DATA MINING (16:958:588:01)

Project Report on Paper : LassoNet: A Neural Network with Feature Sparsity

Team 13: Phan Nguyen Huong Le   Anton Vernikov

Alvin Alex   Annshu Prajapati

ENHANCING FEATURE SELECTION IN NEURAL NETWORKS:

AN EVALUATION OF LASSONET ON THE MNIST DATASET

I. Introduction & Problem Description

The paper introduces LassoNet, a neural network framework that incorporates feature selection directly into the architecture of the network. The main goal of the authors was to extend the Lasso regression's feature sparsity capabilities to neural networks. Lasso regression is commonly used in linear models for feature selection by assigning zero weights to the least important features. However, traditional Lasso cannot be directly applied to neural networks, which are inherently non-linear.

The problem they aim to solve is the lack of an efficient and integrated feature selection mechanism within neural networks. Feature selection is crucial in many machine learning applications to reduce model complexity, improve interpretability, and enhance performance by eliminating irrelevant or redundant features. By embedding feature selection directly into the network architecture, LassoNet aims to provide a method that is both effective in handling high-dimensional data and capable of capturing complex patterns in the data.

To achieve this, LassoNet employs a unique architecture that includes a skip-layer connection (similar to residual connections in ResNets) which allows it to control feature sparsity across the network. This method combines the advantages of linear component guidance in feature selection with the power of deep learning for handling non-linear relationships. The approach not only simplifies the integration of feature selection but also optimizes the neural network training process to achieve better feature selection and classification performance.

II. LassoNet Algorithms

As introduced above, lassonet integrates Lasso (L1) regression with neural networks to address the limitations of Lasso with linear models. By combining these approaches, LassoNet assigns zero weights to the most irrelevant features and extends this capability to feed-forward neural networks, allowing the model to capture both linear and non-linear relationships in the data.

LassoNet is stemmed from the hierarchy principle, which maintains that linear components, such as those in traditional Lasso, should be prominent, allowing the model to manage the regularization effectively. The principle helps tremendously in addressing issues of data redundancy and overfitting.

2.1 Mathematical Formulation

The objective function of LassoNet combines the loss function with neural networks with an L1 regularization term. This regularization is applied not just across the entire model but specifically integrates a hierarchical constraint between linear and non-linear model components. In LassoNet we consider the class of all fully connected feed-forward residual neural networks, abbreviated as

$$\mathcal{F} = \left\{ f \equiv f_{\theta,W} : x \mapsto \theta^T \mathbf{x} + g_W(\mathbf{x}) \right\},$$

Notation: W denotes the network parameters, K denotes the size of the first hidden layer, W(θ) ∈ RdxK denotes the first hidden layer, x ∈ Rd denotes an input data point, and $L(\theta, W) = \frac{1}{n}\sum_{i=1}^{n} \ell(\mathbf{x_i}, y_i; \theta, W)$ denotes the loss on the training data set.

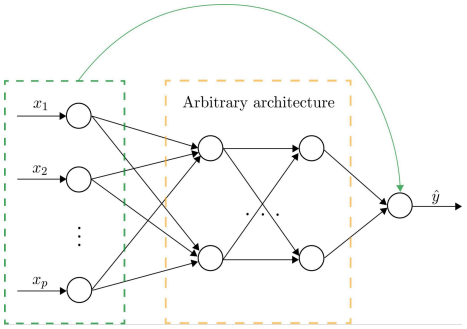
The LassoNet objective function is formally defined as

$$\begin{aligned} &\underset{\theta,W}{\text{minimize}} \;\; L(\theta, W) + \lambda \|\theta\|_1 \\ &\text{subject to} \;\; \left\|W_j^{(1)}\right\|_\infty \le M|\theta_j|, \; j = 1, \ldots, d. \end{aligned}$$

where λ is a regularization strength parameter and M is a hierarchy coefficient. The constraint $|W_{jk}^{(1)}| \le M \cdot |\theta_j|, \; k = 1, \ldots, K$ budgets the total amount of non-linearity involving variable j according to the relative importance of Xj as a linear variable.

It follows that Wj = 0 as soon as θj= 0. In other words, variable j is completely inactive from the model, without the need for an explicit penalty on W.

The hyper-parameters consist of the hierarchy coefficient, denoted as M, which regulates the sparsity level within the initial hidden layer, and the regularization strength, represented by λ, which maintains a balance between the loss function and the L1 penalty.



The architecture of LassoNet consists of a single residual connection, shown in green, and an arbitrary feed-forward neural network, shown in black. The residual layer and the first hidden layer are optimized jointly using a hierarchical operator.

2.2 Training LassoNet

LassoNet employs a gradient descent optimization approach, utilizing a stochastic rate of descent. This allows for adaptive feature selection during the training process, where features are dynamically included or excluded based on their weights and contributions to the model's performance.

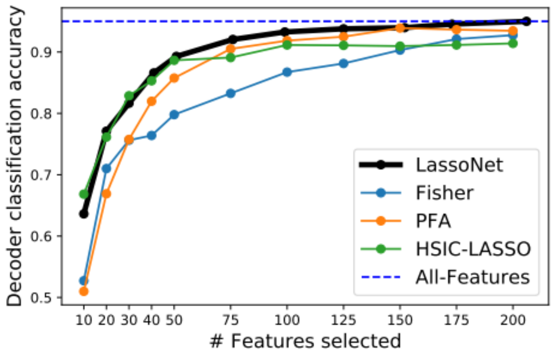
Unlike traditional methods that may require separate phrases or multiple models to evaluate feature importance, LassoNet integrates feature selection directly within the training loop. The integration is facilitated by a proximal operator, which effectively updates the model and feature selection simultaneously, thereby reducing the computational overhead. Utilizing LassoNet for feature selection on datasets like MNIST can lead to shorter training times, reduced model overfitting, and enhanced interpretability. Additionally, it helps with managing high-dimensional data efficiently, which can consequently improve model accuracy and robustness.

Compared to other feature selection methods, LassoNet is categorized under embedded feature selection techniques, sharing similarities methods like random forest importance but uniquely applying L1 regularization to enforce sparsity more explicitly and hierarchically.

In all LassoNet experiments, an Adam optimizer with a learning rate of 10−3 was utilized to train the initial dense model. Subsequently, a vanilla gradient descent with a momentum of 0.9 was employed on the regularization path. In the case of LassoNet, a one-hidden-layer feed-forward neural network with a ReLU activation function was implemented. Additionally, for the matrix completion problems, a two-hidden-layer network was introduced. The number of neurons in the hidden layer varied within the range of [d/3, 2d/3, d, 4d/3], where d represents the total number of features. The network with the highest validation accuracy was chosen and evaluated on the test set. An early stopping criterion of 10 was applied. While the hierarchy parameter could potentially be chosen based on a validation set, it was found that the default value M = 10 yielded satisfactory results across a range of datasets.

III. Application of LassoNet on MNIST and MNIST Fashion dataset

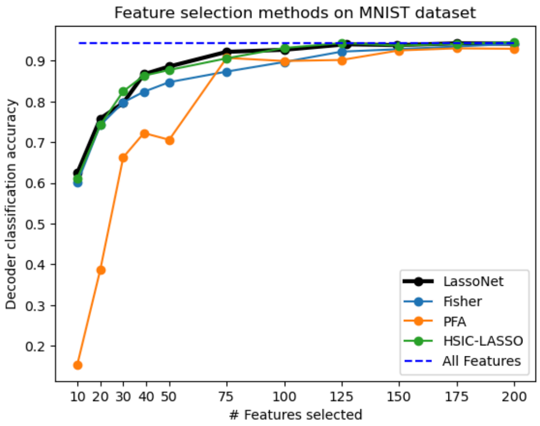
LassoNet performs a step of vanilla gradient descent and subsequently solves a constrained minimization problem. Since the problem is decomposable across features, each iteration of the algorithm decouples into d single-feature optimization problems. The paper is able to use this to track how well the feature selection works at different levels of k features, comparing the results training the same model on different levels of K features versus other feature selection methods, Fisher Score, Principal Feature Analysis, and HSIC Lasso on the ISOLET dataset. As we can see from the chart below, regardless of the k features selected, LassoNet performed the best in essentially every single iteration, regardless of features selected, which suggests that it is the best model regardless of the number of features being looked for.

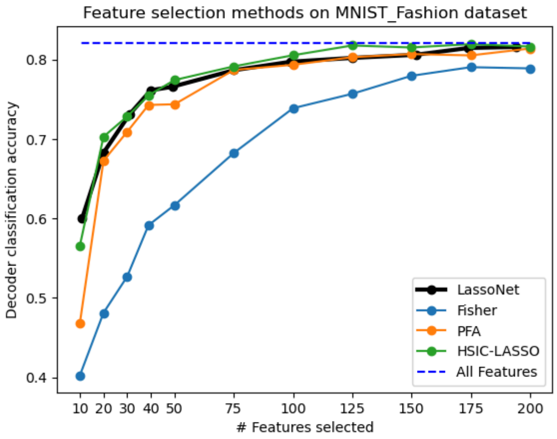


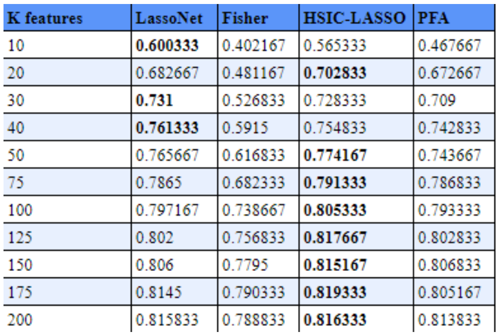
For our empirical evaluation of the methods, we looked to recreate this comparison using two other datasets used in the paper, MNIST and MNIST-Fashion, which consist of 28x28 grayscale images of handwritten digits and clothing items, respectively. Comparing these datasets to ISOLET, they are both even larger (n) compared to ISOLET, and contain a significant amount of features (d) as well, so we felt these two datasets should work well as a comparison.

Dataset	(n, d)	# Classes
MNIST	(10000, 784)	10
MNIST-Fashion	(10000, 784)	10
ISOLET	(7797, 617)	26

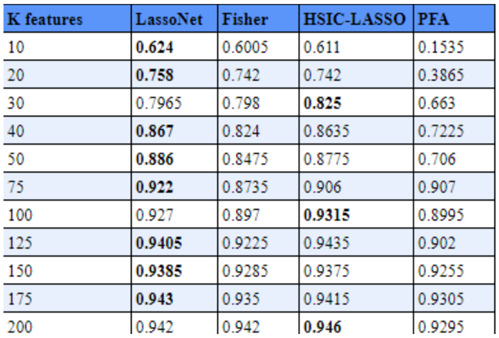
To get the same results, we first trained LassoNet using both datasets, and then looked through each iteration of the neural network to find the iterations that had the k number of features we were looking for. We used the selected features for each of these iterations, and trained a 1 hidden layer neural network with those features, and classifying a test set to get the accuracy for each k number of features. For the other feature selection methods, we had to run them 11 times, for each k number of features we were looking to select, and then trained another 1 hidden layer neural network using the features they found to get those selection method's accuracy as well. Looking at the results in the charts and tables below, we find that LassoNet does get outperformed slightly more in these two datasets by HSIC-LASSO compared to that of ISOLET, but for most k features, it is only marginally, and still LassoNet consistently ranks as one of the 2 best feature selection methods.







K features	LassoNet	Fisher	HSIC-LASSO	PFA
10	**0.600333**	0.402167	0.565333	0.467667
20	0.682667	0.481167	**0.702833**	0.672667
30	**0.731**	0.526833	0.728333	0.709
40	**0.761333**	0.5915	0.754833	0.742833
50	0.765667	0.616833	**0.774167**	0.743667
75	0.7865	0.682333	**0.791333**	0.786833
100	0.797167	0.738667	**0.805333**	0.793333
125	0.802	0.756833	**0.817667**	0.802833
150	0.806	0.7795	**0.815167**	0.806833
175	0.8145	0.790333	**0.819333**	0.805167
200	0.815833	0.788833	**0.816333**	0.813833



K features	LassoNet	Fisher	HSIC-LASSO	PFA
10	**0.624**	0.6005	0.611	0.1535
20	**0.758**	0.742	0.742	0.3865
30	0.7965	0.798	**0.825**	0.663
40	**0.867**	0.824	0.8635	0.7225
50	**0.886**	0.8475	0.8775	0.706
75	**0.922**	0.8735	0.906	0.907
100	0.927	0.897	**0.9315**	0.8995
125	**0.9405**	0.9225	0.9435	0.902
150	**0.9385**	0.9285	0.9375	0.9255
175	**0.943**	0.935	0.9415	0.9305
200	0.942	0.942	**0.946**	0.9295

IV. Discussion

4.1 Advantages of Lassonet :

- LassoNet is a data-driven feature selection method that provides a path of regularized models at the cost of training a single model, making it cost-effective.
- It involves a nonconvex optimization problem with hierarchy constraints to ensure feature sparsity, and uses proximal gradient descent to solve it iteratively.
- LassoNet is able to efficiently explore and converge over an entire regularization path with varying numbers of input features, setting it apart from other feature selection methods.
- LassoNet can extend to other tasks, such as unsupervised reconstruction and matrix completion.
- Implementing LassoNet in popular machine learning frameworks requires minimal code modification and has a runtime similar to training a single model, which improves with hardware acceleration and parallelization techniques.
- The only additional hyperparameter of LassoNet is the hierarchy coefficient, with the default value working well for a variety of datasets.

4.2 Disadvantages of Lassonet:

- LassoNet, does not provide p-values or statistical quantification for the features selected, requiring validation with hypothesis testing or additional analysis using domain knowledge.
- While LassoNet can output varying number of input features, when a desired number of features is already selected, it may perform significantly slower than other feature selection methods, due to requirement of training of a Neural Network to do so

V. Conclusion

In conclusion, LassoNet offers a promising solution to feature selection challenges in high-dimensional data by integrating Lasso regularization directly into neural network architectures. This approach improves model interpretability, reduces computational overhead, and enhances performance in handling complex data relationships. The mathematical formulation and training process of LassoNet enables dynamic feature selection, leading to shorter training times, reduced overfitting, and improved accuracy. Empirical evaluations on datasets like MNIST and MNIST-Fashion demonstrate LassoNet's competitive performance against other methods. Overall, LassoNet contributes significantly to feature selection in machine learning, providing an efficient and principled approach that can be further explored in real-world scenarios.