Can error-reduced active learning improve model fairness?

Use active learning to iteratively select samples that have greater impact over model fairness

Wenxuan Huang

Notion of unfairness

- **Disparate treatment (lack of independence)**

  Outcome changes when sensitive attributes changes
  $P(\hat{y}|x, z) \neq P(\hat{y}|x)$

- **Disparate impact (consequence of a lack of independence)**

  Outcomes disproportionally biased against people grouped by sensitive attributes
  $P(\hat{y} = 1|z = 0) \neq P(\hat{y} = 1|z = 1)$

Avoiding disparate treatment (unfairness measurement)

- **Equal opportunity**

  $P(\hat{y} = 1 \mid z = 0, y = 1) = P(\hat{y} = 1 \mid z = 1, y = 1)$ *(**Current method**, plan to use better methods for fairness measurement later)*

- **Disparate mistreatment (equal misclassification rate grouped by sensitive attributes)**

  Misclassification rate $P(\hat{y} \neq y)$ for data points in each sensitive attributes are the same
  $P(\hat{y} \neq y \mid z = 0) = P(\hat{y} \neq y \mid z = 1)$

  Branch of directions:

  False positive rate: $P(\hat{y} = 1 \mid z = 0, y = 0) = P(\hat{y} = 1 \mid z = 1, y = 0)$

  False negative rate: $P(\hat{y} = 0 \mid z = 0, y = 1) = P(\hat{y} = 0 \mid z = 1, y = 1)$

Question

- Research question: Can active learning improve model fairness with unfairness reduction and iterative sampling?

- Sub-question:

  A. For each AL iteration, can the sampling method be used to select potentially biased samples for human labeling (or ground truth)?

  B. Which fairness measurement to be used as loss function to determine potentially biased samples?

  C. Is active learning-based model fairer than the non-AL models?

Methodology

- For each round of active learning iteration, sampling algorithm select a group of samples, determined as samples which brings higher risk of model biases. These samples are removed from the validation pool, and added in the training dataset, with their ground truth.

- Model is retrained, each iteration, by the updated training dataset, presumably improving model fairness.

- Testing dataset, for each iteration, is used to judge (un)fairness for each iteration, to track changes in model fairness.

Data initiation

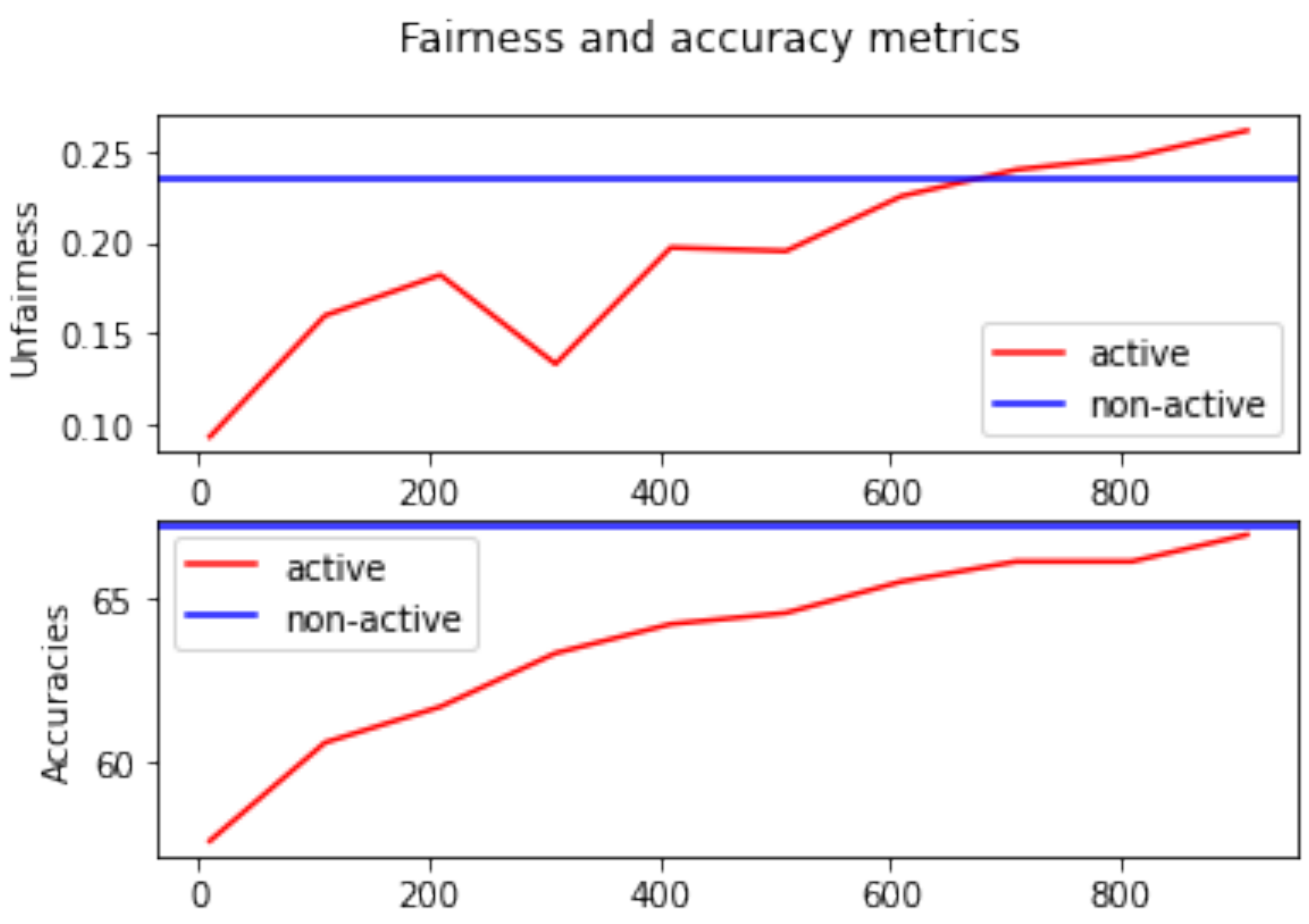
```
$\#D_{total} = n$
$D_{train} = \{\}$
$D_{test} = split(n, fold=5)$
$D_{validation} = D_{total} - D_{test}$ → Unlabelled data U

For each class, #p samples has been selected → $D_{train}$
$D_{validation} = D_{validation} - D_{train}$
$D_{fair\_test} = D_{validation}$
```

Learning & active sample selection

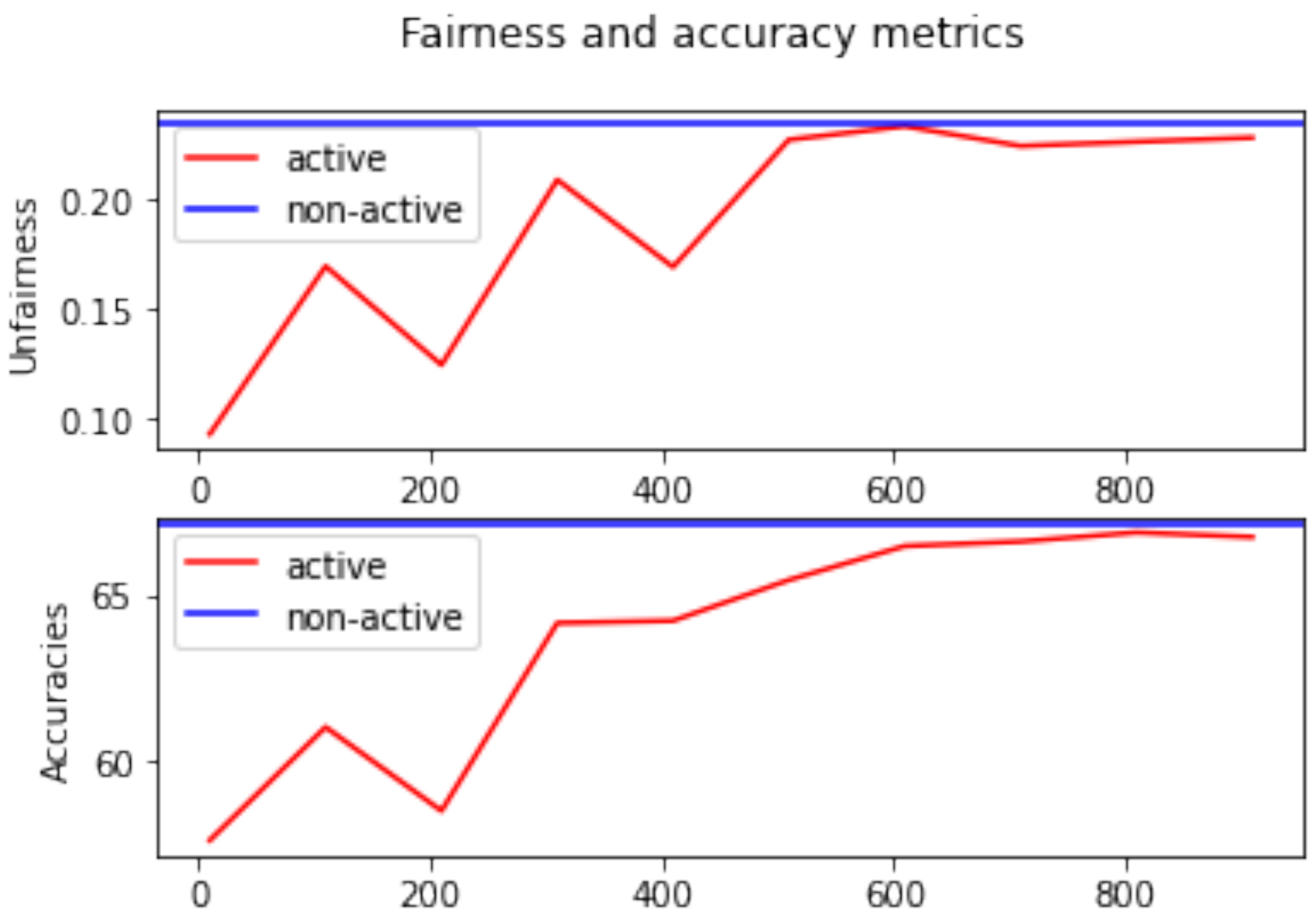
```
**Do** {

    Train
    **Train** a (Logistic) classifier model $M_{Logistic}$ with $D_{train}$
    **Calculate** estimated P^(y|x) for each U in $D_{validation}$
    **Get** unfairness A with $D_{test}$

    Validation and data updates
    **For** X iteration do:

        $Prob_{fair\_test} = M_{Logistic}(D_{fair\_test})$

        **For** each $D_i$ in $D_{fair\_test}$:
            $D_{train} = D_{train} + D_i$
            **Generate** a classification label $L_i$ for $D_i$, $L_i = \{0,1\}$

            **For** L in $L_i$:
                **Retrain** $M_{Logistic}$ for the updated $D_{train}$ with [original label of $D_{train}$ + L]
                **Calculate** Fair Loss F($D_{train}$, [original label of $D_{train}$ + L]) as unfairness
                F = abs($P (\hat{y} = 1 \mid z = 0, y = 1) - P (\hat{y} = 1 \mid z = 1, y = 1)$)
            **Sort** each F from largest to smallest

        **Select** top Q largest F, corresponding $D_i$ with L → $D_{unfair}$
            $D_{validation} = D_{validation} - D_{unfair}$
            $D_{train} = D_{train} + D_{unfair}$

    } **while** ( $D_{validation}$ =! {} and $D_{fair\_test}$ =! {} )

    Test
    Get fairness $A_{test}$ with $D_{test}$
```

Preliminary result

Comparing AL (random sampling) without unfairness reduction to regular Logistic ML algorithm:



Comparing AL with unfairness reduction to regular Logistic ML algorithm:


Fairness and accuracy metrics
Unfairness
active
non-active
Accuracies
active
non-active