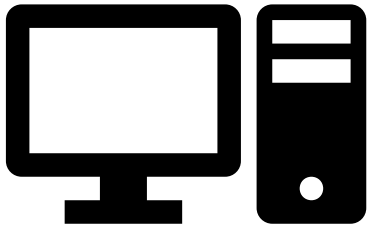
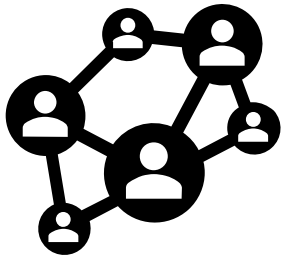
Machine Learning Verification

Brian Hyeongseok Kim
brian.hs.kim@usc.edu

CSCI 698, Fall 2025

Software is everywhere





Software is everywhere

```
int max(int a, int b) {
    int result;
    if (a > b)
        result = a;
    else
        result = a; // BUG: should be b

    return result;
}
```

Testing and Verification

Does φ hold
for some given input?

`max(5,2) = 5` `max(10, -3) = 10` `max(0, -10) = 0` All pass…?

Software is everywhere

```
int max(int a, int b) {
    int result;
    if (a > b)
        result = a;
    else
        result = a; // BUG: should be b

    return result;
}
```

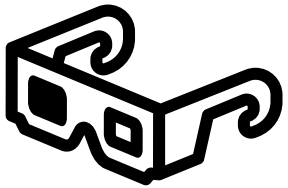
Testing and **Verification**

Does φ hold
for all inputs?

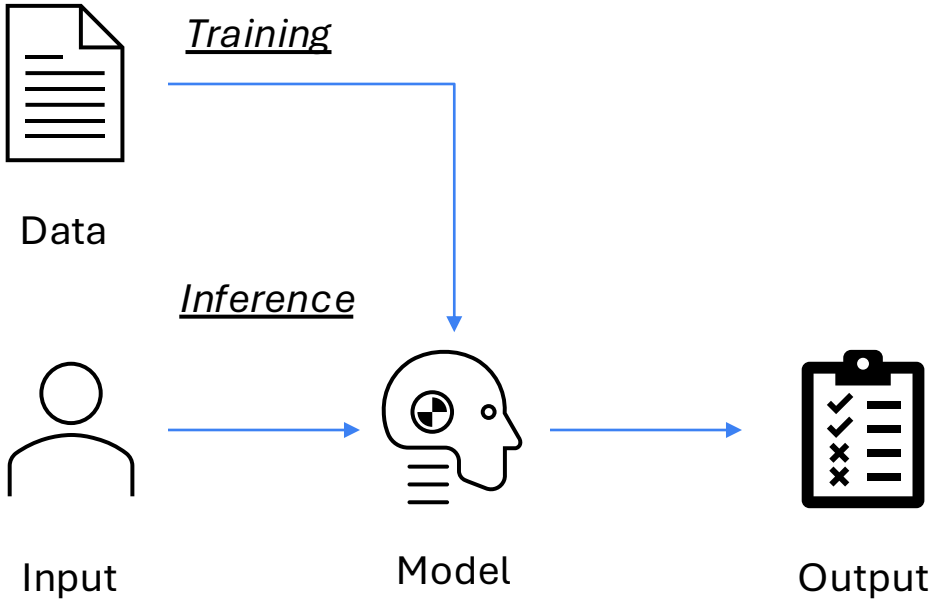
```
INVARIANT: assert(result >= a && result >= b)
```

Violated!

Machine Learning is everywhere

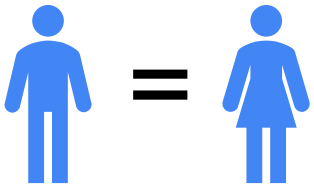


Machine Learning is everywhere

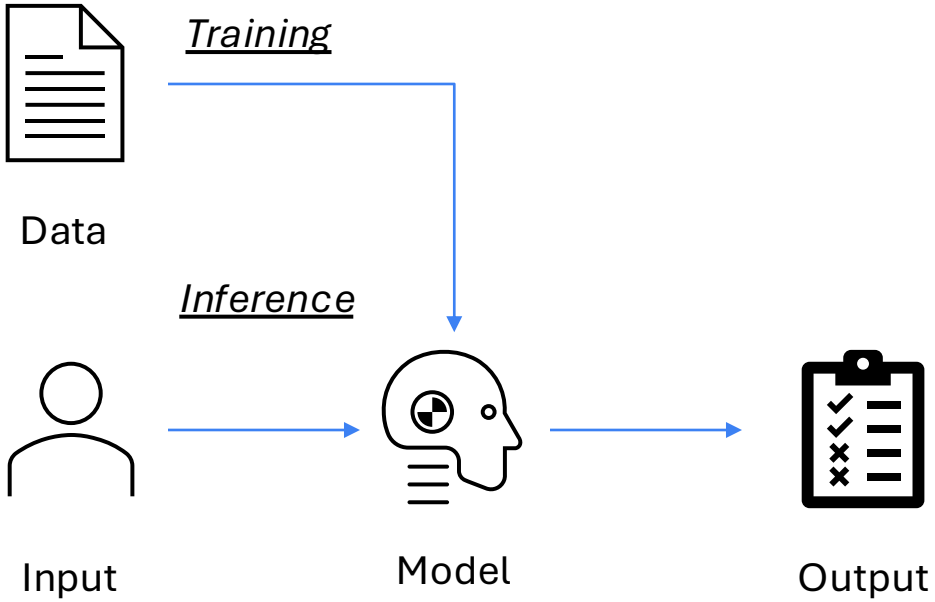


Testing and Verification

Does φ hold
for some given input?

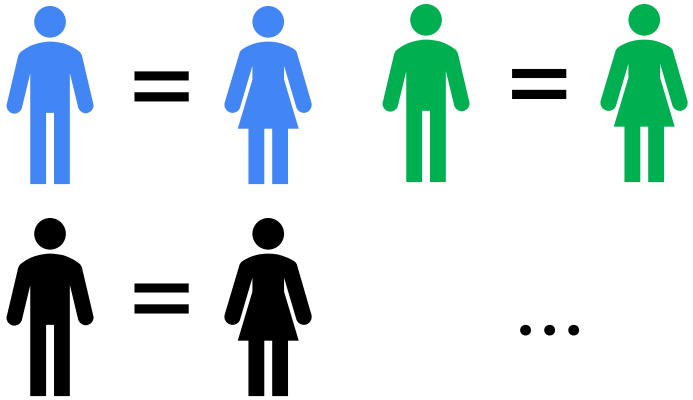


Machine Learning is everywhere



Testing and **Verification**

Does φ hold
for all inputs?



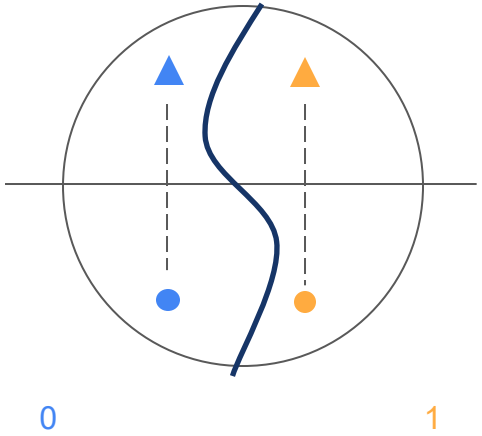
Possible definitions of φ

Non-exhaustive:

1. Counterfactual Fairness
2. Epsilon Fairness
3. Local Robustness
4. Global Robustness

Counterfactual Fairness

Definition 2 (Individual Fairness for the Input Domain): Given a classifier f, an input domain X, and a protected attribute $j \in \mathcal{P}$, we say that f is individually fair for the input domain X if and only if, for all $x \in X$, $f(x) = f(x')$ holds for any $x' \in X$ that differs from x only in the protected attribute x_j.

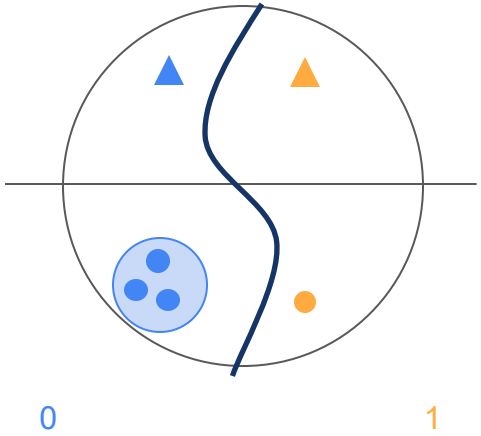


No (x, x') pair **for the entire domain, but with different protected attributes** can have different labels

Kim et al., FairQuant: Certifying and Quantifying Fairness of Deep Neural Networks, ICSE '25

Epsilon Fairness

Definition 2 (ϵ**-Fairness**). *For an input x, the classification output $M(x)$ is fair if, for any input x' such that (1) $x_j \neq x'_j$ for some $j \in \mathcal{P}$ and (2) $|x_i - x'_i| \leq \epsilon$ for all $i \notin \mathcal{P}$, we have $M(x) = M(x')$.*



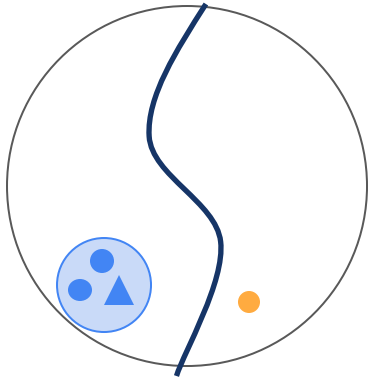
No (x, x') pair within epsilon distance **for a given x, but with same protected attribute** can have different labels

Li et al., Certifying the Fairness of KNN in the Presence of Dataset Bias, CAV '23

Local Robustness

Definition 1. *(Local Robustness) A model, F, is* ϵ-locally-robust *at point, x, with respect to norm, $||\cdot||$, if* $\forall\, x'$,

$$||x - x'|| \leq \epsilon \implies F(x) = F(x').$$



No (x, x') pair within epsilon distance **for a given x** can have different labels

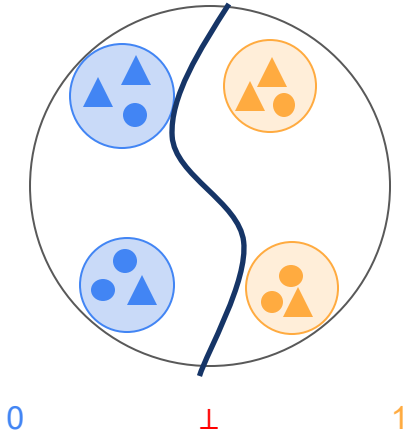
Leino et al., Globally-Robust Neural Networks, ICML '21

Global Robustness

Definition 2. *(Global Robustness) A model, F, is ϵ-globally-robust, with respect to norm, $||\cdot||$, if $\forall\, x_1, x_2$,*

$$||x_1 - x_2|| \leq \epsilon \implies F(x_1) \stackrel{\perp}{=} F(x_2).$$

c1 $\perp=$ c2 if (c1=$\perp$) || (c2=$\perp$) || (c1=c2), where
$\perp$ labels "not locally robust" points along the margin



No (x1, x2) pair within epsilon distance **for the entire domain** can have both 0 and 1 labels

Leino et al., Globally-Robust Neural Networks, ICML '21

Conclusion

- Difference between testing and verification
- Why we need verification for machine learning models
- Various popular properties related to fairness and robustness
- Motivating examples