

Neural Active Learning with Performance Guarantees

Zhilei Wang ¹

Joint work with

Pranjal Awasthi ², Christoph Dann ², Claudio Gentile ²,
Ayush Sekhari ³

¹New York University

²Google Research

³Cornell University

Neural Information Processes System 2021

Active learning: **reduce data requirement** in supervised learning by studying the design of algorithms that can **learn** and **generalize** from a small subset of the training data.

Version 1: Pool Based Active Learning

- Has access to a large unlabeled set of data points;
- Can ask for a subset of the data to be labeled

Version 2: Sequential/Streaming Active Learning (**our focus**)

- Data points arrive in a **streaming manner**, adversarially or i.i.d.
- The algorithm must decide whether to query the label of a given point or not

Agnostic statistical learning model:

- A pre-specified class \mathcal{H} of functions either containing the Bayes classifier or has a good approximation inside
- (data, label) pairs are generated in an **i.i.d. fashion**
- **Goal**: query a small number of labels and produce a hypothesis of low error, i.e., get fast convergence rate w.r.t the number of queries the algorithm makes

Two learning settings

Parametric Setting: \mathcal{H} has finite VC-dim (or finite disagreement coefficient)

- Excess risk decays at least as $\nu N^{-1/2}$, ν being the infimum of population loss in class \mathcal{H}
- When $\nu > 0$ active learning \approx passive learning
- Fast rates only under $\nu \approx 0$
- When $\nu \approx 0$: there are (adaptive) algorithms achieving minimax active learning rate $N^{-\frac{\alpha+1}{2}}$ [Hanneke, 2009, Koltchinskii, 2010], where α is the low noise exponent

To shrink the **approximation error** ν , consider wider class of functions, leading to non-parametric learning.

Two learning setting

Non-Parametric Setting: minimax active learning bounds achieved [Locatelli A. and Kpotufe, 2017, Minsker, 2012] assuming

- Marginal distribution $\mathcal{D}_{\mathcal{X}}$ is (quasi-)uniform
- Low noise condition with exponent α
- Regression function is β -Hölder smooth

The results in [Locatelli A. and Kpotufe, 2017, Minsker, 2012] recovers the parametric setting when $\beta \rightarrow \infty$. However, such algorithms are not efficient (curse of dimensionality).

Popular **empirical** approach: use DNNs to perform active learning, no provable guarantees.

Is **provable** and **computationally efficient** active learning possible in **non-parametric** setting?

Our Contributions

Our answer: **yes!** We provide **computationally efficient** algorithms for active learning in **sequential setting** based on DNNs.

- Avoid fixing a function class a-priori
- Use over-parametrized DNNs
- Propose a simple algorithm that forms an uncertainty estimate for the current data point based on the output of a DNN
- Use theory of **Neural Tangent Kernel** (NTK) approximation to analyze the dynamics of GD by considering linearization of the network around random initialization
[Arora et al., 2019, Allen-Zhu et al., 2019, Cao and Gu, 2019]
- Have fast rate of convergence which depend on a data-dependent complexity term under low-noise condition
- Algorithms automatically adapt to the magnitude of the unknown complexity term by a novel model selection approach

Preliminary and Notation

Binary classification: $y \in \{\pm 1\}$.

Non-parametric setting: no assumption on $h(x) = \mathbb{P}(y = 1|x)$.

Low noise condition: [Mammen and Tsybakov, 1999]

$$\mathbb{P}(|h(x) - \frac{1}{2}| < \epsilon) \leq \epsilon^\alpha, \forall \epsilon \in (0, 1/2).$$

Fully connected NN (ReLU activation): $f(x, \theta) = \sqrt{m}W_n\sigma(\dots\sigma(W_1x))$.

NTK matrix: the Gram matrix H of the NTK is a data-dependent matrix defined recursively which measures the complexity of the network projected on given data points [Jacot et al., 2018].

Data dependent complexity terms:

$$L_H = \log \det(I + H), \quad S_{T,n}(h) = \sqrt{\vec{h}^\top H^{-1} \vec{h}}.$$

Quantify the algorithms' performance by **(pseudo-)regret** R_T and **number of queries** N_T .

Goal: bound R_T and N_T simultaneously w.h.p over the generation of the sequence $\{(x_t, y_t)\}_{t=1, \dots, T}$.

Base Learner: Frozen NTK Selective Sampler

Combining techniques from **selective sampling** [Dekel et al., 2012] and **Neural bandits** [Zhou et al., 2020, Zhang et al., 2020] in an original and non-trivial way.

- Input: **complexity parameter** S that upper bounds $S_{T,n}(h)$
- Initialization: sample neurons' weights independently from Gaussian distribution with appropriate variance
- Feature map: $\nabla f(x; \theta_0) / \sqrt{m}$, where θ_0 is the (**frozen**) weight vector of the neural network generated during init
- Generate upper confidence bound and uncertainty threshold
- **Query condition**: if $|u.c.b - 1/2|$ is less than the threshold
- If query condition triggered: updates least-squares estimator θ_t using the feature map

Theorem

Let Frozen NTK Selective Sampler be run with parameter S on an i.i.d. sample $(x_1, y_1), \dots, (x_T, y_T) \sim \mathcal{D}$, where the marginal distribution $\mathcal{D}_{\mathcal{X}}$ fulfills the low-noise condition with exponent $\alpha \geq 0$ and such that $S_{T,n}(h) \leq S$. Then w.h.p R_T and N_T are simultaneously upper bounded as follows:

$$R_T = O\left(\left(L_H(L_H + S^2)\right)^{\frac{\alpha+1}{\alpha+2}} T^{\frac{1}{\alpha+2}}\right)$$
$$N_T = O\left(\left(L_H(L_H + S^2)\right)^{\frac{\alpha}{\alpha+2}} T^{\frac{2}{\alpha+2}}\right),$$

where $L_H = \log \det(I + H)$, H being the NTK matrix of depth n over the training set.

Remark on the guarantees

- Takes as input the complexity parameter $S_{T,n}(h)$ which quantifies the complexity of the function h to be learned projected onto the data
- If h too complex, i.e. $S_{T,n}^2(h) = \Omega(T)$, all bounds vacuous
- If h belongs to the RKHS induced by the NTK, then $S_{T,n}^2(h)$ is upper bounded by the RKHS norm of h
- L_H measure the complexity of DNN in terms of the NTK matrix
- L_H is tightly related to the decaying rate of the eigenvalues of NTK matrix, and is poly-log(T) in many important cases [Valko et al., 2013]

Online to batch conversion

Pick one function uniformly at random, **excess risk** is bounded w.h.p by

$$\left(\frac{L_H(L_H + S^2)}{N_T} \right)^{\frac{\alpha+1}{2}}$$

- $L_H(L_H + S^2)$ plays the role of a compound complexity term projected onto the data x_1, \dots, x_T
- When restricted to VC-class, the convergence rate $N_T^{-\frac{\alpha+1}{2}}$ is the minimax rate
- When L_H is poly-log(T) and $S^2 = O(T^\beta)$ ($\beta < 1$), the excess risk is bounded by

$$N_T^{-\frac{(1-\beta)(\alpha+1)}{2+\alpha\beta}}$$

Model Selection

In practice, L_H and S are a-priori **unknown**.

To make the algorithm **oblivious** to these complexity terms, we operate on a pool of base learners, each being parametrized by (S_i, d_i) .

- S_i plays the role of S
- d_i plays the role of $L_H(L_H + S^2)$

Choose over pool of base learners \mathcal{M}_t with a probability distribution \vec{p}_t where

$$p_{t,i} = \begin{cases} \frac{d_i^{-(\alpha+1)}}{\sum_{j \in \mathcal{M}_t} d_j^{-(\alpha+1)}}, & \text{if } i \in \mathcal{M}_t \\ 0. & \text{otherwise} \end{cases}$$

The algorithm undergoes a series of ad hoc elimination tests (inspired by [Pacchiano et al., 2020a, Pacchiano et al., 2020b]) to drop mis-specified models on the fly.

Guarantees for Base Learners with Model Selection

Theorem

Let base learners with model selection be run on a pool of base learners $\mathcal{M}_1 = \{(2^{i_1}, 2^{i_2})\}$ for $(i_1, i_2) \in [\log T] \times [\log T]$ on an i.i.d. sample $(x_1, y_1), \dots, (x_T, y_T) \sim \mathcal{D}$, where the marginal distribution \mathcal{D}_X fulfills the low-noise condition with exponent $\alpha \geq 0$. Then w.h.p.,

$$R_T = O\left(\left(L_H(L_H + S_{T,n}^2(h))\right)^{\alpha+1} T^{\frac{1}{\alpha+2}}\right),$$

$$N_T = O\left(\left(L_H(L_H + S_{T,n}^2(h))\right)^{\frac{\alpha}{\alpha+2}} T^{\frac{2}{\alpha+2}}\right).$$

- **Excess risk bound:** $\left(\frac{[L_H(L_H + S_{T,n}^2(h))]^{\frac{3\alpha+2}{\alpha+2}}}{N_T}\right)^{\frac{\alpha+1}{2}}$
- If L_H is poly-log(T) and $S^2 = O(T^\beta)$, then excess risk is

$$N_T^{-\frac{(1-\beta(\alpha+1))(\alpha+1)}{2+\beta\alpha}}$$

Non-Frozen Version of Base Algorithm

In practice it is believed that training the DNN parameter by (S)GD is better than fixing the parameters.

We extend all our results to the case where the network weights are not frozen, but are updated according to a GD procedure based on the (data,label) pair queried so far.

Difference compared to frozen version:

- The feature map becomes $\nabla f(x; \theta_{t-1})/\sqrt{m}$
- θ_t trained by GD on the labeled data gathered so far

Key Ingredient: the approximation result between the neural network f and its first order approximation in the over-parametrized regime.

[Arora et al., 2019, Allen-Zhu et al., 2019, Cao and Gu, 2019]

Conclusions

- A **rigorous analysis** of selective sampling and active learning in general **non-parametric scenarios**
- We are in the most liberal non-parametric regime **without any constraint** on the regression function h , with the consequence that all key complexity terms are **data-dependent**
- Algorithms automatically adapt to the magnitude of the unknown complexity term by model selection
- Gives rise to **efficient and manageable** algorithms for modular **DNN architecture** design and deployment



Allen-Zhu, Z., Li, Y., and Song, Z. (2019).

A convergence theory for deep learning via over-parameterization.

In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 242–252. PMLR.



Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R., and Wang, R. (2019).

On exact computation with an infinitely wide neural net.

In *Advances in Neural Information Processing Systems*.
Curran Associates, Inc.



Cao, Y. and Gu, Q. (2019).

Generalization bounds of stochastic gradient descent for wide and deep neural networks.

In Advances in Neural Information Processing Systems.
Curran Associates, Inc.



Dekel, O., Gentile, C., and Sridharan, K. (2012).

Selective sampling and active learning from single and multiple teachers.




J. Mach. Learn. Res., 13(1).






Hanneke, S. (2009).




Adaptive rates of convergence in active learning.

In Proc. of the 22th Annual Conference on Learning Theory.

-  Jacot, A., Gabriel, F., and Hongler, C. (2018).
Neural tangent kernel: convergence and generalization in neural networks.
In Advances in neural information processing systems, page 8571–8580. MIT Press.
-  Koltchinskii, V. (2010).
Rademacher complexities and bounding the excess risk of active learning.
Journal of Machine Learning Research, 11:2457–2485.
-  Locatelli A., C. A. and Kpotufe, S. (2017).
Adaptivity to noise parameters in nonparametric active learning.
In Proceedings of the 2017 Conference on Learning Theory, volume 65 of *Proceedings of Machine Learning Research*, pages 1383–1416.

References IV

-  Mammen, E. and Tsybakov, A. (1999).
Smooth discrimination analysis.
The Annals of Statistics, 27(6):1808–1829.
-  Minsker, S. (2012).
Plug-in approach to active learning.
Journal of Machine Learning Research, 13:67–90.
-  Pacchiano, A., Dann, C., C., G., and Bartlett, P. (2020a).
Regret bound balancing and elimination for model selection
in bandits and RL.
arXiv preprint arXiv:2012.13045.

-  Pacchiano, A., Phan, M., Abbasi Yadkori, Y., Rao, A., Zimmert, J., Lattimore, T., and Szepesvari, C. (2020b). Model selection in contextual stochastic bandit problems. In *Advances in Neural Information Processing Systems*, volume 33, pages 10328–10337. Curran Associates, Inc.
-  Valko, M., Korda, N., Munos, R., Flaounas, I., and Cristianini, N. (2013). Finite-time analysis of kernelised contextual bandits. In *arxiv:1309.6869*.
-  Zhang, W., Zhou, D., Li, L., and Gu, Q. (2020). Neural thompson sampling. In *arXiv:2010.00827*.



Zhou, D., Li, L., and Gu, Q. (2020).

Neural contextual bandits with ucb-based exploration.

In Proceedings of the 37th International Conference on Machine Learning.