Исследуем, как влияет батч-нормализация (БН) на различные модификации метода стохастического градиентного спуска (СГД) при обучении нейронных сетей.

Предположения:

- БН улучшает все методы – то есть ускоряет их сходимость
- Чем метод лучше работает сам по себе, тем БН слабее ускоряет его сходимость
- БН улучшает методы сильнее на глубоких сетях

1 Влияние БН на все методы

Проверим, улучшает ли БН все методы и если улучшает, то оценим насколько. Для этого выполним следующие задачи:

1. Выберем модификации СГД: СГДм (СГД с моментумом), Адам (адаптивный моментум), Адаград, Ададельта, RMSprop (root mean squared gradients)
2. Выберем датасеты: MNIST, CIFAR-10, cluttered MNIST(?)
3. Выберем архитектуры сети: MLP (multilayer perceptron, 3 полносвязных скрытых слоя по 100 нейронов), CNN (convolutional neural network, 2 сверточных слоя (32 фильтра 5x5 + макс-пулинг с окном 2x2) + 1 скрытый полносвязный с 256 нейронами).
4. Для каждого метода подберем примерный рейт на всех комбинациях датасетов, архитектур и использования БН
5. Запустим все методы на всех архитектурах и датасетах в комбинации с БН. Сохраним для всех запусков качество на тренировочной и валидационной выборках по эпохам (итерациям)
6. Составим большую таблицу тестового качества со всеми комбинациями датасетов, архитектур, методов и БН
7. Составим большую таблицу относительного улучшения качества всех методов

8. Нарисуем графики для визуализации работы БН (какие графики, как изобразить?)

Возникли следующие проблемы:

1. Для выбранной CNN БН методы переобучаются на CIFAR-10 при повышении рейта. Возможные пути решения:
   - брать рейт поменьше
   - добавить dropout
   - уменьшить количество нейронов
   - выбрать датасет посложнее (на MNIST не тестировали, скорее всего, там тоже переобучается)

2 Влияние глубины сети на БН

Проверим, как улучшает БН методы на глубоких сетях. Для этого выполним следующие задачи:

1. Выберем те же методы, что и в предыдущем эксперименте (СГД, СГДм, Адам, Адаград, Ададельта, RMSprop)
2. Возьмем датасеты из предыдущего эксперимента: MNIST, CIFAR-10, cluttered MNIST(?)
3. Выберем архитектуры для глубокой сети: (?)
4. Подберем рейт для всех методов на всех комбинациях архитектур, датасетов и БН
5. Запустим все методы на всех архитектурах и датасетах в комбинации с БН. Сохраним для всех запусков качество на тренировочной и валидационной выборках по эпохам (итерациям)
6. Составим большую таблицу тестового качества со всеми комбинациями датасетов, архитектур, методов и БН
7. Составим большую таблицу относительного улучшения качества всех методов
8. Сравним результаты с результатами из предыдущего эксперимента

3 Влияние БН на методы, использующие моментум

Гипотеза: БН и моментум уменьшают дисперсию градиента, поэтому их сочетание не приводит к дополнительному улучшению. Для проверки проведем следующие эксперименты:

1. Отберем методы, использующие моментум: СГДм, Адам, Ададельта (в некоторой степени)
2. Все последующие эксперименты нужно проводить, если из предыдущих результатов мы получили, что на этих методах БН дает меньшую прибавку, чем на остальных (?)
3. Запустим на MLP (3 слоя по 100 нейронов) методы СГД, СГДм с БН и без БН – сохраним отклонения стохастических градиентов от полных на выбранных итерациях по 50 запускам, а затем усредним их по запускам
4. Нарисуем график для полученных дисперсий 4-х методов

Возникли следующие проблемы:

1. Получили, что БН увеличивает дисперсию, а моментум уменьшает. Возможные пути решения:
   - Отказаться от эксперимента, так как БН меняет архитектуру сети (оптимизируемую функцию) (число параметров в сети с БН много больше)
   - Полный градиент при БН считать не сразу по всей выборке, а суммировать по батчам (так как при проходе по всей выборке точнее оцениваются средние и дисперсии, то возможно из-за этого стохастический градиент по одному батчу так сильно отклоняется от полного градиента)

4 Прыжки БН Адама

Гипотеза: на простых датасетах БН Адам имеет на графиках непредсказуемые скачки, которые исчезают при более тщательном подборе параметров метода Адам. Для проверки и подтверждения выполним следующие задачи:

1. Выберем простой датасет: MNIST
2. Выберем архитектуры сети: MLP, CNN (неглубокие)
3. Добавим в исходники сохранение v_t по итерациям (скользящее среднее квадратов градиентов)
4. Запустим БН Адам несколько раз при стандартных параметрах – убедимся в наличии скачков, посмотрим, как себя ведут v_t в эти моменты
5. Запустим Адам при стандартных параметрах
6. Запустим БН Адам несколько раз при $\epsilon = 10^{-4}$ и $\epsilon = 10^{-2}$ – убедимся, что скачки стали меньше, и что v_t стали больше
7. Проверим, как влияет на качество увеличение ϵ и сравним все со стандартным методом Адам

Вопросы:

1. Нужно показать, что на более сложных датасетах БН Адам не прыгает?