Exercise on validation

Typical random access protocol to a common channel (CSMA family)

while Message not send **do**
    Send message
    if Collision **then**
        Wait some amount of time
    end if
end while

What should be the *amount of time* ?

Protocol dimensioning

Waiting time :

- Random
- Uniform on an interval $[0, I_n]$
- Length of the interval depends on the number of collisions
- Adaptive scheme $I_{n+1} = 2 \times I_n$,
- I_0 fixed, characteristic of the protocol

Exercise on validation

Typical random access protocol to a common channel (CSMA family)

while Message not send **do**
  Send message
  if Collision **then**
    Wait some amount of time
  end if
end while

What should be the *amount of time* ?

Protocol dimensioning

Waiting time :

- Random
- Uniform on an interval $[0, I_n]$
- Length of the interval depends on the number of collisions
- Adaptive scheme $I_{n+1} = 2 \times I_n$,
- I_0 fixed, characteristic of the protocol

Exercise on validation

Typical random access protocol to a common channel (CSMA family)

while Message not send **do**
  Send message
  if Collision **then**
    Wait some amount of time
  end if
end while

What should be the *amount of time* ?

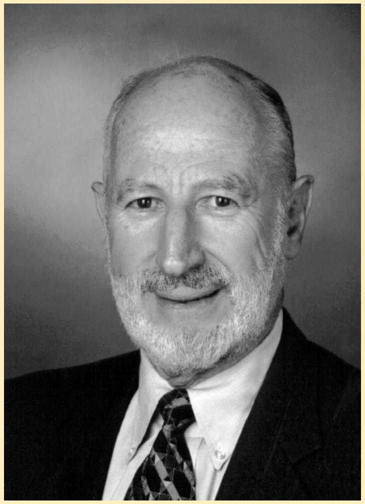
Protocol dimensioning

Waiting time :

- Random
- Uniform on an interval $[0, I_n]$
- Length of the interval depends on the number of collisions
- Adaptive scheme $I_{n+1} = 2 \times I_n$,
- I_0 fixed, characteristic of the protocol

Protocol history

University of Hawaiì 1970 `http://www.hicss.hawaii.edu/`

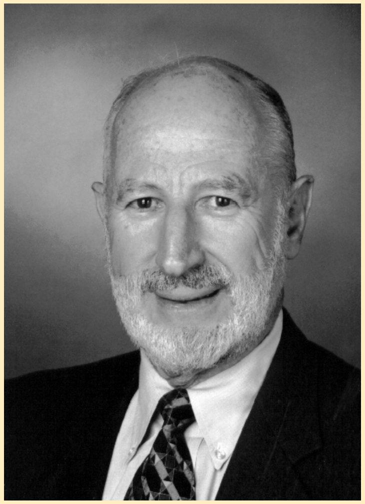


Norman Abramson et al.
Use of a radio network to provide computer communications without centralization or vacations

Ancestor of CSMA/CD (ethernet), CSMA/CA (WiFi)...

Protocol history

University of Hawaiì 1970 `http://www.hicss.hawaii.edu/`



Norman Abramson et al.
Use of a radio network to provide computer communications without centralization or vacations

Ancestor of CSMA/CD (ethernet), CSMA/CA (WiFi)...

Quantitative specification validation

Experiment

Propose an experiment to check the specification of the protocol

Estimation

How could l_0 be estimated ?

Decision

How could you conclude on the validity of the implementation of the protocol ?

Quantitative specification validation

Experiment

Propose an experiment to check the specification of the protocol

Estimation

How could l_0 be estimated ?

Decision

How could you conclude on the validity of the implementation of the protocol ?

Quantitative specification validation

Experiment

Propose an experiment to check the specification of the protocol

Estimation

How could l_0 be estimated ?

Decision

How could you conclude on the validity of the implementation of the protocol ?

Performance Evaluation

A not so Short Introduction
Analyzis of experimental results and inference

Jean-Marc Vincent[1]

[1] Laboratoire LIG, projet Inria-Mescal
UniversitéJoseph Fourier
Jean-Marc.Vincent@imag.fr

2014

Outline

Architecture comparison

Performance characterization

Distributed protocol (consensus)

- List of benchmarks (with some parameters)
- Several types of architecture

Problem: decide which architecture is the best one

Comparison of results

Decision problem

Two hypothesis :
- $\mathcal{H}_0$: (null hypothesis) A is equivalent to B
- $\mathcal{H}_1$: (alternative hypothesis) A is better than B

Decision error:
type 1 error : reject $\mathcal{H}_0$ when $\mathcal{H}_0$ is true
type 2 error : accept $\mathcal{H}_0$ when $\mathcal{H}_1$ is true.

According the observation find the decision function minimizing some risk criteria
Rejection region : if $(x_1, \cdots, x_n) \in C$ reject H_0

Danger : errors are not symmetric

Testing Normal Distributed Variables

Observations : $\mathcal{N}(m_0, \sigma_0^2))$ under hypothesis $\mathcal{H}_0$ and $\mathcal{N}(m_1, \sigma_1^2))$ under hypothesis $\mathcal{H}_1$ with $m_1 > m_0$

$$\text{Rejection region } C = \left\{ \frac{1}{n}(x_1 + \cdots + x_n) \geqslant K \right\}.$$

Computation of the rejection region type 1 error : choose α

$$\begin{aligned}
\alpha &= \mathbb{P}_{\mathcal{H}_0}(\frac{1}{n}(X_1 + \cdots + X_n) \geqslant K_\alpha) \\
&= \mathbb{P}_{\mathcal{H}_0}\left((\frac{\sqrt{n}}{\sigma}(\frac{1}{n}(X_1 + \cdots + X_n) - m_0) \geqslant \frac{\sqrt{n}}{\sigma}(K_\alpha - m_0) \right) \\
&= \mathbb{P}(Y \geqslant \frac{\sqrt{n}}{\sigma}(K_\alpha - m_0)) \text{ with } Y \sim \mathcal{N}(0, 1).
\end{aligned}$$

$$\Phi_\alpha = \frac{\sqrt{n}}{\sigma}(K_\alpha - m_0) \text{ then } K_\alpha = m_0 + \frac{\sigma}{\sqrt{n}}\Phi_\alpha.$$

Numerical example

- $\alpha = 0.05$ (a priori confidence)
  $\Phi_\alpha = 1.64$ (read on the table of the Normal distribution)
- Under $\mathcal{H}_0$, $m_0 = 6$ and $\sigma_0 = 2$
  Sample size $n = 100$

$$K_\alpha = 6 + \frac{2}{10} 1.64 = 6.33.$$

If $\frac{1}{n}(x_1 + \cdots + x_n) \geqslant 6.33$ reject $\mathcal{H}_0$ (accept $\mathcal{H}_1$), else accept $\mathcal{H}_0$

Type 2 error: Depends on the alternative hypothesis

- $m_1 = m'$ (known) σ_1 known

$$\beta = \mathbb{P}_{\mathcal{H}_1}(\frac{1}{n}(X_1 + \cdots + X_n) \leqslant K_\alpha) = \mathbb{P}(Y \leqslant \frac{\sqrt{n}}{\sigma_1}(K_\alpha - m_1)).$$

- $m_1 > m_0$ or $m_1 \neq m_0$: cannot compute

Numerical example

- $\alpha = 0.05$ (a priori confidence)
  $\Phi_\alpha = 1.64$ (read on the table of the Normal distribution)
- Under $\mathcal{H}_0$, $m_0 = 6$ and $\sigma_0 = 2$
  Sample size $n = 100$

$$K_\alpha = 6 + \frac{2}{10} 1.64 = 6.33.$$

If $\frac{1}{n}(x_1 + \cdots + x_n) \geqslant 6.33$ reject $\mathcal{H}_0$ (accept $\mathcal{H}_1$), else accept $\mathcal{H}_0$

Type 2 error: Depends on the alternative hypothesis

- $m_1 = m'$ (known) σ_1 known

$$\beta = \mathbb{P}_{\mathcal{H}_1}(\frac{1}{n}(X_1 + \cdots + X_n) \leqslant K_\alpha) = \mathbb{P}(Y \leqslant \frac{\sqrt{n}}{\sigma_1}(K_\alpha - m_1)).$$

- $m_1 > m_0$ or $m_1 \neq m_0$: cannot compute

Application example (1)

Test if algorithm 1 is better than algorithm 0

- Generate n random inputs $i_1, \cdots, i_n$
- Compute $A_0(i_k)$ $A_1(i_k)$
- $x_k = A_1(i_k) - A_0(i_k)$
- Reject the hypothesis $m = 0$ if $\frac{1}{n}(x_1 + \cdots + x_n) \geqslant K_\alpha$

Application example (2)

Test if system 1 is better than system 0

- Generate n_0 random inputs $i_1, \cdots, i_{n_0}$
- Compute $S_0(i_k)$
- Generate n_1 random inputs $i_1, \cdots, i_{n_1}$
- Compute$S_1(i_k)$
- Compute the mean difference
- Compute the standard deviation of the difference
- Reject the hypothesis $m = 0$ if $\bar{x}_1 - \bar{x}_0 \geqslant K_\alpha$

Outline

Experiment with one factor

Evaluate complexity as a function of the size of data
Response time as function of the message sizes
Load of a web server function of the number of connexion
etc

Observations

Couple (x, y) paired observations

- x predictor variable (known without error or noise)
- y response variable

Methodology

1. Plot data and analyse separately x and y (histogram, central tendency,...)
2. Plot the cloud of points (x, y)
3. Analyse the shape of the cloud
4. Propose a dependence function (fix the parameters $y = ax + b$, $y = be^{ax}$,...)
5. Give the semantic of the function
6. Give an error criteria with its semantic
7. Compute the parameters minimizing a criteria
8. Compute the confidence intervals on parameters (precision of the prediction)
9. Explain the unpredicted variance (ANOVA)
10. Analyse the result

What is a regression?

Regression analysis is the most widely used statistical tool for understanding relationships among variables. Several possible objectives including:

1. Prediction of future observations. This includes extrapolation since we all like connecting points by lines when we *expect* things to be continuous
2. Assessment of the effect of, or relationship between, explanatory variables on the response
3. A general description of data structure (generally expressed in the form of an equation or a model connecting the response or dependent variable and one or more explanatory or predictor variable)
4. Defining what you should "expect" as it allows you to define and detect what does not behave as expected

The linear relationship is the most commonly found one

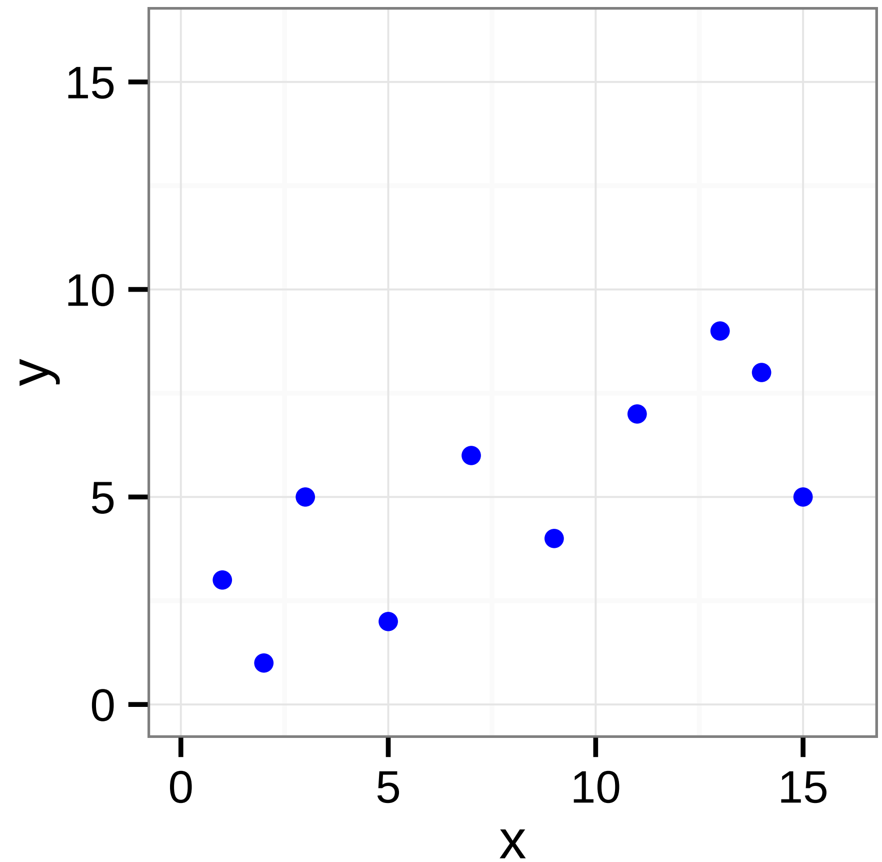
- we will illustrate how it works
- it is very general and is the basis of many more advanced tools (polynomial regression, ANOVA, . . .)

Starting With a Simple Data Set

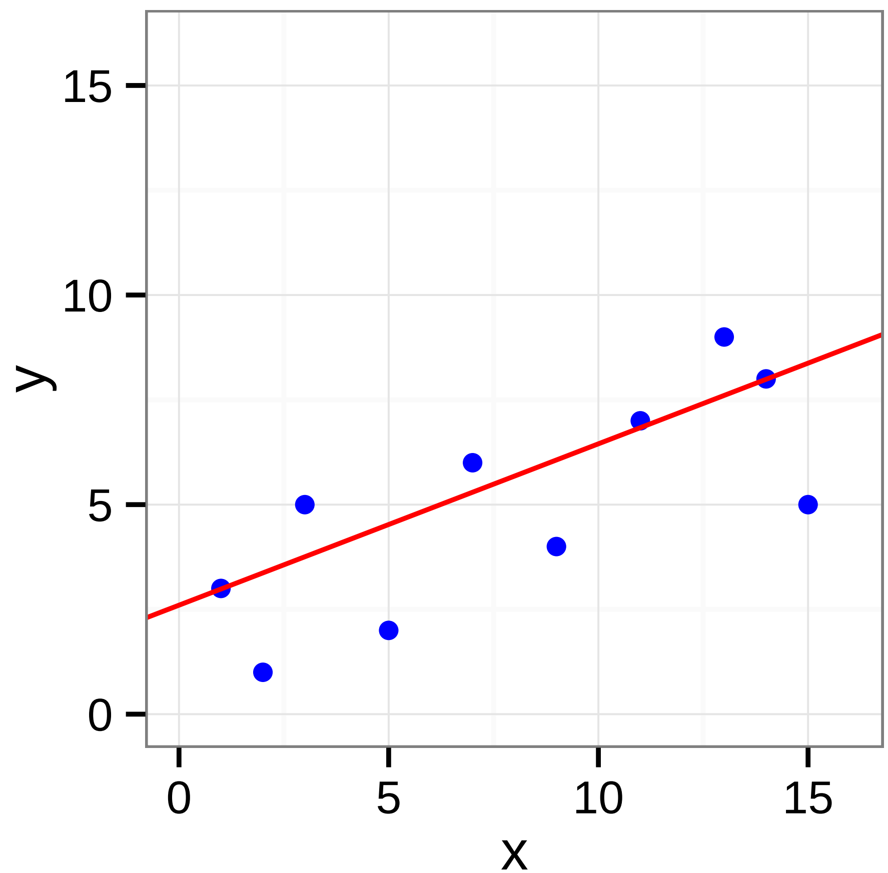
Descriptive statistics provides simple summaries about the sample and about the observations that have been made.

How could we summarize the following data set ?

	x	y
1	1.00	3.00
2	2.00	1.00
3	3.00	5.00
4	5.00	2.00
5	7.00	6.00
6	9.00	4.00
7	11.00	7.00
8	13.00	9.00
9	14.00	8.00
10	15.00	5.00

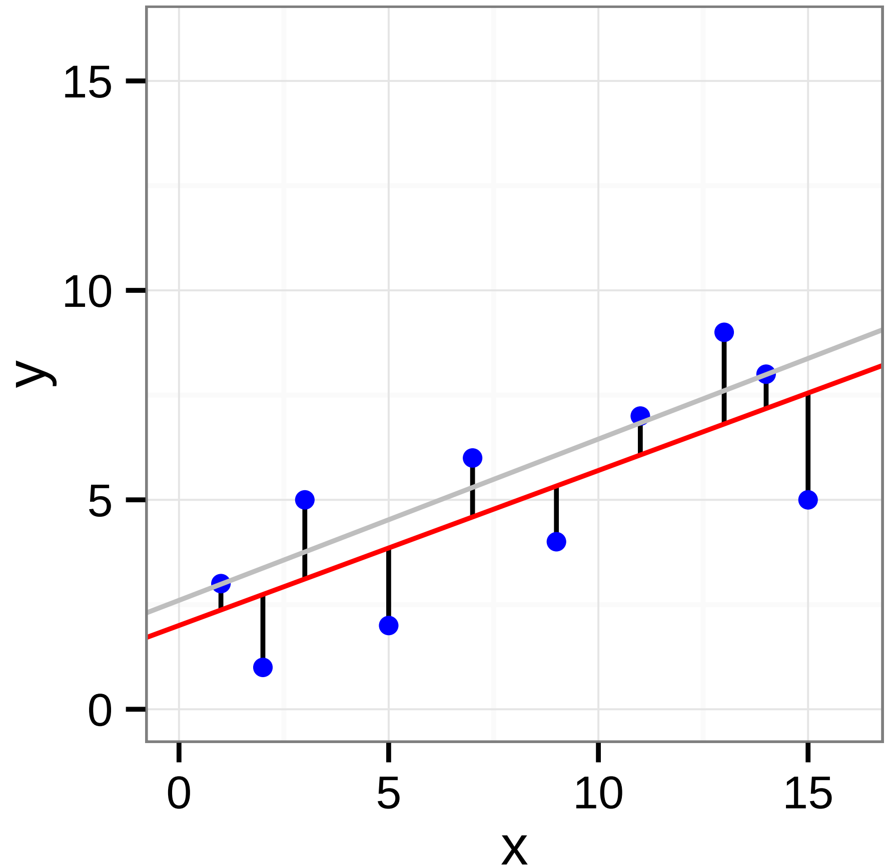


Eyeball Method

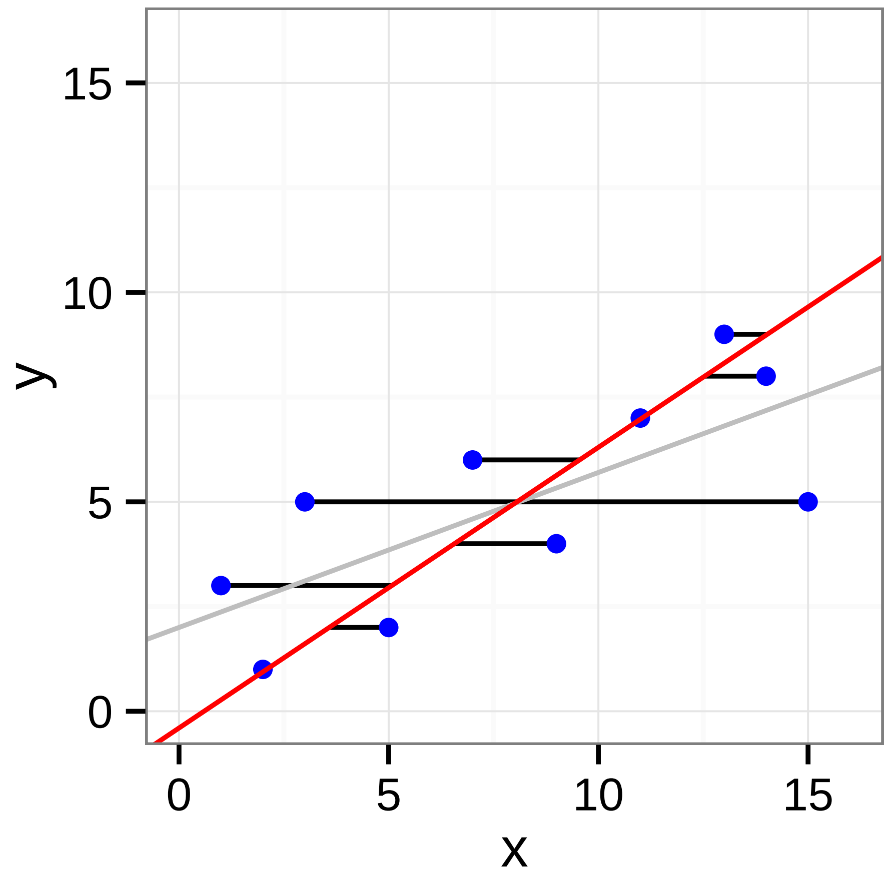


- A straight line drawn through the maximum number of points on a scatter plot balancing about an equal number of points above and below the line
- Some points are rather far from the line. Maybe we should instead try to minimize some kind of *distance to the line*

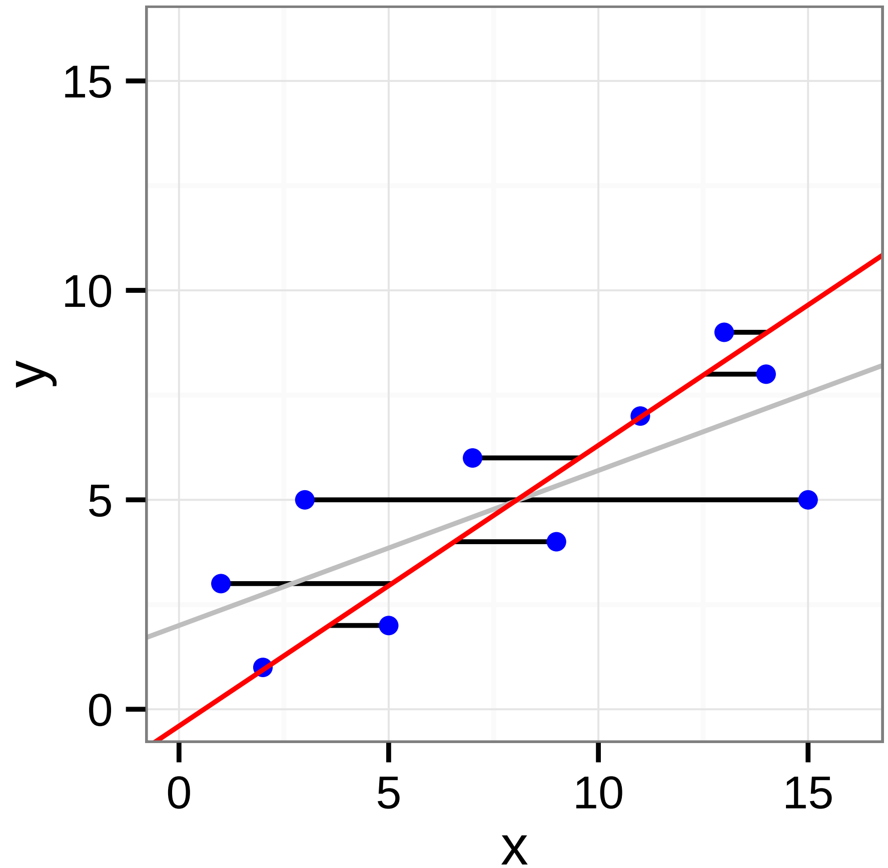
Least Squares Line (3): y as a function of x or the opposite?



Least Squares Line (3): y as a function of x or the opposite?

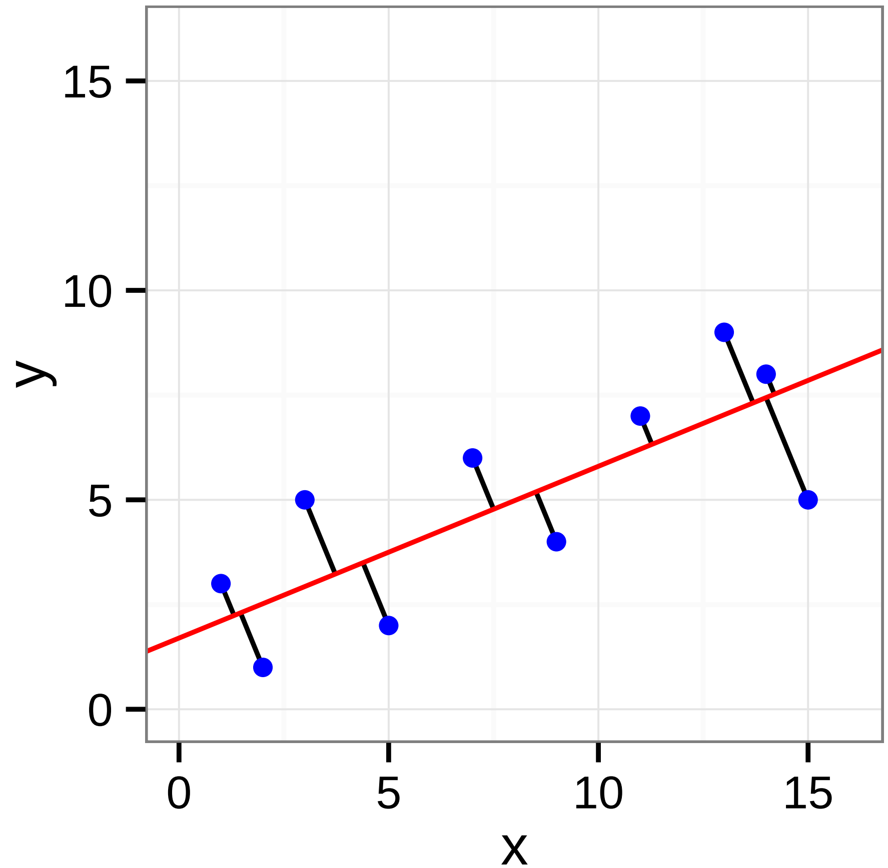


Least Squares Line (3): y as a function of x or the opposite?



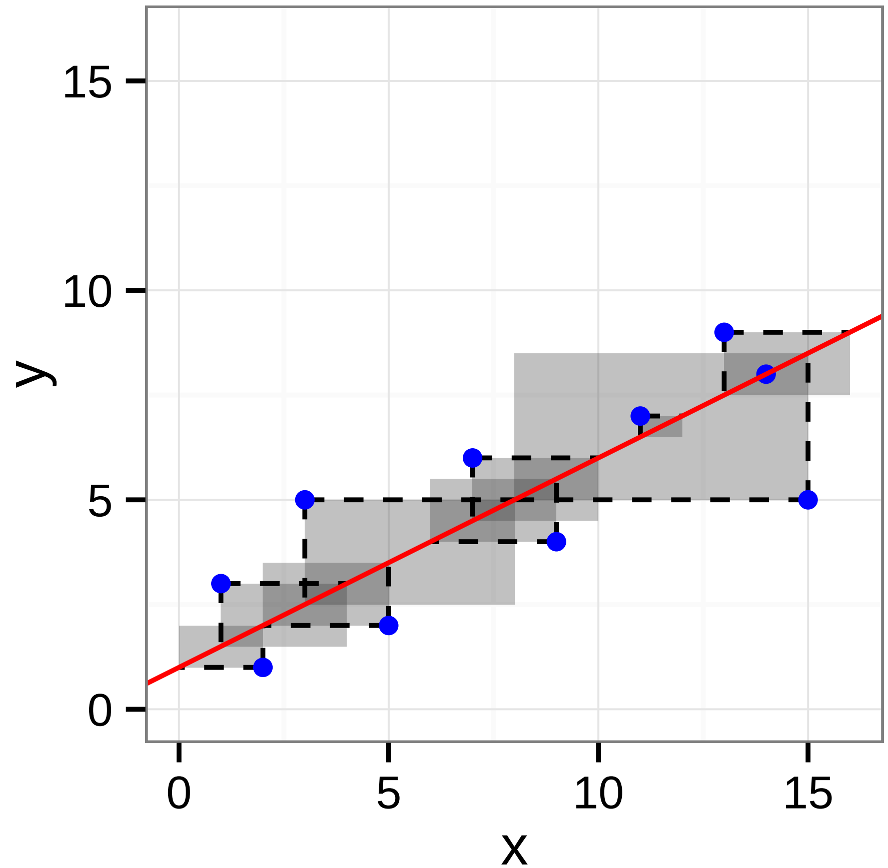
OK, do we have less asymetrical options?

Least Distances Line (a.k.a. Deming Regression)



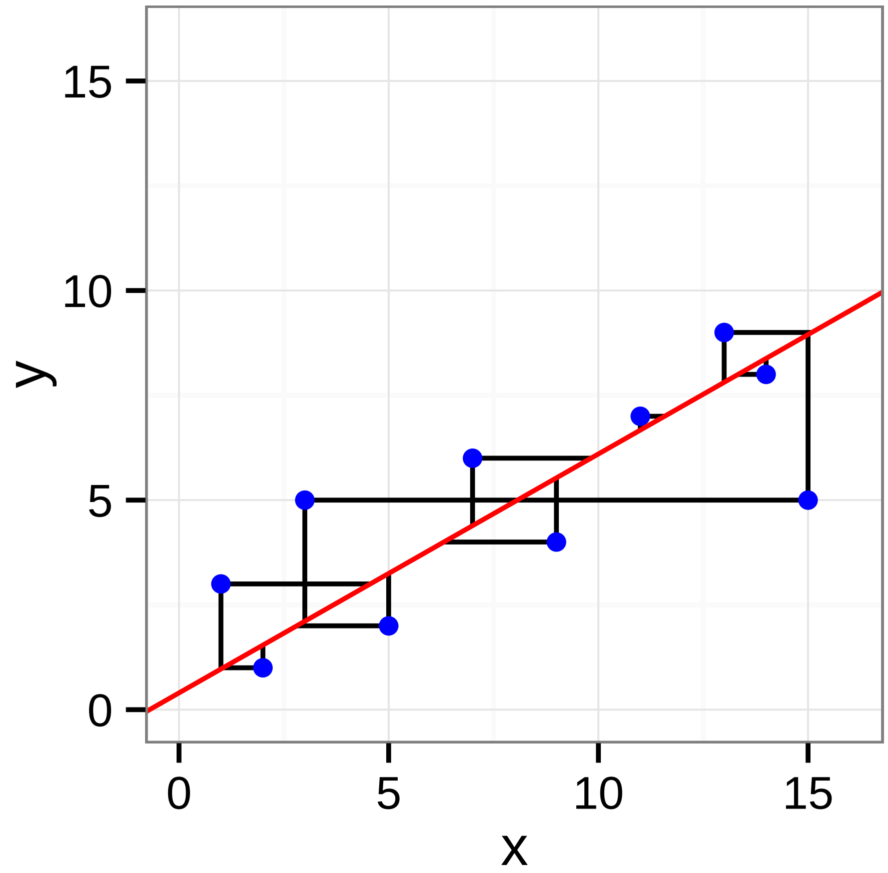
- Note that somehow, this makes sense only if we have a square plot, i.e., if x and y have the same units

Least Rectangles Line



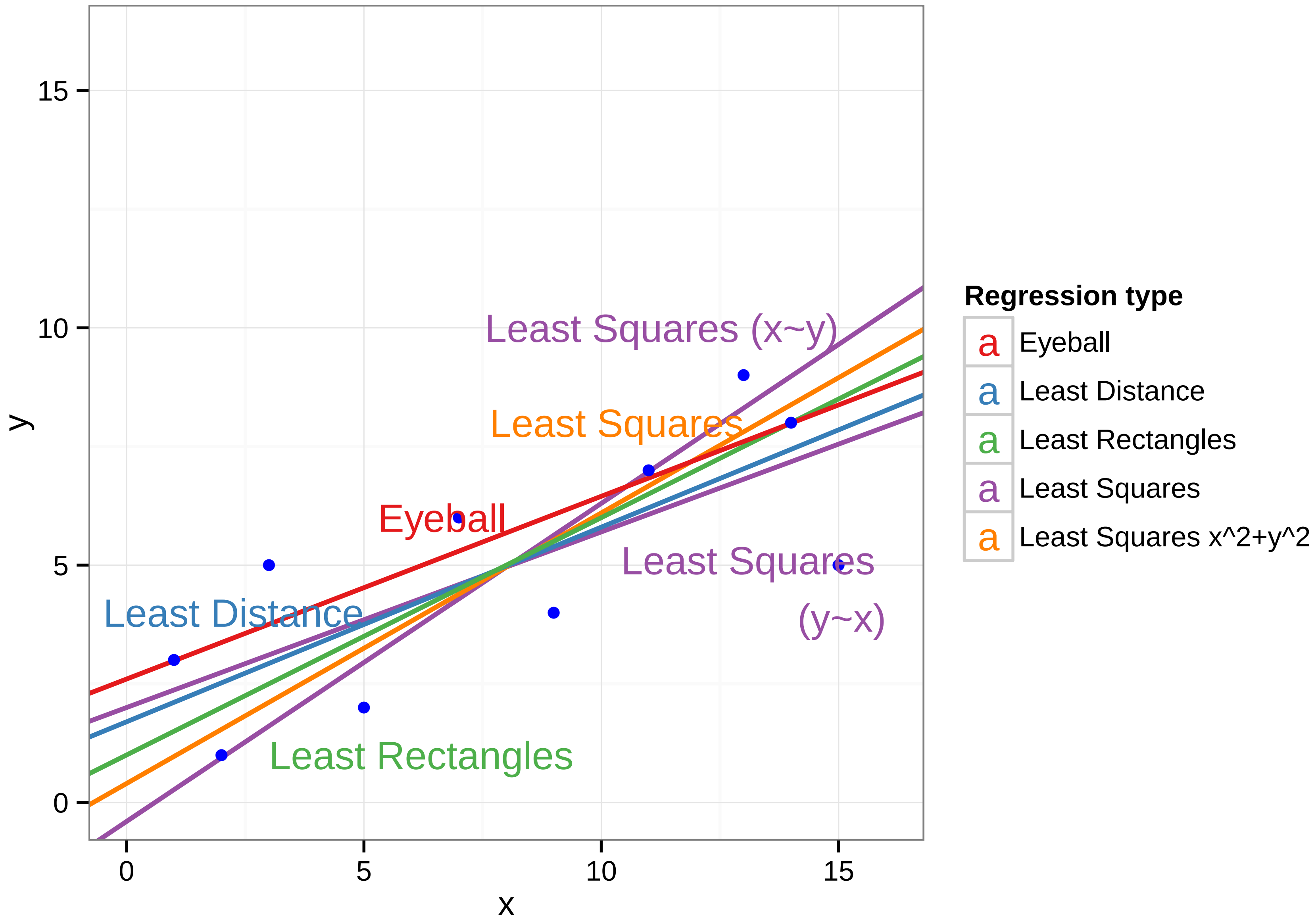
- Minimize $E(\alpha, \beta) = \sum_{i=1}^{n} \left| x_i - \frac{y_i - \alpha}{\beta} \right| \cdot |y_i - \alpha - \beta x_i|$
- This leads to the regression line $y = \frac{s_y}{s_x}(x - \bar{x}) + \bar{y}$.

Least Squares (in Both Directions) Line



- Minimize $D(\alpha, \beta) = \sum_{i=1}^{n} \left(x_i - \frac{y_i - \alpha}{\beta} \right)^2 + (y_i - \alpha - \beta x_i)^2$
- Has to be computed analytically

Which line to choose?



What does correspond to each line?

- Eyeball: AFAIK nothing
- Least Squares: classical linear regression $y \sim x$
- Least Squares in both directions: I don't know
- Deming: equivalent to Principal Component Analysis
- Rectangles: may be used when one variable is not "explained" by the other, but are inter-dependent

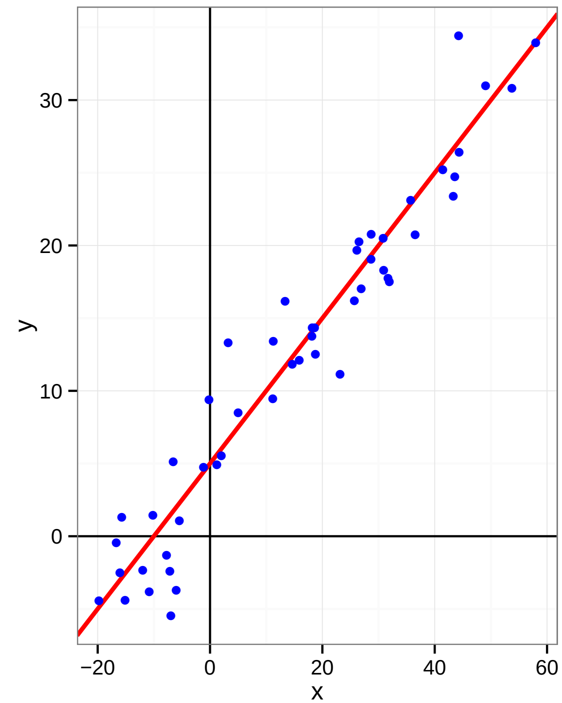
This is not just a geometric problem. You need a model of to decide which one to use

The Simple Linear Regression Model

We need to invest in a probability model

$$Y = a + bX + \varepsilon$$

- Y is the response variable
- X is a continuous explanatory variable
- a is the intercept
- b is the slope
- ε is some noise



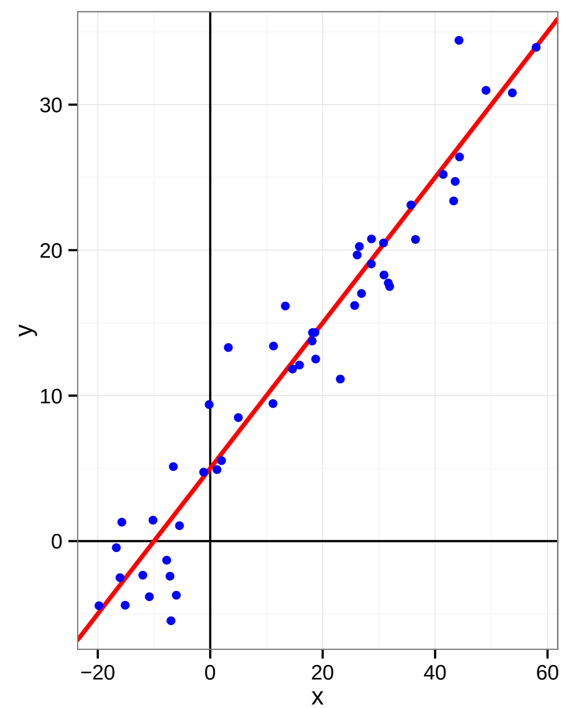
- $a + bX$ represents the "true line", the part of Y that depends on X
- The error term ε is independent "idosyncratic noise", i.e., the part of Y not associated with X

The Simple Linear Regression Model

We need to invest in a probability model

$$Y = a + bX + \varepsilon$$

- Y is the response variable
- X is a continuous explanatory variable
- a is the intercept
- b is the slope
- ε is some noise



- $a + bX$ represents the "true line", the part of Y that depends on X
- The error term ε is independent "idosyncratic noise", i.e., the part of Y not associated with X

Gauss-Markov Theorem

Under a few assumptions, the least squares regression is the best linear unbiased estimate

- $\mathbb{E}(\hat{\beta}) = b$ and $\mathbb{E}(\hat{\alpha}) = a$
- $\mathrm{Var}(\hat{\beta})$ and $\mathrm{Var}(\hat{\alpha})$ are minimal

Multiple explanatory variables

- The same results hold true when there are several explanatory variables:
$$Y = a + b^{(1)}X^{(1)} + b^{(2)}X^{(2)} + b^{(1,2)}X^{(1)}X^{(2)} + \varepsilon$$
The least squares regressions are good estimators of a, $b^{(1)}$, $b^{(2)}$, $b^{(1,2)}$

- We can use an arbitrary linear combination of variables, hence
$$Y = a + b^{(1)}X + b^{(2)}\frac{1}{X} + b^{(3)}X^3 + \varepsilon$$
is also a linear model
- Obviously the closed-form formula are much more complicated but softwares like R handle this very well

Linear regression

Theoretical model

(X, Y) follows a correlation model

$$Y = \alpha X + \beta + \epsilon;$$

with ϵ a white noise $\epsilon \sim \mathcal{N}(0, .)$

Objective function

Find estimator $(\hat{a}, \hat{b})$ minimizing the SSE (sum of square errors)

$$\sum_{i=1}^{n}(y_i - ax_i - b)^2 = \sum_{i=1}^{n} e_i^2.$$

$e_i = y_i - ax_i - b$ is the error prediction when the coefficients are a and b
$(\hat{a}, \hat{b})$ is the estimator of (α, β) minimizing SSE

Coefficients estimation

Statistics

- Empirical mean of x: $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$.
- Empirical mean of y: $\overline{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$.
- Empirical variance of x: $S_X^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2 = \overline{x^2} - \overline{x}^2$.
- Empirical variance of y: $S_Y^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \overline{y})^2 = \overline{y^2} - \overline{y}^2$.
- Empirical Covariance of (x, y): $S_{XY} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y}) = \overline{x \cdot y} - \overline{x} \cdot \overline{y}$.

Estimators

$$y_i = \frac{S_{XY}}{S_X^2}(x_i - \overline{x}) + \overline{y}$$

$$\hat{a} = \frac{S_{XY}}{S_X^2} \text{ and } \hat{b} = \overline{y} - \frac{\overline{x} \cdot S_{XY}}{S_X^2} = \overline{y} - \hat{a} \cdot \overline{x}$$

Coefficients estimation

Statistics

- Empirical mean of x: $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$.
- Empirical mean of y: $\overline{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$.
- Empirical variance of x: $S_X^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2 = \overline{x^2} - \overline{x}^2$.
- Empirical variance of y: $S_Y^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \overline{y})^2 = \overline{y^2} - \overline{y}^2$.
- Empirical Covariance of (x, y): $S_{XY} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y}) = \overline{x \cdot y} - \overline{x} \cdot \overline{y}$.

Estimators

$$y_i = \frac{S_{XY}}{S_X^2}(x_i - \overline{x}) + \overline{y}$$

$$\hat{a} = \frac{S_{XY}}{S_X^2} \text{ and } \hat{b} = \overline{y} - \frac{\overline{x} \cdot S_{XY}}{S_X^2} = \overline{y} - \hat{a} \cdot \overline{x}$$

Error analysis

Total error :

$$SST = \sum_{i=1}^{n}(y_i - \overline{y})^2 = \sum_{i=1}^{n} y_i^2 - n\overline{y}^2 = SSY - SS0.$$

Prediction error:

$$SSE = \sum_{i=1}^{n}(y_i - \hat{a}x_i - \hat{b})^2 = n(\overline{y^2} - \hat{b}\overline{y} - \hat{a}\overline{x \cdot y}$$

Residual error (that has not been predicted): $SSR = SST - SSE$
Determination coefficient:

$$R^2 = \frac{SSR}{SST}$$

Prediction quality

- $R^2 = 1$ perfect fit
- $R^2 = 0$ no fit

Usually we accept the model when $R^2 \geqslant 0.8$

Important Hypothesis (1)

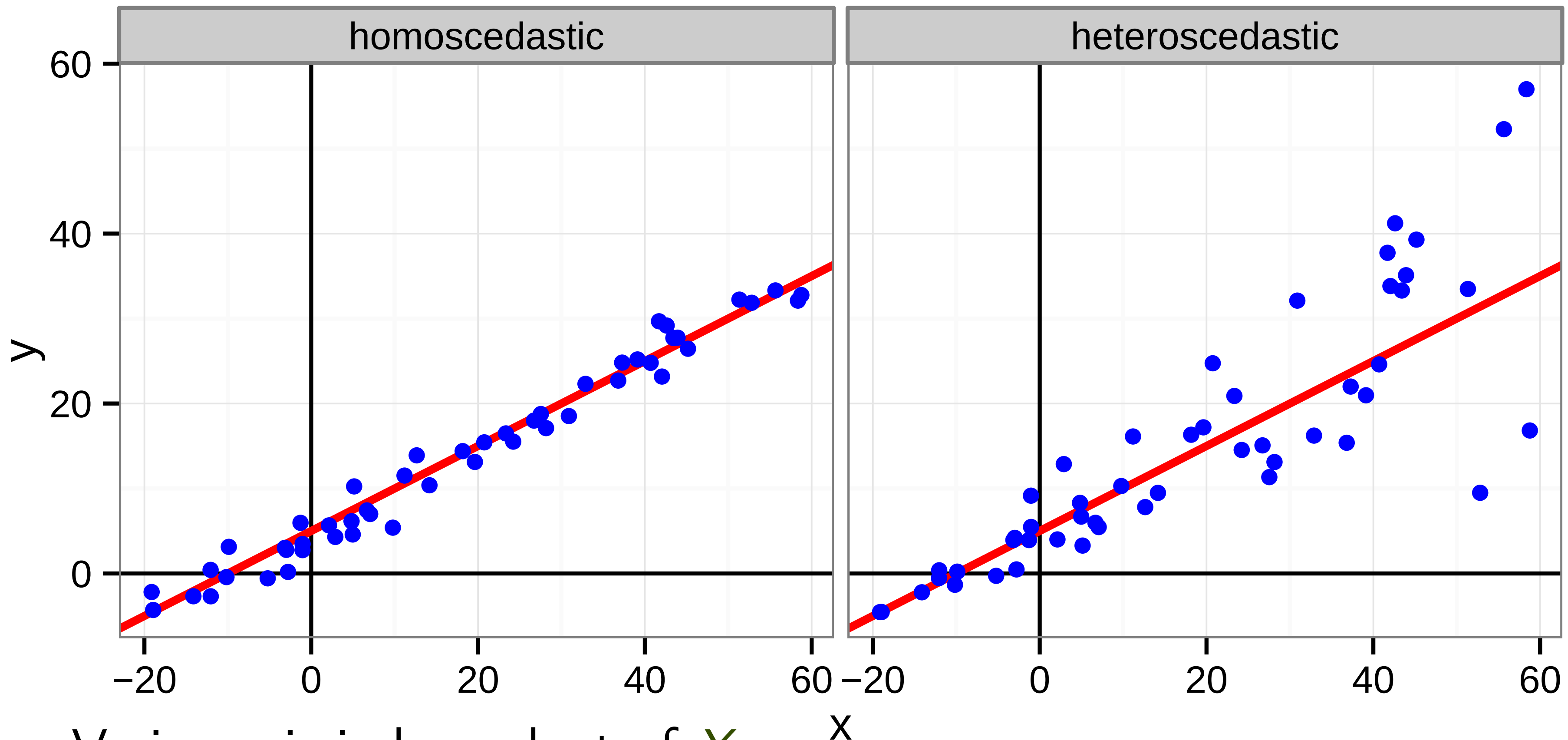
Weak exogeneity The predictor variables X can be treated as fixed values, rather than random variables: the X are assumed to be error-free, i.e., they are not contaminated with measurement errors
Although not realistic in many settings, dropping this assumption leads to significantly more difficult errors-in-variables models

Linearity the mean of the response variable is a linear combination of the parameters (regression coefficients) and the predictor variables
Since predictor variables themselves can be arbitrarily transformed, this is not that restrictive. This trick is used, for example, in polynomial regression, but beware of overfitting

Independance of Errors if several responses Y_1 and Y_2 are fit, ε_1 and ε_2 should be independant

Other Very Important Hypothesis

Constant variance (a.k.a. homoscedasticity)

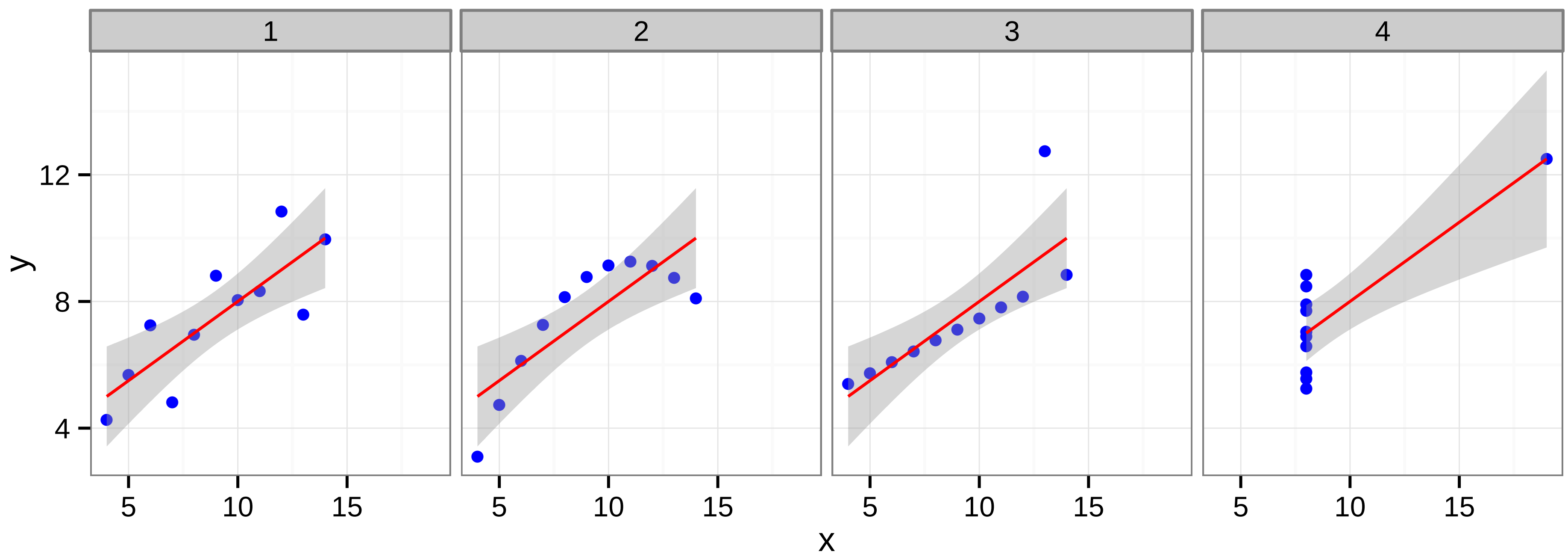


- Variance is independant of X
- If several responses Y_1 and Y_2 are fit, ε_1 and ε_2 should have the same variance
- Either normalize Y or use an other estimator

Other Classical Hypothesis (3)

Normal and iid errors This is not an assumption of the Gauss Markov Theorem. Yet, it is quite convenient to build confidence intervals of the regression

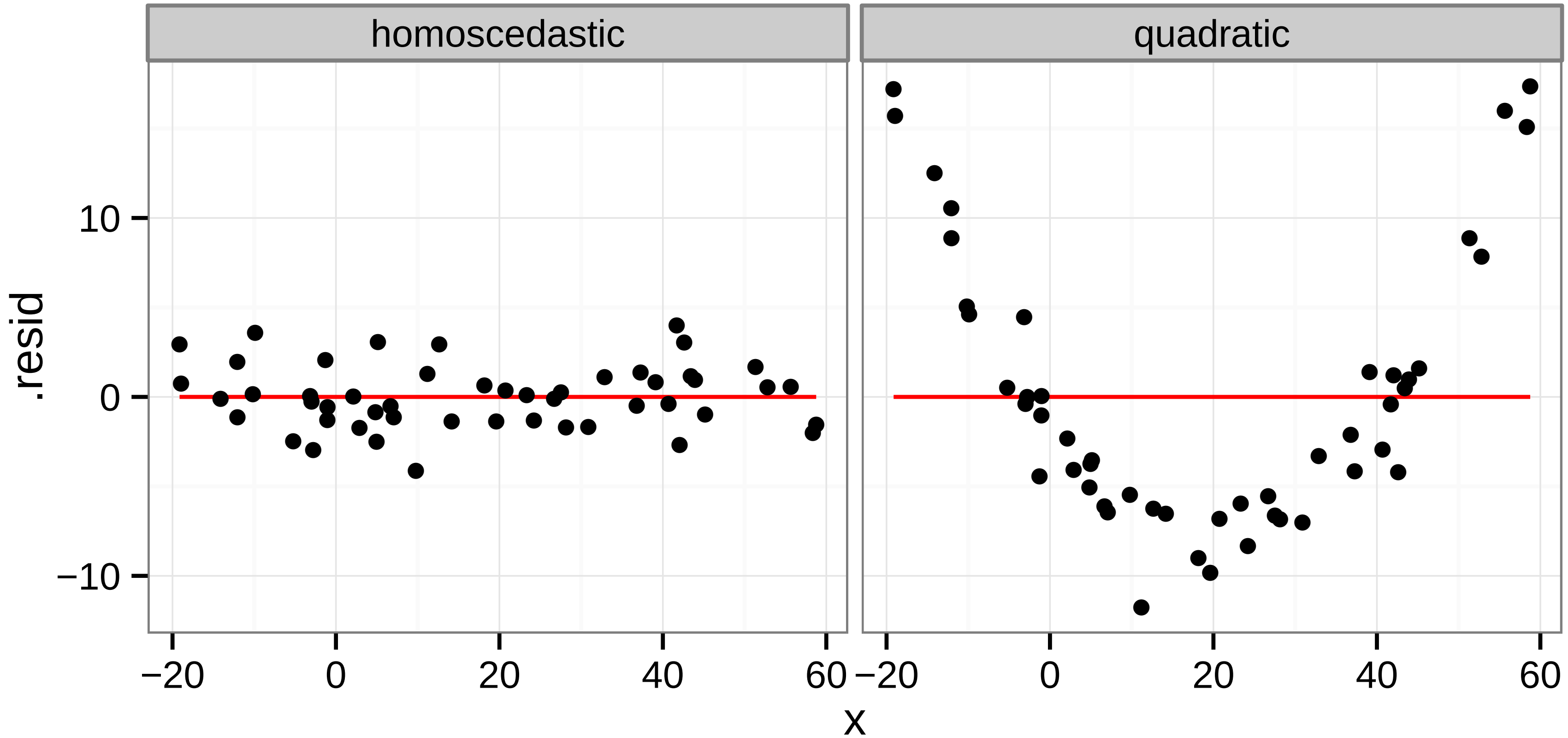
Arrangement of the predictor variables X it has a major influence on the precision of estimates of β (remember Anscombe's quartet).



This is part of your design of experiments:

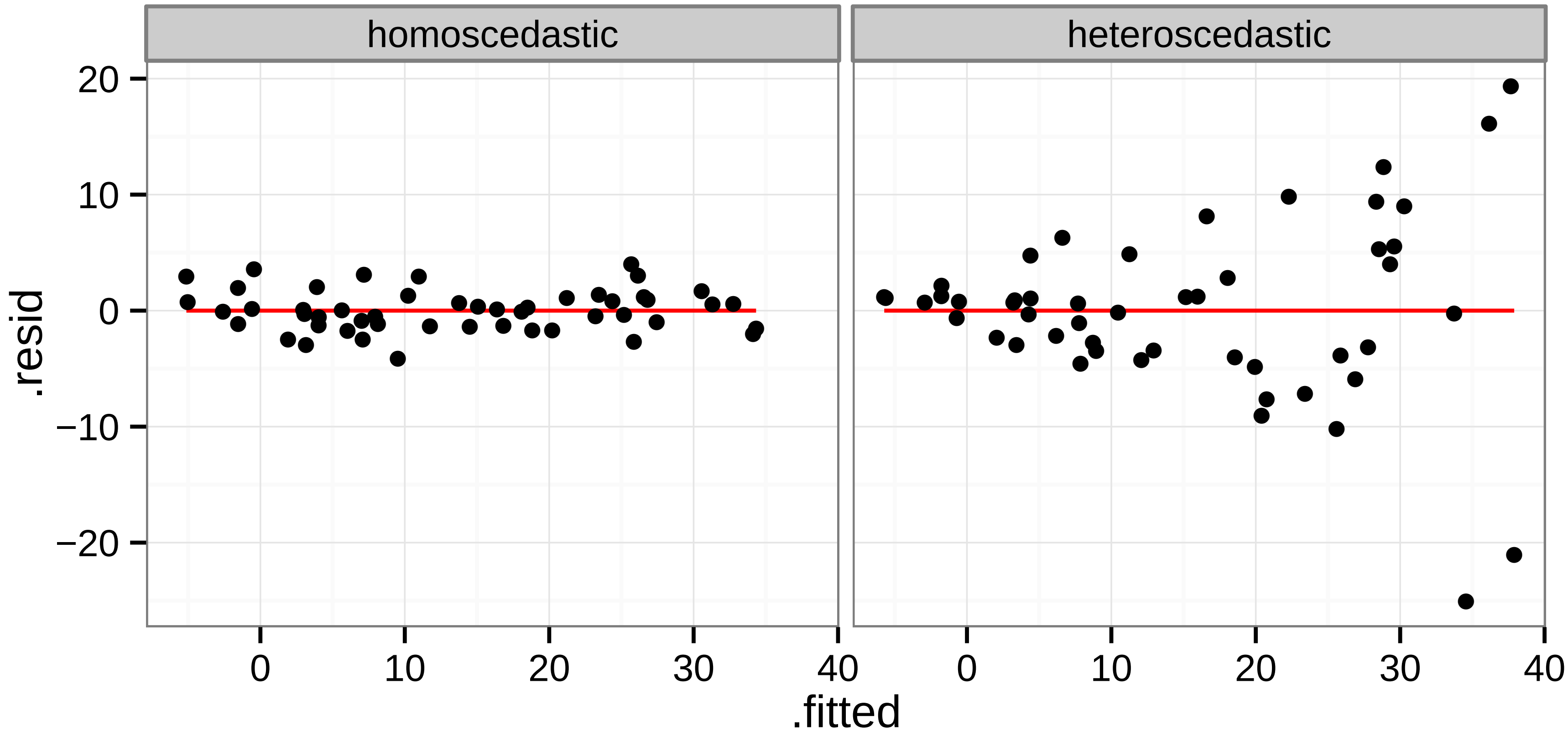
- If you want to test linearity, X should be uniformly distributed
- If you want the best estimation, you should use extreme values of X

Linearity: Residuals vs. Explanatory Variable

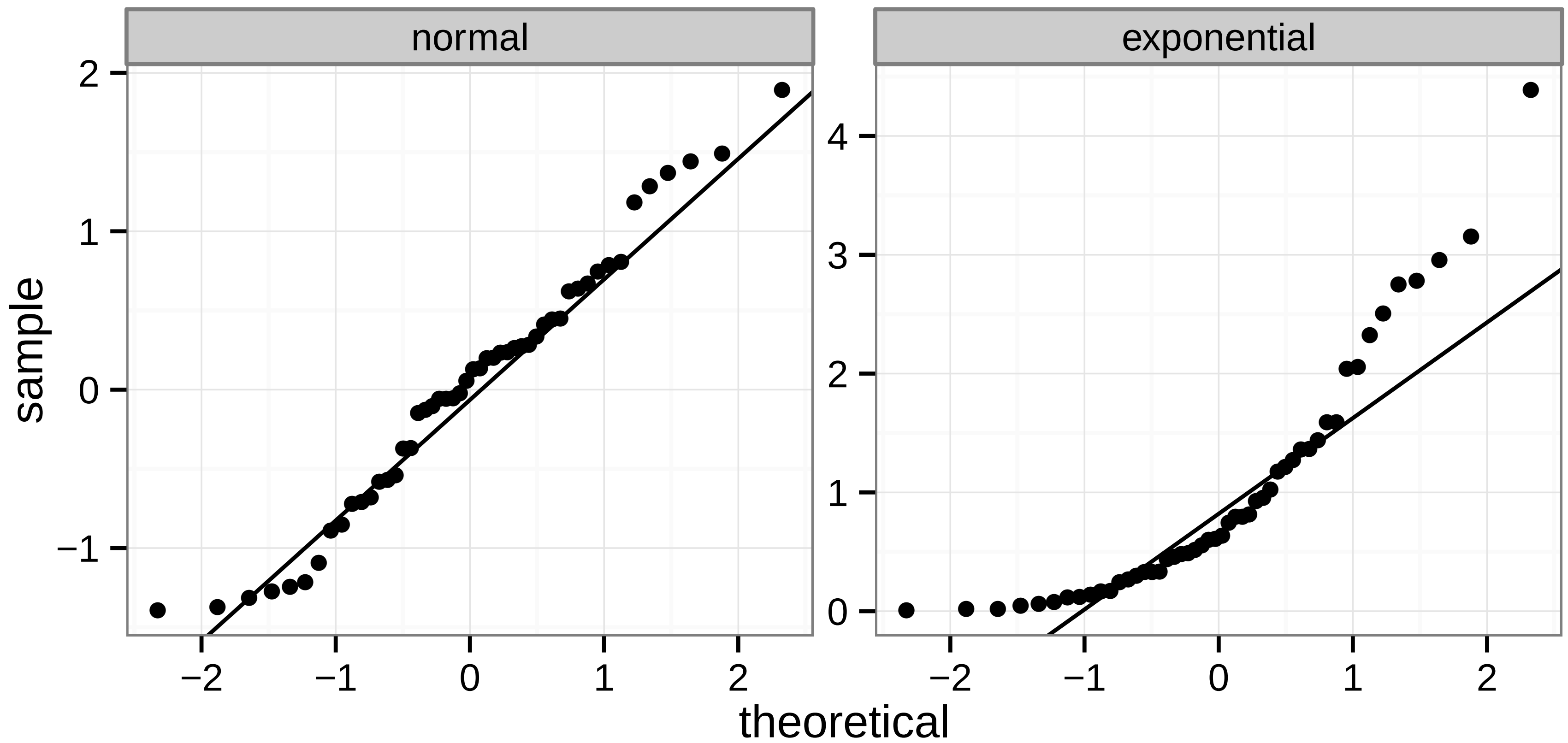


When there are several factors, you have to check for every dimension. . .

Homoscedasticity: Residuals vs. Fitted values



Normality: qqplots



A quantile-quantile plot is a graphical method for comparing two probability distributions by plotting their quantiles against each other

Model Formulae in R

The structure of a model is specified in the formula like this:

response variable ~ explanatory variable(s)

~ reads "is modeled as a function of " and `lm(y~x)` means $y = \alpha + \beta x + \varepsilon$

On the right-hand side, on should specify how the explanatory variables are combined. The symbols used here have a different meaning than in arithmetic expressions

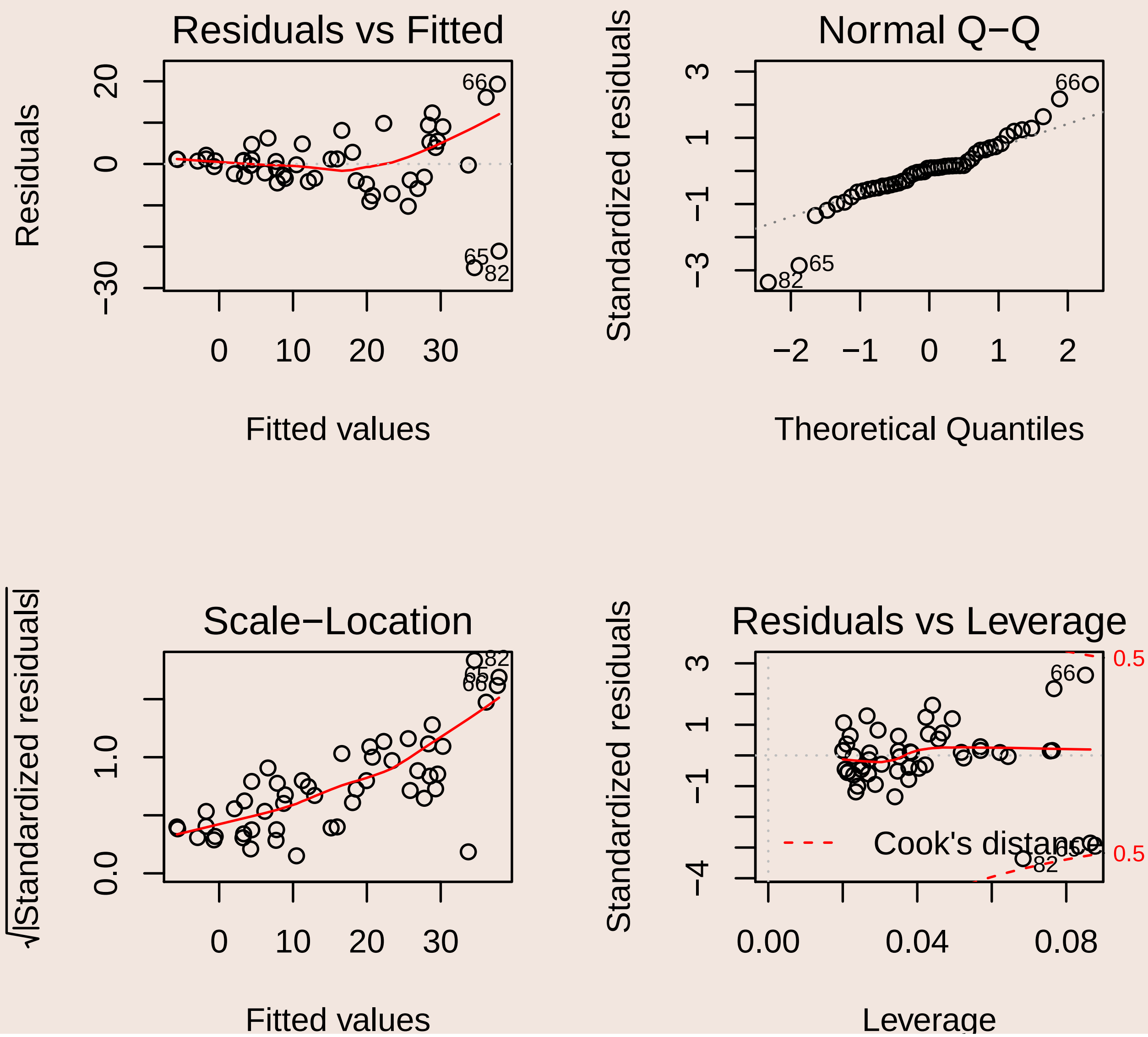
- + indicates a variable inclusion (not an addition)
- - indicates a variable deletion (not a substraction)
- * indicates inclusion of variables and their interactions
- : means an interaction

Therefore

- `z~x+y` means $z = \alpha + \beta_1 x + \beta_2 y + \varepsilon$
- `z~x*y` means $z = \alpha + \beta_1 x + \beta_2 y + \beta_3 xy + \varepsilon$
- `z~(x+y)^2` means the same
- `log(y)~I(1/x)+x+I(x^2)` means $z = \alpha + \beta_1 \times \frac{1}{x} + \beta_2 x + \beta_3 x^2 + \varepsilon$

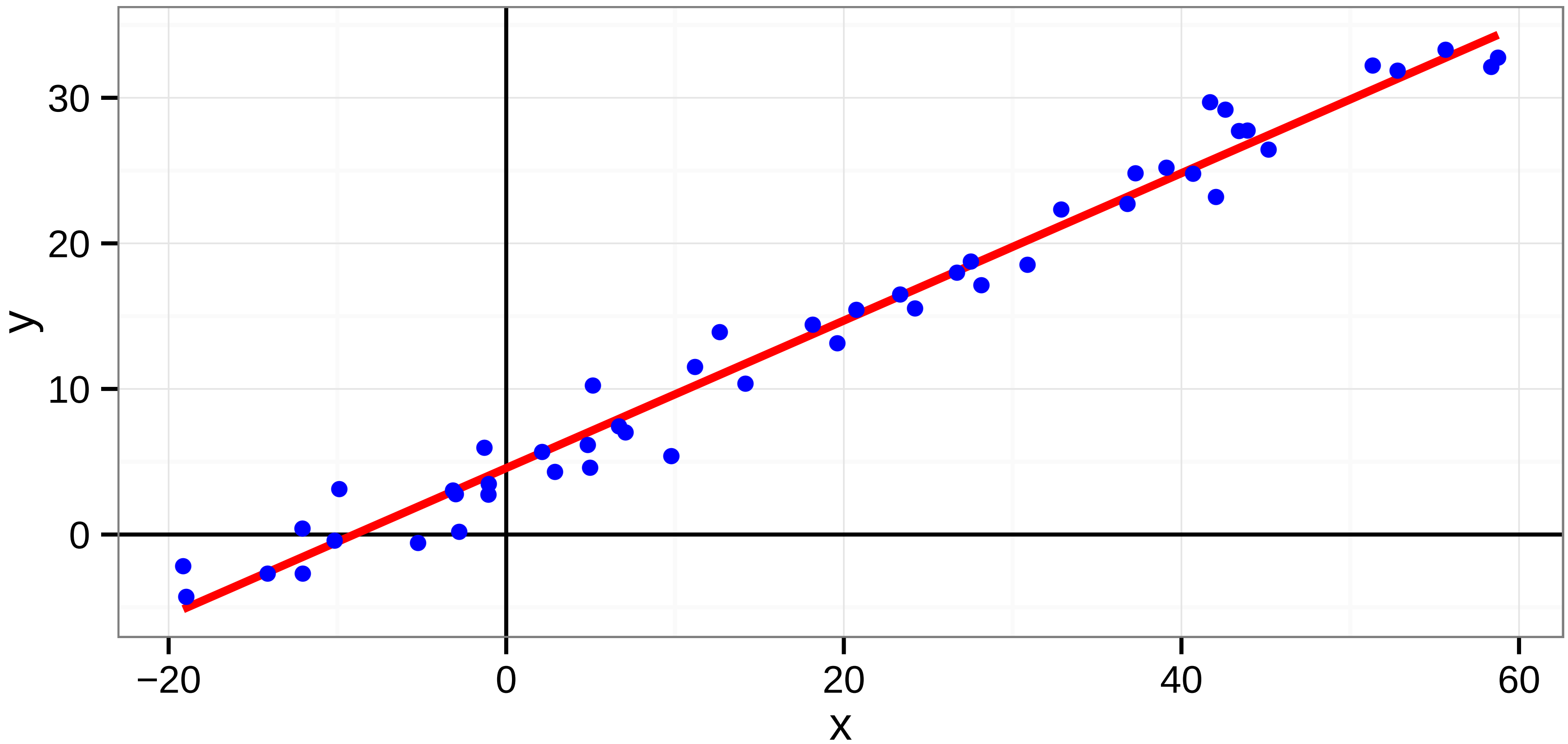
Checking the model with R

```
reg <- lm(data=df[df$type=="heteroscedastic",],y~x)
par(mfrow=c(2,2));   plot(reg);   par(mfrow=c(1,1))
```



Decomposing the Variance

How well does the least squares line explain variation in Y?



Decomposing the Variance

How well does the least squares line explain variation in Y?

Remember that $Y = \hat{Y} + \varepsilon$ ($\hat{Y}$ is the "true mean").
Since $\hat{Y}$ and ε are uncorrelated, we have

$$\text{Var}(Y) = \text{Var}(\hat{Y} + \varepsilon) = \text{Var}(\hat{Y}) + \text{Var}(\varepsilon)$$

$$\frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \bar{Y})^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\hat{Y}_i - \bar{\hat{Y}})^2 + \frac{1}{n-1}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})^2$$

Since $\bar{\varepsilon} = 0$ and $\bar{Y} = \bar{\hat{Y}}$, we have

$$\underbrace{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}_{\text{Total Sum of Squares}} = \underbrace{\sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2}_{\text{Regression SS}} + \underbrace{\sum_{i=1}^{n}\bar{\varepsilon}^2}_{\text{Error SS}}$$

- SSR = Variation in Y explained by the regression line
- SSE = Variation in Y that is left unexplained

$$SSR = SST \Rightarrow \text{perfect fit}$$

A Goodness of Fit Measure: R^2

The coefficient of determination, denoted by R^2, measures goodness of fit:

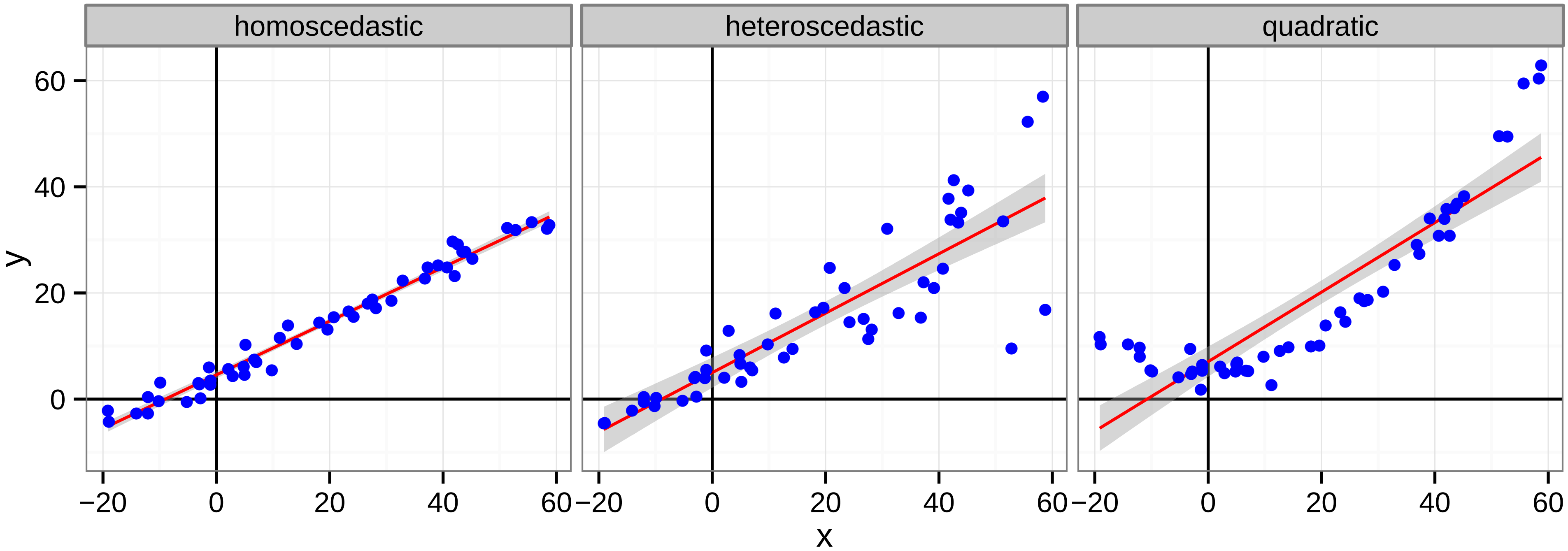
$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

- $0 \leqslant R^2 \leqslant 1$
- The closer R^2 is to 1, the better the fit

Warning:

- A not so low R^2 may mean important noise or bad model
- As you add parameters to a model, you inevitably improve the fit. There is a trade-off beteween model simplicity and fit. Strive for simplicity!

Illustration with R (homoscedastic data)



Conclusion

1. You need a model to perform your regression
2. You need to check whether the underlying hypothesis of this model are reasonable or not

This model will allow you to:

1. Assess and quantify the effect of parameters on the response
2. Extrapolate within the range of parameters you tried
3. Detect outstanding points (those with a high residual and/or with a high lever)

This model will guide on how to design your experiments:

- e.g., the linear model assumes some uniformity of interest over the parameter space range
- if your system is heteroscedastic, you will have to perform more measurements for parameters that lead to higher variance

Outline

Time dimensioning problems

Time out estimation

Distributed protocol (consensus)

- Crash of processes
- Variable communications (wireless network)
- Failure detection mechanism (parametrized)

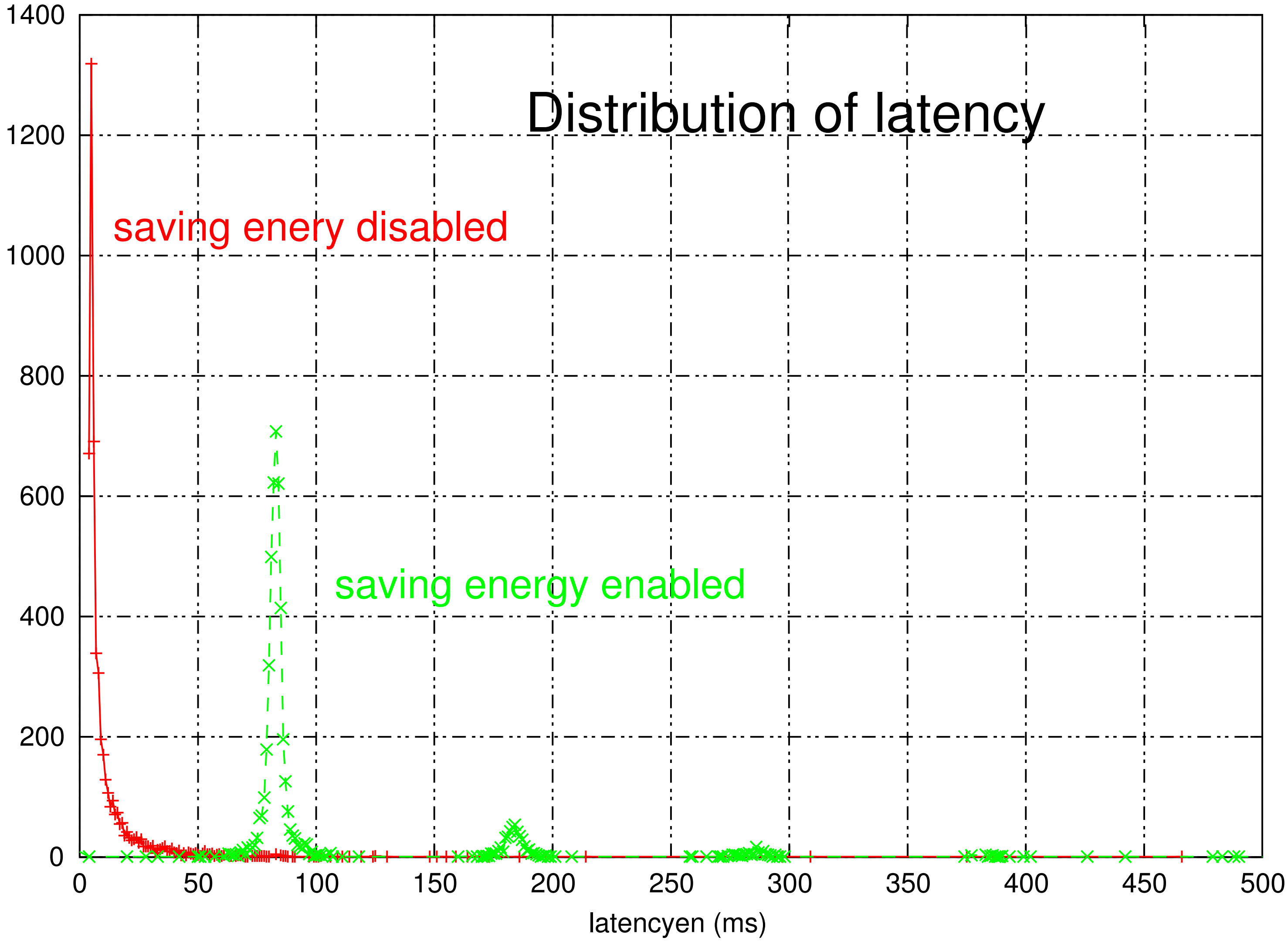
Factors

- Crash of processes
- Variable communications (wireless network)
- Failure detection mechanism (parametrized)

$\Rightarrow$ **Evaluation of the latency**

Latency estimation

PDA $\rightarrow$ PDA communication (ping)

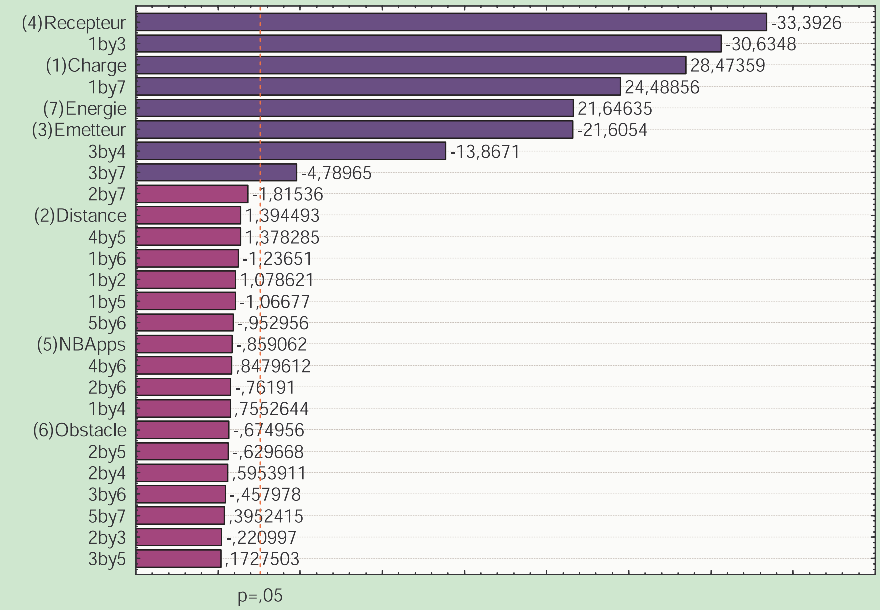


Factors Analysis

Factors (a priori)

- Distance
- Number of obstacles
- Number of nodes
- Network load
- Sender type
- Receiver type
- Saving energy

Tagushi analysis

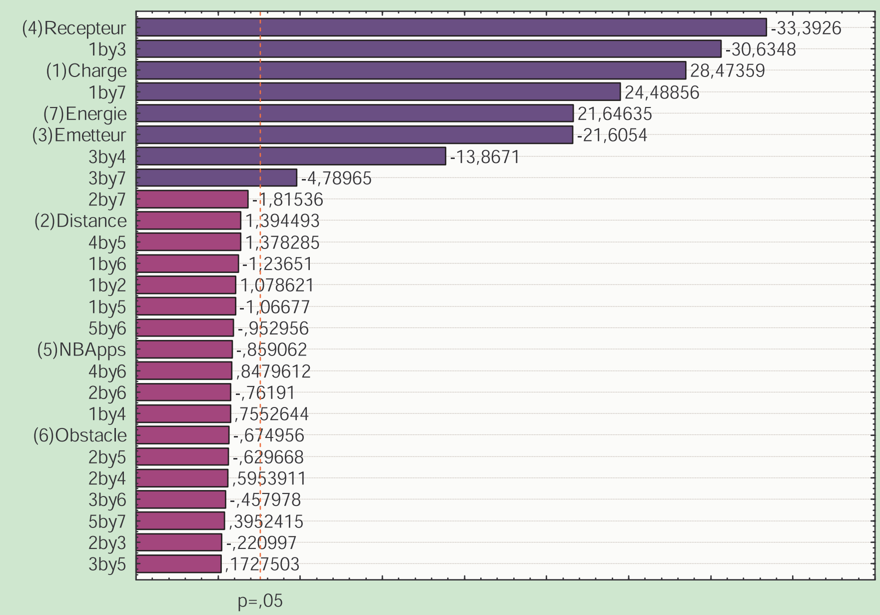


Factors Analysis

Significant factors

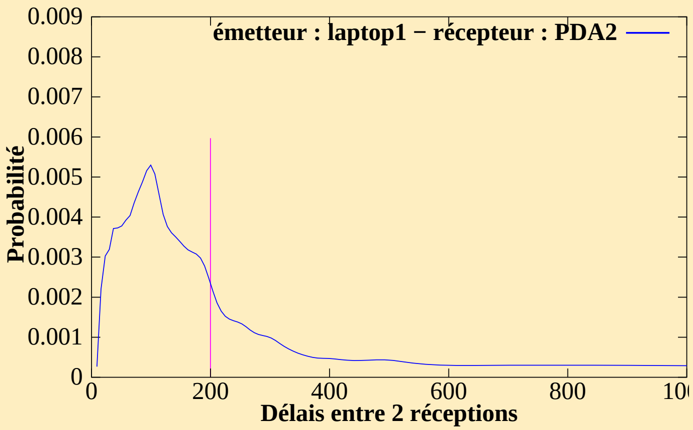
- Distance
- Number of obstacles
- Number of nodes
- **Network load (2)**
- **Sender type (4)**
- **Receiver type (1)**
- **Saving energy (3)**

Tagushi analysis

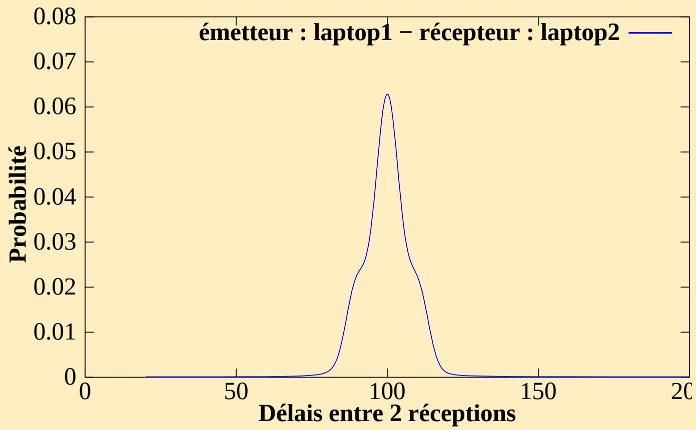


Time out estimation

Laptop $\rightarrow$ PDA



Laptop $\rightarrow$ laptop



Outline

Trace analysis example

Presentation of the paper available on `http://fta.inria.fr`

Mining for Statistical Models of Availability in Large-Scale Distributed Systems: An Empirical Study of SETI@home

Bahman Javadi[1], Derrick Kondo[1], Jean-Marc Vincent[1,2], David P. Anderson[3]

[1]Laboratoire d'Informatique de Grenoble, MESCAL team, INRIA, France
[2]University of Joseph Fourier, France
[3]UC Berkeley, USA

IEEE/ACM International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2009)

Mining for Statistical Models of Availability in Large-Scale Distributed Systems: An Empirical Study of SETI@home

Bahman Javadi[1], Derrick Kondo[1], Jean-Marc Vincent[1,2], David P. Anderson[3]

[1]Laboratoire d'Informatique de Grenoble, MESCAL team, INRIA, France
[2]University of Joseph Fourier, France
[3]UC Berkeley, USA

IEEE/ACM International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2009)

- P2P, Grid, Cloud, and Volunteer computing systems

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts
  - Uncertainty of host availability

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts
  - Uncertainty of host availability

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts
  - Uncertainty of host availability

Main Motivation

Effective Resource Selection for Stochastic Scheduling Algorithms

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts
  - Uncertainty of host availability

Main Motivation

Effective Resource Selection for Stochastic Scheduling Algorithms

Goal

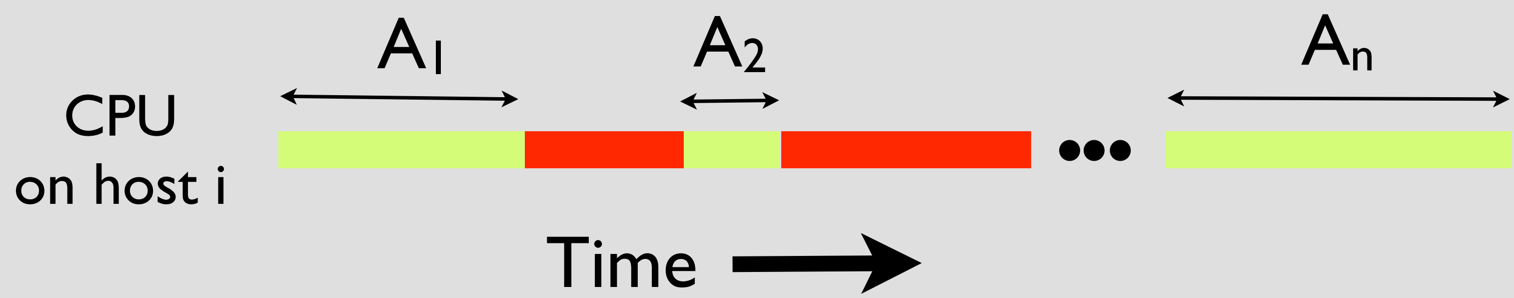
Model of host availability
(i.e., subset of hosts with the same availability distribution)

Outline

1. Introduction and Motivation
2. Measurement
   - Remove outliers
3. Modelling Process
   - Randomness Tests
   - Clustering
   - Model fitting
4. Discussions
   - Significance of Clustering Criteria
   - Scheduling Implications
5. Related Work
6. Conclusion and Future Work

Define Availability

CPU availability on each host



Length of Availability Intervals: A1, A2, ..., An

Measurement Method

BOINC

- Middleware for volunteer computing systems
- Underlying software infrastructure for projects such as SETI@home

Measurement Method

BOINC

- Middleware for volunteer computing systems
- Underlying software infrastructure for projects such as SETI@home

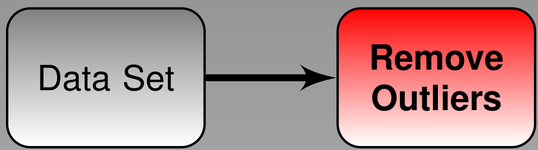
We instrumented the BOINC client to collect CPU availability traces:

- Total number of host traces: 226,208
- Collection period: April 1, 2007 - Jan 1, 2009
- Total CPU time: 57,800 years
- Number of intervals: 102,416,434
- Assume 100% or 0% availability

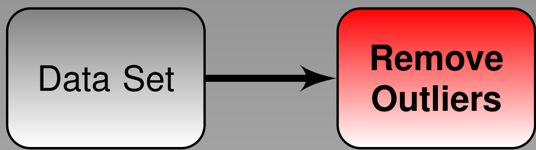
Outline

1. Introduction and Motivation
2. Measurement
   - Remove outliers
3. Modelling Process
   - Randomness Tests
   - Clustering
   - Model fitting
4. Discussions
   - Significance of Clustering Criteria
   - Scheduling Implications
5. Related Work
6. Conclusion and Future Work

Outliers



Outliers

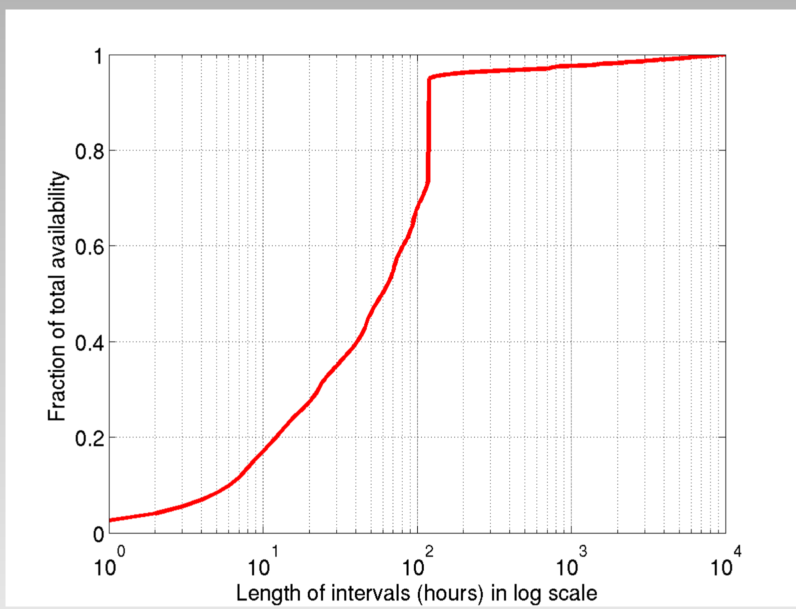


Check for outliers: Artifacts resulted from a benchmark run periodically every five days

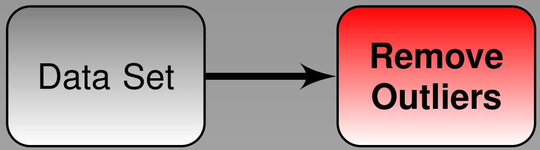
Outliers

Data Set → **Remove Outliers**

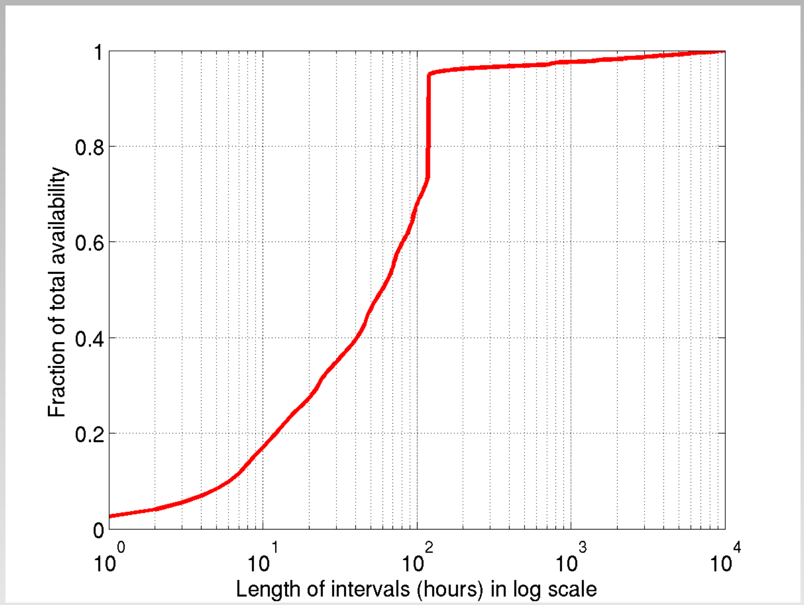
Check for outliers: Artifacts resulted from a benchmark run periodically every five days

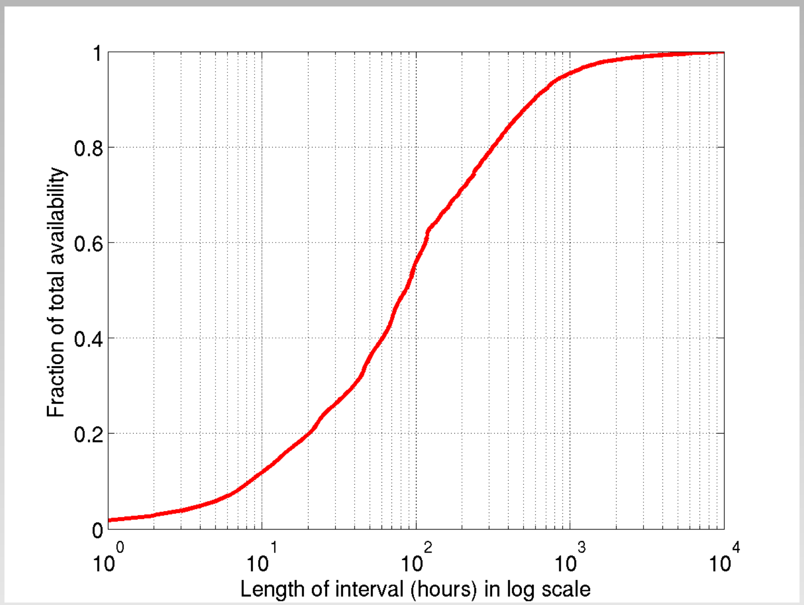


Outliers



Check for outliers: Artifacts resulted from a benchmark run periodically every five days

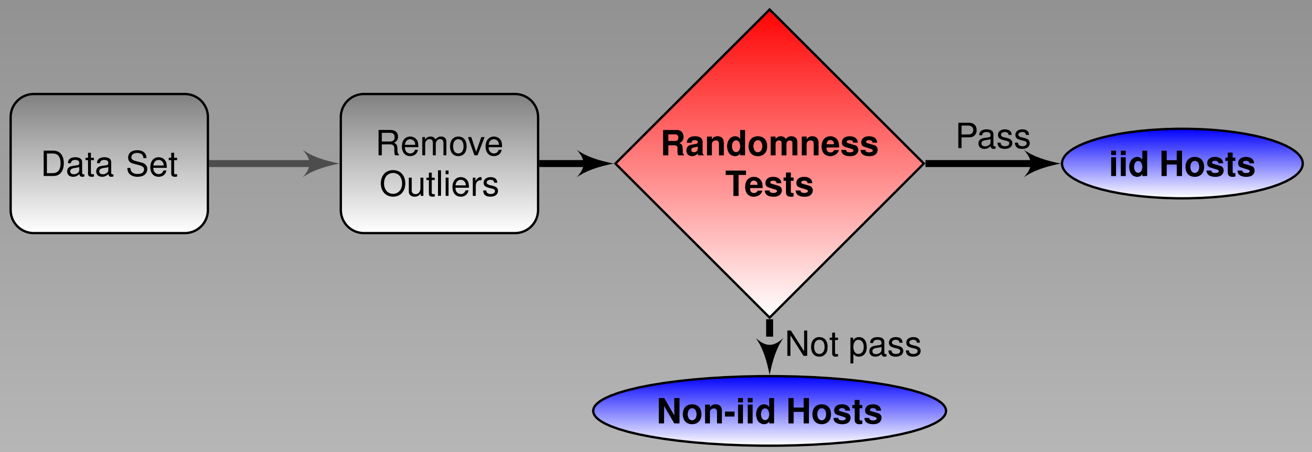




Outline

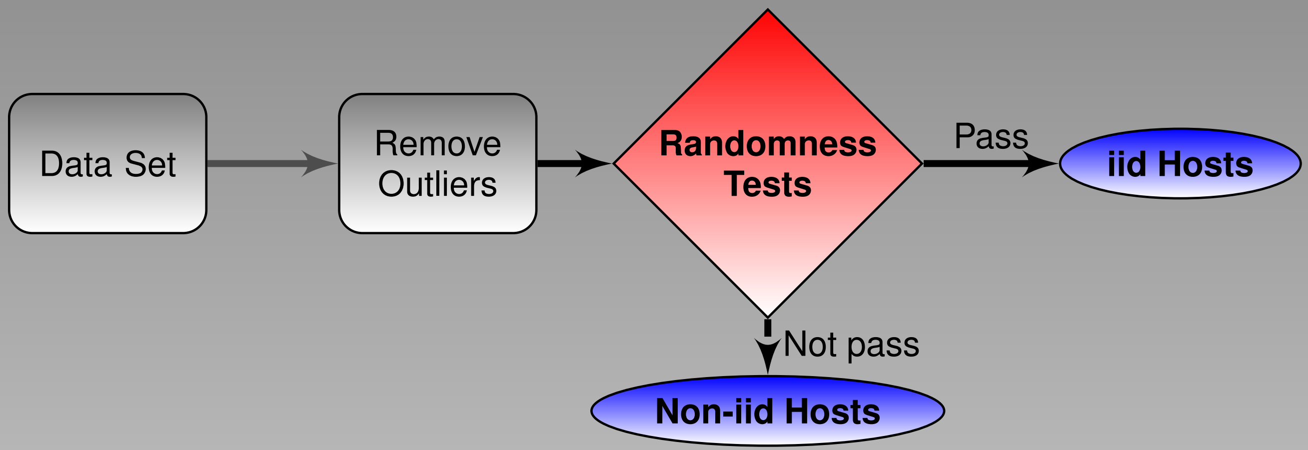
1. Introduction and Motivation
2. Measurement
   - Remove outliers
3. Modelling Process
   - Randomness Tests
   - Clustering
   - Model fitting
4. Discussions
   - Significance of Clustering Criteria
   - Scheduling Implications
5. Related Work
6. Conclusion and Future Work

Randomness Tests



To determine which hosts have truly random availability intervals

Randomness Tests

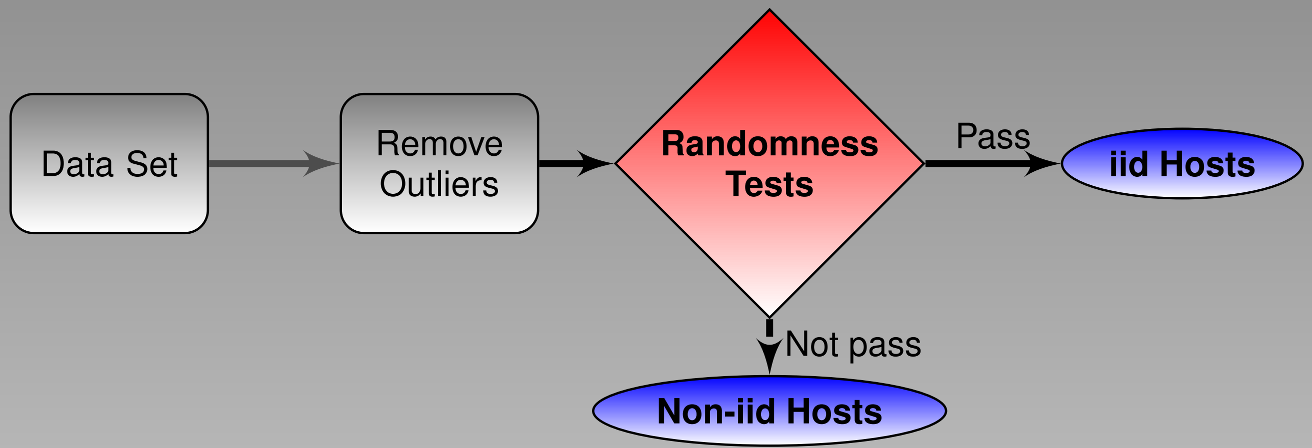


To determine which hosts have truly random availability intervals

Four well-known non-parametric tests:

- Runs test
- Runs up/down test
- Mann-Kendall test
- Autocorrelation function test (ACF)

Randomness Tests

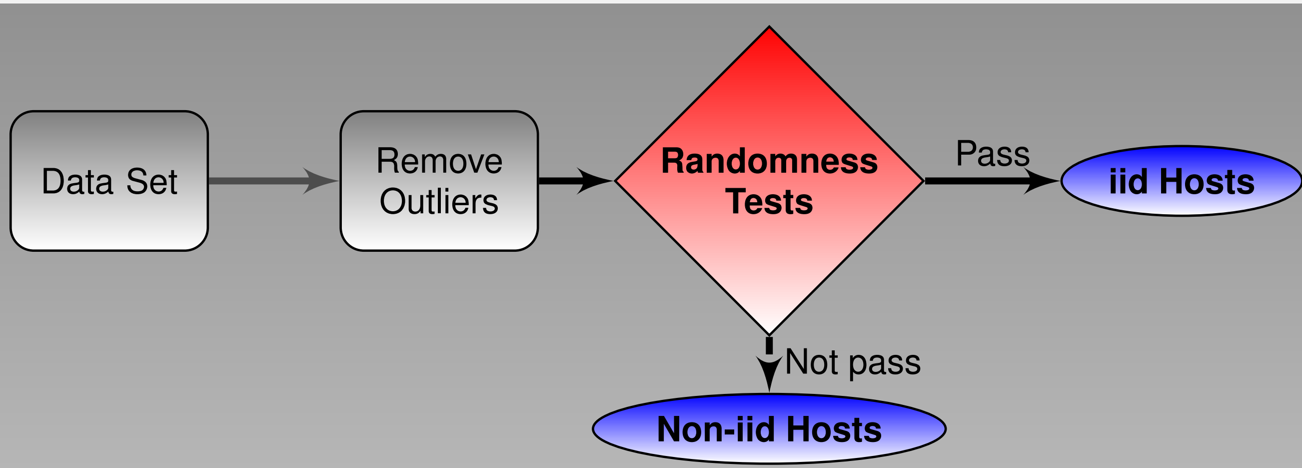


To determine which hosts have truly random availability intervals
Four well-known non-parametric tests:

- Runs test
- Runs up/down test
- Mann-Kendall test
- Autocorrelation function test (ACF)

Test	Runs std	Runs up/down	ACF	Kendall	All
# of hosts	101649	144656	109138	101462	**57757**
Fraction	0.602	0.857	0.647	0.601	**0.342**

Randomness Tests



To determine which hosts have truly random availability intervals

Four well-known non-parametric tests:

- Runs test
- Runs up/down test
- Mann-Kendall test
- Autocorrelation function test (ACF)

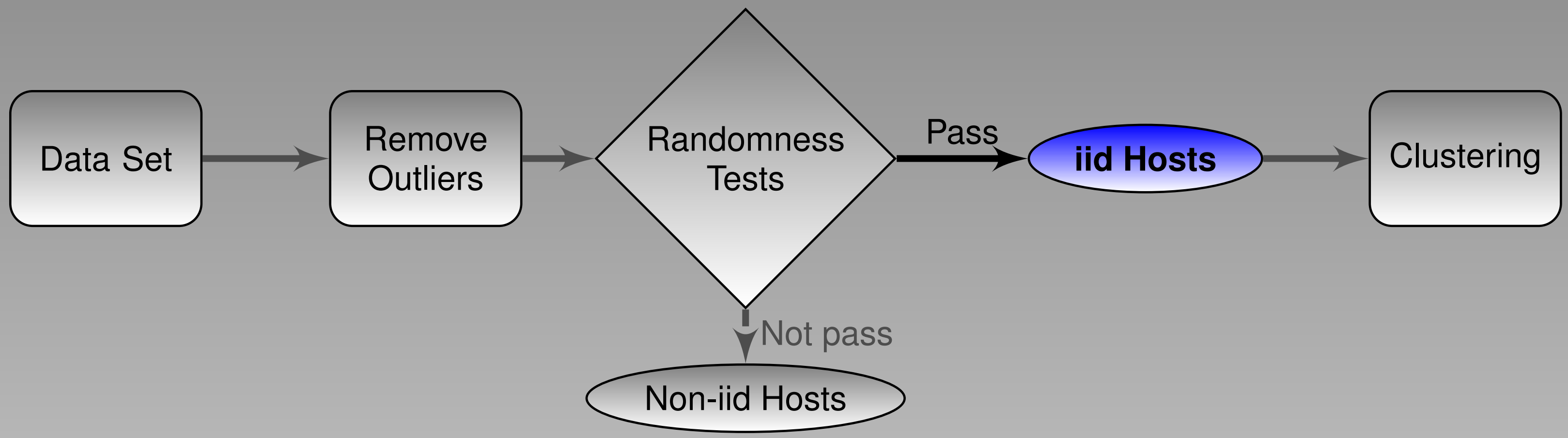
Test	Runs std	Runs up/down	ACF	Kendall	All
# of hosts	101649	144656	109138	101462	**57757**
Fraction	0.602	0.857	0.647	0.601	**0.342**

Result: 34% are i.i.d. hosts (2.2 PetaFLOPS)

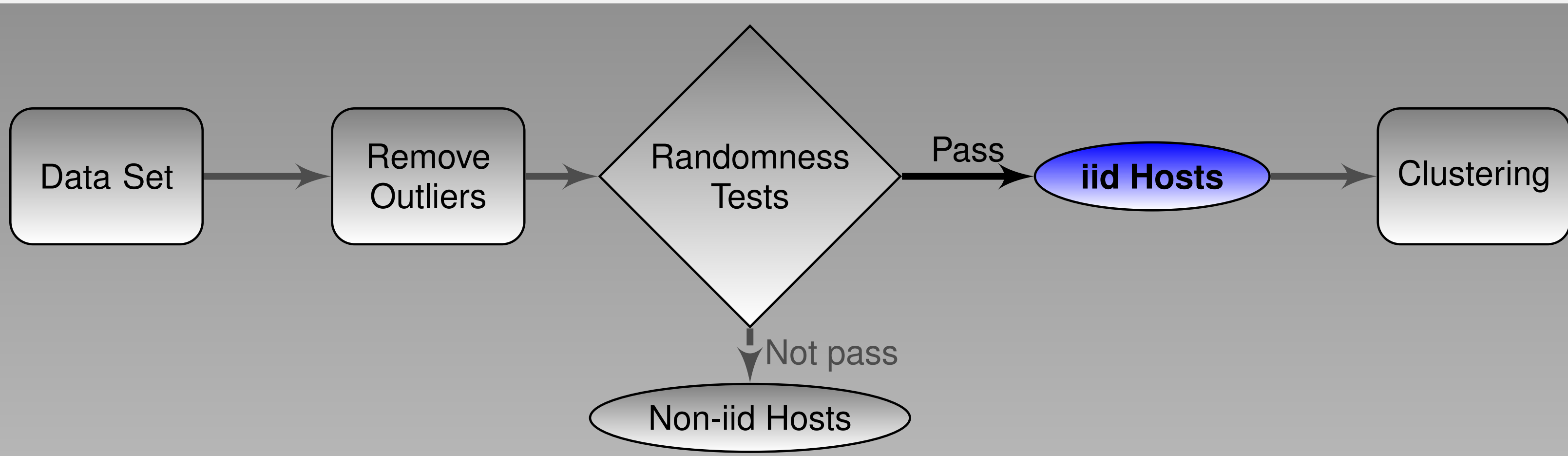
Outline

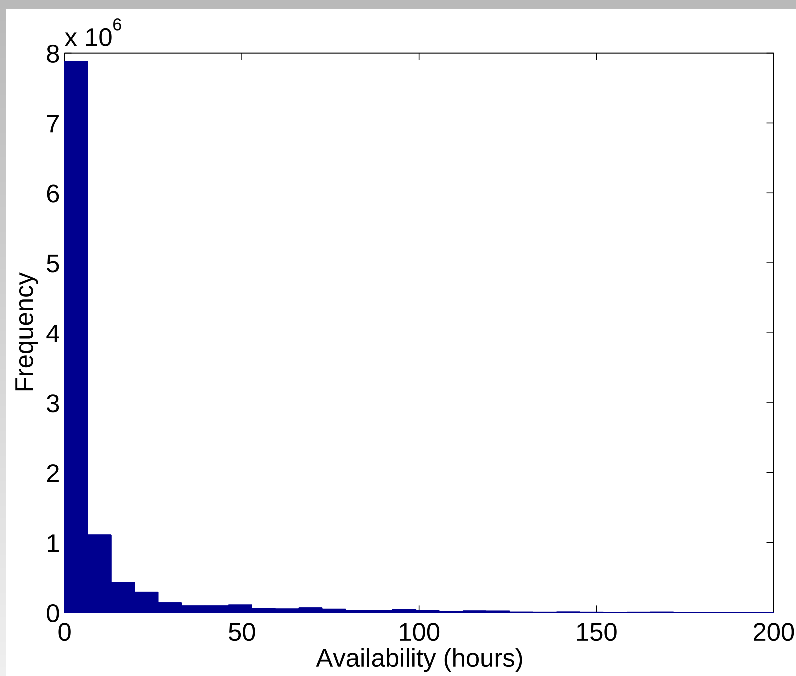
1. Introduction and Motivation
2. Measurement
   - Remove outliers
3. Modelling Process
   - Randomness Tests
   - Clustering
   - Model fitting
4. Discussions
   - Significance of Clustering Criteria
   - Scheduling Implications
5. Related Work
6. Conclusion and Future Work

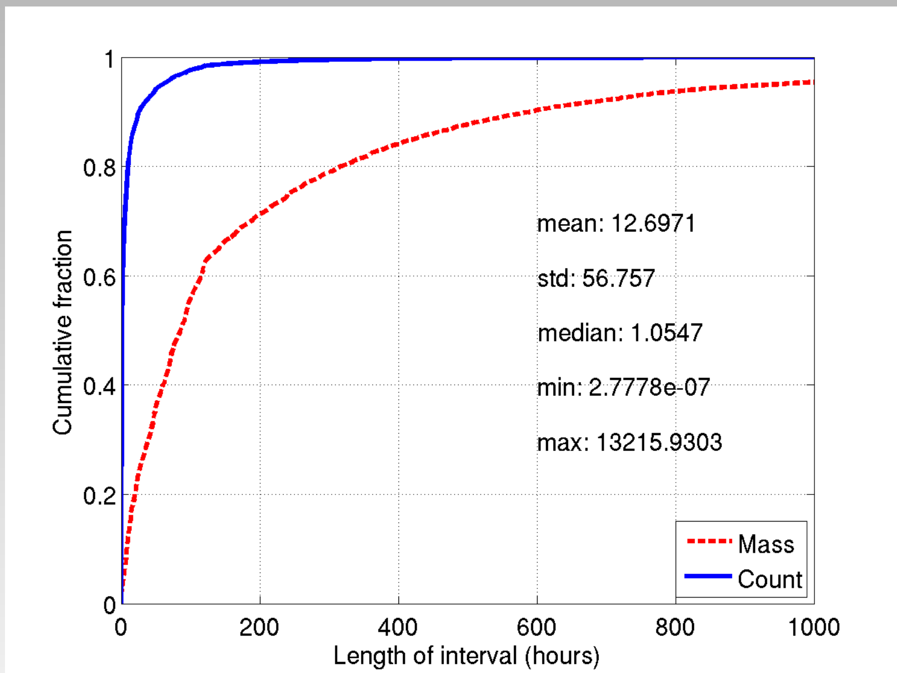
Distribution of Availability Intervals



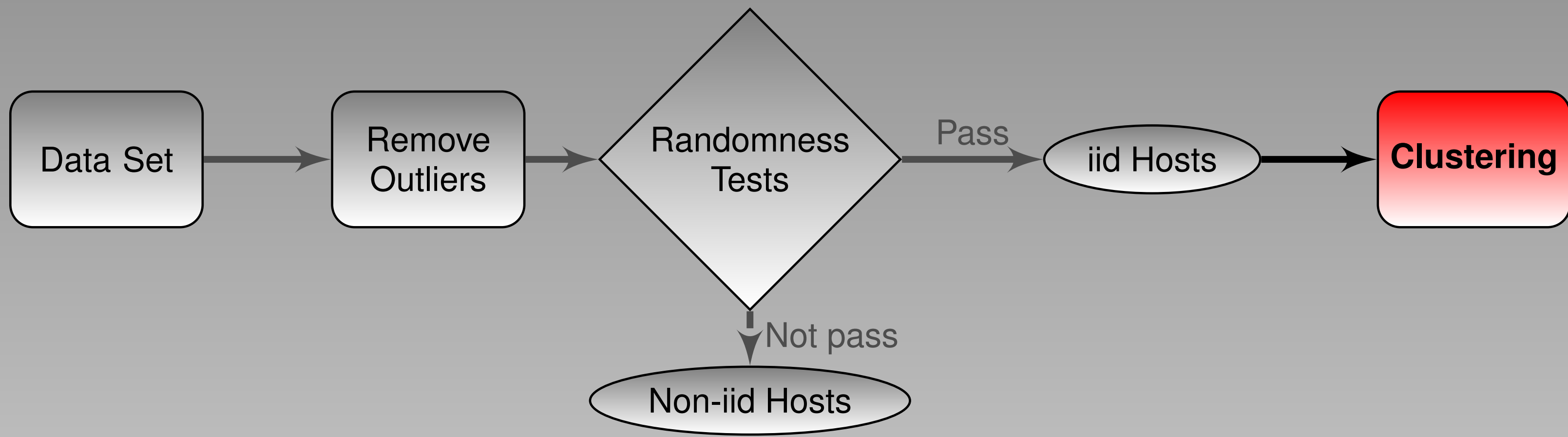
Distribution of Availability Intervals





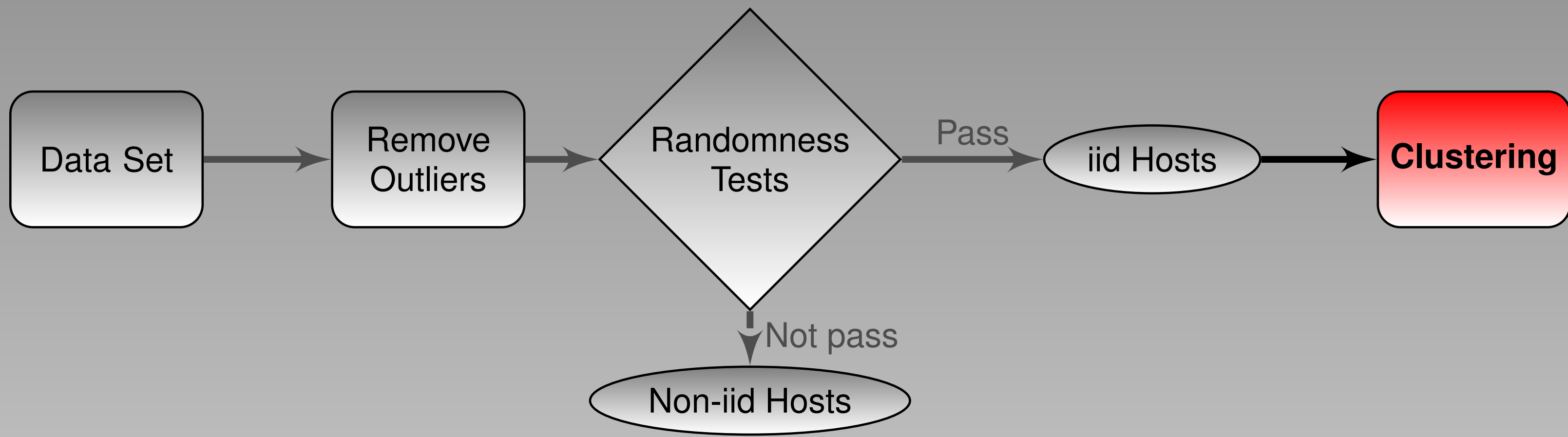


Clustering Method



Generate a few clusters based on availability distribution function

Clustering Method

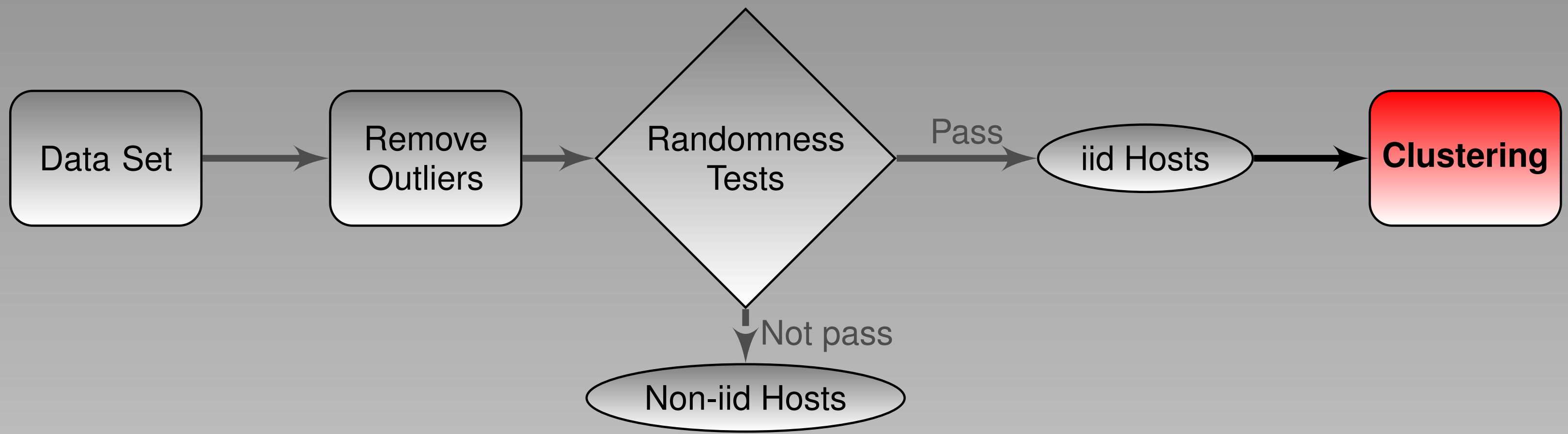


Generate a few clusters based on availability distribution function

Method:

- Hierarchical

Clustering Method

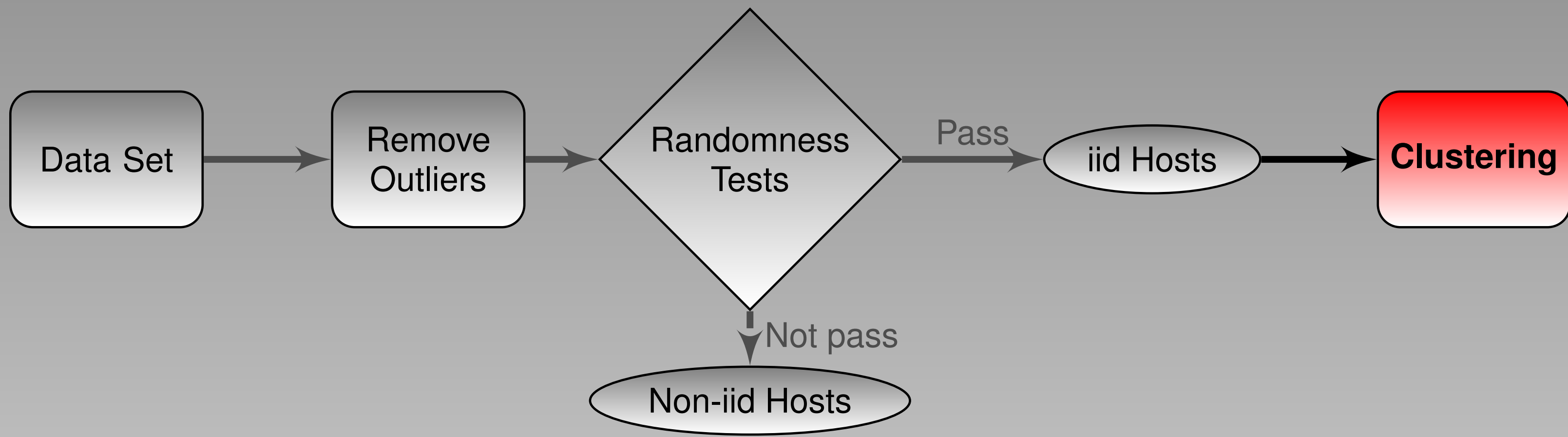


Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations

Clustering Method

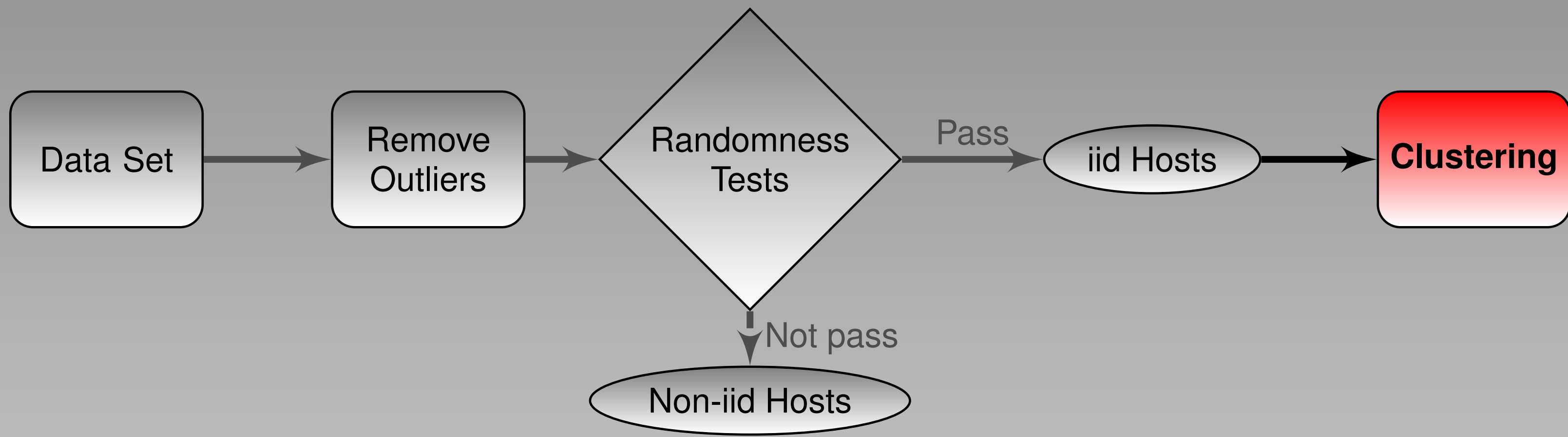


Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations
  - Memory intensive

Clustering Method

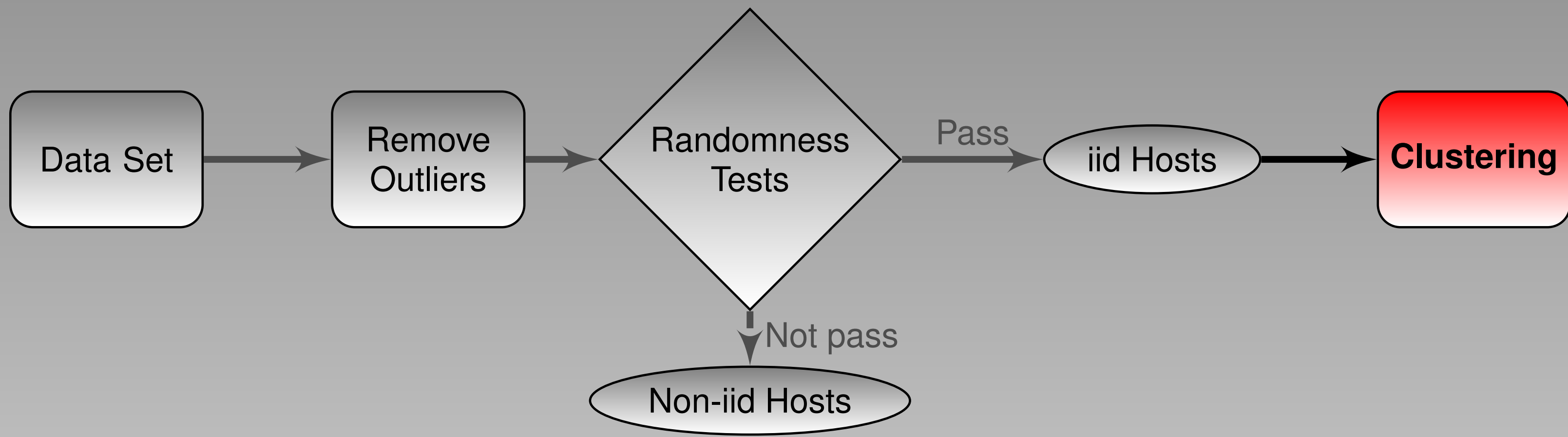


Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations
  - Memory intensive
- K-means (fast K-means)

Clustering Method

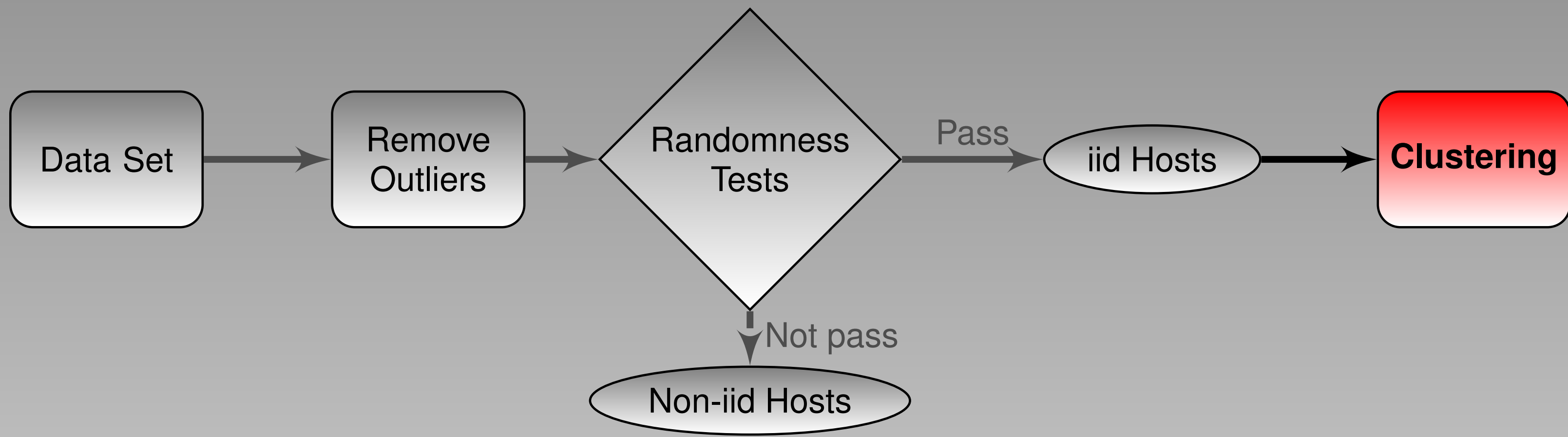


Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations
  - Memory intensive
- K-means (fast K-means)
  - Fast convergence

Clustering Method



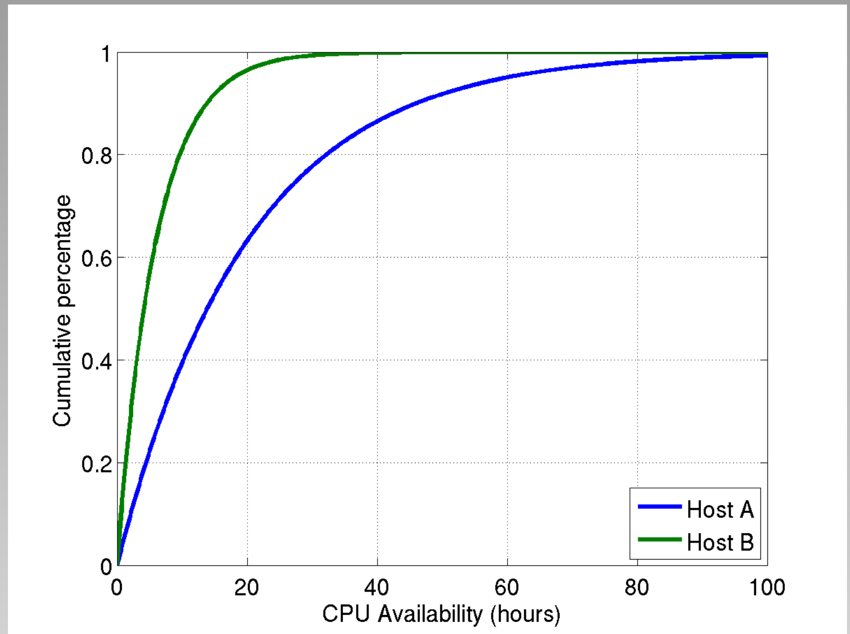
Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations
  - Memory intensive
- K-means (fast K-means)
  - Fast convergence
  - Dependent on initial centroids

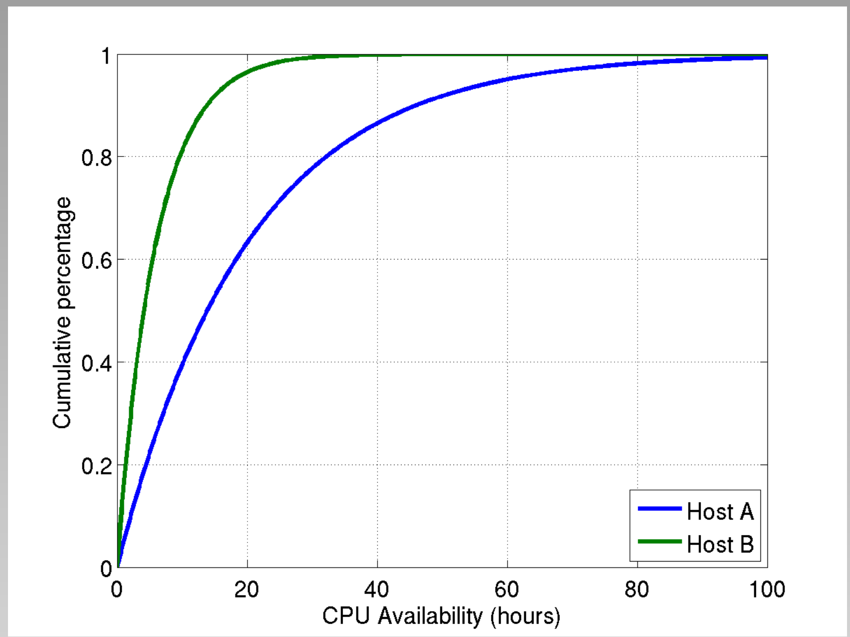
Distance Metrics

Distance between CDF of two hosts



Distance Metrics

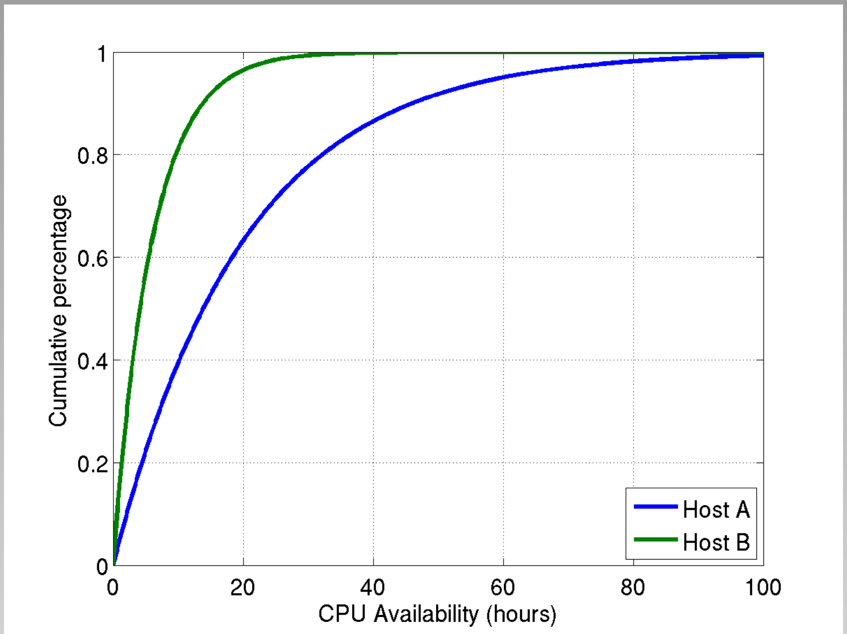
Distance between CDF of two hosts



- Kolmogorov-Smirnov: Maximum difference between two CDFs

Distance Metrics

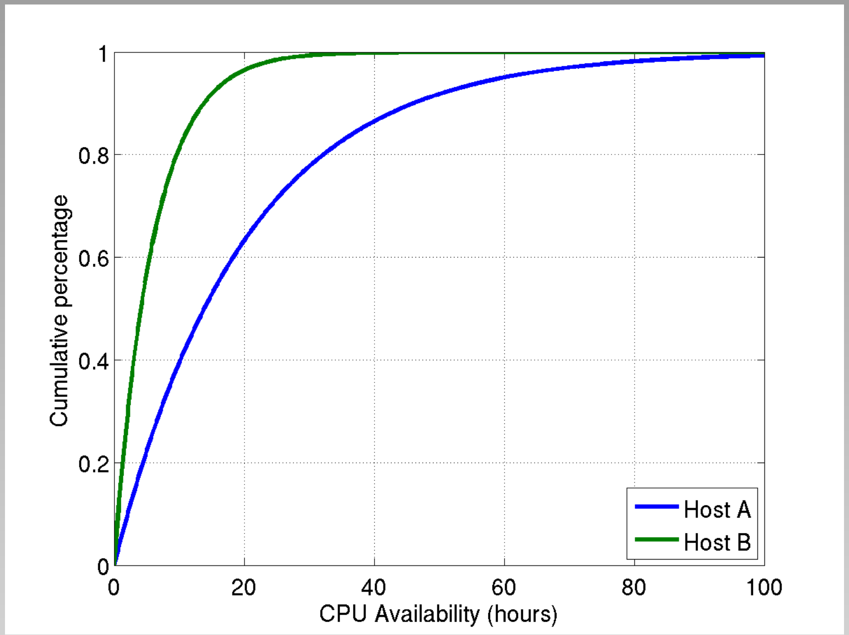
Distance between CDF of two hosts



- Kolmogorov-Smirnov: Maximum difference between two CDFs
- Kuiper: Maximum deviation above and below of two CDFs

Distance Metrics

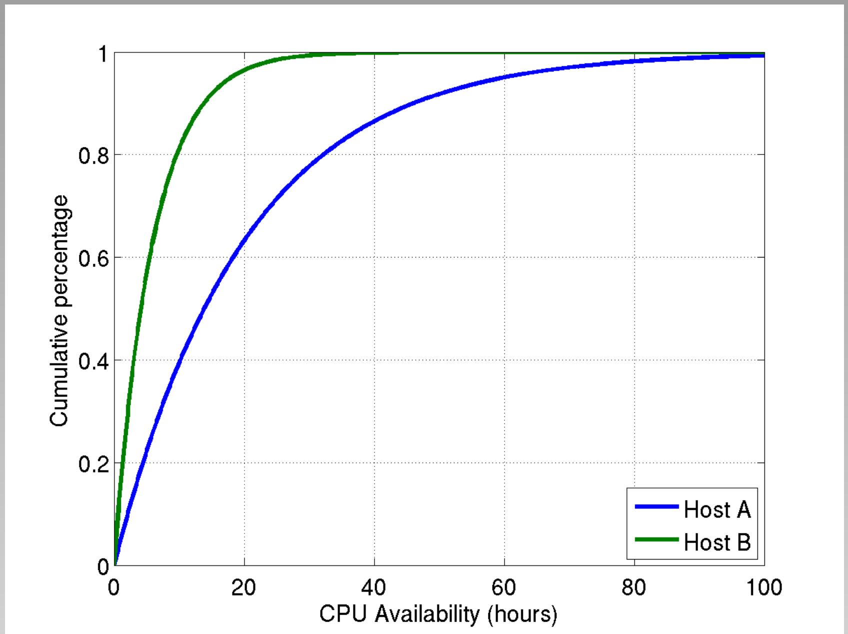
Distance between CDF of two hosts



- Kolmogorov-Smirnov: Maximum difference between two CDFs
- Kuiper: Maximum deviation above and below of two CDFs
- Cramer-von Mises: Area between two CDFs

Distance Metrics

Distance between CDF of two hosts



- Kolmogorov-Smirnov: Maximum difference between two CDFs
- Kuiper: Maximum deviation above and below of two CDFs
- Cramer-von Mises: Area between two CDFs
- Anderson-Darling: Area between two CDFs, more weight on the tail

Distance Metrics

Important Challenge:

Number of samples in each CDF

- Few samples –> not enough confidence on the result

Distance Metrics

Important Challenge:

Number of samples in each CDF

- Few samples –> not enough confidence on the result
- Too much samples –> the metric will be too sensitive

Distance Metrics

Important Challenge:

Number of samples in each CDF

- Few samples –> not enough confidence on the result
- Too much samples –> the metric will be too sensitive

- Data Set: different hosts have different number of samples

Distance Metrics

Important Challenge:

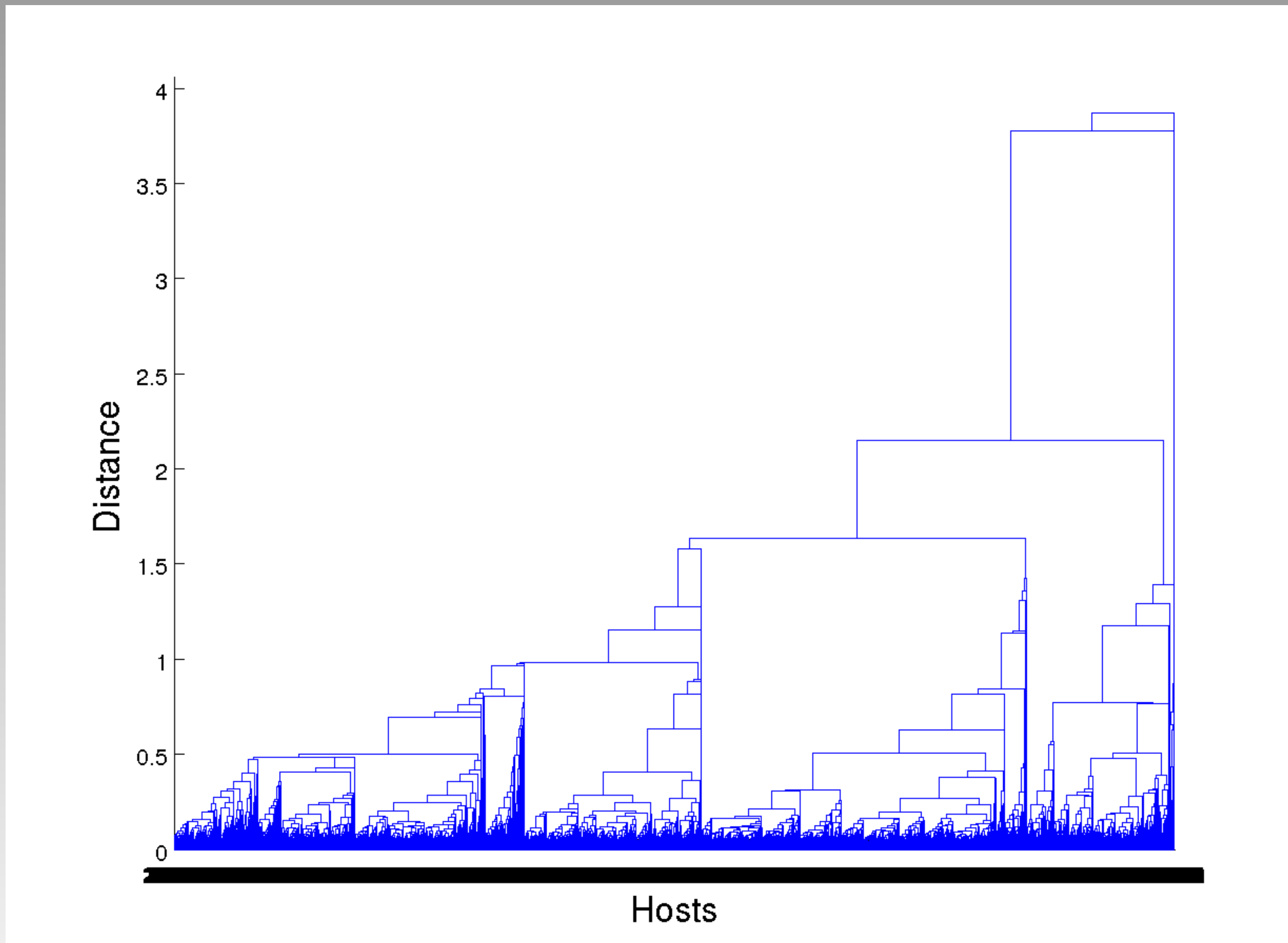
Number of samples in each CDF

- Few samples –> not enough confidence on the result
- Too much samples –> the metric will be too sensitive

- Data Set: different hosts have different number of samples
- Our solution: randomly select a fixed number of intervals from each host (i.e., 30 samples)

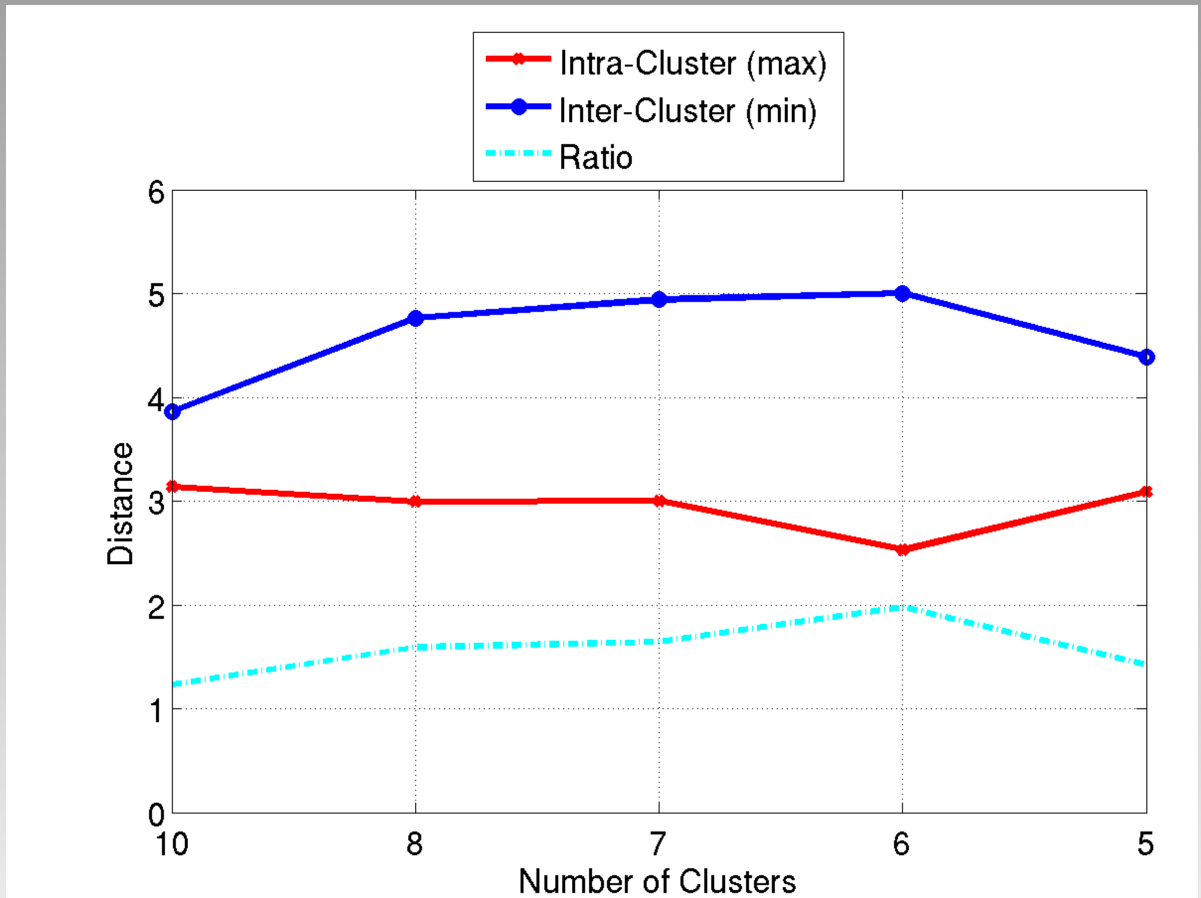
Clustering Results

Dendrogram of hierarchical clustering: 5-10 distinct groups (bootstrap)

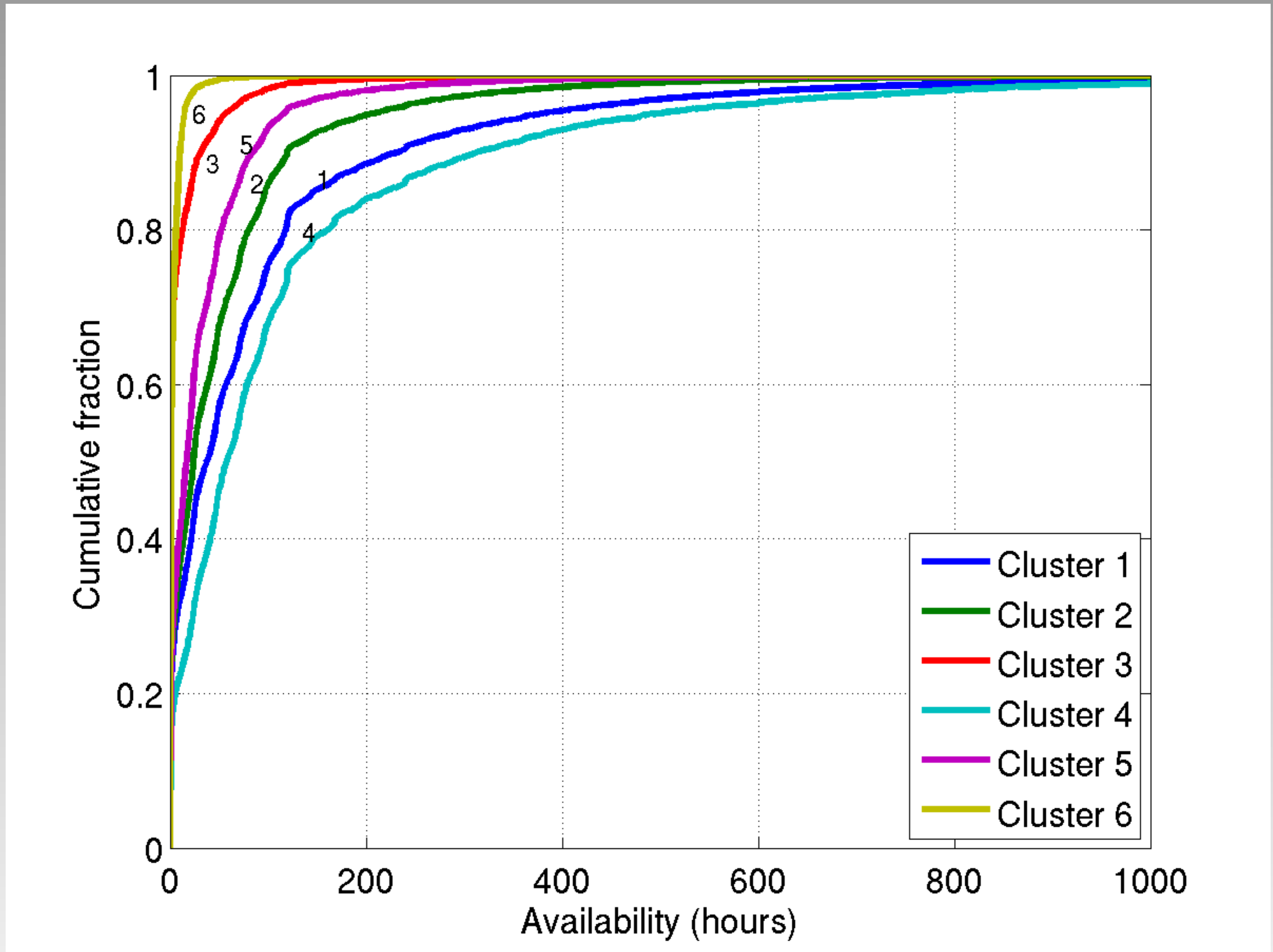


Clustering Results

Comparison of distances in clusters (k-means for all iid hosts):

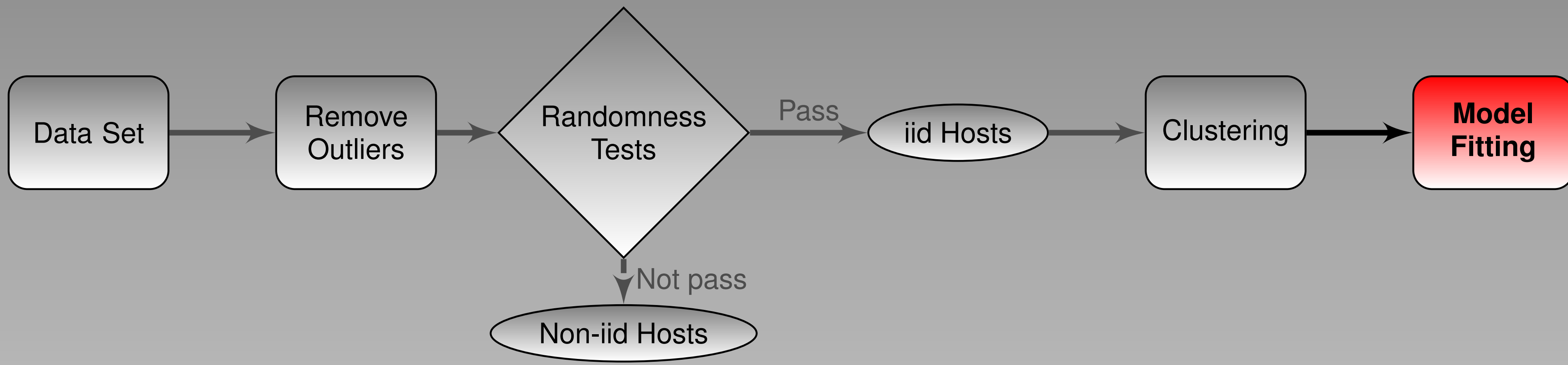


EDF of clusters

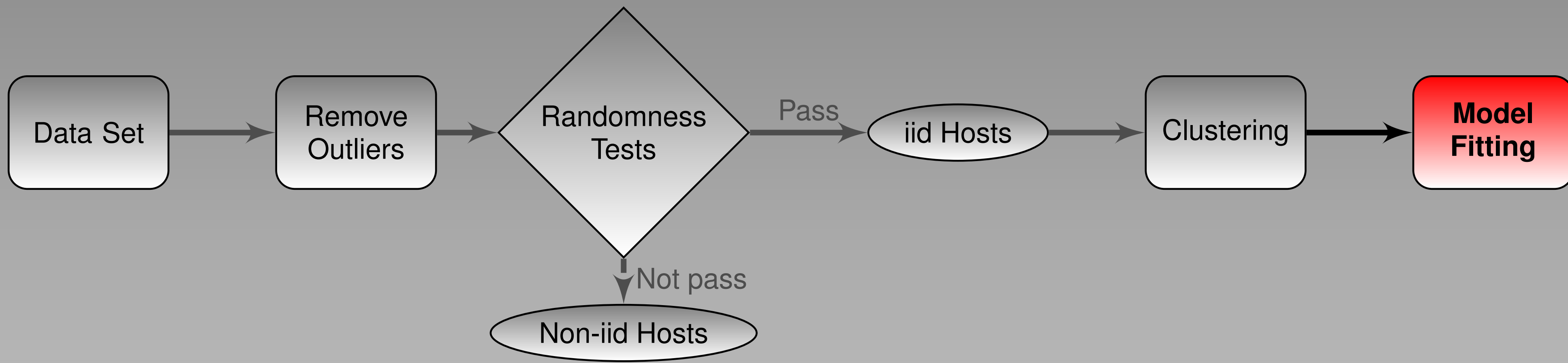


Outline

Methods


Data Set
Remove Outliers
Randomness Tests
Pass
iid Hosts
Clustering
Model Fitting
Not pass
Non-iid Hosts

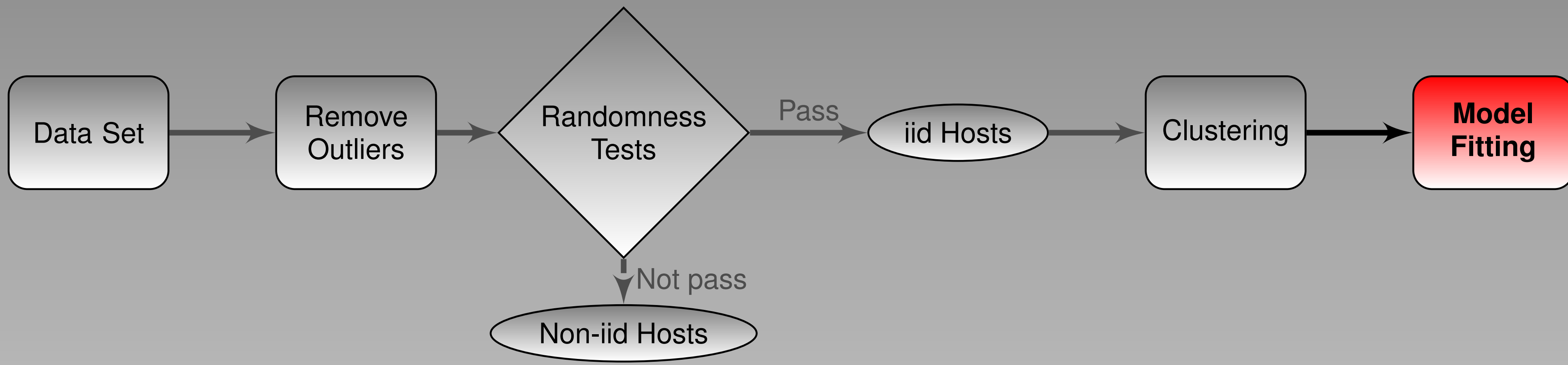
Methods



Method:

- Maximum Likelihood Estimation (MLE)
- Moment Matching (MM)

Methods



Method:

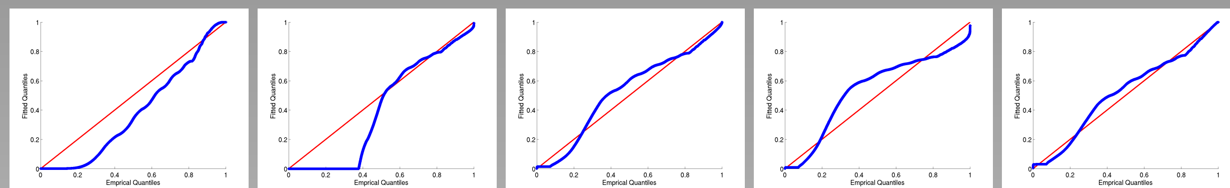
- Maximum Likelihood Estimation (MLE)
- Moment Matching (MM)

Target Distributions:

- Exponential
- Pareto
- Weibull
- Log-normal
- Gamma

Graphical Test

PP-plots: Exponential, Pareto, Weibull, Log-normal, Gamma

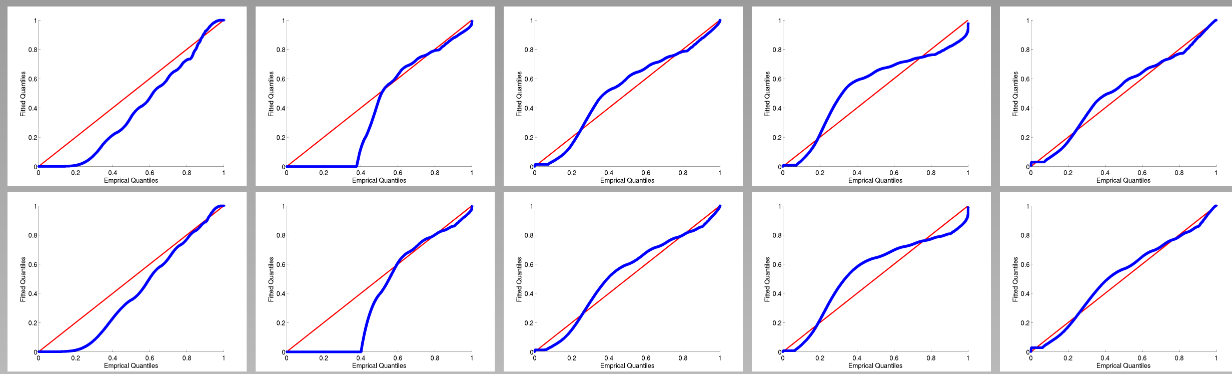
Cluster 1: 

Graphical Test

PP-plots: Exponential, Pareto, Weibull, Log-normal, Gamma

Cluster 1:

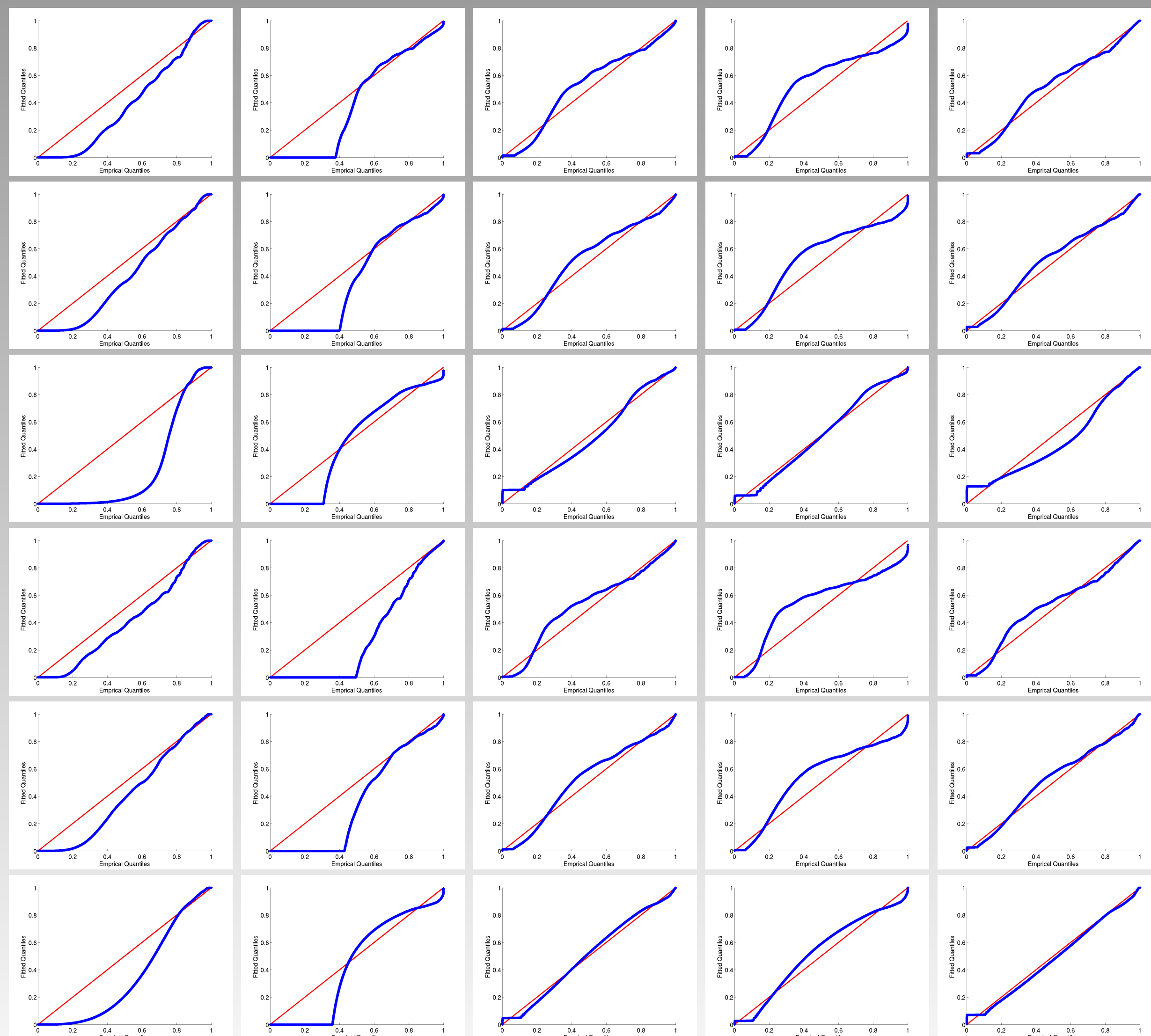
Cluster 2:



Graphical Test

PP-plots: Exponential, Pareto, Weibull, Log-normal, Gamma

Cluster 1:
Cluster 2:
Cluster 3:
Cluster 4:
Cluster 5:
Cluster 6:



Goodness Of Fit Tests

Generate p-values by two GOF tests (average over 1000 runs):

- Kolmogorov-Smirnov (KS) test
- Anderson-Darling (AD) test

Goodness Of Fit Tests

Generate p-values by two GOF tests (average over 1000 runs):

- Kolmogorov-Smirnov (KS) test
- Anderson-Darling (AD) test

Data sets	Exponential		Pareto		Weibull		Log-Normal		Gamma	
	AD	*KS*	*AD*	*KS*	*AD*	*KS*	*AD*	*KS*	*AD*	*KS*
All iid hosts	0.004	0.000	0.061	0.013	**0.581**	**0.494**	0.568	0.397	0.431	0.359
Cluster 1	0.155	0.071	0.029	0.008	0.466	0.243	0.275	0.116	**0.548**	**0.336**
Cluster 2	0.188	0.091	0.020	0.004	0.471	0.259	0.299	0.128	**0.565**	**0.384**
Cluster 3	0.002	0.000	0.068	0.023	0.485	0.380	**0.556**	**0.409**	0.372	0.241
Cluster 4	0.264	0.163	0.002	0.000	0.484	0.242	0.224	0.075	**0.514**	**0.276**
Cluster 5	0.204	0.098	0.013	0.002	0.498	0.296	0.314	0.153	**0.563**	**0.389**
Cluster 6	0.059	0.016	0.033	0.009	**0.570**	**0.439**	0.485	0.328	0.538	0.467

Some properties of clusters

Clusters	# of hosts	% of total avail.	mean (hrs)	Best fit	Parameters	
					shape	*scale*
All iid hosts	57757	1.0	12.697	Weibull	0.3787	3.0932
Cluster 1	3606	0.16	90.780	Gamma	0.3131	289.9017
Cluster 2	9321	0.35	54.563	Gamma	0.3372	161.8350
Cluster 3	13256	0.22	11.168	Log-Normal	-0.8937	3.2098
Cluster 4	275	0.01	123.263	Gamma	0.3739	329.6922
Cluster 5	1753	0.05	34.676	Gamma	0.3624	95.6827
Cluster 6	29546	0.20	4.138	Weibull	0.4651	1.8461

- Cluster sizes are different and often significant

Some properties of clusters

Clusters	# of hosts	% of total avail.	mean (hrs)	Best fit	Parameters	
					shape	*scale*
All iid hosts	57757	1.0	12.697	Weibull	0.3787	3.0932
Cluster 1	3606	0.16	90.780	Gamma	0.3131	289.9017
Cluster 2	9321	0.35	54.563	Gamma	0.3372	161.8350
Cluster 3	13256	0.22	11.168	Log-Normal	-0.8937	3.2098
Cluster 4	275	0.01	123.263	Gamma	0.3739	329.6922
Cluster 5	1753	0.05	34.676	Gamma	0.3624	95.6827
Cluster 6	29546	0.20	4.138	Weibull	0.4651	1.8461

- Cluster sizes are different and often significant
- Heterogeneity in distribution parameters (different *scale* parameters)

Some properties of clusters

Clusters	# of hosts	% of total avail.	mean (hrs)	Best fit	Parameters	
					shape	*scale*
All iid hosts	57757	1.0	12.697	Weibull	0.3787	3.0932
Cluster 1	3606	0.16	90.780	Gamma	0.3131	289.9017
Cluster 2	9321	0.35	54.563	Gamma	0.3372	161.8350
Cluster 3	13256	0.22	11.168	Log-Normal	-0.8937	3.2098
Cluster 4	275	0.01	123.263	Gamma	0.3739	329.6922
Cluster 5	1753	0.05	34.676	Gamma	0.3624	95.6827
Cluster 6	29546	0.20	4.138	Weibull	0.4651	1.8461

- Cluster sizes are different and often significant
- Heterogeneity in distribution parameters (different *scale* parameters)
- Decreasing hazard rate

Outline

Significance of Clustering Criteria

Could the same clusters have been found using some other static criteria?

Significance of Clustering Criteria

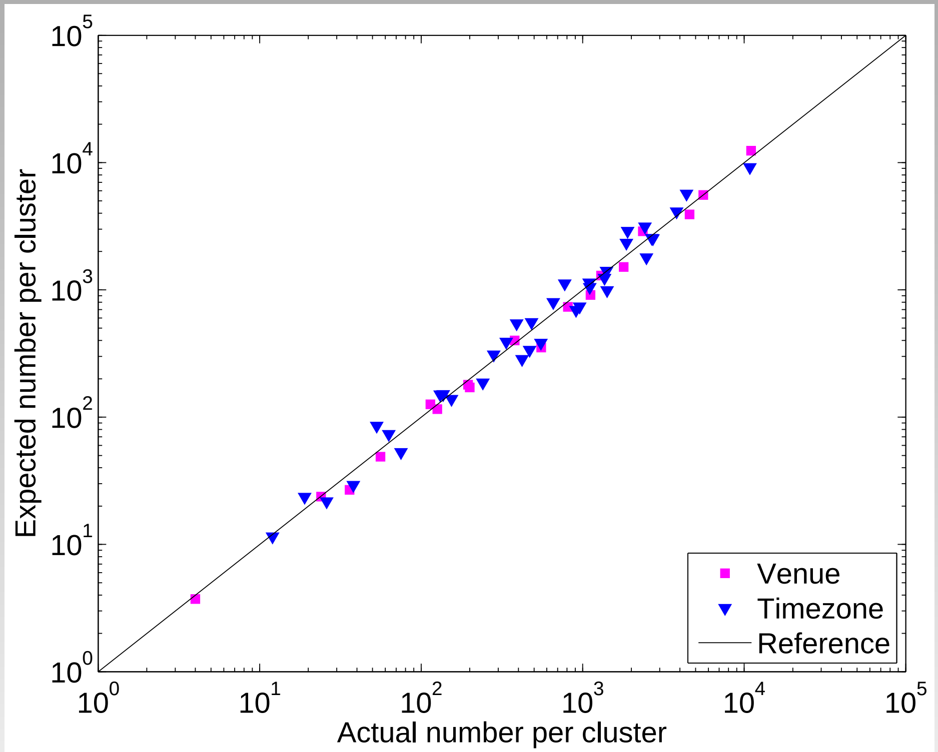
Could the same clusters have been found using some other static criteria?

- Cluster by venue: Work, Home, School
- Cluster by Time zone: 6 different time zones

Significance of Clustering Criteria

Could the same clusters have been found using some other static criteria?

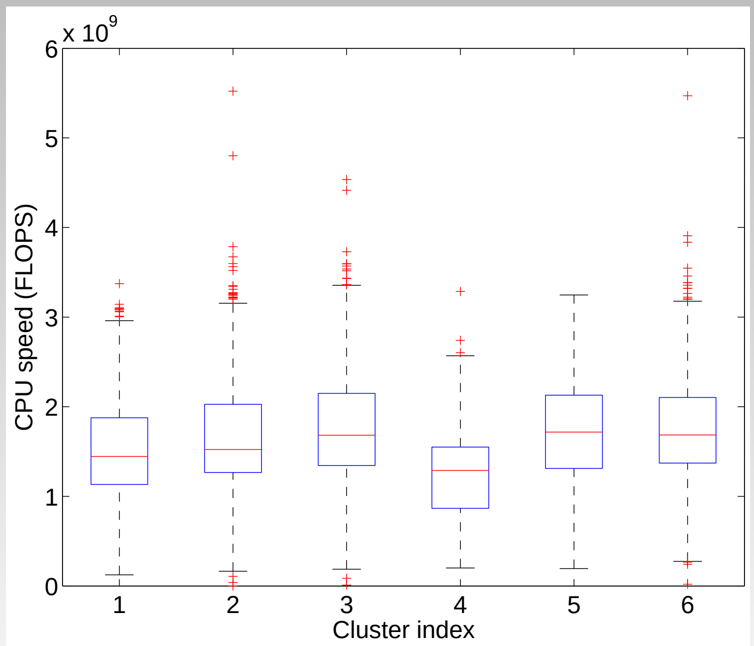
- Cluster by venue: Work, Home, School
- Cluster by Time zone: 6 different time zones



Significance of Clustering Criteria

Could the same clusters have been found using some other static criteria?

- Cluster by venue: Work, Home, School
- Cluster by Time zone: 6 different time zones
- Cluster by CPU speed



Outline

1. Introduction and Motivation
2. Measurement
   - Remove outliers
3. Modelling Process
   - Randomness Tests
   - Clustering
   - Model fitting
4. Discussions
   - Significance of Clustering Criteria
   - Scheduling Implications
5. Related Work
6. Conclusion and Future Work

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

- Global model: <20%
- Cluster 4: 70%

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

- Global model: <20%
- Cluster 4: 70%

Resource Selection/Replication

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

- Global model: <20%
- Cluster 4: 70%

Resource Selection/Replication

- Single job: Prediction of task failure

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

- Global model: <20%
- Cluster 4: 70%

Resource Selection/Replication

- Single job: Prediction of task failure
- Multi-job: How the task size distribution follows the availability distribution

Related Work

Different from other research

- Measurement

Related Work

Different from other research

- Measurement
    - Resource type: home, work, and school

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts
  - Duration: 1.5 years

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts
  - Duration: 1.5 years
  - Availability : CPU availability

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts
  - Duration: 1.5 years
  - Availability : CPU availability
- Modelling

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts
  - Duration: 1.5 years
  - Availability : CPU availability
- Modelling
  - Classification according to randomness tests

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts
  - Duration: 1.5 years
  - Availability : CPU availability
- Modelling
  - Classification according to randomness tests
  - Cluster-based Model vs Global Model

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)
  - 34% of hosts have truly random availability intervals

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)
  - 34% of hosts have truly random availability intervals
  - Six clusters with three different distributions: Gamma, Weibull, and Log-normal

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)
  - 34% of hosts have truly random availability intervals
  - Six clusters with three different distributions: Gamma, Weibull, and Log-normal

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)
  - 34% of hosts have truly random availability intervals
  - Six clusters with three different distributions: Gamma, Weibull, and Log-normal

Future Work

- Apply the result for improving makespan of DAG-applications
- Explore ability of clustering dynamically while the system is on-line

Failure Trace Archive

http://fta.inria.fr

- Repository of availability traces of parallel and distributed systems, and tools for analysis
- Facilitate design, validation and comparison of fault-tolerance algorithms and models
- 15 data sets including SETI@home data set

Failure Trace Archive

http://fta.inria.fr

- Repository of availability traces of parallel and distributed systems, and tools for analysis
- Facilitate design, validation and comparison of fault-tolerance algorithms and models
- 15 data sets including SETI@home data set

More Details

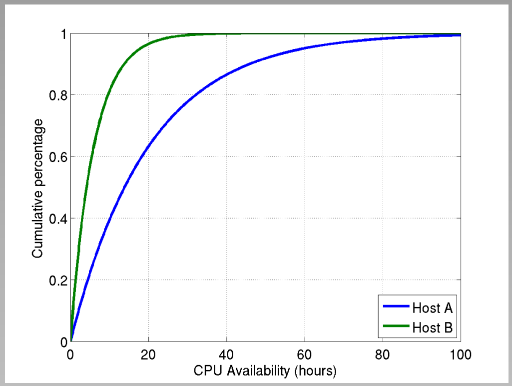
- Poster Session at MASCOTS 2009 (Today 19:00-21:00)
- Website: http://fta.inria.fr

Thank You

Questions?

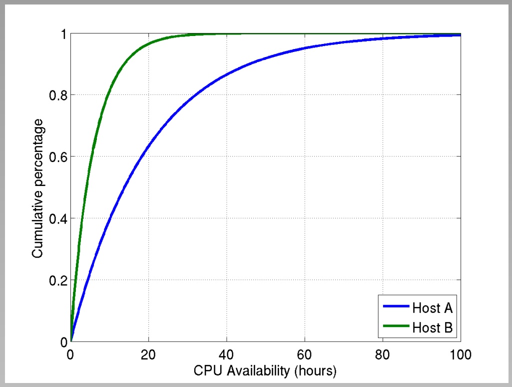
Distance Metrics

Distance between CDF of two hosts



Distance Metrics

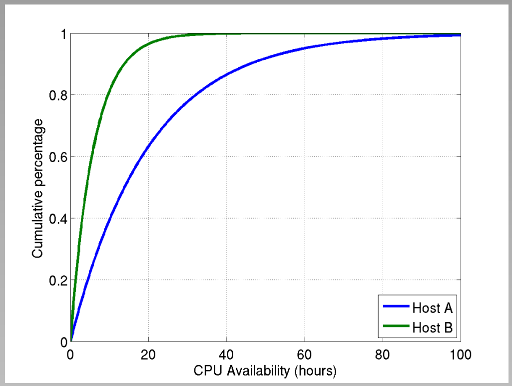
Distance between CDF of two hosts



- Kolmogorov-Smirnov: $D_{n,m} = sup \mid F_n(x) - G_m(x) \mid$

Distance Metrics

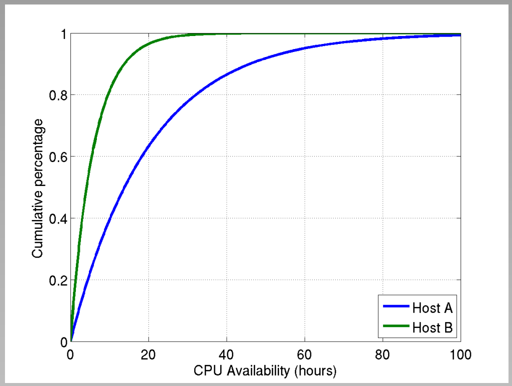
Distance between CDF of two hosts



- Kolmogorov-Smirnov: $D_{n,m} = sup \mid F_n(x) - G_m(x) \mid$
- Kuiper: $V_{n,m} = sup \mid F_n(x) - G_m(x) \mid + sup \mid G_m(x) - F_n(x) \mid$

Distance Metrics

Distance between CDF of two hosts

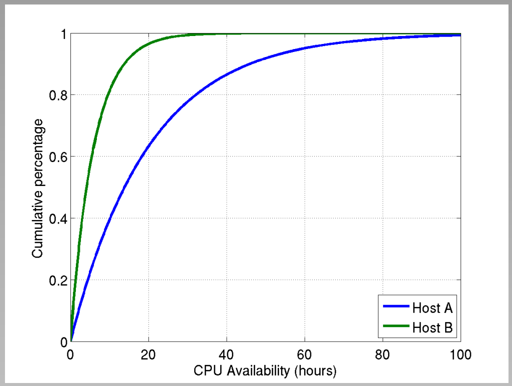


- Kolmogorov-Smirnov: $D_{n,m} = sup \mid F_n(x) - G_m(x) \mid$
- Kuiper: $V_{n,m} = sup \mid F_n(x) - G_m(x) \mid + sup \mid G_m(x) - F_n(x) \mid$
- Cramer-von Mises:

$$T_{n,m} = \frac{nm}{(n+m)^2} \left\{ \sum_{i=1}^{n} [F_n(x_i) - G_m(x_i)]^2 + \sum_{j=1}^{m} [F_n(y_j) - G_m(y_j)]^2 \right\}$$

Distance Metrics

Distance between CDF of two hosts



- Kolmogorov-Smirnov: $D_{n,m} = sup \mid F_n(x) - G_m(x) \mid$
- Kuiper: $V_{n,m} = sup \mid F_n(x) - G_m(x) \mid + sup \mid G_m(x) - F_n(x) \mid$
- Cramer-von Mises:

$$T_{n,m} = \frac{nm}{(n+m)^2} \left\{ \sum_{i=1}^{n} [F_n(x_i) - G_m(x_i)]^2 + \sum_{j=1}^{m} [F_n(y_j) - G_m(y_j)]^2 \right\}$$

- Anderson-Darling: $Q_n = \int_{-\infty}^{\infty} [F(x) - F_n(x)]^2 \psi(F(x)) dF$

$\psi(F(x)) = \frac{1}{F(x)(1-F(x))}$

Fitting with Hyper-Exponential

Fitting Method:

- Expectation Maximization (EM) [using EMpht package]
  - Accurate
  - Flexible
  - Slow

Fitting with Hyper-Exponential

Fitting Method:

- Expectation Maximization (EM) [using EMpht package]
  - Accurate
  - Flexible
  - Slow
- Moment Matching (MM)
  - Less accurate
  - Not flexible
  - Very fast

Fitting with Hyper-Exponential

Fitting Method:

- Expectation Maximization (EM) [using EMpht package]
  - Accurate
  - Flexible
  - Slow
- Moment Matching (MM)
  - Less accurate
  - Not flexible
  - Very fast

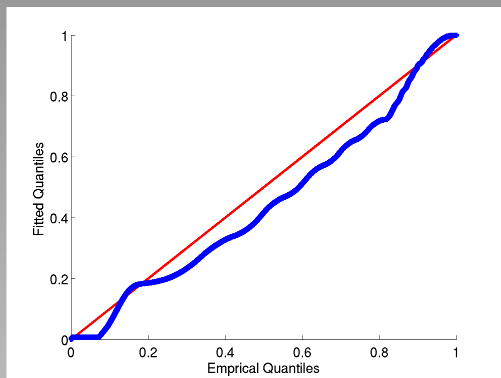
We used MM for 2-phase hyper-exponential by the first two moments as follows:

$$p = \frac{1}{2}\left(1 - \sqrt{\frac{CV^2-1}{CV^2+1}}\right)$$

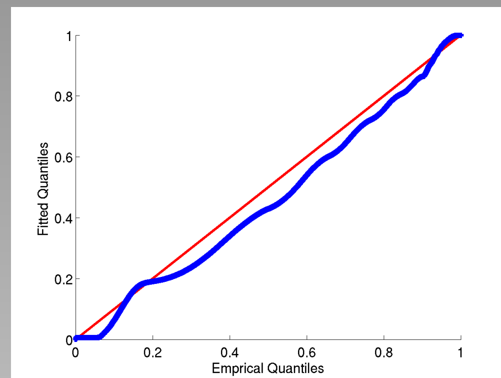
$$\lambda_1 = \frac{2p}{\mu}$$

$$\lambda_2 = \frac{2(1-p)}{\mu}$$

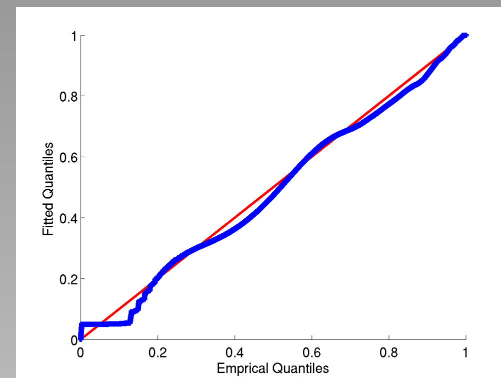
PP-Plots



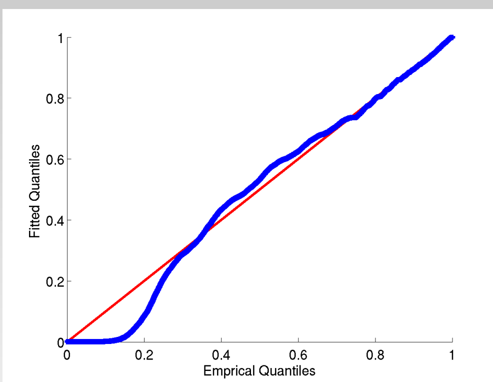
(a) Cluster 1



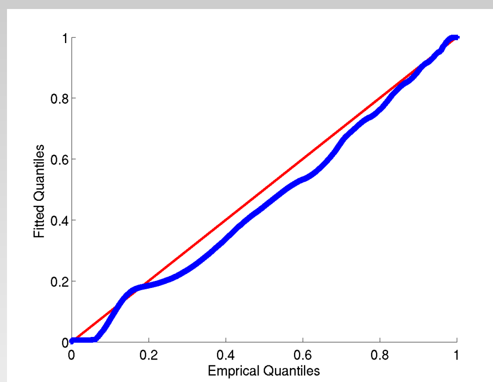
(b) Cluster 2



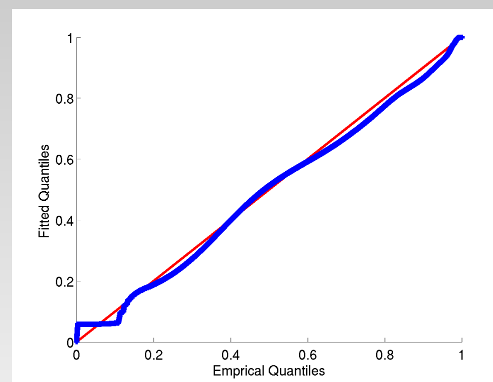
(c) Cluster 3



(d) Cluster 4



(e) Cluster 5



(f) Cluster 6

Goodness of Fit Tests

Data sets	Hyper-Exponential (MM) *Parameters*	*AD*	*KS*	Hyper-Exponential (EM) *Parameters*	*AD*	*KS*
All iid hosts	$p_1 = 0.024$ $\lambda_1 = 0.004$ $p_2 = 0.976$ $\lambda_2 = 0.154$	0.026	0.005	$p_1 = 0.197$ $\lambda_1 = 0.0179$ $p_2 = 0.279$ $\lambda_2 = 29.171$ $p_3 = 0.524$ $\lambda_3 = 0.316$	0.531	0.375
Cluster 1	$p_1 = 0.115$ $\lambda_1 = 0.003$ $p_2 = 0.885$ $\lambda_2 = 0.019$	0.287	0.119	$p_1 = 0.180$ $\lambda_1 = 14.401$ $p_2 = 0.820$ $\lambda_2 = 0.009$	0.450	0.318
Cluster 2	$p_1 = 0.114$ $\lambda_1 = 0.004$ $p_2 = 0.886$ $\lambda_2 = 0.032$	0.275	0.113	$p_1 = 0.183$ $\lambda_1 = 12.338$ $p_2 = 0.817$ $\lambda_2 = 0.015$	0.512	0.403
Cluster 3	$p_1 = 0.030$ $\lambda_1 = 0.005$ $p_2 = 0.970$ $\lambda_2 = 0.174$	0.005	0.000	$p_1 = 0.341$ $\lambda_1 = 0.031$ $p_2 = 0.261$ $\lambda_2 = 71.852$ $p_3 = 0.398$ $\lambda_3 = 1.923$	**0.561**	**0.434**
Cluster 4	$p_1 = 0.136$ $\lambda_1 = 0.002$ $p_2 = 0.864$ $\lambda_2 = 0.014$	0.448	0.273	$p_1 = 0.694$ $\lambda_1 = 0.020$ $p_2 = 0.306$ $\lambda_2 = 0.003$	0.473	0.274
Cluster 5	$p_1 = 0.105$ $\lambda_1 = 0.006$ $p_1 = 0.895$ $\lambda_2 = 0.052$	0.295	0.122	$p_1 = 0.173$ $\lambda_1 = 13.374$ $p_2 = 0.827$ $\lambda_2 = 0.024$	0.523	0.393
Cluster 6	$p_1 = 0.010$ $\lambda_1 = 0.005$ $p_2 = 0.990$ $\lambda_2 = 0.478$	0.114	0.038	$p_1 = 0.516$ $\lambda_1 = 0.131$ $p_2 = 0.150$ $\lambda_2 = 163.771$ $p_3 = 0.334$ $\lambda_3 = 2.411$	**0.572**	**0.470**

Outline

Synthesis : principles

1. Formulate the **hypothesis**
2. Design the experiment to **validate** the hypothesis
3. Check the validity of the experience
4. Analyse the experiments to validate or invalidate the hypothesis
5. Report the arguments in a convincing form

Synthesis : Steps for a Performance Evaluation Study [Jain]

1. State the goals of the study and define system boundaries.
2. List system services and possible outcomes.
3. Select performance metrics.
4. List system and workload parameters
5. Select factors and their values.
6. Select evaluation techniques.
7. Select the workload.
8. Design the experiments.
9. Analyze and interpret the data.
10. Present the results. Start over, if necessary.

Common mistakes in experimentation [Jain]

1. The variation due to experimental error is ignored
2. Important parameters are not controlled
3. Simple one-factor-at-a-time designs are used
4. Interactions are ignored
5. Too many experiments are conducted

References

Bibliography

- **The Art of Computer Systems Performance Analysis : Techniques for Experimental Design, Measurment, Simulation and Modeling.** Raj Jain *Wiley 1991* `http://www.rajjain.com/`
- **Measuring Computer Performance: A Practitioner's Guide** David J. Lilja Cambridge University Press, 2000.
- **Performance Evaluation of Computer and Communication Systems** Jean-Yves Le Boudec EPFL `http://perfeval.epfl.ch/lectureNotes.htm`

Common tools

- Mathlab, Matematica
- Scilab `http://www.scilab.org/`
- gnuplot `http://www.gnuplot.info/`
- R `http://www.r-project.org/`