

Universidad Politécnica de Cartagena



Departamento de Tecnologías de la
Información y las Comunicaciones

Tesis Doctoral

Aprendizaje Máquina Multitarea mediante Edición de Datos y Algoritmos de Aprendizaje Extremo

Autor: Andrés Bueno Crespo

Director: Dr. José Luis Sancho Gómez

2013

UMI Number: 3606178

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



UMI 3606178

Published by ProQuest LLC (2013). Copyright in the Dissertation held by the Author.

Microform Edition © ProQuest LLC.

All rights reserved. This work is protected against unauthorized copying under Title 17, United States Code



ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

PREVIEW

*A mis padres, a mi mujer, y a esos
dos peques que espero que algún
día entiendan "esas cosas raras
que escribe papá".*

PREVIEW

PREVIEW

PREVIEW

PREVIEW

PREVIEW

Agradecimientos

Este apartado es algo que todo el mundo se lee, bien porque me conoce y espera encontrar su nombre en estas líneas o bien porque aunque no me conozca le lleva la propia curiosidad. Si eres del grupo de los que me conocen, espero no defraudarte, aunque cuando uno empieza a citar nombres corre el peligro de no nombrar a alguien, que no quiere decir que se ha olvidado, ya que son muchas las personas de las que de alguna manera u otra les pertenece algo de esta Tesis.

Quiero empezar por mi director, el Dr. José Luis Sancho, Profesor Titular de la *Universidad Politécnica de Cartagena*, quién durante años me ha ayudado con su amistad, consejos y apoyo incondicional que ha hecho posible todo este trabajo. Gracias por tu compromiso.

A Pedro, quién de primera hora hemos compartido las mismas inquietudes, por esos bocetos hechos en servilletas de cafetería, aun las guardo, gracias.

A Antonio, que compartió conmigo aquellas primeras investigaciones.

Como olvidar a mis compañeros de trabajo, también un trocito de esta Tesis se los debo a ellos. Un equipo fantástico donde todos reman en la misma dirección.

También una parte de este trabajo se la debo a mi familia: a mis padres, Andrés y Antonia, que siempre me han animado a seguir por este camino y a los que tanto debo. A Diego y Conchi, por tenerme como a un hijo más. A Juan, Mamen, Pedro y Gloria, por hacer que me sienta como en casa y a Diego, por

ser como un hermano.

A María José, por compartir día a día los buenos y malos momentos y sufrir esta Tesis, prácticamente, es casi suya. Y a mis hijos, Álvaro y Andrea, por todo este tiempo robado, espero ahora recuperar todas esas horas de parque.

A todos, gracias por ejercer un *sesgo inductivo positivo* en la realización de esta Tesis, aunque, claro, seguramente no sabrás que es eso del *sesgo inductivo*, pero si te animas y sigues leyendo, seguro que hasta te gusta...

PREVIEW

Resumen

Cuando los seres humanos nos enfrentamos a un nuevo concepto que queremos aprender, nuestro cerebro no lo hace de forma aislada, sino que utiliza todo el conocimiento previamente aprendido para ayudarse en este nuevo aprendizaje. Además, nuestro cerebro es capaz de aislar lo que no va a beneficiarnos y a utilizar lo que realmente nos va a ser útil, esto lo hace muy bien y de forma inconsciente. Sin embargo, cuando una máquina de aprendizaje es entrenada para resolver una determinada tarea, por ejemplo, a diagnosticar una determinada enfermedad, normalmente esta máquina aprende en solitario sólo con los datos disponibles sobre esa enfermedad.

Hay una metodología llamada Aprendizaje Multitarea, MTL (“Multi-Task Learning”), que se fundamenta en la idea inicialmente expuesta. De esta forma, la tarea a resolver (tarea principal) se aprende conjuntamente con otras tareas relacionadas (tareas secundarias), se produce una transferencia de información entre ellas que puede ser ventajosa para el aprendizaje de la primera. Sin embargo, en problemas reales, es difícil encontrar tareas que estén relacionadas o incluso, encontrándolas, es sumamente complejo determinar el grado en que se va a realizar esa ayuda, ya que una tarea puede contener información que puede ayudar pero también perjudicar.

Esta Tesis incorpora una nueva metodología que permite obtener tareas secundarias relacionadas con la que se pretende aprender (tarea principal). La segunda contribución de este trabajo se enmarca también dentro del MTL,

en este caso, diseñando de forma automática una máquina MTL que elimine todos aquellos factores que perjudiquen o no beneficien al aprendizaje de la tarea principal. Esta arquitectura es única y además se obtiene sin necesidad de metodologías de ensayo/error que aumentan la complejidad de cálculo.

PREVIEW

Abstract

When humans faced with a new concept that we want to learn, our brain does not work in isolation, but rather uses all previously learned knowledge to assist in this new learning. In addition, our brain is able to isolate what is not going to benefit and use what will be really useful, it does very well and unconsciously. However, when learning machines are trained for solving an specific problem, for example, to diagnose a particular disease, usually this machine learns only the available data on this disease.

There is a methodology called Multi-Task Learning (MTL), which is based on the idea initially exposed. Thus, the task to be solved (main task) is learned together with other related tasks (secondary tasks), by producing a transfer of information among them which may be advantageous for learning of the main one. But in real problems, it is extremely difficult to determine how the simultaneous learning with other related tasks affects to the performance of the main one, because a task can contain information that can be helpful (i.e. the main task learning is improved) or harmful (i.e. the main task learning gets worse).

This PhD. Thesis, proposes a new methodology that allows to obtain related secondary tasks in order to be helpful to the main one. The second contribution of this work is also included in the MTL framework: the complete automatically removing those factors which harm or no benefit the learning of the main task. This architecture is also unique and it is obtained without the traditional test/fail

methodologies which usually increase the computational complexity.

PREVIEW

Índice

1. Introducción	1
1.1. Introducción	1
1.2. Definición del problema	2
1.3. Objetivos	3
1.4. Estructura de la Tesis	4
2. Aprendizaje máquina y aprendizaje inductivo	7
2.1. Introducción	7
2.2. Aprendizaje máquina	8
2.2.1. Espacio de hipótesis	9
2.2.2. Sesgo inductivo	10
2.2.3. Redes Multi-capas Unidireccionales	13
2.3. Aprendizaje inductivo	18
2.3.1. “Hints”	19
2.3.2. Tareas relacionadas	24
2.4. Transferencia inductiva en las ANNs	28
2.4.1. Transferencia inductiva representacional y funcional	28
2.4.2. Enfoques para la transferencia inductiva usando ANNs	29
2.5. Arquitectura de las ANNs Multitarea	30
2.6. Ejemplo de aprendizaje MTL: Logic Domain	31
3. Generación artificial de “hints” basada en la Edición de Datos	37

3.1. Introduction	37
3.2. Edición de Datos	39
3.3. Método propuesto	40
3.4. Experimentos	47
3.4.1. Problemas artificiales	48
3.4.2. Conjunto de datos de la “UCI MLR”	54
3.5. Data Editing para MTL	56
3.6. Conclusiones	61
4. Diseño de arquitecturas MLP mediante ELM	63
4.1. Introducción	63
4.2. Extreme Learning Machine	65
4.3. Métodos de poda para las ELM	69
4.3.1. Poda ELM: P-ELM	71
4.3.2. Poda Óptima ELM: OP-ELM	72
4.4. Optimización para clasificación	78
4.5. Diseño automático de MLP's	79
4.6. Experimentos. Diseño de arquitecturas MLP	80
4.7. Conclusiones	90
5. Diseño de arquitectura MTL mediante ELM	93
5.1. Introducción	93
5.2. Diseño mediante regularización	94
5.3. Arquitectura MTL basada en ASELM	101
5.4. Experimentos	103
5.4.1. Logic Domain	104
5.4.2. Iris	108
5.4.3. Monk's Problems	110
5.4.4. Telugu	111
5.5. Conclusiones	113
6. Contribuciones, conclusiones y trabajos futuros	117
6.1. Introducción	117
6.2. Contribuciones	118

6.2.1. Capítulos de Libro	121
6.2.2. Revistas	121
6.2.3. Congresos	122
6.2.4. Otras Publicaciones	123
6.3. Conclusiones	124
6.4. Trabajos futuros	127
A. Tabla de acrónimos	129
Bibliografía	131

PREVIEW

Lista de Tablas

2.1. Descripción de las tareas del “Logic Domain”. Cada tarea es una expresión lógica formada por la combinación de cuatro características de entrada, estando cada una de ellas más o menos relacionada con la principal (T_0).	33
2.2. Probabilidad de acierto para clasificación del conjunto de test (media \pm desviación estándar) para una arquitectura de 6 neuronas en la capa oculta, para los esquemas <i>STL</i> , <i>MTL</i> y <i>MTL_{tareasecundaria}</i> para el problema del “Logic Domain”	35
3.1. Media de la probabilidad de acierto (sobre el conjunto de test) y desviación estándar sobre 30 simulaciones para diferentes arquitecturas entrenadas para aprender los datos de Ripley.	51
3.2. Probabilidad de acierto (Media y desviación estándar) para diferentes arquitecturas sobre el conjunto de datos artificial de la Figura 3.7 sobre 30 simulaciones.	52
3.3. Resumen de las base de datos utilizadas de la UCI Databases Repository.	55
3.4. Probabilidad de clasificación para el conjunto de test (media \pm desviación estándar y media de las épocas de entrenamiento) para los esquemas <i>STL</i> y “hints” para diferentes conjuntos de datos. La última columna (%Hints) representa el porcentaje de reducción de muestras de los “hints” ($L=3,5,7$) para $K=10$	56

3.5. Probabilidad de acierto (sobre el conjunto de test) y desviación estandar sobre varios esquemas.	61
4.1. Características de entrada y salida, número de muestras de entrenamiento y test de los diferentes conjuntos de datos.	85
4.2. Test de clasificación "TC" (media \pm desv. estándar) para diferentes métodos y conjuntos de datos, "NN" representa el número de neuronas en la capa oculta, "NC" es el número de conexiones que tiene la arquitectura, "TE" es el tiempo de ejecución en segundos (proceso completo: selección de la arquitectura y entrenamiento), y "CD" es la característica (o características) de entrada que el método ASELN ha descartado.	88
5.1. Descripción de las tareas del "Logic Domain". Cada una de las tareas es una combinación lógica de cuatro características de entrada.	97
5.2. Test de clasificación "TC" (media \pm desv. estandar) con diferentes arquitecturas sobre el conjunto de datos "Logic Domain", donde se han penalizado muestras añadiendo ruido en las salidas deseadas.	107
5.3. Test de clasificación "TC" (media \pm desv. estandar) con diferentes arquitecturas sobre el conjunto de datos "Iris", donde se han penalizado muestras añadiendo ruido en las salidas deseadas de la tarea principal.	109
5.4. Test de clasificación "TC" (media \pm desv. estandar) con diferentes arquitecturas sobre el conjunto de datos "Monk's Problems", donde se han penalizado muestras añadiendo ruido en las salidas deseadas de la tarea principal.	112
5.5. Número de patrones (NP) seleccionados mediante edición de datos para las tareas secundarias del problema Telugu.	113
5.6. Test de clasificación "TC" (media \pm desv. estandar) con diferentes arquitecturas sobre el conjunto de datos "Telugu", donde se han penalizado muestras añadiendo ruido en las salidas deseadas de la tarea principal.	114

Lista de Figuras

2.1. Relación entre hipótesis, espacio de búsqueda y espacio de hipótesis. La hipótesis buscada, h_0 , está contenida en el espacio de búsqueda H_1 que a su vez lo está en H_3 . H_A representa todo el espacio de hipótesis.	11
2.2. Red Multi-capa Unidireccional. La información se transmite desde los nodos de entrada hasta la capa de salida, mientras que el error producido al comparar la salida deseada con la obtenida se propaga en sentido inverso (BP).	14
2.3. Relación entre mínimo local y global. Dependiendo del punto en que nos situemos de la curva del error tendremos más o menos posibilidades de alcanzar el mínimo global.	16
2.4. Relación entre el error proporcionado por las muestras de entrenamiento y las de validación. El punto de parada óptimo para evitar el sobreaprendizaje se produce para el menor valor del error de validación.	17
2.5. Esquema básico de aprendizaje inductivo. El aprendizaje inductivo permite crear un modelo de clasificación a partir de los patrones de entrenamiento. Los patrones de test nos permitirán evaluar la capacidad de generalización del modelo.	20

- 2.6. Esquema de aprendizaje utilizando sesgo inductivo. Este esquema está basado en la ayuda que proporciona el conocimiento auxiliar que incorpora el sesgo inductivo gracias al conocimiento del dominio. 20
- 2.7. Espacio de soluciones para esquemas de aprendizaje STL y MTL. El uso de tareas relacionadas hace que el espacio de soluciones incluya la solución óptima para la tarea principal, mientras que las tareas no relacionadas dificultan el aprendizaje de dicha tarea. 26
- 2.8. Diferentes esquemas de aprendizaje. En (a) se muestra un esquema donde una única tarea se aprende en solitario (STL), mientras que en (b) se aprenden un conjunto de tareas de forma simultánea (MTL), existiendo una zona común (desde la entrada a la capa oculta) y otra específica (desde la capa oculta a la de salida) de cada tarea. 32
- 2.9. Arquitectura óptima para la tarea principal del "Logic Domain". 34
- 3.1. Procedimiento de Edición de Datos para obtener los "hints". Estos "hints" se obtienen a partir de la tarea principal usando los algoritmos $\{K, L\}$ -NN y Condensed-NN. 41
- 3.2. Aprendizaje con "hints". Cada tarea secundaria actúa como un "hints", ya que al forzar su aprendizaje simultáneamente al de la tarea principal, se producirá una transferencia de conocimiento de unas tareas a otras. 44
- 3.3. El método BP sufre algunas modificaciones. Las tareas secundarias son un subconjunto de la principal, por lo que algunas muestras no pertenecen a una determinada tarea secundaria. Cuando esto ocurre, α_k^n toma el valor 0 y el error δ_k^n no se propaga. Para la tarea principal α_1^n toma el valor 1 para cualquier muestra, ya que todas las muestras pertenecen a la tarea principal. 46

3.4. Ejemplo del procedimiento de Edición de Datos para la obtención de tareas secundarias. Se ha eliminado la zona de solapamiento mediante el algoritmo $\{K,L\}$ -NN con diferentes valores de L ($K=10$). Posteriormente, se ha seleccionado un conjunto destacado de muestras utilizando el algoritmo Condensed-NN.	49
3.5. Selección de la arquitectura. Para las dos simulaciones, conjunto completo de datos (a) y nuevo conjunto de datos obtenido mediante la Edición de Datos a partir del conjunto completo (b), se ha utilizado el mismo número de neuronas en la capa oculta, obteniéndose una frontera similar.	50
3.6. Relación del MSE (sobre el conjunto de datos de Ripley) de la tarea principal para los esquemas STL y MTL con Edición de Datos. Se puede apreciar como se acelera el aprendizaje de la tarea principal al utilizar un esquema MTL con tareas secundarias generadas mediante la Edición de Datos.	50
3.7. En esta figura se muestra un conjunto de datos creado artificialmente junto con su frontera ideal (a) para mostrar la capacidad del método propuesto para evitar mínimos locales. En (b) se muestra una situación de mínimo local en donde frecuentemente quedan atrapados los pesos de la red STL. En (c) se muestra la frontera de clasificación obtenida mediante Edición de Datos del conjunto inicial. Este nuevo conjunto será utilizado como un "hint" creando un esquema de aprendizaje MTL (d) que mejora el aprendizaje de la tarea principal.	53
3.8. Conjunto de entrenamiento de un problema de clasificación de dos clases.	58
3.9. Muestras seleccionadas utilizando la edición de datos para la tarea principal y secundaria.	59
3.10. Épocas de entrenamiento frente a probabilidad de acierto (medido sobre el conjunto de test) para diferentes esquemas.	60
4.1. Etapas del algoritmo OP-ELM.	73