# Условия

* 5 обязательных лабораторных с дедлайнами 2-3 недели, всего 50 баллов
* N необязательных заданий с дедлайном 30.04.2025
* До экзамена допускаются сдавшие 2 лабораторные
* Баллы за задания учитываются, если сданы 3 лабораторные
* Штраф за просрочку каждые 2 недели (10, 7.5, 5, 3.5)
* Можно «фиксировать дедлайн» на неделю, показав более 30% лабораторной.
* **По каждой лабораторной и каждому заданию написать вывод**

# Лабораторные

## Лабораторная 1 - 10 баллов (дедлайн 14.03.2024)

Нужно сгенерировать 3 cлучайных *нелинейных* регрессии (разные функции!). Для обоих частей задания зафиксировать размерности 2-3х линейных слоев, их активации, функцию потери, lr и оптимизатор, кол-во эпох.

Первая часть задания: протестировать модель только с батчнормом с 3 разными размерами батча.

Вторая часть: протестировать модель с лучшим размером батча из первой части

1) без дропаута и без батчнорма,

2) только с батчнормом без дропаута,

3) только с дропаутом без батчнорма,

4) и с батчнормом и с дропаутом.

**Визуализировать** результаты (график).

В пару предложений записать выводы.

## Лабораторная 2 - 10 баллов (дедлайн 21.03)

Задача классификации **цветов** автомобилей.

Датасет DVM: <https://deepvisualmarketing.github.io/> **фронтальный**!

Ход работы:

1. Написать своими руками классификатор любой на выбор: ResNet, InceptionV3, DenseNet, MobileNet, ShuffleNet и обучить его на полученном датасете.
2. Также взять аналогичный классификатор, но предобученный на ImageNet или на Cityscapes и дообучить на собранном датасете. Выяснить, чей классификатор лучше.

Оценка качества производится при помощи F1\_macro, **требуется** получить F1\_macro > 0.8.

Сравнить полученное качество и сделать вывод.

## Лабораторная 3 - 10 баллов (дедлайн 18.04)

**Вариант 1 (простой) – 5 баллов**

* Обучить Encoder предсказывать локацию текста (место его действия), используя датасет COCO Locations: <https://www.kaggle.com/datasets/viacheslavshalamov/coco-locations> .
* Выберите не стандартный BERT, а какую-нибудь его модификацию (RoBERTa, DeBERTa, ModernBERT и т.п.)
* Показать примеры на тестовой выборке (10 шт) и на сочиненных вами примерах.
* Показать, как алгоритм отрабатывает на предложениях без явного упоминания локации (10 шт).
* Добиться качества accuracy >= 0,89.
* Сделайте вывод (с числами) о работе модели, когда локация присутствует в тексте и когда явно не указана.

**Вариант 2 (сложный) – 10 баллов**

* Обучить свой трансформер на задачу перевода текста с английского языка на русский (или наоборот). Используйте для этого torch.nn.Transformer.
* Требуется показать графики лоссов и показать примеры перевода для предложений и текстов разной сложности.
* **Обязательно быть готовыми запустить** модель прямо на практике на примерах от проверяющих.
* Заранее составить примеры генерации на 20 разнообразных примерах из тестовой выборки и на сочиненных вами примерах.

## Лабораторная 4 - 10 баллов (дедлайн 18.04.2025)

* Взять код с практики и переделать его под задачу Breakout <https://gymnasium.farama.org/environments/atari/breakout/#breakout>.
* Обучить модель, использовать разные оптимизации (препроцессинг стейтов и другие).
* Визуализировать лоссы и сколько очков выбивает модель на игре при обучении, также **показать графики** оценки состояний во время игры (+ скриншоты из игры для каких-нибудь локальных пиков оценки состояний). Показать видео-запуск модели в онлайне.
* Советую начинать заранее, потому что вероятно придется потратить время на обучение.
* Если ваша модель набирает 30 очков за одну жизнь - это в целом уже успех.
* Сделать вывод.

### Лабораторная 5 – 10 баллов (дедлайн 25.04.2025)

**Вариант 1 (простой) – 5 баллов**

* С помощью библиотеки Optuna настройте гиперпараметры сверточной нейронной сети (например, число слоев, размер фильтра и тд, оптимизатор — не гиперпараметр).
  + Гиперпараметры определите сами.
  + Гиперпараметров должно быть не менее 5.
  + Обучение проводить на датасете CIFAR-10.
  + Показать диаграмму важности гиперпараметров.

**Вариант 2 (сложный) – 10 баллов**

* Придумайте и опишите письменно или кодом векторное представление сверточной нейронной сети, его получение из архитектуры сети и восстановление архитектуры по векторному представлению.
  + Векторное представление должно быть фиксированного размера.
  + Восстановленная архитектура сети должна быть корректной, нужно описать, как обрабатываются ошибки.
  + Если вы предполагаете использование обучаемого векторного представления, то что делать, если оно не обучится? Как гарантировать результат обучения?
  + Заметьте, что сети бывают произвольного размера.
  + Между слоями есть связи (ResNet, Inception тоже хотелось бы уметь кодировать)

# Задания

## Задание 1 (5 баллов)

* Реализовать линейную или сверточную нейронную сеть из двух скрытых слоев.
* Обучить ее на датасете MNIST.
* Сравнить качество обучения при использовании различных функций активации (ReLU, GELU, Swish, Softplus обязательно, остальные – по желанию). Функции активации реализовывать!
* Построить график для функций активации точности от эпох..

## Задание 2 (7 баллов)

* Используя код практики или PyTorch, реализовать 3 оптимизатора (например, Nesterov, Adam, RMSProp).
* Реализовать любые 2 шедулера (например, StepLR и ExponentialLR). Проверить реализованное на сгенерированной регрессии.
* Показать графики изменение LR в течение 100 эпох.
* Доказать корректность реализации оптимизаторов.

## Задание 3 (5 баллов)

* Сгенерировать КАЧЕСТВЕННОЕ (не менее 512х512 и четкое) изображение вас в лесу, в эльфийском городе, на космическом корабле, на Марсе, в стиле стимпанк.
* Продемонстрировать, что сеть не разучилась генерировать других людей.
* Не использовать веб-сервисы с доступом по API.

### Задание 4 (10+ баллов)

* Взять 5+ современных моделей генерации музыки (c 2022 года), сгенерировать современную композицию в формате wav (лучше в стиле поп).
* Можно воспользоваться средами (как Ableton), но тогда требуется записать видео с инструкцией по генерации. Телеграм и прочие боты подходят, только если вы найдете к ним подробное описание архитектуры.
* Написать пару предложений про каждую архитектуру.
* За запуск каждой модели по 2 балла.

Задание 5 (в презентации)

Запустит LLM (5шт и более, если запускаете по API то попробовать 10 шт)

Сгенерировать валидное векторное изображение – файл SVG, возможно без заголовка – как текстовую последовательность тегов

Постараться получить адекватное зображение, оно должно быть сложнее фигуры и смайлика