Institute for Digital Communications
Prof. Dr.-Ing. Ralf Müller
Friedrich-Alexander-Universität Erlangen-Nürnberg

Research Project

Centroid Informed Approximate Nearest Convex Hull Classification

Background Deep learning and specifically neural networks (NNs) [5] have emerged as some of the most powerful algorithms applicable to a wide range of problems [4, 7, 6]. Due to the universal-approximation theorem, ReLU networks are typically assumed to be able to implement any function. However, not any function may be learnable by backpropagation and gradient descent under general assumptions and with finite number of training samples. Recent work found that certain convex separable problems are hard to learn for multi-layer perceptrons (MLPs) while being solvable with vanishing generalization error by nearest convex hull classification (NCHC) [2]. Applying NCHC to a classification task of n classes requires to solve n constrained convex optimization problems. These can be solved by projected gradient descent [3]. NCHC, hence, moves the computational load of gradient descent from training to inference, and is, therefore, prohibitively complex for real-world deployment.

Approximate Nearest Convex Hull Classification Approximate algorithms to estimate the distance of a point to a convex hull exist [1]. These are however neglected in the literature as being to complex [8]. An alternative approach may be found in approximating the convex hull by a set of M_n planes which touch the hull tangential, as illustrated in Fig. 1. Consider one plane defined by its normal vector $\mathbf{w}_i$ and bias b_i. The plane effectively cuts the space into two half-planes: The positive half-plane in which $\mathbf{w}_i^T\mathbf{x} + b_i > 0$, illustrated in green, and the corresponding negative half plane for which $\mathbf{w}_i^T\mathbf{x} + b_i < 0$. Adding more half-planes ($M_n \uparrow$) cuts out more precise areas of the space such that we can asymptotically isolate the convex hull, illustrated in purple. Given a point $\mathbf{x}$, we aim to compute its distance to the convex hull, $d^1(\mathbf{x})$. This distance shall be obtained as a function of the distances to the M_n planes: $\mathbf{d}^1(\mathbf{x}) = [d_1^1(\mathbf{x}), \ldots, d_{M_n}^1(\mathbf{x})]^T$. If $\mathbf{x}$ is located on the flat side of the hull, i.e. is closer to the centroid, then the infinity norm is a good estimate: $d^1(\mathbf{x}) \approx ||\mathbf{d}^1(\mathbf{x})||_\infty = \max_i d_i^1(\mathbf{x})$. If the distance to the centroid is large, this is no longer a good approximation. Instead, by observing that the cut-out half-planes are asymptotically orthogonal in high-dimensional space, we note that the L2-norm is now a good estimator $d^1(\mathbf{x}) \approx ||\mathbf{d}^1(\mathbf{x})||_2$. Approximating the distance to the hull by the L2-norm is referred to as approximate NCHC (ANCHC).

Overall Project Aim The project aims to refine the simple ANCHC algorithm. This shall be done by taking the distance of each plane to the centroid of the hull into account. We refer to this approach as centroid informed approximate NCHC (CNCHC). An initial mathematical study of the problem is followed by an implementation phase in which an improved distance estimator shall be developed. The obtained algorithm is then evaluated to NCHC, ANCHC and the algorithm presented in [1] from a performative and computational complexity perspective.

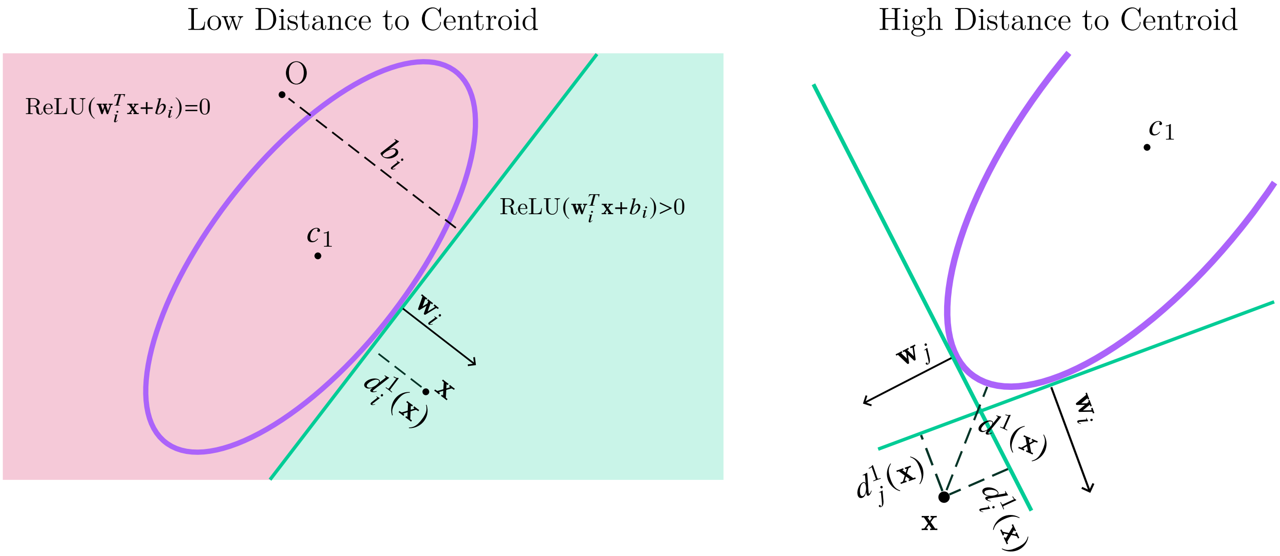

Figure 1: Aproximating a convex hull (purple) by a set of planes.

Project Objectives The main objectives of this thesis can be summarized as follows:

- Mathematical study of the underlying problem and the relation of the distance metric to the centroid of a convex hull.
- Development and Implementation of CNCHC (analytically or by deep-learning methods).
- Refinement of CNCHC.
- Implementation and Evaluation of the approximate algorithm presented in [1].

Prerequisites The following qualifications are required in order to pursue this project:

- Very Good knowledge of linear algebra.
- Basic knowledge of (convex) optimization.
- Good programming skills in Python (JAX, PyTorch or Haiku, numpy).
- Beneficial: foundational knowledge on random matrices and free probability theory.
- Beneficial: experience with Git.
- Beneficial: experience with convex solvers (e.g. cvxpy or jaxopt).
- Beneficial: experience with bash and slurm.

Bibliography

[1] Avrim Blum, Sariel Har-Peled, and Benjamin Raichel. "Sparse Approximation via Generating Point Sets". en. In: *ACM Transactions on Algorithms* 15.3 (July 2019), pp. 1–16. DOI: 10.1145/3302249.

[2] Levi-Pascal V. G. Bohnacker and Ralf R. Müller. "Learning Convex Decision Regions is Hard: When Neural Networks Fail on Convexity". Unpublished manuscript.

[3] Sébastien Bubeck. "Convex Optimization: Algorithms and Complexity". en. In: *Foundations and Trends in Machine Learning* 8.3-4 (Nov. 2015), pp. 231–357. DOI: 10.1561/2200000050.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet classification with deep convolutional neural networks". en. In: *Communications of the ACM* 60.6 (May 2017), pp. 84–90. DOI: 10.1145/3065386.

[5] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". en. In: *Nature* 521.7553 (May 2015), pp. 436–444. DOI: 10.1038/nature14539.

[6] James Murphy, John E Ward, and Brian Mac Namee. "An Overview of Machine Learning Techniques for Onboard Anomaly Detection in Satellite Telemetry". en. In: *2023 European Data Handling & Data Processing Conference (EDHPC)*. Juan Les Pins, France, Oct. 2023, pp. 1–6. DOI: 10.23919/EDHPC59100.2023.10396403.

[7] Ashish Vaswani et al. "Attention is All you Need". en. In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc., 2017.

[8] Roozbeh Yousefzadeh. "Deep learning generalization and the convex hull of training sets". en. In: *Information Geometry* (Oct. 2025). DOI: 10.1007/s41884-025-00178-z.

SUPERVISOR Levi-Pascal Bohnacker, levi.bohnacker@fau.de, Room 04.037

Start date: TBD
End date: TBD

Prof. Dr.-Ing. Ralf R. Müller