A Systems Theoretic Perspective on Transfer Learning

Tyler Cody
Stephen Adams
Peter Beling

University of Virginia

High Level Motivation

Observation

Machine learning formulations classify methods and literature, but lack top-down design principles and consideration of systems-level interactions.

Idea

As machine learning techniques mature, systems theoretic frameworks ought to be developed to guide their design and implementation into real-world systems.

Actuator Health Monitoring (*Running Example*)

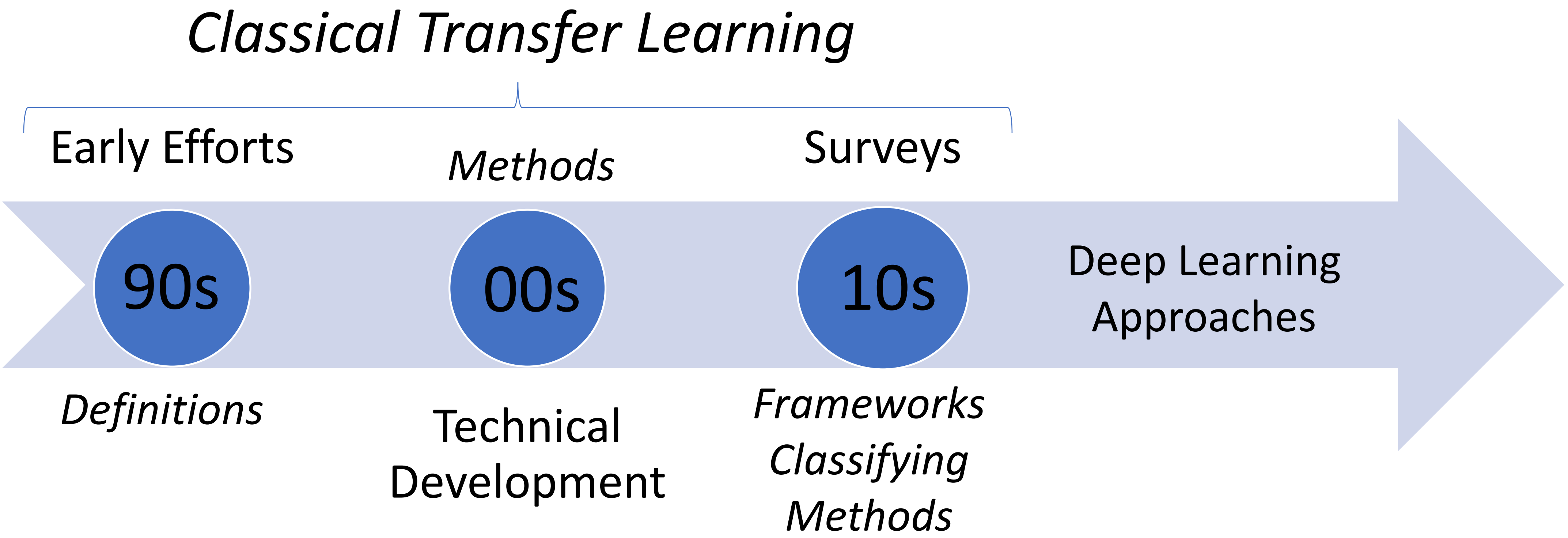
- Learning algorithms are commonly used to predict current and future health states of actuators
- Similar underlying physics, however physical and functional differences exist between actuators and over time

How do we transfer knowledge between actuators to make learning easier/feasible while accounting for individual differences?

Transfer Learning (TL)

“the ability of a system to recognize and apply knowledge and skills learned in previous tasks to novel tasks”

- DARPA BAA 05-29



Machine Learning Formulation of TL

Dichotomizes supervised learning problems into their **domain** $\mathcal{D}$ and **task** $\mathcal{T}$

Notations

Domain $\mathcal{D}$

1. Input space $\mathcal{X}$
2. Marginal distribution $P(X)$, where $X \in \mathcal{X}$

Task $\mathcal{T}$ (Given $\mathcal{D} = \{\mathcal{X}, P(X)\}$)

1. Output space $\mathcal{Y}$
2. Learn a $\phi: X \rightarrow Y$ to approach the underlying $P(Y|X)$, where $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$

Supervised Learning

Classification, Regression

Machine Learning Formulation of TL

Definition

Given a source domain $\mathcal{D}_S$ and a learning task $\mathcal{T}_S$, and a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$, **transfer learning aims to help improve the learning of the target predictive function $\boldsymbol{\phi_T}$ using knowledge in $\boldsymbol{\mathcal{D}_S}$ and $\boldsymbol{\mathcal{T}_S}$** where,

$$\mathcal{D}_S \neq \mathcal{D}_T \text{ (either } \mathcal{X}_S \neq \mathcal{X}_T \text{ or } P(X_S) \neq P(X_T)),$$

or, $$\mathcal{T}_S \neq \mathcal{T}_T \text{ (either } \mathcal{Y}_S \neq \mathcal{Y}_T \text{ or } P(Y_S|X_S) \neq P(Y_T|X_T))$$

S.J. Pan and Q. Yang, "A survey on transfer learning." IEEE Transactions on Knowledge and Data Engineering, 2010.

What is Systems Theory?

A system is a relation on sets,

$$S \subset \times \{V_i : i \in I\}$$

The components of S, V_i are termed the systems objects, and we are primarily concerned with input-output systems,

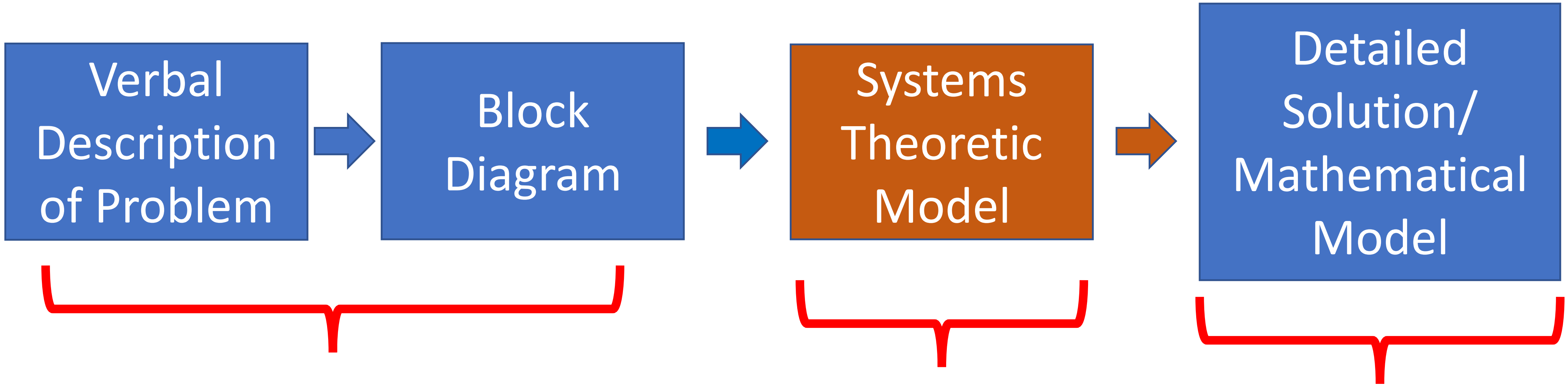
$$S \subset X \times Y$$

Further development of theory introduces additional structure to elements of the systems objects $v \in V_i$ or in the systems objects V_i themselves.

General Systems Theory, Mesarovic & Takahara 1975

Abstract Systems Theory, Mesarovic & Takahara 1989

Between Block Diagrams and Detailed Mathematical Models

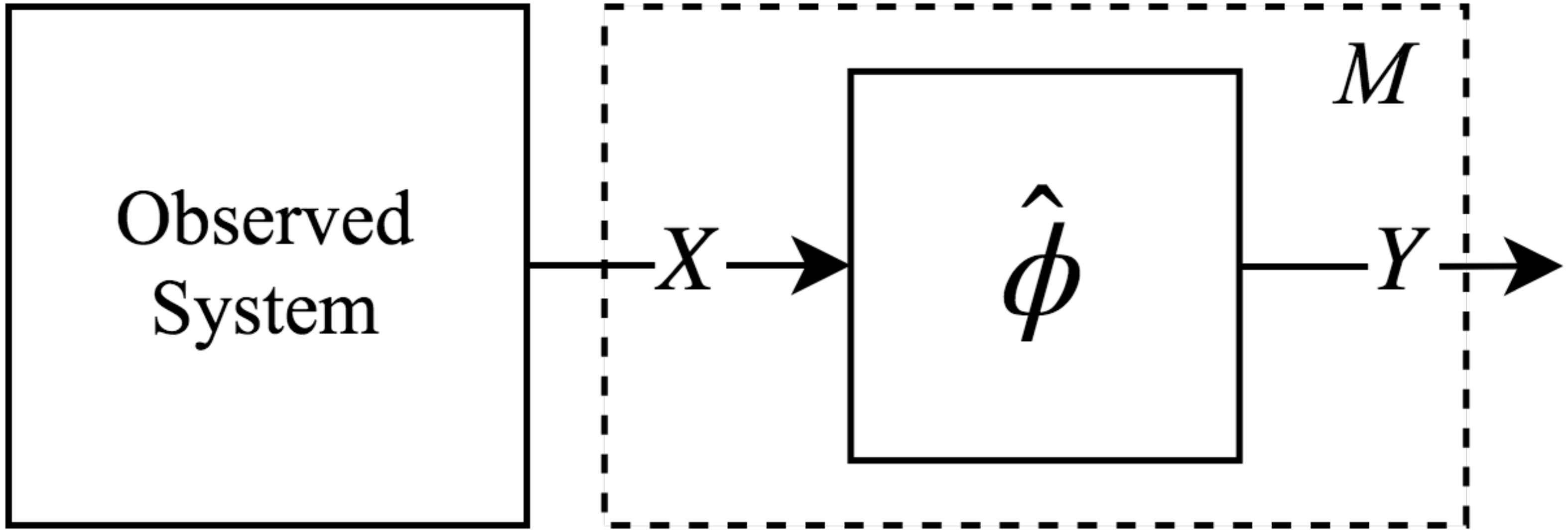


What is the learning problem, and where does it reside within the context of our system?

How do we represent it in explicit, mathematical terms?

How do we algorithmically solve the problem?

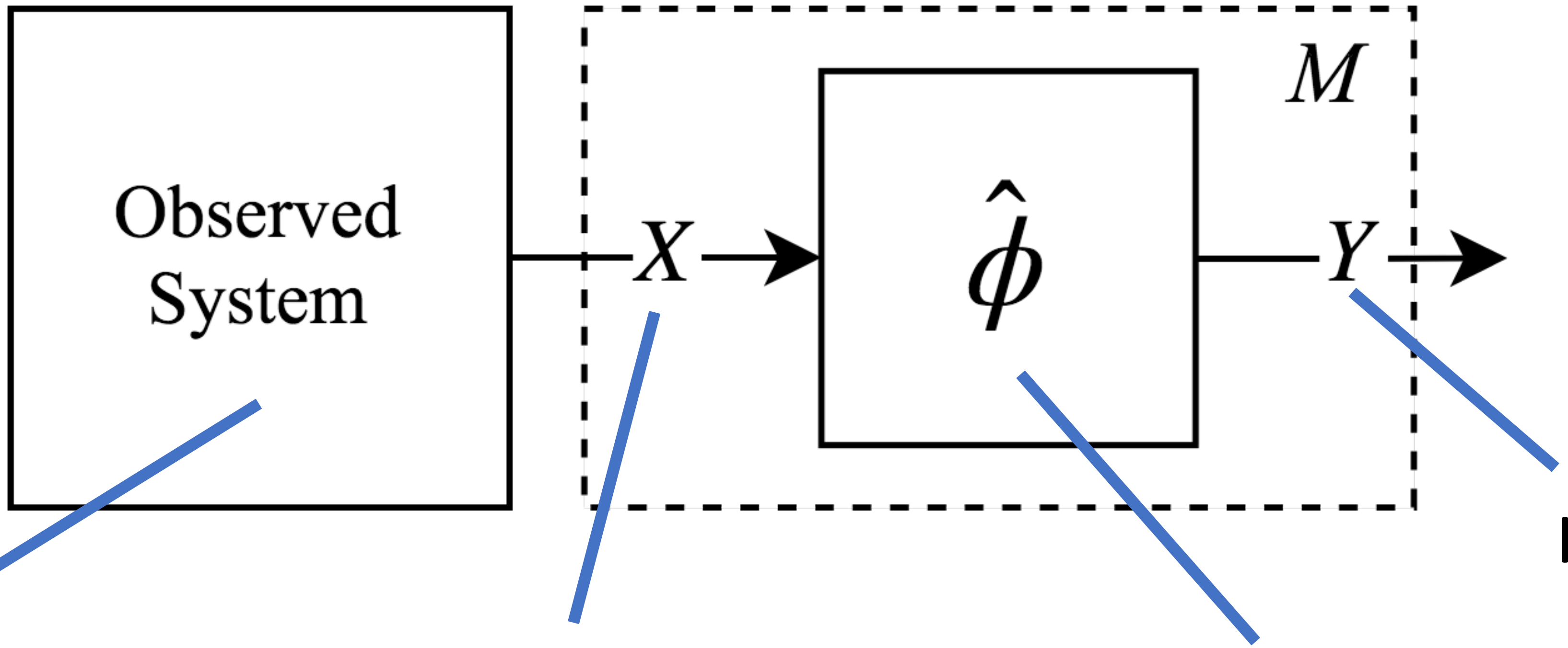
Supervised learning as Input-Output System



System structure is given by the input-output space $\{\mathcal{X}, \mathcal{Y}\}$

System dynamics are given by the joint distribution $P(X, Y)$

Actuator Health Monitoring Algorithm



Health State

Actuator

Sensor data output by actuator system

Algorithm mapping sensor data to health state

Systems Theoretic Formulation of TL

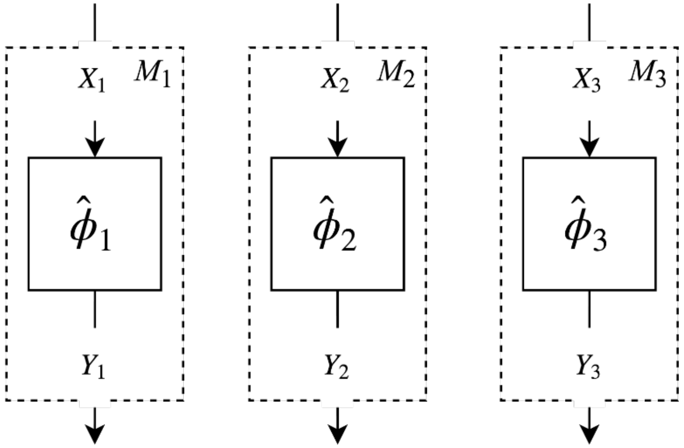
Definition

Given a source inference system $M_S = \{\mathcal{X}_S, \mathcal{Y}_S, \hat{\phi}_S\}$ and a target inference system $M_T = \{\mathcal{X}_T, \mathcal{Y}_T, \hat{\phi}_T\}$, transfer learning tries to use knowledge from M_S to improve the learning of $\hat{\phi}_T$, where $\mathcal{X}_S \times \mathcal{Y}_S \neq \mathcal{X}_T \times \mathcal{Y}_T$ or $P(X_S, Y_S) \neq P(X_T, Y_T)$.

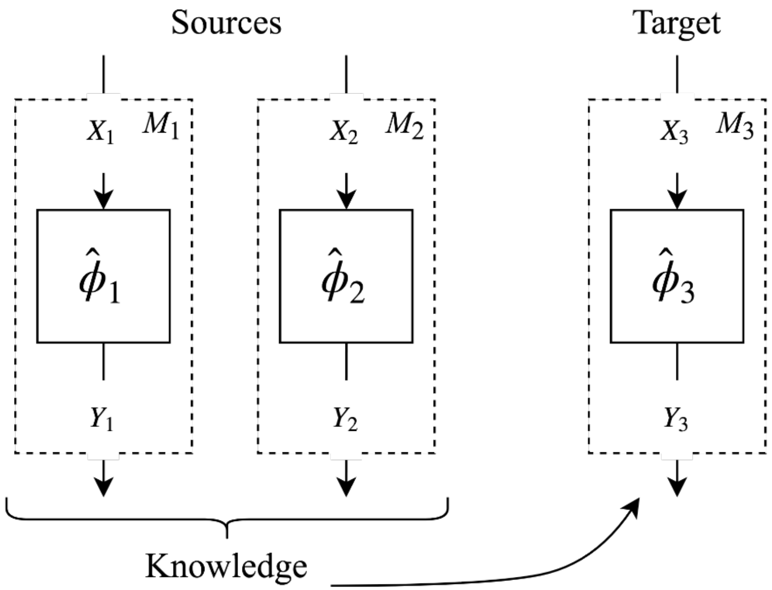
Note, the machine learning and systems theoretic formulations are different in that:

1. We consider inputs $\mathcal{X}$ to come from a **coupled observed system**, and
2. We breakdown the differences using **system structure $\boldsymbol{\mathcal{X}} \times \boldsymbol{\mathcal{Y}}$ and system dynamics $\boldsymbol{P}(\boldsymbol{X}, \boldsymbol{Y})$** instead of using domain $\mathcal{D}$ and task $\mathcal{T}$

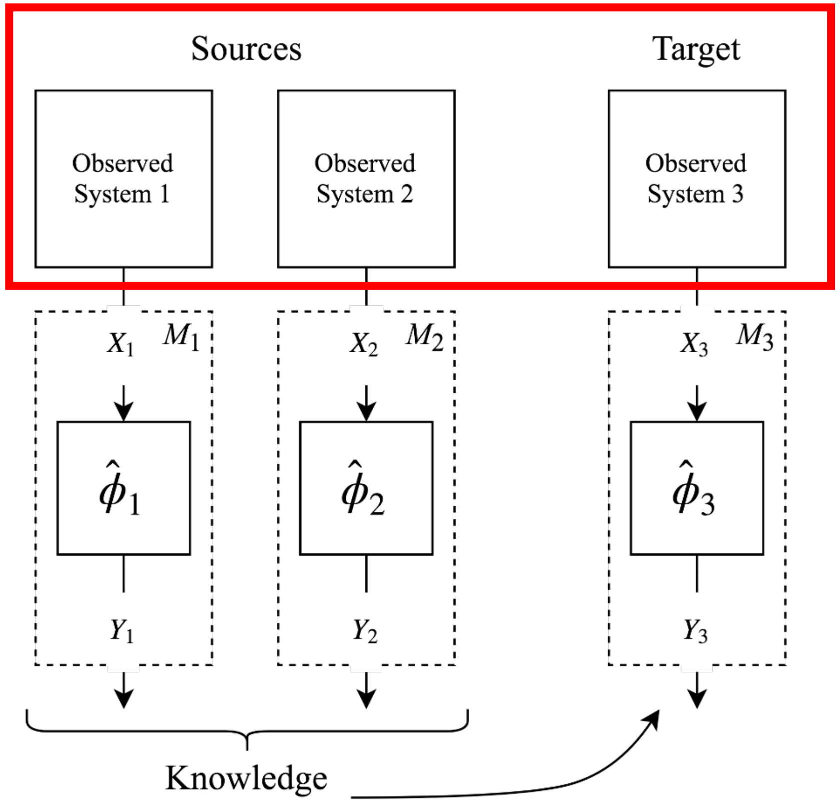
Comparing Formulations of TL



Traditional
Machine Learning

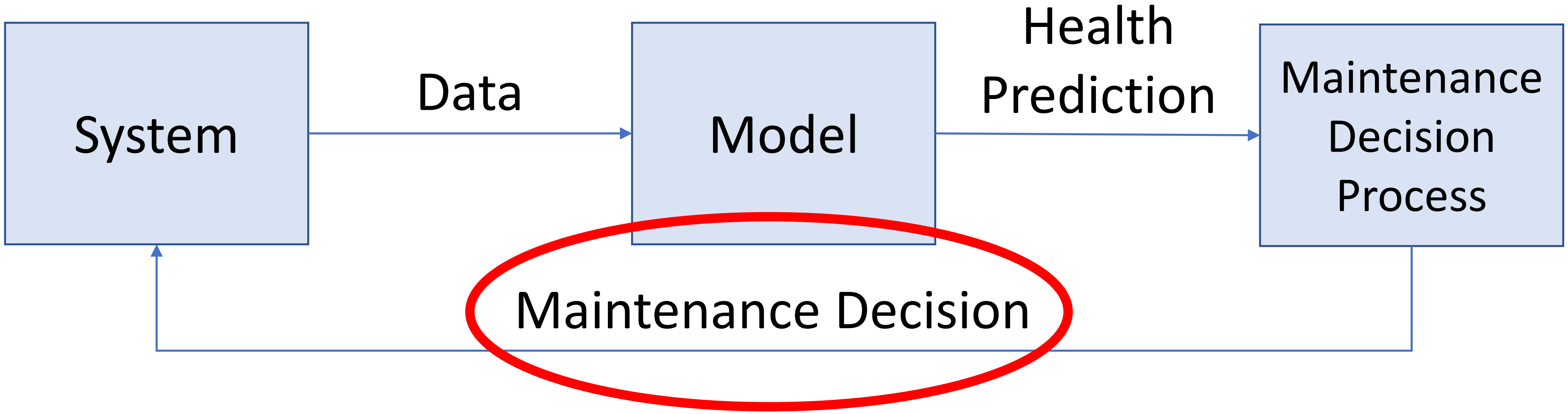


Transfer Learning
(Machine Learning)



Transfer Learning
(**Systems Theory**)

Maintenance Changes System Behavior



The maintenance decision changes the system.

How can we update our model to account for these changes?

System Rebuild Causes Model Failure

- Need for transfer learning…
  - Original model performance: 0.997
  - Performance on post-rebuild system: 0.775

- How can we use knowledge of the rebuild process to update the model?

Extending Classical Transfer Learning

Classical Transfer Learning

- *Source*: X^S space, Y^S space, joint sample
- *Target*: X^T space, Y^T space, joint sample

Model-Based Transfer Learning

- *Source*: X^S space, Y^S space, joint sample, $E[P(X^S, Y^S)]$
- *Target*: X^T space, Y^T space, joint sample, $E[P(X^T, Y^T)]$

Actuator Transfer Learning Setting

Classical Transfer Learning

- *Source*: X^S space, Y^S space, joint sample
- *Target*: X^T space, Y^T space, joint sample

Model-Based Transfer Learning

- *Source*: X^S space, Y^S space, joint sample
- *Target*: X^T space, Y^T space, joint sample, $E[P(X^T, Y^T)]$

Model gives a general idea of where in the $X \times Y$ space the rebuilt actuator will be operating

Actuator Transfer Learning Setting

The Y spaces are binary healthy/damage indicators. There is no damage data available in the target, post-rebuild actuator.

Tested on predicting healthy and damaged classes on post-rebuild actuator.

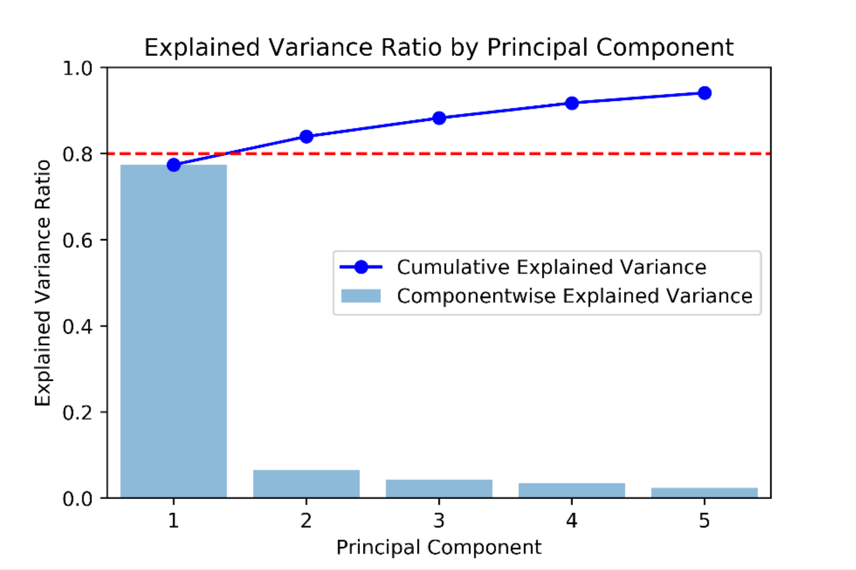
Approaches:

- Subspace Sample Transfer
  - Transfer samples of the source damage class to the target
- Model-Based Subspace Sample Transfer
  - Transfer sample of the source damage class AND sample drawn from a model $P(X_T | Y_T = damage)$ to the target

Fitting the a Model for Post-Rebuild Damage Behavior

Post-Rebuild Data
$\{(x_T, y_T)\}$

PCA Transformation



Post-Rebuild Model
$P(X_T|Y_T = Damage)$

Model-Based Transfer Learning Results

Training Sample		Accuracy (vs. 77%)
Source Damage Data Target Healthy Data	$f^T: X^T \rightarrow Y^T$	85.4%
Source Damage Data Target Healthy Data $P(X_T \mid Y_T = damage)$	$f^T: X^T \rightarrow Y^T$	87.2%
Target Healthy Data $P(X_T \mid Y_T = damage)$	$f^T: X^T \rightarrow Y^T$	88.4%

How do we characterize distributional changes from data?

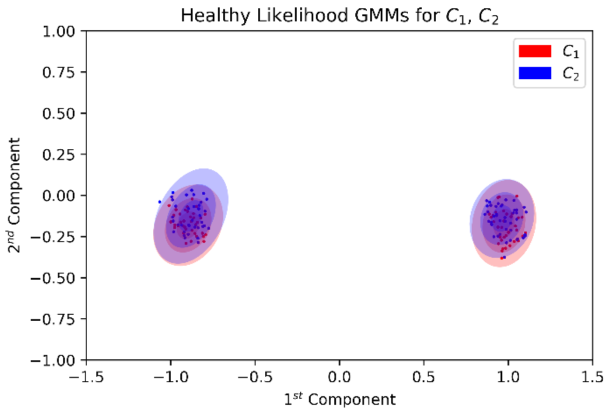
- Use metrics to measure differences in probability distributions
  - e.g. Hellinger Distance, KL-Divergence

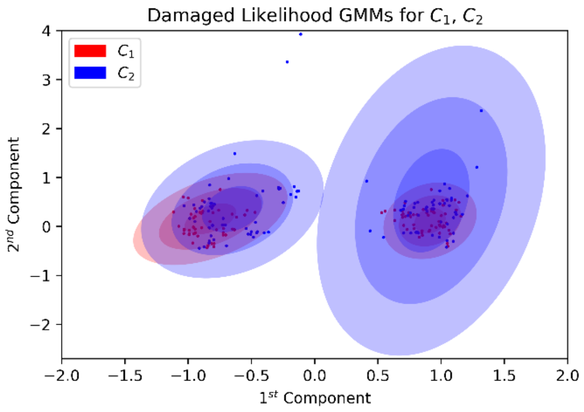
- We can characterize the likelihoods P$(X|Y)$, marginals P(X), and posteriors P$(Y|X)$ of Bayes Theorem:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

How do we characterize distributional changes from data?

Likelihoods

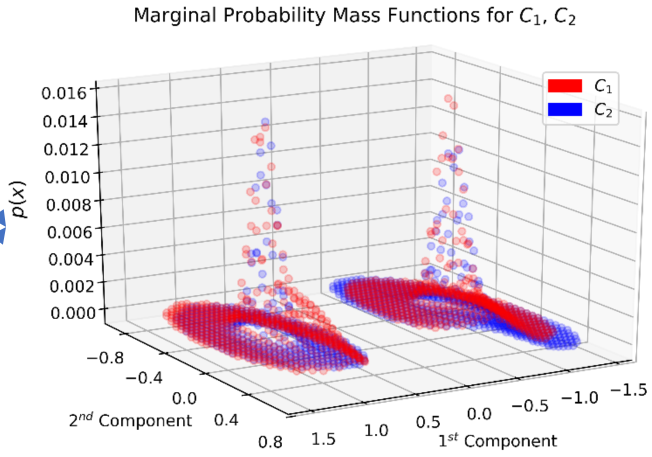




Δ Healthy Likelihood: 0.23

Δ Damage Likelihood: 0.55

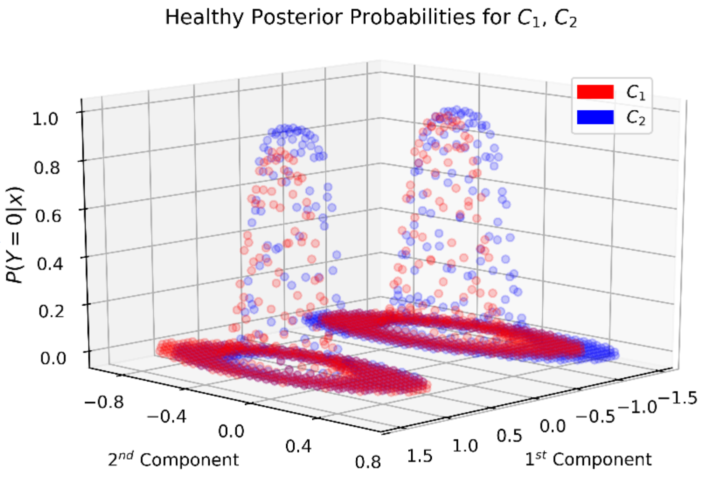
Marginals



Δ Marginal: 0.41

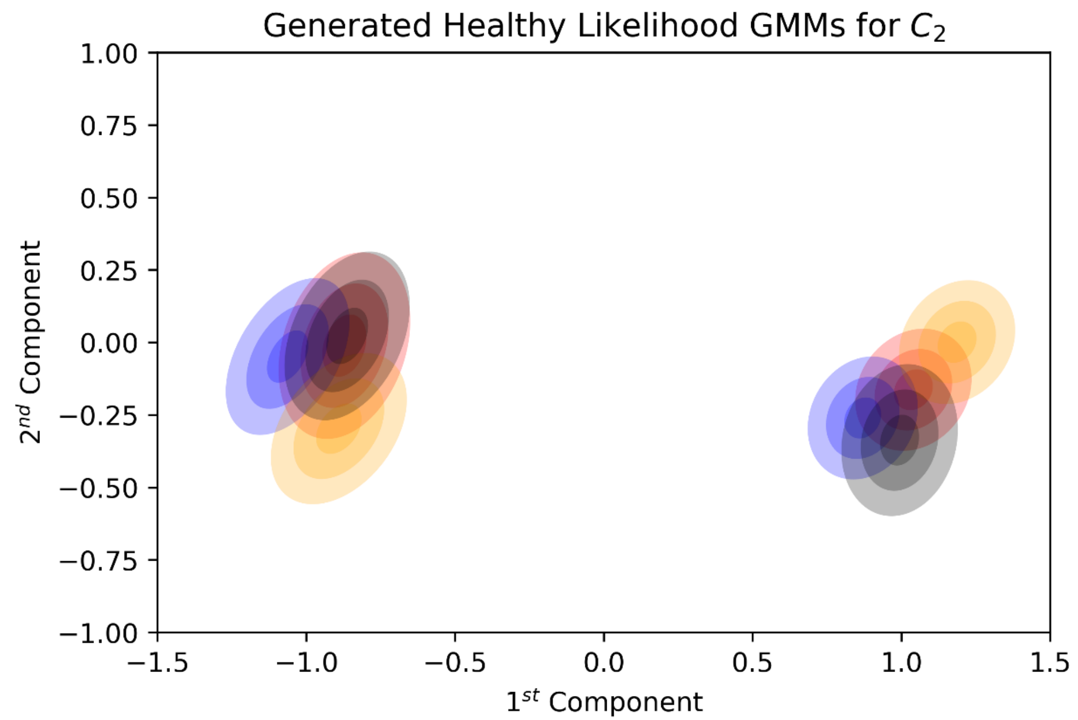
Δ Posterior: 0.27

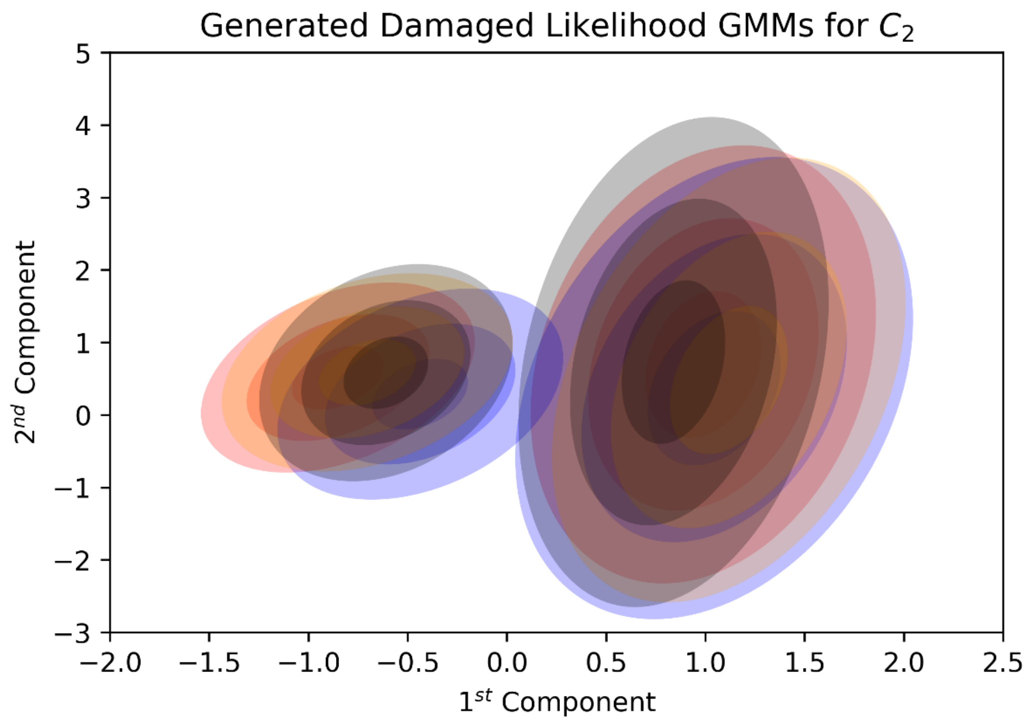
Posteriors



Characteristics

Family of Models





while(generating):

generate new models
if characterization matches, then save models

Conclusions

- In our systems theoretic formulation of transfer learning, algorithm design is secondary to system design
- The key design parameters for transfer learning are:

1. the instrumentation of the observed systems $\mathcal{X}_S, \mathcal{X}_T$
2. the output of the inference systems $\mathcal{Y}_S, \mathcal{Y}_T$
3. the complexity of and variability between $P(X_S, Y_S)$ and $P(X_T, Y_T)$

Transfer learning system design proceeds by analyzing the trade-offs of these design parameters under the goals, metrics, and requirements of a particular system.

Future Work

- Formalize the definition of transfer learning systems, the complexity of and variability between inference systems, and the usefulness of system structure
- Extend framework to explicitly consider multiple source systems
- Study fundamental concepts from systems theory such as coupling and subsystems in the context of transfer learning
- Real world case studies of the design methodology

Thank You, Questions?