

COMPARACIÓN ENTRE MÉTODOS DE INTERPOLACIÓN ESPACIAL GEOESTADÍSTICOS PARA ESTIMACIÓN DE RECURSOS CON USO DE PYTHON

MSc. Heber Hernández Guerra



NUBE MINERA
PLATAFORMA E LEARNING & CONSULTORIA



Agenda:

- 1 Presentación caso Cu/Au
- 2 Estimación vs co-estimación geoestadística
- 3 Resultados
- 4 Beneficios y oportunidades del uso de Python en Jupyter Notebook

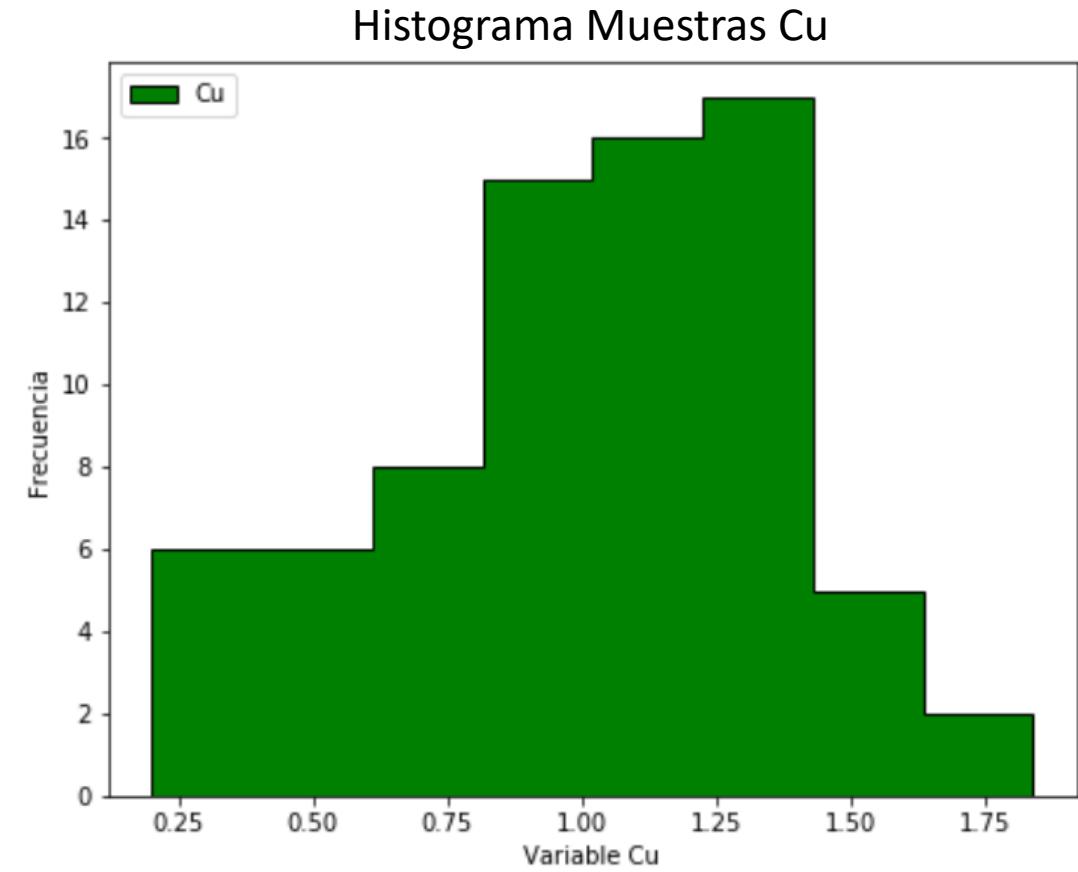
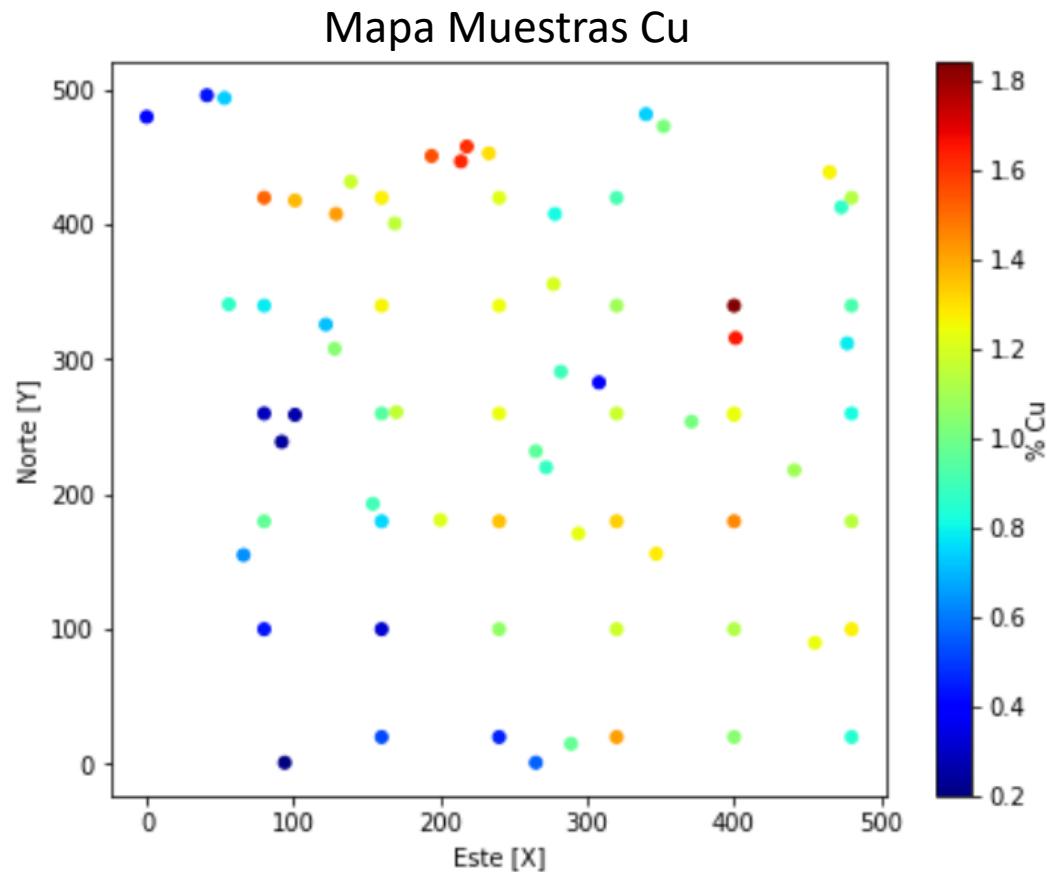
Caso: Muestras puntuales Cu y Au 2D

$\text{Cu} = Z_1$
 $\text{Au} = Z_2$

El muestreo de Z_1 se produjo años posteriores al de Z_2
(exploración superficial)



Muestras puntuales: Variable Principal Z1



n: 75

\bar{x} : 1.01 %Cu

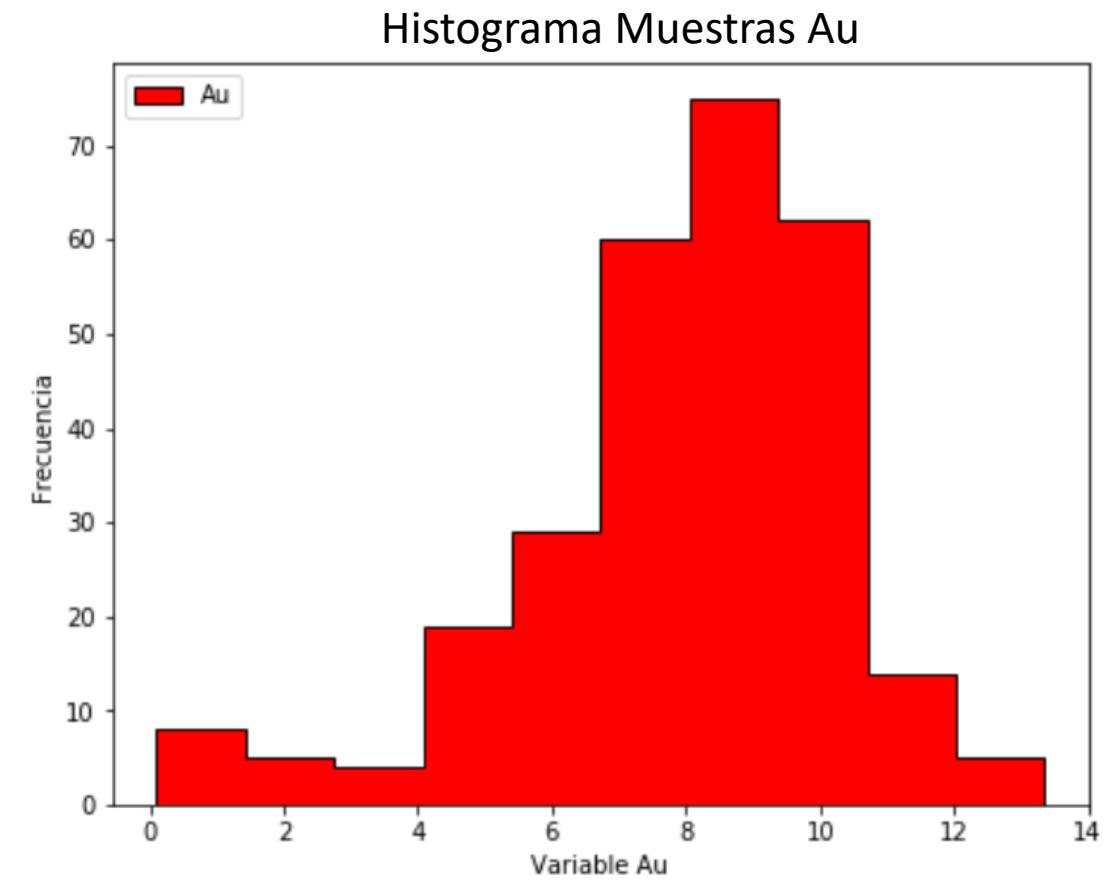
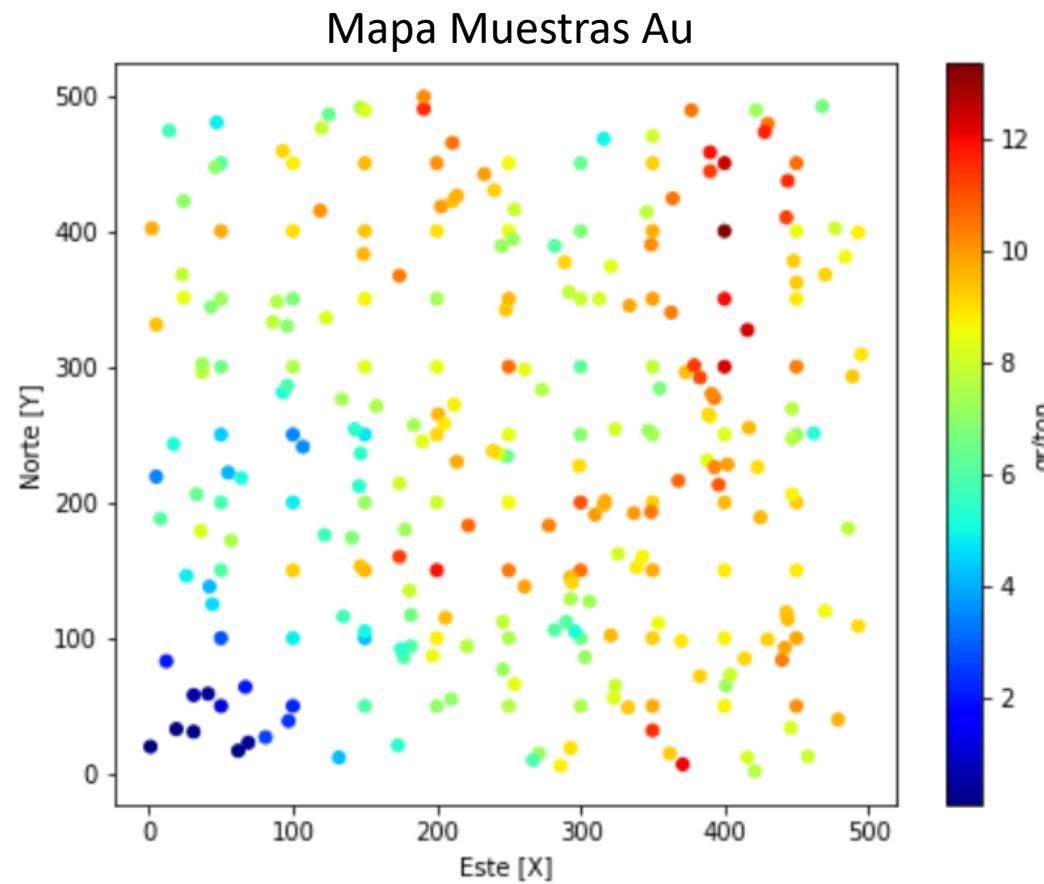
s^2 : 0.14 %Cu²

s: 0.37 %Cu

g1: -0.38

g2: -0.26

Muestras puntuales: Variable Auxiliar Z2



n: 281

\bar{x} : 7.97 g/tAu

s^2 : 5.8 g/tAu²

s: 2.4 g/tAu

g1: -1.04

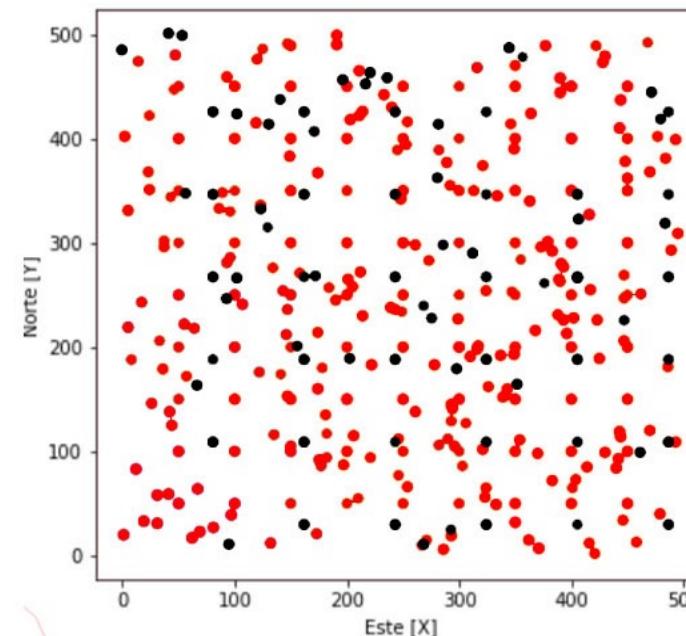
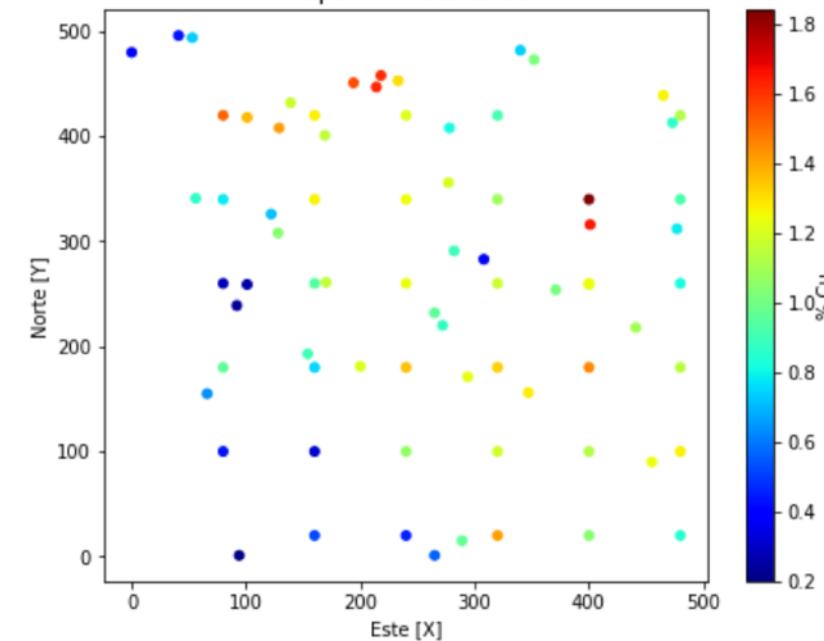
g2: 1.59

Muestras puntuales: Relación Z1 y Z2

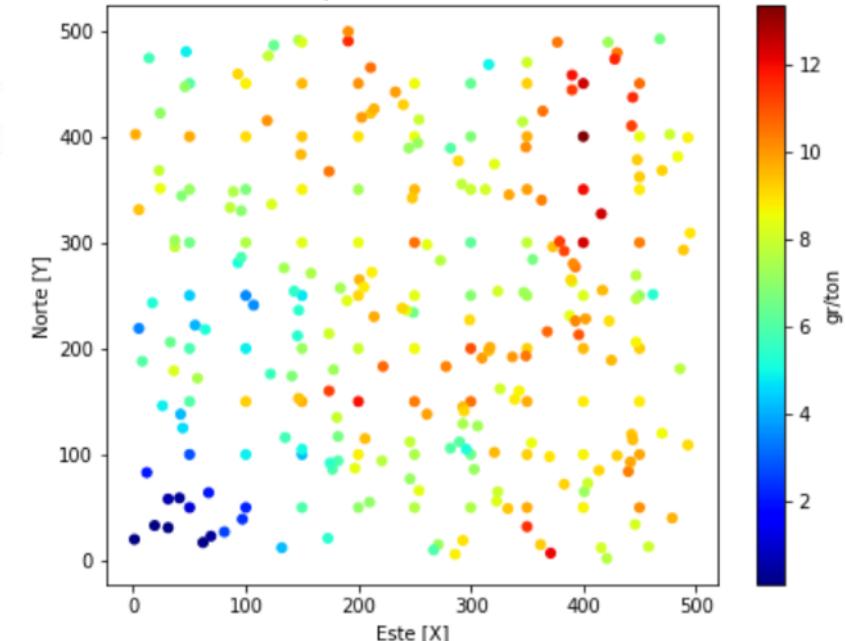
Ubicación X, Y de Z1 no coincidente con Z2

Esto deriva en la imposibilidad de CoKriging Completo
como método de co-estimación al no poder obtener el SV cruzado

Mapa de muestras Cu

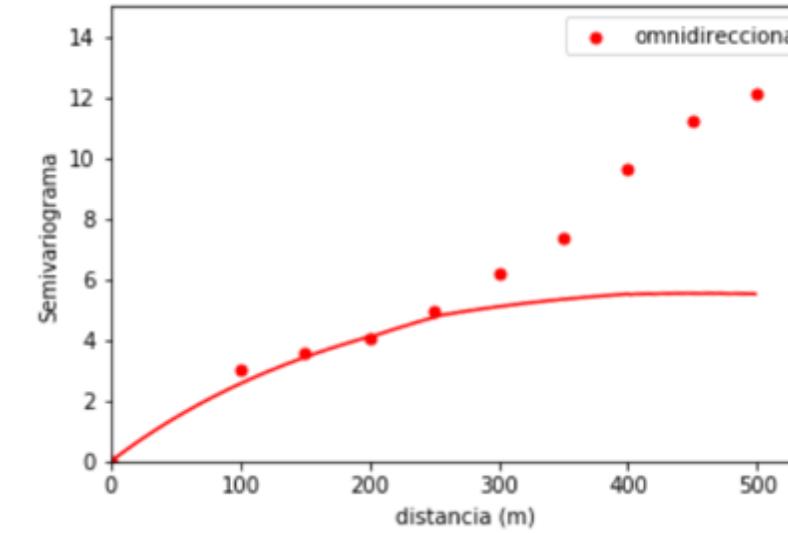
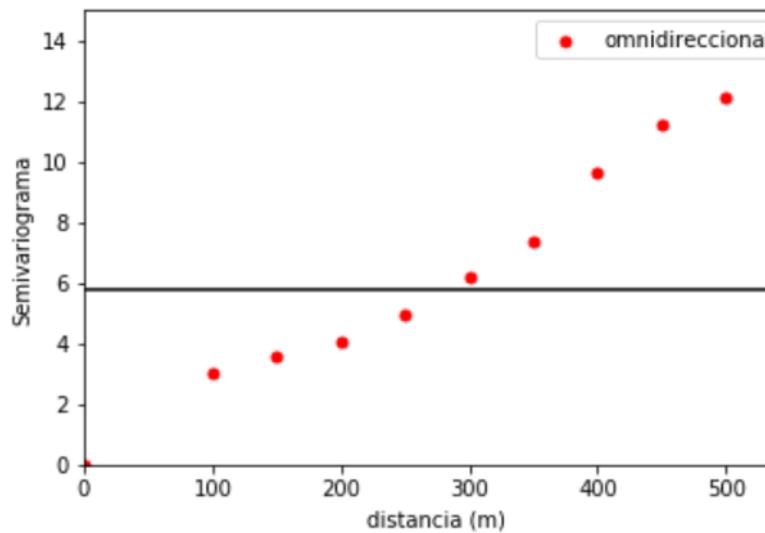
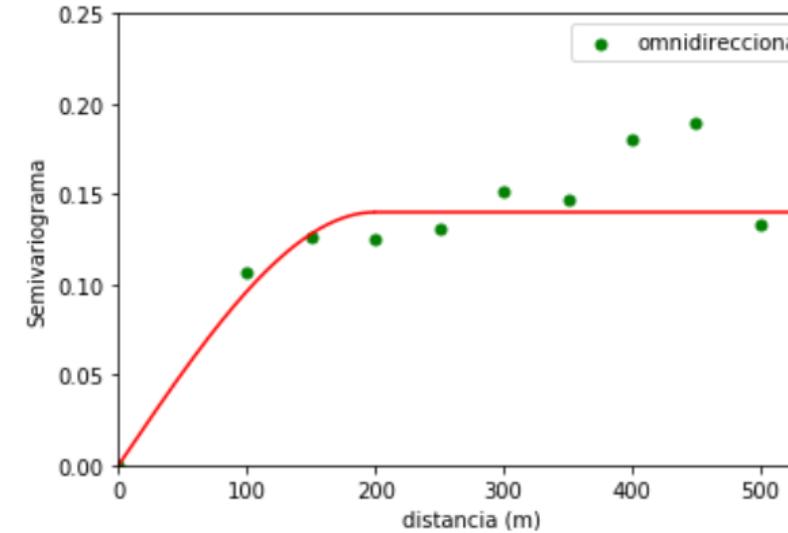
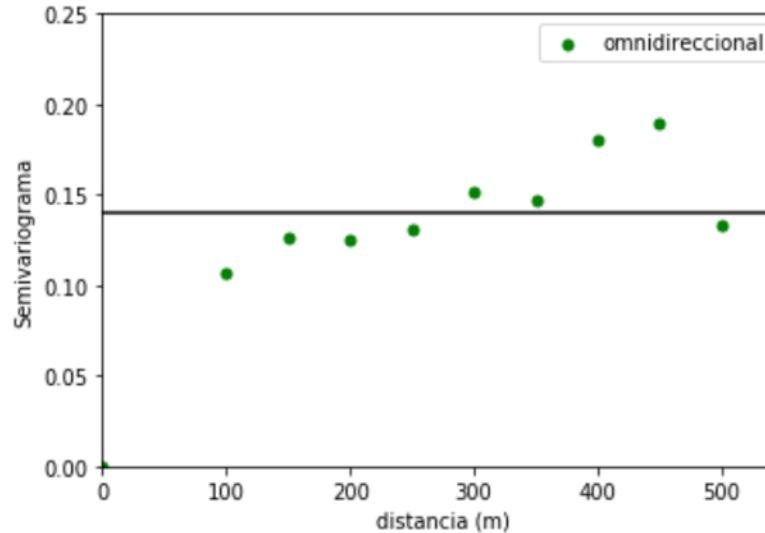


Mapa de muestras Au



$$\gamma_{ij}^*(h) = \frac{1}{2|N(h)|} \sum_{N(h)} [Z_i(xa) - Z_i(xb)] [Z_j(xa) - Z_j(xb)] \quad \text{Donde } N(h) = \{\text{Número de pares } (a, b) \text{ tales que } xa - xb = h\}$$

Muestras puntuales: Modelo semivariográfico Z1 y Z2



$$\gamma(h) = 0.14 \text{ esf (200,200)}$$

$$a = 200 \text{ m}$$

$$C = 0.14$$

$$C_0 = 0$$

$$\gamma(h) = 5.8 \text{ exp (250,250)}$$

$$a = 250 \text{ m}$$

$$C = 5.8$$

$$C_0 = 0$$

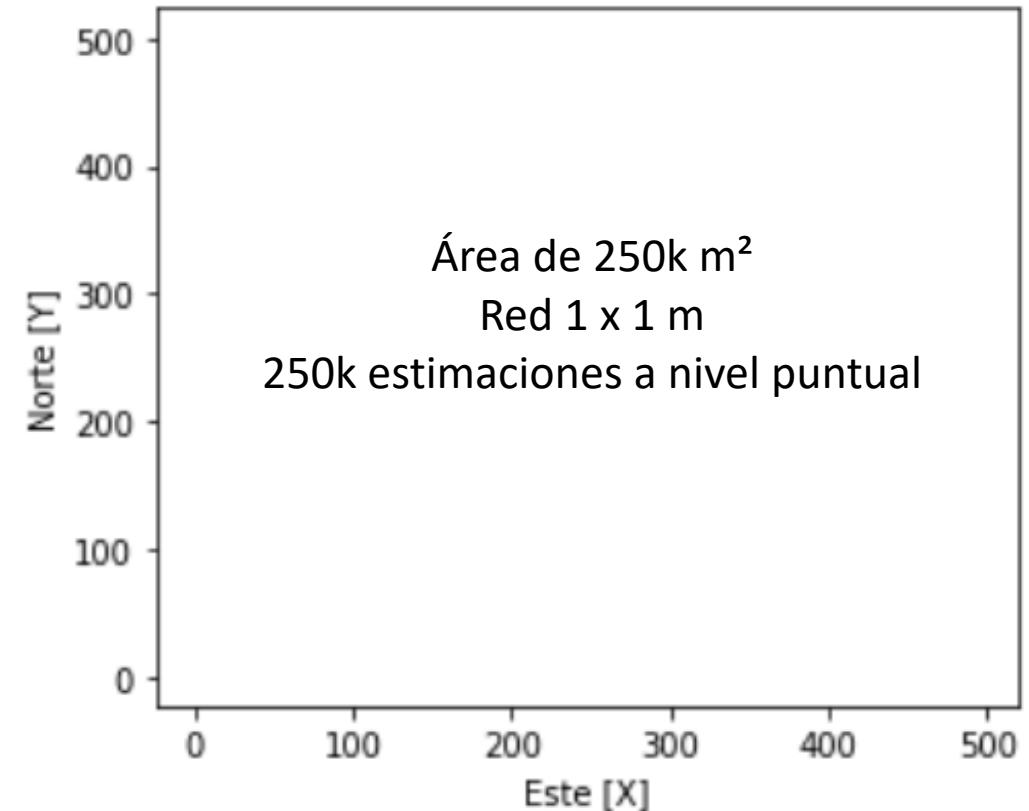
$$\gamma^*(h) = \frac{1}{2N(h)} \sum_{i=1}^{N(h)} (Z(x_i) - Z(x_i + h))^2$$

$$\gamma(h) = C \left[\frac{3}{2} \frac{h}{a} - \frac{1}{2} \frac{h^3}{a^3} \right] \quad \text{Si } h \leq a$$

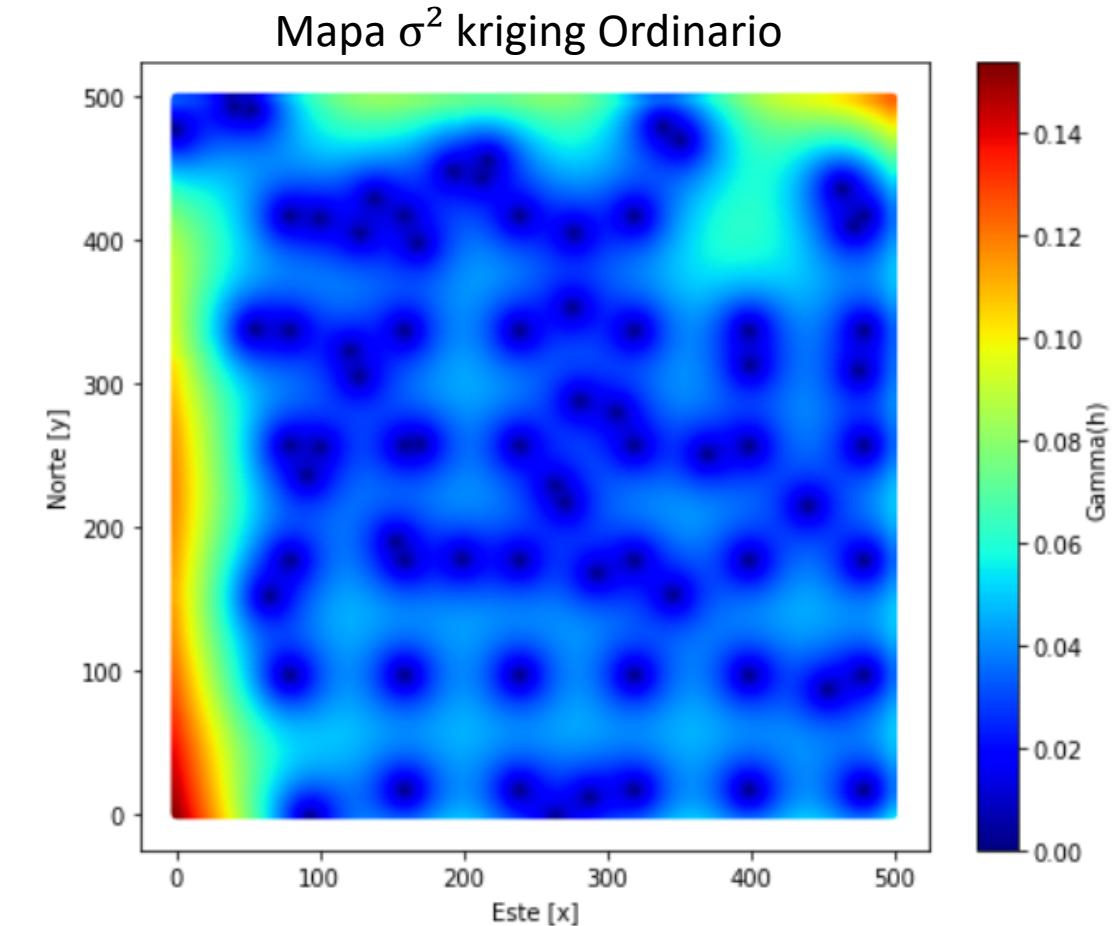
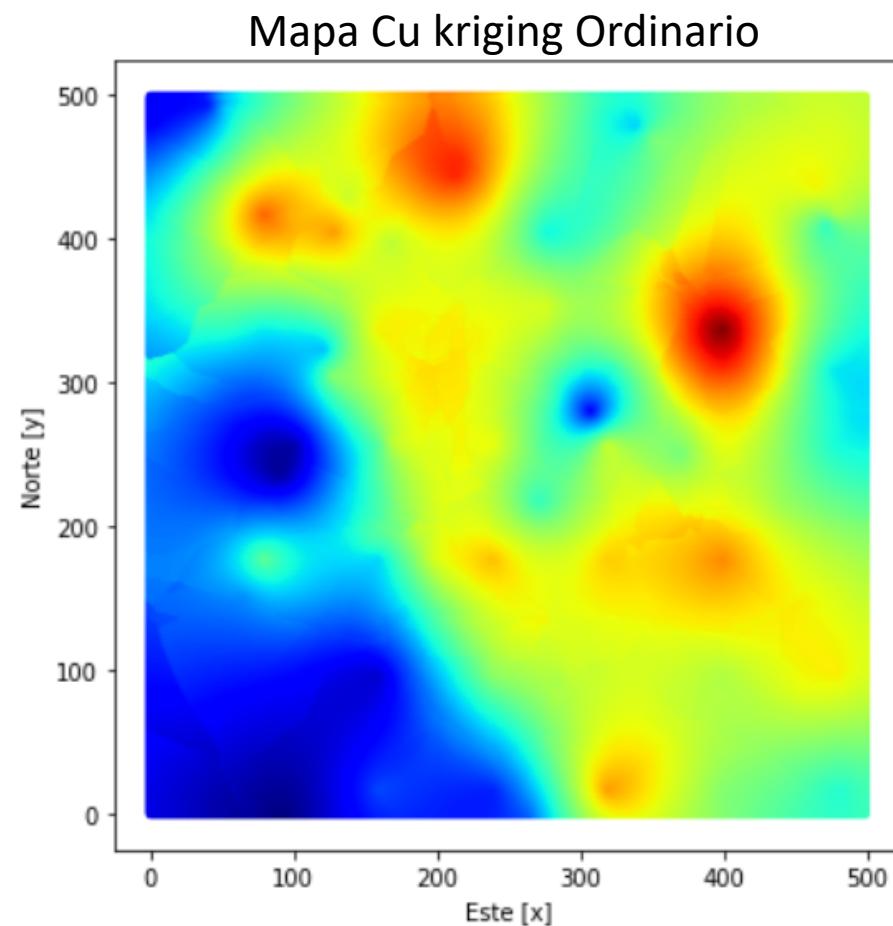
$$\gamma(h) = C \left[1 - \exp \left[-\frac{h}{a} \right] \right]$$

Métodos geoestadísticos de interpolación a emplear:

1. Kriging Ordinario
2. Kriging Simple
3. Co-Kriging Simple Co localizado con modelo de Markov I
4. Simulación Secuencial Gaussiana
5. Co-Simulación Secuencial Gaussiana



Kriging Ordinario:



$$Z^*(x_0) = \sum_{i=1}^N Z(x_i) \cdot \lambda_i$$

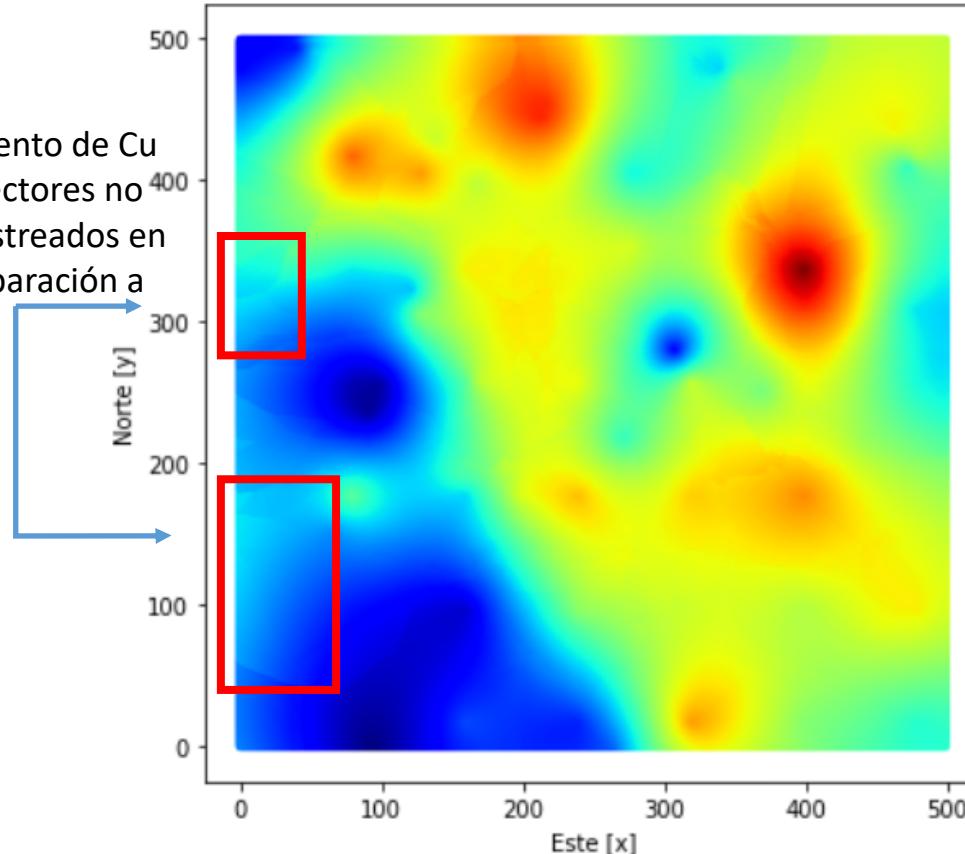
$$\sigma_{Ko}^2(x_0) = \sum_{i=1}^n \lambda_i \cdot \gamma(x_i - x_0) - \mu$$

$$\begin{pmatrix} \gamma(x_1 - x_1) & \cdots & \gamma(x_1 - x_n) & 1 \\ \vdots & \ddots & \vdots & \vdots \\ \gamma(x_n - x_1) & \cdots & \gamma(x_n - x_n) & 1 \\ 1 & \cdots & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \\ -\mu \end{pmatrix} = \begin{pmatrix} \gamma(x_1 - x_0) \\ \vdots \\ \gamma(x_n - x_0) \\ 1 \end{pmatrix}$$

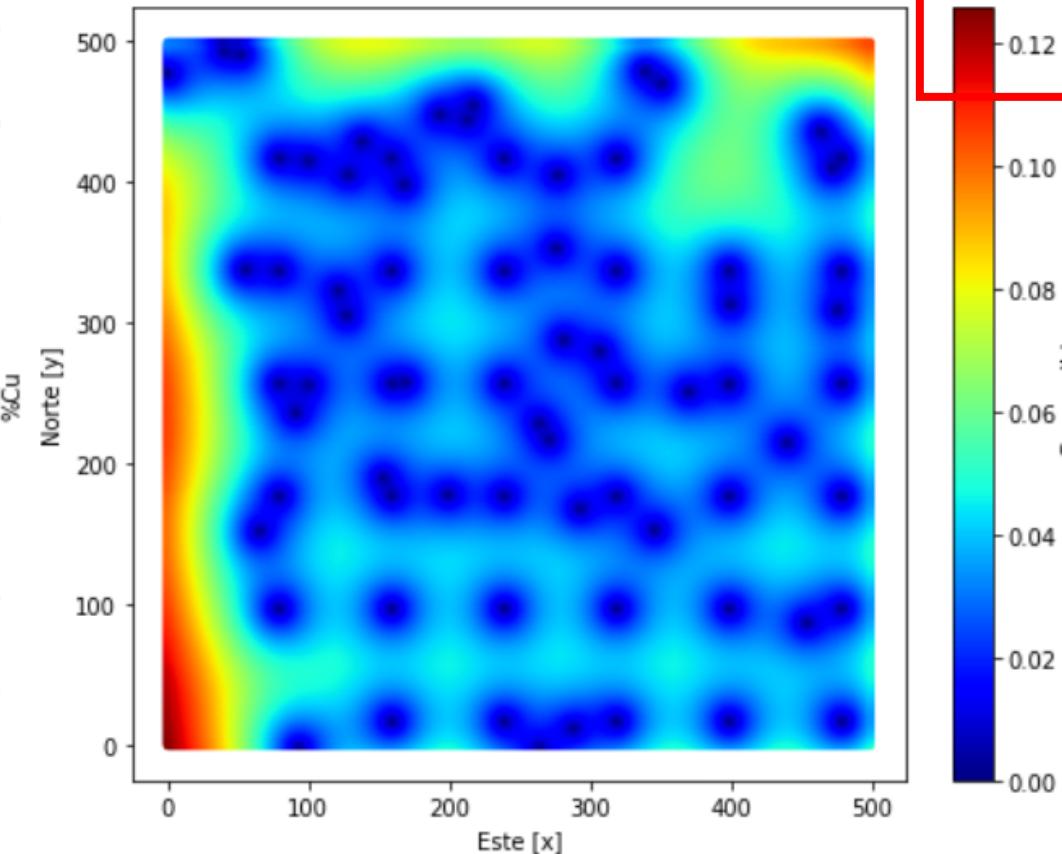
Kriging Simple:

Mapa Cu kriging Simple

Aumento de Cu en sectores no muestreados en comparación a KO



Mapa σ^2 kriging Simple



Disminuye σ^2_{ks} en comparación a σ^2_{ko}

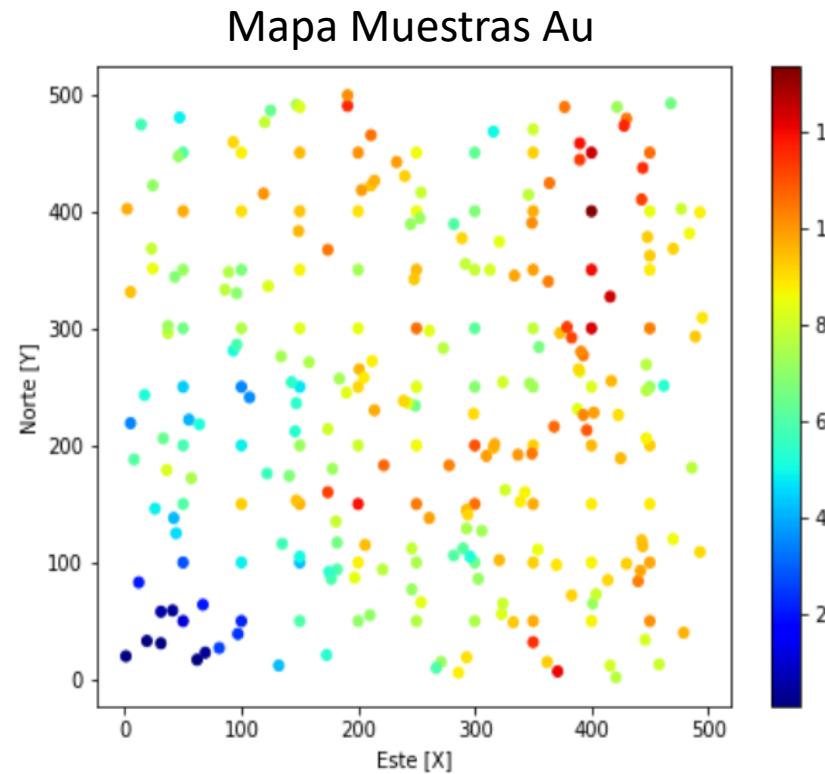
$$Z^*(x_0) = \sum_{i=1}^N Z(x_i) \cdot \lambda_i + \{1 - \sum_{i=1}^N \lambda_i\} \cdot m$$

$$\sigma_{KS}^2(x_0) = \sigma^2 - \sum_{i=1}^n \lambda_i \cdot C(x_i - x_0)$$

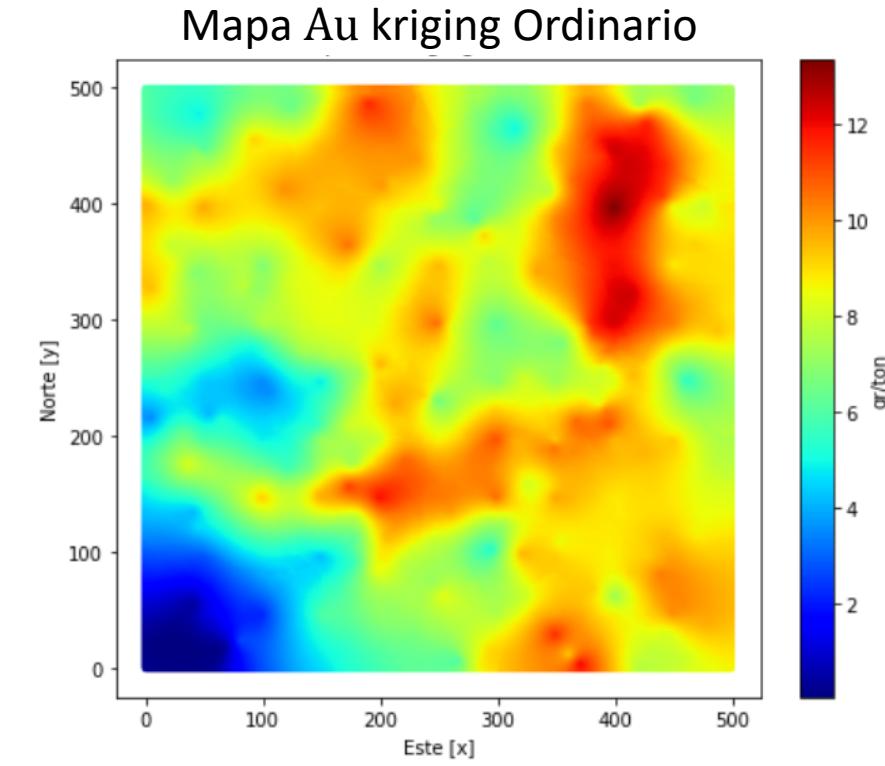
$$\begin{pmatrix} C(x_1 - x_1) & \cdots & C(x_1 - x_n) \\ \vdots & \ddots & \vdots \\ C(x_n - x_1) & \cdots & C(x_n - x_n) \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix} = \begin{pmatrix} C(x_1 - x) \\ \vdots \\ C(x_n - x) \end{pmatrix}$$

CoKriging Simple Co localizado Markov I:

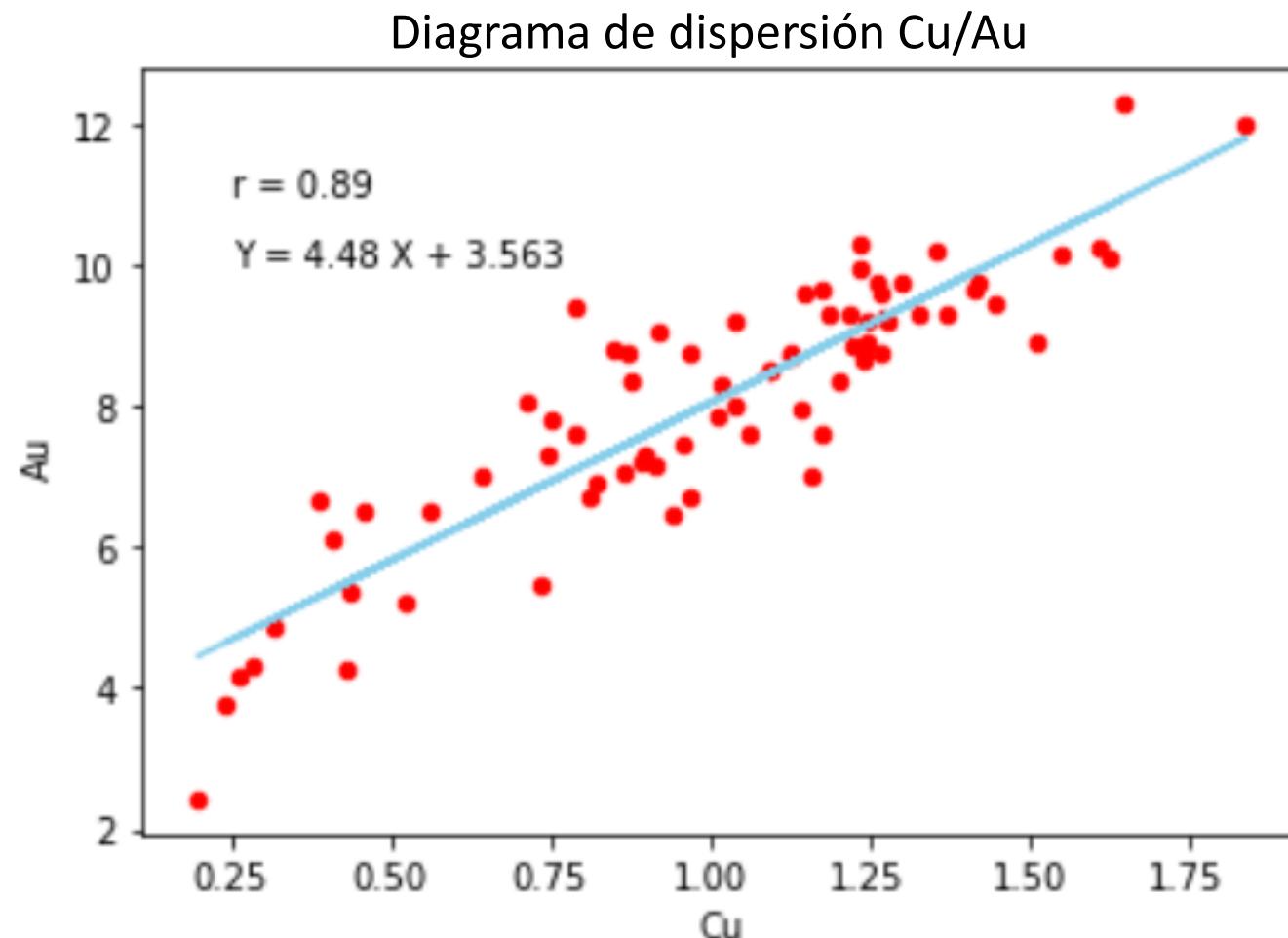
Z1: Cu; Z2: Au (mayormente muestreada que Z1 pero no exhaustiva)



Alternativa para Z2
Interpolación
univariada



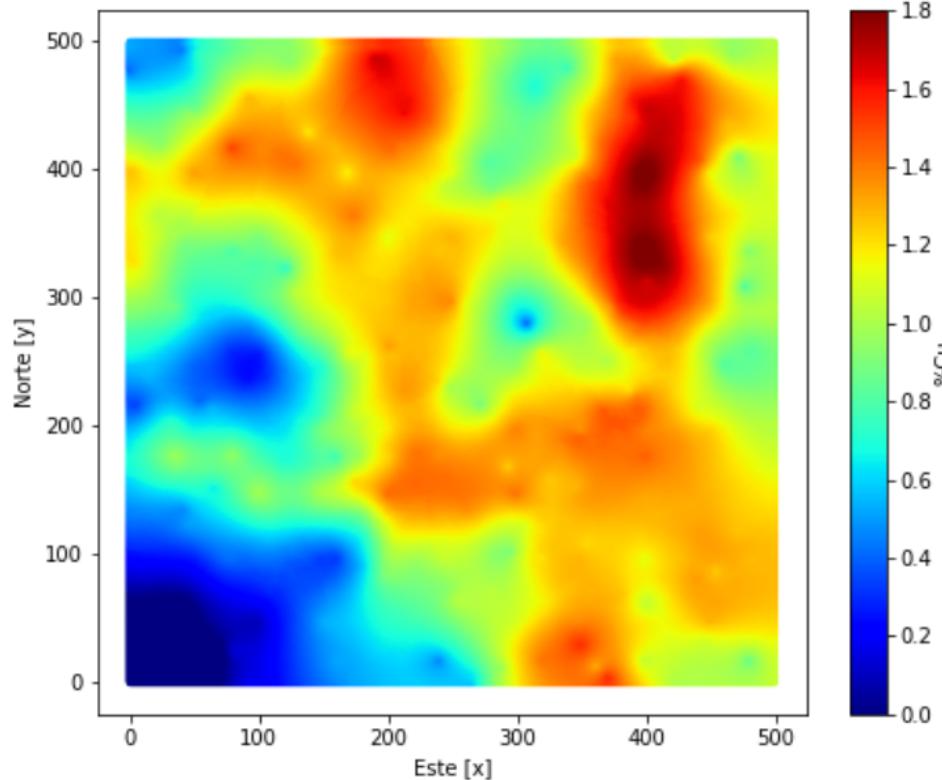
Correlación Z1 y Z2 soft



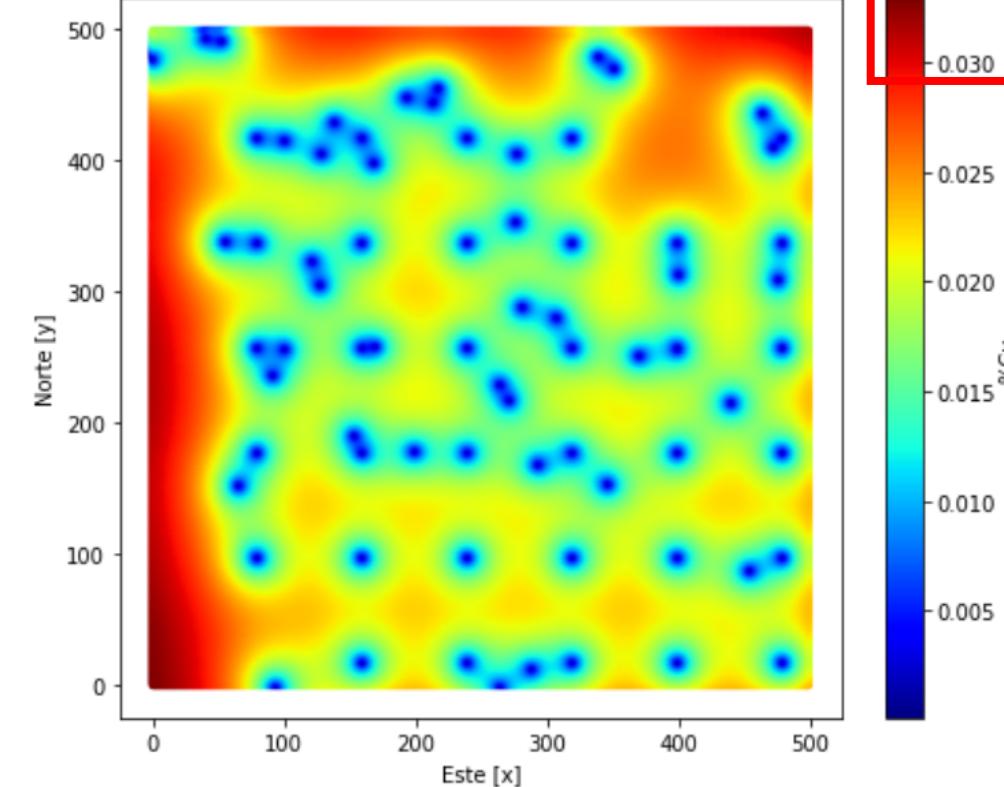
$s^2(Z1) < s^2(Z2)$; **Modelo de Markov I**
 $s^2:(0.14) < s^2:(5.8)$

CoKriging Simple Co localizado Markov I:

Mapa Cu Cokriging Colocalizado Markov I



Mapa σ^2 Cokriging Colocado Markov I

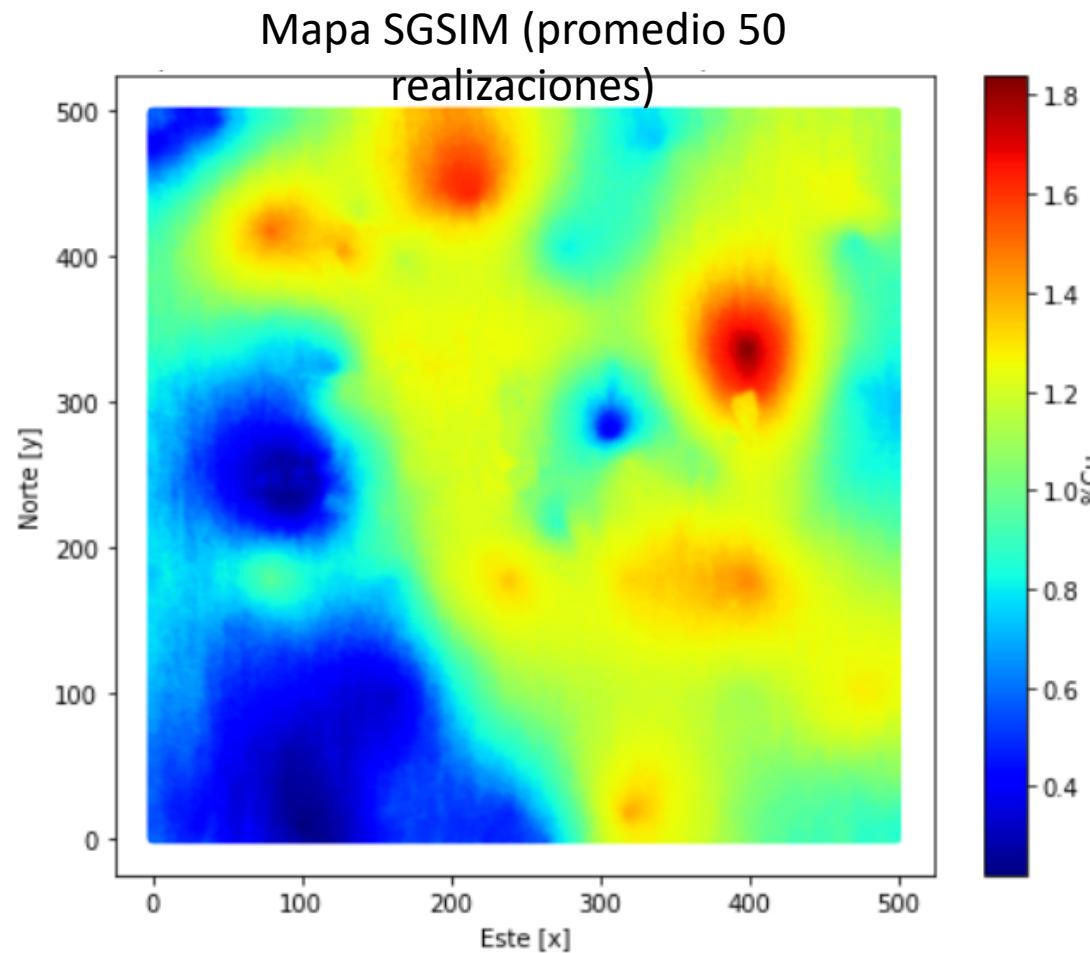


Disminuye σ^2_{ck} en comparación al Kriging

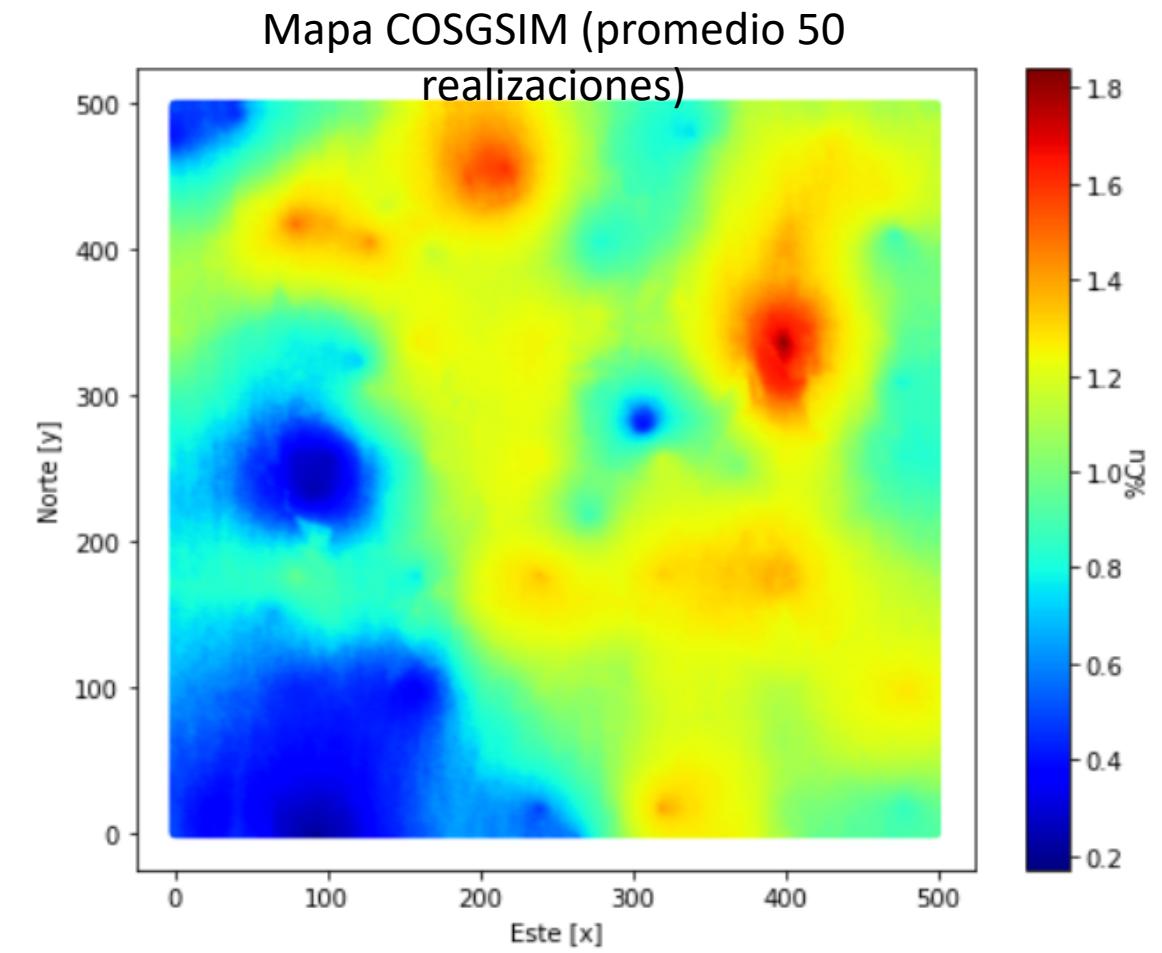
$$Y^*(u) = \lambda_\alpha [Y(u_\alpha) - m_Y] + v[B(u) - m_B]$$

$$\begin{cases} \sum_{\beta=1}^b \lambda_\beta C_Y(u_\alpha - u_\beta) + v C_{YB}(u_\alpha - u), & \alpha = 1, \dots, n \\ \sum_\beta \lambda_\beta C_{YB}(u - u_\beta) + v C_B(0) \end{cases}$$

Simulación & Co Simulación Secuencial Gaussiana:

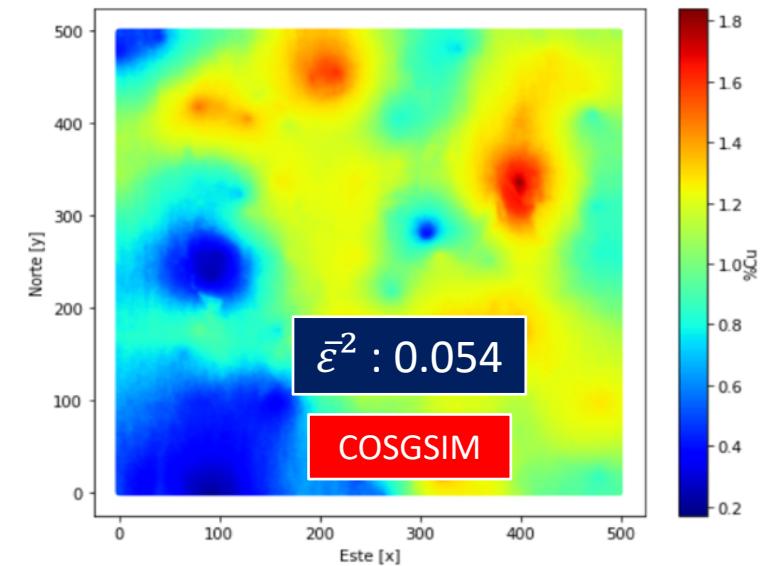
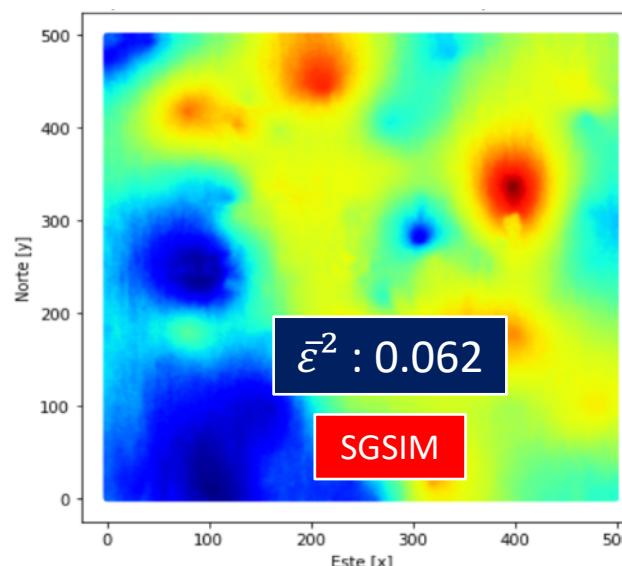
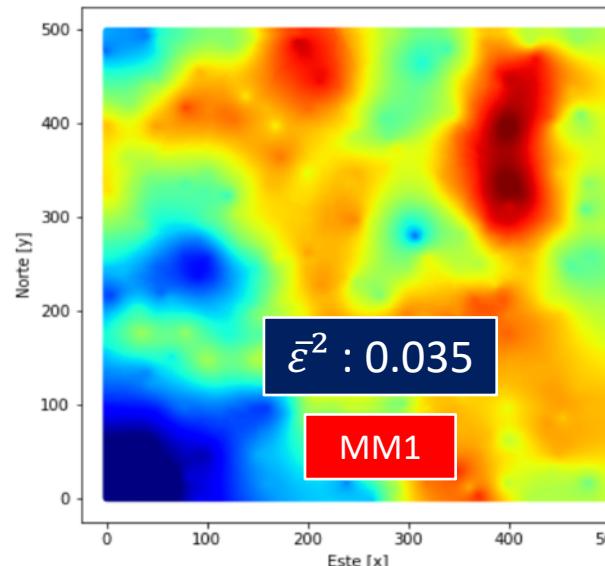
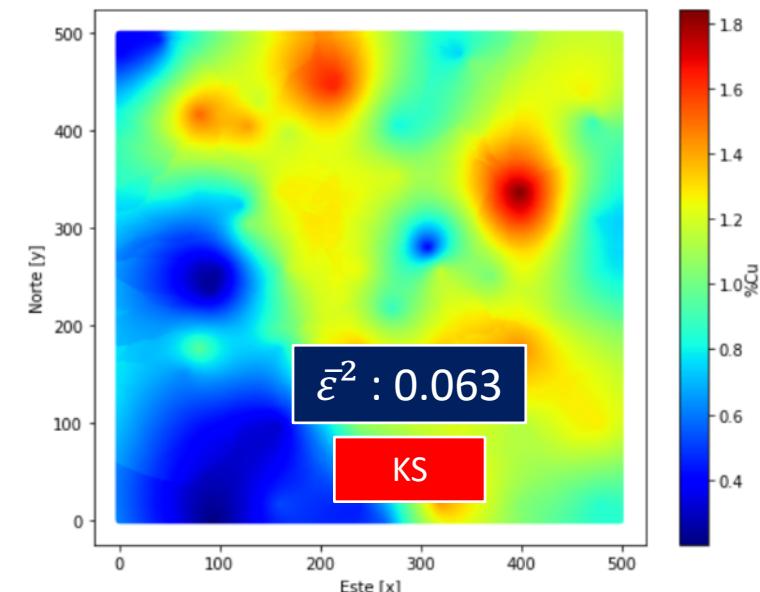
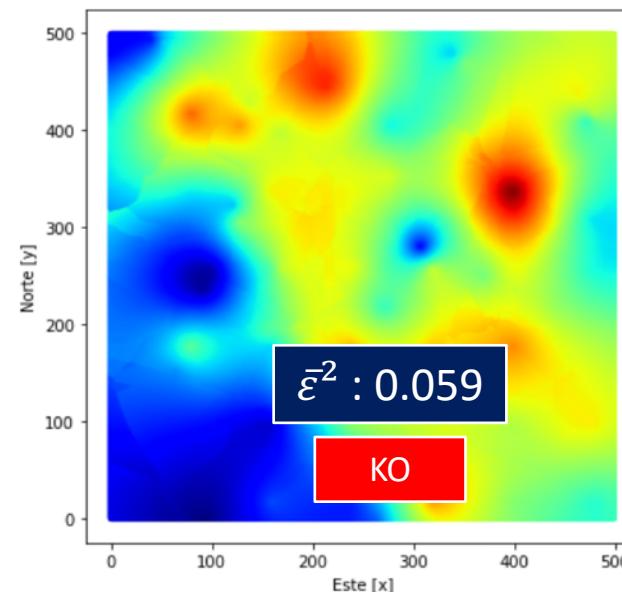
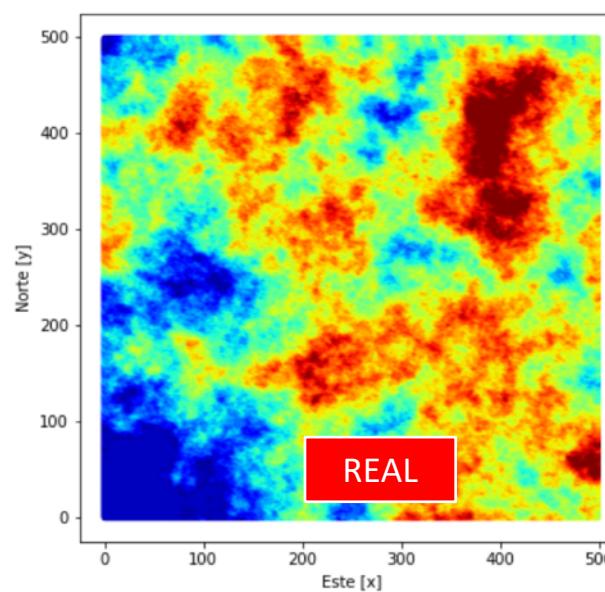


SGSIM: Sequential Gaussian
Simulation

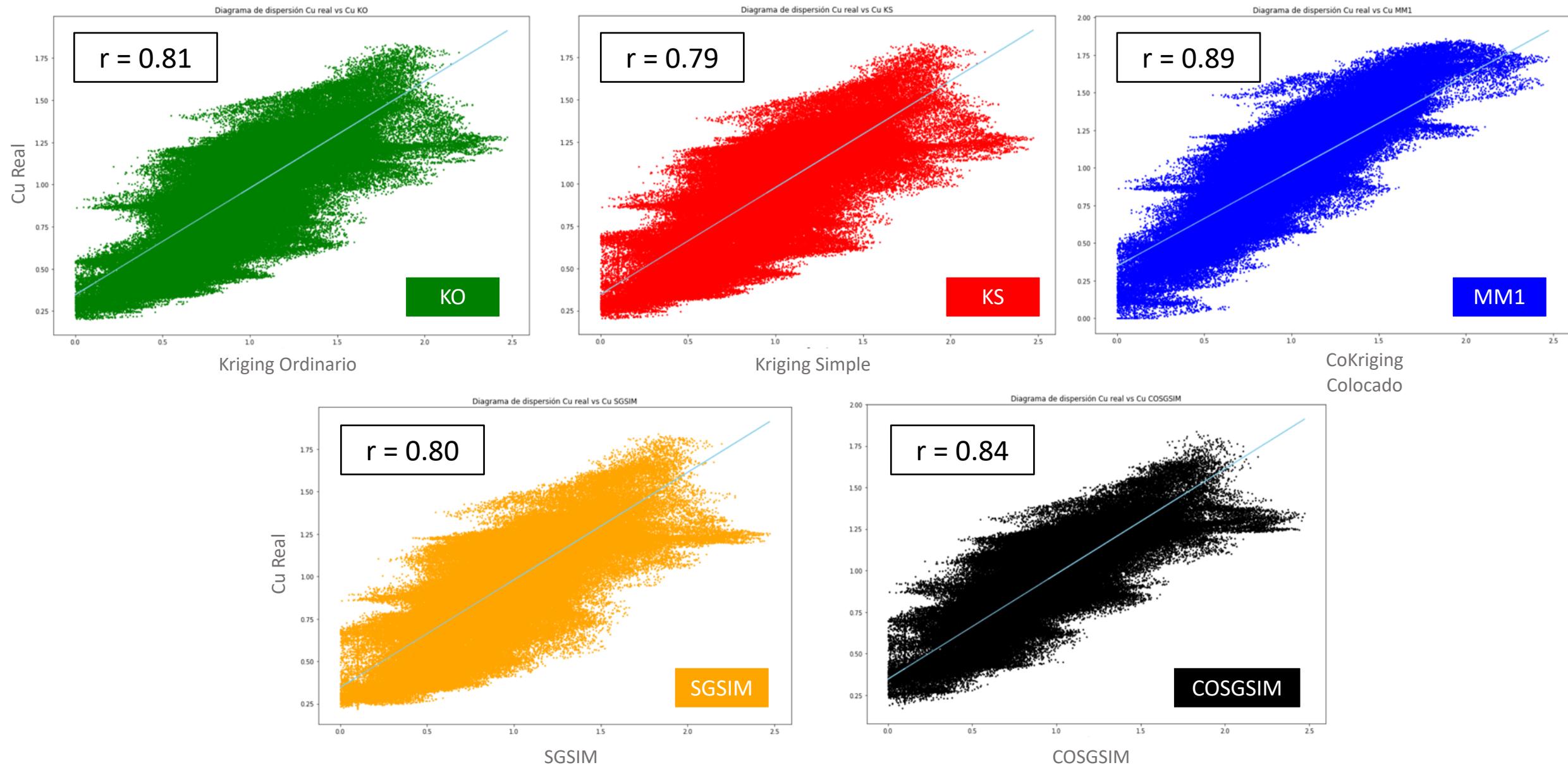


COSGSIM: Sequential Gaussian Co
Simulation

Comparación métodos de interpolación vs real Cu

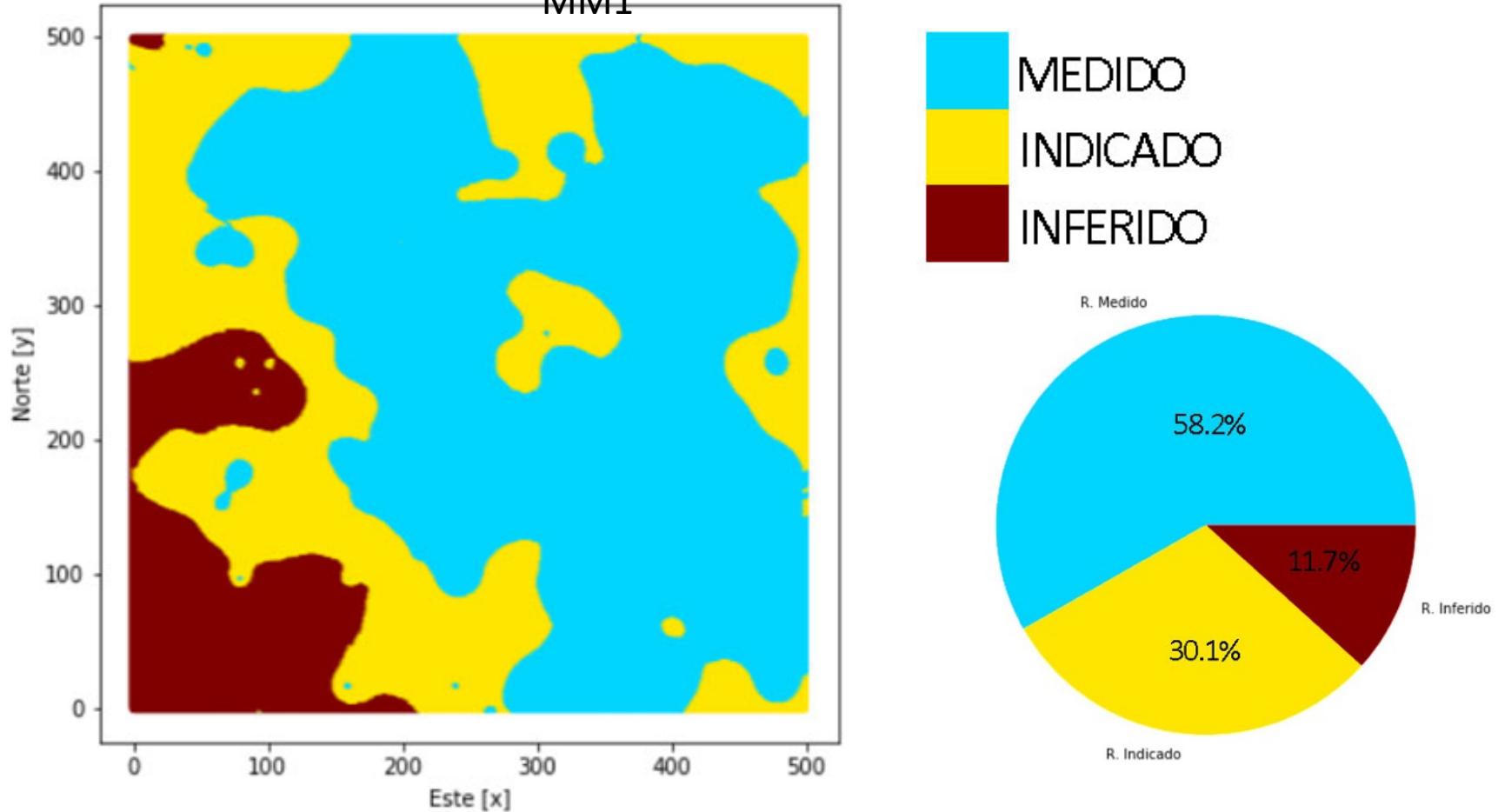


Comparación métodos de interpolación vs real Cu



Categorización del modelo por $CV(95\%) = 1.96 \cdot [\sigma/Z^*(x_0)]_{MM1}$

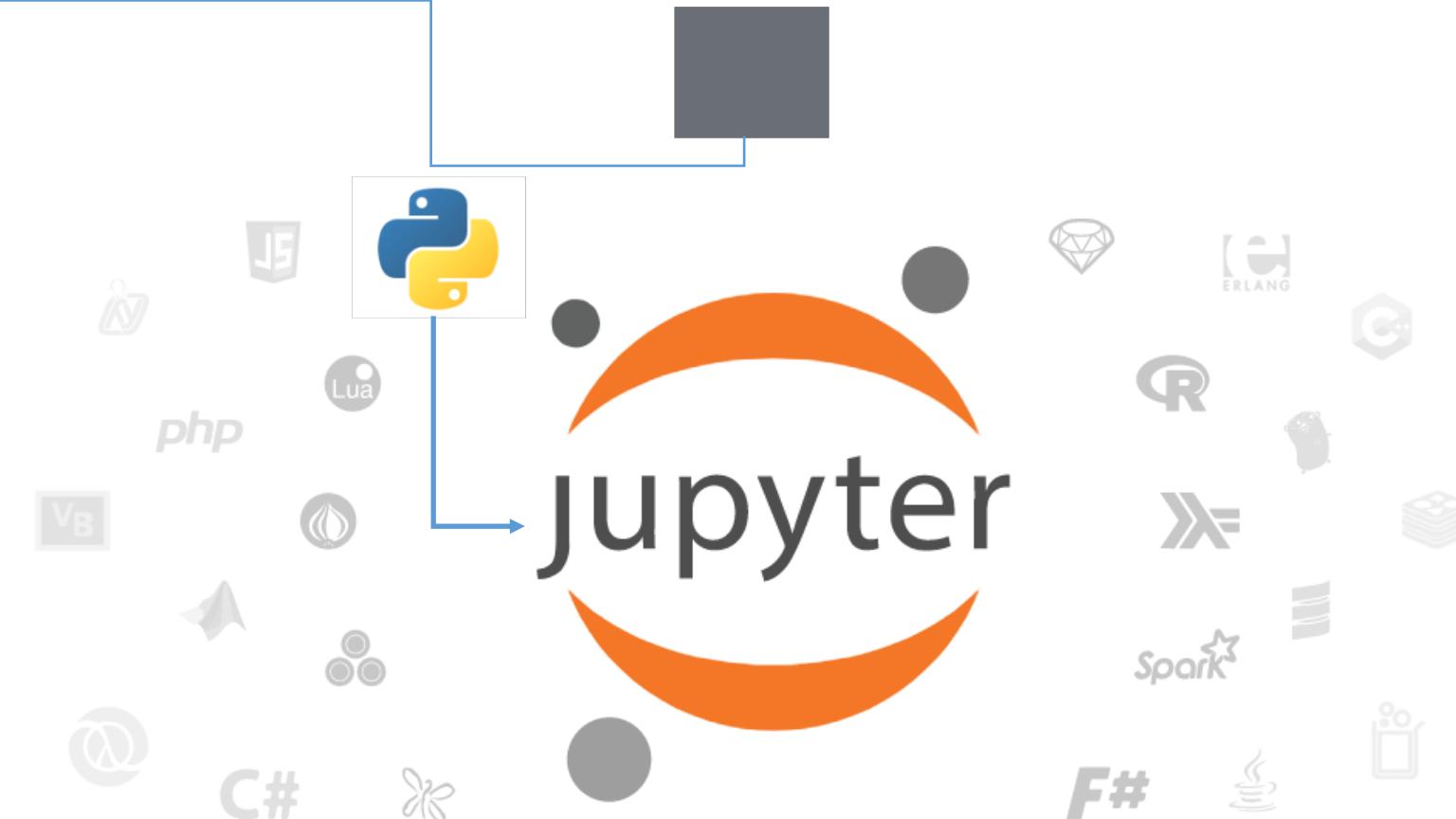
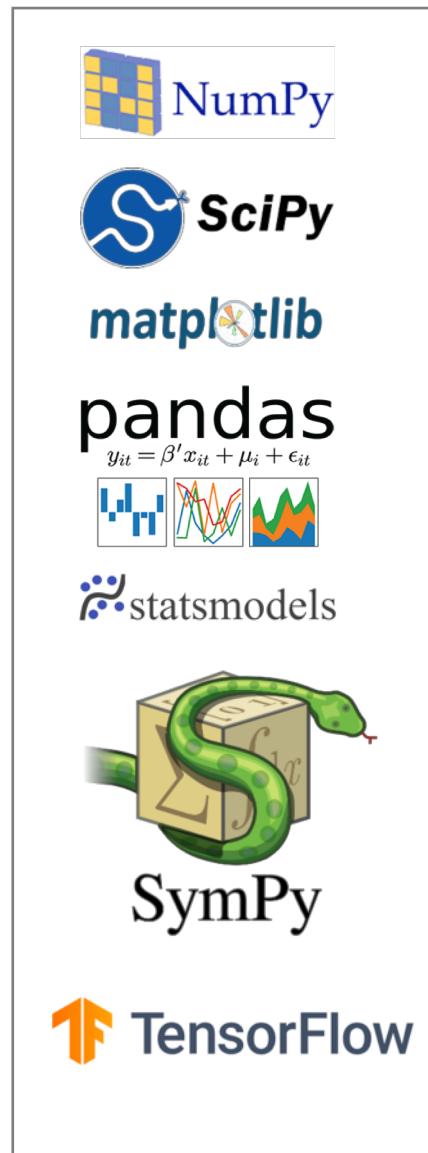
Mapa de categoría de recursos por CV CoKriging Co-localizado
MM1



$CV < 0.25$: 1, $CV \geq 0.25 \text{ & } \leq 0.5$: 2, $CV > 0.5$: 3

1: MEDIDO, 2: INDICADO, 3: INFERIDO

Beneficios Python en Jupyter Notebook:

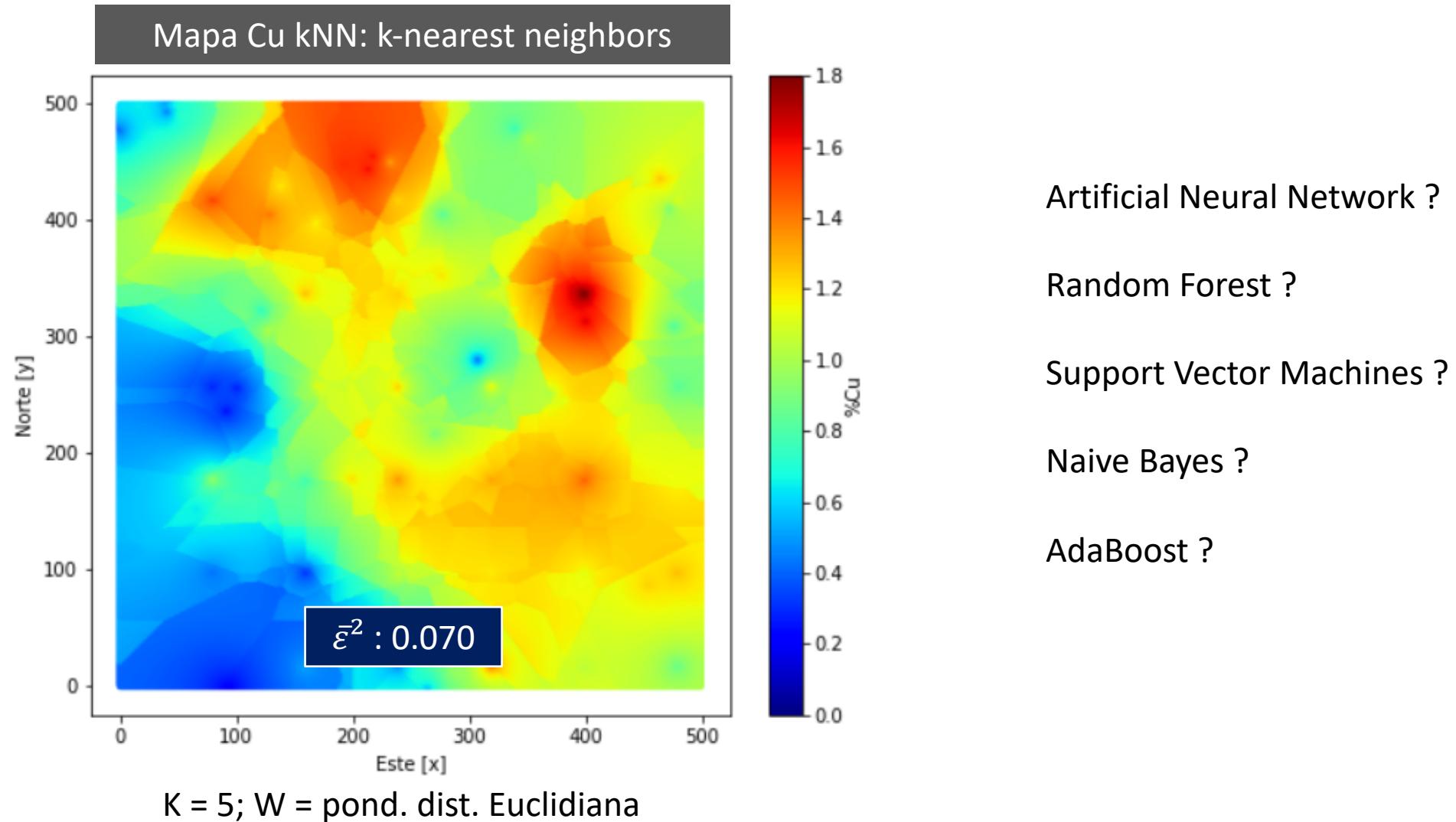


Big Data, Data Science y AI

PYPL Index (Worldwide)

| | | | |
|---|------------|--------|--------|
| 1 | Python | 27.34% | 4.50% |
| 2 | Java | 20.25% | -2.10% |
| 3 | Javascript | 8.51% | 0.00% |
| 4 | C# | 7.38% | -0.50% |
| 5 | PHP | 7.34% | -0.90% |

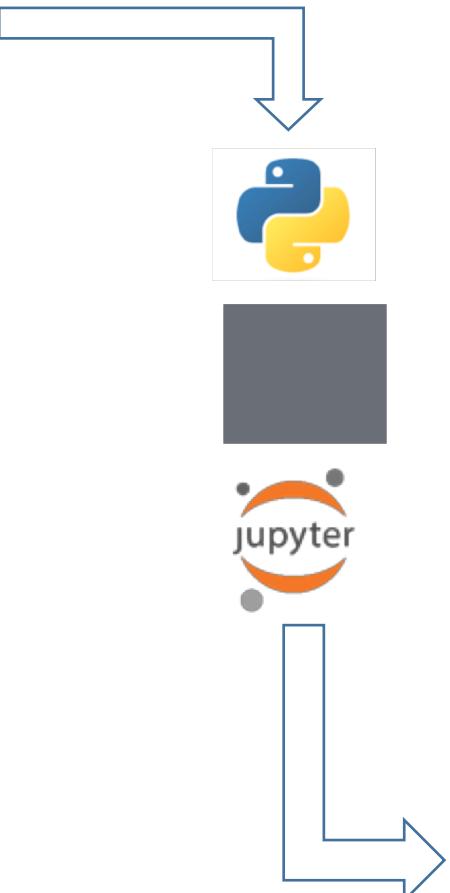
Oportunidad: Implementación de algoritmos de Machine Learning (IA) vs geoestadísticos



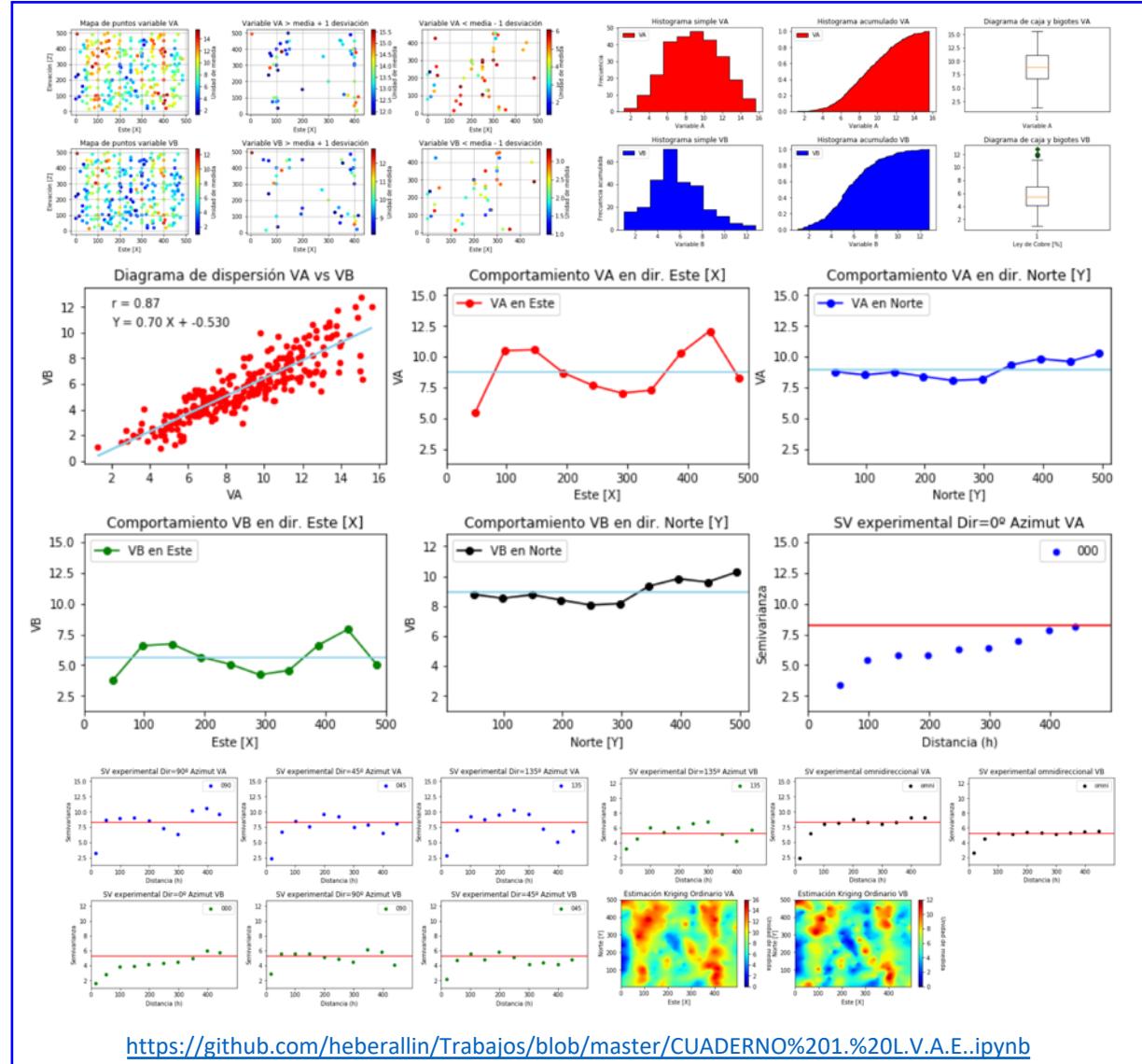
Trabajo en 2D: código que genera 29 gráficos a partir de una tabla de datos en 5 segundos (Intel corei7 7th gen).

| ENTRADA | | | | |
|---------|----------|-----------|-------|------|
| | Este [X] | Norte [Y] | VA | VB |
| 0 | 50 | 450 | 11.55 | 8.90 |
| 1 | 50 | 400 | 10.99 | 7.14 |
| 2 | 50 | 350 | 9.78 | 6.00 |
| 3 | 50 | 300 | 9.62 | 5.29 |
| 4 | 50 | 250 | 7.92 | 4.20 |

Cabecera tabla 2k datos (2D)



SALIDA



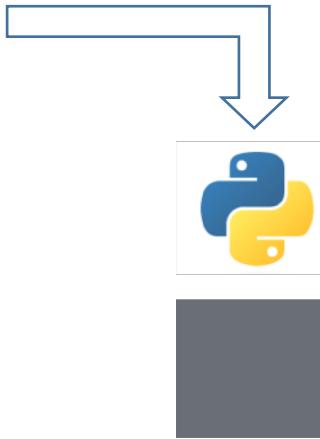
<https://github.com/heberallin/Trabajos/blob/master/CUADERNO%201.%20L.V.A.E.ipynb>

Trabajo en 3D: código que genera un modelo de bloques (KO) en 3 min., previamente configurado (Intel corei7 7th gen).

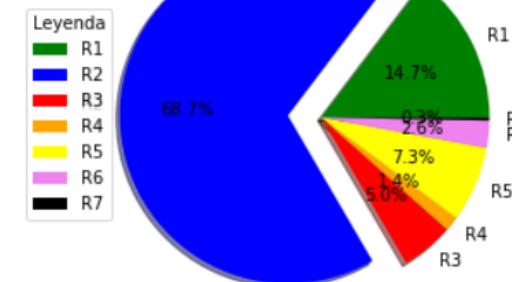
ENTRADA

| | Este [X] | Norte [Y] | Cota [Z] | Ley de Cobre [%] | Ley de Oro [ppm] | Tipo de roca |
|---|----------|-----------|----------|------------------|------------------|--------------|
| 0 | 193.0 | 528.6 | 39.0 | 0.12 | 0.000 | 1 |
| 1 | 335.1 | 38.0 | 97.0 | 0.13 | 0.028 | 1 |
| 2 | 250.7 | 593.4 | 36.0 | 0.13 | 0.027 | 1 |
| 3 | 275.8 | 517.2 | 86.1 | 0.19 | 0.000 | 1 |
| 4 | 256.1 | 529.1 | 61.8 | 0.19 | 0.039 | 1 |

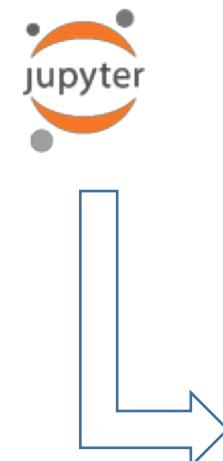
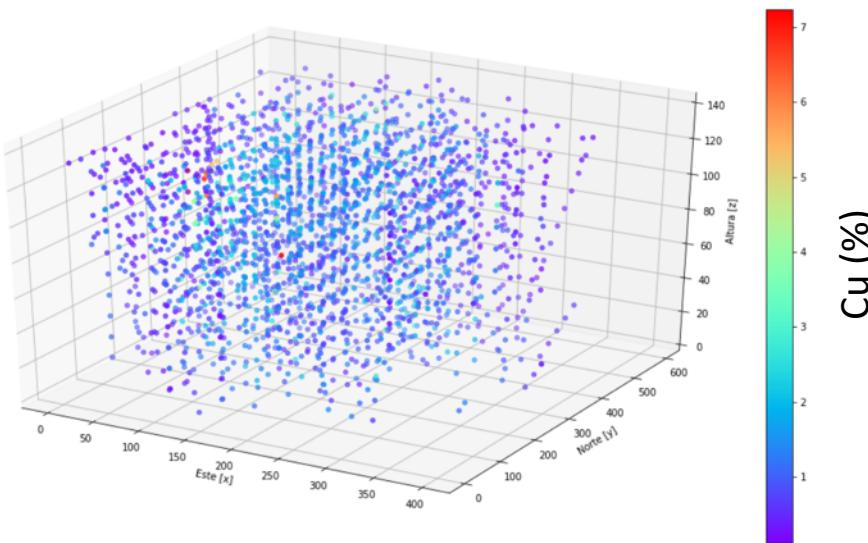
Cabecera de tabla de datos 3D



Tipo de roca predominante (R2)

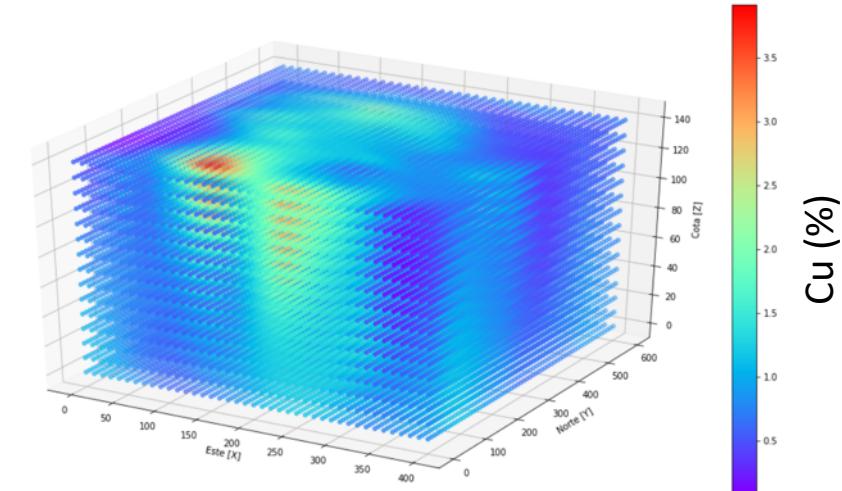


2,380 Compositos de Cobre (Cu)



SALIDA

33,600 bloques(10x10x10) de Cu



https://github.com/heberallin/Trabajos/blob/master/modelo_3d_cu.ipynb

Gracias por su atención

MSc. Heber Hernández Guerra
heber@nubeminera.cl



NUBE MINERA
PLATAFORMA E LEARNING & CONSULTORIA

