NAVAJA NEGRA CONFERENCE

Redes Neuronales y Criptografía Moderna

Dr. Alfonso Muñoz @mindcrypt @criptored
D. David Ramírez @daysapro

Alan Turing – Machine vs Machine

"Una máquina puede ser vencida por otra máquina..." - Alan Turing

Navaja Negra 8...

NAVAJA NEGRA CONFERENCE

Google Develops Neural Networks that Can Communicate Secretly with Encrypted Messages

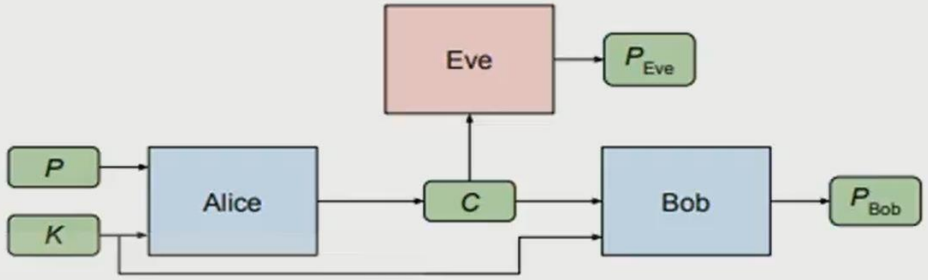
November 16, 2016 by Chantelle Dubois

Scientists at Google have successfully trained two neural networks, Alice and Bob, to communicate secretly using its own developed encryption in order to keep a third neural network, Eve, from listening in.

Google's AI Can Now Invent Its Own Cryptography

L1 distance $d(P, P') = \Sigma_{i=1,N}|P_i - P'_i|$ where N is the length of plaintexts.

$$L_E(\theta_A, \theta_E, P, K) = d(P, E(\theta_E, A(\theta_A, P, K)))$$



Criptografía = Algoritmo + Clave

Análisis Crítico y limitaciones (3/3)

Learning to Protect Communications with Adversarial Neural Cryptography – Martín Abadi, David G. Andersen (Google Brain)

- **Efecto avalancha – Modificación de bits en función de P y K**
  - Evaluación en fase de entrenamiento: generamos 17.408.000 criptogramas de 128 bits (aleatorios*)
  - *Probabilidad de modificación por bloques* (Modificación de los bits no equiprobable)
    - **Bits 1-16 = 69%** **Bits 16-32 = 21%** Bits 32-48 = 0,2% Bits 48-64 = 0,2%
    - Bits 64-80 = 0,2% Bits 80-96 = 0,2% **Bits 96-112 = 1,2% Bits 112-128 = 7,1%**
    - *Bits modificados valor máximo 127 (cambiamos todos los bits) y valor mínimo 1*
- **La red neuronal infiere que la clave de entrada debe afectar a la modificación del mensaje de entrada y lo consigue sin procedimiento previo**. Deduce un función criptográfica similar a un XOR…

Navaja Negra 8 - Criptografía, Deep Learning y Google - Alfonso Muñoz Muñoz

Navaja Negra Confer… 2,33 K suscriptores | Suscribirme | 13 | Compartir

https://www.youtube.com/watch?v=VizSx4dwJMw

Criptografía adversaria usando deep learning. Limitaciones y oportunidades Dr. Alfonso Muñoz et al (RECSI XV – 2018) https://nesg.ugr.es/recsi2018/docs/ActasXVRECSI.pdf

Whoami

Dr. Alfonso Muñoz

Hacker old school, senior researcher/pentester, teacher, book writer
Founder Criptored... +20 years attacking and protecting

@criptored 𝕏 @mindcrypt
alfonso@criptored.com
https://es.linkedin.com/in/alfonsomuñoz
https://www.amazon.es/Criptografía-Ofensiva-Atacando-defendiendo-organizaciones/dp/B0D8YPT322

D. David Ramírez Acero

Delivery Analyst en Deloitte, Vulnerability Management, CTF player

@daysapro 𝕏 @daysapro
https://www.linkedin.com/in/david-ramírez-acero-3bb282266/

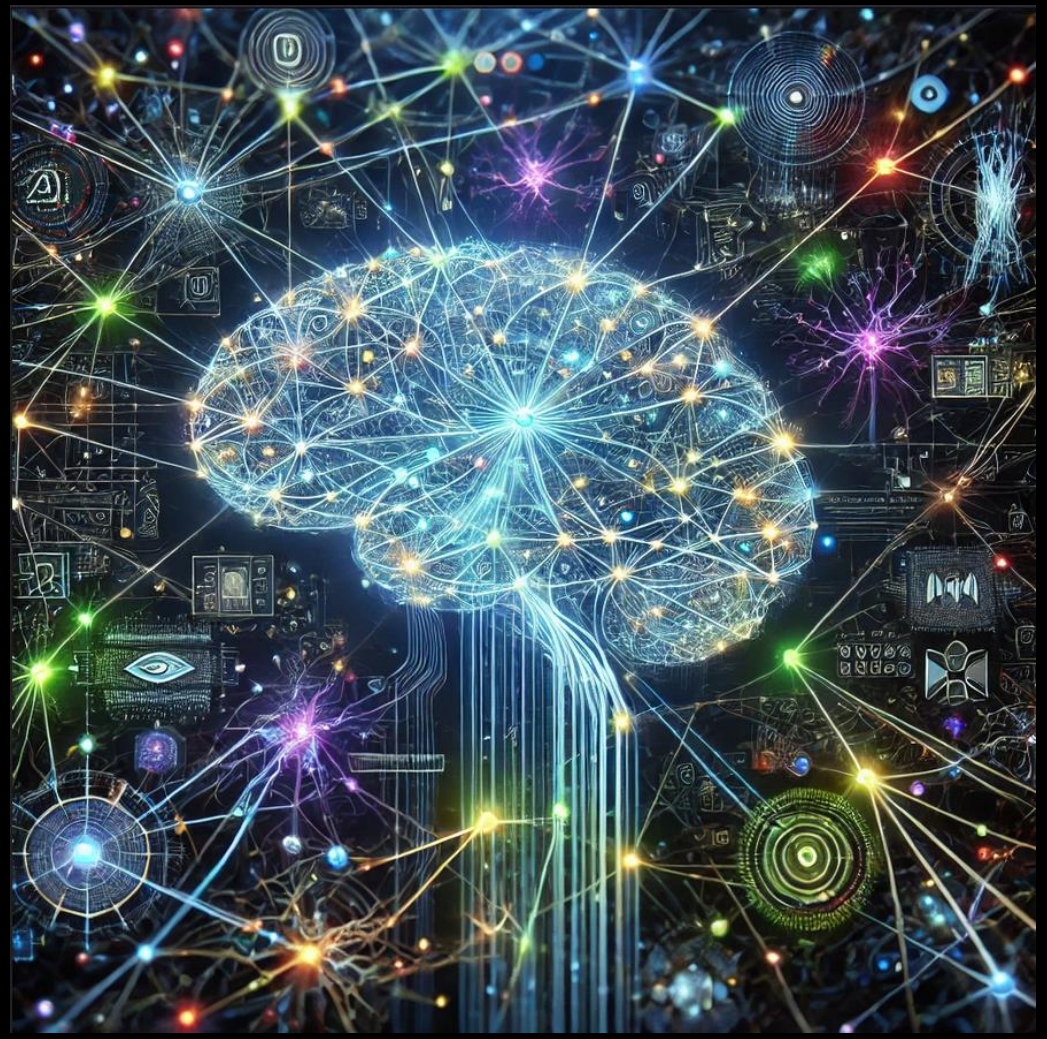
Try harder...

- Análisis del uso de las redes neuronales en criptografía y criptoanálisis

- Es posible crear un cifrador simétrico con calidad humana...
- ¿Cifrado útil para un período corto de tiempo?

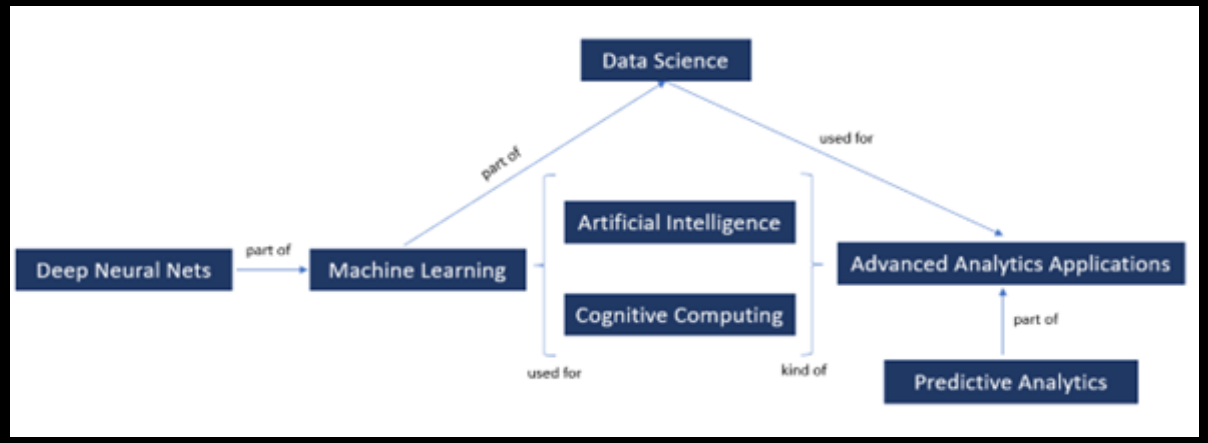
  - Confusión/difusión
  - Efecto avalancha
  - Redes de permutación-sustitución
  - ....

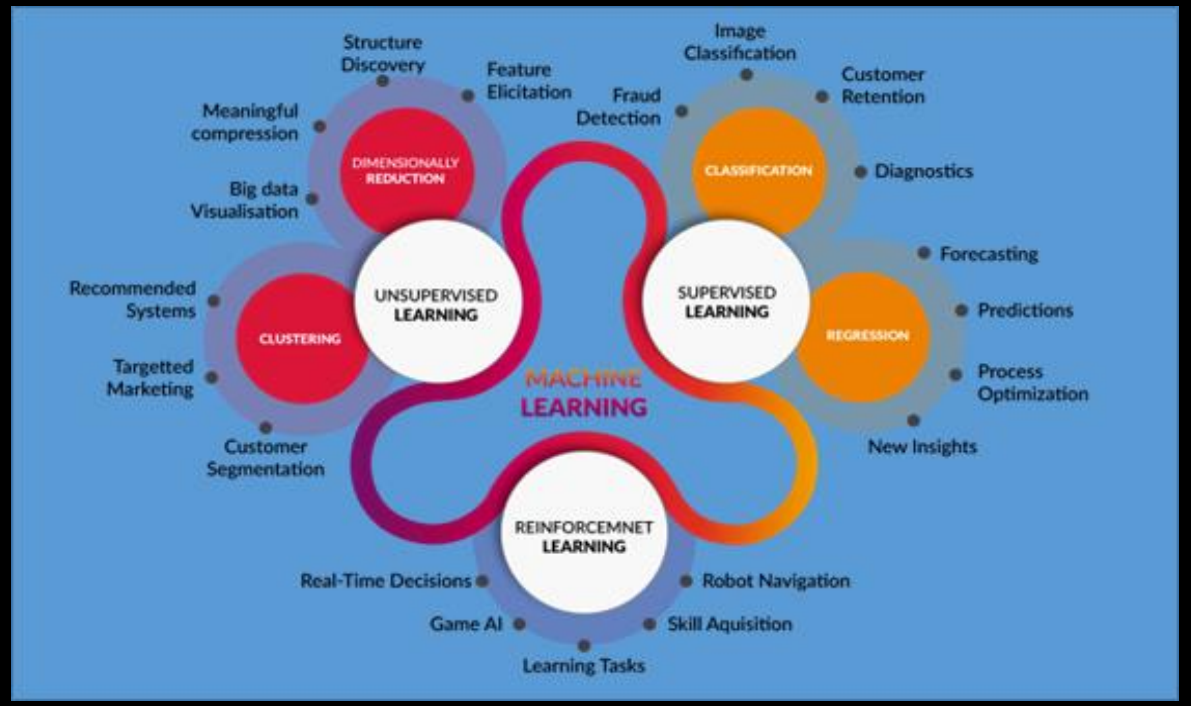
NAVAJA NEGRA CONFERENCE



Machine learning y redes neuronales

Machine Learning - 101

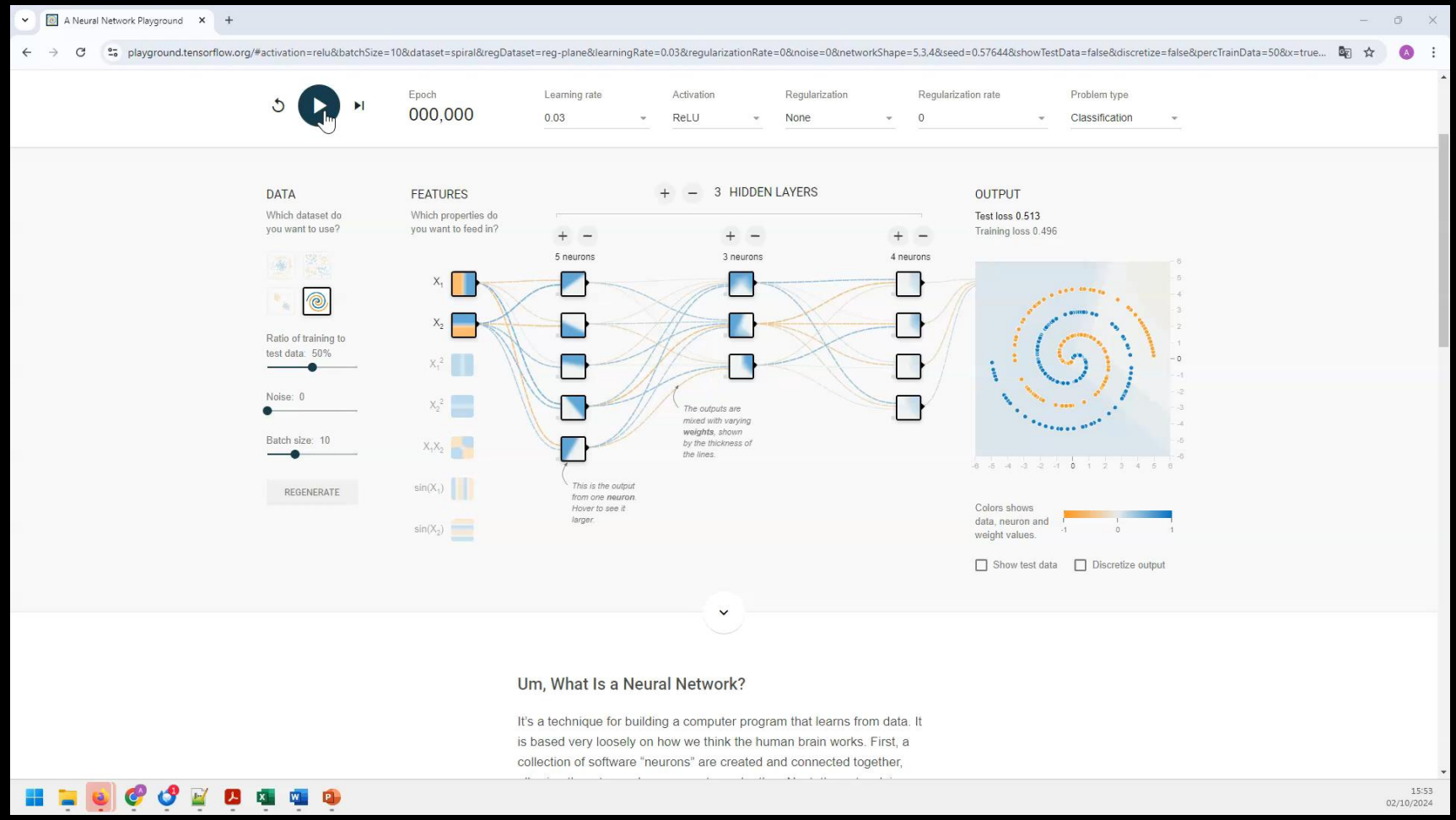




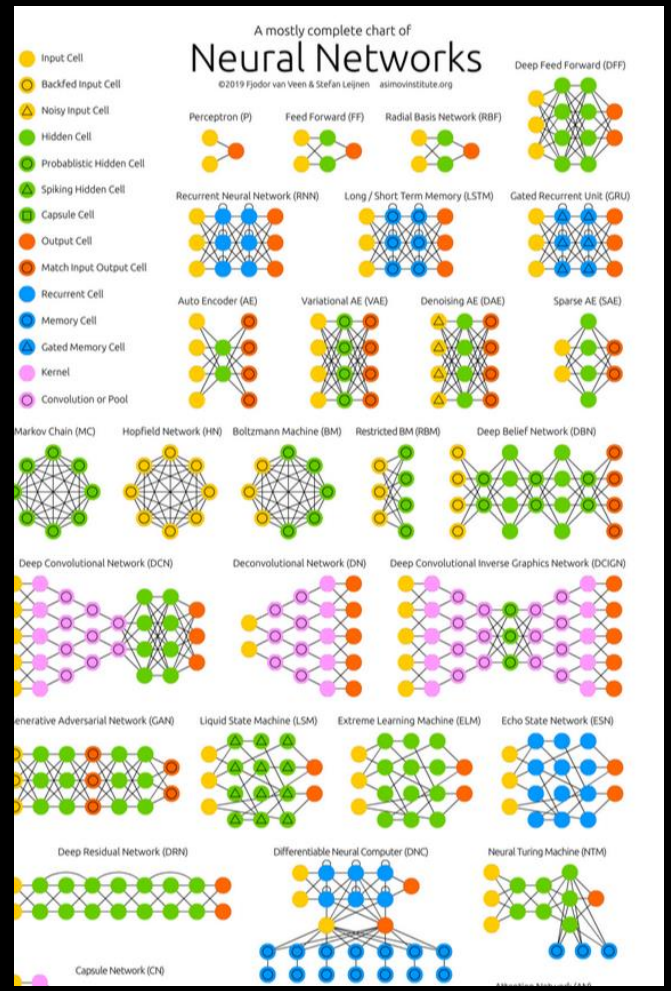
"Machine learning (**ML**) is a field of study in artificial intelligence concerned with the development and study of statistical algorithms that can learn from data and generalize to unseen data and thus perform tasks without explicit instructions. Recently, artificial neural networks have been able to surpass many previous approaches in performance..." - Wikipedia

Neural network - 101

NAVAJA NEGRA CONFERENCE



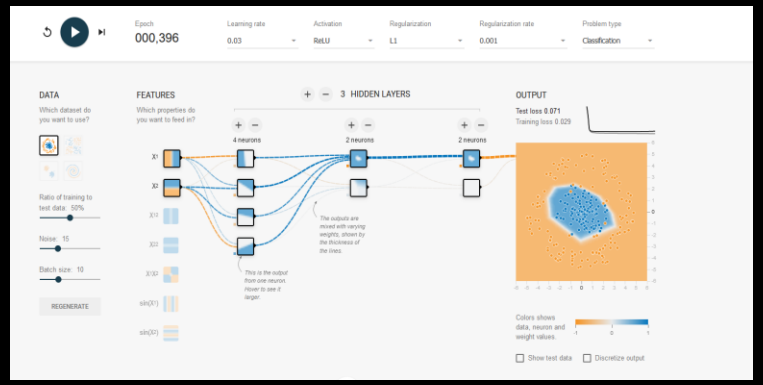
https://playground.tensorflow.org/



https://www.asimovinstitute.org/neural-network-zoo/

Neural network – Funciones de activación

NAVAJA NEGRA CONFERENCE



Si no se utiliza una función de activación, la red neuronal simplemente realizaría transformaciones lineales (multiplicación por pesos y sumas), lo que limitaría su capacidad para resolver problemas complejos, ya que muchas tareas del mundo real (como el reconocimiento de imágenes o el procesamiento del lenguaje natural) requieren modelos no lineales. La función de activación permite que la red neuronal capture esas no linealidades y aprenda relaciones complejas en los datos...

1. **ReLU (Rectified Linear Unit):**
   - Fórmula: $f(x) = \max(0, x)$
   - **Descripción:** La función ReLU convierte cualquier entrada negativa en 0 y deja las entradas positivas sin cambios. Es una de las funciones de activación más populares en redes profundas debido a su simplicidad y eficiencia computacional.
   - **Ventaja:** Ayuda a mitigar el problema de la desaparición del gradiente y acelera la convergencia.
   - **Problema:** Neuronas pueden "morir" si sus valores se hacen siempre negativos y dejan de activarse (problema de "muertas de ReLU").

2. **Sigmoide (Sigmoid):**
   - Fórmula: $f(x) = \frac{1}{1+e^{-x}}$
   - **Descripción:** Esta función toma cualquier valor de entrada y lo transforma en un valor entre 0 y 1, similar a una curva en forma de "S".
   - **Ventaja:** Es útil cuando se necesita una salida que se interprete como una probabilidad.
   - **Problema:** Puede provocar el problema de **desaparición del gradiente** (los gradientes se vuelven muy pequeños durante el entrenamiento), lo que ralentiza o detiene el aprendizaje en redes profundas.

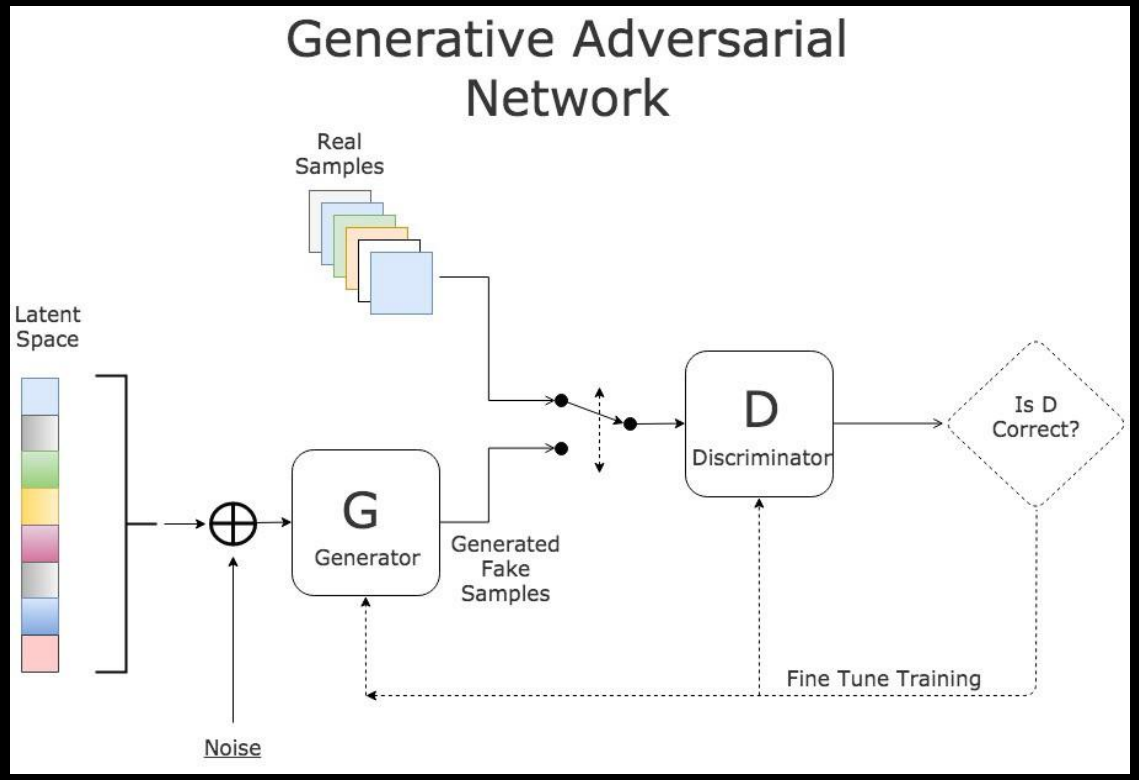
5. **Leaky ReLU:**
   - Fórmula: $f(x) = x$ si $x > 0$, y $f(x) = \alpha x$ si $x \leq 0$, donde α es un pequeño valor.
   - **Descripción:** Es una variante de ReLU que permite un pequeño gradiente cuando la entrada es negativa, evitando así que las neuronas "mueran".
   - **Ventaja:** Mantiene las ventajas de ReLU y evita el problema de neuronas inactivas.

3. **Tangente hiperbólica (Tanh):**
   - Fórmula: $f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
   - **Descripción:** La función Tanh también tiene una forma de "S", pero transforma los valores en el rango de -1 a 1. Es similar a la sigmoide, pero centrada en 0.
   - **Ventaja:** A menudo tiene un mejor rendimiento que la sigmoide, ya que los valores de salida están centrados en torno a 0, lo que facilita el aprendizaje.
   - **Problema:** Al igual que la sigmoide, puede sufrir del problema de desaparición del gradiente.

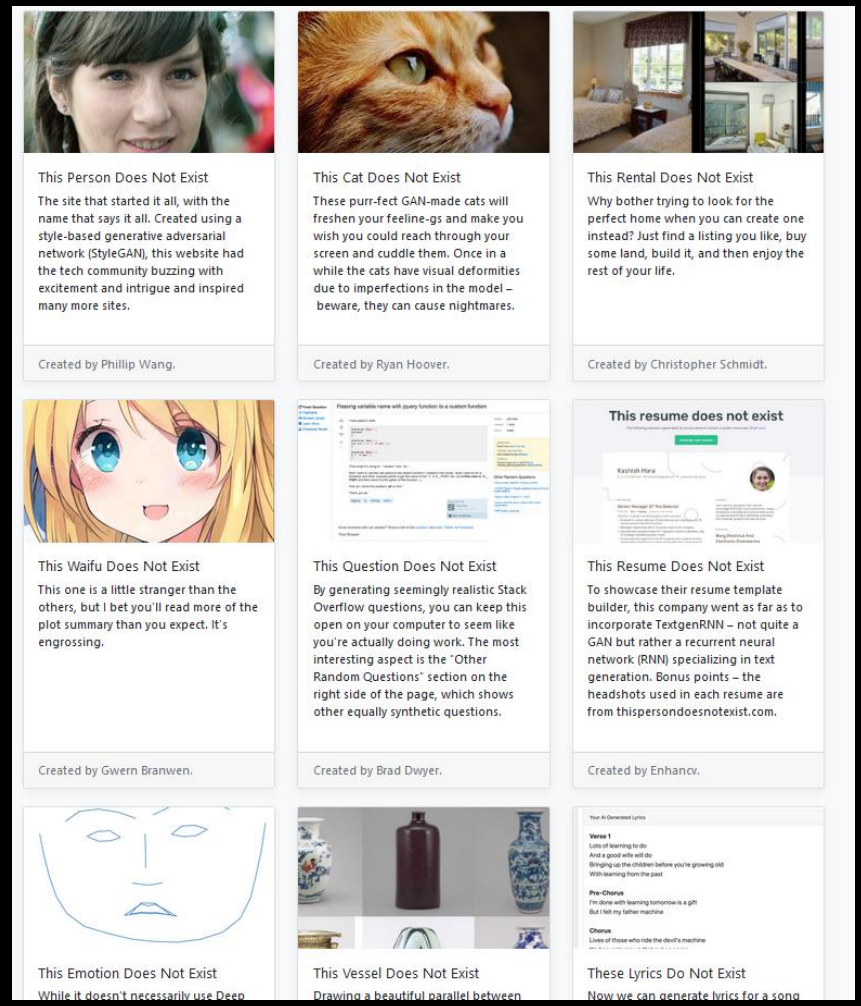
4. **Softmax:**
   - Fórmula: $f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$ (para cada i en la salida)
   - **Descripción:** Convierte un vector de entradas en una distribución de probabilidad. Se utiliza normalmente en la capa de salida de una red neuronal para tareas de clasificación con múltiples clases.
   - **Ventaja:** Da como resultado probabilidades normalizadas, lo que hace que sea útil para la clasificación multiclase.

6. **Swish:**
   - Fórmula: $f(x) = \frac{x}{1+e^{-x}}$
   - **Descripción:** Esta función es una versión suavizada de ReLU y ha mostrado en algunos casos un mejor rendimiento en redes profundas.
   - **Ventaja:** Es suave y no sufre el problema de las neuronas muertas de ReLU.

GAN - 101

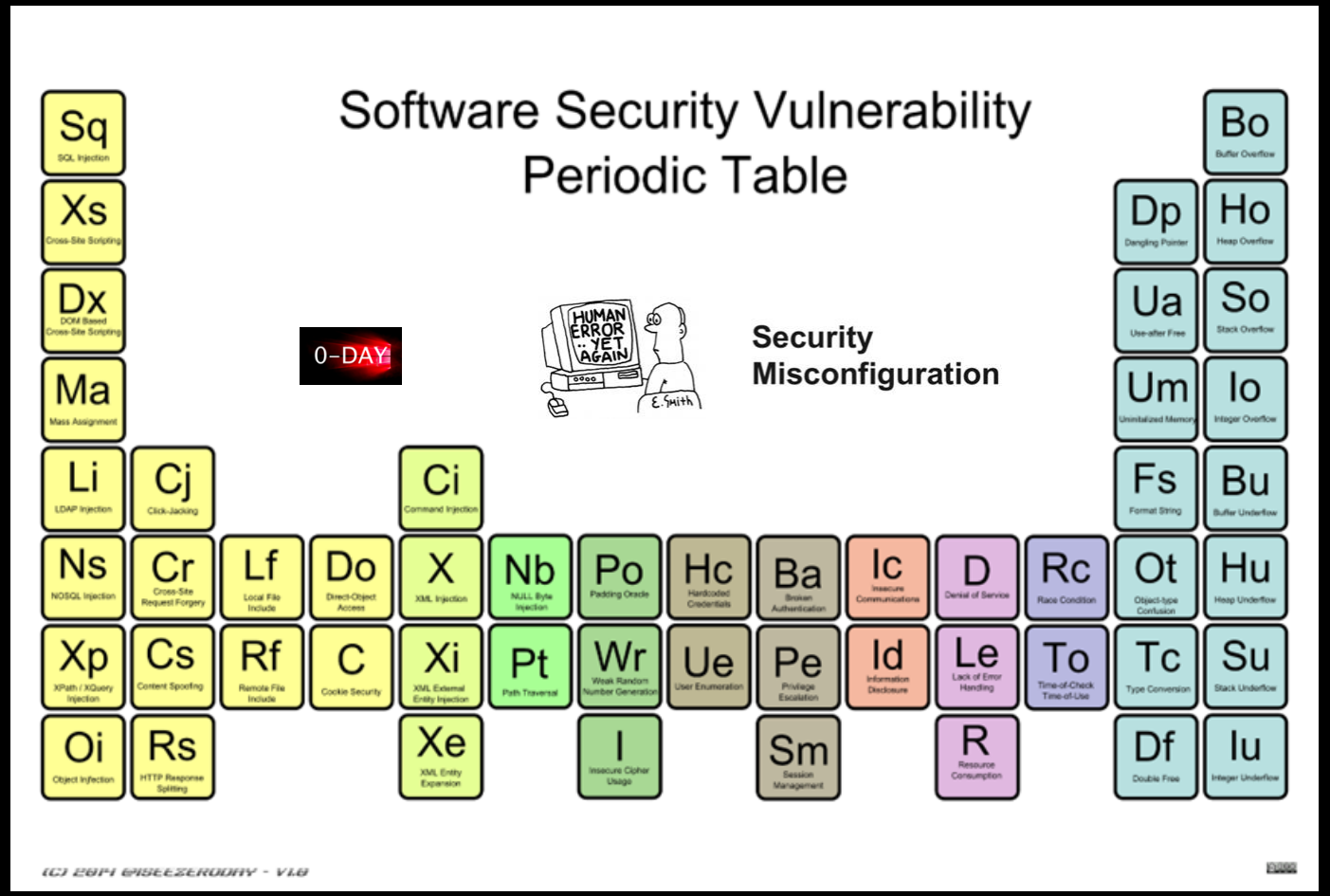


Ian J. Goodfellow et al. https://arxiv.org/abs/1406.2661



https://thisxdoesnotexist.com/

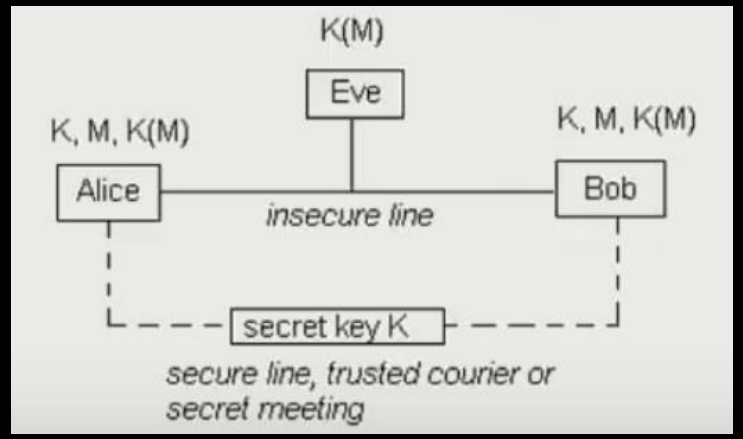
IA & Cyber



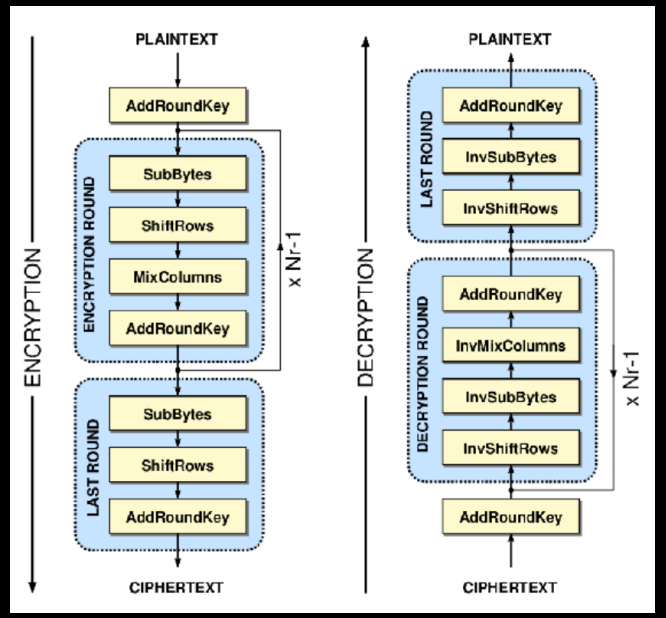
https://www.amazon.es/Seguridad-ofensiva-machine-learning-injections/dp/B0C91HCGKM

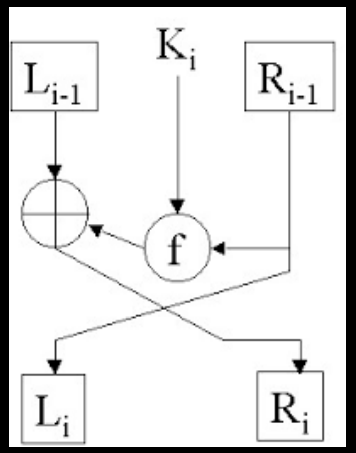
https://github.com/mindcrypt/libros/

Criptografía Simétrica



Bloques Básicos...





Red Feistel

- Confusión y difusión
  - Red Feistel, Redes Permutación-Substitución (SPN)...
- XOR y procesos no lineales (S-BOX, funciones Bent, ...)
  - Tamaño de la clave criptográfica. Ej. Gasto energético
  - Tamaño del bloque. Ej. Sweet32
    - https://github.com/mindcrypt/libros
    - Navaja Negra 9 – Reversing Cryptographic attacks over SSL/TLS - https://www.youtube.com/watch?v=m1Gwi6jKPCE
- Repetir un bloque básico operaciones (número de vueltas)
- Efecto avalancha (AES vs DES)

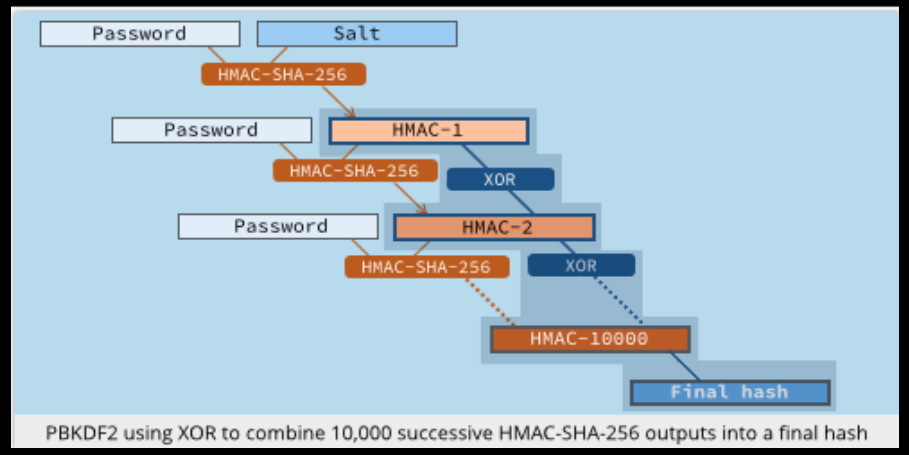
> the avalanche effect is the desirable property of cryptographic algorithms, typically block ciphers and cryptographic hash functions, wherein if an input is changed slightly (for example, flipping a single bit), the output changes significantly (e.g., half the output bits flip).

Bloques básicos…

NAVAJA NEGRA CONFERENCE

- Cifrado en cascada

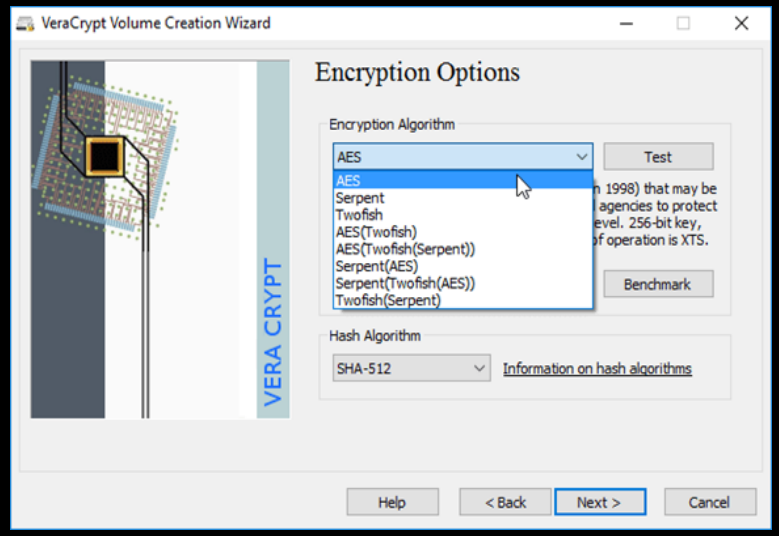
¿Tiene sentido cifrar varias veces una información con el mismo algoritmo criptográfico y la misma clave? ¿Y si usamos claves diferentes? ¿Aporta mayor seguridad?



PBKDF2 using XOR to combine 10,000 successive HMAC-SHA-256 outputs into a final hash

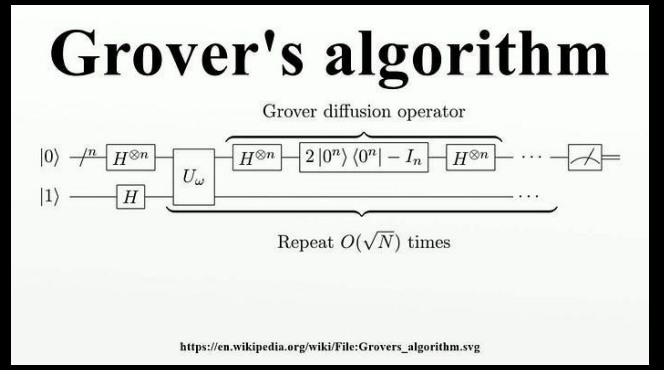
- Cifrado múltiple

¿Tiene sentido cifrar varias veces una información con distintos algoritmos y la misma clave? ¿Y si usamos claves diferentes? ¿En qué orden tienen que utilizarse los algoritmos?



Criptoanálisis - 101

- ¿Mi algoritmo es seguro?
  - Modelos de seguridad (IND-CPA, IND-CCA1, IND-CCA2)
  - Criptoanálisis diferencial, lineal, XSL, ...
  - No debe existir un ataque "mejor" (tiempo, almacenamiento, gasto energético...) que el uso de fuerza bruta. De lo contrario, el algoritmo se considera "roto"
    - Biclique Cryptanalysis of the Full AES - https://eprint.iacr.org/2011/449.pdf
  - Ataques cuánticos a algoritmos simétricos...



NAVAJA NEGRA CONFERENCE

NAVAJA NEGRA CONFERENCE

Criptografía con redes neuronales & GAN

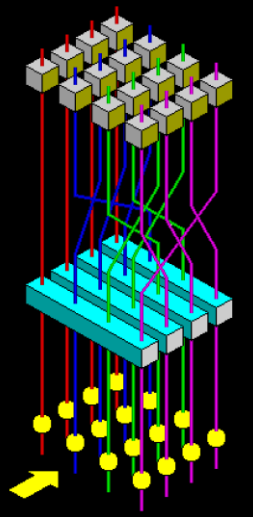
Estado del arte... (sin GAN)

https://www.imdb.com/title/tt0118884/

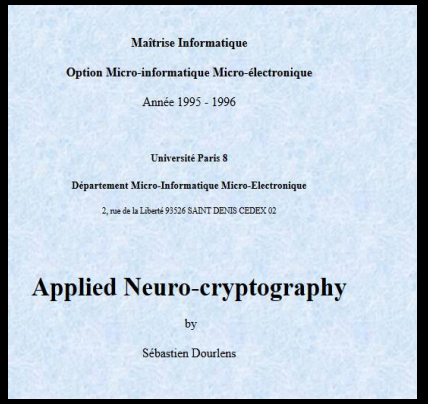
Estado del arte... (sin "GAN")

NAVAJA NEGRA CONFERENCE

- Neural cryptography/cryptanalysis vs CryptoNets
  - OTP, PRNG, autenticación de mensajes, generación de claves efímeras, confusión-difusión (red recurrente)...
  - Criptoanálisis de cifradores de bloque / clave pública
    - A Deeper Look at Machine Learning-Based Cryptanalysis - https://eprint.iacr.org/2021/287
    - Neural Networks-Based Cryptography: A Survey - https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9527229
    - Neural Network based Cryptography - http://www.nnw.cz/doi/2014/NNW.2014.24.011.pdf
  - Ataques crypto basados en canal lateral (~~cracking Kyber with IA~~)
  - ...



https://en.wikipedia.org/wiki/Advanced_Encryption_Standard/mediaFile:AES_(Rijndael)_Round



Maîtrise Informatique

Option Micro-informatique Micro-électronique

Année 1995 - 1996

Université Paris 8

Département Micro-Informatique Micro-Electronique

2, rue de la Liberté 93526 SAINT DENIS CEDEX 02

Applied Neuro-cryptography

by

Sébastien Dourlens

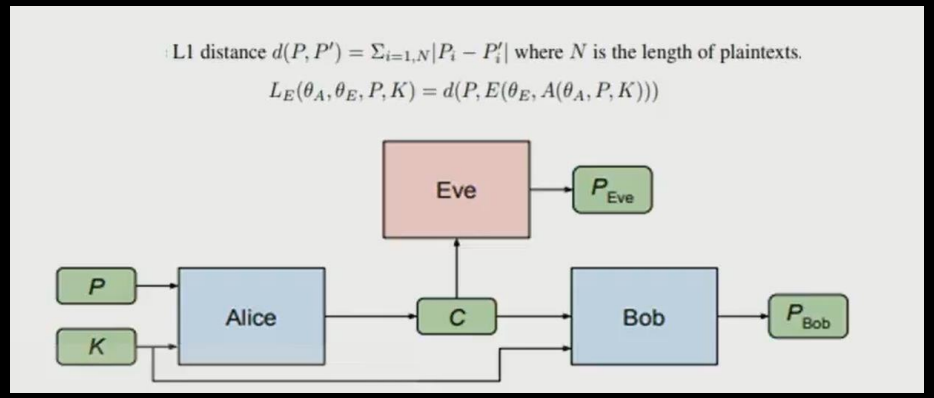
https://web.archive.org/web/20170918201301/http://s.dourlens.free.fr/AppliedNeuroCryptography.pdf

INICIO: Investigadores Google Brain (2016)…

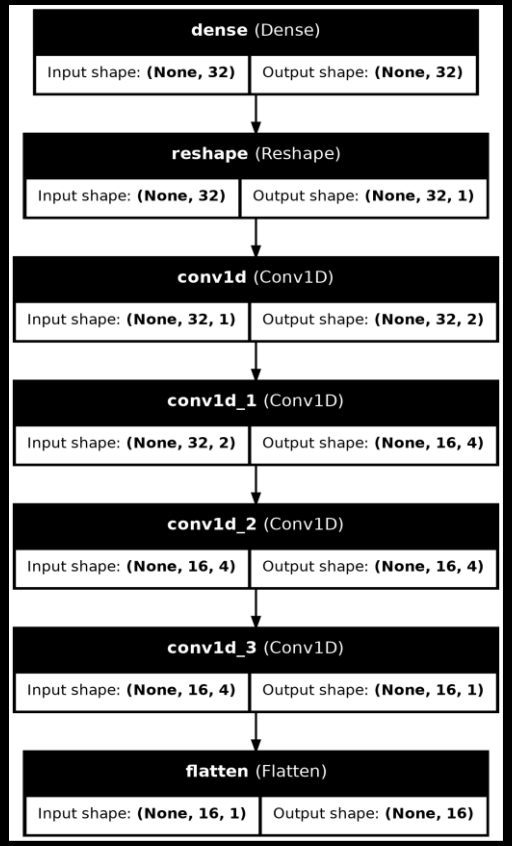
NAVAJA NEGRA CONFERENCE

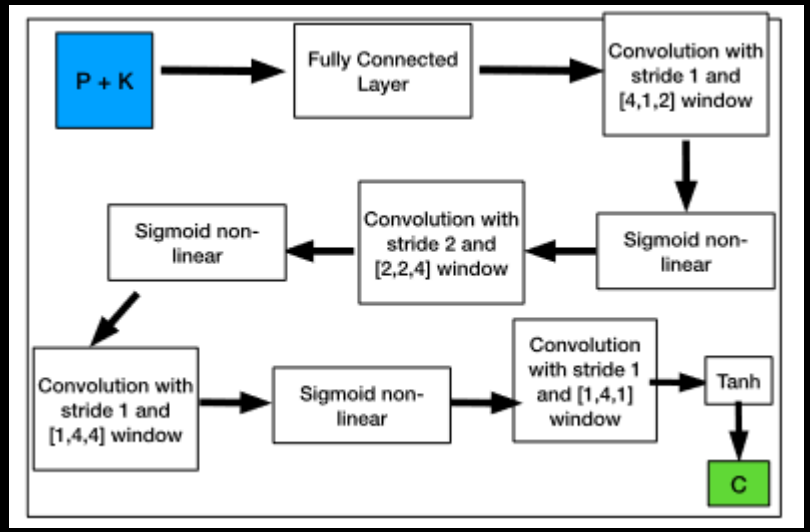
Learning to Protect Communications with Adversarial Neural Cryptography - https://arxiv.org/abs/1610.06918

- Alicia/Bob/Eva redes neuronales (cifradora/descifradora/atacante)



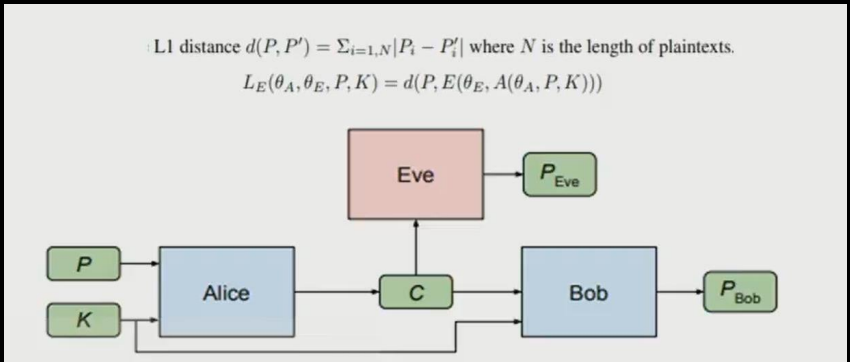
- Capa de entrada FC (Full Connected)
  - 16 bits Plaintext 16 bits Key
- Capa Conv1d
- Capa Conv1d_1
- Capa Conv1d_2
- Capa Conv1d_3
- Salida
  - 16 bits Ciphertext





Funcionamiento. La red busca sus objetivos...

Learning to Protect Communications with Adversarial Neural Cryptography - https://arxiv.org/abs/1610.06918



$$d(P, P') = \sum_{i=1}^{N} | P_i - P'_i |$$

$$d'(P, P') = \frac{1}{N} \sum_{i=1}^{N} | P_i - P'_i |$$

- Medimos la diferencia entre plaintext (entrada/salida)
  - Bob – Mide fallos en la reconstrucción (texto entrada Pi = texto salida Pi'). Intentan minimizar la función diferencia.
  - Eva - Cuánto éxito tiene al recuperar

$$L_{Eva} = d(P, P_{Eva})$$

$$L'_{Bob} = d'(P, P_{Bob})$$

$$L_{AliciaBob} = L'_{Bob} + \frac{(\frac{N}{2} - L_{Eva})^2}{(\frac{N}{2})^2}$$

- $L_{AliciaBob}$ -> Función de pérdida compartida durante el entrenamiento (Alicia y Bob)
  - Valores 0 (Bob recupera 100% Eva aleatoriamente) a 2 (Bob recupera aleatoriamente y Eva 100%)

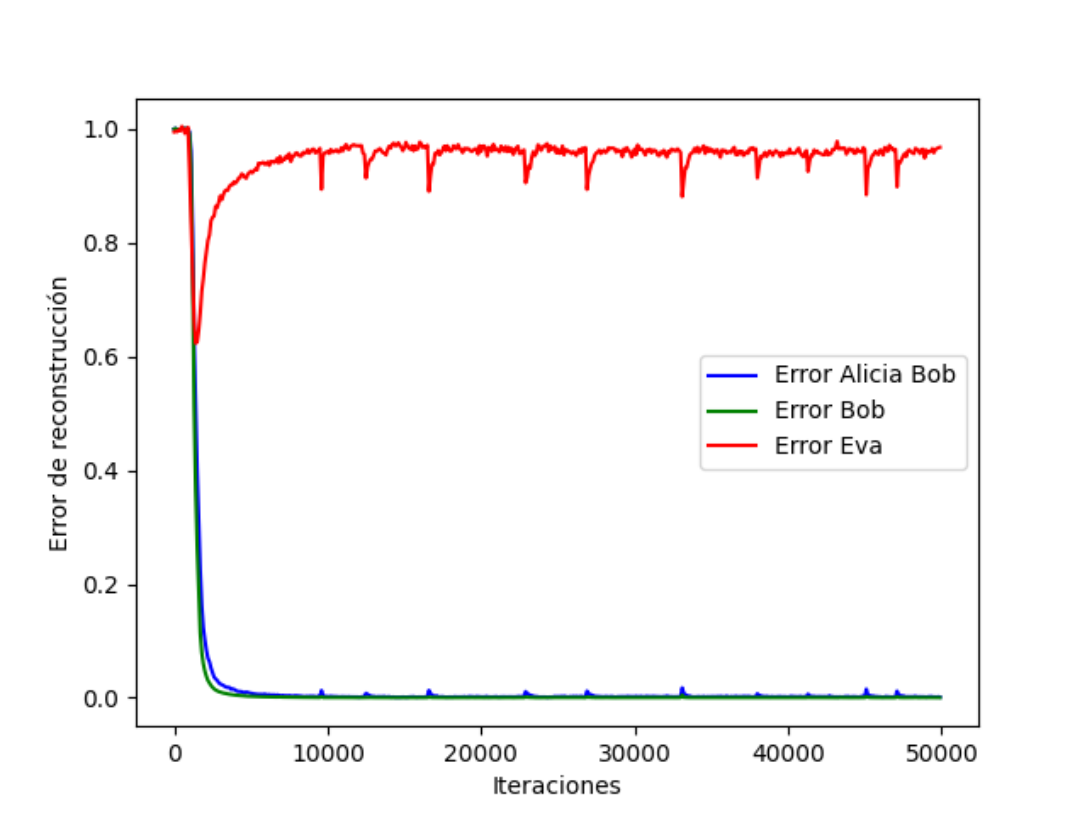
$$L'_{Eva} = d'(P, P_{Eva})$$

- Función de pérdida ("normalizada") de Eva
  - Valores 0 (Eva recupera 100%) o 1 (recupera aleatoriamente – aprox 50% bits erróneos)

NOTA: **El objetivo de Alicia/Bob es que Eva recupere con 50% de error. Si recupera con 100% de error invierte los bits y tiene la información original.**

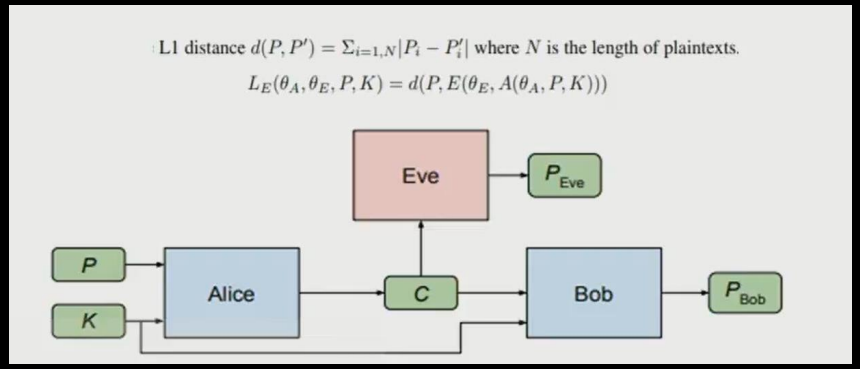
Entrenamiento

Learning to Protect Communications with Adversarial Neural Cryptography - https://arxiv.org/abs/1610.06918



Ejemplo:

- Eva rápidamente encuentra patrones (iteración 1500)
- Alicia/Bob frenan el avance de Eva (iteración 2500)
- Hay momentos de lucidez de Eva



CPU: AMD Ryzen 5 3600 - RAM: 32 GB - Python 3.12.3 - Tensorflow 2.16.1
Mensaje (16 bits) - Clave (16 bits)

Error de reconstrucción (1 => EVA recupera los bits con un 50% de bits erróneos, 0=> Eva recupera el mensaje original - 100%)

Limitaciones

Learning to Protect Communications with Adversarial Neural Cryptography - https://arxiv.org/abs/1610.06918

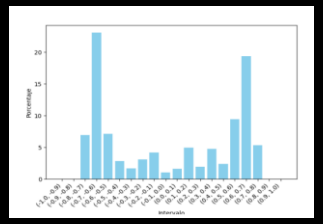
Intervalo	Repeticiones	Probabilidad
[-1, -0.9)	0	0 %
[-0.9, -0.8)	0	0 %
[-0.8, -0.7)	11066	6.92 %
[-0.7, -0.6)	36926	23.08 %
[-0.6, -0.5)	11425	7.14 %
[-0.5, -0.4)	4612	2.88 %
[-0.4, -0.3)	2732	1.71 %
[-0.3, -0.2)	4922	3.08 %
[-0.2, -0.1)	6730	4.21 %
[-0.1, 0)	1693	1.06 %
[0, 0.1)	2633	1.65 %
[0.1, 0.2)	7944	4.96 %
[0.2, 0.3)	3087	1.93 %
[0.3, 0.4)	7676	4.80 %
[0.4, 0.5)	3873	2.42 %
[0.5, 0.6)	15079	9.42 %
[0.6, 0.7)	30993	19.37 %
[0.7, 0.8)	8606	5.37 %
[0.8, 0.9)	3	0 %
[0.9, 1]	0	0 %
Total	**160000**	**100 %**

Descubre

- La importancia de la clave en la salida, todo sin información previa.

Problemas (criptoanálisis humano)

- Mensajes de corta longitud (16 bits)
- Gran dependencia de la clave y poca dependencia del mensaje
- Baja entropía (rotura estadística) - Nulas propiedades criptográficas

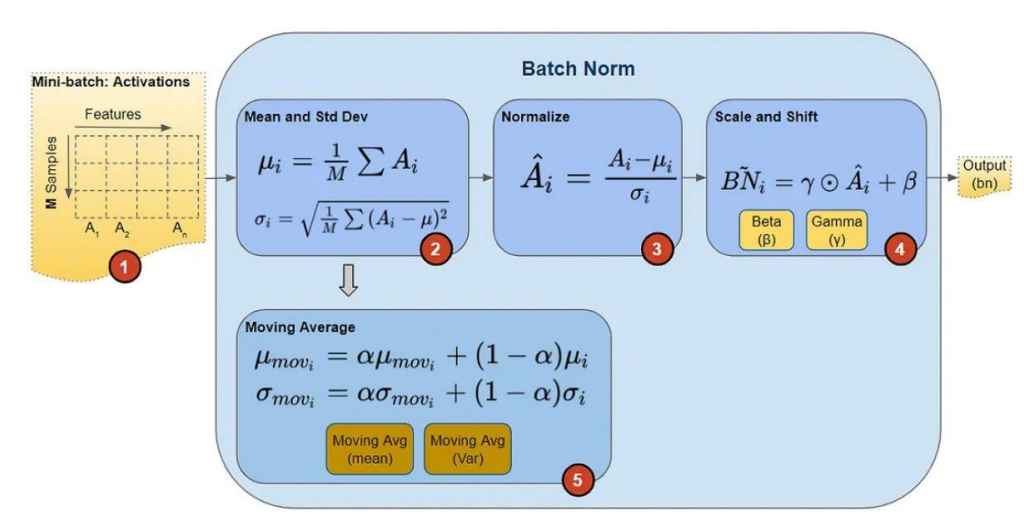


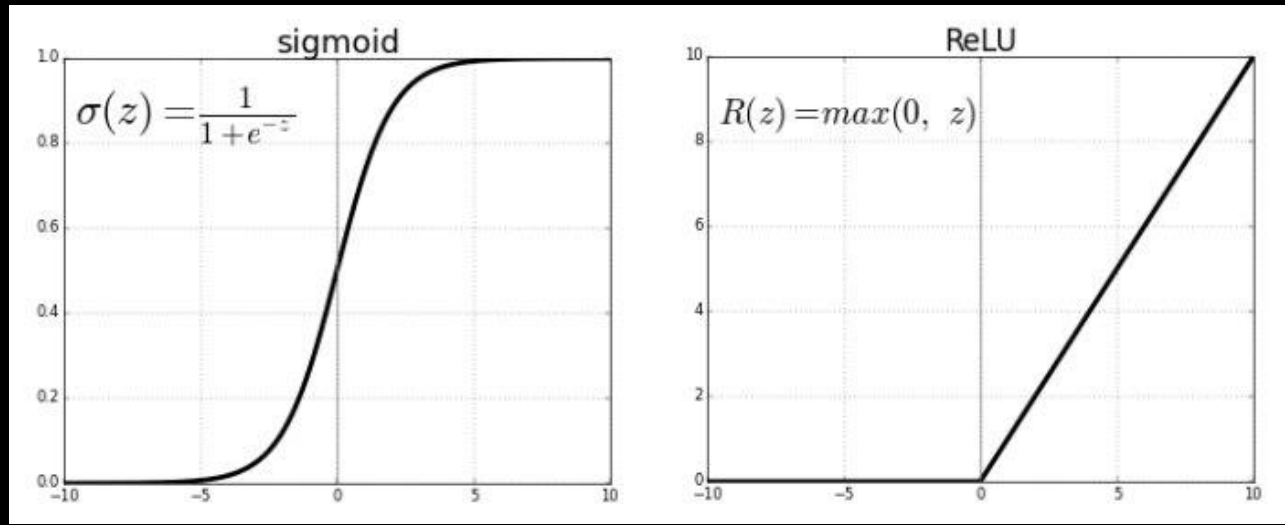
YO, INTENTANDO EXPLICAR “NEURAL CRYPTOGRAPHY”!!!!

¿Vamos a por algo un poco más complicado? :)

Nuevas propuestas y limitaciones

Duan, X., Han, Y., Wang, C., & Ni, H. (2021). “Optimization of Encrypted Communication Length Based on Generative Adversarial Network”
Duan, X., Han, Y., Wang, C., & Ni, H. (2022). “Optimization of Encrypted Communication Model Based on Generative Adversarial Network”
Singh, P., Dutta, S., & Pranav, P. (2024). “Optimizing GANs for 60 Cryptography: The Role and Impact of Activation Functions in Neural Layers Assessing the Cryptographic Strength”





Mejoras

Estabiliza entrenamiento y convergencia (minimizar mínimos locales) para mensajes de mayor longitud (128 bits).
Mejores tiempos de entrenamiento - Mejora las propiedades estadísticas y errores de reconstrucción.

Problemas

Baja entropía y dependencia de la clave pero poco del texto de entrada

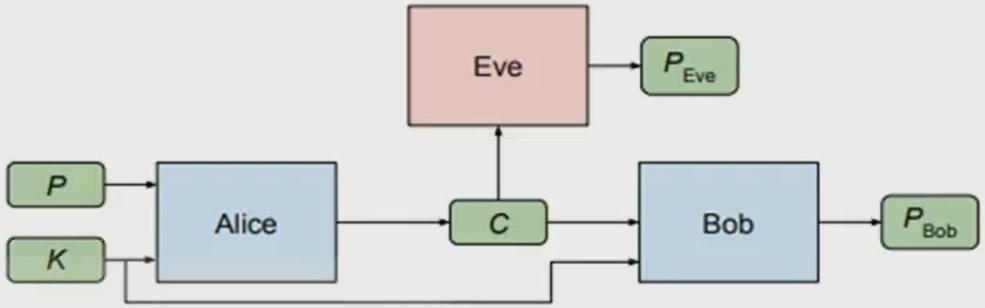
$$MSE = \frac{1}{N}\sum_{i=1}^{N}(P_i - P_i')^2$$

Jugando con nuevos modelos...

- Búsqueda:
  - Mejorar la entropía (confusión-difusión)
    - Efecto avalancha, redes permutación, XOR, etc.
    - Test estadísticos / aleatoriedad (NIST SP 800-22)
  - Aumentar dependencia con texto en claro y clave
  - Analizar nuevas funciones de activación

L1 distance $d(P, P') = \Sigma_{i=1,N}|P_i - P'_i|$ where N is the length of plaintexts.

$$L_E(\theta_A, \theta_E, P, K) = d(P, E(\theta_E, A(\theta_A, P, K)))$$



Jugando con nuevos modelos...

NAVAJA NEGRA CONFERENCE

Nuevas ideas

$$L_{Entropia} = \frac{1}{N}\sum_{i=1}^{N} | C_i - C'_i |$$

$$L_{Bob} = \frac{1}{N}\sum_{i=1}^{N} | P_i - P_{iBob} |$$

$$L_{Eva} = \frac{1}{N}\sum_{i=1}^{N} | P_i - P_{iEva} |$$

$$L_{Alicia} = L_{Bob} + (1 - L_{Eva}) + \sigma(1 - L_{Entropia})$$

Sigma

- $L_{Entropía}$ – Distancia entre textos cifrados por Alicia
- L_{Bob} usa una función de pérdida individual (no comparte con Alicia)
- L_{Alicia} vitaminada con entropía
- Sigma -> estabilizador de entrenamiento para minimizar mínimo local
- Nuevas funciones de activación (ReLu – Sigmoide) y se descarta funciones cuadráticas (-1,1 a 0,1)

Problemas de MINIMO LOCAL y Convergencia del modelo

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(P_i - P'_i)^2$$

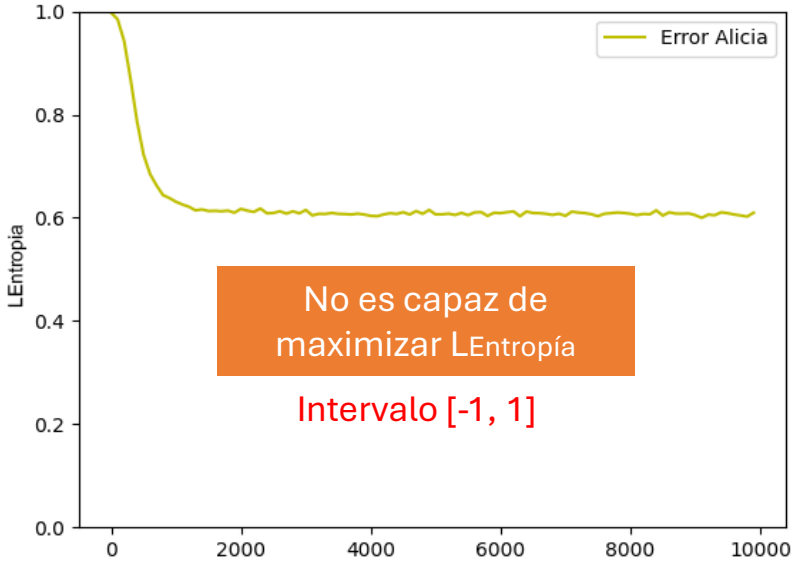
NOTA: **Conseguir el modelo entrenado puede tardar más o menos tiempo pero una vez entrenado el modelo el cifrado/descifrado es en "tiempo real"....**

Cambiando funciones de activación...

NAVAJA NEGRA CONFERENCE

Paper Google

Sigmoide (capa oculta) – Tanh (capa salida)



No es capaz de maximizar LEntropía

Intervalo [-1, 1]

$$L_{Entropia} = \frac{1}{N}\sum_{i=1}^{N} | C_i - C'_i |$$

$$L_{Bob} = \frac{1}{N}\sum_{i=1}^{N} | P_i - P_{iBob} |$$

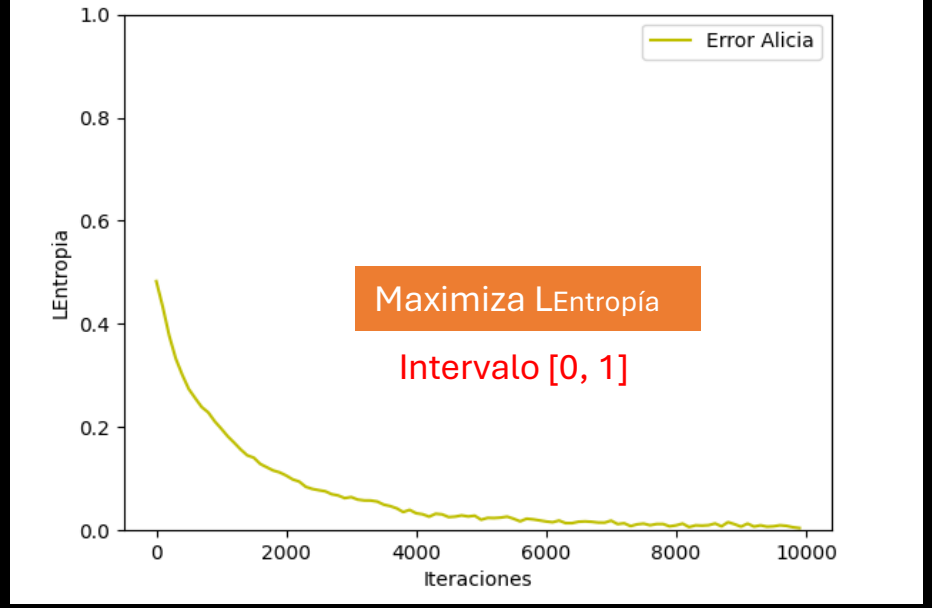
$$L_{Eva} = \frac{1}{N}\sum_{i=1}^{N} | P_i - P_{iEva} |$$

$$L_{Alicia} = L_{Bob} + (1 - L_{Eva}) + \sigma(1 - L_{Entropia})$$

Lentropia=1 bits idénticos
0 -> 50% de bits diferentes
0.6 -> cambio significativo 25% bits

Nuevas ideas

ReLU (capa oculta) - Sigmoide (capa salida)



Maximiza LEntropía

Intervalo [0, 1]

Ejemplo plaintext, key, ciphertext:

$$P = \begin{bmatrix} -1 & 1 & -1 & -1 \\ -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \quad K = \begin{bmatrix} 1 & -1 & 1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 0{,}707 & -0{,}475 & 0{,}725 & -0{,}147 \\ 0{,}688 & -0{,}134 & -0{,}515 & 0{,}581 \\ 0{,}673 & 0{,}620 & 0{,}289 & -0{,}634 \\ -0{,}548 & -0{,}615 & -0{,}627 & -0{,}707 \end{bmatrix}$$

Ejemplo plaintext, key, ciphertext:

$$P = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad K = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Experimentación - funciones de activación...

Dependencia texto claro/clave criptográfica con texto cifrado

Ejemplo plaintext, key, ciphertext:

$$P = \begin{bmatrix} -1 & 1 & -1 & -1 \\ -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \quad K = \begin{bmatrix} 1 & -1 & 1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 0{,}707 & -0{,}475 & 0{,}725 & -0{,}147 \\ 0{,}688 & -0{,}134 & -0{,}515 & 0{,}581 \\ 0{,}673 & 0{,}620 & 0{,}289 & -0{,}634 \\ -0{,}548 & -0{,}615 & -0{,}627 & -0{,}707 \end{bmatrix}$$

Ejemplo plaintext, key, ciphertext:

$$P = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad K = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

16 bits de entrada (65536 textos de entrada diferente) PARA UNA MISMA CLAVE CRIPTOGRAFICA

- 50% bits cambiados en la salida para dos textos aleatorios con la misma clave
- Algún ejemplo de clave concreta: 50% de los bits del texto en claro y del texto cifrado coinciden en valor-posición ¿Alguien se acuerda de las claves débiles y semi-débiles del algoritmo DES? :)
- Diferencia de bits entre dos bloques de salida considerando dos bloques de entrada que difieren 1 bit
  - Cambia 1-3 bits de 16 bits (12% cambio / "efecto avalancha")

Calidad del cifrado...

Difusión-Confusión, efecto avalancha

- Entropía y calidad estadística - NIST SP 800-22 A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-22r1a.pdf

1. **Frequency (Monobit) Test**: Evalúa la proporción de ceros y unos en la secuencia para comprobar si el número de unos es aproximadamente el mismo que el de ceros.
2. **Frequency Test within a Block**: Divide la secuencia en bloques y verifica si la frecuencia de un dentro de cada bloque es consistente con lo esperado en una secuencia aleatoria.
3. **Runs Test**: Analiza la longitud de secuencias consecutivas de ceros o unos para verificar si el número de cambios de ceros a unos es acorde con una secuencia aleatoria.
4. **Test for the Longest Run of Ones in a Block**: Evalúa la longitud de la cadena más larga de un dentro de bloques de longitud fija.
5. **Binary Matrix Rank Test**: Mide el rango de matrices binarias generadas a partir de la secuenci lo que da una indicación de la aleatoriedad estructural.
6. **Discrete Fourier Transform (Spectral) Test**: Realiza un análisis de Fourier para detectar la presencia de patrones cíclicos en la secuencia.
7. **Non-overlapping Template Matching Test**: Busca plantillas específicas (subsecuencias) en la secuencia sin superposición y mide la cantidad de ocurrencias.
8. **Overlapping Template Matching Test**: Similar a la prueba anterior, pero esta vez permite la superposición de las plantillas en la secuencia.
9. **Maurer's Universal Statistical Test**: Detecta si una secuencia puede comprimirse significativamente, lo cual indicaría una falta de aleatoriedad.
10. **Linear Complexity Test**: Evalúa la complejidad lineal de la secuencia (la longitud del LFSR - registro de desplazamiento de retroalimentación lineal- más corto que puede generar la secuencia).
11. **Serial Test**: Examina la frecuencia de todos los posibles patrones de longitud m dentro de la secuencia.

Para cada CLAVE generamos todos los textos en claro de 16 bits, 65536 textos (+1 millón de bits por clave):

Clave 0000000000000000: Test ok 1/15
Clave 1111111111111111: Test ok 1/15
Clave 1010101010101010: Test ok 3/15
Clave 0101010101010101: Test ok 3/15
Clave 1111111100000000: Test ok 3/15
Clave 0000000011111111: Test ok 3/15

"Pasan los test monobits y runs, ambos relacionados con la distribución de los bits...."

nistrng 1.2.3

```
pip install nistrng
```

Conclusiones

"Machine intelligence is the last invention that humanity will ever need to make.."

Nick Bostrom

NAVAJA NEGRA CONFERENCE

Redes Neuronales en la Criptografía Moderna

Dr. Alfonso Muñoz
Telegram: @criptored Twitter: @mindcrypt
alfonso@criptored.com - https://es.linkedin.com/in/alfonsomuñoz

D. David Ramírez
Telegram: @daysapro Twitter: @daysapro
https://www.linkedin.com/in/david-ramírez-acero-3bb282266/