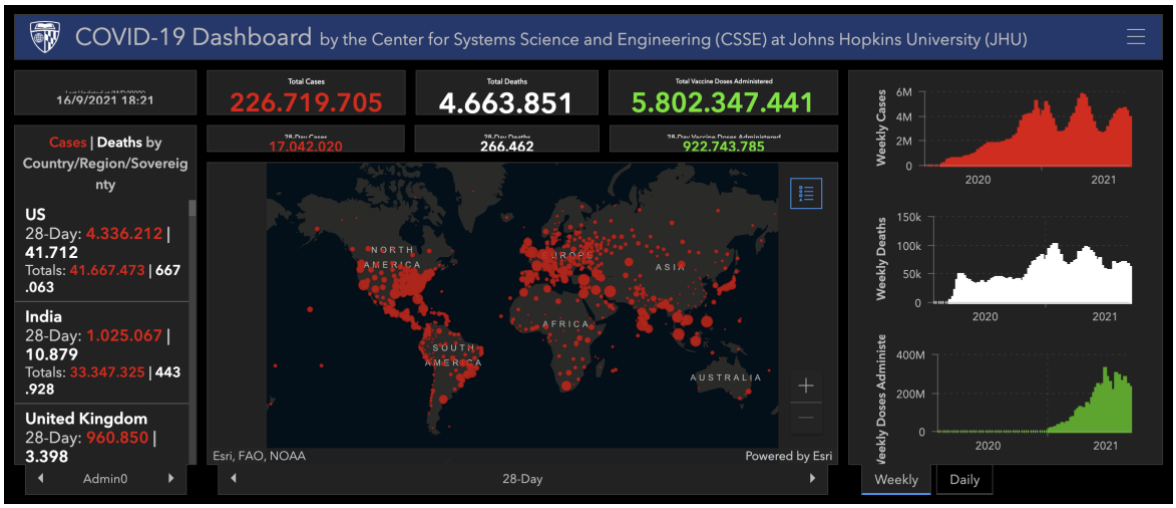
# PRÁCTICA 1
# Aplicación de RNA

## Inteligencia Artificial en las Organizaciones

### Grado en Ingeniería Informática
### Curso 2021/22

## Introducción

La situación actual que se atraviesa, causada por la pandemia global producida por el COVID-19, obliga a que estados, organizaciones y personas tomen las medidas adecuadas para combatir los efectos de dicha pandemia. Una herramienta fundamental en esta lucha es el análisis de datos y, en particular, el uso de herramientas de Aprendizaje Automático (AA), entre ellas, las Redes de Neuronas Artificiales.




Son muchas las tareas que se pueden abordar con el Aprendizaje Automático para ayudar a luchar contra la pandemia. El AA se podría utilizar, por ejemplo, para[1]:

- Identificar quién está en mayor riesgo
- Diagnosticar a los pacientes
- Desarrollar los medicamentos más rápido
- Predecir la propagación de la enfermedad
- Comprender mejor los virus
- Mapear de donde vienen los virus
- Predecir la próxima pandemia

[1] https://towardsdatascience.com/fight-covid-19-with-machine-learning-1d1106192d84

El objetivo de esta práctica es utilizar Redes de Neuronas Artificiales (RNA) para predecir la evolución de la enfermedad. Para ello, partiendo de un conjunto de datos de entrenamiento, se debe construir una RNA determinando, entre otros parámetros, su arquitectura y razón de aprendizaje. Para el uso de las RNA se utilizará el entorno de análisis de datos Weka[2] que incluye numerosas funcionalidades de análisis de datos.

El manual de Weka está disponible en este enlace[3].

# Descripción de los datos

## A. Fuente de datos

Para crear los modelos de predicción se utilizarán los datos de *The Humanitarian Data Exchange*, en particular, el *Novel Coronavirus (COVID-19) Cases Data*[4] que recopila datos epidemiológicos del COVID-19 desde el 22 de enero de 2020. Los datos son recopilados por el *Center for Systems Science and Engineering* de la Universidad Johns Hopkins (JHU CCSE) a partir de varias fuentes.

Los datos para la realización de la práctica están disponibles en:

https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases

## B. Formato de los datos

El fichero que se debe utilizar es el fichero diario de casos confirmados[5]. El conjunto de datos que tiene un total de 279 registros que corresponden a Provincias/Estados de distintos países/regiones. El conjunto de datos recopila el número de infectados totales desde el día 22 de enero de 2020 hasta la fecha. A día de hoy (16 de septiembre de 2021), hay un total de 604 atributos (columnas) por cada registro. Los primeros cuatro atributos corresponden al nombre de la provincia/estado, la región/país y la longitud y latitud de esta. El resto de los atributos reflejan el número de contagiados acumulados por día. El conjunto de datos está en formato separado por comas (.csv).

## C. Procesamiento de los datos

Una vez descargado el fichero de datos, se debe llevar a cabo el procesamiento mediante los siguientes pasos:

- Eliminar los caracteres comilla simple (‘), guión (-), asterisco (*) y paréntesis del fichero de datos dado que a la hora de importar el fichero en Weka este carácter provoca error (e.g. *Cote d'Ivoire*).

[2] Para la realización de la práctica se pueden utilizar otras herramientas si se estima conveniente.
[3] Se puede encontrar más información en Weka Wiki *(*https://waikato.github.io/weka-wiki/*)*
[4] https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases
[5] time_series_covid19_confirmed_global.csv

- Importar el fichero csv en Excel para renombrar los atributos correspondientes a las fechas. Esto se realiza para que el modelo sea compatible con nuevos datos a la hora de realizar predicciones. Es importante tener en cuenta los separadores de decimales y de miles a la hora de importar/exportar desde Excel. Weka utiliza el punto para los decimales.
- Exportar el fichero Excel a csv. Es posible que Excel utilice puntos y comas en lugar de comas a la hora de separar los campos. Se debe tener esto en cuenta a hora de abrir el fichero en Weka.
- Convertir los ficheros csv en arff. Para llevar a cabo esta tarea, es necesario abrir el fichero "csv" con el *Explorer* y guardar el fichero en formato "arff". En caso de ser necesario, recuerde invocar el cuadro de diálogo para especificar el separador de campos.
- Verificar que el fichero se haya generado correctamente.

## I Parte: Regresión (RNA-MLP)

Para resolver un problema de regresión, como el que se plantea en este caso, se pueden utilizar los *k* días anteriores de todas las regiones/países para obtener el valor de los próximos días de cualquier región/país. Cada uno de estos *k* valores serán una entrada de la red de neuronas artificiales, así como los cuatro primeros atributos del conjunto de datos. El valor que estimar será el valor de casos positivos acumulados del próximo día[6].

Una vez generado el fichero de datos en formato de Weka, se utilizará el Perceptrón Multicapa implementado en Weka para construir el modelo. Hay que tener en cuenta que Weka puede generar la arquitectura de la red. Sin embargo, esto no garantiza que se obtengan los mejores resultados. Por esta razón, es necesario experimentar con distintas arquitecturas y valores de los meta-parámetros. Por ejemplo, distintos números de entradas (e.g. variando el número de días), distintos números de nodos en la capa oculta, etc.

Para utilizar el Perceptrón Multicapa, una vez cargado el conjunto de datos, en la pestaña *Classify*, se selecciona el algoritmo *Functions.MultilayerPerceptron* que hace referencia al clasificador representado por una RNA con Modelo Perceptrón Multicapa.

El entrenamiento y testeo de la red se deberá realizar utilizando *Cross-Validation* y *Percentage Split.* Este aspecto debe analizarse en los resultados obtenidos. Además, para analizar los resultados, en "*More Options*" se puede activar la opción: *Output predictions* para mostrar las predicciones de la red.

[6] De forma alternativa, se puede utilizar el incremento de casos positivos diarios como entradas en lugar de los casos acumulados.

Analizando la salida del experimento, se obtiene la siguiente información:

- Modelo que genera el algoritmo *Functions.MultilayerPerceptron*.
- Coeficiente de correlación.
- Error cuadrático medio, error absoluto y error cuadrático relativo.
- Predicciones sobre los datos de entrenamientos

Los parámetros de la RNA también deben ser modificados para comprobar cómo varía el resultado en función de estos valores. Así, si se hace *doble click* sobre el nombre del clasificador, nos aparecerá una pantalla con distintos valores que podemos modificar. Pulsando el botón *"More"* se obtiene más información sobre todos los datos. Son interesantes los siguientes valores:

> ***GUI – True**: Muestra una representación gráfica de la RNA.*
> ***HiddenLayers:** Indica el número de capas ocultas de la RNA. Este número puede modificar considerablemente el resultado.*
> ***LearningRate:** Valor entre 0 y 1 que indica la velocidad de aprendizaje de la red. Este parámetro indica el porcentaje en que se permite varíen los pesos de la red en cada uno de los ciclos de entrenamiento.*
> ***TrainingTime:** Indica el número de ciclos que utiliza la RNA para entrenar.*

Por último, y dado que se quiere obtener pronósticos sobre la evolución de los contagios a nivel de país/región, se debe generar un nuevo fichero para llevar a cabo esta predicción que contenga[7]:

- El mismo encabezado que el fichero de datos utilizado para generar el modelo.
- Una única fila con todos los valores salvo el último valor que será la salida del modelo. Dicho valor se representa con un "?" en el nuevo fichero de datos. Dado que el nuevo fichero debe tener el mismo número de atributos que el fichero utilizado para crear el modelo, deberá eliminar el primer valor correspondiente a la primera fecha del fichero original de datos.

En la ventana de "*Classify*", en la casilla "*Supplied test set"*, se selecciona dicho fichero y se genera la predicción en base a esos "nuevos" datos.

En esta parte de la práctica, se deberá hacer la **predicción para TRES días consecutivos en España y en un país adicional a elección del grupo de trabajo.**

# II Parte: Series temporales

El ingrediente fundamental de la minería de datos son los datos. Una de las formas en que se pueden presentar los datos de un dominio o problema en particular es mediante una

[7] Deberán compararse, al menos, 6 configuraciones distintas de RNA (variando parámetros – uno a la vez) y seleccionar el mejor modelo para llevar a cabo el pronóstico. Dichas comparaciones se pueden llevar a cabo a través del *"Experimenter"* en Weka.

serie temporal. Una serie temporal es una sucesión de datos medidos en determinados momentos y ordenados cronológicamente.

Para resolver un problema de regresión en una serie temporal, se utilizan los *k* valores anteriores para obtener el próximo valor ($x_{t+1}$). Cada uno de estos *k* valores serán una entrada del algoritmo de regresión (e.g. una red de neuronas artificiales) como se aprecia en la Fig. 1.

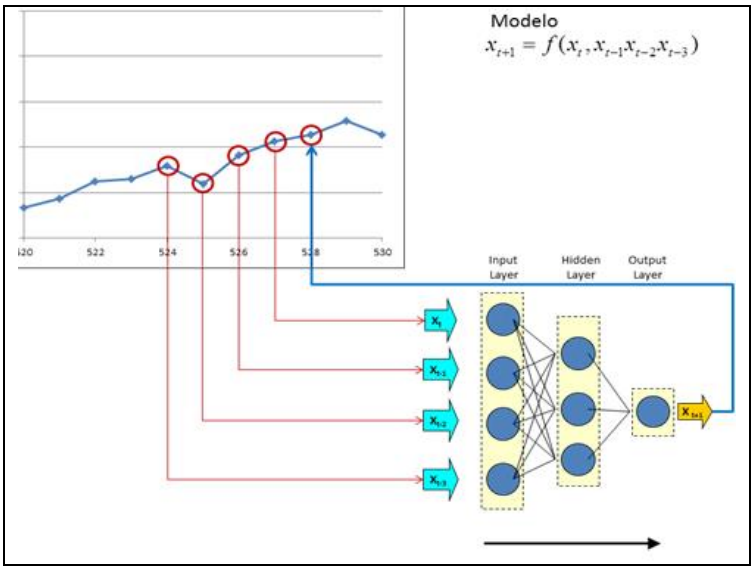



Fig. 1 Predicción en series temporales con RRNN

Lo primero que hay que hacer es generar el conjunto de entrenamiento para el algoritmo de regresión (e.g. RNA). Este conjunto se crea a partir de los datos de la serie temporal teniendo en cuenta los k valores que formarán parte de la entrada (ver Fig. 2).

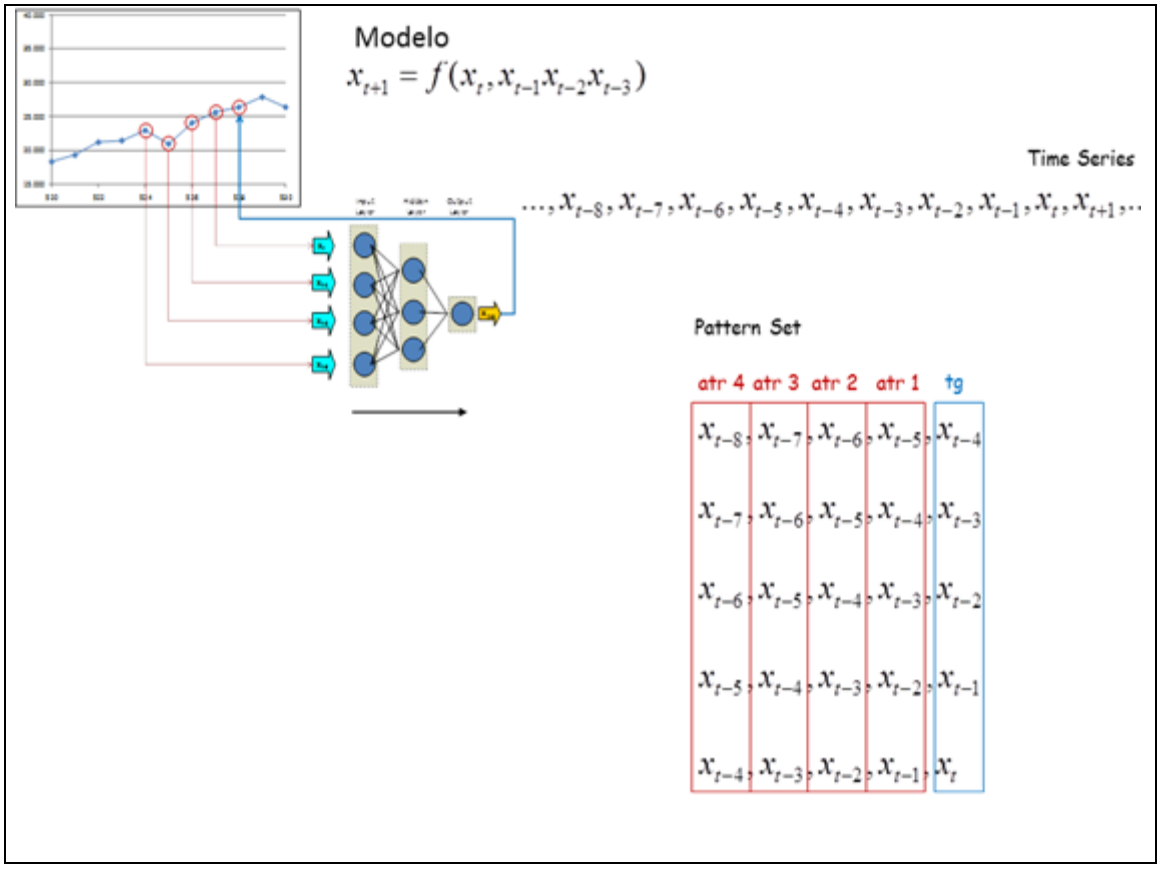



Fig. 2 Generación patrones entrada para una RNA

A pesar de que la tipología de las RNA es muy variada, en este caso, al igual que en el apartado anterior, se utiliza el modelo de RNA denominado Perceptrón y con una topología Multicapa.

Dado que las RNA tienen múltiples meta-parámetros, para determinar cuál es el modelo que mejor se ajusta a los datos, deben analizarse varias configuraciones del RNA. Para ello se podría utilizar el *Experimenter* de Weka.

# II Parte: Desarrollo

En esta parte de la práctica, se realizará la tarea de predicción utilizando el entorno de modelado de series temporales que proporciona la herramienta *Weka*. Para poder utilizar este entorno, se requiere una versión de *Weka* superior o igual a 3.7.3 y se instala como un paquete a través del gestor de paquetes.

*Tools → Package Manager → timeseriesForecasting*

En esta parte de la práctica se deberán, utilizando Redes de Neuronas Artificiales, **predecir los 15 valores siguientes** de dos conjuntos de series relacionadas con la evolución de

contagios del COVID-19 en dos países (España y el país seleccionado en la Parte I). En esta parte de la práctica, en lugar de utilizar el número de casos acumulados, se utilizarán las series temporales correspondientes al número de casos confirmados por día.

Además, se analizarán en conjunto dos series temporales de cada país dado que el paquete *timeSeriesForecasting* de *Weka* puede modelar conjuntamente varias series temporales simultáneamente y capturar posibles dependencias entre ellas. Debido a esto, modelar varias series simultáneamente puede dar resultados diferentes para cada serie que modelarlas de forma individual.

**IMPORTANTE**
El funcionamiento detallado de cada una de las funciones del paquete *timeSeriesForcasting* está detallado en:
https://wiki.pentaho.com/display/DATAMINING/Time+Series+Analysis+and+Forecasting+with+Weka

## A. Obtención de los Datos:

- Los datos se pueden obtener del sitio web de The Humanitary Data Exchange[8].
- Seleccionar los dos ficheros diarios disponibles con información completa (*confirmed global, deaths global*).
- Seleccionar dos países (España y el país seleccionado en la Parte I) sobre los que se realizará el análisis.
- Calcule el incremento diario de casos/fallecidos para cada país. Estos datos constituyen las series temporales sobre la que se deberá trabajar.
- Generar un fichero csv por cada país que se analizará. Cada fichero debe contener las dos series temporales que se desean analizar (i.e. casos confirmados y muertes).

## B. Proceso de entrenamiento:

- Una vez seleccionadas las dos series temporales (de cada país) que se van a analizar, el paquete *timeSeriesForecasting,* teniendo en cuenta las opciones seleccionadas, creará un conjunto de atributos que serán la entrada al algoritmo de regresión seleccionado (en este caso, deberá ser una red de neuronas).
- Algunos de estos atributos generados se obtienen modificando el valor de la fecha para que éste sea más representativo y relevante en la serie temporal. Así, la fecha se cambia principalmente porque la representación interna de las fechas es sólo el número de milisegundos transcurridos desde la época (y esto hace que el coeficiente sea enorme en cualquier modelo de regresión que se aprenda). Así, los valores marcados como *remapped* han sido obtenidos considerando la primera fecha observada en los datos de entrenamiento es 0.
- En el proceso de entrenamiento es necesario probar con distintas arquitecturas; es decir, distintos números de entradas (para ello se seleccionarán diferentes valores en la opción *Lag Length*), distintos números de nodos en la capa oculta, etc.

[8] https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases

### C. Predicción:

- Con cada uno de los dos modelos de RNA elegidos, se realizará la predicción de los siguientes 15 valores.
- Estos resultados deberán ser analizados y comparados.

## II Parte: Requisitos

Es importante incluir en la documentación de esta parte de la práctica lo siguiente:

- La descripción de las series temporales utilizadas.
- El proceso de entrenamiento: Las arquitecturas probadas, los problemas encontrados, etc. Además, se debe justificar la solución tomada para realizar la predicción.
- Mostrar los resultados: Mostrar gráficamente los resultados obtenidos en la fase de entrenamiento con las RNA.
- Comparación de los resultados: Pueden compararse los resultados obtenidos con una línea de tendencia adecuada.

## Evaluación de la práctica

Aspectos para evaluar en la corrección de la práctica:

- Planteamiento y desarrollo del problema: 25%
- Resultados del problema: 25%
- Análisis de resultados y conclusiones: 25%
- Presentación: 15%
- Contexto de la práctica (información complementaria sobre el desarrollo de la práctica): 10%

Debe darse importancia a la presentación para mostrar los resultados, el análisis de estos, conclusiones, etc. Es importante tener en cuenta que el contexto de la práctica en la que se incluirá cualquier información relevante conocida por los estudiantes, casos similares, noticias al respecto o cualquier otra información de interés y relacionada con la práctica.

## Entrega de la práctica

- La práctica deberá realizarse en grupos de **3/4 personas** (y entregarse únicamente por uno de los integrantes del grupo)**.**
- Esta práctica está dividida en dos partes. En este documento se detalla la primera parte. Sin embargo, la entrega de la práctica será un único documento en el que se detallen y analicen y relacionen las dos partes de la Práctica 1. La entrega será un documento con los apartados sugeridos en el punto anterior. Además, se debe incluir

en el documento un enlace a un fichero comprimido con todos los ficheros utilizados para la realización de la práctica (*.arff, *.model, etc.)

- La entrega de esta Práctica 1 (Parte I y II) se realizará con fecha máxima de entrega (por Aula Global):
  - Grupo 84: **7 de octubre – 12:00h**
  - Grupos 83: **8 de octubre – 12:00h**
- No hay un formato de documento establecido. Sin embargo, es importante que toda la información se muestre de forma clara y su presentación también se considera como parte de la nota de la práctica.

## Características de Weka

Dependiendo de la configuración del ordenador y la instalación de Weka, puede que sea necesario ampliar la memoria de la máquina virtual de Java para que Weka no se quede sin memoria durante el proceso de aprendizaje. Para ello, se debe utilizar el siguiente comando en una terminal:

```
java -jar -Xmx2G weka.jar
```

Este comando lanzará Weka con 2GB de memoria para la máquina virtual de Java. Se debe tener en cuenta que el fichero weka.jar debe estar en el `PATH` o debe incluirse el `PATH` completo. Puede adecuar el tamaño de la memoria de la máquina virtual a los requisitos de la tarea.