Parametric Matrix Models for General Machine Learning

Patrick Cook

Facility for Rare Isotope Beams
Michigan State University

STREAMLINE Symposium
May 2024

Theory Alliance

Introduction

Picking up where Danny left off...

Introduction

Picking up where Danny left off...

- Parametric Matrix Models (PMMs) are powerful for sparse and noisy data

Introduction

Picking up where Danny left off...

- Parametric Matrix Models (PMMs) are powerful for sparse and noisy data
- This strongly suggests they are parameter- and computationally-efficient and robust

Introduction

Picking up where Danny left off...

- Parametric Matrix Models (PMMs) are powerful for sparse and noisy data
- This strongly suggests they are parameter- and computationally-efficient and robust

Does this advantage generalize to general machine learning problems?

Why should it matter?

Why should it matter?

General machine learning is invaluable

Why should it matter?

General machine learning is invaluable[citation needed]

Why should it matter?

General machine learning is invaluable[citation needed]

The number of trainable parameters determines the cost and flexibility of a model

Why should it matter?

General machine learning is invaluable[citation needed]

The number of trainable parameters determines the cost and flexibility of a model

- Training time and difficulty
- Dataset size

Why should it matter?

General machine learning is invaluable[citation needed]

The number of trainable parameters determines the cost and flexibility of a model

- Training time and difficulty
- Dataset size
- Model storage
- Inference delay

Why should it matter?

General machine learning is invaluable[citation needed]

The number of trainable parameters determines the cost and flexibility of a model

- Training time and difficulty
- Dataset size
- Model storage
- Inference delay

These factors directly determine the applications of a model

Why should it matter?

General machine learning is invaluable[citation needed]

The number of trainable parameters determines the cost and flexibility of a model

- Training time and difficulty
- Dataset size
- Model storage
- Inference delay

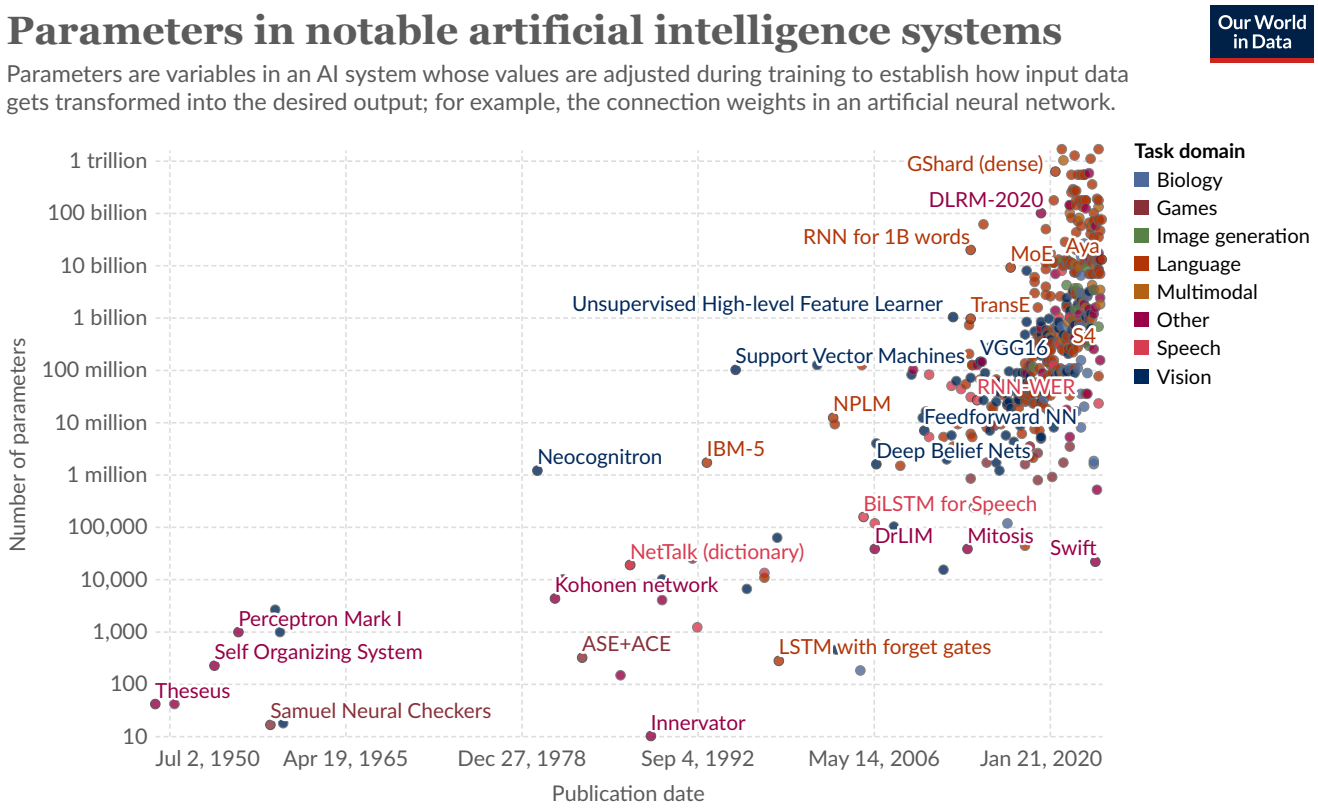
These factors directly determine the applications of a model

- Real-time applications
- Mobile platforms
- Resource-constrained environments

Current “Solution”

Current "Solution"

More parameters, more data, more compute, more time



Generalizing PMMs

Generalizing PMMs

Make a "Hamiltonian"

Use its eigensystem

Generalizing PMMs

Make a "Hamiltonian"

Use its eigensystem

$$H(\mathbf{x})$$

Generalizing PMMs

Make a "Hamiltonian"

$$H(\mathbf{x})$$

Use its eigensystem

$$H(\mathbf{x}) \rightarrow V\Lambda V^{\dagger} \rightarrow \mathbf{z}$$

Generalizing PMMs

Make a "Hamiltonian"

$$H(\mathbf{x})$$

$$H(\mathbf{x}) = \underline{H_0} + \sum_i x_i \underline{H_i}$$

Use its eigensystem

$$H(\mathbf{x}) \rightarrow V \Lambda V^\dagger \rightarrow \mathbf{z}$$

Generalizing PMMs

Make a "Hamiltonian"

$$H(\mathbf{x})$$

$$H(\mathbf{x}) = \underline{H_0} + \sum_i x_i \underline{H_i}$$

$$H(\mathbf{x}) = \underline{H_0} + \mathbf{x}\underline{H_1}\mathbf{x}^\dagger$$

Use its eigensystem

$$H(\mathbf{x}) \to V\Lambda V^\dagger \to \mathbf{z}$$

Generalizing PMMs

Make a "Hamiltonian"

$$H(\mathbf{x})$$

$$H(\mathbf{x}) = \underline{H_0} + \sum_i x_i \underline{H_i}$$

$$H(\mathbf{x}) = \underline{H_0} + \mathbf{x}\underline{H_1}\mathbf{x}^\dagger$$

$$H(\mathbf{X}) = \underline{H_0} + \underline{H_1}\mathbf{X}\underline{H_1}^\dagger$$

Use its eigensystem

$$H(\mathbf{x}) \to V\Lambda V^\dagger \to \mathbf{z}$$

Generalizing PMMs

Make a "Hamiltonian"

$$H(\mathbf{x})$$

$$H(\mathbf{x}) = \underline{H_0} + \sum_i x_i \underline{H_i}$$

$$H(\mathbf{x}) = \underline{H_0} + \mathbf{x}\underline{H_1}\mathbf{x}^\dagger$$

$$H(\mathbf{X}) = \underline{H_0} + \underline{H_1}\mathbf{X}\underline{H_1}^\dagger$$

Use its eigensystem

$$H(\mathbf{x}) \to V\Lambda V^\dagger \to \mathbf{z}$$

$$\mathbf{z} = \mathrm{diag}\,(\Lambda)$$

Generalizing PMMs

Make a "Hamiltonian"

$$H(\mathbf{x})$$

$$H(\mathbf{x}) = \underline{H_0} + \sum_i x_i \underline{H_i}$$

$$H(\mathbf{x}) = \underline{H_0} + \mathbf{x}\underline{H_1}\mathbf{x}^\dagger$$

$$H(\mathbf{X}) = \underline{H_0} + \underline{H_1}\mathbf{X}\underline{H_1}^\dagger$$

Use its eigensystem

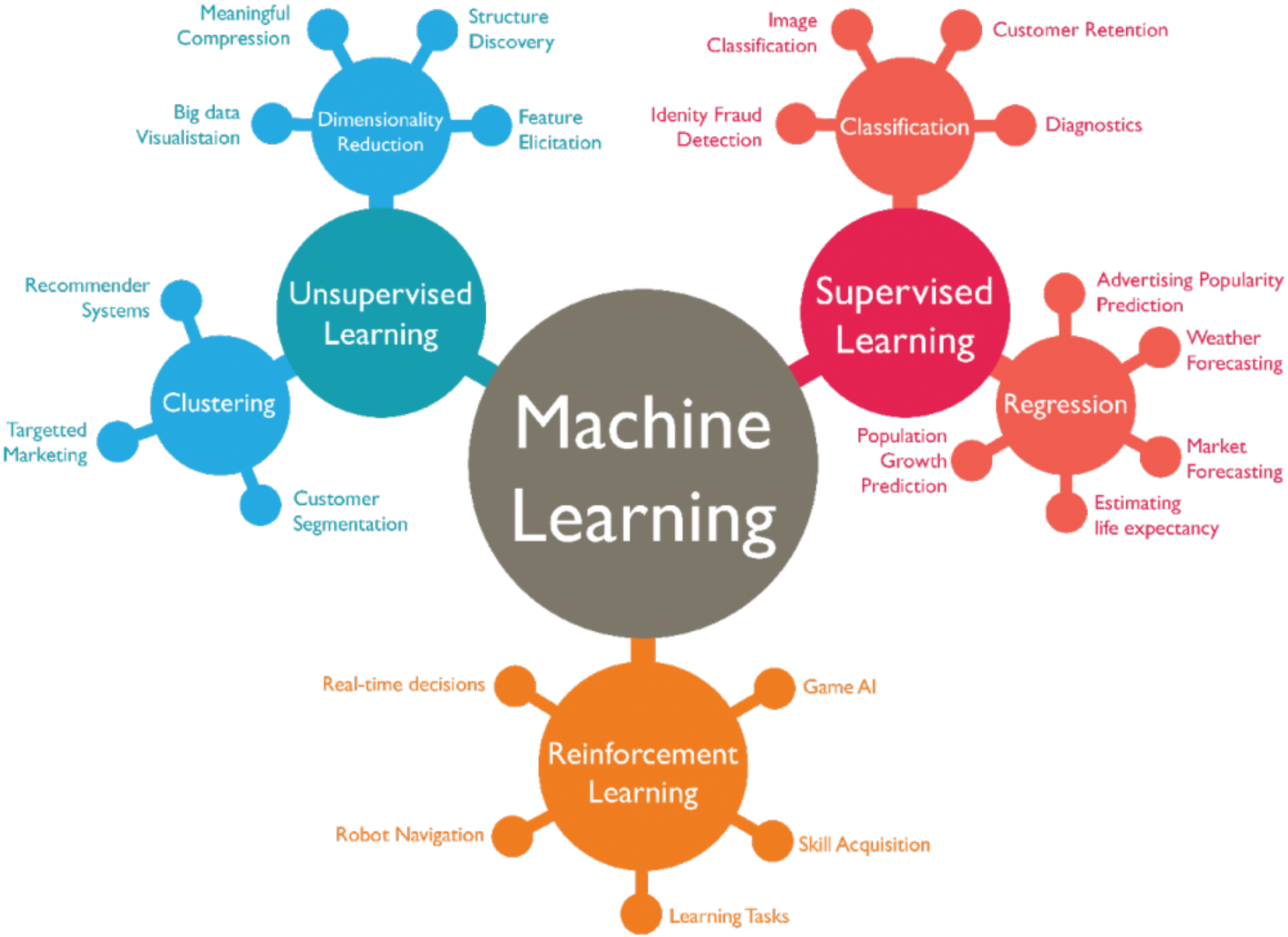
$$H(\mathbf{x}) \to V\Lambda V^\dagger \to \mathbf{z}$$

$$\mathbf{z} = \mathrm{diag}\,(\Lambda)$$

$$z_i = v_i^\dagger \underline{\Delta} v_i$$

Generalizing PMMs

Make a "Hamiltonian"	**Use its eigensystem**
$H(\mathbf{x})$	$H(\mathbf{x}) \rightarrow V\Lambda V^\dagger \rightarrow \mathbf{z}$
$H(\mathbf{x}) = \underline{H_0} + \sum_i x_i \underline{H_i}$	$\mathbf{z} = \mathrm{diag}\,(\Lambda)$
$H(\mathbf{x}) = \underline{H_0} + \mathbf{x}\underline{H_1}\mathbf{x}^\dagger$	$z_i = v_i^\dagger \underline{\Delta} v_i$
$H(\mathbf{X}) = \underline{H_0} + \underline{H_1}\mathbf{X}\underline{H_1}^\dagger$	$z_i = v_0^\dagger \underline{\Delta_i} v_0$


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Regression PMM

Regression PMM

Make a "Hamiltonian"

Use its eigensystem

Regression PMM

Make a "Hamiltonian"

Use its eigensystem

$$H(\mathbf{x}) = \underline{H_0} + \sum_i x_i \underline{H_i}$$

Regression PMM

Make a "Hamiltonian"

$$H(\mathbf{x}) = \underline{H_0} + \sum_i x_i \underline{H_i}$$

Use its eigensystem

$$z_i = v_i^\dagger \underline{\Delta_i} v_i$$

Using the eigensystem more effectively

Using the eigensystem more effectively

How can we use more of the information in the eigensystem?

Using the eigensystem more effectively

How can we use more of the information in the eigensystem?

$$z_k = \sum_{i \leq j} v_i^\dagger \underline{\Delta_{ijk}} v_j$$

Using the eigensystem more effectively

How can we use more of the information in the eigensystem?

$$z_k = \sum_{i \leq j} \left| v_i^\dagger \underline{\Delta_{ijk}} v_j \right|^2$$

Using the eigensystem more effectively

How can we use more of the information in the eigensystem?

$$z_k = \sum_{i \leq j} \left| v_i^\dagger \underline{\Delta_{ijk}} v_j \right|^2 - \frac{1}{2} \left\| \underline{\Delta_{ijk}} \right\|_2^2$$

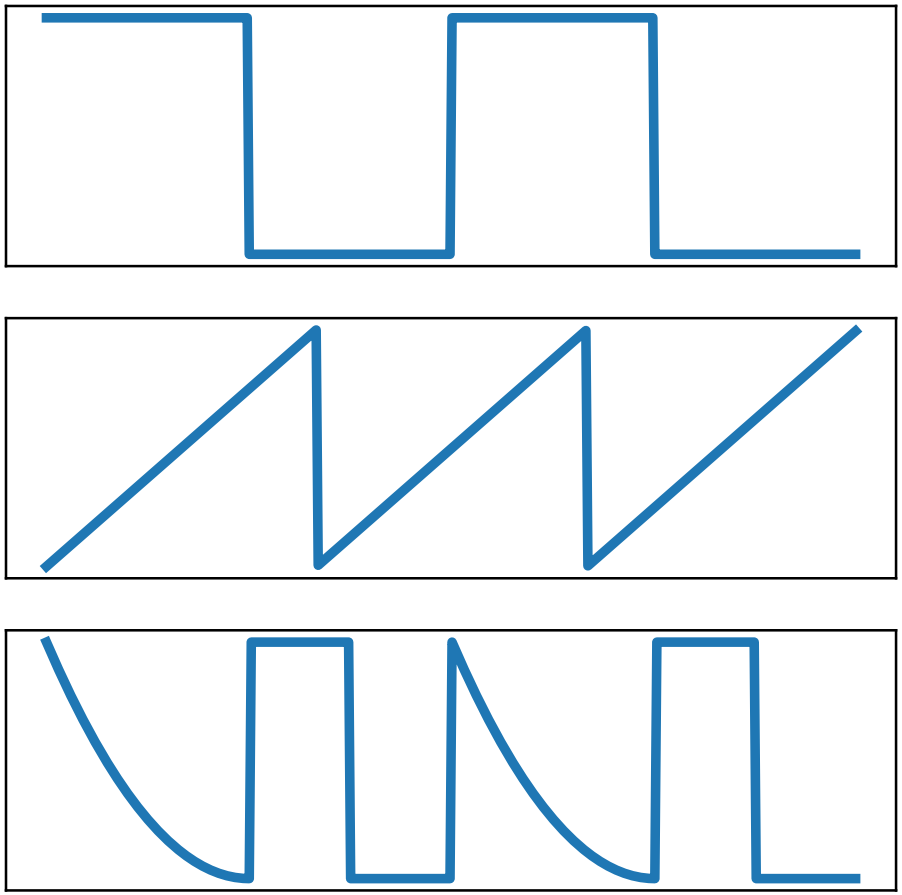
Using the eigensystem more effectively

How can we use more of the information in the eigensystem?

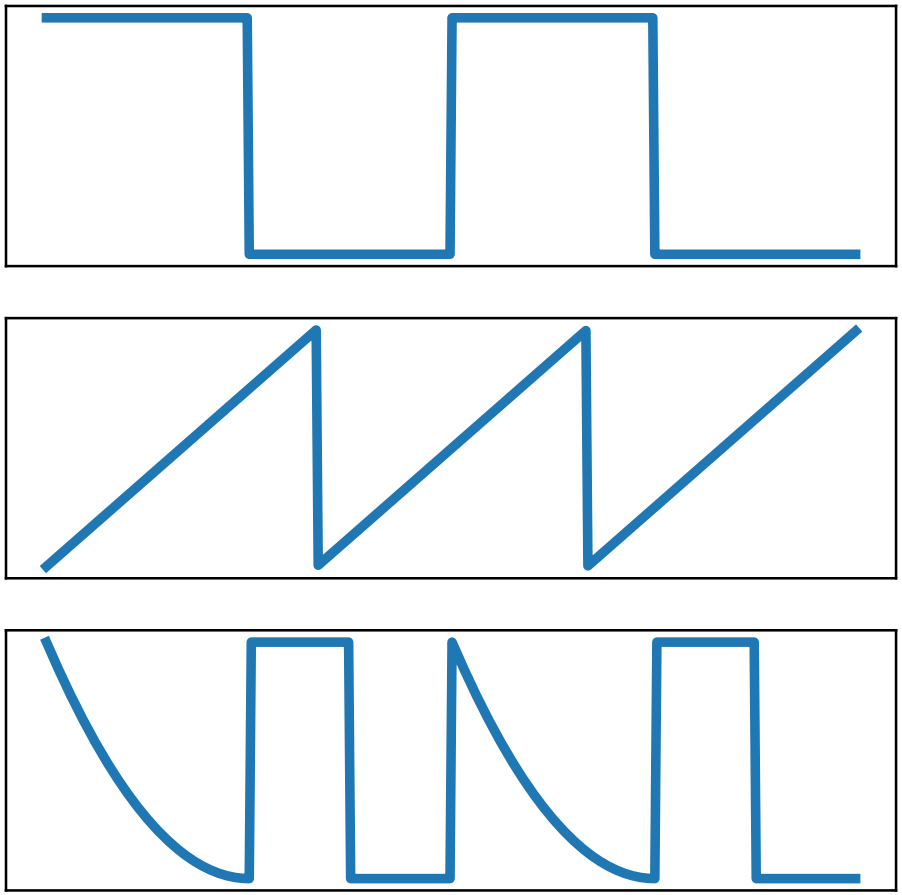
$$z_k = \sum_{i \leq j} \left| v_i^\dagger \underline{\Delta_{ijk}} v_j \right|^2 - \frac{1}{2} \left\| \underline{\Delta_{ijk}} \right\|_2^2$$

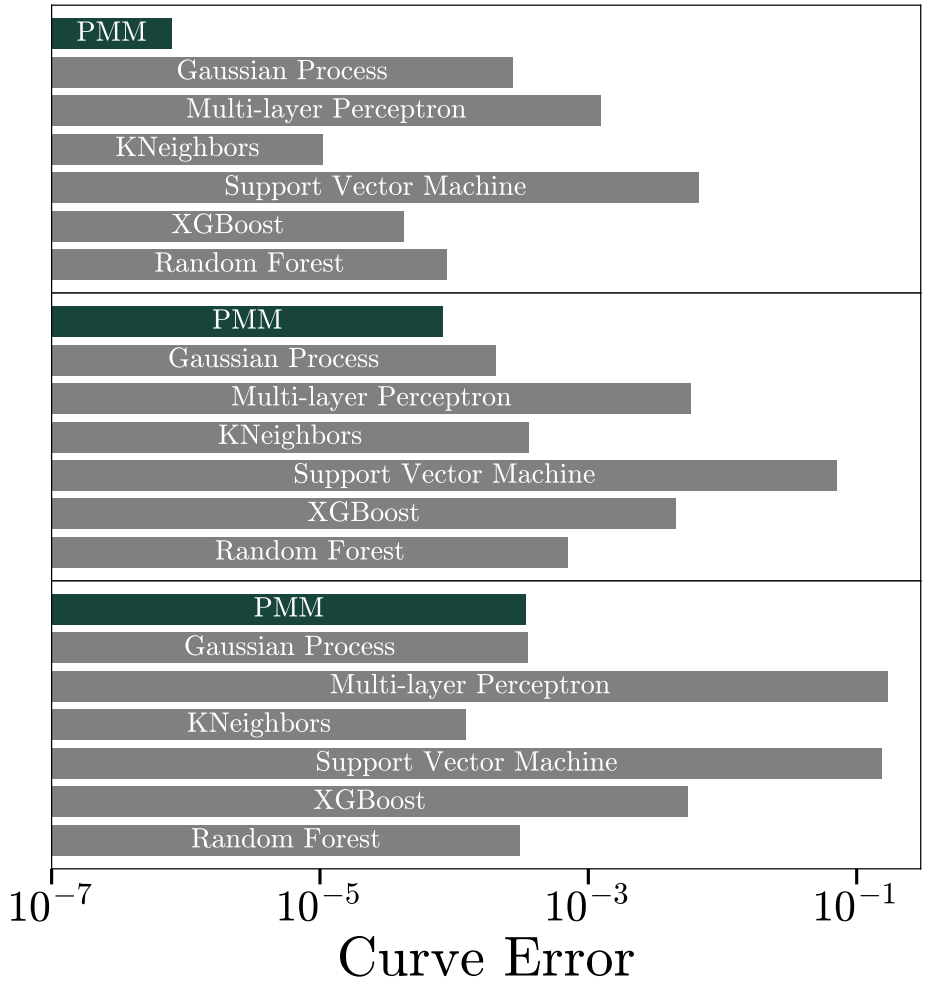
Each $\underline{\Delta_{ijk}}$ is known as a "decoder" matrix

1D Waveforms

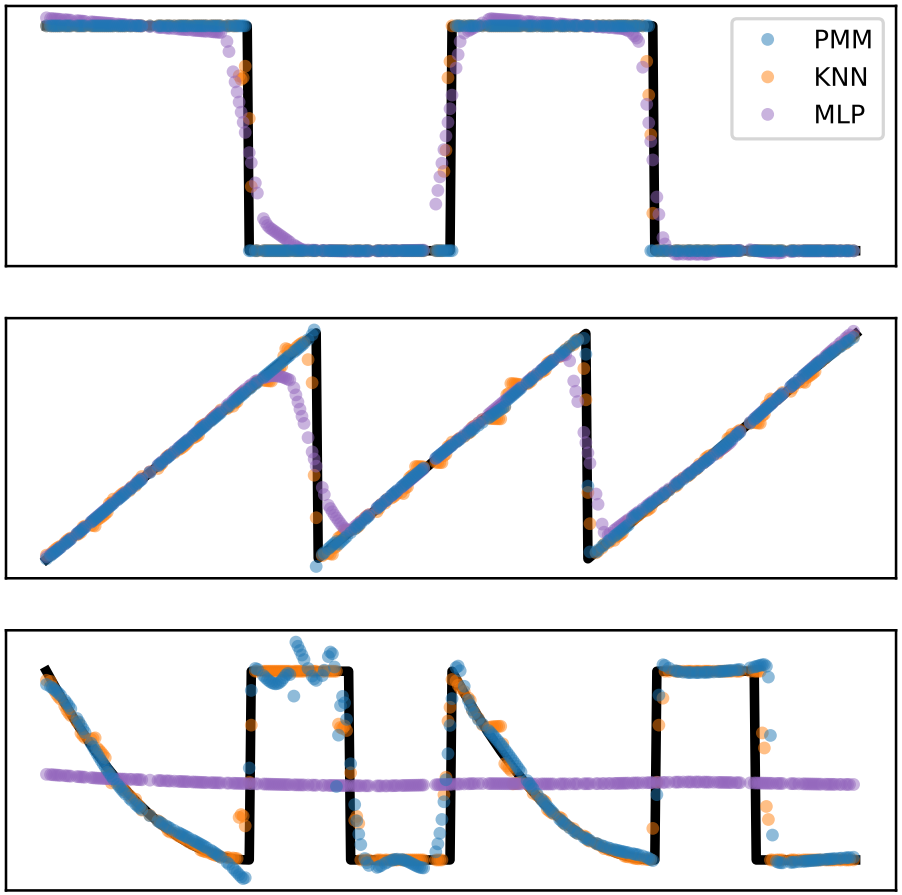


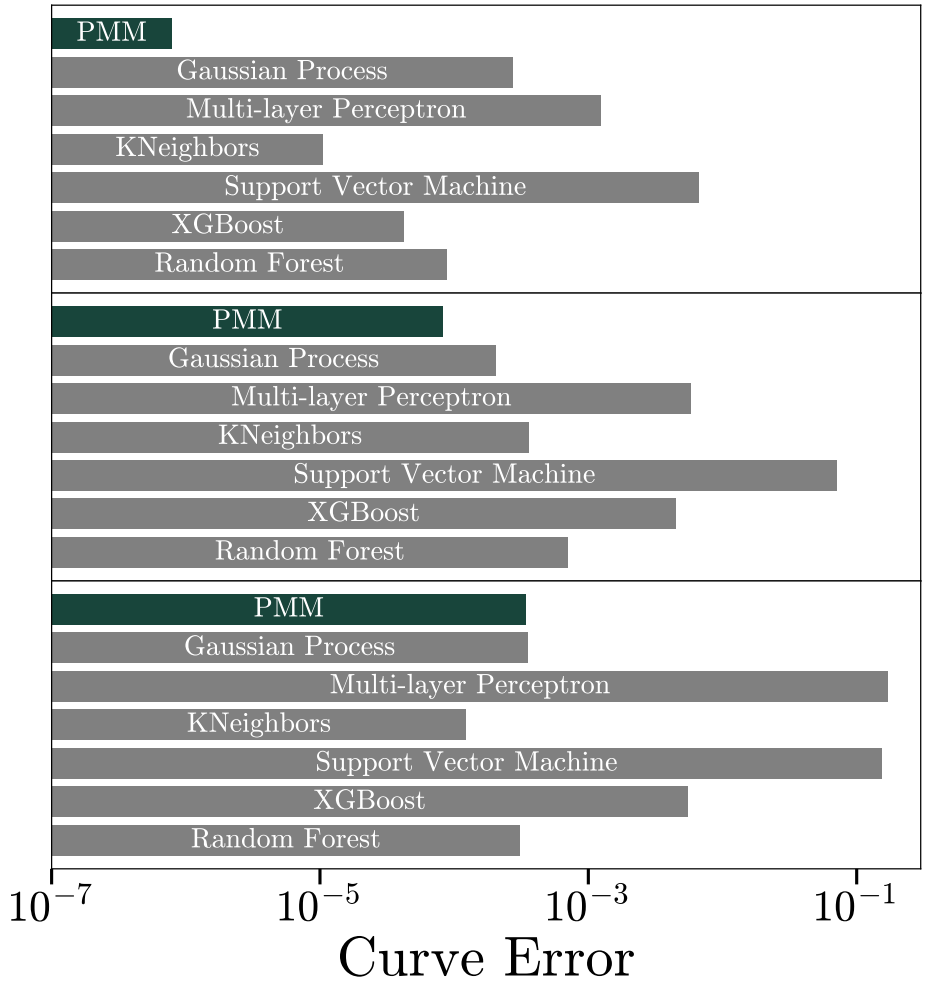
1D Waveforms





1D Waveforms

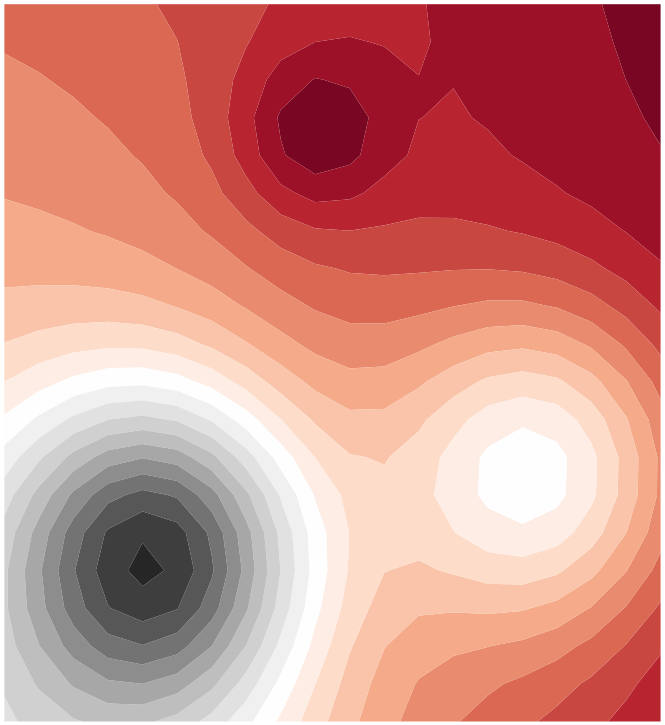


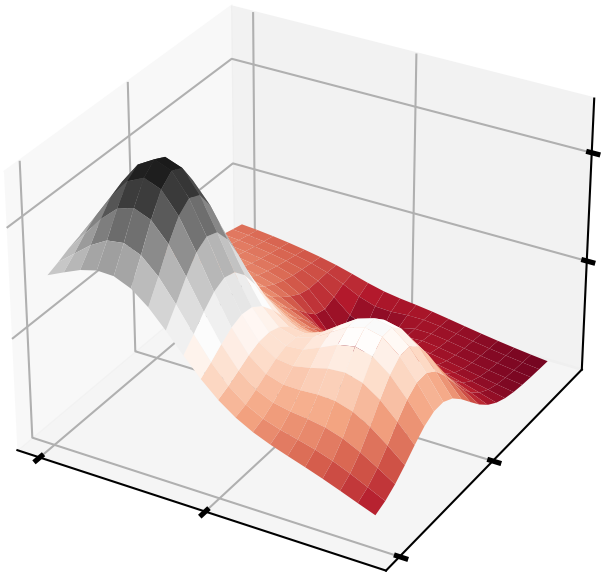


2D Surfaces

2D Surfaces

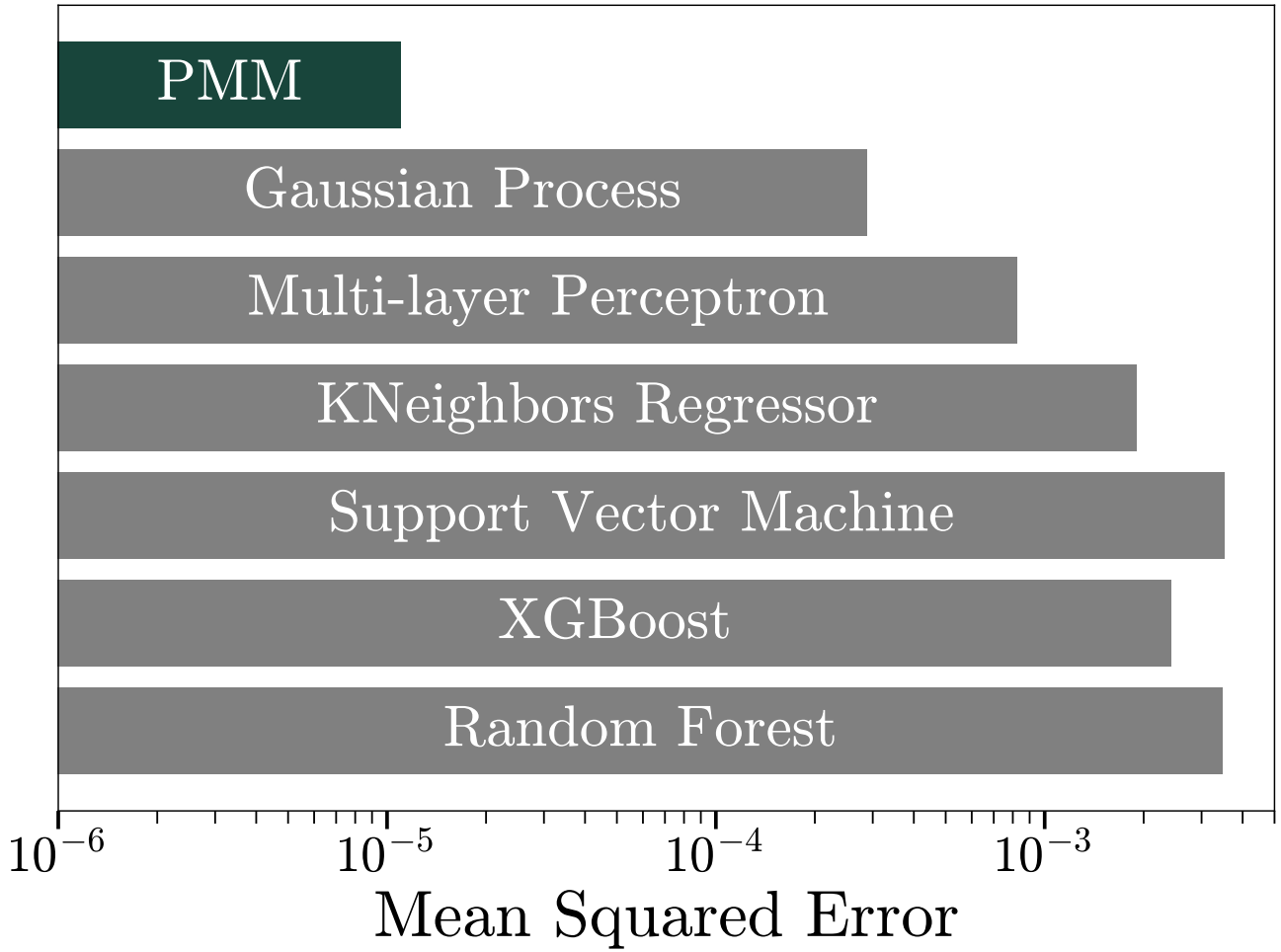
Franke's Bivariate Test Function





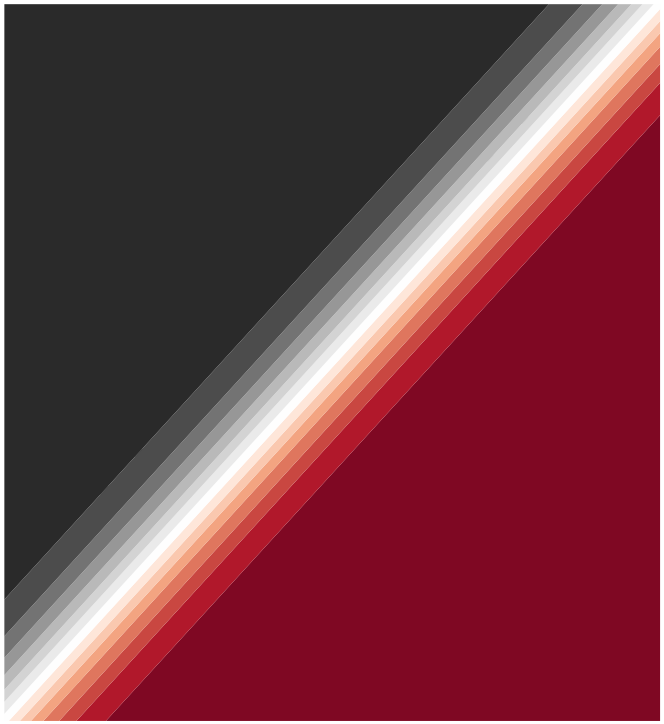
2D Surfaces

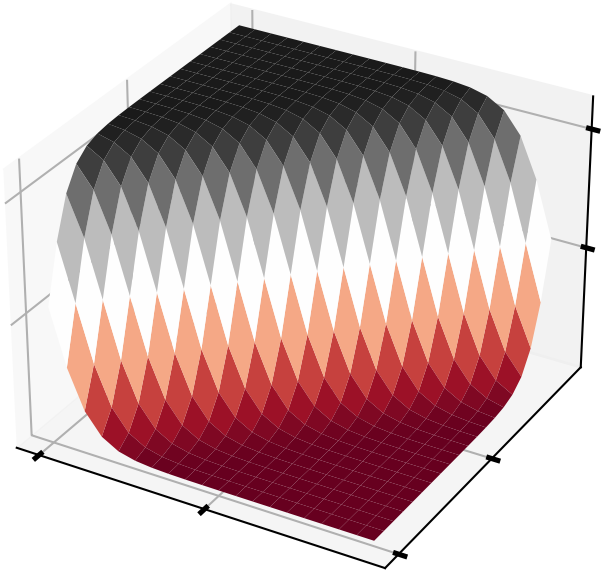
Franke's Bivariate Test Function



2D Surfaces

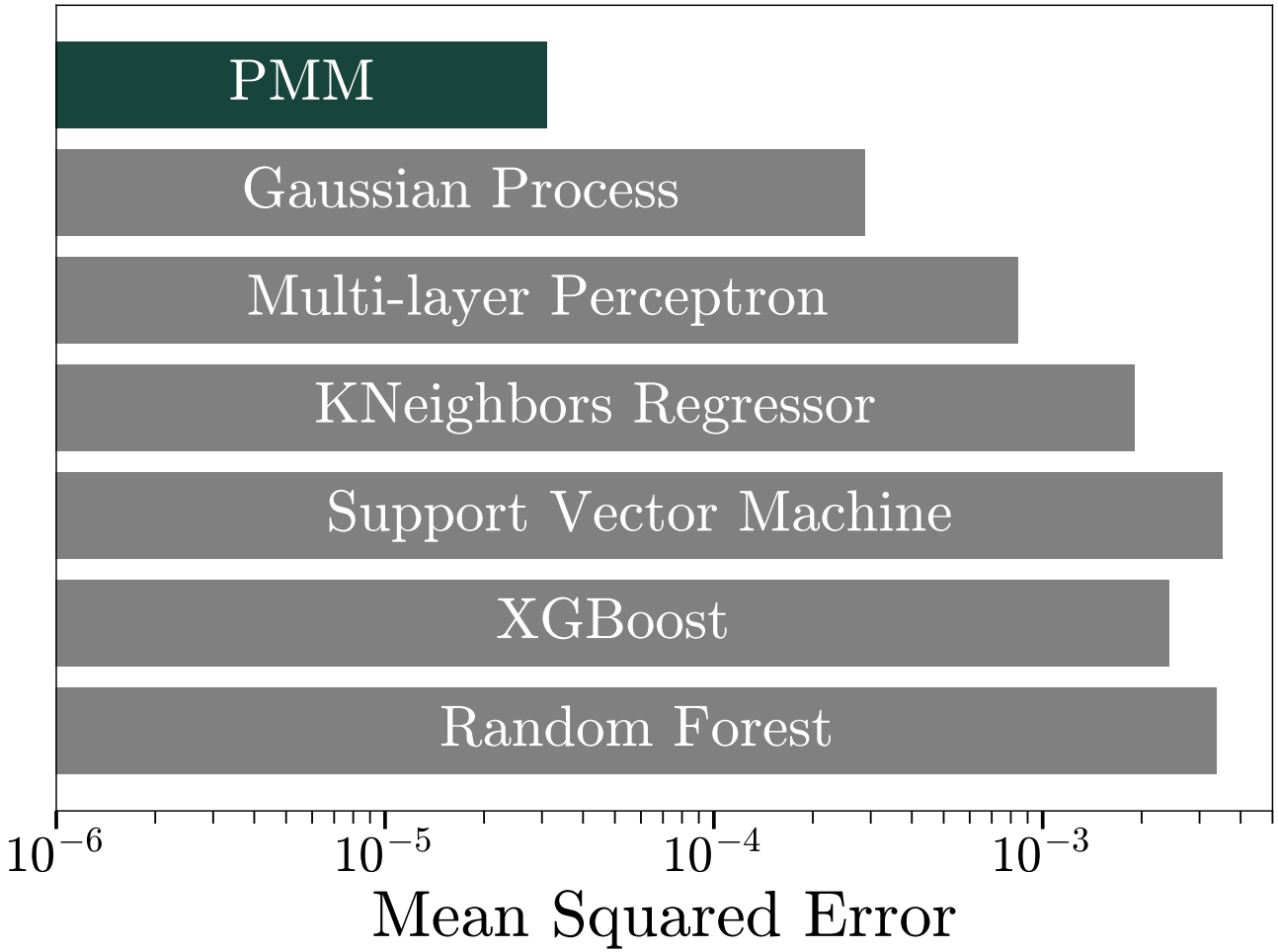
Cliff Function





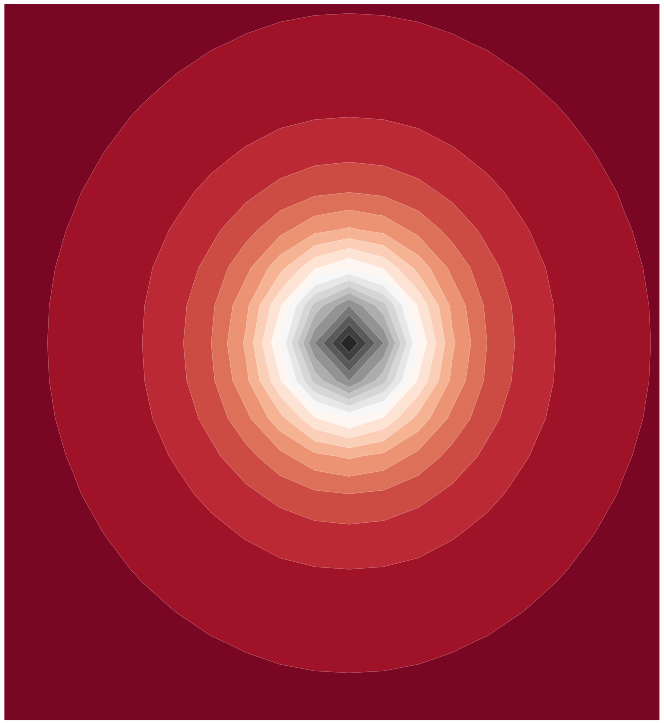
2D Surfaces

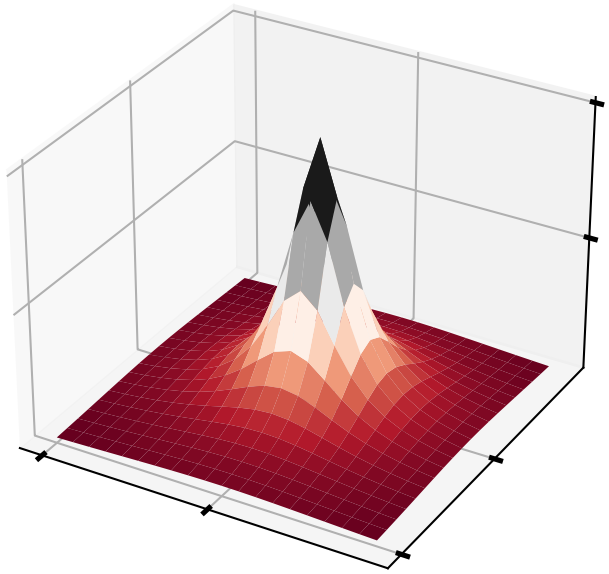
Cliff Function



2D Surfaces

Runge Function





2D Surfaces

Runge Function

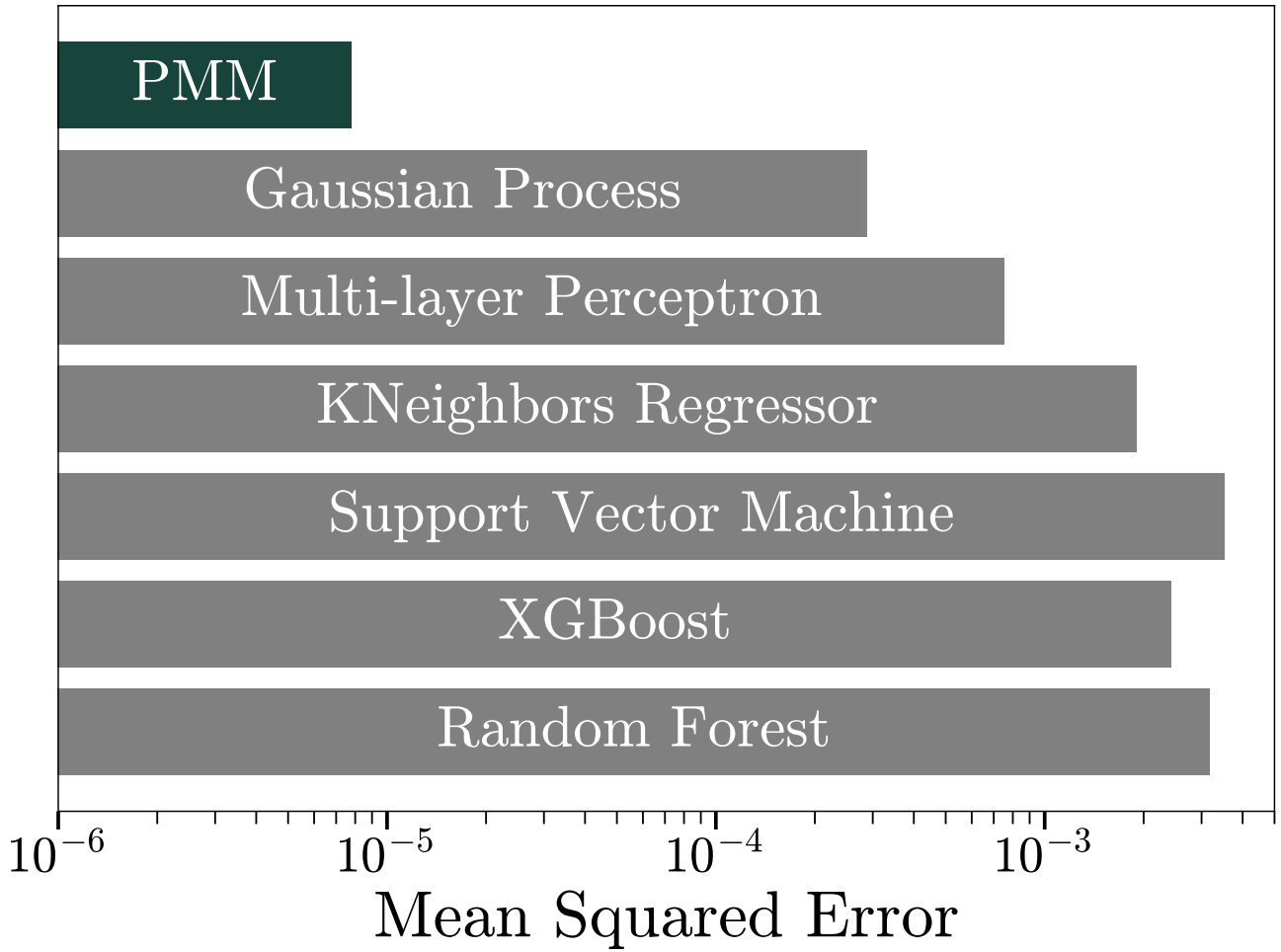


Image Classification PMM

Image Classification PMM

Make a "Hamiltonian"

Use its eigensystem

Image Classification PMM

Make a "Hamiltonian"

Use its eigensystem

$$H(\mathbf{X}) =$$

Image Classification PMM

Make a "Hamiltonian"

Use its eigensystem

$$H(\mathbf{X}) = \ ?$$

Image Classification PMM

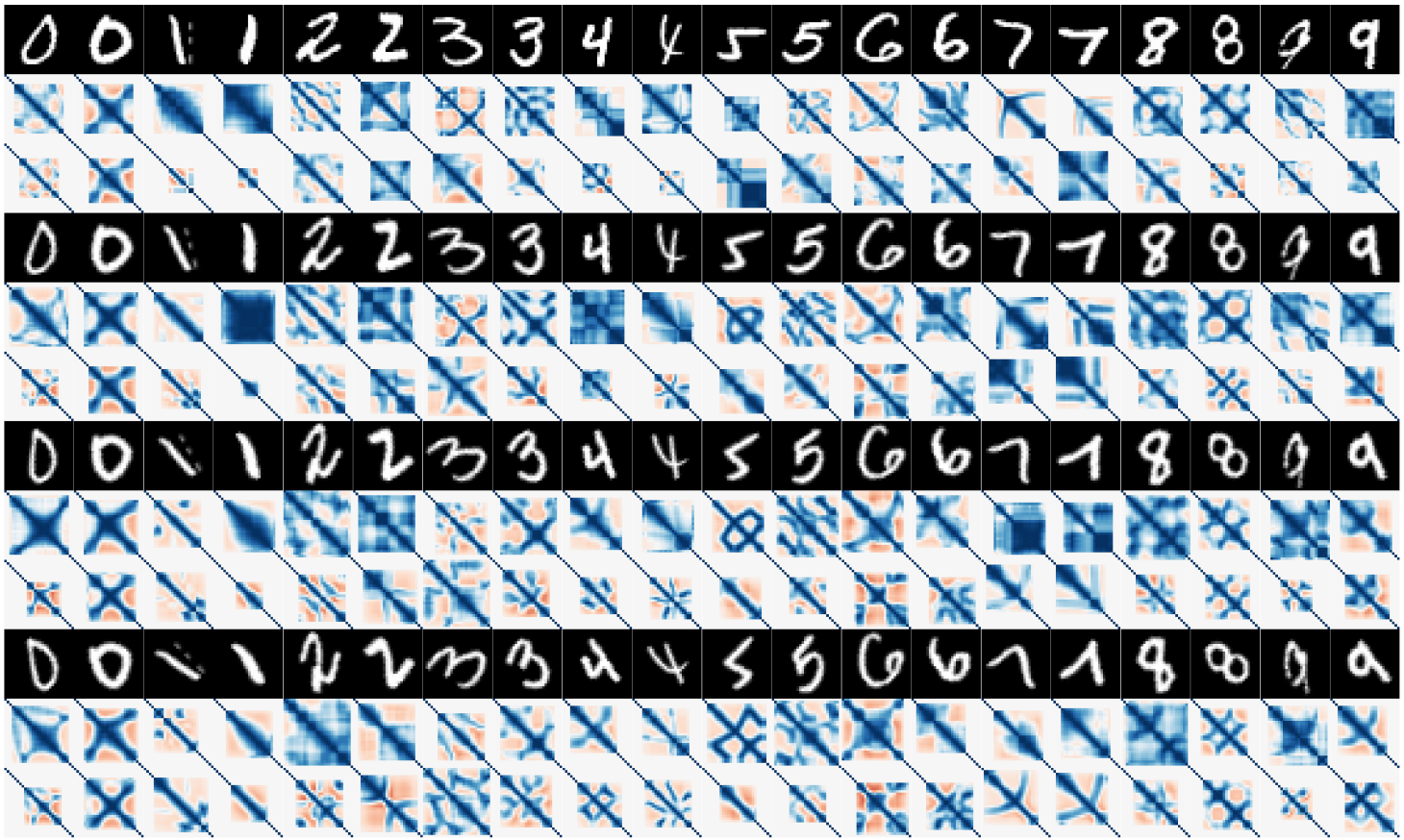
Make a "Hamiltonian"

$$H(\mathbf{X}) = \ ?$$

Use its eigensystem

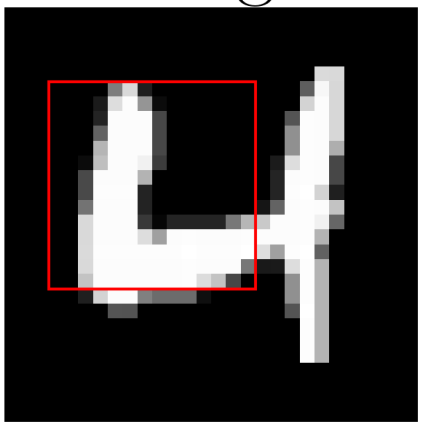
Decoder formalism

Images as Hermitian Matrices

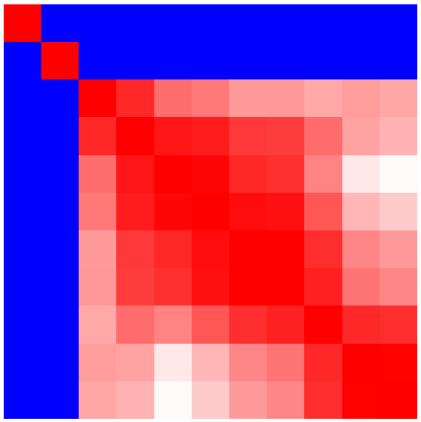


Images as Hermitian Matrices

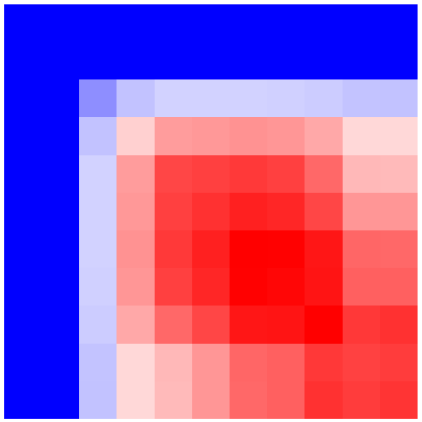
Image



C_R



$XX^\dagger$



Make a "Hamiltonian"

Use its eigensystem

$$\mathbf{X} \xrightarrow{\text{blocks}} X_i$$

Make a "Hamiltonian"

$$\mathbf{X} \xrightarrow{\text{blocks}} X_i$$

$$R_i \mathrel{\widetilde{\propto}} X_i X_i^\dagger$$
$$C_i \mathrel{\widetilde{\propto}} X_i^\dagger X_i$$

Use its eigensystem

Make a "Hamiltonian"

Use its eigensystem

$$\mathbf{X} \xrightarrow{\text{blocks}} X_i$$

$$R_i \mathrel{\tilde{\propto}} X_i X_i^\dagger$$

$$C_i \mathrel{\tilde{\propto}} X_i^\dagger X_i$$

$$M = \sum_i \underline{K_i} R_i \underline{K_i}^\dagger + \underline{L_i} C_i \underline{L_i}^\dagger$$

Make a "Hamiltonian"

Use its eigensystem

$$\mathbf{X} \xrightarrow{\text{blocks}} X_i$$

$$R_i \mathrel{\tilde{\propto}} X_i X_i^\dagger$$

$$C_i \mathrel{\tilde{\propto}} X_i^\dagger X_i$$

$$M = \sum_i \underline{K_i} R_i \underline{K_i}^\dagger + \underline{L_i} C_i \underline{L_i}^\dagger$$

$$H_c = \underline{D_c} M \underline{D_c}^\dagger + \underline{B_c}$$

Make a "Hamiltonian"

$$\mathbf{X} \xrightarrow{\text{blocks}} X_i$$

$$R_i \,\widetilde{\propto}\, X_i X_i^\dagger$$
$$C_i \,\widetilde{\propto}\, X_i^\dagger X_i$$

$$M = \sum_i \underline{K_i} R_i \underline{K_i}^\dagger + \underline{L_i} C_i \underline{L_i}^\dagger$$

$$H_c = \underline{D_c} M \underline{D_c}^\dagger + \underline{B_c}$$

Use its eigensystem

$$H_c \to V_c \Lambda_c V_c^\dagger$$

Make a "Hamiltonian"

$$\mathbf{X} \xrightarrow{\text{blocks}} X_i$$

$$R_i \,\widetilde{\propto}\, X_i X_i^\dagger$$
$$C_i \,\widetilde{\propto}\, X_i^\dagger X_i$$

$$M = \sum_i \underline{K_i} R_i \underline{K_i}^\dagger + \underline{L_i} C_i \underline{L_i}^\dagger$$

$$H_c = \underline{D_c} M \underline{D_c}^\dagger + \underline{B_c}$$

Use its eigensystem

$$H_c \to V_c \Lambda_c V_c^\dagger$$

$$z_c = \sum_{i \le j} \left| v_i^{(c)\dagger} \underline{\Delta_{cij}} v_j^{(c)} \right|^2 - \frac{1}{2} \left\| \underline{\Delta_{cij}} \right\|_2^2$$

Make a "Hamiltonian"

$$\mathbf{X} \xrightarrow{\text{blocks}} X_i$$

$$R_i \,\widetilde{\propto}\, X_i X_i^\dagger$$
$$C_i \,\widetilde{\propto}\, X_i^\dagger X_i$$

$$M = \sum_i \underline{K_i} R_i \underline{K_i}^\dagger + \underline{L_i} C_i \underline{L_i}^\dagger$$

$$H_c = \underline{D_c} M \underline{D_c}^\dagger + \underline{B_c}$$

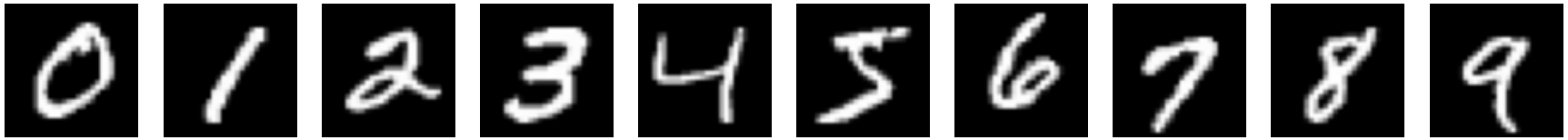
Use its eigensystem

$$H_c \to V_c \Lambda_c V_c^\dagger$$

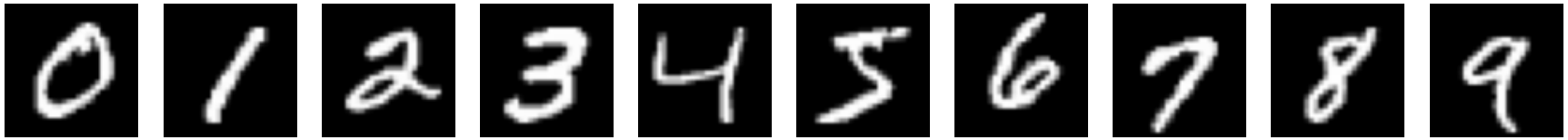
$$z_c = \sum_{i \le j} \left| v_i^{(c)\dagger} \underline{\Delta_{cij}} v_j^{(c)} \right|^2 - \frac{1}{2} \left\| \underline{\Delta_{cij}} \right\|_2^2$$

$$p_c = \frac{\exp\left\{\left(z_c + \underline{b_c}\right)/\tau\right\}}{\sum_k \exp\left\{\left(z_k + \underline{b_k}\right)/\tau\right\}}$$

MNIST Digits

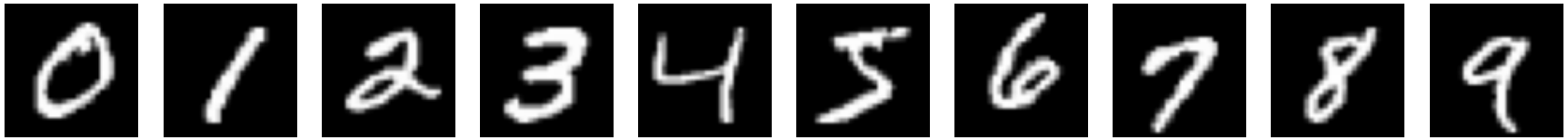


MNIST Digits



DNN-2 96.4% with 5500 parameters

MNIST Digits



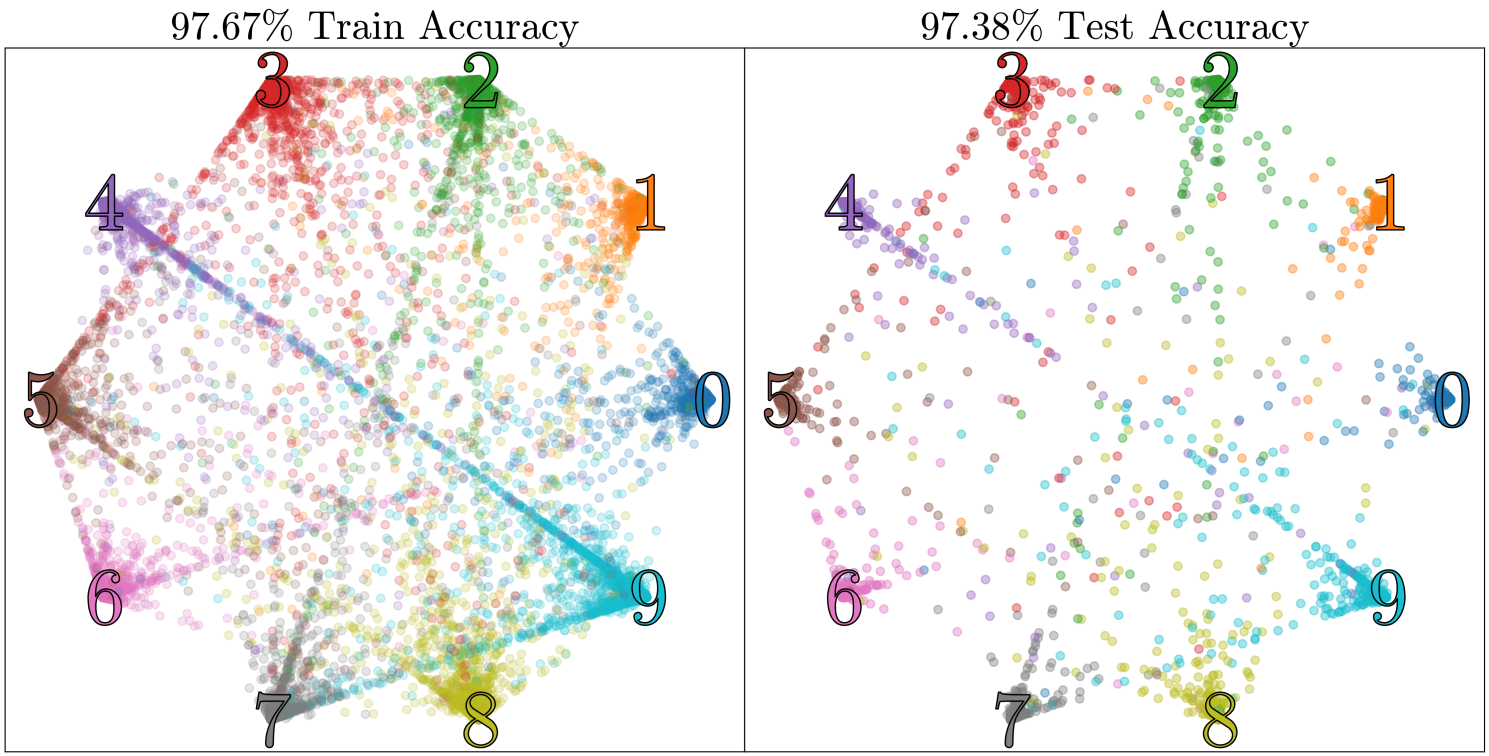
DNN-2 96.4% with 5500 parameters

PMM 97.4% with 2560 parameters

MNIST Digits

DNN-2 96.4% with 5500 parameters

PMM 97.4% with 2560 parameters



Fashion MNIST

Fashion MNIST

GECCO 88.09% with 19 000 parameters

Fashion MNIST

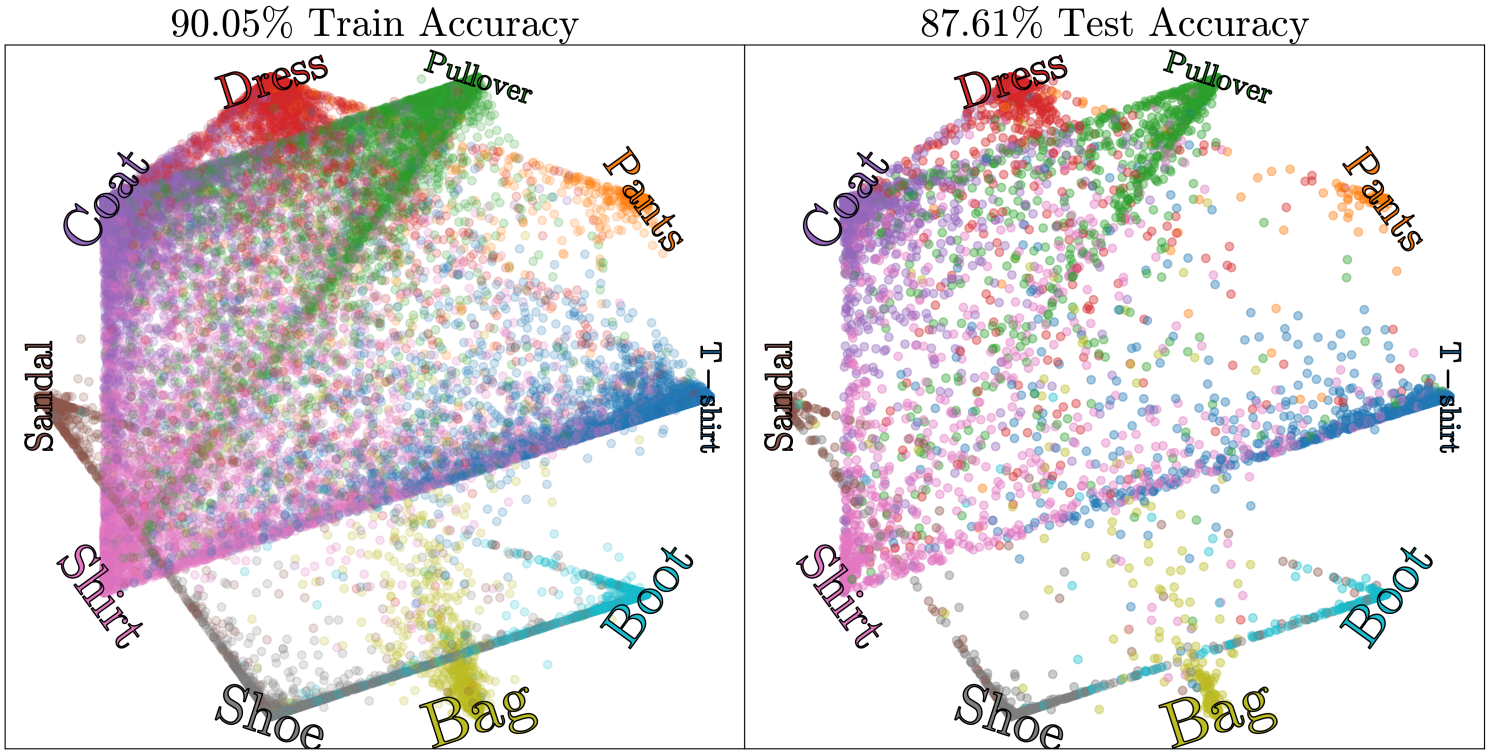
GECCO 88.09% with 19 000 parameters

PMM 87.61% with 4210 parameters

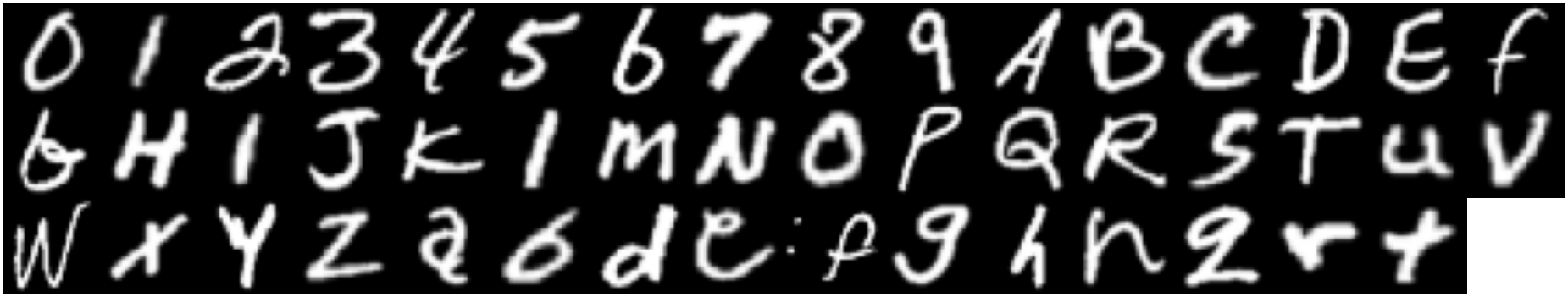
Fashion MNIST

GECCO 88.09% with 19 000 parameters

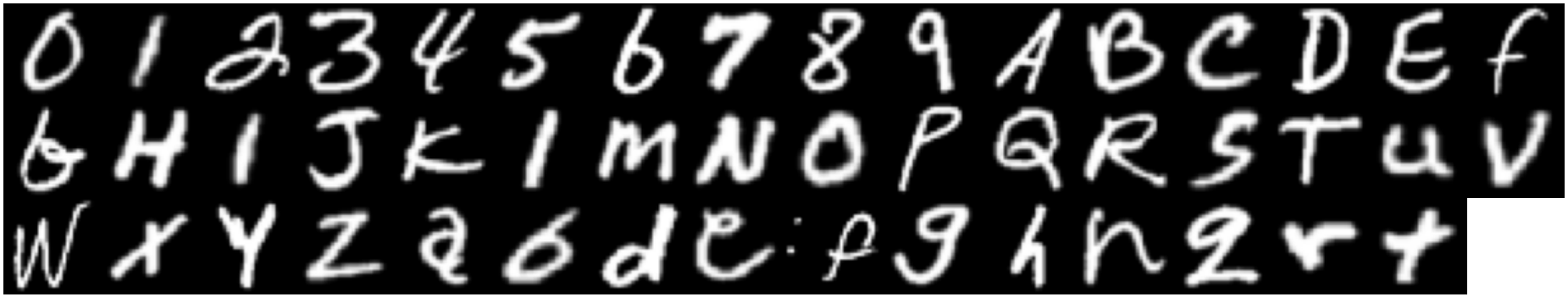
PMM 87.61% with 4210 parameters

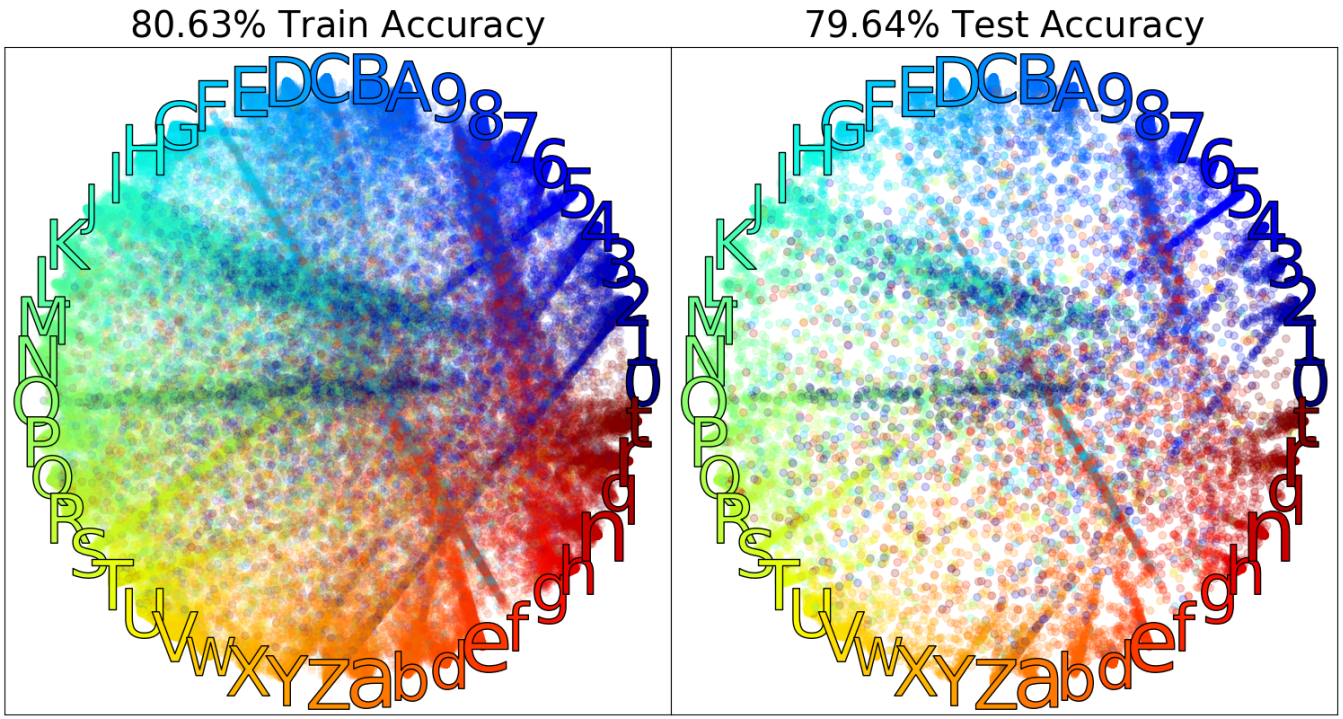


Extended MNIST



Extended MNIST





5567 Trainable Parameters

Conclusions and Future Work

Conclusions and Future Work

- PMMs are parameter-efficient and robust

Conclusions and Future Work

- PMMs are parameter-efficient and robust
- PMMs generalize beyond physics and scientific computing

Conclusions and Future Work

- PMMs are parameter-efficient and robust
- PMMs generalize beyond physics and scientific computing
- PMMs are competitive with state-of-the-art models

Conclusions and Future Work

- PMMs are parameter-efficient and robust
- PMMs generalize beyond physics and scientific computing
- PMMs are competitive with state-of-the-art models

Conclusions and Future Work

- PMMs are parameter-efficient and robust
- PMMs generalize beyond physics and scientific computing
- PMMs are competitive with state-of-the-art models

Various algorithmic improvements

- Leveraging unitary invariance for accelerated training

Conclusions and Future Work

- PMMs are parameter-efficient and robust
- PMMs generalize beyond physics and scientific computing
- PMMs are competitive with state-of-the-art models

Various algorithmic improvements

- Leveraging unitary invariance for accelerated training
- Uncertainty quantification

References

Cohen, Gregory et al. (2017). arXiv: 1702.05373 [cs.CV].

Cook, Patrick et al. (2024). https://arxiv.org/abs/2401.11694. arXiv: 2401.11694 [cs.LG].

Fein-Ashley, Jacob et al. (2024). arXiv: 2402.00564 [cs.CV].

Frame, Dillon et al. (2018). In: *Phys. Rev. Lett.* 121 (3), p. 032501. DOI: 10.1103/PhysRevLett.121.032501.

Hyvärinen, A. and E. Oja (2000). In: *Neural Networks* 13.4, pp. 411–430. DOI: https://doi.org/10.1016/S0893-6080(00)00026-5.

Ji, Xu et al. (2019). arXiv: 1807.06653 [cs.CV].

König, S. et al. (2020). In: *Physics Letters B* 810, p. 135814. DOI: https://doi.org/10.1016/j.physletb.2020.135814.

Krizhevsky, Alex (2009). https://api.semanticscholar.org/CorpusID:18268744.

Lecun, Y. et al. (1998). In: *Proceedings of the IEEE* 86.11, pp. 2278–2324. DOI: 10.1109/5.726791.

Li, Yuwen et al. (2021). arXiv: 2109.09948 [cs.LG].

Maaten, Laurens van der (2009). In: *Journal of Machine Learning Research - Proceedings Track* 5, pp. 384–391.

— (2014). In: *Journal of Machine Learning Research* 15.93, pp. 3221–3245.

Maaten, Laurens van der and Geoffrey Hinton (2008). In: *Journal of Machine Learning Research* 9, pp. 2579–2605.

(N.d.). https://www.theverge.com/2023/9/26/23889956/microsoft-next-generation-nuclear-energy-smr-job-hiring. Accessed: 02-09-2024.

(N.d.). https://paperswithcode.com/. Accessed: 02-09-2024.

(N.d.). https://ourworldindata.org/grapher/artificial-intelligence-parameter-count. Accessed: 05-08-2024.

Pishchik, Evgenii (2023). In: DOI: 10.20944/preprints202301.0463.v1.

Sarkar, Avik and Dean Lee (2021). In: *Phys. Rev. Lett.* 126 (3), p. 032501. DOI: 10.1103/PhysRevLett.126.032501.

Schwarz Schuler, Joao Paulo et al. (2022). In: *MENDEL* 28.1, pp. 23–31. DOI: 10.13164/mendel.2022.1.023.

Xiao, Han et al. (2017). In: *arXiv e-prints*, arXiv:1708.07747, arXiv:1708.07747. DOI: 10.48550/arXiv.1708.07747. arXiv: 1708.07747 [cs.LG].