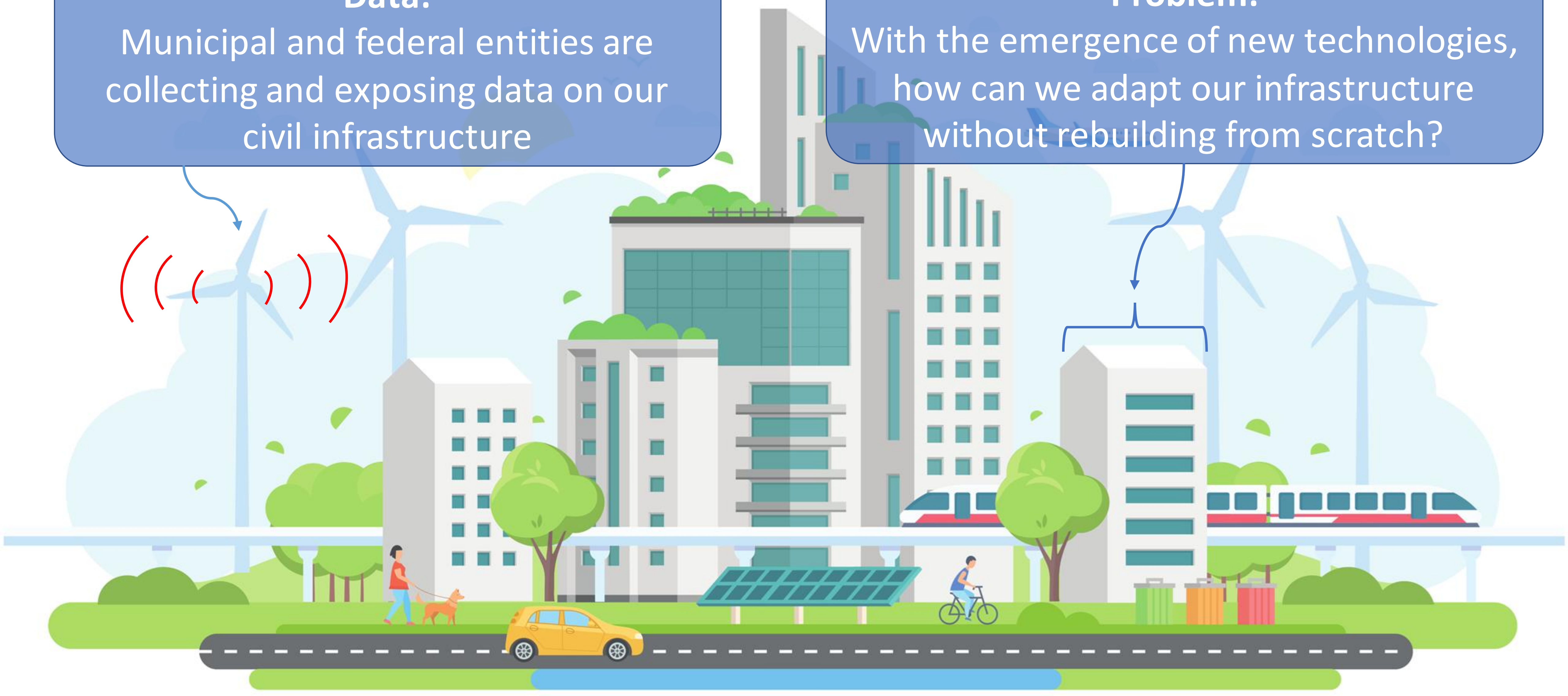
Applications of Statistical & Machine Learning in Civil Infrastructure

Chase Dowling
University of Washington
Electrical and Computer Engineering

Problem:
With the emergence of new technologies, how can we adapt our infrastructure without rebuilding from scratch?

Data:
Municipal and federal entities are collecting and exposing data on our civil infrastructure

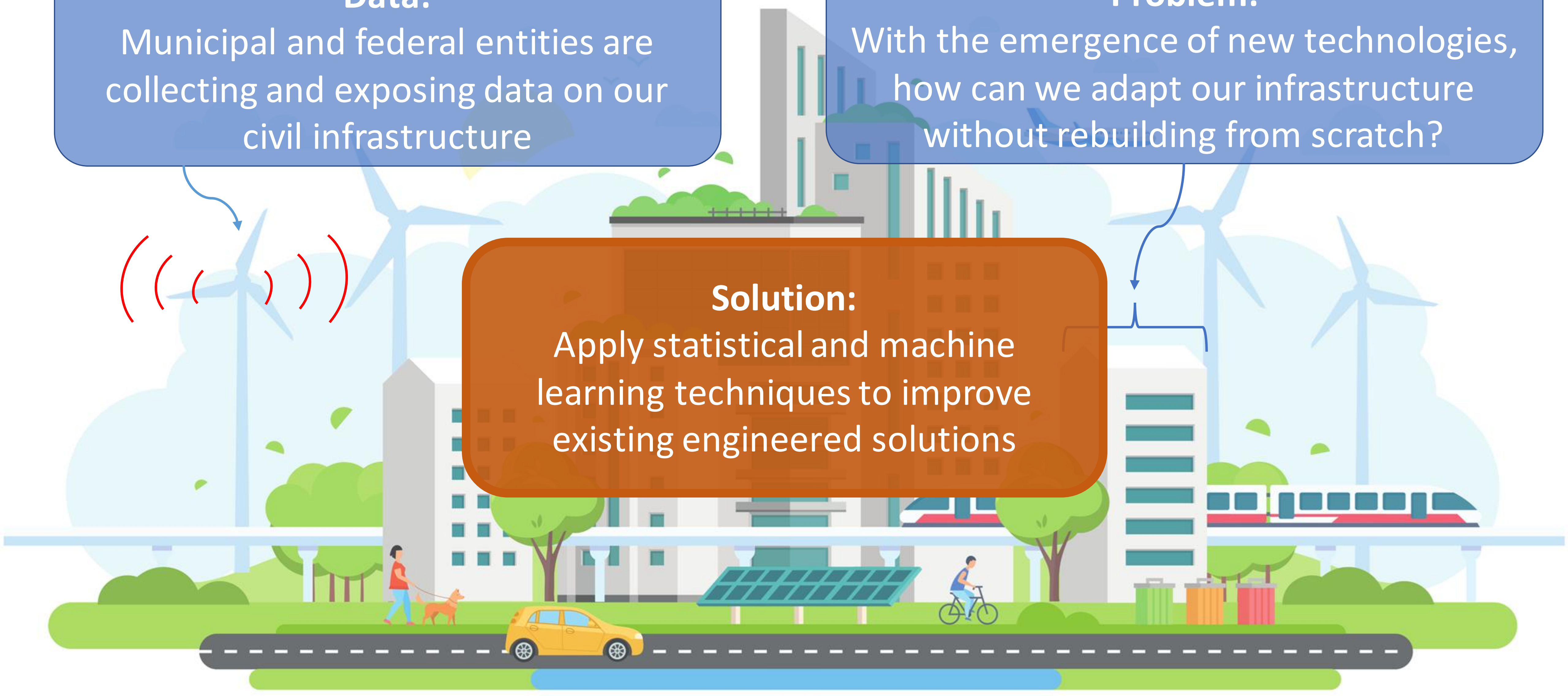
Problem:
With the emergence of new technologies, how can we adapt our infrastructure without rebuilding from scratch?



Data:
Municipal and federal entities are collecting and exposing data on our civil infrastructure

Problem:
With the emergence of new technologies, how can we adapt our infrastructure without rebuilding from scratch?

Solution:
Apply statistical and machine learning techniques to improve existing engineered solutions



Case studies in combining civil data with statistical and machine learning

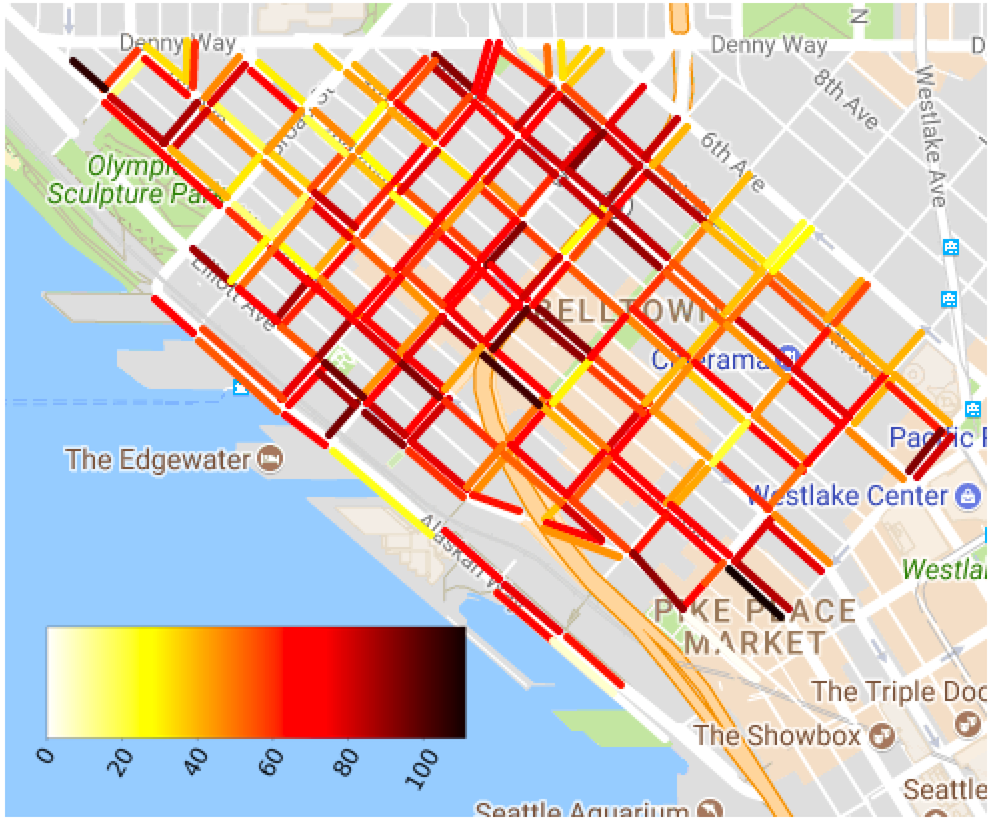
1. Lots of opportunity
2. Growing # of examples where lack of domain knowledge leads to inactionable solutions in high reliability areas
3. Immature use in control tasks

Transportation: Modeling Congestion Due to Parking

1. SDOT has made city transportation data available.
2. How much congestion due to parking?
3. How can we control resulting congestion?

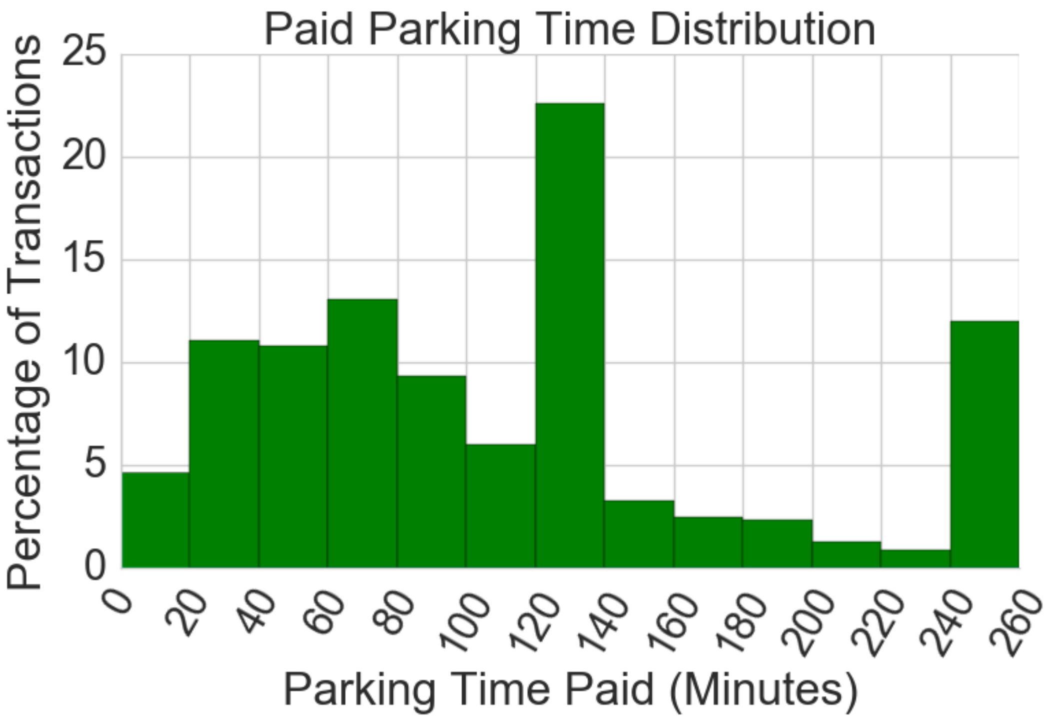
Curbside Parking in Belltown

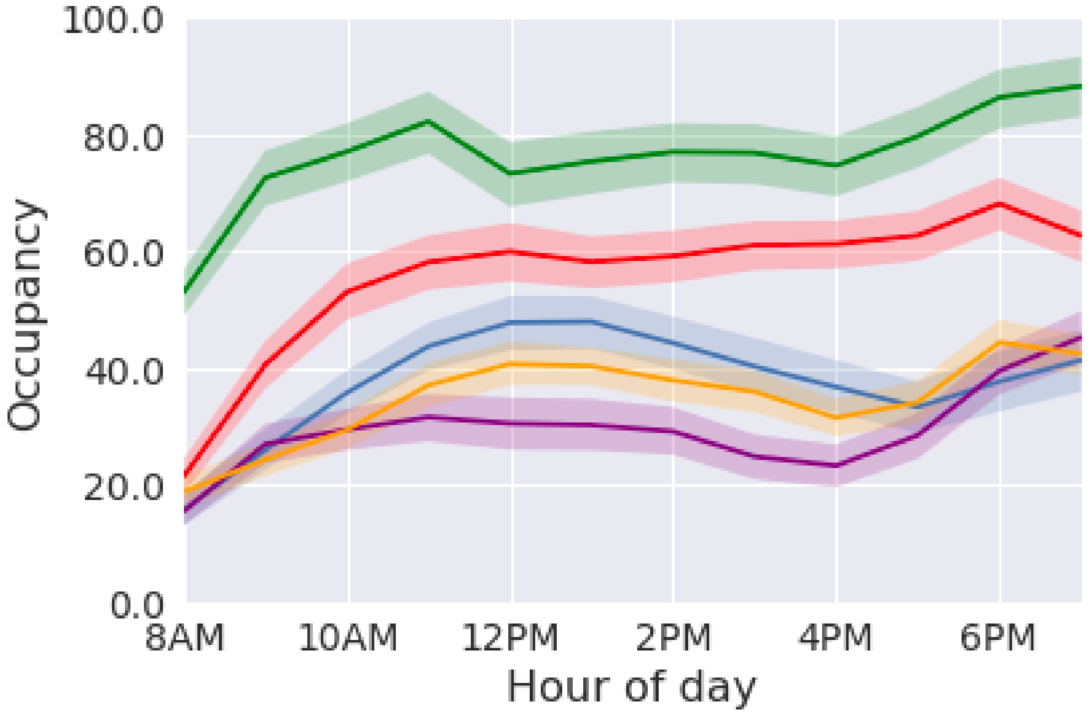
1. Cars spend most of their time parked taking up space in the city
2. Economists have been trying to balance the cost of parking since the 1950’s
3. Price sensitivity traditionally measured by manual survey
4. Lots of new data



SDOT Parking Transaction Data

Seattle Department of Transportation exposes date, time, location, and paid parking time for curbside parking by neighborhood





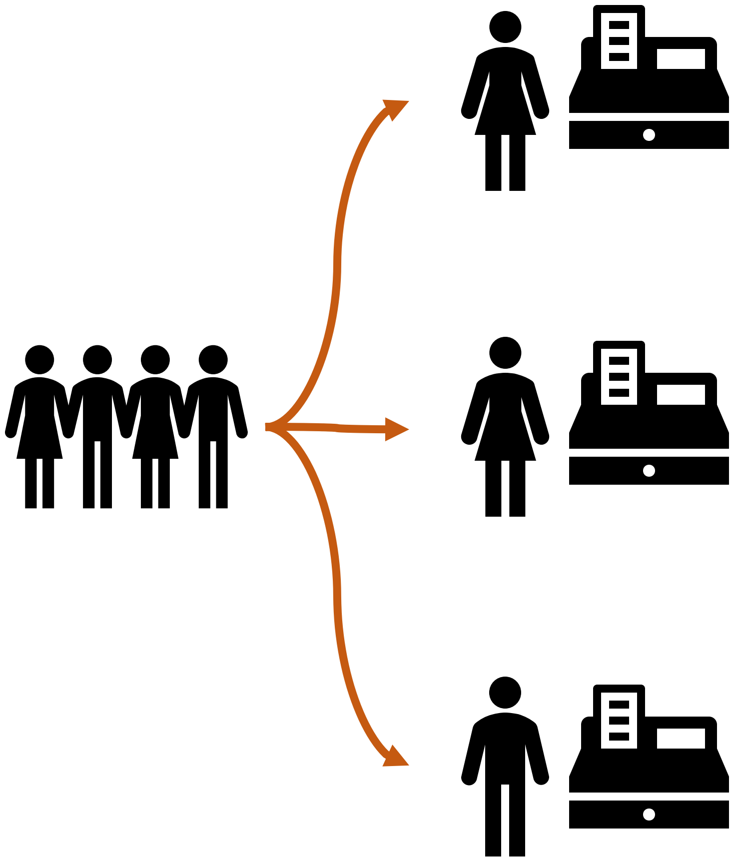
Occupancy over time at 5 different block-faces in Belltown

Queueing Theory

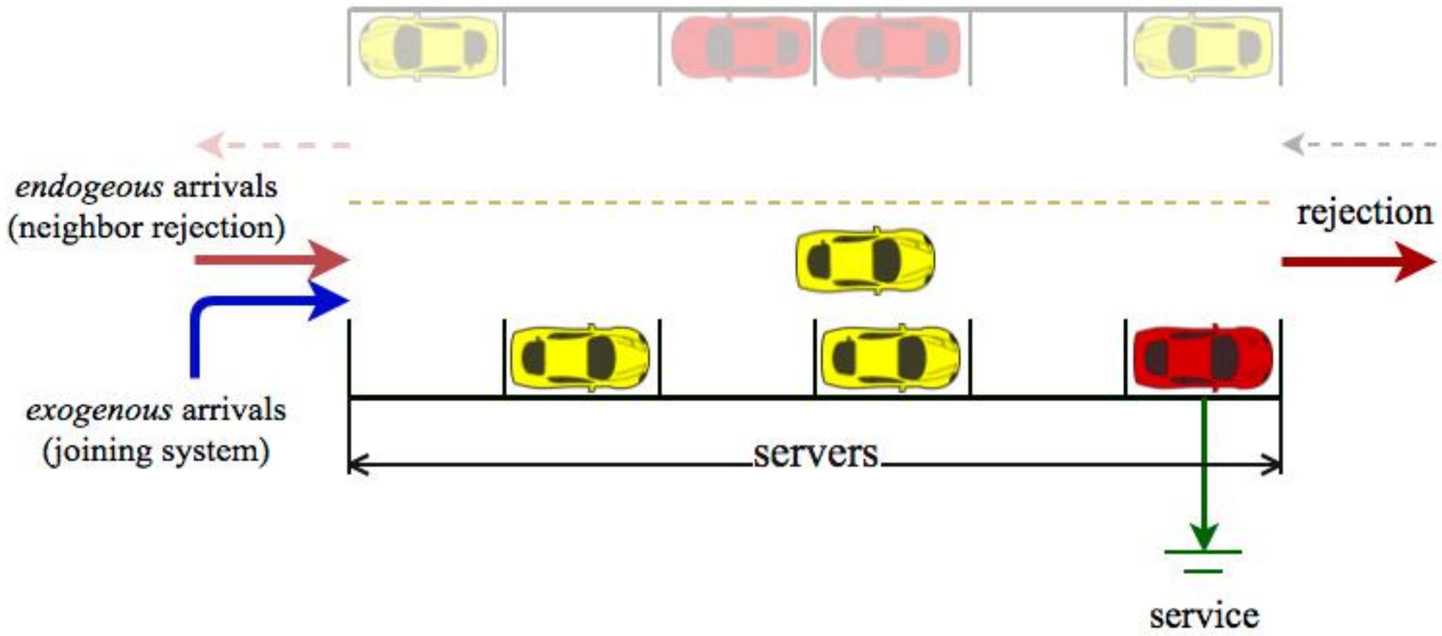
Curbside parking with transaction data is amendable to analysis by queueing theory

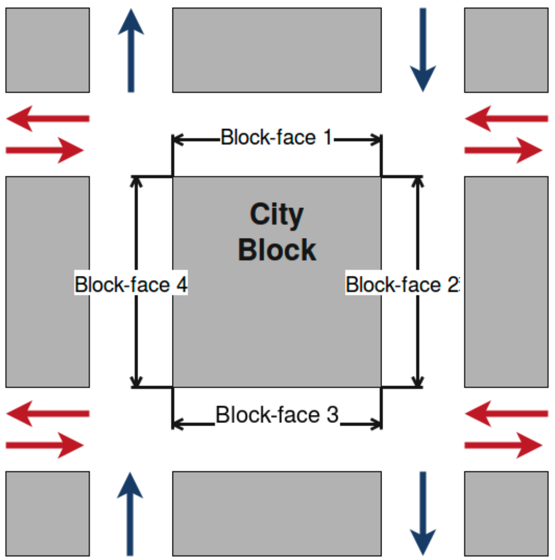
Queue is a **random process:** the number of users currently in the queue

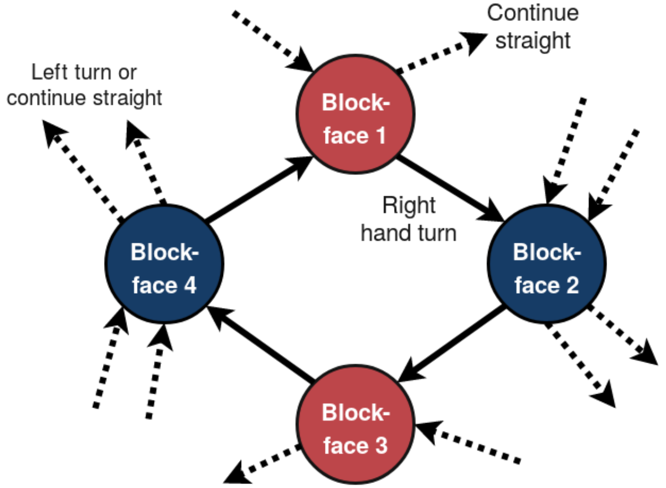
Given properties of the queue: arrival rate, service rate, number of servers, we can compute expected number of users in the queue, how long a user might expect to wait



Queueing Network







Arrival Rate

- Have hourly occupancy u from transaction data
- Want to compute total arrival rate y that attains observed occupancy in a network of M/GI/k/k queues with average parking time μ

Little's Law

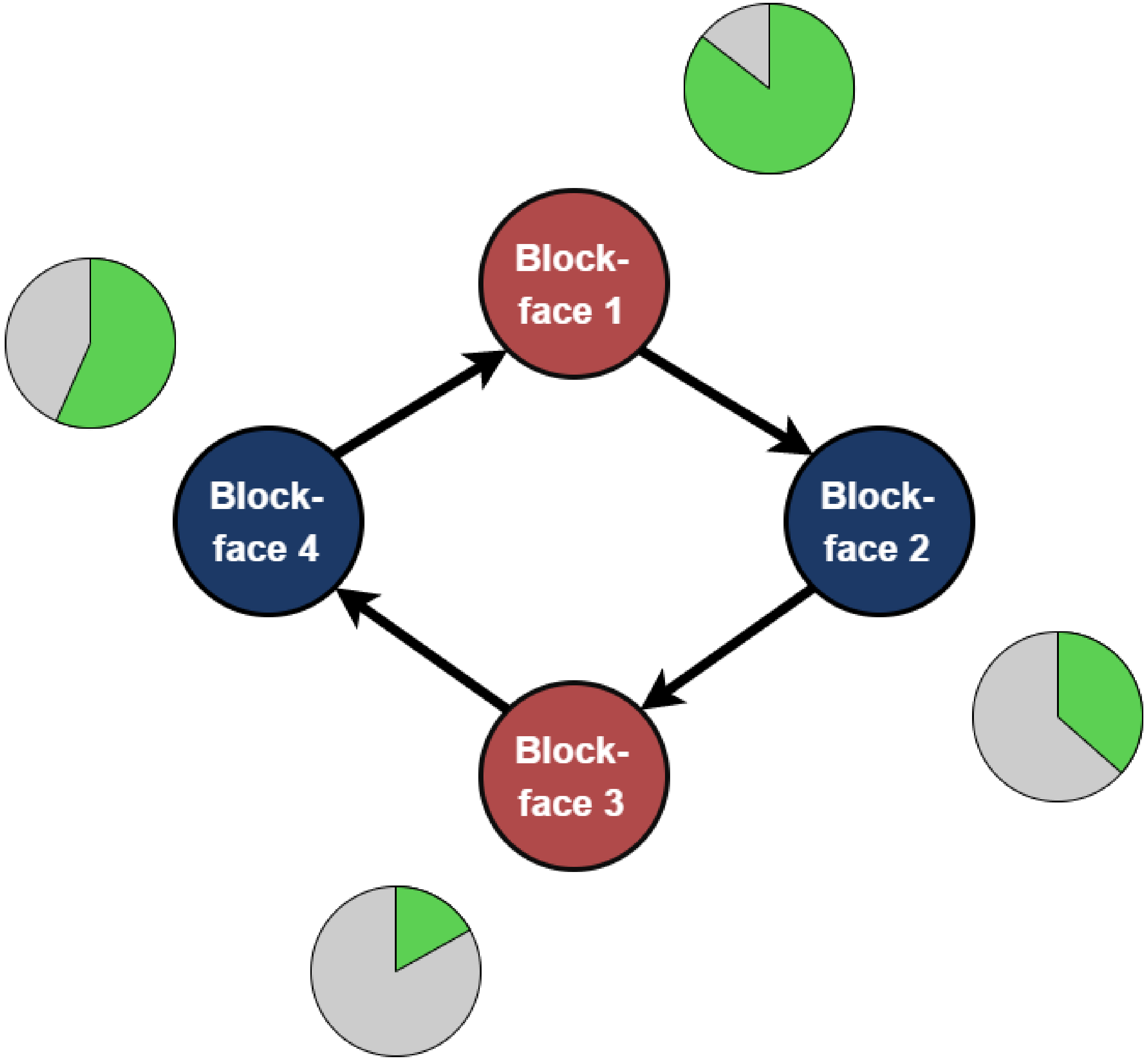
$$u = \frac{1}{k}\left[\frac{1}{\mu}\lambda(1-\pi_k)\right]$$

Total Arrival Rate

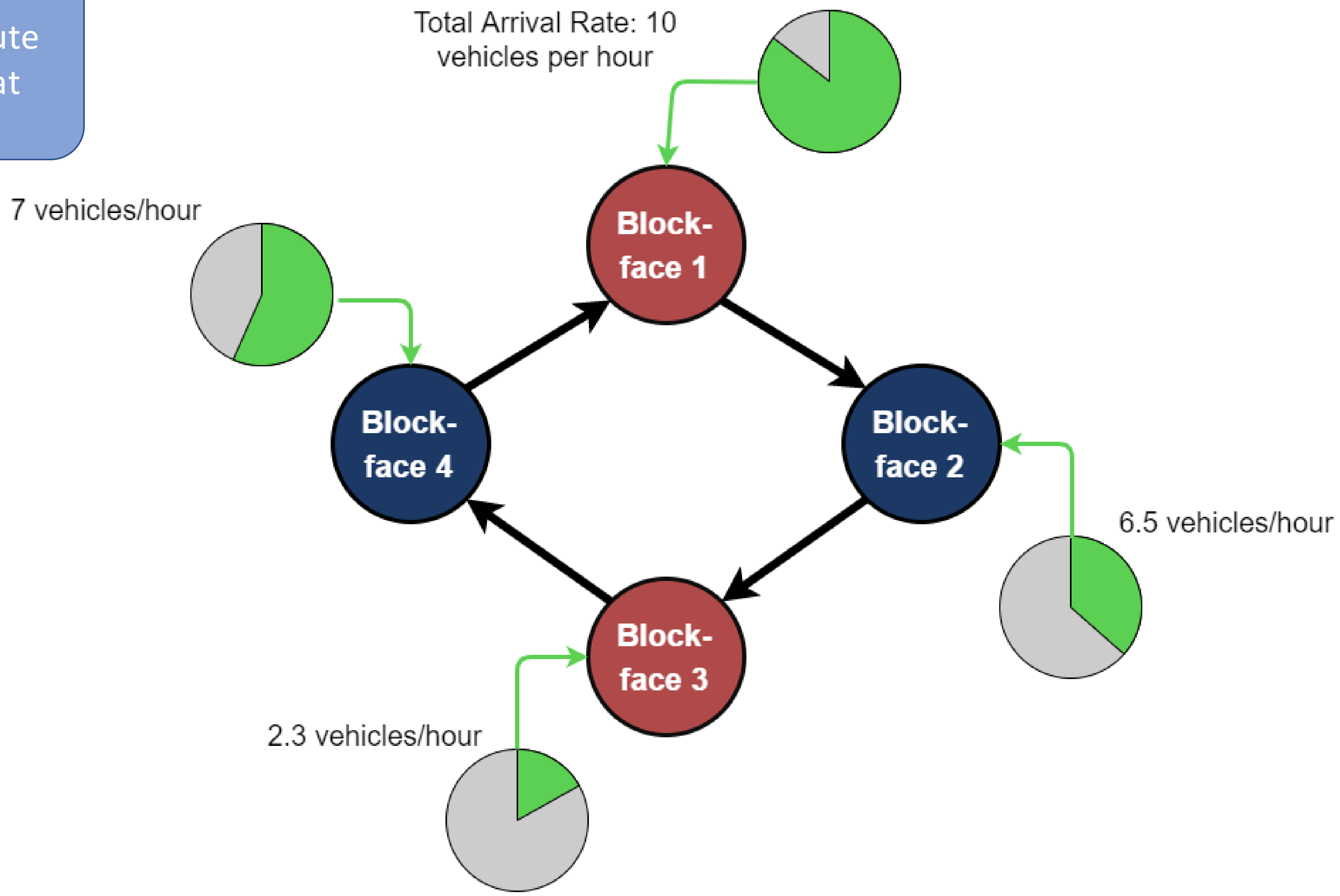
$$0 = \sum_{i=0}^{k} \frac{1}{\mu^{i-1}}\left[\frac{i - uk}{i!}\right] y^k$$

Dowling, C. P., Ratliff, L. J., & Zhang, B. (2019). Modeling Curbside Parking as a Network of Finite Capacity Queues. *IEEE Transactions on Intelligent Transportation Systems*.

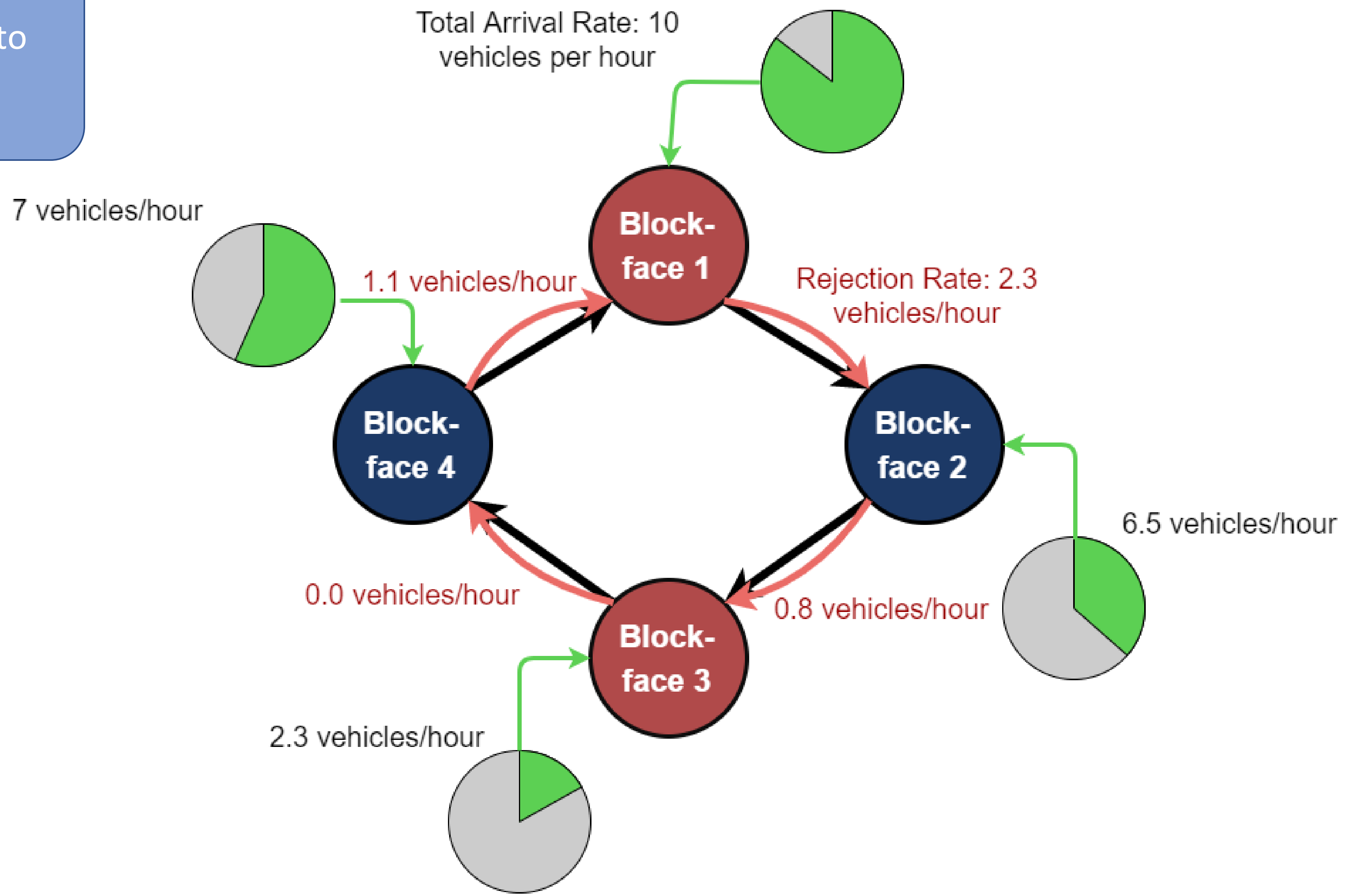
Step 1:
Compute occupancy at block-face using transaction data



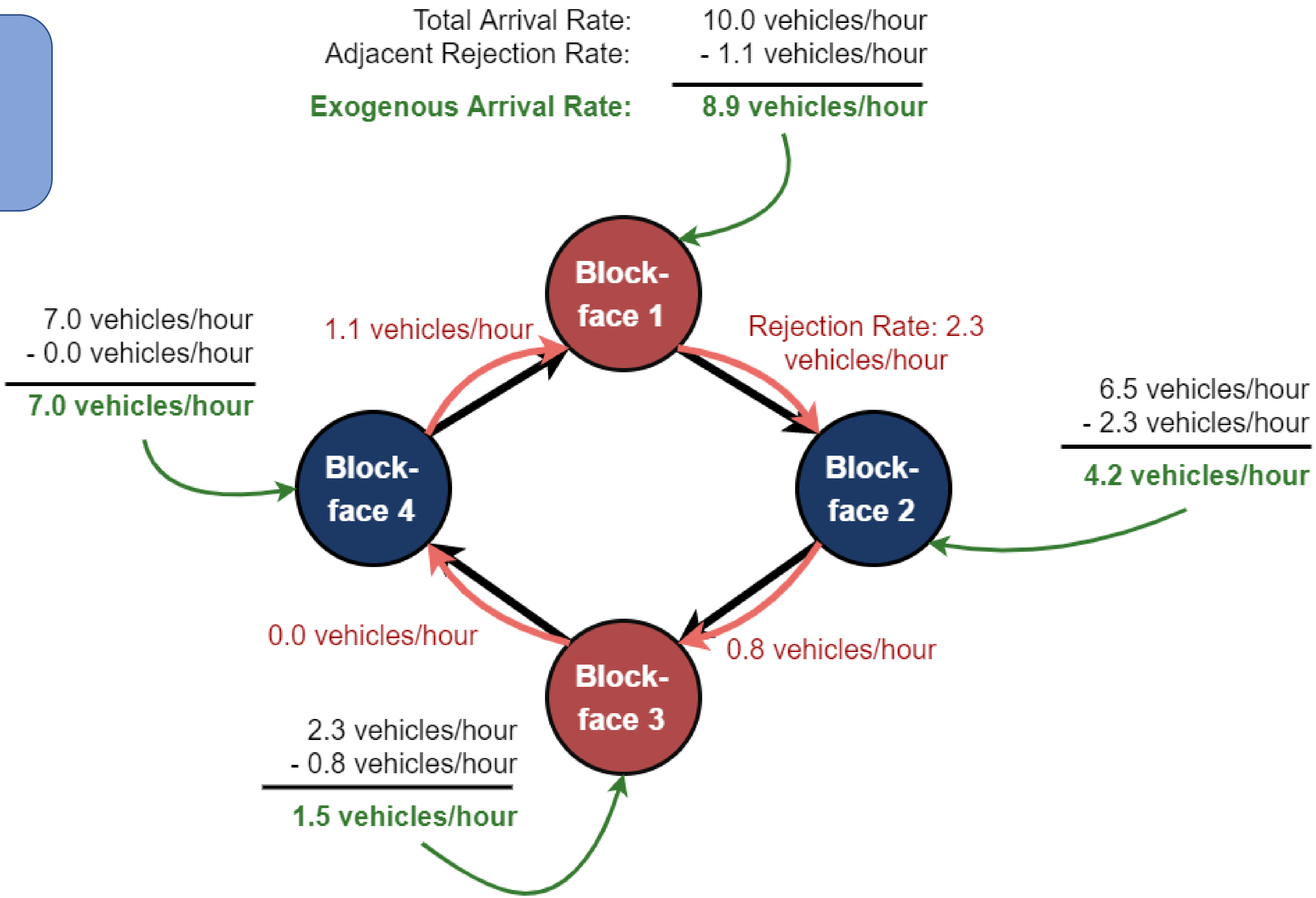
Step 2:
Little's Law to compute total arrival rate that attains occupancy



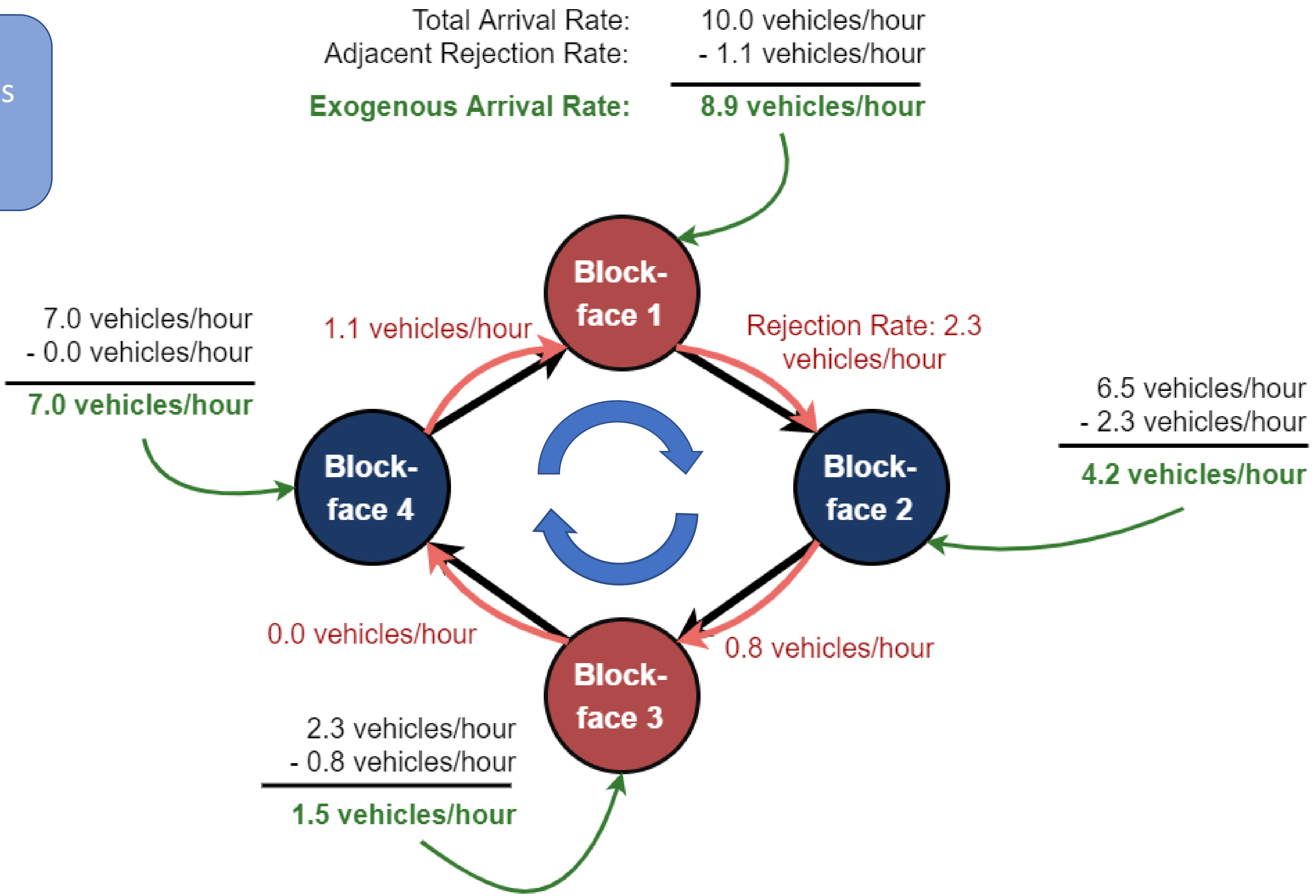
Step 3:
Use stationray dist. to determine rate of rejection



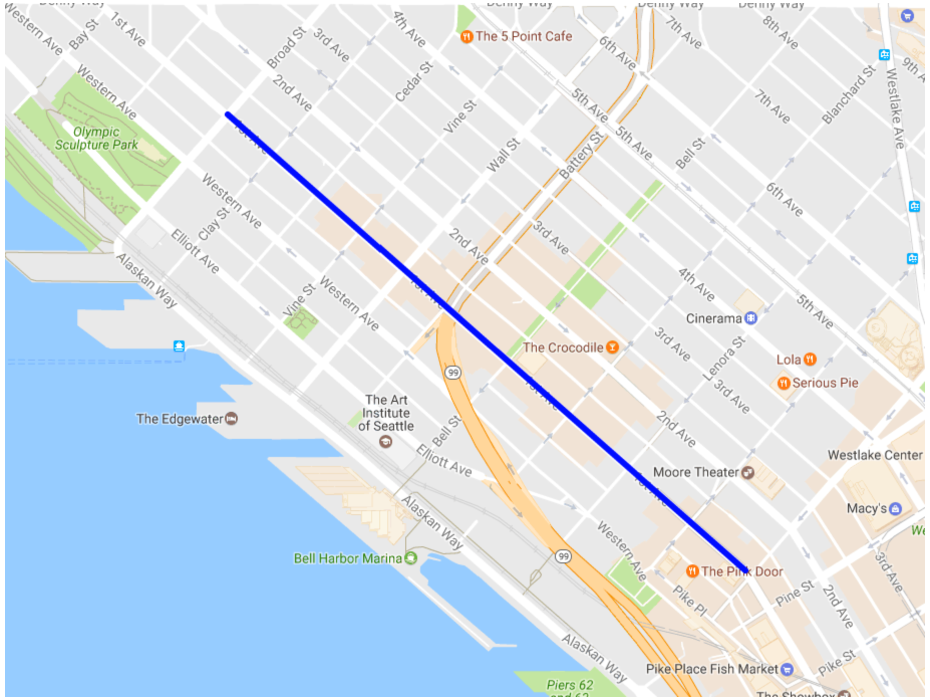
Step 4:
Subtract out rejection rates from totals by street topology



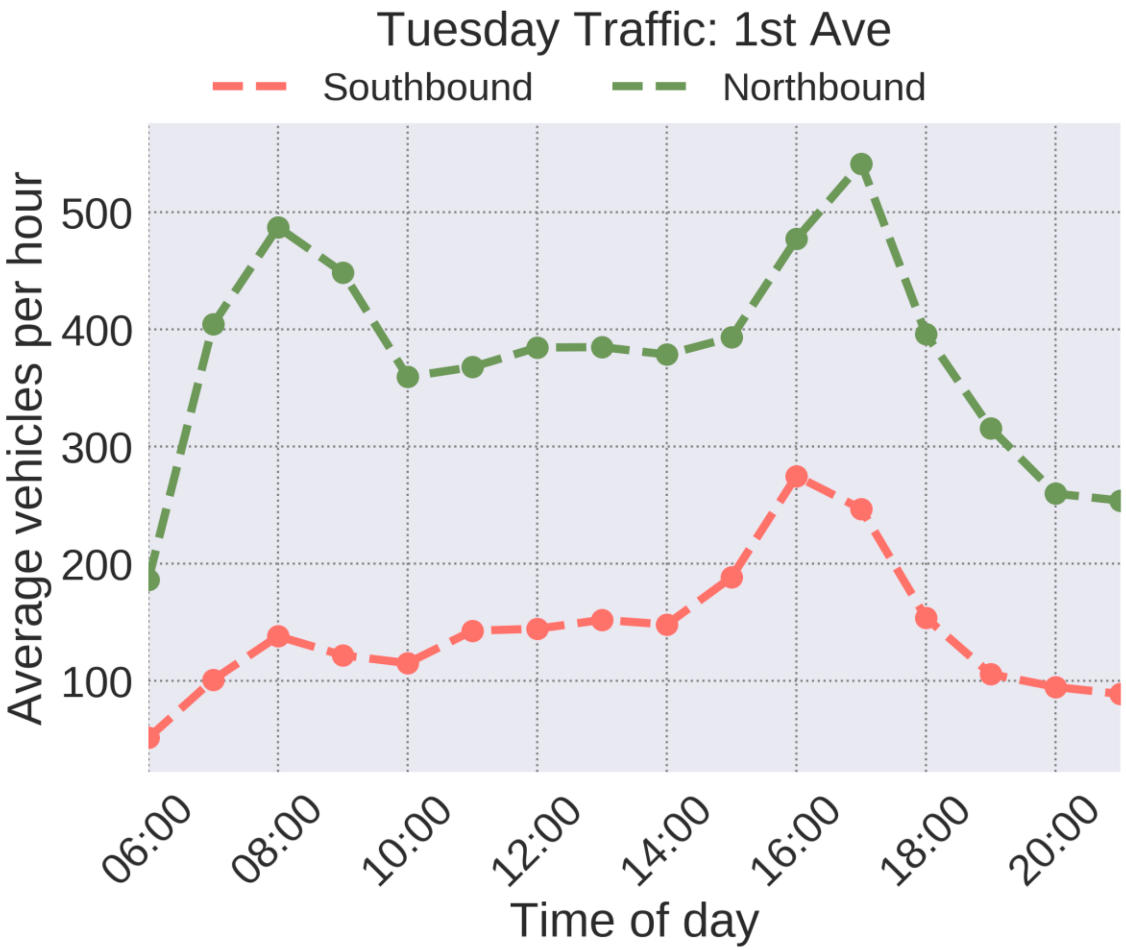
Step 5:
Compare rejection rates to roadway vehicles sensors



Computing Congestion

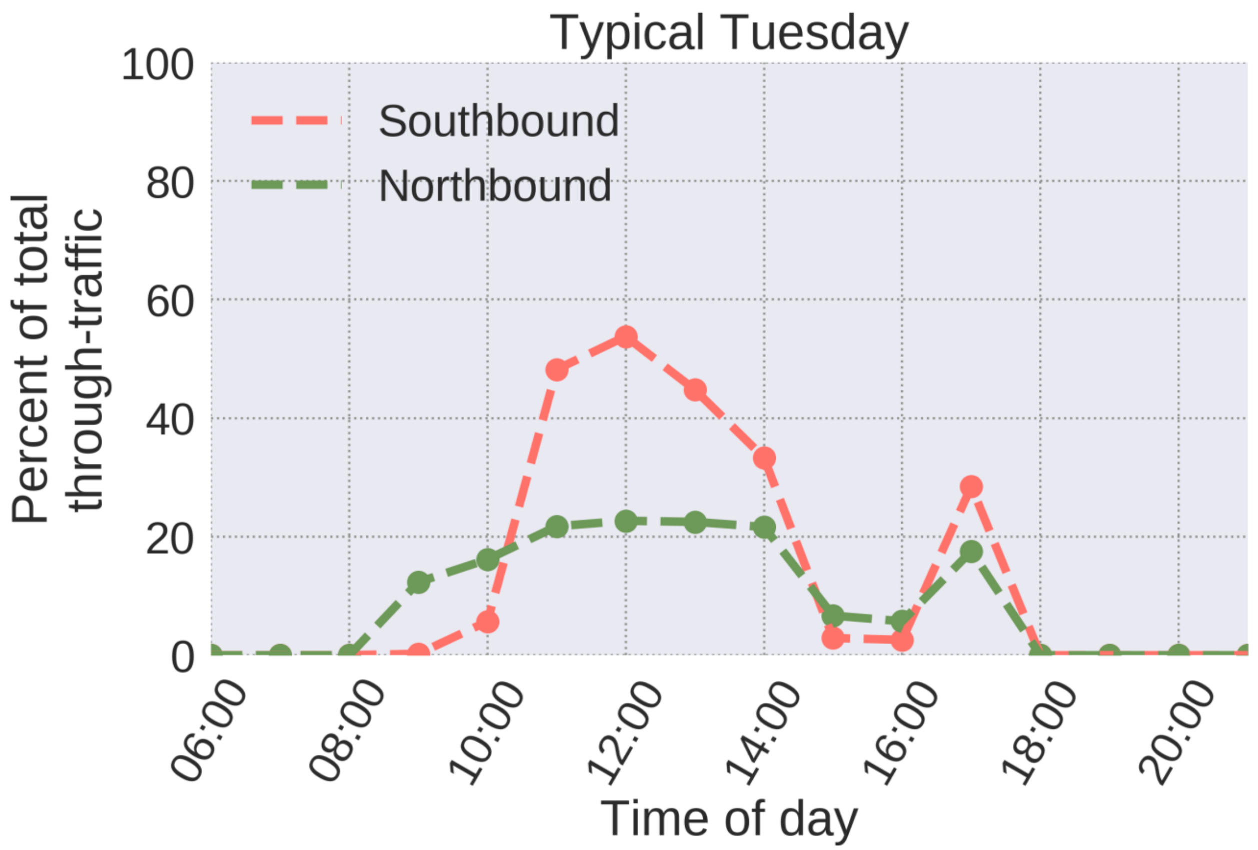


First Ave. in Belltown



Proportion Searching for Parking

1. Queueing network parameters learned from occupancy data along 1st Ave
2. Compute rejection rates along block-faces north and south-bound along 1st Ave & block-faces on cross-streets feeding into 1st Ave
3. Compare rejection rates along 1st to total through-traffic volume measured by SDOT roadway sensors

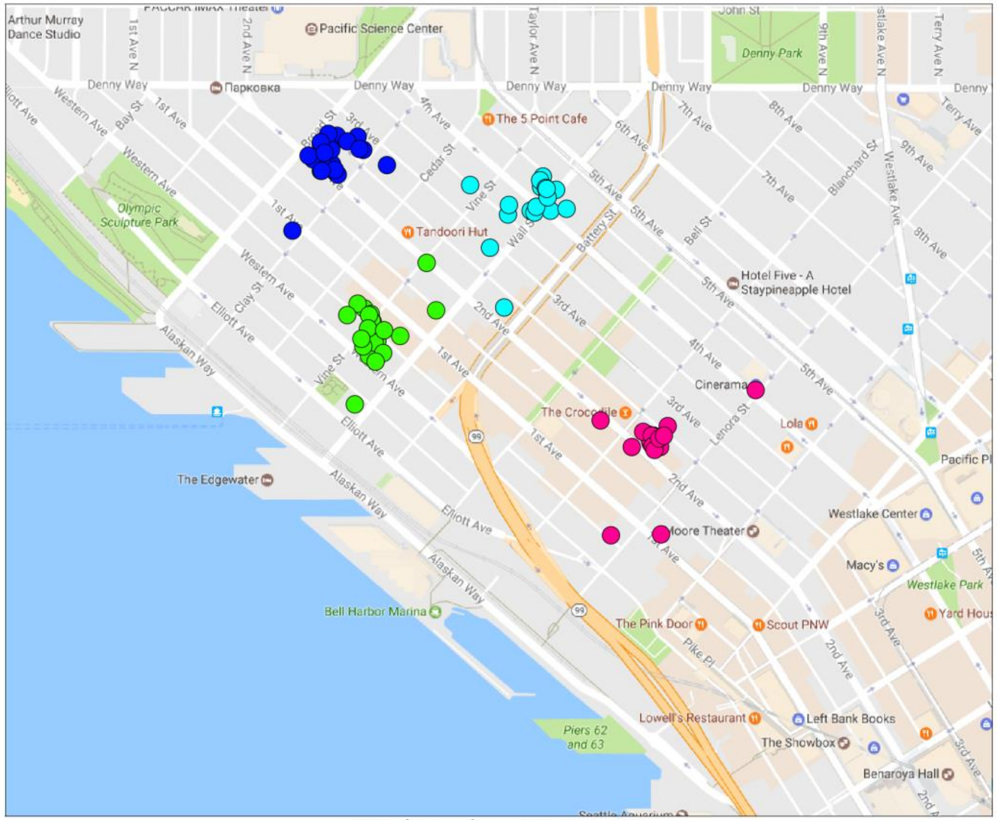


Maximizing Occupancy

Data shows most congestion originating from a handful of high occupancy block-faces

Occupancy as function of concave price elasticity U(pi) is convex

$$\begin{aligned} &\underset{\mathbf{p}}{\text{maximize}} && \sum_i \mathcal{U}(p_i) \\ &\text{subject to} && g_i(p_i) \le \bar{x}_i, \quad i = 1, \ldots, m. \end{aligned}$$



Wednesday 11:00 AM

Dowling, C., Fiez, T., Ratliff, L., & Zhang, B. (2017, December). Optimizing curbside parking resources subject to congestion constraints. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)* (pp. 5080-5085). IEEE.

Transportation: Summary

1. Parking transaction data & queueing theory provide a structural model for curbside parking in cities.
2. Occupancy is convex in price; can maximize subject to congestion constraints
3. Provides a mechanistic means to evaluate policy changes where ML predictors would fall short.

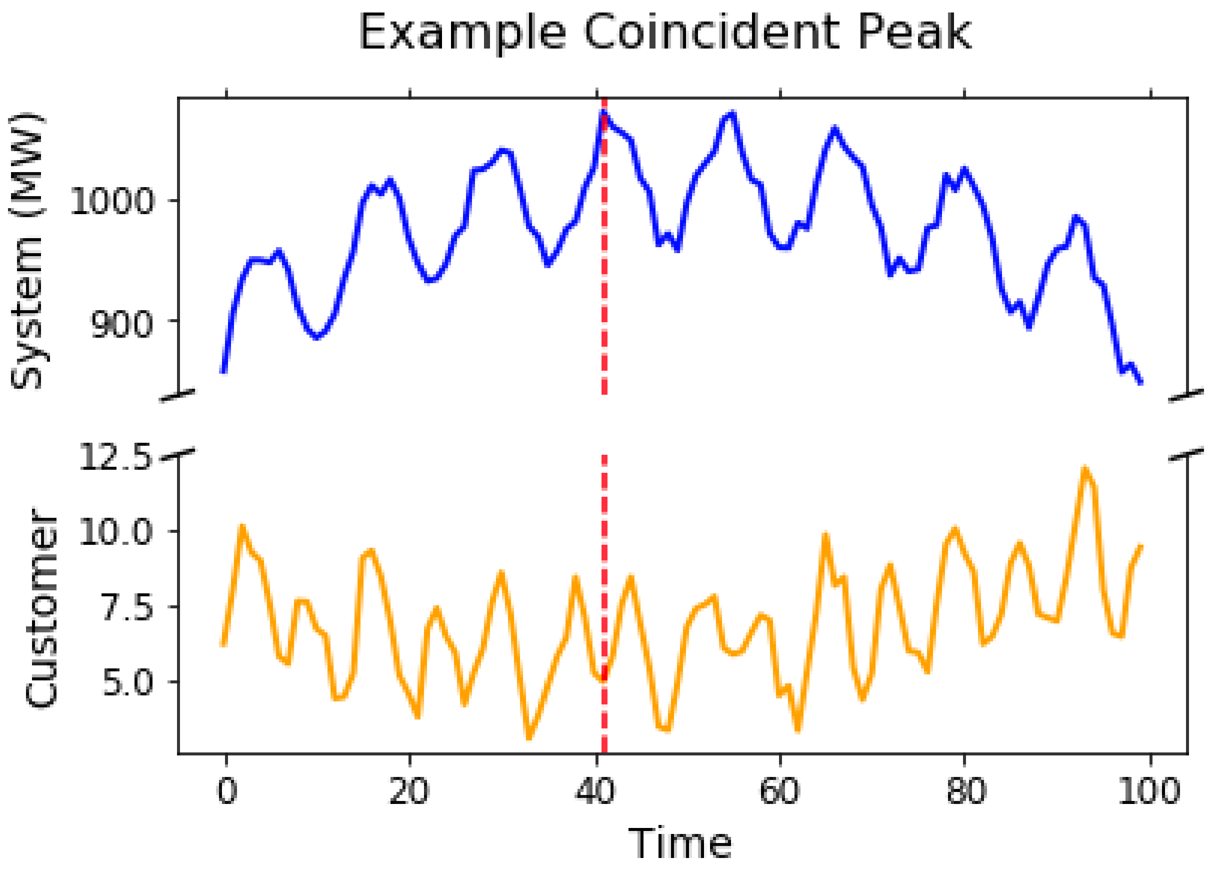
Electrical Grids: Mitigating Coincident Peak Pricing

1. What is a Coincident Peak (CP) and why do we care?
2. What do system operators do with system data?
3. What can power customers do with system data?
4. Can system data tell us if CP Pricing is effective?

Coincident Peaks

An electrical customer's coincident peak (CP) is their demand at the moment of the entire system's peak.

Systems levy transmission surcharges via CP electrical rates to reduce system peaks.

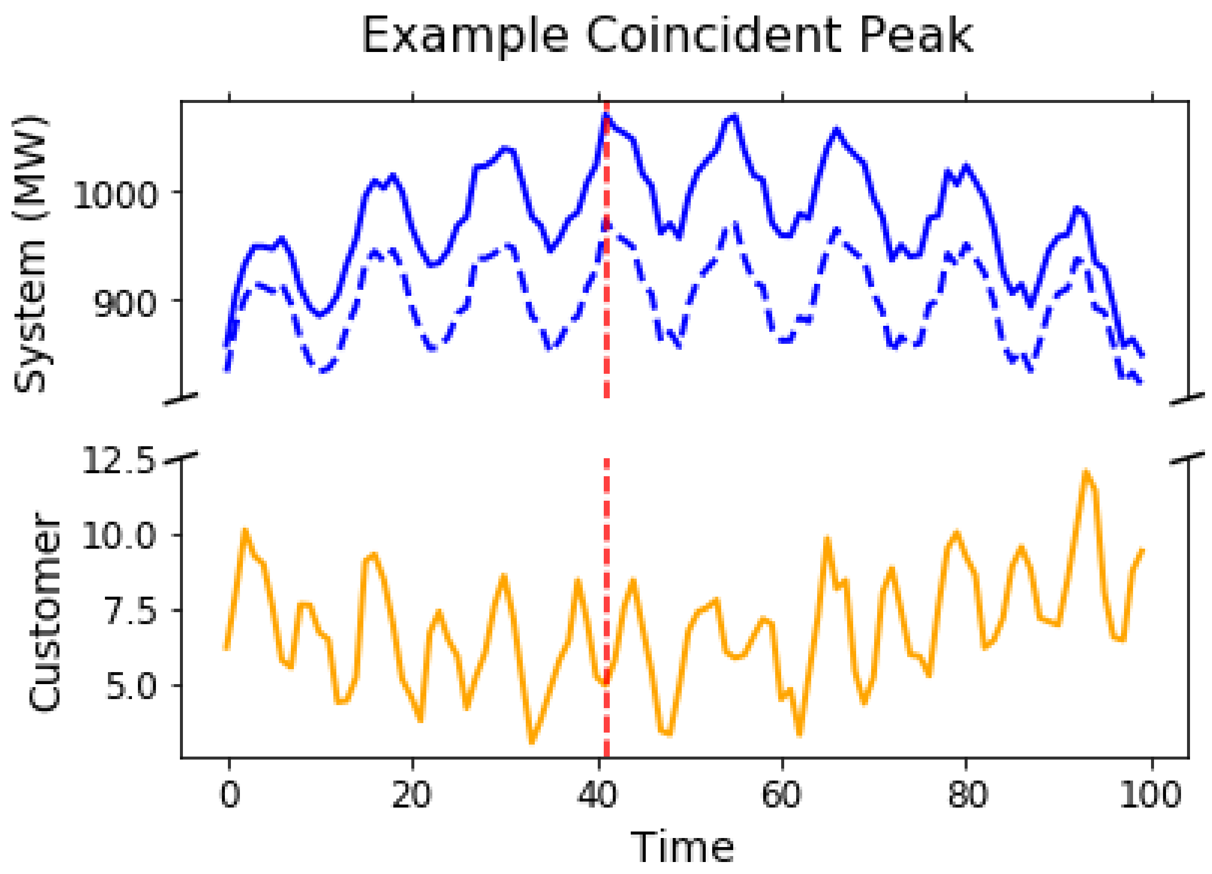


Coincident Peaks

CP rate roughly 100x more than normal time-of-use rates

Consumers participate in exchange for discounted time-of-use rates at all other times---breaks out long term expansion costs.

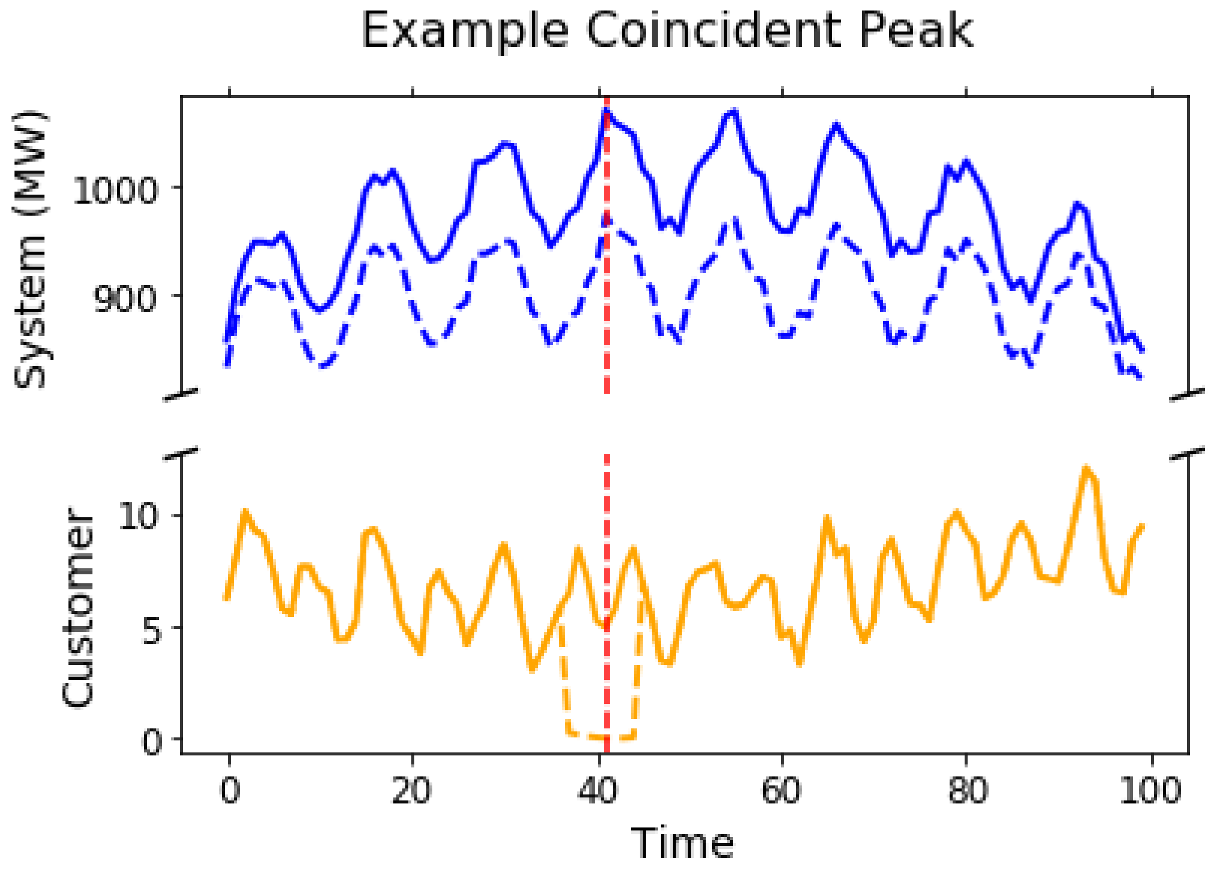
Goal is to curtail consumer demand at peaks



Coincident Peaks

4 MW consumer paying average ERCOT wholesale prices ($40/MWh), roughly $1.4 million in electricity costs per working year, $300k of which per year to consume electricity at CP hour

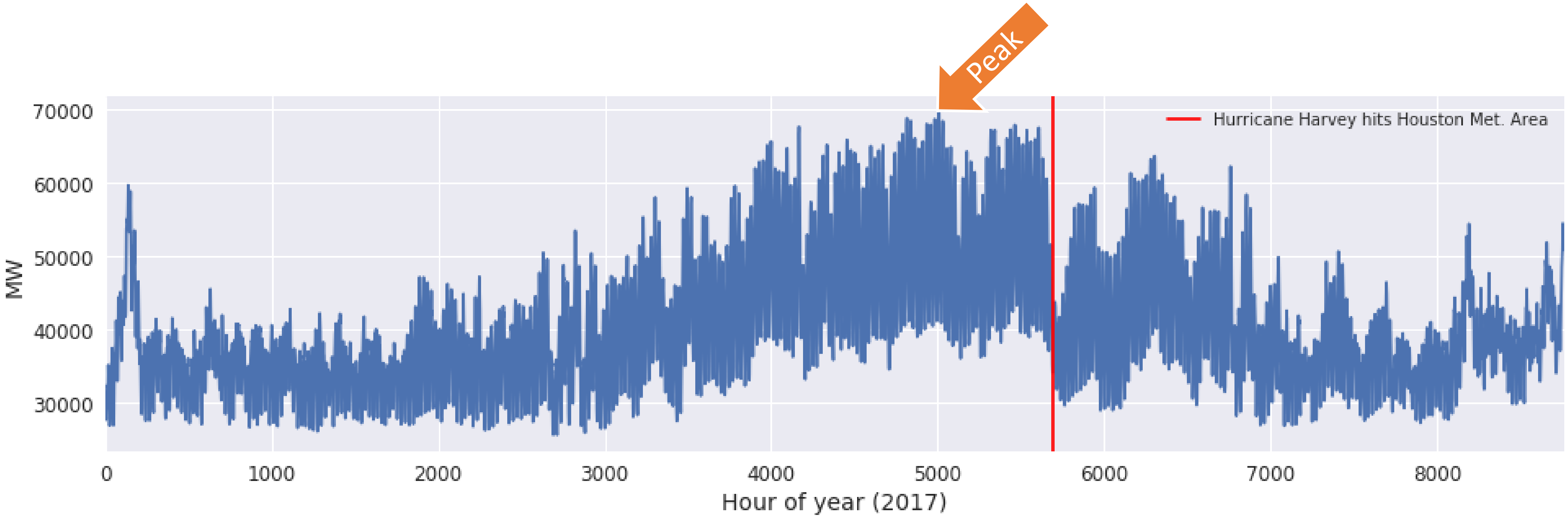
Consumers are incentivized to curtail demand during the moment of the CP



Variations

- Seasonal: UK, PJM, DEOK, winter ACS
- Monthly: ERCOT 4-CP, CAISO 12-CP
- Annually: "Peak Load Pricing" [Boiteux 1949]

Assumption #1:
1-CP pricing hour over a known, finite time horizon



Total hourly electrical demand in Texas, 2017

Current Solutions

Operators broadcast signals, e.g. Fort Collins PUD:

- Sends out signals about 10 days out of month
- Signals can come with less than one hour lead time, can last multiple hours
- Customers know when CP's should occur, e.g. hot day, afternoon

Too many signals, still hard to predict rare, non-causal events

Utilities/distributors   Large consumers

Core Assumptions

1. Single peak over known finite time period
   (No averaging of multiple peaks/time periods)
2. System noise is Gaussian, corresponding to forecast error

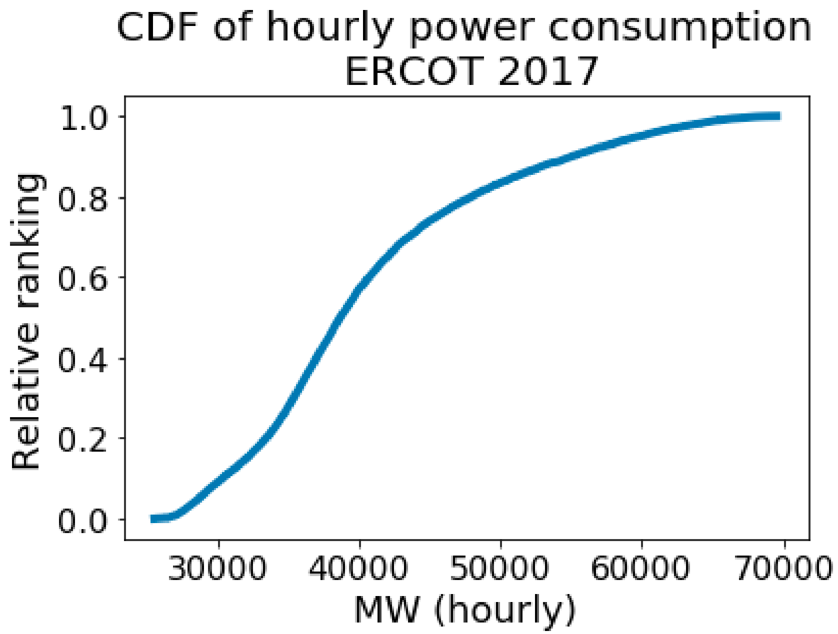
$$R = \sum_{t=1}^{T} g(x_t) - \pi_{cp} \left\{ x_{c^*} \middle| c^* = \underset{c \in T}{\operatorname{argmax}}\, s_c \right\}$$

Predicting Coincident Peaks

Can we do better than optimizing over Monte Carlo? (i.e. is more data going to help us?)

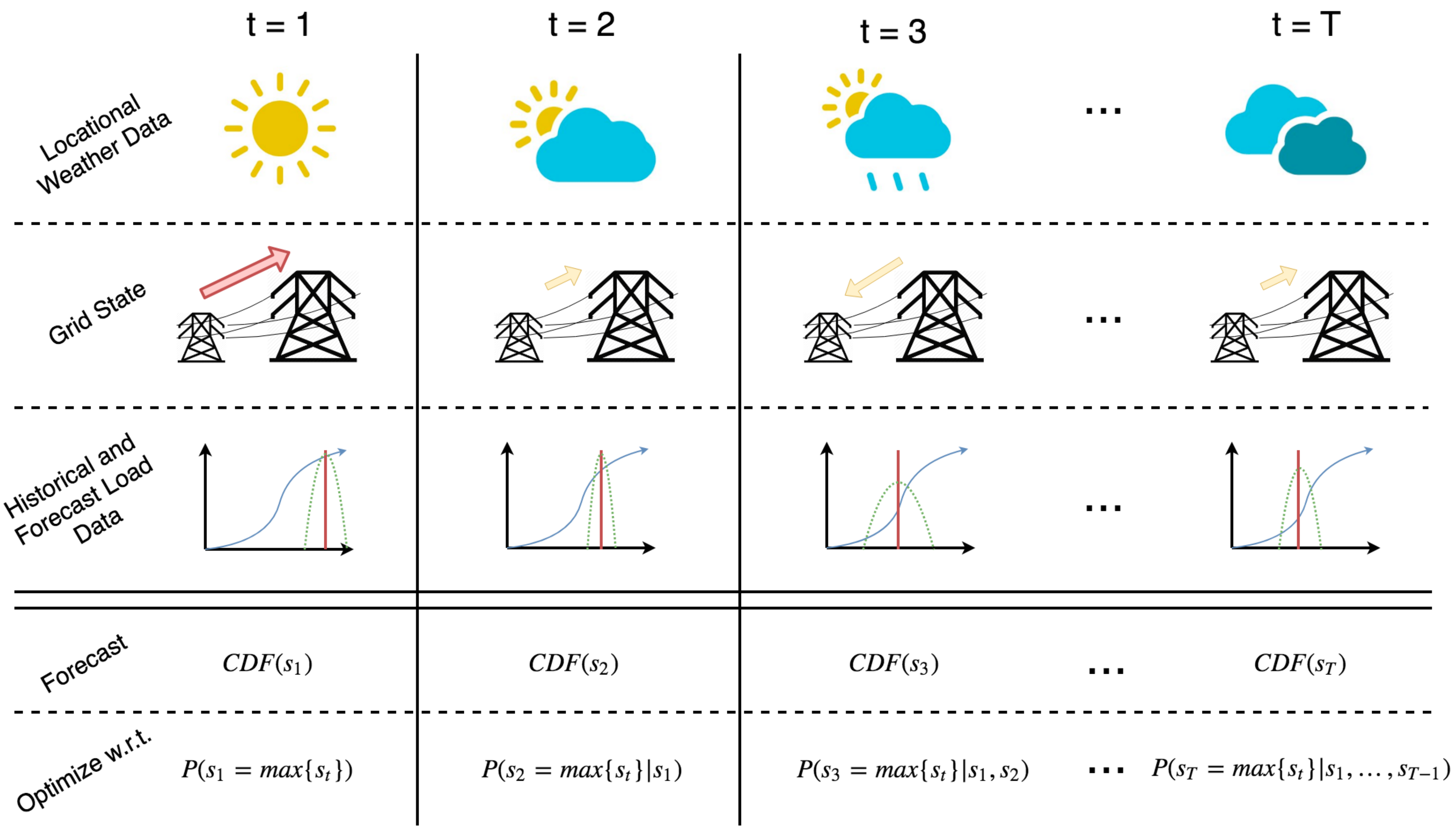
System operators are constrained to sending out early signals (> 24 hours)

Replace strict max operator with cumulative distribution function



$$\left\{ x_{c^*} \middle| c^* = \underset{c \in T}{\operatorname{argmax}} \, s_c \right\} \rightarrow \mathrm{CDF}(x_t) = P(X \leq x_t)$$

Dowling, C. P., Kirschen, D., & Zhang, B. (2018, November). Coincident Peak Prediction Using a Feed-Forward Neural Network. In *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)* (pp. 912-916). IEEE.

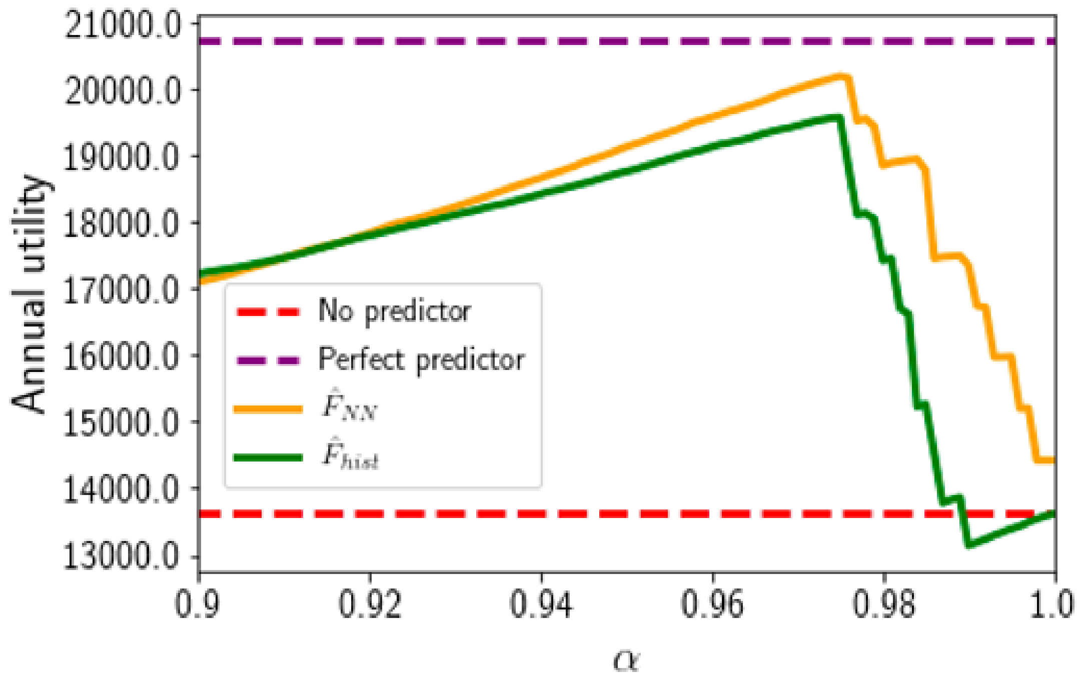

t = 1
t = 2
t = 3
t = T
Locational Weather Data
Grid State
Historical and Forecast Load Data
Forecast
Optimize w.r.t.
$CDF(s_1)$
$CDF(s_2)$
$CDF(s_3)$
…
$CDF(s_T)$
$P(s_1 = max\{s_t\})$
$P(s_2 = max\{s_t\}|s_1)$
$P(s_3 = max\{s_t\}|s_1, s_2)$
…
$P(s_T = max\{s_t\}|s_1, \dots, s_{T-1})$

Predicting Coincident Peaks

Hypothetical business, *very* simple NN as predictor

Curtail demand linearly up to some budget

Some traction to be gained predicting system peaks --- let's take a more principled approach



CDF Curtailment Threshold

Current Solution: Small Consumer Perspective

Responding to operator signals

40% CP consumers in ERCOT <10 MW

Consumer's CP timing determined by system noise (forecast is known) independent of their power demand at any time

Small Consumer Perpsective

System Noise

$$s_t \sim \mathcal{N}(0, \sigma^2)$$

Consumer Reward

$$R = \sum_{t=1}^{T} g(x_t) - \pi_{cp} \left\{ x_{c^*} \,|\, c^* = \operatorname{argmax}_{c \in T} s_c \right\}$$

Naive Solution

Ignore system noise; amortize coincident peak costs across all time periods

$$0 = T \cdot g'(x) - \pi_{cp} x$$

Proposed Solution: Small Consumer Perspective

- No ramping constraints
- Dynamic Programming

$$\underset{x_1, x_2, \ldots, x_T}{\text{maximize}} \quad \mathbb{E}[R_T]$$
$$\text{subject to} \quad x_t \in [0, \bar{x}]$$

- Ramping constraints
- Approximate dynamic programming

$$\underset{x_1, x_2, \ldots, x_T}{\text{maximize}} \quad \mathbb{E}[R_T]$$
$$\text{subject to} \quad x_t \in [0, \bar{x}]$$
$$x_t \in [x_{t-1} - \delta, x_{t-1} + \delta]$$

$$p_t := P(s_{t+1} \text{is peak for all } T | s_1, s_2, \ldots, s_t)$$

Dowling, C.P., Zhang, B. Mitigation of Coincident Peak Charges via Approximate Dynamic Programming IEEE Conference on Decision and Control, 2019 (to appear)

Dynamic Programming

Optimize going backwards in time, let t = T-1

$$\mathbb{E}_{s_T}[R] = \mathbb{E}_{s_T}\left[\sum_{t=1,\dots T-1} g(x_t) + g(x_T) - \pi_{cp}\{x_{c^*}|c^* = \underset{c\in T}{\text{argmax}}[s_c]\}\right] \quad (1)$$
$$= \sum_{t=1,\dots T-1} g(x_t) + g(x_T) - \pi_{cp}[(1-p_T)x_{c^*} + p_T x_T] \quad (2)$$

And we optimize w.r.t to x_T . Continuing backwards, we have that the optimal play for any t is x_t such that

$$0 = g'(x_t) - \pi_{cp} p_t$$

Adding Ramping Constraints

If we add a ramping constraint $x_t \in [x_{t-1} - \delta, x_{t-1} + \delta]$ then we have that,

x'_t solves $0 = g'(x_t) - \pi_{cp} p_t$

The optimal $x^*_t \in [x_{t-1} - \delta, x_{t-1} + \delta]$

and minimizes $|x'_T - x^*_T|$

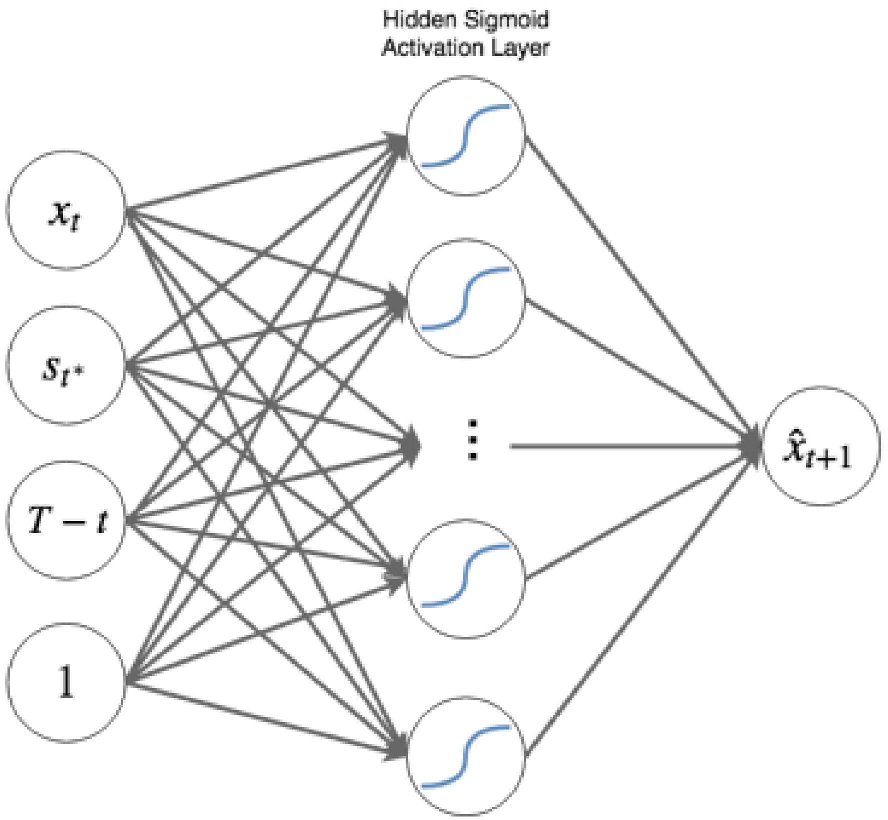
$$\mathbb{E}[R_T] = \mathbb{E}_{s_1}\left[g(x_1) - \pi_{cp} p_1 x_1 + \mathbb{E}_{s_2}\left[g(x_2) - \pi_{cp} p_2 x_2 \ldots \; + \mathbb{E}_{s_T}\left[g(x_T) - \pi_{cp} p_T x_T\right]\right]\right]$$

Approximate Dynamic Programming

At each time t, sample paths amongst ramp-constrained options using known forecast error distribution

Typically this Monte Carlo path sampling procedure chooses the best path

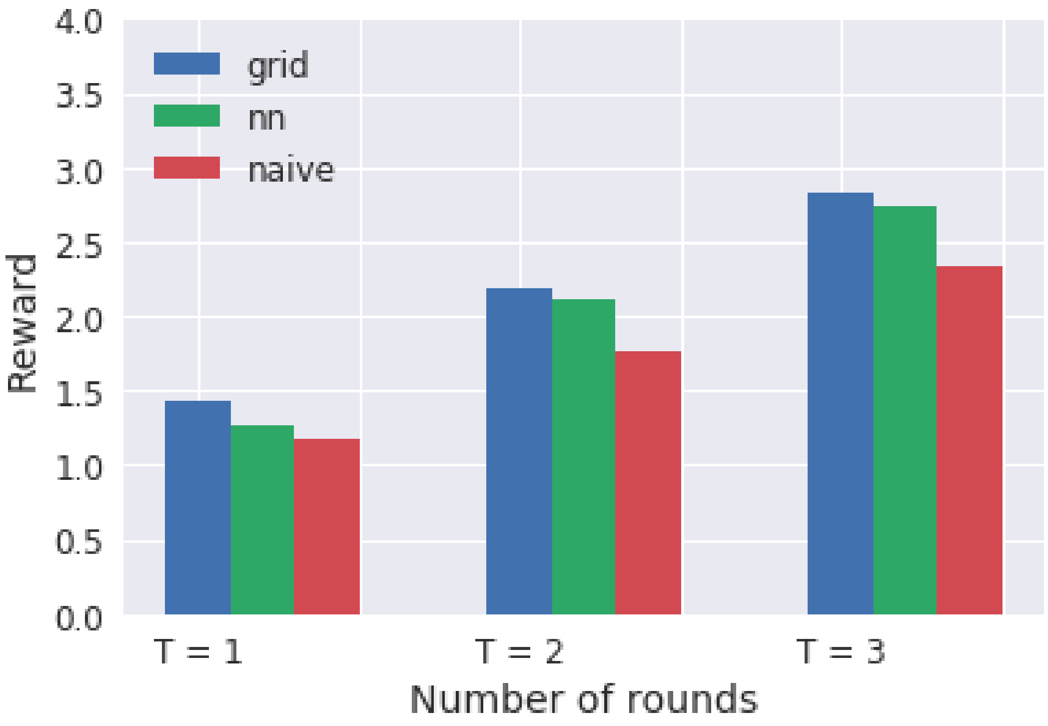
We use realizations to train deterministic policy to choose approximately optimal plays



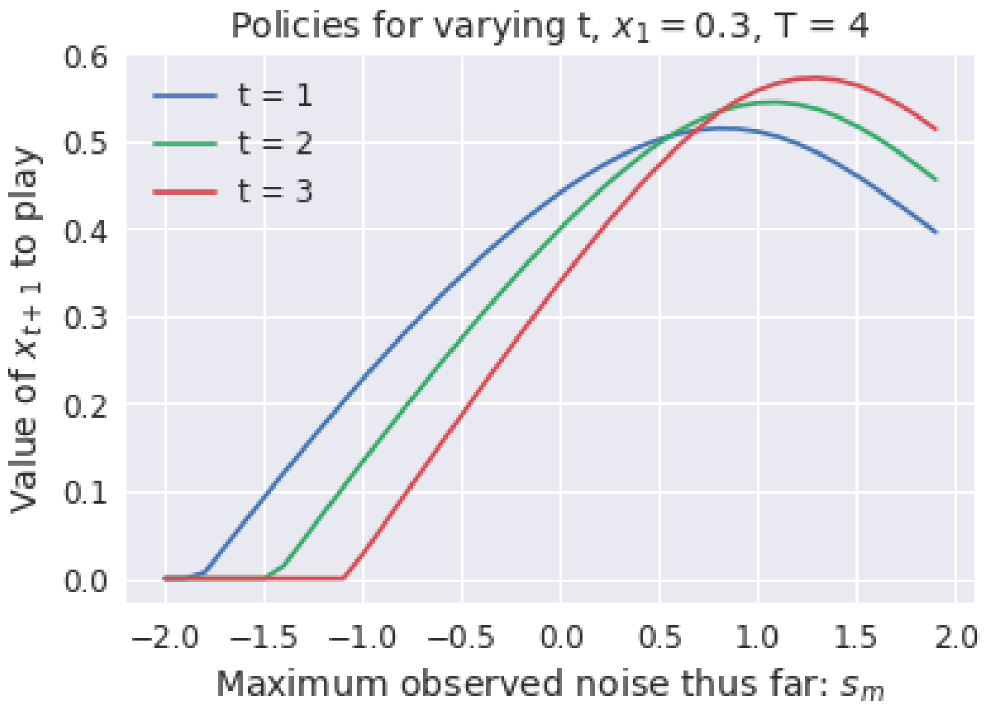
Approximate Dynamic Programming

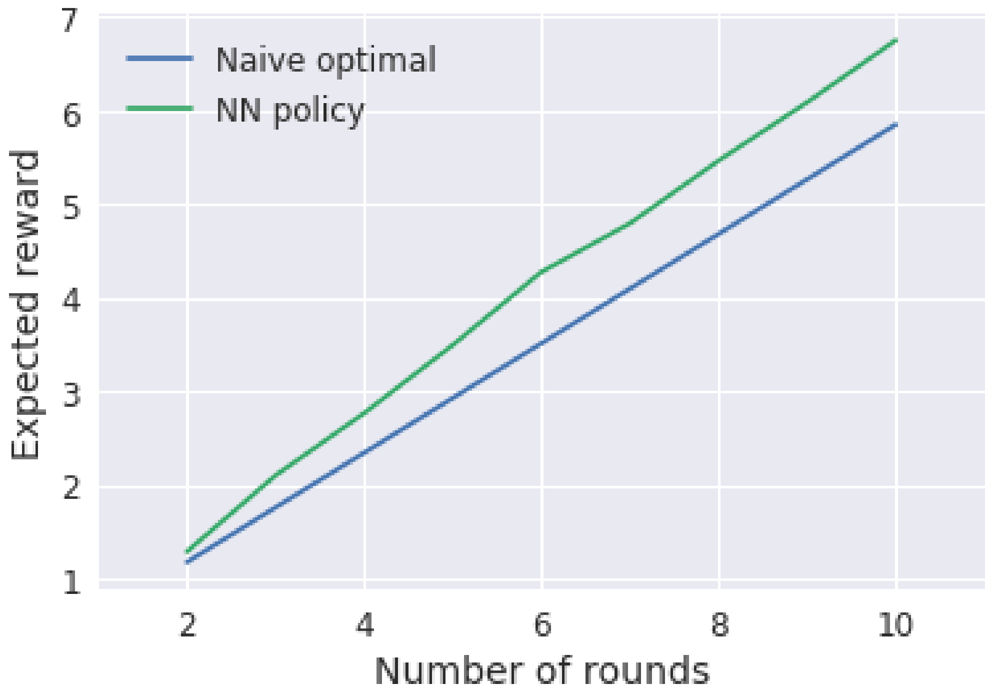
Utility function: $g(x_t) = 2\log(1 + x_t^2)$

For small number of rounds we can brute force grid search to ensure a deterministic policy learned from Monte Carlo sampled paths approaches the true optimal solution



Approximate Dynamic Programming





Small Consumer Perspective

System Noise

$$s_t \sim \mathcal{N}(0, \sigma^2)$$

Consumer Reward

$$R = \sum_{t=1}^{T} g(x_t) - \pi_{cp} \left\{ x_{c^*} | c^* = \operatorname{argmax}_{c \in T} s_c \right\}$$

Large Consumer Perspective

System Noise

$$s_t = \sum_{j=1}^{N} x_t^{(j)} + \epsilon_t$$

Consumer Reward

$$R_{(i)} = \sum_{t=1}^{T} g_{(i)}(x_t^{(i)}) - \pi_{cp} \left\{ x_{c^*} \,|\, c^* = \operatorname{argmax}_{c \in T} s_c \right\}$$

Current Solution: Large Consumer Perspective

Studies have suggested 4% peak reduction efficacy [Zarnikau 2013]

Many large consumers lack flexibility, are highly correlated

- Increasingly diverse energy products in deregulated markets
- Increasingly flexible grid; what happens when large consumers try to learn an optimal policy for curtailment during system peak?

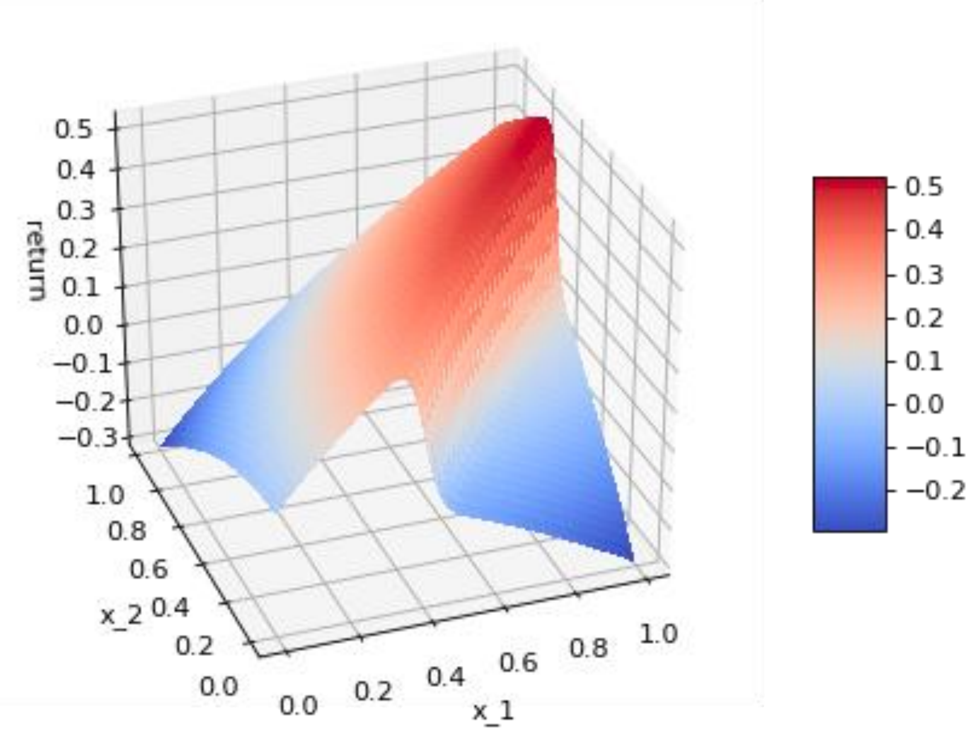
Large Consumer Perspective

1. Now a game theory setting (Cournot competition); consumer choices impact all other consumers' rewards

2. Not concave game

3. No obvious potential function

4. Need to iteratively play game & learn from results (a multi-agent RL problem)



Two player, two round game, fixed choice of plays for opposing player

Large Consumer: Guided Policy Gradient

Initialize player policies: $\phi_{(i)}(x_t^{(i)}, \max\{s_1, \ldots s_t\}, \bar{s}_{t+1}, T - t, 1) = x_{t+1}^{(i)}$

Policy Gradient Procedure:

For epochs:

-Realize game sequence over T

Ignoring non-concavity

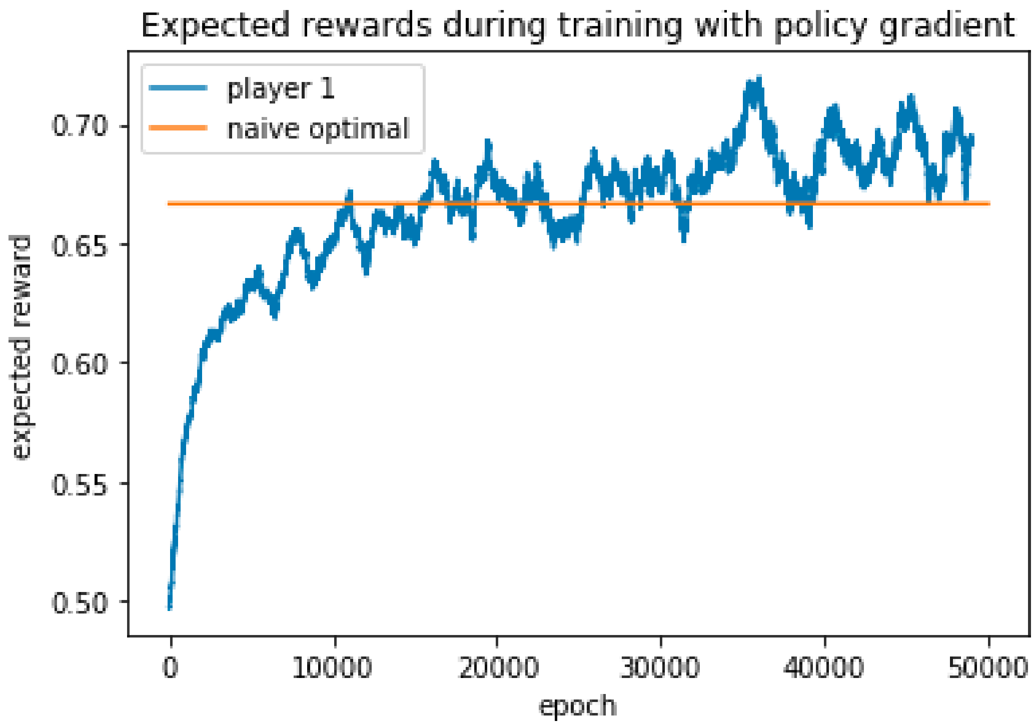
-For each player compute: $\hat{x}_t^{(i)} = x_t^{(i)} + \eta \left(\frac{\partial R}{\partial x_t^{(i)}} \right)$

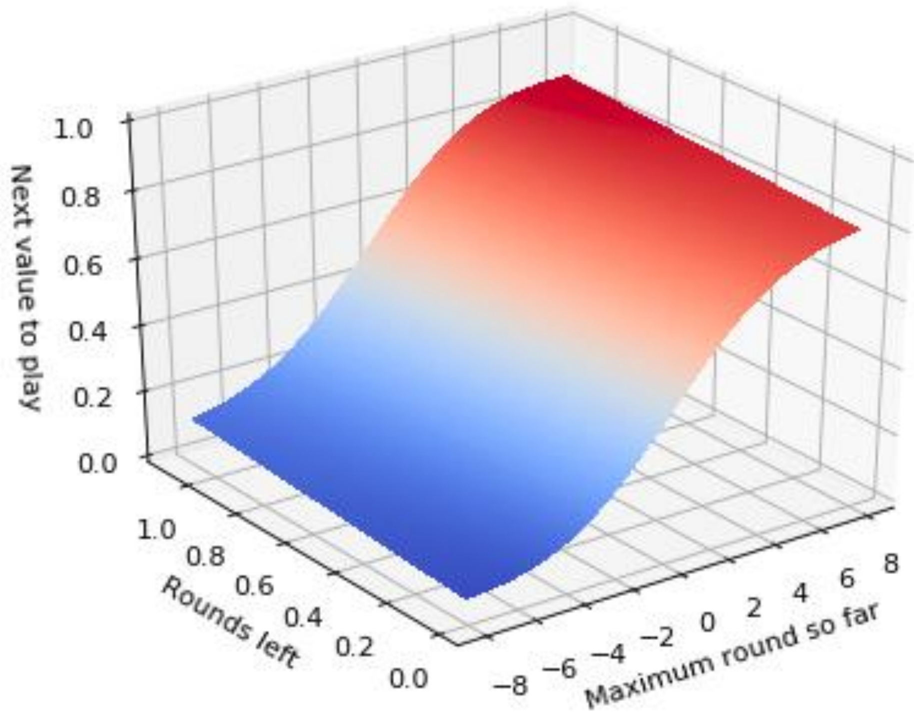
-Gradient descent on new plays: $\mathcal{L}\left(\phi_i(x_t^{(i)}), \phi_i(\hat{x}_t^{(i)})\right)$

Single Player Guided Policy Gradient

No access to prediction of next round

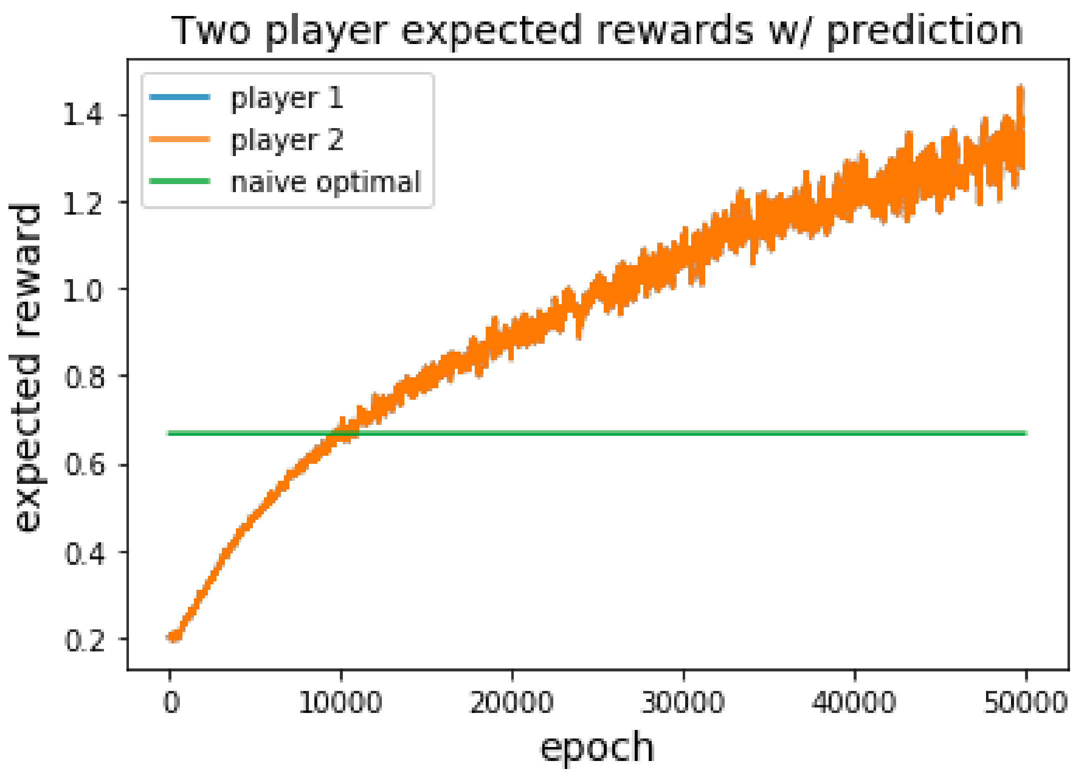
All utility functions: $g_t = \log(1 + x_t)$





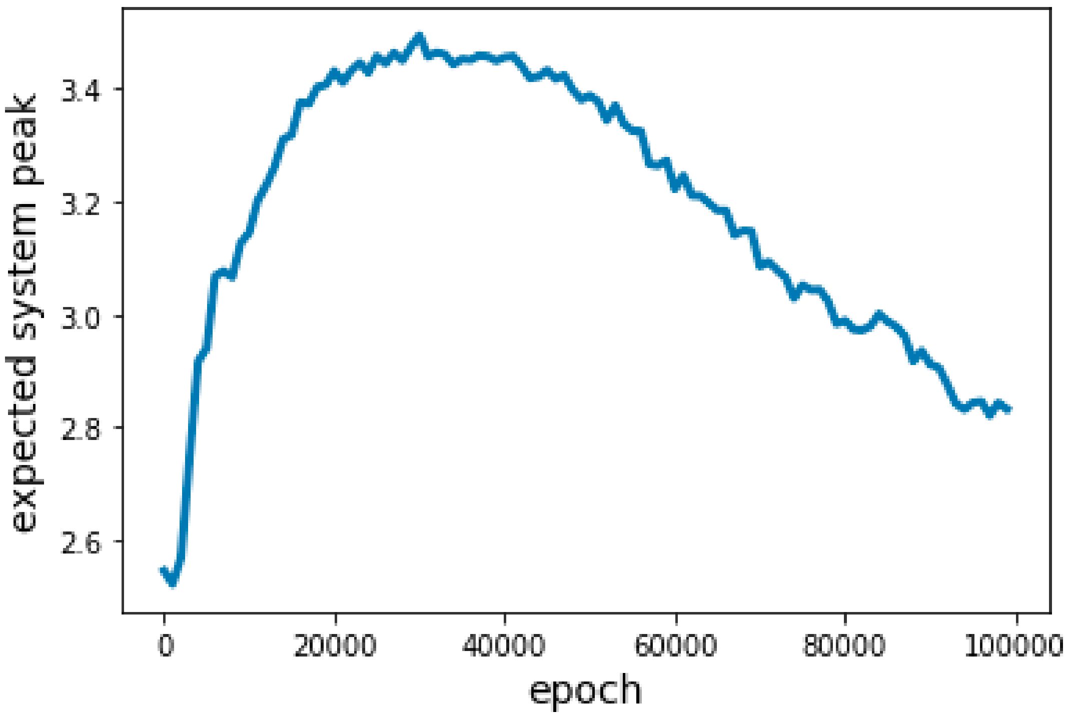
Multi Player Guided Policy Gradient

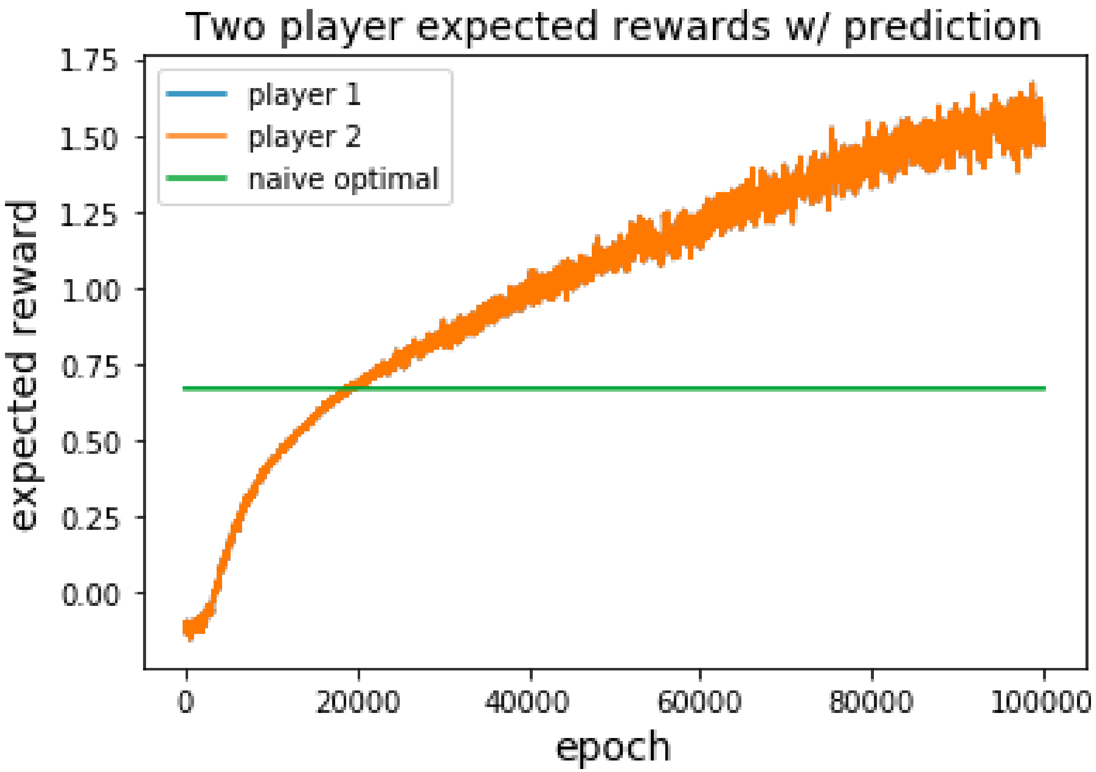
Identical utility functions



Multi-Player Guided Policy Gradient

Identical utility functions



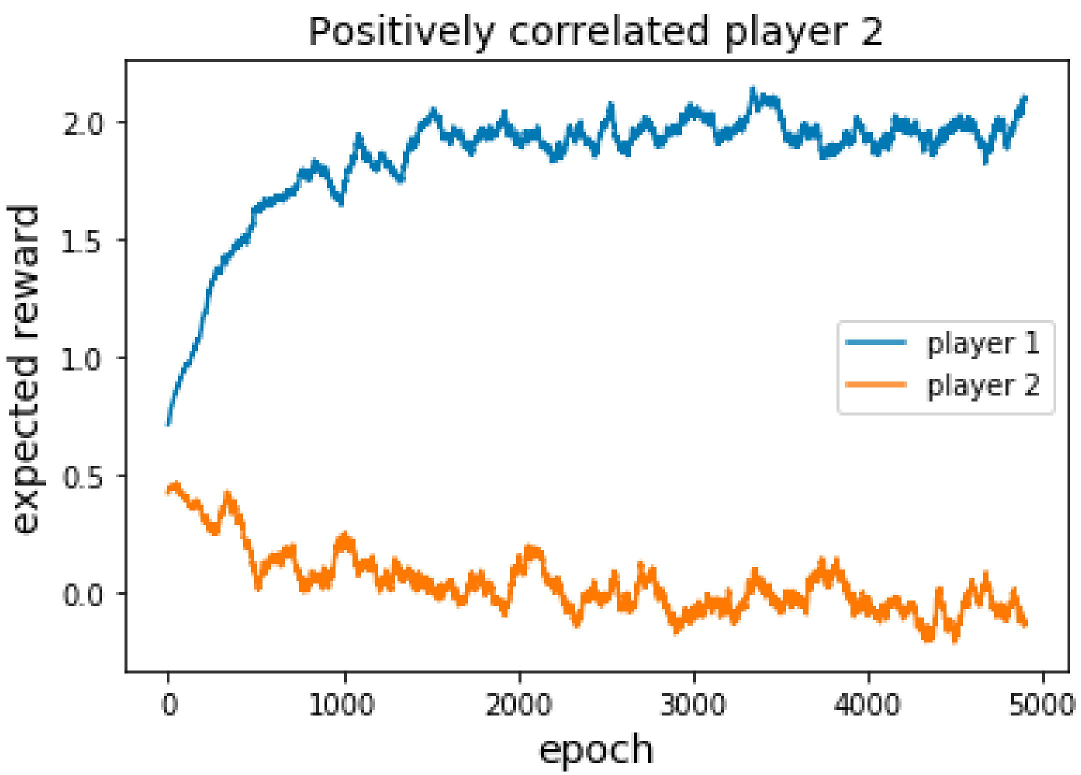


Multiple Correlated Players

- Player 1 independent, player 2 positively correlated (i.e. stochastic function of) player 1

$$x_t^{(2)} = x_t^{(2)} + \alpha \cdot \mathrm{Unif}(0,1) \cdot x_t^{(1)}$$

- Both players have access to noisy predictions
- Large consumers are strongly correlated in markets that currently use CP pricing



Electrical Grids: Summary

1. Simple machine learning methods can combine data sources to predict coincident peaks.
2. In an increasingly flexible grid, consumers are incentivized to respond to coincident peak price signals.
3. Coincident peak pricing games provide a mechanistic means to analyze pricing as grid flexibility evolves and predictors fail.

Buildings: Learning Transferable Fault Detectors

1. How is ML enabling energy management in buildings? What data is available?
2. Can we detect HVAC faults in an unsupervised setting.
3. The fault detector is transferred to deal with the scarcity of labeled data.

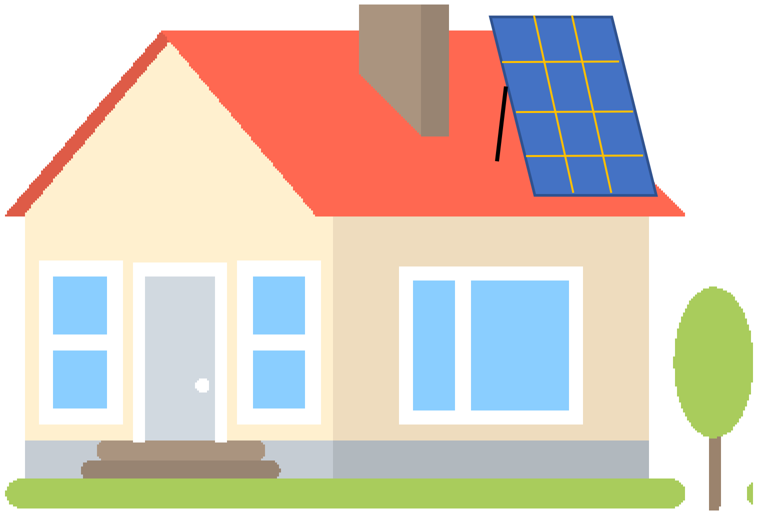
Smart Buildings

1. Smart-meters that can communicate with the home's electrical utility

2. Consumer power generation

3. On-site storage from home battery or electric vehicle

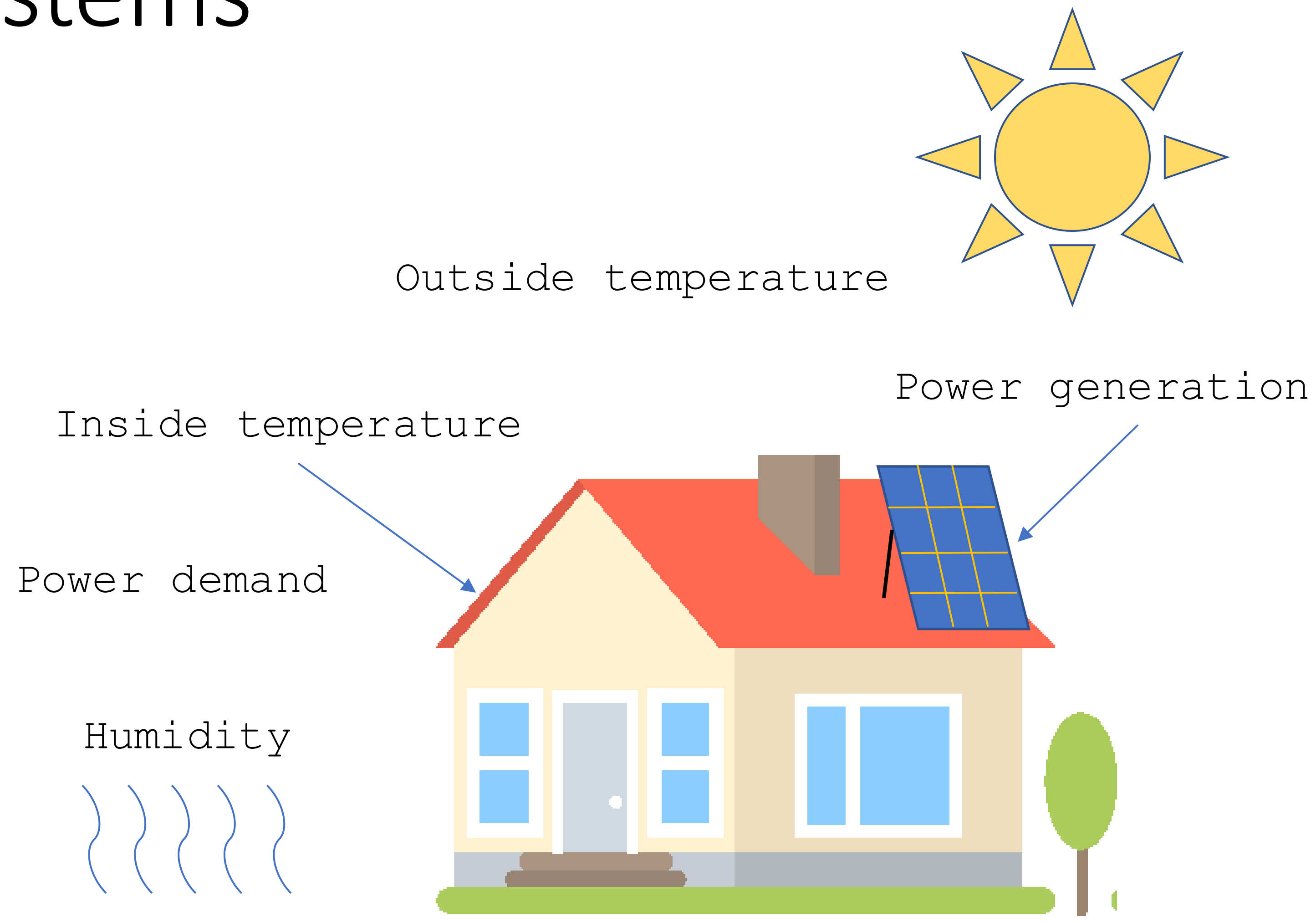
4. Energy-conserving behind the meter devices like programmable thermostats and occupancy-based home energy management



Modeling HVAC Systems

Inputs: $s_1, s_2, \dots s_T$

Outputs: $x_2, x_3, \dots x_{T+1}$



State Transition Model

$$x_{t+1} = As_{t+1} + \epsilon_t$$

Fault Detection

When something breaks in the HVAC system, A is no longer an accurate model, two probabilities:

Operational	Faulty
$P(x_{t+1} \mid A, s_t)$	$P(x_{t+1} \mid \tilde{A}, s_t)$

Matrix normal prior on $\tilde{A}$ to derive a classification rule:

$$0 \lessgtr \mathrm{Tr}\left[(x - As)^T (x - As)\right] - \mathrm{Tr}\left[xx^T + AA^T - C^{-1} D^T D\right] - p \log(|C^{-1}|)$$

$$C := (ss^T + I)$$
$$D := (xs^T + A)$$

Dowling, C.P., Zhang, B. Transfer Learning for HVAC System Fault Detection American Control Conference (2020) (in review)

Naïve Fault Detection

The classifier does not depend on the modality of the fault, only on the true state transition model A

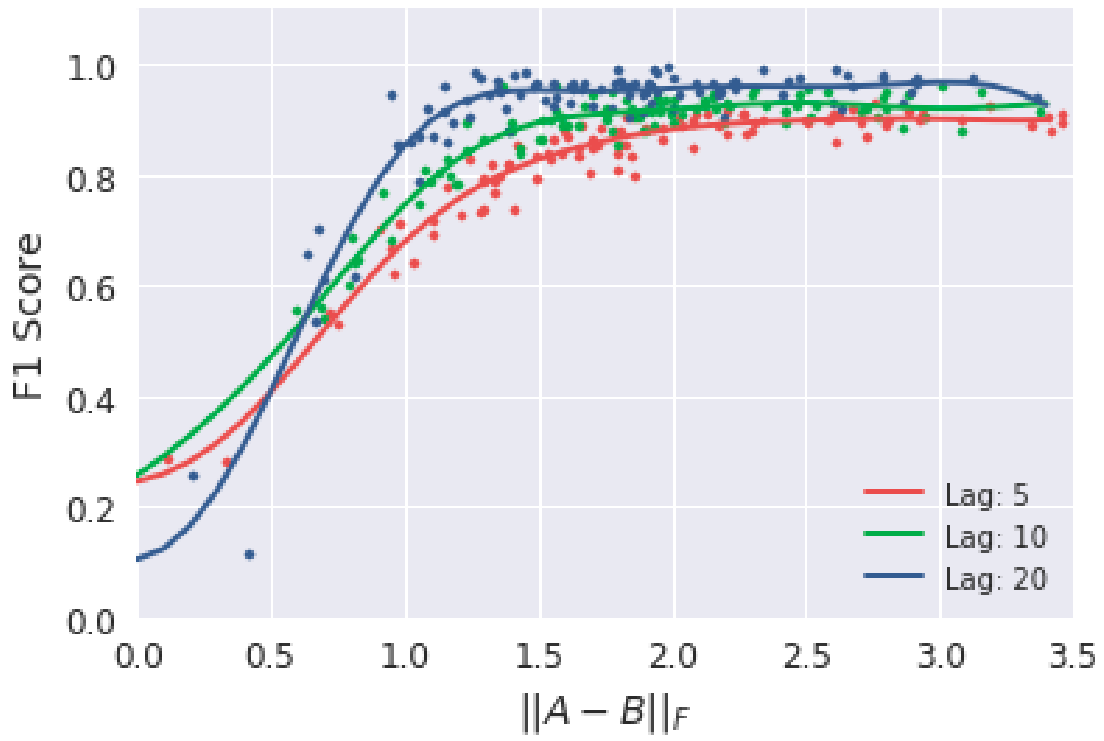
Normal Operations

$$x_{t+1} = As_{t+1} + \epsilon_t$$

Faulty Operations

$$y_{t+1} = Br_{t+1} + \epsilon_t$$

As A and B diverge, classification accuracy increases



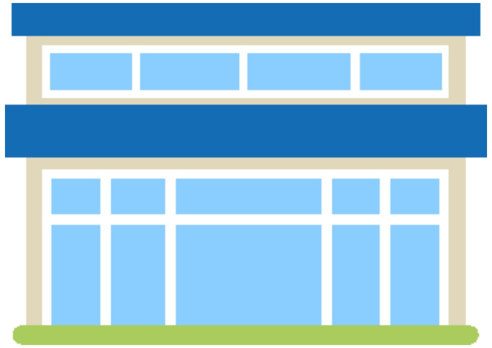
Transfer Learning

Transfer learning has been used successfully in things like image classification

Buildings have the same thermodynamic properties

Here we'll learn a model A on a building with lots of sensor data, and for a building with less sensor data, use model A as a starting point

Lots of sensor data



Limited sensor data

Transfer Learning

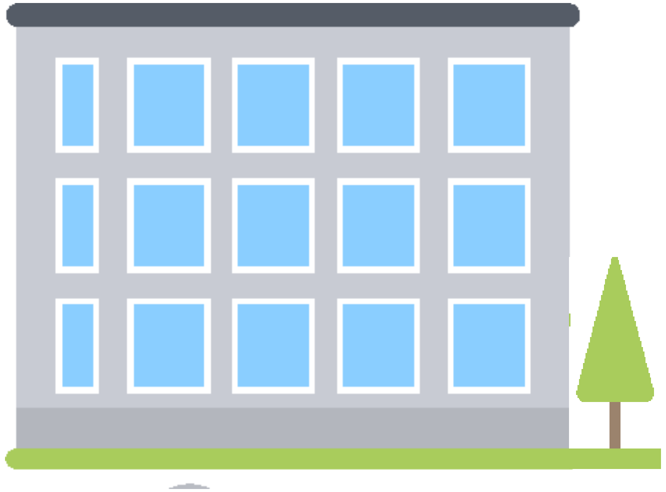
Simulated Building

- 3-story, ~50k sq ft office building
- Cool, wet climate in Seattle

Real Building

- 2 story, ~25k sq ft office building
- Dry, arid climate in Eastern Washington

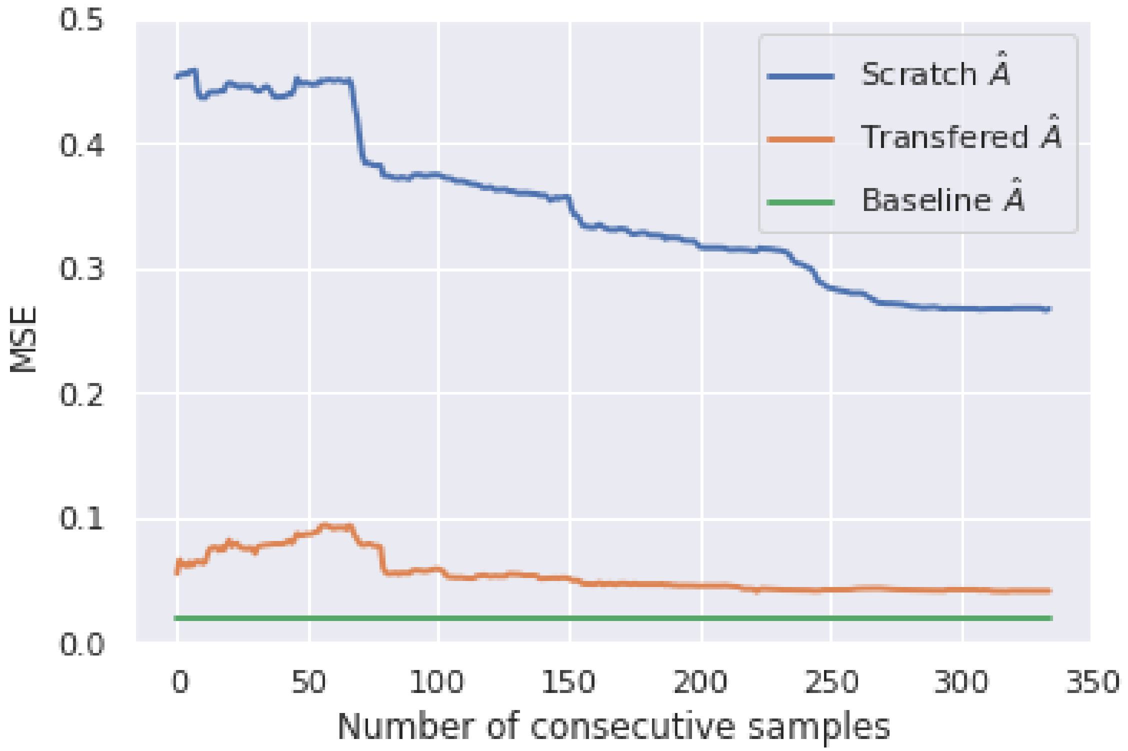
Learn model A with lots of samples → Weighted LS → Learn model C using A as starting point



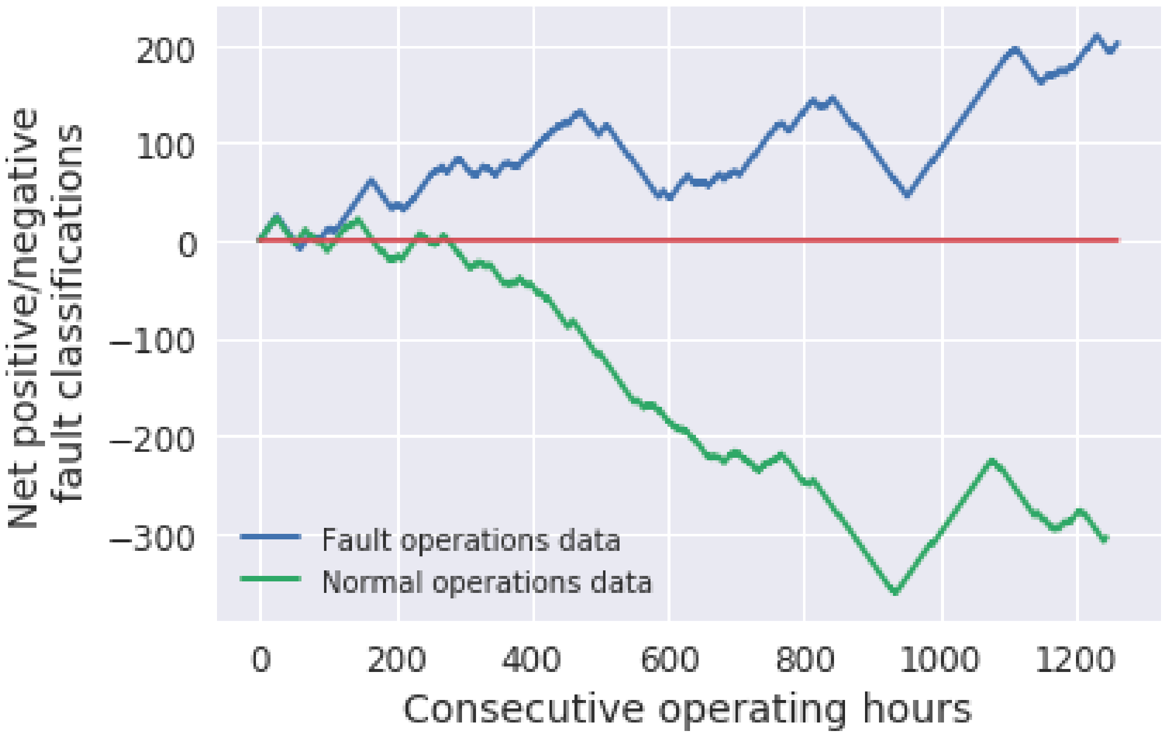
Systems Engineering Building, PNNL, Dong, J. et al. [2019] "Online Learning for Commercial Buildings"

Transfering Fault Detection

Loss of Model C on real building



Classification performance trained on 2 weeks of data

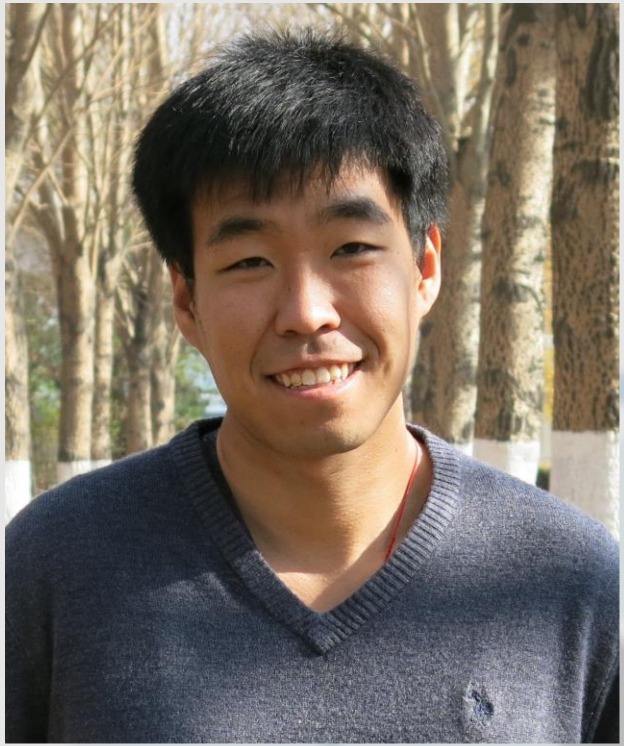


Buildings: Summary

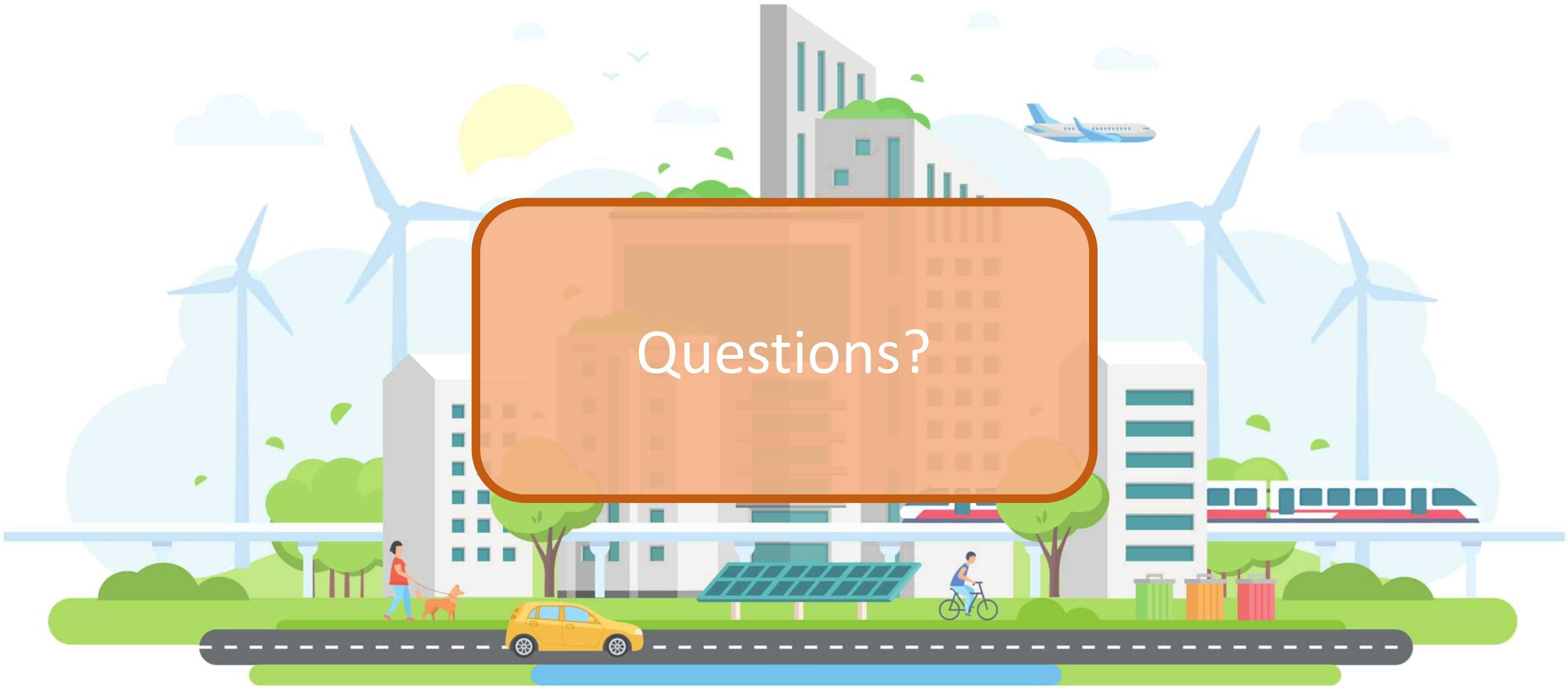
1. Machine learning methods can be used to detect faults in HVAC systems
2. Many, many buildings do not have access to large amounts of labeled sensor data to train ML models
3. Transfer learning using a Bayesian classifier can be used to learn ML models for buildings with access to limited, unlabeled sensor data

Future Work

- SDOT taking over curbside parking research to analyze price effects
- Continuing electrical market and HVAC system modeling work at PNNL starting in January




TRICAL ENGINEERING
RESTROOMS



Questions?

Extra Slides: Curbside Parking

Unique positive root of polynomial for total arrival rate in network of M/GI/k/k queues

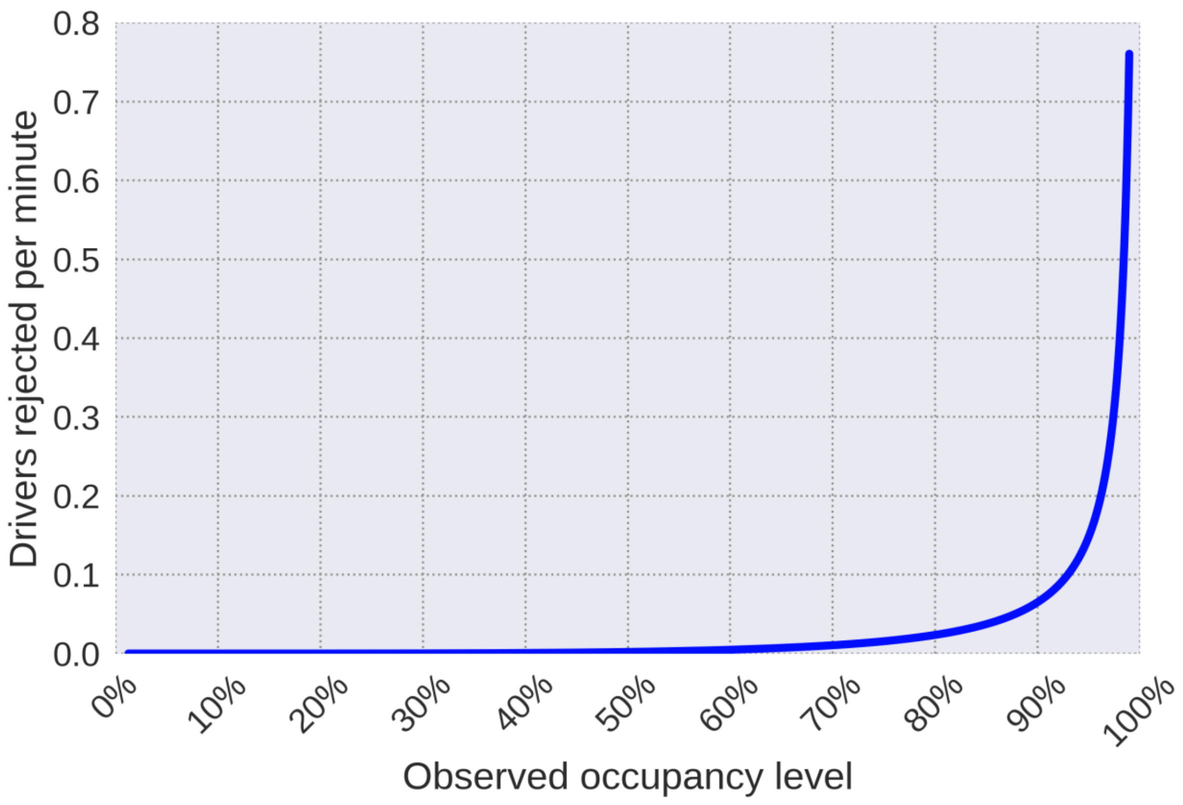
$$0 = \sum_{i=0}^{k} \frac{1}{\mu^{i-1}} \left[\frac{i - uk}{i!}\right] y^k$$

Probability of single block-full

$$\pi_k = \frac{y^k}{\mu^k k!} \cdot \left[\sum_{j=0}^{k} \frac{y^j}{\mu^j j!}\right]^{-1}$$

Rejection rate

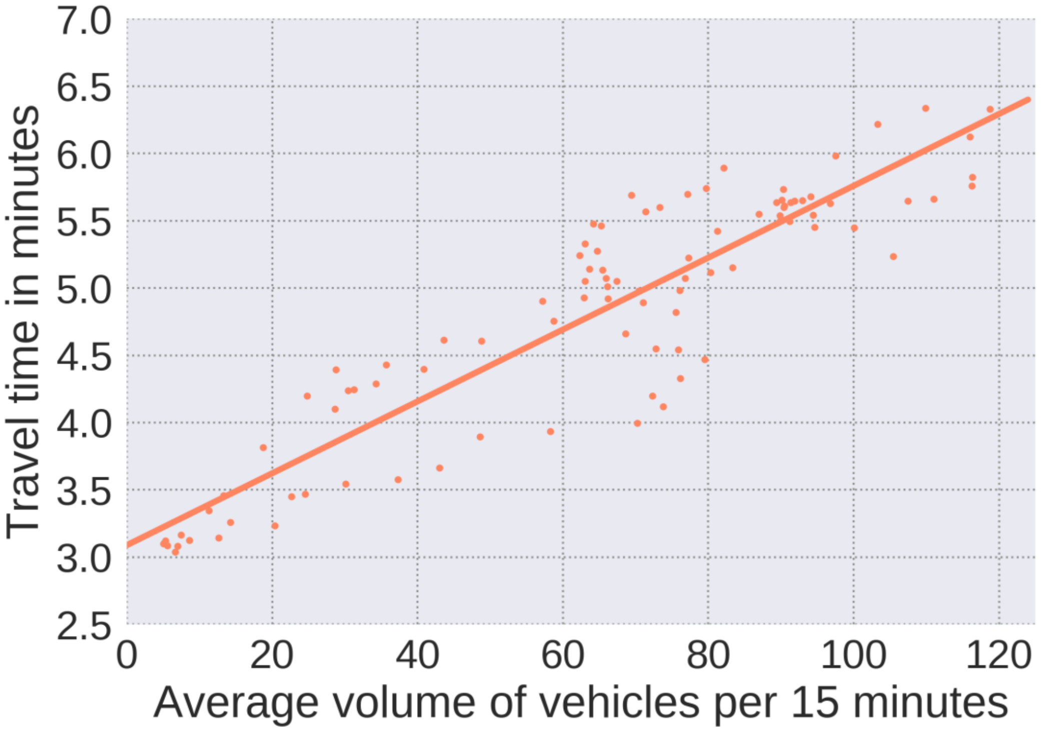
$$x = y\pi_k$$

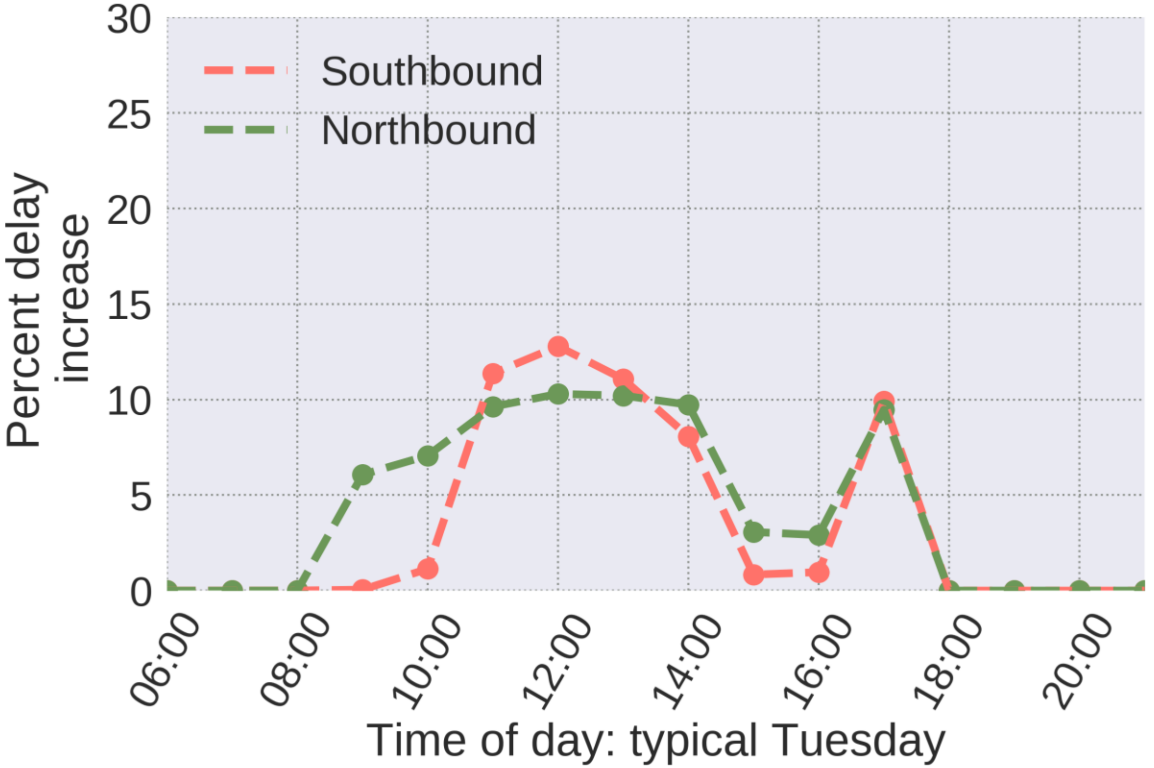


Rate of vehicle rejection as a function of observed occupancy in data

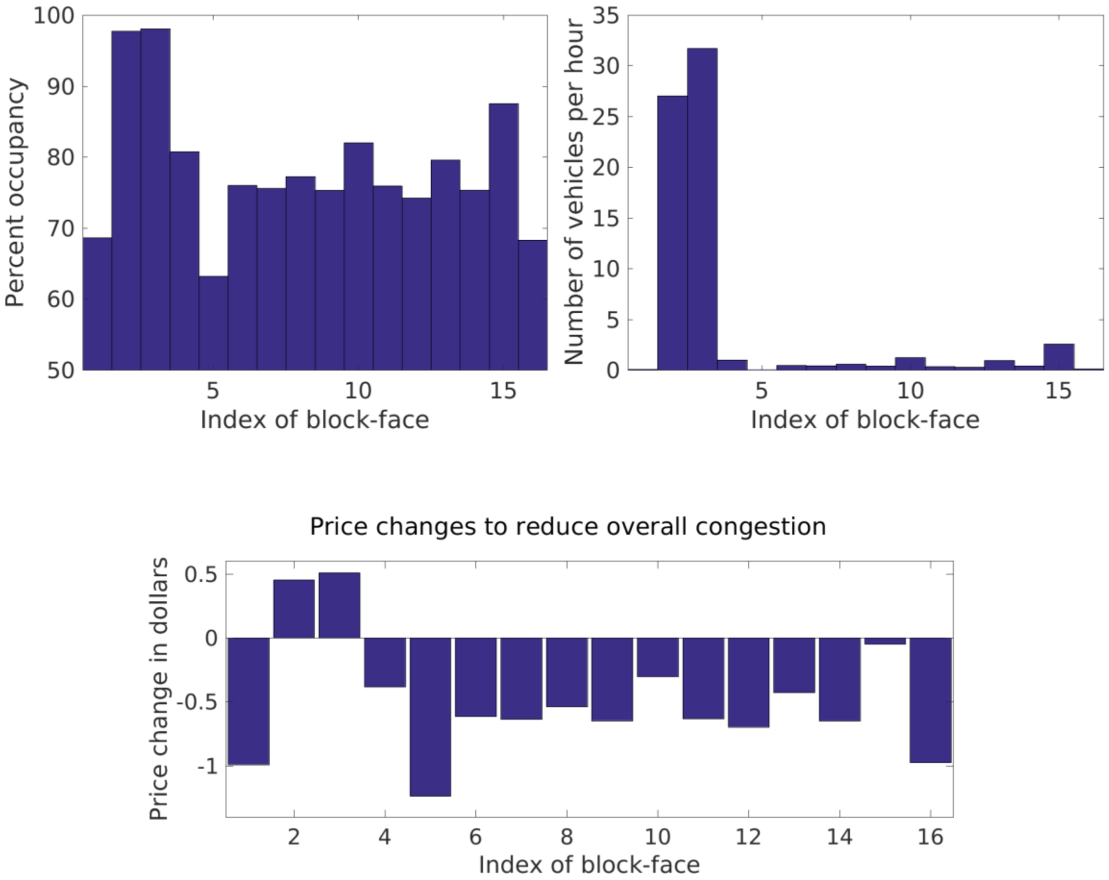
Delays Due To Congestion

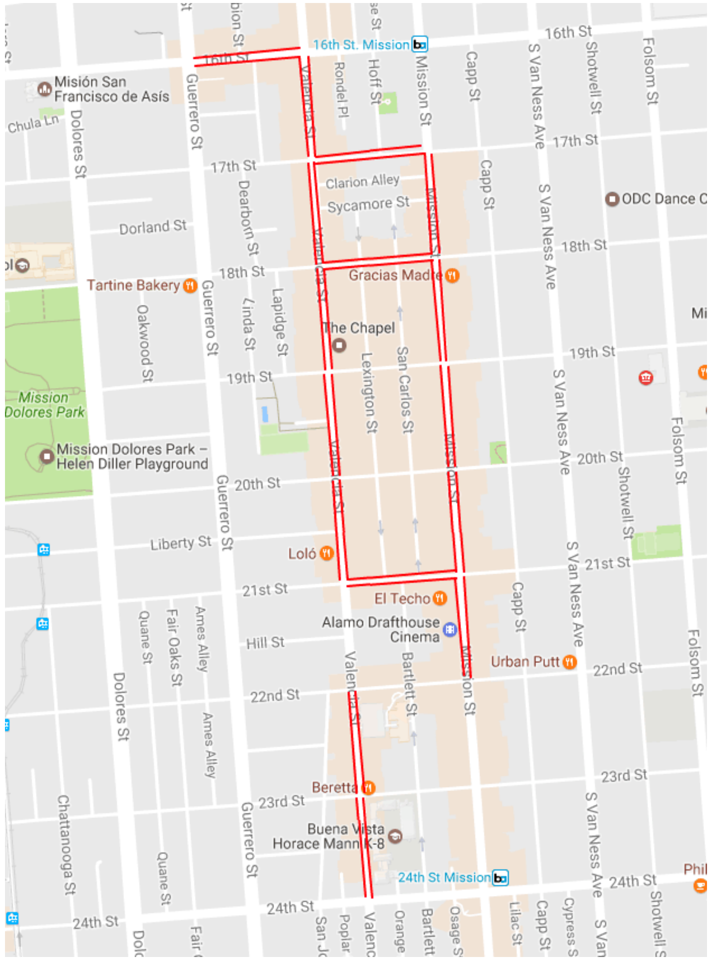
We use Google Maps travel-time data along 1st Ave to learn a relationship between vehicle volume and travel time.





Optimizing Occupancy/Congestion





Dowling, C., Fiez, T., Ratliff, L., & Zhang, B. (2017, December). Optimizing curbside parking resources subject to congestion constraints. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)* (pp. 5080-5085). IEEE.

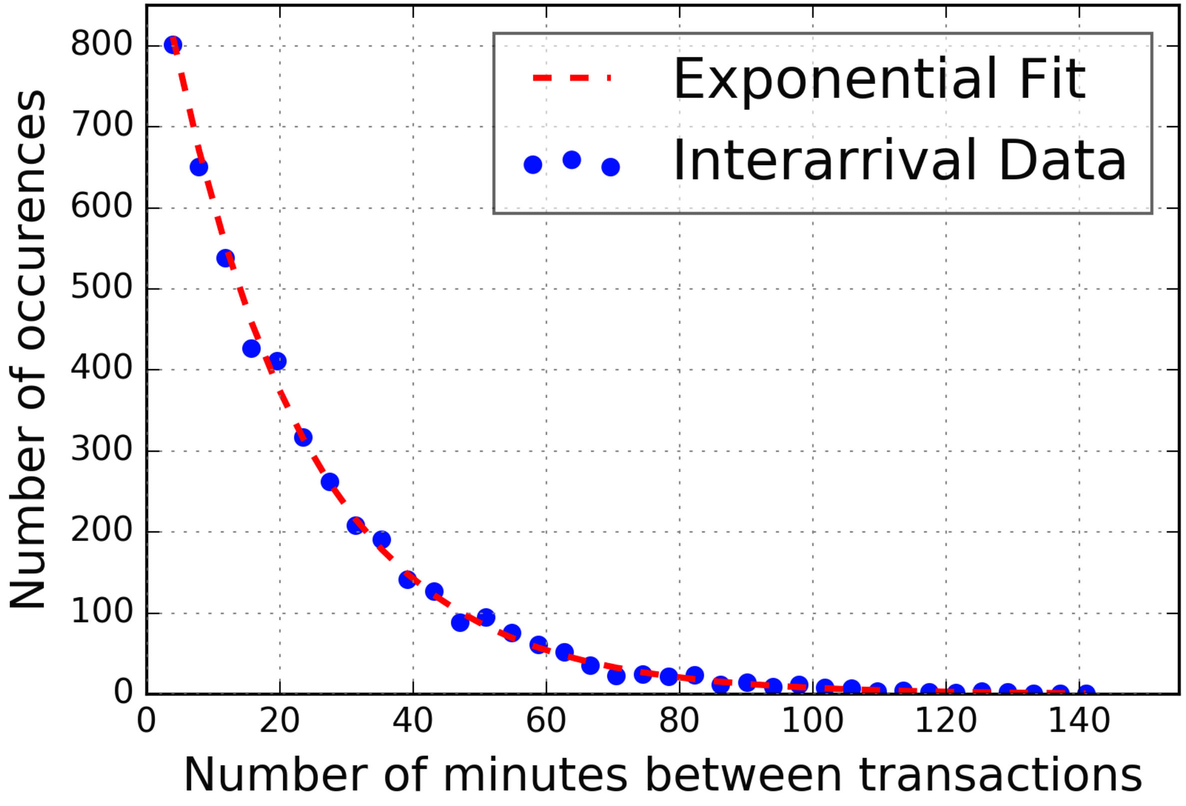
M/GI/k/k queue network assumptions

1. Transactions are representative of actual parking times [Qian 2017]

2. Inter-transaction start times exhibit exponential distribution --- network-wide inter-arrival rate is Poisson

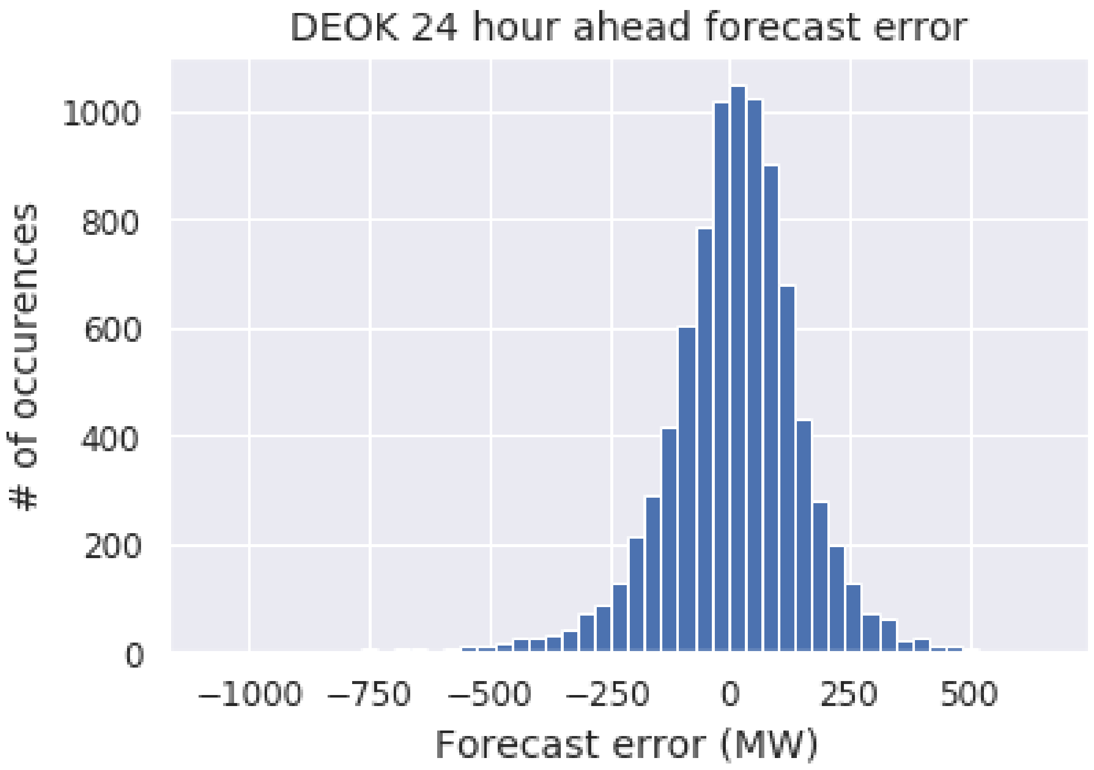
3. Little's Law does not depend on the distribution of service time, only stationarity

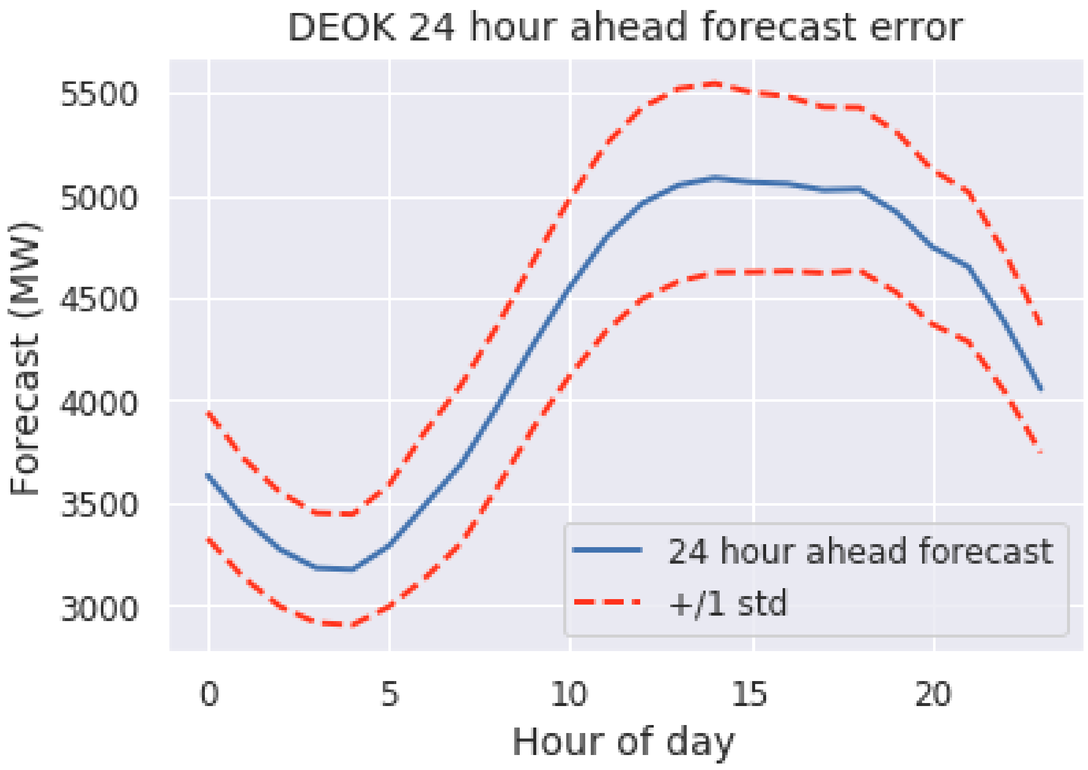
4. To satisfy *hourly* stationarity total network arrival rate is less than total network service rate and conditional dependence between block-face occupancies (desired assumption, independence not justified)



Coincident Peak Timing

Assumption #2:
Noise in the system is Gaussian, 0 mean





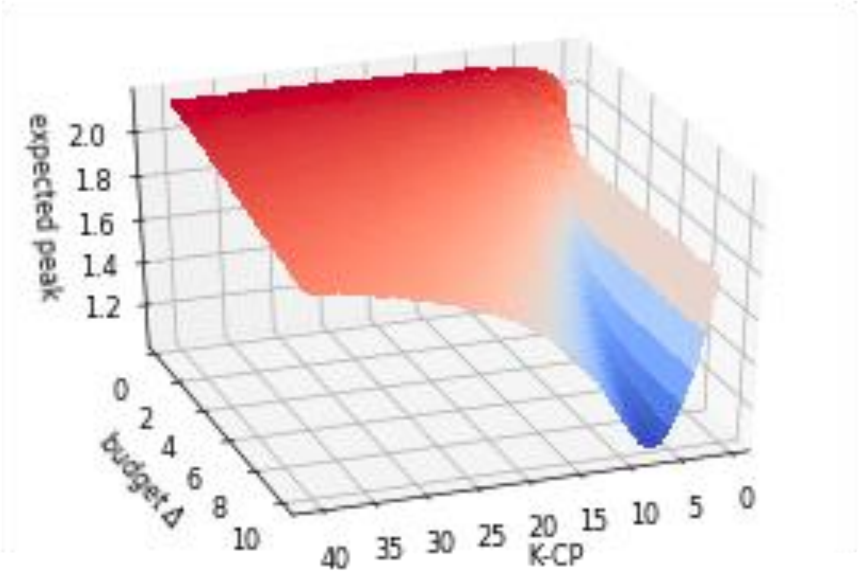
July 8, 2018

Coincident Peak: Order Statistics

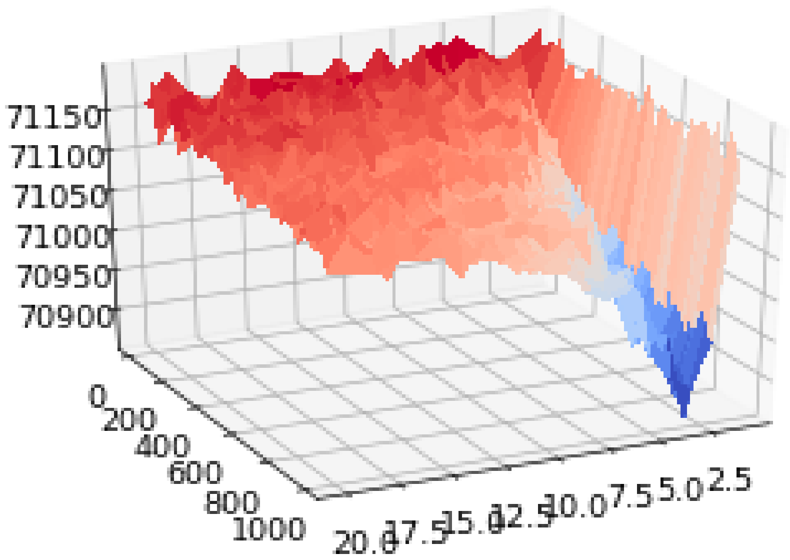
A limited number of CP billing periods yeilds the best peak reduction regardless of budget

For a total budget M, reduce top K CP by K/M

X ~ N(0,1) , T = 40



ERCOT August 2018 Peak Days , T = 40



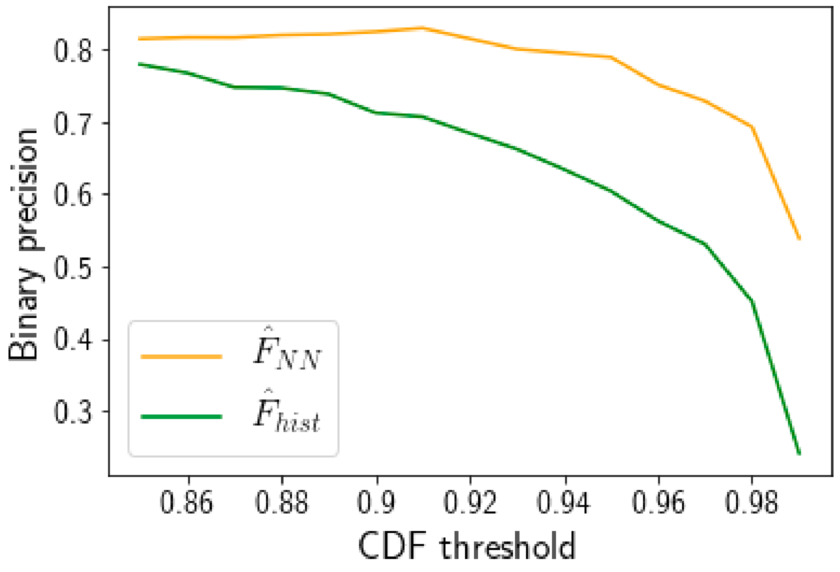
Predicting Coincident Peaks

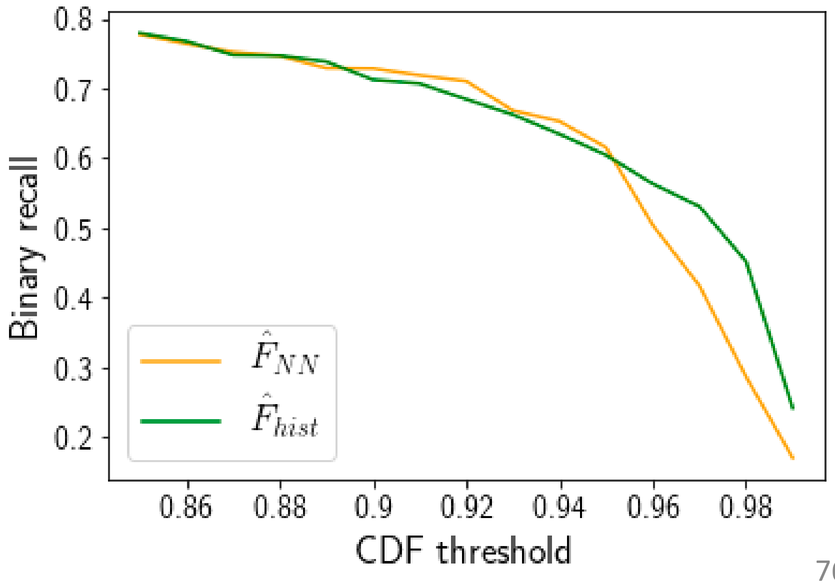
- Train a simple, feedforward NN to predict CDF output of next 24 hours system demands
  - Exponentially weighted L1 loss

$$F(s_{t+1}) = \begin{cases} 1 & CDF(s_{t+1}) > \alpha \\ 0 & \text{otherwise} \end{cases}$$

- Training data: weather, transmission, hourly demand ERCOT 2010-2016

- Test data: ERCOT 2017





Probability of a Coincident Peak

Can do better than optimizing over Monte Carlo. Can we do better than trying to forecast CDF and arbitrarily hedging?

We know forecast error is a unimodal distribution, then probability of a peak directly:

$$p_t := P(s_{t+1} \text{is peak for all T} | s_1, s_2, \ldots, s_t)$$

If IID

$$p_t = P(t+1 \text{is max of any} T-t) \cdot P(\text{any next} T-t > s_m) = \frac{1}{T-t}(1 - P(s \leq s_m)^{(T-t)})$$

Kernel Regression

Polynomial Basis

$$\phi_d(s) = [s^d, s^{d-1}, \ldots, s^1, \mathbf{1}] \qquad s_t \in \mathbb{R}^p$$

Least Squares Regression

$$\hat{A}_1 = \underset{W}{\operatorname{argmin}} ||WS^{(1)} - X^{(1)}||_2^2$$

Matrix Normal Prior

$$P(Z|M,U,V) = \frac{1}{(2\pi)^{np/2}|V|^{n/2}|U|^{p/2}} \cdot e^{-\frac{1}{2}\mathrm{Tr}\left[V^{-1}(X-M)^T U^{-1}(X-M)\right]}$$

Click to add text

$$P(x_{t+1}|A, s_t) = \frac{1}{(2\pi)^{n/2}} e^{-\frac{1}{2}\left[(x_{t+1}-As_t)^T(x_{t+1}-As_t)\right]}$$

$$P(x_{t+1}|\tilde{A}, s_t) = \int \frac{1}{(2\pi)^{n/2}} e^{-\frac{1}{2}\left[(x_{t+1}-Bs_t)^T(x_{t+1}-Bs_t)\right]} \cdot \frac{1}{(2\pi)^{np/2}} e^{-\frac{1}{2}\left[(B-A)^T(B-A)\right]} \partial B$$

Transferring more complex models

Same procedure on simulated and real building

Transfer a simple NN with polynomial features via SGD initialized randomly (Scratch) or by the learned model A (Transfer)