



UNIVERSIDAD PONTIFICIA DE SALAMANCA  
FACULTAD DE INFORMÁTICA  
Grado en Ingeniería Informática

Trabajo Fin de Grado

# **La Inteligencia Artificial aplicada a la Inteligencia Emocional**

Jorge de Andrés González

Director:  
Dr. D. Manuel Martín – Merino Acera

Salamanca, Junio de 2019



## Agradecimientos

La ejecución de este trabajo no ha sido una tarea fácil, ni he estado sólo mientras que lo desarrollaba, por lo que considero que hay varias personas que merecen un agradecimiento especial.

En primer lugar, me gustaría agradecer y dedicar este trabajo a mis padres. Siempre han estado ahí cuando lo necesitaba, me han “soportado” muchas veces en momentos en los que las cosas no iban bien y han sido un pilar fundamental para desarrollar todo esto y obtener el título de Ingeniero Informático.

Por otra parte, me gustaría agradecer a mi tutor, D. Manuel Martín-Merino por las facilidades que ha mostrado, así como por la enseñanza que me ha ofrecido sobre lo que espero que sea mi futuro laboral: La Inteligencia Artificial y la Ciencia de Datos.

También me gustaría hacer mención al resto de profesores que he tenido en la carrera. La mayoría de ellos me han enseñado diferentes conceptos y tecnologías que han aportado granitos de arena para la construcción de este proyecto, por lo que también les doy las gracias.

Jorge



## Resumen

La categorización de trastornos de inteligencia emocional es realizada habitualmente por psicólogos expertos en la materia. Este proceso requiere tiempo y gran capacidad de diálogo, por lo que es interesante crear un sistema automático que permita asesorar al humano a la hora de diagnosticar.

En este trabajo, se han explorado técnicas avanzadas para la recogida de datos que caractericen los trastornos, y técnicas de data science y machine learning complejas para su correcta categorización.

Así mismo, se han obtenido numerosas imágenes que explican los resultados obtenidos a través de gráficos fácilmente interpretables por expertos humanos.

Junto con esto, se ha desarrollado una base de datos MongoDB desde la que se pueden obtener los datos, y un backend en NodeJS para poder obtenerlos mediante llamadas API.

Los resultados indican que la elección correcta de las variables, junto con un preprocesado adecuado y técnicas de machine learning avanzadas permiten categorizar, de manera genérica, los trastornos psicológicos de una manera precisa, con errores que llegan a mínimos de 6.5%. Estos errores ratifican que estos sistemas podrían asesorar a los expertos humanos.

## Abstract

Obtaining categories of emotional intelligence diseases is a task often made by psychologists experts in the subject. This process is time consuming and requires a lot of dialog, so creating an automatic system to advise the human is interesting.

In this work, advanced techniques for data collecting and picking are used, as well as data science and machine learning techniques for the categorizing properly.

As well, many images have been obtained that explain the results by easy graphics that are understandable by human experts.

Alongside, a MongoDB database has been developed, as well as a NodeJS backend for obtaining the data.

Results make us figure out that the right choosing of the variables, alongside with a proper pre processing and advanced machine learning techniques let us categorize, in a generic way, psychological diseases in a precise way, with errors that decrease up to 6.5%. This errors support that this systems could advice human experts.

## Descriptores

Small Data, Data science, Inteligencia Artificial, Psicología, Psicopatología



## Índice

Agradecimientos .....	iii
<b>Introducción .....</b>	<b>1</b>
Presentación y Motivación del Trabajo .....	1
Estado del Arte.....	2
<b>1. Capítulo 1: Fundamentos de Psicología.....</b>	<b>5</b>
1.1 La Psicología Conductista .....	6
1.2 La Psicología Cognitiva .....	7
1.3 La Psicología Cognitivo-Conductual .....	11
1.4 Los Trastornos Psicológicos y la Psicopatología .....	13
1.4.1 Los trastornos psicológicos.....	13
1.4.1.1 Trastornos infantiles .....	14
1.4.1.2 Trastornos de ansiedad .....	14
1.4.1.2.1 Trastorno Obsesivo Compulsivo ( <i>TOC</i> ).....	15
1.4.1.3 Trastornos del estado de ánimo .....	15
1.4.1.3.1 Trastornos Depresivos.....	15
1.4.1.4 Trastornos sexuales.....	15
1.4.1.5 Trastornos de la conducta alimentaria .....	16
1.4.1.6 Trastornos de la personalidad .....	16
1.4.2 En este trabajo: Grupos de trastornos psicológicos.....	16
1.4.3 Distorsiones de la percepción de la realidad: Las distorsiones cognitivas .....	17
1.4.4 En este trabajo: Distorsiones cognitivas usadas .....	21
1.4.4.1 En este trabajo: Otras variables usadas .....	21
<b>2. Capítulo 2: Data Science.....</b>	<b>23</b>
2.1 Data Mining Vs Machine Learning Vs Data Science Vs Big Data .....	23
2.1.1 Data Mining .....	23
2.1.2 Machine Learning.....	25
2.1.3 Data Science .....	26
2.1.4 Big Data .....	27
2.2 Antes de hacer Data Mining .....	29
2.2.1 En este trabajo: Obtención de los datos .....	29
2.3 Data Mining .....	30
2.3.1 Pasos previos y preparación de los datos.....	30
2.3.2 En este trabajo: Preparación de los datos .....	35

2.3.3	Análisis Exploratorio o Descriptivo .....	37
2.3.3.1	Resumen de las estadísticas del Dataset.....	37
2.3.3.2	OLAP y Análisis Multidimensional.....	40
2.3.4	Machine Learning .....	42
2.3.4.1	¿Cómo funciona un algoritmo de clasificación en machine learning?.....	42
2.3.4.2	Problemas y soluciones con los clasificadores .....	43
2.3.4.3	Algoritmos Supervisados.....	47
2.3.4.3.1	K Nearest Neighbours (KNN) .....	48
2.3.4.3.2	Árboles de Decisión.....	50
2.3.4.3.3	Regresión .....	51
2.3.4.3.4	Support Vector Machines (SVM).....	53
2.3.4.4	Algoritmos no Supervisados .....	55
2.3.4.4.1	K Means .....	55
2.3.4.4.2	Reglas de Asociación .....	60
2.3.4.5	Algoritmos Semi-Supervisados .....	64
2.3.4.6	Algoritmos de Aprendizaje por Refuerzo .....	65
2.3.4.6.1	Q-Learning .....	65
2.3.4.6.2	Diferencia Temporal (TD) .....	66
2.3.4.7	Algoritmos de Redes Neuronales y Deep Learning.....	69
2.3.4.7.1	Redes Neuronales .....	69
2.3.4.7.2	Deep Learning .....	73
2.3.4.7.2.1	Redes Convolucionales .....	74
2.3.4.7.2.2	Redes Recurrentes.....	75
2.3.5	Visualización .....	76
<b>3.</b>	<b>Capítulo 3: REST .....</b>	<b>85</b>
3.1	MongoDB .....	86
3.2	NodeJS.....	88
<b>4.</b>	<b>Capítulo 4: Resultados Obtenidos y Conclusiones Finales .....</b>	<b>89</b>
4.1	Resultados Obtenidos .....	89
4.2	Conclusiones.....	103
4.3	Líneas Futuras, Ampliaciones y Entornos de Aplicación .....	104
	<b>Anexo de Términos .....</b>	<b>105</b>
	<b>Bibliografía.....</b>	<b>107</b>



## Índice de Figuras

1-1. Resumen Economía de Fichas .....	13
1-2. Ejemplo Maximización .....	19
2-1 Proceso KDD detallado .....	24
2-2. Subdivisiones de la Inteligencia Artificial .....	26
2-3. Las 3 V's del Big Data .....	28
2-4. Reducción Dimensionalidad Dataset Iris .....	33
2-5. Percentiles sobre una normal .....	38
2-6. Ejemplo Data Cube .....	41
2-7. Underfitting, Óptimo y Overfitting .....	44
2-8. Ejemplo 5-fold Cross Validation .....	46
2-9. KNN .....	48
2-10. Estructura básica de un árbol de decisión .....	50
2-11. Regresiones .....	52
2-12. SVM Linealmente Separable .....	53
2-13. SVM No Lineal .....	54
2-14. K-Means paso a paso .....	56
2-15. Ejemplo en inglés de tabla de reglas .....	62
2-16. Ejemplo de Tabla Q .....	66
2-17. Red Neuronal Multicapa Feed Forward .....	71
2-18. Esquema de Red Convolutacional .....	74
2-19. Red SRN vs Red LSTM .....	75
2-20. Mismos datos, diferentes escalas .....	77
2-21. Clustering Colores .....	78
2-22. Gráfico de Líneas .....	80
2-23. Explicación de BoxPlot .....	82
2-24. Ejemplo Cartograma .....	83
3-1. Petición GET en API REST .....	86
3-2. Logo de MongoDB .....	87
3-3. Entrada/Salida Bloqueante Vs Entrada Salida no Bloqueante .....	88
4-1. Wordcloud Nombres .....	89
4-2. Scatterplot Edad - Sexo .....	90
4-3. Inhibido Vs Impulsivo .....	90
4-4. Matriz de correlación .....	91
4-5. Estadísticos principales del Dataset .....	92
4-6. Distribución Pacientes Componentes Principales .....	94
4-7. Precisión de Entrenamiento Deep Learning .....	101
4-8. Dendrograma Final .....	102



## Índice de Tablas

Tabla 2-1. Ejemplos Reglas Finales .....	60
Tabla 4-1. Resultado de Redes Neuronales.....	95
Tabla 4-2. Matriz Confusión Test KNN .....	96
Tabla 4-3. Matriz Confusión Random Forest .....	97
Tabla 4-4. Matriz Confusión SVM .....	98
Tabla 4-5. Matriz Confusión KNN Balanceado.....	99
Tabla 4-6. Matriz Confusión Random Forest Balanceado .....	100
Tabla 4-7. Matriz Confusión SVM Balanceado .....	100
Tabla 4-8. Mejores Resultados Finales.....	102



# Introducción

---

## Presentación y Motivación del Trabajo

El presente trabajo versa sobre la predicción de trastornos psicológicos mediante el uso de data science, y más concretamente, técnicas de inteligencia artificial. Para conseguir este objetivo, se hará un análisis exhaustivo tanto de las diferentes distorsiones cognitivas que producen los trastornos psicológicos, como de los trastornos psicológicos en sí, y posteriormente otro análisis de las diferentes técnicas y pasos del proceso de data science que hay que llevar a cabo para conseguir unos resultados óptimos.

Es interesante este problema debido a la resolución de un problema real, puesto que todos los psicólogos al inicio de sus terapias primero deben de identificar las diferentes distorsiones cognitivas que posee el paciente para poder entonces valorar el trastorno psicológico que sufren. El tratamiento de la inteligencia emocional mediante técnicas de inteligencia artificial es extremadamente interesante, puesto que ambas inteligencias poseen numerosas características en común, como la capacidad de aprendizaje, y el uso de una para determinar los problemas de la otra resulta en un problema interdisciplinar muy atrayente.

Se podría decir que es un problema altamente complicado debido a la existencia de datos reales en muy pequeña cantidad, por lo que las cantidades de ruido y *outliers* principalmente se plantean como dos impedimentos de gran magnitud. Además, la pequeña cantidad de pacientes que se han podido recoger a mano plantea una dificultad extra hacia los *modelos*, puesto que prácticamente hay solo 3 veces más de pacientes que dimensiones tiene el problema.

También, es un problema motivante debido a la actualidad de ambos temas, puesto que la inteligencia artificial está presente en todo nuestro entorno, desde los móviles hasta internet, pasando por televisores, sistemas recomendadores o incluso la mayoría de procesos de ingeniería. También está en auge el uso de la misma para la transformación digital de los negocios, como las data driven enterprises. Por otra parte, la psicología es un tema que lleva en auge numerosos años, pero cada vez en mayor medida debido al aumento de los trastornos psicológicos en nuestra sociedad debido al ritmo de vida de los adultos y los cambios en educación hacia los niños.

En resumidas cuentas, debido a todo esto, creo que el problema que se plantea es un reto importante debido a las dificultades técnicas del mismo, pero a la vez muy interesante dados los temas, su actualidad y la conexión que se va a establecer entre ellos.

## Estado del Arte

El objetivo del trabajo es la construcción de un sistema automático de predicción de trastornos psicológicos en base a una serie de datos de pacientes reales, obtenidos a mano a partir de una consulta privada profesional de psicología. Este problema, tras una larga consulta en bibliografía, no ha sido encontrado, por lo que se puede afirmar que es el primero en abordar este problema de esta manera.

Junto con este primer objetivo de la construcción del sistema, también se hará un análisis exhaustivo de los datos para su correcta interpretación, además de la obtención de características interesantes de los mismos que puedan estar ocultas.

La metodología que se seguirá será, en primera instancia, la obtención de los datos a mano, debido a la falta de repositorios con datos reales de los mismos.

Posteriormente, se hará un estudio exhaustivo de los diferentes métodos y técnicas que se pueden utilizar para el análisis, clasificación y predicción de los datos, dejando estos conocimientos plasmados en la memoria y posteriormente implementados en código.

Finalmente, se hará un estudio de las diversas técnicas de visualización de los datos, y se procederá a la visualización de características interesantes de los mismos mediante herramientas profesionales de visualización.

Como elemento global, para el control de los errores principalmente, usaré un sistema de control de tareas y errores muy similar a Kanban llamado Glo, ofrecido gratuitamente por GitKraken. Además, todo el código y la memoria estará continuamente siendo versionado mediante un VCS (GitHub en este caso) para el control del mismo y de los cambios, así como la sincronización entre los ordenadores en los que se pueda trabajar y como medida preventiva de "Backup", puesto que durante el desarrollo este proyecto estará como repositorio privado.

Analizando los resultados esperados, es importante tener en cuenta de que este no es un "problema de laboratorio" con unos datos controlados, sino que es un problema real con datos reales que incluso algunos psicólogos profesionales sin una preparación óptima tienen problemas para resolver. De este modo, espero que los resultados no estén cerca del 100% de acierto, aunque bien es cierto de que, ante las variables escogidas cautelosamente, y el tratamiento de los datos, espero obtener porcentajes de acierto que estén alrededor del 70% - 80% en los algoritmos que más se adapten al problema.

Atendiendo a la organización de la memoria, esta comenzará con un primer capítulo introductorio de psicología. En él, se analizará el término de psicología en sí y las dos vertientes que han tenido que existir para llegar al enfoque actual y desde el cual se analizan los trastornos psicológicos, el cognitivo-conductual, el cual también será explicado.

Posteriormente, habrá un extenso capítulo sobre data science, en el cual se abordarán de una manera muy pormenorizada todos los pasos que se deben de tener en cuenta para un

análisis de datos completo y fiable, haciendo especial hincapié en la zona de algoritmos de machine learning, que consistirá en el núcleo del trabajo. También habrá un apartado de visualización.

En el siguiente capítulo, se hará un análisis de todos los resultados obtenidos mediante las diferentes técnicas, de tal manera que se pueda comparar la eficacia de los algoritmos entre sí, e incluso de un mismo algoritmo con diferentes parámetros.

Finalmente, habrá un anexo de términos que puedan resultar desconocidos y que no se hayan explicado en profundidad, y una bibliografía del trabajo.





# 1. Capítulo 1: Fundamentos de Psicología

---

Empezando desde el principio nos podemos preguntar: ¿Qué es la psicología? Y no es una respuesta fácil de dar, debido a que el campo que abarca la psicología es muy amplio y profundo. La mayoría de expertos darían una definición cercana a: “Es la ciencia que investiga y trata la conducta y los procesos que se llevan a cabo en la mente”, pero esta definición no es suficientemente concreta, aunque puede ser válida.

La psicología trata numerosos ámbitos, desde explicar cómo percibimos información y como la procesamos, hasta como nos relacionamos con otras personas en las diferentes situaciones que se pueden dar en nuestra vida. También, la psicología da cabida y respuesta a todas aquellas distorsiones mentales que pueden darse y que derivan en distorsiones emocionales, que se reúnen bajo el nombre de psicopatologías.

La psicología tiene tres vías principales de investigación, que son las siguientes:

## 1) Psicología del desarrollo

La psicología del desarrollo trata de estudiar el desarrollo humano desde la niñez hasta la vejez, teniendo en cuenta todos los factores ambientales, culturales e individuales de cada persona.

## 2) Psicología fisiológica y la neurociencia

Esta rama de la psicología investiga las bases del comportamiento humano a partir de los efectos que producen elementos naturales en nuestro cerebro, actuando como repartidores de información. Estos mensajeros son principalmente hormonas.

También, esta rama estudia toda la farmacología que pueda estar relacionada con la mente humana, como pueden ser los medicamentos psicoactivos (como los recetados contra la depresión o los calmantes) o las llamadas drogas sociales (alcohol, tabaco o marihuana principalmente).

### 3) Psicología experimental

La psicología experimental se centra en responder una serie de cuestiones relacionadas con el aprendizaje humano, la memoria y las emociones entre otros elementos. Algunas de las preguntas que intentan responder son:

- ¿Qué es lo que nos hace olvidar cosas?
- ¿Las emociones son universales o son personalizadas?
- ¿La cultura influye en las emociones?

De esta manera, podemos ver que la psicología es una ciencia activa y con numerosas ramas, pero debido a la naturaleza del trabajo, a partir de ahora sólo me centraré en la primera de ellas, la rama de la psicología del desarrollo.

Dentro de la psicología del desarrollo, al igual que con numerosas ciencias a lo largo de la historia, ha habido diferentes convicciones sobre lo que es la verdad. A continuación, veremos las tres ramas más importantes de la historia de la psicología, lo que enunciaban y cuáles eran sus puntos fuertes y débiles.

## 1.1 La Psicología Conductista

En psicología, la rama del conductismo es aquella que estima que el estudio que debe hacer la psicología debe ser sobre únicamente sobre los comportamientos observables y los efectos que puedan tener los estímulos que rodean a la persona sobre estos comportamientos. El conductismo nació de la mano de John Broadus Watson (1878-1958). Watson, en la entrevista que se considera el inicio del conductismo (1913), afirma que la psicología debería de convertirse en una rama totalmente científica, y que para ello lo que debería de hacer es centrarse en el análisis de las conductas totalmente visibles de las personas, en vez de divagar entre estados mentales y la diferencia de conceptos como conciencia o mente.

Para Watson, así como para toda la vertiente conductista, los seres humanos somos “cajas negras” cuyo interior nunca es observable, y cada estímulo que llega es procesado de una manera desconocida, obteniendo finalmente una respuesta por parte de la persona. Watson sostiene que, al ser este procesamiento inobservable, no debe de ser estudiado ni tenido en cuenta.

Esta es una posición muy radical dentro de la psicología, y como no podía ser de otra manera, otros psicólogos conductistas fueron matizando posteriormente estas afirmaciones, aseverando que los procesos que tenían lugar dentro del cuerpo sí tenían una gran importancia, pero que la psicología no tenía que tenerlos en cuenta para poder tener explicaciones sobre la conducta humana.

Uno de los elementos más importantes del conductismo es su oposición al concepto de “enfermedad mental”. Es decir, según las raíces de esta vertiente, no pueden existir conductas patológicas, ya que estas conductas que tiene un ser humano siempre han de valorarse respecto a la adecuación de las mismas a un contexto. Así, los conductistas sostienen que las enfermedades deben de ser patologías bien aisladas y definidas.

Esto nos lleva a que los psicólogos conductistas se opongan frontalmente al uso de fármacos para poder tratar algunos problemas psicológicos como las fobias o las alucinaciones.

Algunos de los elementos básicos del conductismo son:

- Estímulo
- Respuesta
- Condicionamiento
- Refuerzo
- Castigo

Se explican brevemente a continuación:

El estímulo es cualquier señal, elemento o mensaje que produce una reacción, conocida como respuesta, en un organismo. En ese momento, al generar la respuesta, automáticamente tenemos un condicionamiento, que consiste en un aprendizaje que se deriva de la asociación entre estímulos y respuestas.

Una vez que la respuesta ha sido dada, acorde con nuestro condicionamiento, podemos entrenarnos para obtener diferentes respuestas las próximas veces. Esto lo haremos mediante refuerzos y castigos.

Los refuerzos son premios, actitudes, o cualquier elemento que nos invita a seguir manteniendo una cierta conducta al recibir un estímulo.

Los castigos son la oposición a los refuerzos. Como su propio nombre indica, consiste en cualquier elemento o acción que nos invita a no seguir manteniendo la respuesta obtenida ante un estímulo.

El conductismo empezó a entrar en declive a partir de los años 50, cuando surgió el cognitivismo. En esas fechas, las teorías conductistas estaban muy suavizadas respecto al discurso original de Watson.

El cognitivismo surgió como un modelo puramente teórico, y fue una reacción frontal al análisis de sólo las conductas observables del conductismo, dejando aparte la cognición de las situaciones. Este cambio es conocido como la “revolución cognitiva”. Esta revolución, entre otras cosas previamente comentadas, surgió por un conjunto de anomalías empíricas en el conductismo que dieron lugar a una gran deceleración en diversas líneas de investigación y desarrollo.

## 1.2 La Psicología Cognitiva

La psicología cognitiva es una rama de la psicología, encargada de estudiar, tal como su nombre dice, la cognición. Se entiende como cognición el conjunto de procesos mentales que están implicados en la obtención del conocimiento al ser humano y que, por lo tanto, no son observables. Se podría decir que la psicología cognitiva es esa pieza de la mitad del puzle que le faltaba a la psicología conductual.

Como ya he expuesto anteriormente, entendemos el inicio de la psicología cognitiva en la década de 1950 en Estados Unidos aproximadamente, donde había una serie de teorías del aprendizaje y un sistema de psicología conductista que no acababa de cuadrar a la sociedad, y que se quedaba estancado en las experimentaciones.

Con la psicología cognitiva se obtiene el concepto de “representación mental”, pieza clave de esta variante de la psicología debido a su carácter central y a la posibilidad de poder operar

con las mismas. Estas representaciones mentales tienen que ser analizadas de una forma separada, pero esto para los cognitivistas no es excusa para no tenerlas en cuenta a la hora de analizar el comportamiento humano. Además, todo esto coincide con una disminución de la importancia del contexto, sea afectos, cultura o historia, lo que centra aún más en el interior a esta vertiente psicológica. Es importante tener en cuenta que, al contrario que la conductual, totalmente restrictiva, la cognitiva no cierra la puerta de inicio a los factores externos, pero sí es verdad que los considera una parte secundaria de la psicología humana. Si se sigue esta vertiente, la investigación psicológica se facilita enormemente.

Para la investigación cognitiva hubo varios avances tecnológicos que facilitaron la supremacía de esta teoría a partir de los años 50:

a. Los avances en informática y cálculo:

Personas como Alan Turing (1912 - 1954) tuvieron mucho que ver indirectamente con los avances en psicología, porque las máquinas que crearon eran programables. Esto significa que estas máquinas pueden seguir una serie de pasos y finalmente tomar decisiones, tal como los seres humanos. Por ello, para el estudio del pensamiento humano estas máquinas tuvieron una importancia capital.

b. Los avances en cibernética

En cibernética, podemos destacar a Norbert Wiener (1894 – 1964), quien construyó servomecanismos. Estos elementos son aparatos que son capaces de mantener un cierto rumbo dependiendo sólo de factores externos. Hacían cálculos de variaciones del exterior y, mediante un sistema de retroalimentación, podían calcular internamente los cambios a hacer y ejecutarlos, funcionando de una manera similar a la teoría cognitiva. Aunque, como se desarrollará posteriormente, estos aparatos son muy cercanos a la teoría cognitivo-conductual.

c. Los avances en la teoría de la información

En este ámbito destacó mucho Claude E. Shannon (1916 – 2001), que hizo grandes aportes a este ámbito. Shannon afirmaba que la información no era más que “una poda de las diferentes alternativas mediante elecciones, de una forma totalmente separada a los contenidos concretos que la forman”.

Es interesante que la unidad básica para Shannon es el bit, ya que según su teoría la información se construye a partir de dos alternativas posibles.

De esta manera, y tal como se ha comentado anteriormente, los elementos sobre los que se apoya la psicología cognitiva tienen que ser elementos que estén principalmente en el interior de la persona, y que no sean tangibles ni observables. Por ello, y a la vista de los diferentes experimentos que ayudaron al cognitivismo a seguir adelante, se pueden definir dos elementos como base de esta vertiente:

### 1) La representación de la información

Una definición simple pero acertada de representación es la que da Jean Matter Mandler (1929 - ) afirmando que “la representación es información almacenada por un sistema mental y dispuesta para ser utilizada por ese sistema”. No es una definición aceptada unánimemente, pero es simple y lo suficientemente precisa para usarla durante este trabajo.

Así, según esta autora, representación y conocimiento son dos conceptos realmente unidos entre sí, aunque enfatiza especialmente en que la representación es el formato en el cual se almacena el conocimiento. Es importante enfatizar que, al igual que en los ordenadores, para que haya una representación de la información primero hay que procesarla y hacer una serie de transformaciones. Pero, a partir de este punto, lo que puede pasar con la información puede seguir múltiples caminos:

Puede ser de diferentes tipos: Implícito o explícito, proposiciones o imágenes...

Puede ser de diferente nivel de abstracción: La información se puede representar de una manera muy simple y cercana a nuestra percepción visual (como ocurre a la hora de aprender las letras del abecedario), o de una manera muy compleja y elaborada, como ocurriría a la hora de razonar y memorizar las diferentes vías de resolución de un problema.

### 2) El procesamiento de la información

Este pilar de la psicología cognitiva quizás es el más importante, ya que las corrientes de estudio de esta rama llegaron a abordar casi por completo el estudio de la psicología cognitiva.

En el procesamiento de la información no hay una teoría unificada y aceptada por todos, ni siquiera por una mayoría, tal como puede suceder con otras teorías. En cambio, la teoría del procesamiento de la información está conformada por un conjunto de teorías muy diversas. Pero, a pesar de ello, todas estas teorías comparten una base común y unas características generales. Estos son:

- a. Los fenómenos cognitivos en los seres humanos son bastante parecidos a los fenómenos que regulan el funcionamiento de los ordenadores:

Esto es lo que se ha llamado la “metáfora del ordenador”, e indica que la forma en la que las personas procesan la información es muy similar a la forma en la que un ordenador la procesa. Esto se puede ver en diferentes conceptos, como:

1. Ambos tienen que hacer conversiones a un lenguaje que entienden:

En el caso del ser humano, tenemos que configurar las representaciones mentales anteriormente descritas, mientras que en el ordenador se traduce a lenguaje máquina, es decir, lenguaje binario.

2. Ambos tienen que actuar sobre la información ya transformada en el paso anterior:

En el caso del ordenador, el procesador ejecuta las acciones que le llegan en lenguaje binario, mientras que en el caso del ser humano la mente ejecuta acciones a partir de estructuras conceptuales en la mente.

3. Ambos dan respuestas hacia el exterior a través de elementos que están fabricados para tal uso:

En el caso del ordenador, emite una respuesta a través de los denominados periféricos de salida (pantalla, impresora...) mientras que en el ser humano se puede hacer mediante la voz o el movimiento.

4. Una cantidad ciertamente pequeña de procesos básicos subyace a toda la cognición:

Se podría decir que la actividad cognitiva que procesa la información entre la llegada del estímulo y la emisión de la respuesta se puede subdividir en elementos más básicos, que pueden ser subdivididos a su vez. Así, se puede simplificar el problema de la cognición a axiomas y componentes fundamentales cognitivos.

Esto, en los ordenadores, se puede ver en las arquitecturas RISC en los procesadores, donde un conjunto reducido de instrucciones es capaz de hacer todas las funciones de un procesador, ante una arquitectura CISC que tiene más instrucciones. Estas instrucciones subyacen a todos los procesamientos que hace una CPU.

Por parte de la psicología, todavía no se ha llegado a una conclusión y aceptación de cuáles son estos elementos fundamentales de la cognición, pero sí se está de acuerdo en que esta subdivisión a elementos más simples es posible.

- b. Los procesos que son individuales pueden cooperar y ejecutarse de manera organizada:

En el caso de la informática, el procesador puede hacer un número de instrucciones muy reducido a la vez (dependiendo principalmente del número de núcleos que tenga, o de tecnologías como el Hyper-Threading de Intel). Estas operaciones de poco sirven si no se juntan entre ellas para formar operaciones más complejas, y con ello finalmente formar rutinas y programas de muy alto nivel.

En psicología ocurre el mismo fenómeno. La comprensión de elementos fundamentales en la cognición, y la puesta en conjunto de ellos es lo que hace que hacemos una determinada acción como humanos. En este caso, las relaciones y el orden en el que se perciban los estímulos son vitales, ya que hay percepciones más importantes que otras, debido a que muchas veces se pueden clasificar de forma jerárquica.

- c. El procesamiento tiene supuestamente limitaciones:

En el caso de la informática, como se ha comentado anteriormente, el número de operaciones que se pueden realizar al mismo tiempo es limitado. De esto, podemos inferir que hay un máximo en el número de tareas que podemos procesar en una unidad de tiempo. También se tiene que tener en cuenta de que, de manera simplificada, por cada golpe de reloj del procesador podemos procesar un bit por cada núcleo, lo cual no significa que completemos una cierta tarea. Con esta afirmación se puede intuir que completar tareas suele conllevar un cierto tiempo.

Además, hay tareas que necesitan procesarse de una forma secuencial (por ejemplo, para poder tener un dato en memoria que le da una tarea anterior), de tal manera que pueden tener que esperar a la finalización de otra para poder ejecutarse.

En psicología, respecto a la mente humana, hay diferentes tareas que se pueden tener que llevar a cabo. Cada una de estas tareas demanda un "procesamiento" variable en nuestra mente, consumiendo una cantidad también variable de recursos. Así, el ser humano tiene la capacidad de ordenar las tareas en "automáticas" y "con esfuerzo", dependiendo de la cantidad de recursos que consuman.

Además, la mente humana al igual que un procesador de ordenador puede procesar diversas tareas de forma simultánea si no necesitan de otra y consumen pocos recursos, mientras que si esto no es así la mente podrá hacerlo secuencialmente.

### 1.3 La Psicología Cognitivo-Conductual

La psicología cognitivo-conductual es aquella que junta las bases de la teoría cognitiva de la psicología, y de su anterior vertiente conductual. Nace de 5 hechos primordiales:

#### 1) El condicionamiento clásico

Investigado por el filósofo ruso Ivan Pavlov (1849 – 1936), se basa en que los individuos pueden relacionarse de una manera predictiva entre los diferentes estímulos que plantea el ambiente.

En el experimento de Pavlov, se sabía que los perros al darles comida generaban una respuesta en forma de salivar. Para conducir el experimento, Pavlov empezó a tocar una campana antes de dar comida al perro, de tal manera que el estímulo de la campana acabó haciendo salivar al perro sin llegar a ver la comida; es decir, un estímulo neutro que nada tenía que ver con la comida acabó produciendo la respuesta en el perro.

#### 2) El condicionamiento operante

Investigado por Burrhus Frederick Skinner (1904 – 1990), el condicionamiento operante se basa en el hecho de que las conductas del ser humano se pueden adquirir, se pueden mantener y se pueden extinguir. Así, el ser humano asocia comportamientos con consecuencias.

Este condicionamiento operante tiene uno de los pilares en la teoría de la economía de fichas, que será explicada más adelante en este trabajo.

#### 3) El aprendizaje social u observacional

Investigado por un grupo liderado por Albert Bandura (1925 - ), la teoría del aprendizaje social conjunta una serie de hipótesis mediante las cuales se afirma que el aprendizaje no solo viene de la experiencia de la propia persona, sino también de la información que puede recibir la persona mediante estímulos auditivos o visuales entre otros.

Se lleva a cabo a través de dos elementos:

a. Moldeamiento

Consiste en el proceso de observar e imitar un comportamiento en concreto que se ha visto en otra persona.

Un ejemplo de esto es el niño que ve a su padre ponerse la corbata, y quiere imitarle poniéndose una.

b. Neuronas Espejo

Las neuronas espejo son un conjunto de neuronas que, cuando se observa a una persona realizando una acción, emiten una serie de descargas eléctricas que impulsan a la persona a repetir la acción.

Un ejemplo de esto se da con los recién nacidos, con los que la acción de sacarles la lengua es imitada por ellos.

4) El trabajo de Beck y Ellis

Aaron Temkin Beck (1921 - ) y Albert Ellis (1913 – 2007) usaron los tres principios anteriormente explicados del condicionamiento clásico, condicionamiento operante y aprendizaje social para crear el enfoque cognitivo-conductual de hoy en día.

5) La visión incompleta de las dos teorías anteriores

Tal como se ha comprobado anteriormente, las dos ramas (cognitiva y conductual) son un puzle incompleto. Repasando, la conductual sería el principio y el final del puzle, obviando el centro, mientras que la cognitiva sería el núcleo del puzle, sin tener demasiado en cuenta el inicio y el fin. Por ello, la unión de las dos teorías ofrece una visión mucho más completa del individuo y de su comportamiento.

Uno de los elementos más importantes que se heredan en la psicología cognitivo-conductual de la psicología cognitiva es el llamado “aprendizaje por economía de fichas”. En nuestra vida, toda acción conlleva una reacción. Por ejemplo, si alguien roba, se le multa para que obtenga un castigo y deje de hacerlo. Si alguien trabaja y es responsable en la empresa, seguirá cobrando y es posible que obtenga un ascenso para premiar su dinámica positiva. Esto es la base de la economía de fichas, los llamados “castigo” y “refuerzo” respectivamente.

Así, una persona recibirá un refuerzo cuando tras hacer una acción es premiada por ello, lo que se denomina “refuerzo positivo”. Con ello, la persona tenderá a repetir más la conducta. También, un refuerzo consiste en la evitación de un castigo tras hacer una acción. En este caso, el hecho de no recibir algo desagradable es algo que impulsará a repetir esa acción en el futuro, y recibe el nombre de “refuerzo negativo”.

Por la otra parte, una persona recibirá un castigo si tras hacer una acción recibe algo desagradable. Con esto, la persona tenderá a extinguir la existencia de dicha conducta. También, se puede considerar castigo el hecho de que una persona se quede sin algo agradable tras una acción, lo cual también impulsará a no repetir la acción en situaciones futuras.



Lo podemos ver resumido en la siguiente figura:

	AGRADABLE	DESAGRADABLE
DAR	REFUERZO POSITIVO	CASTIGO
QUITAR	CASTIGO	REFUERZO NEGATIVO

1-1. Resumen Economía de Fichas

Otro elemento muy importante de la psicología cognitivo-conductual, basado en el punto 5 anterior, es la sucesión siguiente, conocida como “registro cognitivo-conductual”:

1. Situación
2. Pensamiento
3. Emoción
4. Conducta

Este registro, si nos fijamos, en los puntos 1 y 4 tenemos la vertiente conductual, y en los puntos 2 y 3 la cognitiva.

Este esquema es fundamental, ya que todos seguimos esta secuencia a la hora de actuar. Primeramente, nos encontramos en una situación en la vida: pongamos que estamos en un restaurante abarrotado donde no hay aire acondicionado. Lo primero que hacemos es pensar, y un pensamiento posible ante esta situación sería algo como: “Me estoy agobiando, creo que me voy a desmayar”. En este momento, nuestro cuerpo sufre una serie de emociones, o de reacciones físicas, que en nuestro caso sería un aumento de la cadencia de respiración y un gran agobio. Finalmente, actuamos en consecuencia, lo que se ve en la conducta, como podría ser desarrollar un malestar cada vez que vemos un lugar con mucha gente y evitar entrar a toda costa.

El problema de que las personas no controlen esta secuencia es que desarrollarán una serie de pensamientos automáticos (no controlados e instantáneos) a partir de ciertos estímulos, que mayoritariamente irán con una carga emocional, y que es posible que la reacción conductual a estos estímulos sea irracional.

## 1.4 Los Trastornos Psicológicos y la Psicopatología

### 1.4.1 Los trastornos psicológicos

Los trastornos psicológicos son una enfermedad mental que puede tener muy distintas manifestaciones hacia el exterior. Normalmente, se caracterizan por una serie de distorsiones cognitivas (que se expondrán posteriormente), y unas alteraciones severas en la conducta y las relaciones con el resto de personas.

Para la cura de trastornos psicológicos, es muy importante el apoyo social que puedan tener estas personas, pero más importante si cabe es que acudan a un profesional cualificado para poder recibir un tratamiento especializado que necesiten. Este tratamiento puede dársele el psicólogo, el psiquiatra o entre ambos profesionales, dependiendo del caso.

En los siguientes apartados, se hará un análisis pormenorizado de los grupos más importantes de trastornos psicológicos que están aceptados en la actualidad. Existen también otros grupos, pero al ser más secundarios se evitará su explicación.

#### **1.4.1.1 Trastornos infantiles**

Los trastornos infantiles se definen como aquellos que se pueden diagnosticar por primera vez en la infancia o adolescencia de la persona, aunque no es una regla fija, sino más bien difusa. Es interesante destacar de estos trastornos que, aunque puede que se den en etapas tempranas de la vida de la persona, es posible que no sean diagnosticados hasta la edad adulta.

Es importante destacar que estos trastornos están encuadrados en esta sección por conveniencia, y no se debe desdeñar el hecho de que el paciente pueda estar también encuadrado en otro trastorno que pueda pertenecer a otro grupo.

Algunos de los subgrupos más comunes en el grupo de trastornos infantiles son:

- Retraso mental
- Trastornos del aprendizaje (lectura, cálculo...)
- Trastornos de las habilidades motoras (coordinación)
- Trastornos de la comunicación (expresión, fonología)
- Trastornos del desarrollo (autismo, Asperger...)
- *Trastornos por déficit de atención (TDAH)*
- Trastornos TICS (Tourette)
- Otros

#### **1.4.1.2 Trastornos de ansiedad**

Los trastornos de ansiedad son aquellos que se pueden dar bajo un contexto de angustia o agorafobia.

Un trastorno por crisis angustiosa es aquel en el que, repentinamente, la persona sufre un ataque de pánico, dándose algunos síntomas como palpitaciones, sudoración o ahogo.

Un trastorno por agorafobia es aquel en el que la persona sufre una aparición de angustia en un lugar donde la escapada es complicada, debido a que no dispone de ayuda o que se encuentra en un lugar público o embarazoso.

Es interesante que, en la combinación o falta de presencia de estos dos grupos se pueden obtener diferentes trastornos y fobias, como la social.

Mención aparte merece el trastorno obsesivo-compulsivo (TOC), ya que es tan común que, como se verá en el apartado siguiente, se usará en este trabajo como un grupo propio.

#### 1.4.1.2.1 Trastorno Obsesivo Compulsivo (TOC)

El trastorno obsesivo compulsivo consiste en un conjunto de pensamientos o impulsos que le aparecen al enfermo de una forma continua, y causan un malestar muy significativo en la persona. Estos pensamientos, llamados obsesiones, no tienen por qué ser ni siquiera de problemas de la vida real, ya que pueden ser irracionales.

En este caso, la persona sabe que estos pensamientos vienen de su mente, y los intenta ignorar de una forma incorrecta.

También, en este caso tenemos las compulsiones, definidas como comportamientos repetitivos que hace una persona para contrarrestar o contestar a una obsesión que posee, siguiendo unas estrictas reglas inventadas por la propia persona. Con esto, la persona busca reducir la obsesión y, por lo tanto, el malestar.

#### 1.4.1.3 Trastornos del estado de ánimo

Los trastornos del estado de ánimo son aquellos que se dan tras un episodio afectivo, que sirve como fundamento al trastorno.

En este grupo, hay dos grupos de trastornos principalmente: Los trastornos bipolares y en el que nos vamos a centrar especialmente a continuación, debido a que es tan común que también se usará como grupo en sí, llamado trastornos depresivos.

##### 1.4.1.3.1 Trastornos Depresivos

Los trastornos depresivos son aquellos en los que la persona ha sufrido uno o varios episodios afectivos depresivos anteriormente, y consisten en una alteración del estado de ánimo muy severa, similar a la tristeza pero en mayores dimensiones.

Algunas de los elementos más importantes que indican la presencia de este trastorno son:

- Tristeza muy severa y continua
- Desmotivación por los objetivos
- Gran irritabilidad
- Puede llegar incluso a ideas suicidas

Un caso muy concreto y a la vez vistoso de estos trastornos depresivos es el trastorno distímico, que de forma simplificada consiste en un trastorno depresivo que solo se da ciertos días y a ciertas horas del día de forma crónica.

#### 1.4.1.4 Trastornos sexuales

Los trastornos sexuales, como su propio nombre indica, son aquellos que psicológicamente conllevan alteraciones en cualquier ámbito relacionado con la sexualidad humana.

Se pueden dividir en diversos grupos, como son:

- Trastornos del deseo sexual (Deseo hipoactivo, trastornos en la excitación, dolores...)

- Parafilias (Fetichismo, pedofilia, masoquismo...)
- Trastornos de la identidad sexual (Transgénero, inadecuación con ningún sexo...)

#### **1.4.1.5 Trastornos de la conducta alimentaria**

Los trastornos de la conducta alimentaria son aquellos que conllevan, por un trastorno psicológico, un cambio en la alimentación de la persona, provocando reacciones también físicas en su cuerpo.

Existen diversos trastornos, pero hay dos que destacan por encima del resto:

- Anorexia nerviosa

La anorexia nerviosa es el rechazo frontal a mantener el peso corporal, intentando bajarlo muy por debajo del umbral que daría el *IMC*. Así, estas personas también tienen una alteración de la visión de su peso y figura, debido al miedo irracional que generan a ganar peso.

Un trastorno muy grave, y a la vez muy actual dentro de la anorexia es la drunkorexia. Consiste en el rechazo a la alimentación para no engordar, pero sin dejar las bebidas alcohólicas.

- Bulimia nerviosa

La bulimia nerviosa consiste en un trastorno en el que la persona comete “atracones”. Un atracón se define como una ingesta de una gran cantidad de comida en un tiempo corto, donde la persona pierde totalmente el control sobre lo que come y cuanto come. Muchas veces estos atracones se dan debido a la búsqueda de refugio de malestar en la comida.

La bulimia conlleva otro problema, consistente en que la persona se da cuenta de que no quiere ganar peso, y por lo tanto se provoca el vómito o abusa de diuréticos o laxantes.

#### **1.4.1.6 Trastornos de la personalidad**

Los trastornos de la personalidad son aquellos en los que la conducta de la persona enferma es muy distinta a la que sería normal en el resto de la sociedad, siendo estas diferencias en la cognición, en la afectividad, en las relaciones interpersonales o en el control de los impulsos.

Así, estas personas suelen tener un gran deterioro social y/o laboral, especialmente porque se suele empezar a dar al final de la adolescencia o primeros años de la edad adulta.

Es interesante saber que no se conoce todavía de donde proceden estos trastornos de la personalidad, pero se sospecha que pueden tener una base genética.

### **1.4.2 En este trabajo: Grupos de trastornos psicológicos**

Para poder clasificar con más sencillez los trastornos psicológicos, me he visto obligado a resumir la gran cantidad que hay en grupos. Tras el estudio del DSM IV (manual de referencia aceptado internacionalmente) y la entrevista con la psicóloga profesional, se ha determinado

que los pacientes se deben de clasificar en cuatro grandes grupos de trastornos psicológicos, que son los siguientes:

- 1) Trastornos TOC
- 2) Trastornos de la Ansiedad
- 3) Trastornos Depresivos
- 4) Trastornos de la Personalidad

Estos grupos de trastornos han sido escogidos debido a la altísima frecuencia con la que se presentan en la consulta privada, además de las diferencias entre ellos, por lo que hace la obtención de datos de pacientes más sencilla.

La única similitud entre los grupos se puede observar entre los trastornos TOC y los trastornos de ansiedad, puesto que un TOC es un tipo muy determinado y específico de trastorno de ansiedad, pero debido a su altísima frecuencia como caso se ha valorado la separación del mismo como grupo aparte.

#### **1.4.3 Distorsiones de la percepción de la realidad: Las distorsiones cognitivas**

Las distorsiones de la percepción de la realidad son pensamientos automáticos distorsionados que se deben a errores en el procesamiento de la información que le llega al individuo. Son la base de toda la psicopatología ya que la unión de ellas da lugar a cambios emocionales que pueden derivar en un trastorno psicológico.

Gracias a, entre otros, David D. Burns (1942 - ) hay una serie de distorsiones cognitivas aceptadas dentro de la terapia cognitivo-conductual, y son las siguientes:

##### **1) Pensamiento Dicotómico:**

El pensamiento dicotómico, también conocido como “Pensamiento Todo o Nada” o “Pensamiento Binario”, consiste en evaluar las cualidades de la propia persona en dos categorías extremas: blanco o negro. Esta distorsión cognitiva constituye la base de lo que denominamos perfeccionismo. Un ejemplo de esto sería la persona que sale de un examen de 50 preguntas, y tiene 3 mal. Si esa persona piensa que el examen ha sido un desastre, estará cayendo en un pensamiento dicotómico, pues el examen no ha sido ni perfecto ni horrible.

Esta visión es falsa, ya que aporta una visión de la vida que no es realista, porque muy escasas veces la vida acaba siendo blanca o negra. Así, una persona que intente situar sus experiencias y emociones en categorías absolutas lo único que va a conseguir es estar de una manera constante en depresión, debido a que las percepciones no se ajustarán a la realidad que esa persona anhela con una exactitud total.

##### **2) Generalización Excesiva**

La generalización excesiva consiste en llegar a la conclusión irracional de que algo que le ha ocurrido una vez, o de una manera escasa, volverá a sucederle de nuevo en el futuro. Normalmente, esta distorsión cognitiva se da en el ámbito negativo, de tal manera que las

personas que la sufren se sienten constantemente abatidas debido a que se piensan que la situación desagradable que han vivido inevitablemente la volverán a vivir.

Un ejemplo de generalización excesiva es el aseverar que nunca se tendrá pareja e hijos debido al rechazo de una persona. Esto es un error por dos motivos:

- a. Todas las personas no tienen el mismo gusto
- b. Por el simple hecho de que una persona te haya rechazado, no tienen que rechazarte el resto

### 3) Filtro Mental

El filtro mental, también conocido como “abstracción selectiva”, consiste en estar en una situación o haber pasado la misma, y al analizarla centrarse en sólo un elemento de esa situación, haciendo caso omiso al resto. Normalmente, se desarrolla un filtro mental negativo, por lo que las personas que padecen esta distorsión ven toda la situación rodeada de negatividad.

Esto suele pasar cuando una persona está deprimida. La persona que la sufre “se pone unas gafas” que le sirven de filtro para que nada sea positivo, y todo lo que llega a la mente son pensamientos negativos.

### 4) Descalificación de lo positivo

Esta distorsión es una maximización de la anterior, y consiste en transformar situaciones que pueden ser neutras, o incluso positivas, en situaciones negativas. Esto se hace ignorando la parte positiva de la situación, y dándole la vuelta con algún cierto argumento irracional para convertirlo en algo negativo.

Un ejemplo de esto sería cuando un compañero de trabajo te felicita por tus últimos logros, y tú sólo piensas en que lo que quieren es quedar bien, no te quieren felicitar de verdad. En ese momento, le has dado la vuelta a la situación y ya no te estás centrando en tus logros y trabajo bien hecho, sino que te estás centrando en tu creencia de que la gente que te rodea no se alegra por ti de una forma real.

Esta distorsión cognitiva es bastante común, y por desgracia es una de las distorsiones cognitivas más importantes a la hora de diagnosticar un cuadro depresivo.

### 5) Conclusiones Arbitrarias

Las conclusiones arbitrarias, también conocidas como “apresuradas”, son una distorsión cognitiva mediante la cual la persona toma decisiones y saca conclusiones de una determinada situación, normalmente negativas, de una manera no justificada.

Esta distorsión cognitiva se puede dividir en dos tipos de casos:

- a. Lectura del pensamiento

La lectura del pensamiento consiste en estar convencido de una afirmación que es negativa para la propia persona, sin tener hechos fehacientes que lo demuestren, y no molestarse por comprobarlo, dándolo por hecho.

Un ejemplo de esto sería una chica que sale todos los fines de semana con sus amigas. Un fin de semana, las otras dos o tres chicas no pueden quedar por diversos motivos de fuerza mayor, y se lo hacen saber a la primera. Si esta chica empezara a pensar que no quieren quedar con ella “porque es una persona aburrida”, estaría cayendo en una lectura del pensamiento.

b. Error del Adivino

El error del adivino consiste en suponer, sin pruebas para ello, que va a ocurrir algo malo, y que siempre va a ser malo. Además, no solo lo supone, sino que lo toma como un hecho, algo asegurado.

Un ejemplo de esto se puede dar con las personas mayores que sufren alguna enfermedad. Siempre piensan que se van a morir con ello, y luego al tomarse alguna medicina se sienten mejor y ven que su pensamiento era erróneo. Otro ejemplo, muy común entre los estudiantes, es el pensamiento de los días antes del examen de “a ver si suspendo”, “a ver si me quedo en blanco”...

6) Maximización y Minimización

También conocido como magnificación y minimización, esta distorsión cognitiva se basa en el hecho de aumentar o disminuir las situaciones de una manera totalmente desproporcionada a la importancia que tienen, dando lugar a pensamientos catastróficos sobre pequeños errores y dando poca importancia a elementos positivos.

En la siguiente imagen, con la ayuda de Homer Simpson se puede ver un ejemplo de esta distorsión:



1-2. Ejemplo Maximización

7) Razonamiento Emocional

El razonamiento emocional es otra distorsión cognitiva basada en la toma de las emociones propias como prueba irrefutable de verdad. Este razonamiento no es correcto, debido a que los sentimientos únicamente reflejan pensamientos, y estos son subjetivos.

Un ejemplo de razonamiento emocional sería: “Me siento como un fracasado, por lo tanto jamás aprobaré la carrera y nunca conseguiré un trabajo”.

Es importante destacar que, obviamente, el razonamiento emocional es una de las distorsiones cognitivas fundamentales para el diagnóstico de trastornos depresivos.

#### 8) Los debería

“Los debería” es una distorsión cognitiva ciertamente pintoresca, pues parten del hecho de animarse a sí mismo y motivarse diciendo: “Debería hacer esto”. Pero esto es un arma de doble filo, ya que estas frases ejercen una presión sobre nosotros mismos.

Además, los debería no solo se suelen dirigir a la persona propia, sino que muchas veces se suelen dirigir hacia otras personas, criticando elementos que suponemos que deberían de hacer. Pero la generación de críticas con los debería hacia los demás lo único que hace es generar un cierto resentimiento en la persona propia.

De esta manera, los debería son “fácilmente” cambiables. Lo único que deberá de hacer la persona es cambiar sus expectativas de la realidad, antes irreales, hacia algunas más reales, porque de lo contrario la persona se convertirá en amargada y cínica.

#### 9) Etiquetación

La etiquetación consiste en la creación de una imagen de sí mismo errónea y negativa, basada únicamente en los errores que se han cometido. Es el extremo del punto 2, la generalización excesiva.

Burns (1980) afirma: “La filosofía en la que se basa [la etiquetación] consiste en que la medida de un hombre la dan los errores que comete”. Por ejemplo, tras aprobar todos los exámenes de la carrera, uno suspende uno en el cuarto curso, ya cerca de sacarse el título. Una persona que no use la etiquetación podría decir: “He estudiado mal, es un problema, pero lo sacaré”. En cambio, una persona que padezca la distorsión cognitiva de la etiquetación pensará algo como: “Soy un perdedor, no soy capaz ni de sacarme la carrera”.

Así, a estas personas hay que hacerles ver que la vida no es sólo lo que hace uno, y que poner etiquetas consiste en detallar un hecho con palabras que conllevan una fortísima carga emocional y por lo tanto no son objetivas.

#### 10) Personalización

La personalización es la distorsión máxima por la cual nos culpamos. Consiste en asumir la responsabilidad de algo malo que haya pasado, aun no teniendo relación ninguna con ese hecho.

El problema de la personalización, como he comentado antes, es que se siente en sí mismo una gran culpa, y se siente como si muchos elementos no positivos que ocurren a su alrededor solo dependieran de sí mismo.

Un ejemplo de esto es el profesor al que los alumnos no le hacen los deberes. Este profesor, tras razonar, puede pensar que la culpa es de los alumnos porque no cumplen con su deber, o puede pensar que la culpa es suya, y que por lo tanto es un malísimo profesor. En este segundo caso, el profesor estaría cayendo en una personalización.



#### 1.4.4 En este trabajo: Distorsiones cognitivas usadas

Debido a la naturaleza científica del trabajo en busca de predicciones y clasificaciones mediante técnicas de inteligencia artificial, se debe de hacer una selección de qué distorsiones serán las que sean usadas para poder hacer predicciones de los grupos de trastornos.

Debido a lo aprendido, y tras consultar con una psicóloga profesional con despacho privado, se usarán todas las distorsiones cognitivas a excepción de la de la descalificación de lo positivo, debido a que prácticamente todas las descalificaciones de lo positivo se centran en un filtro mental previo, y habiendo este filtro no es necesario usar esta distorsión cognitiva en la recopilación de datos y en la predicción.

Esto es una ventaja, debido a que me será ahorrada una variable tanto en la toma de datos como en el procesamiento, lo cual para el *dataset* no será demasiada diferencia a la hora de computarlo, pero sí lo sería en un dataset con muchos más pacientes. En esos casos, la toma de estas decisiones resulta de un papel fundamental a la hora de hacer cálculos para ser más eficientes computacionalmente.

##### 1.4.4.1 En este trabajo: Otras variables usadas

A continuación se expone una lista del resto de variables que serán usadas en este trabajo. Estas variables han sido recopiladas de la entrevista que se tuvo con una psicóloga clínica profesional con despacho privado y más de 25 años de experiencia, y junto con estas variables se expondrá seguidamente su importancia.

###### 1) Nombre

El nombre, dado sin apellidos ni otros datos personales debido a la LOPD, no se utilizará en predicciones ya que carece de relación con los trastornos. Únicamente se utilizará como identificativo de los pacientes y para el análisis exploratorio del dataset.

###### 2) Edad

La edad es un factor que, aunque no es determinante en el análisis psicopatológico de la persona, puede tener que ver, ya que algunos trastornos como la anorexia nerviosa se dan con más frecuencia en un rango de edad determinado.

###### 3) Sexo

El sexo es un factor importante, debido a que las personas de género femenino normalmente suelen ser más propensas a los trastornos psicológicos que las masculinas, y por ende acuden más a consulta. Además, los cuadros no son iguales en un sexo o en otro. Si por ejemplo analizamos en los trastornos de la alimentación, la anorexia es un trastorno que lo tiene un ratio de mujeres muy superior al de los hombres, pero en cambio en la vigorexia ocurre todo lo contrario.

#### 4) Relación con el Contexto

La relación con el contexto es una de las variables más importantes que se recopilan. Esta se refiere a la relación que tiene el paciente con las personas más cercanas a su entorno, que normalmente suelen ser los padres o la pareja.

Debido a la complejidad de esta variable, y al número de opciones que se pueden dar, se ha tomado la decisión de dividir esta variable en tres, obteniendo lo siguiente:

##### a. Relación con el contexto mala:

En el caso de una relación mala con el contexto, la persona se lleva mal con sus personas cercanas debido a una discusión o un enfrentamiento similar.

##### b. Relación con el contexto de trauma:

En este caso, la persona no se lleva bien con el contexto debido a algún trauma que haya sufrido, como puede ser la pérdida de algún familiar cercano, abusos sexuales o situaciones similares.

##### c. Relación con el contexto buena:

En este caso, la persona tiene una buena relación con sus personas cercanas.

#### 5) Habilidades sociales

Con las habilidades sociales, hacemos referencia a la forma que tiene la persona de relacionarse con el exterior. Debido a esto, hay reconocidas tres grandes respuestas:

##### a. Inhibición

Cuando una persona es inhibida, no suele expresar lo que piensa o siente, y si lo hace suele hacerlo en momentos que no son los más adecuados para ello, o torpemente.

##### b. Asertividad

Cuando una persona es asertiva (también conocida como hábil), expresa lo que piensa de forma directa y adecuada, respetando tanto sus derechos como los de los demás.

##### c. Agresividad

Cuando una persona es agresiva, da a conocer lo que piensa y siente normalmente en segunda persona, y lo hace de una manera alterada y alienando la conversación con el resto de personas.

#### 6) Impulsividad

Conocemos impulsividad como la tendencia que tiene una persona a realizar unos actos sin premeditación, y por ello sin tener en cuenta las consecuencias.

Las personas impulsivas son más propensas a la agresividad (se verá posteriormente en la práctica), y a veces pueden ser personas con problemas debido al consumo de sustancias como pueden ser el alcohol o las drogas, especialmente si se trata de un trastorno de la personalidad.

Las personas que tienen impulsividad suelen ser más propensas a la pérdida del autocontrol, y por ende, trastornos como la hiperactividad o cuadros de ansiedad son comunes en estas personas.

## 2. Capítulo 2: Data Science

---

### 2.1 Data Mining Vs Machine Learning Vs Data Science Vs Big Data

Cuando se habla de temas como Big Data o data science, hay varios conceptos que se pueden venir a la cabeza, pero dos de ellos sin duda son el data mining (o, en español, minería de datos) y machine learning (aprendizaje automático).

Las diferencias entre estos términos pasan por alto a la mayoría de las personas, pero a continuación se hará hincapié en ellas para tener los conceptos bien separados, y a partir de ahora saber exactamente a qué nos referimos:

#### 2.1.1 Data Mining

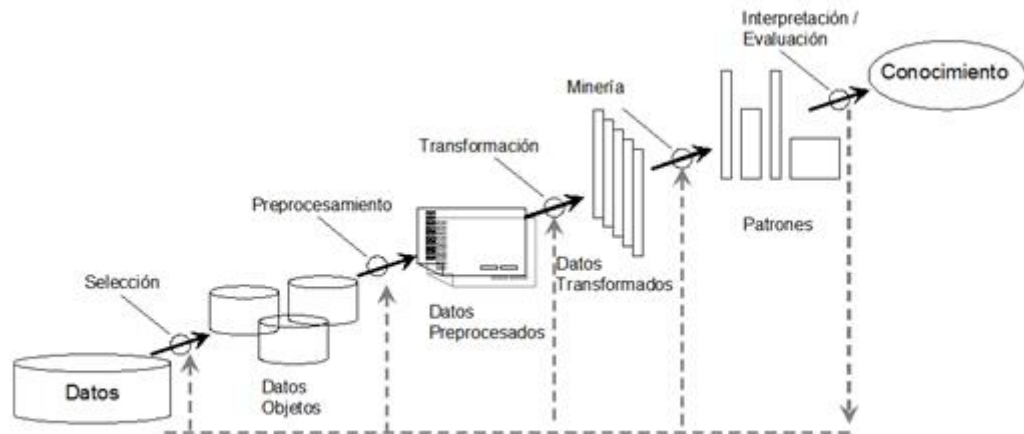
Los orígenes del data mining vienen de la dificultad de poder manejar diferentes tipos de datos con las herramientas existentes. De este trabajo se acabó derivando en ideas que se tomaron prestadas de otros campos, como la estimación o el muestreo tomados de la estadística, o los algoritmos y las técnicas de aprendizaje provenientes de la inteligencia artificial. También otras áreas tienen un papel esencial en todo lo que rodea al data mining, como es el área de visualización, de bases de datos o de computación.

Así, con data mining, o minería de datos, hacemos referencia al proceso de descubrir información que pueda ser útil, a través del análisis de grandes repositorios de datos. De este modo, con la minería de datos se intenta encontrar patrones y soluciones de preguntas que, de otro modo, estarían ocultas entre los datos.

Es importante distinguir entre data mining y la recogida de información. Mientras que data mining usa técnicas estadísticas y matemáticas para la obtención de información dentro de un dataset, la recogida de información consistiría en, por ejemplo, una búsqueda en una base de datos para un sujeto concreto. A pesar de centrarse los dos en los datos, son elementos y técnicas distintas y, por lo tanto, deberán de mantenerse por separado.

El descubrimiento de conocimiento es la última meta de la minería de datos. Conocido en la comunidad anglosajona como *KDD (Knowledge Discovery in Databases)*, el descubrimiento de

conocimiento podríamos decir que es el proceso total de convertir los datos puros de la base de datos en una información útil. Es decir, el descubrimiento de conocimiento es el concepto por el que, mediante el uso de data mining, obtenemos información útil de una gran cantidad de datos que, a priori, no nos da ninguna información a simple vista.



2-1 Proceso KDD detallado

Este descubrimiento de información consiste en una serie de pasos, que van desde un pre-procesamiento de los datos para su preparación, hasta un post-procesamiento para su posterior obtención de información. Observemos este proceso con más detenimiento:

#### 1) Pre-procesamiento de los datos:

El pre-procesamiento de los datos es un paso esencial en data mining, debido a que los datos pueden estar guardados en una gran cantidad de formatos y formas, o incluso estar distribuidos en diferentes repositorios.

Una vez importado el dataset, o el conjunto de datasets con los que se va a trabajar, se debe de hacer este pre-procesamiento de los datos para prepararlos de cara al data mining. De esta manera, acciones como la unión de tablas, la reducción de la cantidad de variables (también conocido como reducción de la dimensionalidad), o la obtención de subgrupos de datos, serán pasos muy importantes de cara a preparar los datos para los próximos pasos.

Normalmente, este pre-procesamiento suele ser la parte que más tiempo consume en el proceso de la minería de datos, debido a que es muy manual y laboriosa.

#### 2) Data Mining:

En este paso, usaremos las numerosas técnicas estadísticas y matemáticas que conforman el data mining, como pueden ser la unión por grupos, el estudio de la variabilidad, el estudio de las relaciones entre las observaciones o el estudio de la frecuencia entre muchas otras. Este conjunto de tareas recibe el nombre de "tareas descriptivas", ya que el objetivo de las mismas es obtener patrones que resuman las relaciones que haya por debajo en los datos.

Además, junto con las tareas descriptivas podremos hacer un análisis predictivo para poder predecir ciertas características de futuras observaciones. Esto se analizará más adelante.

### 3) Post-Procesamiento de los datos:

Con el post-procesamiento de los datos, se da referencia esencialmente a la quizás necesaria transformación final de los datos de cara a la siempre necesaria visualización. Fuera de data mining, junto con esta visualización, siempre se debe hacer un análisis e interpretación de los datos obtenidos, de cara a la aclaración de los mismos y la finalización del proceso, obteniendo información útil.

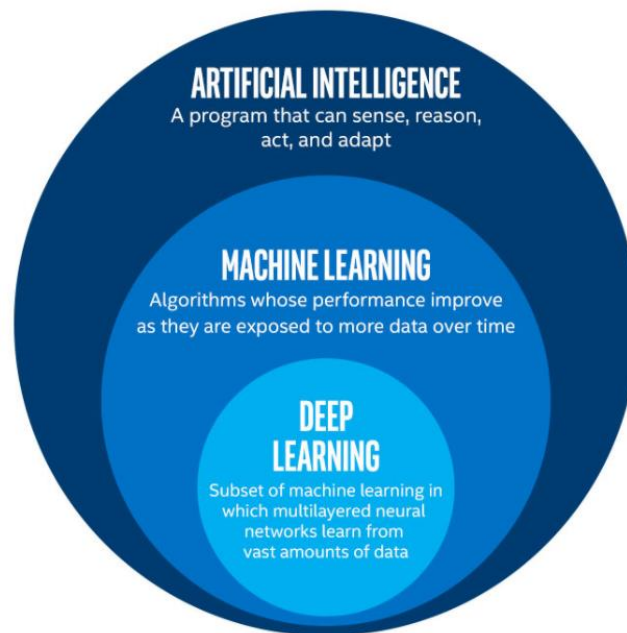
También, otro post-procesamiento muy usado es la unión de los resultados obtenidos de data mining a otras herramientas, como pueden ser las de marketing, de tal manera que estos datos se puedan usar en otros ámbitos. Este proceso es conocido como “closing the loop”, que se puede traducir por “cierre del círculo”.

#### 2.1.2 Machine Learning

Si se habla de machine learning, o como también se conoce en países de habla hispana, aprendizaje automático, se hace referencia a un subconjunto de la inteligencia artificial que, además de usar los principios de data mining, es capaz de hacer correlaciones de una manera automática, y también es capaz de aprender de los datos que se tienen y se tendrán, de tal manera que el modelo puede seguir mejorando con el paso de mayor cantidad de datos.

El uso que se hace del machine learning por cualquier usuario es diario, ya que se utiliza en numerosos ámbitos, tales como la publicidad (especialmente en internet), algoritmos de búsqueda, algoritmos de predicción del tiempo atmosférico, optimización de la duración de baterías, sistemas de recomendación...

Se podría decir que, aunque son muy parecidas, data mining se centra un poco más en las relaciones de los datos que hay ahora mismo, y la obtención de información de ello, mientras que el machine learning usa estos principios de data mining para también obtener predicciones y clasificaciones, y aprender de los datos para mejorar estas técnicas. De este modo, el aprendizaje automático puede observar patrones y aprender un comportamiento, mientras que el data mining es el recurso en el cual se basa el machine learning y mediante el que se puede ir obteniendo cierta información útil.



## 2-2. Subdivisiones de la Inteligencia Artificial

Así, las tareas más relacionadas con machine learning se denominan tareas predictivas, ya que se basan en predecir un atributo particular a través de los valores del resto de atributos. A partir de ahora, una de las formas con las que se podrá referirse a estas tareas será como “modelos predictivos”.

Respecto a la posición relativa del machine learning respecto al data mining a la hora de hacer data science, hay varias versiones de donde se debe colocar. Es verdad que, mediante las técnicas de machine learning, se está obteniendo información a partir de los datos, y además se usan muchos conceptos básicos de data mining en esta parte, por lo que podría incluirse como un subapartado de data mining, que se realiza después del pre-procesamiento y análisis exploratorio de los datos.

Otras teorías afirman que el machine learning deberá de ir como un apartado diferente de data mining y siempre después de este, debido a la naturaleza predictiva en vez de descriptiva.

Debido a estos dos enfoques, y a la vista de tomar una decisión, en este trabajo se analizará el machine learning como un apartado dentro de data mining, pero no sin antes volver a remarcar la gran conexión que tienen ambos conjuntos, con la única diferencia de una naturaleza diferente.

### 2.1.3 Data Science

De esta manera, si ya se poseen unos algoritmos que permiten obtener información, y otros algoritmos que van mejorando con el paso de los datos y consiguen la predicción de la información, ¿qué cabida tiene data science?

Con data science se hace referencia al elemento que cobija a data mining y a machine learning. Data science no es más que un término genérico que aúna un conjunto de técnicas o subdisciplinas, como data mining, machine learning y visualización de datos, entre otras, para

obtener un conjunto de “insights” o conclusiones que sean útiles al usuario final, como puede ser una empresa (*Business Intelligence*) o cualquier otro usuario interesado.

De una manera más espectral, se podría afirmar que data science consiste en la mezcla de una serie de procedimientos matemáticos que, junto con conocimientos del problema tratado y de tecnología especializada, consiguen obtener conclusiones efectivas y fácilmente entendibles para el usuario final.

#### 2.1.4 Big Data

Con Big Data se hace referencia a un término muy de moda en los últimos tiempos. Se ha mostrado anteriormente que data science recoge todo el conjunto de técnicas desde la importación de la información hasta la obtención de los resultados finales con su información útil y entendible por cualquiera.

De este modo, “big data” hace referencia simplemente a la disciplina que trabaja con unas grandes cantidades de datos. Es una disciplina que, al igual que el “medium data” y el “small data”, están presentes en proyectos de data science.

##### 1) Small Data

Con small data, se hace referencia a proyectos en los cuales los datos están en un formato CSV/TSV pequeño, una base de datos pequeña o incluso en un Microsoft Excel. En estos proyectos, se puede trabajar con un ordenador estándar, y los datos se pueden cargar perfectamente en memoria RAM.

En el caso de este proyecto, se usará un estilo de small data debido a la pequeña cantidad de observaciones y variables que se poseen.

##### 2) Medium Data

Los proyectos que usan medium data son aquellos que usan una cantidad de datos más grande que los de small data, y, aunque los datos se pueden albergar normalmente en un ordenador, las técnicas de extracción de los mismos son distintas, ya que no se puede pedir toda la información de golpe a riesgo de bloquear el ordenador sobrecargando la memoria.

Para el tratamiento de los problemas de medium data se suele utilizar Apache Spark sobre un cluster, puesto que este software permite evitar esta sobrecarga de memoria y de esta manera poder brindar y tratar los datos según se vayan requiriendo.

##### 3) Big Data

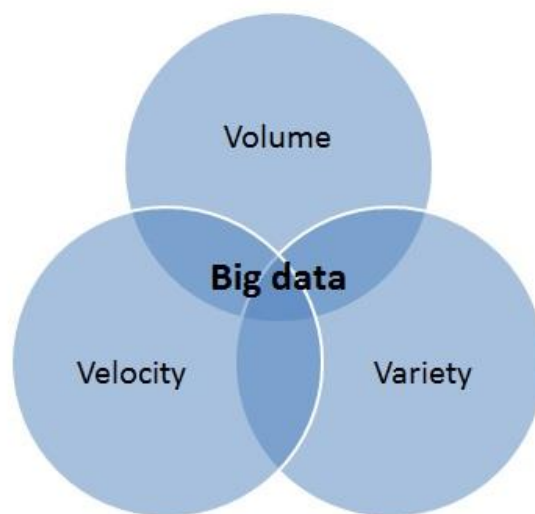
Finalmente, los proyectos de big data son aquellos en los que se necesitan varios ordenadores en cluster, o un servidor de grandes dimensiones, para poder almacenar y procesar toda la información. Estos proyectos son mucho más complejos, ya que suelen necesitar técnicas de sincronización entre ordenadores, cálculo en paralelo o en grid y técnicas similares.

En este tipo de problemas, soluciones combinadas como el uso de Apache Hadoop (junto con técnicas como MapReduce) y Apache Spark (para evitar los overflow de RAM) son

imprescindibles para obtener las conclusiones en un tiempo razonable, además de hacer una gestión eficiente y segura de los datos, resistente a fallos (replicación) y a caídas.

De este modo, se puede afirmar que “Small Data”, “Medium Data” o “Big Data” son un contexto, un “*framework*” donde se mueven los proyectos de data science, y que dependiendo de cual sea necesario se necesitará una tecnología u otra para el tratamiento de los datos.

También, otro método más informal pero muy efectivo para saber bajo que paradigma vamos a trabajar es el de tener en cuenta las 3 V's del Big Data: Velocity, Volume, Variety. Antes de entrar en detalle con ellas, es importante aclarar que hay más V's, donde algunos expertos afirman que existen hasta 12. En este trabajo se repasarán las 3 centrales, las que más aceptadas están por todos estos expertos en la materia.



2-3. Las 3 V's del Big Data

- Velocity

Velocity se refiere a la velocidad de la creación de nuevos datos. Si se crea una gran cantidad de datos en breve lapso de tiempo, podremos decir que podríamos estar en un problema de Big Data y por lo tanto una base de datos no convencional (como una NoSQL) sería una opción interesante.

- Volume

Quizás el más obvio, el volumen de datos con los que se va a trabajar influye enormemente en la manera de encarar el problema. No hay reglas escritas, pero un problema de Big Data se supone que no se puede resolver en un ordenador de casa, ni se pueden almacenar los datos en el mismo. De tal manera, los problemas que se encuadrarán dentro de Big Data serán problemas de cientos de terabytes, petabytes o incluso mayores.

- Variety

La variedad de los datos, debido a la gran cantidad de estructuras de datos y fuentes, es otro de los indicadores principales de que existe un problema resoluble con Big Data. No es el



requerimiento más importante a la hora de determinar bajo que paradigma se resolverá el problema (los dos anteriores se antojan vitales), pero sí es necesario tener en cuenta que en Big Data los datos de diferentes fuentes y diferentes formatos son muy comunes.

## 2.2 Antes de hacer Data Mining

Un paso fundamental a la hora de afrontar un problema con datos es obtener estos datos. Los datos pueden ser obtenidos de numerosas fuentes, y más cuando se puede afrontar un problema que sea bajo big data o medium data, que, como se ha visto anteriormente, pueden estar distribuidos entre numerosos ordenadores. Además, las técnicas de petición de los datos son más complejas que una simple carga en memoria RAM.

Así, el conocimiento de tecnologías como Apache Spark, para el control del flujo de datos a memoria, se antoja esencial en aquellos proyectos que no pertenezcan al grupo de small data.

### 2.2.1 En este trabajo: Obtención de los datos

En mi caso, se podría considerar el problema de “small data”, y el dataset se ha obtenido a partir de la entrevista con una psicóloga profesional, donde me han sido dados datos de pacientes reales de su consulta privada. Para mantener la confidencialidad, de dichas personas solo he recibido los datos clínicos, nombre, edad y sexo. Los apellidos, la ciudad e imágenes, además de irrelevantes para el trabajo, no han sido proporcionados por temas de privacidad.

Posteriormente, se ha confeccionado un dataset a mano, con formato CSV (Comma Separated Values), para poder importarlo posteriormente a RStudio y Jupyter Lab. Esta entrevista y posterior confección del dataset conllevaron aproximadamente unas 5 horas de trabajo continuo. Se optó por la confección a mano de un dataset real, y no la obtención en internet de uno, debido a que en internet no existe ningún dataset con esta información que es necesaria para estas predicciones, a fecha de enero de 2019.

La elección del formato CSV fue debido a varios motivos, como son la facilidad de confección de dicho dataset, la amplia variedad de datasets con el mismo formato para problemas de Big Data e Inteligencia Artificial, y la facilidad de importación posterior para los programas que serán utilizados.

También, se considera importante comentar que la variable más importante se encuentra al final de cada fila del dataset, y corresponde al grupo al que pertenece el paciente. Estos grupos son una variable discreta que versa del 1 al 4, y se corresponde de la forma siguiente:

1. Trastorno Obsesivo – Compulsivo
2. Trastorno de Ansiedad
3. Trastorno de Depresión
4. Trastorno de Personalidad

Esta transformación a variable numérica discreta se ha hecho en la recolección para no tener que hacer posteriores transformaciones mediante código, además de que un número es más rápido de anotar que el nombre completo de un trastorno.

## 2.3 Data Mining

Como se ha visto, con data mining damos referencia a todo el proceso de conseguir una información útil y entendible a partir de un conjunto de datos. Así, este apartado estará dividido en la explicación de los diferentes procesos que se han llevado a cabo para conseguir hacer data mining sobre el problema, y la explicación de cómo han sido implementados.

### 2.3.1 Pasos previos y preparación de los datos

Según IBM, empresa líder mundial en ventas de máquinas para negocios, a la hora de preparar los datos antes de un proceso de minería de los mismos es importante seguir tres pasos:

#### 1) Entendimiento del negocio

Respecto a entendimiento del negocio, se entiende la comprensión de cuál es el objetivo que se busca, así como cuál es el indicador que dirá que hemos tenido éxito. Obtener información puede estar muy bien sin entender nada, pero con un background previo en el área de aplicación de esos datos se podrá saber si se va por el buen camino o hay que pivotar.

#### 2) Entendimiento de los datos

Con el entendimiento de los datos, y parte más informática de los dos, se entiende la selección de los datos que se consideren más relevantes, y la completa comprensión de estos datos. Este paso se hace ciertamente “a ciegas”, por lo que el primer paso de comprensión del negocio se antoja vital. Después, mediante pre-procesamiento de los datos se podrá obtener matemáticamente cuales son las variables más importantes.

Es importante esta comprensión de los datos debido a que muchas veces los mismos, especialmente en problemas de medium data y big data, pueden presentarse en diferentes tablas. La comprensión de cada dimensión en profundidad para conocer lo que aporta hacia el problema, saber si los datasets tienen sentido juntos, y conocer si el cambio de algún dato o de alguna dimensión es necesario, se antoja necesario puesto que empezar a operar “a ciegas” no es la forma óptima para obtener buenos resultados en la etapa de machine learning. Así, es necesaria una comprensión en profundidad del problema para la identificación de los datos más relevantes.

También es importante el entendimiento de los datos y del problema debido a que los datos nos pueden llegar de una manera desestructurada, como ocurre por ejemplo con los datos de las redes sociales. Estos datos tienen una estructura interna que no se puede apreciar a simple vista, puesto que no siguen un formato específico, lo cual implica un esfuerzo y un entendimiento extra para poder operar satisfactoriamente con los mismos.

#### 3) Preparación de los datos en sí

Una vez que se poseen una serie de datos con los que se va a trabajar, seleccionados y entendidos, y una serie de objetivos en mente, es el momento de codificar. Para ello, lo primero que se tendrá que hacer es una segunda preparación de los datos. Con este objetivo, habrá que comprobar los tipos de datos, y la calidad de los datos:

a. Tipos de datos

La comprobación de los tipos de datos se antoja como una parte fundamental de la preparación de los datos, debido a que un dataset no es más que un conjunto de objetos de datos. A la hora de hacer clasificaciones y predicciones, la diferencia entre un carácter, una variable continua o una discreta puede marcar la diferencia de la calidad de la predicción, o incluso si el algoritmo predictor funciona o no.

b. Comprobación de que los datos no están comprometidos

La comprobación de la seguridad de los datos, especialmente si se trata de datos sensibles de personas, es fundamental tanto para la seguridad y el honor de las mismas como para la legalidad de la empresa. De esta manera, los servidores u ordenadores que posean los datos deberán de tener una serie de medidas de seguridad, algunas de las cuales vienen recomendadas en la Guía del Reglamento General de Protección de Datos para Responsables del Tratamiento, expedida por la Agencia Española de Protección de Datos (AEPD).

Por otra parte, si los algoritmos de machine learning producen cualquier resultado que pueda ser también confidencial (como, por ejemplo, la predicción de la ideología política de una persona), estos nuevos datos generados también se deberán de salvaguardar al igual que los anteriores.

c. Calidad de los datos:

En muchas ocasiones se hace minería de datos sobre datos que no han sido recogidos específicamente para ese momento o esa intención. Debido a esto, se debe de hacer una pequeña valoración de calidad al principio, en la obtención. La evitación de problemas de calidad de los datos es un elemento indispensable a la hora de hacer data mining. Así, se llega hasta la primera parada técnica de un data scientist: La limpieza de datos. Con una correcta limpieza de los datos se obtiene un dataset que, aunque puede estar más incompleto en algunas ocasiones, posee todos sus datos de una manera igual, óptimamente usable y entendible por los diferentes procesos por los que pasarán después estos datos.

Posteriormente, para continuar con la preparación de los datos, se debe de valorar la opción de un pre-procesamiento y transformación de los mismos, atendiendo a algunas técnicas como:

i. Agregación

La agregación consiste en la técnica de unir dos o más objetos en uno solo, y se basa en la filosofía de que a veces “menos es más”.

Con la agregación, siempre y cuando tenga sentido hacerla, se puede reducir el número de filas o columnas, de tal manera que el dataset puede ser computacionalmente más eficiente y menos complejo, con lo que el tiempo de computación se reduce. Si se toma la visión contraria, con un dataset más pequeño se pueden usar algoritmos más complejos en un tiempo razonable de computación.

Uno de los problemas más obvios que tiene la agregación es la pérdida de información que conlleva una unión de datos, por lo que la valoración de si se hace una agregación o no es esencial de cara a los resultados que se pueden obtener posteriormente, teniendo en cuenta la precisión necesaria en los resultados.

## ii. Muestreo

El muestreo es una técnica que se utiliza para coger sólo una cantidad fraccional de los datos totales que se tienen, y analizar ese nuevo conjunto. En ámbitos como la estadística que, como ya se ha visto, está muy relacionada con todo este mundo, esta técnica se lleva usando durante muchísimo tiempo para hacer un análisis preliminar de los datos.

El uso de muestras se suele usar cuando se trabaja bajo big data o medium data, debido a que el análisis de estas cantidades de datos es demasiado costoso y, si la muestra es fielmente representativa, se obtendrán unos resultados casi tan buenos como con los del dataset completo.

El hecho de que una muestra sea representativa depende de si tiene un valor similar en una propiedad al valor de la misma propiedad en el conjunto de datos total. Es decir, si por ejemplo se toma como medida de representatividad la media, si el valor de la media del conjunto de muestra es similar al valor de la media del conjunto total se podrá afirmar que el conjunto es representativo. Pero obtener un conjunto representativo no es tan sencillo como coger unas muestras y usarlas. Hay diferentes aproximaciones que se pueden hacer, como son el muestreo aleatorio (cualquier elemento tiene la misma probabilidad de ser cogido que el resto) y el muestreo estratificado, que comienza con un conjunto de grupos ya preestablecidos.

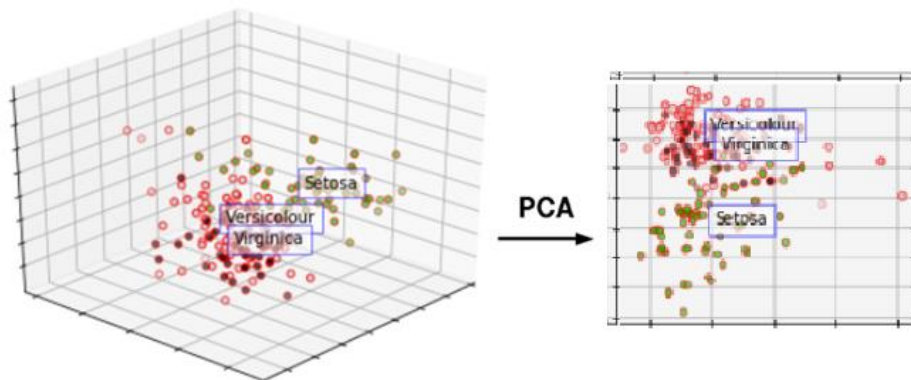
## iii. Reducción de la dimensionalidad

La reducción de la dimensionalidad es una técnica consistente en la eliminación de columnas del dataset (y por lo tanto, dimensiones), de tal manera que mejore la eficacia de los algoritmos de data mining. En parte, esto se debe a que en la reducción de dimensiones, algunas características innecesarias son eliminadas, además del posible ruido. Por otra parte, esta mejora viene dada por la llamada “maldición de la dimensionalidad”.

La maldición de la dimensionalidad consiste en el fenómeno basado en que la minería de datos aumenta en complejidad conforme el número de dimensiones aumenta. Esto, si se observa en el espacio, significa que cada vez el conjunto de los datos se vuelve más difuso, lo que complica la clasificación de los datos, y esto conlleva modelos más imprecisos y complicados de clasificar.

De este modo, la técnica más utilizada para el análisis de las dimensiones es conocida como PCA, acrónimo de Principal Components Analysis, que en castellano significa “Análisis de Componentes Principales”. PCA consiste en una técnica de álgebra lineal, aplicable a variables continuas, que encuentra nuevas variables a partir de la combinación lineal de variables anteriores. Haciendo uso de esta técnica, si las variables son ortogonales, se obtendrá mucha información, mientras que si las variables tienen vectores muy similares en tamaño y dirección, se explicarán mucho la una a la otra y de este modo una de las dos será “innecesaria” en una gran medida.

En la siguiente figura podemos ver de forma gráfica una reducción de la dimensionalidad, pasando de tres a dos dimensiones:



2-4. Reducción Dimensionalidad Dataset Iris

PCA posee numerosas características muy atractivas, como la posibilidad de encontrar eficientemente patrones en los datos, la posibilidad de reducir drásticamente la dimensión en una gran cantidad de problemas o la posibilidad de eliminar gran parte del ruido. Debido a todos estos beneficios, imprescindibles en data science, PCA posee una gran fuerza a la hora de hacer análisis de datos, y es una técnica ampliamente usada.

Es importante destacar que la diferencia entre las dimensiones las calcula mediante variabilidad, de tal manera que supone que el dataset entero tiene una variabilidad del 100%. Cuando dos vectores son muy similares, una de esas dos dimensiones no aporta prácticamente nada de variabilidad que pueda ayudar a explicar el problema, mientras que dos vectores muy distintos en tamaño y dirección (como los ortogonales) aportarán una gran variabilidad, y por lo tanto explicación del problema de cara a los futuros algoritmos. De esta manera, con un porcentaje de explicación alto, pero no necesariamente del 100%, los algoritmos de machine learning podrán trabajar de una manera muy aproximada reduciendo en un gran porcentaje la dimensionalidad, de tal manera que será computacionalmente más eficiente a costa de pequeños errores de clasificación o predicción.

Respecto al punto en el que se debe de parar de reducir, hay que saber asignarlo a cada problema, puesto que este tipo de técnicas variarán respecto del dataset. Por regla general, si el problema no necesita ser extremadamente preciso, con una explicación del 80% - 85% suele ser suficiente. En otros problemas, como de bioinformática, la explicación deberá de ser mucho mayor.

Para mirar esta explicación, es interesante observar unos valores conocidos como "eigenvalues". Estos valores explican, para una cierta cantidad de dimensiones que tendría el problema, la explicación total que poseería, normalmente expresado en tanto por ciento o tanto por uno. También puede venir expresado de una manera distinta, donde se expresa como por cada dimensión que se añada la explicación que se añadiría al problema. Sea cual sea la representación que se obtenga, habrá siempre que mirar en qué momento la explicación baja de una manera más acusada con el aumento de las dimensiones, y ese punto, en la mayoría de los casos, será el óptimo para obtener el dataset con el que se trabajará posteriormente.

#### iv. Creación de características

Como oposición a la reducción de la dimensionalidad, se puede poner en acción otra técnica llamada creación de características, o como se conoce en inglés, “Feature Creation”.

Esta técnica destaca por la creación de dimensiones a partir de las dimensiones ya existentes, de tal manera que se crea un nuevo dataset con unas dimensiones que capturan la información de una manera mucho más efectiva. Además, tiene como ventaja que se produce una reducción de la dimensionalidad, con todos los beneficios que hemos visto anteriormente. Por ello, en datasets con una muy alta dimensionalidad esta técnica es muy valiosa.

Existen tres métodos para la creación de características:

##### 1. Extracción de características original

Básicamente, la extracción de características consiste en la creación de un nuevo conjunto de características a partir de las anteriores. Desafortunadamente, esta técnica no puede ser usada demasiado a menudo, debido a que es muy específica de ciertos dominios, como el del análisis de píxeles de fotografías.

##### 2. Mapeado de los datos a un nuevo espacio

La traslación de datos a un nuevo espacio para intentar ver patrones y características que antes pasaban desapercibidos debido a ruido u otros factores es una técnica sencilla y útil en muchos casos. Se podría definir como “la búsqueda de un nuevo punto de vista”.

En caso de estar buscando patrones, una gran ayuda puede ser la aplicación de la transformada de Fourier, especialmente en el caso de las series temporales, ya que revelará información que, en este caso, tiene de forma explícita la frecuencia.

##### 3. Construcción de características

Finalmente, existe la construcción de características como tercer método de la creación de las mismas. Este método se utiliza cuando en el dataset se tienen los datos correctos para obtener una información determinada, pero el algoritmo de data mining que se usa no acepta esta información. En este caso, la construcción de nuevas características construidas a partir de las originales puede dar lugar a unas características más útiles y aceptadas por el algoritmo determinado.

#### v. *Discretización* y transformación a binario

Muchas veces, cuando se tiene que ejecutar un algoritmo de clasificación, los datos deben de estar en forma categórica. También, algunos algoritmos para encontrar patrones necesitan que la información se encuentre de forma binaria. Así, para el primer caso se habla de técnicas de discretización, mientras que para el segundo se habla de técnicas de binarización. A continuación se ven con mayor detalle:

##### 1. Binarización

La binarización consiste en la técnica mediante la cual, para “m” valores categóricos, se le asigna un valor a cada uno que entre dentro del *intervalo* siguiente:  $[0, m - 1]$ . Una vez hecho esto, se pueden convertir estos números en binario, de tal manera que se obtendrá para cada observación un vector con un valor binario correspondiente a la clase.

La binarización también se puede hacer de forma asimétrica, de tal manera que cada columna represente un estado, y por cada observación sólo puede haber un uno en este vector,

representando al estado al que pertenece, mientras que en el resto de estados esto es igual a cero. Esta técnica se suele dar, por ejemplo, cuando se poseen variables totalmente excluyentes entre ellas, como ocurre en el problema que se plantea en este trabajo.

## 2. Discretización

La discretización de variables continuas depende del algoritmo que va a ser usado principalmente. Todas las discretizaciones conllevan dos subtarefas imprescindibles, que son la elección del número de categorías que habrá, y la selección de intervalos de valores de pertenencia a cada variable.

## vi. Transformación de variables

La transformación de variables es una técnica que se aplica a todos los valores de una variable por diversos motivos. Hay dos tipos de transformación de variables, que se explican a continuación:

### 1. Funciones Simples

Este proceso es tan sencillo como la aplicación de una función matemática a cada valor de la variable en cuestión. Las más usadas son la raíz cuadrada, el logaritmo y la inversa, para poder transformar un conjunto de datos que no siguen una curva gaussiana en otro conjunto que lo cumple.

Estas transformaciones deben siempre de aplicarse con cautela, debido a que cambian la naturaleza de los datos, y no se trabaja por lo tanto con los originales. Por ejemplo, si se aplica la función inversa, los valores superiores a 1 estarán siendo disminuidos, mientras que los menores estarán siendo aumentados. Por ello, siempre hay que hacerse unas preguntas previas, como pueden ser: ¿Es importante el valor exacto del dato? ¿Se necesita tener un orden, una idea de qué observación tiene esa variable mayor que otra? ¿Cómo se aplica esa transformación a las variables extrañas, como al cero?

### 2. Estandarización

La estandarización de variables, también conocida como normalización (no confundir con una transformación gaussiana), es una técnica cuyo objetivo es que todo el dataset, o una parte de él, siga una determinada norma o propiedad.

Esta estandarización es muy necesaria en caso de que alguna de las variables destaque sobre el resto por algún motivo, especialmente si el valor es muy distinto al resto, debido a que cualquier método que use distancias euclídeas tendrá en cuenta la distancia y el peso de las variables, y si se pretende que los modelos aprendan de una manera imparcial se deben de poner todas las variables bajo la misma norma.

### 2.3.2 En este trabajo: Preparación de los datos

Siguiendo las bases teóricas expuestas en el punto anterior, en este trabajo los datos han sido preparados de una manera premeditada para evitar cualquier problema. De este modo, primeramente, se ha conseguido un entendimiento del negocio y de los datos a través de la lectura de libros y entrevistas con una psicóloga profesional.

Para la confección del dataset, se han tenido que tener en cuenta las necesidades posteriores de los algoritmos a usar. Para la simplificación del problema, se han subdividido y posteriormente codificado muchas variables de forma binaria, de tal manera que la variable

obtenga el valor de un uno cuando se dé el caso, y obtenga el valor de cero cuando no se dé, creando manualmente una binarización asimétrica. Esto se puede observar en:

- Relación con el contexto

Para la relación con el contexto, como se comentó anteriormente, hay tres opciones: Relación mala, relación mala por trauma y relación buena. De este modo, estas tres variables formarán un array donde solo una de las tres puede ser posible, de tal manera que, por cada paciente, sólo una obtendrá el valor de un uno, y las otras dos obtendrán el valor de cero.

- Habilidades sociales

Para las habilidades sociales, también explicadas anteriormente, se ha visto que se subdividen en tres valores: Inhibido, asertivo y agresivo. Como, al igual que en las variables anteriores, un paciente solo puede tener 1 tipo de habilidad social, por cada paciente una de ellas tendrá un uno, y las otras dos restantes un cero.

- Distorsiones cognitivas

Las distorsiones cognitivas, al contrario de las variables anteriores, sí pueden darse varias a la vez en un paciente, incluso con pacientes llegando a tener todas. Por ello, las distorsiones cognitivas formarán otro array donde cada distorsión representa a una variable. Si esta variable está presente en el paciente, se marcará con un uno, mientras que si no se presenta se marcará con un cero.

- Impulsividad

Respecto a la impulsividad, esta variable no forma array debido a que no se consideran en este trabajo los distintos tipos de impulsividad, y de este modo sólo será una variable binaria, donde una persona impulsiva tendrá un uno, y una persona no impulsiva tendrá un cero.

Para el resto de datos no se ha tenido que hacer ninguna preparación previa, debido a que posteriormente, mediante código, cualquier cambio puede ser hecho, como la eliminación de columnas o el centrado y escalado (normalización) que posteriormente se explicará y realizará.

Respecto a la discretización, se efectúa cuando es necesaria y en las variables necesarias, siendo la variable correspondiente al grupo la más importante, debido a que es una variable puramente categórica.

Finalmente, si nos atenemos a calidad de los datos, como es un dataset obtenido a mano no hay datos incongruentes ni datos que falten, por lo que, excepcionalmente, en este problema no hay que realizar ningún movimiento en el ámbito de la calidad de los datos.



### 2.3.3 Análisis Exploratorio o Descriptivo

El análisis exploratorio de datos consiste en un conjunto de técnicas estadísticas y de visualización para resumir y visualizar en primera instancia la realidad que se tiene, así como intentar encontrar patrones y relaciones entre los mismos, de tal manera que se pueda responder a alguna pregunta que previamente no se podría con una mirada simple hacia los datos. Además, la visualización de los datos históricos (especialmente en las series temporales) puede dar mucha idea de la posición en la que se encuentra ahora mismo el problema. Esta técnica fue inventada por el estadista John Turkey en la década de 1970.

Además de encontrar patrones y relaciones, el análisis exploratorio de los datos es un elemento que se antoja fundamental a aplicar antes del machine learning, debido a que en la mayor parte de los datasets hay datos *outliers*, que faltan o inconsistentes. Debido a ello, realizar un análisis exploratorio ciertamente profundo, comprobar si las relaciones que existen cuadran con la realidad (mediante un conocimiento previo del negocio), y eliminar variables outliers o que no aportan nueva información debido a su varianza cercana a cero se es realmente requerido para obtener unos modelos posteriores de machine learning que sean rápidos, eficaces y precisos.

Como causa de lo anterior, primero se deben hacer unas pequeñas comprobaciones para comprobar los datos de los que se dispone, que se dividen en la comprobación de los tipos de datos y la comprobación de la calidad de los datos.

Una vez que se han comprobado ambos, se puede empezar con la parte estadística del análisis exploratorio. Hay varios pasos que se pueden cubrir, que los se irán viendo en los siguientes subapartados.

#### 2.3.3.1 Resumen de las estadísticas del Dataset

El resumen de las estadísticas del dataset no consiste en más que un conjunto de números indicando varias características de un dataset. Dichas estadísticas están formadas por un conjunto de apartados que se detallan a continuación:

##### a. Frecuencias y la moda

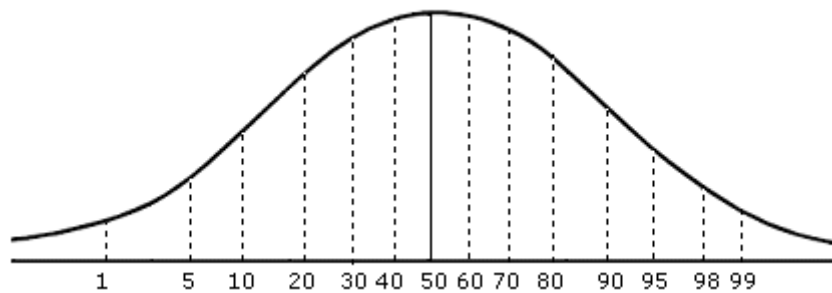
La frecuencia de una variable consiste en un número continuo, con un rango entre 0 y 1, que indica el tanto por uno de ocurrencias de dicho valor de la variable en una lista de  $m$  objetos. Por ello, sigue la siguiente fórmula:

$$F(x) = \frac{n^{\circ} \text{objetos con valor } x}{n^{\circ} \text{objetos totales}}$$

Respecto a la moda, simplemente se hace referencia al valor de cierta variable que tiene una frecuencia mayor que los otros.

### b. Percentiles

Para datos ordenados, el percentil de un conjunto de valores es capaz de aportar una gran cantidad de información. Dado un número  $p$  entre 0 y 100, el  $p$ -ésimo percentil es un valor de  $x$  donde el  $p\%$  de los datos totales son inferiores a ese valor  $p$ -ésimo. Así, se puede obtener qué valores destacan por encima del resto en un determinado porcentaje.



2-5. Percentiles sobre una normal

### c. Media y mediana

Para datos continuos, dos de las estadísticas más básicas y a la vez más solicitadas son la media y la mediana.

Se podría definir la mediana como el valor que está en el medio de un conjunto de datos. Si hubiera un número par de datos, sería la media de los dos valores que ocupan el centro.

La media se puede definir mediante la siguiente función:

$$\bar{X} = media(x) = \frac{\sum Xi}{n^{\circ} \text{ valores total}}$$

La media suele conllevar problemas de medición, debido a que puede estar hecha incluyendo valores outliers, o incluso que sin ser outliers la distorsionen en cierto modo. Por ello fue inventado el concepto de media recortada, que consiste en coger un porcentaje, que suele rondar entre el 1% y el 5%, y desechar ese porcentaje de datos tanto de la parte superior como de la inferior del dataset ordenado, aplicando la media al nuevo dataset recortado.

### d. Rango y varianza

El rango y la varianza son las llamadas medidas de dispersión, ya que miden el dominio en el que se proyectan los datos, y la diferencia entre los valores.

La más simple de las dos es el rango, que se puede medir tanto con la resta del valor más alto menos el valor más bajo, como con un intervalo cerrado donde el primer valor del mismo sea el más pequeño, y el segundo el más grande.

La varianza, aunque más complicada, es el valor preferido a la hora de calcular la dispersión de los datos, y se suele representar como  $s^2$ . La fórmula a la que atiende, siendo  $m$  el total de datos, es la siguiente:

$$s^2 = \frac{\sum (xi - \bar{x})^2}{m - 1}$$

La desviación típica entonces respondería a:

$$s = \sqrt{\frac{\sum (xi - \bar{x})^2}{m - 1}}$$

#### e. Resumen de estadísticas multivariable

Las estadísticas multivariable son aquellas que poseen más de un atributo.

Para calcular las estadísticas en este caso, se debe de calcular la media y la mediana de manera separada a cada una de las variables.

En caso de la dispersión, se puede calcular también de manera separada para cada variable, aunque en este caso se pueden comparar unas medidas con otras mediante la matriz de covarianza. Ésta es una matriz que se representa bidimensionalmente (comparando variables dos a dos), y los valores que se muestran en la matriz son la covarianza de las que forman la fila y la columna.

Dadas las variables  $X_i$  y  $X_j$ , y la cantidad total de variables  $m$ , se podrá calcular la covarianza atendiendo a la siguiente fórmula:

$$cov(X_i, X_j) = \frac{\sum_{k=1}^m (X_{ki} - \bar{X}_i) * (X_{kj} - \bar{X}_j)}{m - 1}$$

También es muy utilizada en data science la llamada matriz de correlación. Esta es otra matriz bidimensional que representa la fuerza de relación lineal entre todas las variables de un dataset, y en cada posición se muestra la correlación entre las dos afectadas. Esta matriz es muy interesante, ya que es simétrica respecto a la diagonal, y además la diagonal consiste en las correlaciones de cada variable consigo misma, lo cual da un resultado siempre de 1 sobre 1.

La correlación sigue la siguiente fórmula:

$$R_{ij} = corr(X_i, X_j) = \frac{cov(X_i, X_j)}{s_i * s_j}$$

### 2.3.3.2 OLAP y Análisis Multidimensional

La visión de la información en arrays multidimensionales conlleva una serie de técnicas determinadas, y unos sistemas de bases de datos que soporten este formato. Muchos sistemas gestores de bases de datos ya soportan este formato, especialmente los sistemas conocidos como OLAP (OnLine Analytical Processing). Debido a esto, el enfoque que se aportará a este apartado estará basado en estos sistemas OLAP.

#### 1) Primer Paso: Representación de los datos

Al igual que muchas veces los datos pueden ser representados en forma de tabla, también en ciertas ocasiones se pueden representar en arrays multidimensionales. Para la visualización, aflora el problema de que no somos capaces de representar más de tres dimensiones, por lo que si se desean representar más dimensiones se deben de hacer varias representaciones en un máximo de 3 a 3. También, para mayor claridad a la hora de visualizar los datos, las tablas y los arrays bidimensionales son las mejores opciones.

Así, en términos generales, el primer paso que se suele dar en la representación de los datos multidimensionales es la creación de una “fact table”, que no deja de ser una tabla donde se representan las combinaciones distintas que se pueden dar de los datos y la cantidad de observaciones que lo cumplen. Cada una de las observaciones de esta “fact table” es única, puesto que cada fila representa una combinación única.

Hacen falta dos pasos para la representación de los datos en un array multidimensional: La identificación de las dimensiones y la identificación de un atributo que sea el objetivo del análisis. Las dimensiones deberán de ser atributos categóricos.

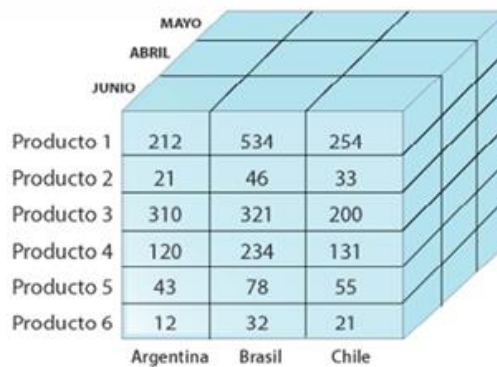
Cada combinación de esta “fact table” será una celda del array multidimensional, que contendrá como valor la cantidad de observaciones que lo cumplían. Así, se podrá también decir que este valor es la cantidad de valores que se pretenden analizar.

#### 2) Análisis de los datos multidimensionales

Para el análisis de los datos multidimensionales hay diferentes técnicas que se pueden usar. A continuación, se analizarán cuatro técnicas muy utilizadas:

##### 1. Data Cubes (Cubos de Datos)

La técnica de data cubes, o los cubos de datos en castellano, consiste en la aplicación de una operación aritmética de las dimensiones restantes de una determinada operación sobre unas dimensiones concretas.



	MAYO	ABRIL	JUNIO
Producto 1	212	534	254
Producto 2	21	46	33
Producto 3	310	321	200
Producto 4	120	234	131
Producto 5	43	78	55
Producto 6	12	32	21
	Argentina	Brasil	Chile

2-6. Ejemplo Data Cube

La representación multidimensional de los datos con todos los resultados de estas operaciones aritméticas se denomina data cube (o cubo de datos). A pesar de llevar el nombre de cubo, las dimensiones de esta figura no tienen por qué ser del mismo tamaño, ni tener tres dimensiones.

## 2. Reducción de la dimensionalidad y pivotaje

Las operaciones aritméticas que se han visto en los data cubes hacen una reducción de la dimensionalidad, puesto que colapsan las celdas de una determinada columna en una única celda.

Si se hace referencia al pivotaje, se habla de reducir todas las dimensiones excepto dos de ellas mediante la agregación. Se podría decir que finalmente el resultado serían los totales de dos dimensiones.

## 3. "Slicing and Dicing"

Esta técnica consiste en dos operaciones muy simples. El "slicing" consiste en la selección de un grupo determinado de celdas de la matriz especificando un cierto valor, que se puede aplicar a una o a varias dimensiones. En el caso de "dicing", se estará haciendo lo mismo pero, en vez de determinar un cierto valor, se determinan un rango de valores.

## 4. "Roll-up and Drill-Down"

Estas técnicas se basan en el concepto de jerarquía, que se le puede aplicar a numerosos datos. Por ejemplo, una fecha se puede dividir en año, mes y día, y una localización se puede dividir en país, región, ciudad, calle.

Así, con este concepto en mente se podrán hacer las operaciones de roll-up y de drill-down. La primera consiste en la agregación de todos los elementos que estén bajo un determinado nivel para conseguir un valor. Por ejemplo, la suma de todas las ventas diarias para conseguir las ventas mensuales. Por el contrario, la segunda consistiría en la subdivisión de todas las ventas mensuales por día, y esto solo podrá darse en el caso de que se tengan los datos de cada día.

Como se puede observar, con estas cuatro técnicas lo que se consigue es la unión de datos similares bajo un determinado filtro que se impone para juntarlos en una determinada celda, y finalmente conseguir reducciones de la dimensionalidad.

#### 2.3.4 Machine Learning

A continuación se expone el núcleo de data science y de este trabajo: El aprendizaje automático o machine learning. Este apartado es el centro debido a que las máquinas pueden aprender de los datos que se les brindan, y con ello predecir o clasificar otros datos de los cuales no se les aporta la solución. Debido a ello, si los algoritmos de machine learning están bien entrenados se abre un gran abanico de posibilidades y respuestas ante las preguntas que se puedan plantear.

La clasificación se podría definir como la tarea de asignar un objeto o un conjunto de ellos a una categoría, normalmente predefinida anteriormente. Algunos ejemplos de uso de clasificación en machine learning son la clasificación de correos para la detección de spam o la clasificación de tumores a partir de imágenes, entre otras muchas.

Dentro de la clasificación podemos hacer una pequeña distinción con los algoritmos predictivos, puesto que estos algoritmos predicen un valor de observaciones desconocidas y continuas, recibiendo también el nombre de regresiones. Los clasificadores trabajan prediciendo grupos en variables discretas.

Las técnicas de clasificación funcionan bien con prácticamente cualquier conjunto de datos, pero siempre se debe tener cuidado, puesto que con los datos ordinales y con las jerarquías los algoritmos de clasificación no funcionan óptimamente. Debido a ello, es importante conocer la naturaleza de los datos antes de empezar con este tipo de algoritmos, tal y como se explicó anteriormente que se debía hacer antes del análisis exploratorio de los datos.

Tras conocer estos pequeños detalles, a continuación se dará un paso más para irse acercando a los algoritmos de clasificación. Es importante antes de codificar saber cómo funcionan estos algoritmos, puesto que no son banales los datos que se le deben de transferir para la creación del modelo y la posterior resolución.

##### 2.3.4.1 *¿Cómo funciona un algoritmo de clasificación en machine learning?*

Un algoritmo de clasificación en machine learning normalmente sigue unos pasos definidos, y que son ciertamente simples.

El primer elemento a tener en cuenta es que los datos deben de tener una dimensión (es decir, una columna) donde se indique la categoría real a la que pertenecen los mismos, en caso de querer realizar aprendizaje supervisado. Tras la obtención de esta columna, se debe de dividir el dataset en dos grupos, siendo el primer grupo para el entrenamiento del modelo y el segundo para el test del mismo. Es importante tener en cuenta que esta división no debe de ser igual, sino que el conjunto de entrenamiento debe de ser muy significativamente mayor al de test, con un ratio de aproximadamente 8 a 2, aunque esto depende de los datos, llegando a obtener resultados muy variados dependiendo de esta agrupación, como se verá posteriormente.

Una vez que se tienen el conjunto de entrenamiento y de test preparados, se puede aplicar este nuevo conjunto de datos de entrenamiento al algoritmo de clasificación que se esté usando

( , K-NN...). La unión del algoritmo y los datos es lo que se conocerá a partir de ahora como modelo, y en este caso será el modelo de entrenamiento.

Este modelo de entrenamiento se deberá poner a prueba ahora con el conjunto de test. Tras obtener la matriz de confusión y calcular el número de aciertos en la diagonal de la misma respecto al número de fallos, se puede calcular la eficacia del modelo y, si es necesario, ajustarlo más para obtener mejores resultados en este test.

Así, la matriz de confusión es una gran aliada en los métodos de clasificación y predicción para ver la eficacia del modelo. Se puede calcular el acierto y el error en tanto por uno de la siguiente manera:

$$Acierto = \frac{\text{Predicciones correctas}}{\text{Total de Predicciones}}$$

$$Error = 1 - Acierto$$

### 2.3.4.2 Problemas y soluciones con los clasificadores

Los errores que se suelen cometer en un problema de clasificación se pueden dividir en dos tipos: Errores de generalización y errores de entrenamiento. El primer tipo se puede definir como el error del modelo cuando se le aplican datos no vistos anteriormente, mientras que el segundo podría expresarse como el número de elementos clasificados erróneamente en el entrenamiento. Es importante destacar que un error de entrenamiento alto no conlleva obligatoriamente un error de test alto, pues puede haber generalizado de una forma correcta y obtener resultados de test aceptables, mientras que un error de entrenamiento muy bajo puede ser debido a que se haya caído en overfitting o underfitting.

Se puede definir overfitting como un sobreajuste del modelo, de tal forma que no generaliza de una manera óptima, sino que está demasiado ajustado hacia el conjunto de entrenamiento, lo que conlleva una mayor tasa de fallos a la hora de clasificar otros datos no vistos previamente.

De la misma forma, se puede definir underfitting como lo contrario al overfitting, que sería una generalización demasiado simple debido a que el algoritmo no ha conseguido aprender bien la estructura y relaciones de los datos.

De estos dos problemas, el más común es el overfitting, puesto que se tiende siempre a intentar mejorar el modelo y a veces esta mejora lo único que conlleva es el empeoramiento del mismo. Debido a ello, a continuación se analizarán algunas de las causas por las que puede haber overfitting en un modelo clasificatorio:

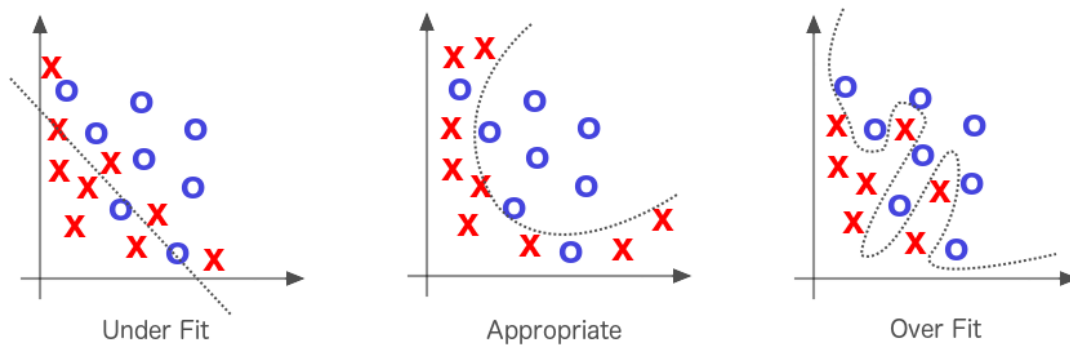
#### a. Debido a la presencia de ruido

La presencia de errores en el dataset hace que los resultados obtenidos de un árbol de clasificación se desvirtúen. Así, cuando unos datos mal clasificados se introduzcan para entrenar un modelo, ese modelo tanto para esos casos como para casos muy similares brindará unos resultados erróneos cuando se prediga con él.

Esto, si lo ponemos en contexto de un árbol de decisión, dará como resultado un aumento del número de bifurcaciones en el árbol, y con ello se obtendrá el overfitting que se intenta evitar. De este modo, y más en datasets de gran tamaño, se suele tomar como éxito un error ciertamente pequeño, puesto que estos errores rara vez son evitables cuando se tienen grandes cantidades de datos.

b. Debido a la falta de muestras realmente representativas

Cuando se hacen clasificaciones con un número de muestras muy pequeño (como en el caso de este proyecto), la probabilidad de que aparezca overfitting es muy alta. La falta de elementos similares con características parecidas que pertenezcan al mismo grupo hace que el modelo generalice peor, con su correspondiente aumento del error de test.



2-7. Underfitting, Óptimo y Overfitting

Como respuesta a este problema, se deben de buscar soluciones para evitar caer en underfitting y overfitting en cualquier clasificador. Para ello hay 3 métodos ampliamente usados, que se exponen seguidamente:

1. Método Holdout

El método holdout comienza dividiendo los datos en dos conjuntos disjuntos, llamados “conjunto de entrenamiento” y “conjunto de test”. Una vez hecho esto, se crea el modelo con el conjunto de entrenamiento y se prueba su eficacia con el conjunto de test.

Este método, como el lector ha podido comprobar a estas alturas, no es en sí un método para mejorar los errores de generalización, pero como se comentó anteriormente hay que tener en cuenta la proporción en la que se dividan los datos, y esto queda a juicio del experto. Si se utilizan demasiados datos para el entrenamiento, se pueden tener demasiados pocos registros para el test y el acierto no ser del todo preciso, mientras que si se obtienen pocos datos de entrenamiento se puede caer en el apartado b anterior: Falta de muestras representativas en el entrenamiento.



Por lo tanto, el método holdout es el primer paso en cualquier creación de modelo de inteligencia artificial. Es un método necesario y de alta importancia, y por ello hay que tenerlo siempre en cuenta.

## 2. Método Random Subsampling

El método de random subsampling (en castellano, submuestras aleatorias) simplemente consiste en la repetición del método holdout  $n$  veces para probar las mejoras que se pueden dar en el rendimiento del clasificador.

Pero random subsampling encuentra problemas respecto al método holdout, puesto que este método no utiliza todos los datos disponibles para el entrenamiento, además de que no tiene ningún control de cuantas veces se utiliza una observación para el entrenamiento o el test, por lo que algunos se usarán para entrenar más veces que otros, y esto puede hacer que quede un modelo desigual y no entrenado óptimamente. Para solucionar este problema, se desarrolló el método Cross-Validation, que se expone a continuación.

## 3. Método Cross Validation

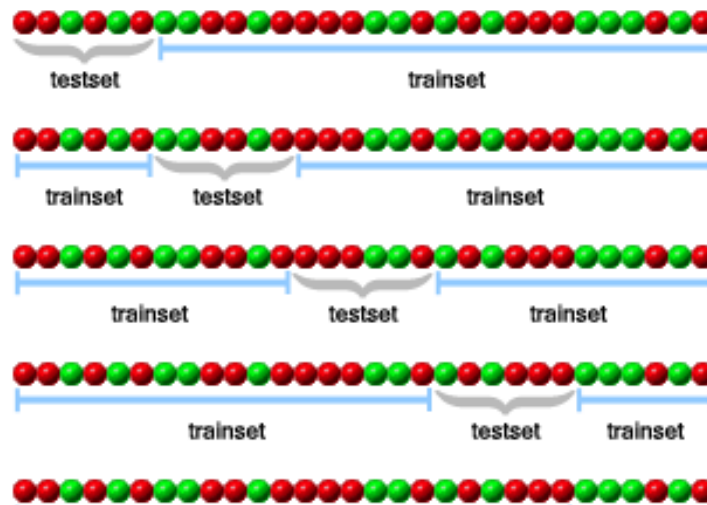
Ante los problemas de los métodos anteriores surge cross-validation (también conocido como X-Validation, y en castellano validación cruzada) y consiste en la utilización de cada uno de los registros el mismo número de veces para el entrenamiento, y solo una vez para test, por lo que se obtiene un modelo mucho más ajustado.

A continuación, debido a la importancia de este método, se procede a explicar detenidamente paso por paso como este algoritmo funciona:

El primer paso es aplicar el método holdout; es decir, particionar los datos en un grupo de entrenamiento y un grupo de test. Se entrena el modelo y entonces se cambian los roles de los grupos, siendo el de test el que hará el entrenamiento y el de entrenamiento el que hará de test. Esto es lo que se conoce como “2 fold cross validation”, puesto que se han utilizado dos grupos.

Si este método se realiza  $k$  veces, se estarán partiendo los datos en  $k$  grupos, y mediante el cambio de los roles de todos los grupos se entrena el modelo. Así, cada uno de los grupos será en un único entrenamiento grupo de test, mientras que será en  $k-1$  entrenamientos grupo de train. Es importante destacar que todos los grupos de entrenamiento se unen a la hora de entrenar un algoritmo en cada una de las iteraciones, formando “un único grupo de entrenamiento”.

En la figura siguiente se puede apreciar un ejemplo de 5 fold cross validation, siguiendo la metodología expuesta anteriormente:



2-8. Ejemplo 5-fold Cross Validation

Un caso especial de cross validation es en el que el número de grupos coincide numéricamente con la cantidad de datos que se tienen en el dataset, y este método es conocido como “leave one out”, que significa dejar uno fuera.

Este método posee una gran desventaja en que computacionalmente es extremadamente costoso, especialmente en datasets ciertamente grandes, y por si no fuera suficiente, la varianza en las métricas de cada entrenamiento y test serán enormes, ya que en algunos tests se obtendrá un acierto del 100% y en otros del 0%. Debido a ello, este método deberá de ser usado como complementario a una validación cruzada con un k más pequeño, y sólo en datasets pequeños.

#### 4. Bootstrap

Con los tres métodos anteriores se ha supuesto que los grupos de entrenamiento eran conjuntos de datos, aleatorios o no, sin reemplazamiento, y por ello no podía haber duplicidades en dicho set de datos, así como en el conjunto de test.

En el método Bootstrap esto cambia, creando muestras con reemplazamiento, por lo que tanto en el conjunto de entrenamiento como en el de test se podrán encontrar muestras duplicadas.

Este método consiste en que los elementos que no se hayan elegido para el conjunto de entrenamiento se usarán en el conjunto de test. Observándolo desde una perspectiva matemática, si en el dataset original se podían encontrar N observaciones totales, la probabilidad de que una observación sea escogida para pertenecer al conjunto de entrenamiento sigue la siguiente fórmula:

$$1 - \left(1 - \frac{1}{N}\right)^N$$

Si N es suficientemente grande, se llega a una asíntota que indica que las posibilidades de elección para cada elemento son del 63,2%. De este modo, con el método Bootstrap se tendrá

un conjunto de entrenamiento de alrededor del 63,2% de los datos, y un conjunto de test que estaría formado por las observaciones restantes.

Una vez que se obtienen estos grupos, se aplican estos datos al algoritmo para obtener el modelo, y se prueba la eficacia. Se puede probar  $k$  veces la eficacia de Bootstrap haciendo la selección de los grupos y luego aplicando el modelo estas  $k$  veces.

Hay diferentes variaciones de cómo se calcula la precisión general del algoritmo. Una de las más importantes es la llamada “.632 Bootstrap”, que combina las precisiones de cada conjunto de entrenamiento hecho por método Bootstrap ( $C_i$ ) con la precisión de un set de test de los datos originales con todos los datos con el grupo al que pertenecen ( $C_t$ ). La fórmula queda expuesta a continuación:

$$\text{Precisión .632} = \frac{1}{k} * \sum_{i=1}^k (0,632 * C_i + 0,368 * C_t)$$

Ahora que se han visto los principales problemas, es hora de hacer una clasificación en profundidad de los algoritmos de machine learning que existen. Hay diferentes métodos de clasificación, puesto que se puede atender al modo en el que aprenden, la estructura que siguen... pero la clasificación que se seguirá en este trabajo será la siguiente debido a las grandes diferencias que existen entre los siguientes grupos:

- Algoritmos supervisados
- Algoritmos no supervisados
- Algoritmos de aprendizaje por refuerzo
- Algoritmos de redes neuronales y deep learning.

### 2.3.4.3 Algoritmos Supervisados

El aprendizaje supervisado comienza con un conjunto determinado de datos, y un entendimiento ciertamente profundo de la estructura de los datos. Este tipo de aprendizaje lo que busca es encontrar patrones en los datos, de tal manera que se puedan hacer procesos analíticos sobre unos datos ya etiquetados.

Estos algoritmos se entrenan usando ejemplos pre-procesados, y su precisión se mide con un conjunto de test que es, como hemos visto anteriormente, excluyente respecto al conjunto de entrenamiento.

Este tipo de algoritmos se utilizan en numerosos ámbitos, como la detección de fraudes, análisis de riesgos, algoritmos de recomendación o incluso el reconocimiento de la voz.

Algunos de los algoritmos más importantes que se encuentran bajo este grupo son:

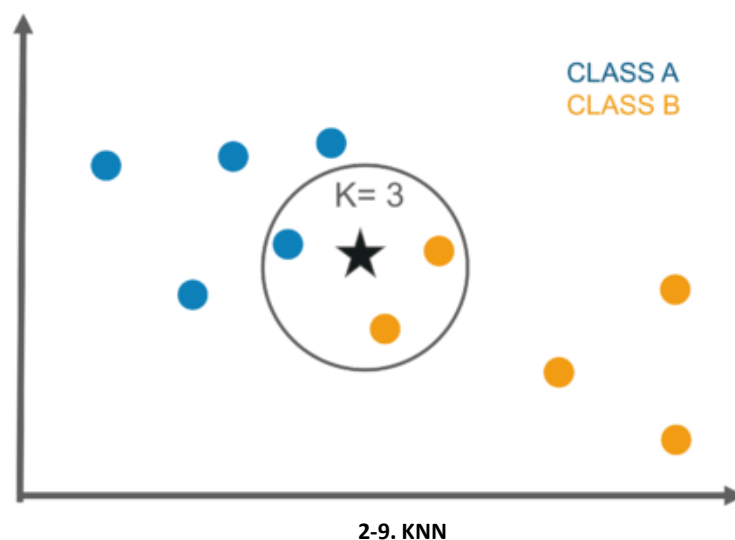
- KNN
- Árboles de Decisión
- Regresión Linear
- SVM

### 2.3.4.3.1 K Nearest Neighbours (KNN)

El algoritmo “k nearest neighbours”, traducido como “los k vecinos más cercanos”, es aquel que se basa en la búsqueda de atributos similares dentro del conjunto de los datos, y así poder predecir la clase a la que pertenece dicho atributo. Llevado hacia un razonamiento más humano, se podría explicar así: “Si se parece a un avión, es tan grande como un avión, vuela y van personas dentro, entonces es un avión”.

Este algoritmo destaca por tener la K delante, que viene a indicar el número de vecinos con los que se va a comparar la observación determinada para obtener cuál es su clase. Así, si K es igual a 1, el elemento que esté más cerca de la observación sobre la que se quiere saber la clase será la que determine la clase de la misma. En caso de que el número K sea un número mayor, la clase que posea más elementos cerca de la observación será la que determine el tipo de la observación. En caso de que haya dos o más clases en estado de empate, se resolverá de forma arbitraria.

Es importante incidir en que la búsqueda de estos K vecinos se realiza de una forma radial mediante distancias gaussianas, de tal manera que, cuanto mayor sea K, más grande será el círculo que se formará alrededor de la observación determinada para buscar los vecinos de la misma.



Como se puede apreciar en la figura 2-9, el elemento sobre el que se quiere predecir la clase, representado con una estrella, ha buscado a los 3 elementos más cercanos y los ha introducido dentro de un círculo. Como se puede observar, es mayoría el número de elementos naranjas a los de azul dentro de este círculo, de tal manera que el elemento cuya clase estamos determinando tendrá inferida la clase B, correspondiente al grupo naranja.

El algoritmo funciona usando la distancia como elemento de similitud, puesto que dos elementos muy cercanos se supone que serán muy parecidos. Así, del elemento del que se quiere saber el grupo se obtiene una lista de elementos cercanos, y usando modernas técnicas de indexación las computaciones que se deben de hacer para obtener esta lista son mucho menores.

Una vez que se tiene la lista, se clasifica en función del grupo que posea la mayoría, donde todas las observaciones de la lista poseen el mismo peso.

El algoritmo KNN alberga una serie de características propias que determinan cuando debe ser utilizado y los “peligros” que entraña, exponiéndose todo ello a continuación:

1. Este algoritmo no requiere de la construcción de un modelo

Como se explicó anteriormente, un modelo es el conjunto de un algoritmo con una serie de datos. El algoritmo KNN no crea modelo, de tal manera que no se pierde tiempo a la hora de la creación del mismo, pero a la hora de la computación del algoritmo es bastante costoso debido a la necesidad de calcular las distancias de todos los elementos a determinar con el resto de elementos.

2. Es un algoritmo que toma decisiones en zonas locales, no globalmente

Como se puede deducir, al mirar sólo a los elementos más cercanos se incurre en que se toma una decisión a nivel local, no como otros algoritmos como los árboles de decisión (que serán vistos posteriormente), que las toman a nivel global. De esta forma, se deberá tener especial cuidado con los valores de K, puesto que un valor pequeño es susceptible al ruido que pueda generar una zona y, por lo tanto, brindar un valor erróneo.

Además, este modelo sufre de underfitting y overfitting muy fácilmente si no se obtiene un valor de K preciso. En caso de obtener un valor demasiado pequeño, debido al ruido se caerá en overfitting, puesto que solo miraremos el elemento más cercano. Si el valor de K, por el contrario, es demasiado grande, se puede sufrir de underfitting y el modelo se volverá demasiado simple. De este modo, como aproximación se suele aceptar que el valor de K sea la raíz cuadrada del número total de elementos a determinar, aunque este valor siempre debe de ser corroborado puesto que depende de los datos del problema.

3. KNN produce predicciones erróneas si no se hace un pre-procesamiento correcto

KNN es un algoritmo muy delicado en términos de pre-procesamiento, puesto que al trabajar con distancias son importante los cambios que se hagan a las medidas de los datos. Por ejemplo, si hay numerosas dimensiones cercanas a un número, y también hay otra dimensión con una variabilidad enorme, el algoritmo no funcionará bien puesto que esta última será la más influyente de todas.

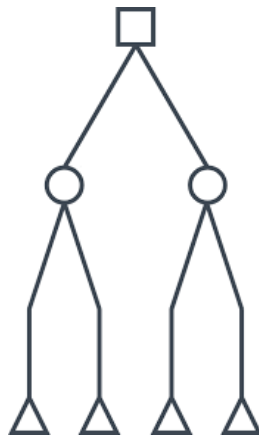
De este modo, para trabajar con este algoritmo un pre-procesamiento a base de centrado y escalado, y una eliminación de las columnas menos importantes escogidas con un PCA sería óptimo.

#### 2.3.4.3.2 Árboles de Decisión

Una técnica muy utilizada en la clasificación que requiere una pequeña mención aparte es la utilización de los llamados árboles de decisión. Esta técnica es de una simpleza extrema, pero a la vez de una gran eficacia en una gran cantidad de problemas.

Los árboles de decisión recordemos que entran dentro de las técnicas de clasificación, y por lo tanto quieren hallar una respuesta a partir de unos datos previos. De este modo, un árbol de clasificación parte de un nodo raíz, que no posee ninguna entrada, pero tiene salidas. Este nodo raíz se formula una pregunta, y según la respuesta que obtenga se irán desarrollando caminos. En cada uno de estos caminos se plantarán nodos, donde se seguirá haciendo preguntas, y seguirá bifurcándose por cada respuesta hasta que sea capaz de llegar a una decisión final en cada camino. Estos nodos que se han ido formando a base de preguntas a partir del raíz serán denominados nodos intermedios, y las respuestas finales se denominarán como hojas, de las que por supuesto no saldrá ningún camino. Por cada nodo por el que pasen los datos, se va haciendo una criba, de tal manera que al final de cada rama, en las hojas, solo queda un pequeño grupo de datos que poseen numerosas características en común.

La construcción de los árboles de decisión, como se puede apreciar, no es demasiado sencilla a simple vista, debido a que se deben hacer las preguntas adecuadas en el momento adecuado, y en un dataset de alta dimensionalidad el gran número de preguntas que se pueden hacer hace que el número de árboles construibles tienda a infinito. Por ello, se han creado algunos algoritmos que construyen árboles de decisión dentro de un espacio óptimo en tiempos razonables, como el de Hunt.



**2-10. Estructura básica de un árbol de decisión**

Además, en la construcción se plantean otros interrogantes, como la elección de la pregunta adecuada o la condición de parada del algoritmo.

Respecto a la primera, el algoritmo que se use deberá de tener un sistema implementado para la evaluación de la bonanza de cada pregunta hacia el propio algoritmo, de cara a aprender si la pregunta ha sido buena o no.

Respecto a la condición de parada del algoritmo, es obvio que es algo obligatorio ya que, en caso contrario, el algoritmo seguiría ejecutándose hasta que se acabaran las dimensiones sobre las que preguntar, y eso no siempre es algo positivo de cara al resultado final. Normalmente se usan criterios tales como que todos los elementos restantes tras las preguntas tengan el mismo

valor, y ese valor será el que se utilizará como hoja final de esa rama y como condición de parada al mismo tiempo.

A continuación, se procede a hacer una explicación de cómo se puede controlar y solucionar el overfitting, puesto que es el problema más común, de tal manera que se puedan mejorar estos árboles de clasificación y obtener resultados más certeros.

### 1. Método de la Pre-poda

En el caso de usar este método, el algoritmo que hace crecer el árbol para antes de formar el árbol completo que encajaría perfectamente con todos los datos de entrenamiento.

Para hacer esto, se debe de poner una condición muy restrictiva para dar por finalizado el algoritmo, como el aumento de una cierta impureza o esencialmente en el error de la generalización.

El problema de esta solución es que, si la restricción es demasiado restrictiva, el modelo quedará en underfitting y por lo tanto será poco certero, mientras que si la restricción es demasiado liviana el modelo caerá en overfitting y por lo tanto generalizará también mal.

### 2. Método de la Post-poda

En caso de decantarse por este método, el primer paso es dejar al algoritmo crecer hasta su máxima extensión, y tras la finalización del algoritmo comienza la poda. Esta se suele hacer obteniendo subárboles, y cambiando estos subárboles por una hoja final perteneciente al grupo que tiene a la mayoría de los individuos en ese subárbol.

Este método es el más utilizado debido a que da mejores resultados, fruto de una poda posterior donde las decisiones de donde recortar vienen dadas de un árbol completamente formado.

#### 2.3.4.3.3 Regresión

La regresión es una técnica predictiva para estimar variables continuas. Algunos ejemplos de predicción a través del uso de regresión pueden incluir el precio de acciones en la bolsa, predicción de ventas en relación con gasto en publicidad...

Una definición más formal de regresión podría darse de la siguiente forma: Tarea de aprender una función objetivo de tal manera que al aplicar un valor  $x$  sobre la misma se obtenga correctamente un valor continuo  $y$ .

De esta manera, el objetivo que tienen los métodos de regresión es encontrar esta función objetivo que posea el mínimo error en la predicción de los datos. La función de error de la misma se suele dar de dos maneras:

$$Error\ Absoluto = \sum_i |y_i - f(x_i)|$$

$$\text{Error cuadrático} = \sum_i (y_i - f(x_i))^2$$

Con estas ecuaciones se aprecia el sumatorio de la diferencia entre el valor real y el valor esperado aplicando  $x$  sobre la función objetivo. En el primer caso, ya que se pueden dar sustracciones negativas debido a que el valor de la función objetivo sea mayor que el real, a cada iteración del sumatorio se le aplica el valor absoluto, puesto que lo que es interesante es la distancia del fallo.

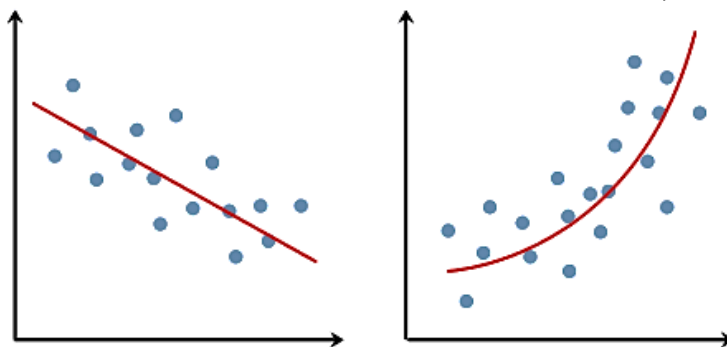
En el caso de la segunda ecuación, al estar calculando el error cuadrático no es necesario aplicar el valor absoluto, puesto que cualquier valor elevado al cuadrado será positivo. En esta ecuación, los errores grandes son penalizados exponencialmente, mientras que los pequeños son minimizados.

Para ajustar la función objetivo al máximo a los datos, normalmente se suele utilizar un método, conocido en la comunidad anglosajona como "Least Square Method", y en la hispanoblante como "Método del Mínimo Cuadrado".

Supongamos que se tiene que calcular la función objetivo para una serie de puntos que llevará el siguiente patrón:

$$f(x) = a * x + b$$

Donde  $a$  y  $b$  son los llamados coeficientes de regresión. Usando el método del mínimo cuadrado, se debe de hallar  $a$  y  $b$  para que la suma de los errores cuadrados sea mínima, y por lo tanto la función objetivo elegida sea la más cercana a la ideal. Sobre esta función objetivo posteriormente se podrán hacer las predicciones de valores futuros.



2-11. Regresiones

Es importante destacar que no todas las regresiones son lineales, como en el caso de la imagen anterior. También existen regresiones no lineales, que son aquellas en las que se ajusta el modelo con una ecuación con coeficiente 2 o superior, de tal forma que la recta de regresión no es recta sino curva. Estas ecuaciones siguen el siguiente formato:

$$f(x) = h * x^k + \dots + i * x^2 + j * x + k$$

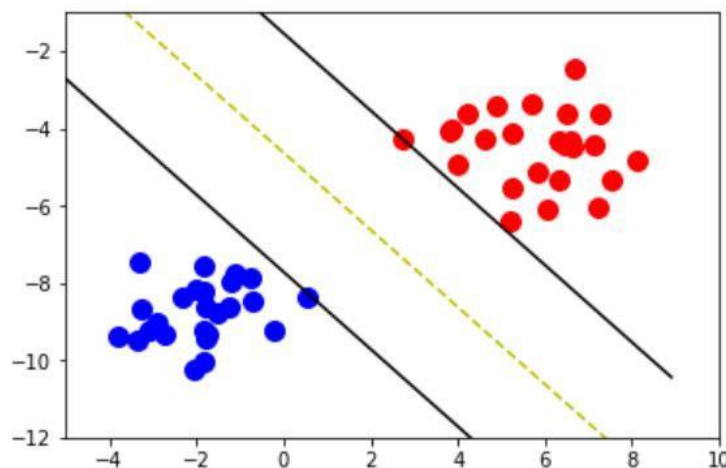


#### 2.3.4.3.4 Support Vector Machines (SVM)

El algoritmo SVM es uno de los algoritmos más utilizados dentro del machine learning, puesto que tiene unas fuertes bases estadísticas y matemáticas y ha demostrado solvencia en muchísimas aplicaciones prácticas, como por ejemplo el reconocimiento de caracteres manuscritos. Para comprender cómo funciona SVM, es importante entender la idea de los hiperplanos, puesto que es el fundamento de esta técnica.

Los hiperplanos son planos infinitos en un determinado espacio. En el caso de SVM, estos hiperplanos se denominan “hiperplanos de margen máximo”. Estos se utilizan como soporte de las fronteras para separar dos o más muestras de datos por el grupo al que pertenecen, de tal manera que se pueda conseguir la mejor separación posible. Cada frontera, al ser infinita en una dirección, solo tendrá dos hiperplanos que la soporten, uno a cada lado.

Si los datos son linealmente separables, habrá infinitas fronteras que los separen, puesto que por un punto pueden pasar infinitas rectas, y por lo tanto fronteras. Pero aquí se plantea un problema, y es que en el caso de que haya un nuevo dato, se tendrá que saber cuál de todas esas infinitas fronteras será la que discierna mejor a qué grupo pertenecerá dicho punto. Aquí es donde entran en acción los hiperplanos. Como se ha visto, cada frontera posee dos hiperplanos, que serán paralelos a la frontera, y su ubicación estará determinada por el elemento más cercano que se pueda encontrar desde la frontera hacia cada una de las clases de una forma ortogonal. De este modo, la frontera ideal será la que tenga una mayor distancia con sus hiperplanos de soporte, puesto que será la que tenga un mayor margen con cada una de las clases, y por lo tanto un menor índice de error, puesto que se hace una mejor generalización. Todo ello es fácilmente apreciable en la siguiente imagen:



**2-12. SVM Linealmente Separable**

En caso de que la frontera deba de ser muy pequeña, normalmente se cae en un caso de overfitting, puesto que el margen de error para tomar las decisiones es pequeño y la frontera tendrá que ser más ajustada para poder tener unos hiperplanos de soporte lo más amplios posibles.

Es importante a partir de ahora destacar que hay dos tipos de SVM, que son el lineal y el no lineal. En el caso del primero, se busca la frontera con el máximo margen con sus hiperplanos de apoyo, lo que hace que se le conozca también con el nombre de “clasificador de máximo margen”. En caso de dar referencia a SVM no lineal, la técnica consiste en la transformación del

espacio en el que se encuentran los datos, de tal forma que se pueda aplicar una frontera lineal para separar las clases del problema. A continuación, se expondrán estos diferentes casos que se pueden dar de acuerdo con esta clasificación, y se detallarán en profundidad:

### 1. SVM Lineal: Caso Separable

En caso de tener un caso separable en un SVM lineal, como se ha comentado anteriormente se está en un caso de clasificador de márgenes máximos. Por ello, la frontera será una recta, y cumplirá la siguiente ecuación:

$$Ax + b = 0$$

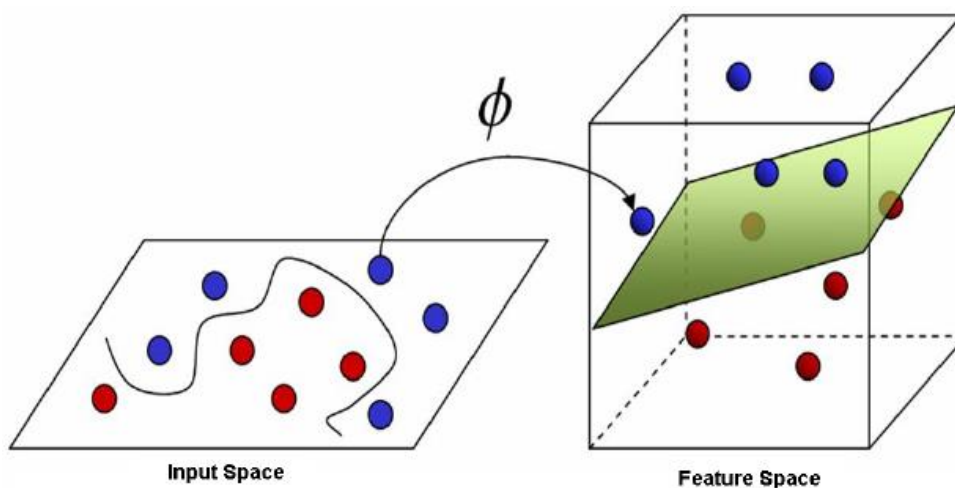
De este modo, el aprendizaje de este algoritmo será la determinación de los valores A y b de la ecuación anterior mediante los datos de entrenamiento.

### 2. SVM Lineal: Caso No Separable

El caso de tener unos datos no separables linealmente hace que haya que tener mucho más cuidado a la hora de elegir la frontera, puesto que a veces muchas fronteras que no incurren prácticamente en errores en el entrenamiento tienen unos márgenes muy pequeños, y generalizan francamente mal. Por ello, en estos casos se debe hacer una aproximación llamada “soft margin”, que consiste en la búsqueda de un equilibrio entre los márgenes de la frontera y el número de clasificaciones erróneas que se dan en el entrenamiento. Esto es reducible en cierto modo en la etapa previa al algoritmo, puesto que si se dividen los datos por su grupo real creando subgrupos, y se detectan los outliers, se puede simplificar este proceso de “soft margin”.

### 3. SVM No Lineal

En el caso de haber un problema de kernel no lineal, se debe hacer la transformación del espacio tal como se explicó anteriormente y como se muestra simplificada en la figura 2-13.



2-13. SVM No Lineal

Así, tras la transformación del espacio, se habrá convertido un problema no lineal en un problema lineal, y se podrán aplicar las técnicas anteriormente descritas para resolver el problema.

Algunas de las características principales de SVM son:

1. Los problemas planteados con SVM son problemas de optimización, que por regla general llegan a la solución encontrando un mínimo global, mientras que otros algoritmos como las redes neuronales artificiales suelen caer en mínimos locales.
2. Al afrontar un problema con SVM, es importante anotar el tipo de kernel que se usará (lineal o radial), y controlar también la función de coste C.
3. SVM también funciona con datos categóricos, pero para ello hay que aplicar una binarización, que se ha visto anteriormente.

#### **2.3.4.4 Algoritmos no Supervisados**

El aprendizaje no supervisado se caracteriza por hacerse sobre un conjunto de datos sin etiquetas de grupo. Así, estos algoritmos tendrán que encontrar patrones en los datos y clasificarlos respecto a estos patrones sin ninguna intervención humana.

Estos algoritmos se utilizan en ámbitos muy diversos, como los sistemas anti-spam de los correos, reconocimiento de imágenes, obtención de información de redes sociales...

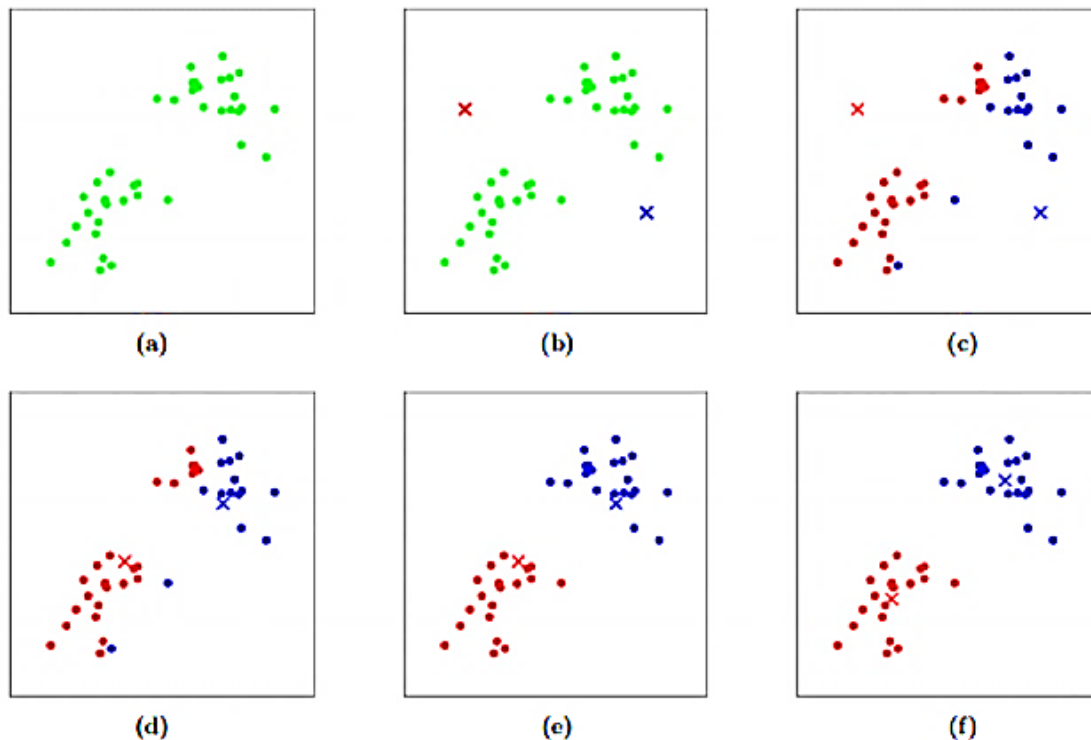
Algunos de los algoritmos más importantes que se encuentran encuadrados en este grupo son:

- KMeans
- Reglas de Asociación

##### **2.3.4.4.1 K Means**

K-Means es una técnica de clustering de datos que se basa en la creación de un centroide, normalmente creado como la media de un grupo de objetos, que se aplica a objetos en un espacio n-dimensional.

El funcionamiento de K-Means es simple: Se empieza con la elección de K centroides, siendo K el número de grupos que se pretende discernir. Entonces, a base de distancias gaussianas, se obtienen las distancias de cada elemento del dataset con los centroides y se va asignando cada elemento a un cierto grupo. Una vez finalizado este primer paso, se redefine el centroide de cada grupo en función de los elementos que estén formando en ese momento el cluster, y se vuelve a empezar. La condición de parada es la falta de elementos que cambien cualquier objeto de un cluster a otro, y por lo tanto que ninguno de los centroides tenga que cambiar su posición. Para evitar costes computacionales innecesarios, en datasets extremadamente grandes se puede aplicar una parada cuando menos de un 1% de los puntos hagan cambios, de tal manera que no se tengan que recalcular ni las distancias ni los centroides y, por lo tanto, evitar



2-14. K-Means paso a paso

iteraciones no necesarias en el algoritmo.

Otra visión que se puede tener del algoritmo es la de que es un algoritmo de optimización, donde la función objetivo debe minimizar las distancias de los puntos con el centroide más cercano.

Es importante destacar que K-Means, como se ha comentado anteriormente, utiliza distancias euclídeas, pero estas no son las únicas distancias que existen. Otra distancia que también sería compatible con este algoritmo sería la distancia de Manhattan, aunque con esta distancia en vez de utilizar la media para calcular los centroides se utilizaría la mediana. Por otra parte, la distancia de Jaccard es una distancia que se suele usar más en el análisis de documentos y la similitud entre los mismos y, por lo tanto no es la más indicada para este algoritmo ni para este problema.

Una vez que se tiene clara la distancia a usar en este algoritmo, es interesante la explicación de la “suma del error cuadrado”. Esta suma, de forma similar a lo visto previamente en la

regresión, consiste en la adición de las distancias de todos los puntos de un cluster con su centroide más cercano. Si esto se realiza para varios clusters, el cluster más acertado será el que posea una suma del error cuadrado menor. Al igual, si se tienen varios sets de clusters distintos, la mejor elección será la que posea la suma del error cuadrado más pequeña. La elección que se haga de los centroides al principio es vital para la suma del error cuadrado final.

La elección de los centroides iniciales es de gran importancia a la hora de iniciar el algoritmo de K-Means, puesto que las diferentes elecciones que se puedan hacer producen diferentes resultados y variar la suma del error cuadrado. Por ello, existen diferentes técnicas para la inicialización de estos centroides:

#### 1. Inicialización de forma aleatoria

La inicialización de los centroides en un punto aleatorio del espacio conlleva que se pueda encontrar un mínimo local que pueda parecer óptimo, pero rara vez se consigue un mínimo global que sea la mejor solución del problema.

#### 2. Sucesión de inicializaciones aleatorias

Una técnica que se suele utilizar es la inicialización del algoritmo  $N$  veces de forma aleatoria, llegando hasta el final y seleccionando los clusters con menor suma de error cuadrado. Esta técnica presenta numerosos problemas, puesto que por una parte es muy costosa computacionalmente, pero además de ello, la re-inicialización del algoritmo sobre los mismos datos conlleva que muchos intentos sean fallidos. Por ejemplo, si se pasa como parámetro al algoritmo  $K = 4$  y existen 4 grupos bien diferenciados, pero tres de los centroides comienzan en uno de los grupos, dicho grupo acabará siendo dividido y por lo tanto la formación de los clusters será errónea.

De esta manera, debido a estos grandes problemas que a veces los algoritmos no son capaces de superar, dependiendo especialmente de los datos y de las necesidades, se han desarrollado otras técnicas que se explican seguidamente:

Un acercamiento que suele resultar bastante efectivo es la obtención de una muestra de puntos y ejecutar el algoritmo de clustering de los mismos mediante clustering jerárquico. Tras la formación de estos primeros clusters, se pueden obtener los centroides de los mismos y aplicar K-Means desde ese punto. Esta aproximación es muy efectiva especialmente si la cantidad de elementos a hacer clustering es pequeña, y es extremadamente efectiva si además  $K$  es un valor también muy reducido respecto al número de elementos a agrupar. Esto es debido a que el clustering jerárquico es una técnica muy costosa computacionalmente, y un clustering de este tipo con una gran cantidad de datos y numerosos grupos tomaría demasiado tiempo computacional como para ser efectivo.

Otro acercamiento también sería la selección “a dedo” del centroide inicial, estando este situado en un punto determinado o siendo el centro de todos los puntos. Una vez realizado esto, los sucesivos centroides que se elijan deberán estar lo más separados posibles de este centroide primero. El problema que conlleva esta técnica es que se debe de hacer un análisis de los outliers perfecto en las etapas previas al machine learning, puesto que al elegir los elementos más separados se puede escoger con gran facilidad un outlier y, por lo tanto, realizar un clustering incorrecto. Otro problema que posee este método es que es bastante costoso

computacionalmente el calcular el punto más alejado de un centroide. Debido a esto, esta técnica solo se utiliza normalmente en subconjuntos, y no en datasets enteros.

K-Means suele tener otros problemas, además de la elección del centroide inicial. En las siguientes líneas se analizan estos problemas y las posibles soluciones que se pueden dar, o las recomendaciones a seguir a la hora de aplicar esta técnica:

a. Manejo de clusters vacíos

Uno de los problemas con los algoritmos de K-Means básicos es que se puede dar que ningún punto sea asignado a un centroide y, por lo tanto, se obtenga un cluster totalmente vacío en las etapas de asignación de puntos a centroides anteriormente vistas. Por esta razón, los algoritmos de K-Means deberán de tener una serie de políticas de reemplazamiento de centroides por otros en caso de que esto pase, porque en caso contrario la suma del error cuadrado será demasiado alta debido a las grandes distancias que se pueden acabar formando en el resto de grupos debido a las clasificaciones erróneas.

Una aproximación que suelen hacer estos algoritmos consiste en coger el punto más alejado, y que por tanto más suma al error cuadrado, y eliminarlo, de tal manera que ningún centroide pueda establecer allí su primera base y por lo tanto no haya posibilidad de obtener grupos vacíos. Si se analiza esta acción con otra perspectiva, se puede observar que este método está realizando una eliminación de un outlier.

b. Problemas con los outliers

De forma obvia se puede inferir que si se usa el error cuadrado, el hecho de que haya elementos outliers influirá de gran manera al resultado final. Esto se debe a que los centroides, tras la última iteración del algoritmo, es improbable que estén situados en el punto óptimo donde deberían de estar, sino demasiado influenciados por los outliers, de tal manera que la suma del error cuadrado aumentará.

Por lo tanto, uno de los mayores problemas a enfrentarse a la hora de hacer clustering, y en especial con K-Means, es el problema de los outliers y su identificación. Existen numerosas aproximaciones para identificar outliers, pero una de las más sencillas es la eliminación de puntos que presenten un error mucho más alto que los compañeros del cluster. También, en el caso de la existencia de clusters especialmente pequeños, es interesante una valoración especial de si el cluster es válido, puesto que puede ser simplemente un grupo de outliers y no un grupo real válido.

Debido a estos dos problemas que se han encontrado, la búsqueda de soluciones para mejorar los algoritmos de clustering es obligatoria. A consecuencia de esto, técnicas como el post-procesado del clustering para reducir la suma del error cuadrado se plantean como técnicas interesantes a emplear, además de otras estrategias que se presentarán a continuación:

Normalmente se puede mejorar el error obtenido aumentando la K, puesto que al haber más centroides, si se inicializan de una manera correcta, los puntos estarán más cercanos a ellos y por lo tanto las distancias disminuirán. Pero normalmente no se pretende aumentar el número de grupos que existen, por lo que el analista debe de empezar a pensar de una manera más global. Si no se está conforme con el clustering realizado, es posible que K-Means haya caído en

un mínimo local, lo que significa que hay una solución mejor pero que no ha sido capaz de llegar a ella. La repetición del algoritmo puede llevar hacia un mínimo global.

En caso de que estas técnicas anteriores no sean satisfactorias, se deberá de pensar en hacer un post-procesado del clustering. Hay dos métodos:

#### 1. Incremento del número de clusters

En caso de que no se haya encontrado un mínimo global y se quiera aumentar la precisión del algoritmo, el aumento del número de clusters, como se ha comentado anteriormente, es una solución. Así, se tienen dos alternativas:

##### a. División de clusters

En caso de la elección de la división de un cluster, se deberá de elegir el que posea un mayor error. También, como opción alternativa, aquel que tenga una mayor varianza puede ser el que sea elegido. Una vez que se tenga elegido el cluster en cuestión, se procederá a la división del mismo en dos o más grupos, de tal manera que se obtengan grupos mucho más cohesionados y cercanos.

##### b. Introducción de un nuevo centroide

En caso de elegir la introducción de un nuevo centroide, la técnica que se suele escoger es la de la elección del punto más alejado de cualquier centroide de los clusters, y la introducción de un nuevo centroide en ese punto.

Debido a elementos explicados anteriormente, esta estrategia tiene dos problemas: El primero es el gran coste computacional que conlleva el cómputo del elemento más alejado de un dataset, y el segundo es la gran posibilidad de obtener un cluster muy reducido con el punto outlier que se elija, de tal manera que se debería de pensar en la eliminación de ese punto en caso de que esto pasara.

#### 2. Decrementar el número de clusters

La reducción del número de clusters, mientras que se intenta minimizar el aumento en el error, puede seguir otras dos estrategias, que serán explicadas a continuación:

##### a. Dispersión de un cluster

La dispersión de un cluster consiste en la eliminación del centroide del cluster en cuestión, y la reasignación de los puntos de ese antiguo cluster a los restantes, siguiendo el mismo procedimiento que se sigue en el algoritmo mediante negociación por distancias euclídeas.

De una forma ideal, el cluster que debe de ser dispersado será aquél que aumenta el error total de una forma mínima.

##### b. Fusión de dos clusters

La unión de dos clusters es una opción muy interesante en caso de tener en el problema dos clusters que sean muy unidos, puesto que al aplicar esta técnica aumentará de una manera ligera el error total. También se puede realizar una computación que, aunque costosa, permite determinar qué dos clusters al unirse producirán un aumento del error total más pequeño y actuar en consecuencia.

A continuación se explican algunas de las fortalezas y debilidades que K-Means posee, puesto que al ser un algoritmo tan utilizado y conocido deberán de ser expuestas claramente:

K-means tiene grandes dificultades para obtener los clusters “naturales” que se pueden dar en la realidad, puesto que se centra principalmente, debido a sus distancias euclídeas, en clusters con un tamaño similar y una forma esférica. Además, suele tener problemas en clusters donde la densidad varía, puesto que suele poner un centroide donde haya una mayor densidad, a pesar de que ahí pueda haber dos o más grupos.

Por otra parte, K-means es un algoritmo que se puede utilizar para numerosos tipos de datos (pero no todos), además de que es bastante eficiente, especialmente cuando el número de datos a clasificar no es extremadamente alto. De esta manera, se pueden realizar varias ejecuciones del algoritmo para poder encontrar el mínimo global. Además, como se ha visto anteriormente, los errores por outliers son fácilmente identificables y solucionables.

#### 2.3.4.4.2 Reglas de Asociación

Los clasificadores basados en reglas de asociación son aquellos que se basan en la clasificación de elementos usando reglas condicionales, de tal manera que siguen la estructura básica de programación if-then, que se puede ver en la forma siguiente:

```

If
...
Then
...
```

Las reglas, una vez formadas, se expresan mediante condiciones disjuntas donde el operador  $\wedge$ , equivalente a la conjunción “y”, delimita las reglas. Finalmente, las reglas acaban con una flecha con dirección hacia la derecha donde se muestra el resultado de la regla. En la tabla a continuación se puede ver gráficamente este formato de presentación de las reglas, donde se hace presentación de 3 reglas distintas:

**Tabla 2-1. Ejemplos Reglas Finales**

Condición 1	Operador	Condición 2	Dirección	Consecuente
Es coche = T	$\wedge$	Es eléctrico = F	$\rightarrow$	Contaminante
Es coche = T	$\wedge$	Es eléctrico = T	$\rightarrow$	No Contaminante
Es coche = F			$\rightarrow$	No es coche

Como se puede ver en la tabla superior, si un elemento es un coche y es eléctrico las reglas lo incluirán dentro de la categoría de no contaminante, mientras que, si es un coche y no es



eléctrico, se introducirá dentro de la categoría de contaminante. En cambio, si no es un coche directamente se introducirá dentro de una tercera categoría “no es coche”.

Es importante remarcar que las reglas que están más a la izquierda son más determinantes a la hora de tomar las decisiones, puesto que son unos “antecedentes” o “precondiciones”, por lo que los elementos que pasen las reglas que estén situadas más a la izquierda serán más similares. Este sistema se parece en gran medida a los árboles de decisión (ver figura 2-10). También es importante comentar que se pueden poner tantas condiciones como dimensiones tenga el problema, o incluso más, puesto que, por ejemplo, en los elementos numéricos pueden darse dos condiciones sobre una misma variable en caso de querer acotarla tanto superior como inferiormente.

Es importante definir dos términos a la hora de hablar de reglas, que son cobertura y precisión:

Cobertura se define como la fracción de los elementos en un dataset que activan una determinada regla.

Precisión, también conocido como factor de confianza, se define como la fracción de los elementos activados por una determinada regla cuyas clases sean iguales a las predichas.

En términos matemáticos, para un total  $D$  de elementos en un dataset, con una regla  $R$ , un número de elementos  $C$  que satisfacen las condiciones, y un grupo real  $G$ , se podrían definir en los siguientes términos:

$$Cobertura(R) = \frac{C}{D}$$

$$Precisión(R) = \frac{C \cap G}{C}$$

Calculemos como ejemplo la cobertura y la precisión de una determinada regla sobre la ilustración 2-15. Pongamos como ejemplo la siguiente regla

$$(Outlook = Rainy) \wedge (Temp = Cool) \rightarrow No$$

en la cual se está diciendo que si el tiempo atmosférico está lluvioso y la temperatura es fría no se jugará. Para la simplificación del ejemplo, no se tendrán en cuenta las condiciones de humedad ni de viento.

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

2-15. Ejemplo en inglés de tabla de reglas

En este caso, lo primero que se deberá de calcular a la cobertura, es decir, cuantas reglas respecto del total cumplen estas dos condiciones. Como se puede observar en las líneas 5 y 6, esas dos suposiciones cumplen la regla ejemplo respecto del total de 14, por lo que la cobertura será la siguiente:

$$Cobertura = \frac{2}{14} = \frac{1}{7} = 0.142 = 14,2\%$$

A continuación, habrá que investigar la precisión de la regla, es decir, de estas dos reglas qué porcentaje han cumplido la predicción

$$Precisión = \frac{1}{2} = 0.5 = 50\%$$

ya que la línea 5 afirma que se puede jugar, mientras que la línea 6 lo desmiente. Ya que mi regla de ejemplo lo desmentía, podemos solo tener un 50% de acierto.

Un clasificador en base a reglas clasifica en función de las reglas que haya cumplido un cierto elemento. De esta manera, el elemento central en estos clasificadores obviamente son las reglas, que pueden ser de dos tipos:

- Reglas mutuamente excluyentes

En un conjunto de reglas, dos reglas serán mutuamente exclusivas si no hay dos reglas en todo el conjunto que puedan ser cumplidas por el mismo elemento del dataset.

- Reglas exhaustivas

Un conjunto de reglas es exhaustivo si hay una regla por cada combinación que se pueda dar de los atributos o dimensiones de los elementos del dataset. Eso asegura que absolutamente todos los elementos del dataset estarán cubiertos por al menos una regla.

Si se junta la propiedad de la mutua exclusividad y la exhaustividad, se obtiene un conjunto de reglas donde cada elemento del dataset está cubierto siempre y sólo por una regla. Desafortunadamente, prácticamente todas las clasificaciones por reglas no cumplen esta situación idílica.

Cuando el conjunto de reglas no es mutuamente excluyente, un elemento del dataset puede ser cubierto por varias reglas. De este modo, el algoritmo se halla ante un problema de decisión. Hay dos estrategias para resolver este conflicto:

- 1) Reglas ordenadas

En esta aproximación a la resolución, las reglas se ordenan de una forma decreciente atendiendo a la prioridad que tengan, que puede atender a parámetros tan diversos como la cobertura o la precisión. Una vez ordenadas, los elementos del dataset irán recorriéndolas en sentido descendente hasta encontrar la adecuada, y en ese momento el elemento quedará clasificado y no buscará más, por lo que se estarán excluyendo las reglas duplicadas menos prioritarias.

- 2) Reglas desordenadas

En esta segunda aproximación, un elemento del dataset puede clasificarse siguiendo varias reglas de clasificación, y el consecuente de cada regla es un voto hacia una categoría. Una vez recorridas todas las reglas, se hace el recuento de los votos hacia cada grupo, y el grupo mayoritario será el que decida la categoría del elemento.

En el caso de haber un empate, otros factores como la precisión de la regla serán vitales para atribuirle unos pesos a los votos y de esa forma desempatar la votación.

El algoritmo que más destaca en la generación de reglas es el RIPPER. Es un algoritmo que escala muy fácilmente con el aumento del dataset, y es especialmente efectivo cuando las clases son desiguales en términos de población, lo que es bastante frecuente. También es bastante interesante a la hora de trabajar con datasets ruidosos, ya que usa un set de validación para prevenir el overfitting, por lo que se plantea como el algoritmo más preciso a la hora de trabajar con datos reales.

Para los problemas que tienen dos clases, el algoritmo RIPPER elige la clase que tiene la mayoría de elementos como clase base, y aprende desde ahí las reglas para conseguir detectar los elementos de la clase minoritaria. En el caso de que haya más de dos clases, el algoritmo

RIPPER ordena las clases según la frecuencia en la que aparezcan de forma que la clase que tenga menos ocurrencias sea la primera, y la más ocurrente sea la última. Tras hacer esta primera ordenación, los elementos que pertenecen a la primera clase son etiquetados como muestras positivas, mientras que el resto de elementos pasan a ser muestras negativas. Una vez realizada esta clasificación, el algoritmo generará reglas para diferenciar entre los elementos positivos y los negativos. Finalizada esta iteración, el algoritmo RIPPER pasará a la segunda clase, donde realizará el etiquetado y la posterior generación de reglas para la diferenciación de los elementos de esa clase. El algoritmo hará sucesivas repeticiones hasta alcanzar la última clase, coincidente con la más frecuente, la cual designará como clase base.

Hay algunas características peculiares de los sistemas clasificadores basados en reglas que son importantes tener en cuenta antes de la ejecución de un modelo de este tipo:

1. Los clasificadores basados en reglas ofrecen modelos descriptivos mucho más fáciles de interpretar que otros métodos de machine learning, pero no tan buen rendimiento, quedándose en un lugar similar al árbol de decisión.
2. El coste computacional de crear un conjunto de reglas para un dataset es muy comparable con el coste que sería el hacer un árbol de decisión del mismo dataset, puesto que un árbol de decisión, como se comentó anteriormente, está compuesto de reglas exhaustivas y exclusivas entre ellas. El problema que poseen los conjuntos de reglas es que raramente se obtienen reglas exhaustivas y exclusivas, por lo que se necesitan rehacer los parámetros de las reglas (y con ello cambiar las fronteras de decisión) para que el algoritmo funcione de una manera óptima.

#### **2.3.4.5 Algoritmos Semi-Supervisados**

Los algoritmos semi supervisados son un tipo de algoritmos que están cogiendo una gran fuerza hoy en día. Aunque se pueden hacer diversas aproximaciones al grupo, son ampliamente conocidos como unos algoritmos en los que solo una parte de los datos están etiquetados, mientras que otra parte, que suele ser la mayoría, no poseen etiqueta.

El procedimiento que realizan es aprender a partir de los algoritmos etiquetados, de tal manera que al pasarle los elementos no etiquetados el modelo pueda hacer predicciones.

Un ejemplo de un ámbito en el que se suele utilizar este algoritmo es un “call center” o centro de llamadas, donde se puede hacer análisis a la voz de la gente y obtener datos como su estado de ánimo o el género, e intentar inferir qué tipo de problema tiene y por lo tanto con qué interlocutor debe de ser redirigido de una manera automática.

Muchos algoritmos supervisados ya explicados pueden hacer la función como algoritmos semi-supervisados, entre los que se incluyen Support Vector Machines (SVM) o Random Forest.

### 2.3.4.6 Algoritmos de Aprendizaje por Refuerzo

Los algoritmos de aprendizaje por refuerzo se basan en un cambio de comportamiento y filosofía respecto a los anteriores. En estos, el algoritmo recibe un feedback desde la parte de analítica de datos, de tal forma que se le va guiando a la mejor solución. Como se puede observar, estos algoritmos no están entrenados a la hora de que el usuario lo use, sino que van aprendiendo a base de prueba y error. Esto conlleva que una serie de errores harán al algoritmo aprender, mientras que una serie de aciertos le aplicarán un refuerzo que le acerquen a la solución.

Como se puede observar, este tipo de algoritmos son muy parecidos a la anteriormente explicada economía de fichas, puesto que una serie de errores (castigos) harán que el algoritmo no siga dicho camino, mientras que una serie de aciertos (premios) harán que el algoritmo siga por esa vía, puesto que está llevando un buen camino de cara al futuro.

Este tipo de algoritmos se utilizan especialmente en robótica y en los personajes de los videojuegos. Un ejemplo del segundo caso es en el que se lucha contra el personaje controlado por inteligencia artificial de cara a un objetivo, y el personaje aprende de los movimientos del jugador que le perjudican para mejorar y poder conseguir el objetivo de una manera más óptima.

También se utiliza este algoritmo para los coches de conducción autónoma. En este caso, el uso del algoritmo es de una dificultad extrema, puesto que la cantidad de obstáculos que puede haber en la carretera, así como imprevistos, es altísimo. Si todos los coches fueran autónomos, mediante comunicación entre ellos resultaría más sencillo, pero en la vida real con conducción realizada por humanos el movimiento de los coches es impredecible.

Este tipo de algoritmos no son demasiado interesantes hacia este trabajo, puesto que como se ha comentado, son mucho más usados en temas de robótica. Debido a esto, simplemente se hará comentario teórico de los mismos a continuación.

Algunos de los algoritmos más importantes de aprendizaje por refuerzo son:

- Q-Learning
- Diferencia Temporal (TD)

#### 2.3.4.6.1 Q-Learning

Q-Learning es considerado como uno de los algoritmos más importantes de aprendizaje por refuerzo. El algoritmo funciona de la siguiente manera:

El primer paso es la inicialización de la llamada “Tabla Q”, de la que se puede ver un ejemplo en la siguiente figura. Esta tabla consiste simplemente en una serie de celdas actualizables en cada iteración donde se calculan los beneficios futuros esperados por cada acción que se haga en la iteración. Así, esta tabla ayuda a guiar hacia la mejor acción en cada momento. Esta tabla Q se inicializará con una dimensión  $m \times n$ , donde  $m$  sea el número de acciones disponibles y  $n$  sea el número de estados a los que se puede optar. Todas las celdas de la tabla comenzarán inicializadas a cero.

		Action					
State		0	1	2	3	4	5
$R=$	0	-1	-1	-1	-1	0	-1
	1	-1	-1	-1	0	-1	100
	2	-1	-1	-1	0	-1	-1
	3	-1	0	0	-1	0	-1
	4	0	-1	-1	0	-1	100
	5	-1	0	-1	-1	0	100

2-16. Ejemplo de Tabla Q

Los siguientes pasos consisten en la elección de la acción a realizar y la realización de la misma. Estos pasos se repetirán en bucle hasta que el entrenamiento del algoritmo cese. En los primeros bucles, como toda la tabla tiene como puntuación cero, las acciones que se elijan serán totalmente arbitrarias, pudiendo caer en elementos nocivos fácilmente, pero actualizando la tabla Q para no volver a caer en los mismos errores. Normalmente estos valores suelen ser negativos en caso de traer perjuicios, y positivos en caso de traer beneficios.

Una vez que el algoritmo está entrenado y la tabla Q correctamente cumplimentada, comienza la fase de evaluación, donde mediante funciones matemáticas se puede conocer la efectividad del modelo entrenado previamente.

#### 2.3.4.6.2 Diferencia Temporal (TD)

El algoritmo de diferencia temporal es un algoritmo que, al igual que el de Q-Learning, aprende del entorno a base de sucesivas iteraciones sin tener un conocimiento previo del mismo. De esta manera, y como parece lógico, tendrá características en común con los algoritmos no supervisados.

Dentro del algoritmo de diferencia temporal, se pueden distinguir 3 algoritmos:

- TD (0)
- TD (1)
- TD ( $\lambda$ )

Pero antes de explicar cada uno de estos algoritmos, es importante explicar cuatro parámetros que deben ser tenidos en cuenta, y son normalmente anotados con letras griegas:

- Parámetro Gamma ( $\gamma$ ): Este parámetro se define como el ratio de descuento, y será un valor que fluctúe de 0 a 1. Cuanto más se acerque a uno, menor será el descuento que se hará en el algoritmo.
- Parámetro Lambda ( $\lambda$ ): Lambda se define como la variable de asignación de crédito. Al igual que el parámetro anterior, fluctuará entre 0 y 1, y cuanto mayor sea el valor mayor crédito se puede asignar a acciones anteriores.
- Parámetro Alfa ( $\alpha$ ): El parámetro alfa se puede denominar como el ratio de aprendizaje. Como los parámetros anteriores, este ratio se moverá entre 0 y 1, y un valor más cercano al 1 conllevará un ajuste muy agresivo del modelo, mientras que un valor más cercano a cero incurrirá en un modelo con un aprendizaje más conservador.
- Parámetro Delta ( $\delta$ ): Se define como delta cualquier cambio o diferencia de valor en el algoritmo.

Una vez que se conocen estos parámetros, se puede proceder a explicar los tres algoritmos que quedaron pendientes anteriormente: TD (0), TD (1) y TD ( $\lambda$ ).

#### - TD (1)

El primer algoritmo que se debe de explicar es el TD (1). Este algoritmo se caracteriza por actualizar los valores de la misma manera que el *método de Monte-Carlo*, es decir, al final de cada pasada.

Cuando se ejecuta una acción con este algoritmo, se hace una actualización a los estados previos, con un lambda correspondiente al número entre paréntesis. En este caso, al ser lambda con valor uno, el crédito que se le puede aplicar a las acciones anteriores tiene el valor extremo por encima. Esto es de vital importancia, debido a que este algoritmo funciona de una manera similar que el método de Monte-Carlo; es decir, de una manera episódica que necesita un final establecido.

Este algoritmo se basa en la fórmula siguiente:

$$G_t = R(t + 1) + \gamma * R(t + 2) + \dots + \gamma^{t-1} * R(t)$$

Si se interpreta parte a parte, podría leerse de la siguiente manera:  $G_t$  es la suma de los beneficios descontados. Cuando se va avanzando por el entorno, se van anotando los beneficios y los perjuicios, y todos los futuros se van multiplicando por un descuento (recordemos que estará entre 0 y 1, por lo que hará la función de descontar). Finalmente, en un futuro más lejano hay un descuento más acusado, puesto que se eleva el descuento por el número de iteraciones menos uno.

Una vez que se tiene solventada esta ecuación, hay que realizar una actualización al valor estimado, representado por  $V(s)$ . Al igual que en Q-Learning, se empieza con una tabla toda inicializada a ceros o a valores aleatorios, de tal forma que los movimientos son totalmente descontrolados y erráticos al principio hasta que el algoritmo empieza a aprender.

Con esto, lo que se hará es la sustracción del  $G_t$  recién calculado menos el error estimado anterior, lo que dará un resultado conocido como "Error TD". Esta sustracción será multiplicada

por alfa, de tal manera que se pueda ajustar el ajuste del error. Por lo tanto, esta fórmula quedaría de la siguiente manera:

$$V(s) = V(s) + \alpha(Gt - V(s))$$

Una vez efectuada esta cuenta, se habrá completado la primera pasada del algoritmo o episodio.

#### - TD (0)

Una vez que se ha explicado TD (1), TD (0) es un algoritmo mucho más simple de entender, puesto que solo hay una diferencia entre ambos, aparte del valor de lambda.

Esta diferencia radica en la ecuación del cálculo de  $V(s)$ , que es conveniente recordar que es el valor estimado. En este caso, en vez de usar  $Gt$  para el cálculo de la diferencia teniendo en cuenta todas las recompensas futuras, solo se mira la recompensa futura más inmediata, representada por  $R_{t+1}$ , más el descuento que hay que aplicar a esa misma iteración, representado por  $V_{s+1}$ . De este modo, las ecuaciones quedan de la siguiente manera:

$$Gt = R(t + 1) + \gamma * R(t + 2) + \dots + \gamma^{t-1} * R(t)$$

$$V(s) = V(s) + \alpha( R(t + 1) + \gamma * V(s + 1) - V(s) )$$

#### - TD ( $\lambda$ )

Este algoritmo se utiliza cuando se quieren poder actualizar valores antes del fin de ciclo (restricción del TD (1)) o cuando se quiere utilizar más de un valor futuro para la estimación (restricción del TD (0)).

Es importante destacar que hay dos implementaciones de TD ( $\lambda$ ), una hacia delante y otra marcha atrás. Ambas implementaciones son equivalentes en efectividad, pero a continuación se explicará la implementación marcha atrás para no alargar demasiado el apartado.

En esta visión del algoritmo se actualizan los valores en cada paso. De este modo, tras cada paso se actualizan todos los valores de los pasos previos. En estos casos hay que hacer la asignación de crédito explicada anteriormente, pero no se ha determinado un valor lambda para esta función, por lo que se utiliza un valor calculado llamado “Rastreador de Elegibilidad”. Este valor guarda un registro de la cantidad de veces que el algoritmo recae en un estado determinado, y asigna crédito teniendo en cuenta el número de veces que se ha visitado ese estado y si ese estado tiene relación con llegar al estado final, con lo que se conseguirá una tabla finalmente muy precisa en la búsqueda del camino hacia este último estado.

Las ecuaciones que rigen este algoritmo son las siguientes:

$$RE = \lambda * \gamma * E(t - 1) (s) + 1(S = s)$$

$$V(s) = V(s) + \alpha( R(t + 1) + \gamma * V(s + 1) - V(s) ) * Et(s)$$



### 2.3.4.7 Algoritmos de Redes Neuronales y Deep Learning

Deep Learning, o aprendizaje profundo, es una técnica de machine learning que recrea una red neuronal artificial formada por una serie de capas ocultas, de tal manera que el algoritmo pueda aprender de una manera iterativa. Estos algoritmos han sido clasificados en un grupo distinto debido a que pueden ser tanto supervisados como no supervisados, por lo que poseen un trato distinto.

Las redes neuronales artificiales, especialmente al principio, surgieron como un intento de copia de las redes neuronales biológicas para poder trabajar con abstracciones, al igual que la mente humana. Una red neuronal artificial consiste en una red de nodos, llamados neuronas, que se distribuyen normalmente en un mínimo de tres capas:

1. Capa de entrada: Capa que recibe los datos introducidos a la red.
2. Capa oculta: Capa con un número muy variable de neuronas donde los datos se modifican para el entrenamiento de la red. Esta capa es opcional, y puede haber más de una, lo que marca una de las diferencias entre una red neuronal o una red deep learning.
3. Capa de salida: Capa en la que también se modifican los datos y finalmente se ofrece un resultado. Esta capa suele tener funciones de activación diferentes al resto.

Con Deep Learning se hace referencia a una técnica dentro de machine learning en la que se usan redes neuronales de una forma jerárquica, donde cada red neuronal puede tener hasta millones de nodos densamente interconectados. Además, como se indicó previamente, se diferencia también de las redes neuronales tradicionales en que suele tener más de una capa oculta.

Las redes neuronales se suelen utilizar en ámbitos como el reconocimiento de imágenes y la visión artificial, aunque también pueden actuar como algoritmos de regresión y clasificación.

#### 2.3.4.7.1 Redes Neuronales

Como se ha comentado anteriormente, las redes neuronales artificiales surgieron con la intención de simular redes neuronales humanas. El cerebro humano consiste principalmente en neuronas que se intercomunican con otras neuronas mediante unos extremos llamados axones, que producen y reciben impulsos eléctricos. De una manera simplificada, podríamos decir que las neuronas humanas están conectadas a los axones de otras neuronas mediante las dendritas, que son extensiones del cuerpo de la neurona. La acción de enviar información por una dendrita hacia el axón de otra neurona es conocido como sinapsis.

De forma análoga, las redes neuronales artificiales tienen una serie de nodos interconectados entre sí que pasan una cierta información a otros nodos, de tal manera que se acaba llegando a una solución final.

La red neuronal más sencilla es el perceptrón. Debido a su sencillez, a continuación se explicará el funcionamiento del mismo, así como los modelos pueden ser entrenados para resolver problemas de clasificación.

El perceptrón consiste en un tipo de red neuronal artificial que posee dos tipos de nodos, siendo unos de entrada y otro de salida. Los nodos de entrada sirven para representar la entrada de datos, y habrá un nodo por cada dimensión del problema. El nodo de salida obtendrá y sacará el resultado del problema. Estos nodos son los llamados neuronas.

En los perceptrones, los nodos están directamente conectados al nodo de salida mediante una especie de enlace con un peso determinado, que se podría comparar análogamente con la sinapsis. Este peso es el elemento más importante de la conexión, puesto que controla la fuerza de dicha conexión y por lo tanto su importancia en el resultado final, puesto que el nodo de salida hace la suma de cada dato de cada neurona, pero multiplicado por el peso de dicha conexión. Cuando un modelo de perceptrón se entrena, se calculan estos pesos hasta que el resultado final coincide con el que debería de obtenerse. De este modo, al introducir nuevos datos, las relaciones y sus pesos estarán ya establecidos y por lo tanto el resultado de este nuevo dato será, en condiciones óptimas, correcto. Los pesos, al iniciar el entrenamiento de la red neuronal normalmente son inicializados de manera aleatoria.

Es importante destacar que en el perceptrón no hay ninguna capa intermedia u oculta, ya que si no estaríamos entrando en el campo del deep learning. Estas capas ocultas son otras capas que se pueden añadir a la red neuronal entre la capa de entrada y la capa de salida, y permiten hacer más operaciones de cara a la obtención del resultado final, complicando más la red y pudiendo obtener resultados óptimos de relaciones más complejas. Además, es también importante destacar que el número de funciones de activación de los nodos es muy reducido, lo cual también contribuye a que los resultados obtenidos por el perceptrón sólo sean buenos en problemas de bajísima complejidad.

En el proceso de aprendizaje o entrenamiento del perceptrón se tiene en cuenta un factor muy importante, llamado “ratio de aprendizaje”. Se podría definir como un hiperparámetro de las redes neuronales en el que se controla cuanto se están cambiando los pesos de una red neuronal en función del descenso del *gradiente*. En el caso de que este ratio sea pequeño, la red neuronal avanzará poco a poco, lo cual es un elemento a favor ya que se buscará con cautela un mínimo local donde converger, pero por otra parte cuanto más pequeño sea el ratio de aprendizaje más costoso computacionalmente será entrenar a la red. El ratio de aprendizaje también es ampliamente conocido con otro nombre: Decay.

Por lo tanto, en resumen, se puede afirmar que el cálculo de los nuevos pesos de la red neuronal atiende a la siguiente fórmula:

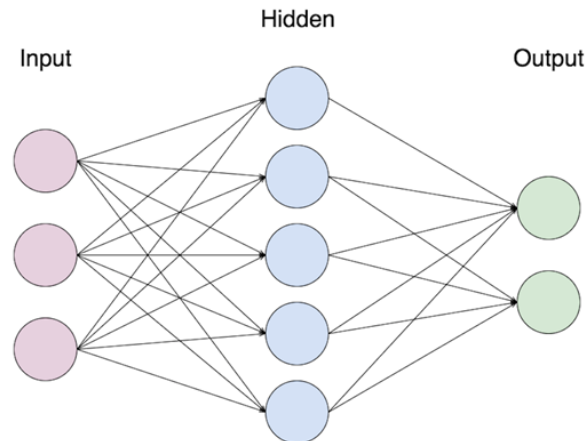
$$newPeso = oldPeso - rApren * gradiente$$

Continuando con la explicación, ahora el análisis se deberá de hacer sobre las redes neuronales artificiales multicapa. Estas redes son la evolución del perceptrón, debido a la falta de potencia del mismo ya que no es capaz de separar dos grupos de datos que no sean separables mediante un hiperplano.

Las redes neuronales multicapa tienen una serie de características propias:

- 1) La existencia de capas ocultas con nodos ocultos en ellas, al contrario del perceptrón.
- 2) El número de funciones de activación en este caso no se reduce sólo al signo, de tal manera que se pueden producir resultados no lineales.

Estos elementos hacen que estas redes neuronales más complejas permitan obtener resultados positivos sobre datos que poseen relaciones más complejas entre ellos, aunque



**2-17. Red Neuronal Multicapa Feed Forward**

computacionalmente sean más costosas. Es importante destacar que, cuantas más neuronas haya en la capa oculta, la red neuronal va a poder resolver problemas más complejos, pero a la vez hay mayores posibilidades de caer en overfitting. Debido a ello, el control del mismo con técnicas como el decay es algo que siempre se debe de tener en cuenta.

Además del hiperparámetro decay, es interesante el uso de otro hiperparámetro llamado softmax. Esta *función de activación*, de una manera simple, lo que hace es la transformación de los números que llegan a una cierta capa (normalmente la de salida) de la red neuronal en probabilidades, cuya suma total es igual a uno. Con ella, se consigue suavizar la posibilidad de que dos elementos puedan tener el mismo peso o pesos demasiado diferenciados, donde en este segundo en caso de equivocación el error tiende a infinito.

Suele usarse en clasificaciones, aunque uno de sus mayores problemas es el hecho de que no tiene en cuenta la posibilidad de que un elemento pueda pertenecer a dos categorías. En caso de que en el problema esto no sea así, siempre es bueno intentar resolver el problema con una red neuronal donde se tenga softmax activado para intentar mejorar los resultados en la capa de salida, puesto que con este hiperparámetro conseguimos probabilidades de pertenencia a cada grupo en las neuronas de salida.

Dentro de las redes neuronales multicapa, es interesante distinguir entre dos tipos:

#### 1) Redes Neuronales Feed Forward

En las redes neuronales feed forward, los nodos de una capa están conectados únicamente a los nodos de la siguiente capa. Esta red se puede ver de forma esquematizada en la figura 2-17.

#### 2) Redes Neuronales Recurrentes

En las redes neuronales recurrentes, los nodos pueden estar conectados a cualquier nodo, incluyendo los de la siguiente capa, los de capas anteriores o incluso a nodos en la misma capa.

Para entender cómo funciona una red neuronal multicapa, es importante entender cómo funciona el perceptrón, puesto que el fundamento es el mismo: El cálculo de los pesos de las conexiones entre nodos para poder reducir el error al clasificar al máximo.

En la mayoría de ocasiones, el output que ofrece una red neuronal multicapa no es una función lineal, y esto es debido a las funciones de activación que se hayan elegido. Esto lo que hace es que las soluciones que se puedan obtener no tengan por qué ser las globalmente óptimas, y algunas soluciones como el gradiente descendente han sido codificadas para intentar mejorar este problema de optimización.

El método del gradiente descendente puede ser usado para aprender los pesos de las capas intermedias y de salida de la red neuronal. Para los nodos que estén en la capa oculta, la computación a realizar no es para nada trivial, puesto que es difícil saber el error que generan esos nodos sin saber cual es el valor real que deberían de tener, debido a que no son la capa final. Ante este problema, se construyó una solución llamada backpropagation (en español, retro propagación), en la que se distinguen dos fases claramente diferenciadas:

1) Hacia delante

En esta primera fase, los pesos obtenidos en la iteración anterior son usados para computar el valor de salida de cada neurona en la red. Esta computación se da en orden, empezando por las neuronas de entrada y siguiendo un estricto orden hasta las neuronas de salida.

2) Hacia atrás:

En la segunda fase, y siguiendo un estricto orden desde las neuronas de salida hasta las de entrada, la fórmula de actualización de pesos se vuelve a ejecutar. Esta segunda pasada, elemento que aporta la retro propagación, ayuda a usar los errores de las neuronas en la capa  $n+1$  para estimar los errores de las neuronas anteriores, de la capa  $n$ .

A continuación se hace un análisis de las características de las redes neuronales, así como sus ventajas y desventajas:

- 1) Las redes neuronales con una capa oculta al menos son unos aproximadores universales, lo que significa que se pueden utilizar para aproximar cualquier función a su óptimo. Esto las hace realmente útiles para hacer una primera aproximación a cualquier problema, pero por otra parte es frecuente caer en overfitting con ellas debido al intento de mejorar en exceso el problema.
- 2) Las redes neuronales artificiales son esencialmente buenas cuando se tenga la duda de si alguna variable no es demasiado interesante, puesto que ellas mismas reajustarán los pesos en función de la relación que tengan los datos, por lo que no hay que preocuparse de ello. Sin embargo, esto no indica que no se deba de hacer alguna técnica de reducción de la dimensionalidad previamente, puesto que a mayor dimensionalidad del problema, más neuronas en la capa de entrada tendrá que haber y mayor coste computacional habrá en el problema.
- 3) Las redes neuronales son muy sensibles hacia la presencia de ruido, especialmente en el set de datos de entrenamiento. Debido a esto, el planteamiento de soluciones es algo importante, donde se puede introducir un decay para reducir el overfitting o usar un set de test para comprobar el estado de la red neuronal tras el entrenamiento.
- 4) El método del gradiente descendente normalmente dirige el modelo hacia un mínimo local. Esto deberá ser tomado en cuenta siempre, y entrenar el algoritmo sucesivas veces

comprobando el mejor resultado en el test siempre es una buena práctica en caso de ser computacionalmente posible.

- 5) En relación con el apartado anterior, el entrenamiento de una red neuronal artificial puede conllevar un gran tiempo de procesamiento, especialmente cuando la cantidad de datos y/o dimensiones es muy elevada. Esto también puede verse agravado por el número de nodos intermedios, puesto que se tendrán que hacer más cálculos por cada pasada en la red neuronal.

#### 2.3.4.7.2 Deep Learning

El deep learning, o aprendizaje profundo, consiste en una técnica de aprendizaje automático que permite a los modelos neuronales multicapa de machine learning aprender representaciones de datos con un nivel de abstracción mucho mayor que con las redes neuronales convencionales. Las dos principales diferencias con las redes neuronales multicapa son las siguientes:

- 1) Las redes de Deep Learning suelen tener varias capas ocultas, debido a que cada capa se suele especializar en la obtención o transformación de una cierta información.
- 2) Las redes de Deep Learning son redes que se basan en la extracción de características, ideología contraria a las redes neuronales multicapa que se basan en la transformación matemática de datos mediante funciones.

Su principal baza, siguiendo con el punto 2, es que la extracción de características del dataset es más potente que la extracción con métodos tradicionales, incluso cuando estos han sido refinados para el problema concreto con expertos humanos. Los modelos de deep learning son utilizados hoy en día en numerosos ámbitos, tan diversos como el reconocimiento de voz, reconocimiento visual, genómica e ingeniería genética o creación de imágenes.

Para la creación y el correcto funcionamiento de un modelo de deep learning hace falta un dataset ciertamente grande, donde la red neuronal profunda pueda aprender la estructura del mismo y las relaciones entre las observaciones. Esto se suele realizar mediante una técnica que se ha visto anteriormente, llamada backpropagation. Otra razón por la que las redes deep learning deben usar grandes datasets es debido a que normalmente estas redes tienen una gran cantidad de neuronas divididas en más de una capa oculta para conseguir funciones complejas que permitan obtener resultados precisos y evitar el ruido, pero para ello hace falta hacer numerosas actualizaciones de los pesos de la red.

Las redes neuronales de deep learning poseen algunas variaciones en su estructura para poder enfrentarse a diversos problemas. Una de estas variaciones, ampliamente reconocida en la comunidad internacional, es la de las redes neuronales convolucionales, mientras que otra también ampliamente usada es la de las redes recurrentes. Se explicarán a continuación en los siguientes apartados:

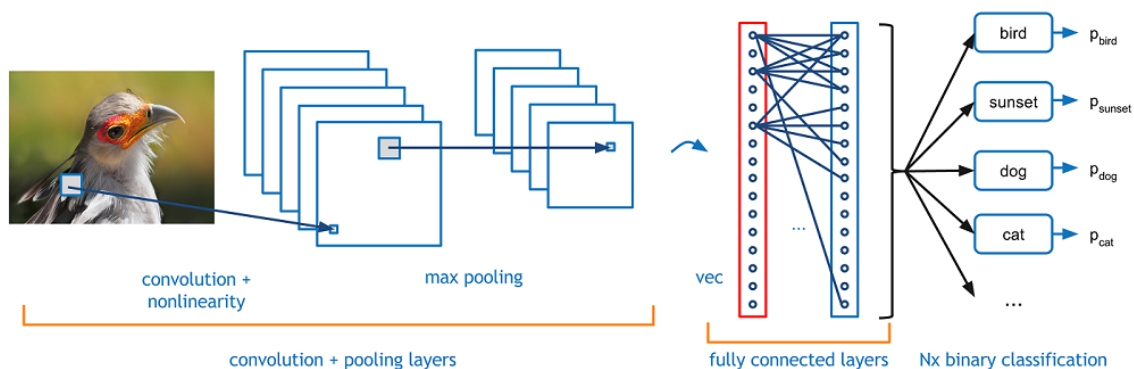
### 2.3.4.7.2.1 Redes Convolucionales

Las redes neuronales convolucionales son redes designadas para tener un input de varios arrays. Un ejemplo de esto es una imagen que pueda venir dada en tres arrays de dos dimensiones, donde en cada array esté incluido la intensidad de cada pixel en el formato RGB. Estas redes destacan principalmente por tener cuatro características principales: Conexiones locales, pesos compartidos, pooling y la posesión de numerosas capas ocultas.

Las redes convolucionales pueden ser concebidas como un conjunto de etapas. Las primeras etapas conllevan dos tipos de capas: Capas convolucionales y capas de pooling.

En el caso de las capas convolucionales, las neuronas se organizan en unas estructuras llamadas “mapas de características”, donde cada una está conectada al mapa de características de la capa anterior a través de una serie de pesos, que reciben el nombre de “banco de filtros”, que comparten todas las neuronas dentro del mapa de características.

La razón para usar esta arquitectura de capas convolucionales es doble. Por una parte, especialmente cuando se trabaja con imágenes, hay muchos datos fuertemente correlacionados, por lo que los no correlacionados se encuentran fácilmente. Si esta característica se la aplicamos a la estructura de la red, donde una serie de neuronas comparten los mismos pesos y pueden buscar patrones y diferencias por toda la imagen, podemos resaltar con facilidad las diferencias entre dos imágenes y por lo tanto clasificarlas correctamente a pesar de que puedan tener una gran similitud.



2-18. Esquema de Red Convolutional

Con las capas de pooling se pretende juntar características similares en una sola característica, de tal manera que se pueda hacer una reducción de la dimensionalidad del problema. Con esto se consigue una invarianza de los pequeños cambios que pueda tener la imagen, mientras que los que sean más grandes quedarán aún más descubiertos, por lo que estas capas de pooling serán una ayuda a las capas convolucionales a la hora de buscar pequeñas diferencias, por ejemplo, en imágenes.

Normalmente, unas dos o tres capas de estas redes convolucionales suelen ser apiladas al principio de la red neuronal profunda, seguidas por más capas convolucionales totalmente conectadas.

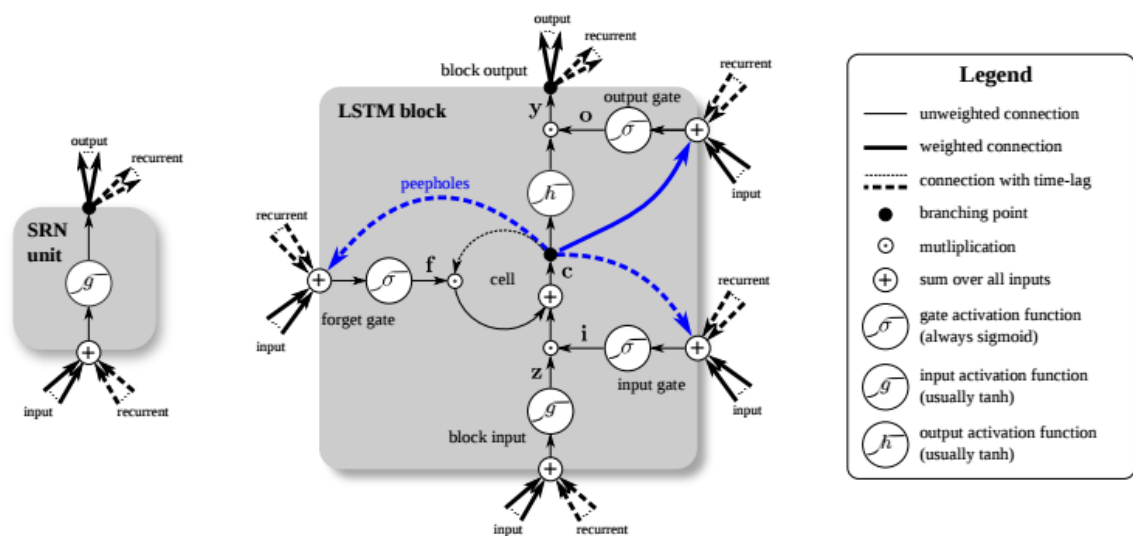
### 2.3.4.7.2.2 Redes Recurrentes

Las redes neuronales recurrentes son aquellas que son perfectas para resolver tareas que requieren inputs secuenciales, como por ejemplo problemas que tengan que ver con el habla.

Estas redes neuronales se caracterizan por tener un vector de todo lo que ha ocurrido anteriormente con la secuencia introducida en la red, llamado “vector de estado”. También se caracterizan por tratar cada output de la red como etapas distintas en el tiempo. Juntando estos dos elementos queda clara la importancia de la retro propagación en este tipo de redes neuronales.

Las redes recurrentes han sido probadas como sistemas extremadamente potentes, pero a la vez extremadamente frágiles. Esta fragilidad se debe a que al hacer retro propagación de tantos pasos los pesos suelen tender hacia los extremos, que se corresponden con 0 y el infinito. Debido a ello, los desarrolladores de las redes neuronales recurrentes tuvieron que hacer investigaciones y avances en la arquitectura de las mismas y en la manera de entrenarlas, obteniendo resultados muy favorables con la creación de las redes LSTM, que han propiciado que hoy en día las redes recurrentes se utilicen en cualquier tarea relacionada con el texto o el habla.

Las redes recurrentes más sencillas son las redes SRN, también conocidas con el nombre de Elman. Estas redes Elman se basan en una retroalimentación de la salida hacia la misma red, permitiendo que el vector de estado tenga memoria y temporalidad. Suelen utilizarse como primera aproximación hacia el reconocimiento de la voz.



2-19. Red SRN vs Red LSTM

Otra red recurrente también ampliamente utilizada es la anteriormente comentada red LSTM, inventada en 1997. Estas redes surgieron para solventar los problemas teóricos que tenían las redes Elman, como el crecimiento o decrecimiento exagerado de los pesos. Debido a ello, estas redes incorporan una memoria con la siguiente información:

- Control de cuando puede entrar nueva información en la memoria
- Control de cuando se puede olvidar la memoria de cierta información
- Control del uso de ciertos datos almacenados en la memoria

Al ser más avanzadas que las redes Elman (ver figura 2-19), estas redes neuronales se utilizan para la comprensión del lenguaje natural o de la escritura a mano, ya que estos problemas son realmente complejos.

### 2.3.5 Visualización

La visualización es una de las partes fundamentales de cualquier proyecto relacionado con datos, puesto que el entendimiento de las conclusiones depende mayoritariamente de los gráficos que se muestren al público interesado. De este modo, las visualizaciones pueden ser una fuente de profundo entendimiento, pero también pueden ser una fuente de confusión.

Uno de los elementos más importantes que hay que tener en cuenta es el público al que se dirige la presentación, tanto por el registro lingüístico a usar como por los gráficos a utilizar, siendo ambos factores fundamentales.

Se debe distinguir entre gráficos para presentación y gráficos para exploración de datos:

- Gráficos para presentación

Estos gráficos suelen ser en su mayoría estáticos y únicos. Normalmente deberán de tener una alta calidad de imagen, y es recomendable que haya una leyenda que explique las variables para la total comprensión del oyente.

Estos gráficos deben de ofrecer una visión convincente de los resultados a los que se ha llegado, e ir fuertemente correlacionados al resto de la presentación para que el oyente no se pierda.

- Gráficos para la exploración

Estos gráficos se usan para la búsqueda rápida de resultados. En ellos prima la rapidez con la que se obtengan para ver los resultados, y no tanto que sean perfectamente precisos ni de alta calidad. Si el analista comprende en profundidad las variables, no es necesario que incluyan leyenda ni ningún tipo de elemento aclaratorio, puesto que su ciclo de vida será extremadamente corto.

Es importante tener también en cuenta que no todos los tipos de gráficos, ni los colores, ni incluso las dimensiones son elementos que se tengan que elegir al azar. Hay una serie de parámetros y decisiones que se deben de tener siempre en cuenta a la hora de hacer un gráfico, y son las siguientes:



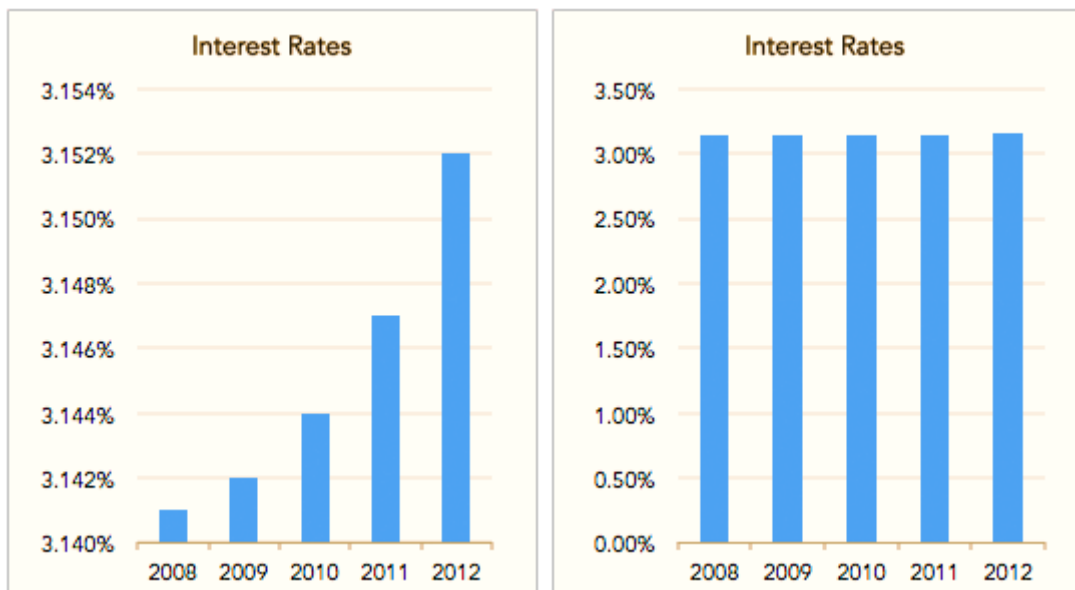
- 1) Escalas
- 2) Ordenamiento
- 3) Color
- 4) Tamaño y Ratio
- 5) Tipo de gráfico

A continuación, se explican detalladamente cada uno de estos elementos:

#### - Escalas

Escoger la escala a utilizar en un gráfico cuando las variables son categóricas es una decisión complicada. Tanto, que incluso software de alta calidad a veces falla a la hora de elegir la escala de visualización, por lo que esta tarea, aunque sea a veces fácil de discernir, no es siempre sencilla. En caso de que las variables sean continuas, la decisión se complica exponencialmente, puesto que habrá que elegir además unas ciertas divisiones y finalizaciones.

Una práctica muy extendida es la de coger como escala los extremos de los datos, es decir, el mínimo que obtienen y el máximo, pero esto es una práctica incorrecta, puesto que algunos puntos, barras o líneas estarán sobre los ejes y no se apreciarán correctamente. Además, si una serie de elementos no tienen como mínimo el cero, se puede estar incurriendo en un caso de falseo visual de los datos al no poner un mínimo absoluto como referencia. Todos estos elementos se pueden ver en la figura a continuación:



2-20. Mismos datos, diferentes escalas

De esta manera, a la hora de escoger escalas es importante no engañar con los datos, puesto que como se puede observar en la figura anterior, en la presentación de la izquierda parece un aumento de los tipos de interés extremo, cuando puesto en perspectiva respecto al cero podemos ver que el aumento no es para nada preocupante.

- Ordenamiento

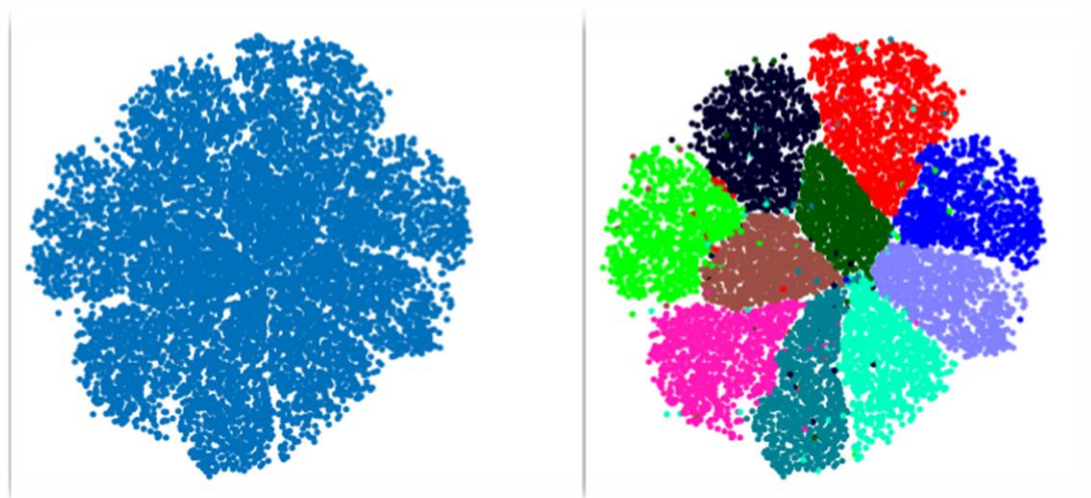
Cuando más de una variable es representada, caso que se da en la gran mayoría de las ocasiones, la forma en la que se ordenen estas variables puede marcar una gran diferencia. Debido a esto, y para no alterar los datos, se suelen ordenar los mismos mediante criterio alfabético, o si corresponde, mediante criterio geográfico o agrupaciones relevantes.

- Color

El color, aunque parezca una banalidad, es uno de los elementos más importantes a decidir. Y esto es porque no todos los colores se perciben igual en el cerebro, no todos son igualmente fáciles de ver, especialmente en conjunción con otros (como, por ejemplo, el de fondo), y también hay que tener en cuenta que puede haber personas con enfermedades relacionadas con la percepción del color, como el daltonismo.

Además, se ha demostrado científicamente que los gráficos cuyos elementos más importantes son el color y el tamaño son de los más complicados de interpretar por la mente humana, especialmente si los tamaños o los colores son ciertamente similares entre sí.

Ante todos estos problemas, el color destaca por que, si se consigue acertar en su elección, puede ser un elemento diferencial a la hora de crear gráficos. Esto destaca especialmente en clusters, donde todos los datos suelen ser representados con la misma forma y tamaño, y el color se plantea como elemento diferencial entre el entendimiento y la ignorancia del significado del mismo. Esto se puede apreciar en la figura siguiente:



**2-21. Clustering Colores**

En resumen, a la hora de crear gráficos uno de los elementos más a tener en cuenta es el color, puesto que puede marcar la diferencia a la hora de entender o no entender los datos. Además del color, habrá que tener en cuenta el tamaño del elemento representado, puesto que si dos elementos distintos poseen el mismo color y un tamaño similar muchos oyentes interpretarán que pertenecen al mismo grupo o clase.

- Tamaño y Ratio

Los gráficos deben de ser lo suficientemente grandes como para que el oyente pueda observarlos sin problema ninguno, a la vez que no deben pasarse de grandes debido a que se desaprovecha espacio.

Si se quieren añadir marcos a los gráficos, hay que tener en cuenta de que también ocupan espacio de la visualización, por lo que la recomendación general suele ser añadirlos en caso de querer hacer una separación de las representaciones.

Finalmente, el ratio es quizás el elemento más complicado de este apartado, puesto que tiene un gran impacto en cómo se visualizará. De este modo, un aumento del eje y conllevará una dramatización de cualquier cambio, mientras que una elongación del eje x muestra un cambio más gradual en las series temporales. Sea como fuere, el ratio es un parámetro muy delicado y se debe de actuar con cautela a la hora de elegirlo.

- Tipo de gráfico

Una vez se han tenido las recomendaciones anteriores en cuenta se tiene que elegir el gráfico que representará la información deseada. A continuación, se explican algunos de los más utilizados:

- 1) Gráfico de Barras

El gráfico de barras es el gráfico más conocido y usado mundialmente. Este destaca por representar los valores de los datos a través de longitudes de barras, dando igual la anchura de las mismas. Puede ser mejorado mediante la aplicación de diferentes colores a cada barra, o aplicando colores según variables que pertenezcan a un grupo superior.

Este gráfico es ideal para cuando se quieren representar datos con variables discretas, como por ejemplo con frecuencias en las que se da un determinado evento. Es muy importante tener en cuenta en este gráfico el parámetro del ratio, puesto que este es de los gráficos más sensibles al cambio si las barras se sitúan en posición vertical y se amplía la componente y.

- 2) Gráfico de sectores

El gráfico de sectores es un gráfico con forma circular donde los datos se dividen proporcionalmente en formas triangulares, por lo que recibe informalmente nombres como “gráfico de quesitos” o “gráfico de tarta”.

El gráfico de sectores tiene una fuerte oposición entre los expertos, debido a que incumple el parámetro del color-tamaño que se expuso previamente. Dos franjas separadas pueden tener tamaños distintos pero similares, y no se apreciará bien esta diferencia. Debido a ello, para subsanarlo, normalmente se añade el porcentaje o el valor del elemento que representa cada división.

Estos gráficos son esencialmente usados a la hora de comparar porcentajes o tamaños sin querer obtener demasiada precisión, sino pudiendo ver similitudes y grandes diferencias.

### 3) Histograma

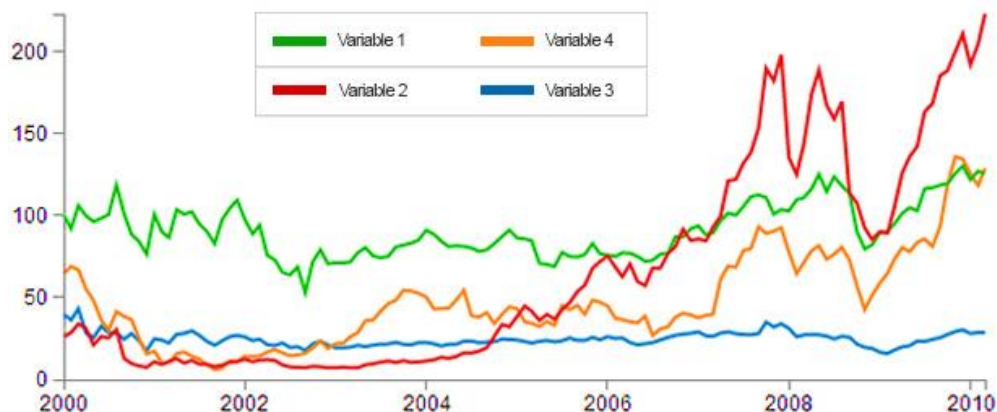
Usado ampliamente en estadística, el histograma es un gráfico de barras que se usa para representar mediante las mismas una serie de valores, pero al contrario que en el gráfico de barras no presenta la frecuencia de una categoría, sino de un intervalo de valores.

Uno de los elementos que se deben de evitar en los histogramas es el representar intervalos de diferente tamaño, a excepción de los que van hasta infinito en los extremos. Esto es debido a que puede llevar a confusión y es una alteración de la representación de los datos y del tamaño de los grupos.

A veces, este gráfico suele incorporar una línea ascendente sobre él que indica la frecuencia acumulada en cada intervalo.

### 4) Gráfico de líneas

Los gráficos de líneas son representaciones que sirven para observar la evolución de una o más variables conforme al paso del tiempo. Para hacer esto, en el eje horizontal se muestran una serie de fechas, y en el eje vertical se muestran los valores que pueden tomar las variables. Con esta disposición, se puntúa cada uno de los valores de cada variable para cada fecha, y se juntan con una línea los puntos que pertenecen a cada variable.



2-22. Gráfico de Líneas

Este gráfico tiene como principal ventaja su facilidad de interpretación, puesto que muestra a lo largo del tiempo las mejoras y empeoramientos de cada variable. Además, si la última fecha pertenece a la actualidad, se puede observar qué variables están por encima del resto para el valor elegido en el eje vertical.

Una de sus principales desventajas radica en la posibilidad de que haya demasiadas variables. En ese caso, el gráfico puede no quedar claro a simple vista para el lector u oyente.

### 5) Gráfico de dispersión o scatterplot

El scatterplot es un gráfico conocido como representador de todos los elementos de un dataset en los ejes x e y.

De esta forma, el scatterplot es un gráfico muy interesante para la búsqueda de correlaciones entre variables, puesto que se puede ver para dos dimensiones la tendencia que poseen, en caso de que posean alguna. Sobre este gráfico suele incluirse la recta de correlación, junto con la variable  $R^2$  para mostrar el índice de correlación, que fluctuará entre 0 y 1.

También se puede observar el nivel de dispersión de los datos con este gráfico.

Una de las principales limitaciones que tiene esta representación es la dificultad de interpretación en caso de llevarlo a tres dimensiones, y la imposibilidad de visualización en caso de llegar a 4 dimensiones o más. Debido a esto, este gráfico suele estar limitado a la búsqueda de correlaciones entre variables de dos a dos.

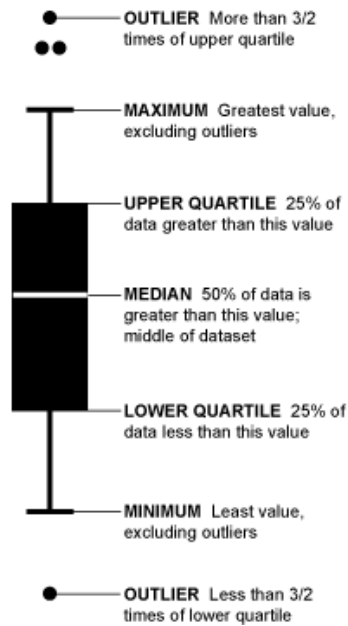
### 6) Gráfico de cajas o boxplot

El gráfico de cajas, también conocido como boxplot, es otro sistema de visualización usado muy frecuentemente en ámbitos estadísticos. Este gráfico permite hacer una representación de los estadísticos principales de cada variable, de tal manera que se pueden llegar a representar:

- Media
- Mediana
- Cuartiles
- Bigotes

Especial mención merece este último elemento, pues consiste en la representación del valor al que pueden llegar los datos de dicha variable para no considerarse outliers, tanto por encima de la media como por debajo.

Esta representación es realmente útil a la hora de comparar valores y dispersión de variables, y suele ser bastante usada en los análisis exploratorios.



2-23. Explicación de BoxPlot

La desventaja que posee esta representación es que la persona a la que se le explica debe de tener una formación estadística para poder entenderlo y comprenderlo en profundidad, por lo que suele ser más un gráfico de exploración que de presentación.

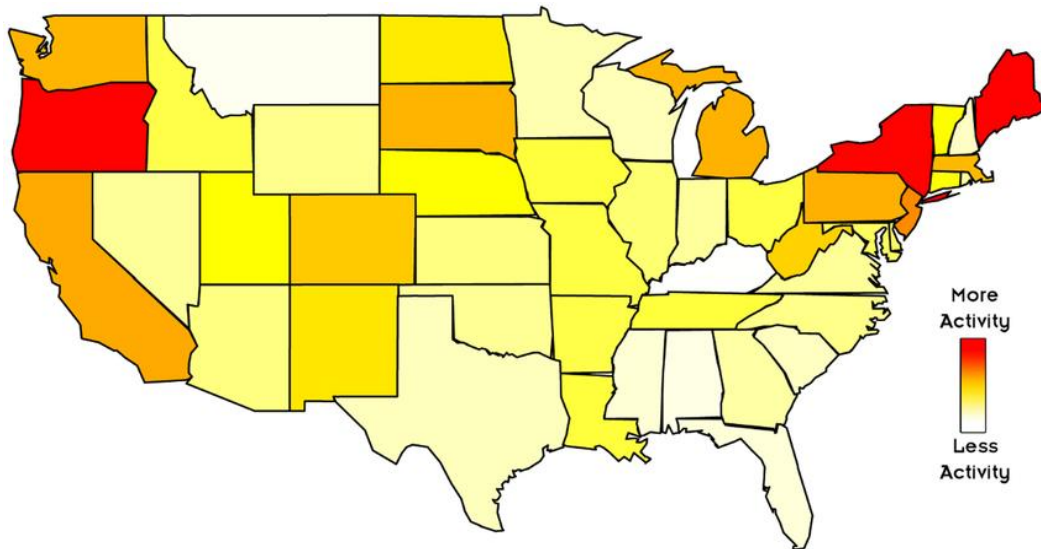
## 7) Cartograma

El cartograma es un tipo de representación de datos basado en mapa, donde cada país o región de interés posee un valor, normalmente representado con escalas de color acompañadas de una leyenda para su aclaración.

Estos mapas suelen ser utilizados en el tratamiento de ciertas variables por regiones, como por ejemplo de epidemias víricas o representaciones relacionadas con la economía.

Una de las ventajas de estos cartogramas es que, además de jugar con el color, se puede trabajar con la forma de las regiones deseadas, de tal forma que se puede distorsionar el tamaño de las mismas para representar una cierta variable. Aunque esta práctica es ampliamente usada, hay que tener cuidado porque muchas veces desvirtuará en exceso el mapa e incluso algunas regiones pueden acabar desaparecidas, por lo que no se verá el color de la variable principal, además de que dificultan el entendimiento inmediato de la representación.

Una de las principales desventajas que tiene este tipo de representaciones es la de que suelen usar escalas de color, y los seres humanos, como se ha comentado anteriormente, somos seres que nos cuesta distinguir entre colores parecidos. De esta forma, dos países con valores similares para una cierta variable serán difícilmente diferenciables. Esta desventaja desaparece si se representa un número de colores pequeño relacionados con valores discretos.



2-24. Ejemplo Cartograma

La conclusión de este apartado de visualización radica en darse cuenta de que la elección de la representación de los datos no es banal, puesto que son el método de exposición de los mismos hacia el oyente o lector interesado. Así, se deben tener en cuenta numerosos factores que pueden dificultar la interpretación de los gráficos, como el ratio o el color, y elegir correctamente la representación para poder alcanzar un nivel de entendimiento y precisión adecuados, ya que no sirve de nada obtener conclusiones importantes si no se transmiten de una manera correcta.





## 3. Capítulo 3: REST

---

Roy Fielding, uno de los padres de la arquitectura HTTP, inventó esta arquitectura en el año 2000, en su libro “Architectural Styles and the Design of Network Based Software Architectures”. Actualmente, REST es ampliamente usado en la construcción de todo tipo de aplicaciones.

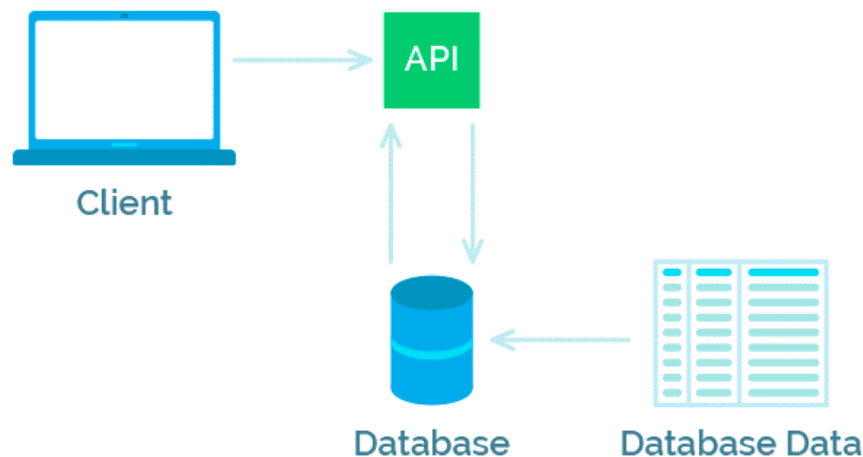
Es muy raro llegar a ver un proyecto donde no haya al menos una API REST para el intercambio de datos con el servidor. Es una arquitectura del lado del servidor, y se podría definir como una interfaz, totalmente compatible dentro del protocolo HTTP, que se utiliza para obtener, modificar, eliminar o introducir datos, o incluso hacer operaciones con dichos datos. Normalmente, REST trabaja con dos tipos de estructuras de datos: XML y JSON.

REST tiene una serie de características que lo hacen único, que se pueden ver a continuación:

- 1) Sin Estado: No guarda el estado ni el contexto de la operación, sino que cada petición lleva unas cabeceras y un cuerpo con toda la información necesaria para hacer la petición.
- 2) Operaciones HTTP: Como se dijo anteriormente, REST se basa en el protocolo HTTP, de tal manera que las operaciones que se pueden hacer con esta arquitectura son las mismas:
  - a. POST: Creación
  - b. GET: Obtención
  - c. PUT: Edición
  - d. DELETE: Eliminación

Aunque existan las cuatro, las dos más utilizadas son GET y POST, debido a que con esas dos suele ser suficiente para hacer las cuatro operaciones. En caso de que pueda haber sobrecarga de estas dos peticiones, es una buena práctica usar las cuatro.

- 3) URI como entrada: Para hacer llamadas a una API REST, es necesario siempre llamar a una URI. Cada recurso (operación) de la API REST está identificada con una URI, que debe ser única dentro del conjunto de recursos de la misma operación HTTP, y mediante el envío de la cabecera y el cuerpo a la misma se efectuarán las operaciones.



**3-1. Petición GET en API REST**

A continuación, se explicará el funcionamiento básico de la estructura REST:

Como se puede apreciar en la imagen 3-1, el cliente envía una petición al servidor que contiene la API REST. Esta petición, tal como se explicó anteriormente, puede ser enviada o en XML o en JSON. Una vez recibida la petición, que en este caso es de GET, la API se pone en contacto con la base de datos mediante el protocolo HTTP, que puede estar en el mismo servidor u en otro. En ese momento, la base de datos ejecuta la petición deseada, y devuelve el resultado a la API REST. La API lo parsea, y lo devuelve en formato XML o JSON al cliente, donde el frontend lo modificará para enseñárselo al usuario de una manera “amigable”.

En este trabajo, la base de datos a la que se va a hacer la petición es una de las bases de datos No SQL más famosas del mundo: MongoDB. Para la creación de la API REST, se usará un framework basado en JavaScript bastante moderno, llamado NodeJS. A continuación, estas dos tecnologías serán desarrolladas en mayor profundidad para la comprensión de las mismas.

### 3.1 MongoDB

El nombre de MongoDB proviene del término anglosajón “humongous”, que tiene como traducción “enorme”. Este término viene a referirse a las grandes cantidades de datos que esta base de datos puede almacenar.

MongoDB es una base de datos No SQL, orientada a documentos. Esto significa que los datos se guardan en una serie de “líneas” consistentes en un conjunto de clave – valor, conocidas como documentos. Los documentos se guardan en una serie de “tablas” de documentos similares, que se conocen como colecciones. Una base de datos será un conjunto de una o más colecciones.

Debido a estas características, las bases de datos No SQL, y entre ellas MongoDB, son unas bases de datos muy cómodas a la hora de hacer cambios, puesto que no es necesario remodelar la base de datos entera para añadir el soporte de nuevos campos.

Otras características muy interesantes de MongoDB son la escalabilidad y la alta disponibilidad de la que dispone. Es fácilmente auto escalable y replicable en servidor, de tal manera que la tolerancia a fallos es muy alta.



3-2. Logo de MongoDB

Algunas de sus características más importantes son:

- 1) Esquemas dinámicos: Como se ha comentado anteriormente, MongoDB es muy cómoda a la hora de hacer cambios, puesto que no afecta a la estructura de la base de datos ni al resto de los datos. Debido a ello, es una base de datos ideal cuando los requerimientos de una aplicación pueden cambiar.
- 2) Inteligencia operacional: MongoDB posee un sistema interno de agregación y Map Reduce que permite obtener conocimiento en tiempo real para las aplicaciones, por lo que en algunos aspectos mejora a otras tecnologías como Hadoop o las aplicaciones antiguas y tradicionales de Business Intelligence.
- 3) Flexibilidad en la implementación: MongoDB fue concebida para ser usada en arquitecturas Cloud especialmente. Las peticiones a la base de datos son robustas y aseguran un buen rendimiento.
- 4) Escalado simple: Otra de las características para las que se concibió MongoDB es para ser escalada en múltiples servidores sin demasiadas trabas. Así, si los datos crecen las organizaciones pueden añadir más nodos a otros clusters, y MongoDB balanceará los datos de forma nativa entre todos ellos.

¿Cuándo se debe usar MongoDB?

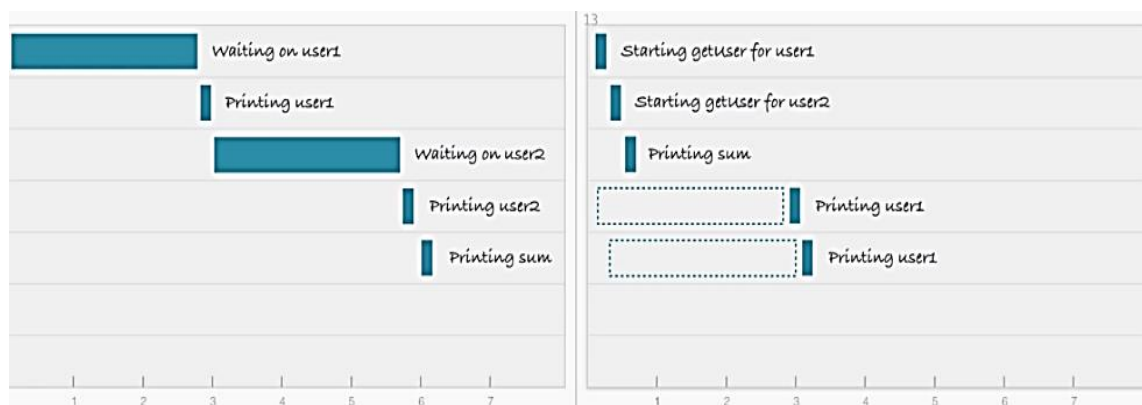
- 1) Cuando se quiera tener analíticas en tiempo real, o se tenga esquemas complejos.
- 2) Cuando se necesite una latencia pequeña, alta disponibilidad y posibilidad de escalado.
- 3) Cuando se quieran poder hacer pequeños pero importantes cambios en la base de datos sin necesidad de cambiar toda su estructura.

## 3.2 NodeJS

Se podría definir NodeJS como un entorno de JavaScript, lo que significa que incluye todos los elementos necesarios para poder ejecutar un programa escrito en dicho lenguaje.

NodeJS fue inventado cuando se quiso cambiar la filosofía de ejecución de JavaScript. De este modo, NodeJS se concibió para poderse ejecutar en una máquina como una aplicación en sí en vez de en un buscador web, como ocurre con JavaScript.

Como se comentó anteriormente, una API REST tiene que ser un programa conductor de entradas y salidas. NodeJS cumple dicha función, puesto que es capaz de cumplir las peticiones HTTP que se le hagan con órdenes tanto de input como de output. Uno de los elementos más importantes que posee NodeJS respecto a otros lenguajes de APIs es el hecho de que su entrada salida es no bloqueante a pesar de ser de procesamiento mononúcleo. Esto lo que significa es que, aunque dos usuarios hagan petición, las dos peticiones se pueden iniciar al mismo tiempo, haciendo que una petición no tenga que esperar a la finalización de una petición anterior. Esto se puede observar claramente en la imagen siguiente:



3-3. Entrada/Salida Bloqueante Vs Entrada Salida no Bloqueante

Como se puede apreciar, en la imagen de la izquierda hasta que el usuario 1 no ha recibido la respuesta a su petición el usuario 2 no inicia su petición al sistema. En cambio, en la imagen de la derecha, la entrada/salida no es bloqueante, lo que significa que ante dos peticiones muy juntas, el sistema las ejecutará "en paralelo" y según se vayan teniendo las salidas se irán entregando a los dispositivos correspondientes. Esta es una de las principales ventajas que aporta NodeJS sobre otros lenguajes y frameworks de desarrollo backend.

Finalmente, es importante destacar el papel que hace NPM en el desarrollo de las aplicaciones NodeJS.

NPM, siglas de "Node Package Manager", es un manejador de un conjunto de librerías creadas por la comunidad que son capaces de resolver la mayoría de los problemas que se pueden presentar a la hora de desarrollar una aplicación NodeJS. Los comandos que se ejecuten con este manejador siempre empezarán por "npm", seguido de un verbo que indicará la acción a ejecutar. Posteriormente, podrán ir una serie de parámetros.

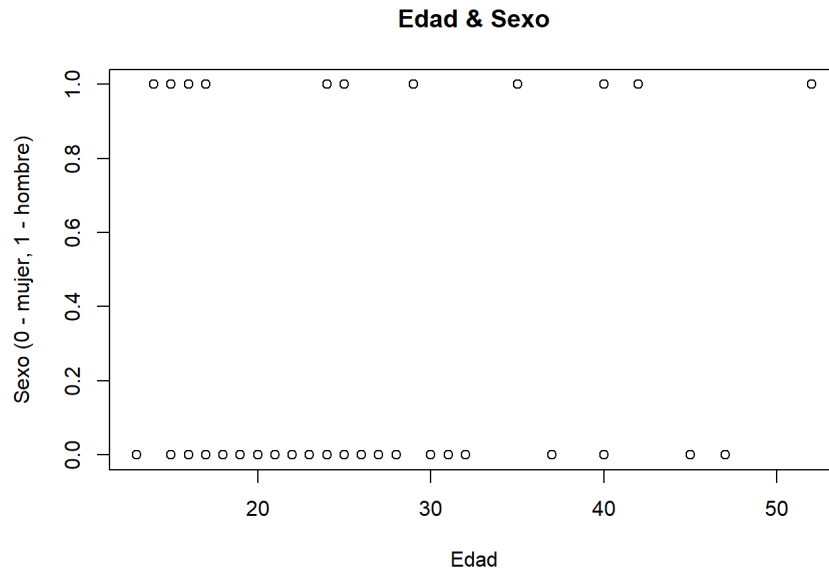
## 4.1 Resultados Obtenidos

- Análisis Exploratorio

A word cloud of names in various colors and sizes. The names are arranged in a circular pattern around the center. The largest and most prominent name is 'Raquel' in black. Other large names include 'Clara' in green, 'Marta' in green, 'Blanca' in green, 'Sara' in red, 'Noelia' in red, 'María' in red, 'Andrea' in red, 'Paula' in red, 'Mar' in red, 'Soraya' in purple, 'Pablo' in purple, 'Juan' in purple, 'Eugenia' in purple, 'Teresa' in purple, 'Yolanda' in purple, 'Henar' in purple, 'Julia' in purple, 'delio' in purple, 'Ana' in purple, 'Alicia' in purple, 'Adrian' in purple, 'Daniel' in purple, 'Irene' in purple, 'Carmen' in purple, 'Miriam' in purple, 'Marina' in purple, 'Esther' in purple, 'Vega' in purple, 'Victor' in purple, 'Diego' in purple, 'Marie' in purple, 'Angela' in purple, 'Cati' in purple, 'Borja' in purple, 'Jorge' in purple, 'Laura' in purple, 'Izan' in purple, 'Tatiana' in purple, 'Amparo' in purple, 'Lucia' in purple, 'Roberto' in purple, 'Gabriela' in purple, 'Mónica' in purple, 'Silvia' in purple, and 'Julio' in purple. The names are in various orientations, some horizontal and some vertical.

Como se puede analizar, el nombre más común entre los pacientes analizados es el de Raquel, mientras que otros nombres como Clara o Marta también son bastante destacados.

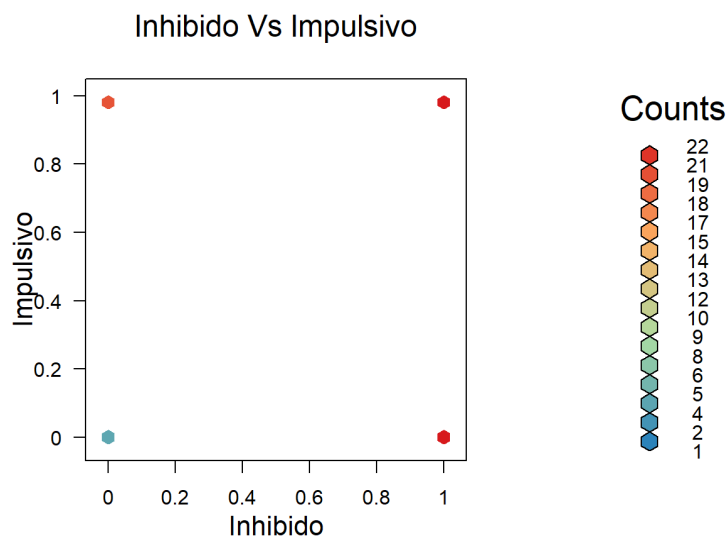
Posteriormente, se ha hecho un análisis de la relación entre la edad y el sexo de los pacientes. Al tener poca muestra se ha considerado que la creación de un histograma sería menos fácilmente apreciable que un scatterplot, cuyo resultado se muestra a continuación:



**4-2. Scatterplot Edad - Sexo**

Como se puede apreciar en la figura obtenida, las mujeres que acuden a la consulta suelen ser más jóvenes, mientras que los hombres tienen un rango mucho más esparcido en la edad.

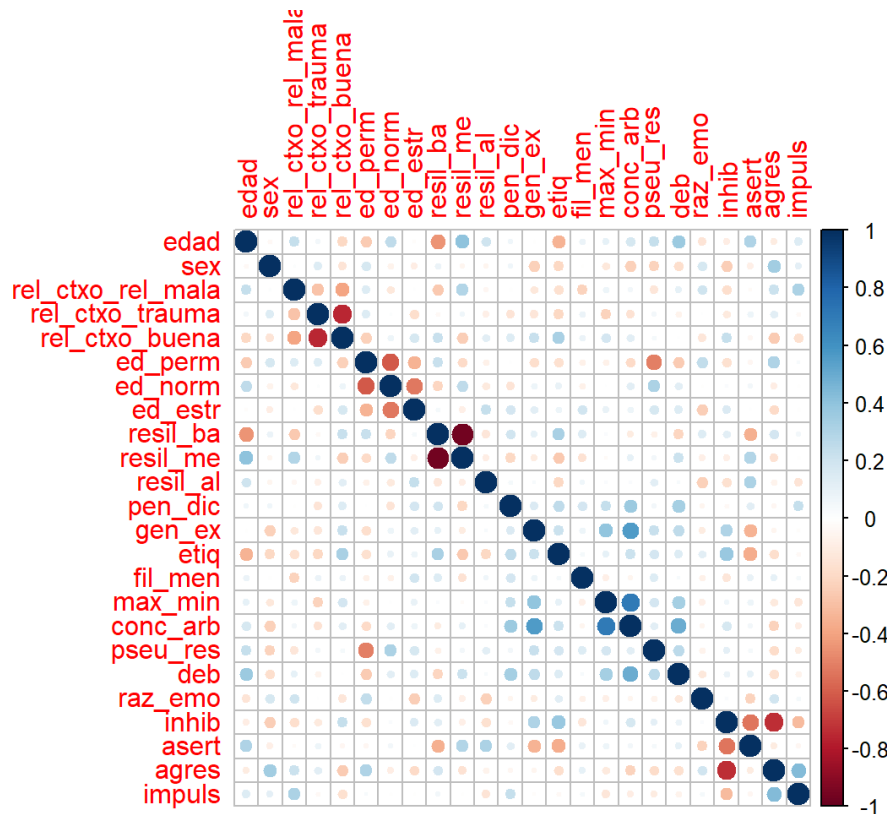
Posteriormente se analizaron varias relaciones entre variables, también con scatterplots basados en colores para comprobar la cantidad de individuos residentes en cada caso. Un ejemplo de este estudio es el siguiente, que analiza la inhibición y la impulsividad:



**4-3. Inhibido Vs Impulsivo**

Como se puede observar en la figura 4-3, hay numerosos pacientes que son inhibidos e impulsivos, pero también hay bastantes que son inhibidos y no impulsivos. Si nos fijamos en los pacientes no inhibidos, podemos observar que hay más que son impulsivos que los que no lo son.

Posteriormente, siguiendo una metodología científica y estadística más rigurosa se obtuvo una matriz de correlación con las variables significativas del problema, habiendo quitado variables que no aportan información como el nombre previamente.



4-4. Matriz de correlación

En esta matriz de correlación se pueden observar varias correlaciones obvias, como por ejemplo que una relación con el contexto buena es fuertemente inversamente correlacionada con una relación con el contexto de trauma, o que la resiliencia baja y media también tienen una fuerte correlación inversa. También es destacable que la inhibición está inversamente correlacionada con la agresividad.

Posteriormente se pueden observar otras correlaciones menos conocidas, como que la pseudo-resiliencia está inversamente relacionada con una educación permisiva. Esto tiene cierto sentido, porque ante una educación permisiva, donde la persona puede hacer lo que quiera desde niño sin consecuencias, esta persona tendrá menos experiencia en soportar los aspectos negativos o de presión que puede traer la vida, y por lo tanto aparecerá la pseudo-resiliencia.

Respecto a relaciones directas, es interesante destacar la que existe entre las distorsiones cognitivas de las conclusiones arbitrarias y la maximización y minimización. Esta relación afirma

que las personas que tienden a exagerar lo malo y minimizar lo bueno, suelen también obtener conclusiones falsas sin argumentos fehacientes.

También existe relación directa entre las conclusiones arbitrarias y la generalización excesiva. Esta relación tiene mucho sentido debido a que si una persona saca conclusiones arbitrarias, tiene sentido que también generalice y saque conclusiones que de cosas que le han ocurrido en el pasado, aunque sea solo una vez, le vuelvan a ocurrir. Como se puede apreciar, en ambas distorsiones cognitivas se tiene poca o nula prueba de lo que puede pasar, y aun así se dicta lo que se cree que va a pasar con total seguridad.

También la agresividad y la impulsividad tienen una cierta correlación, la cual, según estudios recientes como el hecho por Vigil-Colet, Morales-Vives y Tous en 2008, afirman que una gran cantidad de jóvenes agresivos son impulsivos, pero que con el paso del tiempo, a la llegada de la edad adulta esto cambia, ya que los comportamientos son más controlados. Si se vuelve a mirar la figura 4-2, se puede observar como la gran mayoría de los pacientes que se tienen en el dataset son pacientes jóvenes, por lo que esta correlación está llena de sentido.

Siguiendo con la estadística, se obtuvieron los siguientes resultados estadísticos del dataset:

edad	sex	rel_ctxo_rel_mala	rel_ctxo_trauma
Min. :13.00	Min. :0.000	Min. :0.0000	Min. :0.0000
1st Qu.:19.50	1st Qu.:0.000	1st Qu.:0.0000	1st Qu.:0.0000
Median :25.00	Median :0.000	Median :0.0000	Median :0.0000
Mean :26.46	Mean :0.209	Mean :0.1343	Mean :0.3582
3rd Qu.:30.50	3rd Qu.:0.000	3rd Qu.:0.0000	3rd Qu.:1.0000
Max. :52.00	Max. :1.000	Max. :1.0000	Max. :1.0000
rel_ctxo_buena	ed_perm	ed_norm	ed_estr
Min. :0.0000	Min. :0.0000	Min. :0.0000	Min. :0.0000
1st Qu.:0.0000	1st Qu.:0.0000	1st Qu.:0.0000	1st Qu.:0.0000
Median :1.0000	Median :0.0000	Median :0.0000	Median :0.0000
Mean :0.5075	Mean :0.2836	Mean :0.4925	Mean :0.2239
3rd Qu.:1.0000	3rd Qu.:1.0000	3rd Qu.:1.0000	3rd Qu.:0.0000
Max. :1.0000	Max. :1.0000	Max. :1.0000	Max. :1.0000
resil_ba	resil_me	resil_al	pen_dic
Min. :0.0000	Min. :0.0000	Min. :0.00000	Min. :0.0000
1st Qu.:0.0000	1st Qu.:0.0000	1st Qu.:0.00000	1st Qu.:1.0000
Median :1.0000	Median :0.0000	Median :0.00000	Median :1.0000
Mean :0.5672	Mean :0.4179	Mean :0.01493	Mean :0.8955
3rd Qu.:1.0000	3rd Qu.:1.0000	3rd Qu.:0.00000	3rd Qu.:1.0000
Max. :1.0000	Max. :1.0000	Max. :1.00000	Max. :1.0000
gen_ex	etiq	fil_men	max_min
Min. :0.0000	Min. :0.0000	Min. :0.000	Min. :0.0000
1st Qu.:1.0000	1st Qu.:0.5000	1st Qu.:1.000	1st Qu.:1.0000
Median :1.0000	Median :1.0000	Median :1.000	Median :1.0000
Mean :0.9552	Mean :0.7463	Mean :0.791	Mean :0.9701
3rd Qu.:1.0000	3rd Qu.:1.0000	3rd Qu.:1.000	3rd Qu.:1.0000
Max. :1.0000	Max. :1.0000	Max. :1.000	Max. :1.0000
conc_arb	pseu_res	deb	raz_emo
Min. :0.0000	Min. :0.0000	Min. :0.0000	Min. :0.000
1st Qu.:1.0000	1st Qu.:0.0000	1st Qu.:1.0000	1st Qu.:1.000
Median :1.0000	Median :1.0000	Median :1.0000	Median :1.000
Mean :0.9851	Mean :0.5075	Mean :0.9403	Mean :0.791
3rd Qu.:1.0000	3rd Qu.:1.0000	3rd Qu.:1.0000	3rd Qu.:1.000
Max. :1.0000	Max. :1.0000	Max. :1.0000	Max. :1.000
inhib	asert	agres	impuls
Min. :0.0000	Min. :0.0000	Min. :0.0000	Min. :0.0000
1st Qu.:0.0000	1st Qu.:0.0000	1st Qu.:0.0000	1st Qu.:0.0000
Median :1.0000	Median :0.0000	Median :0.0000	Median :1.0000
Mean :0.6567	Mean :0.1343	Mean :0.2239	Mean :0.6119
3rd Qu.:1.0000	3rd Qu.:0.0000	3rd Qu.:0.0000	3rd Qu.:1.0000
Max. :1.0000	Max. :1.0000	Max. :1.0000	Max. :1.0000

4-5. Estadísticos principales del Dataset



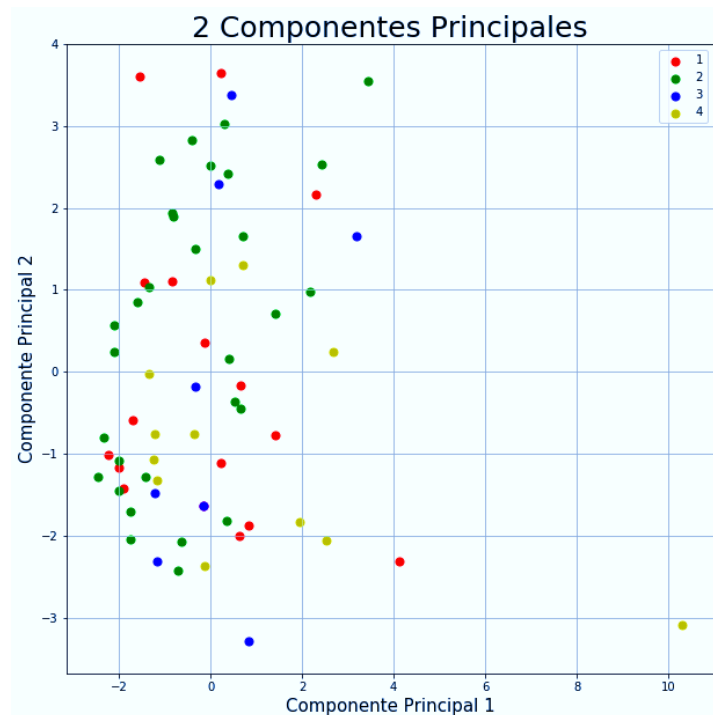
Se pueden hacer algunas valoraciones con estos datos:

- 1) Edad: La media de la muestra que acude a consulta tiene 26 años y medio, estando el primer cuartil en 19,5 años y el tercer cuartil en 30,5 años. Esto quiere decir que el 50% de los pacientes son veinteañeros. También se puede observar que la edad mínima de un paciente en la muestra es de 13 años.
- 2) Sexo: A pesar de ser una variable categórica, se pueden sacar conclusiones ya que es binaria a partir de la media y los cuartiles. Esta variable nos indica que hay muchas más mujeres que hombres en la muestra (media de 0,20 y tercer cuartil de 0).
- 3) Relación con el contexto: Respecto a esta variable es interesante fijarse en la media, ya que como la anterior, es binaria, donde se puede observar que la mayoría de gente tiene una relación con el contexto buena, seguido de trauma y finalmente de mala.
- 4) Educación: La mayor parte de la gente posee una educación normal, siendo la permisiva la siguiente.
- 5) Resiliencia: Se puede observar que en la única en la que está la mediana con un valor distinto a cero es en la resiliencia baja, lo cual nos indica que esa variable está “activada” numerosas veces. Además, esto también se puede observar fácilmente por la media y el tercer cuartil.
- 6) Distorsiones cognitivas: Se aprecia que las medias son bastante cercanas a 1, por lo que se aprecia que están muy presentes en los pacientes en su mayoría. Es interesante también observar que todas, a excepción de la pseudo resiliencia ya obtienen el valor 1 en el primer cuartil.
- 7) Habilidades sociales: Observando la media y la mediana, la mayoría de personas son inhibidas, siguiendo por la agresividad y acabando por la asertividad.
- 8) Impulsividad: La media reside en 0.61, y la mediana en 1 (lógico debido a que es una variable categórica de 2 categorías), lo que quiere decir que hay una mayoría de personas impulsivas en la consulta.

#### - Métodos de Inteligencia Artificial

Se han utilizado diversos métodos de inteligencia artificial para la clasificación de los pacientes en sus respectivos grupos, obteniendo resultados no extremadamente satisfactorios con el dataset original, aunque bastante diversos en su calidad. Posteriormente, tras la igualación de los grupos se han obtenido resultados mucho más satisfactorios.

Es importante destacar la distribución de los pacientes en una gráfica, para ver si son fácilmente diferenciables o no. Para esto, se ha reducido el problema a las dos componentes principales (es decir, las dos dimensiones más significativas) y se ha obtenido una gráfica donde



**4-6. Distribución Pacientes Componentes Principales**

se pueden ver los pacientes diferenciados por colores según el trastorno.

Como se puede observar, bajo las dos componentes que más información aportan al problema la clusterización de los pacientes se antoja como una tarea difícilísima, puesto que están realmente mezclados entre sí y bajo estas dimensiones es muy complicado separarlos. Debido a esto, es probable que muchos métodos acaben teniendo problemas a la hora de clasificar, en especial los métodos basados en distancias, ya que al aumentar el número de componentes para intentar visualizar más diferencias se pueden encontrar con la maldición de la dimensionalidad.

Veamos a ver los resultados obtenidos sobre el dataset original:

- Redes Neuronales

Las redes neuronales fueron desarrolladas en lenguaje R, donde se probaron numerosos parámetros para afinarlas. La siguiente tabla muestra los mejores resultados obtenidos con los mejores parámetros para dicha configuración, tras 50 entrenamientos y tests:

Tabla 4-1. Resultado de Redes Neuronales

Neuronas	1	1	2	2	<b>3</b>	3	3	3	5	5	5	5
Softmax	No	Sí	No	Sí	<b>No</b>	Sí	No	Sí	No	Sí	No	Sí
Decay	-	-	-	-	-	-	0,2	0,03	-	-	0,1	0,05
Resultado Test	66,6%	66,6%	66,6%	75%	<b>75%</b>	75%	75%	66,6%	75%	66,6%	75%	50%

Como se comentó en la parte teórica, estos resultados no son totalmente correctos, pues hay que tener en cuenta dos elementos:

- 1) Las personas que tienen un TOC tienen un trastorno de ansiedad, por lo que una clasificación del grupo 1 en el 2 sería técnicamente correcta, y esto no se está teniendo en cuenta a la hora de valorar las redes neuronales.
- 2) Softmax no tiene en cuenta que un elemento pueda pertenecer a varias categorías, por lo que sus resultados deben de ser rechazados en este estudio.

Debido a esto, y de un modo estricto (sin tener en cuenta la condición multigrupo del TOC), la configuración de perceptrón multicapa idónea para este problema será la de 3 neuronas sin softmax ni decay, debido a que con menor número de neuronas (y por lo tanto menor overfitting) se ha obtenido un resultado máximo, con un valor de acierto del 75%.

#### ○ KNN

Este método, al igual que todos los siguientes, han sido desarrollados en Python 3, usando el paquete de inteligencia artificial del mismo: Scikit Learn, también conocido como SKLearn.

Con K Nearest Neighbors se han obtenido resultados no tan positivos como con las redes neuronales. Para evitar la gran ineficiencia de ir probando los métodos cambiando pequeños parámetros, se ha formado una malla de parámetros para que el algoritmo determine cuál es la óptima, y así poder trabajar con los mejores. También se debe remarcar que se ha aplicado una cross-validation de 3 folds.

Ante los resultados obtenidos, el algoritmo determinó que los mejores parámetros se daban usando la distancia de Manhattan y con un K igual a 12.

La puntuación máxima obtenida, tanto en entrenamiento como en test es del 50%, obteniendo la siguiente matriz de confusión:

**Tabla 4-2. Matriz Confusión Test KNN**

2	1	0	1	G1
1	7	0	0	G2
0	1	0	0	G3
3	1	0	0	G4
G1	G2	G3	G4	Predicción\Real

Como se puede apreciar en la tabla, de los 18 pacientes a clasificar 9 los ha clasificado correctamente en los grupos 1 y 2 (TOC y ansiedad), confundiendo 1 paciente de TOC por ansiedad, lo que elevaría la cifra de acierto hasta el **55%**.

También se puede observar que ha clasificado muy mal los pacientes correspondientes a los grupos 3 y 4, no habiendo acertado ninguno. Esto se debe a la falta de muestras recogidas de estos grupos (en parte por la escasez de los pacientes con estos cuadros en el gabinete en el momento de la entrevista), por lo que se supone que el entrenamiento ha sido poco fiable hacia estos grupos con su correspondiente consecuencia en el test.

○ Random Forest

Con el método Random Forest se ha seguido una estrategia similar. Se ha creado una malla con parámetros para el entrenamiento del algoritmo, se ha aplicado una cross-validation de 5 folds y se ha obtenido el mejor modelo posible, con el que se han obtenido los siguientes resultados:

Parámetros óptimos:

- Criterio de aceptación: Gini
- Profundidad máxima del árbol: 5 unidades
- Máximas Características: Automático
- Número de estimadores: 100

Con estos parámetros se ha conseguido un acierto de entrenamiento del 92%, mientras que el acierto de test se ha quedado en un 47,1%.

La matriz de confusión de este método se queda de la forma siguiente:

**Tabla 4-3. Matriz Confusión Random Forest**

Predicción \ Real	G1	G2	G3	G4
G1	1	2	0	1
G2	2	6	0	0
G3	0	1	0	0
G4	1	2	0	1

Como se puede observar, especialmente del grupo 2 ha obtenido muy buenos resultados este modelo, especialmente si tenemos en cuenta las dos clasificaciones de TOC en ansiedad, que también se darán como válidas. Respecto a los grupos 3 y 4 vuelve a haber gran cantidad de errores.

El resultado final, teniendo en cuenta estas clasificaciones sería de un **55,5%** de acierto en test.

#### ○ SVM

Support Vector Machines es el siguiente algoritmo que se probó. Continuando con la estrategia de la malla, se probó con los 3 kernels posibles (lineal, RBF y polinómico), y el modelo declaró que los mejores parámetros, para una cross-validation de 3, eran los siguientes:

- Kernel: RBF
- C: 2
- Grado: 2

Hay que tener en cuenta que el grado solo se utiliza si el kernel es polinómico, puesto que sería el grado de la ecuación del polinomio. Al usar kernel RBF, este parámetro no deberá ser tenido en cuenta.

Los resultados obtenidos datan de un 88% de acierto en el entrenamiento, pero de un 41,2% de acierto en el test, siguiendo con la dinámica de los modelos anteriores.

**Tabla 4-4. Matriz Confusión SVM**

Predicción \ Real	G1	G2	G3	G4
G1	1	2	0	1
G2	2	6	0	0
G3	0	1	0	0
G4	1	3	0	0

Como se puede apreciar, la matriz de confusión obtenida en SVM se parece mucho a la matriz de confusión obtenida en Random Forest, a excepción de la predicción del grupo 4 donde obtiene un error más. Debido a esto, los resultados serán similares.

Teniendo en cuenta las dos clasificaciones correctas hacia la relación entre TOC y Ansiedad, se puede afirmar que el resultado final es de un **50%** de acierto, habiendo acertado 9 de 18 casos.

Ante estos resultados pobres, se tuvo que analizar la situación, llegando a la conclusión de que las clases poco balanceadas podían afectar fuertemente en la obtención de resultados positivos, puesto que las matrices de confusión mostraban resultados extremadamente negativos en los grupos 3 y 4 en la mayoría de los algoritmos. Debido a ello, se nivelaron los grupos 1, 3 y 4 haciendo que estuvieran mucho más cerca del número de observaciones del grupo 2, mediante técnica de resampling.

De este modo, se volvieron a ejecutar los algoritmos con sus respectivas mallas de valores, obteniendo los siguientes resultados:

- KNN Balanceado

El KNN Balanceado obtuvo de la malla los siguientes valores como óptimos:

- Métrica: Manhattan
- K = 1

A pesar de que este K = 1 lleva a overfitting (puntuación de train de 100%), el resultado de test mejoró considerablemente, llegando al **77,7%** de acierto. Es también importante el que se destaque que la cross validation aplicada en este caso ha sido de 4.

La matriz de confusión se muestra a continuación:

**Tabla 4-5. Matriz Confusión KNN Balanceado**

Predicción \ Real	G1	G2	G3	G4
G1	4	0	0	2
G2	0	5	1	2
G3	1	0	8	0
G4	0	0	0	4

Como se puede observar, en todas las clases el acierto es muchísimo mayor que el error, siendo la clase 2 la menos acertada con 5 aciertos de 8, un 62,5%. Es interesante observar como las clases 3 y 4, que anteriormente eran clases donde apenas se acertaba una observación como máximo, actualmente aciertan prácticamente la totalidad de sus observaciones.

Para la interpretación de estos resultados es importante destacar que se ha hecho sobre el conjunto de test que tiene el resample, de tal manera que hay más observaciones que en los anteriores, pero ello no distorsiona el resultado final, que se medirá siempre en porcentaje.

#### ○ Random Forest Balanceado

En esta aplicación de Random Forest se ha vuelto a utilizar un cross validation de 3, y con ello se han obtenido los siguientes valores como óptimos:

- Criterio: Entropía
- Máxima Profundidad del Árbol: 7
- Máximas características: Automático
- Número de estimadores: 50

Con estas características, se han obtenido en entrenamiento un resultado del 100%, mientras que en test el resultado obtenido en este caso es del **77,7%** de acierto.

La matriz de confusión se muestra a continuación:

**Tabla 4-6. Matriz Confusión Random Forest Balanceado**

Predicción \ Real	G1	G2	G3	G4
G1	4	0	0	2
G2	0	5	1	2
G3	1	0	8	0
G4	0	0	0	4

Como se puede apreciar, la matriz de confusión del Random Forest es la misma que la obtenida con KNN, por lo que ambos métodos resultan igual de efectivos a la hora de resolver el problema.

○ SVM Balanceado

En el caso del algoritmo de Support Vector Machines, se ha mantenido la cross validation de 3 para comparar resultados con los no balanceados, y con esta premisa se han obtenido los siguientes valores de la malla como óptimos:

- C = 1
- Grado = 2 (Para Kernel Polinómico)
- Kernel = RBF

La puntuación obtenida es similar a la de los métodos anteriores, donde en este caso el entrenamiento ha tenido un valor del 92,3% de acierto y el test se ha quedado en un **74,1%**.

La matriz de confusión de este SVM se muestra a continuación:

**Tabla 4-7. Matriz Confusión SVM Balanceado**

Predicción \ Real	G1	G2	G3	G4
G1	5	0	0	1
G2	0	4	1	3
G3	1	0	8	0
G4	1	0	0	3

La matriz de confusión en este caso de SVM es menos precisa tanto en el grupo 2 como en el grupo 4, es igual de precisa en el grupo 3 y el único grupo que aumenta la precisión con SVM es el grupo 1.



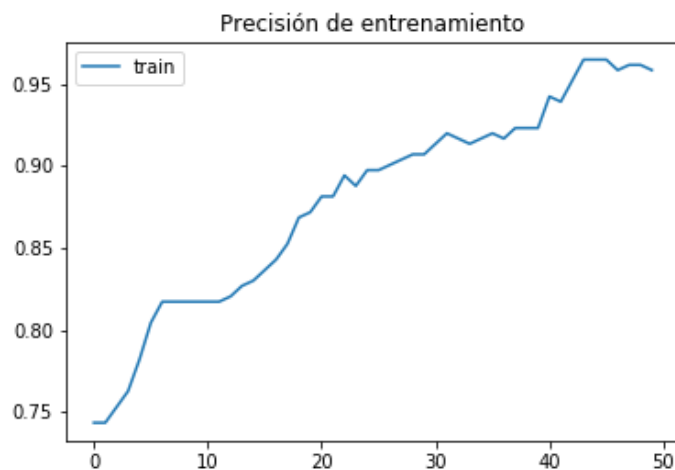
### ○ Deep Learning

El modelo de deep learning ha sido entrenado de una manera supervisada, ya que se poseen las etiquetas del mismo. Además, ya que se conoce que los modelos pueden aprender mejor con clases balanceadas, se ha entrenado con estos datos obteniendo un resultado del **93,5%** de acierto máximo en test, con un 96,4% de acierto en train.

Para mejorar el modelo de deep learning, la estrategia ha sido la creación de 3 capas ocultas con pocas neuronas cada una (6, 5, 5), de tal manera que cada capa pueda “especializarse” en una extracción de características determinada y puedan funcionar mejor de manera conjunta.

Además, para hacer deep learning se ha tenido que codificar como variables dummies los grupos, y así tener 4 neuronas de salida, una por grupo. Junto con ello, hay una capa de entrada de 24 neuronas, con un tamaño de input de 24 dimensiones.

A continuación se expone una imagen del aumento del acierto de entrenamiento en el modelo de deep learning con el paso de las iteraciones:



**4-7. Precisión de Entrenamiento Deep Learning**

Ante los resultados obtenidos, la siguiente tabla muestra la comparación de los resultados de test para su visualización final:

**Tabla 4-8. Mejores Resultados Finales**

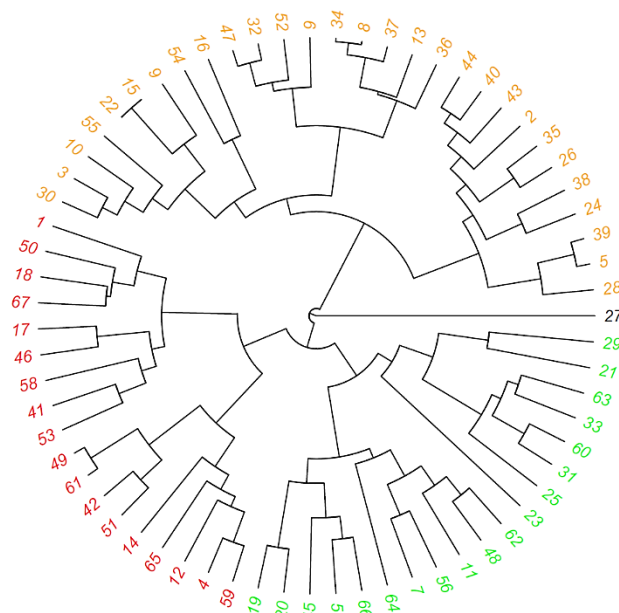
	No Balanceado, CV = 3	Balanceado, CV = 3
KNN	55,5%	77,7%
Random Forest	55,5%	77,7%
SVM	50%	74,1%
	No Balanceado	Balanceado
Redes Neuronales	75%	-
Deep Learning	-	93,5%

Como se puede observar, hay una clara mejoría de los algoritmos supervisados ante el balanceo de los grupos. Estos resultados es posible que se puedan mejorar con un cross validation mayor, pero a efectos de investigación se acaba de demostrar que estos resultados mejoran, bajo las mismas condiciones, si los grupos se balancean.

Respecto al algoritmo no supervisado, no se puede obtener una valoración objetiva sobre él ya que funcionan sin la etiqueta “grupo”, de tal manera que los resultados obtenidos no son comparables con la realidad. Teniéndolo en cuenta, sus resultados se publicarán a continuación:

- Dendrograma

Con el dendrograma se ha obtenido el siguiente resultado:

**4-8. Dendrograma Final**

Como se puede observar, el elemento outlier se ha obtenido como un grupo aparte, mientras que el resto de pacientes han sido clasificados en tres grupos. Se ha dejado este elemento outlier como prueba, debido a que la psicóloga ha afirmado que es un chico sin ninguna patología especial, por lo que resulta extremadamente interesante como el único paciente sin patología ha sido determinado como grupo único.

## 4.2 Conclusiones

De este trabajo multidisciplinar se pueden obtener numerosas conclusiones, las cuales se expondrán a continuación:

- 1) Los trastornos psicológicos son un elemento a la orden del día, cuyo diagnóstico y tratamiento está enfocado en el “framework” de la psicología cognitivo-conductual. La mayor parte de los psicólogos actualmente trabajan bajo este marco, y este trabajo ha sido enfocado de la misma manera, obteniendo resultados satisfactorios según lo establecido.
- 2) Estos trastornos psicológicos PUEDEN ser predichos en términos generales, con grupos generales, en acierto cercano al 80%, a través de las distorsiones cognitivas y otros factores, como la edad, el sexo o las habilidades sociales de la persona. Este trabajo no intenta demostrar que una máquina puede predecir mejor que una persona los trastornos, puesto que aparte de que ha quedado demostrado que la fiabilidad no se acerca al 100% (excepto en Deep Learning), hay numerosos factores que pueden hacer que una persona posea varios trastornos al mismo tiempo, y es bastante complicado que una máquina llegue a dichas conclusiones.
- 3) Un problema real de ciencia de datos, y por ende de machine learning, nunca va a tener un dataset idílico sobre el que trabajar, por lo que sobre el mismo se deben de hacer numerosas transformaciones para su posterior uso en machine learning. En este caso, aparte de la ligera limpieza que hubo que hacer (se hizo en la recogida), es muy interesante el hecho de haber clases poco balanceadas en la muestra, por lo que hubo que balancearlas. Además, otras técnicas como el PCA tuvieron que ser aplicadas.
- 4) Los algoritmos de machine learning no trabajan bien con pocas observaciones (pobre generalización), lo cual probablemente ha hecho que no se puedan afinar más los resultados, ni tampoco con demasiadas dimensiones (maldición de la dimensionalidad), fenómeno que no ha llegado a ocurrir en este trabajo. Tampoco se deben reducir demasiado las dimensiones del dataset (ver figura 2-4), debido a que cuantas menos dimensiones haya menor explicación habrá, y más difícil será una correcta clasificación de los datos. En este trabajo, ha sido realmente importante la elección de las variables, y por ende el business understanding, para obtener buenos resultados, puesto que la cantidad de observaciones era extremadamente pequeña para un problema de este tipo.

- 5) Si el dataset está balanceado, los algoritmos supervisados más comunes como K Nearest Neighbors, Support Vector Machines y Random Forest (Tree) son algoritmos que funcionan muy bien con este problema. En el caso de KNN, se ha observado que consigue buenos datos haciendo overfitting, pudiendo ser por la escasez de los pacientes, en SVM se ha comprobado que el mejor kernel es RBF, lo que indica que no existe una buena separación lineal, y el algoritmo de Tree Random Forest consigue resultados bastante buenos en base a la diferenciación con preguntas.

### 4.3 Líneas Futuras, Ampliaciones y Entornos de Aplicación

El actual trabajo está hecho con una intencionalidad puramente investigadora y académica, no teniendo una finalidad de despliegue comercial. Debido a ello, una línea futura podría ser el despliegue de este sistema aplicando alguno de los algoritmos más exitosos sobre una plataforma web, o una aplicación móvil, de tal manera que al introducir las variables del paciente se pudiera visualizar la predicción del trastorno que sufre.

Junto con ello, se debería aumentar el espectro de los trastornos psicológicos, puesto que en este trabajo se están clasificando los cuatro tipos más comunes, pero se podría ampliar hacia el resto de los trastornos.

Como está indicado en las conclusiones, un sistema de esta índole nunca deberá de suplir al experto humano, puesto que muchas otras variables y sensaciones en la entrevista personal son importantes a la hora de diagnosticar un trastorno, pero ello no impide que un sistema de este tipo sirva de confirmación hacia las sospechas que un psicólogo pueda tener hacia el trastorno de un paciente.

Otra línea de ampliación consistiría en la prueba de otros algoritmos, como AdaBoosting, Extreme Gradient Boosting, Spectral Clustering o incluso Bayes, ya que son otros métodos de clasificación que podrían dar buenos resultados, aunque personalmente no creo que fueran muy superiores a los actualmente obtenidos.

Finalmente, una línea de ampliación muy interesante, para hacer más útil la primera ampliación, sería la creación de un sistema de envío de datos de pacientes totalmente anonimizado, de tal manera que la base de datos de pacientes aumentara de los 67 actuales a numerosos pacientes más, de tal forma que los algoritmos puedan ir mejorando con el paso de los mismos.

# Anexo de Términos

---

**Business Intelligence:** Proceso por el cual se transforman una serie de datos y conclusiones en información, y esta información en conocimiento usable para la empresa, de tal manera que con este conocimiento la empresa pueda tomar decisiones basadas en los datos. Suele abreviarse como BI.

**Dataset:** Representación de datos, conteniendo también las columnas (dimensiones), que proporcionará los datos para hacer los métodos de data science y obtener las conclusiones pertinentes.

**Discretización:** Consiste en el proceso mediante el cual una serie de variables cuantitativas o cualitativas se separan en clases, obteniendo una clase por cada valor.

**Framework:** Consiste en un entorno de trabajo basado en un conjunto de estándares referentes a conceptos y criterios que deben de ser respetados por todos los que lo usen.

**Función de activación:** Es una función determinada que determina el valor de la salida de nodo de la red neuronal, obteniendo como parámetros las entradas que tenga dicho nodo. Existen numerosas funciones de activación, siendo la más común la ReLU.

**Gradiente (Red Neuronal):** Vector consistente en las derivadas parciales de una cierta función, cuya dirección indica la dirección de mayor aumento de dicha función, que se corresponde con la búsqueda del mínimo en las redes neuronales.

**IMC:** Siglas referentes al índice de masa corporal. Es una estimación de la cantidad de grasa corporal que posee una persona en su cuerpo, aunque es poco fiable debido a que no es capaz de distinguir entre el músculo y la grasa y tampoco tiene en cuenta la complexión de las personas. Su fórmula es la siguiente:

$$IMC = \frac{\text{Peso en Kg}}{\text{Altura}^2}$$

**Intervalo:** Rango de valores, representado con el valor mínimo de dicho rango y con el valor máximo. Puede ser abierto o cerrado, dependiendo de si los valores que forman sus extremos están incluidos o no en el intervalo.

**KDD:** Siglas de Knowledge Discovery in Databases, consiste en un proceso, dividido en diversas etapas de recolección, tratamiento y visualización, que consiste en la identificación de patrones útiles, usables y entendibles entre una gran cantidad de datos.

**Monte-Carlo (Método):** Algoritmo computacional basado en la obtención de muestras totalmente aleatorias para obtener una serie de resultados numéricos. Su principal objetivo, por

lo tanto, es usar la aleatoriedad para solucionar un problema determinístico. Es normalmente usado en matemáticas, física y ciencia de datos, cuando aproximaciones hacia la solución son demasiado complicadas.

**Modelo (Machine Learning):** Consiste en la unión de un algoritmo de machine learning junto con una serie de datos. Estos datos primeramente entrenan al algoritmo, para que posteriormente, al introducir nuevos datos, el algoritmo pueda clasificar o predecir correctamente.

**Outlier:** Conocido en español como “valor extremo”, es un valor de un dataset cuyo valor es muy distante del resto de los valores y puede desvirtuar tanto el análisis estadístico como los modelos de machine learning.

**Sistema Gestor de Base de Datos:** Software que controla los accesos a la base de datos y sirve como interfaz del usuario hacia los datos.

**TDAH:** Siglas referentes al trastorno por déficit de atención con hiperactividad. Consiste en un trastorno neuronal, desarrollado desde las primeras etapas de la infancia, en el que el sujeto presenta un déficit de atención por encima de lo normal acompañado con impulsividad o hiperactividad. Suele ser un trastorno que conlleva rechazo social y problemas académicos.

## Bibliografía

1. **Beriso Gómez-Escalonilla, Á., Plans Beriso, B., Sánchez-Guerra Roig, M. y Sánchez Peláez, D.** (2003). *Cuaderno de Terapia Cognitivo-Conductual (Una orientación pedagógica e integradora)*. Madrid: EOS.
2. **Burns, D.** (1980). *Sentirse Bien*. Barcelona: Editorial Paidós.
3. **Morris, C. y Maisto, A.** (2005). *Introducción a la Psicología*. México: Prentice Hall.
4. **González Muñoz, M.M.** (2010). *Estrategias metodológicas para el desarrollo de las habilidades sociales en el ámbito educativo*. Salamanca: JetPrint.
5. **American Psychiatric Association** (2002). *Manual Diagnóstico y Estadístico de los Trastornos Mentales DSM-IV-TR*. Barcelona: Masson.
6. **Tan, P., Steinbach, M. y Kumar, V.** (2006). *Introduction to Data Mining*. Boston: Pearson.
7. **Hurwitz, J. y Kirsch, D.** (2018). *Machine Learning*. Hoboken: John Wiley and Sons.
8. **LeCun, Y., Bengio, Y. y Hinton, G.** (28 de Mayo de 2015). Deep Learning. *Nature*. 521, 436-444.
9. **Chun-Houh, C., Härdle, W. y Unwin, A.** (2008). *Handbook of data visualization*. Leipzig: Springer.
10. Monografías.com [En Línea]. Obtenido de: <https://www.monografias.com/trabajos90/la-psicologia-cognitiva/la-psicologia-cognitiva.shtml>
11. Universidad de Barcelona [En Línea] Obtenido de: [http://www.ub.edu/dpsed/fvillar/principal/pdf/proyecto/cap\\_06\\_proc\\_info.pdf](http://www.ub.edu/dpsed/fvillar/principal/pdf/proyecto/cap_06_proc_info.pdf)
12. Psicología y Mente [En Línea]. Obtenido de: <https://psicologiymente.com/psicologia/conductismo>
13. Slideshare [En Línea]. Obtenido de: <https://www.slideshare.net/Arlinzon/enfoque-cognitivo-conductual-historia-de-la-psicologia>
14. Universidad de Alicante [En Línea]. Obtenido de: [https://rua.ua.es/dspace/bitstream/10045/3834/29/TEMA%205\\_PROCESOS%20PSICOL%C3%93GICOS%20BASICOS.pdf](https://rua.ua.es/dspace/bitstream/10045/3834/29/TEMA%205_PROCESOS%20PSICOL%C3%93GICOS%20BASICOS.pdf)
15. Organización Mundial de la Salud [En Línea]. Obtenido de: <https://www.who.int/mediacentre/factsheets/fs396/es/>
16. Inside Big Data [En Línea]. Obtenido de: <https://insidebigdata.com/2014/11/09/ask-data-scientist-importance-exploratory-data-analysis/>
17. Aukera [En Línea]. Obtenido de: <https://aukera.es/blog/data-science-que-es-y-que-no-es/>
18. IBM [En Línea]. Obtenido de: [https://www.ibm.com/support/knowledgecenter/en/SSEPQG\\_10.1.0/com.ibm.datatools.datamining.doc/c\\_dp\\_datapreparationoverview.html](https://www.ibm.com/support/knowledgecenter/en/SSEPQG_10.1.0/com.ibm.datatools.datamining.doc/c_dp_datapreparationoverview.html)
19. Medium.com [En Línea]. Obtenido de: <https://medium.freecodecamp.org/an-introduction-to-q-learning-reinforcement-learning-14ac0b4493cc>
20. Medium.com [En Línea]. Obtenido de: <https://medium.com/@violante.andre/simple-reinforcement-learning-temporal-difference-learning-e883ea0d65b0>

21. BBVA [En Línea]. Obtenido de: <https://bbvaopen4u.com/es/actualidad/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos>

22. Credera [En Línea]. Obtenido de: <https://www.credera.com/blog/technology-insights/java/mongodb-explained-5-minutes-less/>

23. Medium [En Línea]. Obtenido de: <https://medium.freecodecamp.org/what-exactly-is-node-js-ae36e97449f5>