April 2025

Final Report

Ali Moughnieh

Supervised Dimensionality Reduction for High-Cardinality Categorical Variables via Group Lasso Coefficient Clustering

Abstract

This study presents a novel supervised methodology for reducing the dimensionality of high-cardinality categorical variables. Our approach leverages Group Lasso (GL) regularization to obtain regression coefficients that reflect the predictive importance of subcategories. These coefficients are then clustered using a mean shift algorithm to form a compact representation of the original feature space. We refer to this two-stage approach as Lasso Clustering (LC). Our method was benchmarked against Entity Embeddings approach as described in [8]. Both methods were evaluated on a 200k-datapoint subsample of the Rossmann Store Sales dataset, focusing on five categorical variables: Day of Week, Day, State, Month, and Store. Predictive performance was assessed using Mean Absolute Percentage Error across Neural Networks, XGBoost, K-Nearest Neighbors, and Random Forests, alongside dimensionality reduction ratios. Results demonstrate that our method achieves over 95% reduction in feature space, comparable to Entity Embeddings (96.53%), while offering clearer interpretability of category groupings based on linear predictive contributions. Although LC shows higher MAPE scores than EE across all models, it still provides improvement over the full high-dimensional feature set for certain models, highlighting a trade-off between interpretability and raw predictive power.

1 Introduction

High-cardinality categorical variables—those with a large number of distinct values—are common in real-world datasets but pose challenges for machine learning models. Standard one-hot encoding (OHE) leads to sparse, high-dimensional feature spaces that can increase computational costs and the risk of overfitting [3, 6]. Although entity embeddings have been explored to create dense representations [8], these approaches typically require complex architectures and may lack interpretability.

Motivated by the need for simpler yet effective alternatives, this research introduces a supervised dimensionality reduction approach based on GL regularization. Our method, which we term Lasso Clustering (LC), first identifies informative subcategories through their regression coefficients, which are then clustered using a mean shift algorithm to form a compact representation. In our final implementation, we focus on five categorical variables from the Rossmann Store Sales dataset [5], with evaluations based on the Mean Absolute Percentage Error and dimensionality reduction ratios.

2 Literature Review

Existing research on dimensionality reduction for high-cardinality categorical data can be broadly categorized into two streams. The first stream utilizes unsupervised techniques—such as Multiple Correspondence Analysis (MCA) [1] and similarity encoding [3]—to project categorical data into low-dimensional spaces by capturing inter-category associations without reference to a target variable. Although these approaches yield interpretable factor maps, they may overlook predictive relationships crucial for certain tasks.

The second stream focuses on supervised techniques that incorporate target information during dimensionality reduction. Methods such as target encoding [10, 7] and neural network-derived entity embeddings [8] learn dense representations that directly capture relationships between categories and outcomes. Additionally, techniques like Value Clustering for Categorical Attributes [2] group categories based on target-dependent probabilities, and regularization approaches—most notably Group Lasso [12] and its sparse variant [11]—enforce sparsity by applying L1 regularization to eliminate non-informative groups. Collectively, these methods underscore the benefits of integrating target signals for enhanced model performance.

3 Proposed Methodology

Group Lasso extends traditional Lasso by promoting sparsity at the group level rather than at the individual coefficient level. While standard Lasso applies an L1 penalty directly to individual coefficients, Group Lasso penalizes groups of coefficients by applying the L2 norm within each predefined group and then summing these norms—this summation effectively acts as an L1 penalty on the groups.

In our approach, we employ a sparse GL framework [11], which not only applies an L2 penalty within each group, but also incorporates an additional L1 penalty across individual coefficients. This dual penalty structure ensures that uninformative groups are eliminated while also inducing sparsity within the selected groups, leading to a more parsimonious model.

This approach is particularly suitable for categorical variables, where each category, after one-hot encoding, forms a group of dummy variables. Group Lasso can then select or eliminate entire categorical predictors rather than individual dummy variables, preserving the structural integrity of the categorical features.

Our methodology addresses the dimensionality challenges of high-cardinality categorical variables through a two-stage approach that preserves predictive power while significantly reducing the feature space. First, we apply GL regularization to identify and eliminate entire groups of non-informative categorical variables. Then, for the retained categorical variable groups (those with at least one non-zero coefficient), we cluster subcategories based on their regression coefficients to create a compact representation that maintains essential predictive relationships. This approach balances noise elimination with signal preservation.

3.1 Stage 1: Group Lasso Regularization

We employ GL regularization using the Sparse GL framework [11] to identify informative subcategories:

$$\arg\min_{\beta} \left(L(\beta, \mathbf{X}, \mathbf{y}) + \lambda_1 \|\beta\|_1 + \lambda_2 \sum \|\beta_{\text{group}}\|_2 \right), \tag{1}$$

where L represents the task-appropriate loss function, $\|\beta\|_1$ denotes the L1 norm encouraging sparsity across subcategories, and $\|\beta_{group}\|_2$ represents the L2 norm within each group. The hyperparameters λ_1 and λ_2 control within-group and between-group sparsity respectively and are optimized via cross-validation.

Bayesian optimization was employed to automatically tune the hyperparameters λ_1 and λ_2 by minimizing the average MAPE across the splits. We defined the search space for both λ_1 and λ_2 over a logarithmic scale (from 10^{-6} to 10^2) to capture a wide range of potential values. Using a **Gaussian Process surrogate model**—a probabilistic model that approximates the unknown relationship between hyperparameters and MAPE—along with an **Expected Improvement** acquisition function (which balances exploration of uncertain regions and exploitation of promising candidates), the optimizer iteratively evaluated the MAPE over a specified number of iterations (we used 10) to efficiently converge to near-optimal regularization parameters. This approach avoids the computational expense of exhaustive grid search.

Only groups that were not eliminated after regularization are retained for the next stage, as they are deemed to carry predictive signal.

3.2 Stage 2: Coefficient Clustering via Mean Shift

For the categorical variables that remain after GL regularization, we cluster their corresponding regression coefficients. Although various clustering methods were considered, the mean shift

algorithm yielded the best performance in detecting local density changes by identifying clusters via local density maxima [4].

A key aspect of the LC methodology was the choice of clustering algorithm for the GL coefficients. While standard algorithms like K-Means were initially considered, experimentation revealed limitations for this specific task. K-Means partitioned the coefficients based on minimizing variance around global cluster means, sometimes failing to respect clear visual breaks evident in the sorted coefficient distributions. The **Mean Shift algorithm** was ultimately selected as more appropriate. Its density-based approach, which does not require pre-specifying the number of clusters, and utilizes a bandwidth window to find local modes, proved much better suited to identifying these natural groupings and breaks inherent in the coefficient values.

A crucial parameter for the Mean Shift algorithm is the bandwidth, which determines the size of the region to search for density maxima. Smaller bandwidths are more sensitive to local density variations and tend to produce more clusters. In this study, the bandwidth was treated as a tunable parameter, adjusted individually for each categorical variable's coefficient set. The selection was guided by a combination of visual inspection of the resulting clusters on the sorted coefficient plots and the goal of achieving a final feature dimensionality comparable to the EE baseline. This allowed the clustering to align more closely with the goal of grouping subcategories exhibiting similar estimated linear effects.

3.3 Implementation Pipeline

The complete pipeline is summarized below:

1. **Preprocessing:** All features in the dataset are categorical and were OHE.
2. **Group Lasso Modeling:** The full dataset consists of 844,338 datapoints. To extract robust coefficient estimates, we split the data into four roughly equal parts (approximately 211k datapoints each) and applied GL regularization on each split separately. The coefficients from each split are then averaged, which not only leverages the full dataset for reliable estimation but also reduces variability across individual splits. The purpose of this step is to align with the downstream modeling as explained in Section 4.1.
3. **Coefficient Clustering:** The averaged coefficients were clustered using the Mean Shift algorithm (see Section 4.2). While formal cluster quality metrics weren't used, the resulting clusters aligned well with visually discernible groupings in the coefficient distributions, reinforcing our choice in the context of this application.
4. **Feature Reduction:** The original features were replaced with cluster indicator variables, and then OHE was applied to them.
5. **Downstream Modeling:** Predictive models were trained on the reduced feature set using four models: Neural Network, XGBoost, K-Nearest Neighbors, and Random Forests.

With the methodology established, we now describe the experimental setup used to evaluate our approach.

4 Experimental Setup

4.1 Dataset

The Rossmann Store Sales dataset [5] was used for evaluation, with `Sales` as the prediction target. To evaluate the performance of the downstream predictive models, a subsample consisting of 200,000 datapoints was utilized. This specific subsample size was chosen to directly align with the experimental setup described in the EE paper [8], thereby simulating practical

scenarios where large datasets might not be accessible or where model performance on sparser data is of interest.

4.2 Feature Sets Compared

Each model was trained and tested on three datasets:

1. **Full Set**: Standardized numerical features and OHE categorical variables.
2. **Entity Embedding Reduced Set**: High-cardinality categorical variables replaced by embeddings generated using the code from [8] on our subsample.
3. **Lasso Clustering Reduced Set**: The five categorical features undergoing reduction (`Store`, `DayOfWeek`, `State`, `Month`, `Day`) replaced by cluster indicators from GL + Mean Shift. Other features remain as in the Full Set.

See Appendix, Table 2 for a summary of the features in each dataset.

4.3 Downstream Models and Baselines

Models mirrored from the EE paper: Neural Network (NN), XGBoost, K-Nearest Neighbors (KNN), and Random Forest (RF). The primary baseline was the EE method [8]. To ensure a direct and fair comparison, the performance results for EE reported in this study were generated by re-executing the original authors' publicly available code using the exact same data subsample and train/validation split that were used to evaluate both our Lasso Clustering method and the full feature set.

4.4 Metric

The primary metric was **Mean Absolute Percentage Error (MAPE)**. MAPE was chosen because percentage deviation from the actual value is more interpretable for sales predictions than metrics such as Mean Squared Error (MSE). We also report the overall **dimensionality reduction ratio**, which intuitively indicates the performance gain achieved through feature reduction.

4.5 Validation and Parameter Settings

To closely replicate the EE study for a fair comparison, we used the following settings:

- **Validation**: Models were evaluated on a single, fixed train/validation split of the 200k data subsample.
- **Downstream Model Parameters**: We directly used the hyperparameter configurations for the four models from the EE paper without further tuning. The same parameters were used for our re-run EE baseline and the LC/Full set evaluations. Refer to Appendix, Table 1 for the parameters used in each model.

5 Results

The following figures present the key outcomes of our experiments.

5.1 Coefficient Clustering Visualization

Figure 1 shows the GL coefficients for high-cardinality variables (sorted in descending order) with subcategories colored by the cluster label obtained from the mean shift algorithm.

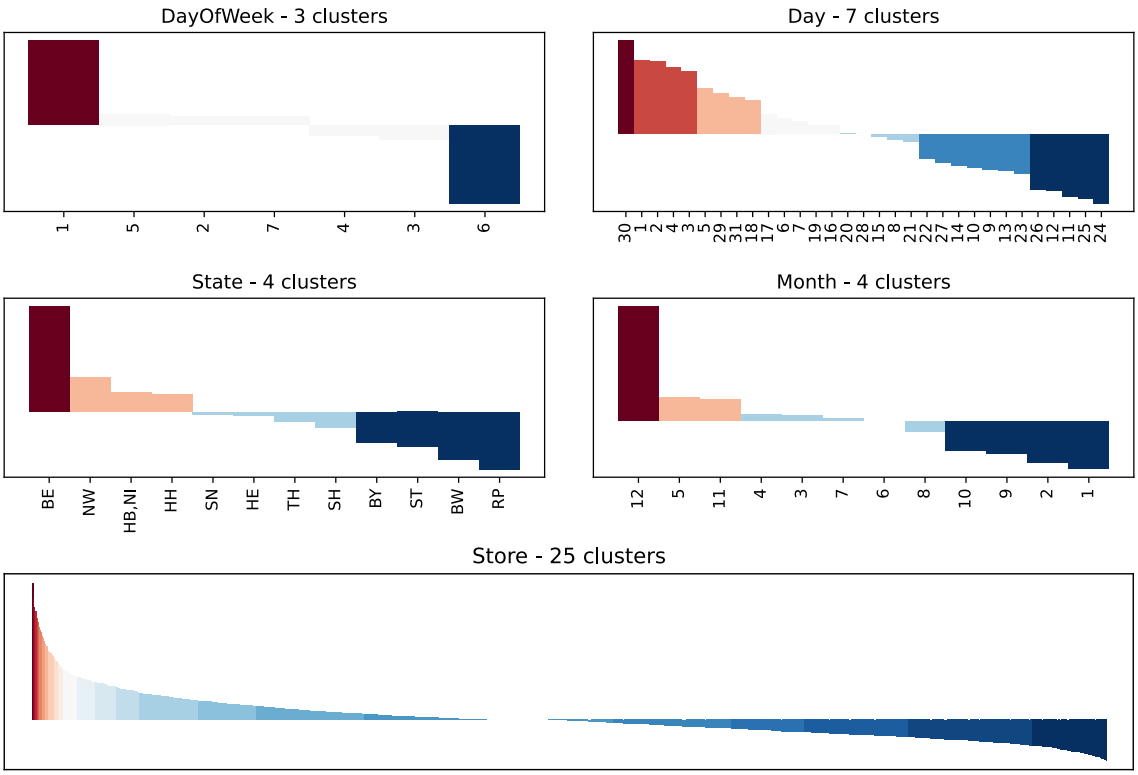
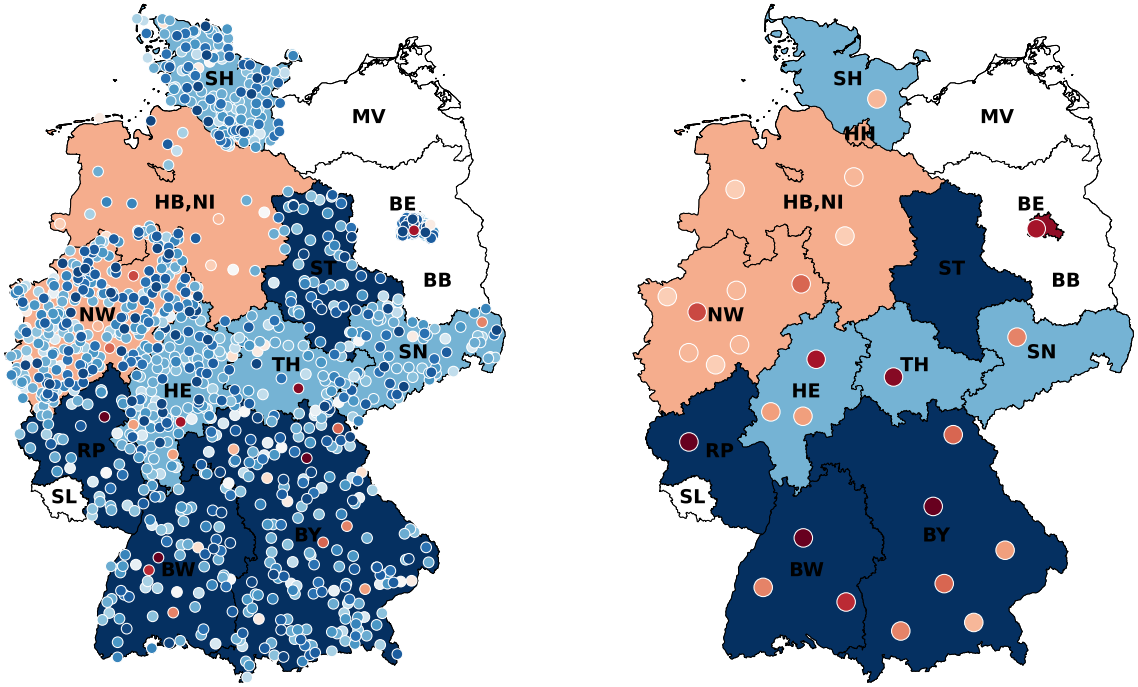


Figure 1: Group Lasso coefficients for high-cardinality variables sorted in descending order and colored by cluster label.

5.2 State Clustering Map



(a) All stores colored by cluster labels with state regions showing state-level cluster labels

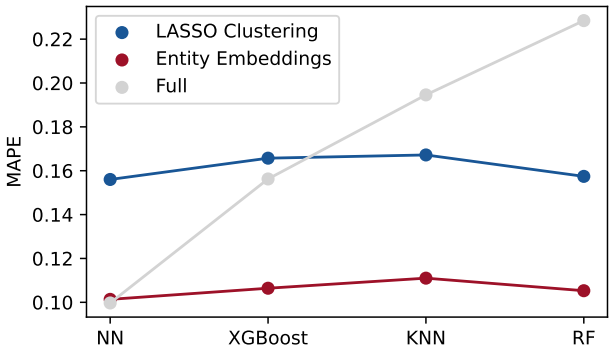
(b) Top 10 most influential store clusters (highest positive Lasso coefficients)

Figure 2: Store-level clusters overlaid on state-level clusters in Germany

The maps in Figure 2 visualize the geographical distribution of the clusters derived from the Group Lasso coefficients. States are colored based on their state-level cluster assignment, while individual store locations are shown as circles colored according to their store-level cluster membership.

5.3 Model Performance

Figure 3 presents the MAPE scores for the four models across the three feature sets.



Model	LC	EE	Full
NN	0.1560	0.1013	0.0997
XGBoost	0.1657	0.1064	0.1562
KNN	0.1672	0.1110	0.1946
RF	0.1574	0.1053	0.2284
Dimension	48	41	1182
Dim. Reduction (%)	95.94%	96.53%	0.00%

Figure 3: Model performance comparison using Mean Absolute Percentage Error (MAPE)

The results show that the LC feature set led to noticeably higher MAPE scores, indicating lower predictive accuracy, compared to the EE feature set across all tested models. For instance, with the Neural Network, LC achieved a MAPE of 0.1560, whereas EE achieved 0.1013 and the Full set achieved 0.0997. Similarly, for XGBoost, LC's MAPE was 0.1657 compared to 0.1064 for EE. While both LC and EE offered dimensionality reduction, EE consistently provided much better predictive accuracy than LC. However, it's worth noting that LC, despite its higher MAPE compared to EE, still showed improvements over the high-dimensional Full set for certain models like KNN and RF, an aspect explored further in the Discussion.

6 Discussion

The experimental results demonstrate the application of the proposed LC mechanism and offer insights into its characteristics.. Key observations include:

- **Effective Clustering Visualization:** The GL coefficients provided a basis for clustering, and the Mean Shift algorithm effectively identified visually distinct clusters within these coefficients, as illustrated in Figure 1.
- **Multi-level Interpretability:** The LC approach allows for visualization of relationships between different categorical variables, as demonstrated in Figure 2. Examining the store-level clusters overlaid on the state-level clusters reveals a pattern where most stores associated with the strongest positive impact on sales predictions (dark red circles) appear to be located within states whose overall coefficient suggests a negative impact (dark blue states). This observation suggests that the LC method might potentially allow for the identification of store-level effects that differ from broader state-level patterns derived from the linear model. Such visualization can help identify potential interactions between different categorical variables that might otherwise be difficult to observe in high-dimensional data or with less interpretable reduction techniques.

- **High Dimensionality Reduction:** The LC method achieved a substantial dimensionality reduction of 95.94% for the features considered, comparable to the 96.53% achieved by Entity Embeddings (Figure 3), reducing the feature count to 48 dimensions versus 41 for EE.

However, when evaluating predictive performance, a clear trade-off emerges. Despite the effective clustering visualization and high dimensionality reduction, the LC feature set resulted in consistently and substantially higher MAPE scores compared to the EE baseline across all models (Figure 3). A key reason likely stems from the fundamental difference in the nature of the reduced representations, despite the similar dimension counts. LC yields sparse OHE vectors indicating cluster membership, while EE produces dense, continuous vectors for each category. These dense vectors can encode more detailed information about inter-category relationships and nuances compared to sparse indicators. This difference in representational richness likely contributes significantly to EE's superior predictive performance. This highlights a core trade-off: LC's interpretable grouping process results in a loss of information compared to EE's dense representation.

Despite this representational difference affecting overall prediction accuracy, LC offers the benefit of clear, linear interpretability of categorical impacts through its coefficients used for clustering (Figure 1 and 2). This transparency contrasts with EE's less directly interpretable dense embeddings.

Furthermore, an interesting nuance can be observed. Although XGBoost with the full feature set slightly outperformed LC, the difference was minimal. When we disregard the Neural Network (which is inherently complex) and focus on the remaining models, the LC set not only provides a substantial reduction in the number of variables, but also allows models—especially those that typically struggle with high-dimensional datasets—to perform comparably to XGBoost. This suggests that LC offers a practical balance between interpretability, computational efficiency, and predictive performance in scenarios where the interpretability of the reduction process itself is highly valued.

Limitations: The findings are based on a single dataset and one specific implementation (GL + Mean Shift targeting 5 variables). The choice of GL hyperparameters and Mean Shift parameters may influence outcomes. The comparison was limited to EE. Furthermore, the final sparse OHE representation generated by LC inherently limits the detail passed to downstream models compared to dense methods.

Future Work: Testing the LC framework on diverse datasets and target variable types remains crucial. Since the linear coefficients from GL may not capture non-linear patterns, potentially leading to information loss upon clustering, future work could explore generating the input scores for clustering using **non-linear models**. Deriving coefficients, rankings, or importance scores from models that account for non-linear interactions might lead to more informative groupings. Clustering based on such richer scores could potentially improve the balance between dimensionality reduction and predictive accuracy.

7 Conclusion

This project implemented and evaluated Lasso Clustering, a two-stage technique combining Group Lasso regularization with Mean Shift coefficient clustering for supervised dimensionality reduction of high-cardinality categorical variables. The method achieved substantial feature space reduction (95.94%), comparable to Entity Embeddings (96.53%), while offering clearer interpretability regarding the formation of category groups based on linear predictive contributions.

However, this interpretability came at a cost in predictive accuracy, where LC consistently yielded higher MAPE scores than EE across all tested models. This is largely because LC's final output is a sparse representation (OHE cluster indicators) which encodes less detailed information than the dense embeddings produced by EE.

Despite underperforming EE, LC's performance relative to the full, high-dimensional feature set was more nuanced, especially with XGBoost. This suggests LC can still be a competitive alternative when dramatic dimensionality reduction and interpretability of the reduction process are paramount, particularly for models that struggle with high dimensional data.

In summary, LC presents a viable trade-off favouring interpretability and dimensionality reduction over raw predictive power compared to EE, partly due to its sparse output format. Future work should focus on enhancing predictive performance, potentially by exploring non-linear methods to generate scores, thereby aiming to bridge the gap between interpretability and accuracy.

Project Repository

The source code for this project is available on GitHub: https://github.com/amoughnieh/Lasso-Clustering.

References

[1] Hervé Abdi and Dominique Valentin. Multiple correspondence analysis. *Encyclopedia of measurement and statistics*, 2:651–657, 2007.

[2] Emanuel Mineda Carneiro, Carlos Henrique Quartucci Forster, Lineu Fernando Stege Mialaret, Luiz Alberto Vieira Dias, and Adilson Marques da Cunha. High-cardinality categorical attributes and credit card fraud detection. *Mathematics*, 10(20), 2022. ISSN 2227-7390. URL https://www.mdpi.com/2227-7390/10/20/3808.

[3] Patricio Cerda, Gaël Varoquaux, and Balázs Kégl. Similarity encoding for learning with dirty categorical variables. *Mach. Learn.*, 107(8–10):1477–1494, September 2018. ISSN 0885-6125. doi: 10.1007/s10994-018-5724-2. URL https://doi.org/10.1007/s10994-018-5724-2.

[4] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002. doi: 10.1109/34.1000236.

[5] Will Cukierski. Rossmann store sales dataset. URL https://www.kaggle.com/competitions/rossmann-store-sales/overview.

[6] Pedro Domingos. A few useful things to know about machine learning. *Commun. ACM*, 55(10):78–87, October 2012. ISSN 0001-0782. doi: 10.1145/2347736.2347755. URL https://doi.org/10.1145/2347736.2347755.

[7] Anna Veronika Dorogush, Vasily Ershov, and Andrey Gulin. Catboost: gradient boosting with categorical features support. *ArXiv*, abs/1810.11363, 2018. URL https://api.semanticscholar.org/CorpusID:26037613.

[8] Cheng Guo and Felix Berkhahn. Entity embeddings of categorical variables. *ArXiv*, abs/1604.06737, 2016. URL https://api.semanticscholar.org/CorpusID:40629394.

[9] Ali Moughnieh. Lasso clustering repository, 2025. URL https://github.com/amoughnieh/Lasso-Clustering.

[10] Florian Pargent, Florian Pfisterer, Janek Thomas, and Bernd Bischl. Regularized target encoding outperforms traditional methods in supervised machine learning with high cardinality features. *Comput. Stat.*, 37(5):2671–2692, November 2022. ISSN 0943-4062. doi: 10.1007/s00180-022-01207-6. URL https://doi.org/10.1007/s00180-022-01207-6.

[11] Noah Simon, Jerome H. Friedman, Trevor J. Hastie, and Robert Tibshirani. A sparse-group lasso. *Journal of Computational and Graphical Statistics*, 22:231 – 245, 2013. URL https://api.semanticscholar.org/CorpusID:2208574.

[12] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(1):49–67, 12 2005. ISSN 1369-7412. doi: 10.1111/j.1467-9868.2005.00532.x. URL https://doi.org/10.1111/j.1467-9868.2005.00532.x.

Appendix

XGBoost	
max_depth	10
eta	0.02
objective	reg:linear
colsample_bytree	0.7
subsample	0.7
num_round	3000
Random Forest	
n_estimators	200
max_depth	35
min_samples_split	2
min_samples_leaf	1
KNN	
n_neighbors	10
weights	distance
p	1
Neural Network	
epochs	10
batch_size	128
hidden_layers	2
units_per_layer	1000, 500
activation	relu (hidden)
	sigmoid (output)
optimizer	adam
loss	mean_absolute_error

Table 1: Parameters of downstream models

Features	**EE**	**LC**	**Full**
store	10	25	1115
day of week	6	3	7
day	10	7	31
month	6	4	12
year	2	3	3 (2013-2015)
promotion	1	2	2
state	6	4	12
Total	**41**	**48**	**1182**

Table 2: Number of features of the EE, LC, and full datasets