

## Базовая домашняя работа

Для успешной сдачи домашней работы нужно решить и защитить минимум 50% задач (на оценку "удовлетворительно") или минимум 80% (на оценку "хорошо" или "отлично"). Любые попытки сдать чужие решения отрицательно скажутся на вашей итоговой оценке.

1. Загрузите Wine Dataset (`sklearn.datasets.load_wine()`), выполните анализ главных компонент (PCA) на данном датасете, спроецируйте его на одномерное пространство. Постройте кривую ROC для задачи классификации (выберите любой из трех классов в качестве положительного). Код для построения кривой ROC нужно написать самостоятельно.
2. Используйте функцию `sklearn.datasets.load_breast_cancer`, чтобы загрузить датасет классификации рака груди. **Реализуйте** метод LDA для бинарной классификации. Спроецируйте данные на одномерное пространство и построьте кривую ROC (код для которой нужно написать самостоятельно).
3. Сгенерируйте три различных непериодических сигнала (например, логарифмический или полиномиальный) достаточной длины (100-200 значений), создайте 5 различных линейных комбинаций этих сигналов. Выполните анализ независимых компонент (используя стандартную функцию), чтобы восстановить исходные сигналы.
4. Загрузите данные из файла `problem3_data.npy`. Отобразите данные, чтобы иметь представление об их распределении. Используйте RANSAC для построения оптимальной модели. Отобразите получившуюся лучшую модель.
5. Используйте прямой метод Эйлера для решения следующего обыкновенного дифференциального уравнения

$$\frac{dy}{dx} = 3x^2y, \quad y(0) = 1$$

Найдите  $y(2)$ , используя разные значения шага. Отобразите график точности решения (относительно аналитического решения) в зависимости от размера шага (рассмотрите шаги в диапазоне  $[10^{-8}; 10^{-1}]$ ).

6. Загрузите данные из файла `problem5_data.npy`. Отобразите данные, чтобы получить представление об их распределении. Используйте линейную задачу наименьших квадратов, чтобы приблизить эти данные многочленом седьмой степени. Отобразите получившуюся кривую. Найдите среднеквадратичную ошибку.

7. Загрузите датасет с цифрами (`sklearn.datasets.load_digits()`). Спроецируйте данные для цифр 3, 5, 8 на двумерную плоскость с помощью алгоритмов t-SNE, Isomap и PCA. Отобразите проекции, используя разные цвета для разных цифр. Будьте готовы объяснить полученный результат.
8. Вычислите интеграл, используя метод Монте-Карло (используйте  $10^2, 10^3, 10^4, 10^5$  точек), найдите абсолютную и относительные ошибки (в сравнении с аналитическим решением)

$$\int_0^{\sqrt{2\pi}} x \sin\left(\frac{x^2}{2}\right) dx$$

9. Реализуйте метод градиентного спуска и примените его к функции Розенброка, начиная с точки  $[5, -3]$ . Используйте фиксированный размер шага (можете начать с 0.1) и уменьшайте его в два раза через каждые 100 итераций. Выведите конечный результат (после сходимости) и число итераций, необходимых для достижения этого результата.
10. АГибридное изображение получается сложением двух изображений, одно из которых было обработано фильтром низких частот, а другое фильтром высоких частот.



В этом случае есть параметр, меняя который можно варьировать то, как выглядит конечный результат – частота, по которой идет отсеивание в фильтрах, какие частоты оставить в одном изображении, а какие в другом.

Вам даны 4 пары изображений. Ваша задача – создать 1-2 гибридных изображения (не забудьте отобразить промежуточные результаты).

Полезные функции: `numpy.fft.fft2`, `numpy.fft.fftshift`, `numpy.fft.ifft2`