

CONVERGENCIA Y CLASIFICACIÓN DE ALGORITMOS DE OPTIMIZACIÓN

Semestre 2011-2

CONVERGENCIA

- Definición **errónea** común:
 - “Cuando se llega al óptimo global.”
 - “Cuando se llega a un mínimo/máximo.”

DISCLAIMER:

LO SIGUIENTE ES LO QUE
NORMALMENTE SE EXPONE
EN MATERIA DE CONVERGENCIA

CONVERGENCIA

- Según Bonnans et. al.
 - “Cuando se llega a *lo deseado*.”

LO DESEADO

- Lo que satisface las **condiciones de optimalidad**.
- Depende de la función a optimizar y del algoritmo empleado.

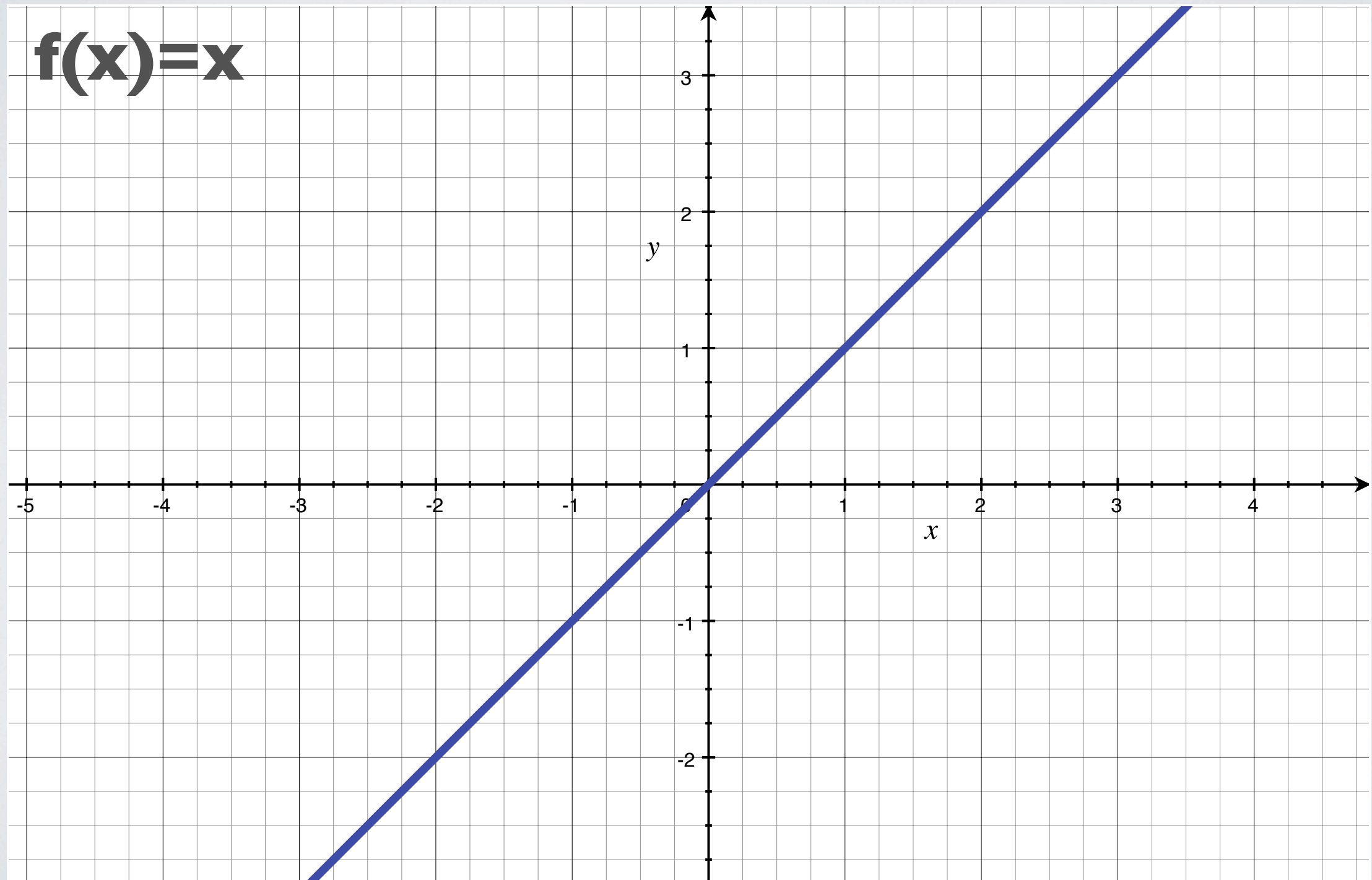
CONDICIONES DE OPTIMALIDAD

- Para una función continua derivable, si:
 - $\nabla f(x) = 0$ [requisito mínimo]
 - El algoritmo ha **convergido globalmente** en x .

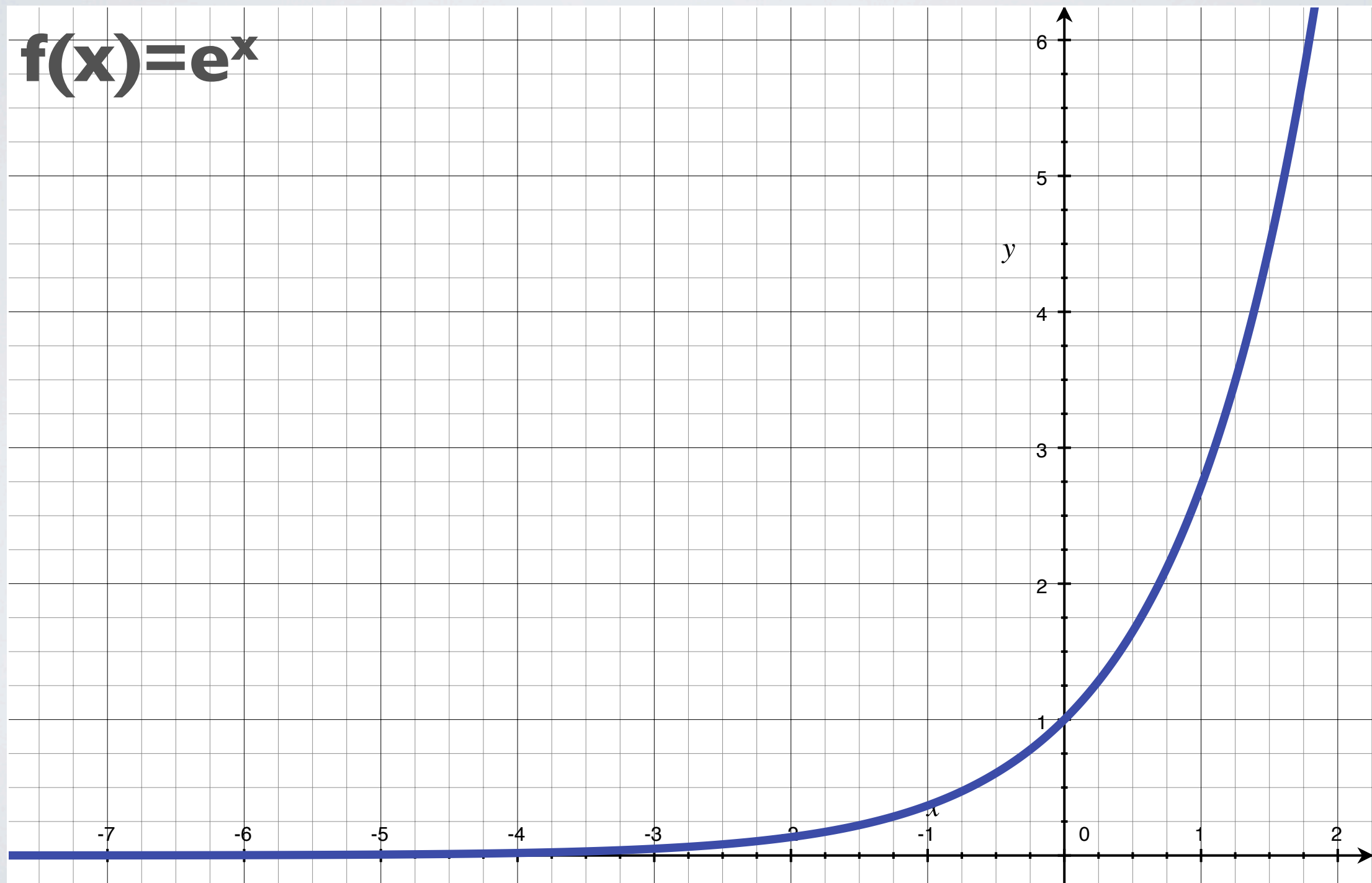
¿PUEDE CONVERGER?

- *Boundedness.*
- Si $f(x)$, para toda x , está en un rango menor a $(-\infty, \infty)$, es decir, si existe $\liminf(f(x))$ y/o $\limsup(f(x))$:
 - Existe por lo menos *una* secuencia de x_k 's que arriben a una x^* tal que $\nabla f(x^*) = 0$.
- Por ende, **sí**.

¿CUÁNDO NO?



¿Y AHORA?



GENERALIDADES DE CONVERGENCIA

- Habiendo asegurado la convergencia global de la optimización de f , lo siguiente a resolver es:
 - ¿Cuán tan rápido lo hace? (# de iteraciones)
 - ¿Cuáles valores de los parámetros del algoritmo *maximiza* la **velocidad de convergencia**?
- Un problema de optimización en sí mismo.

ACLARACIÓN

- Converger $\neq$ Optimizar
 - Excepción: $\nabla f(x) = 0$ es una condición suficiente para un mínimo global sólo si f es convexa. [Bonnans et. al.]
- Ejemplo: casos en los que NO conocemos las “entrañas” de f .

FIN A LO NORMALMENTE
EXPUESTO EN MATERIA DE
CONVERGENCIA

CONVERGENCIA IGNORADA

- Tratar a la convergencia así es resultado de una manera de trabajar poco pragmática hacia el problema de optimización.
- Por ejemplo...

CLASIFICACIÓN DE PROBLEMAS

[BONNANS ET.AL.]

1. Sin Restricciones

1.1. Cuadráticos (*Menos difícil*)

1.2. No Lineales

2. Con Restricciones Lineales

2.1. Con Restricciones de Equidad

2.1.1. Lineales-Cuadráticos

2.1.2. No Lineales

2.2. Con Restricciones de Inequidad

2.2.1. De Programación Lineal

2.2.2. Lineales-Cuadráticos

2.2.3. No Lineales con
Restricciones Lineales

3. De Programación No Lineal

3.1. Con Restricciones de Equidad

3.2. Generales (*Más difícil*)

PROBLEMA

- Todos estos tipos de problemas requieren conocimiento de f .
 - Por ende, del espacio de solución.
- Asumen conocer a ∇f en todo momento.
- Ignorando el problema de convergencia principal:
 - ¿Cómo sabemos que el algoritmo ha convergido?

CLASIFICACIÓN DE ALGORITMOS

[RASCÓN Y LENNOX]

- Métodos Basados en Gradientes
 - Gauss-Sidel
 - Newton-Raphson
 - Búsqueda de Líneas (Armijo-Wolfe)
- Algoritmos de Caja Negra
 - Templado Simulado
 - Algoritmos Genéticos
 - Inteligencia de Enjambre
 - Evolución Diferencial

ALGORITMOS DE CAJA NEGRA

- Caja Negra: sólo entradas y salidas.
 - No hay conocimiento *a-priori* de f .
- Versátiles, requiriendo relativamente poca sintonización de parámetros.
- Emplean métodos estocásticos.
 - Usualmente, simulan fenómenos naturales.

PARA MATEMÁTICOS...

- Son algoritmos que calculan el salto de x_i a x_{i+1} como:
 - $x_{i+1} \leftarrow x_i + \text{random}$
 - Donde “random” es un factor estocástico que puede o no tomar en cuenta el valor de $f(x_i)$.

CLARO...

- Obviamente, lo anterior es sólo una sobre-simplificación.
- Pero, “introduce” un problema que no les gusta a los matemáticos:
 - Ya no está $\nabla f(x)$; ya no sabemos si ha convergido.
- “Introduce” está en comillas, porque, en realidad, **este problema siempre ha estado ahí**, nada más se ha ignorado.

PERO...

¿CUANDO SABEMOS QUE YA
CONVERGIÓ?

ALTERNATIVA #1

LA MÁS POPULAR

- Cuando $f(x)$ pare de cambiar.
 - Numéricamente hablando, ¿Qué significa eso?

CONTINUACIÓN...

- ¿Cómo se decidió cuántas iteraciones esperar?

NÚMERO DE ITERACIONES

- “Empíricamente” es una mala palabra en ambas la academia y en la industria: lo óptimo tiene que asegurarse.
 - Es un sinónimo de “a ojo de buen cubero”.
- El número de iteraciones a esperar a que $f(x)$ no cambie, es un parámetro que comparten los algoritmos de caja negra.

LA PREGUNTA DE LOS 64,000

- Si el número de iteraciones fueron 1000, ¿Cómo sabemos que a la iteración 1001 no se hubiera encontrado un valor mejor de $f(x)$ de con el que terminamos?
- ¿Están preparados para vivir con eso pesando sobre su conciencia?

ALTERNATIVA #2

- ¿Sugerencias?

ESE ES EL PROBLEMA:
**NO HA HABIDO OTRA
ALTERNATIVA VIABLE**

De hecho, sí, pero nada significativamente mejor que
Alternativa #1

ALGO QUE LO ALIVIA

- Convergencia en Probabilidad
 - La probabilidad de una solución “inusual” decrece mientras la secuencia de números crece.
- Establecido en la Ley Débil de Números Grandes:
 - $\bar{A}_n \rightarrow \mu$ cuando $n \rightarrow \infty$
 - Donde $\bar{A}_n$ es el promedio de n muestras, y μ es el valor esperado.

EN ESPAÑOL

- Dadas suficientes muestras, la función proveerá el valor esperado.

¿QUÉ SIGNIFICA ESTO?

- Tenemos una forma de garantizar convergencia:
 - El algoritmo tiene que “arrastrar” información pasada.
 - Un número GRANDE de iteraciones:
 - No sólo para esperar a que $f(x)$ no cambie, sino de todo el proceso.
- Esta forma de “arrastre” puede indicarnos, a grosso modo, el número *mínimo* de iteraciones para garantizar convergencia.

SIGUIENTE CLASE

- Algoritmos Genéticos
- Inteligencia de Enjambre
- Evolución Diferencial
- Tarea para el **Lunes 7 de Marzo:**
 - Ensayo de 5 cuartillas (letra de 12 puntos, espacio sencillo) resumiendo y opinando sobre el artículo “*Simple Explanation of the No Free Lunch Theorem of Optimization*”.