Data-Driven Control and Data-Poisoning attacks in Buildings: the KTH Live-In Lab case study

Alessio Russo*, Marco Molinari and Alexandre Proutiere

Mediterranean Conference on Control and Automation (MED), 2021

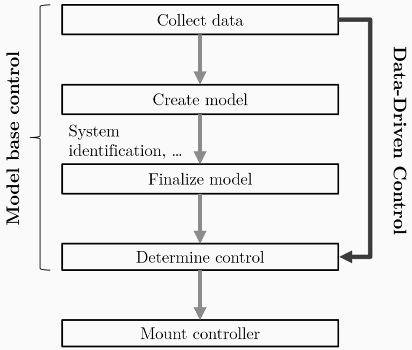
KTH, Royal Institute of Technology, Stockholm

Problem motivation and background

Problem motivation

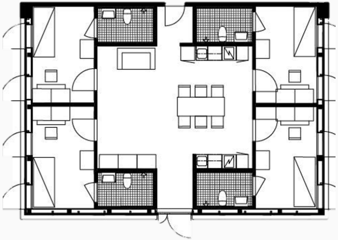
- **Temperature control in buildings may be complicated.**
- Data-driven control approaches: use data to directly compute a control law.
    1. Model-reference based methods: Virtual Reference Feedback Tuning (VRFT) [1], Iterative Feedback Tuning [2], correlation-based [3]...
    2. Methods based on Willems' et al. lemma [4,5].
- The data can be poisoned.
- We focus on VRFT, a popular model-reference based method.

Problem motivation

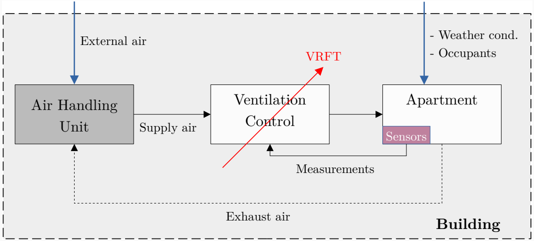


- **Temperature control in buildings may be complicated.**
- Data-driven control approaches: use data to directly compute a control law.
  1. Model-reference based methods: Virtual Reference Feedback Tuning (VRFT) [1], Iterative Feedback Tuning [2], correlation-based [3]...
  2. Methods based on Willems' et al. lemma [4,5].
- The data can be poisoned.
- We focus on VRFT, a popular model-reference based method.

KTH Live-in Lab Testbed



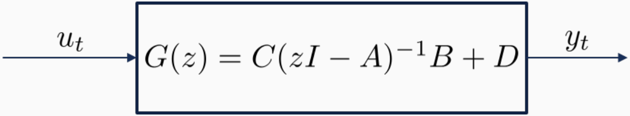
KTH Live-in Lab Testbed



1. We **modeled the building using IDA-ICE**, a building performance simulation software [6].
2. We focused on the problem of **ventilation control of a single apartment**.
3. We applied **VRFT to derive a control law**, directly from the data of an (empty) apartment.
4. Finally, we tested whether VRFT is susceptible to **data poisoning attacks**.

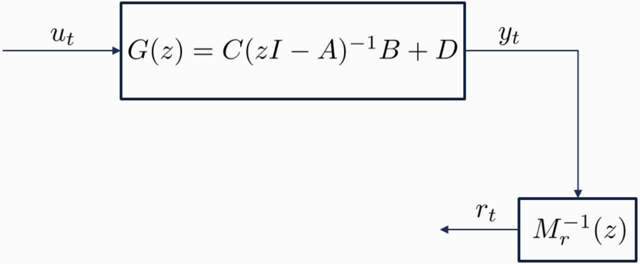
Temperature control

Virtual Reference Feedback Tuning [1]



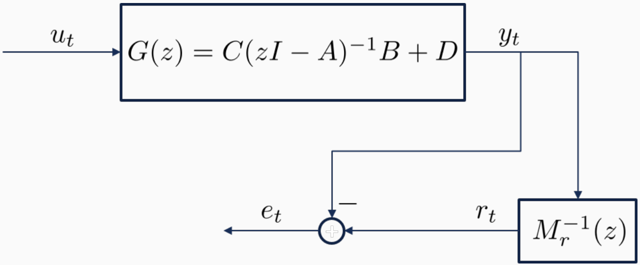
1. Feed a pre-designed signal u_t and measure y_t.

Virtual Reference Feedback Tuning [1]



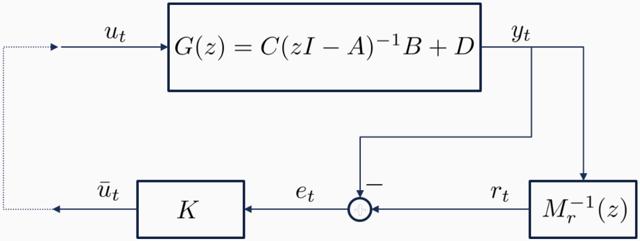
1. Feed a pre-designed signal u_t and measure y_t.
2. Given a reference model $M_r(z)$, compute the reference signal r_t.

Virtual Reference Feedback Tuning [1]



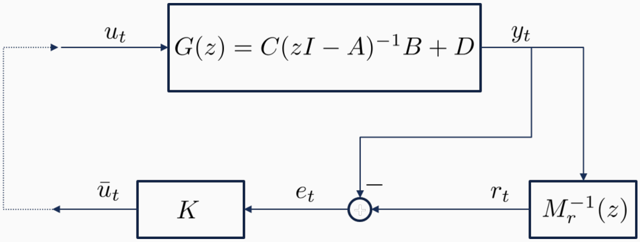
1. Feed a pre-designed signal u_t and measure y_t.
2. Given a reference model $M_r(z)$, compute the reference signal r_t.
3. Compute the *virtual error* $e_t = r_t - y_t$.

Virtual Reference Feedback Tuning [1]



1. Feed a pre-designed signal u_t and measure y_t.
2. Given a reference model $M_r(z)$, compute the reference signal r_t.
3. Compute the *virtual error* $e_t = r_t - y_t$.
4. Design a control law K that outputs a signal $\bar{u}_t$ that is *close* to u_t.

Virtual Reference Feedback Tuning [1]

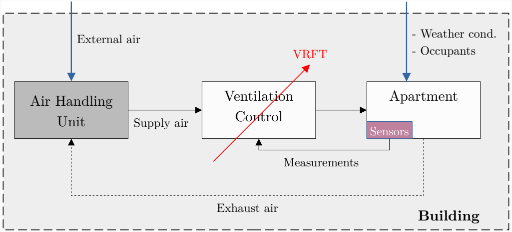


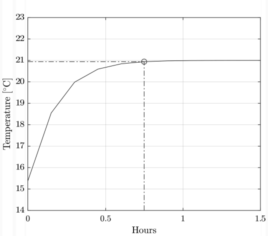
1. Feed a pre-designed signal u_t and measure y_t.
2. Given a reference model $M_r(z)$, compute the reference signal r_t.
3. Compute the *virtual error* $e_t = r_t - y_t$.
4. Design a control law K that outputs a signal $\bar{u}_t$ that is *close* to u_t.

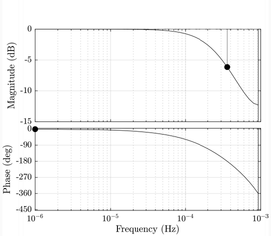
Under some assumptions, it is possible to show that minimizing $\frac{1}{N}\sum_{t=1}^{N}(\bar{u}_t - u_t)^2$, for $N \to \infty$, yields a law K that converges to the minimum of

$$\min_K \|M_r(z) - (1 - M_r)KG(z)\|_2.$$

Temperature control: method

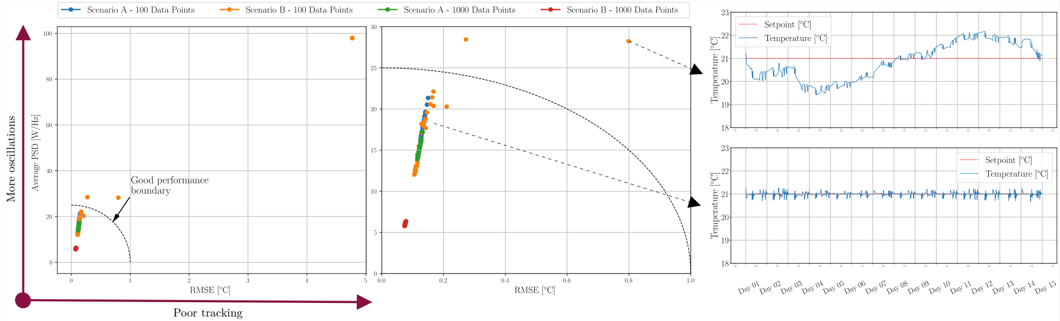






1. Data was sampled every 540 [s].
2. The control signal is a real number in $[0, 1]$. We designed 2 experiments for VRFT.
   - **Scenario A:** $u_t \sim \mathcal{N}(\frac{1}{2}, \frac{1}{6})$.
   - **Scenario B:** $u_t \sim \mathcal{N}(\frac{1}{2}, 1)$.
3. Goal of VRFT: compute $K_\theta(z)$, where $K_\theta(z) = \sum_{k=1}^{3} \theta_k \frac{z^{-k+2}}{z-1}$ (PID-like controller).
4. We used a 2nd order reference model (see plot on the left).

Temperature control: results



1. **Scenario A:** $u_t \sim \mathcal{N}(\frac{1}{2}, \frac{1}{6})$; **Scenario B:** $u_t \sim \mathcal{N}(\frac{1}{2}, 1)$.
2. January was used for training of VRFT (empty apartment); February for evaluation of the control law (1 person). For each case we run 50 simulations.

Data poisoning

Attack Formulation

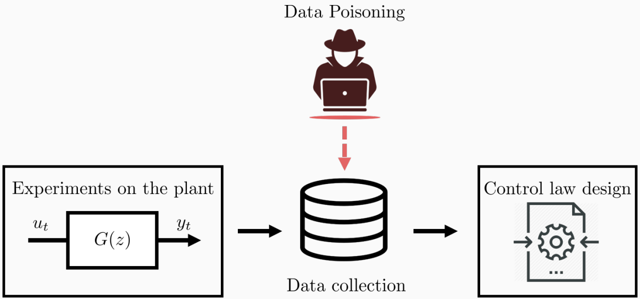


Figure 1: Data poisoning scheme.

Attack Formulation

We can cast the attacker's problem as a bi-level optimization problem.

$$\begin{aligned} \max_{\boldsymbol{u}',\boldsymbol{y}'} \quad & \mathcal{A}(\boldsymbol{u},\boldsymbol{y},K(\boldsymbol{u}',\boldsymbol{y}')) \\ \text{s.t.} \quad & K(\boldsymbol{u}',\boldsymbol{y}') \in \arg\min_K \mathcal{L}(\boldsymbol{u}',\boldsymbol{y}',K) \\ & \|\boldsymbol{u}'-\boldsymbol{u}\|_2 \le \varepsilon_u \|\mathbf{u}\|_2, \quad \|\boldsymbol{y}'-\boldsymbol{y}\|_2 \le \varepsilon_y \|\mathbf{y}\|_2, \end{aligned}$$

- We denote by $u_t' = u_t + a_{u,t}$ the poisoned input, where $\mathbf{a_u}$ is the poisoning signal (similarly for y_t').
- We denote by $\mathcal{L}$ the learner's criterion (e.g., the MSE loss of VRFT).
- Similarly, $\mathcal{A}$ is the attacker's criterion.

Attack based on *Russo, A., Proutiere, A.. Poisoning attacks against data-driven control methods. American Control Conference, 2021.*

VRFT: Attack Formulation

1. **Remember the VRFT criterion** $\frac{1}{N}\sum_{t=1}^{N}(u_t - \bar{u}_t)^2$, where $\bar{u}_t = K_\theta(z)(M_r^{-1}(z) - 1)y_t$.
2. The learner's criterion under attack can be rewritten in matrix form as

$$\mathcal{L}(\mathbf{u}', \mathbf{y}', \theta) = \frac{1}{N}\|\mathbf{u}' - \Phi(\mathbf{y}')\theta\|_2^2$$

   where Φ is a matrix that includes the effect of $M_r(z)$ (ref. model) and $K_\theta(z)$.
3. **How do we choose the attacker's criterion? Simplest choice is to just maximize the original VRFT criterion!**

$$\begin{aligned} \max_{\boldsymbol{u}', \boldsymbol{y}'} \quad & \mathcal{A}(\boldsymbol{u}, \boldsymbol{y}, \hat{\theta}(\boldsymbol{u}', \boldsymbol{y}')) = \frac{1}{N}\left\| \boldsymbol{u} - \Phi(\boldsymbol{y})\hat{\theta}(\boldsymbol{u}', \boldsymbol{y}') \right\|_2^2 \\ \text{s.t.} \quad & \hat{\theta}(\boldsymbol{u}', \boldsymbol{y}') = \left(\Phi^\top(\boldsymbol{y}')\Phi(\boldsymbol{y}')\right)^{-1}\Phi^\top(\boldsymbol{y}')\boldsymbol{u}' \\ & \|\boldsymbol{u}' - \boldsymbol{u}\|_2 \le \varepsilon_u \|\mathbf{u}\|_2, \quad \|\boldsymbol{y}' - \boldsymbol{y}\|_2 \le \varepsilon_y \|\mathbf{y}\|_2. \end{aligned}$$

The problem is concave in the input signal $\mathbf{u}'$, and non-convex in the output signal $\mathbf{y}'$.

VRFT: Attack Formulation

1. **Remember the VRFT criterion** $\frac{1}{N}\sum_{t=1}^{N}(u_t - \bar{u}_t)^2$, where $\bar{u}_t = K_\theta(z)(M_r^{-1}(z) - 1)y_t$.
2. The learner's criterion under attack can be rewritten in matrix form as

$$\mathcal{L}(\mathbf{u}', \mathbf{y}', \theta) = \frac{1}{N}\|\mathbf{u}' - \Phi(\mathbf{y}')\theta\|_2^2$$

   where Φ is a matrix that includes the effect of $M_r(z)$ (ref. model) and $K_\theta(z)$.
3. **How do we choose the attacker's criterion? Simplest choice is to just maximize the original VRFT criterion!**

$$\begin{aligned}
\max_{\boldsymbol{u}', \boldsymbol{y}'} \quad & \mathcal{A}(\boldsymbol{u}, \boldsymbol{y}, \hat{\theta}(\boldsymbol{u}', \boldsymbol{y}')) = \frac{1}{N}\left\|\boldsymbol{u} - \Phi(\boldsymbol{y})\hat{\theta}(\boldsymbol{u}', \boldsymbol{y}')\right\|_2^2 \\
\text{s.t.} \quad & \hat{\theta}(\boldsymbol{u}', \boldsymbol{y}') = \left(\Phi^\top(\boldsymbol{y}')\Phi(\boldsymbol{y}')\right)^{-1}\Phi^\top(\boldsymbol{y}')\boldsymbol{u}' \\
& \|\boldsymbol{u}' - \boldsymbol{u}\|_2 \leq \varepsilon_u\|\mathbf{u}\|_2, \quad \|\boldsymbol{y}' - \boldsymbol{y}\|_2 \leq \varepsilon_y\|\mathbf{y}\|_2.
\end{aligned}$$

The problem is concave in the input signal $\mathbf{u}'$, and non-convex in the output signal $\mathbf{y}'$.

VRFT: Attack Formulation

Input: Data-set $(\boldsymbol{u}, \boldsymbol{y})$; objective function $\mathcal{A}$;
parameters $\varepsilon_u, \varepsilon_y$, η

Output: Attack vectors $\boldsymbol{a}_u, \boldsymbol{a}_y$

$i \leftarrow 0, (\boldsymbol{a}_u^{(i)}, \boldsymbol{a}_y^{(i)}) \leftarrow (\mathbf{0}, \mathbf{0})$

$\hat{\theta}^{(i)} \leftarrow \hat{\theta}(\boldsymbol{u} + \boldsymbol{a}_u^{(i)}, \boldsymbol{y} + \boldsymbol{a}_y^{(i)})$

$J^{(i)} \leftarrow \mathcal{A}(\boldsymbol{u}, \boldsymbol{y}, \hat{\theta}^{(i)})$

do

- $\boldsymbol{a}_u^{(i+1)} \leftarrow$ solve attacker's problem in $\boldsymbol{a}_u$ using CCP [9]
- $\boldsymbol{a}_y^{(i+1)} \leftarrow \mathrm{PGA}(\varepsilon_y, \hat{\theta}(\boldsymbol{u} + \boldsymbol{a}_u^{(i+1)}, \boldsymbol{y} + \boldsymbol{a}_y^{(i)}))$
- $\hat{\theta}^{(i+1)} \leftarrow \hat{\theta}(\boldsymbol{u} + \boldsymbol{a}_u^{(i+1)}, \boldsymbol{y} + \boldsymbol{a}_y^{(i+1)})$
- $J^{(i+1)} \leftarrow \mathcal{A}(\boldsymbol{u}, \boldsymbol{y}, \hat{\theta}^{(i+1)})$
- $i \leftarrow i + 1$

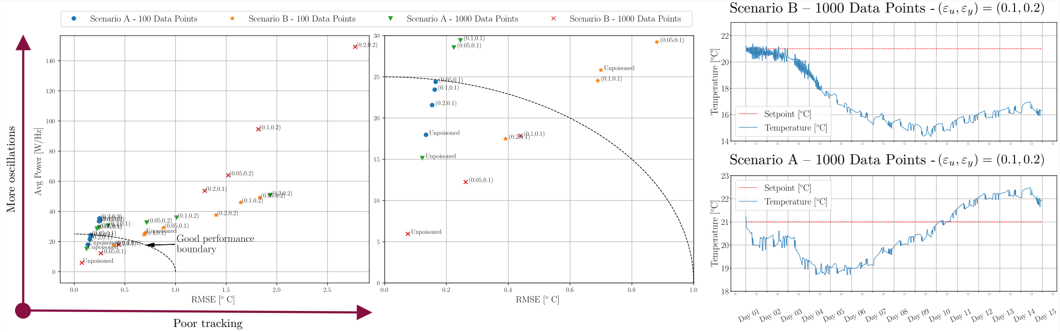
while $|J^{(i+1)} - J^{(i)}| > \eta$

-Remember that $\mathbf{u}' = \mathbf{u} + \mathbf{a_y}$ (resp. $\mathbf{y}'$).
-The attacker wants to solve

$$\begin{aligned}
\max_{\boldsymbol{u}', \boldsymbol{y}'} \quad & \frac{1}{N} \left\| \boldsymbol{u} - \Phi(\boldsymbol{y}) \hat{\theta}(\boldsymbol{u}', \boldsymbol{y}') \right\|_2^2 \\
\text{s.t.} \quad & \hat{\theta}(\boldsymbol{u}', \boldsymbol{y}') = \left(\Phi^\top(\boldsymbol{y}') \Phi(\boldsymbol{y}') \right)^{-1} \Phi^\top(\boldsymbol{y}') \boldsymbol{u}' \\
& \|\boldsymbol{u}' - \boldsymbol{u}\|_2 \leq \varepsilon_u \|\mathbf{u}\|_2, \quad \|\boldsymbol{y}' - \boldsymbol{y}\|_2 \leq \varepsilon_y \|\mathbf{y}\|
\end{aligned}$$

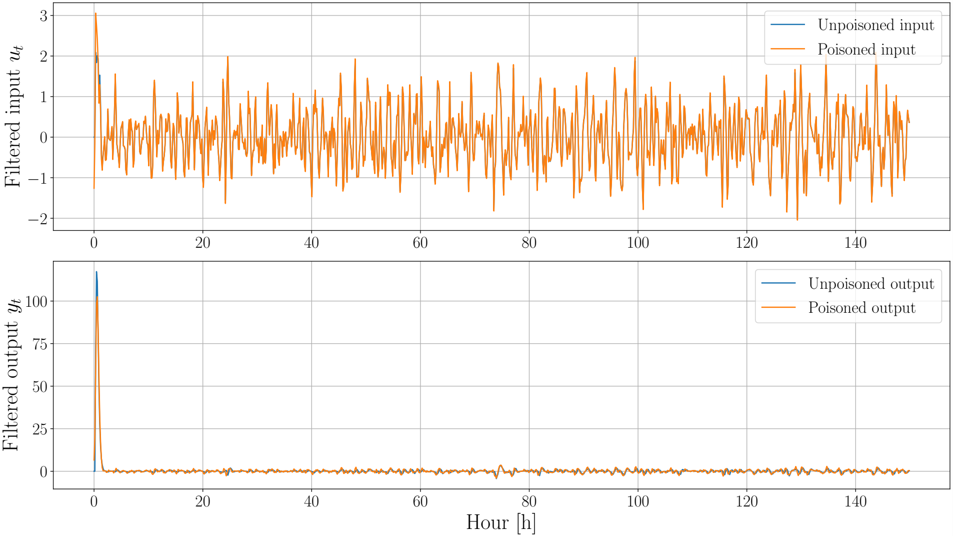
-The problem is concave in the input signal $\mathbf{u}'$: **we use convex-concave programming techniques.**
-The problem is non-convex in the output signal $\mathbf{y}'$: **we use projected gradient ascent.**

Data poisoning: results



1. **Scenario A:** $u_t \sim \mathcal{N}(\frac{1}{2}, \frac{1}{6})$; **Scenario B:** $u_t \sim \mathcal{N}(\frac{1}{2}, 1)$.
2. Each point on the left plots represents the average across 50 simulations for a specific set of values $(\varepsilon_u; \varepsilon_y)$, displayed on the top of each point (also the unpoisoned cases are depicted in the plots).

Data poisoning: original vs poisoned data



Conclusions

Conclusions

- Data-driven control methods can be used to derive control laws directly from data.
- Data Poisoning is not a new concept in Machine Learning (see Biggio et al. [10]).
- We must pay attention to the security aspects of data-driven methods!

Thank you for listening!

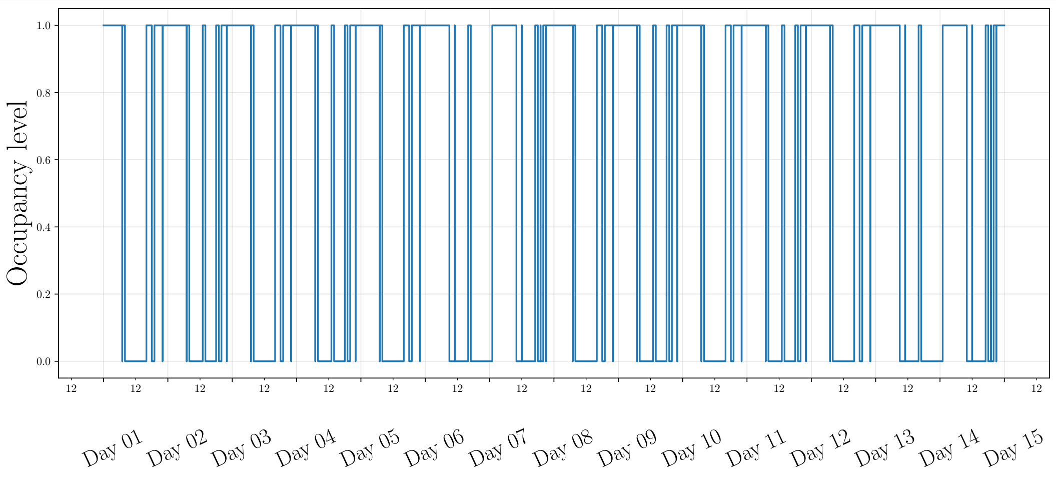
References

1. Campi, Marco C., Andrea Lecchini, and Sergio M. Savaresi. "Virtual reference feedback tuning: a direct method for the design of feedback controllers." Automatica 38.8 (2002): 1337-1346.
2. Hjalmarsson, Hakan, et al. "Iterative feedback tuning: theory and applications." IEEE control systems magazine 18.4 (1998): 26-41.
3. Karimi, A., L. Mikovi, and D. Bonvin. "Iterative correlationbased controller tuning." International journal of adaptive control and signal processing 18.8 (2004): 645-664.
4. Willems, Jan C., et al. "A note on persistency of excitation." Systems & Control Letters 54.4 (2005): 325-329.
5. De Persis, Claudio, and Pietro Tesi. "Formulas for data-driven control: Stabilization, optimality, and robustness." IEEE Transactions on Automatic Control 65.3 (2019): 909-924.
6. EQUA Simulation AB, IDA Indoor Climate and Energy (IDA-ICE), 2020, version 5.0. [Online]. Available: https://www.equa.se/en/ida-ice.
7. Russo, A., Proutiere, A. (2021). Poisoning attacks against data-driven control methods. American Control Conference, 2021.
8. Sinha, Ankur, Pekka Malo, and Kalyanmoy Deb. "A review on bilevel optimization: from classical to evolutionary approaches and applications." IEEE Transactions on Evolutionary Computation 22.2 (2017): 276-295.

References

9. Shen, Xinyue, et al. "Disciplined convex-concave programming." 2016 IEEE 55th Conference on Decision and Control (CDC). IEEE, 2016.
10. Biggio, Battista, Blaine Nelson, and Pavel Laskov. "Poisoning attacks against support vector machines." arXiv preprint arXiv:1206.6389 (2012).

Backup



Occupancy level
1.0
0.8
0.6
0.4
0.2
0.0
12
Day 01
Day 02
Day 03
Day 04
Day 05
Day 06
Day 07
Day 08
Day 09
Day 10
Day 11
Day 12
Day 13
Day 14
Day 15

Attack Formulation

$$\begin{aligned}\max_{\boldsymbol{u}',\boldsymbol{y}'} \quad & \mathcal{A}(\boldsymbol{u},\boldsymbol{y},K(\boldsymbol{u}',\boldsymbol{y}')) \\ \text{s.t.} \quad & K(\boldsymbol{u}',\boldsymbol{y}') \in \arg\min_K \mathcal{L}(\boldsymbol{u}',\boldsymbol{y}',K) \\ & \|\boldsymbol{u}'-\boldsymbol{u}\|_{q_u} \le \delta_u, \quad \|\boldsymbol{y}'-\boldsymbol{y}\|_{q_y} \le \delta_y,\end{aligned}$$

1. Assume the inner problem $K(\boldsymbol{u}',\boldsymbol{y}') \in \arg\min_K \mathcal{L}(\boldsymbol{u}',\boldsymbol{y}',K)$ is convex and sufficiently regular.
   - We can perform single-level reduction [6] and replace the inner problem with its KKT conditions.
2. Then, assume K is parameterized by θ (we will write K_θ). We can conclude that

$$\nabla_\theta \mathcal{L}(\mathbf{u}',\mathbf{a}',K_\theta) = 0 \Rightarrow \nabla_{\mathbf{a}_u}\theta = -(\nabla_{\mathbf{a}_u}\nabla_\theta\mathcal{L})(\nabla_\theta^2\mathcal{L})^{-1}$$

   (similarly also for $\mathbf{a}_y$).
3. **This allows us to find approximate attacks** by using gradient ascent methods.

Attack Formulation

$$
\begin{aligned}
\max_{\boldsymbol{u}',\boldsymbol{y}'} \quad & \mathcal{A}(\boldsymbol{u},\boldsymbol{y},K(\boldsymbol{u}',\boldsymbol{y}')) \\
\text{s.t.} \quad & K(\boldsymbol{u}',\boldsymbol{y}') \in \arg\min_K \mathcal{L}(\boldsymbol{u}',\boldsymbol{y}',K) \\
& \|\boldsymbol{u}'-\boldsymbol{u}\|_{q_u} \leq \delta_u, \quad \|\boldsymbol{y}'-\boldsymbol{y}\|_{q_y} \leq \delta_y,
\end{aligned}
$$

1. Assume the inner problem $K(\boldsymbol{u}',\boldsymbol{y}') \in \arg\min_K \mathcal{L}(\boldsymbol{u}',\boldsymbol{y}',K)$ is convex and sufficiently regular.
   - We can perform single-level reduction [6] and replace the inner problem with its KKT conditions.
2. Then, assume K is parameterized by θ (we will write K_θ). We can conclude that

$$\nabla_\theta \mathcal{L}(\mathbf{u}',\mathbf{a}',K_\theta) = 0 \Rightarrow \nabla_{\mathbf{a}_u}\theta = -(\nabla_{\mathbf{a}_u}\nabla_\theta\mathcal{L})(\nabla_\theta^2\mathcal{L})^{-1}$$

   (similarly also for $\mathbf{a}_y$).
3. **This allows us to find approximate attacks** by using gradient ascent methods.