Лабораторная работа № 4

Линейная алгебра

Сунгурова Мариян

07 декабря 2024

Российский университет дружбы народов, Москва, Россия

Информация

Докладчик

- Сунгурова Мариян Мухсиновна
- НКНбд-01-21
- Российский университет дружбы народов

Вводная часть

Цель работы

Основной целью данной работы является изучение возможностей специализированных пакетов Julia для выполнения и оценки эффективности операций над объектами линейной алгебры.

Задание

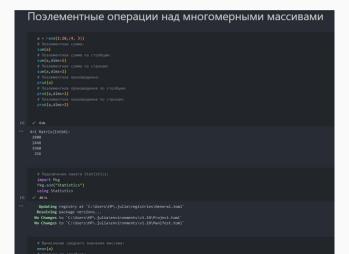
- 1. Используя JupyterLab, повторите примеры.
- 2. Выполните задания для самостоятельной работы.

Теоретическое введение

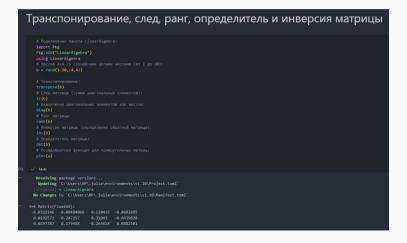
Julia – высокоуровневый свободный язык программирования с динамической типизацией, созданный для математических вычислений. Эффективен также и для написания программ общего назначения. Синтаксис языка схож с синтаксисом других математических языков, однако имеет некоторые существенные отличия.

Для выполнения заданий была использована официальная документация Julia.

Выполним примеры из раздела про поэлементные операции над многомерными массивами (рис. (fig:001?)-(fig:002?)).



Выполним примеры из раздела про транспонирование, след, ранг, определительи инверсия матрицы (рис. (fig:003?)).



Выполним примеры из раздела про вычисление нормы векторов и матриц, повороты и вращения (рис. (fig:003?) - (fig:004?)).

```
Вычисление нормы векторов и матриц, повороты, вращения
```

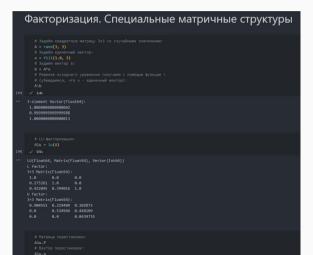
Рис. 3: Нормы векторов и матриц, повороты и вращения

Выполним примеры из раздела про матричное умножение,единичная матрица,скалярное произведение (рис. (fig:005?)).

```
Матричное умножение, единичная матрица, скалярное произведение
203 193 138 141
111 88 100 70
```

Рис. 5: Матричное умножение, единичная матрица, скалярное произведение

Выполним примеры из раздела про факторизацию и специальные матричные структуры (рис. (fig:006?)-(fig:008?)).



```
O factor: 3×3 LinearAlgebra.ORCompactWYO(Float64, Matrix(Float64), Matrix(Float64))
-1.00841 -0.452695 -0.308471
         -0.524902 -0.458847
```

```
4.61853e-14 -8.88178e-14
1888×1888 Symmetric(Float64, Matrix(Float64)):
```

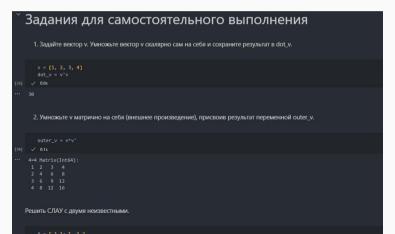
Далее для оценки эффективности выполнения операций над матрицами большой размерности и специальной структуры воспользуемся пакетом BenchmarkTools

```
Pkg.add("BenchmarkTools")
  Resolving package versions...
  Installed RenchmarkTools - v1.5.0
   Undating 'C:\Users\HP\_iulia\environments\v1.10\Project.toml'
   Updating `C:\Users\HP\.julia\environments\v1.10\Manifest.toml
 [6e4b80f9] † BenchmarkTools v1.4.0 = v1.5.0
Precompiling project...
 193.655 ms (11 allocations: 7.99 MiB)
```

Выполним примеры из раздела про общую линейную алгебру (рис. (fig:010?)).

```
Общая линейная алгебра
     Arational = Matrix{Rational{BigInt}}(rand(1:10, 3, 3))/10
     x = fill(1, 3)
     b = Arational*x
     Arational\b
     lu(Arational)
   ./ 675
  LU(Rational(BigInt), Matrix(Rational(BigInt)), Vector(Int64))
  I factor:
  3×3 Matrix{Rational{BigInt}}:
        7//34 1
  U factor:
  3×3 Matrix{Rational{BigInt}}:
          34//35 11//70
```

Зададим вектор v. Умножим вектор v скалярно сам на себя и сохраним результат вdot_v Затем умножим v матрично на себя(внешнее произведение), присвоив результат переменной outer_v .(puc. (fig:011?)).



Решим СЛАУ с двумя неизвестными (рис. (fig:012?)-(fig:013?)).

Рис. 12: Системы линейных уравнений

Решим СЛАУ с тремя неизвестными (рис. (fig:014?)).

```
Решить СЛАУ с тремя неизвестными.
        println("Нет решений")
```

Приведем матрицы к диагональному виду (рис. (fig:015?)).

```
Операции с матрицами. Приведите приведенные ниже матрицы к диагональном
Вычислите
 29524 29525
```

Вычислим (рис. (fig:016?)).

```
Операции с матрицами. Приведите приведенные ниже матрицы к диагональном
Вычислите
 -29524 29525
```

Найдем собственные значения матрицы А. Создадим диагональную матрицу из собственных значений матрицы А. Создадим нижнедиагональную матрицу из матрицы А. Оценим эффективность выполняемых операций (рис. (fig:017?)).

```
łайдите собственные значения матрицы A
  A - [140 97 74 168 131; 97 106 89 131 36; 74 89 152 144 71; 168 131 144 54 142; 131 36 71 142 36]
 97 106 89 131 36
 74 89 152 144 71
168 131 144 54 142
5×5 LowerTriangular(Int64, Matrix(Int64))
 74 89 152
168 131 144 54
 131 36 71 142 36
```

Линейная модель может быть записана как СЛАУ

$$x - Ax = y,$$

где элементы матрицы A и столбца y – неотрицательные числа. По своему смыслу в экономике элементы матрицы A и столбцов x, y не могут быть отрицательными числами.

Матрица A называется продуктивной, если решение x системы при любой неотрицательной правой части y имеет только неотрицательные элементы x_i . Используя это определение, проверим, являются ли матрицып родуктивными (рис. (fig:018?)-(fig:019?)).

```
Линейные модели экономики
```

Рис. 18: Линейные модели экономики

Выводы

Выводы

В процессе выполнения данной лабораторной работы были изучены возможности специализированных пакетов Julia для выполнения и оценки эффективности операций над объектами линейной алгебры.