Predicting Humans around Robots

Sanjiban Choudhury
The story thus far ...

- Decision-making
- Perception
- Models of humans
- Aligning robots to human values

Today-> Predicting humans around robots
Today’s class

- Why do we need prediction / forecasting?

- Forecasting as a Machine Learning problem
  - Model?
  - Loss?
  - Data?

- Connection between Forecasting and Model-based RL
Why do robots need to forecast humans?
Two motivating applications

Collaborative Cooking
Two motivating applications

Self-driving

Collaborative Cooking
What do these have in common?

**Forecasting human motion around robots**
Two motivating applications

Self-driving

Collaborative Cooking
Why do robots need to *forecast* humans?

To enable **safe**, **responsive**, and **interpretable** actions
Two motivating applications

Self-driving

Collaborative Cooking
Forecasting human motion is essential

No human prediction: Unresponsive robots are discomforting
Forecasting human motion is essential

No human forecast:
Unresponsive robots are discomforting

Human forecast:
Robot anticipates human and makes room
Forecasting human motion is essential
Why do robots need to forecast humans?

To enable safe, responsive, and interpretable actions
Today’s class

- Why do we need prediction / forecasting?
  (Enable safe, responsive, and interpretable robot actions)

- Forecasting as a Machine Learning problem
  - Model?
  - Loss?
  - Data?

- Connection between Forecasting and Model-based RL
Merging on the Highway
Think-Pair-Share
Learn forecasts for merging actors

Forecast 5s future trajectory

Once we have the forecast, we can plan to merge safely
Train a learner to forecast 5s future.

Model: Input / Output?

Data?

Loss?
Think-Pair-Share!

Think (30 sec): Train a learner to forecast 5s future.

Pair: Find a partner

Share (45 sec): Partners exchange ideas

Model: Input / Output?
Data?
Loss?
A first attempt at model, data, and loss
Model: Use a sequence model that maps past sequence (input) to future sequence (output)
Data: Drive around the car and collect data

Merge after

Merge before
Loss: L2 Loss from Ground Truth

Ground Truth: $s_{t+1}, s_{t+2}, \ldots, s_{t+k}$

Forecast: $\hat{s}_{t+1}, \hat{s}_{t+2}, \ldots, \hat{s}_{t+k}$

Loss: $\sum_{\tau=t}^{t+k} (s_\tau - \hat{s}_\tau)^2$
Loss: L2 Loss from Ground Truth

Ground Truth: $s_{t+1}, s_{t+2}, \ldots, s_{t+k}$

Suppose I am predicting both mean and variance.

Forecast: $(\hat{\mu}_{t+1}, \hat{\sigma}_{t+1}), (\hat{\mu}_{t+2}, \hat{\sigma}_{t+2}), \ldots, (\hat{\mu}_{t+k}, \hat{\sigma}_{t+k})$,

Loss: $\sum_{\tau=t}^{t+k} \frac{(s_{\tau} - \hat{\mu}_{\tau})^2}{\hat{\sigma}_{\tau}}$
Today’s class

- Why do we need prediction / forecasting?
  (Enable safe, responsive, and interpretable robot actions)
- Forecasting as a Machine Learning problem (First attempt)
  - Model?
  - Loss?
  - Data?
- Connection between Forecasting and Model-based RL
We have model, data, loss.

Let’s deploy the model!
Forecasts have huge variance!

Forces robot to brake aggressively!
Why is the forecast so whacky?
Why is the forecast so whacky?

There are two modes in the data

Mode A:
Robot merges after

Mode B:
Robot merges before
What happens when you try to fit a single Gaussian on multi-modal data?

Gaussian averages (marginalizes) over both modes
Okay .. so why can’t we just predict multi-modal distributions?
Multi-modal forecasts do not solve the issue

We are (incorrectly) telling the planner both modes can happen simultaneously
Forecast humans conditioned on what the robot will do
Solution: Train a conditional forecast

“If I slow down, what will happen?”

“If I speed up, what will happen?”
Solution: Train a conditional forecast

\[ \ldots, s_{t-2}, s_{t-1}, s_t, a_{t_1}, a_{t+2}, a_{t+3}, s_{t+1}, s_{t+2}, s_{t+3}, s_{t+4} \]
Today’s class

- Why do we need prediction / forecasting?  
  (Enable safe, responsive, and interpretable robot actions)

- Forecasting as a Machine Learning problem
  - Model?  (Conditional vs marginal forecasts)
  - Loss?
  - Data?

- Connection between Forecasting and Model-based RL
Two motivating applications

Self-driving

Collaborative Cooking
Are all time steps equally important in the loss?
We need accurate forecasts when humans come in close proximity.
How does forecasting error vary over time?

[Graph showing forecasting error over time with two lines, one for ManiCast and one for Base, with peaks and troughs over a 30-second period.]
How does forecasting error vary over time?

Error is low here.
But this is not a critical state as humans are far apart.
How does forecasting error vary over time?

Error shoots up here!
And it’s a very important state as humans in close proximity!
How does forecasting error vary over time?
Why is the error low here but higher here?
A simple fix:
Upweight critical transition points
Importance Sampling

Identify “transitions” when the human comes into the robot’s workspace

Task 1

Task 2

Task 3
Importance Sampling

Identify “transitions” when the human comes into the robot’s workspace
Train equally on all data + transition data

∀ \xi_R, \xi_H
Train equally on all data + transition data

∀ ξ_R, ξ_H

Sample Equally

∀ ξ_R, ξ_H

Transition Data

All Data
Generalization of the idea:

Forecasts should match the ground truth in terms of the cost it induces.
Solution: Replace L2 loss with cost weighted loss

\[
\forall \xi_R, \xi_H \quad \text{minimize} \quad \mathbb{E} \left[ \left| C(\xi_R, \xi_H) - C(\xi_R, \hat{\xi}_H) \right| \right]
\]

where, $\xi_H$ is the observed future human motion and, $\hat{\xi}_H$ is the predicted / forecasted human motion and, $\xi_R$ is the planned robot trajectory.
Evaluation across different tasks
Today’s class

- Why do we need prediction / forecasting?
  (Enable safe, responsive, and interpretable robot actions)

- Forecasting as a Machine Learning problem
  - Model? (Conditional vs marginal forecasts)
  - Loss? (Cost-weighted vs L2 loss)
  - Data?

- Connection between Forecasting and Model-based RL
Quiz
Refresher on Model-based RL

In model-based RL, what data distribution should we train transition models on?

When poll is active respond at  PollEv.com/sc2582

Send sc2582 to 22333
What happens when we deploy model?

**Robot:** “The car will probably merge ahead, so I can slow down very smoothly ...”

**Human:** “What the heck does this truck want to do, go ahead or behind ?!?!?”

“?!@#!@“

“?!@#!@“

“?!@#!@“
What went wrong?
What went wrong?

Robot: “The car will probably merge ahead, so I can slow down very smoothly …”

Humans never drive in such an ambiguous manner during merges!
We trained on data when human was driving
We trained on human driving data

We are testing on robot driving

If robot driving is different from human driving, we have a train-test mismatch
DAGGER for Forecasting!

Collect Data

Aggregate Data

Plan with forecasts

Train Forecaster
Today’s class

☑ Why do we need prediction / forecasting?
   (Enable safe, responsive, and interpretable robot actions)

☐ Forecasting as a Machine Learning problem
  ☑ Model? (Conditional vs marginal forecasts)
  ☑ Loss? (Cost-weighted vs L2 loss)
  ☑ Data? (Train on-policy on robot data)

☐ Connection between Forecasting and Model-based RL
Forecasts are really just transition models
Forecasting $\leftrightarrow$ Model-based RL

Conditional Forecasts

\[ P(s_{t:t+k} \mid s_{t:t-k}, a_{t:t+k}) \]

Model

\[ M(s_{t+1} \mid s_t, a_t) \]

We know how to solve model-based RL (previous lecture!)
Today's class

✔ Why do we need prediction / forecasting?
   (Enable safe, responsive, and interpretable robot actions)

☐ Forecasting as a Machine Learning problem
  ✔ Model?  (Conditional vs marginal forecasts)
  ✔ Loss?   (Cost-weighted vs L2 loss)
  ✔ Data?   (Train on-policy on robot data)

✔ Connection between Forecasting and Model-based RL