Ансамблирование моделей

Московский Физико-Технический Институт

2022

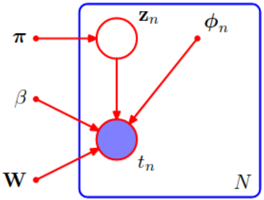
Ансамбли моделей

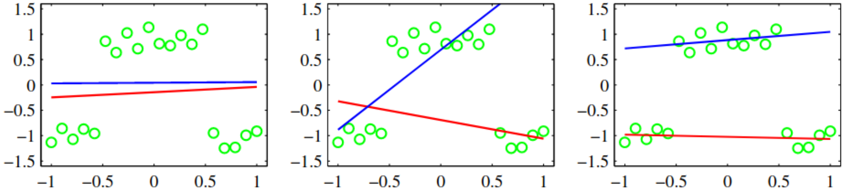
Определение (Wiki)

использование несколько обучающих алгоритмов с целью получения лучшей эффективности прогнозирования, чем могли бы получить от каждого обучающего алгоритма по отдельности. Ансамбль методов в обучении машин состоит из конкретного конечного множества альтернативных моделей, но, обычно, позволяет существовать существенно более гибким структурам.

Смесь моделей

$$f = \sum \gamma_i f_i(x)$$





Смесь моделей vs. байесовское усреднение

Смесь:

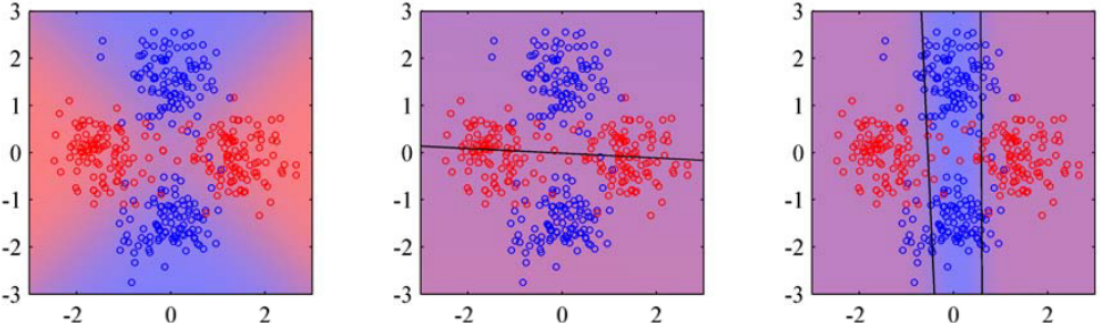
$$\mathsf{f} = \sum \gamma_i \mathsf{f}_i(\mathsf{x})$$

Усреднение:

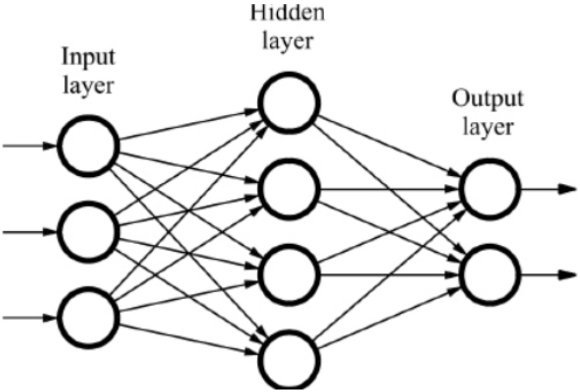
$$\mathsf{f} = \sum p(\mathsf{f}_i)\mathsf{f}_i(\mathsf{x}).$$

Смесь экспертов

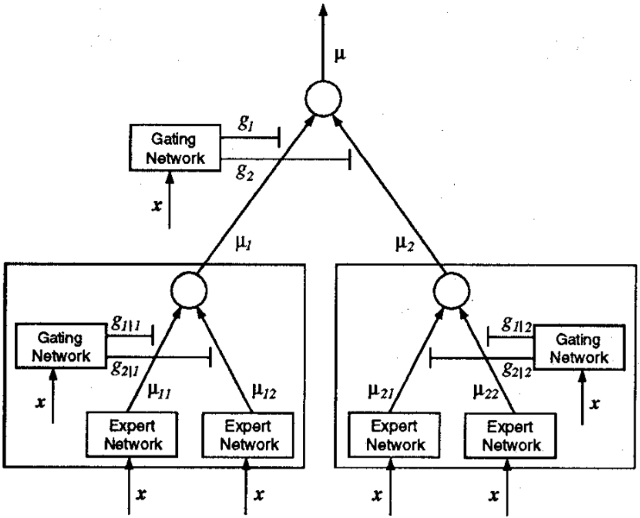
$$f = \sum \gamma_i(x) f_i(x)$$



Нейросеть как смесь



Иерархия на смесях экспертов



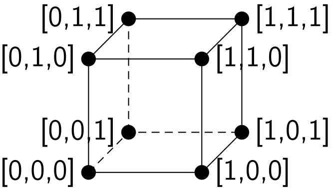
Многомерные модели (мультимодели)

$$\mathsf{f} = \sum \gamma_i(\mathsf{x})\mathsf{f}_i(\mathsf{x}),$$

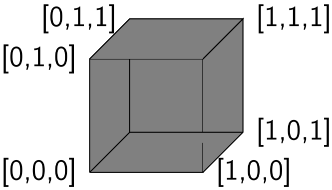
$$\sum \boldsymbol{\gamma} = 1, \quad \boldsymbol{\gamma}_i \in 0, 1.$$

Structure restrictions

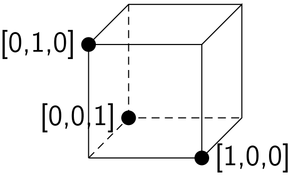
An example of restrictions for structure parameter γ, $|\gamma| = 3$.



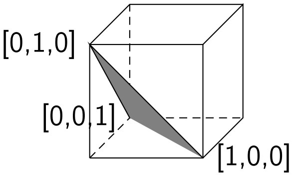
Cube vertices



Cube interior



Simplex vertices

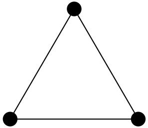


Simplex interior

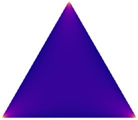
Prior distribution for the model structure

Every point in a simplex defines a model.

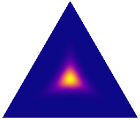
Gumbel-Softmax distribution: $\boldsymbol{\Gamma} \sim \mathrm{GS}(\mathbf{s}, \lambda_{\text{temp}})$



$\lambda_{\text{temp}} \to 0$

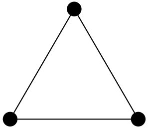


$\lambda_{\text{temp}} = 0.995$

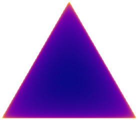


$\lambda_{\text{temp}} = 5.0$

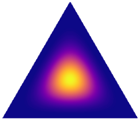
Dirichlet distribution: $\boldsymbol{\Gamma} \sim \mathrm{Dir}(\mathbf{s}, \lambda_{\text{temp}})$



$\lambda_{\text{temp}} \to 0$

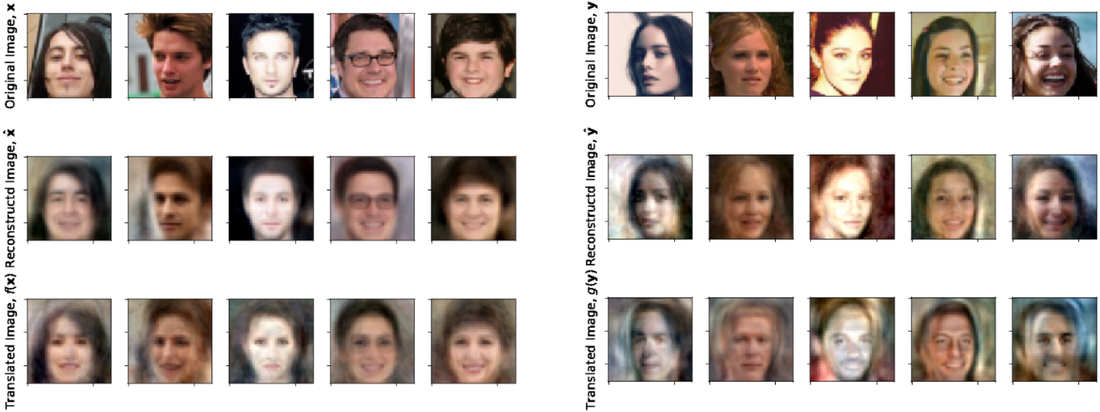


$\lambda_{\text{temp}} = 0.995$

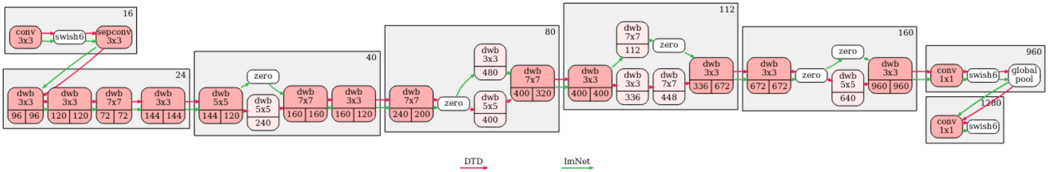


$\lambda_{\text{temp}} = 5.0$

Мультидоменность



Мультидоменность



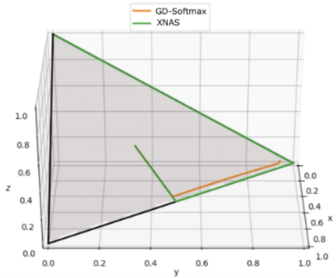
XNAS

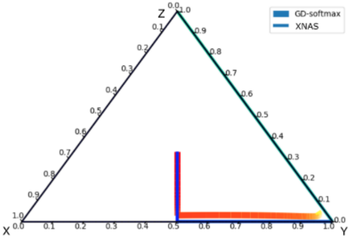
Algorithm 1 XNAS for a single forecaster

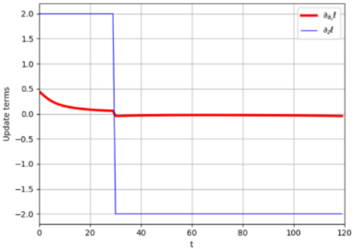
```
Input: The learning rate η,
  Loss-gradient bound 𝓛 ,
  Experts predictions {f_{t,i}}_{i=1}^{N}  ∀t = 1, ..., T
Init: I_0 = {1, ..., N},  v_{0,i} ← 1,  ∀i ∈ I_0
for rounds t = 1, ..., T do
    Update ω by descending ∇_ω ℓ_train(ω, v)
    p_t ← (Σ_{i∈I_{t-1}} v_{t-1,i} · f_{t-1,i}) / (Σ_{i∈I_{t-1}} v_{t-1,i})    #Predict
    {loss gradient revealed: ∇_{p_t} ℓ_val(p_t)}
    for i ∈ I_{t-1} do
        R_{t,i} = −∇_{p_t} ℓ_val(p_t) · f_{t,i}    #Rewards
        v_{t,i} ← v_{t-1,i} · exp{η R_{t,i}}    #EG step
    end for
    θ_t ← max_{i∈I_{t-1}} {v_{t,i}} · exp{−2η𝓛(T − t)}
    I_t ← I_{t-1} \ {i | v_{t,i} < θ_t}    #Wipeout
end for
```

ImageNet Architecture	Test error	Params (M)	Search cost
SNAS [50]	27.3	4.3	1.5
ASAP [29]	26.7	5.1	0.2
DARTS [25]	26.7	4.9	1
NASNet-A [56]	26.0	5.3	1800
PNAS [24]	25.8	5.1	150
Amoeba-A [33]	25.5	5.1	3150
RandWire [48]	25.3	5.6	0
SharpDarts [17]	25.1	4.9	0.8
Amoeba-C [33]	24.3	6.4	3150
XNAS	**24.0**	5.2	0.3

XNAS







Ансамблирование для оценки неопределенности

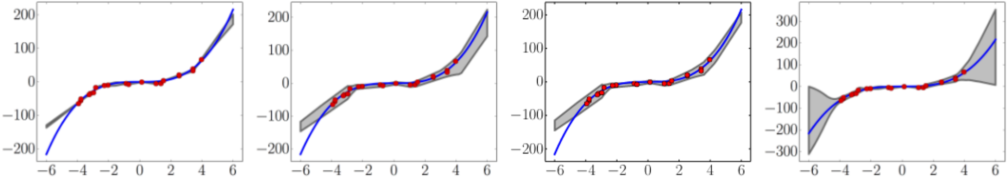


Figure 1: Results on a toy regression task: x-axis denotes x. On the y-axis, the blue line is the *ground truth* curve, the red dots are observed noisy training data points and the gray lines correspond to the predicted mean along with three standard deviations. Left most plot corresponds to empirical variance of 5 networks trained using MSE, second plot shows the effect of training using NLL using a single net, third plot shows the additional effect of adversarial training, and final plot shows the effect of using an ensemble of 5 networks respectively.

Neural ensebmle search

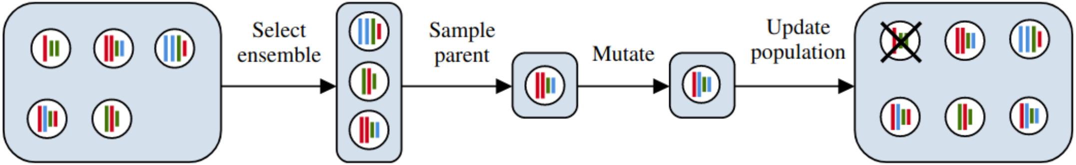


Figure 3: Illustration of one iteration of NES-RE. Network architectures are represented as colored bars of different lengths illustrating different layers and widths. Starting with the current population, ensemble selection is applied to select parent candidates, among which one is sampled as the parent. A mutated copy of the parent is added to the population, and the oldest member is removed.

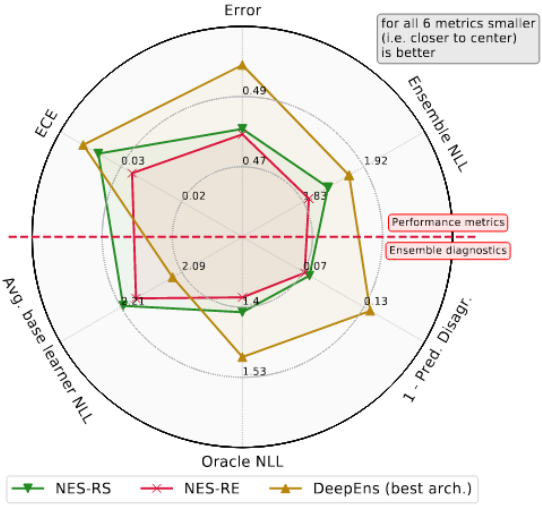
Neural ensebmle search

Algorithm 1: NES with Regularized Evolution

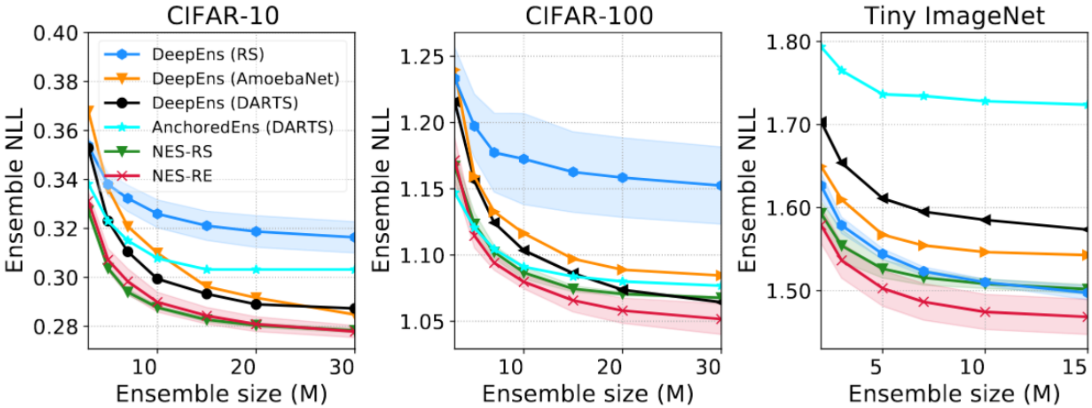
Data: Search space $\mathcal{A}$; ensemble size M; comp. budget K; $\mathcal{D}_{\text{train}}$, $\mathcal{D}_{\text{val}}$; population size P; number of parent candidates m.

Sample P architectures $\alpha_1, \dots, \alpha_P$ independently and uniformly from $\mathcal{A}$.
Train each architecture α_i using $\mathcal{D}_{\text{train}}$, and initialize $\mathfrak{p} = \mathcal{P} = \{f_{\theta_1,\alpha_1}, \dots, f_{\theta_P,\alpha_P}\}$.
while $|\mathcal{P}| < K$ **do**
    Select m parent candidates $\{f_{\widetilde{\theta}_1,\widetilde{\alpha}_1}, \dots, f_{\widetilde{\theta}_m,\widetilde{\alpha}_m}\} = \texttt{ForwardSelect}(\mathfrak{p}, \mathcal{D}_{\text{val}}, m)$.
    Sample uniformly a parent architecture α from $\{\widetilde{\alpha}_1, \dots, \widetilde{\alpha}_m\}$. `// ` α `stays in` $\mathfrak{p}$.
    Apply mutation to α, yielding child architecture β.
    Train β using $\mathcal{D}_{\text{train}}$ and add the trained network $f_{\theta,\beta}$ to $\mathfrak{p}$ and $\mathcal{P}$.
    Remove the oldest member in $\mathfrak{p}$. `// as done in RE [49].`
Select base learners $\{f_{\theta_1^*,\alpha_1^*}, \dots, f_{\theta_M^*,\alpha_M^*}\} = \texttt{ForwardSelect}(\mathcal{P}, \mathcal{D}_{\text{val}}, M)$ by forward step-wise selection without replacement.
return ensemble $\texttt{Ensemble}(f_{\theta_1^*,\alpha_1^*}, \dots, f_{\theta_M^*,\alpha_M^*})$

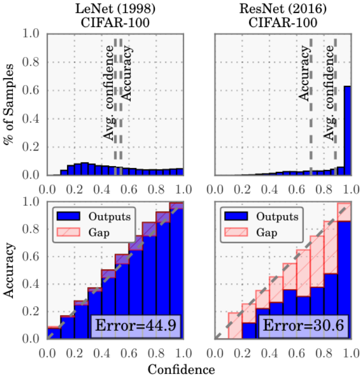
Neural ensebmle search



Neural ensebmle search



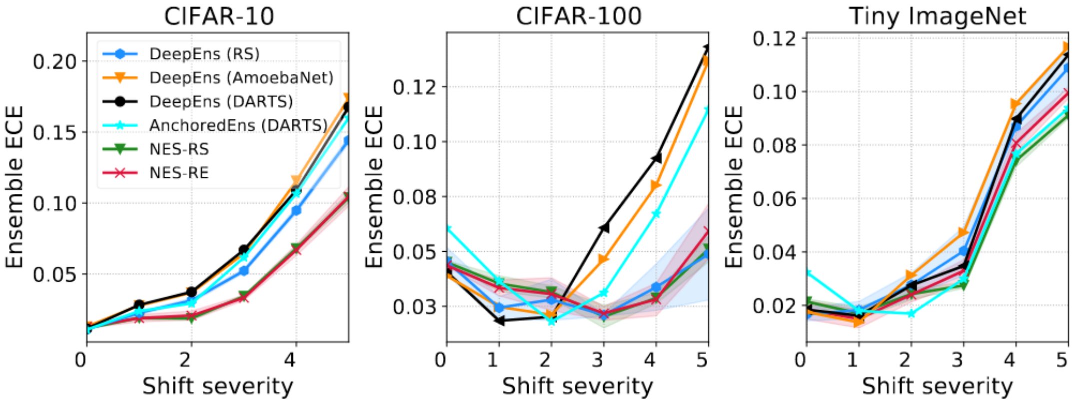
Neural ensebmle search



$$\mathop{\mathbb{E}}_{\hat{P}}\left[\left|\mathbb{P}\left(\hat{Y}=Y \mid \hat{P}=p\right)-p\right|\right]$$

$$\mathrm{ECE}=\sum_{m=1}^{M}\frac{|B_m|}{n}\left|\operatorname{acc}(B_m)-\operatorname{conf}(B_m)\right|,$$

Neural ensebmle search



Литература

- Bishop C. M., Nasrabadi N. M. Pattern recognition and machine learning. – New York : springer, 2006. – Т. 4. – №. 4. – С. 738.
- Адуенко А. А., Выбор мультимоделей в задачах классификации, 2017 (диссертация)
- Jordan M. I., Jacobs R. A. Hierarchical mixtures of experts and the EM algorithm //Neural computation. – 1994. – Т. 6. – №. 2. – С. 181-214.
- Бахтеев О. Ю., Байесовский выбор субоптимальной структуры модели глубокого обучения, 2020 (диссертация)
- Kuznetsova R., Bakhteev O., Ogaltsov A. Variational learning across domains with triplet information //arXiv preprint arXiv:1806.08672. – 2018.
- Nayman N. et al. Xnas: Neural architecture search with expert advice //Advances in Neural Information Processing Systems. – 2019. – Т. 32.
- Lakshminarayanan B., Pritzel A., Blundell C. Simple and scalable predictive uncertainty estimation using deep ensembles //Advances in neural information processing systems. – 2017. – Т. 30.
- Zaidi S. et al. Neural ensemble search for uncertainty estimation and dataset shift //Advances in Neural Information Processing Systems. – 2021. – Т. 34.
- Guo C. et al. On calibration of modern neural networks //International Conference on Machine Learning. – PMLR, 2017. – С. 1321-1330.