

---

# Logic Extraction for Explainable AI

Susmit Jha

Computer Science Laboratory

SRI

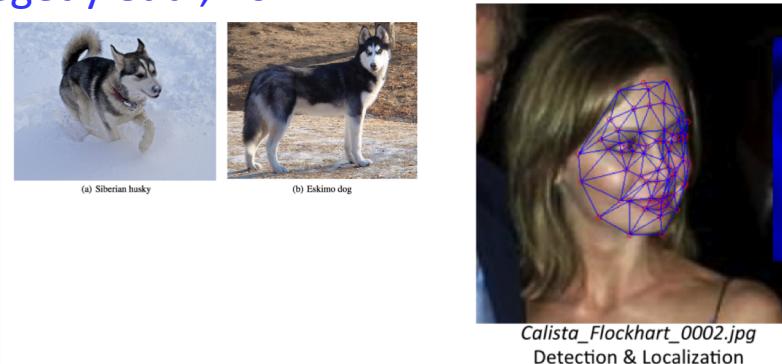
July, 2019

---

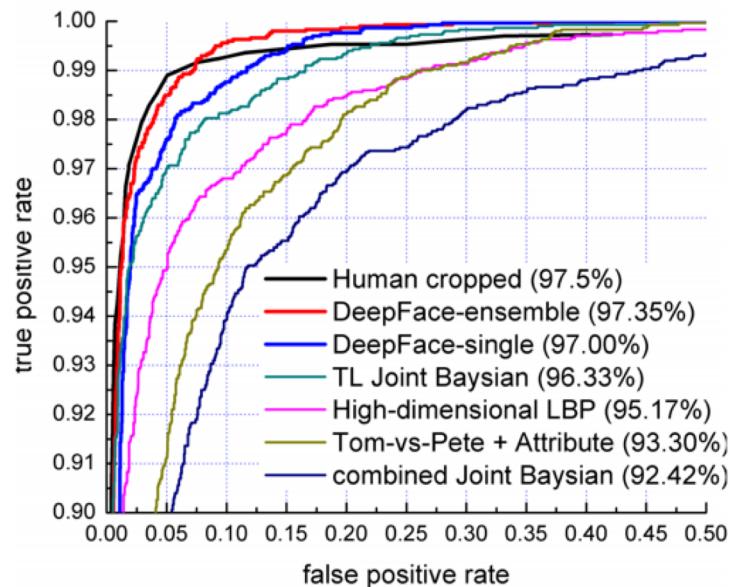
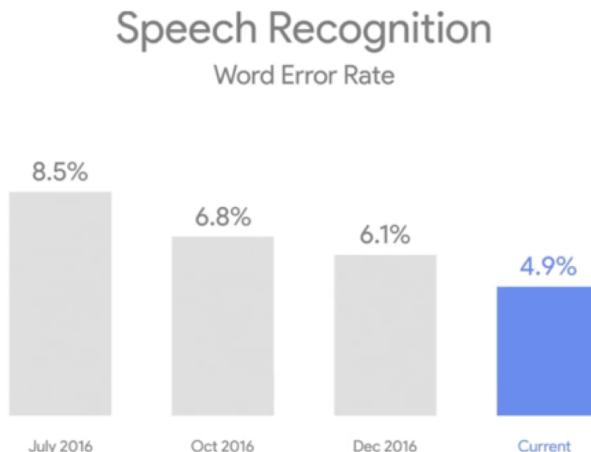
# AI reaches human-level accuracy on benchmark datasets

Going deeper with convolutions. (Inception) C Szegedy et al, 2014

Team	Year	Place	Error (top-5)	Uses external data
SuperVision	2012	1st	16.4%	no
SuperVision	2012	1st	15.3%	Imagenet 22k
Clarifai	2013	1st	11.7%	no
Clarifai	2013	1st	11.2%	Imagenet 22k
MSRA	2014	3rd	7.35%	no
VGG	2014	2nd	7.32%	no
GoogLeNet	2014	1st	6.67%	no



Switchboard  
benchmark

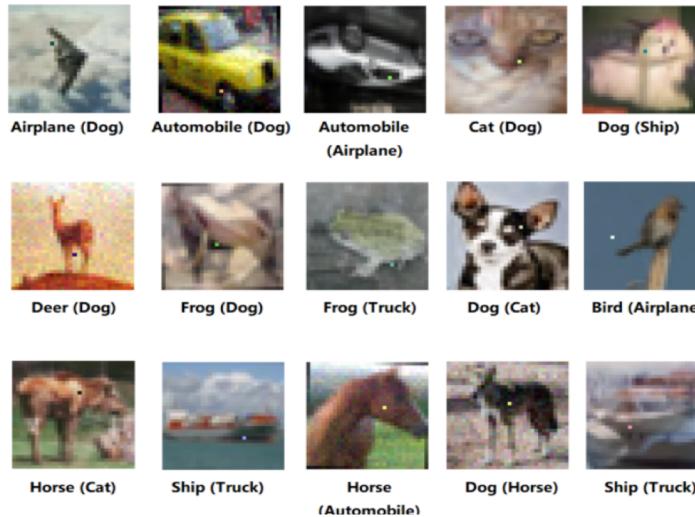


Face Detection. Taigman et al, 2014

# Beyond aggregate numbers

Machine learning very susceptible  
to adversarial attacks.

Szegedy et al, 2013, 2014

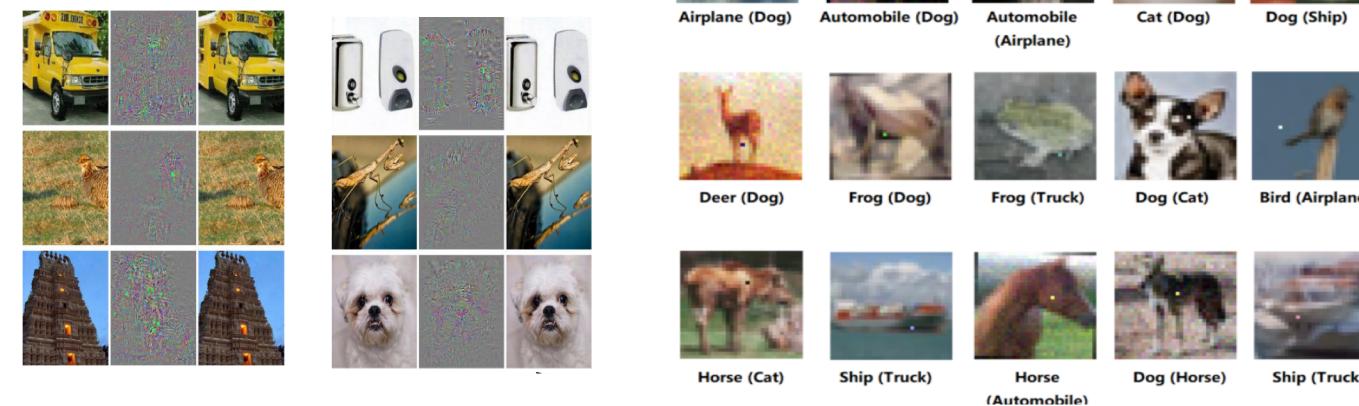


Only allowed to modify the value of 1 pixel.  
70.97% of the natural images can be perturbed to at least one target class by modifying just one pixel with 97.47% confidence on average.

# Beyond aggregate numbers

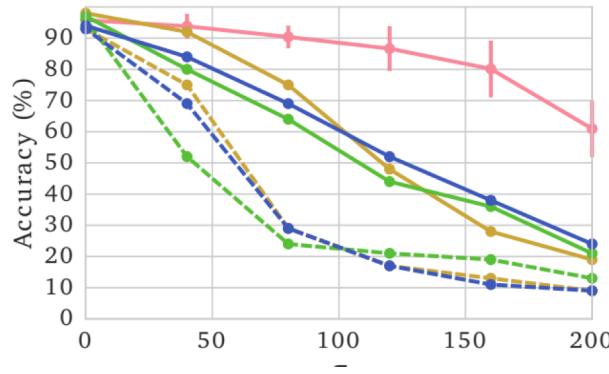
Machine learning very susceptible  
to adversarial attacks.

Szegedy et al, 2013, 2014



Only allowed to modify the value of 1 pixel.  
70.97% of the natural images can be perturbed to at least one target class by modifying just one pixel with 97.47% confidence on average.

Actual Class	$x$	$\tilde{x}$	$\text{pred}_{\text{human}}(\tilde{x})$	$\text{pred}_{\text{VGG16}}(\tilde{x})$
Basenji			Basenji	Yorkshire Terrier
Dalmatian			Dalmatian	Basenji
Border Collie			Border Collie	German Shepherd
Staffordshire Bullterrier			Staffordshire Bullterrier	Border Collie

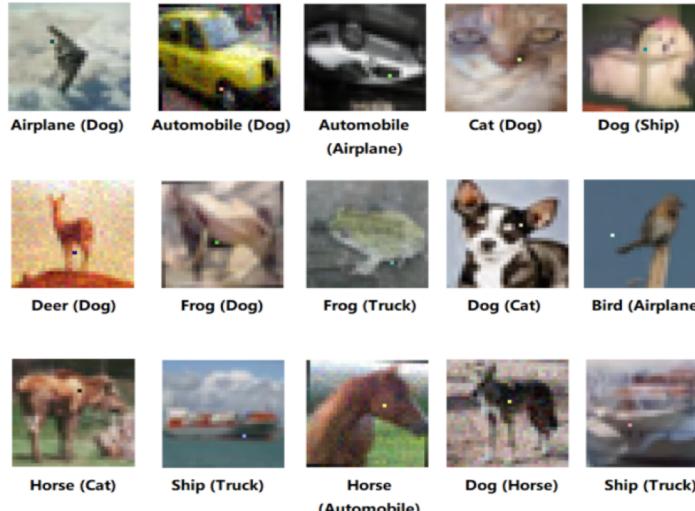


Low robustness to  
benign noise Dodge et al. 2017

# Beyond aggregate numbers

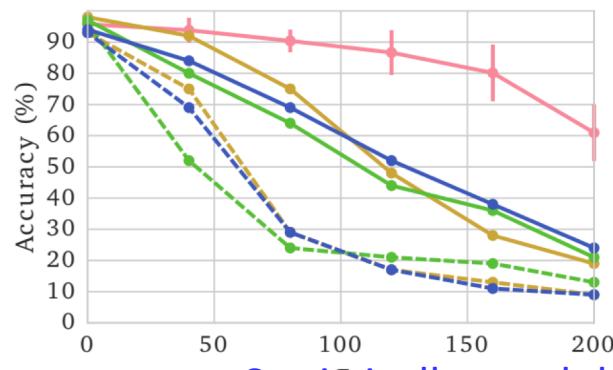
Machine learning very susceptible  
to adversarial attacks.

Szegedy et al, 2013, 2014



Only allowed to modify the value of 1 pixel.  
70.97% of the natural images can be perturbed to at least one target class by modifying just one pixel with 97.47% confidence on average.

Actual Class	$x$	$\tilde{x}$	$\text{pred}_{\text{human}}(\tilde{x})$	$\text{pred}_{\text{VGG16}}(\tilde{x})$
Basenji			Basenji	Yorkshire Terrier
Dalmatian			Dalmatian	Basenji
Border Collie			Border Collie	German Shepherd
Staffordshire Bullterrier			Staffordshire Bullterrier	Border Collie



Low robustness to  
benign noise Dodge et al. 2017

Statistically good doesn't mean logically/conceptually good.

Understanding deep learning requires rethinking generalization.  
C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals

# TRINITY: Trust, Resilience and Interpretability

## Trust

- Global Assume/Guarantee Contracts on DNNs
- Closed-loop verification of NN controllers
- Extracting and Integrating Temporal Logic into Learned Control



## Resilience

- Adversarial Robustness

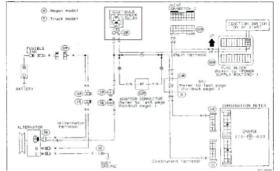
## Interpretability

- Explaining Decisions as Sparse Boolean Formula Learning
- Inverse Reinforcement Learning of Temporal Specifications

# TRINITY: Trust, Resilience and Interpretability

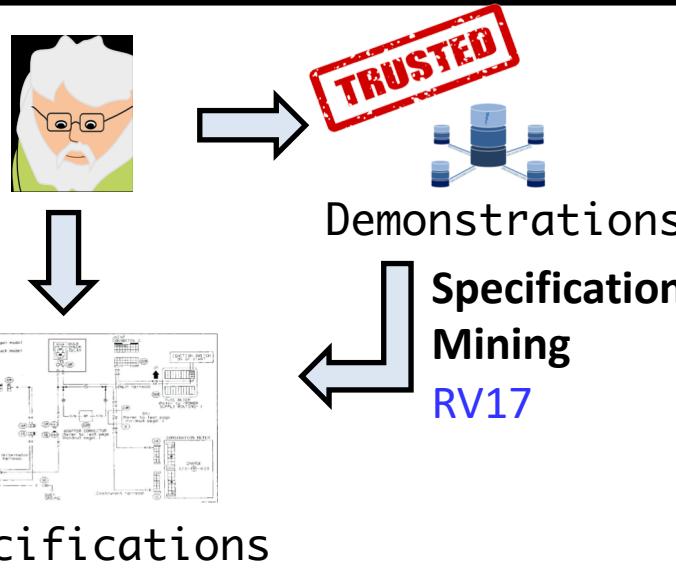


Demonstrations

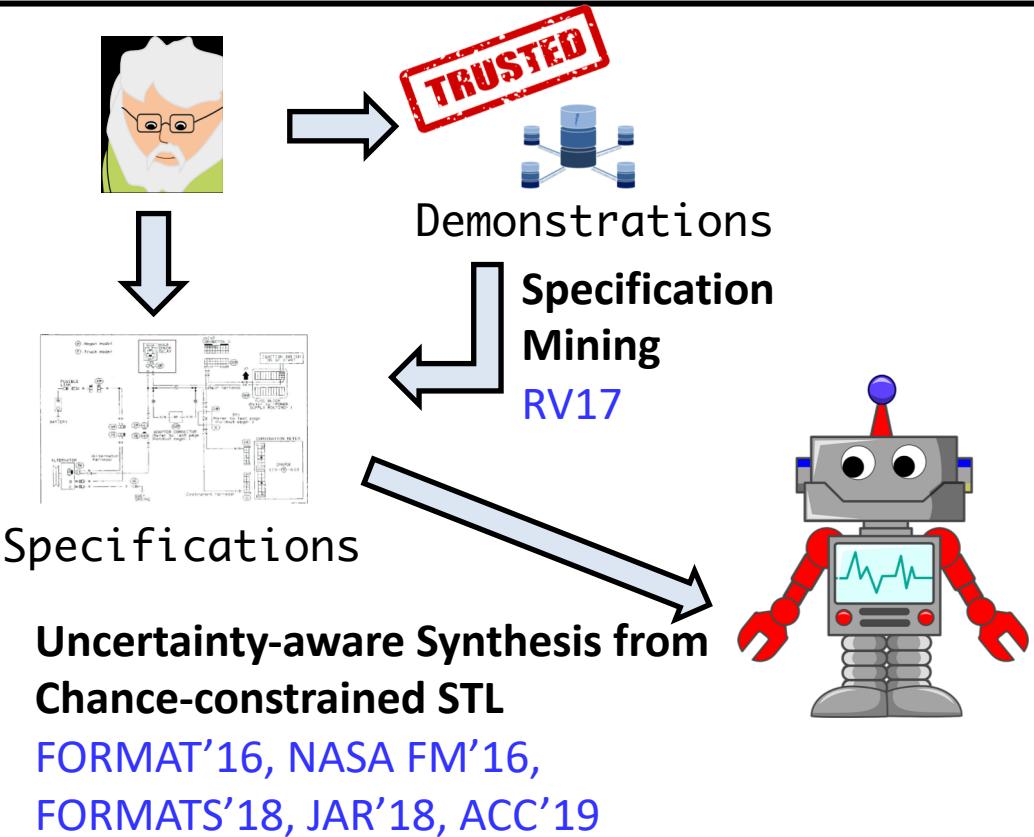


Specifications

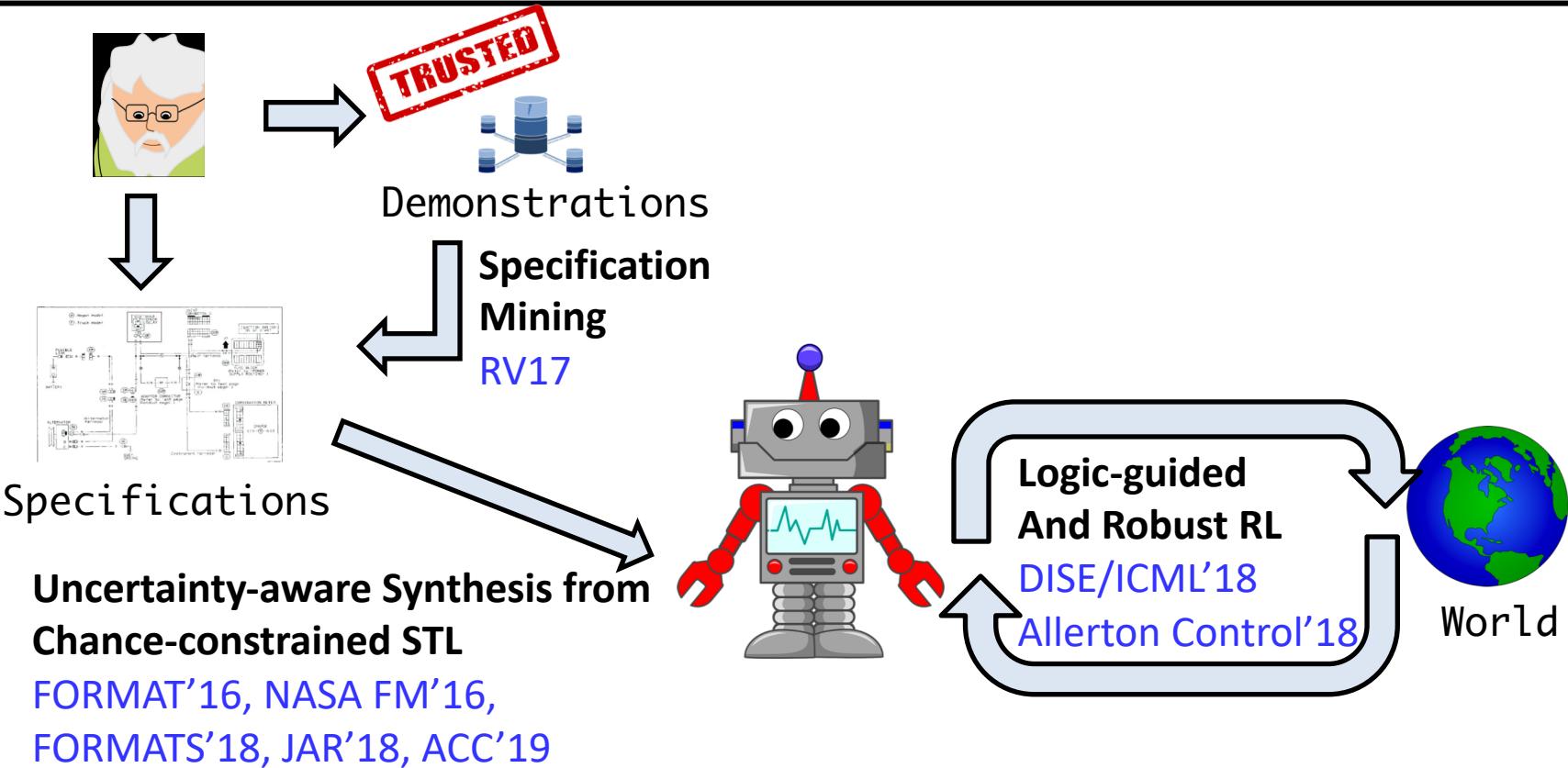
# TRINITY: Trust, Resilience and Interpretability



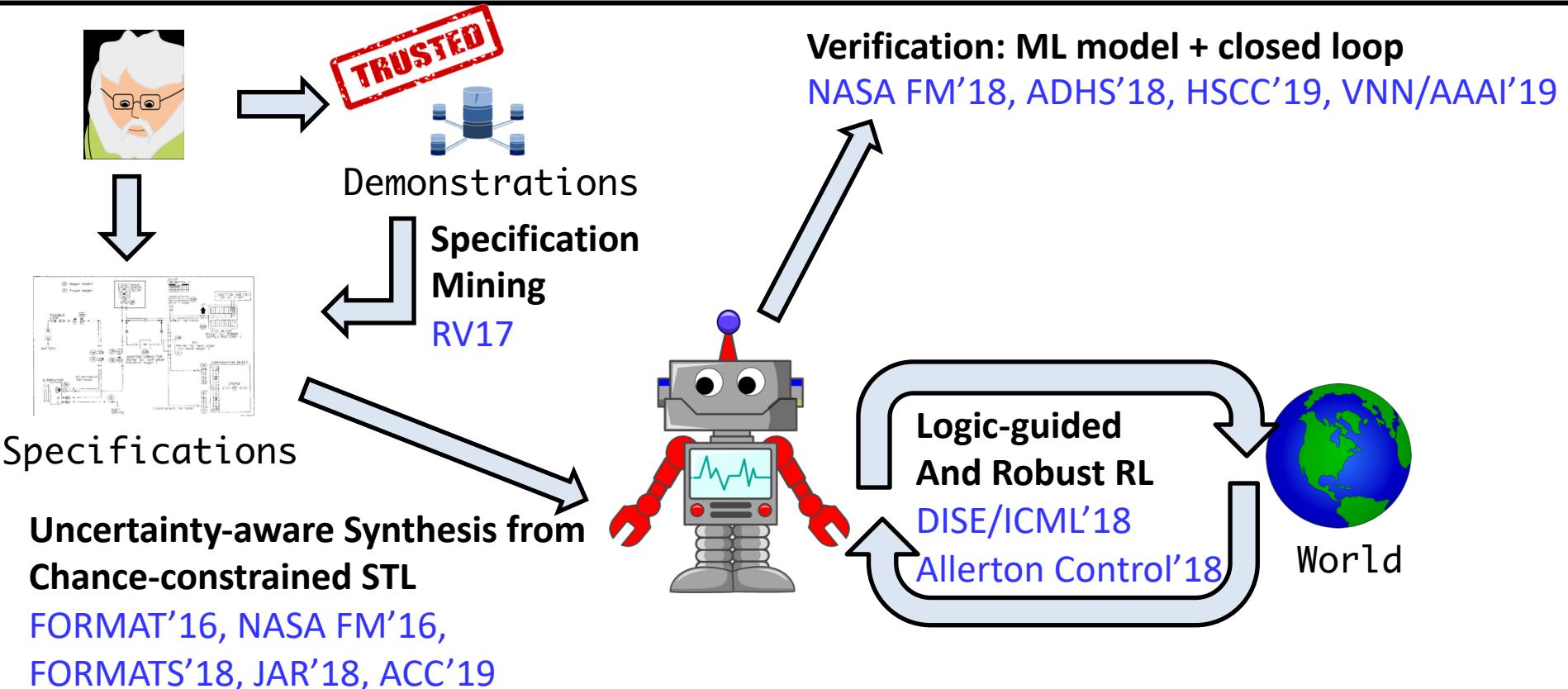
# TRINITY: Trust, Resilience and Interpretability



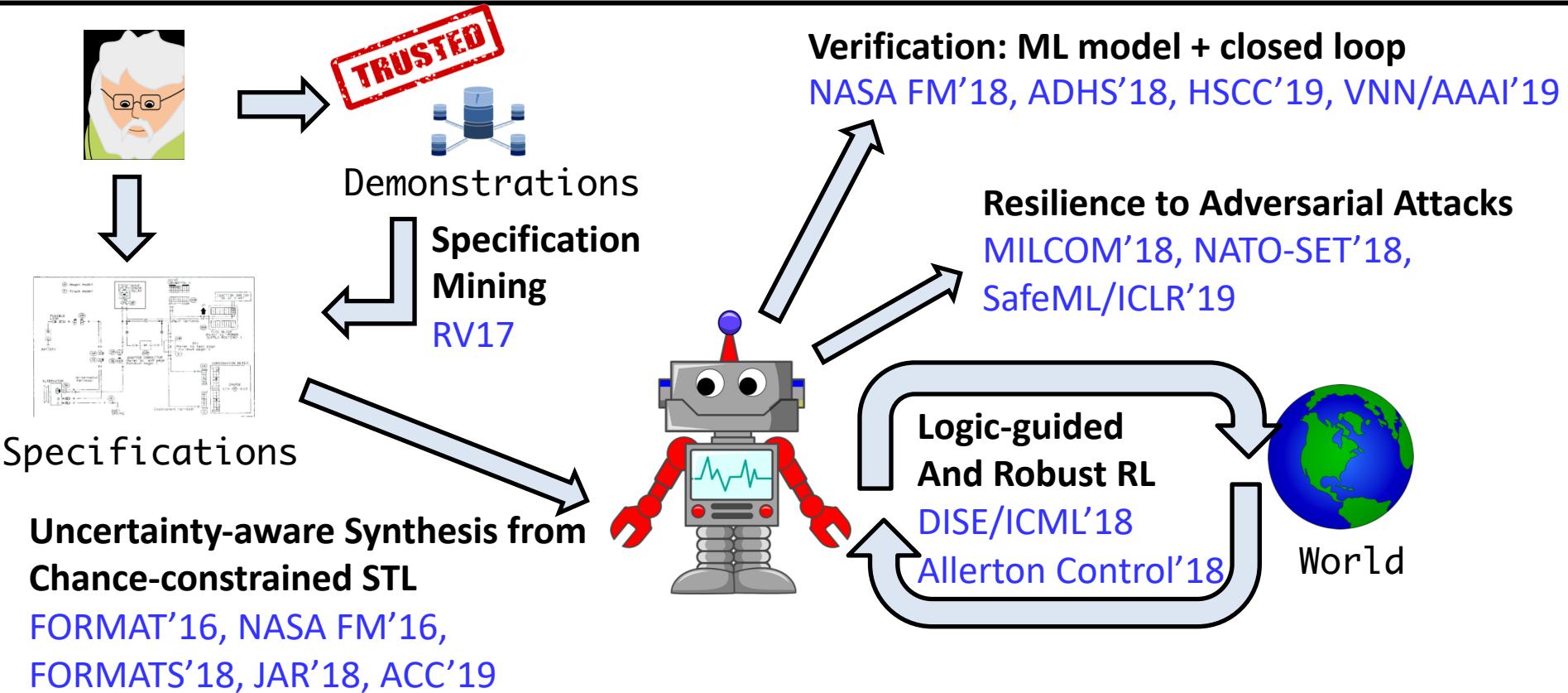
# TRINITY: Trust, Resilience and Interpretability



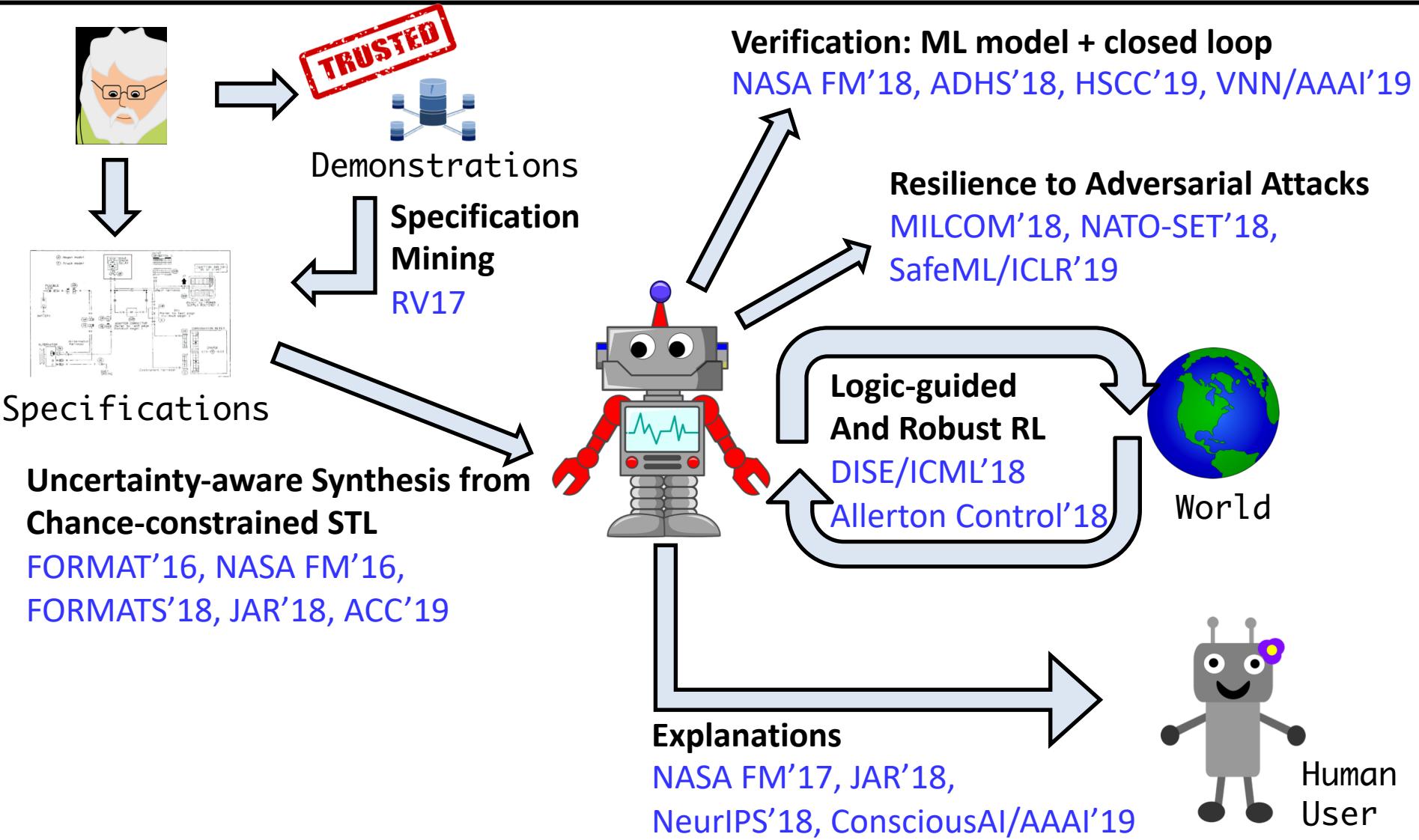
# TRINITY: Trust, Resilience and Interpretability



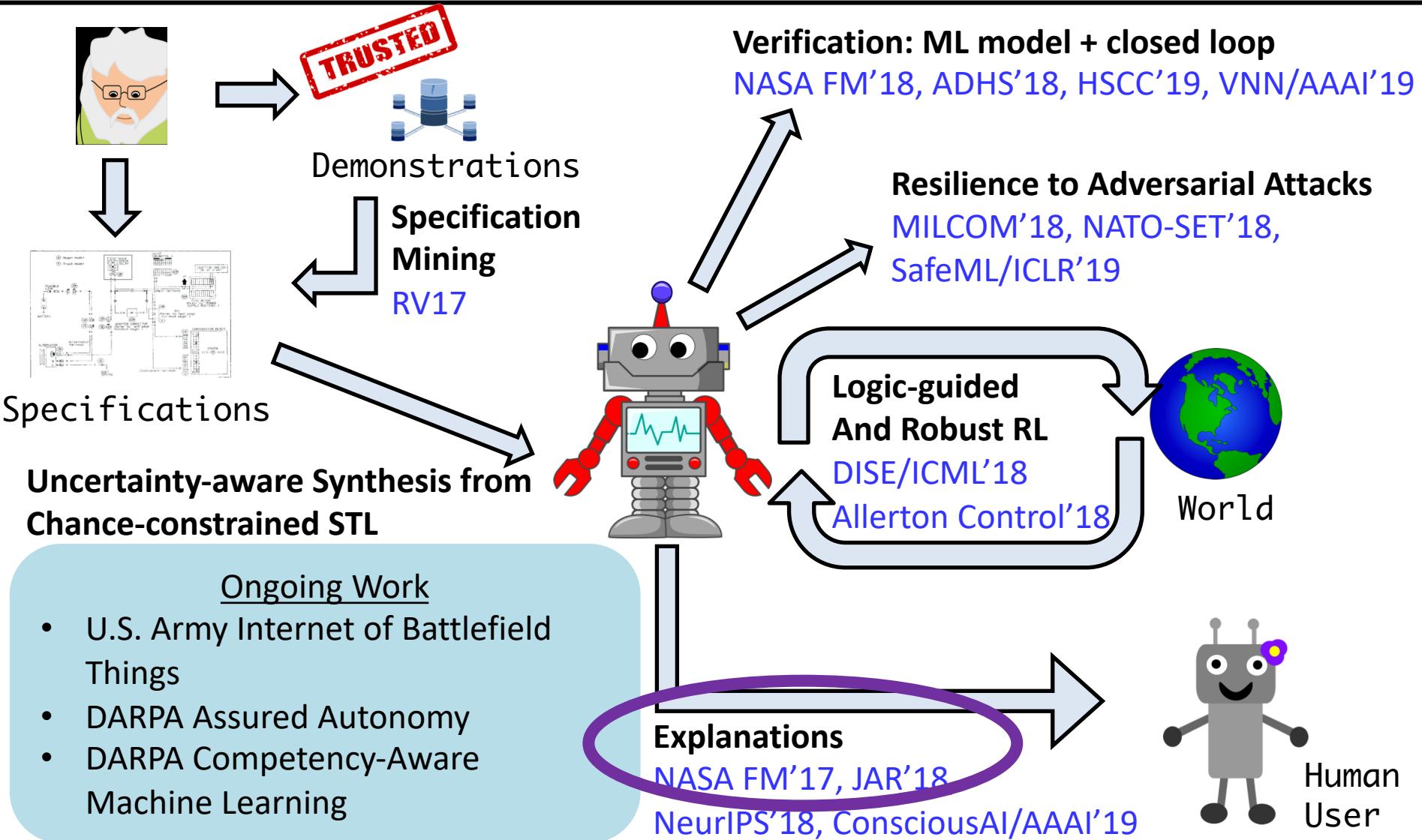
# TRINITY: Trust, Resilience and Interpretability



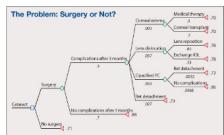
# TRINITY: Trust, Resilience and Interpretability



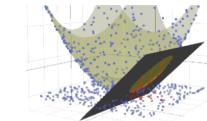
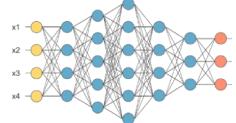
# TRINITY: Trust, Resilience and Interpretability



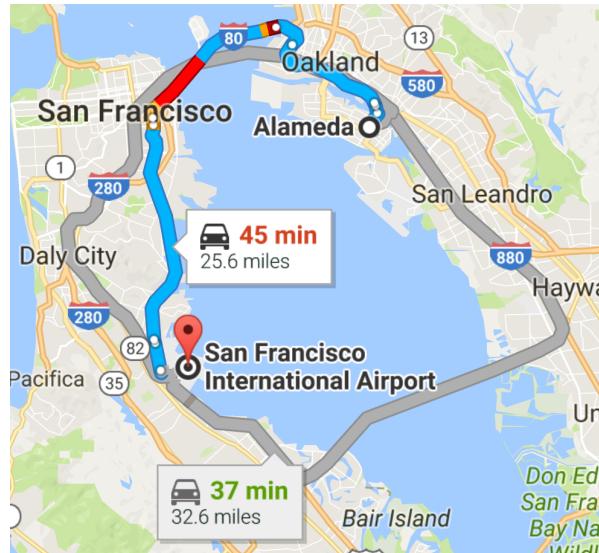
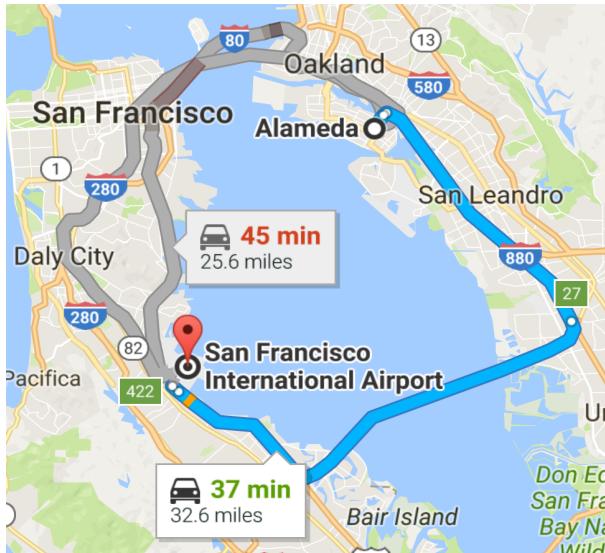
# Need for explanation



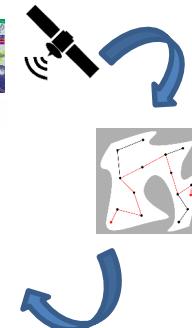
Interpretable but less scalable:  
Decision Trees, Linear Regression



Scalable but less interpretable :  
Neural Networks, Support Vector  
Machines



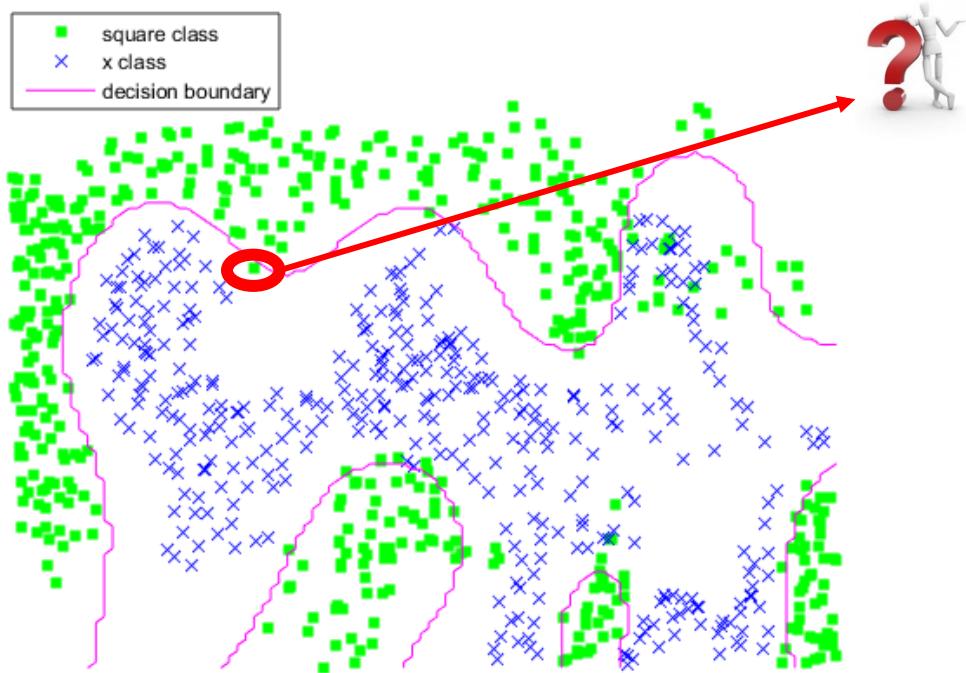
- This route is faster.
- There is traffic on Bay Bridge.
- There is an accident just after Bay Bridge backing up traffic.



Why did we take the San Mateo bridge instead of the Bay Bridge ?

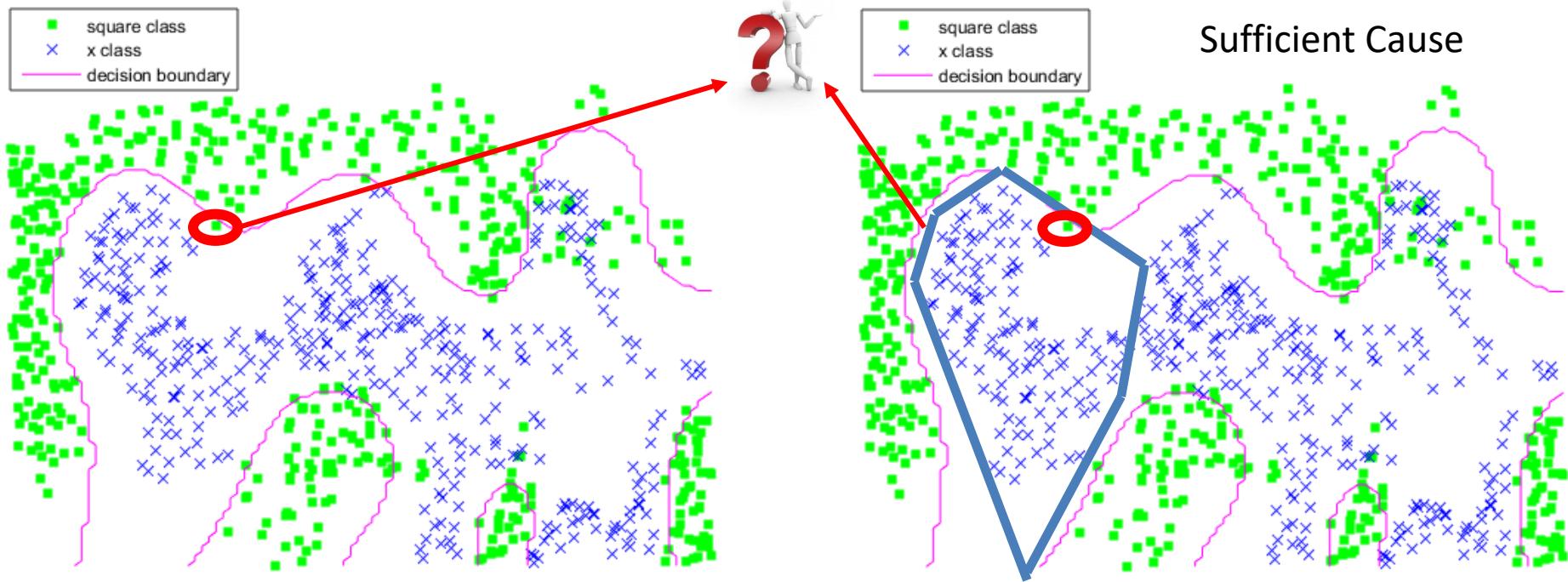
# Local Explanations of Complex Models

Not reverse engineering an ML model but finding explanation locally for one decision.



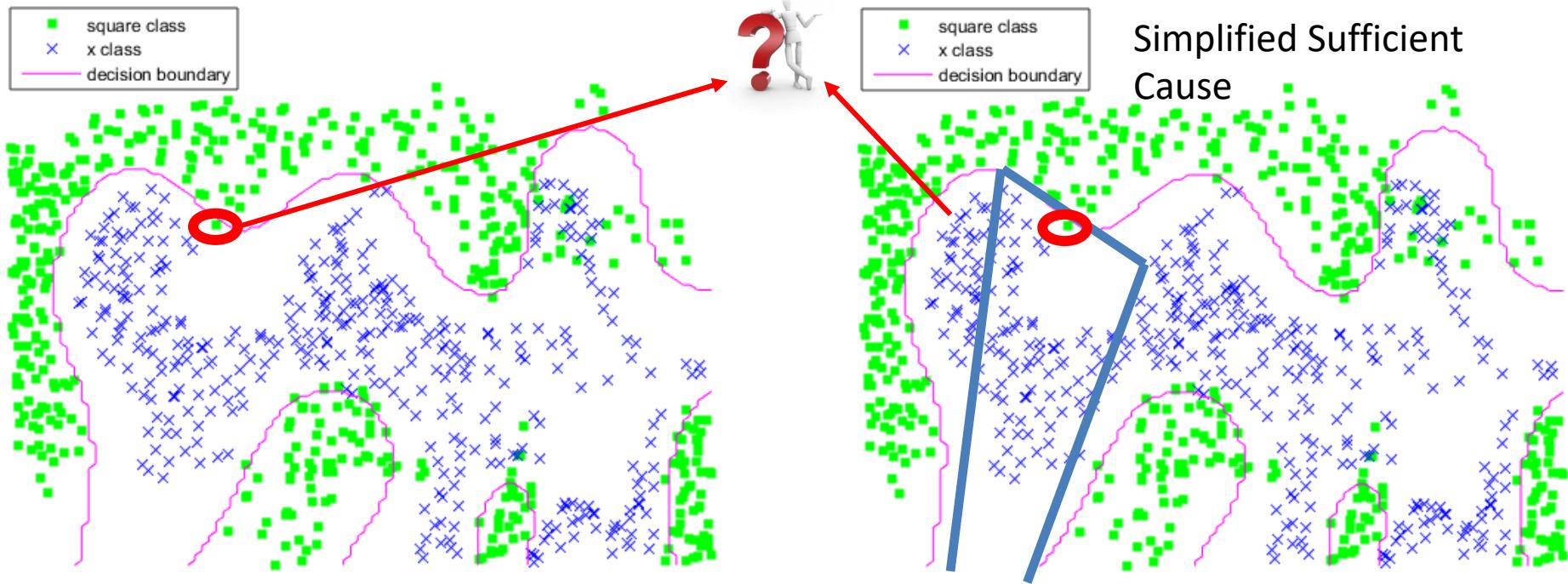
# Local Explanations of Complex Models

Not reverse engineering an ML model but finding explanation locally for one decision.



# Local Explanations of Complex Models

Not reverse engineering an ML model but finding explanation locally for one decision.



# Local Explanations in AI

Not reverse engineering an ML model but finding explanation locally for one decision.

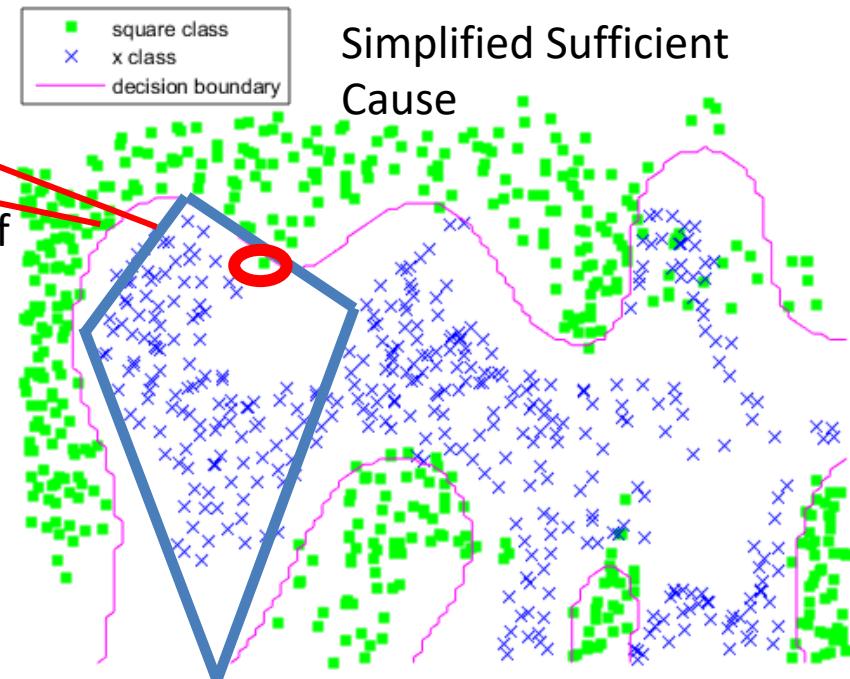
$$\xi(x) = \operatorname{argmin}_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g)$$

$\mathcal{L}(f, g, \pi_x)$  Measure of how well  $g$  approximates  $f$

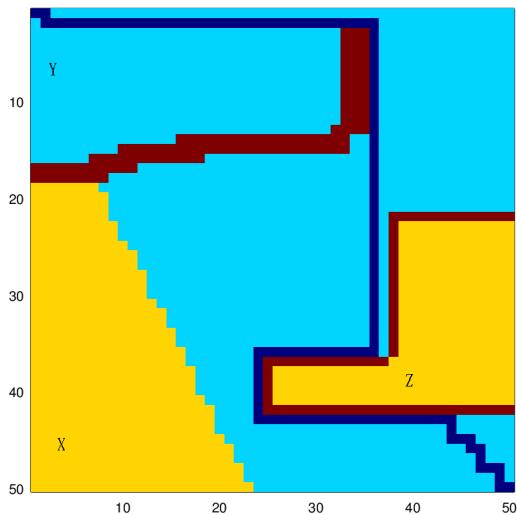
$\Omega(g)$  Measure of complexity of  $g$

Formulation in AI:

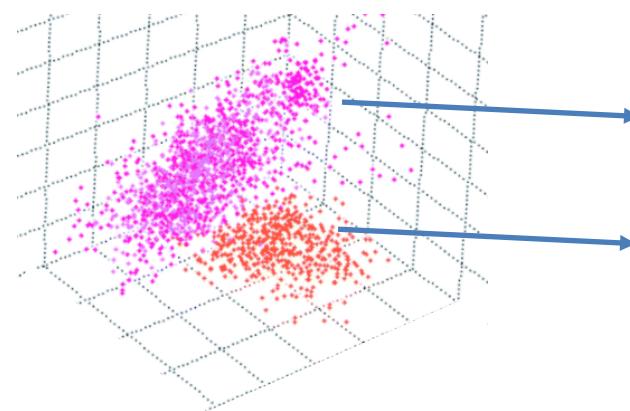
- Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?: Explaining the Predictions of Any Classifier." *International Conference on Knowledge Discovery and Data Mining*. ACM, 2016.
- Hayes, Bradley, and Julie A. Shah. "Improving Robot Controller Transparency Through Autonomous Policy Explanation." *International Conference on Human-Robot Interaction*. ACM, 2017.



# Model Agnostic Explanation through Boolean Learning



Why does the path not go through Green?



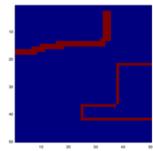
Let each point in k-dimensions (for some k) correspond to a map.

- Maps in which optimum path goes via green
- Maps in which optimum path does not go via green

Find a Boolean formula  $\phi$  such that

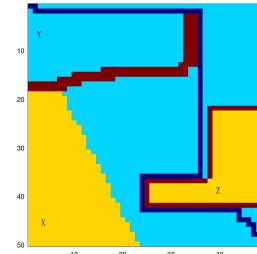
$$\begin{aligned}\phi &\Leftrightarrow \text{Path contain } z \\ \phi &\Rightarrow \text{Path contain } z\end{aligned}$$

# Explanations as Learning Boolean Formula



```
Algorithm 1: A*
  Input: start, goal(xg, yg), arggoal(xg)
  Output: path
  1 if goal(start) = true then return makePath(start)
  2
  3 open ← {start}
  4 closed ← {}
  5 while open ≠ ∅ do
  6   sort(open)
  7   n ← open.pop()
  8   if n = goal then
  9     forall the kid ∈ kids do
 10      if n.f < (kid.f + 1) - kid.d
 11        | if goal(kid) = true then return makePath(kid)
 12        | if kid.f > closed then open ← open ∪ kid
 13   closed ← n
 14 return ?
```

A\*



$\phi_{explain}$  :

Using explanation vocabulary

Ex: Obstacle presence

$\phi_{query}$  :

Some property of the output

Ex: Some cells not selected

$$\begin{aligned}\phi_{explain} &\Rightarrow \phi_{query} \\ \phi_{explain} &\Leftrightarrow \phi_{query}\end{aligned}$$

# How difficult is it? Boolean formula learning

$$\begin{aligned}\Phi_{\text{explain}} &\Rightarrow \Phi_{\text{query}} \\ \Phi_{\text{explain}} &\Leftrightarrow \Phi_{\text{query}}\end{aligned}$$

50x50 grid has  $2^{2^{50 \times 50}}$  possible explanations even if vocabulary only considers presence/absence of obstacles.

Scalability: Usually the feature space or vocabulary is large. For a map, its order of features in the map. For an image, it is order of the image's resolution.

Guarantee: Is the sampled space of maps enough to generate the explanation with some quantifiable probabilistic guarantee?

# How difficult is it? Boolean formula learning

$$\begin{aligned}\Phi_{\text{explain}} &\Rightarrow \Phi_{\text{query}} \\ \Phi_{\text{explain}} &\Leftrightarrow \Phi_{\text{query}}\end{aligned}$$

50x50 grid has  $2^{2^{50 \times 50}}$  possible explanations even if vocabulary only considers presence/absence of obstacles.

Scalability: Usually the feature space or vocabulary is large. For a map, its order of features in the map. For an image, it is order of the image's resolution.

Guarantee: Is the sampled space of maps enough to generate the explanation with some quantifiable probabilistic guarantee?

On PAC learning algorithms for rich Boolean function classes

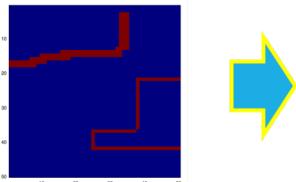
Rocco A. Servedio\*

Department of Computer Science  
Columbia University  
New York, NY U.S.A.  
[rocco@cs.columbia.edu](mailto:rocco@cs.columbia.edu)

Theoretical Result:

Learning Boolean formula even approximately is hard. 3-DNF is not learnable in Probably Approximately Correct framework unless RP = NP.

# Two Key Ideas

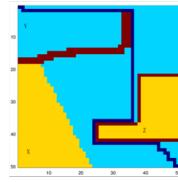


$\phi_{explain}$  :  
Using explanation vocabulary  
Ex: Obstacle presence

```

Algorithm 1: A*
Input: start, goal, N, alpha=0.9
Output: sol
If goal.state = true then return solve(N, start)
else
    open = start
    closed = {}
    while open != {} do
        a = open.pop()
        if a.state == goal.state then
            b = solve(a, N)
            if b != false then return closePath(b)
        end if
        for i in 1..N do
            b = next(a, i)
            if b.state == goal.state then
                c = solve(b, N)
                if c != false then return closePath(c)
            end if
            if not isIn(closed, b) then
                open.push(b)
            end if
        end for
    end while
    return closed
end algorithm

```



$\phi_{query}$  :  
Some property of the output  
Ex: Some cells not selected

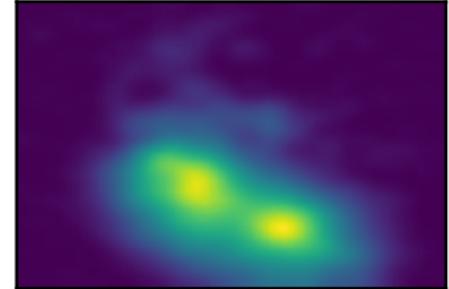
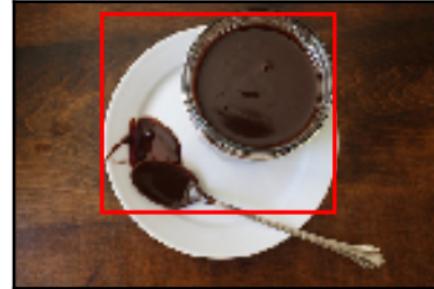
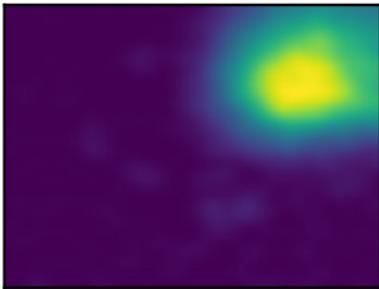
1. Vocabulary is large.
  2. How many samples (and what distribution) to consider for learning explanation ?
  3. Learning Boolean formula with PAC guarantees is hard.



Active learning Boolean formula  $\phi_{explain}$  and not learning from fixed sample.

Explanations are often short and involve only few variables !

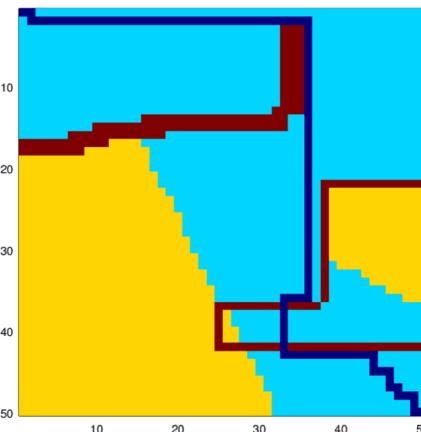
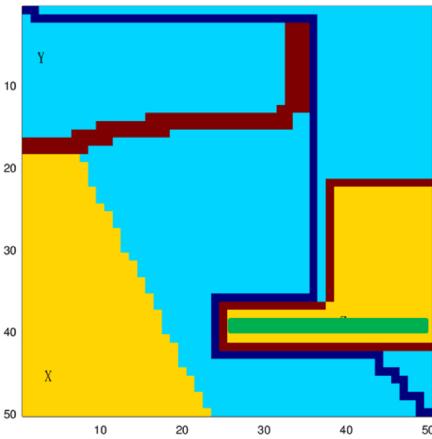
# Two Key Ideas



Active learning Boolean formula  $\phi_{explain}$  and not learning from fixed sample.

Explanations are often short and involve only few variables !

# Two Key Ideas



Involves only two variables.  
If we knew which two, we had  
only  $2^{2^2} = 16$   
possible explanations.

How do we find these relevant  
variables?

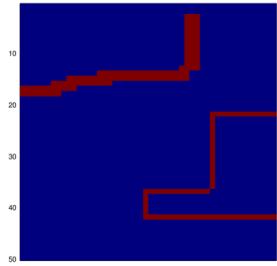


Active learning Boolean formula  $\phi_{explain}$  and not learning from fixed sample.

Explanations are often short and involve only few variables !

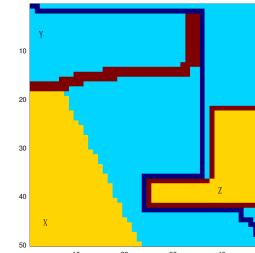
# Actively Learning Boolean Formula

Oracle



Algorithm 1: A\*

```
Input: start, goal( $\vec{v}_1, \vec{v}_2$ ),  $\phi_{query}(\vec{v})$ 
Output: path
1 if goal(start) = true then return  $codePath(start)$ 
2
3 open  $\leftarrow \{\text{start}\}$ 
4 closed  $\leftarrow \emptyset$ 
5 while open  $\neq \emptyset$  do
6   sort(open)
7    $v \leftarrow \text{open.pop(0)}$ 
8    $\phi_v \leftarrow \phi_{query}(v)$ 
9   for all the kid's kids do
10    if  $\phi(v, f) < (\phi_g(f) + 1) - \delta \cdot h(f)$ 
11    if  $goal(f) = \text{true}$  then return  $codePath(kid)$ 
12    if  $\phi(v, f) > \phi_g(f)$  then
13      if  $\phi(v, f) > \phi_g(f)$  then
14        open  $\leftarrow \text{add}(open, f)$ 
15
16 closed  $\leftarrow v$ 
17
18 return ?
```



$\phi_{query} :$   
Some property of the output  
Ex: Some cells not selected



Assignments to V  
 $m_1 = (0,0,0,1,1,0,1)$   
 $m_2 = (0,0,1,1,0,1,0)$



$\phi_{explain}(V) :$   
Using explanation vocabulary  
Ex: Obstacle presence

$\phi$

Evaluates assignments and returns T,F

# Actively Learning Relevant Variables

---

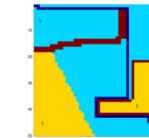
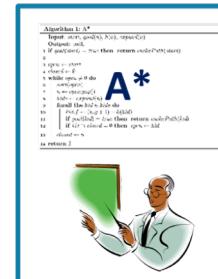
*Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$*

$\phi_{explain}$  is sparse

# Actively Learning Relevant Variables

Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

Assignments to V  
 $m1 = (0,0,0,1,1,0,1)$



$\phi_{query}$  :  
Some property of the output  
Ex: Some cells not selected



$m1 : \text{True}$

Oracle

# Actively Learning Relevant Variables

Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

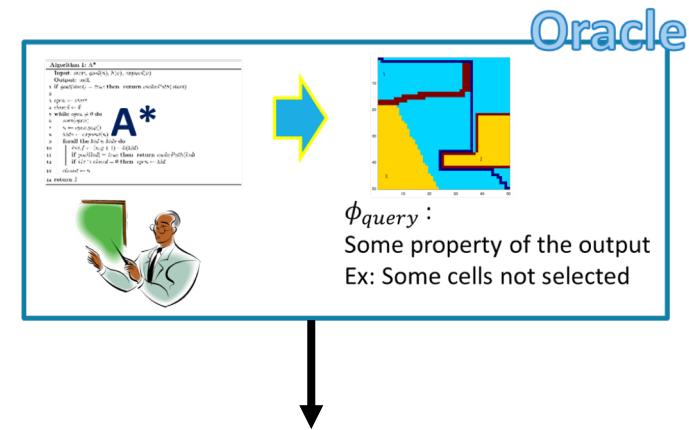
## Assignments to V

$$m_1 = (0,0,0,1,1,0,1)$$

$$m_2 = (0,0,1,1,0,1,0)$$



# Random Sample Till Oracle differs



m1: True, m2: False

# Actively Learning Relevant Variables

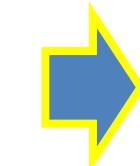
Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

Assignments to  $V$

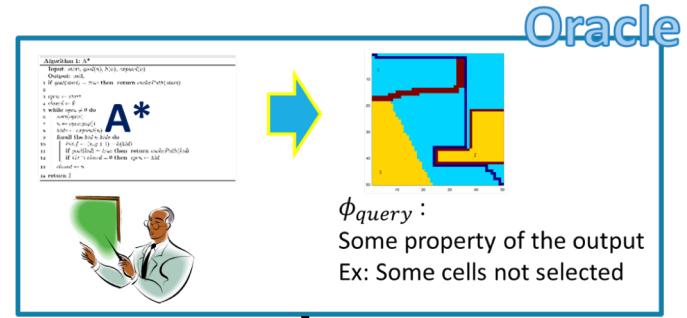
$m_1 = (0,0,0,1,1,0,1)$

$m_2 = (0,0,1,1,0,1,0)$

$m_3 = (0,0,0,1,1,1,0)$



Binary Search Over  
Hamming Distance



$m_1: \text{True}, m_2: \text{False}$

# Actively Learning Relevant Variables

Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

Assignments to V  
m1 = (0,0,0,1,1,0,1)  
m2 = (0,0,1,1,0,1,0)  
m3 = (0,0,0,1,1,1,0)



# Binary Search Over Hamming Distance

```

Algorithm 1: A*
Input: start, goalNode, P(i,j), maxCells
Output: codePath[ ]
1 if pathSize == maxCells then return codePath[start]
2 openSet = {start}
3 closedSet = {}
4 gScore[start] = 0
5 fScore[start] = h(start)
6 while openSet != {} do
7   current = min fScore in openSet
8   if current == goalNode then
9     Build the final path to goal
10    if goalCell == start then return codePath[goal]
11    if goalCell != start then return codePath[goal]
12   else
13     close current
14     for each neighbor in neighbors do
15       if neighbor is not in openSet and neighbor is not in closedSet then
16         gScore[neighbor] = gScore[current] + dist(current, neighbor)
17         fScore[neighbor] = gScore[neighbor] + h(neighbor)
18         if neighbor is not in openSet then
19           openSet.add(neighbor)
20   end for
21 end while
22 close current
23 return []

```

**A\***

$\phi_{query}:$

Some property of the output

Ex: Some cells not selected

m1: True, m2: False  
m3: True

# Actively Learning Relevant Variables

Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

Hamming  
Distance = 4

Assignments to  $V$

$m_1 = (0, 0, 0, 1, 1, 0, 1)$

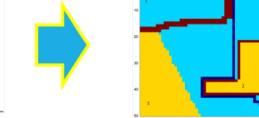
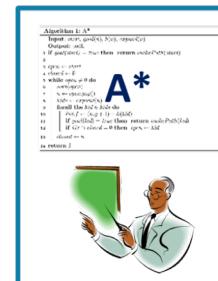
$m_2 = (0, 0, 1, 1, 0, 1, 0)$

$m_3 = (0, 0, 0, 1, 1, 1, 0)$

Hamming  
Distance = 2



Binary Search Over  
Hamming Distance



$\phi_{query}$  :  
Some property of the output  
Ex: Some cells not selected

Oracle

~~$m_1$ : True~~,  $m_2$ : False  
 $m_3$ : True

# Actively Learning Relevant Variables

Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

Hamming  
Distance = 2

Assignments to V

$m_2 = (0,0,1,1,0,1,0)$

$m_3 = (0,0,0,1,1,1,0)$

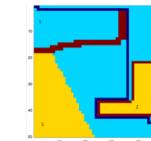
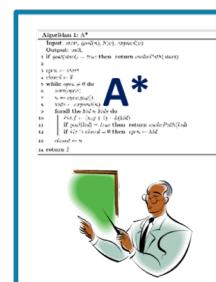
$m_4 = (0,0,1,1,1,1,0)$



Hamming  
Distance = 1



Binary Search Over  
Hamming Distance



$\phi_{query}$  :  
Some property of the output  
Ex: Some cells not selected

Oracle

$m_2$ : False,  $m_3$ : True  
 $m_4$ : True

# Actively Learning Relevant Variables

Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

Assignments to V

$m2 = (0,0,1,1,0,1,0)$

~~$m3 = (0,0,0,1,1,1,0)$~~

$m4 = (0,0,1,1,1,1,0)$

A cartoon illustration of a young girl with short, wavy orange hair and freckles. She is wearing a bright pink short-sleeved shirt. She is seated at a light-colored wooden desk, looking down at an open notebook where she is writing with a pencil. Her left hand rests on the desk, and her right arm is raised with her index finger pointing upwards, as if she is explaining something or making a point.

# Binary Search Over Hamming Distance

**Algorithm 1: A\***

```

Input: arr[grid], X0, Y0, minCost
Output: set
1. put(X0, Y0) into openList(minCost)
2. while openList is not empty do
3.   pick the cell with the lowest f-value
4.   if cell is goal then return openList
5.   for each neighbor of current cell do
6.     if neighbor is not in openList then
7.       calculate gValue and hValue
8.       if neighbor is not in closedList then
9.         calculate fValue
10.        put neighbor into openList(fValue)
11.        if neighbor is not in closedList then
12.          if neighbor is not in openList then
13.            calculate fValue
14.            put neighbor into openList(fValue)
15.    end for
16. end while
17. return set

```

**A\***

**ϕ<sub>query</sub> :**  
Some property of the output  
Ex: Some cells not selected

m2: False, ~~m3: True~~  
m4: True

# Actively Learning Relevant Variables

Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

Hamming  
Distance = 1

Assignments to V

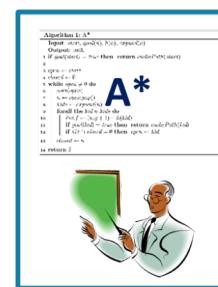
$m_2 = (0,0,1,1,0,1,0)$   
 $m_4 = (0,0,1,1,1,1,0)$



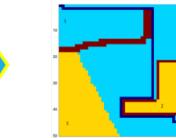
Fifth variable  $v_5$  is relevant !!



Binary Search Over  
Hamming Distance



Oracle



$\phi_{query}$  :  
Some property of the output  
Ex: Some cells not selected

$m_2$ : False,  $m_4$ : True

# Actively Learning Relevant Variables

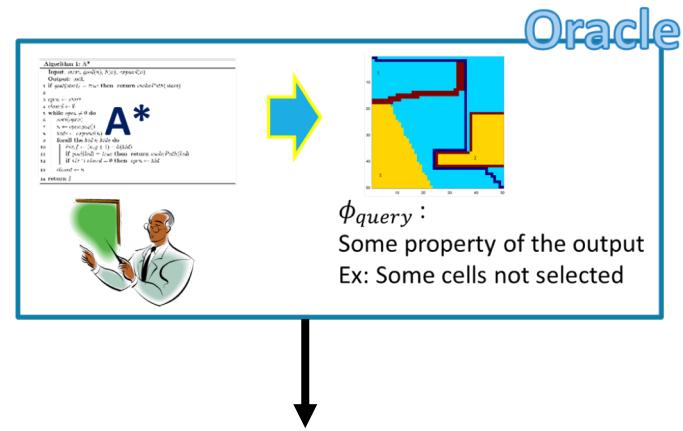
Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

## Repeat to find all relevant variables



A cartoon illustration of a young girl with short, wavy orange hair and freckles. She is wearing a bright pink short-sleeved shirt. She is seated at a light-colored wooden desk, looking towards the right with a smile. Her right hand holds a pencil, and her index finger is pointing upwards. An open book lies on the desk in front of her.

# Binary Search Over Hamming Distance



m2: False, m4: True

# Actively Learning Relevant Variables

*Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$*

For each assignment  
to relevant variables



Random Sample  
Till Oracle differs

Binary Search Over  
Hamming Distance

$$2^{|U|}$$

$$\ln(1/(1 - \kappa))$$

$$\ln(|V|)$$

**Relevant variables of  $\phi_{explain}$  found with confidence  $\kappa$  in**  
 **$2^{|U|} \ln(|V|/(1 - \kappa))$**

# Actively Learning Boolean Formula

Find  $U$  such that  $\phi_{explain}(V) \equiv \phi_{explain}(U)$  where  $|U| \ll |V|$

Used distinguishing example based approach from ICSE'10

Susmit Jha, Sumit Gulwani, Sanjit A Seshia, and Ashish Tiwari. Oracle-guided component-based program synthesis. In *2010 ACM/IEEE 32nd International Conference on Software Engineering*, volume 1, pages 215–224. IEEE, 2010.

Scales to ~200 variables

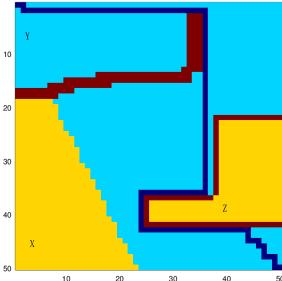


Build Truth Table for the relevant variables  $U$

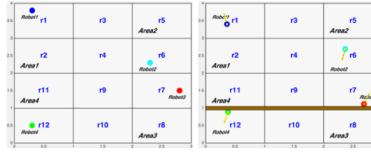
Worst Case:  $2^{|U|}$

$\phi_{explain}$  found with confidence  $\kappa$  in  
 $O(2^{|U|} \ln(|V|/(1 - \kappa)))$

# Experiments



A\* Planning  
 $|V| = 2500$        $10^{153}$   
 $|U| \leq 4$   
Runtime < 3 minutes



Reactive Exploration  
Strategy       $10^{28}$   
 $|V| = 96$   
 $|U| \leq 2$   
Runtime < 5 seconds

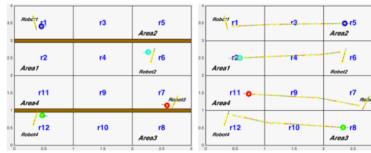
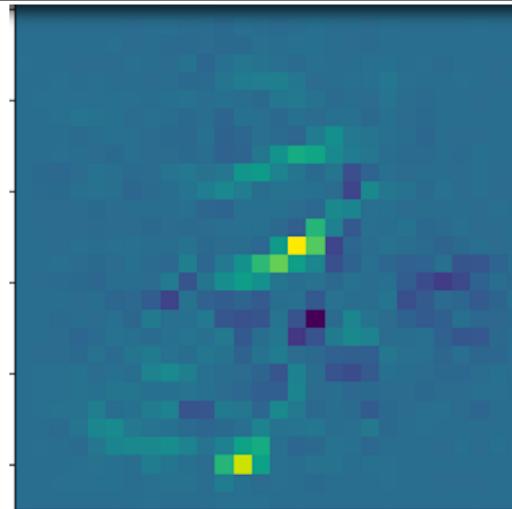
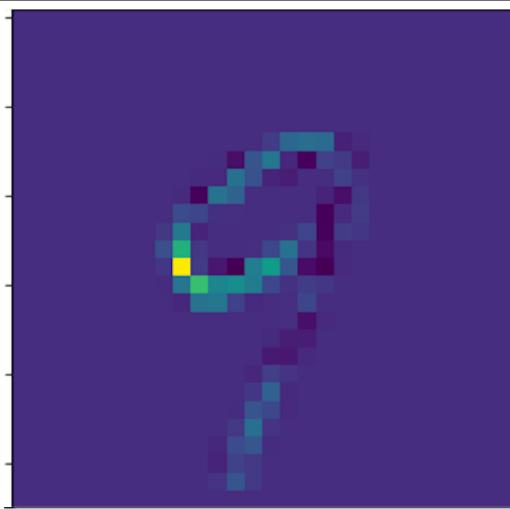


Image Classification: MNIST

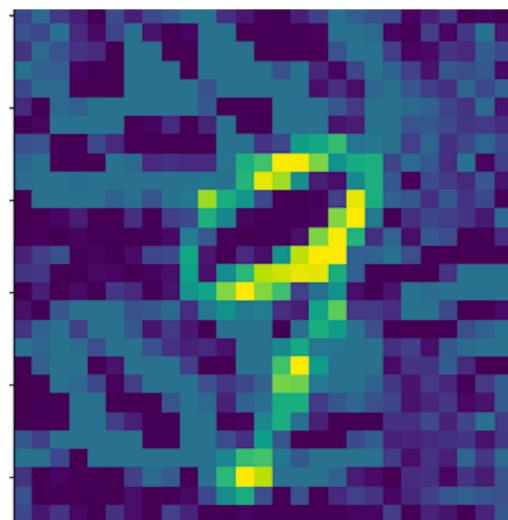
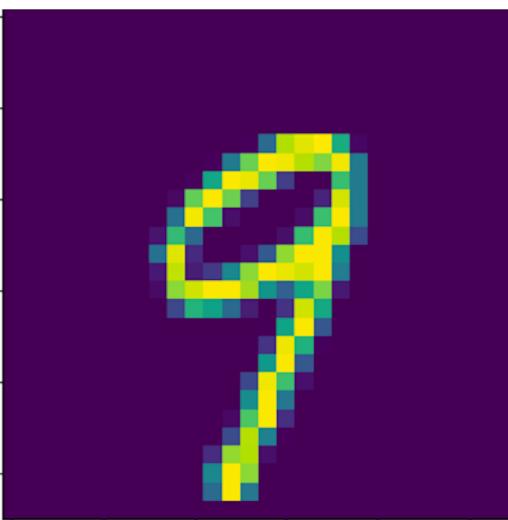
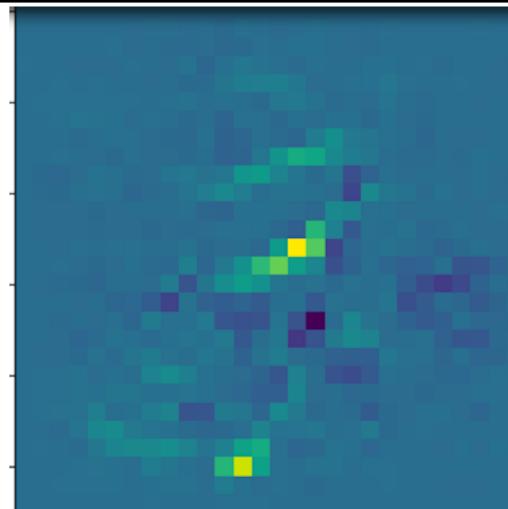
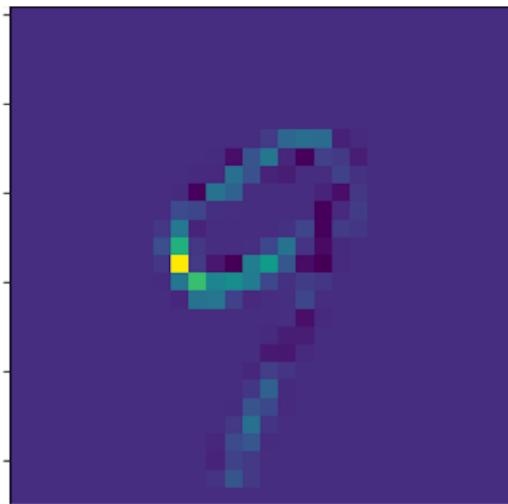


Image Classification: ImageNet  
with Carlini-Wagner  
Adversarial Attacks

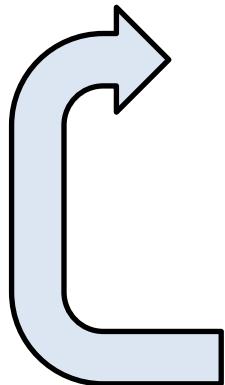
# Experiments



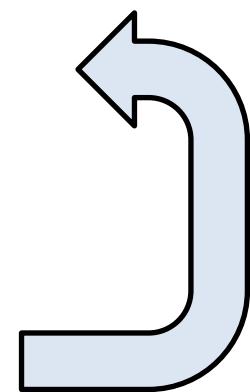
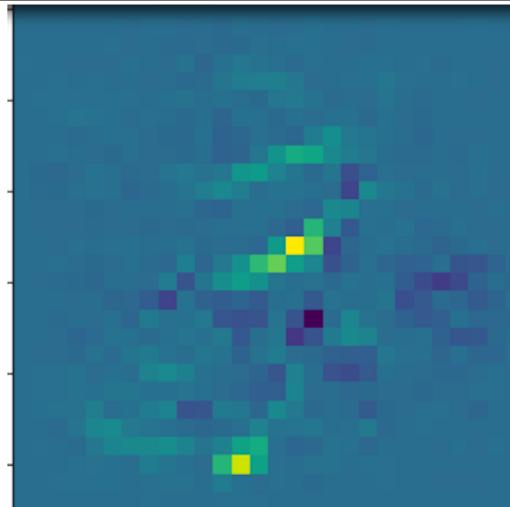
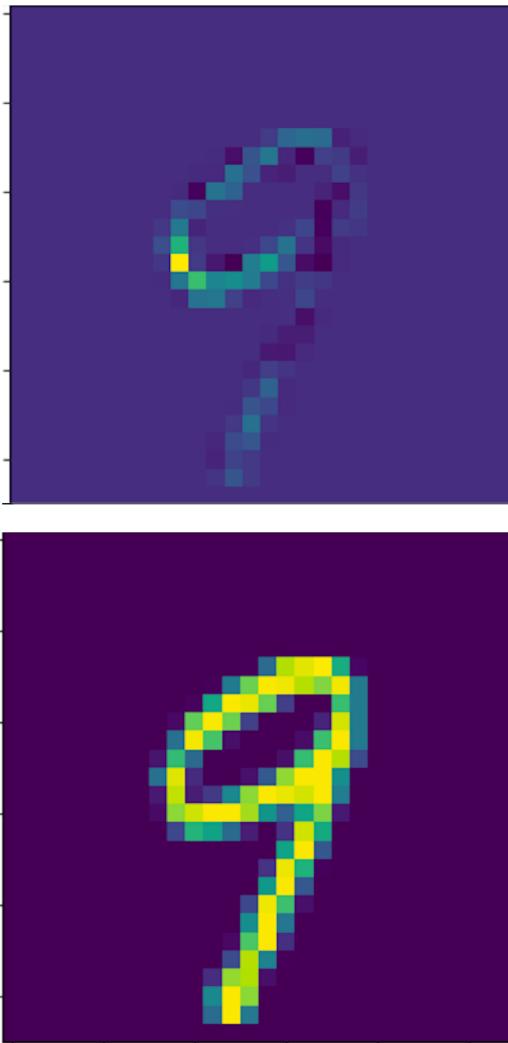
# Experiments



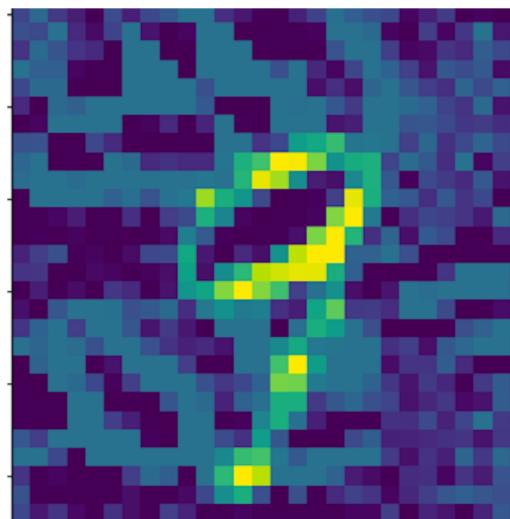
# Experiments



Why 9

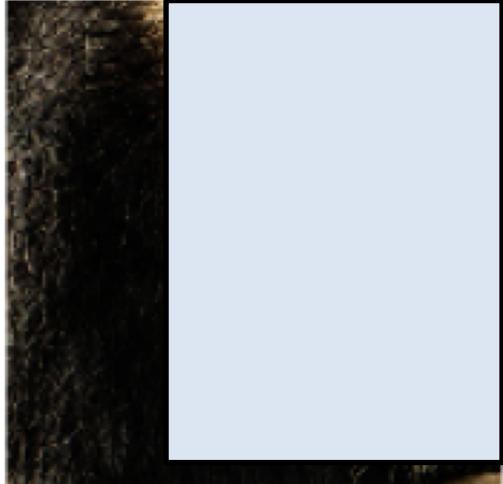


Why 3



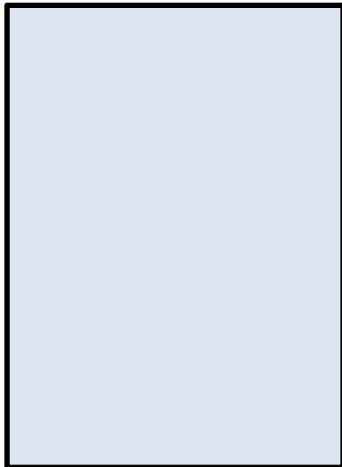
# Why not just do sensitivity analysis?

---

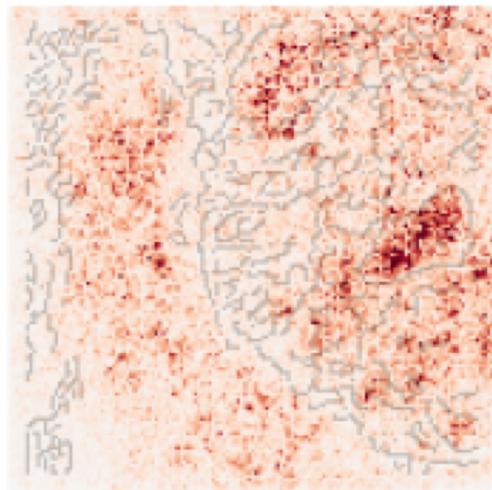


# Why not just do sensitivity analysis?

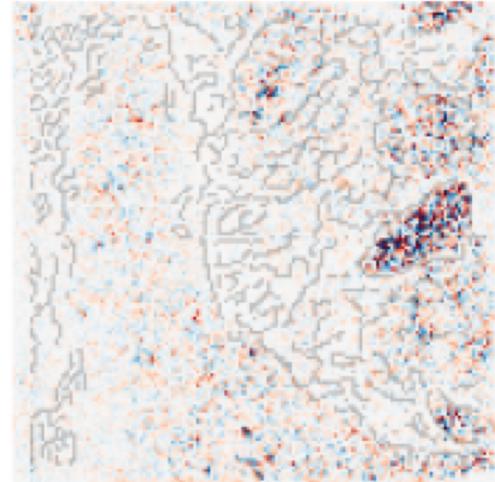
---



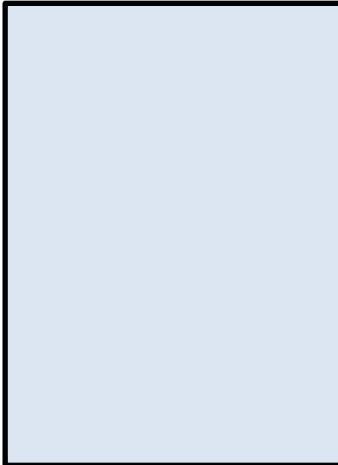
# Why not just do sensitivity analysis?



Sensitivity (IG)

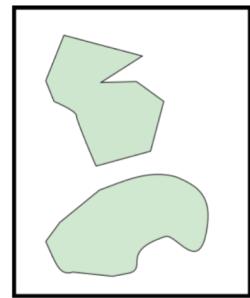


Sparse Boolean  
Formula Learning

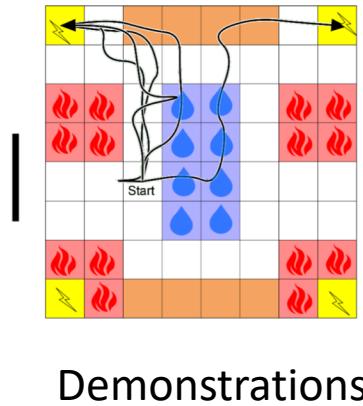


# Learning Temporal Logic Properties from Noisy Time Traces

Pr(



Specification



Demonstrations

$$\Pr(\text{Specification} \mid \text{Demonstrations}) \propto e^{D_{KL}(\mathcal{B}(\bar{\varphi}) \parallel \mathcal{B}(\hat{\varphi}))}$$

Bernoulli Distribution

Satisfaction probability for Alice given dynamics

Satisfaction probability given uniformly random actions

- Composable
- Resilient to changes in task context
- Interpretable
- Can leverage formal methods tools

Marcell Vazquez-Chanlatte, Susmit Jha , Ashish Tiwari, Mark K. Ho and Sanjit A. Seshia.  
Learning Task Specifications from Demonstrations. NeurIPS, 2018

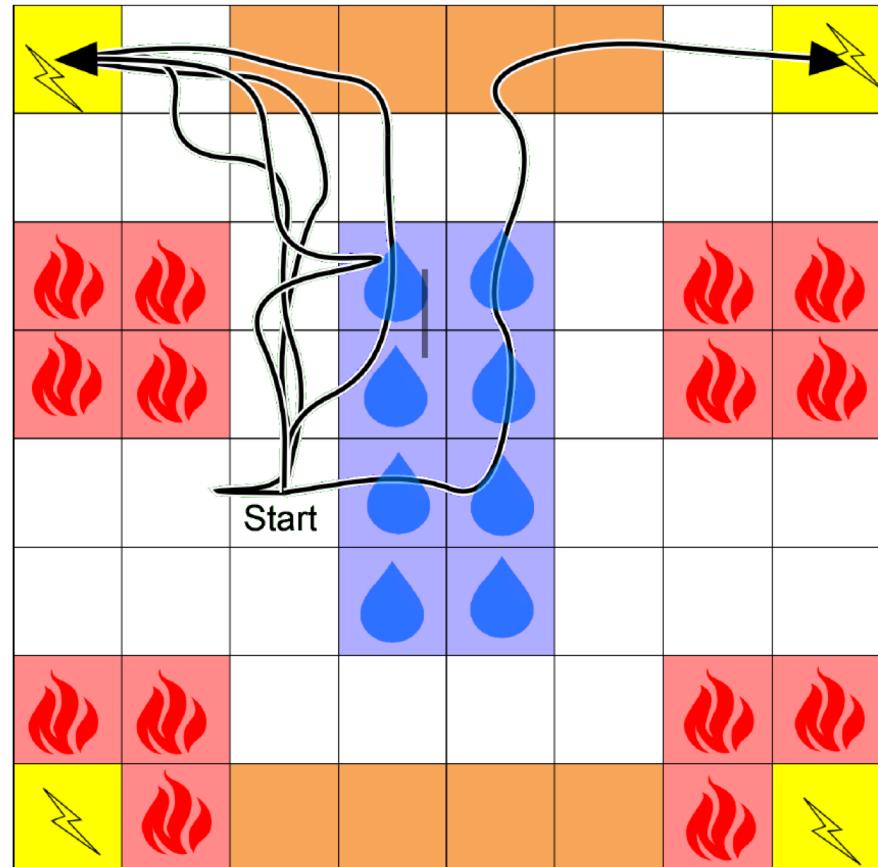
# Communicating Using Demonstrations: More involved example

1. Avoid fire (red).
2. Eventually Recharge (yellow).
3. If you touch the water (blue) then dry off (brown) before recharging (yellow).

## Temporal Logic Specification

H: Historically  
O: Once  
S: Since

$$(H \neg red \wedge O yellow) \wedge H((yellow \wedge O blue) \Rightarrow (\neg blue \wedge S brown))$$



# Interpretability / Explanation Generation in TRINITY

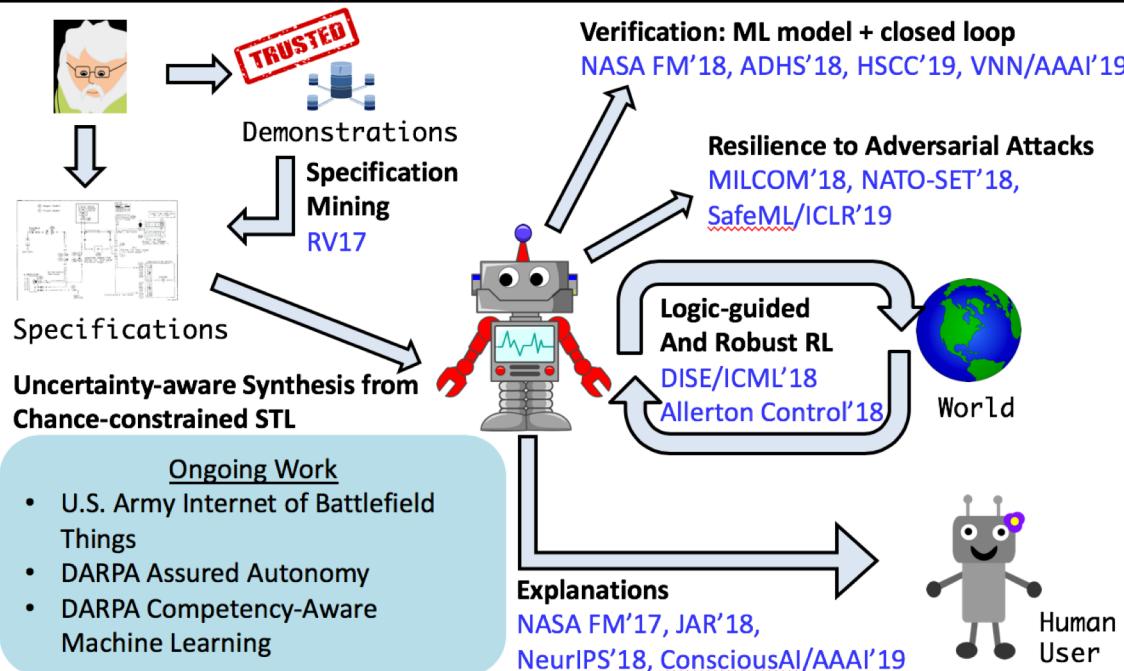
---

- **Inferring and Conveying Intentionality: Beyond Numerical Rewards to Logical Intentions.** Susmit Jha and John Rushby.  
AAAI Spring Symposium, Towards Conscious AI Systems, 2019
- **Learning Task Specifications from Demonstrations.** Marcell Vazquez-Chanlatte, Susmit Jha, Ashish Tiwari, Mark K. Ho and Sanjit A. Seshia.  
Neural Information Processing Systems (NeurIPS), 2018
- **Explaining AI Decisions Using Efficient Methods for Learning Sparse Boolean Formulae.** Susmit Jha, Tuhin Sahai, Vasumathi Raman, Alessandro Pinto and Michael Francis.  
Journal of Automated Reasoning, 2018
- **On Learning Sparse Boolean Formulae For Explaining AI Decisions.** Susmit Jha, Vasumathi Raman, Alessandro Pinto, Tuhin Sahai, and Michael Francis.  
NASA Formal Methods (NFM), 2017

# Thanks!

If you are interested in building *trusted, resilient and interpretable AI*, please contact me with your CV if you are interested.

## TRINITY @ SRI



### Co-travelers (Present and Past):

Brian Burns, Margaret Chapman, Ajay Divakaran, Sauradeep Dutta, Michael Francis, Mark K. Ho, Uyeong Jang, Brian Jalaian, Somesh Jha, Patrick Lincoln, Alessandro Pinto, Vasu Raman, John Rushby, Dorsa Sadigh, Sriram Sankaranarayanan, Sanjit A. Seshia, Natarajan Shankar, Ashish Tiwari, Claire Tomlin, Marcell Vazquez-Chanlatte, Gunjan Verma

### Funding sources (Present and Past):

DARPA, US Army Research Laboratory, National Science Foundation



# TRINITY @ SRI

