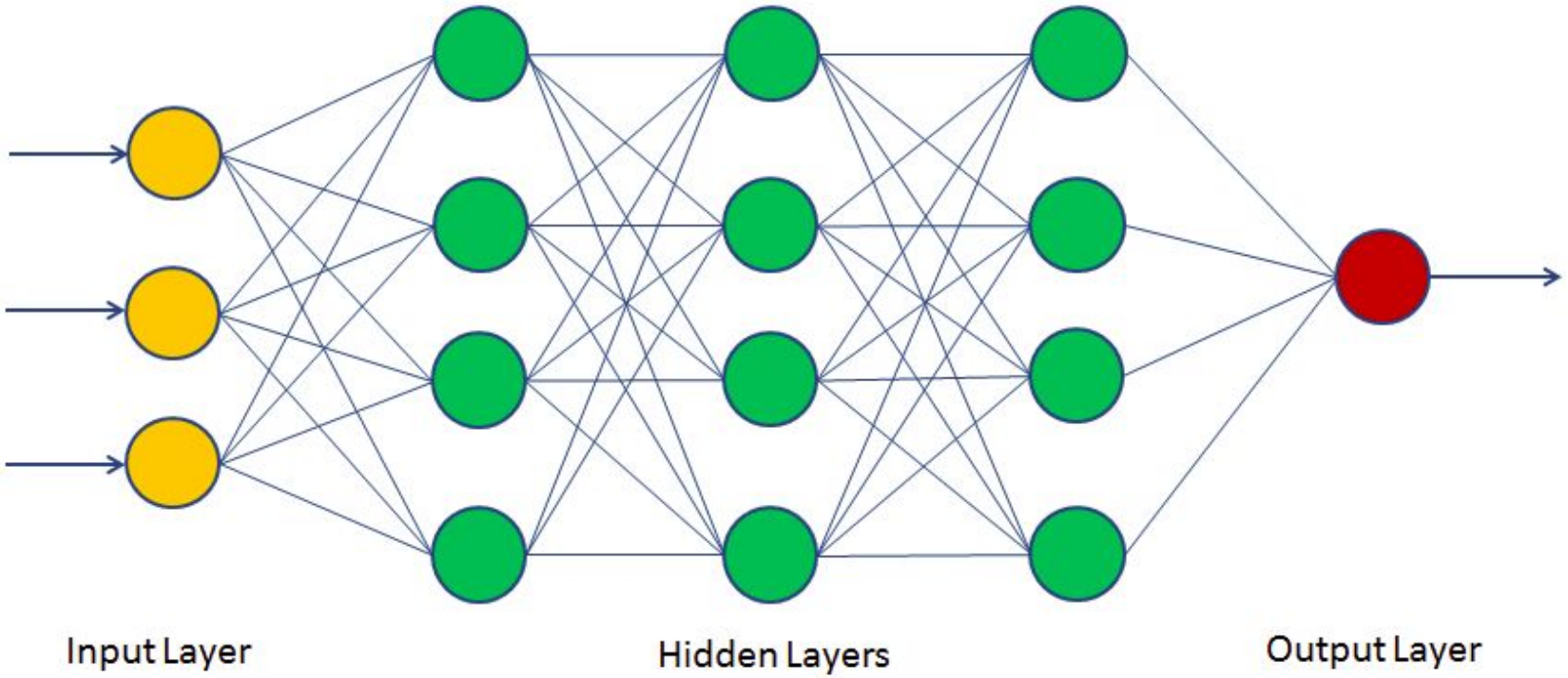
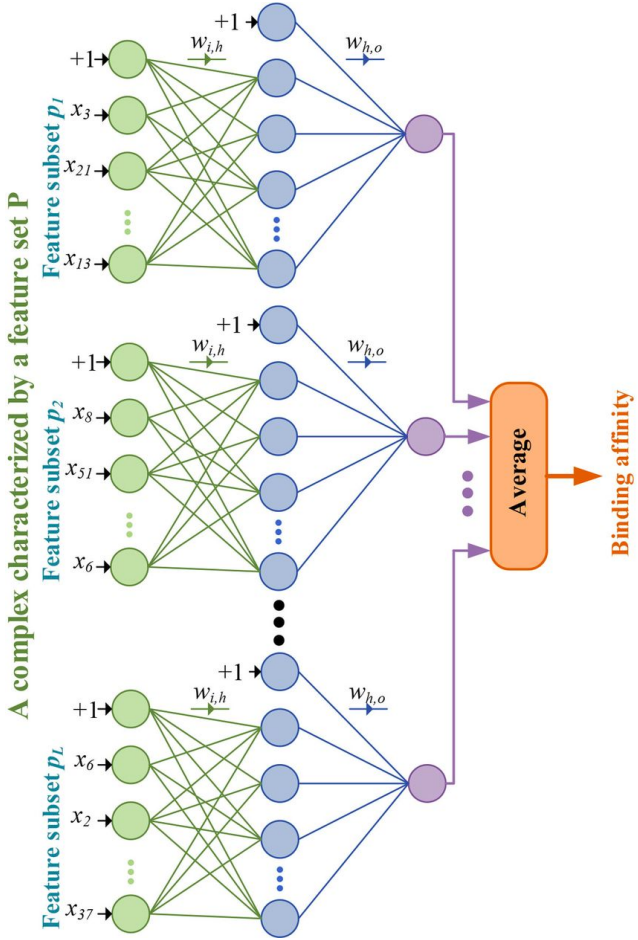
Ансамблирование нейронных сетей

Болотин Арсений 182

Neural network



Ансамблирование - метод при котором несколько моделей обучаются на разных данных и/или разными методами для решения одной и той же проблемы и объединяются для получения лучших результатов.



Bias Variance trade-off

$$y = f(x) + \epsilon \qquad \epsilon \sim N(0, \sigma^2)$$

$$E[(y - \hat{f}(x))^2] = (Bias[\hat{f}(x)])^2 + Var[\hat{f}(x)] + \sigma^2$$

$$Bias[\hat{f}(x)] = E[\hat{f}(x) - f(x)]$$

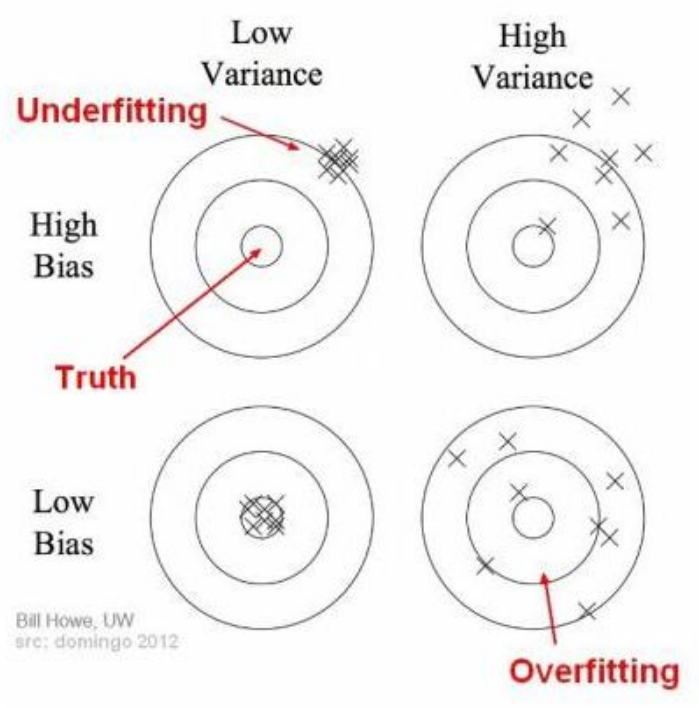
$$Var[\hat{f}(x)] = E[\hat{f}(x)^2] - (E[\hat{f}(x)])^2$$

Bias Variance trade-off

Нейронная сеть - метод с низким смещением и **высокой дисперсией**.

Решение проблемы высокой дисперсии нейронных сетей - ансамблирование нескольких моделей.

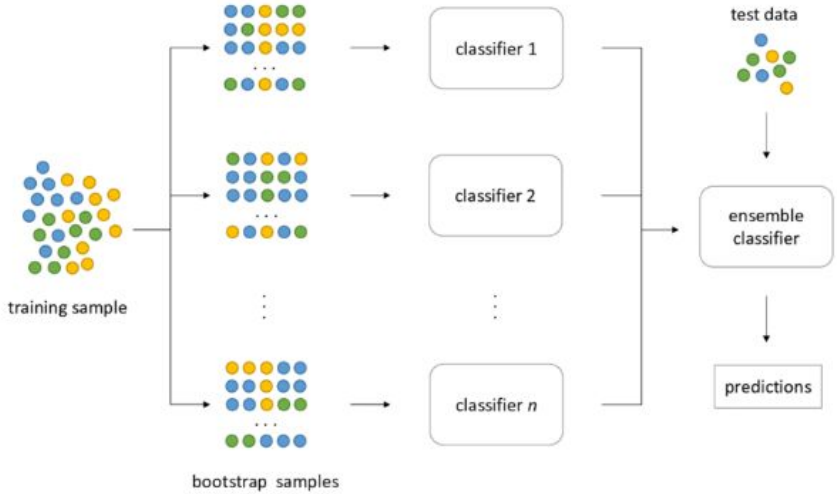
Модели должны быть разными!



Как ансамблировать?

Регулируем:

- Обучающие выборки для моделей
- Комбинирующая функция
- Модели



Обучающая выборка для моделей

Обучение одного и того же алгоритма на разных выборках данных.

- Random Training Subset Ensemble
- K-fold Cross-Validation Ensemble
- Bagging - Bootstrap Aggregation

Random Training Subset Ensemble

- Для каждой модели генерируем случайную подвыборку из изначальных данных.

K-fold Cross-Validation Ensemble

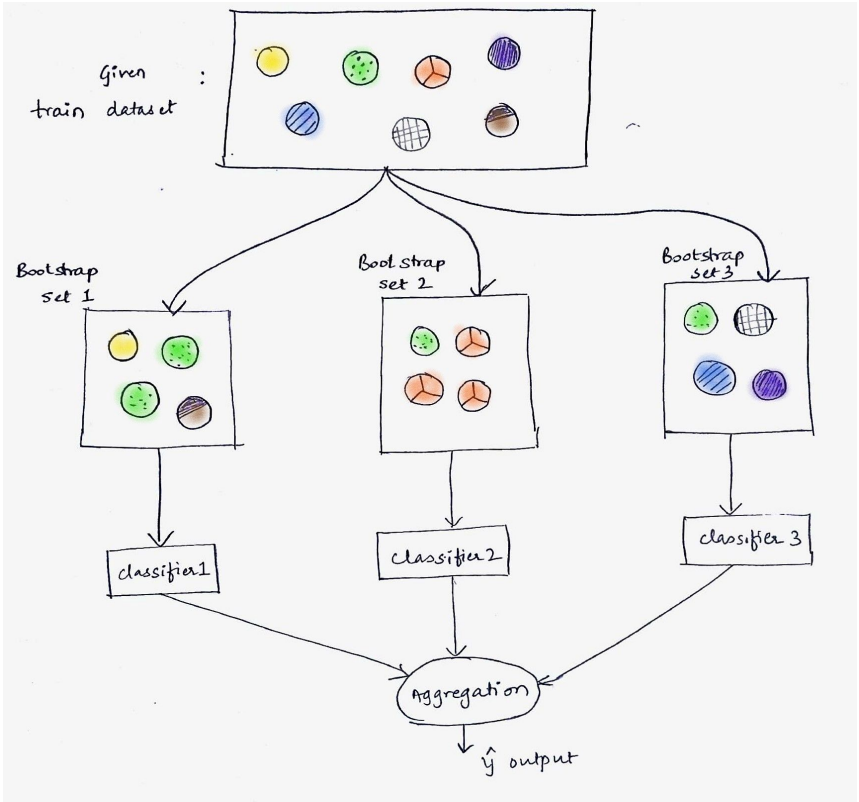
- Разделим данные на k равных частей и обучим k разных моделей.
- Каждая модель обучается на k - 1 частей, качество для одной модели оценивается на оставшейся части.
- Ансамбль строится комбинацией k моделей, каждая из которых была обучена на своих k-1 частях изначальной выборки

Bagging: bootstrap aggregation

Bootstrap - выбираем элементы из выборки случайно с повторениями.

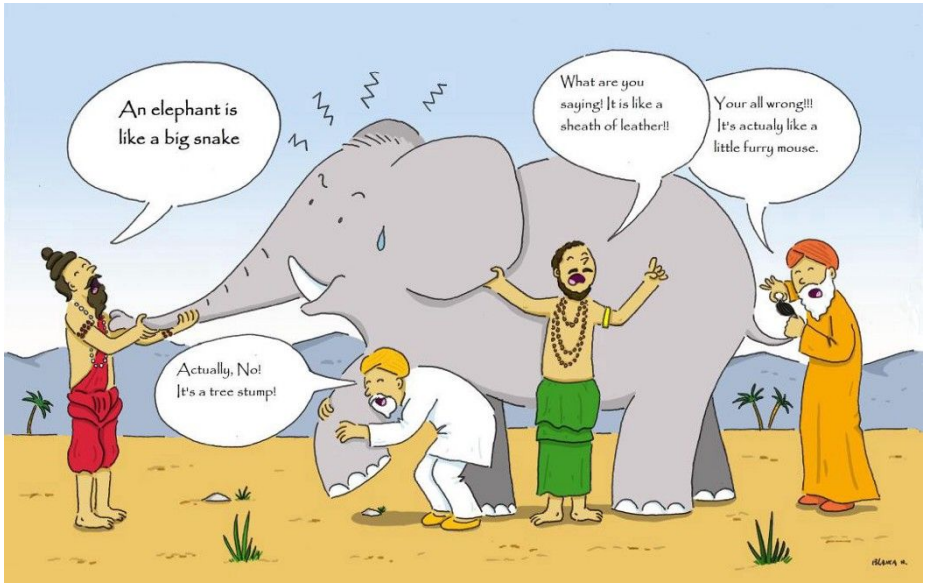
Обучаем модели на полученных bootstrap-выборках

Агрегируем результат



Комбинирующая функция

- Model Averaging Ensemble
- Weighted Average Ensemble
- Stacked Generalization (stacking) Ensemble
- Boosting Ensemble



Model Averaging Ensemble

$$b(x) = F(b_1(x), \ldots b_T(x)) = \frac{1}{T} \sum_{t=1}^{T} b_t(x)$$

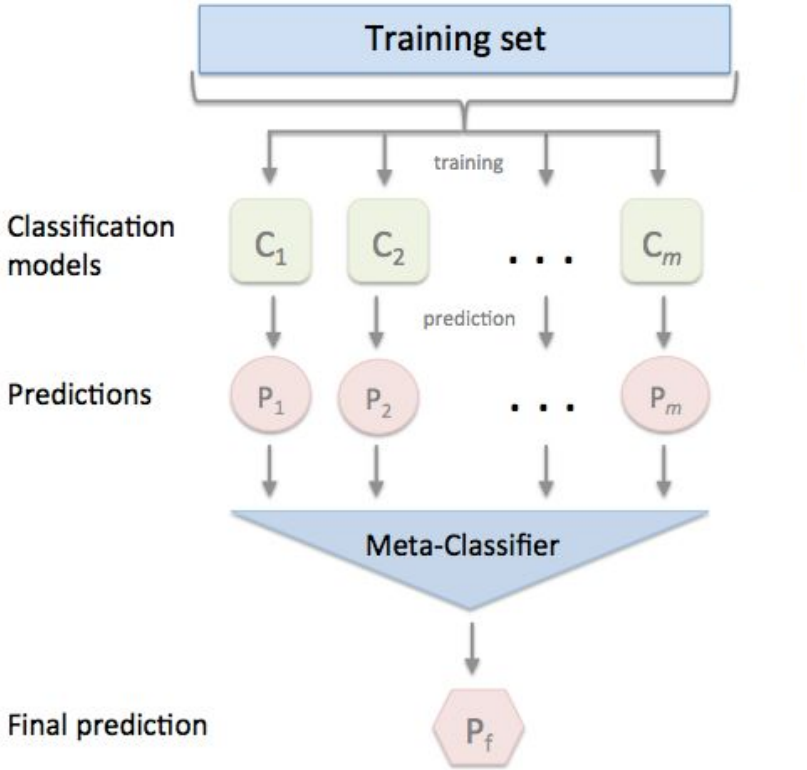
Weighted Average Ensemble

$$b(x) = F(b_1(x), \ldots, b_T(x)) = \sum_{t=1}^{T} w_t b_t(x),$$

$$\sum_{t=1}^{T} w_t = 1, \; w_t \geqslant 0;$$

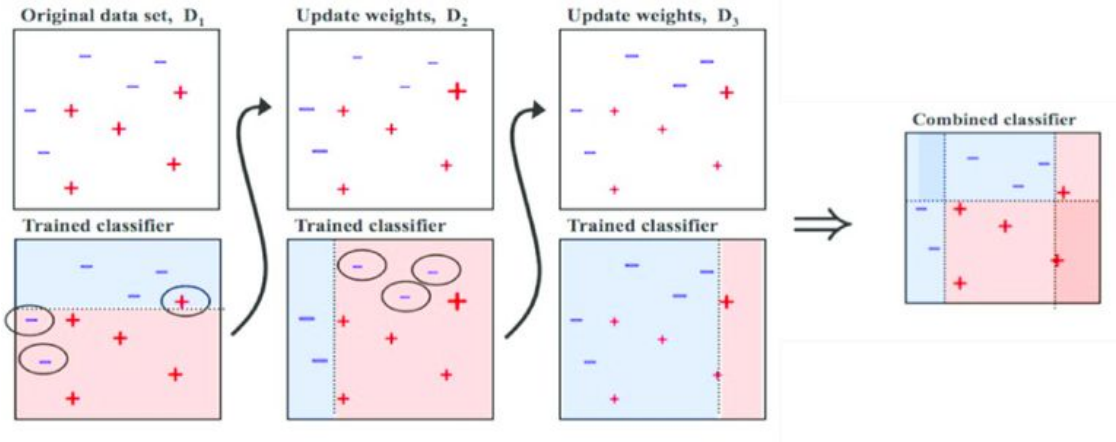
Stacked Generalization Ensemble - meta-learner

Результаты наших моделей в ансамбли подадим на вход новой модели(мета-модели), которая будет выдавать окончательный результат



Boosting Ensemble

- По очереди обучаем модели
- Каждая модель обновляет веса, выставля большие там, где ошиблась.
- Следующие модели стараются исправить ошибки предыдущих
- Собираем все полученные модели в ансамбль.



Deep Ensemble

- Обучаем несколько одинаковых нейронных сетей из разных начальных приближений на одной и той же выборке или на разных (варьирование обучающей выборки, например, Bagging)
- Комбинируем ответы одним из рассмотренных способов

Deep Ensemble дают лучше результат, чем последующие методы и чаще применяются на практике.

Очевидный недостаток: надо обучить много нейронных сетей в ансамбле

Изменение модели

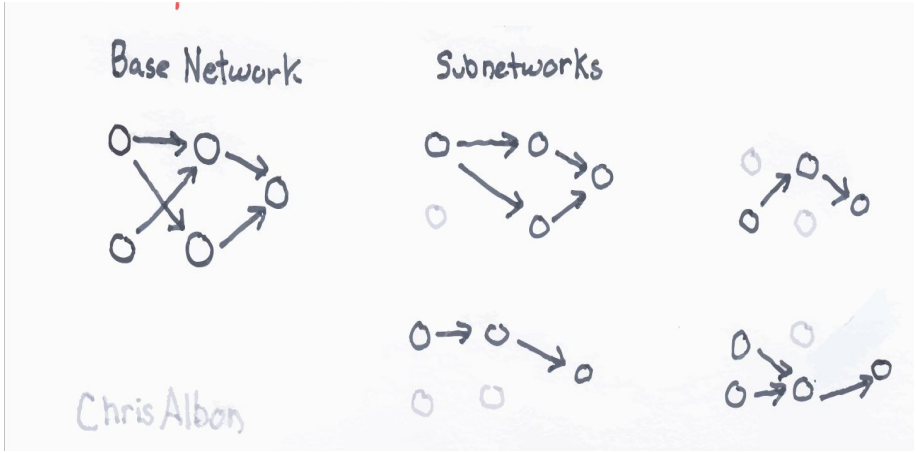
Получим из одной модели разные, а затем соберём их в ансамбль

- Dropout Ensemble
- Snapshot Ensemble
- FGE -> SWA

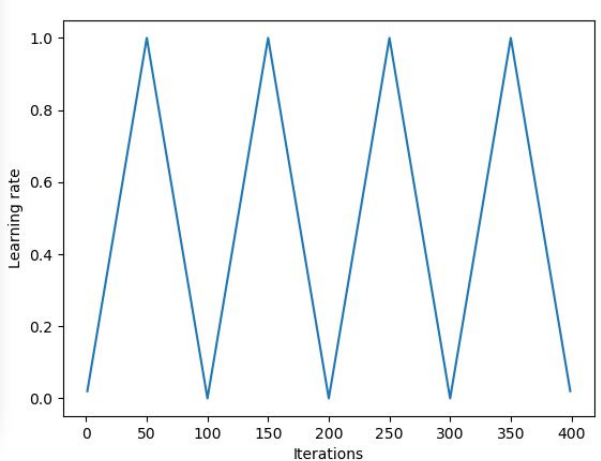
Dropout ensemble

Получим все возможные подсети нашей нейронной сети.

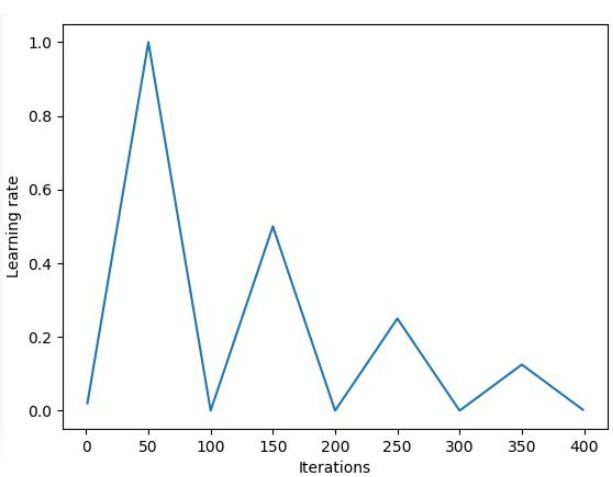
Такие подсети будут моделями в нашем ансамбле.



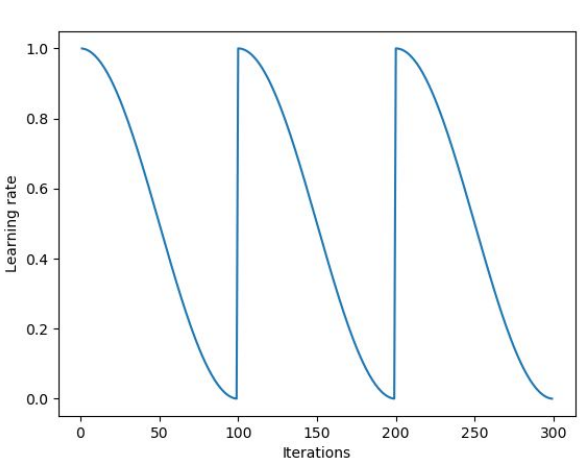
Cyclical learning rates



Triangular method



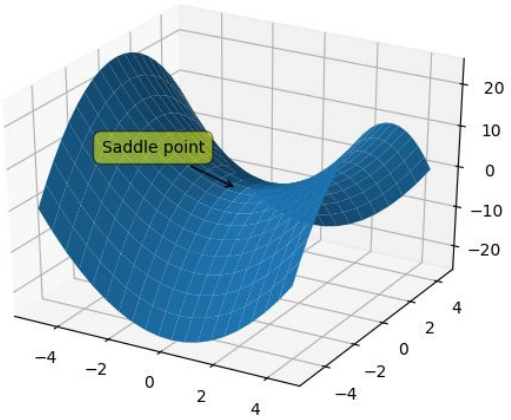
Triangular2 method

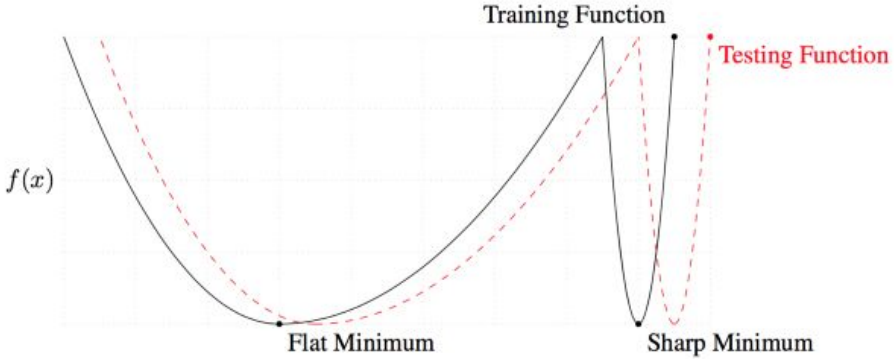


Cosine annealing

Cyclical learning rates

- Быстрее проходит седловые точки
- Позволяет находить Flat minimum

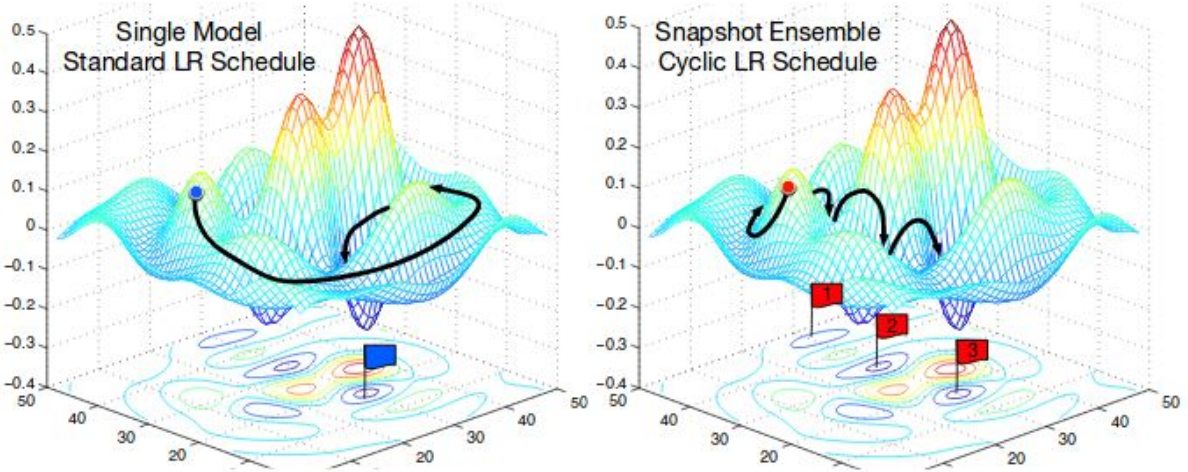




Snapshot ensembles: train 1, get M for free

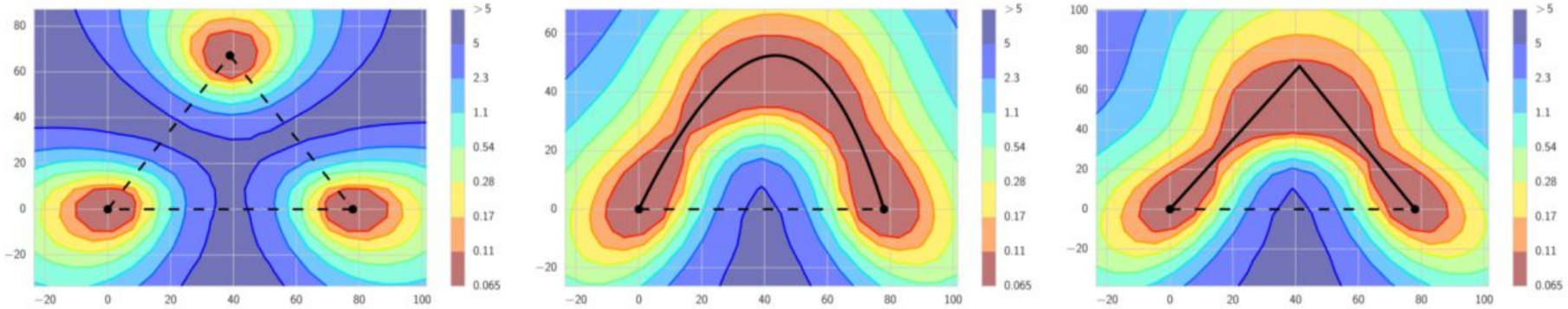
В ходе градиентного спуска с циклической длиной шага - Cosine annealing будем сохранять модели в точках локального минимума.

Из сохраненных моделей построим ансамбль.



Fast Geometric Ensembling

- Cyclical learning rates - кусочно-линейная функция
- Цикл значительно меньше, чем у Snapshot Ensemble
- Градиентный спуск проходит через путь с низкой ошибкой между разными локальными минимумами

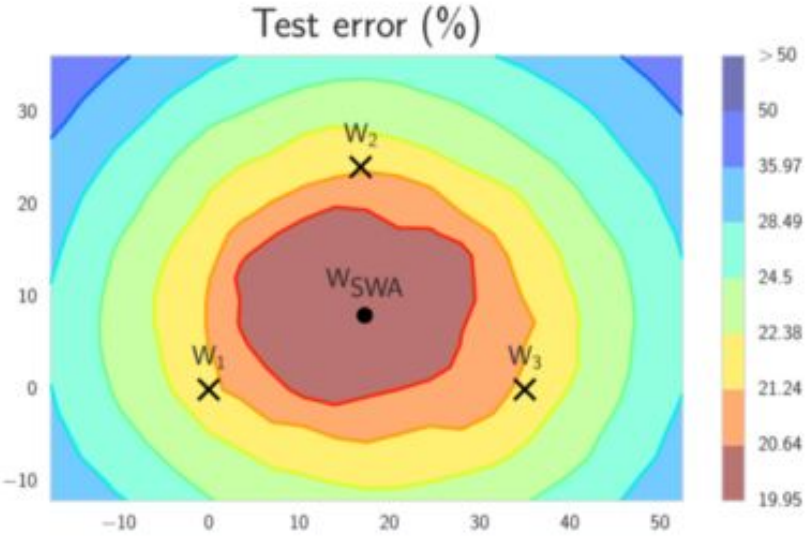


SWA: Stochastic Weight Averaging

- Эмпирическое наблюдение локальные минимумы в конце каждого цикла накапливаются на линии примерно одного уровня
- Будем обучать модели как в FGE и накапливать средние веса
- Получим одну модель с средними весами

$$w_{\text{SWA}} \leftarrow \frac{w_{\text{SWA}} \cdot n_{\text{models}} + w}{n_{\text{models}} + 1},$$

Интуиция:



Итог

- Ансмаблирование помогает бороться с переобучением нейронных сетей == бороться с высокой дисперсией
- В ансамблях регулируем: обучающая выборка, модели, комбинирующую функцию
- Cyclical learning rates - в принципе крутая идея
- С помощью этой идеи получаем алгоритмы ансамблирования: Snapshot ensemble, Fast Geometric Ensembling
- SWA - логическое продолжение Fast Geometric Ensembling(не ансамбль, но знать полезно)

Источники

Bias-Variance trade-off [1] Bias-Variance trade-off [2]

Что можно регулировать?

Snapshot Ensemble

Cyclical learning rates

FGE + SWA

FGE

SWA