



# Deep Learning Aplicado al procesamiento de Imágenes

## Día 2 – Implementando Deep Learning

**Dr. Rodrigo Salas Fuentes**  
[rodrigo.salas@uv.cl](mailto:rodrigo.salas@uv.cl)



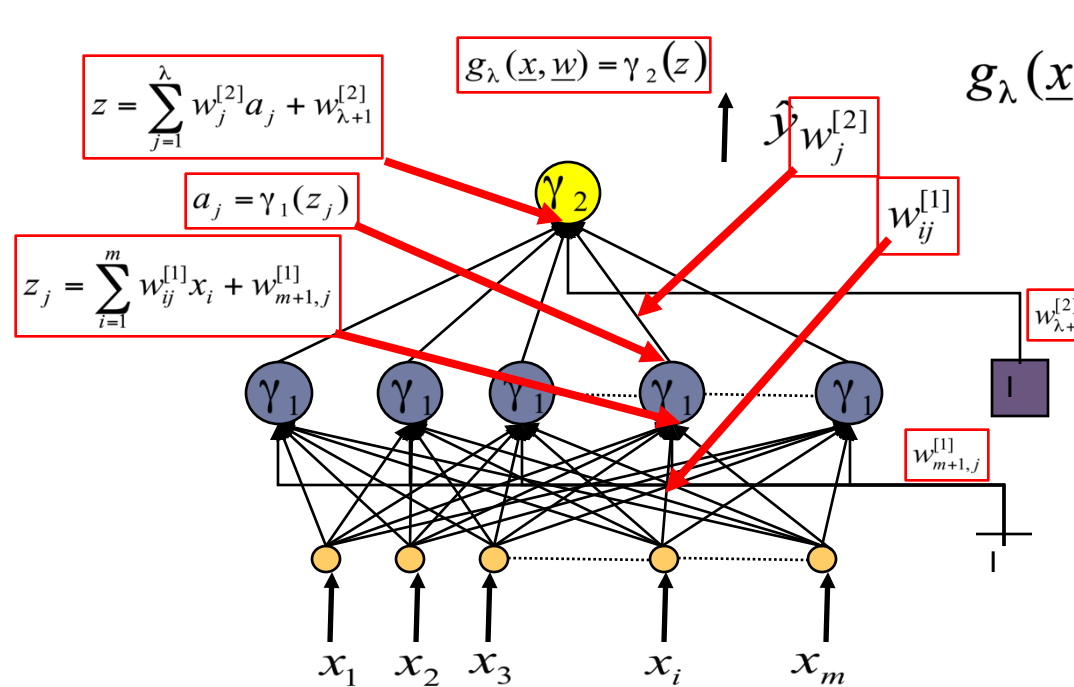
# El Perceptrón Multicapa

# El Perceptrón Multicapa (MLP)

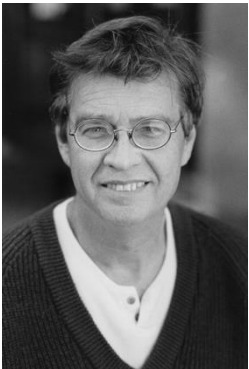
Output  
Layer

Hidden  
Layer

Input Layer

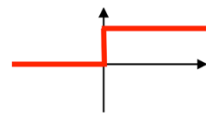
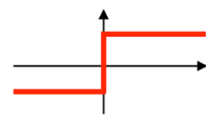
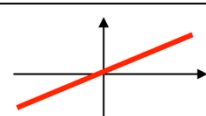
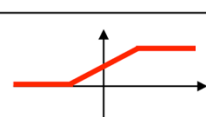
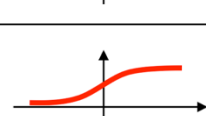
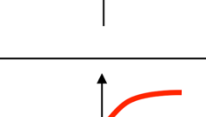
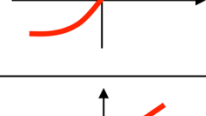
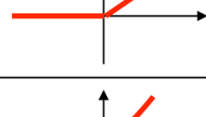


$$g_{\lambda}(\underline{x}, \underline{w}) = \gamma_2 \left( \sum_{j=1}^{\lambda} w_j^{[2]} \gamma_1 \left( \sum_{i=1}^m w_{ij}^{[1]} x_i + w_{m+1,j}^{[1]} \right) + w_{\lambda+1}^{[2]} \right)$$



Rumelhart, David E., Geoffrey E. Hinton, and R. J. Williams. "Learning Internal Representations by Error Propagation", Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1: Foundations. MIT Press, 1986

# Funciones de Activación

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	Multi-layer Neural Networks	
Rectifier, ReLU (Rectified Linear Unit)	$\phi(z) = \max(0, z)$	Multi-layer Neural Networks	
Rectifier, softplus	$\phi(z) = \ln(1 + e^z)$	Multi-layer Neural Networks	

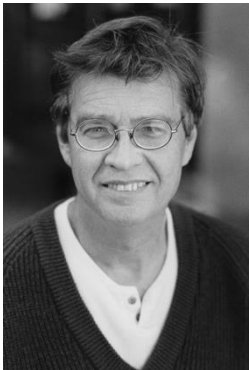
Copyright © Sebastian Raschka 2016  
(<http://sebastianraschka.com>)

# Retropropagación del Error (Backpropagation)

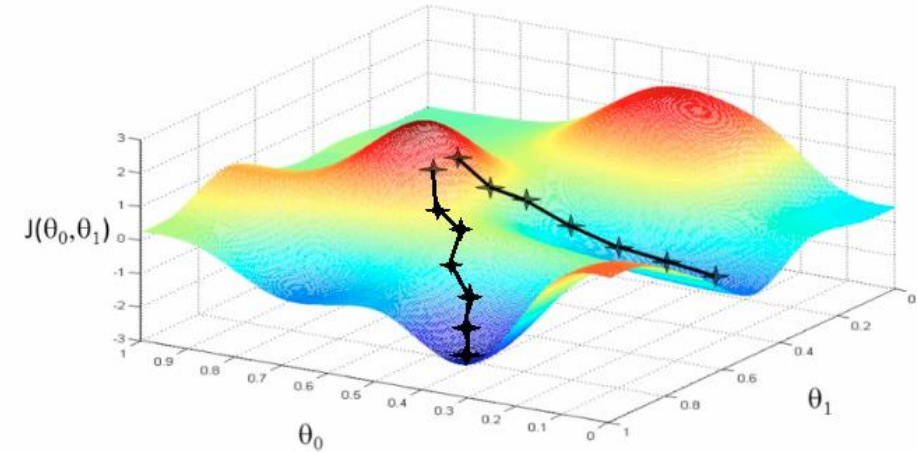
$$\mathbf{w}(t+1) = \mathbf{w}(t) - \alpha(t) \nabla_{\mathbf{w}} l(h(\cdot, \mathbf{w}); \mathbf{z}_t)$$



**Paul Werbos.**  
1974 | In his Harvard PhD thesis, Paul Werbos describes training neural networks through backpropagation.



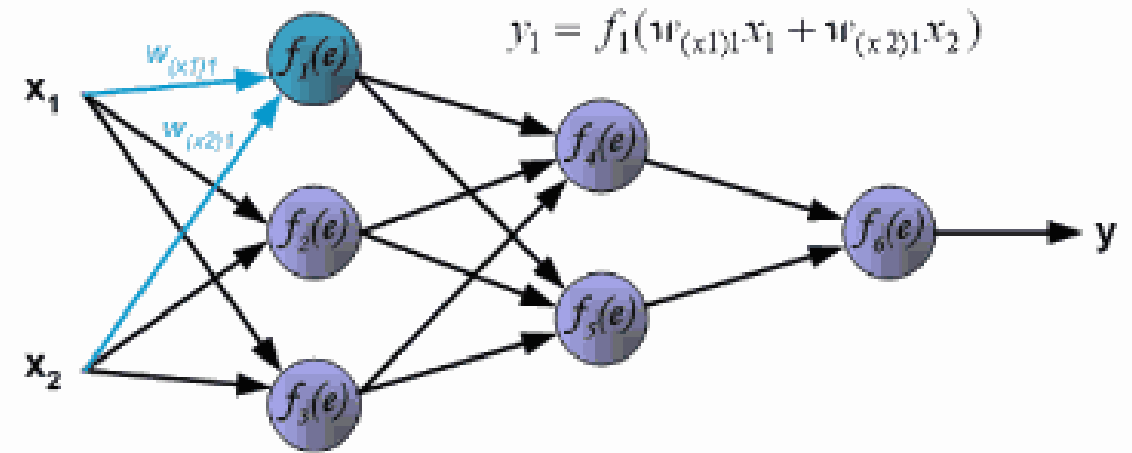
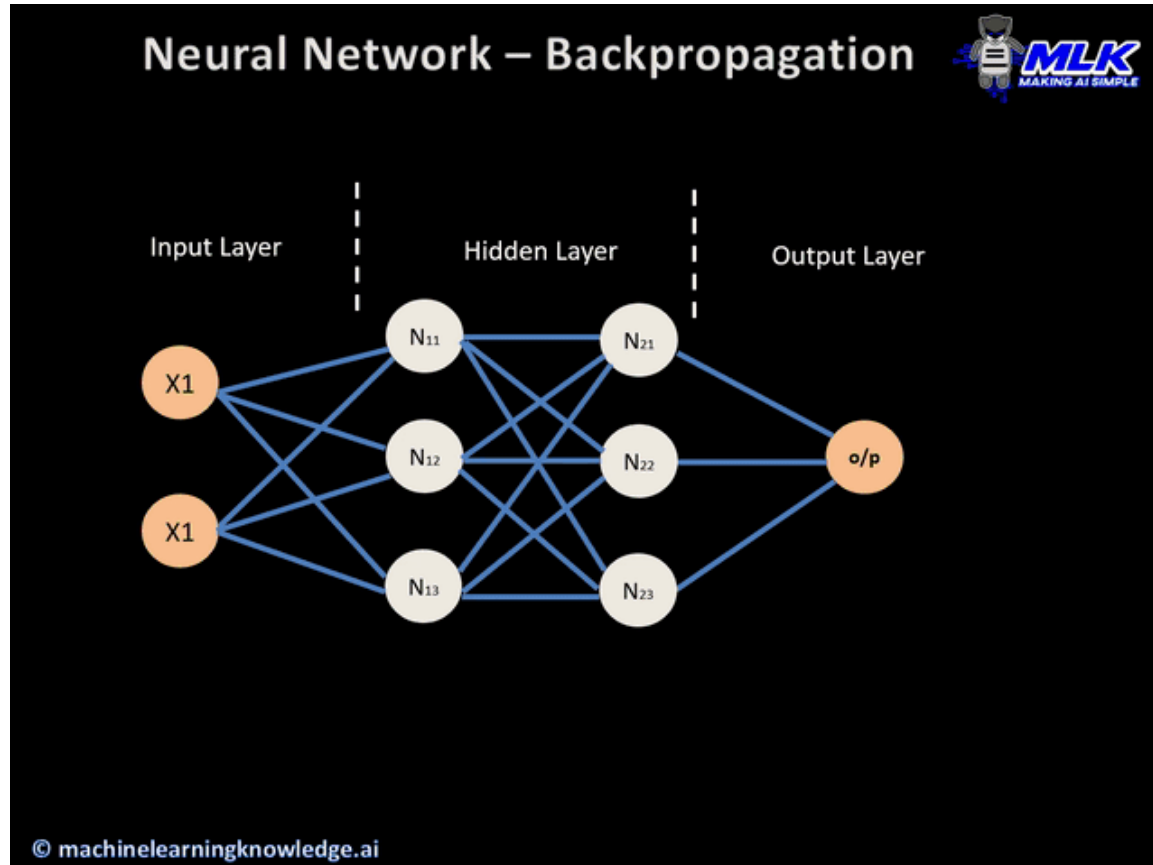
Rumelhart, David E., Geoffrey E. Hinton, and R. J. Williams. "Learning Internal Representations by Error Propagation", Parallel distributed processing: Explorations in the microstructure of cognition, Volume I: Foundations. MIT Press, 1986



## Algoritmo de Aprendizaje: Backpropagation

- El algoritmo de **retropropagación** o **backpropagation** aprovecha la regla de la cadena del cálculo diferencial, que calcula los gradientes de error en términos de sumas de productos de gradiente local en las diversas rutas desde un nodo hasta la salida.
- Tiene 2 fases:
  - **Fase de avance o inferencia:**
    - Se ejecuta el perceptrón multicapa para una entrada dada y se computan todos los resultados de salidas de cada nodo y sus derivadas hasta el nodo de salida.
  - **Fase de retropropagación o retroceso:**
    - El objetivo principal de la fase hacia atrás es aprender el gradiente de la función de pérdida con respecto a los diferentes pesos mediante el uso de la regla de la cadena del cálculo diferencial.
    - Estos gradientes se utilizan para actualizar los pesos. Los gradientes se aprenden en la dirección hacia atrás, comenzando desde el nodo de salida.

# Retropropagación del Error (Backpropagation)

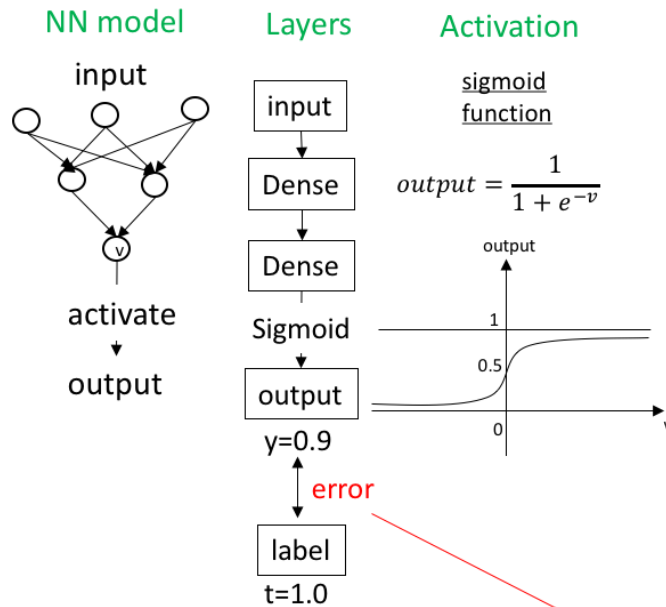


<https://medium.com/deeper-deep-learning-tr/ad1m-ad1m-forward-and-back-propagation-cf4cd18276ee>

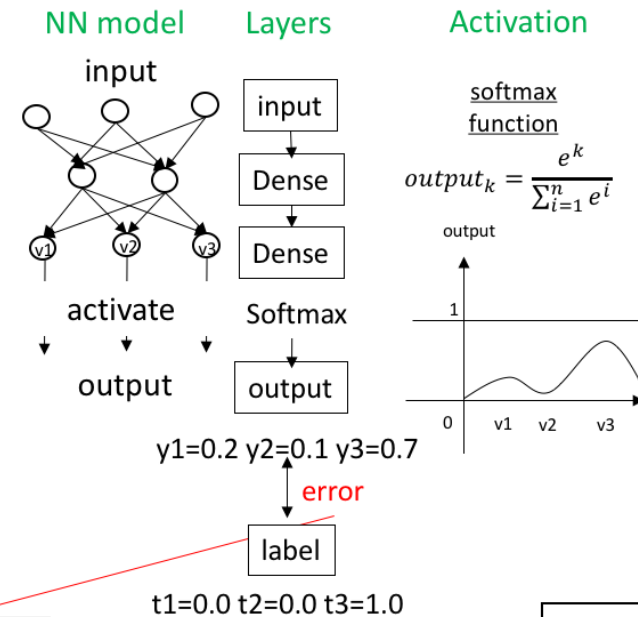


# Funciones de Pérdida

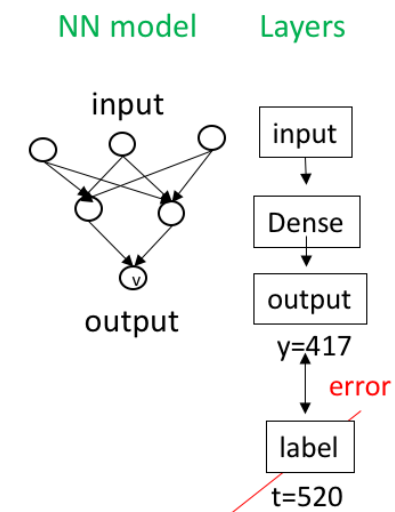
## 1.Binary Classification



## 2.Multiclass Classification



## 3.Regression



Cross Entropy(CE)

$$L = - \sum t_i \log y_i$$

Mean Squared Error(MSE)

$$L = \frac{1}{2} (t - y)^2$$

[https://www.renom.jp/notebooks/tutorial/basic\\_algorithm/lossfunction/notebook.html](https://www.renom.jp/notebooks/tutorial/basic_algorithm/lossfunction/notebook.html)



## Aproximadores Universales

### Theorem of Universal Approximation:

“Every bounded continuous function with bounded support can be approximated arbitrarily closely by a multi-layer perceptron by selecting enough but a finite number of hidden neurons with appropriate transfer function”



**Cybenko., G. (1989) "Approximations by superpositions of sigmoidal functions", Mathematics of Control, Signals, and Systems, 2 (4), 303-314**

**Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. Neural Networks, 2:359--366, 1989**

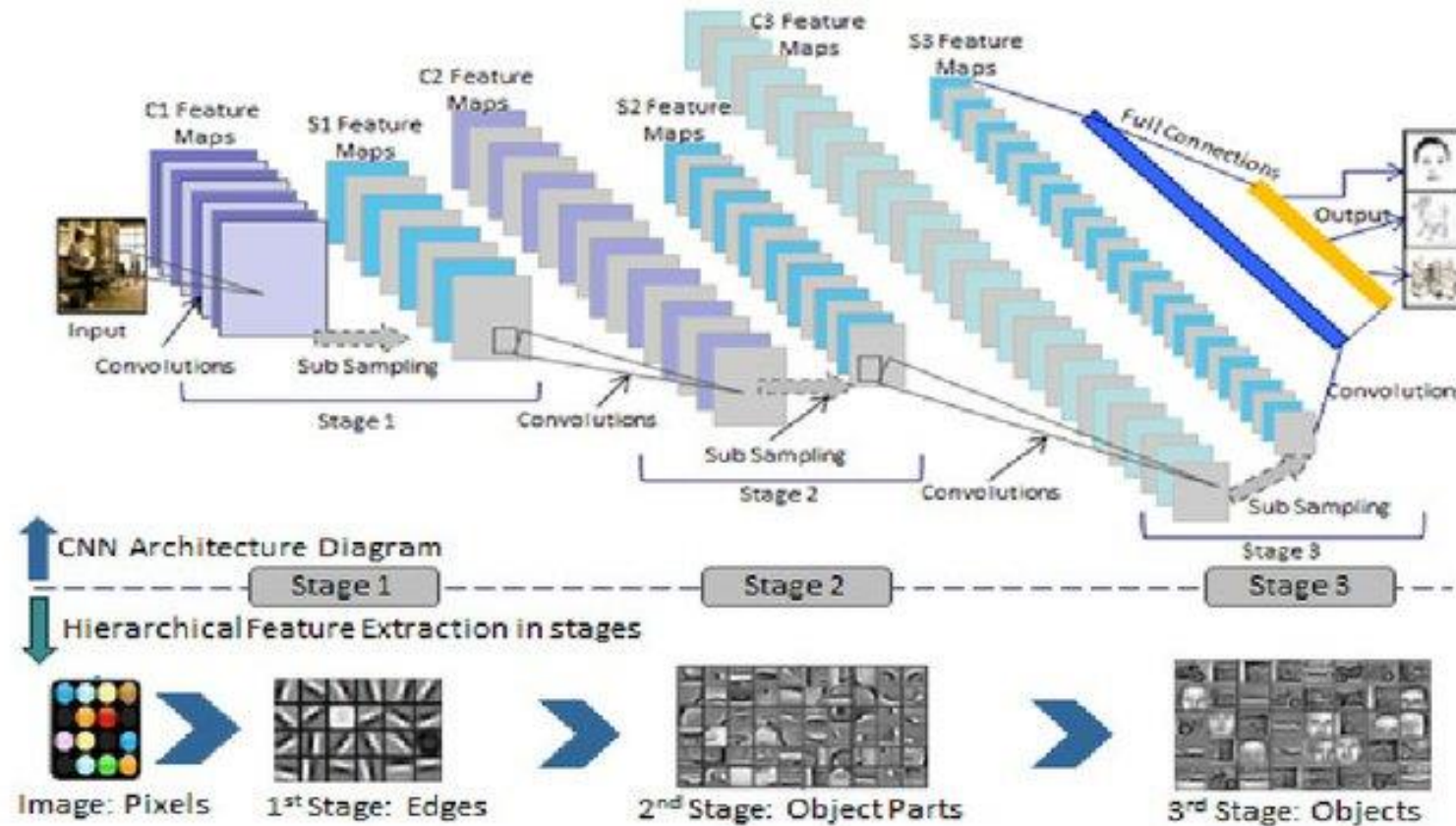
## Taller 1 – MLP para regresión



<https://colab.research.google.com/drive/1yW5la5hxcaDRNtPGAG3dDb4UQ2PeXvz2?usp=sharing>

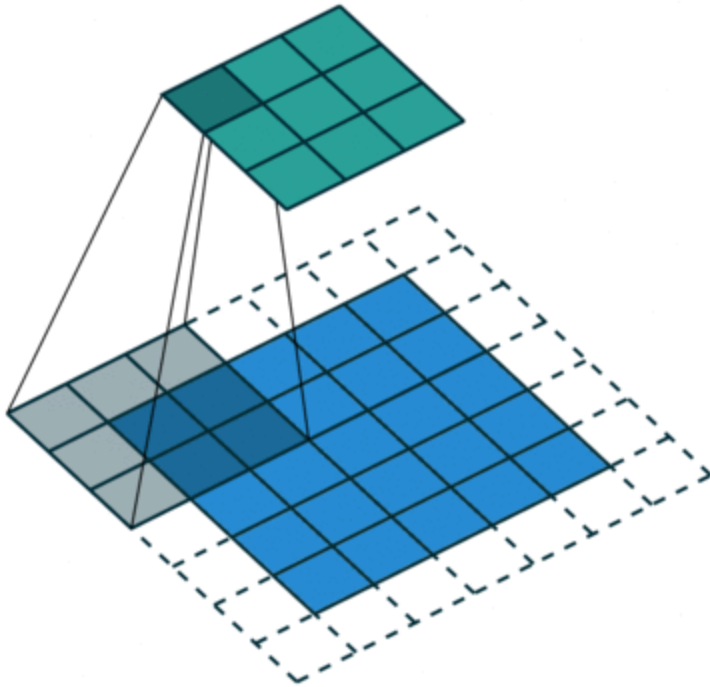
# Red Neuronal Convolucional

# Red Neuronal Convolucional



Katole et al. Hierarchical Deep Learning Architecture For 10K Objects Classification. Doi: 10.5121/csit.2015.51408

# Operaciones convolucionales



1 <sub>x1</sub>	1 <sub>x0</sub>	1 <sub>x1</sub>	0	0
0 <sub>x0</sub>	1 <sub>x1</sub>	1 <sub>x0</sub>	1	0
0 <sub>x1</sub>	0 <sub>x0</sub>	1 <sub>x1</sub>	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved  
Feature

**Kernel:** Pequeño filtro que recorre la imagen y extrae características locales como bordes, texturas o patrones.

**Convolución:** Operación que multiplica y suma los valores del kernel con regiones locales de la imagen para generar un mapa de características.

# Operaciones convolucionales

Input

7	3	5	2
8	7	1	6
4	9	3	9
0	8	4	5

maxpool →

Output

8	6
9	9

**Max-Pooling:** Técnica de reducción de dimensionalidad que conserva solo el valor máximo de regiones locales, manteniendo las características más importantes.

0	0	0	0	0	0	0
0	60	113	56	139	85	0
0	73	121	54	84	128	0
0	131	99	70	129	127	0
0	80	57	115	69	134	0
0	104	126	123	95	130	0
0	0	0	0	0	0	0

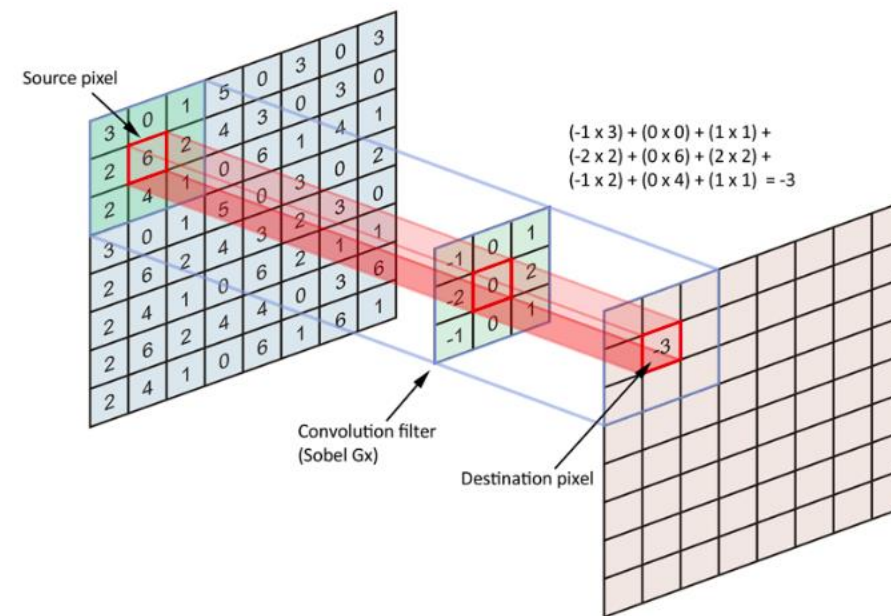
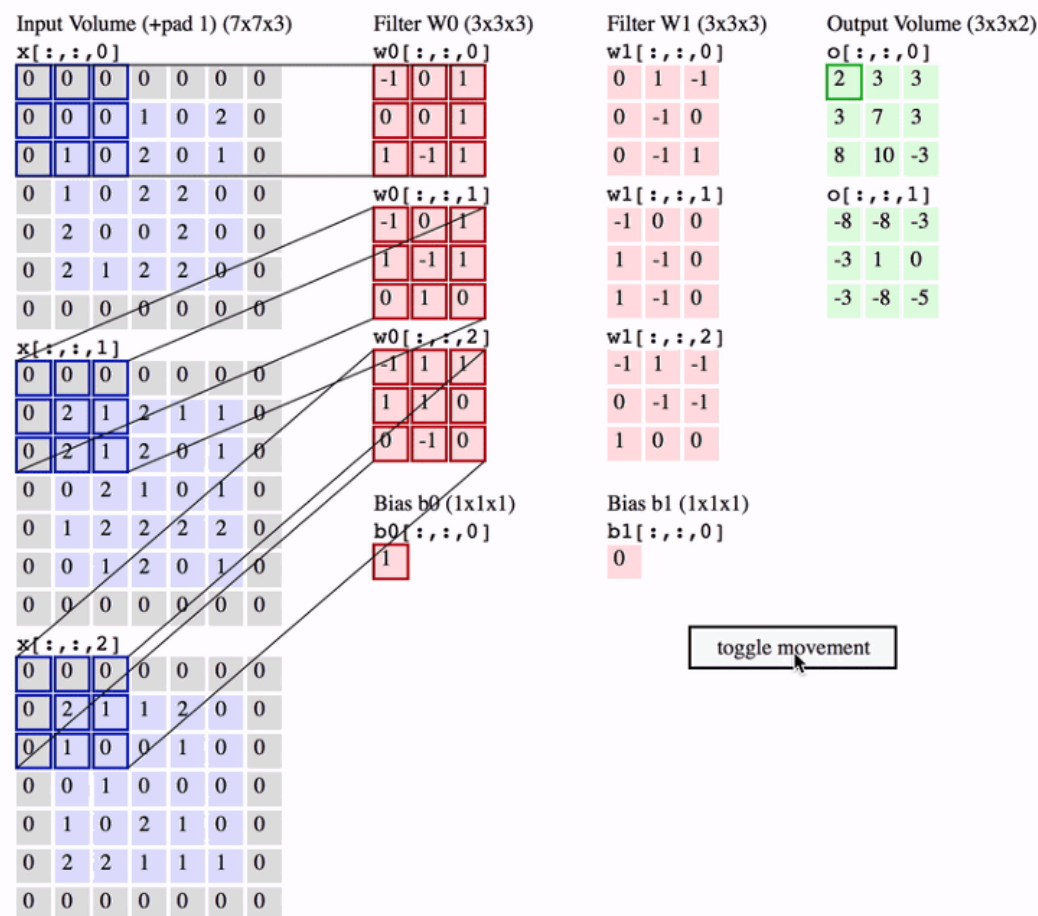
Kernel

0	-1	0
-1	5	-1
0	-1	0

114				

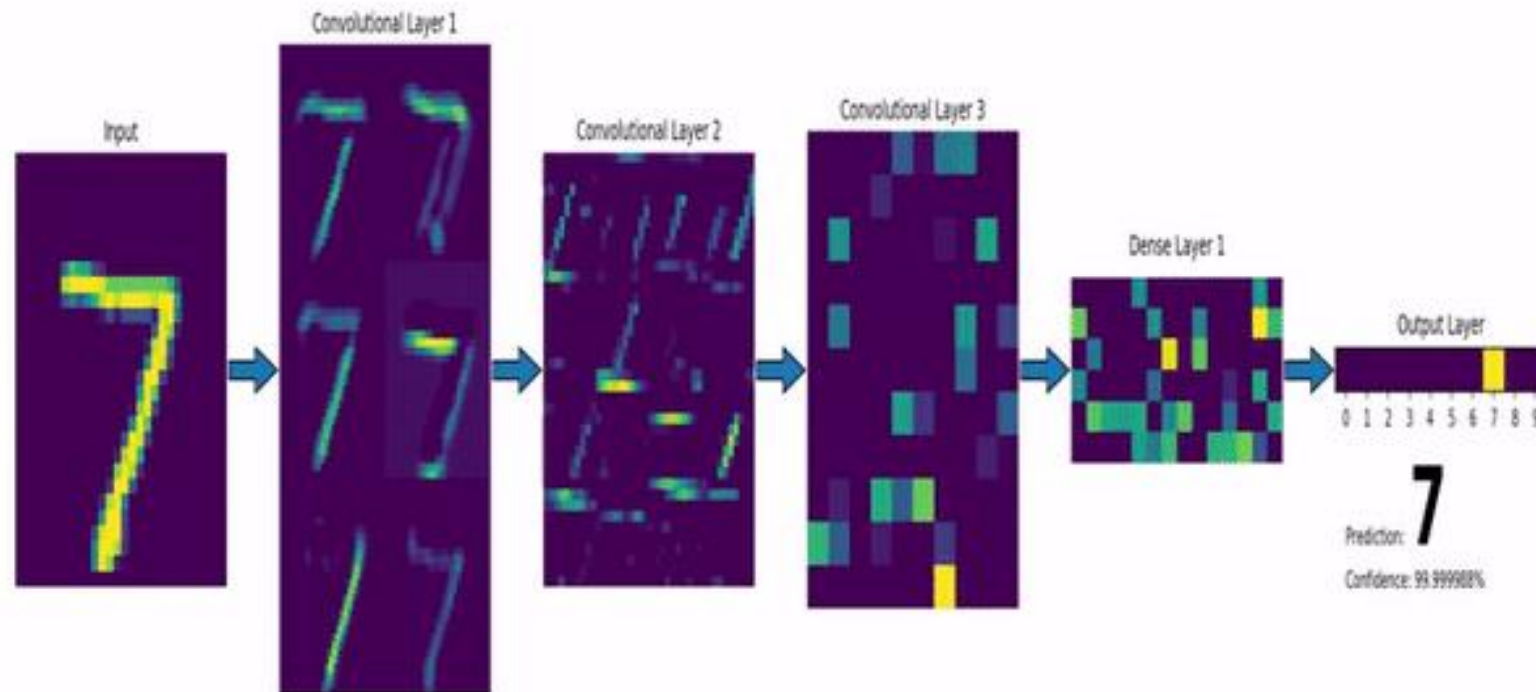
**Padding:** Adición de bordes (normalmente ceros) alrededor de la imagen para mantener el tamaño de salida o evitar pérdida de información en los bordes.

# Capa Convolutiva – Extracción de Características



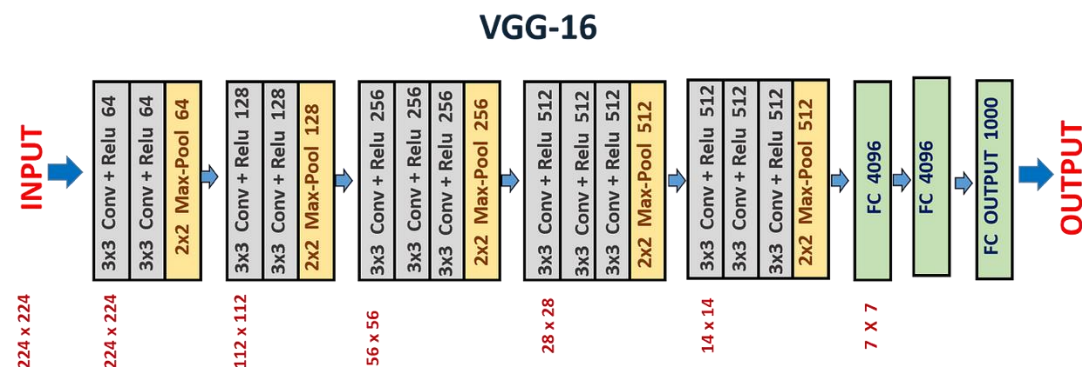
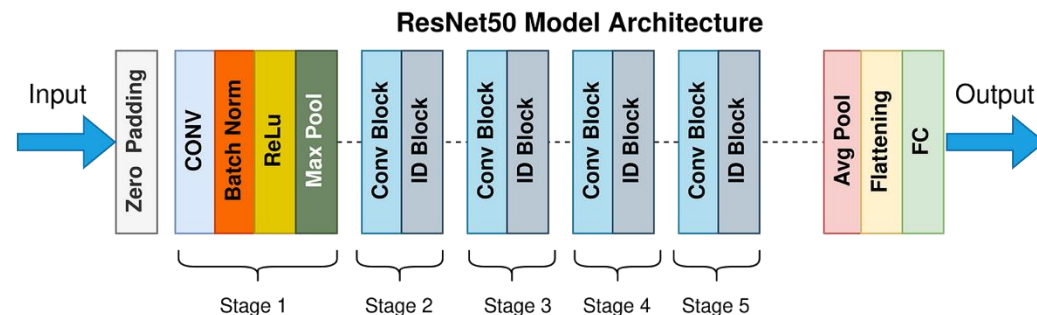


# Visualización de aprendizaje de filtros



[https://www.louisbouchard.ai/densenet-explained/of semantic segmentation, classification and localization, object detection and instance segmentation](https://www.louisbouchard.ai/densenet-explained/of-semantic-segmentation,classification-and-localization,object-detection-and-instance-segmentation) (Li, Johnson and Yeung, 2017)

# Arquitecturas basadas en CNN

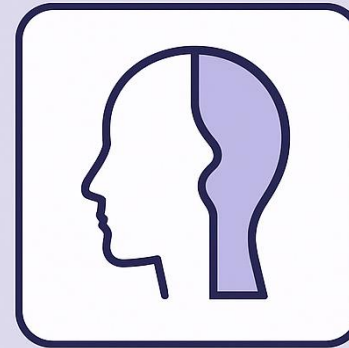


Estas arquitecturas CNN (VGG, ResNet, U-Net) aprenden representaciones visuales jerárquicas y se adaptan a tareas como clasificación y segmentación pixel a pixel.

# Aplicaciones de Deep Learning en imágenes



Classification



Segmentation



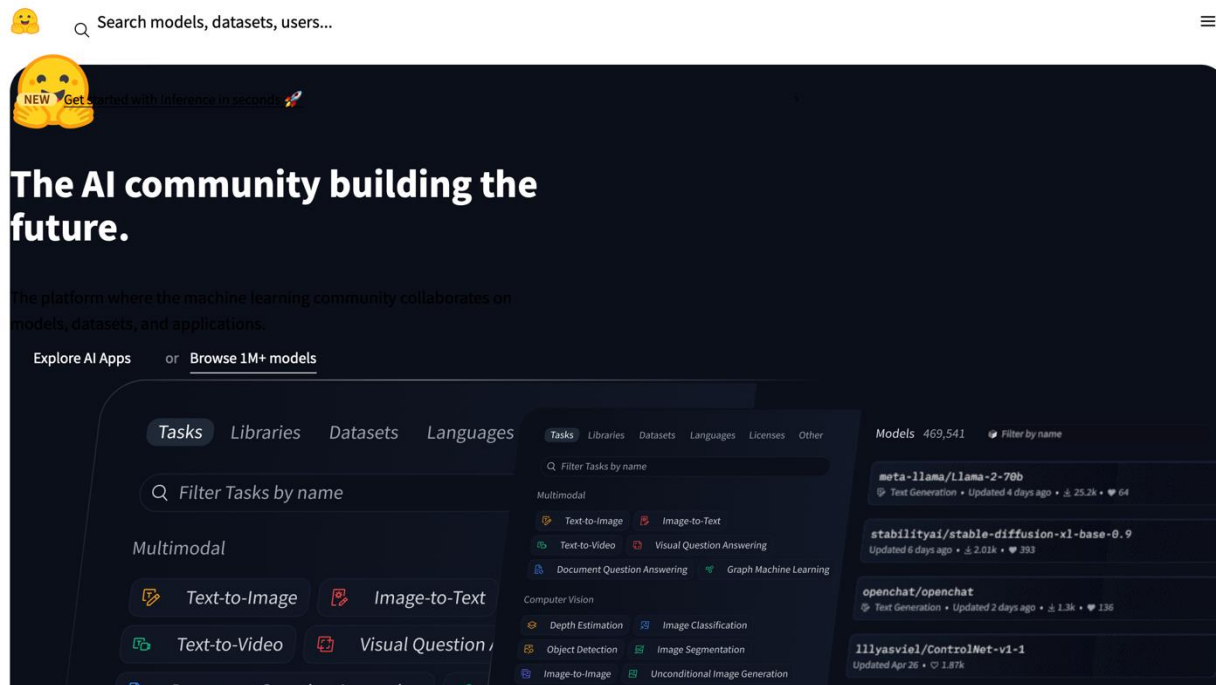
Object detection




## Taller 2 – CNN aplicado a MNIST



<https://colab.research.google.com/drive/18Hr-Bz6W1FcGce67vXwdKviPn8W8IbjI?usp=sharing>

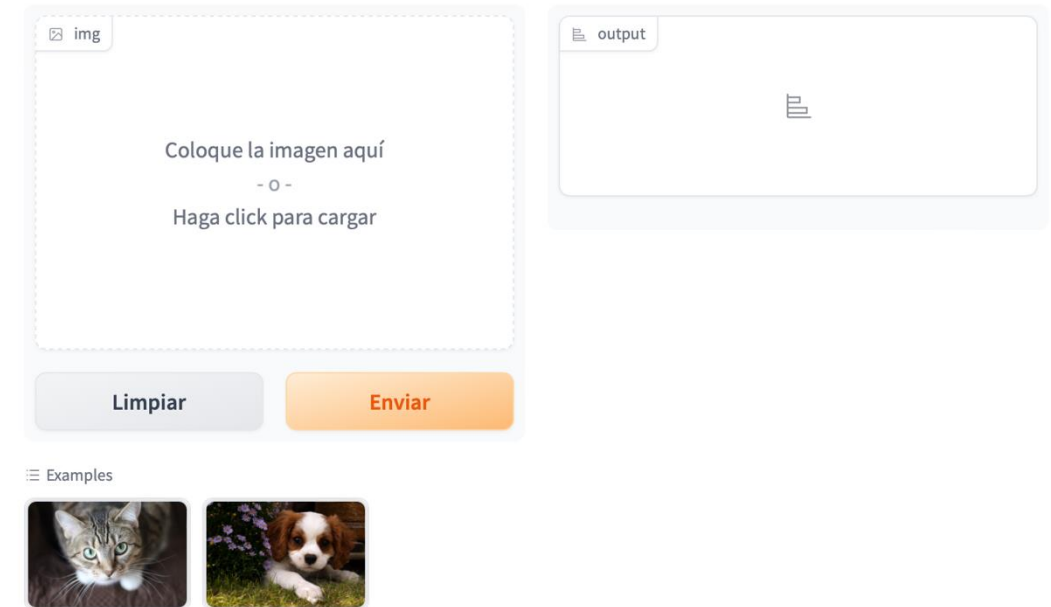
# Hugging Face





 mawady/demo-catsvsdogs-gradio  like 2 ● Running

## CatsVsDogs Classifier

Playground: Inference of Object Classification (Binary) using ResNet18 model and CatsVsDogs dataset. Libraries: PyTorch, Gradio.



[By Dr. Mohamed Elawady](#)

<https://huggingface.co/spaces/mawady/demo-catsvsdogs-gradio>

Dr. Ing. Rodrigo Salas (UV-MEDING-iHealth)

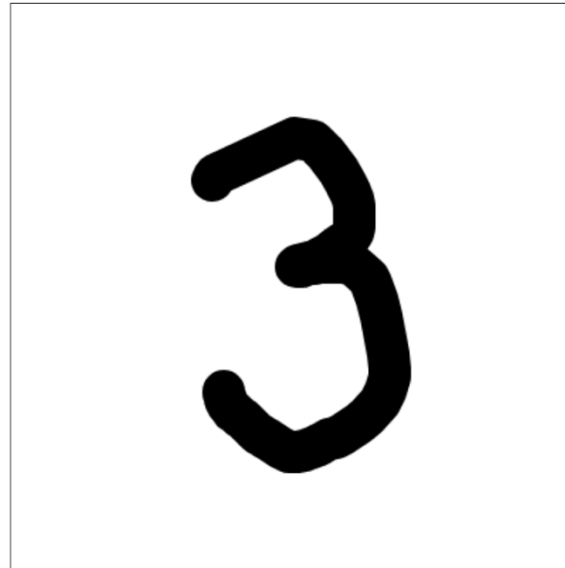
## Taller 3 – CNN aplicado a MNIST de Lenguaje de Señas



<https://www.kaggle.com/code/rodsalasf/simposio-estadistica-taller-3-cnn>

# MNIST Web Demo

## MNIST Web Demo



☒ MNIST Preprocessing

Thinner Black

Thicker Black

White Stroke

Undo Stroke

Reset

Image	0	1	2	3	4	5	6	7	8	9
	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

[https://ufal.mff.cuni.cz/~courses/npfl129/2425/demos/mnist\\_web.html](https://ufal.mff.cuni.cz/~courses/npfl129/2425/demos/mnist_web.html)



# Deep Learning Aplicado a Imágenes Médicas



MSc. Student  
Gabriel Guerra

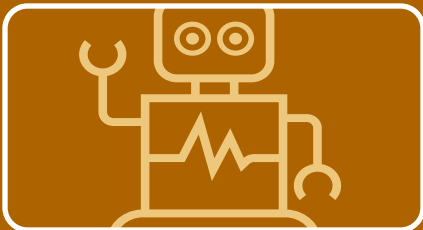
Material desarrollado con el apoyo del estudiante del  
Magister en Ciencias e Ingeniería para la Salud,  
Universidad de Valparaíso  
Sr. Gabriel Guerra

## ¿Por qué se usa Deep learning en salud?



### Mejora la precisión en el diagnóstico

- Identifica patrones sutiles que pueden pasar desapercibidos por el ojo humano
- Mejora la detección temprana de enfermedades como el cáncer



### Automatización y eficiencia clínica

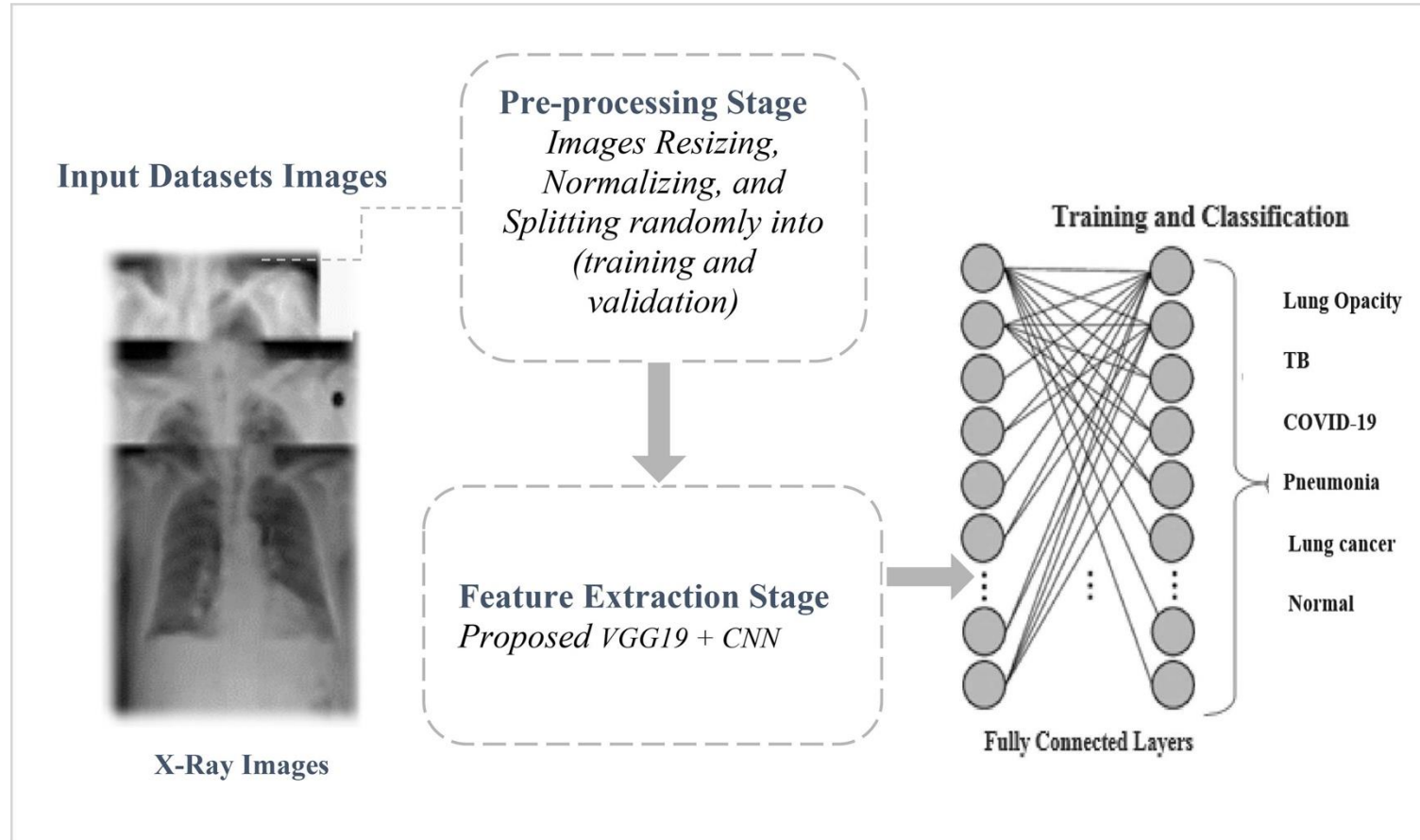
- Ayuda a acelerar el análisis de grandes volúmenes de datos médicos
- Permite reducir la carga operativa en hospitales y centros de salud



### Soporte a decisiones médicas

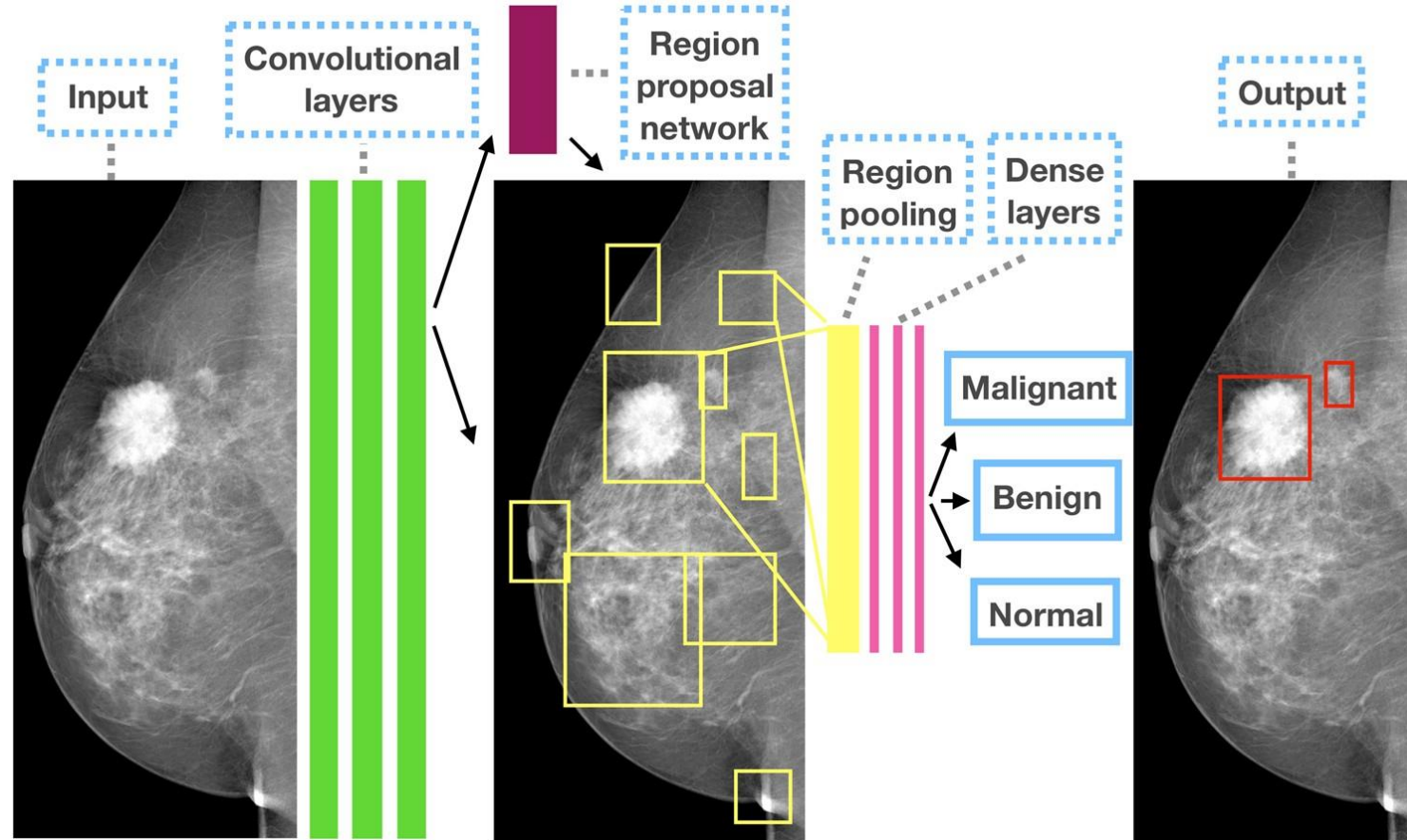
- Provee segundas opiniones basadas en miles de casos similares
- Ayuda en contextos de escasez de especialistas

# Clasificación de enfermedades



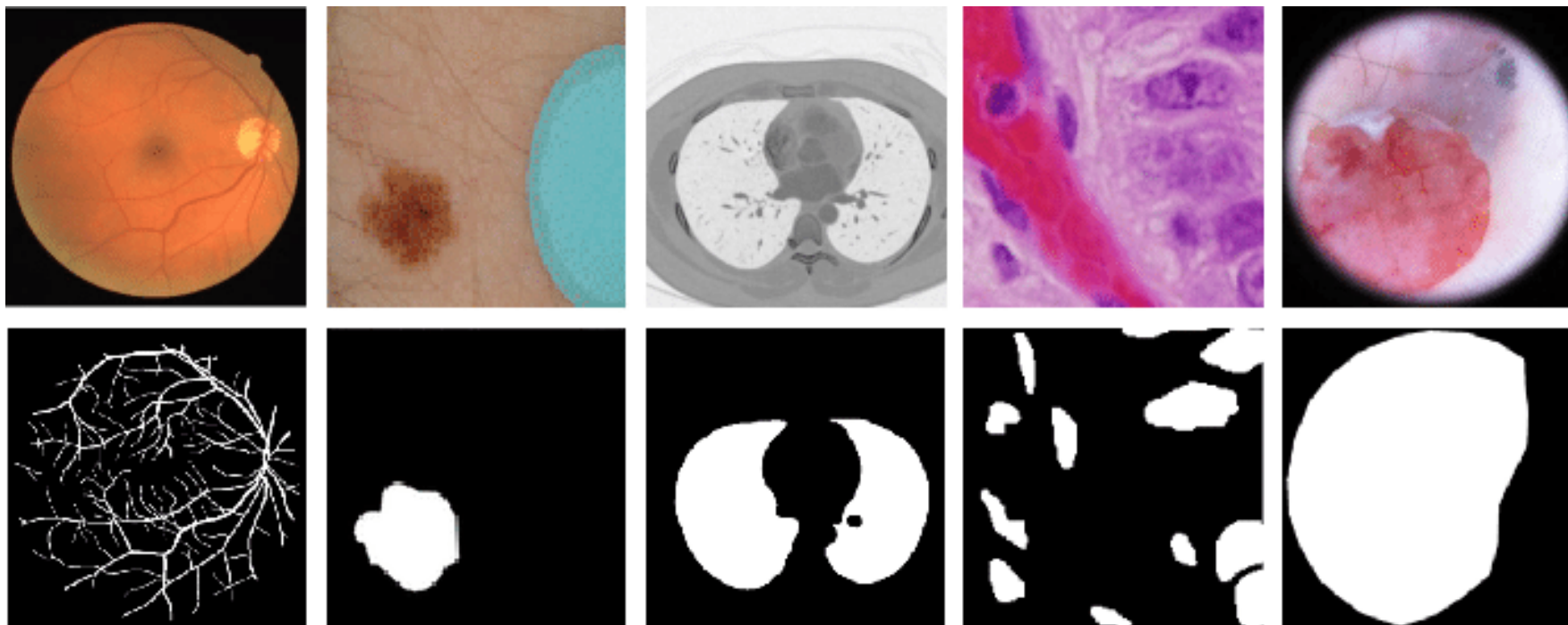
Alshmrani, G. M. M., Ni, Q., Jiang, R., Pervaiz, H., & Elshennawy, N. M. (2023). A deep learning architecture for multi-class lung diseases classification using chest X-ray (CXR) images. Alexandria Engineering Journal, 64, 923–935. <https://doi.org/10.1016/j.aej.2022.10.053>

# Clasificación y detección



Ribli, D., Horváth, A., Unger, Z., Pollner, P., & Csabai, I. (2018). *Detecting and classifying lesions in mammograms with deep learning*. *Scientific Reports*, 8(1), Article 4165. <https://doi.org/10.1038/s41598-018-22437-z>

## Segmentación en Imágenes Médicas



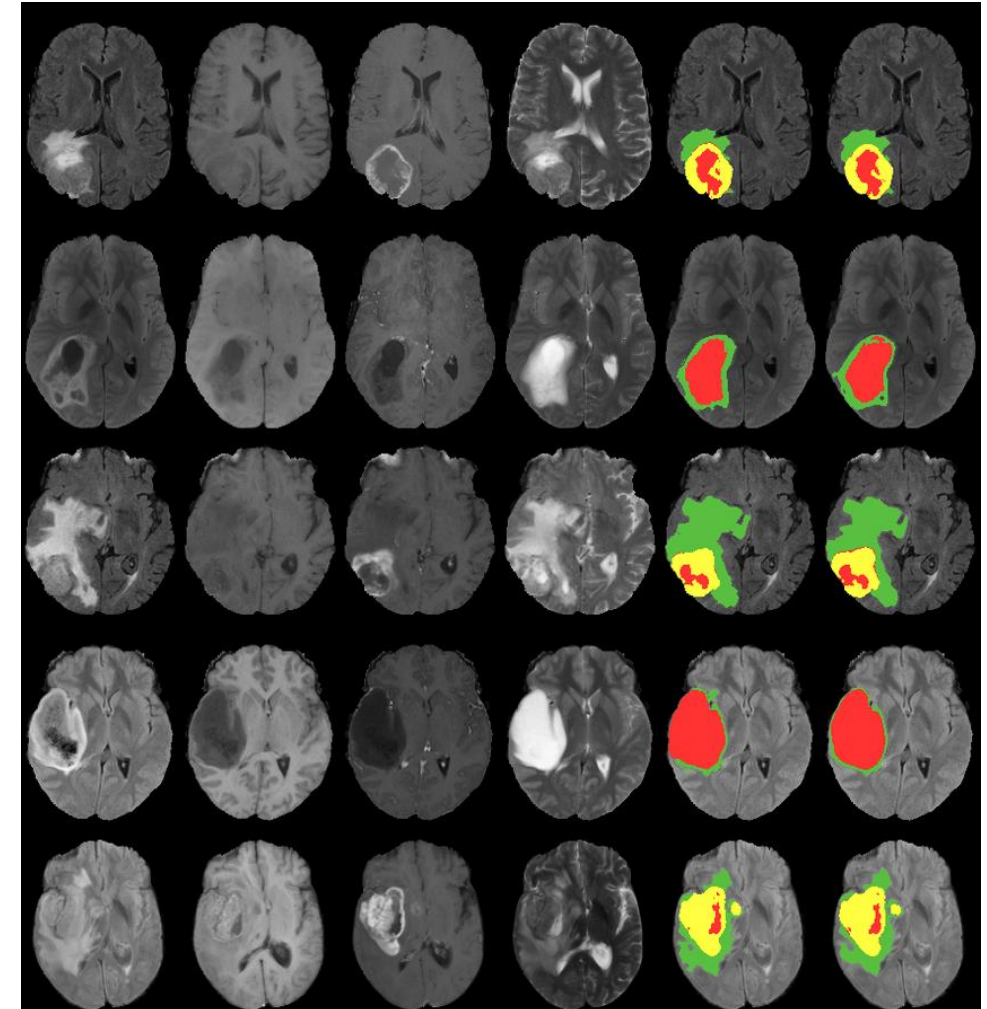
La segmentación médica permite identificar estructuras y regiones de interés en diferentes modalidades de imagen (retina, piel, TAC, histología, etc.), separando píxel a píxel áreas relevantes como lesiones, órganos o tejidos patológicos para apoyar el diagnóstico y tratamiento clínico.



# Segmentación de tumores cerebrales



BraTS es un conjunto de datos público que contiene resonancias magnéticas cerebrales multimodales (T1, T1c, T2, FLAIR) de pacientes con gliomas. Incluye segmentaciones manuales de tres regiones tumorales: tumor completo, tumor realzado y tumor no realzado. Su objetivo es evaluar algoritmos de segmentación automática en contextos clínicos realistas.



# U-NET Aplicado a Segmentación Semántica

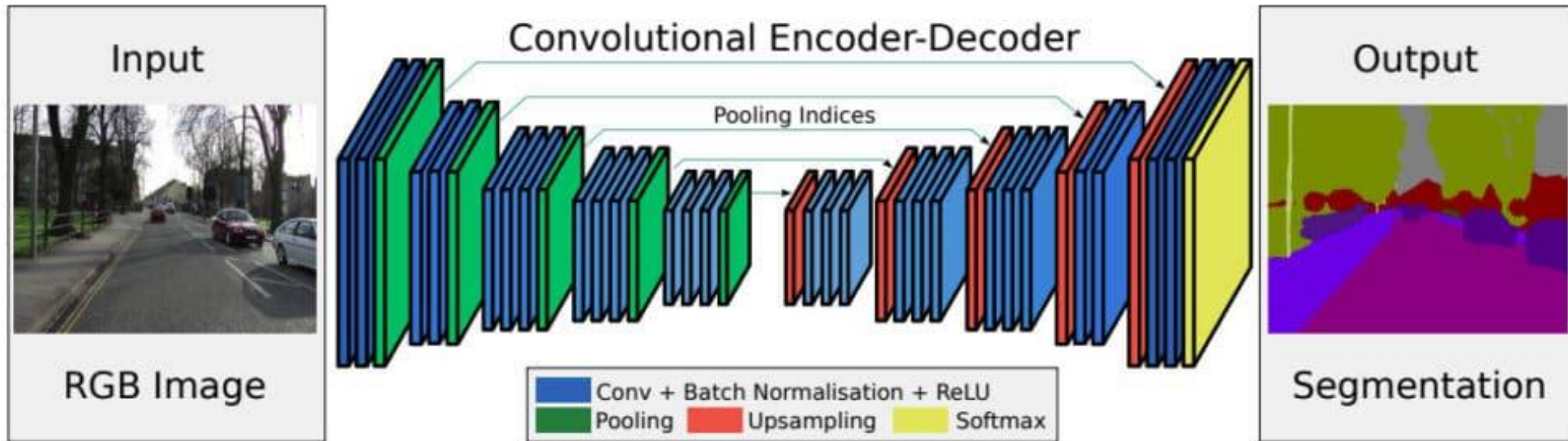


MSc. Student  
Gabriel Guerra

Material desarrollado con el apoyo del estudiante del  
Magister en Ciencias e Ingeniería para la Salud,  
Universidad de Valparaíso  
Sr. Gabriel Guerra



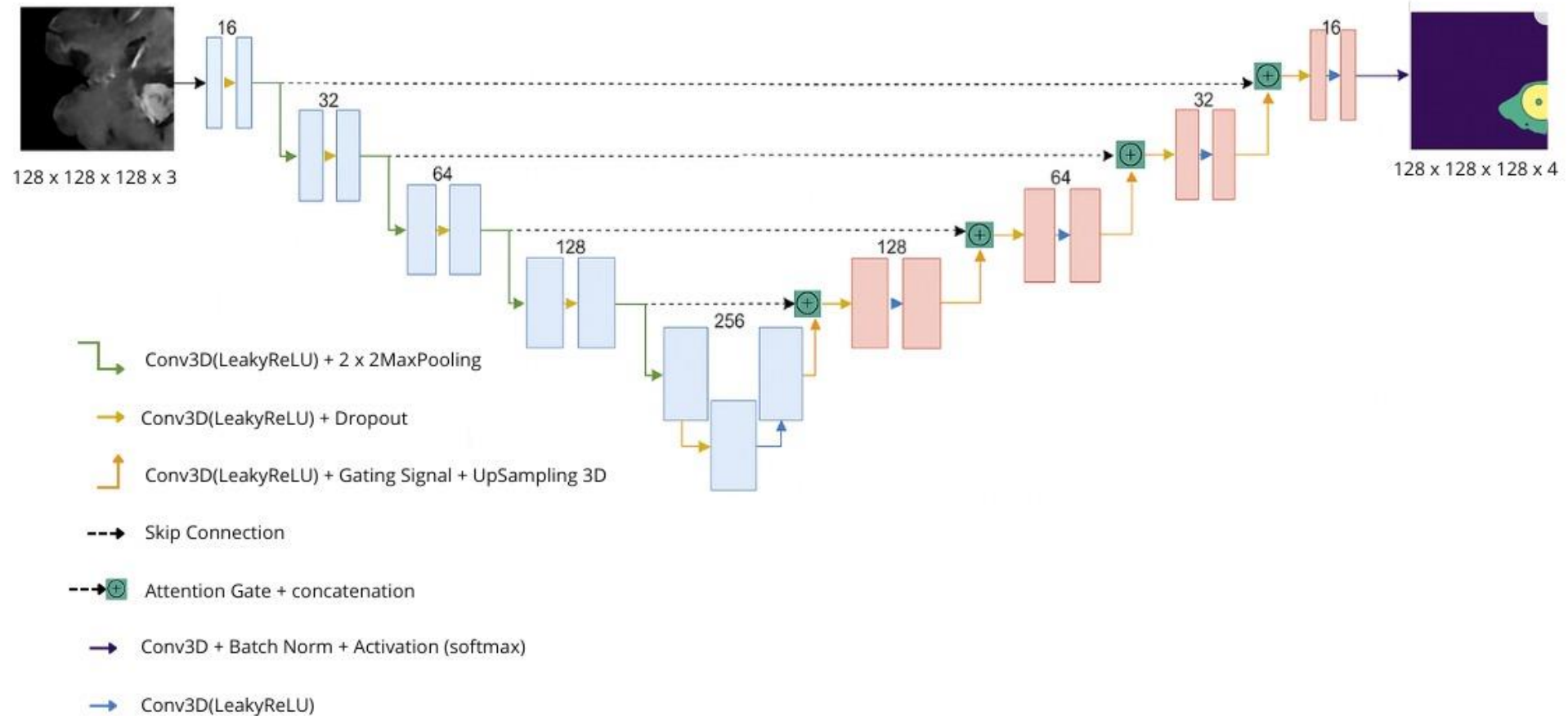
## Segmentación semántica



La **segmentación semántica** asigna una etiqueta a cada píxel de la imagen, identificando qué clase pertenece a cada región (por ejemplo: calle, auto, persona).

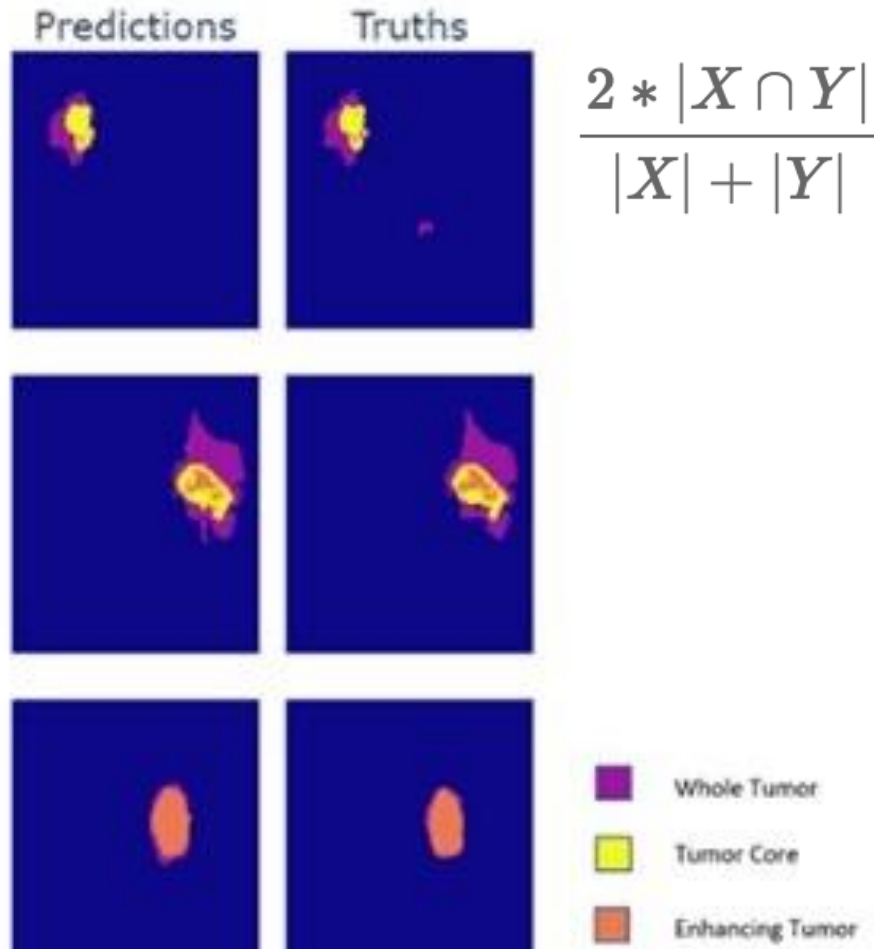
Este proceso se realiza con redes tipo **encoder-decoder**, donde el encoder extrae características y el decoder reconstruye un mapa de clases con la misma resolución que la imagen original.

# Attention U-net

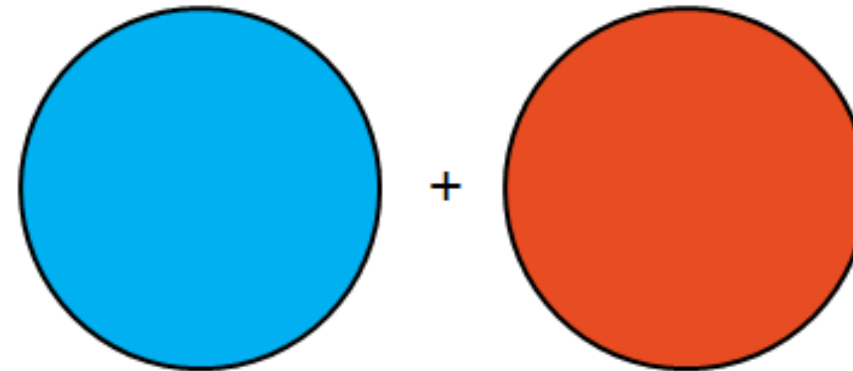
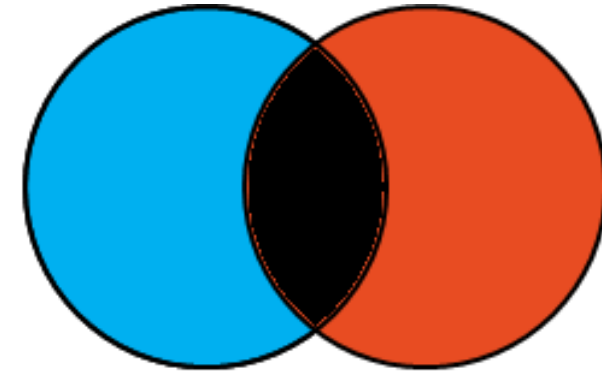


Gitonga, M. M. (2023). *Multiclass MRI Brain Tumor Segmentation using 3D Attention-based U-Net*. arXiv preprint arXiv:2305.06203. <https://arxiv.org/abs/2305.06203>

## Validación: coeficiente de dice



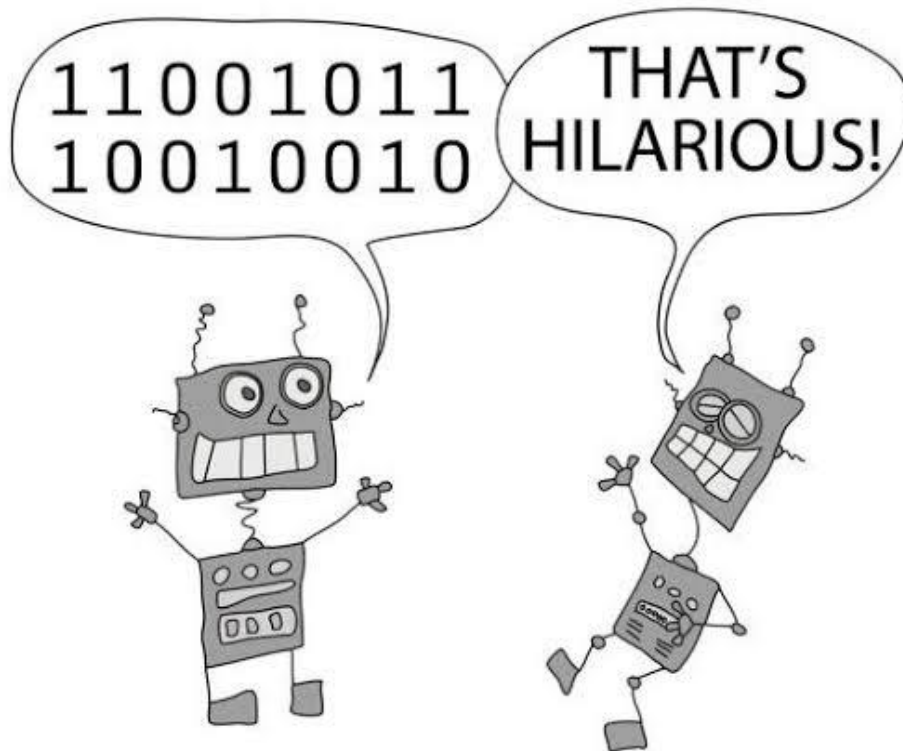
2 x



## Taller 4 – Segmentación Semántica de Tumores





<https://colab.research.google.com/drive/1F5xrndrCGmRqiyZbsMDymTnrrHXWjcV8?usp=sharing>

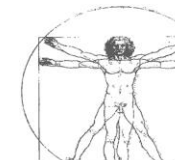


**Dr. Rodrigo Salas  
Fuentes**  
[rodrigo.salas@uv.cl](mailto:rodrigo.salas@uv.cl)



## Thanks for your Attention

-  LinkedIn: [linkedin.com/in/rodrigo-salas-fuentes/](https://www.linkedin.com/in/rodrigo-salas-fuentes/)
-  Google Scholar: [scholar.google.com/citations?user=ZaqDIPcAAAAJ](https://scholar.google.com/citations?user=ZaqDIPcAAAAJ)
-  ORCID: [orcid.org/0000-0002-0350-6811](https://orcid.org/0000-0002-0350-6811)
-  Email: [rodrigo.salas@uv.cl](mailto:rodrigo.salas@uv.cl)



Ingeniería Civil Biomédica  
Facultad de Ingeniería  
Universidad de Valparaíso  
[www.biomedica.uv.cl](http://www.biomedica.uv.cl)