Appendix - Online Heterogeneous Feature Selection

I. Supplementary Details of the Proposed Method

This section presents detailed theoretical analysis to further claim the underlying principles of the proposed method.

A. Pseudocode of the ANDR Module

The procedure for constructing the neighborhood relation $R_{\mathcal{F}'}(\mathbf{x}_i)$ to evaluate feature subset significance $S(\mathcal{F}'_t)$ is summarized in Algorithm 1.

Algorithm 1 ADNR-based significance quantification of feature subset.

```
Input: $U$, $C$, $V$, $\mathcal{F}'$, $\mathcal{D}^r$.
Output: $S\left(\mathcal{F}'\right)$.
Calculate the distance matrix for the set of $U$ on $\mathcal{F}$.
for $i = 1$ to $n$ do
    Calculate $R^{\mu}_{\mathcal{F}'}(\mathbf{x}_i)$ for sample using Eqs. (10) and (11).
end for
Compute $S\left(\mathcal{F}'\right)$ according to Eqs. (1), (2) and (3).
```

B. Detailed Theoretical Analysis

Theorem 1. *The time complexity of GRADE at any given timestamp t is $O(mn^2 + mn^2 \log n)$.*

Proof. *Analyzing from the worst-case scenario, it is assumed that all incoming features are categorical. At timestamp t, let $m = |\mathcal{F}'_t|$ denote the number of features in the current subset, $\mathrm{V} = \max\left(\mathrm{v}^1, \mathrm{v}^2, \ldots, \mathrm{v}^m\right)$ the maximum number of distinct values, and l the number of class labels.*

Transformation Cost Matrix Computation. *For categorical feature $\mathbf{f}_t$, computing the transformation cost matrix M_t involves calculating the conditional probability distribution for each feature value via Eq. (6), scanning n samples over l labels for V values, with complexity $O(nl\mathrm{V})$. Pairwise distances between feature values are then computed using Eq. (9), which requires mutual information weights from Eq. (8). Constructing the joint distribution takes $O(nl\mathrm{V})$, and computing distances for $\frac{\mathrm{V}(\mathrm{V}-1)}{2}$ value pairs costs $O(l\mathrm{V}^2)$. The total complexity is $O(nl\mathrm{V} + l\mathrm{V}^2)$.*

Feature Selection Process. *Calculating the significance of the current feature subset $S\left(\mathcal{F}'\right)$ involves two key steps, neighborhood set construction and positive region calculation. To construct neighborhood sets, an $n \times n$ sample-wise distance matrix is first built with a complexity $O(n^2)$. The rows of the matrix are then sorted, adding a complexity of $O(n^2 \log n)$. To establish the neighborhood boundaries, density calculations via Eqs. (10) and (11) for each of the n samples involve $O(n^2)$ operations. The overall time complexity for determining the neighborhood sets is thus $O(n^2 + n^2 \log n + n^2)$, simplifying to $O(n^2 + n^2 \log n)$. For positive region calculation, Eq. (1) requires verifying whether each of the n neighborhood sets is a subset of the class C_m, corresponding to $O(n^2 l)$ operations. Eq. (2) then describes the union of l set, each containing at most n samples, with a time complexity of $O(nl)$. The total time complexity for the significance of the feature subset $S\left(\mathcal{F}'\right)$ is $O(n^2 + n^2 \log n + n^2 l)$ Since $S\left(\mathcal{F}'\right)$ must be computed at most m times, the overall time complexity is $O(nl\mathrm{V} + l\mathrm{V}^2 + m(n^2 + n^2 \log n + n^2 l))$. Given that V and l are constants with relatively small values, the final time complexity simplifies to $O(mn^2 + mn^2 \log n)$.* □

Theorem 2. *The time complexity of GRADE can be reduced to $O(mn + mn \log n)$ through simple parallelization.*

Proof. *The two main computational stages, i.e., pairwise distance calculation and adaptive neighborhood construction, are independent at the sample level. Distributing the $O(mn^2)$ distance computations and $O(mn^2 \log n)$ neighborhood evaluations across u computing nodes can reduce the per-unit cost to $O(\frac{mn^2}{u})$ and $O(\frac{mn^2 \log n}{u})$, respectively. By expanding the computing node scale to n, the complexity can be reduced to $O(mn + mn \log n)$.* □

II. Supplementary Experimental Results

This section contains supplementary experiments providing additional validation of the proposed method's effectiveness.

A. Statistical Significance Test Results

To evaluate the statistical significance of performance differences, the Wilcoxon signed-rank test is conducted between GRADE and its counterparts, based on the Acc-S and Acc-K results reported in Table V of the main paper. The corresponding p-values are shown in Fig. 1, where darker colors denote higher significance levels. The results clearly confirm that GRADE achieves statistically significant improvements over most competitors at the 95% confidence level. Even compared with the advanced and competitive IFSNR, the superiority of GRADE remains statistically significant at approximately the 90% confidence level w.r.t. Acc-S and Acc-K, respectively. These findings confirm that GRADE delivers statistically robust improvements across classifiers, underscoring its generalizable contribution to effective online feature selection.

B. Redundancy-Handling Strategy Evaluation

To further validate the rationality of the feature selection process, three alternative workflow designs are compared with GRADE. Hereafter, "RC" denotes "redundancy check", which refers to the procedure of first incorporating a feature into the subset and then checking for potential redundancy. The three variants are: 1) performing RC immediately after including a significant feature (GRADE-IncludeRC), 2) performing RC after identifying an irrelevant feature (GRADE-DiscardRC), 3) applying RC in both cases (GRADE-FullStageRC).

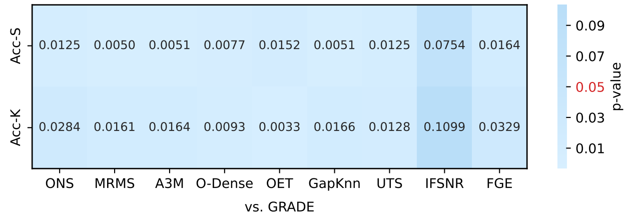

Fig. 1: p-values of the Wilcoxon signed rank test in comparing our method against the other methods in terms of Acc-S and Acc-K.

TABLE I: Performance comparison of GRADE and its variants with different redundancy-handling strategies across twelve datasets, reported in terms of Acc-S, Acc-K, the number of selected features (# Feat.), and runtime (Runt.).

Datasets	Index	GRADE	GRADE-IncludeRC	GRADE-DiscardRC	GRADE-FullStageRC
MI	Acc-S	**0.7974 ± 0.08**	0.7775 ± 0.06	0.7803 ± 0.06	0.7718 ± 0.05
	Acc-K	**0.7921 ± 0.07**	0.7347 ± 0.07	0.7320 ± 0.05	0.7290 ± 0.07
	# Feat.	7.30 ± 1.35	6.40 ± 1.20	7.70 ± 1.00	7.00 ± 0.63
	Runt.	13.89 ± 3.13	22.07 ± 1.76	18.78 ± 1.72	8.87 ± 3.86
SC	Acc-S	**0.7857 ± 0.17**	0.7286 ± 0.14	0.7571 ± 0.20	0.7429 ± 0.21
	Acc-K	**0.7714 ± 0.13**	0.6429 ± 0.23	0.7000 ± 0.19	0.7143 ± 0.21
	# Feat.	8.20 ± 0.78	7.00 ± 1.34	7.90 ± 1.45	6.20 ± 0.60
	Runt.	2.00 ± 0.15	2.70 ± 0.24	2.68 ± 0.32	2.55 ± 0.15
AR	Acc-S	**0.6738 ± 0.10**	0.5810 ± 0.08	0.5833 ± 0.08	0.5905 ± 0.10
	Acc-K	**0.6881 ± 0.09**	0.5429 ± 0.06	0.5571 ± 0.07	0.5571 ± 0.06
	# Feat.	13.70 ± 3.20	7.10 ± 0.70	8.90 ± 0.83	7.90 ± 1.04
	Runt.	62.36 ± 6.78	90.94 ± 2.87	108.71 ± 16.85	98.11 ± 3.84
PD	Acc-S	**0.7457 ± 0.10**	**0.7457 ± 0.10**	**0.7457 ± 0.10**	**0.7457 ± 0.10**
	Acc-K	**0.7457 ± 0.10**	**0.7457 ± 0.10**	**0.7457 ± 0.10**	**0.7457 ± 0.10**
	# Feat.	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
	Runt.	118.67 ± 1.39	118.20 ± 1.33	156.00 ± 4.49	123.13 ± 4.15
MU	Acc-S	**0.6055 ± 0.17**	0.5282 ± 0.06	0.5974 ± 0.07	0.5596 ± 0.06
	Acc-K	**0.6468 ± 0.17**	0.6175 ± 0.05	0.6324 ± 0.07	0.6156 ± 0.05
	# Feat.	10.70 ± 0.78	6.80 ± 2.09	10.90 ± 0.70	6.30 ± 1.00
	Runt.	28.19 ± 5.73	60.93 ± 8.50	85.21 ± 6.54	68.86 ± 6.08
PC	Acc-S	0.6889 ± 0.42	**0.7000 ± 0.12**	0.6889 ± 0.42	0.6556 ± 0.16
	Acc-K	0.6111 ± 0.38	**0.6783 ± 0.13**	0.5987 ± 0.16	0.5844 ± 0.15
	# Feat.	5.80 ± 4.26	4.80 ± 0.87	4.10 ± 0.54	4.60 ± 0.92
	Runt.	7.45 ± 0.73	19.11 ± 0.49	18.01 ± 1.65	21.00 ± 0.72
TO	Acc-S	0.6647 ± 0.44	0.6706 ± 0.10	0.6647 ± 0.10	**0.6725 ± 0.10**
	Acc-K	0.4775 ± 0.08	0.5085 ± 0.11	**0.5153 ± 0.10**	0.5081 ± 0.08
	# Feat.	6.10 ± 0.83	4.30 ± 0.78	6.10 ± 0.70	4.30 ± 0.64
	Runt.	20.13 ± 2.69	59.15 ± 1.69	64.45 ± 4.95	65.43 ± 3.72
AC	Acc-S	**0.7550 ± 0.10**	0.6450 ± 0.06	0.6450 ± 0.04	0.6650 ± 0.10
	Acc-K	**0.8250 ± 0.06**	0.6900 ± 0.05	0.7150 ± 0.04	0.6900 ± 0.10
	# Feat.	5.00 ± 0.00	5.00 ± 0.00	4.40 ± 0.49	4.60 ± 0.49
	Runt.	354.73 ± 32.69	372.15 ± 63.12	2098.22 ± 505.81	1500.46 ± 533.39
MA	Acc-S	0.5267 ± 0.05	0.5150 ± 0.03	0.4817 ± 0.05	**0.5417 ± 0.06**
	Acc-K	0.5200 ± 0.04	0.5050 ± 0.04	0.5050 ± 0.06	**0.5233 ± 0.05**
	# Feat.	2.00 ± 0.00	2.00 ± 0.00	2.00 ± 0.00	2.00 ± 0.00
	Runt.	7.98 ± 0.21	7.92 ± 0.21	7.22 ± 0.13	8.97 ± 0.45
QA	Acc-S	0.8917 ± 0.28	0.8945 ± 0.02	**0.8957 ± 0.03**	0.8821 ± 0.02
	Acc-K	**0.8798 ± 0.02**	0.8786 ± 0.02	0.8738 ± 0.03	0.1678 ± 0.03
	# Feat.	23.80 ± 4.21	24.30 ± 1.79	23.20 ± 1.47	4.30 ± 1.27
	Runt.	335.18 ± 26.25	376.07 ± 26.33	519.14 ± 321.99	15.07 ± 2.48
HI	Acc-S	**0.9634 ± 0.03**	**0.9634 ± 0.03**	**0.9634 ± 0.03**	**0.9634 ± 0.03**
	Acc-K	**0.9634 ± 0.03**	**0.9634 ± 0.03**	**0.9634 ± 0.03**	**0.9634 ± 0.03**
	# Feat.	10.60 ± 0.49	10.70 ± 0.64	10.20 ± 0.75	10.50 ± 0.67
	Runt.	88.66 ± 10.99	91.20 ± 10.22	75.19 ± 6.39	125.04 ± 12.77
LE	Acc-S	**0.9152 ± 0.05**	0.8752 ± 0.10	0.8314 ± 0.10	0.8210 ± 0.13
	Acc-K	**0.8321 ± 0.18**	0.7333 ± 0.13	0.7200 ± 0.16	0.8067 ± 0.11
	# Feat.	2.70 ± 0.30	2.60 ± 0.80	2.80 ± 0.75	2.80 ± 0.75
	Runt.	71.46 ± 0.50	149.24 ± 31.88	234.48 ± 25.16	476.50 ± 57.71
$\overline{\text{Rank}}$	Acc-S	2.1	2.9	2.6	3.1
	Acc-K	1.9	2.9	2.6	3.2
	# Feat.	3.5	2.6	2.4	2.1
	Runt.	1.8	3.0	3.6	3.6

The performance of GRADE and its variants with different redundancy-handling strategies is compared in Table I. Notably, GRADE and its variant GRADE-DiscardRC, which performs enhanced checking upon discarding any feature, achieve the highest accuracy, confirming that deferring redundancy checks helps retain discriminative features. However, GRADE-DiscardRC incurs significantly higher runtime, revealing the computational cost of exhaustive redundancy analysis. In contrast, the more aggressive GRADE-IncludeRC and GRADE-FullStageRC produce more compact subsets at the cost of accuracy, reflecting the potential loss of complementary features. Overall, GRADE demonstrates a balanced trade-off between performance and computational efficiency among the evaluated variants.

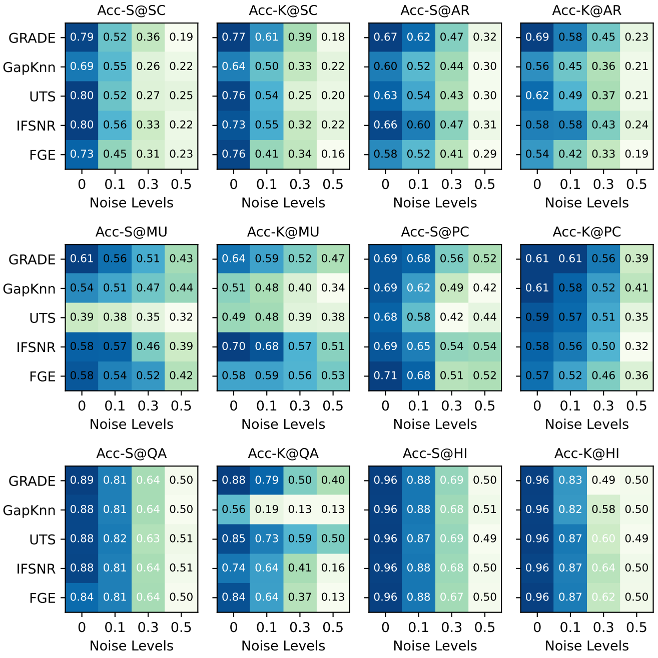

Fig. 2: Performance in terms of Acc-S and Acc-K with competitive methods on SC, AR, MU, PC, QA, and HI datasets under different noise ratios. For the heatmap, darker colors indicate better performance.

C. Noise Robustness Evaluation

To further assess GRADE's performance under noisy conditions, a controlled noise injection experiment is conducted with noise ratios of 0.0, 0.1, 0.3, and 0.5. GRADE is compared with four competitive counterparts. Noise is introduced as follows. For numerical features, Gaussian noise proportional to each feature's standard deviation is added. For categorical features and labels, a subset of samples is randomly replaced with other valid values to mimic entry errors and mislabeling. As shown in Fig. 2, GRADE maintains competitive accuracy under mild noise. This resilience stems from the IGUM metric, which buffers minor corruptions by leveraging distributional statistics, and the ADNR neighborhood relation, which isolates sparse noise through density guidance. At severe noise levels, the performance of the evaluated methods deteriorates sharply as the underlying data distributions become unrecognizable. This suggests that when noise severely distorts the data, the discriminative relationships underlying feature selection are invalid. This also motivates further exploration into extreme-noise scenarios. GRADE thus demonstrates robust performance under realistic noise, confirming its practical utility.

D. Visualization of Datasets with Complex Class Distributions

To intuitively demonstrate the challenges posed by complex data distribution, t-SNE visualization is conducted on four

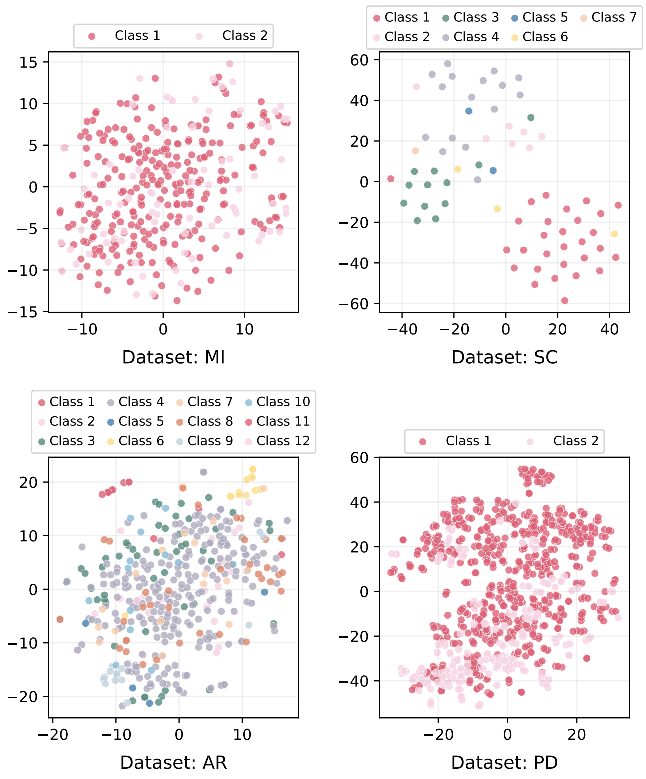

Fig. 3: t-SNE visualization of four mixed datasets.

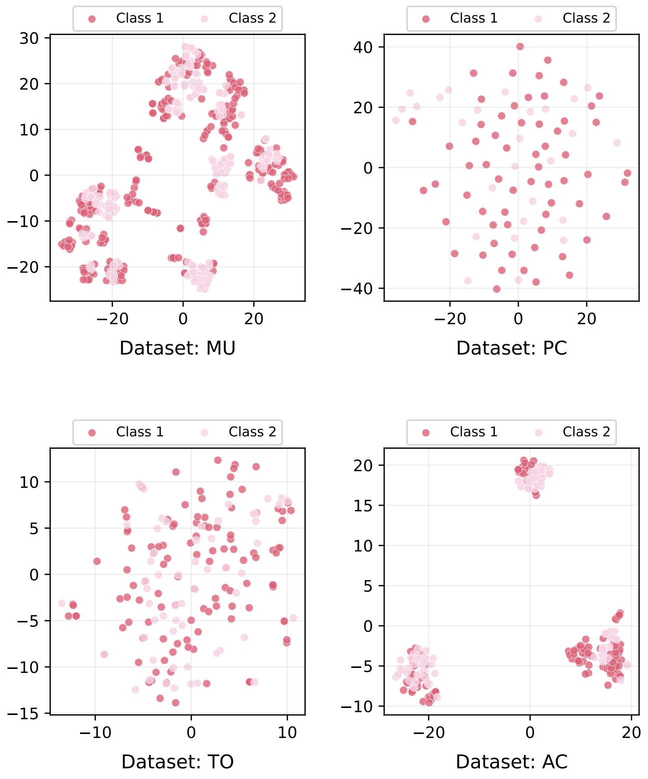

Fig. 4: t-SNE visualization of four numerical datasets.

mixed and four numerical datasets used in the experiments. The mixed datasets are preprocessed with one-hot encoding for categorical features and standardization for numerical features before two-dimensional t-SNE projection. The visualization results are shown in Fig. 3 and Fig. 4, respectively.

For mixed datasets shown in Fig. 3, the AR dataset exhibits the highest complexity with 12 overlapping classes and intricate non-linear boundaries, posing significant classification challenges. The SC dataset shows moderate complexity with seven dispersed classes. Even for the binary-class datasets, i.e., MI and PD, substantial class overlap and dispersion exist. These findings corroborate the observations in Section IV-D of the main paper, especially the superior performance of GRADE on the AR and SC datasets, justifying the importance of the proposed IGUM metric and ADNR neighborhood relation in capturing complex data distributions.

For numerical datasets shown in Fig. 4, the AC dataset exhibits relatively clear separation between classes, yet presents a multi-cluster structure within the binary classification. In contrast, TO and PC datasets demonstrate higher complexity with substantial class overlap, posing greater challenges for classification. The MU dataset shows complexity with partial class separation but noticeable overlap. These visual patterns are consistent with the analysis in Section IV-D of the main paper. That is, the density-guided ADNR neighborhood relation adapts to ambiguous regions, as presented in the PC and TO datasets, and to the multi-cluster structures of the AC dataset, outperforming the traditional k-nearest neighborhood relation approach.