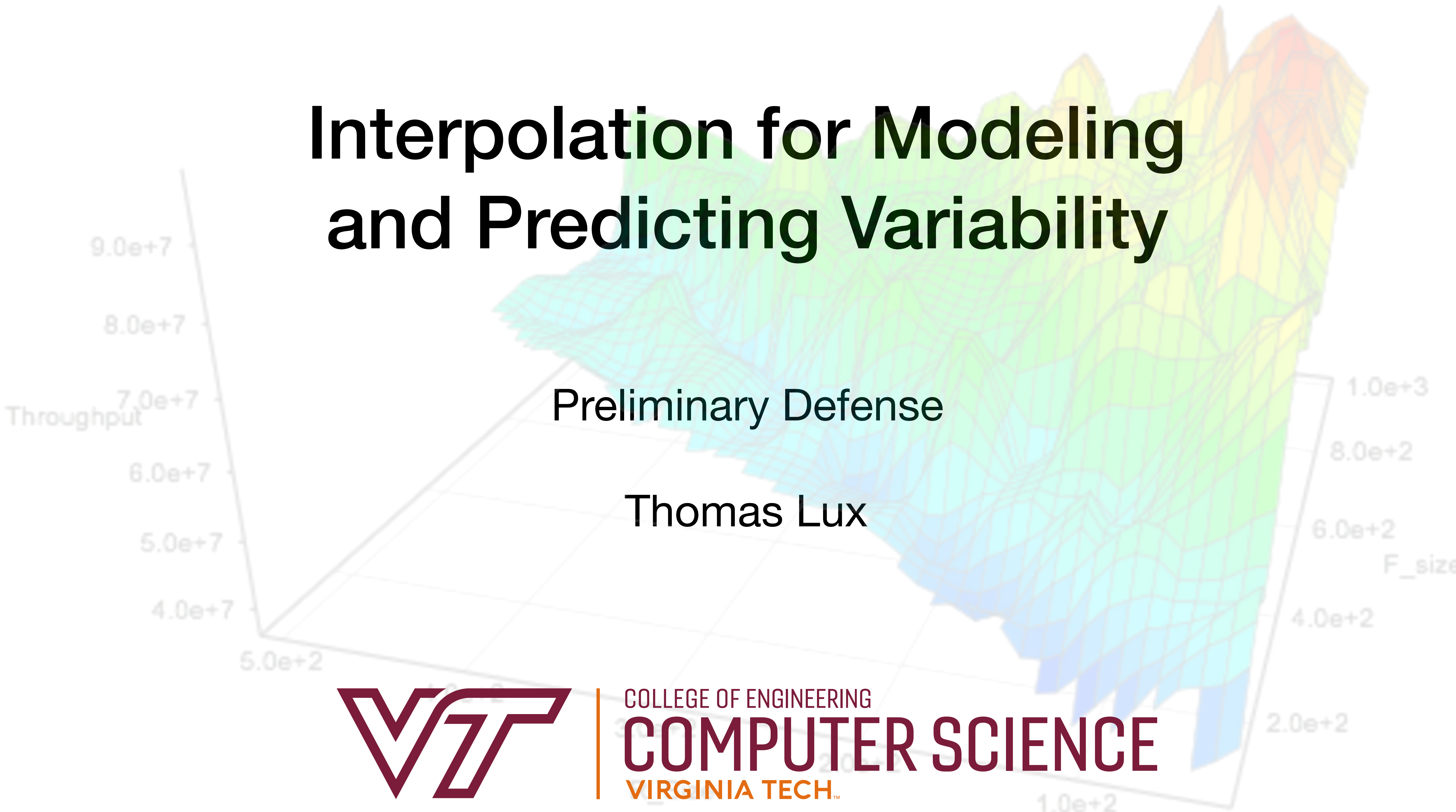
Interpolation for Modeling and Predicting Variability



Preliminary Defense

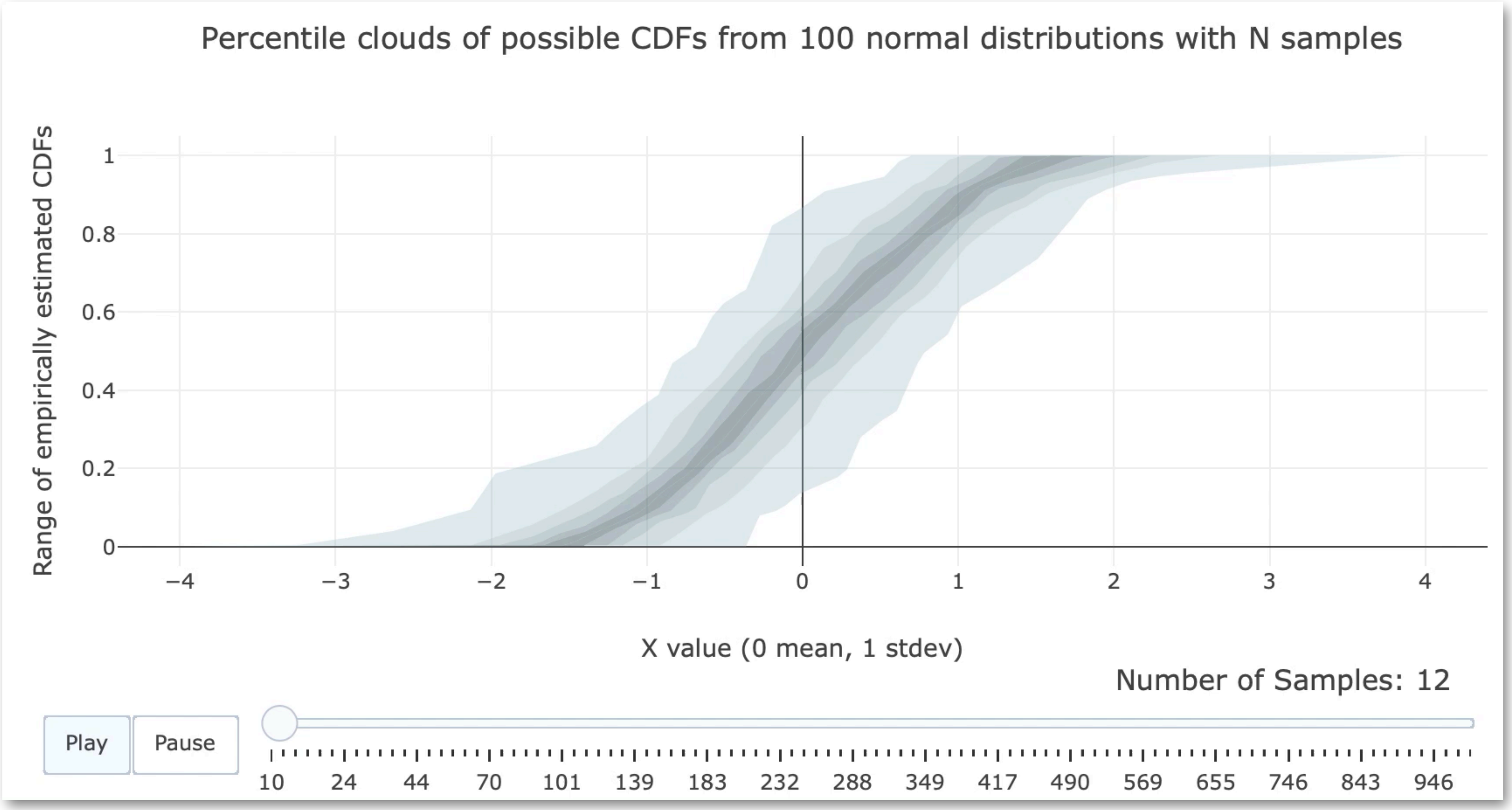
Thomas Lux

COLLEGE OF ENGINEERING
COMPUTER SCIENCE
VIRGINIA TECH

Chapters

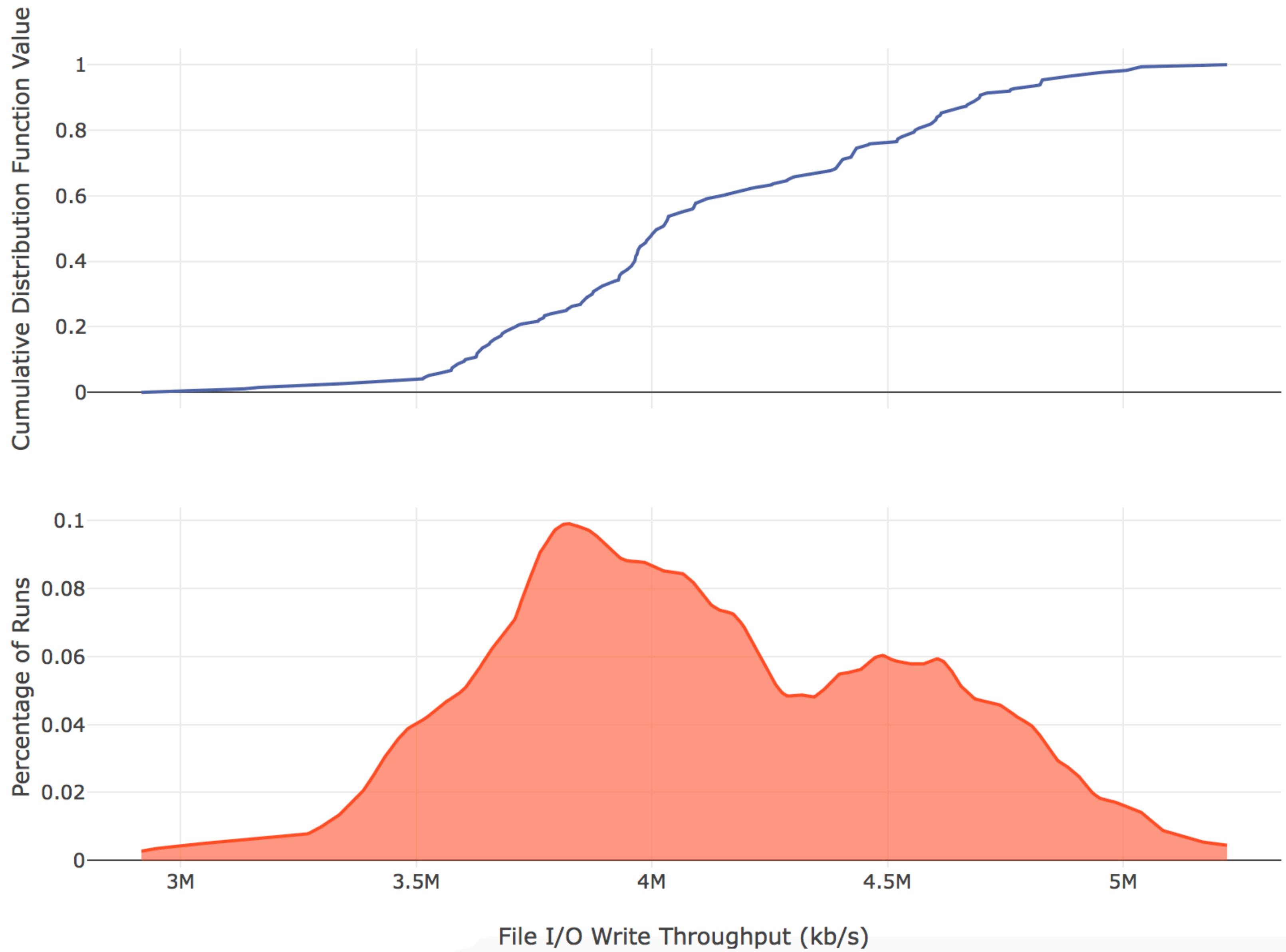
What is “Variability”?

A *random variable* uniquely defined by its Cumulative Distribution Function (CDF) / Probability Density Function (PDF)



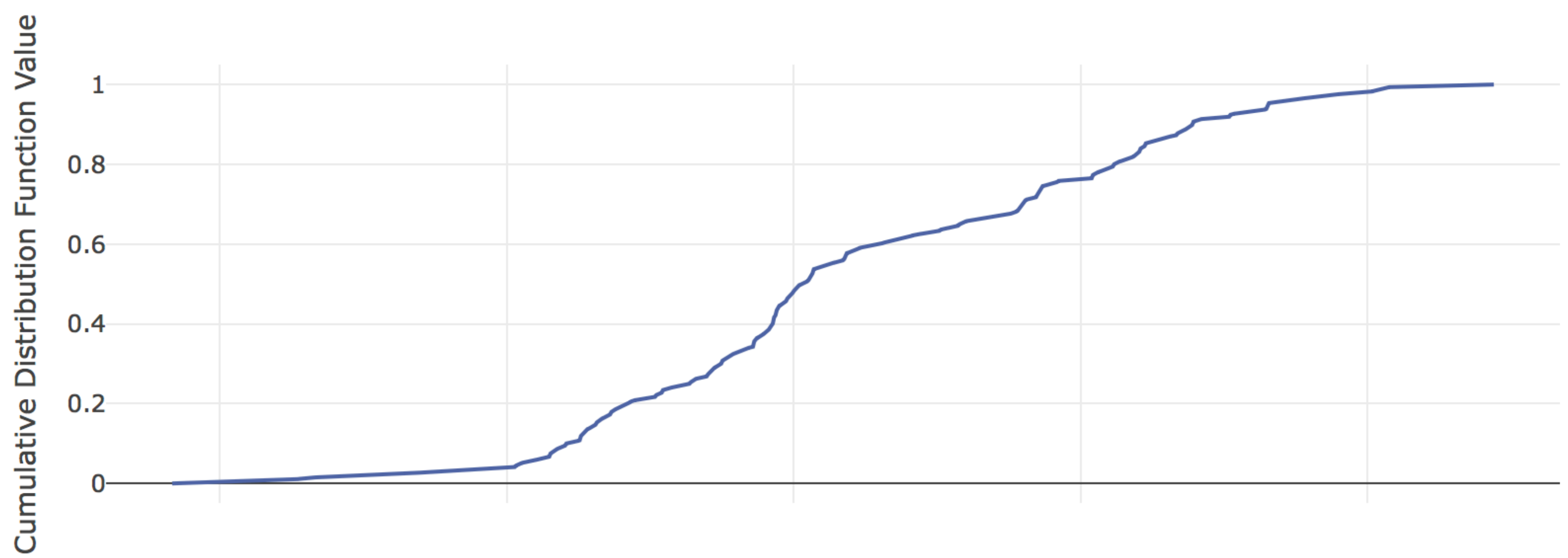
Variability in Systems

When the same computer is used to execute the same program repeatedly, most measurable performance characteristics will vary.



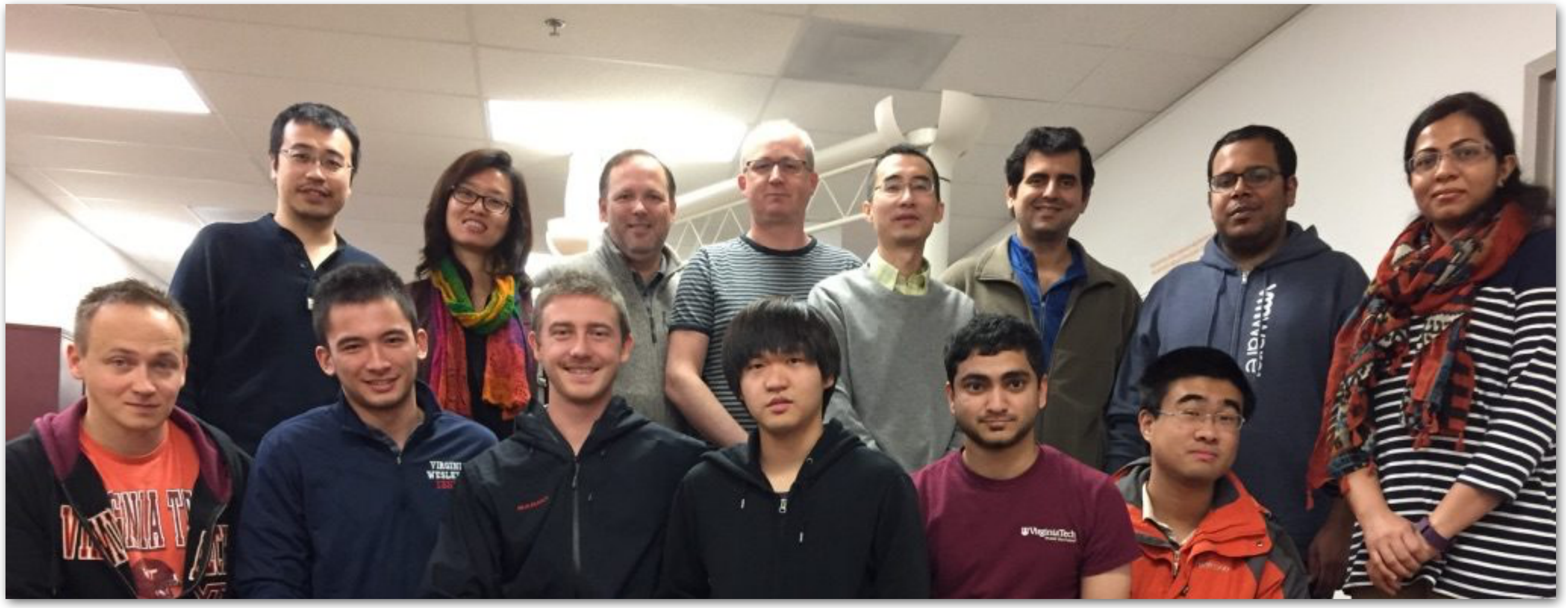
Variability in Systems

When the same computer is used to execute the same program repeatedly, most measurable performance characteristics will vary.



Managing this performance variability can be very important in applications where we want to meet some set requirements.

VarSys: Modeling and Managing Variability



High Performance Computing
HPC systems consume a lot of energy and time, both are functions of how the system was built and configured. Models can be used to optimize a configuration.

Cloud Computing
Small savings in compute time and performance magnify greatly when 1000's of machines are involved. Service Level Agreements (SLAs) can be tightened.

Computer Security
A strong understanding of variability can improve defenses against malicious users by demonstrating new vulnerabilities, and helping prevent side channel attacks.

Variability is important in many aspects of computation.

Quantifying variability and constructing models of it may lead to improvements in all of these aspects of computation.

Chapters

1. The Importance and Applications of Variability
2. Algorithms for Constructing Approximations
3. Naive Approximations of Variability
4. Box-Splines: Uses, Constructions, and Applications
5. Stronger Approximations of Variability
6. An Error Bound for Piecewise Linear Interpolation
7. Improving Variability estimates

COMPUTER SCIENCE
VIRGINIA TECH.

Approximation: The Problem Description

Given

underlying function $f : \mathbb{R}^d \to \mathbb{R}$

data matrix $X^{d \times n}$ with column vectors $x^{(i)} \in \mathbb{R}^d$

function values $f(x^{(i)})$ for all $x^{(i)}$

vector $f(X)$ has elements $f(x^{(i)})$

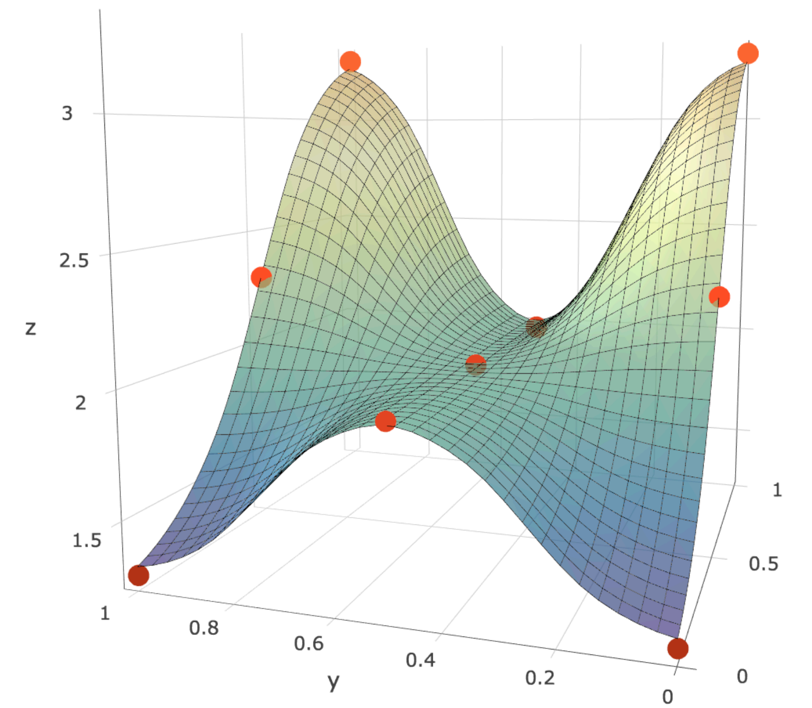
Generate a function $\hat{f} : \mathbb{R}^d \to \mathbb{R}$ such that:

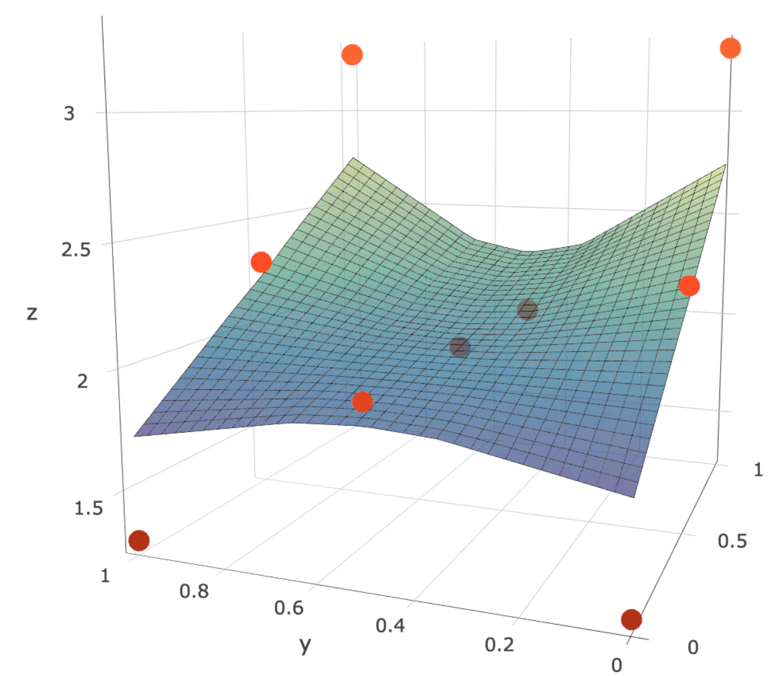
Interpolation

$\hat{f}(x^{(i)})$ equals $f(x^{(i)})$ for all $x^{(i)}$

Regression

$\hat{f}_p$ has parameters p and is the solution to $\min_p \| \hat{f}(X) - f(X) \|$





COMPUTER SCIENCE
VIRGINIA TECH.

Regression Techniques

Multivariate Adaptive Regression Splines

$$B_{2j-1}(x) = B_l(x)\left(x_i - x_i^{(p)}\right)_+$$

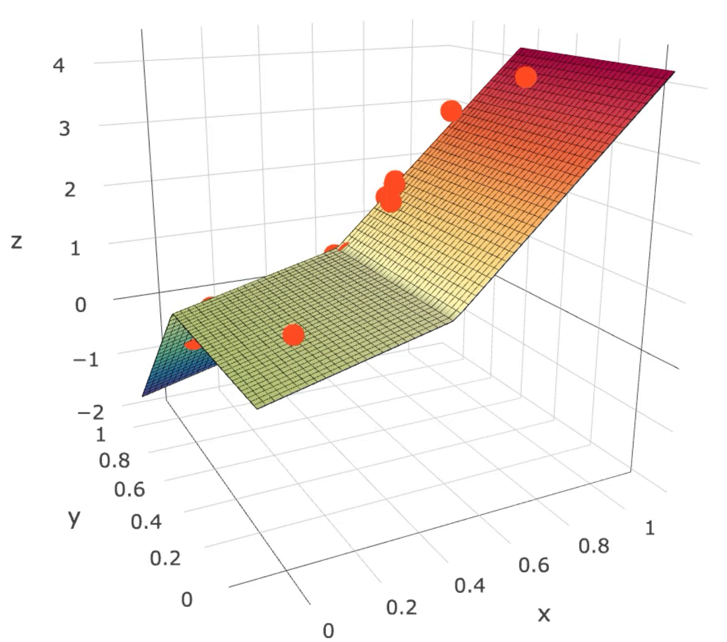
$$B_{2j}(x) = B_k(x)\left(x_i - x_i^{(p)}\right)_-$$

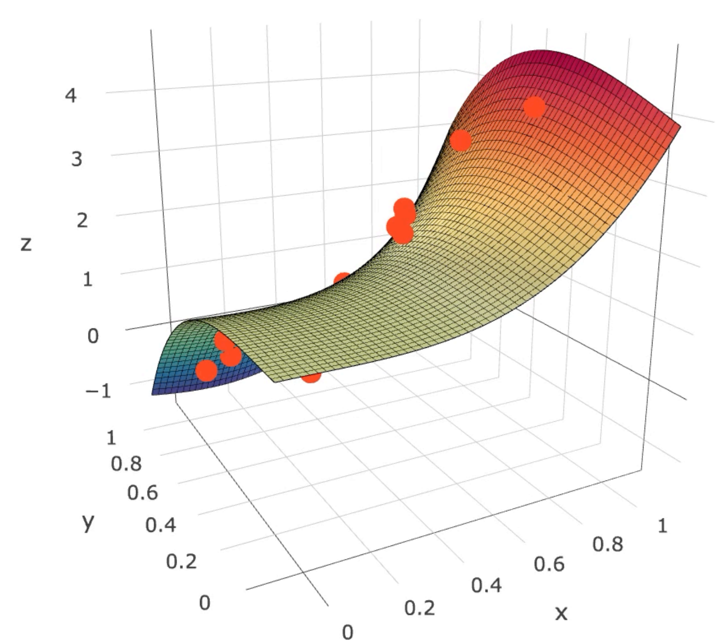
Support Vector Regressor

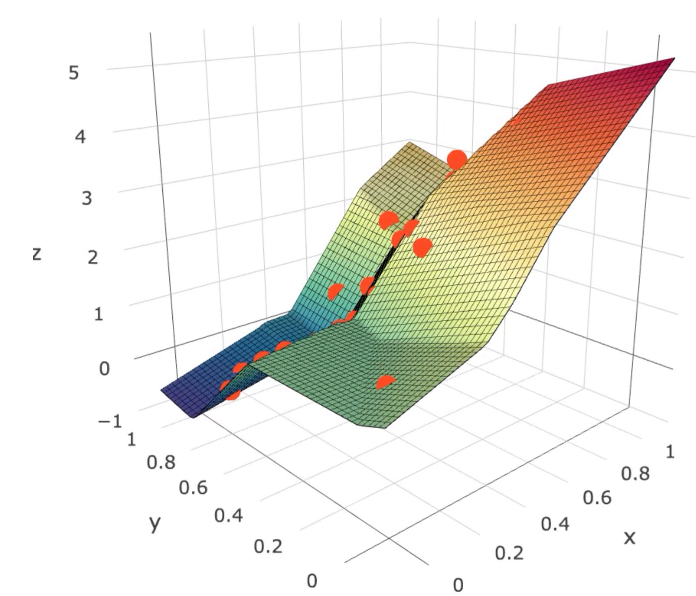
$$p(x) = \sum_{i=1}^{n} a_i K\left(x, x^{(i)}\right) + b$$

Multilayer Perceptron Regressor

$$l(u) = \left(u^t W_l + c_l\right)_+$$







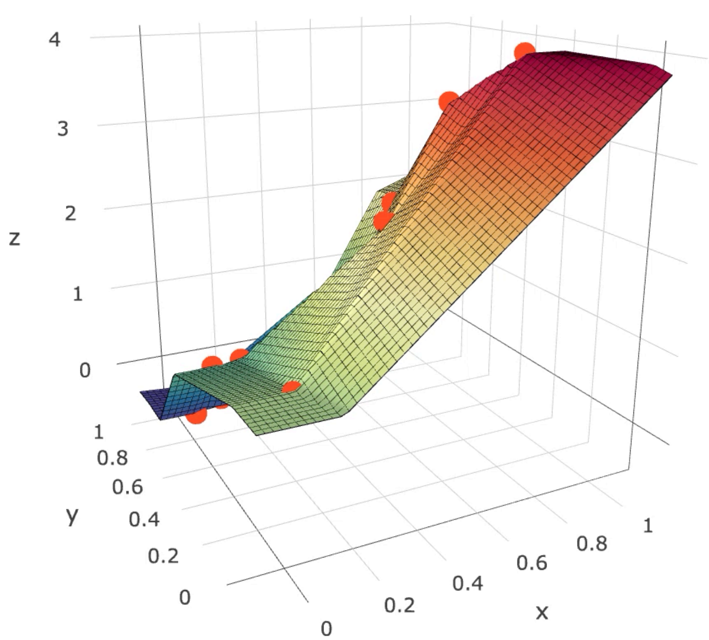
Interpolation Techniques

Delaunay

simplicial mesh

$$y=\sum_{i=0}^{d} w_i x^{(i)}, \quad \sum_{i=0}^{d} w_i = 1, \quad w_i \geq 0, \quad i=0,\ldots,d$$

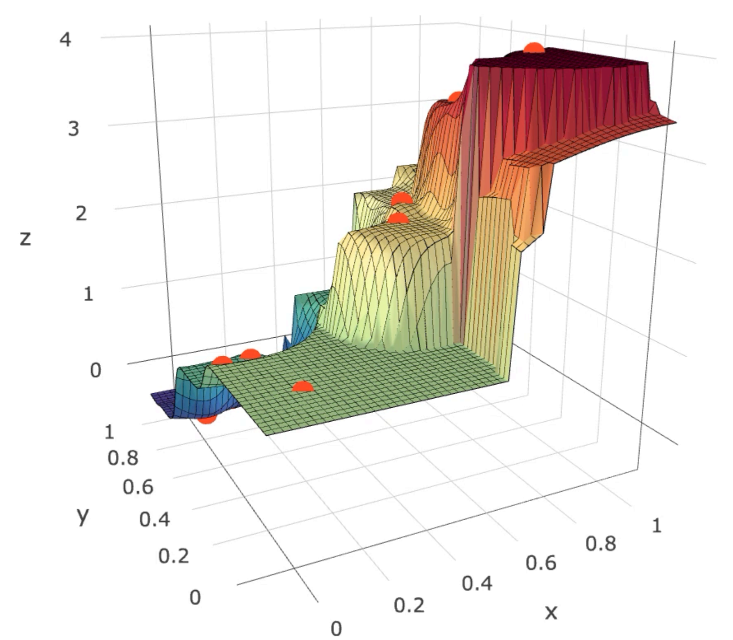
$$\hat{f}(y)=\sum_{i=0}^{d} w_i f\left(x^{(i)}\right)$$



Modified Shepard

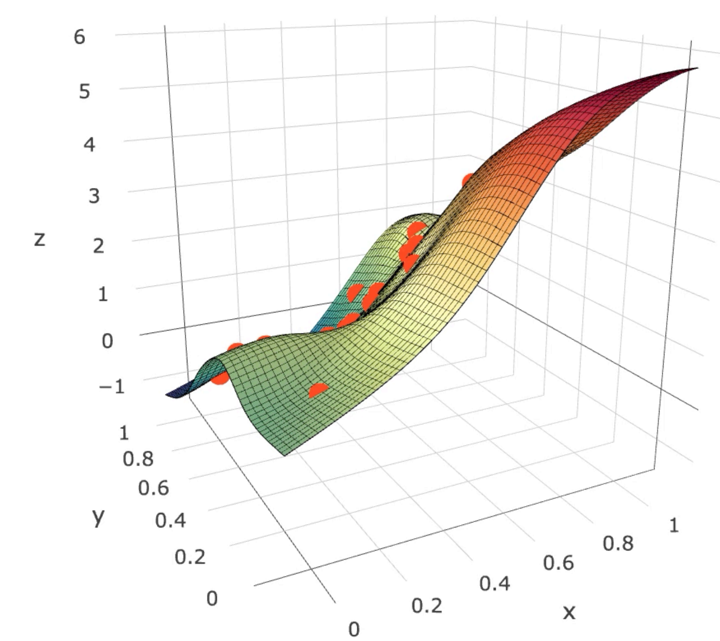
squared inverse distance

$$p(x)=\frac{\sum_{k=1}^{n} W_k(x) f\left(x^{(k)}\right)}{\sum_{k=1}^{n} W_k(x)}$$



Linear Shepard

local linear fit



COMPUTER SCIENCE
VIRGINIA TECH

Chapters

COMPUTER SCIENCE
VIRGINIA TECH.

IOzone — A System Benchmark

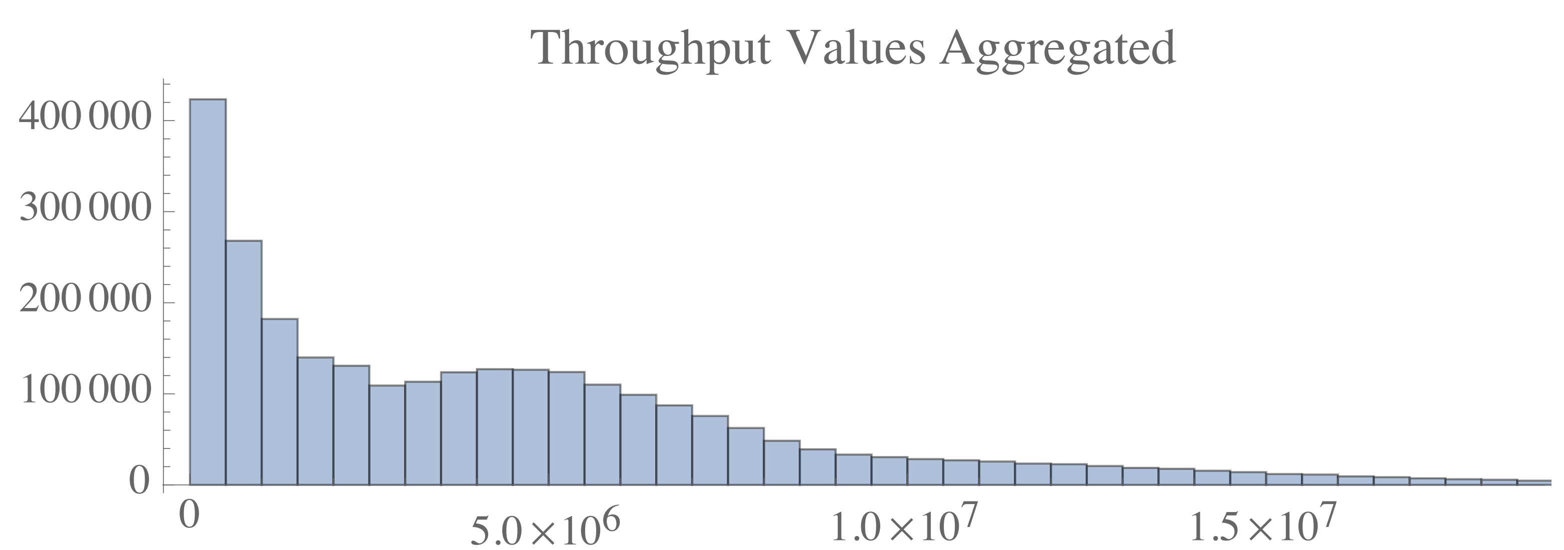
System Specs:

Two Intel Xeon E5-2637 CPUs with total 16 CPU cores and 16GB DRAM per node, at 12 nodes.

Ext4 filesystem above an Intel SSDSC2BA20 SSD drive.

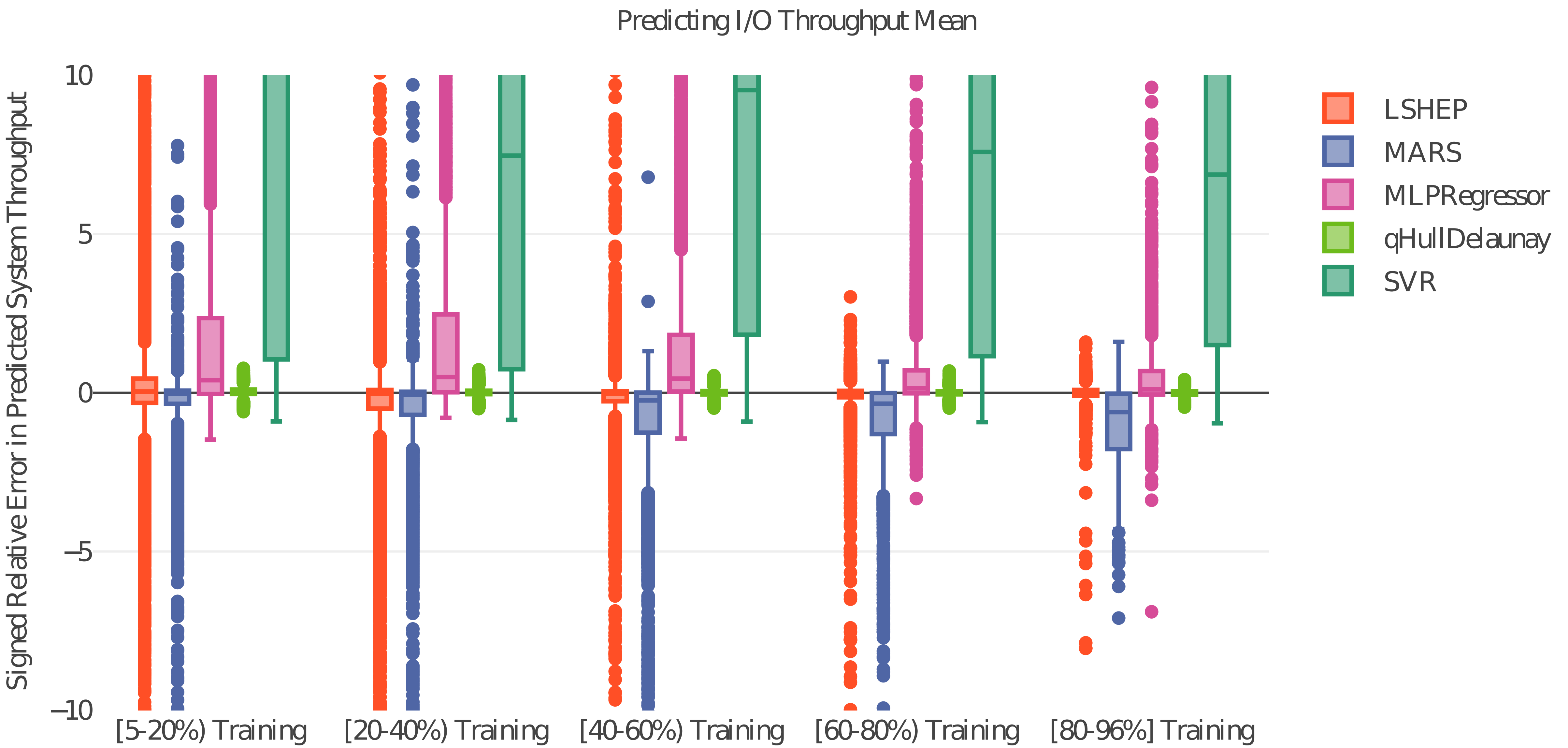
Each of ~20K unique system configurations were run 150 times.

System Parameter	**Values**
File Size (KB)	4, 16, 64, 256, 1024, 4096, 8192, 16384
Record Size (KB)	4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384
Thread Count	1, 8, 16, 24, 32, 40, 48, 56, 64
Frequency (GHz)	1.2, 1.6, 2, 2.3, 2.8, 3.2, 3.5
Test Type	Readers, Rereaders, Random Readers, Initial Writers, Rewriters, Random Writers

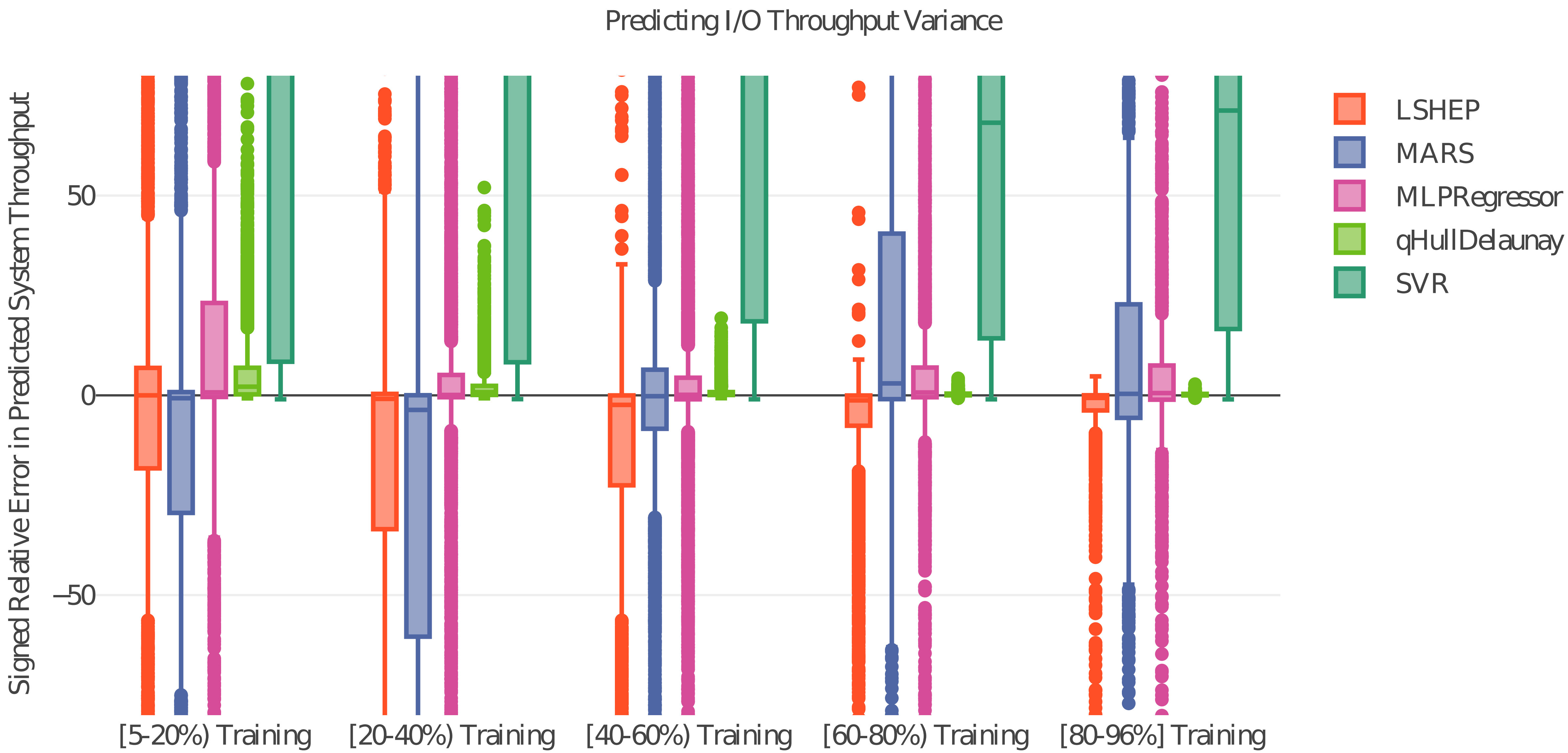


COMPUTER SCIENCE
VIRGINIA TECH

Mean Prediction Results



Variance Prediction Results



Chapter Takeaways

Multivariate models of HPC system performance can predict I/O throughput mean and variance.

The Delaunay method produces considerably better results for mean and variance prediction.

Throughput variance is harder to predict than mean throughput.

Chapters

1. The Importance and Applications of Variability
2. Algorithms for Constructing Approximations
3. Naive Approximations of Variability
4. Box-Splines: Uses, Constructions, and Applications
5. Stronger Approximations of Variability
6. An Error Bound for Piecewise Linear Interpolation
7. Improving Variability estimates

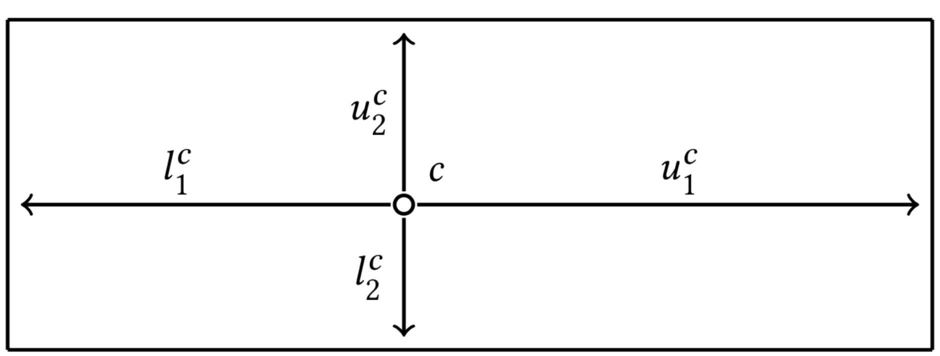
COMPUTER SCIENCE
VIRGINIA TECH.

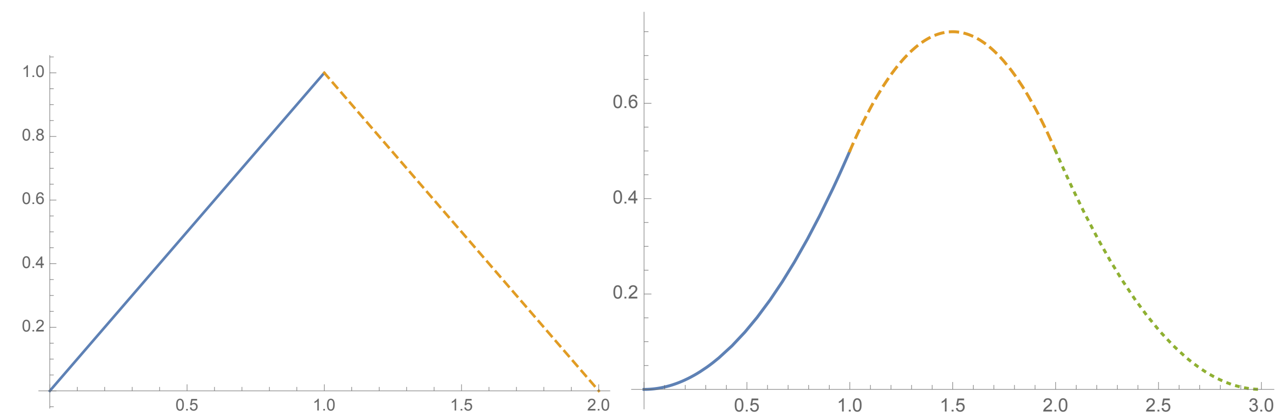
Box Splines

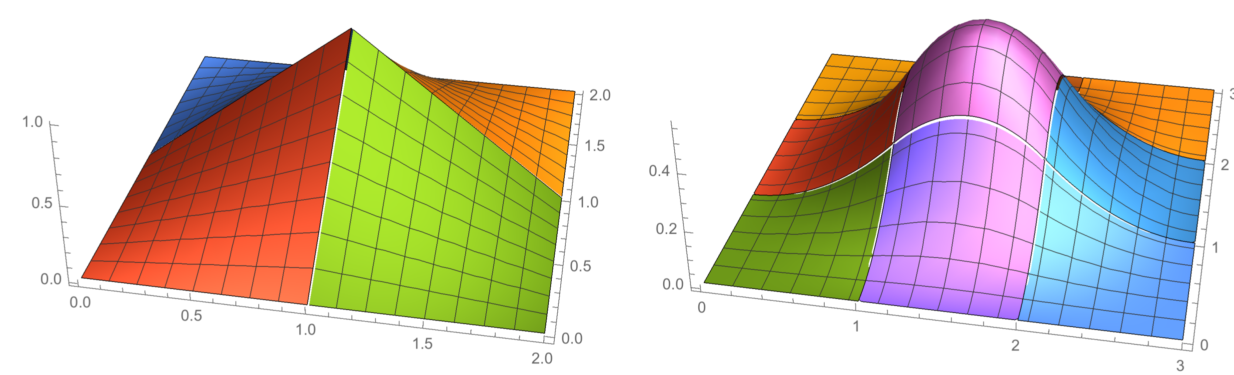
Proposed by C. de Boor as an extension of B-Splines into multiple dimensions (without using tensor products).

Can be shifted and scaled without losing smoothness.

Computationally expensive.

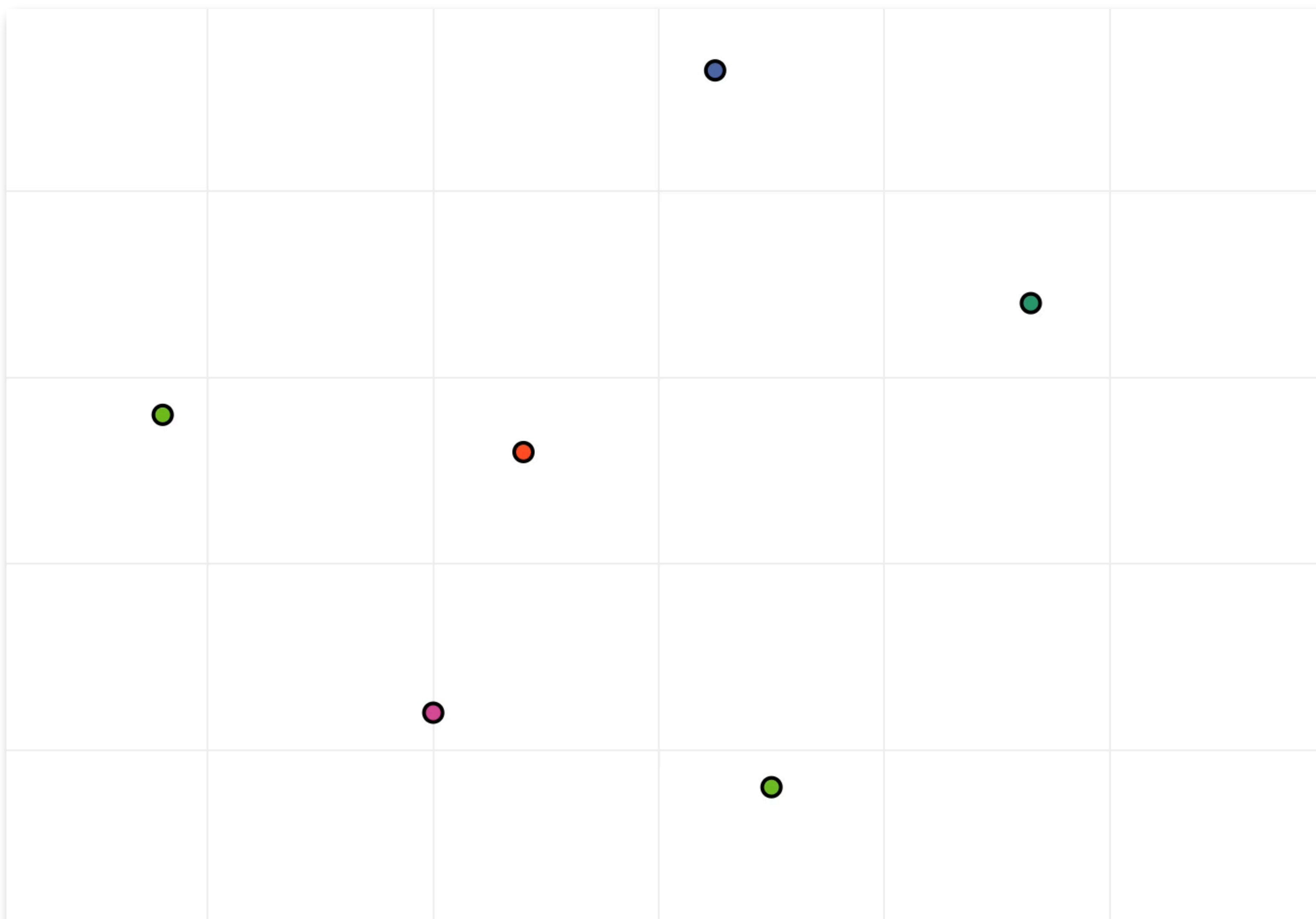






Construction of a *Mesh*

Max Boxes	Iterative Boxes	Voronoi Cells
$\mathcal{O}(n^2 d \log n)$	$\mathcal{O}(n^2 d)$	$\mathcal{O}(n^2 d)$



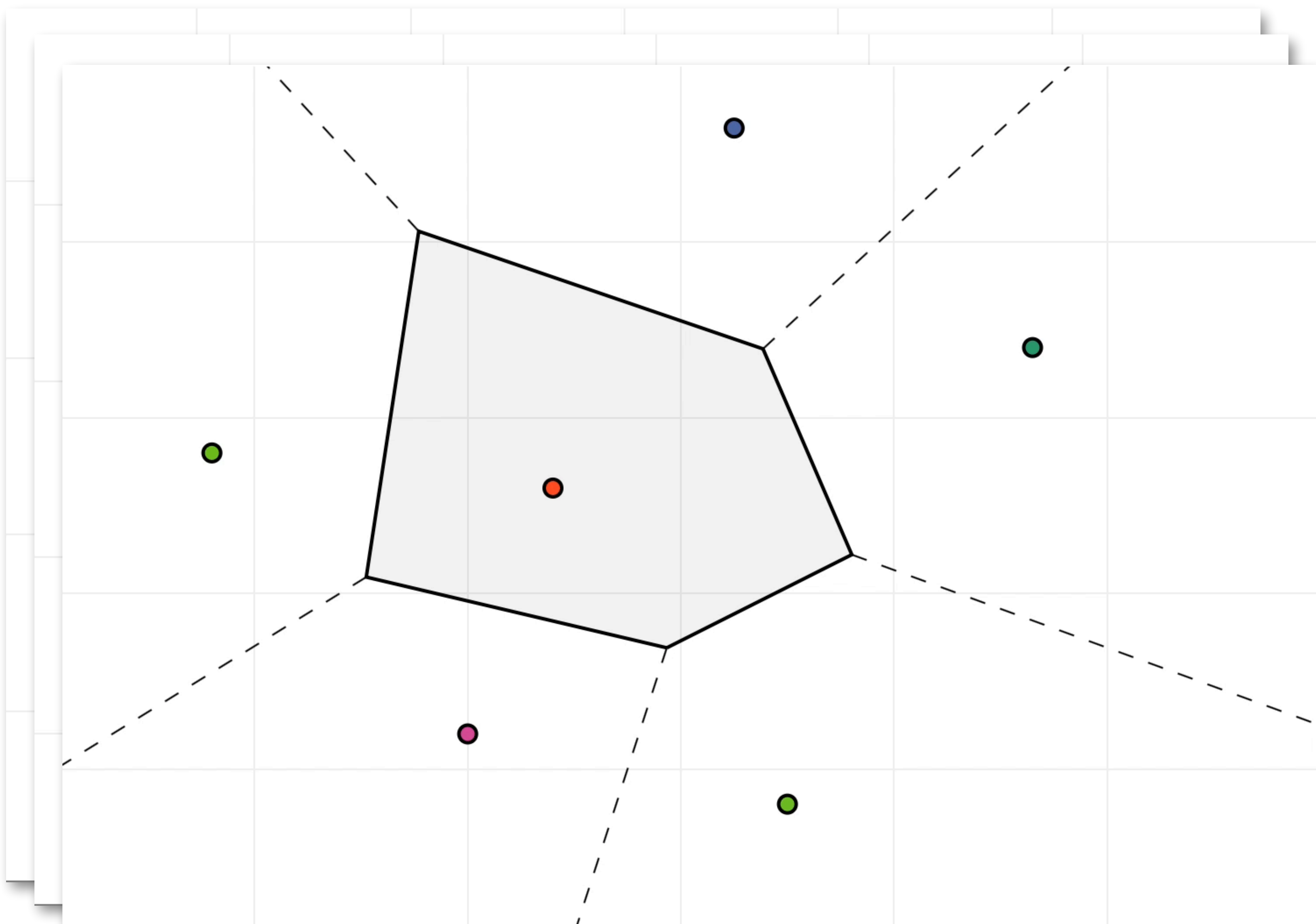
Construction of a *Mesh*

Max Boxes	Iterative Boxes	Voronoi Cells
$\mathcal{O}(n^2 d \log n)$	$\mathcal{O}(n^2 d)$	$\mathcal{O}(n^2 d)$

COMPUTER SCIENCE
VIRGINIA TECH

Construction of a *Mesh*

Max Boxes	Iterative Boxes	Voronoi Cells
$\mathcal{O}(n^2 d \log n)$	$\mathcal{O}(n^2 d)$	$\mathcal{O}(n^2 d)$



Fitting and Bootstrapping

Fitting

Evaluate all basis functions in the mesh at all points *n*.

When "c" is the number of control points used for a mesh, using an ($n \times c$) matrix *A* of basis function evaluations at all points, solve the least squares problem $A\,x = f\,(\mathrm{X})$ with cost $\mathcal{O}(nc^2 + c^3)$.

Bootstrapping

Initialize mesh only using the most central point
 Fit mesh and evaluate error at all other points
 Add (batch of) point(s) with largest error to mesh
 If average error is not below ***error tolerance***, repeat

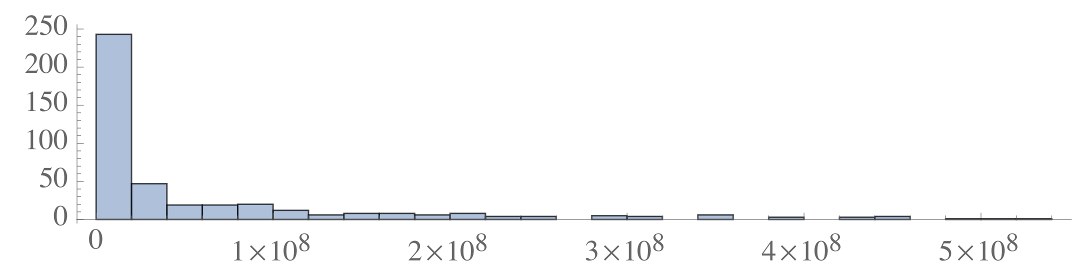
Increased cost up to $\mathcal{O}(n)$

Testing and Evaluation: Data

High Performance Computing File I/O

$n = 532, d = 4$

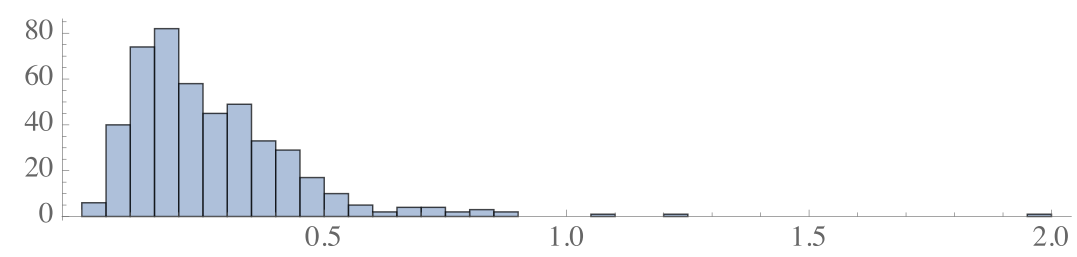
predicting *file I/O throughput*



Forest Fire

$n = 517, d = 12$

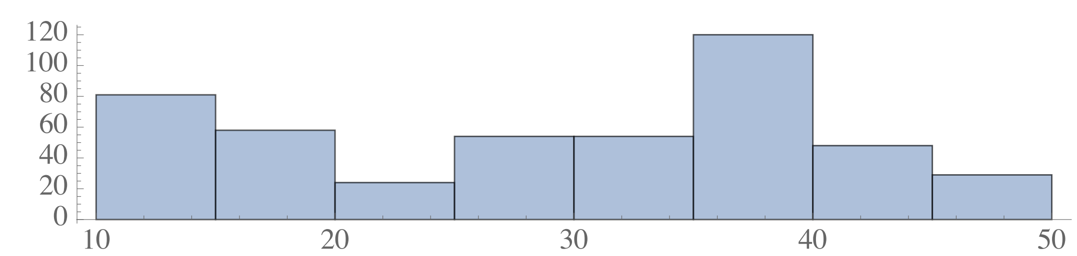
predicting *area burned*



Parkinson's Clinical Evaluation

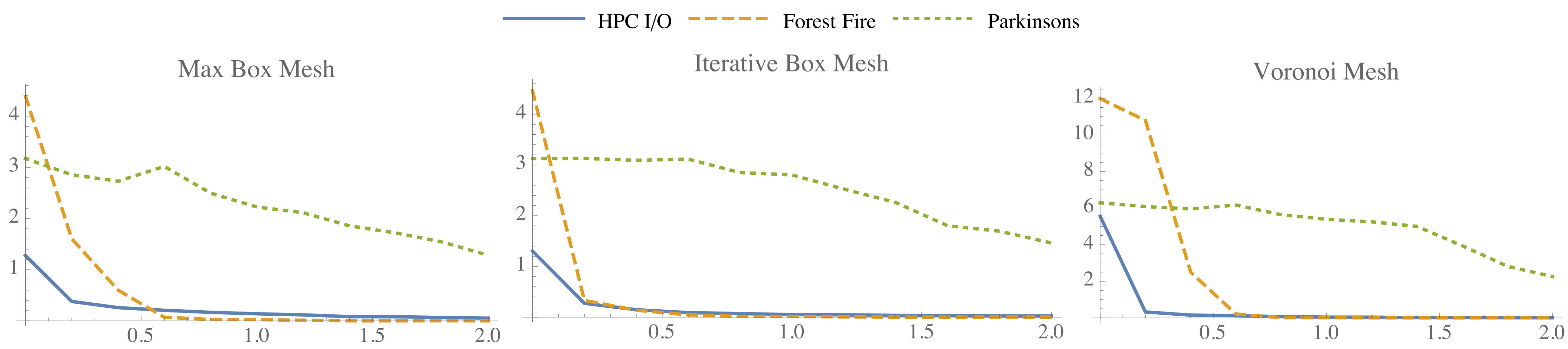
$n = 468, d = 16$

predicting *total clinical "UPDRS" score*



COMPUTER SCIENCE
VIRGINIA TECH.

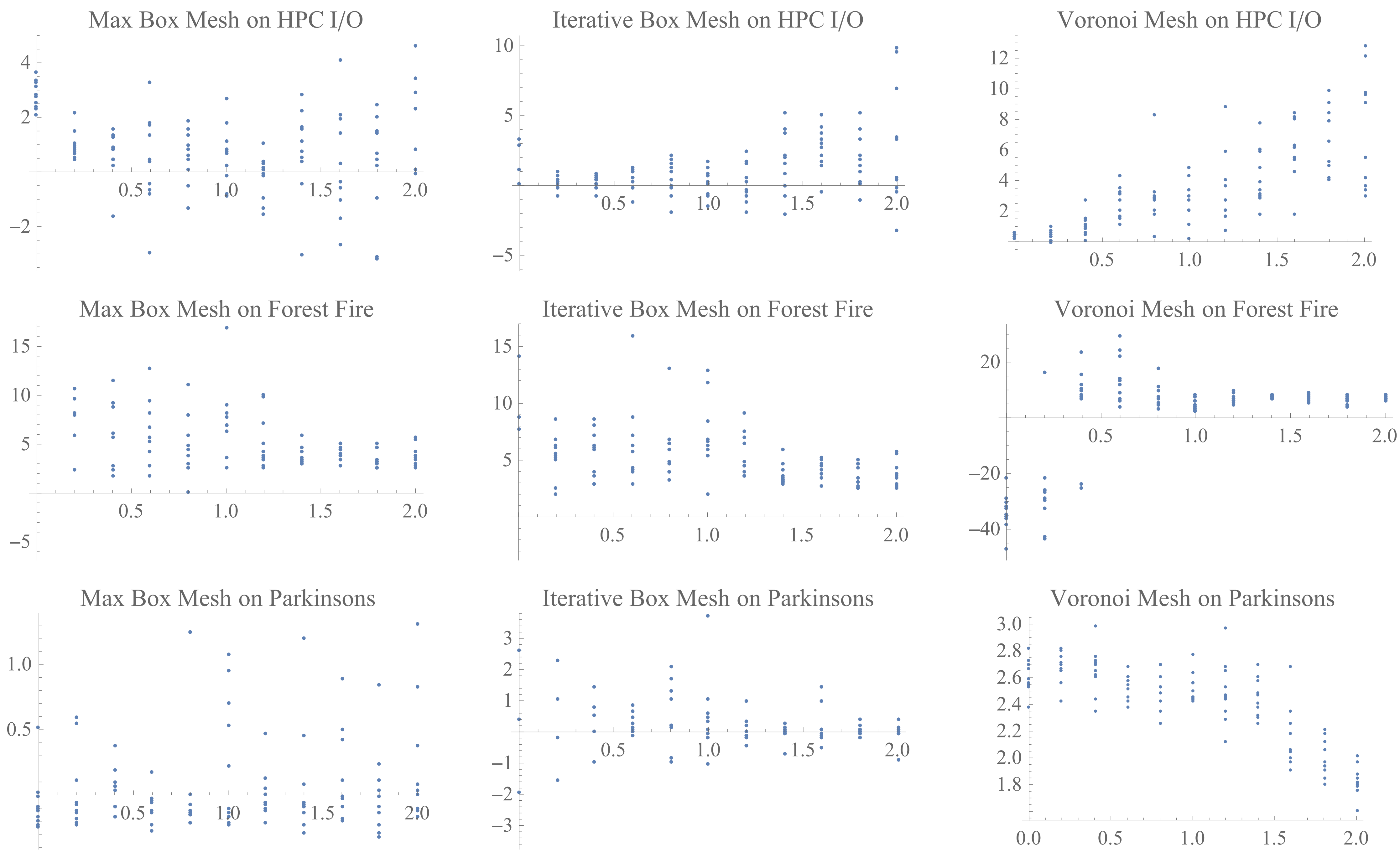
Time to Fit (y-axis) versus Error Tolerance (x-axis)



Optimal Tolerance & Accuracy

Data Set	Technique	Tolerance	Average Error
HPC I/O	MBM	1.2	0.597
Forest Fire	MBM	1.8	3.517
Parkinson’s	MBM	0.6	0.114
HPC I/O	IBM	0.4	0.419
Forest Fire	IBM	1.8	3.615
Parkinson’s	IBM	1.8	0.121
HPC I/O	VM	0.2	0.382
Forest Fire	VM	1.0	4.783
Parkinson’s	VM	2.0	1.824

Average Relative Testing Error (y-axis) versus Relative Error Tolerance (x-axis)



COMPUTER SCIENCE
VIRGINIA TECH

Chapter Takeaways

The “Max” method appears to produce better results than the “Iterative” method. The Voronoi Cell method is best for I/O, but worst for all other tests.

The bootstrapping combined with the least squares computation incurs a lot of computational expense. This methodology **cannot be scaled** to more than 100’s of points.

Chapters

1. The Importance and Applications of Variability
2. Algorithms for Constructing Approximations
3. Naive Approximations of Variability
4. Box-Splines: Uses, Constructions, and Applications
5. Stronger Approximations of Variability
6. An Error Bound for Piecewise Linear Interpolation
7. Improving Variability estimates

Interpolating Distributions

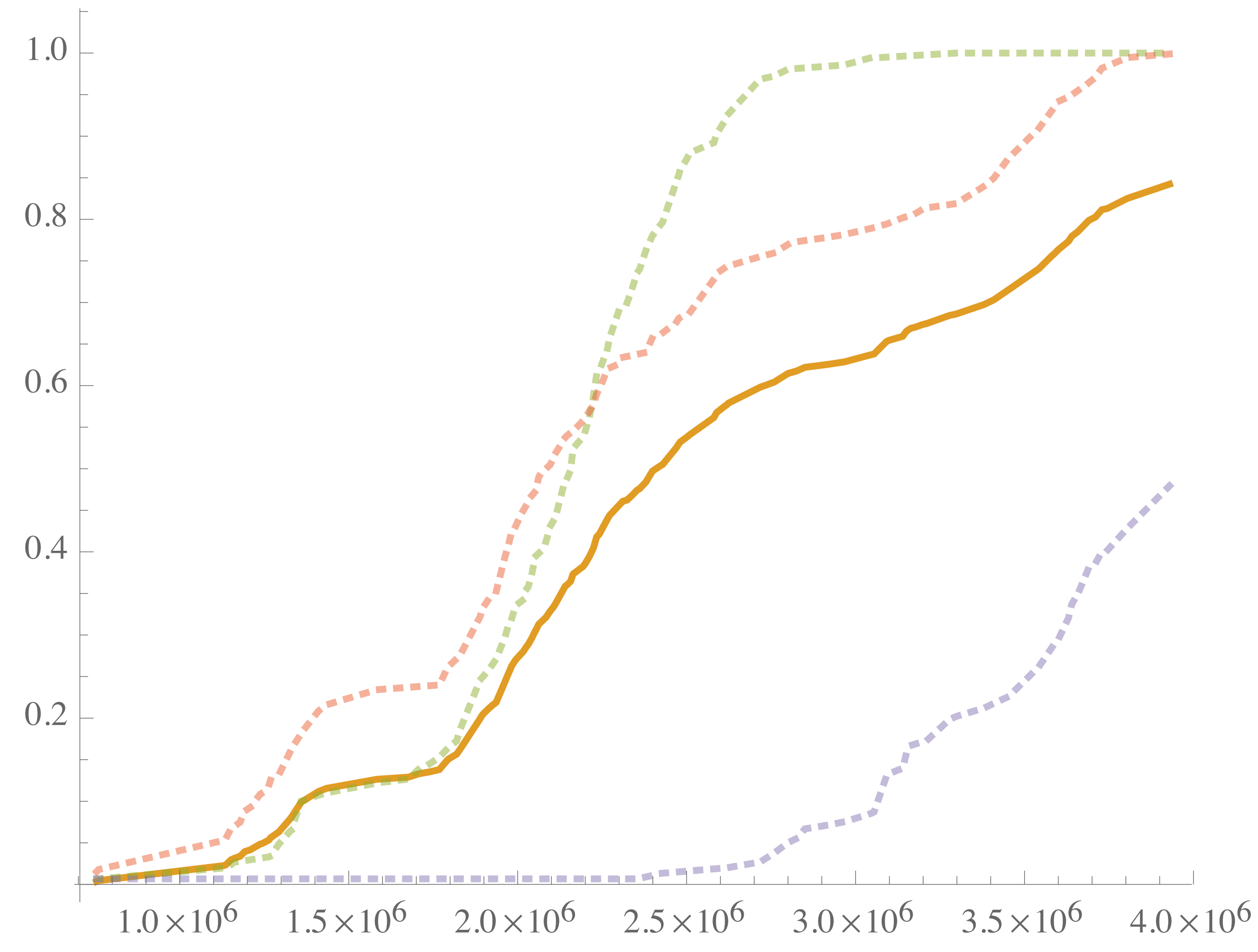
A Cumulative Distribution Function (CDF) $F : \mathbb{R} \rightarrow \mathbb{R}$ must maintain the properties:

$$F(x) \in [0, 1]$$

$F(x)$ is absolutely continuous and nondecreasing.

A convex combination of CDFs results in a valid CDF. Consider this example, solid line is the weighted sum:

{.3 Red, .4 Green, .3 Blue}



Measuring Error in a Prediction

Kolmogorov Smirnov (KS) statistic, max-norm difference.

Null hypothesis (of distributions being same) is rejected at confidence level p according to

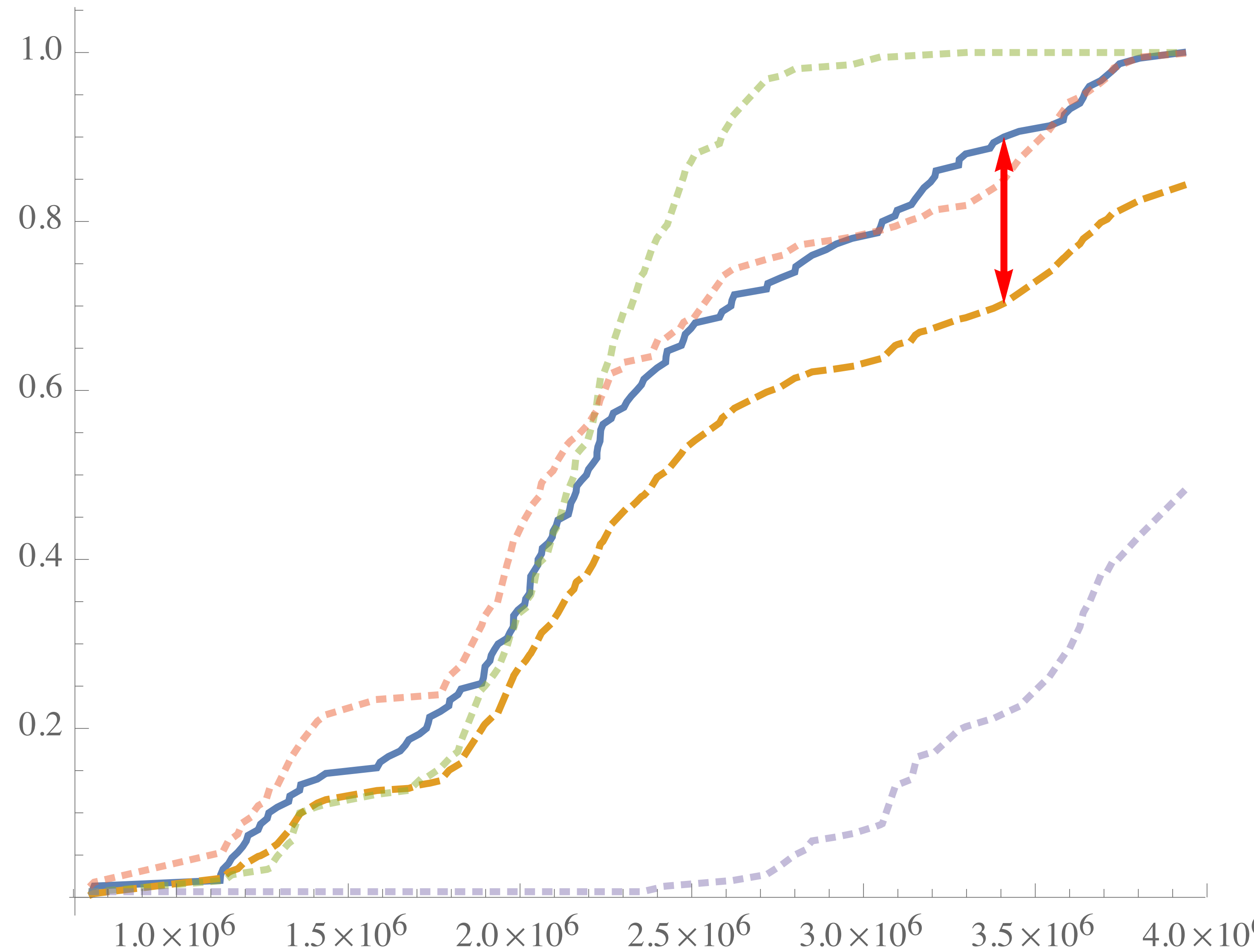
$$KS > \sqrt{-\frac{1}{2}\ln\left(\frac{p}{2}\right)}\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}$$

Dotted lines –
source CDFs

Dashed line –
predicted CDF (Delaunay)

Solid line –
true CDF

Red arrow –
KS statistic between predicted and true (.2)



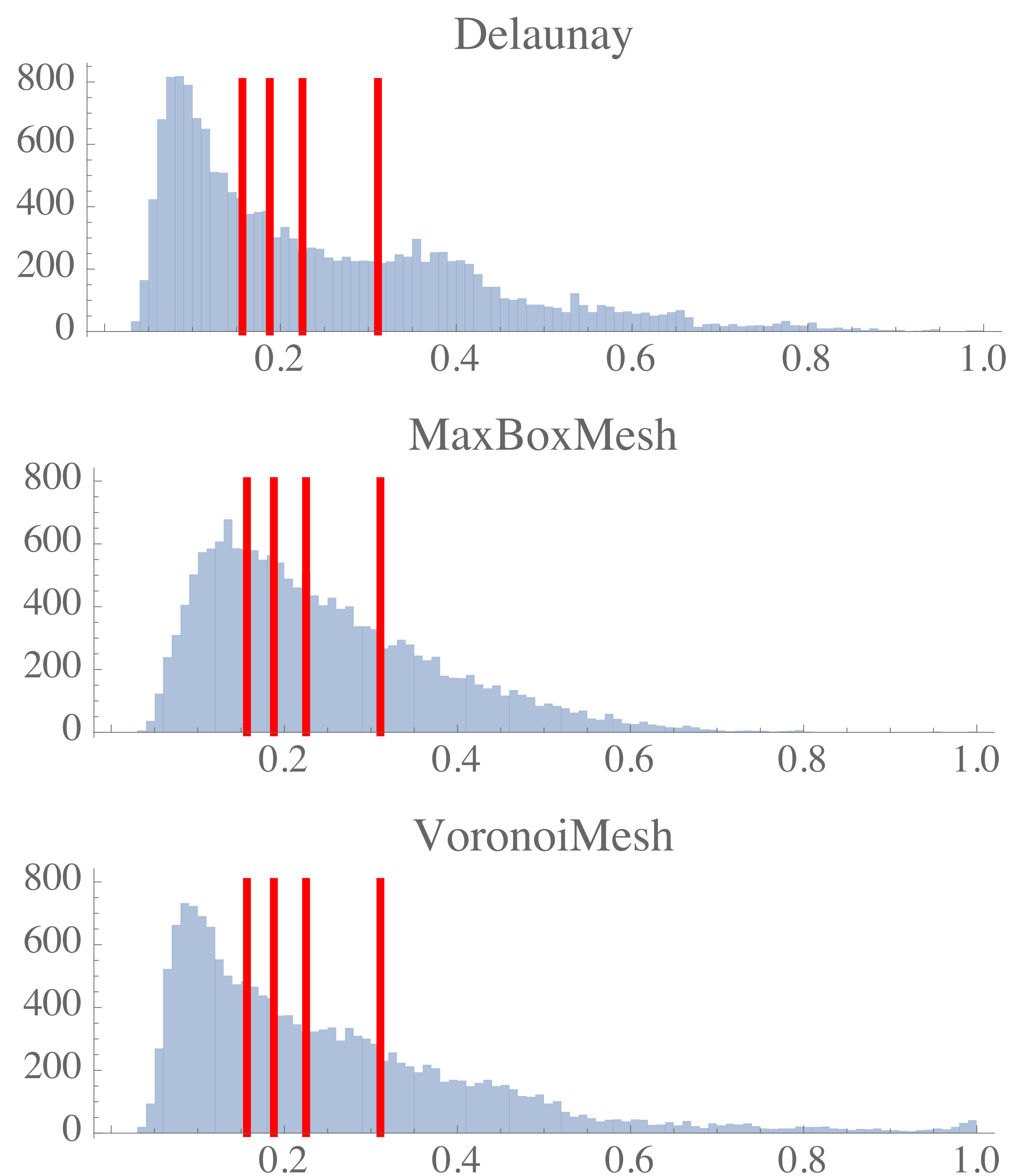
COMPUTER SCIENCE
VIRGINIA TECH

Results Applied to IOzone

x-axis – KS Statistic
y-axis – Number of predictions

Red lines:
  KS significance levels at
  {.1 .05, .01, .001}

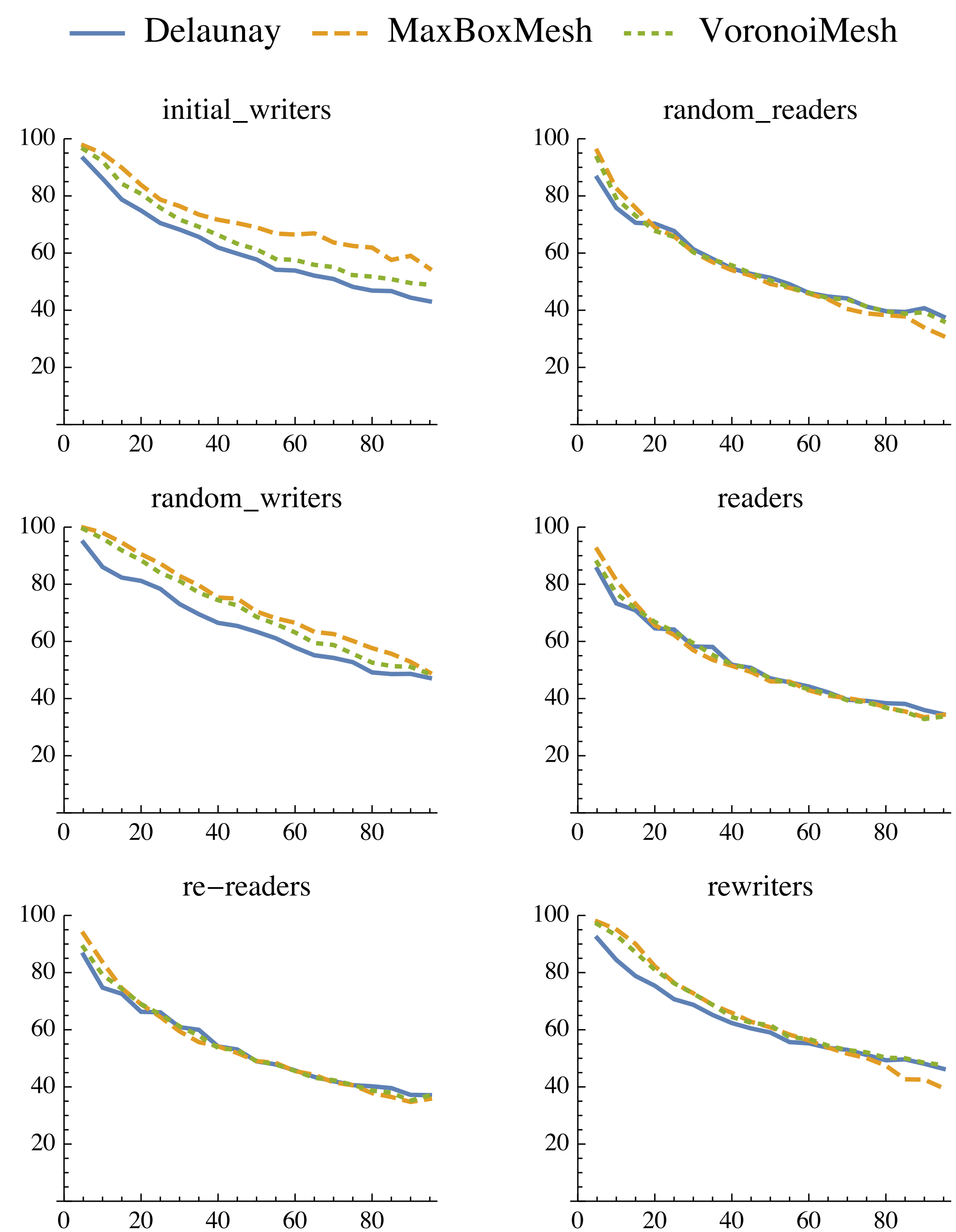
Consider all values to the right of a red line an “incorrect” prediction at that significance.

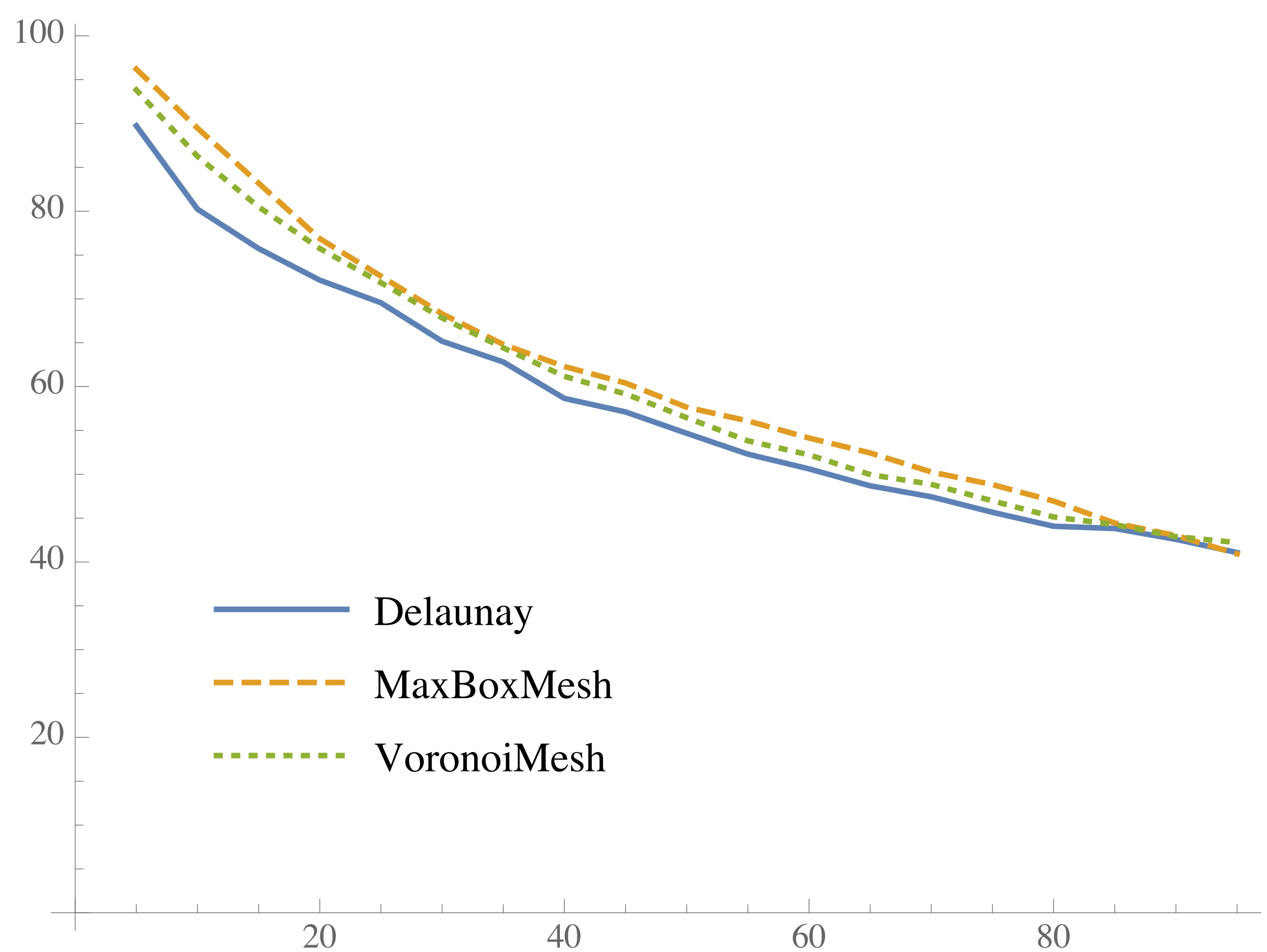


COMPUTER SCIENCE
VIRGINIA TECH.

Increasing Training Data

x-axis – Percentage training data
y-axis – Percentage N.H. rejections
Below: Aggregate
Right: Breakdown by Test





COMPUTER SCIENCE
VIRGINIA TECH.

Improving Performance with *Tuning*

Algorithm	P-Value	Unweighted % N.H. Rejection	Weighted % N.H. Rejection
Delaunay	.05	24.9	30.2
Max Box Mesh		21.3	21.2
Voronoi Mesh		18.7	11.3
Delaunay	.01	21.6	27.4
Max Box Mesh		16.4	16.4
Voronoi Mesh		14.9	7.0
Delaunay	.001	19.7	25.4
Max Box Mesh		13.1	13.1
Voronoi Mesh		12.3	4.6
Delaunay	1.0e-6	17.9	23.4
Max Box Mesh		11.3	11.3
Voronoi Mesh		8.5	2.3

Consensus optimal weighting of (.001, 2, 1.7, 1.5), for frequency, file size, record size, and number of threads. Frequency is unimportant.

Chapter Takeaways

Without any modification, many interpolants can be used to predict distributions! Particularly, those that make predictions with convex combinations of known function values.

Distribution prediction performs well, impressively so with tuning (however the tuning is less provably useful).

20K system configurations appears to approach the limit of distribution prediction accuracy. If we had a better way to approximate distributions, we might reduce error further.

Chapters

1. The Importance and Applications of Variability
2. Algorithms for Constructing Approximations
3. Naive Approximations of Variability
4. Box-Splines: Uses, Constructions, and Applications
5. Stronger Approximations of Variability
6. An Error Bound for Piecewise Linear Interpolation
7. Improving Variability estimates

The Theory

$$|f(z) - \hat{f}(z)| \leq \frac{\gamma\|z - x_0\|_2^2}{2} + \sqrt{d}\frac{\gamma k^2}{\sigma_d}\|z - x_0\|_2$$

The Theory

$$\left|f(z) - \hat{f}(z)\right| \leq \frac{\gamma\|z - x_0\|_2^2}{2} + \sqrt{d}\frac{\gamma k^2}{\sigma_d}\|z - x_0\|_2$$

The absolute error of a linear interpolant is tightly upper bounded by

COMPUTER SCIENCE
VIRGINIA TECH.

The Theory

$$\left|f(z) - \hat{f}(z)\right| \leq \frac{\gamma\|z - x_0\|_2^2}{2} + \sqrt{d}\frac{\gamma k^2}{\sigma_d}\|z - x_0\|_2$$

The absolute error of a linear interpolant is tightly upper bounded by

the max change in slope of the function

COMPUTER SCIENCE
VIRGINIA TECH.

The Theory

$$\left|f(z)-\hat{f}(z)\right| \leq \frac{\gamma\|z-x_0\|_2^2}{2} + \sqrt{d}\frac{\gamma k^2}{\sigma_d}\|z-x_0\|_2$$

The absolute error of a linear interpolant is tightly upper bounded by

the max change in slope of the function

times the distance to the nearest known point squared

COMPUTER SCIENCE
VIRGINIA TECH.

The Theory

$$|f(z) - \hat{f}(z)| \leq \frac{\gamma \|z - x_0\|_2^2}{2} + \sqrt{d}\frac{\gamma k^2}{\sigma_d}\|z - x_0\|_2$$

The absolute error of a linear interpolant is tightly upper bounded by

the max change in slope of the function

times the distance to the nearest known point squared

plus the square root of the dimension times the max change in slope

COMPUTER SCIENCE
VIRGINIA TECH.

The Theory

$$\left|f(z) - \hat{f}(z)\right| \leq \frac{\gamma \|z - x_0\|_2^2}{2} + \sqrt{d}\frac{\gamma k^2}{\sigma_d}\|z - x_0\|_2$$

The absolute error of a linear interpolant is tightly upper bounded by

the max change in slope of the function

times the distance to the nearest known point squared

plus the square root of the dimension times the max change in slope

times the longest edge length between points defining the linear interpolant squared

The Theory

$$\left|f(z) - \hat{f}(z)\right| \leq \frac{\gamma\|z - x_0\|_2^2}{2} + \sqrt{d}\frac{\gamma k^2}{\sigma_d}\|z - x_0\|_2$$

The absolute error of a linear interpolant is tightly upper bounded by

the max change in slope of the function

times the distance to the nearest known point squared

plus the square root of the dimension times the max change in slope

times the longest edge length between points defining the linear interpolant squared

divided by how close the interpolated points are to being planar.

COMPUTER SCIENCE
VIRGINIA TECH

The Importance

The approximation error of a linear (simplicial) interpolant tends quadratically towards zero when approaching observed data only when the diameter of the simplex is also reduced proportionally.

In practice, only linear convergence to the true function can be achieved (because the evaluation points don't move).

Approximation error is largely determined by data spacing!

This theory only directly applies to Delaunay, but may give insight into the approximation behavior of other techniques.

Data Sets for Empirical Evaluation

Forest Fire (*n = 504, d = 12*)
given meteorological information about a park, predict the amount of land that would be burned in a forest fire.

Parkinson's Telemonitoring (*n = 5875, d = 19*)
given features of audio recorded in the home of someone with Parkinson's, predict their next clinical evaluation score.

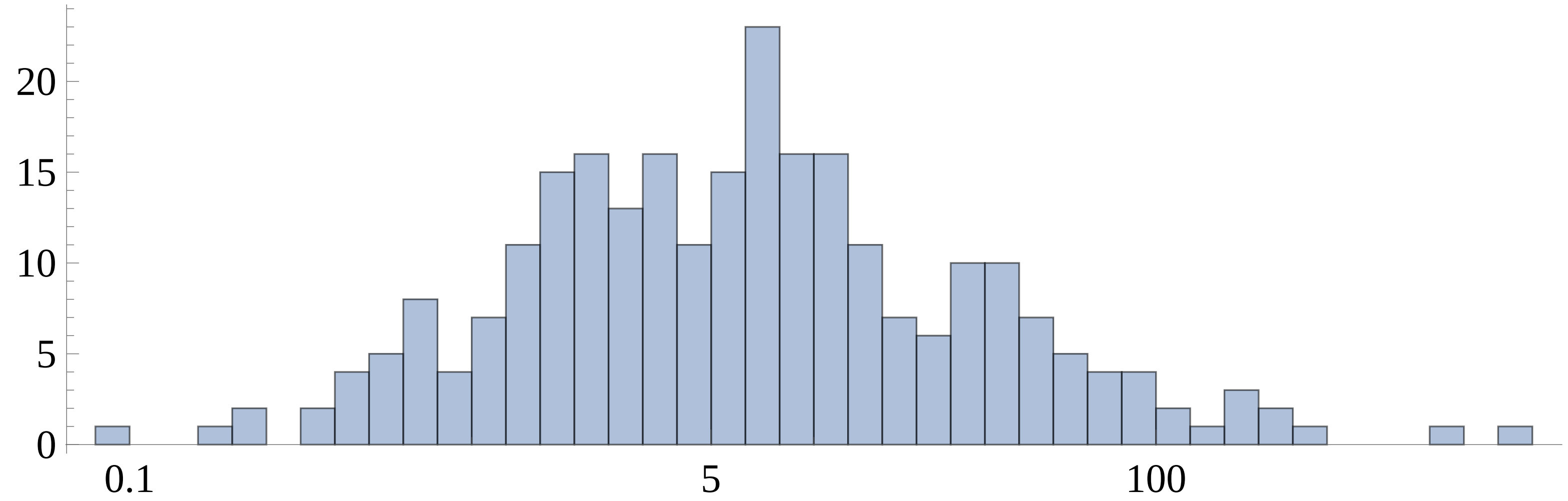
Australian Daily Rainfall Volume (*n = 2609, d = 23*)
given meteorological data around Sydney, Australia, predict the amount of rainfall that will occur on the next day.

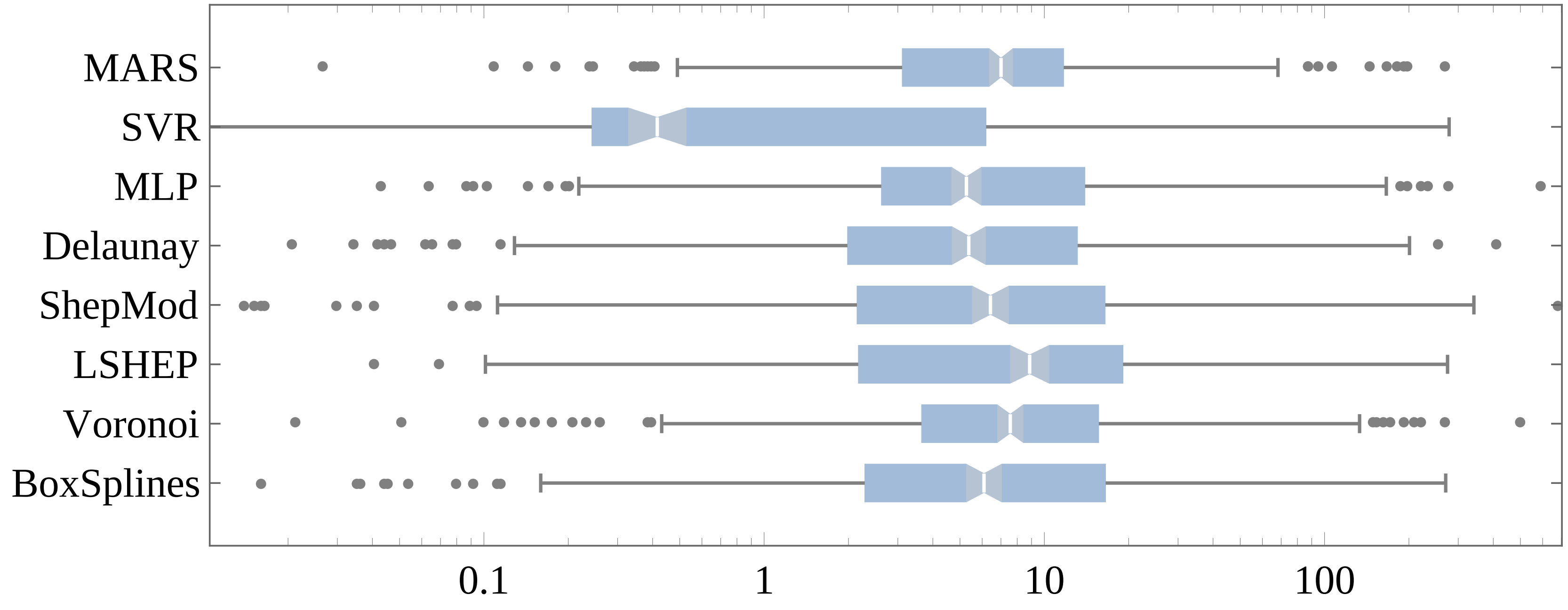
Credit Card Transaction Amount (*n = 5562, d = 28*)
given anonymized electronic transaction features (output of PCA) predict the amount of money that the transaction will process.

High Performance Computing I/O (*n = 3016, d = 4*)
given system configuration information, predict the distribution of I/O throughput that will be seen at a new configuration.

Empirical Results for Each Data Set

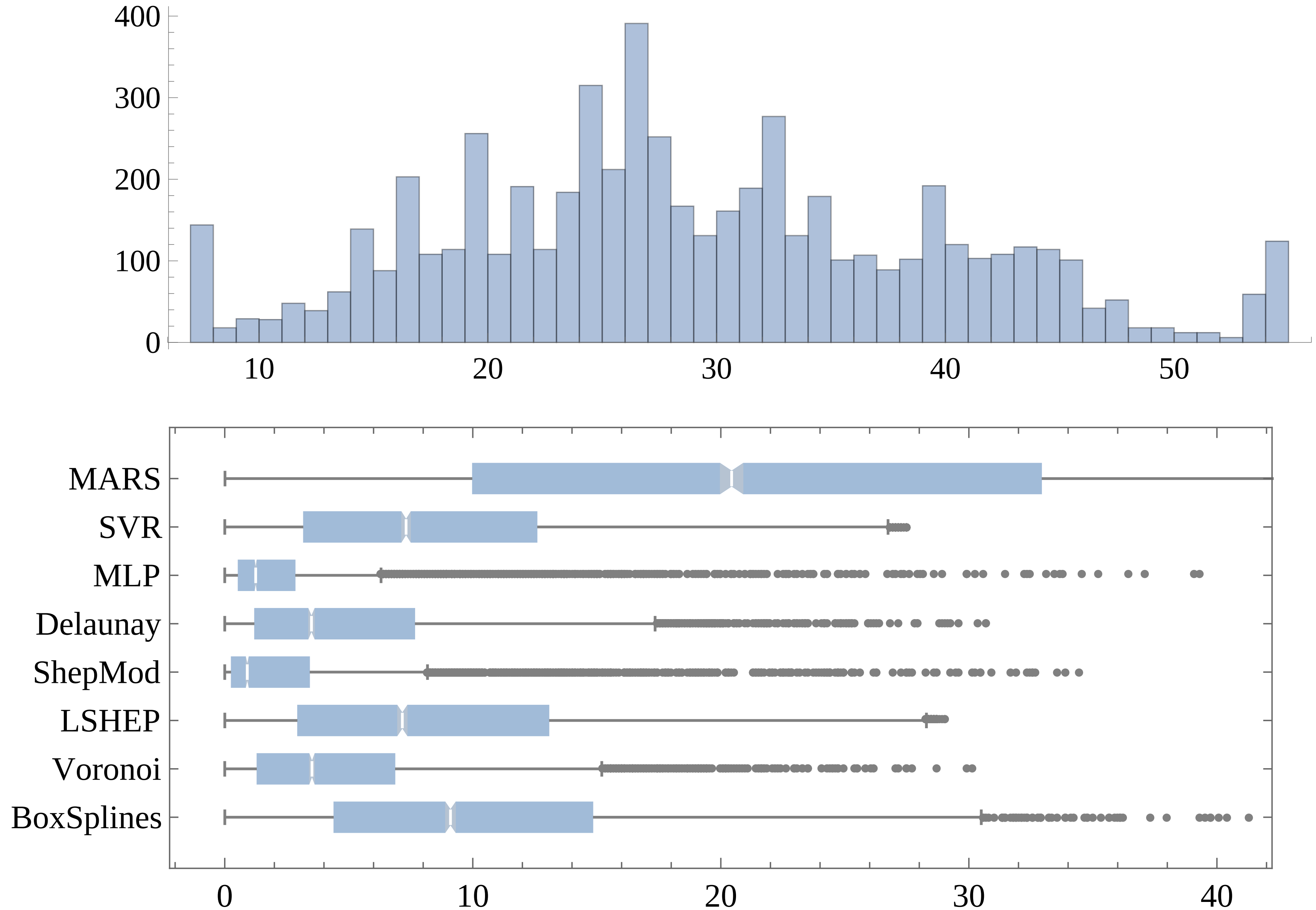
Forest fire data





Empirical Results for Each Data Set

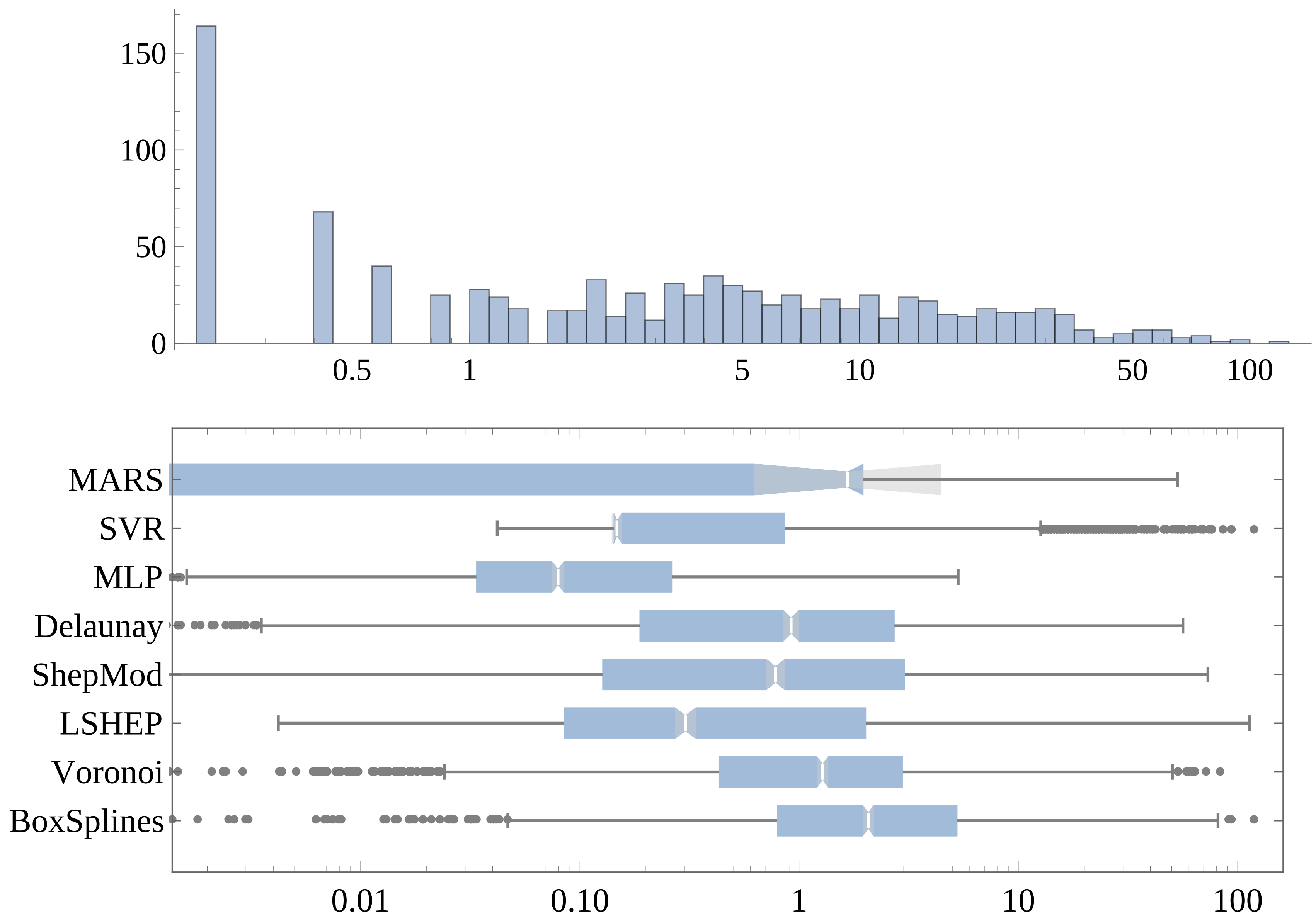
Parkinson's Telimonitoring



COMPUTER SCIENCE
VIRGINIA TECH.

Empirical Results for Each Data Set

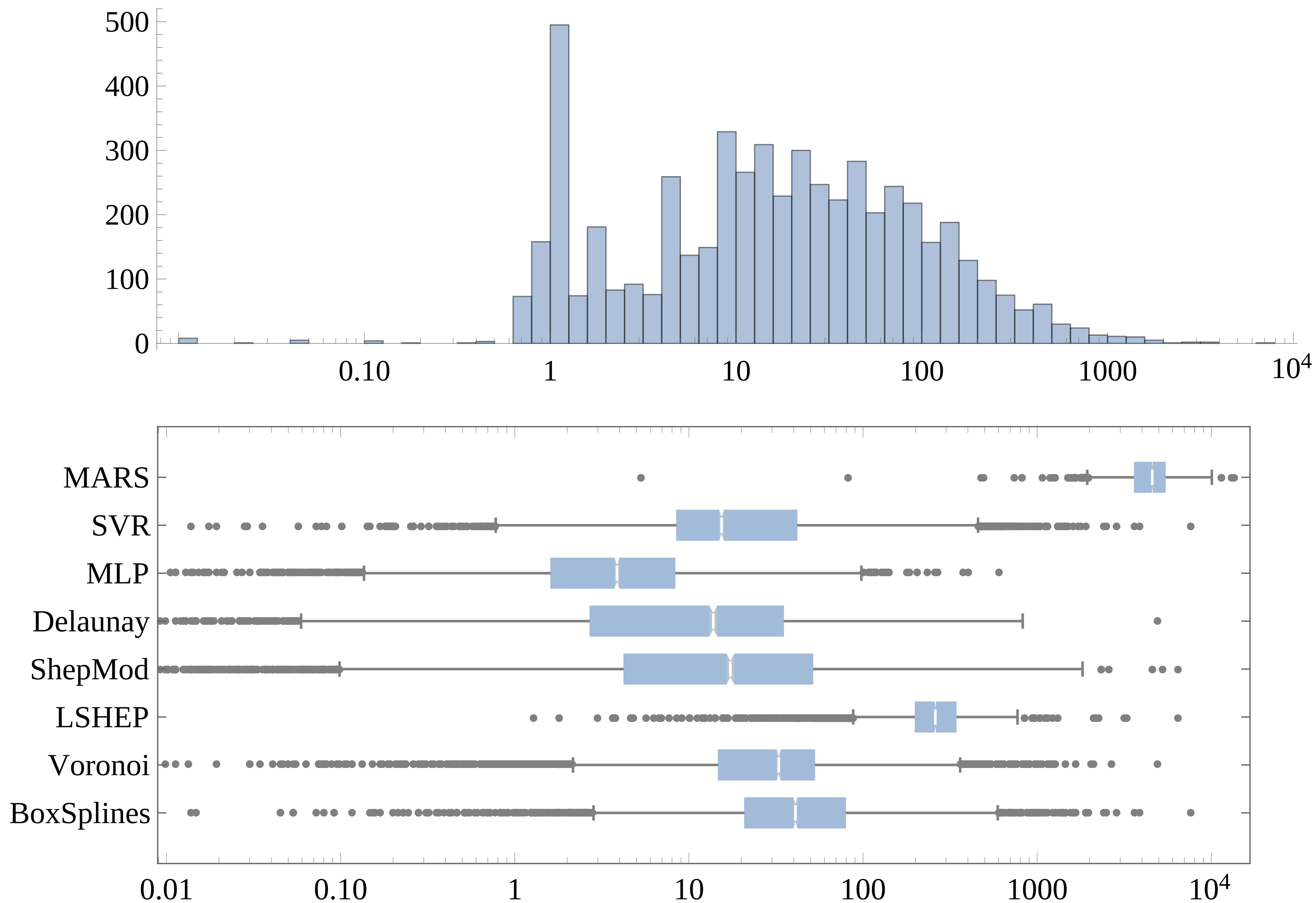
Australian Rainfall



COMPUTER SCIENCE
VIRGINIA TECH.

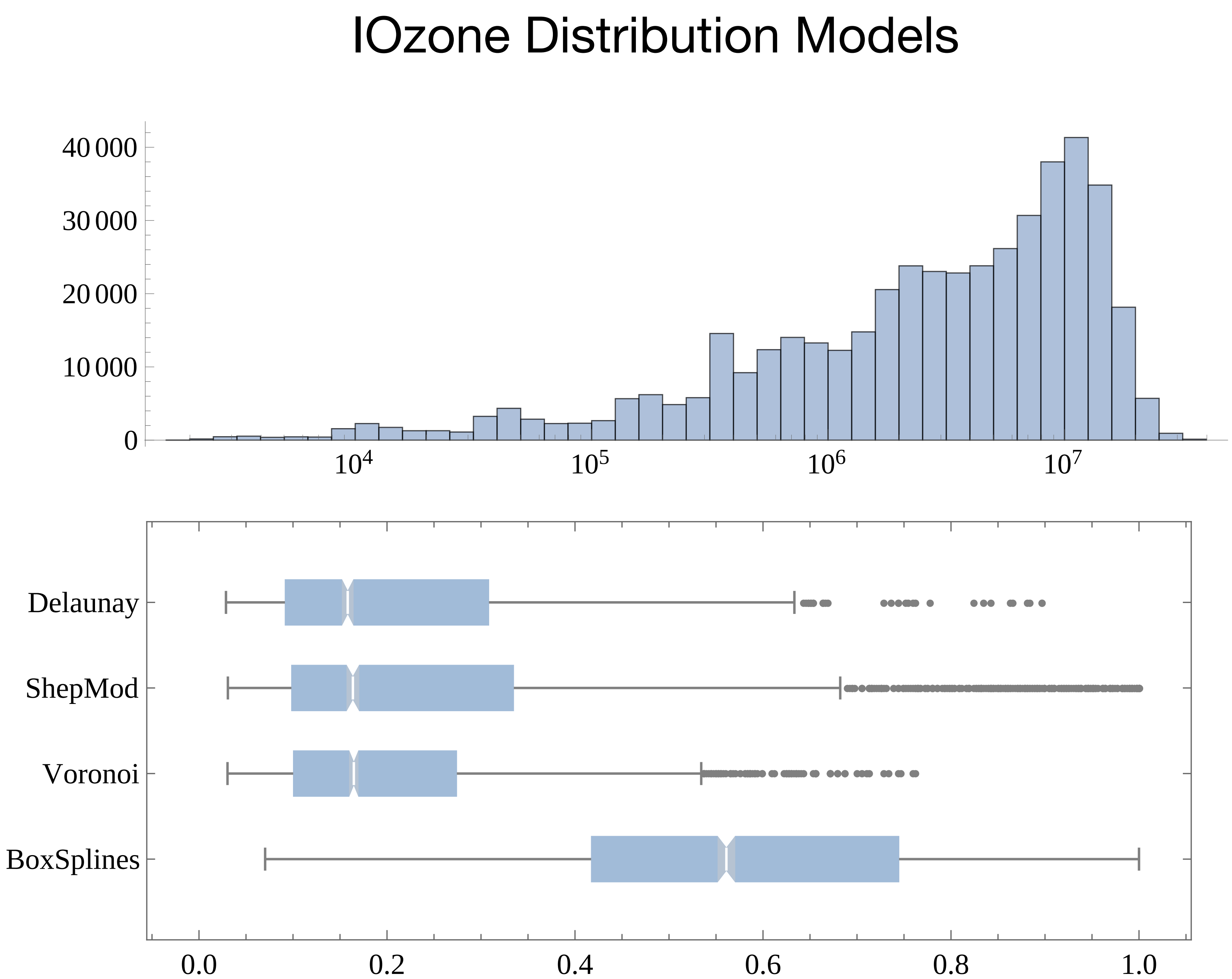
Empirical Results for Each Data Set

Credit Card Transactions



COMPUTER SCIENCE
VIRGINIA TECH.

Empirical Results for Each Data Set



COMPUTER SCIENCE
VIRGINIA TECH.

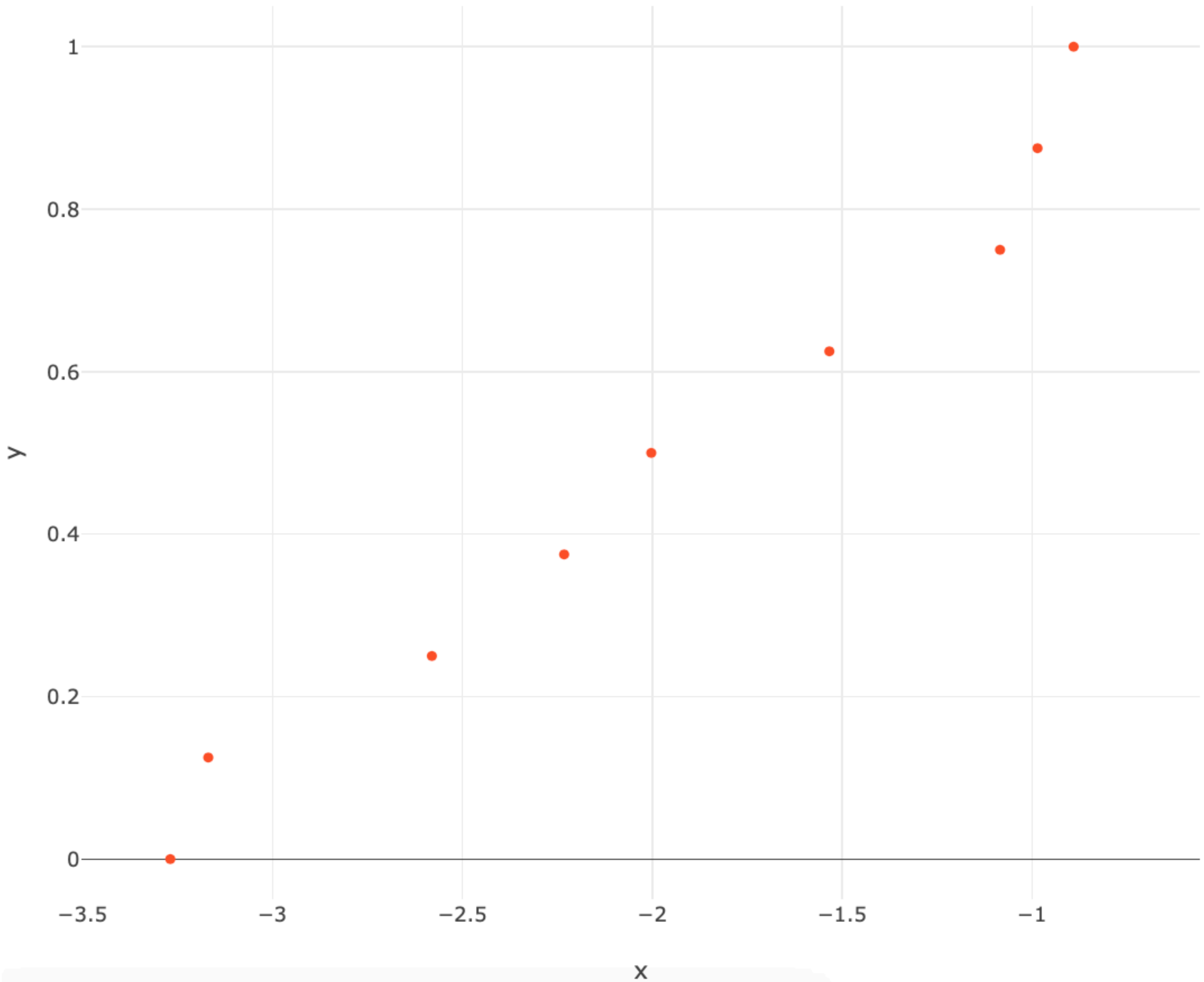
Chapters

1. The Importance and Applications of Variability
2. Algorithms for Constructing Approximations
3. Naive Approximations of Variability
4. Box-Splines: Uses, Constructions, and Applications
5. Stronger Approximations of Variability
6. An Error Bound for Piecewise Linear Interpolation
7. Improving Variability estimates

COMPUTER SCIENCE
VIRGINIA TECH

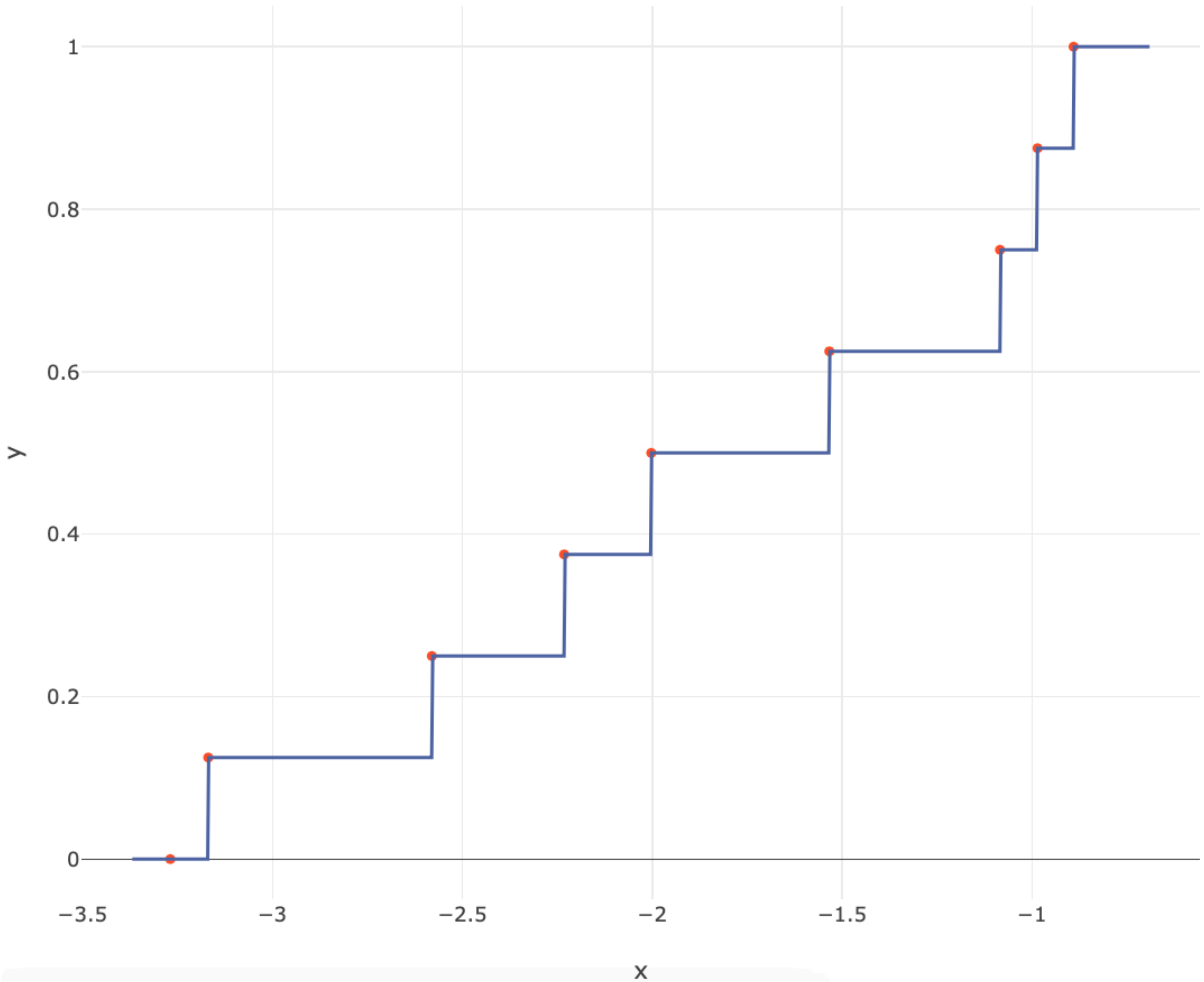
Empirical Distribution Functions

[-3.17, -2.58, -2.23, -2.0, -1.53, -1.08, -0.99, -0.89]



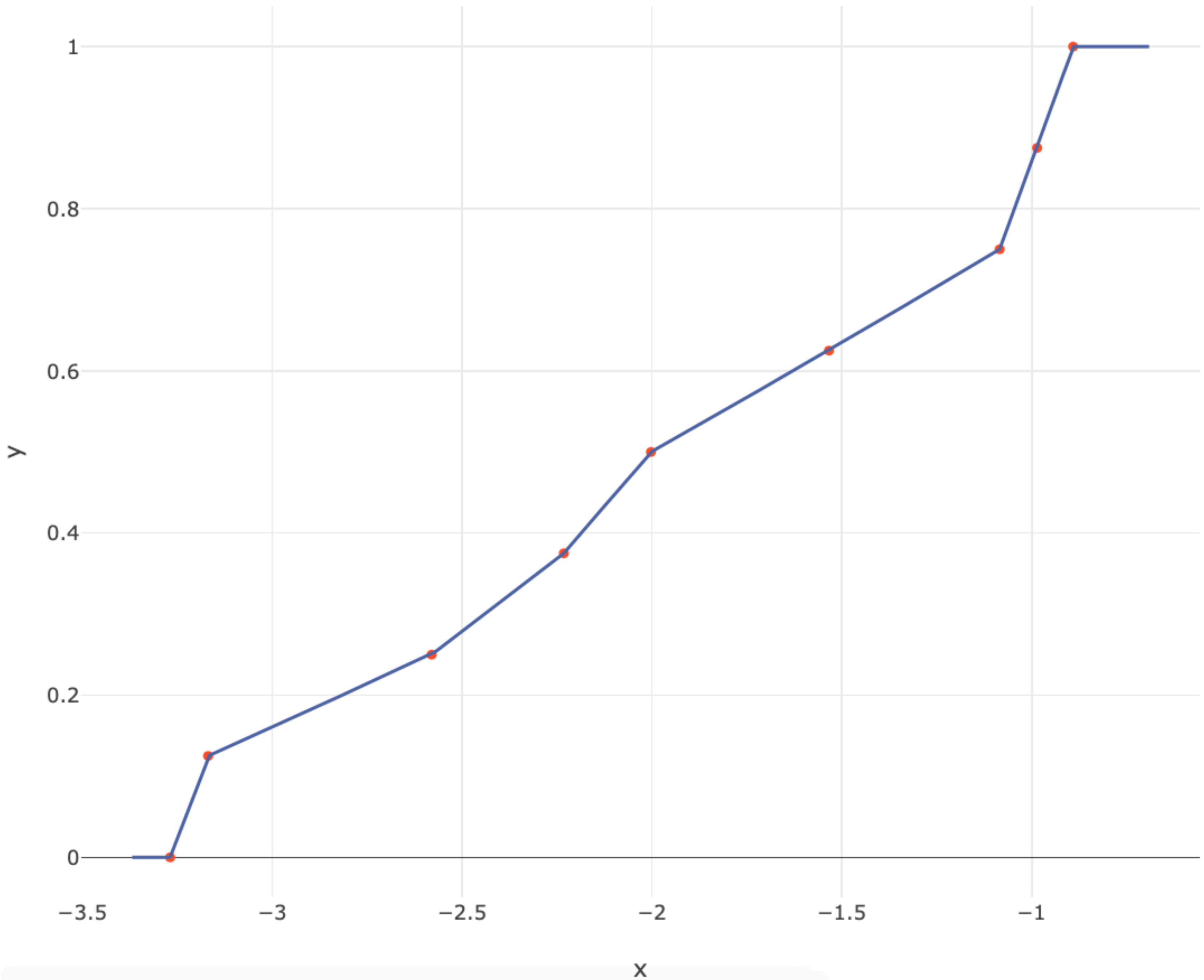
Empirical Distribution Functions

[-3.17, -2.58, -2.23, -2.0, -1.53, -1.08, -0.99, -0.89]



Empirical Distribution Functions

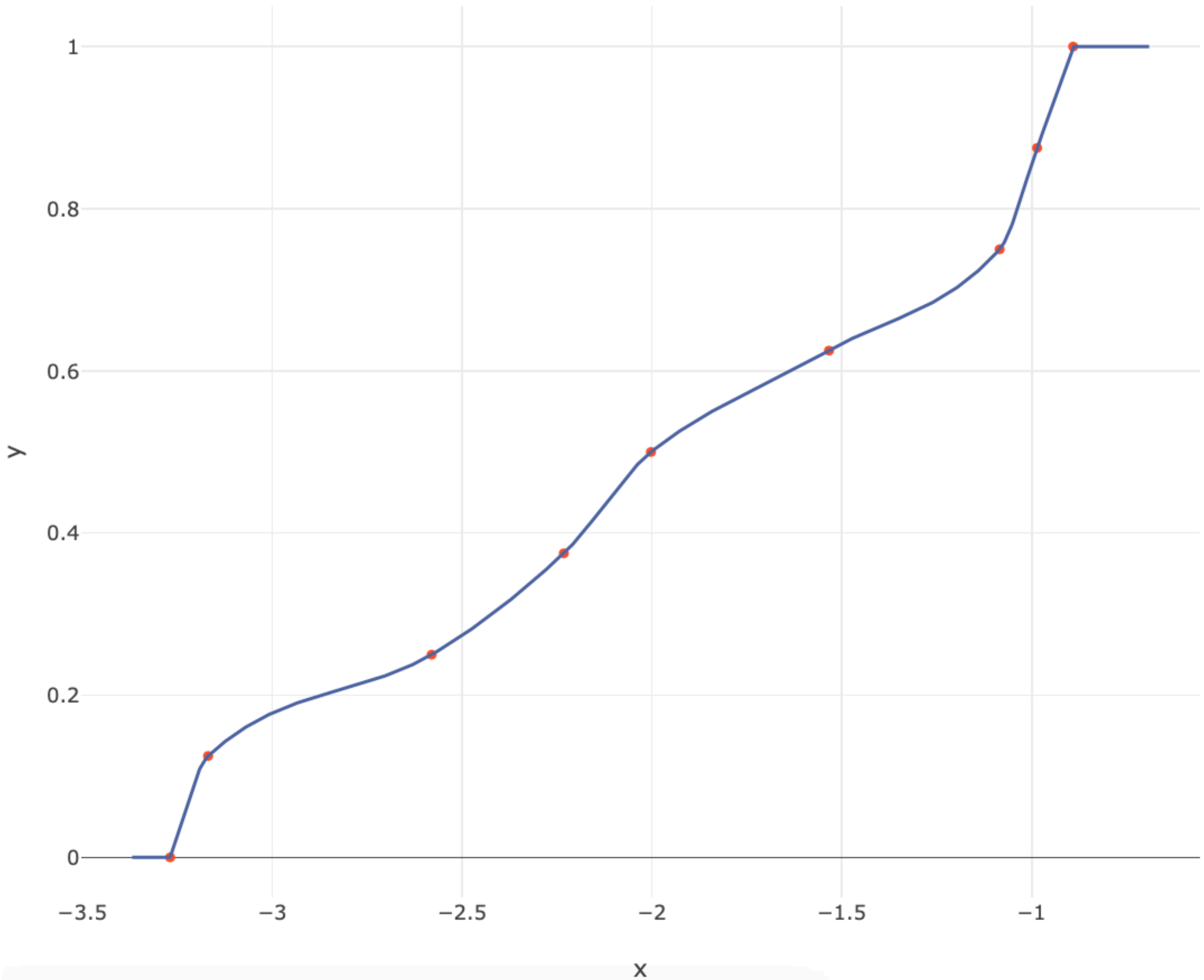
[-3.17, -2.58, -2.23, -2.0, -1.53, -1.08, -0.99, -0.89]



COMPUTER SCIENCE
VIRGINIA TECH.

Empirical Distribution Functions

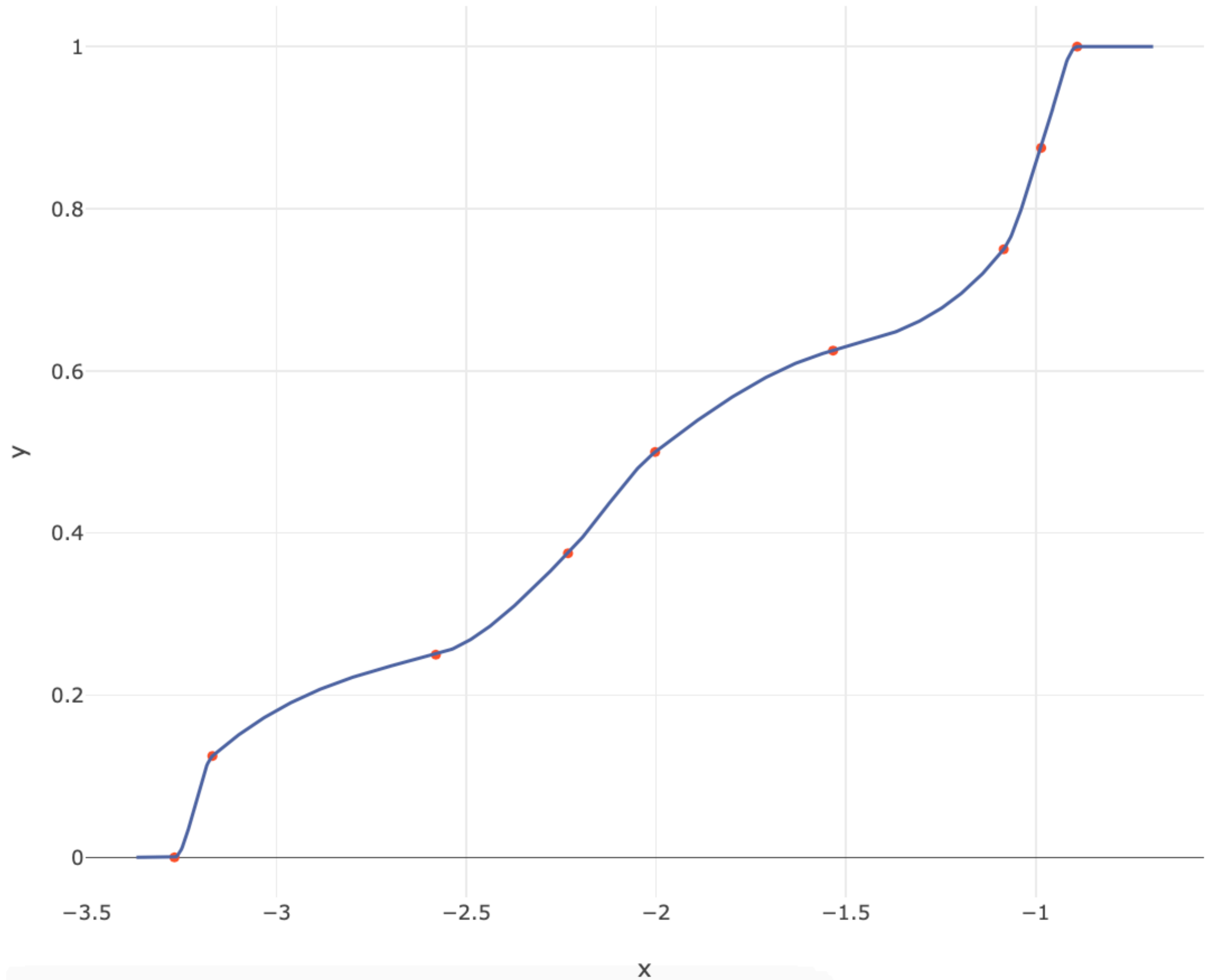
[-3.17, -2.58, -2.23, -2.0, -1.53, -1.08, -0.99, -0.89]



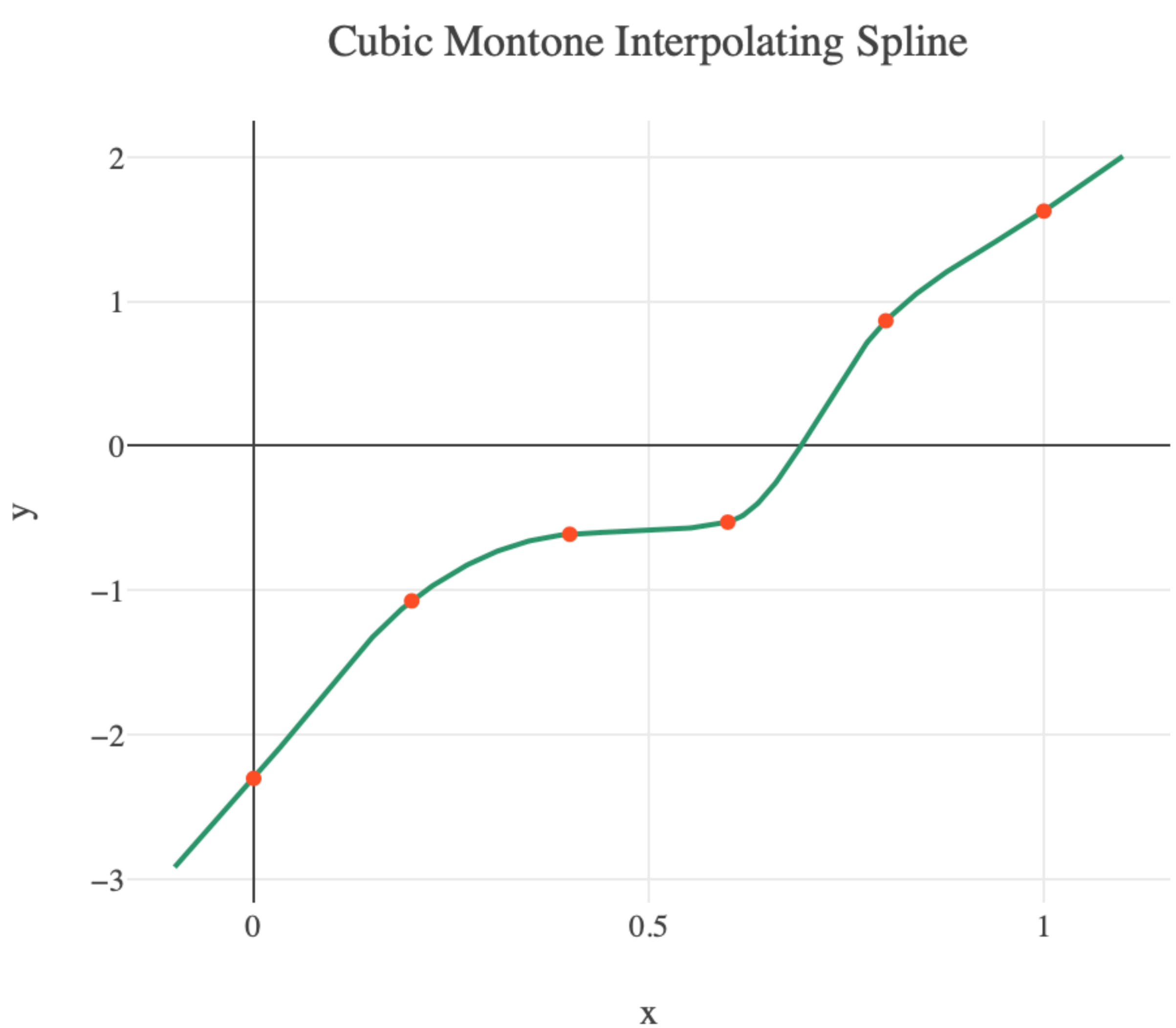
COMPUTER SCIENCE
VIRGINIA TECH.

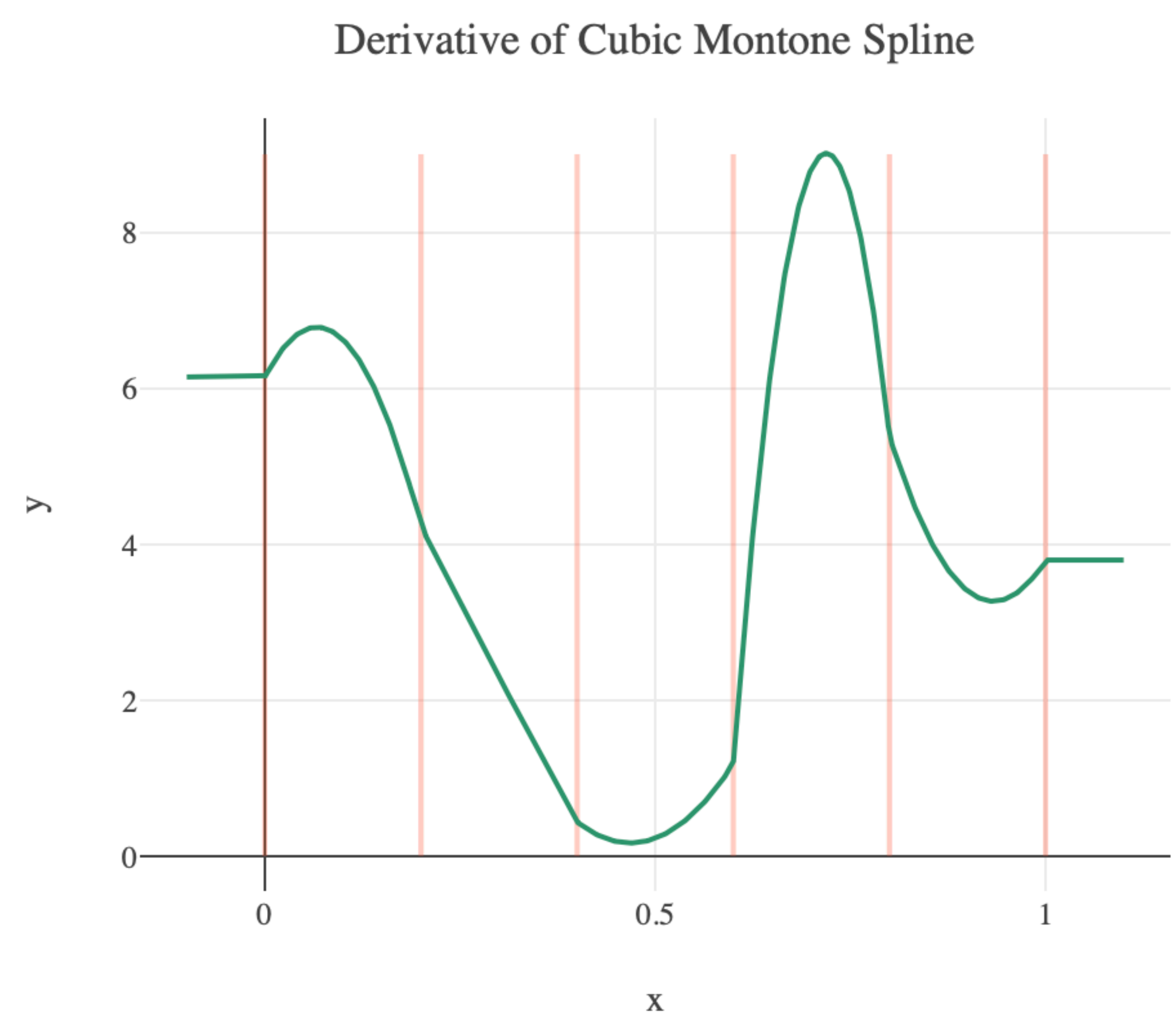
Empirical Distribution Functions

[-3.17, -2.58, -2.23, -2.0, -1.53, -1.08, -0.99, -0.89]

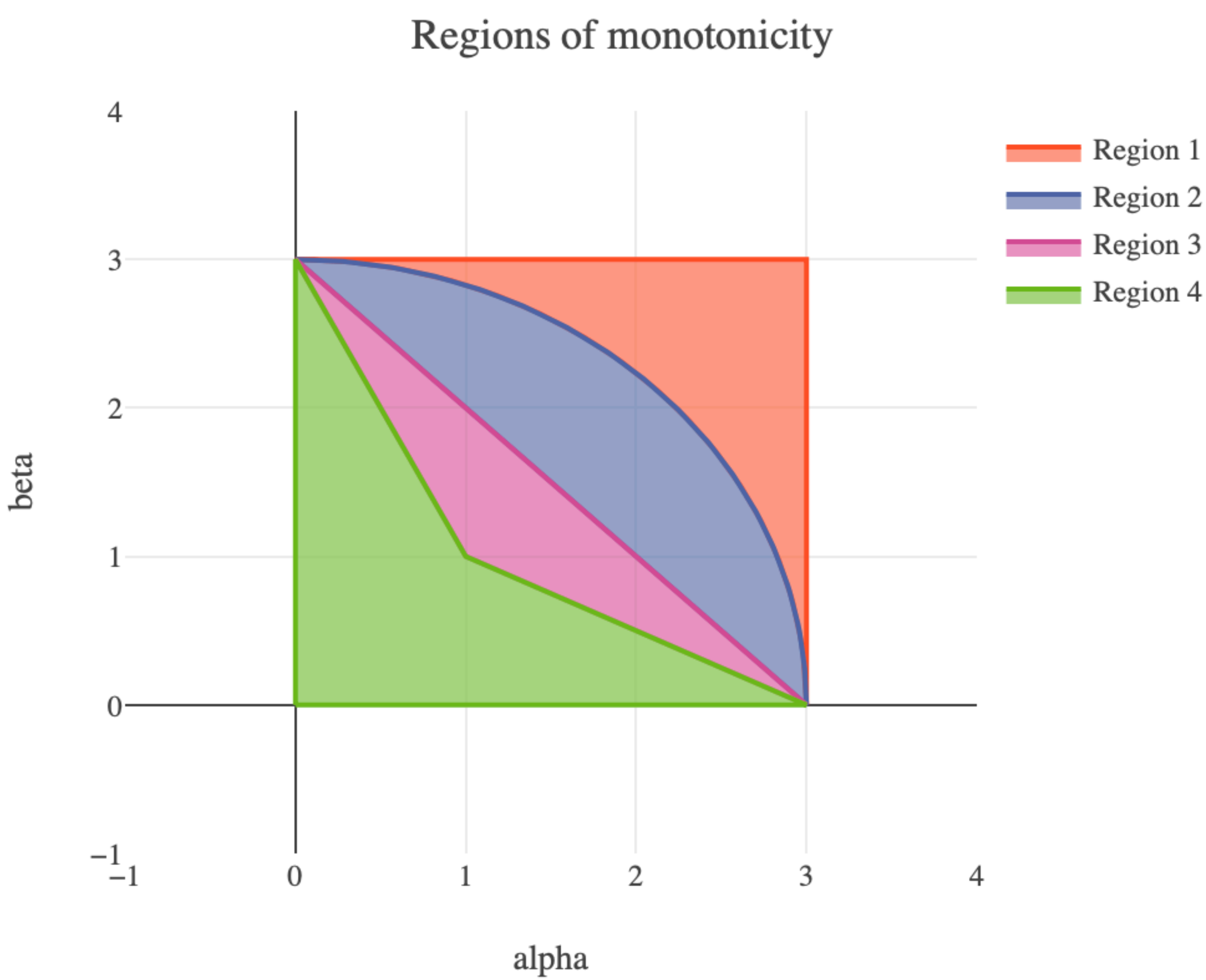


Monotone Cubic Approximations



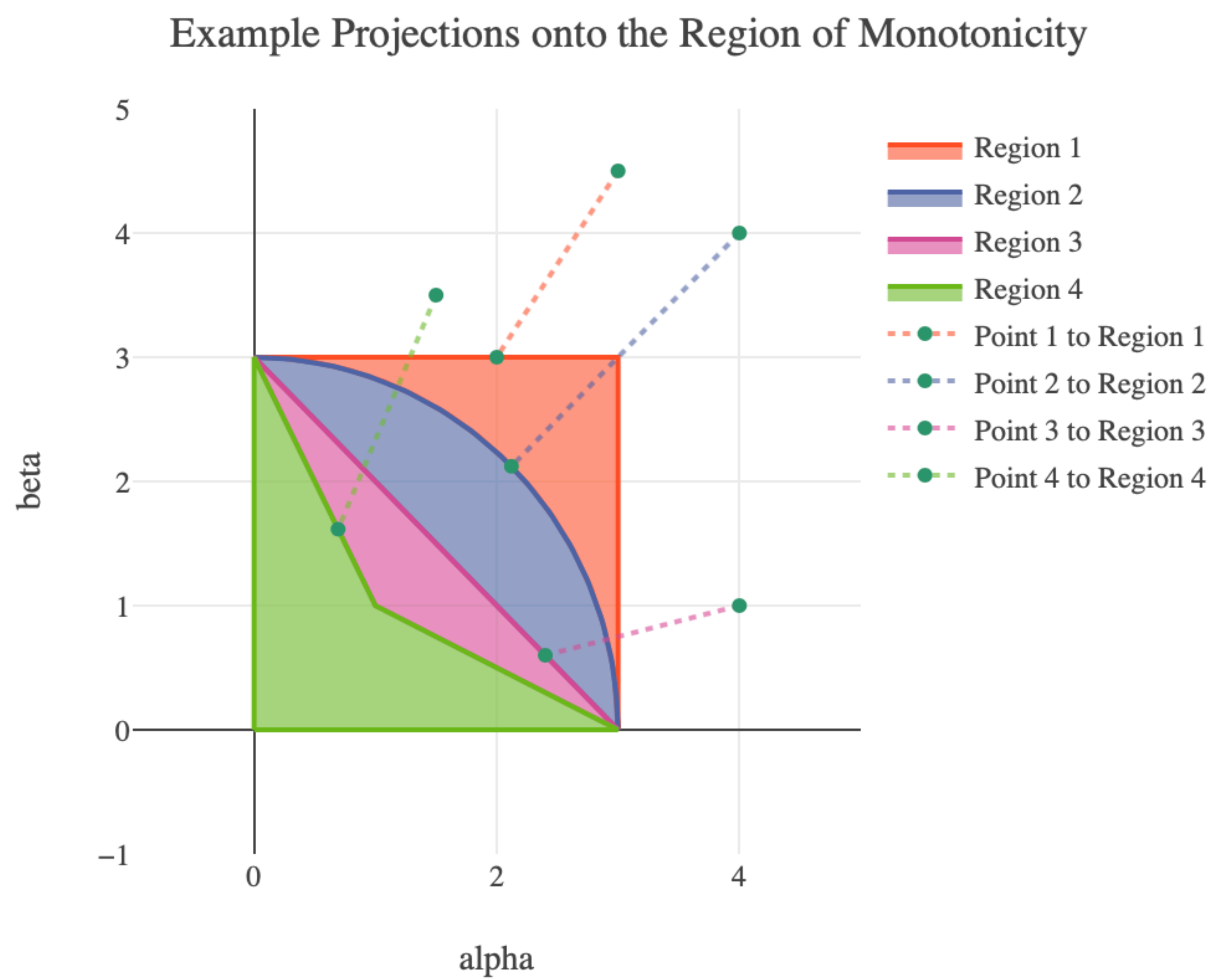


PCHIP Spline Construction



COMPUTER SCIENCE
VIRGINIA TECH

PCHIP Spline Construction



Monotone Piecewise Quintic Splines

Theory exists to create splines that are composed of quintics, but no mathematical software has been produced!

G. Ulrich and L. Watson. Positivity conditions for quartic polynomials. SIAM Journal on Scientific Computing, 15(3):528–544, 1994. doi: 10.1137/0915035. URL https://doi. org/10.1137/0915035.

Walter Hess and Jochen W Schmidt. Positive quartic, monotone quintic c2-spline interpolation in one and two dimensions. Journal of Computational and Applied Mathematics, 55(1): 51–67, 1994. doi: 10.1016/0377-0427(94)90184-8.

Dougherty, Randall L., Alan S. Edelman, and James M. Hyman. Nonnegativity-, monotonicity-, or convexity-preserving cubic and quintic Hermite interpolation. Mathematics of Computation 52.186 (1989): 471-494.

Goals and Potential Impact

Produce a mathematical software for the computation of piecewise monotonic quintic interpolating splines.

Apply this technique as an improved CDF estimator for the VarSys distribution approximation test problems.

Publish the mathematical software as a TOMS algorithm.

Potential impact: This code may get incorporated into consumer computational science tools (Matlab, Mathematica, R, etc.).

Proposed Timeline

Date	Milestone
June 2019	Prototype implementation of monotone quintic polynomial *interpolant*.
August 2019	Implementation of monotone quintic polynomial *spline*.
October 2019	First draft of TOMS paper on algorithm.
December 2019	Research Defense of work.
March 2020	Submission of TOMS paper and code.
April 2020	Final defense of Ph.D.

Overview

1. The Importance and Applications of Variability
   define variability, why is it important?

2. Algorithms for Constructing Approximations
   approximation, regression and interpolation techniques

3. Naive Approximations of Variability
   mean, variance, and standard deviation prediction with IOzone

4. Box-Splines: Uses, Constructions, and Applications
   spline overview, box splines, meshes, fitting, and data sets

5. Stronger Approximations of Variability
   predicting distributions, measuring error, and tuning

6. An Error Bound for Piecewise Linear Interpolation
   theoretical results, importance, and empirical results

7. Improving Variability estimates
   empirical distributions, monotone splines, goals, and timeline