

Санкт-Петербургский политехнический университет Петра Великого
Физико-механический институт

Переобучение нейронных сетей

L1&L2 регуляризация и DropOut

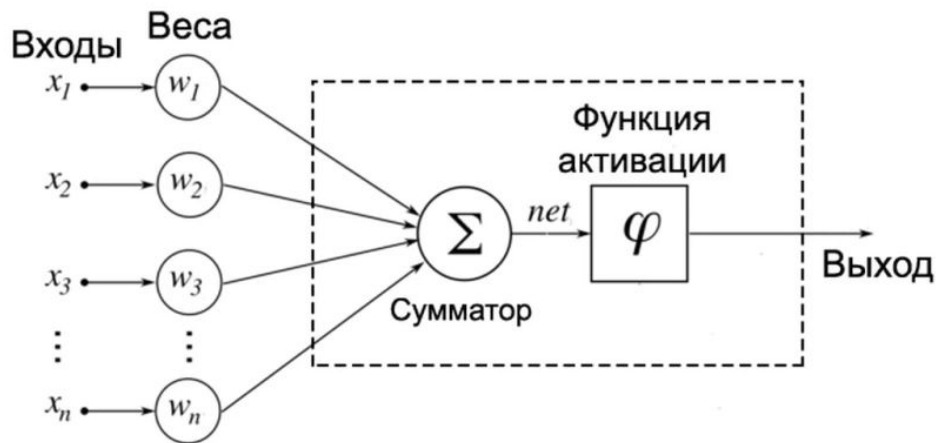
Выполнили студенты гр. 5030102/10101
Белоус Ф. В., Ушкарёв С. Д.

Постановка задачи

Провести исследование и оценить эффективность оптимизации процесса обучения нейронных сетей при использовании

L1 и L2 методов регуляризации и Dropout.

Провести качественные эксперименты для этих методов.



Переобучение

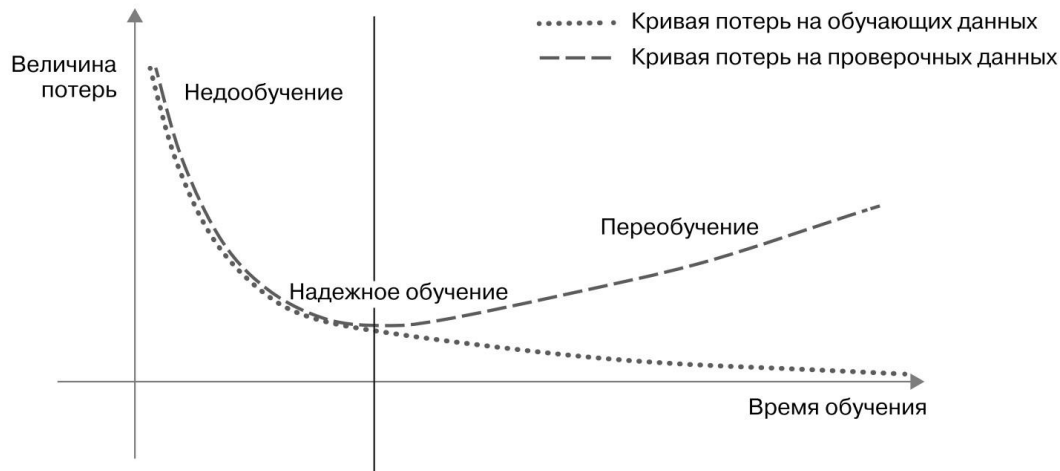
Переобучение нейронных сетей

явление, возникающее, когда модель "запоминает" обучающие данные, вместо того, чтобы обобщать закономерности и шаблоны, существующие в данных.

Причины возникновения

- Ограниченный объем обучающей выборки
- Проблемы мультиколлинеарности признаков

(* Мультиколлинеарность подразумевает сильно коррелированные признаки, из-за чего веса моделей становятся нестабильными и трудно интерпретируемыми.)



Шолле Ф. Глубокое обучение на Python. — Санкт-Петербург : изд-во "Питер", 2023.

L1 и L2 регуляризация

- **L1 и L2 регуляризация** – методы, основанные на добавлении к функции потерь штрафа на веса модели во время обучения.

$$L2(y_{pred}, y_{true}, w) = L(y_{pred}, y_{true}) + \lambda \sum_{i=1}^n w_i^2$$

$$\nabla L2(y_{pred}, y_{true}, w) = \nabla L(y_{pred}, y_{true}) + 2\lambda w$$

$$w^{i+1} = w^i - \mu \nabla L2_i(y_{pred}, y_{true}, w_i)$$

$$w^{i+1} = w^i - \mu (\nabla L(y_{pred}, y_{true}) + 2\lambda w_i)$$

$$w^{i+1} = (1 - 2\mu\lambda)w^i - \mu \nabla L(y_{pred}, y_{true})$$

L1 & L2 регуляризации на практике

Идея добавлять штраф пропорциональный совокупности весов является вполне естественной, однако стоит учитывать, что **L1** регуляризация приводит к занулению некоторых весов модели, но из-за использования модуля не позволяет использовать классический градиентный спуск в процессе обучения.

$$L1 = \text{Loss} + \lambda \sum_{i=1}^n |w_i|$$

$$L_1(y_{pred}, y_{true}, w) = L(y_{pred}, y_{true}) + \lambda \sum_{i=1}^n |w_i|$$

$$\nabla L_1(y_{pred}, y_{true}, w) = \nabla L(y_{pred}, y_{true}) + \lambda * \text{sign}(w)$$

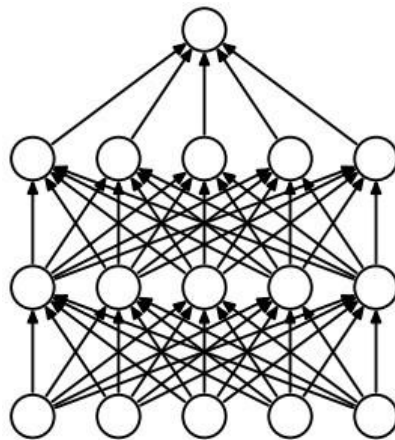
$$w^{i+1} = w^i - \mu \nabla L_{1i}(y_{pred}, y_{true}, w_i)$$

$$w^{i+1} = w^i - \underbrace{\mu L(y_{pred}, y_{true})}_{\text{Стремится к 0}} - \underbrace{\mu \lambda * \text{sign}(w_i)}_{\text{const}}$$

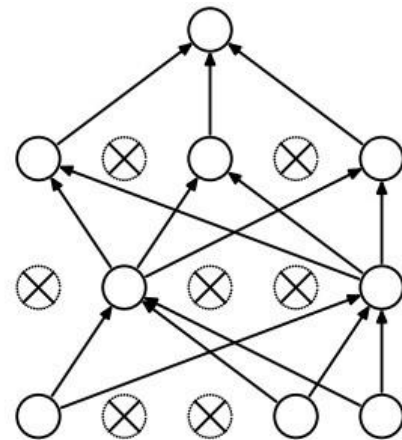
Метод исключения (Dropout)

- **Метод исключения или Dropout** – метод, главной идеей которого является обучение группы из нескольких сетей и усреднение полученных весов вместо обучения одной нейронной сети.

$$O_i = X_i \cdot f\left(\sum_{k=1}^{d_i} w_k x_k + b\right) = \begin{cases} f(\sum_{k=1}^{d_i} w_k x_k + b), & X_i = 1 \\ 0, & X_i = 0 \end{cases}$$



(a) Standard Neural Net



(b) After applying dropout.

DropOut на практике

На практике применяется два варианта метода DropOut: прямой и обратный, их различия состоят в том, в какой момент мы исключаем нейроны, при обучении или при тестировании.

Прямой

На этапе обучения: $O_i = X_i a\left(\sum_{k=1}^{d_i} w_k x_k + b\right),$

На этапе тестирования: $O_i = qa\left(\sum_{k=1}^{d_i} w_k x_k + b\right)$

Обратный

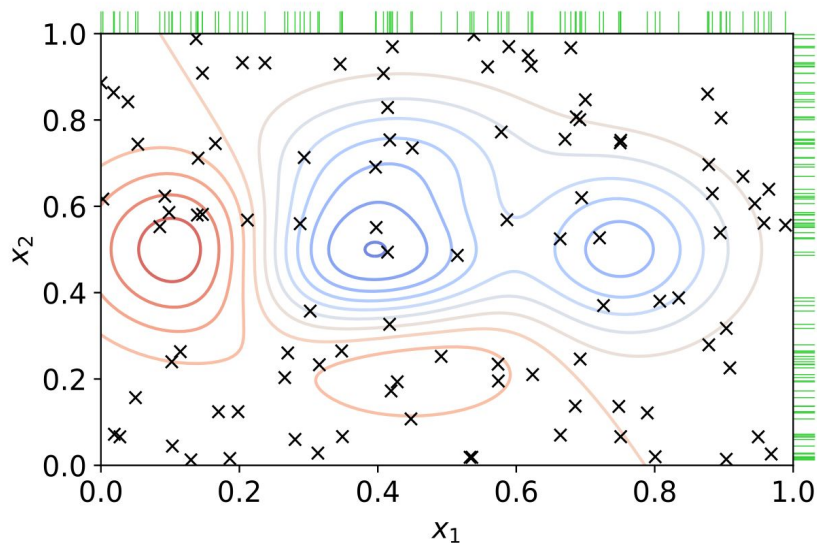
На этапе обучения: $O_i = \frac{1}{q} X_i a\left(\sum_{k=1}^{d_i} w_k x_k + b\right),$

На этапе тестирования: $O_i = a\left(\sum_{k=1}^{d_i} w_k x_k + b\right)$

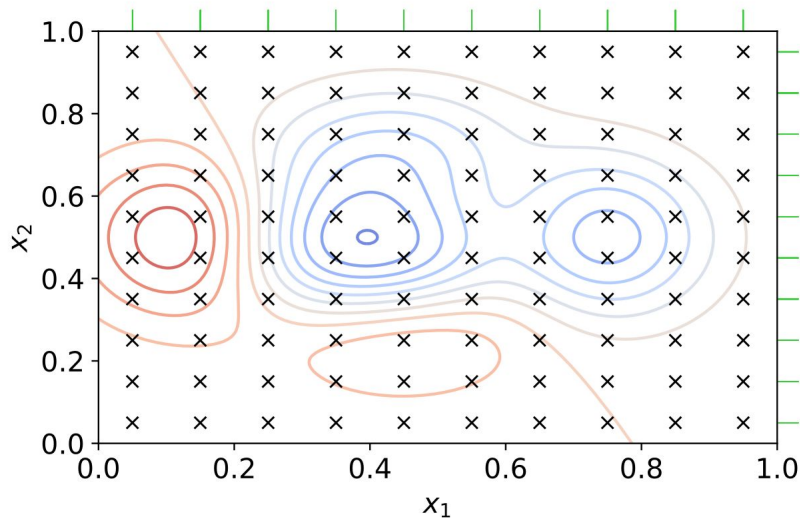
Подбор гиперпараметров

В обоих методах были использованы гиперпараметры (параметры задающиеся до начала процесса обучения и не изменяемые в процессе).
От грамотного подбора гиперпараметров зависит эффективность процесса обучения и его скорость.

Случайный подбор

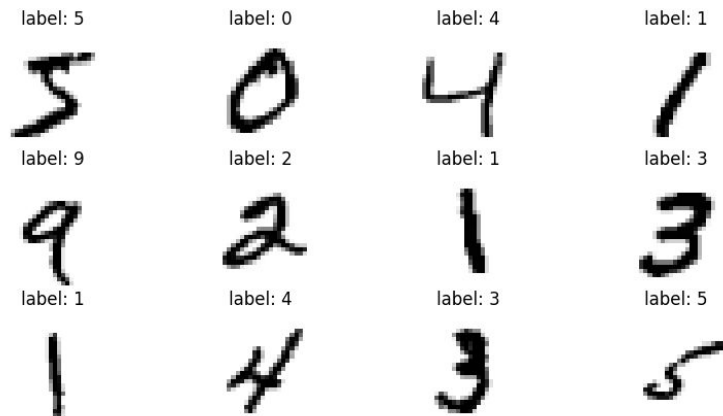


Линейный подбор



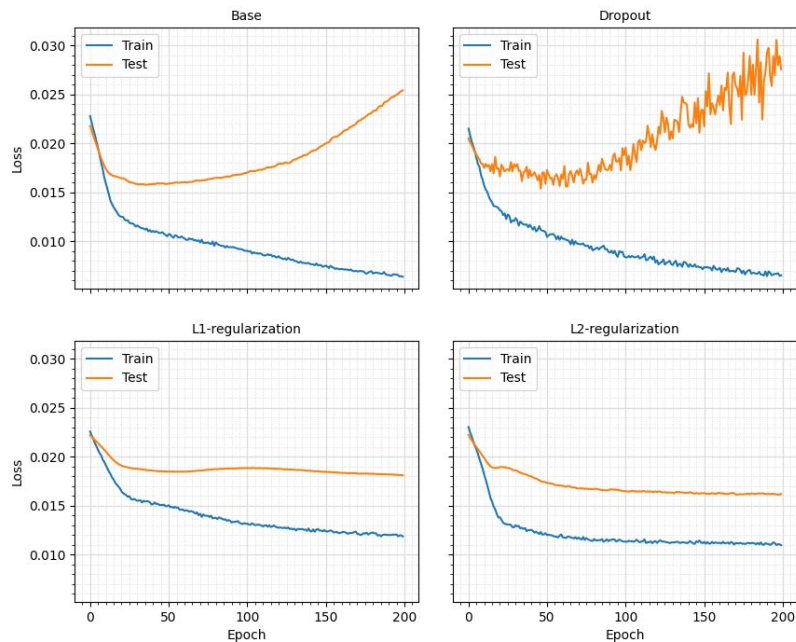
Описание эксперимента

В рамках эксперимента выполнялась задача регрессии на наборе данных Титаника и задача классификации изображений по данным набора MNIST

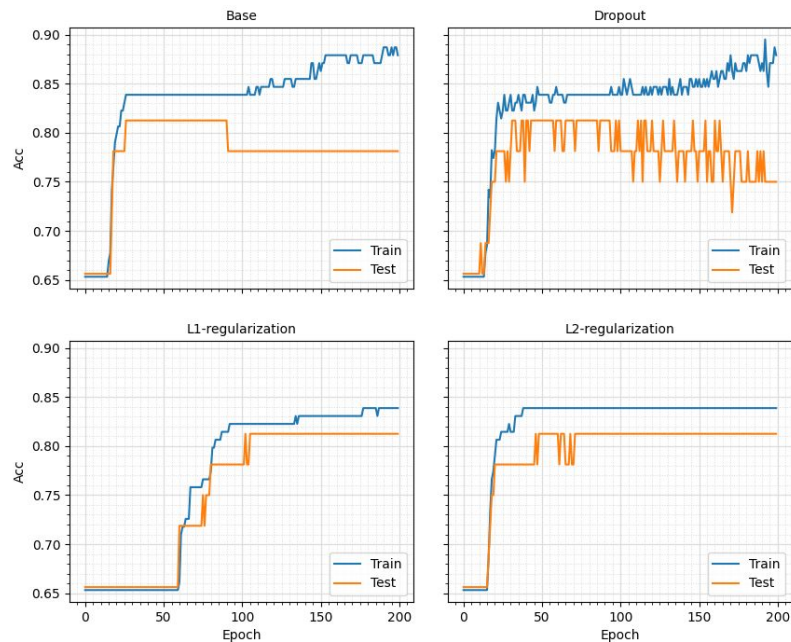


Результаты (Titanic)

Кривые обучения для различных методов регуляризации

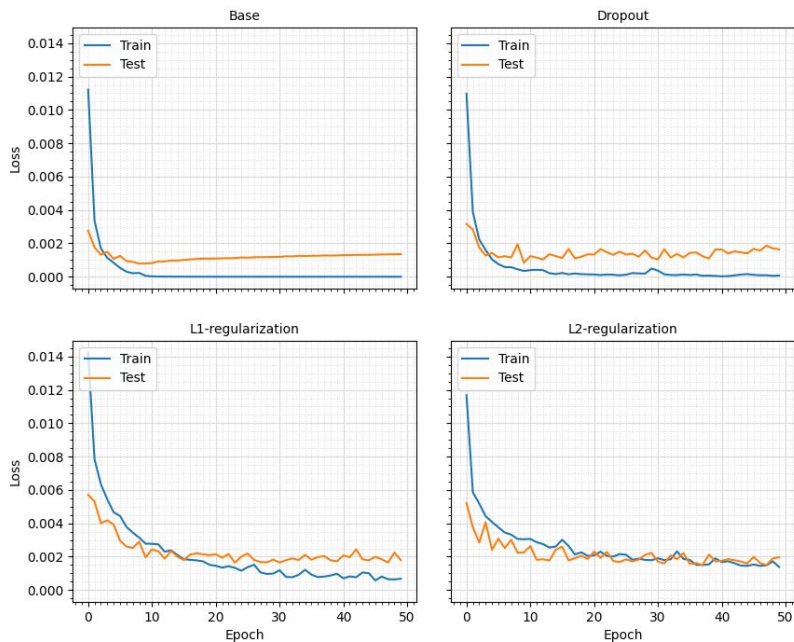


Кривые точности (acc) для различных методов регуляризации

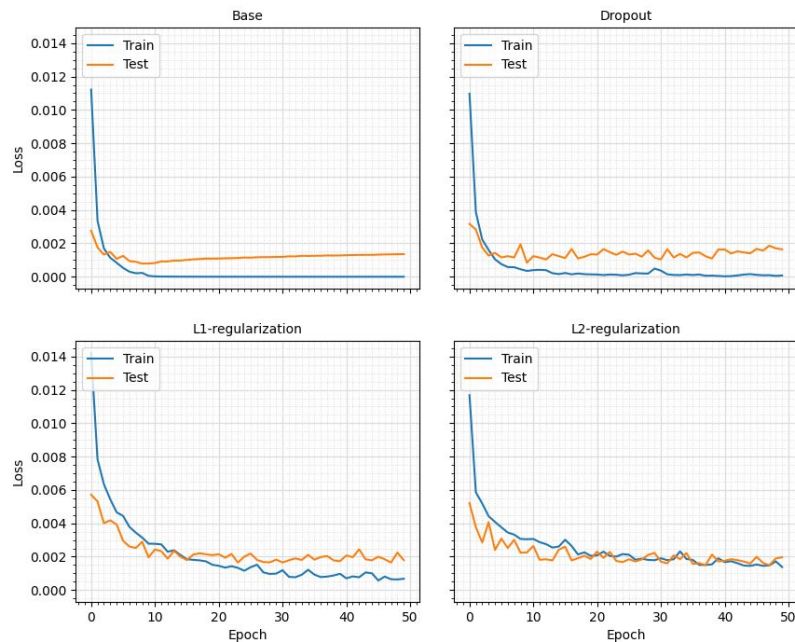


Результаты (MNIST)

Кривые обучения для различных методов регуляризации



Кривые обучения для различных методов регуляризации



Заключение

В данной работе были подробно рассмотрены методы регуляризации L1 & L2 и Dropout. По результатам исследований можно сделать вывод о **небходимости использования таких методов для оптимизации процесса обучения** и получения более эффективных моделей.

Данная работа показала, что у методов L1 & L2 и Dropout есть **свои достоинства и недостатки**, поэтому стоит заранее продумать, какой метод регуляризации использовать при разработке собственных нейронных сетей. **Обычно используется Dropout, как стандартный метод оптимизации процесса обучения, однако L1 и L2 регуляризации могут быть более эффективны в ряде случаев.**

Список использованных источников

- [1]БРЭ. Большая Российская Энциклопедия. Нейронные сети. — URL: <https://bigenc.ru/c/neironnyeseti-e734b3> ; (дата обр. 16.04.2024).
- [2]Шолле Ф. Глубокое обучение на Python. — Санкт-Петербург : изд-во "Питер", 2023.
- [3]Liu M., Cheng G. Early Stopping for Nonparametric Testing. — 2018. — URL: <https://arxiv.org/pdf/1805.09950> ; (дата обр. 01.05.2024).
- [4]Тихонов А. Н. О методах регуляризации задач оптимального управления // Доклады Академии наук СССР. — 1965. — т. 162, № 4. — с. 3—4.
- [5]Regularization methods / A. N. Tikhonov [и др.] // Numerical Methods for the Solution of Ill-Posed Problems. — Dordrecht : Springer Netherlands, 1995. — с. 7—63.
- [6]Srivastava N., Geoffrey Hinton Alex Krizhevsky I. S., Salakhutdinov R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting // Journal of Machine Learning Research. — 2014. — т. 15, № 56. — с. 1929—1958.
- [7]Википедия. MNIST (база данных). — URL: https://en.wikipedia.org/wiki/MNIST_database ; (дата обр. 06.05.2024).