

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

ОТЧЕТ
по практической работе №2
по дисциплине «Машинное обучение»

Студент гр. 6304

Антонов С.А.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2020

Задание 1

Исходные данные:

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

Вычисление ядерной матрицы по формуле: $K(X_i, X_j) = ||X_i - X_j||^2$

```
def kernel_function(x_i, x_j):  
    result = np.linalg.norm(x_i - x_j)**2  
    return result
```

```
kernel_matrix = np.array([[kernel_function(x1, x2) for x2 in X] for x1 in X])
```

Полученная матрица:

| | | | |
|------|------|------|------|
| 0.0 | 5.86 | 1.46 | 4.64 |
| 5.86 | 0.0 | 10.0 | 1.46 |
| 1.46 | 10.0 | 0.0 | 5.86 |
| 4.64 | 1.46 | 5.86 | 0.0 |

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: (5, 2)

4. Первая главная компонента:

```
new_data_centered = data - data.mean(axis=0)  
val_index = np.argmax(eigenvals)  
projection_mat = -vecs[:,val_index]  
first_comp = np.dot(new_data_centered, projection_mat)
```

```
[-8.13363886 12.4804344 -8.01463621 -8.93159638 12.59943705]
```

5. μ и Σ характеризуют нормальное распределение, тогда построим график 2-мерной функции нормальной плотности:

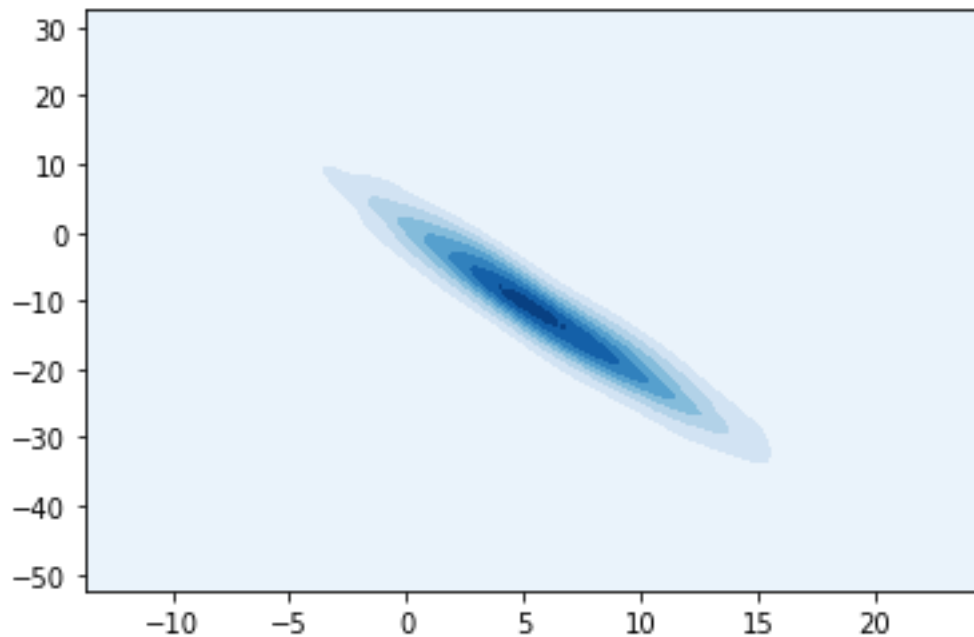


Рисунок 1 2-мерная функция нормальной плотности

Задание 3

```
transformation = np.array(kernel_matrix)/100 + np.ones((len(X), X.shape[0]))/2
kernel_matrix = transformation@kernel_matrix@transformation
precomputed_data = KernelPCA(1, 'precomputed').fit_transform(kernel_matrix)
precomputed_data.reshape(-1).round(3)
```

Итоговый результат:

```
array([-0.057, 0.057, 0.057, -0.057])
```