Tabular Deep Learning vs Classical Machine Learning for Urban Land Cover Classification

[1]Muntasir Tabasum
Department of Geology and Geography
West Virginia University
Morgantown, WV 26506, USA
mt00079@mix.wvu.edu

[2]Tanpia Tasnim
Department of CSE
Green University of Bangladesh
Narayanganj-1461, Bangladesh
tanpia@cse.green.edu.bd

[3]Md. Ekramul Islam
Department of CSE
Stamford University Bangladesh
Dhaka-1217, Bangladesh
eislam706@gmail.com

[4]Al Zadid Sultan Bin Habib*
Lane Department of CSEE
West Virginia University
Morgantown, WV 26506, USA
ah00069@mix.wvu.edu

Abstract

Urban Land Cover (ULC) classification plays a crucial role in urban planning, environmental monitoring, and sustainable development. We study this task using the ULC dataset from the UCI Machine Learning Repository, which includes tabular features derived from high-resolution aerial imagery across nine classes (e.g., roads, trees, grass, water). The dataset presents typical remote sensing challenges, including high dimensionality, heterogeneous features, and class imbalance. In a unified, reproducible pipeline, we benchmark classical machine learning models (e.g., Logistic Regression, SVM, Random Forest, XGBoost, CatBoost) against Tabular Deep Learning (TDL) models (TabNet, FT-Transformer, TabTransformer, TabSeq, and 1D CNNs). To address class imbalance, we employ weighted cross-entropy loss for TDL models and evaluate performance using accuracy, macro-precision, macro-recall, macro-F1, AUC-ROC, and confusion matrices. Our results show that while tree ensembles remain strong general baselines, TDL models can match or exceed their performance when non-linear interactions are significant and imbalance handling is effective, providing complementary advantages for urban land cover mapping. See code: https://github.com/mtesha/tdl-vs-ml-urbanlandcover

1 Introduction

Urban Land Cover (ULC) classification supports sustainable development, urban planning, and environmental management by charting the geographical distribution and dynamics of essential surface types in urban areas (e.g., impervious surfaces, vegetation, and water). Accurate ULC maps support downstream analyses such as heat-island mitigation, stormwater management, and land-use monitoring. In this work, we examine ULC classification using the ULC dataset from the UCI Machine Learning Repository [Johnson, 2013], which provides tabular descriptors derived from a high-resolution aerial image. The features span multiple descriptor families (e.g., spectral, textural, shape/size) and nine classes (e.g., roads, trees, grass, water), and present typical remote-sensing challenges: high dimensionality, heterogeneous feature types, and class imbalance.

*Corresponding author. Website: https://www.zadidhabib.com.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: The 5th Muslims In ML (MusIML).

We build robust traditional Machine Learning (ML) baselines, including Logistic Regression (LR) [Rao, 1973], Decision Trees [Quinlan, 1986], Random Forests (RF) [Breiman, 2001], Support Vector Machines (SVM) [Cortes and Vapnik, 1995], k-Nearest Neighbors (KNN) [Cover and Hart, 1967], Naive Bayes [Duda and Hart, 1973], and boosting methods such as AdaBoost [Freund and Schapire, 1997], Gradient Boosting [Friedman, 2001], XGBoost [Chen and Guestrin, 2016], and CatBoost [Prokhorenkova et al., 2018], and compare them to some of the Tabular Deep Learning (TDL) models: TabNet [Arik and Pfister, 2021], TabTransformer [Huang et al., 2020, Wang, 2020], FT-Transformer and related modern TDL baselines [Gorishniy et al., 2021], as well as TabSeq [Habib et al., 2024, Habib, 2024] and a 1D CNN variant [LeCun et al., 1998]. To mitigate class imbalance in TDL, we use weighted cross-entropy.

Our study is motivated by mixed evidence in the literature on when TDL provides gains over tree ensembles for tabular problems. In remote sensing and urban mapping, the integration of diverse features and the management of imbalance are paramount [Johnson and Iizuka, 2016, Johnson, 2015, Wickham et al., 2014, Jozdani et al., 2019, Jin et al., 2019, Ducey et al., 2018]. We therefore ask: (i) How do strong classical ML methods compare to TDL on ULC? (ii) Under what conditions (e.g., non-linear interactions, imbalance handling) do TDL models match or surpass tree ensembles? (iii) What practical recommendations emerge for reproducible ULC pipelines?

Contributions. (1) A unified, reproducible pipeline benchmarking strong ML baselines against some of the TDL models on UCI ULC [Johnson, 2013]. (2) A systematic evaluation with accuracy, macro-precision/recall/F1, AUC-ROC, and confusion matrices. (3) An analysis of imbalance mitigation (weighted cross-entropy) and when representation learning in TDL closes or exceeds the performance of tree ensembles, yielding actionable guidance for urban mapping practitioners.

2 Related Work

ULC mapping has been a long-standing focus in remote sensing for monitoring urban expansion and environmental change. Methodological advances span data fusion, object-based analysis, and large-scale national products. For rapid land use/land cover mapping, [Johnson and Iizuka, 2016] integrate OpenStreetMap (OSM) with Landsat time-series and report strong performance with classical classifiers (e.g., Random Forest, Naive Bayes), while noting challenges from noise and class imbalance. At the descriptor level, [Johnson, 2015] propose spatially-weighted segment-level fusion (SWSF) to better aggregate low-spatial-resolution signals, improving small/narrow class discrimination when the coarse resolution is $\geq 3\times$ the high-resolution reference. At national scale, the MRLC consortium underpins NLCD and related products (C-CAP, CDL, LANDFIRE), leveraging Landsat to deliver consistent land cover layers for the United States that enable diverse environmental applications [Wickham et al., 2014, Jin et al., 2019]. Beyond mapping itself, demographic land cover interactions have been quantified statistically; for the U.S. Great Lakes States, [Ducey et al., 2018] show population growth, housing density, and recreational/retirement status as key drivers of land conversion.

Comparisons of modeling paradigms for urban LULC remain mixed. In an object-based setting, [Jozdani et al., 2019] find that a multilayer perceptron can outperform gradient boosting, XGBoost, and SVMs, while CNN integrations offer limited gains under complex urban morphology and segmentation errors. In parallel, tabular-learning research has introduced modern deep models for mixed-type features (e.g., TabNet, TabTransformer, FT-Transformer) alongside strong tree ensembles [Arik and Pfister, 2021, Huang et al., 2020, Gorishniy et al., 2021, Chen and Guestrin, 2016, Prokhorenkova et al., 2018, Breiman, 2001]. However, few studies systematically benchmark some of the recent TDL architectures against well-tuned classical baselines on tabular ULC descriptors where heterogeneity, high dimensionality, and class imbalance are prominent. Earlier UHI-based land cover studies relied on Landsat imagery and GIS-based statistical analyses but did not use modern AI/ML/TDL methods for ULC change classification [Ritu, 2023, Ritu and Bruce, 2025]. A Khulna traffic-delay study similarly used survey/GIS and speed-flow analysis but did not employ modern ML/TDL prediction pipelines [Bashar et al., 2020].

Our focus. We address this gap by evaluating strong classical ML methods and recent TDL models within a unified, reproducible pipeline on the UCI ULC dataset, with attention to imbalance mitigation and multi-metric reporting. This complements prior work on fused imagery and object-based analysis by centering the tabular descriptor regime common in operational ULC workflows.

3 Methodology

Dataset. We utilize the ULC dataset from the UCI Machine Learning Repository [Johnson, 2013]. It provides engineered tabular descriptors from a single high-resolution aerial image with samples of n_{train}=168, n_{test}=507, d=147 features, and K=9 classes (asphalt, building, car, concrete, grass, pool, shadow, soil, tree), exhibiting heterogeneous descriptors and class imbalance.

Preprocessing. Labels are integer-encoded $\{1, \ldots, K\}$. Features are z-scored using training statistics, i.e., $\tilde{x}_{ij} = (x_{ij} - \mu_j)/\sigma_j$ with (μ_j, σ_j) from X_{train} and reused for validation/test. A stratified 10% of X_{train} is held out for validation.

Models. *Classical ML and GBDT:* Logistic Regression (LR) [Rao, 1973], Decision Tree (DT) [Quinlan, 1986], Random Forest (RF) [Breiman, 2001], SVM [Cortes and Vapnik, 1995], kNN [Cover and Hart, 1967], Naive Bayes (NB) [Duda and Hart, 1973], Gradient Boosting (GBM) [Friedman, 2001], AdaBoost [Freund and Schapire, 1997], XGBoost [Chen and Guestrin, 2016], CatBoost [Prokhorenkova et al., 2018], and a shallow MLP [Rumelhart et al., 1986]. *TDL:* TabNet [Arik and Pfister, 2021], TabTransformer [Huang et al., 2020, Wang, 2020], FT-Transformer [Gorishniy et al., 2021], a 1D CNN [LeCun et al., 1998], and TabSeq [Habib et al., 2024, Habib, 2024].

Discussion. Tree ensembles (RF, GBM, XGBoost, CatBoost) are strong defaults on heterogeneous tabular data. They capture non-linearities and higher-order interactions with modest tuning, tolerate mixed feature scales, and are comparatively robust on small-n regimes like ULC. CatBoost additionally handles categorical variables and reduces target leakage via ordered boosting, which often improves calibration and minority-class recall. Linear/margin methods (LR, SVM) and kNN/NB provide interpretable, low-variance references that help contextualize gains from more complex models. The shallow MLP probes whether simple neural capacity suffices without tabular-specific inductive bias. TDL models introduce representation learning tailored to feature interactions (transformers) or sparse attentive selection (TabNet); however, they typically require stronger regularization and benefit from imbalance-aware training to avoid overfitting at this scale. The 1D CNN imposes a sequential inductive bias over features (useful for local groups of correlated descriptors but sensitive to feature order), whereas TabSeq explicitly learns a stable ordering and denoised representation before classification. Together, the suite spans linear $\rightarrow$ tree $\rightarrow$ neural paradigms, enabling a controlled comparison under identical preprocessing, validation, and metrics.

Training, loss, and metrics. All models train on $(X_{\text{train}}^{\text{scaled}}, y_{\text{train}})$, select by validation loss/accuracy, and report on $(X_{\text{test}}^{\text{scaled}}, y_{\text{test}})$. For TDL we use class-weighted cross-entropy $\mathcal{L}_{\text{wCE}} = -\frac{1}{N}\sum_{i=1}^{N} w_{y_i} \log p_\theta(y_i \mid x_i)$ with $w_k = \frac{N}{KN_k}$, where N_k counts class-k training samples; tree ensembles are left unweighted. We report test accuracy; macro-precision/recall/F1; one-vs-rest macro AUC-ROC; and confusion matrices for error analysis.

Key hyperparameters. Optimizer: Adam, batch size 32, learning rate 10^{-3}, early stopping on validation loss. *TabNet* [Arik and Pfister, 2021]: n_d=64, n_a=64, n_{steps}=3, γ=1.3, λ_{sparse}=10^{-3}, lr 2×10^{-2}, patience 20. *TabTransformer* [Huang et al., 2020, Wang, 2020]: dim 32, depth 6, heads 8, attn/ffn dropout 0.1, MLP multipliers $(4, 2)$, lr 10^{-3}, weighted cross-entropy. *FT-Transformer* [Gorishniy et al., 2021]: dim 32, depth 6, heads 8, dropout 0.1. *1D CNN* [LeCun et al., 1998]: conv (64 ch, k=3) $\rightarrow$ max-pool (k=2, stride 2) $\rightarrow$ FC(128)$\times$2 with dropout 0.3 $\rightarrow$ softmax. *TabSeq* [Habib et al., 2024, Habib, 2024]: feature ordering via k-means (k=5) + dispersion minimization; DAE encoder/decoder $\{128, 64\}$ with dropout 0.2 and noise 0.1; classifier MLP $\{128, 64\}$ with dropout 0.5. Optuna Akiba et al. [2019] was used to tune the hyperparameters of the TDL models, while the classical ML baselines followed scikit-learn defaults, and the GBDT models used their respective library defaults (e.g., CatBoost [Prokhorenkova et al., 2018], XGBoost [Chen and Guestrin, 2016]).

Implementation. Python/Scikit-learn for classical models; PyTorch/TensorFlow for TDL; fixed seeds and persisted preprocessing stats for reproducibility.

4 Results and Analysis

Setup. We evaluate 16 models (classical ML and TDL) on the UCI ULC test set and report Test Accuracy (%), macro Precision/Recall/F1, and one-vs-rest macro AUC-ROC. Summary tables (Tabs. 1–2) are complemented by six visualizations (Fig.1 and Figs. A3–A7 in Appendix A.1).

Baseline performance (all models). CatBoost attains the highest test accuracy (82.25%), followed by Random Forest (81.66%) and Naive Bayes (77.51%), confirming the strength of tree ensembles and simple probabilistic baselines on tabular descriptors. Among TDL models, TabTransformer (74.75%) and TabSeq (73.96%) are competitive with the best non-ensemble classical baselines. In Appendix A.1, Fig. A3 contrasts accuracies across all models; Fig. A4 compares Precision/Recall/F1/AUC for top contenders; Fig. A5 shows a full metric heatmap, highlighting trade-offs (e.g., models with similar accuracy but different Recall/AUC). Fig. 1 focuses on the five deep models. Table 2 presents the full results. See Appendix A.3 for ROC curves and confusion matrices for all models.

TDL with weighted cross-entropy. To address class imbalance, we train 1D-CNN, TabNet, FT-Transformer, TabTransformer, and TabSeq with class-weighted cross-entropy. Accuracy results (Fig. A6 in Appendix A.1) show TabSeq (73.57%) and TabTransformer (73.37%) leading this group. Fig. A7 in Appendix A.1 also indicates consistent gains in Recall and F1 for most models under weighting, echoing improvements noted in Table 1. See Appendix A.2 for ROC curves and confusion matrices for TDL models with weighted cross entropy.

Findings. (1) *Ensemble dominance.* CatBoost/RF remain strong on heterogeneous tabular features. (2) *TDL potential.* Attention-based TDL (TabTransformer/TabSeq) are competitive and close the gap with better imbalance handling. (3) *Weighted loss helps.* Class-weighted cross-entropy generally boosts Recall/F1, improving minority-class sensitivity without large AUC degradation. (4) *Metric trade-offs.* Accuracy alone obscures differences; macro Recall/F1 and AUC are crucial for imbalanced ULC.

Table 1: TDL models with class-weighted cross-entropy.

Model	Acc.%	Prec.	Rec.	F1	AUC
1D CNN	73.18	0.70	0.74	0.71	0.94
TabNet	66.67	0.65	0.70	0.66	0.93
FT-Transformer	72.39	0.66	0.72	0.68	0.93
TabTransformer	73.37	0.69	0.74	0.71	0.96
TabSeq	73.57	0.70	0.74	0.71	0.95

Table 2: Performance of different models on ULC (test set).

Model	Test Acc. (%)	Precision	Recall	F1	AUC
Logistic Regression [Rao, 1973]	70.61	0.70	0.73	0.71	0.94
Random Forest [Breiman, 2001]	81.66	0.81	0.83	0.81	0.97
SVM [Cortes and Vapnik, 1995]	68.44	0.67	0.71	0.68	0.96
Decision Tree [Quinlan, 1986]	74.75	0.72	0.75	0.73	0.96
KNN [Cover and Hart, 1967]	70.22	0.69	0.69	0.69	0.93
Naive Bayes [Duda and Hart, 1973]	77.51	0.76	0.78	0.76	0.96
MLP [Rumelhart et al., 1986]	69.63	0.65	0.70	0.67	0.97
CatBoost [Prokhorenkova et al., 2018]	**82.25**	0.81	0.82	0.82	0.98
AdaBoost [Freund and Schapire, 1997]	63.12	0.67	0.62	0.64	0.93
XGBoost [Chen and Guestrin, 2016]	68.05	0.69	0.64	0.64	0.95
GB [Friedman, 2001]	72.78	0.70	0.72	0.71	0.93
1D CNN [LeCun et al., 1998]	73.37	0.73	0.75	0.75	0.97
TabNet [Arik and Pfister, 2021]	64.10	0.57	0.60	0.57	0.91
FT-Transformer [Gorishniy et al., 2021]	73.18	0.70	0.74	0.70	0.93
TabTransformer [Huang et al., 2020]	74.75	0.71	0.72	0.72	0.96
TabSeq [Habib et al., 2024]	73.96	0.70	0.74	0.74	0.95

Statistical comparison of model accuracies. Table 2 reports single-run test accuracies for all baselines, and Figure 2 visualizes the same results with 95% binomial confidence intervals computed from the test-set size. Because the intervals for many methods overlap, the evidence for large performance gaps on this single split is limited. Even so, the plot indicates that CatBoost, Random Forest, and Naive Bayes form the strongest group, with CatBoost achieving the highest observed accuracy. Since per-instance predictions or multiple randomized splits were not available, we used an approximate two-proportion (unpaired) comparison rather than a paired test such as McNemar or Friedman.

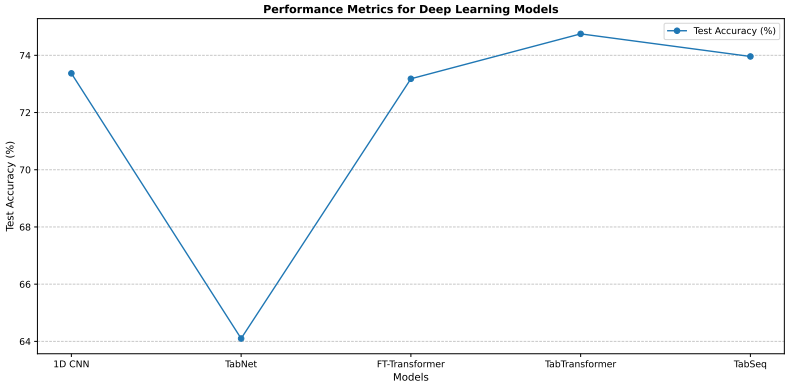


Figure 1: Deep-model accuracies (TDL only). Remaining visualizations appear in Appendix A.1.

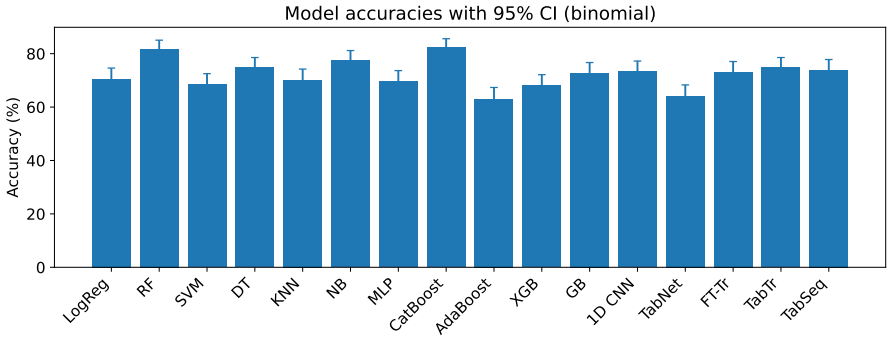


Figure 2: Model accuracies on the ULC test set with 95% binomial confidence intervals.

Future direction. For future work, we will extend this study to a broader set of tabular benchmarks and develop a tabular Transformer-style model for remote-sensing and flood-related tabular datasets to better handle heterogeneous features and limited labels. This direction directly addresses the current single-dataset and novelty concerns and further clarifies the role of tabular deep learning for ULC. The same pipeline can also be applied in Muslim-majority or rapidly urbanizing cities (e.g., Dhaka, Jakarta, Lahore, Cairo), where recent urban land-cover products are scarce.

5 Conclusion

We evaluated robust classical ML models alongside newer TDL techniques for ULC classification using the UCI ULC dataset. Tree ensembles especially CatBoost and Random Forest achieved the highest accuracy and competitive macro metrics, reaffirming their suitability for heterogeneous, imbalanced tabular features. Nevertheless, TDL models (TabTransformer, TabSeq) were competitive and improved further with class-weighted losses, indicating clear potential when representation learning and imbalance handling are aligned with data characteristics. Practically, we recommend ensembles as robust baselines and TDL as complementary models when non-linear feature interactions and minority-class sensitivity are priorities. Limitations include a single-dataset scope and modest sample size; future work will expand to additional ULC benchmarks, explore stronger calibration/uncertainty modeling, and refine TDL architectures and sampling/weighting strategies to close the remaining gap.

Acknowledgments

We thank Dr. Aaron Maxwell, Associate Professor in the Department of Geology and Geography at West Virginia University, USA, for his insightful feedback on this project. See his webpage: https://www.wvview.org/maxwell.html.

References

B. Johnson. Urban Land Cover (ULC) Data Set. UCI Machine Learning Repository, 2013. URL `https://archive.ics.uci.edu/dataset/295/urban+land+cover`. Accessed 2025-09-18.

C. Radhakrishna Rao. *Linear Statistical Inference and Its Applications*. Wiley, 1973.

J. Ross Quinlan. Induction of Decision Trees. *Machine Learning*, 1(1):81–106, 1986. doi: 10.1007/BF00116251.

Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001. doi: 10.1023/A:1010933404324.

Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995. doi: 10.1007/BF00994018.

Thomas Cover and Peter Hart. Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967. doi: 10.1109/TIT.1967.1053964.

Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis*. Wiley, 1973.

Yoav Freund and Robert E. Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997. doi: 10.1006/jcss.1997.1504.

Jerome H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*, 29(5):1189–1232, 2001. doi: 10.1214/aos/1013203451.

Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016. doi: 10.1145/2939672.2939785.

Liudmila Prokhorenkova et al. CatBoost: Unbiased Boosting with Categorical Features. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Sercan Ö. Arik and Tomas Pfister. TabNet: Attentive Interpretable Tabular Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6679–6687, 2021.

Xin Huang et al. TabTransformer: Tabular Data Modeling Using Contextual Embeddings. *arXiv preprint arXiv:2012.06678*, 2020. URL `https://arxiv.org/abs/2012.06678`.

Phil Wang. TabTransformer and FT-Transformer: Code Repository. GitHub, 2020. URL `https://github.com/lucidrains/tab-transformer-pytorch`.

Yury Gorishniy et al. Revisiting Deep Learning Models for Tabular Data. In *Advances in Neural Information Processing Systems*, volume 34, pages 18932–18943, 2021.

Al Zadid Sultan Bin Habib et al. TabSeq: A Framework for Deep Learning on Tabular Data via Sequential Ordering. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, 2024. Springer LNCS.

Al Zadid Sultan Bin Habib. TabSeq: Code Repository. GitHub, 2024. URL `https://github.com/zadid6pretam/TabSeq`.

Yann LeCun et al. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Bryan A. Johnson and Kazuhiro Iizuka. Integrating OpenStreetMap Crowdsourced Data and Landsat Time-Series Imagery for Rapid Land Use/Land Cover Mapping. *Applied Geography*, 67:140–149, 2016. doi: 10.1016/j.apgeog.2015.12.006.

Bryan Johnson. Remote Sensing Image Fusion at the Segment Level Using a Spatially-Weighted Approach: Applications for Land Cover Spectral Analysis and Mapping. *ISPRS International Journal of Geo-Information*, 4(1):172–184, 2015. doi: 10.3390/ijgi4010172.

James Wickham et al. The Multi-Resolution Land Characteristics (MRLC) Consortium—20 Years of Development and Integration of USA National Land Cover Data. *Remote Sensing*, 6(8):7424–7441, 2014. doi: 10.3390/rs6087424.

Shahrokh E. Jozdani et al. Comparing Deep Neural Networks, Ensemble Classifiers, and Support Vector Machine Algorithms for Object-Based Urban Land Use/Land Cover Classification. *Remote Sensing*, 11(14):1713, 2019. doi: 10.3390/rs11141713.

Suming Jin et al. Overall Methodology Design for the United States National Land Cover Database 2016 Products. *Remote Sensing*, 11(24):2971, 2019. doi: 10.3390/rs11242971.

Mark J. Ducey et al. The Influence of Human Demography on Land Cover Change in the Great Lakes States, USA. *Environmental Management*, 62(6):1089–1107, 2018. doi: 10.1007/s00267-018-1102-x.

Sadia Islam Ritu. *Impacts of Land Cover Change on Urban Heat Island (UHI) in Denver from 1985 to 2020*. South Dakota State University, 2023.

Sadia Islam Ritu and Millett Bruce. Three Decades of Changes in the Urban Heat Island Effect in Denver, Colorado, Revealed by Landsat. *Journal of Earth Observations and Geospatial Applications*, 1(1):64–82, 2025.

TM Junaid Bashar et al. Finding the Reasons for the Delay Time in A Highway by Analyzing the Travel Time, Delay Time and Traffic Flow Data. *Journal of Engineering Advancements*, 1(03): 76–84, 2020.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986. doi: 10.1038/323533a0.

Takuya Akiba et al. Optuna: A Next-Generation Hyperparameter Optimization Framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631, 2019.

A Technical Appendices and Supplementary Material

This supplementary material accompanies our main paper entitled *Tabular Deep Learning vs Classical Machine Learning for Urban Land Cover Classification* (submitted to the 5th Muslims In ML (MusIML) Workshop co-located with NeurIPS 2025). It provides additional experimental results that complement the main text.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state that the paper (i) benchmarks strong classical ML models against Tabular Deep Learning (TDL) models on the UCI Urban Land Cover (ULC) dataset, (ii) uses a unified, reproducible pipeline, and (iii) analyzes metric-level differences (accuracy, macro F1, AUC). These are exactly the contributions developed in the main sections, without overclaiming generalization to large multimodal or multisensor settings.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly note that experiments are conducted on a single public UCI ULC dataset with a modest test split, and that we do not report paired significance tests because per-instance predictions or multiple randomized splits were not available. We also mention that future work will extend the study to additional remote-sensing and flood-related tabular benchmarks and stronger TDL variants.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: This paper is empirical/benchmarking in nature (TDL vs classical ML for ULC) and does not introduce new theoretical results or formal proofs. We therefore state the modeling assumptions (dataset, metrics, splits, baselines) but no completeness of proofs is required.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper specifies the dataset (UCI ULC), the train/validation/test split, the set of classical ML and TDL baselines, the evaluation metrics (accuracy, macro precision/recall/F1, AUC), and the main training settings for TDL models; it also provides the public GitHub link for code and scripts, enabling independent reproduction of the reported tables and plots.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The dataset used in this study (UCI ULC) is publicly available, and we provide an open GitHub repository containing the training/evaluation scripts and configuration files needed to reproduce the reported results: `https://github.com/mtesha/tdl-vs-ml-urbanlandcover`.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

Justification: We describe the UCI ULC dataset, the train/validation/test split, preprocessing (scaling using train statistics), the classical ML vs TDL model list, optimizer and loss settings for TDL (Adam, weighted CE), and the use of Optuna for TDL hyperparameter tuning, so the evaluation protocol and reported tables can be fully understood.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report model accuracies with 95% binomial confidence intervals and include an approximate two-proportion comparison against the top model on the single test split. We also state the limitation that paired tests (e.g., McNemar, Friedman/Nemenyi) are not applicable without per-instance predictions or multiple randomized splits.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We specify the software stack (scikit-learn, PyTorch/TensorFlow), dataset size and splits, batch size and optimizer settings, and note that all runs are reproducible on a standard workstation (CPU or a single commodity GPU). Given the modest scale of UCI ULC, exact runtime is negligible and not required to reproduce the results.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics (`https://neurips.cc/public/EthicsGuidelines`)?

   Answer: [Yes]

   Justification: The work uses a public, non-PII remote-sensing/tabular dataset (UCI ULC), cites it properly, and targets urban/environmental analytics with no foreseeable harmful use. No human subjects, private data, or sensitive attributes are involved, and results are reported transparently.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Our study targets ULC classification from tabular/remote sensing descriptors, which can support urban planning, environmental monitoring, and service provision in data-scarce cities, including Muslim-majority and rapidly urbanizing regions. At the same time, any automated land/asset mapping pipeline could be misused for inequitable resource allocation or area-level surveillance if applied without transparency or local validation, so we recommend responsible, policy-aware deployment.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We use a small, public, non-sensitive UCI ULC dataset and release task-specific tabular classification code; no large pretrained or generative models, scraped data, or high-risk assets are introduced, so additional safeguards are not required.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We use the publicly available UCI ULC dataset and standard open-source libraries (scikit-learn, PyTorch), all of which are properly cited and used within their research/OSS licenses. We do not redistribute modified versions of proprietary assets.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We release a lightweight, documented codebase for reproducing the TDL vs classical ML experiments on the UCI ULC dataset (data loading, preprocessing, model configs, and evaluation). No new dataset is introduced; the only new asset is the reproducible pipeline, which is described in the paper and in the repository.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No crowdsourcing or human-subject data collection was conducted; we only used a public remote-sensing/tabular dataset.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The work uses only publicly available, non-PII remote-sensing/tabular data and does not involve human subjects, so IRB review was not required.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?

    Answer: [NA]

    Justification: LLMs were used only for minor language polishing and LaTeX formatting; they did not contribute to the core methodology, experiments, or results, so a formal declaration is not required.

A Technical Appendices and Supplementary Material

This supplementary document supports our main paper titled *Tabular Deep Learning vs Classical Machine Learning for Urban Land Cover Classification* (Submitted to the 5th Muslims In ML (MusIML) Workshop co-located with NeurIPS 2025). Specifically, it includes:

- Additional Figures for Comparative Results
- ROC Curves and Confusion Matrices for TDL Models with Weighted Cross Entropy
- ROC Curves and Confusion Matrices for All Models with Standard Cross Entropy

A.1 Additional Figures for Comparative Results

Fig. A3 (All-model accuracy). Overall leaderboard across 16 methods; tree ensembles (CatBoost, RF) top the chart, while most TDL models sit mid-pack.

Fig. A4 (Top-model multi-metric). Across precision/recall/F1/AUC, CatBoost and RF remain consistently strong with balanced precision-recall; TabTransformer trails slightly on F1 but is competitive on AUC.

Fig. A5 (Full metric heatmap). Side-by-side macro metrics reveal where models trade precision for recall; CatBoost/RF are uniformly high, while TabNet and AdaBoost show weaker recall/F1.

Fig. A6 (TDL with weighted CE accuracy). Adding class weights narrows gaps among deep models; TabSeq/TabTransformer edge out FT-Transformer and 1D-CNN, with TabNet still behind.

Fig. A7 (TDL with weighted CE multi-metric). Weighted loss mainly boosts recall and F1 while maintaining high AUC, indicating better minority-class sensitivity without sacrificing ranking quality.

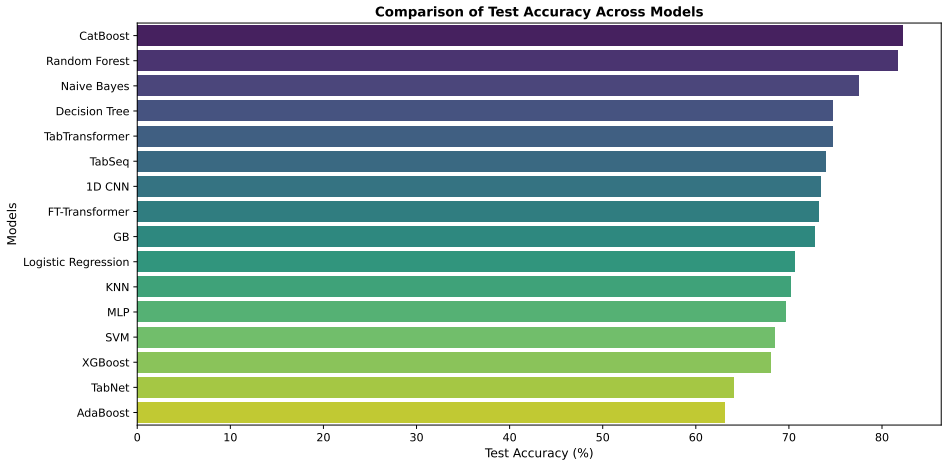


Figure A3: All models test accuracy.

A.2 ROC Curves and Confusion Matrices for TDL Models with Weighted Cross Entropy

Figure A8 (1D CNN, W). ROC curves cluster near the top-left, indicating strong separability for most classes; macro AUC is high with only a few classes showing shallower curves. The confusion matrix is diagonally dominant, with residual errors dispersed across a handful of visually similar categories. Weighted loss improves recall for minority classes without noticeably degrading well-separated ones.

Figure A9 (TabNet, W). ROC profiles are competitive overall but exhibit slightly flatter segments for some classes, consistent with greater sensitivity to class imbalance. The confusion matrix shows more off-diagonal mass than 1D CNN, reflecting occasional swaps among related built-surface or vegetation classes. Attention-based feature selection helps, but errors concentrate in harder pairs.

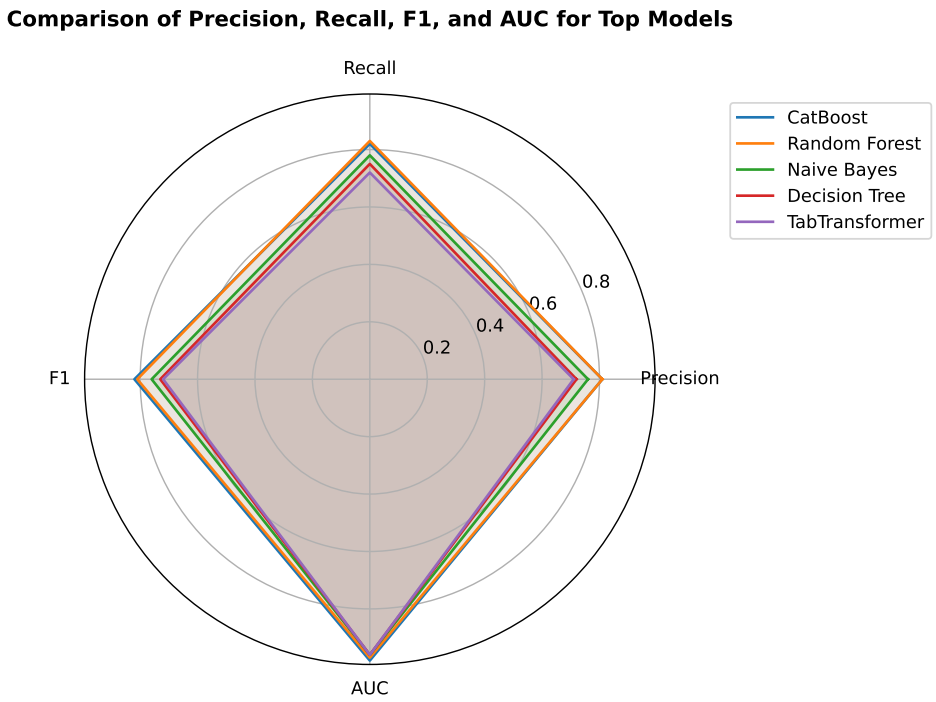


Figure A4: Top models: Precision/Recall/F1/AUC.

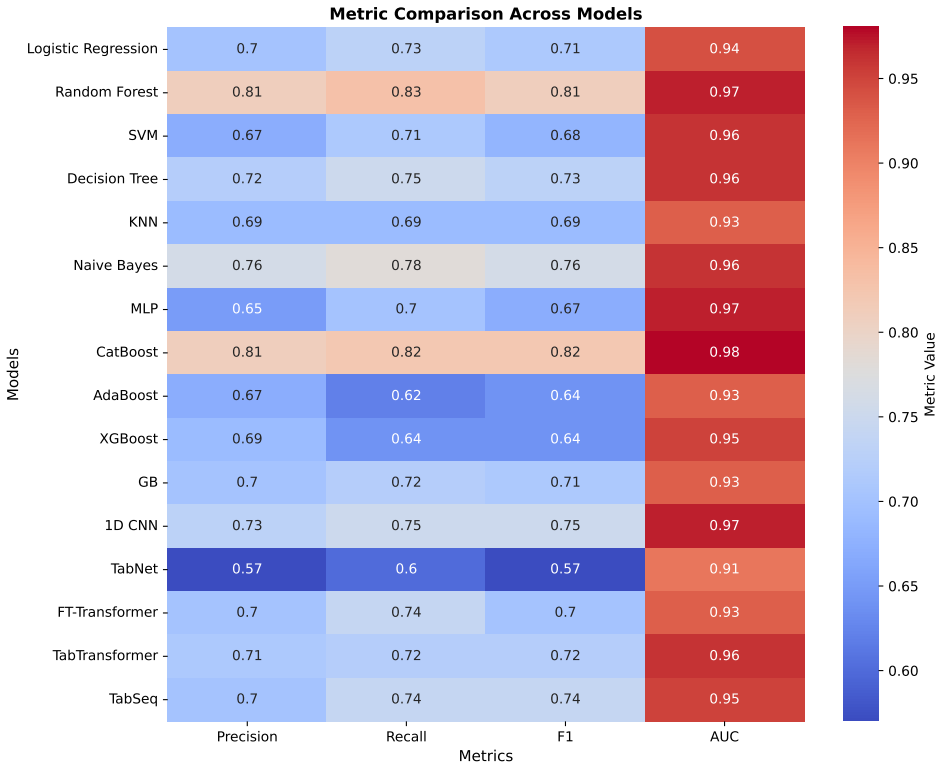


Figure A5: Metric heatmap for all models.

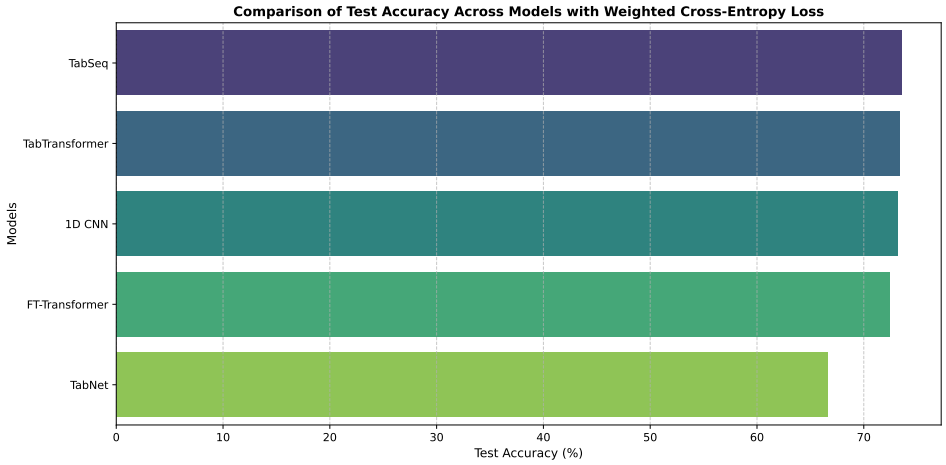


Figure A6: Weighted CE (TDL): test accuracy.

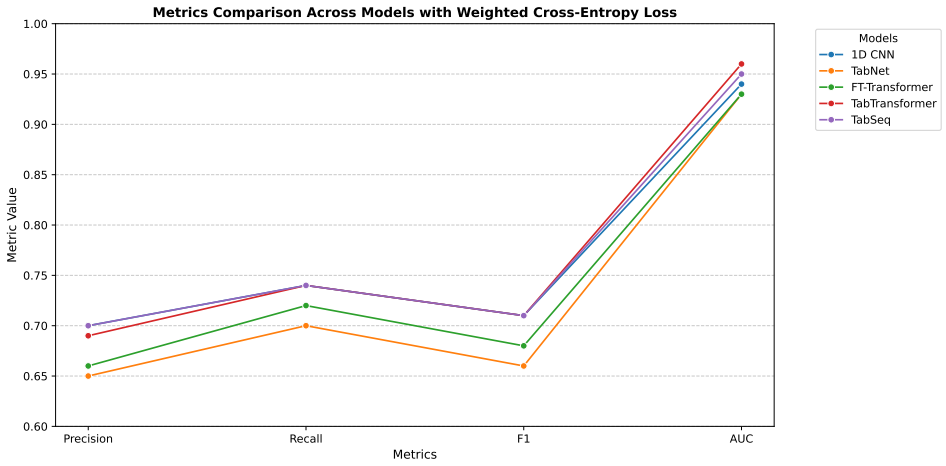


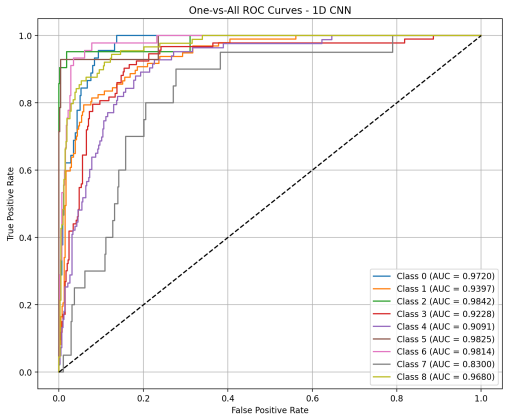
Figure A7: Effect of class-weighted cross-entropy on multi-metric performance (TDL).

Figure A10 (FT-Transformer, W). Transformer-based representations yield uniformly strong ROC curves with tight variance across classes. The confusion matrix remains largely diagonal, with misclassifications concentrated in a few class pairs suggesting remaining ambiguity in descriptors rather than systematic bias. Weighted CE chiefly raises recall on the rarer classes.

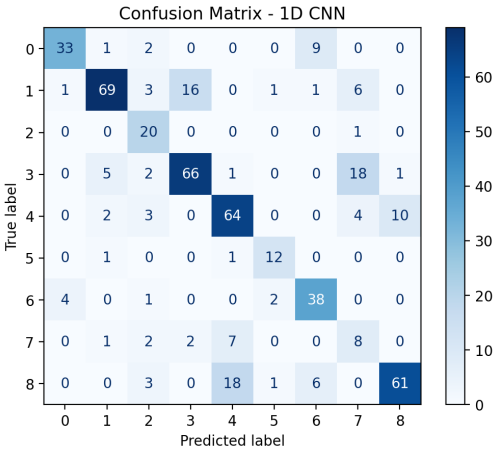
Figure A11 (TabTransformer, W). This model achieves some of the steepest ROC traces across classes, pointing to effective modeling of feature interactions. The confusion matrix shows high true-positive counts along the diagonal and fewer large off-diagonal cells, indicating balanced performance; remaining errors align with classes that are spectrally/structurally similar.

Figure A12 (TabSeq, W). ROC curves are consistently high; ordering-and-denoising prior to classification appears to aid class separability. The confusion matrix is cleanly diagonal with a small number of localized confusions, suggesting that TabSeq's learned feature ordering mitigates some overlap but challenging pairs persist.

Reading guide. Across models, AUC values near 1.0 denote strong one-vs-rest separability; diagonally dominant confusion matrices reflect balanced accuracy. Off-diagonal clusters typically indicate confusable, semantically related classes or minority-class scarcity; weighted cross-entropy mainly improves recall in these regions.

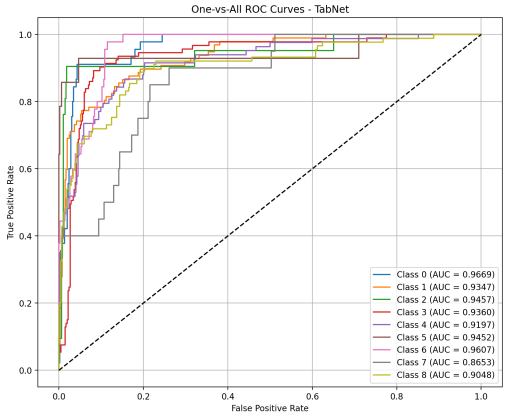


(a) ROC (1D CNN, W)

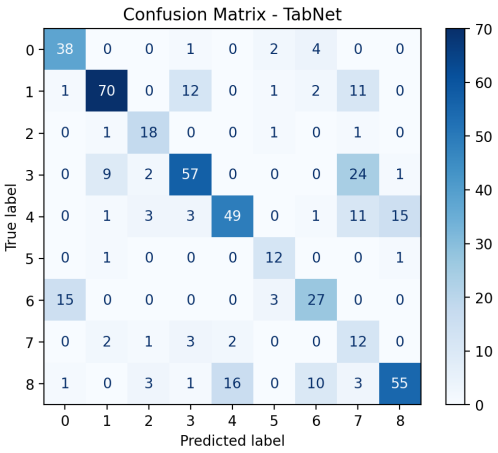


(b) Confusion matrix (1D CNN, W)

Figure A8: One-vs-all ROC and confusion matrix for 1D CNN with weighted cross-entropy.

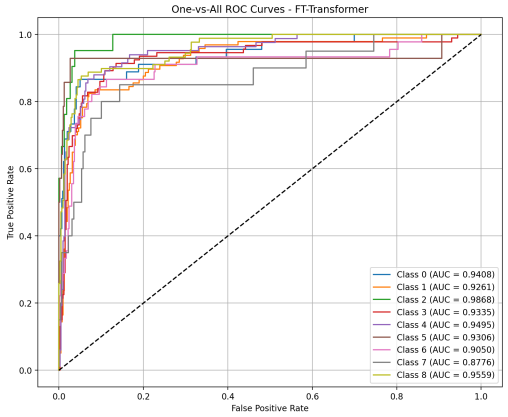


(a) ROC (TabNet, W)

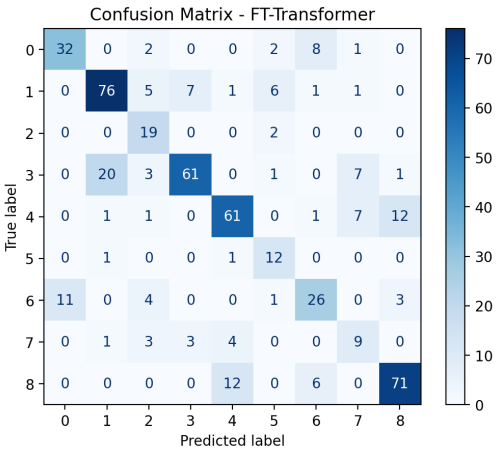


(b) Confusion matrix (TabNet, W)

Figure A9: One-vs-all ROC and confusion matrix for TabNet with weighted cross-entropy.

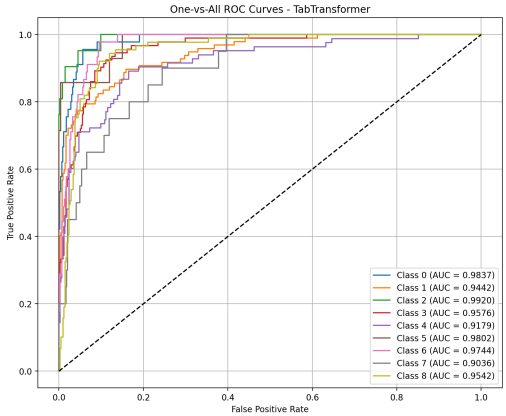


(a) ROC (FT-Transformer, W)

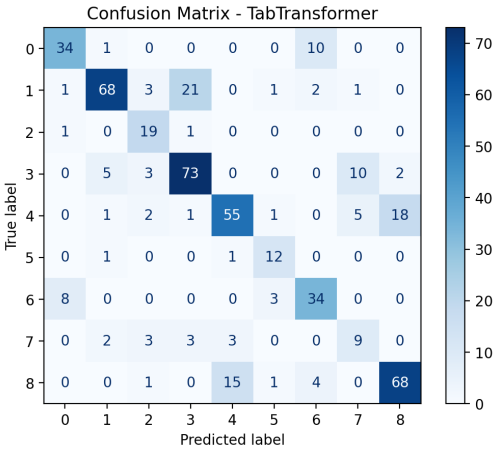


(b) Confusion matrix (FT-Transformer, W)

Figure A10: One-vs-all ROC and confusion matrix for FT-Transformer with weighted cross-entropy.

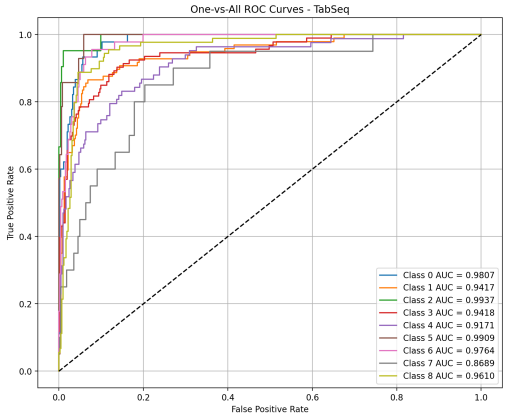


(a) ROC (TabTransformer, W)

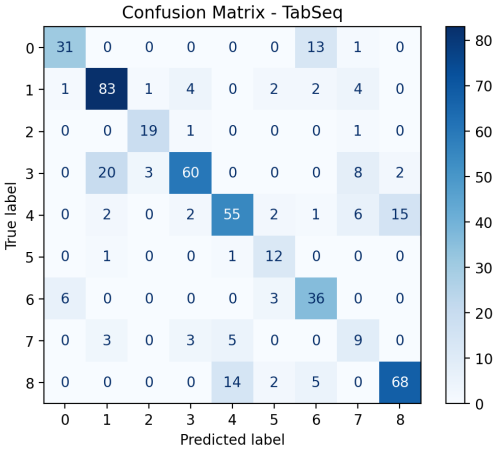


(b) Confusion matrix (TabTransformer, W)

Figure A11: One-vs-all ROC and confusion matrix for TabTransformer with weighted cross-entropy.



(a) ROC (TabSeq, W)



(b) Confusion matrix (TabSeq, W)

Figure A12: One-vs-all ROC and confusion matrix for TabSeq with weighted cross-entropy.

A.3 ROC Curves and Confusion Matrices for All Models with Standard Cross Entropy

Fig. A13 (TabSeq). High AUC across classes; the confusion matrix is largely diagonal with remaining errors concentrated in a few visually similar pairs consistent with TabSeq's denoising + ordering benefit on correlated features.

Fig. A14 (TabTransformer). Steep ROC traces indicate strong one-vs-rest separability; off-diagonal mass is modest and localized, suggesting effective modeling of feature interactions without class weighting.

Fig. A15 (FT-Transformer). Uniformly high ROC curves with slightly larger variance for minority classes; confusion matrix shows a few systematic swaps, hinting at remaining imbalance sensitivity under standard CE.

Fig. A16 (TabNet). Good but flatter ROC for some classes relative to transformer models; confusion patterns show occasional confusion among built-surface categories, aligning with attentive feature selection's sensitivity to imbalance.

Fig. A17 (1D CNN). Strong ROC overall; the confusion matrix is diagonal-dominant with clustered mistakes where descriptors overlap. Sequential bias can help local feature groups but remains sensitive to feature permutations.

Fig. A18 (GBM). High AUC across most classes; confusion matrix is diagonally dominant with residual errors in a few built-surface vs. soil confusions.

Fig. A19 (XGBoost). Consistently strong ROC traces and a clean confusion matrix; misclassifications are localized, reflecting robust handling of heterogeneous tabular features.

Fig. A20 (AdaBoost). ROC performance is competitive but shows larger variance for minority classes; confusion matrix reveals a few systematic swaps, typical for boosting with limited data.

Fig. A21 (CatBoost). Steep ROC curves with near-ceiling AUC on several classes; confusion matrix is the most diagonal among ensembles, aligning with CatBoost's strong overall accuracy.

Fig. A22 (MLP). Neural baseline shows high AUC but slightly more off-diagonal mass, indicating sensitivity to class imbalance and feature scaling compared to tree ensembles.

Fig. A23 (Naive Bayes). Consistently high AUCs and a largely diagonal matrix; remaining errors cluster among a few visually similar classes, reflecting conditional-independence limits.

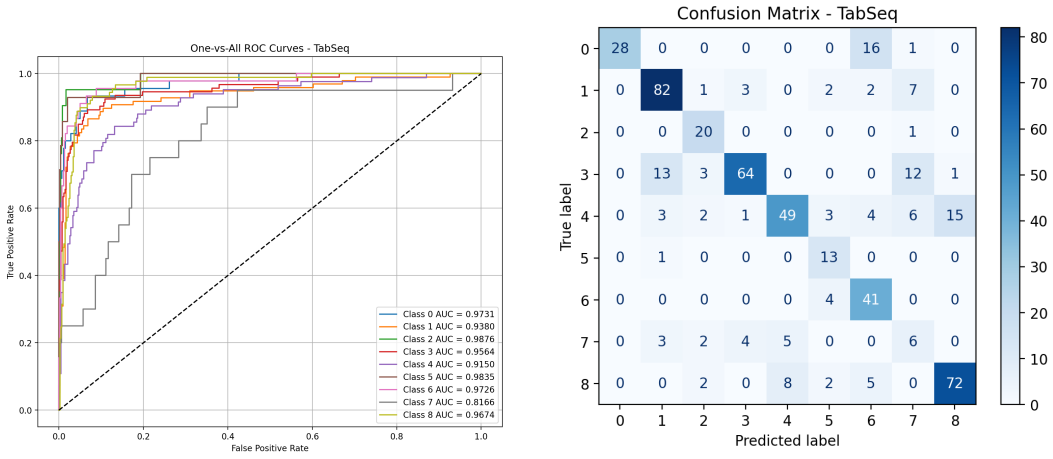
Fig. A24 (KNN). Competitive AUCs with thinner margins on several classes; confusion is diffuse rather than localized, typical of distance-based decision boundaries.

Fig. A25 (Decision Tree). Lower and more variable AUCs across classes; single-tree decision boundaries yield heavier off-diagonal mass, suggesting underfitting/instability relative to ensembles.

Fig. A26 (SVM). Strong one-vs-rest separability and clean diagonals for most classes; residual confusion is limited to a few neighboring categories, matching SVMs' large-margin behavior.

Fig. A27 (Random Forest). Steep ROC curves with near-uniform high AUC across classes; the confusion matrix is sharply diagonal, matching RF's strong accuracy and robustness on heterogeneous tabular features.

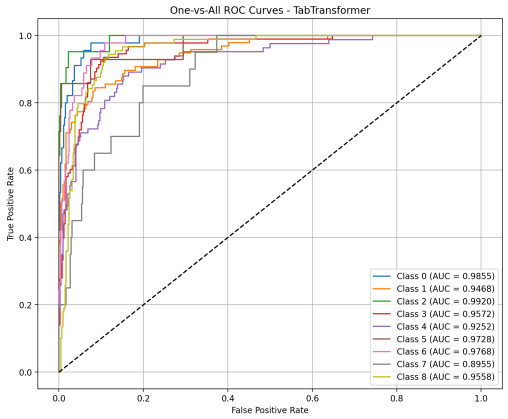
Fig. A28 (Logistic Regression). High AUC on several classes, but the confusion matrix shows increased spillover among visually similar categories, consistent with linear boundaries under class overlap.



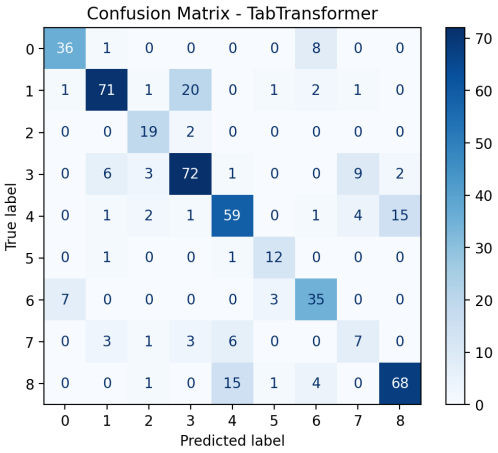
(a) ROC (TabSeq)

(b) Confusion matrix (TabSeq)

Figure A13: One-vs-all ROC and confusion matrix for TabSeq with standard cross-entropy.

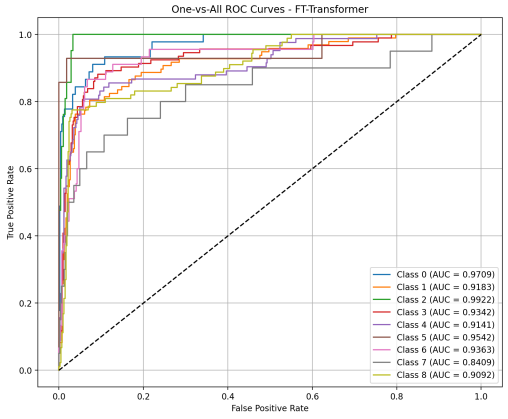


(a) ROC (TabTransformer)

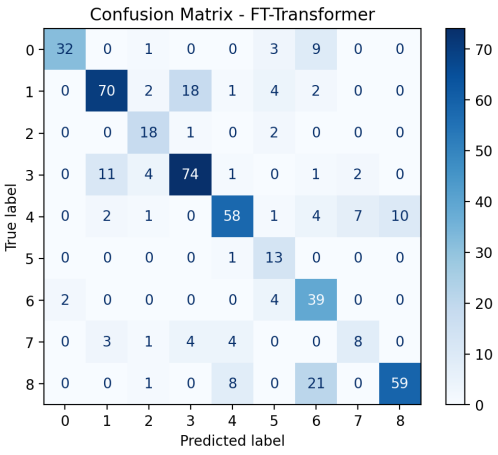


(b) Confusion matrix (TabTransformer)

Figure A14: One-vs-all ROC and confusion matrix for TabTransformer with standard cross-entropy.

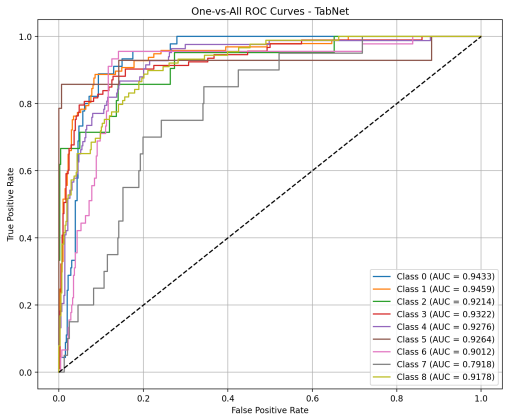


(a) ROC (FT-Transformer)

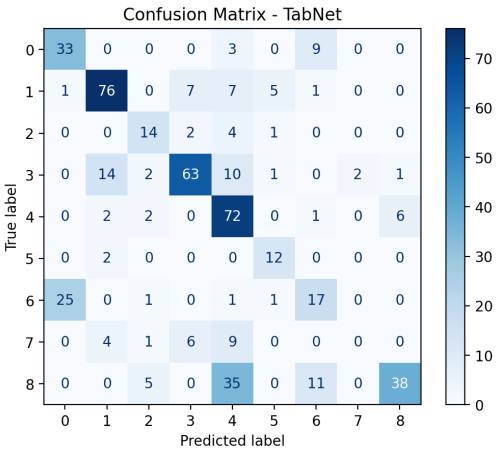


(b) Confusion matrix (FT-Transformer)

Figure A15: One-vs-all ROC and confusion matrix for FT-Transformer with standard cross-entropy.

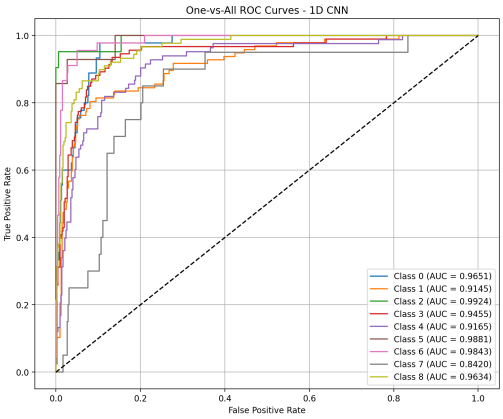


(a) ROC (TabNet)

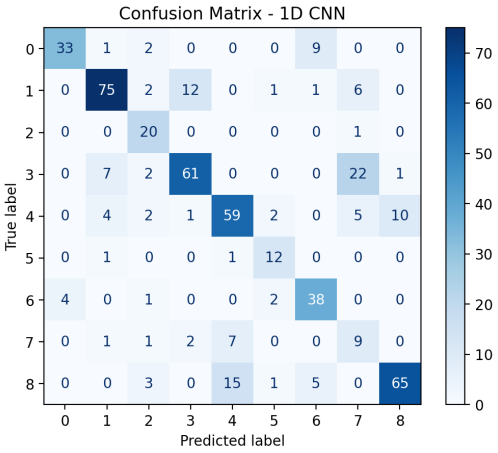


(b) Confusion matrix (TabNet)

Figure A16: One-vs-all ROC and confusion matrix for TabNet with standard cross-entropy.

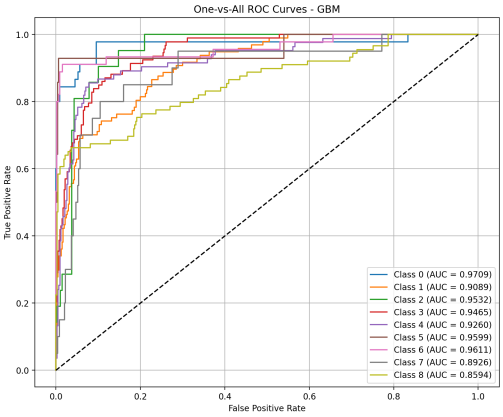


(a) ROC (1D CNN)

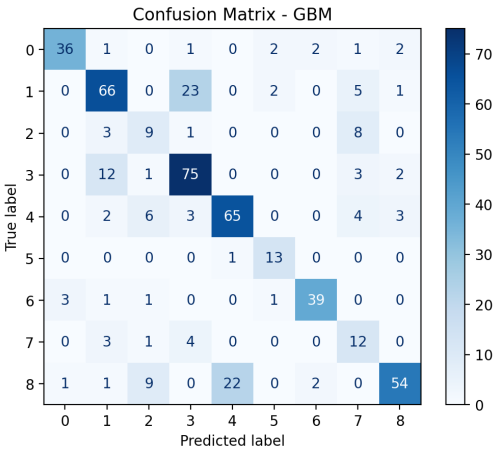


(b) Confusion matrix (1D CNN)

Figure A17: One-vs-all ROC and confusion matrix for 1D CNN with standard cross-entropy.

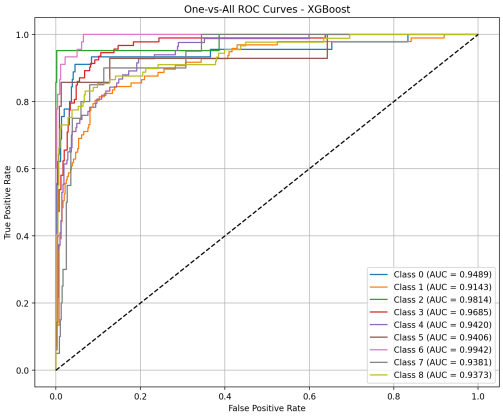


(a) ROC (GBM)

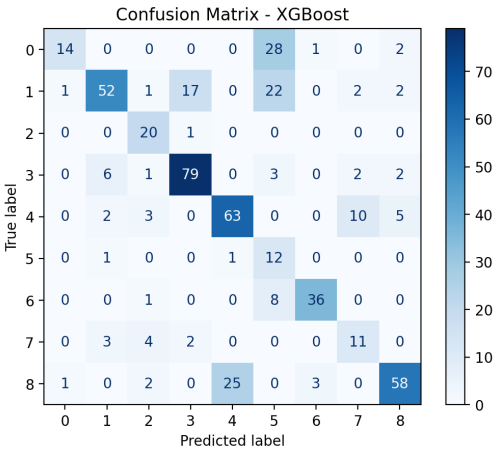


(b) Confusion matrix (GBM)

Figure A18: One-vs-all ROC and confusion matrix for Gradient Boosting (GBM).

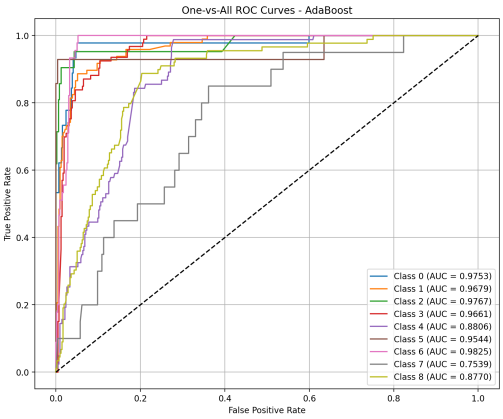


(a) ROC (XGBoost)

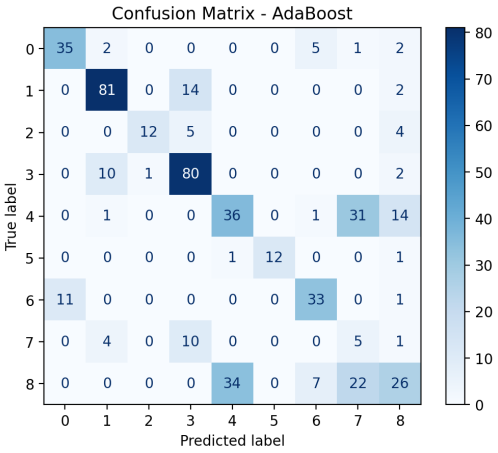


(b) Confusion matrix (XGBoost)

Figure A19: One-vs-all ROC and confusion matrix for XGBoost.

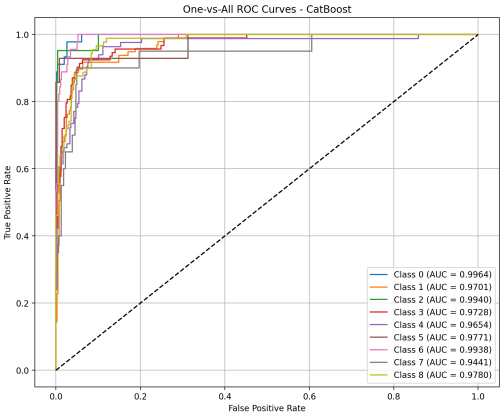


(a) ROC (AdaBoost)

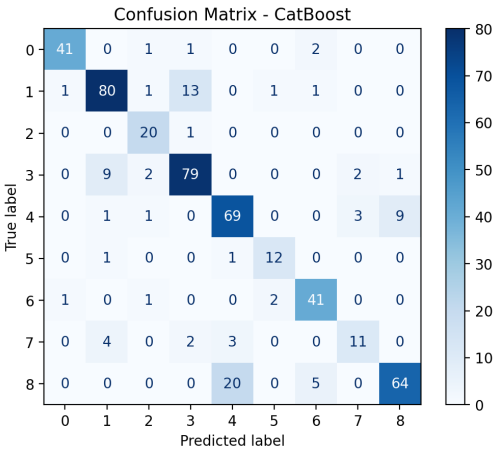


(b) Confusion matrix (AdaBoost)

Figure A20: One-vs-all ROC and confusion matrix for AdaBoost.

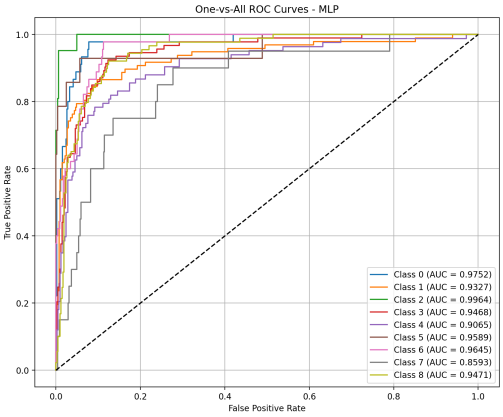


(a) ROC (CatBoost)

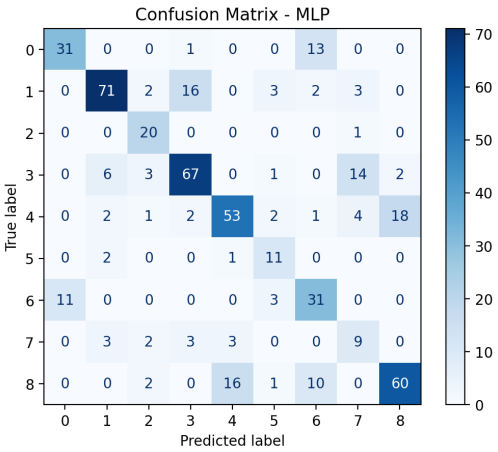


(b) Confusion matrix (CatBoost)

Figure A21: One-vs-all ROC and confusion matrix for CatBoost.

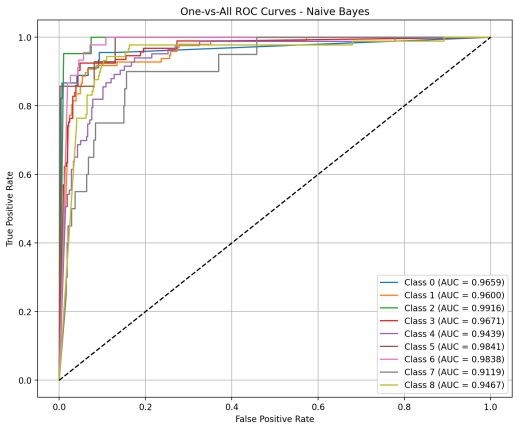


(a) ROC (MLP)

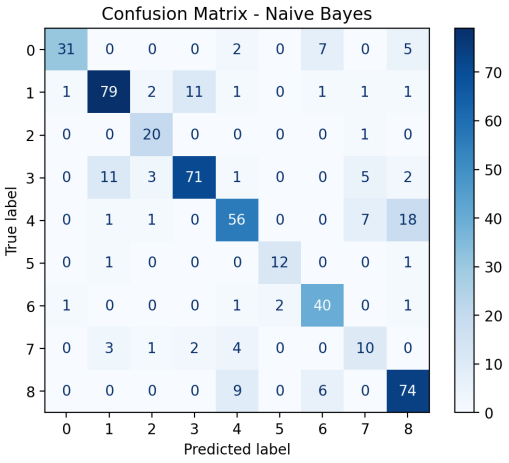


(b) Confusion matrix (MLP)

Figure A22: One-vs-all ROC and confusion matrix for a shallow MLP.

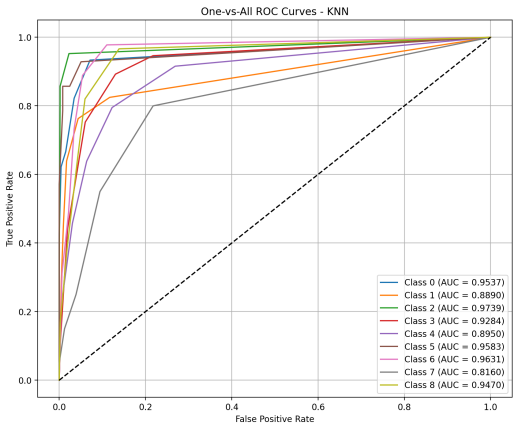


(a) ROC (Naive Bayes)

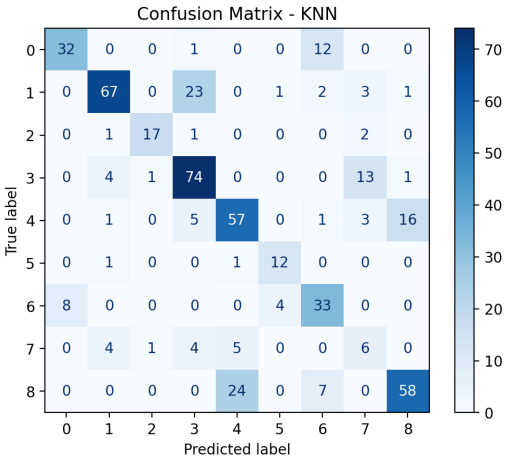


(b) Confusion matrix (Naive Bayes).

Figure A23: One-vs-all ROC and confusion matrix for Naive Bayes.

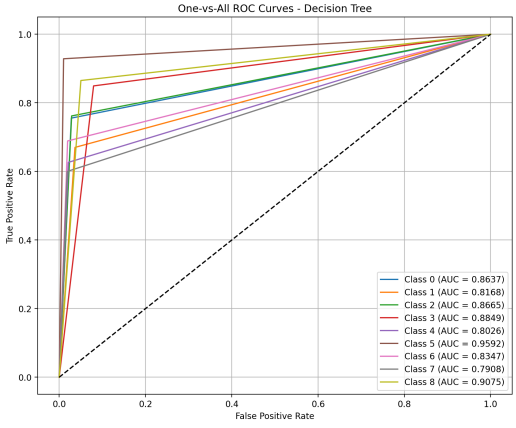


(a) ROC (KNN)

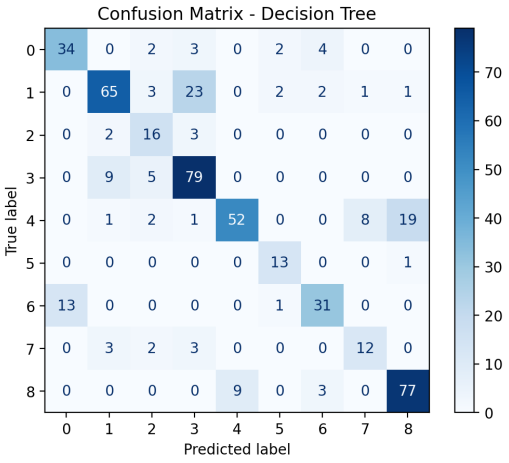


(b) Confusion matrix (KNN).

Figure A24: One-vs-all ROC and confusion matrix for KNN.

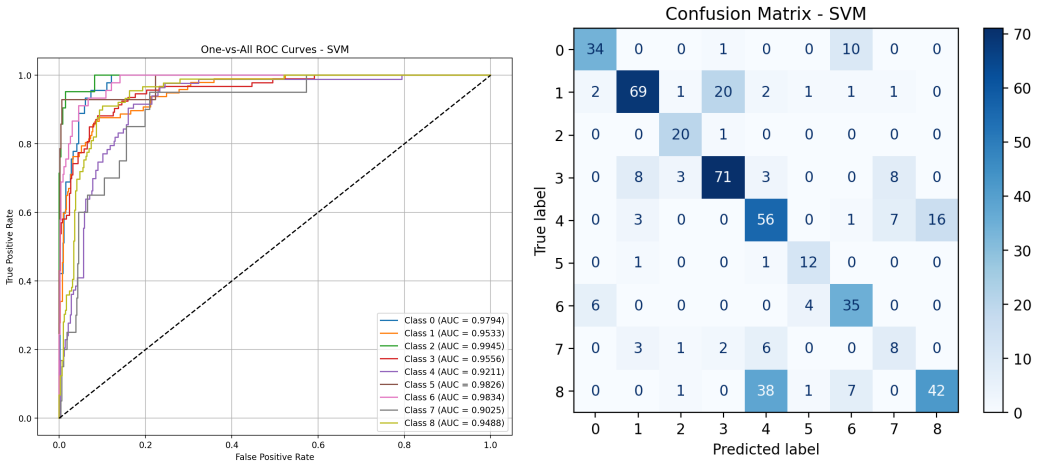


(a) ROC (Decision Tree).



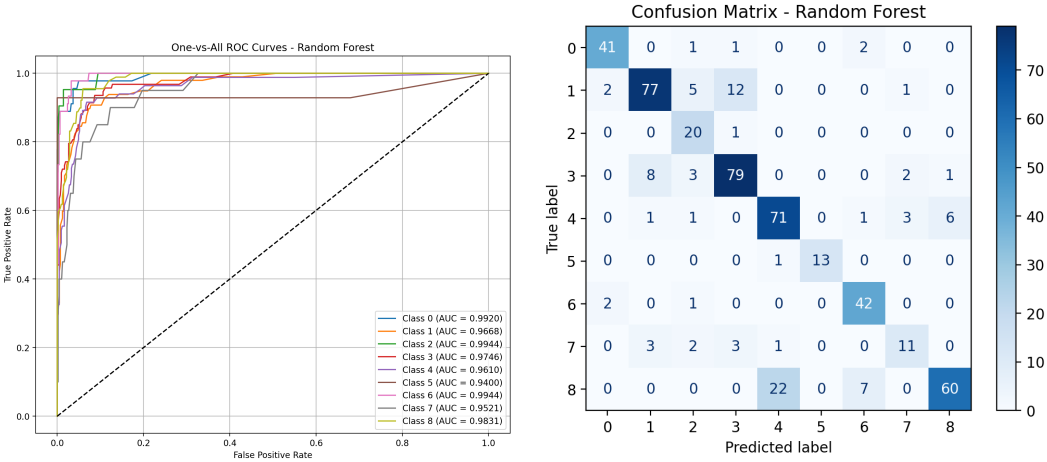
(b) Confusion matrix (Decision Tree).

Figure A25: One-vs-all ROC and confusion matrix for Decision Tree.



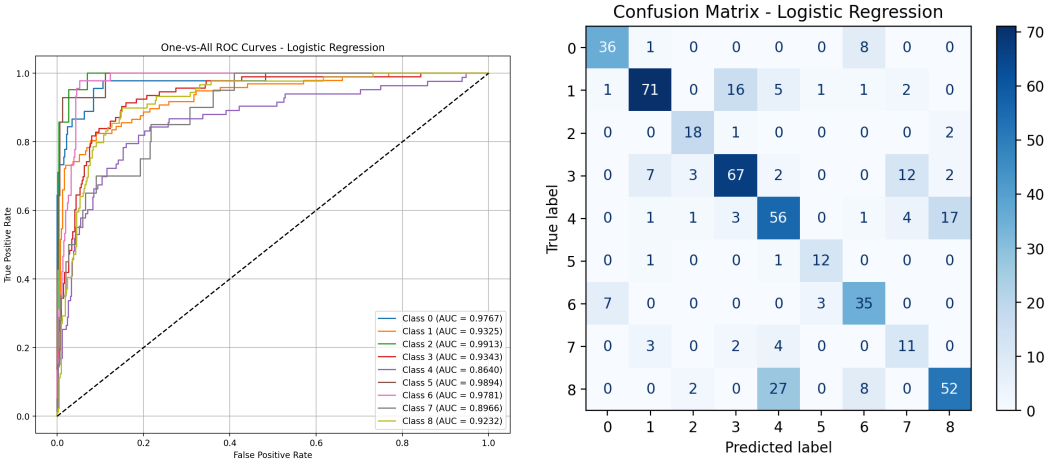
(a) ROC (SVM). (b) Confusion matrix (SVM).

Figure A26: One-vs-all ROC and confusion matrix for SVM.



(a) ROC (Random Forest). (b) Confusion matrix (Random Forest).

Figure A27: One-vs-all ROC and confusion matrix for Random Forest.



(a) ROC (Logistic Regression). (b) Confusion matrix (Logistic Regression).

Figure A28: One-vs-all ROC and confusion matrix for Logistic Regression.