Learning from Dynamics

Ankur Moitra (MIT)

von Neumann Lecture, January 3rd

based on joint work with Ainesh Bakshi (MIT),
Allen Liu (MIT) and Morris Yau (MIT)

LINEAR DYNAMICAL SYSTEMS

Canonical model for time series data

$$\underbrace{x_{t+1}}_{\text{hidden state}} = Ax_t + B\underbrace{u_t}_{\text{control}} + w_t$$

$$\underbrace{y_t}_{\text{observation}} = Cx_t + Du_t + z_t$$

LINEAR DYNAMICAL SYSTEMS

Canonical model for time series data

$$x_{t+1} = \underbrace{A}_{\text{transition matrix}} x_t + \underbrace{B}_{\text{control matrix}} u_t + w_t$$

$$y_t = \underbrace{C}_{\text{sensing matrix}} x_t + \underbrace{D}_{\text{feedthrough matrix}} u_t + z_t$$

LINEAR DYNAMICAL SYSTEMS

Canonical model for time series data

$$x_{t+1} = Ax_t + Bu_t + \underbrace{w_t}_{\text{process noise}}$$

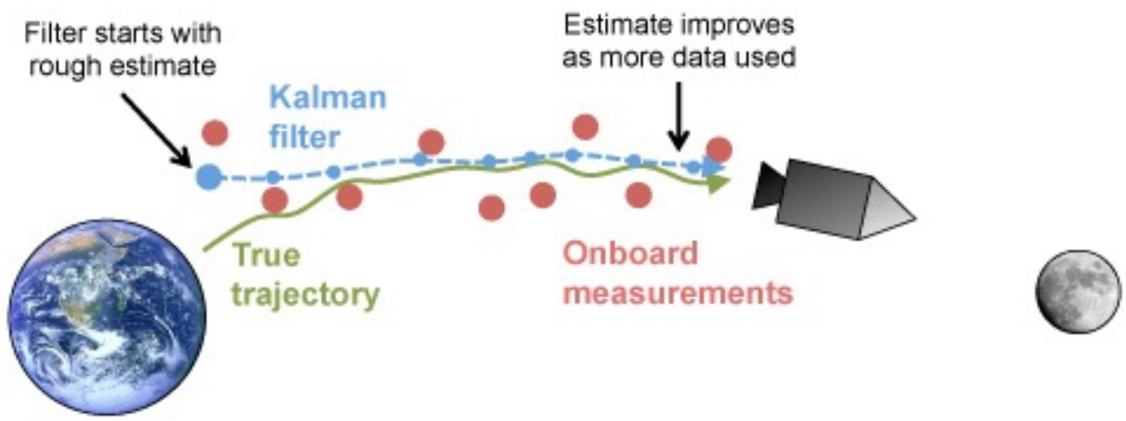
$$y_t = Cx_t + Du_t + \underbrace{z_t}_{\text{observation noise}}$$

APPLICATIONS

Robotics/Navigation/Tracking

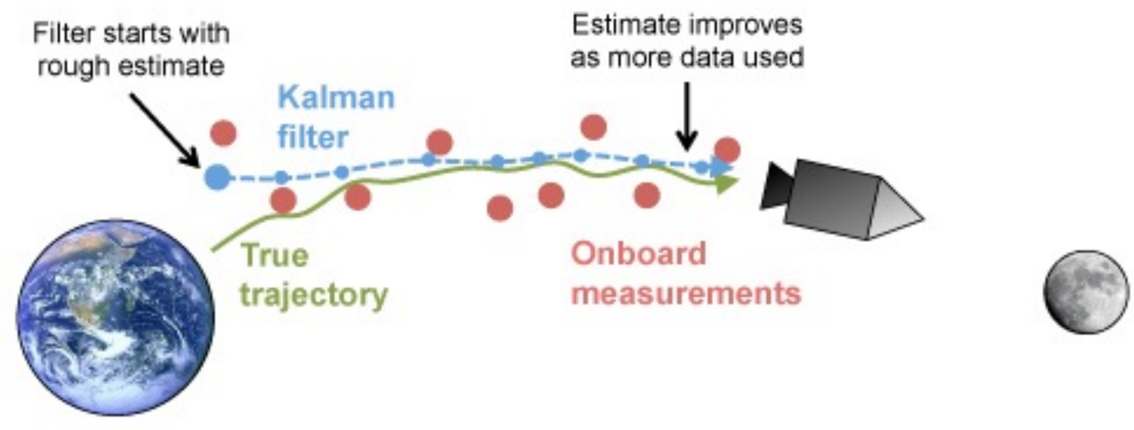
APPLICATIONS

Robotics/Navigation/Tracking



APPLICATIONS

Robotics/Navigation/Tracking



In particular

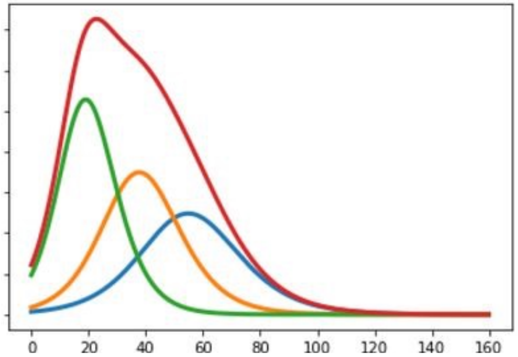
$$x_t = \begin{bmatrix} \text{position} \\ \text{velocity} \\ \text{acceleration} \end{bmatrix} \qquad A = \begin{bmatrix} \text{Laws of} \\ \text{Motion} \end{bmatrix}$$

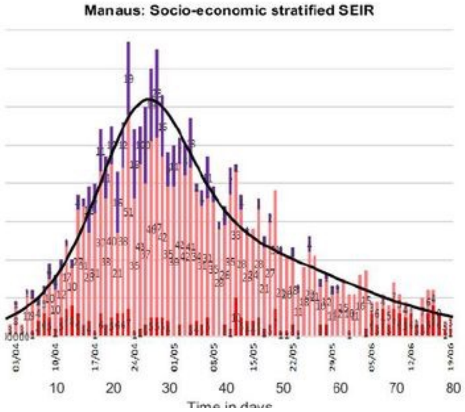
APPLICATIONS

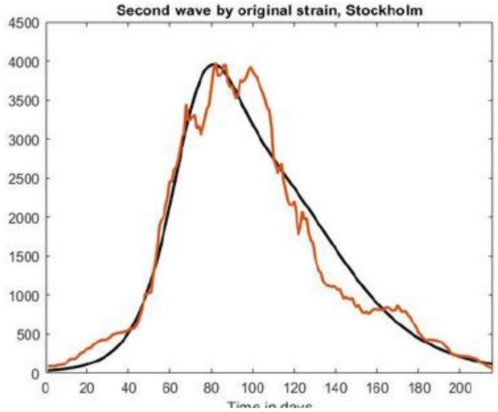
Biology/Epidemiology

APPLICATIONS

Biology/Epidemiology

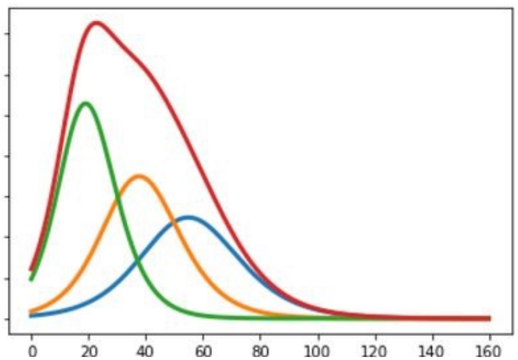


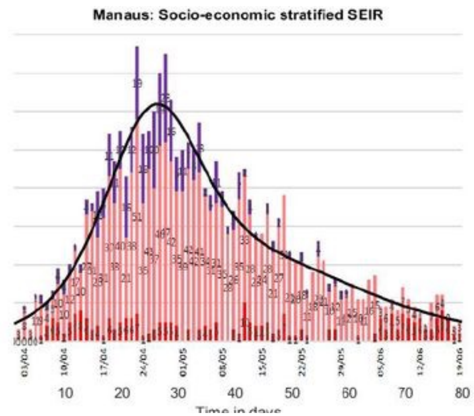


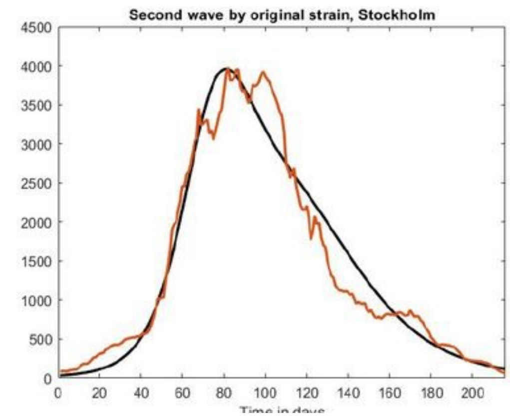


APPLICATIONS

Biology/Epidemiology







In particular

$$x_t = \begin{bmatrix} \text{susceptible} \\ \text{exposed} \\ \text{infected} \\ \text{recovered} \end{bmatrix} \qquad A = \begin{bmatrix} \text{State} \\ \text{Machine} \end{bmatrix}$$

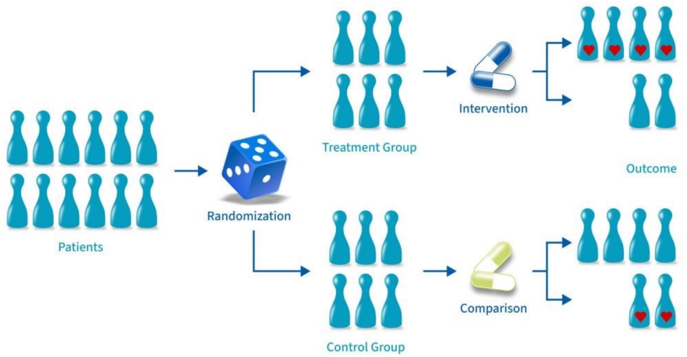
APPLICATIONS

Medicine

APPLICATIONS

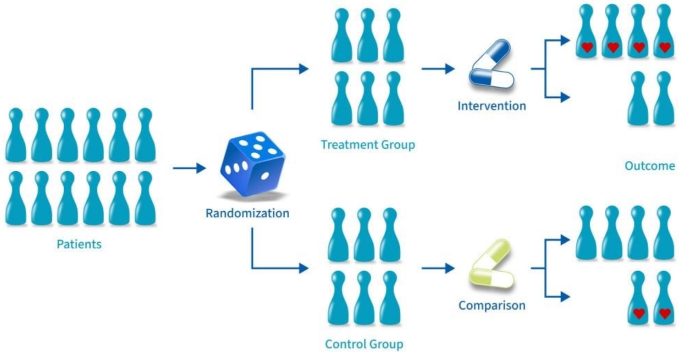
Medicine

Randomized Control Trial



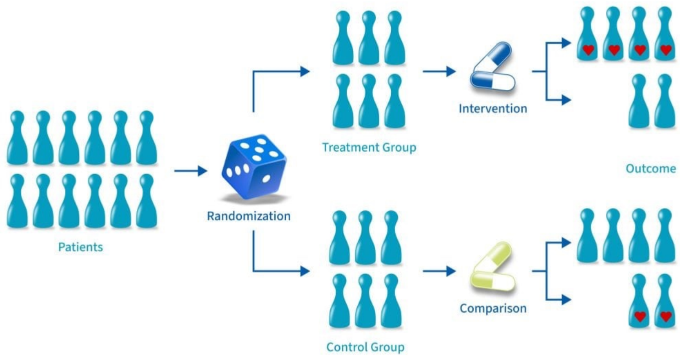
APPLICATIONS

Medicine

Randomized Control Trial	N-of-1 Trial
	Single patient, random switching

APPLICATIONS

Medicine

Randomized Control Trial	N-of-1 Trial
	Single patient, random switching

Speech Processing/Econometrics/Finance/etc

LINEAR DYNAMICAL SYSTEMS

Canonical model for time series data

$$\underbrace{x_{t+1}}_{\text{hidden state}} = Ax_t + B\underbrace{u_t}_{\text{control}} + w_t$$

$$\underbrace{y_t}_{\text{observation}} = Cx_t + Du_t + z_t$$

LINEAR DYNAMICAL SYSTEMS

Canonical model for time series data

$$\underbrace{x_{t+1}}_{\text{hidden state}} = Ax_t + B\underbrace{u_t}_{\text{control}} + w_t$$

$$\underbrace{y_t}_{\text{observation}} = Cx_t + Du_t + z_t$$

+ When the parameters are known, making predictions and inferences is easy!

LINEAR DYNAMICAL SYSTEMS

Canonical model for time series data

$$\underbrace{x_{t+1}}_{\text{hidden state}} = Ax_t + B\underbrace{u_t}_{\text{control}} + w_t$$

$$\underbrace{y_t}_{\text{observation}} = Cx_t + Du_t + z_t$$

+ When the parameters are known, making predictions and inferences is easy!

? But how do you learn its parameters?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- **Main Problem**
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

MAIN PROBLEM (INFORMAL)

Given one long trajectory

Inputs/Controls: $u_1, u_2, \ldots, u_T$

Outputs: $y_1, y_2, \ldots, y_T$

can we estimate A, B, C and D ?

MAIN PROBLEM (INFORMAL)

Given one long trajectory

Inputs/Controls: $u_1, u_2, \ldots, u_T$

Outputs: $y_1, y_2, \ldots, y_T$

can we estimate A, B, C and D ?

How do you measure closeness of the parameters?

AN ASIDE

Definition: We say that two linear dynamical systems are equivalent if for any sequence of adaptively chosen inputs

$$u_{t+1} = f(y_1, \ldots, y_t, u_1, \ldots, u_t)$$

they generate same distribution on outputs, up to a transformation of the noise

AN ASIDE

Definition: We say that two linear dynamical systems are equivalent if for any sequence of adaptively chosen inputs

$$u_{t+1} = f(y_1, \ldots, y_t, u_1, \ldots, u_t)$$

they generate same distribution on outputs, up to a transformation of the noise

Proposition: Two linear dynamical systems with Gaussian noise are equivalent iff $\exists U$

$$A = U^{-1}\widehat{A}U, \; B = U^{-1}\widehat{B}, \; C = \widehat{C}U \text{ and } D = \widehat{D}$$

AN ASIDE

Definition: We say that two linear dynamical systems are equivalent if for any sequence of adaptively chosen inputs

$$u_{t+1} = f(y_1, \ldots, y_t, u_1, \ldots, u_t)$$

they generate same distribution on outputs, up to a transformation of the noise

Proposition: Two linear dynamical systems with Gaussian noise are equivalent iff $\exists U$

$$A = U^{-1}\widehat{A}U,\ B = U^{-1}\widehat{B},\ C = \widehat{C}U \text{ and } D = \widehat{D}$$

i.e. they differ by a reparameterization of the hidden state

AN ASIDE

Definition: We say that two linear dynamical systems are equivalent if for any sequence of adaptively chosen inputs

$$u_{t+1} = f(y_1, \ldots, y_t, u_1, \ldots, u_t)$$

they generate same distribution on outputs, up to a transformation of the noise

Proposition: Two linear dynamical systems with Gaussian noise are equivalent iff $\exists U$

$$A = U^{-1}\widehat{A}U,\ B = U^{-1}\widehat{B},\ C = \widehat{C}U \text{ and } D = \widehat{D}$$

This defines a natural parameter distance

MAIN PROBLEM (FORMAL)

Given one long trajectory

Inputs/Controls: $u_1, u_2, \ldots, u_T$

Outputs: $y_1, y_2, \ldots, y_T$

can we find $\widehat{A}, \widehat{B}, \widehat{C}$ and $\widehat{D}$ such that $\exists U$

$$\|A - U^{-1}\widehat{A}U\|_F \leq \epsilon \text{ , } \|B - U^{-1}\widehat{B}\|_F \leq \epsilon$$

$$\|C - \widehat{C}U\|_F \leq \epsilon \text{ and } \|D - \widehat{D}\|_F \leq \epsilon \text{ ?}$$

MAIN PROBLEM (FORMAL)

Given one long trajectory

Inputs/Controls: $u_1, u_2, \ldots, u_T$

Outputs: $y_1, y_2, \ldots, y_T$

can we find $\widehat{A}, \widehat{B}, \widehat{C}$ and $\widehat{D}$ such that $\exists U$

$$\|A - U^{-1}\widehat{A}U\|_F \leq \epsilon \text{ , } \|B - U^{-1}\widehat{B}\|_F \leq \epsilon \text{ ,}$$

$$\|C - \widehat{C}U\|_F \leq \epsilon \text{ and } \|D - \widehat{D}\|_F \leq \epsilon \text{ ?}$$

Is there a polynomial time/sample algorithm for learning?

PRIOR WORK

Widespread assumption on **spectral radius**, often unreasonable:

strict stability $\rho(A) < 1$	marginal stability $\rho(A) \leq 1$

PRIOR WORK

Widespread assumption on **spectral radius**, often unreasonable:

strict stability $\rho(A) < 1$	marginal stability $\rho(A) \leq 1$
fails even in simple cases	

PRIOR WORK

Widespread assumption on **spectral radius**, often unreasonable:

strict stability $\rho(A) < 1$	marginal stability $\rho(A) \leq 1$
fails even in simple cases	
no long-range correlations	

PRIOR WORK

Widespread assumption on **spectral radius**, often unreasonable:

strict stability $\rho(A) < 1$	marginal stability $\rho(A) \leq 1$
fails even in simple cases	
no long-range correlations	
get fresh samples	

PRIOR WORK

Widespread assumption on **spectral radius**, often unreasonable:

strict stability $\rho(A) < 1$	marginal stability $\rho(A) \leq 1$
fails even in simple cases **no long-range correlations** **get fresh samples**	**otherwise system would explode**

PRIOR WORK

Widespread assumption on **spectral radius**, often unreasonable:

strict stability $\rho(A) < 1$	marginal stability $\rho(A) \leq 1$
fails even in simple cases **no long-range correlations** **get fresh samples**	**otherwise system would explode**

Bounds depend on $\dfrac{1}{1-\rho(A)}$, degrade as $\rho(A) \to 1$

PRIOR WORK

Widespread assumption on **spectral radius**, often unreasonable:

strict stability $\rho(A) < 1$	marginal stability $\rho(A) \leq 1$
fails even in simple cases **no long-range correlations** **get fresh samples**	**otherwise system would explode**

Bounds depend on $\dfrac{1}{1-\rho(A)}$, degrade as $\rho(A) \to 1$

Do long-range correlations actually obstruct learning?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- **Well-Posedness and Our Results**

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

OBSERVABILITY AND CONTROLLABILITY

Definition: The **observability matrix of order s** is

$$O_s = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{s-1} \end{bmatrix}$$

OBSERVABILITY AND CONTROLLABILITY

Definition: The **observability matrix of order s** is

$$O_s = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{s-1} \end{bmatrix}$$

Proposition [informal]: If it doesn't have full column rank, there is some portion of the state space we miss even over s steps

OBSERVABILITY AND CONTROLLABILITY

Definition: The **observability matrix of order s** is

$$O_s = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{s-1} \end{bmatrix}$$

Proposition [informal]: If it doesn't have full column rank, there is some portion of the state space we miss even over s steps

Proof: If we move the state x_t in some direction z then…

OBSERVABILITY AND CONTROLLABILITY

Definition: The **observability matrix of order s** is

$$O_s = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{s-1} \end{bmatrix}$$

Proposition [informal]: If it doesn't have full column rank, there is some portion of the state space we miss even over s steps

Proof: If we move the state x_t in some direction z then

$$Cx_t = C(x_t + z)$$

OBSERVABILITY AND CONTROLLABILITY

Definition: The **observability matrix of order s** is

$$O_s = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{s-1} \end{bmatrix}$$

Proposition [informal]: If it doesn't have full column rank, there is some portion of the state space we miss even over s steps

Proof: If we move the state x_t in some direction z then

$$Cx_t = C(x_t + z) \implies \text{no effect on } y_t$$

OBSERVABILITY AND CONTROLLABILITY

Definition: The **observability matrix of order s** is

$$O_s = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{s-1} \end{bmatrix}$$

Proposition [informal]: If it doesn't have full column rank, there is some portion of the state space we miss even over s steps

Proof: If we move the state x_t in some direction z then

$$CAx_t = CA(x_t + z)$$

OBSERVABILITY AND CONTROLLABILITY

Definition: The **observability matrix of order s** is

$$O_s = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{s-1} \end{bmatrix}$$

Proposition [informal]: If it doesn't have full column rank, there is some portion of the state space we miss even over s steps

Proof: If we move the state x_t in some direction z then

$$CAx_t = CA(x_t + z) \implies \text{no effect on } y_{t+1}\text{, etc} \quad \blacksquare$$

OBSERVABILITY AND CONTROLLABILITY

Similarly:

Definition: The **controllability matrix of order s** is

$$Q_s = \left[\, B\,,\; AB\,, \cdots, A^{s-1}B \,\right]$$

OBSERVABILITY AND CONTROLLABILITY

Similarly:

Definition: The **controllability matrix of order s** is

$$Q_s = \left[B,\ AB, \cdots, A^{s-1}B \right]$$

Proposition [informal]: If it doesn't have full row rank, there is some portion of the state space that cannot be reached by appropriate inputs

OBSERVABILITY AND CONTROLLABILITY

Similarly:

Definition: The **controllability matrix of order s** is

$$Q_s = \left[B, \; AB, \cdots, A^{s-1}B \right]$$

Proposition [informal]: If it doesn't have full row rank, there is some portion of the state space that cannot be reached by appropriate inputs

Necessity of these assumptions goes back to Kalman in 1960

MAIN RESULTS

Theorem [Bakshi, Liu, Moitra, Yau]: There is a polynomial time algorithm for learning any marginally stable linear dynamical system from one long trajectory under **quantitative observability and controllability***

***i.e. condition number bounds**

MAIN RESULTS

Theorem [Bakshi, Liu, Moitra, Yau]: There is a polynomial time algorithm for learning any marginally stable linear dynamical system from one long trajectory under **quantitative observability and controllability***

***i.e. condition number bounds**

Moreover these conditions are essentially minimal

Theorem [Bakshi, Liu, Moitra, Yau]: If the observability and controllability matrices are ill-conditioned for all s then learning is **information-theoretically impossible**

COMMENTS

[Simchowitz, Boczar, Recht] also gave algorithms under marginal stability, but **unspecified dependence on system parameters***

***i.e. can be exponential**

COMMENTS

[Simchowitz, Boczar, Recht] also gave algorithms under marginal stability, but **unspecified dependence on system parameters***

***i.e. can be exponential**

Also, renewed interest because of connections to recurrent neural networks (RNNs)

COMMENTS

[Simchowitz, Boczar, Recht] also gave algorithms under marginal stability, but **unspecified dependence on system parameters***

***i.e. can be exponential**

Also, renewed interest because of connections to recurrent neural networks (RNNs)

Before using LDS's as a prototype for reasoning about RNNs, need to understand their fundamental limits --- e.g. **what are minimal assumptions for learnability?**

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

METHOD OF MOMENTS

Introduced by Karl Pearson in 1894

- Estimate moments of the distribution from samples
- Setup system of equations in unknown parameters
- Solve to compute estimates

METHOD OF MOMENTS

Introduced by Karl Pearson in 1894

- Estimate moments of the distribution from samples
- Setup system of equations in unknown parameters
- Solve to compute estimates

Many successes in unsupervised learning, e.g. **HMMs**, **mixtures of gaussians**, **topic modeling**, **robust estimation**, etc

METHOD OF MOMENTS

Introduced by Karl Pearson in 1894

- Estimate moments of the distribution from samples
- Setup system of equations in unknown parameters
- Solve to compute estimates

Many successes in unsupervised learning, e.g. **HMMs**, **mixtures of gaussians**, **topic modeling**, **robust estimation**, etc

Is there a recipe for non-stationary data?

A BLUEPRINT

Definition: The **Markov parameters up to order 2s+1** are

$$G = \left[D, CB, CAB, \ldots, CA^{2s}B \right]$$

- **Estimate** the Markov parameters from samples
- Solve for the estimates using the **Ho-Kalman algorithm**

A BLUEPRINT

Definition: The **Markov parameters up to order 2s+1** are

$$G = \left[D, \ CB \ , \ CAB \ , \ \ldots , \ CA^{2s}B \ \right]$$

- **Estimate** the Markov parameters from samples
- Solve for the estimates using the **Ho-Kalman algorithm**

Proposition: Can find good estimates from just the Markov parameters

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- **The Ho-Kalman Algorithm**
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

THE HO-KALMAN ALGORITHM

Step #1: Form the Hankel matrix

$$H = \begin{bmatrix} CB & CAB & \cdots & CA^sB \\ CAB & CA^2B & & \vdots \\ \vdots & & \ddots & \\ CA^sB & \cdots & & CA^{2s}B \end{bmatrix}$$

THE HO-KALMAN ALGORITHM

Step #1: Form the Hankel matrix

H_1 $\quad$ H_2

$$H = \begin{bmatrix} CB & CAB & \cdots & CA^sB \\ CAB & CA^2B & & \vdots \\ \vdots & & \ddots & \\ CA^sB & \cdots & & CA^{2s}B \end{bmatrix}$$

THE HO-KALMAN ALGORITHM

Step #1: Form the Hankel matrix

H_1 H_2

$$H = \begin{bmatrix} CB & CAB & \cdots & CA^sB \\ CAB & CA^2B & & \vdots \\ \vdots & & \ddots & \\ CA^sB & \cdots & & CA^{2s}B \end{bmatrix}$$

Claim: $H = O_{s+1}Q_{s+1}$

THE HO-KALMAN ALGORITHM

Step #1: Form the Hankel matrix

H_1 $\quad H_2$

$$H = \begin{bmatrix} CB & CAB & \cdots & CA^sB \\ CAB & CA^2B & & \vdots \\ \vdots & & \ddots & \\ CA^sB & \cdots & & CA^{2s}B \end{bmatrix}$$

Claim: $H = O_{s+1} Q_{s+1}$

This is the hidden factorization we are looking for

Can we compute another factorization, and show equivalence?

Can we compute another factorization, and show equivalence?

Step #2: Compute the SVD

$$H_1 = U \Sigma V^T = \left(U \Sigma^{1/2} \right) \left(\Sigma^{1/2} V^\top \right)$$

Can we compute another factorization, and show equivalence?

Step #2: Compute the SVD

$$H_1 = U\Sigma V^T = \underbrace{\left(U\Sigma^{1/2}\right)}_{\widehat{O}}\underbrace{\left(\Sigma^{1/2}V^\top\right)}_{\widehat{Q}}$$

Can we compute another factorization, and show equivalence?

Step #2: Compute the SVD

$$H_1 = U\Sigma V^T = \underbrace{\left(U\Sigma^{1/2}\right)}_{\widehat{O}}\underbrace{\left(\Sigma^{1/2}V^\top\right)}_{\widehat{Q}}$$

$$= O_{s+1}Q_s$$

Can we compute another factorization, and show equivalence?

Step #2: Compute the SVD

$$H_1 = U\Sigma V^T = \underbrace{\left(U\Sigma^{1/2}\right)}_{\widehat{O}}\underbrace{\left(\Sigma^{1/2}V^\top\right)}_{\widehat{Q}}$$

$$= O_{s+1}Q_s$$

Lemma: If O_{s+1} and Q_s have full column and row rank resp. then

$$O_{s+1} = \widehat{O}T \text{ and } Q_s = T^{-1}\widehat{Q}$$

for some invertible transformation T

Now how do we estimate A ?

Now how do we estimate A ?

Step #3: Using what we know already

$$H_2 = O_{s+1} A Q_s$$

Now how do we estimate A ?

Step #3: Using what we know already

$$H_2 = O_{s+1} A Q_s$$

$$= \widehat{O} T A T^{-1} \widehat{Q}$$ **(from Step #2)**

Now how do we estimate A ?

Step #3: Using what we know already

$$H_2 = O_{s+1} A Q_s$$
$$= \widehat{O} T A T^{-1} \widehat{Q}$$ **(from Step #2)**

So if we set $\widehat{A} = \widehat{O}^{+} H_2 \widehat{Q}^{+}$ …

Now how do we estimate A ?

Step #3: Using what we know already

$$H_2 = O_{s+1} A Q_s$$
$$= \widehat{O} T A T^{-1} \widehat{Q} \quad \textbf{(from Step \#2)}$$

So if we set $\widehat{A} = \widehat{O}^+ H_2 \widehat{Q}^+$ we get

$$\Rightarrow \quad \widehat{A} = T A T^{-1} \quad \blacksquare$$

A BLUEPRINT

Definition: The **Markov parameters up to order 2s+1** are

$$G = \left[D, \; CB \, , \; CAB \, , \; \ldots , \; CA^{2s}B \right]$$

- **Estimate** the Markov parameters from samples
- Solve for the estimates using the **Ho-Kalman algorithm**

Proposition: Can find good estimates from just the Markov parameters

A BLUEPRINT

Definition: The **Markov parameters up to order 2s+1** are

$$G = \left[D, \; CB \;, \; CAB \;, \; \ldots, \; CA^{2s}B \right]$$

- **Estimate** the Markov parameters from samples
- Solve for the estimates using the **Ho-Kalman algorithm**

Proposition: Can find good estimates from just the Markov parameters

[Oymak, Ozay] gave stability analysis, if condition number is bdd

A BLUEPRINT

Definition: The **Markov parameters up to order 2s+1** are

$$G = \left[D, CB, CAB, \ldots, CA^{2s}B \right]$$

- **Estimate** the Markov parameters from samples
- Solve for the estimates using the **Ho-Kalman algorithm**

Main Challenge: How do we estimate the Markov parameters?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- **Controlling the Variance?**
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

A NAÏVE APPROACH

Observation: If the control and noises are independent and have identity covariance, then

$$\mathbb{E}[y_{t+j}u_t^\top] = \begin{cases} D & \text{if } j = 0 \\ CA^{j-1}B & \text{else} \end{cases}$$

A NAÏVE APPROACH

Observation: If the control and noises are independent and have identity covariance, then

$$\mathbb{E}[y_{t+j}u_t^\top] = \begin{cases} D & \text{if } j = 0 \\ CA^{j-1}B & \text{else} \end{cases}$$

Proof: Expand the recurrence, e.g. if $j = 1$

$$y_{t+1} = Cx_{t+1} + Du_{t+1} + z_{t+1}$$

A NAÏVE APPROACH

Observation: If the control and noises are independent and have identity covariance, then

$$\mathbb{E}[y_{t+j}u_t^\top] = \begin{cases} D & \text{if } j = 0 \\ CA^{j-1}B & \text{else} \end{cases}$$

Proof: Expand the recurrence, e.g. if $j = 1$

$$y_{t+1} = Cx_{t+1} + Du_{t+1} + z_{t+1}$$

$$= CAx_t + CBu_t + Cw_t + Du_{t+1} + z_{t+1}$$

A NAÏVE APPROACH

Observation: If the control and noises are independent and have identity covariance, then

$$\mathbb{E}[y_{t+j}u_t^\top] = \begin{cases} D & \text{if } j = 0 \\ CA^{j-1}B & \text{else} \end{cases}$$

Proof: Expand the recurrence, e.g. if $j = 1$

$$y_{t+1} = Cx_{t+1} + Du_{t+1} + z_{t+1}$$

$$= CAx_t + CBu_t + Cw_t + Du_{t+1} + z_{t+1}$$

$$\Rightarrow \mathbb{E}[y_{t+1}u_t^\top] = CB\mathbb{E}[u_tu_t^\top] = CB \quad \blacksquare$$

A MAJOR COMPLICATION

So why aren't we done?

A MAJOR COMPLICATION

So why aren't we done?

We could try to estimate $CA^{j-1}B$ using

$$\frac{1}{T}\sum_{t=1}^{T} y_{t+j}u_t^\top$$

A MAJOR COMPLICATION

So why aren't we done?

We could try to estimate $CA^{j-1}B$ using

$$\frac{1}{T}\sum_{t=1}^{T} y_{t+j} u_t^\top$$

But there is dependence across timesteps and this estimator can have **unbounded variance**

A MAJOR COMPLICATION

So why aren't we done?

We could try to estimate $CA^{j-1}B$ using

$$\frac{1}{T}\sum_{t=1}^{T} y_{t+j}u_t^\top$$

But there is dependence across timesteps and this estimator can have **unbounded variance**

Aside: This is why strict stability trivializes the problem: Otherwise just wait long enough to get almost independent samples

STABILIZING THE MOMENTS

Main Idea: Form a new time series

$$\widehat{y_t} \triangleq y_t - \sum_{j=1}^{n} c_j y_{t-j} \quad \text{such that...}$$

STABILIZING THE MOMENTS

Main Idea: Form a new time series

$$\widehat{y}_t \triangleq y_t - \sum_{j=1}^{n} c_j y_{t-j} \quad \text{such that}$$

(1) Expectation is **unchanged** i.e. $\mathbb{E}[\widehat{y}_{t+j} u_t^\top] = \mathbb{E}[y_{t+j} u_t^\top]$

STABILIZING THE MOMENTS

Main Idea: Form a new time series

$$\widehat{y}_t \triangleq y_t - \sum_{j=1}^{n} c_j y_{t-j} \quad \text{such that}$$

(1) Expectation is **unchanged** i.e. $\mathbb{E}[\widehat{y}_{t+j} u_t^\top] = \mathbb{E}[y_{t+j} u_t^\top]$

(2) Its variance is bounded, **independently of t**

First Attempt: Take the c_j's = coefficients of the characteristic poly

First Attempt: Take the c_j's = coefficients of the characteristic poly

Then the **Cayley-Hamilton Theorem** tells us

$$A^n - \sum_{j=1}^{n} c_j A^{n-j} = 0$$

First Attempt: Take the c_j's = coefficients of the characteristic poly

Then the **Cayley-Hamilton Theorem** tells us

$$A^n - \sum_{j=1}^{n} c_j A^{n-j} = 0$$

And can cancel all but the **transient terms** (proof by picture)

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$$\widehat{y}_{t+1} \quad = \quad y_{t+1} \quad -c_1 y_t \quad -c_2 y_{t-1} \quad \cdots$$

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$$\widehat{y}_{t+1} \;=\; Cx_{t+1} \qquad -c_1 Cx_t \qquad -c_2 Cx_{t-1} \ldots$$

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$$
\begin{array}{lll}
\widehat{y}_{t+1} = Cx_{t+1} & & \\
 & -c_1 C x_t & \\
 & & -c_2 C x_{t-1} \\
 & & \\
 & & \ddots
\end{array}
$$

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$$\widehat{y}_{t+1} = \boxed{Cx_{t+1}}$$

$$-c_1 C x_t$$

$$-c_2 C x_{t-1}$$

$$\ddots$$

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$\widehat{y}_{t+1} =$	CBu_t		
	CAx_t	$-c_1 C x_t$	
			$-c_2 C x_{t-1}$

$$\ddots$$

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$$
\begin{array}{rcc|c|c}
\widehat{y}_{t+1} & = & CBu_t & & \\
\hline
 & & \boxed{CAx_t} & \boxed{-c_1 C x_t} & \\
\hline
 & & & & -c_2 C x_{t-1} \\
\hline
 & & & & \\
 & & & & \ddots
\end{array}
$$

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$$
\begin{array}{rl|c|c}
\widehat{y}_{t+1} = & CBu_t & & \\
\hline
& CABu_{t-1} & -c_1 CBu_{t-1} & \\
\hline
& CA^2 x_{t-1} & -c_1 CAx_{t-1} & -c_2 Cx_{t-1} \\
\hline
& & & \\
\end{array}
$$

$$\ddots$$

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$\widehat{y}_{t+1} \;=\; CBu_t$		
$CABu_{t-1}$	$-c_1CBu_{t-1}$	
$\boxed{CA^2x_{t-1}}$	$\boxed{-c_1CAx_{t-1}}$	$\boxed{-c_2Cx_{t-1}}$

$$\ddots$$

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$$\widehat{y}_{t+1} = CBu_t$$

$CABu_{t-1}$	$-c_1 CBu_{t-1}$	
CA^2Bu_{t-2}	$-c_1 CABu_{t-2}$	$-c_2 CBu_{t-2}$
CA^3x_{t-2}	$-c_1 CA^2x_{t-2}$	$-c_2 CAx_{t-2}$

$\ddots$

For simplicity suppose $x_{t+1} = Ax_t + Bu_t$ and $y_t = Cx_t$

Then we have

$\widehat{y}_{t+1} = CBu_t$		
$CABu_{t-1}$	$-c_1CBu_{t-1}$	
CA^2Bu_{t-2}	$-c_1CABu_{t-2}$	$-c_2CBu_{t-2}$
CA^3x_{t-2}	$-c_1CA^2x_{t-2}$	$-c_2CAx_{t-2}$

$$\ddots$$

Eventually get cancellation!

First Attempt: Take the c_j's = coefficients of the characteristic poly

Then the **Cayley-Hamilton Theorem** tells us

$$A^n - \sum_{j=1}^{n} c_j A^{n-j} = 0$$

And can cancel all but the **transient terms** (proof by picture)

First Attempt: Take the c_j's = coefficients of the characteristic poly

Then the **Cayley-Hamilton Theorem** tells us

$$A^n - \sum_{j=1}^{n} c_j A^{n-j} = 0$$

And can cancel all but the **transient terms** (proof by picture)

$$\widehat{y}_t = \sum_{i=1}^{n} \left(CA^{i-1}B - \sum_{j=1}^{i-1} c_j CA^{i-j-1}B \right) u_{t-i}$$

$$+ \sum_{i=n+1}^{t} \left(CA^{i-n-1} \underbrace{\left(A^n - \sum_{j=1}^{n} c_j A^{n-j} \right)}_{\text{zero}} B \right) u_{t-i}$$

First Attempt: Take the c_j's = coefficients of the characteristic poly

Then the **Cayley-Hamilton Theorem** tells us

$$A^n - \sum_{j=1}^{n} c_j A^{n-j} = 0$$

And can cancel all but the **transient terms** (proof by picture)

$$\widehat{y}_t = \sum_{i=1}^{n} \Big(CA^{i-1}B - \sum_{j=1}^{i-1} c_j CA^{i-j-1}B \Big) u_{t-i}$$

First Attempt: Take the c_j's = coefficients of the characteristic poly

Then the **Cayley-Hamilton Theorem** tells us

$$A^n - \sum_{j=1}^{n} c_j A^{n-j} = 0$$

And can cancel all but the **transient terms** (proof by picture)

$$\widehat{y}_t = \sum_{i=1}^{n} \Big(C A^{i-1} B - \sum_{j=1}^{i-1} c_j C A^{i-j-1} B \Big) u_{t-i}$$

Thus the variance is bounded independently of t

First Attempt: Take the c_j's = coefficients of the characteristic poly

And so direct computation shows the new time series satisfies

✓ (2) Its variance is bounded, **independently of t**

First Attempt: Take the c_j's = coefficients of the characteristic poly

And so direct computation shows the new time series satisfies

✓ (2) Its variance is bounded, **independently of t**

Unfortunately

✗ (1) Expectation is **unchanged** i.e. $\mathbb{E}[\widehat{y}_{t+j} u_t^\top] = \mathbb{E}[y_{t+j} u_t^\top]$

First Attempt: Take the c_j's = coefficients of the characteristic poly

And so direct computation shows the new time series satisfies

(2) Its variance is bounded, **independently of t**

Unfortunately

(1) Expectation is **unchanged** i.e. $\mathbb{E}[\widehat{y}_{t+j} u_t^\top] = \mathbb{E}[y_{t+j} u_t^\top]$

because we pick up extra terms

$$\widehat{y}_t = \sum_{i=1}^{n} \Big(CA^{i-1}B - \sum_{j=1}^{i-1} c_j CA^{i-j-1}B \Big) u_{t-i}$$

Second Attempt: Same strategy, **but using lag**

$$\widehat{y}_t = y_t - \sum_{j=1}^{n} \alpha_j y_{t-j-r}$$

Second Attempt: Same strategy, **but using lag**

$$\widehat{y}_t = y_t - \underbrace{\sum_{j=1}^{n} \alpha_j y_{t-j-r}}$$

Claim: If $r > k$ there is no u_{t-1}, u_{t-2} … $, u_{t-k}$

Second Attempt: Same strategy, **but using lag**

$$\underbrace{\widehat{y}_t = y_t - \sum_{j=1}^{n} \alpha_j y_{t-j-r}}$$

Claim: If $r > k$ there is no u_{t-1}, u_{t-2} … , u_{t-k}

Hence we now have

(1) Expectation is **unchanged** i.e. $\mathbb{E}[\widehat{y}_{t+j} u_t^\top] = \mathbb{E}[y_{t+j} u_t^\top]$

Second Attempt: Same strategy, **but using lag**

$$\widehat{y}_t = y_t - \sum_{j=1}^{n} \alpha_j y_{t-j-r}$$

Claim: If $r > k$ there is no $u_{t-1}, u_{t-2} \dots, u_{t-k}$

Hence we now have

(1) Expectation is **unchanged** i.e. $\mathbb{E}[\widehat{y}_{t+j} u_t^\top] = \mathbb{E}[y_{t+j} u_t^\top]$

But we still cancel out long-range dependencies, so the variance stays bounded

Third Attempt: …

Third Attempt: …

(I lied)

Third Attempt: …

(I lied)

Problem: The coefficients of the characteristic poly can be **exponentially large**

Third Attempt: …

(I lied)

Problem: The coefficients of the characteristic poly can be **exponentially large**

Proposition [informal]: Can show good, bounded c_j's exist by appealing to condition number bds on O_s/Q_s instead

Third Attempt: …

(I lied)

Problem: The coefficients of the characteristic poly can be **exponentially large**

Proposition [informal]: Can show good, bounded c_j's exist by appealing to condition number bds on O_s/Q_s instead

If we already knew O_s/Q_s $\Rightarrow$ Can find good c_j's

Third Attempt: …

(I lied)

Problem: The coefficients of the characteristic poly can be **exponentially large**

Proposition [informal]: Can show good, bounded c_j's exist by appealing to condition number bds on O_s/Q_s instead

If we already knew O_s/Q_s $\Rightarrow$ Can find good c_j's

Isn't this all circular?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- **Convex Programming to the Rescue**

Epilogue: Is This Just the Start?

A CONVEX PROGRAM

Can write a convex program to search for the c_j's, **now that we know they exist**

A CONVEX PROGRAM

Can write a convex program to search for the c_j's, **now that we know they exist**

Find $(c_1, c_2, \ldots, c_s)$

such that $|c_j| \leq \varepsilon_1$ for all j

and $\left\| y_t - \sum_{k=1}^{s} c_j y_{i-j-r} \right\|^2 \leq \varepsilon_2$ for all t

A CONVEX PROGRAM

Can write a convex program to search for the c_j's, **now that we know they exist**

Find $(c_1, c_2, \ldots, c_s)$

such that $|c_j| \leq \varepsilon_1$ for all j

and $\left\| y_t - \sum_{k=1}^{s} c_j y_{i-j-r} \right\|^2 \leq \varepsilon_2$ for all t

(1) Define a function Φ_c that captures the **potential variance**

A CONVEX PROGRAM

Can write a convex program to search for the c_j's, **now that we know they exist**

Find $(c_1, c_2, \ldots, c_s)$

such that $|c_j| \leq \varepsilon_1$ for all j

and $\left\| y_t - \sum_{k=1}^{s} c_j y_{i-j-r} \right\|^2 \leq \varepsilon_2$ for all t

(1) Define a function Φ_c that captures the **potential variance**

(2) If it's large, whp a constraint is violated via **anticoncentration**

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

OUTLINE

Part I: Introduction

- Linear Dynamical Systems and Applications
- Main Problem
- Well-Posedness and Our Results

Part II: A Method of Moments Approach

- The Ho-Kalman Algorithm
- Controlling the Variance?
- Convex Programming to the Rescue

Epilogue: Is This Just the Start?

LOOKING FORWARD

The method of moments saved the day (again)

LOOKING FORWARD

The method of moments saved the day (again)

More ambitiously, we can ask:

> **Is there a dictionary mapping algorithmic tools in unsupervised learning to their dynamical counterparts?**

FURTHER DIRECTIONS

What if, at some time, we switch between different systems?

e.g. different variants of COVID

FURTHER DIRECTIONS

What if, at some time, we switch between different systems?

e.g. different variants of COVID

Generalization of the classic **change point detection problem**

FURTHER DIRECTIONS

What if, at some time, we switch between different systems?

e.g. different variants of COVID

Generalization of the classic **change point detection problem**

What about heterogeneity?

FURTHER DIRECTIONS

What if, at some time, we switch between different systems?

e.g. different variants of COVID

Generalization of the classic **change point detection problem**

What about heterogeneity?

Can learn mixture models even when the trajectories cannot be clustered, via **tensor methods**

FURTHER DIRECTIONS

What if, at some time, we switch between different systems?

e.g. different variants of COVID

Generalization of the classic **change point detection problem**

What about heterogeneity?

Can learn mixture models even when the trajectories cannot be clustered, via **tensor methods**

There is even more to say about reinforcement learning, but that is another topic for another time…

Summary:

- Linear dynamical systems have wide-ranging applications, but how do we learn them?
- New algorithm via the method of moments with **essentially minimal assumptions**
- **Is there a dictionary for mapping tools from unsupervised learning to their dynamical counterpart?**

Summary:

- Linear dynamical systems have wide-ranging applications, but how do we learn them?
- New algorithm via the method of moments with **essentially minimal assumptions**
- **Is there a dictionary for mapping tools from unsupervised learning to their RL/dynamical counterpart?**

Thanks! Any Questions?