Minimizing the Cost of Iterative Compilation with Active Learning

William Ogilvie, Pavlos Petoumenos, Zheng Wang, Hugh Leather

THE UNIVERSITY of EDINBURGH
informatics

William

wfogilvie@gmail.com

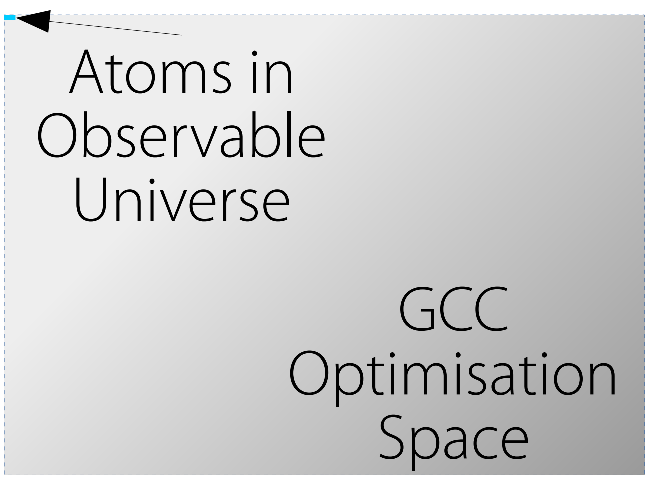
William

wfogilvie@gmail.com

Finishing his PhD soon :)

Performance tuning is hard!

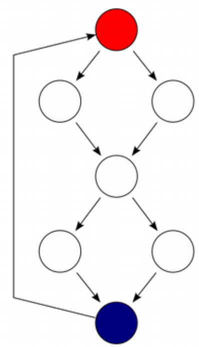
Performance tuning is hard!

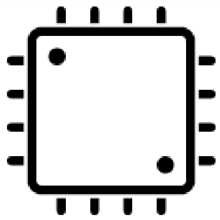


Performance tuning is hard!

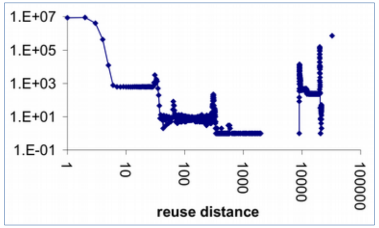
Atoms in Observable Universe

GCC Optimisation Space





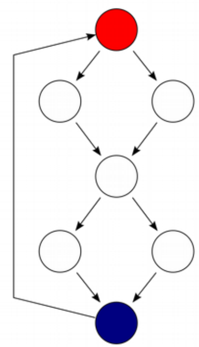
Runtime: …
Memory Usage: …
DL1 misses: …
L2 misses:…
…
…

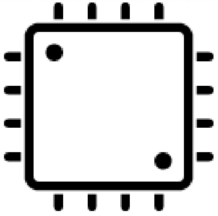


Performance tuning is hard!

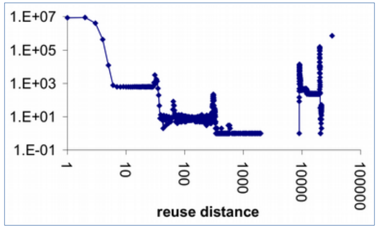
Atoms in Observable Universe

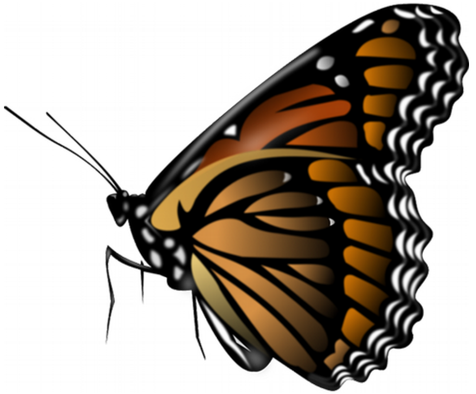
GCC Optimisation Space





Runtime: …
Memory Usage: …
DL1 misses: …
L2 misses:…
…
…





Performance tuning is hard!

ML to the rescue

Heterogeneous mapping

Heterogeneous mapping

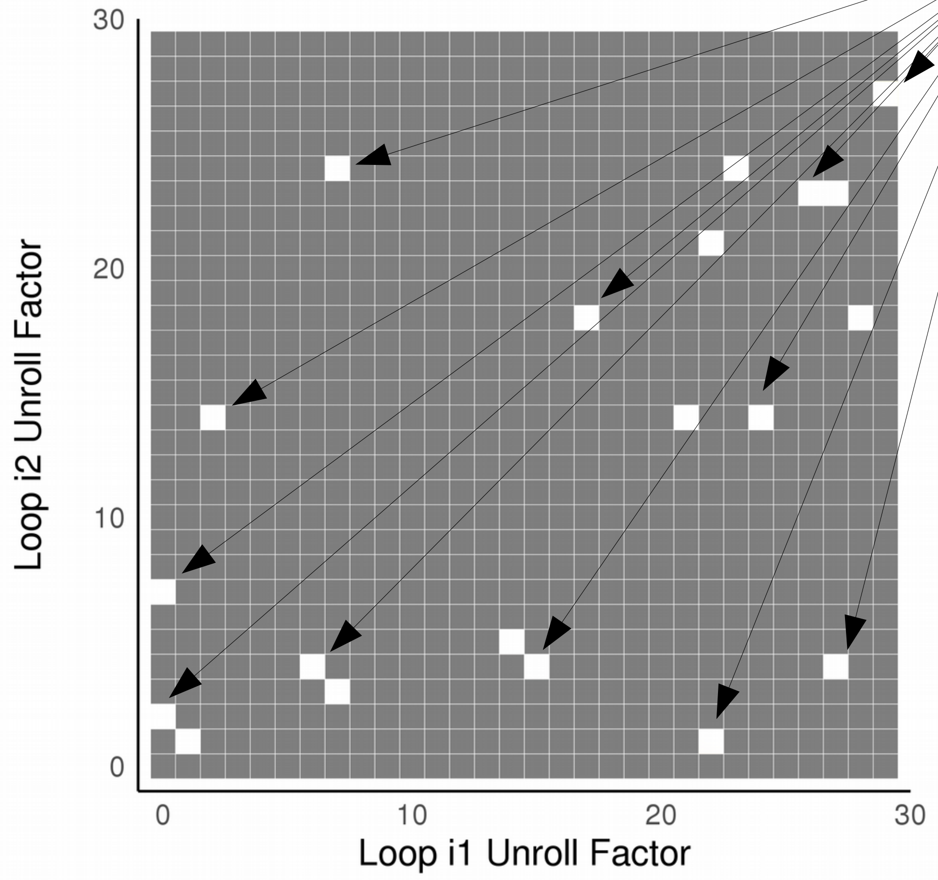
Workgroup size autotuning

Heterogeneous mapping

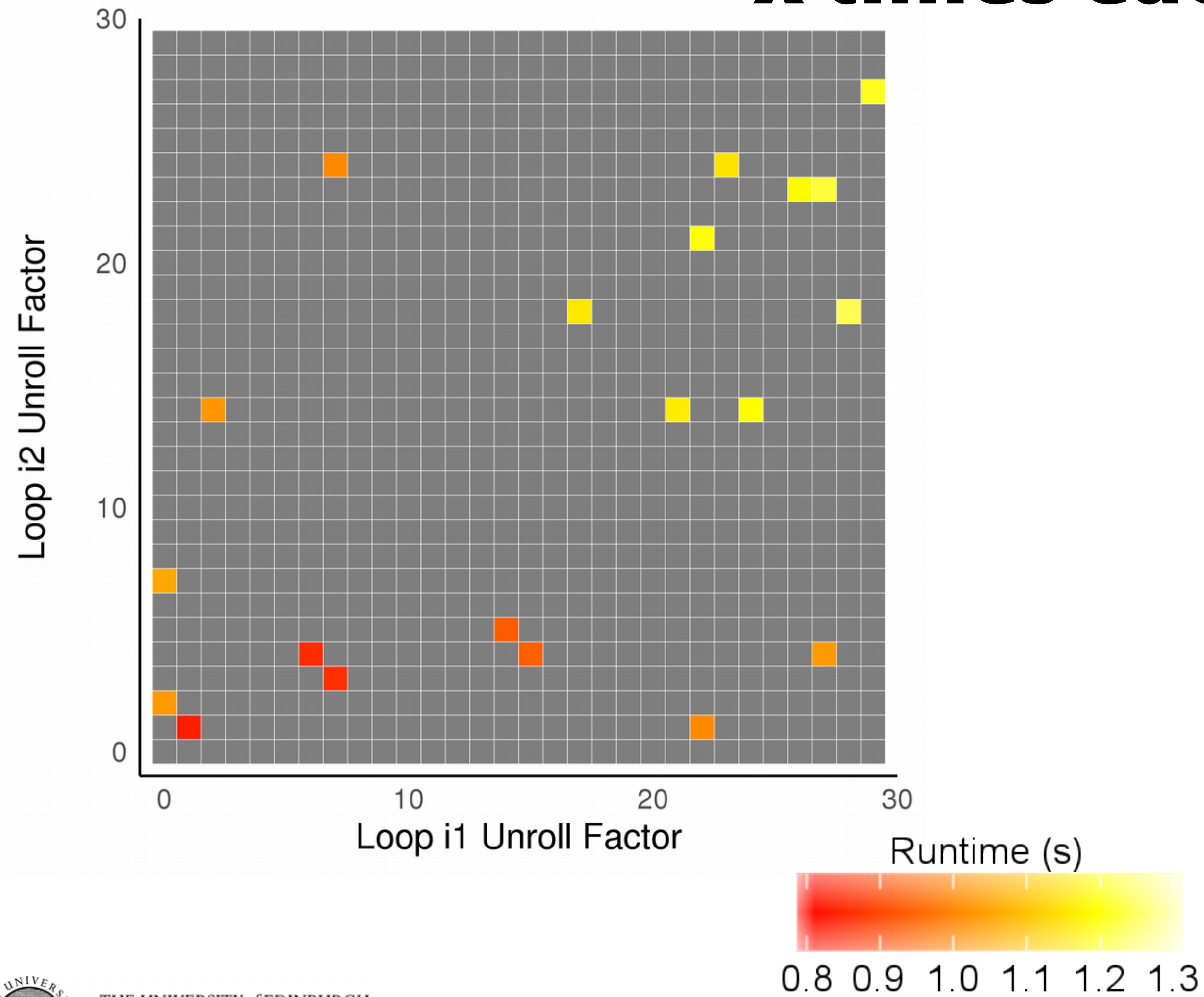
Workgroup size autotuning

Compiler optimizations

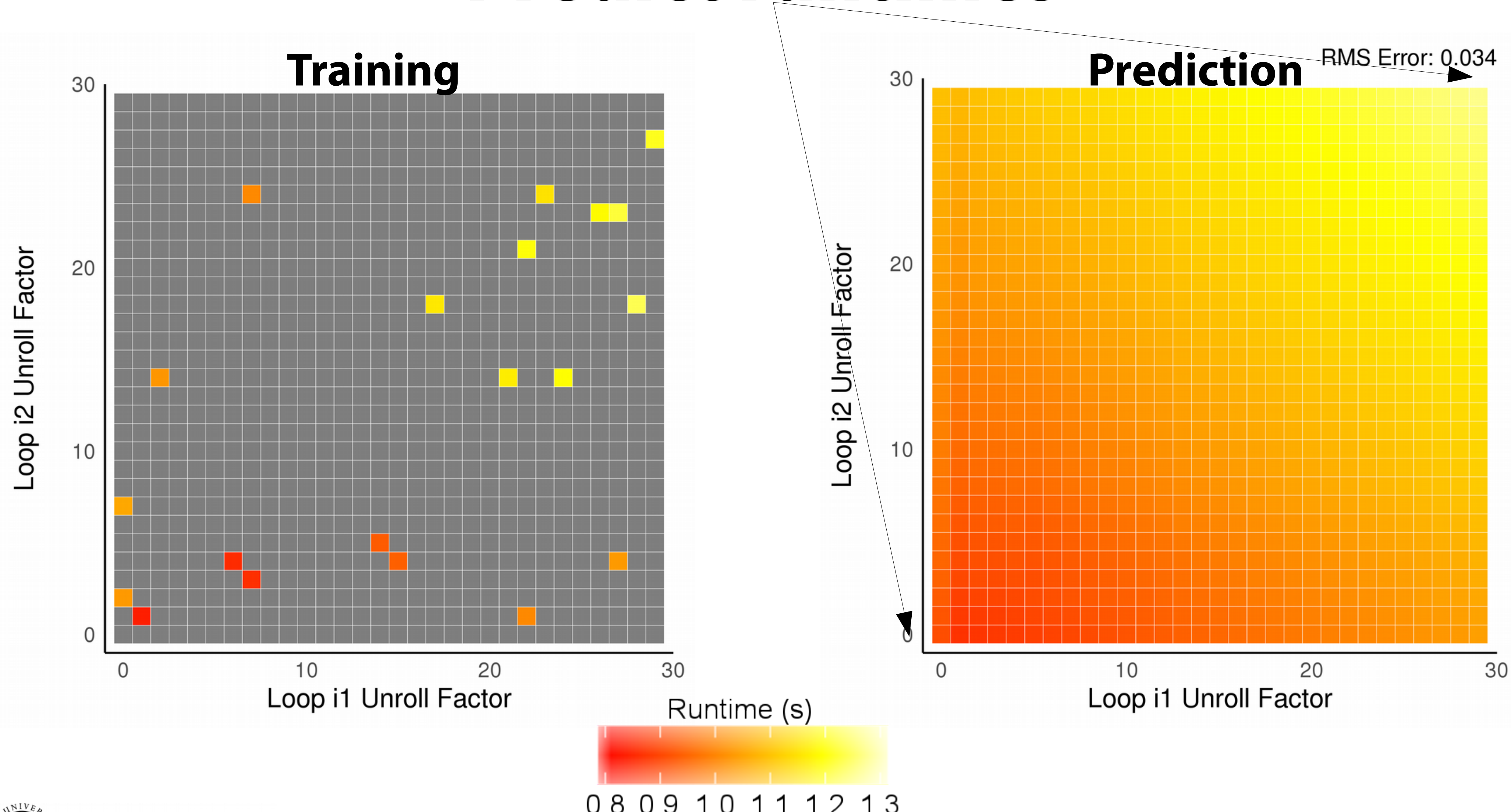
Select Points



Profile
x times each



Predict runtimes



Too slow!

1Ks to1Ms of points
10s of observations for each

Faster?

Faster?

	Machine Learning		
Training Points	Random, preselected		
Profiling runs	Fixed, preselected		

THE UNIVERSITY of EDINBURGH
informatics

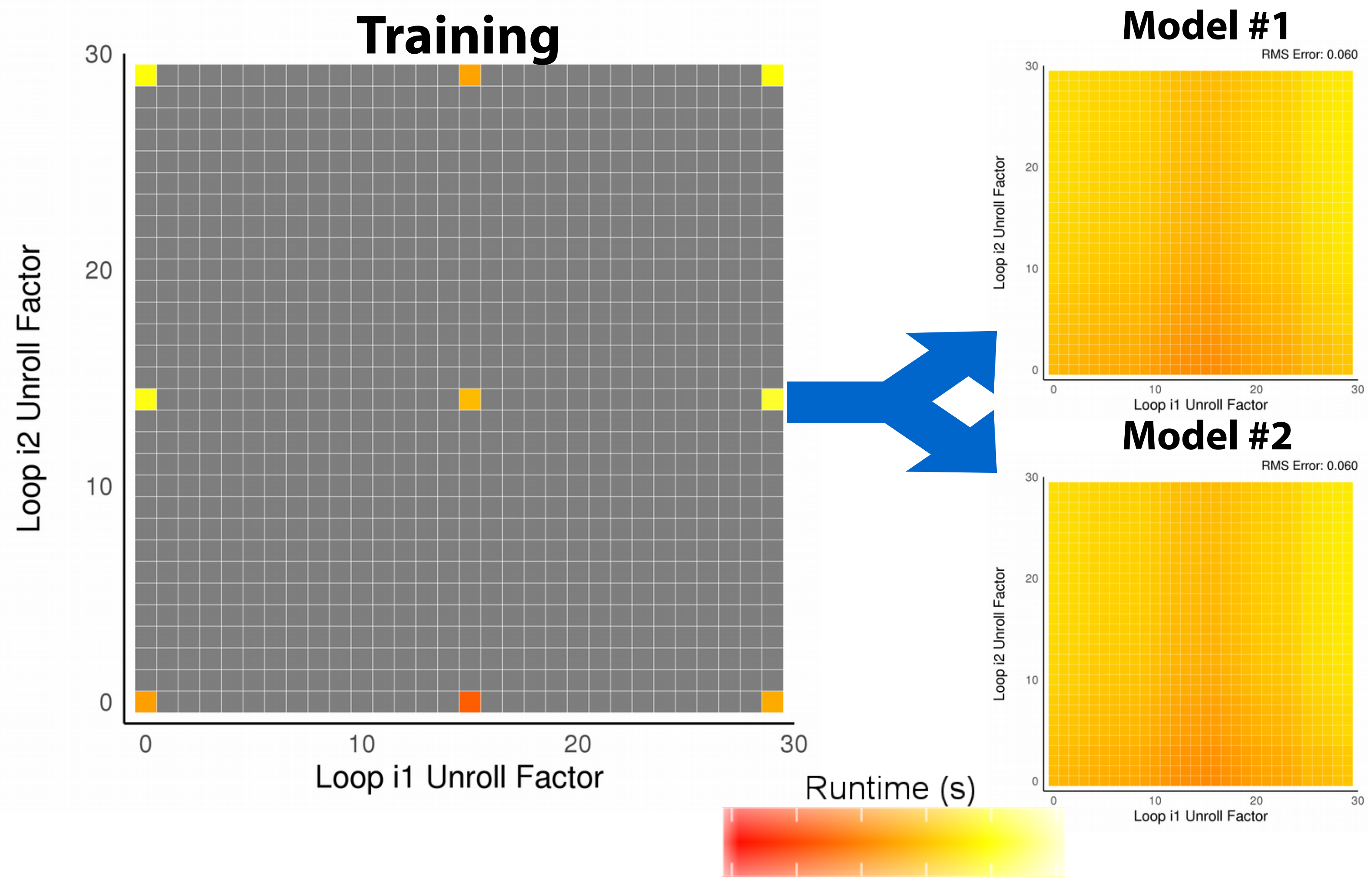
Faster?

	Machine Learning	**?**	
Training Points	Random, preselected	**High info points**	
Profiling runs	Fixed, preselected	Fixed, preselected	

Faster?

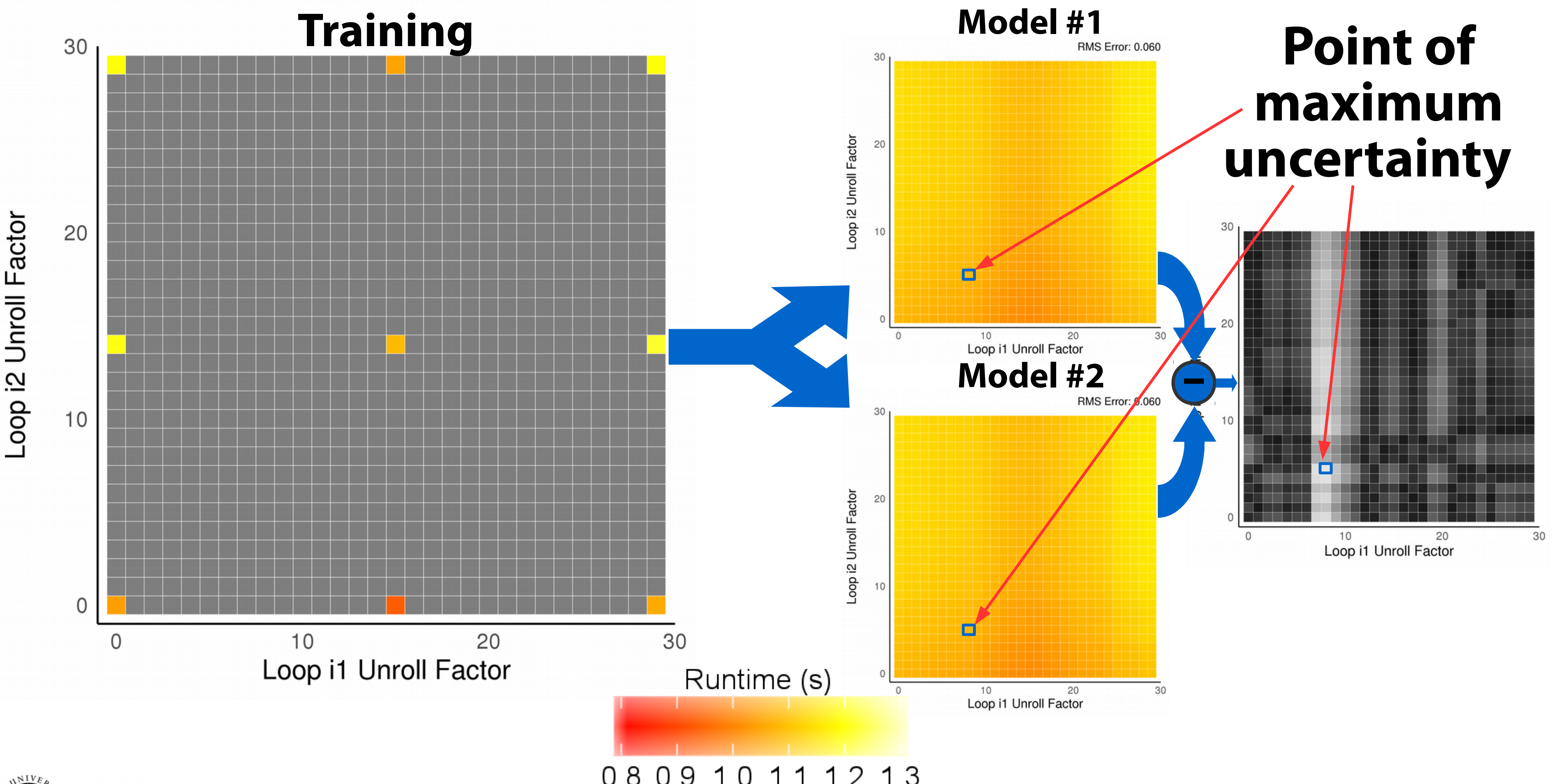
	Passive Learning	**Active Learning**	
Training Points	Random, preselected	**ML-guided selection**	
Profiling runs	Fixed, preselected	Fixed, preselected	

Multiple models

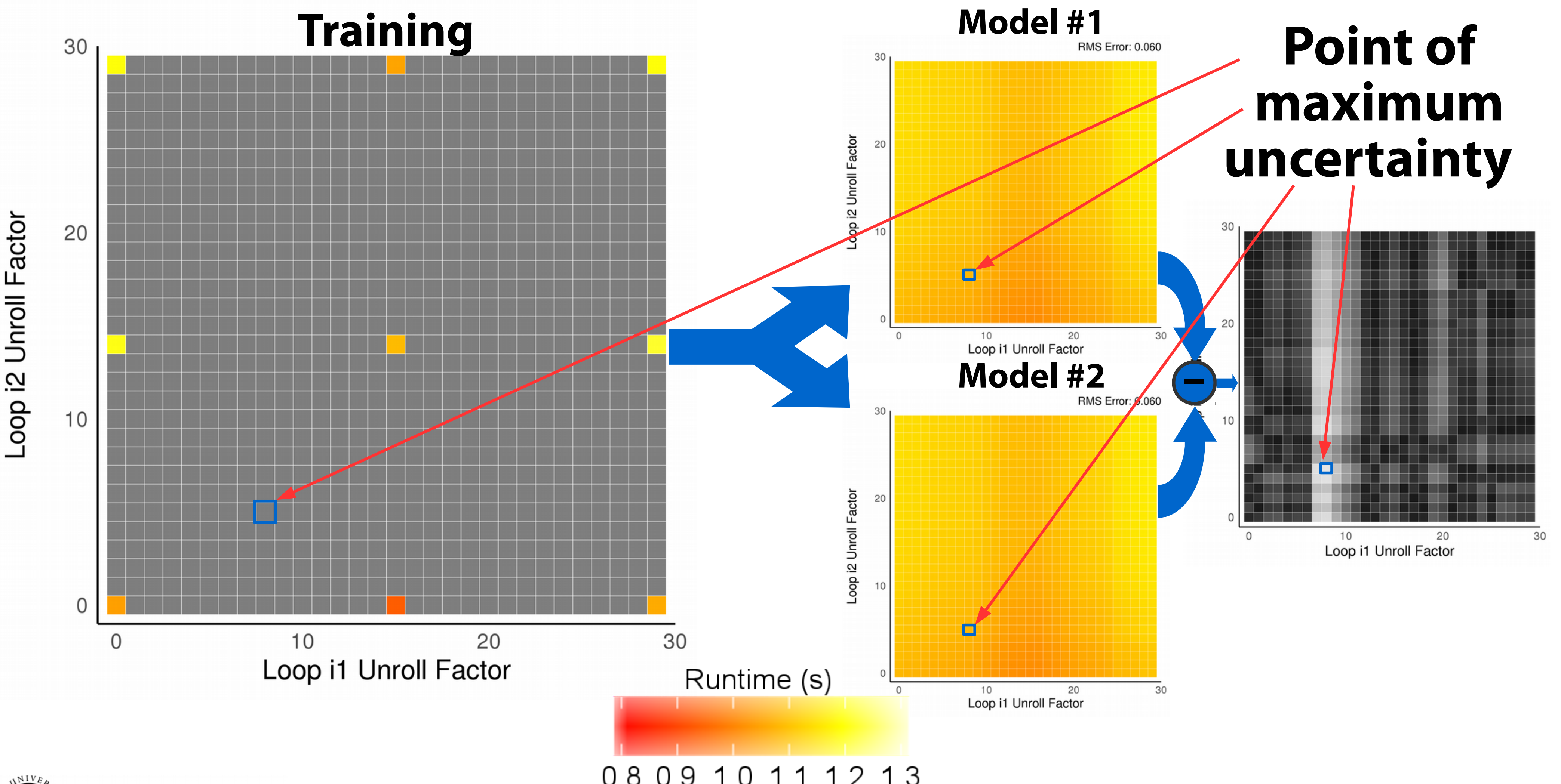


THE UNIVERSITY *of* EDINBURGH
informatics

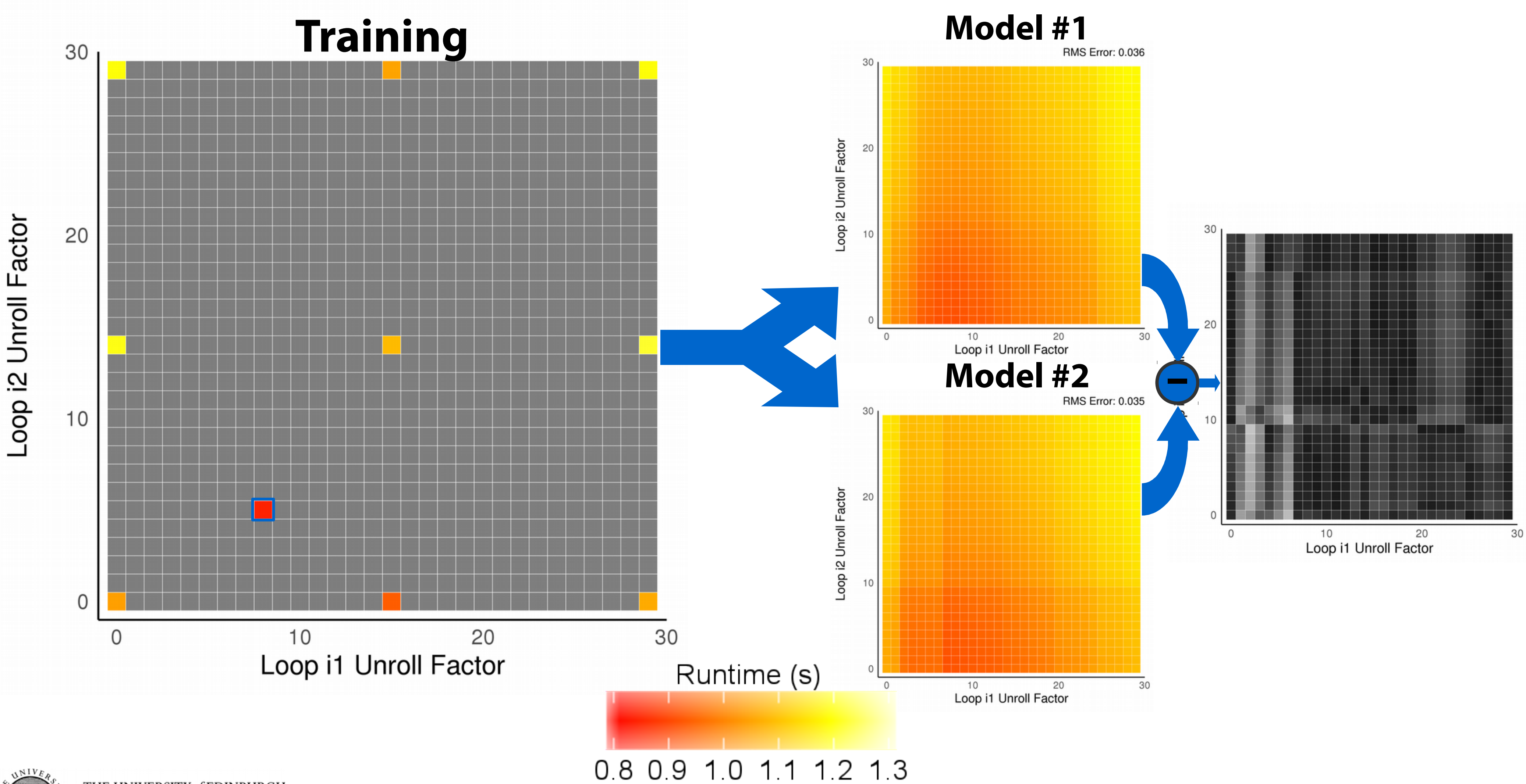
Find high uncertainty points



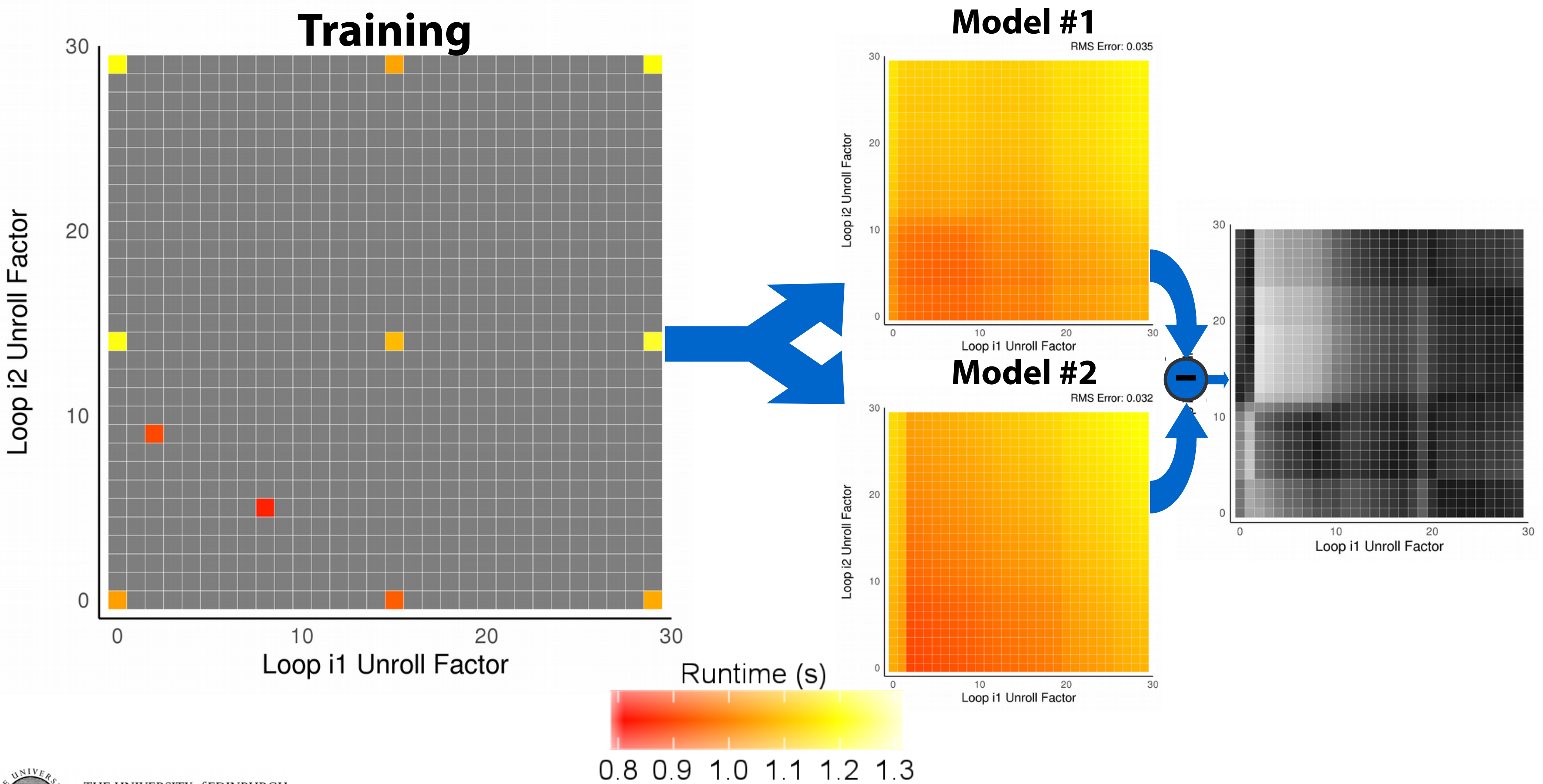
Select them


Training
Loop i2 Unroll Factor
Loop i1 Unroll Factor
0
10
20
30
Runtime (s)
0.8 0.9 1.0 1.1 1.2 1.3
Model #1
RMS Error: 0.060
Model #2
RMS Error: 0.060
Point of maximum uncertainty

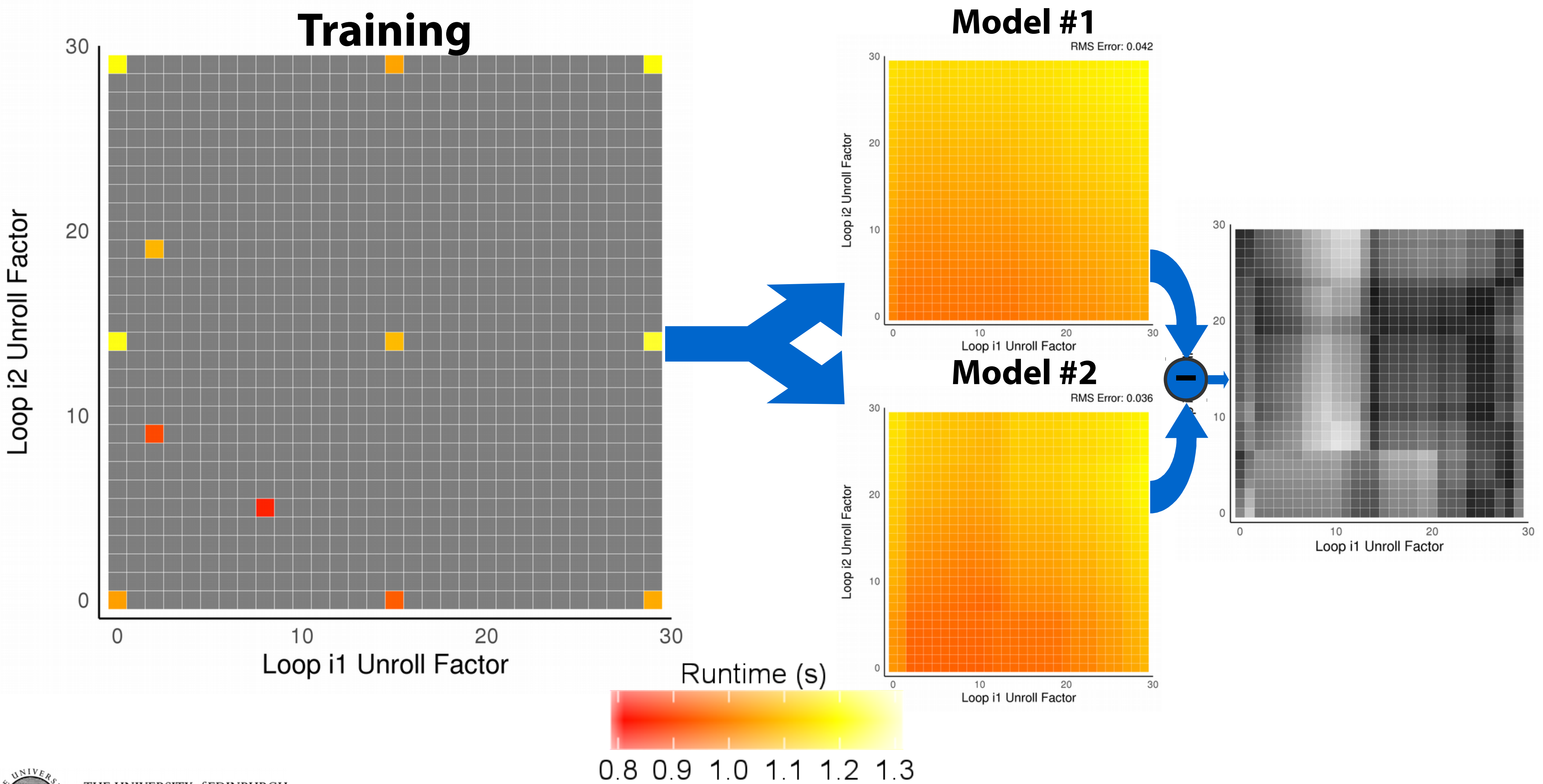
Profile them



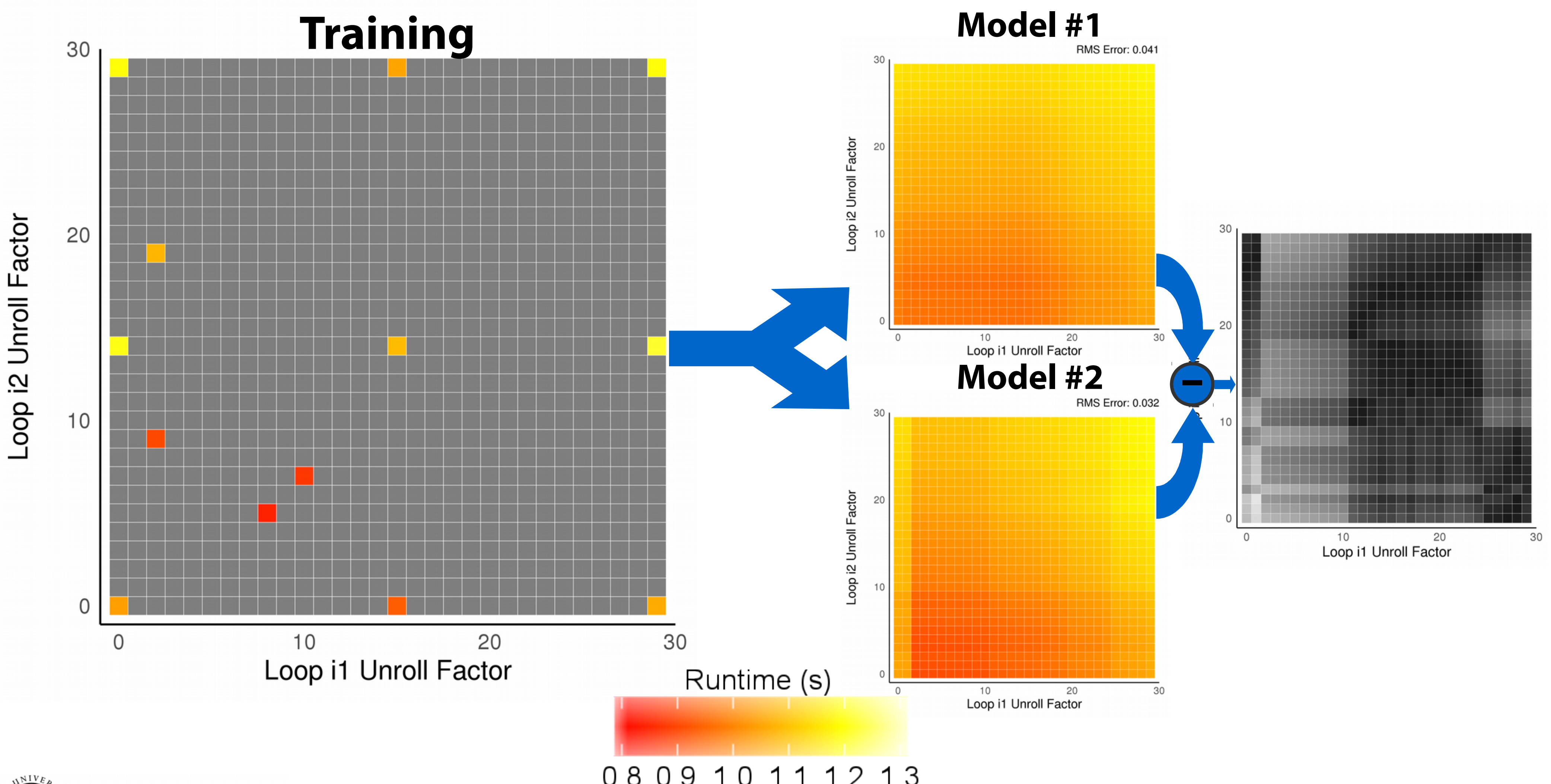
Repeat


Training
30
20
10
0
Loop i2 Unroll Factor
0
10
20
30
Loop i1 Unroll Factor
Runtime (s)
0.8 0.9 1.0 1.1 1.2 1.3
Model #1
RMS Error: 0.035
Loop i2 Unroll Factor
Loop i1 Unroll Factor
Model #2
RMS Error: 0.032
Loop i2 Unroll Factor
Loop i1 Unroll Factor
Loop i1 Unroll Factor

Repeat


Training
30
20
10
0
Loop i2 Unroll Factor
0
10
20
30
Loop i1 Unroll Factor
Model #1
RMS Error: 0.042
Model #2
RMS Error: 0.036
Runtime (s)
0.8 0.9 1.0 1.1 1.2 1.3

Repeat

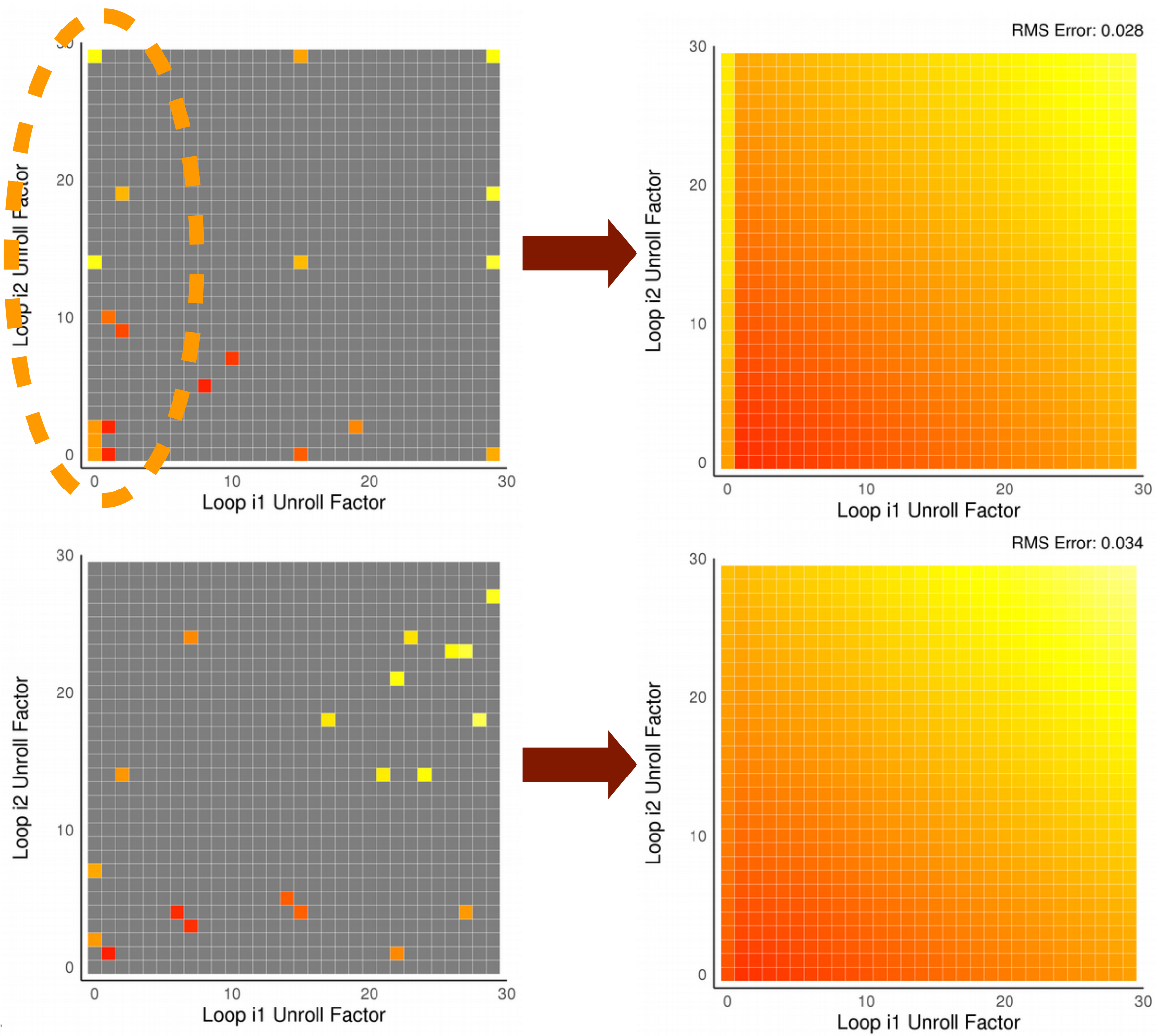


Training
30
20
10
0
0
10
20
30
Loop i2 Unroll Factor
Loop i1 Unroll Factor
Model #1
RMS Error: 0.041
Model #2
RMS Error: 0.032
Runtime (s)
0.8 0.9 1.0 1.1 1.2 1.3

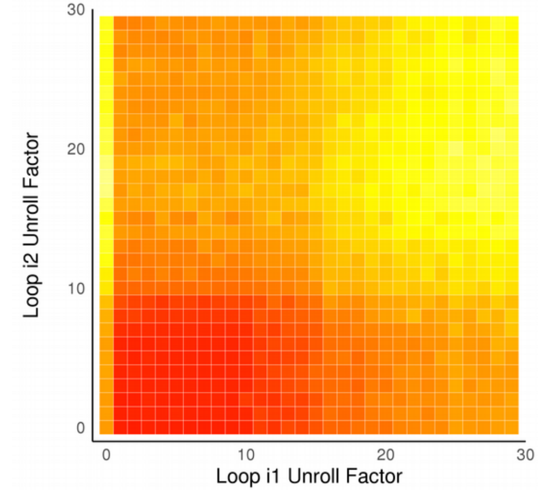
Training | Prediction

Active

Passive



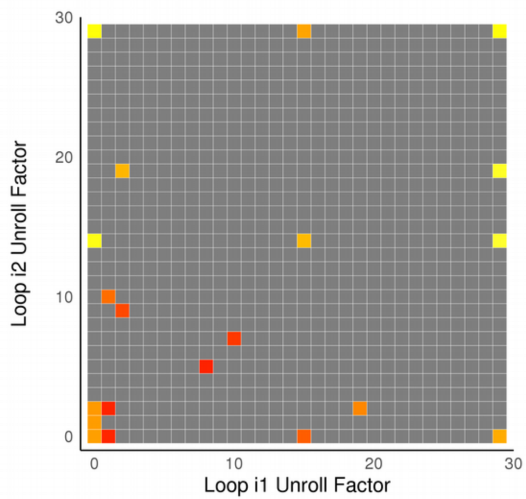
Runtime Measurements

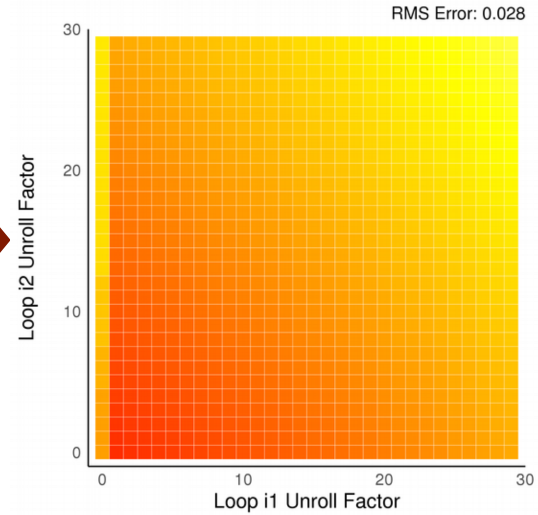


Training

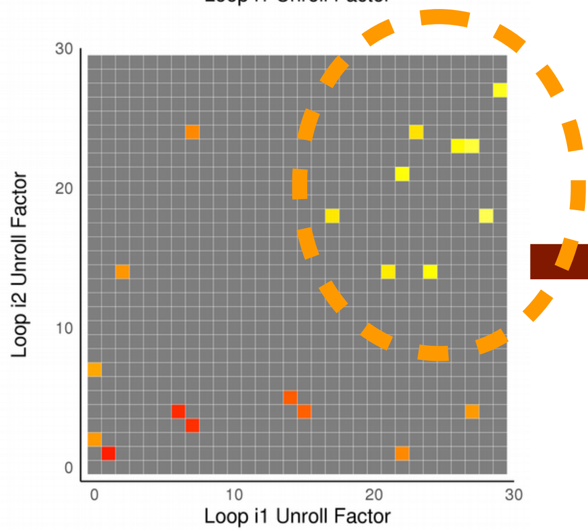
Prediction

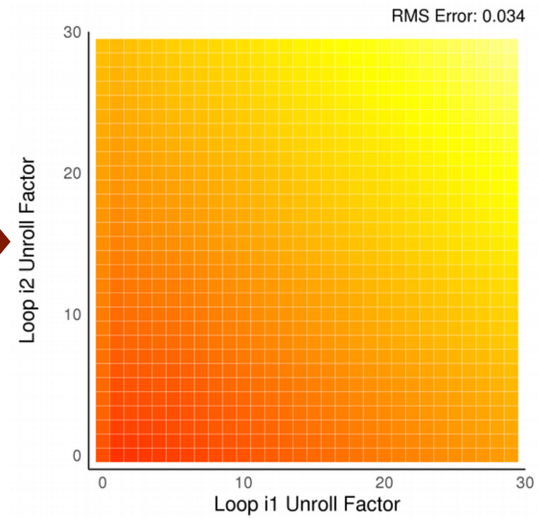
Active



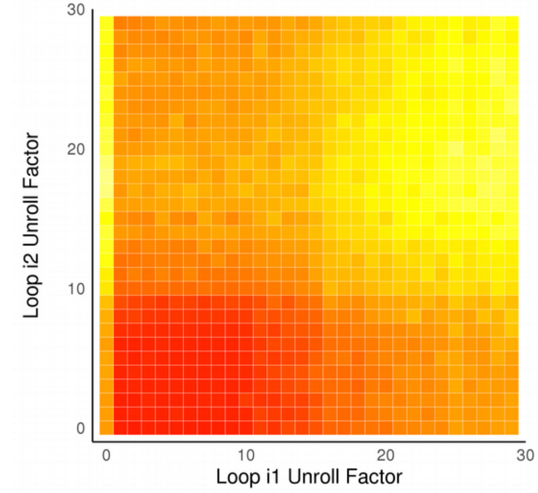


Passive





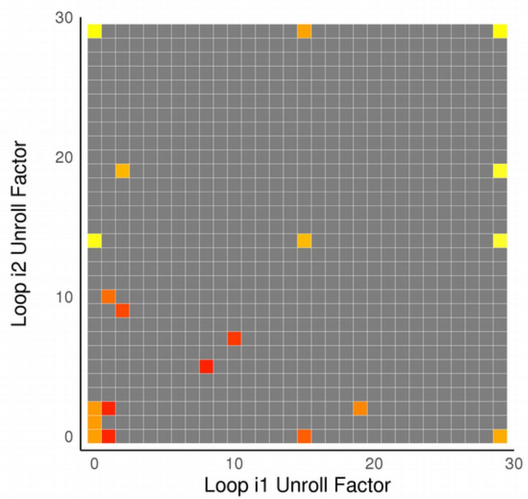
Runtime Measurements

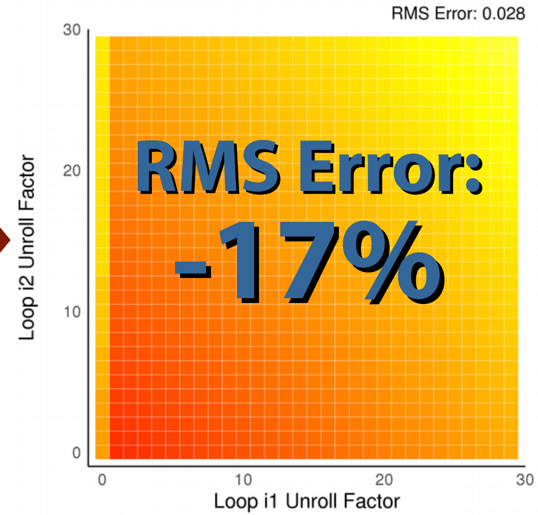


Training

Prediction

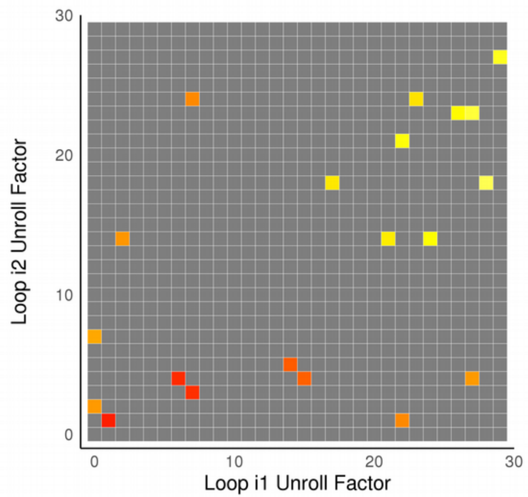
Active

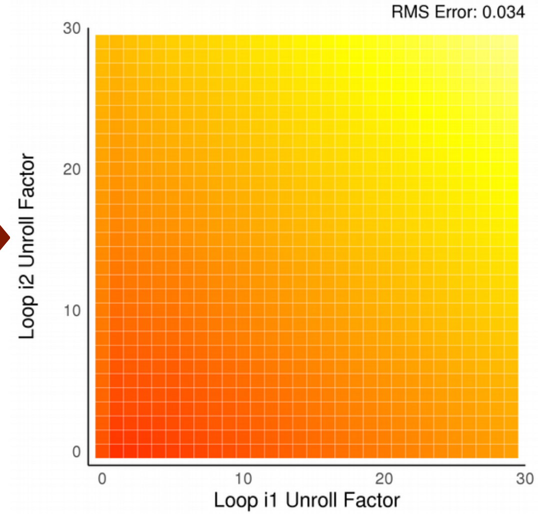




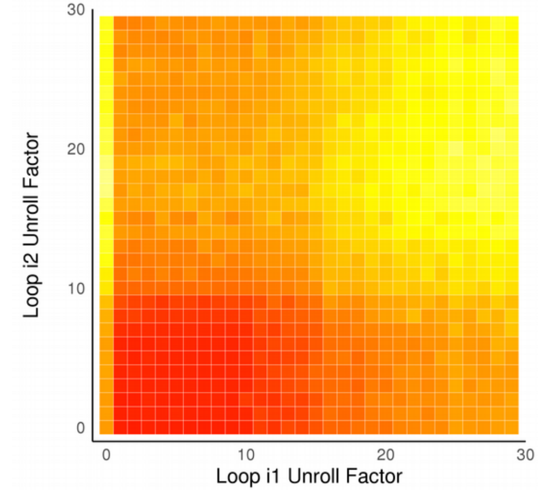
RMS Error: -17%

Passive





Runtime Measurements



3x faster

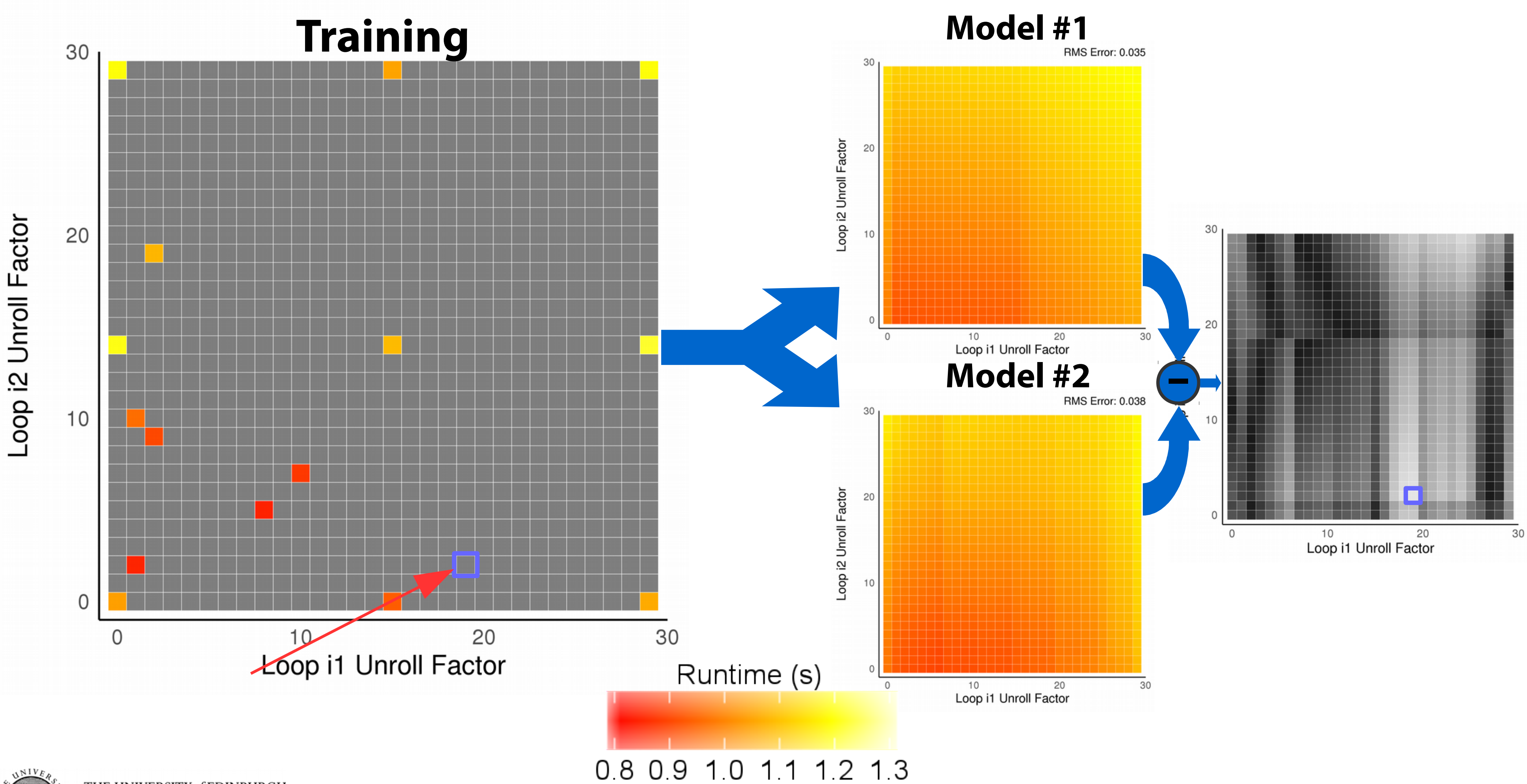
	Passive Learning	Active Learning	
Training Points	Random, preselected	ML-guided selection	
Profiling runs	Fixed, preselected	Fixed, preselected	

3x faster

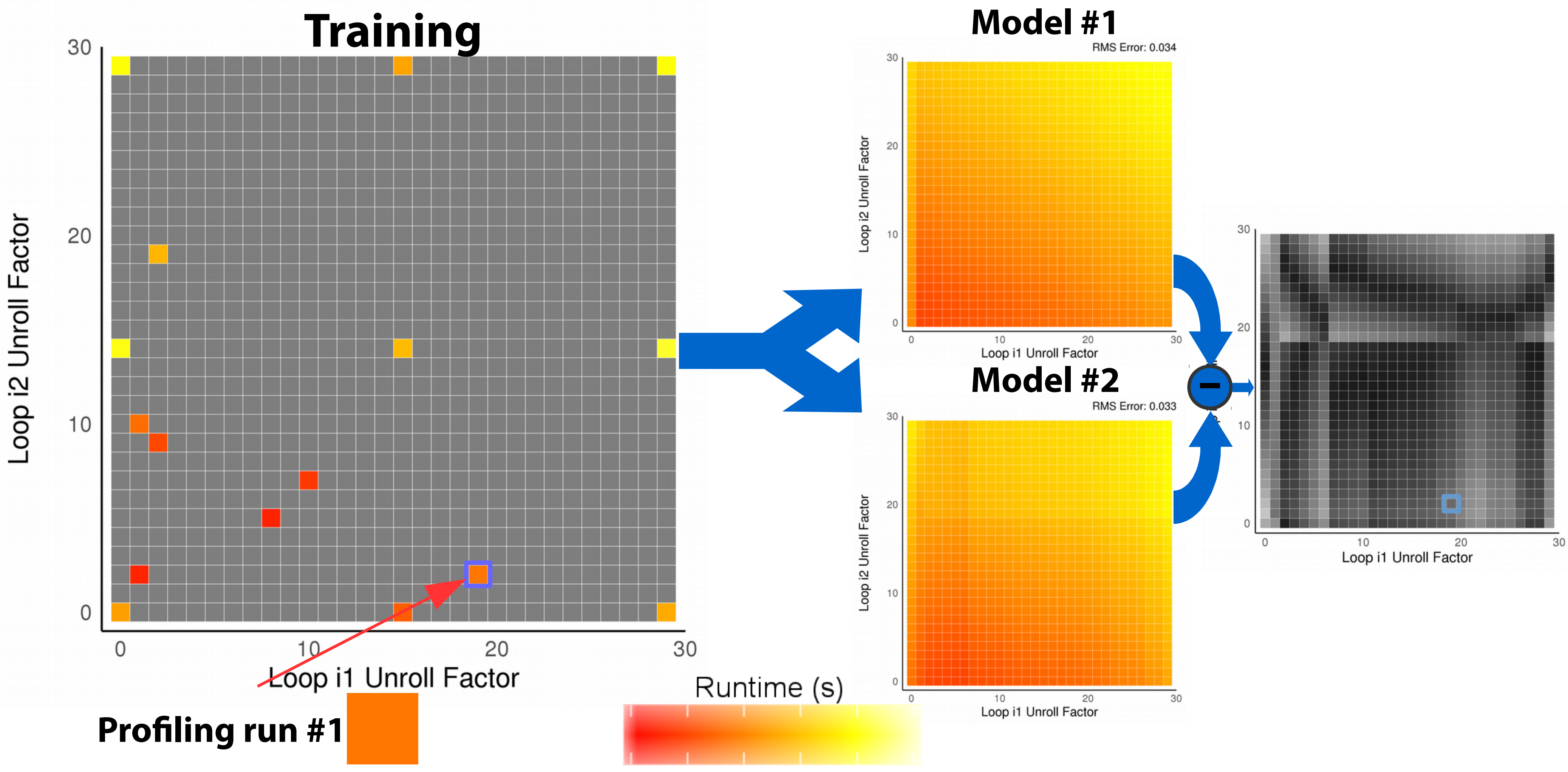
	Passive Learning	Active Learning	**Something Better?**
Training Points	Random, preselected	ML-guided selection	**?**
Profiling runs	Fixed, preselected	Fixed, preselected	**?**

THE UNIVERSITY of EDINBURGH
informatics

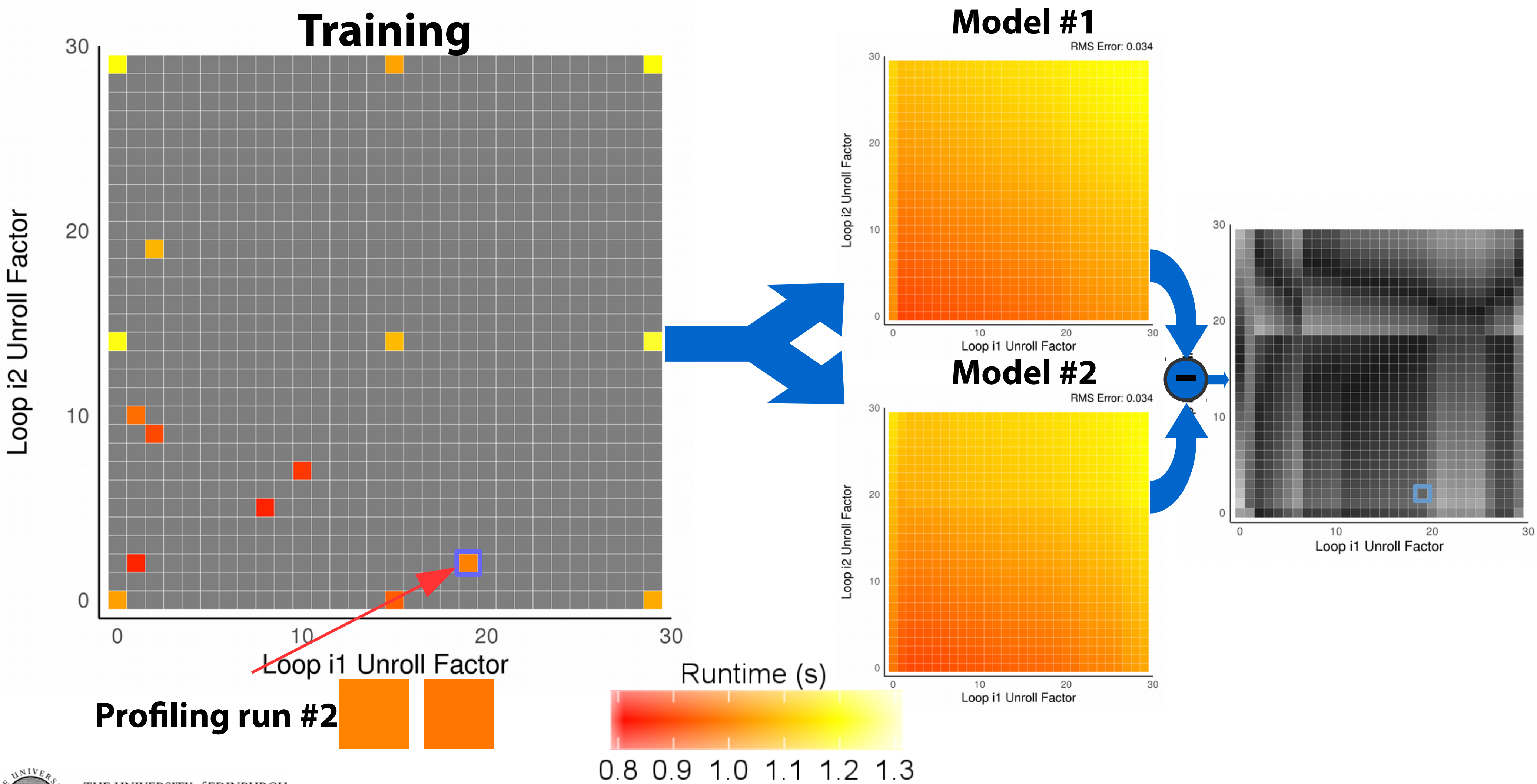
Can we do better?



Can we do better?

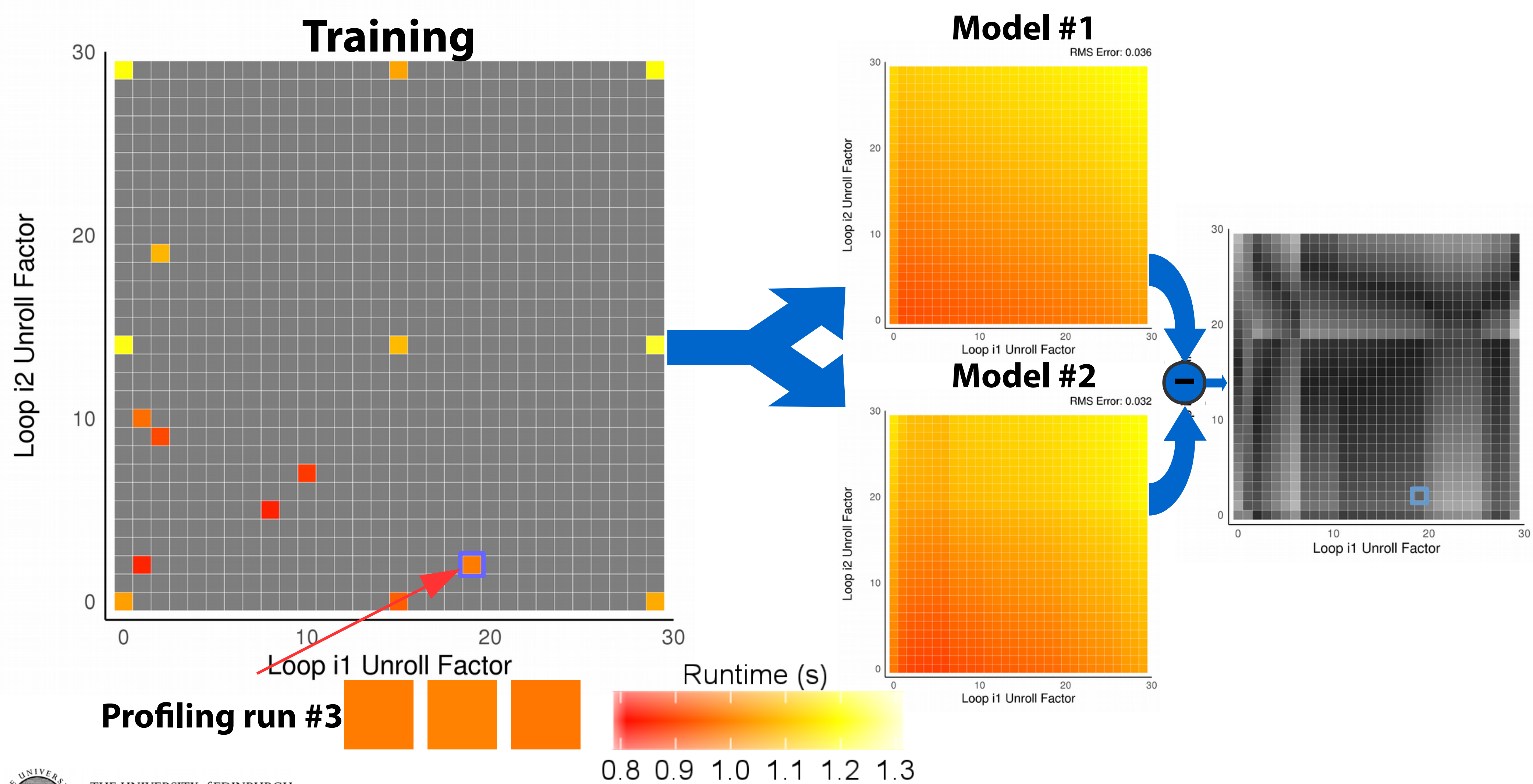


Can we do better?

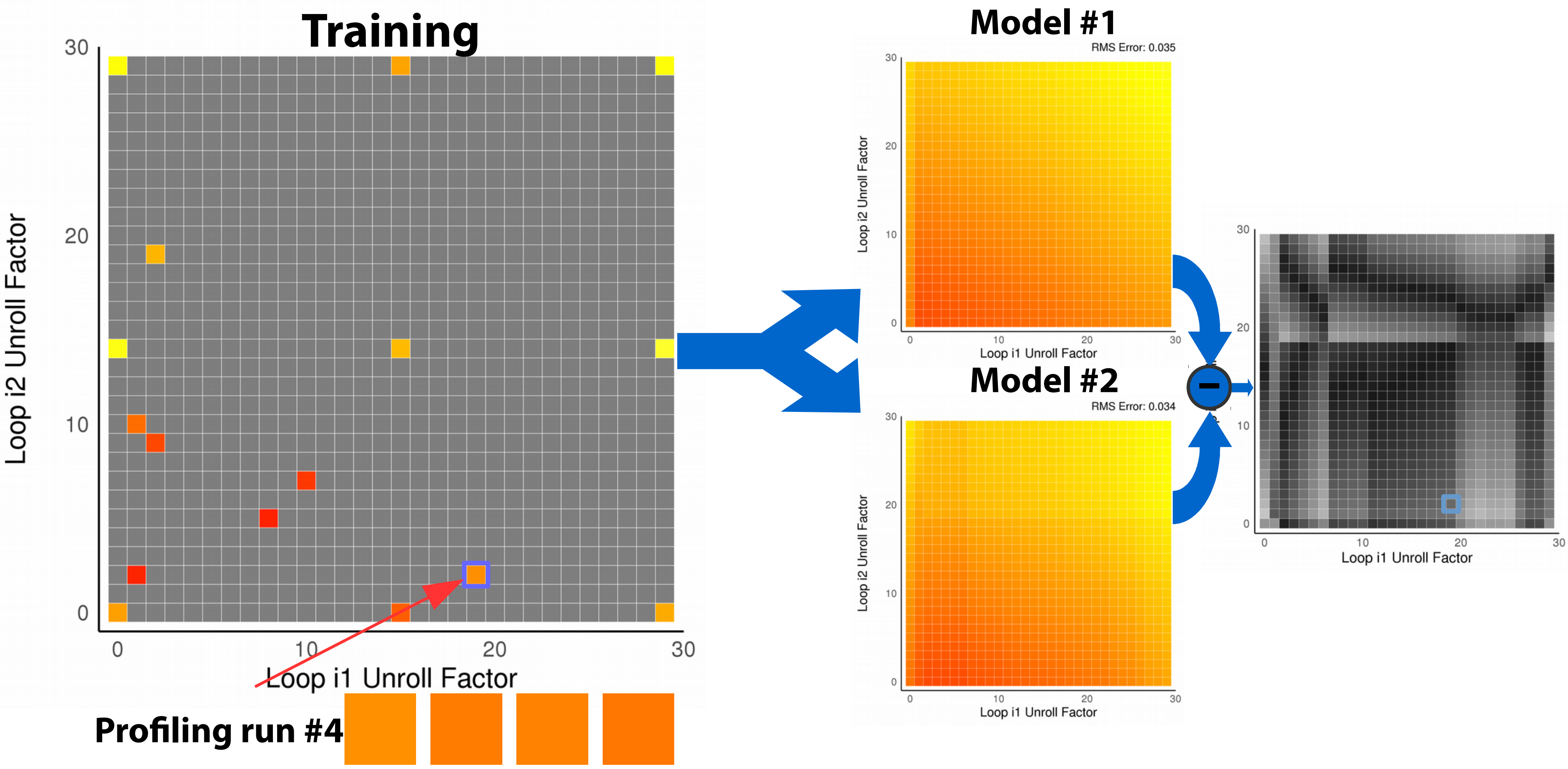


THE UNIVERSITY of EDINBURGH
informatics

Can we do better?



Can we do better?



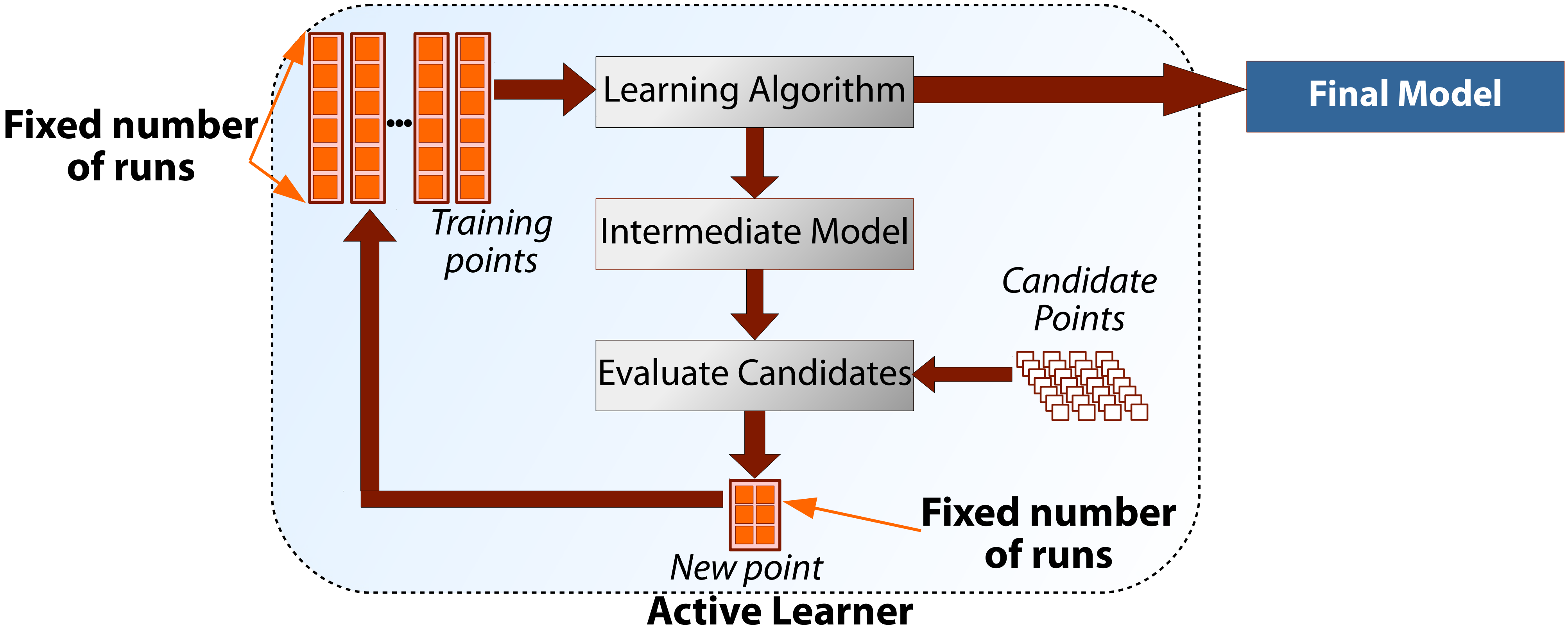
Noise

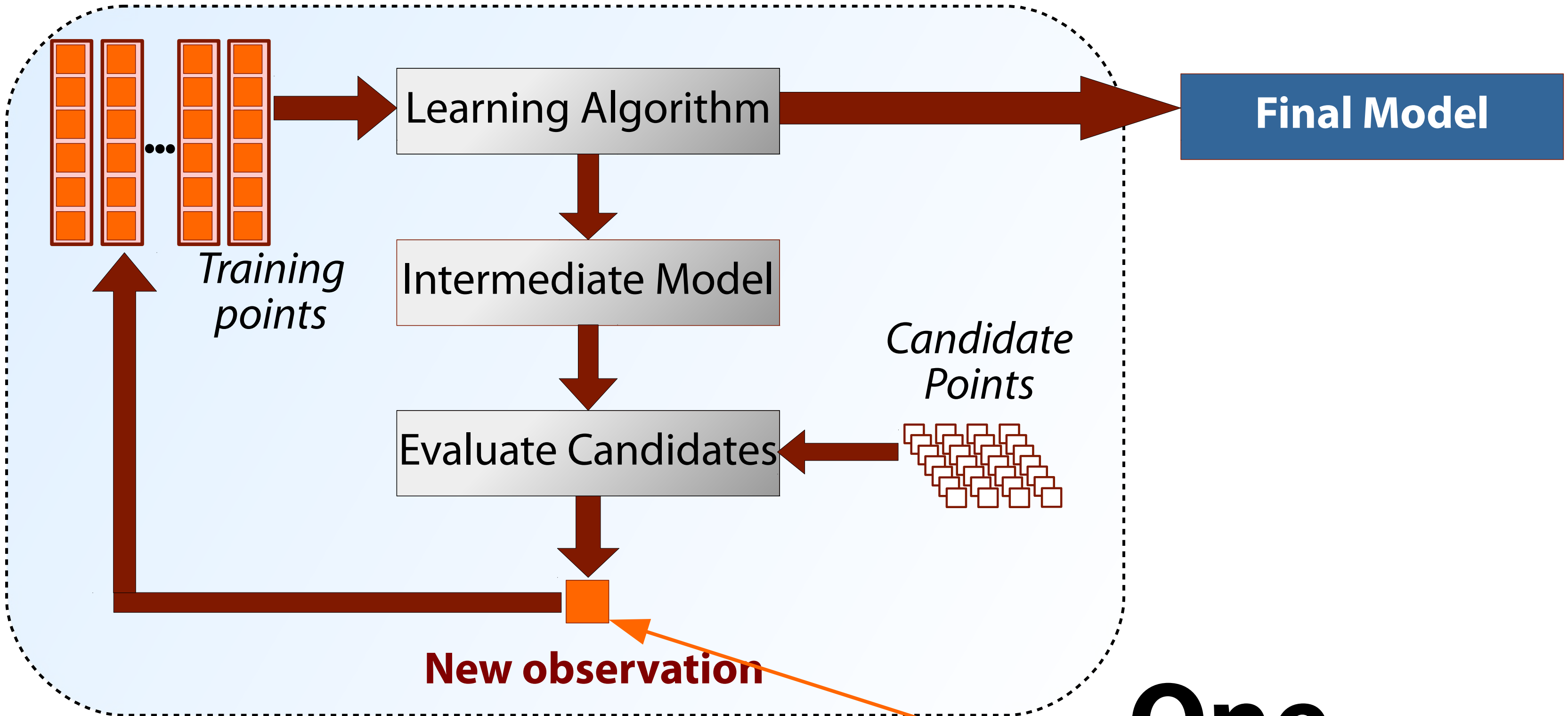
determines optimal # of observations

Noise
+ prediction uncertainty
determine optimal # of observations

	Passive Learning	Active Learning	**Our approach**
Training Points	Random, preselected	ML-guided selection	**ML-guided selection**
Profiling runs	Fixed, preselected	Fixed, preselected	**ML-guided selection**

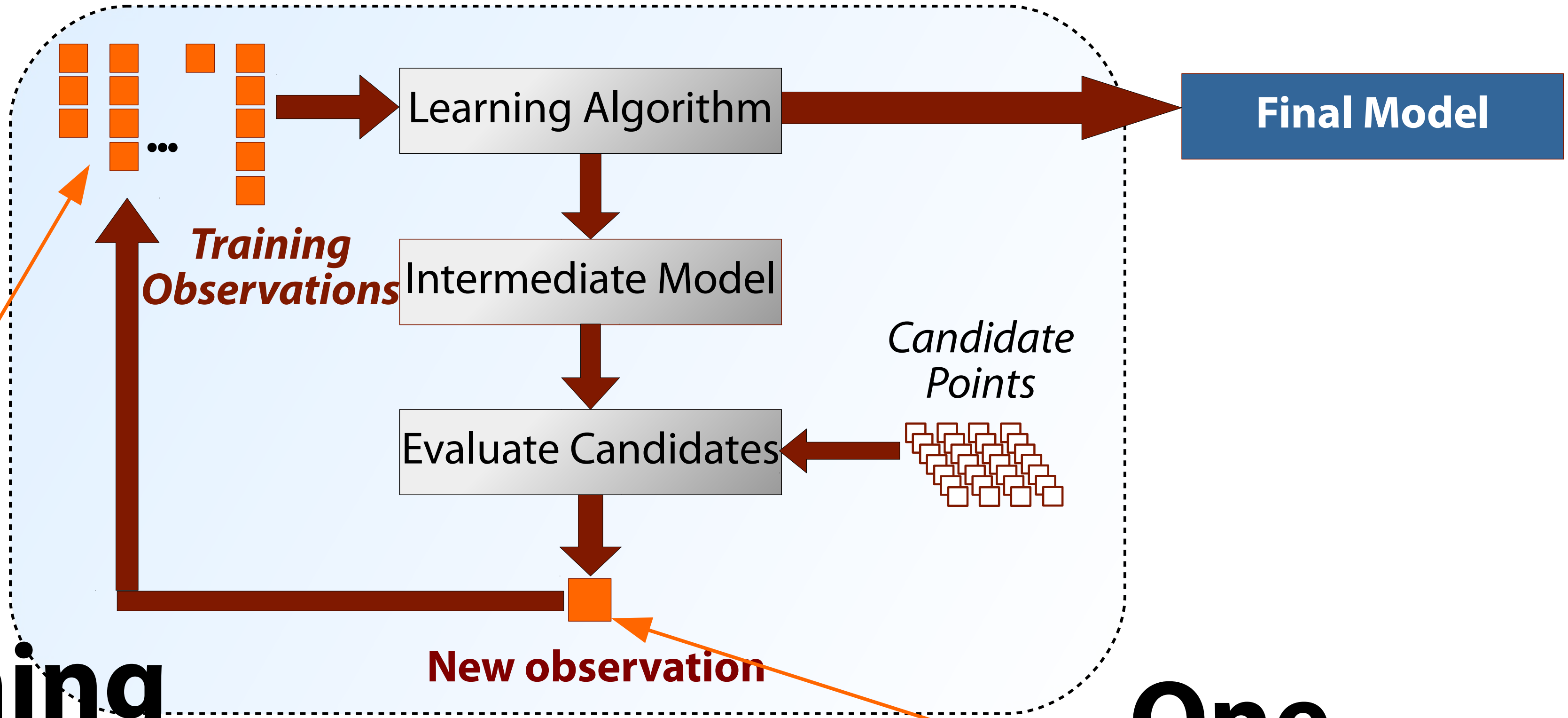
Sequential analysis with **noisy active learning**


Fixed number of runs
Learning Algorithm
Final Model
Training points
Intermediate Model
Candidate Points
Evaluate Candidates
New point
Fixed number of runs
Active Learner



Noisy Active Learner

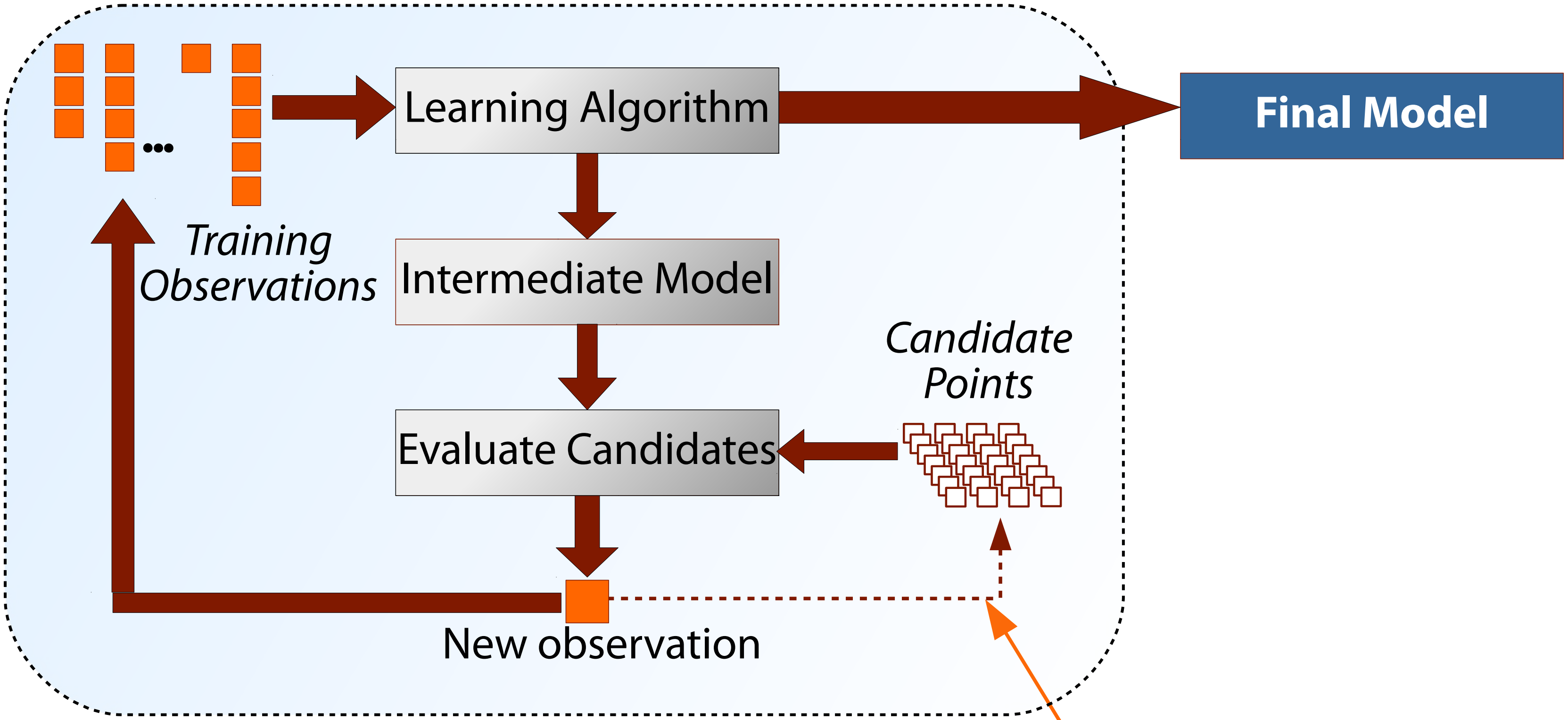
One observation at a time



Training data noisy

Noisy Active Learner

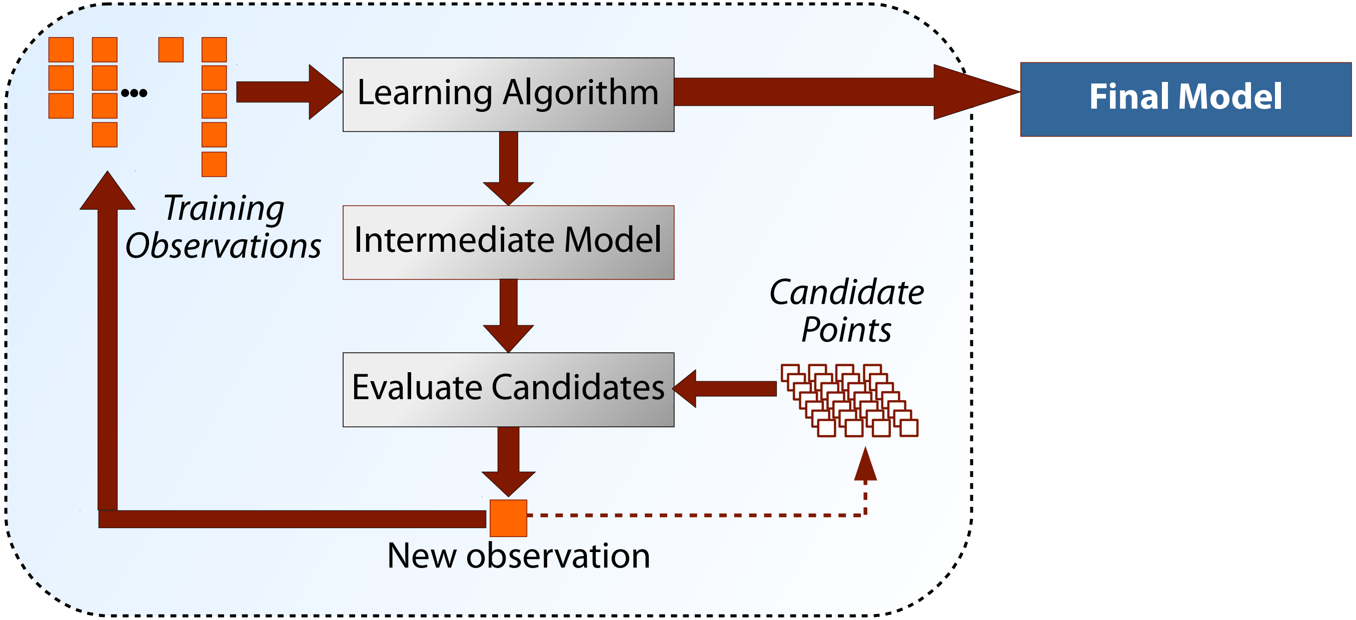
One observation at a time



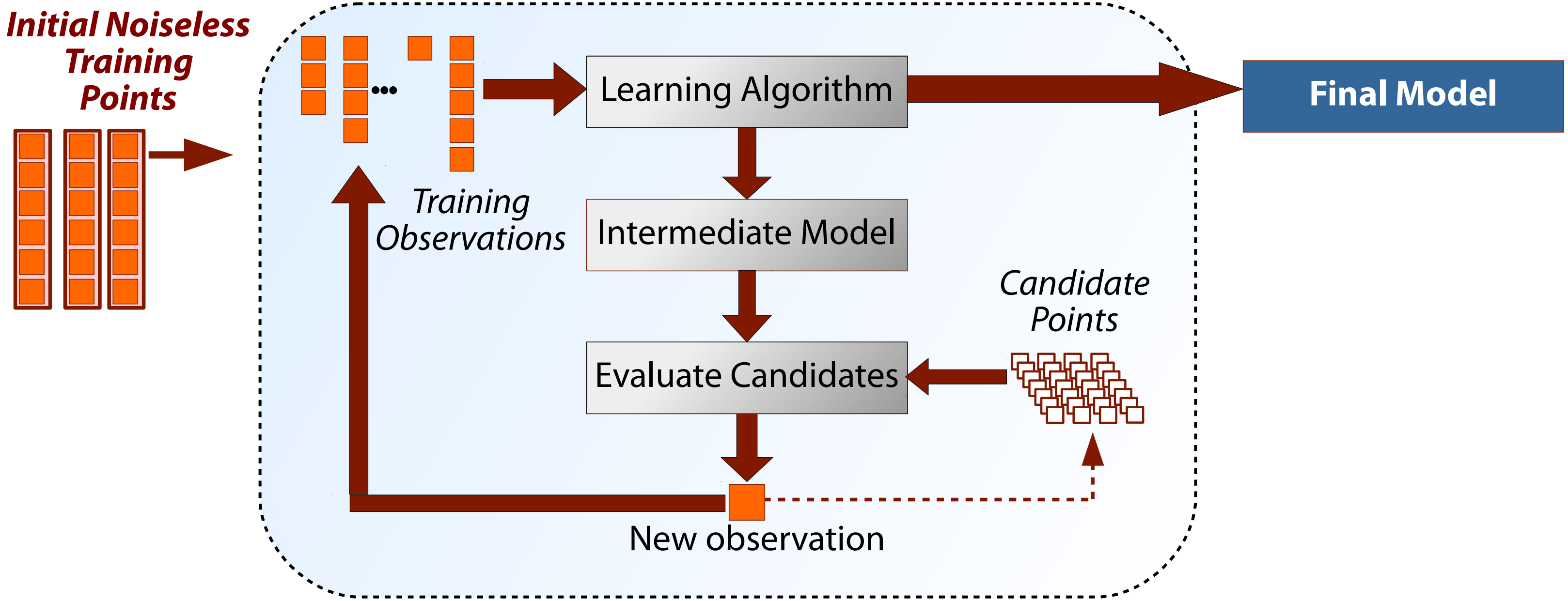
Noisy Active Learner

Selected point stays in candidate set

THE UNIVERSITY of EDINBURGH
informatics



Challenges:
1) Noisy data→ Completely wrong IM
2) Overfitting
3) Too many updates to the IM

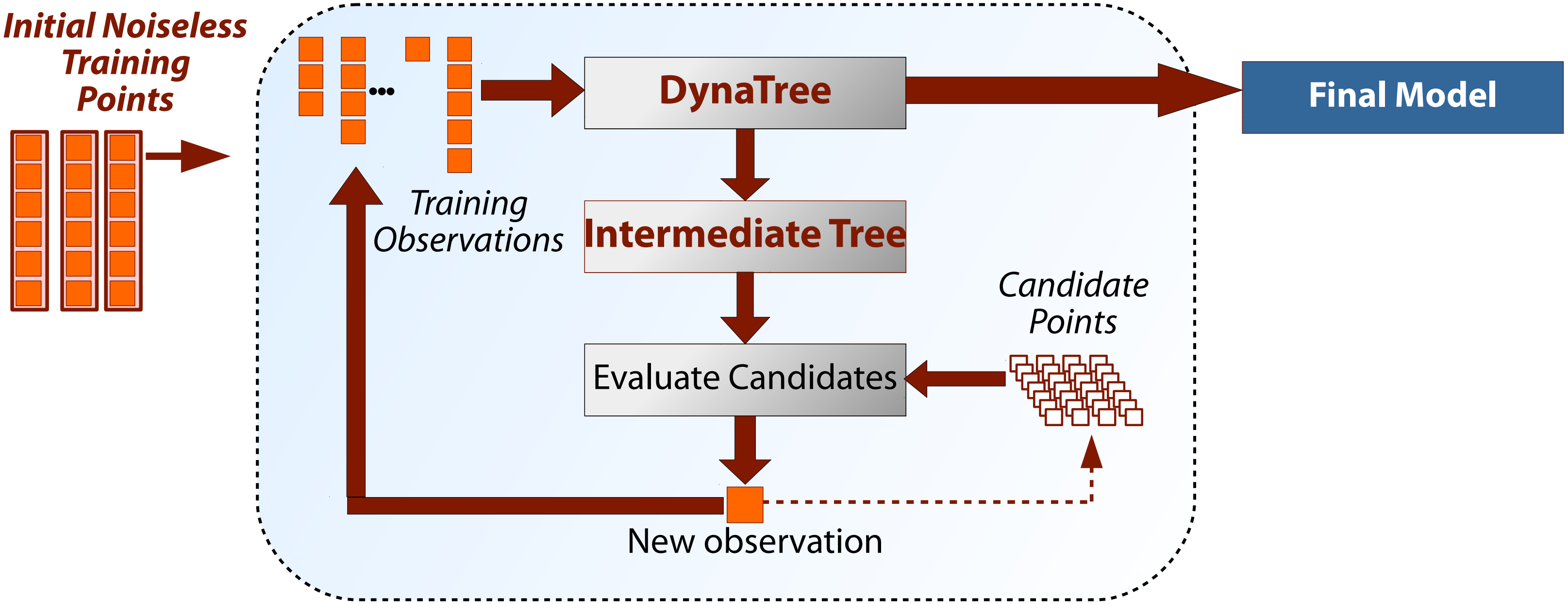


Challenges:

1) Noisy data→ Completely wrong IM

2) Overfitting

3) Too many updates to the IM



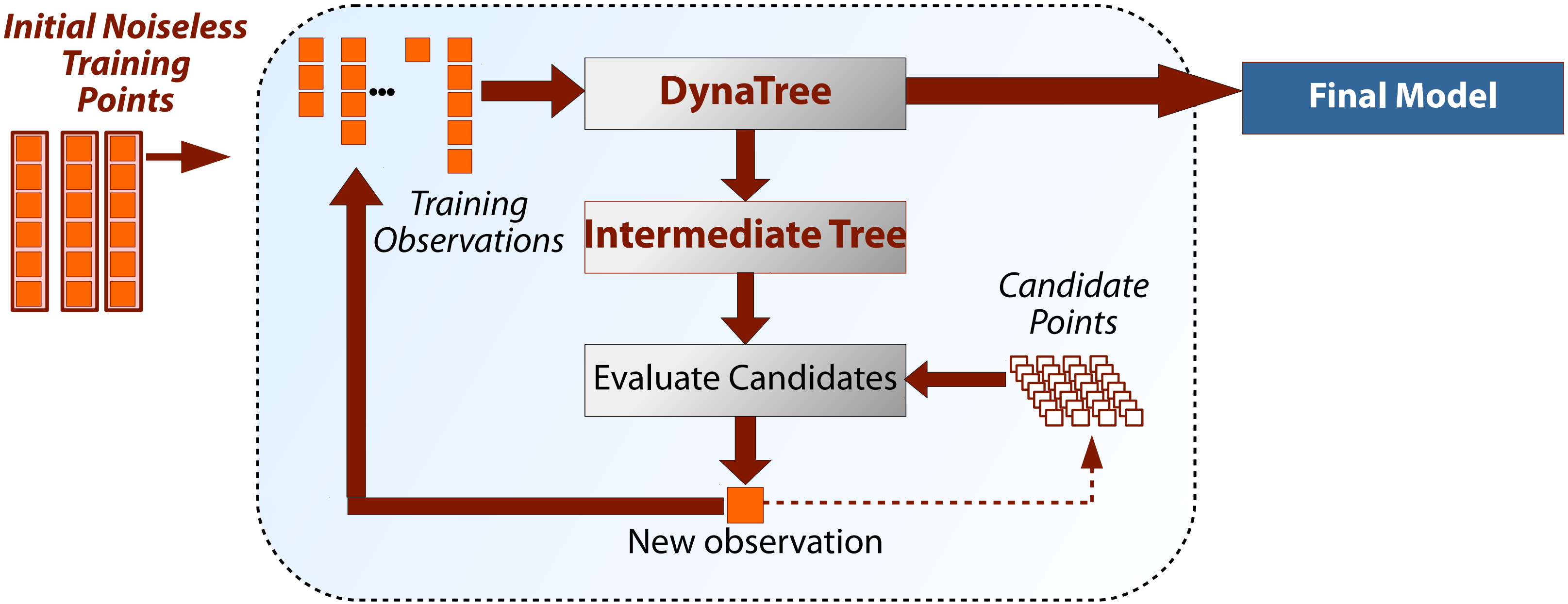
Challenges:

1) Noisy data → Completely wrong IM

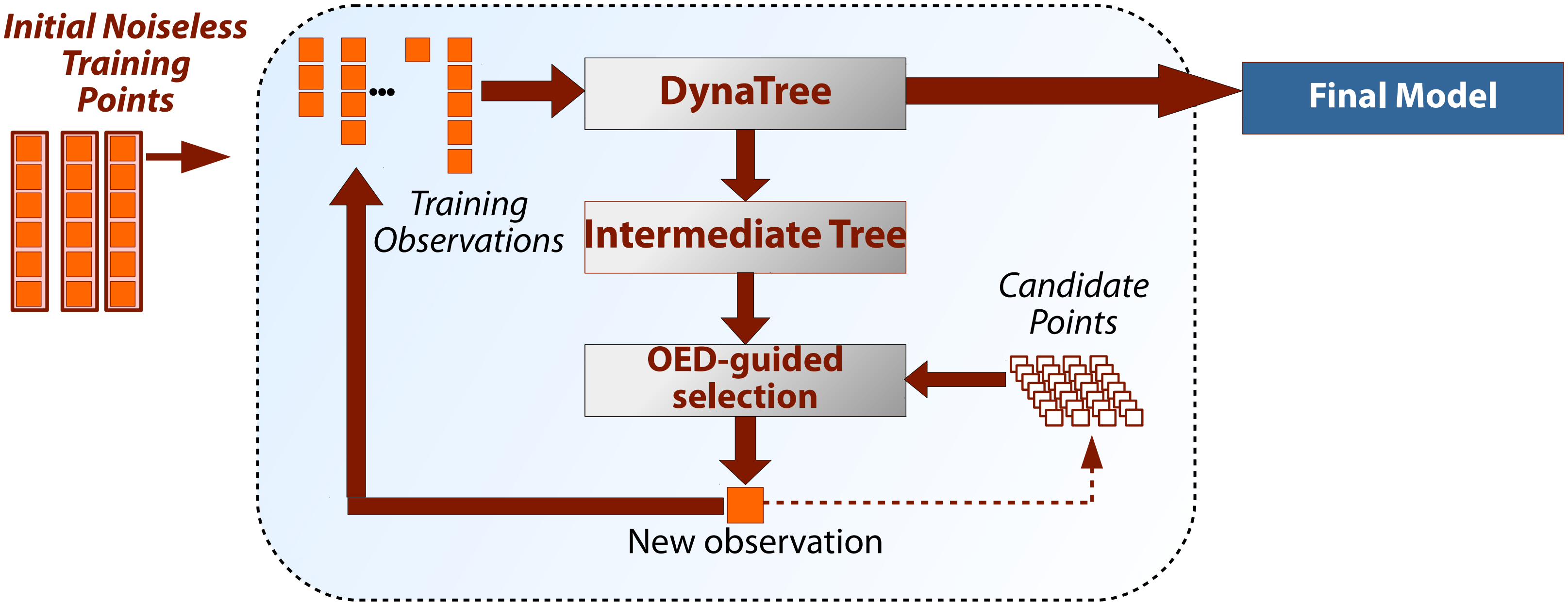
2) Overfitting

3) Too many updates to the IM

THE UNIVERSITY of EDINBURGH
informatics



Challenges:
1) Noisy data→ Completely wrong IM
2) Overfitting
3) Too many updates to the IM



Challenges:
1) Noisy data→ Completely wrong IM
2) Overfitting
3) Too many updates to the IM

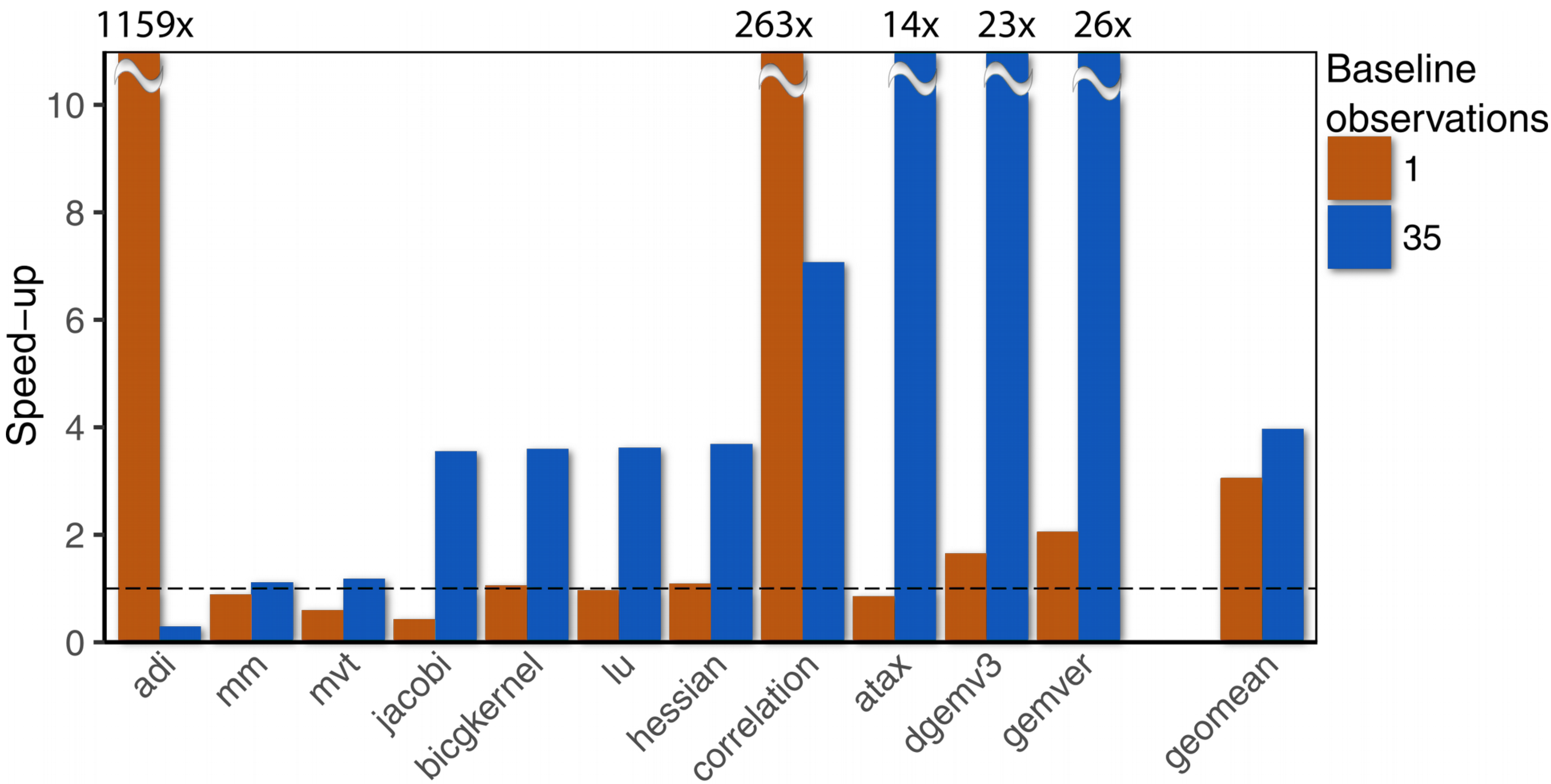
Test Case:

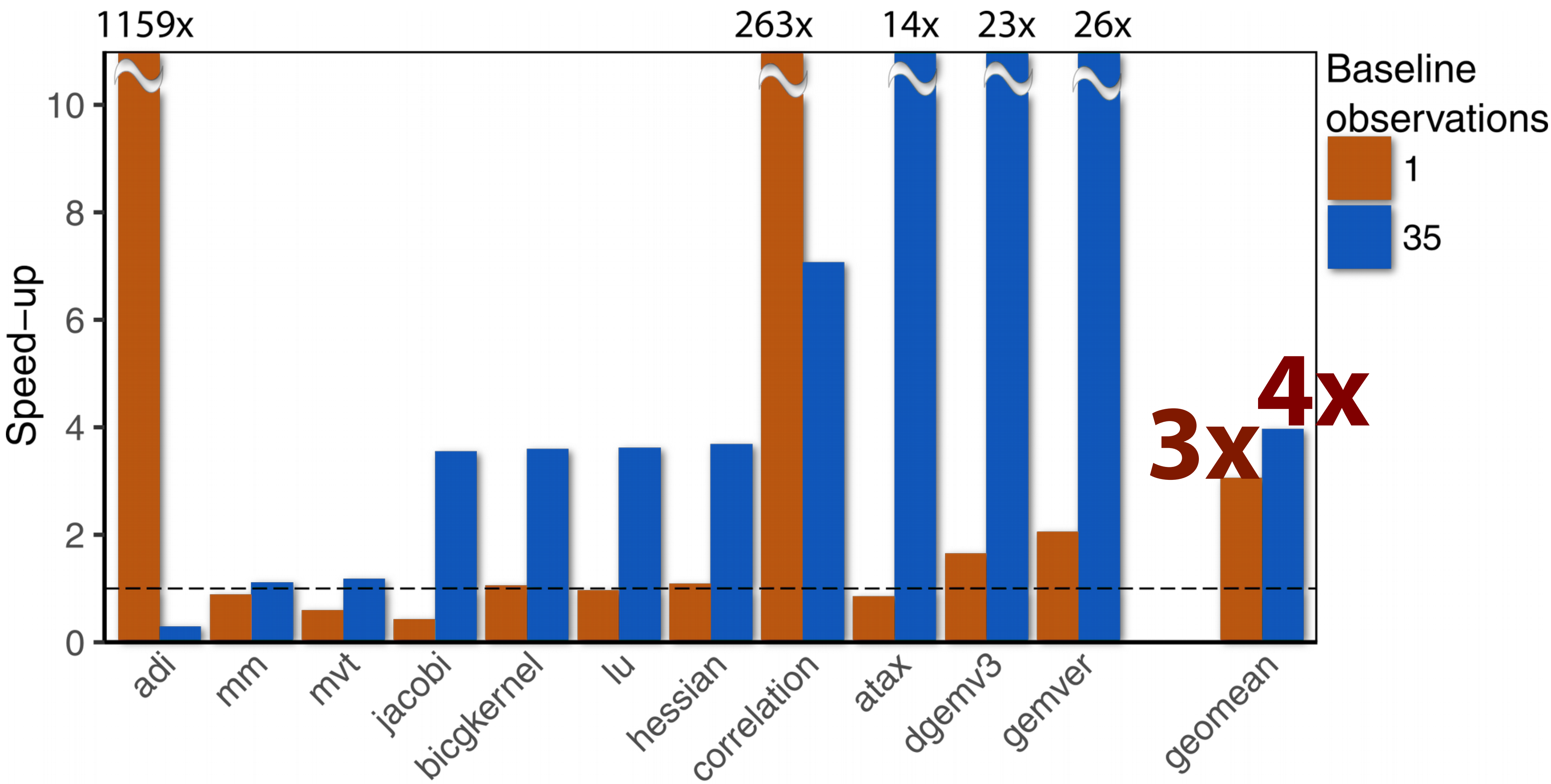
Prediction of runtime under different optimizations

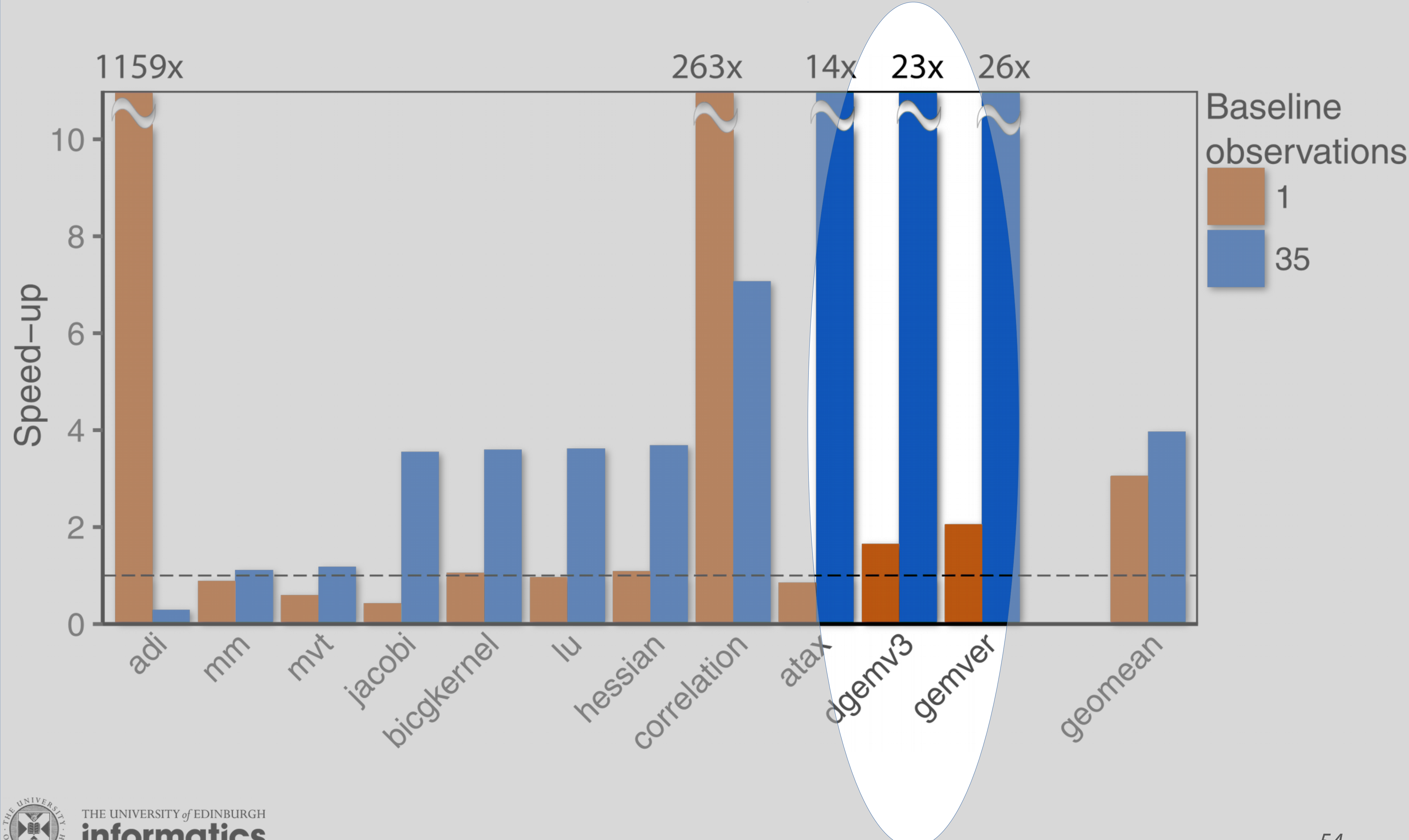
SPAPT Benchmarks

Huge optimization space
$5x10^8$ to $1x10^{27}$

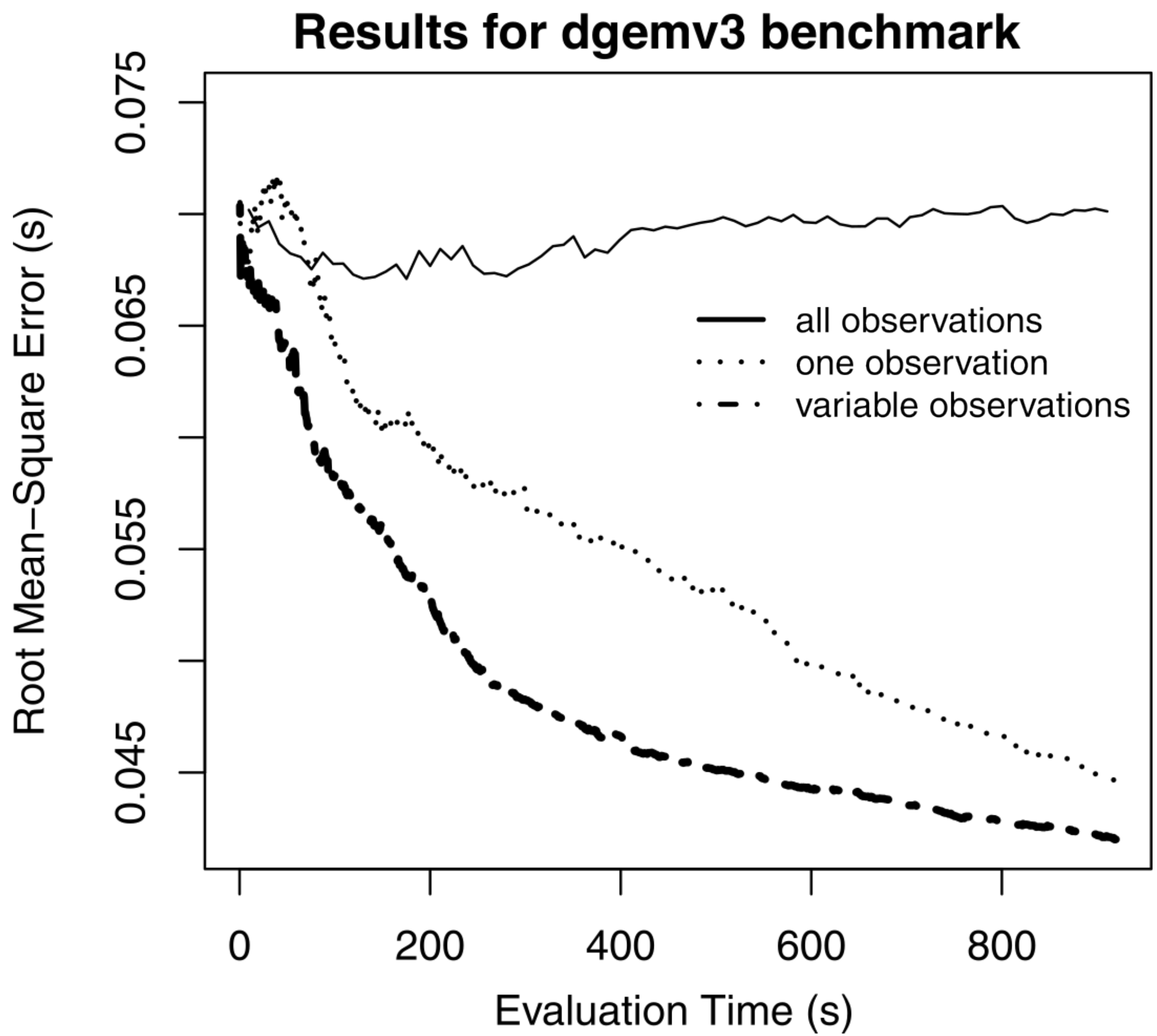
Benchmarks with both low and high noise


1159x
263x
14x
23x
26x
Speed−up
0
2
4
6
8
10
Baseline observations
1
35
adi
mm
mvt
jacobi
bicgkernel
lu
hessian
correlation
atax
dgemv3
gemver
geomean


1159x
263x
14x
23x
26x
Speed−up
0
2
4
6
8
10
Baseline observations
1
35
3x
4x
adi
mm
mvt
jacobi
bicgkernel
lu
hessian
correlation
atax
dgemv3
gemver
geomean



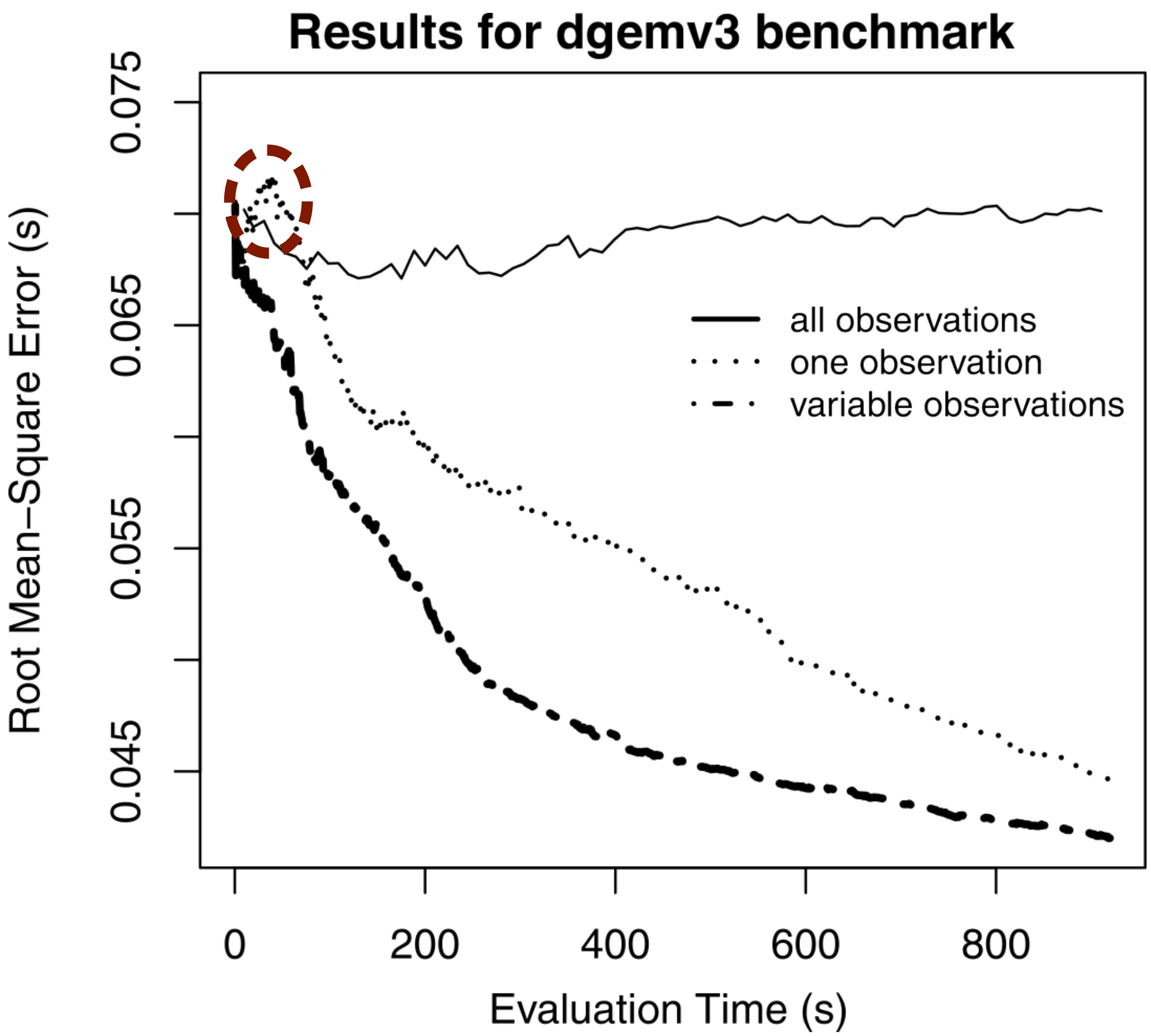
THE UNIVERSITY of EDINBURGH
informatics



High noise for some points

Handles noisy points

Does not waste time for the rest

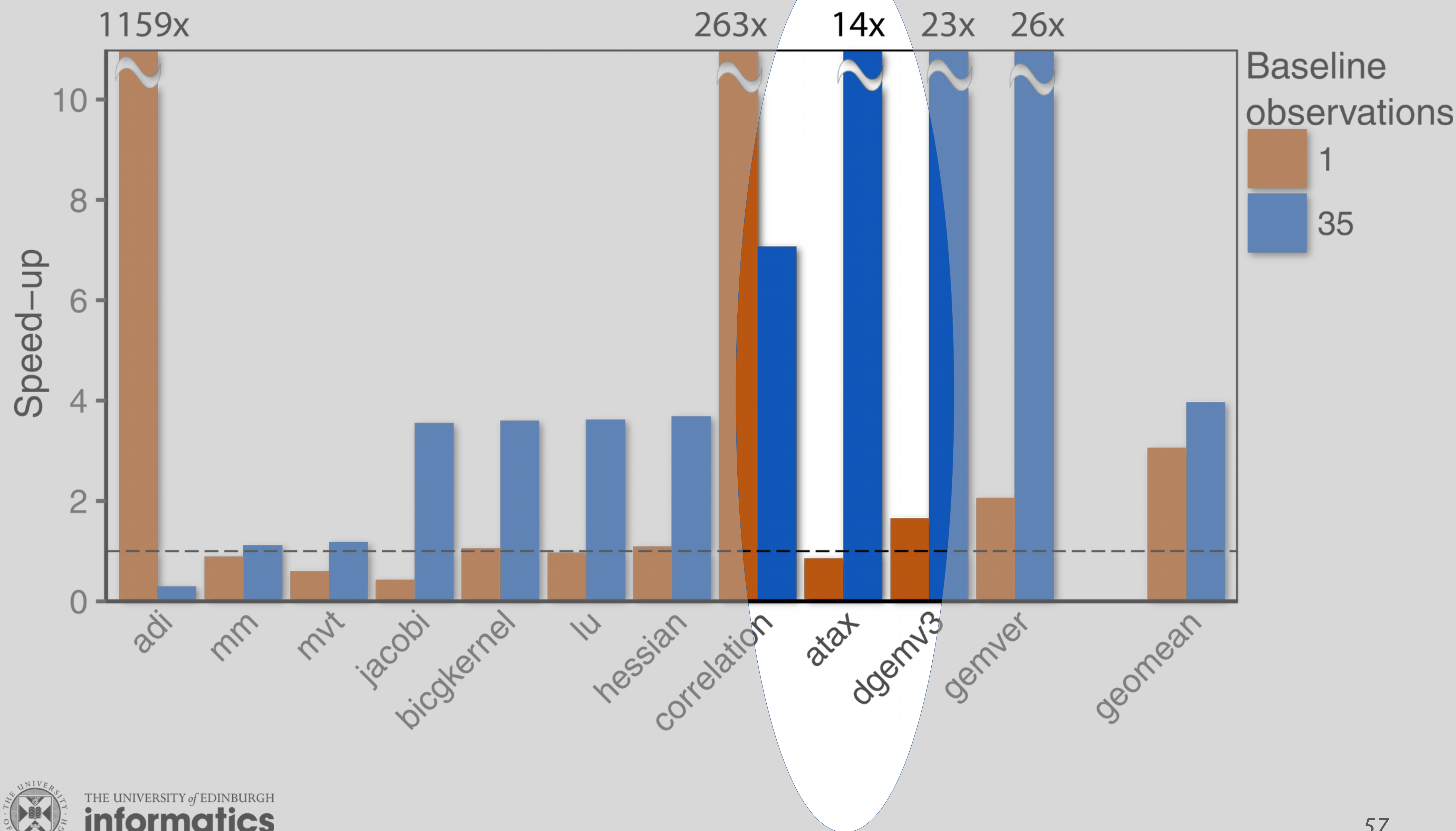


High noise for some points

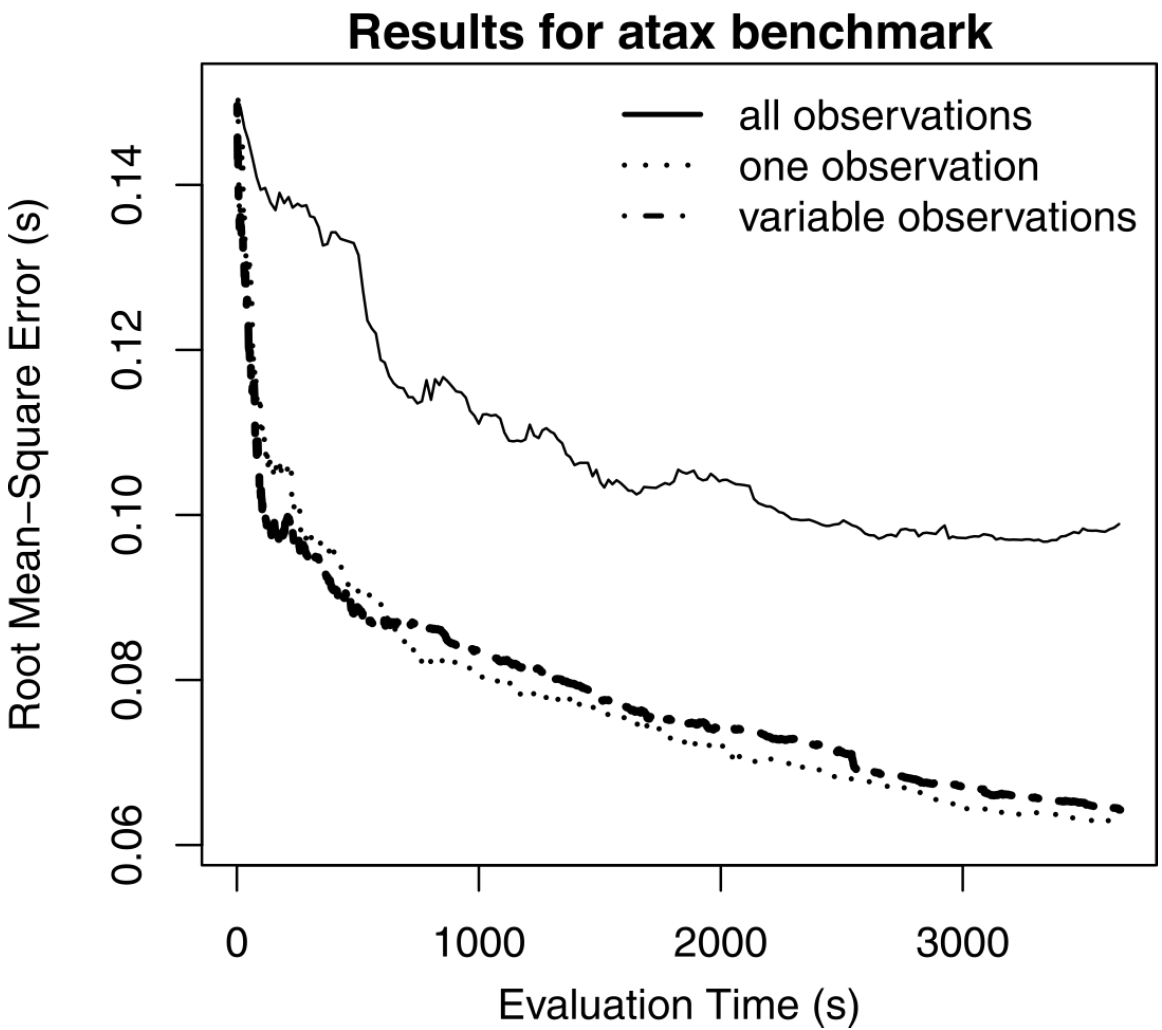
Handles noisy points

Does not waste time for the rest

THE UNIVERSITY of EDINBURGH
informatics



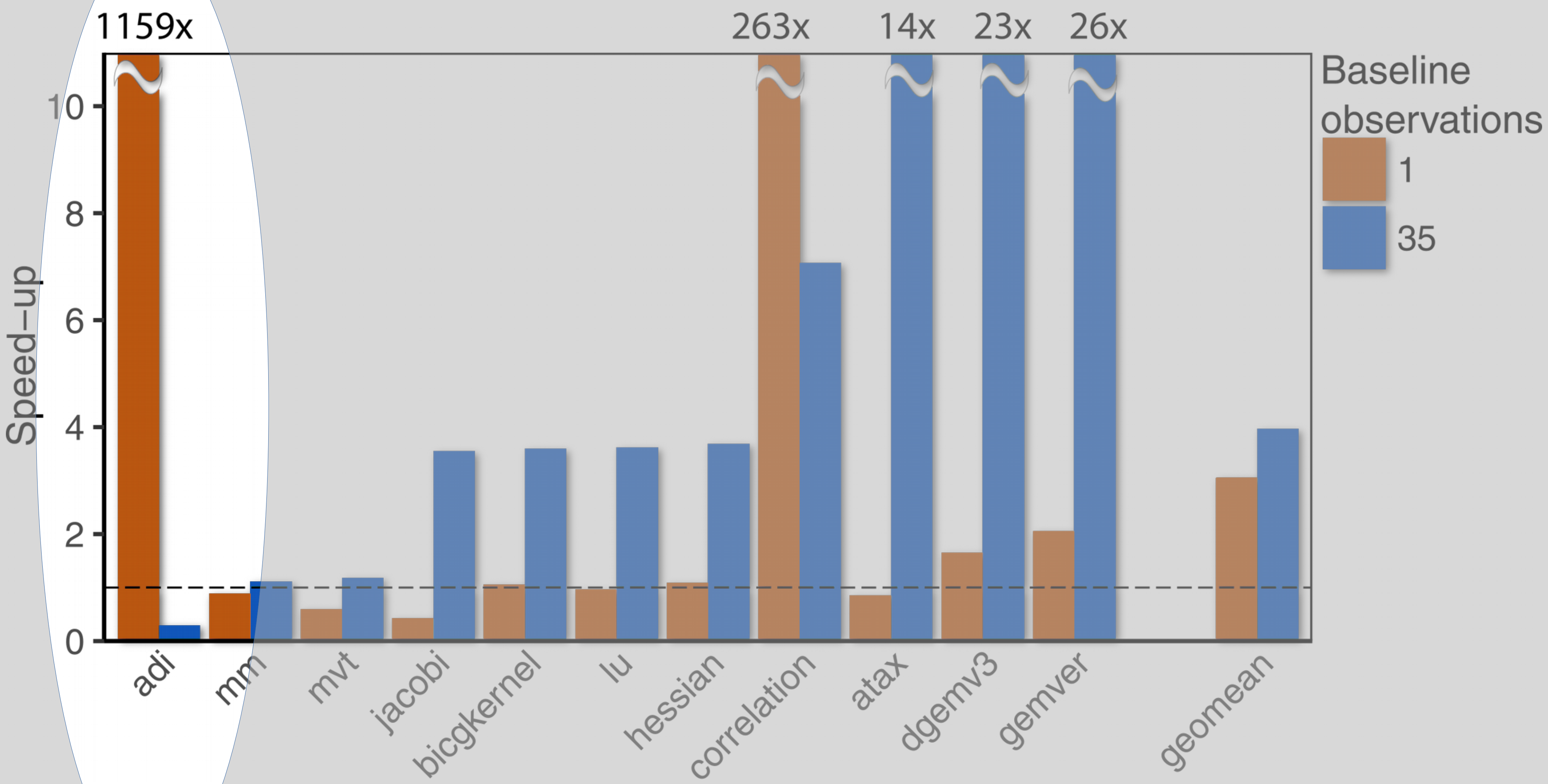
1159x
263x
14x
23x
26x
Baseline observations
1
35
Speed−up
0
2
4
6
8
10
adi
mm
mvt
jacobi
bicgkernel
lu
hessian
correlation
atax
dgemv3
gemver
geomean

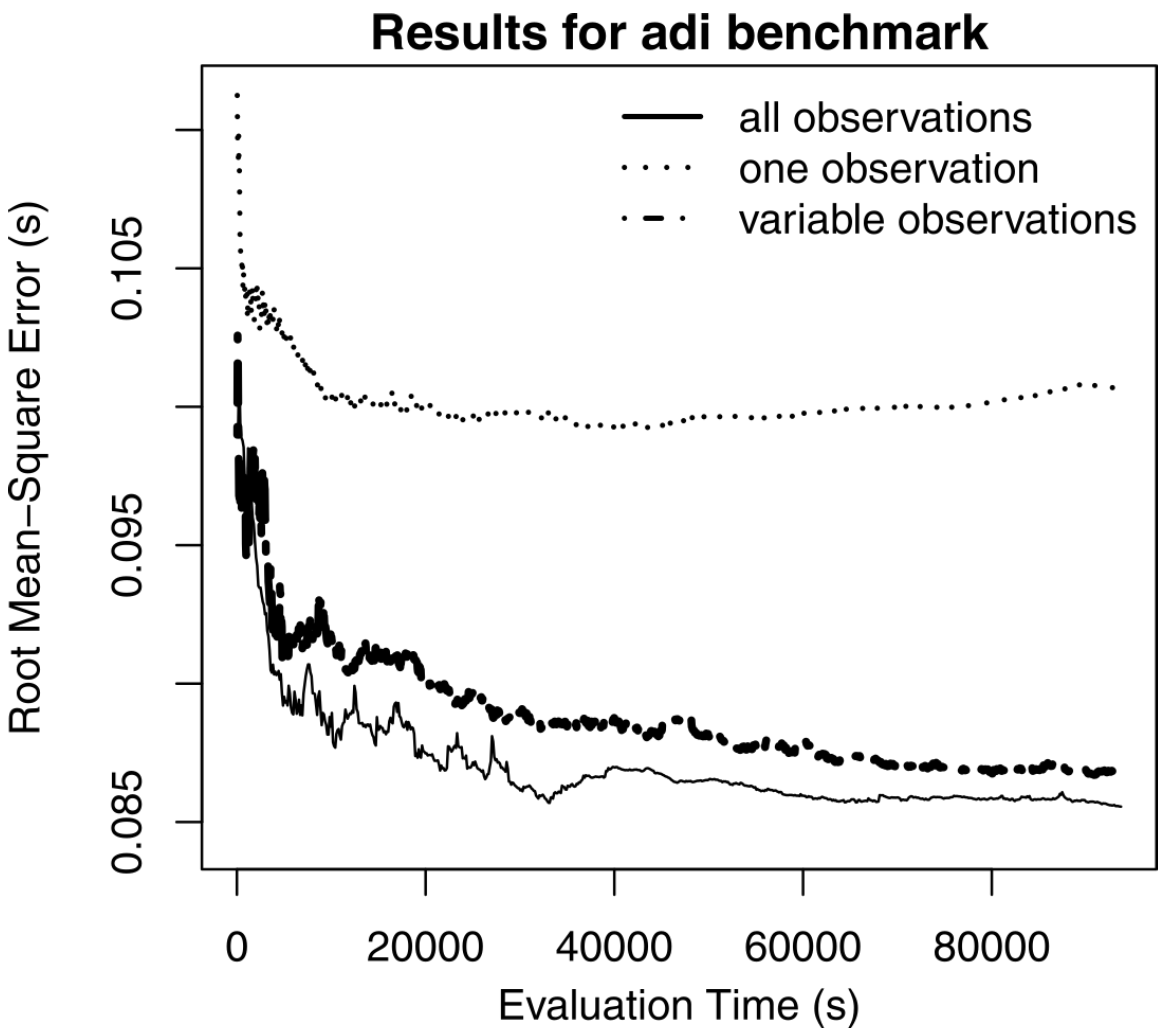


Low noise

Similar to one observation

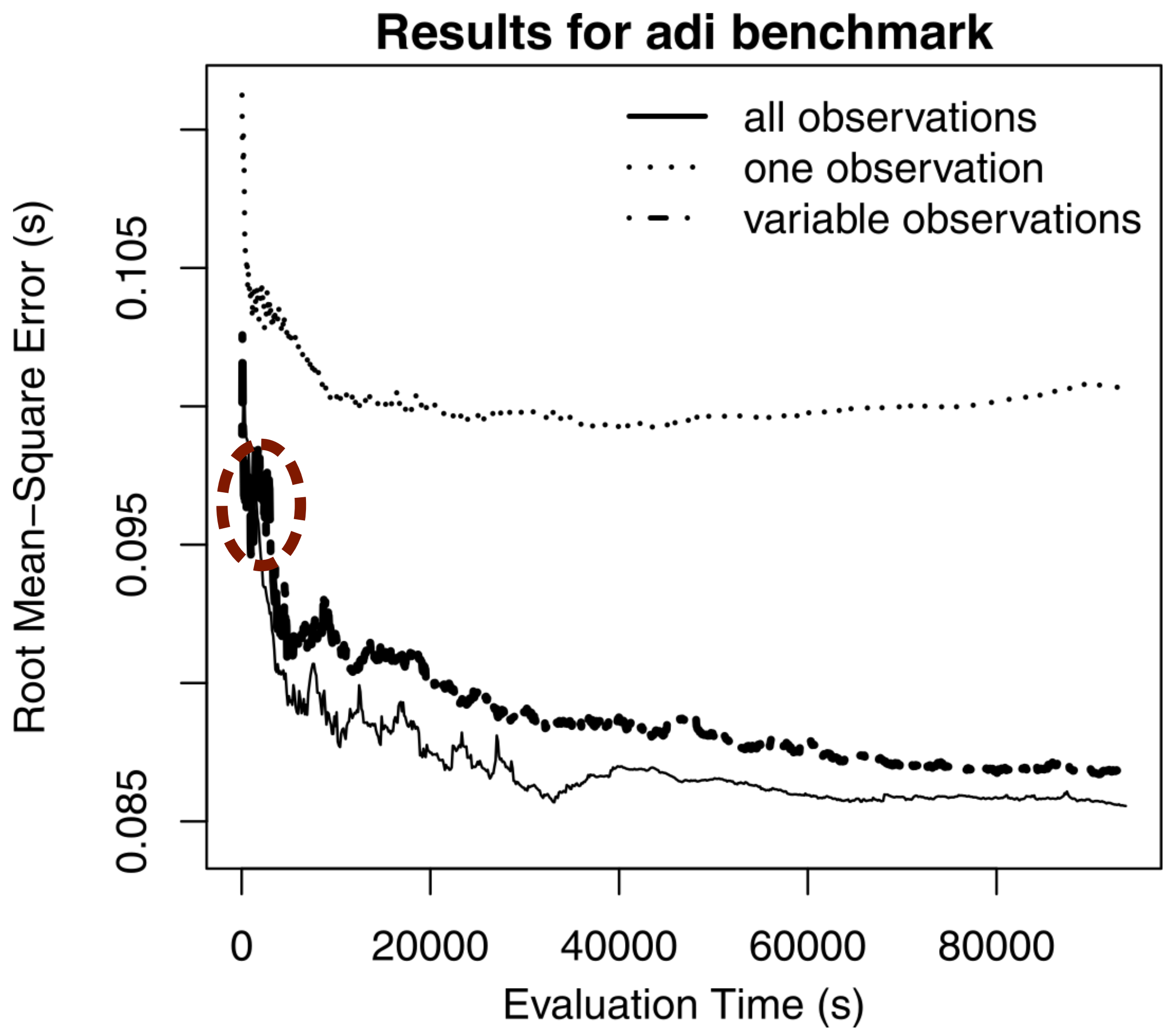
No noise characterization needed


1159x
263x
14x
23x
26x
Speed–up
0
2
4
6
8
10
adi
mm
mvt
jacobi
bicgkernel
lu
hessian
correlation
atax
dgemv3
gemver
geomean
Baseline observations
1
35



High noise

Similar to 35 observations



High noise

Similar to 35 observations

Conclusions

	Passive Learning	Active Learning	Our approach
Training Points	**Inefficient**	**Efficient**	**Efficient**
Profiling runs	**Inefficient**	**Inefficient**	**Efficient**

THE UNIVERSITY of EDINBURGH
informatics

	Passive Learning	Active Learning	Our approach
Training Points	Inefficient	Efficient	Efficient
Profiling runs	Inefficient	Inefficient	Efficient
Obser-vations	**Multiple and fixed**	**Multiple and fixed**	**One unless necessary**

THE UNIVERSITY *of* EDINBURGH
informatics

	Passive Learning	Active Learning	Our approach
Training Points	Inefficient	Efficient	Efficient
Profiling runs	Inefficient	Inefficient	Efficient
Obser-vations	Multiple and fixed	Multiple and fixed	One unless necessary
Assum-ptions?	**Enough runs for noise**	**Enough runs for noise**	**None**

THE UNIVERSITY *of* EDINBURGH
informatics

4x faster training

4x faster training
No assumptions

4x faster training
No assumptions
No risk

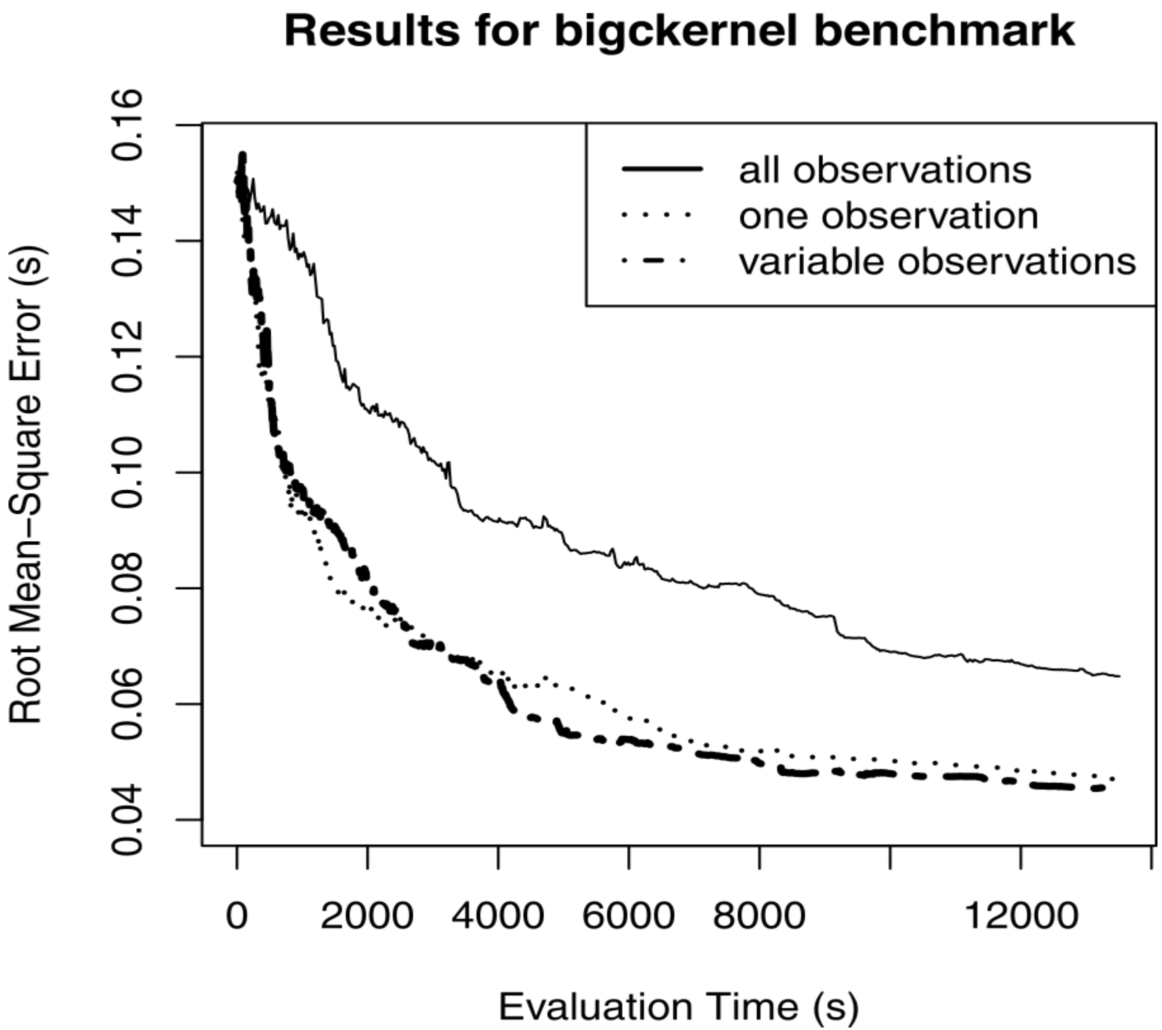
Thanks!

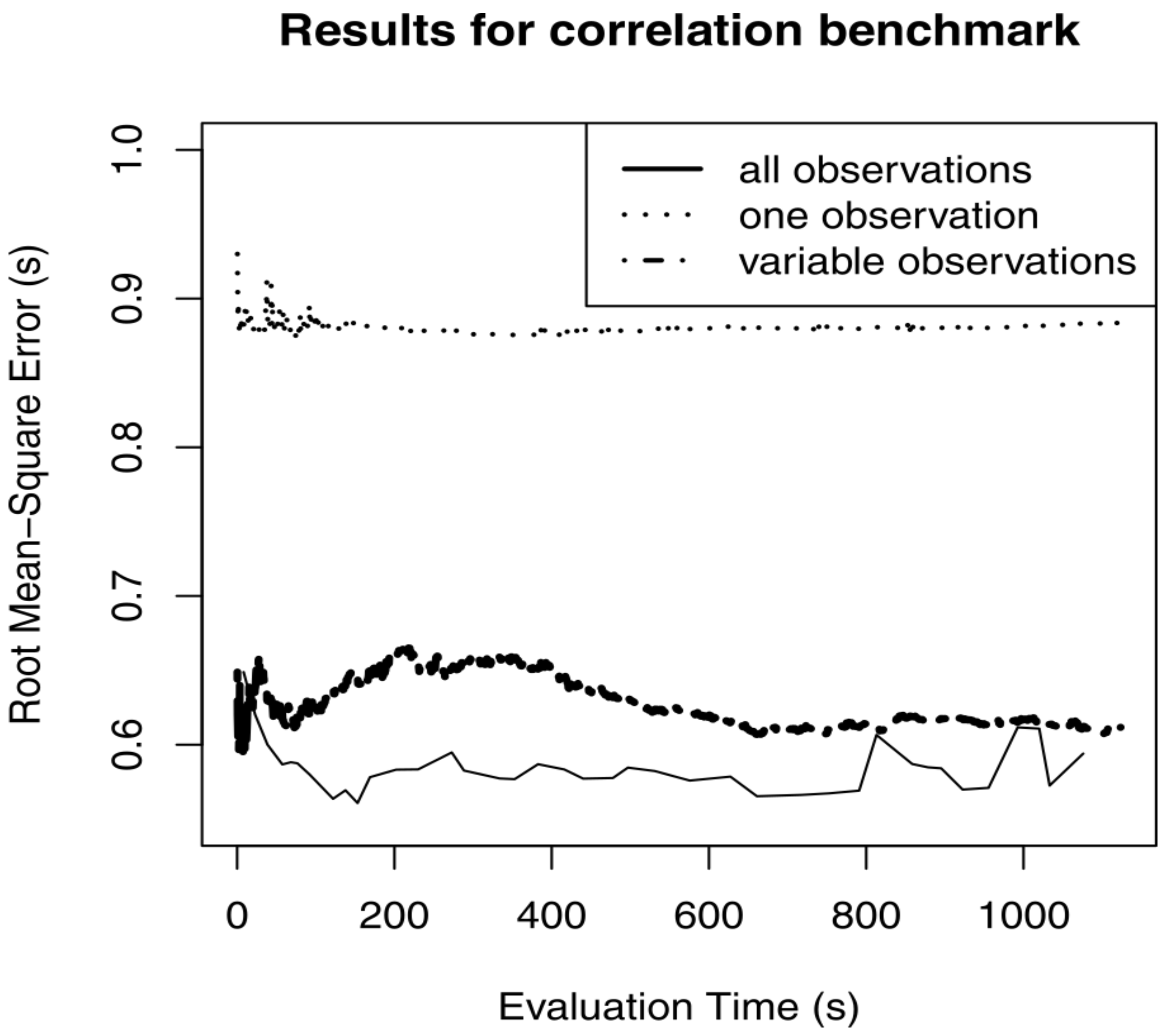

Minimizing the Cost of Iterative Compilation with Active Learning

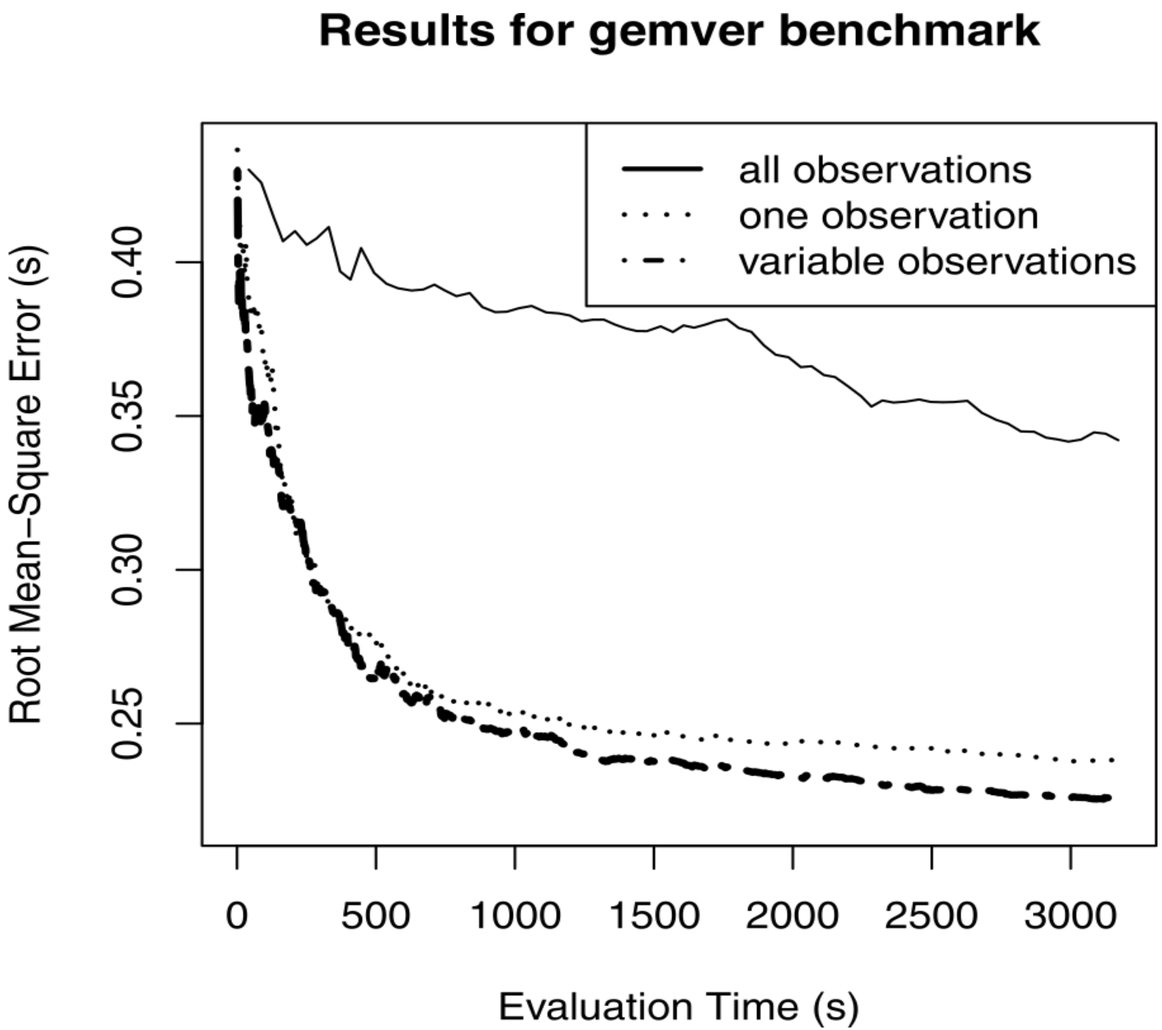
https://goo.gl/nG92CN

wfogilvie@gmail.com

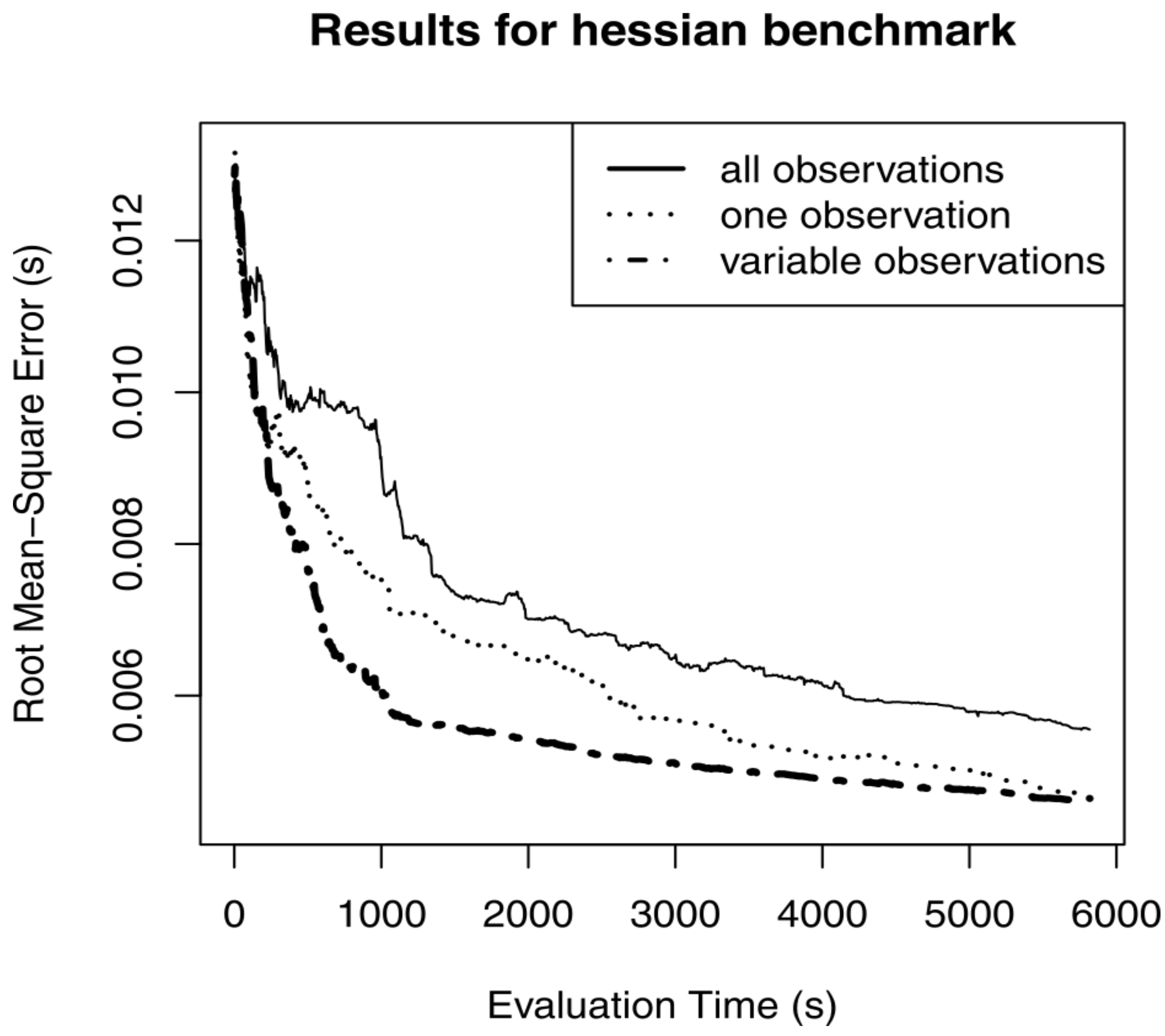
ppetoumenos@gmail.com


Results for bigckernel benchmark
all observations
one observation
variable observations
Root Mean−Square Error (s)
0.04
0.06
0.08
0.10
0.12
0.14
0.16
0
2000
4000
6000
8000
12000
Evaluation Time (s)

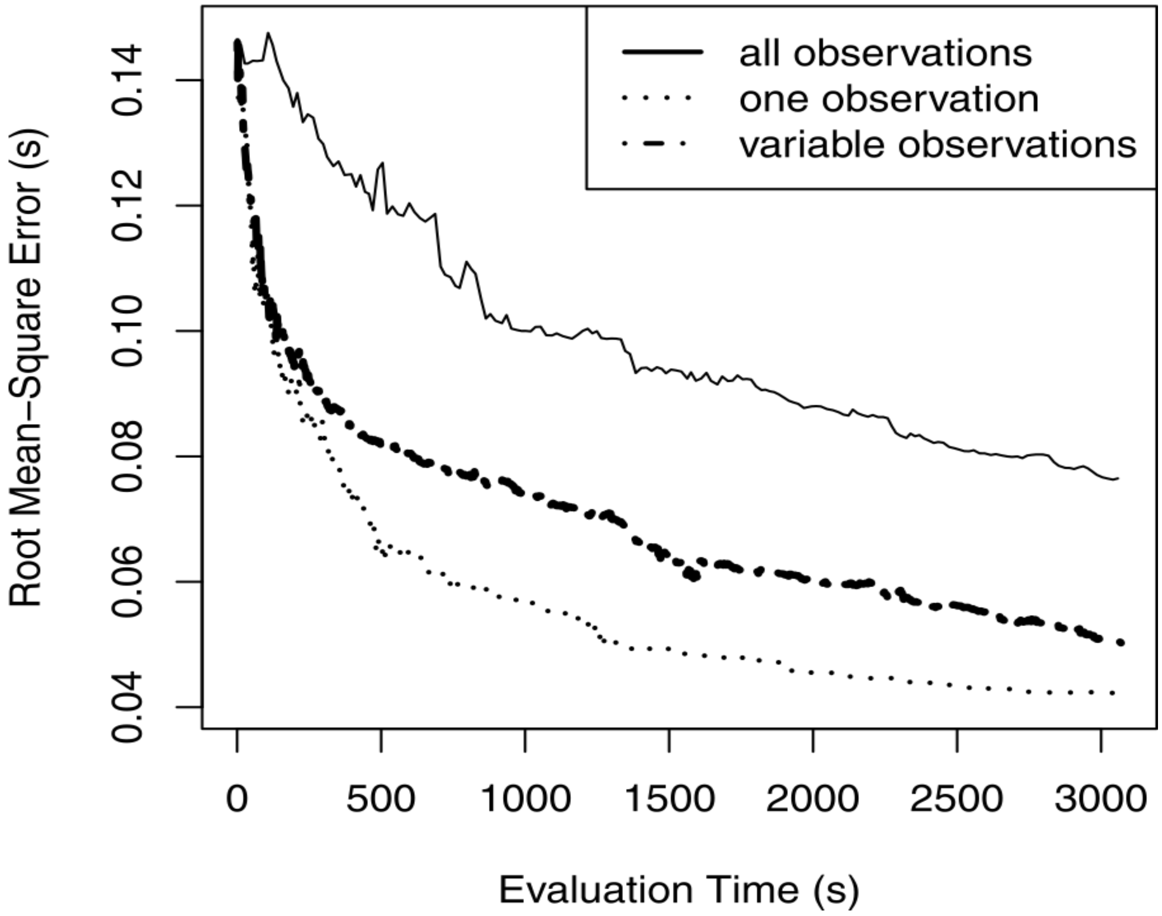

Results for correlation benchmark
all observations
one observation
variable observations
Root Mean−Square Error (s)
1.0
0.9
0.8
0.7
0.6
0
200
400
600
800
1000
Evaluation Time (s)



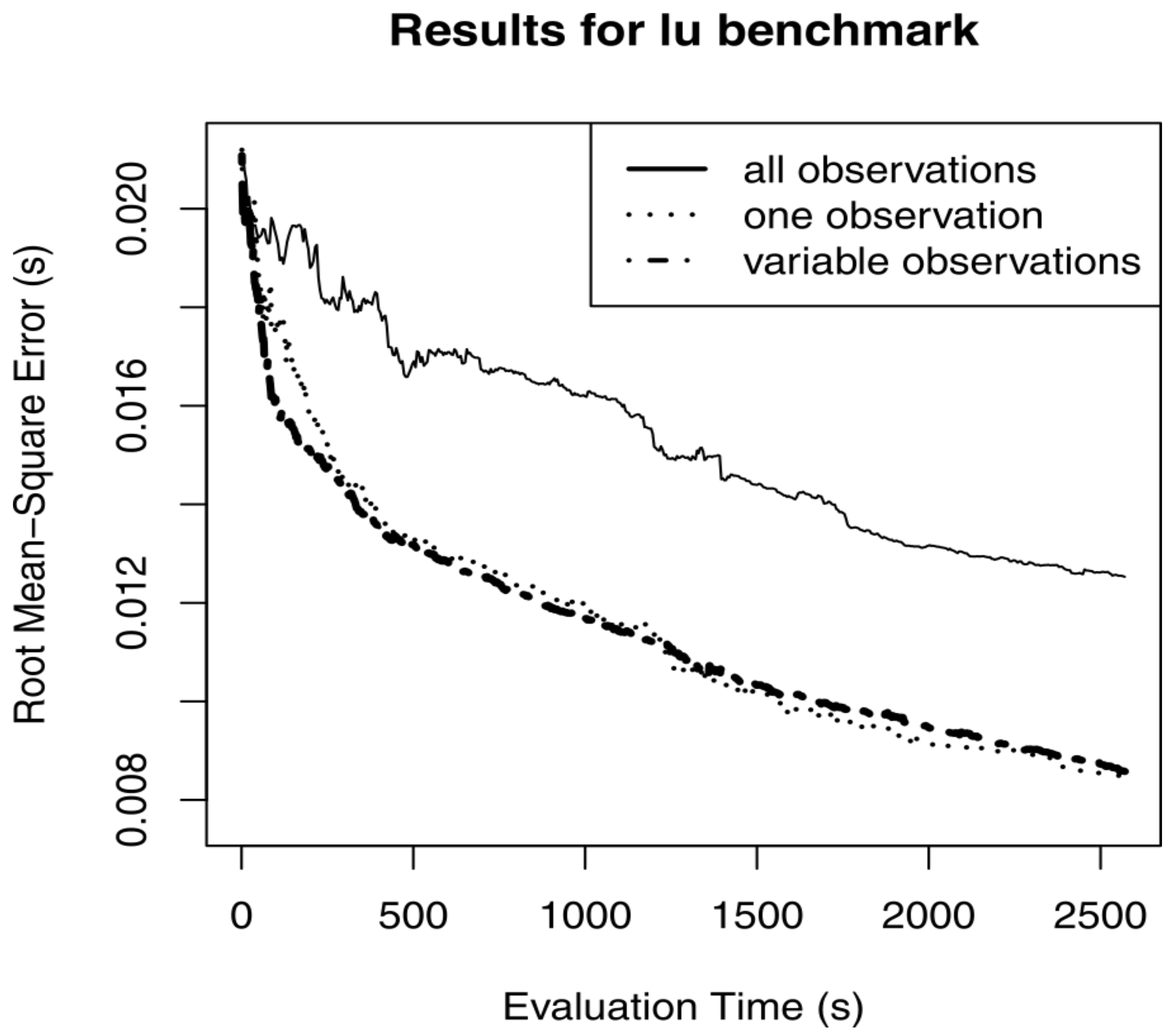
THE UNIVERSITY of EDINBURGH
informatics


Results for hessian benchmark
all observations
one observation
variable observations
Root Mean−Square Error (s)
0.006
0.008
0.010
0.012
0
1000
2000
3000
4000
5000
6000
Evaluation Time (s)

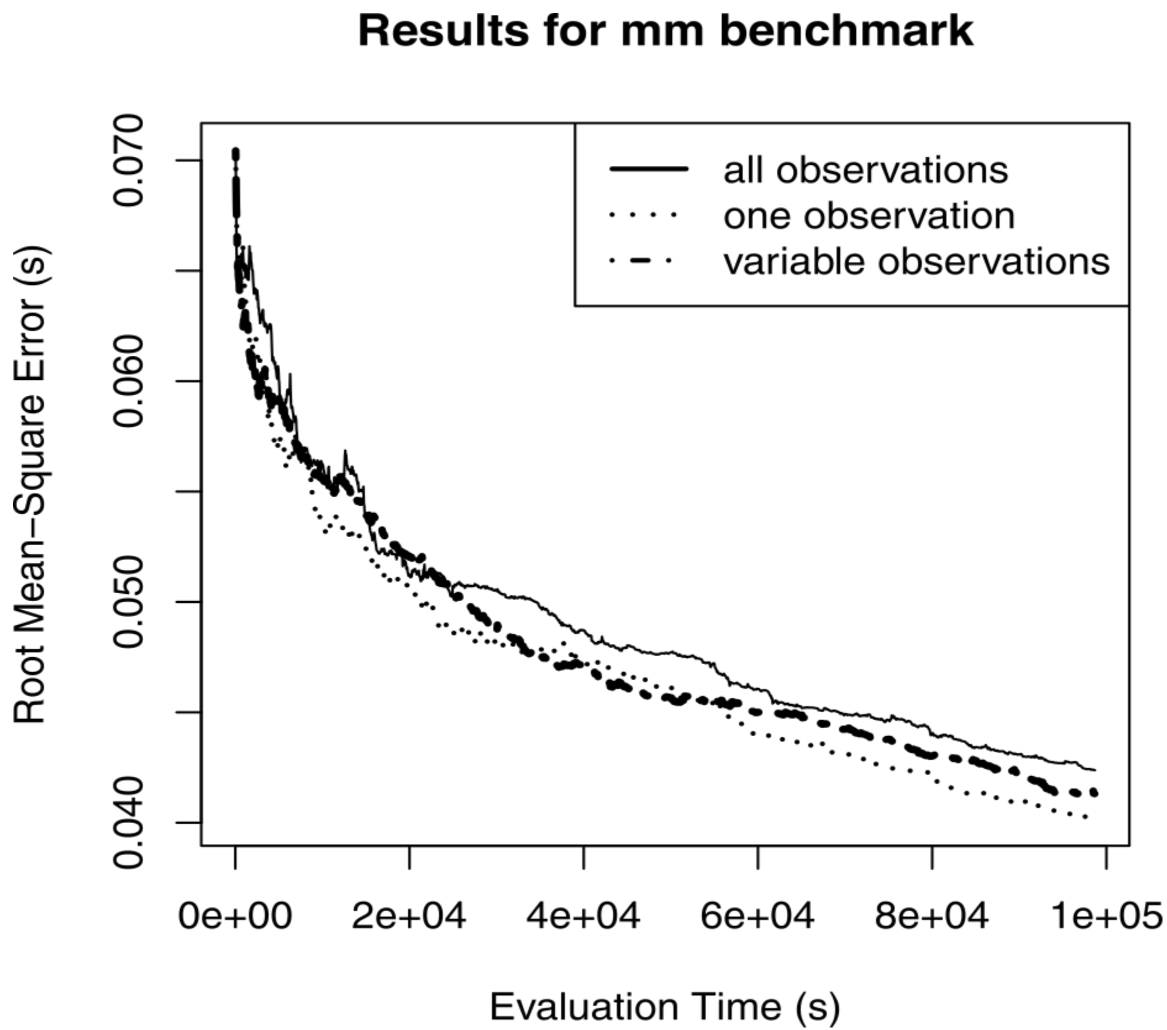
Results for jacobi benchmark



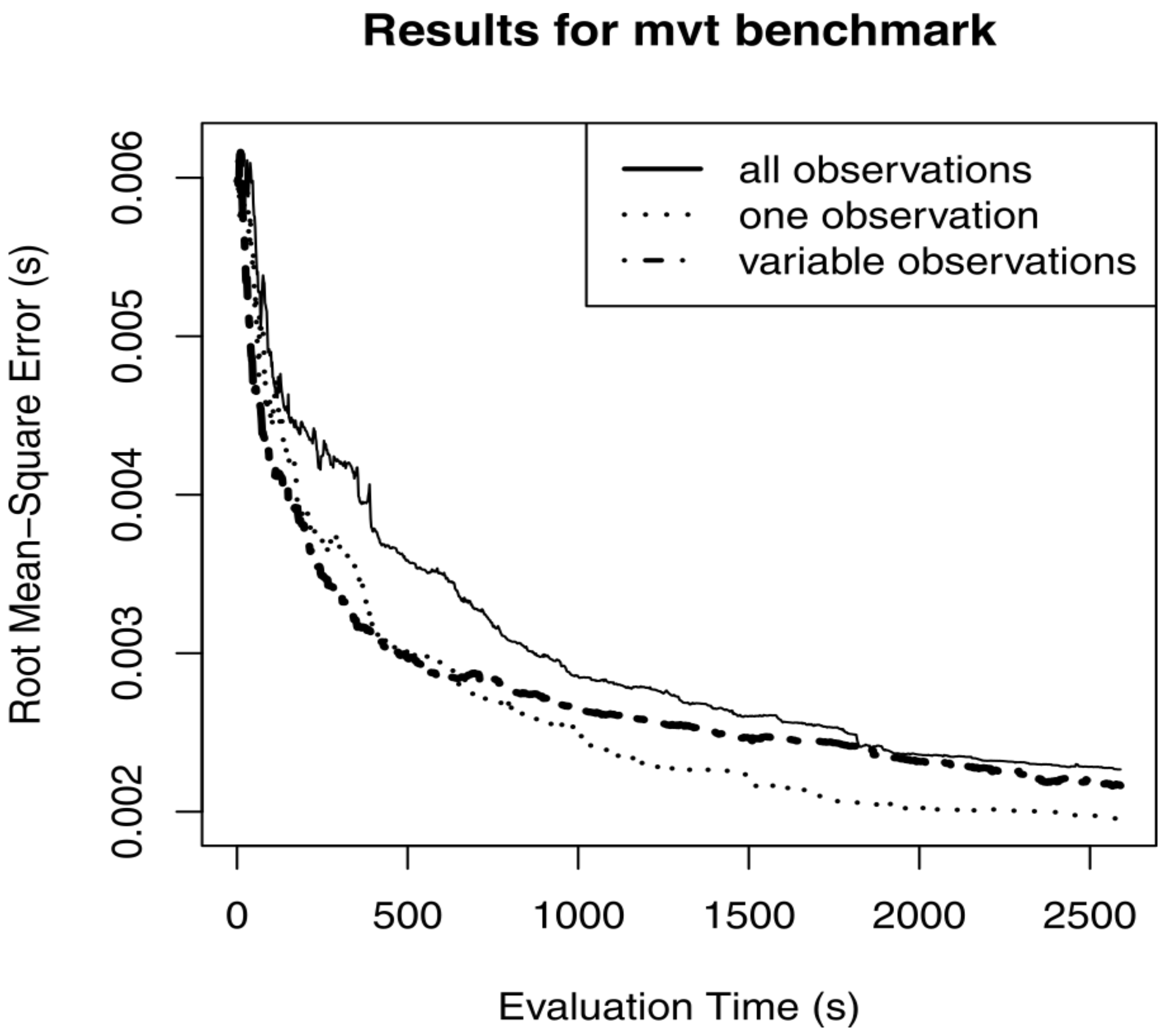
THE UNIVERSITY of EDINBURGH
informatics

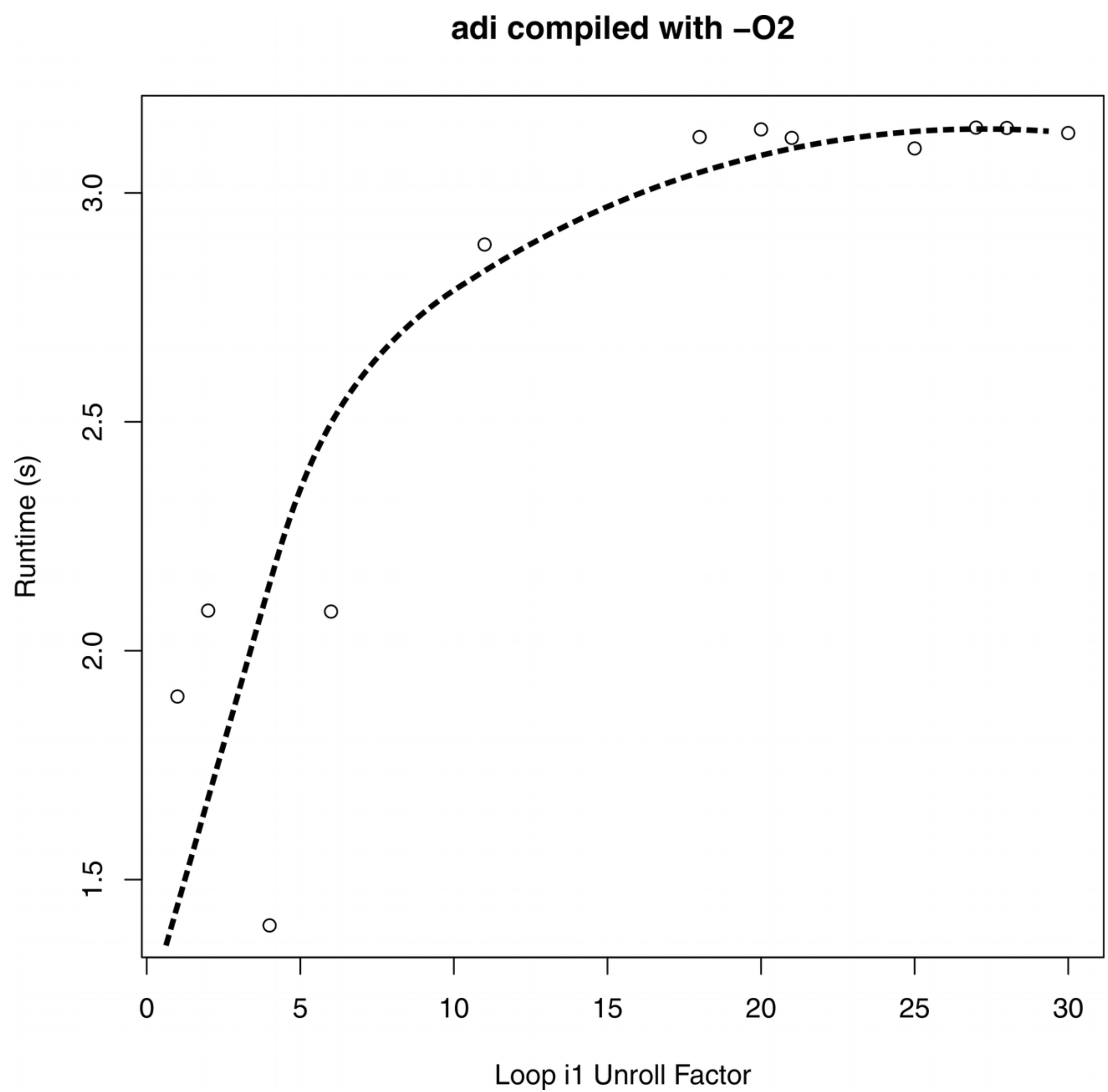


Results for lu benchmark
all observations
one observation
variable observations
Root Mean-Square Error (s)
0.008
0.012
0.016
0.020
0
500
1000
1500
2000
2500
Evaluation Time (s)

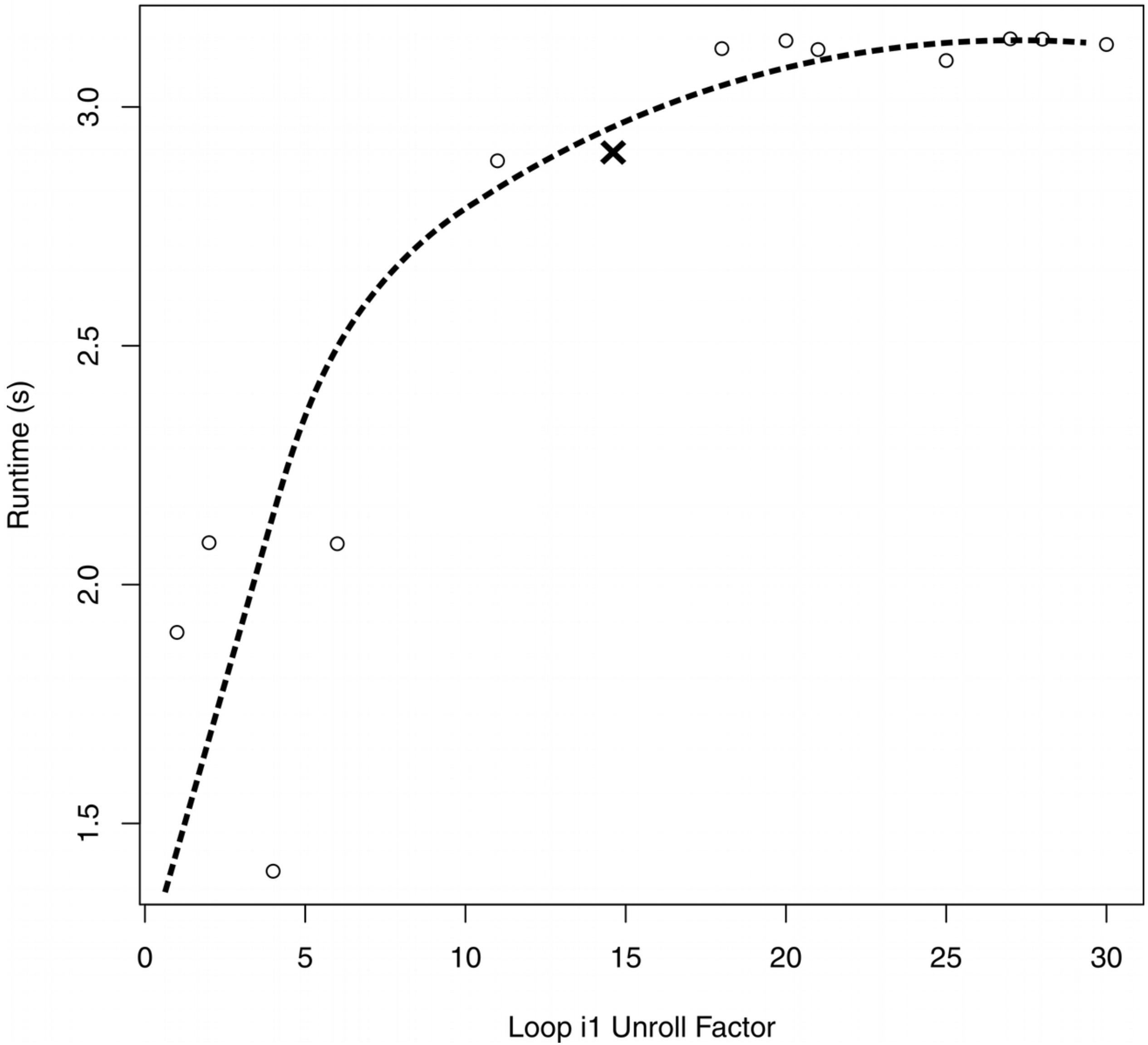


THE UNIVERSITY *of* EDINBURGH
informatics

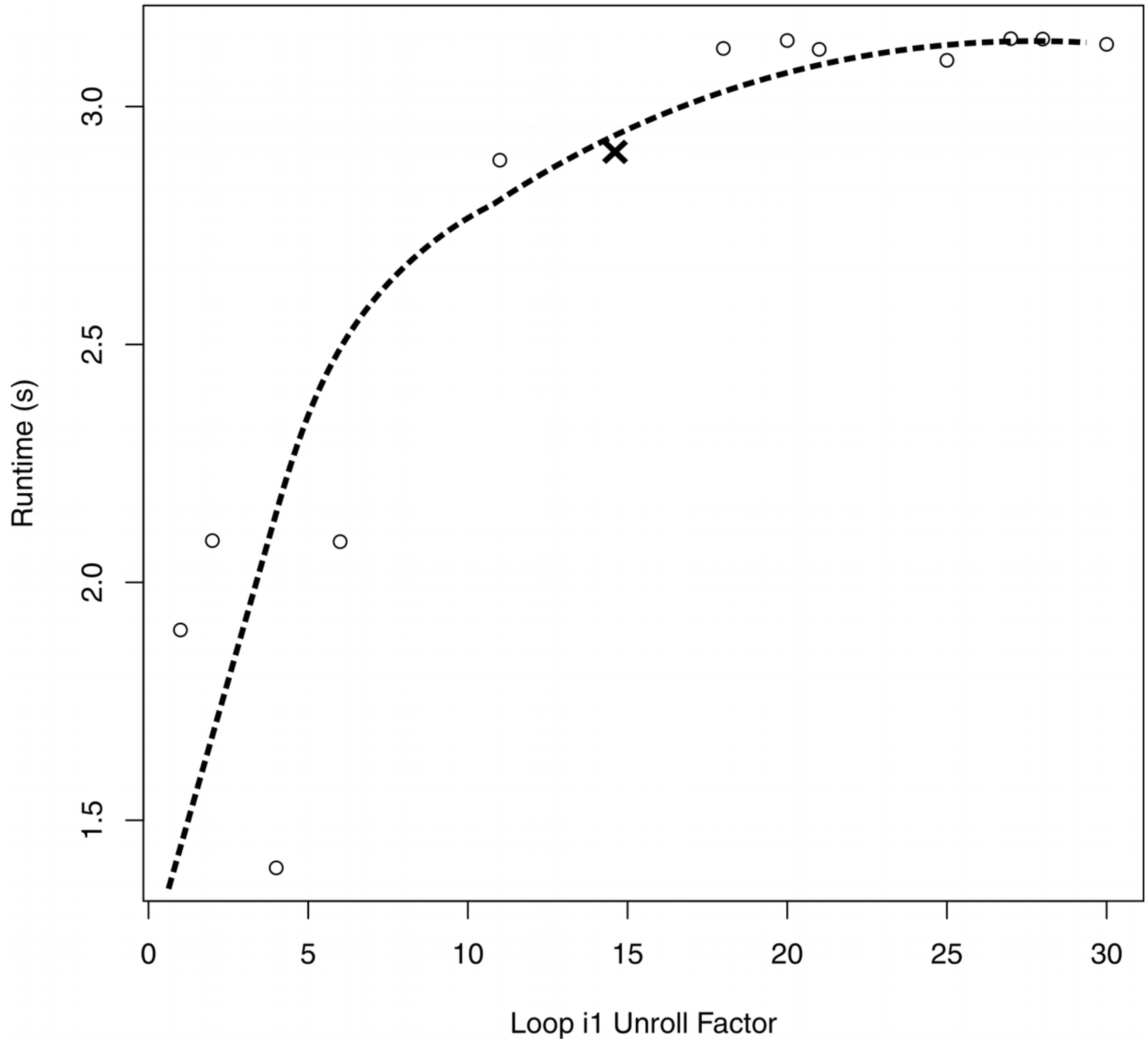

Results for mvt benchmark
all observations
one observation
variable observations
Root Mean−Square Error (s)
0.002
0.003
0.004
0.005
0.006
0
500
1000
1500
2000
2500
Evaluation Time (s)

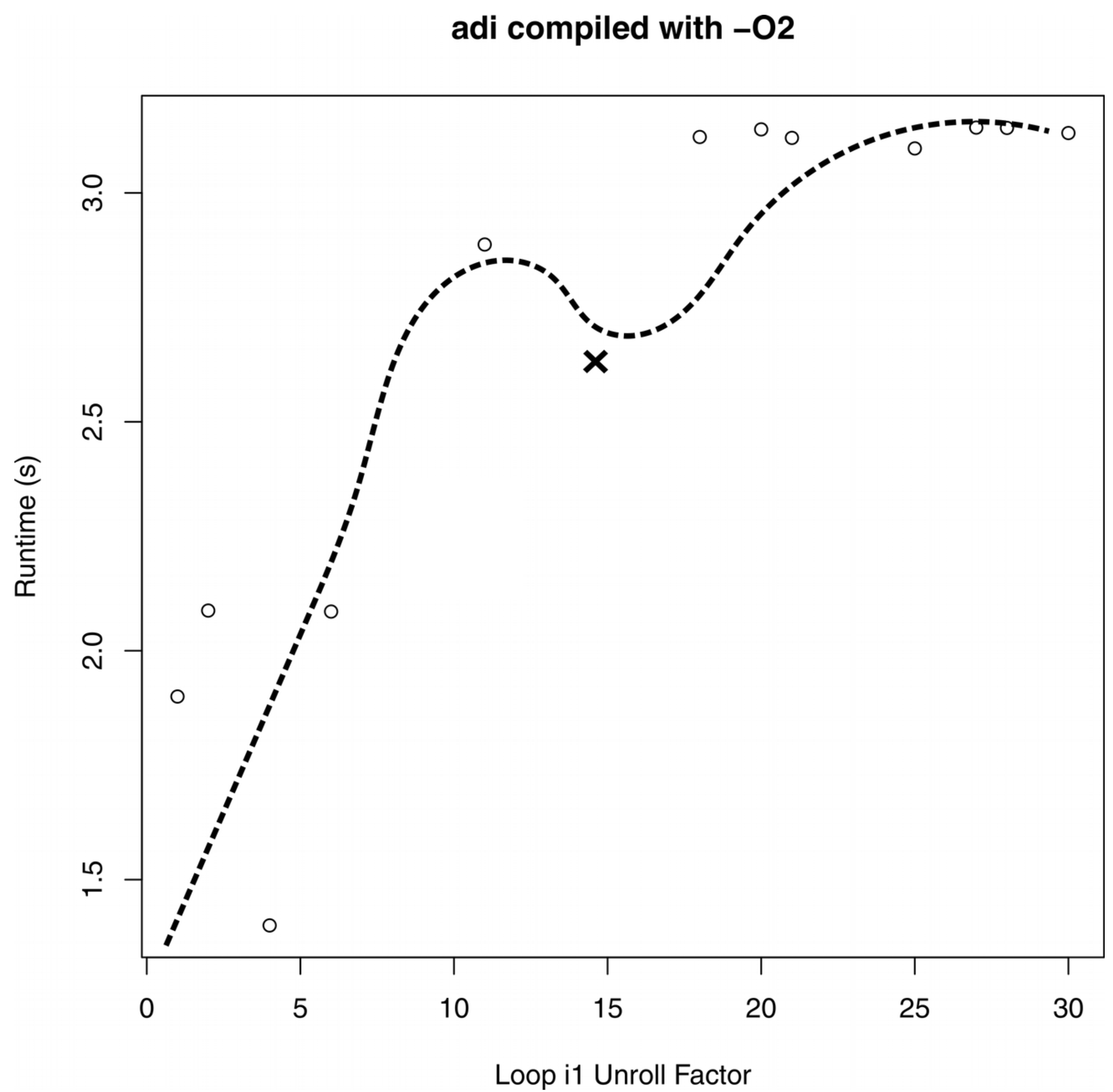

adi compiled with −O2
Runtime (s)
3.0
2.5
2.0
1.5
0
5
10
15
20
25
30
Loop i1 Unroll Factor

adi compiled with −O2



adi compiled with −O2





THE UNIVERSITY of EDINBURGH
informatics