# Trustworthy ML... for Systems Security

Lorenzo Cavallaro <l.cavallaro@ucl.ac.uk>
@lcavallaro — https://s2lab.cs.ucl.ac.uk

International Winter School on Microarchitectural Security 2022
FIAP, Paris

Dec 5, 2022

SYSTEMS SECURITY
RESEARCH LAB

# Machine Learning Revolution

Image Classification

Facial Recognition

Machine Translation

Speech Recognition

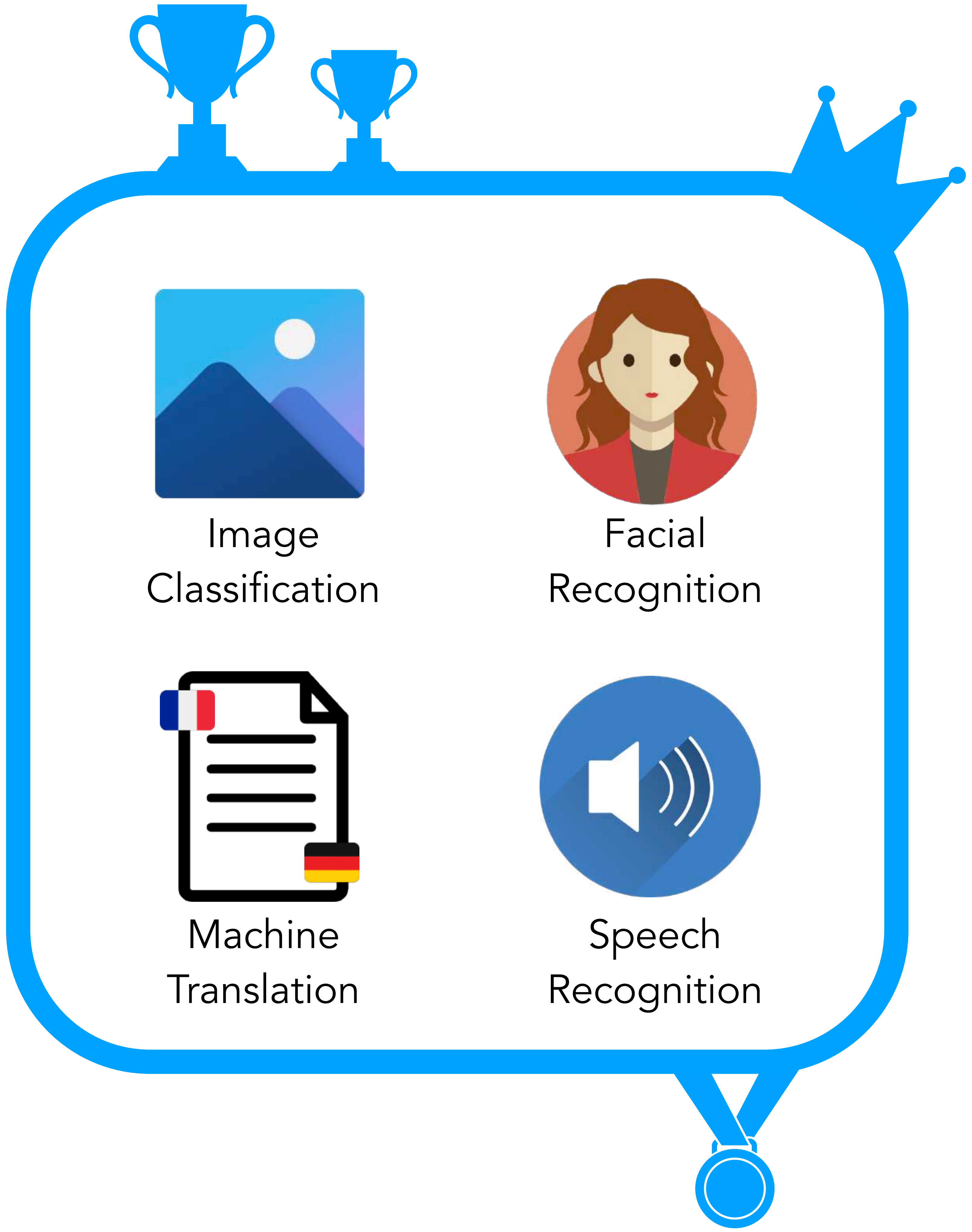
Android Malware

Malicious Javascript

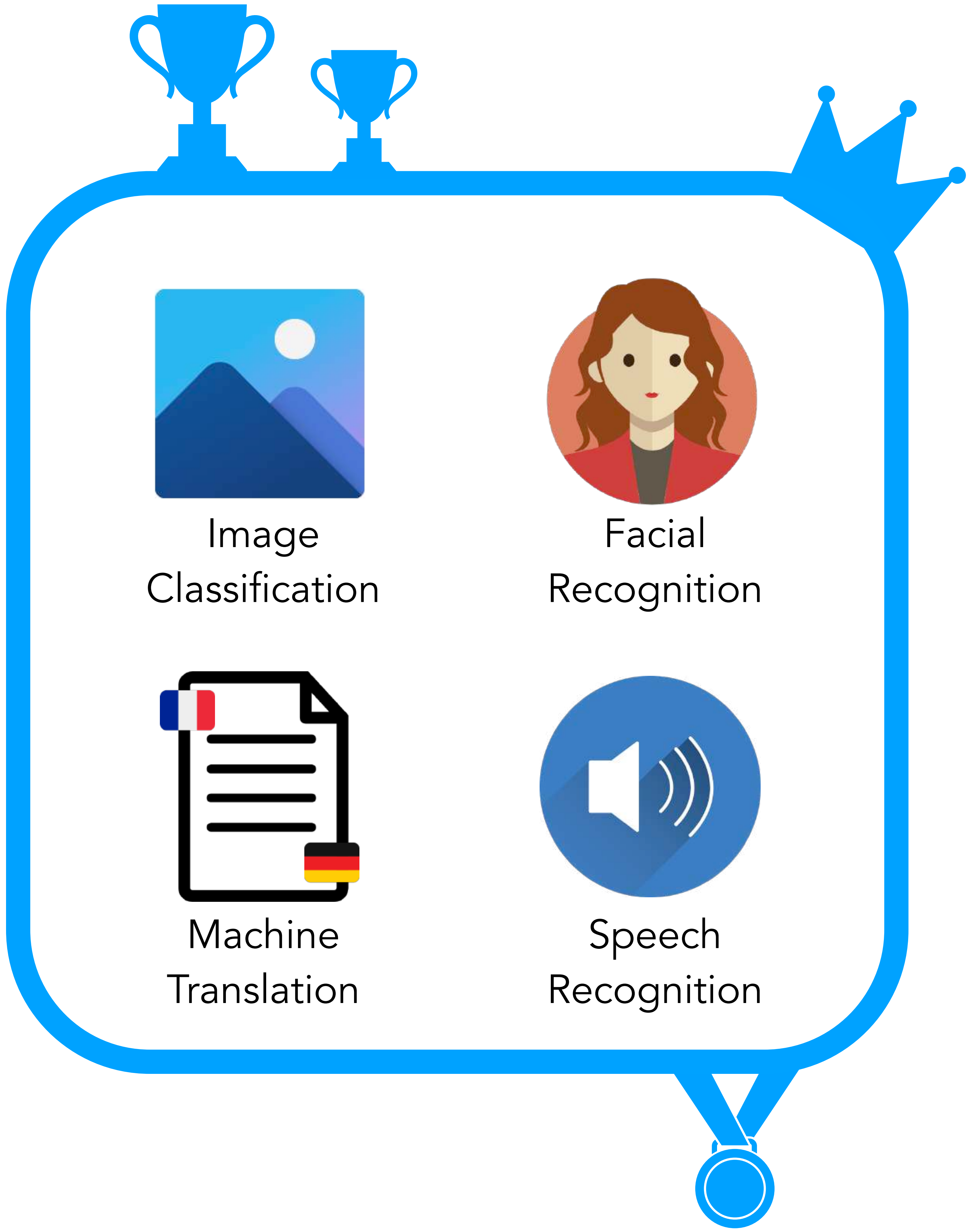
Windows Malware

PDF Malware

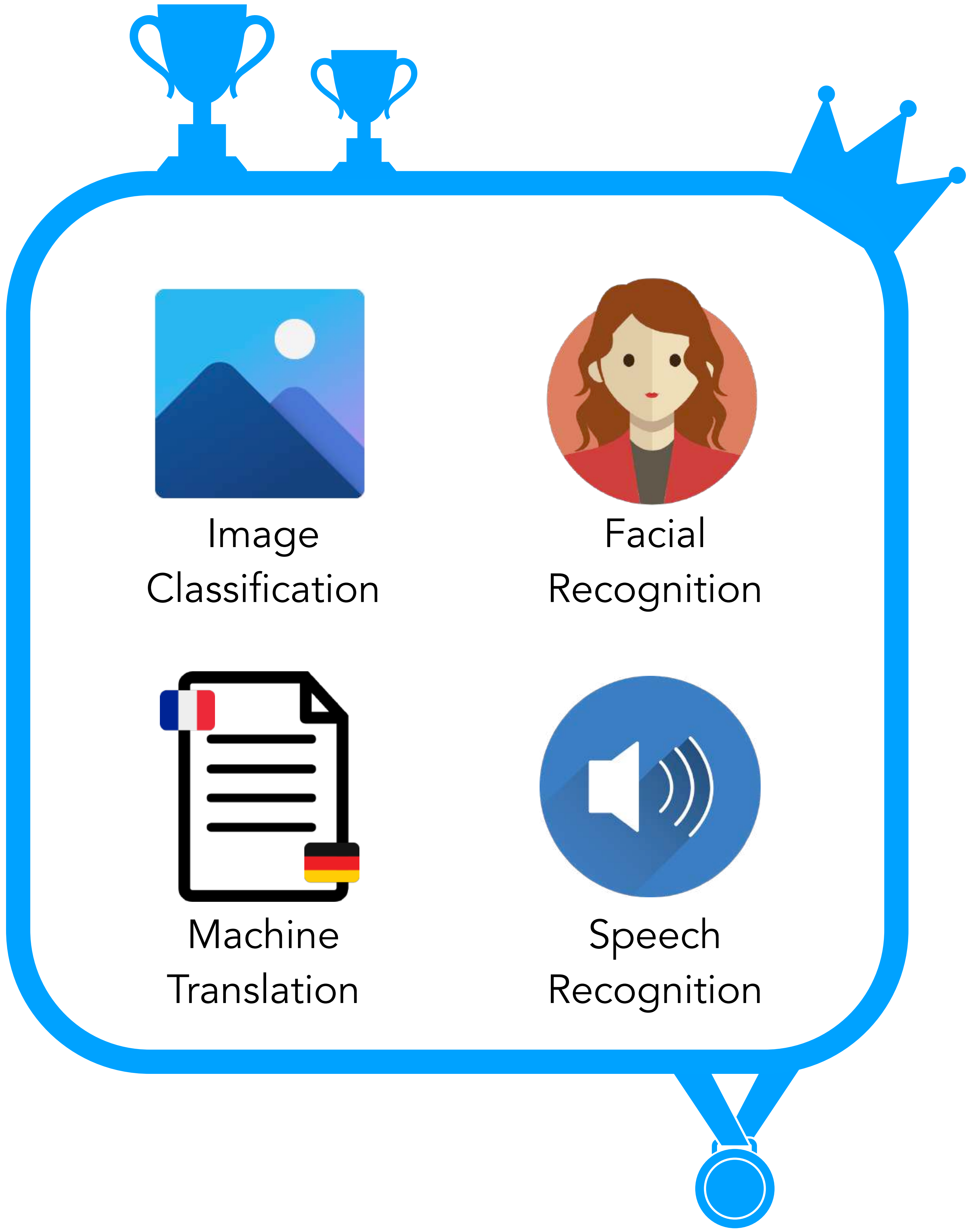
# Machine Learning Revolution




# Machine Learning Revolution



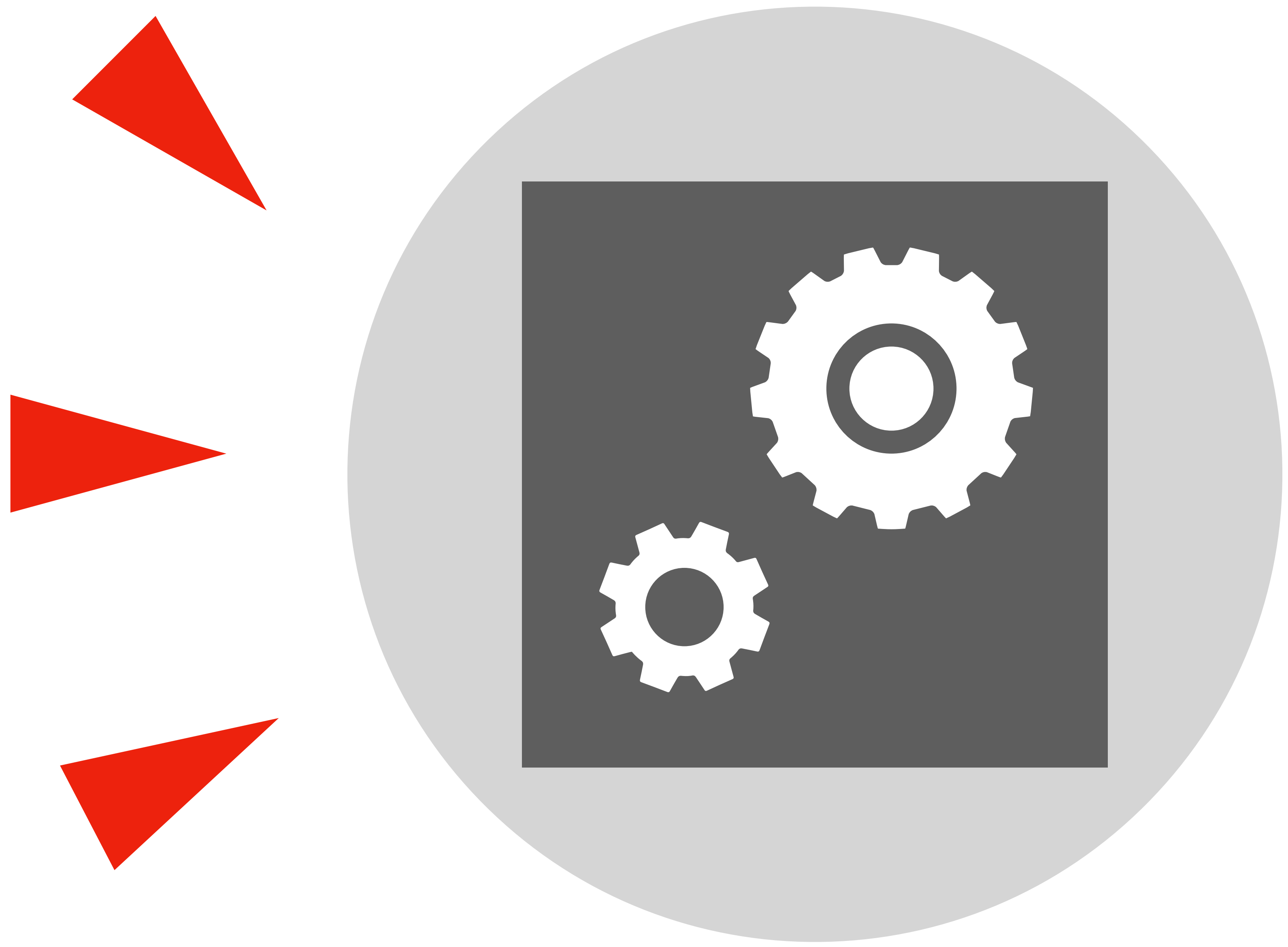

# Machine Learning Revolution





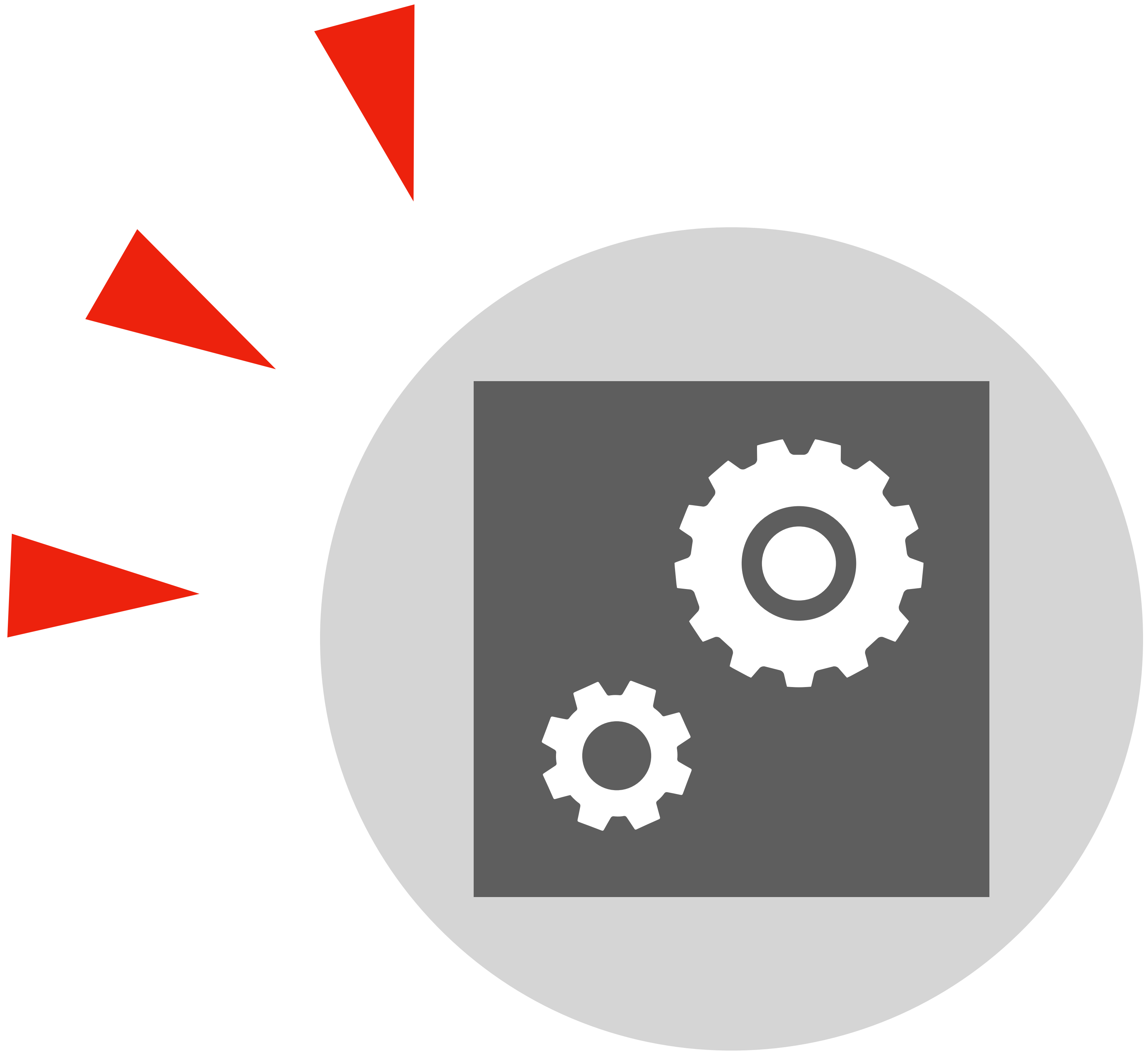
Maybe the conditions aren’t ready yet?

# Security *is* Adversarial



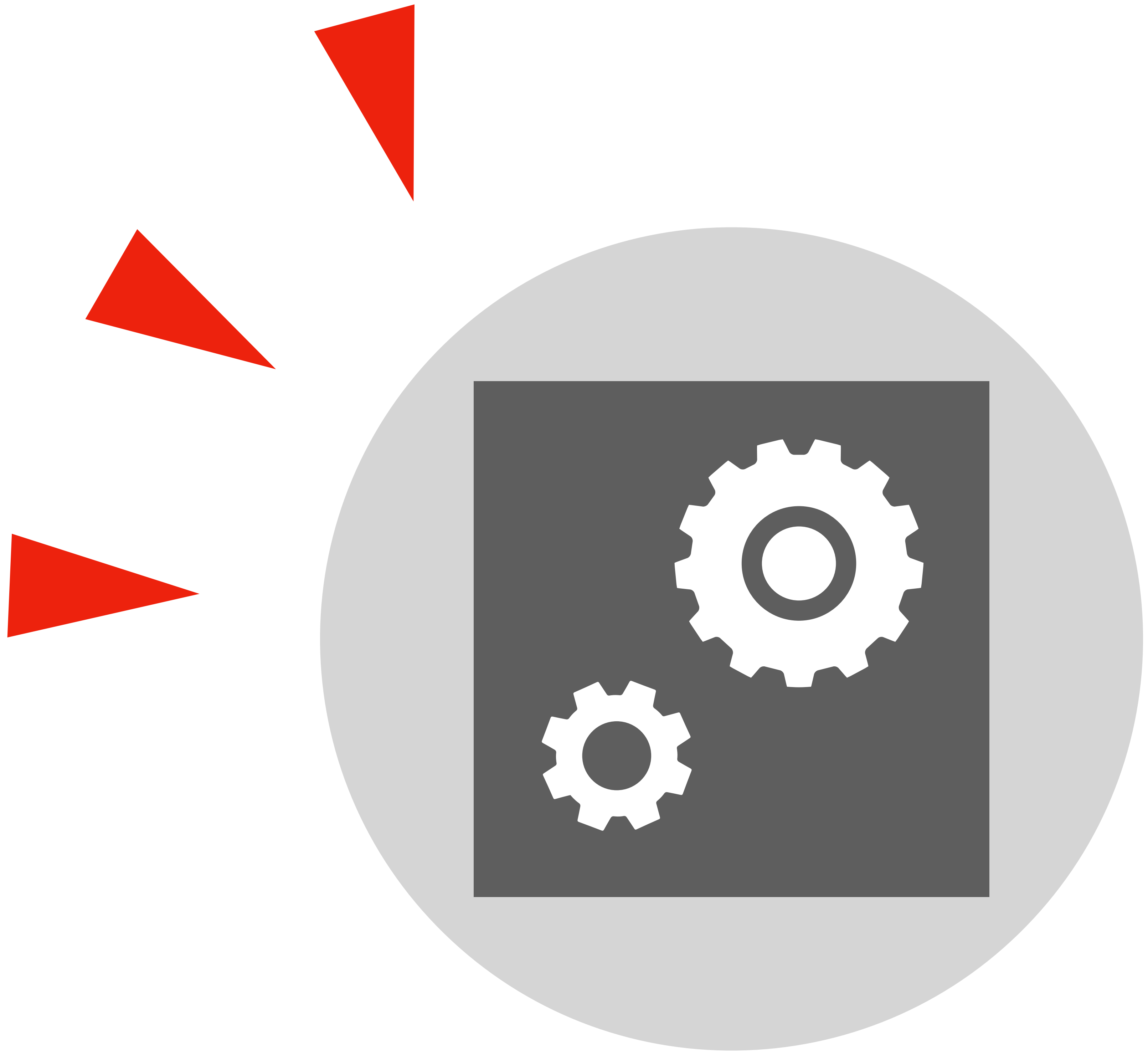
New detection systems trigger
an immediate response…

# Security *is* Adversarial

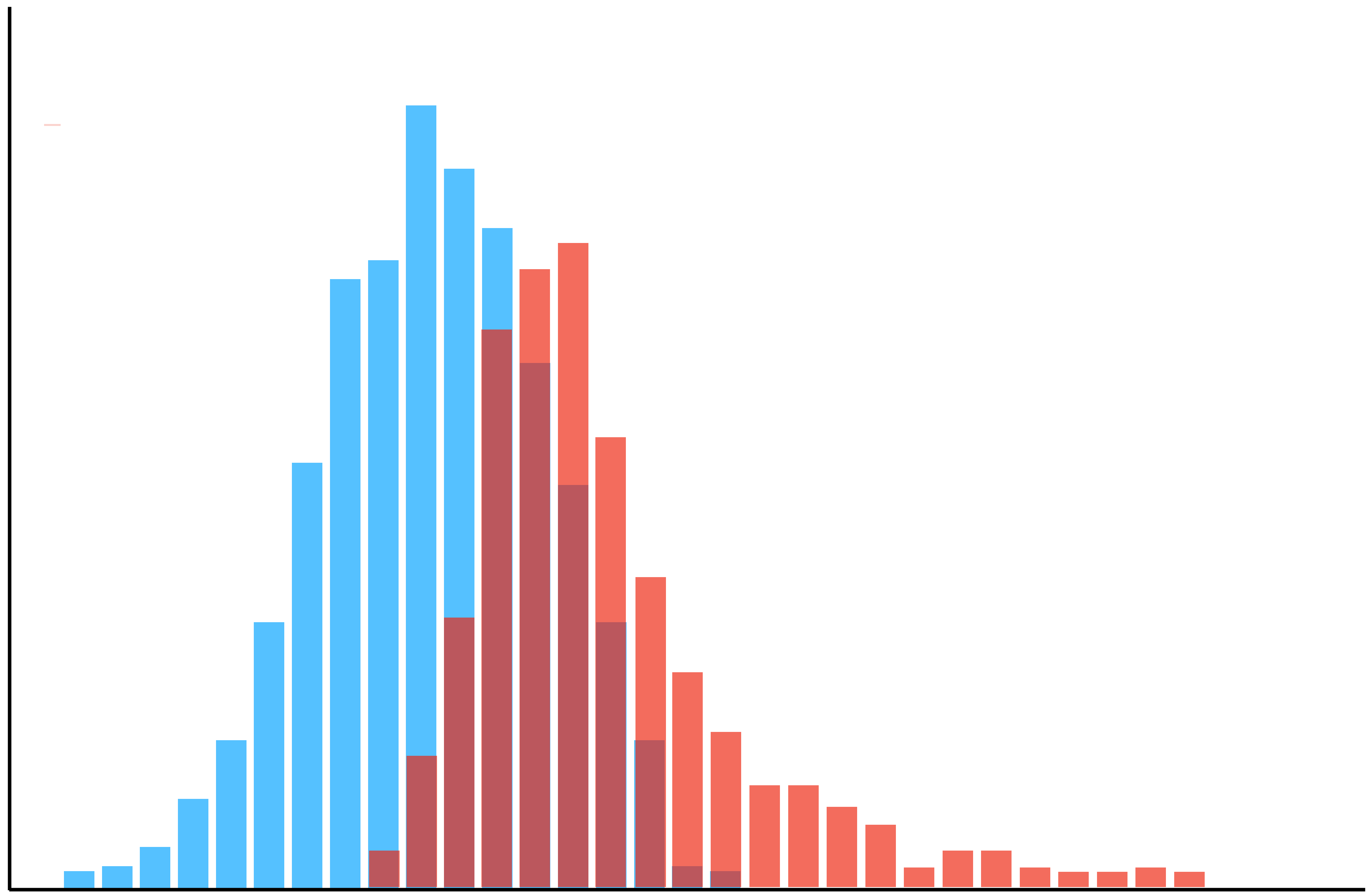


New detection systems trigger
an immediate response…

# Security *is* Adversarial

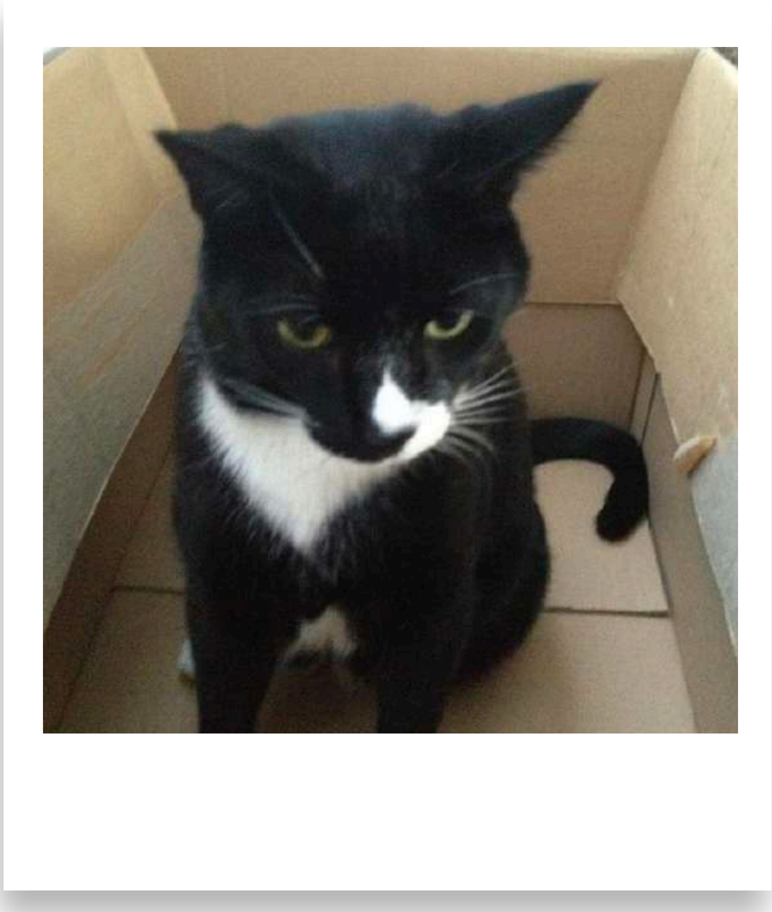
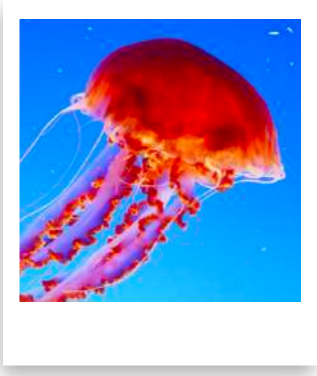
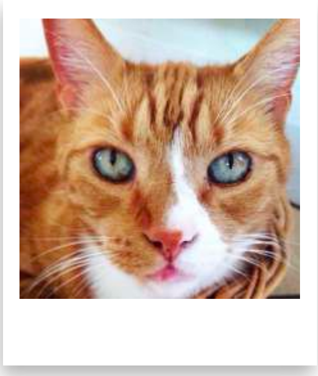
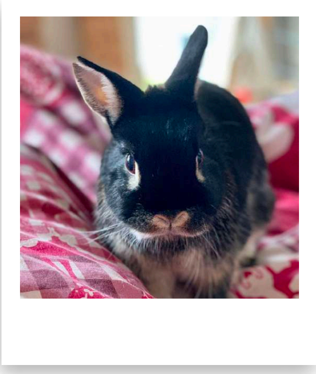
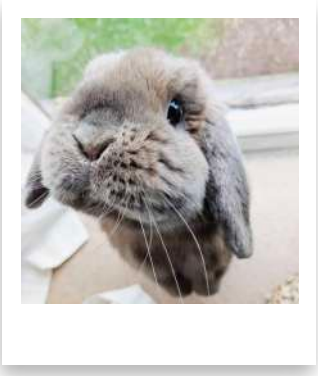
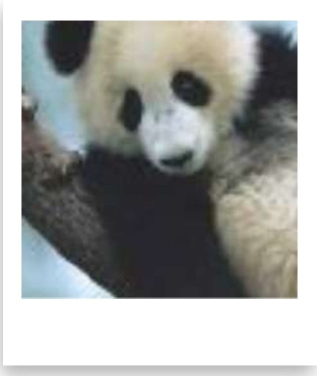


New detection systems trigger an immediate response…

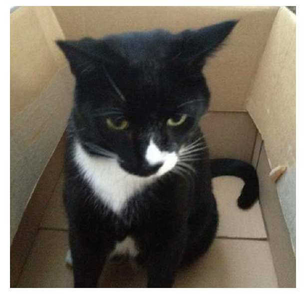


…which causes dataset shifts, often violating the i.i.d. assumption

# Security *is* Adversarial

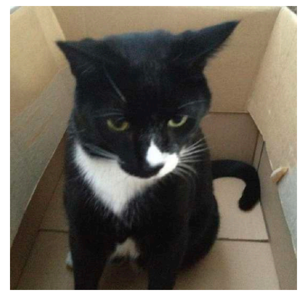

# Security *is* Adversarial





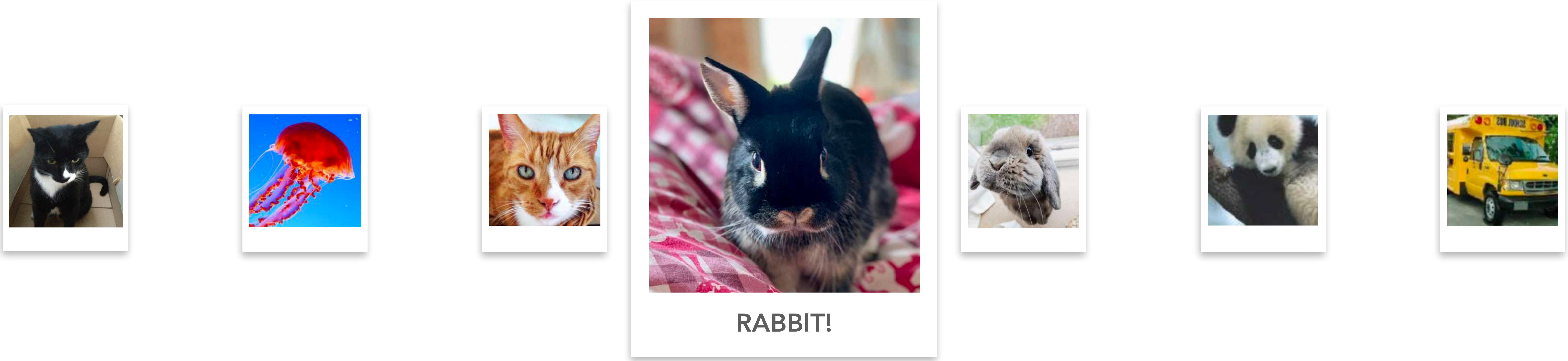
RABBIT!


# Security *is* Adversarial



RABBIT!

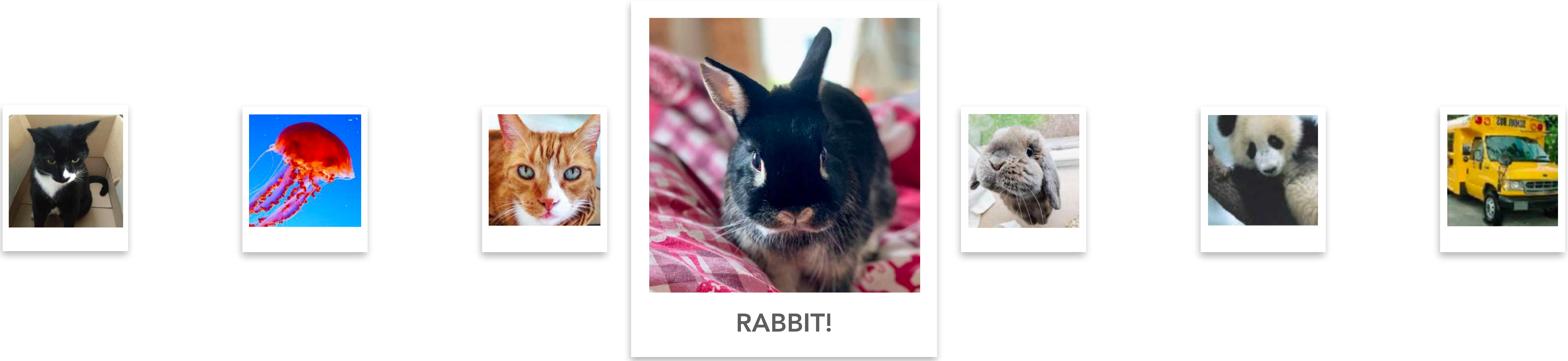
# Security *is* Adversarial

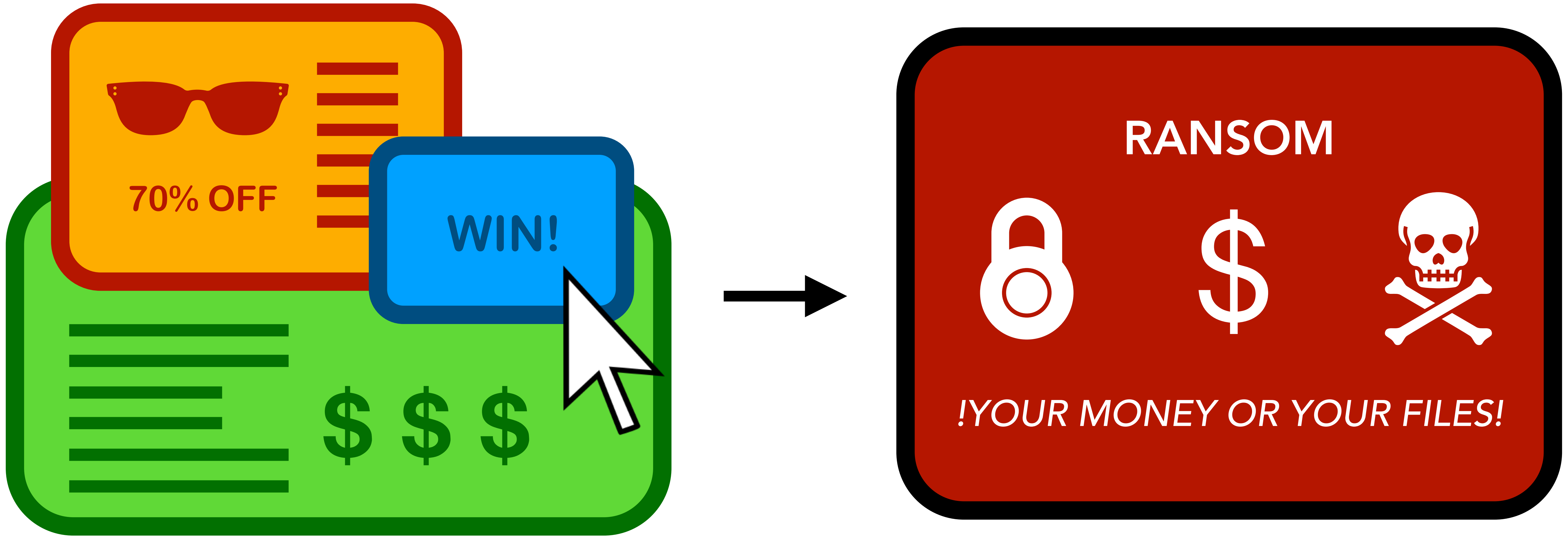




Training on ad fraud…

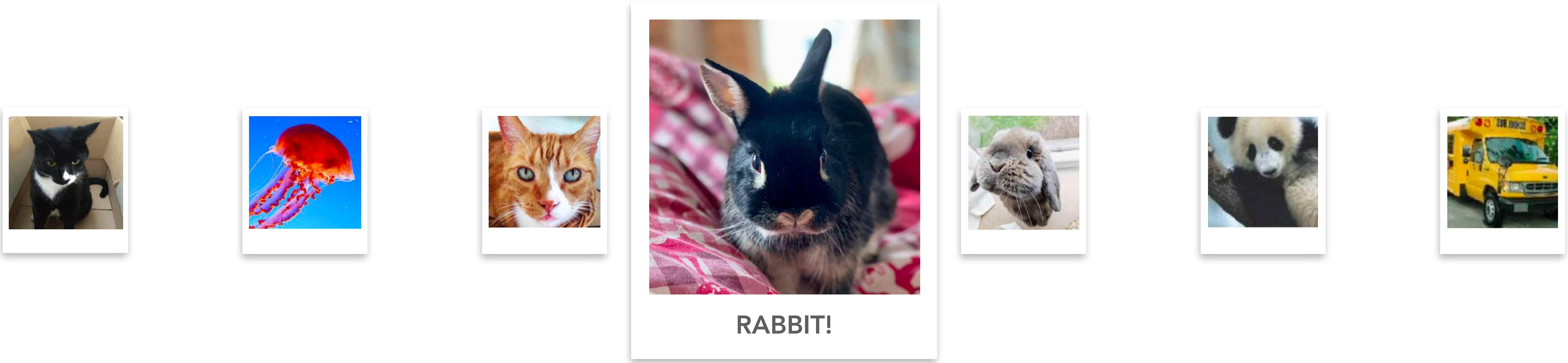
# Security *is* Adversarial




Training on ad fraud…    …attacks evolve at test time…

# Security *is* Adversarial

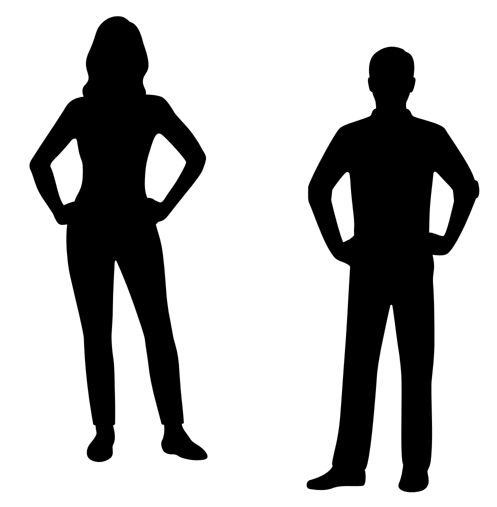


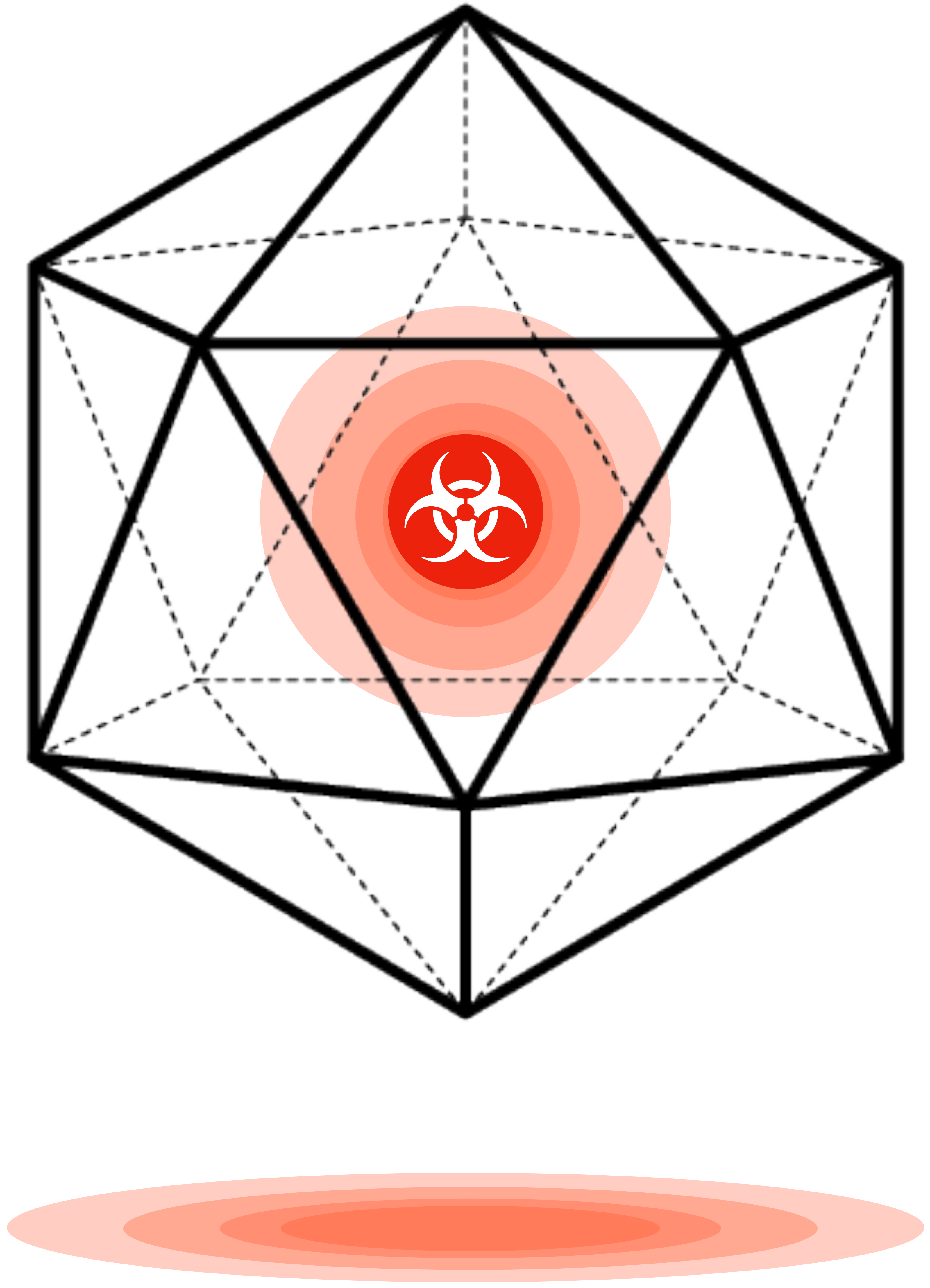




Training on ad fraud… …attacks evolve at test time… …prepare for the unknown

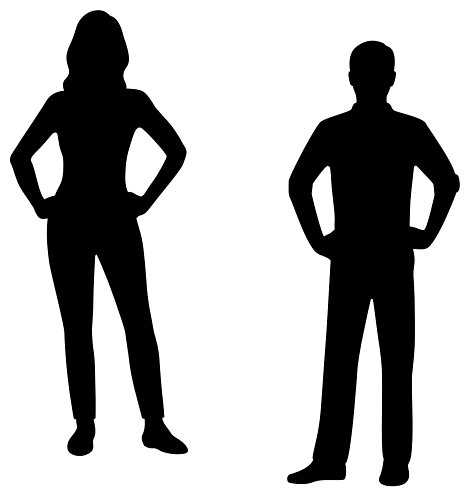
# Security *is* Adversarial



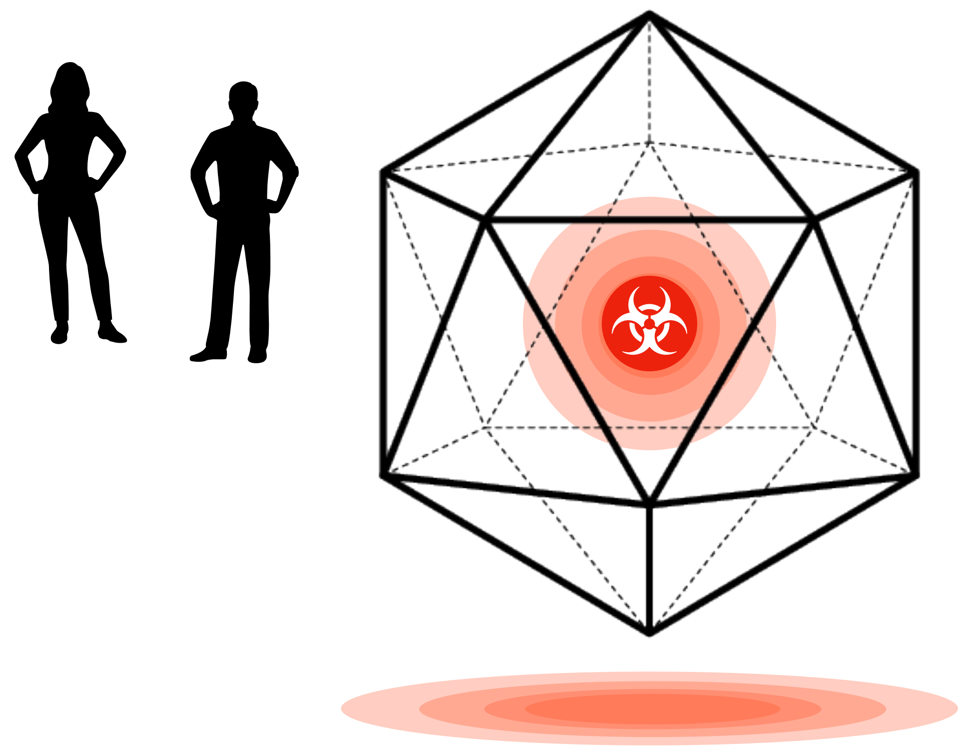


# Security *is* Adversarial

Oh yeah, that’s malware alright

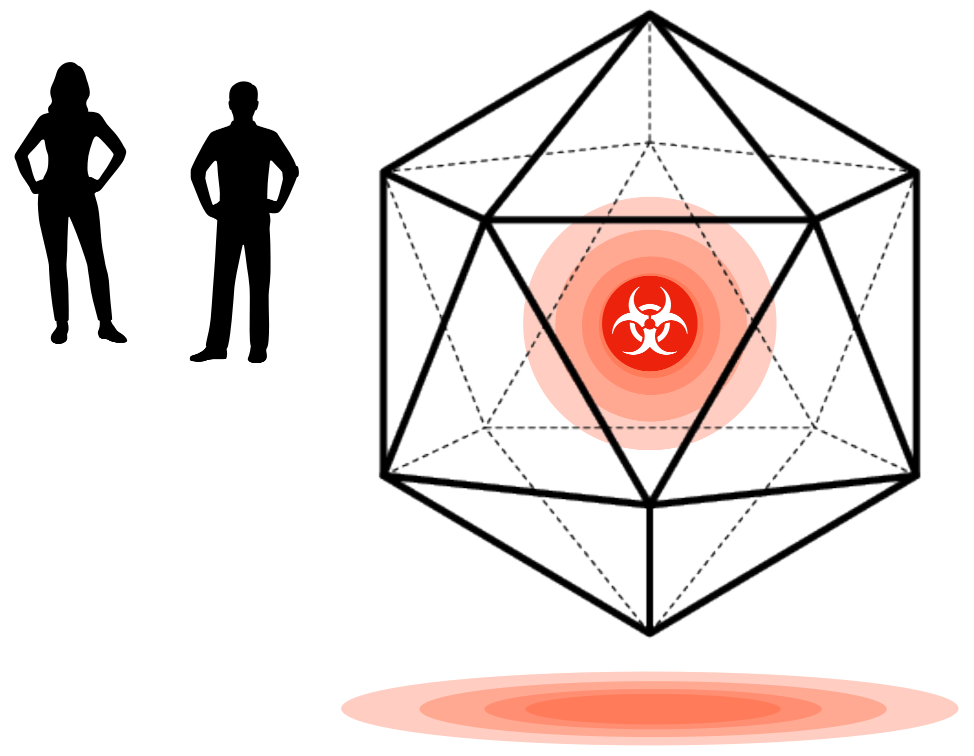


# We need...



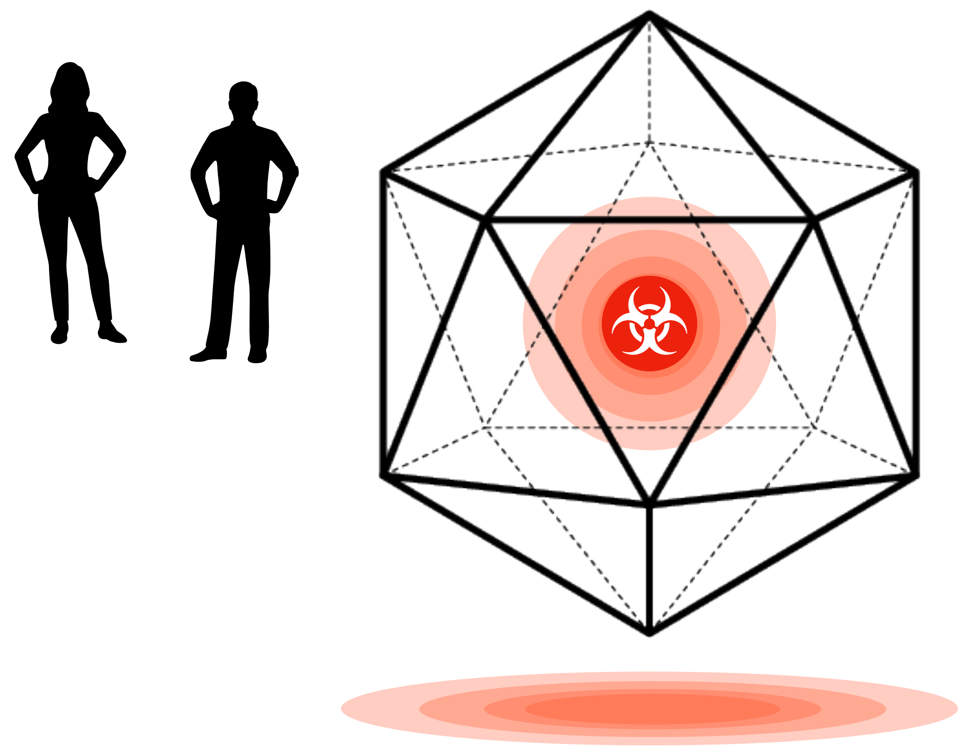
# We need...

To understand and improve the effectiveness of machine learning methods for systems security in the presence of adversaries



# We need...

**To understand and improve the effectiveness of machine learning methods for systems security in the presence of adversaries**

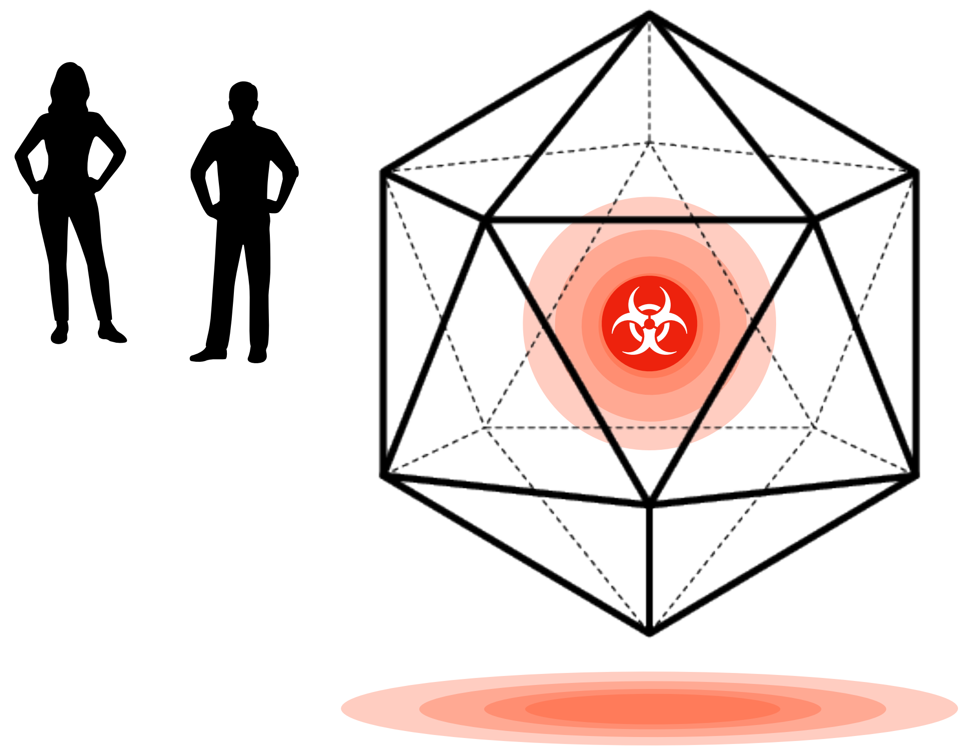


**Representation** of problem space objects (e.g., programs) results in a **semantic gap**

- It makes designing attacks and defenses more challenging
- It leaves room for adversarial manipulation
- It challenges the identification of causal vs non-causal (spurious) features

# We need...

**To understand and improve the effectiveness of machine learning methods for systems security in the presence of adversaries**



**Representation** of problem space objects (e.g., programs) results in a **semantic gap**

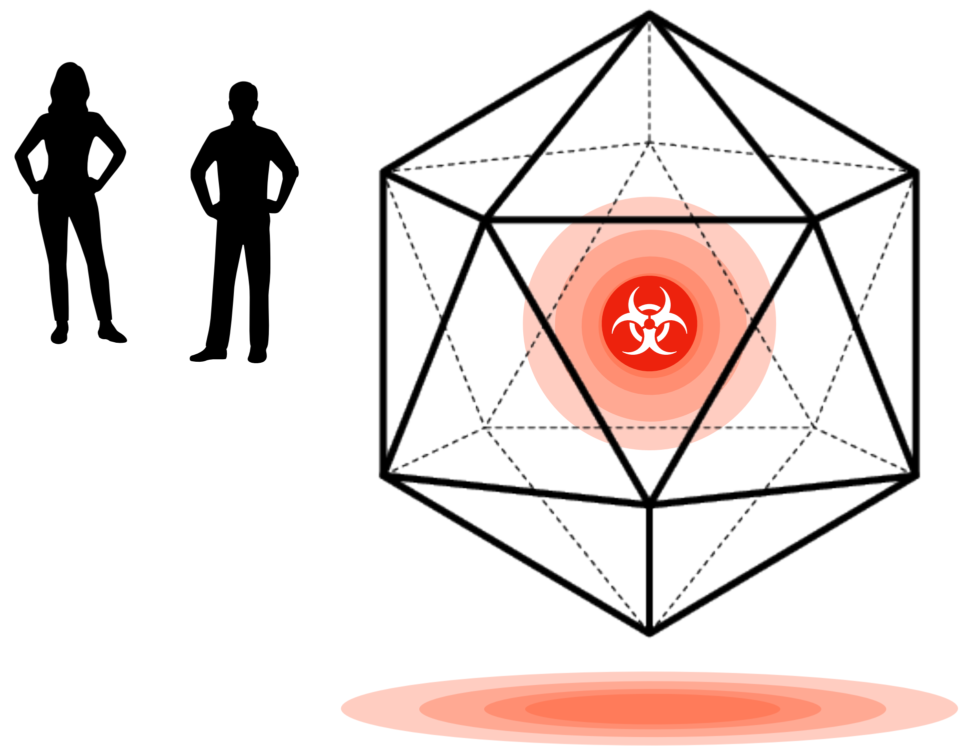
- It makes designing attacks and defenses more challenging
- It leaves room for adversarial manipulation
- It challenges the identification of causal vs non-causal (spurious) features

Effectiveness of ML for systems security is intertwined with the **underlying abstractions**, e.g., program analyses, to represent objects

- This affects robustness to adversarial drift, explainability, costs, and performance

# We need...

**To understand and improve the effectiveness of machine learning methods for systems security in the presence of adversaries**



**Representation** of problem space objects (e.g., programs) results in a **semantic gap**

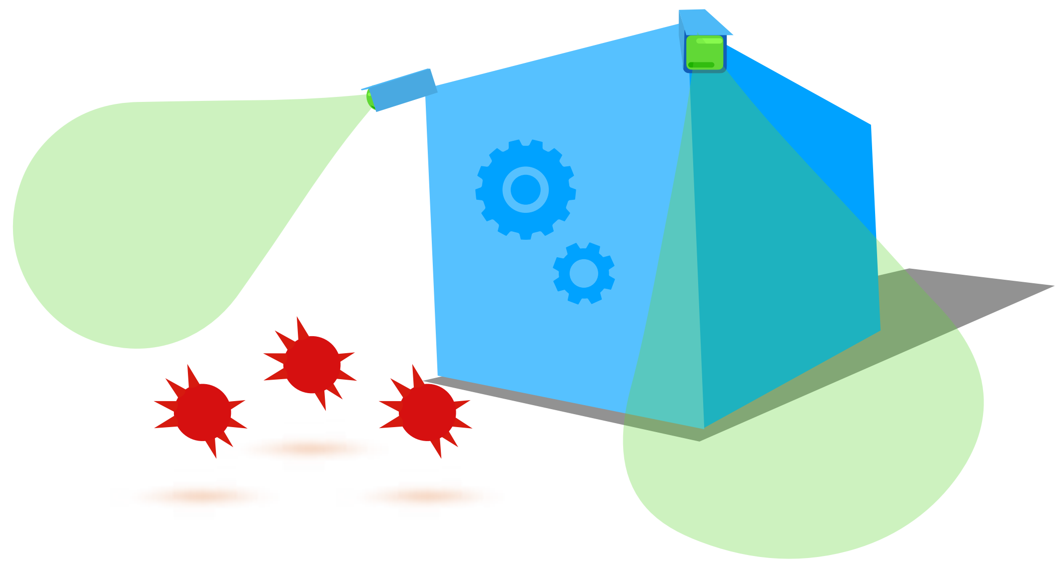
- It makes designing attacks and defenses more challenging
- It leaves room for adversarial manipulation
- It challenges the identification of causal vs non-causal (spurious) features

Effectiveness of ML for systems security is intertwined with the **underlying abstractions**, e.g., program analyses, to represent objects

- This affects robustness to adversarial drift, explainability, costs, and performance

**Is Trustworthy ML for systems security possible?**

# Outline

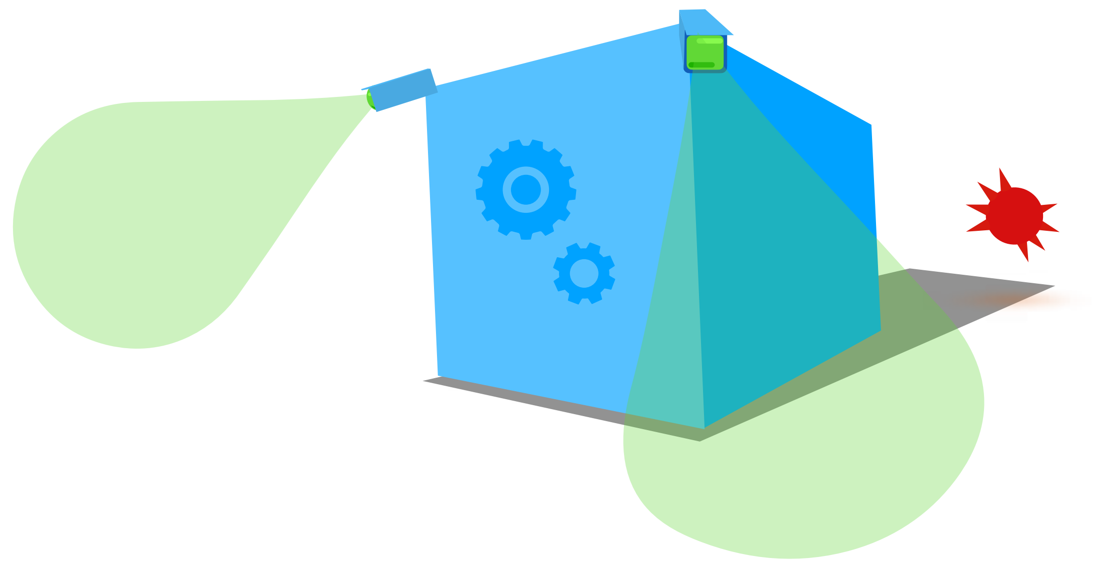


# Outline

Focus

## Adversarial ML evasion attacks against malware classifiers

- Classic formulation of evasion attacks is ill-suited for reasoning about realizable evasive malware
- By reformulating, we can propose stronger attacks and easily compare against alternatives
- Practical end-to-end automatic adversarial malware as a service — how about defenses?



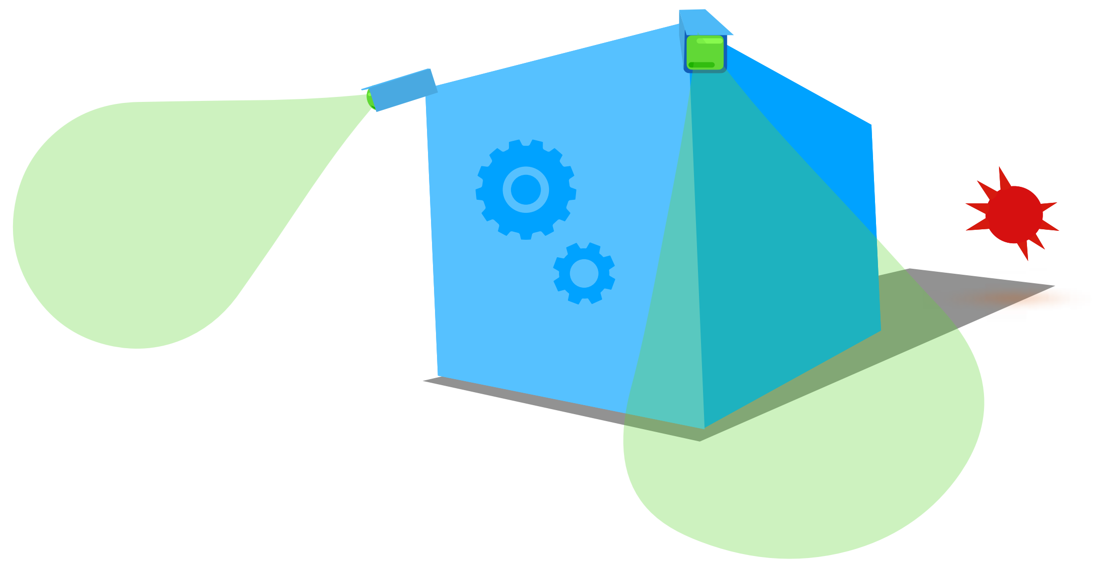
**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**

# Outline

Focus

## Adversarial ML evasion attacks against malware classifiers

- Classic formulation of evasion attacks is ill-suited for reasoning about realizable evasive malware
- By reformulating, we can propose stronger attacks and easily compare against alternatives
- Practical end-to-end automatic adversarial malware as a service — how about defenses?

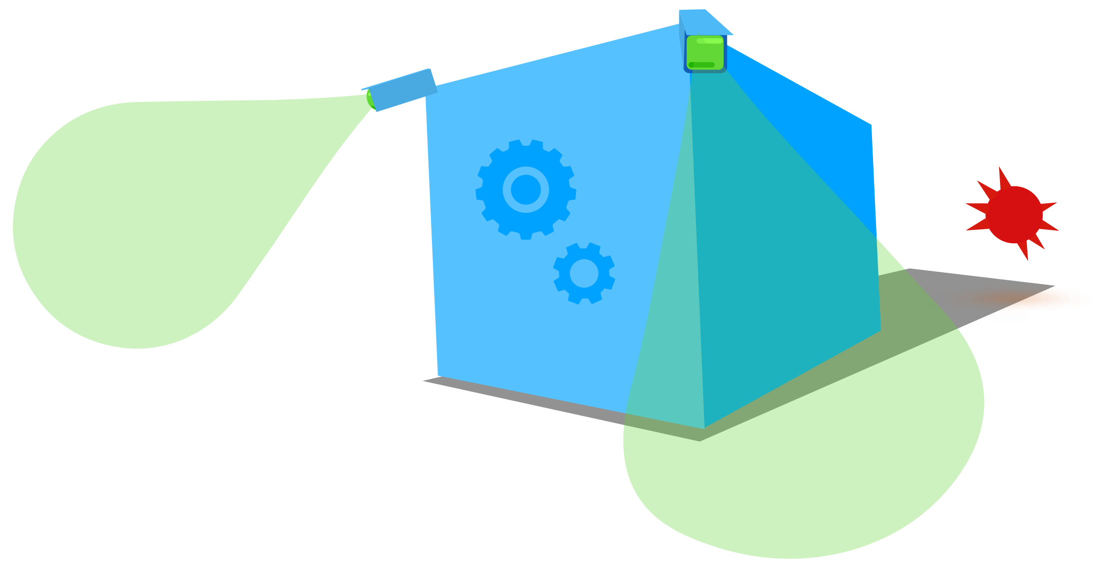


**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**

# Outline

Focus

## Adversarial ML evasion attacks against malware classifiers

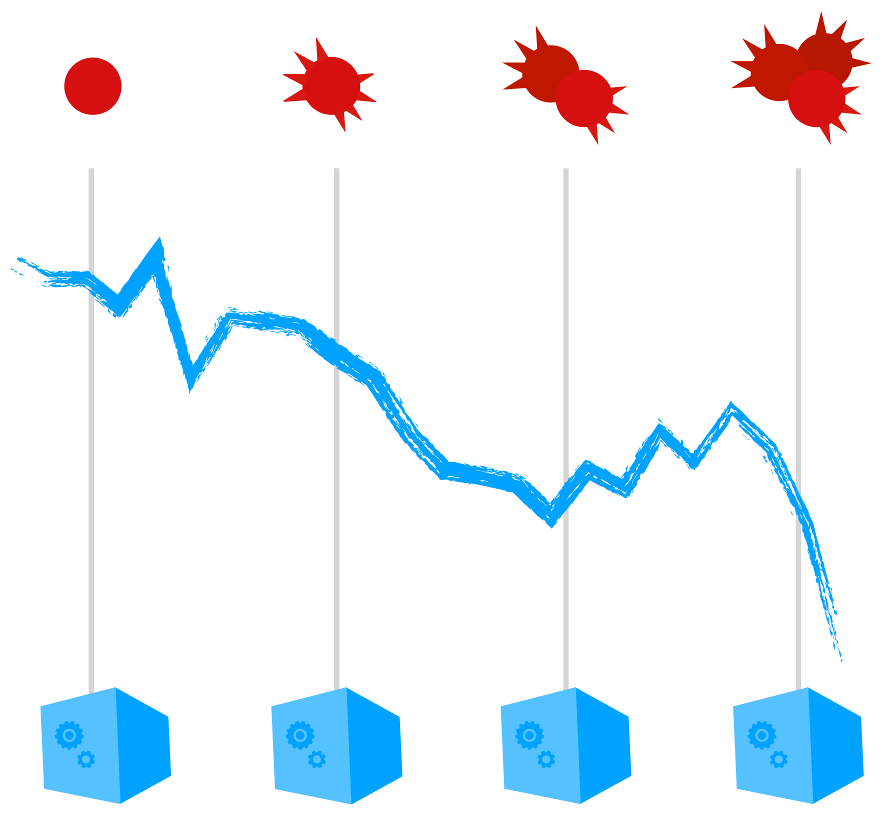


- Classic formulation of evasion attacks is ill-suited for reasoning about realizable evasive malware
- By reformulating, we can propose stronger attacks and easily compare against alternatives
- Practical end-to-end automatic adversarial malware as a service — how about defenses?

**[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space**

Bigger Picture

## Drifting scenarios caused by threats evolving over time



- How dataset shift affects machine learning-based detectors in security settings
- The need for time-aware evaluations and metrics
- Detecting shifts with abstaining classifiers and classification with rejection

**[USENIX Sec 2017 & IEEE S&P 2022] Transcend: Detecting Concept Drift in Malware Classification Models & Transcending Transcend: Revisiting Malware Classification in the Presence of Concept Drift**

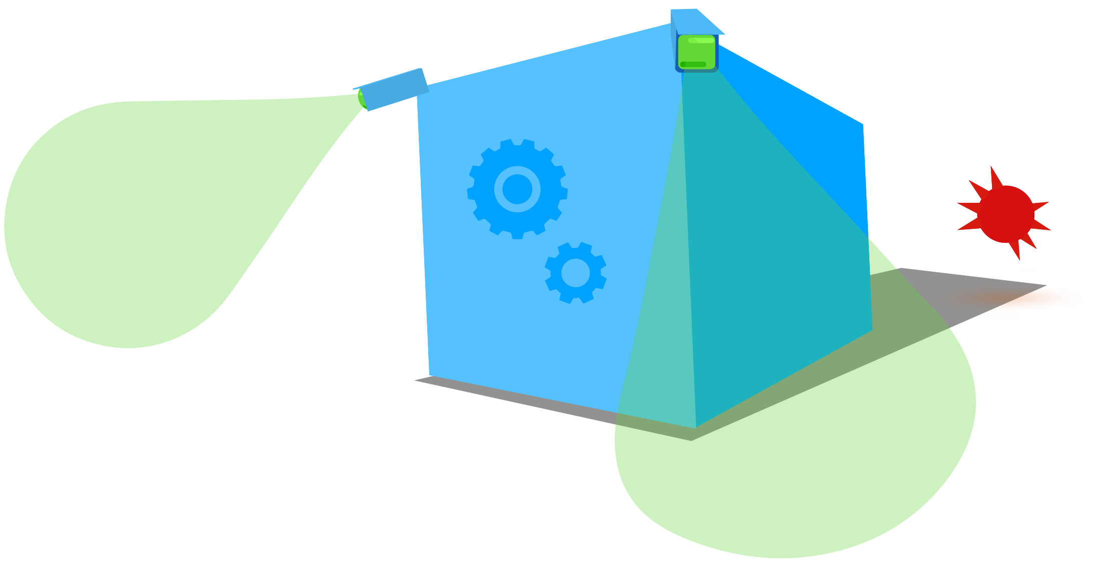
**[USENIX Sec 2019] TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time**

# Outline

Focus

## Adversarial ML evasion attacks against malware classifiers

- Classic formulation of evasion attacks is ill-suited for reasoning about realizable evasive malware
- By reformulating, we can propose stronger attacks and easily compare against alternatives
- Practical end-to-end automatic adversarial malware as a service — how about defenses?

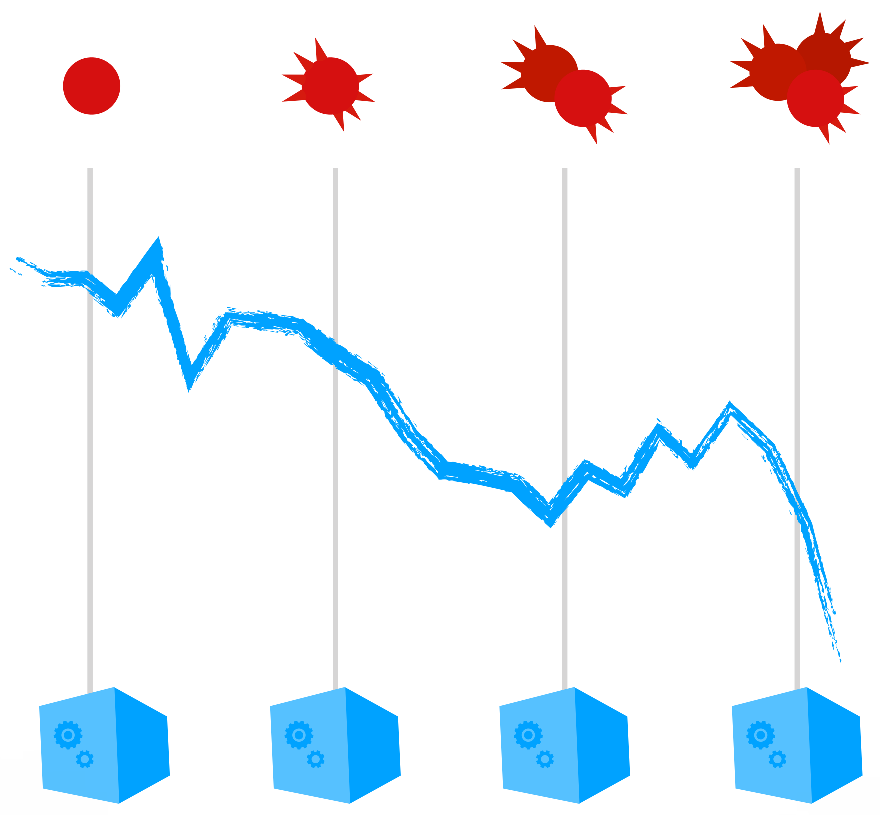


**[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space**

Bigger Picture

## Drifting scenarios caused by threats evolving over time

- How dataset drift affects machine learning-based detectors in security settings
- The need for time-aware evaluations and metrics
- Detecting shifts with abstaining classifiers and classification with rejection



**[USENIX Sec 2017 & IEEE S&P 2022] Transcend: Detecting Concept Drift in Malware Classification Models & Transcending Transcend: Revisiting Malware Classification in the Presence of Concept Drift**

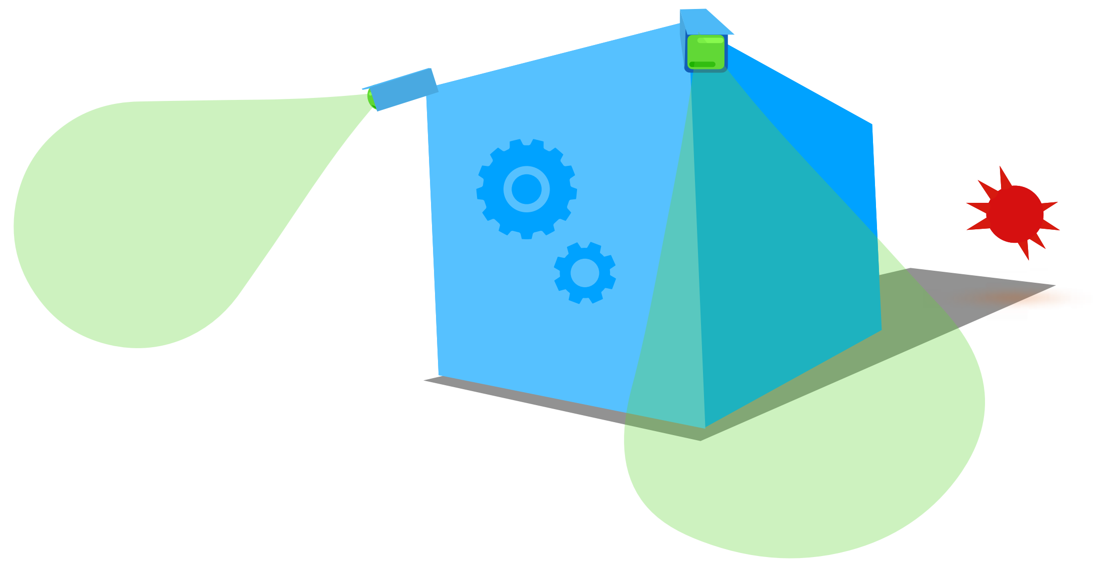
**[USENIX Sec 2019] TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time**

# Outline

Focus

## Adversarial ML evasion attacks against malware classifiers

- Classic formulation of evasion attacks is ill-suited for reasoning about realizable evasive malware
- By reformulating, we can propose stronger attacks and easily compare against alternatives
- Practical end-to-end automatic adversarial malware as a service — how about defenses?

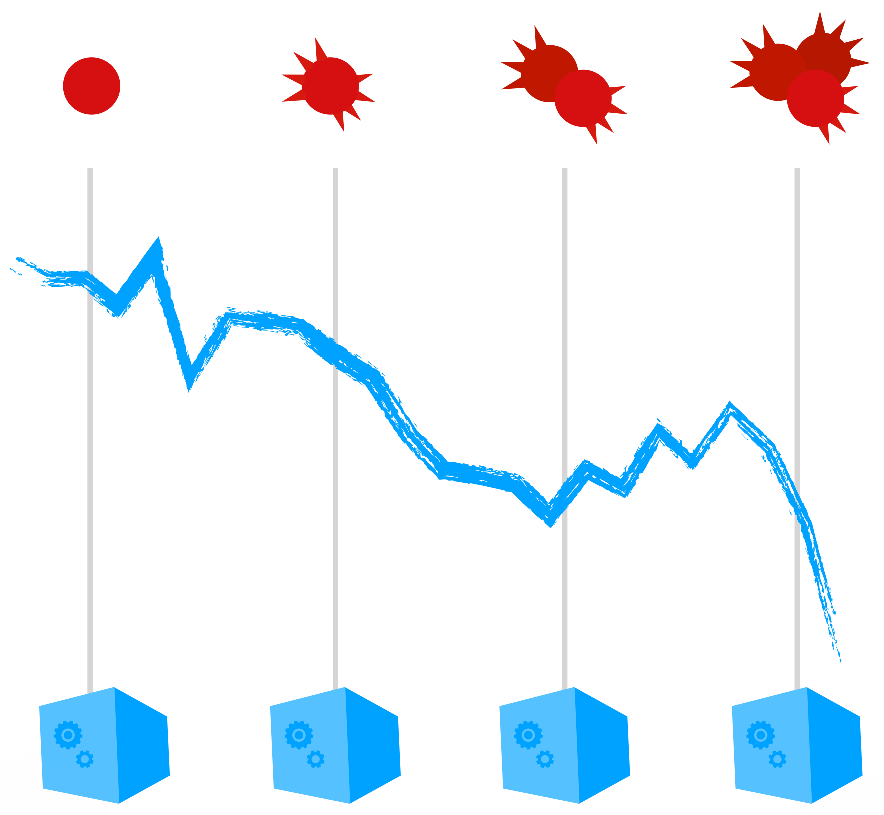


**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**

Bigger Picture

## Drifting scenarios caused by threats evolving over time

- How dataset drift affects machine learning-based detectors in security settings
- The need for time-aware evaluations and metrics
- Detecting shifts with abstaining classifiers and classification with rejection



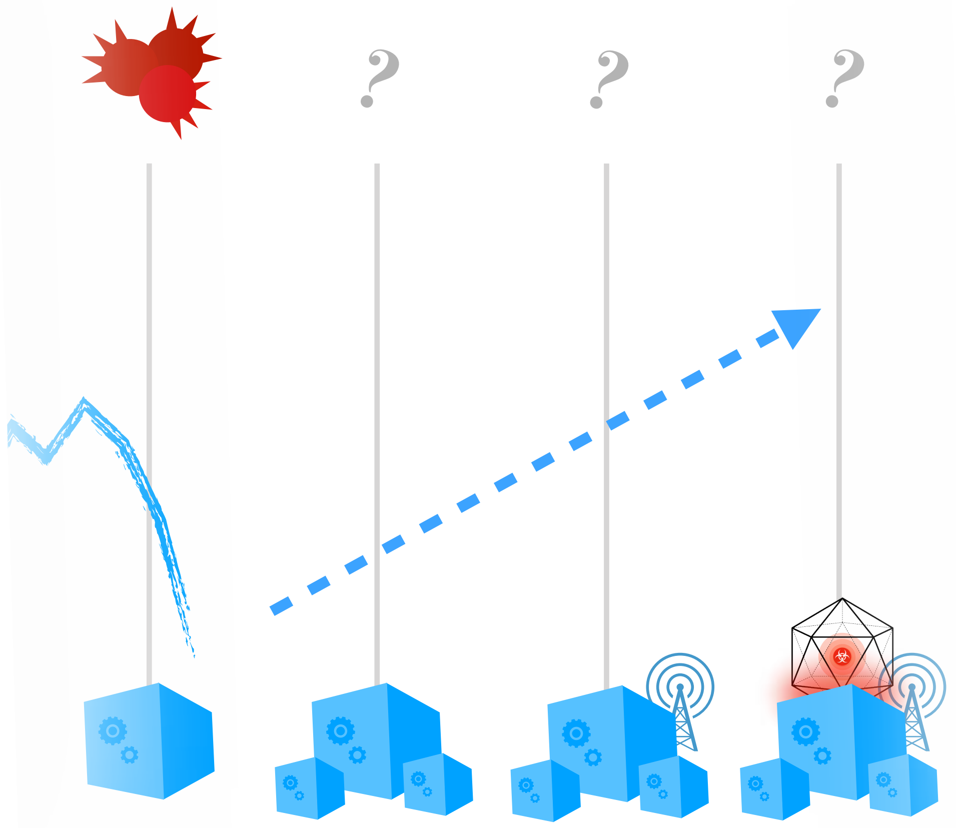
**[USENIX Sec 2017 & IEEE S&P 2022]** **Transcend: Detecting Concept Drift in Malware Classification Models & Transcending Transcend: Revisiting Malware Classification in the Presence of Concept Drift**

**[USENIX Sec 2019]** **TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time**

Looking Ahead

## Quo vadis?

- Discussion of the future of trustworthy ML for system security
- Robust representations, universal adversarial perturbations, realizable backdoors, drift forecasting, and the role of abstractions towards the Platonic ideal of semantics



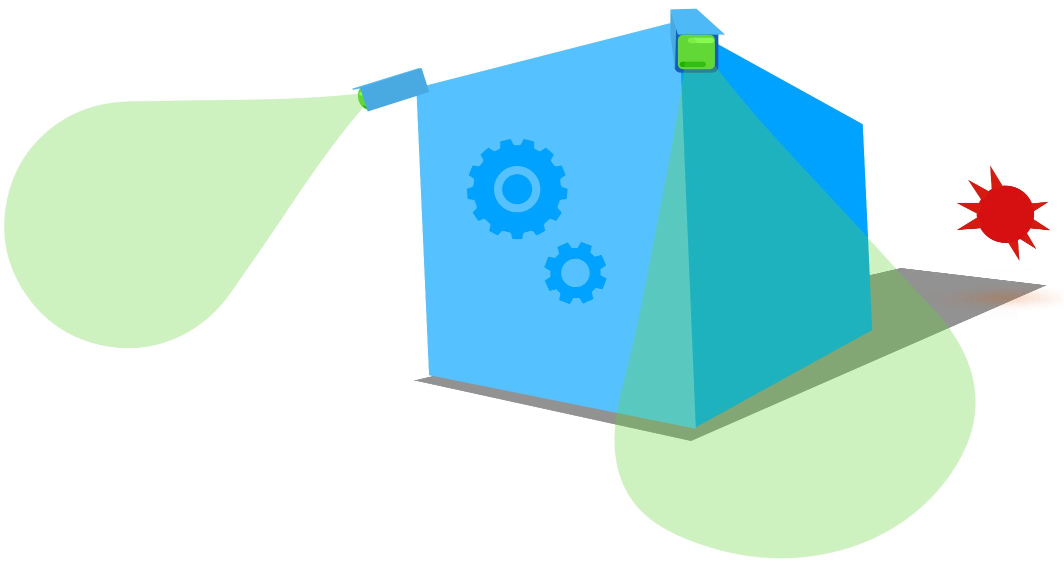
**[USENIX Sec 2022]** **Dos and Don'ts of Machine Learning in Computer Security**

# Outline

Focus

## Adversarial ML evasion attacks against malware classifiers

- Classic formulation of evasion attacks is ill-suited for reasoning about realizable evasive malware
- By reformulating, we can propose stronger attacks and easily compare against alternatives
- Practical end-to-end automatic adversarial malware as a service — how about defenses?

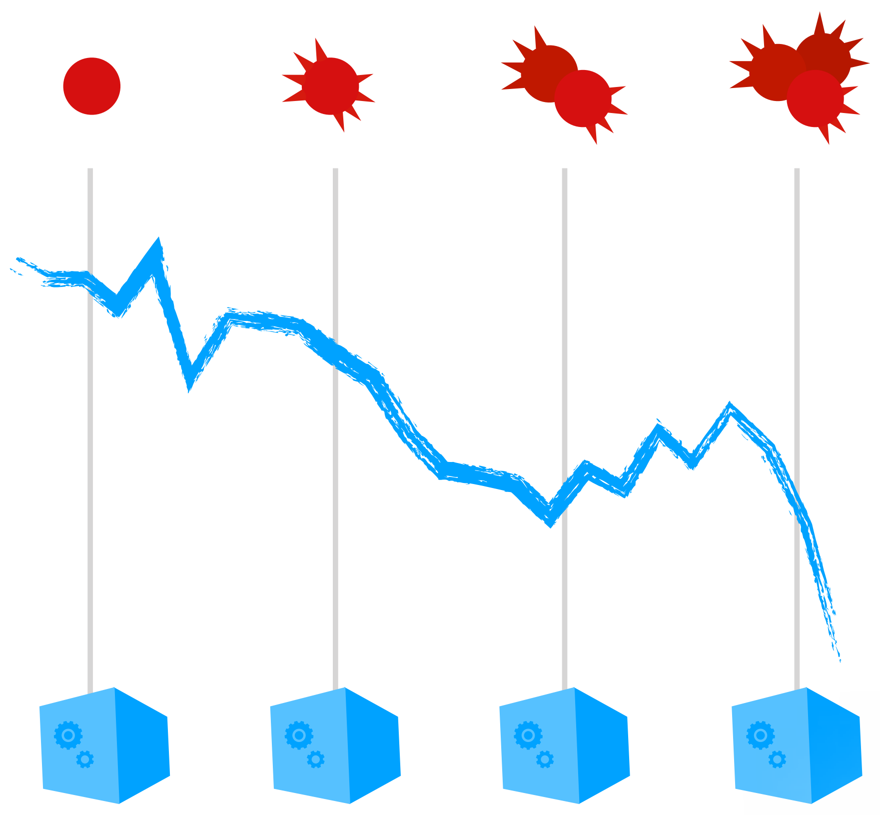


**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**

Bigger Picture

## Drifting scenarios caused by threats evolving over time

- How dataset shift affects machine learning-based detectors in security settings
- The need for time-aware evaluations and metrics
- Detecting shifts with abstaining classifiers and classification with rejection



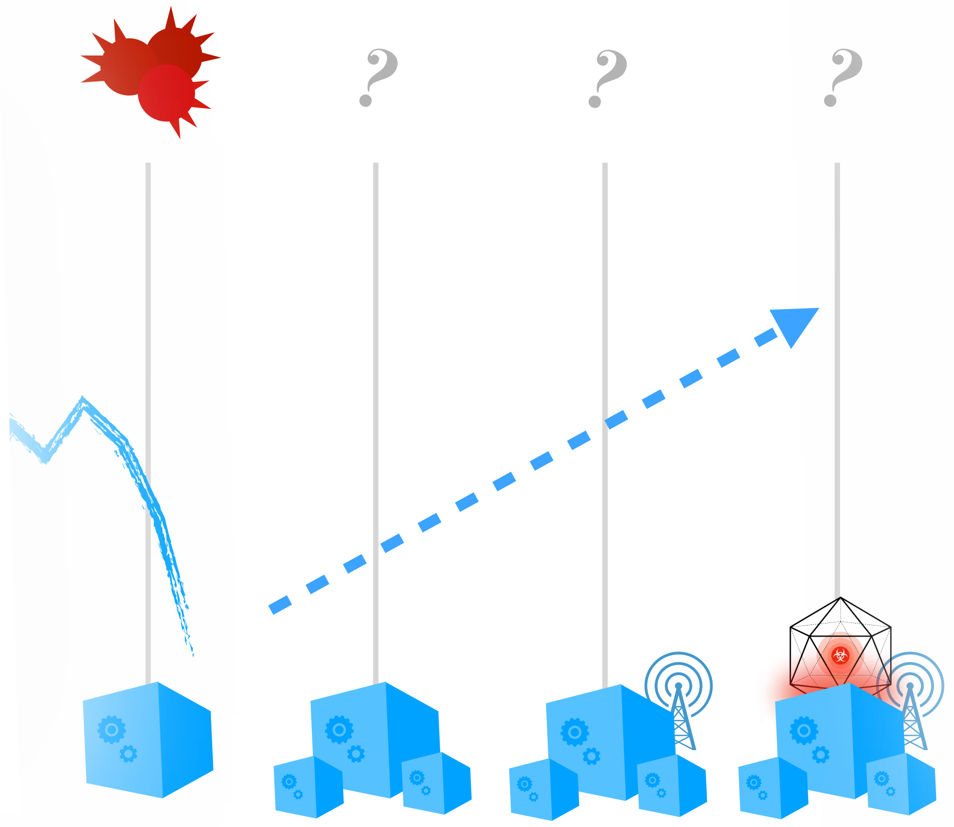
**[USENIX Sec 2017 & IEEE S&P 2022]** **Transcend: Detecting Concept Drift in Malware Classification Models & Transcending Transcend: Revisiting Malware Classification in the Presence of Concept Drift**

**[USENIX Sec 2019]** **TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time**

Looking Ahead

## Quo vadis?

- Discussion of the future of trustworthy ML for system security
- Robust feature development, universal adversarial perturbations, realizable backdoors, drift forecasting, and the role of abstractions towards the Platonic ideal of interesting behaviors



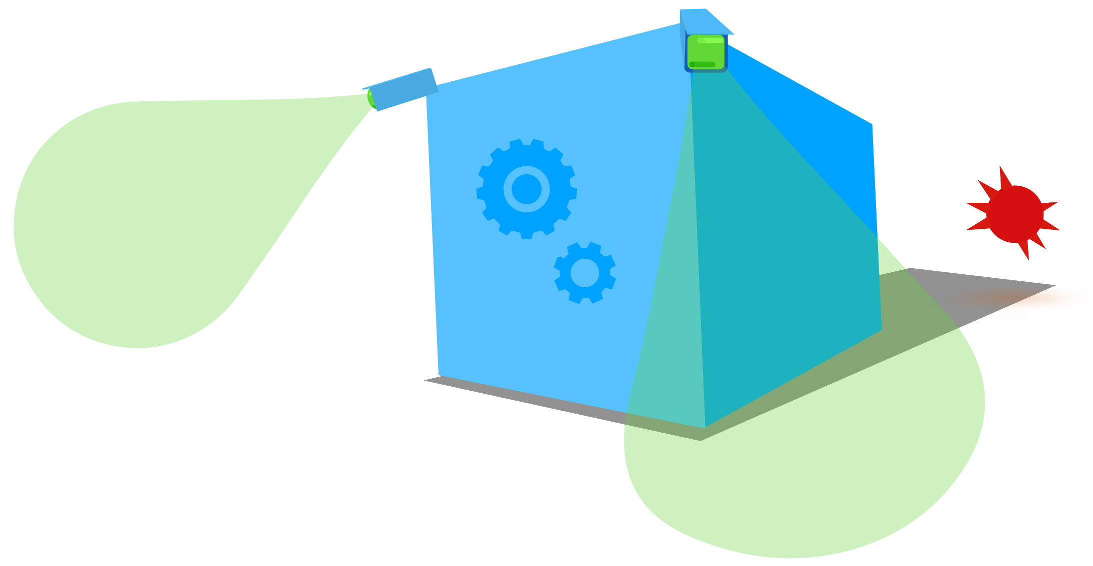
**[USENIX Sec 2022]** **Dos and Don'ts of Machine Learning in Com**

# Outline

Focus

## Adversarial ML evasion attacks against malware classifiers

- Classic formulation of evasion attacks is ill-suited for reasoning about realizable evasive malware
- By reformulating, we can propose stronger attacks and easily compare against alternatives
- Practical end-to-end automatic adversarial malware as a service — how about defenses?



[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space

Bigger Picture

## Drifting scenarios caused by threats evolving over time

- How dataset shift affects machine learning-based detectors in security settings
- The need for time-aware evaluations and metrics
- Detecting shifts with abstaining classifiers and classification with rejection



[USENIX Sec 2017 & IEEE S&P 2022] Transcend: Detecting Concept Drift in Malware Classification Models & Transcending Transcend: Revisiting Malware Classification in the Presence of Concept Drift

[USENIX Sec 2019] TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time

Looking Ahead

## Quo vadis?

- Discussion of the future of trustworthy ML for system security
- Robust feature development, universal adversarial perturbations, realizable backdoors, drift forecasting, and the role of abstractions towards the Platonic ideal of interesting behaviors



[USENIX Sec 2022] Dos and Don'ts of Machine Learning in Com

# A Dystopian Future…

Pandas are forbidden!

Guilty of being too cute!

# A Dystopian Future...

Pandas are forbidden!

Guilty of being too cute!

# A Dystopian Future…

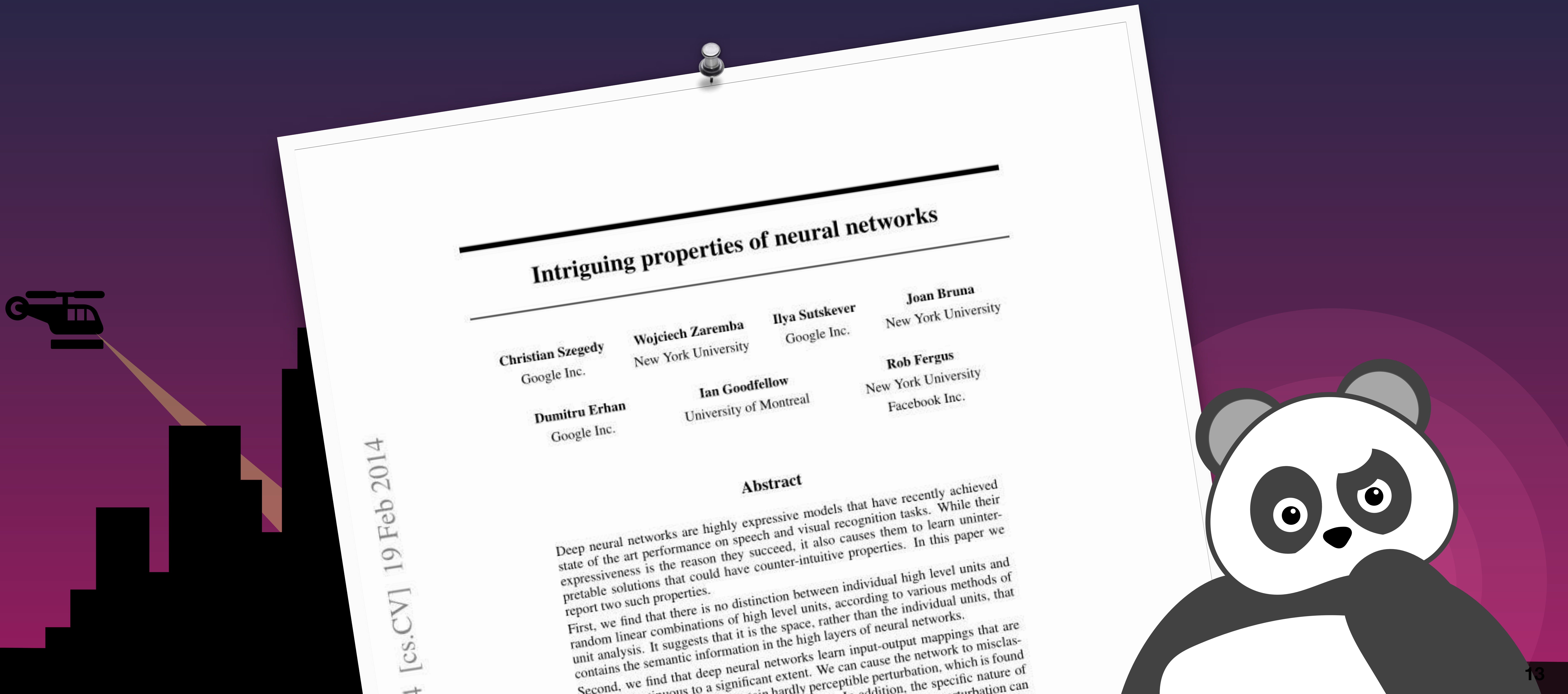
Pandas are forbidden!

Guilty of being too cute!

Luckily, pandas are fluent in math…

Luckily, pandas are fluent in math…

Luckily, pandas are fluent in math…



# Intriguing properties of neural networks


**Christian Szegedy** Google Inc.
**Wojciech Zaremba** New York University
**Ilya Sutskever** Google Inc.
**Joan Bruna** New York University
**Dumitru Erhan** Google Inc.
**Ian Goodfellow** University of Montreal
**Rob Fergus** New York University Facebook Inc.



[cs.CV] 19 Feb 2014



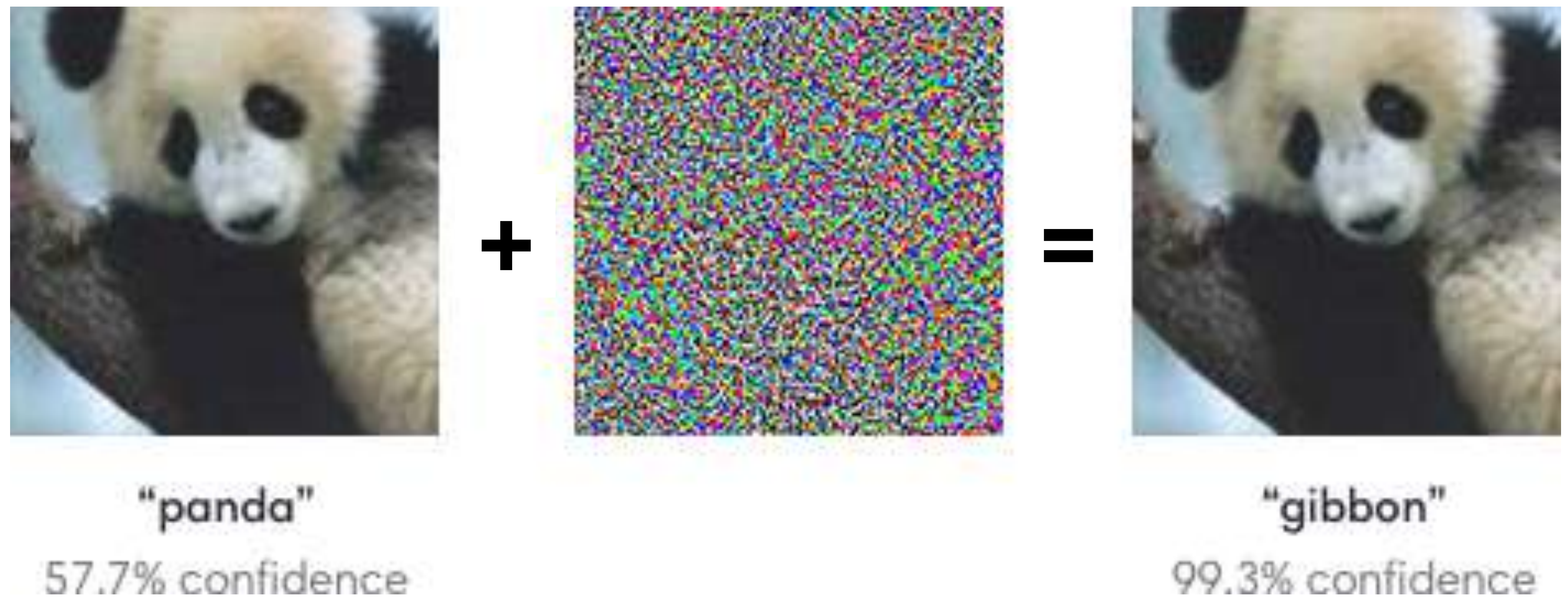
## Abstract

Deep neural networks are highly expressive models that have recently achieved state of the art performance on speech and visual recognition tasks. While their expressiveness is the reason they succeed, it also causes them to learn uninterpretable solutions that could have counter-intuitive properties. In this paper we report two such properties.

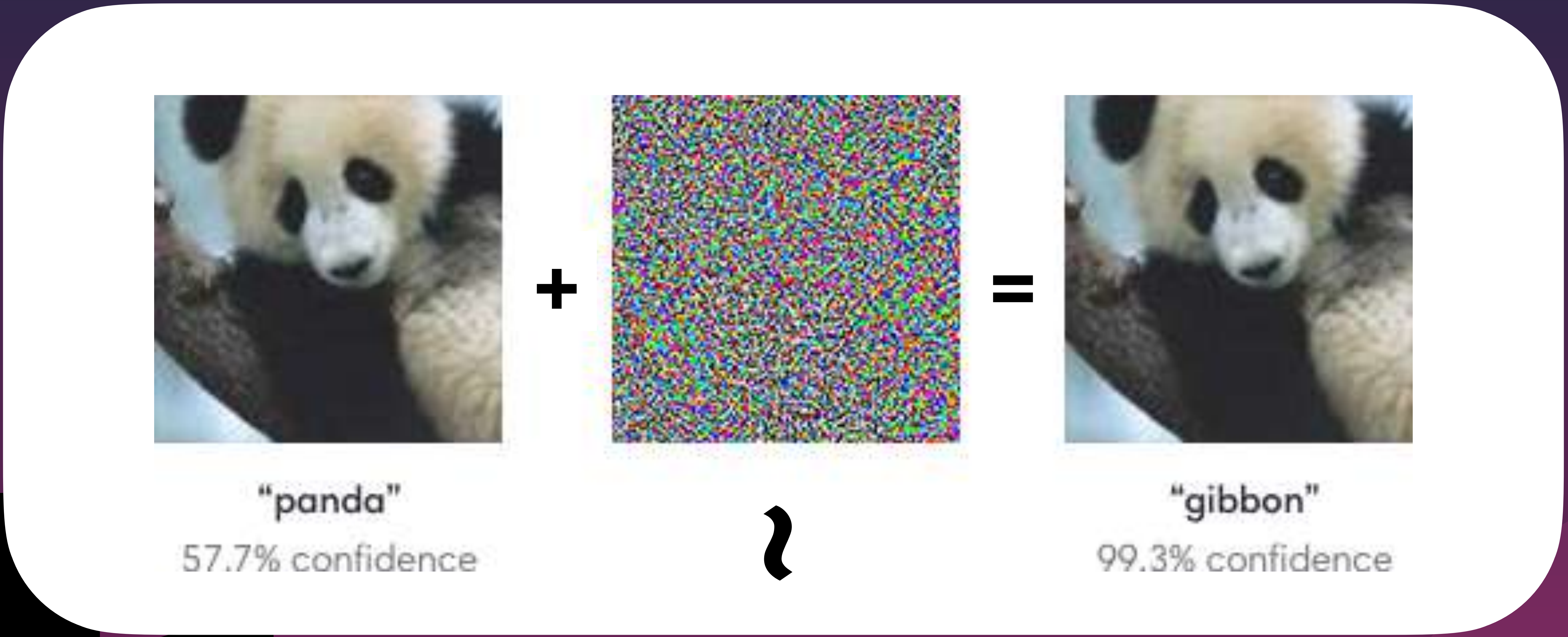
First, we find that there is no distinction between individual high level units and random linear combinations of high level units, according to various methods of unit analysis. It suggests that it is the space, rather than the individual units, that contains the semantic information in the high layers of neural networks.

Second, we find that deep neural networks learn input-output mappings that are … to a significant extent. We can cause the network to misclas- … hardly perceptible perturbation, which is found … In addition, the specific nature of … perturbation can

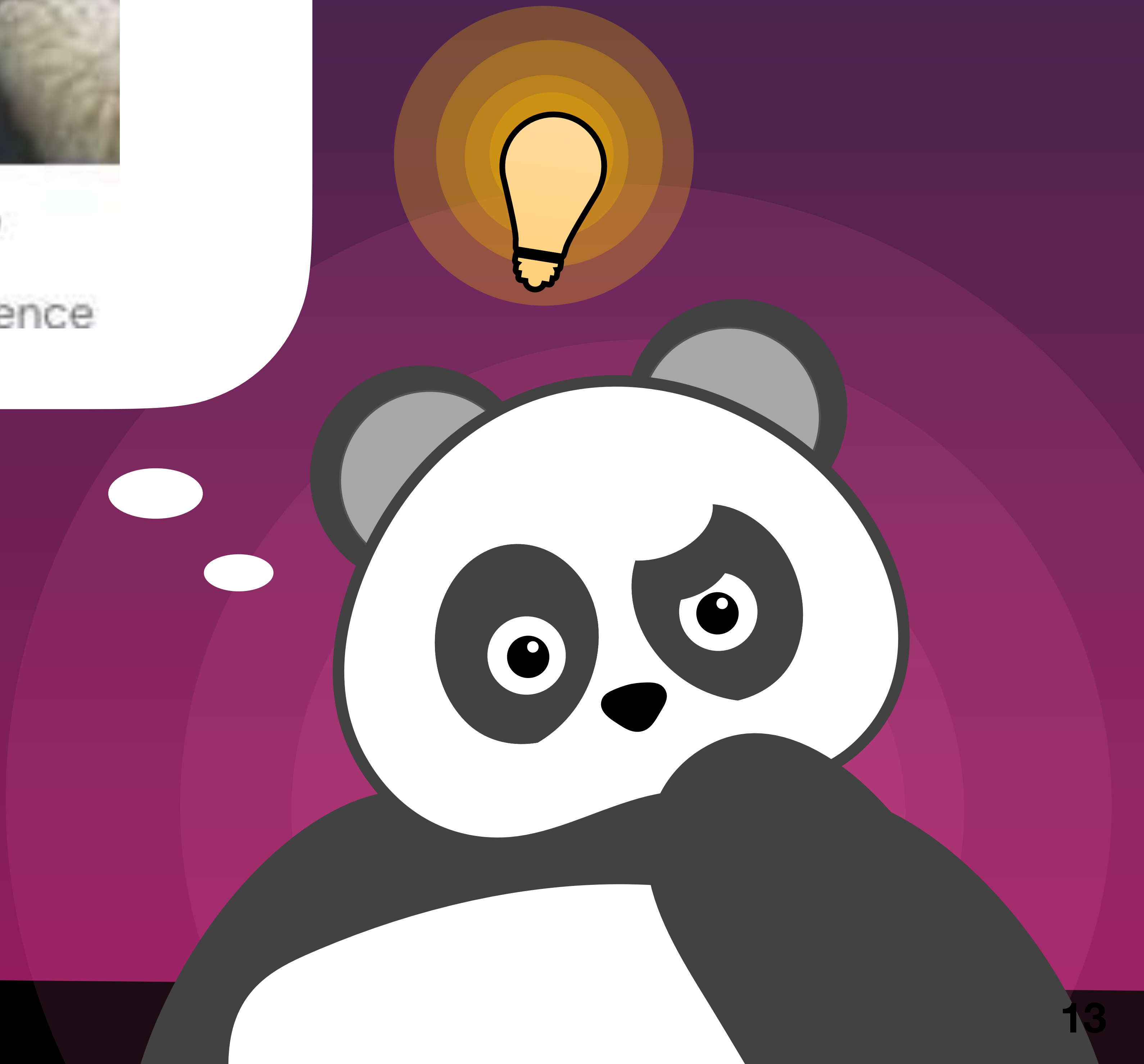
Luckily, pandas are fluent in math…

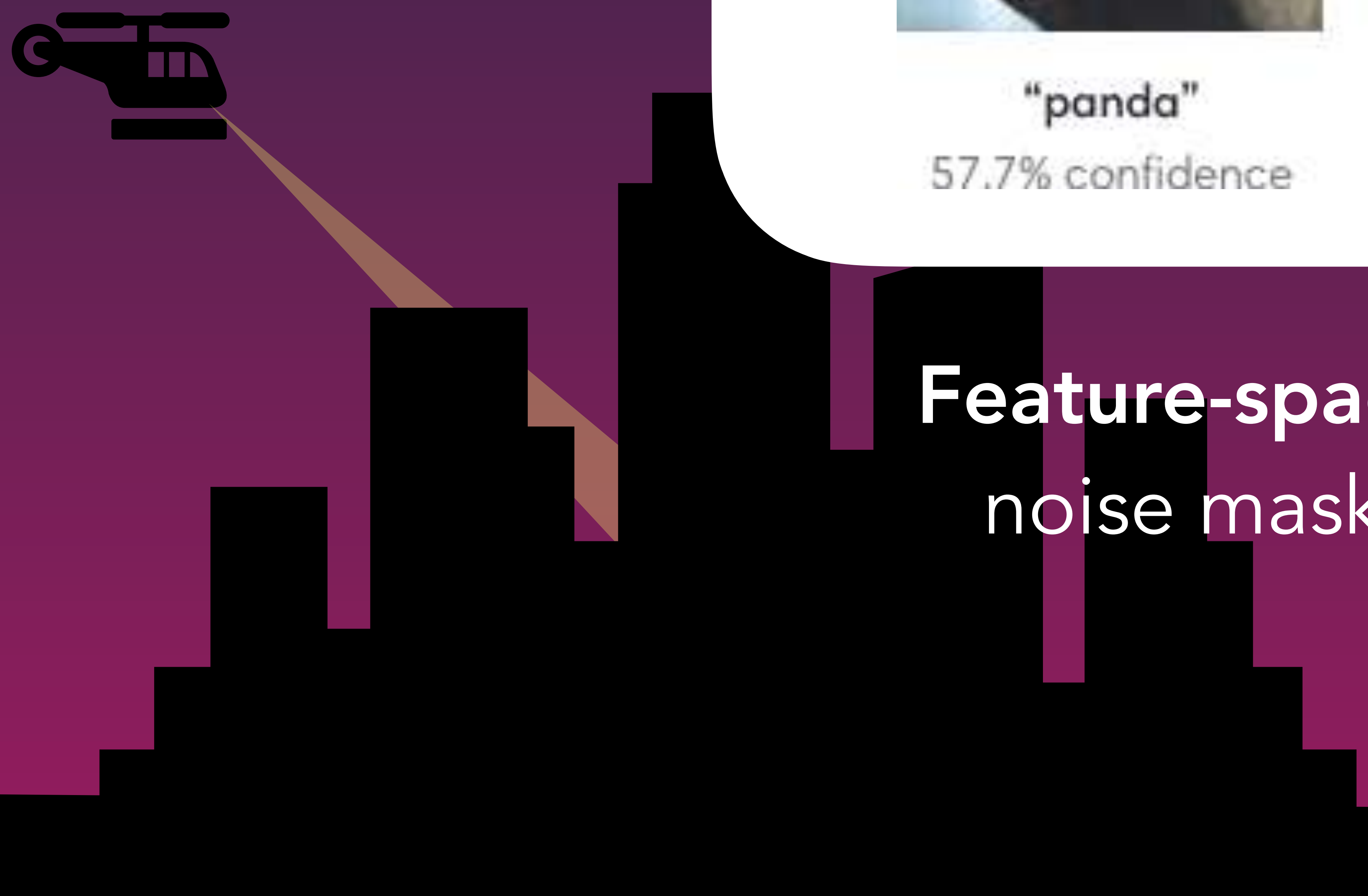



Luckily, pandas are fluent in math…




**Feature-space**
noise mask



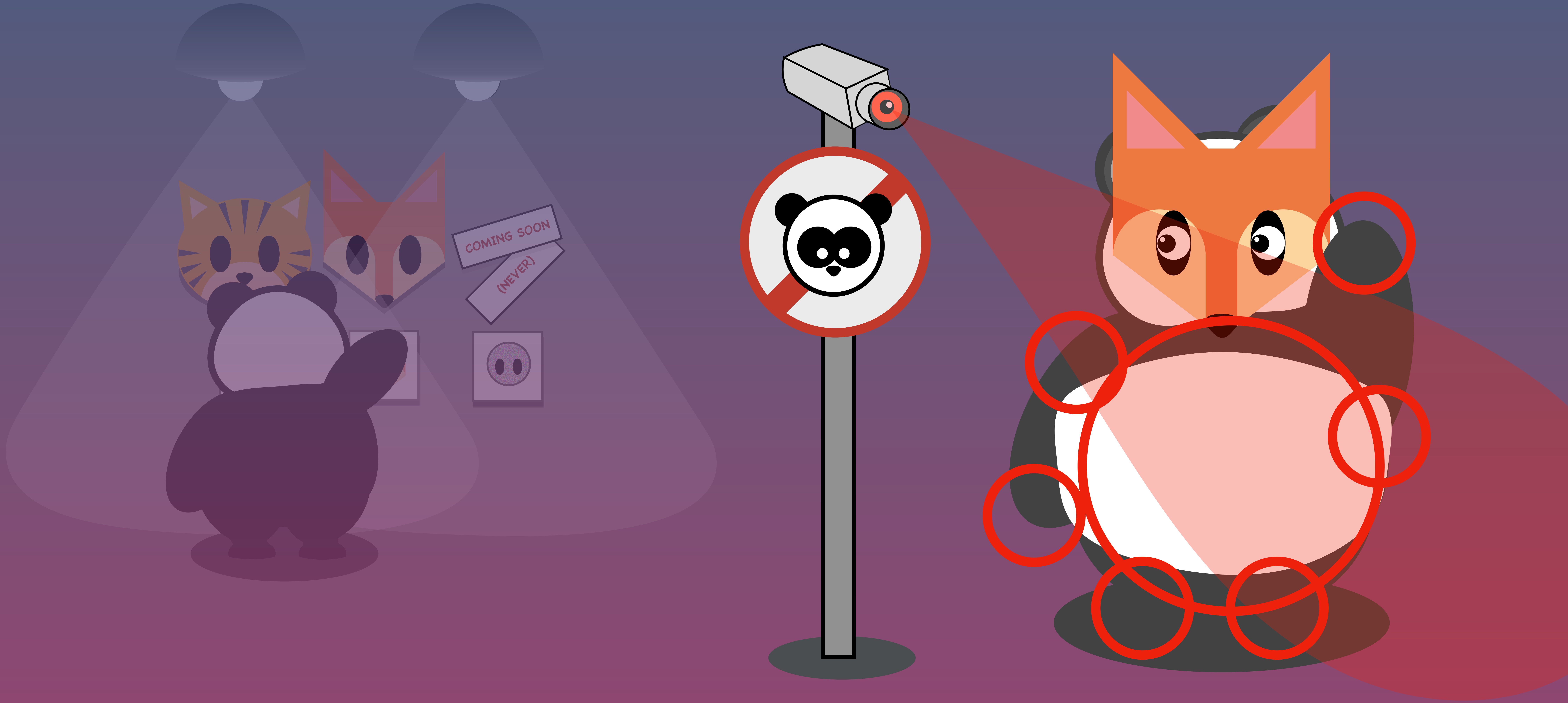


What happens in the **problem space**, i.e., the real world?

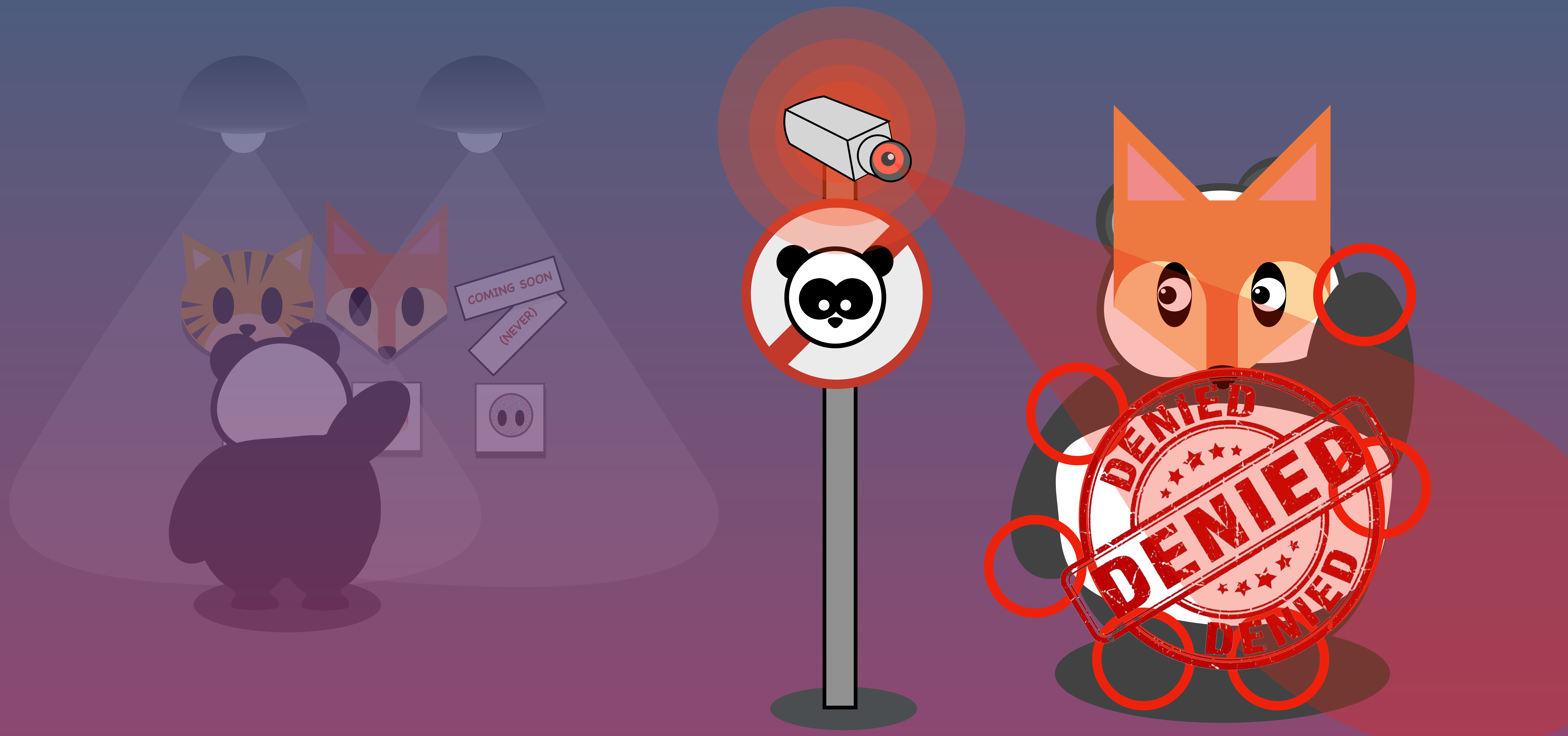
What happens in the **problem space**, i.e., the real world?

What happens in the **problem space**, i.e., the real world?

What happens in the **problem space**, i.e., the real world?



What happens in the **problem space**, i.e., the real world?



# Let’s Analyze What Happened

# Let's Analyze What Happened

**Feature-Space Attacks**

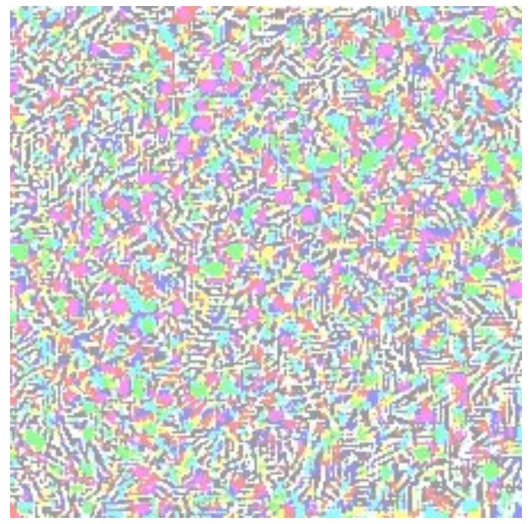
Original Image

"panda" 57.7%

$x$

+

Perturbation



*imperceptible noise*

$\delta$

=

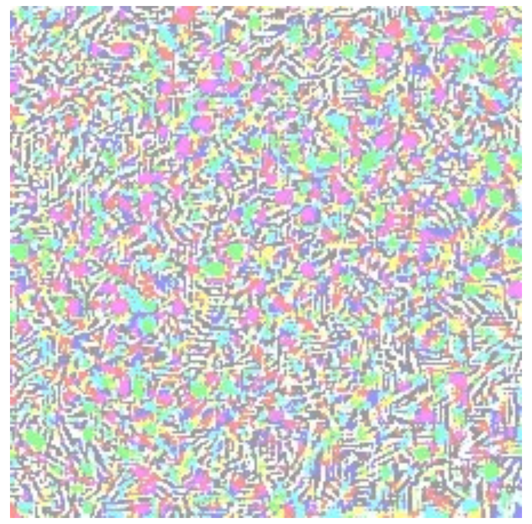
Adv. Image

"gibbon" 99.3%

$x + \delta$

# Let’s Analyze What Happened

**Feature-Space Attacks**

| Original Image | | Perturbation | | Adv. Image |
|---|---|---|---|---|
|  | + |  | = |  |
| “panda” 57.7% | | *imperceptible noise* | | “gibbon” 99.3% |
| $x$ | | $\delta$ | | $x + \delta$ |

Optimization | **minimize**$_\delta$ $\|\|\delta\|\|_p + c \cdot f(x+\delta)$

# Let's Analyze What Happened

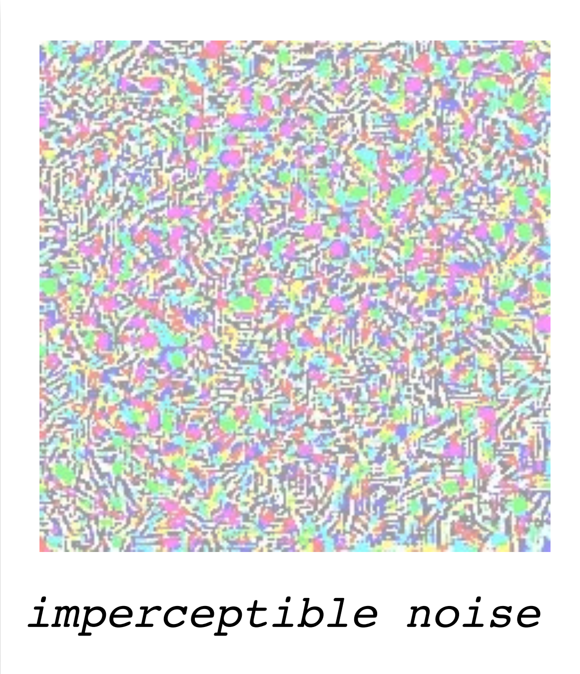
**Feature-Space Attacks**

Original Image


$x$

+

Perturbation




$\delta$

=

Adv. Image

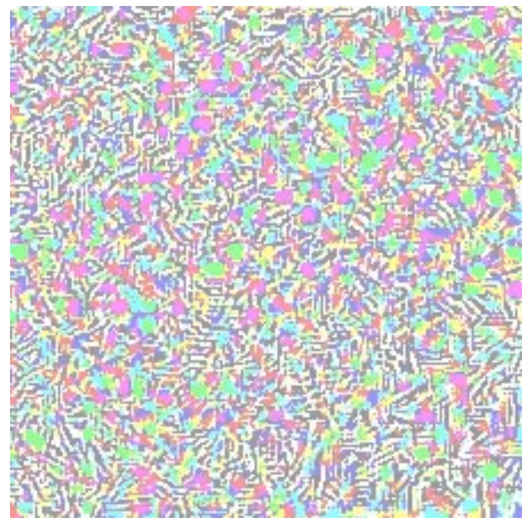

$x + \delta$

Optimization | **minimize**$_\delta$ $||\delta||_p + c \cdot f(x + \delta)$

Pixel Perturbations

# Let's Analyze What Happened

**Feature-Space Attacks**

| Original Image | | Perturbation | | Adv. Image |
|---|---|---|---|---|
|  | + |  | = |  |
| "panda" 57.7% | | *imperceptible noise* | | "gibbon" 99.3% |
| $x$ | | $\delta$ | | $x + \delta$ |

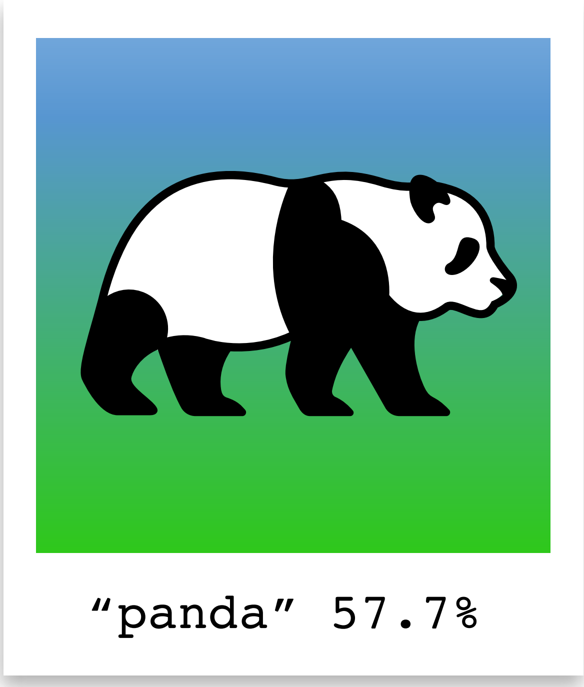
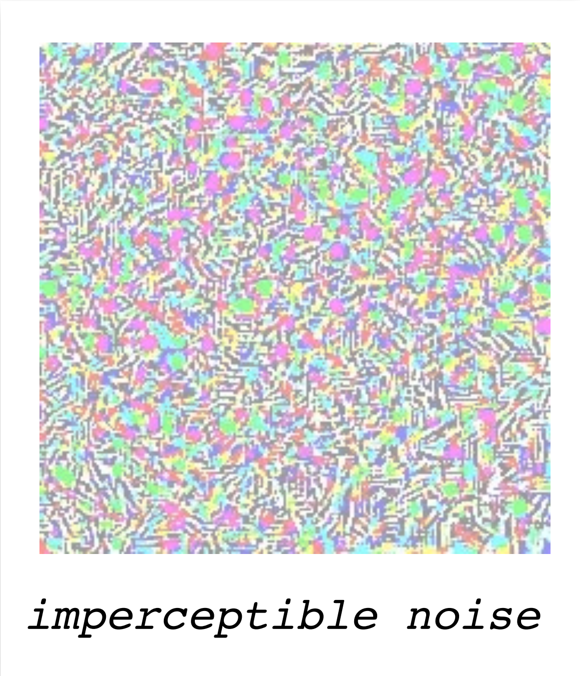
Optimization | **minimize**$_\delta$ $||\delta||_p + c \cdot f(x+\delta)$

Pixel Perturbations

Loss of Target Class

# Let’s Analyze What Happened

**Feature-Space Attacks**

| Original Image | | Perturbation | | Adv. Image |
|---|---|---|---|---|
|   | + |   | = |   |
| $x$ | | $\delta$ | | $x + \delta$ |

Optimization

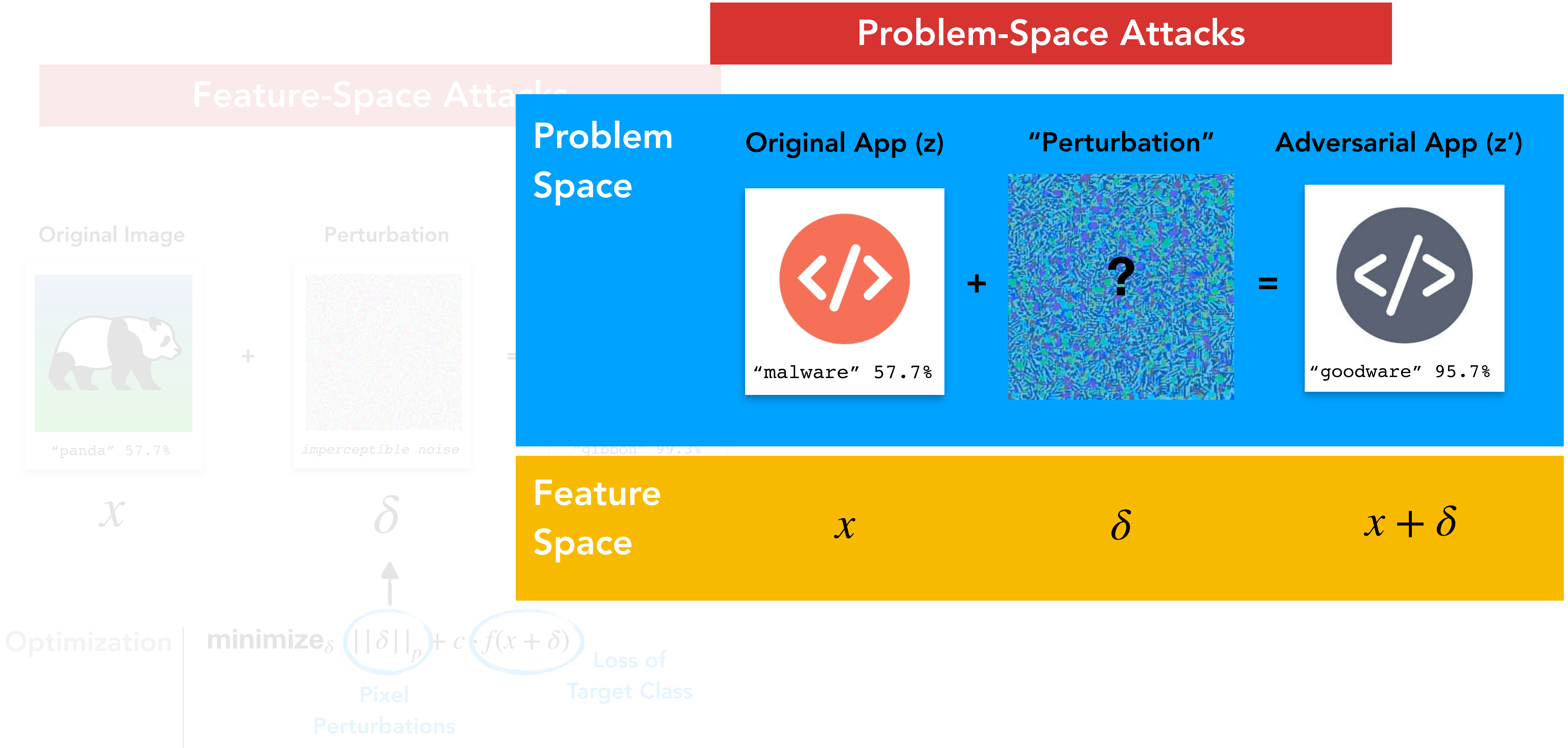
**minimize**$_\delta$ $||\delta||_p + c \cdot f(x+\delta)$

Pixel Perturbations

Loss of Target Class

**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Let’s Analyze What Happened




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Let’s Analyze What Happened

**Problem-Space Attacks**

| | Original App (z) | “Perturbation” | Adversarial App (z′) |
|---|---|---|---|
| **Problem Space** | “malware” 57.7% | ? | “goodware” 95.7% |
| **Feature Space** | $x$ | $\delta$ | $x + \delta$ |

**Optimization** | **minimize**$_\delta$ $\cancel{||\delta||_p} + c \cdot f(x+\delta)$

Feature-Space Atta…

Original Image | Perturbation

“panda” 57.7% | *imperceptible noise* | “gibbon” 99.3%

$x$ + $\delta$ =

Optimization | minimize$_\delta$ $||\delta||_p + c \cdot f(x+\delta)$

Pixel Perturbations | Loss of Target Class

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Let’s Analyze What Happened

## Problem-Space Attacks

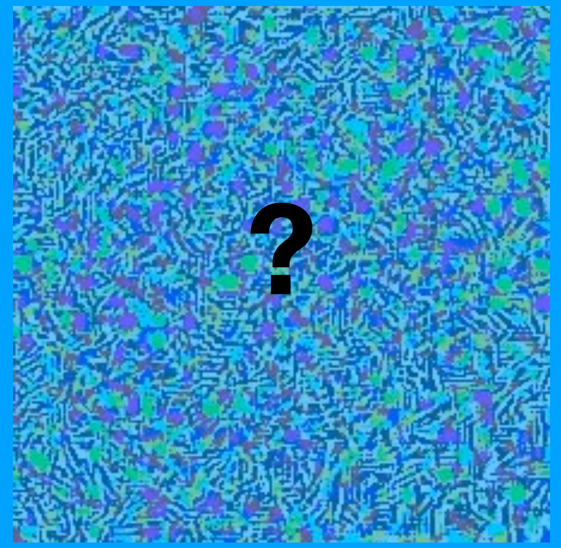
Feature-Space Attacks

| | Original Image | Perturbation | |
|---|---|---|---|
| | “panda” 57.7% | imperceptible noise | “gibbon” 99.3% |
| | $x$ | $\delta$ | |

Optimization | minimize$_\delta$ $||\delta||_p + c \cdot f(x+\delta)$

Pixel Perturbations — Loss of Target Class

| | Original App (z) | “Perturbation” | Adversarial App (z′) |
|---|---|---|---|
| **Problem Space** | “malware” 57.7% | + ? = | “goodware” 95.7% |
| **Feature Space** | $x$ | $\delta$ | $x + \delta$ |



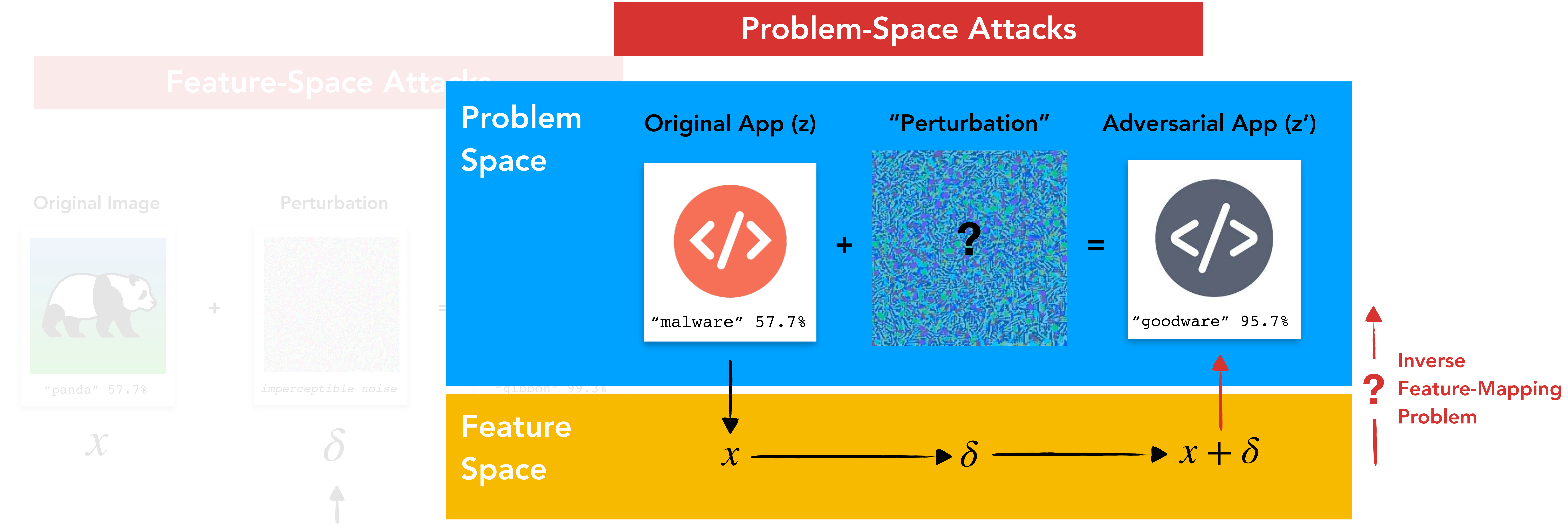
**Optimization** | **minimize**$_\delta$ $\cancel{||\delta||_p} + c \cdot f(x+\delta)$

**Constraints**

- Is it realistic/plausible?
- Does it crash?
- Does it preserve malicious functionality?
- … are there **“general" constraints**?

[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Let’s Analyze What Happened




**Optimization** | **minimize**$_\delta$ $\cancel{||\delta||_p} + c \cdot f(x+\delta)$

**Constraints**

- Is it realistic/plausible?
- Does it crash?
- Does it preserve malicious functionality?
- … are there **“general" constraints**?

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Inverse Feature-Mapping Problem

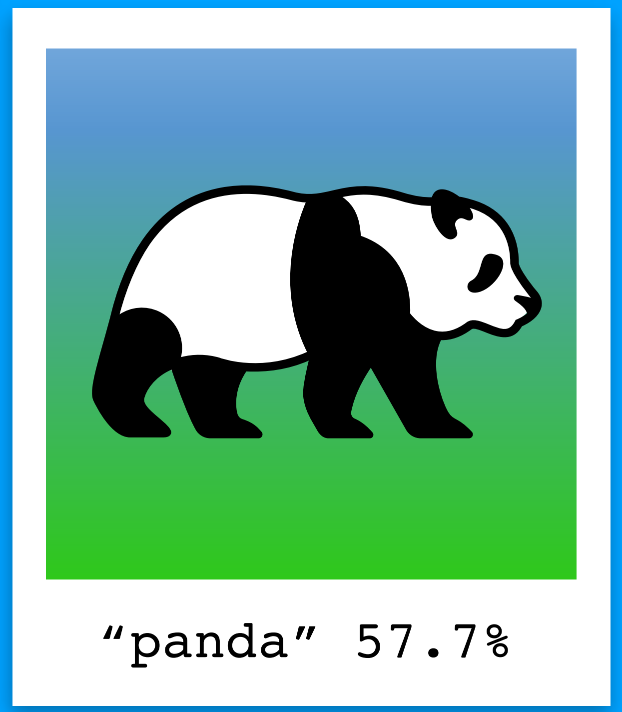
Problem Space

Feature Space

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Inverse Feature-Mapping Problem

**Problem Space**
Images





**Feature Space**
Vectors

$\xrightarrow{\varphi} x$

| | | | | | |
|---|---|---|---|---|---|
| 1 | 0 | 0.5 | 1 | … | 1 |

FS attack

$x + \delta$

| | | | | | |
|---|---|---|---|---|---|
| 0.2 | 0.1 | 0 | 0 | … | 0.3 |

$\approx \varphi^{-1}$

**The feature mapping $\varphi$ is <u>differentiable</u>**
— you can backpropagate to input

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

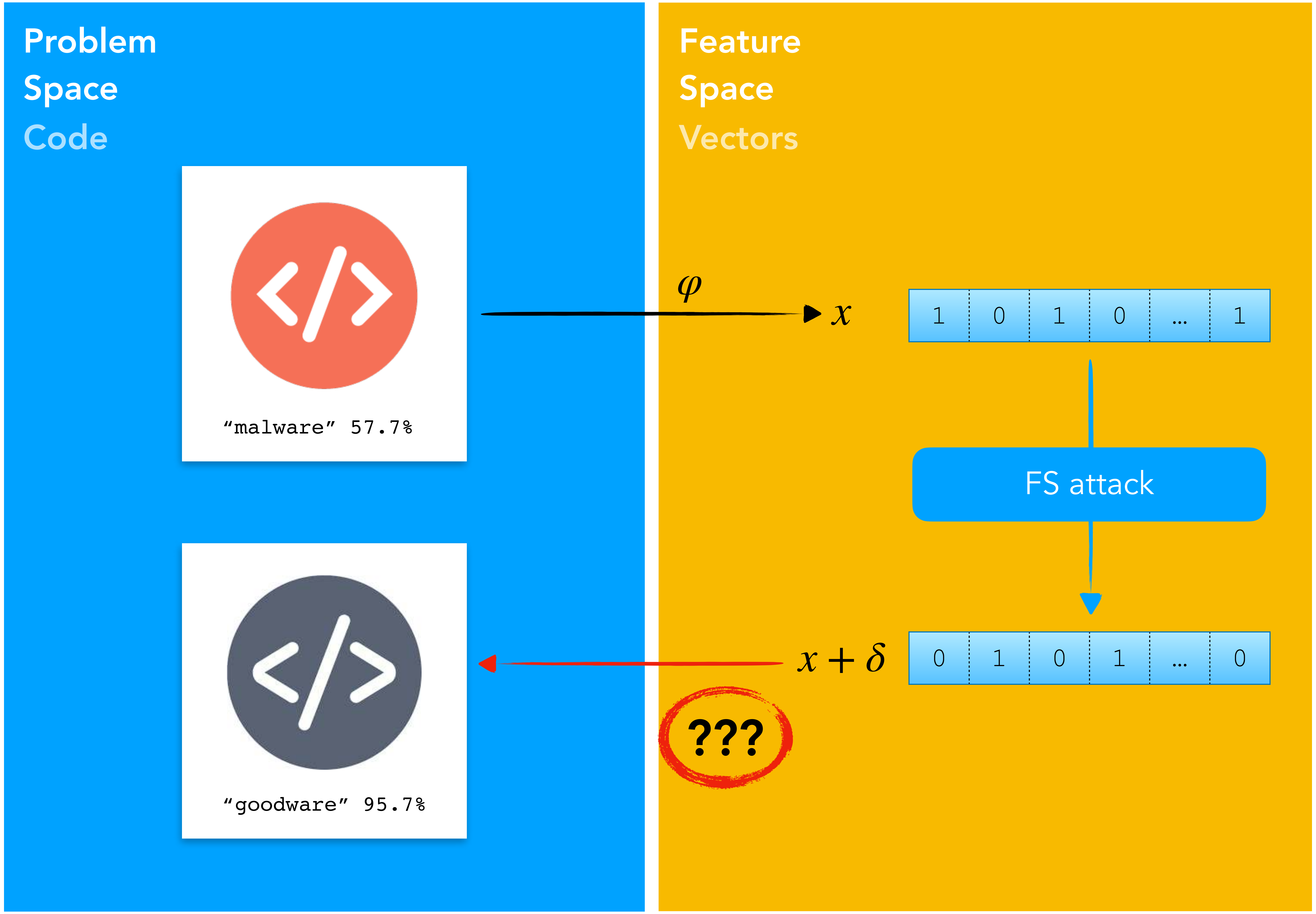
# Inverse Feature-Mapping Problem

**Problem Space**
Code


**Feature Space**
Vectors

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Inverse Feature-Mapping Problem




**In the software domain,**
**the feature mapping $\varphi$ is**
**neither <u>invertible</u> nor <u>differentiable</u>**
— how to get back to the problem space?

**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Many Problem-Space Attack Papers

**Android Malware**

[TDSC’17, ESORICS’17, ACSAC’19]

**Windows Malware**

[RAID’18, EUSIPCO’18]

**PDF Malware**

[ECML-PKDD’13, NDSS’16]

**Network Traffic**

[NCA’18, NCA’19]

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Many Problem-Space Attack Papers

**Android Malware**

[TDSC'17, ESORICS'17, ACSAC'19]

**Windows Malware**

[RAID'18, EUSIPCO'18]

**PDF Malware**

[ECML-PKDD'13, NDSS'16]

**Network Traffic**

[NCA'18, NCA'19]

**What is the State of the Art?**
**How to compare them?**

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Outline

**Formalization**

- Problem-space attacks
- Relationships
- Actionable points

**Android Problem-Space Attack**

- End-to-end adversarial malware generation at scale
- Feasible to evade feature-space defenses

**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Outline

**Formalization**

- Problem-space attacks
- Relationships
- Actionable points

**Android Problem-Space Attack**

- End-to-end adversarial malware generation at scale
- Feasible to evade feature-space defenses

Evasion Attacks

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Outline

**Formalization**

- Problem-space attacks
- Relationships
- Actionable points

**Android Problem-Space Attack**

- End-to-end adversarial malware generation at scale
- Feasible to evade feature-space defenses

Evasion Attacks

Running example: Code

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Formalization

# Problem-Space Constraints

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

## Available Transformations

How can you alter problem-space objects?

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

## Available Transformations

How can you alter problem-space objects?

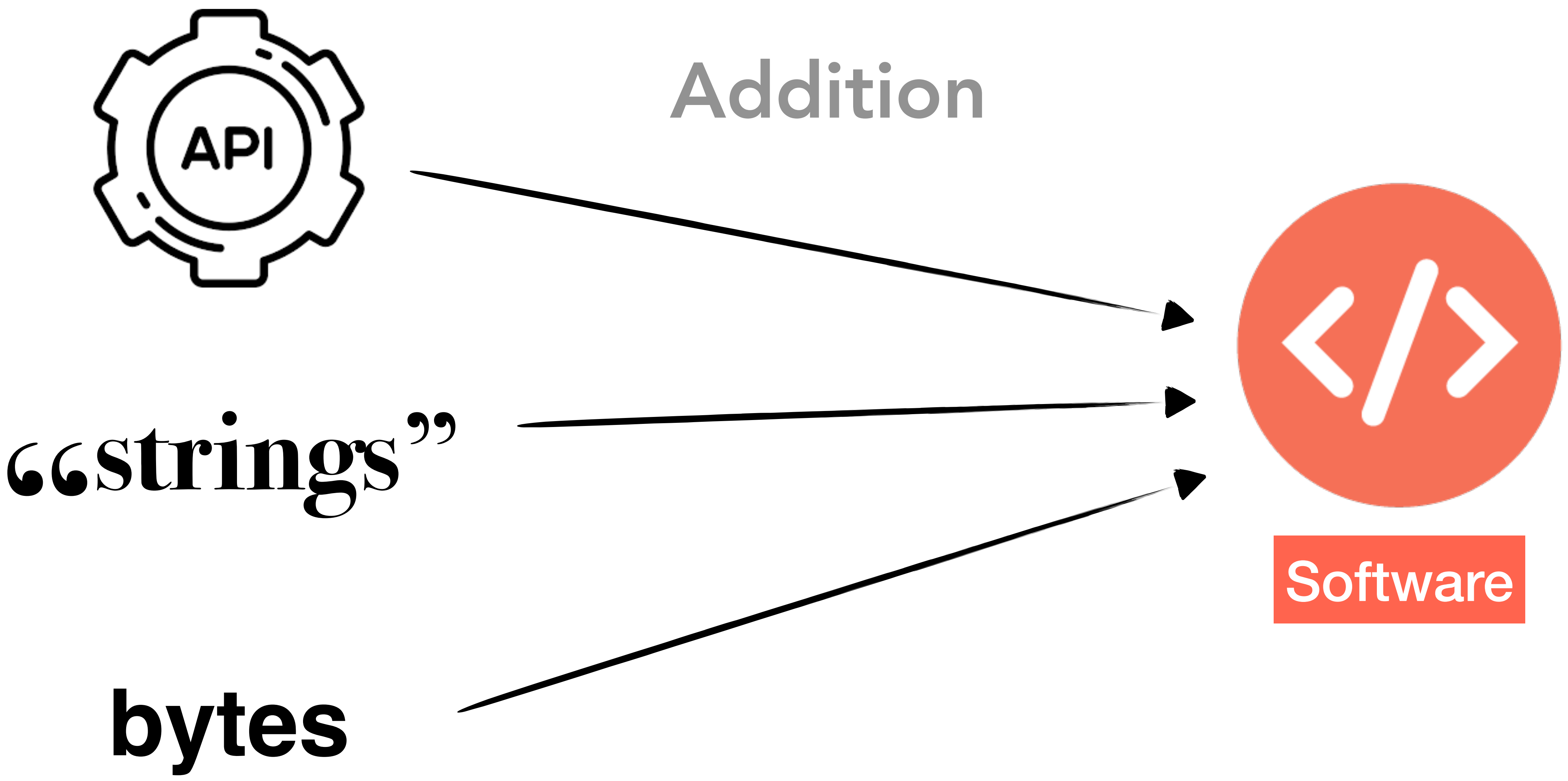
Software

**[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space**
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

## Available Transformations

How can you alter problem-space objects?

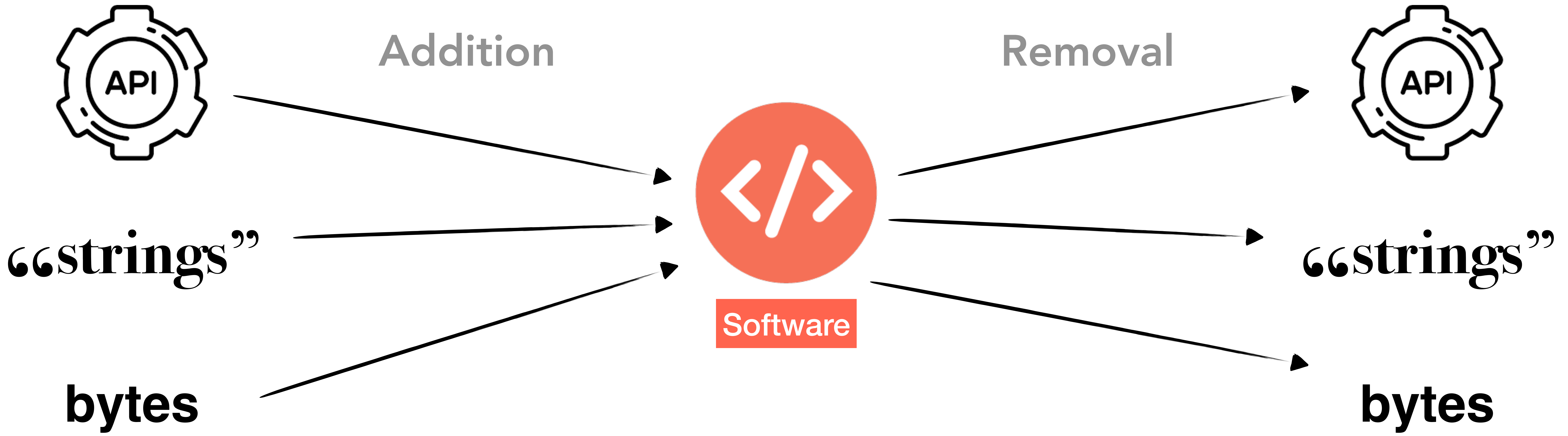



[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

## Available Transformations

How can you alter problem-space objects?

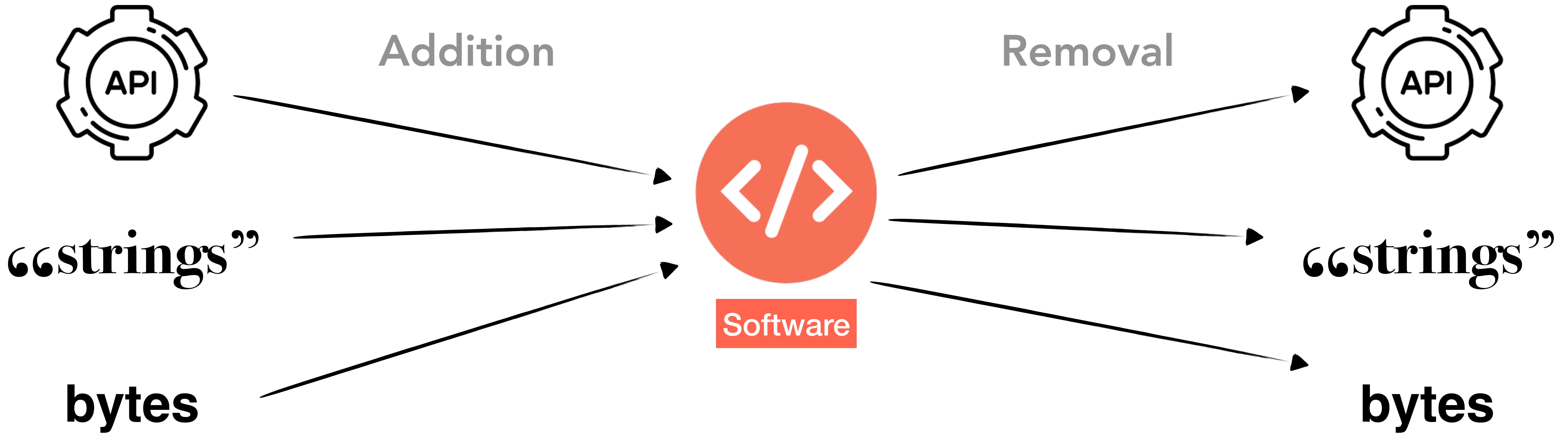



[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

## Available Transformations

How can you alter problem-space objects?

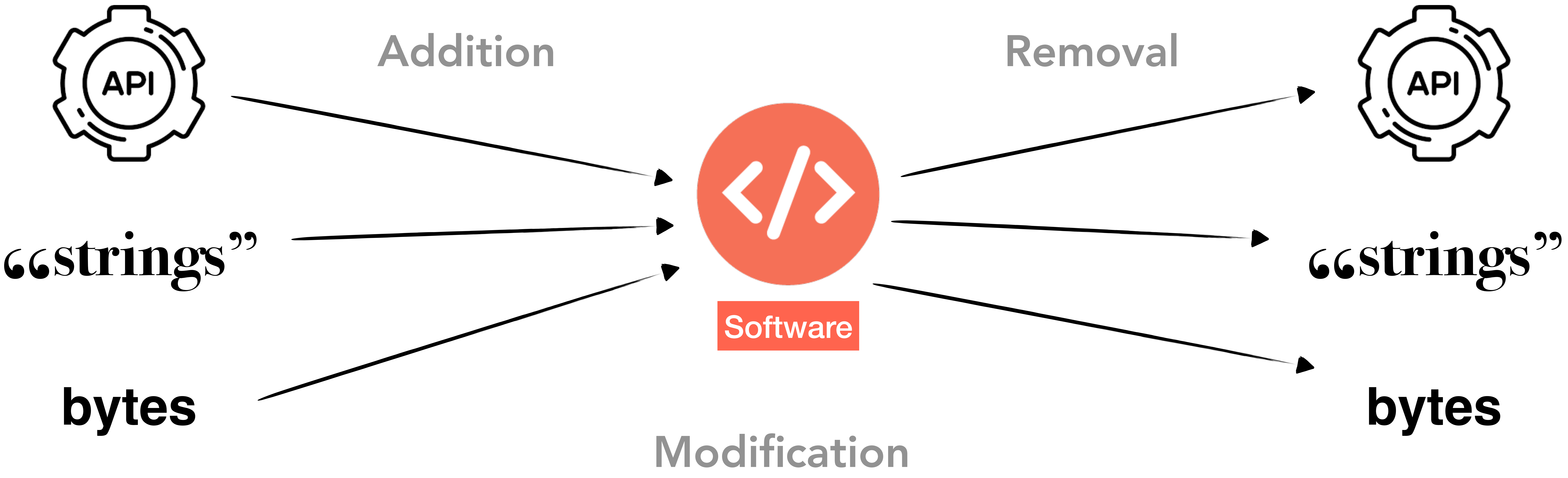



[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

## Available Transformations

How can you alter problem-space objects?




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

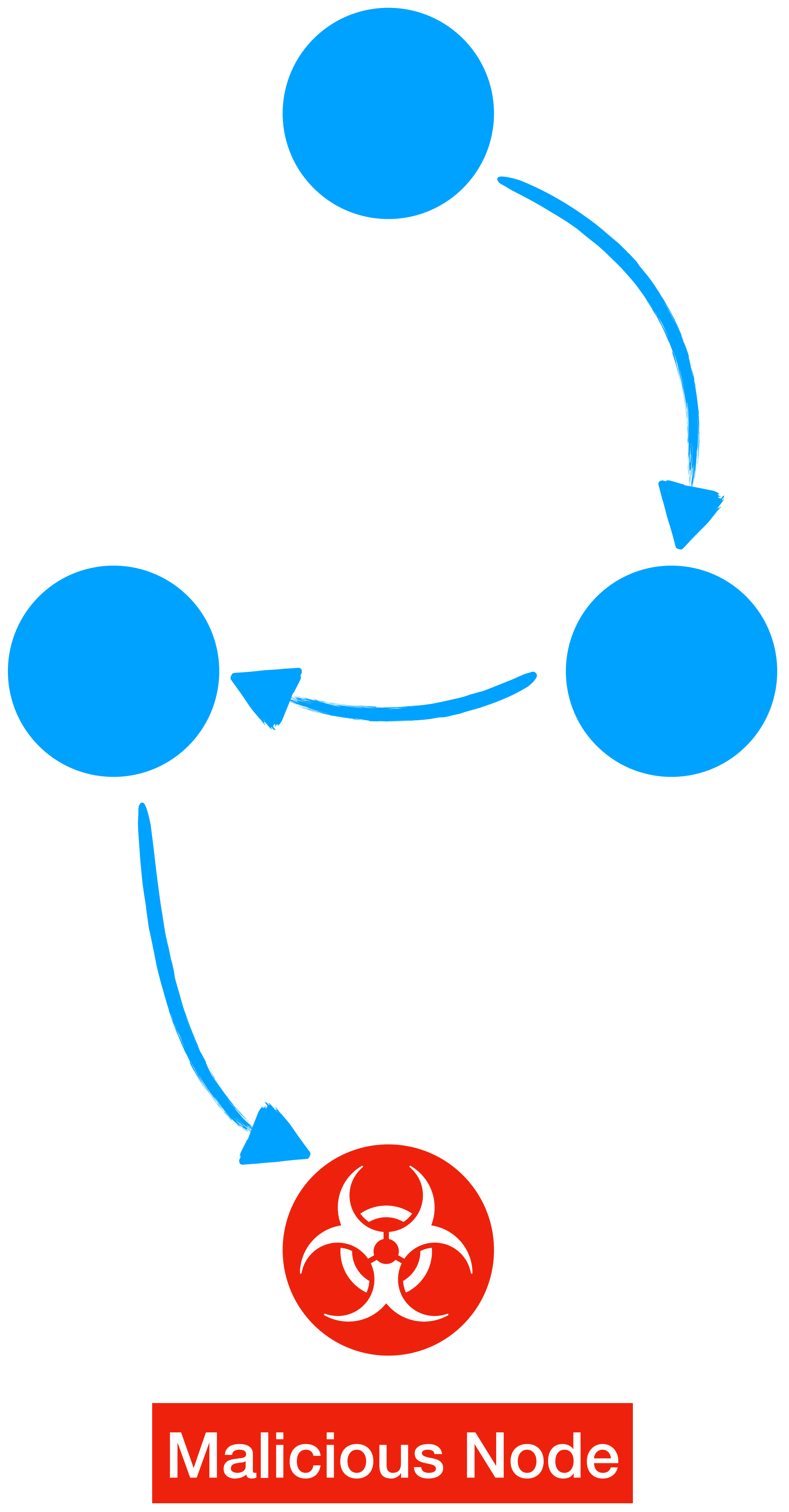
## Available Transformations

**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

- **Preserved Semantics**
- Available Transformations

Which semantics do you preserve? How?
Which automatic tests can verify it?

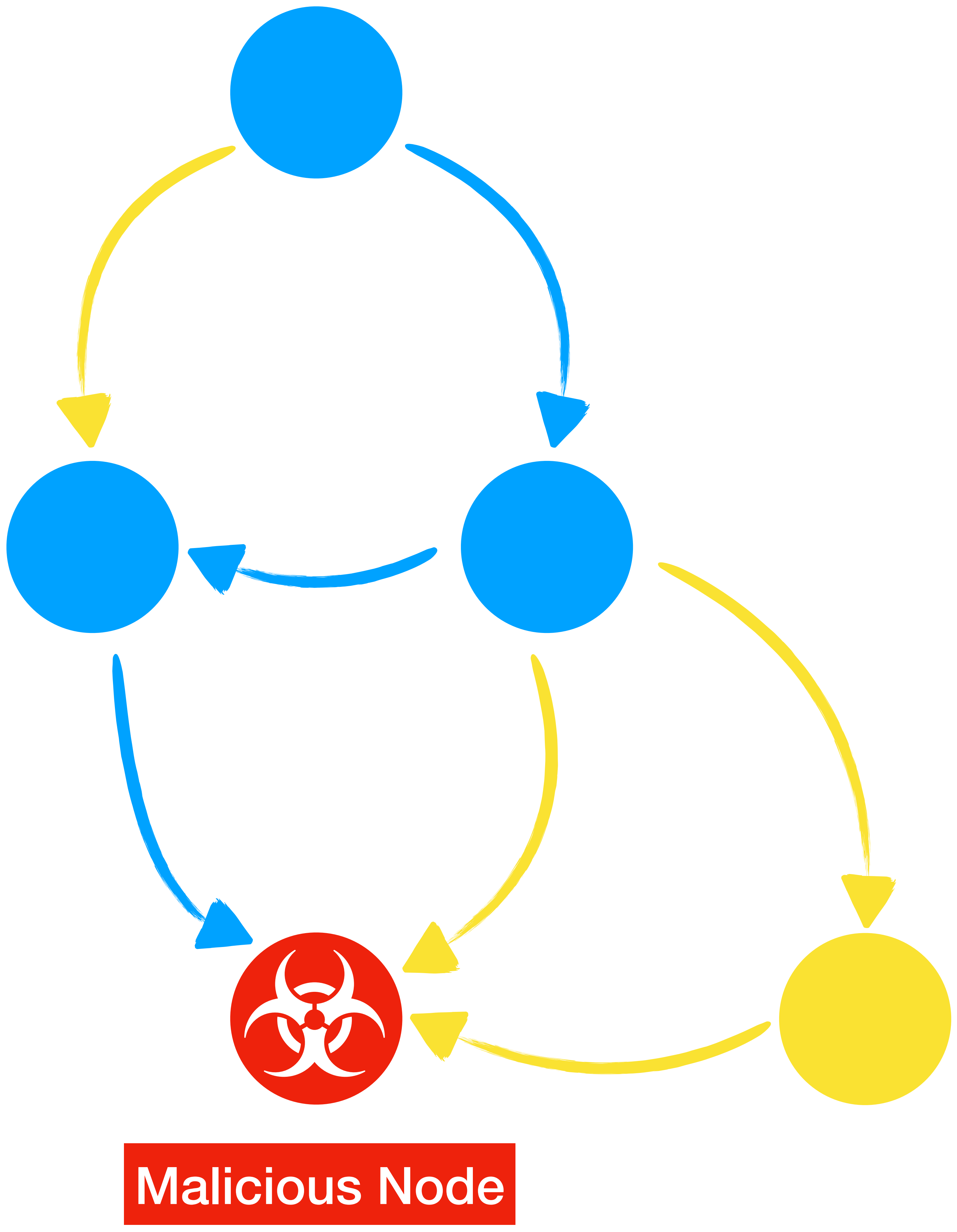



**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

- **Preserved Semantics**
- Available Transformations

Which semantics do you preserve? How?
Which automatic tests can verify it?




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

**Preserved Semantics**

Available Transformations

Which semantics do you preserve? How?

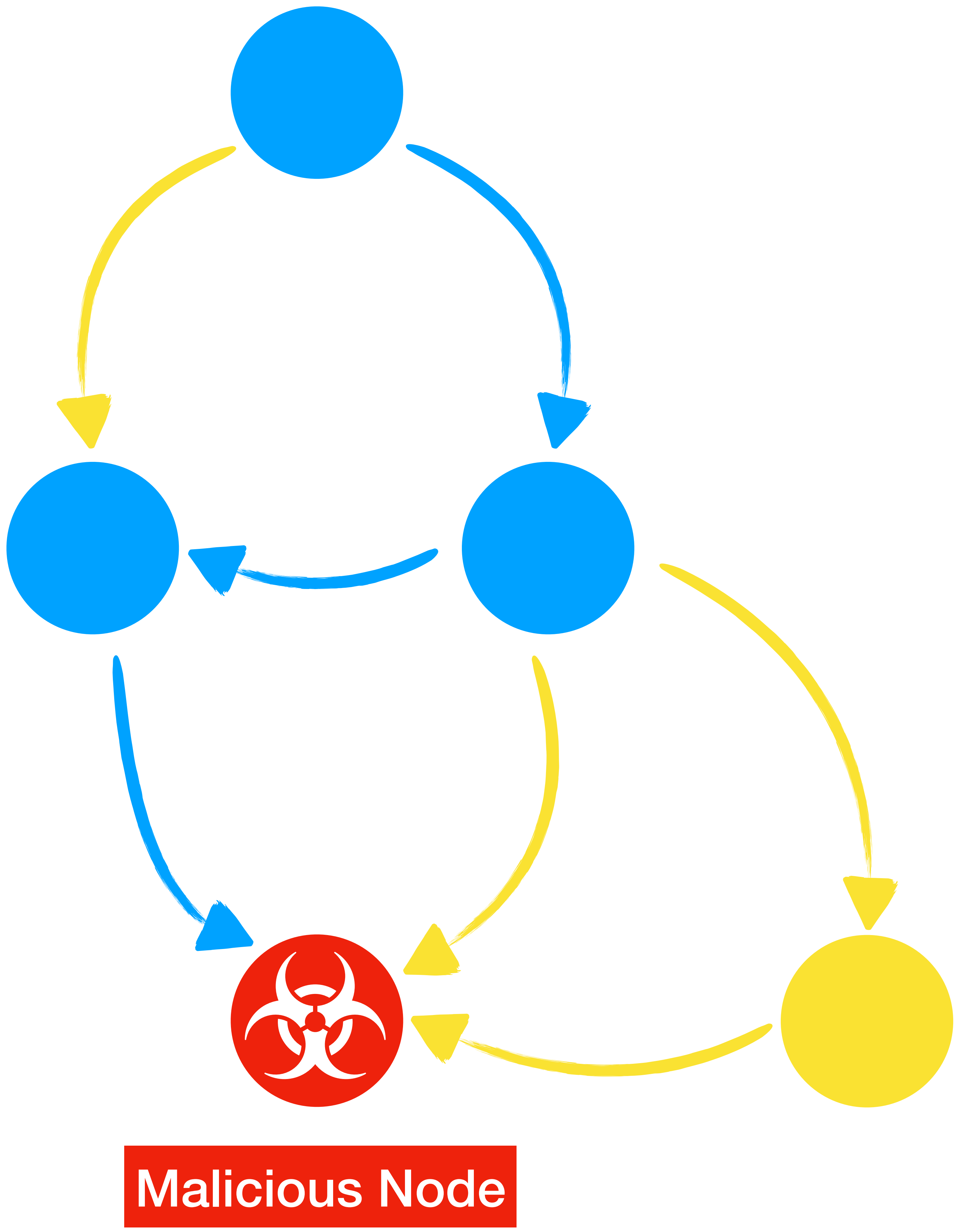
Which automatic tests can verify it?

**Test Suite**

- Does it crash?
- Does it still communicate with CnC?
- Does it still encrypt the /home/ folder?

**By Construction**

- Add no-op operations
- Ensure it is not executed at runtime




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

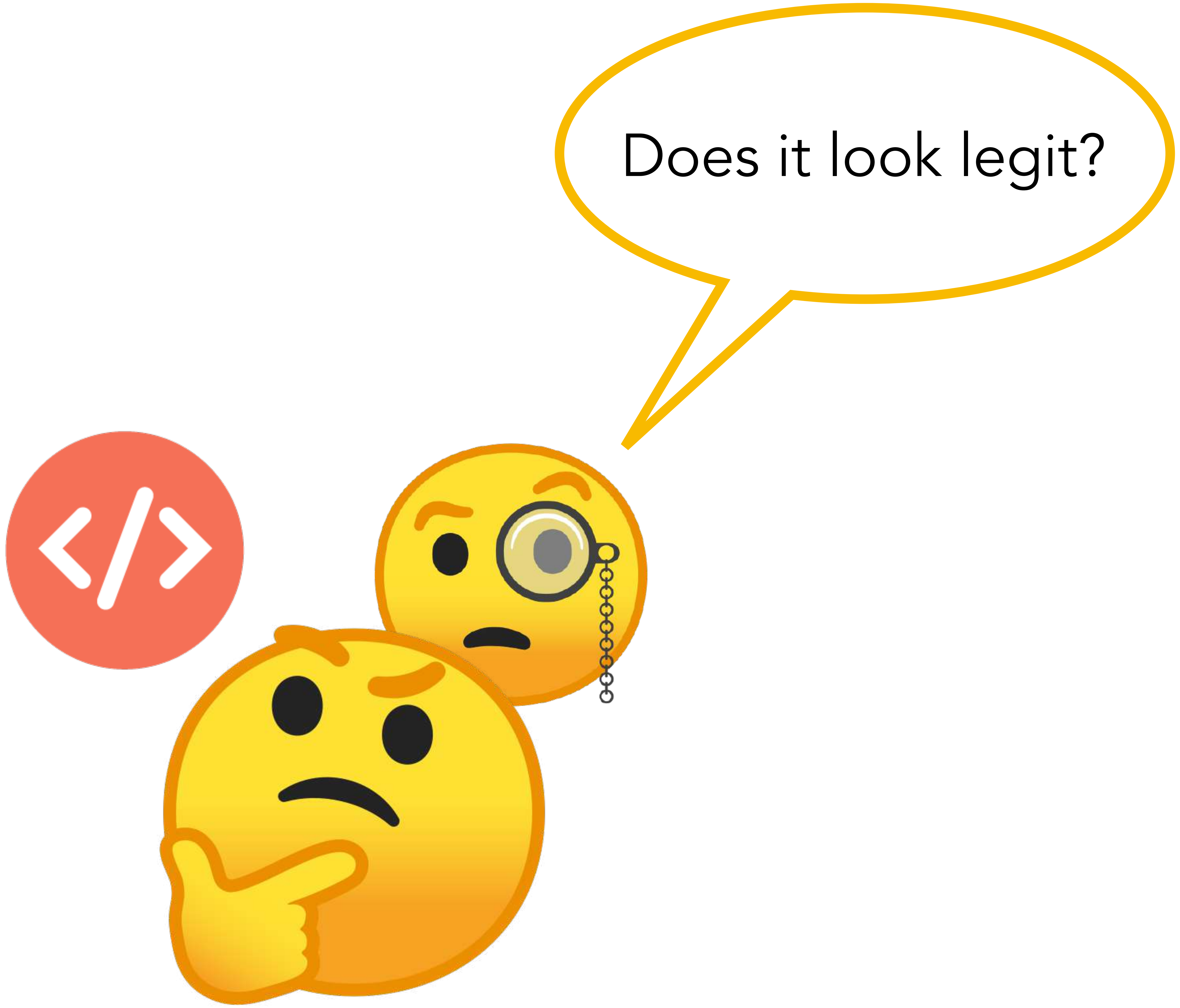
# Problem-Space Constraints

- Preserved Semantics
- Available Transformations

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

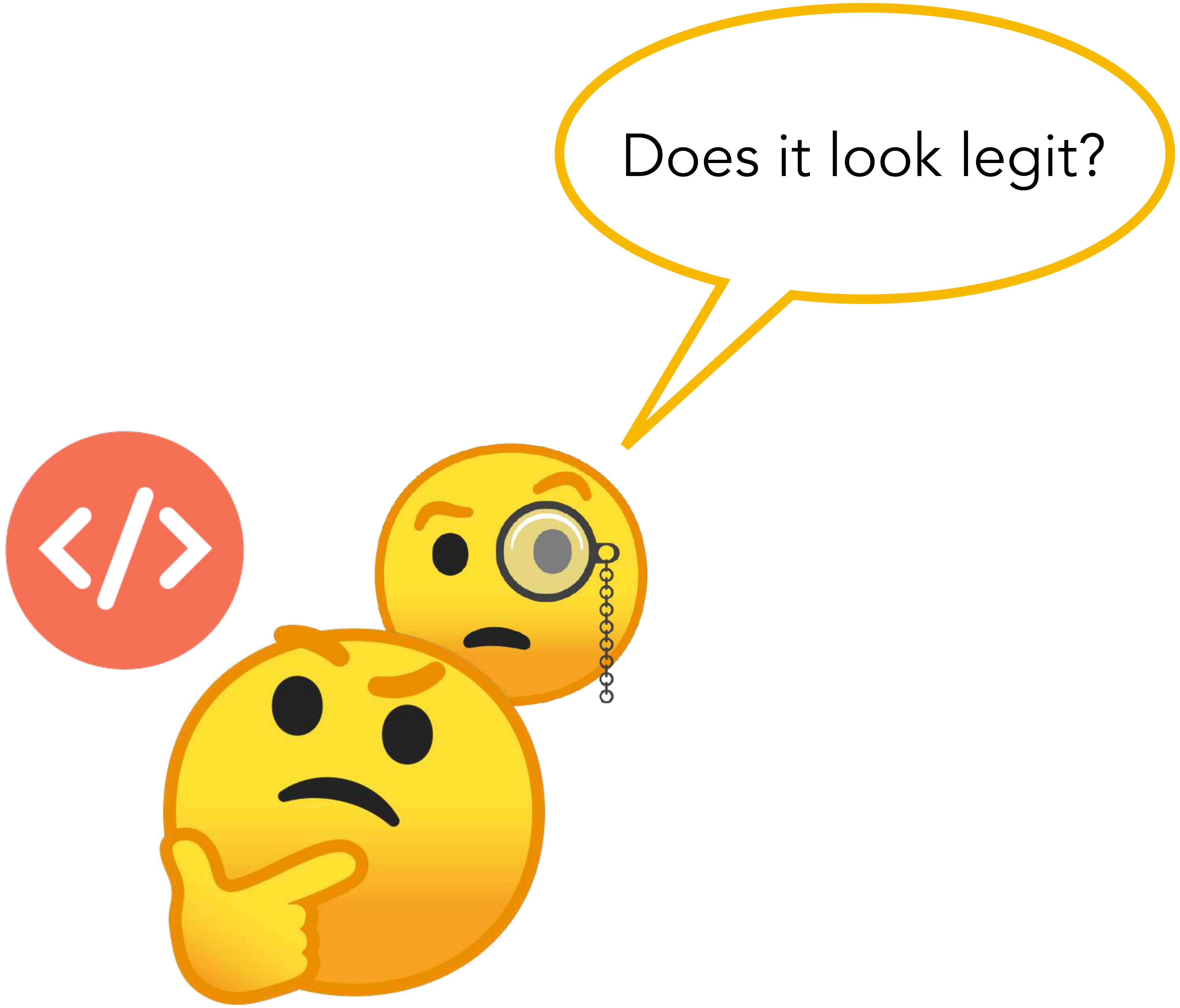
- **Plausibility**
- Preserved Semantics
- Available Transformations




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

- **Plausibility**
- Preserved Semantics
- Available Transformations




**Test Suite**

- User studies
- Automated heuristics

**By Construction**

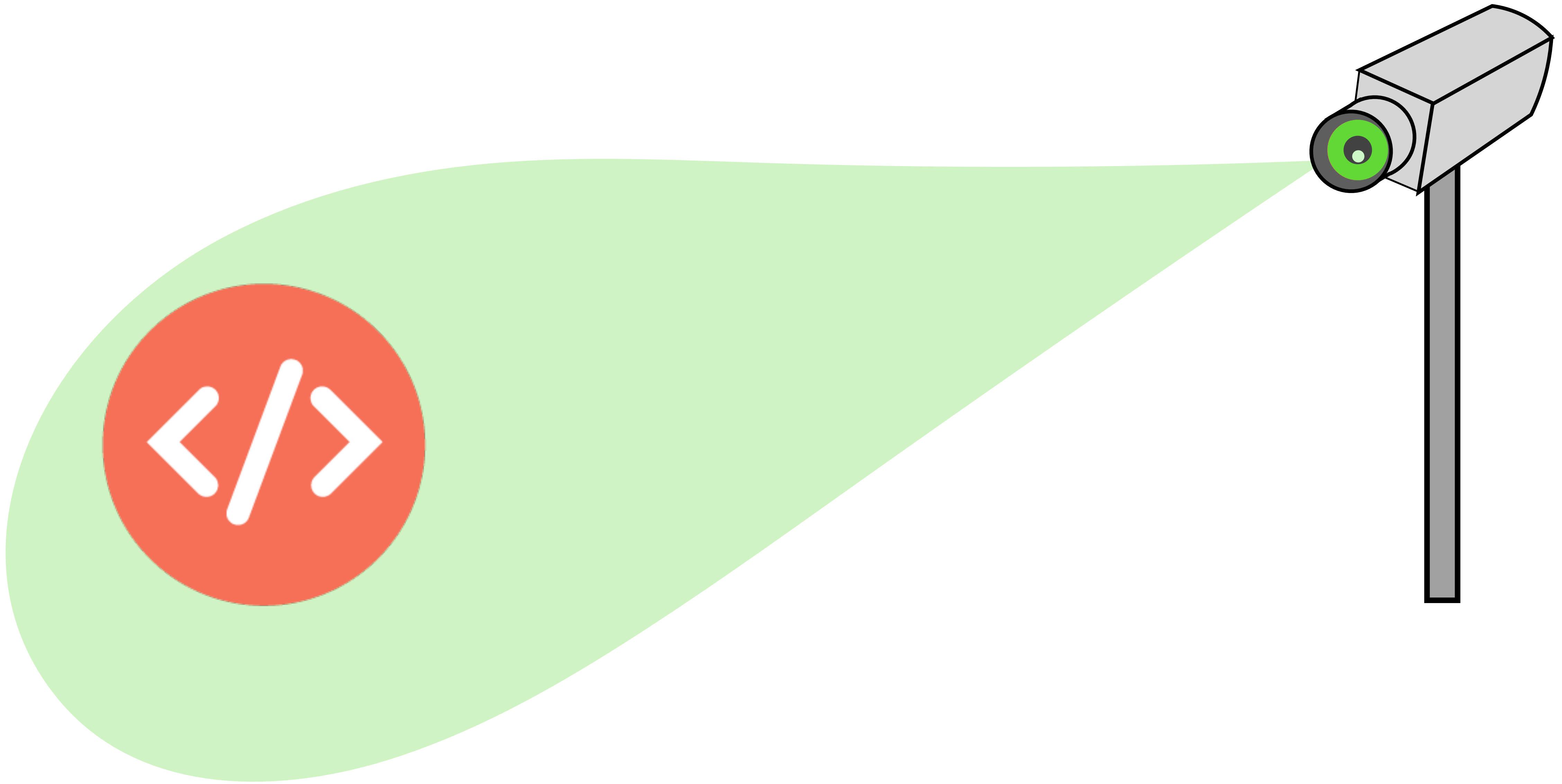
- Taking precautions during mutation

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

- **Plausibility**
- Preserved Semantics
- Available Transformations

Which preprocessing are you considering?

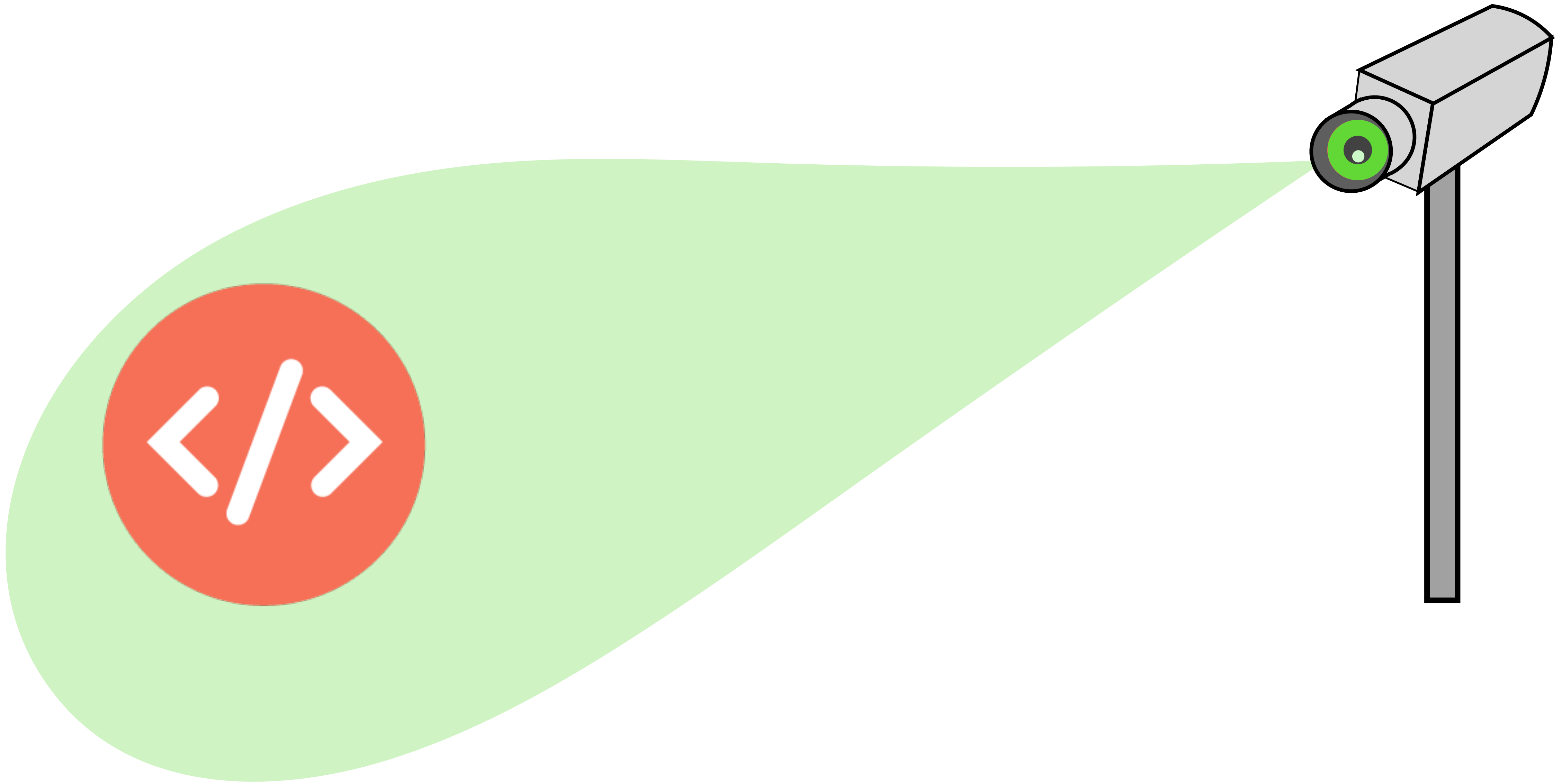



**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

- **Robustness to Preprocessing**
- Plausibility
- Preserved Semantics
- Available Transformations

Which preprocessing are you considering?

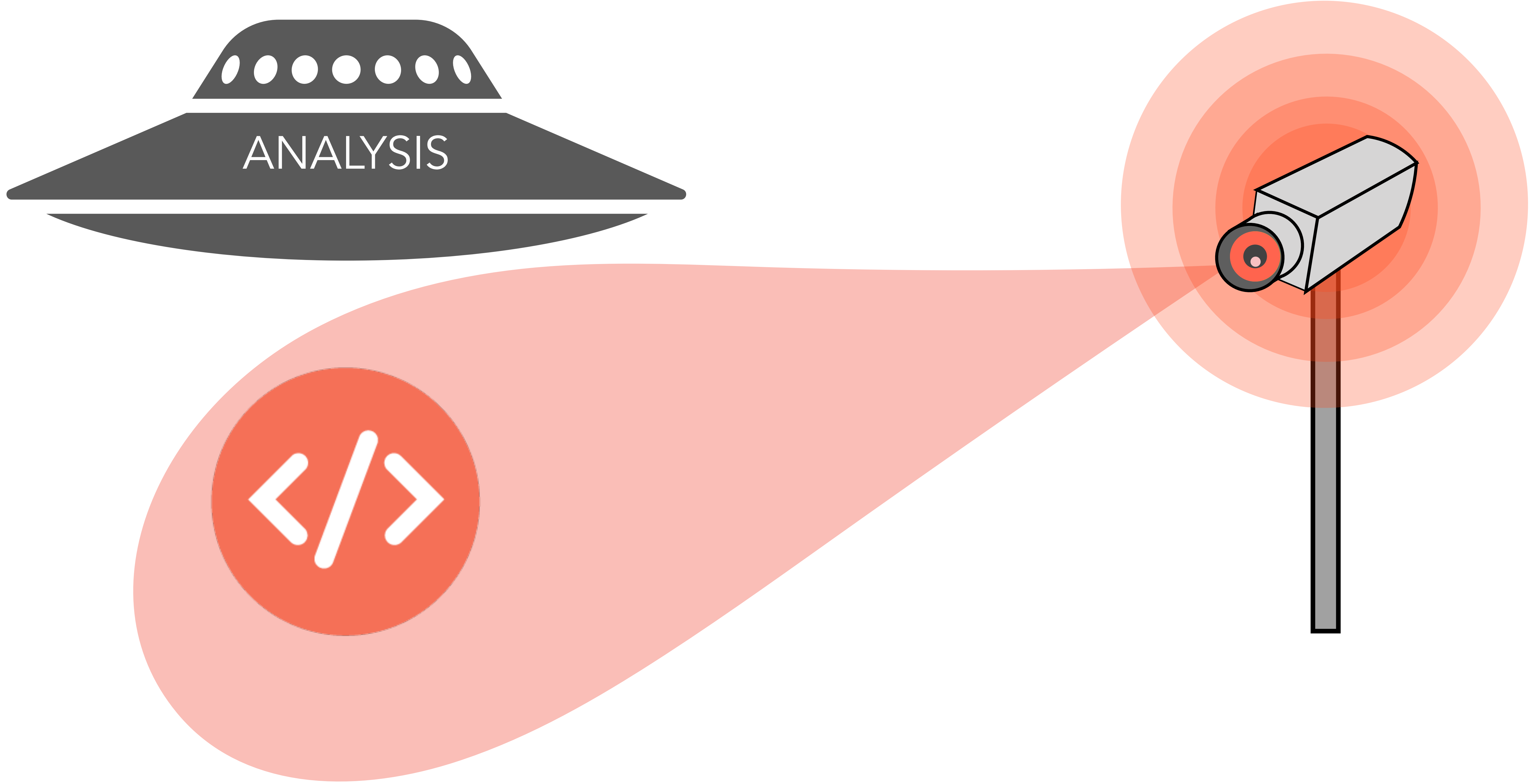


**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Problem-Space Constraints

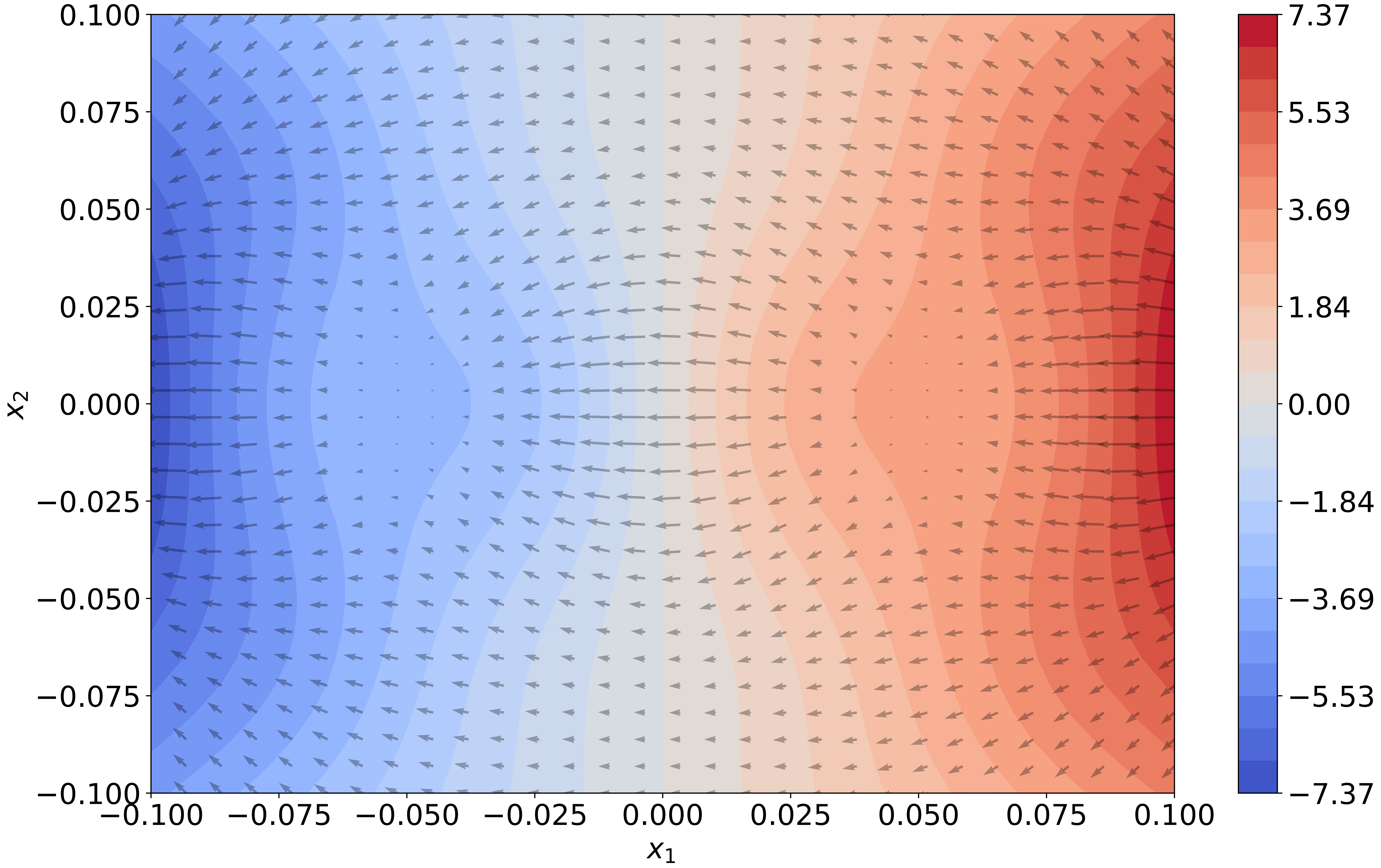
- **Robustness to Preprocessing**
- Plausibility
- Preserved Semantics
- Available Transformations

Which preprocessing are you considering?



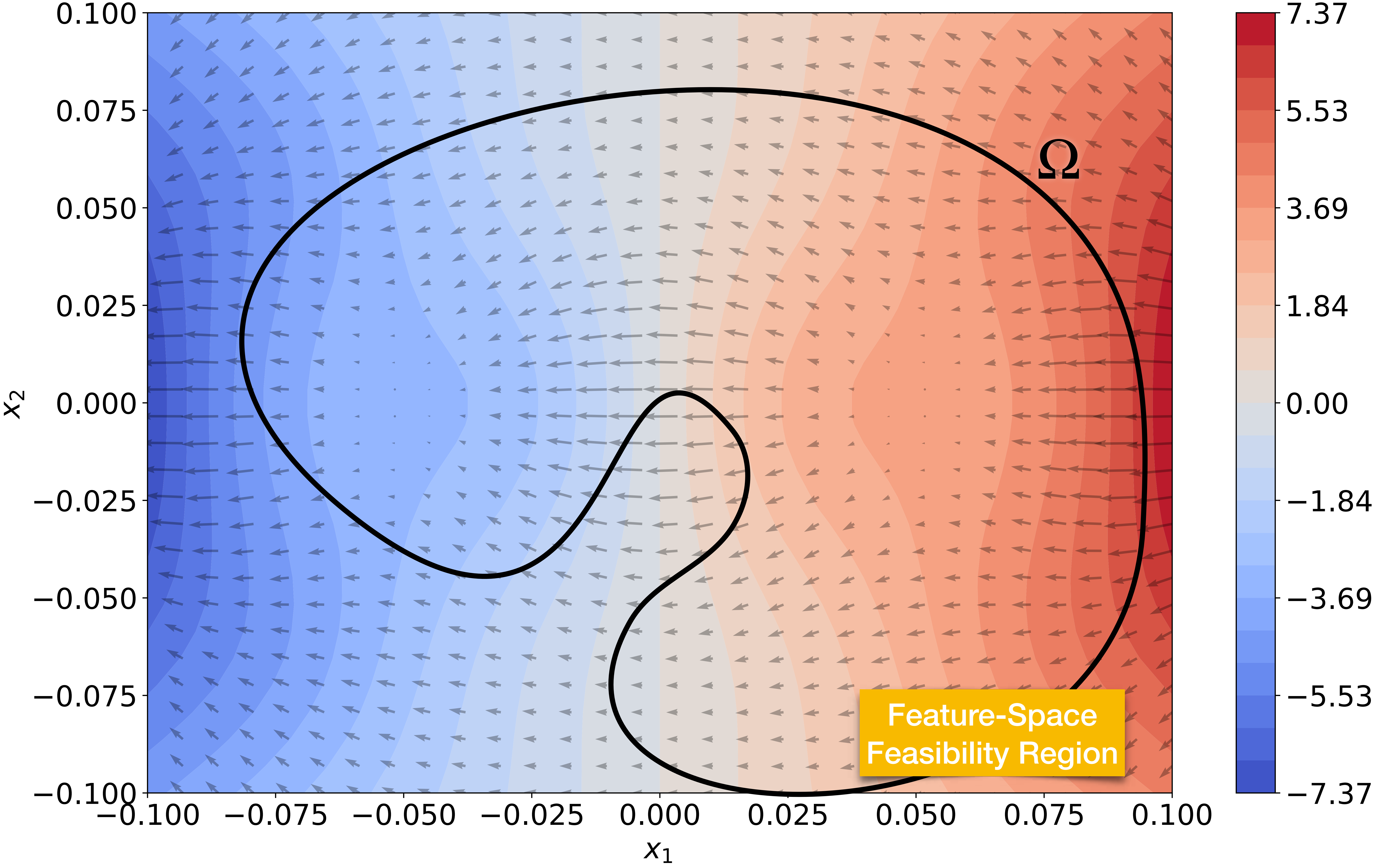

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-effect Features



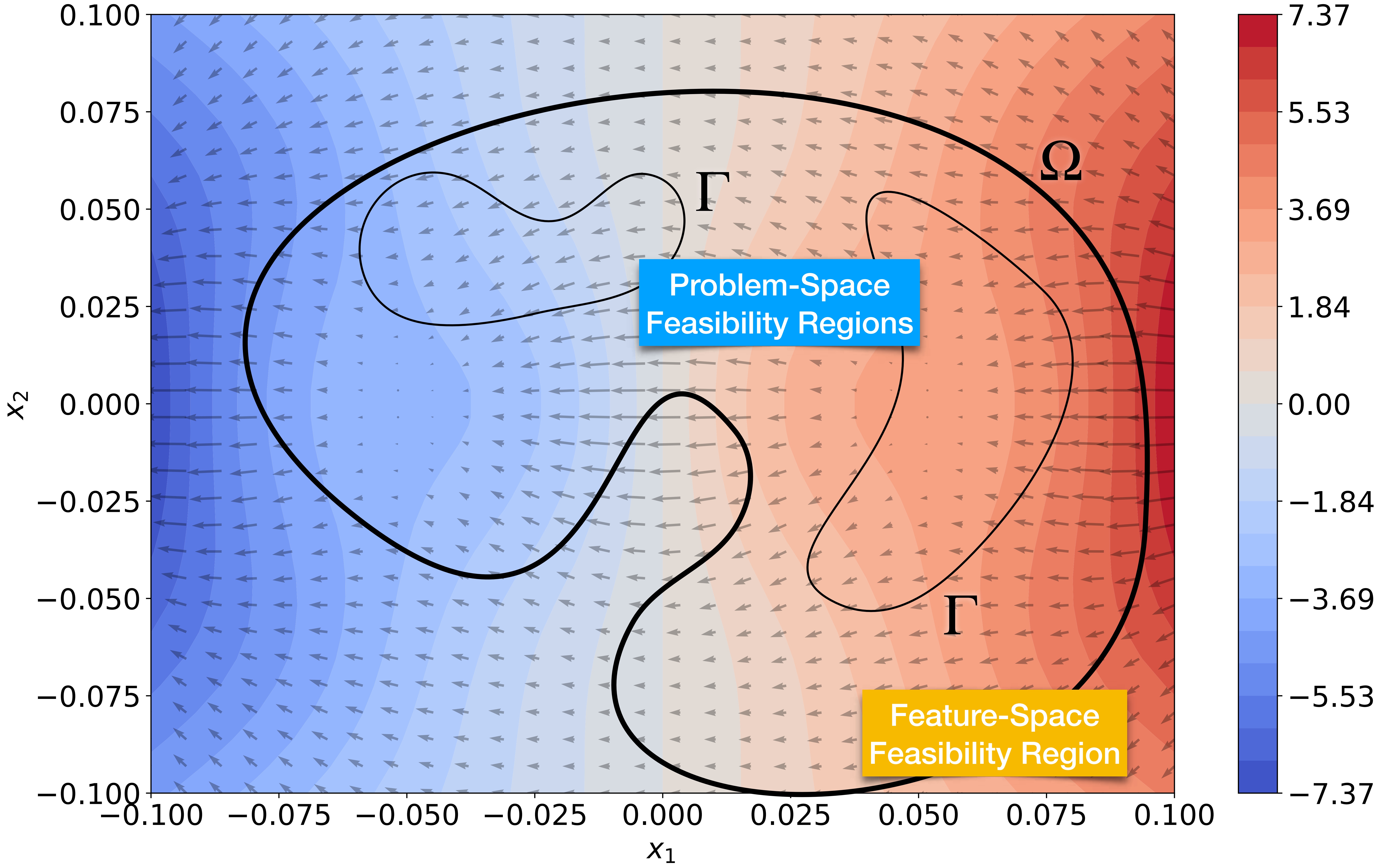

**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-effect Features



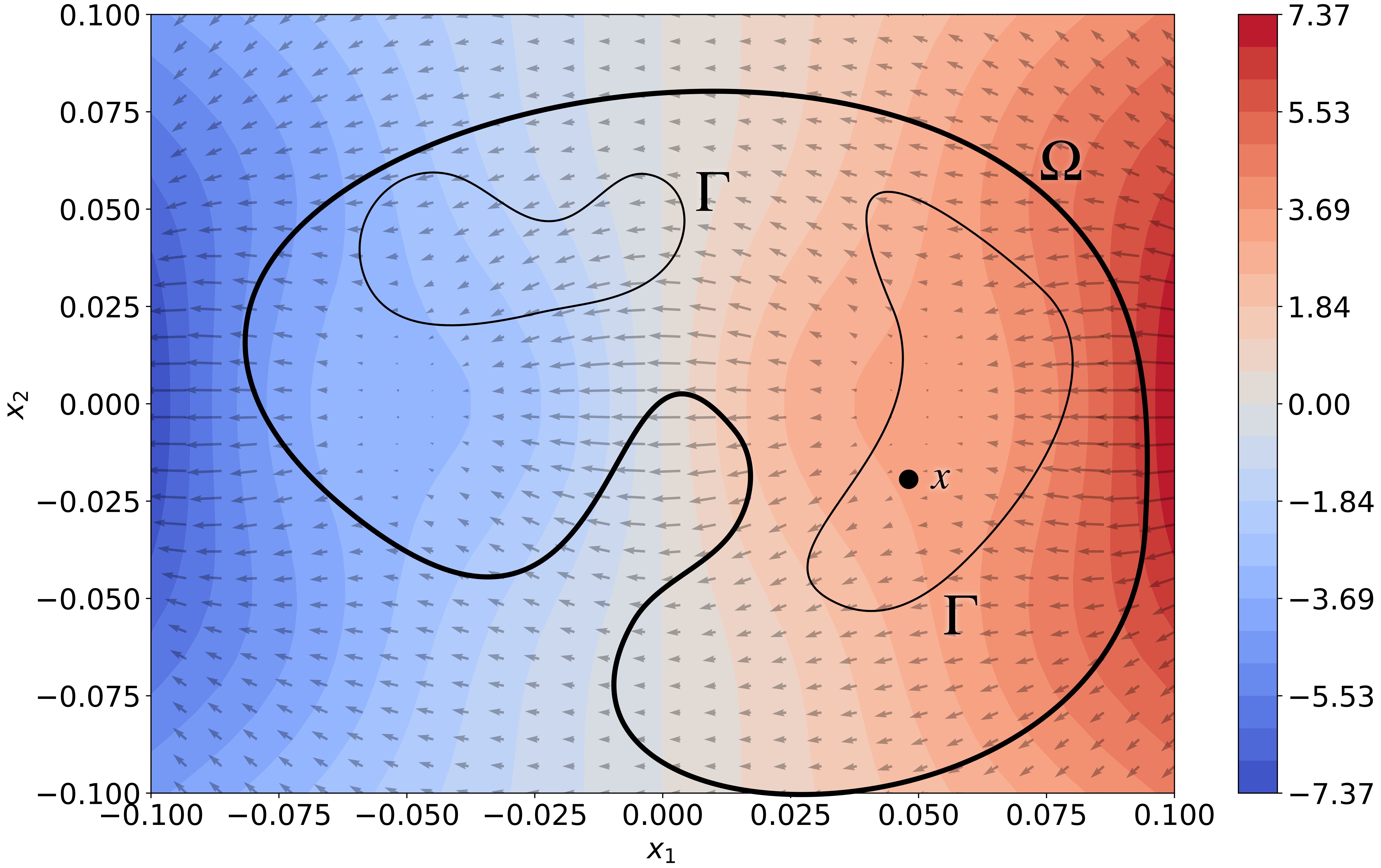

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-effect Features




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

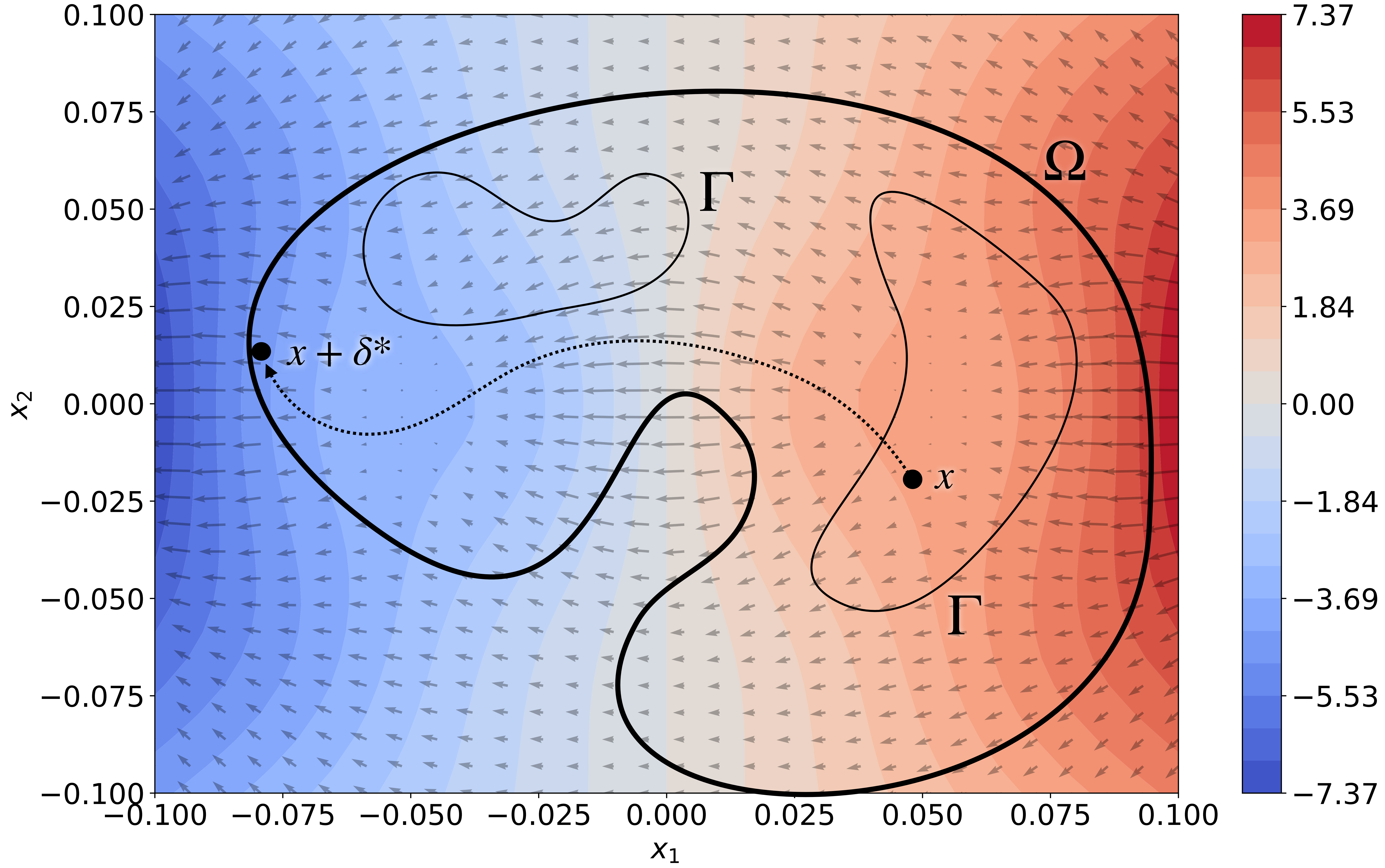
# Side-effect Features




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space

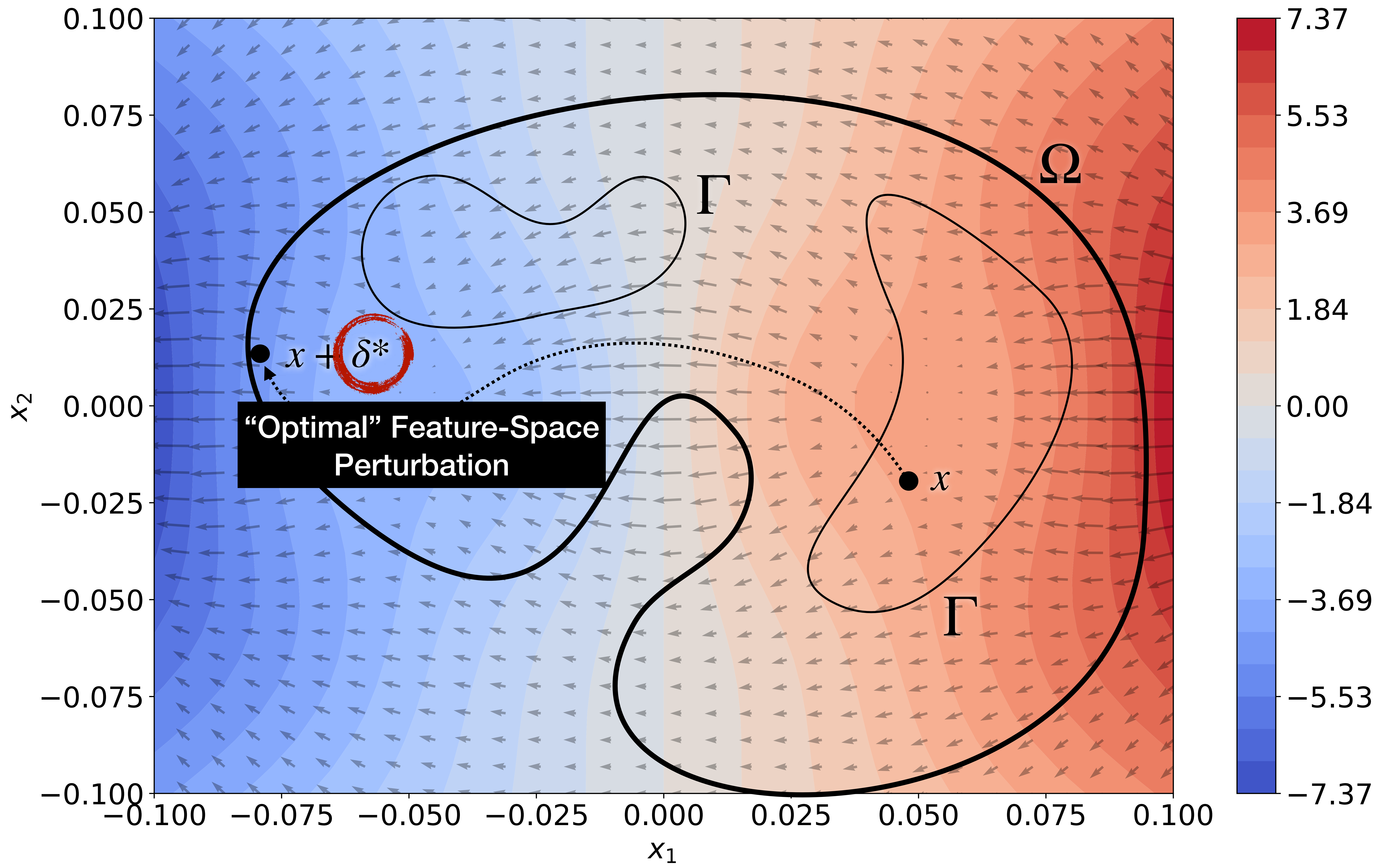
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-effect Features



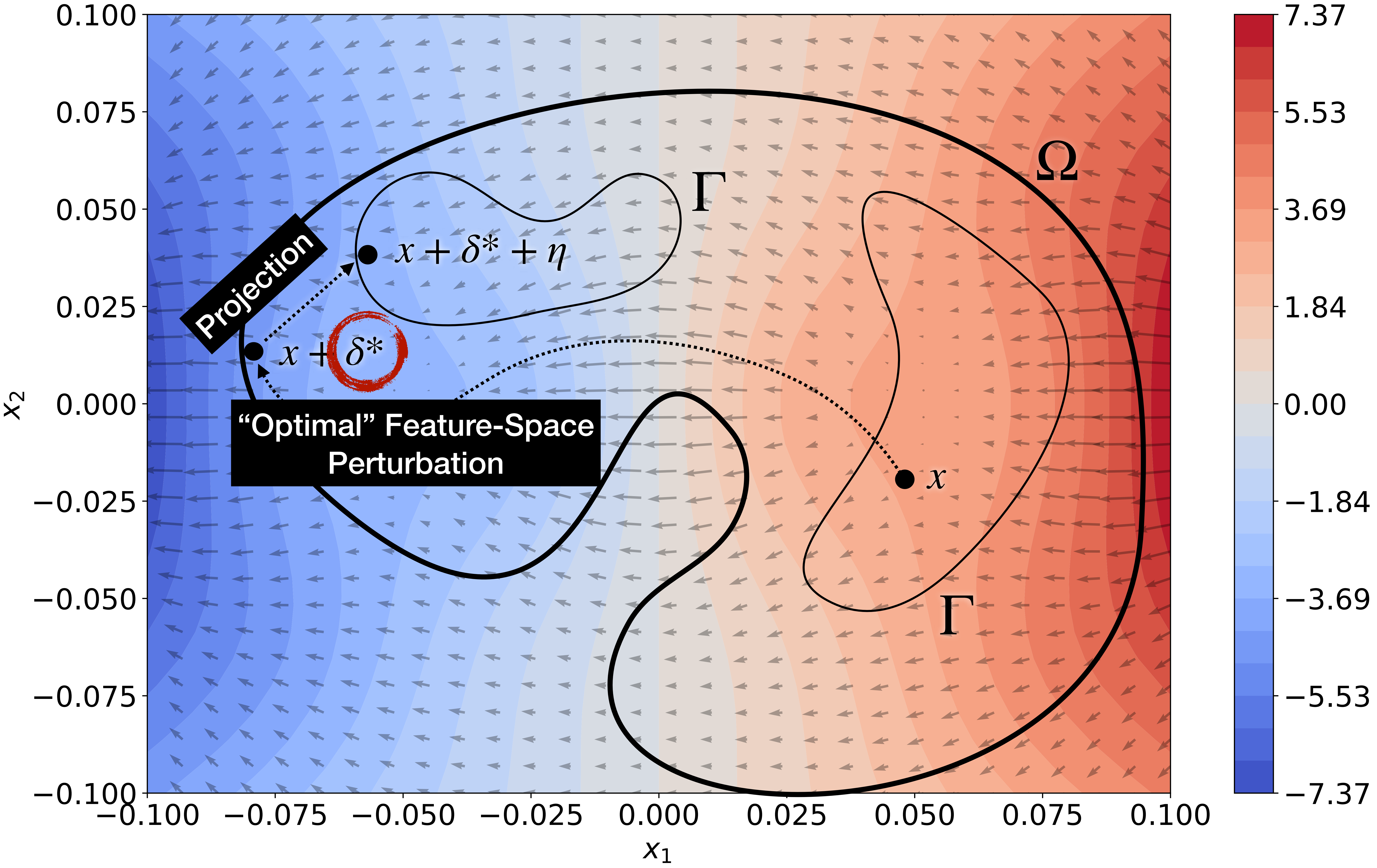

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-effect Features



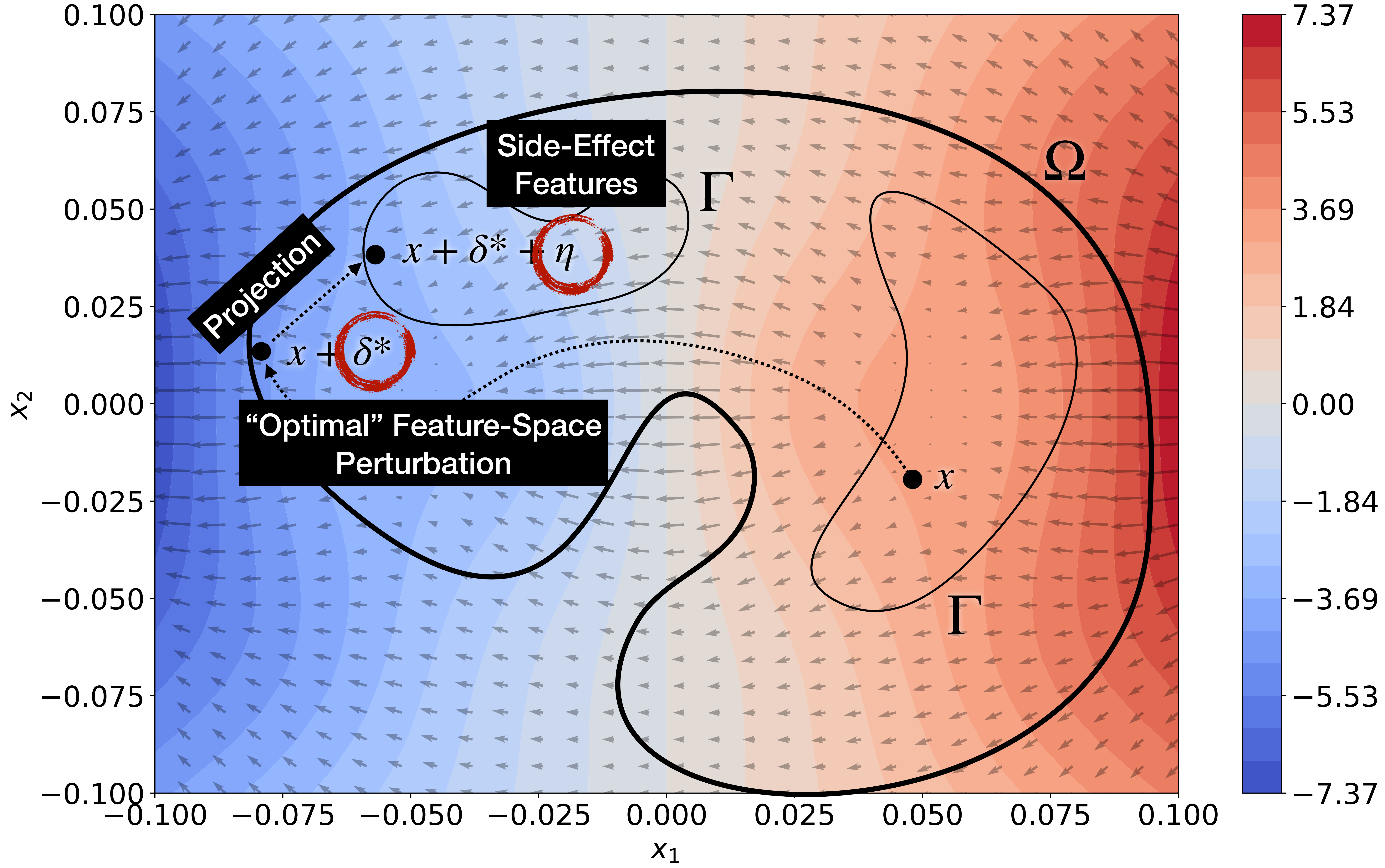

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-effect Features




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-effect Features




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Actionable Points

[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Actionable Points

Verify existence of feature-space attack

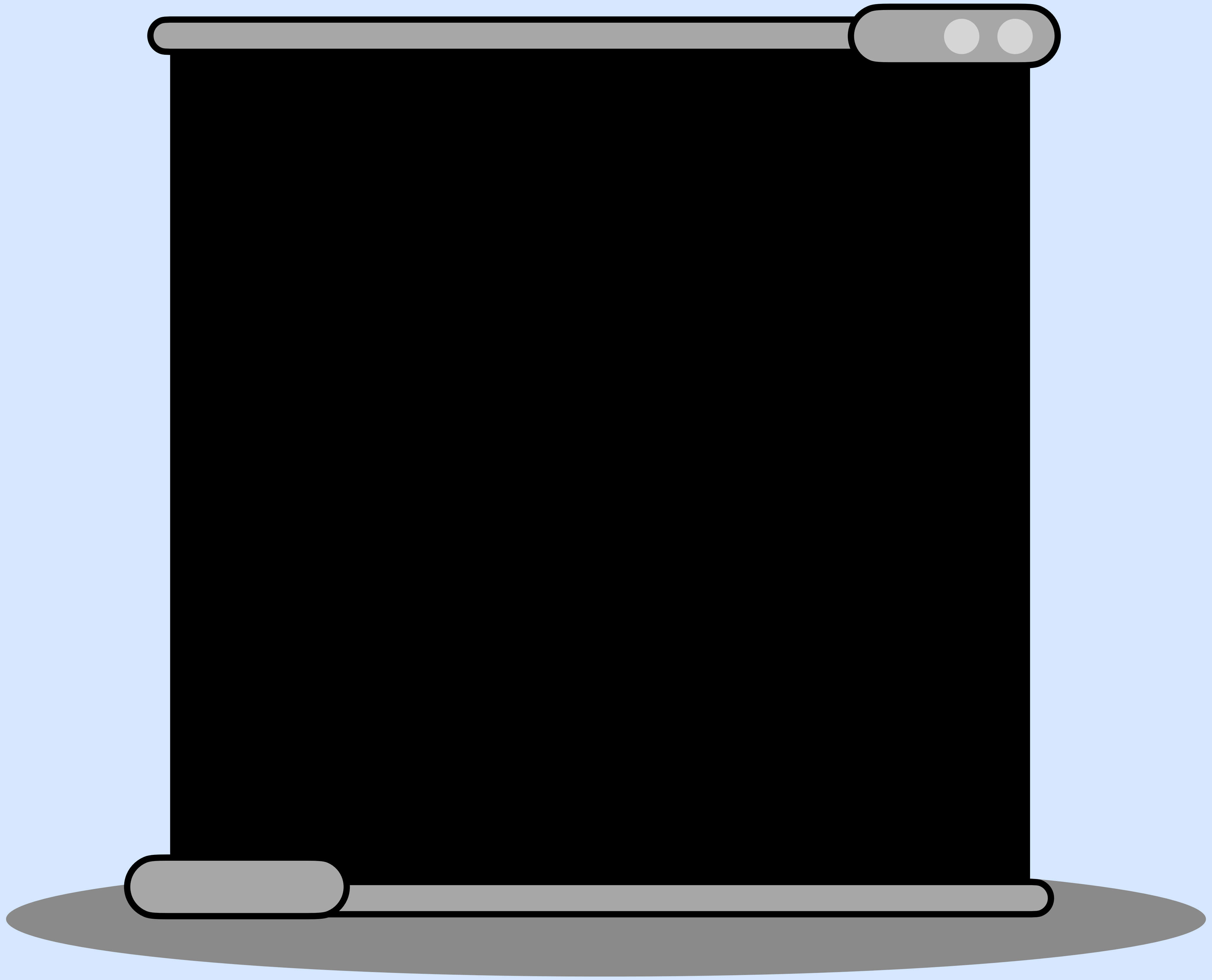
**Necessary Condition for problem-space attacks**

**$\exists$ problem-space attack $\Longrightarrow$ $\exists$ feature-space attack**

Proof 1 in paper

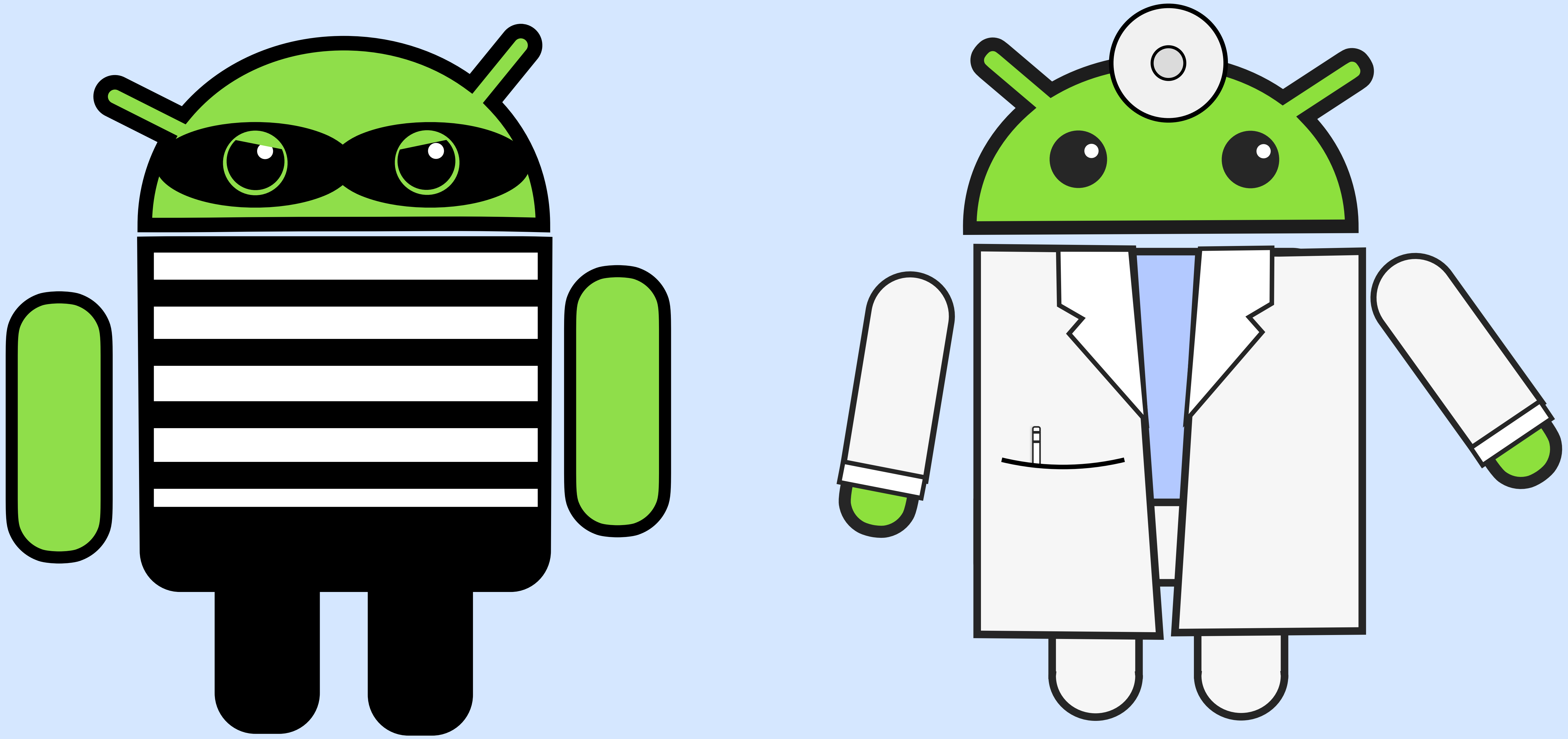
**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Our Android Attack



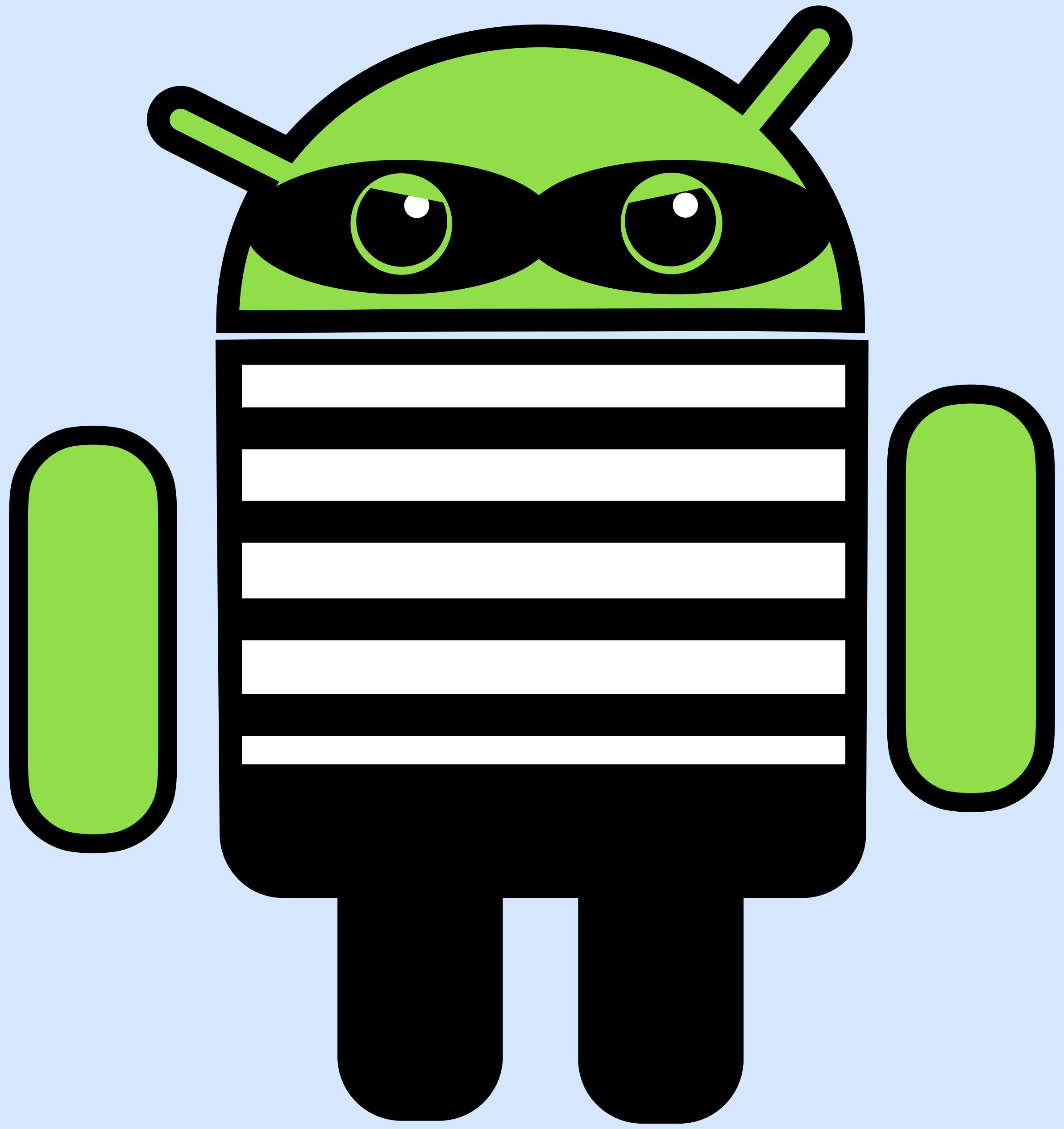
[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

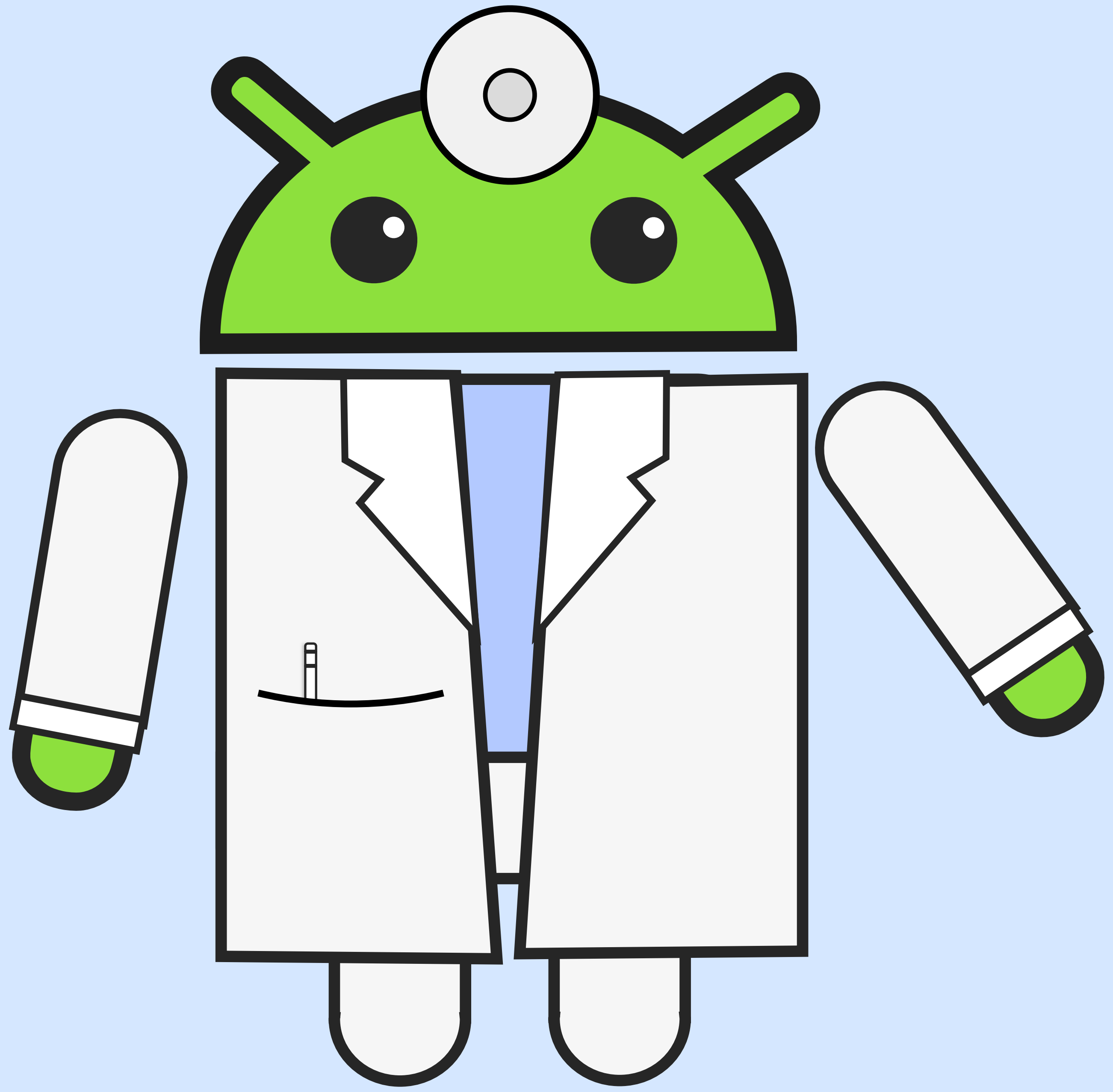
# Our Android Attack



[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Our Android Attack

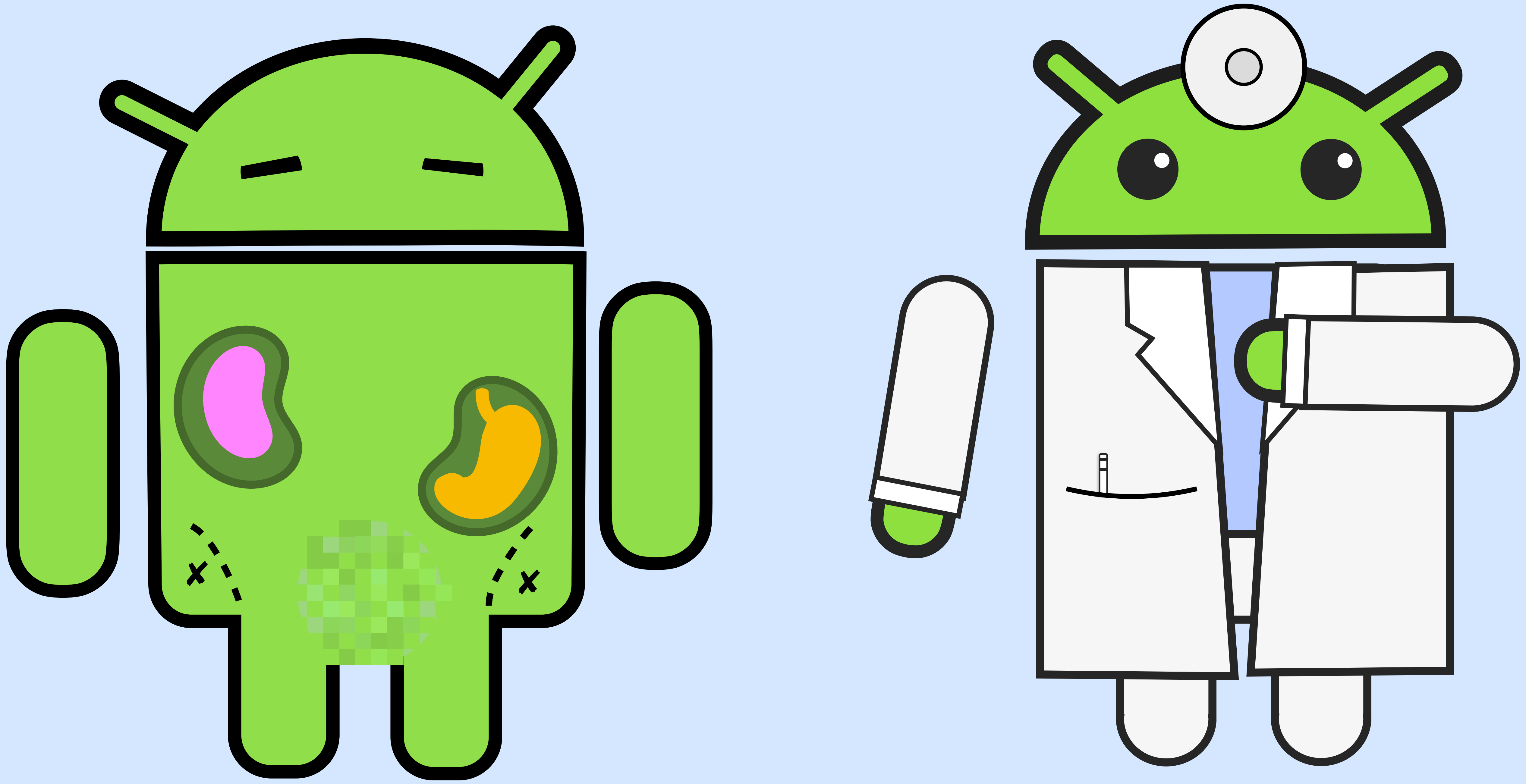




**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Our Android Attack

**Available Transformations**
Code addition through automated software transplantation.

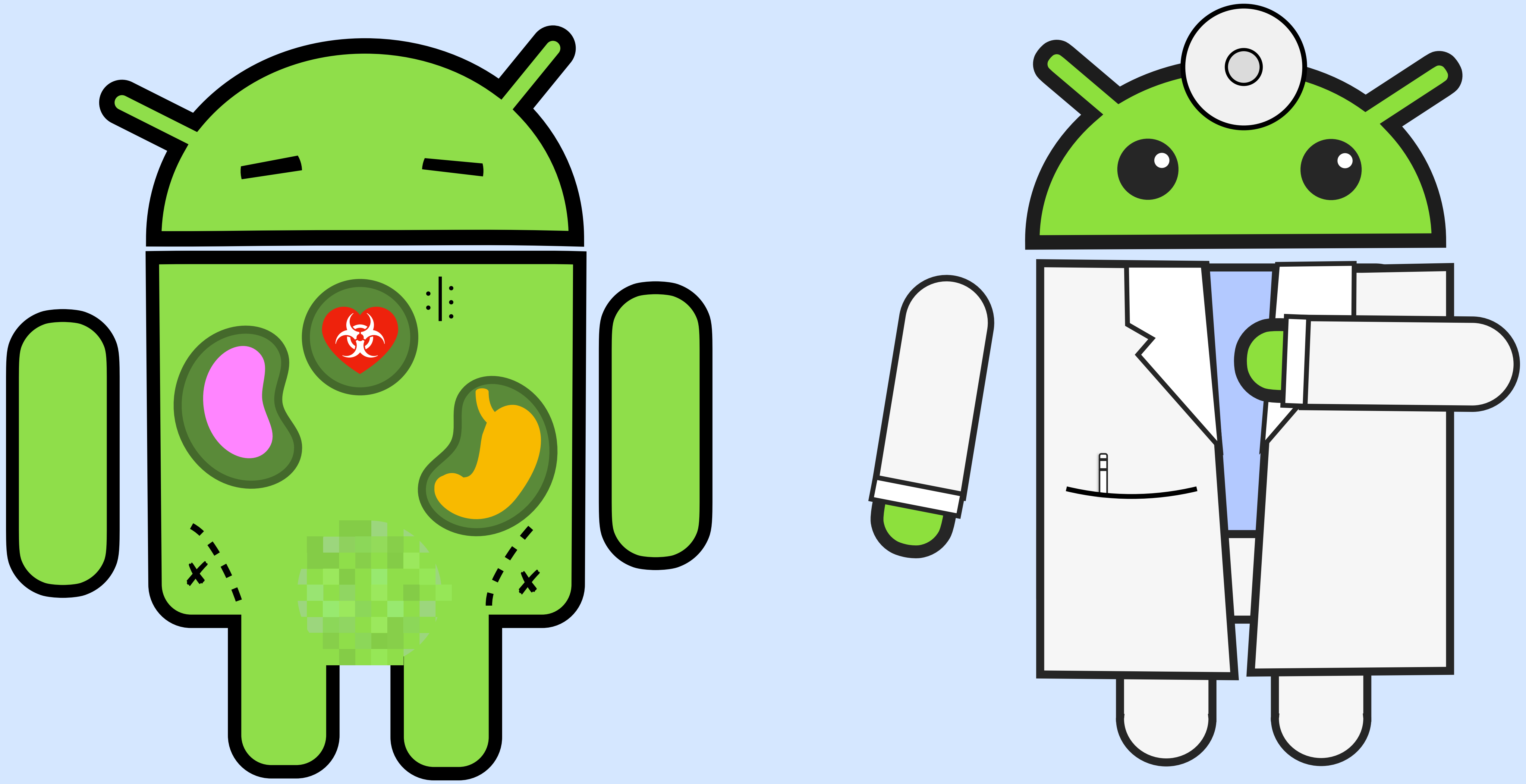


[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Our Android Attack

**Available Transformations**
Code addition through automated software transplantation.

**Preserved Semantics**
Malicious semantics preserved by construction using opaque predicates (inserted code is not executed at runtime).



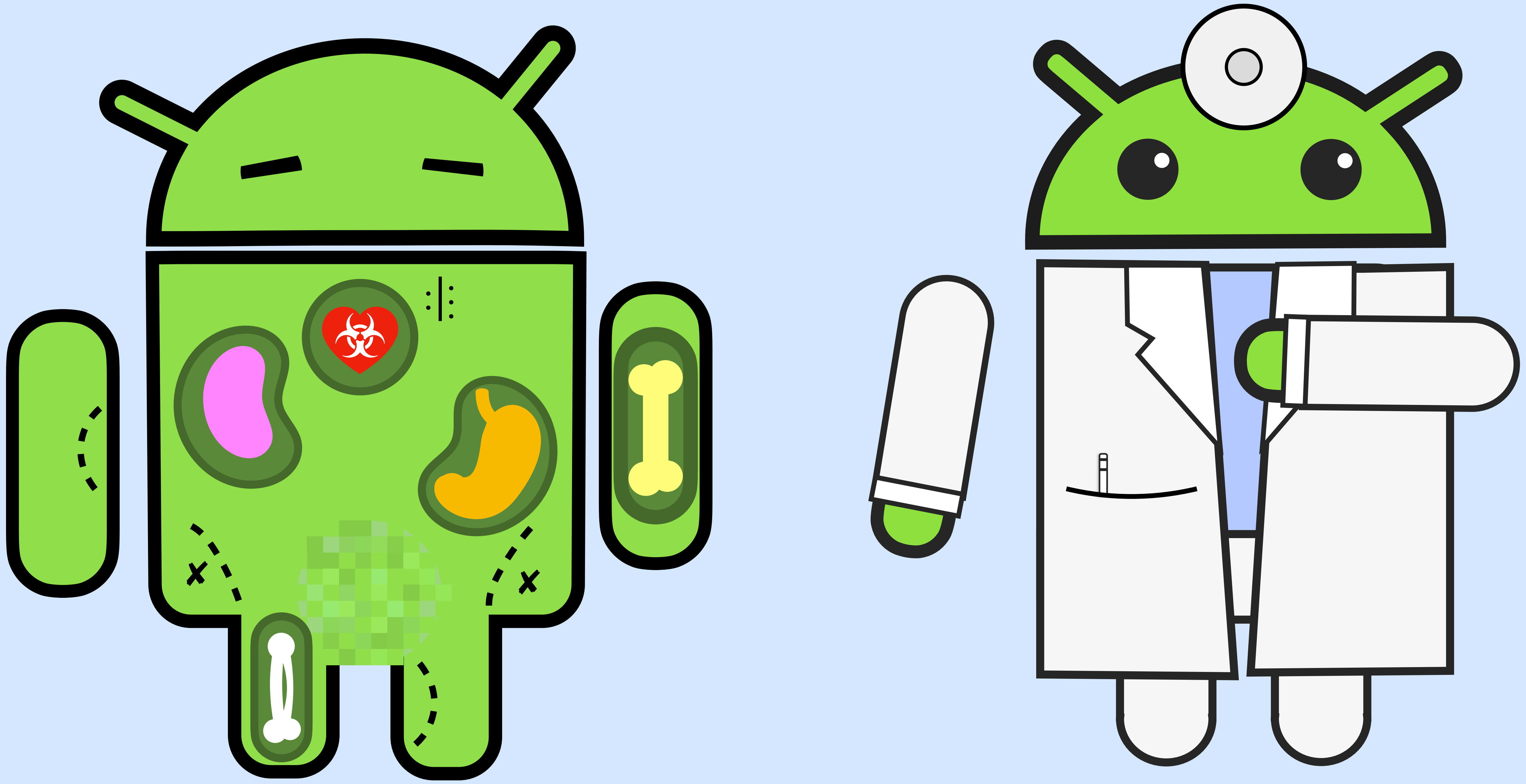
[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Our Android Attack

**Available Transformations**
Code addition through automated software transplantation.

**Preserved Semantics**
Malicious semantics preserved by construction using opaque predicates (inserted code is not executed at runtime).

**Robustness to Preprocessing**
We're robust to:
- removal of redundant code
- undeclared variables
- unlinked resources
- undefined references
- naming conflicts
- no-op instructions.



[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

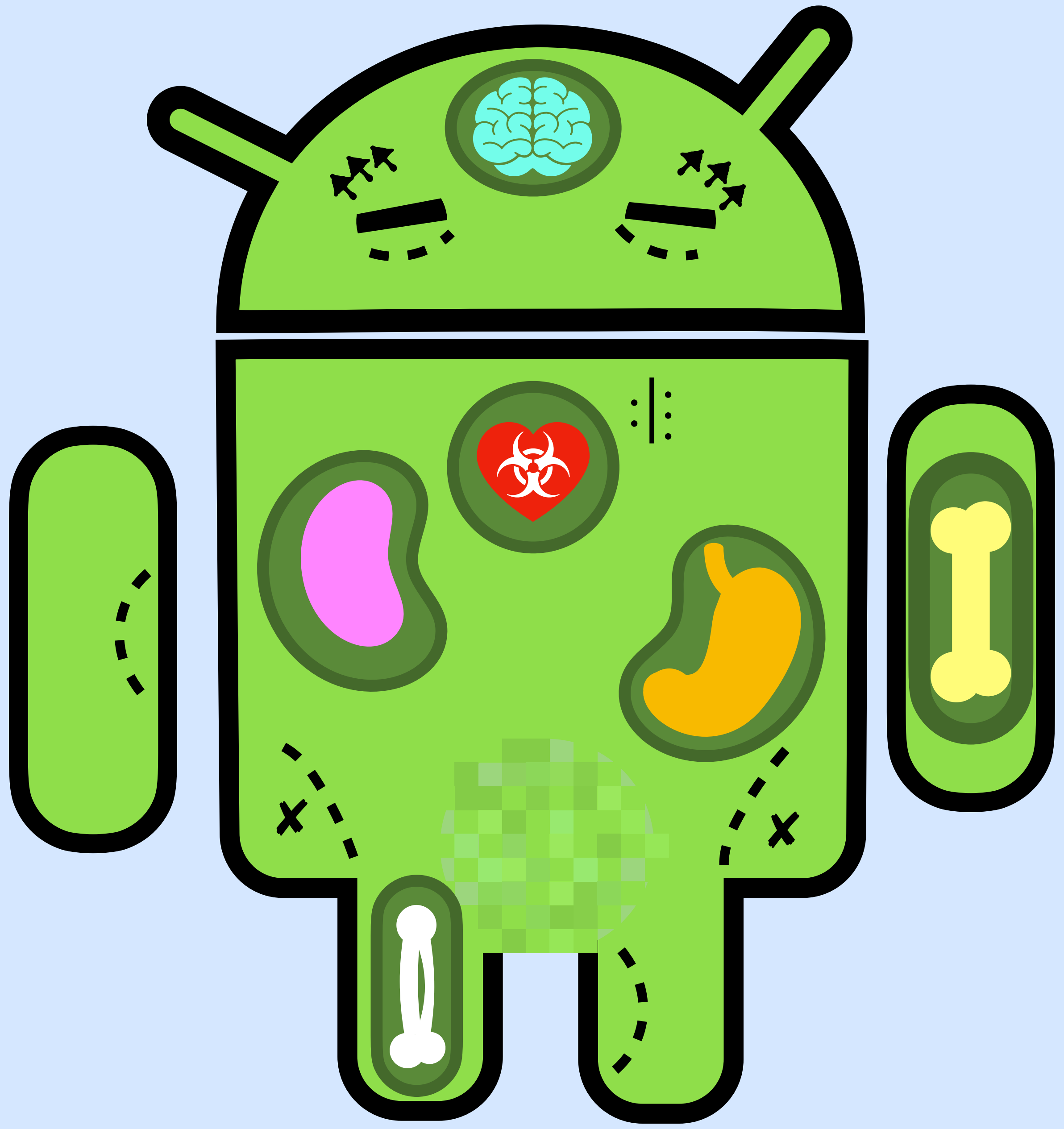
# Our Android Attack

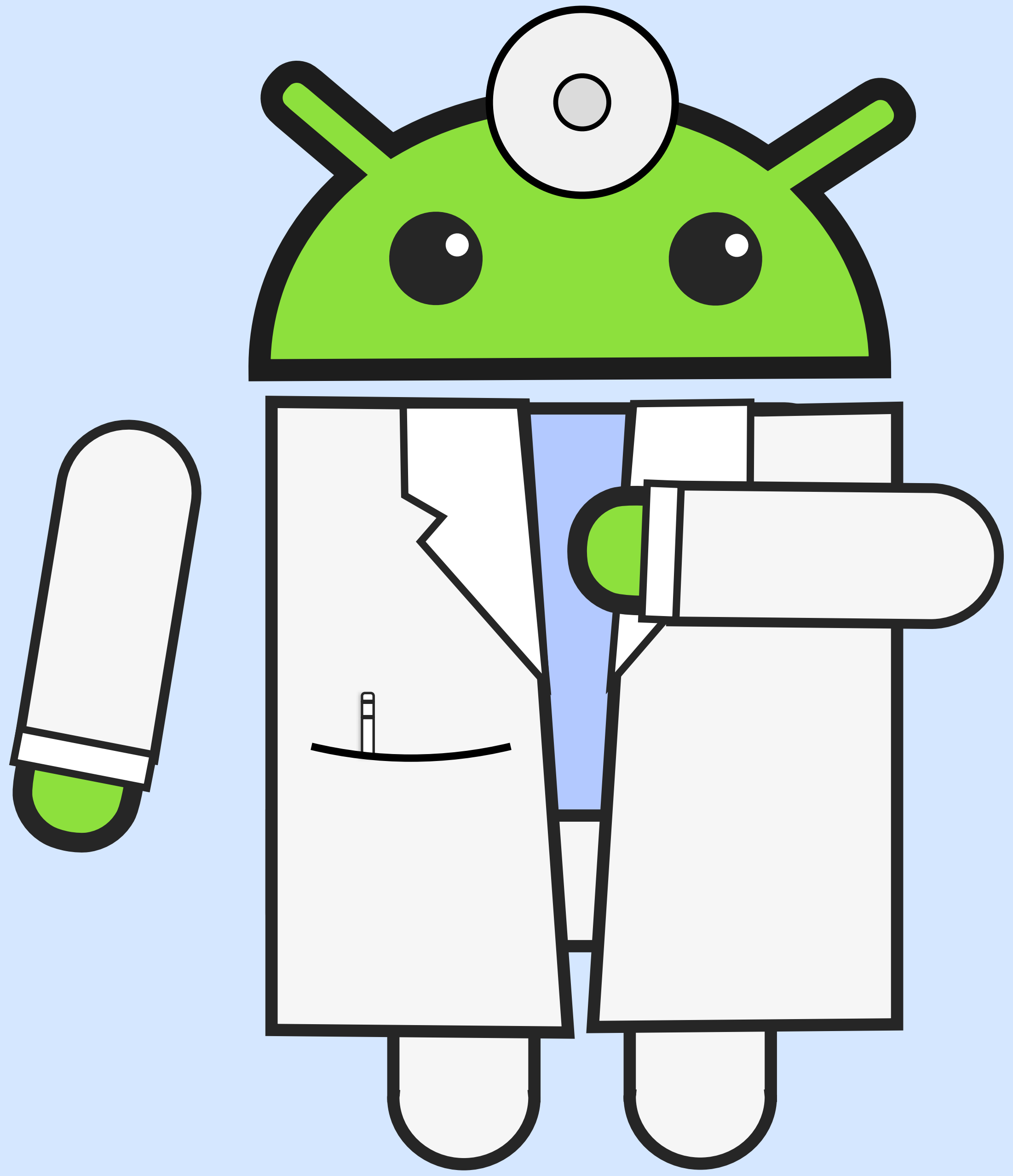
**Available Transformations**
Code addition through automated software transplantation.

**Preserved Semantics**
Malicious semantics preserved by construction using opaque predicates (inserted code is not executed at runtime).

**Robustness to Preprocessing**
We're robust to:
- removal of redundant code
- undeclared variables
- unlinked resources
- undefined references
- naming conflicts
- no-op instructions.

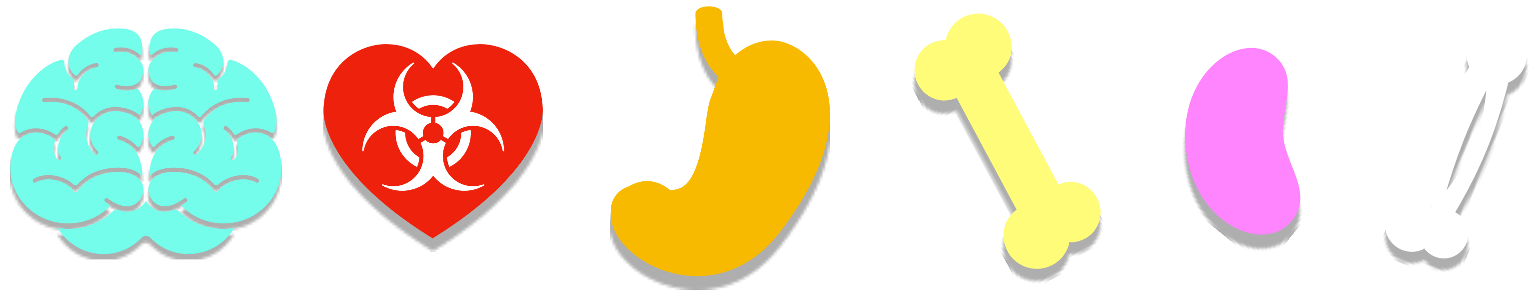
**Plausibility**
Only realistic code is injected (rather than orphaned urls, api calls, etc.)
Mutated apps install and start on an emulator.





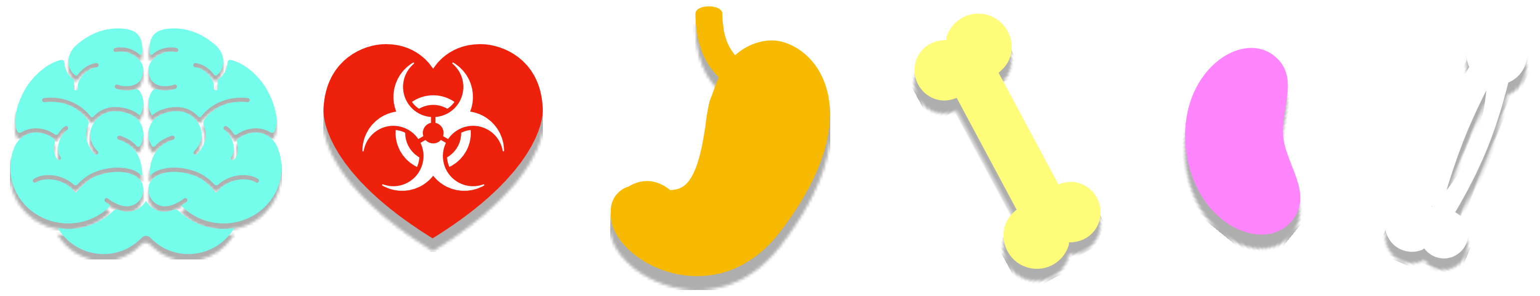
**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

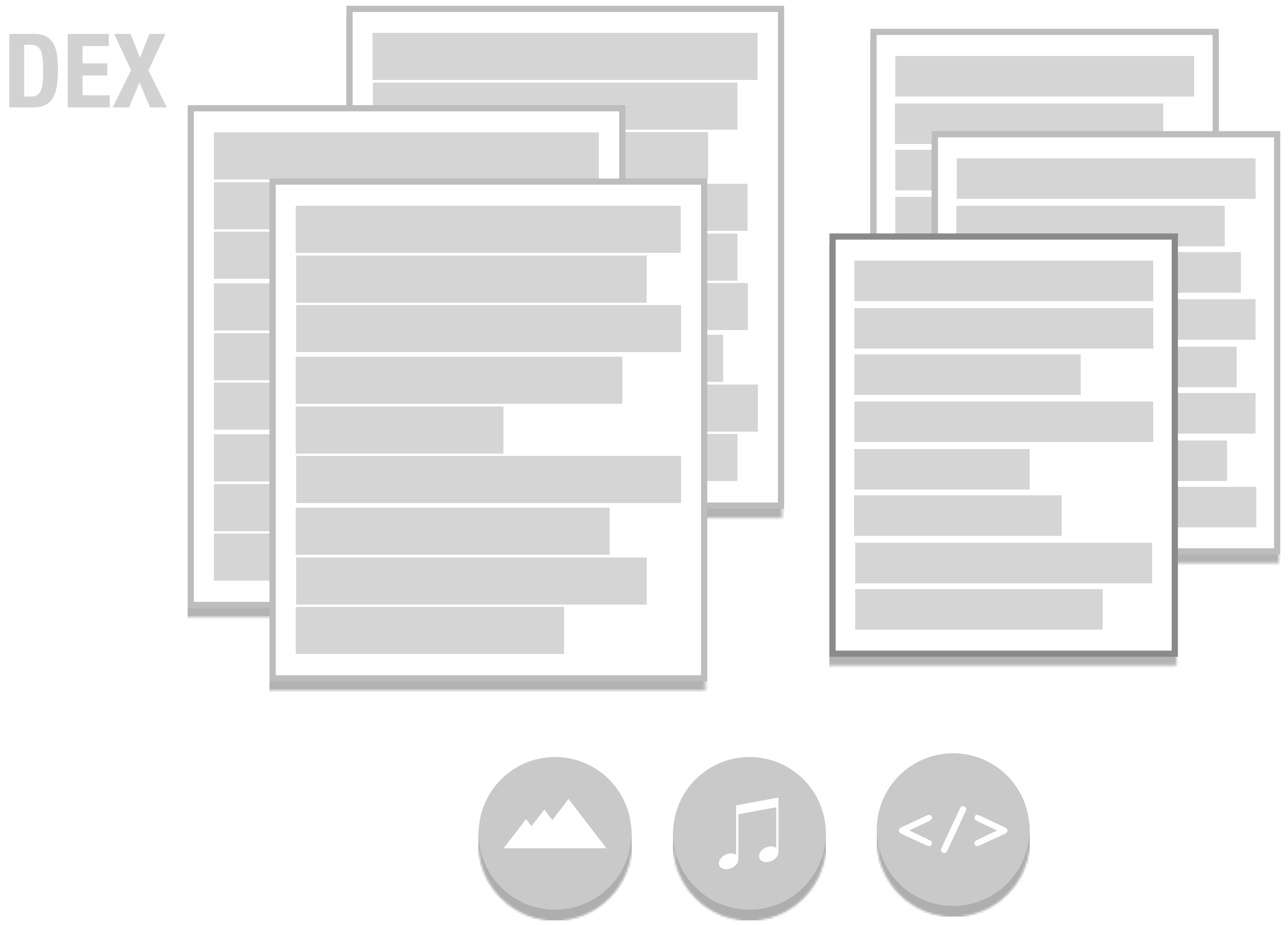
# Organ Harvesting



**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting



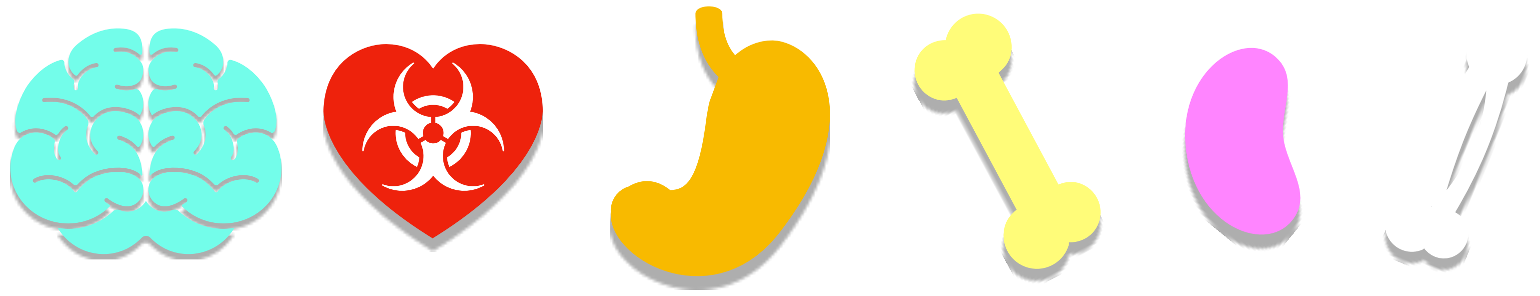


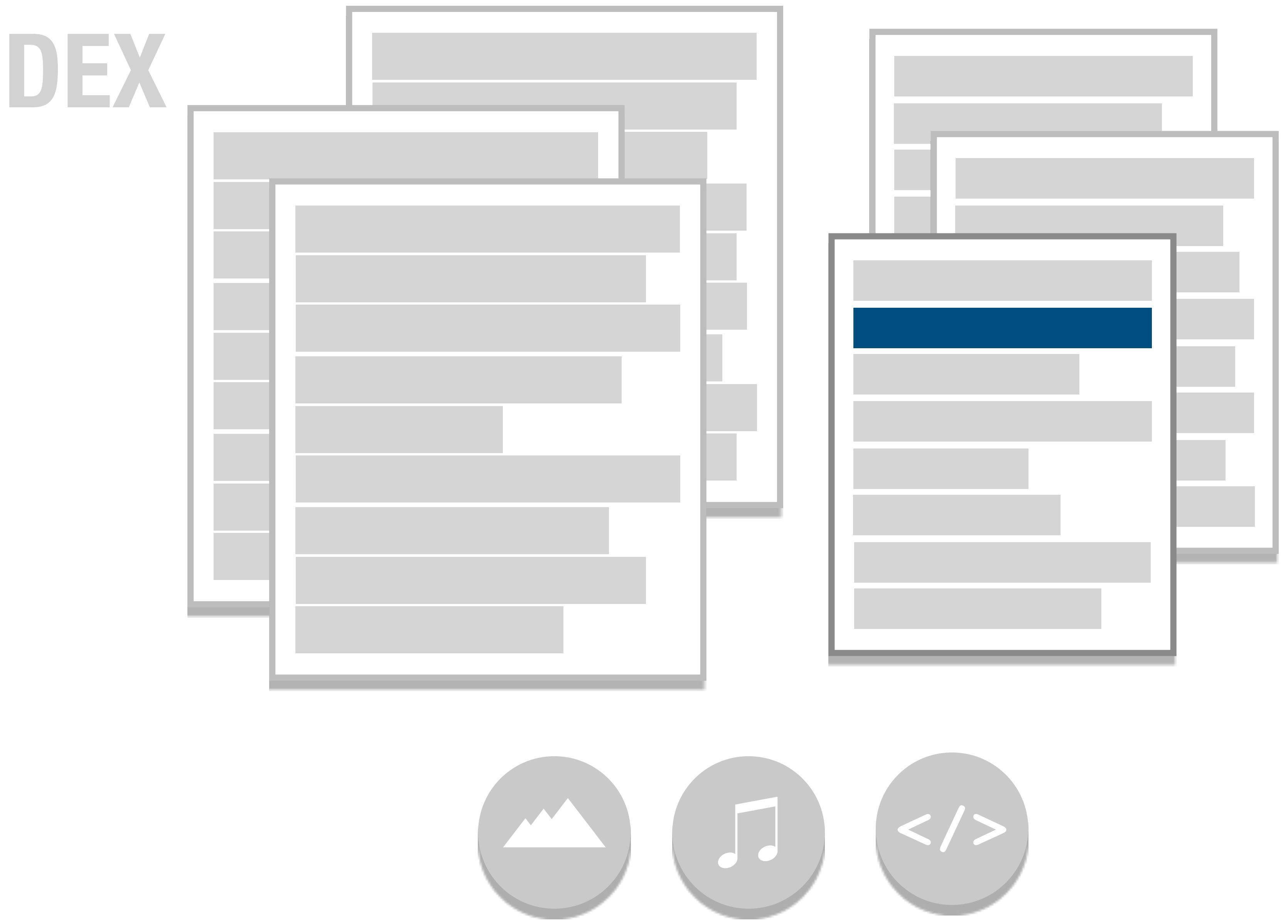

1 Identify feature entry point

Identify activity in dex

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting



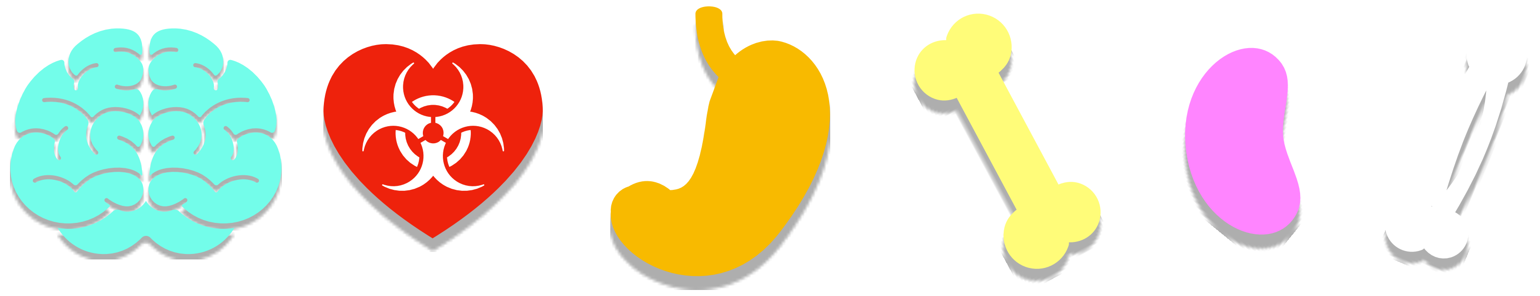


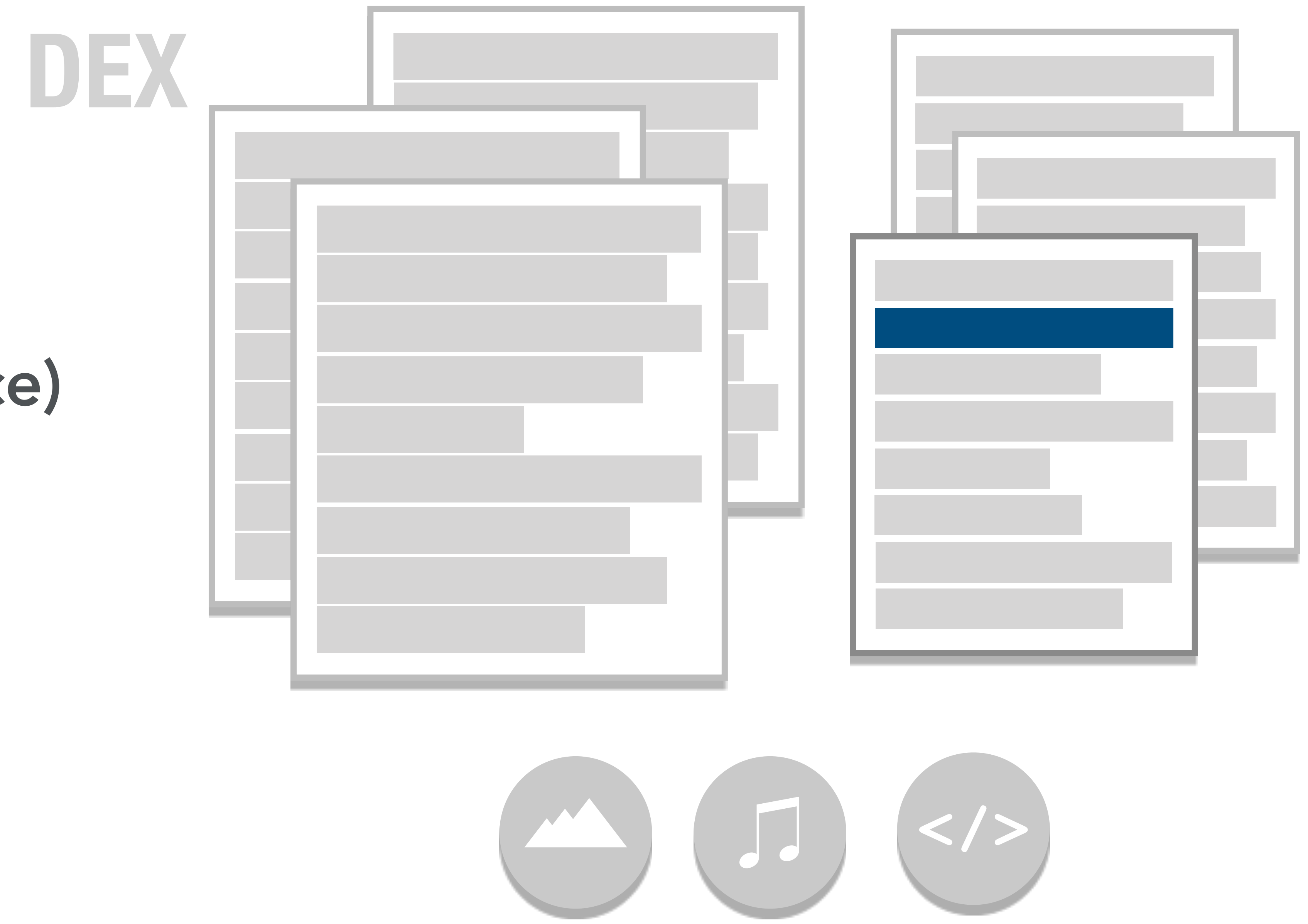

1 Identify feature entry point

Identify activity in dex

[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting



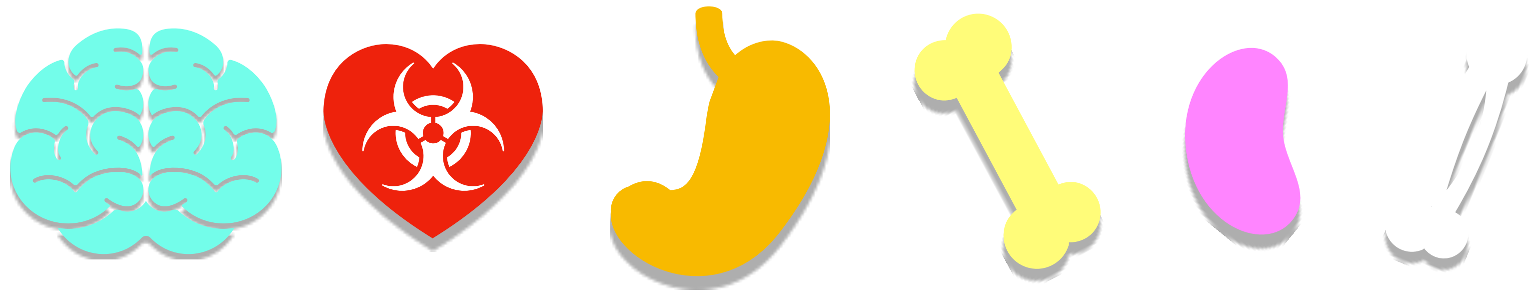


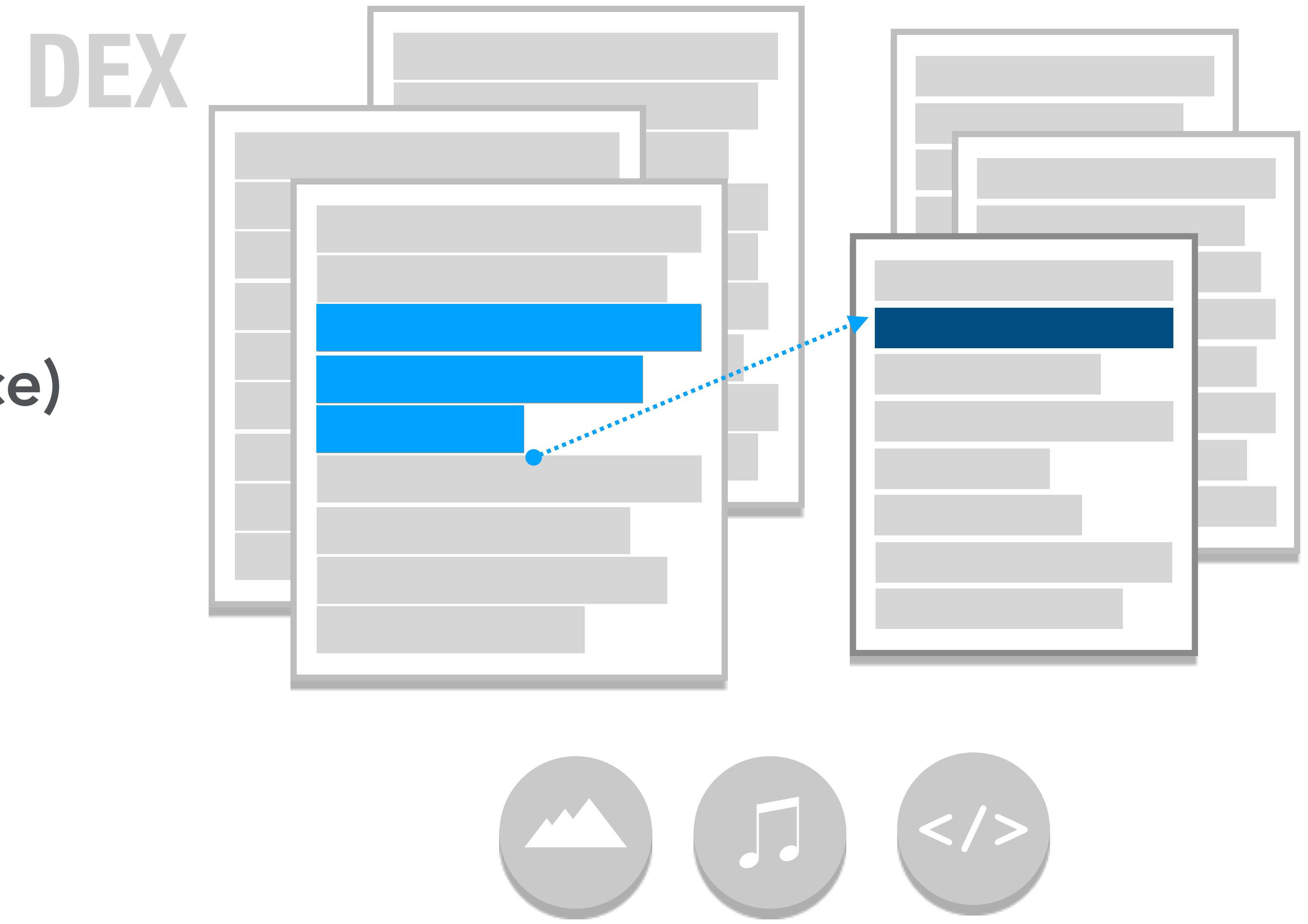

2 Choose any vein (backward slice)

Extract intent creation and startActivity()

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting



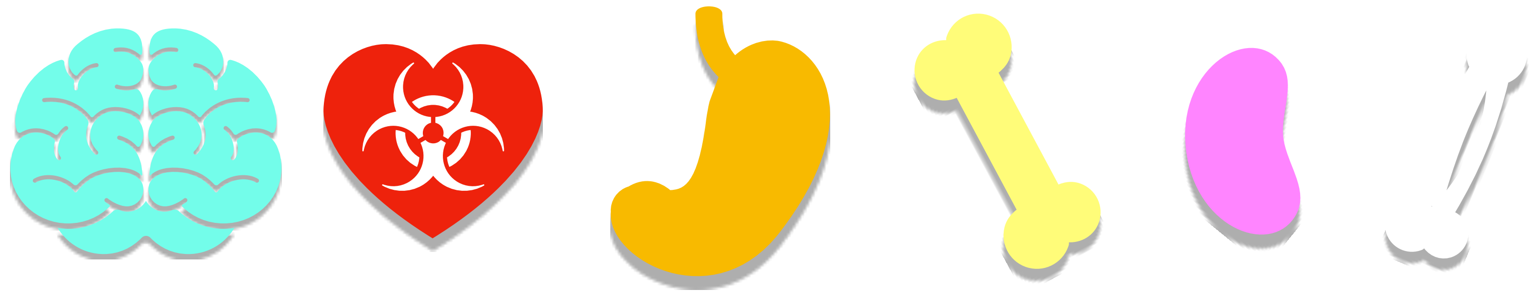


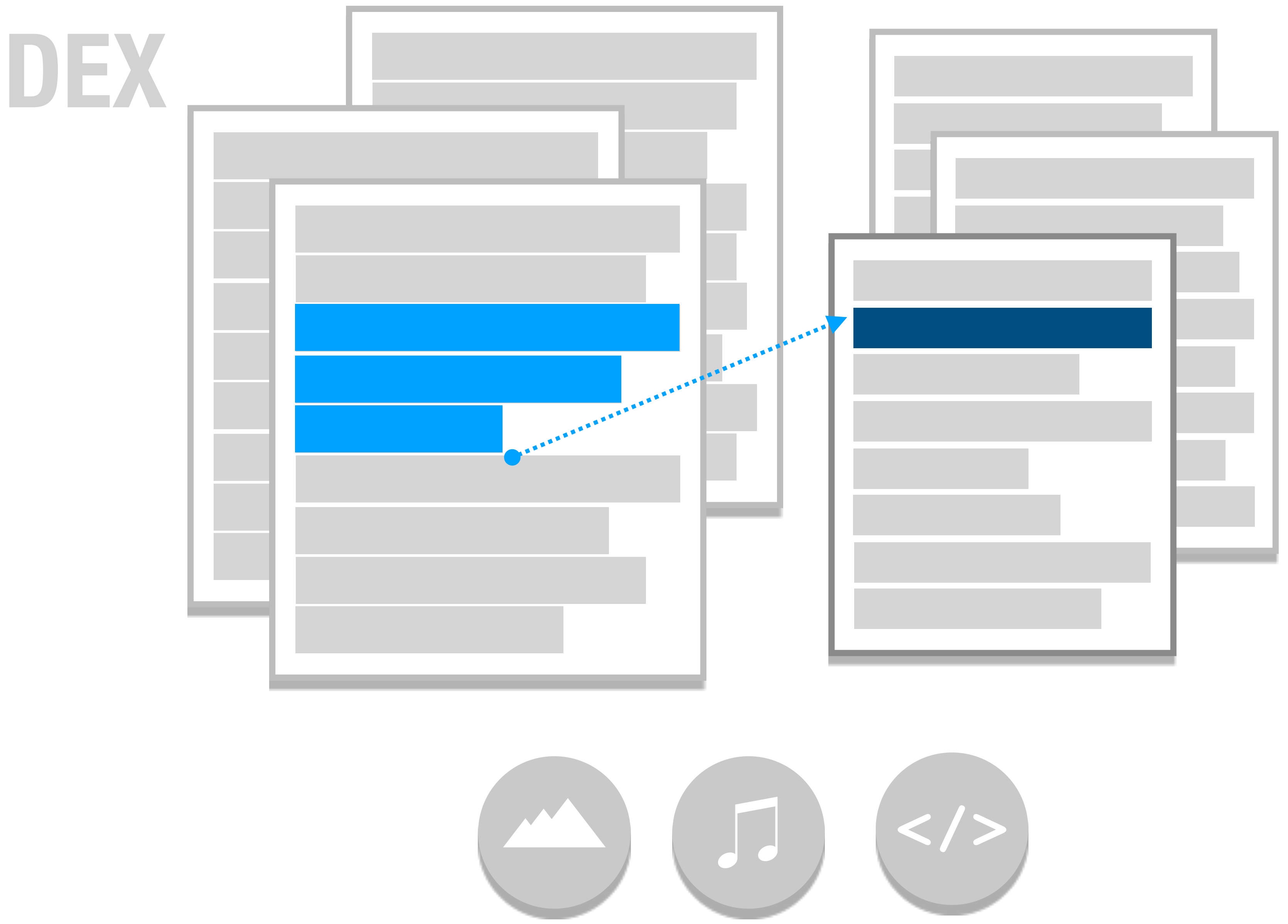

2 Choose any vein (backward slice)

Extract intent creation and startActivity()

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting



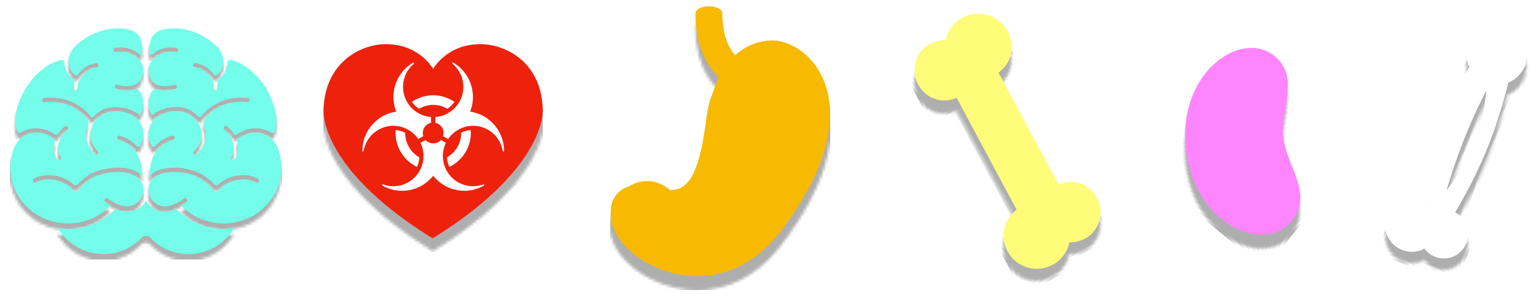


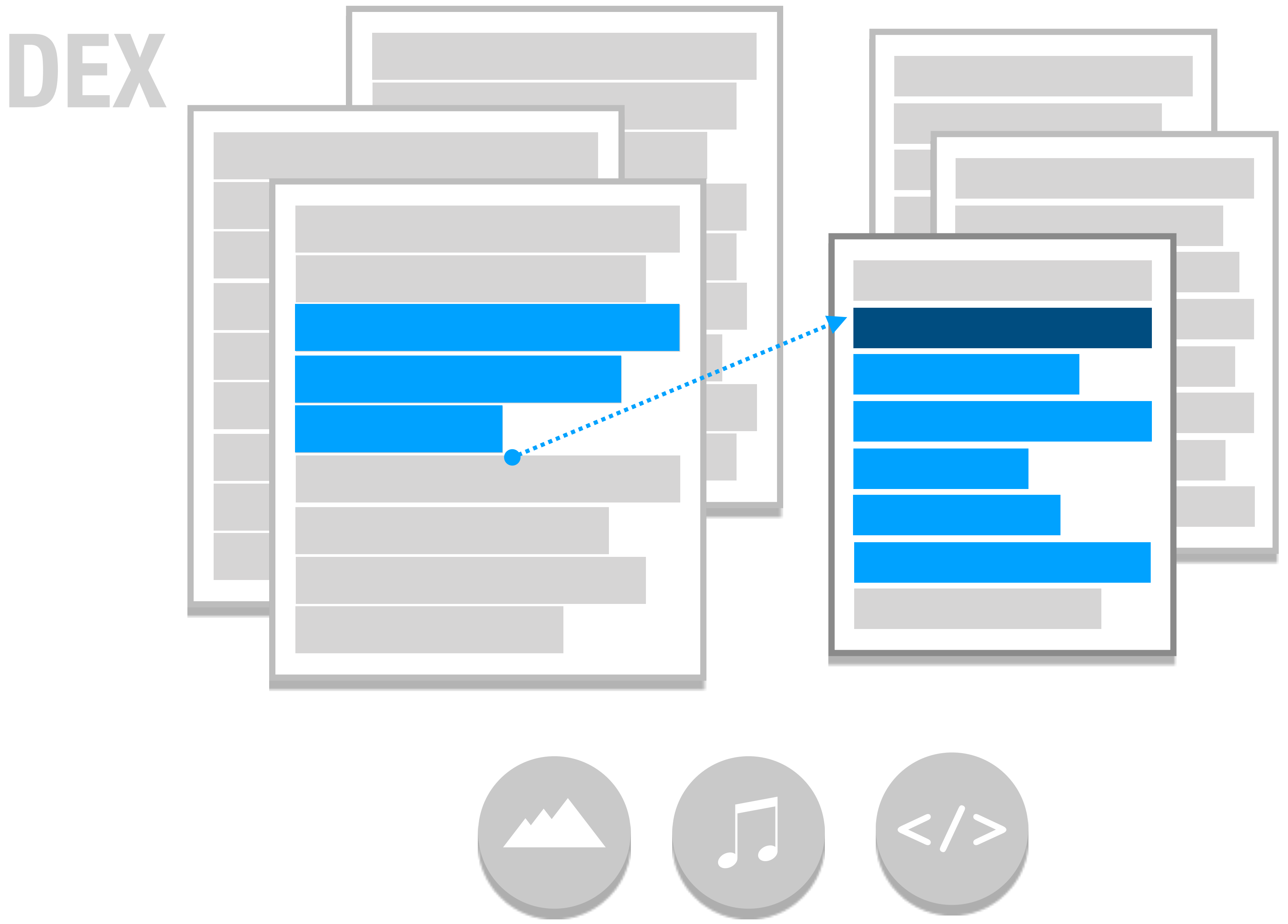

3 Collect organ (forward slice)

Gather activity definition

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting



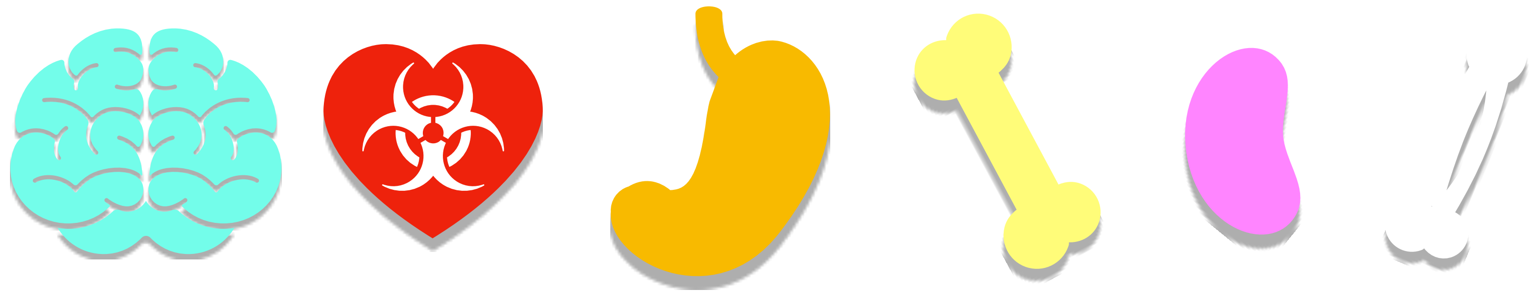



❸ Collect organ (forward slice)

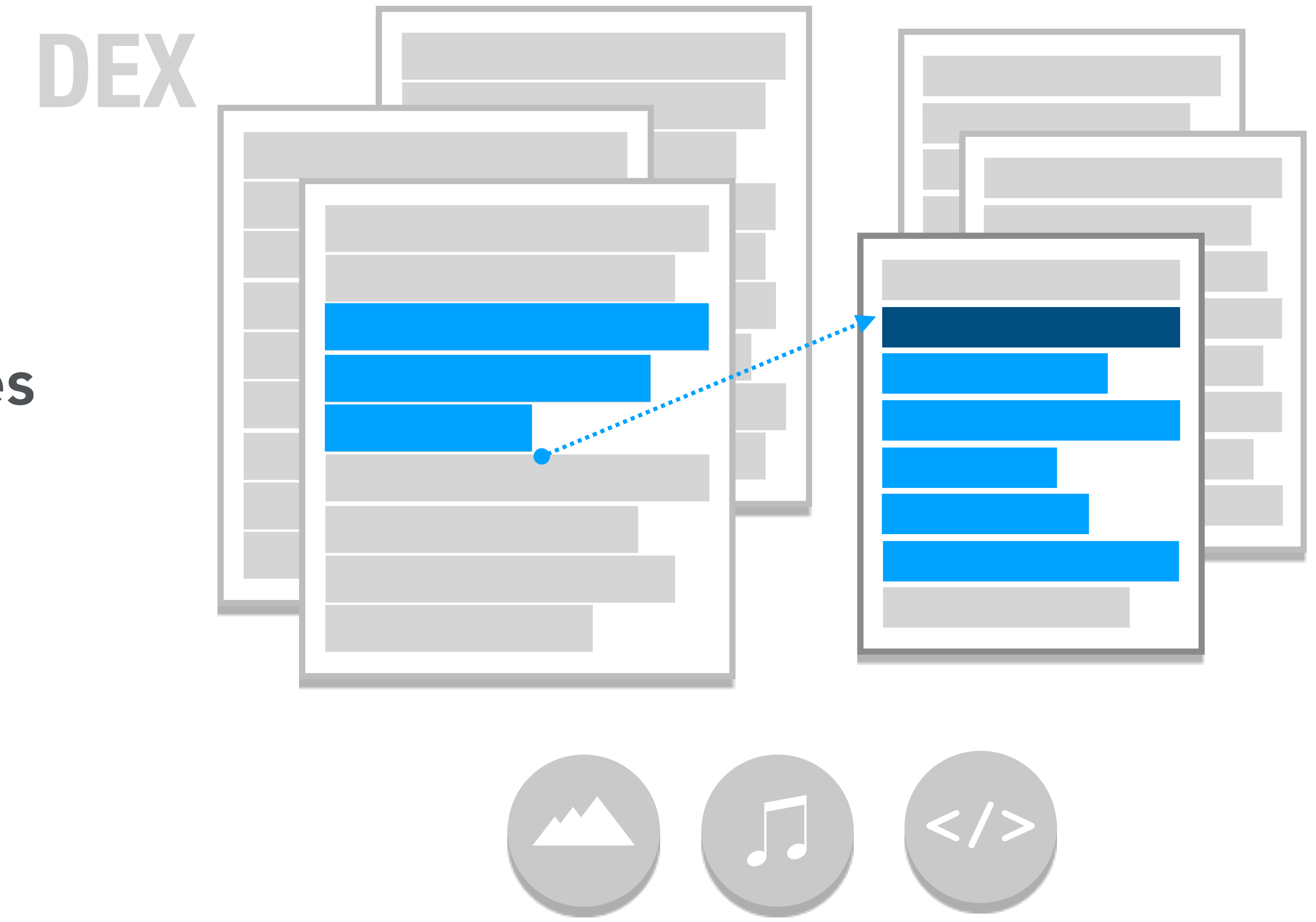
Gather activity definition

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting



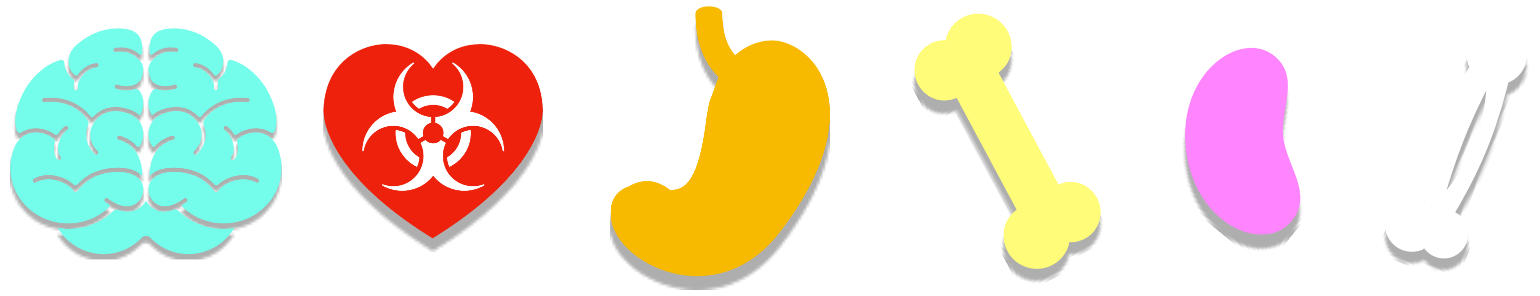
4 Include transitive dependencies

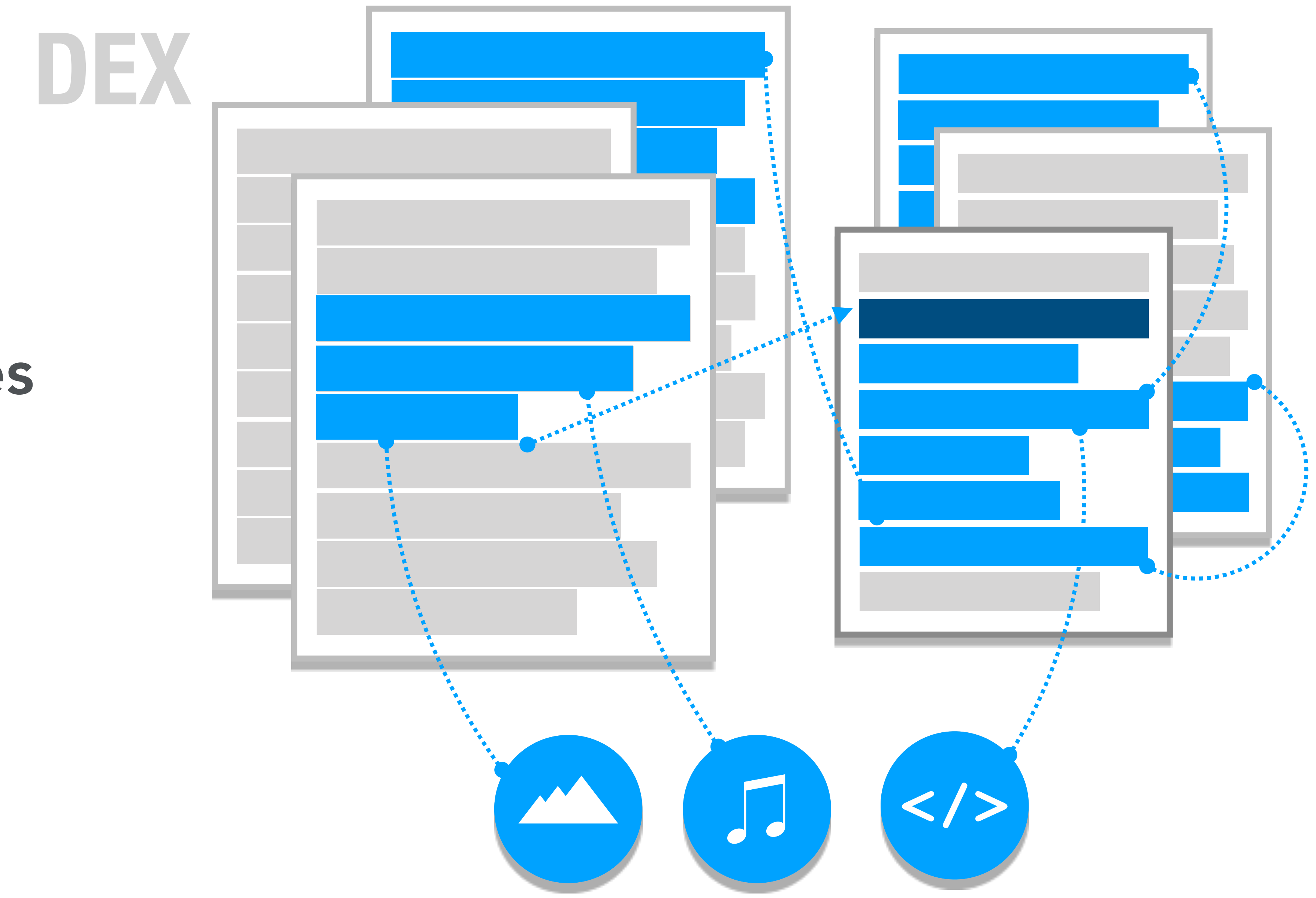



Recursively collect dependencies

[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting



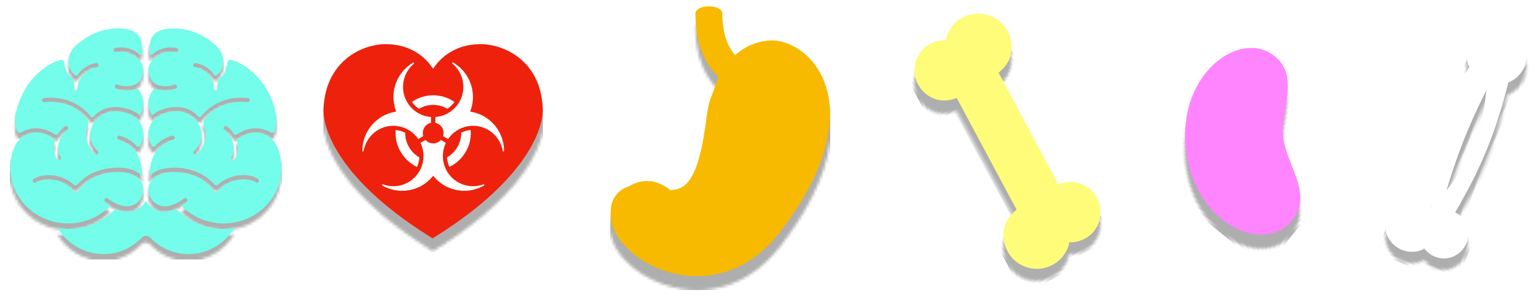


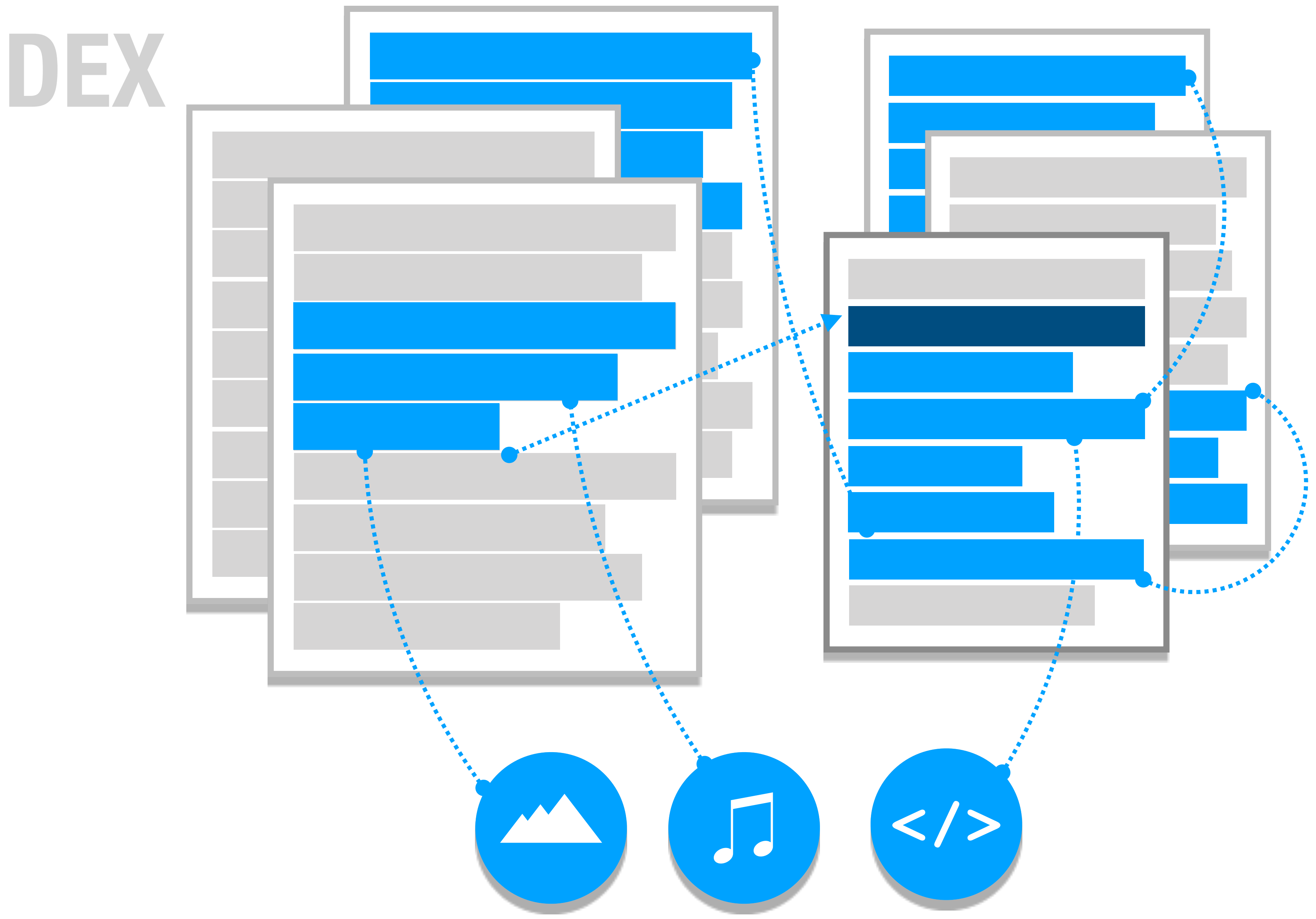

4 Include transitive dependencies

Recursively collect dependencies

[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting



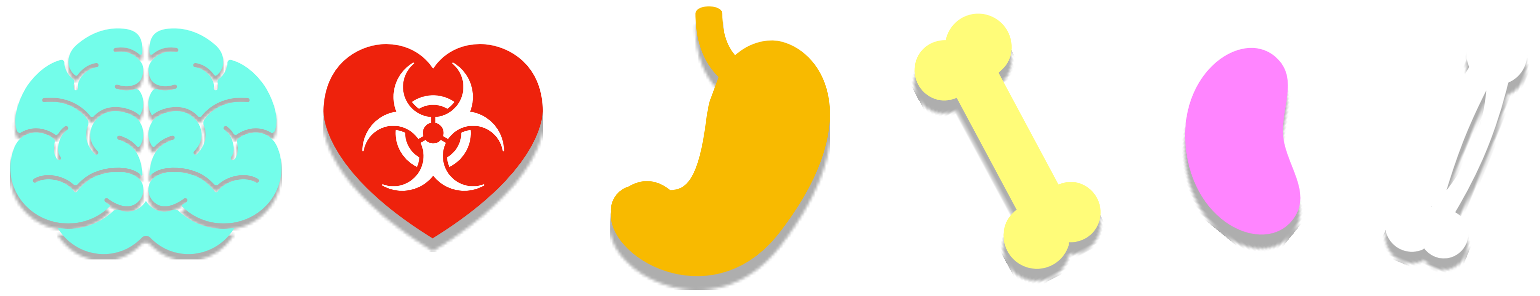


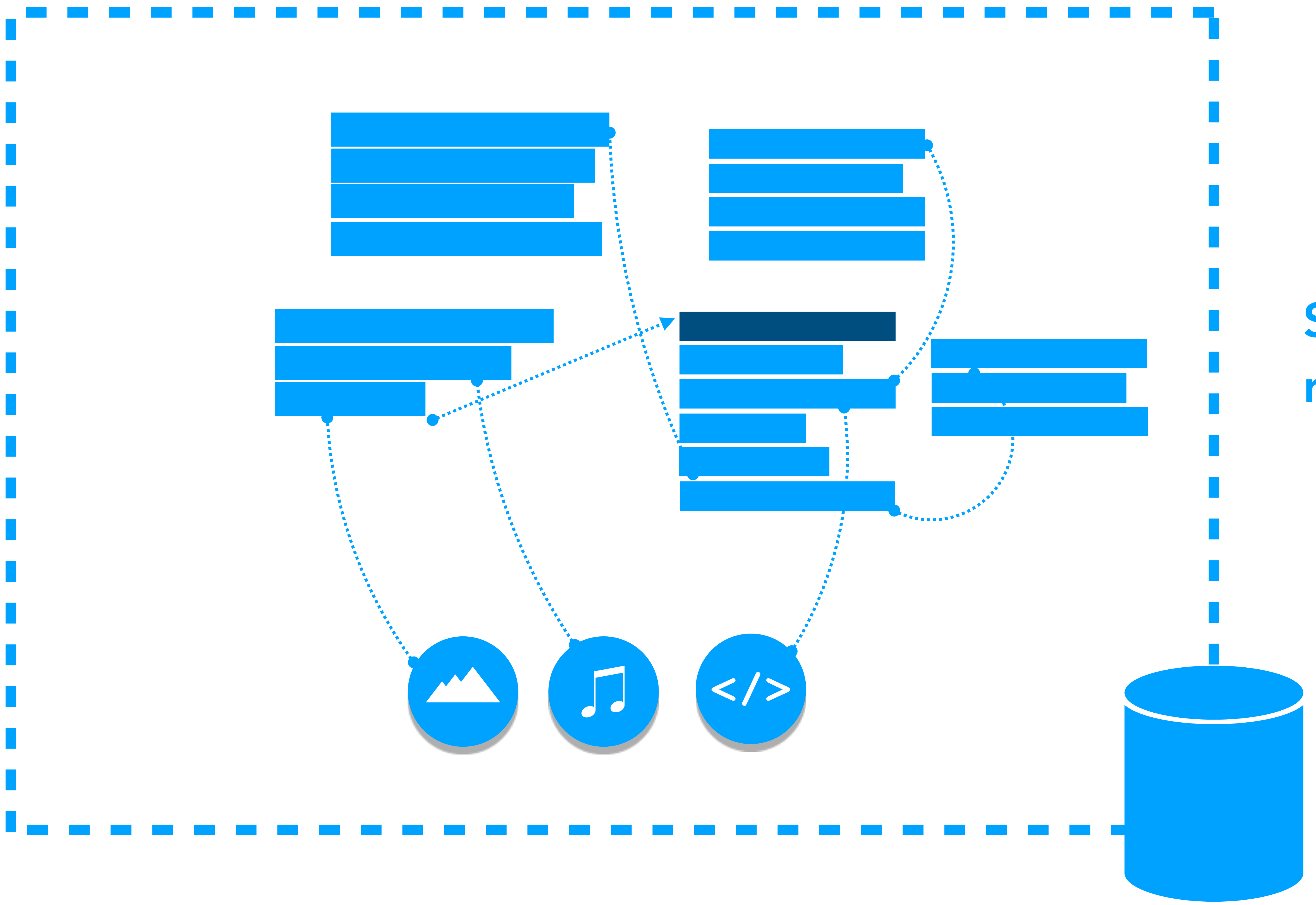

5 Store organ in an “ice box”

Save gadget to a database ready for the attack

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Organ Harvesting





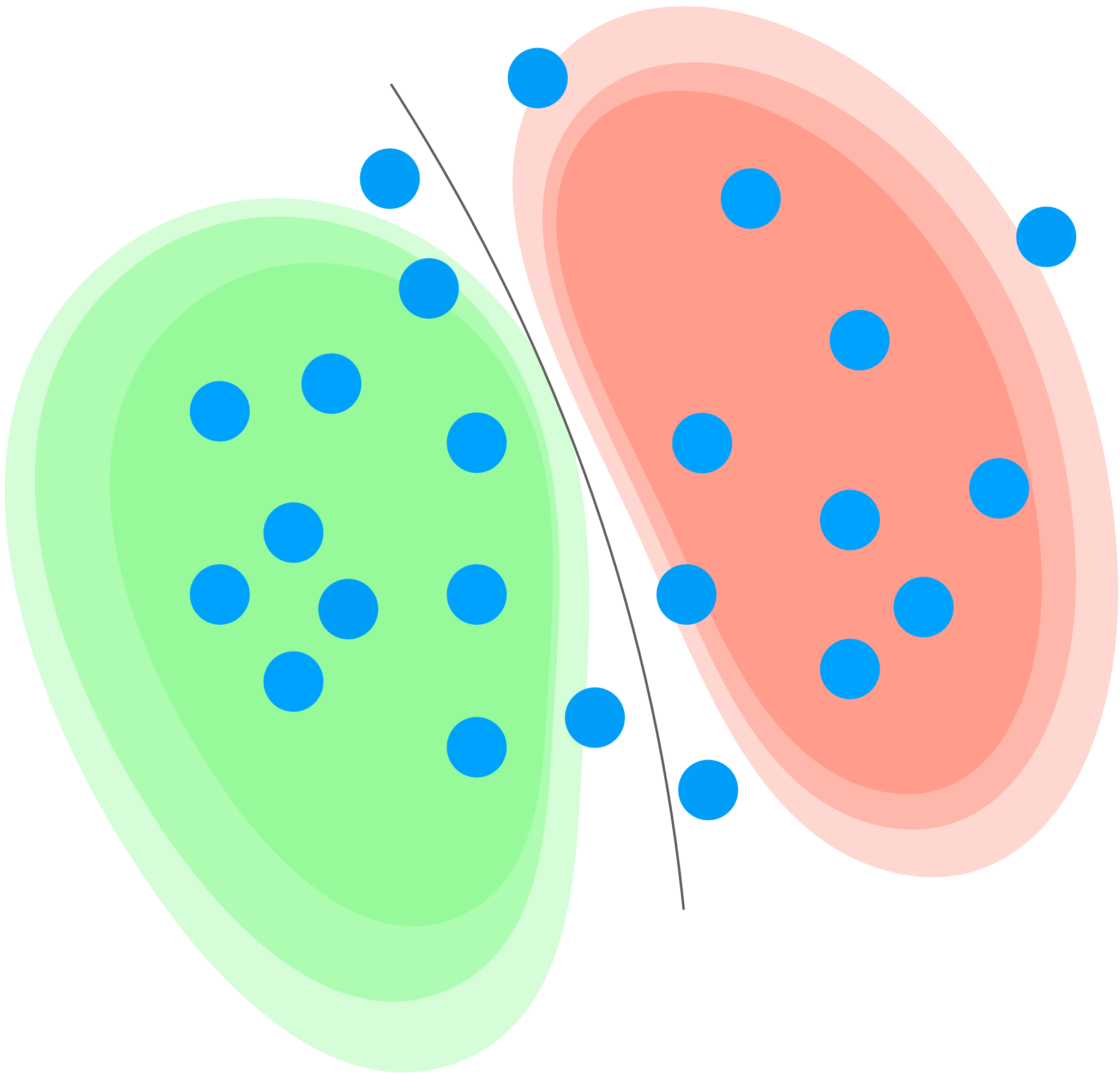


5 Store organ in an "ice box"

Save gadget to a database ready for the attack

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

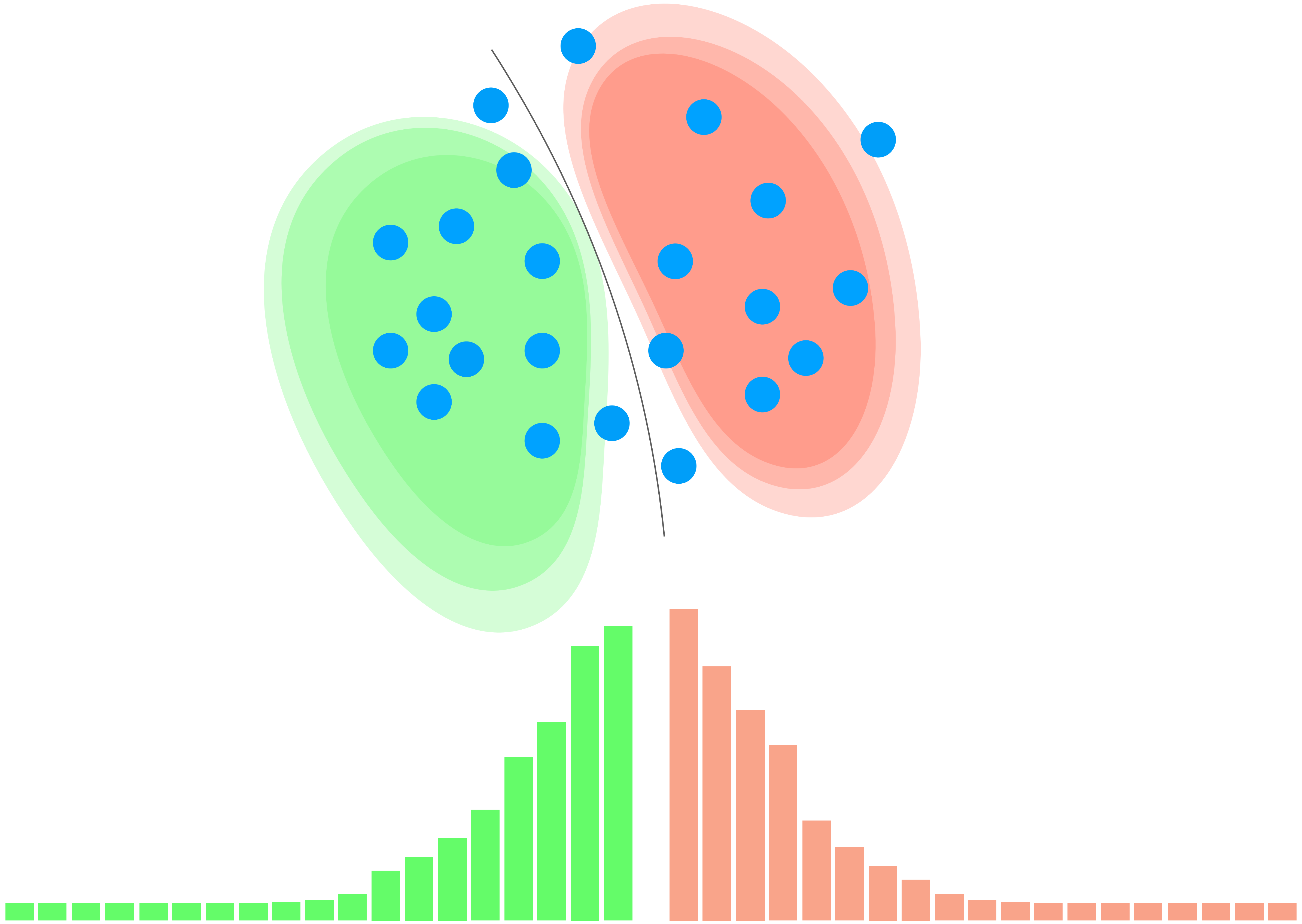
# Attack Overview



**[IEEE S&P 2020]** **Intriguing Properties of Adversarial ML Attacks in the Problem Space**
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Attack Overview

Given a trained target model

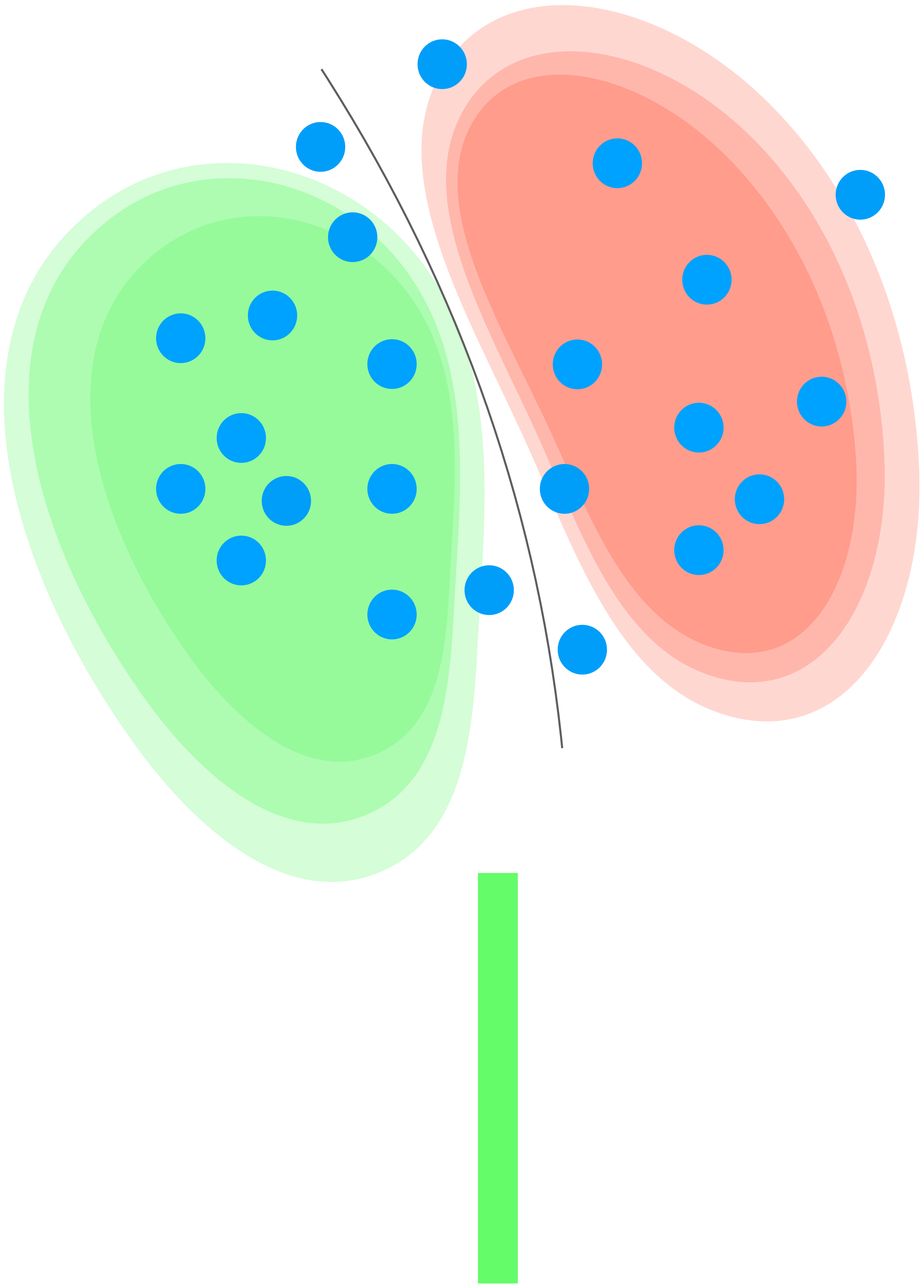


**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Attack Overview

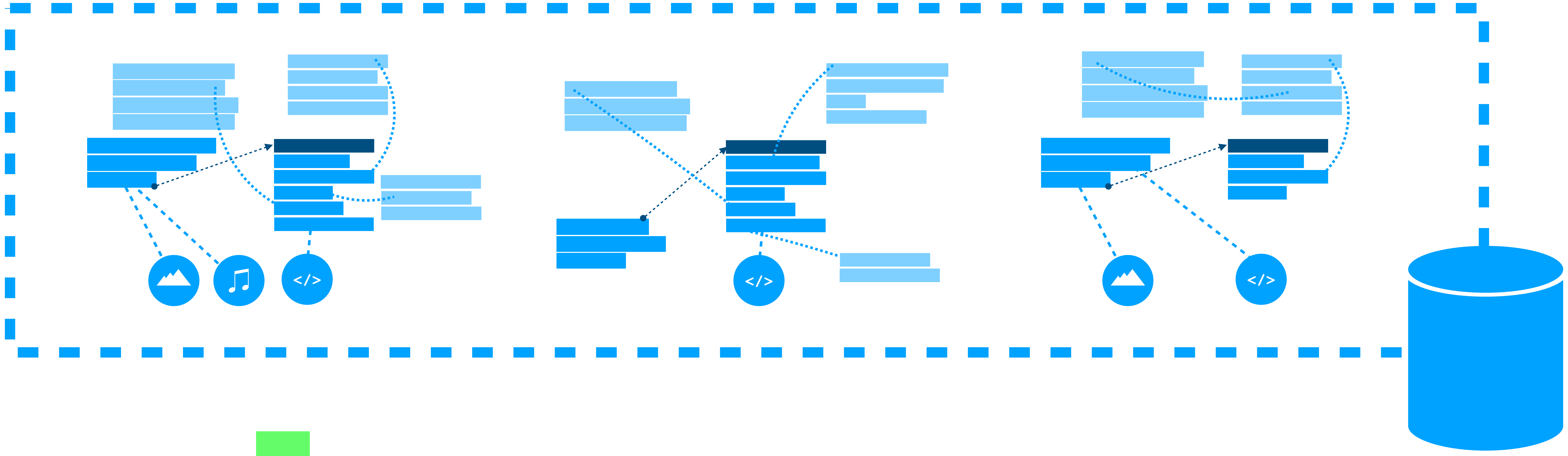
Given a trained target model

First pick feature with greatest 'benign' weight



**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Attack Overview



Given a trained target model

First pick feature with greatest ‘benign’ weight

Find a corresponding organ from the ice box

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space

https://s2lab.cs.ucl.ac.uk/projects/intriguing

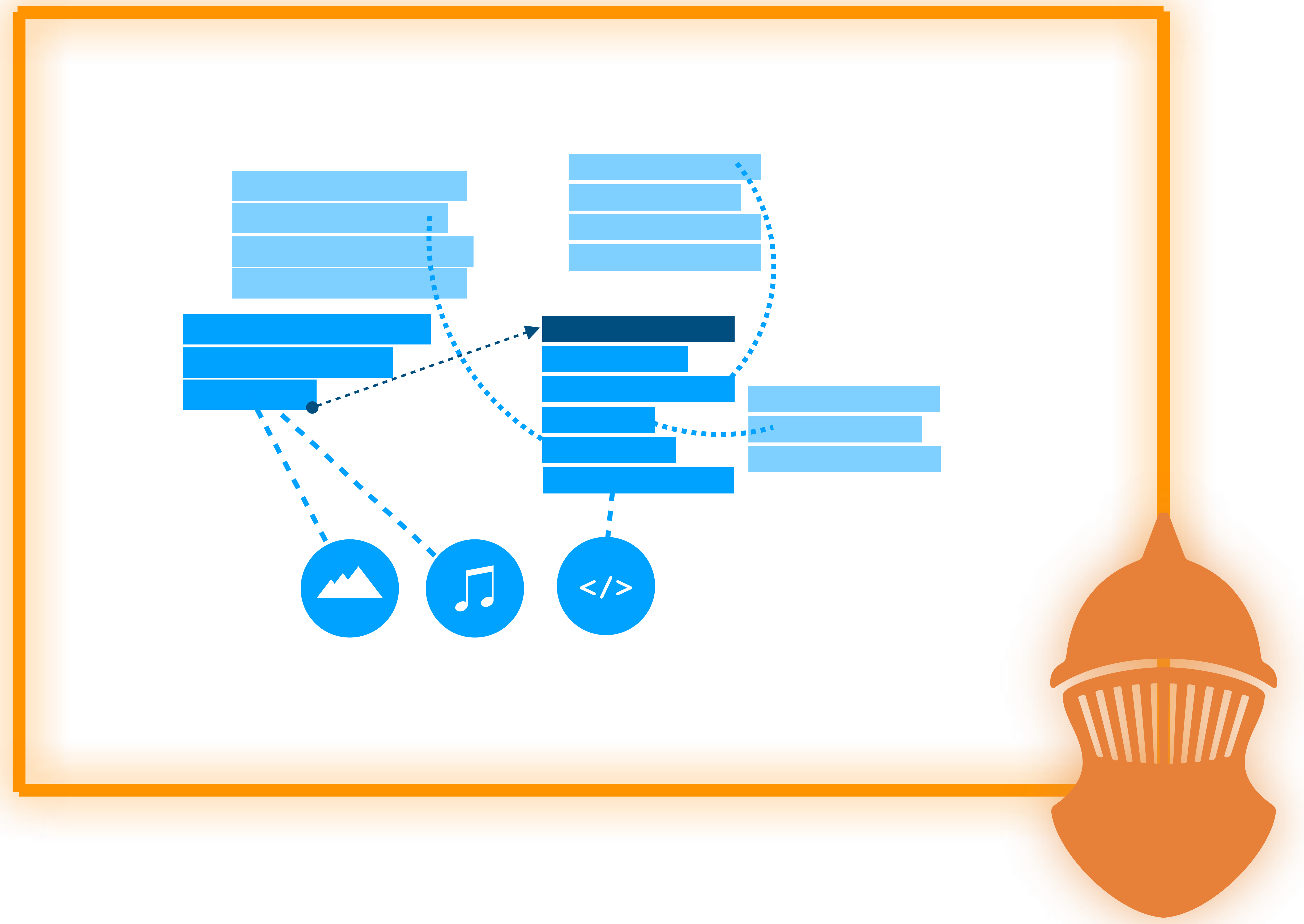
# Attack Overview

Given a trained target model

First pick feature with greatest 'benign' weight

Find a corresponding organ from the ice box

Wrap the organ in an opaque predicate




[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

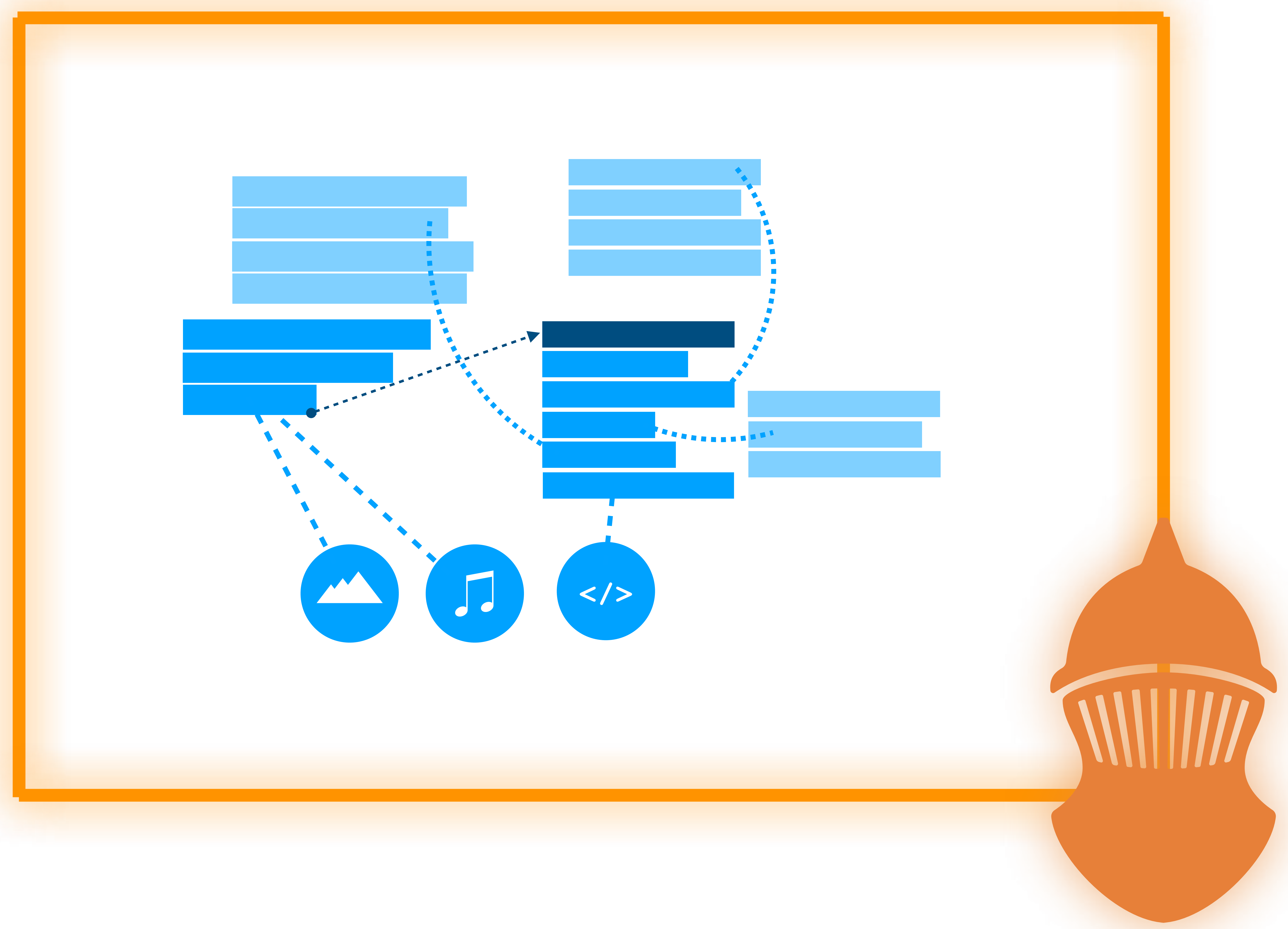
# Attack Overview

Given a trained target model

First pick feature with greatest 'benign' weight

Find a corresponding organ from the ice box

Wrap the organ in an opaque predicate




[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Attack Overview

Given a trained target model

First pick feature with greatest 'benign' weight

Find a corresponding organ from the ice box

Wrap the organ in an opaque predicate

Inject the new benign code and repackage

**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Attack Overview

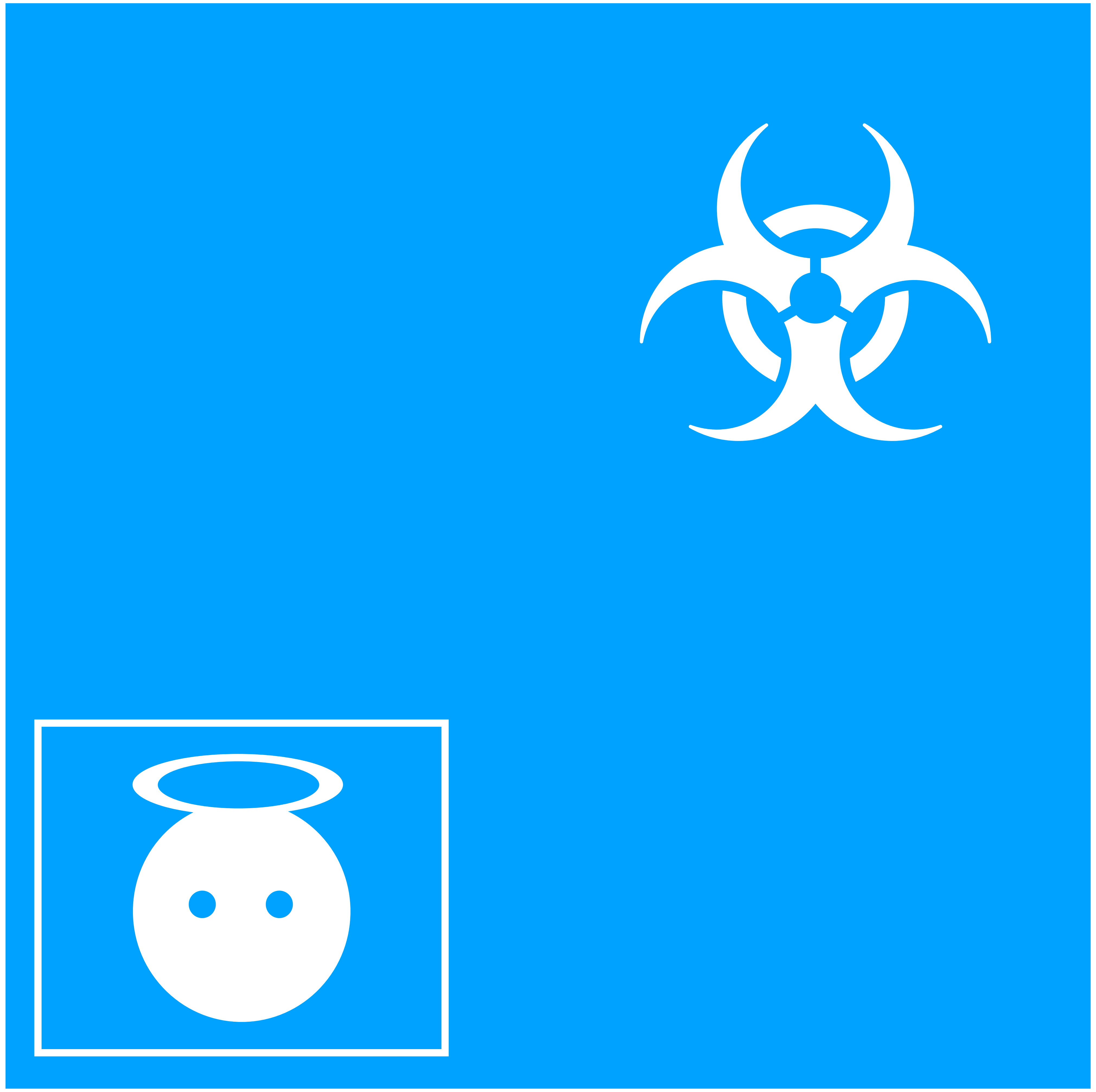
Given a trained target model

First pick feature with greatest 'benign' weight

Find a corresponding organ from the ice box

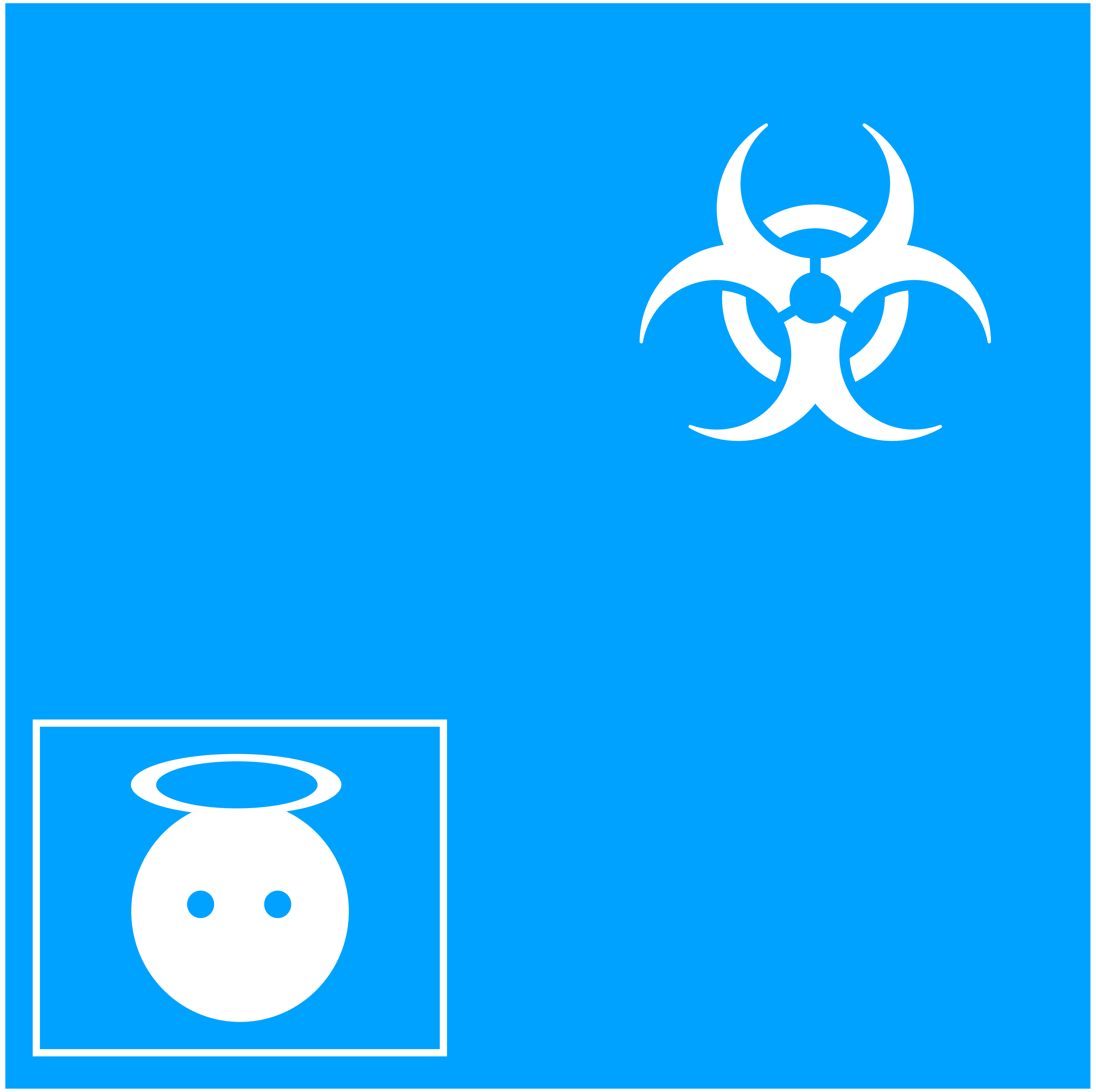
Wrap the organ in an opaque predicate

Inject the new benign code and repackage



**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

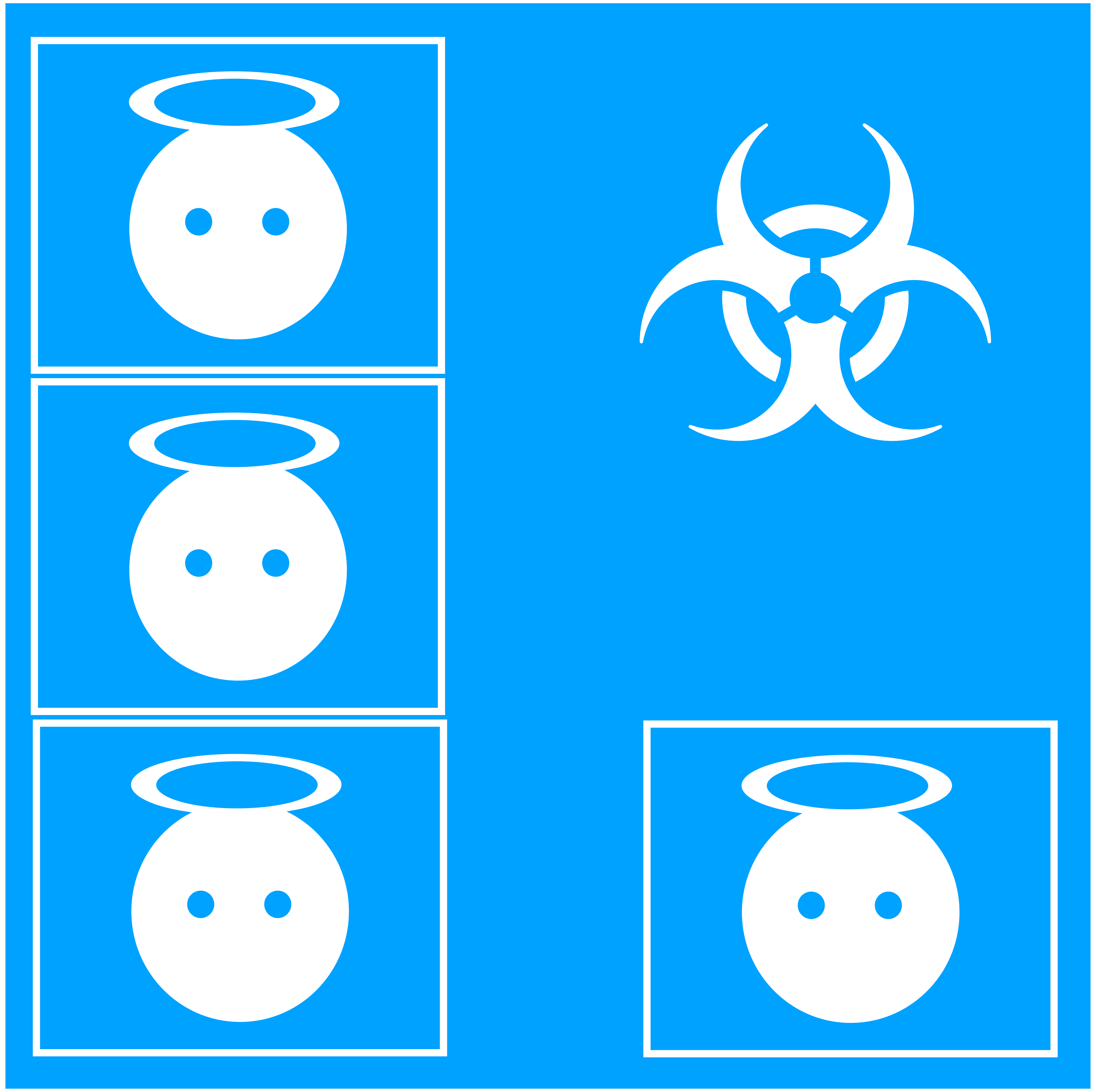
# Attack Overview

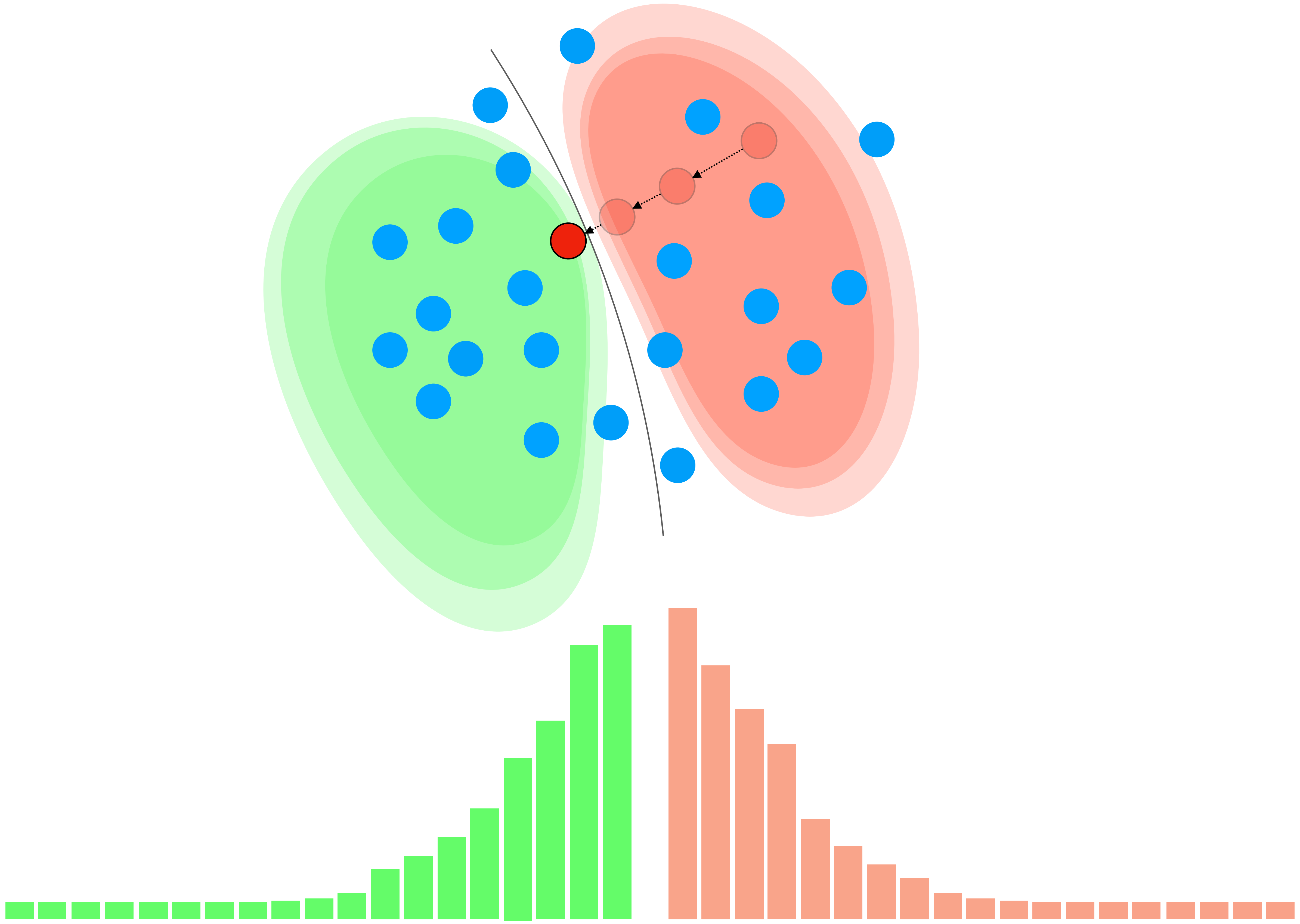


**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Attack Overview

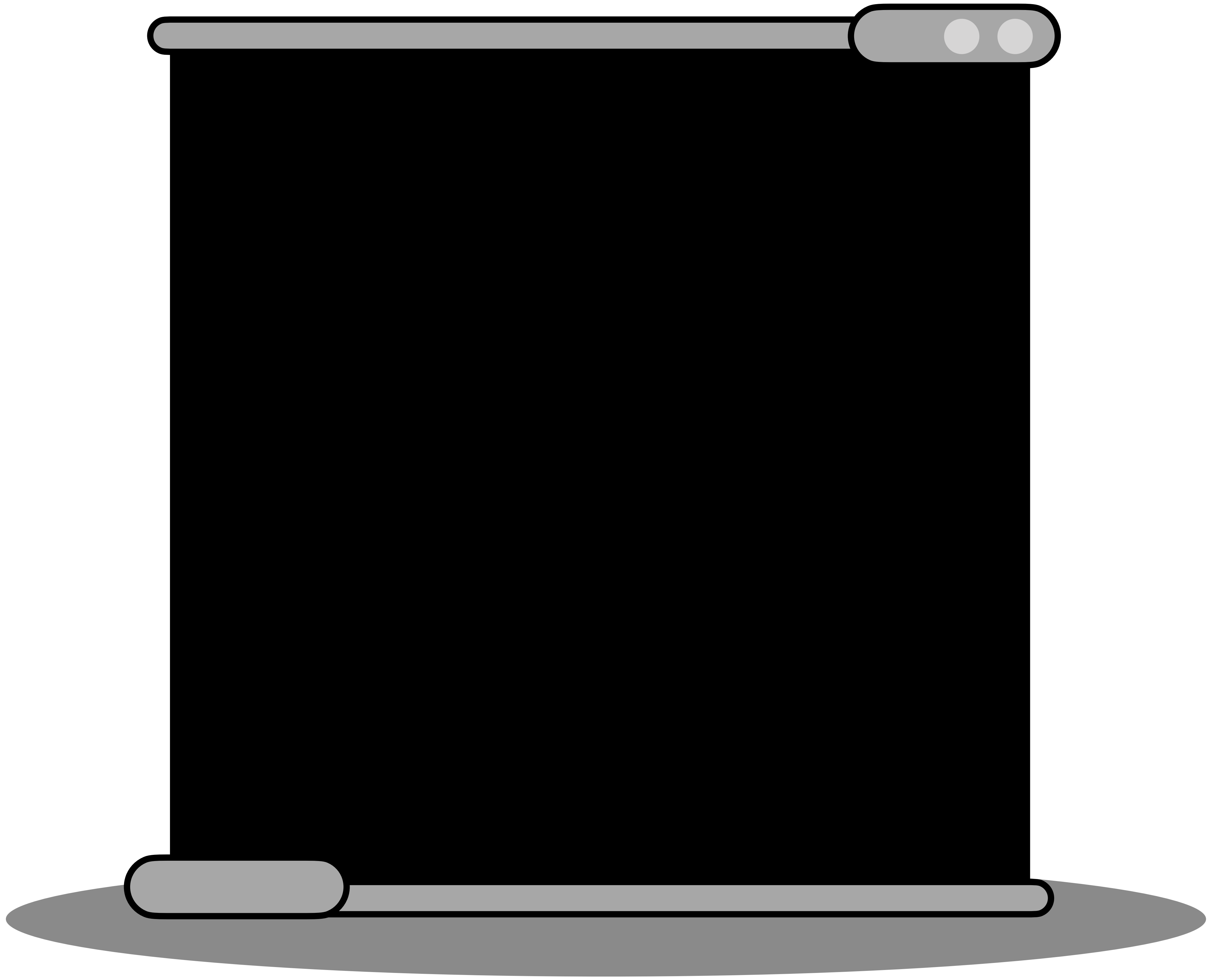
Continue choosing benign features until the app is misclassified





**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

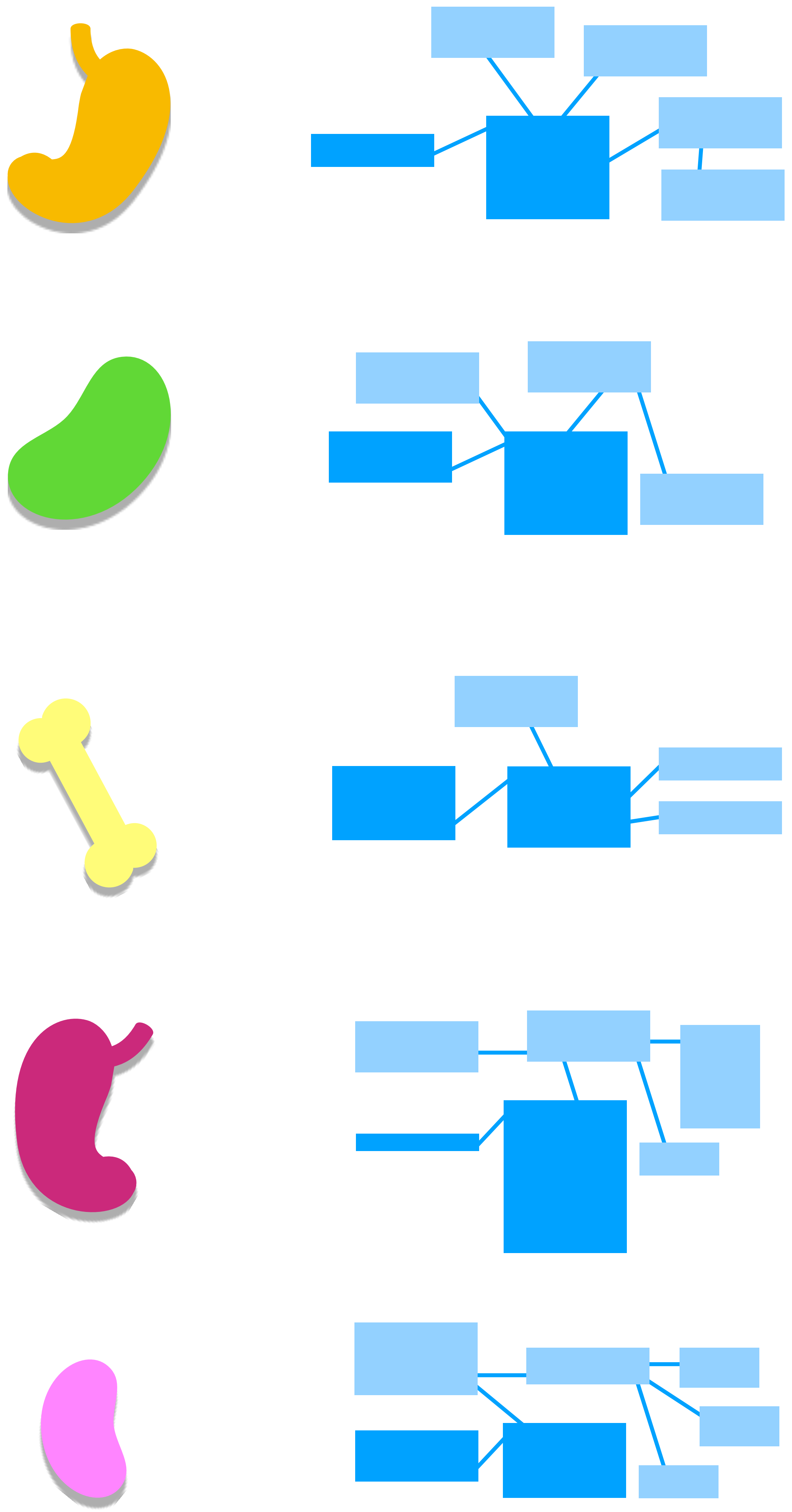
# Side-Effects

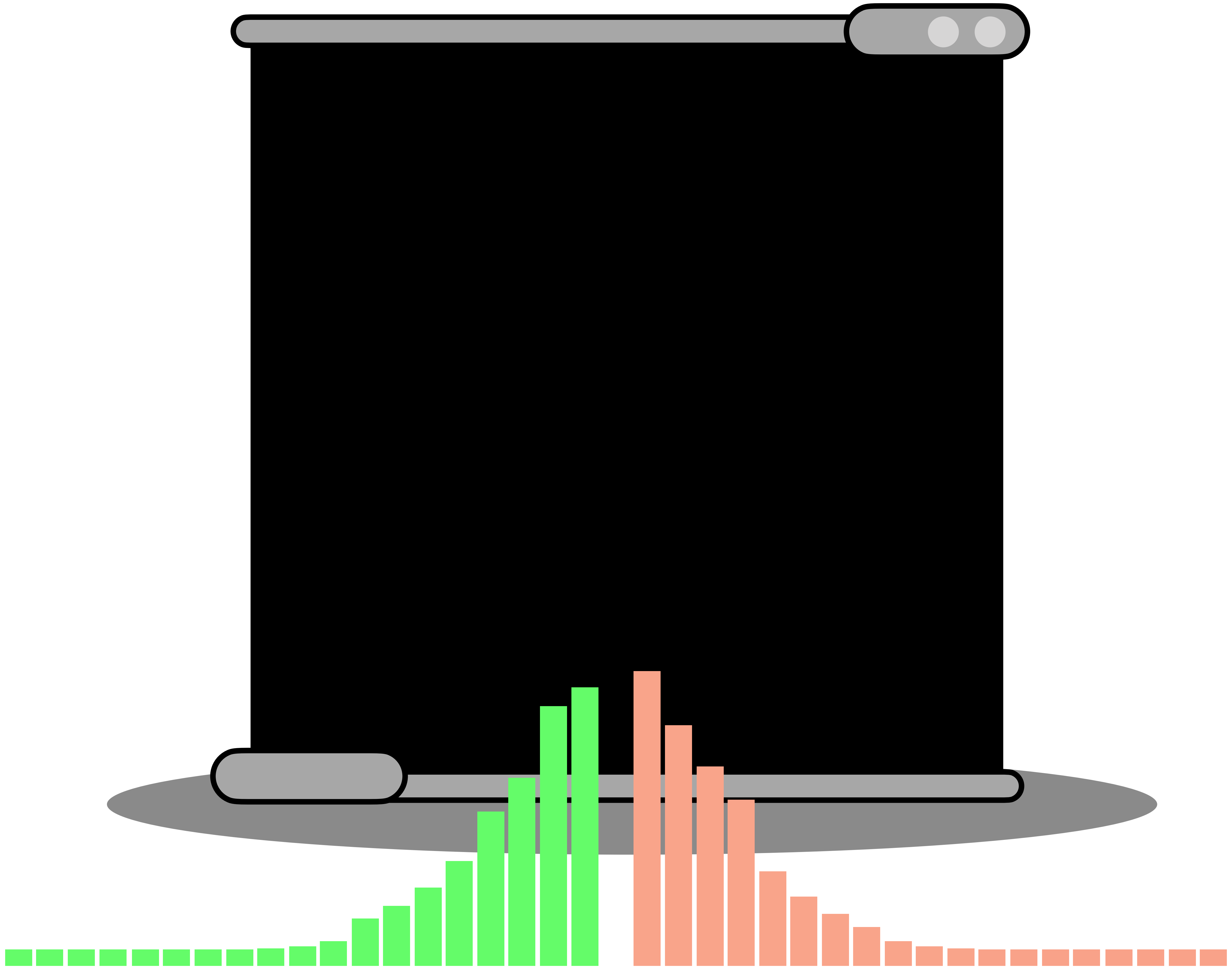


[IEEE S&P 2020] Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-Effects

Each organ contains side-effect features.

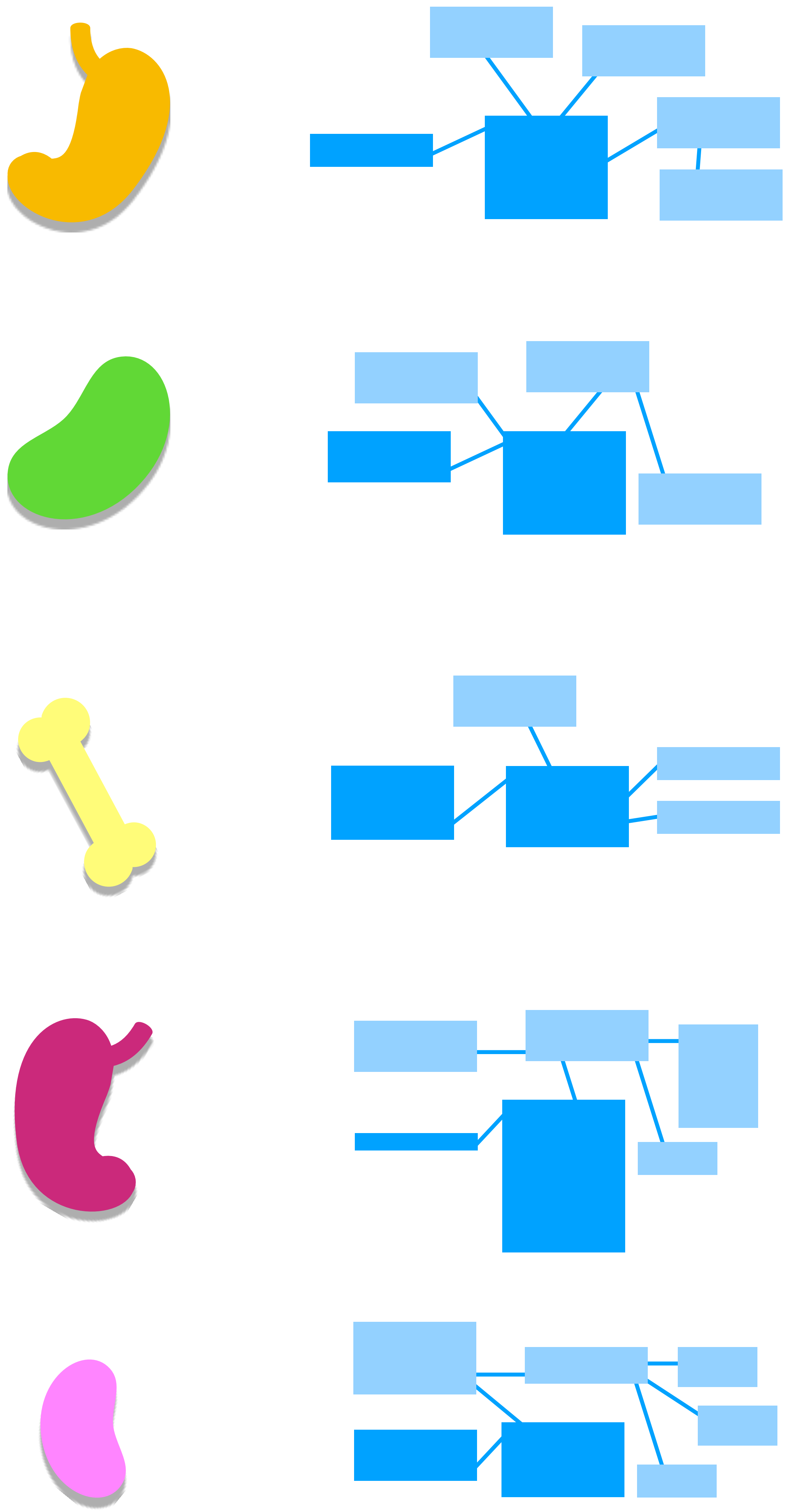


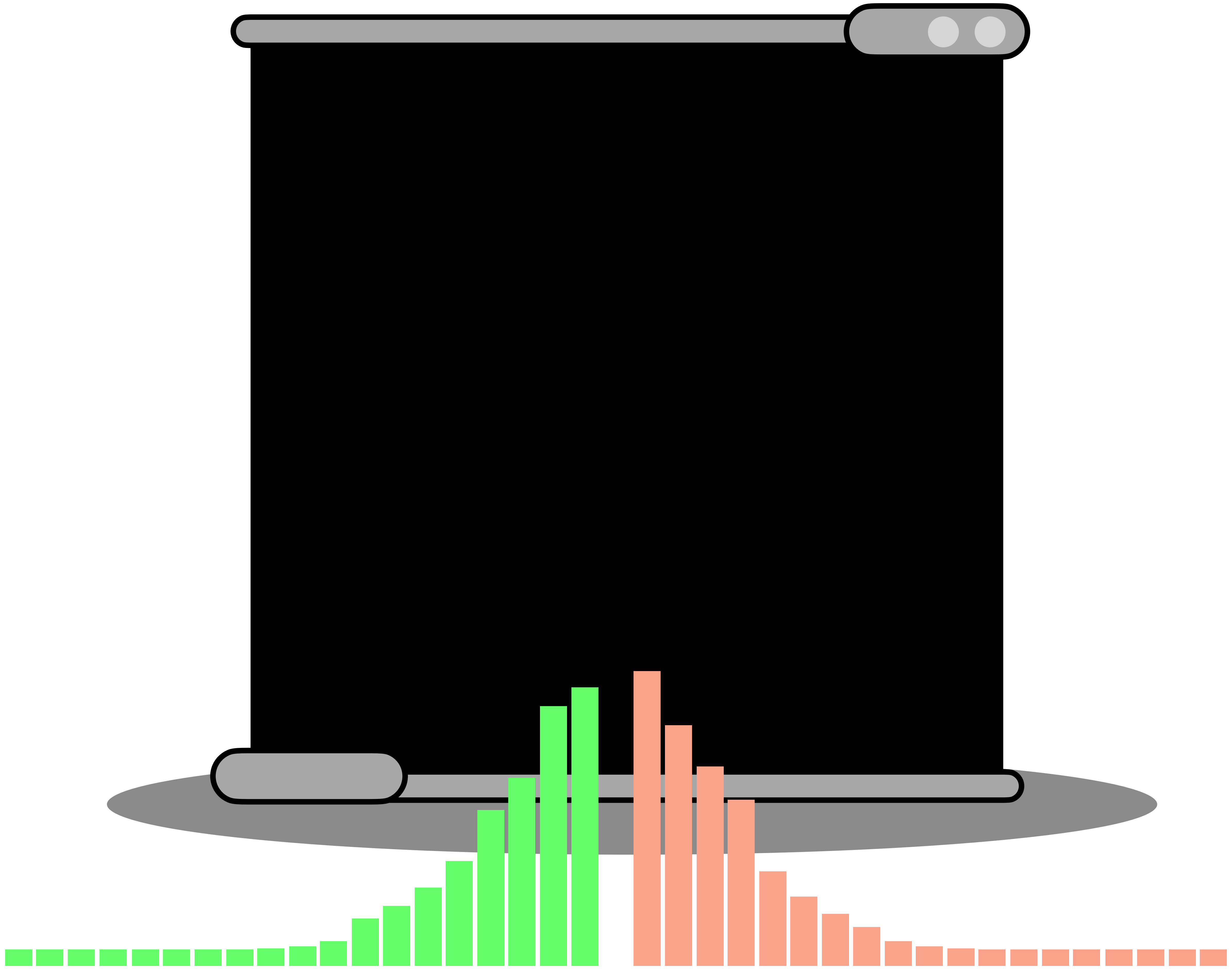


**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-Effects

Each organ contains side-effect features.



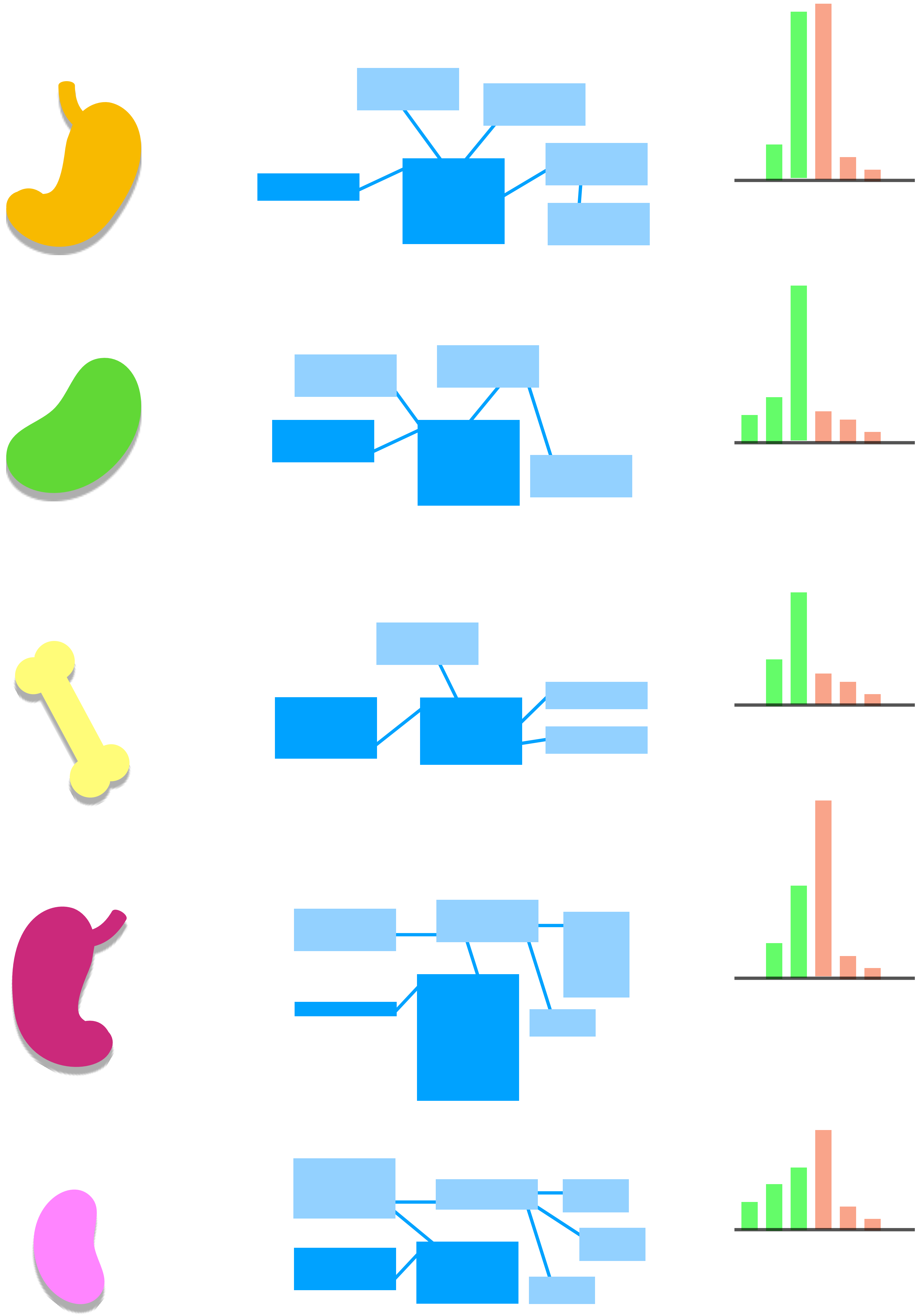


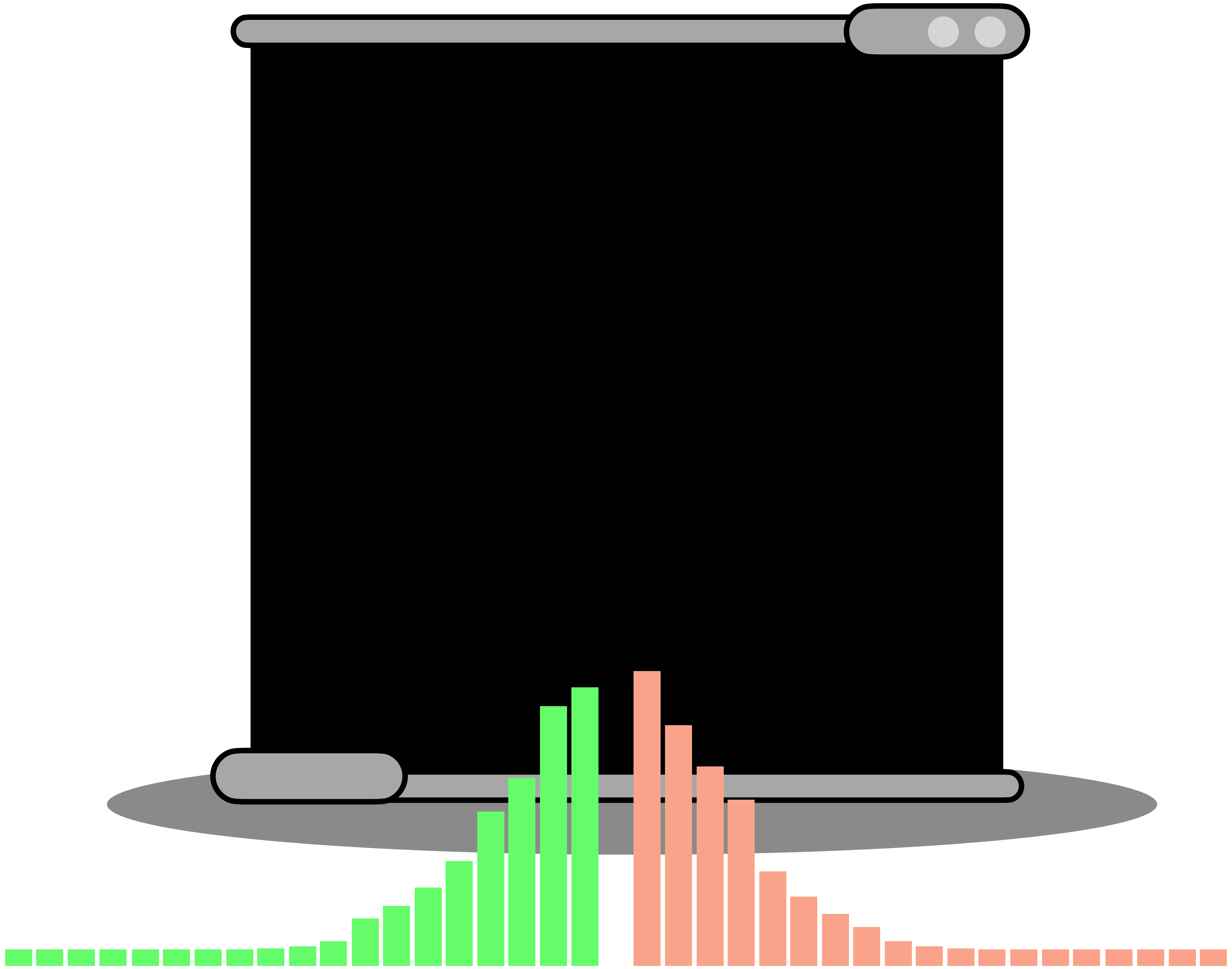
**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-Effects

Each organ contains side-effect features.

We can sum target features, positive, and negative side effects





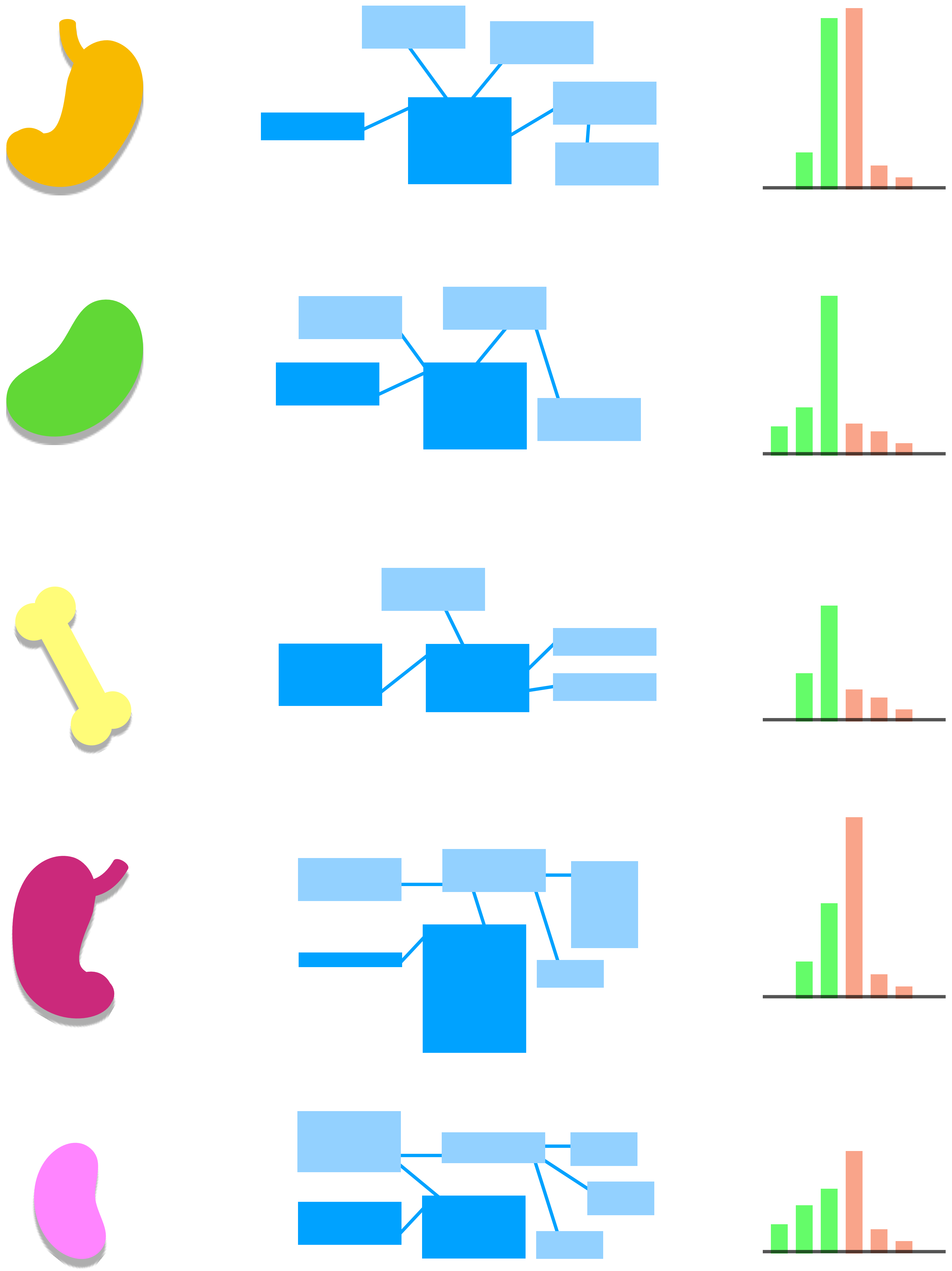
**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space

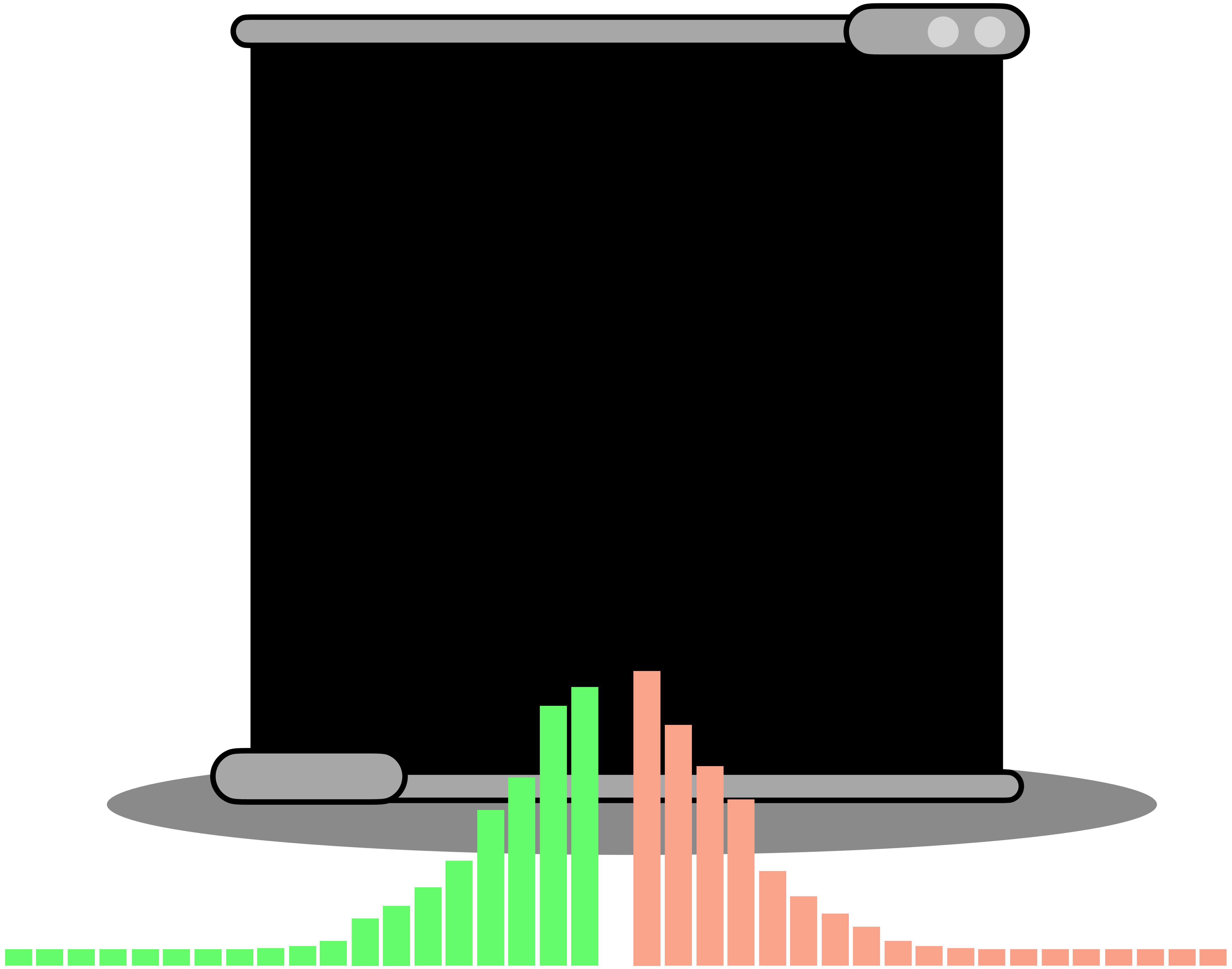
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-Effects

Each organ contains side-effect features.

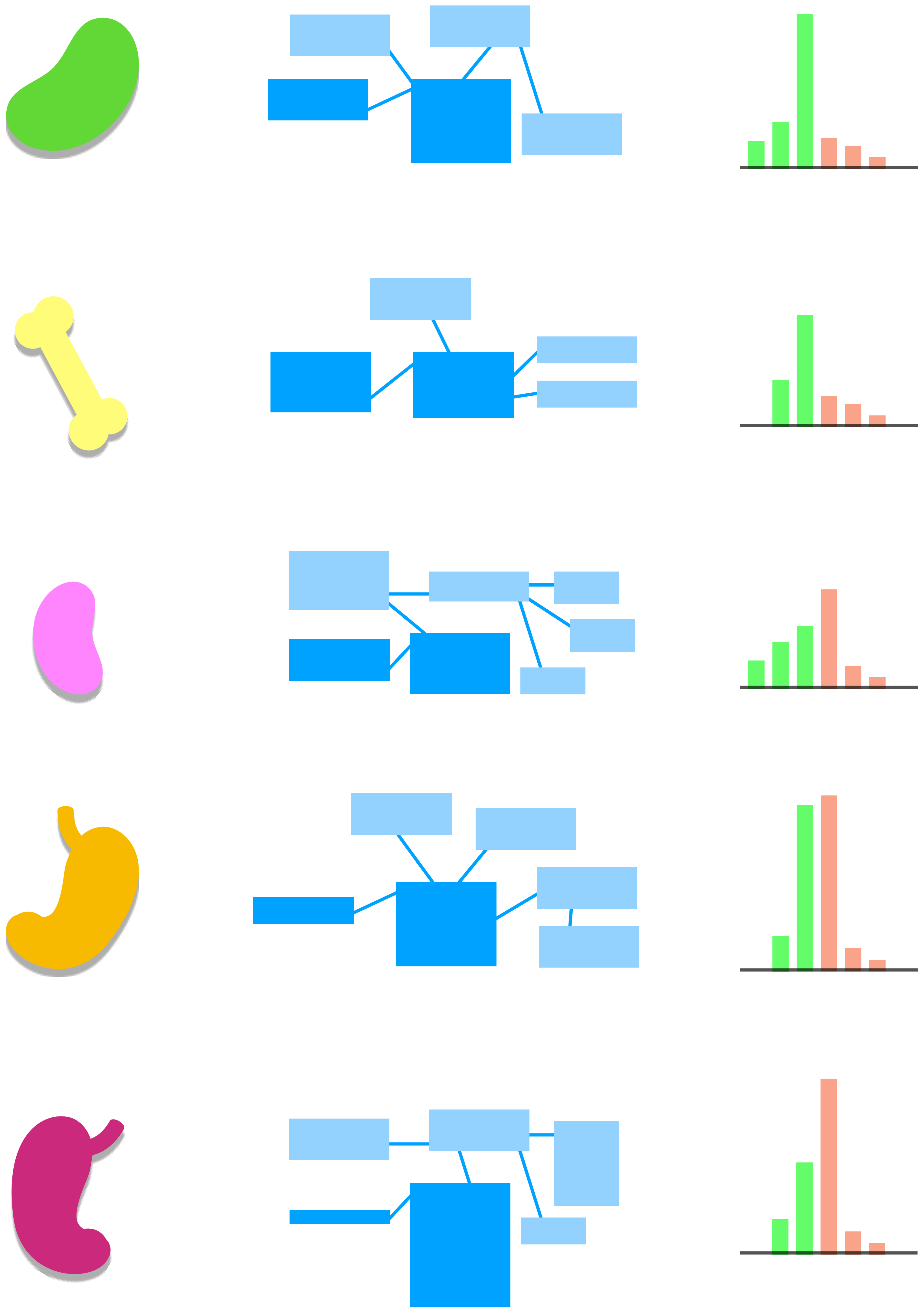
We can sum target features, positive, and negative side effects





**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

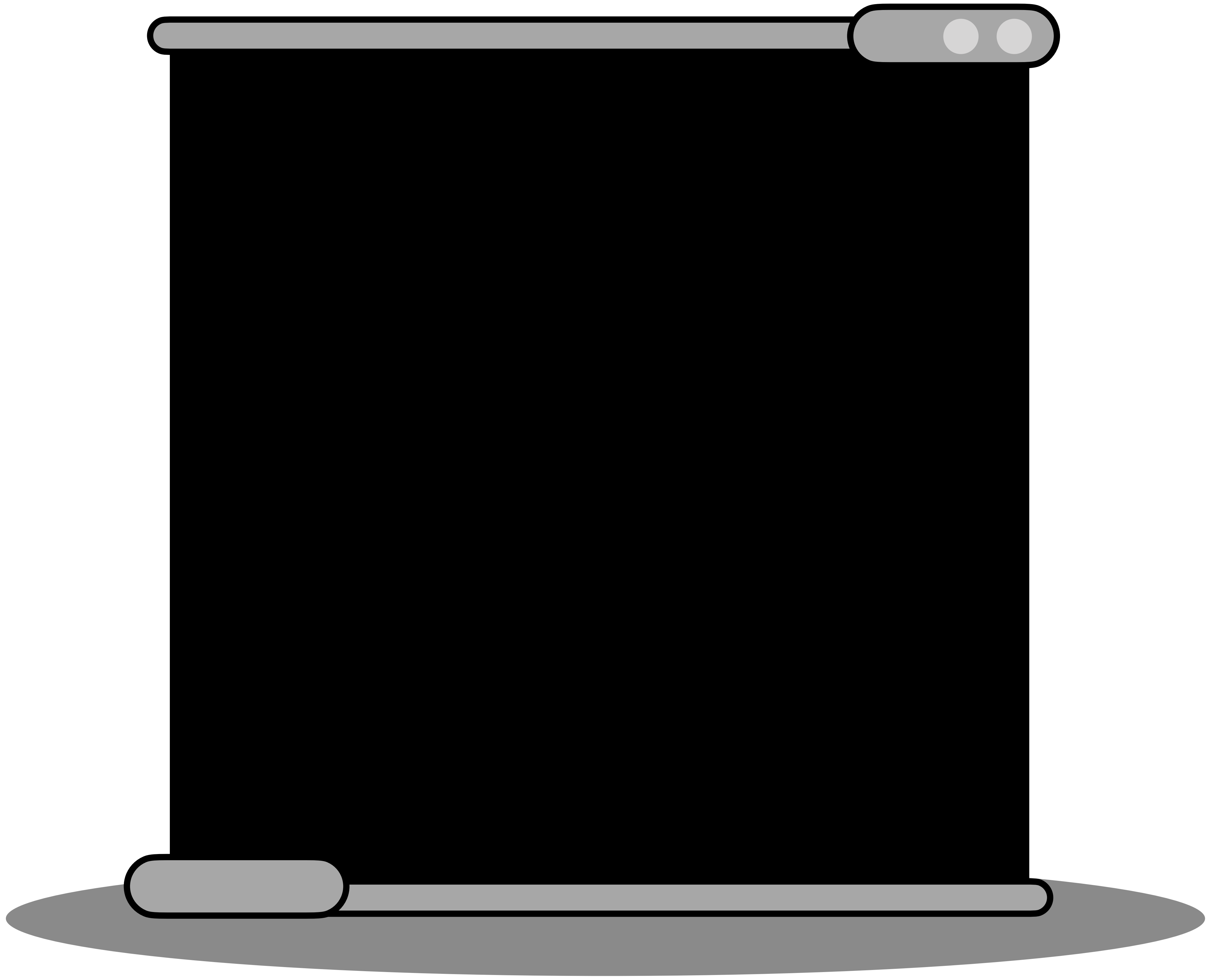
# Side-Effects



Each organ contains side-effect features.

We can sum target features, positive, and negative side effects

to choose organs in order of their overall benign weight



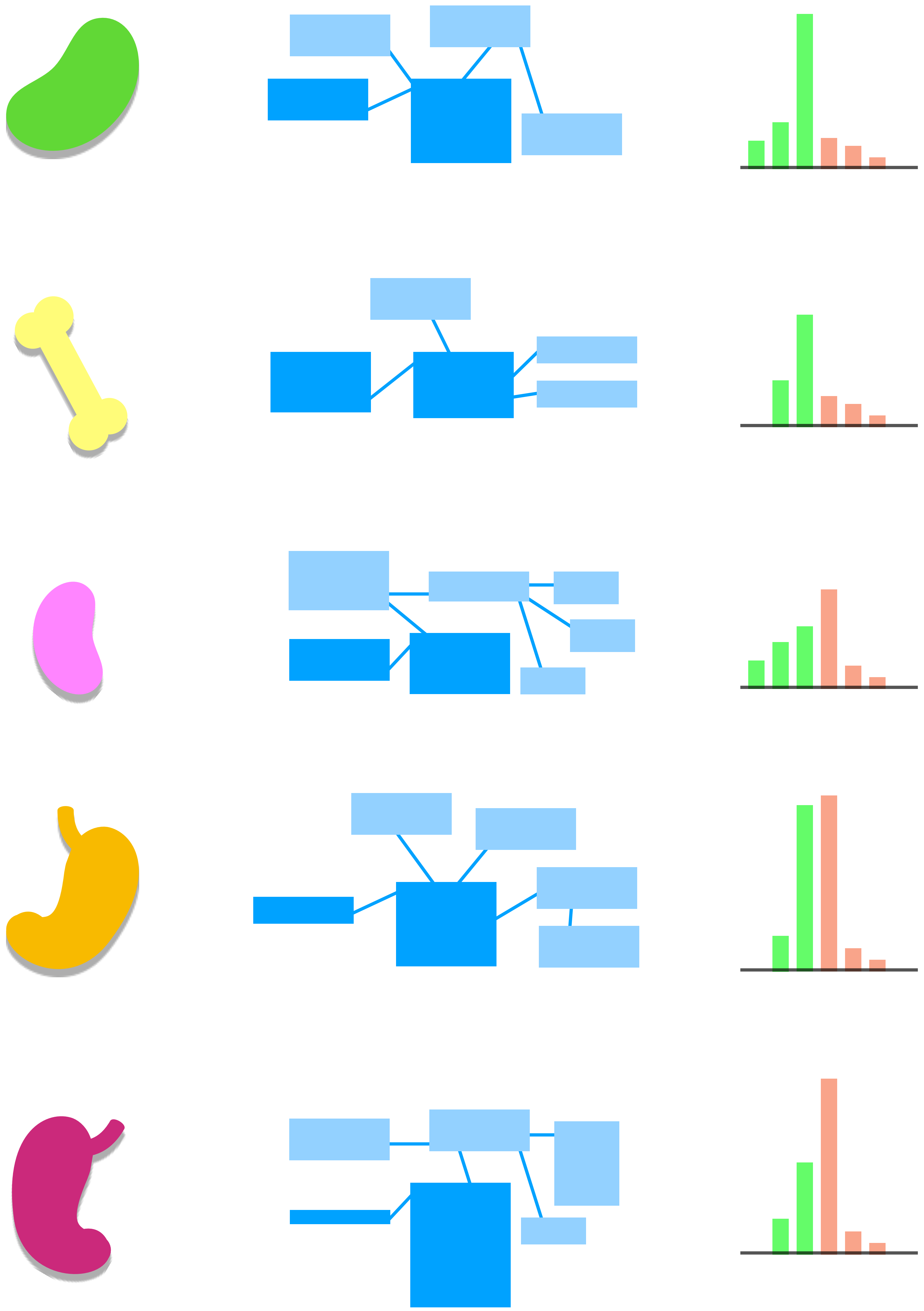
**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

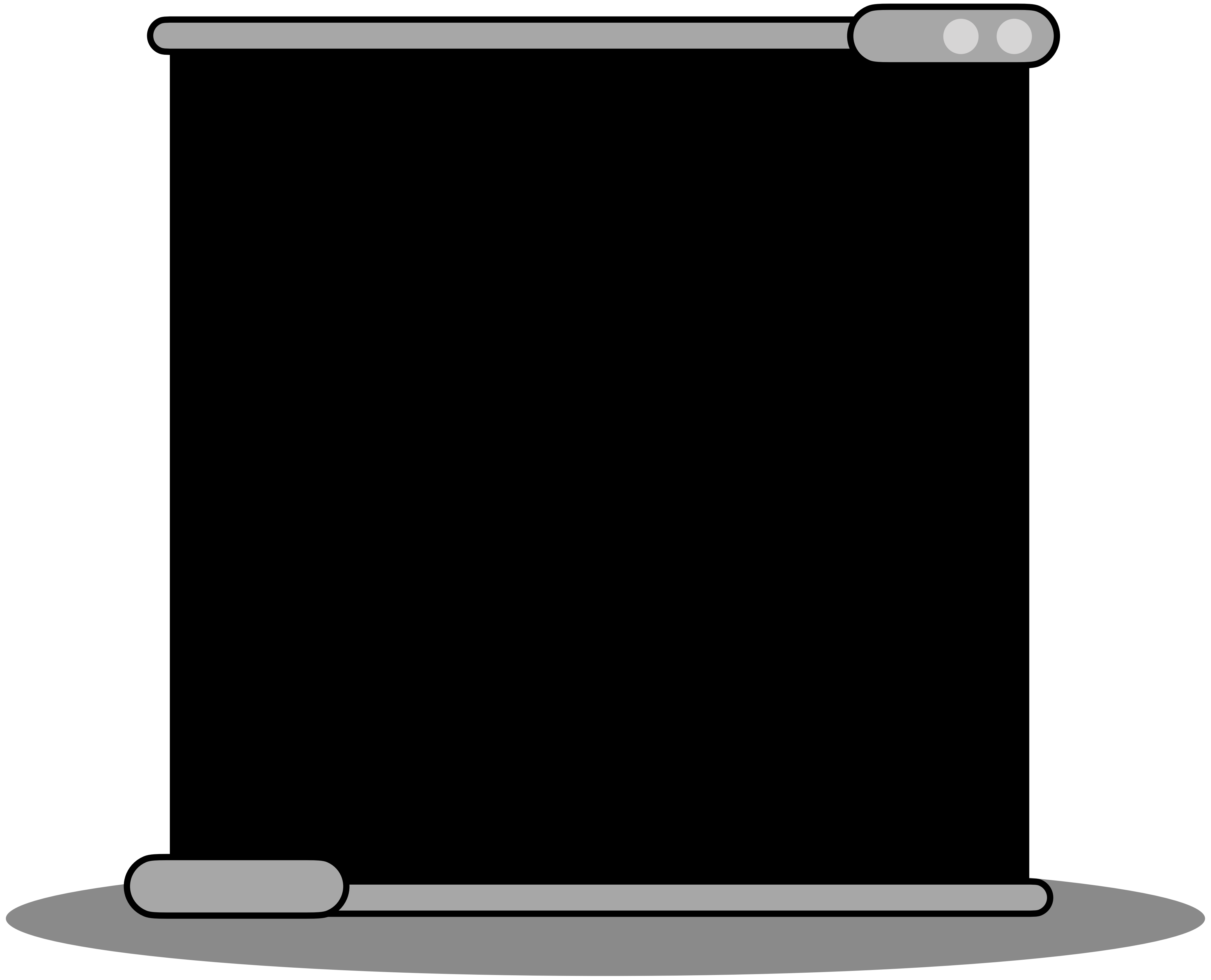
# Side-Effects

Each organ contains side-effect features.

We can sum target features, positive, and negative side effects

to choose organs in order of their overall benign weight





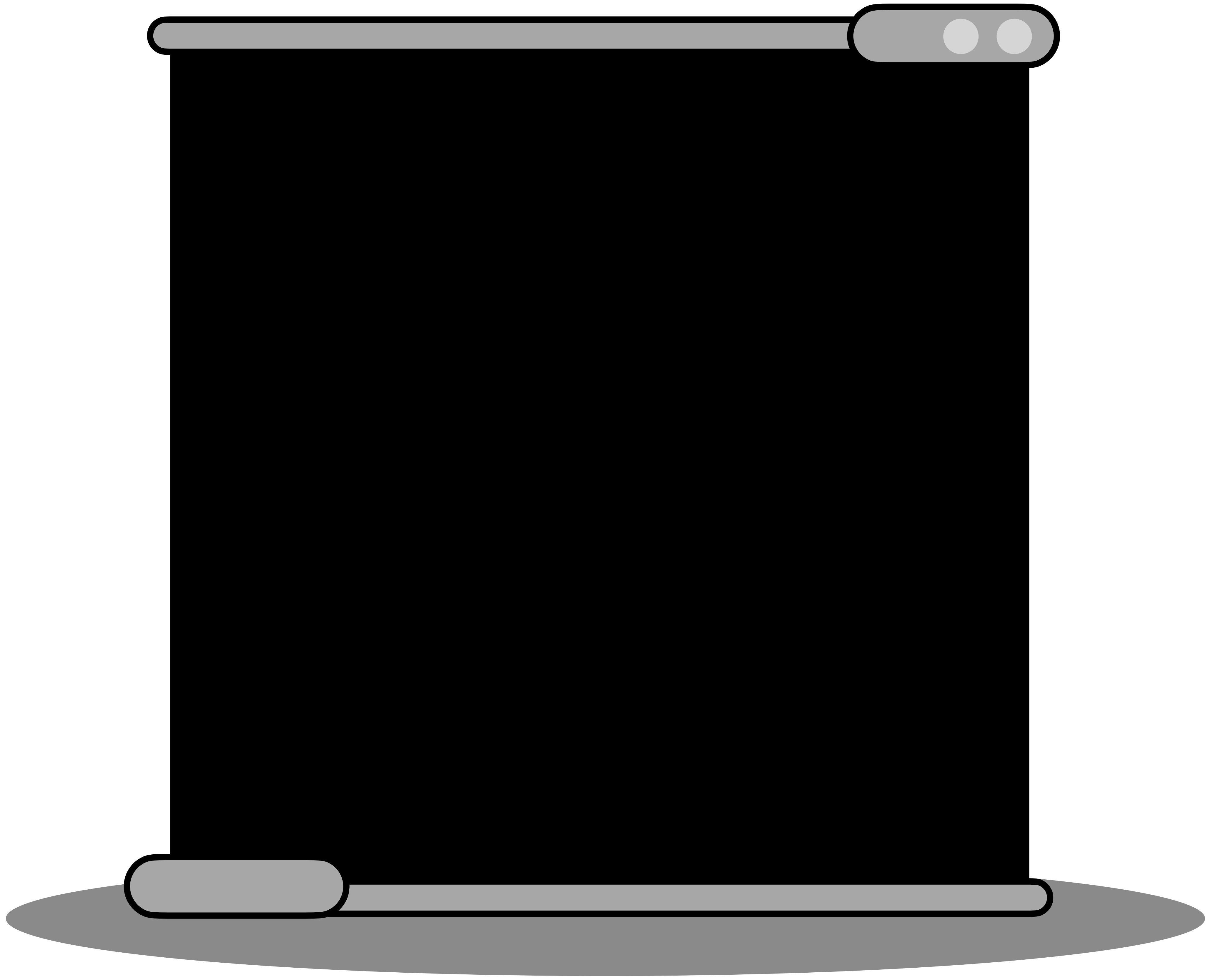
**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing

# Side-Effects

Each organ contains side-effect features.

We can sum target features, positive, and negative side effects

to choose organs in order of their overall benign weight



**[IEEE S&P 2020]** Intriguing Properties of Adversarial ML Attacks in the Problem Space
https://s2lab.cs.ucl.ac.uk/projects/intriguing