# Лабораторный практикум 9. Линейные пространства и операторы. Процесс ортогонализации Грамма-Шмидта.

Загрузите необходимые библиотеки:

>> import matplotlib.pyplot as plt

>> import numpy as np

>> from sympy import \*

## Базис в *n*-мерном пространстве. Матрица перехода

**Выяснить, является ли система векторов линейно-зависимой или нет, а также найти базис этой системы можно с помощью понятия ранга системы.**

**В Python ранг матрицы *А* вычисляется в библиотеке Numpy с помощью функции ***numpy.linalg.matrix\_rank(A*)**, а в библиотеке Sympy: ***A.rank*()**.** Метод ***columnspace*()** позволяет найти базис системы векторов, расположив их в виде столбцов матрицы:

****Пример 1.** Найти ранг и какой-нибудь базис матрицы *А*, состоящей из векторов:**

,,,.

A = Matrix([[2,4,5,6,0,4],[8,-2,0,2,4,-2],[6,-6,-5,-4,4,-6],

[-4,0,2,-2,2,0]])

A.rank()

**A.T.columnspace()**

**Ранг равен 3, базис составляют векторы** , и .

****Упражнение 9.1.** Найти ранг и какой-нибудь базис системы геометрических векторов** , , , . Разложить вектор по этому базису (используйте обратную матрицу). Сделать проверку.

Формула преобразования координат вектора при преобразовании базиса:

,

где — ***матрица перехода*** от базиса к базису .

****Упражнение 9.2.** В пространстве** *R*4 заданы векторы , , , . Доказать, что - базис в *R*4. Написать матрицу перехода, где *В* – канонический базис в *R*4. Найти в базисе координаты вектора . Сделать проверку.

Индексы базисных векторов можно также найти методом **.rref()**. Данный метод применяется к матрице и возвращает два кортежа. Первым элементом возвращаемого кортежа является преобразованная матрица, в левой части которой стоит единичная матрица. Вторым элементом является кортеж, содержащий индексы базисных столбцов входной матрицы.

****Упражнение 9.3.**** Выполнить задание из примера 1 с помощью метода rref.

**Пример 2.** Создать функцию, вычисляющую базисный минор произвольной матрицы, которая принимает на вход матрицу и возвращает подматрицу, образованную базисными строками и столбцами (одну из имеющихся для данной матрицы), а также ее определитель (он является базисным минором). Индексы базисного минора находятся с использованием ранее упомянутого метода rref. Провести вычисления для матрицы из примера 1.

def BasisMinor(A):

#A - матрица произвольного размера.

s\_col = A.rref()[1] #Индексы базисных столбцов

s\_row = A.T.rref()[1] #Индексы базисных строк

#Удаление строк, не входящих в систему базисных

m,n = A.shape

l = 0

for i in range(0,m):

if i in s\_row:

pass

else:

A.row\_del(i-l)

l=l+1

#Удаление столбцов, не входящих в систему базисных

l = 0

for j in range(0,n):

if j in s\_col:

pass

else:

A.col\_del(j-l)

l=l+1

return A, det(A)

В результате работы функции получаем: базисный минор равен -112, образован 1-й, 2-й и 4-й строками и 1-м, 2-м и 3-м столбцами матрицы:

(Matrix([

[ 2, 4, 5],

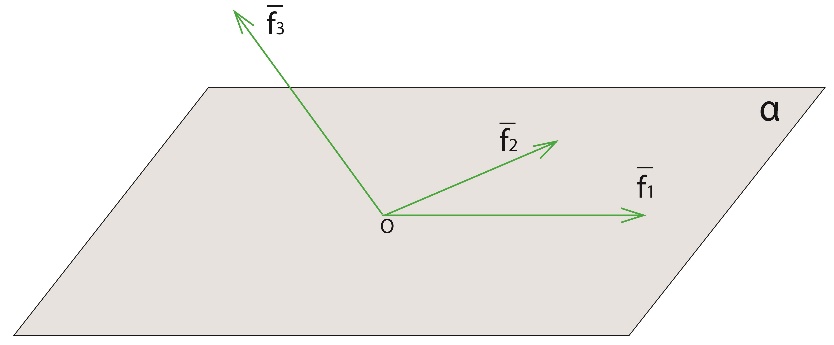
[ 8, -2, 0],

[-4, 0, 2]]),

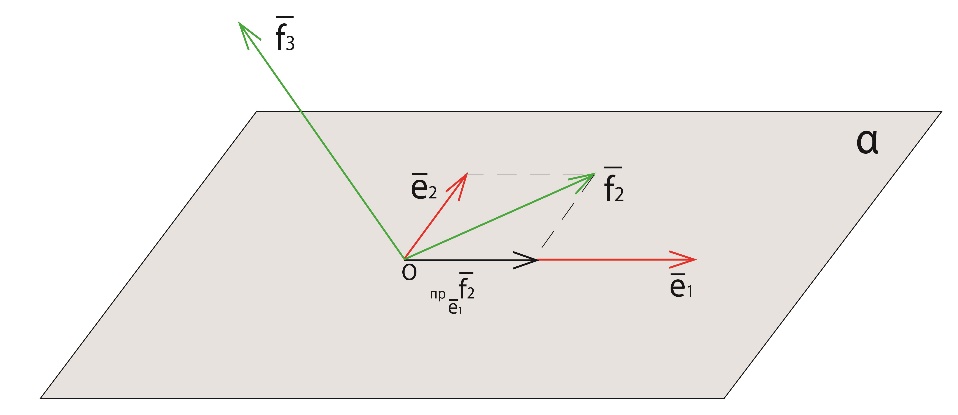
-112)

## Процесс ортогонализации Грамма-Шмидта

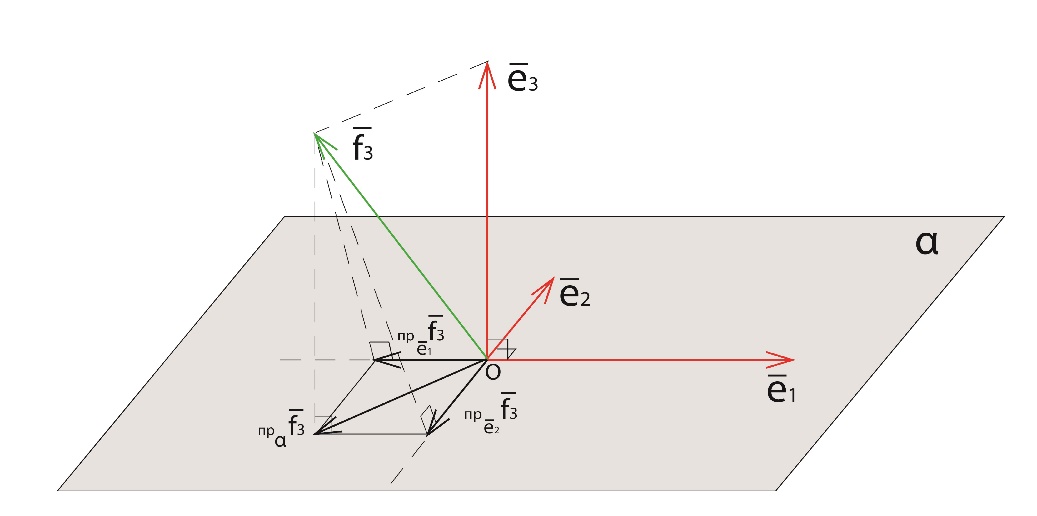
Ортогонализация ― алгоритм построения для данной линейно независимой системы векторов евклидова пространства *V* ортогональной системы ненулевых векторов, порождающих то же самое подпространство в *V*.



Первый шаг: ;



Второй шаг:



Третий шаг:

Если в *n*-мерном евклидовом пространстве задан произвольный базис , то векторы

,

где , образуют ортогональный базис в этом пространстве.

****Упражнение 9.4.** Применить процесс ортогонализации Грамма-Шмидта к указанным системам векторов. Проверить ортогональность полученной системы. Изобразить старую и новую системы векторов. Разложить произвольный вектор в старом и новом базисах.**

1) ;

2) .

## Линейные операторы и действия с ними

****Пример 3.** С примером линейного оператора (преобразования, отображения) линейного пространства мы уже встречались в практикуме 6, когда производили параллельный перенос и поворот систем координат. Покажем, что этот оператор является линейным.**

>> alfa,k,x,y=symbols('alfa k x y')

>> A=Matrix([[cos(alfa), -sin(alfa)],[sin(alfa), cos(alfa)]])

>> A\*(k\*x)

>> k\*(A\*x)

>> A\*(x+y)

>> simplify((A\*x)+(A\*y))

Определение линейного оператора выполняется.

Пусть *А* и – матрицы оператора *А* в базисах и . *Формула преобразования матрицы оператора при преобразовании базиса* имеет вид:

.

****Упражнение 9.5.** Матрица линейного оператора в некотором базисе**  равна . Найти матрицу этого оператора в базисе .

****Упражнение 9.6.** Матрица линейного оператора в некотором базисе**  равна . Найти матрицу этого оператора в базисе .

****Упражнение 9.7.** Линейный оператор в базисе**  задан матрицей *А*. Найти его матрицу в базисе (координаты векторов даны в некотором базисе ):

,

## Дополнительное задание

Применить процесс ортогонализации **Грамма-**Шмидта к системам векторов:

1. *F* = [1, 1, 1, 1; 3, 3, –1, –1; –2, 0, 6, 8];
2. *F* = [1, 2, 2, –1; 1, 1, –5, 3; 3, 2, 8, –7];

Проверить результат, используя процедуру ортогонализации Python – ***GramSchmidt*(*F*)**. Введите систему векторов: F = [Matrix([]), Matrix([]), Matrix([])].