TRAINING GRADIENT BOOSTED DECISION TREES ON TABULAR DATA CONTAINING LABEL NOISE FOR CLASSIFICATION TASKS

ANITA EISENBÜRGER, PROF. FRANK HOPFGARTNER, PROF. ANSELM HUDDE, DR. DANIEL OTTEN

Agenda

1.

Motivation

Problem Statement

- Getting labeled data is time-consuming and expensive [1]
- What if the few labels available are unreliable?

Label noise is the presence of incorrect labels in a dataset [1].

Consequences of Label Noise

- Decrease in model performance
- Increase in model complexity
- Increases the amount of data required for training
- Biases model comparison

[2]

2.

Related Work & Preliminaries

Approaches in Dealing with Label Noise

- Data cleansing: Modify the dataset D
  - Remove or relabel mislabeled instances
- Robust models: Use robust models f or loss functions L
- Tolerant algorithms: Adapt the objective
  - Regularize the model
  - Model the label noise

[2]

Current Landscape of Label Noise Research

- Deep neural networks (DNNs) [3]
- Text and image classification [3]
- Small loss trick [1]

Gradient-Boosted Decision Trees (GBDTs)

- Tabular data is a frequently used data format [8]
- State-of-the-art for tabular data [7]

Boosting

- Approximate y with the sum of multiple weak learners

$$f^{t}(x) = f^{t-1}(x) + \eta \cdot m_t(x)$$

- Each trying to correct the errors of its predecessor, e.g. the residual error [9]

$$m_t(x) = y - f^{t-1}(x)$$

Gradient Boosting & GBDTs

- Gradient Boosting: Fit the negative gradient of the predecessor
- Example: Mean squared error

$$L_{MSE}(x_i, y_i, f^t) = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - f^t(x_i)\right)^2$$

$$g_t(x_i, y_i) = \frac{\partial L_{MSE}}{\partial f^t(x_i)} = -\frac{2}{N}\left(y_i - f^t(x_i)\right)$$

- Shallow decision trees as weak learners

GBDTs and Label Noise

- Boosting algorithms are sensitive to label noise [11]
  - Overcorrect for mislabeled instances
- Calculate training dynamics statistics to identify mislabeled instances [3]

3.

Research Goals & Scope

Research Goals

- Explore the effects of label noise on GBDTs
- Adapt GBDTs to be more robust to label noise

Scope

- Data cleansing (removing and relabeling)
- Tabular data
- Classification tasks

4.

Methodology

Methodology

- Apply two state-of-the data cleansing methods from deep learning to GBDTs
- Combine all noise detection methods with removal and relabeling

Deep Learning Methods

- Likelihood Ratio Testing Correction (LRT) [12]:

$$LR(f, x, \tilde{y}) = \frac{f_{\tilde{y}}(x)}{f_{\hat{y}}(x)}, \qquad \tilde{y}_{new} = \begin{cases} \hat{y}, & \text{if } LR(f, x, \tilde{y}) < \varepsilon \\ \tilde{y}, & \text{otherwise} \end{cases}$$

Deep Learning Methods

- Area under the Margin Ranking (AUM) [13]:

$$M^T(x, \tilde{y}) = z_{\tilde{y}}^t(x) - \max_{i \neq \tilde{y}} z_i^t(x), \qquad AUM(x, \tilde{y}) = \frac{1}{T} \sum_{t=1}^{T} M^T(x, \tilde{y})$$

where z is the logit

Training Dynamics Statistics (ConfCorr)

- Confidence $\mu(x_i) = \frac{1}{T}\sum_{t=1}^{T} p_t(\tilde{y}_i|x_i)$
- Correctness $\gamma(x_i) = \frac{1}{T}\sum_{t=1}^{T}[\hat{y}_i = \tilde{y}_i]$
- $\mu(x_i) \cdot \gamma(x_i) < \varepsilon$ is predicted as noisy

[3]

Noise Correction Methods

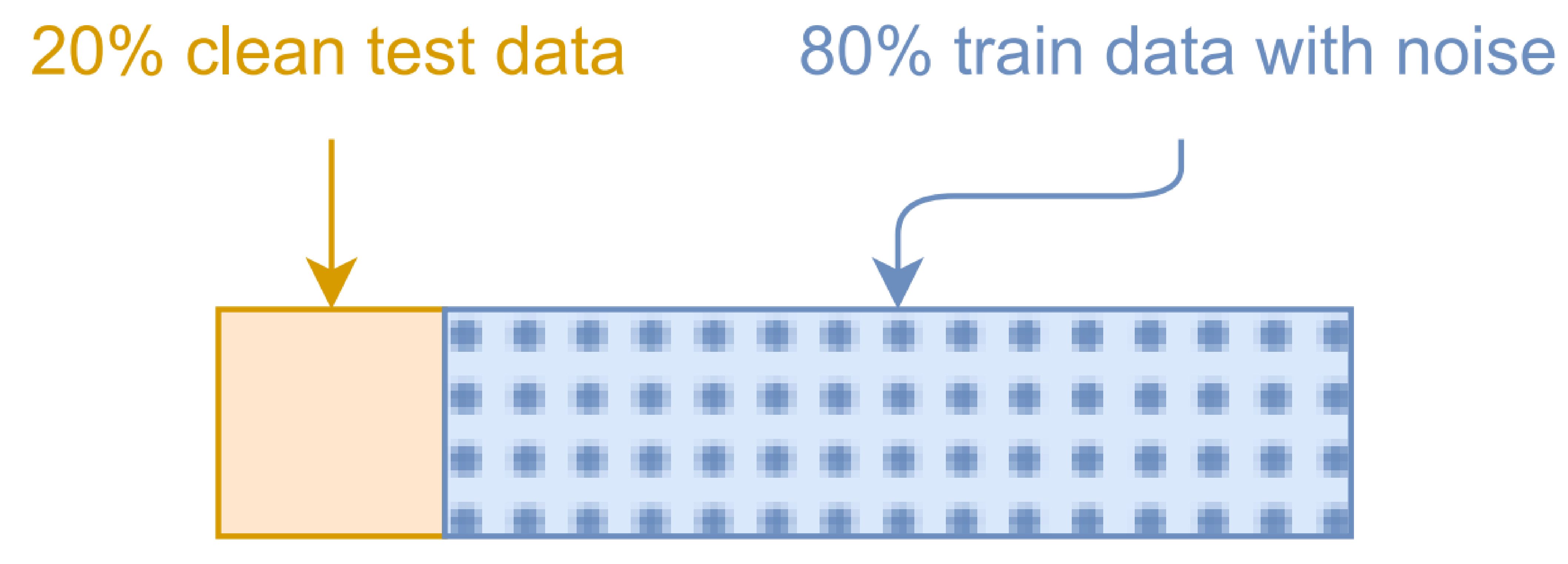
- Remove an instance marked as noisy
- Relabel an instance marked as noisy
    - Most frequent prediction across all epochs

Datasets

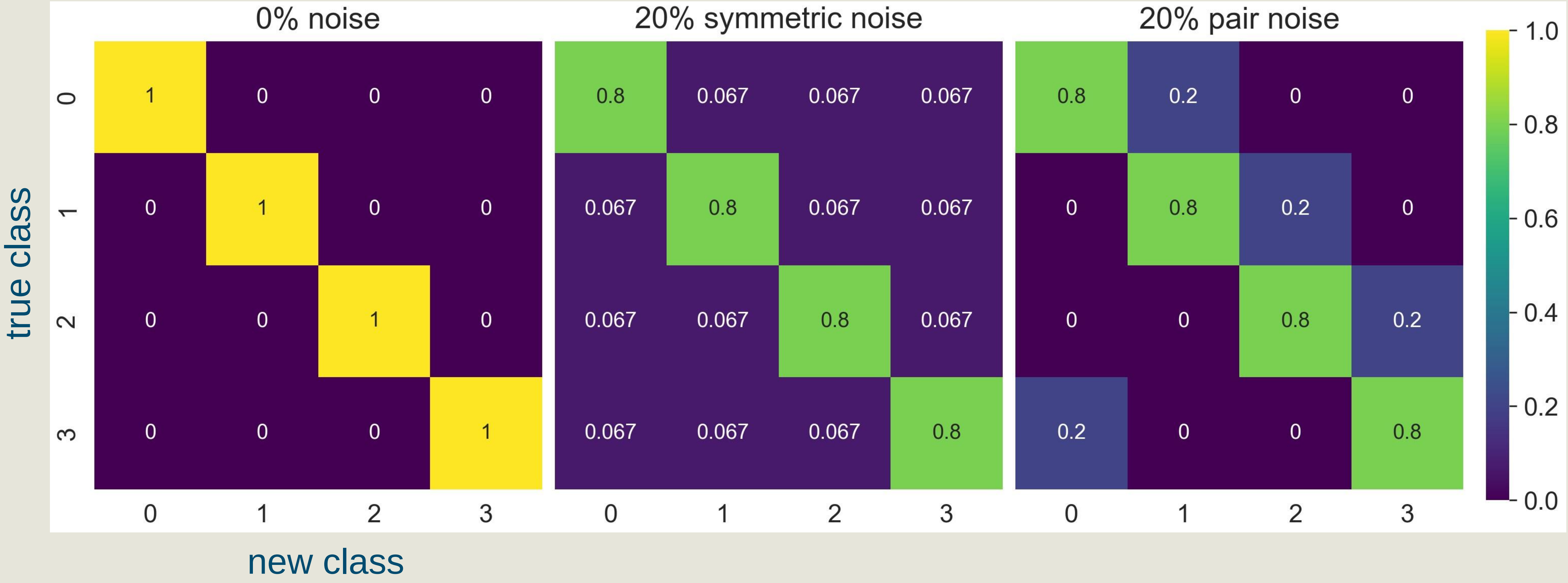
- Assumed to be clean due to the data collection process
- Polluted with label noise

Dataset	# Instances	# Features	# Classes	Data Types
Dry Bean [15]	13611	16	7	Numeric
Census [16]	48842	14	2	Mixed

Noise Injection



Types of Label Noise



Noise transition matrices

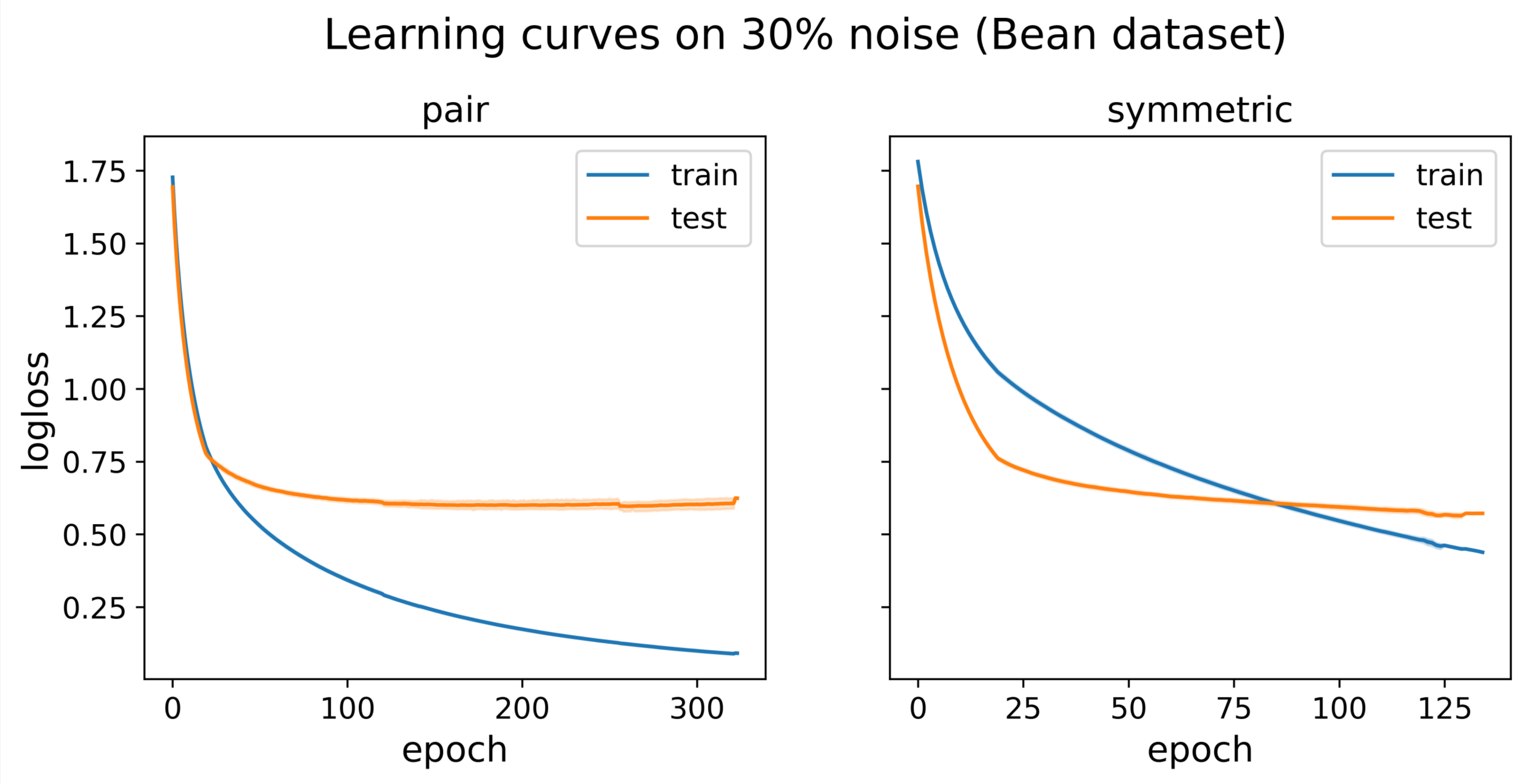
5.

Experiments

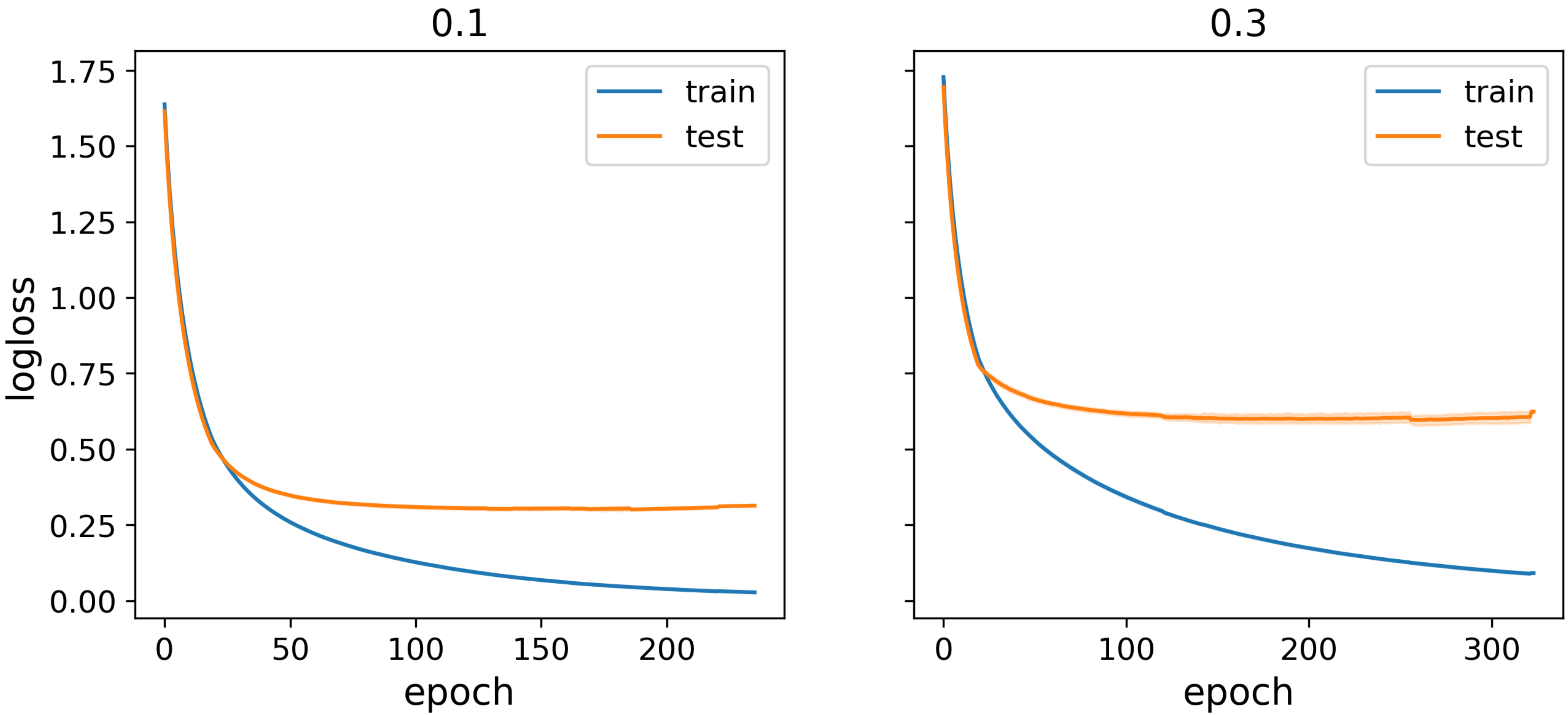
Research Questions (1)

Effects of label noise on GBDTs:

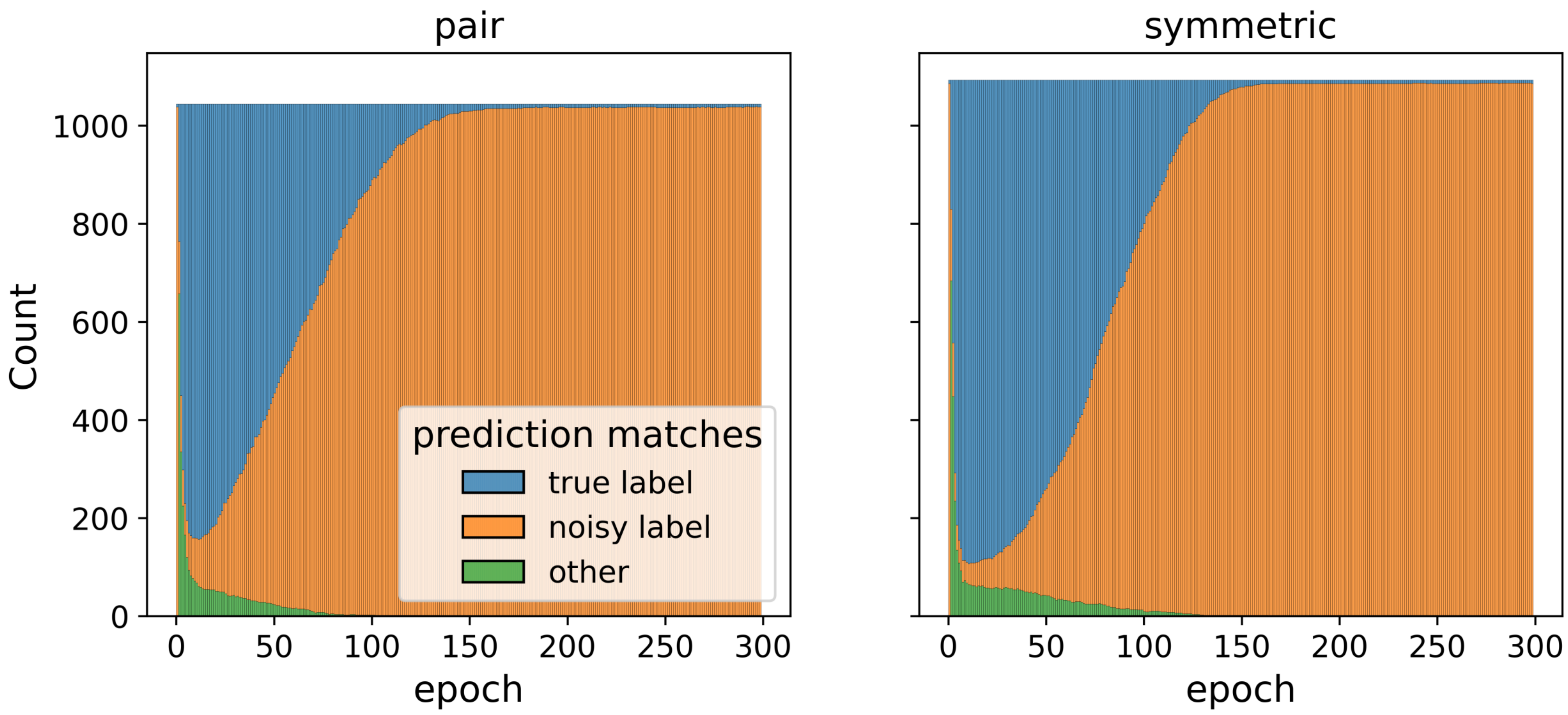
- How does label noise affect GBDTs throughout the training process?
- How do the two noise types affect GBDTs differently?


Learning curves on 30% noise (Bean dataset)
pair
symmetric
train
test
train
test
logloss
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0
100
200
300
epoch
0
25
50
75
100
125
epoch

Learning curves on 10% and 30% pair noise (Bean dataset)



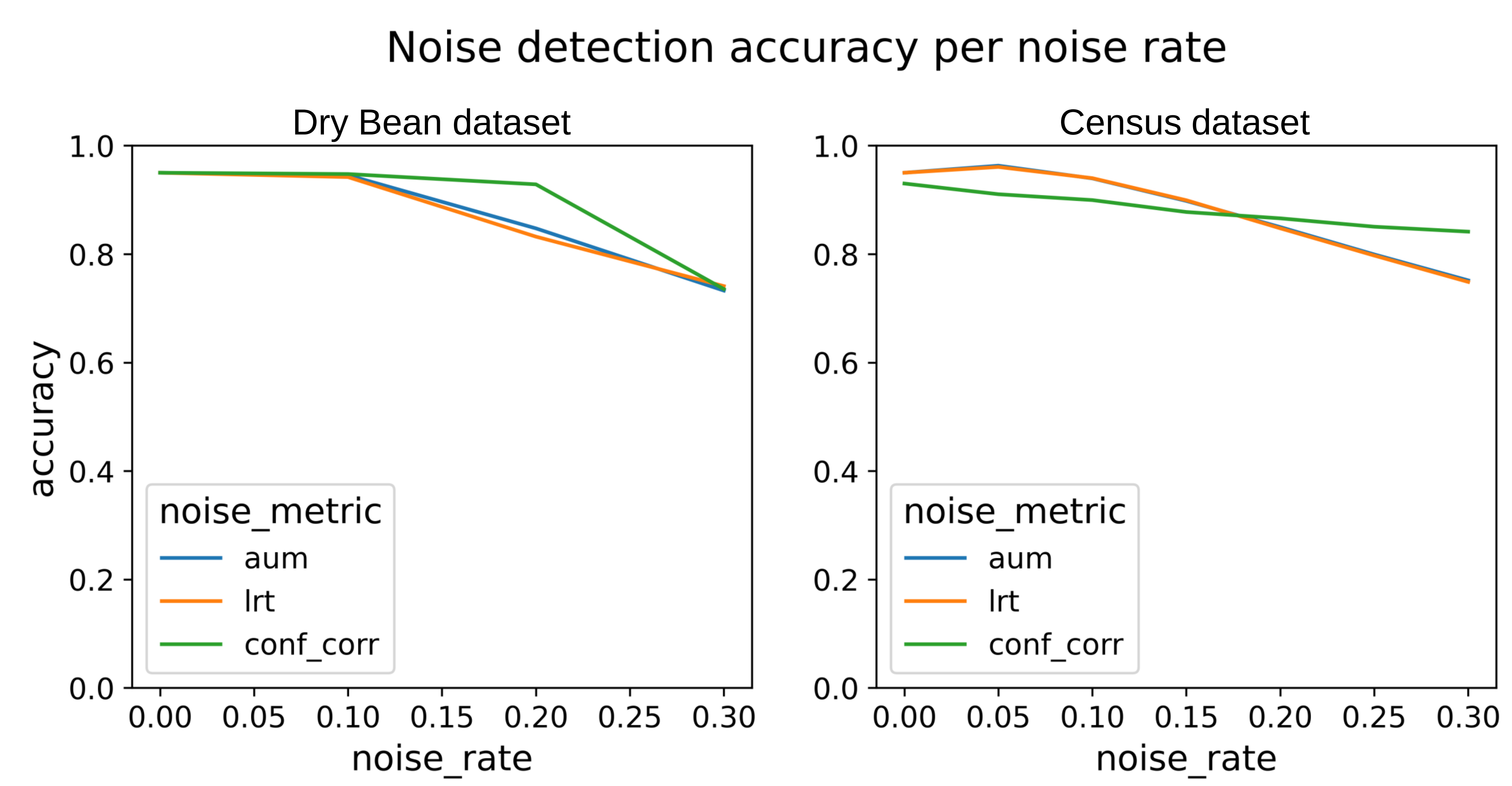
Types of model predictions during training on 10% noise (Bean dataset)



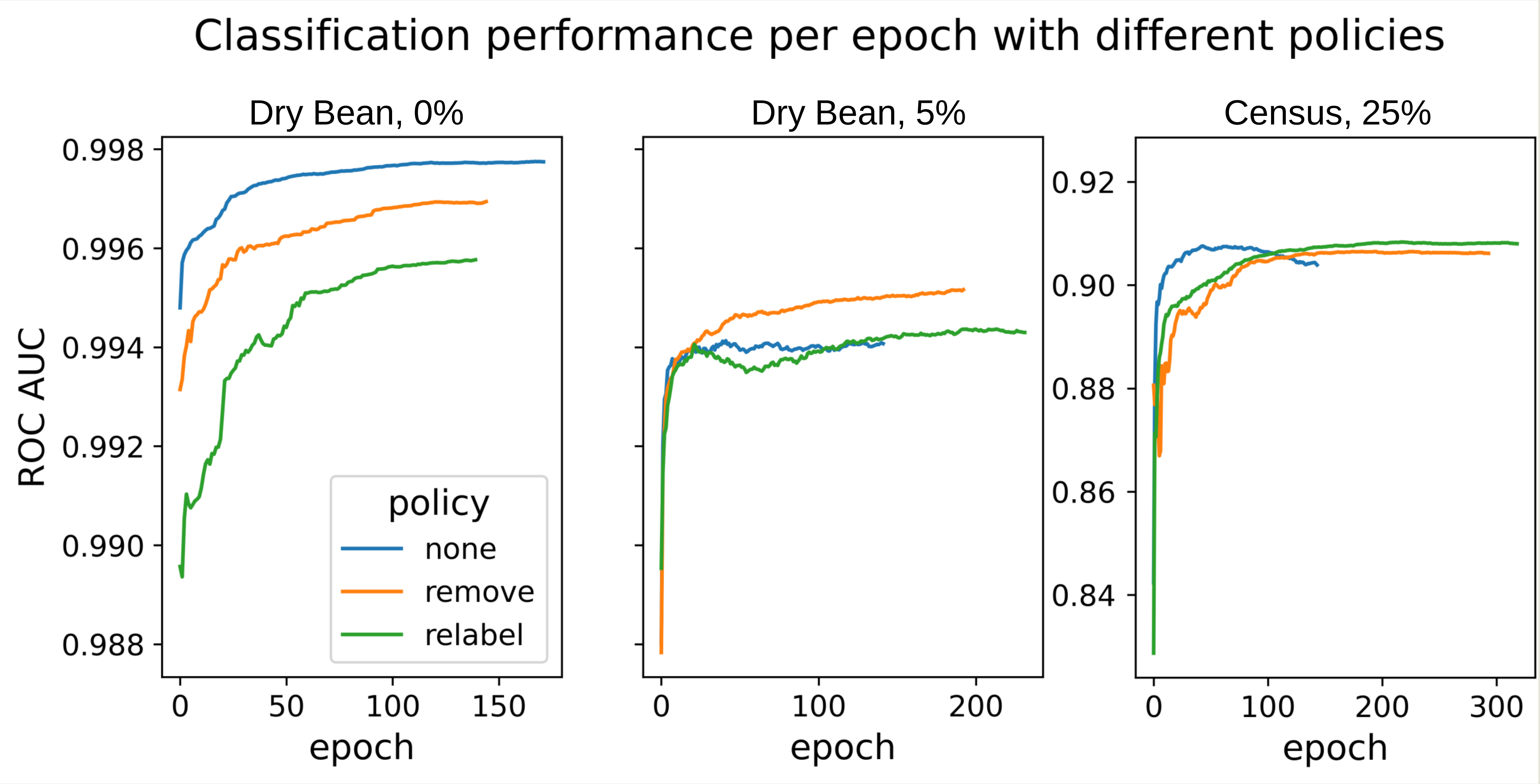
Research Questions (2)

Performance of noise detection and correction methods:

- How well do the detection methods perform?
- Which correction method performs better?


Noise detection accuracy per noise rate
Dry Bean dataset
Census dataset
1.0
0.8
0.6
0.4
0.2
0.0
accuracy
noise_metric
aum
lrt
conf_corr
0.00 0.05 0.10 0.15 0.20 0.25 0.30
noise_rate

pair noise



test set, pair noise, conf_corr metric

6.

Conclusion

Research Questions (1)

- GBDTs are robust to label noise
  - more to symmetric label noise
- They slowly adapt to the noisy labels during training
- Use early stopping to avoid overfitting

Research Questions (2)

- Noise detection and correction methods perform equally well
  - Optimal combination depends on the dataset and amount of noise
- Only correct for noise above a certain noise rate

Contributions

- Investigated effects of label noise on GBDTs
  - and offered practical advice
- Implemented methods to make GBDTs more robust to noise
  - Adapted label noise detection methods from DNNs to GBDTs
  - Expanded ConfCorr to work with relabeling

Future Work

- Estimate the amount of noise present in the data
- Explore different relabeling techniques
- Account for class imbalance

Thank You

You may now ask questions.

A

References

REFERENCES

[1] Song, Hwanjun, et al. "Learning from noisy labels with deep neural networks: A survey." IEEE Transactions on Neural Networks and Learning Systems (2022).

[2] Frénay, Benoît, and Michel Verleysen. "Classification in the presence of label noise: a survey." IEEE transactions on neural networks and learning systems 25.5 (2013): 845-869.

[3] Ponti, Moacir Antonelli, et al. "Improving Data Quality with Training Dynamics of Gradient Boosting Decision Trees." arXiv preprint arXiv:2210.11327 (2022).

[4] Han, Bo, et al. "Co-teaching: Robust training of deep neural networks with extremely noisy labels." Advances in neural information processing systems 31 (2018).

REFERENCES

[5] Li, Junnan, Richard Socher, and Steven CH Hoi. "Dividemix: Learning with noisy labels as semi-supervised learning." arXiv preprint arXiv:2002.07394 (2020).

[6] Malach, Eran, and Shai Shalev-Shwartz. "Decoupling" when to update" from" how to update"." Advances in neural information processing systems 30 (2017).

[7] Grinsztajn, Léo, Edouard Oyallon, and Gaël Varoquaux. "Why do tree-based models still outperform deep learning on typical tabular data?." Advances in Neural Information Processing Systems 35 (2022): 507-520.

[8] Shwartz-Ziv, Ravid, and Amitai Armon. "Tabular data: Deep learning is not all you need." Information Fusion 81 (2022): 84-90.

[9] Brophy, Jonathan, Zayd Hammoudeh, and Daniel Lowd. "Adapting and Evaluating Influence-Estimation Methods for Gradient-Boosted Decision Trees." J. Mach. Learn. Res. 24 (2023): 154-1.

[10] Xiang, Xingchun, Huaixuan Zhang, and Shu-Tao Xia. "Label Aggregation of Gradient Boosting Decision Trees." Proceedings of the 2020 2nd International Conference on Image Processing and Machine Vision. 2020.

[11] Karmaker, Amitava, and Stephen Kwek. "A boosting approach to remove class label noise." International Journal of Hybrid Intelligent Systems 3.3 (2006): 169-177.

REFERENCES

[12] Zheng, Songzhu, et al. "Error-bounded correction of noisy labels." International Conference on Machine Learning. PMLR, 2020.

[13] Pleiss, Geoff, et al. "Identifying mislabeled data using the area under the margin ranking." Advances in Neural Information Processing Systems 33 (2020): 17044-17056.

[13] Pleiss, Geoff, et al. "Identifying mislabeled data using the area under the margin ranking." Advances in Neural Information Processing Systems 33 (2020): 17044-17056.

[14] Covertype. archive.ics.uci.edu/dataset/31/covertype. Accessed 12 Jul. 2023.

Debeka

REFERENCES

[15] Dry Bean Dataset. archive.ics.uci.edu/dataset/602/dry+bean+dataset. Accessed 12 Jul. 2023.

[16] Adult. archive.ics.uci.edu/dataset/2/adult. Accessed 12 Jul. 2023.

[17] Breast Cancer Wisconsin (Diagnostic). archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic. Accessed 12 Jul. 2023.

B

Supplemental Material

Supervised Learning

- Given the dataset $D = \{(x_1, y_1), \dots, (x_N, y_N)\} \in (\mathrm{X}, \mathrm{Y})^N$
- Find a function $f_\theta : X \to Y$ with parameters θ
- Evaluate using a loss function $L: Y \times Y \to \mathbb{R}$
  - e.g. squared loss $L(f_\theta(x), y) = (y - f_\theta(x))^2$
- Optimize to find the optimal parameters θ

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} L(f_\theta(x_i), y_i)$$

Learning with Noisy Labels

- Datasets D contains noisy labels $\tilde{y}$, $D = \{(x_1, \tilde{y}_1), \ldots, (x_N, \tilde{y}_N)\}$
- $L(f(x), \tilde{y})$ is optimized instead of $L(f(x), y)$
- Resulting parameters $\tilde{\theta}^*$ differ from the desired parameters θ^*

[1][2]

Gradient Boosting & GBDTs

- Gradient Boosting: Fit the negative gradient of the predecessor

$$m_t(x) = -g_{t-1}(x, y)$$

$$f^t(x) = f^{t-1} + \eta \cdot \left(-g_{t-1}(x, y)\right)$$

$$g_t(x, y) = \frac{\partial L(f^t(x), y)}{\partial f^t(x)}$$

- Shallow decision trees as weak learners

[9]

Types of Label Noise

- Modeled with a noise transition matrix $S_{ij} = p(\tilde{y} = \mathrm{j} \mid \mathrm{y} = \mathrm{i}), S \in [0, 1]^{c \times c}$
- Symmetric noise: true label is flipped to other labels with equal probability
- Asymetric noise: true label is more likely to be flipped to a certain label than others
- Pair noise: true label is more likely to be flipped to one particular label
- Instance-dependent noise: true label is more likely to be flipped in certain regions of the feature space and to certain labels

[1][2]

Datasets

Method	Advantages	Disadvantages
Robust	No further considerations needed	Ineffective with more complex label noise or data
Tolerant	More grounded in theory	Assumptions about noise model limit applicability
Data Cleansing	Tackle the problem at the root	Overcleansing, error accumulation

[2]

Datasets

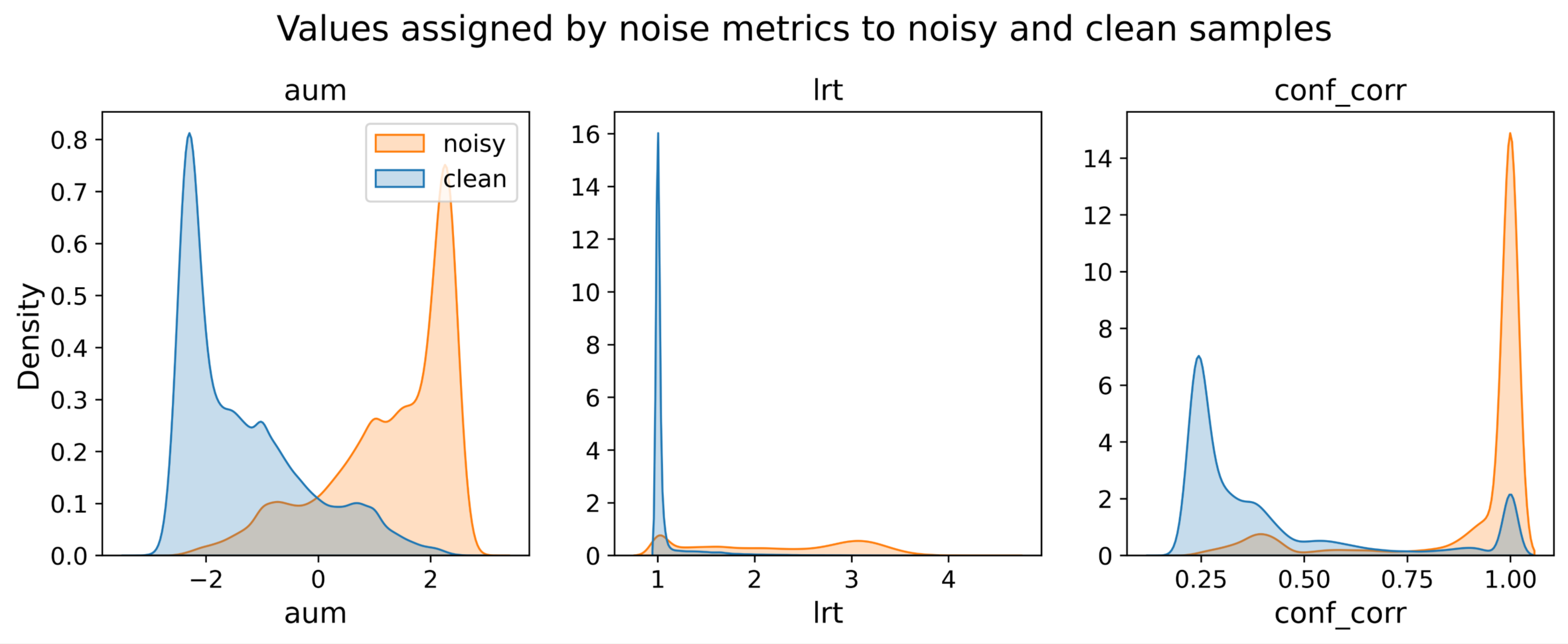
- Preprocessing
  - Impute missing data with median or mode
  - Standardize numeric attributes
  - One-hot encode categorical attributes
  - Discard features leaking information about target
- Added up to 60% label noise to the training set
- Test set remained clean

Experiment Conditions

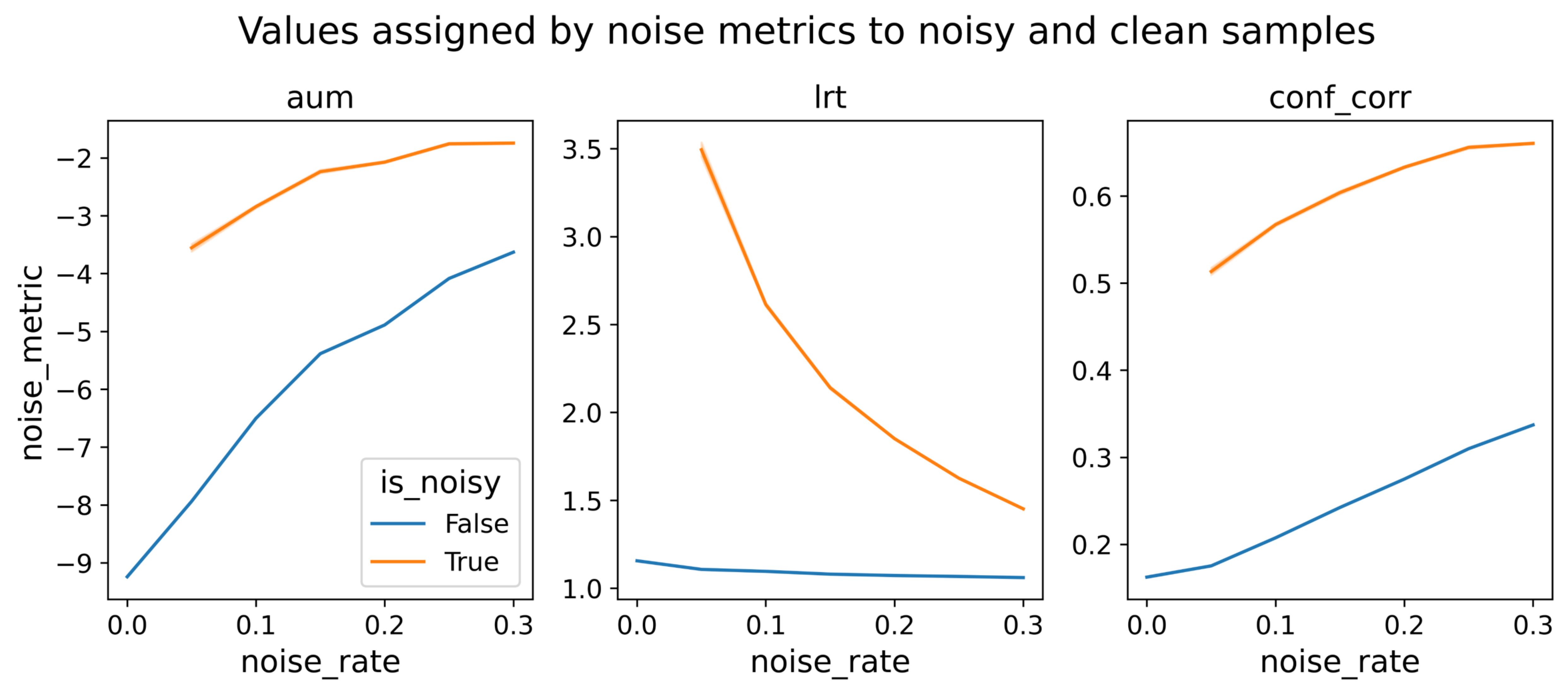
- Pair and symmetric noise from 0%-60%
  - 10%-40% for performance comparison
- XGBoost library, default model parameters
- Early stopping
  - Deactivated for some research questions
- No noise correction in the exploratory phase

Future Work

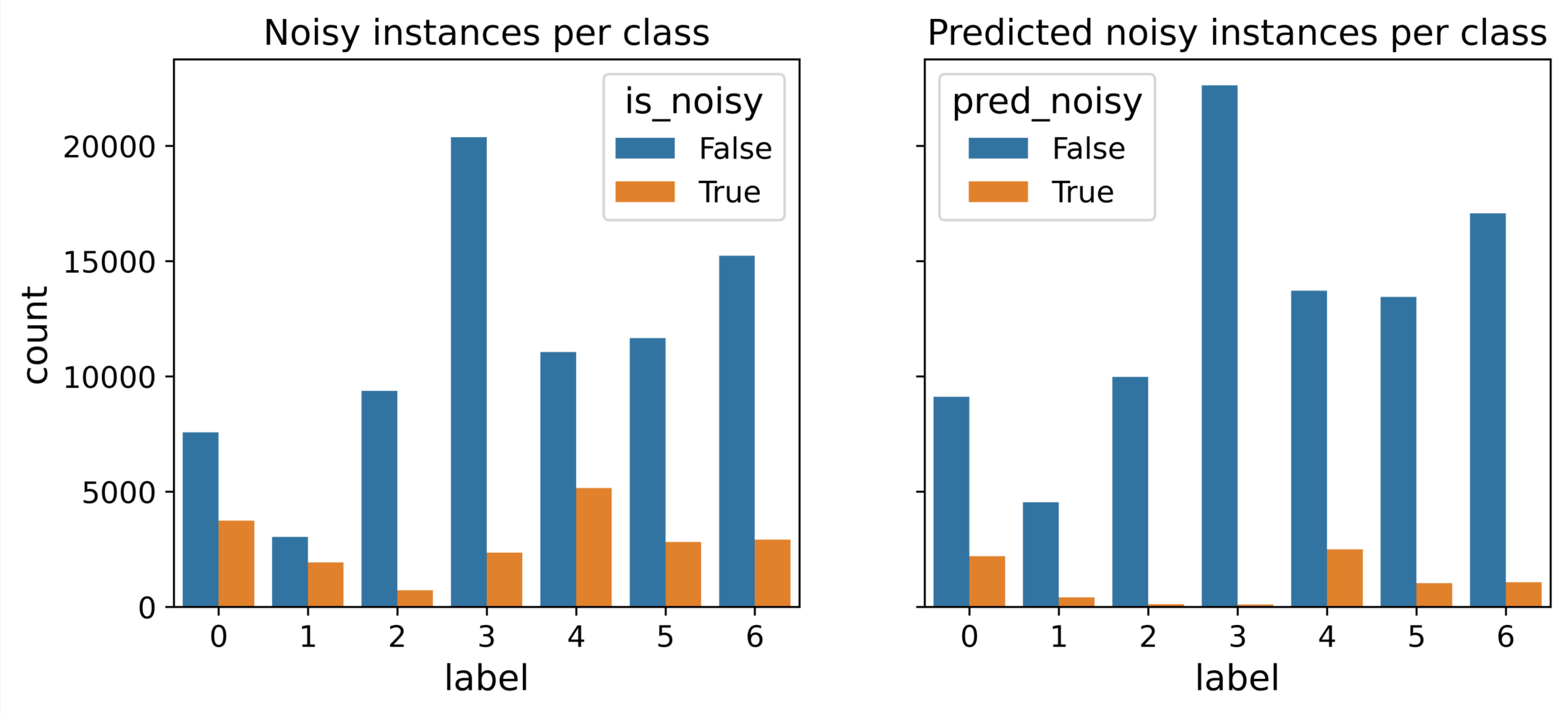
- Other relabeling methods
- Use noise detection methods more effectively, e.g. regularization
- Take class imbalance into consideration
- DNNs on tabular data with label noise

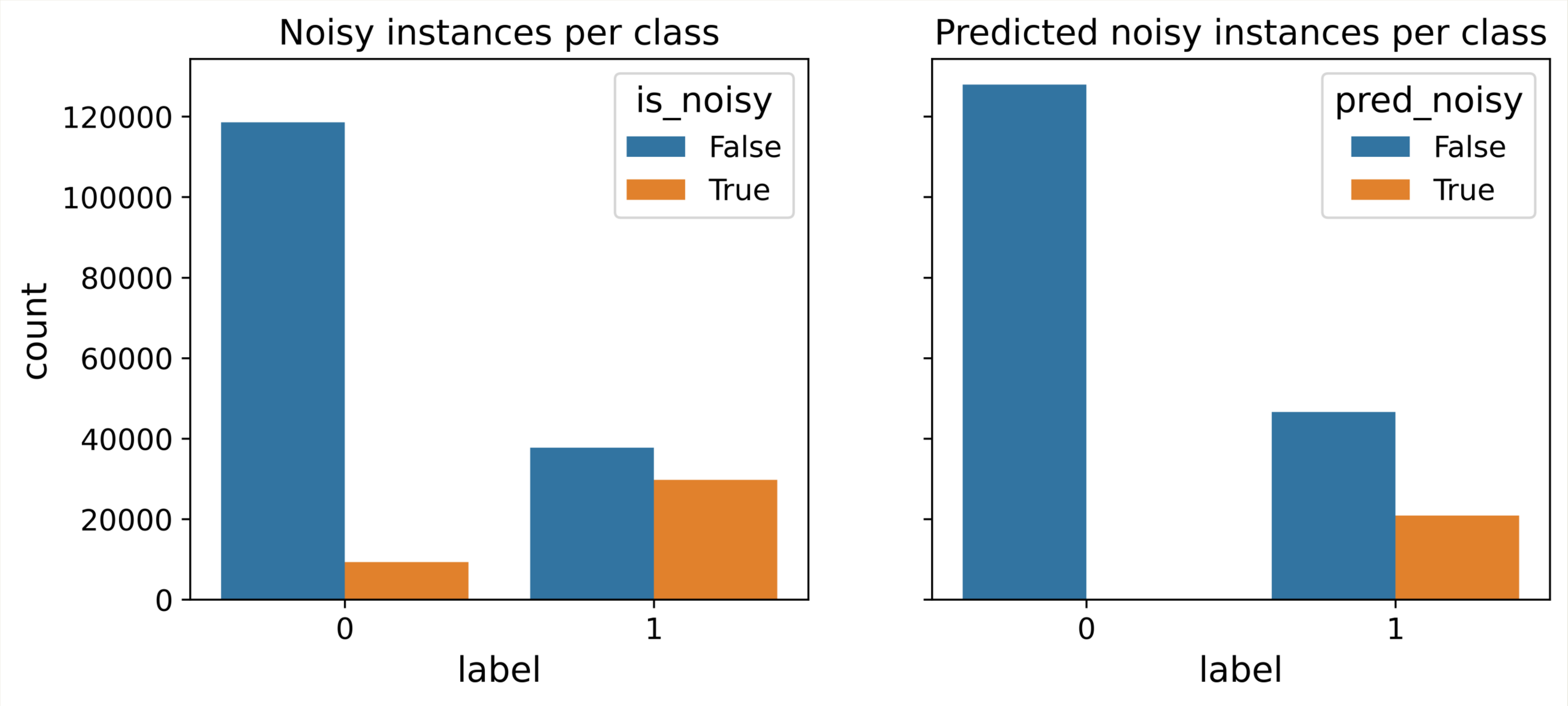


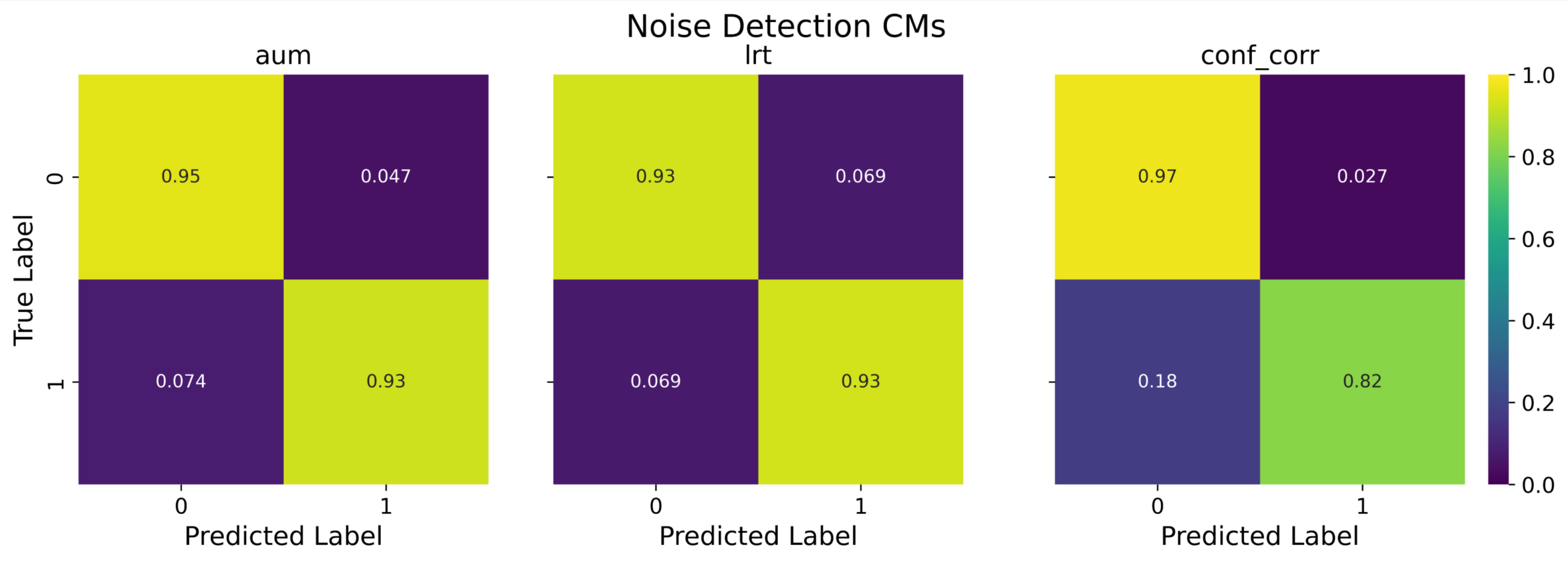
Census dataset, 20% pair noise

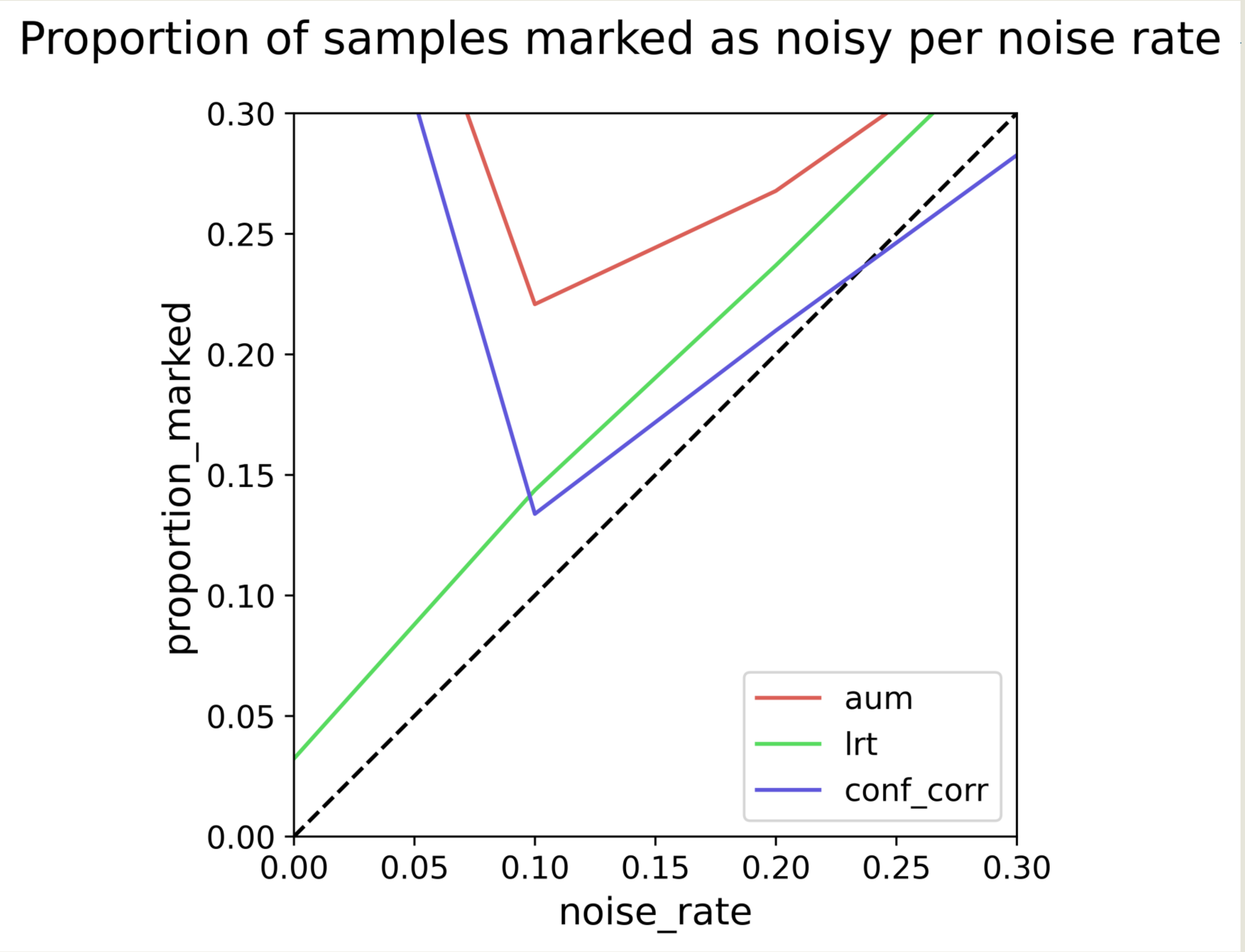


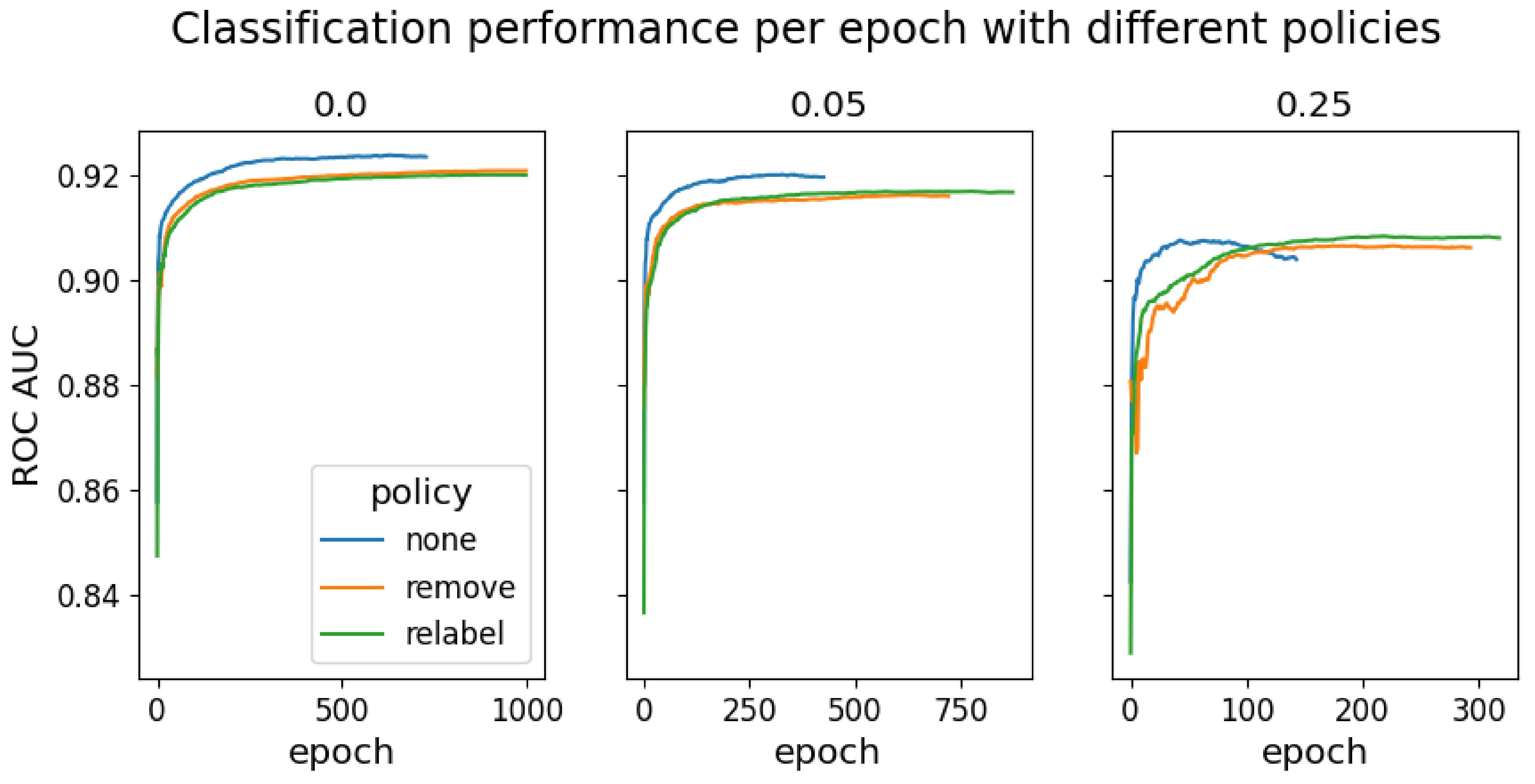
Census dataset, pair noise


Noisy instances per class
Predicted noisy instances per class
is_noisy
pred_noisy
False
True
count
label
0
5000
10000
15000
20000
0 1 2 3 4 5 6

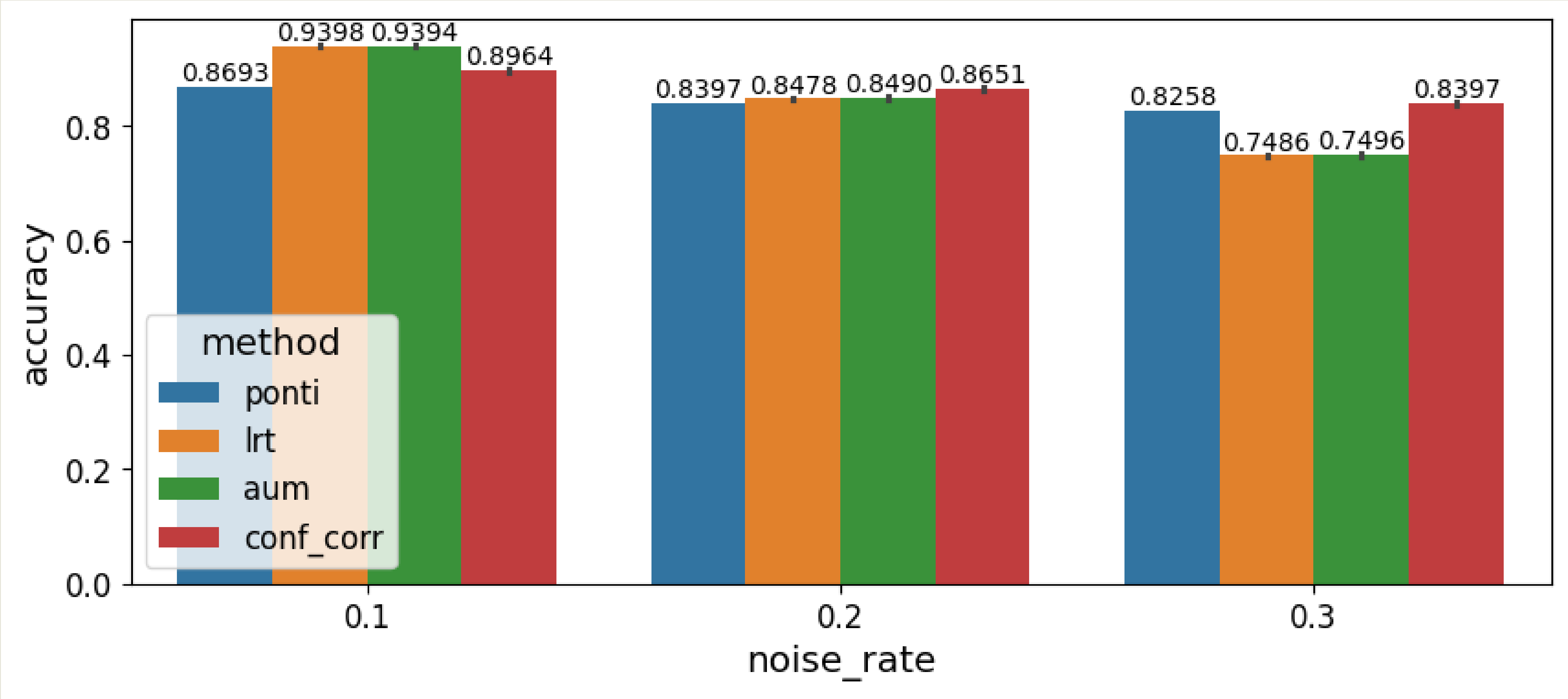

Noisy instances per class
Predicted noisy instances per class
count
120000
100000
80000
60000
40000
20000
0
is_noisy
False
True
pred_noisy
False
True
0
1
label
0
1
label


Noise Detection CMs
aum
lrt
conf_corr
True Label
0
1
0.95
0.047
0.074
0.93
0.93
0.069
0.069
0.93
0.97
0.027
0.18
0.82
0
1
Predicted Label
1.0
0.8
0.6
0.4
0.2
0.0

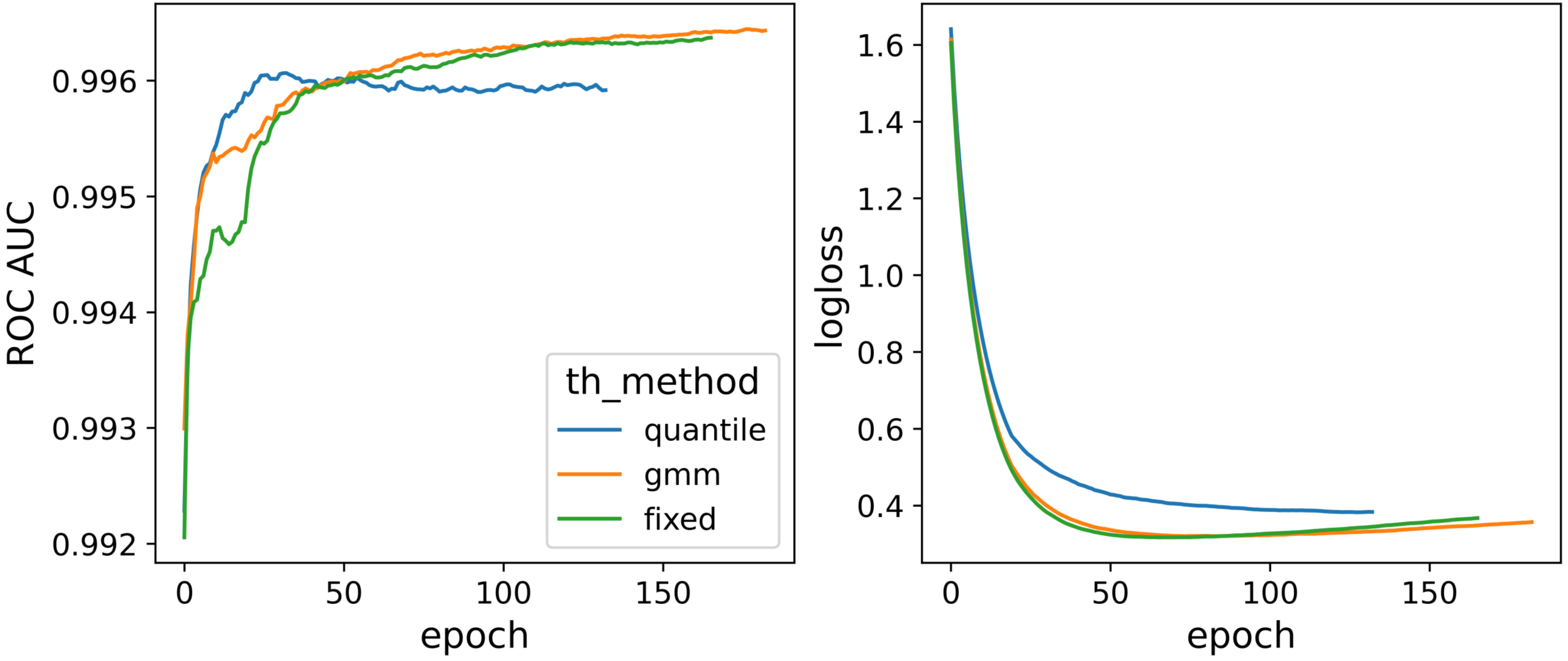

Proportion of samples marked as noisy per noise rate
0.30
0.25
0.20
0.15
0.10
0.05
0.00
proportion_marked
0.00
0.05
0.10
0.15
0.20
0.25
0.30
noise_rate
aum
lrt
conf_corr

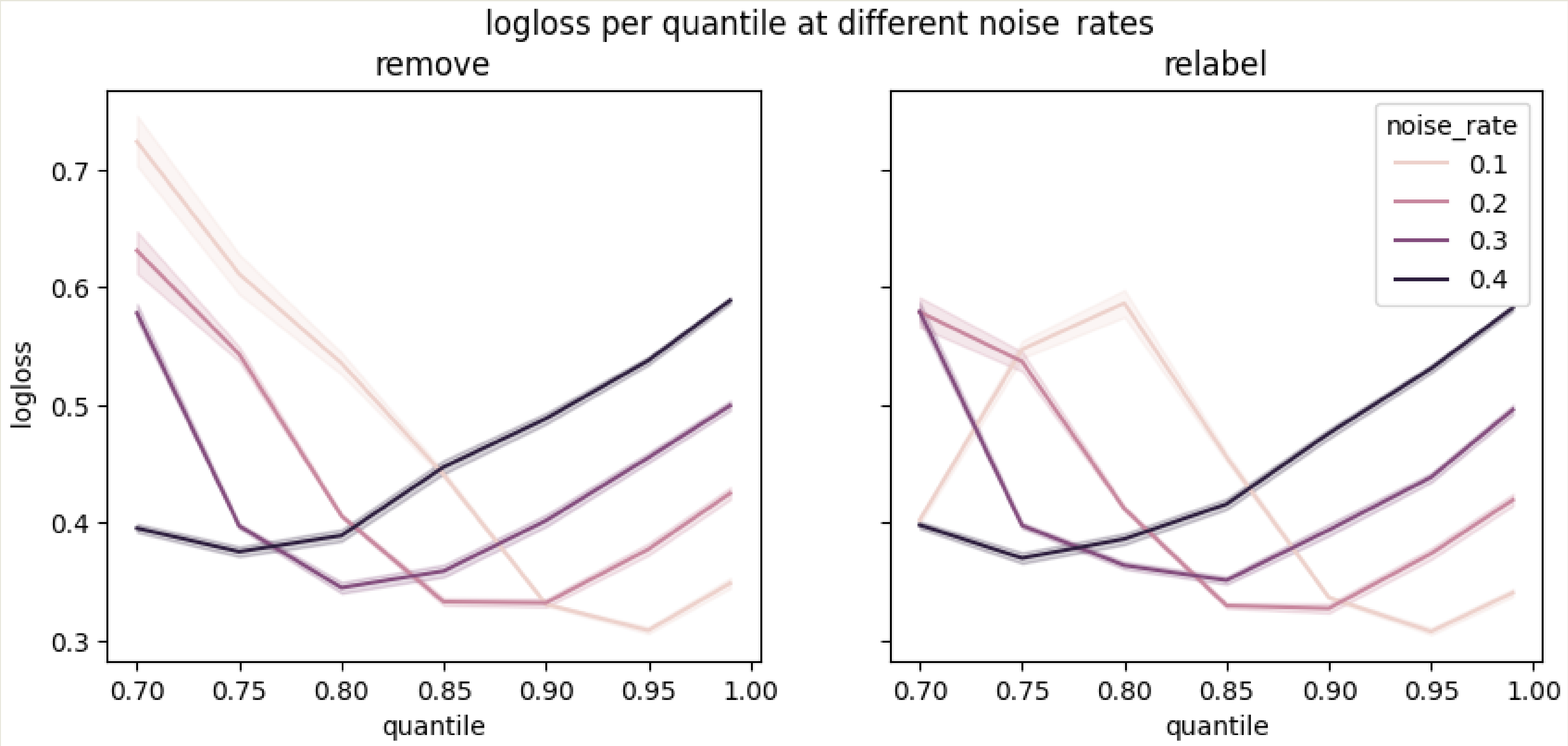


Classification performance per epoch with different policies
0.0
0.05
0.25
ROC AUC
0.92
0.90
0.88
0.86
0.84
policy
none
remove
relabel
0
500
1000
250
750
100
200
300
epoch

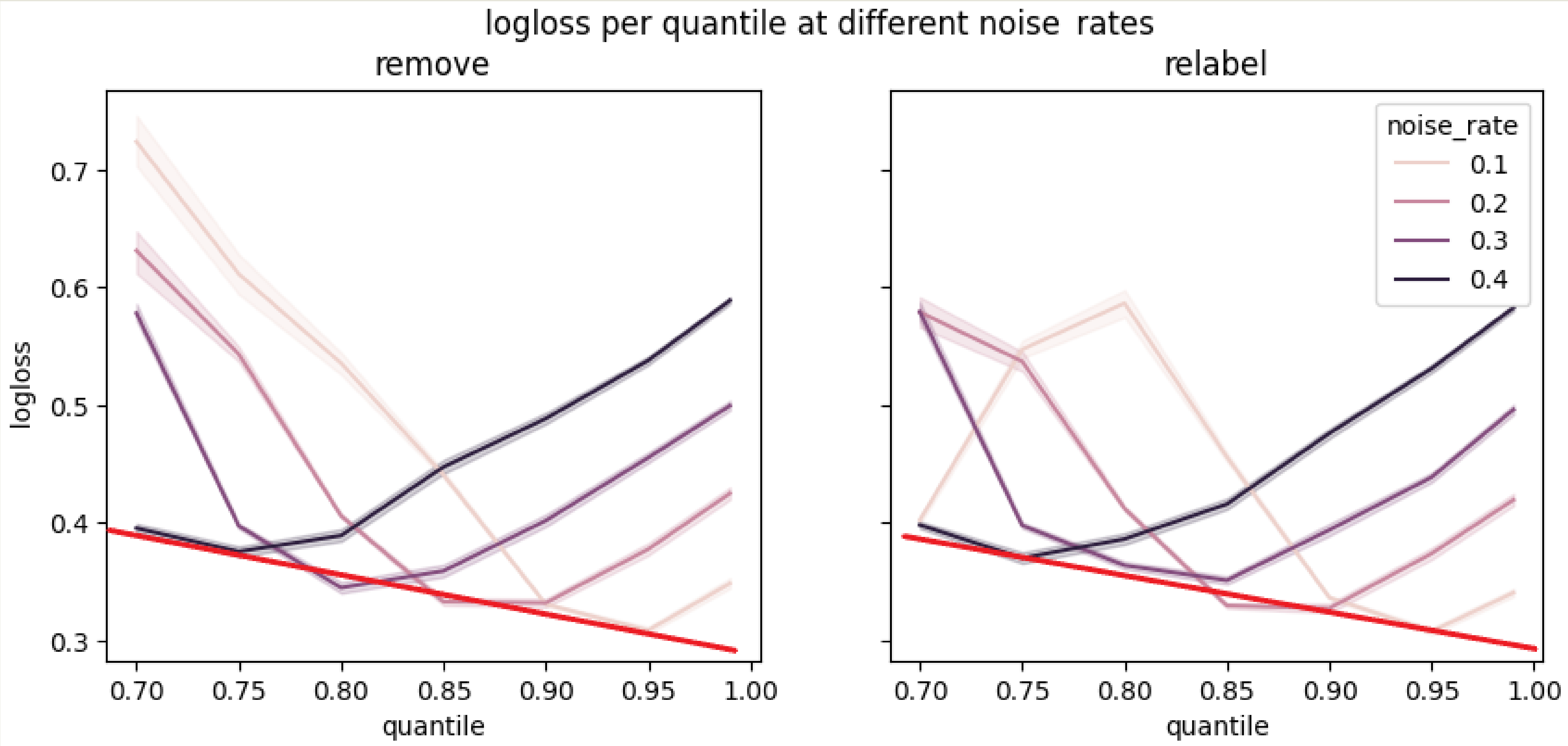

0.8693
0.9398
0.9394
0.8964
0.8397
0.8478
0.8490
0.8651
0.8258
0.7486
0.7496
0.8397
accuracy
0.0
0.2
0.4
0.6
0.8
method
ponti
lrt
aum
conf_corr
0.1
0.2
0.3
noise_rate

Classification performance per epoch with different threshold methods

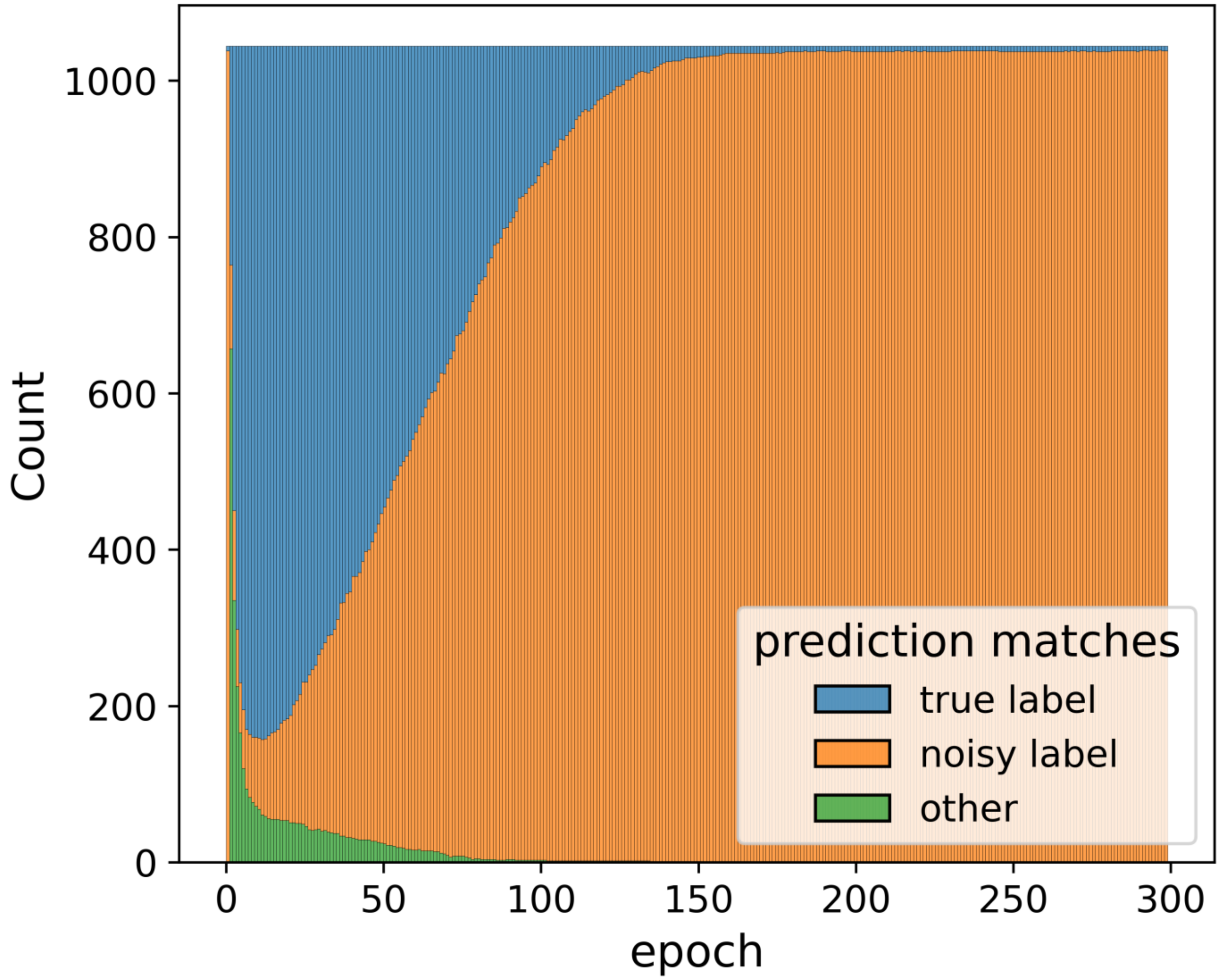




logloss per quantile at different noise rates
remove
relabel
logloss
0.7
0.6
0.5
0.4
0.3
0.70
0.75
0.80
0.85
0.90
0.95
1.00
quantile
noise_rate
0.1
0.2
0.3
0.4


logloss per quantile at different noise rates
remove
relabel
logloss
0.7
0.6
0.5
0.4
0.3
0.70
0.75
0.80
0.85
0.90
0.95
1.00
quantile
noise_rate
0.1
0.2
0.3
0.4

Types of model predictions during training on 10% pair noise (Bean dataset)



Thresholding Methods

- A fixed threshold
- The instances with the top x% noise values
- Fit a 2-component Gaussian Mixture Model (GMM)

Noise Correction Methods

- Remove an instance marked as noisy
  - No more than 80% of the instances in the training set can be removed
  - Retain excess instances if too many were marked
- Relabel an instance marked as noisy
  - Most frequent prediction across all epochs

Classification performance per epoch at different noise metrics (Bean dataset)