

# Clasificacin de ECG usando Redes Neuronales

J. Agustín Barrachina  
Miembro Estudiantil IEEE  
Instituto Tecnológico de Buenos Aires

**Abstract**—Se clasificaron latidos de un segmento de seal de un electrocardiograma segn el tipo de arritmia. El algoritmo utilizado fue un perceptrn multicapa usando backpropagation como clasificador. Se utilizaron PCA y SOM para reducir la dimensionalidad de los datos.

## I. INTRODUCCIÓN

El trabajo consisti en clasificar los latidos de un segmento de una seal de electrocardiograma de dos canales de 21 horas de duracin segn el tipo de arritmia.

El objetivo fue crear y entrenar un sistema que pueda reconocer y clasificar los distintos tipos de latidos.

## II. DATA SET

Se utilizaron las grabaciones de "MIT-BIH Long Term Database" [1] disponible en el repositorio PhysioNet [2].

La grabacin incluye anotaciones que identifican la posicin y tipo de cada uno de los latidos presentes en la misma.

La grabacin se identificar por el nmero 14172, y el identificador de la base de datos es "ltldb" (Long Term DataBase). La misma presenta principalmente 4 tipos de latidos:

- **Normales.** Identificados por la letra 'N'.
- **Ventriculares prematuros** Identificados por la letra 'V'.
- **Supraventriculares prematuros** Identificados por la letra 'S'.
- **Nodales prematuros** Identificados por la letra 'J'.

Para el tratamiento de los datos se utiliz la librería de python "wfdb" [3].

### A. Reduccin de Dimensionalidad

Se usaron dos técnicas distintas para reducir la dimensión de los datos y así acelerar y facilitar las operaciones de la red neuronal.

La primera técnica fue la utilización de *Principal Component Analysis* (PCA). La teoría del mismo no se desarrolló en este informe.

Otra técnica utilizada fue la implementación de una red *Self Organized Map* (SOM) que tampoco se desarrolló en este informe.

Finalmente se decidió optar por dejar la implementación del PCA por un motivo puramente de tiempo de ejecución, ya que la red SOM requiera mucho tiempo de cómputo y no obtenga resultados perceptiblemente superiores a los de la red PCA. Se truncó las dimensiones de PCA reduciendo los datos de cada latido de 250 a 32.

## III. DISEÑO DE LA RED NEURONAL

Según Rezaul Beggs et al. [4], la técnica más comúnmente utilizada de Redes Neuronales hasta el día de la fecha en procesamiento de ECG fue el perceptrn multicapa (MLP) entrenada con backpropagation (Sección 2 Capítulo 3), por dicho motivo se utilizó este mismo como el algoritmo para utilizar en este trabajo.

El mismo trabajo presenta una tabla (figura 1) que presenta diferentes aplicaciones de redes neuronales en el ámbito de cardiología. Lamentablemente ninguno de los casos es aplicable a nuestro caso ya que suelen ser diferenciación entre sí y no sobre la probabilidad de tener AMI. Sin embargo, la mayoría de los casos usan una configuración de MLP de una sola capa oculta con lo cual en este trabajo se utilizó esa misma configuración. Normalmente además se recomienda utilizar solo una capa ya que casi con seguridad no es necesario más capas.

Table 1. Applications of ANNs in cardiology

Authors	Objective	Input	Output	Type of ANN	Results/Remarks
Azuaje et al. (1999)	Coronary disease risk	Poincare plots Binary coded Analog coded	Two, three, and five levels of risk	MLP (144,576,1024/70,200,500/30,100,200/2,3,5)	Even binary coded Poincare plots produce useful results
Baxt (1991); Baxt et al. (1996)	Acute Myocardial infarction (AMI)	20 items including symptoms, past history, and physical and laboratory findings, e.g. palpitations, Angina, Rales or T-wave inversion	AMI yes/no	MLP (20/10/10/1)	MLP had higher sensitivity and specificity in diagnosing a myocardial infarction than physicians
Dorfner and Porets (1994)	Coronary artery disease	45 Parameters of the Stress redistribution scintigram (5 segments times 3 views times 3 phases – stress, rest, washout)	Normal/ pathological	1) MLP 2) RBFN 3) Coac section function networks	MLP can be among the poorest methods Importance of initialization in MLP and CSFN
Kennedy et al. (1997)	AMI	39 items, e.g. Blood and ECG parameters derived to 53 binary inputs	AMI yes/no	MLP (53/18/1)	Trained ANN at least as accurate as medical doctors
Kulkarni et al. (1997)	Ischaemic heart disease (IHD)	77 parameters; IHD data for different diagnostic levels (e.g. signs, symptoms, exercise ECG, and myocardial scintigraphy, angiography)	Classification	1) (semi-naive Bayesian classifier 2) Backpropagation learning of neural networks 3) Two algorithms for induction of decision trees (Assistant-I and -R) 4) k-nearest neighbors	Improvements in the predictive power of the diagnostic process. Analyzing and improving the diagnosis of ischaemic heart disease with machine learning
Mobley et al. (2004)	Coronary stenosis	11 personal parameters	Stenosis yes/no	MLP (11/36/1)	Specificity 26% Sensitivity 100%
Mobley et al. (2000)	Coronary artery stenosis	14 angiographic variables	Stenosis yes/no	MLP (14/26/1)	ANN could identify patients who did not need coronary angiography

Fig. 1. Redes Neuronales Aplicadas en Cardiología según [4]

Si bien existen varios trabajos sobre la elección de la cantidad de elementos en la red oculta como [5] o [6], no existe una receta única e infalible sobre cuántos nodos darle a la capa oculta. Para elegirlos se utilizaron las siguientes reglas *RoT* (Rule of Thumb) que se describirán a continuación.

- 1) *Errar por más y no por menos:* Unos nodos extra no es probable que causen daño a la convergencia de la red, se

los puede pensar como un exceso de capacidad. Mientras que usar pocos nodos pueden afectar a la convergencia.

- 2) *Basado en la entrada y la salida:* Utilizar una cantidad de nodos que se encuentren entre la cantidad de nodos de la entrada y la cantidad de nodos de la salida. (Jeff Heaton autor de "Introduction to Neural Networks for Java")

Utilizando estas dos reglas se eligi utilizar la expresin  $nodos_{hidden} = (nodos_{entrada} + nodos_{salida})/1.5$ .

Para el algoritmo de aprendizaje se utiliz una libreria [7] que implementa redes neurales feed-forward (Redes Multicapa) permitiendo una variedad de algoritmos de aprendizajes de backpropagation.

#### A. Set de Entrenamiento

En un primer momento se entrenaba la red con los primeros casos de latidos. Para dicho set de entrenamiento, el error aumentaba a medida que los datos estaban ms alejados temporalmente de los mismos, es decir, si se tomaban los primeros 10 casos del set de datos para el entrenamiento, el conjunto de datos a partir del onceavo hasta el veinteavo mostraba grandes resultados, mientras que los datos rondando el nmero mil posea resultados muy poco satisfactorios.

Para contrarrestar este problema se tomaron los datos de entrenamiento de forma aleatoria. sto logr mejorar mucho el desempeo pero a coste de tener mayor varianza en los resultados. Adems el caso de seleccionar dos veces los mismos datos puede ocurrir con el cdigo actual.

#### IV. OVERFITTING

La implementacin de la red present overfitting. Como se puede ver en los resultados listados a continuacin, el error del set de datos del test set comienza a incrementar.

Error:	0.290525167175	Epoch:	1000
Error:	0.252432576731	Epoch:	2000
Error:	0.233333077085	Epoch:	3000
Error:	0.226507241997	Epoch:	4000
Error:	0.227502496085	Epoch:	5000
Error:	0.230580316278	Epoch:	6000
Error:	0.233771065888	Epoch:	7000
Error:	0.236764878811	Epoch:	8000
Error:	0.239612203297	Epoch:	9000
Error:	0.242385089641	Epoch:	10000

La libreria utilizada [7] permite el uso de dropout para poder reducir el overfitting. El resultado a continuacin muestra el caso para un dropout del 50%.

Error:	0.330238968525	Epoch:	1000
Error:	0.254160001896	Epoch:	2000
Error:	0.214160846981	Epoch:	3000
Error:	0.193992225022	Epoch:	4000
Error:	0.185100235993	Epoch:	5000
Error:	0.180218777466	Epoch:	6000
Error:	0.176489777988	Epoch:	7000
Error:	0.173452841416	Epoch:	8000

Error:	0.171893196978	Epoch:	9000
Error:	0.172405423062	Epoch:	10000

Se puede observar que el dropout mejor notoriamente el overfitting haciendo que el mismo se de pasados las 9000 iteraciones.

#### A. Cambio a la Libreria

Sin embargo el overfitting no se elimin por completo y existe un nmero ptimo para el cual conviene no seguir entrenando la red. Por dicho motivo, se modific la libreria para poder realizar *early stop*. La implementacin permite elegir la cantidad de iteraciones con un aumento del error antes de abandonar el entrenamiento.

#### V. IMPLEMENTACIN DE LA RED

La siguiente figura (figura 2, levemente modificada de la figura 6 de [4]) muestra el diagrama bsico de la implementacin del proyecto.

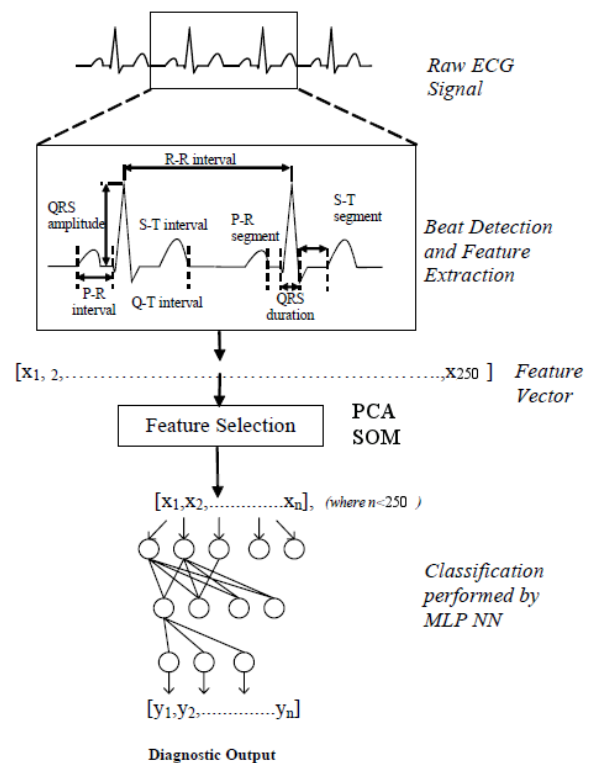


Fig. 2. Diagrama general del proyecto

En primer lugar se extrae y se extraen varios vectores de dimensi3n 250 de cada caso de latido. Luego se reduce su dimensi3n utilizando PCA o SOM a una dimensi3n menor. Finalmente se aplica la red multicapa obteniendo la salida  $y_i$  donde cada valor corresponde a una clasificaci3n posible ( $0 \leq y_i \leq 1$  que representa la probabilidad de que el latido corresponda a dicha clasificaci3n).

- Reducci3n de Dimensi3n usando PCA 32
- Configuraci3n 32 - 20 - 4

- Funcin de activacin: Sigmoidea
- Cantidad de latidos de entrenamiento 10
- Dropout = 50
- Early Stop de hasta 10 valores

## VI. ANLISIS DE LOS RESULTADOS

Para analizar los resultados se usaron dos parmetros que se los denomin *aciertos* y los *hallados*.

El primero cuenta la cantidad de *aciertos* de cada decisin, es decir, si se dijo que cierto numero de latidos corresponden a un categoria, cuntos de ellos realmente eran de esa categoria.

Por ejemplo, si encuentro 100 latidos 'N' y 90 de ellos lo eran, el porcentaje de aciertos ser del 90% lo cual parece ser un nmero muy aceptable. Sin embargo, si se aliment la red con 900 latidos, solo el 10% fueron hallados, haciendo que el desempeo de la red sea indeseado.

Por dicho motivo, para este segundo caso, se midi la cantidad de elementos de esa categoria encontrados o *hallados* sobre el total real.

## VII. RESULTADOS

La cantidad de aciertos totales fue superior al 70% en la mayora de los casos (recordar que como se usan sets se entrenamientos aleatorios en casa caso V, el mismo suele tener varianza).

Para una iteracin con 2000 'N', 2000 'V', 1003 'S' y 148 'J', se tuvo un acierto de 77% usando 10 casos de entrenamiento para cada categoria y un *early stop* que termine el entrenamiento cuando la red aumenta 10 veces su error.

Los latidos que ms satisfactoriamente se detect fueron los Ventriculares ('V') en los cuales se obtuvo bajo varias condiciones, un resultado del 99% tanto de *aciertos* como de cantidad total *hallada*.

sto ltimo se le puede dar una explicacin grfica a partir de la red implementada SOM. El resultado de la misma se encuentra en la figura 3. En dicha figura se puede ver claramente que los valores de los ventriculares estn claramente definidos por lo cual es de esperar que se obtengan tan buenos resultados. Los normales y ventriculares poseen error pero en lneas generales tiene cierto grado de diferenciacin. Los nodales prematuros por otro lado se superponen mucho con los dems resultados haciendo que en varios casos se interprete los dems casos (normales, ventriculares y supraventriculares) como un caso de nodales prematuros.

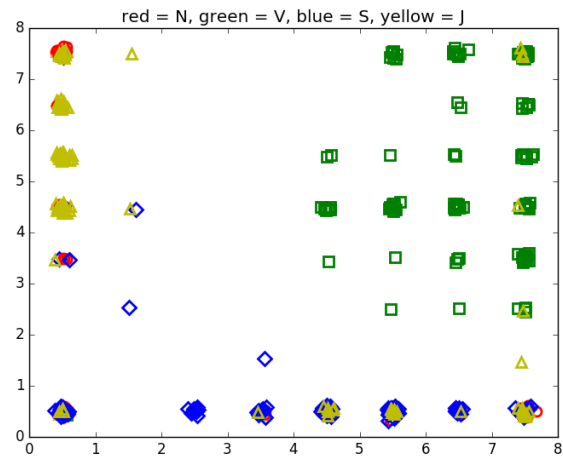


Fig. 3. Resultados de una red SOM sobre algunos datos

Aproximadamente se obtuvo un *porcentaje de enfermos ponderado* superior al 70%.

## VIII. POSIBLES MEJORAS

Se podra implementar el mtodo denominado como *Cascade Correlation* (Fahlman et al. 1990) que permite ajustar dinmicamente la estructura de la red. La misma comienza con una red minimalista (perceptrn simple) y luego va ajustando los nodos de la capa oculta durante el entrenamiento.

Se debera seleccionar los datos de entrenamiento de una forma que asegure que los mismos no se repitan.

La librería permite definir funciones costo propias. sto se podra aprovechar para definir una funcin de costo propia que le otorgue menor peso a los casos Normales respecto a los dems.

La librería ofrece muchas variantes de backpropagation que no fueron probadas ni investigadas. Sera interesante inspeccionar sobre dichas variantes.

## ACKNOWLEDGMENT

## REFERENCES

- [1] The mit-bih long term database. [Online]. Available: <https://physionet.org/physiobank/database/ltdb/>
- [2] G. L. H. J. I. P. M. R. M. J. M. G. P. C.-K. S. H. P. P. Goldberger AL, Amaral LAN and PhysioNet, "Components of a new research resource for complex physiologic signals," *Circulation* 101(23):e215-e220, jun 2000, [Circulation Electronic Pages; <http://circ.ahajournals.org/content/101/23/e215.full>].
- [3] C. Xie. wfdb-python. [Online]. Available: <https://github.com/MIT-LCP/wfdb-python>
- [4] R. S. Rezaul Begg, Joarder Kamruzzaman, *Neural Network in Healthcare: Potential and Challenges*. Australia: IDEA Group Publishing, 2006.
- [5] G. C. Lawrence, S. and A. Tsoi, "What size neural network gives optimal generalization? convergence properties of backpropagation," *Technical Report UMIACS-TR-96-22 and CS-TR-3617*, 1996.
- [6] A. Elisseff and H. Paugam-Moisy, "Size of multilayer networks for exact learning: analytic approach," *Advances in Neural Information Processing Systems 9*, Cambridge, MA: The MIT Press, pp. 162–168, 1997.
- [7] J. Grimnes. (2016) Nimblenet. python-neural-network. [Online]. Available: <https://github.com/jorgenkg/python-neural-network>