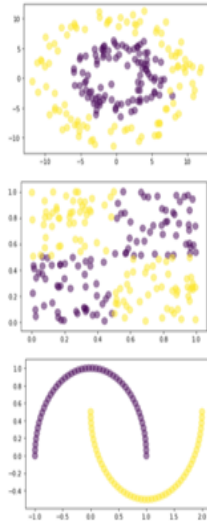


Домашнее задание №5

(срок выполнения: 1 неделя)

KNN, Линейная регрессия

Задание 1. Сгенерировать наборы данных (см. картинки):



Для каждого набора данных:

- Обучить и протестировать KNN
- Построить графики точности (train/test) при разных k (1...8)

Задание 2. Метрики

25 объектов классифицированы на 3 класса

C="Cat"

F="Fish"

H="Hen"

истинные значения

y_true = [C,C,C,C,C,C, F,F,F,F,F,F,F,F,F, H,H,H,H,H,H,H,H,H]

результат системы

y_pred = [C,C,C,C,H,F, C,C,C,C,C,C,H,H,F,F, C,C,C,H,H,H,H,H,H]

Написать код на python, который воспроизводит вывод (не используя сторонние библиотеки, кроме numpy):

```
[[4 1 1]
 [6 2 2]
 [3 0 6]]
```

	precision	recall	f1-score	support
Cat	0.308	0.667	0.421	6
Fish	0.667	0.200	0.308	10
Hen	0.667	0.667	0.667	9
accuracy			0.488	25
macro avg	0.547	0.511	0.465	25
weighted avg	0.581	0.488	0.464	25

Задание 3. Линейная регрессия

1. Самостоятельно выберите набор данных, над которым вы хотите работать.
2. Загрузите данные и напишите код, чтобы разбить набор на обучающий и тестовый наборы данных. Посмотрите, какие переменные вы можете использовать для прогнозирования значений другой переменной.
3. Измерьте точность вашей модели на тренировочном и тестовом наборе данных. Нарисуйте графики зависимостей одной переменной от другой переменной и постройте простую одномерную линейную регрессию. Какая из них дает лучший R^2 ?
4. Определите, сколько переменных вам нужно для получения наилучшего значения R^2 в тестовом наборе (добавляйте переменные «жадно» так, чтобы на каждом шаге после добавления новой переменной ошибка на тренировочном наборе уменьшалась максимально).