GRUPO DE REAÇÕES NUCLEARES APLICAÇÕES & COMPUTAÇÃO

# Allometry and Geometric Programmation Part II

# Geometric Programmation as a natual Allometry formalism

**Geometric Programmation formalism.** Let´s minimize the expression:

$$\min g_0(x_1, x_2 \dots x_n) = \sum_{j=1}^{M_0} C_{0j} x_1^{a_{01j}} x_1^{a_{02j}} \dots x_n^{a_{0nj}}$$

under constraints : $\mathrm{g_k}(x_1, x_2 \dots x_n) = \sum_{i=1}^{M_0} C_{kj} x_1^{a_{k1j}} x_1^{a_{k2j}} \dots x_n^{a_{knj}} \leq 1$

$$C_{kj} \geq 0, \qquad x_i \geq 0$$

We call $g_0(x_1, x_2 \dots x_n)$ and $g_k(x_1, x_2 \dots x_n)$ *Posinomy*.

Ex.: min g=$\frac{400}{x_1 x_2 x_3} + 400 x_2 x_3 + 200 x_1 x_3 + 100 x_1 x_2$

Under the constraints $2x_1 x_3 + x_2 x_3 \leq 4 \ \ or \ \ \frac{x_1 x_3}{2} + \frac{x_2 x_3}{4} \leq 1$

REF.: https://web.stanford.edu/~boyd/papers/pdf/gp

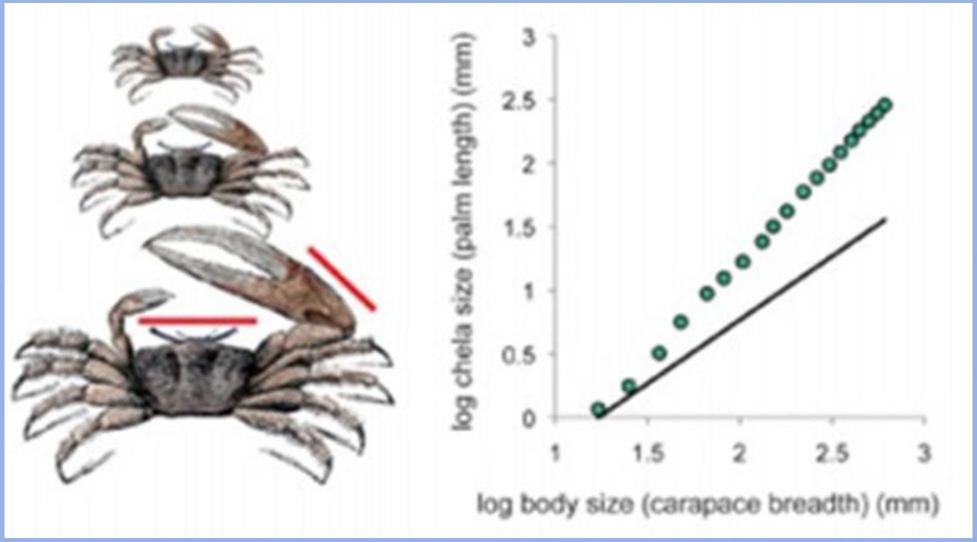
Métodos de Otimização - Antônio Galvão Novaes

Geometric Programming – Theory and Applications

R. J. Duffin, E. L. Peterson e C. Zener

**Allometry elements:**

Allometry is a branch of biology that deal with scale relationship of morphology, ecologic aspects of plants and animals in earth.




**Allometry´s fundamental equation:** $y = cx^a$

$y$ *is a mean of data gueting in a laboratory or field.*

F. Couvier, E. Dubois(1897), Huxley & Teissier(1935)
History of the Consept of Allometry, Jean Gayon
Amer. Zool., 40:748 – 758 (2000)

Perhaps the equation $y = \ cx^a$ allways are not so simple and more complex formalism is used but it depend of autor and subject.

$$\text{ex.:} Total\ Biomass = a_1 x^{b_1} + a_2 x^{b_2} + \cdots$$

The expression $y = \ cx^a$ is allwais corrected by a factor statistical $\epsilon_i$.

$$y = \ cx^a \epsilon_i$$

In general, the the factor $\epsilon_i$ is factorated:

$$Total\ Biomass = (a_1 x^{b_1} + a_2 x^{b_2} + \cdots)\epsilon_i$$

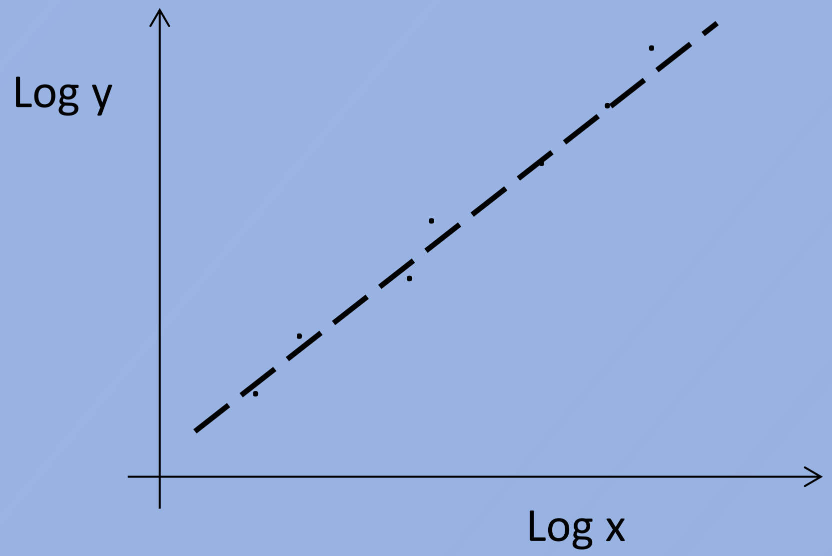
Ref:.Allometric equations for estimating aboveground biomass for common shrubs in northeastern California
S. Huff, Forest Ecology and Management 398 (2017) 48 - 63
Meassurement and Assessment Methods of Forest Aboveground Biomass:

The way to write Allometry in a Geometric Programming.

As example taking the equation:

$Total\ Biomass\ = (a_1 x^{b_1} + a_2 x^{b_2} + \cdots)\epsilon_i$. We must point out that $\epsilon_i = \left(a_1 x^{b_1} + a_2 x^{b_2} + \cdots\right) \leq 1$ that correspond just the form of constraints on Geometric Programation program of otimisation.

Things are no so simple: In allometry the $a_i$ and $b_i$ are estimated from a fit of Log x Log plot.




A Literature Review and the Challenge Ahead.
J. Návar – Biomass Book ISBN 978-953-307-113-8
pp22, september 2010

This procedure have a BIAS due the fact that the experimental data came from **aritmetic means** and we are extracting parameters from a **geometric mean**.

$$\sum_i \delta_i U_i \geq \prod_i U_i^{\delta_i}$$

if $\sum_i \delta_i = 1$

And the data obtained in the field are <u>not statistically independent</u> but MASS dependent.

The objective is to make the **arithmetic mean = geometric mean** and it is done by introducing a convenient multiplicative constant that depends on the process of collecting data in the field, how the terms $c$ and $a$ of the fundamental equation of allometry are treated and finally evaluated by computational fit processes where specific variance criteria are employed.

## Geometric programmation under no constraints´ machinery.

Taking : $\sum_i \boldsymbol{\delta_i U_i} \geq \prod_i \boldsymbol{U}_i^{\delta_i}$ is possibile express:

$$g(x_1, x_2, \dots, x_n) = \sum_{j=1}^{m} C_{0j} x_1^{a_{01j}} x_2^{a_{02j}} \dots x_n^{a_{0nj}}$$

Under the form, called primal function:

$$g(x_1, x_2 \dots x_n) = \sum_{j=1}^{M_0} u_1$$

Making $u_j = C_j x_1^{a_{1j}} x_2^{a_{2j}} \dots x_n^{a_{nj}}$ is possibile to write one $V$ function called dual,

$$V(\boldsymbol{\delta}, \boldsymbol{X}) = \left(\frac{C_1}{\delta_1}\right)^{\delta_1} \left(\frac{C}{\delta_2}\right)^{\delta_2} \left(\frac{C_3}{\delta_3}\right)^{\delta_3} \dots \left(\frac{C_1}{\delta_n}\right)^{\delta_m} x_1^{D_1} x_2^{D_2} \dots x_n^{D_n}$$

As $D_i = \sum_{J=1}^{M} \delta_j a_{ij} = 0$ and remenbering that $\sum_i \delta_i = 1 \Rightarrow \delta_1 + \delta_2 + \delta_3 + \delta_4 = 1$

One can write one $v$ function, called pre-dual, as:

$$v(\boldsymbol{\delta}) = \left(\frac{C_1}{\delta_1}\right)^{\delta_1} \left(\frac{C}{\delta_2}\right)^{\delta_2} \left(\frac{C_3}{\delta_3}\right)^{\delta_3} \dots \left(\frac{C_1}{\delta_n}\right)^{\delta_m}$$

**Duffin et. al. showed that $v(\delta)$ maximum is equal to $g(\delta)$ minimum.**

$$\sum_i \boldsymbol{\delta_i U_i} = \prod_i \boldsymbol{U}_i^{\delta_i}$$

Taking a pratical example:

$$g = \frac{400}{x_1x_2x_3} + 400x_2x_3 + 200x_1x_3 + 100x_1x_2$$

$$\boldsymbol{v(\delta)} = \left(\frac{400}{\delta_1}\right)^{\delta_1}\left(\frac{400}{\delta_2}\right)^{\delta_2}\left(\frac{200}{\delta_3}\right)^{\delta_3}\left(\frac{100}{\delta_4}\right)^{\delta_4}$$

If $D_i = \sum_{J=1}^{M} \delta_j a_{ij} = 0$

$$D_1 = -\delta_1 + \quad + \delta_3 + \delta_4 = 0$$
$$D_1 = -\delta_1 + \delta_2 + \quad + \delta_4 = 0$$
$$D_1 = -\delta_1 + \delta_2 + \delta_3 \quad = 0$$

and : $\delta_1 + \delta_2 + \delta_3 + \delta_4 = 1$

Finaly:

$$\delta_1 = 2\backslash 5 \quad \delta_2 = 1\backslash 5 \quad \delta_3 = 1\backslash 5 \quad \delta_4 = 1\backslash 5$$

$$V = (\frac{400}{\delta_1})^{\delta_1}(\frac{400}{\delta_2})^{\delta_2}(\frac{200}{\delta_3})^{\delta_3}(\frac{100}{\delta_4})^{\delta_4} = 1000$$

Under constraints case the mathematical machinery are more complex perhaps we get $\delta_i$ , $a_1$ and $g_0$ but only one $C_{0j}$.

We can picture out that Geometric Programmation is allometry´s formal language due the fact that:

***Nature always optimaze food resources, locomotion form, space to survive and etc. etc. etc. ...***

So, we can approach the allometry´s problem considering that the expression:

$$g_i(x_1, x_2 \dots x_n) = \sum_{j=1}^{M_0} C_{0j} x_1^{a_{01j}} x_1^{a_{02j}} \dots x_n^{a_{0nj}}$$

by current processes and applying the machinery of Geometrical Programming we can obtain the best of $\delta_i$ , $x_i$ and $g_i$ in this way, the condition is always guaranteed.

$$\sum_{\boldsymbol{i}} \boldsymbol{\delta_i U_i} = \prod_{\boldsymbol{i}} \boldsymbol{U}_i^{\delta_i}$$

Each field measure corresponds to an expression to a $g_i(x_1, x_2 \dots x_n)$ but there is not for values other than $x_i$, the value of $C_{0j}$ does not change and so we have a blurring of our objective.

This difficulty can be better circumvented if we apply the constraints:

$$g_k(x_1, x_2 \dots x_n) = \sum_{i=1}^{M_0} C_{kj} x_1^{a_{k1j}} x_1^{a_{k2j}} \dots x_n^{a_{knj}} \leq 1$$

In this case, we notice that when we collect the experimental data in a classic Log x Log plot of $y = cx^a$ we have approximate values for the **c**'s and **a**'s, suffering from the problems of inequality between arithmetic and geometric mean. But, when we apply the procedures of Geometric Programming with constraints, we have other $\delta_i$ and $a_i$ and $x_i$ correct and $g_0$ which in this case will be the minimum value with constraints. But in the equality between the means arithmetic and geometric and the $C_{0j}$ can be equal for each sample case.

In a pratical example:

$$\min g_0(\mathbf{X}) = \frac{400}{x_1 x_2 x_3} + 400 x_2 x_3$$

$$\text{under constraints: } g_1(\mathbf{X}) = \left(\frac{x_1 x_3}{2}\right) + \left(\frac{x_1 x_2}{4}\right) \leq 1$$

Making use of Geometric Programmation machinery with constraints one can get $\Delta_i$(equivalent to $\delta_i$ without constraints case) and after some mathematical manipulations we have.

$$\Delta_1 = 2\backslash 3 \quad \Delta_2 = 1\backslash 3 \quad \Delta_3 = 1\backslash 3 \quad \Delta_4 = 1\backslash 3$$

Without constraints was:

$$\delta_1 = 2\backslash 5 \quad \delta_2 = 1\backslash 5 \quad \delta_3 = 1\backslash 5 \quad \delta_4 = 1\backslash 5$$

The valures to $x_i$ as well as the valure of $V$ that was 1000 and now is 600.

Do not forgetting that each data sempled are never equal to another, we have that what defines the sparse points in a Log x Log plot of $y = cx^a$ will be the $\delta_i$, $x_i$ correct for that case in the same $C_{0j}$. But in this case, we have different restrictions for each sample collected and the set of $C_{0k}$ is not defined by the y-intercept in a plot but is a specific point of each argument studied.

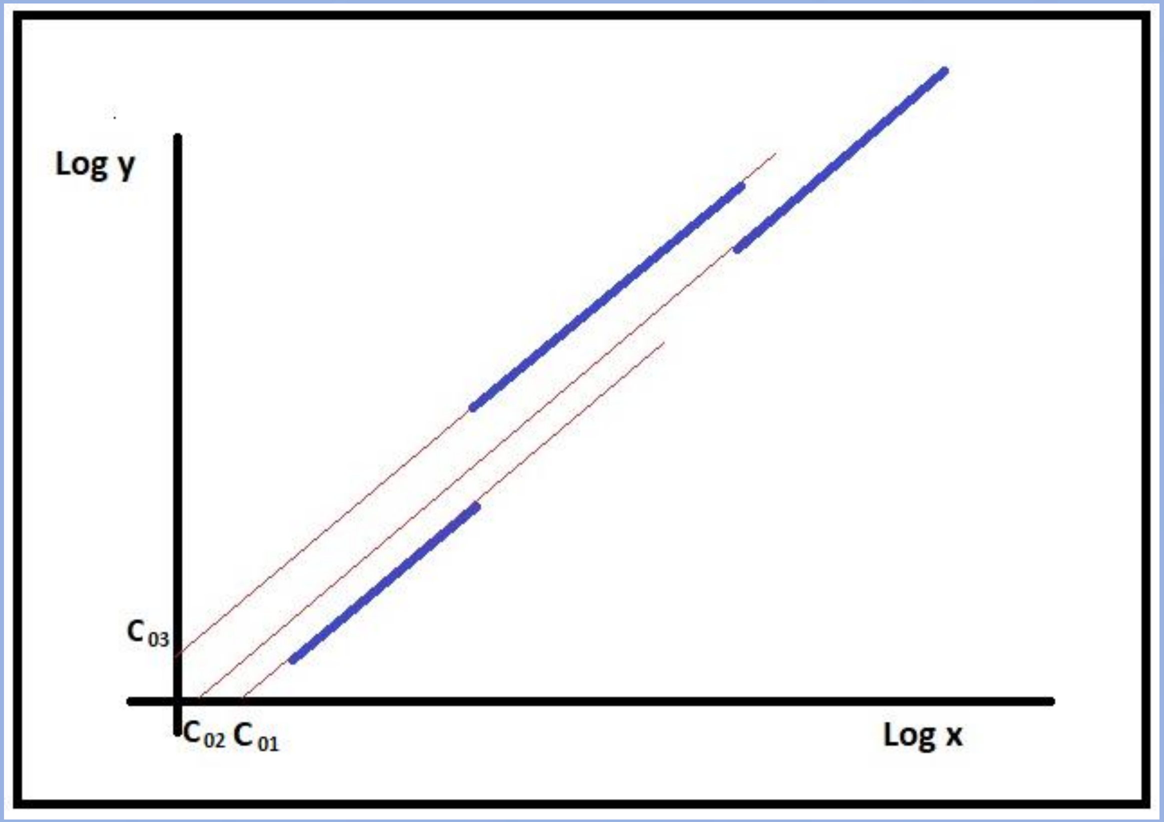
**The ordinate is linked to the restrictions.**

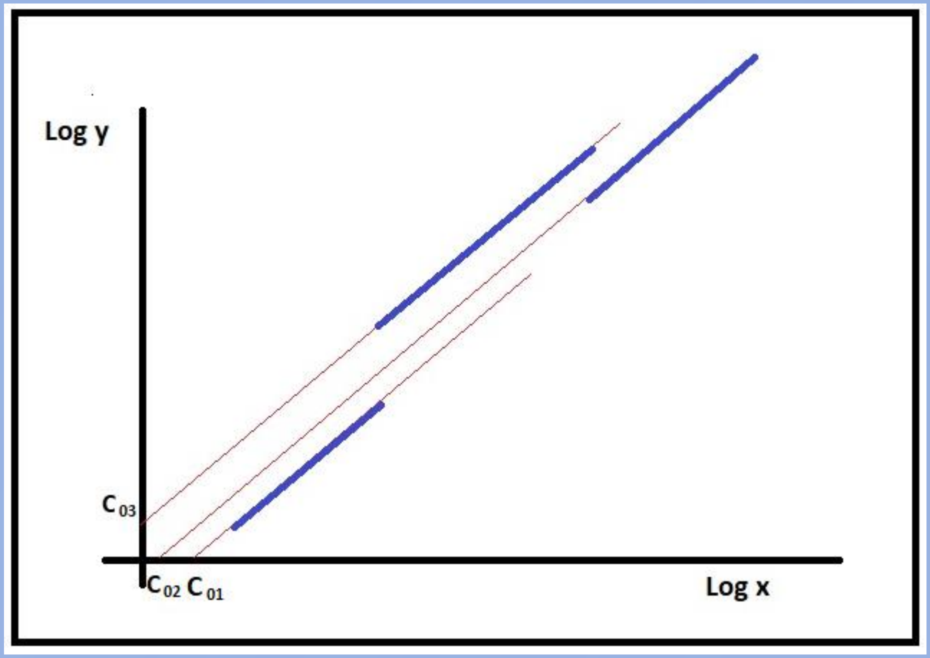
This interpretation is coherent because it is the restrictions of each specimen that will define the characteristics of each specimen, its size, basal metabolism or any other considered variable. But we also conclude that in nature, by observation, the values of $C_{0k}$ are grouped in a mean line on a Log x Log plot as a function of the ordinate $x$ resulting in an abscissa $y$ that when we apply due regression methods we obtain a ***c*** and one ***a*** that it looks like it can be described in the form $y = cx^a$.

One must to think $C_{0j}$ **general constant** as a **pivot** as function of subject of researche and the constraints as mathematical form:

$$g_k(x_1, x_2 \dots x_n) = \sum_{i=1}^{M_0} C_{kj} x_1^{a_{k1j}} x_1^{a_{k2j}} \dots x_n^{a_{knj}} \le 1$$

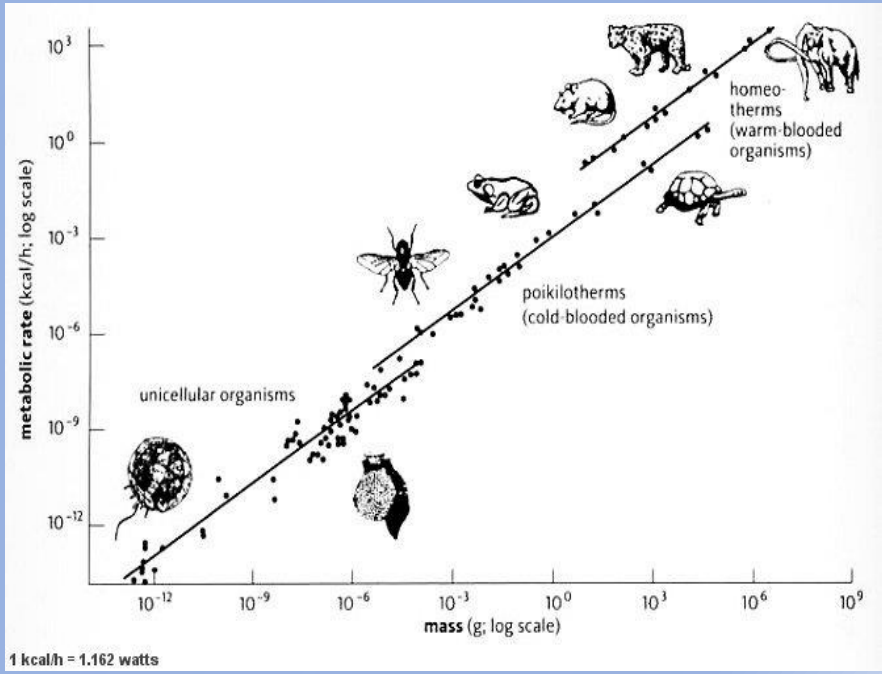
can be suggested *ad hoc* or formulated as experimental demands.

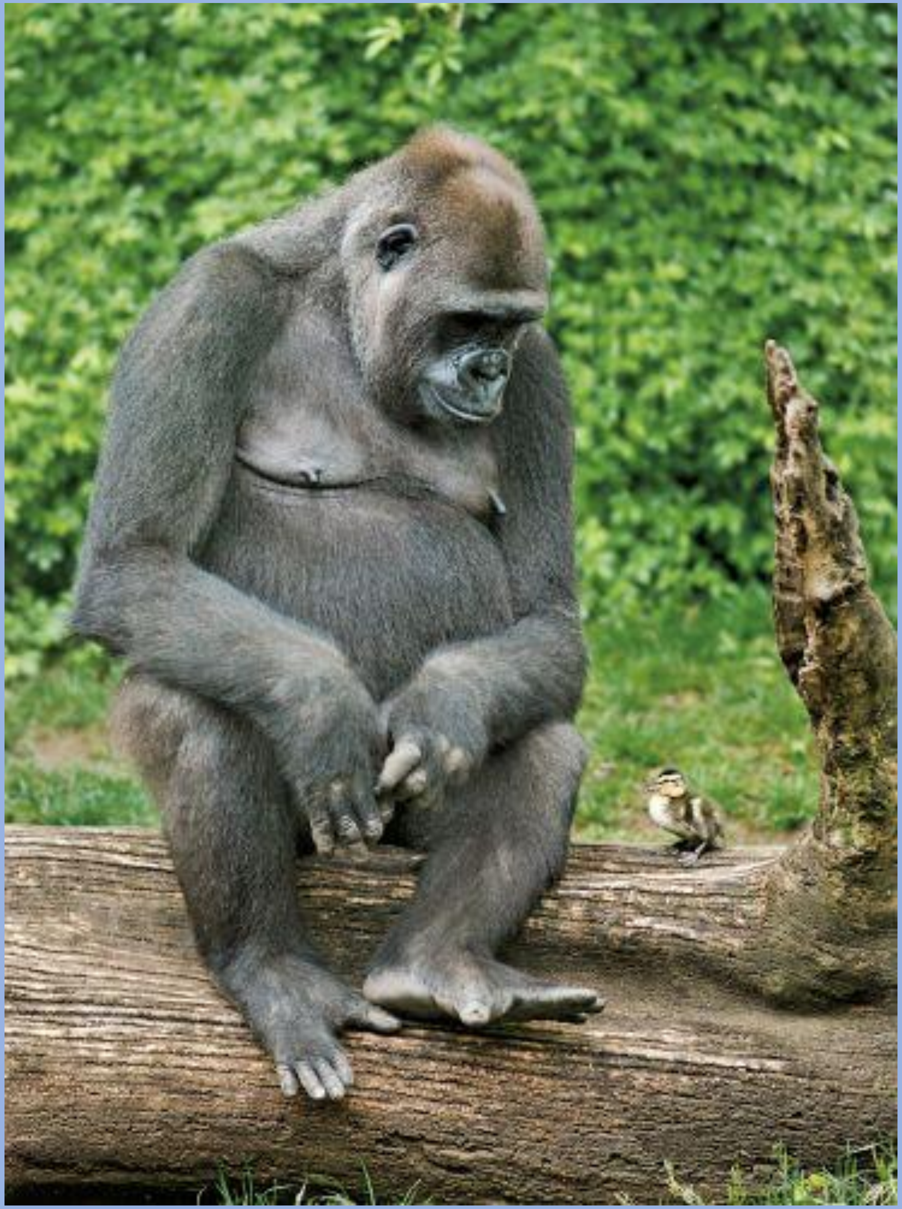





A plot Log x Log under our proposal.

A tipical experimental plot Log x Log.





Sebastião Simionatto - 2021