Predicting Delays in Queues with Invisible Customers

Speaker: Arik Senderovich (Faculty of Information, University of Toronto)

Data Analytics in Healthcare – 4th Annual Research Roundtable
3/23/2021

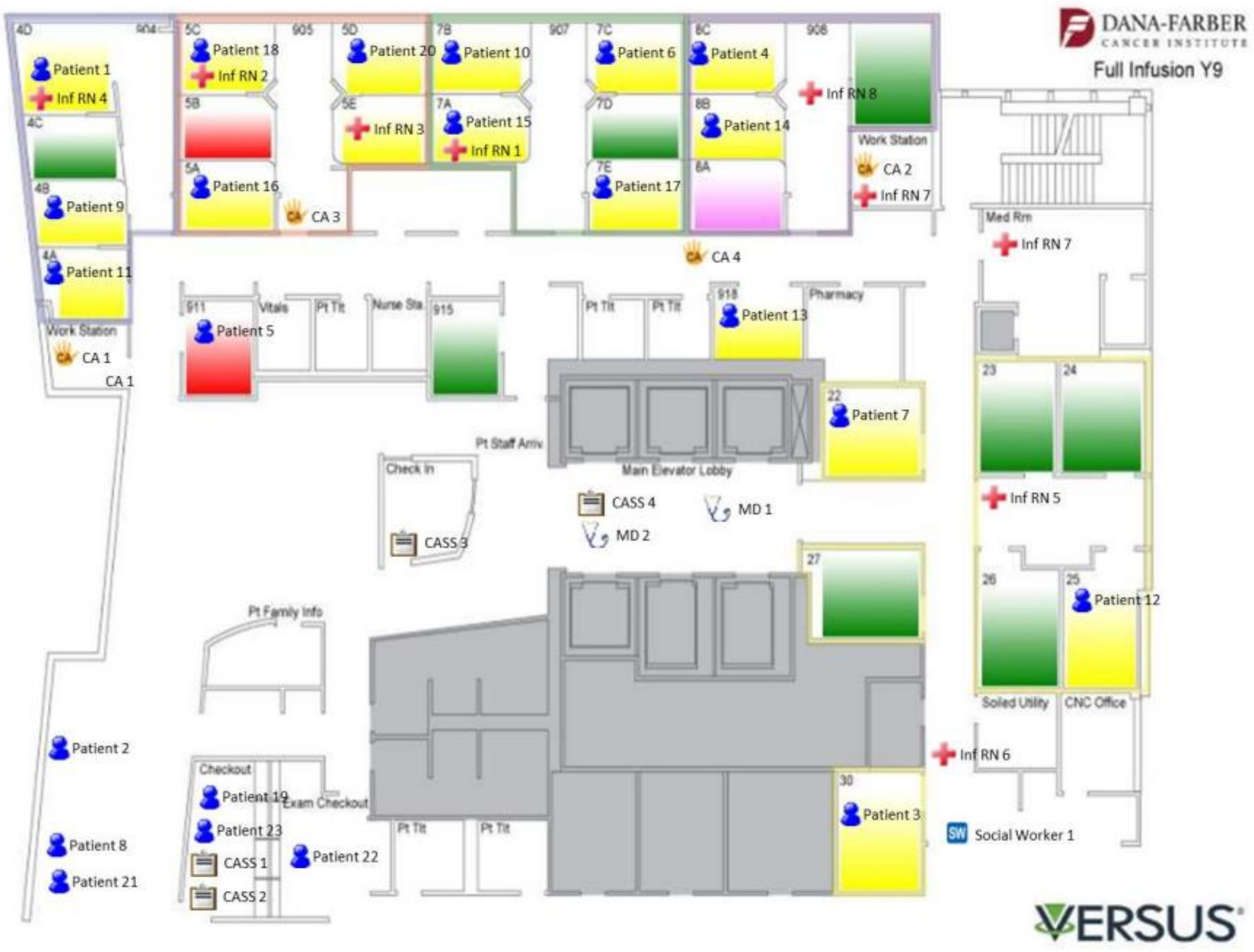
Joint work with:
Yoav Kerner (Ben-Gurion University)
Ricky Roet-Green, Yaron Shaposhnik, Yuting Yuan (University of Rochester)

Motivation: Dana-Farber Cancer Institute

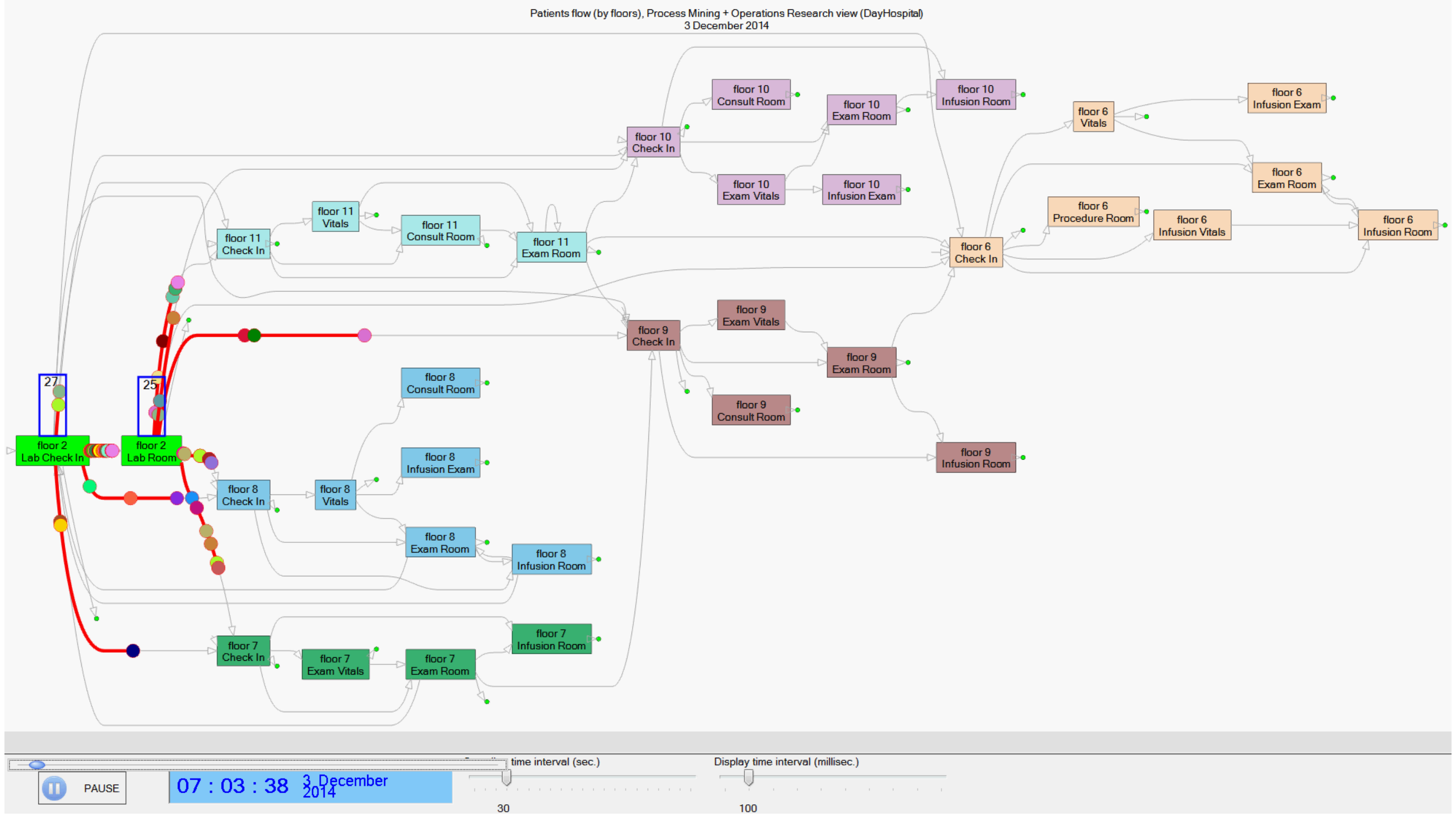
Dana-Farber
Cancer Institute

- 1000 patients / day
- 250 health providers
- 70 administrative staff
- On 7 medical floors
- All tracked via RTLS

Live Monitoring of Patients at DFCI

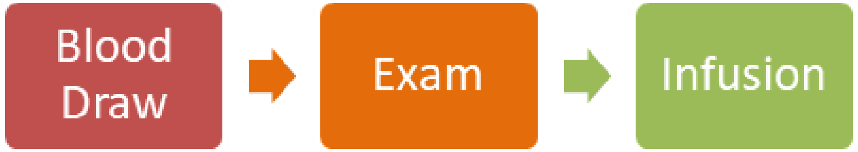


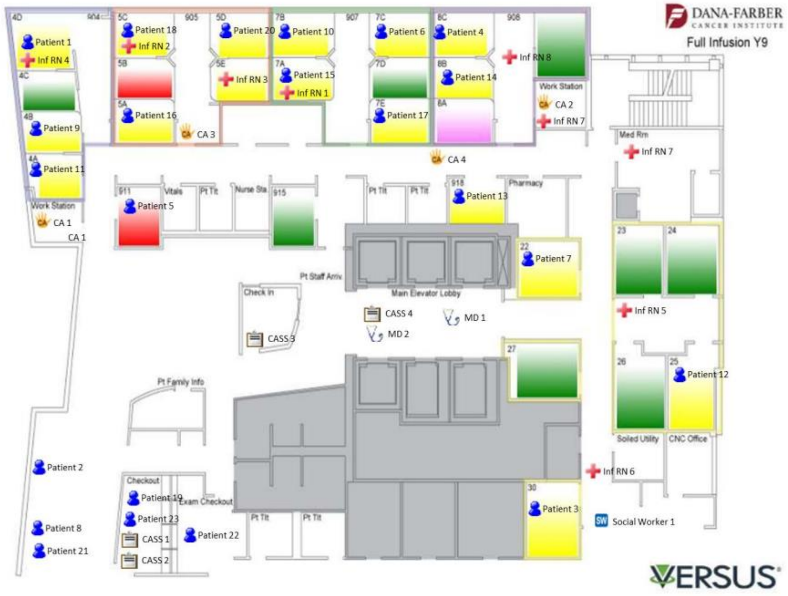
Motivation: Patient Flow



Delay Prediction at Dana-Farber

- ➢ Accurate delay prediction is important:
    - o Informing patients and families
    - o Quality of care: apologies and compensation
    - o Planning the next step in the process





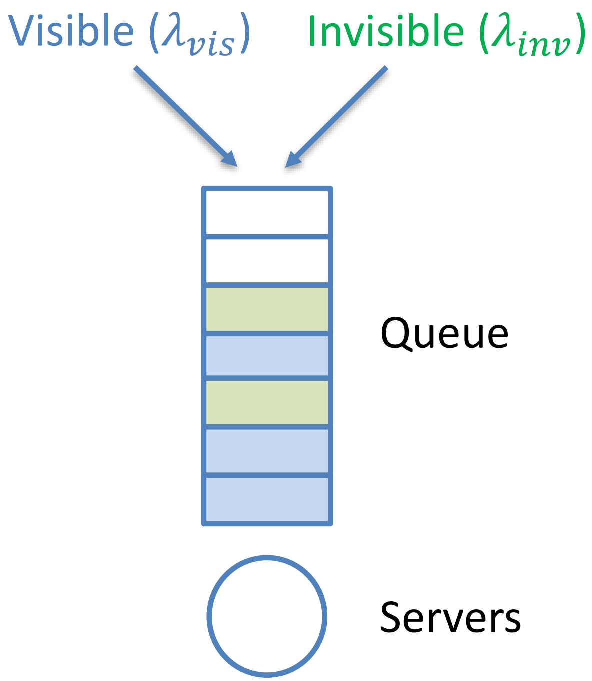
- ➢ Around 25% of patients are not being tracked
- ➢ Current prediction module uses only observed information

Research Question and Applications

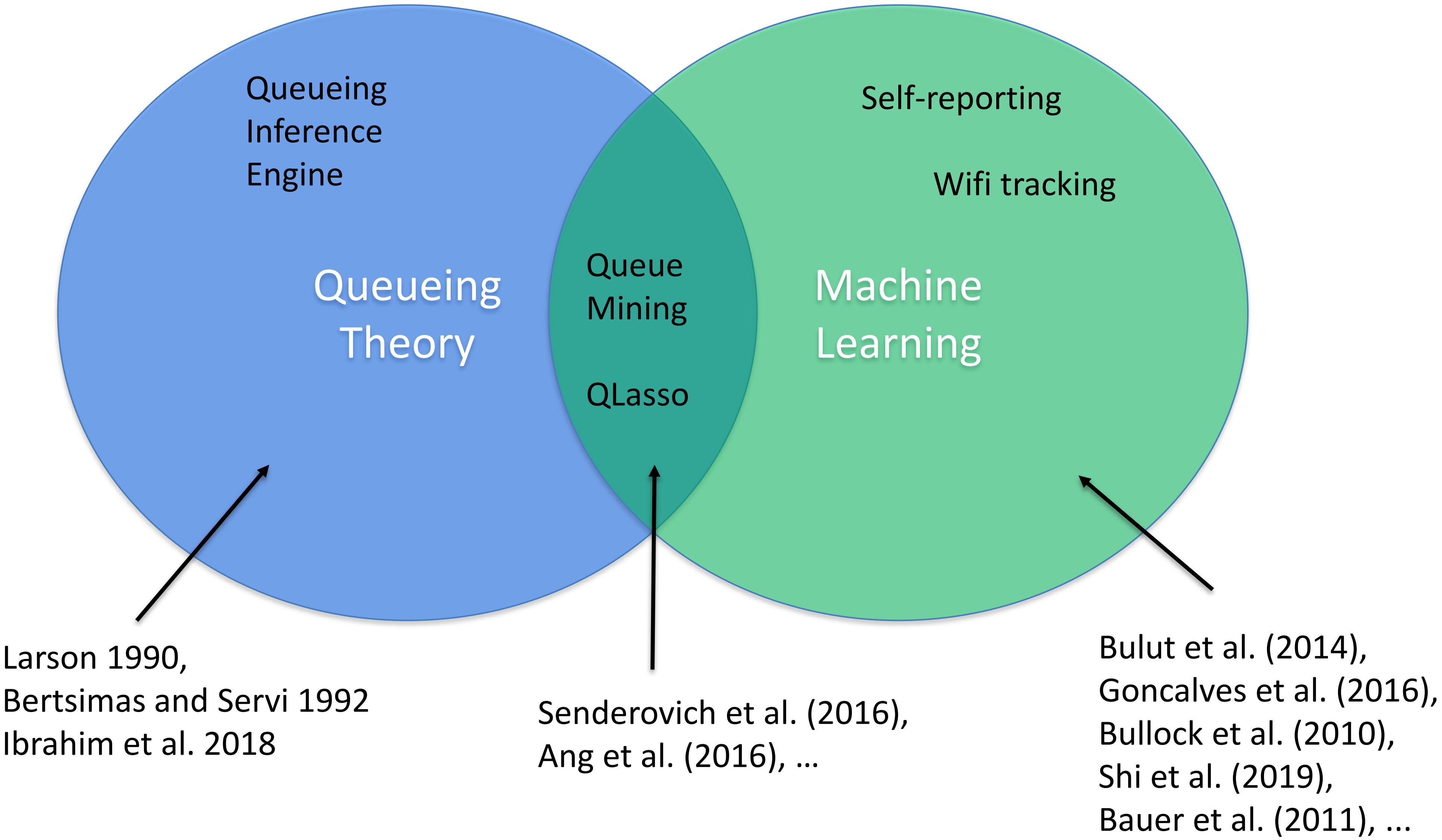
- ➢ Research question: how to predict waiting times when some of the customers in queue are invisible (to the system)?
- ➢ Additional applications:
  - Travel time prediction
  - Hybrid lines (app + in-person)

The Problem of Delay Prediction

- Predict the waiting time of an arriving visible patient given the **observed** system state
- We assume to know:
  - Inter-arrival time distribution (for overall population)
  - Service times distribution
  - Number of servers
  - Proportion of invisible (independent of the above)

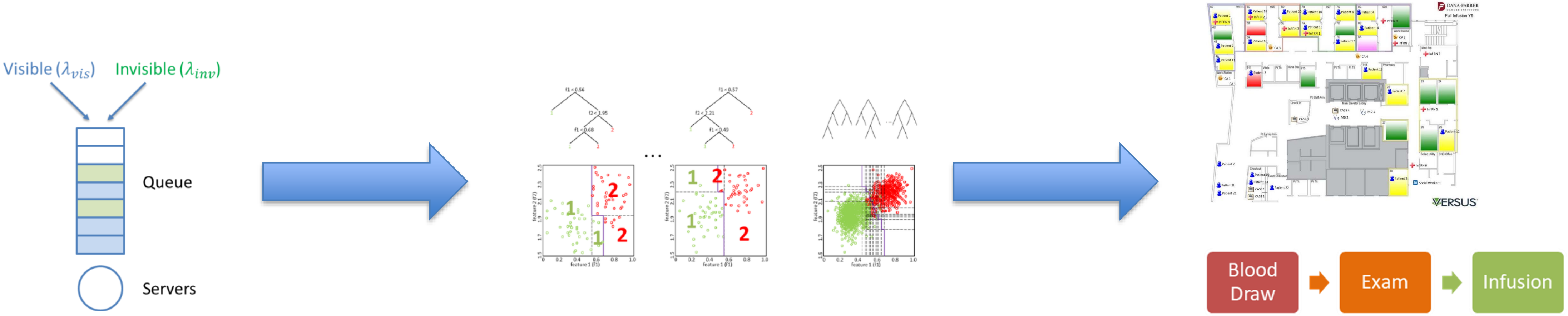


Related work



Our Approach & Outline of Talk

1. Construct exact queue-length predictor for a simple queueing model (M/M/1)
2. Gain insights by combining ML and analytical results from M/M/1
3. Extend the model to intractable queueing systems and validate insights

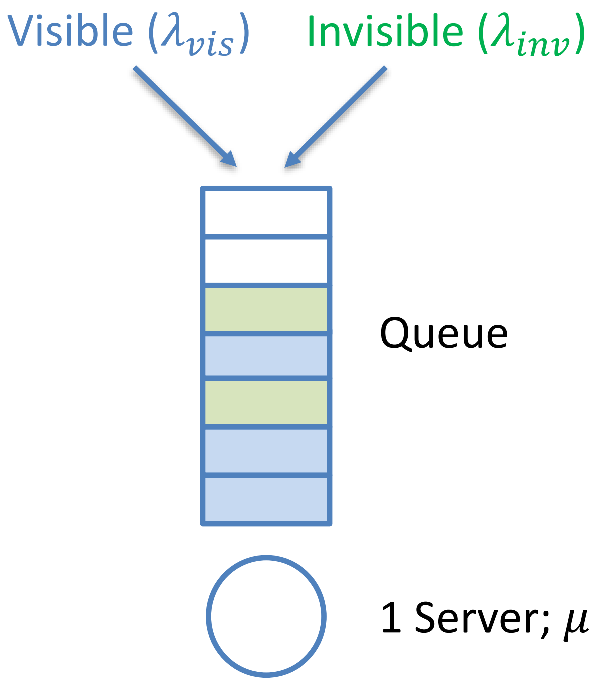


Simple Model: M/M/1 Queue

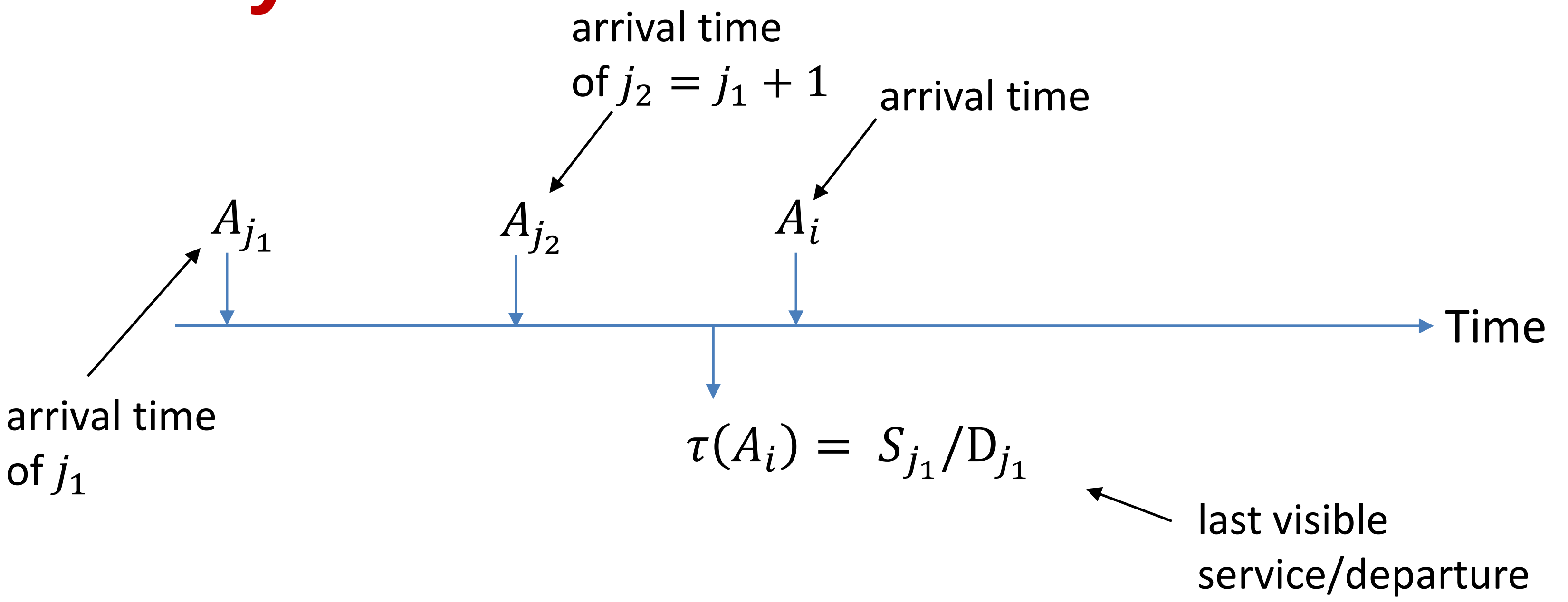
- Single-server, Poisson arrivals, exponential service times
- Arrival rate and service rate are known
- First-come first-served (FCFS)
- Actual system state (n_{vis}, n_{inv})

Best delay predictor: $\frac{n_{vis}+n_{inv}}{\mu}$

- Observed system state is (n_{vis})



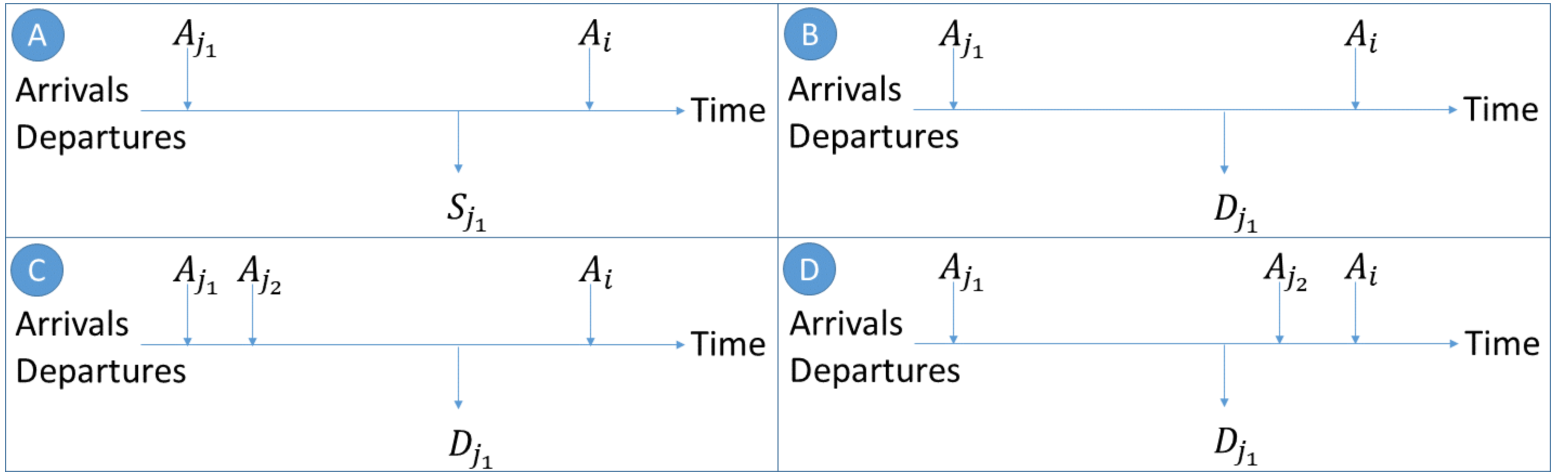
Analysis



Observations:

- Customers arriving before A_{j_1} do not affect prediction (FCFS)
- $S_{j_2} > A_i$: customer j_2 arrived but has not started service yet
- Invisible customers that arrive after A_{j_2} will not be served prior to A_i
- There are several cases based on the interplay between A_{j_2} and $\tau(A_i)$

Analysis – cont.



- There are four cases
- Using conditional expectations repeatedly we compute the expected number of invisible customers n_{inv} at A_i for each case.

Analytical solution for M/M/1

Proposition 1. *The total expected number of customers at time A_i can be written as*

$$\mathbb{E}\left[L^A(A_i)\right] = \lambda_{inv}(A_i - A_{j_1}) + L_{vis}(A_i) - 1.$$

$$\mathbb{E}\left[L^B(A_i)\right] = L_{vis}(A_i) - 1 + \frac{\rho_{inv}}{1-\rho_{inv}} - \frac{2\sqrt{\rho_{inv}}}{\pi}\int_0^\pi \frac{e^{-\gamma(y;\rho_{inv})\mu t}}{(\gamma(y;\rho_{inv}))^2} sin(y) \times$$
$$\left(\sqrt{\rho_{inv}} sin\left(y + \frac{\lambda_B sin(y)}{\sqrt{\rho_{inv}}}\right) - sin\left(\frac{\lambda_B sin(y)}{\sqrt{\rho_{inv}}}\right)\right) e^{-\lambda_B\left(1-\frac{cos(y)}{\sqrt{\rho_{inv}}}\right)} dy.$$

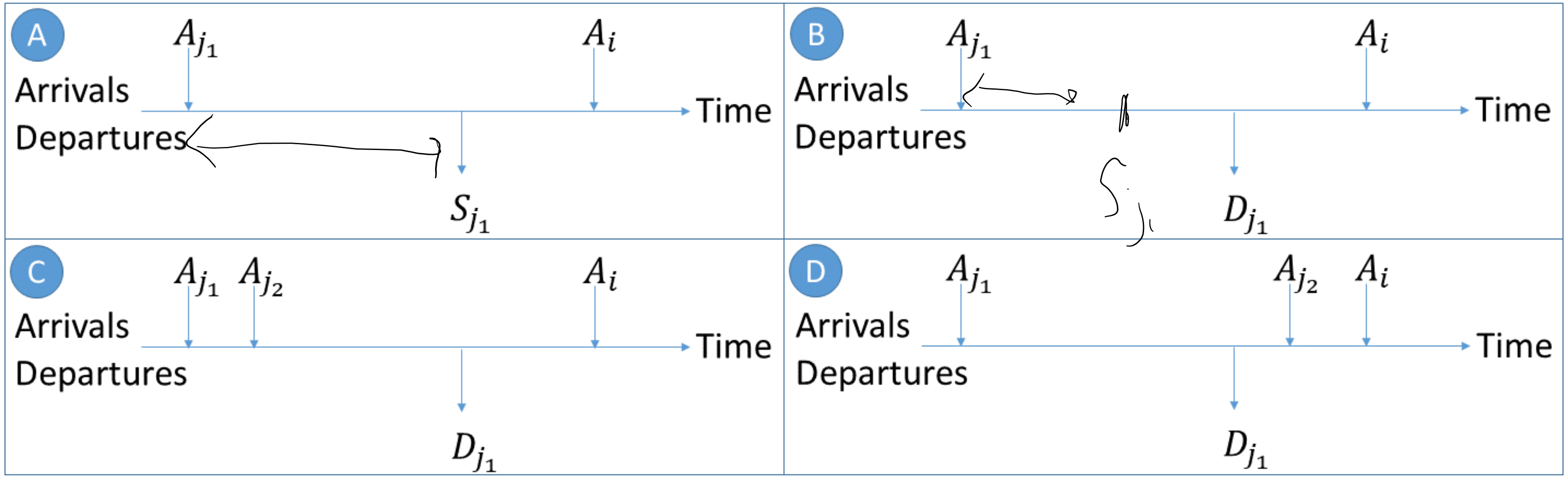
$$\mathbb{E}\left[L^C(A_i)\right] = L_{vis}(A_i) - 1 + \sum_{l=1}^{\infty} l \cdot \frac{\mu_C^l I_l(\sqrt{\lambda_C \mu_C})}{\sum_{l'=1}^{\infty} \mu_C^{l'} I_{l'}(\sqrt{\lambda_C \mu_C})} + \lambda_{inv}(A_i - A_{j_2}).$$

$$\mathbb{E}\left[L^D(A_i)\right] = L_{vis}(A_i) - 1 + \lambda_{inv}(A_i - A_{j_2}) +$$
$$\sum_{r=1}^{\infty} r \cdot \frac{\sum\limits_{l=r+1}^{\infty} \frac{(\mu_D)^{l-r}}{(l-r)!} \sum\limits_{n=0}^{\infty} \frac{(\lambda_D)^n}{n!} \cdot P_{n,l}(A_{j_2} - D_{j_1}; \lambda_{inv}, \mu, \rho_{inv})}{\sum\limits_{r'=1}^{\infty} \sum\limits_{l=r'+1}^{\infty} \frac{(\mu_D)^{l-r'}}{(l-r')!} \sum\limits_{n=0}^{\infty} \frac{(\lambda_D)^n}{n!} \cdot P_{n,l}(A_{j_2} - D_{j_1}; \lambda_{inv}, \mu, \rho_{inv})}.$$

where $\rho_{inv} = \lambda_{inv}/\mu$, $\lambda_B = \lambda_{inv}(D_{j_1} - A_{j_1})$, $\lambda_C = \lambda_{inv}(A_{j_2} - A_{j_1})$, $\mu_C = \mu(A_i - D_{j_1})$, $\lambda_D = \lambda_{inv}(D_{j_1} - A_{j_1})$, *and* $\mu_D = \mu(A_i - A_{j_2})$.

Observations from M/M/1 Analysis

- Important time points and temporal intervals
- Prediction depends on the case (A/B/C/D)



Types of Prediction Methods

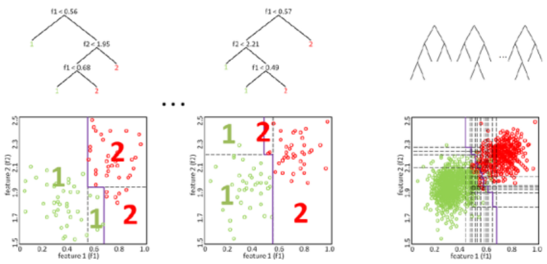
1. **Direct prediction via queue length estimation (no ML):**

   D1: Analytic estimate of invisible queue (Proposition 1) – results only for M/M/1 $\frac{n_{vis} + n_{inv}}{\mu}$

   D2: Adjusted queue length $\frac{Q_{vis}}{p_{vis}}$ (20% visible; I observe 10 -> total = 50)

2. **Snapshot prediction (heavy-traffic). No learning.**

3. **Machine learning with one of the following feature sets:**



   F0: Prophet (fully-observed queue) – lower bound (used for scaling)

Available at prediction time:

   F1: Visible queue length only
   F2: Visible queue length + case (A,B,C or D)
   F3: Visible queue length + time differences
   F4: Visible queue length + estimate of invisible queue + time differences
   F5: Visible queue length + estimate of invisible queue
   F6: Visible queue length + Snapshot predictor as a feature

- Linear regression
- Lasso
- Regression Trees
- KNN
- ….

Experimental setting

- Numerical experiment:
  - $\mu = 1,\ \lambda \in [0.49, 0.54, \dots, 0.99],\ p_{vis} \in [0.1, 0.2, \dots, 1]$
  - 120,000 customers per run, first 1,000 are omitted
  - 80%-20% training-test (time-order respecting) split for ML methods
  - Predict delay for every arriving visible customer using one of the methods (direct, snapshot, ML)
- For ML methods: we use the 6 feature sets together with different ML algorithms (Linear regression, LASSO, Decision Trees,…)

Results for Direct and Snapshot Methods

- Scaled Mean Squared Error (sMSE):
  - **average (stdev)** across all scenarios

D1: Proposition 1	D2: Visible Q adjust	Snapshot Prediction
1.6 (0.46)	3.28 (2.88)	3.18 (0.75)

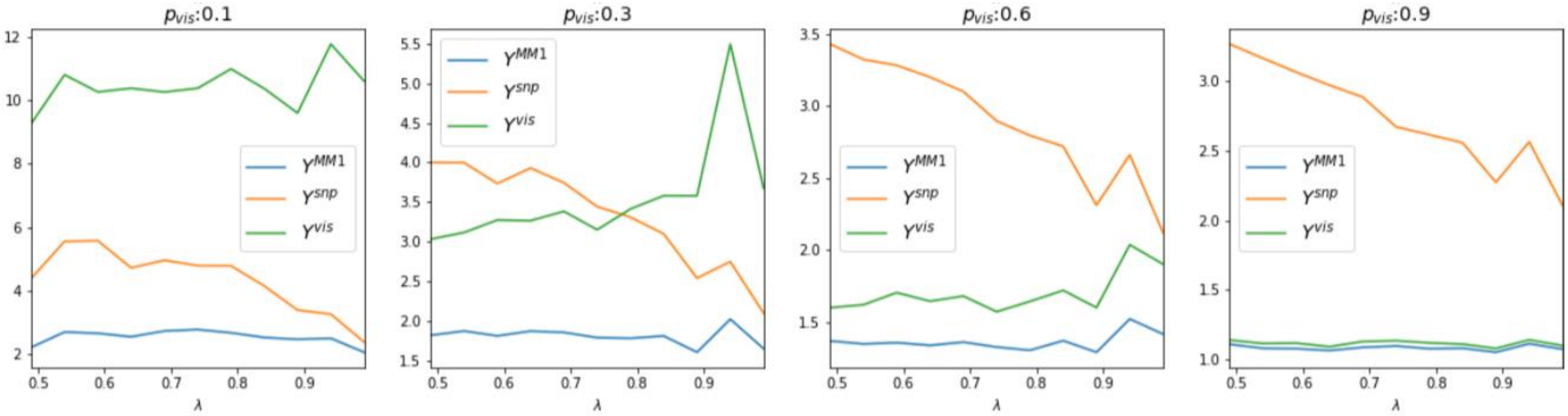
Results for ML-based Methods (sMSE)

Best non-ML method: 1.6 (0.46)

Feature Set	Reg. Trees	KNN	LASSO	Linear Regression
F0: Prophet	1.04 (0.06)	1.16 (0.09)	1 (0.00)	1 (0)
F1: Visible only	2.32 (1.68)	2.66 (1.88)	2.28 (1.62)	2.47 (1.78)
F2: Visible + Case (A/B/C/D)	2.22 (1.60)	2.50 (1.82)	2.18 (1.57)	2.20 (1.50)
F3: Visible + Time diff	3.38 (0.96)	1.89 (0.57)	**1.65 (0.48)**	**1.67 (0.51)**
F4: Visible + Prop. 1+Time diff	3.38 (0.95)	1.88 (0.57)	**1.60 (0.45)**	**1.60 (0.46)**
F5: Visible + Prop. 1	3.28 (0.93)	1.85 (0.55)	**1.60 (0.45)**	**1.60 (0.45)**
F6: Visible + Snapshot	4.57 (0.64)	3.31 (0.47)	2.66 (0.39)	2.97 (0.52)

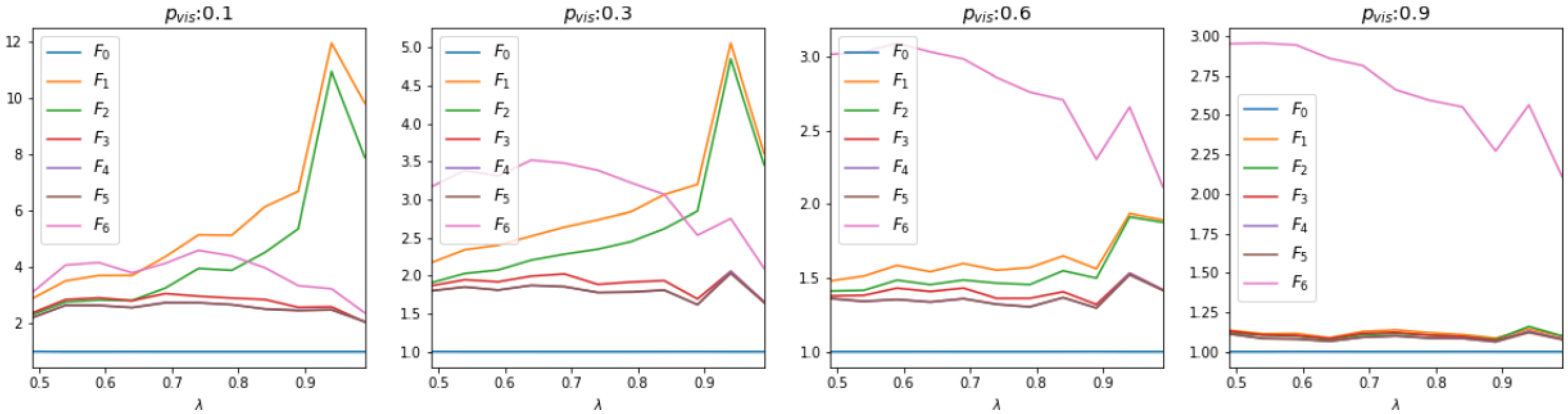
- Linear models work well
- ML methods do not improve over direct estimation (M/M/1) - expected
- The identified time differences improve predictions considerably without prop 1
- Closed-formula does not improve much beyond time differences

non-ML: Sensitivity to Visibility and Load (sMSE)



- ○ Analytical result (blue) outperforms the other non-ML methods (in line with table)
- ○ Snapshot prediction (orange) combined with ML improves for higher load (expected)
  - **High chances for recent/relevant visibility**

LASSO: Sensitivity to Visibility and Load (sMSE)



- For low visibility: methods F3, F4 and F5 that use the analytic result and/or time differences work best
- For high visibility: methods that use # of visible customers are good enough

Prediction in 3 (more) Complex QSystems

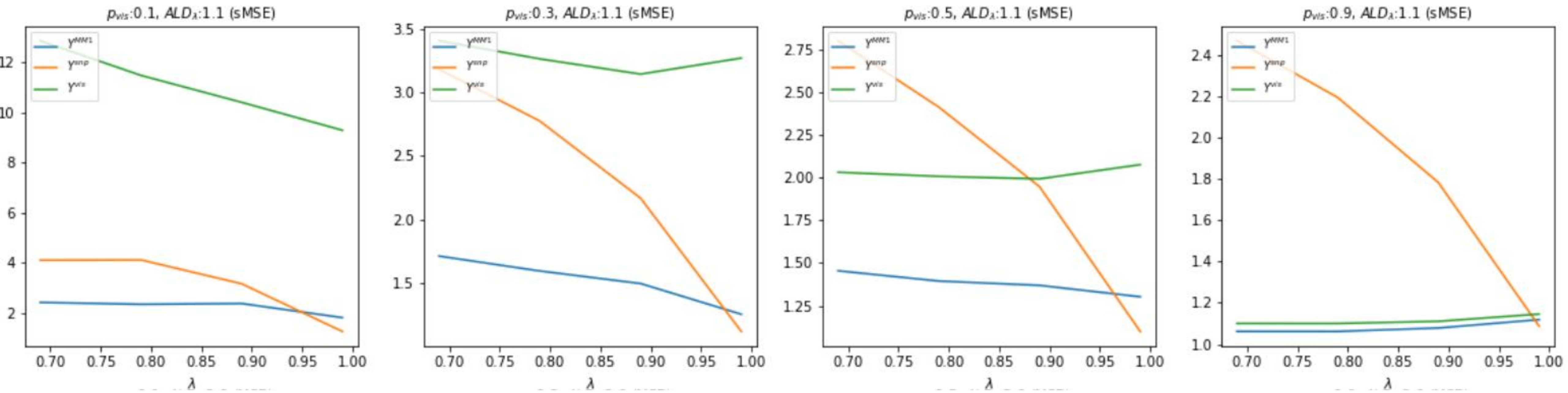
- Before we apply the method to real data, we need insights into 3 complex systems (using their synthetically generated data):
    - **G/M/1 queue – appointment-based arrivals (+noise)**
    - M/G/1 queue - non-exponential service times
    - M(t)/M/1 queue - time-varying Poisson arrivals

- Existing theory breaks in all 3

- We use features from the M/M/1 experiment including the analytical solution for the M/M/1 queue (even though assumptions do not hold)

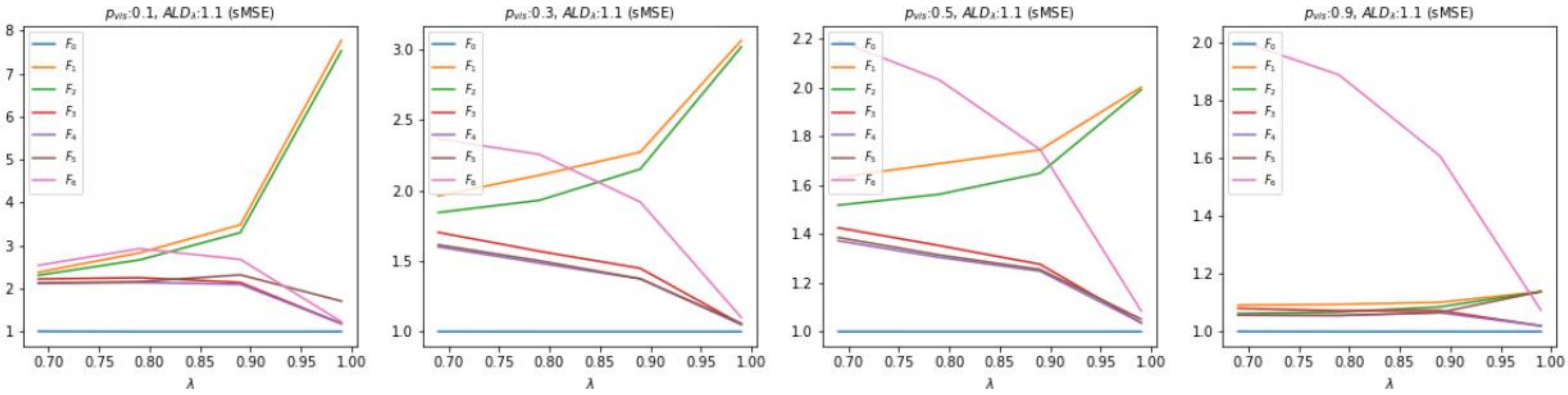
- Reminder: the idea is to complicate the system to resemble real hospital data

G/M/1 Queue: non-ML Results (sMSE)



- **Analytical result (blue) remains relevant when assumptions are violated**
- Snapshot prediction (orange) improves for higher load
- Using the visible queue (green) is always worse

G/M/1 Queue: LASSO Results (sMSE)



- For low visibility methods that use the **time differences** work best
  - **Especially in heavy load!**
- For high visibility, the methods that use # of visible customers work well
- Methods based on analytical result are still highly relevant (when fed into ML)

Conclusion

- New prediction problem: prevalent in sensor data
- Analytical solution for a base case – M/M/1 queues
- Identified potentially useful features
- ML-based approach in more general cases
- Numerical experiments suggest that
    - Existing benchmarks fail
    - Features are effective and linear models work well
    - ML approach seem to work well in general queues

- Ongoing work:
    - Gaining insights for more complex queueing models
    - Apply the methods to Dana-Farber data and compare to naive prediction

Thank you!

arik.senderovich@utoronto.ca