Notes on data analysis

Student version

Last saved: 30/10/2022 17:48:00

Index

1 General Information

Prof. Ing. Giovanni Stea

Dipartimento di Ingegneria dell'Informazione

Largo L. Lazzarino 1, 56122 Pisa - Italy

Ph. : (+39) 050-2217.653 (direct) .599 (switch)

Fax : (+39) 050-2217.600

E-mail: giovanni.stea@unipi.it

References: These notes are based on (in decreasing order):

1. *R. Jain, "The art of computer systems performance analysis", Wiley*
2. *J.-Y. Le Boudec, "Performance Evaluation of Computer and Communications systems", available online*
3. *S. M. Ross, "Introduction to probability and statistics for engineers and scientists", Elsevier, cap. 7-9*

Some of the figures have been taken from book 2 and from the online PDF slides related to book 1. Should their authors dislike this, I will remove the figures.

Pre-requisites: strong background in probability calculus, algebra (factorials, permutations) and mathematic analysis (integrals, derivatives).

Module length: 11 hours, not counting labs.

2 Statistics

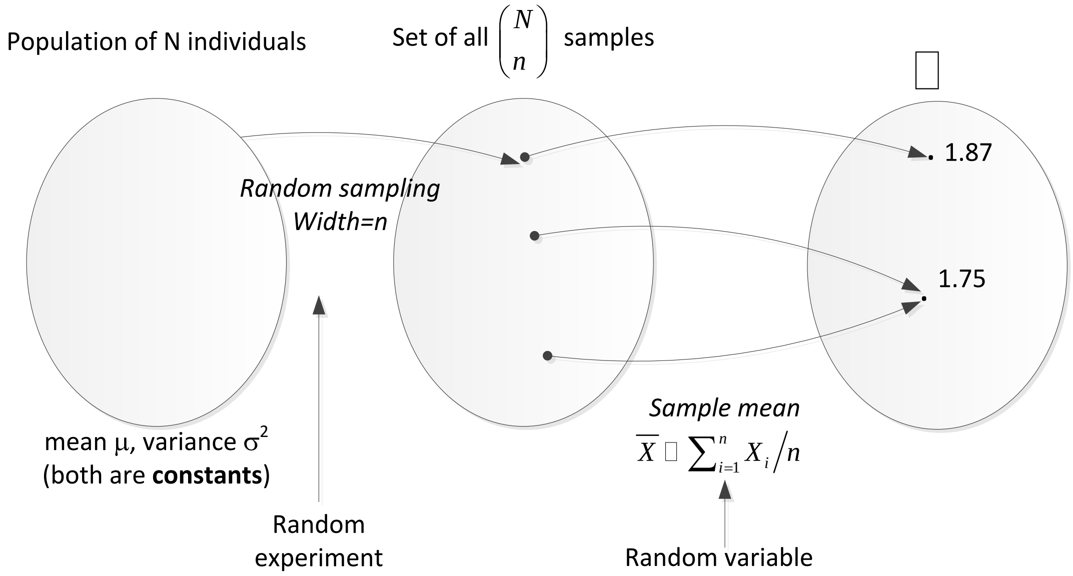
Statistics is the science by which probability results are **inferred from experimental data**. That is, you have the **data**, and you get **a CDF.** It is, in a sense, the dual of **probability theory**, where you start from a CDF and obtain some data.

The following concepts are of key importance in statistics:

- **Population**: the set of individuals on which you want to infer statistics (e.g., their average height). The information related to the population (e.g., its mean and variance) are called **parameters**, and they are constants. Only they are **unknown.** The population mean is a constant, and quite often you need to estimate it the best you can.
- **Sample:** a (possibly very small) **subset** of a population. You measure a sample and try to infer properties that are expected to be true (with some approximation) **of the population**. The information related to the sample are called **statistics**. For instance, the **sample mean** (i.e., the mean of the sample that you take) is a statistic, and it is a **random variable.**

Your task is often to estimate **parameters** through **statistics**.

We call a **random sampling** of n elements taken from a population of N elements one such that every one of the $\binom{N}{n}$ subsets of cardinality n are equally likely to represent it. Sampling is a **random experiment**, since you cannot predict the outcome in advance.


Population of N individuals
Set of all $\binom{N}{n}$ samples
□
1.87
Random sampling
Width=n
1.75
Sample mean
$\overline{X} \square \sum_{i=1}^{n} X_i / n$
mean μ, variance σ^2
(both are **constants**)
Random experiment
Random variable

For instance, the **students in this classroom** are a sample of a population of the students at the university of Pisa. You can measure the sample mean of their height (or age, or whatever), and use it as an estimate of the population of the students at UniPi. However, sampling UniPi students in a classroom within the School of Engineering will not yield an **unbiased sample** (there is no chance that a Law or Vet student can ever be included).

In general, we make the **assumption** that the measure whose parameters we want to estimate does exist. Specifically, that measure has a CDF, and the **elements of a sample** represent **IID RVs** according to that CDF.

The IID assumption is fundamental. For now, we assume that it holds, and then we see what to do if it does not hold.

It has to be observed that the English words **sample** and **example** have the same origin (old French word "essample", from Latin "exemplum"). This should always be kept in mind: a sample is **an example** of what happens in a population. Different conclusions may be reached by taking another sample, or a larger one. When you **measure the output** of a system, you are **taking a sample** of its possible outputs.

A fundamental problem is the following:

Since sampling is a random process, the **observed statistics** might be due to different causes:

a) **Structural properties**, i.e. facts that are true of the population;

b) **Randomness** (i.e., "luck of the draw").

There are ways to **quantify** the effect of randomness on sample statistics, and we will become familiar with some of them. It is important to learn right from the start that **you cannot say anything with certainty, unless $\boldsymbol{n = N}$**. The best that we can settle with is something along the lines of "Given the data, I can be 95% certain that X holds. If you want to be more certain, get me more data".

2.1 Data presentation

Assume that you have a sample of n IID RVs, X_i, $1 \le i \le n$, called **observations**, obtained e.g. by measuring your system, or as an output of a simulation of your system. n is called the **sample width.** How can you **visualize** those data?

There are, of course **several answers** to this question. The thing that you may want to keep in mind is that:

- visualizing those data **will aid you to understand how your system works**, so you had better find the way that gives you the information you really need

- you will often need to find the way to present those data in **the most convincing way**, putting what is really relevant in the most accessible format.

Thus, we review here a **set of alternatives**, with the caveat that you have to select the best according to your needs and situation. Again, there is no simple solution, and patience and perseverance are required.

2.1.1 Empirical CDF

The ECDF can be obtained as a function of x by plotting the following function:

$$F(x) = \frac{1}{n}\sum_{i=1}^{n} 1_{\{X_i \le x\}}$$

In other words, $F(x)$ is the proportion of data that does not exceed value x. This gets you a **staircase** function, which is what you can expect if the distribution is **discrete**.

Operatively, the way to compute an ECDF is the following: you take your sample, and compute its **ordered statistics**. These are obtained by sorting the observations in the sample in ascending order, so that $X_{(1)} = \min\{X_i\}$, $X_{(i)} \le X_{(i+1)}$ and $X_{(n)} = \max\{X_i\}$[1]. Then, you have a staircase with steps whose height is $1/n$, placed at the ordered statistics. When $X_{(i)} = X_{(i+1)}$, you just sum the height.

If you are dealing with a **continuous** distribution, you may want to **interpolate** the data so that they look smooth. The way to do this is the following:

$$F(x) = \begin{cases} 0 & x < X_{(1)} \\ \dfrac{i-1}{n-1} + \dfrac{x - X_{(i)}}{(n-1)\cdot\left(X_{(i+1)} - X_{(i)}\right)} & X_{(i)} \le x < X_{(i+1)} \\ 1 & x \ge X_{(n)} \end{cases}$$

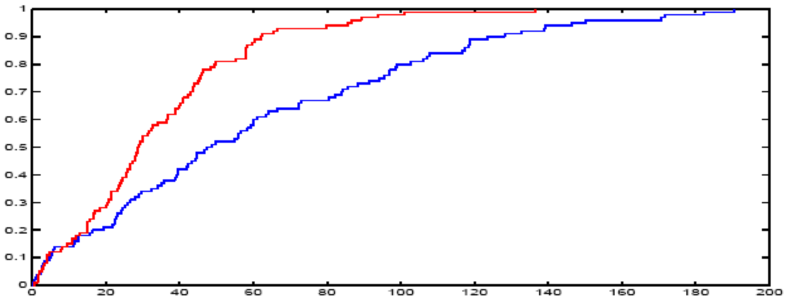
This works if all observations are different. If two are the same, then you just sum up two steps.

The above formula states that the vertical axis is divided into n-1 portions, and the ECDF grows linearly within each interval $\left[X_{(i)}, X_{(i+1)}\right)$.

ECDFs can be compared. Take two alternative designs, A and B. If the ECDF of the metric you are interested in is always *above* in A, then:

- If “lower is better” (e.g., a delay), then A is better than B
- If “higher is better” (e.g., a throughput), then B is better than A

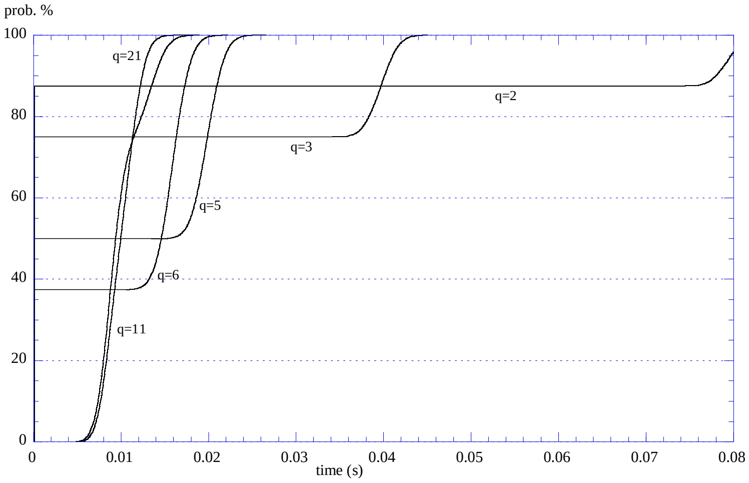
[1] We will use parentheses in the subscript to denote the fact that $X_{(1)}$ is the **smallest** observation in the sample, and X_1 is the first observation obtained.



Code running time with two versions of the code: new (red curve) and old (blue curve)

If the two curves intersect at some point, you cannot make conclusions, at least not easily. However, if the intersection is at the **top or at the bottom**, then you can make some inferences, e.g. by saying that one system is better than the other most of the times **except that** it may sometimes show a worse performance.

In any case, you can get some information out of ECDFs, hence this is the first graph to attempt. In other cases ECDFs can be tricky, and you will need more information.



ECDF of a delay, for various values of a parameter *q*

2.1.2 Histograms

Assume initially that you are dealing with a **continuous** random variable. Histograms allow you to obtain an **empirical PDF** from the sample. The algorithm to create a histogram is the following:

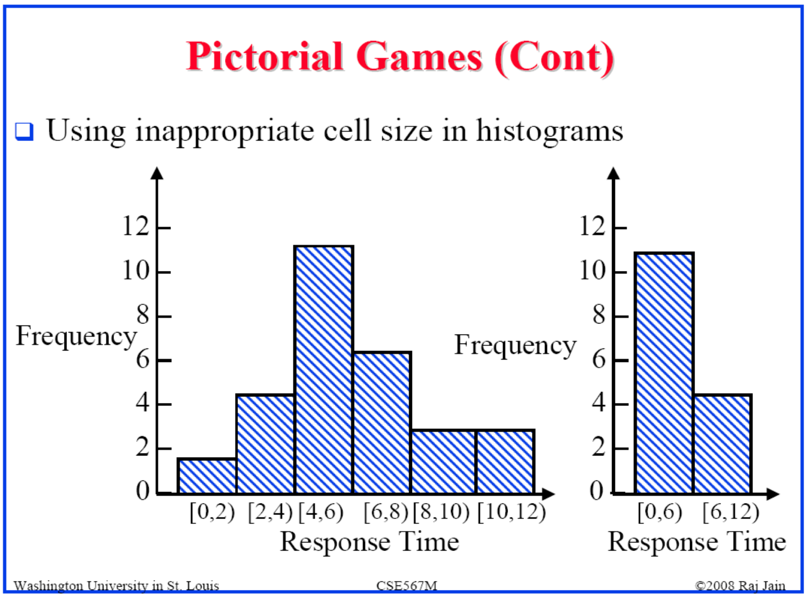
- select a **bucket width** Δ, i.e. the range within which observations will be counted in the same histogram column.
- Initialize all buckets to zero.
- for each observation X_i, add 1 to bucket $\lfloor X_i/\Delta \rfloor$

The problem with histograms is that the way they look will depend a lot on how you choose Δ. At the two extremes, you will have

- too small Δ: zero or one observations per bucket → "comb-like" histogram. If you have a sample of n and select $\Delta \approx \left(X_{(n)} - X_{(1)}\right)/n$, then you will have very few observations per bucket.
- too large Δ: all samples in the same bucket → "block-like" histogram.

In both cases, you can't get anything useful out of it (such as, e.g., the **relative frequency** of your variable in a given bucket).

Changing the bucket width Δ will **make the EPDF look extremely different**, hiding or showing the very details that you might want to discover.



Pictorial Games (Cont)

❑ Using inappropriate cell size in histograms

Washington University in St. Louis CSE567M ©2008 Raj Jain

Note that the vertical scale of the right figure is wrong

By misjudging the width of your bucket Δ, you end up thinking that the probability is decreasing, where in fact it is fairly symmetric and peaks around 5.

In the books, you normally find the rule of thumb that the **average** number of elements per non-empty bucket should be around **five** for the histogram to be meaningful. My advice here is that you draw a histogram **several times**, with different values of Δ, and measure the average number of elements per non-empty bucket every time. This way you are sure that you are not missing important pieces of information. Then you will use the one with the **best trade-off** between compactness and **accuracy**. A useful guideline is to choose the smallest Δ for which the histogram looks "smooth".

Regarding accuracy, a straightforward caveat is to avoid mixing "very different" values in the same bucket. If the observable performance of your system is (looks) completely different when $X_i = 5$ and when $X_i = 9.5$, you may want to use $\Delta < 5$, otherwise you are not showing anything significant. Thus, Δ depends on the data and the purpose for which they are aggregated.

The same applies if you are plotting histograms for a **discrete** RV. In this case, it is somewhat easier since you can build on the fact that you already know a unit.

If you want to compare two systems using histograms, remember to:

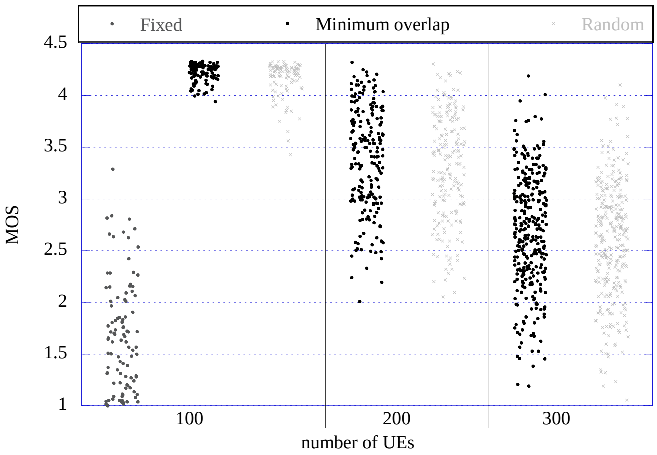
- use the **same bucket width** for both
- either have the same sample width n (if it is up to you to select it), or **normalize the ordinates** to the respective sample widths – otherwise comparisons are impossible.

2.1.3 Scatterplots

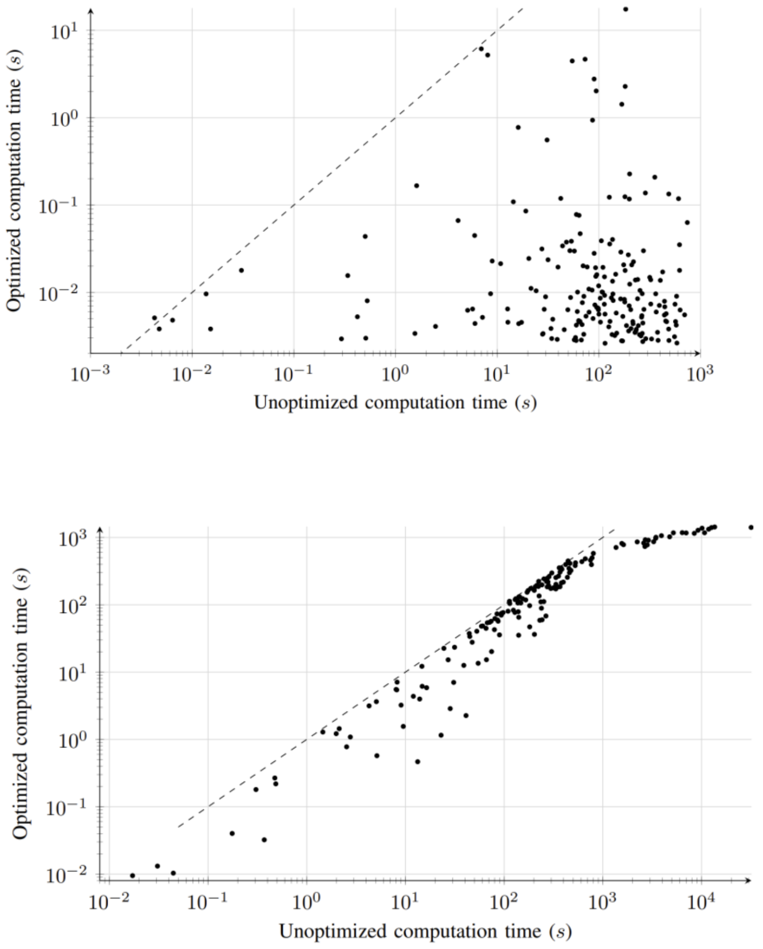
A simple, although not always effective way to show a sample of IID RVs is to use a **scatterplot.** The latter is a plot where points (or markers) are scattered along a given horizontal span, and one point is drawn at the proper height.

Tools that allow you to plot scatterplots normally show you where other statistics are placed (e.g., sample mean, sample median, etc.)

Mean Opionion Sore (MOS, HB), with three different schemes, with an increasing # of users.



This is a good way to see at a glance how dispersed values are, and where you can expect to have the highest fraction of data. You may want to use them to figure out Δ values for histograms. Scatterplots are also good to spot **outliers**, which are very hard to spot in an ECDF or a histogram, or to understand **clustering.**

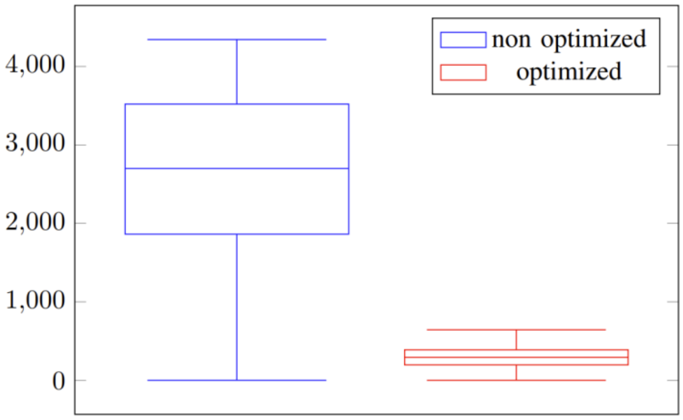


Comparisons of computation times before/after optimization, in a situation in which said times can change by orders of magnitude depending on the observation

2.1.4 Box Plots

Also called “**box and whiskers”** plot. It consists in plotting:

- a **box** whose edges are the 1st and 3rd **quartiles** (i.e., the 25th and 75th percentiles), call them Q_1, Q_3
- a **line** representing the **median** (i.e., the 50th percentile)
- two **whiskers**, protruding from the box, and representing something different, according to different conventions. Common practices are:
    - the **maximum and minimum** (not very common);
    - the 2nd and 98th percentiles;
    - the lowest datum still within $1.5 \cdot (Q_3 - Q_1)$ of the lower quartile, and the highest datum still within $1.5 \cdot (Q_3 - Q_1)$ of the upper quartile. The difference $Q_3 - Q_1$ is called the **inter-quartile range (IQR)**.



Unless you use whiskers to plot the maximum and the minimum, which is however **uncommon**, you may also have **outliers**, i.e., data outside the interval marked by the whiskers. These should be shown with some marker in the graph.

2.1.5 Lorenz Curves

All the above plots can be used – with a grain of salt – to compare alternatives. In the ECDF example, they show clearly that the new code is better than the old code, i.e. that things run generally faster. Sometimes it is useful to assess the **variability** of a sample, i.e. how **dispersed** a sample is. This is because in many cases, we would expect or prefer that the output of a system be **perfectly regular**. The fact that it is not is **unpleasant**, and is usually called **unfairness**.

For instance, we would expect **all customers** making the same request to a web server to observe, statistically speaking, the **same performance** (i.e., the same response time). The ideal situation is that **every customer experiences the same response time**. This will not happen in practice, because of several causes: queuing of near-simultaneous requests, disk swaps, different TCP congestion windows for different connections, cache hits/misses, etc., and those causes affect different customers in a different way.

We would then like to quantify how **unfair** our system is by quantifying how variable the data are.

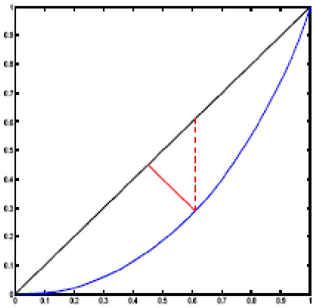
A good way to do so – which works only on **non-negative random variables** – is to plot a **Lorenz curve**.

The latter is defined as follows:

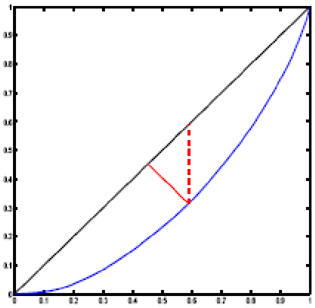
- sort your sample and get the **ordered statistics** $X_{(1)} \leq X_{(2)} \leq \ldots \leq X_{(n)}$;
- Compute $T = \sum_{i=1}^{n} X_i$, the sum total of the sample;
- Plot points $\left(1/n\,, X_{(1)}/T\right)$, $\left(2/n\,, \left(X_{(1)} + X_{(2)}\right)/T\right)$, …, $\left(j/n\,, \sum_{i=1}^{j} X_{(i)}/T\right)$, $1 \leq j \leq n$;
- Interpolate, assuming (0,0) as the first point of the curve.

Obviously, the Lorenz curve touches point (1,1). Let us take a look at its shape.

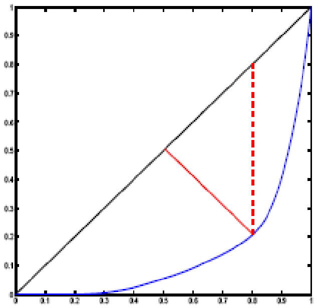
- If all the values were **equal**, the LC would be the bisector of the first quadrant. This is called the **line of maximum fairness**, and should be plotted for reference.
- If $T = X_{(n)}$ and all the other values are null, then the LC follows the x axis until point $(1 - 1/n\,,0)$, and then goes up to point (1,1). Therefore, the horizontal axis and the vertical line passing through (1,0) are a **lower bound** for the LC. They are called the **line of maximum unfairness**, and should be plotted as a reference too.



(a) Execution times in Figure 2.1, old code



(b) Execution times in Figure 2.1, new code



(c) Ethernet Byte Counts (x_n is the byte length of the nth packet of an Ethernet trace [64])

The LC will lie between the two bounds. The fairer the system, the closer the LC will be to the bisector.

Note that you can compare two alternatives **as for variability** using LCs, but you cannot say anything about (e.g.) the fact that the A's average response time will be higher/lower than B's. This is because you are normalizing the vertical axis to a **different sum total**, hence you are losing that piece of information.

2.2 Summarizing data

Quite often you are requested to provide **one number** (or as few numbers as possible) that describes the performance of a system. This is useful, for instance, for **comparisons**, as well as for **ease of storage** (it is **costly** to store an ECDF). Three popular alternatives are to specify one of the **three indexes of central tendencies**, namely the (sample) **mean, median and mode**.

Let us review what these are, and why we should prefer one or the other.

2.2.1 Indexes of central tendencies

We define the **sample mean** as:

$$\overline{X} = \frac{1}{n} \cdot \sum_{i=1}^{n} X_i$$

$\overline{X}$ is itself a RV, since it is the sum of RVs. Let us discuss some of its properties.

It is:

- $E[\overline{X}] = \frac{1}{n} \cdot E[\sum_{i=1}^{n} X_i] = \frac{1}{\cancel{n}} \cdot \cancel{n} \cdot E[X_i] = \mu$;
- $Var(\overline{X}) = Var\left(\frac{1}{n} \cdot \sum_{i=1}^{n} X_i\right) = \frac{1}{n^2} \cdot n \cdot Var(X_i) = \frac{\sigma^2}{n}$.

This is something that we have already shown as a formulation of the CLT. The **larger the sample is,** the more $\overline{X}$ **converges** to a RV which is equal to the population mean μ with no variance.

We also know (still from the CLT) that, for a large sample, $\overline{X}$ is Normal (whatever the CDF of $X_1, \dots, X_n$ – except for HT ones[2]). Of course, if $X_1, \dots, X_n$ are themselves Normal, then $\overline{X}$ is Normal for **any value of n** (this is, in fact, a property of Normal variables). In all other cases, $\overline{X}$ tends to be Normal for large samples.

Therefore, **at least** for large values of n, it is:

$$\frac{\overline{X} - \mu}{\sigma/\sqrt{n}} \sim N(0,1)$$

The **sample median** is obtained from the **ordered statistics** ($X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(n)}$) taking:

- if n is odd, observation $X_{(\lceil n/2 \rceil)}$, i.e. the midpoint of the ordered statistics;
- if n is even, value $\frac{X_{(n/2)} + X_{(n/2+1)}}{2}$, i.e. the average of the two observations left and right of the midpoint.

The **sample mode** is the value with the highest probability. It should not be estimated directly on a sample (especially a sample of a continuous RV). It should instead be sampled on a **histogram**, as the midpoint of the bucket with the highest frequency. The sample mode may not be unique.

Now, which should we use?

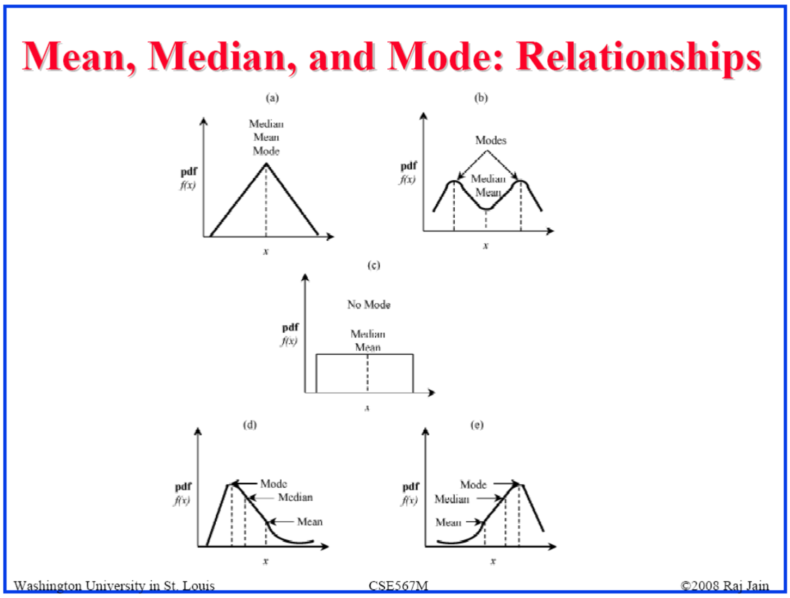
It depends a lot on the **distribution of the sample**. Common sense says that:

- if the distribution is symmetric, the mean and the median are the same, hence either will do;
- the mode may not be unique, whereas the other two certainly are;

[2] From now on, we will omit repeating every time that HT distributions are exceptions to the CLT.

- the mean is **additive**, while the others are not: the mean of a sum is the sum of the means, whereas the median of a sum is definitely **not** the sum of the medians;
- The sample mean is **sensitive to outliers**. If you have a **small** sample, then few outliers can change your mean considerably. The median is instead **very robust** to outliers.

The last observation is important. If your distribution is **very skewed**, or has a **long tail**, the median will be more representative of a typical observation than the mean: the values that you observe will resemble the **median more than the mean**.



Mean, Median, and Mode: Relationships

Washington University in St. Louis CSE567M ©2008 Raj Jain

This is easy to prove. Suppose that you have a sample of n observations, whose sample mean is $\overline{X}$. Suppose that n is odd, so that the median is $X_{0.5} = X_{(\lceil n/2 \rceil)}$. Assume that you add to the sample one more observation, whose value is O. The new mean becomes:

$$\overline{X}' = \frac{\overline{X} \cdot n + O}{n+1}$$

Now, if the two addenda are comparable (e.g., because the sample is small, or O is a very large outlier), then the mean may change significantly. On the other hand, the new median becomes:

$$X'_{0.5} = \begin{cases} \dfrac{X_{(\lceil n/2 \rceil)} + X_{(\lceil n/2 \rceil + 1)}}{2} = X_{0.5} + \dfrac{X_{(\lceil n/2 \rceil + 1)} - X_{(\lceil n/2 \rceil)}}{2} & \text{if } O \geq X_{0.5} \\ \dfrac{X_{(\lceil n/2 \rceil - 1)} + X_{(\lceil n/2 \rceil)}}{2} = X_{0.5} - \dfrac{X_{(\lceil n/2 \rceil)} - X_{(\lceil n/2 \rceil - 1)}}{2} & \text{if } O \leq X_{0.5} \end{cases}$$

Note that O does not appear in the final result, hen O ce the median is unaffected by the magnitude of. The differences cannot be large (they could even be null, if the two consecutive values in the ordered statistics are equal), hence the new median is unlikely to be much different from the previous one.

2.2.2 Indexes of dispersion

As for the indexes of central tendencies that we have seen so far, there are alternatives for **indexes of dispersion**. The most common are:

- Range;
- Sample Variance, Sample Standard Deviation and Sample Coefficient of Variation;
- 10th and 90th percentiles;
- Semi interquartile range;
- Mean Absolute Deviation and Lorenz Curve Gap.

The **range** is the difference between the maximum and the minimum, i.e. $X_{(n)} - X_{(1)}$. This value should **not** be used, **unless** there is reason to believe that the RV we are sampling is **bounded**, and we need to estimate those bounds. Otherwise, the **range** will tell you nothing significant. In fact, it will:

- be equal to a very large outlier minus zero, if the variable is non negative;
- keep growing with n (thus, it is **unstable**);
- reveal nothing about what happens between the two extremes.

Define now the **sample variance** as:

$$S^2 = \frac{\sum_{i=1}^{n}(X_i - \overline{X})^2}{n-1} \qquad \text{(yes, it's } n-1\text{)}$$

S^2 is a RV as well. To compute its **expectation**, it is convenient to observe that:

$$\begin{aligned}
&(n-1)\cdot S^2 = \\
&\sum_{i=1}^{n}(X_i - \overline{X})^2 = \\
&\sum_{i=1}^{n}\left({X_i}^2 + \overline{X}^2 - 2X_i\cdot\overline{X}\right) = \\
&\sum_{i=1}^{n}{X_i}^2 + n\cdot\overline{X}^2 - 2\overline{X}\cdot\sum_{i=1}^{n}X_i = \\
&\sum_{i=1}^{n}{X_i}^2 + n\cdot\overline{X}^2 - 2\cdot n\cdot\overline{X}^2 = \\
&\sum_{i=1}^{n}{X_i}^2 - n\cdot\overline{X}^2
\end{aligned}$$

And take the expectation of both sides:

$$
\begin{aligned}
(n-1)\cdot E[S^2] &= \sum_{i=1}^{n} E\left[{X_i}^2\right] - n\cdot E\left[\overline{X}^2\right] \\
&= n\cdot E\left[{X_i}^2\right] - n\cdot E\left[\overline{X}^2\right] \\
&= n\cdot [Var(X_i) + E[X_i]^2] - n\cdot \left[Var\left(\overline{X}\right) + E\left[\overline{X}\right]^2\right] \\
&= n\cdot [\sigma^2 + \cancel{\mu^2}] - \cancel{n}\cdot \left[\frac{\sigma^2}{\cancel{n}} + \cancel{\mu^2}\right] \\
&= (n-1)\cdot \sigma^2
\end{aligned}
$$

Thus, $E[S^2] = \sigma^2$, therefore the **expectation of the sample variance is the population variance**. The reason why there is $n-1$ instead of n is that **only $\boldsymbol{n-1}$** of the differences $\left(X_i - \overline{X}\right)$ are independent. The n-*th* one depends on the others, since $\sum_{i=1}^{n}\left(X_i - \overline{X}\right) = 0$. It is normally said that the **number of degrees of freedom for the variance** is $n-1$ for a sample of n.

The **sample standard deviation** is the square root of the sample variance, of course.

It is also common to use the **sample coefficient of variation (CoV)**, defined as:

$$C = S/\overline{X}.$$

The sample CoV can be expected to approximate the population CoV, which is $\chi = \sigma/\mu$. Note that it is only meaningful for **non-negative distributions** that have a well-defined unit, such as mass, length, etc. – often called "ratio-scale" measures since it makes sense to take ratios of measures[3]. For these, the CoV is bounded by $0 \le C \le \sqrt{n-1}$. This may represent a problem in practical cases. There are cases of distributions whose **variance is infinite** (**heavy-tailed distributions)**. These can be observed more frequently than you might expect. If you are dealing with such a distribution, then there is a risk that the CoV will increase indefinitely with the sample width. In this case, comparing alternatives through the CoV may be altogether meaningless.

An interesting fact is that the CoV for an exponential RV should be equal to 1, whatever its parameter λ. This will be useful later on, when we talk about **fitting**.

[3] Typical non-ratio-scale measures are *temperature* and *dates*. Both are defined in terms of arbitrary reference points (e.g., freezing and boiling points of water for temperature), which are arbitrary offsets. Now, the variance is insensitive to offsets, whereas the mean is. Therefore, the CoV will depend on the (arbitrary) reference point. This is why it makes no sense to define a CoV for non-ratio scale measures. They are called non-ratio-scale measures because you cannot state that 20 degrees is twice as hot as 10 degrees, hence it makes no sense to compute *ratios* of temperatures, or dates.

The CoV is normally used only with **continuous distributions.** Its homologous for **non-negative discrete distributions** is the **sample Index of Dispersion (IoD)**, sometimes called **Lexis Ratio**, defined as $L = S^2/\overline{X}$ (which approximates the population IoD $\Lambda = \sigma^2/\mu$). You can easily check that this coefficient is:

- equal to 1 for a Poisson RV;
- smaller than 1 for a Binomial RV (equal to $1 - p$);
- larger than 1 for a Geometric RV (equal to $1/p$).

Note that both the CoV and the IoD tend to be **unstable** if the sample mean is close to zero.

<u>**(Sample) percentiles**</u> **(**or **quantiles)** can be computed from the ordered statistics. Note the distinction: you normally denote percentiles with a **percentage**, and quantiles with a **fraction** (i.e., the 0.9 quantile is the same thing as the 90-percentile).

Strange as it may seem, there is **no standard way** to compute percentiles. There are three or four different ways, whose results **agree when the sample is large**. However, significant differences can be observed on small samples, depending on the algorithm used.

Assume you have a sample of 5 as in the table:

Ord. stat.	$X_{(1)}$	$X_{(2)}$	$X_{(3)}$	$X_{(4)}$	$X_{(5)}$

You can readily state that the 20th percentile is $X_{(1)}$, but you have a hard time understanding what the 35th percentile would be. Commonly used alternatives:

- the **next higher** (i.e., $X_{(2)}$ in this case). If you have a sample of n and are looking for the p-quantile, then you take $X_{(\lceil n\cdot p\rceil)}$;
- some **linear interpolation** of the two adjacent values (i.e., the point on the line passing through (20, $X_{(1)}$) and (40, $X_{(2)}$) at abscissa 35);
- Microsoft Excel uses yet another approximation.

If you are **looking at a specific percentile** and you have ordered statistics, then use $X_{(\lceil n\cdot p\rceil)}$. Sometimes the **reverse is required**. You have ordered statistic $X_{(j)}$, and you need to assign it a percentile (this will be useful later on). This is tricky, since – according to the definition of percentile that we use – all percentiles from 0% to 20% are the same value $X_{(1)}$.

In this case, the **correct** procedure is to consider the range of the ordered statistics and divide it into n **consecutive buckets** of equal width. Each ordered statistics is considered to represent the **midpoint** of the bucket, and that is the percentile you assign to it.

In the above example:

Ord. stat.	$X_{(1)}$	$X_{(2)}$	$X_{(3)}$	$X_{(4)}$	$X_{(5)}$
Bucket	0-20	20-40	40-60	60-80	80-100
Midpoint	10	30	50	70	90

In other words, if you have a sample of 5, then you have the 10th, 30th, 50th, 70th and 90th percentile of the sample.

This is equivalent to assigning quantile $\frac{j-0.5}{n}$ to $X_{(j)}$.

Note that assigning another point in the bucket (e.g., the initial or final point) is **conceptually wrong**. If, for instance, you assign the upper end of the bucket interval as the percentile, you implicitly state that no observation can be higher than $X_{(5)}$. This is a bold, unwarranted claim. The reverse holds if you use the lower end, this time stating that $X_{(1)}$ is the minimum.

The **semi-interquartile range** is the difference $(Q_3 - Q_1)/2$, i.e. half the box height in a box plot.

The **Lorenz Curve Gap** i.e. the maximum vertical distance between the bisector (the line of maximum fairness) and the LC. The formula is:

$$LCG = \max_{1 \le j \le n}\left(j/n - \sum\nolimits_{i=1}^{j} X_{(i)}/n \cdot \bar{X}\right) = \frac{1}{2\bar{X}} \cdot \frac{\sum_{i=1}^{n}\left|X_i - \bar{X}\right|}{n}$$

The last equality can be proved (it's not that difficult). You are welcome to try.

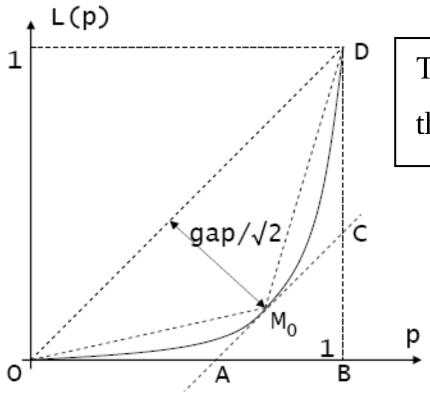


Figure 2.6: Lorenz curve (plain line). The line of perfect equality is OD, of perfect inequality OBD. The Lorenz curve gap is the maximum distance to the line of perfect equality, re-scaled by $\sqrt{2}$. The Gini coefficient is the area between the line of perfect equality and the Lorenz curve, re-scaled by 2.

The second term of the above formula, i.e., $\frac{\sum_{i=1}^{n}|X_i - \bar{X}|}{n}$, is called **mean absolute deviation (MAD)**. The LCG resembles the CoV, in that it normalizes a measure of variability (i.e., the MAD) to an index of central tendency (i.e., the sample mean). However, the LCG varies in $0 \le LCG \le 1 - 1/n$.

This means that, whatever the distribution (even a heavy-tailed one), **the LCG will always be finite**, unlike the CoV.

With indexes of dispersion, like with indexes of central tendencies, it is important to know **when to use one or the other.** Let us review our options:

- regarding **robustness to outliers**, the range is by far the most vulnerable. Any outlier will change it significantly. The variance and STD may also be affected by outliers. The IQR is almost insensitive to outliers.
- If your variable is **bounded** (hence outliers cannot happen), the range may be the best choice.
- Otherwise, if your variable is **symmetric and unimodal**, then the variance (or STD, or CoV) may be a good choice, unless the distribution has **heavy tails**, in which case the LCG is preferable.
- If your RV is highly skewed and/or outliers are likely, then the percentiles and the IQR are more representative of the variability (much as the median would best represent central tendencies than the mean).

Exercise (to be done using Excel)

Take a sample of a suitably large number of IID RVs. Have students

- plot the sample in various ways
- determine and discuss its indexes of central tendencies
- determine and discuss its indexes of dispersion

♦

2.3 Fitting a distribution

A good way to summarize a sample is to state from **what distribution it is (likely to be) sampled**. For instance, saying that your sample has a mean of 13 and a variance of 48 says **considerably less** than saying that it is $\sim U(1,25)$. The problem is how to choose the theoretical distribution your sample should fit.

Quite often, the choice is **not entirely wild**, since you might have at least a rough idea of the underlying phenomena that generated the sample.

If, instead, you are **completely in the dark**, you may start sorting out the sheep from the goats by using two very simple tricks:

- compute **statistics** for your sample;
- draw **histograms.**

Computing **statistics** means computing the **sample mean, median, variance, CoV**. For instance, if the **sample mean and median are approximately equal**, then there is a good chance that we are dealing with a **symmetric distribution**, hence we can put this result to good use when choosing the distribution to fit our to. If, instead, the mean and the median are **significantly different**, whether one or the other is larger will tell us whether the distribution is **skewed** to the left or to the right.
The CoV/IoD can be useful as well. For instance, in an **exponential distribution** it is equal to 1, regardless of the value of the λ parameter. If you obtain a sample CoV close to one, you may want to test whether your sample is exponentially distributed. If your distribution is discrete, you may use the sample IoD to discriminate whether it is Poisson, Binomial or Geometric.

Drawing **histograms** (with the rules previously discussed) will allow you to estimate the PDF or PMF. Some of them have a shape which is easily recognizable (the exponential, the Normal, the uniform, etc.), hence this will point you in the right direction.
Note that we did not mention **comparing the ECDF with the one of the theoretical distribution**. This should never be done, since **most CDFs have an S shape**, and it is visually very difficult, if possible at all, to distinguish one kind from another. You can easily recognize the exponential and the uniform ones, of course, since they are not S-shaped, but everything else will be very difficult.

Once you have **selected one or few candidates**, the easiest way to test whether your distribution fits the theoretical one is to make a **visual test, using QQ plots**. QQ plots have:

- on the *x* axis, the quantiles of the “theoretical” distribution $t_{(1-i)}$;
- on the *y* axis, the quantiles of your sample $q_{(1-i)}$.

Thus, every point is $\left(t_{(1-i)}, q_{(1-i)}\right)$, where i is the quantile level.
The visual test is the following: if the QQ plot is **approximately a straight line**, then the assumption that your sample follows that theoretical distribution is **consistent with your sample data** (whether it is true or not it is an entirely different kettle of fish).

If the theoretical distribution has an **invertible CDF**, then its quantiles can be computed quite easily. In fact, since $p = F(x)$, then $t_{(1-i)} = F^{-1}(i)$. For instance, the i-quantile of an exponential with a rate λ is obtained by solving $i = 1 - e^{-\lambda \cdot t_{(1-i)}}$, i.e. $t_{(1-i)} = -\log(1-i)/\lambda$.

If the CDF is **not known in a closed form** (e.g., the Normal one), then its quantiles can be obtained from **tables**, so it is not a big problem in any case. For the **Standard Normal**, a good approximation, which is often used in practice, is the following expression:

$$t_{(1-i)} \cong 4.91[i^{0.14} - (1-i)^{0.14}]$$

Of course, should you need quantiles for a non-standard Normal $F \sim N(\mu, \sigma)$, then you simply use the results of the above formula and transform them as $t_{(1-i)}' = \mu + \sigma \cdot t_i$.

Exercise (to be done using Excel):

Check if the above formula is a good approximation (it is: its relative error oscillates between -0.3% and +0.7%, hence it is negligible).

♦

The interesting thing about QQ plots is that quite often you don't need to know the **parameters of the theoretical distribution**, e.g. μ, σ for a Normal or λ for an exponential. This happens every time the effect of the parameter is to **rescale or offset** the quantiles (as in the above two cases).

If this is the case, assuming a default value for the parameter will still give you a straight line, whose slope and offset will change if you change the parameters. However, the linear pattern will still be there, so there is no need to worry.

Example

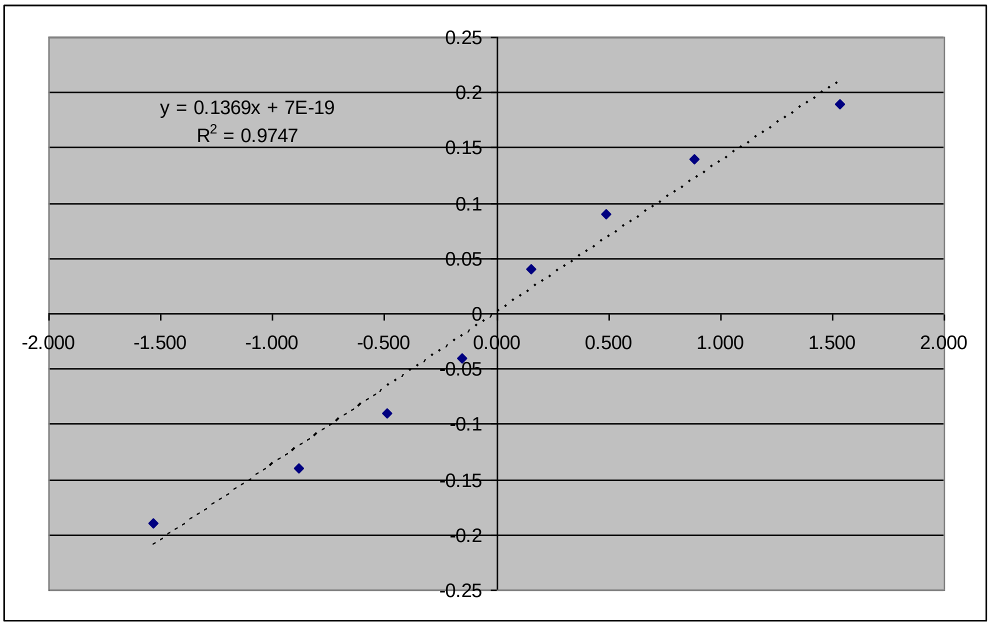
Considering the following sample of 8, and check if it is normally distributed.

	2	1	3
j	Quantile number	ord. Stat	normal q
1	0.0625	-0.19	-1.535
2	0.1875	-0.14	-0.885
3	0.3125	-0.09	-0.487
4	0.4375	-0.04	-0.157
5	0.5625	0.04	0.157
6	0.6875	0.09	0.487
7	0.8125	0.14	0.885
8	0.9375	0.19	1.535

The first thing is to compute the ordered statistics. Then we assign them quantiles $\frac{j-0.5}{n}$. Then, we use the above formula to obtain the quantiles of the standard Normal. Finally, we plot the graph.

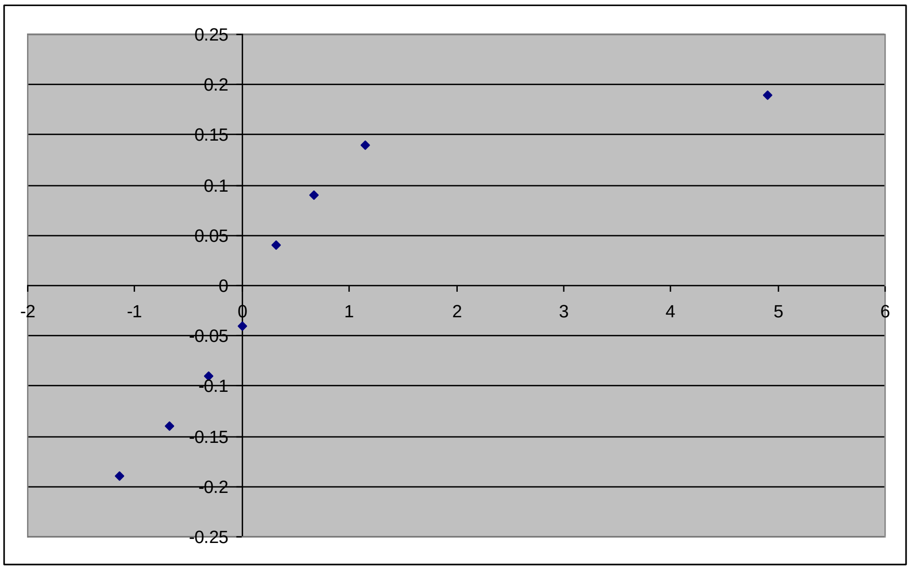
The graph shows a good linearity. The regression analysis (which is automatic with Excel) computes the mean to be (almost) null, and the standard deviation to be around 0.1369.

4



Note that assigning the wrong quantiles gets you completely different results. Assume, in fact, that we use the **upper end of the bucket**, i.e., $\frac{j}{n}$. Then the thing would look like this:

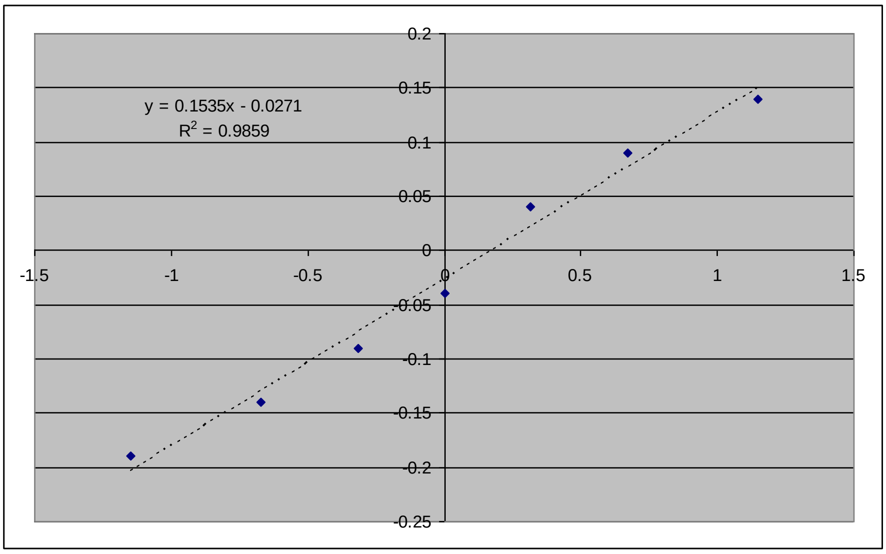
j	(wrong) Quantile number	ord. Stat	Normal q
1	0.125	-0.19	-1.14921
2	0.25	-0.14	-0.67234
3	0.375	-0.09	-0.3173
4	0.5	-0.04	0
5	0.625	0.04	0.317299
6	0.75	0.09	0.672345
7	0.875	0.14	1.149208
8	1	0.19	4.91



Linearity is somewhat distorted, and there is a large outlier at the far right. This is due to our assuming that:

- the highest-order statistic is representative of the 100% percentile, which it is not
- the approximate formula for the standard Normal quantile is reliable for a 100% quantile, which it is not (the 100% quantile of a standard Normal is $+\infty$).

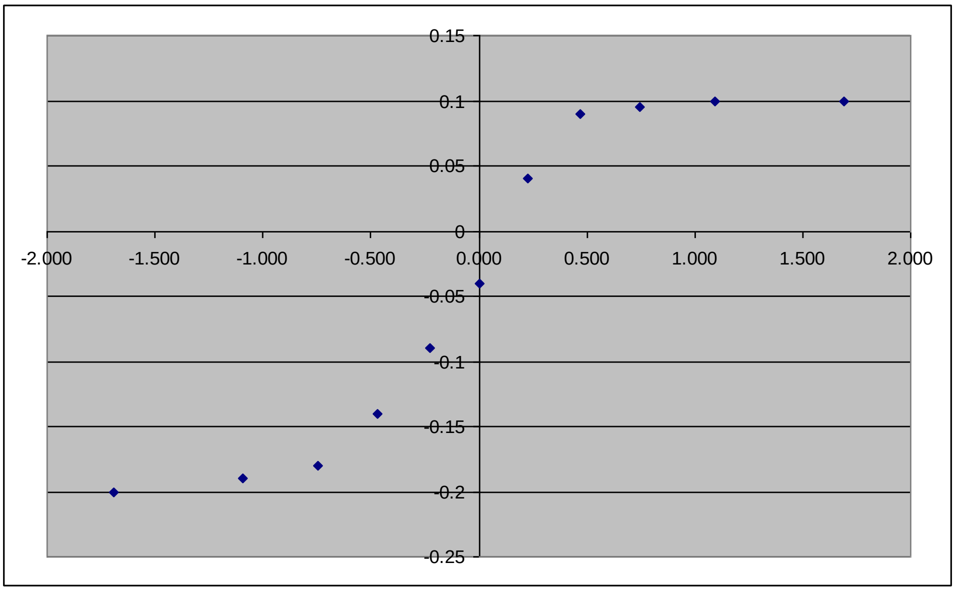
However, if you remove the outlier, then you get approximately the same result as before:



It is often the case that the QQ plot shows an **approximately linear** behaviour in the middle, with some deviation at the **tails**. This is indicative of the fact that tails may be **longer** or **shorter** than those of the theoretical distribution.

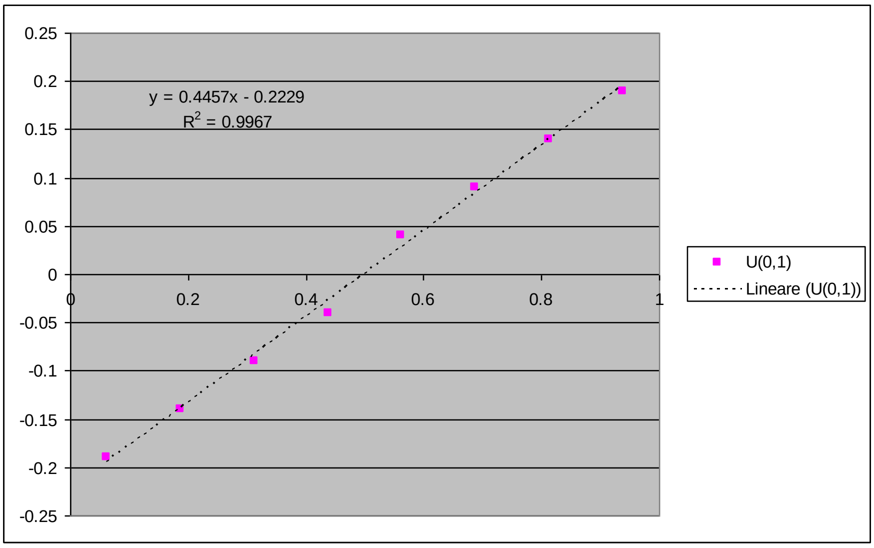
The following graph shows the same sample as above, to which two small and two large observations have been added, which are almost equal to the previous sample minimum and maximum. The

S shape is typical of a case with **shorter tails** than a Normal. This happens quite often, since in many cases variables are **bounded** (due to physical or measurement constraints) and the Normal distribution has **infinite tails instead**. If it happens, test for a **uniform distribution,** which appears to be likely.



If, instead, the tails of the Normal QQ plot go **up and down** (instead of flattening), then there is a chance that a distribution with heavier tails may yield a better fit. Try **lognormal** or **Pareto**. We have seen neither of these, but they may help.

Just as a counterproof, we might want to check whether our initial sample of 8 matches a **uniform** distribution. We might as well assume U(0,1), since we have already ascertained that this changes nothing. The QQ plots is the following:



Our sample shows almost **perfect linearity**. This means that the above sample could equally well have been taken from:

- a Normal distribution with a null mean and a std of around 0.13
- a uniform distribution $U(-0.203, +.203)$

How to discriminate the two cases? In this case the solution is simple: **get a larger sample**. Eight points are not nearly enough to make a difference.

Note that there are other tests, relying on algebra, to assess whether or not data from a sample can be matched to a given distribution. The most well-known ones are:

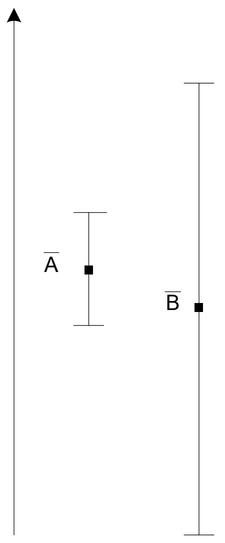
- Chi squared test
- Kolmogorov-Smirnoff test

They are **not** easy to use, since they require a lot of hypotheses and a lot of practice to interpret the results. QQ plots should be preferred when possible. Moreover, these require that the **matching theoretical distribution be entirely specified**: in other words, you cannot just check whether or not your data has a **Normal distribution**. You can only test against a **Normal distributon with mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\sigma^2}$**, which you need to estimate beforehand.

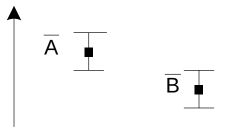
2.4 Confidence intervals

As part of the **scientific method**, you are duty-bound to show the **accuracy of a measurement** whenever you produce one. This is something that engineers (and scientists as well) often forget to do, which makes their statements unverifiable and ultimately not credible. You can **lie blatantly** by omitting to state the accuracy of a measurement. For instance, you can claim that a system performs better than another, when in fact your sample data do not support this statement.

The point is that, when you have **random variables and samples**, you never know how much of the result that you are claiming is due to **luck**, i.e., to the fact that you happened to hit on a **particularly lucky sample** (recall that sampling is a random experiment), and how much it is an intrinsic property that you would like to prove.



Assume that you get two samples of the same measure from two systems A and B, and you compute the sample mean for both systems, let it be $\bar{A}, \bar{B}$. You find that $\bar{A} > \bar{B}$. Since the sample means estimate the population means, you conclude that **system A performs better than B**. However, if you manipulate your data, you can obtain that the interval where you have 95% probability of finding the **population means** μ_A, μ_B are as shown in the figure. When you quantify (by inserting the 95% confidence interval) the **accuracy** of your statistic (i.e., the sample mean), your former conclusion ("A is better than B") simply **does not hold anymore**. It could well be that $\mu_A < \mu_B$. Your data **do not allow you to tell** whether $\mu_A < \mu_B$ or $\mu_A \geq \mu_B$. It may be that your samples are too small, or that the data is too variable, or that you require too high a confidence level (95% in this case).



Of course, if this were the case, instead, then your statement "A is better than B" **would be entirely justified** by your data (at the given confidence level).

Every statistic taken from a sample is itself a **random variable**. Therefore, its accuracy must be quantified and reported in the form of a **confidence interval**. Always, no matter what.

We say that $[l, h]$ is a **confidence interval** for parameter μ at level $(1 - \alpha)$ if $P\{l \leq \mu \leq h\} \geq (1 - \alpha)$.

A **confidence interval** at level $(1 - \alpha)$ is an interval where you have a probability at least $(1 - \alpha)$ to find the parameter that you are estimating using your sample statistics. Value $(1 - \alpha)$ is called the **confidence level**, and values such as 90%, 95%, 98%, 99% are normally used. We will come back later to the choice of an appropriate confidence level for a problem.

Confidence intervals are computed differently from a sample of IID RVs, depending on what parameter you want to estimate. We present a method for the **population mean**, and one for **the probability of success** of a Bernoulli experiment.

Once again, we need to stress that all the methods described so far, as well as those we are yet to see, only hold **under the assumption that the sample consists of IID RVs**. This is something that has to be ensured before even starting; otherwise you will obtain **false conclusions**. We will see later on how to verify or obtain IIDness for your sample.

2.4.1 Confidence interval for the population mean

Assuming that the population we are dealing with **has a finite mean and variance**, we have defined the **sample mean** to be:

$$\overline{X} = \frac{1}{n} \cdot \sum_{i=1}^{n} X_i$$

And found that $E[\overline{X}] = \mu,\ Var(\overline{X}) = \sigma^2/n$, i.e., the **larger the sample,** the more $\overline{X}$ **converges** to a Dirac's delta around μ.

We know that, **at least** for a large value of n, $\overline{X}$ is Normal (due to the CLT), whatever the CDF of $X_1, \dots, X_n$. If $X_1, \dots, X_n$ are themselves Normal, then $\overline{X}$ is Normal for **any value of $\boldsymbol{n}$**.

One (very inefficient) way of computing a 90%-CI for the sample mean is the following: take k **independent** samples of n observations each (e.g., take a sample of $n \cdot k$ observations and slice it into k samples of n), compute the sample means of each of the k samples, call them $\overline{X_1}, \dots, \overline{X_k}$. You can then compute the 0.05-quantile and the 0.95-quantile of the above sample means. This is a confidence interval for the population mean at a confidence of 90% (i.e., removing the two 5% tails). However, this method is inefficient, in that it requires too many observations.

A more efficient use of the amount of information that is inherent in a sample of n IID RV can be done by leveraging the **sample variance**. We have defined the **sample variance** as:

$$S^2 = \frac{\sum_{i=1}^{n} \left(X_i - \overline{X}\right)^2}{n-1}$$

And found that $E[S^2] = \sigma^2$.

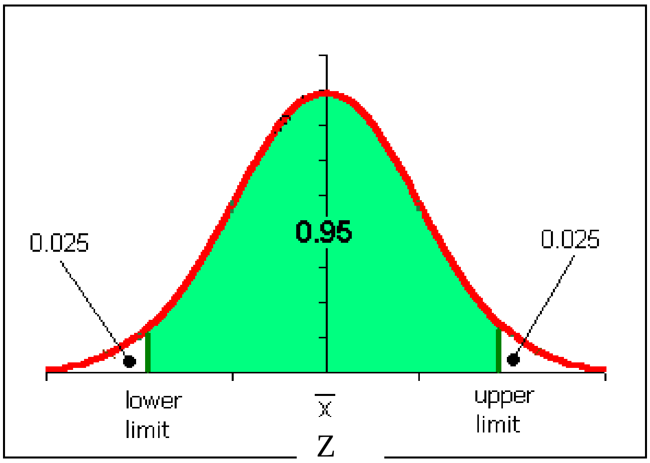
If the sample is large (i.e., $\boldsymbol{n \geq 30}$), and only in this case, then

$$\frac{\overline{X} - \mu}{\sigma/\sqrt{n}} \sim N(0,1)$$

Note that we do not know σ, which is an unknown population parameter. However, we can estimate it using S^2. Therefore, we can hope that

$$Z = \frac{\overline{X} - \mu}{S/\sqrt{n}} \sim N(0,1)$$

if n is large. In this case the (symmetric) interval that defines an area equal to $(1-\alpha)$ in the standard Normal, is in the figure. The point that leaves out a right tail with an area equal to 0.025, i.e. the one such that $P\{Z > z_{0.025}\} = 0.025$, is the 97.5%-percentile of the standard Normal, and you can find it tabulated at the end of these notes.



Therefore, the interval $[-z_{0.025}, z_{0.025}]$ includes RV $\frac{\overline{X}-\mu}{S/\sqrt{n}}$ with 95% probability. More in general, you take the $(1-\alpha/2)$ percentile of the standard Normal, and you can write

$$P\left\{-z_{\alpha/2} \leq \frac{\overline{X}-\mu}{S/\sqrt{n}} \leq +z_{\alpha/2}\right\} = (1-\alpha)$$

Therefore, via some straightforward algebraic manipulations, we obtain:

$$P\left\{\overline{X} - \frac{S}{\sqrt{n}} \cdot z_{\alpha/2} \leq \mu \leq \overline{X} + \frac{S}{\sqrt{n}} \cdot z_{\alpha/2}\right\} = 1-\alpha$$

The $(1-\alpha)$-level confidence interval for the population mean is the interval

$$\left[\overline{X} - \frac{S}{\sqrt{n}} \cdot z_{\alpha/2}, \overline{X} + \frac{S}{\sqrt{n}} \cdot z_{\alpha/2}\right]$$

It is worth repeating that the above computations only hold if:

- the population has a finite variance
- the sample is large (>30)

And that **the above assumptions should be tested** before using the method. For instance, if your sample variance keeps increasing with the width of the sample, this is a clear indication that the variance may be unbounded. In this case, **the CLT does not work**, hence you cannot compute a CI for the mean using the above technique. Note that in this case, the sample mean is considerably less significant than the median, for which you can always compute a CI[4].

[4] The procedure to compute a CI for the population median is described on J.-Y. Le Boudec's book, see the references at the beginning of this batch of notes.

If, instead, the sample is **small** (i.e., $n < 30$), then you can still apply a similar method, but it works only **<u>if the population consists of Normal RVs</u>**. Note that Normal RVs have finite variance. The assumption of Normality can (hence must) be tested beforehand **using a QQ plot**, of course.

We take for granted (because it is boring to prove it) that, for **Normal RVs**, it is $(n-1)\cdot S^2/\sigma^2 \sim\chi^2_{n-1}$, i.e. the LHS expression, which includes the sample variance is distributed like a chi-square with $\boldsymbol{n-1}$ **degrees of freedom**. Moreover, $\overline{X}$ and S^2 are **independent RVs**. As a consequence of these findings, you have that:

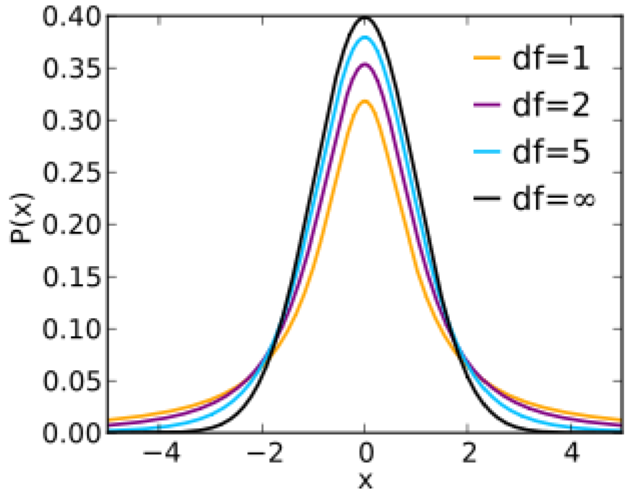
$$T=\frac{\overline{X}-\mu}{S/\sqrt{n}}$$

is a **Student's T distribution** with *n*-1 degrees of freedom, $T\sim t_{n-1}$. In fact:

1) $\frac{\overline{X}-\mu}{\sigma/\sqrt{n}}\sim N(0,1)$ (which we know to be true, since we assumed X_is to be Normal)
2) $(n-1)\cdot S^2/\sigma^2 \sim\chi^2_{n-1}$ (which we give for granted under the same assumption)

and:

$$\begin{aligned}T&=\frac{\dfrac{\overline{X}-\mu}{\sigma/\sqrt{n}}}{\sqrt{[\cancel{(n-1)}\cdot S^2/\sigma^2]/\cancel{(n-1)}}}\\&=\frac{\overline{X}-\mu}{S/\sqrt{n}}\cdot\frac{\cancel{\sigma}}{\cancel{\sigma}}\\&=\frac{\overline{X}-\mu}{S/\sqrt{n}}\end{aligned}$$



This is promising, since we can repeat the same reasoning as before, with the only change that we need the percentiles of a Student's T distribution. These can be found at the end of every probability textbook (and of these notes as well). Therefore, we immediately obtain:

$$P\left\{\overline{X}-\frac{S}{\sqrt{n}}\cdot t_{\alpha/2,n-1}\leq\mu\leq\overline{X}+\frac{S}{\sqrt{n}}\cdot t_{\alpha/2,n-1}\right\}=1-\alpha$$

Or, if you prefer

$$\left[\overline{X}-\frac{S}{\sqrt{n}}\cdot t_{\alpha/2,n-1},\overline{X}+\frac{S}{\sqrt{n}}\cdot t_{\alpha/2,n-1}\right]$$

Is the CI for the population mean at confidence level $(1-\alpha)$. Of course, the larger the sample, the higher the number of degrees of freedom in the Student's T distributions, hence the *smaller* the percentiles $t_{\alpha/2,n-1}$ will be. Note that, when $n\geq 30$, the Student's T distribution and the Normal overlap, therefore this method ends up being the same as the other.

Note that the CI depends on n through:

- Value $t_{\alpha/2,n-1}$: the higher n, the smaller the value (hence the smaller the CI). However, it is $\lim_{n\to\infty} t_{\alpha/2,n-1}=z_{\alpha/2}$, so there is a lower bound.
- $\sqrt{n}$ at the denominator, so the higher n, the smaller the CI.
- Possibly $\bar{X},S$, which can be however expected to be **stable** with n.

This means that the CI is expected to get **smaller** when we increase the sample width.

Exercise

Suppose that a signal having value μ is transmitted from location A. The value received at location B is subject to a Normal noise with null mean and an unknown variance σ^2. To reduce the decoding error, suppose the same value is sent 9 times. The successive values received are 5, 8.5, 12, 15, 7, 9, 7.5, 6.5, 10.5.

a) construct a 95% CI for μ.
b) Repeat the above, assuming that the variance is known and equal to $\sigma^2=4$
c) Is it possible to state in advance whether one of the two confidence intervals will be smaller than the other?

Solution

The sample is **small**. However, it consists of Normal RVs (by definition). Therefore, we can apply the above method.

a) In order to compute a confidence interval, one has to compute the sample mean and sample variance. These are:

$$\bar{X}=\frac{5+8.5+12+15+7+9+7.5+6.5+10.5}{9}=9$$

$$S^2 = \frac{\sum_{i=1}^{n} {x_i}^2 - n \cdot \bar{X}^2}{n-1} = 9.5$$

Then, we have to compute the values such that:

$$P\left\{-t_{\alpha/2,n-1} \le \frac{\overline{X}-\mu}{S/\sqrt{n}} \le t_{\alpha/2,n-1}\right\} = (1-\alpha)$$

$$P\left\{\overline{X} - \frac{S}{\sqrt{n}} \cdot t_{\alpha/2,n-1} \le \mu \le \overline{X} + \frac{S}{\sqrt{n}} \cdot t_{\alpha/2,n-1}\right\} = 1-\alpha$$

With $\alpha = 0.05$. This means that $S = 3.082$, $t_{0.025,8} = 2.306$, and the interval is $(6.63; 11.37)$.

b) Assume now that we know the variance to be σ^2. Then computations are **different**. More specifically,

$$G = \frac{\overline{X}-\mu}{\sigma/\sqrt{n}} \sim N(0,1)$$

since the sample mean is a Normal RV, and no other RVs are involved. Therefore, we can evaluate a confidence interval using the **standard Normal** in place of a Student's T with $n-1$ degrees of freedom:

$$P\left\{-z_{\alpha/2} \le \frac{\overline{X}-\mu}{\sigma/\sqrt{n}} \le +z_{\alpha/2}\right\} = 1-\alpha$$

Hence:

$$P\left\{\overline{X} - \frac{\sigma}{\sqrt{n}} \cdot z_{\alpha/2} \le \mu \le \overline{X} + \frac{\sigma}{\sqrt{n}} \cdot z_{\alpha/2}\right\} = 1-\alpha$$

With the above values, we have $z_{\alpha/2} = 1.96$ and the interval is $(7.69; 10.31)$, which is smaller than the former.

c) Note that $z_{\alpha/2} < t_{\alpha/2,n-1}$, (in fact, it is $z_{\alpha/2} = \lim_{n\to\infty} t_{\alpha/2,n-1}$, the limit value being approached from above), hence the confidence intervals might, in principle, be expected to be smaller when the variance is known. However, **we cannot be sure whether** $S \overset{<}{\underset{>}{=}} \sigma$, and this may make one confidence interval smaller than the other. In this case, we have $\sigma < S$, but it needs not be so. You may still have a large population variance and pick up a sample consisting of values which are quite close to the mean, so as to have $S < \sigma$.

♦

We need to **reflect a bit** on what confidence intervals really are. A confidence interval is a **random measure**, whereas the estimated parameter is a(n unknown) **constant**. Therefore, computing the 95% CI for the population mean means the following:

If we take 100 different samples, and compute the CI for the population mean, then 95 times out of 100 the population mean will be **within** that CI, and 5 times it will be **outside** that CI.

2.4.2 Confidence intervals and paired experiments

A common use of CIs is to test if a **measure is significantly different from zero**. This happens, for instance, with **paired experiments**.

Paired experiments are when two systems A and B are given in input the **same workloads** and measured for the same statistics (e.g., a response time). This is a frequent case.

In this case, one gets two **paired samples,** namely the n responses of A and those of B. However, since there is a 1:1 correspondence between each point in the sample, one can compute the **difference of the responses** for each experiment.

If that difference is **significantly different from 0**, i.e., if the confidence interval for its mean does not include 0, then we can be sure that one system is better than the other. Otherwise we cannot.

Example

Consider a paired experiment where systems A and B are given the same six workloads as an input. Their response times are measured, and the measures are as in the table. Explain if A is better than B with a confidence of 90%.

	A	***B***	**difference**
1	5.4	19.1	-13.7
2	16.6	3.5	13.1
3	0.6	3.4	-2.8
4	1.4	2.5	-1.1
5	0.6	3.6	-3
6	7.3	1.7	5.6
Avg.	5.32	5.63	**-0.32**
Var.	38.22	44.06	81.62
Std	6.18	6.64	9.03

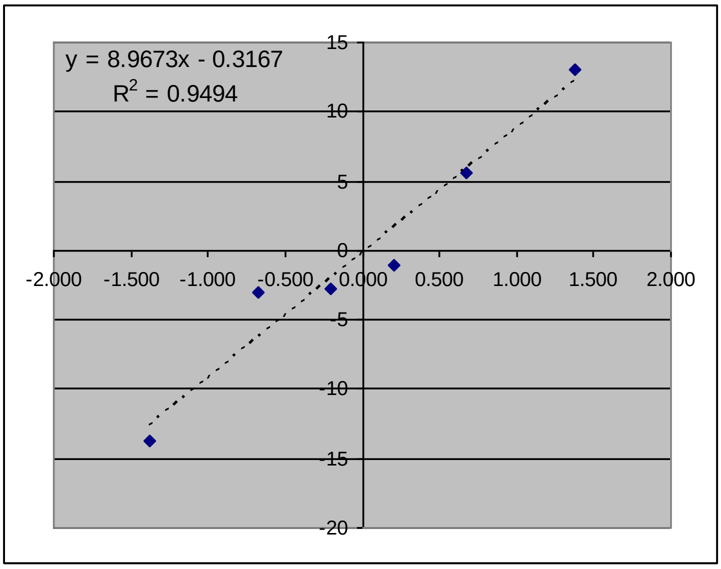
Before we even start: is it possible at all that you get so different response times from two systems? Yes, if your experimental errors are high. For instance, there may be interrupts, disk swaps, etc., which may influence the response time considerably.

Can we be sure that the samples include IID observations? It depends on how we made the experiments. If we have taken care that the result of an experiment cannot influence the next outcome (e.g., because we have reset the system to its initial state after each experiment), then the assumption of IID-ness is reasonable.

The mean difference appears to be below zero, which may indicate that A is faster than B. However, the **variability is high**, and the **sample is small**, which means that it is unlikely that we can be too sure. In order to assess things formally, we need to compute a confidence interval for the mean of the difference.

The sample is small, so we can only do this if the **difference is approximately Normal**. This should be checked visually with a QQ plot.

id	Quantile number	observed	N(0,1)
1	0.0833	-13.7	-1.383
2	0.2500	-3	-0.672
3	0.4167	-2.8	-0.210
4	0.5833	-1.1	0.210
5	0.7500	5.60	0.672
6	0.9167	13.1	1.383



The QQ plot confirms that the sample is approximately Normal, hence we can use our method. The 90% confidence interval is

$$\left[\overline{X} - \frac{S}{\sqrt{n}} \cdot t_{\alpha/2,n-1}, \overline{X} + \frac{S}{\sqrt{n}} \cdot t_{\alpha/2,n-1}\right] =$$

$$\left[-0.32 - \frac{9.03}{\sqrt{6}} \cdot t_{0.05,5}, -0.32 + \frac{9.03}{\sqrt{6}} \cdot t_{0.05,5}\right] =$$

$$[-7.75, +7.11]$$

Therefore, we cannot say with 90% confidence that A is better than B.

♦

2.4.3 Confidence interval for the success probability

Assume that you observe a sample of n observations, call them a_i, and you are interested only in defining the "probability of success" p. For instance, you may measure response times, and declare a success whenever the response time is below a given threshold.

Therefore, you can define a set of *binary outcomes* b_i, such that:

$$b_i = \begin{cases} 1 & a_i \text{ meets criterion} \\ 0 & \text{otherwise} \end{cases}$$

You can thus estimate the **probability of success** of the Bernoullian experiment given by observations b_i. The best estimate of the probability of success of a Bernoulli variable is the sample mean of the observations, i.e.:

$$\hat{p} = \frac{\sum_{i=1}^{n} b_i}{n} = \frac{\sum_{i=1}^{n} 1_{\{a_i \text{ meets criterion}\}}}{n}$$

We want to compute a CI for the success probability. The following method yields approximately correct results if $\min\{n \cdot \hat{p}, n \cdot (1-\hat{p})\} \geq 6$, and the larger the gap the better it works.

Under the above assumption (which is structurally similar to $n \cdot \hat{p} \cdot (1-\hat{p}) \geq 10$) we obtain that the distribution of the number of successes in n trials X (which is a Binomial RV) is **approximately Normal**, with mean $n \cdot \hat{p}$ and variance $n \cdot \hat{p} \cdot (1-\hat{p})$, i.e., $\frac{X - n\cdot\hat{p}}{\sqrt{n\cdot\hat{p}\cdot(1-\hat{p})}} \sim N(0,1)$.

Therefore,

$$P\left\{-z_{\alpha/2} < \frac{X - n \cdot \hat{p}}{\sqrt{n \cdot \hat{p} \cdot (1-\hat{p})}} < z_{\alpha/2}\right\} = (1-\alpha)$$

Where $z_{\alpha/2}$ is the $\alpha/2$ percentile of the standard Normal. Or, equivalently,

$$P\left\{-z_{\alpha/2} \cdot \sqrt{n \cdot \hat{p} \cdot (1-\hat{p})} < n \cdot \hat{p} - X < z_{\alpha/2} \cdot \sqrt{n \cdot \hat{p} \cdot (1-\hat{p})}\right\} = (1-\alpha)$$

Which in turn yields:

$$P\left\{-z_{\alpha/2} \cdot \sqrt{\frac{\hat{p} \cdot (1-\hat{p})}{n}} < \hat{p} - \frac{X}{n} < z_{\alpha/2} \cdot \sqrt{\frac{\hat{p} \cdot (1-\hat{p})}{n}}\right\} = (1-\alpha)$$

What we want to estimate is the probability of success $p = X/n$. Therefore, it is obvious that its CI with level $(1-\alpha)$ is:

$$\left[\hat{p} - z_{\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \hat{p} + z_{\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right]$$

Note that our initial hypothesis, i.e. $\min\{n \cdot \hat{p}, n \cdot (1-\hat{p})\} \geq 6$ requires that you get a **very large sample** if you want to compute CIs for a **very large/very small** success probability. This makes perfect sense, of course.

Exercise

A single experiment was repeated 40 times in independent conditions on two systems A and B. A came out to be superior to B 26 times out of 40. Can we say that A is superior to B (i.e., it outperforms B more than half the times):

- with 99% confidence?

- with 90% confidence?

Solution

We estimate $\hat{p} = 26/40 = 0.65$. This is larger than 0.5, which is the threshold above which we would like A to be, so we start with a good foot. Moreover, $\min\{n \cdot \hat{p}, n \cdot (1 - \hat{p})\} = 14$, so we are on the safe side (note that $n \cdot p \cdot (1 - \hat{p}) \sim 9.1$, which is less than we usually require).

In order to compute the confidence intervals, we compute $\sqrt{\hat{p} \cdot (1 - \hat{p})/n} = 0.075$. Moreover we have $z_{0.005} = 2.576$, and $z_{0.05} = 1.645$. Therefore, the required CIs are:

- at 99% confidence level: $[0.46,0.84]$.

- at 90% confidence level: $[0.53,0.77]$.

Therefore, we can say that we are 90% sure that A outperforms B, but not 99% sure. Our confidence lies somewhere between 90% and 99%, and can be computed with a bit of algebra if needed.

We can immediately observe that, had the sample been of $n = 400$ observations, with $k = 260$ successes (instead of 40 and 26), we would have had the **same estimation** of the success probability, but a **much narrower CI**. In fact, the CI width would have been $1/\sqrt{10} \approx 3.16$ times narrower, which means that we would be 99% confident that A is superior to B. The CI width depends on the **sample width**: if you want high confidence and/or narrow intervals, you need a **large sample**.

♦

2.4.4 Confidence levels and sample width

The above example raises a fundamental question. What **confidence level should we use?** Is 90% enough? The answer is not hard and fast. When you say that "A is better than B with 90% confidence", you are saying that you are 90% sure of what you are saying, and that there is a 10% probability that you are wrong. The same goes for any other level of confidence.

Most of the times, the **confidence level is imposed by the management**. In other words, before investing millions in a new line of hardware, managers may want to be 99.5% sure that this hardware is more performing than the competition. 90% will not be enough for them if the risk of making the wrong decision leads to losing a lot of money.

You have to balance the **gain** (if you are right) against the **loss** (if you are wrong), and then find a confidence level that you can live with.

Obviously enough, the confidence interval is connected to the confidence level **and** the sample width. If you want a **small** CI with a high level of confidence, you have to take a **large** sample. Given an **estimate** of the sample mean and variance, you can compute the **sample width** that is likely to give you a CI of a pre-specified width.

Assume that you want to estimate the population mean, and you want your $(1-\alpha)$ CI to be $[(1-r)\cdot\bar{X},(1+r)\cdot\bar{X}]$, r being the **relative accuracy** (e.g., 10%), and $\bar{X}$ being the **current estimate** of the sample mean. How large a sample do you need? Assuming $n>30$ (so that we can use the Normal approximation), we equate the upper extreme of the CI to $(1+r)\cdot\bar{X}$ (this works if $\bar{X}$ is positive. If it is negative, the upper extreme of the CI should be equated to $(1-r)\cdot\bar{X}$) :

$$\bar{X}+z_{\alpha/2}\cdot\frac{S}{\sqrt{n}}=(1+r)\cdot\bar{X}$$

From the above, via some straightforward computations, you get:

$$n=\left(\frac{z_{\alpha/2}\cdot S}{r\cdot\bar{X}}\right)^2$$

Note that this sample **might not be sufficient**, since, when you add observations, $\bar{X}$ and S are subject to change. Therefore, you may end up with an insufficient sample (or too large a sample). However, the result is representative if the initial estimates are good.

Moreover, note that the required sample width grows with:

- the required confidence level
- the variability of the data, as summarized by S
- the required precision, as represented by r, all of which are pretty obvious
- the **inverse of the sample mean**, which is not so obvious. If $\bar{X}$ is close to zero, in fact, asking for a given relative precision may be too constraining, and may require a **huge sample**.

For instance, if $\bar{X}=0.01$, we will need 100 times as many points as for $\bar{X}=0.1$, all else being equal (i.e., for the same variance, precision and confidence level). Therefore, do not use relative precision if the absolute value $\bar{X}$ is too close to zero.

Exercise

Two packet forwarding algorithms 1 and 2 are believed to exhibit 0.5% and 0.6% loss rate. How many packets do we need to observe to state that 1 is better than 2 with 95% confidence?

Solution

If we want to uphold the above claim, the CIs for the success probabilities of A and B should not overlap. Therefore, we must ensure that the **upper limit of 1's CI is below the lower limit of 2's CI.** 0.5% and 0.6% are "success probabilities", hence we can use the related formulas.

$$\widehat{p_1} + z_{\alpha/2}\sqrt{\frac{\widehat{p_1}(1-\widehat{p_1})}{n}} \leq \widehat{p_2} - z_{\alpha/2}\sqrt{\frac{\widehat{p_2}(1-\widehat{p_2})}{n}}$$

$$\sqrt{\frac{\widehat{p_2}(1-\widehat{p_2})}{n}} + \sqrt{\frac{\widehat{p_1}(1-\widehat{p_1})}{n}} \leq \frac{\widehat{p_2}-\widehat{p_1}}{z_{\alpha/2}}$$

$$\frac{\sqrt{\widehat{p_2}(1-\widehat{p_2})} + \sqrt{\widehat{p_1}(1-\widehat{p_1})}}{\sqrt{n}} \leq \frac{\widehat{p_2}-\widehat{p_1}}{z_{\alpha/2}}$$

$$n \geq \left[z_{\alpha/2} \cdot \frac{\sqrt{\widehat{p_2}(1-\widehat{p_2})} + \sqrt{\widehat{p_1}(1-\widehat{p_1})}}{\widehat{p_2}-\widehat{p_1}}\right]^2$$

By putting numbers in the above computations, you get $n \geq 84340$. Note that, in this case, hypothesis $\min\{n \cdot \hat{p}, n \cdot (1-\hat{p})\} \geq 6$ is largely verified for both systems. Therefore, we need to observe around 85k packets in order to be 95% sure of the result.

♦

2.5 Testing the assumption of independence

Assume that you have a sample of n IID RVs, and that you compute the sample mean, variance and CI for the mean.

Now answer the following question: suppose that you are measuring a system where (you know that) **even** measures are always equal to the preceding **odd** measure. In other words, you always get the same value **twice** when you measure something.

Now, suppose that another system always gives a burst of **ten consecutive equal values** when measured.

Would it be correct to measure a sample of n observations and compute a CI? Recall that the CI formula is

$$\left[\overline{X} - \frac{S}{\sqrt{n}} \cdot z_{\alpha/2}, \overline{X} + \frac{S}{\sqrt{n}} \cdot z_{\alpha/2}\right]$$

Of course not. This is because you cannot **inflate** your sample by adding values that **do not bring any additional information**. A correct way to compute the CI would be to take every second (or tenth) observation, thus using $n/2$ or $n/10$ in the above formula. If we forget to do this, we are using an **inappropriately large value for n, hence we are overestimating our confidence** (or underestimating our CI, which amounts to the same thing).

This is a (paradoxical) example of what happens when the sample does not consist of IID RVs. The assumption of IID-ness implies that the sample contains the **maximum possible information** that can be stored in n observations. If observations are **not independent**, then the amount of information at our disposal is **smaller**, hence our confidence should decrease accordingly.

Everything that we have said so far requires that a sample **consists of IID RV**. This assumption should be **verified** before even starting to make computations.

If you have designed your experiments, then it is **up to you** to ensure that the results are IID. For instance, there are methods to do this **if you are doing simulations**, which we will see later on in the course.

If you are taking **measurements** on a running system, then the **bad news** is that it is **quite unlikely** that your observations will be IID. More specifically, **fine-grained** measures e.g.,

- the delay of consecutive packets at a switch
- the response time of two subsequent transactions in a web server

Are almost invariably **positively correlated**, hence they are **not independent**.

This can be easily explained: if a packet encounters a large delay, it is because it finds many packets queued ahead of it. It is **highly likely** that, when the **next packet** arrives, the situation will not be too different (unless it arrives considerably later in time), since in the meantime the switch will not have been able to transmit many packets. Hence the two consecutive delays will be **positively correlated**: low calls for low, high calls for high.

First of all, **how do you test** the assumption of independence?

You should test that RVs X_i, X_{i+j} have *zero correlation*, for each j. A quick, visual test is the **lag plot**. You plot values $\left(X_i, X_{i+j}\right)$ on a Cartesian plane, and you may be able to observe positive/negative correlations at some lag j (e.g., $j = 1$). Data that are **positively correlated** tend to align on a line going SW-NE. Data that are negatively correlated align NW-SE instead.

If the lag plot does not show any particular trend, then it is likely that data are independent.

Note that, if you have a **large sample**, then you are likely to see a **cloud**, and be unable to make heads or tails of it. In this case, just plot a **subset** of your sample (e.g., the first 100 consecutive points), and see what happens.

A more rigorous way is to compute **the autocorrelation function** for various lags. The population autocorrelation function is defined as:

$$R(j) = \frac{E\left[(X_i - \mu)\left(X_{i+j} - \mu\right)\right]}{\sigma^2}$$

Autocorrelation is between -1 (perfect anticorrelation) and $+1$ (perfect correlation). Ideally, it should be $R(j) = 0$ for any $j > 0$ in order for our observations to be IID. However, we do not know either μ or σ, so we cannot compute it. We can estimate it by computing the **sample autocorrelation function**, which can be computed automatically by most packages (including MS Excel). This is equal to:

$$\overline{R}(j) = \frac{\frac{1}{n} \cdot \sum_{i=1}^{n-j}\left(X_i - \overline{X}\right) \cdot \left(X_{i+j} - \overline{X}\right)}{S^2}$$

The latter should be plotted for various lags (e.g., up until 100 or up until meaningful). If $|R(j)|$ is **small enough**, then the assumption that observations are IID can be upheld.

How small is small enough? The $(1-\alpha)$ CI for the sample autocorrelation coefficients is $\pm z_{\alpha/2}/\sqrt{n}$. We are claiming that **all the autocorrelation coefficients are null**, hence we should compute CIs for the autocorrelation coefficients, and see if they include zero. In practice, this is equal to computing two straight lines at $\pm z_{\alpha/2}/\sqrt{n}$, and see if all the values $\overline{R}(j)$ are in this interval. If they are, we can say that our sample consists of IID RVs **with a significance level equal to** $(1-\alpha)$.

In other words, we cannot **reject the null hypothesis** that our data are IID with that significance if the autocorrelation function is within a $\pm z_{\alpha/2}/\sqrt{n}$ boundary.

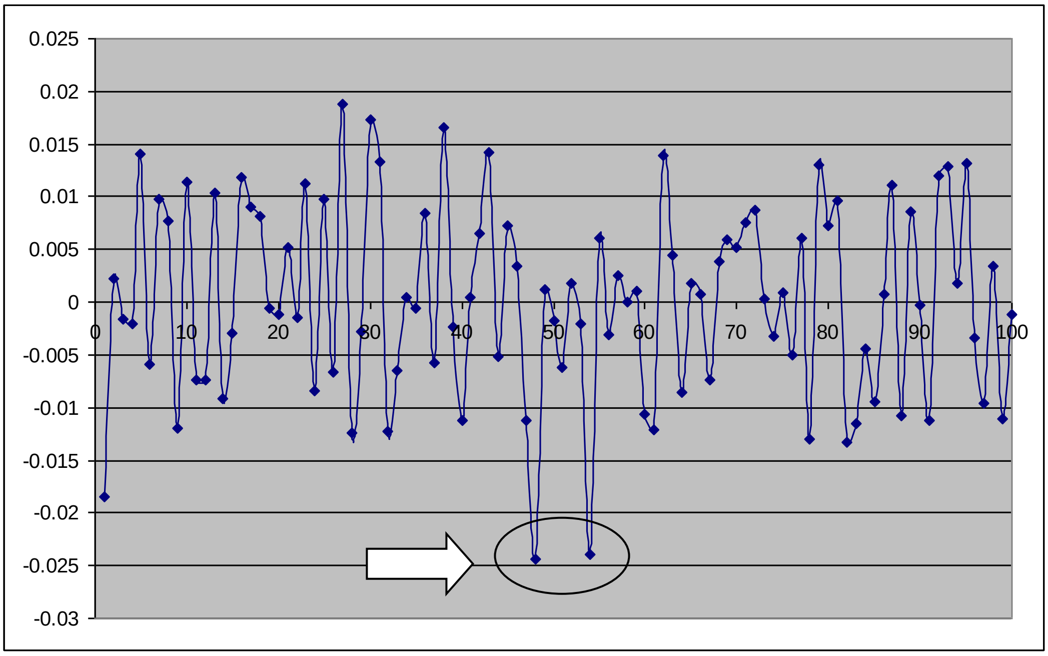
As anticipated, $\overline{R}(j)$ can be easily computed through Excel, using built-in function CORRELAZIONE(). That function takes two arguments, which are the two vectors of the data to be correlated. In this case, the two vectors are the initial vector and the same vector with an offset j.

Example

The following is an autocorrelation plot (**correlogram**) for a multiplicative LCG, using a sample of 10k observations. For a 95% confidence, we get $z_{0.025} = 1.96$, and $\sqrt{n} = 100$. Therefore, the assumption of IID-ness cannot be rejected with the above significance level if all sample autocorrelation coefficients are within ± 0.0196.

This is false, hence we must **take a closer look**. In order to read correlograms, you should keep this in mind:

a) it is perfectly normal that a fraction α of the points are outside the two bands (this is intrinsic in the fact that we are asking for $(1-\alpha)$ confidence). So, if you see **few** values outside the band, keep your cool and check the next bullet.
b) **How large** are these outliers? If their absolute value is **close to the limit**, then you should not worry too much. If they are **abnormally high**, then you should worry. So, the higher (in absolute value), the worse.
c) **Where** are these outliers? If they are at **large lags**, then you should not worry. At large lags, $\overline{R}(j)$ is less reliable due to the fact that the sum by which it is computed has **too few** values, hence it is subject to **higher oscillations**: it may just be bad luck. If they are at **small lags**, then you should really worry. Note that, in order to keep this phenomenon into account, some analysts use **confidence bands of increasing width**, just to signify that higher correlations at large lags should be tolerated.



What should we do when the sample is not IID?

A simple technique is **subsampling**. Construct a **smaller** sample, including every point of the previous sample **at random with probability p.** Try different values of p (typically, $1/2^k$ for increasing values of k), and **repeat the analysis** (lag plots, correlograms).

Unless your data is very wild (long-range dependent), this method should work after a while (i.e., for a suitably large value of k).

Of course, by doing this, you are **reducing the width of your sample**. If you start from n points, you end up with $n/2^k$ points on average. Therefore, it is important to stop as soon as your data shows negligible autocorrelation. This way, you quickly find the **real amount of information** that your sample includes, and you can compute meaningful confidence intervals.

Subsampling can be done automatically with Excel: you need to use item CAMPIONAMENTO from the additional component ANALISI DEI DATI. OO Calc has a similar feature.

Exercise (to be done with Excel/Calc)

Take a large sample of data (e.g., successive delay at some network system). Compute the mean and confidence interval assuming that they are IID, and acknowledge that the CI is small. Check the IID hypothesis using visual lag plot and autocorrelation function.

Since data are **not IID**, then subsample them with $p = 1/2^k$ probability and check the IID-ness hypothesis again. Stop when they are IID, and compute again the mean and CI. Check that it is considerably larger than before.

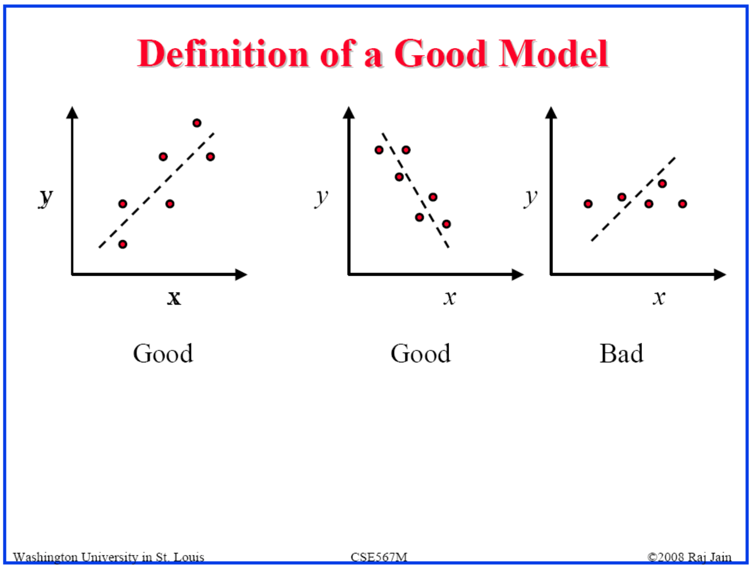
♦

3 Model Fitting

This technique is also called **regression**. It consists in finding a **parametric model** that explains well the correspondence between **a predictor variable** and a **predicted response**. This, in turn, allows one to make **predictions on the future**, which is the interesting aspect.

Suppose we have n **observations** $\{(x_i, y_i), 1 \le i \le n\}$, and we want to find a model that allows us to predict values of the RV Y from those of X.

What is a **good model**? It is one that "fits the data well", i.e., is in accordance with the data.



Definition of a Good Model

Washington University in St. Louis CSE567M ©2008 Raj Jain

This is useful, for instance, if you suspect that *x* is a likely cause for *y*, or if measuring *y* is too tricky or expensive, while measuring *x* is relatively easier.

The shapes are all lines (hence the term "linear regression"). We will describe linear regression first, and then show that non-linear regression techniques are easy to derive.

3.1 Linear regression

For a linear model, we need to find **two regression parameters**, namely the **offset and slope** of the line that "best fits the data". Fitting the data to the model consists in computing the "best" b_0, b_1 such that $\hat{y} = b_0 + b_1 \cdot x$. x is the **predictor variable**, and $\hat{y}$ is the **predicted response**. Of course, what values of b_0, b_1 are the best will depend on the **data at my disposal**, i.e., on the sample of n observations $\{(x_i, y_i), 1 \le i \le n\}$.

The difference between the **predicted** response (i.e., value $\widehat{y_i} = b_0 + b_1 \cdot x_i$) and the **actual** response (i.e., measured value y_i in paired observation (x_i, y_i)) is called **error** or **residual.**

$$e_i = y_i - \widehat{y_i} = y_i - b_0 - b_1 \cdot x_i$$

In practice, the **linear regression technique** (called the **least square method**) consists in finding the line that:

- minimizes the **Sum of Squared Errors (SSE), i.e.,** $\sum_{i=1}^{n} {e_i}^2 = \sum_{i=1}^{n} \left(y_i - b_0 - b_1 \cdot x_i\right)^2$
- subject to the condition that the **mean error is null,** i.e., $\sum_{i=1}^{n} e_i = \sum_{i=1}^{n} \left(y_i - b_0 - b_1 \cdot x_i\right) = 0$

This can be done algebraically (check you notes on numerical analysis), and the result is as follows:

$$b_1 = \frac{\sum_{i=1}^{n} x_i y_i - n \cdot \bar{x} \cdot \bar{y}}{\sum_{i=1}^{n} {x_i}^2 - n \cdot (\bar{x})^2},\ b_0 = \bar{y} - b_1 \cdot \bar{x}.$$

The same procedure (i.e., finding a minimum of a constrained optimization problem) can be used to fit any other (non-linear, or multiple linear) model. The problem is always of the following form:

$$\begin{aligned} &\min \sum\nolimits_{i=1}^{n} \left(y_i - f(x_i)\right)^2 \\ &s.t.\ \ \sum\nolimits_{i=1}^{n} \left(y_i - f(x_i)\right) = 0 \end{aligned}$$

Where $f(\quad)$ can be any function. The only problem is that solving it may require heavy computations (non-linear optimization problems can be hard to solve), especially with **large samples** or functions of many parameters.

Note that **MS Excel** can solve constrained optimization problems (through an add-on component called “solver” or “*risolutore*”). Moreover, it fits automatically several types of models, namely:

- linear (the one we have just seen)
- polynomial (up to a degree of six, which requires up to 7 parameters)
- logarithmic
- exponential
- power relationship ($\hat{y} = a \cdot x^b$)

This is done quite easily:

- you do a scatterplot graph of the observations.
- you add **a regression line on the graph** (“linea di tendenza”) and ask for its equation to be displayed.

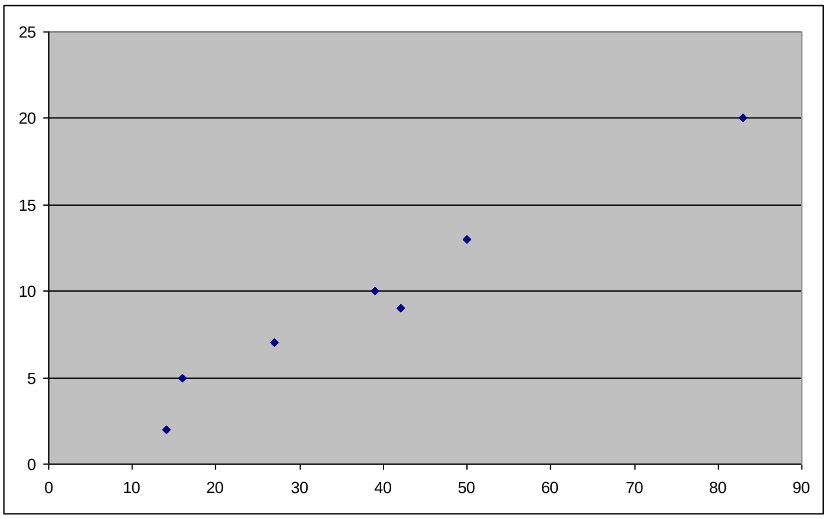
The above equation gives you the regression curve that minimizes the SSE, as well as the **coefficient of determination,** which we will see in a minute.

Example (to be done using Excel)

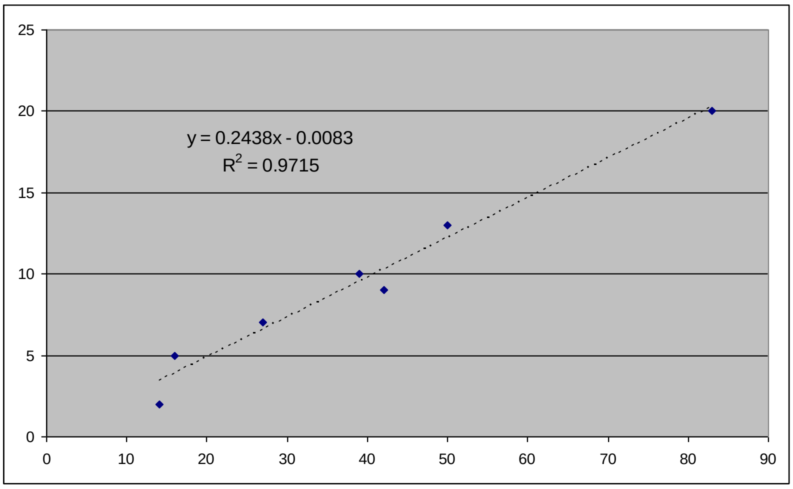
Disk I/O time	CPU time
14	2
16	5
27	7
42	9
39	10
50	13
83	20

The number of Disk I/Os and CPU times of seven runs of a program were measured, and the results in the table were obtained. We want to fit a model which allows one to predict the CPU time from the disk I/O time.

The first thing to do is to draw a **scatterplot**, having disk time on the x axis and the CPU time on the y axis.

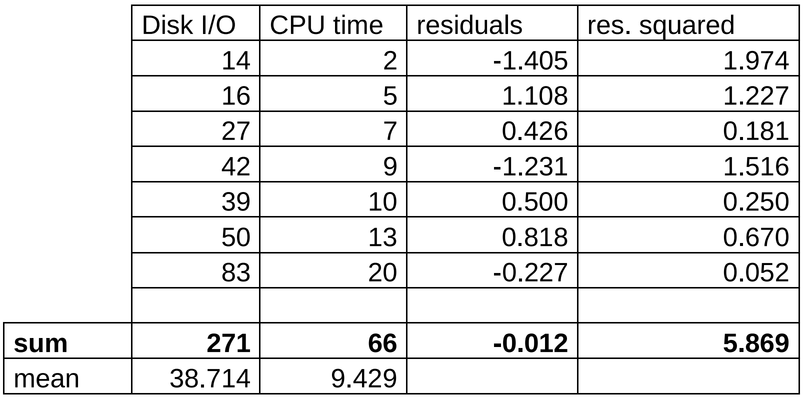


The scatterplot looks reasonably **linear**. Therefore, we can use the least square method and find the best linear prediction to be the following:



We obtain $b_1 = 0.2438$, $b_0 = -0.0083$.

We can compute the residuals and the SSE:



	Disk I/O	CPU time	residuals	res. squared
	14	2	-1.405	1.974
	16	5	1.108	1.227
	27	7	0.426	0.181
	42	9	-1.231	1.516
	39	10	0.500	0.250
	50	13	0.818	0.670
	83	20	-0.227	0.052
sum	**271**	**66**	**-0.012**	**5.869**
mean	38.714	9.429		

Note that the sum of the residuals is not **null**. This is due to numerical approximations (we would need more decimals for the regression coefficients). The SSE is 5.869, and is the minimum possible. Every other line will yield a higher one.

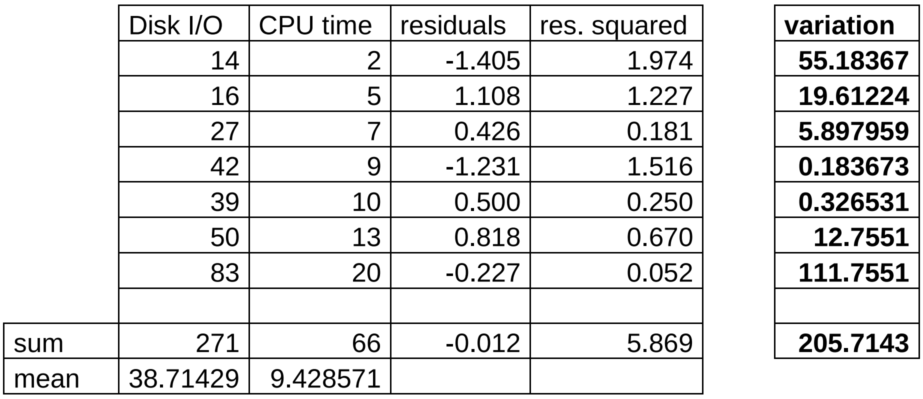
♦

The SSE is an interesting quantity, since it allows you to compute the **part of the variation of the predicted response which is not explained by the model**.

The variation (not the **variance**: the **variation**) of the predicted response is called the **Sum of Squares Total**, defined as $SST = \sum_{i=1}^{n}(y_i - \bar{y})^2$.

The quantity $R^2 = \frac{SST-SSE}{SST}$ is called the **coefficient of determination**, and it explains how much of y's variation can be predicted by the model we have used. It is the square of $corr(X,Y)$. Of course, this quantity is between 0 and 1, where 1 would indicate that there **is no uncertainty** in the ys once we have discovered their dependence from the predictor variable x.

We can compute R^2 manually, or just have Excel do the work for us.



	Disk I/O	CPU time	residuals	res. squared		**variation**
	14	2	-1.405	1.974		**55.18367**
	16	5	1.108	1.227		**19.61224**
	27	7	0.426	0.181		**5.897959**
	42	9	-1.231	1.516		**0.183673**
	39	10	0.500	0.250		**0.326531**
	50	13	0.818	0.670		**12.7551**
	83	20	-0.227	0.052		**111.7551**
sum	271	66	-0.012	5.869		**205.7143**
mean	38.71429	9.428571				

In the above example, we get $R^2 = \frac{SST-SSE}{SST} = \frac{205.714-5.869}{205.714} = 0.9715$. This means that more than 97% of y's variability is explained by the model, and the rest is due to factors that the model cannot explain (e.g., experimental errors).

I am interested in showing you **two aspects of model fitting** which are interesting from a performance evaluation standpoint, namely:

- How to **predict** a value using a regression model, and how to associate a **confidence** to that prediction.
- How to **assess** whether a regression model is good or not. In particular, how to verify if the **underlying assumptions** are met. They may not be, and large errors may occur if they are not.

First of all, we observe that the **model parameters b_0, b_1 are random variables** (since they are obtained from a sample whose predicted variable is affected by random experimental errors), and they **estimate** the true model parameters β_0, β_1. We need to draw a confidence interval for the latter.
The procedure to get to the result is complicated. We will just state the result, so that you can use it as a reference (you can also find it on textbooks).

- For β_0, the $(1-\alpha)$ CI is $b_0 \mp t_{\alpha/2,n-2} \cdot \sqrt{\frac{SSE}{n\cdot(n-2)} \cdot \frac{\bar{x}^2}{\sum_{i=1}^{n} x_i{}^2 - n\cdot\bar{x}^2}}$

- For β_1, the $(1-\alpha)$ CI is $b_1 \mp t_{\alpha/2,n-2} \cdot \sqrt{\frac{SSE/(n-2)}{\sum_{i=1}^{n} x_i{}^2 - n\cdot\bar{x}^2}}$

The uncertainty will depend, of course, on the SSE. The higher the fraction of variation that the model **cannot explain**, the larger the CI for its variables will be.
Especially for the slope β_1, you may want to check that **your CI does not include zero**, otherwise you may not even be sure that there is any dependence at all between the predicted response and the predictor variable. If it includes zero, the dependence may be **a coincidence, due to a lucky sample**.
We have said that the purpose of **regression** is to allow one to **predict** the value of the predicted response for a given value of the predictor variable. Obviously, any prediction has **uncertainty**, and uncertainty has to be quantified.
The uncertainty will depend, of course, on the SSE. The $(1-\alpha)$-**prediction interval** for a **future observation** of the predicted variable at value x_p of the predictor variable is the following:

$$\hat{y}_p \mp t_{\alpha/2,n-2} \cdot \sqrt{\frac{SSE}{n-2} \cdot \left[1 + \frac{1}{n} + \frac{\left(x_p - \bar{x}\right)^2}{\sum_{i=1}^{n} x_i{}^2 - n \cdot \bar{x}^2}\right]}$$

Where $\widehat{y_p} = b_0 + b_1 \cdot x_p$. Note that the **prediction interval** depends on the **value of the predictor variable**, and it is minimal when $x_p = \bar{x}$. In fact, the further you move away from the "center" of the interval, the less information you have regarding the prediction. If you go far outside the interval where you have sampled the predictor variable, then you have a large uncertainty (always assuming that your model is **still valid** outside that range, which may not well be).

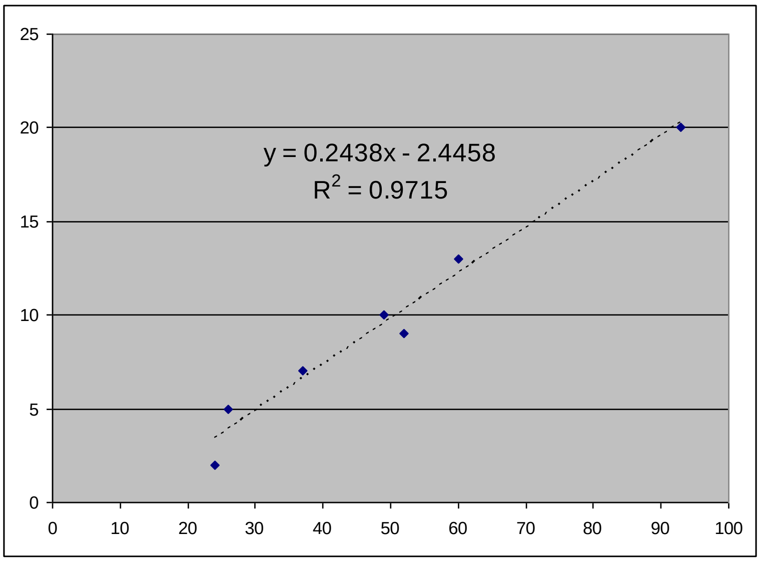
3.1.1 Testing the assumptions

The problem with formal methods – always – is that people **stop using their brain** when they can follow one, and tend to trust the results **blindly**. If they know something about the phenomenon they are modelling, they act like they don't. This should be avoided, of course. Recall that performance evaluation is **an art**.

The first assumption to be tested is if the relationship between the two variables **indeed looks linear**. This can be tested visually by drawing a scatterplot of both quantities.

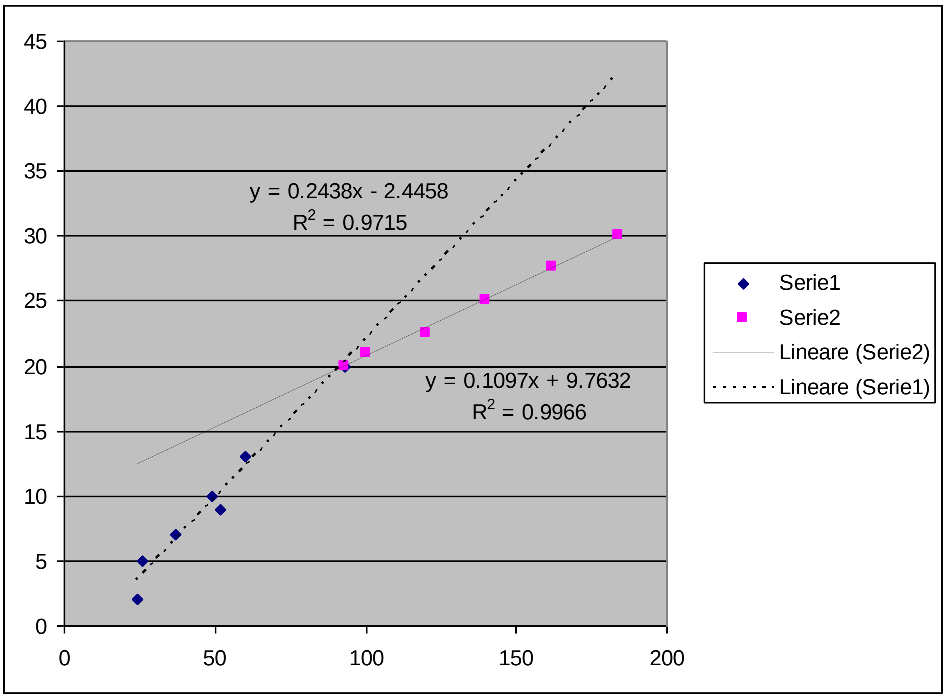
Note that you should **avoid making rash assumptions** regarding the relationship between your variables **outside your measurement interval**. Most phenomena exhibit **threshold behaviour** (because systems tend to reach saturation, and physical quantities are often *positive*), hence this should be taken into account – especially when making predictions.

For instance, the linear regression in the figure below yields the coefficients reported directly in the graph. From these, we might conclude that the best prediction when $x_p = 1$ is $\widehat{y_p} = -2.2$. If those two quantities **must be positive** (e.g., they are completion times for two different activities), it is clear that we are misunderstanding the model. The model looks **linear** in the interval that we are observing, but is in fact **multilinear (**i.e, the predicted variable saturates to zero).

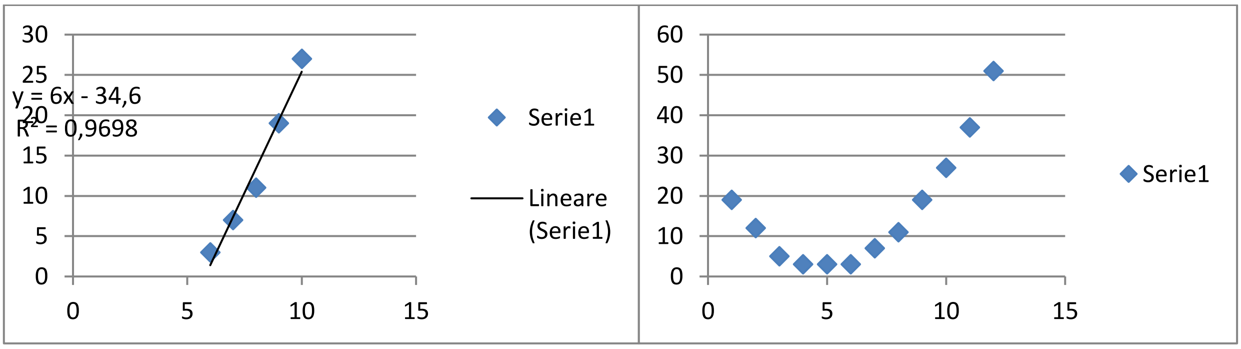


For the same reason, you might be interested to know what happens when $x_p = 120$, and the model states that $\widehat{y_p} = 26.81$, Still assuming that you are measuring a response time, you may know for a fact (e.g., since this is a design constraint) that my system **drops requests** after 20 seconds of computations, hence no response time larger than 20 can be observed. Again, this is an indication of **multilinearity**, and you should be careful to make inferences where none is due.

If a system has a strong **multilinear** tendency, which appears in the observations, then it may be wise to **model-fit** its linear parts separately.



Note that there are other cases when a model **may look linear** when in fact it is not. For instance, the half of a parabola that lies left/right of its vertex can sometimes be approximated through a line, if the range is not too large. If you only look at **more points** to the left and right, you see that the model is nonlinear.



There are some hidden assumptions, which are particularly tricky and **must be validated**. When fitting a model $\hat{y} = f(x)$ using least-square minimization, we are **assuming that**:

residuals are IID Normal variables with a null mean and a constant standard deviation.

The fact that residuals have a null mean is obvious, and it is guaranteed by the fact that we use least-square minimization (where null residual mean is in fact a constraint of the optimization problem). The fact that they are IID, Normal, and with a constant standard deviation is not obvious at all, and it **may well be false**. If it is, then our model is **weak**, meaning that we cannot be confident that it explains the data well, **even when the $\boldsymbol{R^2}$ coefficient is close to 1**.

The fact that residuals are normally distributed can be easily tested with a **Normal QQ** plot, as already seen many times. The **normality assumption** is **only** required for confidence and prediction intervals (in fact, you can use Student's T if and only if variables are Normal). If residuals are not Normal, then confidence and prediction intervals will not be what you think they are, but least-square minimization will **still** give you the best unbiased estimate of the regression coefficients.

The other two assumptions can be tested using simple visual techniques:

- a scatterplot of the residuals vs. the predicted response to test **constant std (*homoskedasticity*)**
- a scatterplot of the residuals vs. the **observation number**, to test **IIDness**.

The two scatterplots must show **no visible trend** of the residuals.

If residuals are ascending/descending with the observation number**,** there may be a **bias** (e.g., some incorrect initialization) that shows that **experiments are not independent** (e.g., you forgot to reset some meter when measuring the output).

If the absolute value of the residuals shows (e.g.) an **increasing trend** with the predicted variable, then it is unlikely that residuals have **constant standard deviation**.

The assumption of **constant standard deviation** is important: in least-square minimization we **weight every point the same**. It is like solving:

$$\min \sum_{i=1}^{n} w_i \cdot \left(y_i - f(x_i)\right)^2$$

$$s.t. \quad \sum_{i=1}^{n} w_i \cdot \left(y_i - f(x_i)\right) = 0$$

With $w_i = 1\ \forall i$. This means that we are assuming that all the points carry the same amount of information. However, if some variables (e.g., at higher values of the predictor) have a higher variance, then they contain **more information** than others, hence they should be weighted **more** than the others.

Something similar happens when residuals are **correlated**. In both cases, the net result is a **loss of precision in the model (hence, in the estimates)**. If needed, **weighted least-square** can be used to get rid of both problems. This is, however, outside the scope of this course.

Exercise (to be done using Excel)

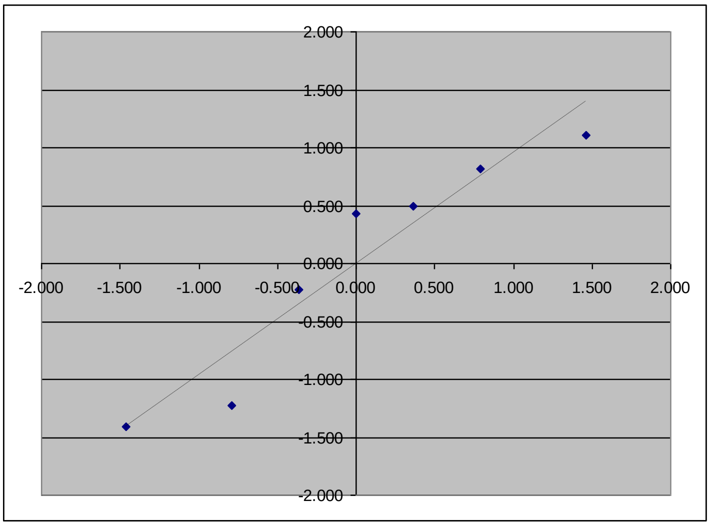
Test the assumptions for the previous example

Solution

1) The model looks linear – we have already ascertained that.

2) As far as normality is concerned, we draw a QQ plot as follows:

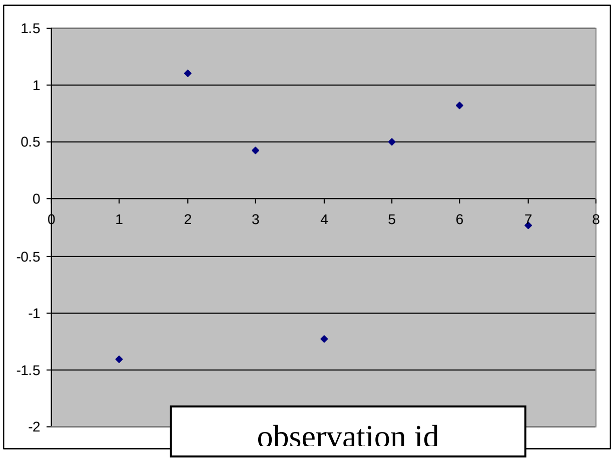
id	quantile	Normal Q	O.S. of residuals
1	0.071	-1.466	-1.405
2	0.214	-0.789	-1.231
3	0.357	-0.365	-0.227
4	0.500	0.000	0.426
5	0.643	0.365	0.500
6	0.786	0.789	0.818
7	0.929	1.466	1.108

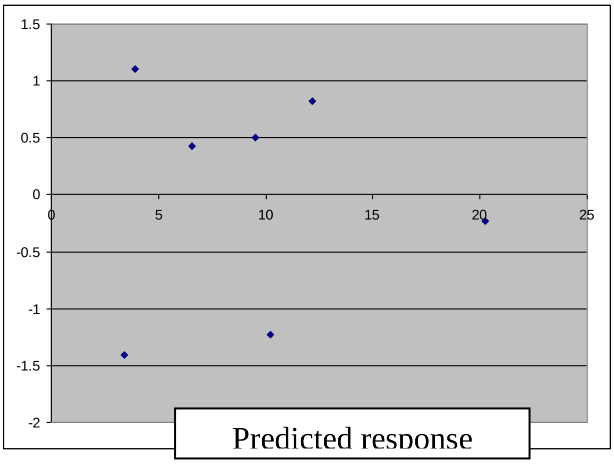


The QQ plot looks approximately linear, so we are ok with normality.

3) let us check whether the std is constant and variables are IID.

The left plot shows the residuals vs. the experiment number, and it yields no visible trend. The right plot shows the residuals vs. the predicted response, and shows no trend as well. Moreover, the order of magnitude of the residual is **small** w.r.t. the one of the predicted response.





We conclude that **our analysis verifies the hypotheses**, hence the model is valid. It is also a good model, since it explains most of the variability.

♦

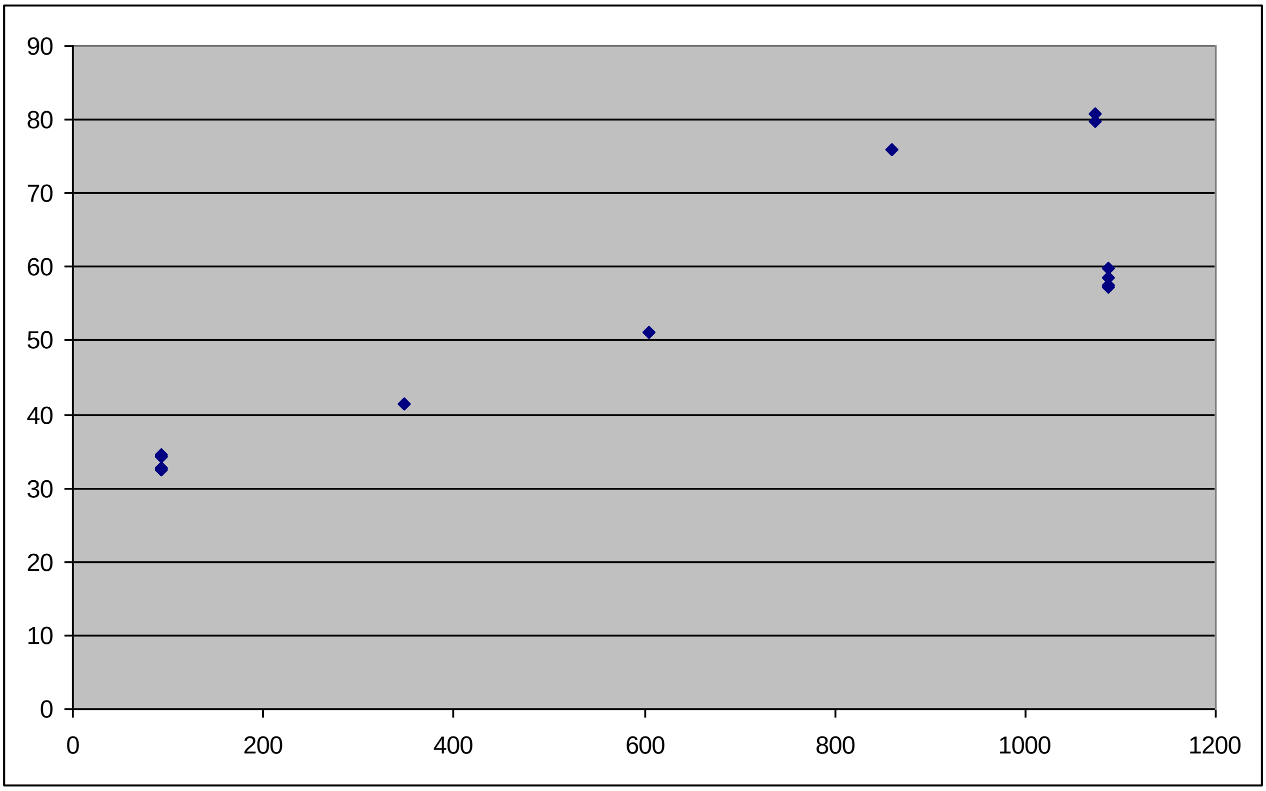
Exercise (to be done with Excel)

Apply linear regression to the set of data below, taken from measurements of a remote procedure call time on a system. Check if the model is correct.

Bytes (predictor)	Time (predicted variable)
92	32.8
92	34.2
92	32.4
92	34.4
348	41.4
604	51.2
860	76
1074	80.8
1074	79.8
1088	58.6
1088	57.6
1088	59.8
1088	57.4

Solution

The first test is the visual one, for linearity:



It appears that the data **can** be approximated through a linear model.

I would observe that there is something **slightly suspicious** at the right end of the graph. Several courses of action are possible, such as:

- **use a multilinear model**. Stop the first linear model at 1074, and start with the other using the last couple of values of the predictor variable. Unfortunately, this does not seem much of an alternative, since a sequence of two values is way too small to make any inference

- **discard the last four measurements** (i.e., those related to the highest value of the predictor), flagging them as outliers, and be content with the first linear model, explaining data up to 1074 bytes. This is not too advisable either, unless you have good cause to do so.

- **measure around the suspicious values**: get measures for values of the predictor variable immediately to the right and left of 1088, so that you gain more information. There may be some bug according to which 1088 creates an unusual behaviour, and everything goes back to normal after that. Otherwise, it may be that at 1074 the system start exhibiting a different behaviour (e.g., a decreasing line, i.e., it starts being multilinear or non-linear in general), and you might want to know that.

We choose none of these alternatives, and use linear regression for the whole set of data.

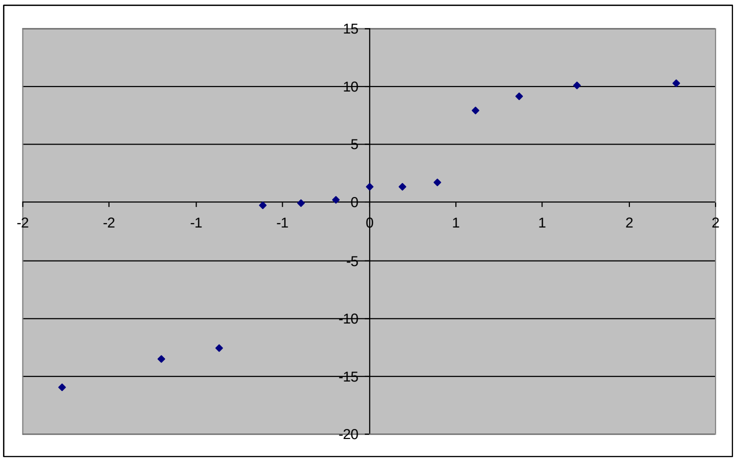
We let Excel do the work, and obtain: $\hat{y} = 0.0337 \cdot x + 31.068$, with $R^2 = 0.7482$, which is not that much. It means that our model cannot explain 25% of the variation of the response.

id	Bytes	Time	pred. Resp.	residuals	SST comp.
1	92	32.8	34.17	1.37	-20.77
2	92	34.2	34.17	-0.03	-19.37
3	92	32.4	34.17	1.77	-21.17
4	92	34.4	34.17	-0.23	-19.17
5	348	41.4	42.80	1.40	-12.17
6	604	51.2	51.42	0.22	-2.37
7	860	76	60.05	-15.95	22.43
8	1074	80.8	67.26	-13.54	27.23
9	1074	79.8	67.26	-12.54	26.23
10	1088	58.6	67.73	9.13	5.03
11	1088	57.6	67.73	10.13	4.03
12	1088	59.8	67.73	7.93	6.23
13	1088	57.4	67.73	10.33	3.83

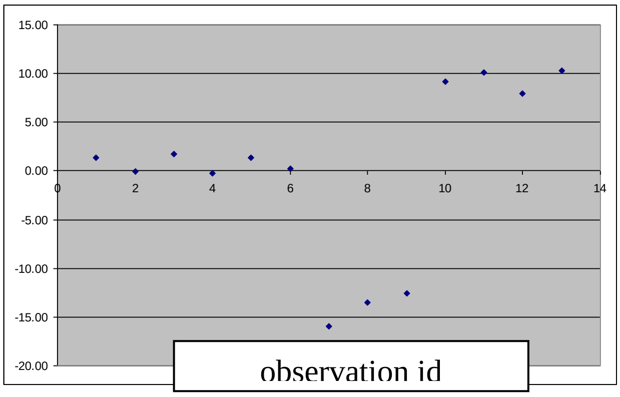
sum	8680	696.4	696.4	0	-2.13E-14
sumSq	8301304	41109.20	40151.35	957.78	3803.59
mean	667.69	53.57	53.57	0	-1.64E-15

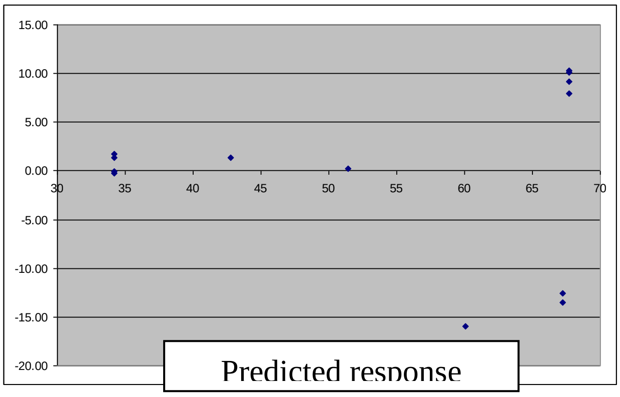
R2	0.748

Now we test the assumptions. First of all, the QQ plot does not seem to be linear at all.



Moreover, the residuals seem to grow in magnitude with the number of experiment (which is bad), and with the magnitude of the predicted variable (which is worse). They do not seem to have **constant standard deviation**.





The result is that this is a **weak model**. It is **false** that it explains 75% of the variation of the predicted response, since the hypotheses are not met. Predictions made with such a model are likely to have much smaller confidence than they should. In other words, confidence intervals will appear smaller than they really are.

The fact that a linear model is not a good one should not be attributed to the fact that the model may look multilinear (i.e., because of the right edge of the graph being unaligned with the rest). If we restrict to the interval [92,1074], then you can check that $R^2 = 0.97$, **but**:

- the normality assumption is still **not verified** (tails are heavier than a Normal)
- the assumption of constant std is not verified either (you can check that the above graph keeps showing an increasing trend with the predicted response even if you remove the four points at the top-right).

♦

Models can also be non-linear (e.g., exponential, etc.). For an exponential model, we need to compute a, α such that $\hat{y} = a \cdot e^{\alpha \cdot x}$. The regression parameters are a, α, and they are computed based on our n observations.
Non linear regression is not different in the hypotheses. The same procedure used to verify **linear regression** should also be applied to **non-linear** regression as well, the only difference being the visual test to determine the shape of the regression model.

We can also use **transformations**. What kind of transformations can you use, and when and why should you use them?

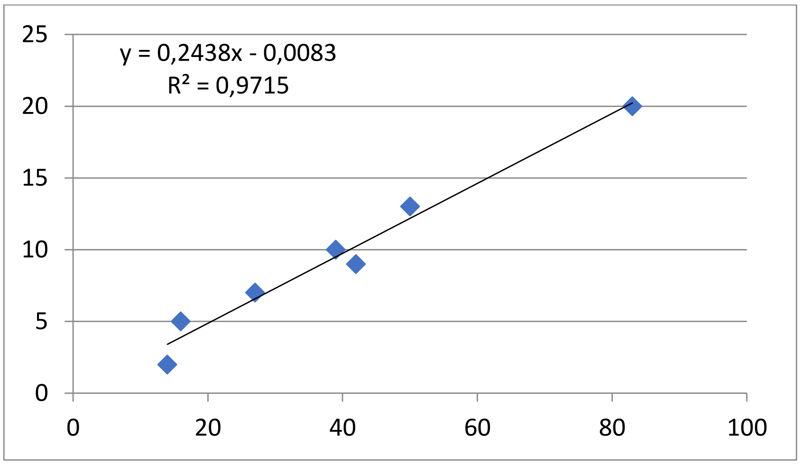
- **when**: there are basically two reasons. The first is to **simplify the analysis**: when you have a complex relationship, it pays to simplify it. There are a lot of transformations that yield a linear model, which is easier to interpret and justify. The other reason is to **make up when the assumptions are not met**. For instance, residuals may not be Normal, the other assumptions may not be met, the ratio $y_{\max}/y_{\min}$ may be too large for a linear model. A **transformation involving the predicted variable changes the distribution of the residuals**. If it wasn't Normal before, it may be Normal **after the transformation**.
- What transformations can you use? **Those that yield a linear model**, because the latter is easy to analyze. For instance:

Model	Transformation	New Model
$y = a + \frac{b}{x}$	$x' = 1/x$	$y = a + b \cdot x'$
$y = \frac{1}{a + b \cdot x}$	$y' = 1/y$	$y' = a + b \cdot x$
$y = a \cdot b^x$	$y' = \log y$	$y' = \log a + (\log b) \cdot x$
$y = a + b \cdot x^n$	$x' = x^n$	$y = a + b \cdot x'$
$y = a \cdot x^b$	$y' = \log y, \quad x' = \log x$	$y' = \log a + b \cdot x'$

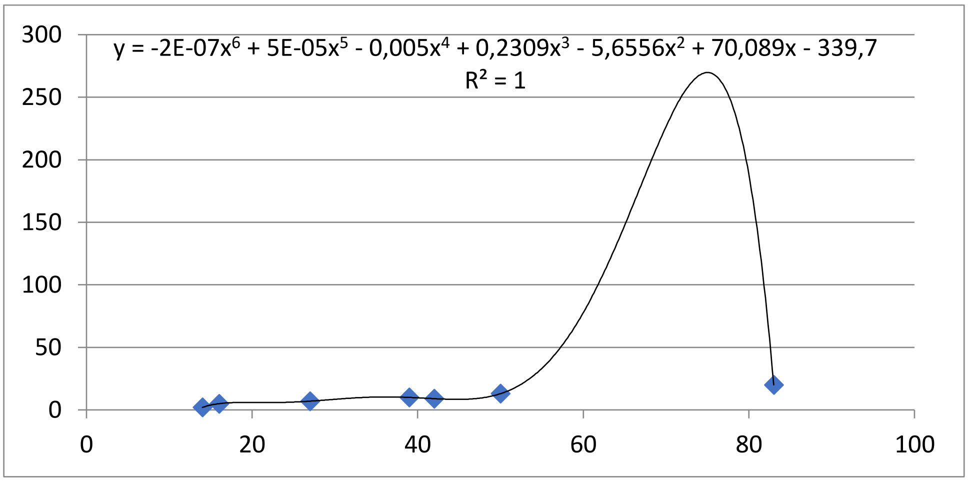
3.2 Overfitting

Overfitting occurs when you try to find a model that fits *exactly* your data, without using knowledge of your system. Take once more the example of disk vs. CPU time, which we have fitted to a linear model with good results:

Disk I/O	CPU time
14	2
16	5
27	7
42	9
39	10
50	13
83	20

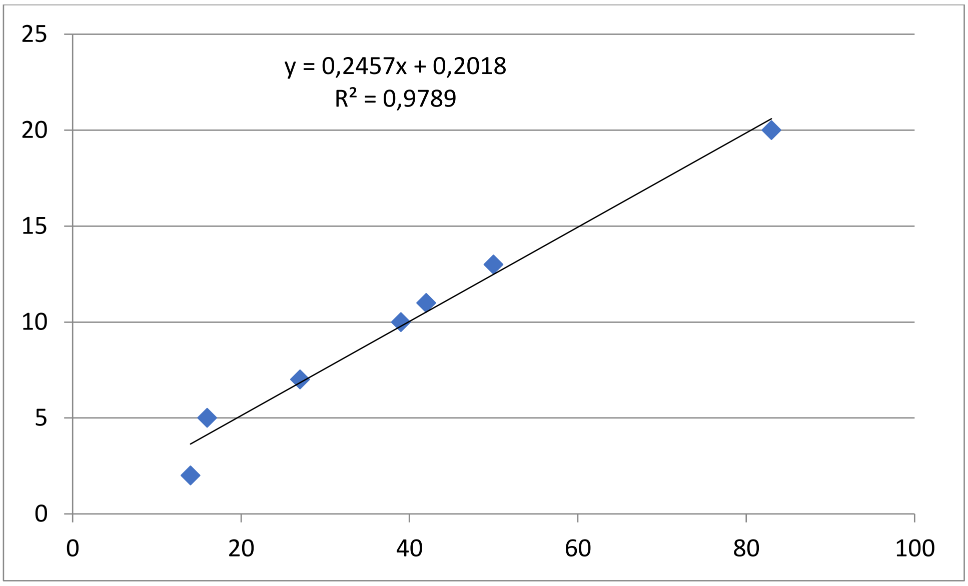


If you want to obtain *a perfect fit* for your set of 7 paired observations, why not using a **high-degree polynomial model?** The result is the following:

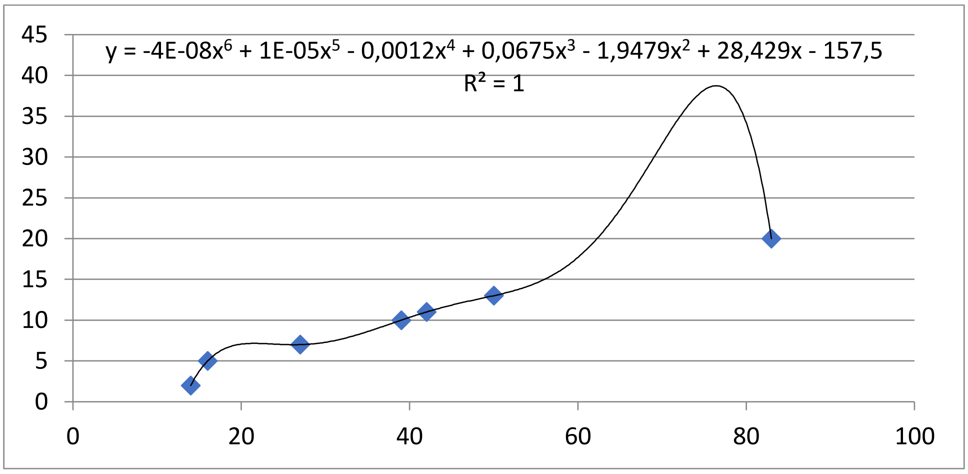


The fit is perfect, since $R^2 = 1$. However, this is **plainly stupid**. First of all, there is no reason whatsoever why the relationship between the predictor and the predicted variable should be polynomial. There is absolutely no reason to envisage a peak in the predicted response around $x = 75$. Second, for each (nondegenere) set of n points there exists a degree-n polynomial that passes through all the points, so it's a foregone conclusion in any case. Third, we are **underestimating the fact that there are errors in the measurements**. Let us see what happens if the measurements are only slightly different, e.g. by adding some error to point (42,9), which becomes (42,11).

Disk I/O	CPU time
14	2
16	5
27	7
42	11
39	10
50	13
83	20



Our linear model stays pretty much the same. The slope changes just a bit. What about the polynomial one?



It is clear that the model is **entirely different** (look at the scale of the y axis). Try predicting the value of the predicted response when $x = 60$ in the two cases, and you will get wildly different estimates (75 vs. 18). Not so with the linear model, where the prediction differs by few percentage points.

This is a clear case of **overfitting**. Overfitting occurs when a model can **only** explain the very data it has been built on, and nothing else. While the linear model has **good predictive capabilities**, the polynomial one hasn't any: it describes **random error or noise instead of the underlying relationship**. Overfitting occurs when a model is **excessively complex**, e.g. includes too many parameters relative to the number of observations. As such, it overreacts to minor variations.

Always go for the **simplest model,** and and give precedence to the **insight on your problem**.

4 Design of experiments

Quite often, you have to analyze one or more systems whose performance depends on several **factors**. The impact of the factors is not known in advance – more to the point, it is exactly what you need to assess.

For instance, assume that you want to evaluate the performance of a **web server**, given a workload (i.e., a trace of transactions, or an artificially generated input process), as a function of:

- the type of CPU (two alternatives)
- the amount of RAM (interval)
- the cache size (interval)
- the number of disks (interval)
- the software architecture (two alternatives: Apache TomCat and OpenLiteSpeed)

The above are called **factors**. The first and last one are **binary**, the others are **discrete (multilevel).**

You may need to answer the following questions:

1) which factors **are the most relevant** to my metric of interest (say, throughput or response time)?
2) How can you **tune these factors** to maximize the performance?

In order to analyze your system, you have essentially three alternatives.

Simple factor analysis

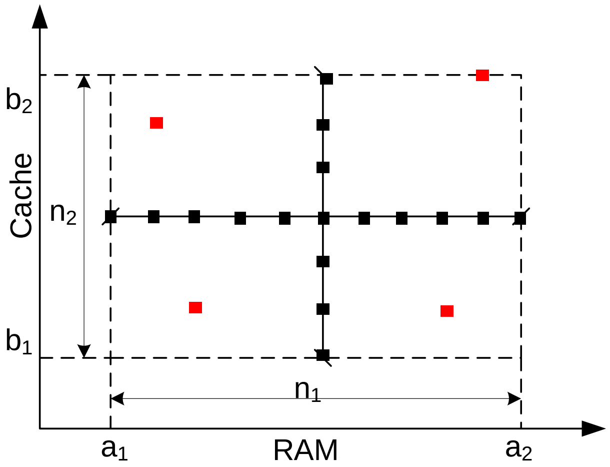
This is what most people end up doing. You choose an arbitrary configuration, i.e. a point in your factor space (e.g., CPU x, 16GB RAM, 1GB cache, 2 disks, TomCat server), and vary **one factor at a time**. For instance, you vary the RAM in its interval first, then you vary the cache, etc.

How many experiments will you need to carry out if you work like this? $1+\sum_{i=1}^{k}(n_i-1)$, if k is the number of factors, and factor i has n_i **levels**. If you need to **repeat each experiment** r times in independent conditions, then your total is $r\cdot\left[1+\sum_{i=1}^{k}(n_i-1)\right]$. The number of experiments grows **linearly** with the number of factors.

Why should you repeat experiments r times? Repeating experiments in independent conditions is the key to generating **IID samples**, on which to compute **sample statistics**. This is **very important** when doing simulations, and there is a recipe to do so. If you are doing **measurement**, repeating measurements helps you to assess the impact of **experimental errors**. Measurements are independent if the output of one measure does not influence the others.

Of course, the downside of this approach is that you are never going to explore **all the space of the factors**, and you are neglecting the possibility that **two (or more) factors may interact**, which limit's the usefulness of this technique for **tuning your system to maximum performance**.

Simple example: two factors. You are only exploring along the two straight lines. Therefore, you will never know if there are other combinations of levels which yield considerably better performance. It may well be that **"small cache, large memory"** is the best combination, but you will never know.



In the end, this type of analysis is quick, but it gives you a very **unreliable picture** of what happens in practice, from which you are tempted to draw **unwarranted conclusions**. For instance, if the performance along the vertical line does not change much, you may end up thinking that the cache size is not significant. Maybe it is not significant **when the RAM size is the one you experimented with**, and it becomes significant to the far left (or right).

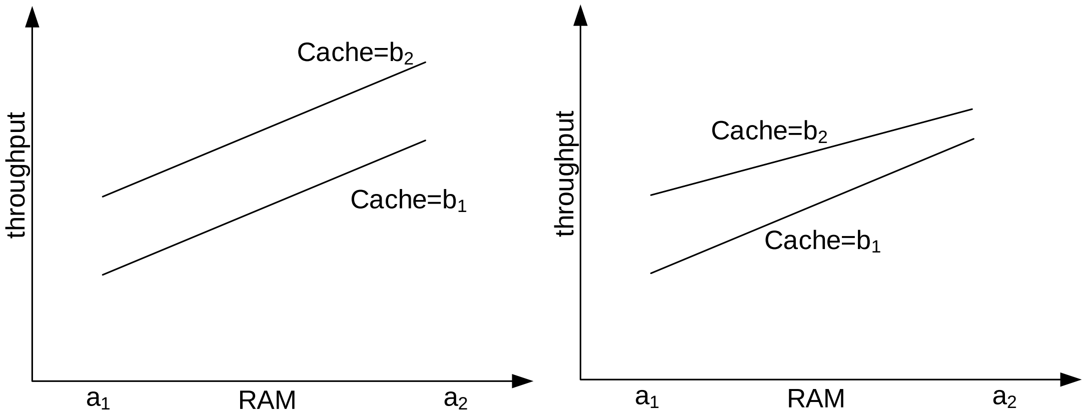
Full factorial analysis

If you want full knowledge of the factor space, an obvious solution is to explore **all the factor space**. With the above notation, the number of required experiments would be $\prod_{i=1}^{k} n_i$, or $r \cdot \prod_{i=1}^{k} n_i$ if you need replicas. It is clear that the number of experiments you need to perform gets quickly out of control, even with relatively few factors.

In the initial example, assuming that factors RAM and cache have 10 levels and factor "no. disks" has five, you end up with $2 \times 10 \times 10 \times 5 \times 2 = 2000$, or $2000 \cdot r$. If each experiment takes one hour (this is quite likely, with both simulation and measurement), you end up requiring **more than one year** to have all the results in with five replicas. And jolly good luck to you with storing the results so that you can remember which output came from which combination of factor levels one year later.

Another downside is that, if at some point you understand that you need **another factor** in your analysis (e.g., the NIC speed), then you have to undergo a lot more experiments. This is typical: very often, you start up an analysis thinking that you **already know** which factors you will need to consider, and it is only later that you discover you need **more**. Using this technique, it is likely that you will stop when you are fed up with doing experiments (or time and money for the project run out), rather than when you get the insight you require.

There is, of course, a **clear advantage** to this technique: the fact that you can assess the **interplay** of two or more factors. For instance, it may be that the benefits of having **a larger RAM** are more evident **if the cache is large,** or if the number of disks is large. You could never discover this with a single-factor analysis, but you surely will in a full-factorial analysis.



In the above graphs, we see that the metric of interest increases with **both** RAM and cache. However, in the left graph, the two factors seem to be **independent.** In fact, the increase in throughput when the RAM size changes from a_1 to a_2 is the same at the two cache levels. Instead, on the right, you can see that the metric still increases with **both** factors, but factors do **interact**: more specifically, the improvement due to increasing the RAM depends on the cache size. With a large cache, the performance improvement due to increasing the RAM is smaller. Knowing how factors interact is particularly important to know if you want to **tune** your system for maximum performance.

<u>$2^k r$ factorial analysis</u>

This is a third technique, which allows you to reap **most of the benefits** of the former, at a moderate implementation cost.

It consists in exploring **only the combinations of <u>extremes</u> of the factor levels**. This technique allows you to assess the **relative importance** of factors on your metric, so that you can decide which to investigate. It also yields information about the interplay between factors.

It requires you to analyze “only” 2^k configurations (or $2^k \cdot r$, if you need replicas), hence its name. Continuing the above example, you would need 32 experiments (or 160, with 5 replicas), instead of, respectively, 2000 or 10000. This would take less than a week, and leave you with a manageable number of output results to store and analyze.

Of course, this technique does not scale well with the number of factors. It scales **better** than the former, but does not allow you to analyze a system with, say, 20 factors. However, it is quite unlikely that you will encounter that many. If you do, there **are** techniques that allow you to reduce the number of experiments to the minimum necessary to get meaningful results (they are called $2^{k-p} \cdot r$, meaning that you discount p out of k factors). We will not go that deep.

Factorial analysis borrows heavily from **model fitting**, hence we will reuse much of the notation and concepts from the previous lectures.

Before delving deep into how it works, it is useful to remind ourselves that the above are techniques to **design experiments**. They work with **any kind of experiment**: from simulation experiments to **measurement** ones, and they can be used seamlessly in one case or the other.

4.1 2^k factorial analysis

The basics of factorial analysis are easier to understand if **errors and replication** are kept out of the way. For now, we will assume that we have one measurement per configuration, which is error-free. We will add back errors later on.

Take the following example. We measure the performance of a workstation in Million Instructions per Second (MIPS), with two configurations of cache and RAM, and obtain the following:

These are the two extremes of the interval

	Memory size (Gb)	
Cache size (Mb)	**4**	**16**
1	15	45
2	25	75

We want to assess what the impact of the two **factors** (namely, RAM and cache), on the performance. Call A the RAM and B the cache for ease of notation.

We define two variables x_A, x_B as follows

$$x_A = \begin{cases} -1 & mem = 4 \\ +1 & mem = 16 \end{cases}, x_B = \begin{cases} -1 & cache = 1 \\ +1 & cache = 2 \end{cases}$$

We can then fit the output variable to a **nonlinear regression model** having x_A, x_B as predictor variables as follows:

$$y = q_0 + q_A \cdot x_A + q_B \cdot x_B + q_{AB} \cdot x_A \cdot x_B$$

Where q_i are some coefficients to be determined. The model is non linear because it includes a product of predictor variables $x_A \cdot x_B$.

We have four observations, corresponding to four combinations of values for the two variables, i.e.

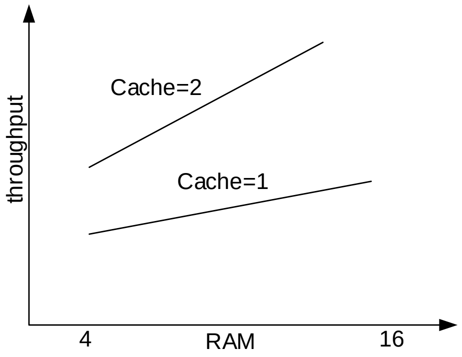
$$\begin{aligned}15 &= q_0 - q_A - q_B + q_{AB}\\ 45 &= q_0 + q_A - q_B - q_{AB}\\ 25 &= q_0 - q_A + q_B - q_{AB}\\ 75 &= q_0 + q_A + q_B + q_{AB}\end{aligned}$$

This is a 4x4 linear system, whose unknowns are the q_i, and – since the known term is non null, it has only one solution. You can solve it using Excel, if needed (but you won't need it). You solve the system and find the following:

$$y = 40 + 20 \cdot x_A + 10 \cdot x_B + 5 \cdot x_A \cdot x_B$$

The interpretation of the result is that:

- 40MIPS is the mean performance among all observations
- Factor A (the RAM) accounts for +20 MIPS in the model
- Factor B accounts for +10 MIPS in the model
- The interaction between A and B accounts for 5 MIPS.



The performance **increases with both factors**, and it increases **more when the other is high**.

In the general case, if you have 2 factors, you need 4 experiments (2 factors times 2 levels), and you get the following results:

Experiment	A	B	y
1	-1	-1	y_1
2	+1	-1	y_2
3	-1	+1	y_3
4	+1	+1	y_4

Where y_1 is the value corresponding to the observation when both factors are at low level, etc.

You can write down four equations, i.e.,

$$\begin{aligned}y_1 &= q_0 - q_A - q_B + q_{AB}\\ y_2 &= q_0 + q_A - q_B - q_{AB}\\ y_3 &= q_0 - q_A + q_B - q_{AB}\\ y_4 &= q_0 + q_A + q_B + q_{AB}\end{aligned}$$

And observe that:

- if you sum all four, you get $y_1 + y_2 + y_3 + y_4 = 4q_0$
- if you sum them with alternate signs (-, +, -, +) you get $-y_1 + y_2 - y_3 + y_4 = 4q_A$
- if you sum them with alternate signs (-, -, +, +) you get $-y_1 - y_2 + y_3 + y_4 = 4q_B$
- again, if you sum them with alternate signs (+ - - +), you get $+y_1 - y_2 - y_3 + y_4 = 4q_{AB}$

Therefore, in the end, you **don't have to use matrix algebra to solve that linear system**: you just need to look at the signs and do some easy sums.

It helps a lot if numbers are organized in a **table**, where signs are substituted by +1 or -1 and column totals are obtained by multiplying the corresponding observation for the specified sign, and summing up. The procedure is as follows:

1) you write down a column of your y values
2) for each measurement, you record the level of the factors which produced that value, in the form of a -1 (lo) or $+1$ (hi), in columns A and B
3) You add a column of $+1$ (I) for the mean response (left), and a column AB for the factors interplay (whose value is the product of the signs of the factors).
4) You get the algebraic sums of the ys times the signs at the end of each column, and you divide these by the number of observations (i.e., 4, in this example).

I	A	B	AB	Y
1	-1	-1	1	15
1	1	-1	-1	45
1	-1	1	-1	25
1	1	1	1	75
160	80	40	20	Algebraic sum of the *y*s
40	20	10	5	qi=total/4

Now, the important thing about factorial analysis is that it allows you to determine **the relative importance of the factors**. This is done through a procedure called **analysis of variation**. Mind the word "**variation**", which is not the same as "variance".

This is done as follows. first, we compute the **sum of squares total (SST)** of the observed response, i.e. $SST = \sum_{i=1}^{4}(y_i - \bar{y})^2$. This is the **total variation** of the response variable.

Then, we acknowledge that:

$$\begin{aligned} SST &= 2^2 \cdot {q_A}^2 + 2^2 \cdot {q_B}^2 + 2^2 \cdot {q_{AB}}^2 \\ &= SSA \quad + \quad SSB \quad + SSAB \end{aligned}$$

The above equality will be proved later on. Each of the three terms is the **sum squares due to A (B, AB)**, and is equal to the term written in the previous line.

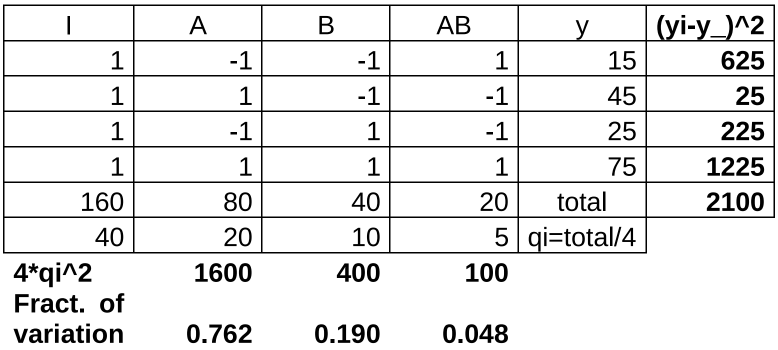
Now, the fraction of variation explained by factor $x, x = A, B, AB$ is:

$$f_x = \frac{SSx}{SST}$$

In our example, we have $\bar{y} = 40$,

$$
\begin{aligned}
SST &= (15-40)^2 + (45-40)^2 + (25-40)^2 + (75-40)^2 \\
&= 25^2 + 5^2 + 15^2 + 35^2 \\
&= 2100
\end{aligned}
$$

And the percentage of variation is computed at the bottom of the table.



I	A	B	AB	y	**(yi-y_)^2**
1	-1	-1	1	15	**625**
1	1	-1	-1	45	**25**
1	-1	1	-1	25	**225**
1	1	1	1	75	**1225**
160	80	40	20	total	**2100**
40	20	10	5	qi=total/4	
4*qi^2	**1600**	**400**	**100**		
Fract. of variation	**0.762**	**0.190**	**0.048**		

In practice, you **never** compute the SST using the sum square total, but it is a lot easier to compute the SSx for each factor and then sum them up to get the SST.

The above results say that 76.2% of the variation is due to memory, 19% to the cache size, and 4.8% to the interaction of both.

If you want to improve your system, increasing memory will yield the highest performance return.

Now, for the proof of the above equality:

$$
\begin{aligned}
SST &= 2^2 \cdot {q_A}^2 + 2^2 \cdot {q_B}^2 + 2^2 \cdot {q_{AB}}^2 \\
&= SSA \quad + \quad SSB \quad + SSAB
\end{aligned}
$$

We start with the definition. Call x_{Ai} the value of A for observation i (i.e., +1 or -1).

$$
\begin{aligned}
SST &= \sum_{i=1}^{4} (y_i - \bar{y})^2 \\
&= \sum_{i=1}^{4} [(\cancel{q_0} + q_A \cdot x_{Ai} + q_B \cdot x_{Bi} + q_{AB} \cdot x_{Ai} \cdot x_{Bi}) - \cancel{q_0}]^2 \\
&= \sum_{i=1}^{4} [{q_A}^2 \cdot {x_{Ai}}^2 + {q_B}^2 \cdot {x_{Bi}}^2 + {q_{AB}}^2 \cdot (x_{Ai} \cdot x_{Bi})^2 + CPT] \\
&= {q_A}^2 \cdot \sum_{i=1}^{4} {x_{Ai}}^2 + {q_B}^2 \cdot \sum_{i=1}^{4} {x_{Bi}}^2 + {q_{AB}}^2 \cdot \sum_{i=1}^{4} (x_{Ai} \cdot x_{Bi})^2 + \sum_{i=1}^{4} CPT \\
&= 4{q_A}^2 + 4{q_A}^2 + 4{q_{AB}}^2 + \sum_{i=1}^{4} CPT
\end{aligned}
$$

Where CPT are the cross-product terms, which we now show to be identically null. These are, in fact:

$$
\begin{aligned}
&2 \cdot q_A \cdot q_B \left(\sum_{i=1}^{4} x_{Ai} \cdot x_{Bi}\right) + 2 \cdot q_A \cdot q_{AB} \left(\sum_{i=1}^{4} \cancel{x_{Ai} \cdot x_{Ai}} \cdot x_{Bi}\right) + 2 \cdot q_B \cdot q_{AB} \left(\sum_{i=1}^{4} \cancel{x_{Bi}} \cdot x_{Ai} \cdot \cancel{x_{Bi}}\right) \\
&= 2 \cdot q_A \cdot q_B \left(\sum_{i=1}^{4} x_{Ai} \cdot x_{Bi}\right) + 2 \cdot q_A \cdot q_{AB} \sum_{i=1}^{4} x_{Bi} + 2 \cdot q_B \cdot q_{AB} \sum_{i=1}^{4} x_{Ai}
\end{aligned}
$$

The last two sums are obviously null, and so is the first one, which includes the **inner product of the columns A and B** of the sign table. The two columns are orthogonal by definition.

This proves the equality.

We now generalize the result to any number of factors. For instance, take an experiment with three factors. You just need to build a **larger sign table**, and everything else stays the same. In the sign table, all the combinations of the signs for the single factors A, B, C must be present (much like at the left-hand side of a **truth table**), and the signs for the **combinations** are computed accordingly.

	I	A	B	C	AB	AC	BC	ABC	y
	1	-1	-1	-1	1	1	1	-1	
	1	1	-1	-1	-1	-1	1	1	
	1	-1	1	-1	-1	1	-1	1	
	1	1	1	-1	1	-1	-1	-1	
	1	-1	-1	1	1	-1	-1	1	
	1	1	-1	1	-1	1	-1	-1	
	1	-1	1	1	-1	-1	1	-1	
	1	1	1	1	1	1	1	1	
sum									
mean (qi)									
8*qi^2									
frac. variation									

Exercise

We measure the MIPS of a system in several configurations, having

- 1 or 2 processors
- 4 or 16 Mb RAM
- 1 or 2 kB cache

The results are as follows:

	4Mb mem		**16Mb mem**	
Cache size (kb)	**1P**	**2P**	**1P**	**2P**
1	14	46	22	58
2	10	50	34	86

Solution

We just need to decide the factors: A is memory, B is cache, C is number of processors. Then we put the numbers into the above table and make some straightforward computations. Using Excel, the whole process takes one minute.

I	A	B	C	AB	AC	BC	ABC	y
1	-1	-1	-1	1	1	1	-1	14
1	1	-1	-1	-1	-1	1	1	22
1	-1	1	-1	-1	1	-1	1	10
1	1	1	-1	1	-1	-1	-1	34
1	-1	-1	1	1	-1	-1	1	46
1	1	-1	1	-1	1	-1	-1	58
1	-1	1	1	-1	-1	1	-1	50
1	1	1	1	1	1	1	1	86

sum	320	80	40	160	40	16	24	8	
mean (qi)	40	10	5	20	5	2	3	1	
8*qi^2		800	200	3200	200	32	72	8	4512
variation		18%	4%	71%	4%	1%	2%	0%	

The result is that the most important factor is C, the number of processors, which accounts for 71% of the variation. Then you have memory, and cache comes third. The interactions are almost negligible. This means that, if you want to study this system, you can just **fix the cache** to some value and vary the memory and the number of processors.

4.2 $2^k \cdot r$ factorial analysis

The limitation of the above design is that it does not account for **experimental errors**. These can only be assessed if **independent replications** are performed. If we repeat the experiment r times, then we have r values for each design alternative. In this case the model is slightly different, i.e. (assuming two factors for simplicity):

$$y = q_0 + q_A \cdot x_A + q_B \cdot x_B + q_{AB} \cdot x_A \cdot x_B + e$$

In this case, we proceed as before to estimate the q coefficients, using the **sample mean** over the r replications as a value for the response.

Assume $r = 3$, i.e. we repeat each experiment three times, hence we get three results.

1) We compute the **row sample mean** to get the mean value for each experiment
2) We repeat the same type of analysis using the sample mean as a result, and we compute the values for the q_i coefficients.

	I	A	B	AB	y(1)	y(2)	y(3)	Row sample mean ym
	1	-1	-1	1	15	18	12	**15**
	1	1	-1	-1	45	48	51	**48**
	1	-1	1	-1	25	28	19	**24**
	1	1	1	1	75	75	81	**77**
Sum	164	86	38	20				
mean (qi)	**41**	**21.5**	**9.5**	**5**				

3) Once the effects have been computed, we can estimate the **experimental errors** and the **sum of squared errors (SSE)** and add them to the table:

observations	**errors**

	I	A	B	AB	y(1)	y(2)	y(3)	Row s.m. ym	**y(1)-ym**	**y(2)-ym**	**y(3)-ym**	
	1	-1	-1	1	15	18	12	15	**0**	**3**	**-3**	
	1	1	-1	-1	45	48	51	48	**-3**	**0**	**3**	
	1	-1	1	-1	25	28	19	24	**1**	**4**	**-5**	
	1	1	1	1	75	75	81	77	**-2**	**-2**	**4**	
Sum	164	86	38	20					**-4**	**5**	**-1**	sum
mean (qi)	41	21.5	9.5	5					**-1**	**1.25**	**-0.25**	mean
Ssq									**14**	**29**	**59**	Ssq

The SSE is 102 (=14+29+59), computed as the sum of the squares of **all the elements in the error matrix**. We have again a similar equation as before (no proof this time), i.e.:

$$\sum_{i,j}\left(y_{i,j}-\bar{y}\right)^2 = 2^2\cdot r\cdot {q_A}^2 + 2^2\cdot r\cdot {q_B}^2 + 2^2\cdot r\cdot {q_{AB}}^2 + \sum_{i,j}{e_{i,j}}^2$$

$$SST = \quad SSA \qquad + \quad SSB \qquad + \quad SSAB \quad + SSE$$

Where $\bar{y}$ is the mean response over **the whole results matrix**. Note that this time each coefficient is **multiplied by the number of replicas as well.**

					observations				**errors**		
	I	A	B	AB	y(1)	y(2)	y(3)	s.m. y	**y(1)-ym**	**y(2)-ym**	**y(3)-ym**
	1	-1	-1	1	15	18	12	15	**0**	**3**	**-3**
	1	1	-1	-1	45	48	51	48	**-3**	**0**	**3**
	1	-1	1	-1	25	28	19	24	**1**	**4**	**-5**
	1	1	1	1	75	75	81	77	**-2**	**-2**	**4**
Sum	164	86	38	20					**-4**	**5**	**-1**
mean (qi)	41	21.5	9.5	5					**-1**	**1.25**	**-0.25**
4*3*qi^2		**5547**	**1083**	**300**							**102**
variation		**78.9%**	**15.4%**	**4.3%**							**1.5%**

This way, we can compute the fraction of variation $f_x = \frac{SSx}{SST}$, this time listing the contribution of the experimental error explicitly.

Fraction $\frac{SSE}{SST}$ is called **unexplained variation**, and it should be small. If it is not, then the model may not be appropriate. The above example states that factor A accounts fort 78.9% of the variation, etc.

There are a **couple of subtleties** that we have not addressed so far. The first one is the fact that we are using a **random sample** (of three observations per experiment), hence our **effects** (i.e., the q coefficients) are themselves **random variables**. Whenever we have randomness, it is fundamental to compute a **confidence interval** for our measures.

In the above example, in fact, if the confidence interval for factor B (the cache) was [-3;15], we could **not conclude that the effect of the cache is significant**. On the other hand, if the CI for the effect does not include 0, then we can be sure that the factor is significant.

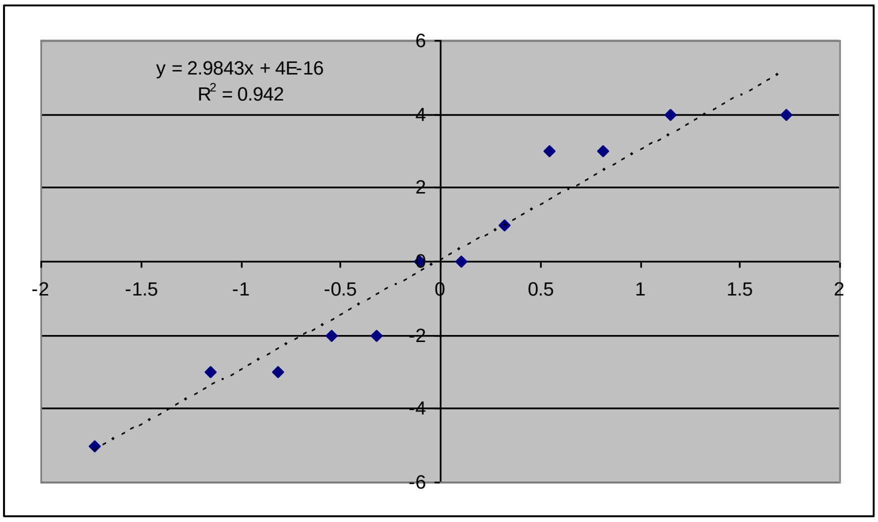
The other missing piece is that the above **nonlinear regression model** works under some assumptions, notably that:

residuals (errors) are IID Normal RVs with a null mean and a constant standard deviation.

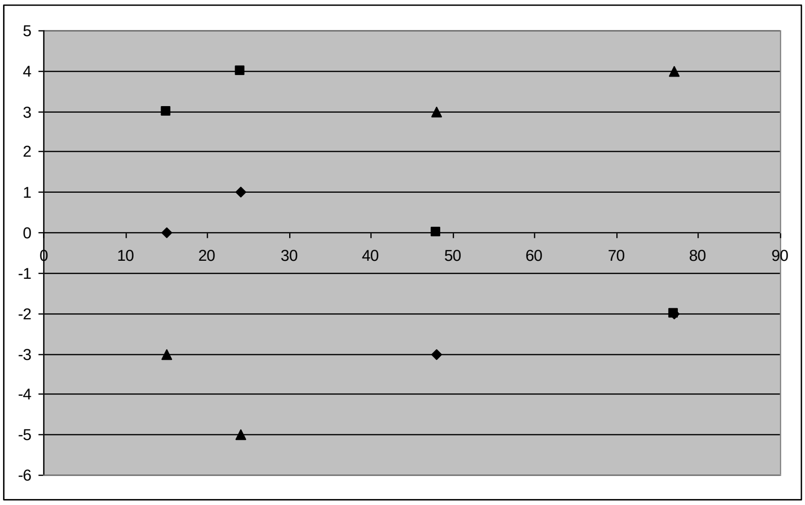
and we need to **test** these hypotheses. More to the point, we can **only** compute confidence intervals given that the above hypotheses hold.

The way to test the Normal hypothesis is to make a **QQ plot of the residuals**, as usual. In the above case we get:

i	quantile	NormalQ	Q
1	0.041667	-1.73416	-5
2	0.125	-1.14921	-3
3	0.208333	-0.81009	-3
4	0.291667	-0.54654	-2
5	0.375	-0.3173	-2
6	0.458333	-0.10416	0
7	0.541667	0.104165	0
8	0.625	0.317299	1
9	0.708333	0.546537	3
10	0.791667	0.810092	3
11	0.875	1.149208	4
12	0.958333	1.734156	4



The way to test for constant standard deviation is to plot the residuals vs. the (average) predicted response, i.e.:



And the plot shows no trend and a fairly constant standard deviation with the predicted response. Note that, in any case, we can **ignore trends** if the errors are one or more orders of magnitude below the predicted response.

We should also plot **the residuals vs. the experiment number.** However, we do not know how the sequence of observations has been derived, so this is pointless. We assume that there is no bias.

Once we have verified all the hypotheses, we can **compute confidence intervals for the effects**. The confidence intervals are computed based on the error variance, which is:

$$\sigma_e^2 = \frac{SSE}{2^2(r-1)}$$

Note that the term at the denominator accounts for the fact that the number of **degrees of freedom** for the errors is $(r-1)$ per configuration, since **all the errors must sum to zero**. Hence, you can only choose $r-1$ errors independently, and the last one will be dependent on the others.

The confidence interval for all the effects is thus:

$$q_i \mp t_{\alpha/2,\left(2^2\cdot(r-1)\right)} \cdot \sqrt{\frac{\sigma_e^2}{2^2 \cdot r}}$$

Where t is the percentile of the Student’s T distribution.

Any effect whose CI does not include zero is **significant** at the specified level of confidence.

Obviously enough, the above theory can be readily **extended to an arbitrary number of factors**, simply by changing few numbers here and there. We will not go through the details, since it is straightforward.

5 Appendix

5.1 Useful probability tables

TABLE 1 - Standard Normal Distribution Function $\Phi(x)$

x	**.00**	**.01**	**.02**	**.03**	**.04**	**.05**	**.06**	**.07**	**.08**	**.09**
.0	.5000	.5040	.5080	.5120	.5160	.5199	.5239	.5279	.5319	.5359
.1	.5398	.5438	.5478	.5517	.5557	.5596	.5636	.5675	.5714	.5753
.2	.5793	.5832	.5871	.5910	.5948	.5987	.6026	.6064	.6103	.6141
.3	.6179	.6217	.6255	.6293	.6331	.6368	.6406	.6443	.6480	.6517
.4	.6554	.6591	.6628	.6664	.6700	.6736	.6772	.6808	.6844	.6879
.5	.6915	.6950	.6985	.7019	.7054	.7088	.7123	.7157	.7190	.7224
.6	.7257	.7291	.7324	.7357	.7389	.7422	.7454	.7486	.7517	.7549
.7	.7580	.7611	.7642	.7673	.7704	.7734	.7764	.7794	.7823	.7852
.8	.7881	.7910	.7939	.7967	.7995	.8023	.8051	.8078	.8106	.8133
.9	.8159	.8186	.8212	.8238	.8264	.8289	.8315	.8340	.8365	.8389
1.0	.8413	.8438	.8461	.8485	.8508	.8531	.8554	.8577	.8599	.8621
1.1	.8643	.8665	.8686	.8708	.8729	.8749	.8770	.8790	.8810	.8830
1.2	.8849	.8869	.8888	.8907	.8925	.8944	.8962	.8980	.8997	.9015
1.3	.9032	.9049	.9066	.9082	.9099	.9115	.9131	.9147	.9162	.9177
1.4	.9192	.9207	.9222	.9236	.9251	.9265	.9279	.9292	.9306	.9319
1.5	.9332	.9345	.9357	.9370	.9382	.9394	.9406	.9418	.9429	.9441
1.6	.9452	.9463	.9474	.9484	.9495	.9505	.9515	.9525	.9535	.9545
1.7	.9554	.9564	.9573	.9582	.9591	.9599	.9608	.9616	.9625	.9633
1.8	.9641	.9649	.9656	.9664	.9671	.9678	.9686	.9693	.9699	.9706
1.9	.9713	.9719	.9726	.9732	.9738	.9744	.9750	.9756	.9761	.9767
2.0	.9772	.9778	.9783	.9788	.9793	.9798	.9803	.9808	.9812	.9817
2.1	.9821	.9826	.9830	.9834	.9838	.9842	.9846	.9850	.9854	.9857
2.2	.9861	.9864	.9868	.9871	.9875	.9878	.9881	.9884	.9887	.9890
2.3	.9893	.9896	.9898	.9901	.9904	.9906	.9909	.9911	.9913	.9916
2.4	.9918	.9920	.9922	.9925	.9927	.9929	.9931	.9932	.9934	.9936
2.5	.9938	.9940	.9941	.9943	.9945	.9946	.9948	.9949	.9951	.9952
2.6	.9953	.9955	.9956	.9957	.9959	.9960	.9961	.9962	.9963	.9964
2.7	.9965	.9966	.9967	.9968	.9969	.9970	.9971	.9972	.9973	.9974
2.8	.9974	.9975	.9976	.9977	.9977	.9978	.9979	.9979	.9980	.9981
2.9	.9981	.9982	.9982	.9983	.9984	.9984	.9985	.9985	.9986	.9986
3.0	.9987	.9987	.9987	.9988	.9988	.9989	.9989	.9989	.9990	.9990
3.1	.9990	.9991	.9991	.9991	.9992	.9992	.9992	.9992	.9993	.9993
3.2	.9993	.9993	.9994	.9994	.9994	.9994	.9994	.9995	.9995	.9995
3.3	.9995	.9995	.9995	.9996	.9996	.9996	.9996	.9996	.9996	.9997
3.4	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9998

TABLE 2 - Values of $t_{\alpha,n}$

n	α =0.10	α =0.05	α =0.025	α =0.01	α =0.005
1	3.078	6.314	12.706	31.821	63.657
2	1.886	2.920	4.303	6.965	9.925
3	1.638	2.353	3.182	4.541	5.841
4	1.533	2.132	2.776	3.474	4.604
5	1.476	2.015	2.571	3.365	4.032
6	1.440	1.943	2.447	3.143	3.707
7	1.415	1.895	2.365	2.998	3.499
8	1.397	1.860	2.306	2.896	3.355
9	1.383	1.833	2.262	2.821	3.250
10	1.372	1.812	2.228	2.764	3.169
11	1.363	1.796	2.201	2.718	3.106
12	1.356	1.782	2.179	2.681	3.055
13	1.350	1.771	2.160	2.650	3.012
14	1.345	1.761	2.145	2.624	2.977
15	1.341	1.753	2.131	2.602	2.947
16	1.337	1.746	2.120	2.583	2.921
17	1.333	1.740	2.110	2.567	2.898
18	1.330	1.734	2.101	2.552	2.878
19	1.328	1.729	2.093	2.539	2.861
20	1.325	1.725	2.086	2.528	2.845
21	1.323	1.721	2.080	2.518	2.831
22	1.321	1.717	2.074	2.508	2.819
23	1.319	1.714	2.069	2.500	2.807
24	1.318	1.711	2.064	2.492	2.797
25	1.316	1.708	2.060	2.485	2.787
26	1.315	1.706	2.056	2.479	2.779
27	1.314	1.703	2.052	2.473	2.771
28	1.313	1.701	2.048	2.467	2.763
29	1.311	1.699	2.045	2.462	2.756
∞	1.282	1.645	1.960	2.326	2.576

TABLE 3 – Chi-square percentiles

	Alpha level					
Degrees of freedom	**0.1**	**0.05**	**0.02**	**0.01**	**0.002**	**0.001**
1	2.706	3.841	5.412	6.635	9.550	10.828
2	4.605	5.991	7.824	9.210	12.429	13.816
3	6.251	7.815	9.837	11.345	14.796	16.267
4	7.779	9.488	11.668	13.277	16.924	18.467
5	9.236	11.071	13.388	15.086	18.908	20.516
6	10.645	12.592	15.033	16.812	20.791	22.458
7	12.017	14.067	16.622	18.475	22.601	24.323
8	13.362	15.507	18.168	20.090	24.352	26.125
9	14.684	16.919	19.679	21.666	26.057	27.878
10	15.987	18.307	21.161	23.209	27.722	29.589
20	28.412	31.410	35.020	37.566	43.073	45.316

TABLE 4 – Kolmogorov-Smirnov Distributions

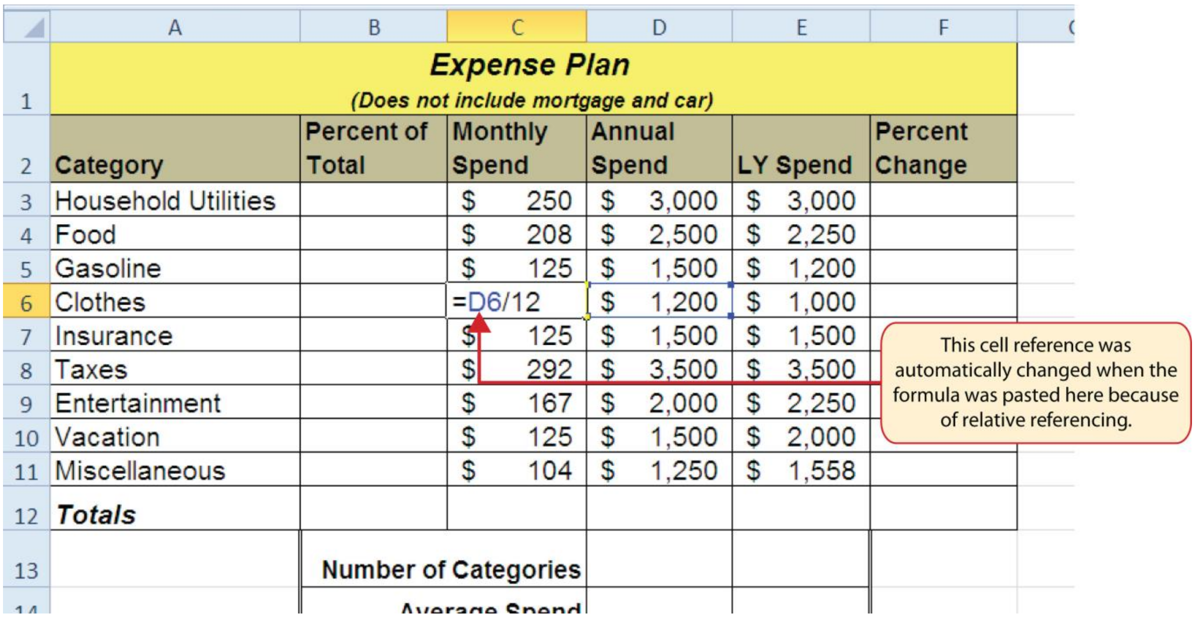
	Level of significance				
Sample size n	**0.20**	**0.15**	**0.10**	**0.05**	**0.01**
1	.900	.925	.950	.975	.995
2	.684	.726	.776	.842	.929
3	.565	.597	.642	.708	.828
4	.494	.525	.564	.624	.733
5	.446	.474	.510	.565	.669
6	.410	.436	.470	.521	.618
7	.381	.405	.438	.486	.577
8	.358	.381	.411	.457	.543
9	.339	.360	.388	.432	.514
10	.322	.342	.368	.410	.490
11	.307	.326	.352	.391	.468
12	.295	.313	.338	.375	.450
13	.284	.302	.325	.361	.433
14	.274	.292	.314	.349	.418
15	.266	.283	.304	.338	.404
16	.258	.274	.295	.328	.392
17	.250	.266	.286	.318	.381
18	.244	.259	.278	.309	.371
19	.237	.252	.272	.301	.363
20	.231	.246	.264	.294	.356
25	.210	.220	.240	.270	.320
30	.190	.200	.220	.240	.290
35	.180	.190	.210	.230	.270
OVER 35	1.07/sqrt(n)	1.14/sqrt(n)	1.22/sqrt(n)	1.36/sqrt(n)	1.63/sqrt(n)

5.2 Notes on Excel[5]

Most of the things that are explained here are good for any version of Excel (you just need to figure out where the menus and items are located), and for the Calc Spreadsheet of Open Office (the same caveat applies).

Excel is a spreadsheet, i.e. a means for automating computations. You can write down formulas referring **cells** as well as constants, and using **functions**. Formulas begin with "=".

Cell references can be either **relative** or **absolute**. Relative references change when you paste the formula to another cell. Absolute references do not.



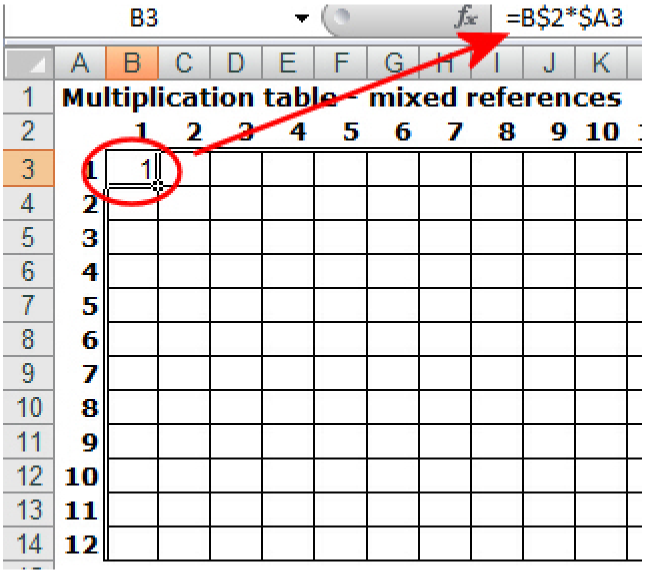
	A	B	C	D	E	F
1	***Expense Plan*** *(Does not include mortgage and car)*					
2	**Category**	**Percent of Total**	**Monthly Spend**	**Annual Spend**	**LY Spend**	**Percent Change**
3	Household Utilities		$ 250	$ 3,000	$ 3,000	
4	Food		$ 208	$ 2,500	$ 2,250	
5	Gasoline		$ 125	$ 1,500	$ 1,200	
6	Clothes		=D6/12	$ 1,200	$ 1,000	
7	Insurance		$ 125	$ 1,500	$ 1,500	
8	Taxes		$ 292	$ 3,500	$ 3,500	
9	Entertainment		$ 167	$ 2,000	$ 2,250	
10	Vacation		$ 125	$ 1,500	$ 2,000	
11	Miscellaneous		$ 104	$ 1,250	$ 1,558	
12	***Totals***					
13		**Number of Categories**				
14		**Average Spend**				

This cell reference was automatically changed when the formula was pasted here because of relative referencing.

In the above example, the formula =D6/12 written in cell C6 includes a **relative reference** to cell D6. This means that, if you copy that formula to the cell below, it will automatically be updated to =D7/12, and so on.

[5] Figures are taken from Internet pages. I hope no one takes offense.

An **absolute reference** is done by “blocking” the column and/or row using the $ symbol. This way, the blocked reference will not change when you copy/paste the formulas, as in the following example. Note that – for the first reference – the column is relative and the row is absolute, whereas for the second reference the column is absolute and the row is relative. Both row/col reference can be absolute in the same reference.



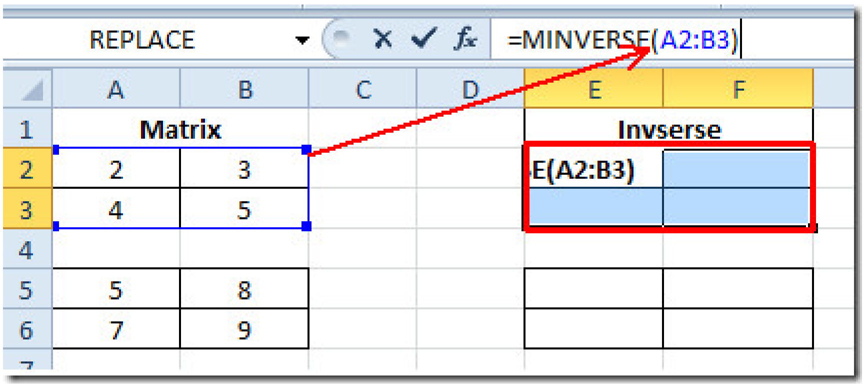
B3 =B$2*$A3

	A	B	C	D	E	F	G	H	I	J	K
1	Multiplication table - mixed references										
2		1	2	3	4	5	6	7	8	9	10
3	1	1									
4	2										
5	3										
6	4										
7	5										
8	6										
9	7										
10	8										
11	9										
12	10										
13	11										
14	12										

Some formulas in Excel either require matrix-form operands or have a matrix-form result. These should be entered as follows

1) You first select the interval of cells where you want the result to appear
2) Then you press “=”. Note that you will be entering the formula in the top-left corner of the interval of cells that you have selected, but the whole interval stays highlighted
3) You write down the formula
4) You press CTRL+SHIFT+ENTER instead of ENTER.

This way, the formula will be attributed as an atomic unit to all the interval, as in the following example. MINVERSE is an Excel function that computes the inverse of a matrix. A2:B3 is the interval where the matrix to be inverted is stored. The formula is written having selected E2:F4 as a target.

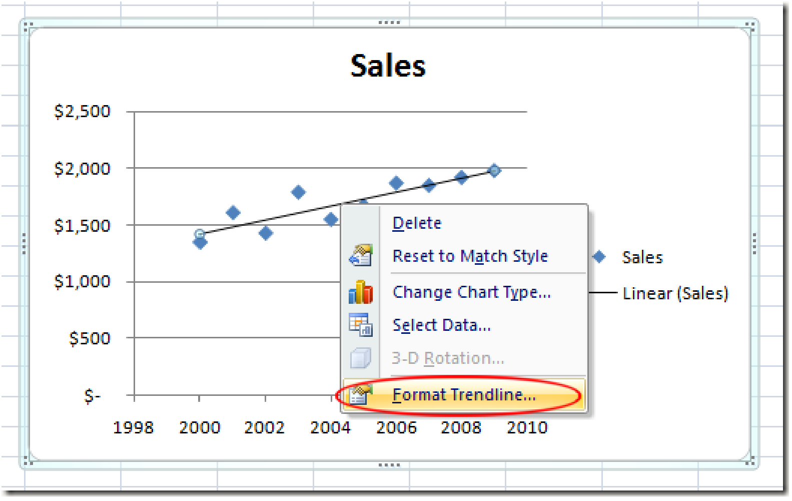


REPLACE =MINVERSE(A2:B3)

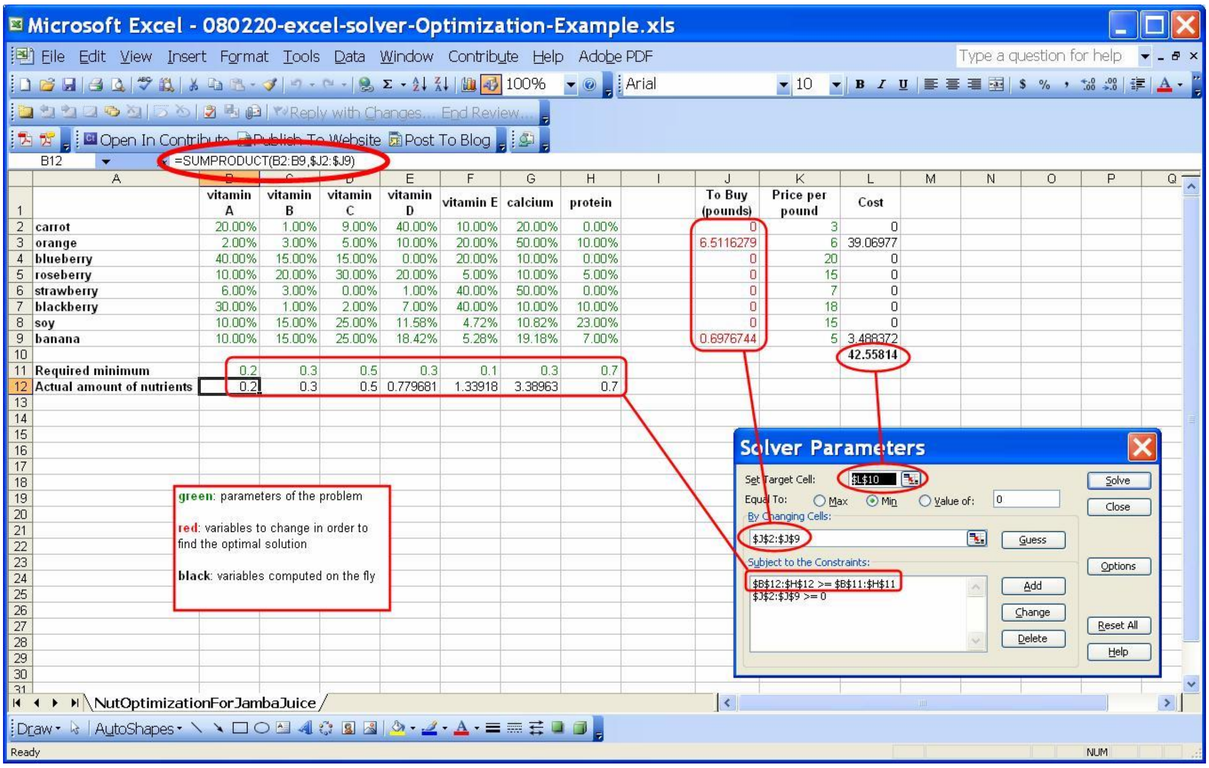
	A	B	C	D	E	F
1	Matrix				Invserse	
2	2	3			E(A2:B3)	
3	4	5				
4						
5	5	8				
6	7	9				

Excel can make graphs. Most of the graphs we need to make are **scatterplots**, which requires the user to specify sequences of tuples (x, y_1, y_2, …. y_n), to be used for the plot. The first element will be plotted on the *x* axis, the others will be *n* curves having abscissa *x* and ordinates y_i.

Trendlines (i.e., regression lines) can be added to a scatterplot by right-clicking on the points. Trendlines can be formatted so as to display the **regression equation** as well as the **coefficient of determination R^2**.



Excel can solve **non-linear constrained optimization problems** through an add-on component called **solver**. You may have to install it separately through the "option" menu. The latter allows you to specify the cell that includes the objective function, the cells including the constraints, and the cells that represent the variables (which should be contiguous). You only have to put formulas that tie the objective function to the variables (using references) and have the solver do the work.


Microsoft Excel - 080220-excel-solver-Optimization-Example.xls
File Edit View Insert Format Tools Data Window Contribute Help Adobe PDF
Type a question for help
100%
Arial
10
Reply with Changes... End Review...
Open In Contribute
Publish To Website
Post To Blog
NutOptimizationForJambaJuice
Draw
AutoShapes
Ready
NUM

B12 =SUMPRODUCT(B2:B9,$J2:$J9)

	A	B	C	D	E	F	G	H	I	J	K	L
1		vitamin A	vitamin B	vitamin C	vitamin D	vitamin E	calcium	protein		To Buy (pounds)	Price per pound	Cost
2	carrot	20.00%	1.00%	9.00%	40.00%	10.00%	20.00%	0.00%		0	3	0
3	orange	2.00%	3.00%	5.00%	10.00%	20.00%	50.00%	10.00%		6.5116279	6	39.06977
4	blueberry	40.00%	15.00%	15.00%	0.00%	20.00%	10.00%	0.00%		0	20	0
5	roseberry	10.00%	20.00%	30.00%	20.00%	5.00%	10.00%	5.00%		0	15	0
6	strawberry	6.00%	3.00%	0.00%	1.00%	40.00%	50.00%	0.00%		0	7	0
7	blackberry	30.00%	1.00%	2.00%	7.00%	40.00%	10.00%	10.00%		0	18	0
8	soy	10.00%	15.00%	25.00%	11.58%	4.72%	10.82%	23.00%		0	15	0
9	banana	10.00%	15.00%	25.00%	18.42%	5.28%	19.18%	7.00%		0.6976744	5	3.488372
10												**42.55814**
11	Required minimum	0.2	0.3	0.5	0.3	0.1	0.3	0.7				
12	Actual amount of nutrients	0.2	0.3	0.5	0.779681	1.33918	3.38963	0.7				

green: parameters of the problem

red: variables to change in order to find the optimal solution

black: variables computed on the fly

Solver Parameters

Set Target Cell: $L$10

Equal To: ○ Max ● Min ○ Value of: 0

By Changing Cells: $J$2:$J$9

Subject to the Constraints:

$B$12:$H$12 >= $B$11:$H$11
$J$2:$J$9 >= 0

Another Excel add-on is the **data analysis** package (to be installed the same way as the solver). It includes several functions, one of which is called **sampling**. The latter allows you to subsample a sample of data, either periodically or randomly, specifying the number of destination.

If your spreadsheet does not support subsampling, then a viable alternative is to work as follows:

- Add an observation id (from 1 to *n*) in an adjacent column.
- Generate a string of **random numbers**, as many as your originale sample, and write them down in another adjacent column. Most spreadsheet can generate random numbers.
- Sort your tuples (id, rand_i, x_i) data according to the second column. This creates a **random sorting** of your sample.
- Pick as many consecutive values of your (randomly sorted) sample as you need (e.g., ½, 1/3, etc.). By doing so, you are subsampling at random with a probability ½, 1/3, etc.
- Re-sort your subsample according to the observation id (otherwise you would alter correlation yourself by changing the order in which the observations are recorded).

6 Labs

6.1 Lab 1 – computing indexes, fitting probabilities

Take the Excel file and

1) Plot the ECDF of the sample
2) Plot a histogram of the sample, selecting the right bucket width. Try using Excel function *frequenza* or the menu tab *analisi dei dati*.
3) Compute the mean, median, mode, quartiles and IQR
4) Are there outliers beyond the 1.5 IQR limit (both above and below)?
5) Plot a Lorenz curve of the sample
6) Compute the sample variance, MAD and Lorenz Curve gap
7) Fit the sample to a Normal distribution, and estimate the mean and variance of the above Normal from the fitting equation
8) Discuss fitting the sample to a Uniform distribution. Is this a good idea? Why? What is the result in any case?
9) Discuss fitting the sample to an Exponential distribution. Is this a good idea? Why? What is the result in any case?

Exercise 2

Using the Excel functions, compute how good an approximation the following formula is for the standard Normal percentiles:

$$t_i = 4.91 \cdot [i^{0.14} - (1-i)^{0.14}]$$

Plot the absolute relative error as a function of the percentile. Where are the highest errors?

Exercise 3

Use the *random (casuale)* Excel function to generate n=1000 random numbers taken from U(0,1).

1) Show that these numbers are uniformly distributed.
2) Estimate the mean using
   a) the sample mean of the observations
   b) the MLE for the uniform
3) assume that the population mean is $\mu = 0.5$. Plot a graph of $|\mu - \bar{X}|$ and $|\mu - MLE|$ against the number of observations in the sample, and observe which of the two converges faster.

6.2 Lab 2 – independence, model fitting

Exercise 1

Consider the sample in the first Excel sheet.

1) Check whether it is IID or not using lag plots and correlograms
2) Subsample it until the assumption of IID-ness is consistent with the data
3) Compute the sample mean and 95% confidence interval
4) As a counterexample, compute the sample mean and 95% CI *assuming* that the original sample consists of IID RVs.

Exercise 2

Consider the data in the second Excel sheet

1. Fit the response times for the two systems using linear models
2. Test the hypotheses
3. Check whether there are significant differences between the two systems, using 90% CIs
4. Make a prediction for the response time with 500 bytes, and associate a confidence interval to that prediction.

Exercise 3

Consider the data in the third Excel sheet

1. fit them to the most appropriate model
2. test the hypotheses underlying the model
3. if the hypotheses are not verified, try a transformation and re-do the fit

6.3 Lab 3 – factorial analysis, testing of RNGs

Exercise

Consider the case of two processors A1, A2, tested with two benchmarks (i.e., "standard" sample workloads) B1, B2. Each experiment was repeated three times, to account for experimental errors. The data (execution times) are as follows:

	B1	**B2**
A1	85.10, 79.50, 147.90	0.955, 0.933, 1.122
A2	0.891, 1.047, 1.072	0.0148, 0.0126, 0.0118

1) Compute the contribution of the two factors using a linear response model
2) Test the hypotheses and discuss the accuracy of the linear response model
3) Re-do the work with a log transformation

Exercise 2

1) Write down a C++ program that generates 1000 random numbers in [0,1) using the rand() function. As a subordinate, use 1000 generation of the Excel Random() function.
2) Using all the techniques you can muster, analyze whether the sequence consists of IID U(0,1) RVs