Robot Learning

Leveraged Gaussian Process Regression

Prof. Songhwai Oh
ECE, SNU

Learning from Do’s and Don’ts

Claim: Using both Do’s and Don’ts can be more beneficial and effective for learning complex tasks.

http://www.c00lstuff.com/1133/Do_s_and_don_ts_with_babies/

We can make learning more
- socially acceptable and
- ethical

by considering what not to do.

Outline

- (Gaussian Process Regression)
- Leveraged Gaussian Process Regression
- Applications
  - Real-Time Autonomous Robot Navigation
  - Leveraged Inverse Reinforcement Learning
  - Learning from Data with Mixed Qualities
  - Leveraged Deep Neural Networks

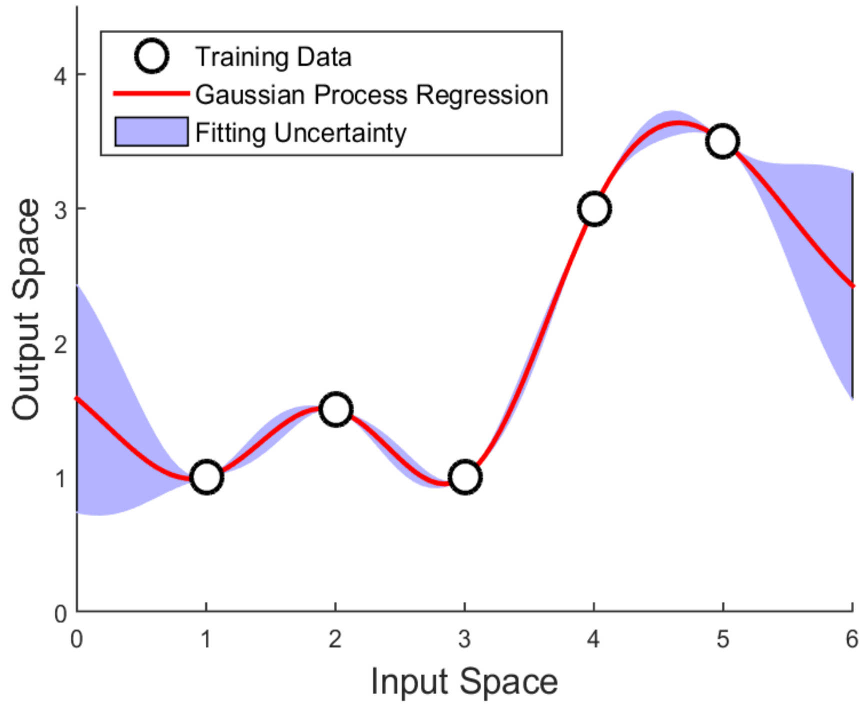
LEVERAGED GAUSSIAN PROCESS REGRESSION

Sungjoon Choi, Eunwoo Kim, Kyungjae Lee, and Songhwai Oh, "**Leveraged Non-Stationary Gaussian Process Regression for Autonomous Robot Navigation**," in Proc. of the IEEE International Conference on Robotics and Automation (ICRA), May 2015.

Regression

The goal of a traditional regression problem is to find a function (regressor) that can best fit given training data.

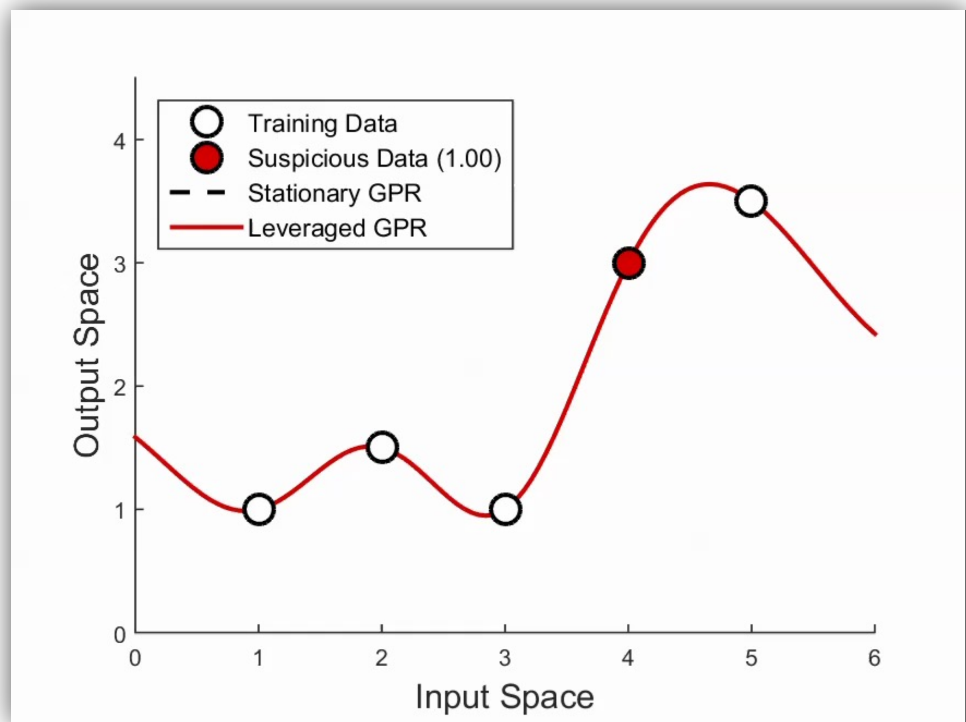
E.g., GPR



Intuitively speaking, each training data works as an attractive force making the regressor as close as possible to such points.

Regression with Positive and Negative Examples

In our formulation, the *positive* data work as an attractive force while the *negative* (suspicious) data work as a repulsive force.



We add an additional ***leverage*** parameter to each training data varying from -1 to +1 where -1 indicates fully negative, +1 indicates fully positive, and the training data with 0 leverage will not affect the resulting regressor.

Leveraged Kernel Function

A (mean-zero) Gaussian process is fully specified by its covariance (kernel) function.

Loosely speaking, a kernel function is usually a decreasing function of some distance measure between two inputs.

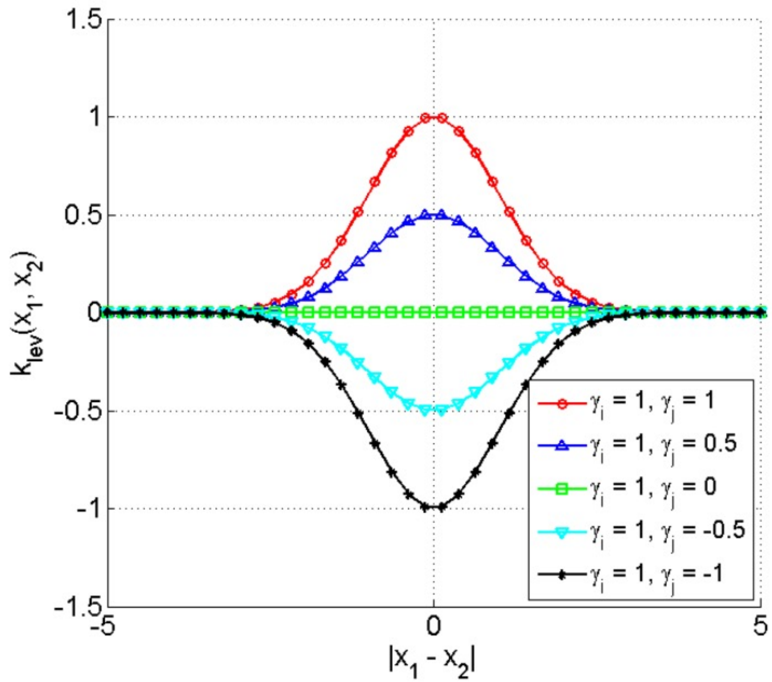
We propose a leveraged kernel function which incorporates not only the inputs of the training data but also the leverage γ of each training data.

$$k(\mathbf{x}_i, \mathbf{x}_j) = (1 - |\gamma_i - \gamma_j|)\, k_{SE}(\mathbf{x}_i,\, \mathbf{x}_j)$$

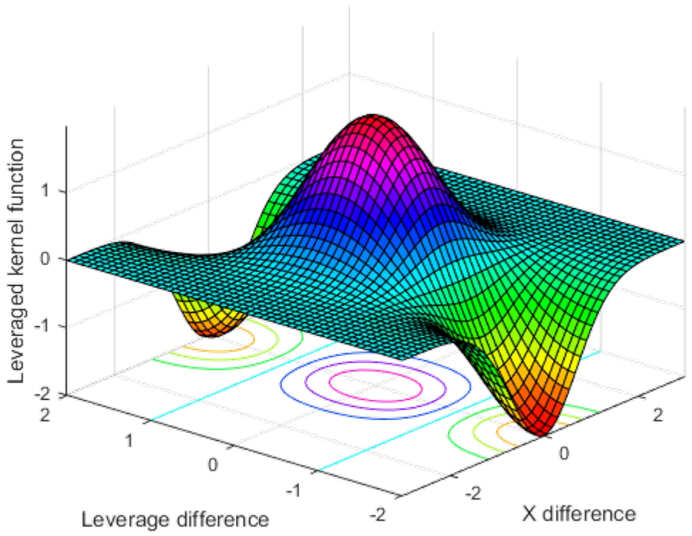
for $\gamma \in [-1, 1]$

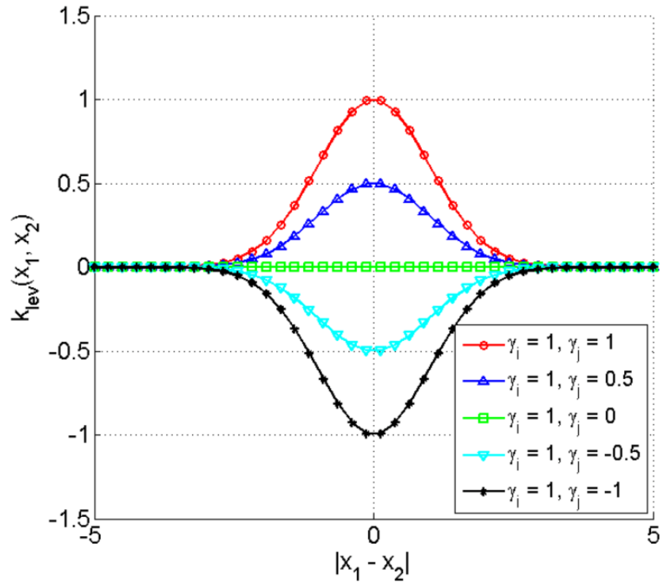
Continuous version:

$$k_{SL}(\mathbf{x}_i, \mathbf{x}_j) = \cos\left(\frac{\pi}{2}(\gamma_i - \gamma_j)\right)\, k_{\text{PSD}}(\mathbf{x}_i,\, \mathbf{x}_j)$$



Leveraged Kernel Function





$$k(\mathbf{x}_i, \mathbf{x}_j) = (1 - |\gamma_i - \gamma_j|)\, k_{SE}(\mathbf{x}_i, \mathbf{x}_j)$$

Each data has its leverage γ whose range is -1 to +1.

When two inputs x_i and x_j have the same leverage value, the kernel function works as an ordinary squared-exponential kernel function.

Between positive ($\gamma = 1$) and negative ($\gamma = -1$) inputs, the correlation decreases as the distance between inputs decreases.

Leveraged Kernel Function: PSD

The positive semi-definiteness of the leveraged kernel function can be shown using the Bochner's theorem [1].

$$k(\mathbf{x}_i, \mathbf{x}_j) = (1 - |\gamma_i - \gamma_j|)\, k_{SE}(\mathbf{x}_i,\, \mathbf{x}_j)$$

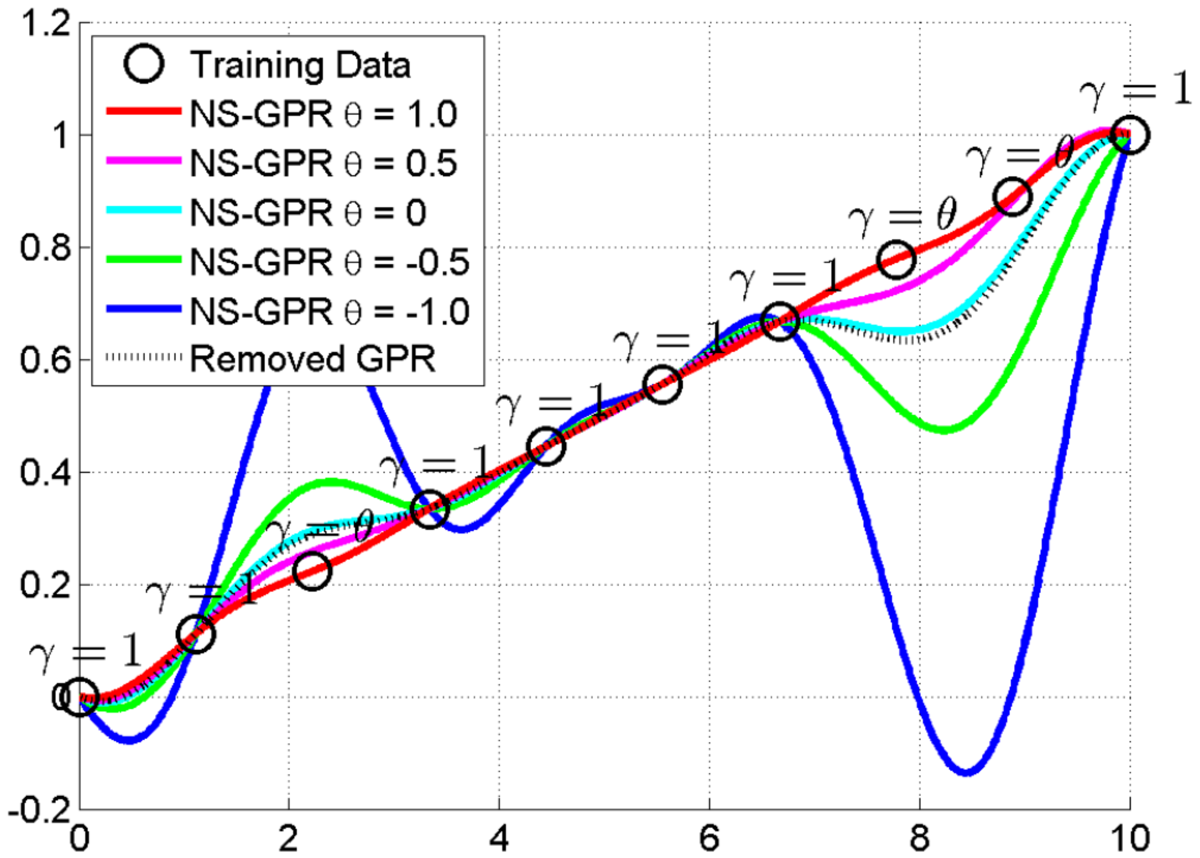
The Bochner's theorem state that if an isotropic kernel function has a non-negative Fourier transform coefficients, it satisfies the positive semi-definiteness.

(Proof Sketch)

1. A tensor product of two valid kernels is valid.
2. Fourier series of $\left(1 - |\gamma_i - \gamma_j|\right) = \begin{cases} 0 & for\ even\ n \\ \frac{4}{(n^2\pi^n)} & for\ odd\ n \end{cases}$
3. Apply Bochner theorem.

[1] S. Bochner, Harmonic analysis and the theory of probability. Courier Dover Publications, 2012.

Leveraged Gaussian Process Regression (LGPR)



The figure shows how the leveraged GPR varies as we vary the leverage of third, eighth, and ninth data points.

Application of Leveraged Gaussian Process Regression (LGPR)

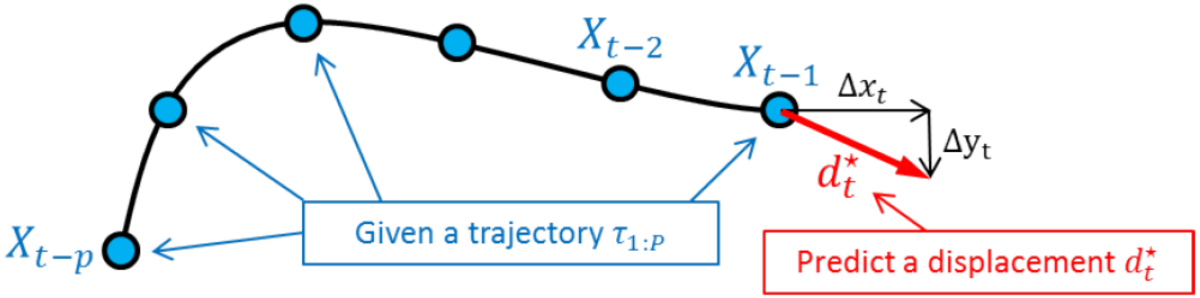
REAL-TIME AUTONOMOUS ROBOT NAVIGATION

Sungjoon Choi, Eunwoo Kim, Kyungjae Lee, Songhwai Oh, "**Real-Time Nonparametric Reactive Navigation of Mobile Robots in Dynamic Environments**," Robotics and Autonomous Systems, vol. 91, pp. 11–24, May 2017.
Sungjoon Choi, Eunwoo Kim, Kyungjae Lee, and Songhwai Oh, "**Leveraged Non-Stationary Gaussian Process Regression for Autonomous Robot Navigation**," in Proc. of the IEEE International Conference on Robotics and Automation (ICRA), May 2015.

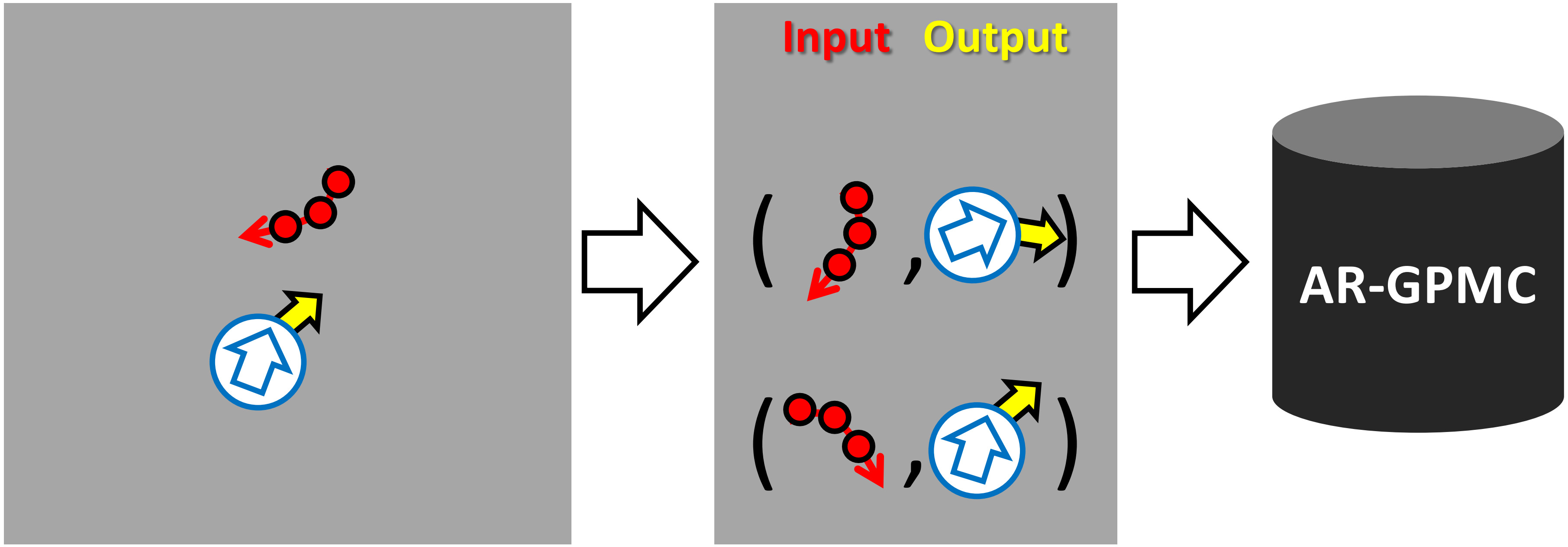
Knightscope K5 Meets a Guy

Real-Time Navigation

Autoregressive Gaussian Process Motion Model



Autoregressive Gaussian Process Motion Controller



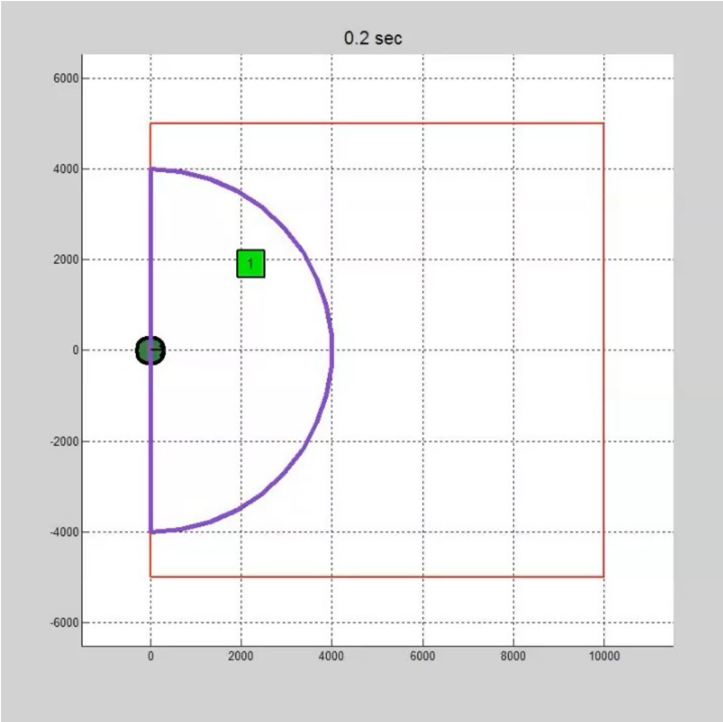
Control a robot to avoid an incoming object (with sample based MPC)

Collect state-action pairs from diverse scenarios

These data will be served as a motion controller

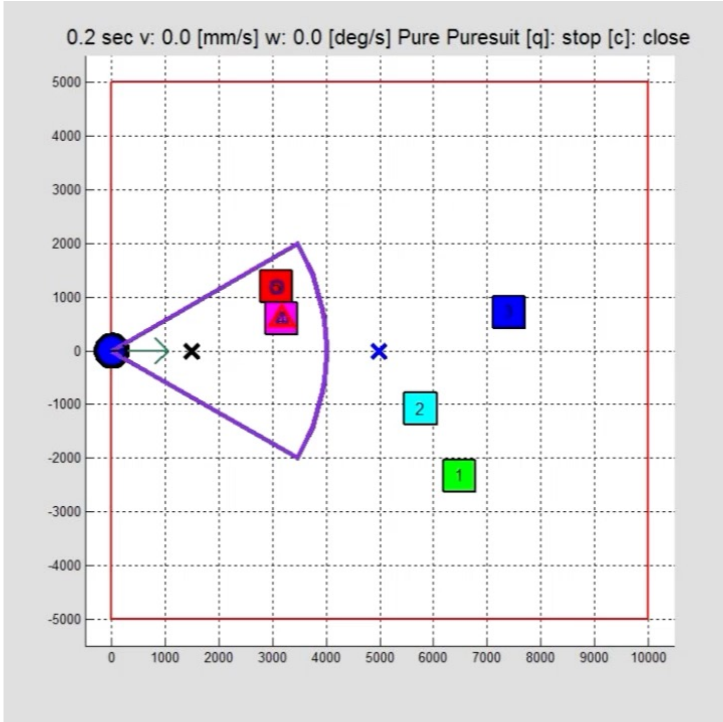
Real-Time Navigation

Training Phase (slow)



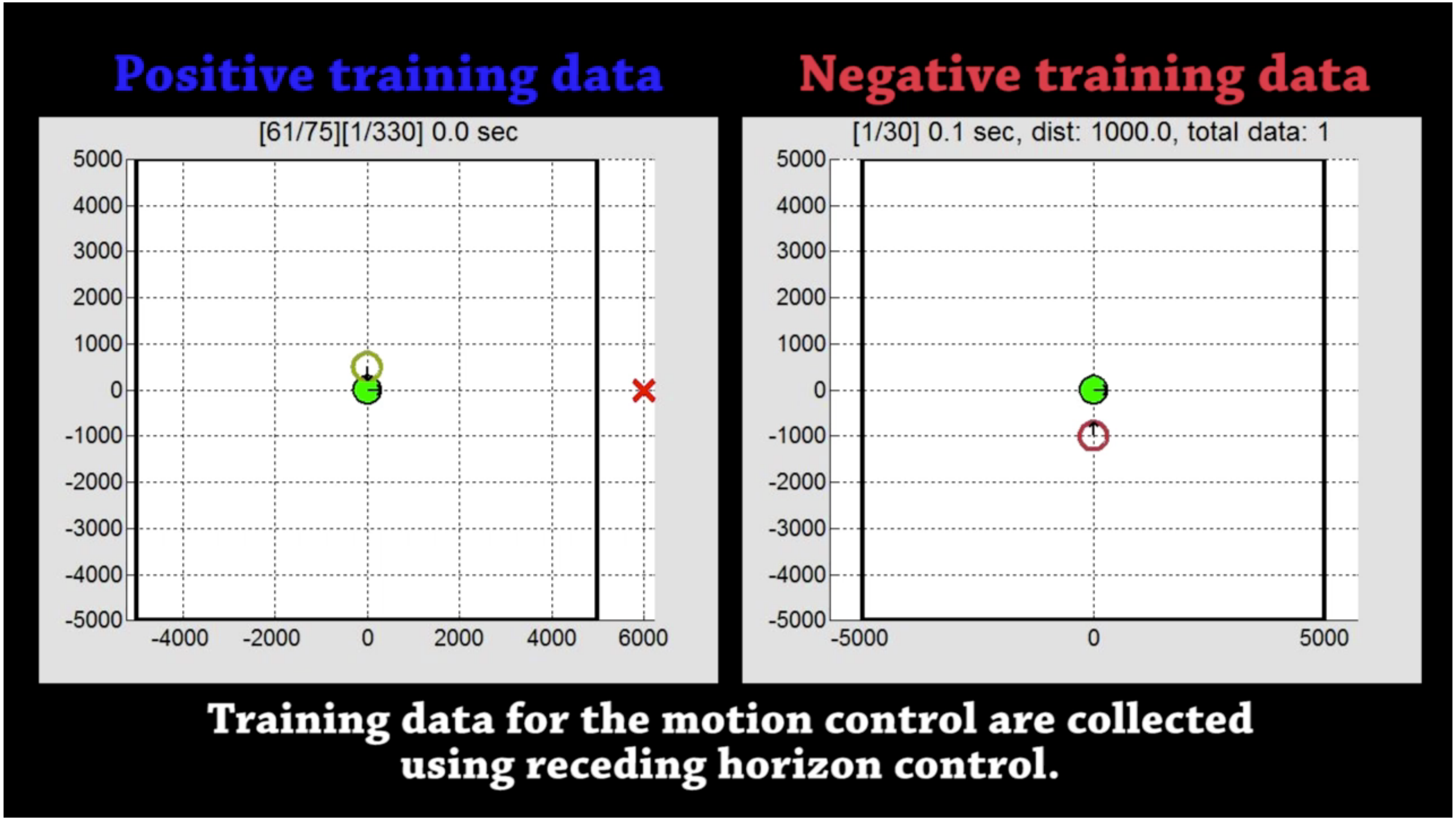
(Trajectory, Control) pairs are collected from diverse scenarios.
This is a time-consuming work.

Execution Phase (fast)

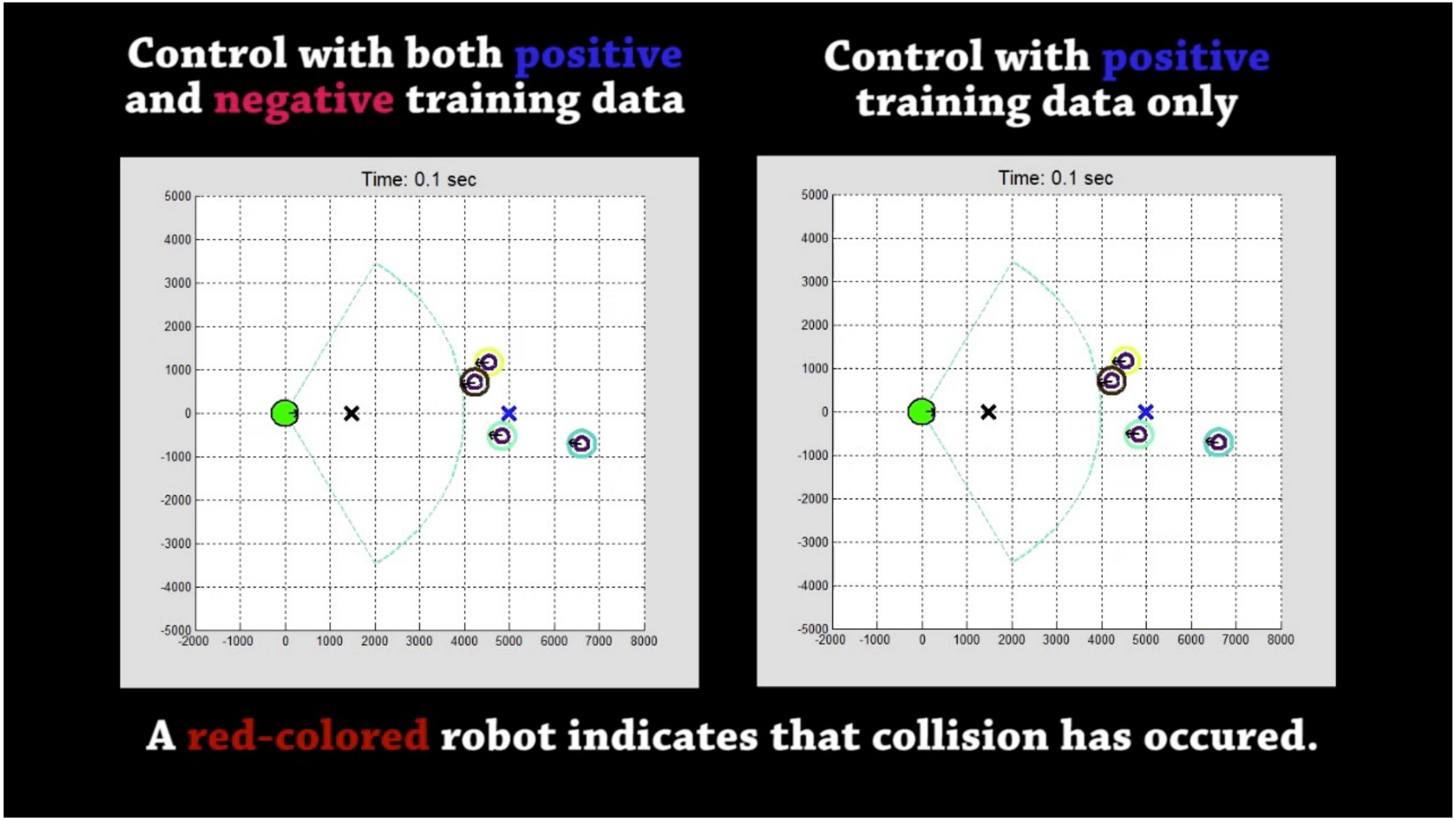


Learned motion controller is used in the execution phase.
It takes less than 50ms to make one control.

LGPR: Learning from Do's and Don'ts



LGPR: Execution



Simulation Results

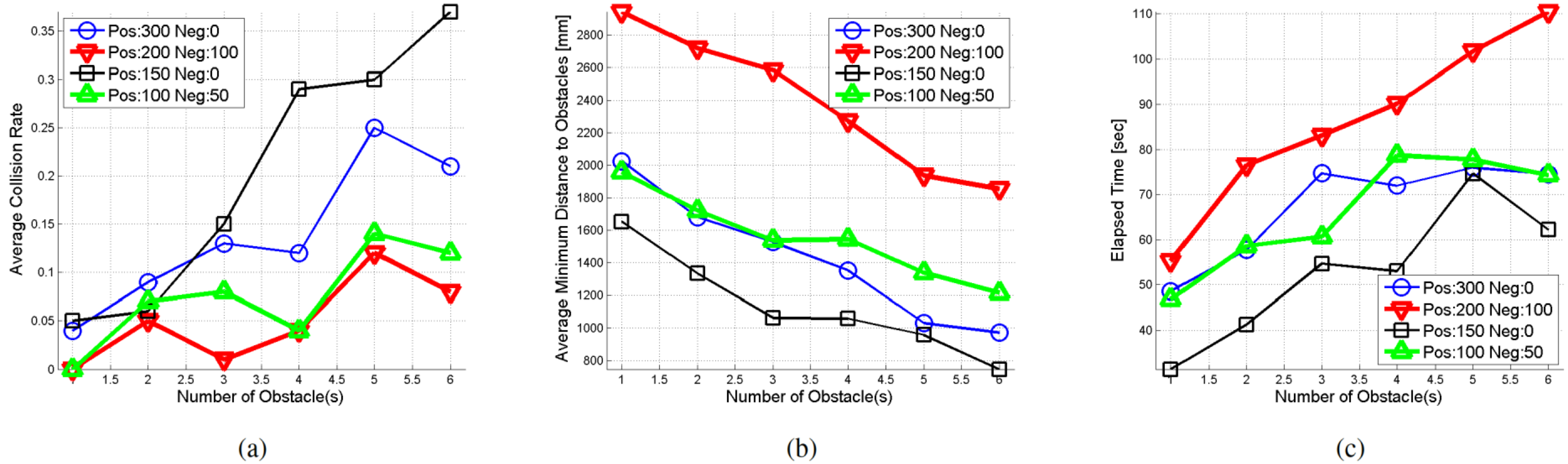
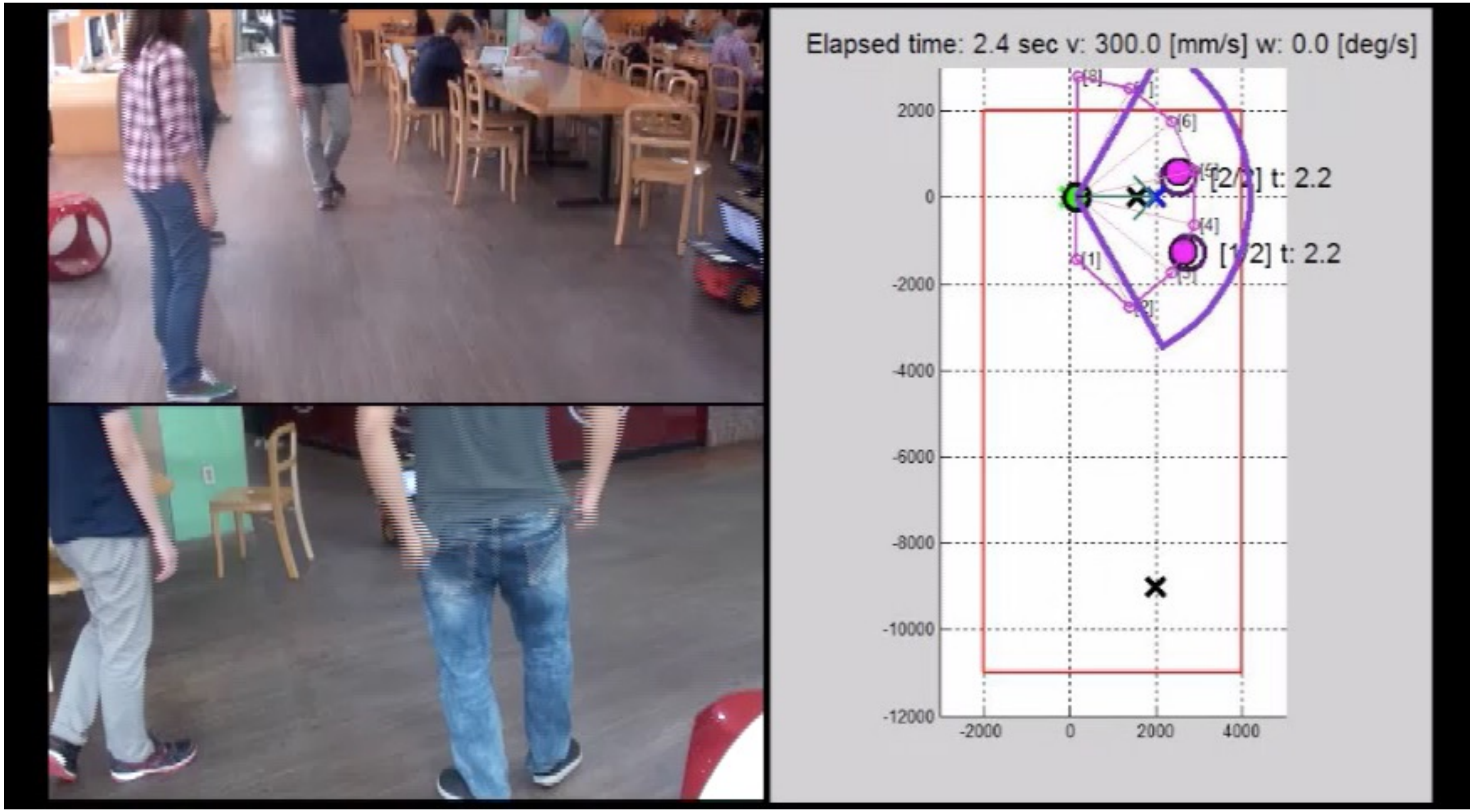


Fig. 3: Simulation results. Thick solid lines indicate results using both positive and negative data. (a) Average collision rates of different scenarios for different numbers of obstacles. (b) Average minimum distances to obstacles. (c) Elapsed times to the goal.

We can see that learning from both **do's** and **don'ts** shows superior performance with respect to the average collision rate compared to learning with only **do's**.

Experiments



Sungjoon Choi, Eunwoo Kim, Kyungjae Lee, and Songhwai Oh, "**Leveraged Non-Stationary Gaussian Process Regression for Autonomous Robot Navigation**," in Proc. of the IEEE International Conference on Robotics and Automation (ICRA), May 2015.

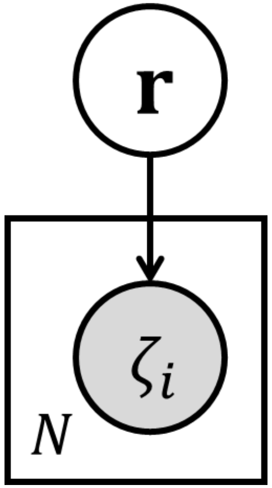
Application of Leveraged Gaussian Process Regression (LGPR)

LEVERAGED INVERSE REINFORCEMENT LEARNING

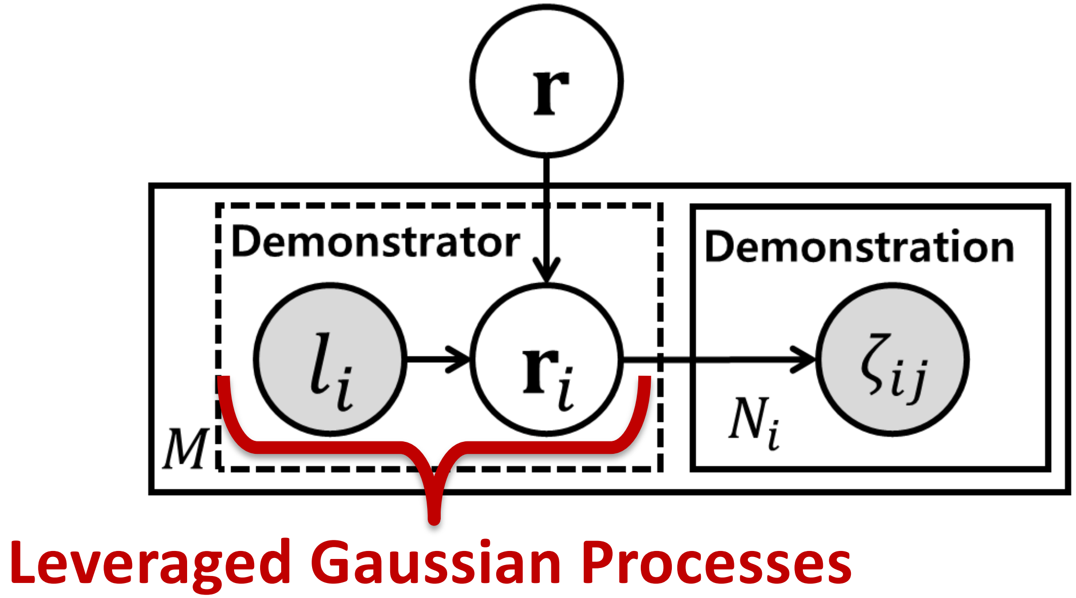
Kyungjae Lee, Sungjoon Choi, and Songhwai Oh, "**Inverse Reinforcement Learning with Leveraged Gaussian Processes**," in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Oct. 2016.

Leveraged Inverse Reinforcement Learning

Original Behavior Model



Proposed Behavior Model



Leveraged Inverse Reinforcement Learning

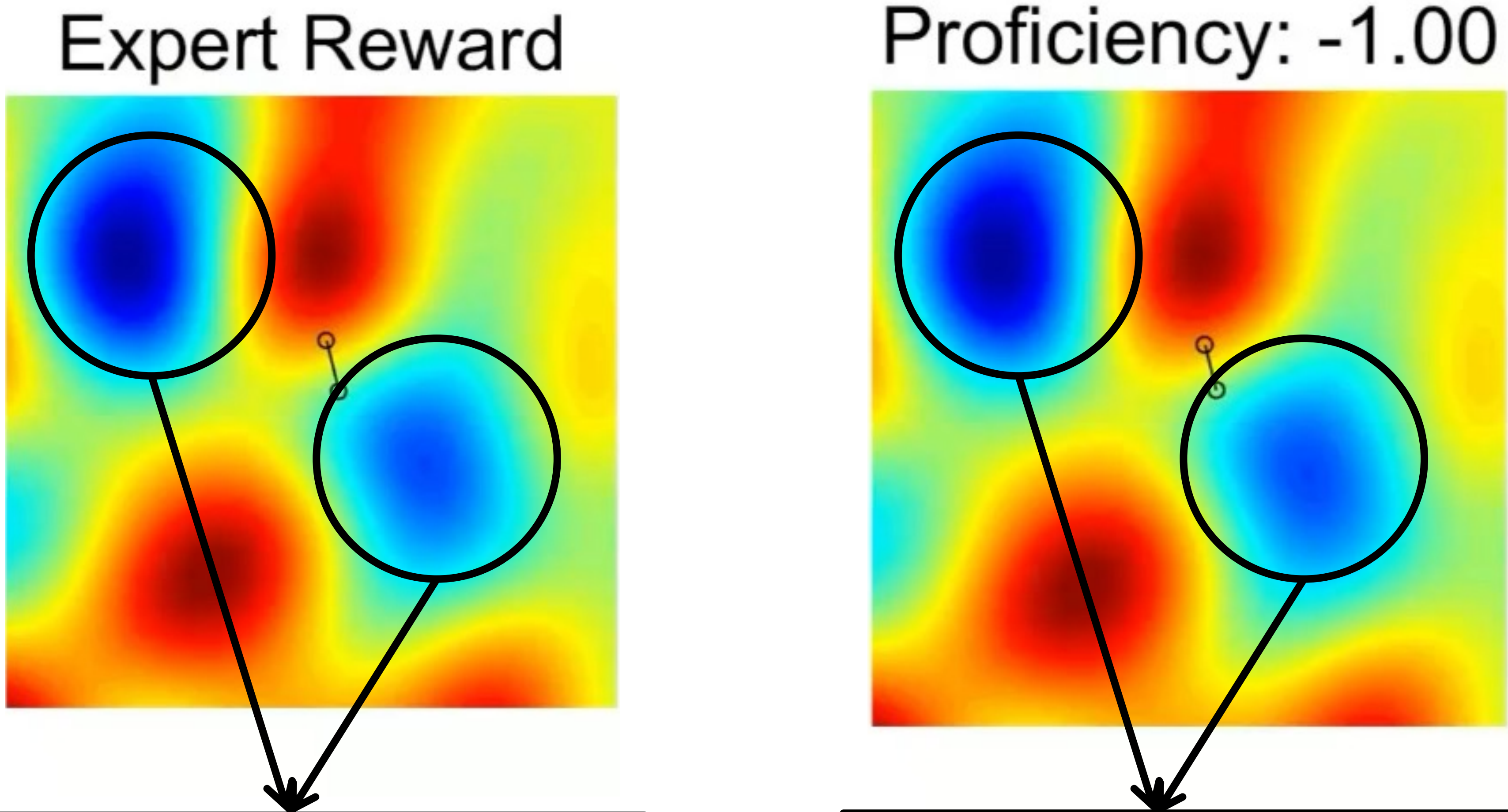
Problem Formulation

- Objective Function

$$\max_{\boldsymbol{u},\boldsymbol{\theta}=\{\boldsymbol{\beta},\Lambda\}} \sum_{\boldsymbol{i}} \underbrace{\log \boldsymbol{P}(\mathcal{D}|\boldsymbol{r} = \boldsymbol{K}_{\boldsymbol{ru}}\boldsymbol{K}_{\boldsymbol{uu}}^{-1}\boldsymbol{u}, \boldsymbol{l}_{\boldsymbol{i}})}_{\text{IRL Likelihood}} + \underbrace{\log \boldsymbol{P}(\boldsymbol{u}|\boldsymbol{X}_{\boldsymbol{u}}, \boldsymbol{\theta})}_{\text{LGP Prior}} + \underbrace{\log \boldsymbol{P}(\boldsymbol{\theta}|\boldsymbol{X}_{\boldsymbol{u}})}_{\text{Prior}}$$

- $\log \boldsymbol{P}(\mathcal{D}|\boldsymbol{r} = \boldsymbol{K}_{\boldsymbol{ru}}\boldsymbol{K}_{\boldsymbol{uu}}^{-1}\boldsymbol{u}, \boldsymbol{l}) = \sum_i \sum_j \sum_t \boldsymbol{Q}_i(\boldsymbol{s}_{jt}, \boldsymbol{a}_{jt}) - \boldsymbol{V}_i(\boldsymbol{s}_{jt})$
- $\log \boldsymbol{P}(\boldsymbol{u}|\boldsymbol{X}_{\boldsymbol{u}}, \boldsymbol{\theta}) = -\frac{1}{2}\boldsymbol{u}^T\boldsymbol{K}_{\boldsymbol{uu}}^{-1}\boldsymbol{u} - \frac{1}{2}\log|\boldsymbol{K}_{\boldsymbol{uu}}| - \frac{\boldsymbol{n}}{2}\log 2\boldsymbol{\pi}$
- $\log \boldsymbol{P}(\boldsymbol{\theta}|\boldsymbol{X}_{\boldsymbol{u}}) = -\frac{1}{2}\boldsymbol{tr}(\boldsymbol{K}_{\boldsymbol{uu}}^{-2}) - \sum_k(\boldsymbol{\lambda}_k + \mathbf{1})$
- Objective function is maximized by a gradient ascent method

Leveraged Inverse Reinforcement Learning



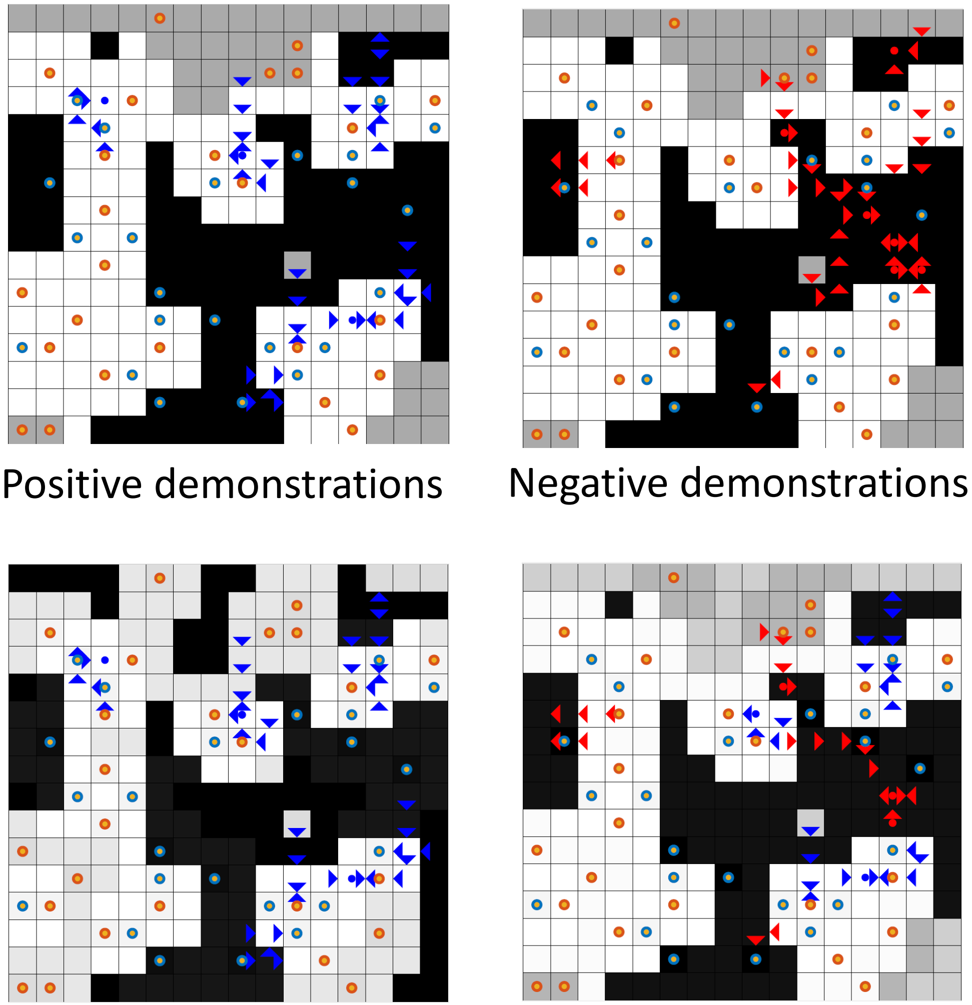
Lack of demonstrations near the low reward region

Negative demonstrations effectively provide information near the low reward region

Benefits of Negative Demonstrations

Objectworld

- First row: positive and negative demonstrations
- Second row: reconstruction results of GPRIL and LIRL (proposed)
- The result from LIRL is more accurate than GPIRL
- Negative demonstrations provide information about the low reward region



Positive demonstrations | Negative demonstrations

GPIRL result | LIRL result

Application of Leveraged Gaussian Process Regression (LGPR)

LEARNING FROM DATA WITH MIXED QUALITIES

Sungjoon Choi, Kyungjae Lee, and Songhwai Oh, "**Robust Learning from Demonstrations with Mixed Qualities Using Leveraged Gaussian Processes**," IEEE Transactions on Robotics, vol. 35, no. 3, pp. 564-576, Jun. 2019.
Sungjoon Choi, Kyungjae Lee, Songhwai Oh, "**Robust Learning From Demonstration Using Leveraged Gaussian Processes and Sparse Constrained Optimization**", in IEEE Conference on Robotics and Automation (ICRA), May 2016. **(Best Conference Paper Award Finalist)**

Leverage Optimization

Leverage Optimization

Leverage Optimization

Why is it **important**?

Suppose that we want to teach an autonomous car how to drive.

There are tons of driving demonstrations.

However, only few of them are actually useful once we assume that demonstrations are collected from the right experts.

The proposed **leverage optimization method** can effectively handle this issue!

Leverage Optimization (PLM)

The key intuition behind the **leverage optimization** is that we cast the leverage optimization problem into a **model selection problem** in Gaussian process regression.

$$\log p(\mathbf{y}|\mathbf{X},\theta) = -\frac{1}{2}\mathbf{y}^T\mathbf{K}_{\mathbf{X}}^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{K}_{\mathbf{X}}| - \frac{n}{2}\log 2\pi.$$

However, the number of **leverage parameters** is equivalent to the number of training data.

To handle this issue, we propose a **sparse constrained leverage optimization** where we assume that the majority of leverage parameters are +1.

Leverage Optimization: PLM

Resulting optimization problem becomes:

$$\underset{\gamma}{\text{minimize}} \quad -L(\mathbf{y}|\mathbf{X}, \bar{\gamma} + \mathbf{1}_n) + \lambda \|\bar{\gamma}\|_1$$

where $-\mathrm{L}(\cdot)$ is the negative log likelihood and $\bar{\gamma} = \gamma - 1$.

Using **proximal linearized minimization (PLM)** [1], the update rule for solving above optimization is

$$\bar{\gamma}^{k+1} \in \text{prox}_{1/t}^{\lambda\|\bar{\gamma}\|_1} \left(\bar{\gamma}^k - t\nabla_{\bar{\gamma}} \left(-L(\mathbf{y}|\, \mathbf{X}, \bar{\gamma}^k + \mathbf{1}_n) \right) \right)$$

where the proximal mapping becomes soft-thresholding:

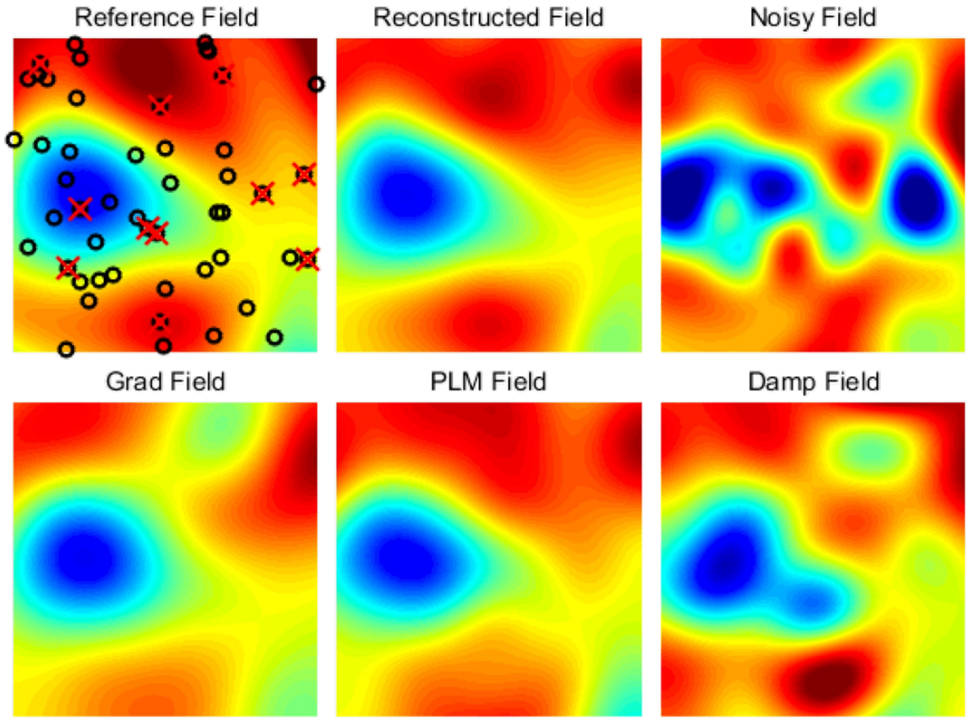
$$\text{prox}^{\lambda\|\gamma-1\|_1}(\gamma) = \begin{cases} \gamma - \lambda & \text{if } \gamma > 1 + \lambda, \\ \gamma + \lambda & \text{if } \gamma < 1 - \lambda, \\ 1 & \text{otherwise.} \end{cases}$$

[1] J. Bolte, S. Sabach, and M. Teboulle, "**Proximal alternating linearized minimization for nonconvex and nonsmooth problems**," Mathematical Programming, vol. 146, no. 1-2, pp. 459–494, 2014.

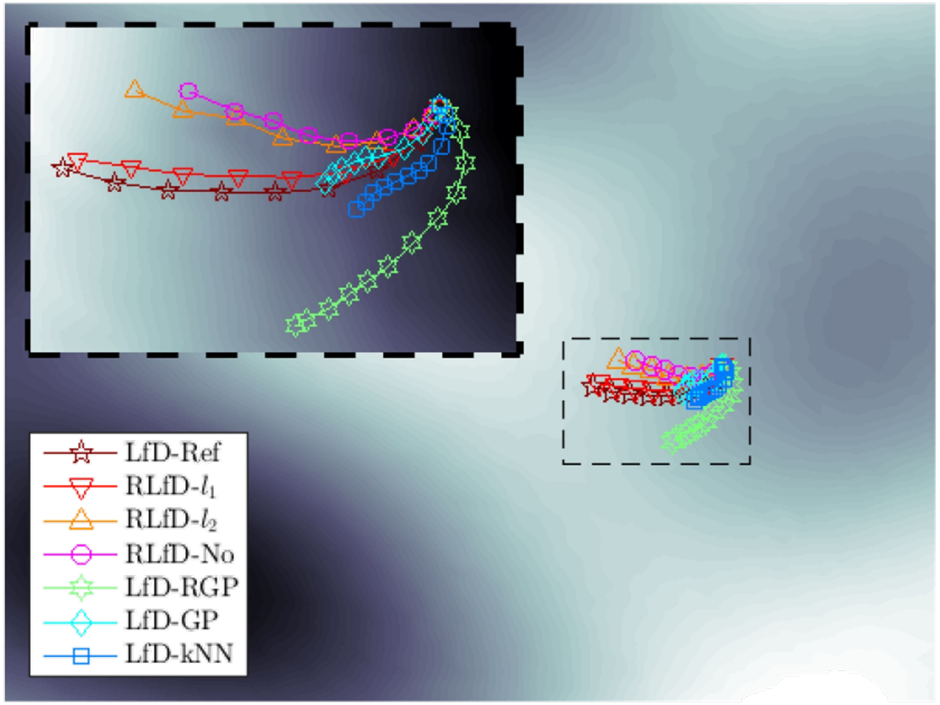
Leverage Optimization (PLM)

Experiments

Sensory field reconstruction



Planar navigation



Noisy Field: Using both (+) and (-) examples
Grad Field: Using leverage optimization without the sparsity constraint
PLM Field: Proposed method
Damp Field: Proposed method but with the l2-norm

Leverage Optimization: DVM

We propose a new leverage optimization method by doubling the leverage parameters to positive and negative parts.

Furthermore, we assume multiple demonstrations are collected from each demonstrator.

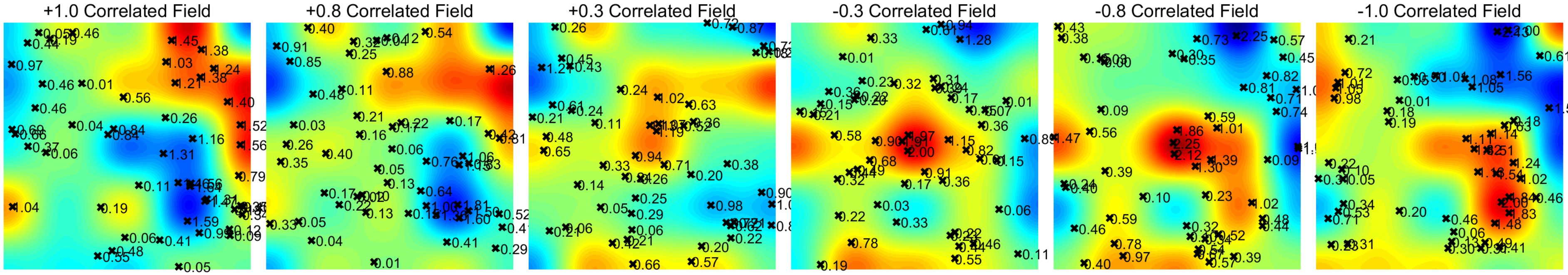
$$\begin{aligned} \underset{\bar{\mathrm{l}}^{+},\bar{\mathrm{l}}^{-}}{\text{minimize}} \quad & \frac{1}{2}\mathbf{y}^{T}\mathbf{K}^{-1}\mathbf{y} + \frac{1}{2}\log\det(\mathbf{K}) + \boxed{\lambda_1 \|\mathbf{1}_m - \bar{\mathrm{l}}^{+}\|_1} \\ \text{subject to} \quad & \mathrm{l} = A\,(\bar{\mathrm{l}}^{+} - \bar{\mathrm{l}}^{-}) \\ & \bar{\mathrm{l}}_i^{+} \times \bar{\mathrm{l}}_i^{-} = 0,\ i = 1, 2, \ldots, m \\ & 0 \leq \bar{\mathrm{l}}_i^{+}, \bar{\mathrm{l}}_i^{-} \leq 1,\ i = 1, 2, \ldots, m \end{aligned}$$

By doing so, we have two major benefits:

1. As the l_1-norm regularizer is only on the **positive parts** of the leverages, negative parts of the leverages can be optimized more accurately.
2. By applying the doubling variable method (DVM), proximal mapping is no longer needed.

Sungjoon Choi, Kyungjae Lee, and Songhwai Oh, "**Robust Learning from Demonstrations with Mixed Qualities Using Leveraged Gaussian Processes**," IEEE Transactions on Robotics, vol. 35, no. 3, pp. 564-576, Jun. 2019.

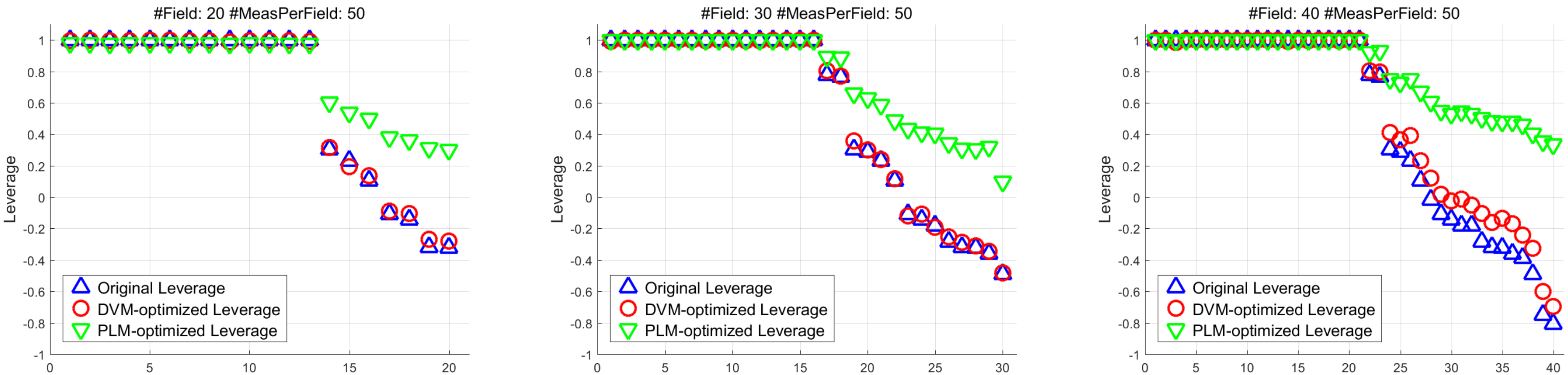
Experiment



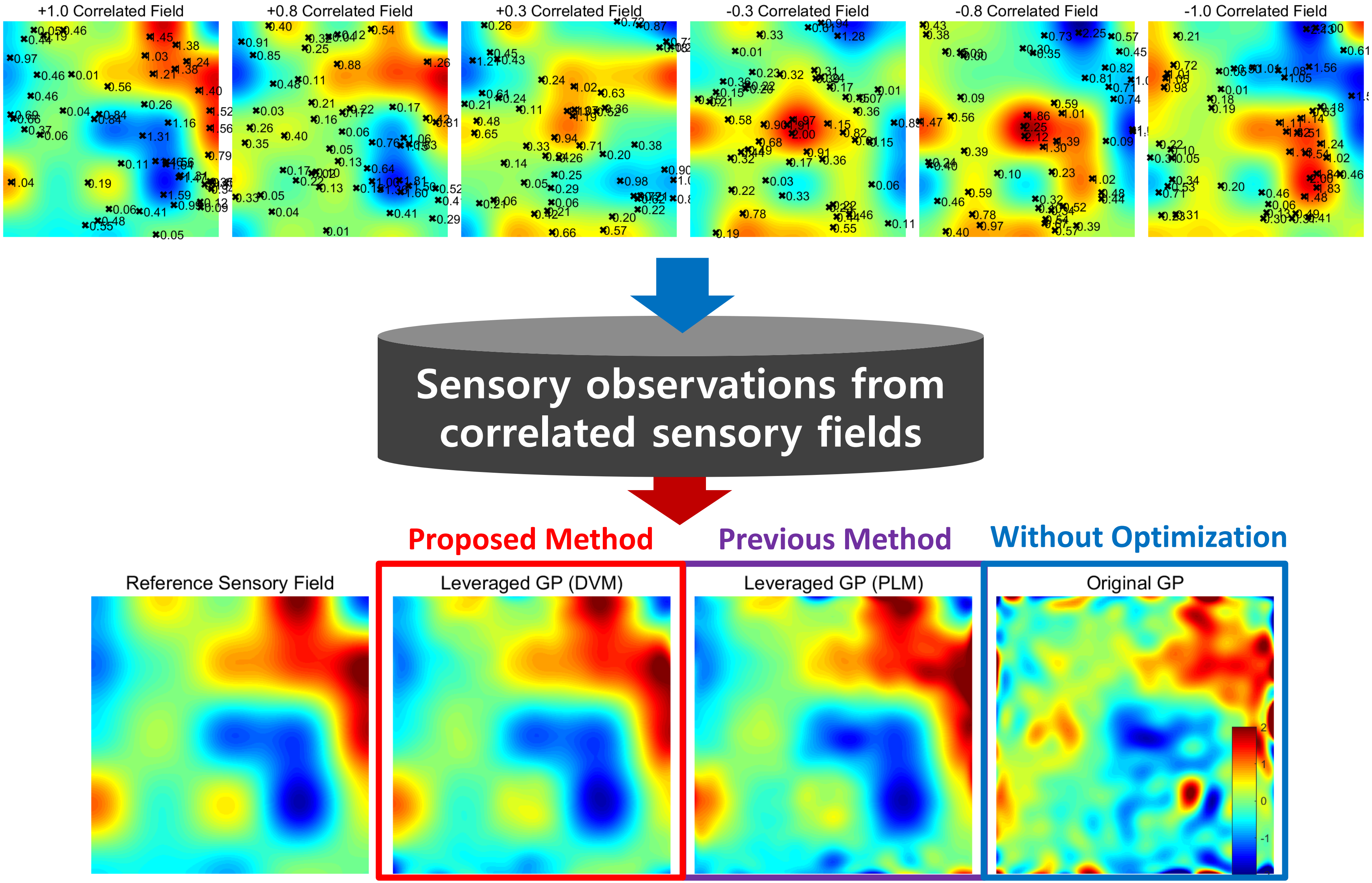
Collect observations

Sensory observations from correlated sensory fields

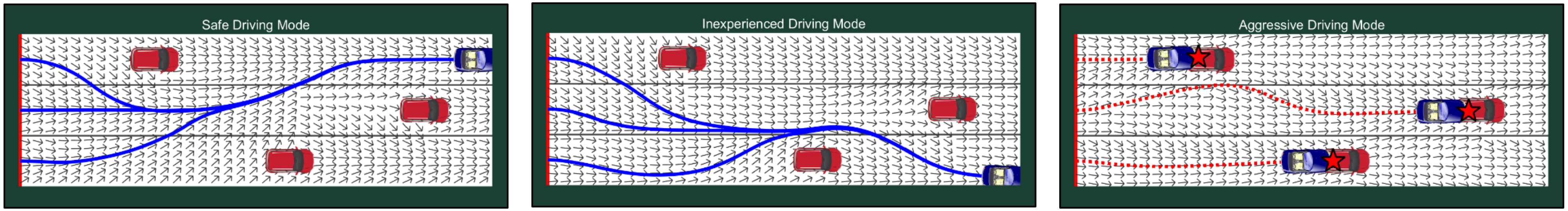
Leverage optimization

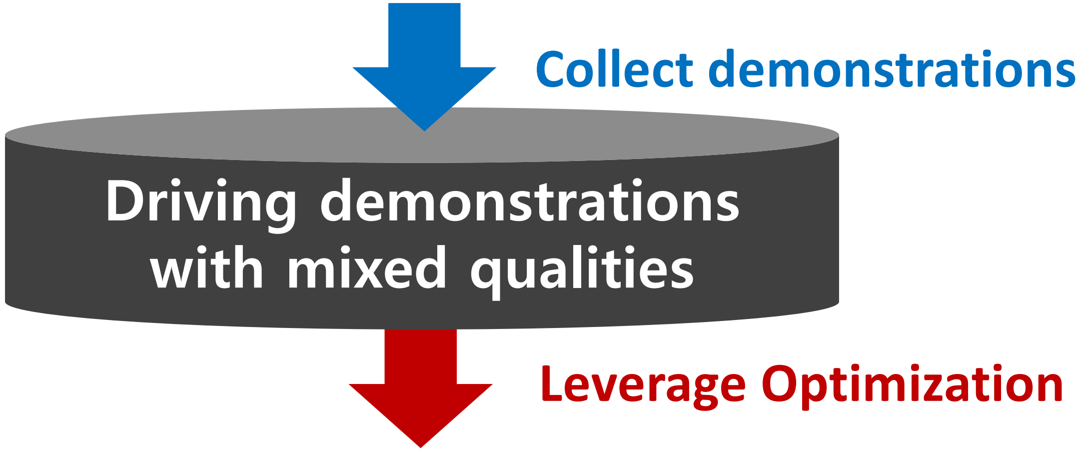


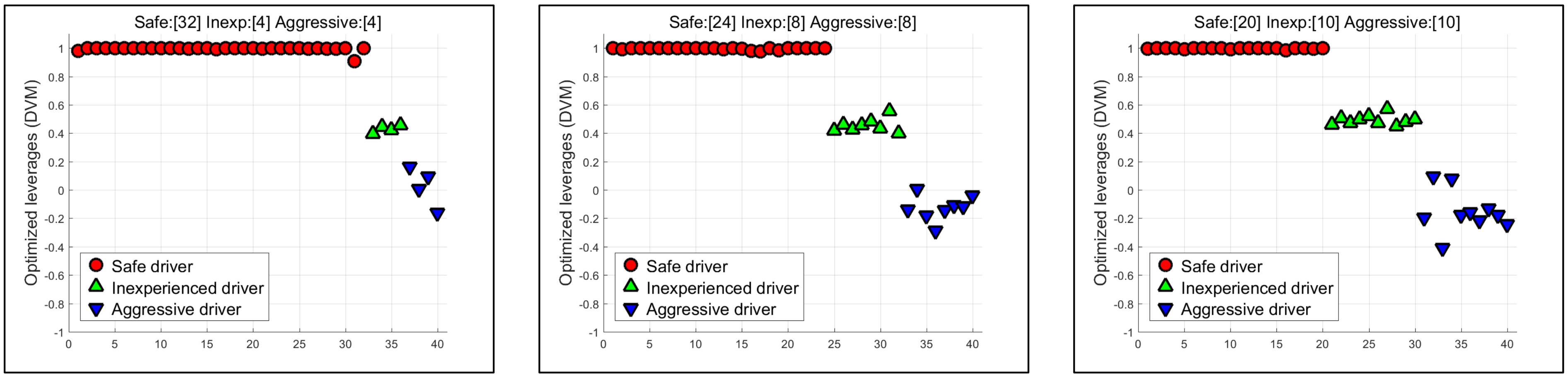
Experiment


+1.0 Correlated Field
+0.8 Correlated Field
+0.3 Correlated Field
-0.3 Correlated Field
-0.8 Correlated Field
-1.0 Correlated Field
Sensory observations from correlated sensory fields
Proposed Method
Previous Method
Without Optimization
Reference Sensory Field
Leveraged GP (DVM)
Leveraged GP (PLM)
Original GP

Experiment

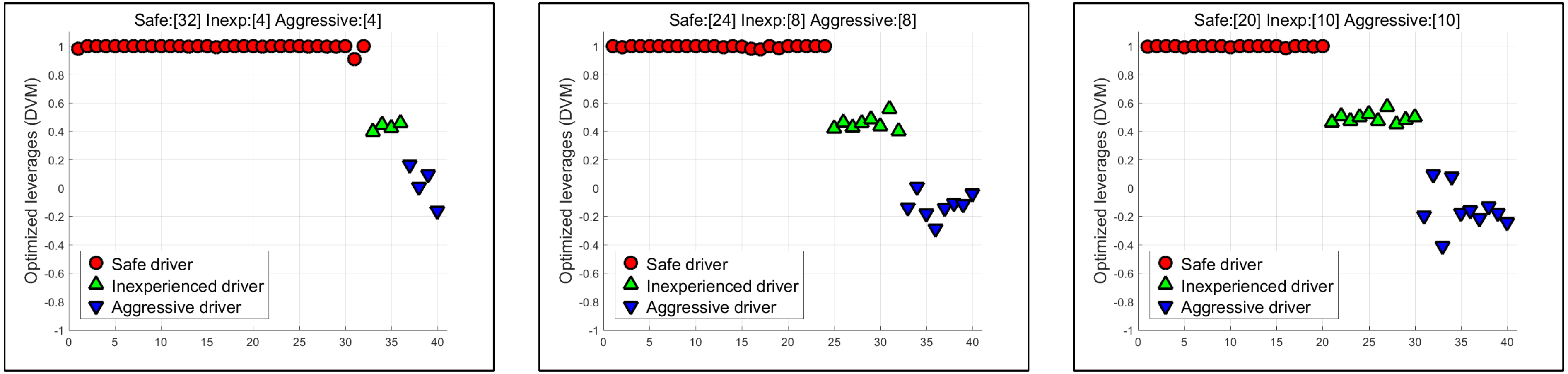

Safe Driving Mode
Inexperienced Driving Mode
Aggressive Driving Mode


Collect demonstrations
Driving demonstrations
with mixed qualities
Leverage Optimization

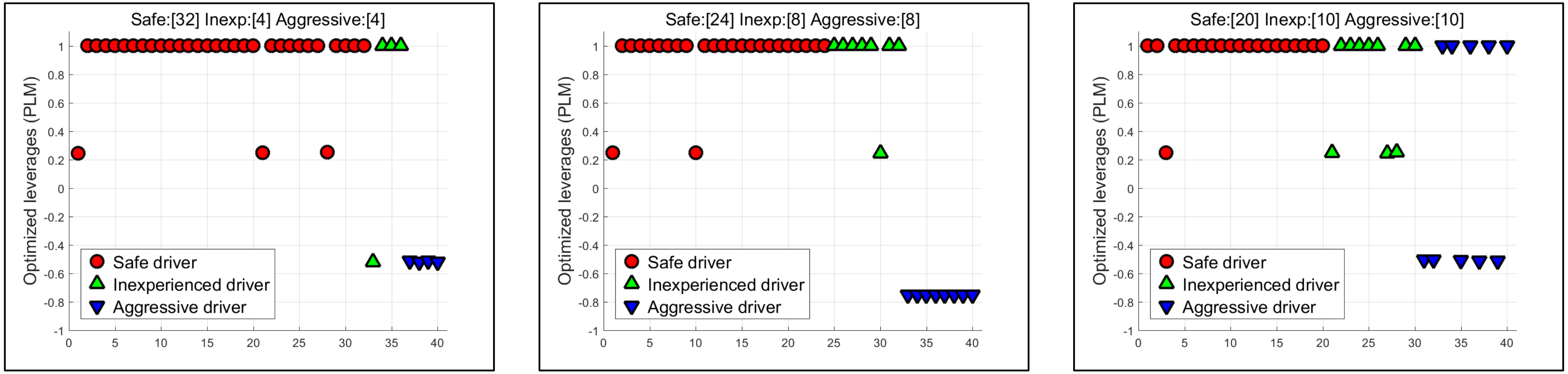

Safe:[32] Inexp:[4] Aggressive:[4]
Safe:[24] Inexp:[8] Aggressive:[8]
Safe:[20] Inexp:[10] Aggressive:[10]
Optimized leverages (DVM)
Safe driver
Inexperienced driver
Aggressive driver

Experiment

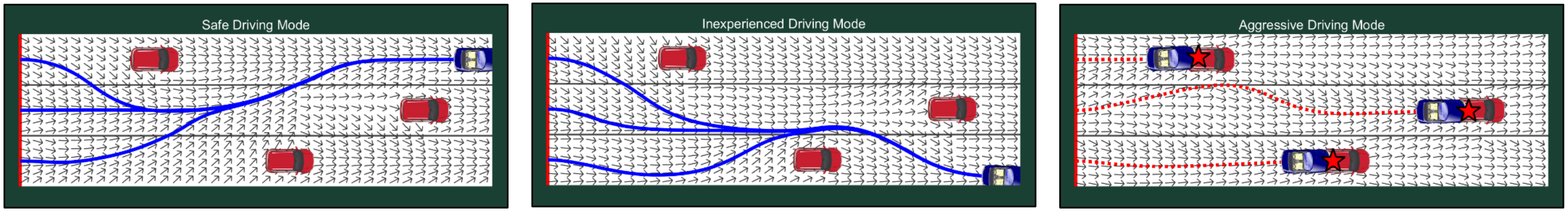
Proposed Method (DVM)

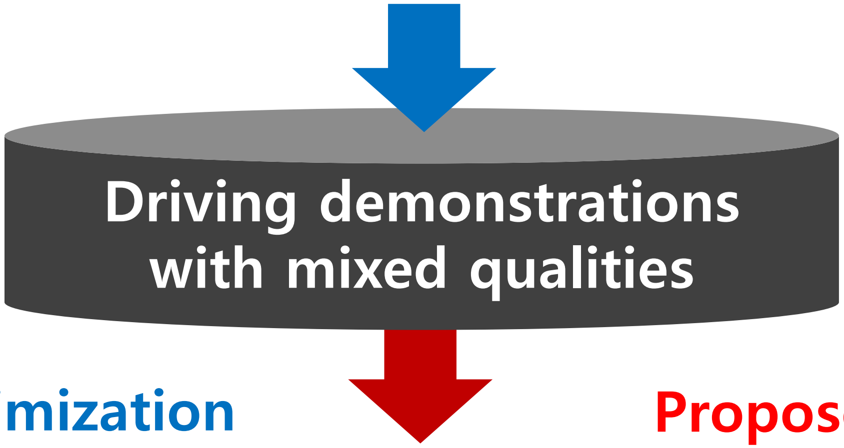


Previous Method (PLM)



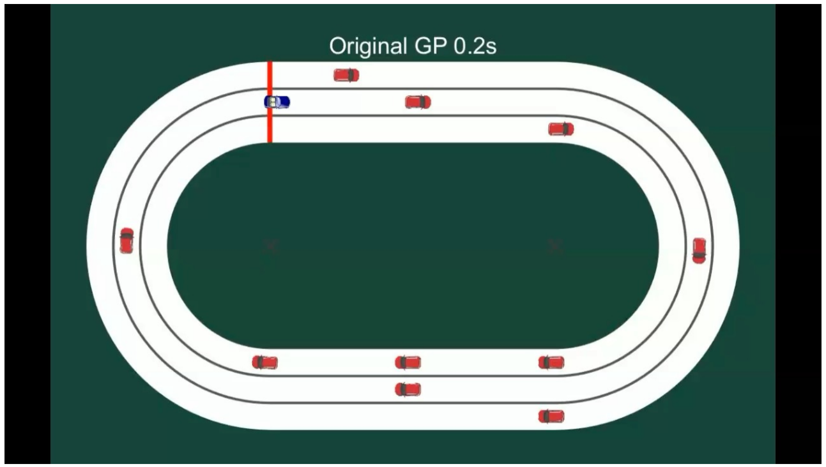
Experiment

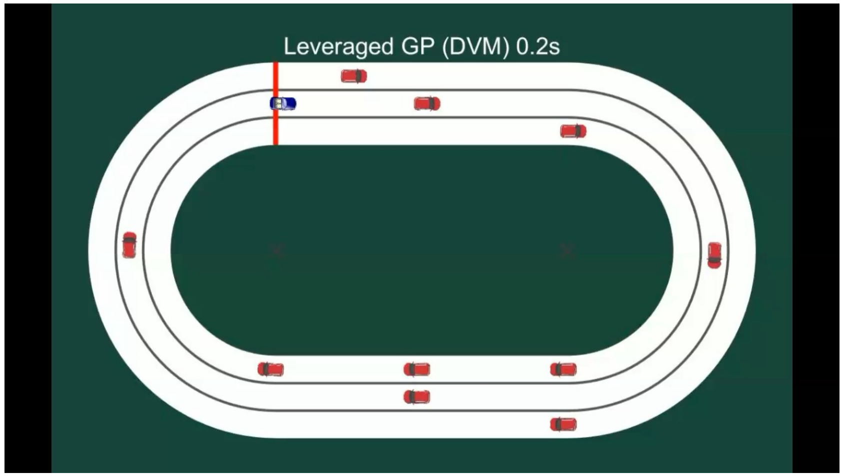




Without Optimization

Proposed Method





Application of Leveraged Gaussian Process Regression (LGPR)

LEVERAGED DEEP NEURAL NETWORKS

Sungjoon Choi, Kyungjae Lee, and Songhwai Oh, "**Scalable Robust Learning from Demonstration with Leveraged Deep Neural Networks**," in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sep. 2017.

Leveraged Deep Neural Network

We propose a **leveraged deep neural network** by proposing a leveraged cost function by interpreting the objective function of the leveraged Gaussian processes using the **representer theorem**.

$$
\begin{aligned}
J_{SE}[f] &= \frac{1}{2\sigma_n^2}\sum_{i=1}^{n}(y_i - f(\mathbf{x}_i))^2 + \frac{1}{2}||f||_{\mathcal{H}}^2 \\
&=: E_{SE}(f|D) + R_{SE}(f),
\end{aligned}
$$

$$
\begin{aligned}
J_L[f] =& \frac{1}{2}||f||_{\mathcal{H}}^2 \\
&+ \frac{1}{2\sigma_n^2}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{n}\alpha_j \cos\left(\frac{\pi}{2}(\gamma_i - \gamma_j)\right) k_{SE}(\mathbf{x}_i, \mathbf{x}_j)\right)^2 \\
=& \frac{1}{2}||f||_{\mathcal{H}}^2 \\
&+ \frac{1}{2\sigma_n^2}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{n}\alpha_j \gamma_i \gamma_j k_{SE}(\mathbf{x}_i, \mathbf{x}_j)\right)^2 \\
=:& R_L(f) + \boxed{E_L(f|D).}
\end{aligned}
$$

* With a simplifying assumption that γ_i and γ_j are either 1 or -1.

$$
\boxed{E_L(f|D)} = \frac{1}{2\sigma_n^2}\sum_{i=1}^{n}\left(\gamma_i y_i - \sum_{j=1}^{n}\gamma_j \alpha_j k_{SE}(\mathbf{x}_i, \mathbf{x}_j)\right)^2 .
$$

Leveraged Deep Neural Network

$$E_L(f|D) = \frac{1}{2\sigma_n^2} \sum_{i=1}^{n} \left(\boxed{\gamma_i y_i} - \sum_{j=1}^{n} \gamma_j \alpha_j k_{SE}(\mathbf{x}_i, \mathbf{x}_j) \right)^2 .$$

Leveraged Cost Function

DNN

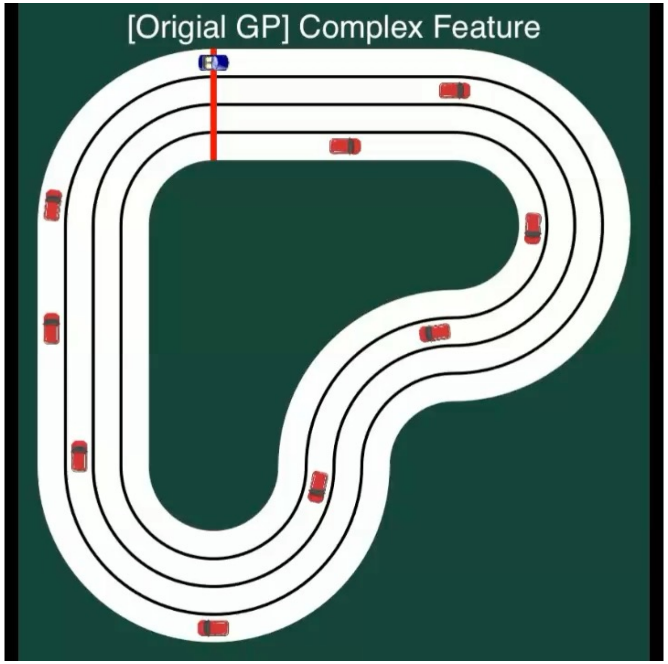
regularizer

$$\mathcal{L}_\theta = \frac{1}{n} \sum_{i=1}^{n} |\gamma_i| (f_\theta(\mathbf{x}_i) - \gamma_i y_i)^2 + R(\theta)$$

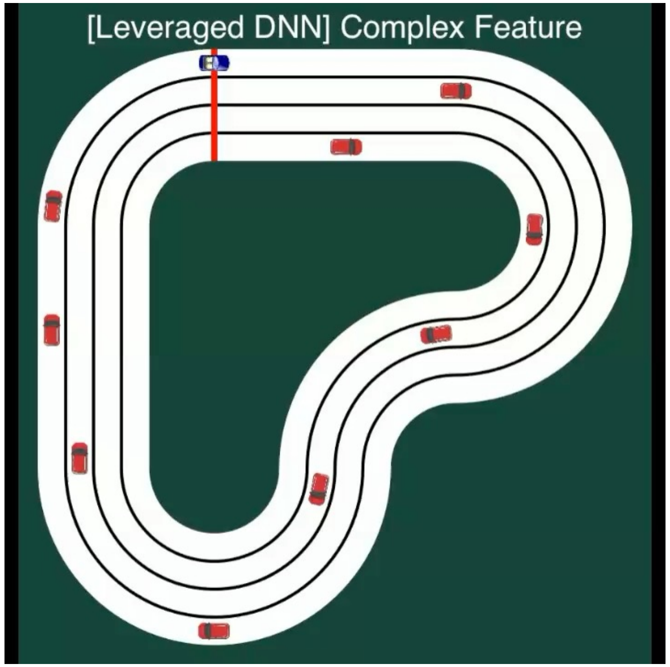
leverage input lev*output

Experiment

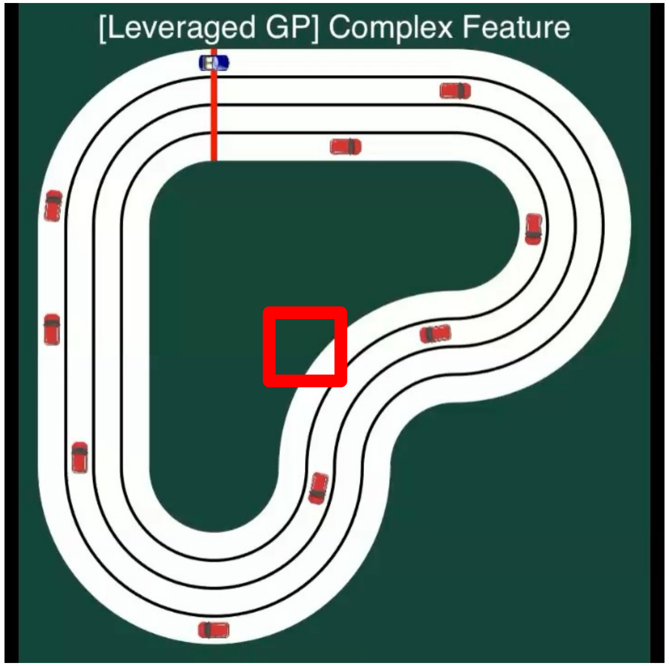
Gaussian process without optimization



Leveraged deep neural network (20,000 demos)



Leveraged Gaussian process (5,000 demos)



Leveraged GPR: Learning from Do's and Don'ts

Leveraged Gaussian process regression:

- Regression + Classification

Negative examples

- speed up learning
- restrict learner's actions
- (make intelligent systems more **socially acceptable** and **ethical**)

References:

- Sungjoon Choi, Kyungjae Lee, and Songhwai Oh, "**Robust Learning from Demonstrations with Mixed Qualities Using Leveraged Gaussian Processes**," IEEE Transactions on Robotics, vol. 35, no. 3, pp. 564-576, Jun. 2019.
  Sungjoon Choi, Kyungjae Lee, and Songhwai Oh, "**Scalable Robust Learning from Demonstration with Leveraged Deep Neural Networks**," in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sep. 2017.
- Sungjoon Choi, Kyungjae Lee, Songhwai Oh, "**Robust Learning From Demonstration Using Leveraged Gaussian Processes and Sparse Constrained Optimization**", in IEEE Conference on Robotics and Automation (ICRA), May 2016. **(Best Conference Paper Award Finalist)**
- Kyungjae Lee, Sungjoon Choi, and Songhwai Oh, "**Inverse Reinforcement Learning with Leveraged Gaussian Processes**," in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Oct. 2016.
- Sungjoon Choi, Eunwoo Kim, Kyungjae Lee, and Songhwai Oh, "**Leveraged Non-Stationary Gaussian Process Regression for Autonomous Robot Navigation**," in Proc. of the IEEE International Conference on Robotics and Automation (ICRA), May 2015.