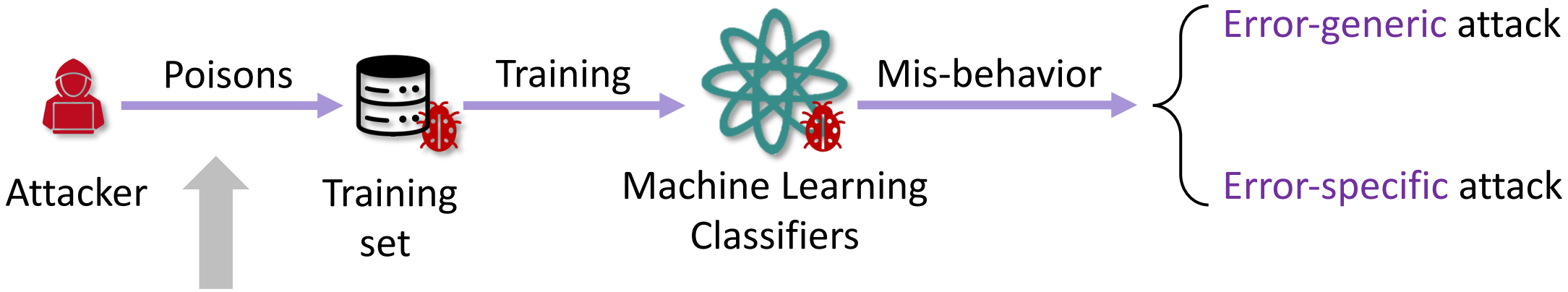
# BIC-based Mixture Model Defense against Data Poisoning Attacks on Classifiers

**Xi Li**, David Miller, Zhen Xiang, and George Kesidis

IEEE Transactions on Knowledge and Data Engineering (TKDE), 2024

PennState

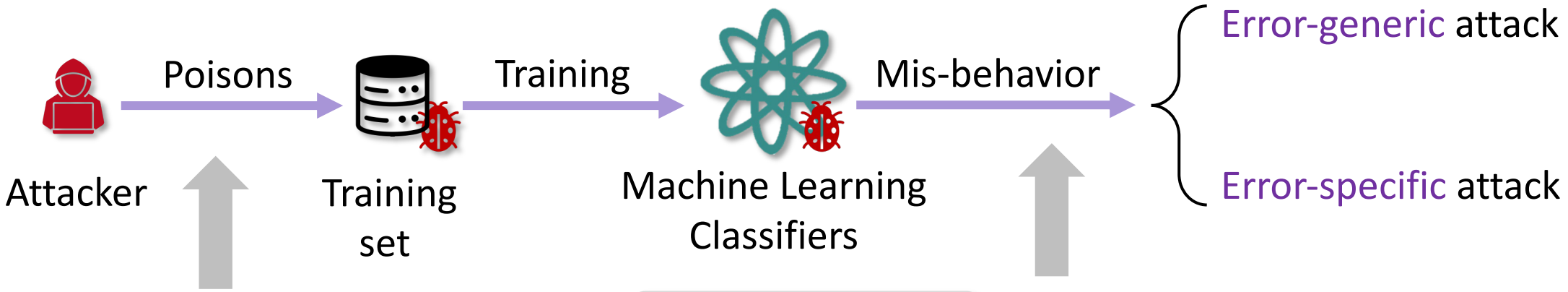
# Poisoning Attacks




Attacker's **ability**:

Inject malicious instances into the training set of the victim model.

# Poisoning Attacks



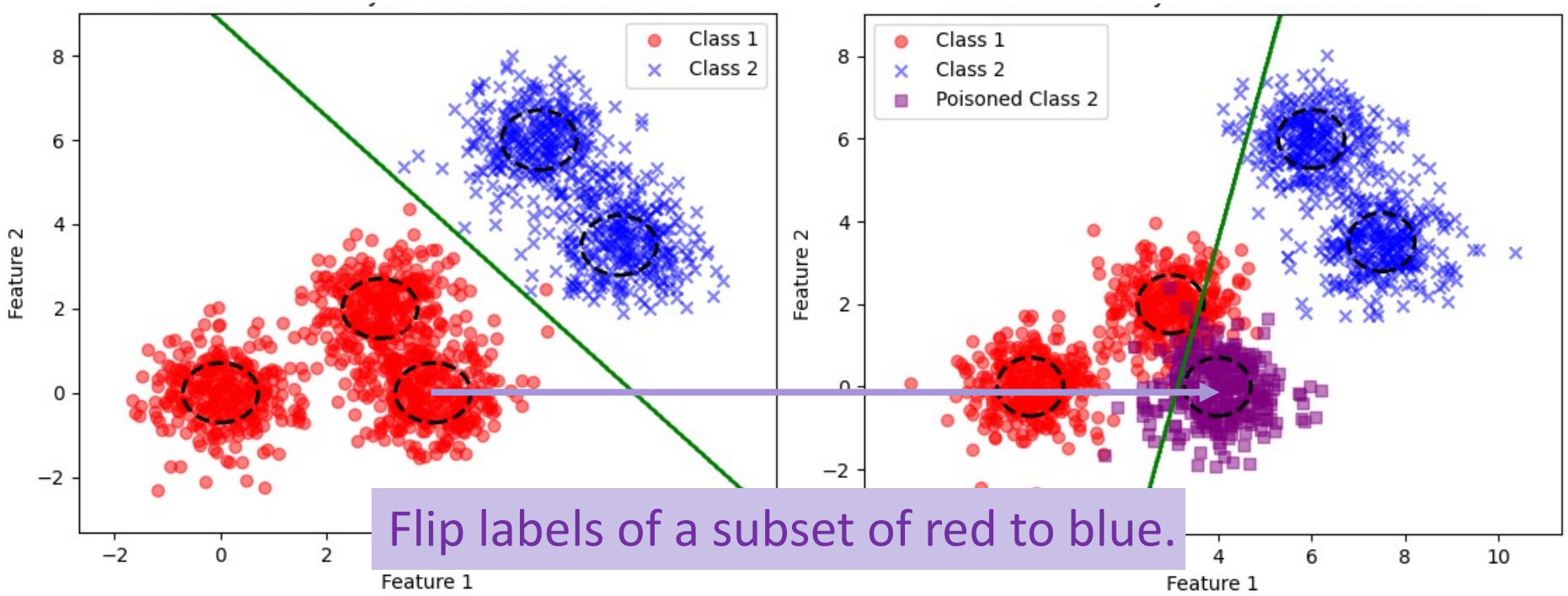

Attacker's **ability**:

Inject malicious instances into the training set of the victim model.

Attacker's **goal**:

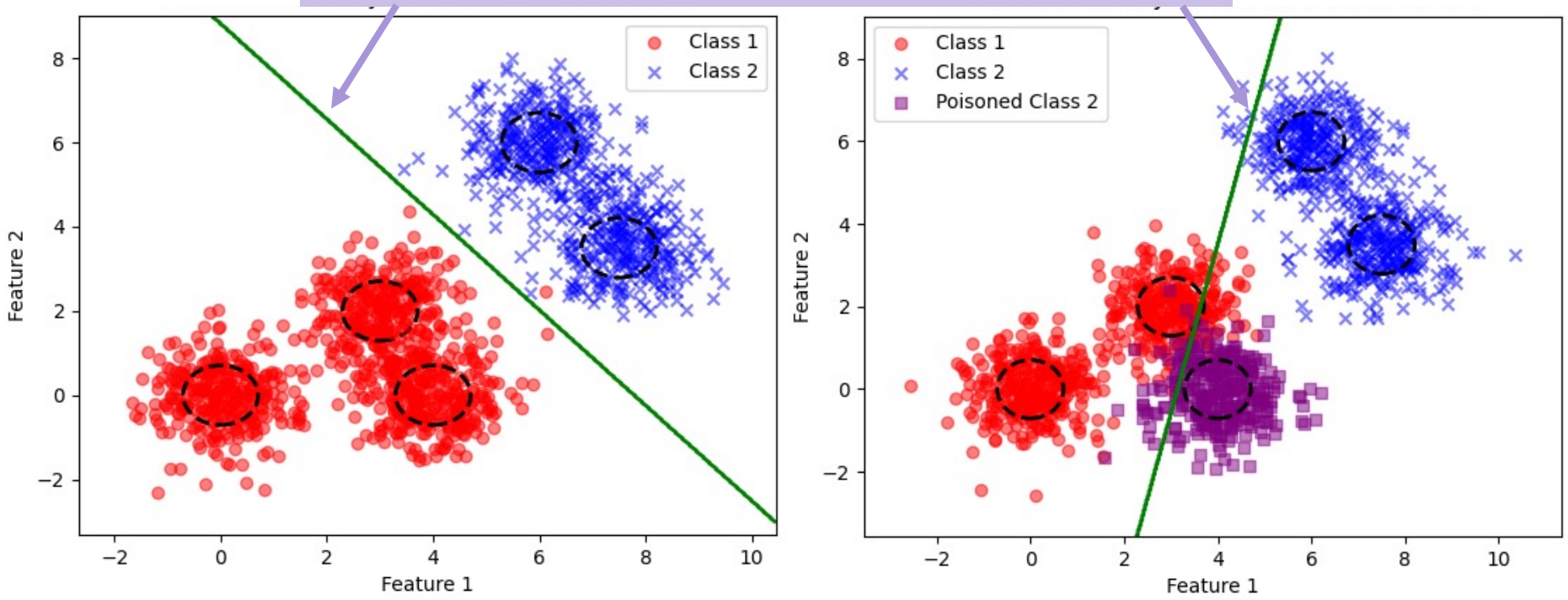
Error-generic attack, degrading the overall performance.
Error-specific attack, causing mis-classification only for specific samples/classes.

# Poisoning Attacks - Label Flipping Attack



# Poisoning Attacks - Label Flipping Attack

Decision boundary significantly changes.

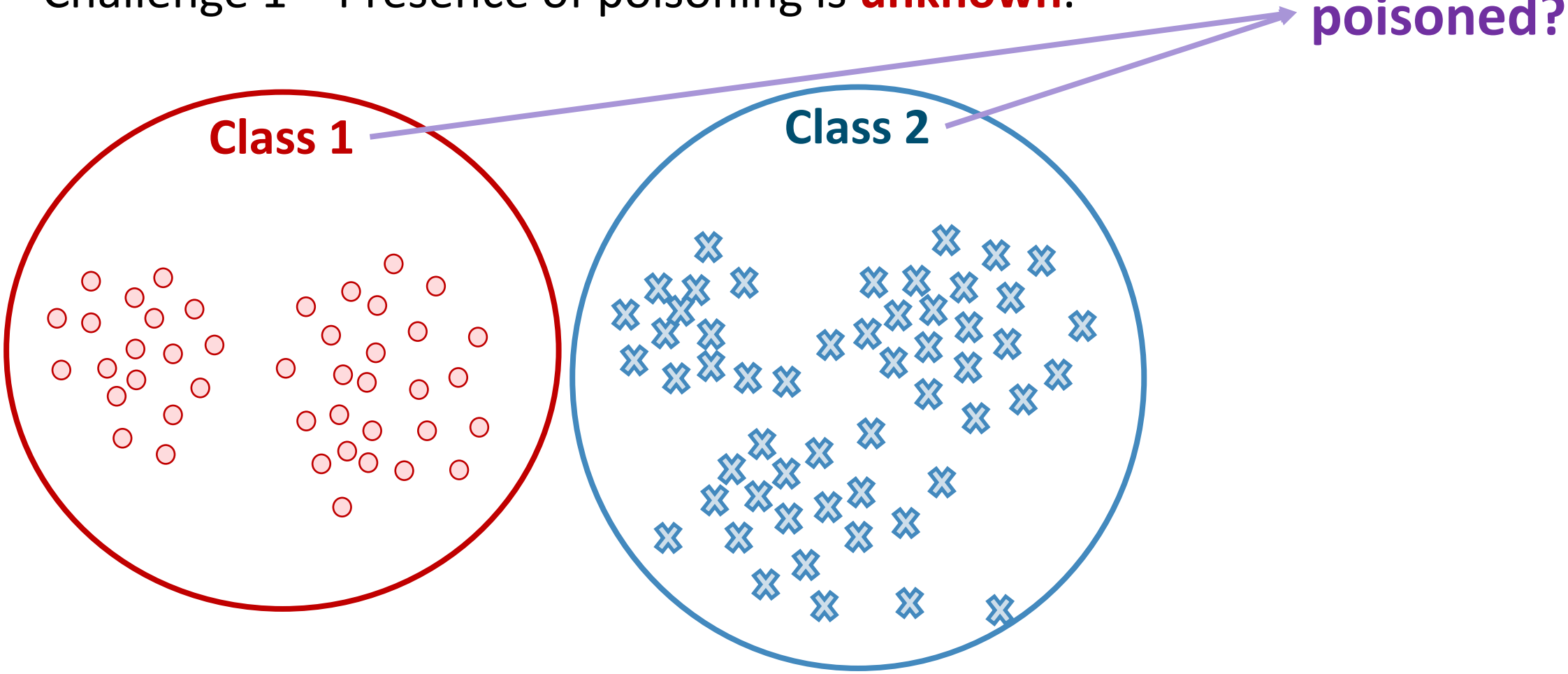


# Limitation of Existing Work

- Hyper-parameter tuning
- Suitable for specific classifiers

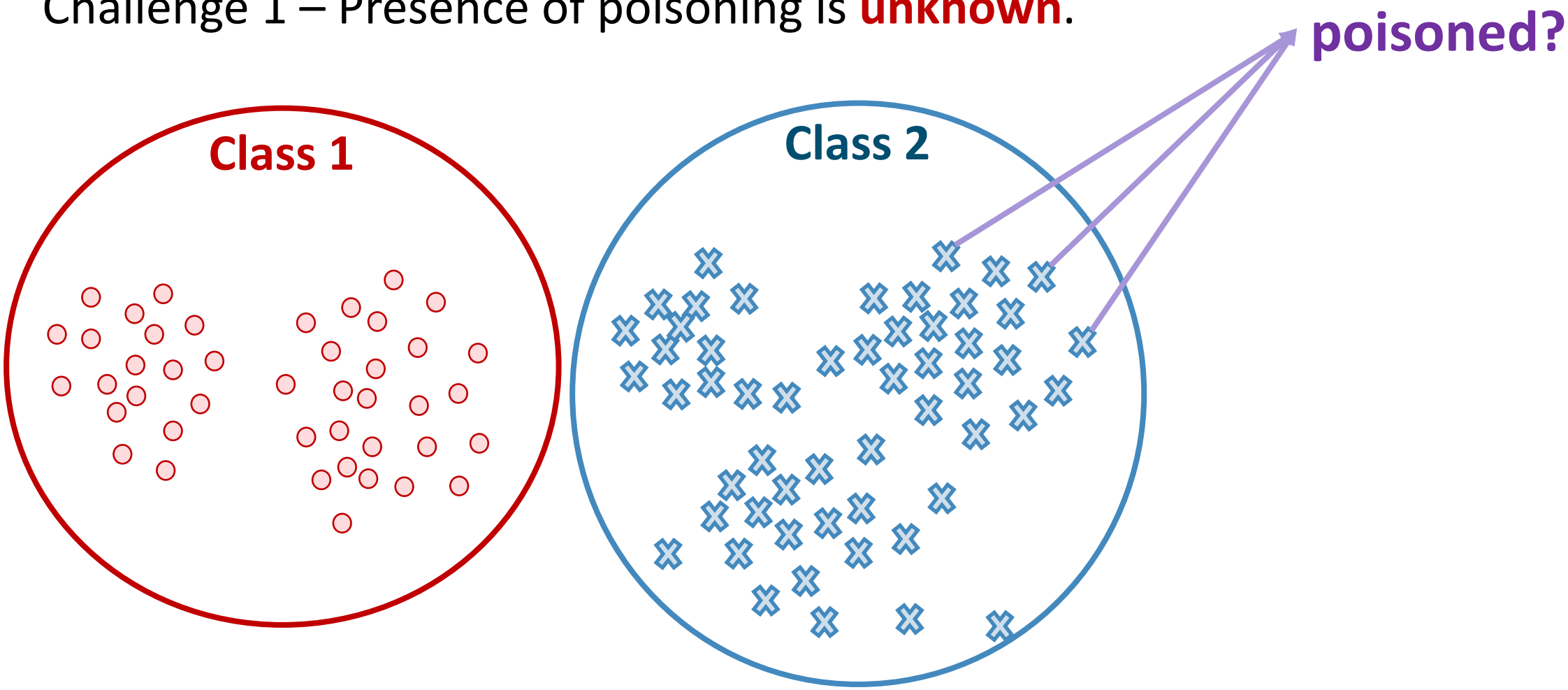
# Challenges in Addressing Label-Flipping Attacks

Challenge 1 – Presence of poisoning is **unknown**.



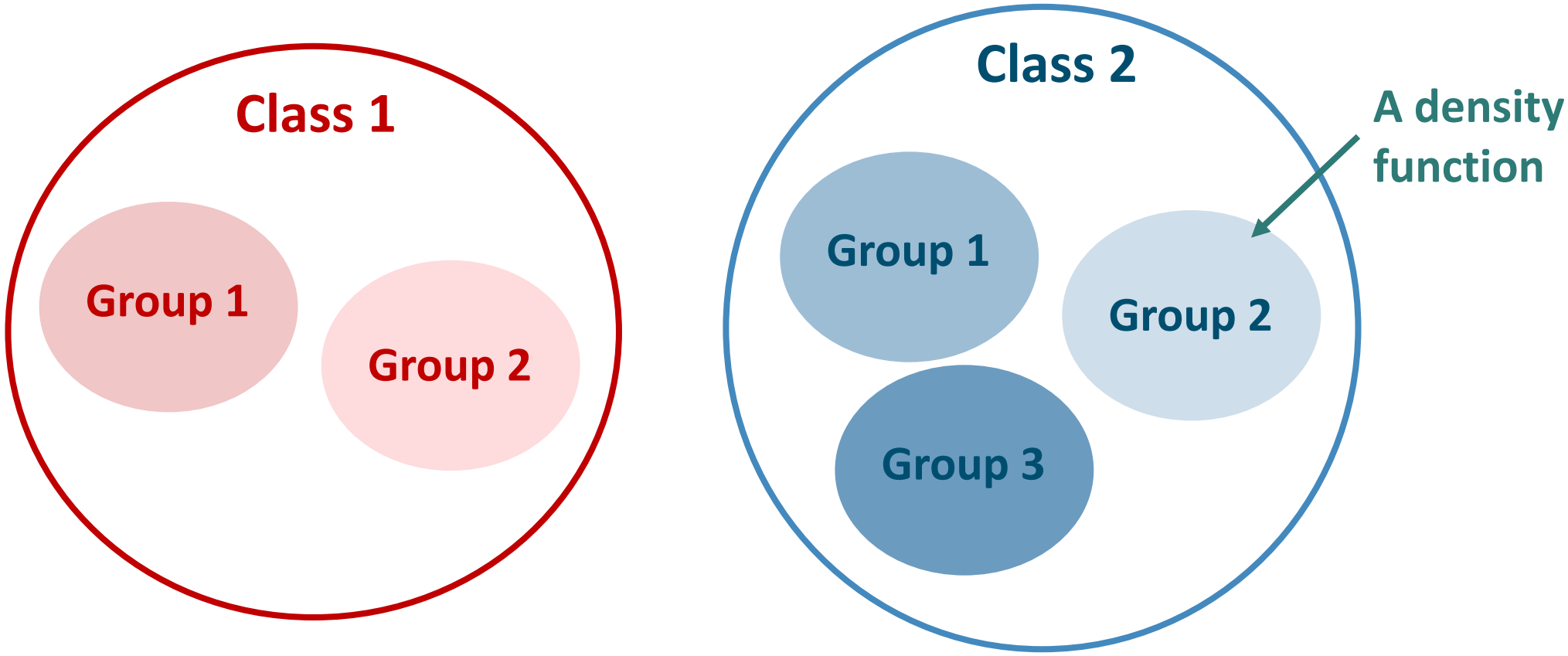
# Challenges in Addressing Label-Flipping Attacks

Challenge 1 – Presence of poisoning is **unknown**.



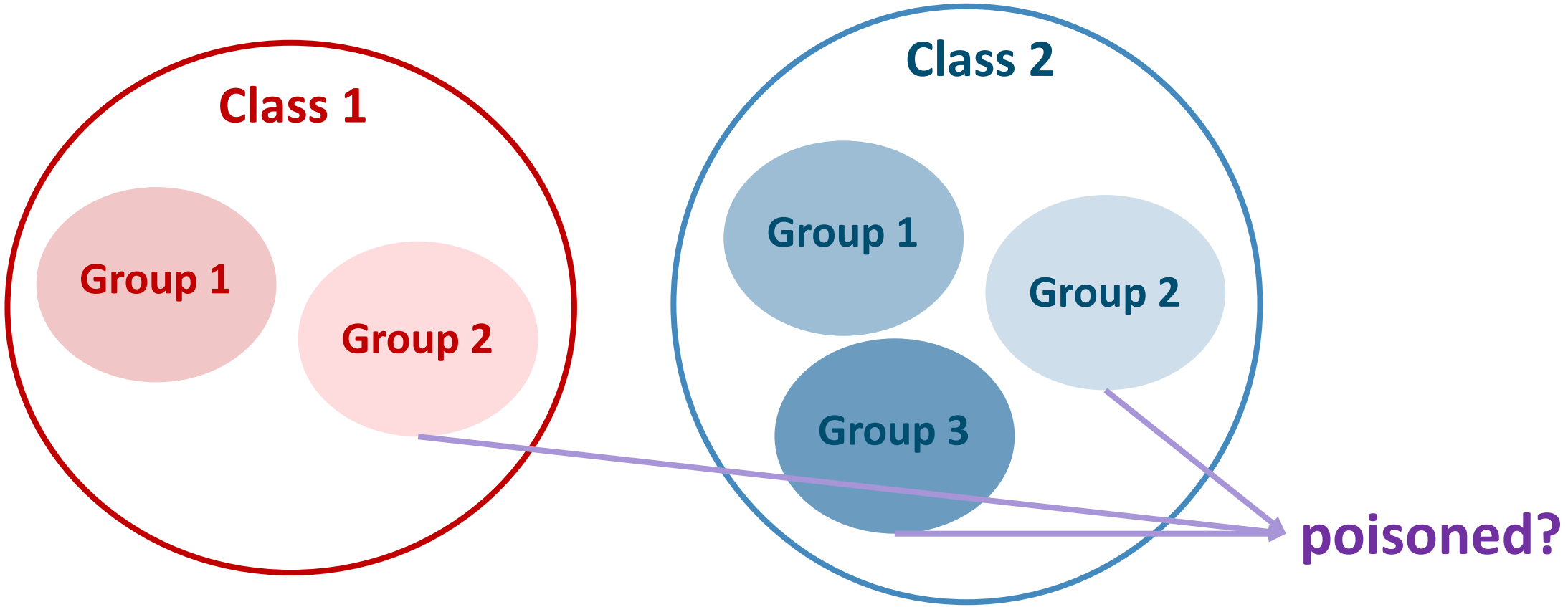
# Solution for Addressing Label-Flipping Attacks

Solve challenge 1 -- **Isolate** poisoned samples from clean samples.



# Challenges in Addressing Label-Flipping Attacks

Challenge 2 – **No clean** samples available for detection.

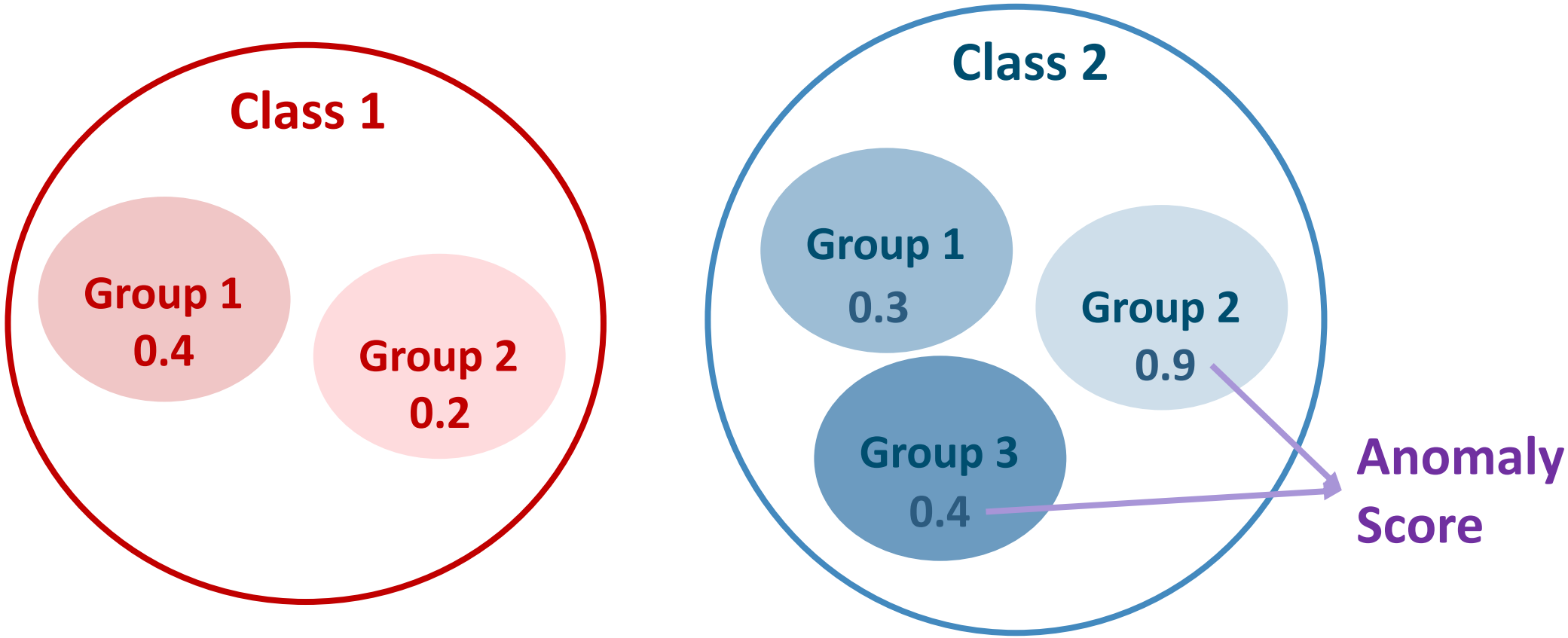





PennState

# Solution for Addressing Label-Flipping Attacks

Solve challenge 2 – Identify and remove the **most likely poisoned** group in each step.

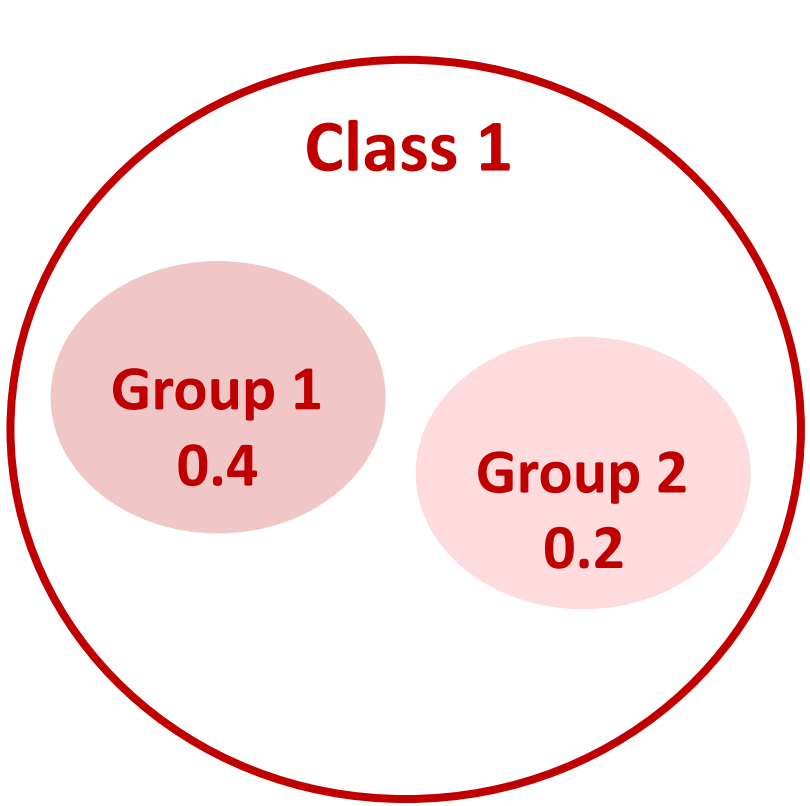


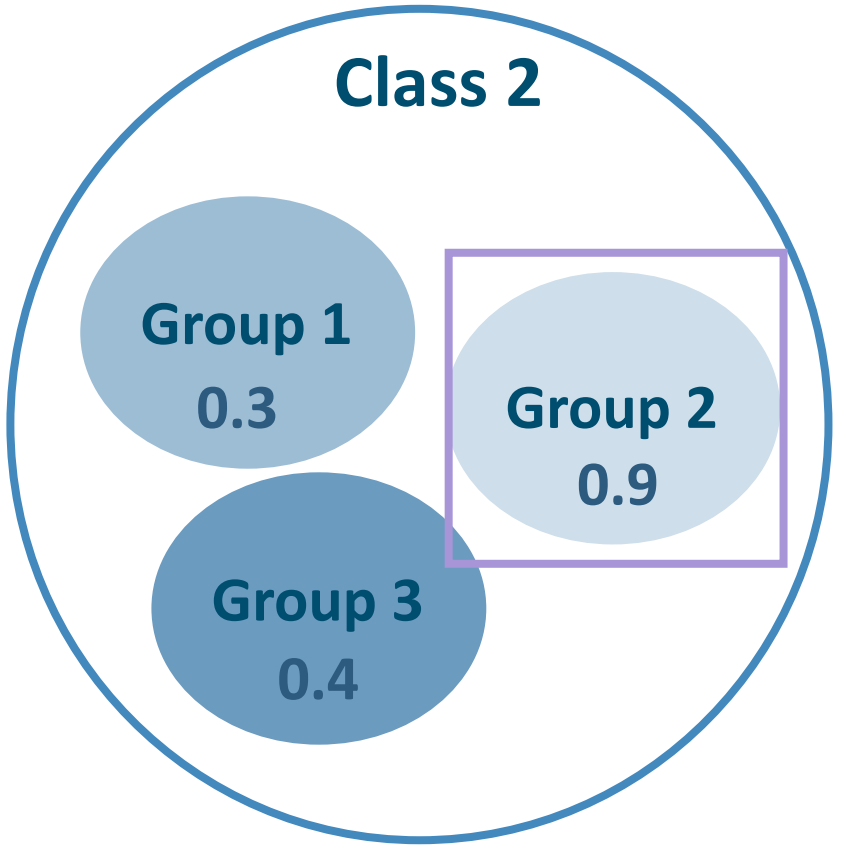



PennState

# Solution for Addressing Label-Flipping Attacks

Solve challenge 2 – Identify and remove the **most likely poisoned** group in each step.

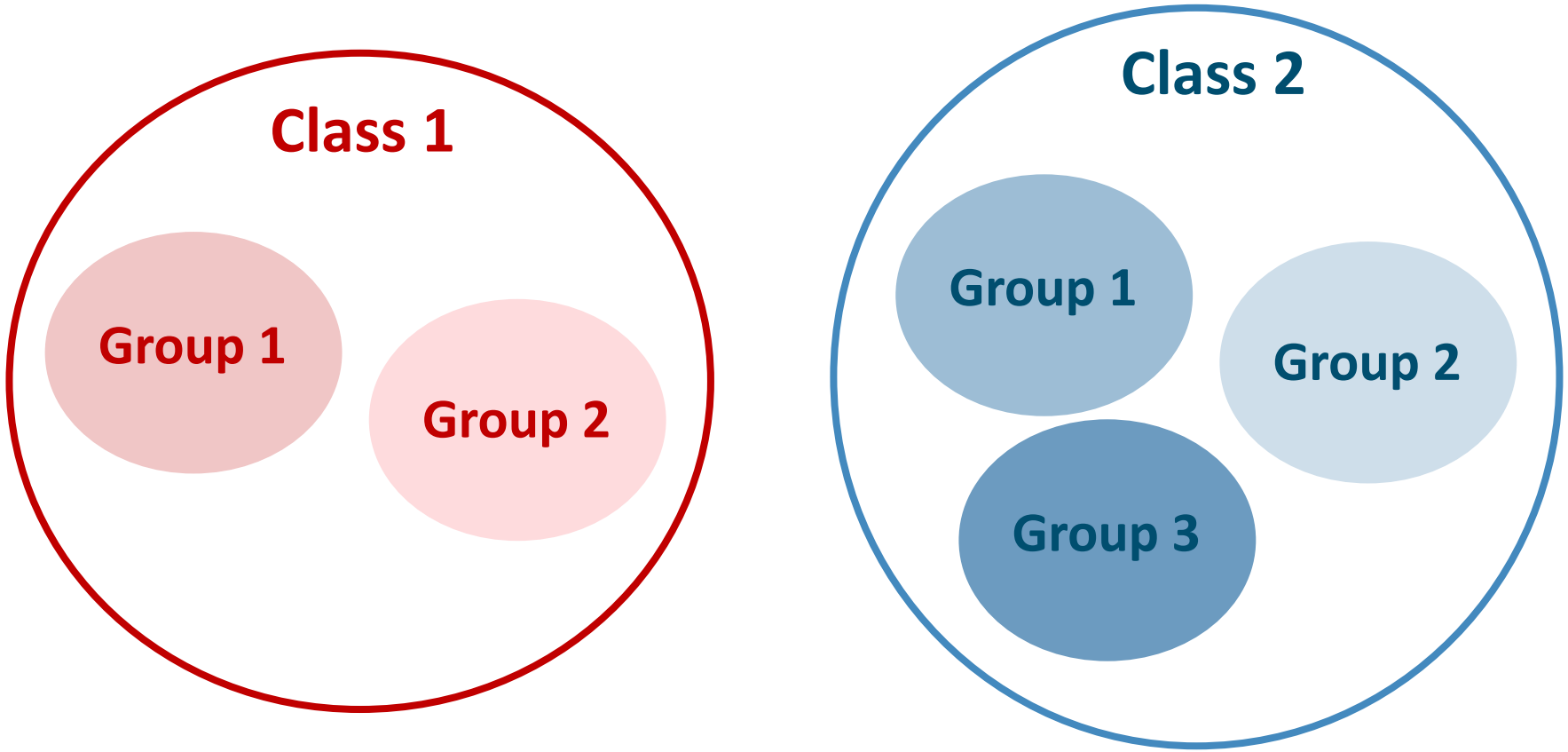



**Poisoned. Remove it.**

# Challenges in Addressing Label-Flipping Attacks

Q: How to identify the **most likely poisoned** group?

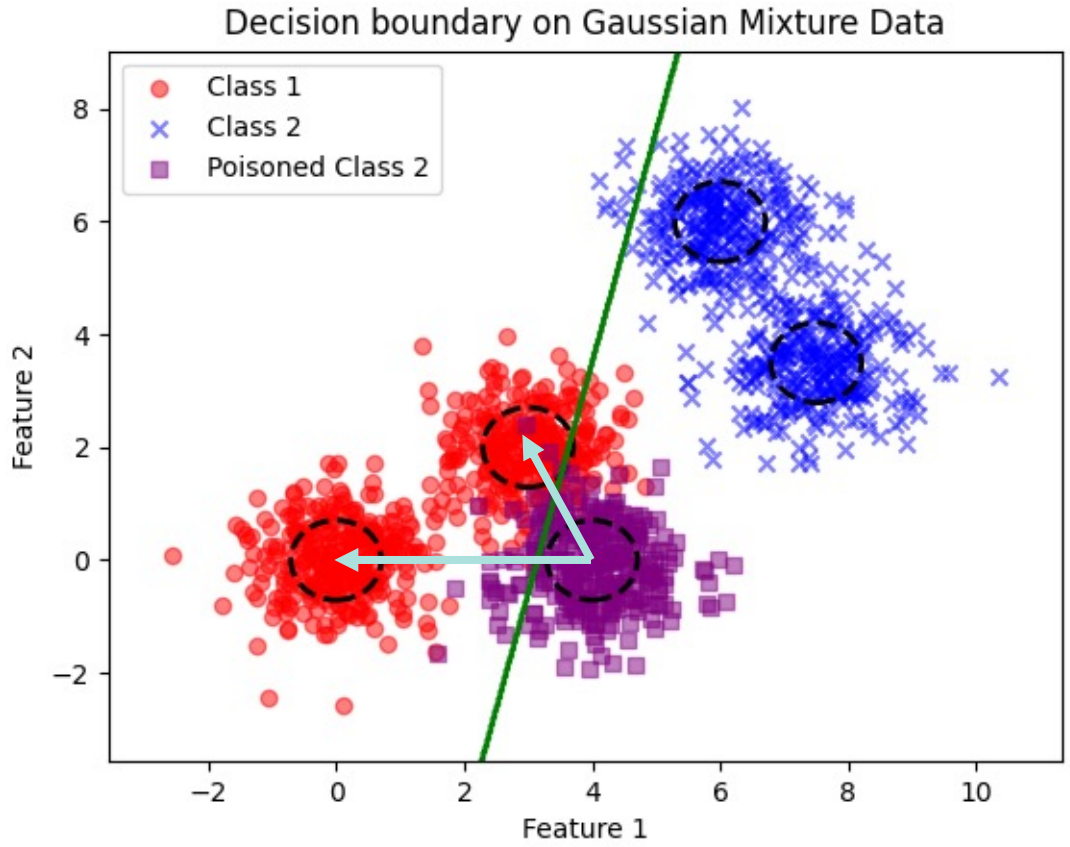


# Solution for Addressing Label-Flipping Attacks

Q: How to identify the **most likely poisoned** group?

Observation 1:

The poisoned group is **better represented** by density functions of red.






PennState

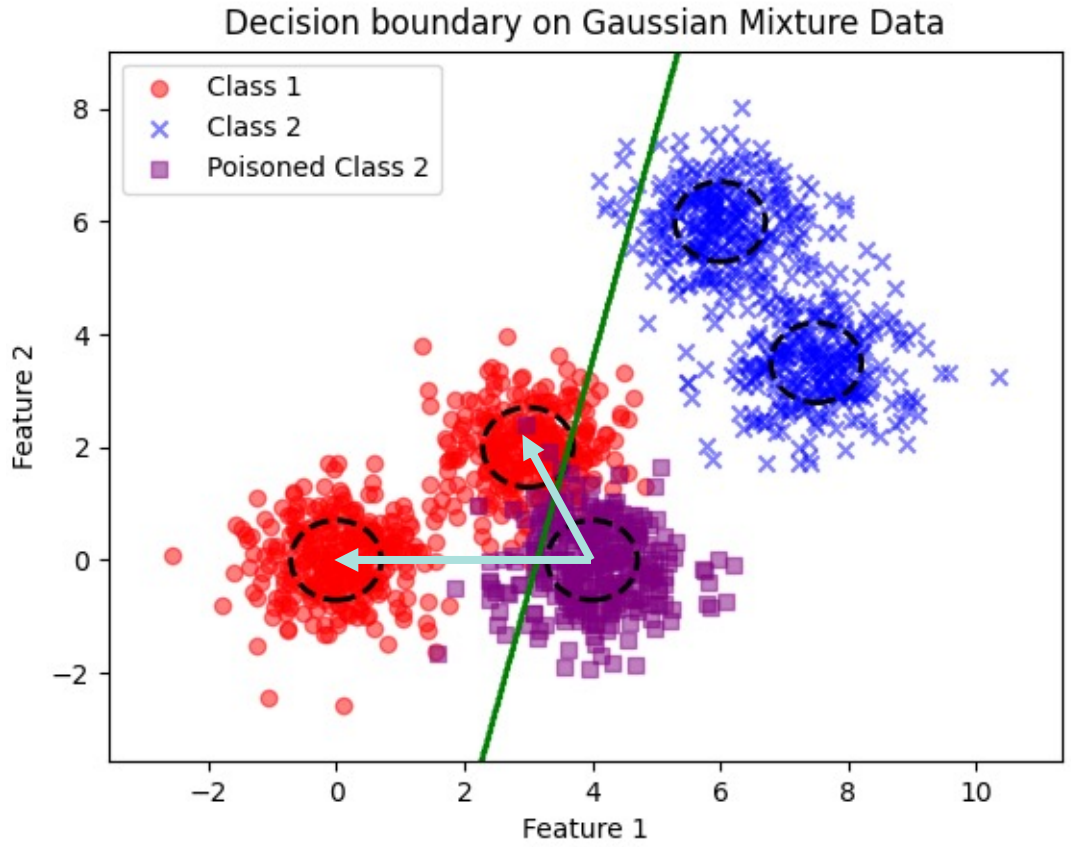
# Solution for Addressing Label-Flipping Attacks

Q: How to identify the **most likely poisoned** group?

Observation 1:

The poisoned group is **better represented** by density functions of red.

Re-assign poisoned samples to red would **increase data likelihood**.




BIC-based Mixture Model Defense against Data Poisoning Attacks on Classifiers. Xi Li et al. TKDE.



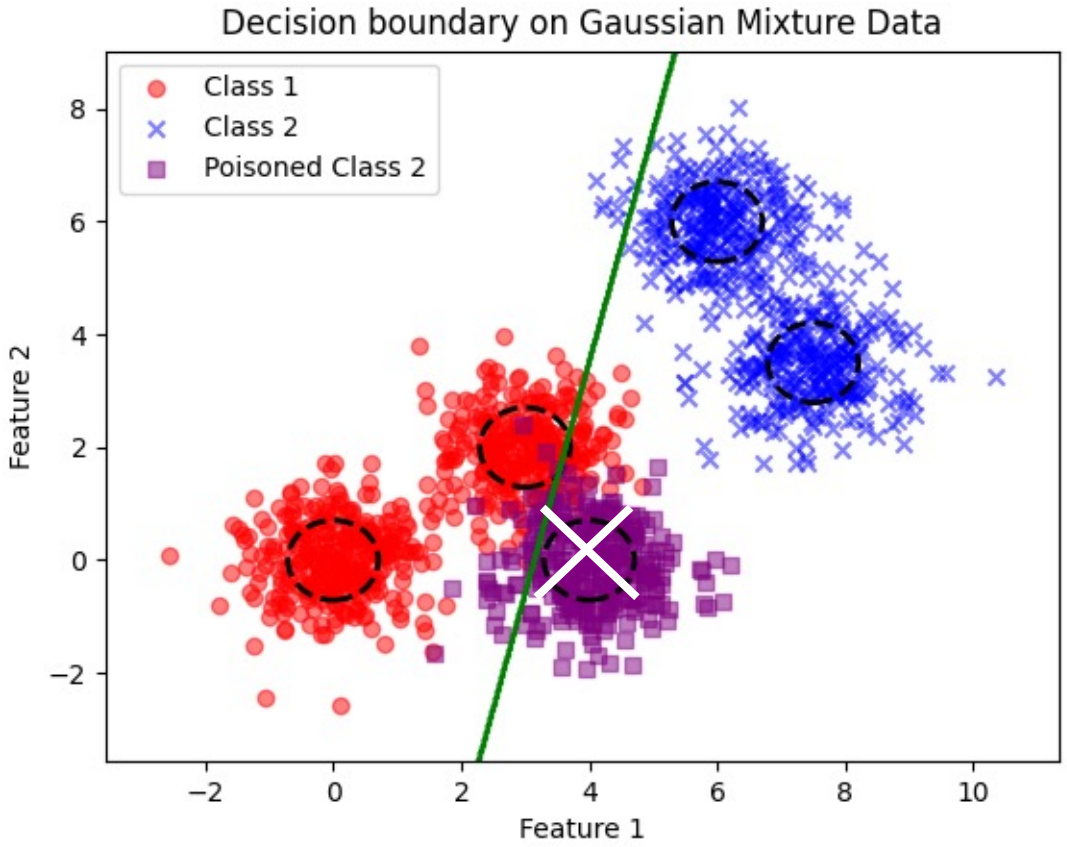
PennState

# Solution for Addressing Label-Flipping Attacks

Q: How to identify the **most likely poisoned** group?

Observation 2:

No need to keep the density function for the poisoned group.



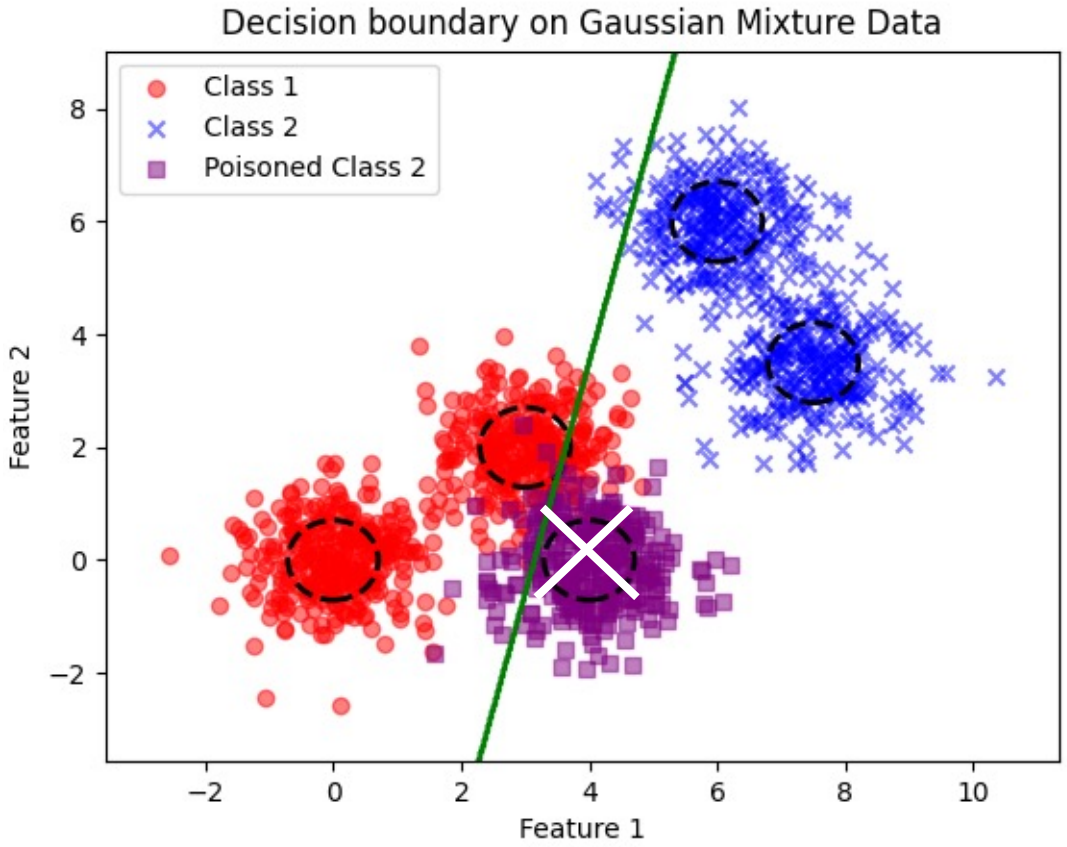
# Solution for Addressing Label-Flipping Attacks

Q: How to identify the **most likely poisoned** group?

Observation 2:

No need to keep the density function for the poisoned group.

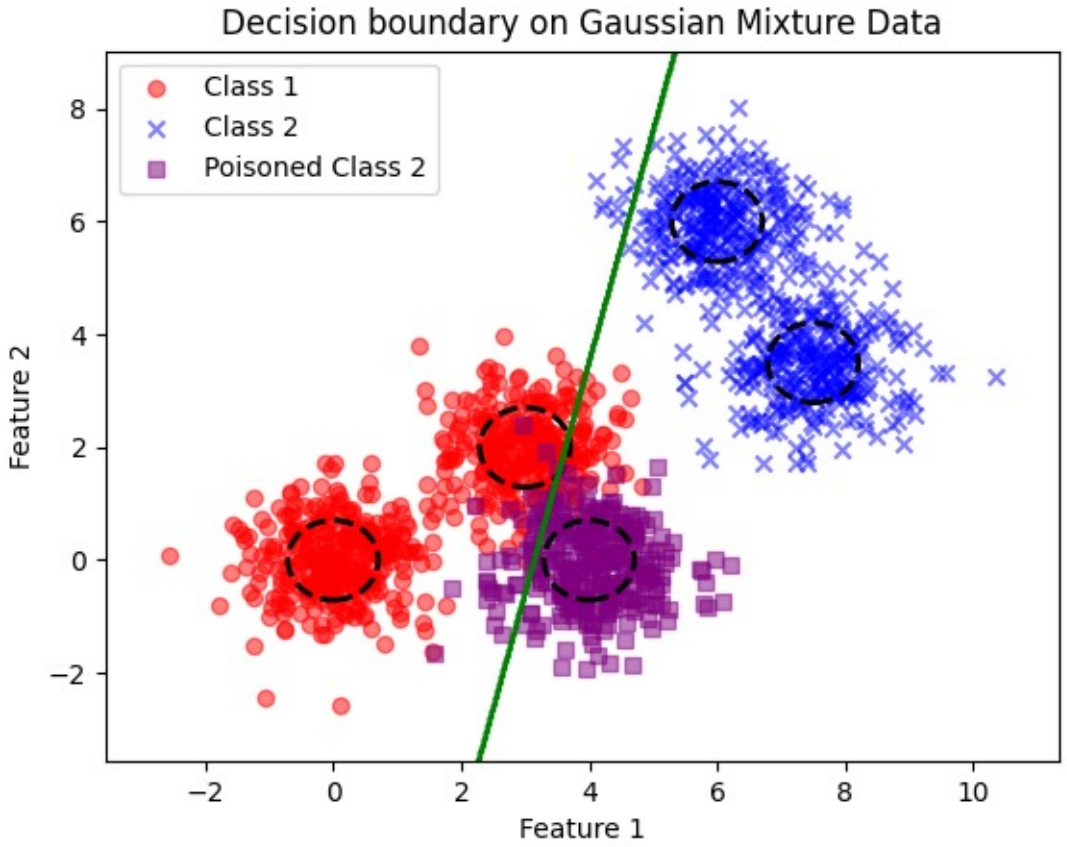
Removing the poisoned density function would **decrease model complexity**.



# Solution for Addressing Label-Flipping Attacks

Q: How to identify the **most likely poisoned** group?

A: Jointly optimize **data likelihood** and **model complexity**.



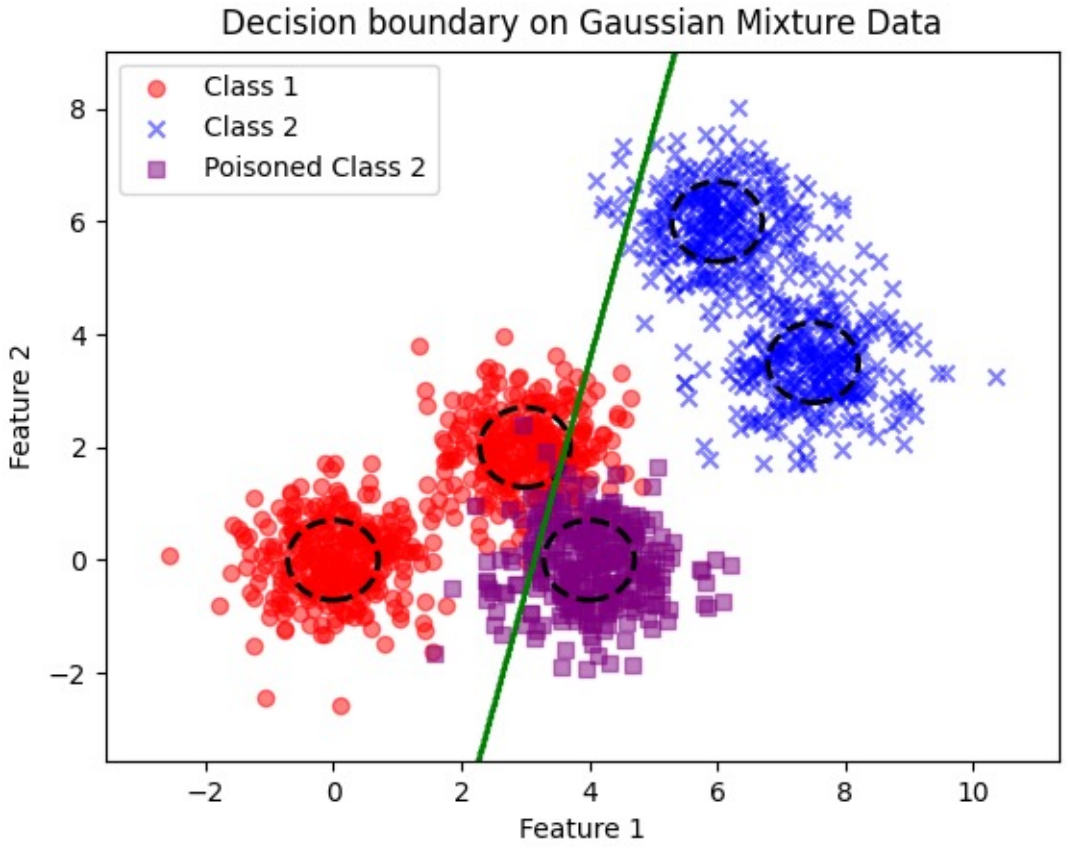
# Solution for Addressing Label-Flipping Attacks

Q: How to identify the **most likely poisoned** group?

A: Jointly optimize **data likelihood** and **model complexity**.

$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$

Aligns with minimizing Bayesian Information Criterion (**BIC**)



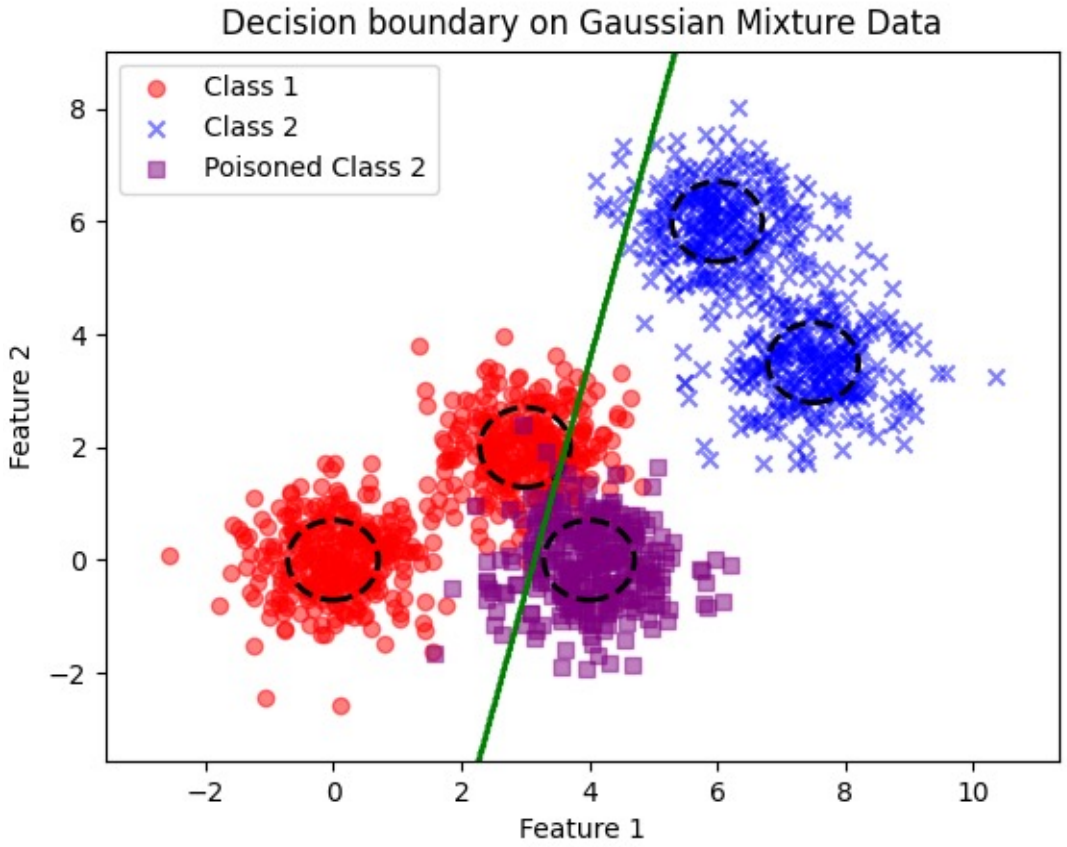



PennState

# Solution for Addressing Label-Flipping Attacks

Q: How to identify the **most likely poisoned** group?

A: **Minimize** Bayesian Information Criterion (**BIC**).






PennState

# Bayesian Information Criterion (BIC)

The Bayesian Information Criterion (BIC) function:

$$\mathrm{BIC} = \underbrace{|\theta| k}_{\text{Model Complexity}} - \underbrace{L(\mathcal{D}; \theta)}_{\text{Data Likelihood}}$$

$\theta$ – Set of parameters specifying **density functions**.
$k$ – Cost for a single parameter.
$\mathcal{D}$ – The dataset.

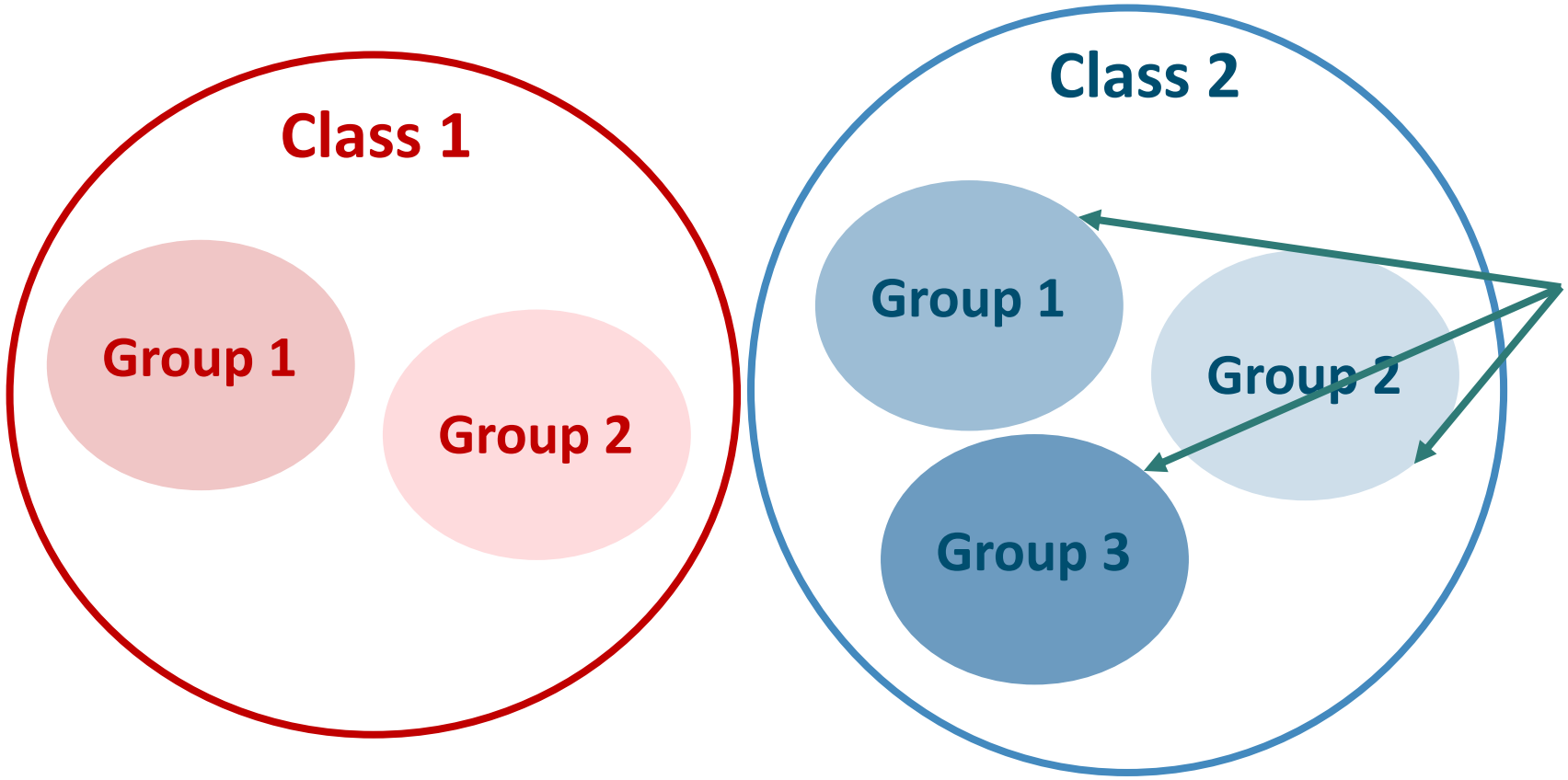
# Method Overview

Key idea:

1. **Isolation**: Isolate poisoned samples.
2. **Identification**: Identify the most likely poisoned group in each step consistent with **BIC** minimization.
3. **Sanitization**: Remove identified poisoned samples from the training set.

# Method Overview

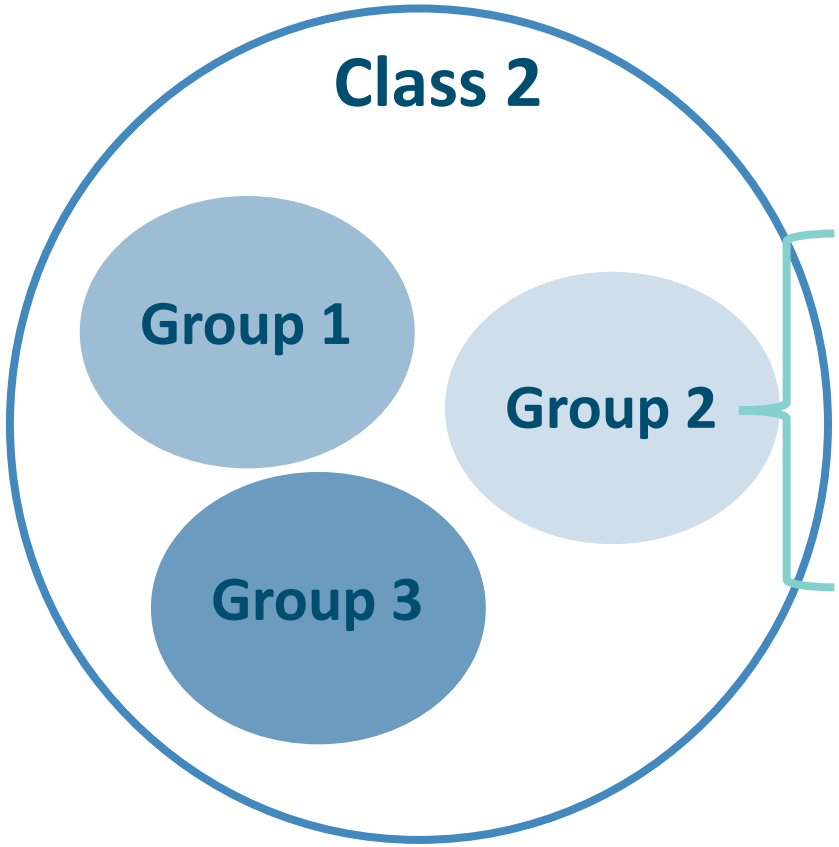
Isolation -> Identification -> Sanitization




Represent the distribution by a **mixture of density functions**.

# Method Overview

Isolation -> Identification -> Sanitization



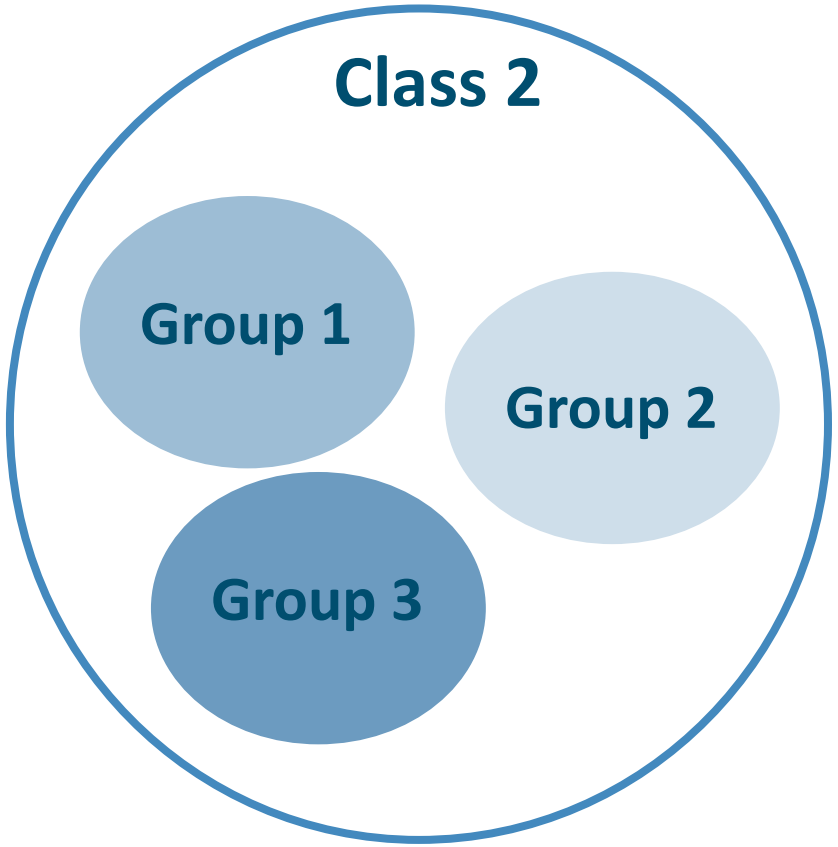

**Case 1**: Clean.

**Case2**: Poisoned, remove it.

**Case 3**: Poisoned, revise it.

# Method Overview

Isolation -> Identification -> Sanitization



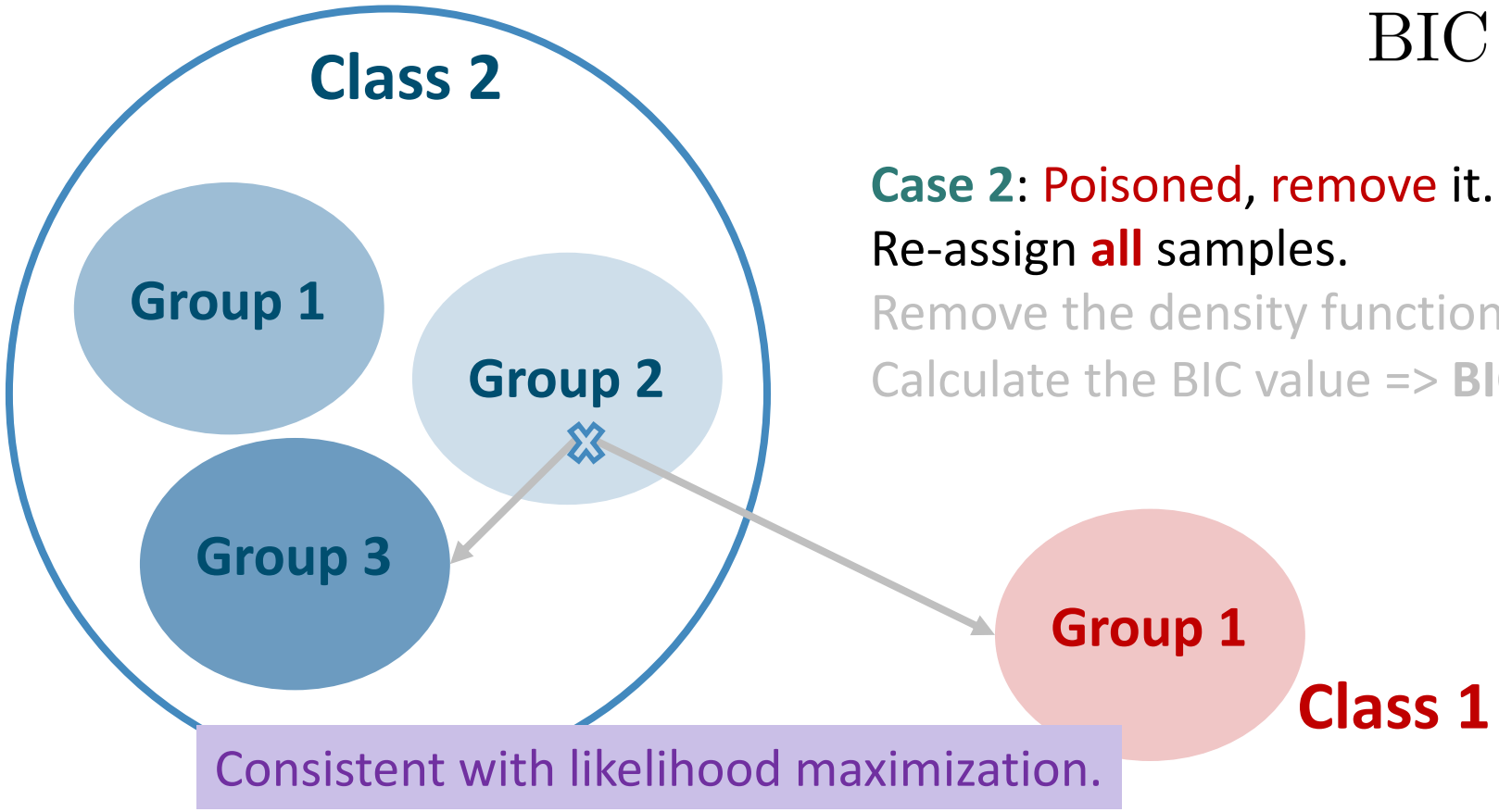

$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$

**Case 1**: Clean.
Do nothing.
Calculate the BIC value => **BIC1**.

# Method Overview

Isolation -> Identification -> Sanitization

$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$




**Case 2**: Poisoned, remove it.
Re-assign **all** samples.
Remove the density function.
Calculate the BIC value => **BIC2**.

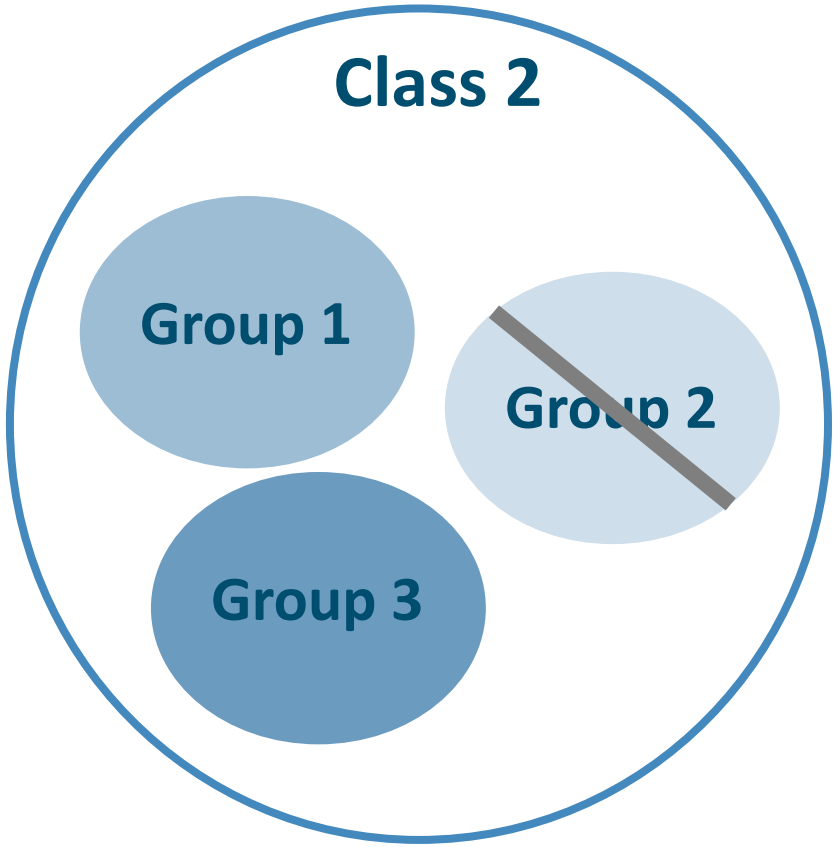
Consistent with likelihood maximization.



PennState

# Method Overview

Isolation -> Identification -> Sanitization




$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$

**Case 2**: Poisoned, remove it.

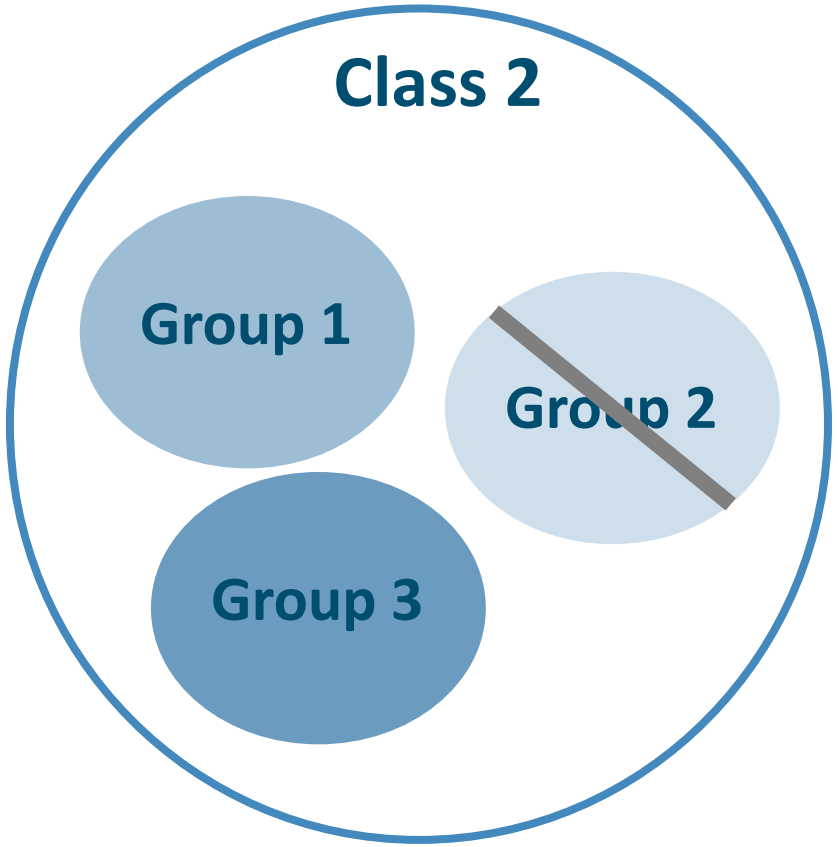
Re-assign **all** samples.

Remove the density function.

Calculate the BIC value => **BIC2**.

# Method Overview

Isolation -> Identification -> Sanitization




$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$

**Case 2**: Poisoned, remove it.

Re-assign **all** samples.

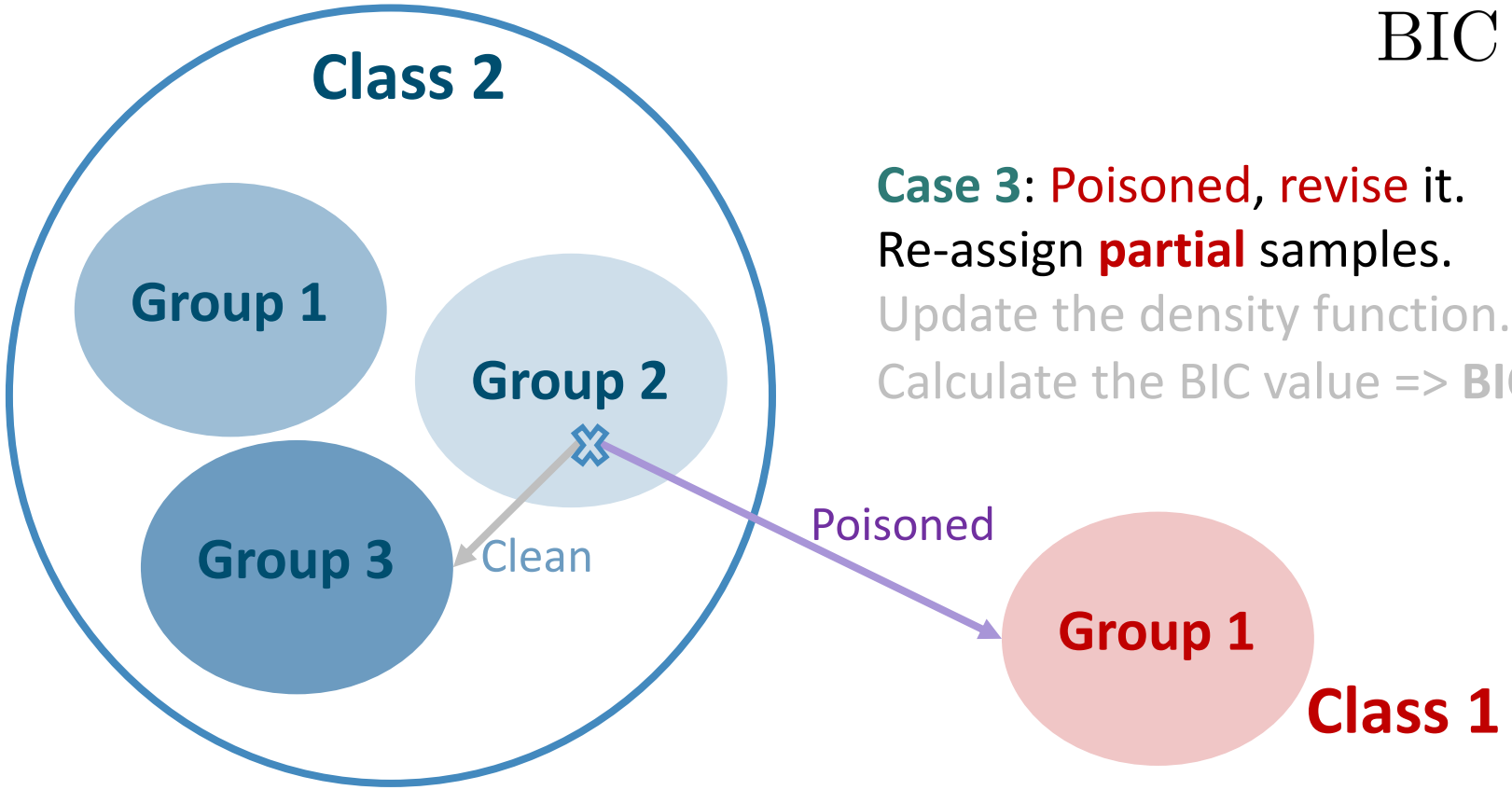
Remove the density function.

Calculate the BIC value => **BIC2**.

# Method Overview

Isolation -> Identification -> Sanitization

$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$




**Case 3**: Poisoned, revise it.

Re-assign **partial** samples.

Update the density function.

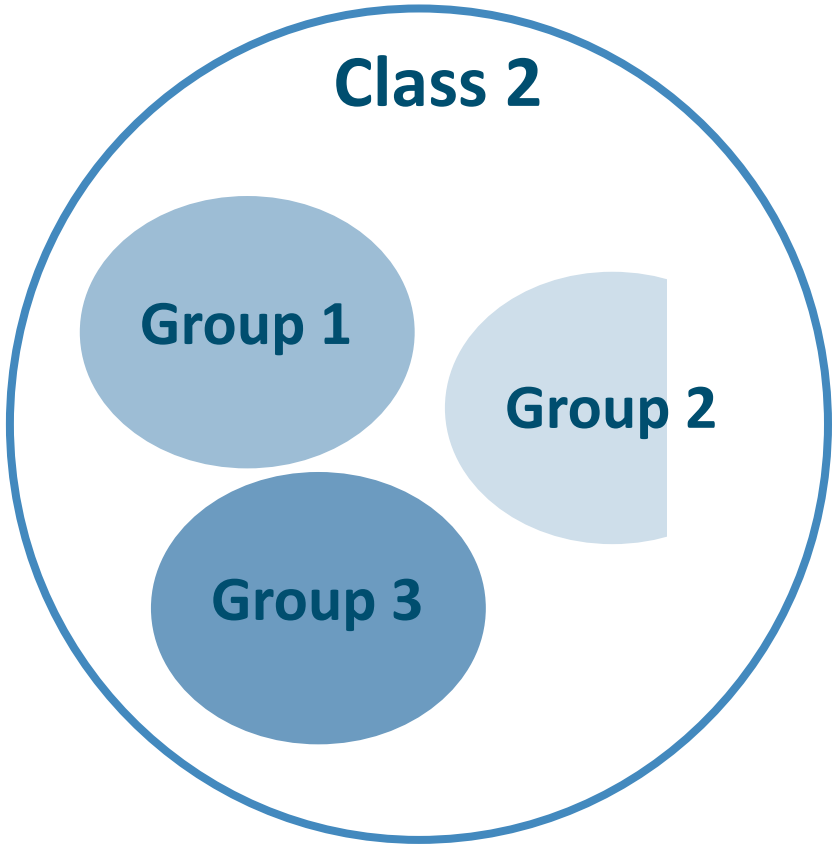
Calculate the BIC value => **BIC3**.



PennState

# Method Overview

Isolation -> Identification -> Sanitization




$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$

**Case 3**: Poisoned, revise it.

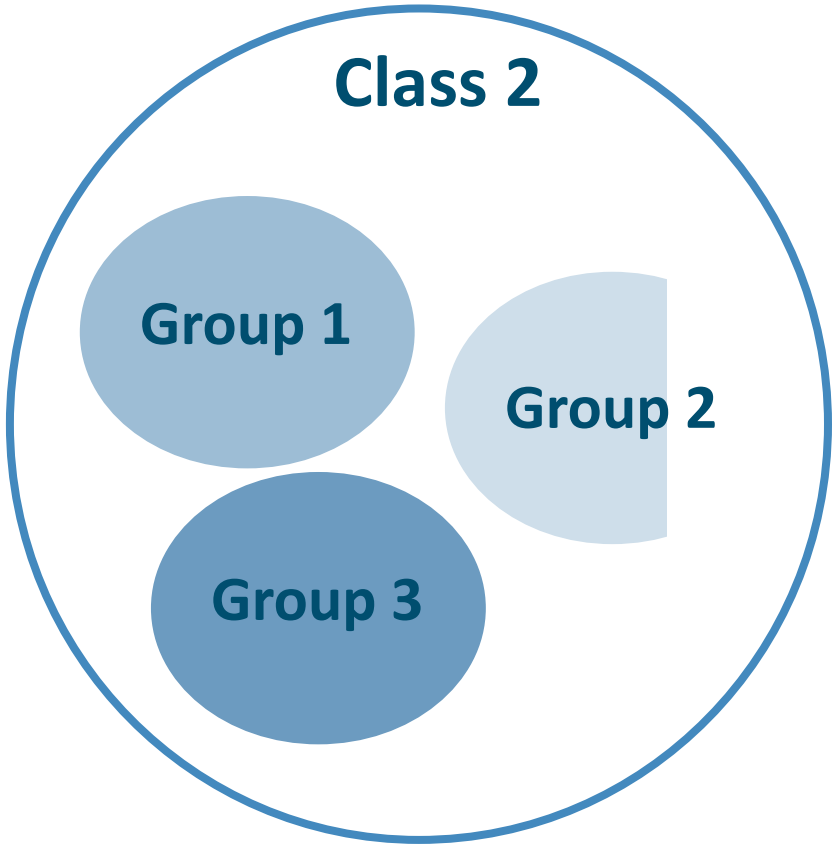
Re-assign **partial** samples.

Update the density function.

Calculate the BIC value => **BIC3**.

# Method Overview

Isolation -> Identification -> Sanitization




$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$

**Case 3**: Poisoned, revise it.

Re-assign **partial** samples.

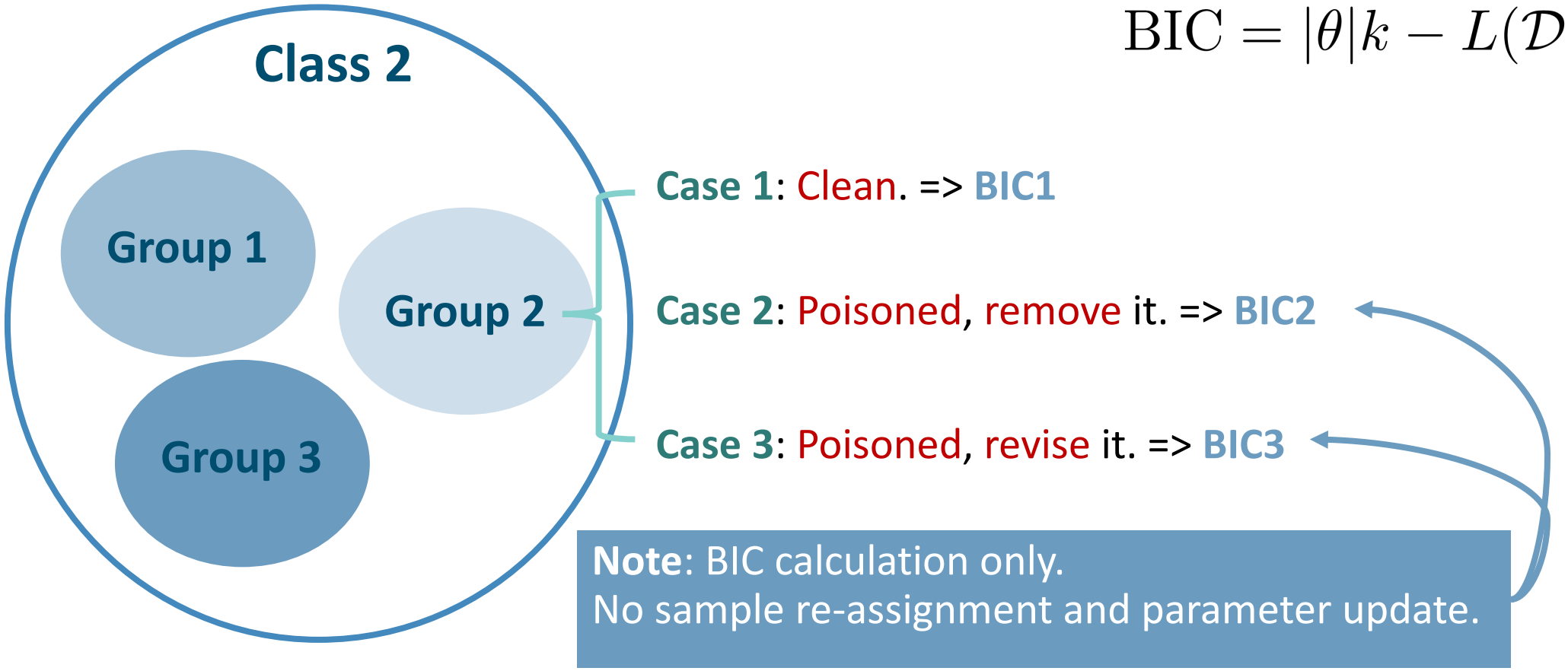
Update the density function.

Calculate the BIC value => **BIC3**.

# Method Overview

Isolation -> Identification -> Sanitization

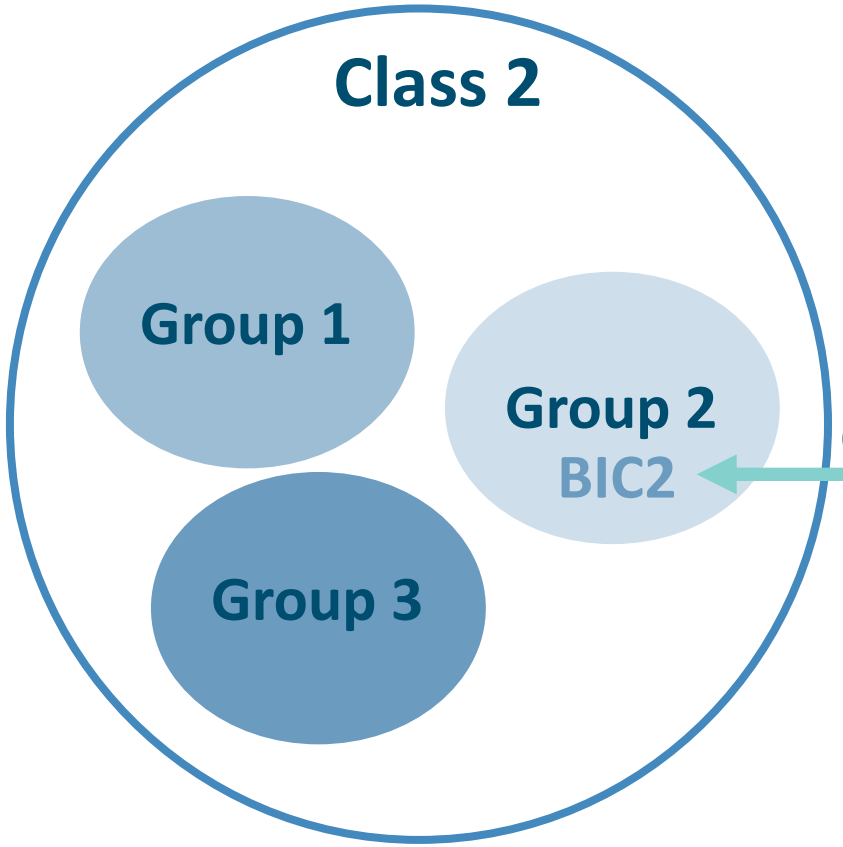
$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$



# Method Overview

Isolation -> Identification -> Sanitization

$$\mathrm{BIC} = |\theta| k - L(\mathcal{D}; \theta)$$




Choose the minimal BIC

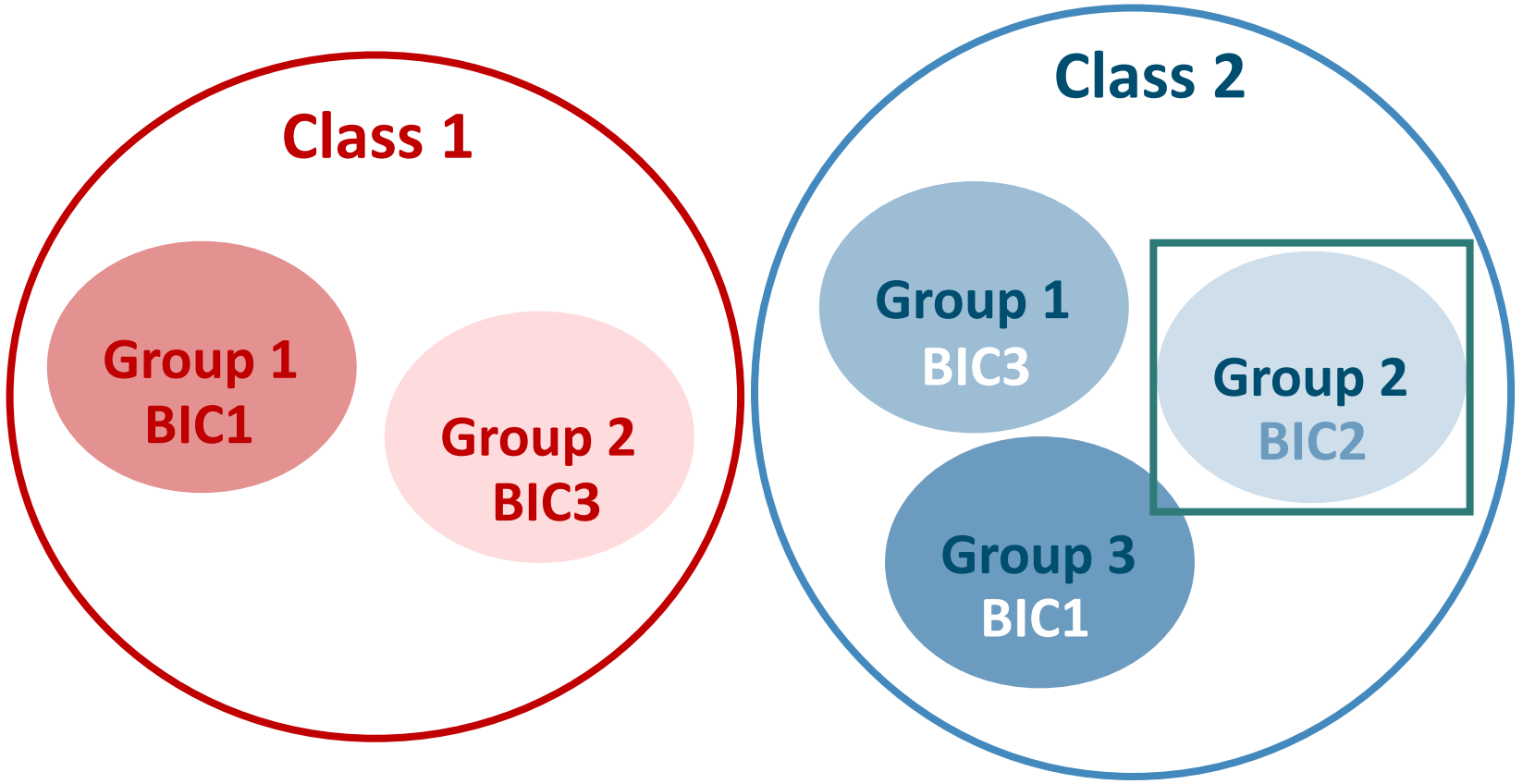
Case 1: Clean. => BIC1

Case2: Poisoned, remove it. => BIC2

Case 3: Poisoned, revise it. => BIC3

# Method Overview

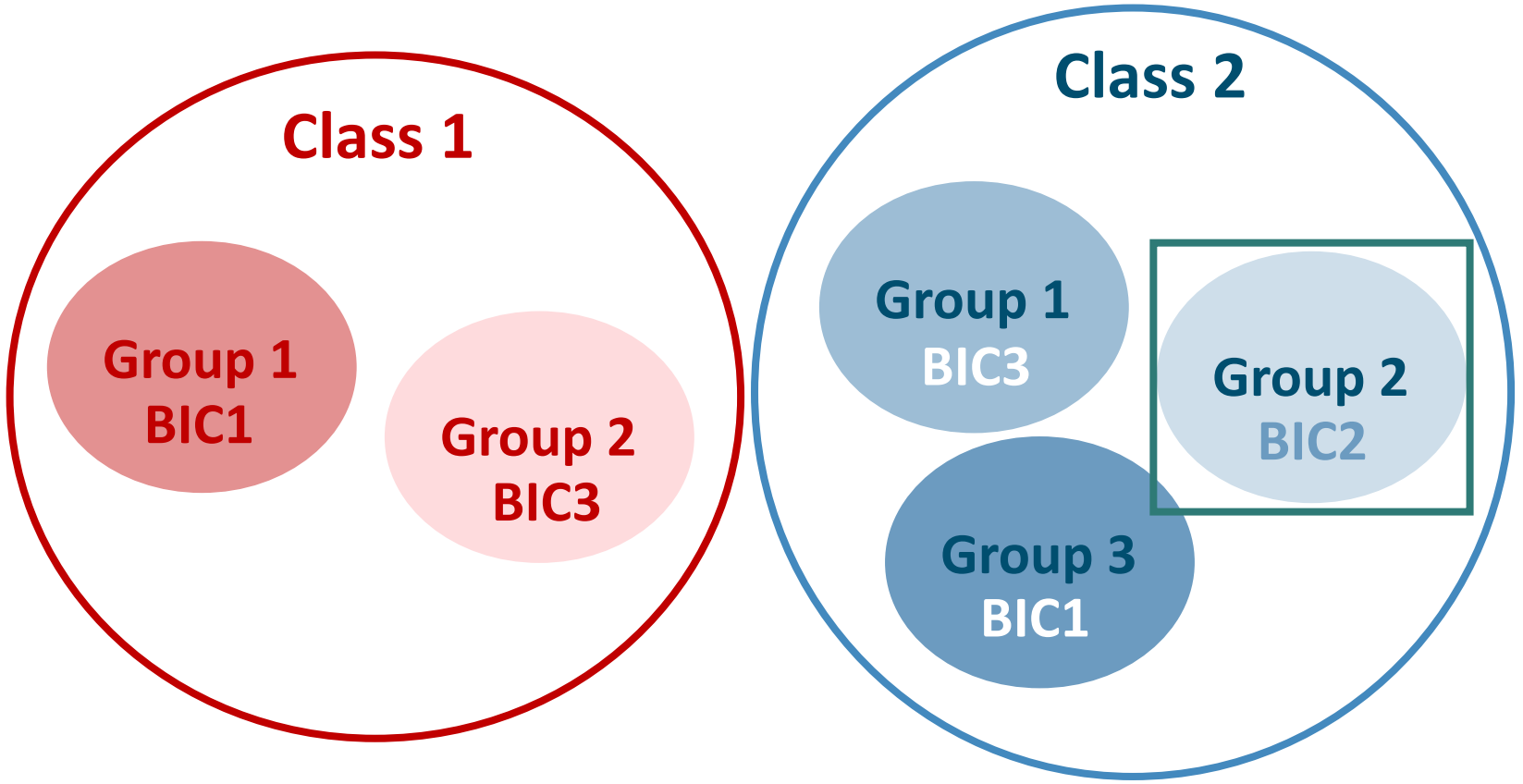
Isolation -> Identification -> Sanitization




The group with the **smallest BIC** is the most likely **poisoned** one.

# Method Overview

Isolation -> Identification -> Sanitization



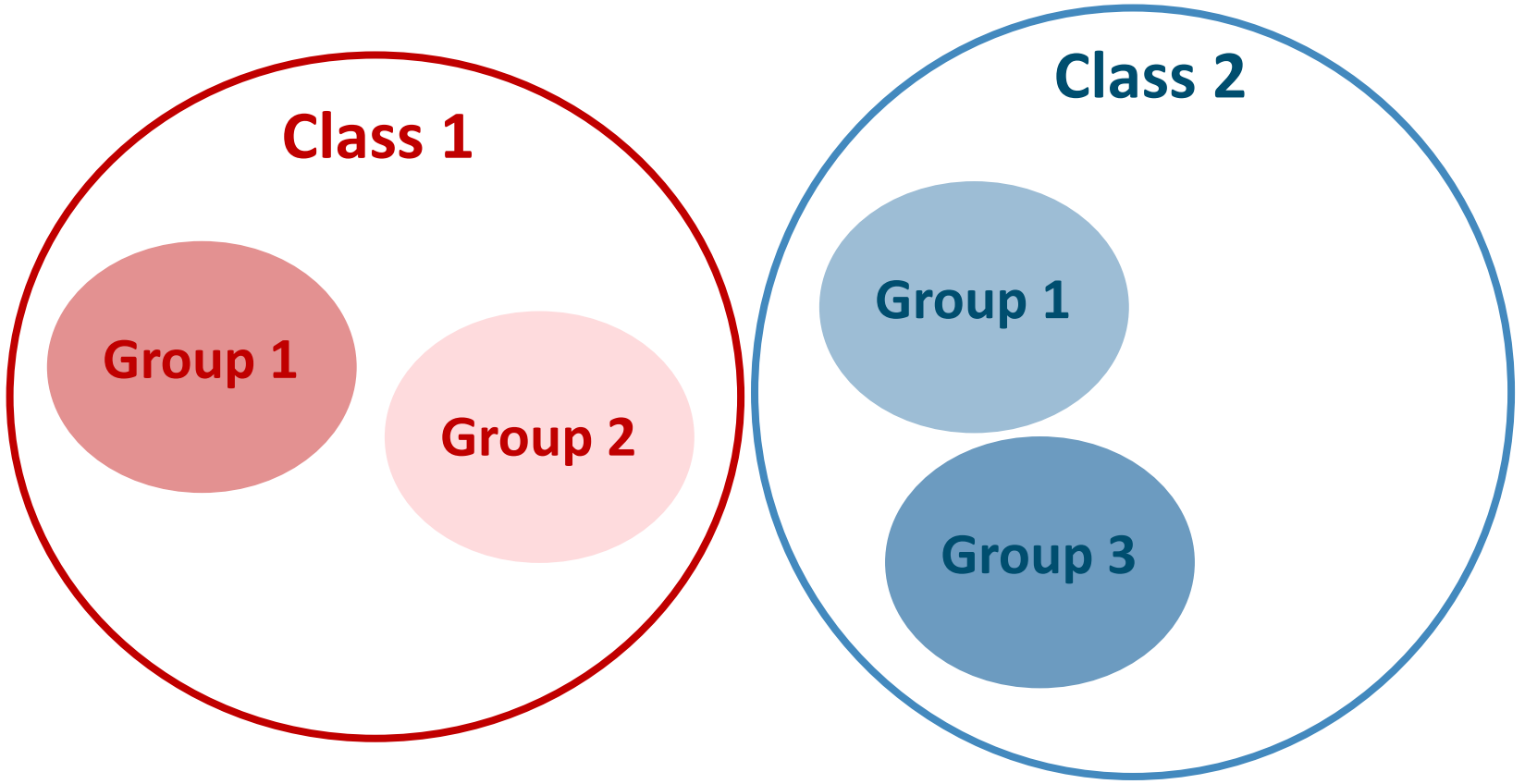

According to case2:

Re-assign samples.

Remove density function.

# Method Overview

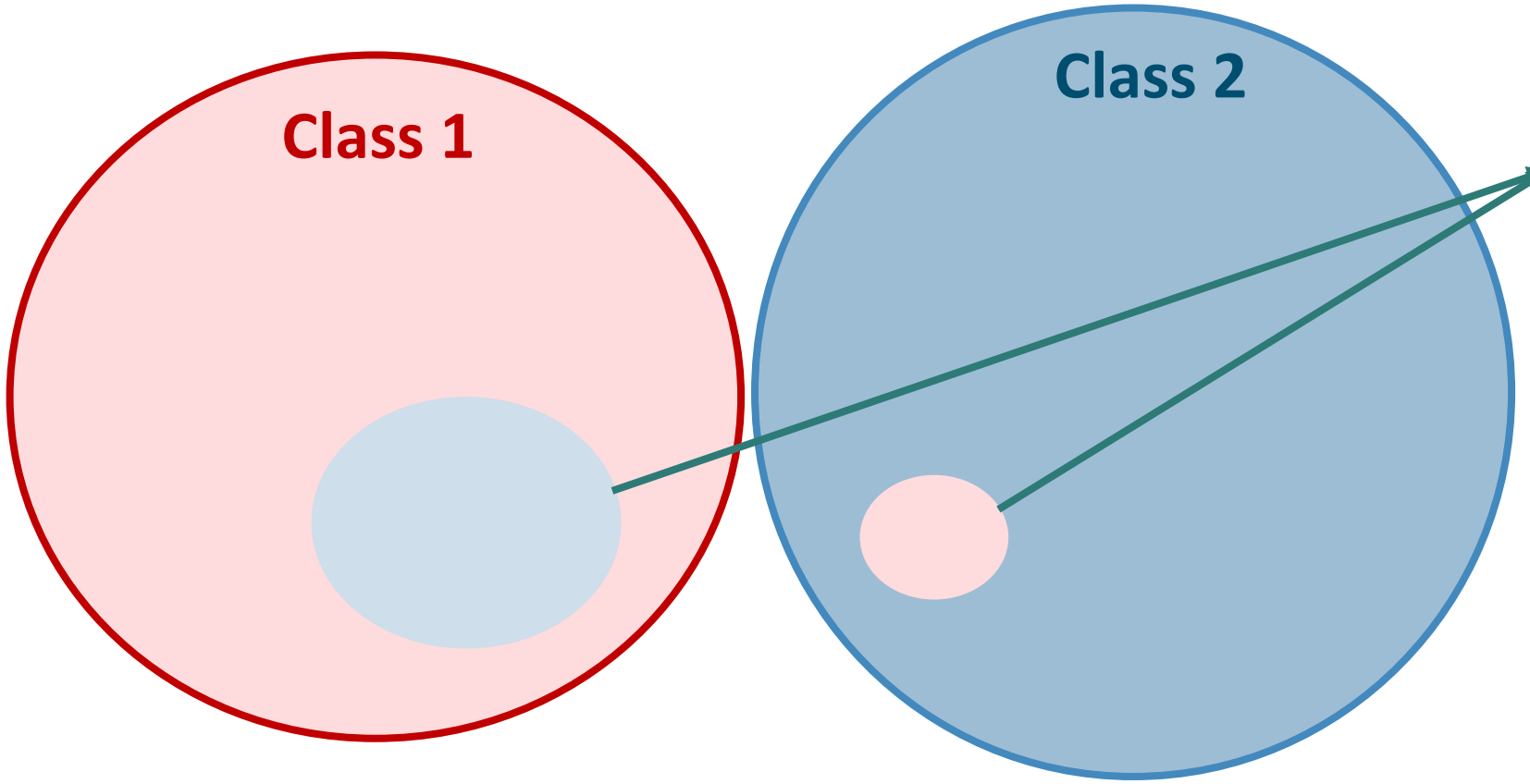
Isolation -> Identification -> Sanitization




Repeat optimizing until there's no further changes in BIC.

# Method Overview

Isolation -> Identification -> Sanitization




Remove:

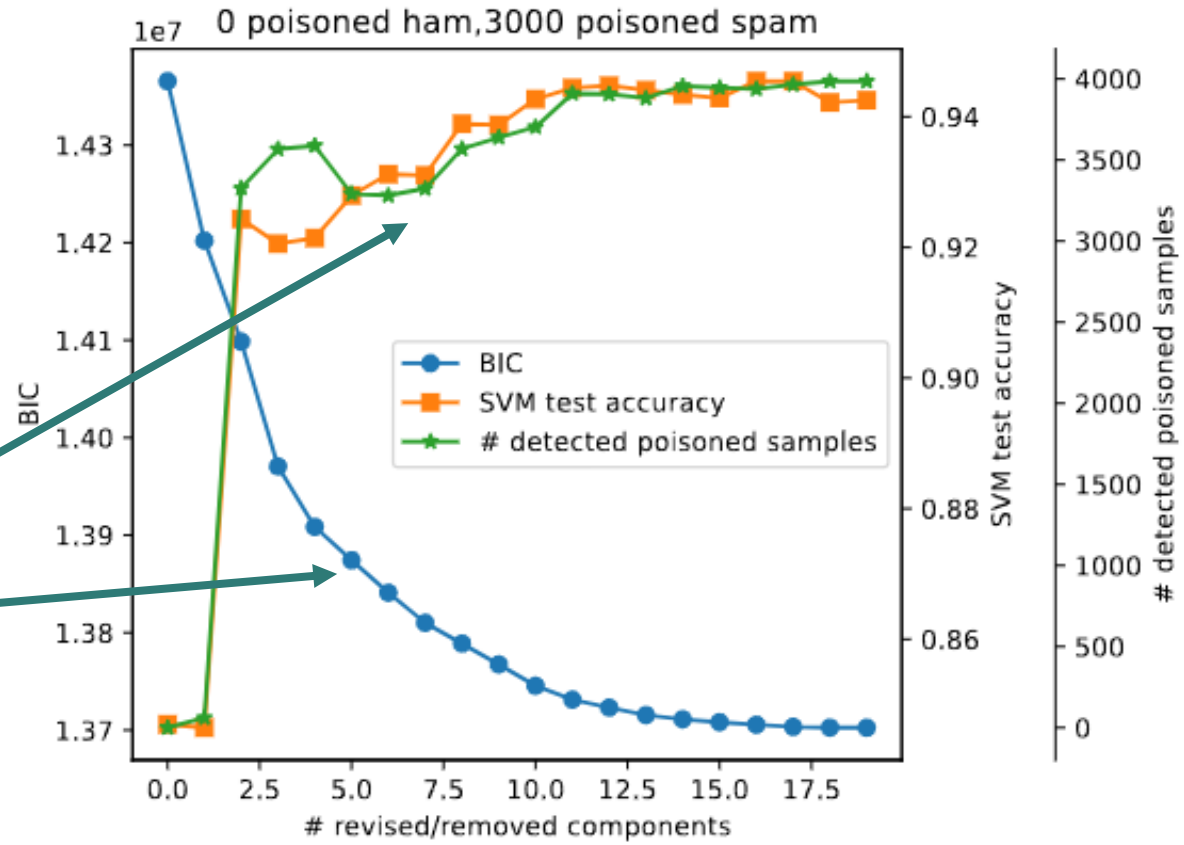
Samples re-assigned to other classes are deemed poisoned.

# Method Effectiveness

Datasets: TREC05 (**Binary** classification).

Victim models: SVM, LSTM, ...

**Test ACC increases as BIC decreases.**




BIC cost, SVM ACC, and the number of detected poisoned samples vs. the number of visited components.

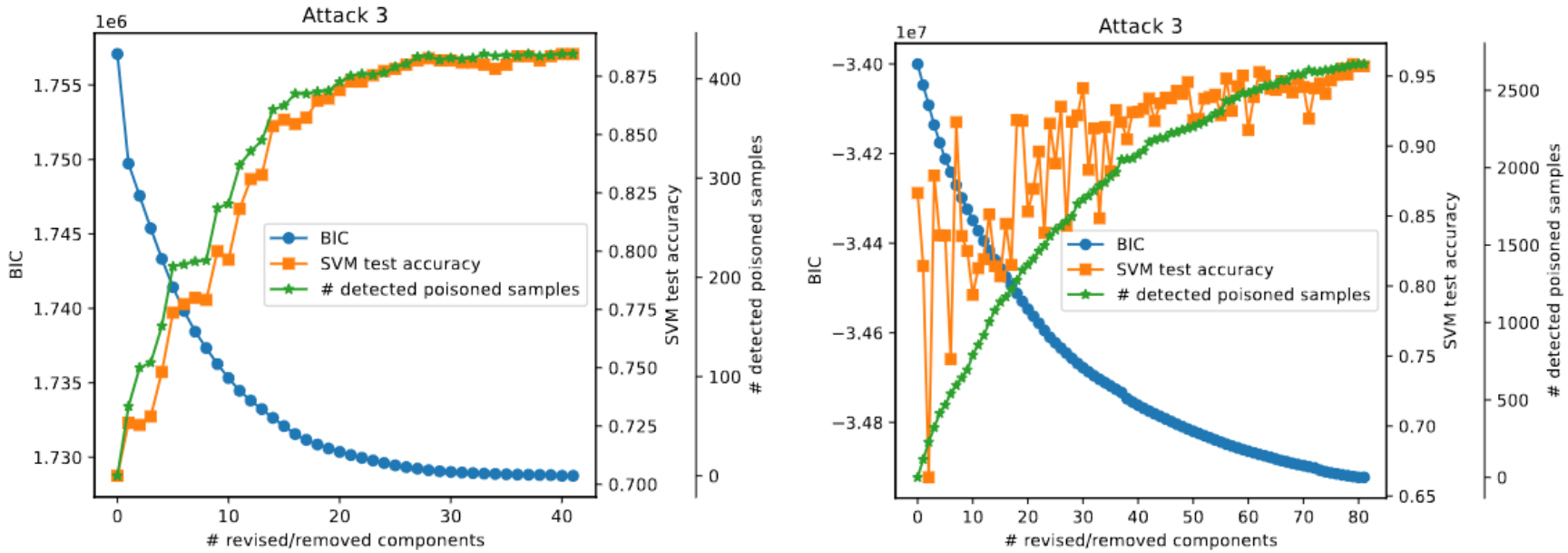


PennState

# Method Effectiveness

Datasets: 20NG, MNIST, CIFAR10, STL10 (Multi-class classification).

Victim models: SVM, logistic regression, LSTM, ResNet-18.




BIC cost, SVM ACC, and the number of detected poisoned samples versus the number of visited components under attacks (a) attack 3 against 20NG; (b) attack 3 against MNIST.



PennState

# Method Effectiveness

**High t-statistics** show our **significant improvement** over other detection methods.

| Attack | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| | | | **20NG** | | | |
| vs KNN-D[1] | 14.17 | 5.69 | 9.27 | 9.27 | 20.04 | 9.05 |
| vs SVD-D[2] | 2.12 | 24.24 | 7.58 | 16.57 | 19.95 | 17.85 |
| vs GS-D[3] | 3.81 | 4.98 | 10.03 | 17.65 | 16.71 | 12.50 |
| | | | **CIFAR10** | | | |
| vs DPA[4] | 7.18 | 12.65 | 11.89 | 7.57 | 7.74 | 22.22 |
| vs FA[5] | 6.09 | 8.20 | 15.31 | 15.39 | 12.92 | 7.61 |

**T-statistics** comparing the performance of our method to other methods.


[1] Andrea Paudice, Luis Munoz-Gonzalez, and Emil C. Lupu. Label Sanitization Against Label Flipping Poisoning Attacks. ECML PKDD Workshops. 2018.
[2] Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A Robust Meta-Algorithm for Stochastic Optimization. ICML. 2019.
[3] Sanghyun Hong, Varun Chandrasekaran, Yigitcan Kaya, Tudor Dumitras, and Nicolas Papernot. On the Effectiveness of Mitigating Data Poisoning Attacks with Gradient Shaping. 2020.
[4] Alexander Levine and Soheil Feizi. Deep Partition Aggregation: Provable Defenses against General Poisoning Attacks. ICLR. 2021.
[5] Wenxiao Wang, Alexander Levine, and Soheil Feizi. Improved Certified Defenses against Data Poisoning with (Deterministic) Finite Aggregation. ICML. 2022.

# Conclusion

The proposed method:

- **Effective** – solve the practical challenging label-flipping attack.
- **Universal** – applicable to various model structures and datasets.
- **Unsupervised** – no hyper-parameter tuning.