



Detecting Label Noise in Kepler Confirmed Planet Catalog using Machine Learning

Miguel Martinho

NASA Ames, Universities Space Research Association

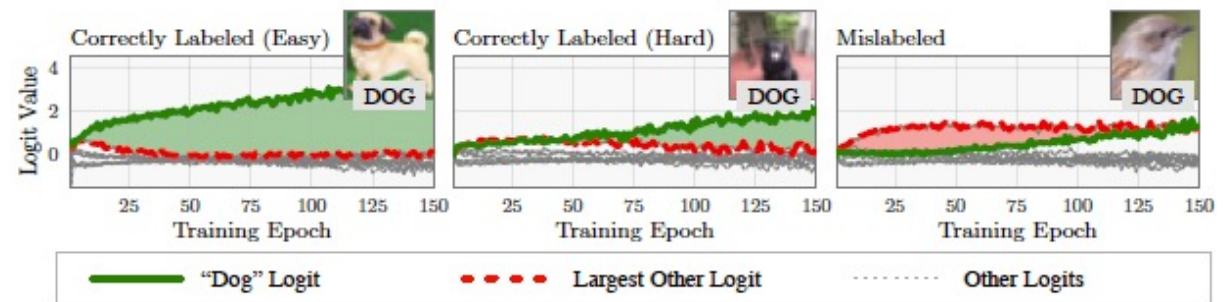
miguel.martinho@nasa.gov

Introduction

- Goal: detect confirmed planets that are false positives.
- Motivation: curate catalogs of true planet detections for a better understanding of the planet population.
- Method: track the training dynamics of a neural network trained on the classification of Kepler transit signals into Planet Candidates (PC) to distinguish mislabeled examples from correctly labeled ones¹.

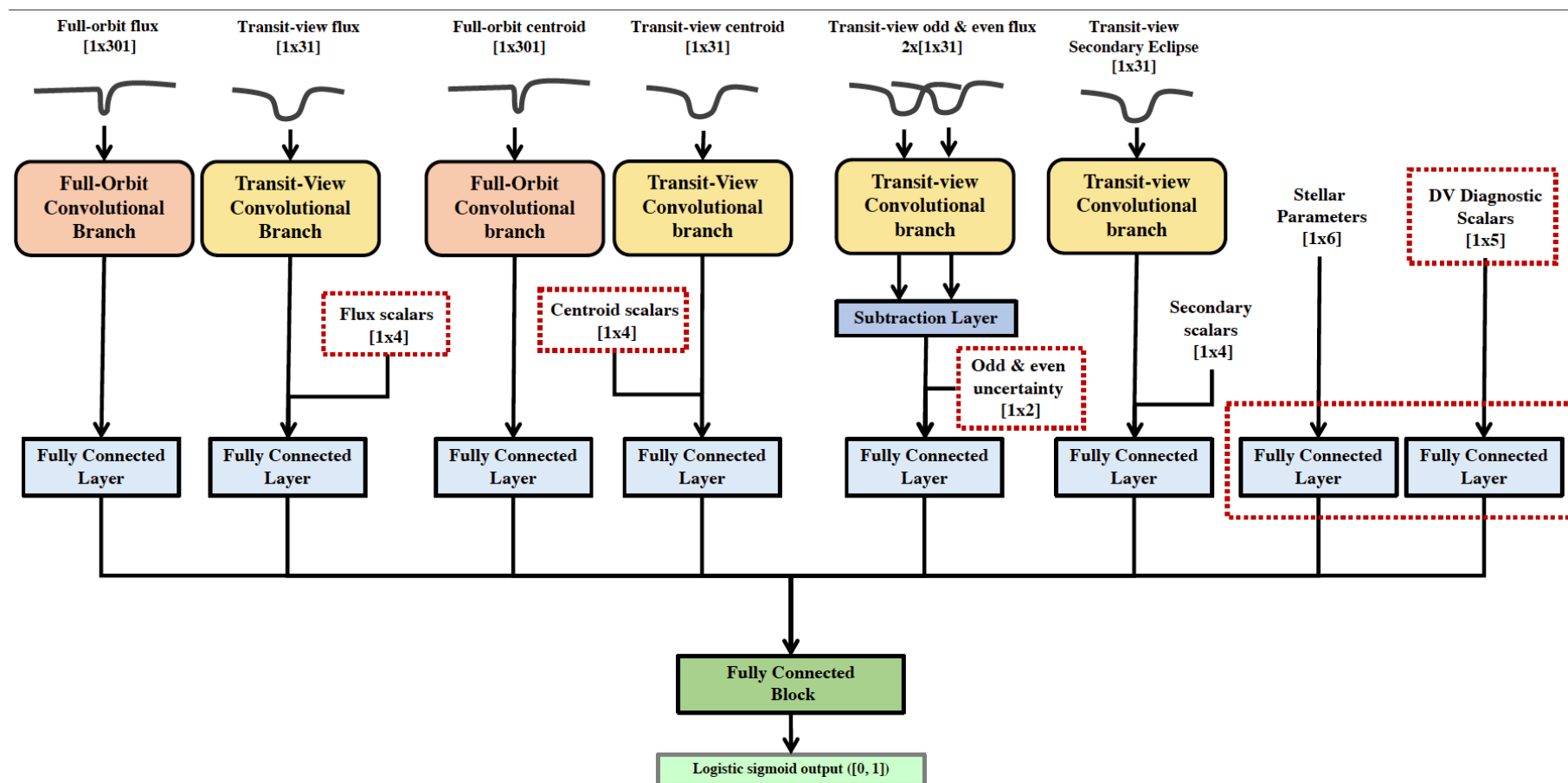
$$\text{Margin: } M^{(t)}(\mathbf{x}, y) = \overbrace{z_y^{(t)}(\mathbf{x})}^{\text{assigned logit}} - \overbrace{\max_{i \neq y} z_i^{(t)}(\mathbf{x})}^{\text{largest other logit}}$$

$$\text{Area Under Margin: } AUM(\mathbf{x}, y) = \frac{1}{T} \sum_{t=1}^T M^{(t)}(\mathbf{x}, y)$$



¹Pleiss, Geoff, et al. "Identifying mislabeled data using the area under the margin ranking." *Advances in Neural Information Processing Systems* 33 (2020): 17044-17056.

Model²



[2] Valizadegan, Hamed, et al. "ExoMiner: A Highly Accurate and Explainable Deep Learning Classifier That Validates 301 New Exoplanets." *The Astrophysical Journal* 926.2 (2022): 120.

Dataset

- Source catalog: Q1-Q17 DR25 TCEs.
- Kepler Confirmed Planets (PCs) + Certified False Positives (FPs).
- Updated disposition for ~350 KOIs recently confirmed as planets [2].

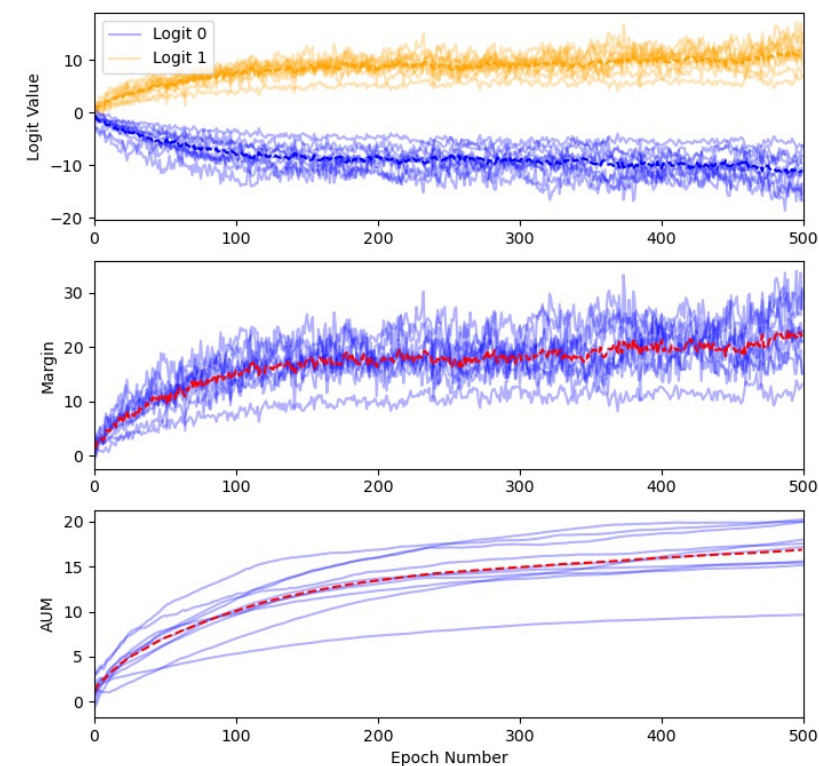
Classes	Counts
FP	3437 (56.5%)
PC	2643 (43.5%)
Total	6080

[2] Valizadegan, Hamed, et al. "ExoMiner: A Highly Accurate and Explainable Deep Learning Classifier That Validates 301 New Exoplanets." *The Astrophysical Journal* 926.2 (2022): 120.

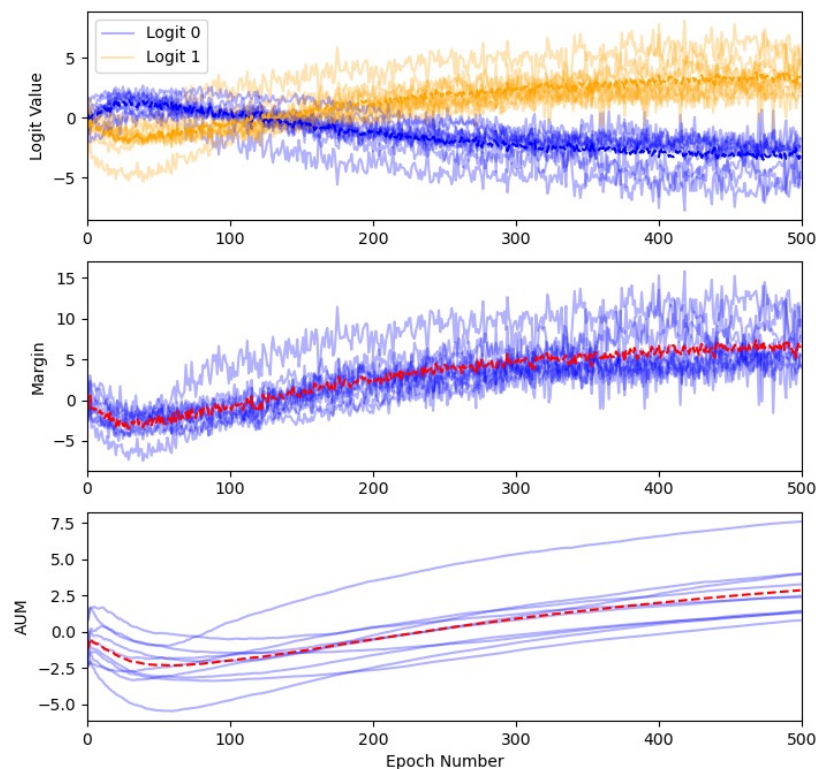
Method

- Conduct 10 runs.
 - For each run
 - Train one model for 500 epochs.
 - Track logits for all examples in the dataset at the end of each epoch.
 - Compute margins and AUM values for each example across epochs based on the logged logit values.
- Aggregate AUM values across runs for each example and epoch.
- Rank examples based on AUM values from lowest to highest at the end of the training.

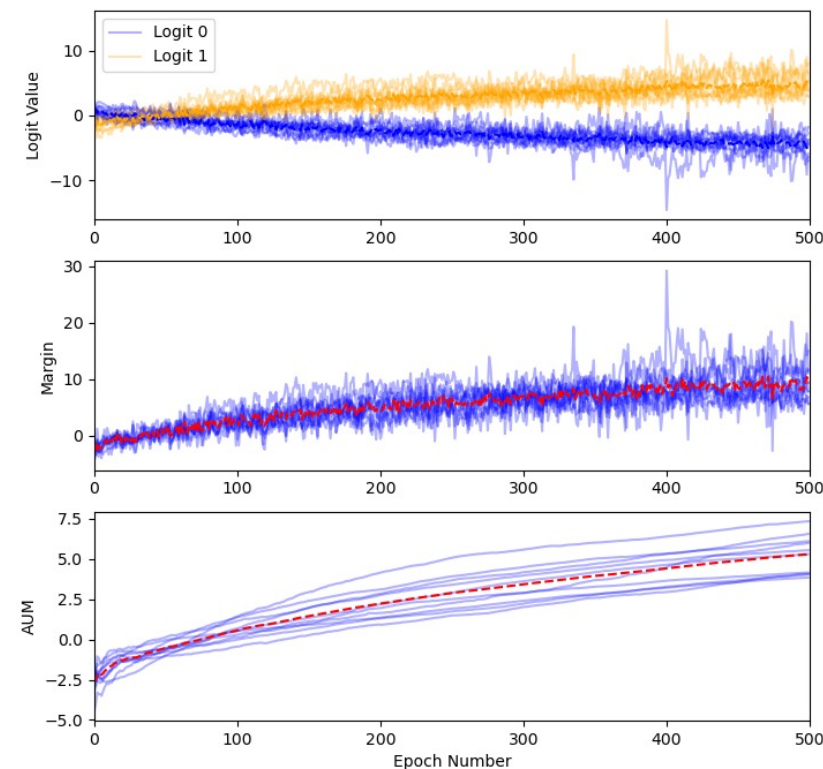
Examples from the Dataset



Kepler-1582 b



Kepler-699 b

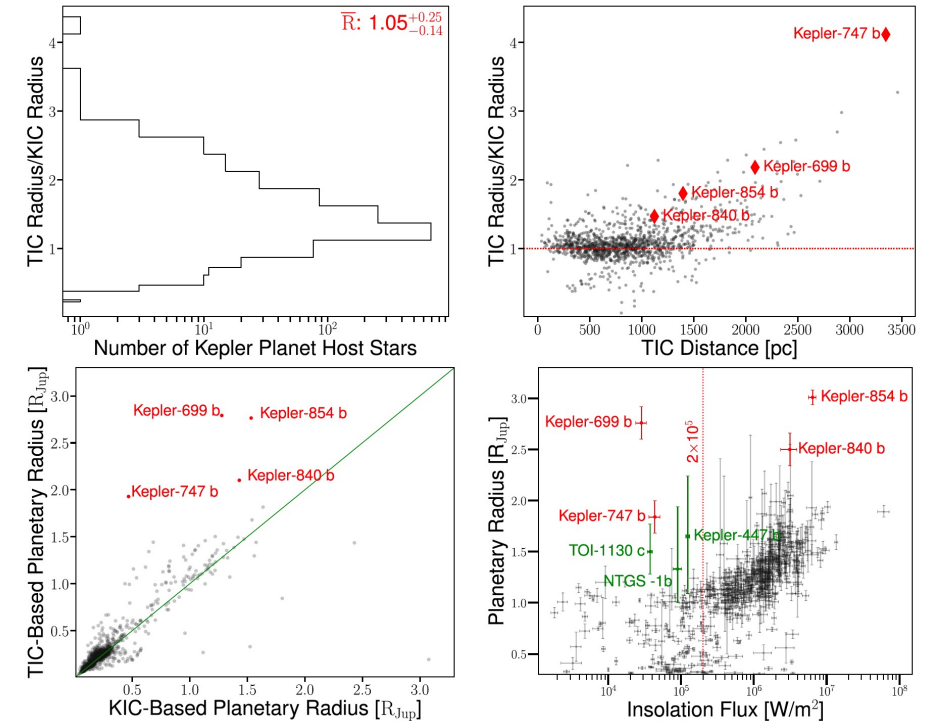


Kepler-1 b

From top to bottom: logits, margins, and AUM values over epochs for a correctly labeled and easy-to-learn example (left), a mislabeled example (center), and a correctly labeled and difficult-to-learn example (right). Logits 0 and 1 are the logits for the PC and FP classes, respectively. Dashed lines represent the mean value across runs.

Results

KIC	TCE Planet Number	Kepler Name	Period (day)	Epoch (BKJD)	ExoMiner Score
10904857	1	Kepler-488 b	3.121	133.224	0.209
7532973	1	Kepler-854 b	2.145	131.840	0.002
6061119	1	Kepler-699 b	27.808	158.906	0.324
11517719	1	Kepler-840 b	2.496	132.840	0.258
3861595	1	Kepler-1658 b	3.849	134.430	0.550
4570949	1	Kepler-76 b	1.545	132.003	0.293
11446443	1	Kepler-1 b	2.471	132.646	0.670
7368664	1	Kepler-434 b	12.875	144.273	0.865
6522242	1	Kepler-706 b	41.408	154.379	0.804



Plots obtained from [3].

[3] Niraula, Prajwal, et al. "Revisiting Kepler Transiting Systems: Unvetting Planets and Constraining Relationships among Harmonics in Phase Curves." *The Astronomical Journal* 163.4 (2022): 172.

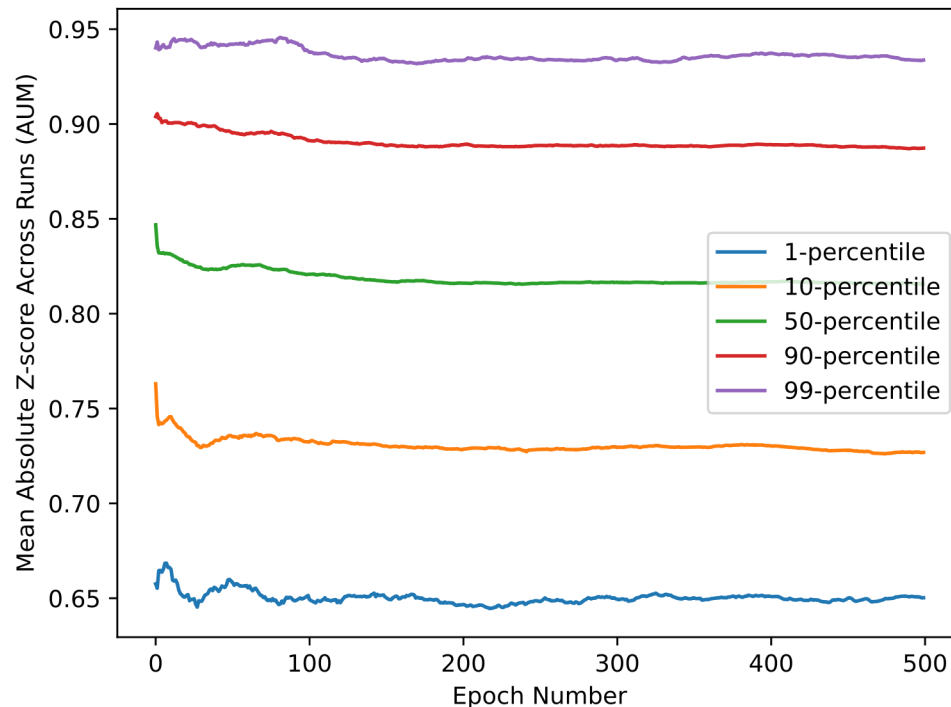
Conclusions & Limitations

- Useful in creating catalogs of mislabeled candidates of confirmed planets that can be used by SMEs for follow-up studies to re-evaluate their status.
- Asymmetric noise on subpopulations of planets and FPs can push mislabeled examples towards higher values of AUM.
 - E.g., brown dwarfs and hot Jupiters.
- The degree of separation between mislabeled and difficult-to-learn correctly labeled examples depends on the model's performance on the classification task and the dataset that it is given.

Extra Slides

Variability in AUM Across Runs

- How good is the mean as an estimate of the average AUM for each example across the 10 runs?
- Computed z-score for each example across the 10 runs, $z_{i,r}$, and reported different percentile values over epochs for the population of z-scores values for all examples, $\bar{z}_i$ s. $z_{i,r}$ describes in terms the distance of each AUM value relative to the mean as a factor of the standard deviation. Most examples have a z-score which is below 1-sigma.



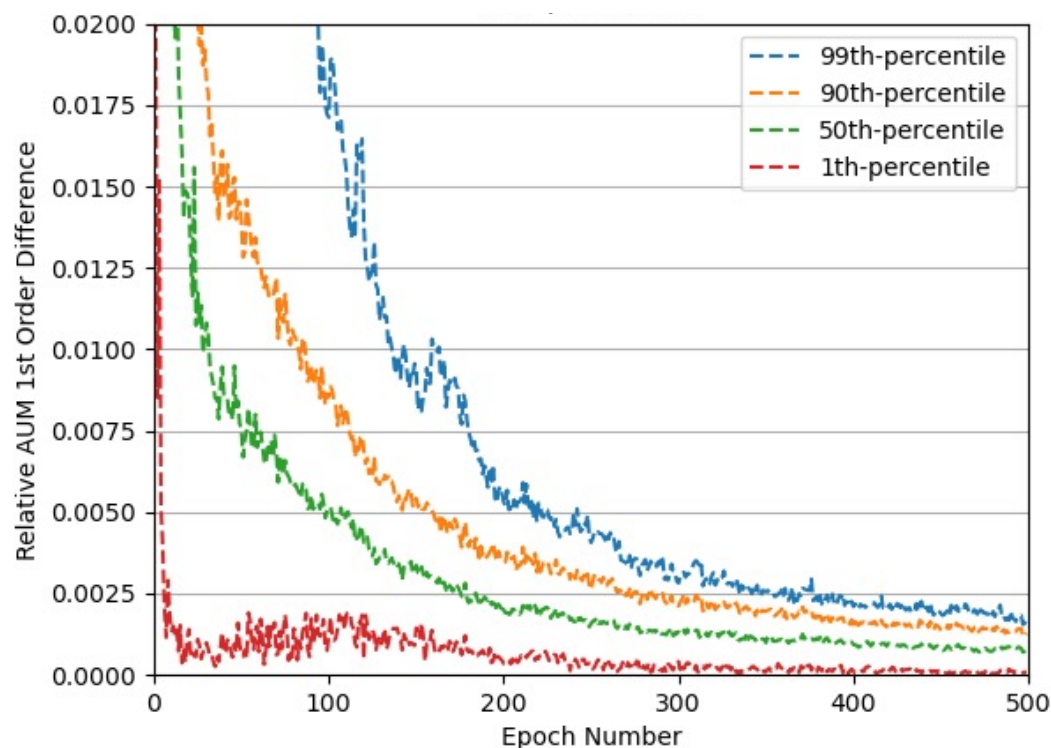
- Variables
 - μ_i : sample mean of AUM for a given example across runs
 - σ_i : sample standard deviation of AUM for a given example across runs

$$z_{i,r} = \text{abs}\left(\frac{x_{i,r} - \mu_i}{\sigma_i}\right)$$

$$\bar{z}_i = \frac{1}{R} \sum_r^R z_{i,r}$$

Convergence of Training Dynamics

- Choosing an epoch to conduct analysis of the AUM ranking.
- Computing mean AUM across runs as estimate of the average AUM for each example over epochs.
- Studying different percentiles of absolute relative 1st order different AUM, $\Delta_{AUM}(x_i, epoch_j)$, as a proxy for training converging to a plateau. This quantity describes the change in AUM between consecutive epochs as a factor of the current AUM value. The normalization puts all examples in the same range.



$$\Delta_{AUM}(x_i, epoch_j) = abs(\frac{AUM(x_i, epoch_j) - AUM(x_i, epoch_{i-1})}{AUM(x_i, epoch_j)})$$

Convergence of Training Dynamics

- **Variables**

- r_1, r_2 : rankings

- **Ranking similarity metrics**

- Overlap at depth d : $o(r_1, r_2, d)$
 - Not top-weighted
 - Non-conjoint
 - Non-indefinite
 - Average Overlap at depth d : $AO(r_1, r_2, d)$
 - Top-weighted
 - Non-conjoint
 - Non-indefinite

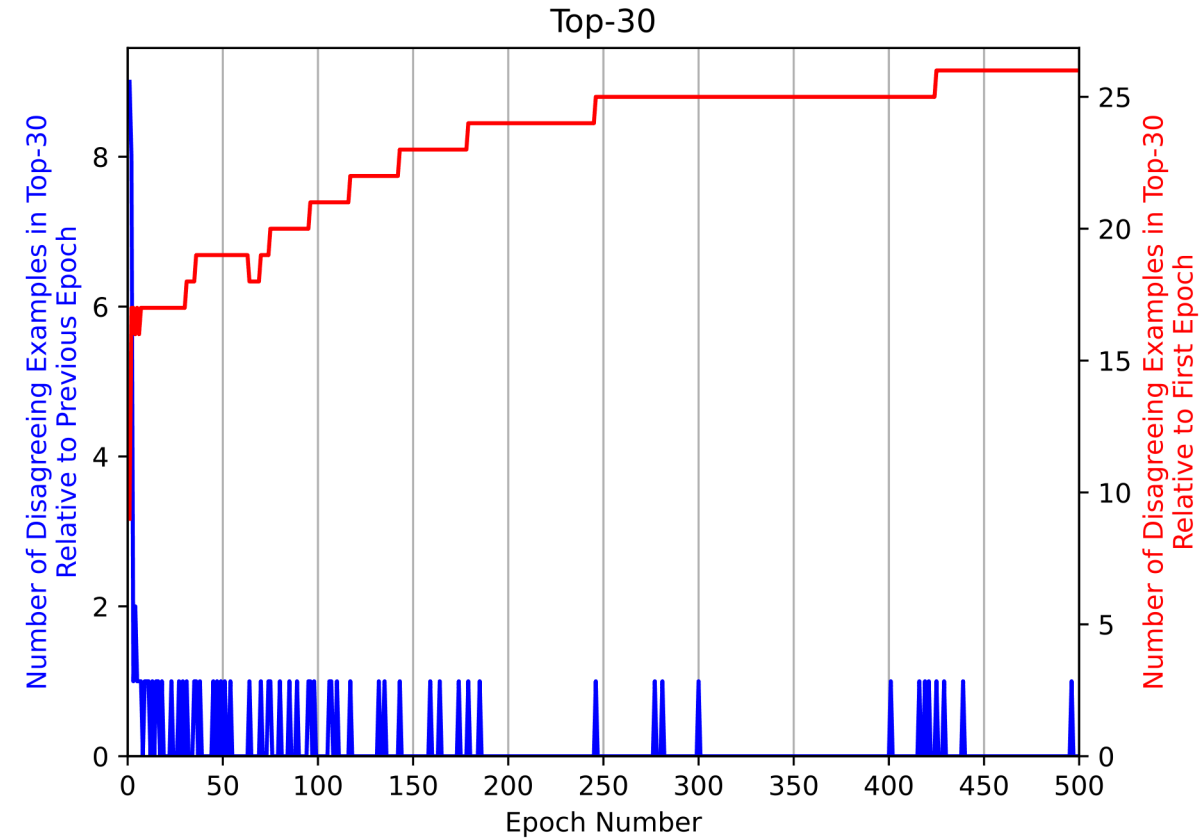
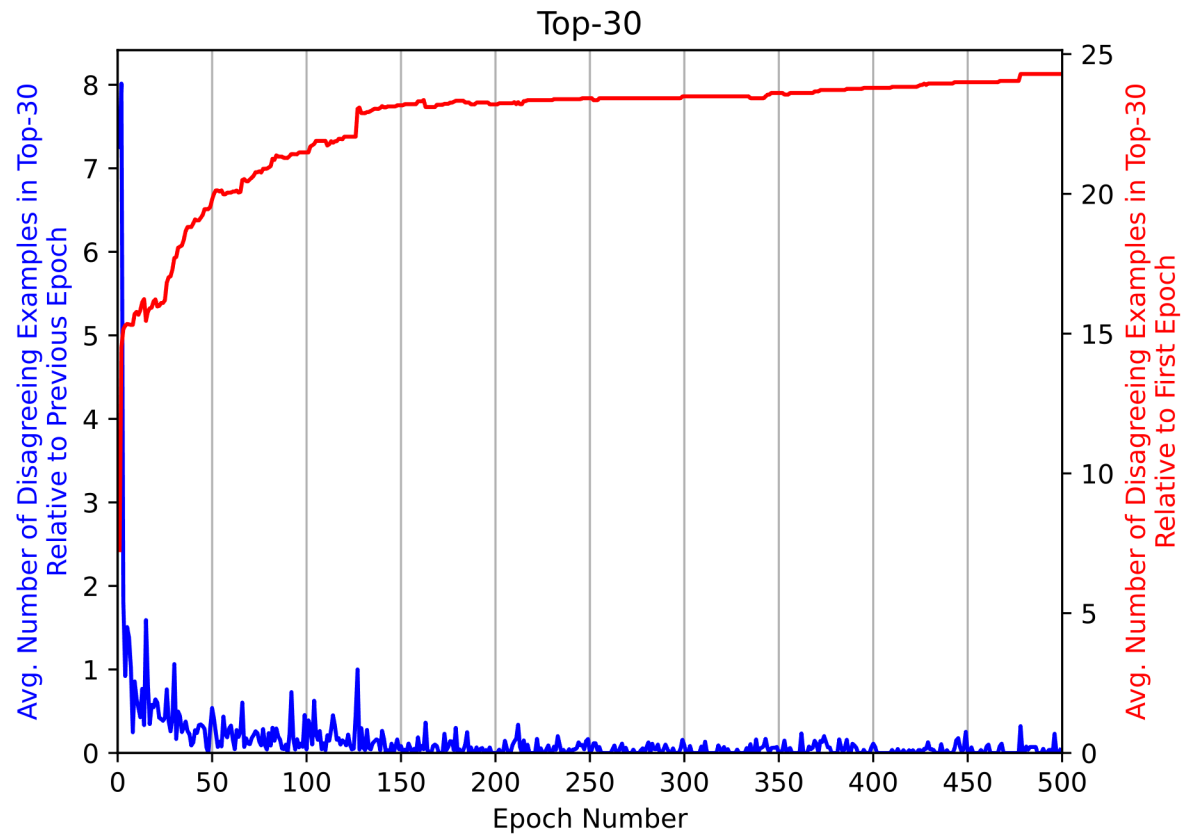
$$o(r_1, r_2, i) = |r_1[:i] \cap r_2[:i]|$$

$$AO(r_1, r_2, d) = \frac{1}{d} \sum_i^d o(r_1, r_2, i)$$

Properties of ranking similarity metrics

- Top-weightedness: assigns more weight to matches at higher positions in the ranking; i.e., takes into account ranking up to depth d .
- Incompleteness (non-conjoint rankings): ranking do not need to share all the same elements between each other (also know as non-conjointness).
- Indefinite: works with incomplete/infinite rankings.

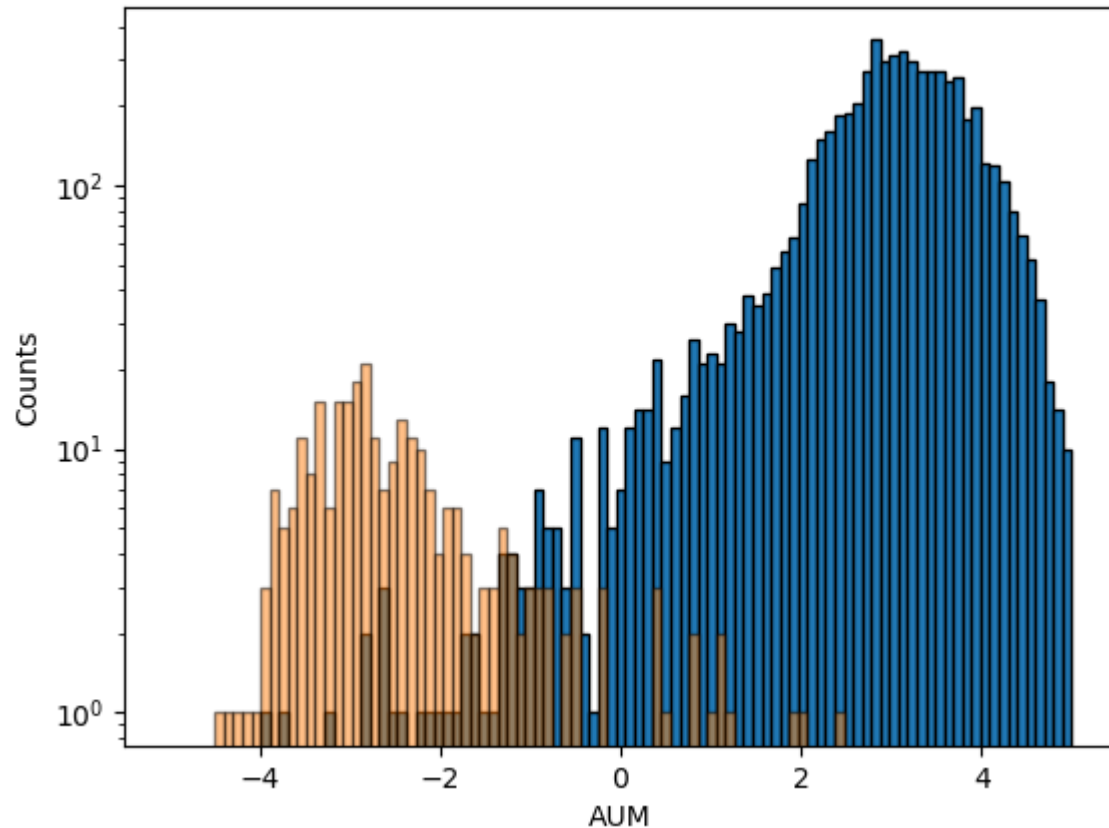
Convergence of Training Dynamics



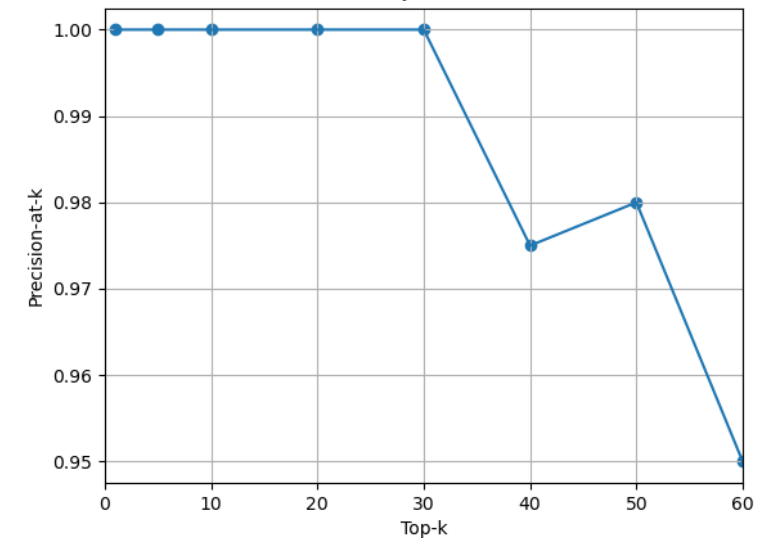
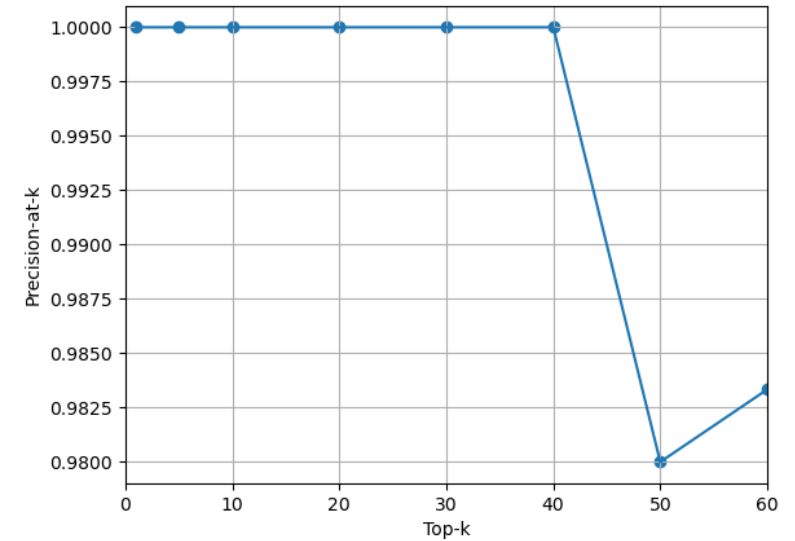
Validation of Method

- Analysis of top-K planet candidates for mislabeling based on AUM value by subject matter experts.
- Track ranking of known or synthetic label noise.
 - **Demoted planets by Niraula et al (2022).**
 - Inject stochastic label noise.
 - Might switch unknown mislabeled examples to correctly labeled.
 - Modelling label noise; several approaches.
 - Class-conditioned label noise: symmetric, asymmetric.
 - Find new subsets of examples based on clustering on features extracted from model after dimensionality reduction (e.g., PCA, t-SNE, UMAP), instead of using PC vs FP formulation (rationale is that subsets of these two classes are more/less prone to mislabeling – e.g., hot Jupiters, brown dwarfs).
 - Feature dependent noise.
 - Use thresholded class as in the AUM paper.
 - Guarantees that all examples in the thresholded class are mislabeled by design.

Stochastic Label Noise

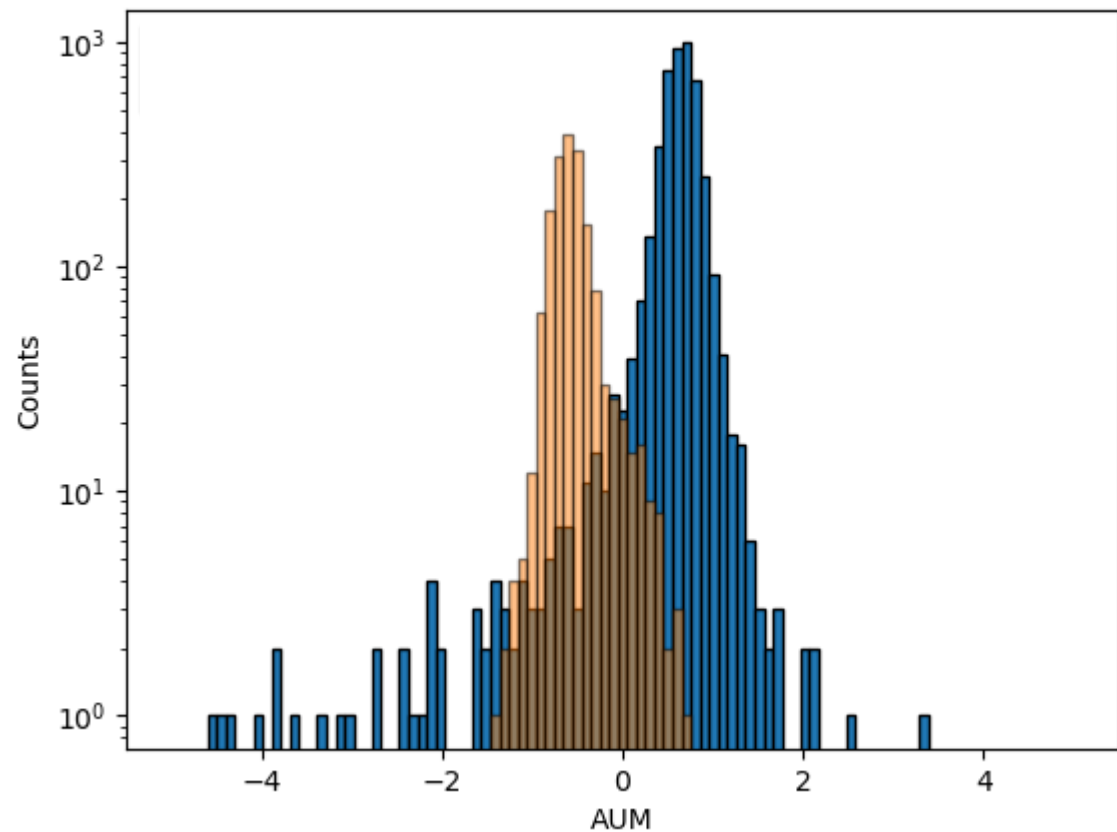


AUM distribution for normal examples (blue) and examples injected with stochastic label noise (orange) for the complete dataset.

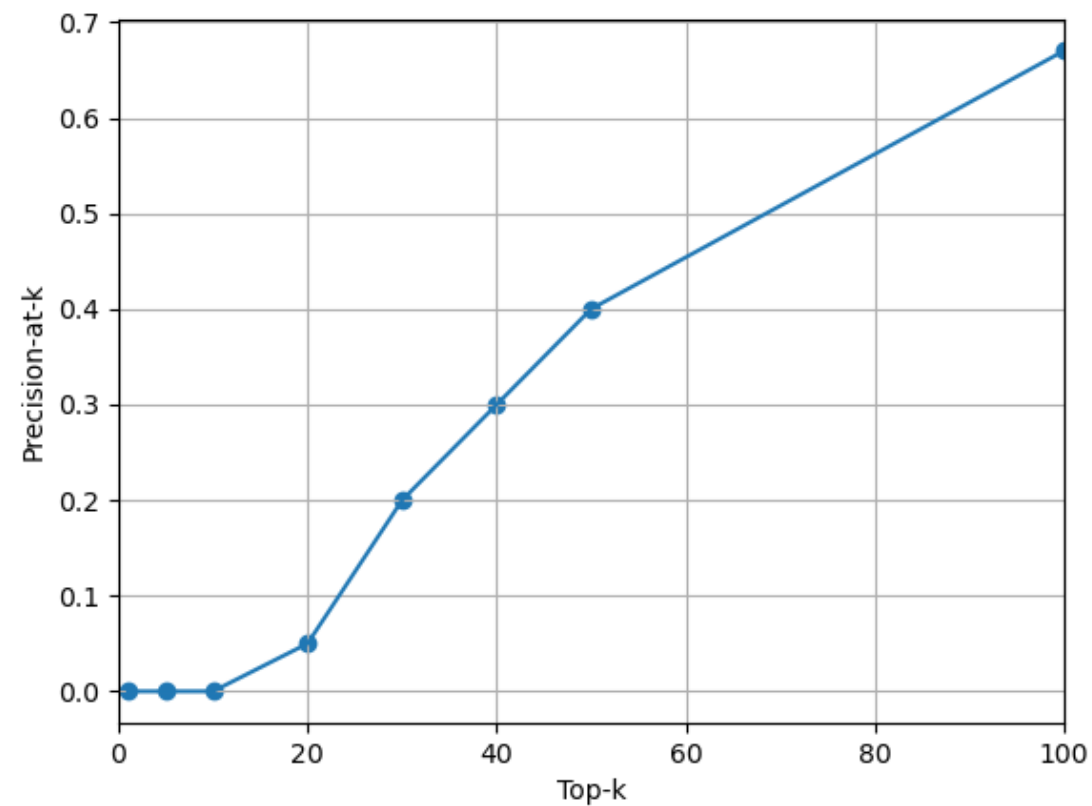


Precision-at-k for train (top) and test (bottom) sets injected with 5% stochastic label noise.

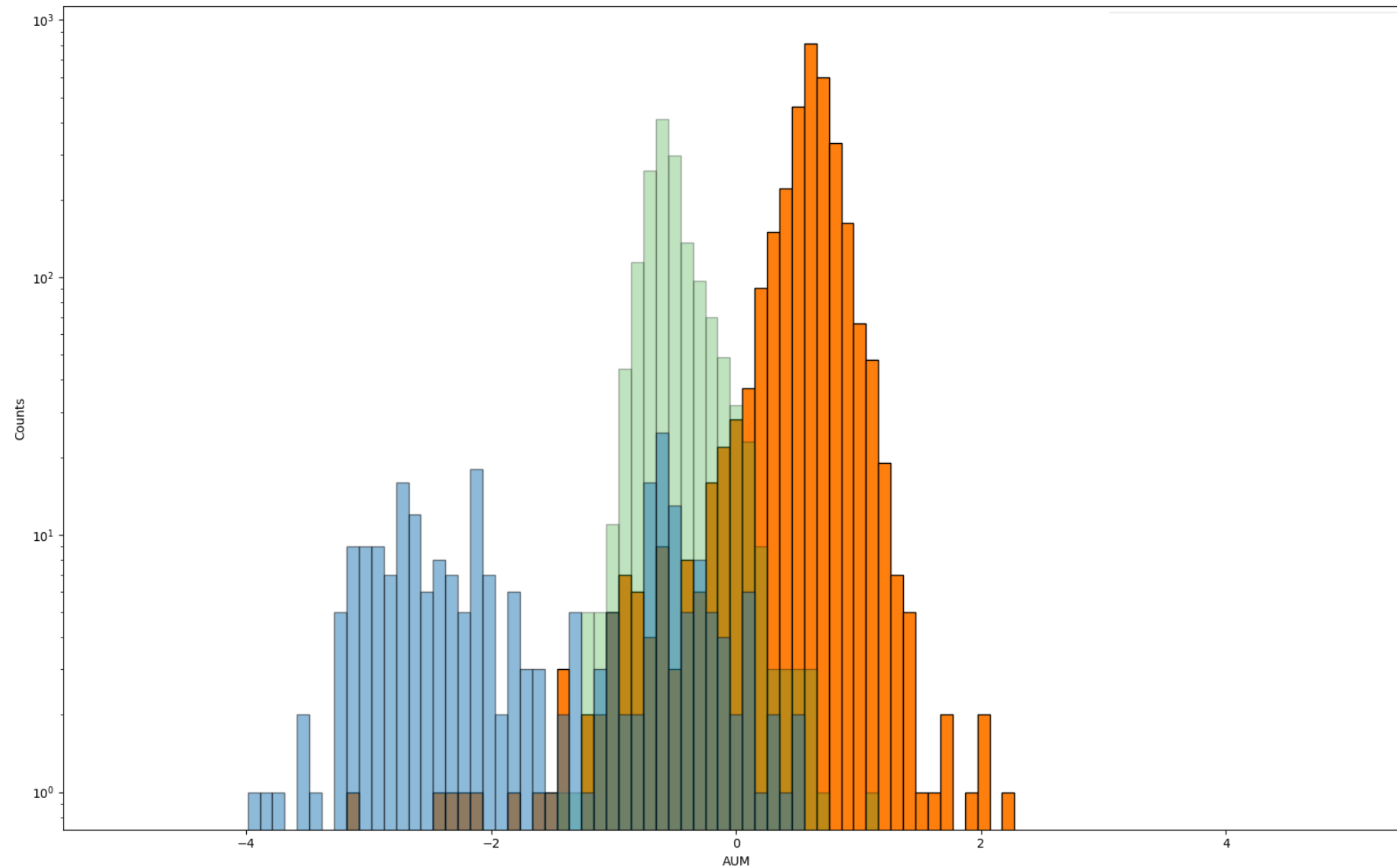
Thresholded Class



AUM distribution for normal (blue) and thresholded (orange) examples for the complete dataset.



Precision-at-k for train set with ~33% thresholded examples.



AUM distribution for normal (orange), thresholded (green), and stochastic label noise injected examples for the complete dataset.