CLASSIFYING LEARNERS BASED ON LEARNING CURVES

Sinan Basaran Karaarslan - S.BasaranKaraarslan@student.tudelft.nl

Introduction

Learning curves can help guide important decisions in machine learning projects:

- data acquisition
- early stopping
- model selection [1]

Currently, parametric formulas are used to extrapolate learning curves.

While this works for many curves, some curves are "ill-behaving" and show unexpected behaviors [2].

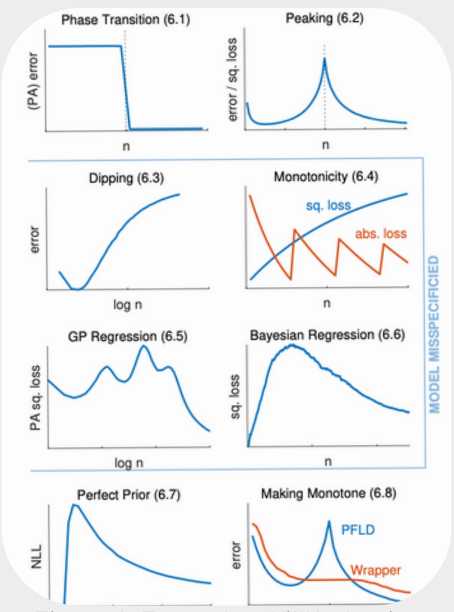

Figure 1: Example of 'ill' behavior

Studies suggest that deep learning models can extrapolate curves more effectively [3].

Research Questions

We can get further insights into learning curves by seeing whether machine learning models can effectively classify learners based on learning curves.

- **RQ1:** Which learners can be reliably classified by their learning curves?
- **RQ2:** Under which conditions does the classification accuracy degrade or improve?
- **RQ3:** Which machine learning model is most suitable for this classification task?

Motivation

Improving our understanding of learning curves through the process of classification:

- Help develop current deep learning extrapolation methods, specifically the **LC-PFN** [3]
- Make inferences on when and why the LC-PFN underperforms or outperforms parametric models
- See whether classification inversely correlates with the domain shift effects on the LC-PFN

Findings

For our experiments we treat learning curves as time series.

How:

- by substituting the time axis of time series with the training set size
- using the fact that all of our learning curves are aligned at each x-axis step

Why:

- take advantage of the sequential nature of learning curves
- apply the many available time series classification (TSC) models

First experiment: training a binary classifier on all pairs of learners.
This will help us figure out:

- which learners are easily distinguished from each other
- which learners curves are similar enough to be grouped together

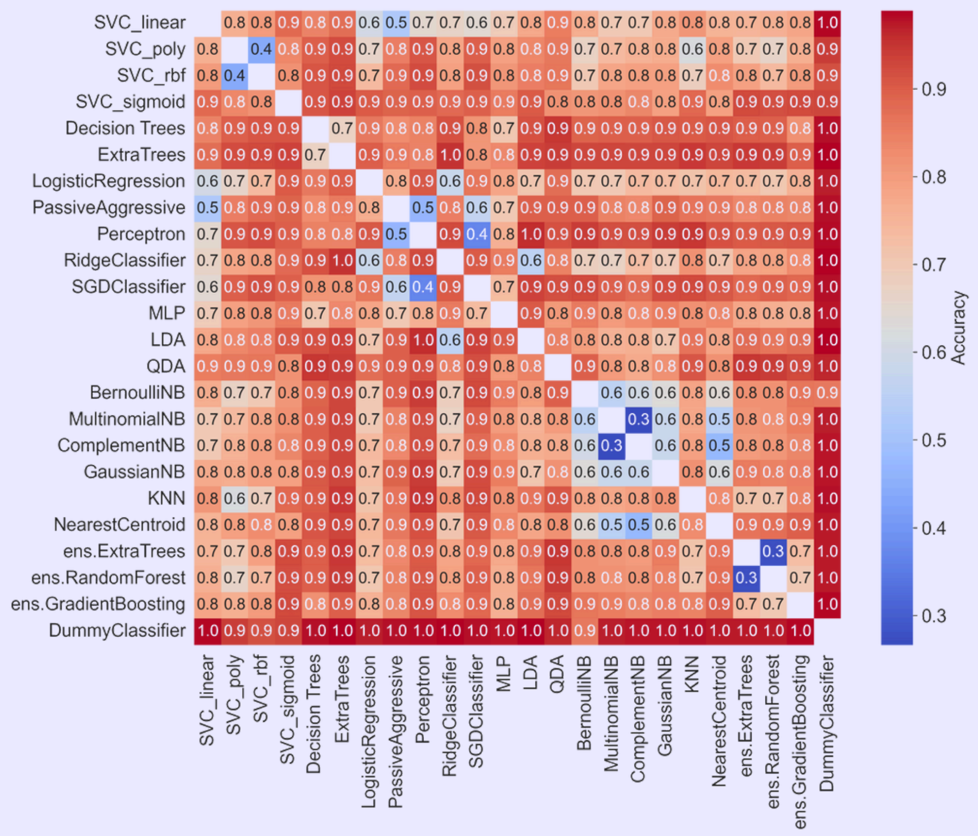

Figure 2: Pairwise Binary Classification Acc. Matrix

Second experiment: training classifiers on different minimum length cutoffs for curves.
This will help us figure out:

- at which anchor points do learning curves start showing distinct behaviors
- how to best preprocess learning curve data for classification

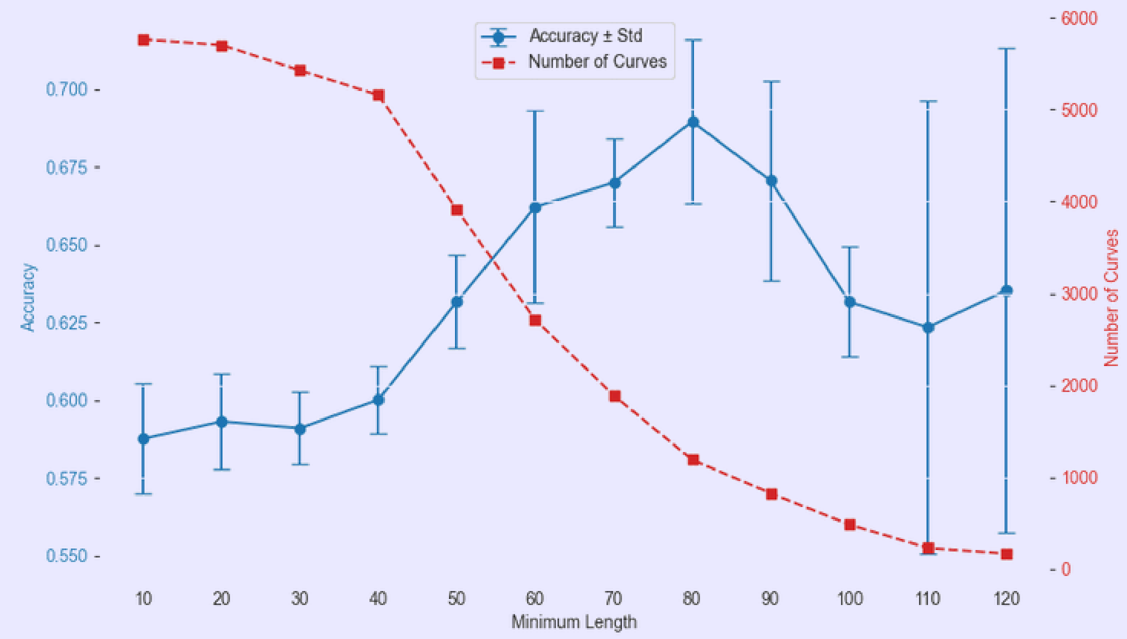

Figure 3: Accuracy of a Classifier vs. Minimum Length Cutoff of Learning Curves

Third experiment: training a classifier on various combinations of types of curves (train, test, and validation)
This will help us figure out:

- if grouping different types of curves help with classification
- which curves contain the most information to classify by

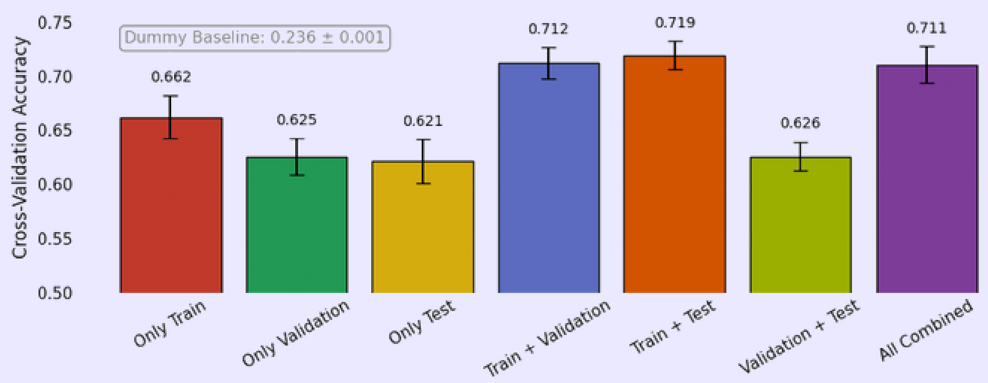

Figure 4: Classification Performance of Various Combinations of Curves

Fourth experiment: comparing the classification performance of various time series classification (TSC) models
This will help us figure out:

- what types of machine learning models are most effective
- what this suggests about the general structure of learning curves

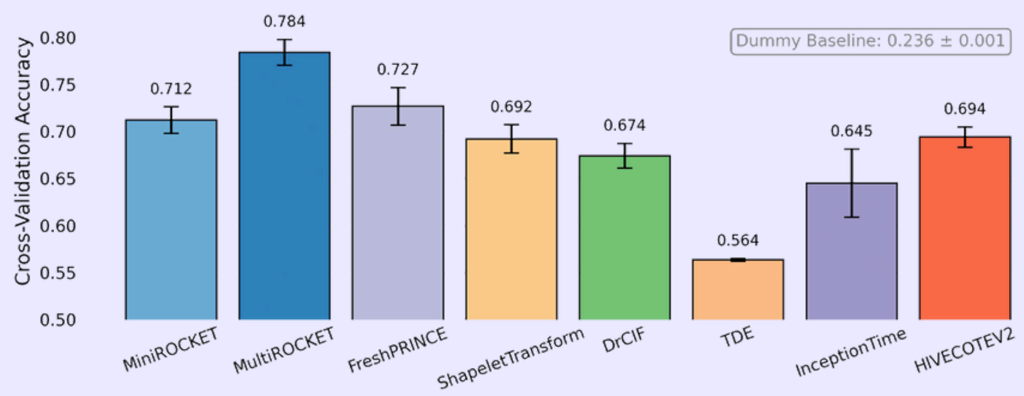

Figure 5: Classification Performance of Various TSC Models

Observations

Figure 2 suggests:

- Some learners produce easily-distinguishable, unique curves (e.g. DummyClassifier, SVC_sigmoid, QDA)
- Some learners produce very similar curves to one another
  - these similarities are sometimes intuitive (e.g. different NB classifiers)
  - and sometimes unexpected (e.g. SVC_linear and PassiveAggressive)

Figure 3 suggests:

- Eliminating short learning curves may be beneficial for clarification
- However, eliminating too much leads to higher variance and lower accuracy.
- Minimum length of 50-90 seems to be a reasonable cutoff point

Figure 4 suggests:

- Test and validation curves hold similar information across learners
- Train curves are more distinguishable compared to the other two
- Combining train curves with either or both validation and test curves leads to the best performance

Figure 5 suggests:

- Feature-based models (MultiRocket, MiniRocket, and FreshPRINCE) perform best across all TSC models
- This suggests that the broader structure of curves, such as slope, variability and trends are the most distinguishable part of them

Conclusion

RQ1: Which learners can be reliably classified by their learning curves?

- Easily distinguishable: *QDA, DummyClassifier, SVC_sigmoid*
- Similar: *DecisionTree* and *ExtraTrees*, *SVC_poly* and *SVC_rbf*, *Naive Bayes variants*, and more

RQ2: Under which conditions does the classification accuracy degrade or improve?

- Longer curves: more information, but elimination of data leads to high variance
- Train curves most distinguishable, combining with either or both of validation and test curves leads to the best performance

RQ3: Which machine learning model is most suitable for this classification task?

- Feature-based models work best
- The broader structure of curves, such as their slope, variability, and long-term trends are what make them distinguishable

References

[1] Felix Mohr and Jan N. van Rijn. Learning curves for decision making in supervised machine learning: a survey. Machine Learning, 113(11â12):8371â8425, December 2024.
[2] Tom J. Viering and Marco Loog. The shape of learning curves: a review. CoRR, abs/2103.10948, 2021.
[3] Tom Julian Viering, Steven Adriaensen, Herilalaina Rakotoarison, and Frank Hutter. From epoch to sample size: Developing new data-driven priors for learning curve prior-fitted networks. In AutoML Conference 2024 (Workshop Track), 2024.