

# Results of experiments with MNIST

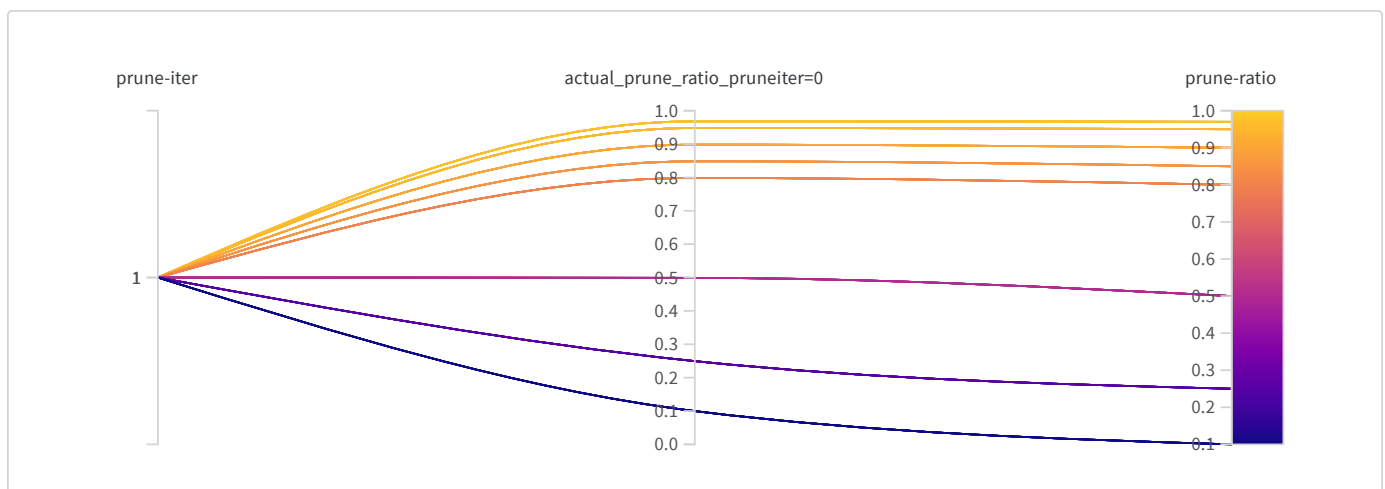
Boris Panfilov

## Desired pruning vs actual pruning

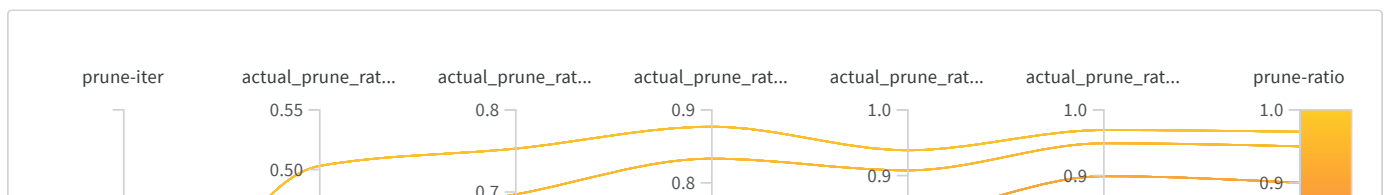
### Описание

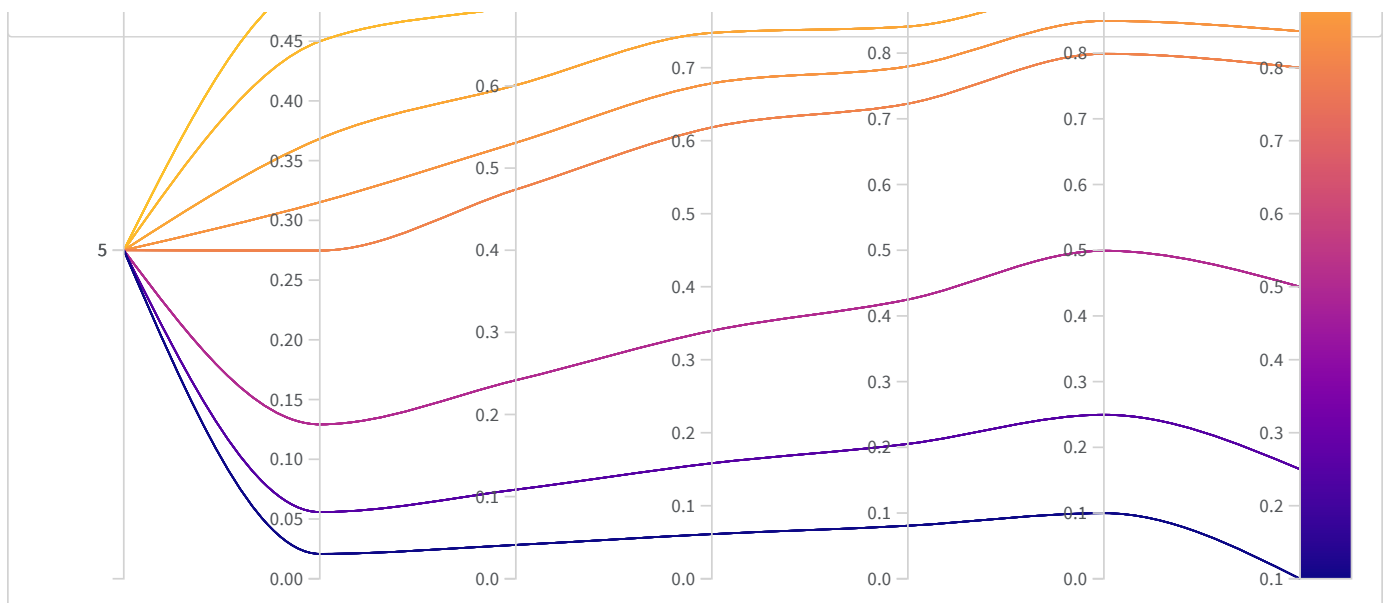
Я обучал модели на датасете MNIST и обрезаю разный процент весов. В статье было описано, что если на шаге нужно обрезать  $p\%$  весов, то мы в каждом слое кроме последнего зануляем  $p\%$  весов, а в последнем  $p/2\%$ . Поэтому можно заметить, что на самом деле обрезанных весов немного меньше, чем мы просили. Ниже есть 2 графика, они отображают долю весов, которую просили обрезать и какая доля обрезалась на самом деле, при обрезании одной итерацией и при обрезании 5-ю итерациями соответственно.

### Onehot pruning



### Iterative pruning





## Вывод

В процессе обучения и правда обрезается нужное количество весов. Сложный процент работает!

## No reinitialisation + random pruning

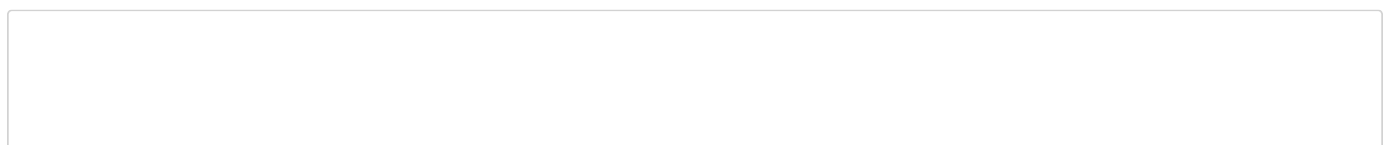
### Описание

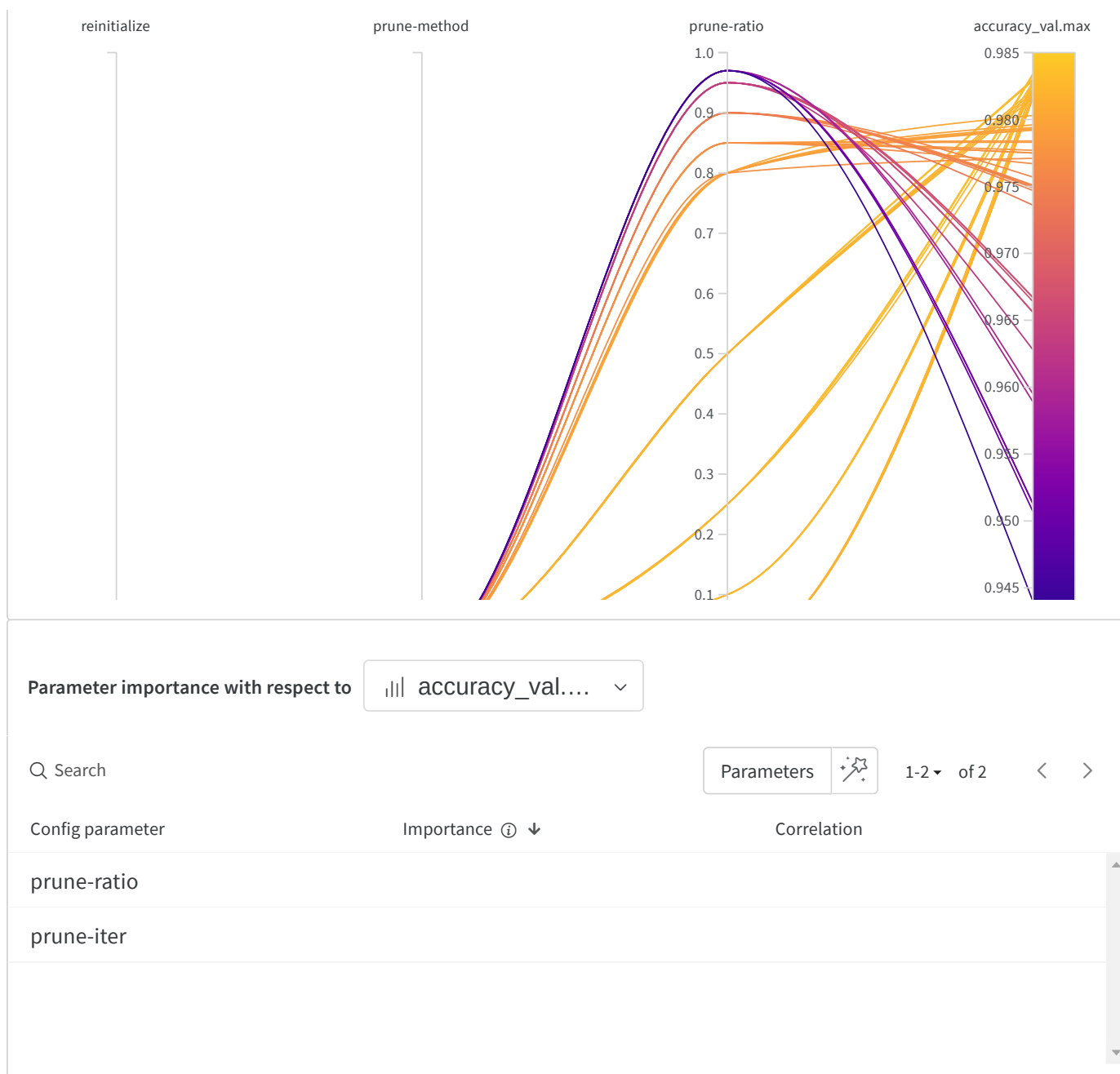
Тут я собрал результаты обучения моделей, в которых был использован следующий алгоритм:

- 1) обрезаем  $p\%$  весов
- 2) присваиваем всем значениям изначальные и дообучаем модель
- 3) повторяем 1-2  $n$  раз
- 4) присваиваем начальным весам изначальные веса
- 5) обучаем модель

То есть смысл этих моделей показать важность обрезки именно меньших по модулю весов, а не рандомных

### График





## Вывод

Нетрудно заметить отрицательную корреляцию и если учитывать весь диапазон `prune-ratio`, и если выбрать некритические уровни обрезки моделей - (0% - 85%). А значит можем сделать вывод, что при таком методе обучения уменьшение весов модели влечет ухудшение качества.

## Reinitialisation + L1 pruning

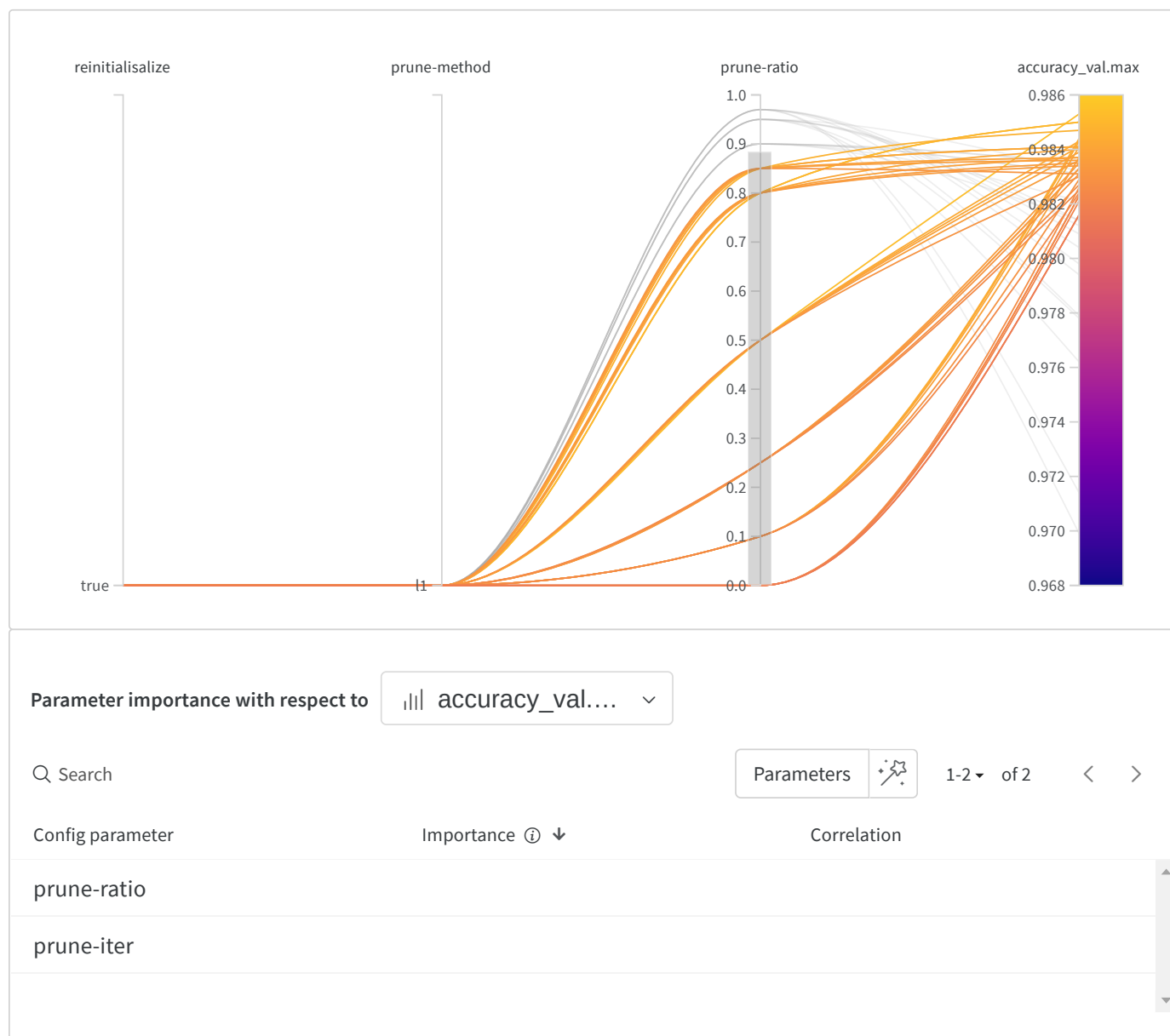
### Описание

Тут я собрал результаты обучения моделей, в которых был использован следующий алгоритм:

- 1) обрезаем  $p\%$  весов
- 2) присваиваем всем значениям изначальные и дообучаем модель
- 3) повторяем 1-2  $n$  раз
- 4) присваиваем начальным весам рандомные веса
- 5) обучаем модель

То есть смысл этих моделей показать важность инициализации модели изначальными весами в конце

## График



## Вывод

Если рассматривать все модели, то корреляция по прежнему будет отрицательной. Тем не менее, если оставить лишь некритические уровни моделей, то мы получим положительную корреляцию порядка 0.596 Это уже довольно круто, но в статье говорится, что это еще не лучший алгоритм, так что идем дальше

## No reinitialisation + L1 pruning (winnig tickets should be here)

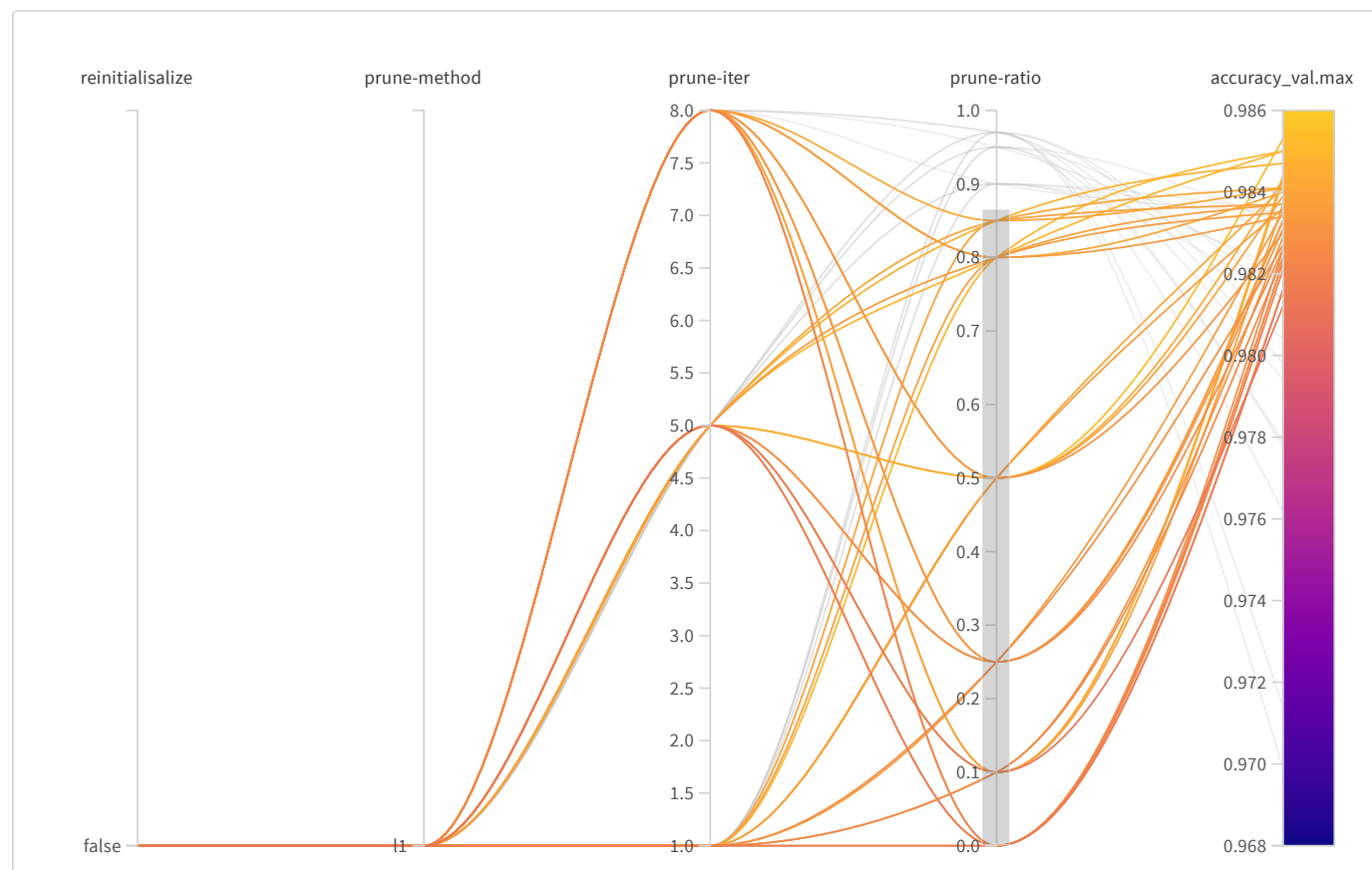
### Описание


Тут я собрал результаты обучения моделей, в которых был использован следующий алгоритм:

- 1) обрезаем  $p\%$  весов
- 2) присваиваем всем значениям изначальные и дообучаем модель
- 3) повторяем 1-2  $n$  раз
- 4) присваиваем начальным весам изначальные веса
- 5) обучаем модель

Исходя из статьи именно тут мы должны получить наилучший результат

### График



Parameter importance with respect to <span>    accuracy_val....</span> <span>▼</span>		
Q Search	Parameters 	1-2 ▼ of 2 < >
Config parameter	Importance ⓘ ↓	Correlation
prune-ratio		
prune-iter		

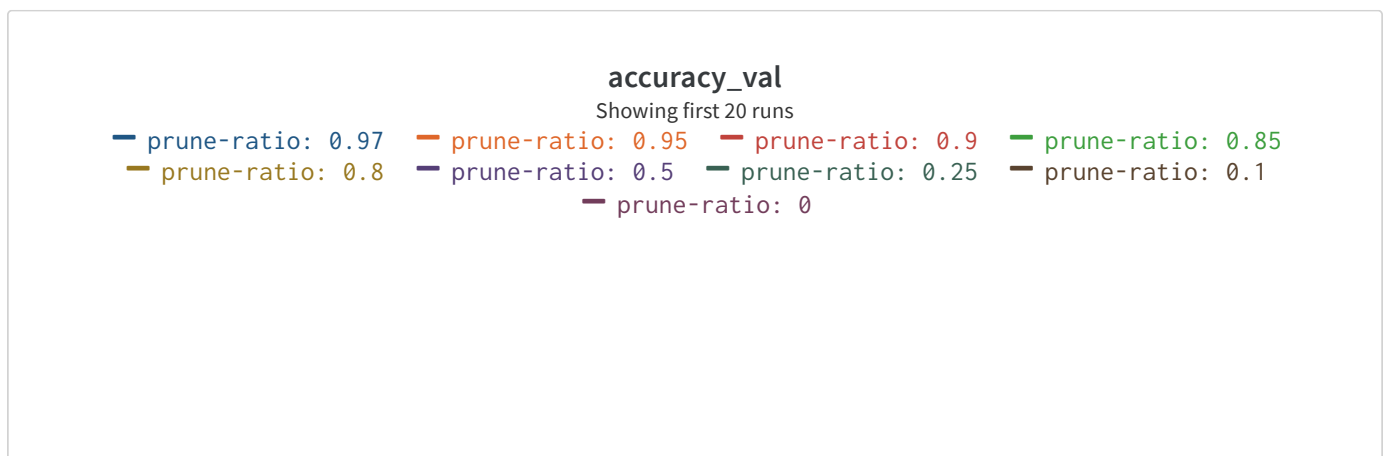
## Вывод

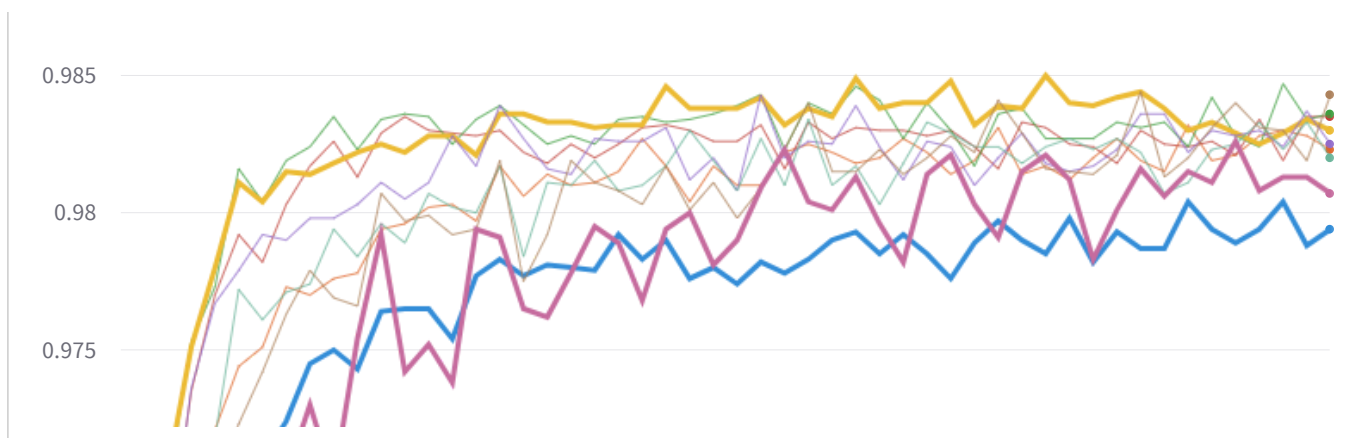
Если рассматривать все модели, то корреляция получается по прежнему отрицательной. Но оно и логично, потому что вряд ли у модели с 5% весов удастся работать так же хорошо, как и со 100%. Тем не менее, если мы как обычно рассмотрим только модели с уровнем обрезки не более 85%, то мы получим наибольшую корреляцию - 0.619. То есть мы можем сделать вывод, что мы нашли алгоритм определяющий выигрышные билеты (наиболее важные веса), такие что при обучении только с ними мы не теряем качество.

## Results with different random states

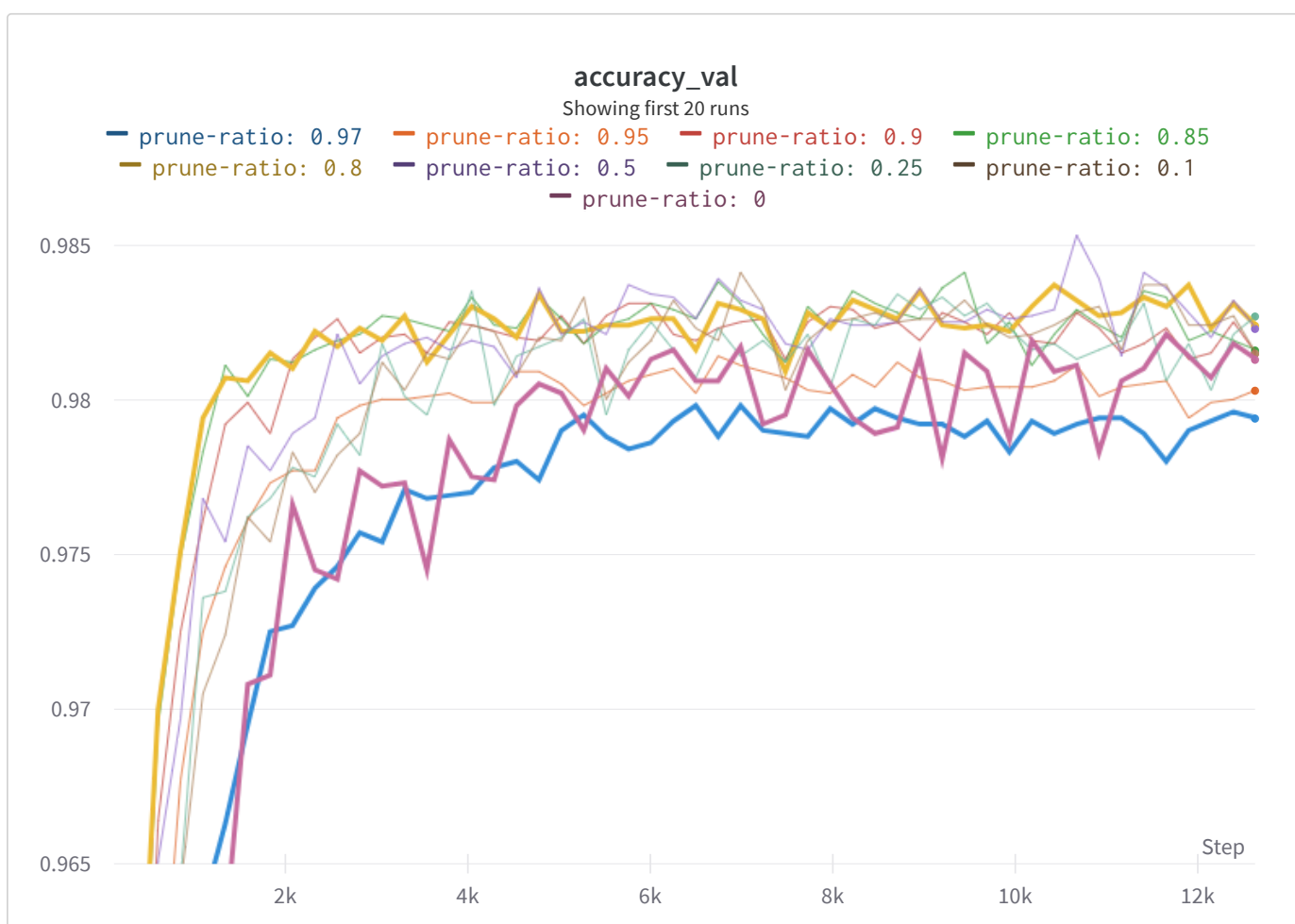
### Random state = 1

В качестве показателя для каждого prune-ratio взял максимальное accuracy\_value из всех моделей





Random state = 3



Еще какие-то мысли

Created with ❤️ on Weights & Biases.

<https://wandb.ai/bspanfilov/experiment%20from%20the%20title/reports/Results-of-experiments-with-MNIST--VmldzozNzMyOTIw>