Panel 1: AI Assurance: Small and Large

Tom Dietterich

Collaborative Robotics and Intelligent Systems (CoRIS)

Oregon State University

Oregon State University

Assurance for Machine Learning

- Assurance by Construction
- Assurance by Run-time Monitoring

Assurance by Construction

- Robust training
  - Adversarial training can improve robustness
    - (Goodfellow, et al., 2015; Madry, et al., 2018)
- Robust query processing
  - Post-processing by stability testing can guarantee robustness
    - (Li, Chen, Wang & Carin, 2019, arXiv 1809.03113)
    - Requires stationarity assumption

Run-Time Assurance

- Rejection
  - Reject queries for which the ML system has low confidence
    - Requires fitting a confidence function or rejection function
    - Calibrated probabilities (Nicolescu-Mizil & Caruana, 2005)
    - Rejection functions (Cortes, DeSalvo & Mohri, 2018)
  - Requires stationarity assumption

Data Shift Detection

- Data Shift:
  - Changes in class probabilities (e.g., increase in cyberattacks)
  - Changes in input distribution (e.g., network traffic shifts)
  - Changes in the decision boundary (e.g., attackers try to hide)
  - New classes to predict (e.g., new kind of cyberattack)
- Methods:
  - For single queries: Anomaly detection (Liu, Garrepalli, et al. ICML 2018)
  - For a batch of queries: Two-sample testing (Lopez-Paz & Oquab, 2018; Gretton, et al. 2007, Anderson, et al. 1994)
    - Provides guarantees

High Reliability Organizations
Todd LaPorte, Gene Rochlin, and Karlene Roberts

- Preoccupation with failure
  - Fundamental belief that the system has unobserved failure modes
  - Treat anomalies and near misses as symptoms of a problem with the system
- Reluctance to simplify interpretations
  - Comprehensively understand the situation
- Sensitivity to operations
  - Maintain continuous situational awareness
- Commitment to resilience
  - Develop the capability to detect, contain, and recover from errors. Practice improvisational problem solving
- Deference to expertise
  - During a crisis, authority migrates to the person who can solve the problem, regardless of their rank

Designing AI Systems to be HROs

- Maintain Situational Awareness
  - AI methods are very good at integrating data from multiple sensors and effectors to estimate a probability distribution over states
- Detect Anomalies and Near Misses
  - Anomalies: Yes
  - Near Misses: Research needed
- Generate Candidate Explanations for Anomalies & Near Misses
  - Very little work: Research needed
- Improvise Solutions
  - Improvisational problem solving that extends or operates outside the system model

Assessment: Designing AI as an HRO

	Assessment
Situational Awareness	A mature methods
Detect Anomalies and Near Misses	B high-dimension, dynamics
Explain Anomalies and Near Misses	D only basic techniques
Improvise Solutions	F

Designing a Human + AI Team as an HRO

- Even very powerful AI systems will be surrounded by a human team
- Situational Awareness
  - AI can track the situation, but humans and AI must establish a shared mental model of the situation: Research needed
  - Humans must be aware of what version of the AI system they are using. When was it last updated/retrained? Research needed
- Detect Anomalies and Near Misses
  - AI system must understand and predict behavior of human team
  - AI and Humans must work together: interactive anomaly detection
- Generate Candidate Explanations for Anomalies & Near Misses
  - Very little work: Research needed
- Improvise Solutions
  - AI should support human improvisational problem solving: Research Needed
  - Example: mixed-initiative planning

Assessment: Human + AI HROs

	Assessment
Situational Awareness	C poor UI, poor communication
Detect Anomalies and Near Misses	C user feedback to anomaly detection
Explain Anomalies and Near Misses	D only basic techniques
Improvise Solutions	D mixed-initiative planning

Backup Material

Assurance by Construction

- Let $f(x;\theta)$ be a predictive model parameterized by θ
- Training data $\{(x_1, y_1), \ldots, (x_N, y_N)\}$
- Standard training

$$\theta^* := \operatorname{argmin}_\theta \sum_{i=1}^{N} L(f(x_i;\theta), y_i)$$

where $L(y, y)$ is the loss function for predicting y when the true answer was y

- Robust (adversarial) training

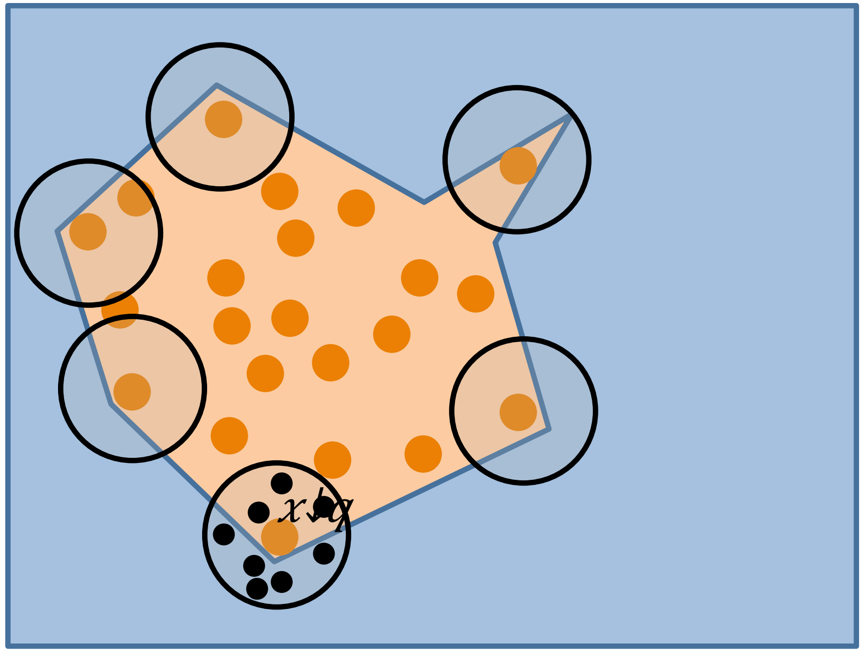
$$\theta^* := \operatorname{argmin}_\theta \max_{\delta_i \in \Delta} \sum_{i=1}^{N} L(f(x_i + \delta_i;\theta), y_i)$$

where Δ is a set of allowed perturbations (Goodfellow, et al., 2015; Madry, et al., 2018)

Equivalent, in some cases, to regularization methods

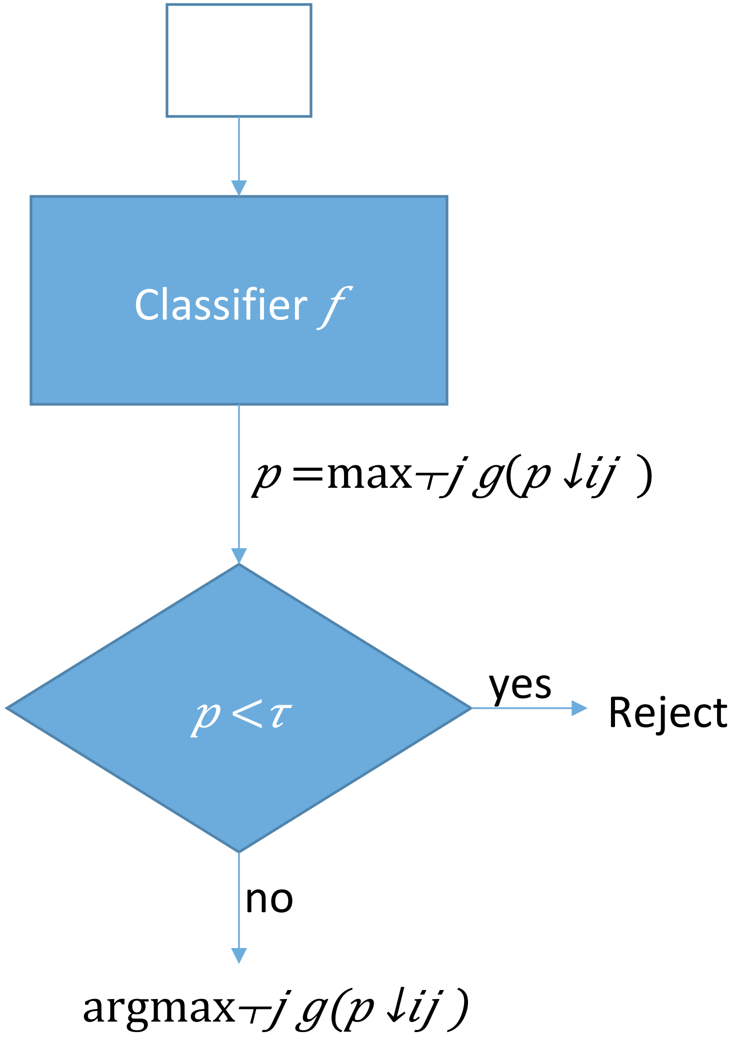
Assurance by Post Processing

- Given a trained f, post-process it to guarantee robustness
- Example: Stability Testing
  - Given query x_q, sample perturbations and predict y using majority vote
  - $f(x_q;\theta)=orange$
  - but the majority of perturbed points have $f(x_q+\delta)=blue$
  - so $y:=blue$
- First method to give a guarantee on ImageNet (1000 classes)
- Li, Chen, Wang & Carin, 2019, arXiv 1809.03113



Assurance by Rejection

- Construct a rejection function g
- Example: g produces a calibrated probability. If the maximum probability is too small, then reject
- This is a type of *competence model*

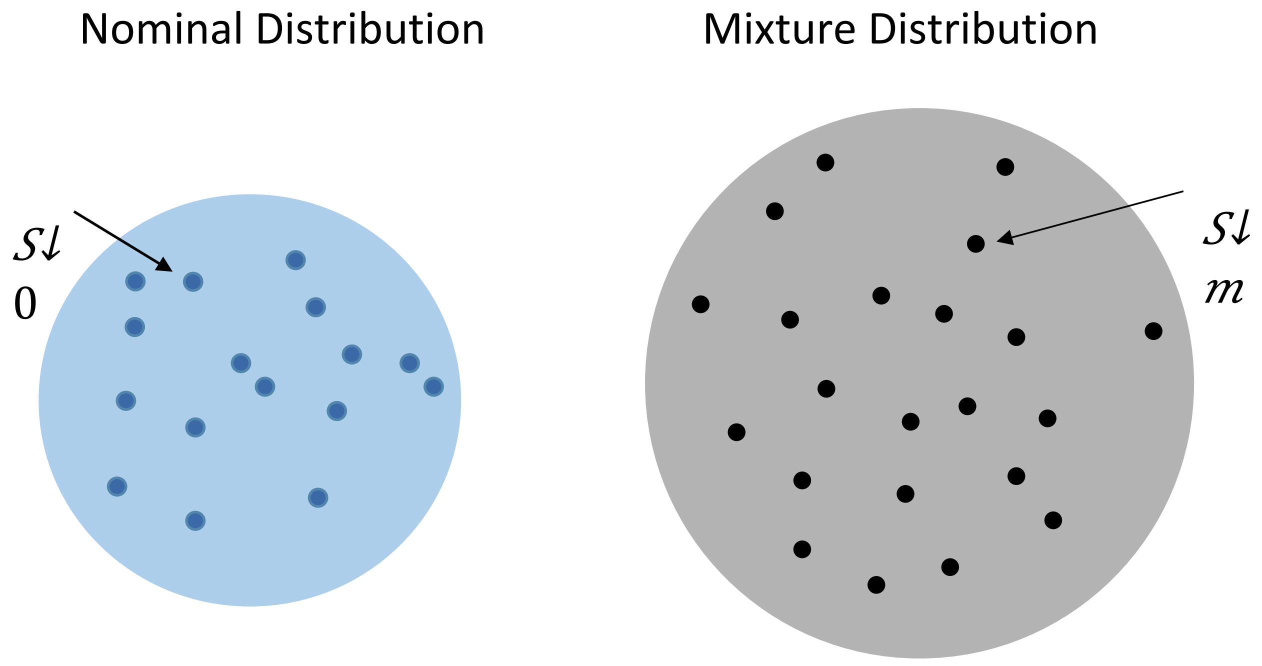


Assurance by Runtime Monitoring

- Construction-time guarantees assume test queries come from the same distribution as training queries
- This assumption rarely holds in practice
  - Changes in class probabilities (e.g., increase in cyberattacks)
  - Changes in input distribution (e.g., network traffic shifts)
  - Changes in the decision boundary (e.g., attackers try to hide)
  - New classes to predict (e.g., new kind of cyberattack)
- Data shift detection
  - Compare recent queries $\{x_{q_1}, x_{q_2}, \ldots, x_{q_m}\}$ to training points $\{x_1, \ldots, x_N\}$
  - Use two-sample tests:
    - typical sets, kernel maximum mean discrepancy, old-vs-new classifier
- Anomaly detection
  - $A(x_q) := -\log P(x_q)$, where P is the distribution of training points
  - Operates on single points => generates many false alarms

Open Category Guarantee

- Assume we know (a bound on) the proportion α of test queries that correspond to new classes "aliens"
- Then we can estimate a threshold τ that with high probability will detect $1-\epsilon$ of the aliens on new test queries
- Liu, Garrepalli, et al. ICML 2018



Proportion of Aliens = α

$$P_m = (1-\alpha)P_0 + \alpha P_a$$