Spectral Algorithms for Latent Variable Models
Part 2: Dynamical Systems

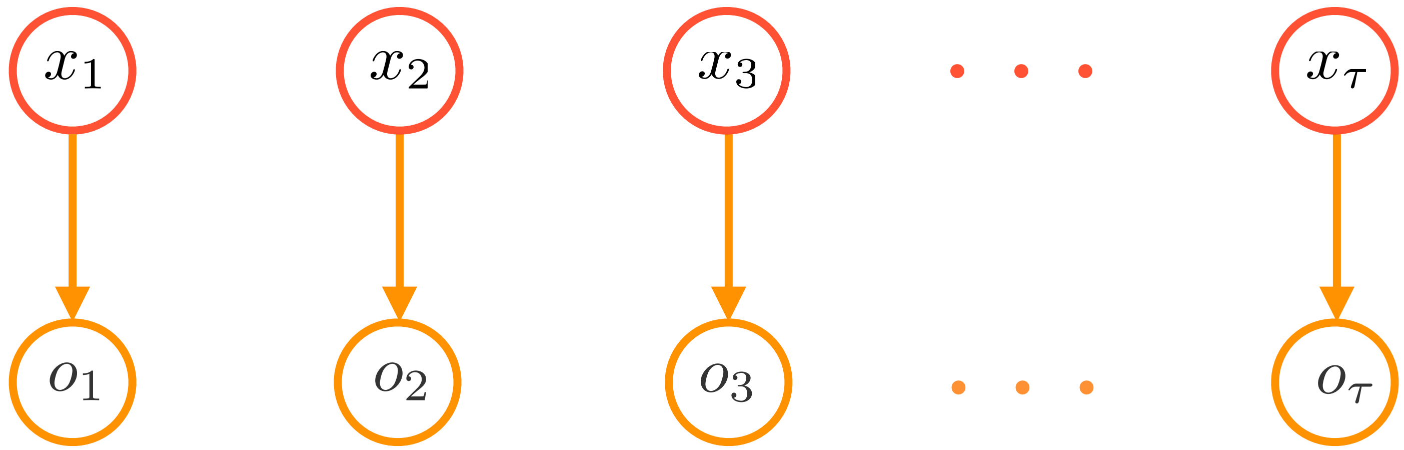
Byron Boots

http://www.cs.cmu.edu/~beb/
Machine Learning Department
Carnegie Mellon University

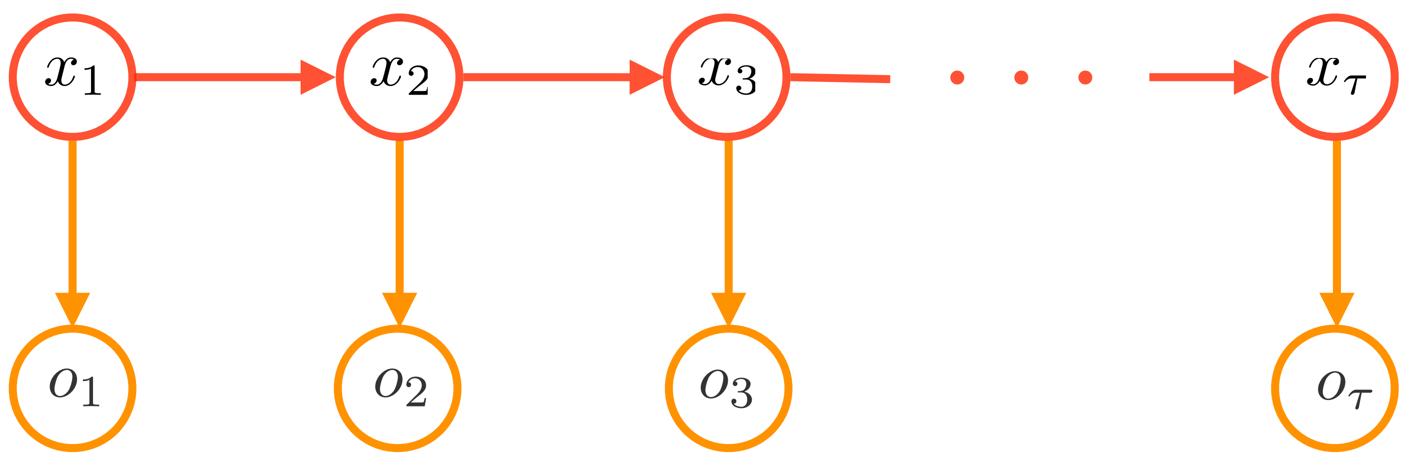
Sequences of Observations

$$o_1 \qquad o_2 \qquad o_3 \qquad \cdots \qquad o_T$$

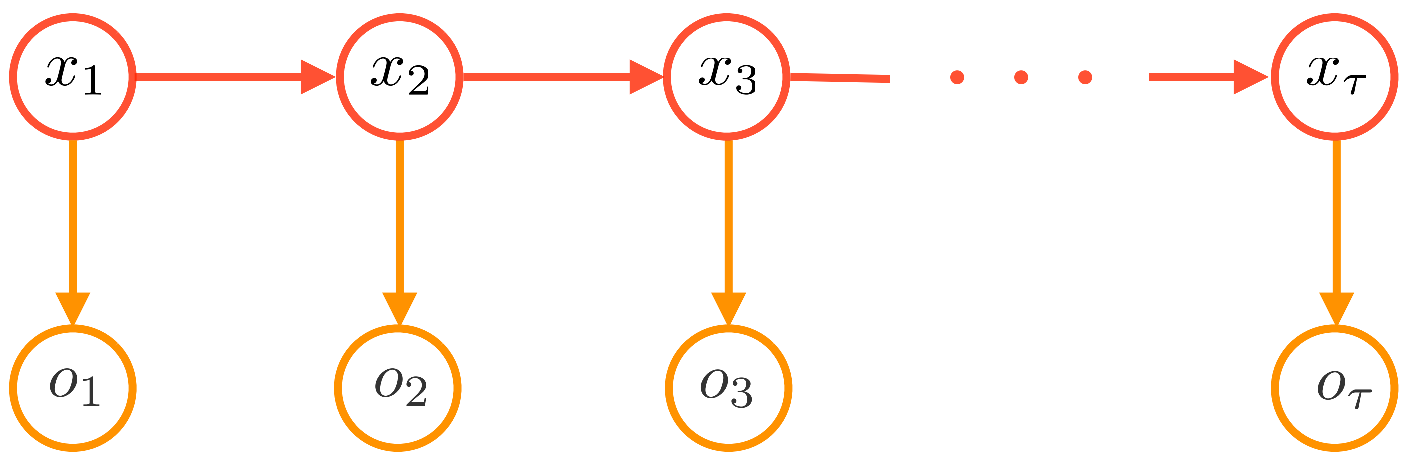
Latent State



Dynamics

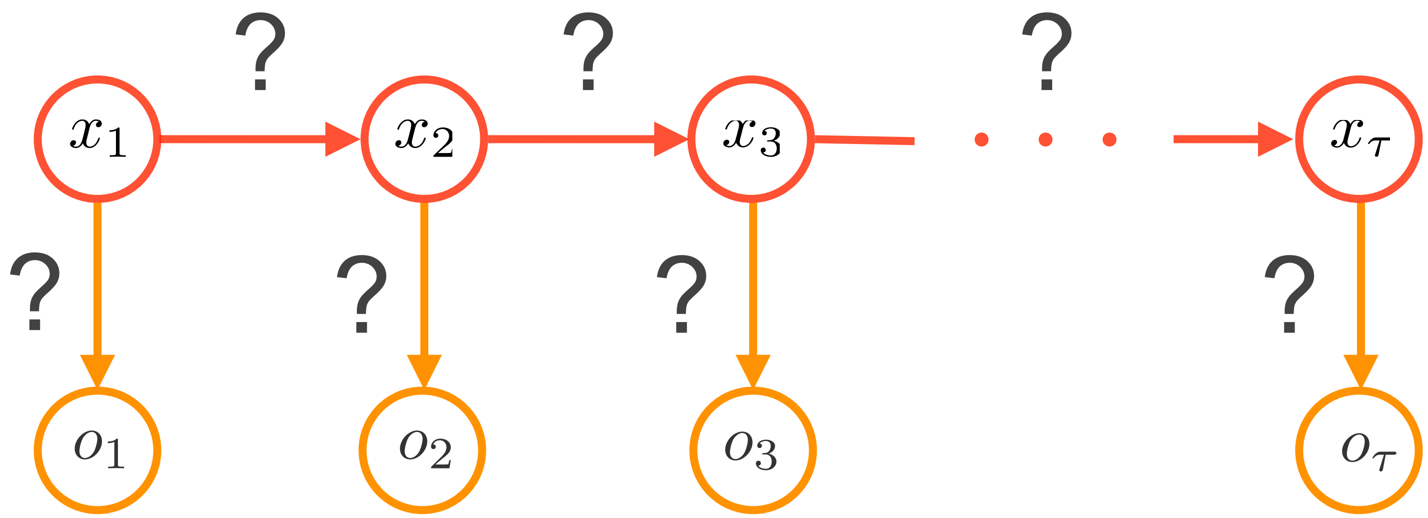


Dynamical Systems



- LTI Systems (Kalman Filter)
- (I-O) Hidden Markov Models
- Predictive State Representations

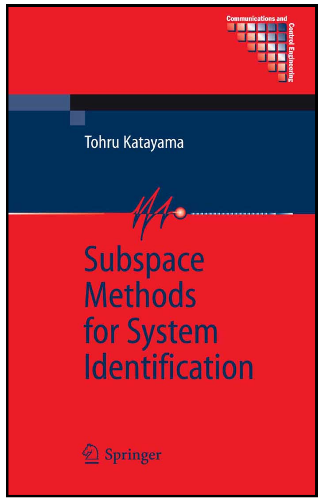
Learning a Dynamical System

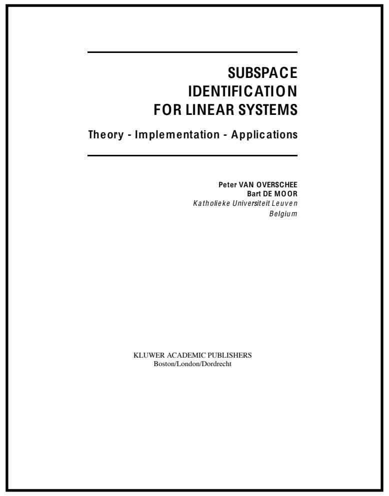


sense
learn
act

Spectral Learning for Dynamical Systems

- LTI Systems (Kalman Filter)





- Hidden Markov Models
  - Spectral learning of HMMs [Andersson, Ryden, 2008]
  - Spectral learning of HMMs [Hsu, Kakade, Zhang, 2009]
  - Spectral learning of RR-HMMs [Siddiqi, Boots, Gordon, 2009]
- Predictive State Representations
  - Spectral learning of PSRs [Boots, Siddiqi, Gordon, 2010]
  - Online spectral learning of PSRs [Boots, Gordon, 2011]

Why Spectral Methods?

There are many ways to learn a dynamical system

- Maximum Likelihood via Expectation Maximization, Gradient Descent, ...
- Bayesian inference via Gibbs, Metropolis Hastings, ...

Why Spectral Methods?

There are many ways to learn a dynamical system

- Maximum Likelihood via Expectation Maximization, Gradient Descent, ...
- Bayesian inference via Gibbs, Metropolis Hastings, ...

In contrast to these methods, spectral learning algorithms give

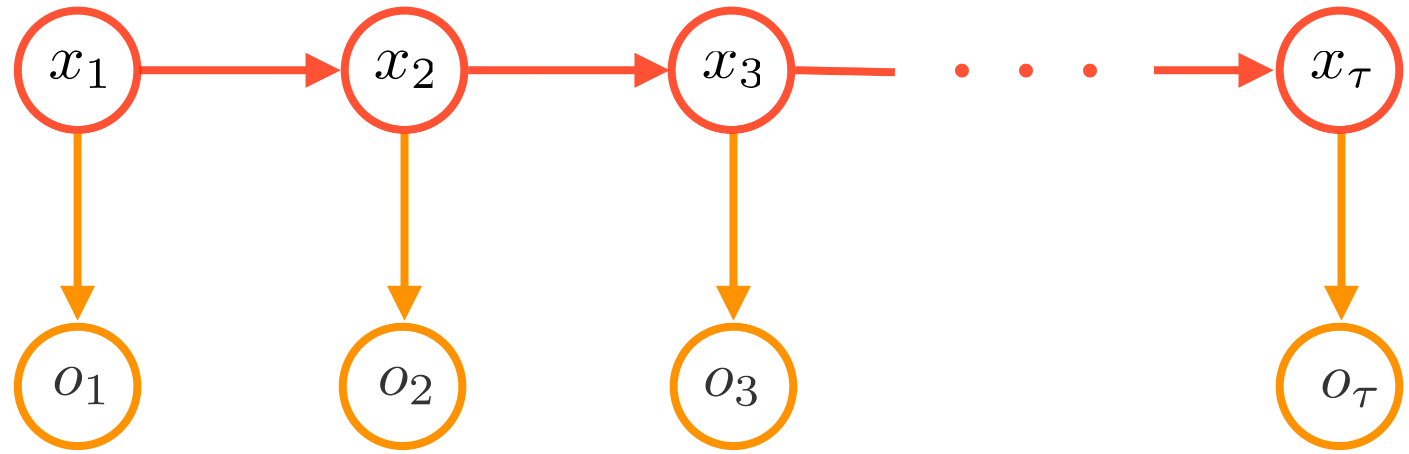
- No local optima:
  - Huge gain in computational efficiency

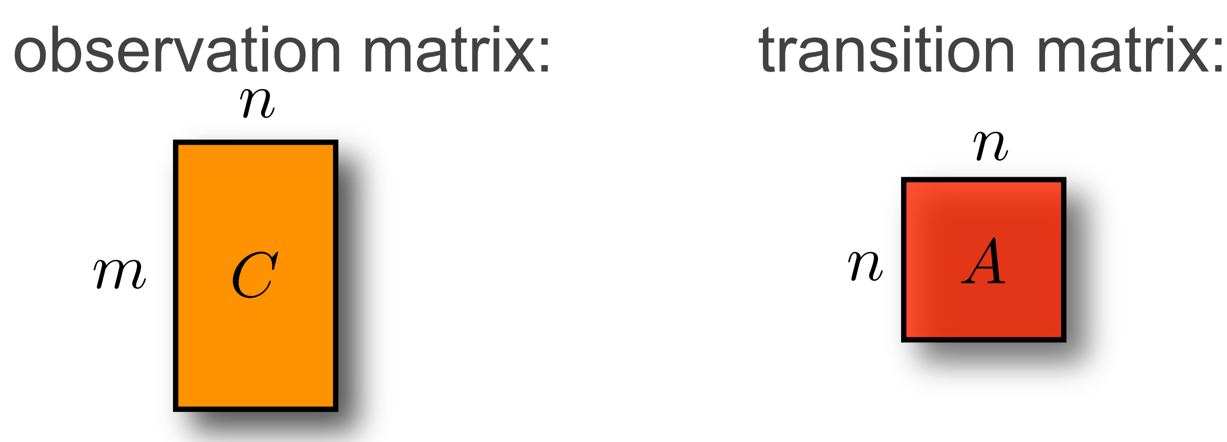
The focus of this part of the tutorial

- A spectral learning algorithm for Kalman filters
- A spectral learning algorithm for HMMs
- Relation to PSRs

Kalman Filters

$$x_{t+1} = Ax_t + noise$$
$$o_t = Cx_t + noise$$





- Assume for simplicity that $m \geq n$ and that A and C are full rank

We can relax both assumptions in practice

Kalman Filters

For $k \geq 1, \quad \Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right]$

$$\Sigma_k = C \; A^k \; P \; C^\top$$

- Assume for simplicity that $m \geq n$ and that A and C are full rank

Kalman Filters

For $k \geq 1, \quad \Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right] = \mathbb{E}\left[\mathbb{E}\left[o_{t+k} o_t^\top \mid x_t\right]\right]$

$$\Sigma_k = C \; A^k \; P \; C^\top$$

- Assume for simplicity that $m \geq n$ and that A and C are full rank

sense
learn
act

Kalman Filters

For $k \geq 1$, $\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right] = \mathbb{E}\left[\mathbb{E}\left[o_{t+k} o_t^\top \mid x_t\right]\right]$

$$= \mathbb{E}\left[\mathbb{E}\left[o_{t+k} \mid x_t\right]\mathbb{E}[o_t^\top \mid x_t]\right]$$

$$\Sigma_k = C \; A^k \; P \; C^\top$$

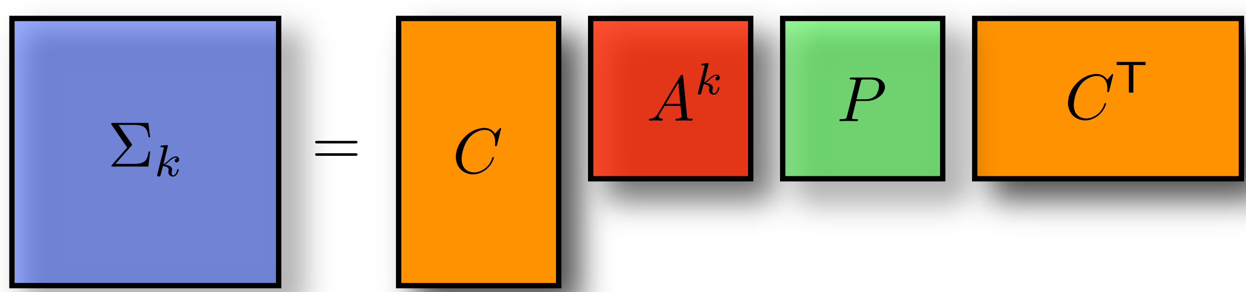
- Assume for simplicity that $m \geq n$ and that A and C are full rank

Kalman Filters

For $k \geq 1$, $\Sigma_k = \mathbb{E}\left[o_{t+k}o_t^\top\right] = \mathbb{E}\left[\mathbb{E}\left[o_{t+k}o_t^\top \mid x_t\right]\right]$

$$= \mathbb{E}\left[\mathbb{E}\left[o_{t+k} \mid x_t\right]\mathbb{E}[o_t^\top \mid x_t]\right]$$

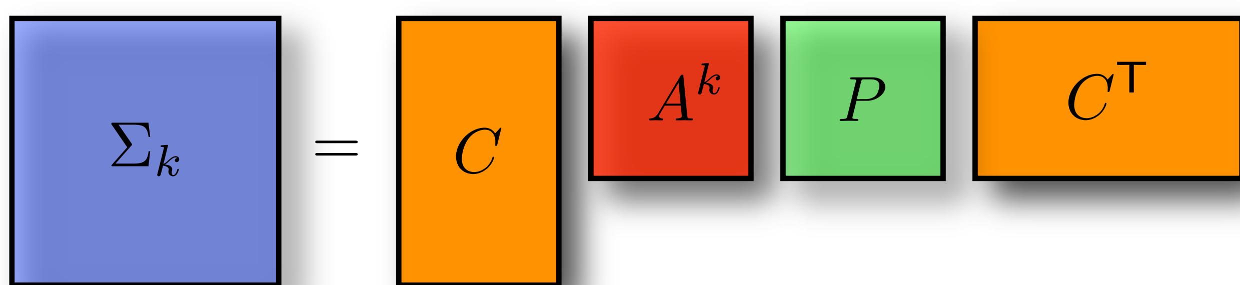
$$= \mathbb{E}\left[\left(CA^k x_t\right)\left(Cx_t\right)^\top\right]$$



- Assume for simplicity that $m \geq n$ and that A and C are full rank

Kalman Filters

For $k \geq 1$, $$\begin{aligned}\Sigma_k = \mathbb{E}\left[o_{t+k}o_t^\top\right] &= \mathbb{E}\left[\mathbb{E}\left[o_{t+k}o_t^\top \mid x_t\right]\right] \\ &= \mathbb{E}\left[\mathbb{E}\left[o_{t+k} \mid x_t\right]\mathbb{E}[o_t^\top \mid x_t]\right] \\ &= \mathbb{E}\left[\left(CA^k x_t\right)\left(Cx_t\right)^\top\right] \\ &= CA^k\mathbb{E}[x_t x_t^\top]C^\top\end{aligned}$$



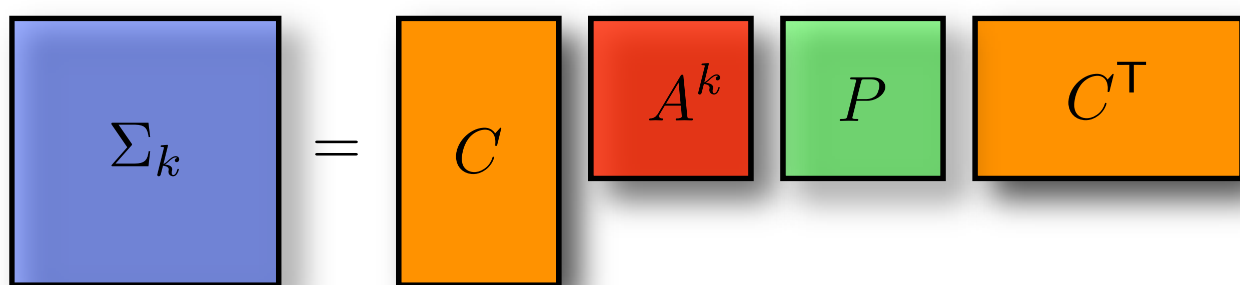
- Assume for simplicity that $m \geq n$ and that A and C are full rank

sense
learn
act

Kalman Filters

For $k \geq 1$,

$$\Sigma_k = \mathbb{E}\left[o_{t+k}o_t^\top\right] = \mathbb{E}\left[\mathbb{E}\left[o_{t+k}o_t^\top \mid x_t\right]\right]$$
$$= \mathbb{E}\left[\mathbb{E}\left[o_{t+k} \mid x_t\right]\mathbb{E}[o_t^\top \mid x_t]\right]$$
$$= \mathbb{E}\left[\left(CA^k x_t\right)\left(Cx_t\right)^\top\right]$$
$$= CA^k\mathbb{E}[x_t x_t^\top]C^\top$$
$$= CA^k P C^\top$$



- Assume for simplicity that $m \geq n$ and that A and C are full rank

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right] = CA^k PC^\top$$

Let U be the left n singular vectors of Σ_1,

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\mathsf{T}\right] = C A^k P C^\mathsf{T}$$

Let U be the left n singular vectors of Σ_1,

$$\widehat{A} \quad := \quad U^\top \Sigma_2 (U^\top \Sigma_1)^\dagger$$

sense
learn
act

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right] = CA^k PC^\top$$

Let U be the left n singular vectors of Σ_1,

$$\begin{aligned} \widehat{A} &:= U^\top \Sigma_2 (U^\top \Sigma_1)^\dagger \\ &= U^\top C A^2 P C^\top (U^\top C A P C^\top)^\dagger \end{aligned}$$

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right] = CA^k P C^\top$$

Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\widehat{A} \quad &:= \quad U^\top \Sigma_2 (U^\top \Sigma_1)^\dagger \\
&= \quad U^\top C A^2 P C^\top (U^\top C A P C^\top)^\dagger \\
&= \quad U^\top C A^2 (U^\top C A)^{-1} (U^\top C A) P C^\top (U^\top C A P C^\top)^\dagger
\end{aligned}$$

sense
learn
act

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^{\top}\right] = CA^k PC^{\top}$$

Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\widehat{A} &:= U^\top \Sigma_2 (U^\top \Sigma_1)^\dagger \\
&= U^\top C A^2 P C^\top (U^\top C A P C^\top)^\dagger \\
&= U^\top C A^2 (U^\top C A)^{-1} (U^\top C A) P C^\top (U^\top C A P C^\top)^\dagger \\
&= (U^\top C A) A (U^\top C A)^{-1}
\end{aligned}$$

sense
learn
act

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right] = CA^k PC^\top$$

Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\widehat{A} &:= U^\top \Sigma_2 (U^\top \Sigma_1)^\dagger \\
&= U^\top CA^2PC^\top (U^\top CAPC^\top)^\dagger \\
&= U^\top CA^2 (U^\top CA)^{-1} (U^\top CA) PC^\top (U^\top CAPC^\top)^\dagger \\
&= (U^\top CA) A (U^\top CA)^{-1} \\
&= \underline{SAS^{-1}} \quad \text{similarity transform of } A
\end{aligned}$$

sense learn act

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right] = CA^k PC^\top$$

Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\widehat{A} &:= U^\top \Sigma_2 (U^\top \Sigma_1)^\dagger \\
&= U^\top CA^2PC^\top (U^\top CAPC^\top)^\dagger \\
&= U^\top CA^2 (U^\top CA)^{-1} (U^\top CA) PC^\top (U^\top CAPC^\top)^\dagger \\
&= (U^\top CA) A (U^\top CA)^{-1} \\
&= \underline{SAS^{-1}} \quad \text{similarity transform of } A
\end{aligned}$$

$$\widehat{C} := U\widehat{A}^{-1}$$

Tuesday, June 26, 2012

sense learn act

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right] = C A^k P C^\top$$

Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\widehat{A} &:= U^\top \Sigma_2 (U^\top \Sigma_1)^\dagger \\
&= U^\top C A^2 P C^\top (U^\top C A P C^\top)^\dagger \\
&= U^\top C A^2 (U^\top C A)^{-1} (U^\top C A) P C^\top (U^\top C A P C^\top)^\dagger \\
&= (U^\top C A) A (U^\top C A)^{-1} \\
&= \underline{S A S^{-1}} \quad \text{similarity transform of } A
\end{aligned}$$

$$\begin{aligned}
\widehat{C} &:= U \widehat{A}^{-1} \\
&= U S A^{-1} S^{-1}
\end{aligned}$$

sense
learn
act

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^\top\right] = CA^k PC^\top$$

Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\widehat{A} &:= U^\top \Sigma_2 (U^\top \Sigma_1)^\dagger \\
&= U^\top CA^2 PC^\top (U^\top CAPC^\top)^\dagger \\
&= U^\top CA^2 (U^\top CA)^{-1} (U^\top CA) PC^\top (U^\top CAPC^\top)^\dagger \\
&= (U^\top CA) A (U^\top CA)^{-1} \\
&= \underline{SAS^{-1}} \quad \text{similarity transform of } A
\end{aligned}$$

$$\begin{aligned}
\widehat{C} &:= U\widehat{A}^{-1} \\
&= USA^{-1}S^{-1} \\
&= U(U^\top CA)A^{-1}S^{-1}
\end{aligned}$$

sense learn act

Kalman Filters

$$\Sigma_k = \mathbb{E}\left[o_{t+k} o_t^{\mathsf{T}}\right] = CA^k PC^{\mathsf{T}}$$

Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\widehat{A} &:= U^\top \Sigma_2 (U^\top \Sigma_1)^\dagger \\
&= U^\top CA^2 PC^\top (U^\top CAPC^\top)^\dagger \\
&= U^\top CA^2 (U^\top CA)^{-1} (U^\top CA) PC^\top (U^\top CAPC^\top)^\dagger \\
&= (U^\top CA) A (U^\top CA)^{-1} \\
&= \underline{SAS^{-1}} \quad \text{similarity transform of } A
\end{aligned}$$

$$\begin{aligned}
\widehat{C} &:= U \widehat{A}^{-1} \\
&= USA^{-1}S^{-1} \\
&= U(U^\top CA) A^{-1} S^{-1} \\
&= \underline{CS^{-1}} \quad \text{linear transform of } C
\end{aligned}$$

sense
learn
act

Kalman Filters

Spectral Learning Algorithm:

- Estimate Σ_1 and Σ_2 from data
- Find $\widehat{U}$ by SVD
- Plug in for $\widehat{A}$ and $\widehat{C}$

Learning is Consistent:

- Law of Large numbers for Σ_1 and Σ_2
- Continuity of formulas for $\widehat{A}$ and $\widehat{C}$

sense
learn
act

Variations on Spectral Learning for Kalman Filters

- Use arbitrary features of past and future observations
  - work from covariance of past, future features
  - good features make a big difference in practice

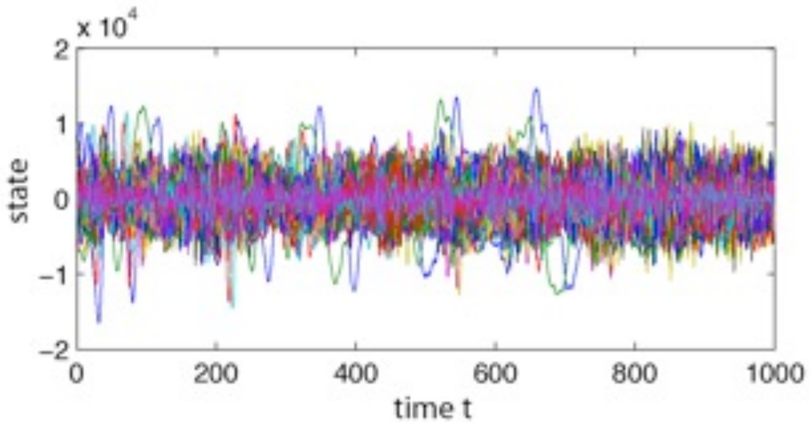
- Use different spectral decompositions to find state space: CCA, RRR
- Impose constraints on learned model (e.g., stability)
- Learn Kalman filters with control inputs

sense
learn
act

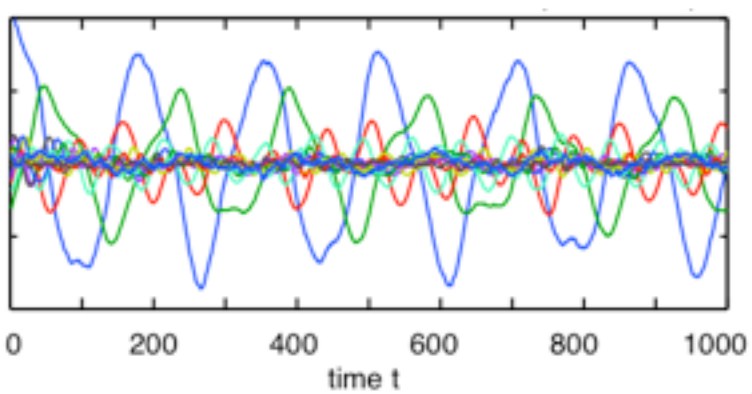
Example: Video Textures

works well for learning models of video textures
observations = raw pixels (vector of reals over time)

simulations from learned models
[*Siddiqi, Boots, Gordon, 2007*]



(40 dimensions)



(40 dimensions)

sense
learn
act

Additional Examples

- Glass oven modeling [Backx, 1987]
- Aircraft wing flutter [Peloubet et al., 1990]
- Control of air temperature and flow [Ljung, 1991]
- Mechanical construction of CD player arms [Van Den Hof et al., 1993]
- Heat flow through walls [Bloem, 1994]
- Chemical processes [Van Overschee, De Moor, 1996]
- Economic forecasting [Aoki, Havenner, 1997]
- ...

sense
learn
act

Kalman Filter Spectral Learning: Failure

given a short video

Learn a model

Kalman Filter Spectral Learning: Failure

given a short video

Learn a model

sense learn act

Kalman Filter Spectral Learning: Failure

Simulations from models trained on clock data

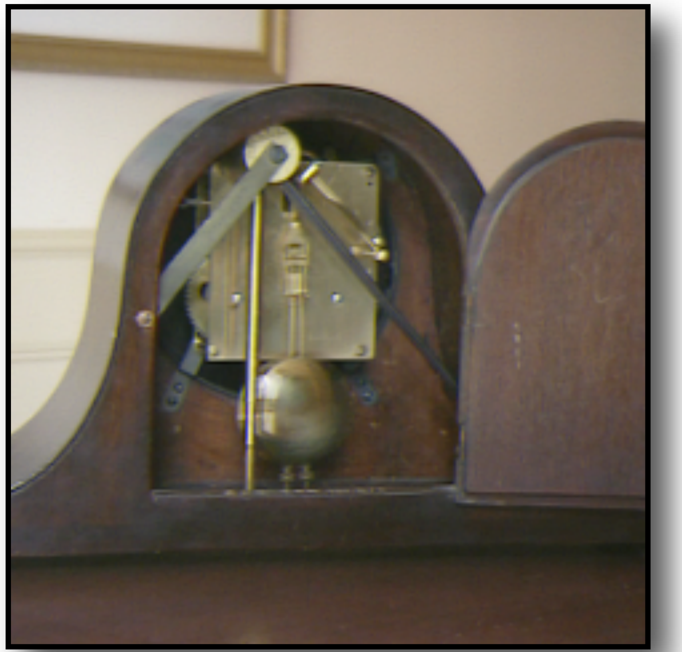
?

Kalman Filter (spectral)	HMM (Baum-Welch)	Something better...
10 dimensions	10 states	10 dimensions

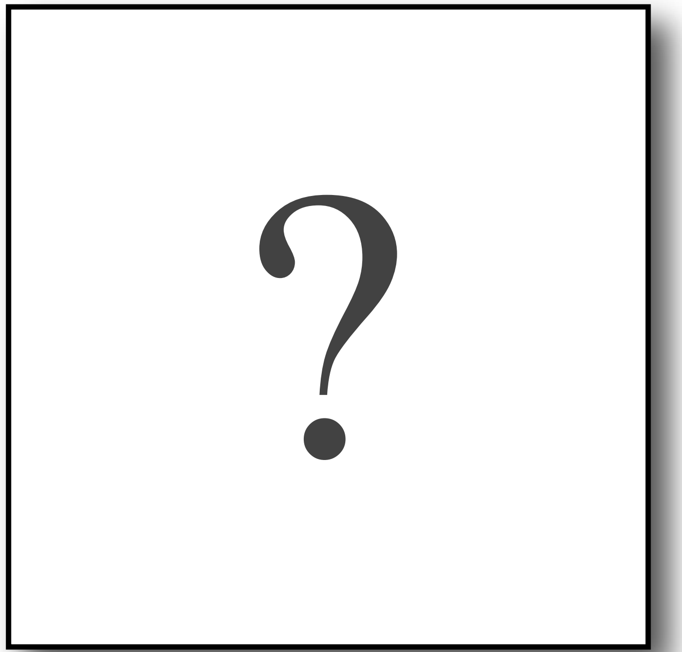
Kalman Filter Spectral Learning: Failure

Simulations from models trained on clock data

Kalman Filter (spectral)
10 dimensions



HMM (Baum-Welch)
10 states



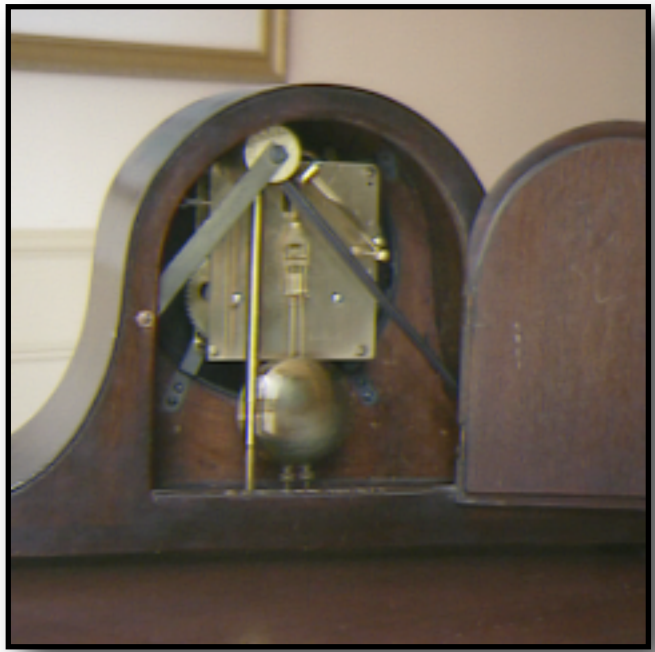
Something better...
10 dimensions

sense learn act

Kalman Filter Spectral Learning: Failure

Simulations from models trained on clock data

Kalman Filter (spectral)
10 dimensions



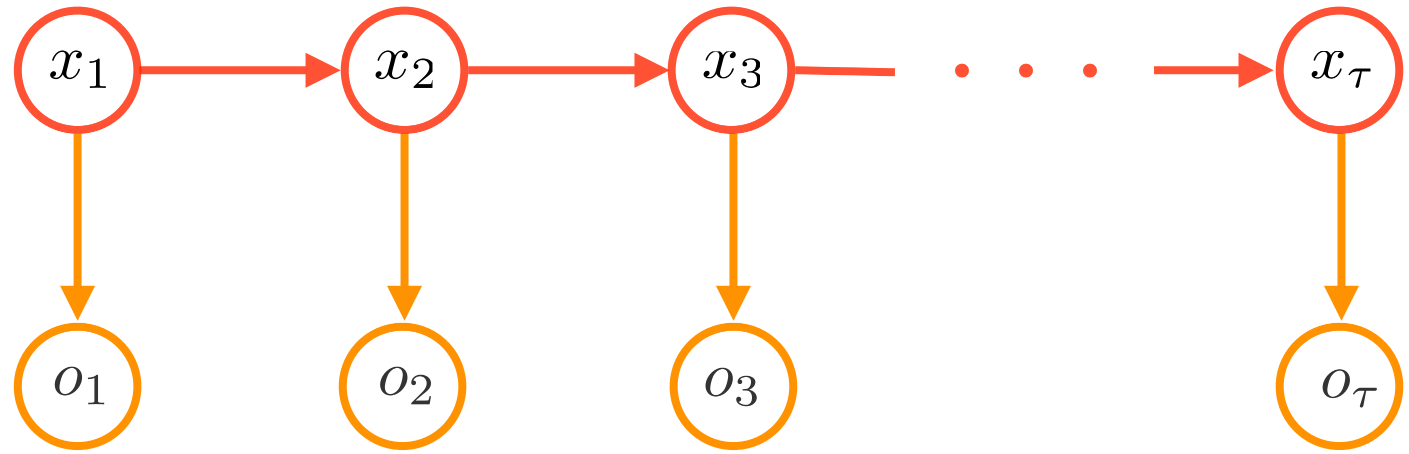
HMM (Baum-Welch)
10 states

?

Something better...
10 dimensions

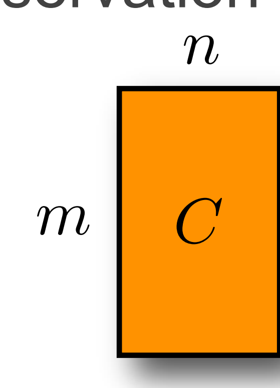
Can We Generalize Spectral Learning?
HMMs

$$x_{t+1} = Ax_t + noise$$
$$o_t = Cx_t + noise$$

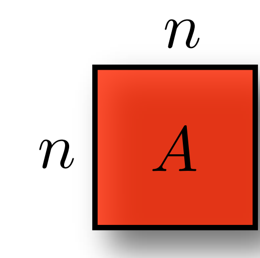


- Get rid of Gaussian noise assumption
- Hidden Markov Model: same form as Kalman Filter but,
  - $A \geq 0,\ A\mathbf{1} = \mathbf{1},\ \mathbf{C} \geq \mathbf{0},\ \mathbf{C1} = \mathbf{1}$
  - noise ~ Multinomial Distribution
  - x and o are indicators: e.g. "4" $= [0\,0\,0\,1\,0]^{\mathsf{T}}$

observation matrix:



transition matrix:



sense learn act

Spectral Learning: Gaussian vs. Multinomial

Kalman Filter	Hidden Markov Model

$$\begin{aligned}
\mathbb{E}\left[o_{t+k}o_t^\top\right] &= \mathbb{E}\left[\mathbb{E}\left[o_{t+k}o_t^\top \mid x_t\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[o_{t+k} \mid x_t\right]\mathbb{E}[o_t^\top \mid x_t]\right] \\
&= \mathbb{E}\left[\left(CA^k x_t\right)\left(Cx_t\right)^\top\right] \\
&= CA^k\mathbb{E}[x_t x_t^\top]C^\top \\
&= CA^kPC^\top
\end{aligned}$$

- Assume for simplicity that $m \geq n$ and that A and C are full rank

Spectral Learning: Gaussian vs. Multinomial

Kalman Filter

$$
\begin{aligned}
\mathbb{E}\left[o_{t+k}o_t^{\mathsf{T}}\right] &= \mathbb{E}\left[\mathbb{E}\left[o_{t+k}o_t^{\mathsf{T}} \mid x_t\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[o_{t+k} \mid x_t\right]\mathbb{E}[o_t^{\mathsf{T}} \mid x_t]\right] \\
&= \mathbb{E}\left[\left(CA^k x_t\right)\left(Cx_t\right)^{\mathsf{T}}\right] \\
&= CA^k\mathbb{E}[x_t x_t^{\mathsf{T}}]C^{\mathsf{T}} \\
&= CA^kPC^{\mathsf{T}}
\end{aligned}
$$

Hidden Markov Model

$$
\begin{aligned}
\mathbb{E}\left[o_{t+k}o_t^{\mathsf{T}}\right] &= \mathbb{E}\left[\mathbb{E}\left[o_{t+k}o_t^{\mathsf{T}} \mid x_t\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[o_{t+k} \mid x_t\right]\mathbb{E}[o_t^{\mathsf{T}} \mid x_t]\right] \\
&= \mathbb{E}\left[\left(CA^k x_t\right)\left(Cx_t\right)^{\mathsf{T}}\right] \\
&= CA^k\mathbb{E}[x_t x_t^{\mathsf{T}}]C^{\mathsf{T}} \\
&= CA^kPC^{\mathsf{T}}
\end{aligned}
$$

exactly the same!

- Assume for simplicity that $m \geq n$ and that A and C are full rank

Spectral Learning for HMMs

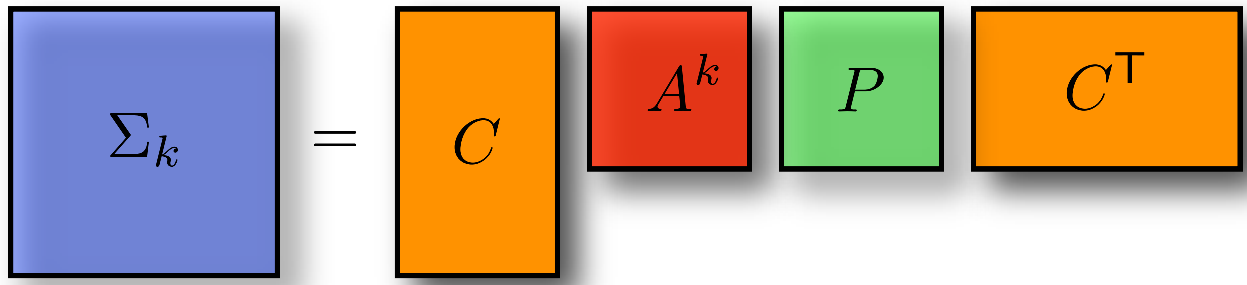
$$\Sigma_k = C \; A^k \; P \; C^\top$$

$$\begin{aligned} \widehat{A} &:= U^\top \Sigma_2 \left(U^\top \Sigma_1\right)^\dagger \\ &= (U^\top C A) A (U^\top C A)^{-1} \\ &= S A S^{-1} \end{aligned}$$

- As before, recover $\widehat{A}$ and $\widehat{C}$ from Σ_1 and Σ_2
- Does not satisfy $A \geq 0,\; A\mathbf{1} = \mathbf{1},\; \mathbf{C} \geq \mathbf{0},\; \mathbf{C1} = \mathbf{1}$
    - is this a problem?

sense learn act

Spectral Learning for HMMs



$$\begin{aligned}
\widehat{A} \quad &:= \quad U^\top \Sigma_2 \left(U^\top \Sigma_1\right)^\dagger \\
&= \quad (U^\top C A) A (U^\top C A)^{-1} \\
&= \quad S A S^{-1}
\end{aligned}$$

- As before, recover $\widehat{A}$ and $\widehat{C}$ from Σ_1 and Σ_2
- Does not satisfy $A \geq 0,\ A\mathbf{1} = \mathbf{1},\ \mathbf{C} \geq \mathbf{0},\ \mathbf{C1} = \mathbf{1}$
  - is this a problem?

Yes. Inference is different in an HMM.

Tuesday, June 26, 2012

sense
learn
act

Inference for HMMs

$$\mathbb{P}\left[o_1, o_2, \ldots, o_\tau\right]$$

Inference for HMMs

$$\mathbb{P}\left[o_1, o_2, \ldots, o_\tau\right]$$

$$\sum_{x_{\tau+1}} \sum_{x_\tau} \mathbb{P}\left[x_{\tau+1} \mid x_\tau\right] \mathbb{P}\left[o_\tau \mid x_\tau\right] \ldots \sum_{x_2} \mathbb{P}\left[x_3 \mid x_2\right] \mathbb{P}\left[o_2 \mid x_2\right] \sum_{x_1} \mathbb{P}\left[x_2 \mid x_1\right] \mathbb{P}\left[o_1 \mid x_1\right] \mathbb{P}\left[x_1\right]$$

factor by chain rule
marginalizing out latent state

sense
learn
act

Inference for HMMs

$$\mathbb{P}\left[o_1, o_2, \ldots, o_\tau\right]$$

$$\sum_{x_{\tau+1}} \sum_{x_\tau} \mathbb{P}\left[x_{\tau+1} \mid x_\tau\right] \mathbb{P}\left[o_\tau \mid x_\tau\right] \ldots \sum_{x_2} \mathbb{P}\left[x_3 \mid x_2\right] \mathbb{P}\left[o_2 \mid x_2\right] \sum_{x_1} \mathbb{P}\left[x_2 \mid x_1\right] \mathbb{P}\left[o_1 \mid x_1\right] \mathbb{P}\left[x_1\right]$$

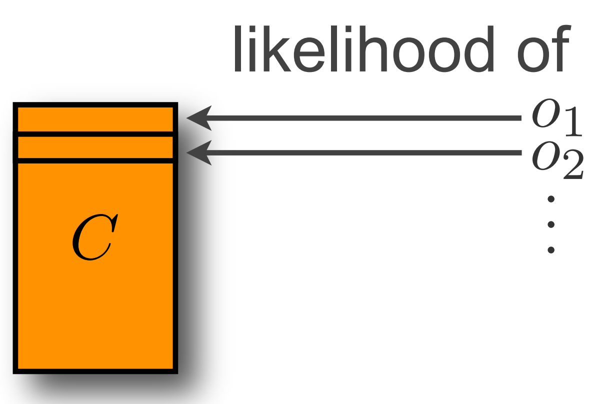
transition probability   observation likelihood

Inference for HMMs

$$\mathbb{P}\left[o_1, o_2, \ldots, o_\tau\right]$$

$$\sum_{x_{\tau+1}} \sum_{x_\tau} \mathbb{P}\left[x_{\tau+1} \mid x_\tau\right] \mathbb{P}\left[o_\tau \mid x_\tau\right] \ldots \sum_{x_2} \mathbb{P}\left[x_3 \mid x_2\right] \mathbb{P}\left[o_2 \mid x_2\right] \sum_{x_1} \mathbb{P}\left[x_2 \mid x_1\right] \mathbb{P}\left[o_1 \mid x_1\right] \mathbb{P}\left[x_1\right]$$

$$\mathbf{1}_n^{\top} A \operatorname{diag}(C_{o_\tau,:}) \ldots A \operatorname{diag}(C_{o_2,:}) A \operatorname{diag}(C_{o_1,:}) \mathbb{P}\left[x_1\right]$$



sense learn act

Inference for HMMs

$$\mathbb{P}\left[o_1, o_2, \ldots, o_\tau\right]$$

$$\sum_{x_{\tau+1}} \sum_{x_\tau} \mathbb{P}\left[x_{\tau+1} \mid x_\tau\right] \mathbb{P}\left[o_\tau \mid x_\tau\right] \ldots \sum_{x_2} \mathbb{P}\left[x_3 \mid x_2\right] \mathbb{P}\left[o_2 \mid x_2\right] \sum_{x_1} \mathbb{P}\left[x_2 \mid x_1\right] \mathbb{P}\left[o_1 \mid x_1\right] \mathbb{P}\left[x_1\right]$$

$$\mathbf{1}_n^\top A \operatorname{diag}(C_{o_{\tau},:}) \ldots A \operatorname{diag}(C_{o_2,:}) A \operatorname{diag}(C_{o_1,:}) \mathbb{P}\left[x_1\right]$$

=

$$\mathbf{1}_n^\top SAS^{-1}\, S\operatorname{diag}(C_{o_{\tau},:})S^{-1} \ldots SAS^{-1}\, S\operatorname{diag}(C_{o_2,:})S^{-1}\, SAS^{-1}\, S\operatorname{diag}(C_{o_1,:})S^{-1} S\mathbb{P}[x_1]$$

sense learn act

Inference for HMMs

$$\mathbb{P}\left[o_1, o_2, \ldots, o_\tau\right]$$

$$\sum_{x_{\tau+1}} \sum_{x_\tau} \mathbb{P}\left[x_{\tau+1} \mid x_\tau\right] \mathbb{P}\left[o_\tau \mid x_\tau\right] \ldots \sum_{x_2} \mathbb{P}\left[x_3 \mid x_2\right] \mathbb{P}\left[o_2 \mid x_2\right] \sum_{x_1} \mathbb{P}\left[x_2 \mid x_1\right] \mathbb{P}\left[o_1 \mid x_1\right] \mathbb{P}\left[x_1\right]$$

$$\mathbf{1}_n^\top A \operatorname{diag}(C_{o_{\tau},:}) \ldots A \operatorname{diag}(C_{o_{2},:}) A \operatorname{diag}(C_{o_{1},:}) \mathbb{P}\left[x_1\right]$$

$$\mathbf{1}_n^\top SAS^{-1}\, S\operatorname{diag}(C_{o_{\tau},:})S^{-1} \ldots SAS^{-1}\, S\operatorname{diag}(C_{o_{2},:})S^{-1}\, SAS^{-1}\, S\operatorname{diag}(C_{o_{1},:})S^{-1} S\mathbb{P}[x_1]$$

We have access to:

$$\widehat{A} = SAS^{-1} \quad \widehat{C} = CS^{-1}$$

but

No good way of finding the observation likelihoods

(e.g. $S\operatorname{diag}(C_{o_{1},:})S^{-1}$)

Inference for HMMs

$$\mathbb{P}\left[o_1, o_2, \ldots, o_\tau\right]$$

$$\sum_{x_{\tau+1}} \sum_{x_\tau} \mathbb{P}\left[x_{\tau+1} \mid x_\tau\right] \mathbb{P}\left[o_\tau \mid x_\tau\right] \ldots \sum_{x_2} \mathbb{P}\left[x_3 \mid x_2\right] \mathbb{P}\left[o_2 \mid x_2\right] \sum_{x_1} \mathbb{P}\left[x_2 \mid x_1\right] \mathbb{P}\left[o_1 \mid x_1\right] \mathbb{P}\left[x_1\right]$$

$$\mathbf{1}_n^{\top} A \operatorname{diag}(C_{o_\tau,:}) \ldots A \operatorname{diag}(C_{o_2,:}) A \operatorname{diag}(C_{o_1,:}) \mathbb{P}\left[x_1\right]$$

Tuesday, June 26, 2012

sense
learn
act

Inference for HMMs

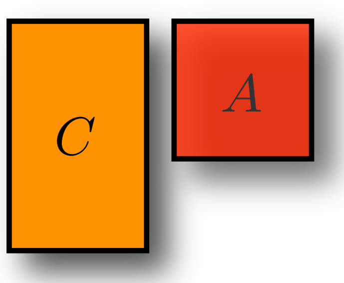
$$\mathbb{P}[o_1, o_2, \dots, o_\tau]$$

$$\sum_{x_{\tau+1}} \sum_{x_\tau} \mathbb{P}[x_{\tau+1} \mid x_\tau]\,\mathbb{P}[o_\tau \mid x_\tau] \dots \sum_{x_2} \mathbb{P}[x_3 \mid x_2]\,\mathbb{P}[o_2 \mid x_2] \sum_{x_1} \mathbb{P}[x_2 \mid x_1]\,\mathbb{P}[o_1 \mid x_1]\,\mathbb{P}[x_1]$$

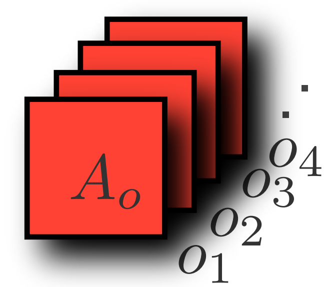
$$\mathbf{1}_n^\top \underbrace{A\,\mathrm{diag}(C_{o_\tau,:})} \dots \underbrace{A\,\mathrm{diag}(C_{o_2,:})}\,\underbrace{A\,\mathrm{diag}(C_{o_1,:})}\,\mathbb{P}[x_1]$$

combine into a single observable operator, one for each observation

standard HMM parameterization



observable operator HMM parameterization [Jaeger, 1998]



Tuesday, June 26, 2012

Inference for HMMs

$$\mathbb{P}\left[o_1, o_2, \ldots, o_\tau\right]$$

$$\sum_{x_{\tau+1}} \sum_{x_\tau} \mathbb{P}\left[x_{\tau+1} \mid x_\tau\right] \mathbb{P}\left[o_\tau \mid x_\tau\right] \ldots \sum_{x_2} \mathbb{P}\left[x_3 \mid x_2\right] \mathbb{P}\left[o_2 \mid x_2\right] \sum_{x_1} \mathbb{P}\left[x_2 \mid x_1\right] \mathbb{P}\left[o_1 \mid x_1\right] \mathbb{P}\left[x_1\right]$$

$$\mathbf{1}_n^\top A \operatorname{diag}(C_{o_\tau,:}) \ldots A \operatorname{diag}(C_{o_2,:}) A \operatorname{diag}(C_{o_1,:}) \mathbb{P}\left[x_1\right]$$

$$\mathbf{1}_n^\top A_{o_\tau} \ldots A_{o_2} A_{o_1} \mathbb{P}\left[x_1\right]$$

sense
learn
act

Inference for HMMs

$$\mathbb{P}\left[o_1, o_2, \ldots, o_\tau\right]$$

$$\sum_{x_{\tau+1}} \sum_{x_\tau} \mathbb{P}\left[x_{\tau+1} \mid x_\tau\right] \mathbb{P}\left[o_\tau \mid x_\tau\right] \ldots \sum_{x_2} \mathbb{P}\left[x_3 \mid x_2\right] \mathbb{P}\left[o_2 \mid x_2\right] \sum_{x_1} \mathbb{P}\left[x_2 \mid x_1\right] \mathbb{P}\left[o_1 \mid x_1\right] \mathbb{P}\left[x_1\right]$$

$$\mathbf{1}_n^\top A \operatorname{diag}(C_{o_\tau,:}) \ldots A \operatorname{diag}(C_{o_2,:}) A \operatorname{diag}(C_{o_1,:}) \mathbb{P}\left[x_1\right]$$

$$\mathbf{1}_n^\top A_{o_\tau} \ldots A_{o_2} A_{o_1} \mathbb{P}\left[x_1\right]$$

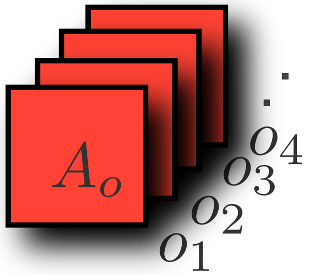
$$\mathbf{1} S^{-1} S A_{o_\tau} S^{-1} \ldots S A_{o_2} S^{-1} S A_{o_1} S^{-1} S \mathbb{P}[x_1]$$

In fact, only need to estimate similarity transforms of parameters $S A_o S^{-1}$

the Ss cancel

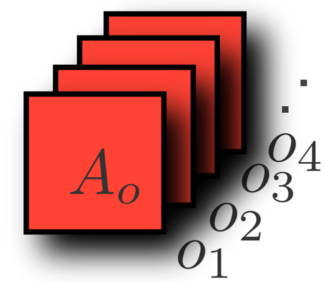
Spectral Learning for HMMs

Goal is to find similarity transforms of A_os

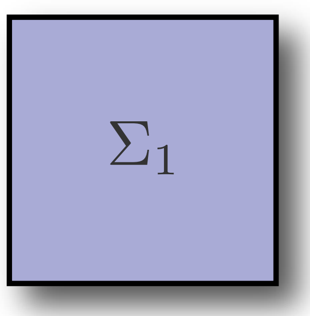


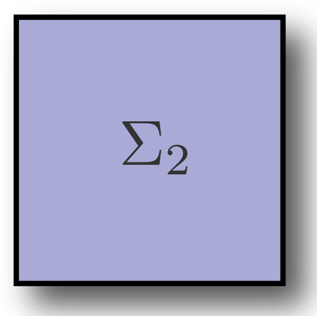
Spectral Learning for HMMs

Goal is to find similarity transforms of A_os



$$\Sigma_1 := \mathbb{E}\left[o_{t+1} o_t^\top\right] = CAPC^\top \qquad \Sigma_2 := \mathbb{E}\left[o_{t+2} o_t^\top\right] = CA^2PC^\top$$

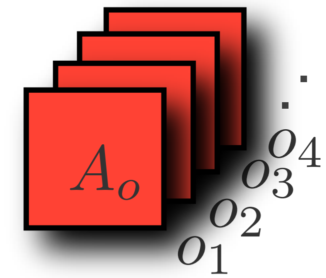




sense
learn
act

Spectral Learning for HMMs

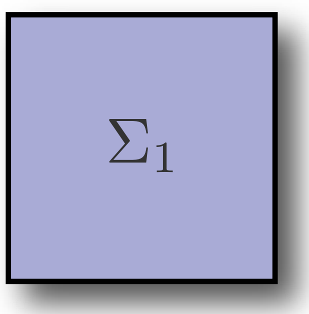
Goal is to find similarity transforms of A_os

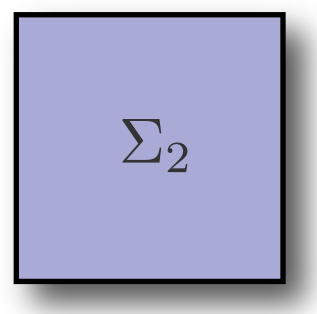


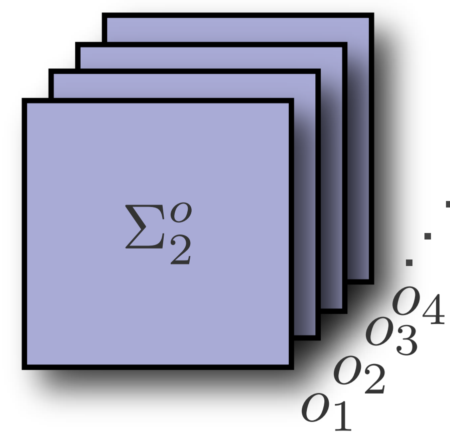
$$\Sigma_1 := \mathbb{E}\left[o_{t+1} o_t^\top\right] = CAPC^\top$$

$$\Sigma_2 := \mathbb{E}\left[o_{t+2} o_t^\top\right] = CA^2PC^\top$$

$$\Sigma_2^o := \mathbb{E}[o_{t+2}(\delta_o^\top o_{t+1}) o_t^\top]$$

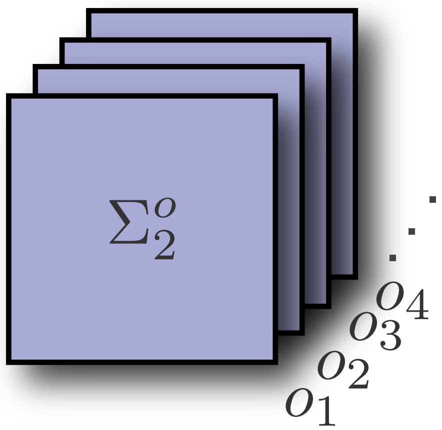






a tensor

Spectral Learning for HMMs

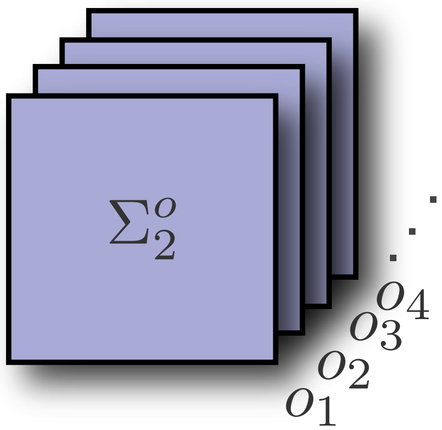


a tensor

$$\Sigma_2^o \quad := \quad \mathbb{E}[o_{t+2}(\delta_o^\top o_{t+1}) o_t^\top]$$

Tuesday, June 26, 2012

Spectral Learning for HMMs



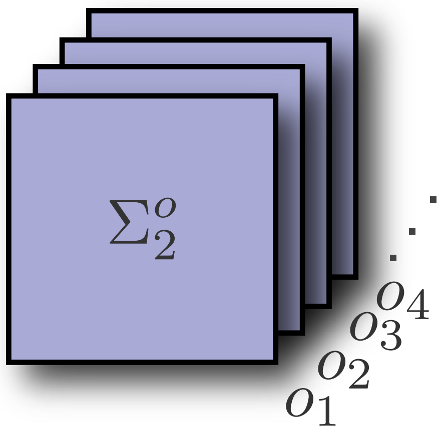
a tensor

$$
\begin{aligned}
\Sigma_2^o &:= \mathbb{E}[o_{t+2}(\delta_o^\top o_{t+1})o_t^\top] \\
&= \mathbb{E}[\mathbb{E}[o_{t+2}(\delta_o^\top o_{t+1})o_t^\top \mid x_t]] \\
&= \mathbb{E}[\mathbb{E}[o_{t+2}(\delta_o^\top o_{t+1}) \mid x_t]\mathbb{E}[o_t^\top \mid x_t]] \\
&= \mathbb{E}[\mathbb{E}[o_{t+2} \mid x_t, o_{t+1} = o]\mathbb{P}[o_{t+1} = o \mid x_t](Cx_t)^\top] \\
&= \mathbb{E}[\mathbb{E}[o_{t+2} \mid x_t, o_{t+1} = o](\mathbf{1}^\top A_o x_t)(Cx_t)^\top] \\
&= \mathbb{E}\left[CA\left[\frac{A_o x_t}{\mathbf{1}^\top A_o x_t}\right](\mathbf{1}^\top A_o x_t)(Cx_t)^\top\right] \\
&= \mathbb{E}[CAA_o x_t(Cx_t)^\top] \\
&= CAA_o\mathbb{E}[x_t x_t^\top]C^\top \\
&= CAA_o PC^\top
\end{aligned}
$$

Tuesday, June 26, 2012

sense
learn
act

Spectral Learning for HMMs



a tensor

$$\Sigma_2^o \quad := \quad \mathbb{E}[o_{t+2}(\delta_o^\top o_{t+1})o_t^\top]$$

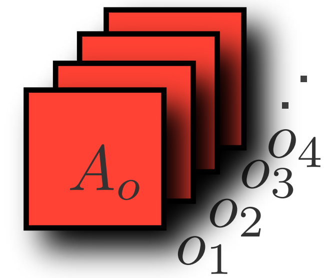
... and then a miracle occurs!

$$= \quad CAA_oPC^\top$$

sense
learn
act

Spectral Learning for HMMs

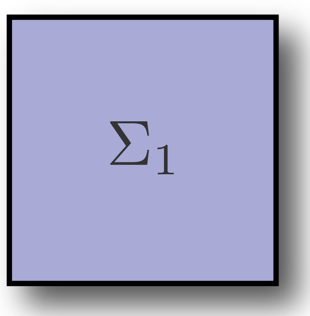
Goal is to find similarity transforms of A_os

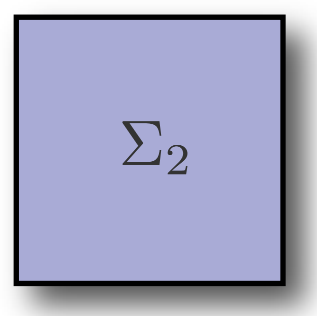


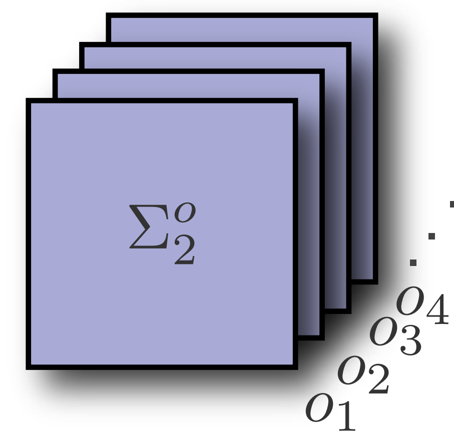
$$\Sigma_1 := \mathbb{E}\left[o_{t+1} o_t^\top\right] = CAPC^\top$$

$$\Sigma_2 := \mathbb{E}\left[o_{t+2} o_t^\top\right] = CA^2PC^\top$$

$$\Sigma_2^o := \mathbb{E}[o_{t+2}(\delta_o^\top o_{t+1}) o_t^\top] = CAA_oPC^\top$$







a **tensor**

Spectral Learning for HMMs

- Assume for simplicity that $m \geq n$ and that A and C are full rank
- Let U be the left n singular vectors of Σ_1,

$$\hat{A}_o \quad := \quad U^\top \Sigma_2^o (U^\top \Sigma_1)^\dagger$$

Spectral Learning for HMMs

- Assume for simplicity that $m \geq n$ and that A and C are full rank
- Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}\hat{A}_o &:= U^\top \Sigma_2^o (U^\top \Sigma_1)^\dagger \\ &= U^\top C A A_o P C^\top (U^\top C A P C^\top)^\dagger\end{aligned}$$

Spectral Learning for HMMs

- Assume for simplicity that $m \geq n$ and that A and C are full rank
- Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\hat{A}_o &:= U^\top \Sigma_2^o (U^\top \Sigma_1)^\dagger \\
&= U^\top C A A_o P C^\top (U^\top C A P C^\top)^\dagger \\
&= (U^\top C A) A_o (U^\top C A)^{-1} (U^\top C A) P C^\top (U^\top C A P C^\top)^\dagger
\end{aligned}$$

Spectral Learning for HMMs

- Assume for simplicity that $m \geq n$ and that A and C are full rank
- Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\hat{A}_o &:= U^\top \Sigma_2^o (U^\top \Sigma_1)^\dagger \\
&= U^\top C A A_o P C^\top (U^\top C A P C^\top)^\dagger \\
&= (U^\top C A) A_o (U^\top C A)^{-1} (U^\top C A) P C^\top (U^\top C A P C^\top)^\dagger \\
&= (U^\top C A) A_o (U^\top C A)^{-1}
\end{aligned}$$

sense
learn
act

Spectral Learning for HMMs

- Assume for simplicity that $m \geq n$ and that A and C are full rank
- Let U be the left n singular vectors of Σ_1,

$$
\begin{aligned}
\hat{A}_o &:= U^\top \Sigma_2^o (U^\top \Sigma_1)^\dagger \\
&= U^\top C A A_o P C^\top (U^\top C A P C^\top)^\dagger \\
&= (U^\top C A) A_o (U^\top C A)^{-1} (U^\top C A) P C^\top (U^\top C A P C^\top)^\dagger \\
&= (U^\top C A) A_o (U^\top C A)^{-1} \\
&= \underline{S A_o S^{-1}} \quad \text{similarity transform of } A_o
\end{aligned}
$$

sense
learn
act

Spectral Learning for HMMs

- Assume for simplicity that $m \geq n$ and that A and C are full rank
- Let U be the left n singular vectors of Σ_1,

$$\begin{aligned}
\hat{A}_o &:= U^\top \Sigma_2^o (U^\top \Sigma_1)^\dagger \\
&= U^\top C A A_o P C^\top (U^\top C A P C^\top)^\dagger \\
&= (U^\top C A) A_o (U^\top C A)^{-1} (U^\top C A) P C^\top (U^\top C A P C^\top)^\dagger \\
&= (U^\top C A) A_o (U^\top C A)^{-1} \\
&= \underline{S A_o S^{-1}} \quad \text{similarity transform of } A_o
\end{aligned}$$

- Additional parameters, like normalizer and initial state can be found in a similar manner
- S always cancels when predicting, filtering, simulating: e.g.

$$\mathbf{1} S^{-1} S A_{o_\tau} S^{-1} \ldots S A_{o_2} S^{-1} S A_{o_1} S^{-1} S \mathbb{P}[x_1]$$

sense
learn
act

Spectral Learning for HMMs

Spectral Learning Algorithm:

- Estimate Σ_1 and Σ_2^o from data
- Find $\widehat{U}$ by SVD
- Plug in for $\widehat{A}_o$ s

Learning is Consistent:

- Law of Large numbers for Σ_1 and Σ_2^o
- Continuity of formulas for $\widehat{A}_o$ s

Example: Clock (Revisited)

sense
learn
act

Example: Clock Pendulum

Simulations from models trained on clock data

Kalman Filter (spectral)
10 dimensions

HMM (Baum-Welch)
10 states

HMM? (spectral)
10 dimensions

sense
learn
act

Example: Clock Pendulum

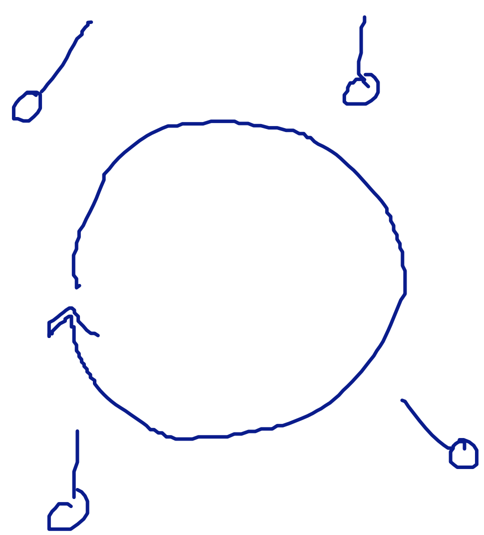
Simulations from models trained on clock data

Kalman Filter (spectral)
10 dimensions

HMM (Baum-Welch)
10 states

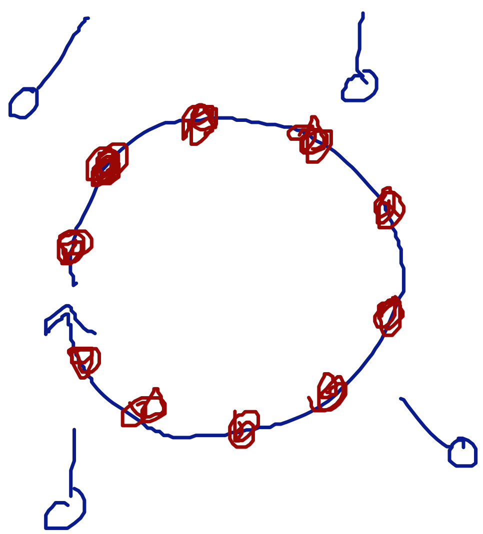
HMM? (spectral)
10 dimensions

Can We Generalize?

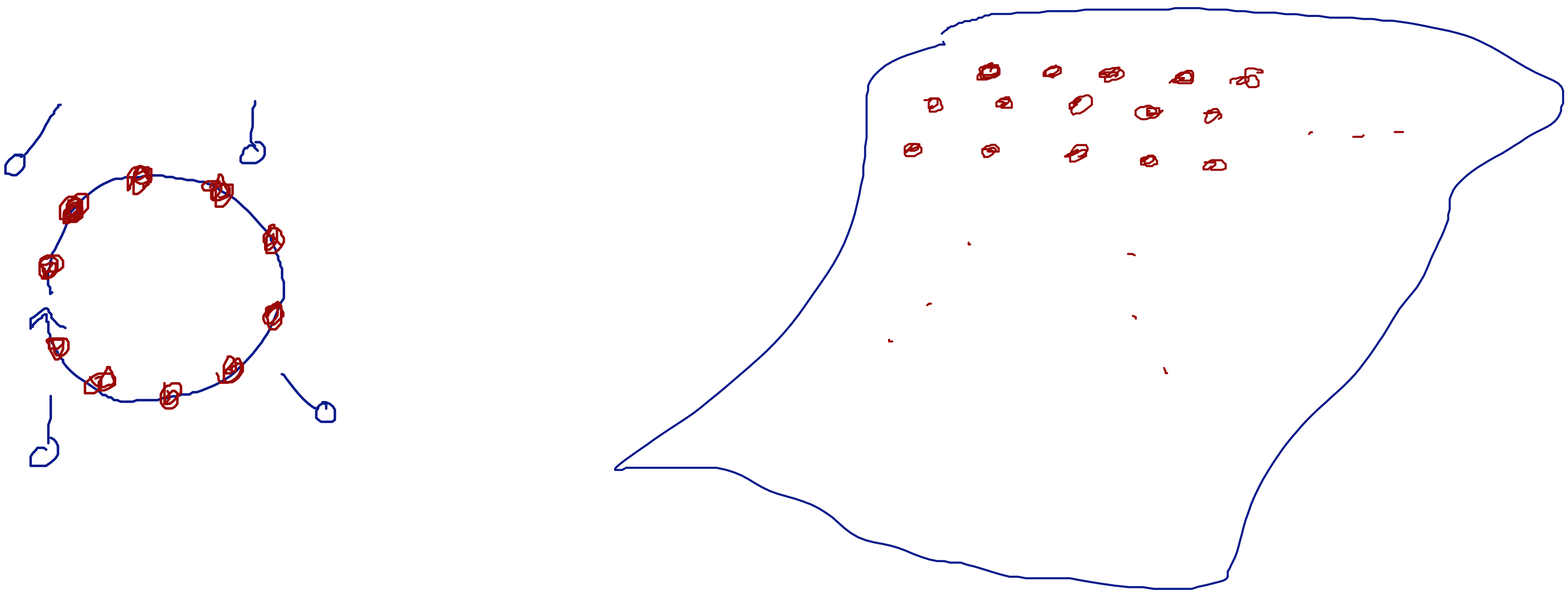


sense
learn
act

Can We Generalize?



Can We Generalize?



Lots of states: not a problem in itself, but means we need lots of data to learn transition & observation models

sense
learn
act

Generalizing HMMs

HMM state space:

- HMMs had $x \in \Delta$
  - intuition: number of discrete states = number of dimensions
- We now have $x \in S\Delta$
  - essentially equally restrictive
- Can we allow a more general state space?
  - e.g. # states > # dimensions
    - discretize more finely while keeping dimensionality the same

sense
learn
act

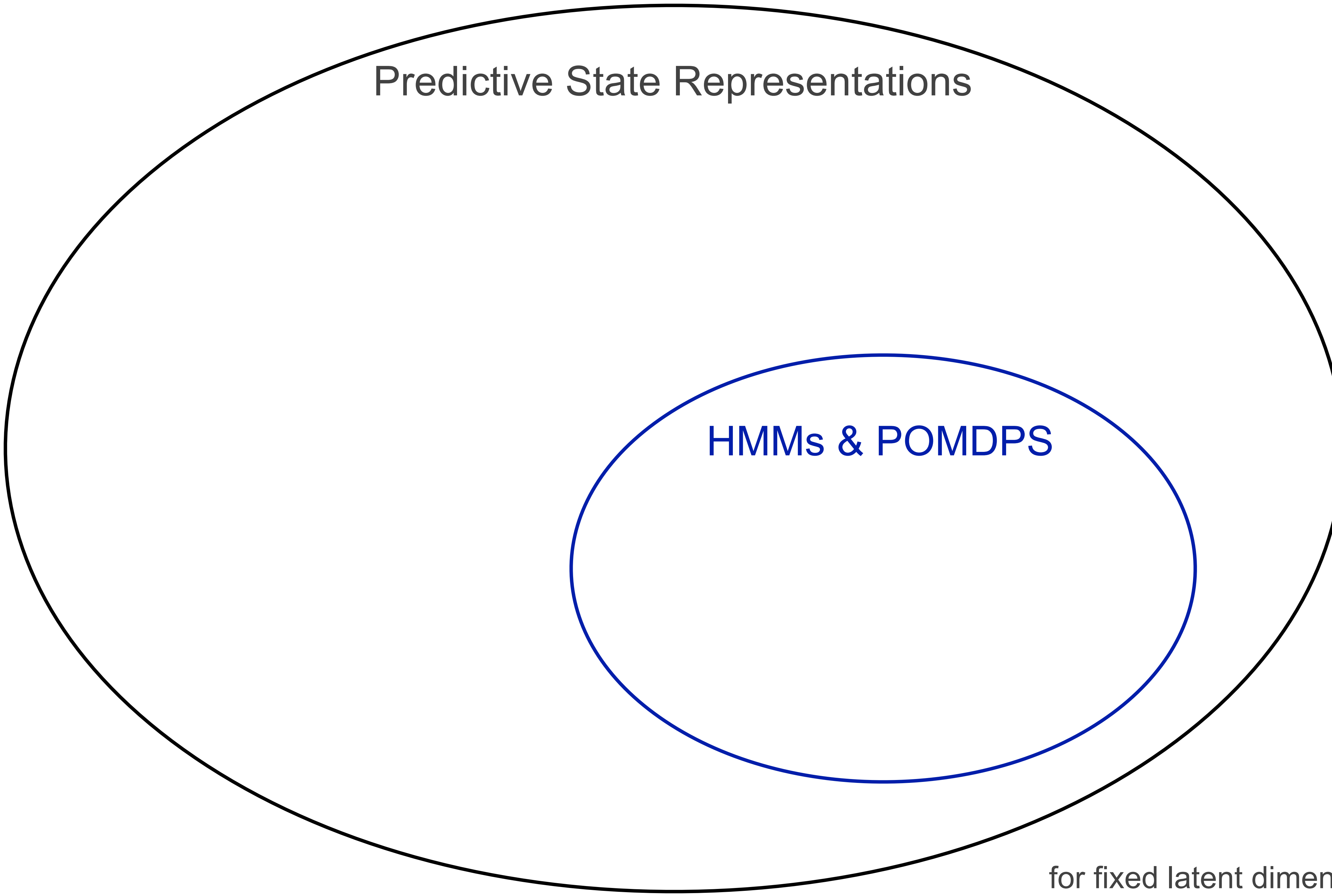
Predictive State Representations

≈ OOMs, multiplicity automata, etc…

- PSR: defined by transition matrices A_o, and a normalization vector
  - like HMM, but lift restriction of $X = S\Delta$
  - lift restrictions on A_os, top eigenvalue of $\sum_o A_o$ must be 1
  - instead of a set of discrete states, can think of state space as a possibly infinite-dimensional simplex projected onto a finite dimensional space
  - includes HMMs (and POMDPs) as special case

SSID for PSRs

PSRs are more expressive than HMMs & POMDPs ... and as easy to learn!



for fixed latent dimension *n*

Example: Clock Pendulum

Simulations from models trained on clock data

Kalman Filter (spectral)
10 states

HMM (Baum-Welch)
10 dimensions

PSR (spectral)
10 dimensions

sense
learn
act

Variations on Spectral Learning for PSRs

- Use arbitrary features of past and future observations
  - work from covariance of past, future features
  - good features make a big difference in practice
  - but still need a discrete set of transition matrices A_o
- Use different spectral decompositions to find state space: CCA, RRR
- Can extend to learn models with actions

sense
learn
act

Can We Generalize? Features!

- So far: allowed finer discretization of state space
- Can we improve? Allow continuous observations?
- Yes: Featurize!
  - let $\phi(o)$ be a feature function

$$\begin{aligned}
\Sigma_2^{\phi} &:= \mathbb{E}[o_{t+2}\phi(o_{t+1})o_t^\top] \\
&= \sum_o \phi(o)\mathbb{E}[o_{t+2}(\delta_o^\top o_{t+1})o_t^\top] \\
&= \sum_o \phi(o)\Sigma_2^o
\end{aligned}$$

$$\begin{aligned}
\hat{A}_\phi &:= U^\top \Sigma_2^\phi (U^\top \Sigma_1)^\dagger \\
&= \sum_o \phi(o)\hat{A}_o
\end{aligned}$$

store $\widehat{A}_\phi$ for many different ϕ , recover $\widehat{A}_o$ as needed

Can We Generalize? Infinite Features!

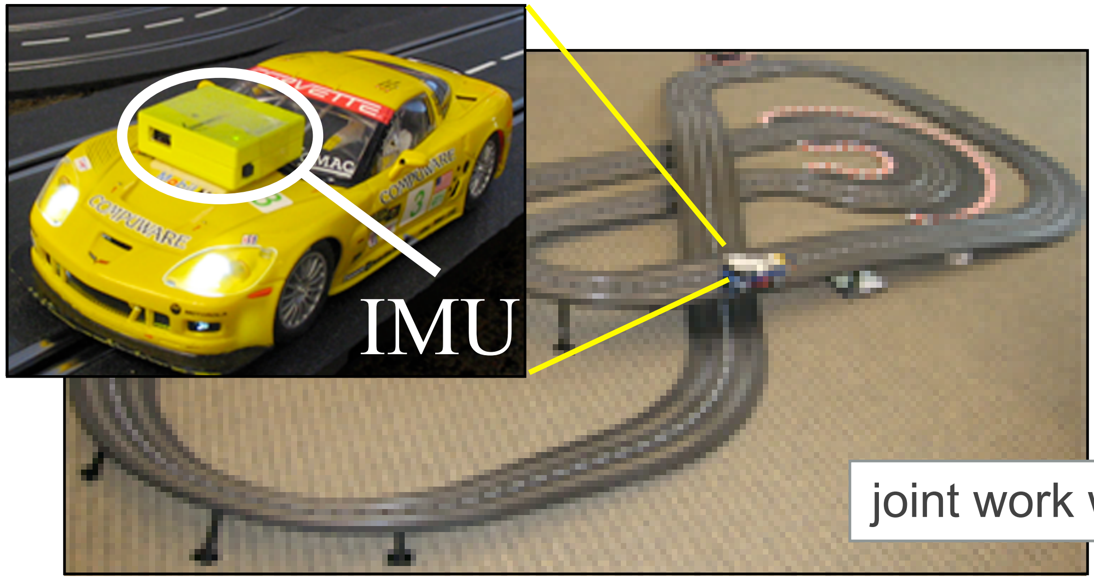
- If some features are good, more must be better!
  - Kernels
- Everything that we have seen is linear algebra
  - works just fine in an arbitrary RKHS
  - Can rewrite all of the formulas in terms of Gram matrices

Result: Hilbert Space Embeddings of Predictive State Representations

- handles near arbitrary observation distributions
- good prediction performance

sense
learn
act

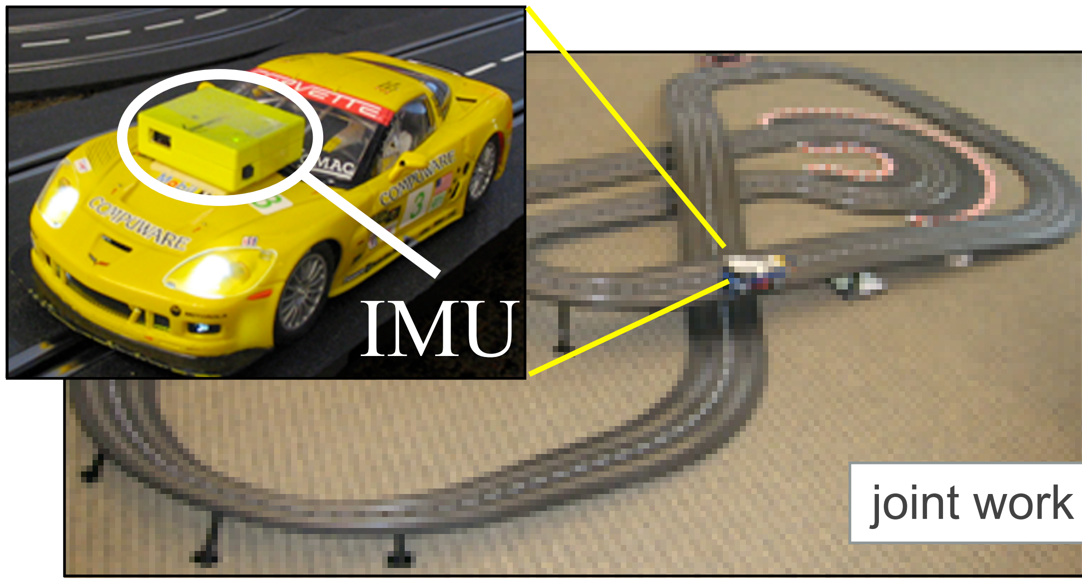
Example: Prediction (Slot Car Domain)



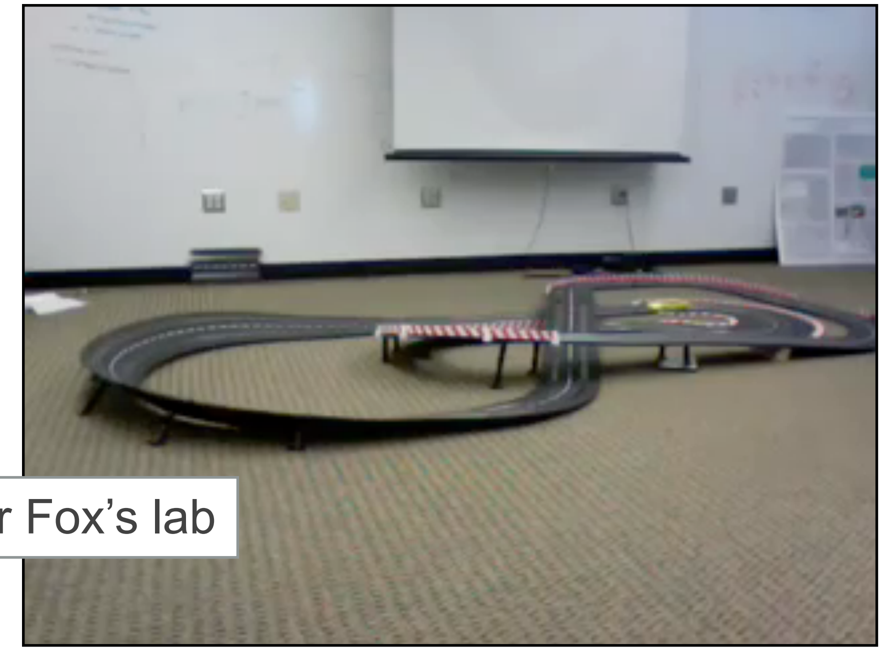
joint work with Dieter Fox’s lab

sense
learn
act

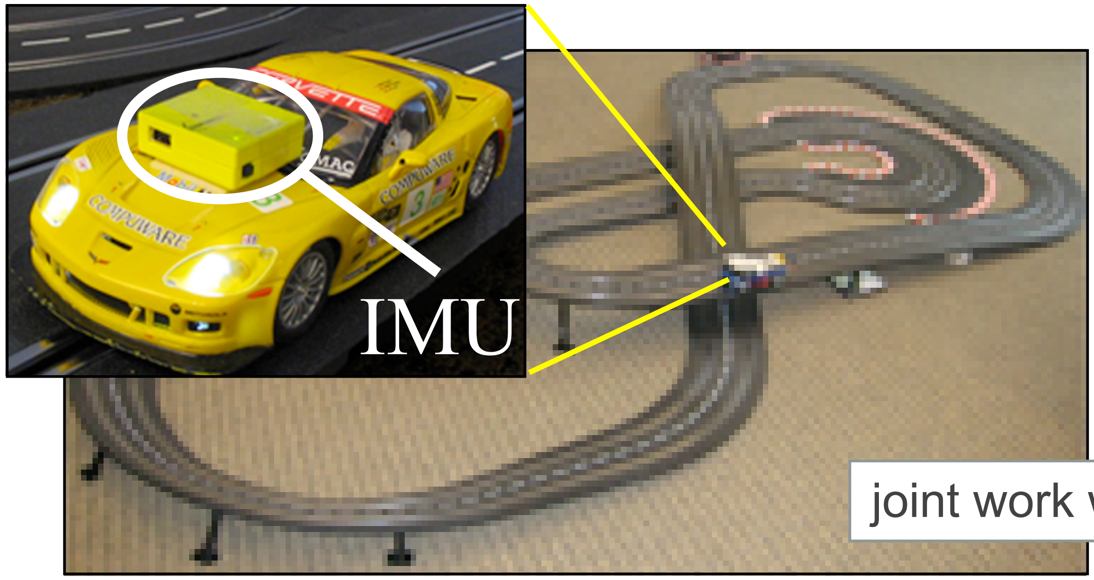
Example: Prediction (Slot Car Domain)



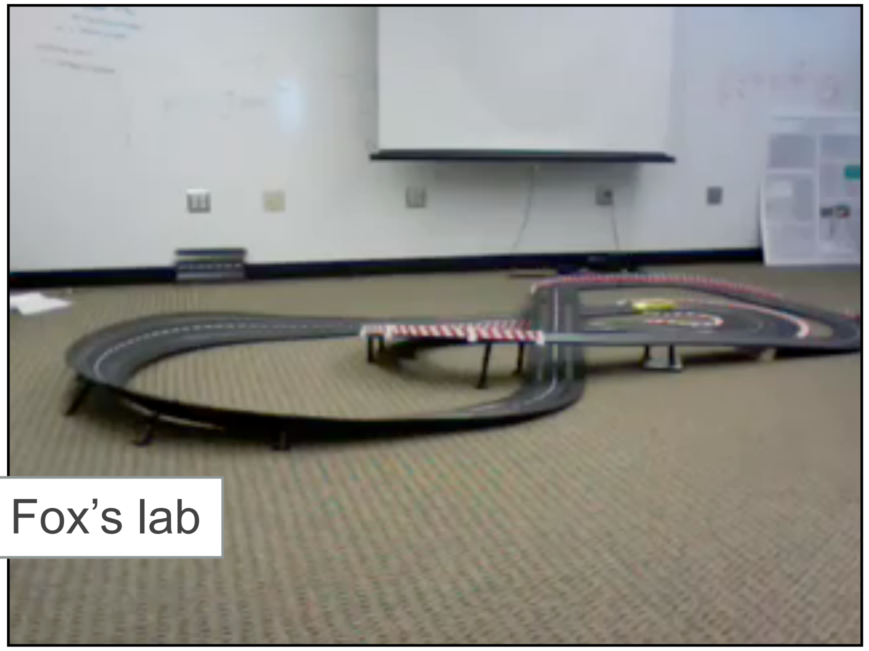
joint work with Dieter Fox’s lab

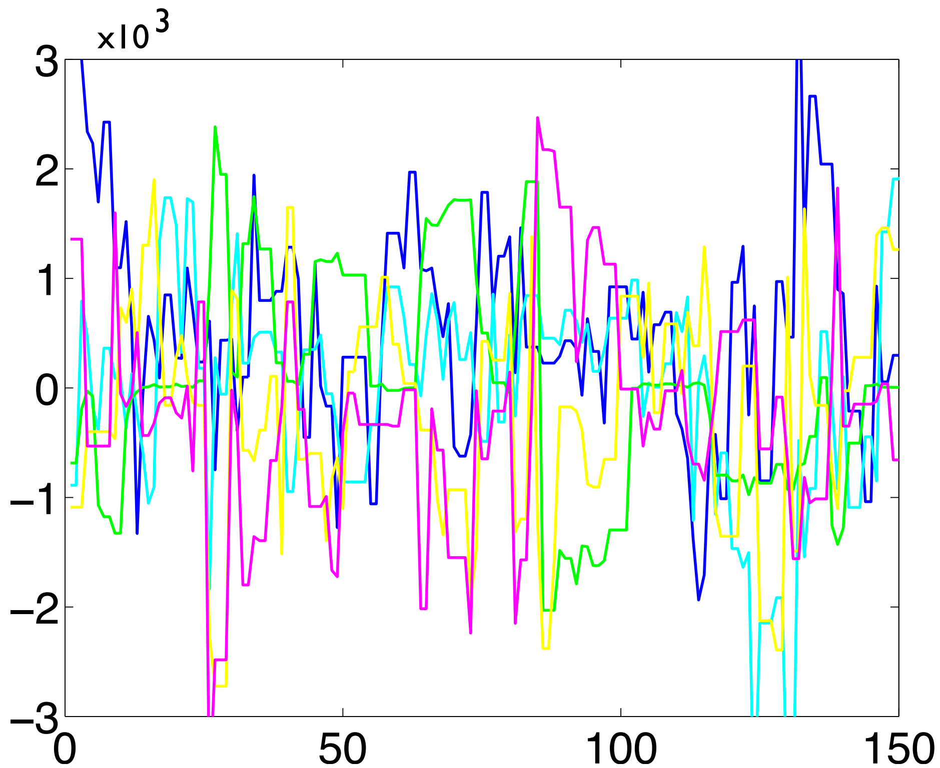


Example: Prediction (Slot Car Domain)



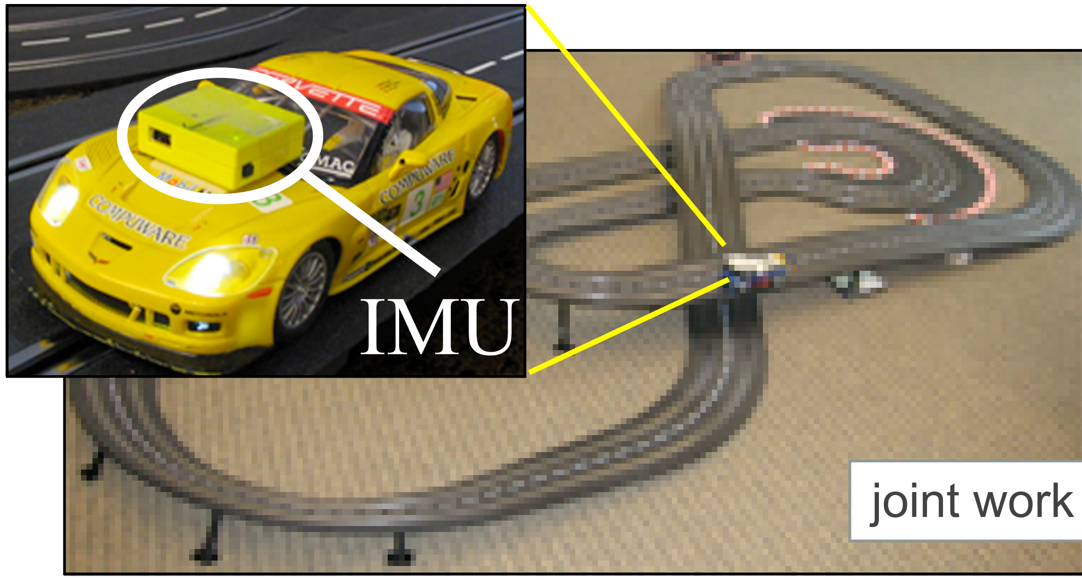
joint work with Dieter Fox's lab



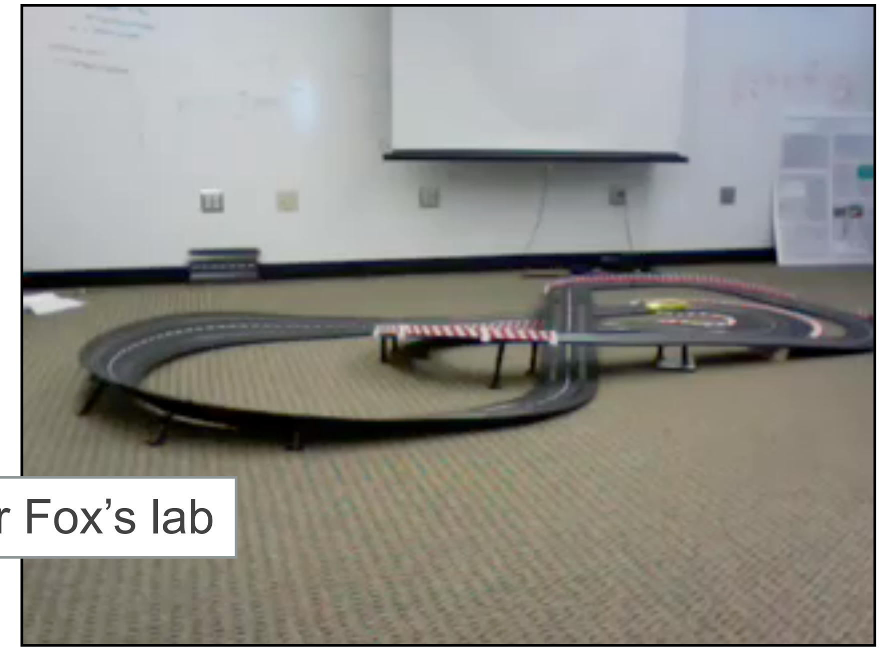


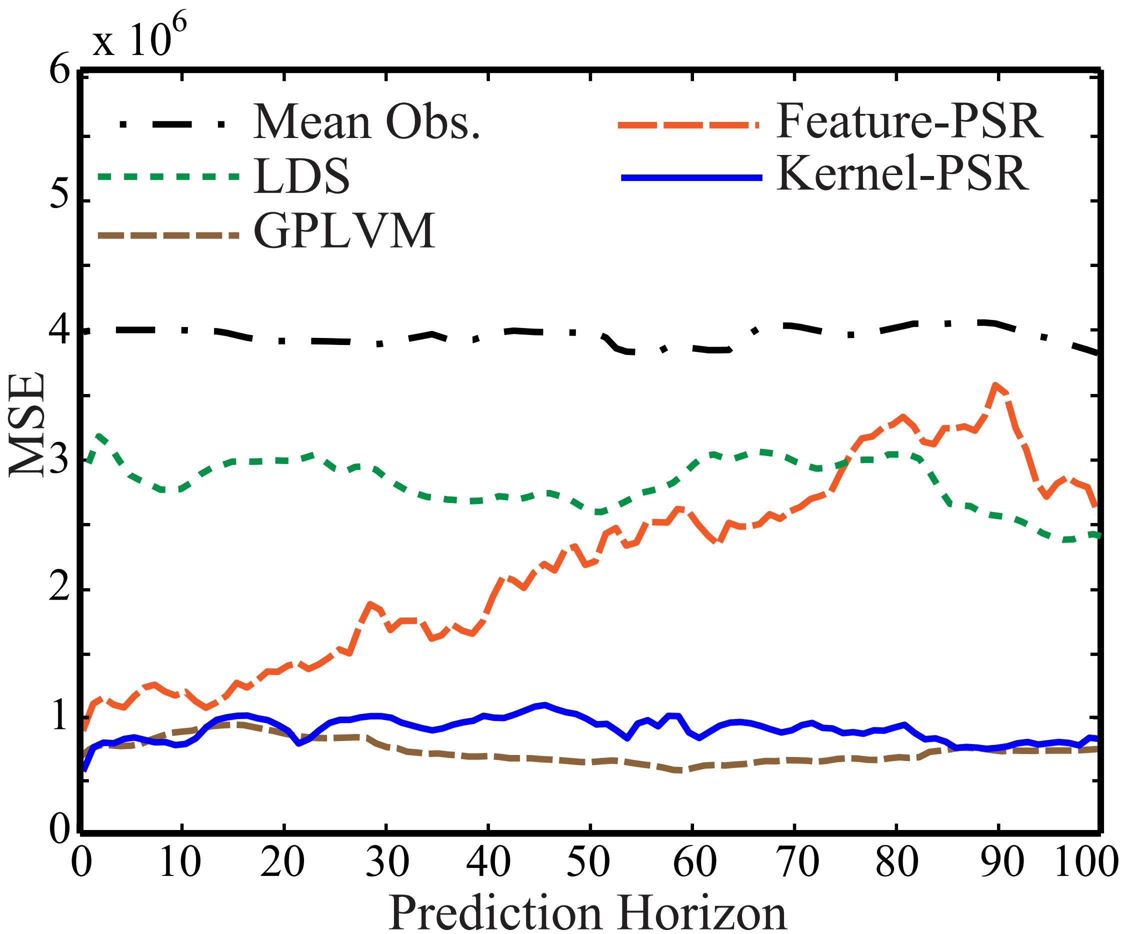
sense
learn
act

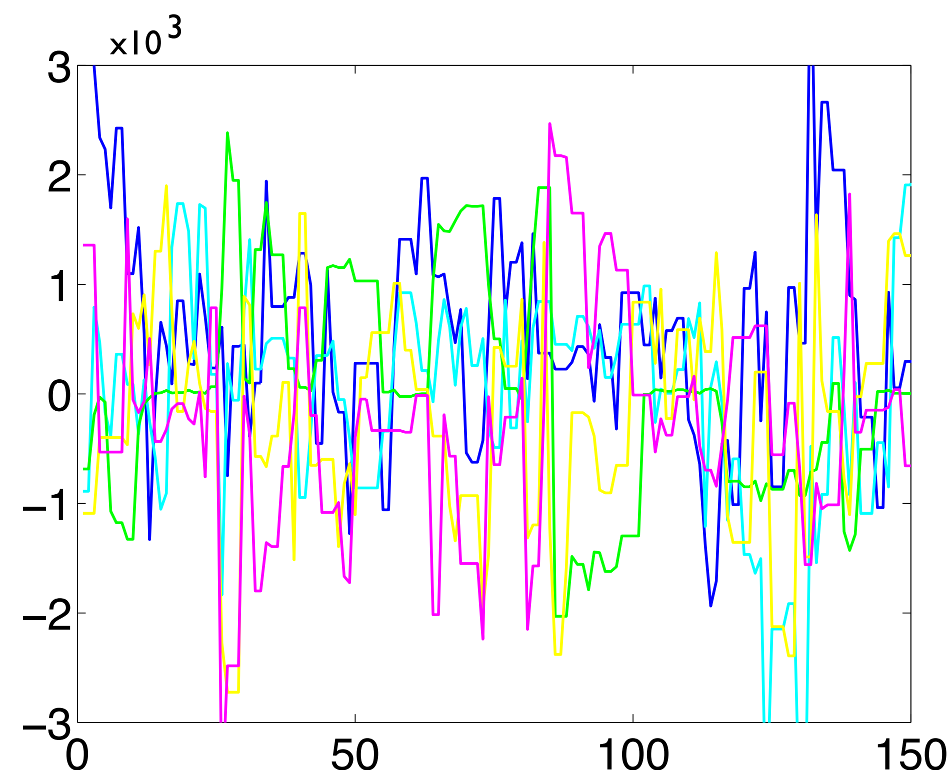
Example: Prediction (Slot Car Domain)



joint work with Dieter Fox's lab

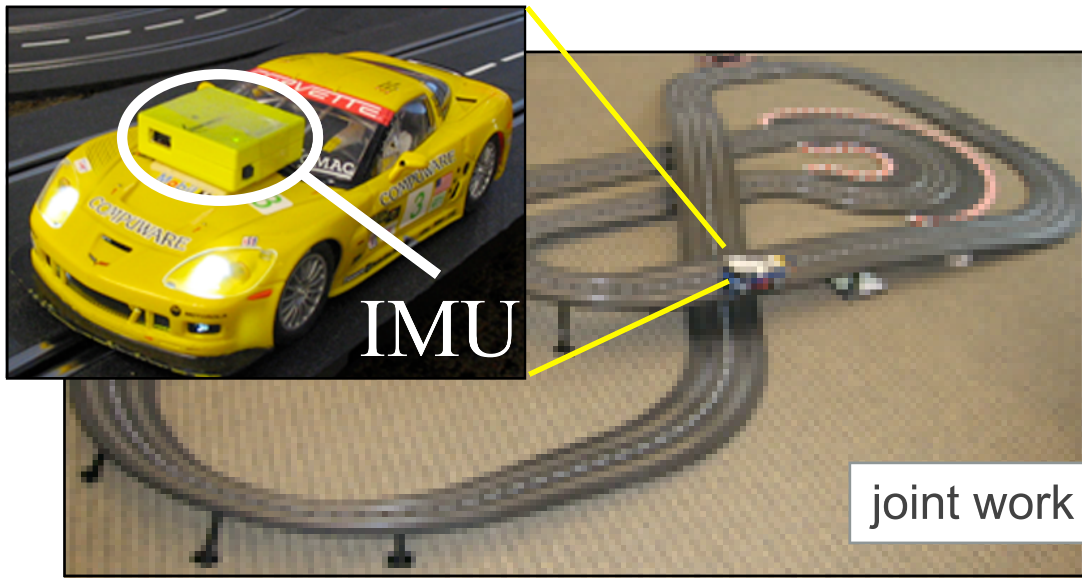




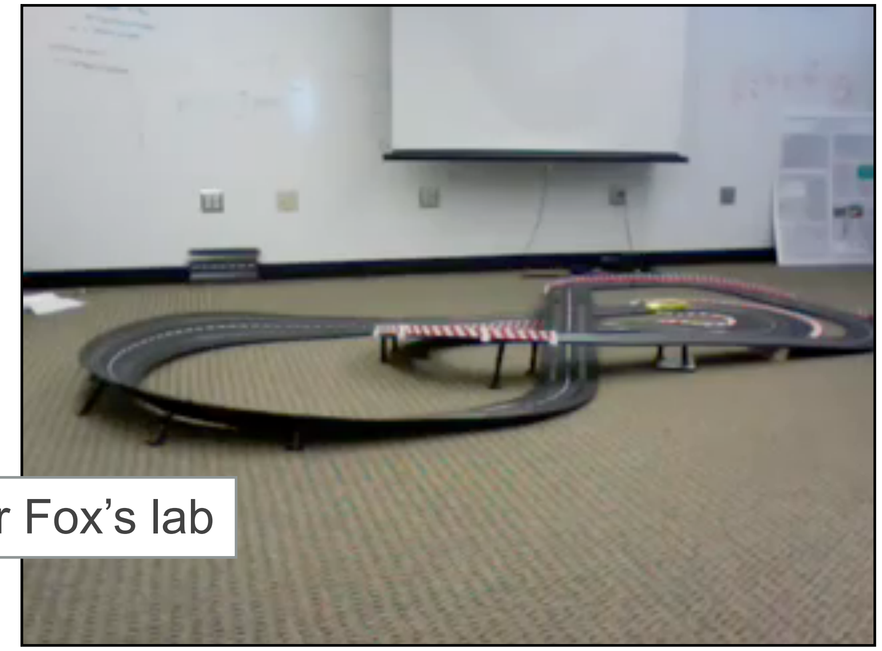


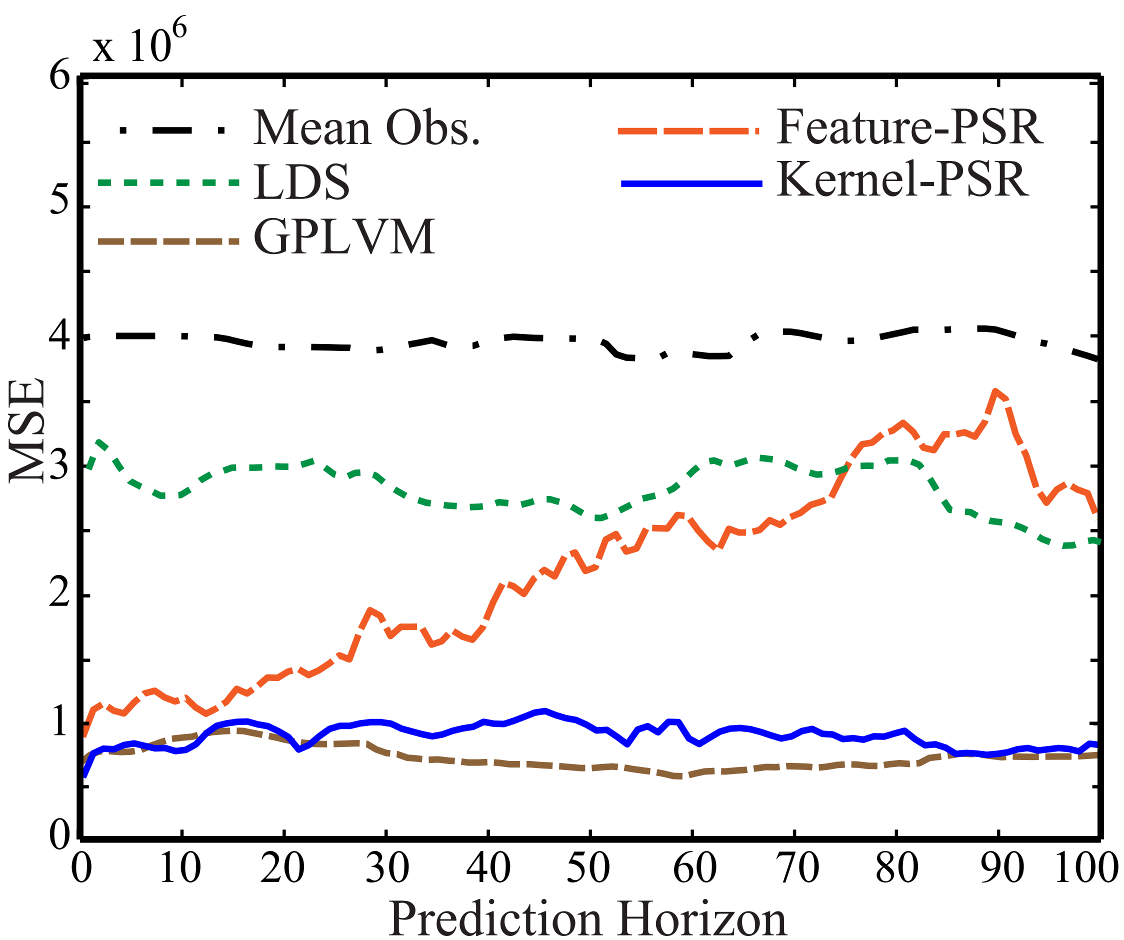
sense
learn
act

Example: Prediction (Slot Car Domain)



joint work with Dieter Fox's lab





Nonparametric Models Win

Gaussian Process Latent Variable Models
[Ko & Fox, 2010]
~1 day on 8-core i7 workstation
in Matlab/C++

Kernel PSRs:
11.6 seconds to learn model
on my laptop in Matlab

sense
learn
act

Making it All Fast:
Online Updates to Spectral Learning

- With each new observation, rank-1 update of:
  - SVD (Brand)
  - inverse (Sherman-Morrison)
- n features; latent dimension d; T steps
  - space = O(nd): may fit in cache!
  - time = $O(nd^2T)$: bounded time per example
- Small loss in statistical efficiency (estimated subspace rotates), but can deal with it
- Problem: no rank-1 update of k-SVD
  - can use random projections

sense
learn
act

Summary

- Learn dynamical system models with no local optima, fast online computation
- In contrast with many other methods, learning and inference is extremely fast and robust
- Nonparametric (kernel-based) version handles near-arbitrary observation distributions
- One general principle yields algorithms for Kalman System ID, HMMs, PSRs
- Good results from a general-purpose algorithm on problems typically tackled by lots of engineering

sense learn act

Thank You!