Rebuilding The Traffic Dynamics From Partial Traffic Data Using Information Theory/ML

A Hybrid Markov Chain/ Neural Network Approach

Alexandros Sopasakis
Mathematics
Lund University, Sweden

on going work with
Markos Katsoulakis &
Luk Ray Bellet,
Univ. of Massachusetts, USA

Center for Scientific and Technical computing
LUNARC Lund University

Ximantis AB

Outline

- *Part 1*. The lattice-free microscopic dynamics: Desirable features for specific applications
  - Specialized potentials: Look-ahead dynamics
  - Conservation
- *Part 2*. Parameters and Calibration
  - Information Theory
  - Neural Networks

Simulations & comparisons throughout

Outline

- *Part 1*. The lattice-free microscopic dynamics: Desirable features for specific applications
  - Specialized potentials: Look-ahead dynamics
  - Conservation

- *Part 2*. Parameters and Calibration
  - Information Theory
  - Neural Networks

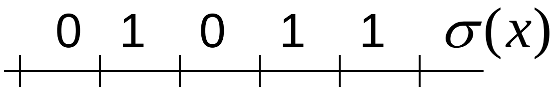
Simulations & comparisons throughout

Main Statistical Mechanics Concepts
Describing the interacting particle system

We let Λ denote a lattice of N cells.

and consider[1] the microscopic spin-like variable $\{\sigma_t\}_{t\geq 0}$ on Λ

We denote by σ(x) the spin at location x,

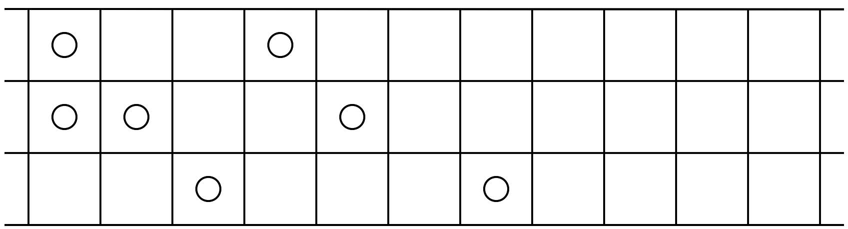


While we denote by $\sigma := \sigma_t$ the complete configuration of the lattice at time t.

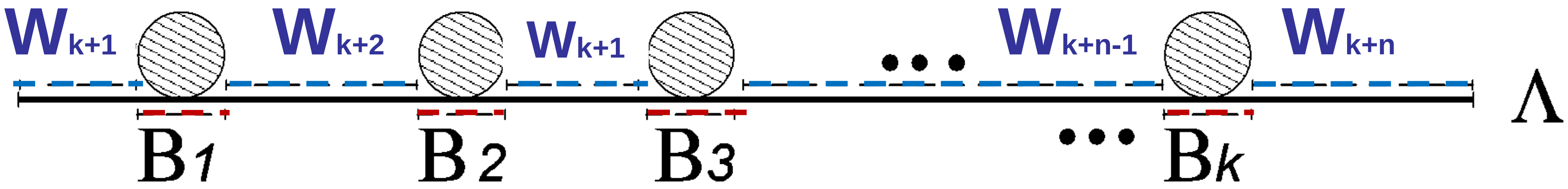
Configuration σ is an element of the configuration space $\Sigma = \{0,1\}^{\Lambda}$ and we write $\sigma = \{\sigma(x) : x \in \Lambda\}$

1. Katsoulakis, Majda and Vlachos, Comput. Phys. 186(1), 2003.

Consider moving away from lattices



Set-theoretic partition. Idealization in 1D



Schematic of disjoint sets comprising Λ for 1-D example.

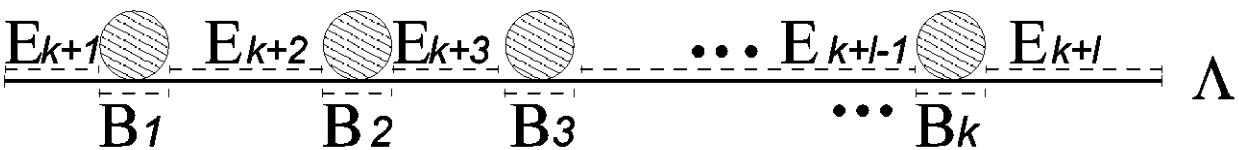
W_i's denote **unoccupied space** and vary in size.

B_i's denote **occupied space** by a single particle and their size is equal to particle size.

Building the continuous time, space Markov Chain

Microscopic Arrhenius Spin-Flip Dynamics

Spin – flip rate for particles adsorbing/desorbing from/to the problem domain.



The rates c are calculated from [1]

$$c(i,\sigma) = \begin{cases} c_d \exp(-\beta U(i,\sigma)) & if \quad \sigma(i) = 1 \\ c_a w(i) & if \quad \sigma(i) = 0 \end{cases}$$

where $w(i) = \begin{cases} |E_i| - |V| & if \quad |E_i| > |V| \\ 0 & otherwise \end{cases}$

and $|V| = \max(B_i)$ for i in (1,k)

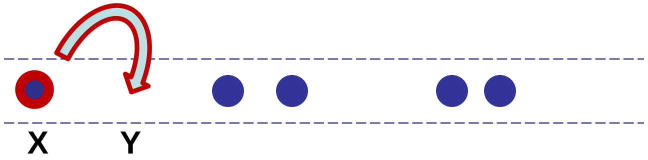
1. Sopasakis. Lattice-free stochastic dynamics, Comm. in Comput. Phys., 2012

Building the continuous time, space Markov Chain

Microscopic Arrhenius Spin-Exchange Dynamics

Following ideas in [1,2] we introduce a lattice-free Arrhenius spin-exchange rate c(x,y,σ),

$$c(x,y,\sigma) = \frac{1}{\tau_0}\sigma(x)[1-\sigma(y)]\,w(y)\,e^{-U(x,\sigma)}$$



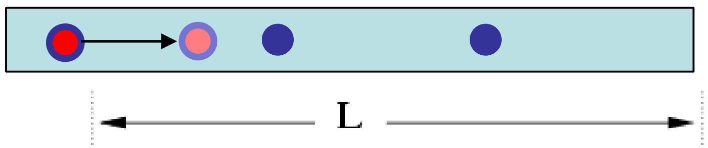
where **parameter** τ_0 denotes the characteristic time of the process and U is the interaction potential.

1. T.M. Liggett, Interacting Particle Systems, Lecture Notes from Trieste, Springer 2002.
2. Katsoulakis, Majda and S., *Nonlinearity*, 19(5), 2006.

Incorporating the Physics and Creating the ASEP

We define the interaction potential $U(x,t) = \sum_{\substack{z \neq x \\ z \in \Lambda}} J(x-y)\sigma(z)$

where $J(r) = \begin{cases} J_0, & \text{if } 0 < r < L \\ 0, & \text{otherwise} \end{cases}$



Here **parameter** *Jo* denotes the strength of the interactions,

and **parameter** *L* denotes the range of interactions.

This potential enforces:
- Vehicles do not move backwards
- Local effect of the interactions

Incorporating multi-lane interactions

Let's look once again at the rate functional to move forward

$$c(x,y,\sigma) = \frac{1}{\tau_0}\sigma(x)[1-\sigma(y)]\,w(y)\,e^{-U(x,\sigma)}$$

We incorporate lane-changing via an additional anisotropy type potential.
Thus our **total interaction potential** now consists of:

$$U_T(x) = U(x) + \boxed{U_a(x)}$$

where $U_a(x) = \sum_{y=nn} \psi(x,y)(1-\sigma(y))$

with $\psi(x,y) = \begin{cases} k_l & \text{if } \ y = \text{left} \\ k_r & \text{if } \ y = \text{right} \\ k_f & \text{if } \ y = \text{forward} \end{cases}$

The Mathematical Model

The process $\{\sigma_t\}_{t\geq 0}$ is a continuous time, continuous space jump Markov Chain on $L^{\infty}(\Sigma, R)$ with generator

$$Mf(\sigma) = \sum_{x\in\Lambda} c(\sigma)[f(\sigma^*) - f(\sigma)]$$

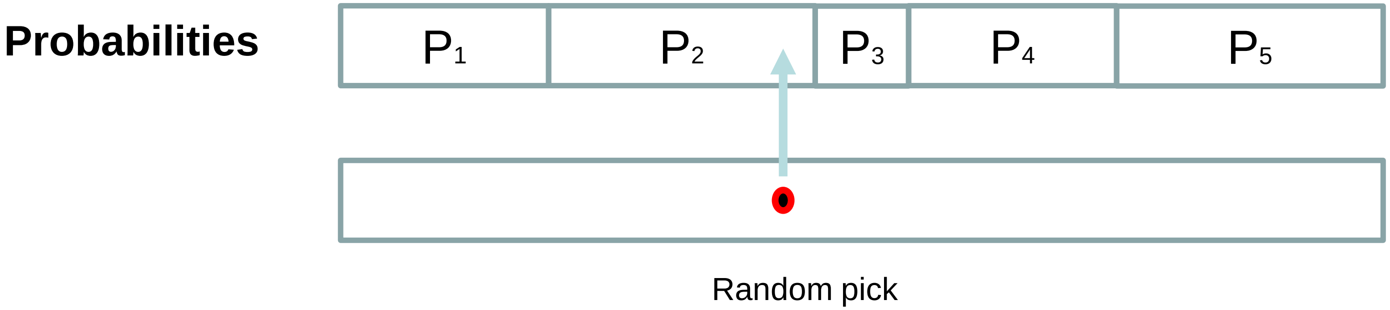
where σ^* denotes a new lattice configuration and $c(\sigma)$ denotes the rate of the stochastic process

The mathematical model therefore is $$\frac{d}{dx} Ef(\sigma) = EMf(\sigma)$$

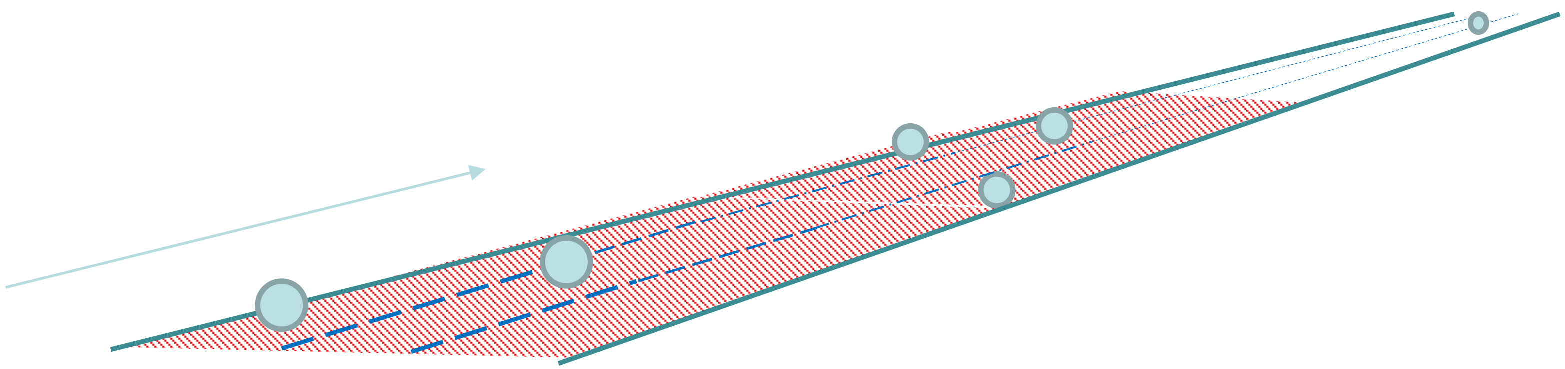
The **probability** of a vehicle moving from x to y during time [t, t+Δt]

is $c(x, y, \sigma)\Delta t + O(\Delta t^2)$

Idealization of Monte Carlo simulation



Monte Carlo moves vehicle 2



A move from location x to location y during time [t, t+Δt]

has probability P = $c(x, y, \sigma)\Delta t + O(\Delta t^2)$

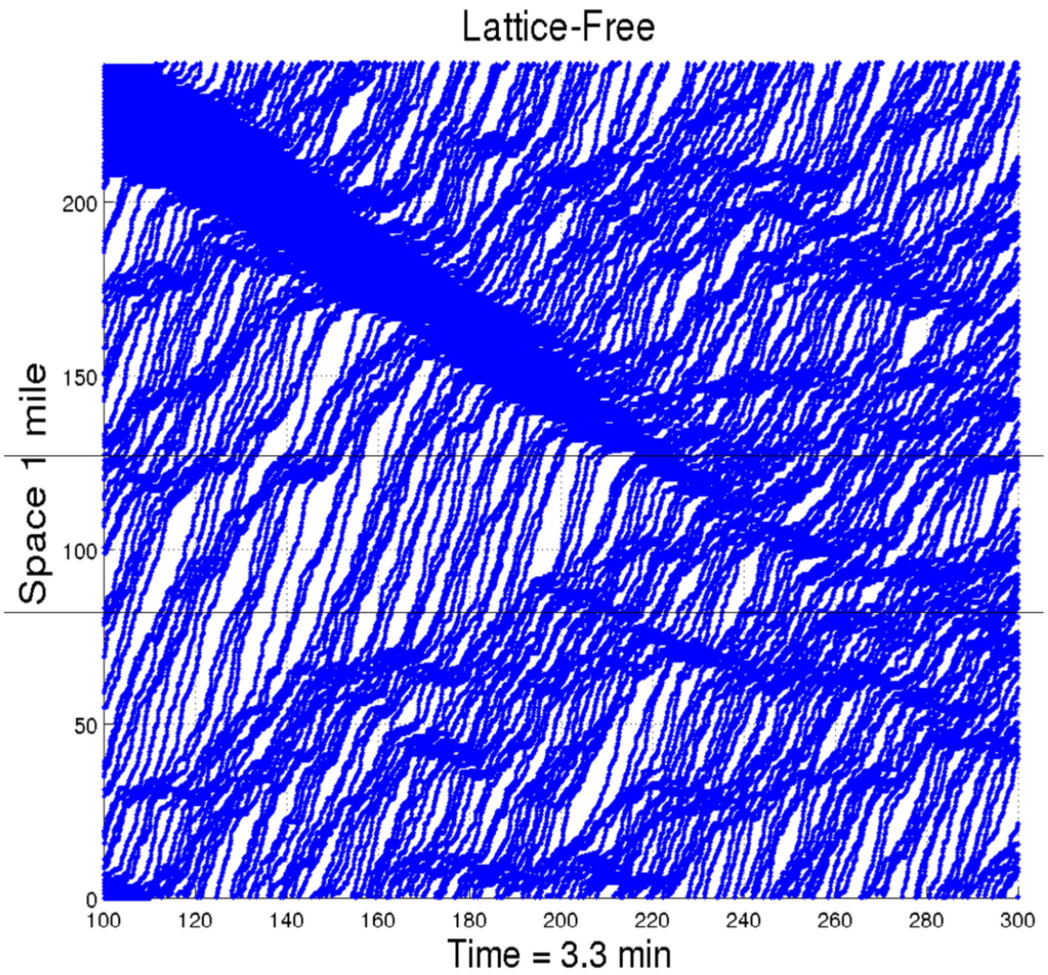
1. Bortz, Kalos and Lebovitz, A new algorithm for Monte Carlo simulations, J. Comp. Phys., 1975

Free Parameters and Calibration

The model is characterized by a number of parameters some of the most important of which are:

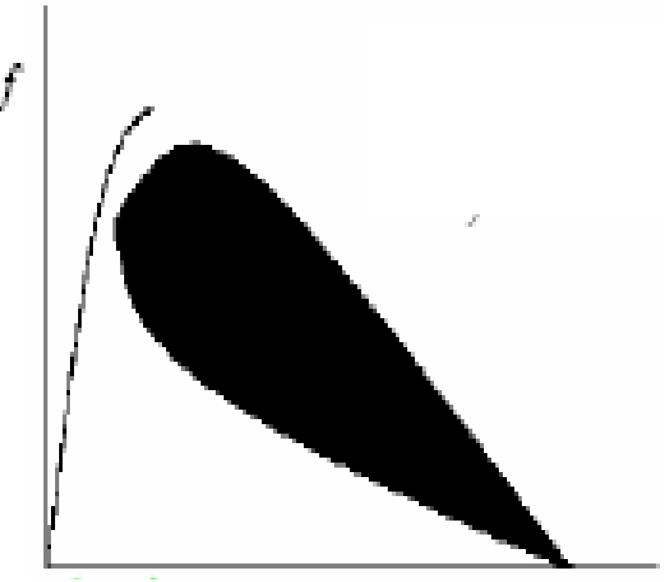
- τ_0 - ***how fast vehicles react*** *to traffic conditions ahead*
- J_0 - ***what is the speed limit*** *of the roadway*
- L - ***how far ahead*** *can drivers perceive traffic (L= 20, 30 or 40 meters)*
- *…*

Testing – Features
Timely breaking/returded acceleration

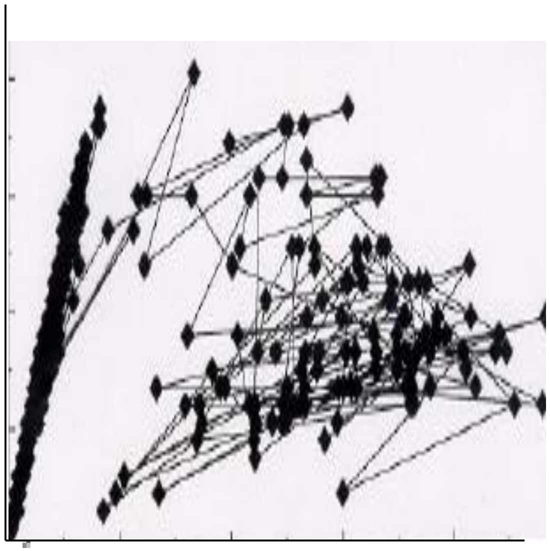


Testing most important requirement
The Fundamental Diagram

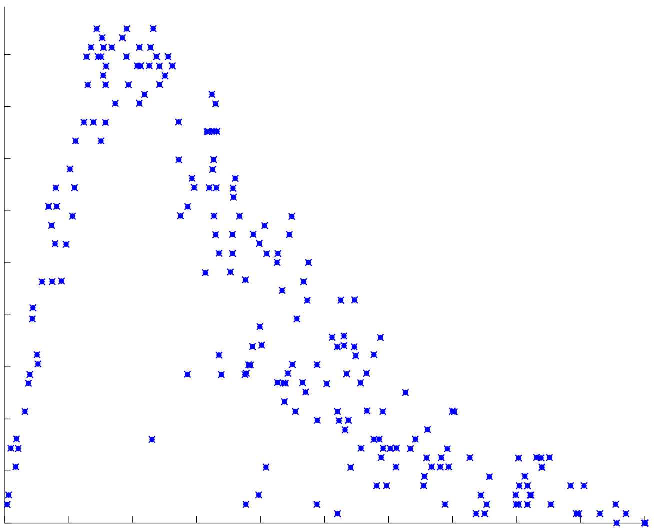
Flow-Density Diagram



Actual Data



Simulated Data



PDE model

Using the LLN[1] and Taylor expansions we obtain the following macroscopic transport equation:

$$u_t + F(u)_z = 0$$

where the PDE flux is $F(u) = c_0 u(1-u)e^{-J \circ u}$

with $J \circ u = \int_z^{\infty} V(y-z)u(y)dy$

1. Sopasakis and Katsoulakis, *Transp. Res. B*, 2012.

PDE model

We obtain the following macroscopic transport equation:

$$\boxed{u_t + F(u)_z = 0}$$

where the PDE flux is $F(u) = c_0 u(1-u)e^{-J \circ u}$

Expanding the convolution

$$J \circ u = \int_0^\infty V(x)u(x+z)dx = J_0 u + J_1 u_z + J_2 u_{zz} + \ldots$$

and approximating the exponential $e^{-J \circ u} \approx e^{-J_0 u}[1 - J_1 u_z - J_2 u_{zz}]$

the traffic model PDE $u_t + F(u)_z = 0$ becomes…

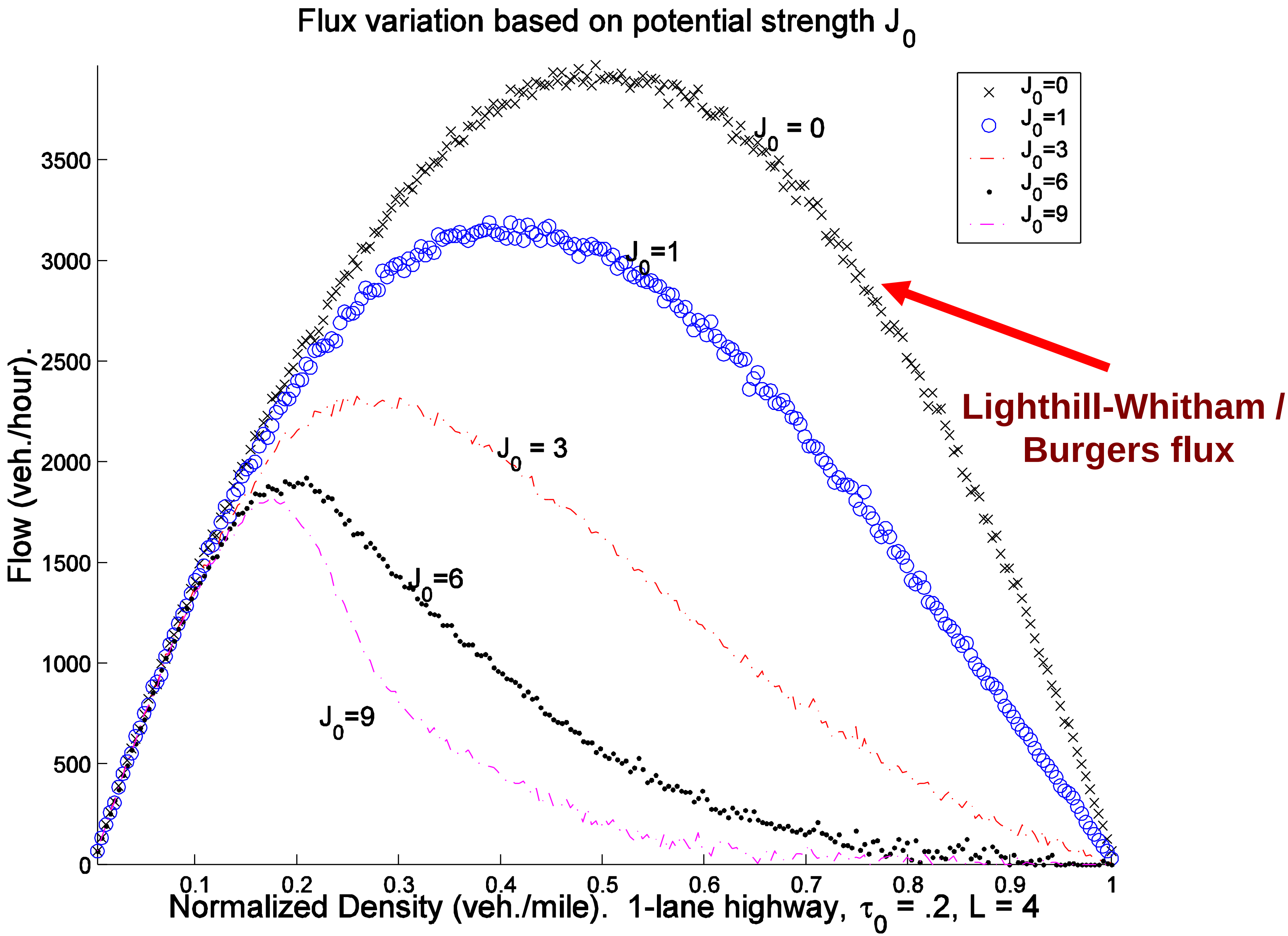
The traffic model PDE, $u_t + F(u)_z = 0$ where $e^{-J \circ u} \approx e^{-J_0 u}[1 - J_1 u_z - J_2 u_{zz}]$

$$u_t + c_0[u(1-u)e^{-J_0 u}]_z = c_0[J_1 u(1-u)e^{-J_0 u} u_z]_z + c_0[J_2 u(1-u)e^{-J_0 u} u_{zz}]_z$$

Note:

- No interactions (J=0):

  Lighhill-Whitham/Burger's eq. $\rightarrow u_t + c_0[u(1-u)]_z = 0$



Flux variation based on potential strength J_0
Flow (veh./hour).
Normalized Density (veh./mile). 1-lane highway, τ_0 = .2, L = 4
J_0=0
J_0=1
J_0=3
J_0=6
J_0=9
J_0 = 0
J_0 =1
J_0 = 3
J_0=6
J_0=9
Lighthill-Whitham / Burgers flux

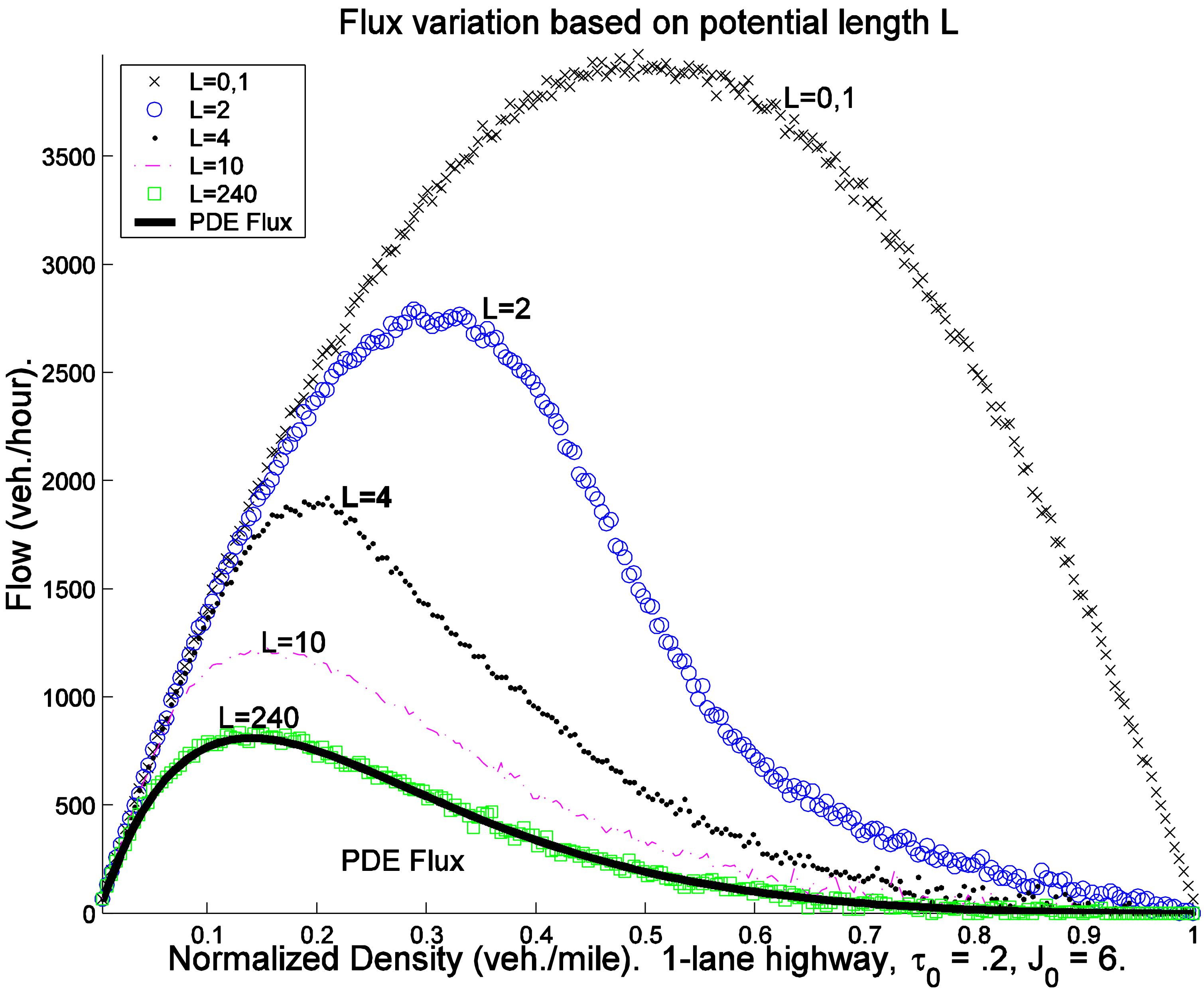
The traffic model PDE, $u_t + F(u)_z = 0$ where $e^{-J \circ u} \approx e^{-J_0 u}[1 - J_1 u_z - J_2 u_{zz}]$

$$u_t + c_0[u(1-u)e^{-J_0 u}]_z = c_0[J_1 u(1-u)e^{-J_0 u} u_z]_z + c_0[J_2 u(1-u)e^{-J_0 u} u_{zz}]_z$$

Note:

- No interactions (J=0):
  Lighhill-Whitham/Burger's eq. $\rightarrow u_t + c_0[u(1-u)]_z = 0$

- Long range (L=N) uniform (J=Jo) interactions:
  Non-local flux $\rightarrow u_t + c_0[u(1-u)e^{-J_0 \bar{u}}]_z = 0$

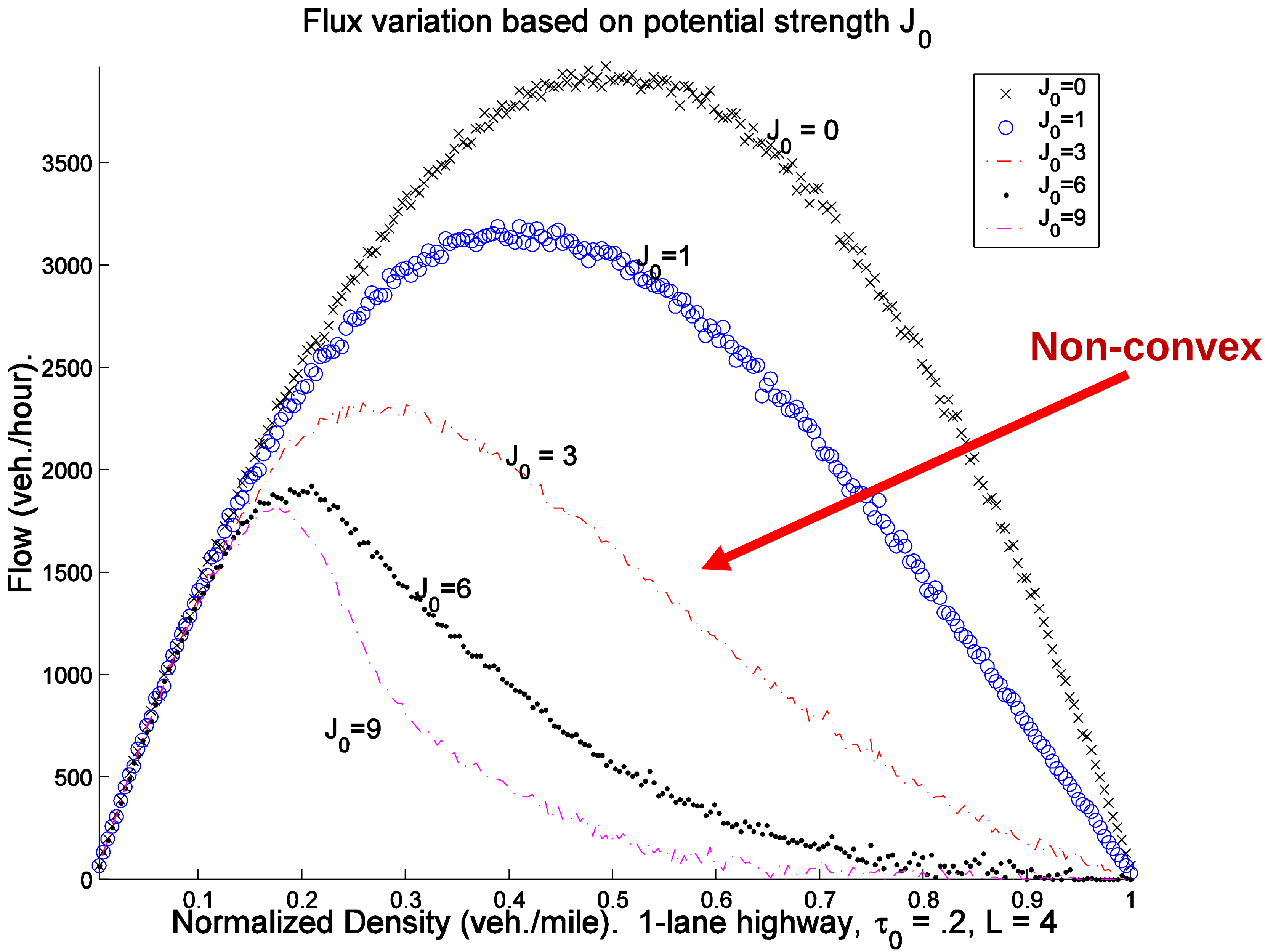

Flux variation based on potential length L
L=0,1
L=2
L=4
L=10
L=240
PDE Flux
L=0,1
L=2
L=4
L=10
L=240
PDE Flux
Flow (veh./hour).
3500
3000
2500
2000
1500
1000
500
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Density (veh./mile). 1-lane highway, τ_0 = .2, J_0 = 6.

The traffic model PDE, $\boxed{u_t + F(u)_z = 0}$ where $e^{-J \circ u} \approx e^{-J_0 u}[1 - J_1 u_z - J_2 u_{zz}]$

$$u_t + c_0[u(1-u)e^{-J_0 u}]_z = c_0[J_1 u(1-u)e^{-J_0 u} u_z]_z + c_0[J_2 u(1-u)e^{-J_0 u} u_{zz}]_z$$

Note:

- No interactions (J=0):
  Lighhill-Whitham/Burger's eq. $\rightarrow u_t + c_0[u(1-u)]_z = 0$
- Long range (L=N) uniform (J=Jo) interactions:
  Non-local flux $\rightarrow u_t + c_0[u(1-u)e^{-J_0 \bar{u}}]_z = 0$
- Including terms up to Jo in the convolution,
  Non-convex flux $\rightarrow u_t + c_0[u(1-u)e^{-J_0 u}]_z = 0$



Flux variation based on potential strength J_0
$J_0=0$
$J_0=1$
$J_0=3$
$J_0=6$
$J_0=9$
$J_0 = 0$
$J_0=1$
$J_0 = 3$
$J_0=6$
$J_0=9$
Non-convex
Flow (veh./hour).
3500
3000
2500
2000
1500
1000
500
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Density (veh./mile). 1-lane highway, τ_0 = .2, L = 4

The traffic model PDE, $u_t + F(u)_z = 0$ where $e^{-J\circ u} \approx e^{-J_0 u}[1 - J_1 u_z - J_2 u_{zz}]$

$$u_t + c_0[u(1-u)e^{-J_0 u}]_z = c_0[J_1 u(1-u)e^{-J_0 u} u_z]_z + c_0[J_2 u(1-u)e^{-J_0 u} u_{zz}]_z$$

Note:

- No interactions (J=0):
  Lighhill-Whitham/Burger's eq. $\rightarrow u_t + c_0 u(1-u) = 0$

- Long range (L=N) uniform (J=Jo) interactions:
  Non-local flux $\rightarrow u_t + c_0[u(1-u)e^{-J_0 \bar{u}}]_z = 0$

- Including terms up to Jo in the convolution,
  Non-convex flux $\rightarrow u_t + c_0[u(1-u)e^{-J_0 u}]_z = 0$

- Terms up to J_1
  Nonlinear diffusive LWR type

$$\rightarrow u_t + c_0[u(1-u)e^{-J_0 u}]_z = c_0[J_1 u(1-u)e^{-J_0 u} u_z]_z$$

- Full model is higher order dispersive (KDV type?)
  with nonlinear coefficients

Outline

- *Part 1*. The lattice-free microscopic dynamics: Desirable features for specific applications
  - Specialized potentials: Look-ahead dynamics
  - Conservation
- *Part 2*. Parameters and Calibration
  - Information Theory
  - Neural Networks
- *Part 3.* Simulations & comparisons

Summary of terms ot keep track of: Local transition rates and probabilities

The process $\{\sigma_t\}_{t\geq 0}$ is defined as a continuous time Markov Chain (CTMC) on the (high dimensional) state space $E \equiv \Sigma^{|\Lambda|}$

Mathematically it is defined completely by specifying the **local transition rates** $c^{\theta}(\sigma,\sigma')$ where $\theta \in R^k$ is a vector of the model parameters.

The transition rates determine the updates (jumps) from any current state σ to a random new state σ'.

Let us also define the total rate $\lambda^{\theta}(\sigma) = \sum_{\sigma' \in E} c^{\theta}(\sigma,\sigma')$

which is the intensity of the exponential waiting time for a jump from the state σ

Finally the **transition probabilities** are simply given by $p^{\theta}(\sigma,\sigma') = \dfrac{c^{\theta}(\sigma,\sigma')}{\lambda^{\theta}(\sigma)}$

Examples of local transition rates $c(\sigma, \sigma')$

Microscopic Arrhenius Spin-Exchange Dynamics

The Arrhenius spin-exchange rate[1,2] c(x,y,σ),

$$c(x, y, \sigma) = \frac{1}{\tau_0} \sigma(x)\,[1 - \sigma(y)]\, w(y)\, e^{-U(x,\sigma)}$$

Similarly, the Arrhenius spin-flip rate c(x,σ) at lattice site x and current state configuration σ is given by

$$c(i, \sigma) = \begin{cases} c_d \exp(-\beta U(i, \sigma)) & if \quad \sigma(i) = 1 \\ c_a w(i) & if \quad \sigma(i) = 0 \end{cases}$$

with adsorption/desoprion constants, c_a, c_d

1. T.M. Liggett, Interacting Particle Systems, Lecture Notes from Trieste, Springer 2002.
2. Katsoulakis, Majda and Sopasakis, *Nonlinearity*, 19(5), 2006.

Information theoretic parametrization approach

We **propose two tools** which quantify information loss in time series (path-space):

- The pathwise Relative Entropy Rate (RER) performs sensitivity analysis by perturbing the parameter space (identifies the most important parameters to be adjusted)
- The pathwise Fisher Information Matrix (FIM) identifies the unimportant model parameters (performs model reduction)

The Relative Entropy

Definition. The **relative entropy (**or Kullback-Leiber divergence) between two probability measures P and Q is defined via

$$R(P\,|\,Q) = \int \log\left(\frac{dP}{dQ}\right) dP$$

The Relative Entropy can be thought as a "distance" or more precisely a semi-metric between the two measures.

Properties: (i) $R(P\,|\,Q) \geq 0$

(ii) $R(P\,|\,Q) = 0$ iff $P = Q$ a.e.

The relative entropy is not necessarily symmetric and may not satisfy the triangle inequality

The relative entropy measures loss/change of information.

The Relative Entropy Rate

The Relative Entropy Rate (RER) can be thought as the change of information per unit time[1]

$$H(Q^{\theta}_{[0,T]} \mid Q^{\theta+\varepsilon}_{[0,T]}) = E_{\mu^{\theta}}\left[\sum_{\sigma'\in E} c^{\theta}(\sigma,\sigma')\log\frac{c^{\theta}(\sigma,\sigma')}{c^{\theta+\varepsilon}(\sigma,\sigma')} - \lambda^{\theta}(\sigma) + \lambda^{\theta+\varepsilon}(\sigma)\right]$$

The **estimator for RER** over time T for a given perturbation vector **ε**

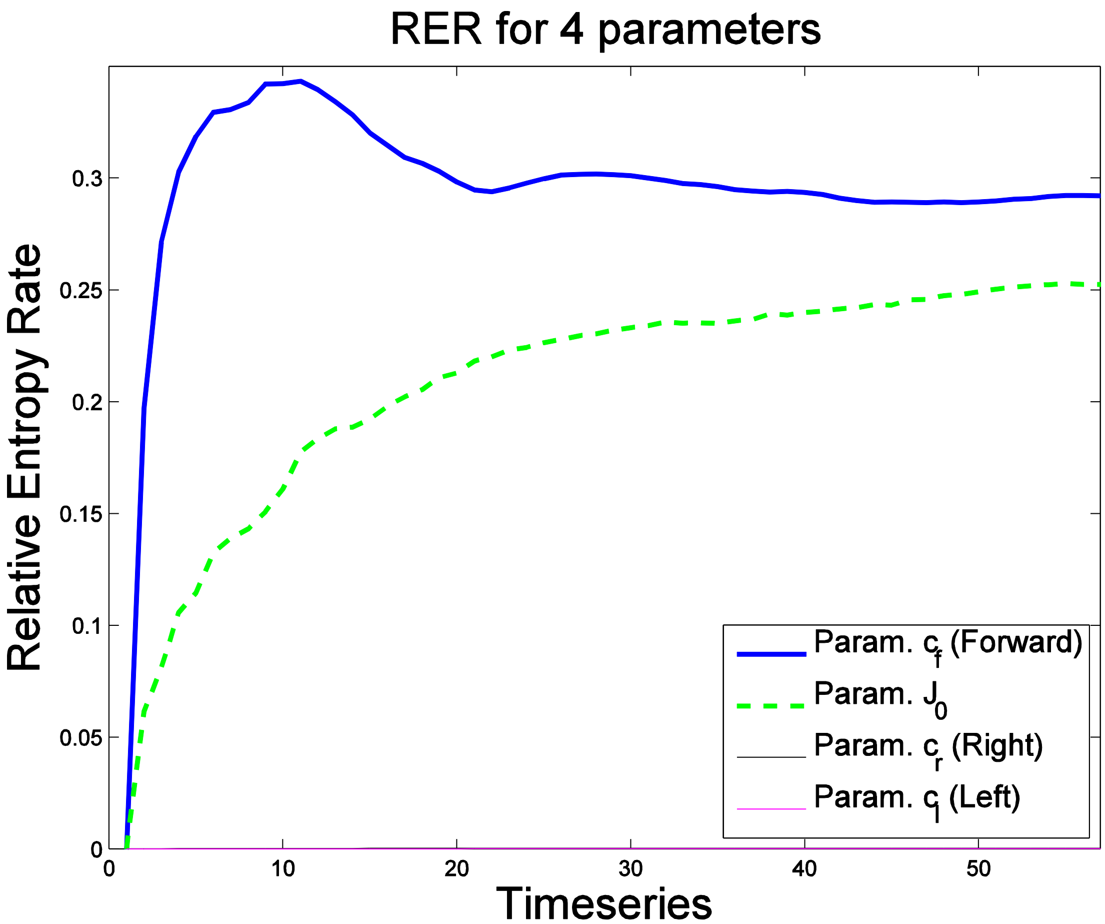
$$R_c(\varepsilon) = \frac{1}{T}\sum_{i=0}^{n-1}\left[\Delta t_i\left[\sum_{j=1}^{M} c_j(\sigma_i;\theta)\log\frac{c_j(\sigma_i;\theta)}{c_j(\sigma_i;\theta+\varepsilon)}\right] + c(\sigma_i;\theta+\varepsilon) - c(\sigma_i;\theta)\right]$$

where $c(\sigma_i; p) = \sum_{j=1}^{M} c_j(\sigma_i; p)$.

- method constitutes an *inverse dynamics Monte Carlo approach* and reconstructs the Markov chain by performing sensitivity analysis
- Infers information regarding path distribution: steady states + transient dynamics
- RER is an observable+statistical estimators=> computationally tractable using (fast) molecular solvers.

1. Sopasakis and Katstoulakis, Transp. Res. B, 2016

Relative Entropy Rate versus time



Four trajectories are computed and presented above. Each trajectory is created by separately perturbing each of the parameters above.

Properties:

- RER works in *path-space.* Analyzing perturbations of trajectories in order to identify *sensitive* parameters
- Extract statistical properties from *non-equilibrium dynamics*
- Optimization of parameter vector θ is possible (note that RER is quadratic in θ).

The Fisher Information Matrix

Definition. The Fisher Information Matrix, F_H for a given probability density $Q^{\theta}_{[0,T]}$ is defined to be the Hessian of the Relative Entropy.

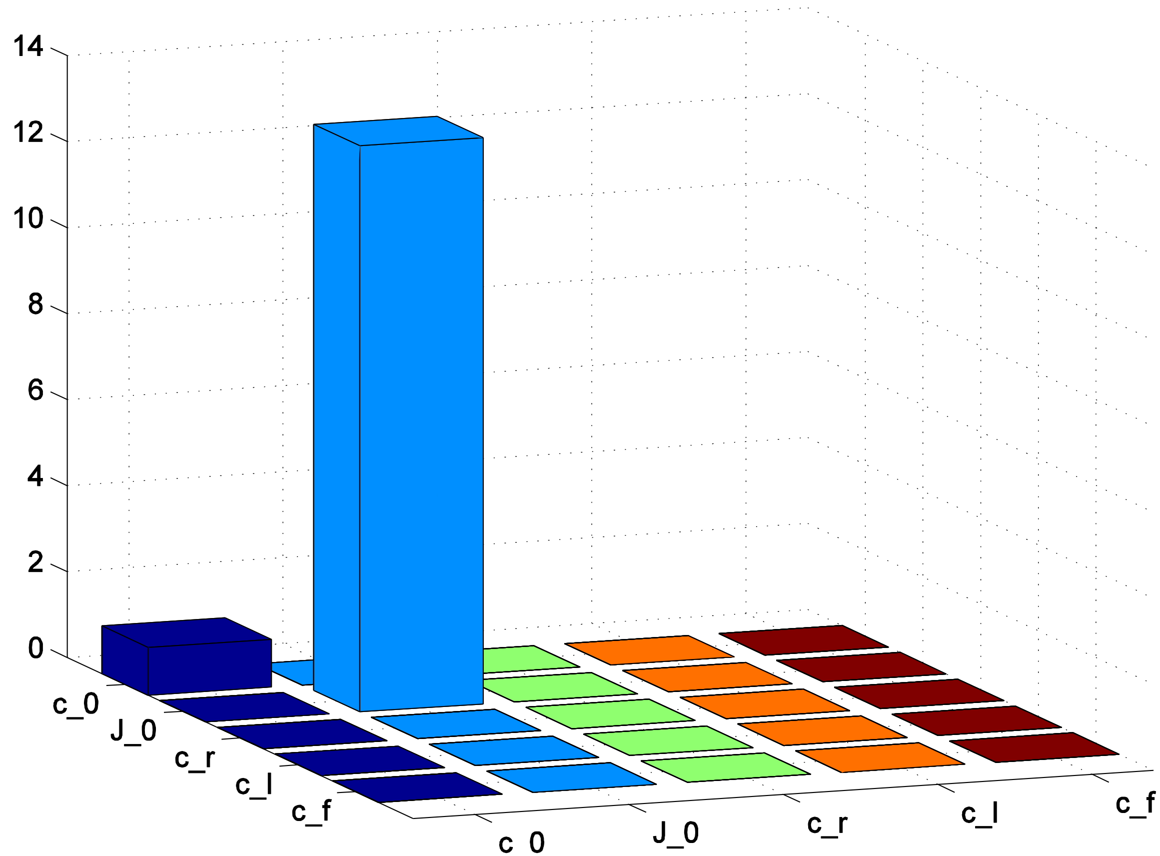
$$F_H(Q^{\theta}_{[0,T]}) = E_{\mu^{\theta}}\left[\sum_{\sigma'\in E} c^{\theta}(\sigma,\sigma')\nabla_{\theta}\log c^{\theta}(\sigma,\sigma')\nabla_{\theta}\log c^{\theta}(\sigma,\sigma')^{T}\right]$$

The **estimator for FIM** is

$$F_n(p) = \frac{1}{T}\sum_{i=0}^{n}\Delta t_i \sum_{j=1}^{M} c_j(\sigma_i;p)\nabla_j \log c_j(\sigma_i;p)\nabla_j \log c_j(\sigma_i;p)^{T}$$

(depends only on **analytic derivatives** of the propencity functions c – computed in advance)

* Spectral analysis of the FIM reveals the least/most sensitive parameters/directions



The Fisher Information Matrix for 5 parameters

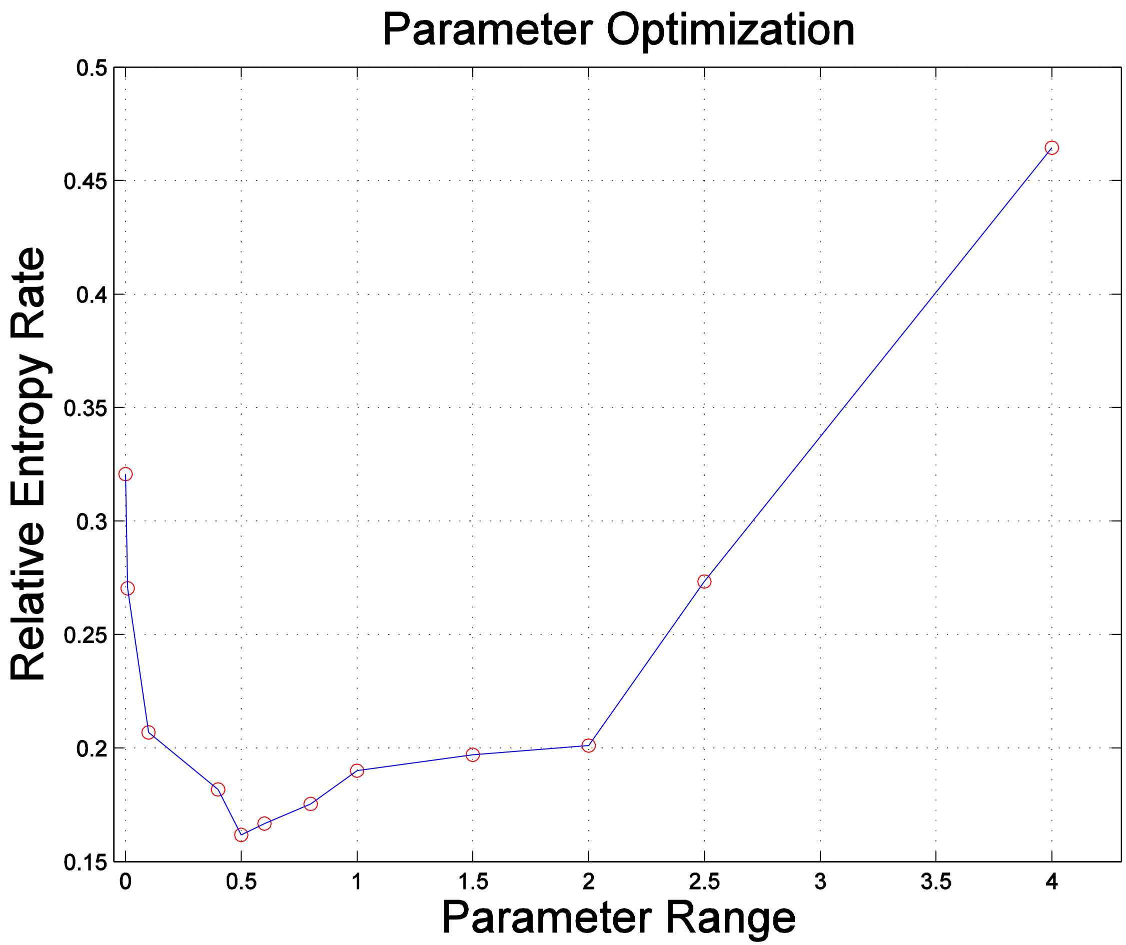
Given synthetic data we computer the FIM for five model parameters in order to understand which of those are important. According to the current dynamics only c_0 and J_0 are important for this model. We can therefore eliminate the parameters c_r, c_l and c_f.

Strategy

- *First apply FIM*.
  this results into a reduced traffic model since we can eliminate the unimportant (given the current dynamics) model parameters

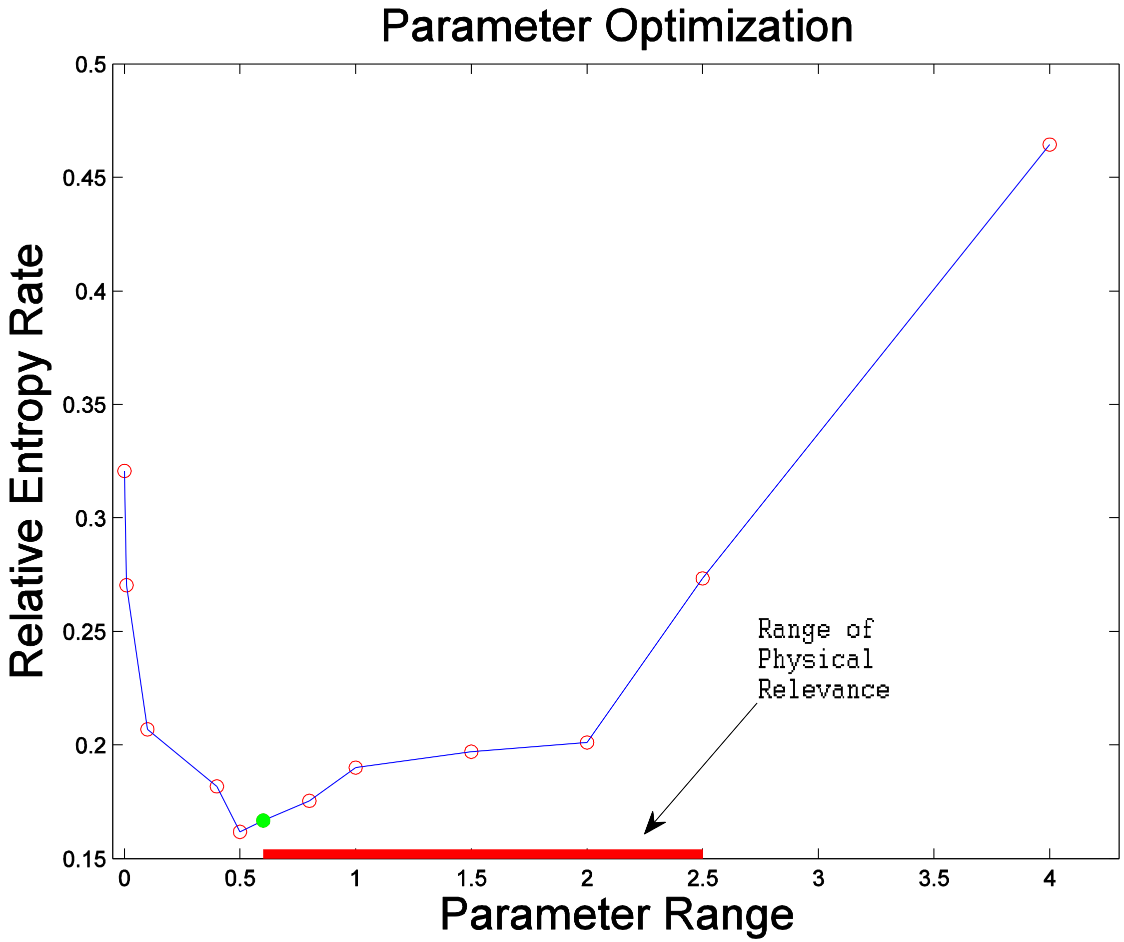
- *Then apply RER*.
  We now perturb the remaining (reduced) model parameters in order to obtain better values for them which represent the current dynamics.

- *Optional Step*. Optimize.
  This step allows us to also, if desired, obtain the best values for those model parameters (although that may not be necessary).



The RER for the parameter c_0

The RER is a convex function and therefore its minimum can be computed. This computation gives an indication as to how to obtain better values for the parameter c_0.

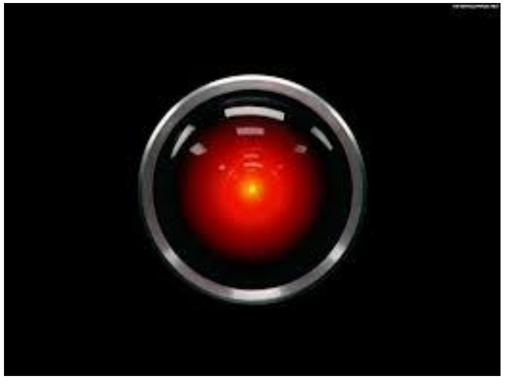


The RER for the parameter c_0

However the actual minimum may be outside the range of physical relevance for the traffic model.

Outline

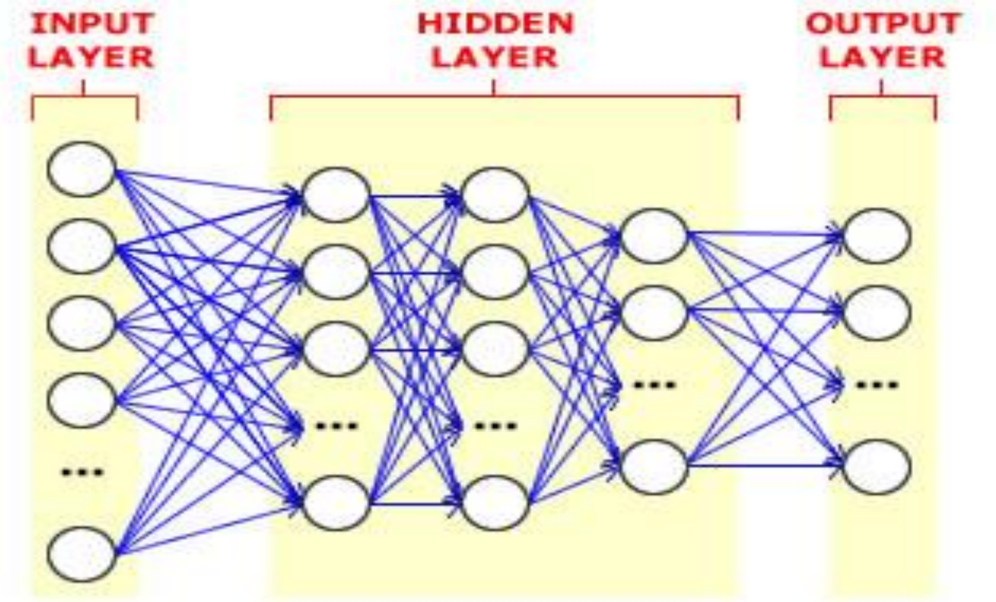
- *Part 1*. The lattice-free microscopic dynamics: Desirable features for specific applications
  - Specialized potentials: Look-ahead dynamics
  - Conservation
- *Part 2*. Parameters and Calibration
  - Information Theory
  - Neural Networks
- *Part 3.* Simulations & comparisons



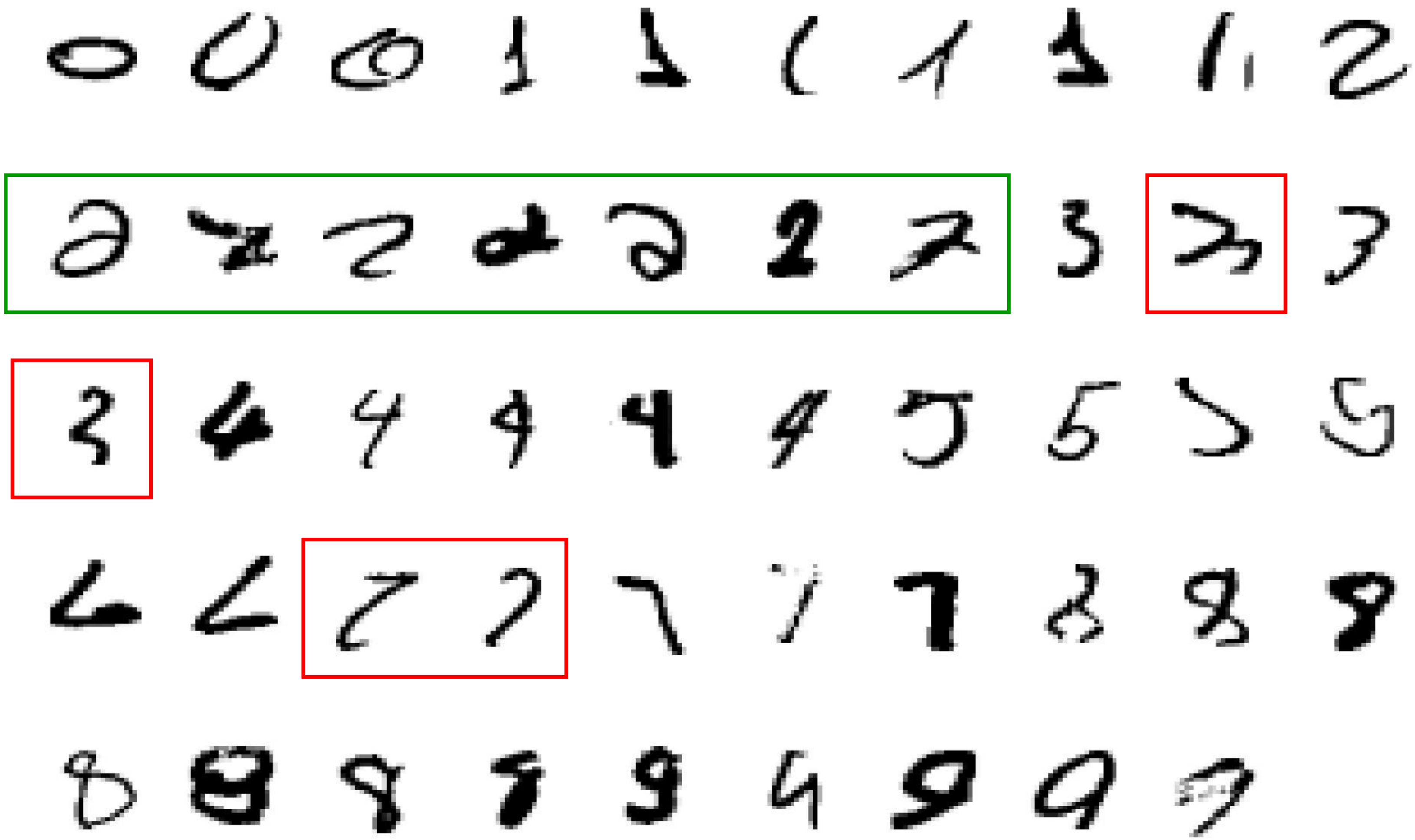
I am sorry Dave, I am afraid I cannot do that!

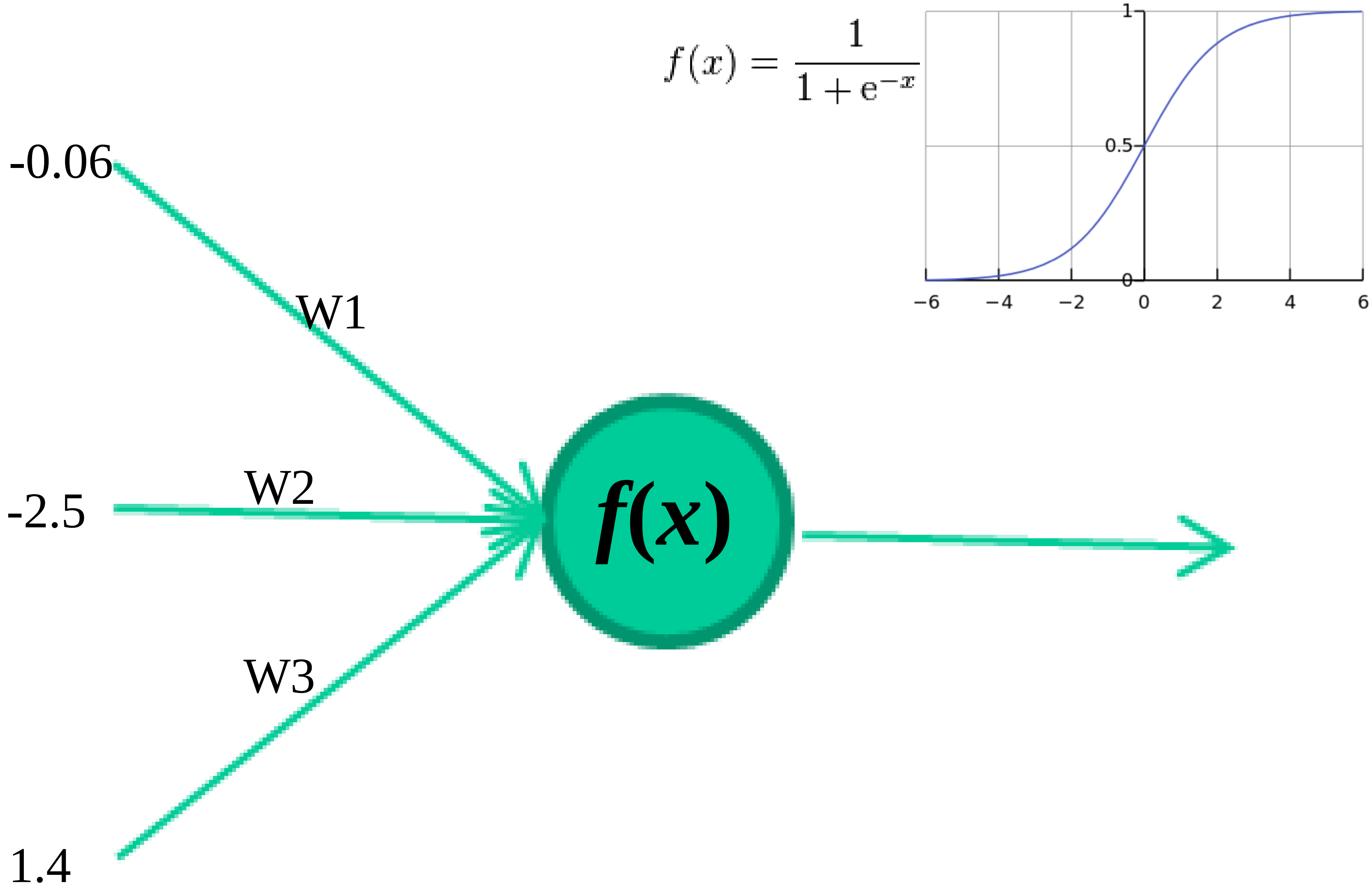
Neural Networks

- Branch of AI (machine learning) imitating how the brain synapses process information
- Basic components: **input layer**, **hidden layers**, **output layer** each comprised of **nodes** and connected by **weights**



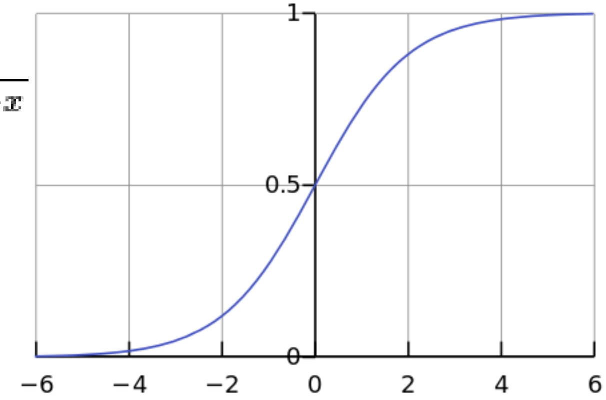
A typical machine learning task
What is a “2”?

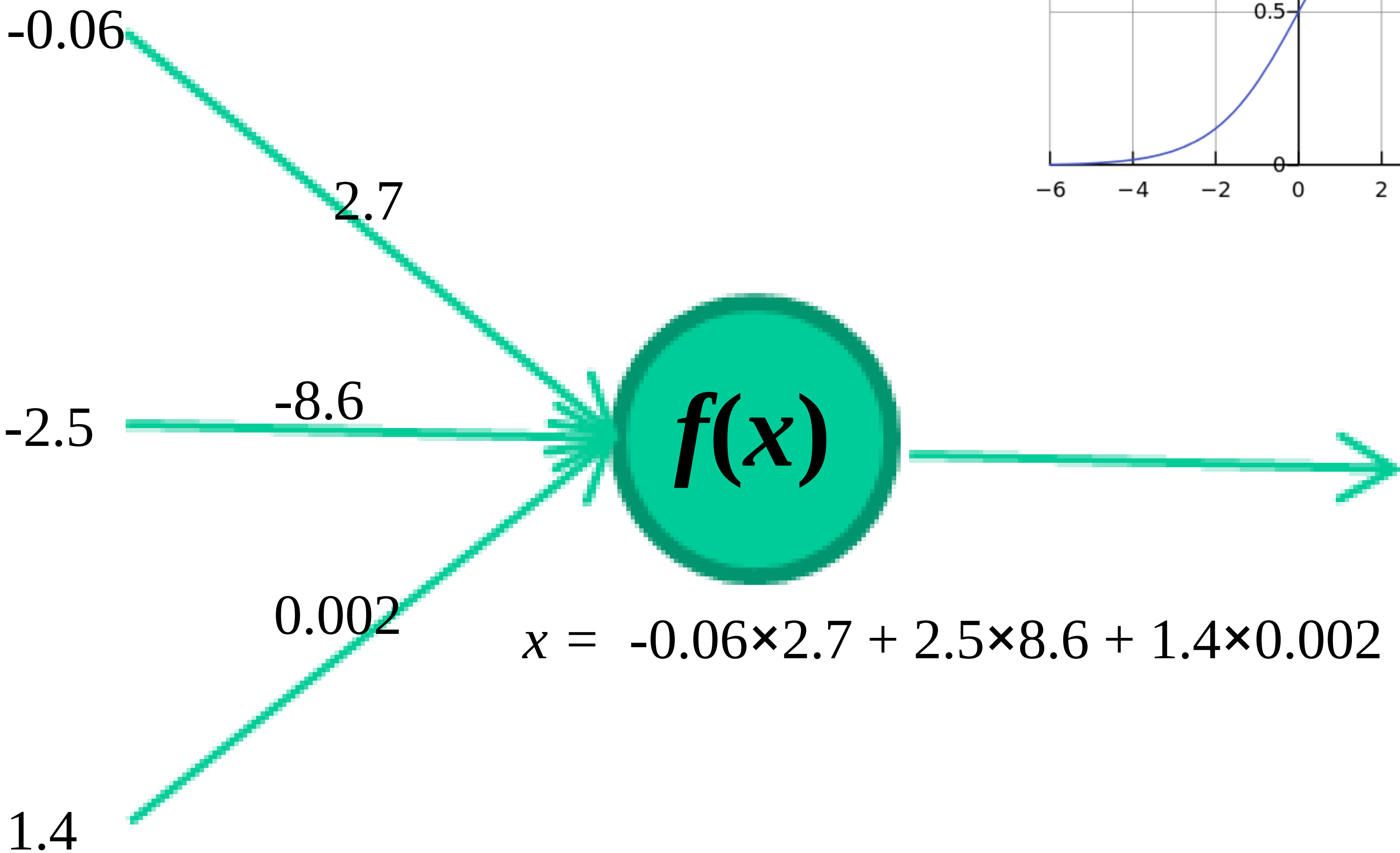




$f(x) = \frac{1}{1 + e^{-x}}$
-0.06
W1
-2.5
W2
W3
1.4
f(x)
1
0.5
0
−6
−4
−2
0
2
4
6

$$f(x) = \frac{1}{1+e^{-x}}$$



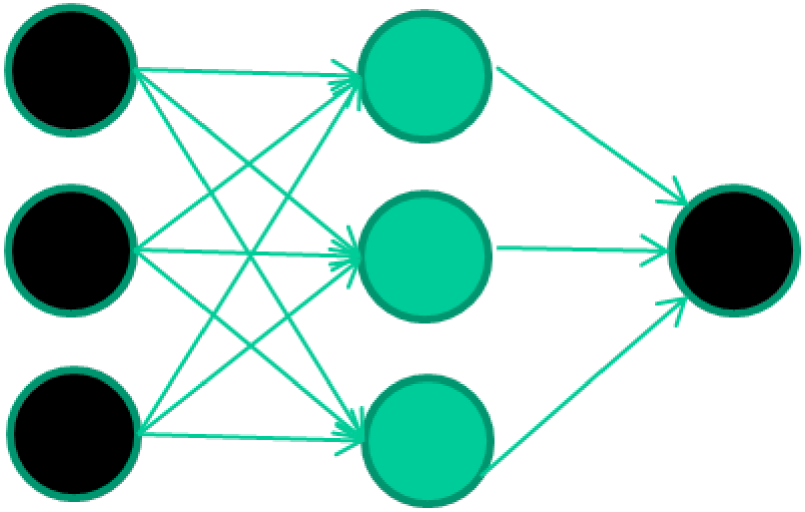


$x =$ -0.06×2.7 + 2.5×8.6 + 1.4×0.002 = 21.34

A dataset

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

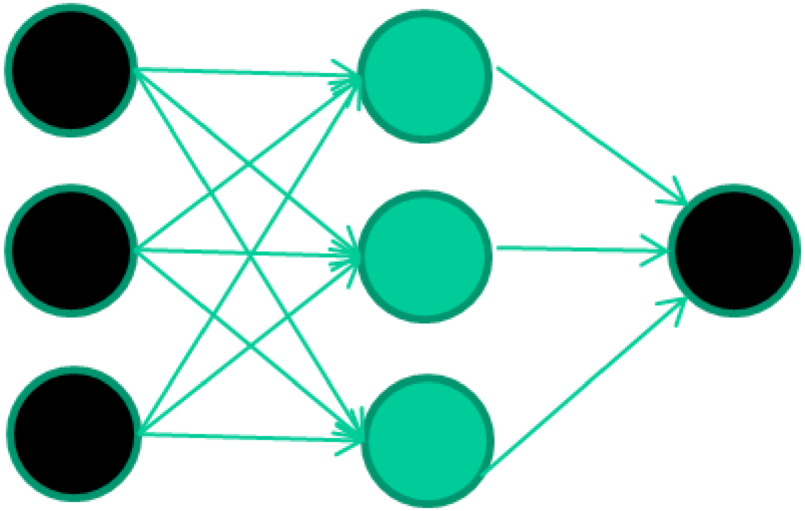


Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

Initialise with random weights

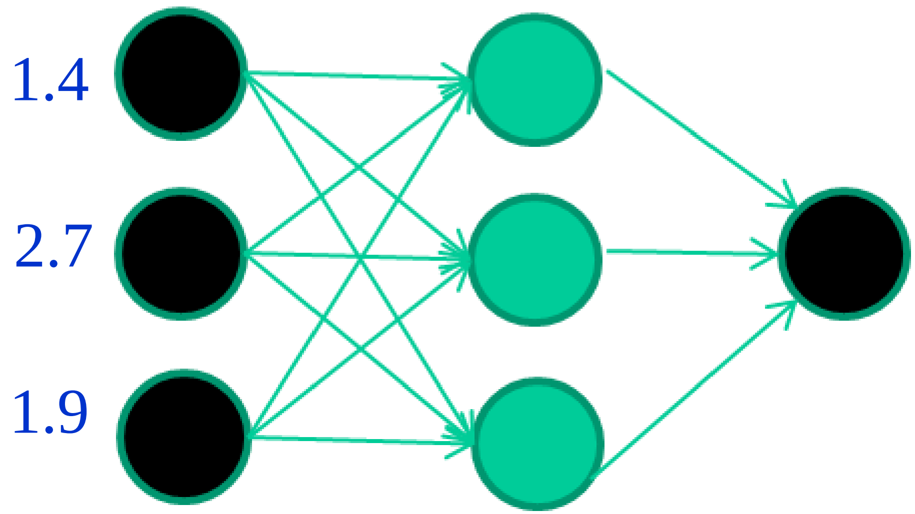


Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

Present a training pattern

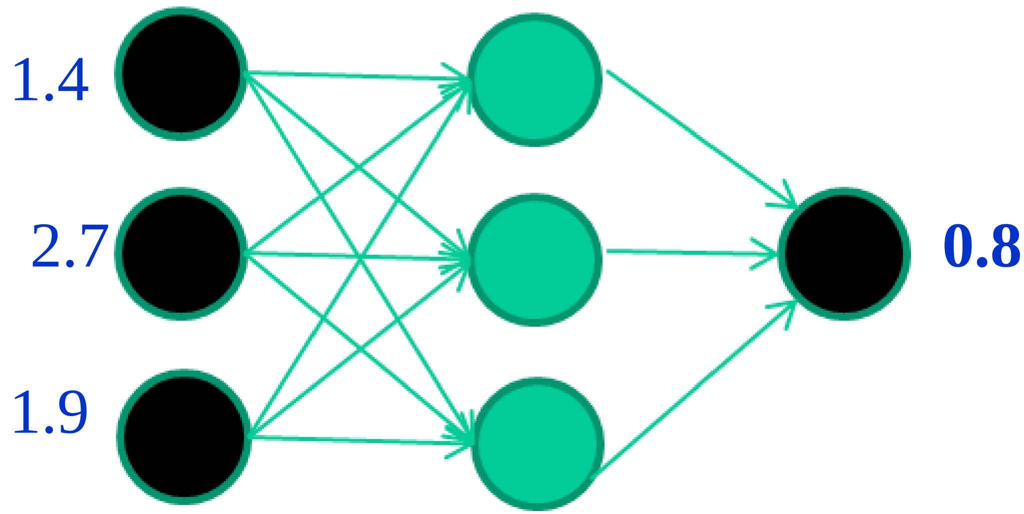


Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

Feed it through to get output

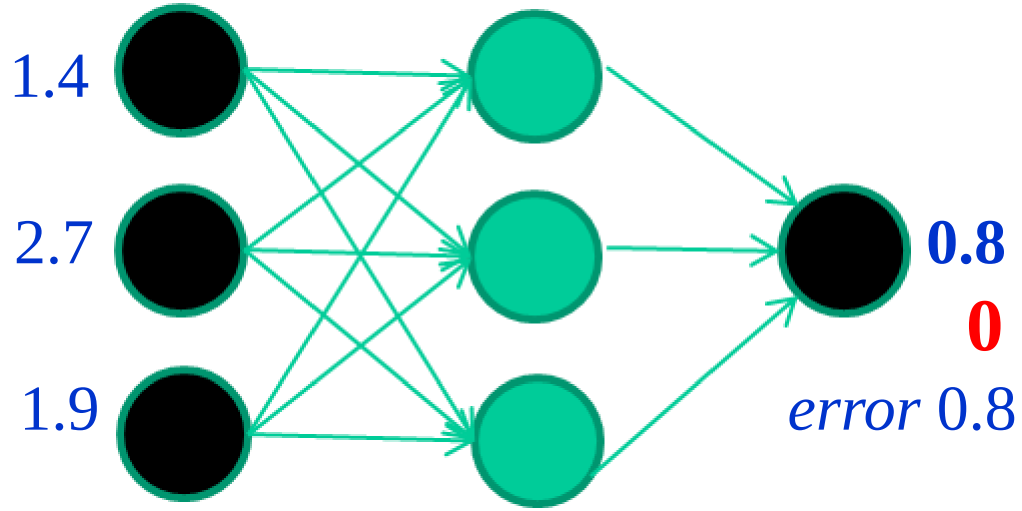


Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

Compare with target output

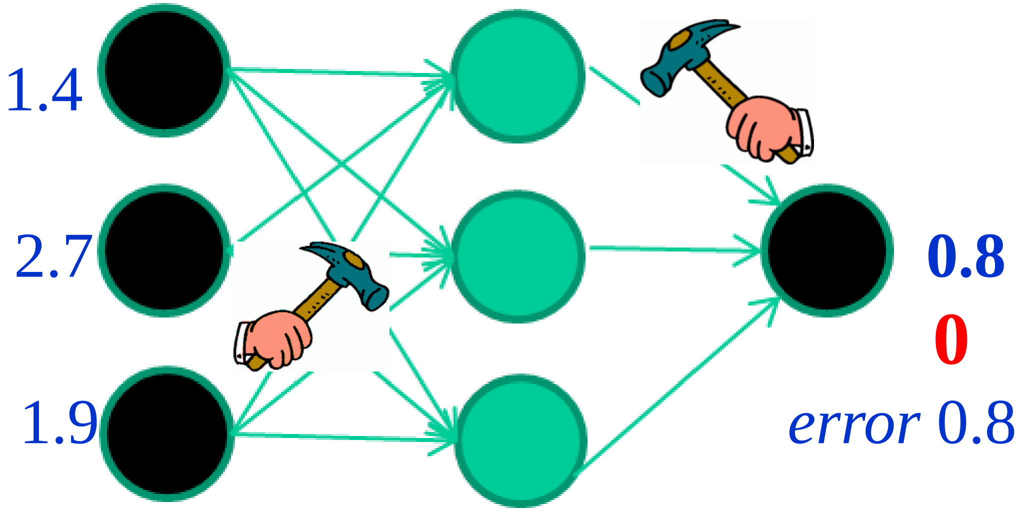


Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

Adjust weights based on error

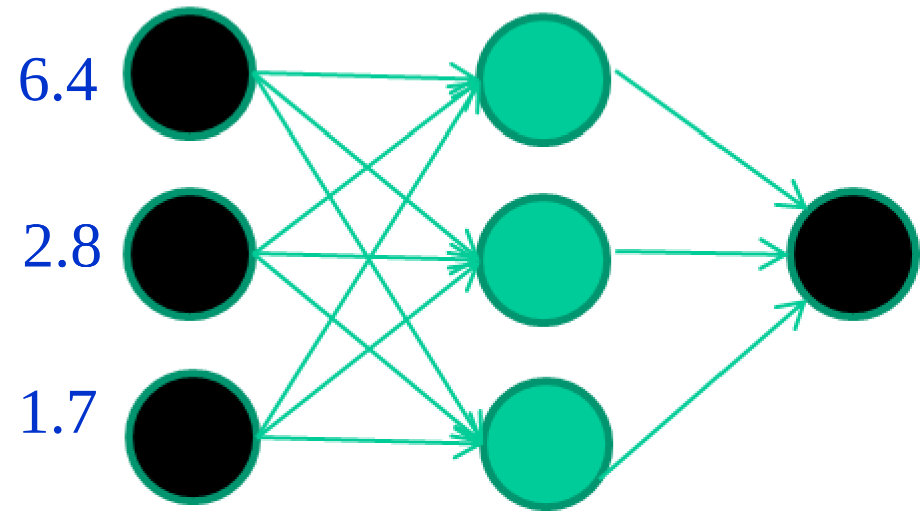


Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

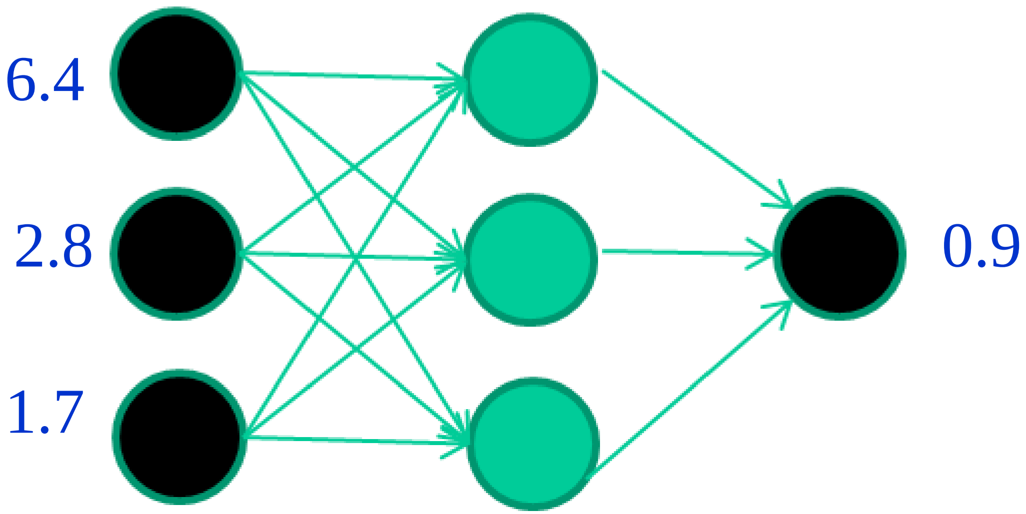
Present a training pattern



Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0
	etc …		

Feed it through to get output

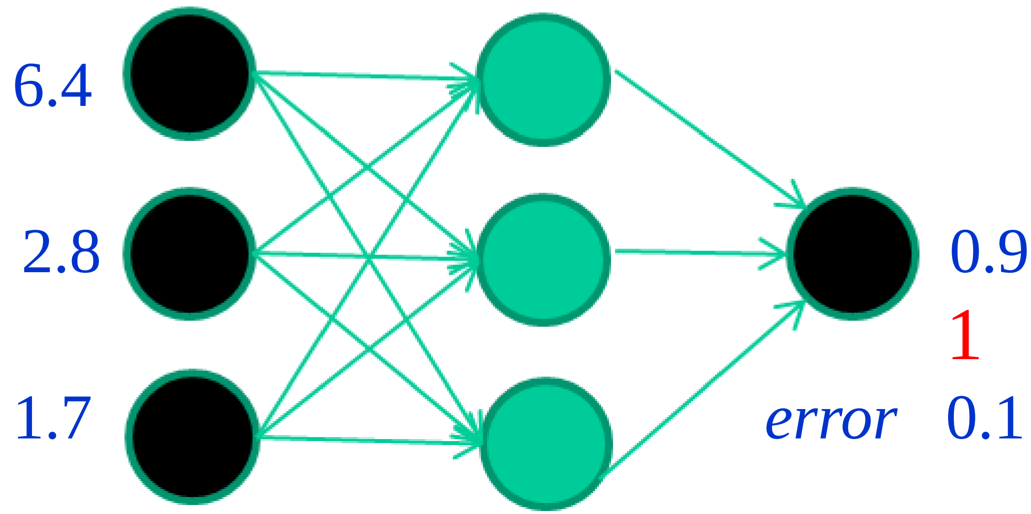


Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

Compare with target output

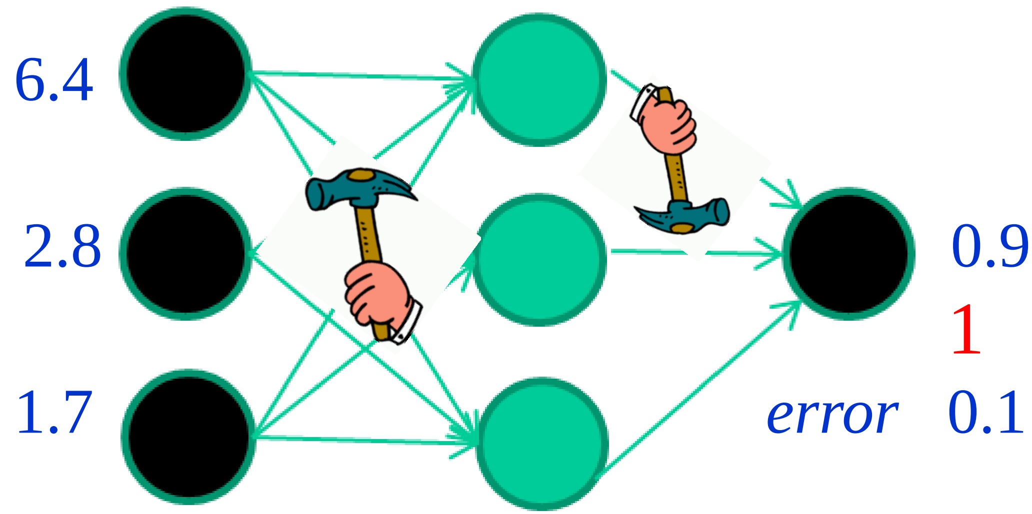


Adjust weights based on error

Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

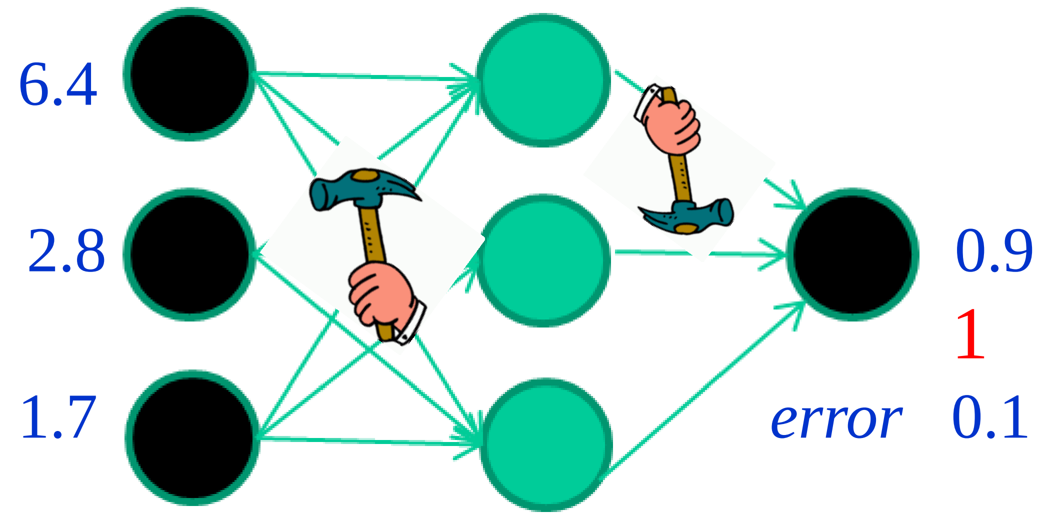


Training data

Fields			***class***
1.4	2.7	1.9	0
3.8	3.4	3.2	0
6.4	2.8	1.7	1
4.1	0.1	0.2	0

etc …

Keep repeating ….



Use Backpropagation Algorithm
to make
changes that will reduce the error

Backpropagation

- Goal minimize network error: $E(x,w) = \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{M}\left\| p_{n,m} - a_{n,m} \right\|^2$

Each partial derivative of grad E is made up of derivatives of succesive activation functions and weights $\left[\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \ldots, \frac{\partial E}{\partial w_N} \right]^T$

IDEA: iteratively follow in the direction of the negative gradient (steepest descent direction) until we arrive at the stopping criterion:

$$\nabla_w E = 0$$

To achieve this, at each step, we update the weights based on its corresponding partial derivative $-\gamma \frac{\partial E}{\partial w_n}$

Thus the updating rule is...

Gauss - Newton

- Thus the updating rule is : $w_{m+1} = w_m - \gamma \frac{\partial E}{\partial w_m}$

but it can be computationally slow...

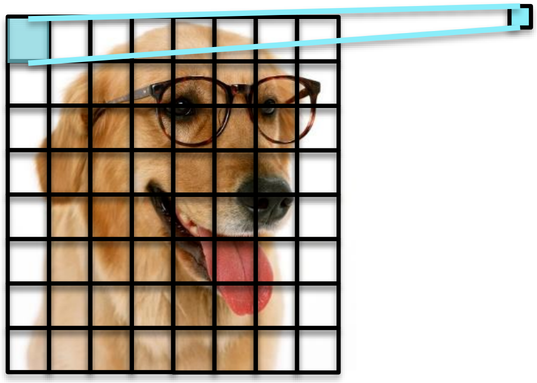
- On the other hand Gauss-Newton is computationally fast but not always stable (not always invertible H)

$$w_{m+1} = w_m - H^{-1} \frac{\partial E}{\partial w_m}$$

- We adapt it using the Levenberg-Marquardt algorithm

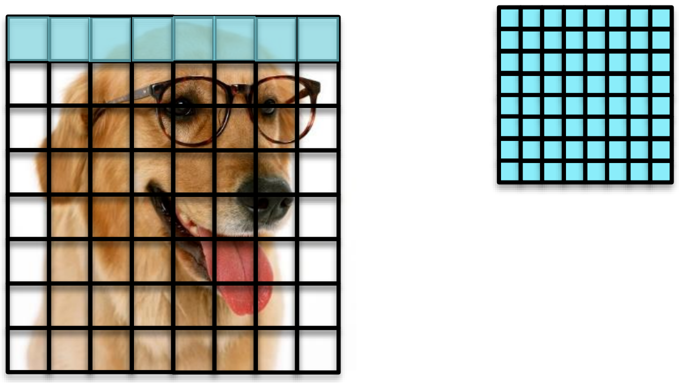
$$w_{m+1} = w_m - (H + \mu I)^{-1} \frac{\partial E}{\partial w_m}$$

CNN for Image Classification



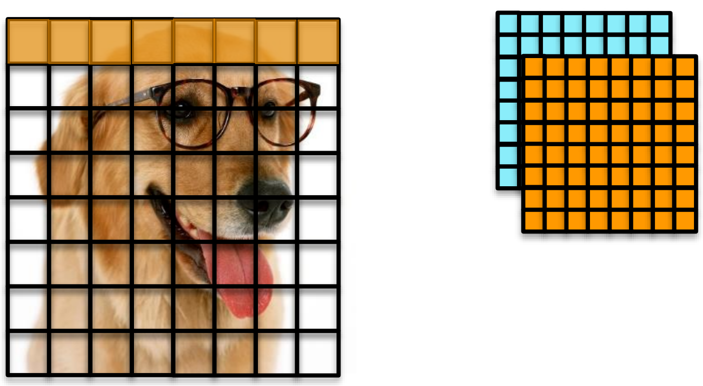
Convolutional Layer

CNN for Image Classification



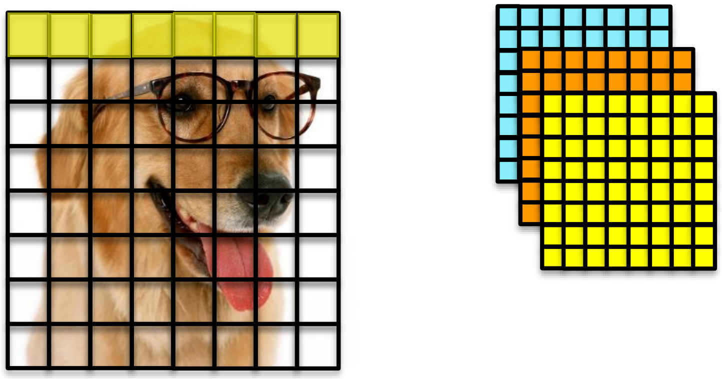
Convolutional Layer

CNN for Image Classification



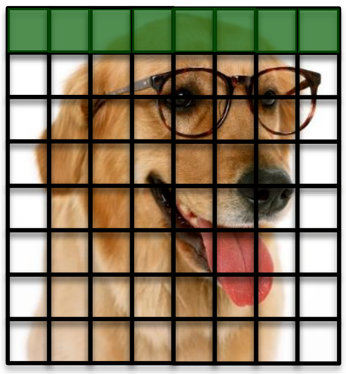
Convolutional Layer

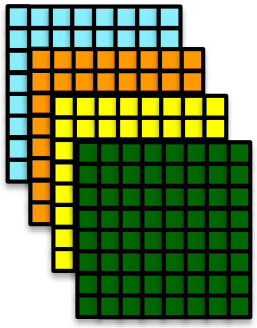
CNN for Image Classification



Convolutional Layer

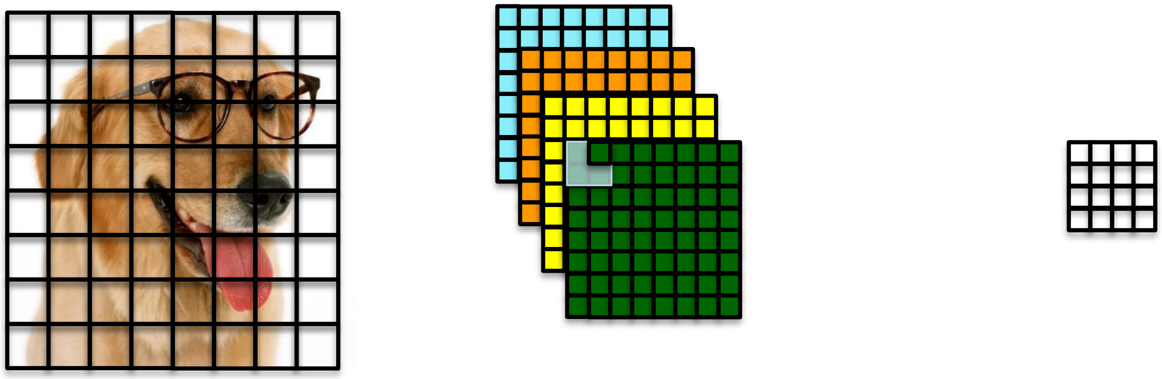
CNN for Image Classification





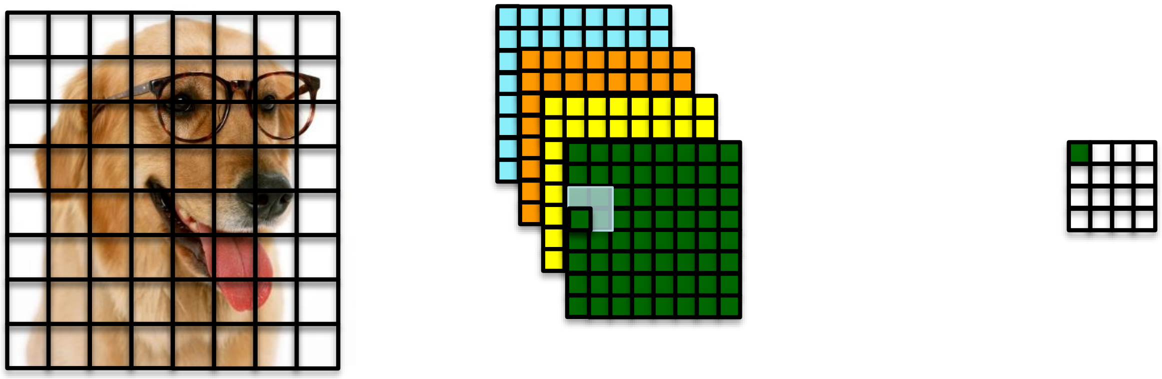
Convolutional Layer

CNN for Image Classification



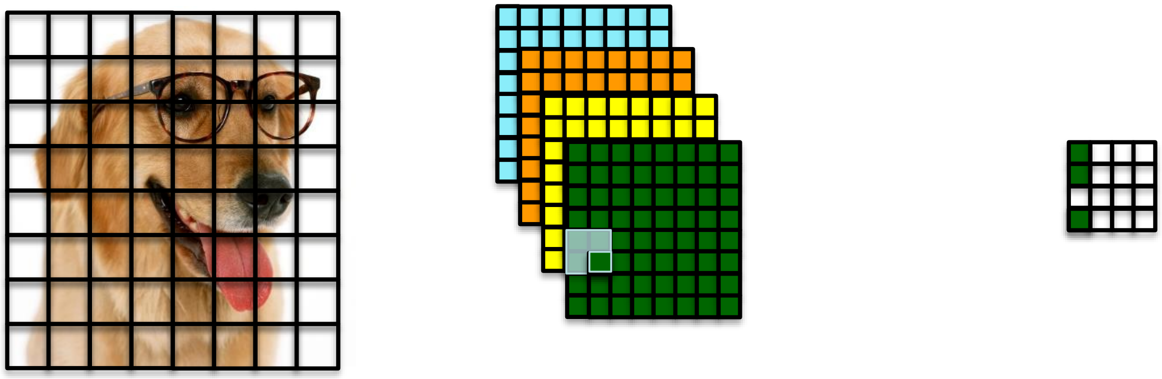
Max Pooling

CNN for Image Classification



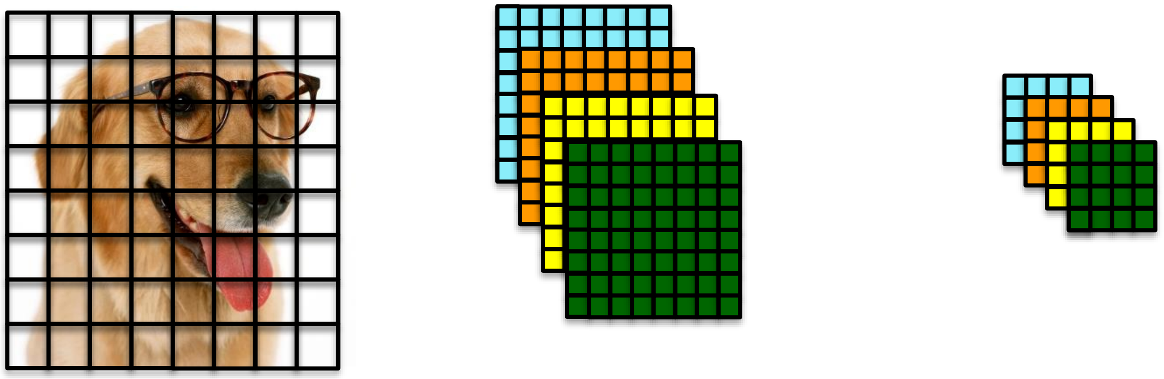
Max Pooling

CNN for Image Classification



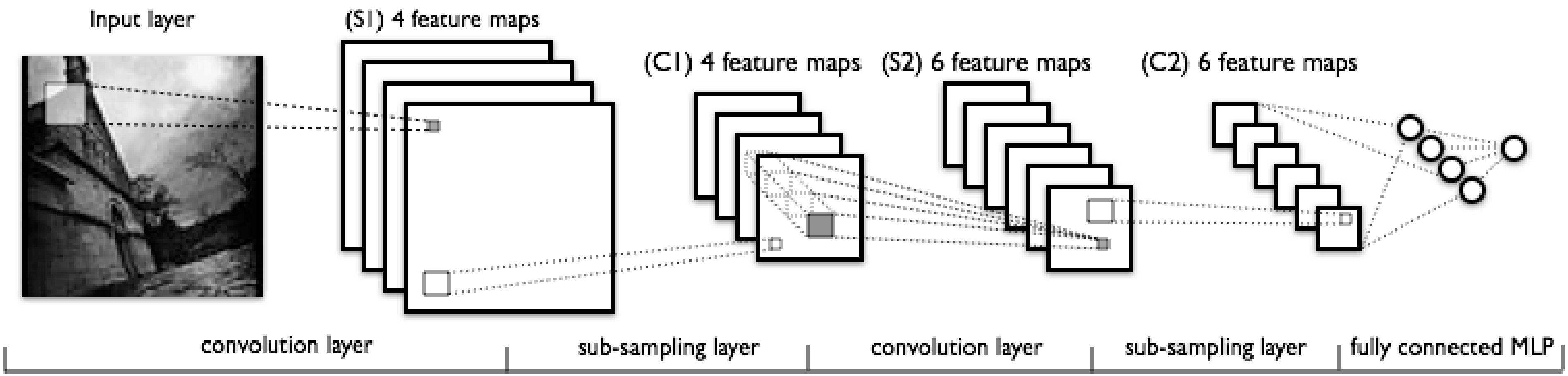
Max Pooling

CNN for Image Classification



Max Pooling

CNN for Image Classification



Potential around this vehicle

http://deeplearning.net/tutorial/lenet.html

Outline

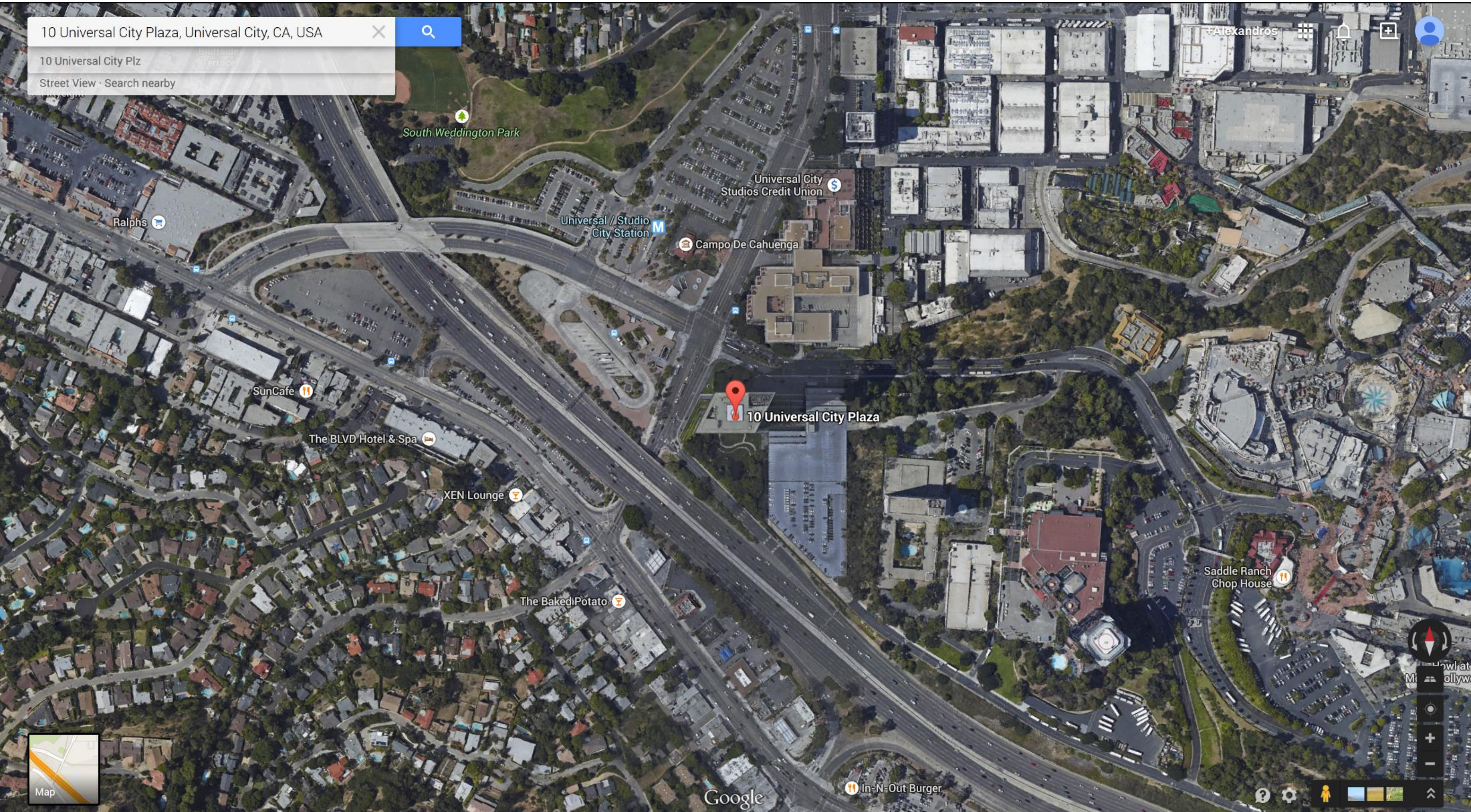
- *Part 1*. The lattice-free microscopic dynamics: Desirable features for specific applications
  - Specialized potentials: Look-ahead dynamics
  - Conservation

- *Part 2*. Parameters and Calibration
  - Information Theory
  - Neural Networks

- *Part 3.* Simulations & comparisons

Test case: a real highway – the NGSIM project

- Highway U.S. 101 near Los Angeles, in California
- 5 lanes with entrances and exits.
- 15 minutes intervals of very detailed data



A Monte Carlo Multi-Lane, Multi-Class Vehicle simulation…

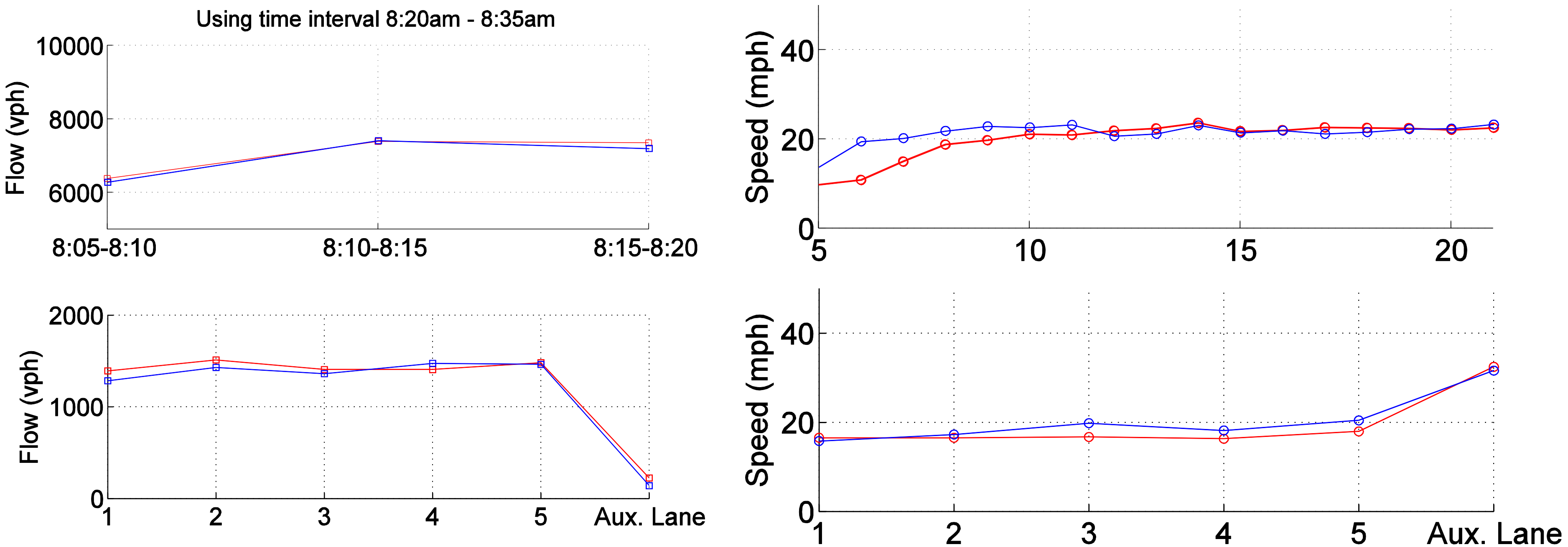
Real Data

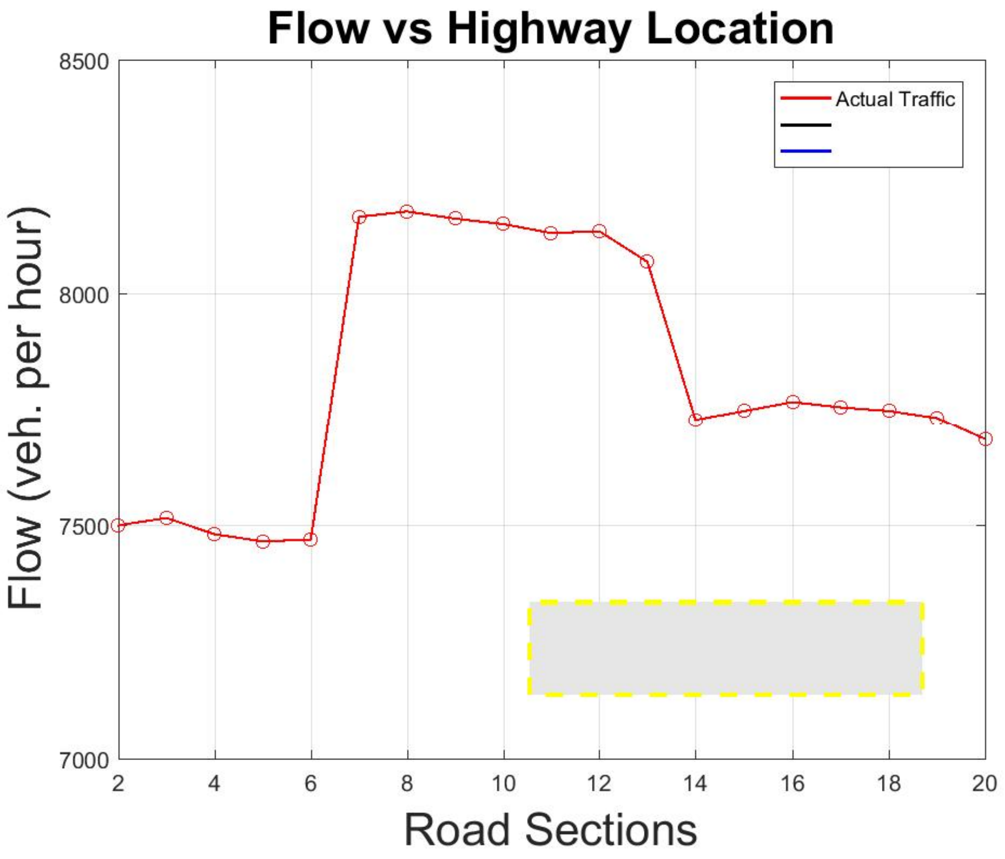
Simulation

carSimulator.html

Test case: Stoc. Sims + RER vs reality

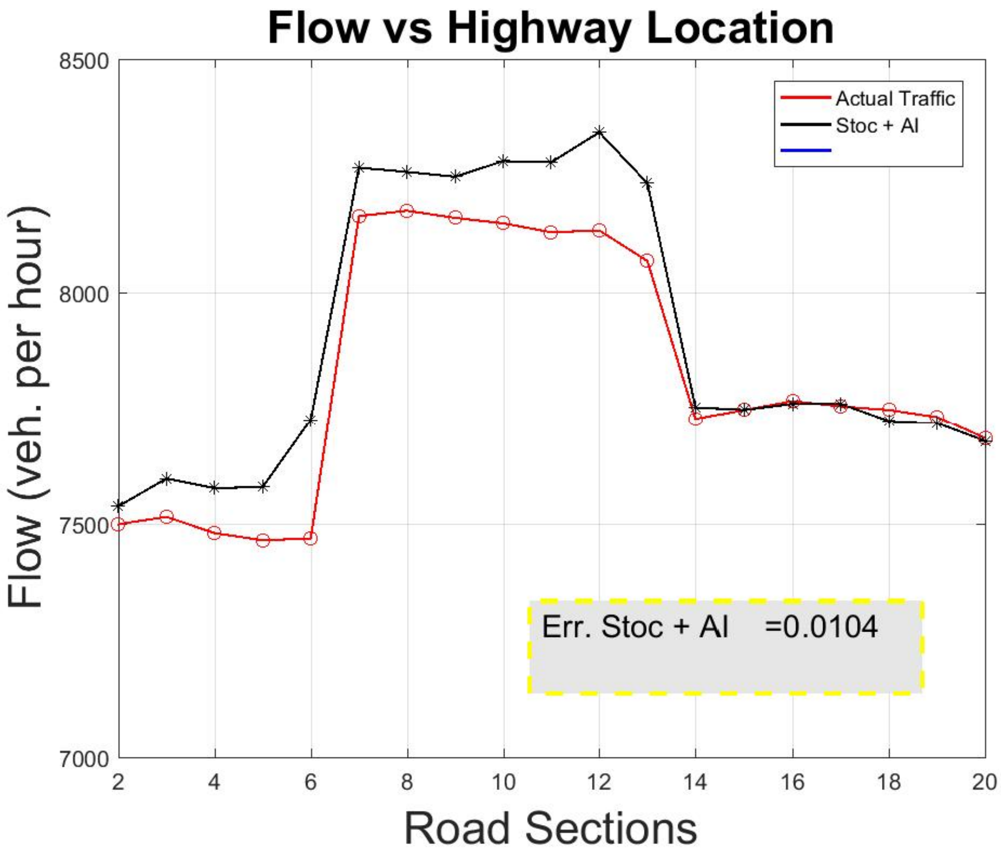
- Highway U.S. 101, Los Angeles, California
- 5 lanes with entrances and exits.
- 15 minutes intervals of very detailed rush hour data: 8:05am to 8:20am





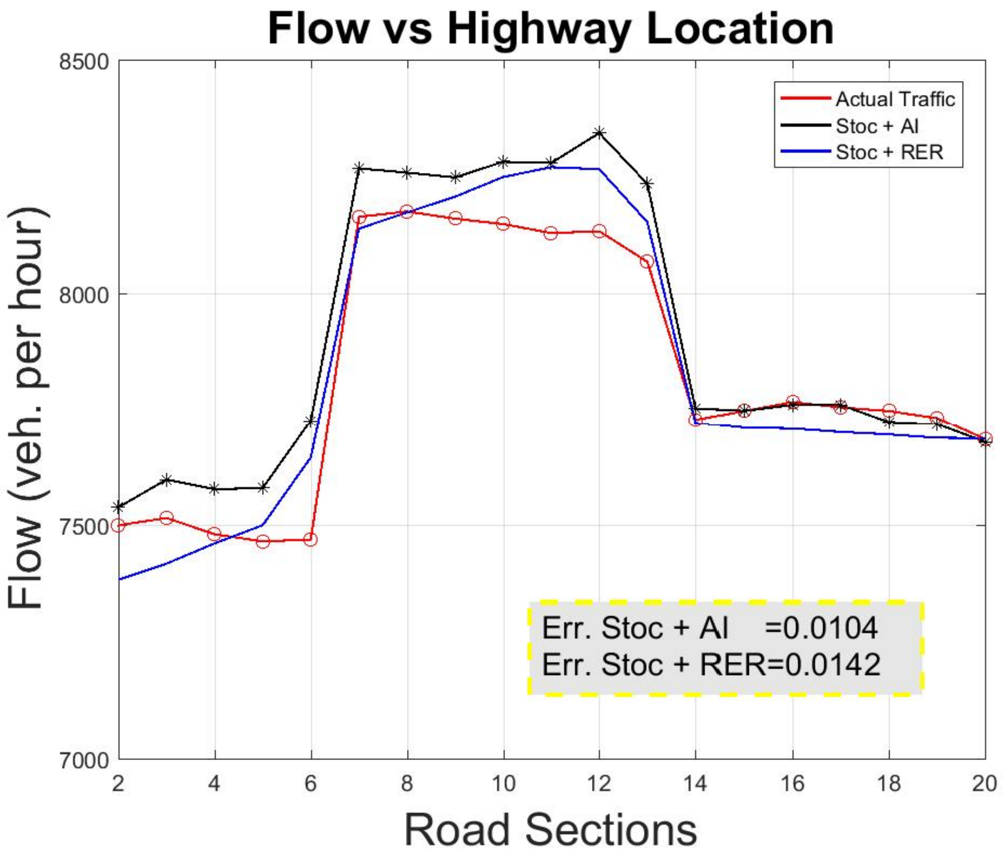
US 101

Actual data vs stochastic



US 101

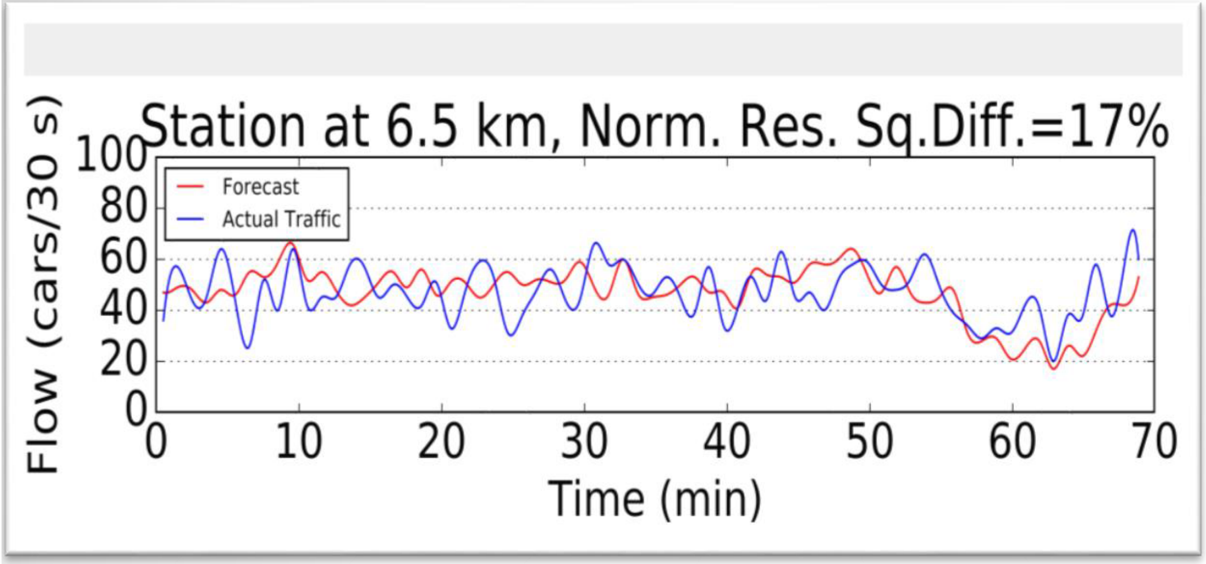
Actual data vs stochastic + RER

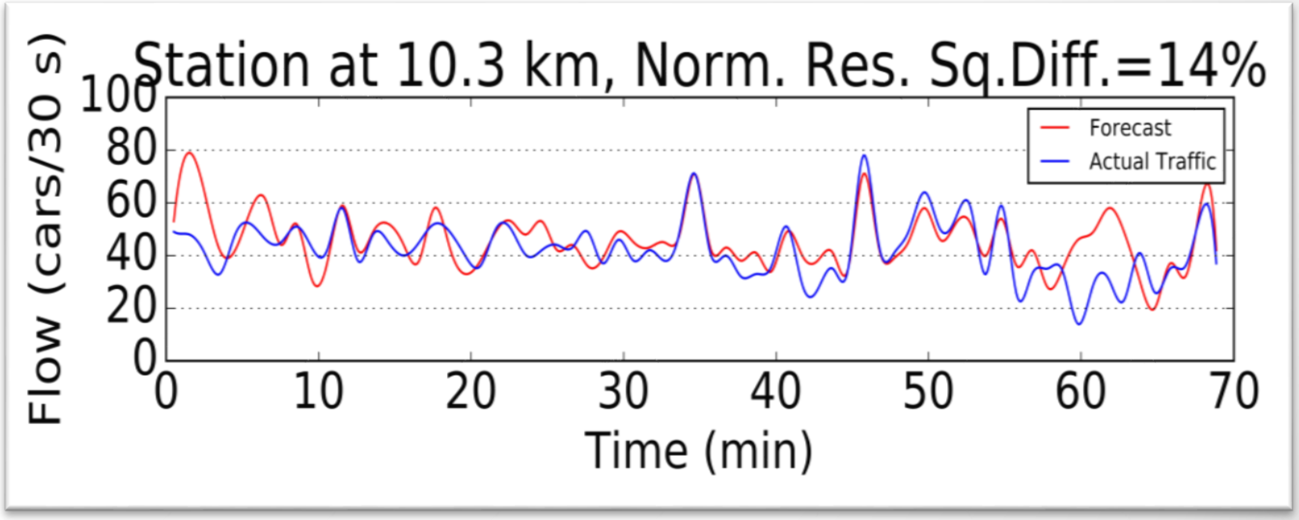


US 101

Actual data vs stochastic + RER vs stochastic + AI

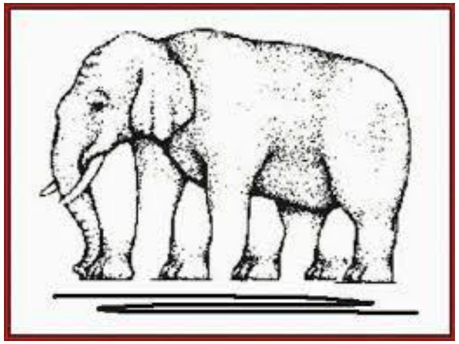
US-880


Station at 6.5 km, Norm. Res. Sq.Diff.=17%
Forecast
Actual Traffic
Flow (cars/30 s)
100
80
60
40
20
0
0
10
20
30
40
50
60
70
Time (min)


Station at 10.3 km, Norm. Res. Sq.Diff.=14%
Forecast
Actual Traffic
Flow (cars/30 s)
100
80
60
40
20
0
0
10
20
30
40
50
60
70
Time (min)


Thank you!

Thank you!



Related Publications

•G. Giacomin, J.L. Lebowitz and E. Presutti, Deterministic and Stochastic Hydrodynamic Equations Arising from Simple Microscopic Model Systems
•Katsoulakis, Majda, Sopasakis, *Nonlinearity*, (2006).
•Katsoulakis, Majda, Sopasakis, *AMS Contemporary Math*., (2010).
•Katsoulakis, Plechac, Sopasakis, *SIAM J. Numerical Analysis*, (2006).
•C. Kipnis and C. Landim, Scaling Limits of Interacting Particle Systems, Springer, 1999.
•N.V.Krylov, Introduction to the Theory of Random Processes, Graduate Studies in Mathematics, Vol 43, AMS, Providence, Rhode Island, 2002.
•Khouider, Majda, PNAS, (2003).
•I. Palasti, On some random space filling problems, Publ. Math. Inst. Hungar. Acad. Sci., 5:353-360, 1960.
•A. Renyi, On a one-dimensional problems concerning random space-filling, Publ. Math. Inst. Hungar. Acad. Sci., 3:109-127, 1958
•Sopasakis, *Physica A*, (2003).
•Sopasakis and Katsoulakis, *SIAM J. Applied Math*. (2005).
•T. M. Liggett, Continuous Time Markov Processes: An Introduction, Graduate Studies in Mathematics, Vol 113, AMS, Providence, Rhode Island, 2010.

Thank you!