ML 2025 Project

Gabriele Righi, Edoardo Fiaschi
“Learning rate: 0”
Computer science (AI curriculum)
g.righi7@studenti.unipi.it
e.fiaschi4@studenti.unipi.it

21/01/2026

Type of project: **B**

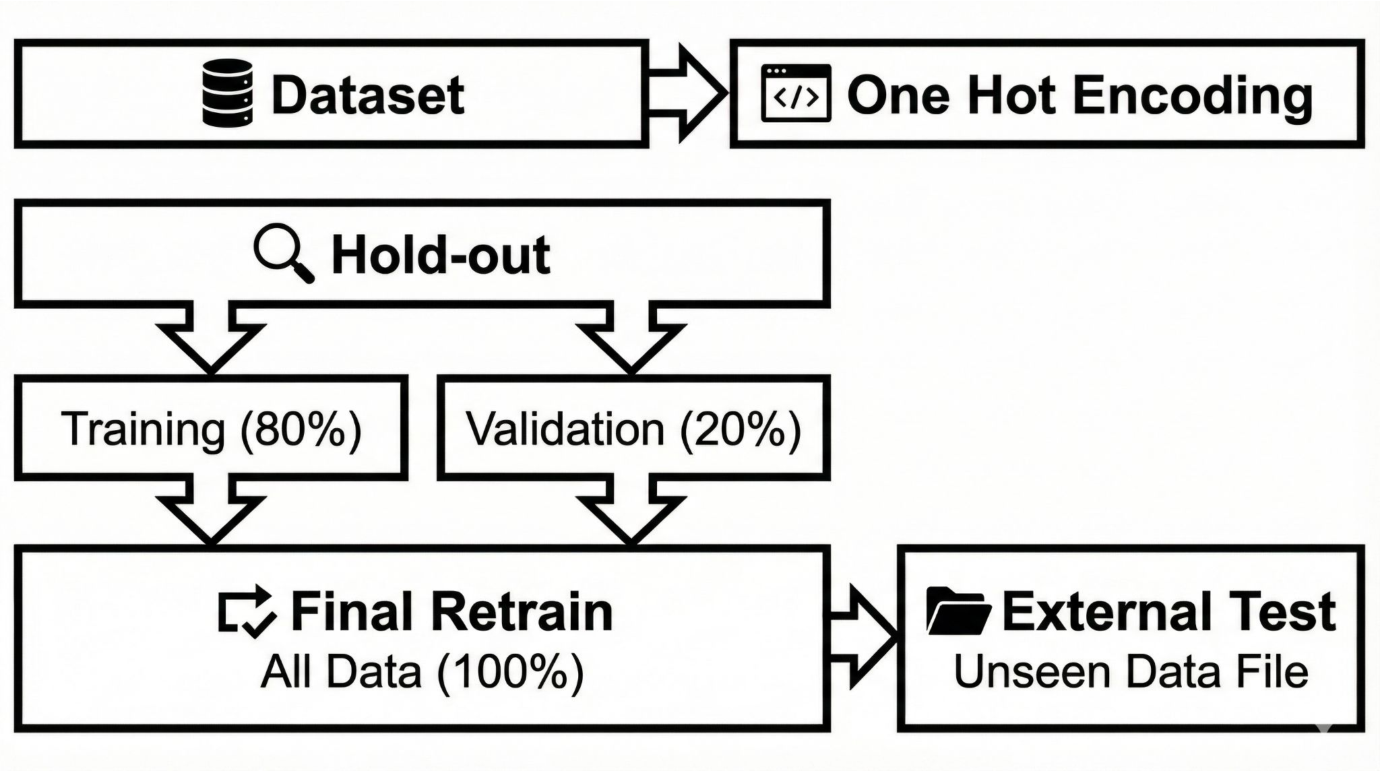
Objectives

Development and Comparative Analysis of Machine Learning Models for Classification and Regression Tasks:

- **MONK Datasets (Classification)**[2]**:**
  - Develop and compare different ML approaches (k-NN, Decision Trees, Neural Networks, SVM) on MONK benchmark problems
  - Evaluate generalization capabilities and robustness
- **ML-CUP 2025 (Regression):**
  - Design and implement regression models for multi-target prediction
  - Optimize hyperparameters through systematic search
  - Apply regularization and validation techniques to ensure generalization
  - Compete on blind test set evaluation

MONK Classification

- Python pipeline: Scikit-learn, TensorFlow/Keras, Pandas, NumPy
- **MONK Classification Models:** MLP, SVM (RBF/Linear), k-NN (k=1-20), Decision Tree, Linear Models



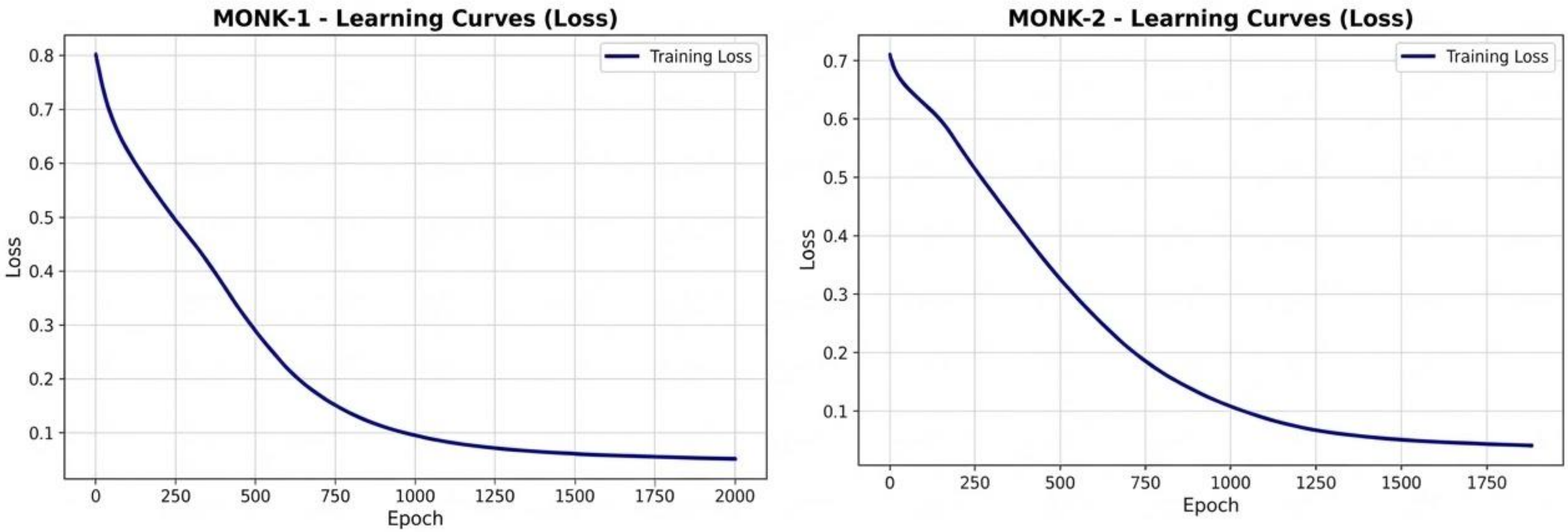
Monk Results 1 (may be more slides, but be schematic)

- Performance (see Table 1) and learning curves (MSE and accuracy plots for the 3 MONK's tasks, see the next slide as an example of schema filled with the plot for the MONK2)

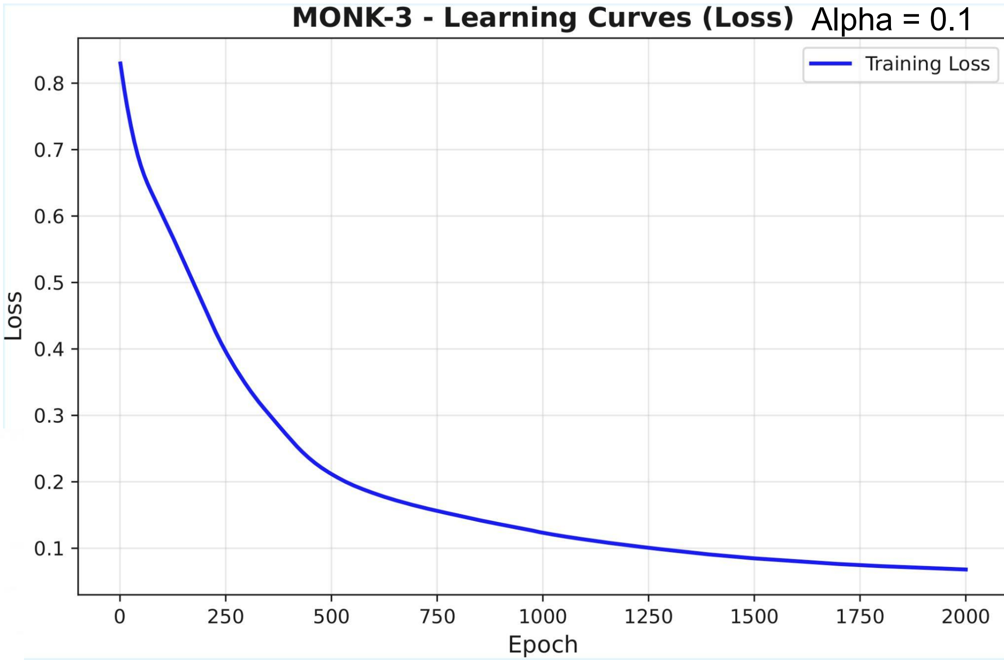
Table 1. Average prediction results obtained for the MONK's tasks.

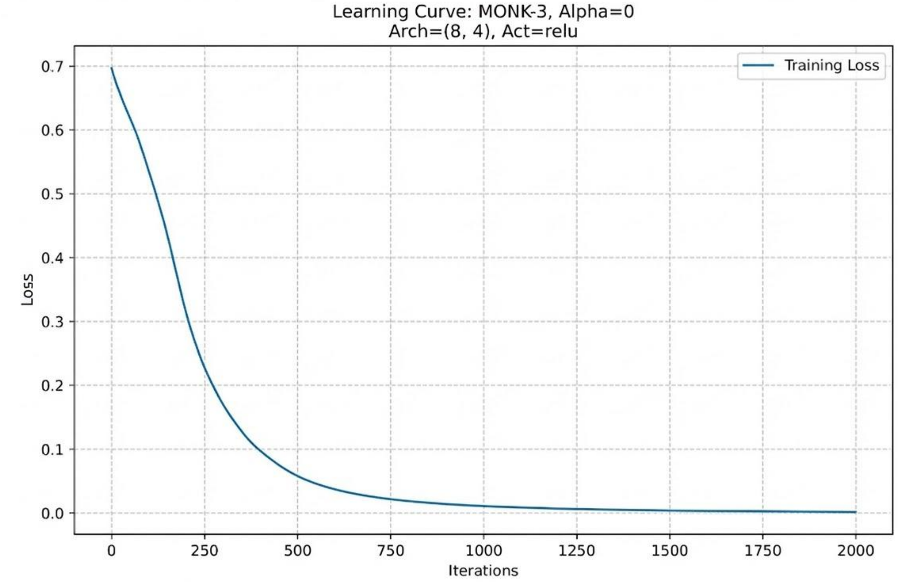
Task	#Units	Eta	Act	Lambda	Mse TR/TS	Accuracy
Monk 1	(8,)	0.01	ReLU	0.001	0.0/0.0301	100%/100%
Monk 2	(4,)	0.01	ReLU	0.001	0.0/0.0	100%/100%
Monk 3	(8,4)	0.01	Relu	0	0.0/0.0833	100%/88%
Monk 3 reg	(8,)	0.01	ReLU	0.1	0.0/0.0602	99%/94.7%

Learning Curves (Loss) for MONK Tasks



Alpha = 0.1


MONK-3 - Learning Curves (Loss)
Training Loss
Loss
Epoch
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
250
500
750
1000
1250
1500
1750
2000


Learning Curve: MONK-3, Alpha=0
Arch=(8, 4), Act=relu
Training Loss
Loss
Iterations
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
250
500
750
1000
1250
1500
1750
2000

SVM RESULTS

DATASET	Hyper Parameters	TRAIN	VALIDATION	TEST
MONK 1	C=10,degree=2, gamma=scale, kernel=poly	100%	96%	100%
MONK 2	C=1,degree=2, gamma=1, kernel=poly	100%	91.18%	100%
MONK 3	C=10, gamma=scale, kernel=rbf	100%	100%	94%

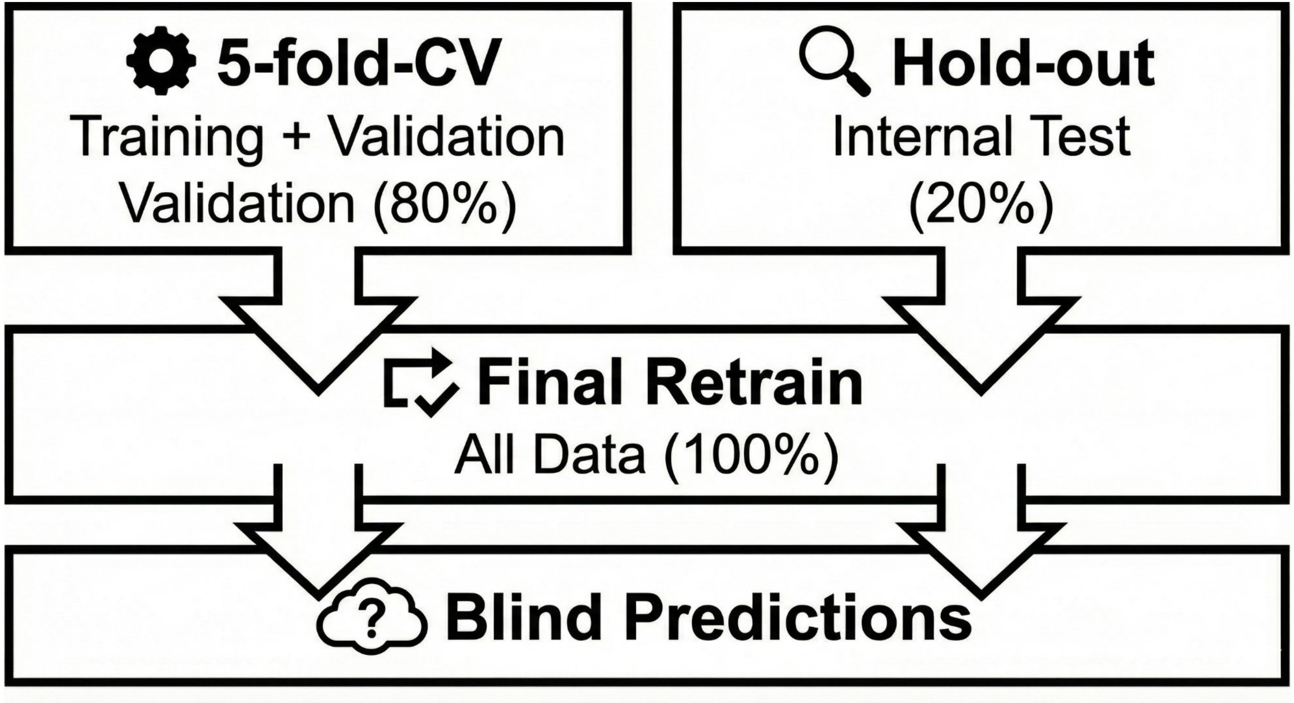
OTHER MODELS

DATASET	Model	TRAIN	VALIDATION	TEST
MONK 1	Decision Tree	100%	84.00%	94.44%
MONK 2	Decision Tree	65.93%	61.76%	64.35%
MONK 3	Decision Tree	92.78%	96.00%	97.22%
MONK 1	Knn	92.74%	88.00%	80.56%
MONK 2	Knn	76.33%	70.59%	61.57%
MONK 3	Knn	90.98%	88.00%	90.05%

*see appendix for hyperparameters

CUP Model Selection & Assessment Schema[1]

- **Model Selection**: 5-Fold CV on Development Set (80%)
- **Model Assessment**: Hold-out Test Set evaluation (20%)
- **Final Retraining**: YES
- **Blind Test**: Final model → predictions on ML-CUP25-TS



CUP Validation schema: model selection

Table 3: Hyperparameter study on various SVR models (coarse-grained results)

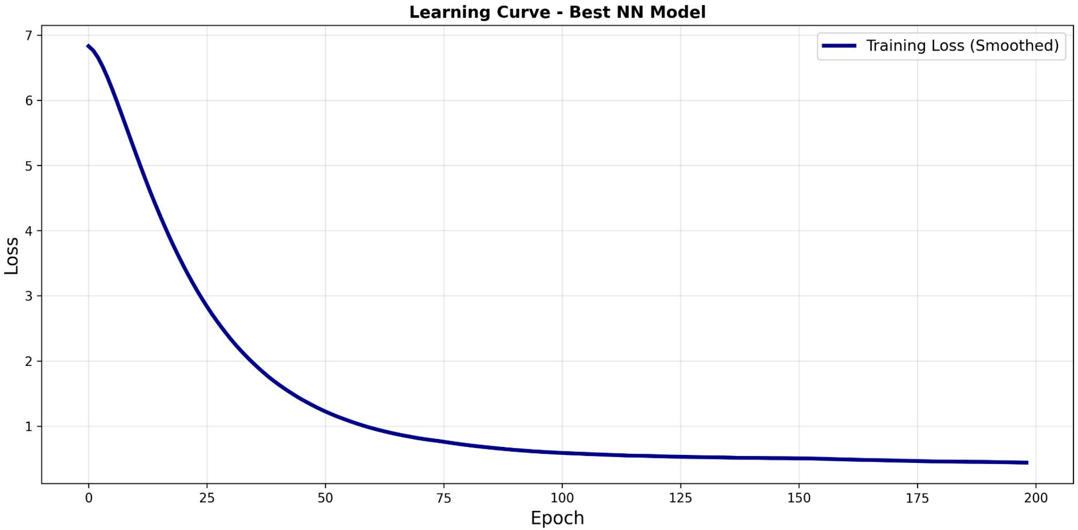
Kernel	C	Epsilon	Gamma	PCA Cmp.	MEE (VL)	MEE (TR)
RBF	39.0	0.54	1.61	5	16.547918	9.855261
LINEAR	1	0.1	\	4	25.8192	25.268
POLY	50	0.1	0.5	6	25.485	22.484

Table 4: Hyperparameter study on Linear Models (coarse-grained results)

Linear Model	Alpha	PCA Cmp.	MEE (VL)	MEE (TR)
Lasso	1.3335214321	5	25.534955	25.374404
ElasticNet (0.1 "l1 ratio")	0.1	10	25.612717	25.233812
Ridge	52.98316906	10	25.587715	25.256472

CUP Validation schema: NN model selection

Model	# units	Eta	Dropout	Lambda	Training MEE	Validation MEE	Test MEE
Neural net	[128, 128, 64, 32]	0.005	0.08	0.001	19.346297	20.604675	21.322351
Neural net with feature engineering	[192, 128, 64]	0.003	0.12	0.008	15.4597	19.3267	19.0503

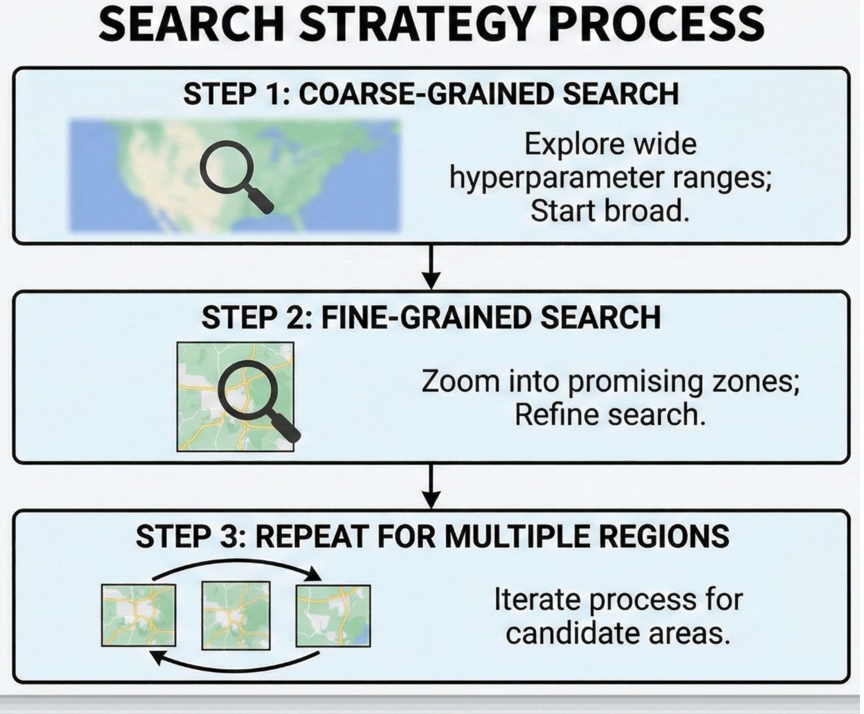


*see the hyperparameter ranges in the appendix for the Gridsearch

OTHER MODELS

MODEL	hyper param.	mee TRAIN	mee VALIDATION	mee TEST
KNN	metric: euclidean n_neighbors: 4 weights: distance	16.1851	16.7402	14.4829
RandomForest	max_depth: 7, n_estimators: 200	20.2016	23.7727	23.2357
XGBOOST	n_estimators: 100 estim_maxdepth: 3	25.8926	27.0717	27.0757
LIGHTGBM	n_estimators: 100	25.6096	26.9505	26.8314

CUP Results strategy



For each candidate model, a **GridSearchCV**[3] was performed, for SVR, Knn and NN.
The best models were the SVR-RBF and GridSearchCV was performed for this model over “PCA-num”, “C”, “ε” and “γ”.

Computational Cost

Coarse-grained search: typically 10–50 minutes, depending on the explored hyperparameter ranges

Fine-grained search: generally shorter for smaller datasets, but still up to ~30 minutes when using very small step sizes

Hardware: experiments were conducted on Google Colab, using both CPU and
T4 GPU

Table SVR Grid Search - Fine and Coarse Grained

Grid Search Type	PCA Cmp	C	Epsilon	Gamma
Coarse-Grained	1 - 6	1.0 – 100.0	0.1 – 1.0	0.01 – 5.0
Fine-Grained	5	38.7 – 39.2	0.53 – 0.55	1.58 – 1.64

CUP Results

The **final model** was selected as the one achieving:

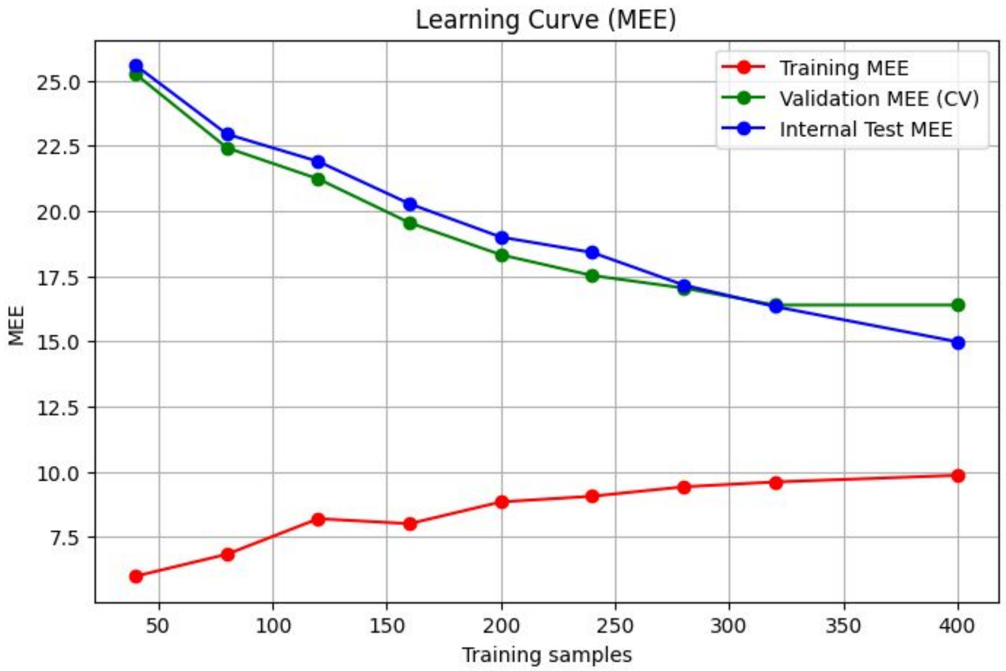
- The **lowest average CV MEE** on the validation folds
- The final model was part of the model found in the GridSearchCV (but in the fine-grained latest GridSearchCV);

After the final model was selected it was retrained on all the Train + Validation and after that tested on the Final Test Set;

Kernel	C	Epsilon	Gamma	PCA Components	MEE (TS)	MEE (VL)	MEE (TR)
RBF	**39.0**	**0.54**	**1.61**	**5**	**14.981123**	**16.547918**	**9.855261**
RBF	39.0	0.52	1.6	5	14.989358	16.555839	9.882236
RBF	38.7	0.52	1.6	5	14.991157	16.555525	9.899068

Those models has been selected based on a fine-grained research on previous candidate models.

Final Model Learning Curve:



Kernel	C	Epsilon	Gamma	PCA Components	MEE (TS)	MEE (VL)	MEE (TR)
RBF	**39.0**	**0.54**	**1.61**	**5**	**14.981123**	**16.547918**	**9.855261**

MONK Discussion

MONK is a **relatively simple classification problem**:

- Small dataset (124-432 samples)
- Low dimensionality (6 categorical features)
- Clear decision boundaries
- **Simple problems** don't require complex models
- Multiple algorithms converge to optimal solution
- Even basic approaches (Decision Trees) perform well
- **M1:** Simple logic → easy for all models (95-100%)
- **M2:** Feature interactions → greedy algorithms fail
- **M3:** Noisy data → regularization needed

Problem structure matters more than dataset size. Greedy methods struggle with feature interactions regardless of dimensionality.

CUP Discussion

Tree ensembles struggle (RF: 23.24, XGBoost: 27.08, LightGBM: 26.83)

- Poor performance suggests smooth, continuous relationships
- Trees partition feature space → not ideal for this data structure

Distance-based models excel (KNN: 14.48, SVR: 14.98)

- Success indicates importance of local similarity patterns
- Smooth decision boundaries favor kernel-based approaches

Linear models (Lasso: 25.53, ElasticNet: 25.61, Ridge: 25.59)

- Differences between Lasso, ElasticNet, and Ridge are marginal
- Unable to match previous models in absolute MEE

Neural Networks:

- Base NN [128,128,64,32]: 21.32 MEE
- **With feature engineering** [192,128,64]: **19.05 MEE** (-11%)
- Gap shows: feature engineering > deep architectures for this task

Conclusions

MONK Dataset: Taught us that complexity hides in structure, not size. MONK-2's feature interactions defeated greedy methods despite having only 6 features.

ML CUP Dataset: Showed us that sophisticated ≠ better. Simple kernel methods beat ensemble giants because they matched the data's smooth structure.

Core insight: Stop chasing the "best" algorithm. Start understanding the data.

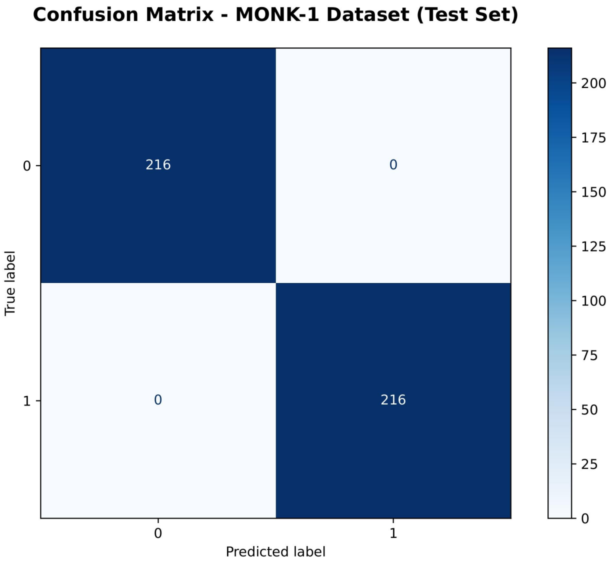
References

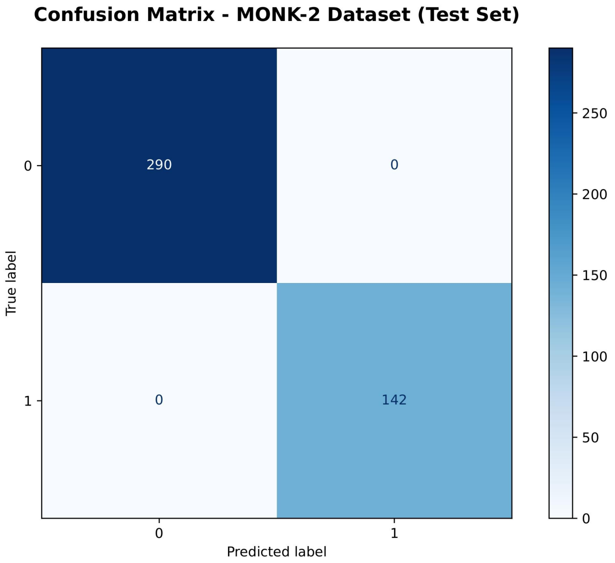
[1] Simon Haykin. Neural Networks and Learning Machines. Pearson, Upper Saddle River, NJ, 3rd edition, 2009

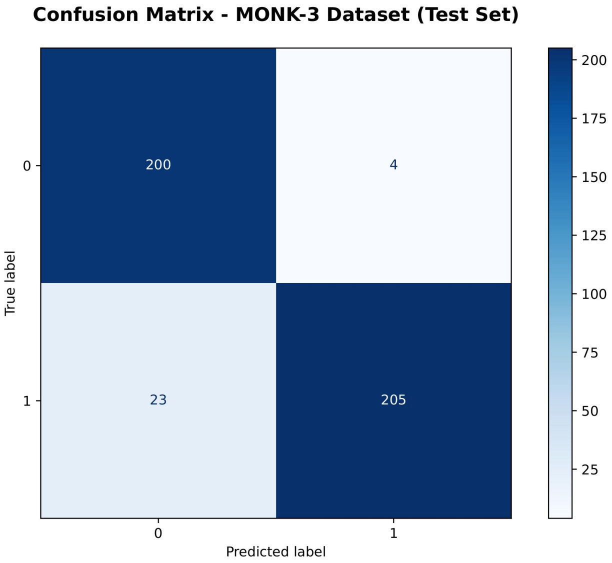
[2] Sebastian B Thrun, J Bala, E Bloedorn, I Bratko, B Cestnik, J Cheng, K De Jong, S Dzeroski, SE Fahlman, D Fisher, et al. The monk's problems: A performance comparison of different learning algorithms. Technical Report CMU-CS-91-197, Carnegie Mellon University, Pittsburgh, PA, 1991.

[3] Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb), 281-305.

Appendix - SVM confusion matrices







Appendix: Hyperparametrs - Monk

KNN

MONK-1
- Best Method: KNN-Hamming
- Optimal k: 3

MONK-2
- Best Method: Modified-KNN
- Optimal k: 3

MONK-3
- Best Method: KNN-Hamming
- Optimal k: 14

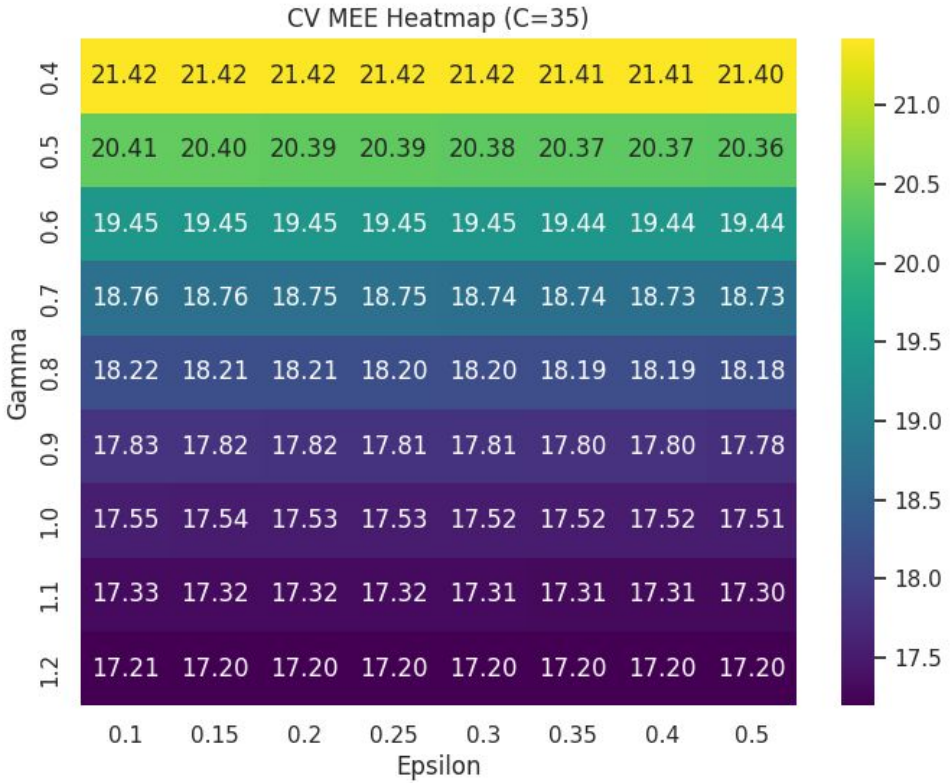
Decision tree

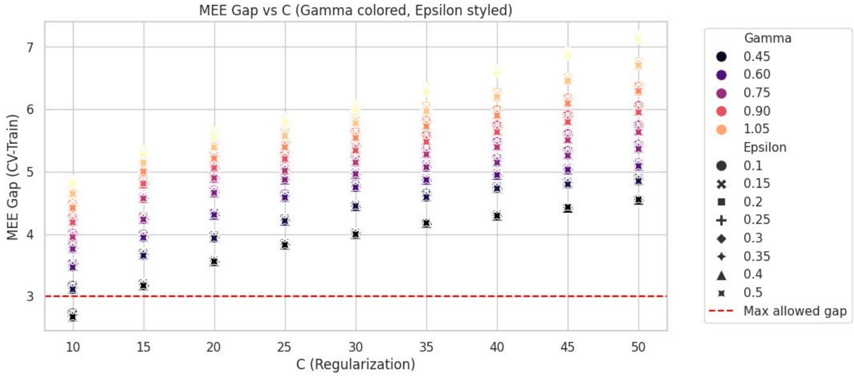
MONK-1
- criterion: gini
- max_depth: None
- min_samples_leaf: 1
- min_samples_split: 2

MONK-2
- Best Parameters:
  - criterion: gini
  - max_depth: 3
  - min_samples_leaf: 2
  - min_samples_split: 5

MONK-3
- criterion: gini
- max_depth: 3
- min_samples_leaf: 2
- min_samples_split: 5

Appendix - Hyperparameters research:





Appendix - SVR and Linear Models:

Search Strategy:

1. Start **coarse-grained** → explore wide hyperparameter ranges;
2. Zoom into **promising zones** → fine-grained search;
3. Repeated for **multiple candidate region;**

SVR models Tested:

- **Best model:** SVR-RBF;
- **Other evaluated models:** LinearSVR, SVR-linear;

Linear Models Tested:

Linear models performed significantly worse than the best SVR-RBF configurations previously listed;

- Linear Regression;
- Ridge Regression;
- Lasso Regression;
- ElasticNet Regression;

Appendix: CUP Validation schema: Other Models

Table 4: SVR models bagging:

Kernel	C	Epsilon	Gamma	PCA Cmp.	Degree	MEE (VL)	MEE (TR)
RBF-bag	32	0.55	0.9	5	\	18.4107	14.4182
LINEAR-bag	28	0.55	\	5	25	26.161	25.197
Poly-bag	28	0.55	3.0	5	25	25.4786	23.092

Appendix: hyperparameters range

table: Hyperparameter range for NN

# units	Eta	Dropout	Lambda
[256, 128, 64], [192, 128, 64], [256, 192, 96], [192, 128, 64, 32], [256, 128, 64, 32], [160, 128, 96, 48], [384, 192, 64], [320, 160, 80], [256, 256, 128], [224, 128, 64]	0.001-0.01	0.1-0.25	0.0001 - 0.02

table: Hyperparameter range for KNN

n_neighbors	knn__weights	knn__metric
3-10	uniform, distance	euclidean, manhattan

Appendix: hyperparameters range

table: Hyperparameter range for tree models

Model	**hyperparameter**	**range**
Random Forest	n_estimators	[200]
	max_depth	[3, 5, 7]
	min_samples_split	[20, 50]
	min_samples_leaf	[10, 20]
	max_features	[0.3, 0.5]
XGBoost, LightGBM	n_estimators	[50, 100]
	learning_rate	[0.01]
	max_depth	[2, 8]
	min_child_weight	[20, 50]
	reg_lambda (L2)	[10.0, 20.0]
	subsample	[0.5, 0.6]