Master Course
Computer Networks
IN2097

Prof. Dr.-Ing. Georg Carle,
Christian Grothoff, Ph.D.,
Dr. Nils Kammenhuber,
Dirk Haage

Chair for Network Architectures and Services
Department of Computer Science
Technische Universität München
http://www.net.in.tum.de

Chapter 7:

Network Measurements

Acknowledgements:

The content of this chapter
is partly based on slides from Anja Feldmann

Chapter 7 Outline – Network Measurements

- Recapitulation: Why do we measure and how?

- Network Traffic
  - Traffic patterns
  - Traffic characterization
  - Traffic models
  - Self-similar traffic

- Interpretation of measurement data
  - Before you start
  - Statistics 1-0-1
  - Dos and don´ts

Why do we measure the network?

- Network Provider View
  - Manage traffic
    - Predict future, model reality, plan network
    - Avoid bottlenecks in advance
  - Reduce cost
  - Accounting

- Client View
  - Get the best possible service
  - Check the service („Do I get what I´ve paid for?)

- Service Provider View
  - Get information about the client
  - Adjust service to demands
  - Reduce load on service
  - Accounting

But why?

- ❑ The network is well engineered
- ❑ Well documented protocols, mechanisms, …
- ❑ In theory we can know everything that is going on
- ⇨ There should be no need for measurements

- ❑ But:
  - ▪ Moving target:
    - requirements change
    - growth, usage, structure changes
  - ▪ Highly interactive system
  - ▪ Heterogeneity in all directions
  - ▪ The total is more than the sum of its pieces

- ❑ And: The network is built, driven and used by humans
  - ▪ Detection of errors, misconfigurations, flaws, failures, misuse, …

Measurement types

- Active Measurements
  - Intrusive
  - Find out what the network is capable of
  - Changes the network state
- Passive Measurements (or network monitoring)
  - Non-intrusive
  - Find out what the current situation is
  - Does not influence the network state (more or less)
- Hybrid
  - Alter actual traffic
  - Reduce the impact of active measurements
  - Might introduce new bias for applications

Network Traffic

Traffic by Port (I)

18 hours of traffic to AT&T dial clients on July 22, 1997

Name	Port	% Bytes	% Packets	Bytes/Packet
www	80	56,75	44,79	819
nntp	119	24,65	12,90	1235
pop3 email	110	1,88	3,17	384
cuseeme	7648	0,95	1,85	333
secure www	443	0,74	0,79	603
irc	6667	0,27	0,74	239
ftp	20	0,65	0,64	659
dns	53	0,19	0,58	210
…				

Traffic by Port (II)

24 hours of traffic to/from MWN clients in 2006

Name	Port	% Conns	% Succes	%Payload
www	80	70,82	68,13	72,59
cifs	445	3,53	0,01	0,00
secure www	443	2,34	2,08	1,29
ssh	22	2,12	1,75	1,71
smtp	25	1,85	1,05	1,71
	1042	1,66	0,00	0,00
	1433	1,06	0,00	0,00
	135	1,04	0,00	0,00
	< 1024	83,68	73,73	79,05
	> 1024	16,32	4,08	20,95

Traffic by Port (III)

- Port 80 dominates traffic mix
  - Still growing
    - More web applications
    - Tunnel everything over port 80
- Characterization of traffic by port is possible
  - Well-known ports
    (1-1024, take a look at /etc/services)
- Growing margin of error
  - Automatic configuration
  - * over http – VPN, P2P, VoIP, …
  - Aggressive applications (e.g. Skype):
    „just find me an open port“

Traffic Flows

18 hours of traffic to AT&T dial clients on July 22, 1997

Name	Port	% Bytes	% Pkts	Bytes/ Pkt	% Flows	Pkts/ Flow	Duration (s)
www	80	56,75	44,79	819	74,58	12	11,2
nntp	119	24,65	12,90	1235	1,20	210	132,6
pop3 email	110	1,88	3,17	384	2,80	22	10,3
cuseeme	7648	0,95	1,85	333	0,03	1375	192,0
secure www	443	0,74	0,79	603	0,99	16	14,2
irc	6667	0,27	0,74	239	0,16	89	384,6
ftp	20	0,65	0,64	659	0,26	47	30,1
dns	53	0,19	0,58	210	10,69	1	0,5
…							

Elephants and Mices

- Many very short flows (30% < 300 bytes)
- Many medium-sized flows (short web transfers)
- Most bytes belong to long flows (large images, files, flash, video)
- Same picture for other metrics
  - Bytes/flow
  - Packets/flow
  - lifetime
- Flow densities are traffic patterns and signatures

More ways to classify traffic

- Distribution of flows over time
- Distribution of packets over time
  - Globally
  - Within a flow
- Distribution of packet sizes

- Payload, Deep Packet Inspection
  - Expensive (time, processing power)
  - Does not work with encrypted traffic
  - Can also be used for intrusion detection
    - Trojans, viruses

Self-Similar Traffic

- It has shown that for some environments the traffic pattern is self-similar rather than Poisson
- Self-similarity is a concept related to two others
  - Fractals
  - Chaos theory
- Statement by Manfred-Schroeder:

  The unifying concept underlying fractals, chaos, and power laws is self-similarity. Self-similarity, or invariance against changes in scale or size, is an attribute of many laws in nature and innumerable phenomena in the world around us. Self-similarity is, in fact, one of the decisive symmetries that shape our universe and our effort to comprehend it.

Self-Similarity – An Example

- Network monitoring, analysis of the interarrival time of single frames
- Minimum transmission time for one frame: 4ms

- Recorded arrivals (ms):

  0 8 24 32 72 80 96 104 216 224 240 248 288 296 312 320

  648 656 672 680 720 728 744 752 864 872 888 896 936 944 960 968

- Clustering all samples with gaps smaller than 20ms:

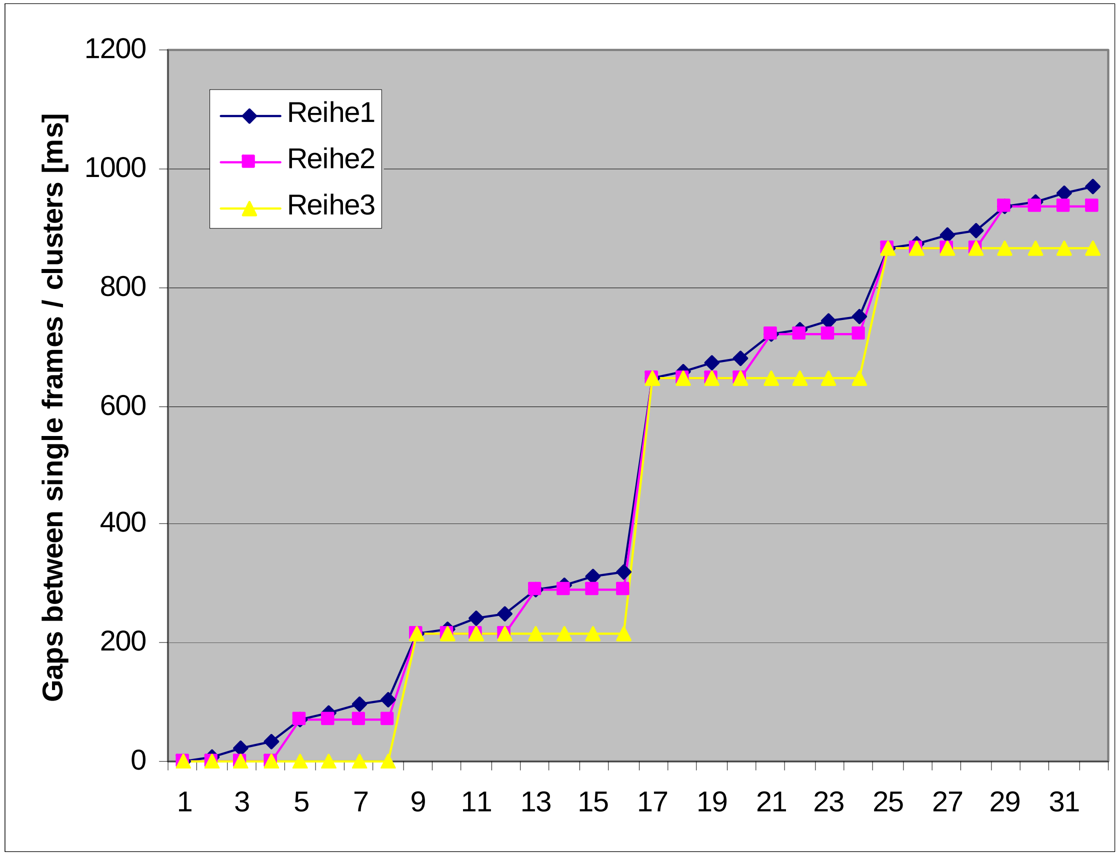
  0 72 216 288 648 720 864 936

- Clustering all samples with gaps smaller than 40ms:

  0 216 648 864

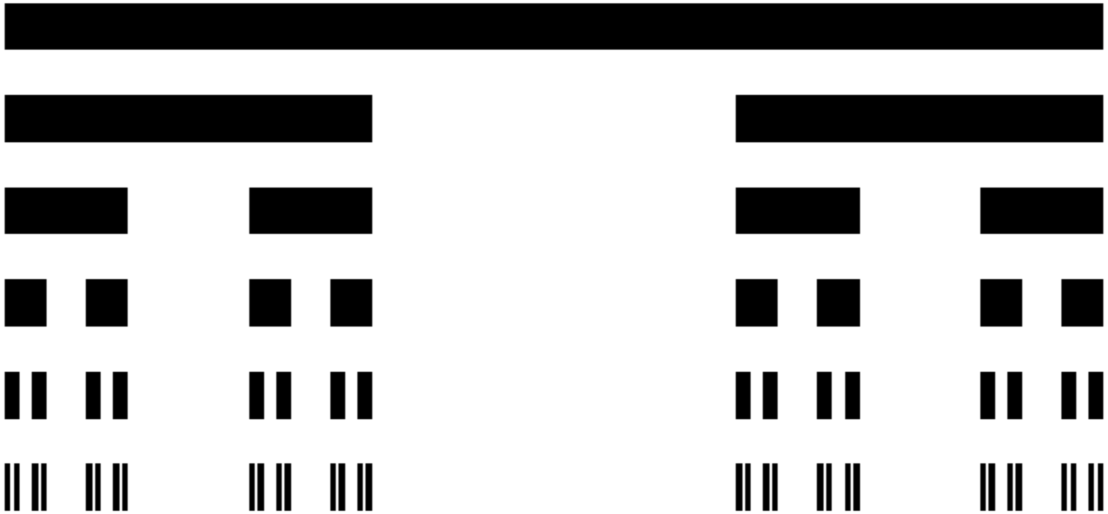
Self-Similarity – An Example II

- Repeating patterns: arrival, short gap, arrival, long gap, arrival, short gap, arrival)



Cantor Set

- Famous construct appearing in virtually every book on chaos, fractals, and nonlinear dynamics
- Construction rules:
  - Begin with the closed interval [0,1], represented by a line segment
  - Remove the open middle third of a line
  - For each succeeding step, remove the middle third of the lines left by the preceding step
- Cantor set:
  - S_0 = [0, 1]
  - S_1 = [0, 1/3] U [2/3, 1]
  - S_3 = [0, 1/9] U [2/9, 1/3] U [2/3, 7/9] U [8/9, 1]



Cantor Set II

- Properties of Cantor sets seen in all self-similar phenomena
  - It has a structure at arbitrarily small scales. If we magnify part of the set repeatedly, we continue to see a complex pattern of points separated by gaps of various sizes. The process seems unending. In contrast, when we look at a smooth, continuous curve under repeated magnification, it becomes more and more featureless.
  - The structure repeat. A self-similar structure contains smaller replicas of itself at all scales. For example, at every step, the left (and right) portion of the Cantor set is an exact replica of the full set in the preceding step.

- These properties do not hold indefinitely for real phenomena. At some point under magnification, the structure and the self-similarity break down. But over a large range of scales, many phenomena exhibit self-similarity.

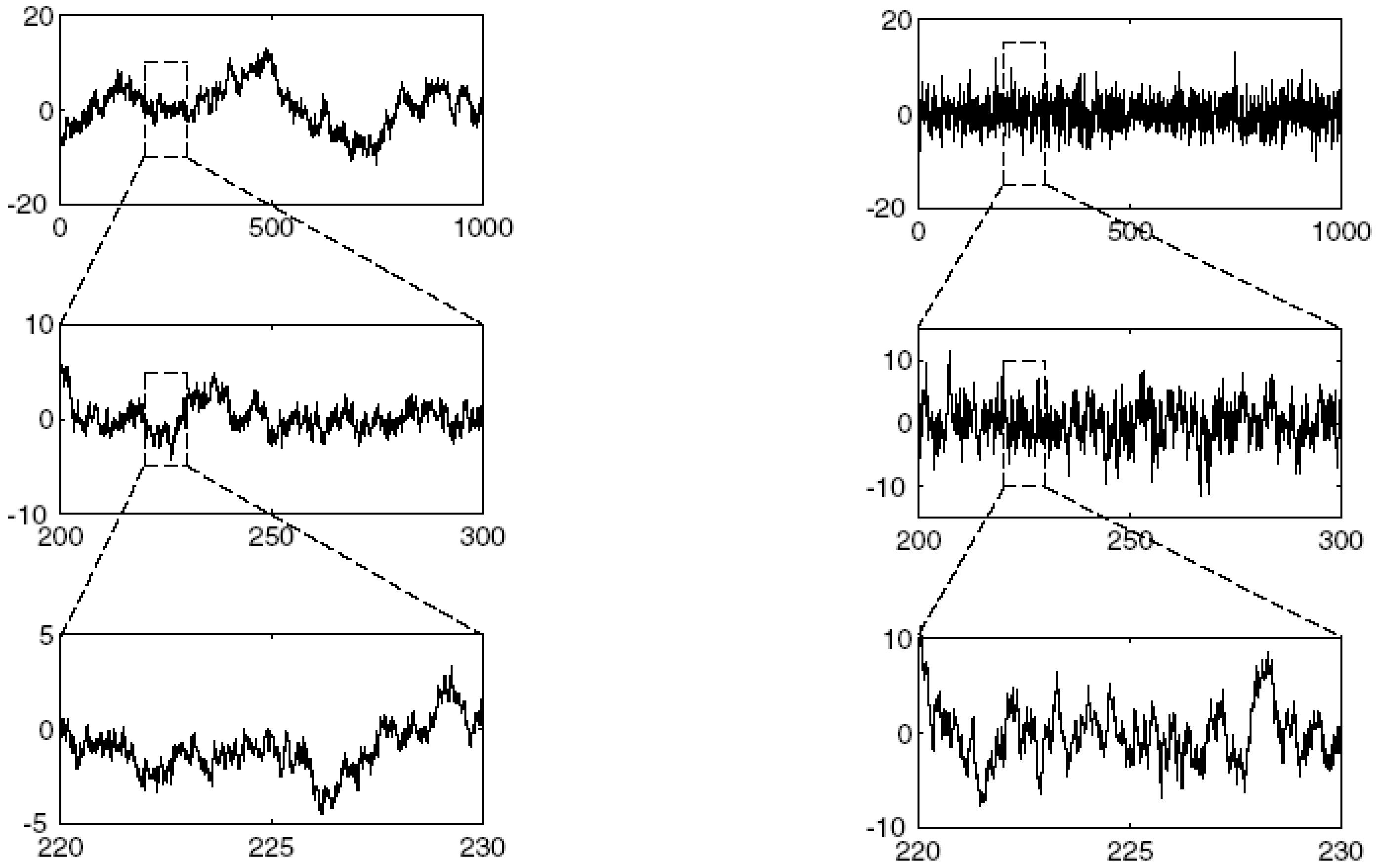
Stochastical Self-Similarity

- So far, we examined exact self-similarity:
  A pattern is reproduced exactly at different scales

- Data traffic is a stochastic process, therefore we talk about statistical self-similarity.
- For a stochastic process, we say that the statistics of the process do not change with the change in the time scale. The average behavior of the process in the short-term is the same as it is in the long term.

- Examples
  - Data traffic
  - Earthquakes
  - Ocean waves
  - Fluctuations in the stock market

Self-Similar Stochastic Process



(a) Self-Similar Process

(b) Non-Self-Similar Process

Network Traffic Characteristics

- Traffic characteristics experienced in the network
  - Changes over time
  - Varies in many dimensions
  - Each application has its characteristic traffic pattern
  - Must match the model used for planning

- Numerous ways of classification
  - Port, Flow sizes, Packet sizes, Packet count, Arrival times, …

- Packet/ Flow/… distribution
  - Poisson
    - Good for performance evaluation, network planning
  - Gauss, Pareto, …
  - Self-similarity

Interpretation of Measurement Results

Literature:

Raj Jain: The Art of Computer Systems Performance Analysis, John Wiley

D.C. Montgomery “Design and Analysis of Experiments”

- „If you require a straight curve, only measure two times“

- „If you can´t reproduce a result, only conduct the experiment once“

- „post hoc ergo propter hoc“
  „from coincidence follows correlation“

Before you start a measurement

- Wanted:
  - Answer to a question
- To be considered:
  - Correctness
  - Significance (of the measured values)
  - Relevance (in regard to the question)
  - Effort

- Modelling the reality
  - Simplify to much
  - Forget important parameters
  - Make assumptions that make life easy

- Modelling our tools: overfitting
  - Change the behaviour of our measurement tool so it works perfectly in the tests
  - What happens in other scenarios?

- Example: a new TCP flavor and we want to know how it performs
  - Cross-traffic: static/dynamic, distribution, number of flows/packets/…?
  - Underlying network: layer 2, topology, …?
  - What did we want to measure again? – ah, the performance:
    - Delay, recovery time, throughput, startup time, …?

Statistics

- Why do we need it?
  - Transform data into information
  - Get rid of noise

- Statistic:
  - Merriam-Webster:
    „A quantity that is computed from a sample [of data]“
  - A single number to summarize a larger collection of values
- Statistic***s***:
  - Merriam-Webster:
    „A branch of mathematics dealing with the collection, analysis, interpretation, and presentation of masses of numerical data.“
  - Analysis and interpretation

Sampling the measured data

- Sample = subset of whole process
  - Not possible to enumerate fully
    - too much data
    - ongoing process
- Selection types
  - Random
  - Systematic – every nth packet, flow, …

- Sample Bias
  - Selection area
    - only use a "good" part of the data
    - Partition the data based on knowledge
  - Interval – start and end at a convenient time
  - Exposure – selection is not independent from the process itself
  - Rejection of „bad" data, outliers, …
  - Overmatching
  - Quantization error

- Examples
  - Heise Browser Statistics
  - Counting the number of cars on the street every Monday at 9:00

The simplest statistic: a mean

- Reduce sample to a single number
- But what does it mean?
  - Tries to capture the „center“ of a distribution of values
    - Mean
    - Median
    - Mode
  - Use this „center“ to summarize
  - „Sample“ implies
    - Values are measured from a discrete random variable
    - Only an approximation of the underlying process
    - True mean value cannot be known
      (requires infinite number of measurements)

- To provide „mean“ value
  - Understand how to choose the best type
  - Detect bad results

Arithmetic Mean

- Common „average“

$$\overline{x}_{arithm} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

- Potential problems
  - Equal weight to all values
  - Outliers can have a large influence
  - Distorts our intuition about central tendency

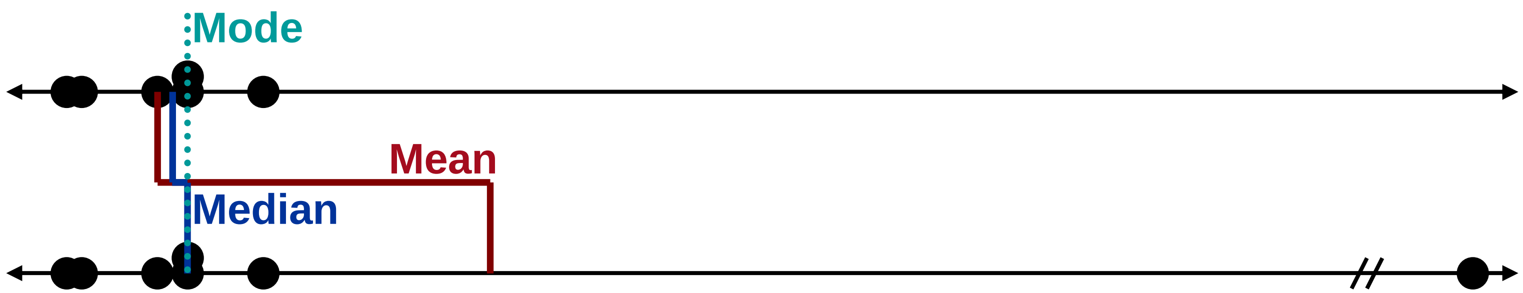
Median and Mode

- Median
  - ½ of the values larger, ½ smaller
  - Algorithm
    - Sort n measurements by value
    - If n is odd: Median = middle value
    - Else: Median = mean of two middle values
  - Reduces skewing effect of outliers

- Mode
  - Value that occurs most often
  - May not exist
  - May not be unique: multiple modes
    - e.g. „bi-modal“ distribution:
      Two values occur with same frequency

Mean vs. Median vs. Mode

- Measured Values: 10, 23, 16, 18, 18, 11
  - Mean: 16
  - Median: 17
  - Mode: 18
- Obtain one more measurement: 173
  - Mean: 38
  - Median: 18
  - Mode: 18



Mean, mode or median?

- Mean
  - If the sum of all values is meaningful
  - Incorporates all information
- Median
  - Intuitive Sense of central tendency with outliers
  - What is „typical" of a set of values?
- Mode
  - When data can be grouped into distinct types, categories

Other means

- Geometric
  - Growth rates, benchmarks
  - Example:
    The usage of a webservice doubles the first year and octuplicates the second year
    - Geometric mean: 4
    - Arithmetic mean: 5
  - Less sensible

$$\bar{x}_{geom} = \sqrt[n]{\prod_{i=1}^{n} x_i}$$

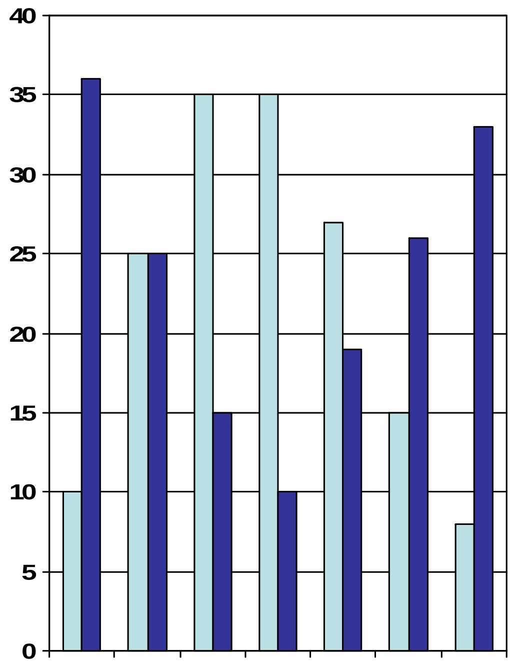
- Harmonic
  - Proportional data, ratios
  - Example:
    Download 10MB of data with 1MB/s, 5MB/s and 10MB/s (10s+2s+1s=13s)
    - Harmonic Mean: 2,33 MB/s
      and: 30 MB with 2,33MB/s: 13s
    - Arithmetic mean: 5,33 MB/s
      but: 30MB with 5,33MB/s: 5,625s
    - Download per time is again arithmetic!

$$\bar{x}_{harm} = \frac{n}{\sum_{i=1}^{n} x_i}$$

Variability

- How spread out are the values?
- How much spread relative to the mean?
- What is the shape of the distribution
- A mean hides information about variability

- Example
  - Similiar mean values
  - Widely different distribution



- How to capture this in one number?

Dispersion

- Range: max-min
- 10- and 90- percentiles
- Maximum distance from mean
  max (| x_i-mean |)
- Neither efficiently incorporates all available information

- Variance
  - Squares of the distances to mean
  - Gives „units-squared" – hard to compare with

$$\text{var} = s^2 = \frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n}$$

- Standard deviation ***s***
  - Square root of variance
  - Same unit as mean

Expectation value

- Also called mean ☹ or first moment
- Limit of sample mean for infinite number of values

- Not „the most probable value“
  - Expectation value might be unlikely or even impossible
  - rolling a dice: Expectation value: 3.5

- „Law of large numbers“
  - Information for large scales
  - No information about single events/ small samples!

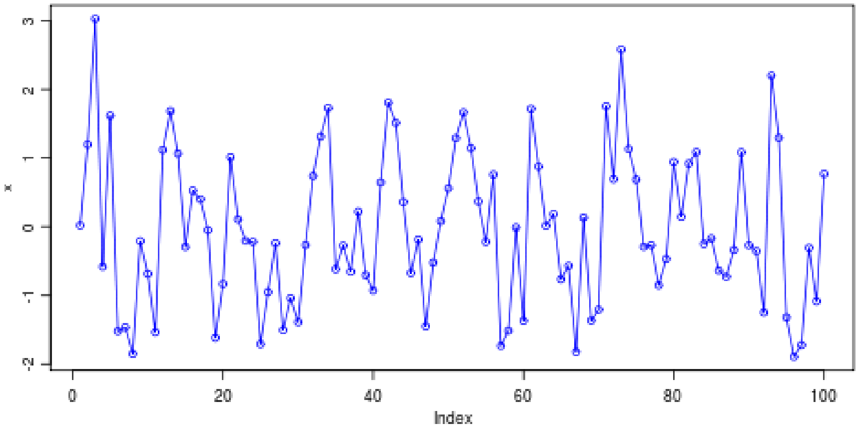
Autocorrelation

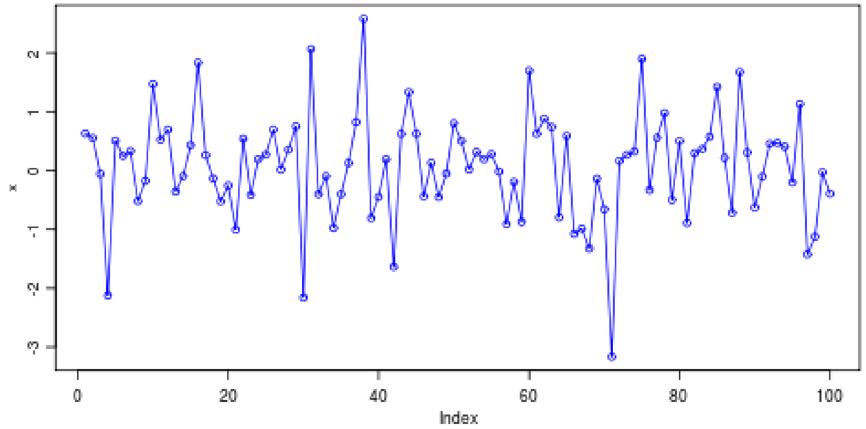
- Correlation of a signal with itself
  - Checking for randomness
    - Most standard statistical tests rely on randomness (validity of the test is directly linked to the validity of the randomness assumption)
    - In short: If you don not check for randomness, the validity of your conclusions are questionable
  - Find repeating patterns (e.g. underlying frequencies)

- Concept
  - Calculate variance C_0 for data set
  - For each lag
    - Calculate variance C_h over the data set
    - Normalize C_h/C_0

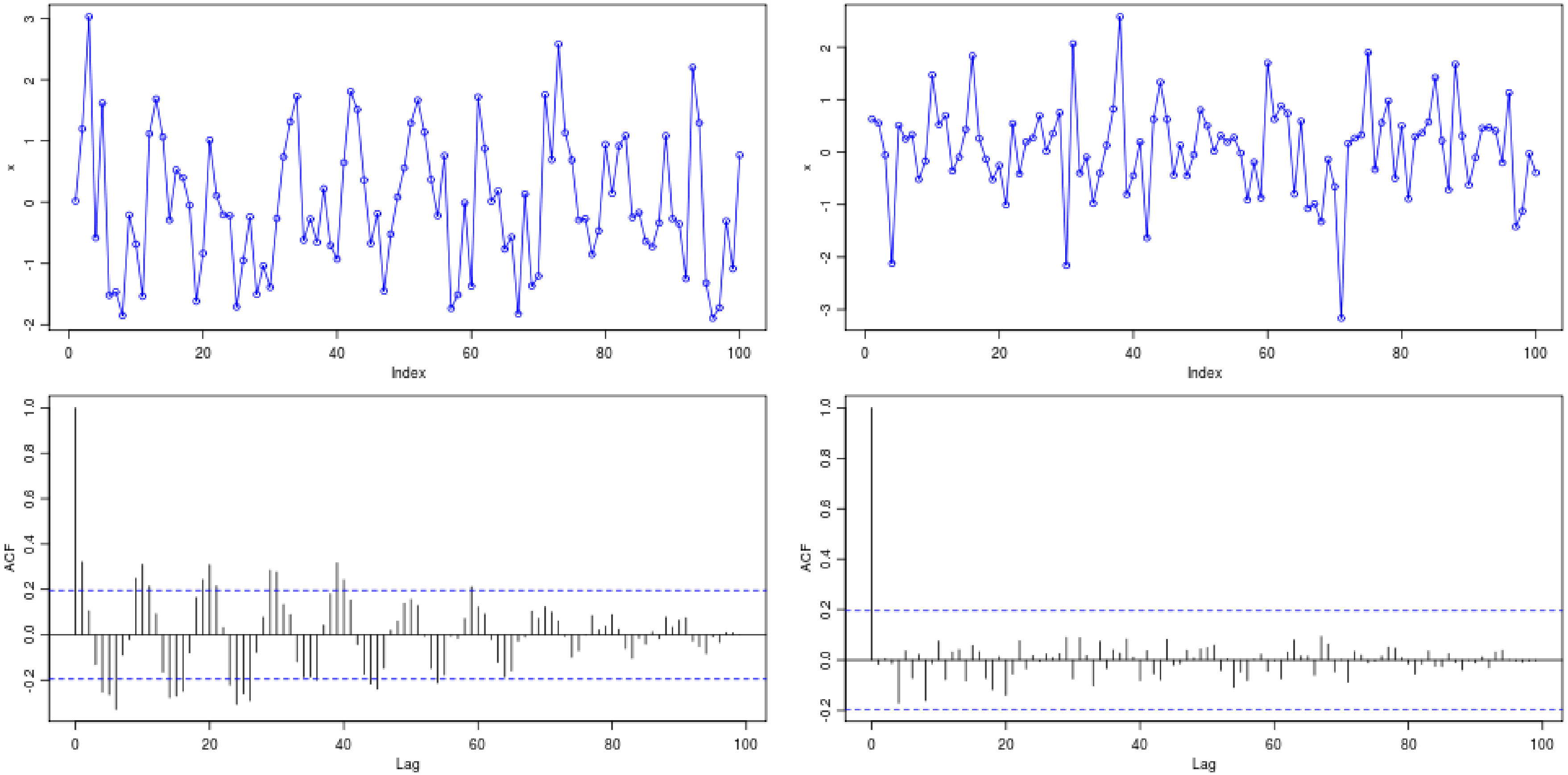
- Interpretation
  - If random: near zero for all and any lag separations
  - If non-random: one or more autocorrelations significantly non-zero
  - Lag shows the frequency for the autocorrelation

Autocorrelation Example (1)





Autocorrelation Example (2)



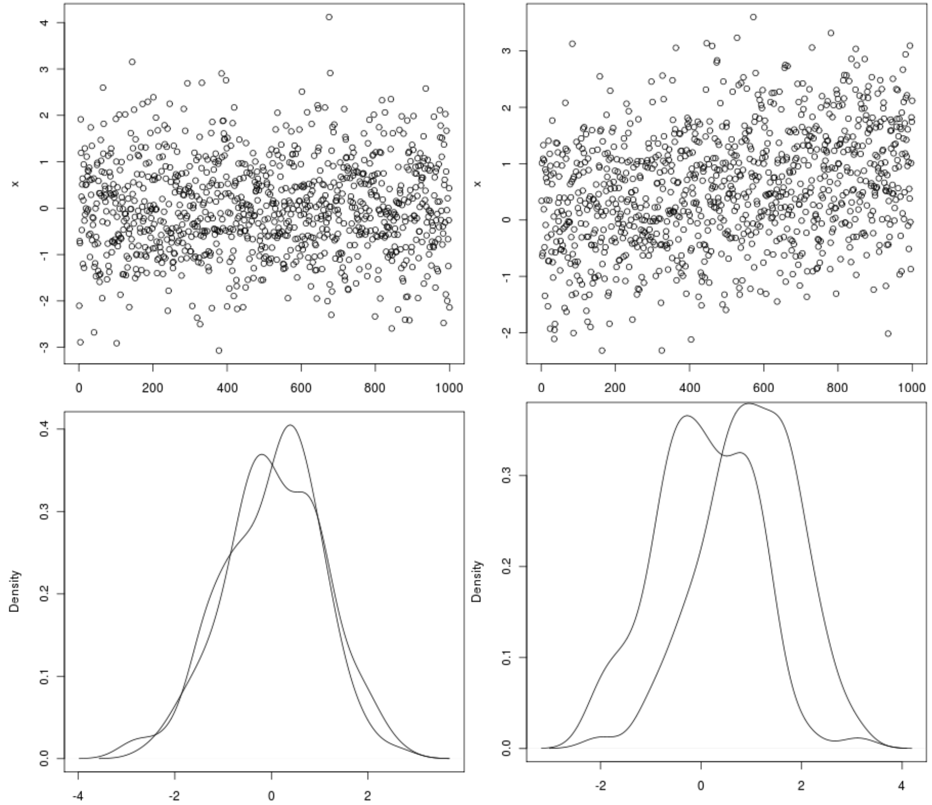
- Vertical Axis: Autocorrelation coefficient
  - Always between -1 and +1
- Horizontal Axis: Time lag
- Reference lines: 95% confidence band

- Left: hidden sinus
  - Positive autocorrelation for f
  - Negative autocorrelation for f/2
- Right: „truly“ random

Stationarity

- a random process where all of its statistical properties do not vary with time

- First order stationary process
  - Mean,
    variance,
    autocorrelation
    do not change over time
- Example:
  - Random
  - Random with trend



- Transformation to achieve stationarity
  - Take the diffs between values
  - Trend:
    fit some type of curve
    (e.g. a straight line), model
    residuals from that fit
  - Non-constant variance:
    try square root or logarithm
    to stabilize the data

Summary

- Network Measurement
  - Why, what and how?

- Network Traffic
  - Traffic Pattern
  - Traffic Models
  - Self-similar traffic

- Evaluation of measurements
  - Statistics
    - Only the tip of the iceberg
    - Common Errors!
    - Think before you start, before you calculate, before you extrapolate!
    - Be careful in every step
  - If you want to play with this
    - Octave – www.gnu.org/software/octave/
    - R – www.r-project.org