SMU
SINGAPORE MANAGEMENT UNIVERSITY

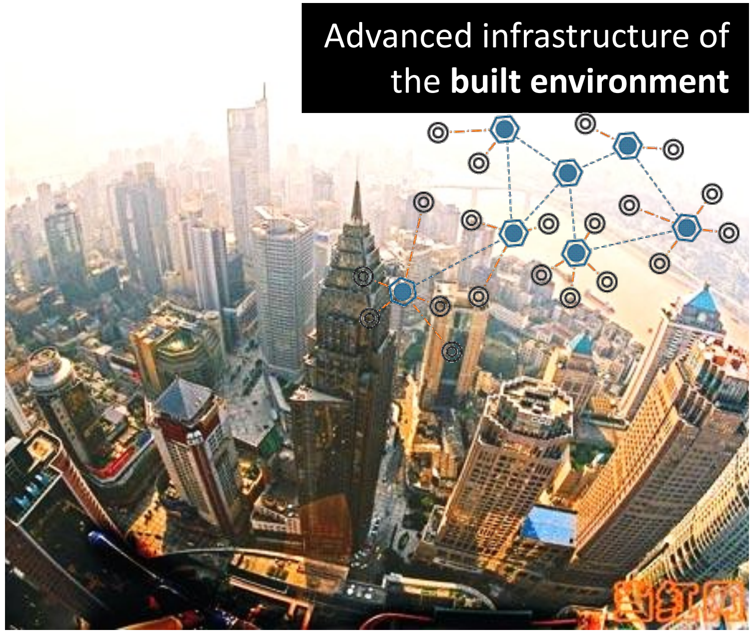
AN INTEGRATED FRAMEWORK FOR MODELING AND PREDICTING SPATIOTEMPORAL PHENOMENA IN URBAN ENVIRONMENTS

Truc Viet Le

September 21, 2017

School of
Information Systems

By 2050, **67%** of the world's population (6 billion people) would live in urban areas[1].



Advanced infrastructure of the **built environment**

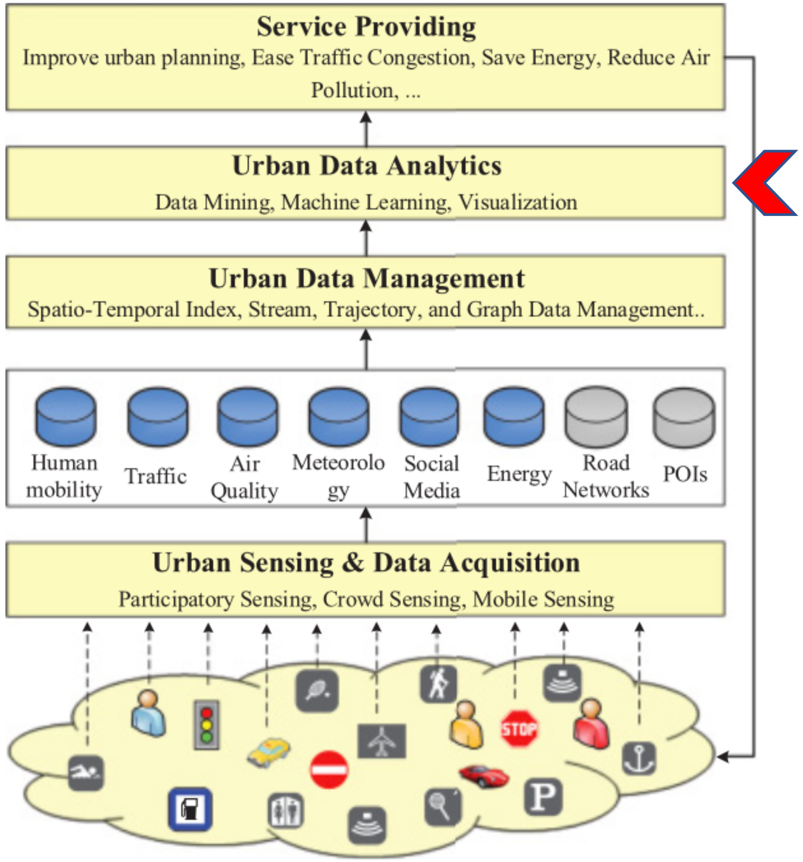
ST data offers **multi-scaled perspectives** at the complex behaviors of urban systems.

Multimodal **transportation networks**

Big Data and the (Big) City

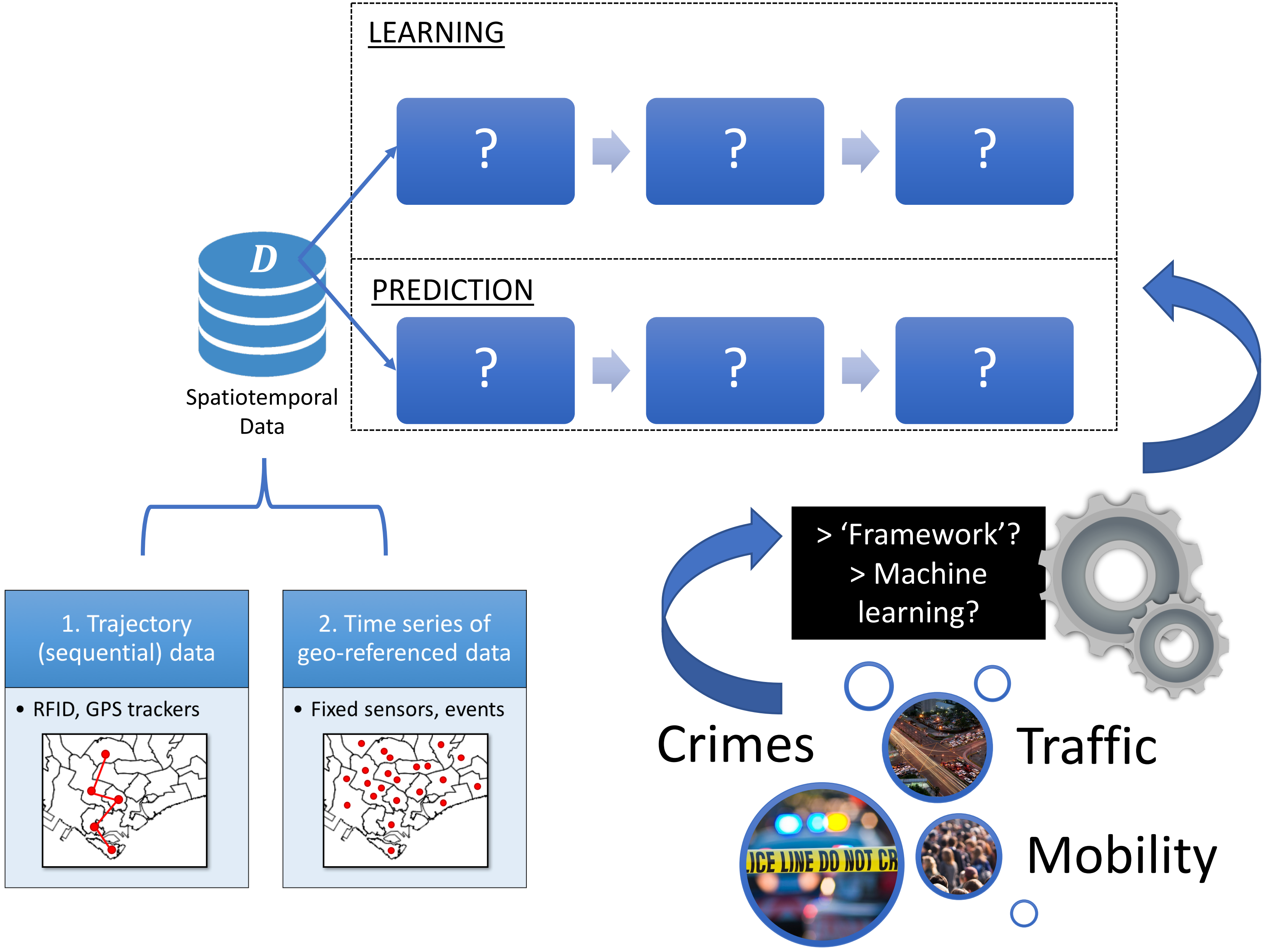
> Use spatiotemporal data to make cities safer and smarter.

[1]Heilig, G. K. (2014). World urbanization prospects the 2014 revision. *United Nations, Department of Economic and Social Affairs (DESA), Population Division, Population Estimates and Projections Section, New York*.



General Framework for Urban Computing Research
Zheng, Y., Capra, L., Wolfson, O., & Yang, H. (2014). Urban computing: Concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *5*(3), 38.

"…unlocks the power of big data collected in urban spaces to solve major issues cities face today."



What is this thesis about?

Environment

- Theme Park
- Road Networks
- Urban Areas

Underlies →

Phenomenon

- Human Mobility
- Traffic Speed
- Crime Incidents

Underlies →

Process

- Knapsack, Markov Decision Process
- Gaussian Process

Generates →

Data

- Trajectories
- Speed Readings
- Incident Counts

Λ Λ Λ

Human Mobility

POIs, Trajectory Prediction

Traffic Speed

Real-time Speed Prediction

Crime Incidents

Distribution of Incidents over Urban Areas

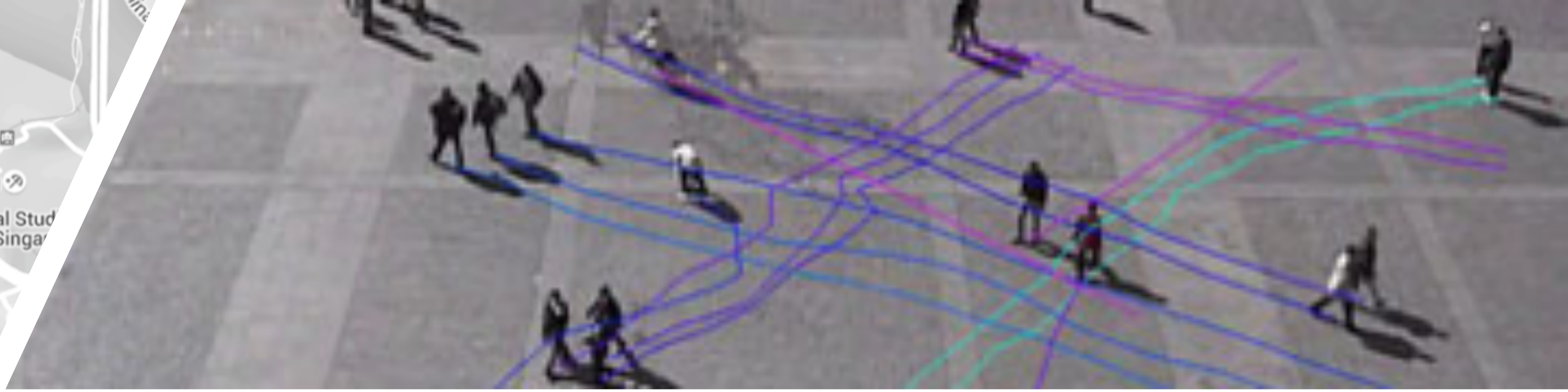
Problems Addressed: Overview

Rest of the Presentation:

- Data and real-world context
- Problem statement and challenges
- **GP**: What is it? Why use it?
- Proposed solution framework
- Experiments: Design, evaluation and results

- The problem-solving processes of the proposed solution frameworks are abstracted into a common 'pipeline'

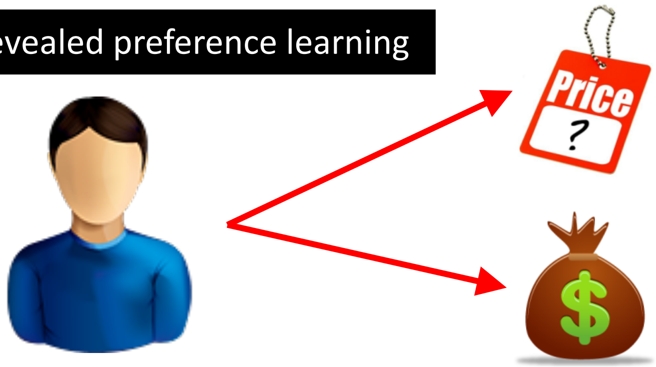
- Motivation ✓
- Problems Addressed: Overview ✓
- Human Mobility Prediction:
  - Spatial bundle prediction
  - Trajectory prediction
- Traffic Speed and Crime Incident Prediction
  - Gaussian process (GP) models
- The Integrated Framework
- Contributions and Summary



Human Mobility Prediction

PROBLEM I-II

Revealed preference learning



$$\max \sum_{i=1}^{n} x_i v_i$$

$$\text{s.t.} \sum_{i=1}^{n} x_i w_i \leq W.$$

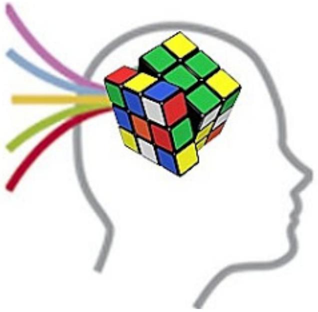
Knapsack problem

Spatial **Bundle** Prediction

Bundled Pass

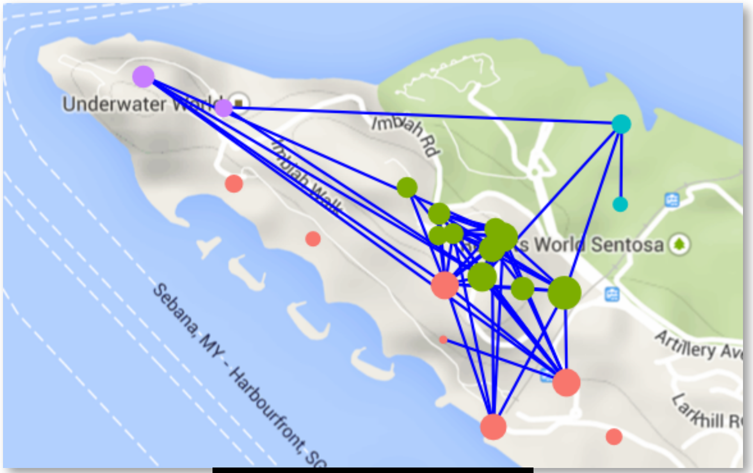
Choice Pass → Choose **4** out of **16** attractions (from 9 a.m. to 7 p.m.)

Day Pass → Visit all **14** attractions (from 9 a.m. to 7 p.m.)



Trajectory Prediction

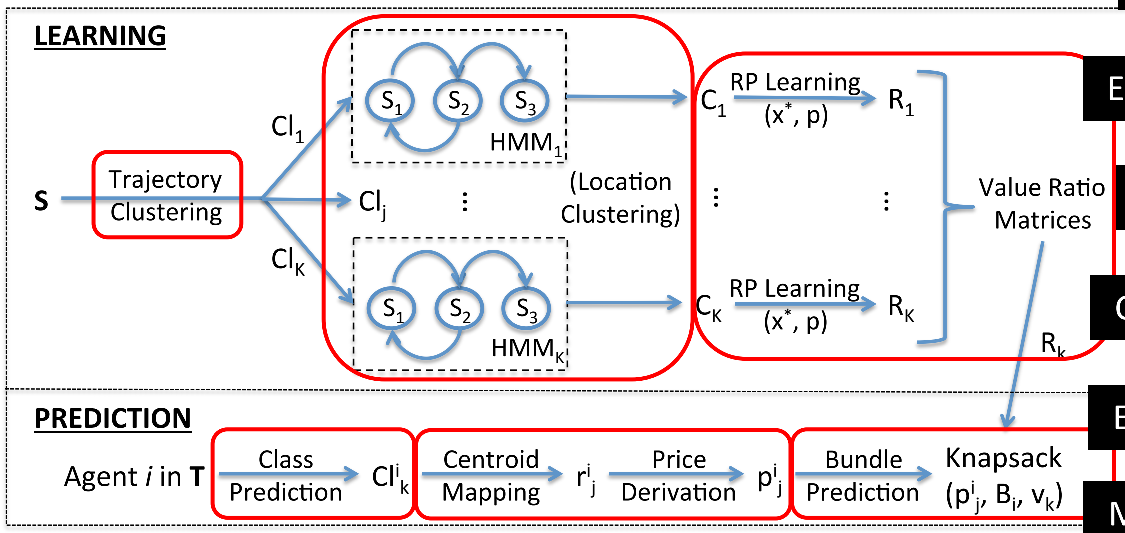
Reinforcement learning

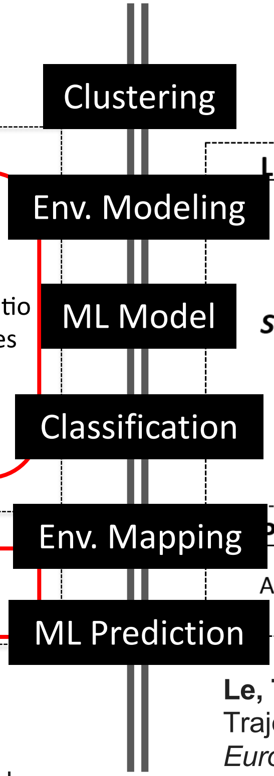


Visitor trajectories

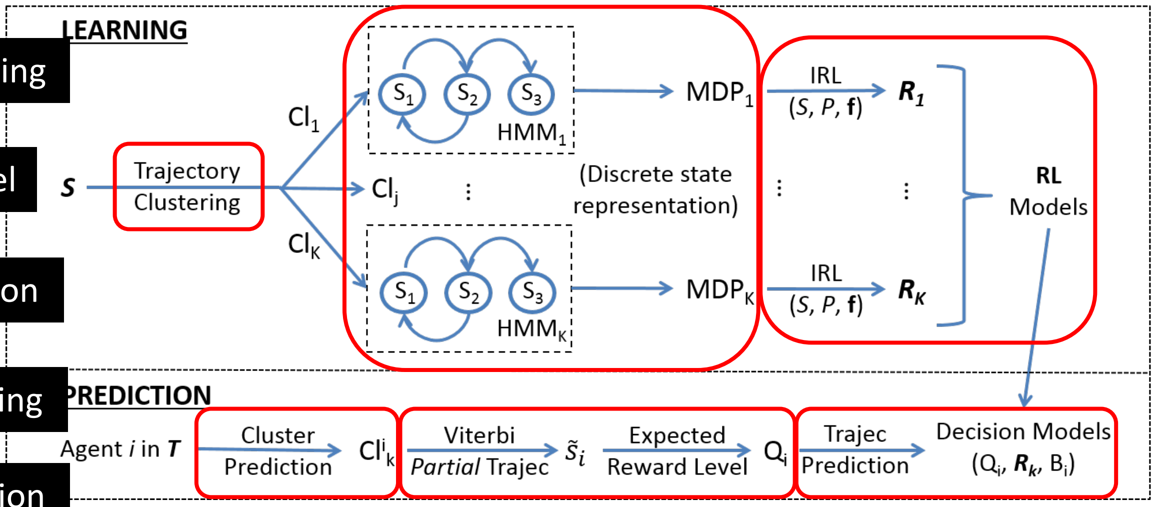
Sentosa Data and Context

Spatial Bundle Prediction (Revealed Preference)





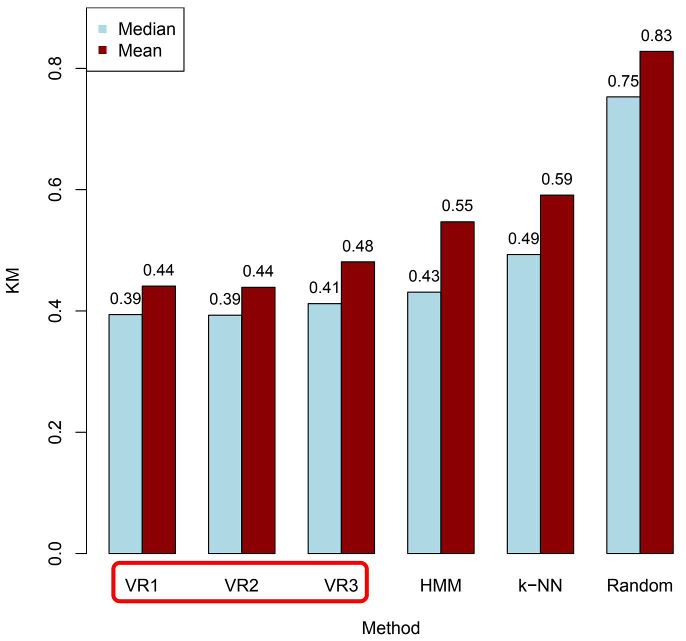
Trajectory Prediction (Reinforcement Learning)



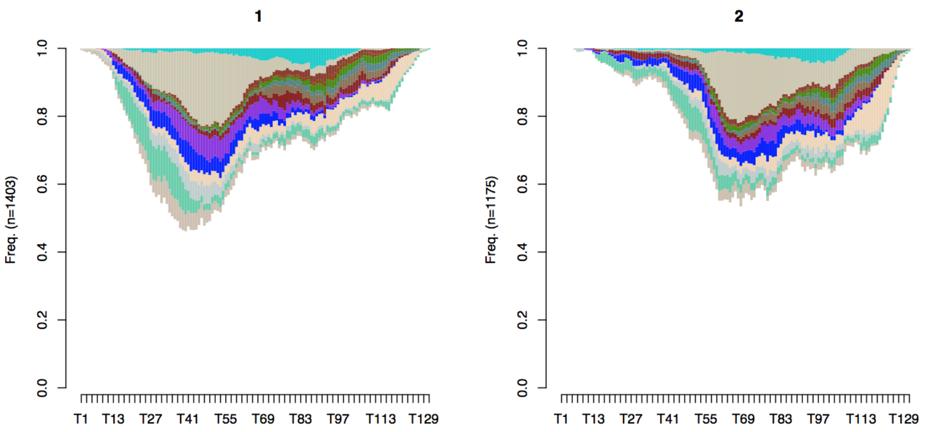
Le, T. V., Liu, S., Lau, H. C., & Krishnan, R. (2015, May). Predicting bundles of spatial locations from learning revealed preference data. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems* (pp. 1121-1129). International Foundation for Autonomous Agents and Multiagent Systems.

Le, T. V., Liu, S., & Lau, H. C. (2016, August). A Reinforcement Learning Framework for Trajectory Prediction Under Uncertainty and Budget Constraint. In *ECAI 2016: 22nd European Conference on Artificial Intelligence, 29 August-2 September 2016, The Hague, The Netherlands-Including Prestigious Applications of Artificial Intelligence (PAIS 2016)* (Vol. 285, p. 347). IOS Press.

Proposed Frameworks

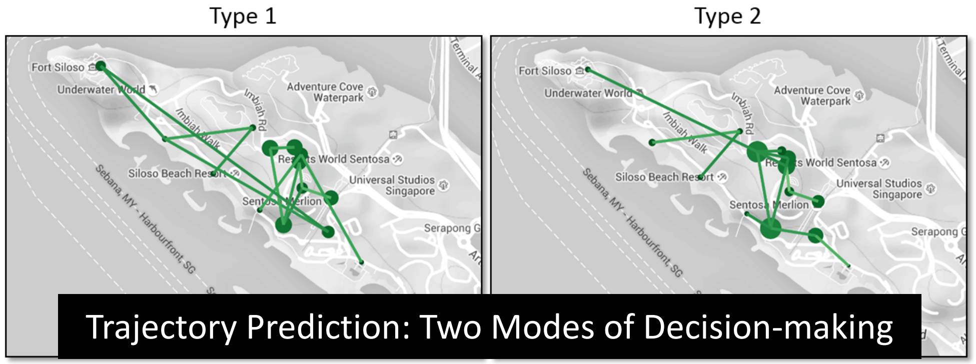


Bundle Prediction: Evaluation and Comparison



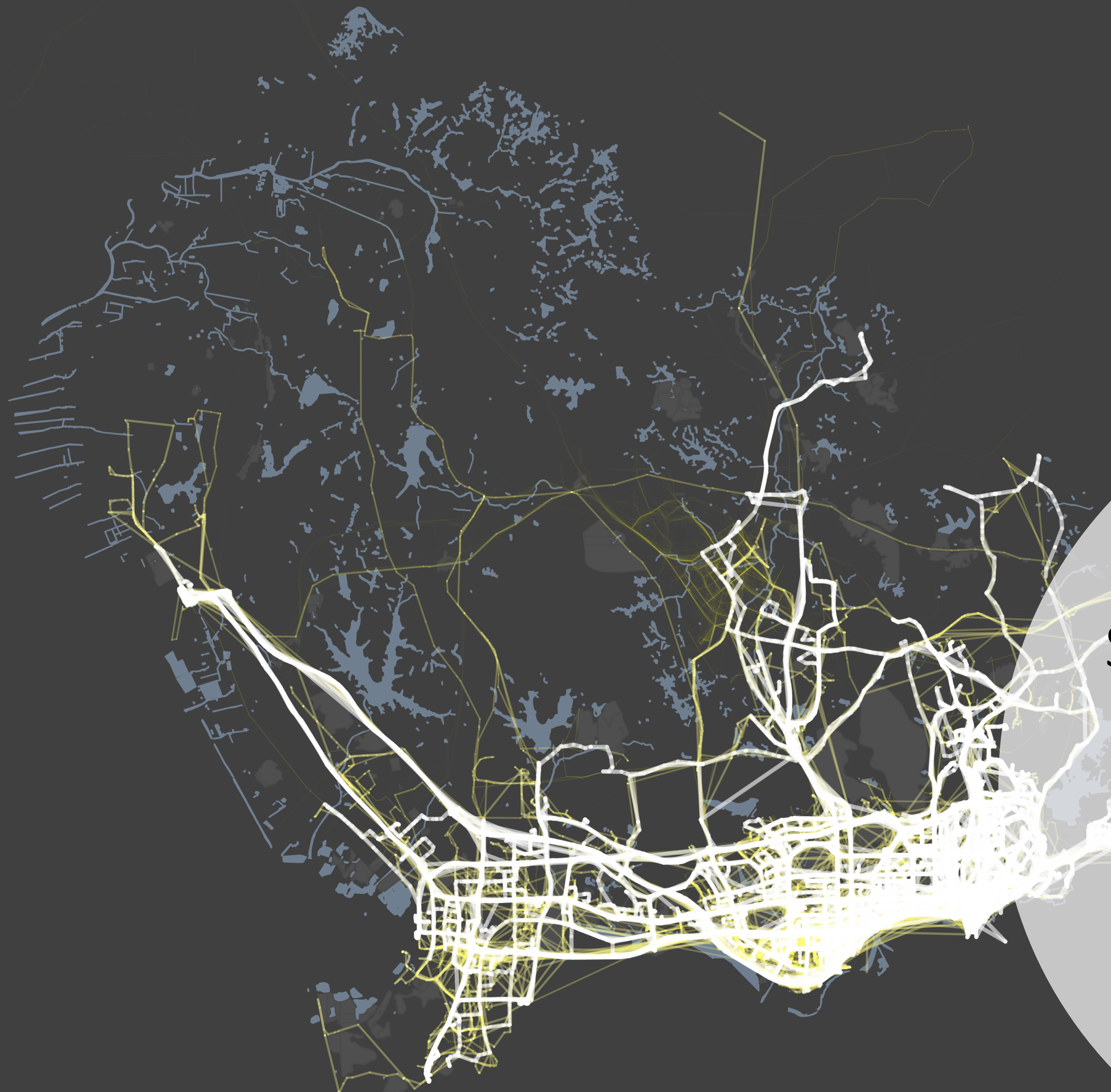
Trajectory Prediction: Two Agent Types (Clusters)

- **Type 1** visitors arrive earlier, have **larger budget** → less time-sensitive and more well-planned → **MDP** better models this group.
- **Type 2** visitors arrive later, have **smaller budget** → more time-sensitive and prone to myopic decision-making → **Greedy** heuristics better model this group.



Trajectory Prediction: Two Modes of Decision-making

Evaluations and Findings



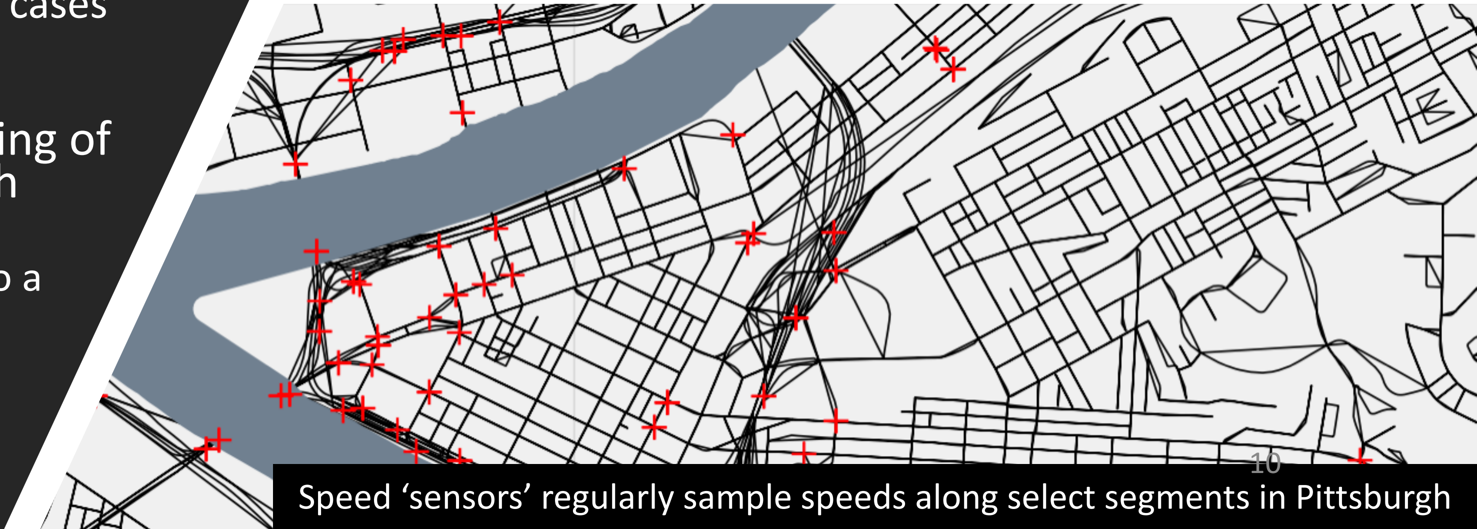
Fine-grained Traffic Speed Prediction Using Local Gaussian Processes

PROBLEM III

U.S. Traffic Data and Problem

- Speed reading every 5 minutes on some road segments in Pittsburgh and Washington, D.C. during March – August, 2014
- Spatially infer speed values for the whole network (unobserved locations)
- Temporally infer speeds at future time steps
  - Fine-grained inferences (extensive spatial coverage and short-term horizon) → Needs accuracy and efficiency for real-time use cases
- **Main idea**: *localization* – efficient clustering of spatiotemporally correlated sensors, each represents a 'local' **Gaussian process**
  - Train and predict in real-time in response to a traffic speed *query*

Spatially inferred speed distribution for the whole road network



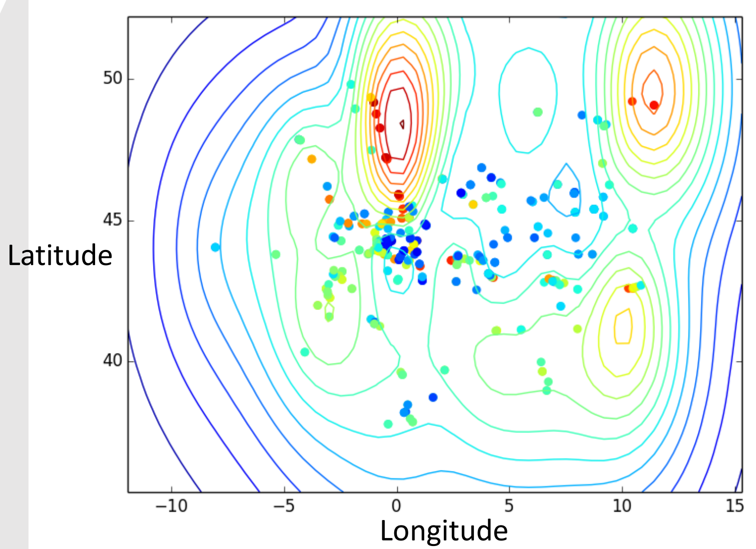
Speed 'sensors' regularly sample speeds along select segments in Pittsburgh

What is a Gaussian Process (GP)?

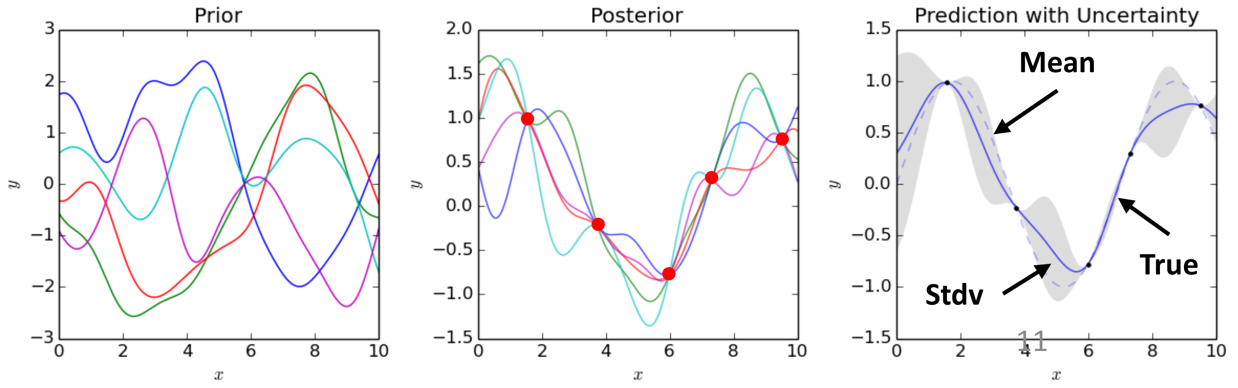
- Consider linear regression: $y = \theta_0 + \theta_1 x + \epsilon$
  - **Bayesian** linear regression finds a (posterior) distribution for the parameters Θ that gets updated whenever new data are observed.
- GP is a **non-parametric** approach that finds a distribution over all possible **functions** $f(x)$ that are consistent with the observed data:
  - Begins with a **prior** distribution
  - Updates it as new data are observed → **Posterior** distribution over all functions:

$$p(f|D) = \frac{p(f)p(D|f)}{p(D)}$$

- *GP defines a prior <u>over functions</u>, which can be converted into a posterior <u>over functions</u> once we have seen some data, which can then be used for <u>Bayesian regression</u>.*



Traffic speed sample in Pittsburgh



Why GP?

- Given a set of data points $x_1, \dots, x_n$, GP assumes that $p\big(f(x_1), \dots, f(x_n)\big)$ is *jointly Gaussian* with some **mean** $\mu(x)$ and **covariance** $\Sigma(x)$ given by $\Sigma_{ij} = K\big(x_i, y_j\big)$:
  - K is a positive-definite **kernel function**
  - If x_i and x_j are close to each other in the input space, the corresponding values in the output space should also be similar.
- GP is a **Bayesian** (regression) method (it gives the mean and 'error bar' estimates). It is also a **kernel** method:
  - Projects inputs into high-dimensional feature space implicitly and efficiently (via the 'kernel trick')
  - Models additive, multiplicative, convolutional, etc. interactions of features via 'kernel arithmetic'.

$$K(x, x') = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}|x - x'|}{\lambda}\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}|x - x'|}{\lambda}\right)$$

Matern kernel (geospatial statistics)

$$K(x, x') = \sigma_0{}^2 \exp\left[-\frac{1}{2}\left(\frac{x - x'}{\lambda}\right)^2\right]$$

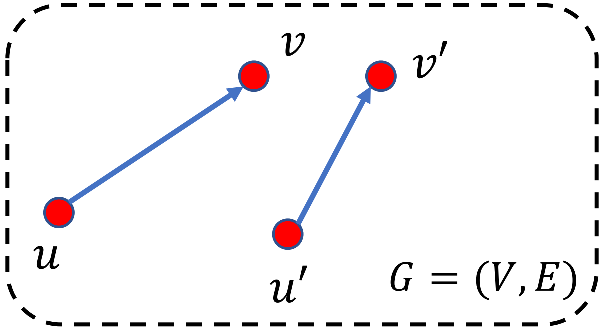
Radial basis function (RBF, aka Gaussian) kernel

- Let f be the posterior of the observed outputs and f_* the posterior of the outputs yet to be observed. Because it is a GP:

$$\begin{pmatrix} f \\ f_* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu \\ \mu_* \end{pmatrix}, \begin{pmatrix} K & K_* \\ K_*^T & K_{**} \end{pmatrix}\right)$$

- We can then 'sample' the **posterior**: $f_* \sim \mu + B \times N(0, I)$, where $BB^T = \Sigma_*$ (called Cholesky decomposition), which takes $O(N^3)$, where N is the sample size.

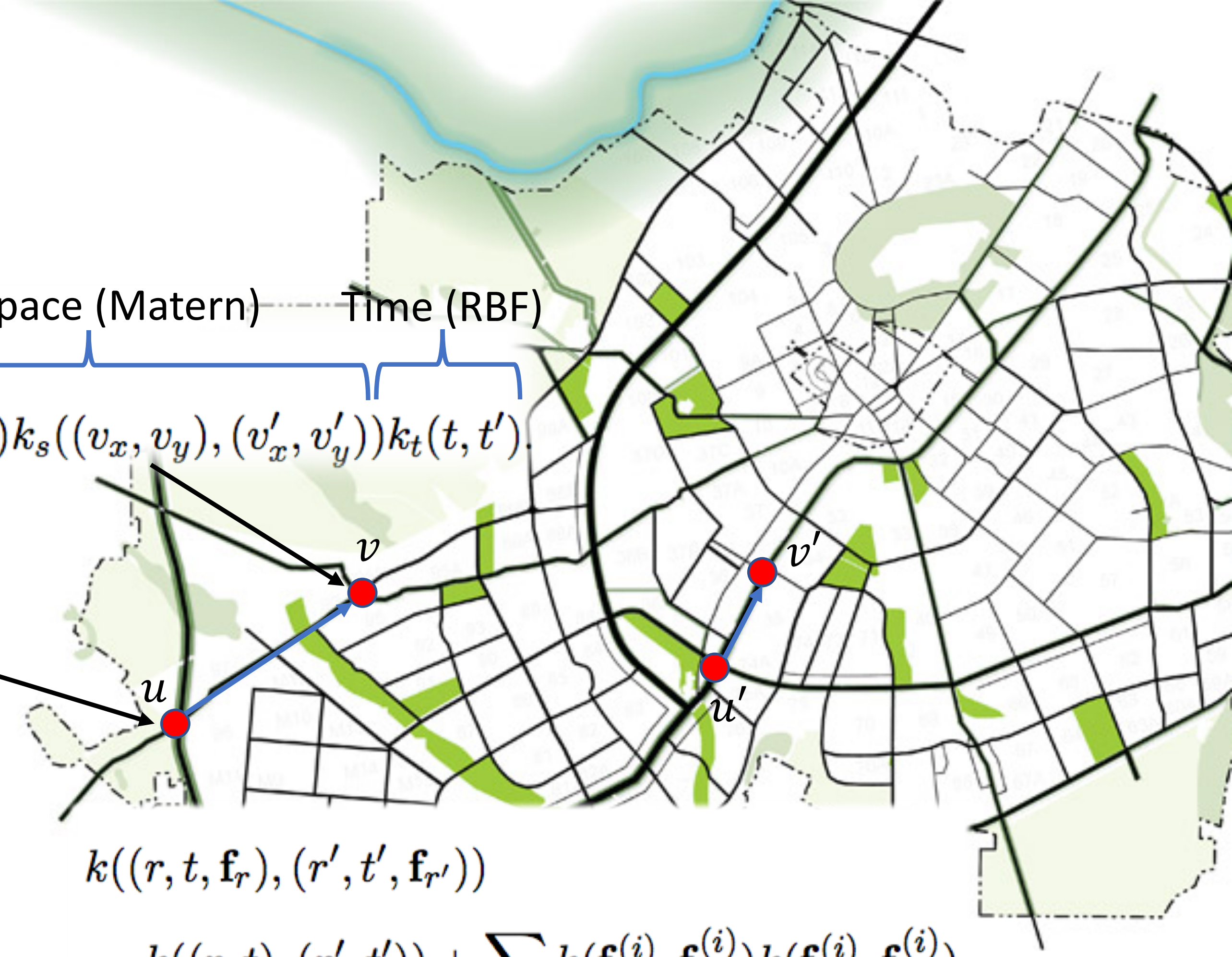
Spatiotemporal GP Kernel for Road Networks



$$k((u,v),(u',v')) = k(u,u')k(v,v'),$$

Yu, K., & Chu, W. (2008). Gaussian process models for link analysis and transfer learning. In *Advances in Neural Information Processing Systems* (pp. 1657-1664).

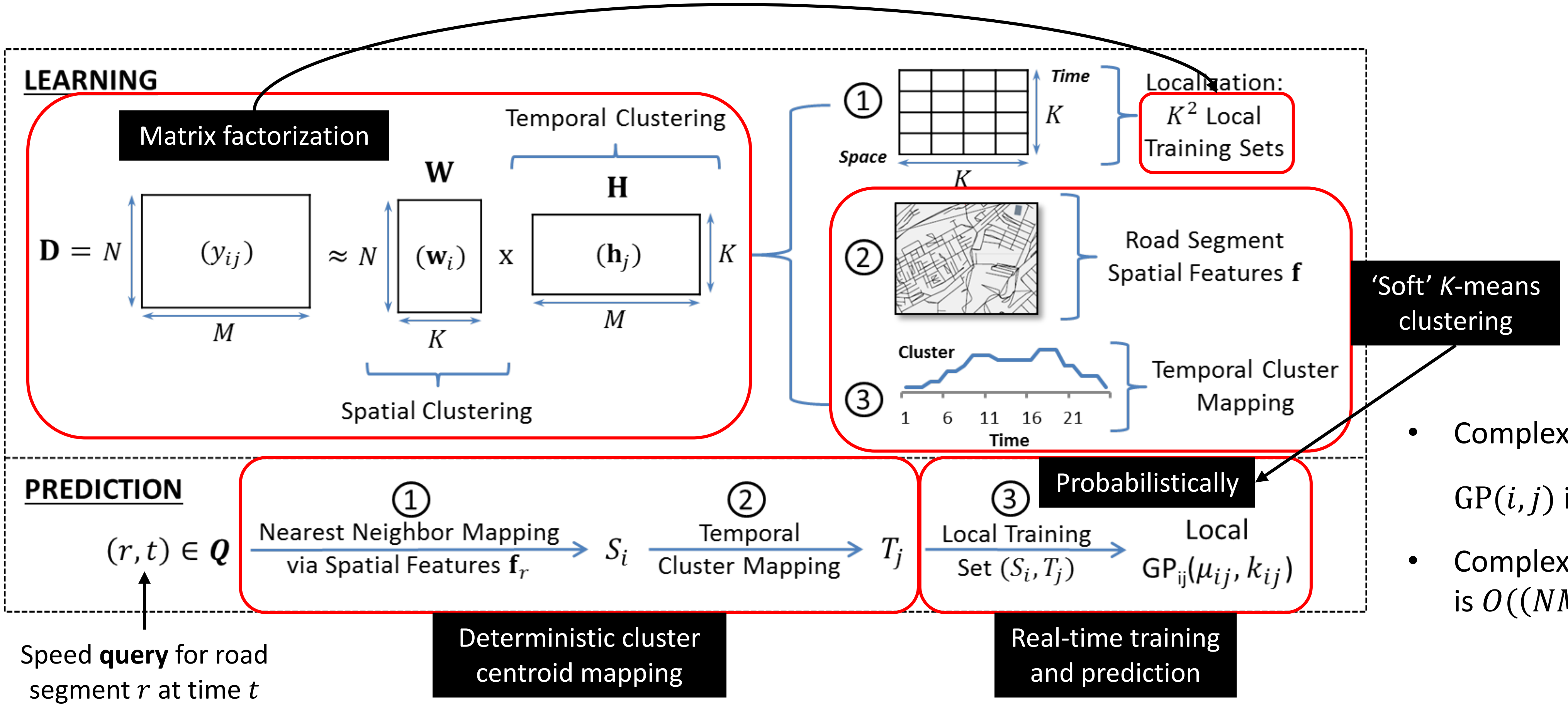
$$k((r,t),(r',t')) = k_s(\underbrace{(u_x,u_y)}_{\text{(lon, lat) of node } u}, \overbrace{(u'_x,u'_y))k_s((v_x,v_y),(v'_x,v'_y))}^{\text{Space (Matern)}}\overbrace{k_t(t,t')}^{\text{Time (RBF)}}.$$



$$k((r,t,\mathbf{f}_r),(r',t',\mathbf{f}_{r'})) = k((r,t),(r',t')) + \sum_i \underbrace{k(\mathbf{f}_u^{(i)},\mathbf{f}_{u'}^{(i)})k(\mathbf{f}_v^{(i)},\mathbf{f}_{v'}^{(i)})}_{\text{Node-wise features}} + \sum_j \underbrace{k(\mathbf{f}_{(u,v)}^{(j)},\mathbf{f}_{(u',v')}^{(j)})}_{\text{Edge-wise features}},$$

Feature	Description
Longitude, latitude	Longitude and latitude coordinates of the two endpoints (nodes) of a segment.
Segment length	Length (in miles) of a segment.
Number of lanes	The number of lanes a segment has in each direction.
Direction	Direction of a segment: northbound, southbound, eastbound, or westbound.
Degree	Degree of two end nodes of an edge (segment).
Betweenness	Edge betweenness centrality of a segment.
One-way	Is this segment one-way?
Road type	One of the 10 defined types: avenue, boulevard, bridge, lane, place, ramp, road,

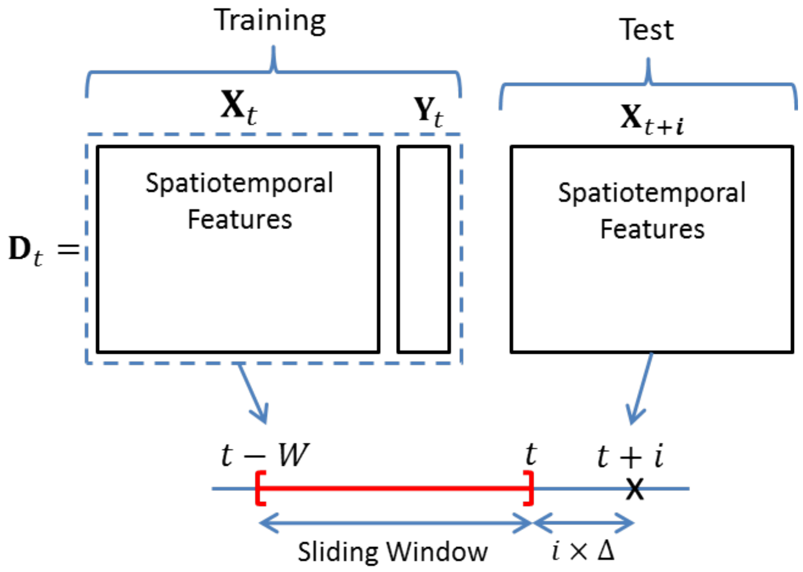
GP Framework for Real-time Traffic Speed Prediction



- Complexity of a **local** GP(i,j) is $O\left(\left(\frac{NM}{K^2}\right)^3\right)$
- Complexity of a **global** GP is $O((NM)^3)$

Experiments: Design and Evaluation

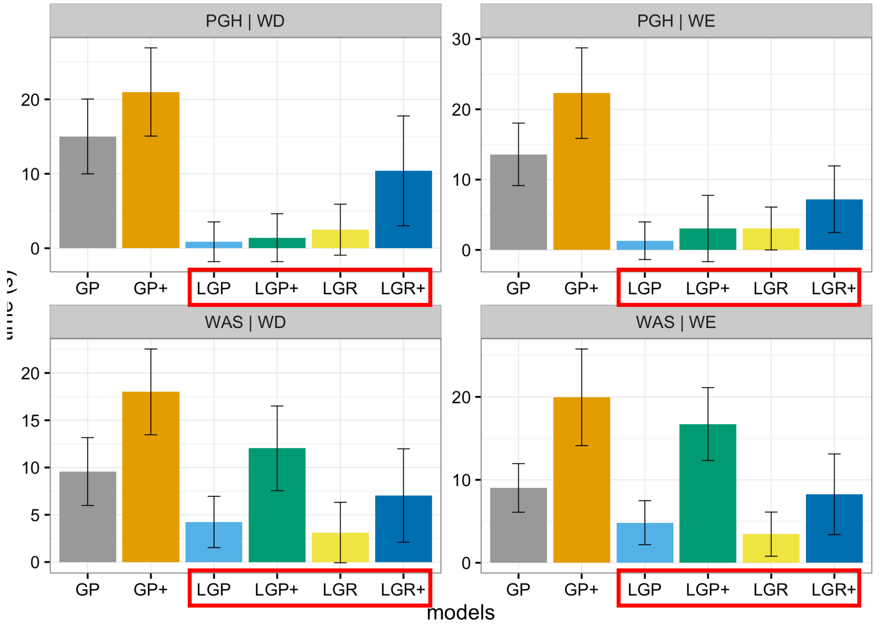
- 'Sliding window' experimental design:
  - Train at time t and test at times $t + i$
  - W is the length of window of observations
  - Δ is the test duration (e.g., 5 minutes)
  - Evaluate separately for **weekdays** and **weekends**
  - Evaluate **six** (6) different models (local vs. global, with/without 'side information')
- Let $\hat{y}$ and y be the estimate and true values, the evaluation **metrics** are:
  - Mean absolute error (MAE)
  - Mean absolute percentage error (MAPE)
  - Root-mean-square error (RMSE)

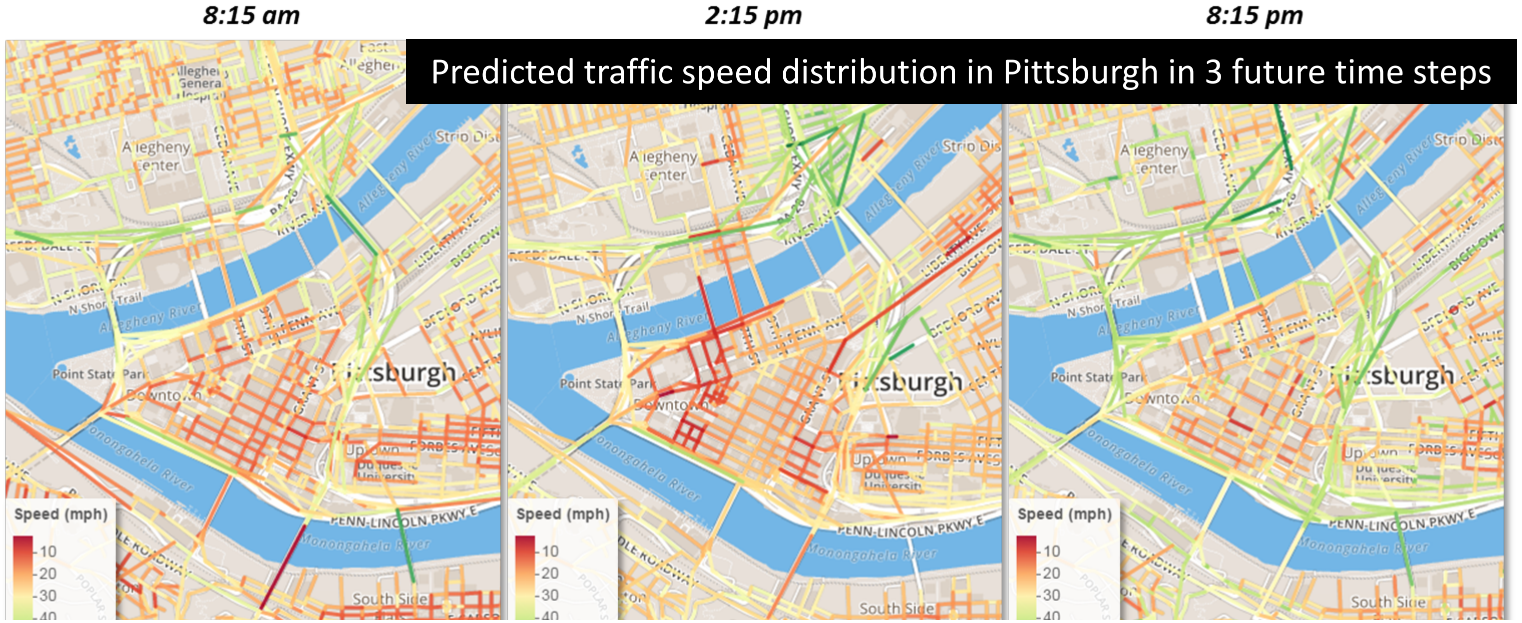


$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2},$$

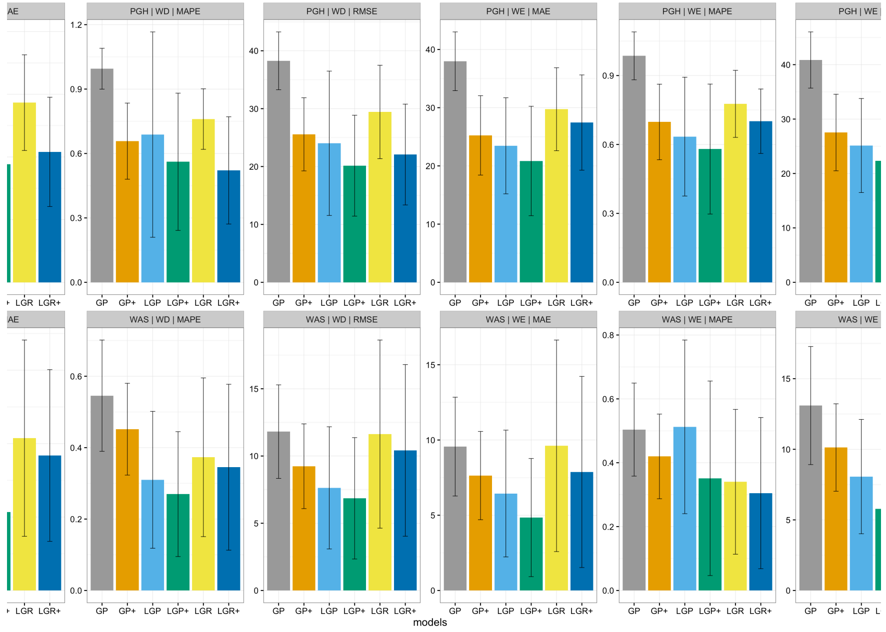
$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i|,$$

$$\mathrm{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{\hat{y}_i - y_i}{y_i}\right|,$$





Predicted traffic speed distribution in Pittsburgh in 3 future time steps



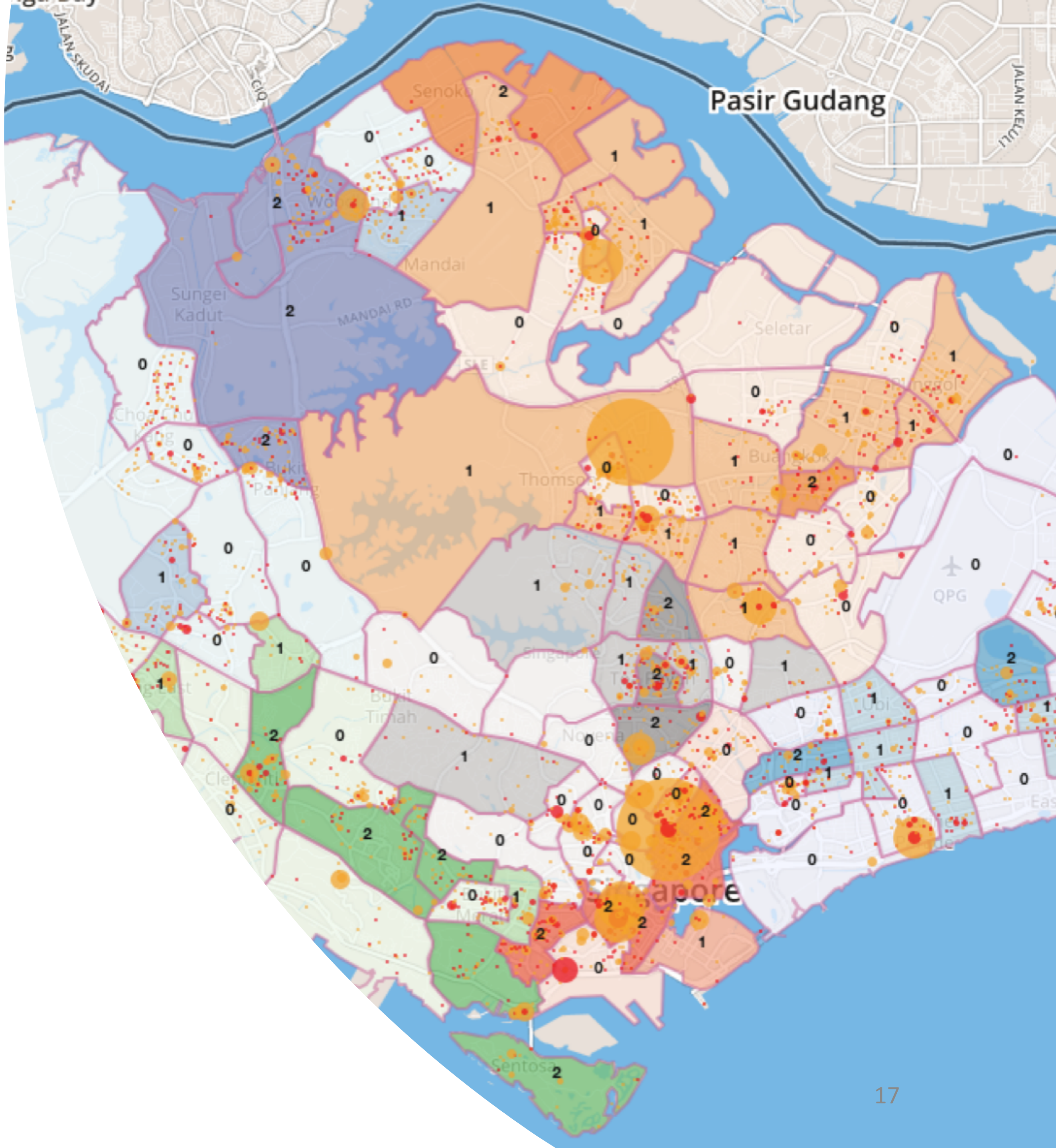
Mean errors measured in MAE, MAPE and RMSE

Experimental Results: Baselines and Comparison

Le, T. V., Oentaryo, R., Liu, S., & Lau, H. C. (2017). Local Gaussian Processes for Efficient Fine-Grained Traffic Speed Prediction. *IEEE Transactions on Big Data*, *3*(2), 194-207.

PROBLEM IV

Incident Prediction for Law Enforcement Resource Optimization

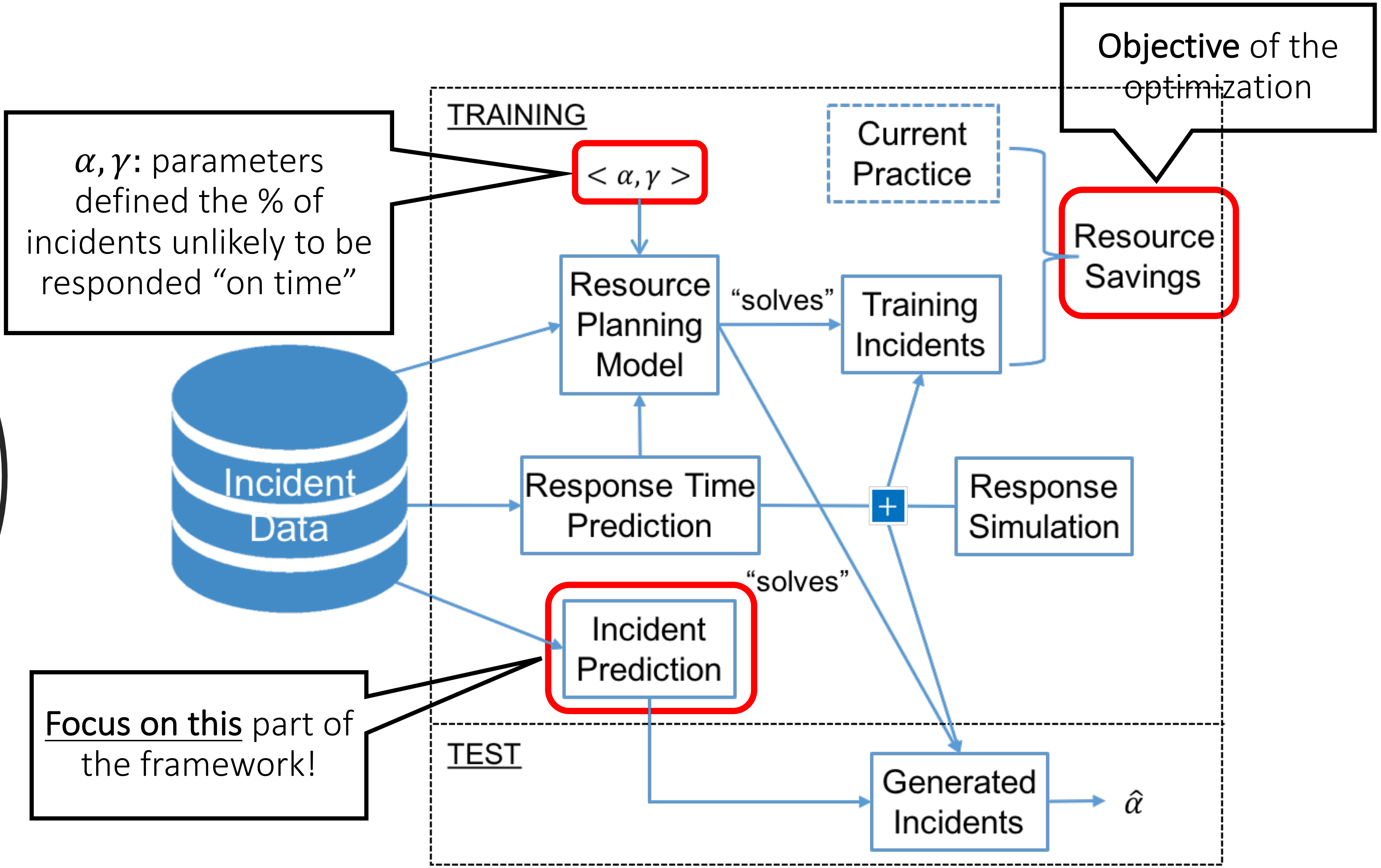


The Law Enforcement Problem

Law Enforcement Resource Optimization Problem

- Densely populated urban areas increasingly puts pressure on law enforcement (LE) agency's manpower trying to meet ever-rising demands
- Large databases of crime incidents provide fine-grained details:
  - Spatiotemporal (where and when)
  - Context (textual description, urgency classification, type and police response)
- It is possible to make high-precision <u>predictions of incident occurrences</u> using ML
- This serves a larger purpose of a data-driven solution framework for <u>LE resource optimization</u>
  - Time taken to respond to an incident (called "response time") is a common KPI for many LE agencies

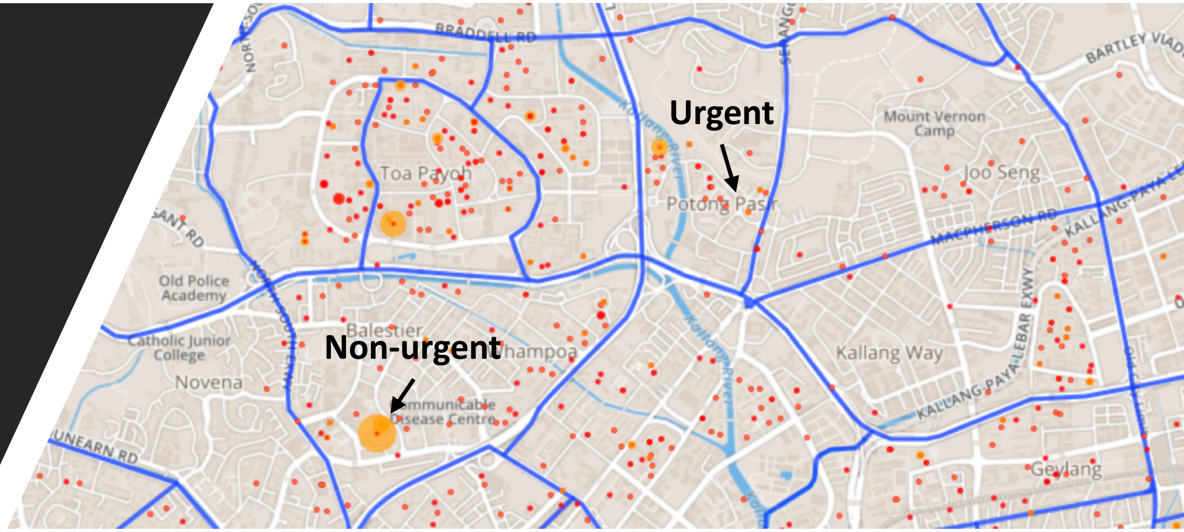
Data-driven Framework for Law Enforcement Resource Optimization



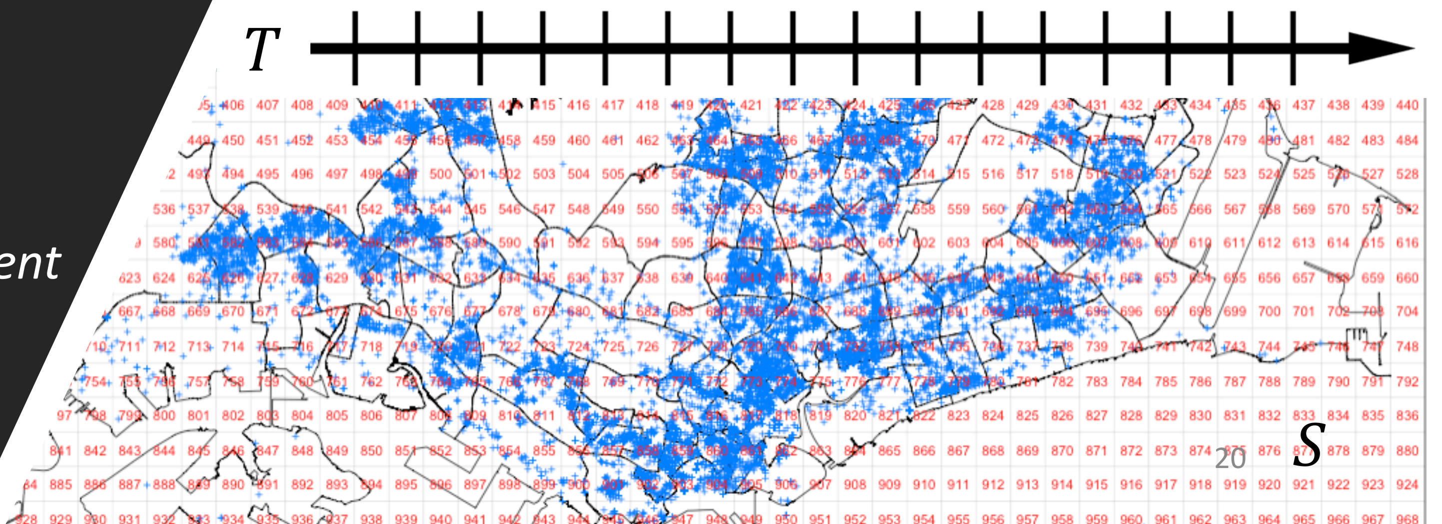
Jonathan Chase, Jiali Du, Na Fu, **Truc Viet Le** and Hoong Chuin Lau. *Law Enforcement Resource Optimization with Response Time Guarantees.* Accepted to the 2017 IEEE Symposium Series on Computational Intelligence (SSCI 2017), Honolulu, Hawaii, USA.

Problem Statement

- Divide the city's maps and timeline into $|S|$ finite grid squares and $|T|$ intervals
- For each 'type' of incident, model the distribution of the *count* (i.e., number of incidents) within each spatiotemporal combination $|S||T|$
- Each such combination is called a 'bin'

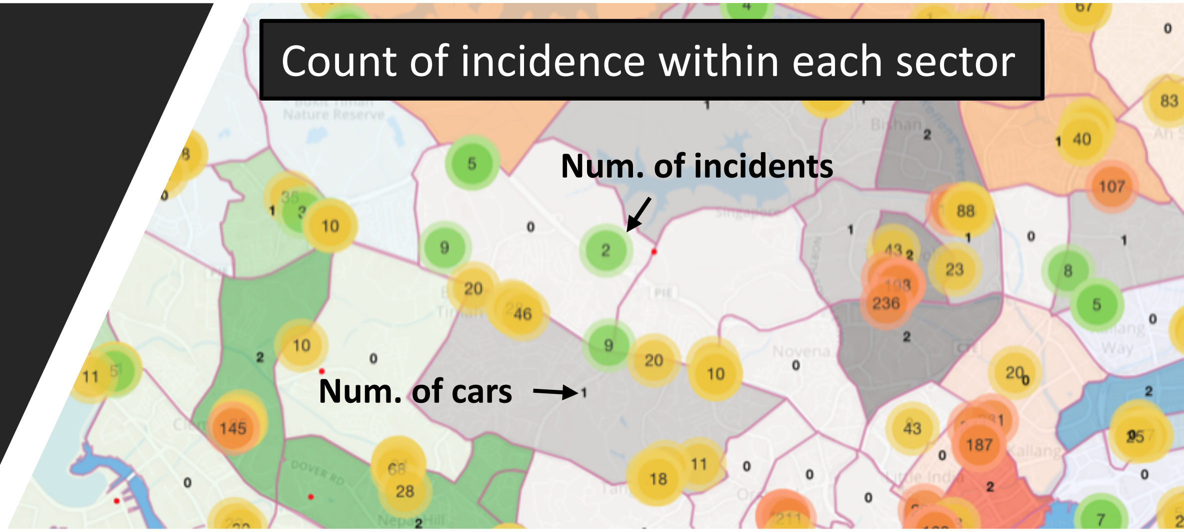


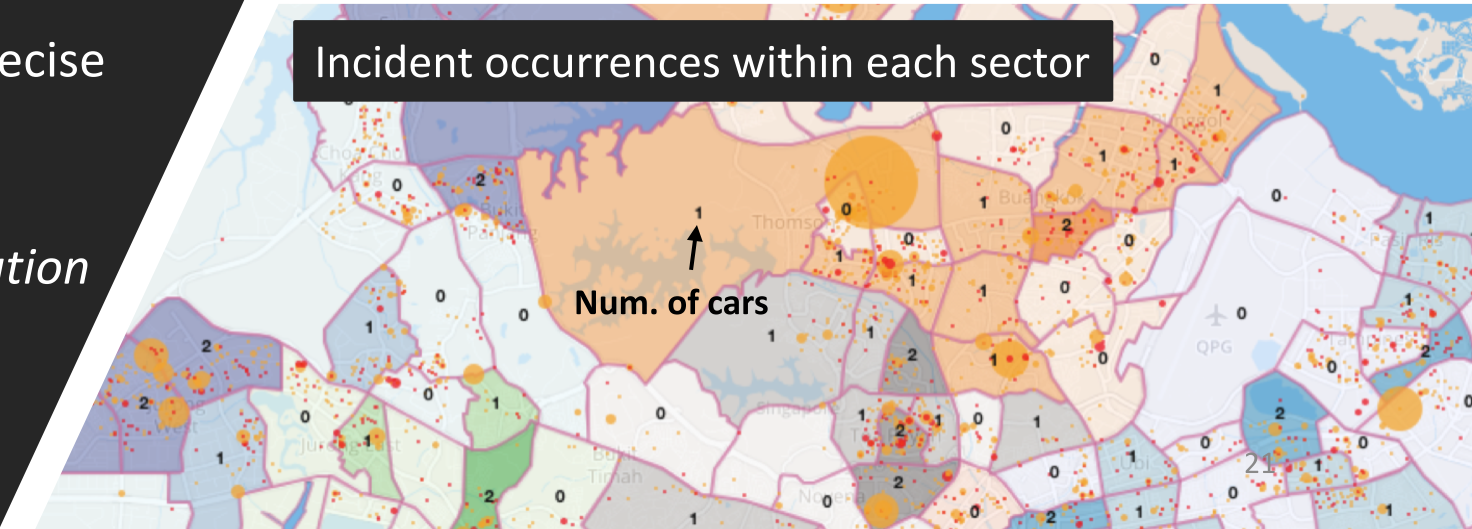
- Given a query (x, y, t), hash it into a bin index i that has features $\mathbf{f}_i$
- Predict the count variable in i
- *Assuming uniform distribution of the incident occurrence within each bin*



Why ‘count’ variable?

- For each sector, given the number of incidents that occur in it at each interval
- The number of cars allocated <u>in that sector at that interval</u> (prescribed by the optimization) guarantees that the response time KPI can be reached with prob. $\geq 1 - \alpha$
- Hence, we don’t need to predict the precise location and time of each incident
- *The simulator assumes uniform distribution of incidents within each sector*



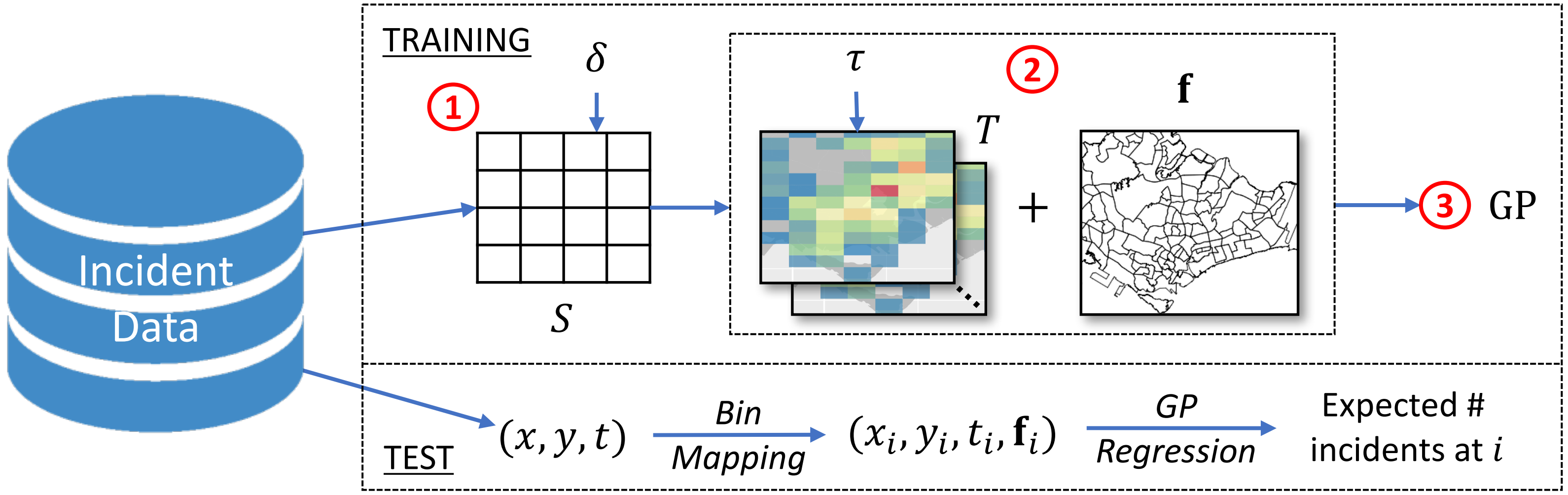


Crime Data and Context

- Real-world data provided by a large national law enforcement agency
- Spans over one-year period 2013-14
- Contains more than 200,000 reported incidents (e.g., from emergency calls)

- Each incident has: location, timestamp, type, urgency classification, dispatch and response information (incl. response time)
- Metadata containing neighborhood/sector boundaries and police deployment information

GP Framework for Incident Prediction



Notation	Description
$S, \lvert S \rvert$	Spatial dimension and number of spatial grids, respectively
$T, \lvert T \rvert$	Temporal dimension and number of time intervals, respectively
δ, τ	Parameters specifying the granularity of the spatial and temporal dimension
x, y, t	Longitude, latitude and timestamp, respectively
x_i, y_i, t_i	Centroid coordinates of bin i and interval index, respectively
$\mathbf{f}_i$	Feature vector of bin i

Training:

- **Step 1**: Discretize the spatial dimension into grid squares
- **Step 2**: Discretize the temporal dimension into intervals → $|S||T|$ bins
  - Compute the feature vector $\mathbf{f}_i$ of each bin i
- **Step 3**: Learn the count distribution using spatiotemporal GP coupled with the features of each bin

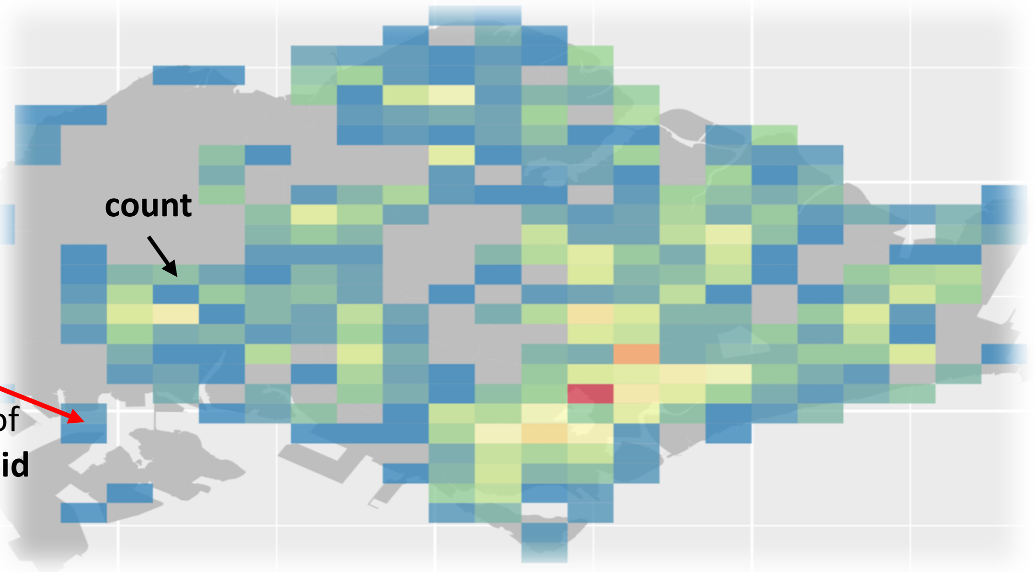
GP Kernel and Features

The spatiotemporal **kernel function** between bins i and j:

$$k((x_i, y_i, t_i), (x_j, y_j, t_j)) = \underbrace{k_s((x_i, y_i), (x_j, y_j))}_{\text{Spatial}} \underbrace{k_t(t_i, t_j)}_{\text{Temporal}},$$

Matern/RBF kernel: Spatial × Temporal

(lon, lat) of the **centroid**



$$k((x_i, y_i, t_i, \mathbf{f}_i), (x_j, x_j, t_j, \mathbf{f}_j))$$

$$= k((x_i, y_i, t_i), (x_j, y_j, t_j)) + \underbrace{\sum_f k(\mathbf{f}_i, \mathbf{f}_j)}_{\text{Additive features}},$$

Linear kernel: Additive features

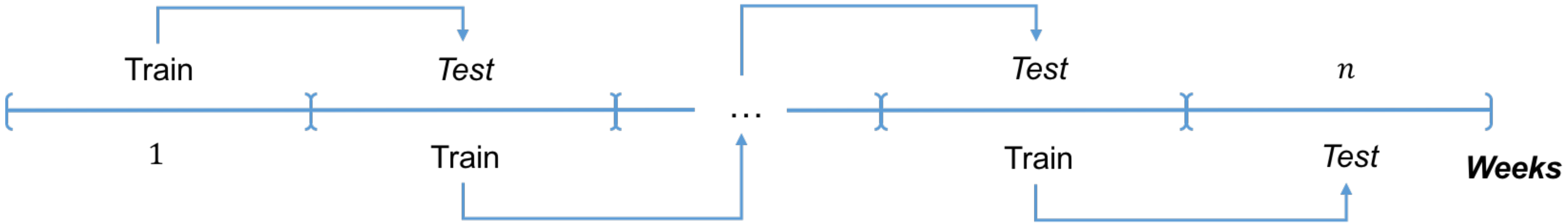
Feature	Description
location	Longitude and latitude coordinates of the incident
hours	The integer hours of the incident's occurrence time (0–23)
is_weekend	Binary variable whether the incident occurs on the weekend
neighborhood	Categorical variable specifying the incident's neighborhood
sector	Categorical variable specifying the incident's sector

Experiments: Design and Evaluation

Two types of incidents:

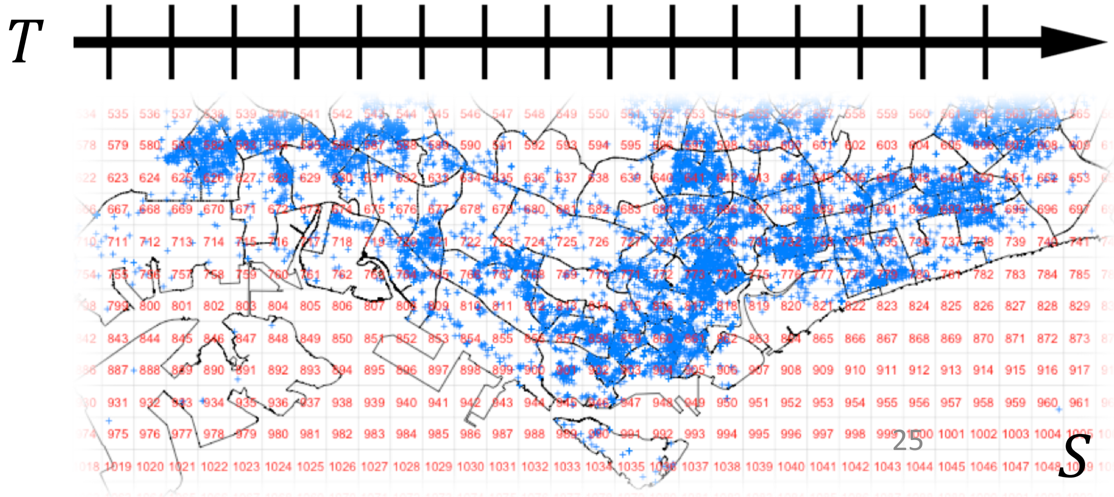
- **Urgent** (33%)
- **Both** (urgent + non-urgent)

“Sliding window” experimental design ($n = 12$):



- $S = 100 \times 100$ squares
- $T = \{24\}$ hours

→ **Avg. count per bin** per weekday/weekend (Training)

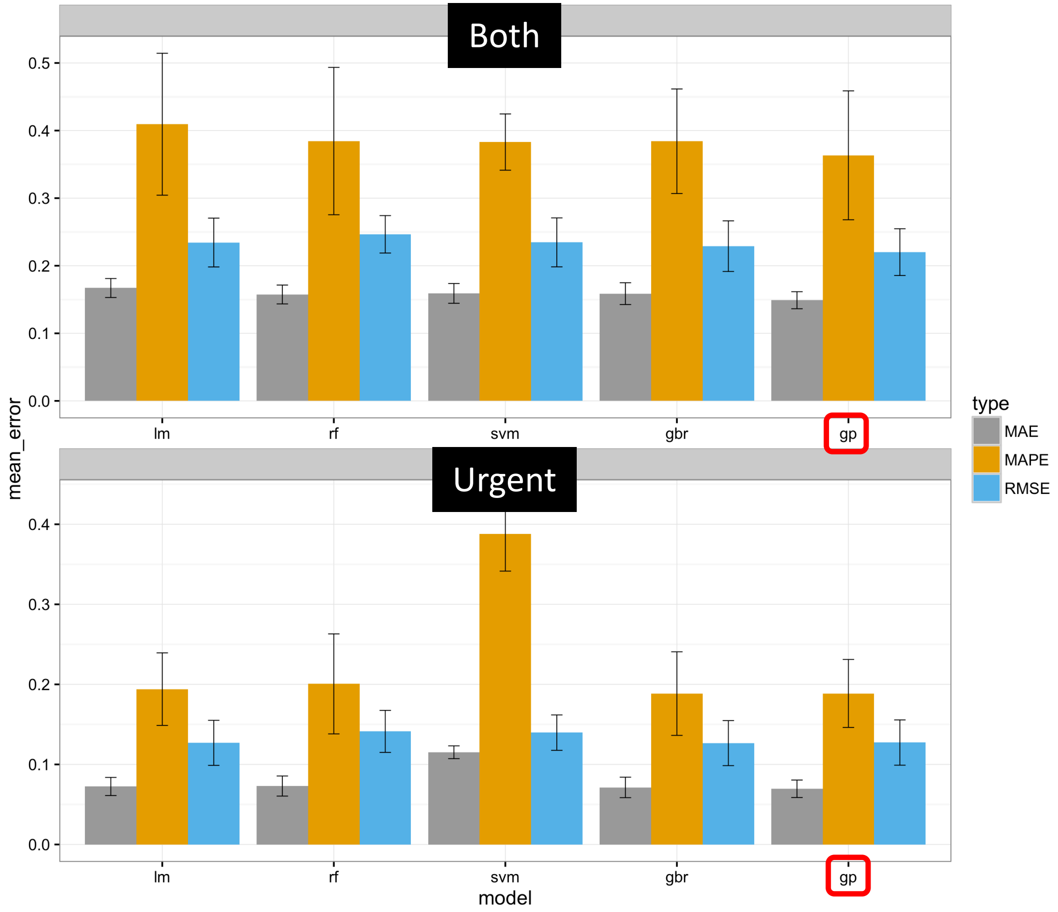


Evaluation **metrics** are:

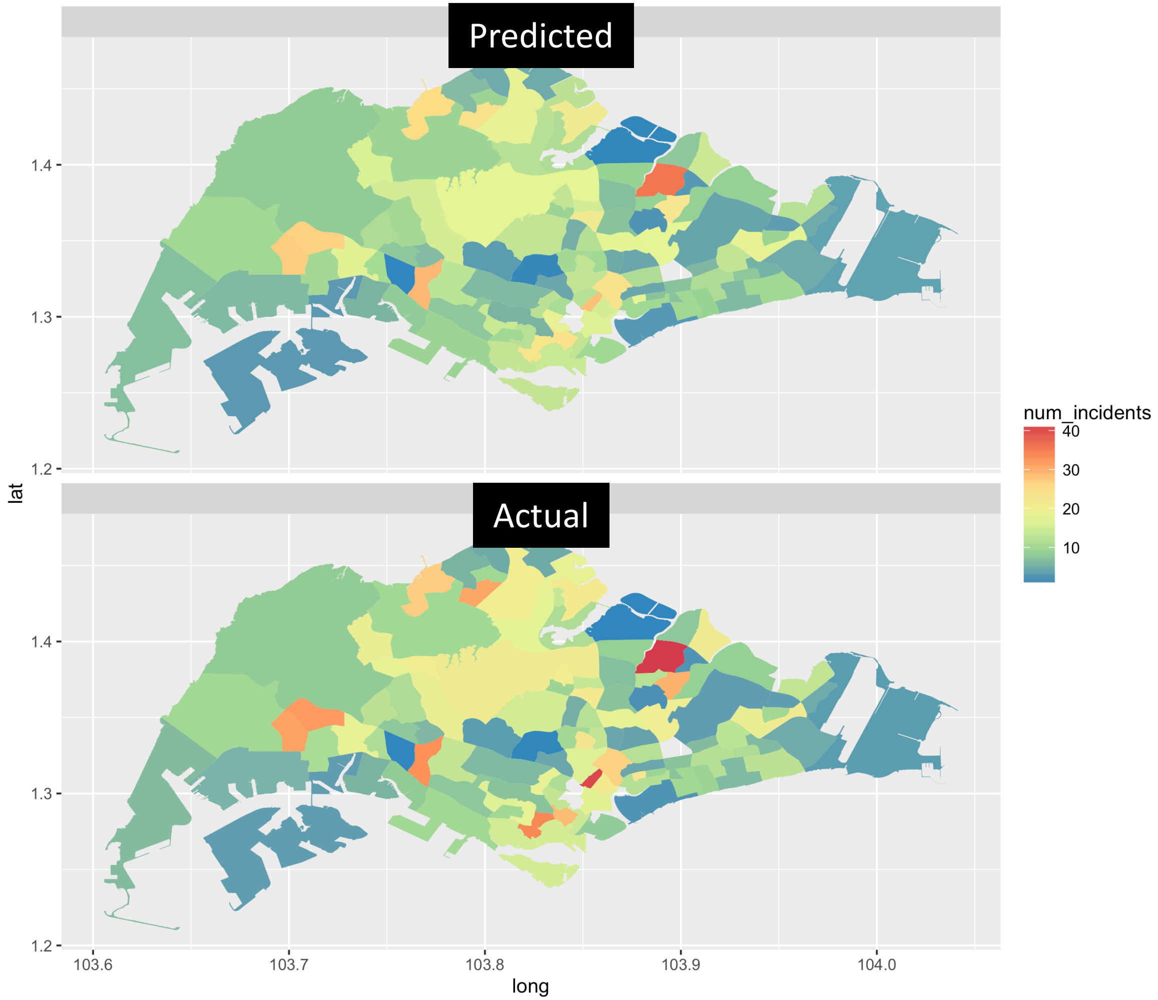
- Mean absolute error (MAE)
- Mean absolute percentage error (MAPE)
- Root-mean-square error (RMSE)

Finally, take the **average** of each of the metrics for all the *test* weeks (6).

Experimental Results: Baselines and Comparison



- Linear regression (LM)
- Random forest (RF)
- Support vector machine (SVM)
- Gradient boosting regression (GBR)

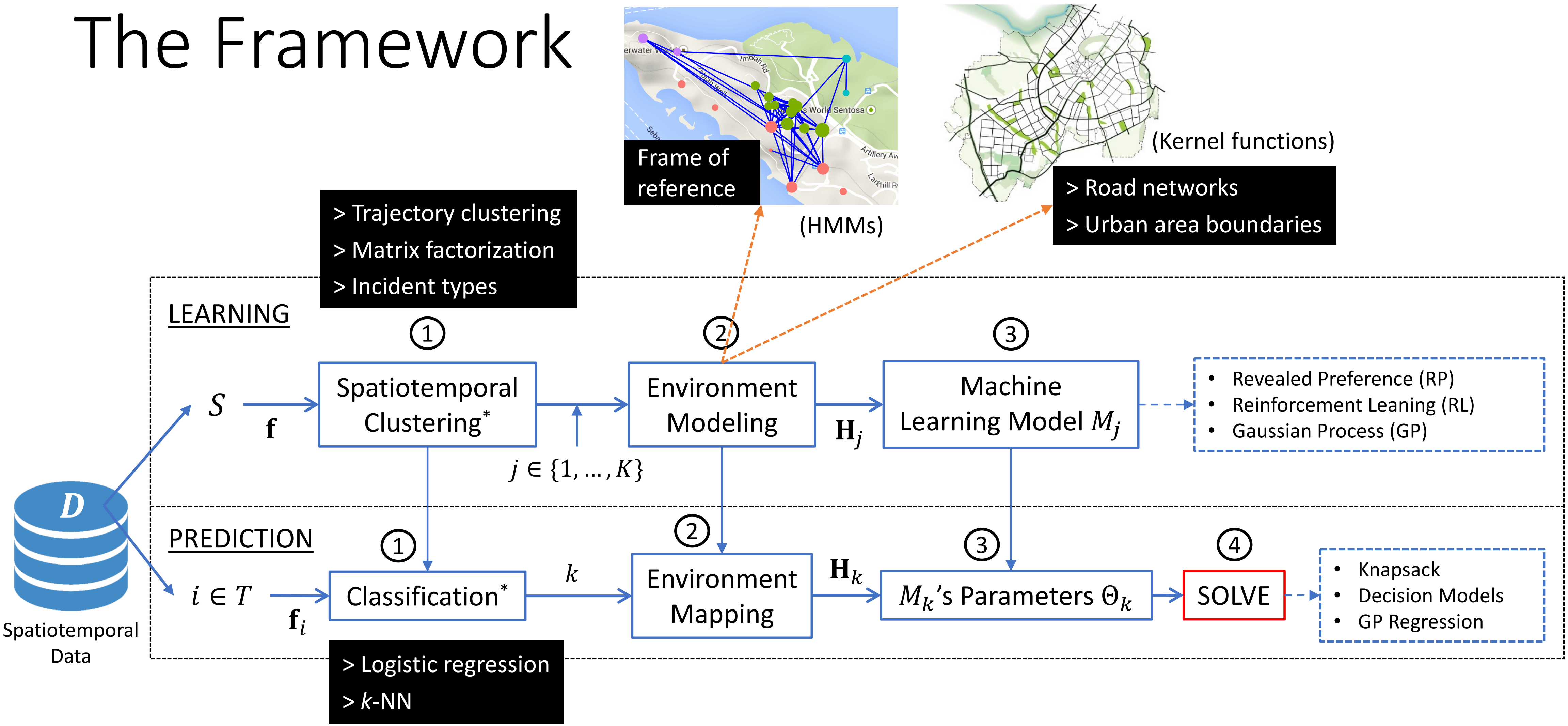


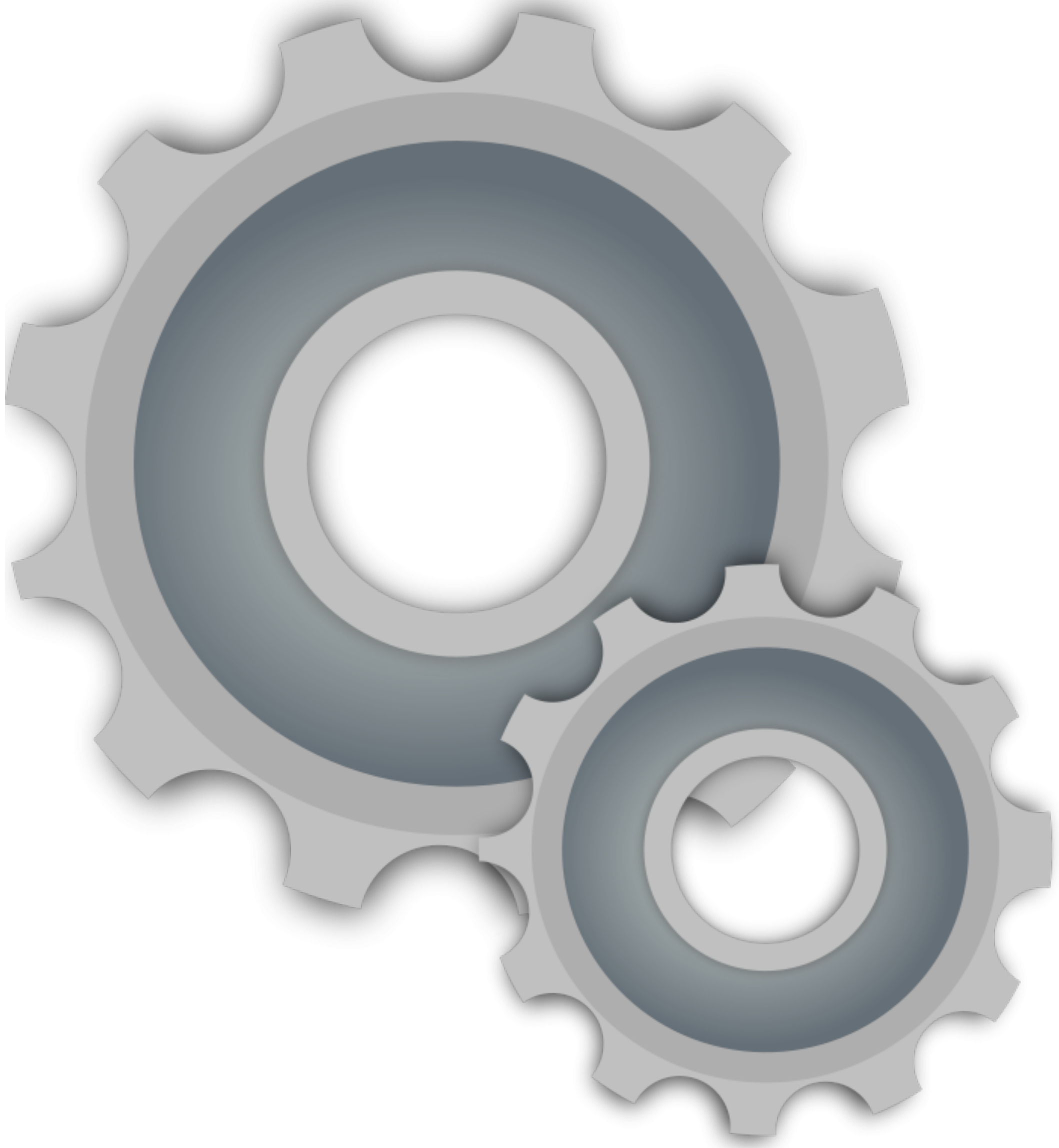
Comparing the **predicted** and **actual** number of incidents (**both** for weekday + weekend) for one particular test week

THE INTEGRATED FRAMEWORK:

Modeling and Predicting Spatiotemporal Phenomena in Urban Environments

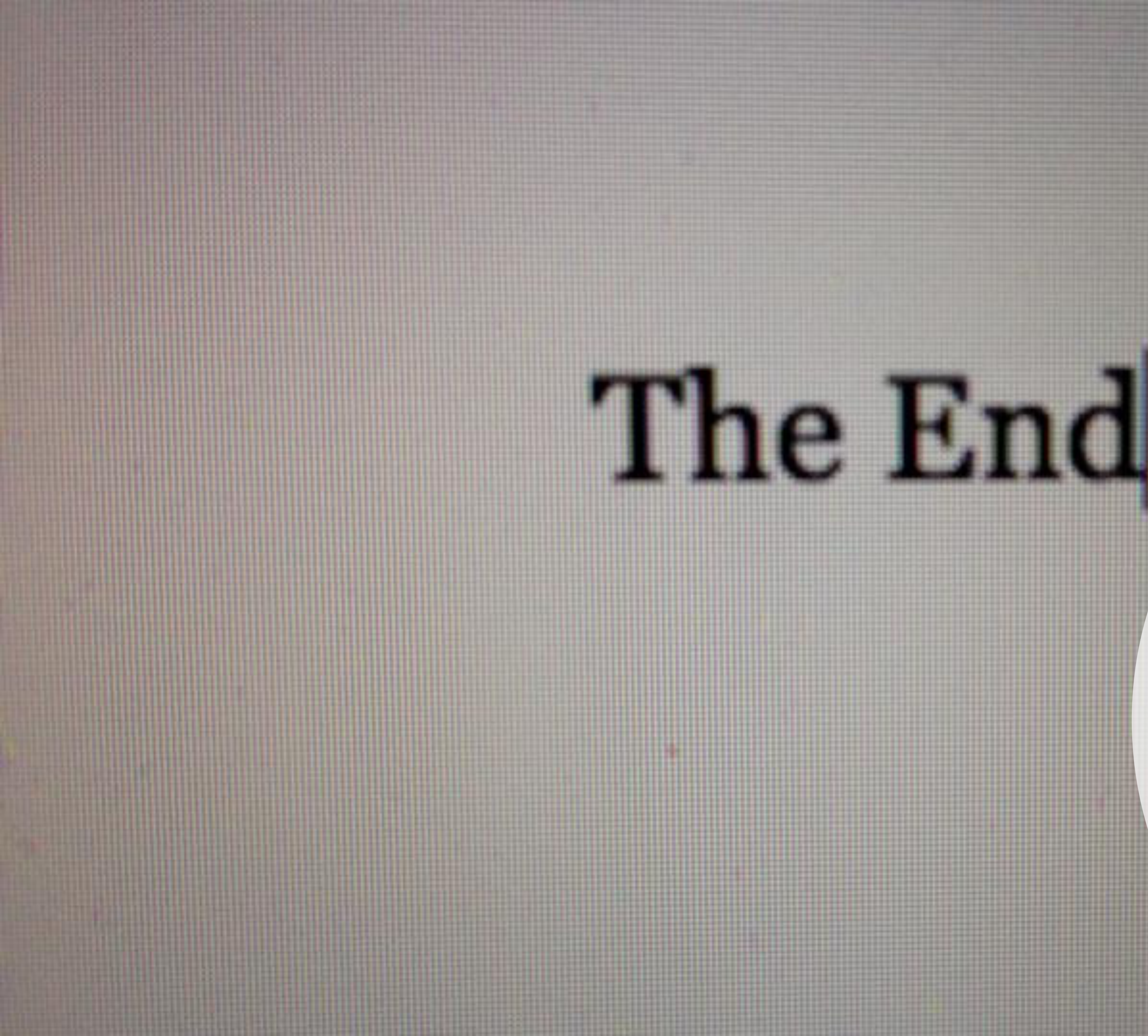
The Framework





Contributions and Summary

- Extends the “Urban Data Analytics” box (data mining, machine learning, visualization) of the Urban Computing Framework (Zheng *et al.*, ’14)
- The framework combines ML methods to solve a diverse set of real-world problems in urban environments using spatiotemporal data
  - Human mobility prediction
  - Traffic speed prediction
  - Crime incident prediction
- The framework abstracts features of the individual solutions into a common problem-solving process that is highly generalizable
  - Clustering and classification
  - Modeling of the environment
  - Environmental mapping and ML parameters retrieval



Thank You and Questions?