Metric Learning for Simulation Analytics

Graham Laidler
Lucy E. Morgan
Barry L. Nelson
Nicos G. Pavlidis

STOR-i Centre for Doctoral Training
Lancaster University, UK

Simulation Analytics

Stochastic simulation generates a dynamic **sample path** - a time-stamped trace of every event and state change.

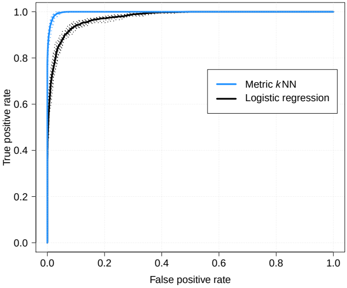
Opportunity: We can store and analyze the sample path from every replication, and use this data to answer **deeper** questions, such as:

- Which events and components really drive the system's performance?
- How sensitive is the stochastic performance to the random input behavior?
- **Why** does one system design perform better than another?

Predicting dynamic performance

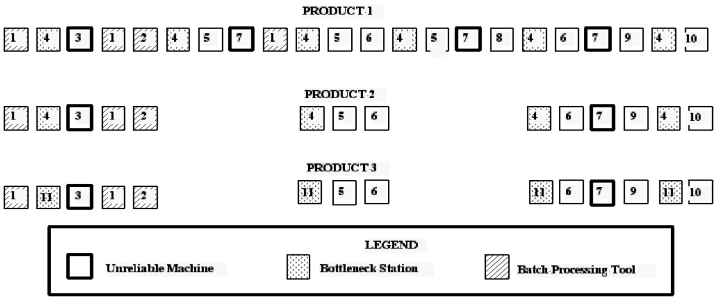
Aim: Predict a system's dynamic performance based on components of the system state

Difficulty: Simulation is a **complex dynamic process** which is hard to model with a parametric function.



Idea: ***k*NN** on the system state

A waferfab model: Due-date prediction



Waferfab model, taken from Kayton, Teyner, Schwartz and Uzsoy.

Aim: Predict whether a cassette will finish on time or late.

Problem: System state is large and heterogeneous. What are **neighbors**? When are two states **similar**?

What is similar? - Metric Learning

System state $\boldsymbol{x} = [x_1, x_2, \ldots, x_d]^\top$ contains d real-valued state variables.

We can learn a **Mahalanobis distance metric** of the form

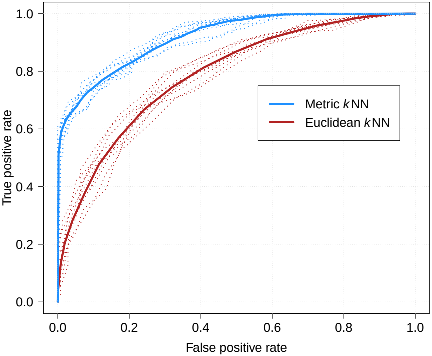
$$d_M(\boldsymbol{x}_i, \boldsymbol{x}_j) = [(\boldsymbol{x}_i - \boldsymbol{x}_j)^\top M(\boldsymbol{x}_i - \boldsymbol{x}_j)]^{1/2},$$

by **optimizing** some criteria over our data points:

$$\begin{aligned} \min_{M \in \mathbb{S}^d_+} & \sum_{(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathcal{S}} d_M^2(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ \text{s.t.} & \sum_{(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathcal{D}} d_M(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq \gamma. \end{aligned}$$

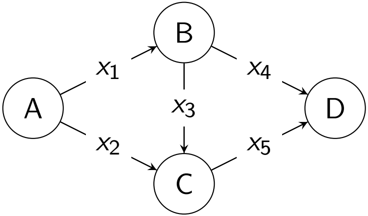
Results

The learned matrix M

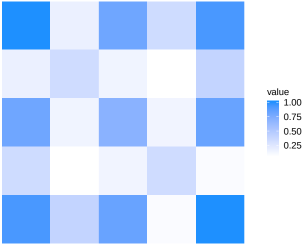


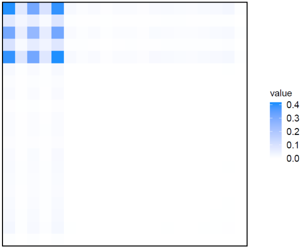
Cross-validated ROC curves

A stochastic activity network



$$T = \max\{x_1 + x_4,\; x_1 + x_3 + x_5,\; x_2 + x_5\}$$





Summary

Applying kNN and metric learning on sample path data allows:

1. Real-time **predictions** of a dynamic performance measure.
2. **Interpretation** as to which components of the system state drive the dynamic performance.

Future direction:

- Tailoring the metric learning to the simulation context