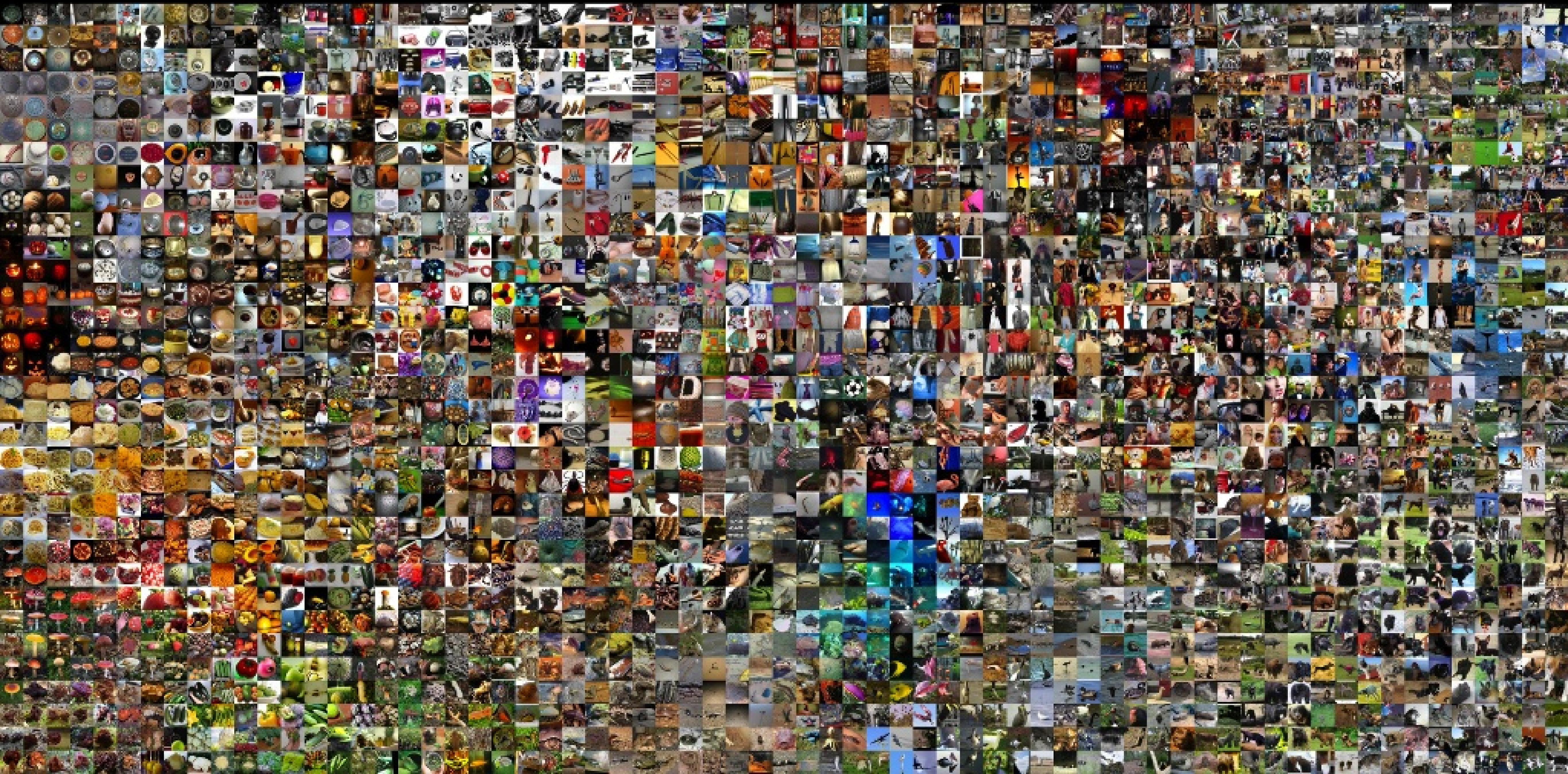
ERNEST

EFFICIENT PERFORMANCE PREDICTION FOR ADVANCED ANALYTICS ON APACHE SPARK

Shivaram Venkataraman, Zongheng Yang
Michael Franklin, Benjamin Recht, Ion Stoica

WORKLOAD TRENDS: ADVANCED ANALYTICS

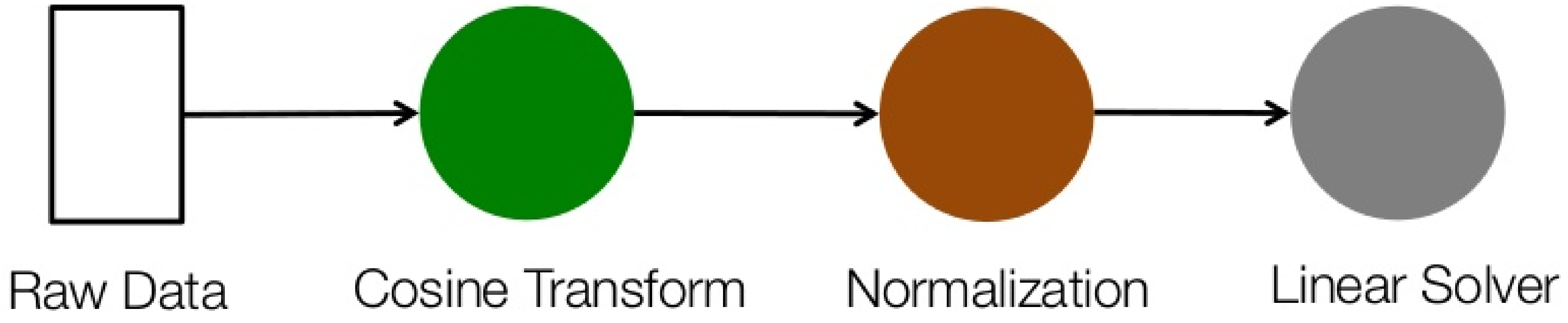


WORKLOAD TRENDS: ADVANCED ANALYTICS

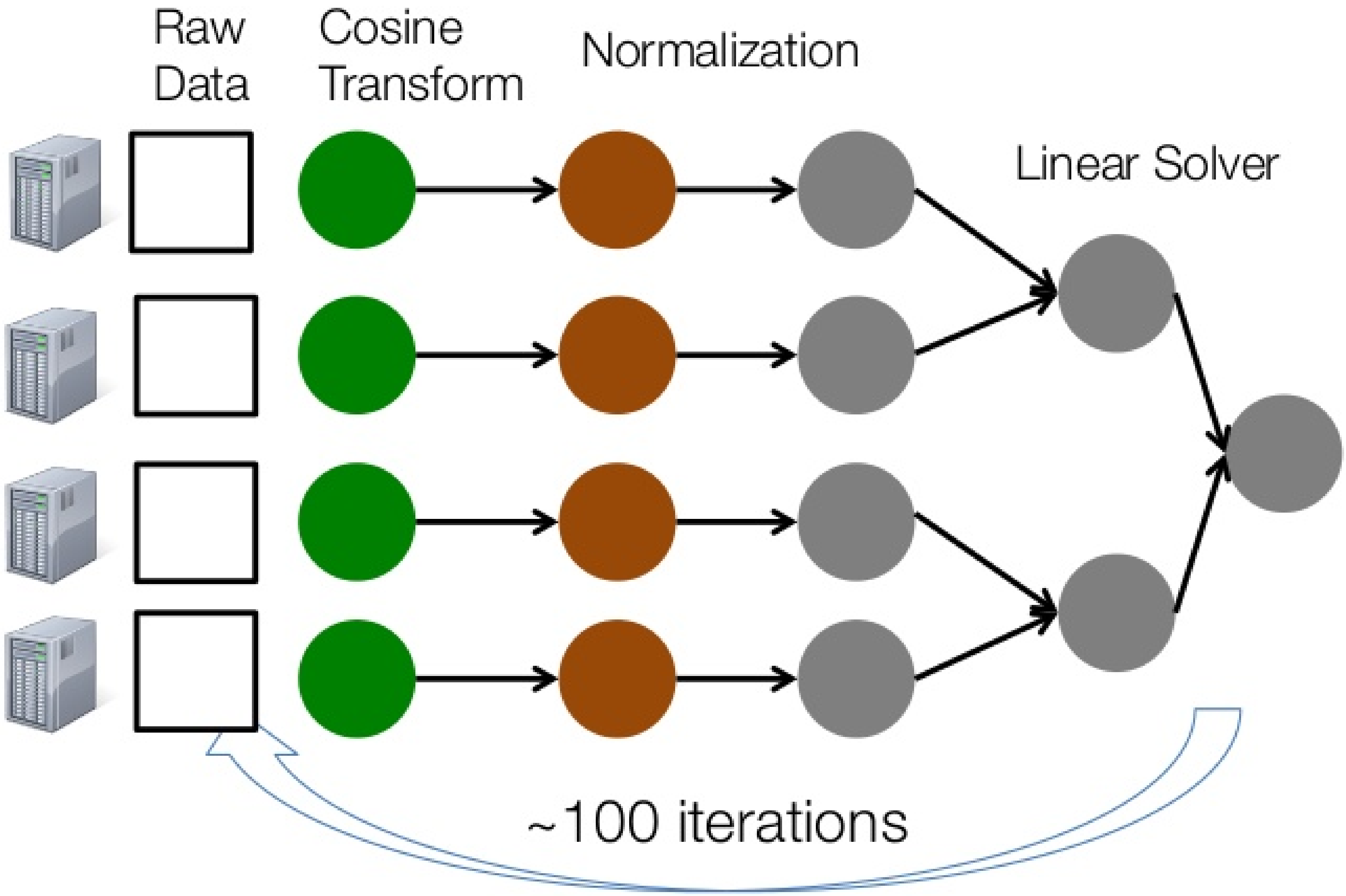
WORKLOAD TRENDS: ADVANCED ANALYTICS

WORKLOAD TRENDS: ADVANCED ANALYTICS

KEYSTONE-ML TIMIT PIPELINE



KEYSTONE-ML TIMIT PIPELINE



PROPERTIES

Numerically Intensive

Iterative
(each iteration many jobs)

Long Running → Expensive

CLOUD COMPUTING CHOICES

t2.nano, t2.micro, t2.small
m4.large, m4.xlarge, m4.2xlarge,
m4.4xlarge, m3.medium,
c4.large, c4.xlarge, c4.2xlarge,
i2.2xlarge, i2.4xlarge, d2.xlarge
d2.2xlarge, d2.4xlarge,…

AMAZON EC2

Basic tier: A0, A1, A2, A3, A4
Optimized Compute : D1, D2,
D3, D4, D11, D12, D13
D1v2, D2v2, D3v2, D11v2,…
Compute Intensive: A10, A11,…

MICROSOFT AZURE

n1-standard-1, ns1-standard-2,
ns1-standard-4, ns1-standard-8,
ns1-standard-16, ns1highmem-2,
ns1-highmem-4, ns1-highmem-8,
highcpu-32, f1-micro, g1-small…

GOOGLE CLOUD ENGINE

Instance Types and Number of Instances

TYRANNY OF CHOICE

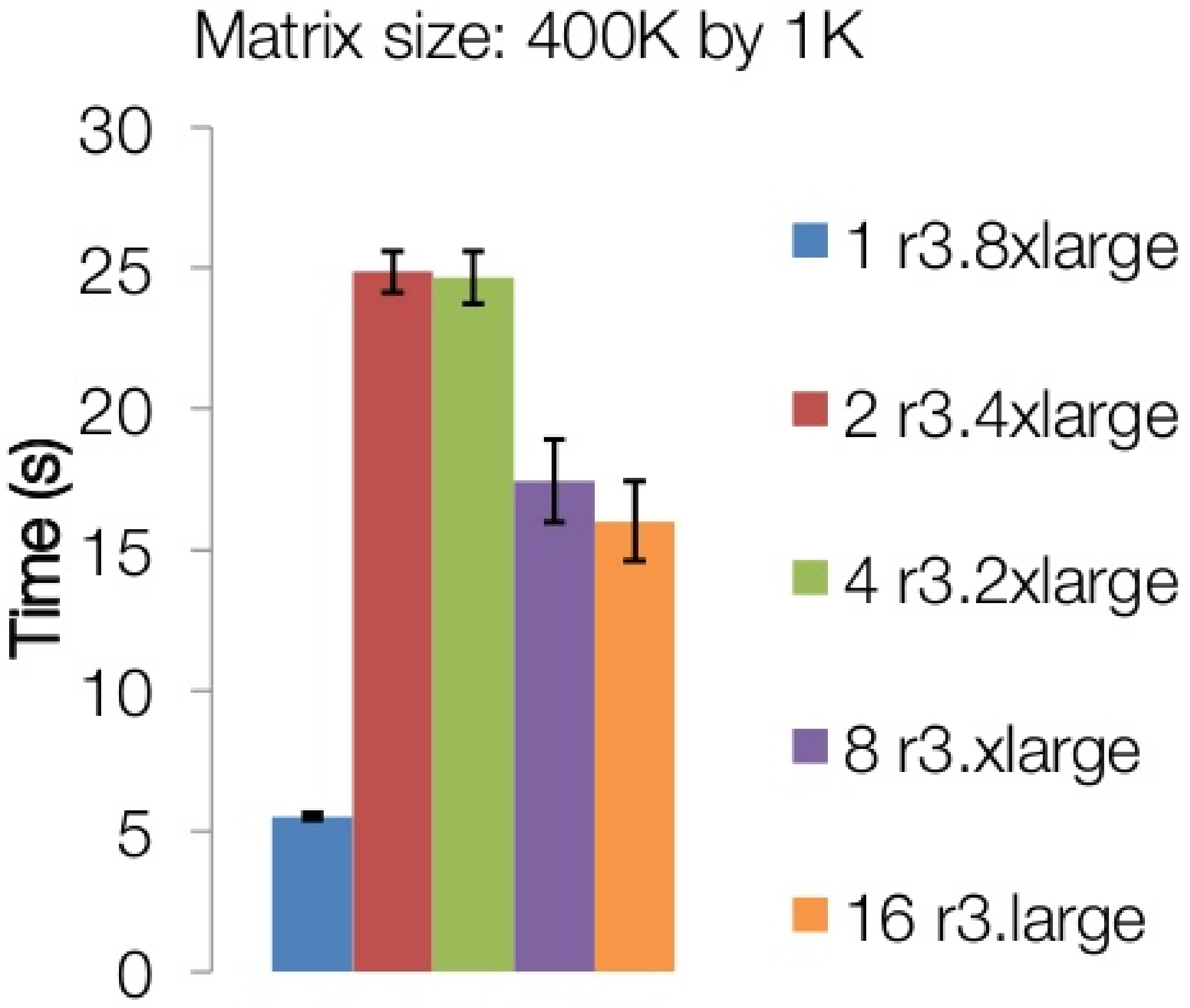
USER CONCERNS

"What is the cheapest configuration to run my job in 2 hours?"

Given a budget, how fast can I run my job ?

"What kind of instances should I use on EC2 ?"

DO CHOICES MATTER ? MATRIX MULTIPLY

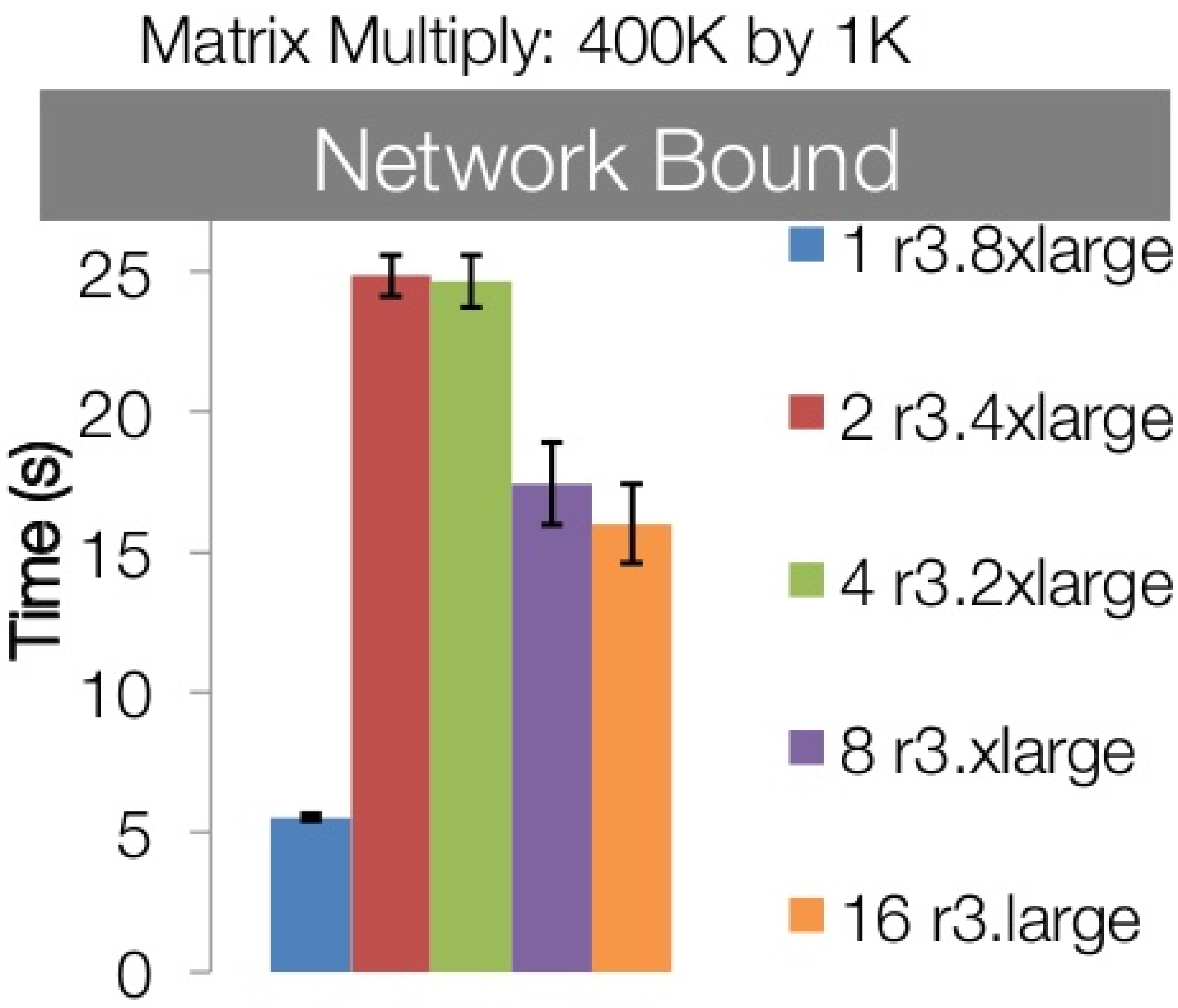


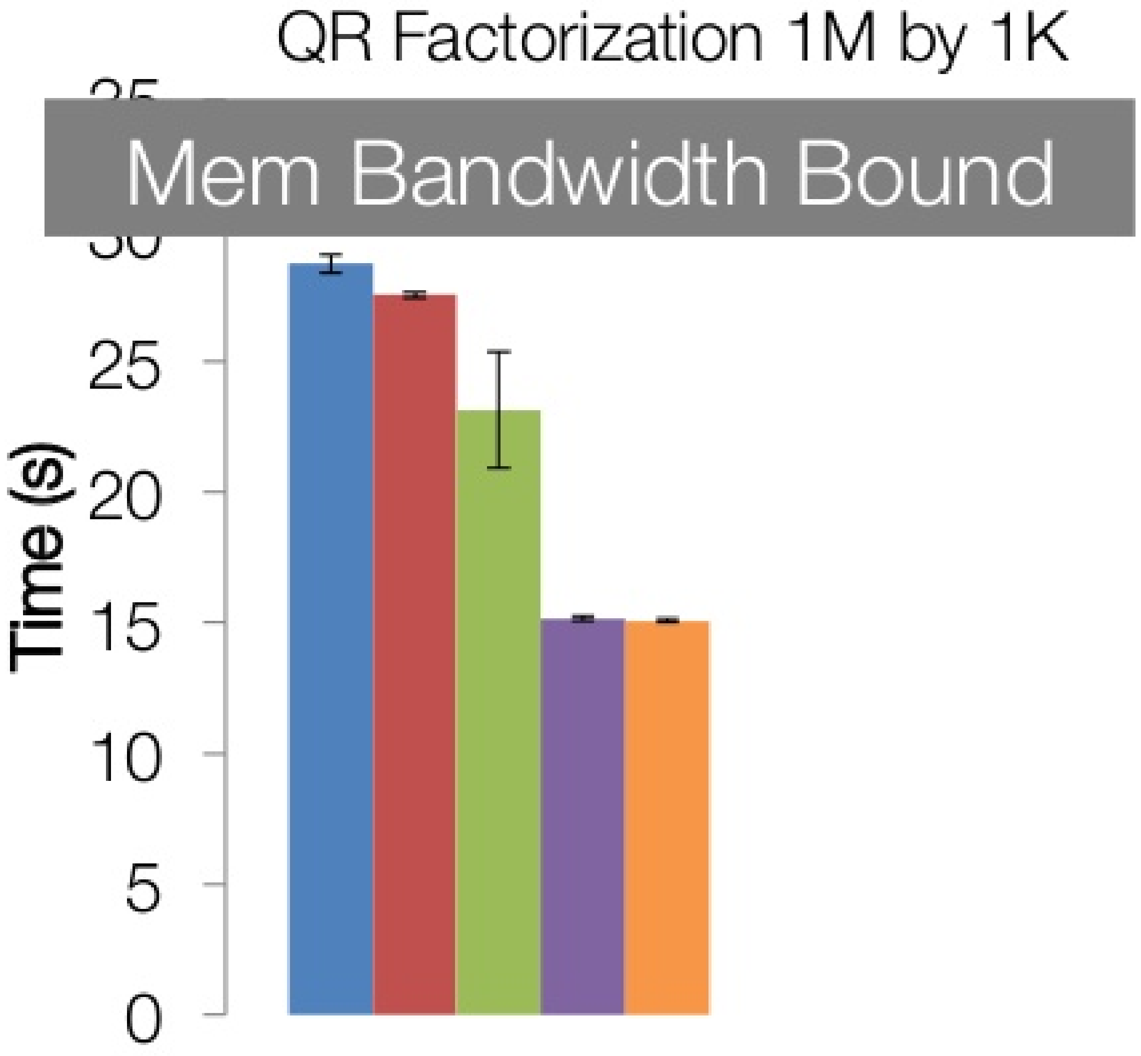
CORES = 16

MEMORY = 244 GB

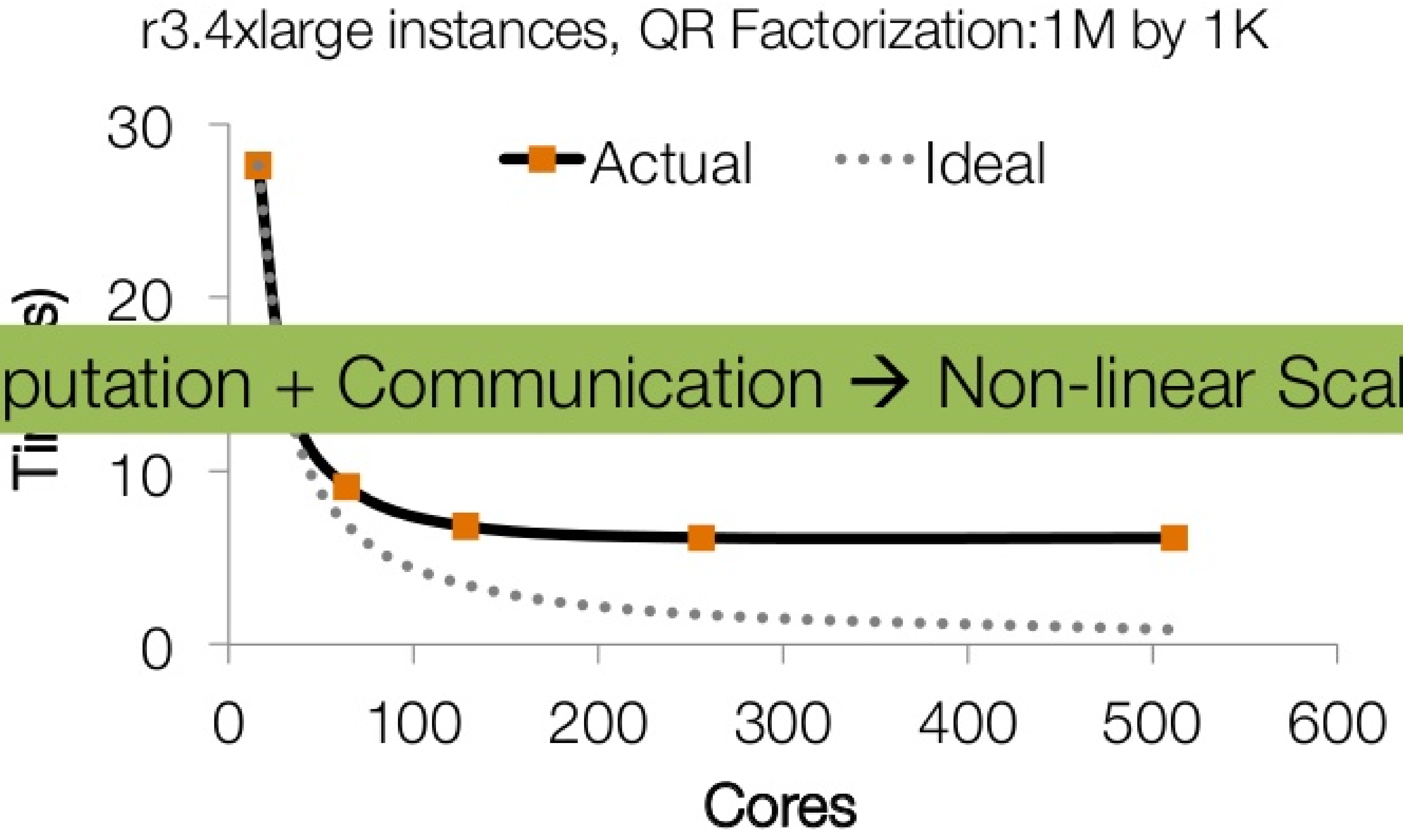
COST = $2.66/HR

DO CHOICES MATTER ?





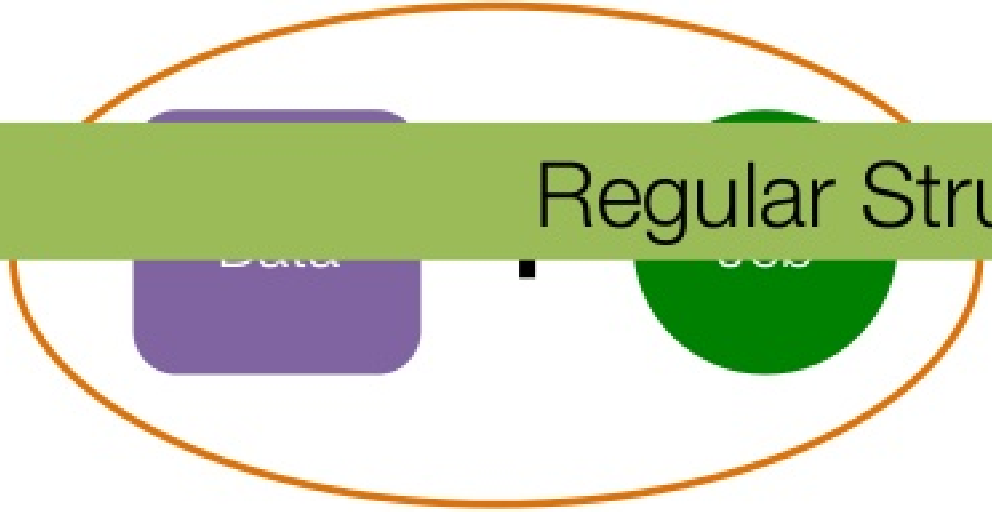
DO CHOICES MATTER ?



Computation + Communication → Non-linear Scaling

APPROACH

Performance Model



Regular Structure + Few Iterations

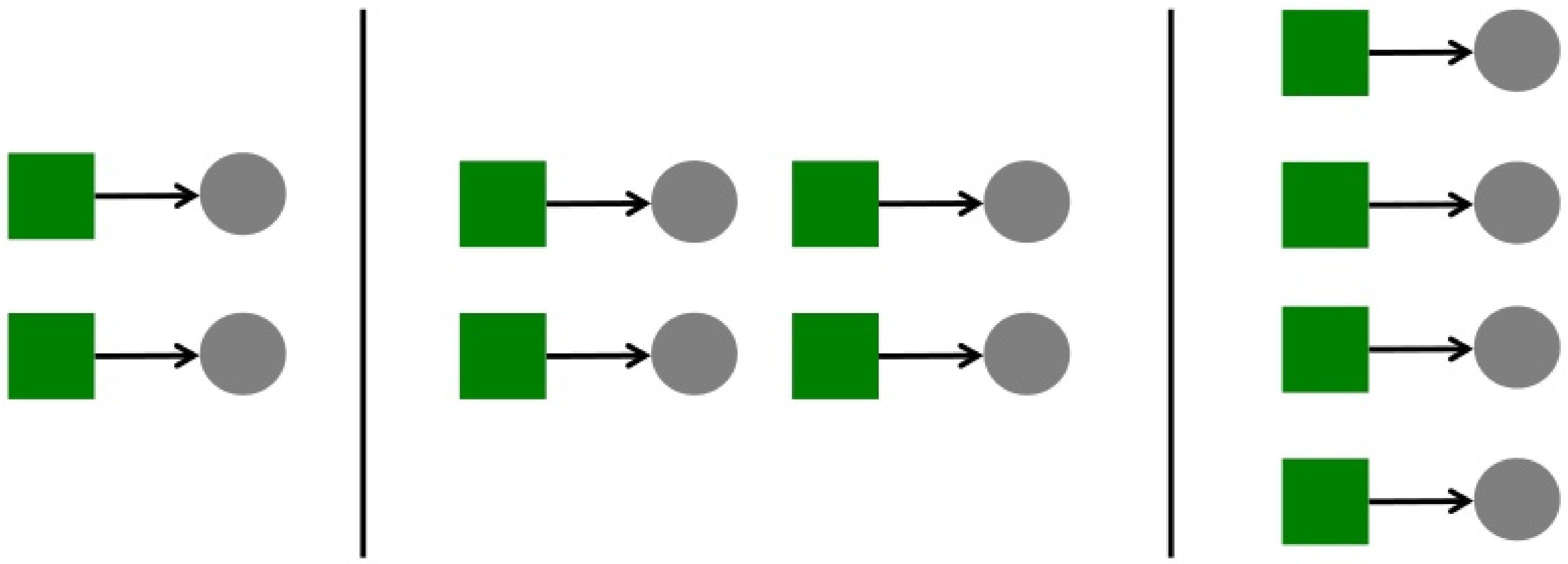
CHALLENGES

Black Box Jobs

Model Building Overhead

MODELING JOBS

COMPUTATION PATTERNS

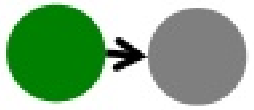


$\text{TIME} \propto \text{INPUT}$

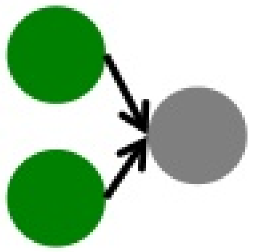
$\text{TIME} \propto \dfrac{1}{\text{MACHINES}}$

COMMUNICATION PATTERNS

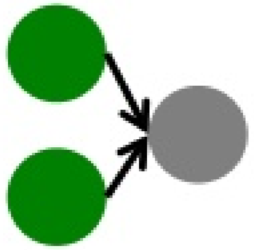
ONE-TO-ONE



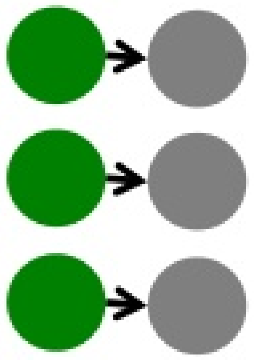
TREE DAG



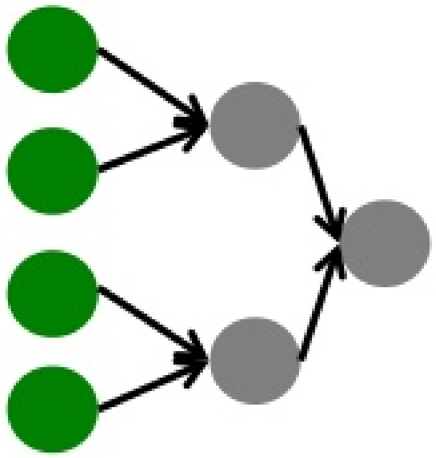
ALL-TO-ONE



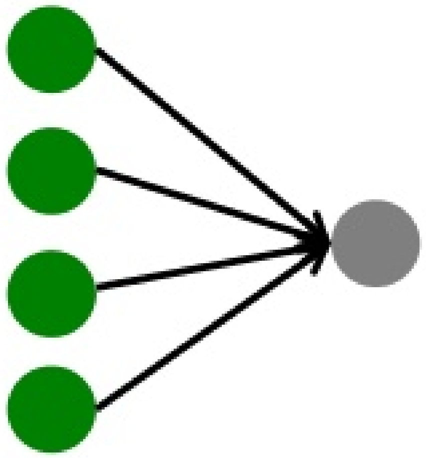
CONSTANT



LOG



LINEAR



BASIC MODEL

Computation (linear)

All-to-One DAG

$$time = x_1 + x_2 * \frac{input}{machines} + x_3 * \log(machines) + x_4 * (machines)$$

Serial Execution

Tree DAG

Collect Training Data

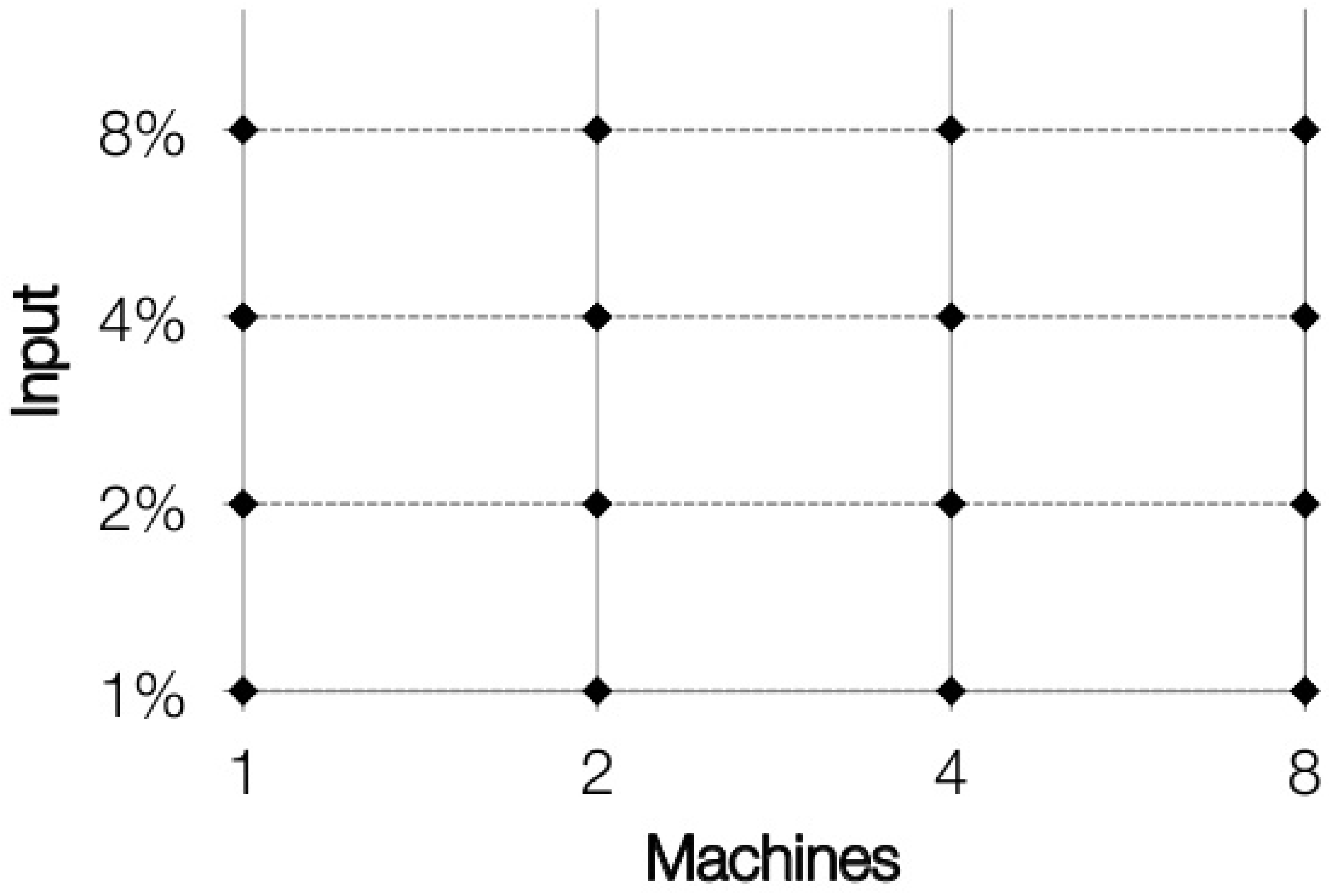
Fit Linear Regression

DOES THE MODEL MATCH REAL APPLICATIONS ?

Number of data items: n, Number of processors: p, Constant number of features

Algorithm	Compute		Communication			
	$O(n/p)$	$O(n^2/p)$	$O(1)$	$O(p)$	$O(\log(p))$	Other
GLM Regression	✔			✔	✔	
KMeans	✔		✔	✔		O(center)
Naïve Bayes	✔			✔		
Pearson Correlation	✔				✔	
PCA	✔			✔	✔	
QR (TSQR)	✔				✔	

COLLECTING TRAINING DATA



Grid of
input, machines

Associate cost with
each experiment

Baseline: Cheapest
configurations first

OPTIMAL DESIGN OF EXPERIMENTS

Given a Linear Model

$$y_i = a_i^T x + w_i, \quad i = 1, \ldots, m,$$

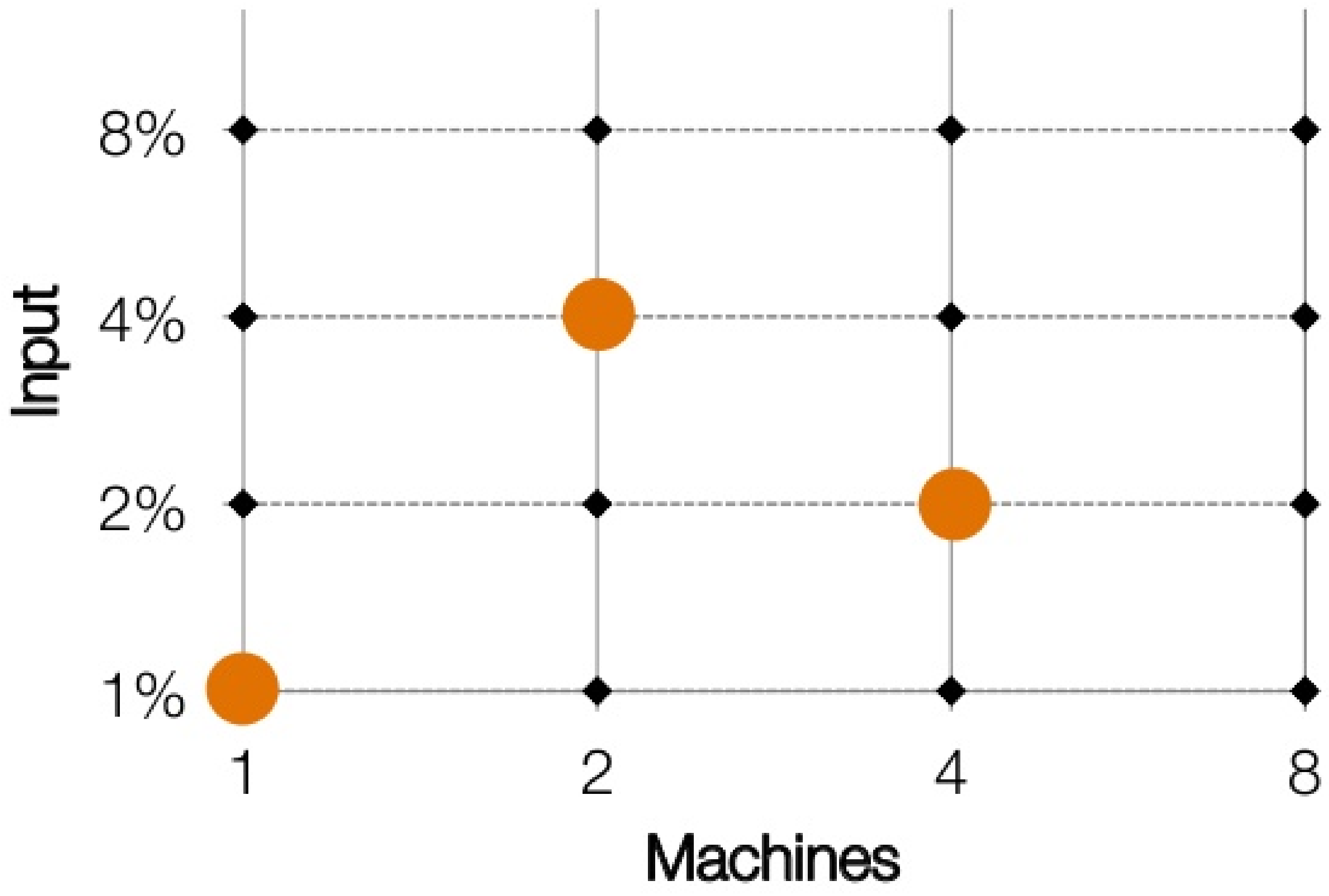
λ_i – Fraction of times each experiment is run

$$\begin{array}{ll} \text{Minimize} & \mathbf{tr}\left(\left(\sum_{i=1}^{m} \lambda_i a_i a_i^T\right)^{-1}\right) \\ \text{subject to} & \lambda_i \geq 0, \lambda_i \leq 1 \\ & \sum_{i=1}^{m} c_i \lambda_i \leq B \end{array}$$

Lower variance → Better model

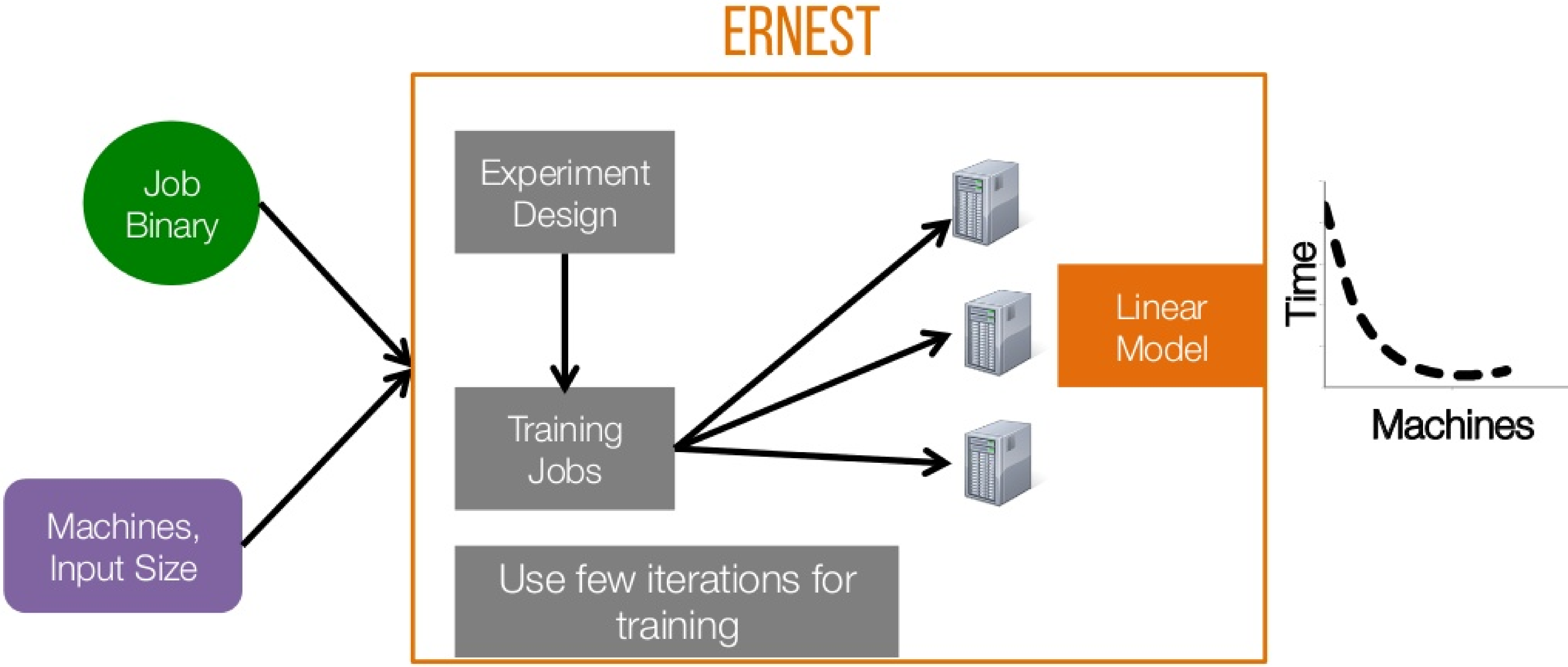
Bound total cost

OPTIMAL DESIGN OF EXPERIMENTS



Use off-the-shelf solver (CVX)

USING ERNEST



EVALUATION

OBJECTIVES

Optimal number of machines
Prediction accuracy
Model training overhead
Importance of experiment design
Choosing EC2 instance types
Model extensions

WORKLOADS

Keystone-ML
Spark MLlib
ADAM
GenBase
Sparse GLMs
Random Projections

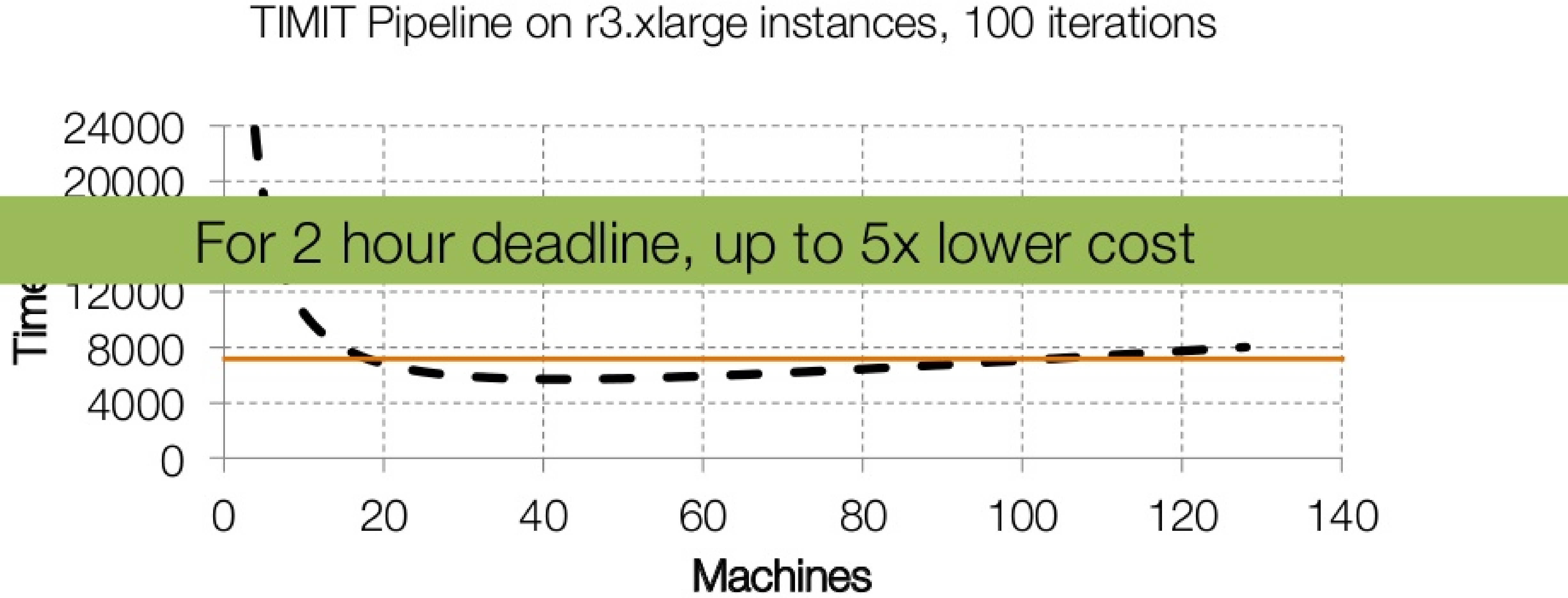
OBJECTIVES

Optimal number of machines
Prediction accuracy
Model training overhead
Importance of experiment design
Choosing EC2 instance types
Model extensions

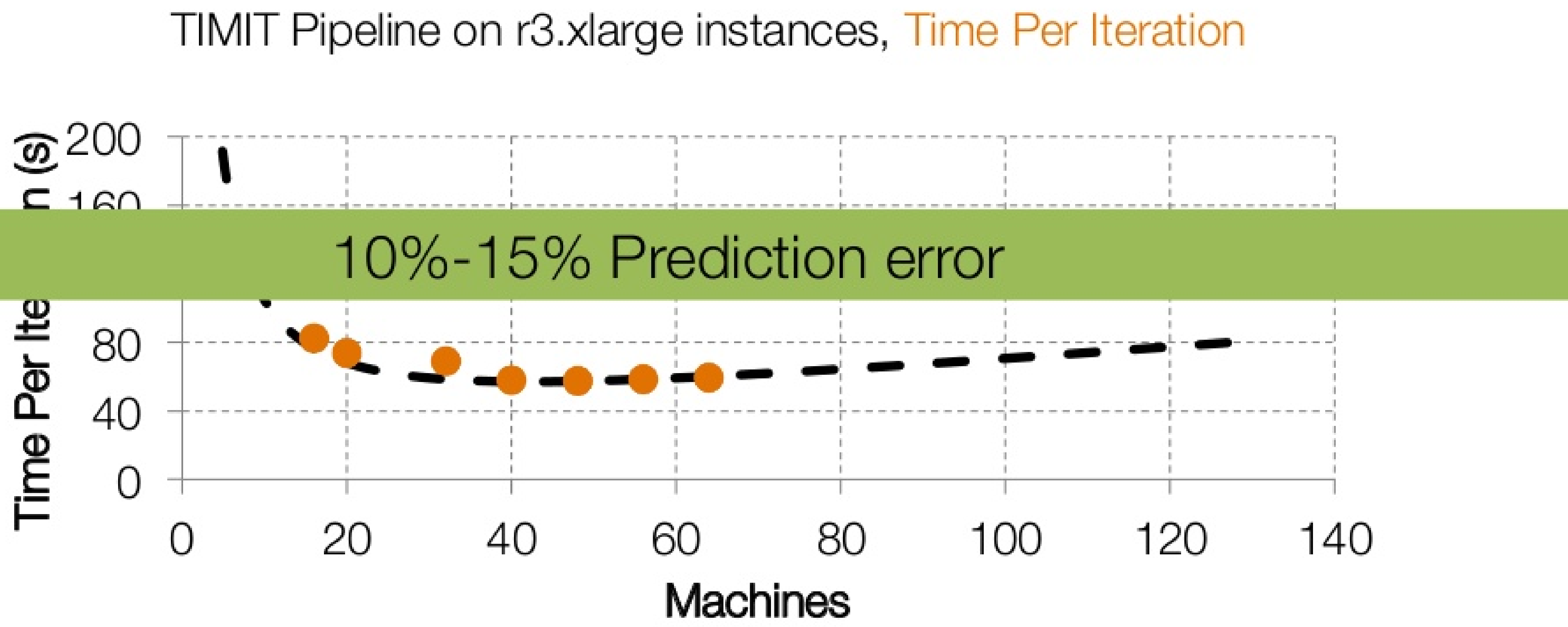
WORKLOADS

Keystone-ML
Spark MLlib
ADAM
GenBase
Sparse GLMs
Random Projections

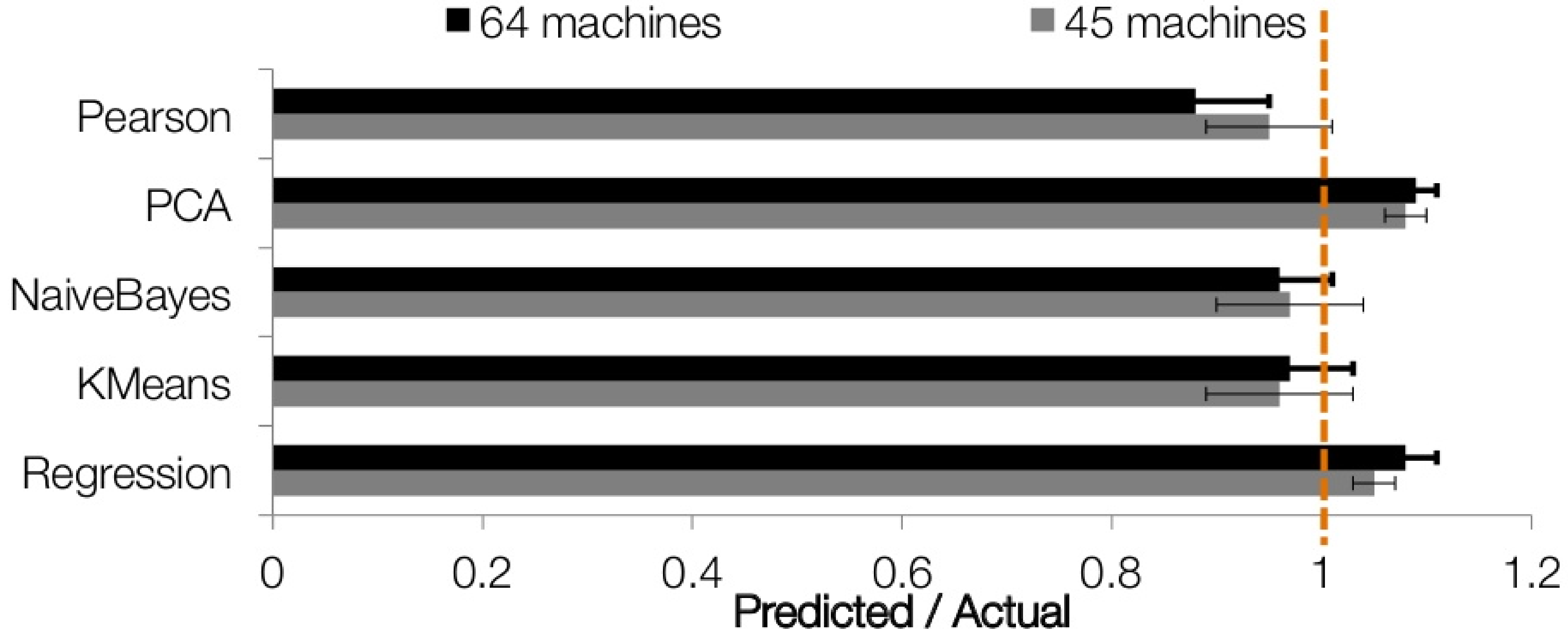
NUMBER OF INSTANCES: KEYSTONE-ML



ACCURACY: KEYSTONE-ML



ACCURACY: SPARK MLLIB

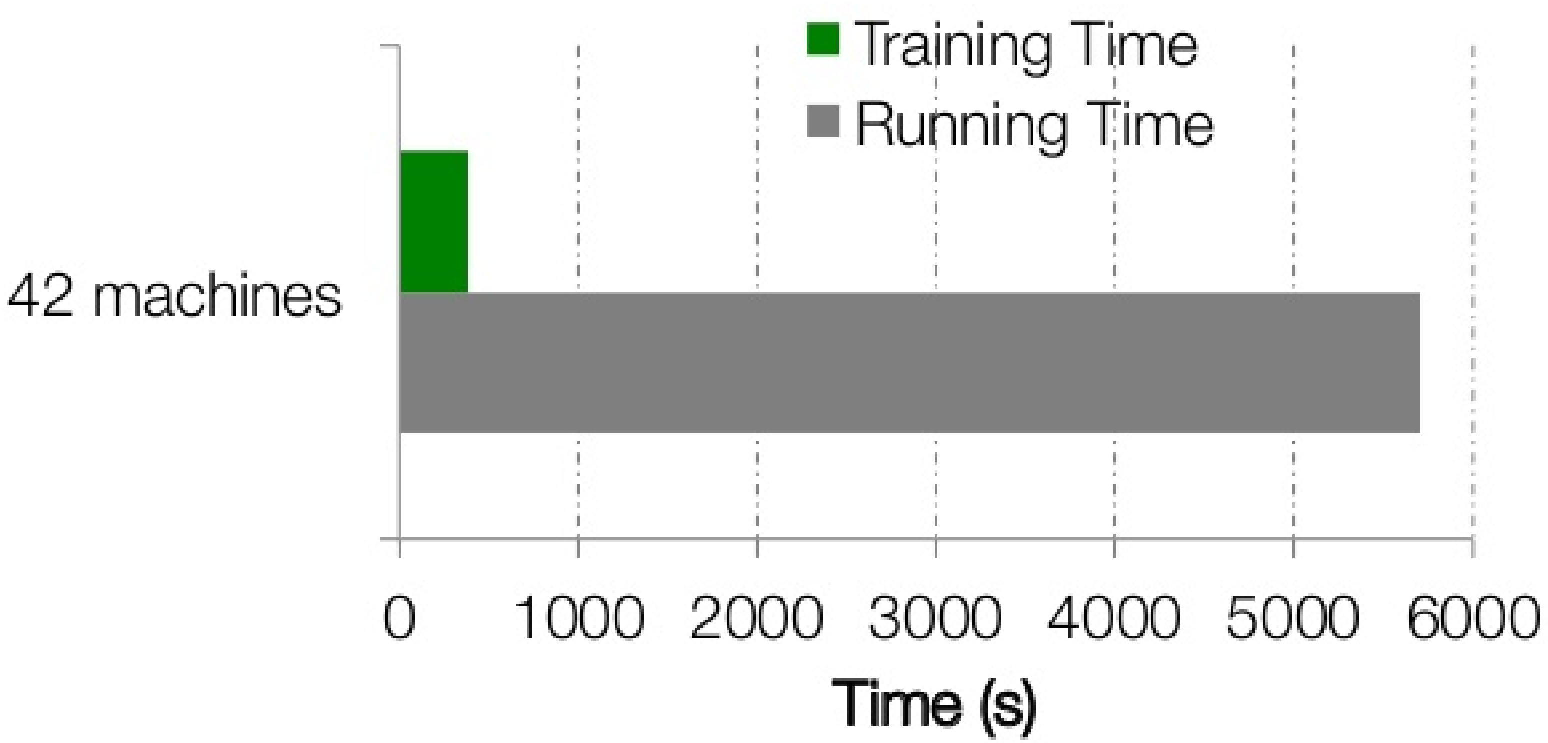


TRAINING TIME: KEYSTONE-ML

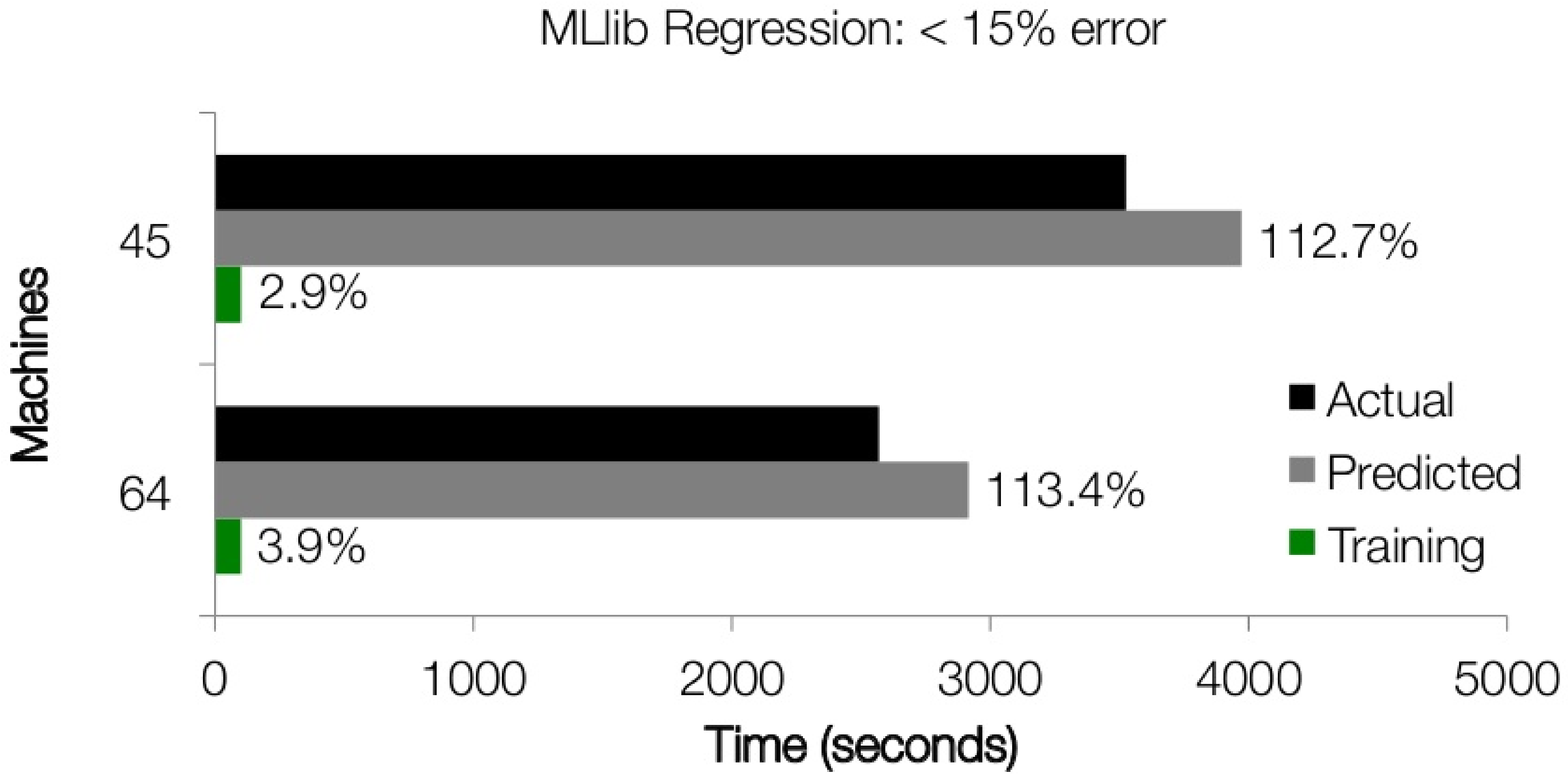
TIMIT Pipeline on r3.xlarge instances, 100 iterations

EXPERIMENT DESIGN

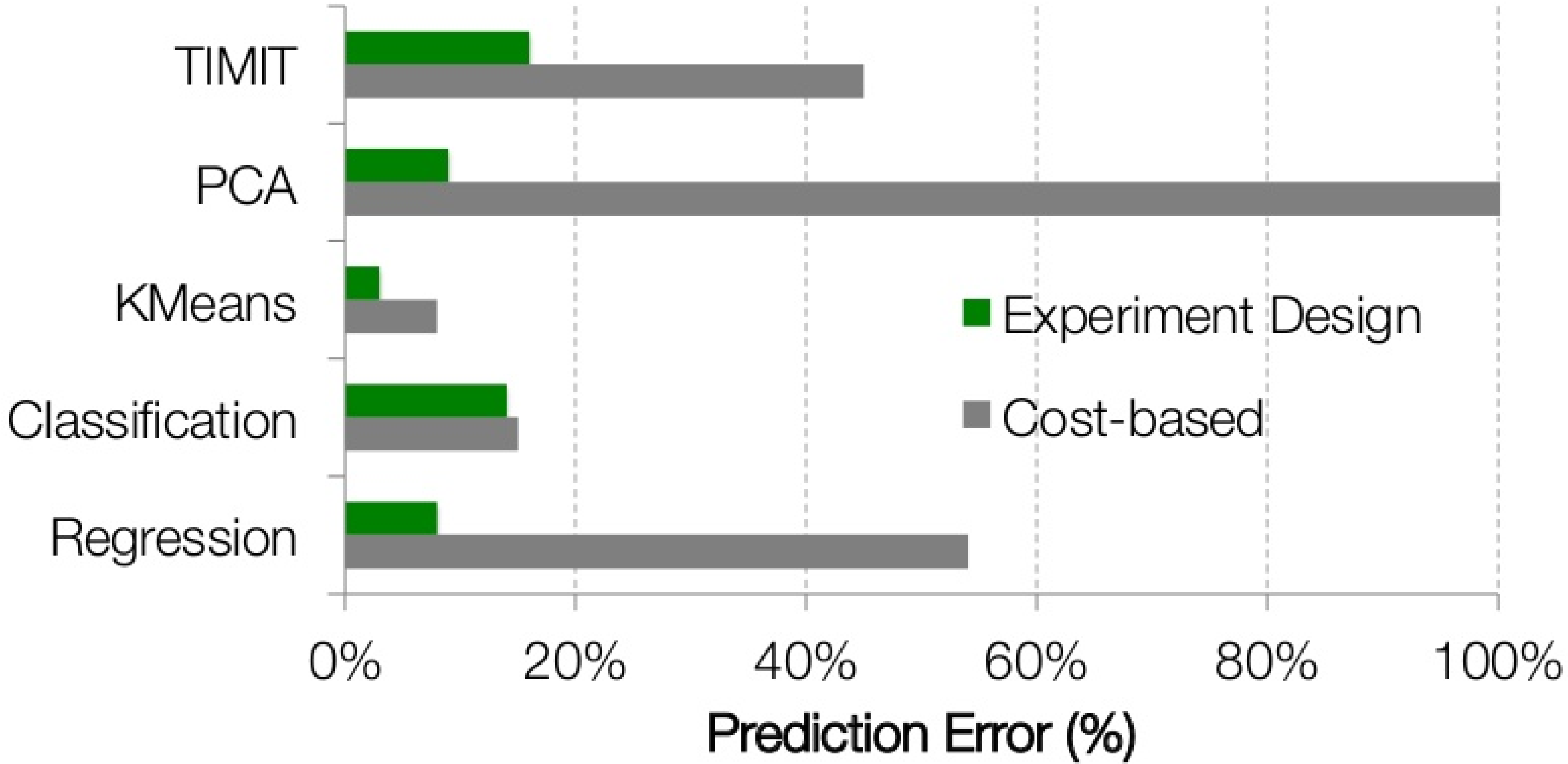
7 data points
Up to 16 machines
Up to 10% data



ACCURACY, OVERHEAD: GLM REGRESSION



IS EXPERIMENT DESIGN USEFUL ?



MORE DETAILS

Detecting when the model is wrong

Model extensions

Amazon EC2 variations over time

Straggler mitigation strategies

http://shivaram.org/publications/ernest-nsdi.pdf

Python-based implementation

Experiment Design, Predictor Modules

Example using SparkML + RCV1

OPEN SOURCE

https://github.com/amplab/ernest

IN CONCLUSION

Workload Trends: Advanced Analytics in the Cloud

Computation, Communication patterns affect scalability

Ernest: Performance predictions with low overhead

- End-to-end linear model
- Optimal experimental design

Workloads / Traces ?

Shivaram Venkataraman shivaram@cs.berkeley.edu