка представляет коэффициенты переменных целевой функции.

x1	**x2**	**x3**	**s1**	**s2**	**z**	**b**
1	3	2	1	0	0	10
1	5	1	0	1	0	8
-8	-10	-7	0	0	1	0

4. Проверка на оптимальность:

Оптимальное решение задачи максимизации целевой функции — это такие значения переменных целевой функции, которые дают нам максимальное значение нашей функции. Оптимальное решение будет существовать на угловых точках графика всей модели.

Для проверки оптимальности с использованием таблицы, все значения в последней строке должны быть больше или равны нулю. Если значение меньше нуля, это означает, что переменная еще не достигла своего оптимального значения. Как видно в предыдущей таблице, три отрицательных значения существуют в последней строке, что указывает на то, что это решение не является оптимальным.

Если таблица не является оптимальной, следующим шагом является определение опорной переменной для создания новой таблицы

5. Определение опорной переменной:

Опорная переменная используется в операциях с строками для определения, какая переменная станет равна единице, и является ключевым фактором в преобразовании единичного значения. Опорную переменную можно определить, рассматривая нижнюю строку таблицы и индикатор. Предполагая, что решение не является оптимальным, выберите наименьшее отрицательное значение в нижней строке. Одно из значений в столбце этого значения будет опорной переменной. Чтобы найти индикатор, разделите значения b линейных ограничений на соответствующие значения из столбца, содержащего возможную опорную переменную. Пересечение строки с наименьшим неотрицательным индикатором и столбца с наименьшим отрицательным значением в нижней строке станет опорной переменной.

6. Создание новой таблицы:

С помощью новой таблицы можно определить новое возможное оптимальное решение. Теперь, когда опорная переменная найдена, можно выполнить операции со строками для оптимизации опорной переменной, сохраняя при этом эквивалентность остальной части таблицы.

1. Для оптимизации опорной переменной необходимо преобразовать ее в единицу. Для этого умножим строку, содержащую опорную переменную, на обратное значение этой переменной.
2. После того как опорная превратилась в 1, другие значения в столбце, содержащем единичное значение, станут равными нулю. Это происходит потому, что переменная x2 во втором ограничении оптимизируется, что требует, чтобы переменная x2 в других уравнениях была равна нулю.
3. Чтобы сохранить эквивалентность таблицы, другие переменные, не содержащиеся в столбце опорной переменной или строке опорной переменной, должны быть вычислены с использованием новых значений опорной переменной. Для каждого нового значения необходимо умножить отрицательное значение в старом столбце опорной переменной на значение в новой строке опорной переменной, которое соответствует вычисляемому значению. Затем добавьте это к старому значению из старой таблицы, чтобы получить новое значение для новой таблицы.

Третий шаг можно свести в простую формулу:

Новое значение переменной = (Отрицательное значение в старом столбце опорной переменной) x (значение в новой строке опорной переменной) + (Значение в старой таблице)

После определения новой таблицы, можем проверить наличие оптимального решения.

7. Проверка оптимальности:

После каждой новой таблицы необходимо проверить оптимальность, чтобы узнать, требуется ли идентификация новой опорной переменной. Решение считается оптимальным, если все значения в последней строке больше или равны нулю. Если все значения больше или равны нулю, решение считается оптимальным.

Если же существуют отрицательные значения, решение все еще не является оптимальным, и необходимо найти новую опорную точку, как показано на шаге 8.

8. Определение новой опорной переменной:

Если решение было определено как неоптимальное, необходимо найти новую опорную переменную. Опорная переменная была введена на шаге 5 и используется в операциях по строкам для определения того, какая переменная станет единичным значением и является ключевым фактором при преобразовании в единичное значение. Опорную переменную можно идентифицировать как пересечение строки с наименьшим неотрицательным индикатором и наименьшим отрицательным значением в последней строке. После этого необходимо снова проверить оптимальность

9. Определение оптимальных значений:

После того как таблица симплекс-метода оптимальная, можно определить оптимальные значения переменных. Это можно сделать, выделив базовые и не базовые переменные. Базовая переменная может быть классифицирована как переменная, у которой в столбце есть только одна единица, а остальные значения равны нулю. Если переменная не соответствует этим критериям, она считается не базовой. Если переменная является не базовой, это означает, что оптимальное решение для этой переменной равно нулю. Если переменная является базовой, строка, содержащая единицу, будет соответствовать значению b. Значение b будет представлять оптимальное решение для данной переменной.

5. Вывод:

Симплекс-метод — это подход для определения оптимального решения задачи линейного программирования. Этот метод позволяет получить оптимальное решение, удовлетворяющее заданным ограничениям и обеспечивающее максимальное значение целевой функции. Для использования симплекс-метода задача должна быть представлена в стандартной форме, где могут быть введены дополнительные переменные. С использованием таблицы и опорных переменных можно достичь оптимального решения. Из примера, рассмотренного в этой лабораторной работе, можно заключить, что оптимальное целевое значение равно max = 7 и может быть достигнуто, когда $x1 = 0$, $x2 = 2$ и $x3 = 1$