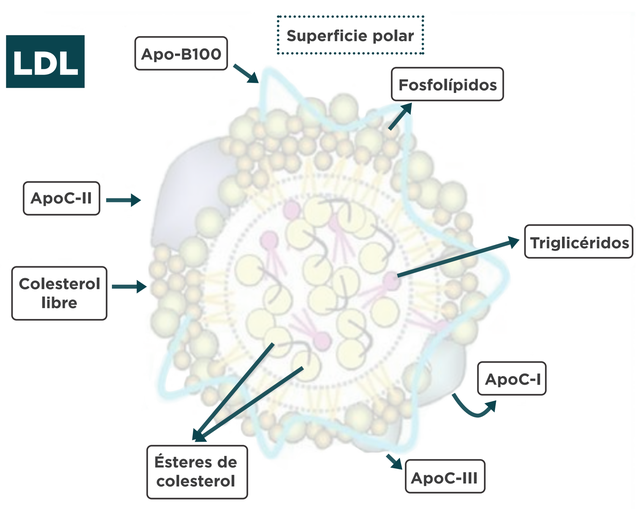
**Salomón Martín Pérez[1], Teresa Arrobas Velilla[1], Isabel Rodríguez Martín[2], Antonio León-Justel[1]**

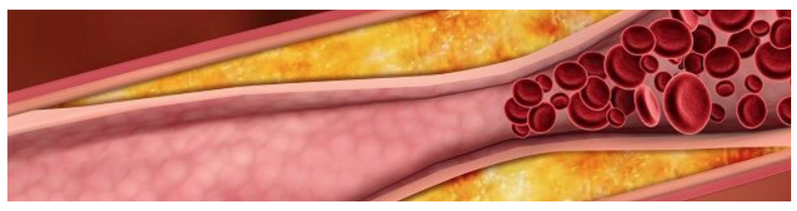
[1] Hospital Universitario Virgen Macarena, Sevilla
[2] Hospital Universitario Virgen del Rocío, Sevilla

2023 What's Next IN LIPIDS & CV PREVENTION

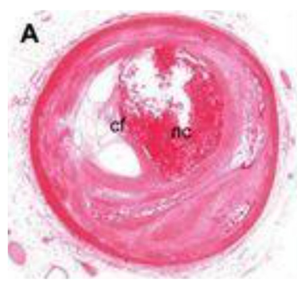
# USO DE REDES NEURONALES PARA LA ESTIMACIÓN DE NIVELES DE COLESTEROL LDL Y COMPARACIÓN CON ESTIMACIÓN MEDIANTE FÓRMULA DE FRIEDEWALD

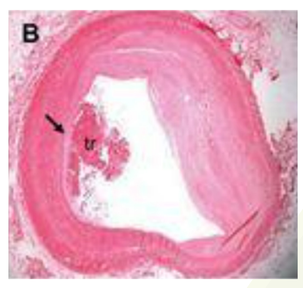
## Cálculo de LDL

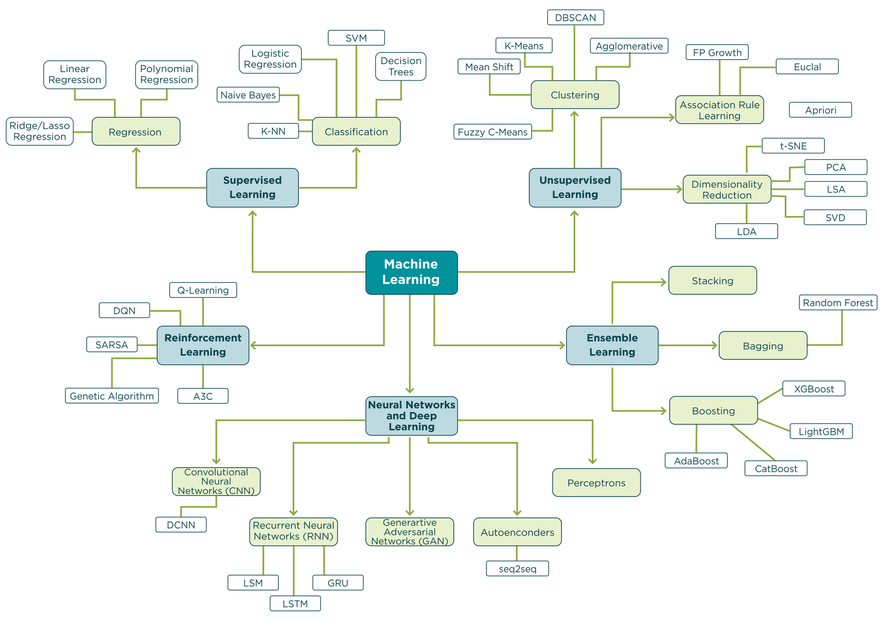


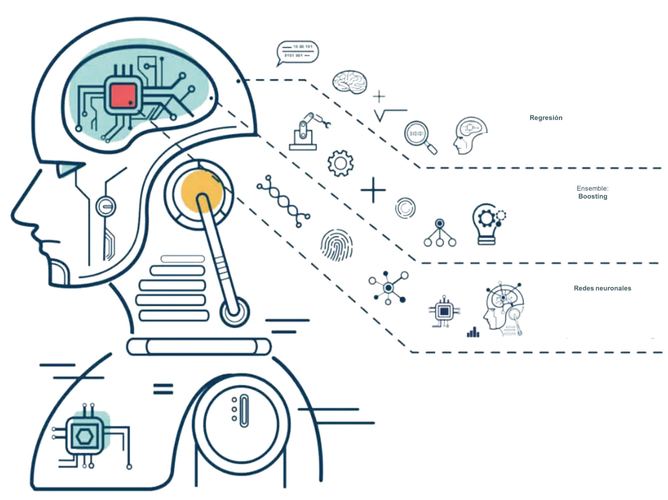



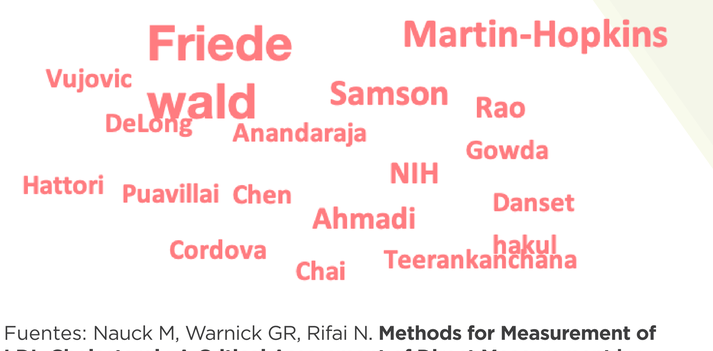
Fuente: Badimón L, Vilahur G, Padró T. **Lipoproteínas, plaquetas y aterotrombosis.** Rev Esp Cardiol. 2009;62(10):1161–78.



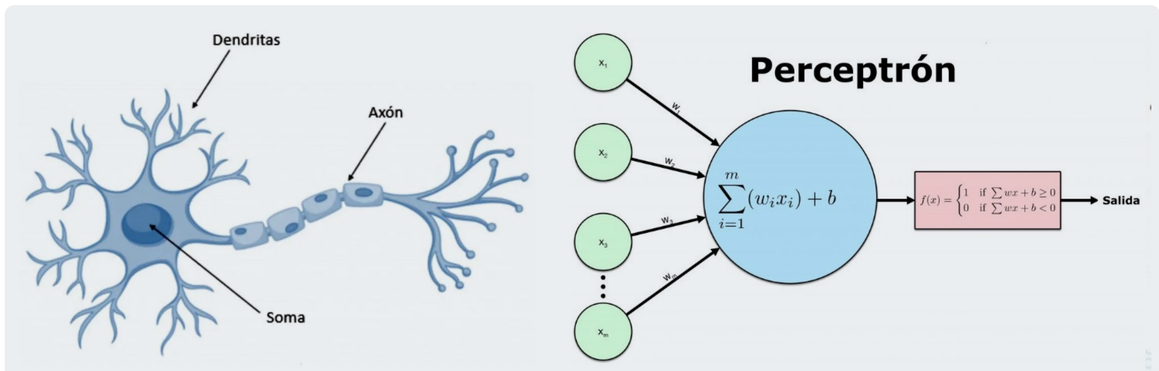



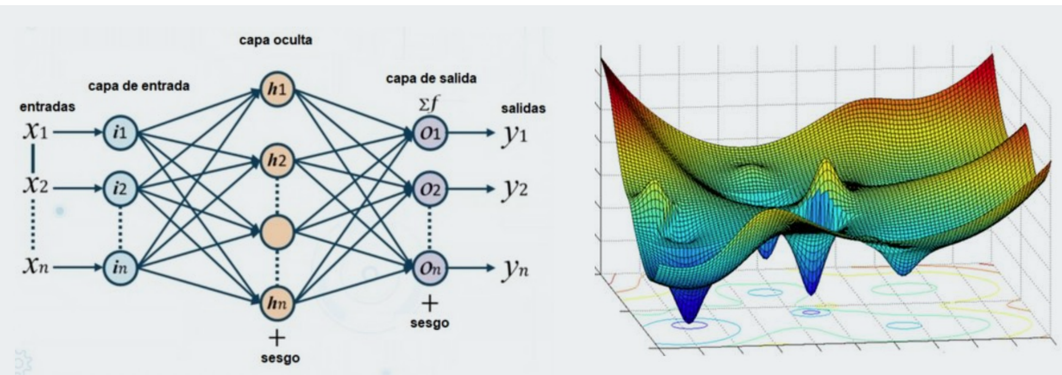




Fuentes: Nauck M, Warnick GR, Rifai N. **Methods for Measurement of LDL-Cholesterol : A Critical Assessment of Direct Measurement by Homogeneous Assays versus Calculation.** 2002;254:236–54.

Sajja A, Park J, Sathiyakumar V, Varghese B, Pallazola VA, Marvel FA, *et al.* **Comparison of Methods to Estimate Low-Density Lipoprotein Cholesterol in Patients with High Triglyceride Levels.** JAMA Netw Open. 2021;4(10):1–14.




Fuente: **historia-redes-neuronales-y-deep-learning** @ www.timetoast.com [Internet]. Available from: https://www.timetoast.com/timelines/historia-redes-neuronales-y-deep-learning




Fuente: Porto A, Industrialdea A. **Estudio de la mejora de modelos de comportamiento de variables energéticas mediante Committee Machine de redes neuronales**. 2020;937–44.

## Fórmula de Friedewald

LDL-c = CT – HDL-c – (TG/5)

### Variables:

Colesterol total
Colesterol HDL
Triglicéridos

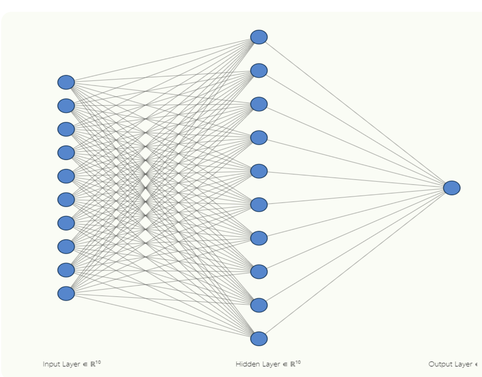
### Cohorte:

10.000 analticas pacientes HUVM

### Conjunto de datos

| Datos para entrenamiento | Datos para test |
|---|---|
| entrenamiento | test |
| **Modelo de entrenamiento** 8.000 analíticas | **Modelo de evaluación** 2.000 analíticas |

## Estructura red neuronal: tipo Densa




**Capa de ENTRADA**
- 10 neuronas
- Activación **Relu**

**Capa OCULTA**
- 10 neuronas
- Activación **Relu**

**Capa SALIDA**
- 1 neuronas
- Activación **Lineal**

**Total parámetros**
- 161 parámetros

**Épocas**
- 100 epoch

**Tamaño de Batch**
- 64 batch_size

# Entorno de trabajo

## Redes Neuronales

```
[ ]  model = tf.keras.Sequential()

     model.add(tf.keras.layers.Dense(10, activation="relu", input_dim=(3)))
     model.add(tf.keras.layers.Dense(10, activation="relu"))
     model.add(tf.keras.layers.Dense(1))

     model.compile(optimizer="Adam", loss="mse", metrics=["mean_absolute_error"])

     model.summary()
```

```
Model: "sequential_19"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_74 (Dense)            (None, 10)                40

 dense_75 (Dense)            (None, 10)                110

 dense_76 (Dense)            (None, 1)                 11

=================================================================
Total params: 161
Trainable params: 161
Non-trainable params: 0
_________________________________________________________________
```

TensorFlow

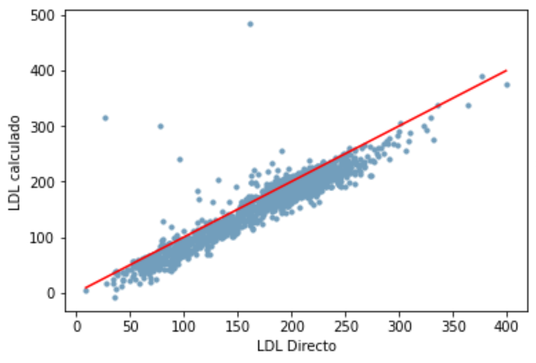
python + Google colab

```
[ ]  fit_history = model.fit(x_train, y_train, validation_data=(x_test, y_test), epochs=100, batch_size=64, verbose=1)

     Epoch 1/100
     125/125 [==============================] - 1s 3ms/step - loss: 8054.4404 - mean_absolute_error: 71.3907 - val_loss: 657.2716 - val_mean_absolute_error: 19.6820
     Epoch 2/100
     125/125 [==============================] - 0s 2ms/step - loss: 594.1553 - mean_absolute_error: 17.2618 - val_loss: 449.5901 - val_mean_absolute_error: 15.4225
     Epoch 3/100
     125/125 [==============================] - 0s 2ms/step - loss: 510.1125 - mean_absolute_error: 15.3716 - val_loss: 397.8989 - val_mean_absolute_error: 14.1904
     Epoch 4/100
     125/125 [==============================] - 0s 2ms/step - loss: 459.3434 - mean_absolute_error: 14.2054 - val_loss: 344.0894 - val_mean_absolute_error: 12.7365
     Epoch 5/100
     125/125 [==============================] - 0s 2ms/step - loss: 409.6082 - mean_absolute_error: 12.8368 - val_loss: 305.0590 - val_mean_absolute_error: 11.4845
     Epoch 6/100
```

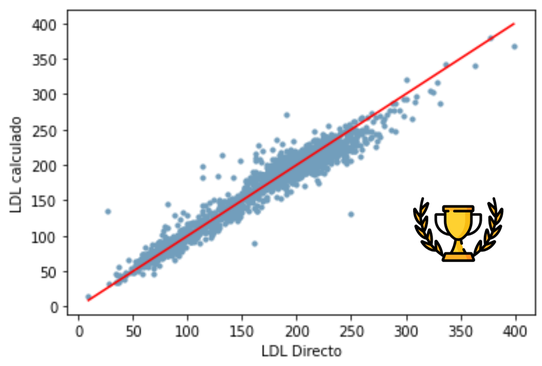
# Resultados

## Friedewald




MAE = 17,30
RMSE = 23,53
Pearson = 0,93

## Red Neuronal

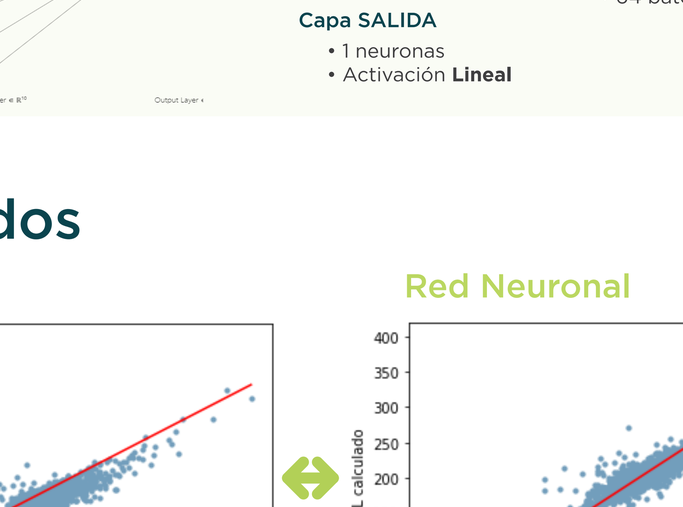



MAE = 10,03
RMSE = 13,91
Pearson = 0,97

| | Media | Desv. Standar |
|---|---|---|
| LDL directo | 182,44 mg/dL | 53,18 |
| Red neuronal | 181,72 mg/dL | 50,45 |
| Friedewald | 168,09 mg/dL | 54,22 |

# Conclusiones

- Los resultados obtenidos mediante la red neuronal son mejores que mediante Friedewald.
- Es relativamente fácil la integración de modelos de IA en los SIL.
- La revolución de los datos ha llegado.




Celebramos 5 AÑOS

MAT-ES-2300730 V1 Abril 2023

sanofi