USEDAT-07: IBRO USA-Europe Data Analysis Training Congress, Cambridge, UK-Bilbao, Spain-Miami, USA, 2021

PTML Artificial Neural Network Chemoinformatics classification model for enantioselective reactions.

Shan He [a], Sonia Arrasate [a], and Humberto González-Díaz [a,b,]*

[a] *Department of Organic and Inorganic Chemistry, Faculty of Science and Technology University of Basque Country UPV-EHU, 48940, Leioa, Basque Country, Spain.*
[b] IKERBASQUE, Basque Foundation for Science, 48011, Bilbao, Spain.

Resumen

El Machine Learning (ML) se utiliza para el aprendizaje de un sistema. Uno de los propósitos de este aprendizaje automático es la construcción de nuevos modelos computacionales. El ML muestra éxito en diversas áreas como por ejemplo, en sistemas de referencia, interfaz cerebro-computadora, robótica y química.[1,2,3,4]

Recientemente, los operadores de la Teoría de Perturbaciones (PT) y las técnicas de ML, se han combinado para crear modelos potentes PTML (PT+ ML), que se aplican a sistemas biológicos complejos en la predicción de la interacción de fármaco-proteína. Como para aquellas proteínas diana involucradas en la vía de dopamina, nanotecnología, ciencias materiales etc.[5,6]

Los modelos PTML añaden a los valores de la $f(v_{ij})_{ref}$ los valores de los operadores. Por ello, necesitamos calcular los valores de los PTOs (operadores de la teoría de perturbación-

Abstract

Machine Learning (ML) is used to learn a system. One of the purposes of this automatic learning is the construction of new computational models. ML shows success in various areas such as reference systems, brain-computer interface, robotics, and chemistry.

Recently, Perturbation Theory (PT) operators and ML techniques have been combined to create powerful PTML (PT+ML) models, which are applied to complex biological systems in predicting drug-protein interaction. As for those target proteins involved in the dopamine pathway, nanotechnology, material science, etc.

The PTML models add the values of the operators to the values of $f(v_{ij})_{ref}$. Therefore, we need to calculate the values of the PTOs (Perturbation Theory Operators) in the data processing step. This allows us to carry out a process of merging information with variables and conditions from

[1] Duda, R.; Hart, P. Stork, D. *Pattern Classification*; 2nd ed. Wiley: New York, 2001.
[2] MacKay, D. *Information theory, Inference, and Learning Algorithms*; Cambridge University Press: Cambridge, 2005.
[3] Bishop, C. *Pattern Recognition and Machine Learning*; Springer: Berlin, 2006.
[4] Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*; 2nd ed. Springer; New York, 2009.
[5] Ferreira da Costa, J.; Silva, D.; Caamaño Olga; Brea José M; Loza, M. I.; Munteanu, C. R.; Pazos, A.; García-Mera Xerardo; González-Díaz Humbert. Perturbation Theory/machine Learning Model of Chembl Data for Dopamine Targets: Docking, Synthesis, and Assay of New. *Acs Chemical Neuroscience*, **2018**, 9 (11), 2572–2587.
[6] Blay, V.; Yokoi, T.; González-Díaz Humbert. Perturbation Theory–machine Learning Study of Zeolite Materials Desilication. *Journal of Chemical Information and Modeling*, **2018**, 58 (12), 2414–2419.

Perturbation Theory Operators) en el paso de procesamiento de datos. Esto nos permiten realizar un proceso de fusión de información con variables y condiciones de diferentes fuentes. Los promedios móviles-Moving Averages (MA), los MA de múltiples condiciones (MMA), los MA dobles, los operadores de covarianza, etc., son algunos ejemplos de PTOs útiles. Después, podemos utilizar la regresión lineal múltiple-Multiple Linear Regression (MLR), el análisis discriminante lineal (LDA)[7] u otras técnicas de ML lineal para buscar el modelo PTML. En los casos no lineales, podemos ajustar los modelos PTML utilizando redes neuronales artificiales (ANN), máquinas de vectores de soporte-Support Vector Machines (SVM), árboles de clasificación y otros métodos de ML.

Una de las aplicaciones importantes de la ANN se encuentra en el estudio de reacciones químicas como alternativa a las técnicas clásicas de regresión y clasificación. Por lo tanto, en este trabajo de fin de master se desarrolla modelos PTML-ANN con intención de visualizar posibles mejoras de las técnicas clásicas. En este contexto, se introduce este término de manera genérica. Las ANNs están inspiradas en las redes neuronales biológicas del cerebro humano. Están constituidas por elementos que se comportan de forma similar a la neurona biológica en sus funciones más comunes.

Las ANN (Figura 1) al margen de "parecerse" al cerebro presentan una serie de características propias del cerebro. Por ejemplo, las ANN aprenden de la experiencia, generalizan de ejemplos previos a ejemplos nuevos y abstraen las características principales de una serie de datos. [8] Con lo cual estas redes funcionan como neuronas interconectadas que crean estímulos. Se considera red tipo ANN a una red biomolecular compleja que consta de nodos y

different sources. Moving Averages (MA), multi-condition MA (MMA), double MA, covariance operators, etc., are some examples of useful PTOs. Then, we can use Multiple Linear Regression (MLR), Linear Discriminant Analysis (LDA), or other linear ML techniques to find the PTML model. In non-linear cases, we can fit PTML models using Artificial Neural Networks (ANN), Support Vector Machines (SVM), Classification Trees, and other ML methods.

One of the important applications of ANN is found in the study of chemical reactions as an alternative to classical regression and classification techniques. Therefore, in this master's thesis, PTML-ANN models are developed with the intention of visualizing possible improvements in classical techniques. In this context, this term is presented in a generic way. ANNs are inspired by the biological neural networks of the human brain. They are made up of elements that behave in a similar way to the biological neuron in its most common functions.

The ANN (Figure 1) aside from "resembling" the brain present a series of characteristics of the brain. For example, ANNs learn from experience, generalize from previous examples to new examples, and abstract the main characteristics of a data series. With which these networks function as interconnected neurons that create stimuli. ANN-type network is considered to be a complex biomolecular network consisting of nodes and edges. These networks are formed by "neurons", where each of them is a function, which will take a certain amount of data and provide an output response. The ANN presents three different types of functions which are:

[7] Hill, T.; Lewicki, P. *STATISTICS Methods and Applications. A Comprehensive Reference for Science*, Industry and Data Mining; StatSoft: Tulsa, 2006.
[8] Basogain, X. *Redes Neuronales Artificiales y sus Aplicaciones*, Publicaciones de la Escualea de Ingenieros, Bilbao, 1998.

bordes. Estas redes están formadas por "neuronas", donde cada una de ella es una función, que tomará una determinada cantidad de datos y proporcionará una respuesta de salida. La ANN presenta tres tipos de funciones diferentes que son:

- Función de entrada: Es la función de suma que se obtiene al multiplicar los datos de entrada y de salida por sus pesos.
- Función de excitación o activación: Esto tomará como datos de entrada/salida anteriores. Los tipos de funciones más comunes dependiendo de la entrada/salida serán el umbral (sigmoide) o tangente hiperbólica.
- Funciones de transferencia: Es la función que se utiliza para cerrar. El valor proporcionado por las funciones de activación es la entrada a la red ANN.[9]

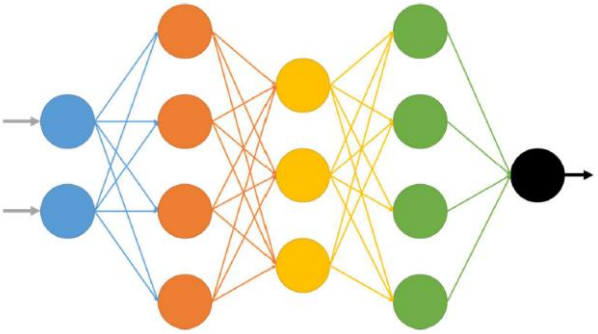


Figura 1: Imagen esquemática de las redes neuronales artificiales.[3]

Existen diferentes tipos de redes ANN, las cuales se denominan como "SHORT NAME" (indicado en letras mayúsculas).

Las variables de entrada: Neurona de entrada- Número de neuronas en la capas intermedias- Neuronas de salida: valor de salida (todos expresados por números. Existe diferentes tipos de redes que se van a mencionar a continuación:

• Input function: It is the sum function obtained by multiplying the input and output data by their weights.

• Excitation or activation function: This will take as previous input / output data. The most common types of functions depending on the input / output will be the threshold (sigmoid) or hyperbolic tangent.

• Transfer functions: It is the function used to close. The value provided by the trigger functions is the input to the ANN network.

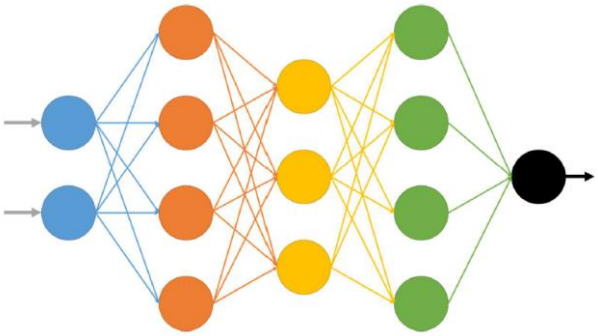


***Figure 1**: Schematic image of artificial neural networks.*

There are different types of ANN networks, which are called "SHORT NAME" (indicated in capital letters).

The input variables: Input neuron- Number of neurons in the intermediate layers- Output neurons: output value (all expressed by numbers. There are different types of networks that will be mentioned below:

• Linear neural network (Linear Neuronal Network-LNN).

• Multi-layer Perceptron (Multi Layer Perceptron-MLP).

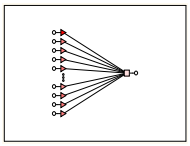
• Basic radial function (Radical Basis Fuction-RBF).

[9] Walczak, S.; Cerpa, N., ≪Artificial Neural Networks≫. In *Encyclopedia of Physical Science and Technology (Third Edition)*, Meyers, R. A., Ed. Academic Press: New York, 2003; pp 631-645.

- Red neuronal lineal (Lineal Neuronal Network-LNN).
- Perceptrón multi-capa (Multi Layer Perceptron-MLP).
- Función radial básica (Radical Basis Fuction-RBF).[10]

Así, en este trabajo se han creado dos tipos de redes: una red neuronal lineal y dos redes no lineales (monocapa y bicapa). Tras el entrenamiento y la validación del modelo ANN, los resultados obtenidos de mayor relevancia, se resumen en las siguientes tablas (Tabla 1, 2 y 3), donde en cada una de ellas aparece una red diferente. En primer lugar, se presentan los valores AUROC (Área Bajo Característica Operativa del Receptor), arriba se establece el valor del conjunto de entrenamiento y, a continuación, se muestra el valor del conjunto de confirmación.

Tabla 1: Red neuronal artificial para un modelo lineal con los valores de AUROC y los resultados del conjunto de entrenamiento y confirmación.

	Entrenamiento				
Perfil	**AUROC**	**f(v_{ij})**	**0**	**1**	**(%)**
LNN 13:13-1:1	0.943	0	49329	1685	94.3
	0.929	1	0	27718	100
	Validación				
	Par.	**(%)**	**f(v_{ij})**	**0**	**1**
	Sp	92.9	0	18711	719
	Sn	100	1	0	9406

Thus, in this work two types of networks have been created: a linear neural network and two non-linear networks (monolayer and bilayer). After the training and validation of the ANN model, the most relevant results obtained are summarized in the following tables (Table 1, 2 and 3), where a different network appears in each of them. First, the AUROC (Area Under Receiver Operating Characteristic) values are displayed, the training set value is set above, and then the confirmation set value is displayed.

***Table 1**: Artificial neural network for a linear model with the AUROC values and the results of the training and validation set.*

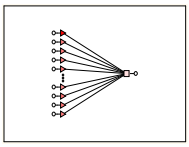
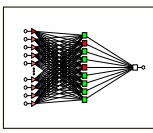
	Training				
Profile	**AUROC**	**f(v_{ij})**	**0**	**1**	**(%)**
LNN 13:13-1:1	0.943	0	49329	1685	94.3
	0.929	1	0	27718	100
	Validation				
	Par.	**(%)**	**f(v_{ij})**	**0**	**1**
	Sp	92.9	0	18711	719
	Sn	100	1	0	9406

***Table 2:** Artificial neural network for a single layer nonlinear model with the AUROC values and the results of the training and validation set.*

	Training				
Profile	**AUROC**	**f(v_{ij})**	**0**	**1**	**(%)**
MLP 13:13-9-1:1	0.999	0	49329	2	99.9
	0.996	1	0	29401	100
	Validation				
	Para	**(%)**	**f(v_{ij})**	**0**	**1**
	Sp	99.6	0	18711	39
	Sn	100	1	0	10086

[10] Bediaga, H; Arrasate, S.; González-Díaz, H. *Kimioinformatika Gidaliburua*; Informe para la Construcción de Redes Neuronales: UPV/EHU, SP, Febrero de 2021.

Tabla 2: Red neuronal artificial para un modelo no lineal de capa única con los valores de AUROC y los resultados del conjunto de entrenamiento y confirmación.

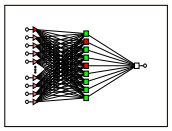
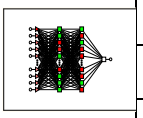
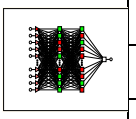
	Entrenamiento				
Perfil	**AUROC**	**f(v_{ij})**	**0**	**1**	**(%)**
MLP 13:13-9-1:1	0.999	0	49329	2	99.9
	0.996	1	0	29401	100
	Validación				
	Para	**(%)**	**f(v_{ij})**	**0**	**1**
	Sp	99.6	0	18711	39
	Sn	100	1	0	10086

Tabla 3: Red neuronal artificial para un modelo no lineal de doble capa con los valores de AUROC y los resultados del conjunto de entrenamiento y confirmación.

	Entrenamiento				
Perfil	**AUROC**	**f(v_{ij})**	**0**	**1**	**(%)**
MLP 13:13-13-13-1:1	0.999	0	49329	3	99.9
	0.999	1	0	29400	100
	Validación				
	Para	**(%)**	**f(v_{ij})**	**0**	**1**
	Sp	99.9	0	18711	2
	Sn	100	1	0	10123

Primeramente, cabe destacar la obtención de excelentes resultados tanto por el modelo lineal como el no lineal. Con lo cual, a simple vista, parece indistinto utilizar cualquiera de los dos. Sin embargo, se debe tener en cuenta la complejidad y el tiempo de cálculo que supondría trabajar con un modelo complejo como es el no lineal. Además, el este último mejora los resultados con un porcentaje despreciable. Por lo que, teniendo en cuenta el principio de parsimonia se concluye que estamos ante un problema de modelos lineales.

***Table 3**: Artificial neural network for a non-linear double layer model with the AUROC values and the results of the training and confirmation set.*

	Training				
Profile	**AUROC**	**f(v_{ij})**	**0**	**1**	**(%)**
MLP 13:13-13-13-1:1	0.999	0	49329	3	99.9
	0.999	1	0	29400	100
	Validation				
	Para	**(%)**	**f(v_{ij})**	**0**	**1**
	Sp	99.9	0	18711	2
	Sn	100	1	0	10123

First of all, it is worth highlighting the obtaining of excellent results from both the linear and non-linear models. With which, at first glance, it seems indistinct to use either of the two. However, the complexity and calculation time involved in working with a complex model such as the non-linear one must be taken into account. In addition, the latter improves the results with a negligible percentage. Therefore, taking into account the principle of parsimony, it is concluded that we are facing a problem of linear models.