

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МО ЭВМ**

**ОТЧЕТ**  
**по практическому заданию №2**  
**по дисциплине «Машинное обучение»**

Студент гр. 6304

Доброхвалов М. О.

Преподаватель

Жангиров Т. Р.

Санкт-Петербург

2020

## Задание 1

Запись исходных данных

```
x = np.array([[4, 2.9], [2.5, 1], [3.5, 4], [2, 2.1]])
```

Вычисление ядерной матрицы по формуле  $K(x_i, x_j) = \|x_i - x_j\|^2$

```
Kernel_matrix = np.array([[np.sum(np.power(X[i] - X[j],2)) for j in  
range(X.shape[0])] for i in range(X.shape[0])])
```

Результирующая матрица

0,00	5,86	1,46	4,64
5,86	0,00	10,00	1,46
1,46	10,00	0,00	5,86
4,64	1,46	5,86	0,00

## Задание 2

```
data = np.array([[8,0,10,10,2], [-20, -1, -19, -20, 0]]).T
```

1. Среднее значение - [ 6. -12.]

Матрица ковариации

22.	-47.5
-47.5	110.5

2. Собственные числа: [1.332 131.168]
3. “Внутренний” размер набора данных
4. Первая главная компонента

```
max_eigenval_idx = np.argmax(eigenvals)  
projection_mat = -eigenvecs[:,max_eigenval_idx]  
first_pc = data_centered@projection_mat
```

First principle component: [-8.134 12.48 -8.015 -8.932 12.599]

5. Пусть  $\mu$  и  $\Sigma$  характеризуют нормальное распределение. Построен график 2-мерной функции нормальной плотности (рис. 1)

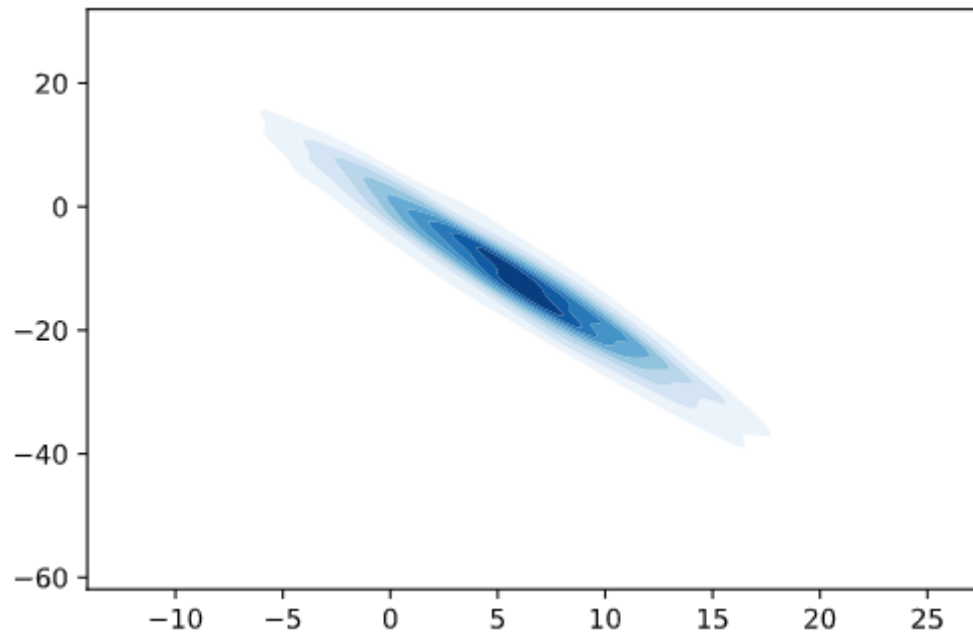


Рисунок 1 — 2-мерная функция нормальной плотности.

### Задание 3

```
centeroid = np.identity(X.shape[0]) - np.ones((X.shape[0],  
X.shape[0]))/X.shape[0]  
Kernel_matrix = centeroid@Kernel_matrix@centeroid  
precomputed_transformed_data = KernelPCA(1,  
'precomputed').fit_transform(Kernel_matrix)  
precomputed_data.reshape(-1).round(3)
```

Результат: [0., 0., 0., 0.]