IE506: Machine Learning - Principles and Techniques **Jan-May 2025**

Final Review Report: Multi-class Feature Selection via Sparse Softmax with a Discriminative Regularization

Team Name: Outliers *Team Members: 24M1503, 24M1511*

1 Problem Statement

In today's world, many real-world machine learning applications such as text categorization, face recognition, handwritten digit recognition, and gene detection involve the use of high-dimensional data. These datasets suffer from the "curse of dimensionality," as they often contain many features that are irrelevant or redundant, leading to increased computational complexity and degraded model performance.

In this project, we tackle the problem of multi-class feature selection (MFS), where the goal is to select the most relevant and discriminative features that enhances classification accuracy across multiple classes.

The project addresses multi-class feature selection (MFS) by implementing a novel framework: the Sparse Softmax Feature Selection (S2FS) method. S2FS combines Softmax classification with L2,0-norm regularization to enforce exact sparsity and adds a discriminative regularization based on Linear Discriminant Analysis (LDA). Together, these elements allow for selecting highly discriminative and uncorrelated features, leading to improved multi-class classification performance.

2 Work Done Before Stage-1 Review

In the period leading up to the Stage-1 review, we focused on building a strong understanding of the assigned paper and understanding the algorithms involved for its implementation. The following key steps outline the methods and approaches tried during this phase:

- Conducted a thorough study of the paper to understand the underlying motivation, mathematical formulation, optimization algorithm, and experimental setup.
- We also read two related works which provided context on prior state-of-the-art methods.
- Analyzed the integration of $\ell_{2,0}$-norm regularization with the Softmax function in S2FS, and how the addition of discriminative regularization overcomes the limitations of prior methods based on simpler loss functions.
- We studied the use of auxiliary variables and the design of an efficient optimization algorithm using the Alternating Direction Method of Multipliers (ADMM).
- The datasets mentioned for the experimental setup were explored. As a first step towards implementation, we attempted to recreate parts of the S2FS algorithm with the help of AI coding tools.

3 Comments/Inputs given during the Stage-1 Review

During the Stage-1 review, the following comments and suggestions were provided by the instructor and TAs:

- As no code was provided in the original paper, we were asked to implement the entire S2FS algorithm from scratch, including all key components such as the Softmax model with $\ell_{2,0}$-norm regularization, the discriminative regularization term, and the ADMM-based optimization.
- Our initial implementation produced suboptimal results. We were instructed to perform hyperparameter tuning using cross-validation techniques to improve model performance.
- We were asked to implement the S2FS method on 5 datasets out of the 15 datasets used in the original paper.

3.1 Addressing Comments after Stage-1 Review

The S2FS method was coded from scratch in Python, ensuring that all algorithms were correctly implemented. To address the issue of suboptimal results, hyperparameter tuning was performed. We experimented with a range of values for the sparsity and regularization parameters, following the tuning ranges specified in the experimental setup of the original paper. We selected five datasets for evaluation: *Madelon*, *MNIST*, *Semeion*, *Musk*, and *Lung* for validating the effectiveness of the implemented method.

4 Experiments and Replications of Algorithms in Paper

4.1 Datasets

The experiments were conducted on five benchmark datasets selected from the original paper. The details of these datasets are summarized below.

Dataset	**No of samples**	**No of Features**	**No of classes**
Madelon	2600	500	2
Mnist	3495	784	10
Semeion	1593	256	2
Musk	6598	166	2
Lung	203	3312	5

Table 1: Summary of the benchmark datasets

4.2 Experimental Settings

The Sparse Softmax Feature Selection (S^{2}FS) algorithm was implemented in Python by integrating a Softmax classifier with $\ell_{2,0}$-norm regularization for sparsity and Linear Discriminant Analysis (LDA)-based discriminative regularization for enhanced class separability, optimized using the Alternating Direction Method of Multipliers (ADMM).

To ensure computational efficiency, we utilized CuPy for GPU-accelerated matrix operations, particularly for computing within-class (S_w) and between-class (S_b) scatter matrices and their square roots, which are critical for the LDA term. The Softmax loss was implemented using matrix-based computations for multi-class probability predictions. The $\ell_{2,0}$-norm regularization was enforced by thresholding the row norms of the weight matrix W, controlled by the sparsity hyperparameter λ. The ADMM optimization involved iterative updates for three variables: W (updated via the Generalized Power Iteration (GPI) algorithm), M (sparsity-inducing auxiliary variable), and O (Softmax optimization variable). Dual

variables Λ_1 and Λ_2 were updated alongside the penalty parameter μ, with an adaptive scaling factor $\rho = 1.1$ to ensure convergence.

A 5-fold cross-validation scheme was employed to evaluate the algorithm's performance. For each fold, the dataset was split into training and validation sets, with features selected by S^{2}FS on the training set. Hyperparameters α (weight of discriminative regularization) and λ (sparsity penalty) were tuned over a grid ($\alpha \in [10^{-6}, 10^2]$, $\lambda \in [10^{-6}, 10^{-2}]$) to select approximately 10% of features to replicate the results as in the paper.

A K-Nearest Neighbors (KNN) classifier, with $k = 5$ neighbors, was used to evaluate the classification performance of the selected features. Classification accuracy on the validation set served as the primary metric for hyperparameter selection, with the best hyperparameters chosen based on the highest average accuracy across folds. We also plotted an classification accuracy vs number of features selected plot to replicate and compare our results with those given in the paper.

Additionally, an ablation study was performed on the Madelon, Semeion and Musk dataset to analyze the impact of the discriminative regularization term in the S2FS framework.

4.3 Experimental Results

The performance of the Sparse Softmax Feature Selection (S^{2}FS) algorithm was evaluated using a 5-fold cross-validation scheme with a k-Nearest Neighbors (KNN) classifier ($k = 5$). The observed accuracy trends were consistent with those reported in the original paper, validating our implementation.

The results obtained for the 5 datasets are as follows:

Musk:

- **Best Hyperparameters:** $\alpha = 10$, $\lambda = 0.001$
- **Best Average Validation Accuracy:** 0.9492
- **Number of Selected Features:** 28

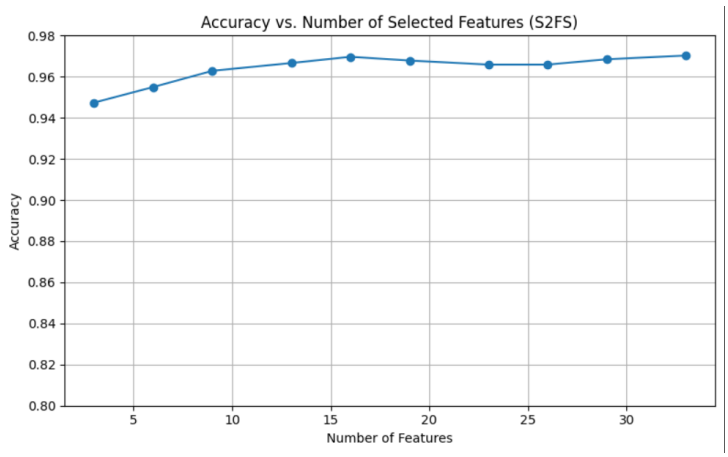

Figure 1: Accuracy vs. Number of Selected Features for Musk Dataset

Semeion:

- **Best Hyperparameters:** $\alpha = 0.0001$, $\lambda = 0.0001$
- **Best Average Validation Accuracy:** 0.7099
- **Number of Selected Features:** 252

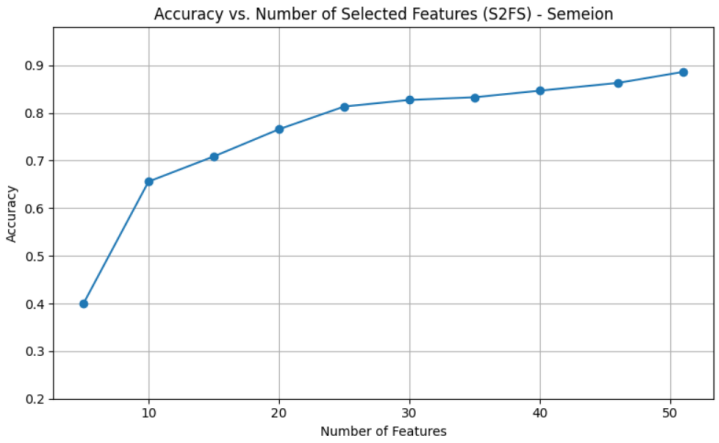

Figure 2: Accuracy vs. Number of Selected Features for Semeion Dataset

Madelon:

- **Best Hyperparameters:** $\alpha = 0.1$, $\lambda = 0.01$
- **Best Average Validation Accuracy:** 0.6641
- **Number of Selected Features:** 92

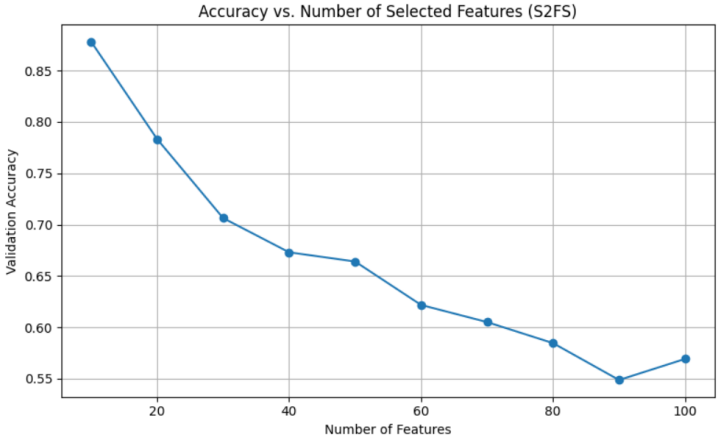

Figure 3: Accuracy vs. Number of Selected Features for Madelon Dataset

Mnist:

- **Best Hyperparameters:** $\alpha = 10^{-6}$, $\lambda = 10^{-5}$
- **Best Average Validation Accuracy:** 0.8742
- **Number of Selected Features:** 665

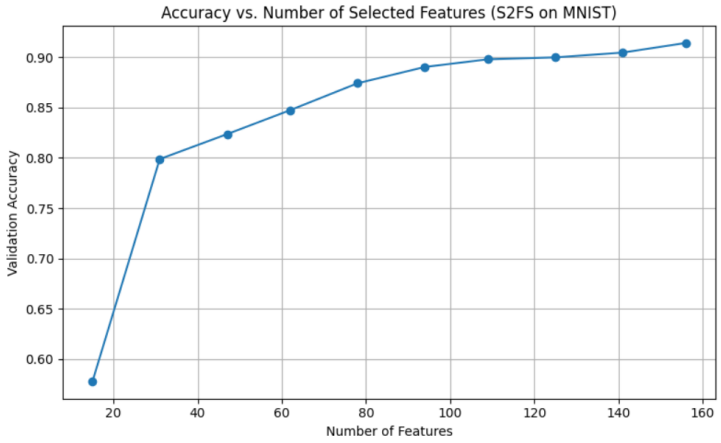

Figure 4: Accuracy vs. Number of Selected Features (S^2FS)

Lung:

- **Best Hyperparameters:** $\alpha = 1$, $\lambda = 0.0001$
- **Best Average Validation Accuracy:** 0.9508
- **Number of Selected Features:** 443

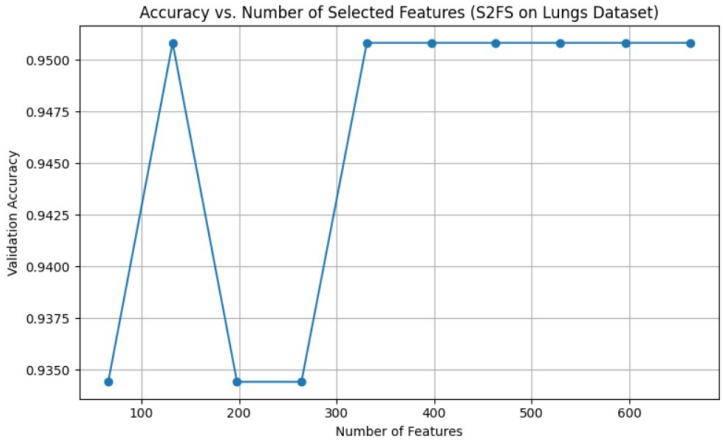

Figure 5: Accuracy vs. Number of Selected Features for Lung Dataset

4.3.1 Ablation Study

The results indicate that the inclusion of the discriminative regularization term significantly improves the ability of S^2FS to select more discriminative features, leading to better classification performance.

Dataset	Accuracy	Accuracy with Ablation
Musk	0.9492	0.9463
Semeion	0.7099	0.6711
Madelon	0.6641	0.5718

Table 2: Summary of the Results of Ablation Study

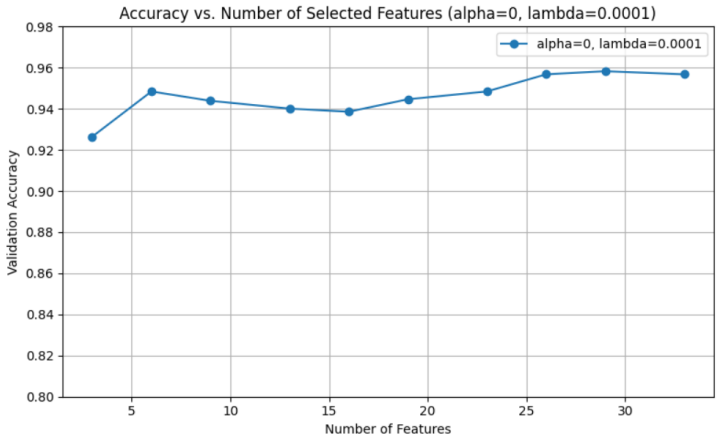

Figure 6: Accuracy vs. Number of Selected Features for Musk Dataset with Ablation (Alpha= 0)

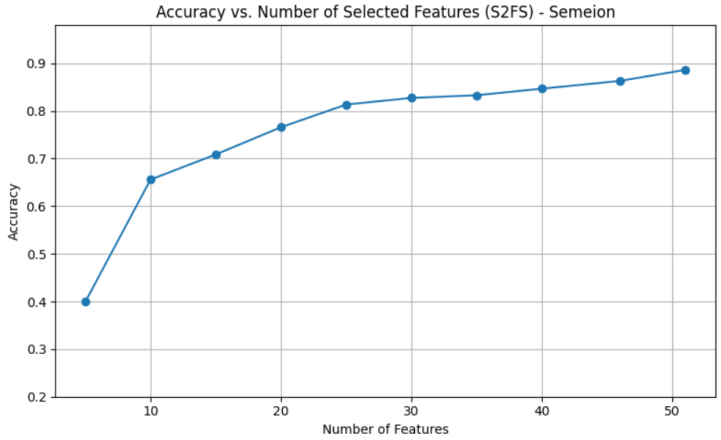

Figure 7: Accuracy vs. Number of Selected Features for Semeion Dataset with Ablation (Alpha= 0)

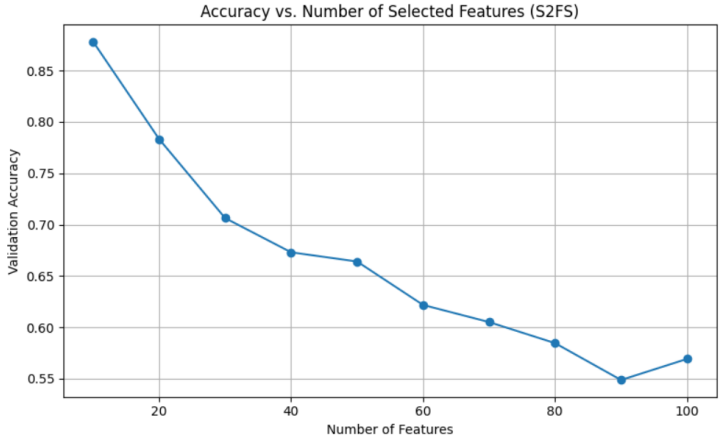

Figure 8: Accuracy vs. Number of Selected Features for Madelon Dataset with Ablation (Alpha= 0)

4.3.2 Parameter Sensitivity

A small α allows the LDA-based term to enhance class separability without overwhelming the Softmax loss, while a moderate λ effectively reduces the feature set without sacrificing discriminative power.

Conversely, higher λ values led to excessive sparsity, reducing the number of selected features which

decreased accuracy due to the exclusion of relevant features. Similarly, larger α values slightly reduced accuracy by overemphasizing the LDA term, potentially causing overfitting to class separation at the expense of generalization.

5 Novel Settings and Experiments

In this project, we extended the Sparse Softmax Feature Selection (S2FS) framework by incorporating an L1 regularizer with weight β into the loss function as a novel aspect. The L1 term enforces element-wise sparsity, complementing the $\ell_{2,0}$-norm's row-wise sparsity, which allows the algorithm to selectively choose features for specific classes.

The modified objective function with the L1 regularizer becomes:

$$\min_{W,M,N,O} \mathcal{L}(O) + \alpha \cdot \mathrm{tr}(W^\top S_w W) - \alpha \cdot \mathrm{tr}(W^\top S_b W) + \lambda \|M\|_{2,0} + \beta \|N\|_1$$

$$\text{subject to } W = M,\ W = N,\ W = O,\ W^\top S_w W = I.$$

The augmented Lagrangian equation for this optimization problem, solved using ADMM, is:

$$\mathcal{L}_\mu(W, M, N, O, \Delta_1, \Delta_2, \Delta_3) = \mathcal{L}(O) + \alpha \cdot \mathrm{tr}(W^\top S_w W) - \alpha \cdot \mathrm{tr}(W^\top S_b W) + \lambda \|M\|_{2,0} + \beta \|N\|_1 + \frac{\mu}{3} \left\| W - M + \frac{\Delta_1}{\mu} \right\|_F^2 + \frac{\mu}{3} \left\| W - N + \frac{\Delta_2}{\mu} \right\|_F^2 + \frac{\mu}{3} \left\| W - O + \frac{\Delta_3}{\mu} \right\|_F^2$$

where $\Delta_1, \Delta_2, \Delta_3$ are dual variables, and μ is the penalty parameter.

For the N-update, we apply soft thresholding:

$$N_{ij} = \mathrm{sign}(a_{ij}) \cdot \max(|a_{ij}| - \beta/\mu, 0)$$

$$\text{where } a_{ij} = W_{ij} - (\Delta_2)_{ij}/\mu.$$

Implementation: We implemented this modified S2FS framework on the Semeion dataset. The best hyperparameters obtained were $\alpha = 0.01$, $\lambda = 0.01$, and $\beta = 0.0001$, resulting in a best average validation accuracy of 0.7389 with 255 selected features, compared to the original accuracy of 0.7099 with 252 features.

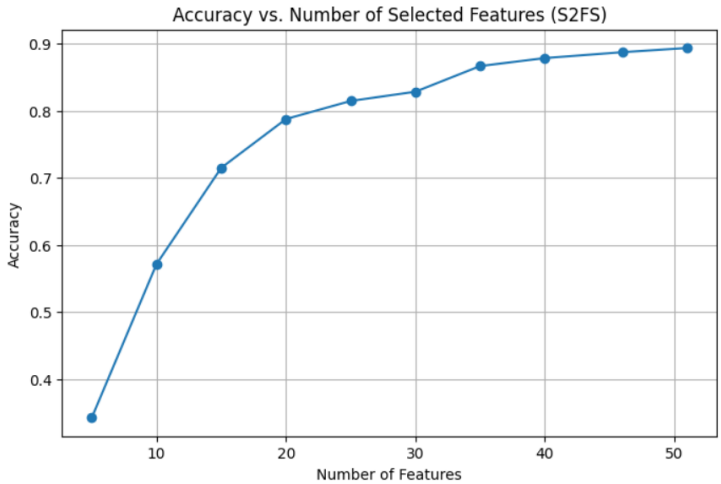

Figure 9: Accuracy vs. Number of Selected Features (Novelty Experiment

Results: The results can be interpreted as follows:

- The $\ell_{2,0}$-norm enforces row-wise sparsity, selecting features that are uniformly relevant across all classes.
- The L1 regularizer enforces element-wise sparsity, enabling the algorithm to selectively choose features for specific classes.
- As a result, the average validation accuracy improved by 2.9%.

6 Conclusion

Thus, our project focuses on implementing and evaluating the Sparse Softmax Feature Selection (S^2FS) model and its related algorithms. The problem of feature selection in high-dimensional datasets was addressed, aiming to identify a sparse subset of discriminative features to enhance classification performance and interpretability.

The methods employed included preprocessing the datasets with standardization and label encoding, followed by 5-fold cross-validation to evaluate S^2FS. Hyperparameters α and λ were tuned to select approximately 10% of features and a k-Nearest Neighbors (KNN) classifier ($k = 5$) was used to assess performance. The ablation study confirmed the LDA term's critical role in maintaining accuracy.

The significance of these results lies in their validation of S^2FS's ability to achieve high classification accuracy with a sparse feature set, aligning with the paper's findings on datasets like Mnist. These outcomes underscore the algorithm's potential for real-world applications where interpretability and efficiency are critical.

Future Work:

Future work could explore the following directions:

- Additional Datasets: Applying S^2FS to other multi-class datasets to compare performance and generalizability.
- Extension to Multi-Label Classification: Exploring and extending our method for other classification tasks, such as multi-label class problems.
- Integration with Deep Neural Networks: Incorporating S2FS with deep neural networks to enhance interpretability, efficiency, and generalization of deep learning models.
- Ensemble Methods: Combining S^2FS with ensemble classifiers (e.g., Random Forests) to further improve accuracy and robustness.

These extensions would build on the project's findings, enhancing S^2FS's applicability and addressing limitations imposed by dataset and resource constraints.

7 Contributions

The individual contributions of the teammates towards the project are as follows:

Aishwarya Jaiswal:

- Acquired a thorough understanding of the paper's content and the associated algorithms.
- Developed the problem overview and motivation and experimental slides for the Stage 1 review presentation.

- Read one related work of literature to understand the current state of work in this field.
- Contributed to the initial code development prior to the Stage 1 review.
- After the Stage 1 review, rewrote and refined the entire code in Python, ensuring proper implementation and resolving bugs in the code.
- Implemented and tested the final S2FS framework code on 3 different datasets independently.
- Conducted an ablation study as described in the original paper.
- Was pivotal in creating the final project presentation slides for the Stage 2 review.

Sayantan Maiti:

- Gained a solid understanding of the framework and algorithms in the paper.
- Designed the problem statement and solution approach slides for the Stage 1 review.
- Studied one related work to further comprehend the underlying concepts of the project.
- Enhanced and fine-tuned the code generated by AI tools to ensure proper implementation of the paper's results and algorithms.
- Independently implemented the S2FS framework on the other 2 datasets as part of experiments.
- Designed and implemented novelty experiment with L1 regularizer.
- Played a key role in compiling and writing the final project report.

References

1. Zhenzhen Sun, Zexiang Chen, Jinghua Liu, and Yuanlong Y. *Multi-class feature selection via Sparse Softmax with a discriminative regularization.* International Journal of Machine Learning and Cybernetics, 2025. `https://link.springer.com/content/pdf/10.1007/s13042-024-02185-5.pdf`

2. Tianji Pang, Feiping Nie, Junwei Han, and Xuelong Li. *Efficient Feature Selection via L2,0-norm Constrained Sparse Regression.* IEEE Transactions on Knowledge and Data Engineering, 2019. `https://ieeexplore.ieee.org/abstract/document/8386668`

3. Zheng Wang, Feiping Nie, Lai Tian, Rong Wang, and Xuelong Li. *Discriminative Feature Selection via A Structured Sparse Subspace Learning Module.* Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI), 2020. `https://www.ijcai.org/proceedings/2020/0416.pdf`