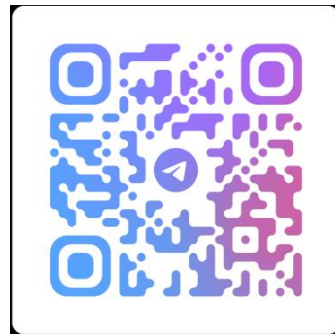


Основы практического использования нейронных сетей.

Лекция 6. Разработка эффективных НС.

Дмитрий Буряк.
к.ф.-м.н.
dyb04@yandex.ru



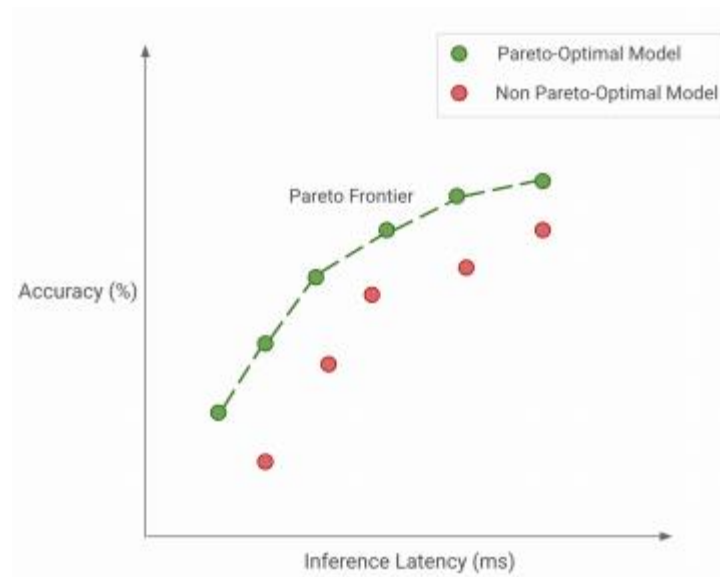
Эффективность НС

□ Эффективность обученной НС

- Размер сети
- Латентность
- Число MAC/FLOP

□ Эффективность проведения обучения

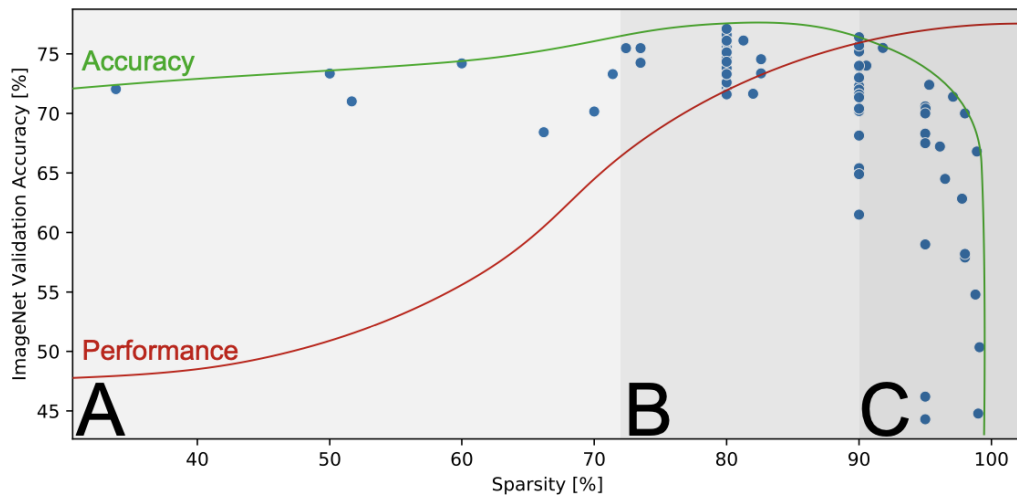
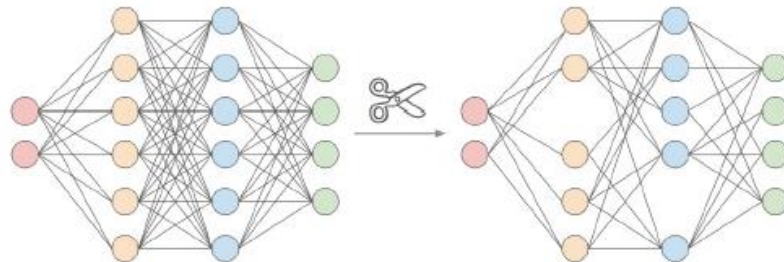
- Размер сети
- Точность



Прореживание НС

Прореживание (pruning) НС

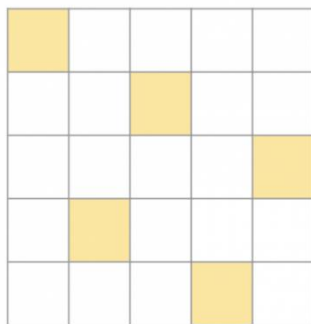
- Удаление элементов архитектуры НС
 - Связи, нейроны, фильтры, ...



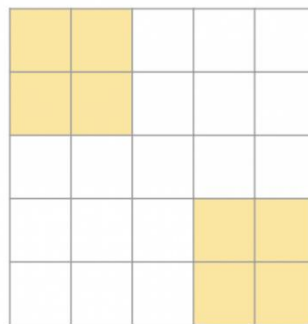
Качество на тестовой выборке от степени прореженности
(ResNet50 на ImageNet, точность топ-1)

Классификация по элементам прореживания

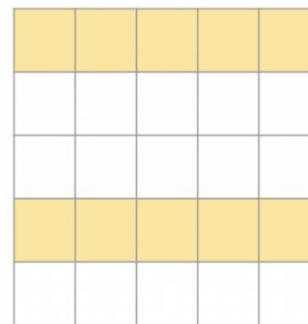
- ❑ Неструктурированный прунинг: прореживание на уровне отдельных весов.
- ❑ Структурированный прунинг: прореживание на уровне целых сверточных фильтров и слоев.



Unstructured



Blocked

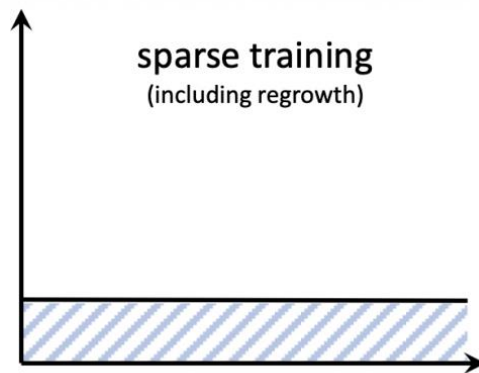
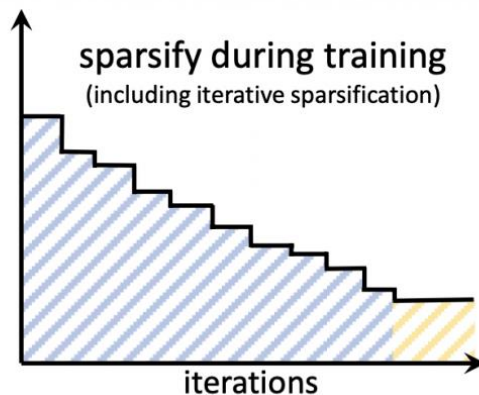
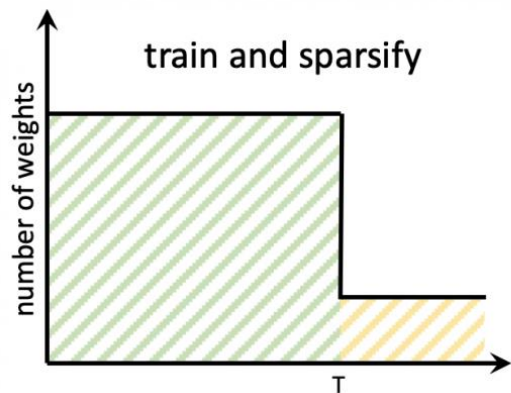


Axis-wise

- ❑ Глобальное прореживание
- ❑ Послойное прореживание

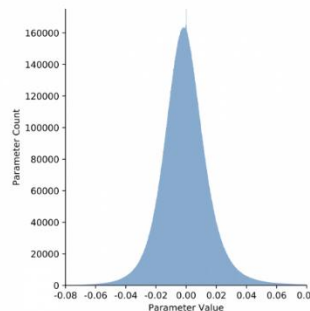
Классификация по типу обучения

- ☐ Однократное прореживание.
- ☐ Итеративное прореживание.
- ☐ Обучение разреженной модели с самого начала

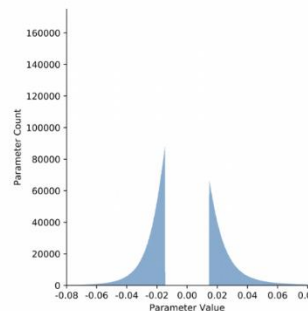


Критерии отбрасывания весов

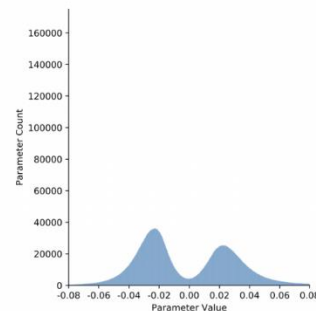
- ❑ Без использования обучающих данных



(a) Dense Network (76.0%)



(b) 70% Pruned (36.1%)



(c) After 3-epoch Retraining (71.4%)

Применение Magnitude-based прореживания для ResNet50 на базе ImageNet

- ❑ С использованием обучающих данных:
 - на основе дисперсии выходов нейронов
 - на основе величины активации
 - на основе взаимной корреляции между весами

- ❑ Training-aware:

- Optimal Brain Damage

$$\mathcal{L}(\mathbf{w}) = \mathcal{L}_0(\mathbf{w}) + \left(\frac{\partial \mathcal{L}}{\partial \mathbf{w}} \right)^T \cdot \delta \mathbf{w} + \frac{1}{2} \delta \mathbf{w}^T \cdot \mathbf{H} \cdot \delta \mathbf{w} + \mathcal{O}(\|\delta \mathbf{w}\|^3)$$
$$\min_{i \in \mathcal{W}} \min_{\delta w} \left(\frac{1}{2} \delta \mathbf{w}^T \cdot \mathbf{H} \cdot \delta \mathbf{w} \mid e_i^T \cdot \delta \mathbf{w} + w_i = 0 \right)$$

удалить веса с наименьшим $h_i^{1/2} w_i$

Стандартный алгоритм прореживания

Algorithm 1: Standard Network Pruning with Fine-Tuning

Data: Pre-trained dense network with weights W , inputs X , number of pruning rounds N , fraction of parameters to prune per round p .

Result: Pruned network with weights W' .

```
1  $W' \leftarrow W$ ;  
2 for  $i \leftarrow 1$  to  $N$  do  
3    $S \leftarrow \text{compute\_saliency\_scores}(W')$ ;  
4    $W' \leftarrow W' - \text{select\_min\_k}(S, \frac{|W'|}{p})$ ;  
5    $W' \leftarrow \text{fine\_tune}(X, W')$   
6 end  
7 return  $W'$ 
```

Стандартный алгоритм прореживания обученной НС

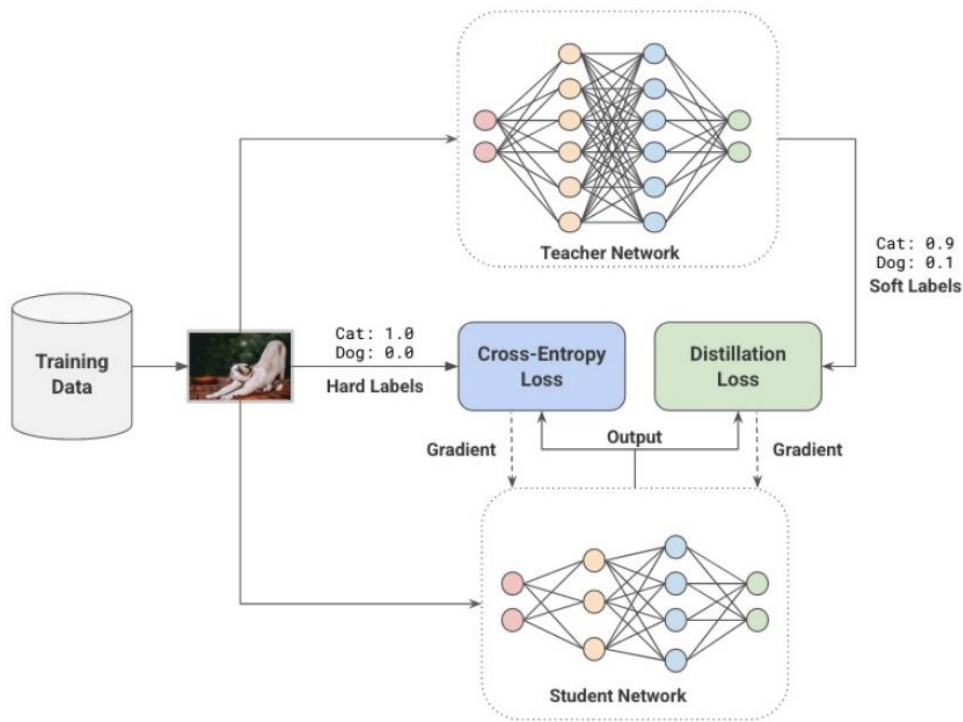
Особенности алгоритмов прореживания НС

- ☐ Критерий выбора элементов для прореживания
 - оценка влияния на функцию потерь
 - абсолютная величина, вторая производная ...
- ☐ Элементы НС для удаления
- ☐ Распределение доли удаляемых элементов по НС
- ☐ Расписание проведения прореживания
- ☐ Возможность восстановления связей
- ☐ «Гипотеза о лотерейном билете»
 - В любой большой НС существует подсеть, которая может быть обучена с такой же эффективностью
- ☐ Сложно добиться реального ускорения для прореженной НС
- ☐ Отсутствие строгой теории, обеспечивающей успех какой-то стратегии

Дистилляция НС

Дистилляция НС. Основная идея

- ❑ Стратегия обучения на выходы сети «учителя».
- ❑ Сеть «учитель» – «большая» НС/ансамбль НС
- ❑ Сеть «ученик» - целевая сеть, которая должна удовлетворять требованиям на число параметров и вычислительную сложность



Дистилляция НС. Функция потерь

$$L = \lambda_1 \cdot L_{\text{ground-truth}} + \lambda_2 \cdot L_{\text{distillation}}$$

- $L_{\text{ground-truth}}$ – базовая функция потерь на основной обучающей выборке .
- $L_{\text{distillation}}$ – функция потерь относительно выходов сети «учитель».
- Задание меток относительно результатов сети «учитель»

$$Y_i^{(t)} = \frac{\exp(Z_i^{(t)}/T)}{\sum_{j=1}^n \exp(Z_j^{(t)}/T)}$$

T - температура

$$\begin{aligned} L &= \lambda_1 \cdot L_{\text{ground-truth}} + \lambda_2 \cdot L_{\text{distillation}} \\ &= \lambda_1 \cdot \text{CrossEntropy}(Y, Y^{(s)}; \theta) + \lambda_2 \cdot \text{CrossEntropy}(Y^{(t)}, Y^{(s)}; \theta) \end{aligned}$$

Автоматизированная настройка гиперпараметров НС

Гиперпараметры НС

- ❑ Параметры НС, которые не настраиваются в процессе обучения.
- ❑ Влияют на емкость (способность к обобщению), скорость обучения и объем требуемой памяти НС.
- ❑ Параметры, определяющие архитектуру НС (число и размеры слоев)
- ❑ Параметры алгоритма обучения

Определение значений гиперпараметров

☐ «Ручной»

- требует понимания влияния гиперпараметров на свойства и характеристики сети;
- использование подтверждающей выборки, кроссвалидация.

☐ Автоматический

- требует большой объем дополнительных вычислений.

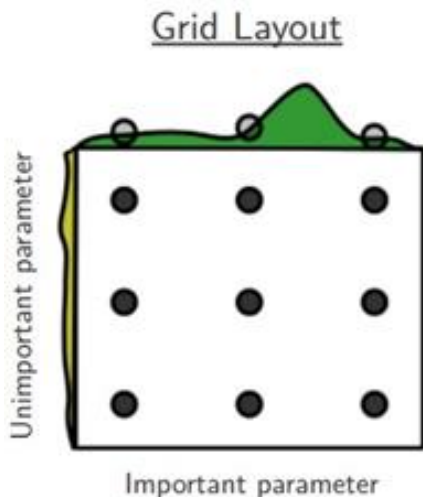
Автоматический поиск значений гиперпараметров

- ❑ Оптимизация гиперпараметров (Hyper-Parameter Optimization – HPO)
- ❑ Создание «обертки» для тестирования различных наборов значений гиперпараметров и выбора оптимального.
- ❑ Функция оптимизации – значение ошибки на подтверждающей выборке.
- ❑ «Обертка» имеет собственные гиперпараметры.

- ❑ Подходы к формированию наборов значений гиперпараметров:
 - регулярный
 - случайный

Регулярное формирование наборов значений гиперпараметров

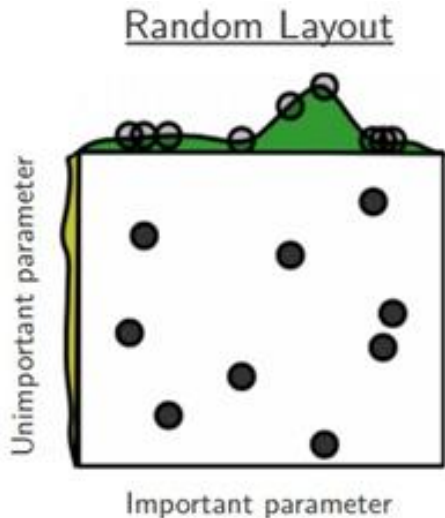
- ☐ Выбор фиксированного числа значений каждого гиперпараметра.
- ☐ Формирование комбинаций выбранных значений.



- ☐ Повторный поиск на границах интервала и на более частой сетке в выбранных областях.
- ☐ Низкая эффективная размерность пространства гиперпараметров.
- ☐ Различная значимость гиперпараметров на емкость НС.
- ☐ Экспоненциальный рост числа вычислений.
- ☐ Эффективен для малого числа гиперпараметров.
- ☐ Хорошо распараллеливается.

Случайное формирование наборов значений гиперпараметров

❑ Случайный выбор вектора значений гиперпараметров в соответствии с заданным распределением.



❑ Повторный поиск в выбранных областях.

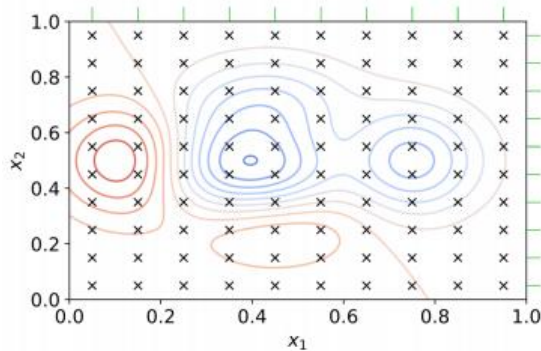
❑ Устойчив к пространствам низкой эффективной размерности.

❑ Более эффективный метод относительно числа требуемых вычислений.

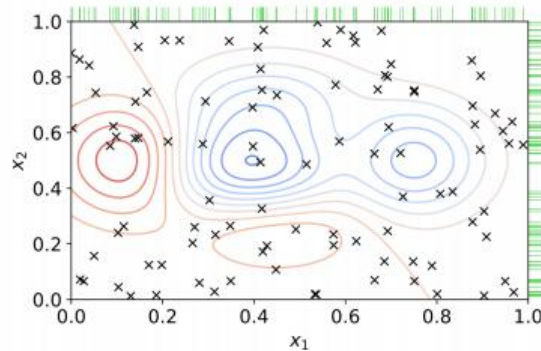
❑ Хорошо распараллеливается.

Байесовская оптимизация

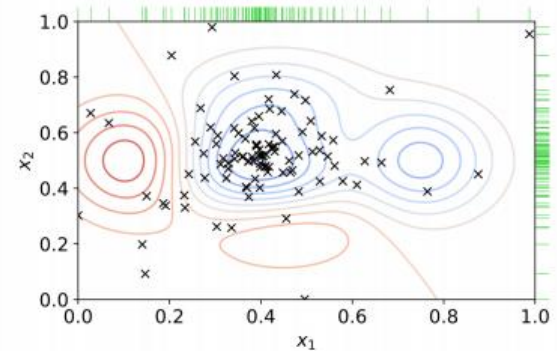
- ❑ Выбор значений для проверки на основе результатов тестирования предыдущих значений гиперпараметров.
- ❑ Построение отдельной функции для выбора значений для проверки.



(a) Grid Search



(b) Random Search



(c) Bayesian Optimization

Вопросы

- ❑ Для чего используют алгоритмы прореживания?
- ❑ В чем состоит основная идея дистилляции?
- ❑ Почему при оптимизации гиперпараметров случайное формирование наборов значений часто эффективнее чем регулярное (по решетке)?