Deep Ensembles

Куйбида Всеволод БПМИ212

Содержание

Введение

Введение

Глубинные нейронные сети - **мощный современный** инструмент, **минусы** -

1. Проблемы с обучением
2. Высокая дисперсия оценок
3. Нет гарантий, что мы сойдемся, куда ходим

Решение проблем (частичное) - **ансамблинг**

Что такое ансамбли?

Что такое ансамбли?

Ансамблинг - метод машинного обучения, когда множество моделей объединяются для предсказывания какого-то одного таргета

Очевидный пример:

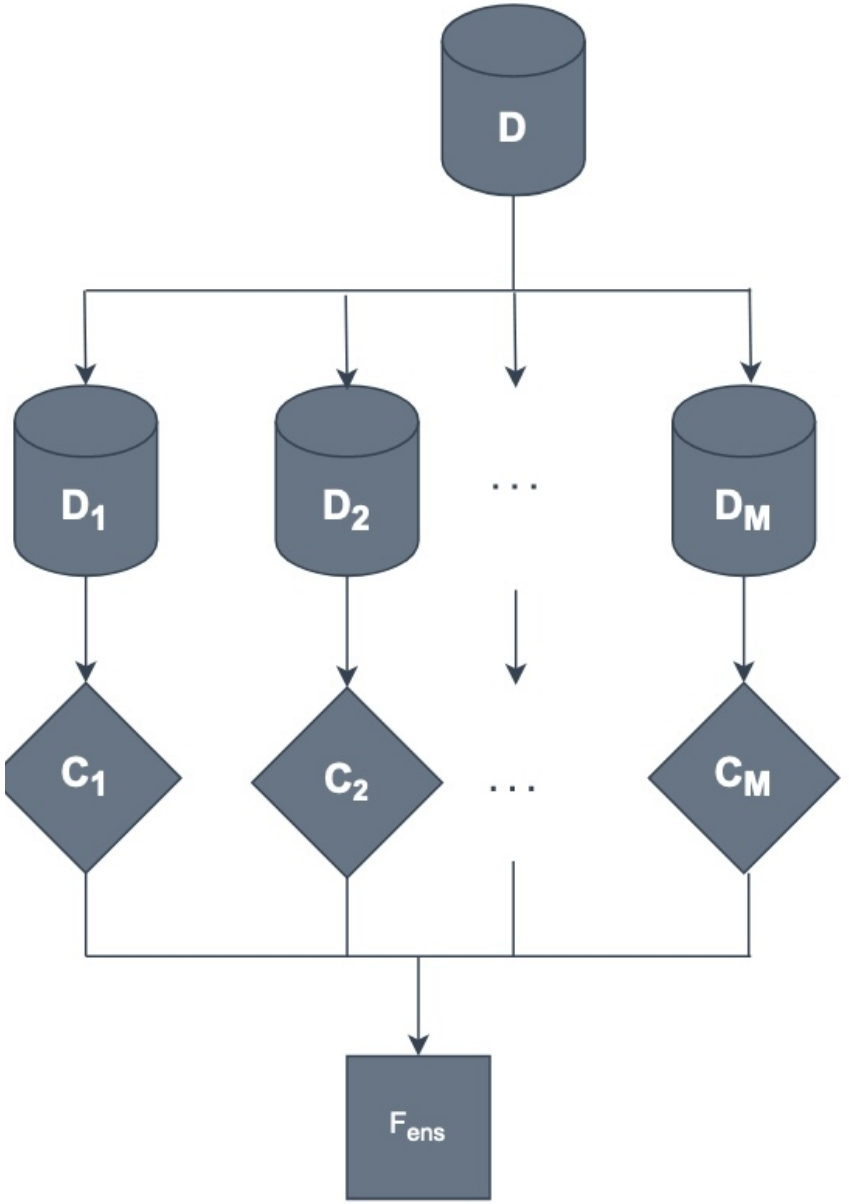
1. Обучим несколько моделей
2. В качестве ответа отдадим среднее арифметическое (или то, за что больше моделей "проголосуют")

Ансамбли с точки зрения статистики, примеры и виды

Известно, что ансамблирование **уменьшает** дисперсию/смещение в bias-variance decomposition или в их аналогах

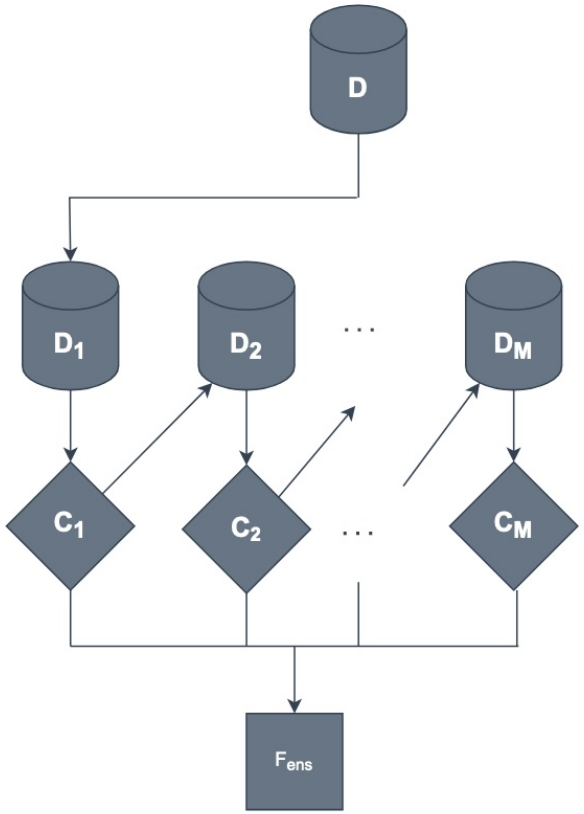
Традиционные виды ансамблей:

1. **Бэггинг** - уменьшает итоговую дисперсию, более робустный, чем модели внутри, используется в методе случайного леса



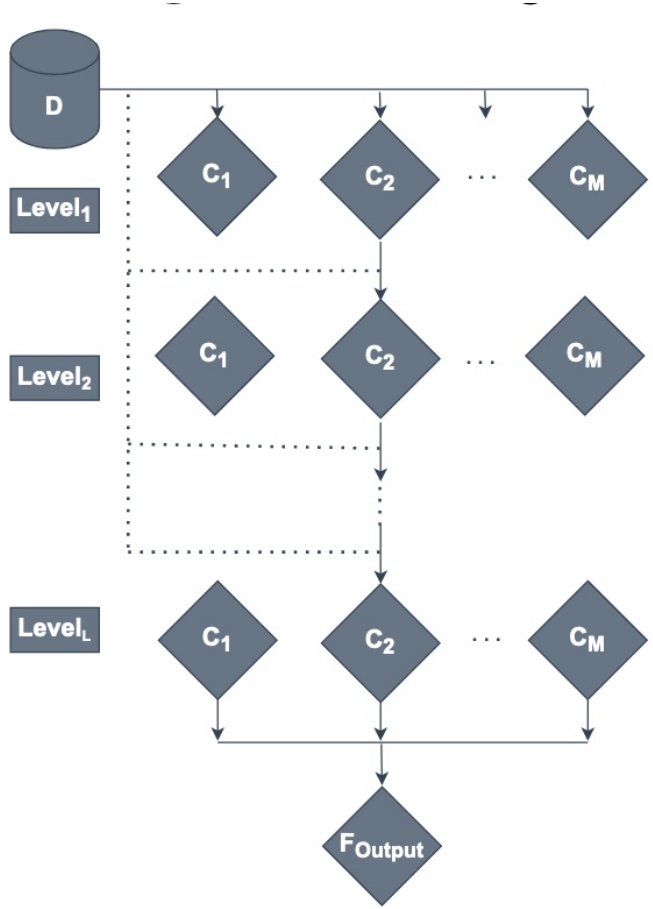
Традиционные виды ансамблей:

2. **Бустинг** - следующая модель улучшает предыдущую, используется в XGBoost, CatBoost, ...



Традиционные виды ансамблей:

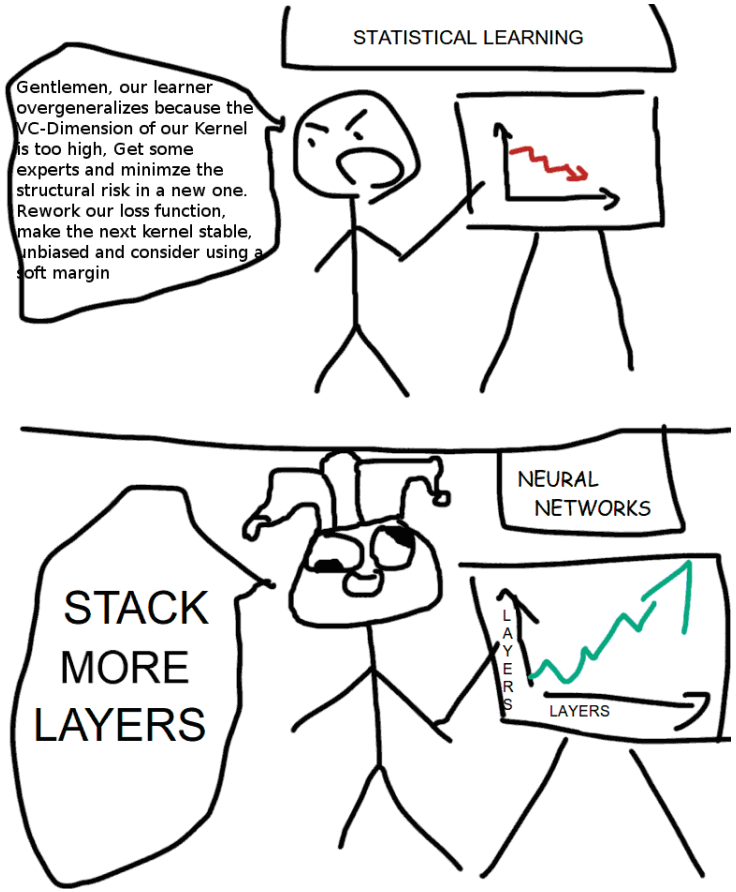
3. **Стэкинг** - ансамблирование разнообразных и так (чаще всего) сильных моделей, любимчик чемпионов Kaggle



Зачем нужны ансамбли?

Зачем нужны ансамбли?

Можем настакать бесконечно много слоёв и радоваться жизни



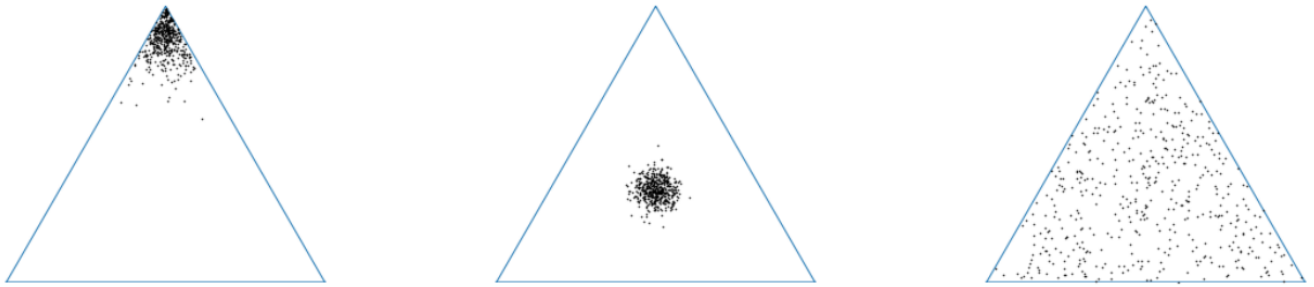
Зачем нужны ансамбли?

Введём понятие **неопределенности**:

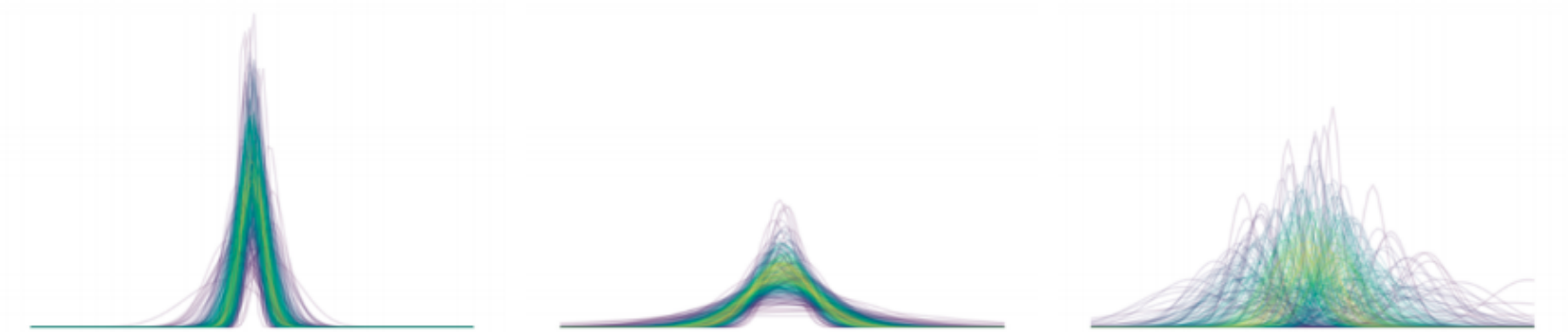
1. **Неопределенность данных** (алеаторическая) - присутствие шума в данных
2. **Неопределенность модели** (эпистемическая) - невозможность модели восстановить истинный ответ

Визуализации неопределённостей

Ensemble $\{\mathrm{P}(y|\boldsymbol{x}^*, \boldsymbol{\theta}^{(m)})\}_{m=1}^{M}$ can be visualized on a simplex



(a) Confident (b) Data Uncertainty (c) Knowledge Uncertainty



(a) Low uncertainty (b) Data uncertainty (c) Knowledge Uncertainty

Зачем нужны ансамбли?

1. **Уменьшаем** эпистемическую неопределенность
2. Можем **параллельно** тренировать относительно неглубокие модели

Как тупо делать ансамбли нейросетей? Deep Ensembles

Как тупо делать ансамбли нейросетей? Deep Ensembles

Что делаем?

1. Тренируем параллельно M нейросетей на всем наборе данных
2. Для классификации считаем вероятность принадлежности к каждому из классов, для регрессии – мат. ожидание и дисперсию
3. Итоговый результат - среднее арифметическое ответов нейросетей

Как тупо делать ансамбли нейросетей? Deep Ensembles

Плюсы:

1. очень хорошее значение точности
2. лучшее отношение размера ансамбля к качеству в данный момент

Минусы - очень дорого учить

Deep Ensembles как практический инструмент

Применяют:

1. в соревновательном Data Science
2. в медицине
3. в антифроде
4. в компьютерном зрении
5. в работе с речью
6. ...

Deep Ensembles как теоретический инструмент: Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles

Рассмотрим статью Lakshminarayanan et al. 2016 (ссылка):

Проблема: хотим измерять неуверенность модели, SOTA подходы предполагают использование байесовских нейронных сетей со своими сложностями

Решение: брать ансамбли нейронных сетей, которые дают сравнимые результаты с меньшими издержками

Итоги: подход даёт сравнимые результаты, при этом обучение Deep Ensembles проще, требуется меньше гиперпараметров, и сам процесс обучения в разы проще реализовывать

Как умно делать ансамбли нейросетей?

Как умно делать ансамбли нейросетей?

Рассмотрим два способа делать ансамбли **умнее**, чем Deep Ensembles:

1. **Snapshot Ensembles**
2. **Dropout Ensembles**

Snapshot ensembles

Что делаем?

1. Обучаем одну нейронную сеть циклами обучения
2. В конце цикла обучения, когда мы “приходим” в локальный минимум, мы делаем “снэпшот” модели - сохраняем в другую модель веса
3. Повторяем шаги 1-2 до тех пор, пока не получим M моделей
4. Наш ответ на тесте - усреднённый ответ M моделей

Snapshot ensembles

Плюсы:

1. сильное уменьшение затрат на обучение ансамбля,
2. улучшение точности предсказания,
3. одно из лучших отношений качества к размеру модели

Минусы:

1. часто проигрывают Deep Ensembles в точности

Snapshot ensembles - визуализации

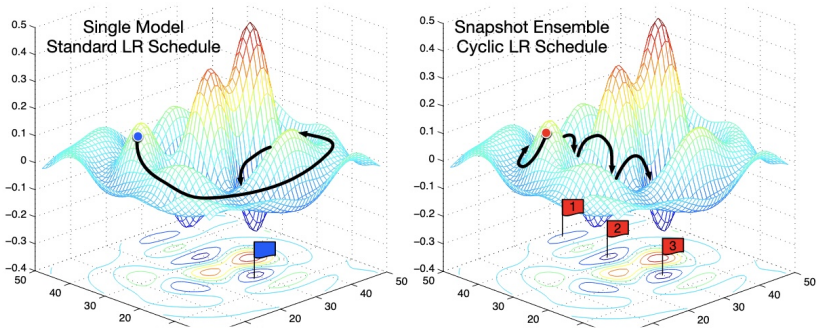


Figure 1: **Left:** Illustration of SGD optimization with a typical learning rate schedule. The model converges to a minimum at the end of training. **Right:** Illustration of Snapshot Ensembling. The model undergoes several learning rate annealing cycles, converging to and escaping from multiple local minima. We take a snapshot at each minimum for test-time ensembling.

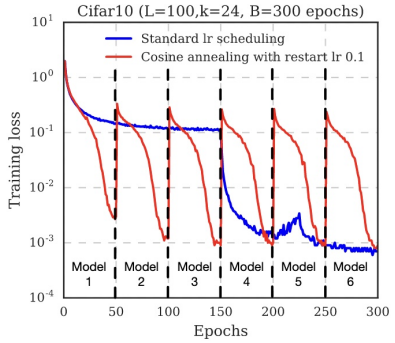


Figure 2: Training loss of 100-layer DenseNet on CIFAR10 using standard learning rate (blue) and $M = 6$ cosine annealing cycles (red). The intermediate models, denoted by the dotted lines, form an ensemble at the end of training.

Dropout ensembles

Что делаем (в самом простом случае - **нейросети**)?

1. Фиксируем p - вероятность того, что каждый нейрон будет “выключен” при обучении
2. При обучении на каждом шаге с вероятностью p “выключаем” нейрон (то есть не тренируем), при этом умножаем выход модели на 1 / (1 - p) для того, чтобы матожидание результата не поменялось
3. В “тестовом” режиме не выкидываем нейроны

Метод можно рассматривать как **простой и очень общий способ** регуляризации + **легко обобщается**

Dropout ensembles

Плюсы:

1. Метод предоставляет хороший, простой и очень общий способ регуляризации
2. Скорость предсказаний хороша
3. Легко и хорошо обобщается

Минусы:

1. требуется больше времени на обучение
2. увеличение размера ансамбля дает меньший прирост качества, нежели в других случаях

Насколько большими должны быть ансамбли?

Насколько большими должны быть ансамбли?

1. Универсальной верной формулы **не существует**
2. Размер зависит от **метода ансамблирования, используемых алгоритмов** и т. д.
3. В статье Bonab et al. 2018 (ссылка) рассказывается об **оптимальном** кол-ве моделей в ансамбле - авторы предлагают брать **число, схожее с кол-вом классов** в мультиклассовой классификации

Статистика, графики

Статистика, графики - Deep Ensemble Equivalent

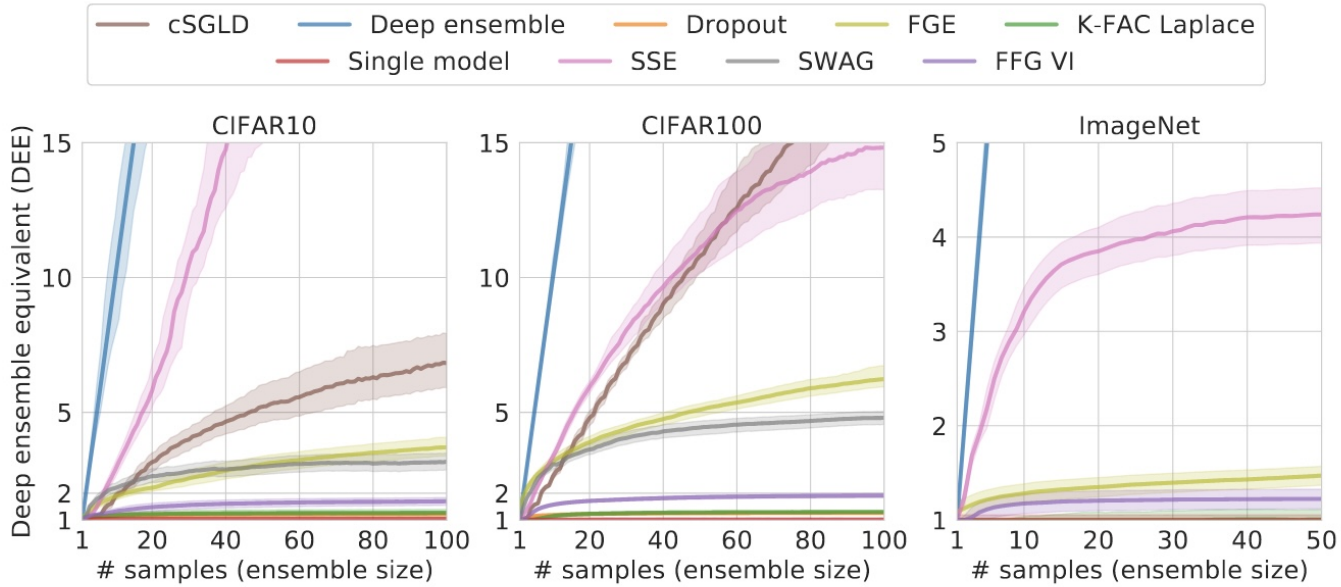
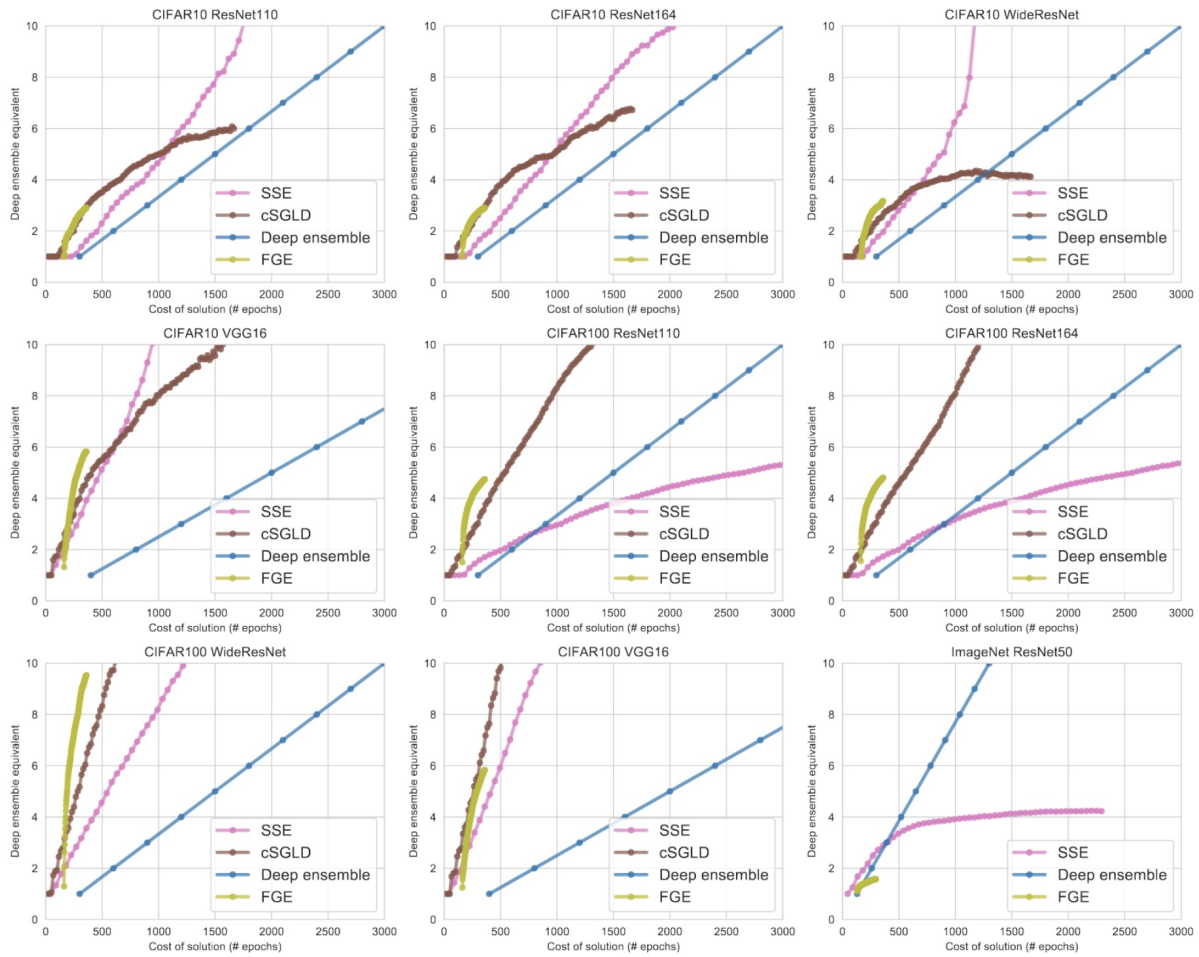


Figure 3: The deep ensemble equivalent score (DEE) for different numbers of samples on CIFAR-10, CIFAR-100, and ImageNet datasets averaged across different deep convolutional architectures. A deep ensemble equivalent score (DEE) of a model is equal to the minimum size of a deep ensemble (an ensemble of independently train networks) that achieves the same performance as the model under consideration. The score is measured in the number of models (higher is better). The area between average lower and upper bounds of DEE is shaded. **The plot demonstrates that all of the ensembling techniques are far less efficient than deep ensembles during inference and fail to produce the same level of performance as deep ensembles.** The comparison that is normalized on training time is presented in Appendix A.

Статистика, графики - Performance in DEE



Статистика, графики - сравнительные результаты моделей

Model	Method	Error (%)				Negative calibrated log-likelihood			
		1	5	10	100	1	5	10	100
VGG16 CIFAR-10	Dropout	$5.86_{\pm 0.09}$	$5.81_{\pm 0.08}$	$5.82_{\pm 0.06}$	$5.79_{\pm 0.07}$	$0.232_{\pm 0.005}$	$0.225_{\pm 0.004}$	$0.224_{\pm 0.004}$	$0.223_{\pm 0.003}$
	SWA-Gaussian	$7.03_{\pm 0.50}$	$5.66_{\pm 0.08}$	$5.49_{\pm 0.12}$	$5.25_{\pm 0.13}$	$0.230_{\pm 0.014}$	$0.182_{\pm 0.003}$	$0.171_{\pm 0.002}$	$0.160_{\pm 0.002}$
	Cyclic SGLD	$7.37_{\pm 0.16}$	$6.56_{\pm 0.09}$	$5.71_{\pm 0.06}$	$4.84_{\pm 0.04}$	$0.234_{\pm 0.004}$	$0.196_{\pm 0.004}$	$0.176_{\pm 0.003}$	$0.147_{\pm 0.003}$
	Fast Geometric Ens.	$6.52_{\pm 0.16}$	$5.95_{\pm 0.16}$	$5.69_{\pm 0.16}$	$5.10_{\pm 0.13}$	$0.213_{\pm 0.005}$	$0.187_{\pm 0.003}$	$0.178_{\pm 0.003}$	$0.155_{\pm 0.004}$
	Deep Ensembles	$5.95_{\pm 0.14}$	$4.79_{\pm 0.11}$	$4.57_{\pm 0.07}$	$4.39_{\pm \text{NA}}$	$0.226_{\pm 0.001}$	$0.158_{\pm 0.002}$	$0.148_{\pm 0.001}$	$0.134_{\pm \text{NA}}$
	Single model	$5.83_{\pm 0.11}$	$5.83_{\pm 0.11}$	$5.83_{\pm 0.11}$	$5.83_{\pm 0.11}$	$0.223_{\pm 0.002}$	$0.223_{\pm 0.002}$	$0.223_{\pm 0.002}$	$0.223_{\pm 0.002}$
	Variational Inf. (FFG)	$6.57_{\pm 0.09}$	$5.63_{\pm 0.13}$	$5.50_{\pm 0.10}$	$5.46_{\pm 0.03}$	$0.239_{\pm 0.002}$	$0.192_{\pm 0.002}$	$0.184_{\pm 0.002}$	$0.175_{\pm 0.001}$
	KFAC-Laplace	$6.00_{\pm 0.13}$	$5.82_{\pm 0.12}$	$5.82_{\pm 0.19}$	$5.80_{\pm 0.19}$	$0.210_{\pm 0.005}$	$0.203_{\pm 0.007}$	$0.201_{\pm 0.007}$	$0.200_{\pm 0.008}$
	Snapshot Ensembles	$7.76_{\pm 0.22}$	$5.52_{\pm 0.13}$	$5.00_{\pm 0.10}$	$4.54_{\pm 0.05}$	$0.247_{\pm 0.005}$	$0.176_{\pm 0.001}$	$0.160_{\pm 0.001}$	$0.137_{\pm 0.001}$
ResNet110 CIFAR-10	SWA-Gaussian	$5.77_{\pm 0.45}$	$4.56_{\pm 0.17}$	$4.46_{\pm 0.12}$	$4.34_{\pm 0.13}$	$0.178_{\pm 0.009}$	$0.143_{\pm 0.004}$	$0.139_{\pm 0.003}$	$0.131_{\pm 0.003}$
	Cyclic SGLD	$6.18_{\pm 0.20}$	$5.32_{\pm 0.15}$	$4.55_{\pm 0.13}$	$3.83_{\pm 0.02}$	$0.185_{\pm 0.006}$	$0.156_{\pm 0.005}$	$0.138_{\pm 0.002}$	$0.115_{\pm 0.001}$
	Fast Geometric Ens.	$5.52_{\pm 0.09}$	$4.83_{\pm 0.08}$	$4.73_{\pm 0.10}$	$4.28_{\pm 0.05}$	$0.163_{\pm 0.002}$	$0.141_{\pm 0.003}$	$0.137_{\pm 0.003}$	$0.126_{\pm 0.002}$
	Deep Ensembles	$4.66_{\pm 0.11}$	$3.77_{\pm 0.11}$	$3.63_{\pm 0.07}$	$3.53_{\pm \text{NA}}$	$0.148_{\pm 0.004}$	$0.117_{\pm 0.002}$	$0.112_{\pm 0.002}$	$0.106_{\pm \text{NA}}$
	Single model	$4.69_{\pm 0.11}$	$4.69_{\pm 0.11}$	$4.69_{\pm 0.11}$	$4.69_{\pm 0.11}$	$0.150_{\pm 0.002}$	$0.150_{\pm 0.002}$	$0.150_{\pm 0.003}$	$0.150_{\pm 0.002}$
	Variational Inf. (FFG)	$5.57_{\pm 0.26}$	$4.91_{\pm 0.15}$	$4.72_{\pm 0.13}$	$4.60_{\pm 0.03}$	$0.178_{\pm 0.003}$	$0.149_{\pm 0.001}$	$0.144_{\pm 0.001}$	$0.140_{\pm 0.000}$
	KFAC-Laplace	$5.81_{\pm 0.39}$	$5.14_{\pm 0.15}$	$4.90_{\pm 0.14}$	$4.78_{\pm 0.08}$	$0.187_{\pm 0.014}$	$0.160_{\pm 0.007}$	$0.153_{\pm 0.005}$	$0.147_{\pm 0.003}$
	Snapshot Ensembles	$8.41_{\pm 0.27}$	$4.85_{\pm 0.11}$	$4.16_{\pm 0.16}$	$3.52_{\pm 0.10}$	$0.252_{\pm 0.006}$	$0.153_{\pm 0.002}$	$0.132_{\pm 0.002}$	$0.107_{\pm 0.001}$
ResNet164 CIFAR-10	SWA-Gaussian	$5.41_{\pm 0.71}$	$4.21_{\pm 0.19}$	$4.21_{\pm 0.23}$	$4.02_{\pm 0.14}$	$0.171_{\pm 0.028}$	$0.130_{\pm 0.004}$	$0.128_{\pm 0.004}$	$0.121_{\pm 0.002}$
	Cyclic SGLD	$5.80_{\pm 0.21}$	$4.97_{\pm 0.12}$	$4.30_{\pm 0.08}$	$3.66_{\pm 0.06}$	$0.178_{\pm 0.004}$	$0.149_{\pm 0.004}$	$0.131_{\pm 0.003}$	$0.110_{\pm 0.001}$
	Fast Geometric Ens.	$5.22_{\pm 0.07}$	$4.49_{\pm 0.06}$	$4.36_{\pm 0.07}$	$4.09_{\pm 0.12}$	$0.157_{\pm 0.003}$	$0.134_{\pm 0.002}$	$0.130_{\pm 0.001}$	$0.119_{\pm 0.002}$
	Deep Ensembles	$4.53_{\pm 0.11}$	$3.51_{\pm 0.09}$	$3.50_{\pm 0.06}$	$3.34_{\pm \text{NA}}$	$0.147_{\pm 0.002}$	$0.113_{\pm 0.001}$	$0.107_{\pm 0.001}$	$0.100_{\pm \text{NA}}$
	Single model	$4.52_{\pm 0.11}$	$4.52_{\pm 0.11}$	$4.52_{\pm 0.11}$	$4.52_{\pm 0.11}$	$0.144_{\pm 0.002}$	$0.144_{\pm 0.003}$	$0.144_{\pm 0.002}$	$0.144_{\pm 0.003}$
	Variational Inf. (FFG)	$5.62_{\pm 0.14}$	$4.78_{\pm 0.05}$	$4.66_{\pm 0.05}$	$4.55_{\pm 0.08}$	$0.183_{\pm 0.004}$	$0.151_{\pm 0.001}$	$0.146_{\pm 0.001}$	$0.141_{\pm 0.001}$
	KFAC-Laplace	$5.23_{\pm 0.29}$	$4.77_{\pm 0.23}$	$4.65_{\pm 0.17}$	$4.60_{\pm 0.09}$	$0.168_{\pm 0.008}$	$0.151_{\pm 0.007}$	$0.146_{\pm 0.005}$	$0.142_{\pm 0.004}$
	Snapshot Ensembles	$8.06_{\pm 0.10}$	$4.50_{\pm 0.04}$	$3.89_{\pm 0.09}$	$3.50_{\pm 0.05}$	$0.241_{\pm 0.004}$	$0.144_{\pm 0.003}$	$0.124_{\pm 0.002}$	$0.104_{\pm 0.001}$
WideResNet CIFAR-10	Dropout	$3.88_{\pm 0.12}$	$3.70_{\pm 0.18}$	$3.63_{\pm 0.19}$	$3.64_{\pm 0.17}$	$0.130_{\pm 0.002}$	$0.120_{\pm 0.002}$	$0.119_{\pm 0.001}$	$0.117_{\pm 0.002}$
	SWA-Gaussian	$4.98_{\pm 1.17}$	$3.53_{\pm 0.09}$	$3.34_{\pm 0.14}$	$3.28_{\pm 0.10}$	$0.157_{\pm 0.036}$	$0.111_{\pm 0.004}$	$0.105_{\pm 0.003}$	$0.101_{\pm 0.002}$
	Cyclic SGLD	$4.78_{\pm 0.16}$	$4.09_{\pm 0.11}$	$3.63_{\pm 0.13}$	$3.19_{\pm 0.04}$	$0.155_{\pm 0.003}$	$0.128_{\pm 0.002}$	$0.114_{\pm 0.001}$	$0.099_{\pm 0.002}$
	Fast Geometric Ens.	$4.86_{\pm 0.17}$	$3.95_{\pm 0.07}$	$3.77_{\pm 0.10}$	$3.34_{\pm 0.06}$	$0.148_{\pm 0.003}$	$0.120_{\pm 0.002}$	$0.113_{\pm 0.002}$	$0.102_{\pm 0.001}$
	Deep Ensembles	$3.65_{\pm 0.02}$	$3.11_{\pm 0.10}$	$3.01_{\pm 0.06}$	$2.83_{\pm \text{NA}}$	$0.123_{\pm 0.002}$	$0.097_{\pm 0.001}$	$0.095_{\pm 0.001}$	$0.090_{\pm \text{NA}}$
	Single model	$3.70_{\pm 0.15}$	$3.70_{\pm 0.15}$	$3.70_{\pm 0.15}$	$3.70_{\pm 0.15}$	$0.124_{\pm 0.005}$	$0.124_{\pm 0.005}$	$0.125_{\pm 0.005}$	$0.124_{\pm 0.005}$
	Variational Inf. (FFG)	$5.61_{\pm 0.04}$	$4.15_{\pm 0.15}$	$3.94_{\pm 0.10}$	$3.64_{\pm 0.07}$	$0.189_{\pm 0.002}$	$0.134_{\pm 0.002}$	$0.127_{\pm 0.002}$	$0.117_{\pm 0.001}$
	KFAC-Laplace	$4.03_{\pm 0.19}$	$3.90_{\pm 0.15}$	$3.88_{\pm 0.22}$	$3.83_{\pm 0.16}$	$0.134_{\pm 0.004}$	$0.124_{\pm 0.004}$	$0.122_{\pm 0.005}$	$0.120_{\pm 0.003}$
	Snapshot Ensembles	$5.56_{\pm 0.15}$	$3.68_{\pm 0.09}$	$3.33_{\pm 0.10}$	$2.89_{\pm 0.07}$	$0.179_{\pm 0.005}$	$0.119_{\pm 0.001}$	$0.105_{\pm 0.001}$	$0.090_{\pm 0.001}$

Заключение

Заключение

- Ансамблирование нейросетей - **хороший инструмент**, который **активно развивается**
- За свои плюсы часто приходится платить **дороговизной** обучения
- Имеет как **практические** и **инженерные** приложения, так и более **теоретические** (uncertainty estimation)