

TRABAJO FIN DE GRADO  
GRADO EN INGENIERÍA INFORMÁTICA  
MENCIÓN EN SISTEMAS DE INFORMACIÓN

# **Selección automática de características mediante evolución diferencial. Aplicación en la selección de ratios financieros para la determinación de solvencia empresarial.**

**Estudiante:** Sestayo Yermo, Óscar  
**Director/a/es/as:** Santos Reyes, José  
Rodríguez López, Manuel

A Coruña, 25 de junio de 2020.



*A Mercedes, Ramón y Xaquín*



### **Agradecimientos**

Quiero dar las gracias a los directores de este proyecto de fin de grado, D. José Santos Reyes y D. Manuel Rodríguez López, ya que, sin su ayuda no sería posible la realización de este trabajo. Y también agradecer a Ángel Beade el tiempo que me ha dedicado.



## Resumen

En este proyecto se utilizará selección automática de características mediante evolución diferencial, aplicada a la selección de ratios financieros para la predicción de fracaso empresarial. Se partirá de un conjunto amplio de ratios financieros, comúnmente usados en el campo del análisis financiero de empresas. Se usará una muestra de empresas de la economía gallega en la que se dispone de tales ratios, y se predecirá el fracaso de la empresa a un año vista.

Un problema fundamental en clasificación es la selección de aquellas variables o características que mejor definen la distinción entre las diferentes categorías o clases. En este proyecto se va a utilizar, principalmente, un método evolutivo (Evolución Diferencial - ED) para determinar de modo automático aquellas características que proporcionan más potencial diferenciador en problemas de clasificación.

En la selección de características se usarán, en primer lugar, análisis clásicos independientes del modelo predictor o de clasificación. Se buscarán características o ratios que presenten poca variabilidad dentro de una misma clase y, al mismo tiempo, presenten valores claramente diferenciadores en diferentes clases. Se usarán las medidas clásicas de T-estadístico y *Fisher score*. Estas técnicas de estimación nos proporcionarán un primer filtrado de posibles ratios significativos en nuestro problema de predicción/clasificación de empresas que fracasan/no fracasan en el año siguiente.

Adicionalmente se usará un algoritmo evolutivo (Evolución Diferencial) para seleccionar aquellos ratios más significativos en la clasificación. Esta selección sí será dependiente del modelo clasificador. La idea es que el método evolutivo optimice el conjunto de ratios que más poder discriminatorio proporcionan al modelo clasificador. La población genética codificará posibles conjuntos o combinaciones de ratios, y la calidad de cada solución codificada vendrá determinada por la calidad del clasificador una vez entrenado. Dados los requerimientos de tiempo implicados en el entrenamiento de un clasificador, por ejemplo, conexionista, usamos en este punto un clasificador sencillo como K-NN. Por tanto, cada solución de la población genética determina una combinación de ratios a utilizar mediante el clasificador K-NN, y la calidad de la solución vendrá determinada por alguna medida de la clasificación proporcionada por K-NN en una muestra de empresas de validación.

Finalmente, aunque la selección evolutiva de los ratios más significativos es dependiente del clasificador considerado (K-NN), se compararán los resultados de clasificación al utilizar el

---

conjunto optimizado de ratios con un clasificador más robusto, como un clásico perceptrón multicapa. La red neuronal artificial (RNA) se entrenará con el conjunto evolutivamente seleccionado de ratios, comparando los resultados con respecto a no utilizar selección de ratios y con respecto a métodos de selección basados en análisis de sensibilidad en la RNA entrenada.

Palabras clave:

- Algoritmos evolutivos
- Clasificación
- Predicción fracaso empresarial
- Selección automática de características





# Índice general

---

<b>1</b>	<b>Historia de la evolución del análisis de ratios financieros</b>	<b>1</b>
1.1	Fracaso Empresarial . . . . .	1
1.1.1	¿Qué es el fracaso empresarial? . . . . .	1
1.1.2	Tipos de fracaso empresarial . . . . .	2
1.2	Ratios financieros . . . . .	3
1.2.1	Definición . . . . .	3
1.2.2	Historia . . . . .	3
1.2.3	Fase Descriptiva . . . . .	4
1.2.4	Fase Predictiva . . . . .	4
<b>2</b>	<b>Métodos utilizados</b>	<b>7</b>
2.1	Métodos de selección . . . . .	7
2.1.1	<i>Fisher Score</i> . . . . .	8
2.1.2	T-estadístico . . . . .	8
2.2	<i>K-nearest neighbors</i> (K-NN) . . . . .	9
2.3	<i>Evolución diferencial</i> (ED) . . . . .	11
2.4	Aplicación del algoritmo ED y k-nearest neighbors al problema de selección automática de características . . . . .	14
2.4.1	Parámetros . . . . .	16
2.5	Matlab . . . . .	17
2.6	La neurona artificial y las redes de neuronas artificiales . . . . .	17
2.6.1	Neurona artificial . . . . .	17
2.6.2	Redes de neuronas artificiales . . . . .	18
2.6.3	Perceptron multicapa . . . . .	19
2.7	Medidas de clasificación . . . . .	22
<b>3</b>	<b>Selección de la muestra de datos de empresas</b>	<b>25</b>
3.1	Datos . . . . .	25

3.1.1	Selección de la población . . . . .	25
3.1.2	Criterio del fracaso . . . . .	26
3.1.3	Selección de los ratios . . . . .	27
3.1.4	Ratios financieros utilizados en el proyecto . . . . .	28
3.1.5	Conjunto de datos utilizados en el proyecto . . . . .	30
3.1.6	Normalización de los datos . . . . .	31
<b>4</b>	<b>Resultados</b>	<b>33</b>
4.1	Primeras pruebas de obtención de calidad de los ratios . . . . .	33
4.1.1	Observaciones y deficiencias del T-estadístico y del <i>Fisher Score</i> . . . . .	37
4.2	Pruebas de obtención de la relevancia de los ratios a través del algoritmo formado por K-NN y ED . . . . .	37
4.2.1	Prueba con los 59 ratios y con K=15 . . . . .	38
4.2.2	Prueba con los 59 ratios y con K=3 . . . . .	39
4.2.3	Prueba con los 30 mejores ratios de Fisher Score y con K=15 . . . . .	40
4.2.4	Prueba con los 30 mejores ratios de Fisher Score y con K=3 . . . . .	40
4.2.5	Prueba con los 30 mejores ratios de T-estadístico y con K=15 . . . . .	41
4.2.6	Prueba con los 30 mejores ratios de T-estadístico y con K=3 . . . . .	42
4.3	Tabla resumen de todas las ejecuciones, generaciones y porcentajes de aciertos	44
4.4	Comparativa de porcentajes de aciertos entre K=3 y K=15 . . . . .	45
4.5	Resultado de las pruebas con el clasificador K-NN y el algoritmo ED . . . . .	45
4.5.1	Mejores ratios para la prueba con 59 ratios y K=15 . . . . .	46
4.5.2	Mejores ratios para la prueba con 59 ratios y K=3 . . . . .	47
4.5.3	Mejores ratios para la prueba con los 30 mejores ratios de Fisher Score y k=15 . . . . .	48
4.5.4	Mejores ratios para la prueba con los 30 mejores ratios de Fisher Score y k=3 . . . . .	48
4.5.5	Mejores ratios para la prueba con los 30 mejores ratios de T-estadístico y k=15 . . . . .	49
4.5.6	Mejores ratios para la prueba con los 30 mejores ratios de T-estadístico y k=3 . . . . .	49
4.6	Deducciones y observaciones a través de los ratios seleccionados por el algoritmo en las distintas pruebas . . . . .	49
4.7	Obtención de las matrices de confusión . . . . .	52
4.7.1	Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo de los 30 mejores ratios de <i>Fisher Score</i> con K=3 . . . . .	53
4.7.2	Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo de los 30 mejores ratios de <i>Fisher Score</i> con K=15 . . . . .	53

4.7.3	Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo de los 30 mejores ratios de T-estadístico con K=3 .	54
4.7.4	Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo de los 30 mejores ratios de T-estadístico con K=15	55
4.7.5	Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo del conjunto completo de los 59 ratios con K=3 . .	55
4.7.6	Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo del conjunto completo de los 59 ratios con K=15 .	56
4.7.7	Resultados del K-NN sin selección de datos . . . . .	57
4.8	Resultado de las pruebas con redes neuronales . . . . .	58
4.8.1	Resultados de la primera prueba (3-3-2-1) . . . . .	60
4.8.2	Resultados de la segunda prueba (10-7-3-1) . . . . .	60
4.9	Comparación de los resultados de la red neuronal con otro estudio . . . . .	62
<b>5</b>	<b>Conclusiones</b>	<b>65</b>
	<b>Bibliografía</b>	<b>69</b>



# Índice de figuras

---

2.1	Representación gráfica de 3-NN. . . . .	10
2.2	Representación gráfica de la operación de mutación en Evolución Diferencial en un espacio de 2 dimensiones. . . . .	11
2.3	Representación gráfica de una neurona artificial. . . . .	18
2.4	Representación gráfica del Perceptron Multicapa. . . . .	20
2.5	Parámetros utilizados para las pruebas de la red neuronal. . . . .	22
2.6	Matriz de confusión. . . . .	23
3.1	Categorización de los diferentes tipos de PYMES . . . . .	26
4.1	Valores obtenidos con la aplicación del método de selección T-estadístico. . . .	36
4.2	Valores obtenidos con la aplicación del método de selección <i>Fisher Score</i> . . . .	36
4.3	Calidad media (porcentaje de aciertos) y calidad del mejor individuo a lo largo de generaciones, usando el conjunto total de 59 ratios en la selección y $k=15$ . . . .	39
4.4	Calidad media (porcentaje de aciertos) y calidad del mejor individuo a lo largo de generaciones, usando el conjunto total de 59 ratios en la selección y $k=3$ . . . .	39
4.5	Calidad media y mejor a lo largo de las generaciones, usando los 30 mejores ratios de <i>Fisher Score</i> en la selección y $k=15$ . . . . .	40
4.6	Calidad media y mejor a lo largo de las generaciones, usando los 30 mejores ratios de <i>Fisher Score</i> en la selección y $k=3$ . . . . .	41
4.7	Calidad media y mejor a lo largo de las generaciones, usando los 30 mejores ratios de T-estadístico en la selección y $k=15$ . . . . .	42
4.8	Calidad media y mejor a lo largo de las generaciones, usando los 30 mejores ratios de T-estadístico en la selección y $k=3$ . . . . .	43
4.9	Número de veces que un ratio ha sido seleccionado en las diferentes pruebas. . .	50
4.10	Número de veces que un ratio ha sido seleccionado en las diferentes ejecuciones independientes y en la totalidad de las pruebas. . . . .	51

4.11 Matriz de confusión obtenida del trabajo de investigación de A. Beade en el conjunto de test. . . . .	62
--	----

# Índice de cuadros

---

3.1	Ratios financieros utilizados en el proyecto. . . . .	28
3.2	Ratios financieros utilizados en el proyecto. . . . .	29
4.1	Valores T-estadístico y <i>FisherScore</i> . . . . .	34
4.2	Valores T-estadístico y <i>FisherScore</i> . . . . .	35
4.3	Porcentajes de acierto de las pruebas en diferentes generaciones. . . . .	44
4.4	Porcentajes de aciertos obtenidos en las pruebas. . . . .	45





# Historia de la evolución del análisis de ratios financieros

---

## 1.1 Fracaso Empresarial

### 1.1.1 ¿Qué es el fracaso empresarial?

**E**L fracaso empresarial se conoce como la incapacidad de una compañía de cumplir con sus obligaciones actuales, o dicho de otra forma, la imposibilidad de una compañía de sostenerse por sí misma sin atender los vencimientos de su endeudamiento.

Los estudios del fracaso empresarial utilizan diferentes interpretaciones y definiciones del concepto de fracaso, lo que causa una importante variabilidad en los resultados obtenidos. La dificultad se plantea en la determinación de la variable dependiente (el fracaso empresarial), puesto que existen diferentes definiciones e interpretaciones del término para la clasificación binaria de una empresa como fracasada o no fracasada. El fracaso, según el diccionario de la Real Academia Española, está definido como “resultado adverso de una empresa o negocio” o como “caída o ruina de algo con estrépito y rompimiento”, que está directamente relacionado con el término crisis, es decir, “situación grave y decisiva que pone en peligro el desarrollo de un asunto o un proceso”. Por tanto, las crisis son situaciones temporales que pueden derivar en un fracaso, en este caso el de la empresa.

Este fracaso sería evitable si se tomaran medidas y soluciones correctas a tiempo, para que la empresa retome una estabilidad financiera. Al fin y al cabo, las empresas pueden sufrir estos periodos de crisis donde se corre un riesgo de fracaso empresarial derivado de una mala gestión. Por tanto, lo que se busca es poder sobrevivir a estas circunstancias y adelantarse al fracaso para poder obtener soluciones que lo eviten.

Cabe destacar que una empresa fracasará cuando se incumplan los objetivos que se han marcado en la empresa con anterioridad y, aunque el fracaso de la empresa se asocia con la desaparición de la misma, se debe tener en cuenta que, durante el periodo de crisis, en donde aún no se ha producido la desaparición de la empresa, habrá un tiempo en el cual se podrá detectar el causante de los malos resultados de la empresa y revertir la situación.

### **1.1.2 Tipos de fracaso empresarial**

De las múltiples acepciones del término “fracaso empresarial”, en términos generales, podría hablarse de tres tipos de fracaso diferentes: fracaso económico, fracaso financiero o insolvencia técnica y fracaso legal.

#### **Fracaso económico**

El fracaso económico comienza cuando la rentabilidad de las inversiones es inferior a la que se podría obtener a través de inversiones similares, o cuando los ingresos no son suficientes para cubrir los costes. En resumen, la empresa u organización comienza a ingresar menos de lo que gasta. Si esta situación no se revierte o detiene, la empresa empezará a tener más deudas y no podrá hacer frente a ellas, derivando en una insolvencia técnica.

#### **Fracaso financiero**

También conocido como insolvencia técnica, está presente cuando la empresa es incapaz de hacer frente a sus obligaciones de pago a medida que vencen, aunque el total de sus activos sea mayor que el total de sus deudas; esto se traduce en una falta de liquidez y puede ser tan solo temporal.

#### **Fracaso legal**

Este implica no solo la definitiva desaparición legal, sino los procesos concursales como procesos judiciales (hipotecarios o embargos), reorganizaciones o reestructuraciones, entre otros acuerdos voluntarios causados por una crisis de insolvencia, lo cual podría resolver sus dificultades y continuar su actividad económica. También es conocido, en la literatura referente al tema, como insolvencia definitiva. Además, en el sentido de quiebra, se identifica con un patrimonio neto negativo, es decir, cuando el total de los pasivos excede al total de los activos; esto conlleva en la gran mayoría de los países a una sanción legal.

## **1.2 Ratios financieros**

### **1.2.1 Definición**

Los ratios financieros (también llamados razones financieras o indicadores financieros) son coeficientes o razones que proporcionan unidades contables y financieras de medida y comparación. La relación por división entre sí de dos datos financieros directos, permite analizar el estado actual o pasado de una organización, en función de niveles óptimos definidos para ella. En las empresas los ratios se utilizan para saber cómo analizar un balance de situación. Al relacionar dos variables del balance o de la cuenta de resultados se obtiene información sobre la situación financiera de la empresa, poniendo de manifiesto una buena o mala gestión.

El objetivo principal de la utilización de ratios financieros es tener una visión global de la empresa a nivel económico-financiero. Los ratios financieros cuantifican numerosos aspectos de una empresa y forman una parte integral del análisis de los estados financieros de la misma.

En este proyecto se analizarán 59 ratios financieros de empresas gallegas, donde el criterio de selección de los mismos ha sido doble: por un lado, su popularidad en la literatura contable y financiera, y por otro, la frecuencia y significación en aquellos estudios más relevantes en cuanto al fracaso empresarial.

### **1.2.2 Historia**

Los primeros estudios referentes a los ratios financieros se efectúan no hace más de ocho décadas. Fueron los banqueros norteamericanos quienes los utilizaron originalmente como técnica de gestión. Después de la crisis de 1929, también conocida como gran depresión, el análisis financiero llevado a cabo por los banqueros hacia los clientes potenciales, se desarrolló utilizando fundamentalmente los ratios.

Los antecedentes de las investigaciones más importantes que se han desarrollado en torno a los ratios dentro de los modelos financieros para predecir quiebras parten del trabajo pionero de Fitzpatrick [1] en 1932. Posteriormente, destacarán los trabajos realizados por Arthur Winakor [2], Raymond Smith [3], y Charles L. Merwin [4]. En sus trabajos destaca el hecho de que representaron el primer paso para resumir un conjunto de procedimientos tendentes a describir el fracaso financiero únicamente a través de ratios. Estas investigaciones pertenecen a una fase denominada descriptiva. Desde entonces dichos trabajos se han enfocado en la comparación de los ratios de empresas en quiebra y empresas que no han quebrado. Actual-

mente, este pensamiento persiste en muchos análisis que tratan sobre el tema y que afirman que la tendencia de los ratios indica si una empresa se dirige o no al fracaso financiero.

Los cambios producidos en cuanto a la literatura sobre los ratios aplicados a la predicción del fracaso empresarial se dieron a finales de los años 60. En esta época es cuando aparecen los trabajos clásicos de Beaver [5] y Altman [6]. Dichos trabajos consideran a los ratios financieros como predictores uno a uno.

### 1.2.3 Fase Descriptiva

En la fase descriptiva, los trabajos empíricos se caracterizan por intentar demostrar que los ratios de las empresas que fracasaban presentaban una tendencia desfavorable conforme se acercaba la fecha del fracaso definitivo de la empresa. Los trabajos se sustentaban en la comparación de los ratios pertenecientes a las empresas con éxito y a las empresas en quiebra.

### 1.2.4 Fase Predictiva

Después de la fase descriptiva, surgió una nueva etapa con un pensamiento totalmente renovado que se basó fundamentalmente en la significancia de cada ratio y en la capacidad predictiva de los ratios más relevantes.

En esta etapa aparecen los primeros trabajos empíricos, basados en modelos estadísticos univariantes, que se concentran en una sola variable independiente y buscan encontrar y representar características propias de la misma como la media y la desviación. Por su parte, las investigaciones se centran en la solvencia a largo plazo y en la posibilidad de predecir la quiebra. Entre estos primeros trabajos empíricos destacan los de Altman [6] y Beaver [5].

Edward I. Altman, en su artículo [6], concluye que el análisis tradicional mediante ratios ya no puede considerarse una técnica analítica relevante en el ambiente académico, dada su forma relativamente no sofisticada, proponiendo un modelo predictivo de quiebras mediante el análisis discriminante. Comienza así la sustitución de los modelos univariantes por los modelos multivariantes, que se concentran en dos o más variables independientes. En este caso, se analizan dos o más ratios. De ellos, forma parte el análisis discriminante, método estadístico a través del cual se busca conocer qué características de ciertas instancias explican mejor la diferenciación entre los grupos a los cuales pertenecen dichas instancias.

Los modelos de análisis financiero predictivo forman la base de ciertas técnicas estadísticas denominadas “análisis multivariable”. Estas sirven también para la toma de decisiones. La idea básica es poder identificar y utilizar ratios de varias empresas con fines de predicción con base

en sus características de mayor predominio [7]. Según algunos analistas, esto permite detectar signos de deterioro en el comportamiento de los ratios, con una anticipación y claridad suficiente para tomar a tiempo iniciativas que eviten riesgos de insolvencia, suspensión de pagos y quiebras. Así, a los ratios se les utiliza como base de datos para medir la solvencia y evitar el fracaso empresarial por medio de estas potentes técnicas estadísticas multivariantes.

Por otra parte, aunque los ratios sean una forma sencilla de simplificar el complejo análisis financiero de una empresa, tenemos que tener en cuenta que un mismo análisis no puede ser aplicado a empresas heterogéneas debido a las diferencias que hay entre los diferentes sectores y los diferentes tamaños que pueden tener las empresas.



# Métodos utilizados

---

EN este proyecto se utilizan como herramientas fundamentales el algoritmo de clasificación *K-nearest neighbors* y el algoritmo de evolución diferencial. Estos dos algoritmos serán los cimientos sobre los que se basará este proyecto.

## 2.1 Métodos de selección

Normalmente, los métodos de selección de características se pueden clasificar en función de si el criterio de selección depende del algoritmo de aprendizaje utilizado para construir el clasificador o predictor. Se consideran normalmente dos tipos de métodos de selección: *filter methods* y *wrapper methods* [8].

*Filter methods* seleccionan las características basándose exclusivamente en el contenido de la información, típicamente mediante medidas estadísticas de las características, y son independientes de cualquier algoritmo de aprendizaje. Debido a esto, los *filter methods* son computacionalmente menos costosos, rápidos y adecuados para grandes conjuntos de datos. Además, son fácilmente aplicables a varios algoritmos de aprendizaje. Estos *Filter methods* se usan normalmente para la clasificación de características, como pueden ser los métodos de selección *Fisher Score* y *T-statistics*.

Sin embargo, los *filter methods* ignoran la interacción de características en el algoritmo de clasificación utilizado, mientras que los *wrapper methods* requieren un algoritmo de aprendizaje predeterminado, utilizando un criterio de evaluación de las características más relevantes sobre el clasificador/predictor entrenado. Es decir, buscan características más adecuadas para el algoritmo de aprendizaje con el objetivo de mejorar el rendimiento, pero también tienden a ser más costosos desde el punto de vista computacional que los *filter methods* [9].



### 2.1.1 Fisher Score

*Fisher Score* es uno de los métodos de selección más utilizados. Este método selecciona cada característica de forma independiente de acuerdo con sus puntuaciones según el criterio de Fisher [10]. Esta técnica de selección se aplicará a los ratios de diferentes empresas, para dar una orientación de cuáles son los más influyentes en el fracaso (o no fracaso) empresarial. La puntuación de Fisher se calcula a través de la siguiente formula:

$$F_r = \frac{\sum_{i=1}^c n_i (u_r^i - u_r)^2}{\sum_{i=1}^c n_i (\sigma_r^i)^2} \quad (2.1)$$

donde  $n_i$  es el número de muestras pertenecientes a la clase  $i$ ,  $u_r^i$  y  $(\sigma_r^i)^2$  representan la media y la varianza de la clase  $i$ ,  $i = (1, \dots, c)$  para la característica  $r$ , y  $u_r$  indica la media utilizando los datos de las dos clases. Cuanto más alto sea el *Fisher Score* más discriminatoria será la característica.

### 2.1.2 T-estadístico

Del mismo modo que el *Fisher Score*, el T-estadístico es una medida de selección; sin embargo, esta medida está basada en la diferencia existente entre la media de las muestras de dos categorías. Se calcula de la siguiente forma, considerando 2 clases etiquetadas con 1 y -1:

$$T(f_j) = \frac{u_j^1 - u_j^{-1}}{\sqrt{\frac{(\delta_j^1)^2}{n^1} + \frac{(\delta_j^{-1})^2}{n^{-1}}}} \quad (2.2)$$

donde  $n^1$  es el número de muestras etiquetadas con el 1,  $n^{-1}$  el número de muestras etiquetadas con el -1,  $u_j^1$  es la media de las muestras etiquetadas con el 1,  $u_j^{-1}$  es la media de las muestras etiquetadas con el -1,  $(\delta_j^1)$  es la desviación estándar de las muestras etiquetadas con el 1 y  $(\delta_j^{-1})$  es la desviación estándar de las muestras etiquetadas con el -1. La característica que más alta puntuación (en valor absoluto) reciba será considerada la más discriminatoria.

## 2.2 *K-nearest neighbors* (K-NN)

*K-nearest neighbors*, conocido normalmente por K-NN, es un método de clasificación supervisada que sirve para estimar la función de densidad  $F(x/C_j)$  de los predictores  $x$  por cada clase  $C_j$ . La idea básica sobre la que se fundamenta este método es que una instancia del problema se va a clasificar en la clase más frecuente a la que pertenecen sus  $K$  vecinos más cercanos.

Este es un método de clasificación no paramétrico, que estima el valor de la función de densidad de probabilidad o directamente la probabilidad a posteriori de que un elemento  $x$  pertenezca a la clase  $C_j$  a partir de la información proporcionada por un conjunto de prototipos.

En el reconocimiento de patrones, el algoritmo K-NN se usa como método de clasificación de objetos (elementos) basado en un entrenamiento mediante ejemplos cercanos en el espacio de los elementos. K-NN es un tipo de aprendizaje vago (*lazy learning*), en el sentido de que se almacena todo el conjunto de entrenamiento y posteriormente se clasifica buscando el ejemplo más “parecido” al que se está analizando. Su mayor debilidad es la lentitud en el proceso de clasificación, puesto que cada instancia de prueba se compara contra todo el juego de datos de entrenamiento.

Este algoritmo funciona básicamente con tres pasos que aplica a cada instancia a clasificar. El primer paso consiste en calcular la distancia euclídea de la instancia con el resto de instancias de un conjunto de entrenamiento (en el cual se conoce la clasificación correcta de cada instancia del entrenamiento). La distancia euclídea es la forma mediante la cual podremos calcular la similitud o la diferencia entre las diferentes instancias. La fórmula de la distancia euclídea es la siguiente:  $d_e(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$  en un espacio de dos dimensiones. Lo siguiente es determinar los “ $k$ ” elementos más cercanos en el conjunto de entrenamiento, es decir, los que tengan menor distancia con respecto a la instancia a clasificar; por tanto, dependiendo del número que le asignemos a la  $k$ , se considerarán más o menos vecinos. Por último, se realizará la selección de la clase a la que pertenecerá la instancia mediante una selección por mayoría. Es decir, se asignará a la clase con el mayor número de vecinos del mismo tipo.

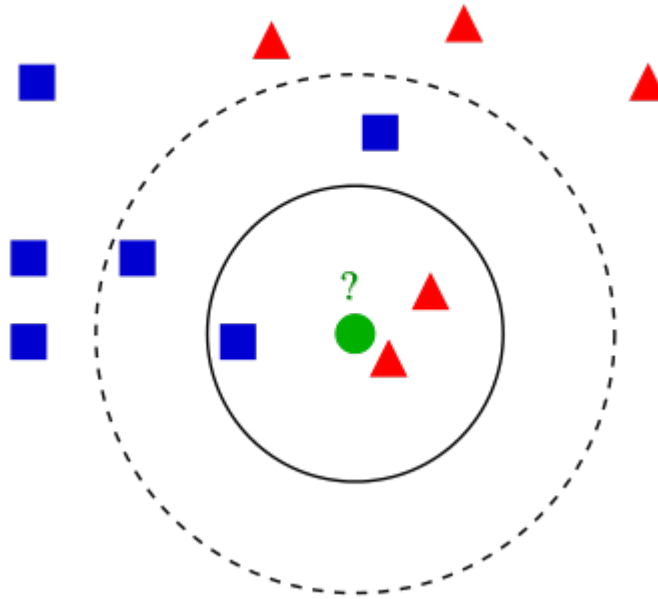


Figura 2.1: Representación gráfica de 3-NN.

En la Figura 2.1 se puede observar una representación gráfica del algoritmo *K*-NN aplicado con un  $K = 3$ , que nos indica el número de vecinos que se utilizarán para clasificar a nuestra instancia (círculo verde). En la figura se puede observar que existen dos clases, una cuyas instancias se caracterizan visualmente por cuadrados azules y, por otro lado, se puede ver a la segunda clase caracterizada en la figura por los triángulos rojos. Teniendo en cuenta que los triángulos son de una clase diferente a los cuadrados, y que entre los 3 elementos más cercanos a nuestra instancia hay dos triángulos y un cuadrado, la instancia (círculo verde) será clasificada en el grupo de los triángulos.

Dependiendo de la precisión con la que se quiera realizar la clasificación, sólo se tendrá que variar el parámetro  $k$ ; a mayor  $k$ , se podría obtener precisión en la clasificación. No existe una regla que determine cuál es el valor más apropiado para  $k$ , sino que este depende del problema de clasificación. En este proyecto usaremos 2 valores ( $k=3$  y  $k=15$ ) para chequear los resultados en función de diferentes valores (típicos) usados en diferentes aplicaciones.

### 2.3 Evolución diferencial (ED)

El algoritmo de Evolución Diferencial (ED) [11][12][13][14] es una simple pero poderosa técnica de búsqueda estocástica basada en una población de soluciones para resolver problemas de optimización global y utilizada con éxito en muchas aplicaciones de optimización. La idea central de ED es el uso de “vectores de diferencia” para generar perturbaciones en una población de vectores o soluciones: en cada individuo o solución se realiza una mutación o perturbación basada en la distribución actual de las soluciones, determinada por la diferencia entre dos vectores seleccionados aleatoriamente. Por lo tanto, las direcciones de búsqueda y la exploración considerada ( $F(x_{2i} - x_{3i})$ ) en el algoritmo (pseudocódigo en Algoritmo 1) dependen de la ubicación de los individuos en la población actual. La primera versión de ED por Storn y Price apareció en su informe técnico en 1995 [15]. La siguiente historia del algoritmo se detalla, por ejemplo, en el informe de ED realizado por Das y Suganthan [13] o en el libro de Feoktistov [14].

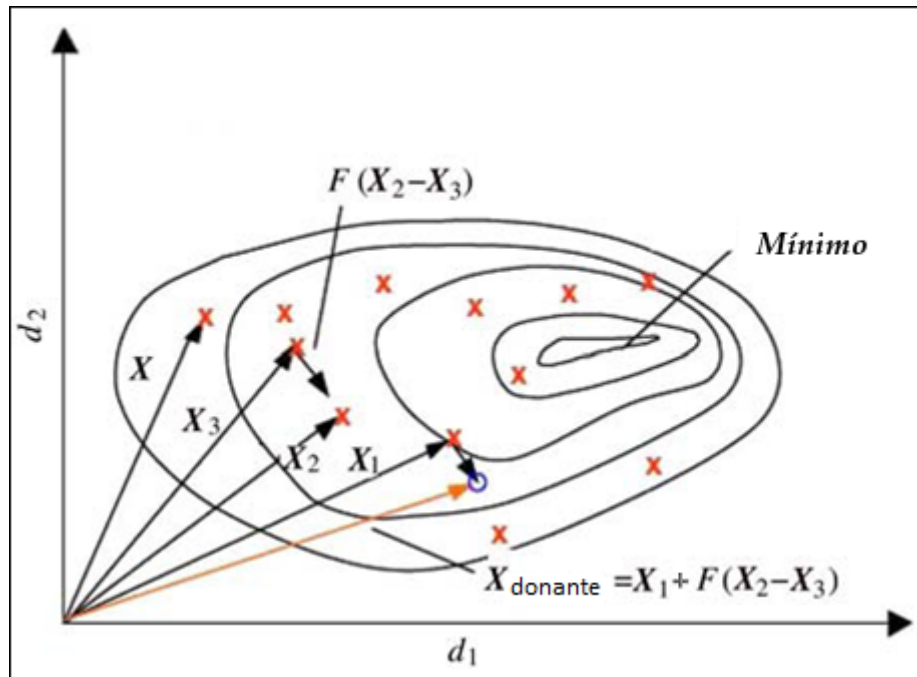


Figura 2.2: Representación gráfica de la operación de mutación en Evolución Diferencial en un espacio de 2 dimensiones.

Este algoritmo es especialmente adecuado para problemas de optimización donde las posibles soluciones se definen mediante un vector de valores reales. Este algoritmo generalmente ofrece convergencia rápida, robustez, simplicidad conceptual, pocos parámetros y fácil implementación [14]. ED usa sólo 3 parámetros de control: el tamaño de la población, el factor de diferenciación ( $F$ ) y la probabilidad de cruce ( $CR$ ). Para cada individuo o vector  $x$  de la población se genera una solución o vector candidato (vector  $u$  en el pseudo-código del Algoritmo 1), a través de las operaciones de mutación y cruce.

Siendo  $x_1, x_2$  y  $x_3$  individuos elegidos aleatoriamente de la población (diferentes al individuo actual  $x$ ), la diferencia vectorial entre dos individuos elegidos al azar,  $F(x_2 - x_3)$  define la dirección y la longitud del paso de búsqueda. Luego, esta diferencia se suma al tercer individuo escogido aleatoriamente,  $x_1$ , para definir un “vector donante”  $v$  (línea 6 en el pseudo-código). Se realiza una operación de cruce posterior entre el vector objetivo  $x$  y su vector donante ( $x_1 + F(x_2 - x_3)$ ) para definir el “vector candidato”  $u$  (línea 15). En el operador de selección final, para cada vector objetivo  $x$ , si su vector candidato tiene una calidad igual o mejor, entonces se reemplaza el vector objetivo  $x$  por el vector candidato  $u$  en la siguiente generación (línea 16).

Al comienzo del proceso evolutivo los individuos iniciales aleatorios tienden a estar separados uno del otro y, por tanto, la diferencia de vectores proporciona alta exploración en el espacio de búsqueda. Cuando la población se concentra, en generaciones posteriores, alrededor de áreas prometedoras, las diferencias tienden a ser menores. De este modo, a medida que progresa el proceso evolutivo, la alta exploración se reduce y se incrementa la explotación en la vecindad de las soluciones ya situadas en áreas prometedoras del espacio de búsqueda, sin perder un cierto grado de exploración al elegir aleatoriamente  $x_2$  y  $x_3$ . Por tanto, el propio funcionamiento de ED genera un control o equilibrio dinámico y adaptativo entre exploración y explotación.

---

**Algoritmo 1** Algoritmo Evolución Diferencial.

---

```

1: Inicializar la población aleatoriamente
2: repeat
3:   for all individuo  $x$  de la población do
4:     Elegir  $x_1, x_2, x_3 \in$  población, obtenidos aleatoriamente  $\{x_1, x_2, x_3, x$  diferentes uno del otro.}
5:     {Proceso de Mutación}
6:      $v_i \leftarrow x_{1i} + F(x_{2i} - x_{3i})$   $\{v = [v_1, v_2 \dots v_n]$  es el vector donante}
7:     {Proceso de Cruce}
8:     Elegir  $R \in \{1, \dots, D\}$ , obtenido aleatoriamente  $\{D$  es la longitud de la cadena (dimensión del espacio de búsqueda).}
9:     for  $i = 1$  to  $n$  do
10:      Escoger  $r_i \in U(0, 1)$  uniformemente en el rango abierto  $(0,1)$ .
11:      if  $(i = R) \vee (r_i < CR)$  then
12:         $u_i \leftarrow x_{1i} + F(x_{2i} - x_{3i})$ 
13:      else
14:         $u_i = x_i$ 
15:      end if
16:      end for  $\{u = [u_1, u_2 \dots u_n]$  es un nuevo vector de prueba o candidato generado}
17:      {Proceso de Selección}
18:      if  $f(u) < f(x)$  then
19:        Reemplazar individuo  $x$  por  $u$ 
20:      end if
21:    end for
22:  until Criterio de finalización
23: return  $z \in$  población  $\forall t \in$  población,  $f(z) \leq f(t)$ 

```

---

## 2.4 Aplicación del algoritmo ED y *k*-nearest neighbors al problema de selección automática de características

Para la selección de los ratios más relevantes se va a utilizar una combinación entre el algoritmo de ED y el clasificador K-NN. Cada individuo o vector de la población genética va a codificar en su genotipo un conjunto de ratios seleccionados. Como en cualquier algoritmo evolutivo, a cada solución codificada se le debe asignar una calidad o bondad, que determina cuán bien soluciona el problema concreto a resolver. En este caso, la calidad vendrá definida por cuán bien un clasificador concreto realiza la clasificación a partir del conjunto de ratios seleccionados especificados en el genotipo de cada solución. Y ese clasificador será un sencillo K-NN. Por tanto, esta combinación o hibridación entre ED y K-NN es un método de selección de la categoría *wrapper methods*, dado que la selección depende del clasificador usado.

En evolución diferencial el primer paso es la generación aleatoria de una población inicial. En primer lugar se creará dicha población, formada por vectores que contendrán 59 valores aleatorios entre -1 y 1, y dichos valores indicarán si el ratio será utilizado por el K-NN o no. En caso de que el valor sea positivo, indicará que el ratio perteneciente a esa posición será utilizado por el clasificador; por el contrario, si el valor es negativo no se usará para la clasificación. La posición de los 59 valores generados aleatoriamente corresponderán, por orden, de 1 hasta el 59, a cada uno de los ratios de las diferentes empresas utilizadas para este proyecto. Por tanto, en caso de que el primer valor de esos 59 salga positivo, se estará indicando que el ratio número uno de las empresas será utilizado en la clasificación del K-NN, y esto mismo pasará para los 58 valores restantes. Es decir, se utiliza una codificación real (como requiere ED) para especificar qué ratios se seleccionan. Por tanto, la indicación binaria del ratio seleccionado puede tener múltiples valores numéricos, lo cual es además interesante para mantener diversidad de soluciones candidatas en el algoritmo de ED.

Para cada conjunto de ratios indicados en cada genotipo se aplica el K-NN. Utilizando el fichero de test, para cada empresa se calculan las distancias euclídeas contra todas las empresas del fichero de entrenamiento, pero usando únicamente los ratios seleccionados en el cálculo de las distancias. Se determinan las *k* empresas “más cercanas” (en términos de distancia euclídea) y se realiza la clasificación del algoritmo K-NN previamente explicada.

Una vez hecho esto, solo tendremos que comparar si la empresa o las empresas que se han seleccionado del fichero de test, a partir de la clasificación, están señaladas como empresas que fracasan o empresas que no fracasan. En el caso de que el K-NN solo clasifique con la empresa más cercana, es decir, que clasifique solo a través del vecino más cercano (1-NN), solo

hará falta comprobar a qué clase (fracasada o no fracasada) pertenece la empresa. En el caso de que la clase de esta empresa del fichero test coincida al mismo tiempo con la clase de la empresa del fichero de entrenamiento con la que se está comparando, se obtendrá un acierto, ya que el K-NN ha clasificado bien esta empresa. Si, por el contrario,  $k > 1$ , se mirará cuál es el número de empresas de una misma clase que más se repite en función del fracaso/no fracaso: si hay un mayor número de empresas de una clase que de otra, se comprobará si la clase de la empresa del fichero de test coincide con la mayoría de empresas de esa determinada clase del fichero de entrenamiento. En caso afirmativo, se producirá un acierto en la clasificación; por el contrario en caso negativo, se hará un clasificación incorrecta.

De esta forma habrá 4 tipos de resultados que posteriormente se indicarán en las matrices de confusión:

- La empresa del fichero test como la del fichero de entrenamiento pertenecen a la clase fracaso; por tanto, se producirá un acierto. Este acierto será considerado como “acierto de fracaso” (Verdadero Positivo - VP).
- La empresa del fichero test como la del fichero de entrenamiento pertenecen a la clase no fracaso; por tanto, se producirá un acierto. Este acierto será considerado como “acierto de no fracaso” (Verdadero Negativo - VN).
- La empresa del fichero test pertenece a la clase no fracaso y la empresa del fichero de entrenamiento pertenece a la clase fracaso; por tanto, se producirá un error. Este error será considerado como “error de no fracaso” (Falso Positivo - FP).
- La empresa del fichero test pertenece a la clase fracaso y la empresa del fichero de entrenamiento pertenece a la clase no fracaso; por tanto, se producirá un error. Este error será considerado como “error de fracaso” (Falso Negativo - FN).

Se tienen que seguir las mismas reglas clasificadoras si se ha seleccionado el clasificador con un número de vecinos mayor que 1. La clase con mayor número de empresas del fichero de entrenamiento será la clase asignada en la clasificación de la empresa del fichero de test.

Una vez obtenidos los resultados tanto del número de aciertos como del número de errores, se procederá a hacer un recuento de ellos y obtener la tabla de confusión de la clasificación. Como el número de empresas en el fichero de test está balanceado entre ambas clases, usaremos como medida de clasificación el porcentaje de aciertos (accuracy) : a mayor número de aciertos mayor calidad, y este porcentaje de aciertos será el valor asignado a cada instancia de la población como su calidad.

Como se ha explicado en el apartado 2.3, para cada individuo  $x$  (vector objetivo) de la población de ED se seleccionan 3 instancias elegidas aleatoriamente de la población  $(x_1, x_2, x_3)$ .



Una vez seleccionadas las 3 instancias aleatorias se aplicarán a la etapa conocida en el algoritmo de evolución diferencial como mutación, generando el vector donante (Algoritmo 1). La fórmula aplicada para obtener el vector donante es la siguiente:  $v = x_1 + F * (x_2 - x_3)$  donde  $F$  es un valor aleatorio entre 0.1 y 0.9, y  $x_1$ ,  $x_2$  y  $x_3$  son las tres instancias aleatorias seleccionadas anteriormente. Una vez tenemos el individuo  $v$  (vector donante) y el individuo  $x$  (vector objetivo) se podrá pasar a la etapa de cruce.

En la etapa de cruce se modifica algún valor del individuo  $x$  de la población inicial, esto se hace a través de CR (probabilidad de cruce) y una condición. En este caso, la condición es que un número aleatorio entre 0 y 1 sea menor que la probabilidad de cruce (línea 10 Algoritmo 1), que en el caso de este programa es una constante igual a 0.1; por lo tanto, una probabilidad bastante baja. Esto se podría traducir a que se obtiene una probabilidad de un 10% de que un valor del individuo  $x$  pase a ser modificado por el valor obtenido por el vector donante  $v$ . Esta probabilidad CR baja permite una presión selectiva baja en la búsqueda de soluciones óptimas. Como se indica en el pseudo-código del algoritmo, se garantiza además que el individuo candidato final tenga al menos un valor proveniente del vector donante.

Una vez se obtiene el vector candidato se volverá a repetir el proceso de transformación de los valores reales entre -1 y 1 dependiendo si el ratio va a ser utilizado (1) o no va a ser usado (0), para indicar qué ratios serán utilizados en el clasificador. Al vector candidato se le determina de nuevo su calidad como se ha explicado anteriormente.

Por último, se hace una comparativa entre la calidad del vector que se obtuvo inicialmente de la población  $x$  y el vector candidato que se obtuvo después de la aplicación del algoritmo de evolución diferencial (línea 16 en Algoritmo 1). El individuo con mejor calidad entre ambos se selecciona para la siguiente generación. Por tanto, ED es “elitista”, porque el mejor individuo o se mejora o se mantiene en la población.

### 2.4.1 Parámetros

Se han realizado diversas pruebas para observar los diferentes resultados introduciendo valores en los parámetros del programa. Los parámetros que se utilizan en el programa son: el número de generaciones (MaxIt); el número de individuos de la población (nPop); el número de variables que tiene cada individuo de la población, que en este caso sería el número de ratios especificados de cada empresa (nVar) y el número de vecinos que se quieren comparar en el K-NN ( $k$ ). En el capítulo de resultados se muestran estas pruebas.

## 2.5 Matlab

Matlab es la herramienta utilizada para el desarrollo de este proyecto. Matlab (abreviatura de MATrix LABoratory, “laboratorio de matrices”) es una herramienta de software matemático que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M). Está disponible para las plataformas Unix, Windows, Mac OS X y GNU/Linux. Todos los algoritmos se han desarrollado en ese lenguaje de programación propio de Matlab y la representación gráfica será visualizada a través de Matlab y Excel. Este lenguaje es interpretado, y puede ejecutarse tanto en el entorno interactivo, como a través de un archivo de *script* (archivos \*.m). Este lenguaje permite operaciones de vectores y matrices, funciones, cálculo lambda y programación orientada a objetos. La versión utilizada es la R2018a.

Entre las funciones básicas del lenguaje de Matlab se hallan la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos hardware. El paquete MATLAB dispone de dos herramientas adicionales que expanden sus prestaciones: Simulink (plataforma de simulación multidominio) y GUIDE (editor de interfaces de usuario). Además, se pueden ampliar las capacidades de MATLAB con las cajas de herramientas (toolboxes); y las de Simulink con los paquetes de bloques (blocksets).

En la última fase del proyecto también se utilizará el módulo de redes de neuronas artificiales que incluye Matlab. La red que se utilizará en este proyecto será un clásico Perceptron Multicapa [16] [17] [18], cuyos aspectos básicos se resumen en el siguiente capítulo.

## 2.6 La neurona artificial y las redes de neuronas artificiales

### 2.6.1 Neurona artificial

El primer modelo de la historia de la neurona artificial fue diseñado por el matemático Walter Pitts [19] [20] [21] y el psiquiatra y neuroanatomista Warren McCulloch en el año 1943. En la figura 2.3 se muestra una representación sencilla de una neurona artificial.

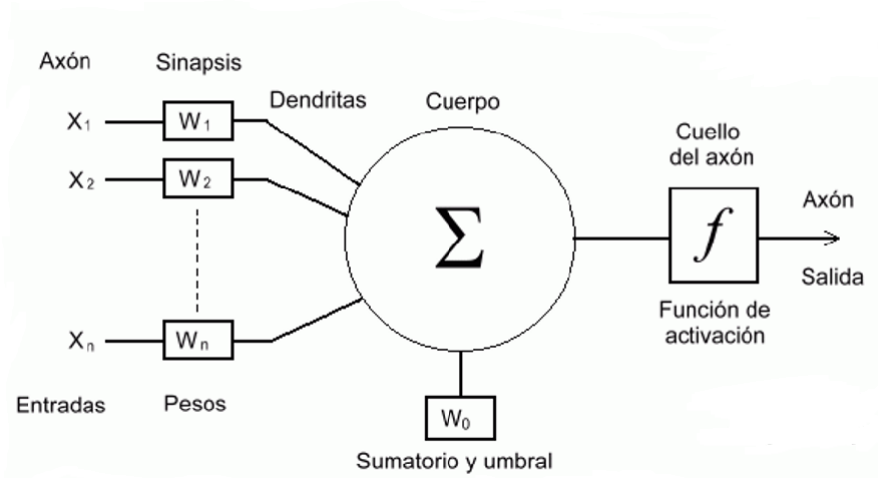


Figura 2.3: Representación gráfica de una neurona artificial.

En la figura anterior se puede contemplar un modelo básico de neurona artificial donde  $x_i$  es un conjunto de entradas,  $w_i \dots w_n$  los pesos sinápticos correspondientes a cada entrada, el sumatorio es la función de agregación,  $f$  la función de activación que genera la salida  $Y$ .

La función de transferencia o función de activación es un componente esencial de las neuronas artificiales, ya que su objetivo es limitar el rango de salida de la neurona a partir de los valores que recibe [22]. Una buena función de activación debe cumplir, en primera instancia, que ella misma y su derivada sean de fácil computo, además de tener una amplia parte lineal para lograr velocidad de entrenamiento y de convergencia en pocos ciclos. Cabe destacar que la función de activación de la neurona puede ser de tres tipos diferentes:

- Función umbral, `hardlim(n)` en Matlab.
- Función lineal, `purelim(n)` en Matlab.
- Función sigmoide, `logsig(n)` y `tansig(n)` en Matlab.

Las funciones sigmoide, logarítmica y tangente hiperbólica son las más utilizadas en problemas de predicción, debido a su rápida convergencia y sus características de derivación.

### 2.6.2 Redes de neuronas artificiales

Las redes neuronales artificiales intentan imitar el funcionamiento de las redes neuronales biológicas. Con estos sistemas lo que se busca es conseguir capacidad de aprendizaje, es decir, la red debe aprender por sí misma. Las neuronas que forman esta red podrán recibir señales, procesarlas y enviarlas de nuevo a otra neurona. Las señales enviadas entre las neuronas son números reales, y las salidas se calculan a través de alguna función no lineal del sumatorio de sus entradas. Las conexiones tendrán un peso que se ajusta a medida que avanza el aprendi-

zaje.

Cada red neuronal está formada por tres tipos de capas, la capa de entrada se puede observar en la parte izquierda de la figura 2.4. Esta primera capa está formada por las neuronas que introducen las entradas en la red, esta capa no produce ningún tipo de procesamiento sobre los datos de entrada. La siguiente capa que podemos observar en la figura justamente conectada con la primera es la llamada capa oculta. En este ejemplo podemos observar una red con 2 capas ocultas. En estas capas las neuronas procesan la información, la condición imprescindible de este tipo de capas es que tengan una capa antecesora y otra predecesora. Por último tenemos una capa de salida, cuyos valores de salida se corresponden con las salidas de toda la red.

La salida de cada neurona será multiplicada por el peso de conexión que le corresponde con la siguiente capa. Estos pesos irán variando en cuanto comience el aprendizaje.

Para crear una red neuronal es necesario saber qué parámetros se utilizarán para su creación. Por un lado, están los parámetros a nivel estructural, como pueden ser: el número de capas que se van a utilizar, el número de neuronas que tendrá cada capa y los tipos de conexiones. Los tipos de conexiones, por norma general, recibirán las señales de la capa anterior y estas enviarán su salida a las neuronas de la capa posterior. Pero, por otro lado, tenemos otro tipo de parámetros que afectan a la red, como el tipo de aprendizaje, que puede ser supervisado o no supervisado.

En cuanto al tipo de aprendizaje, la mayor diferencia entre el supervisado y el no supervisado es que en el caso del supervisado existe una componente con los datos de entrada y otra con los resultados deseados; sin embargo, en el caso del aprendizaje no supervisado, solo existe la componente de datos de entrada.

### 2.6.3 Perceptron multicapa

El Perceptron Multicapa [16] [17] [18] es un tipo de red de neuronas artificiales (RNA), formada por varias capas y estas al mismo tiempo por varias neuronas, donde cada salida de una neurona de la capa  $i$  es entrada de todas las neuronas de la capa  $i + 1$ .

Esta RNA será usada para realizar la clasificación con los ratios más relevantes proporcionados por ED en combinación con K-NN. De este modo se podrá corroborar si un clasificador más robusto como un clasificador neuronal provee buenos resultados de clasificación en comparación con K-NN y con respecto a los resultados al utilizar todo el conjunto de ratios.

El algoritmo de aprendizaje utilizado en esta red será el conocido como *backpropagation* (propagación hacia atrás de errores o retropropagación). Este tipo de aprendizaje se basa en minimizar la función de error producida por la red, típicamente el error cuadrático medio entre las salidas de la red y las salidas deseadas. La salida de la red se compara con la salida deseada, y se calcula una señal de error. Los errores en las salidas se propagan hacia atrás, como el propio nombre indica, desde la capa de salida hacia la capa previa que está directamente conectada a esta, modificando el valor numérico de los pesos de conexión en la dirección que minimiza el error de la red.

Como es bien conocido, una vez entrenada la red para reconocer las asociaciones entre entradas y salidas proporcionadas en los ejemplos de entrenamiento, la RNA tiene 2 propiedades importantes, como son la tolerancia a ruido de entrada y buena generalización a entradas no vistas en el entrenamiento.

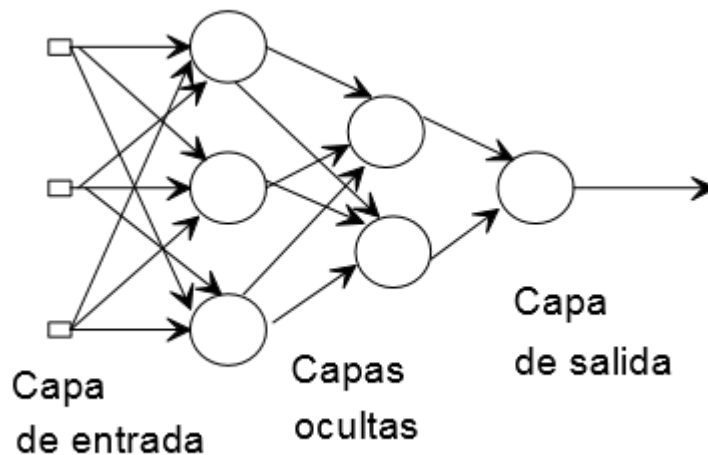


Figura 2.4: Representación gráfica del Perceptrón Multicapa.

Para la creación del Perceptrón Multicapa [16] [17] [18] en Matlab se utiliza el paquete “nntool”. Este paquete proporciona una serie de opciones para poder indicar los parámetros a través de los cuales se creará la red neuronal. Como parámetros principales podemos seleccionar los datos de entrada y los datos objetivo; una vez seleccionados estos datos podremos ya configurar nuestra red.

En este caso, la red neuronal que se utilizará en este proyecto para las pruebas será la *patternet*, una subclase dentro de las *feed-forward*, especializada en el reconocimiento de patrones.

Las características aplicadas a este tipo de red son las siguientes:

- La división de los datos utilizados para modelizar es aleatorio. En nuestro caso se dividirá el fichero de entrenamiento en un 80% de datos de entrenamiento y en un 20% de datos de validación, para evitar sobreaprendizaje (el fichero de entrenamiento y el de testeo se detallan en el próximo capítulo).
- Las funciones de activación de las neuronas son de tipo sigmoide, `tansig()` en Matlab. Ya se ha explicado en la sección 2.6.1 por qué se han utilizado este tipo de funciones de activación.
- La salida de la red tendrá valores entre 0 y 1. Los valores menores que 0.5 corresponderán a empresa clasificada como no fracasada, mientras valores mayores o iguales a 0.5 indicarán empresa predicha o clasificada como fracasada.

Además, esta red será definida con una función de entrenamiento “`trainscg`” (Scaled Conjugate Gradient), función de entrenamiento que actualiza los valores de peso y umbrales de acuerdo con el método de gradiente conjugado escalado. La etapa de entrenamiento terminará cuando se cumpla cualquiera de las siguientes condiciones: se llega al número máximo de iteraciones de entrenamiento, se sobrepasa el límite de tiempo, se minimiza lo suficiente la función de rendimiento como para llegar al objetivo, el gradiente es menor al mínimo establecido o la función de rendimiento se ha incrementado más veces que el máximo establecido.

Como función de rendimiento se utilizará “`mse`” (mean squared error). Como ya se ha comentado, el MSE representa el error cuadrático medio entre las salidas de la RNA y las salidas deseadas.

Como ya se comentó anteriormente, también se pueden incluir las capas ocultas que se deseen, además de modificar el número de neuronas de cada capa. En este proyecto, las pruebas se harán con dos capas ocultas y el número de neuronas será elegido a través de tres tipos de progresiones: aritmética, geométrica y media (entre el número de neuronas de entrada y de salida)

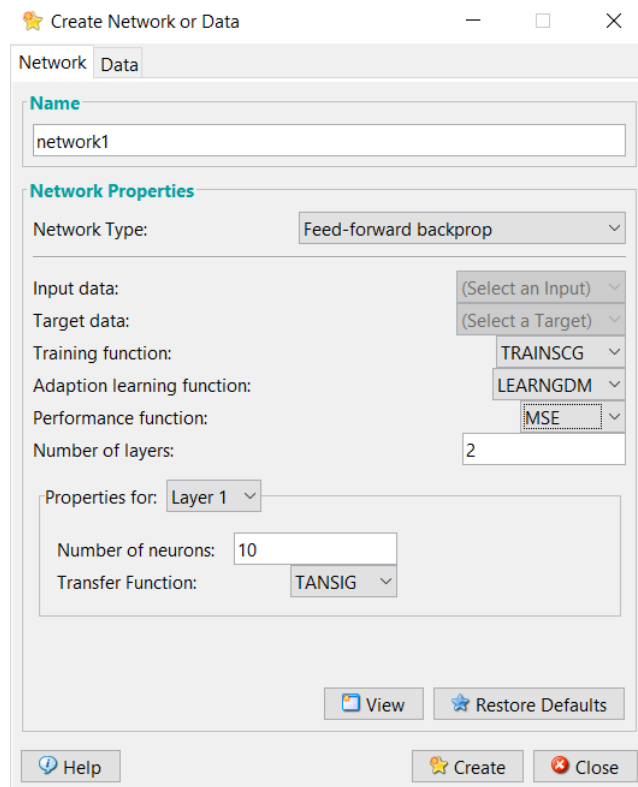


Figura 2.5: Parámetros utilizados para las pruebas de la red neuronal.

En la figura 2.5 podemos observar una interfaz gráfica que nos facilita el paquete “nntool”.

## 2.7 Medidas de clasificación

La matriz de confusión es la herramienta que permite la visualización del desempeño de un algoritmo de clasificación. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real. Uno de los beneficios de las matrices de confusión es que facilitan ver si el sistema está confundiendo dos clases.

		<b>Predicción</b>	
		<b>Positivos</b>	<b>Negativos</b>
<b>Observación</b>	<b>Positivos</b>	Verdaderos Positivos (VP)	Falsos Negativos (FN)
	<b>Negativos</b>	Falsos Positivos (FP)	Verdaderos Negativos (VN)

Figura 2.6: Matriz de confusión.

Normalmente, una vez obtenidos los resultados de las matrices de confusión se utilizan una serie de medidas de clasificación para poder interpretar los datos que obtenemos en el conjunto de la matriz. Las medidas más utilizadas son las mostradas en la siguiente lista:

- Exactitud =  $(VN + VP)/(VN + FP + FN + VP)$
- Precisión =  $VP/(FP + VP)$
- Sensibilidad =  $VP/(VP + FN)$
- Especificidad =  $VN/(VN + FP)$

donde los  $VN$  serán los verdaderos negativos,  $VP$  los verdaderos positivos,  $FN$  los falsos negativos y  $FP$  los falsos positivos. En nuestro caso, la matriz de confusión estará formada por una clase fracaso, correspondiente a la clase “positivos” en la figura 2.6 y otra clase llamada no fracaso correspondiente a la clase “negativos” en la figura 2.6. En es trabajo se hará hincapié en una determinada medida de las listadas anteriormente, la llamada sensibilidad. Esta cobra un valor más importante que el resto, debido a que el objetivo más importante en la aplicación es predecir el futuro fracaso de las empresas. Esta medida se maximiza cuanto menor es el número de falsos negativos ( $FN$ ), es decir, un menor número de empresas que se predicen que no fracasarán pero que en realidad fracasan.





# Selección de la muestra de datos de empresas

---

## 3.1 Datos

LA información económico-financiera es la principal fuente de datos para el análisis de la empresa, y por tanto, se está tratando la situación del dato financiero como el centro de las decisiones estratégicas. En este caso se utilizan los datos para la predicción de resultados financieros que dicha empresa tendrá a corto plazo (a un año en el futuro). Estos datos serán ratios financieros que nos permiten entender el estado de la empresa en un determinado momento.

### 3.1.1 Selección de la población

Para la realización del trabajo se utiliza un conjunto de pymes gallegas, excluyendo financieras, seguros, constructoras e inmobiliarias, debido a ciertas peculiaridades que presentan este tipo de sectores. La escala temporal se sitúa entre el año 2004 y el 2013. La información contable de las empresas se ha obtenido a través de la base de datos SABI (Sistema de Análisis de Balances Ibéricos) de la compañía Informa, SA. Por otro lado, se ha hecho uso de información del Registro Público Concursal y de empresas especializadas en informes empresariales para obtener la información legal.

Para determinar si una empresa cumple con los requisitos necesarios para ser una PYME se ha utilizado el criterio del Boletín Oficial del Estado del 12 de octubre de 2013, que ha publicado la circular del Banco de España que recoge la modificación del concepto de PYME para las entidades de crédito. Las reglas dictaminadas por el Boletín Oficial agruparán las empresas dependiendo del tamaño y serán las descritas a continuación:

1. En la categoría de las PYME, se define a una mediana empresa como una empresa que ocupa a menos de 250 personas y cuyo volumen de negocios anual no excede de 50 millones de euros o cuyo balance general anual no excede de 43 millones de euros.
2. En la categoría de las PYME, se define a una pequeña empresa como una empresa que ocupa a menos de 50 personas y cuyo volumen de negocios anual o cuyo balance general anual no supera los 10 millones de euros.
3. En la categoría de las PYME, se define a una microempresa como una empresa que ocupa a menos de 10 personas y cuyo volumen de negocios anual o cuyo balance general anual no supera los 2 millones de euros.

Estas distinciones entre los diferentes subgrupos dentro de las PYMES se pueden visualizar de una forma más clara en la siguiente figura:


 Categoría de empresa	Efectivos: Unidades de trabajo anual	Volumen de negocios anual	ó Balance general anual
 Medianas	< 250	≤ 50 millones de euros	ó ≤ 43 millones de euros
 Pequeñas	< 50	≤ 10 millones de euros	ó ≤ 10 millones de euros
 Microempresas	< 10	≤ 2 millones de euros	ó ≤ 2 millones de euros

Figura 3.1: Categorización de los diferentes tipos de PYMES

### 3.1.2 Criterio del fracaso

Se han comentado anteriormente, en este trabajo, los diferentes tipos de fracaso que se pueden producir en una empresa (en la sección 1.1.2). Aunque se hayan expuesto los tipos de fracaso, falta definir qué tipo o tipos de fracaso se considerarán en este trabajo, como fracasos verdaderamente aplicables a las empresas.

Los tipos de fracasos considerados serán los de tipo económico y legal excluyendo el financiero. En este trabajo no se podrá aplicar el fracaso financiero debido a que dicha información no es accesible si no es bajo pago y/o para cierto tipo de entidades y condiciones.

Dicho esto, se ha optado por las siguientes reglas para clasificar a una empresa como empresa fracasada. En cuanto al fracaso de tipo legal, se comprobará si está o ha estado en concurso de acreedores, si se ha producido una declaración judicial de insolvencia, o si se ha producido una declaración de crédito incobrable por parte de organismos públicos. Por otro lado, en el caso del fracaso de tipo económico, solo si la sociedad sitúa su patrimonio por debajo de la mitad de su capital social, se clasificará a la empresa como fracasada.

### **3.1.3 Selección de los ratios**

Los ratios serán seleccionados, como ya se ha comentado anteriormente en la sección 1.2.1, basándose en su la popularidad literaria contable y en la significación en aquellos estudios más relevantes en cuanto al fracaso empresarial.

Cabe destacar que los ratios seleccionados se han obtenido del balance de situación y de la cuenta de resultados anual. Ninguno de ellos se ha obtenido mediante algún otro criterio contable alternativo. Los ratios seleccionados para el análisis son los mostrados en el apartado 3.1.4.

### 3.1.4 Ratios financieros utilizados en el proyecto

Referencia	Ratio	Medida Financiera
ACT01	Gastos financieros / Valor agregado	ACTIVIDAD
ACT02	Gastos personal / Activo fijo	ACTIVIDAD
ACT03	Gastos personal + Dotación amortización inmovilizado / Valor agregado	ACTIVIDAD
ACT04	Ingresos explotación / Gastos explotación	ACTIVIDAD
ACT05	Valor agregado / Importe neto cifra de ventas	ACTIVIDAD
APL01	Gastos financieros / EBIT	APALANCAMIENTO
APL02	Gastos financieros / Exigible total	APALANCAMIENTO
APL03	Gastos financieros / Resultados explotación	APALANCAMIENTO
APL04	Resultado ejercicio / Exigible total	APALANCAMIENTO
END01	Exigible total / Fondos propios	ENDEUDAMIENTO
END02	Fondos propios - Resultado ejercicio / Exigible a corto	ENDEUDAMIENTO
END03	Exigible a largo / Exigible total	ENDEUDAMIENTO
EST01	Capital suscrito / Fondos propios	ESTRUCTURA
EST02	Activo circulante / Activo total	ESTRUCTURA
EST03	Dotación amortización inmovilizado / Inmovilizado	ESTRUCTURA
EST04	Capital circulante / Activo total	ESTRUCTURA
EST05	Capital circulante / Exigible total	ESTRUCTURA
EST06	Capital circulante / Importe neto cifra de ventas	ESTRUCTURA
EST07	Otros activos líquidos / Activo total	ESTRUCTURA
EST08	Resultado ejercicio / Capital circulante	ESTRUCTURA
EST09	Medida descomposición del activo	ESTRUCTURA
LIQ01	Cash flow operativo / Activo total	LIQUIDEZ
LIQ02	Cash flow operativo / Exigible total	LIQUIDEZ
LIQ03	Cash flow operativo / Exigible a corto	LIQUIDEZ
LIQ04	Cash flow operativo / Importe neto cifra de ventas	LIQUIDEZ
LIQ05	Cash flow / Activo total	LIQUIDEZ
LIQ06	Cash flow / Exigible total	LIQUIDEZ
LIQ07	Cash flow / Exigible a corto	LIQUIDEZ
LIQ08	Cash flow / Importe neto cifra de ventas	LIQUIDEZ

Cuadro 3.1: Ratios financieros utilizados en el proyecto.

Referencia	Ratio	Medida Financiera
LIQ09	Otros activos líquidos / Exigible a corto	LIQUIDEZ
LIQ10	Existencias / Exigible a corto	LIQUIDEZ
LIQ11	Existencias + Deudores / Exigible a corto	LIQUIDEZ
LIQ12	Intervalo sin crédito	LIQUIDEZ
LIQ13	Deudores / Exigible a corto	LIQUIDEZ
REN01	EBIT / Activo total	RENTABILIDAD
REN02	EBIT / Importe neto cifra de ventas	RENTABILIDAD
REN03	Resultado ejercicio / Importe neto cifra de ventas	RENTABILIDAD
REN04	Resultado ejercicio - Deudores - Existencias / Activo total	RENTABILIDAD
REN05	Resultado ejercicio / Activo total	RENTABILIDAD
REN06	Resultado ejercicio / Fondos propios	RENTABILIDAD
ROT01	Activo circulante - Existencias / Importe neto cifra de ventas	ROTACION
ROT02	Existencias / Importe neto cifra de ventas	ROTACIÓN
ROT03	Importe neto cifra de ventas / Ingresos de explotación	ROTACIÓN
ROT04	Importe neto cifra de ventas / Activo circulante	ROTACIÓN
ROT05	Importe neto cifra de ventas / Inmovilizado	ROTACIÓN
ROT06	Importe neto cifra de ventas / Activo total	ROTACIÓN
ROT07	Importe neto cifra de ventas / Capital circulante	ROTACIÓN
ROT08	Otros activos líquidos / Importe neto cifra de ventas	ROTACIÓN
SOL01	Activo circulante - Existencias / Exigible a corto	SOLVENCIA
SOL02	Activo circulante / Exigible total	SOLVENCIA
SOL03	Activo circulante / Exigible a corto	SOLVENCIA
SOL04	Inmovilizado / Fondos propios	SOLVENCIA
SOL05	Exigible total / Activo total	SOLVENCIA
SOL06	Fondos propios / Activo total	SOLVENCIA
SOL07	Fondos propios / Inmovilizado	SOLVENCIA
SOL08	Exigible a corto / Activo total	SOLVENCIA
SOL09	Resultados ordinarios antes impuestos / Exigible a corto	SOLVENCIA
TES01	Tesorería / Exigible a corto	TESORERÍA
TES02	Tesorería / Importe neto cifra de ventas	TESORERÍA

Cuadro 3.2: Ratios financieros utilizados en el proyecto.

Como se puede observar en las 2 tablas anteriores, los ratios están agrupados en diferentes medidas financieras. Esta agrupación se hace debido a que cada ratio perteneciente a una medida financiera tiene una finalidad similar a los otros ratios que pertenecen a la misma. A continuación, se muestra una breve puntualización de cada grupo:

- **Ratios relativos a la actividad (ACT01-ACT05):** Este tipo de ratios están vinculados al volumen de operaciones realizadas de una empresa.
- **Ratios de apalancamiento (APL01-APL04):**
  - **Apalancamiento operativo:** Es la capacidad de las empresas de emplear costos fijos de operación para aumentar al máximo los efectos de cambios en las ventas sobre el resultado antes de intereses e impuestos.
  - **Apalancamiento financiero:** Al contrario que el apalancamiento operativo, en el apalancamiento financiero sí se tienen en cuenta los costes financieros (impuestos, tasas, etc.).
- **Ratios de endeudamiento (END01-END03):** Los ratios de endeudamiento se encargan de medir el volumen de la financiación ajena del que hace uso la compañía.
- **Ratios de estructura (EST01-EST09):** Se encargan de medir la estructura porcentual del activo y del pasivo.
- **Ratios de liquidez (LIQ01-LIQ13):** Aportan una idea de si una empresa será capaz de pagar sus deudas al vencimiento.
- **Ratios de rentabilidad (REN01-REN06):** A partir de ellos se puede tener una idea de si la empresa genera unos ingresos suficientes para cubrir los costes y remunerar a sus propietarios.
- **Ratios de rotación (ROT01-ROT08):** Cuantifican el rendimiento de una empresa en un período determinado de tiempo.
- **Ratios de solvencia (SOL01-SOL09):** Miden la habilidad para hacer frente a las obligaciones financieras tanto a corto como a largo plazo.
- **Ratios de tesorería (TES01-TES02):** Miden la habilidad para hacer frente a las obligaciones financieras a corto plazo.

### 3.1.5 Conjunto de datos utilizados en el proyecto

Los datos de las empresas se almacenan en 2 ficheros, un fichero llamado matModel y otro matTest. En el matModel se encuentran los datos o registros de 274 empresas, 137 de ellas que fracasan, y otras 137 que no fracasan. Este fichero será el fichero de entrenamiento, utilizado para contrastar los datos y sacar conclusiones contra el fichero matTest. Un “registro” consiste en los datos de una empresa en un ejercicio económico (con su componente categoría de fracaso/no fracaso a 1 año en el futuro). Es decir, una misma empresa puede aportar diferentes registros correspondientes a diferentes ejercicios anuales.

En el fichero `matTest` existen 18360 registros, de los cuáles solo utilizaremos inicialmente 272 registros, debido a la diferencia que habría entre el número de empresas fracasadas y no fracasadas. De las 272, la mitad (136) serán empresas fracasadas y la otra mitad empresas no fracasadas, teniendo de este modo una muestra balanceada en las empresas de ambas categorías. Este fichero `matTest` es el que se usará como test en el clasificador K-NN, mientras `matModel` representa el fichero de entrenamiento con empresas (igualmente balanceadas) categorizadas correctamente.

Es importante destacar que, aunque utilizaremos en casi todas las pruebas este subconjunto de 272 empresas del fichero `matTest`, en la prueba realizada con la red neuronal se utilizarán la totalidad de los registros de este fichero, es decir, los 18360. Además, los resultados de las matrices de confusión se obtendrán tanto para el conjunto `matTest` reducido (272 registros) como para el conjunto `matTest` completo (18360 registros).

### 3.1.6 Normalización de los datos

Los datos extraídos de estos dos ficheros serán transformados en un proceso de normalización debido a la diferencia y variación de valores entre los diferentes ratios. La normalización, para cada ratio individual, ha sido efectuada a través de la siguiente fórmula:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

La normalización se hará conjunta, es decir, utilizaremos los datos de los dos ficheros (`matModel` y `matTest`), donde  $X$  es el valor de un determinado ratio,  $X_{min}$  el valor más bajo de ese ratio en ambos ficheros, y  $X_{max}$  el valor más alto de ese ratio en ambos ficheros. Por tanto, se normalizarán todos los datos convirtiéndolos en valores entre 0 y 1.





# Resultados

---

EN este capítulo se explicarán las diferentes pruebas realizadas y los resultados obtenidos a través de estas. Se explicará todo el proceso desde las pruebas iniciales hechas para la selección y diferenciación de la calidad de los ratios hasta las últimas utilizando la combinación entre K-NN y ED y, finalmente, pruebas con redes neuronales artificiales.

### 4.1 Primeras pruebas de obtención de calidad de los ratios

La primera prueba que se ha hecho, una vez normalizados los ratios, es la obtención de la relevancia de cada ratio. Cada ratio tiene una relevancia que es determinada por diferentes factores. Para calcularla utilizaremos primeramente los métodos de selección mediante filtros estadísticos simples, como se ha explicado previamente en el capítulo 2, sección 2.1.

Dependiendo de los resultados obtenidos se podrá observar cuáles de los ratios podrían ser mejores para la clasificación. Cuanto mayores sean los valores obtenidos por el método T-estadístico y *Fisher Score*, mejores serán los ratios. En el caso de las puntuaciones de T-estadístico, cabe destacar que los resultados se mostrarán en valor absoluto. Se tiene que tener en cuenta que los valores resultantes tanto del T-estadístico como del *Fisher Score* se obtendrán del fichero de entrenamiento (matModel). Los resultados que se obtienen con el T-estadístico y el *Fisher Score* son los mostrados a continuación:

4.1. Primeras pruebas de obtención de calidad de los ratios

Referencia	Valor T-estadístico	Valor Fisher Score
ACT01	0,241787798	7,59683E-05
ACT02	0,776217449	5,84208E-05
ACT03	0,425028449	8,43394E-06
ACT04	10,746903	20,14143376
ACT05	3,575605463	0,185276575
APL01	5,322119505	0,03844481
APL02	3,392481537	11,47271064
APL03	5,322119505	0,03844481
APL04	10,33125685	5,894957846
END01	0,105460741	4,23312E-09
END02	4,194261302	0,009175629
END03	0,112651596	3,72505E-07
EST01	2,327860693	0,328372009
EST02	2,606317436	0,35655926
EST03	0,380047529	0,012850715
EST04	9,779071295	3,277909741
EST05	7,475944134	0,118472819
EST06	4,733233484	0,096620495
EST07	0,814692233	5,788314314
EST08	5,797735774	1,59996E-05
EST09	3,396394852	6,954517764
LIQ01	2,78770737	0,212670128
LIQ02	5,029691823	0,580490752
LIQ03	4,185418537	0,161691835
LIQ04	2,056877012	0,008282508
LIQ05	7,940332042	2,287225019
LIQ06	10,49962004	4,53325473
LIQ07	9,694682655	1,870751293
LIQ08	3,097025748	0,02217973

Cuadro 4.1: Valores T-estadístico y *FisherScore*.

Referencia	Valor T-estadístico	Valor Fisher Score
LIQ09	5,129726238	0,057771167
LIQ10	2,824010881	0,141716932
LIQ11	5,730160498	0,167357573
LIQ12	5,961454088	0,250572537
LIQ13	5,233161119	0,184871709
REN01	9,785417872	7,090801916
REN02	2,865022205	0,017151578
REN03	3,20924399	0,021738101
REN04	5,584025031	0,695016739
REN05	8,124684661	2,323233209
REN06	0,628579987	4,65312E-06
ROT01	0,363628389	0,000240489
ROT02	0,535723806	0,001784113
ROT03	0,121201268	0,030896291
ROT04	0,294769948	7,77661E-06
ROT05	0,008256055	9,94615E-12
ROT06	0,740553036	0,000987087
ROT07	1,940533298	4,69674E-07
ROT08	2,249138426	0,049739653
SOL01	6,453485143	0,058052426
SOL02	7,719081095	0,115976448
SOL03	6,867983215	0,062119092
SOL04	1,091127722	1,42354E-07
SOL05	11,6893482	4,682932777
SOL06	11,6893482	4,682932777
SOL07	1,617731096	4,85952E-06
SOL08	7,395974642	1,570050422
SOL09	10,37399307	1,886977317
TES01	6,383760596	0,511131771
TES02	2,204155764	0,071039857

Cuadro 4.2: Valores T-estadístico y *FisherScore*.

Para poder observar mejor los datos se muestra una figura con los 59 ratios en el eje  $X$  y con los valores T-estadístico en el eje  $Y$ , y otra con el mismo eje  $X$ , pero con los valores del eje  $Y$  correspondientes a los valores del Fisher Score. Con esta figura podemos observar cuáles de los valores obtenidos a través de los métodos de selección son los mejores en cuanto a relevancia de ratio.

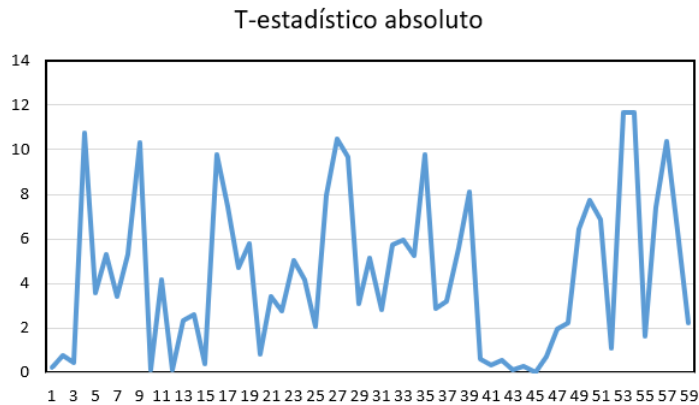


Figura 4.1: Valores obtenidos con la aplicación del método de selección T-estadístico.

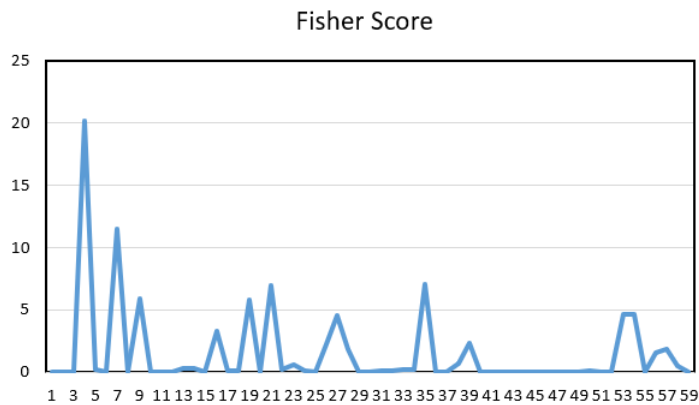


Figura 4.2: Valores obtenidos con la aplicación del método de selección *Fisher Score*.

Como podemos observar, los ratios 53 (SOL05) y 54 (SOL06) son los mejores ratios correspondientes a la primera figura, ya que son los picos más altos de la figura, con una calidad de un 11.68, precedidos por el 4 (ACT04) y el 27 (LIQ06). En la segunda figura se visualiza el valor *Fisher Score*, presentando mayormente una alta correlación en valores altos en el T-estadístico. El mejor valor es el ratio número 4 (ACT04) con un valor de 20.14, seguido por el 7 (APL02), 35 (REN01) y 21(EST09). Podemos comprobar que de estos cuatro valores, que fueron escogidos a modo de ejemplo debido a su alto valor de *Fisher Score*, solo coincide el

ratio número 4 (ACT04) en los dos casos (alto valor de ambas medidas estadísticas). Aún solo coincidiendo este valor podemos comprobar que los mejores ratios de la figura que muestra el T-estadístico como picos más altos, coinciden con los picos más altos de la figura que muestra el *Fisher Score* y viceversa. Esto nos indica que, como es lógico, estas dos medidas coinciden en la mayor parte de los casos.

#### 4.1.1 Observaciones y deficiencias del T-estadístico y del *Fisher Score*

En este proyecto lo que se va a intentar hacer es obtener los ratios que más influyan en el fracaso empresarial. Esta primera prueba de obtención de relevancia de los ratios nos da una aproximación, para ver cuáles son los principales ratios con buena relevancia, pero esto no es suficiente para poder decir con certeza que solo los ratios con un valor alto en las medidas estadísticas afectan al éxito o fracaso de la empresa. Como se ha comentado anteriormente en el capítulo 2, “Métodos utilizados”, estos dos métodos de selección (T-estadístico y *Fisher Score*) se agrupan en los llamados *Filter methods*. Por lo tanto, no se está utilizando ningún algoritmo de aprendizaje que pueda obtener un resultado más preciso y objetivo. Por esta razón, se necesitará aplicar algún algoritmo de aprendizaje para mejorar los resultados, y obtener el menor número de fallos y la mayor fiabilidad.

Dicho esto, se utilizará como algoritmo de aprendizaje una combinación de dos algoritmos, la combinación se hará entre el clásico método de clasificación K-NN (*K-nearest neighbors*) y el algoritmo de Evolución Diferencial (ED). Gracias a la combinación entre estos dos algoritmos tendremos un método que ayudará a hacer una mejor selección de los ratios.

## 4.2 Pruebas de obtención de la relevancia de los ratios a través del algoritmo formado por K-NN y ED

Una vez hecha una pequeña prueba sobre la relevancia de los ratios a través de los *Filter methods* usados, se aplicará el algoritmo híbrido, formado por el clasificador K-NN y por el algoritmo ED. Se han realizado diferentes pruebas, y, aunque se han probado con diferentes valores en los parámetros, las pruebas que ofrecen un mejor resultado son las 6 pruebas que se explicarán a continuación.

Cabe destacar que cada ejecución del algoritmo siempre estará programada para realizar 500 generaciones, con una probabilidad de cruce (CR) de un 0.1, y con una  $F$  que tomará valores aleatorios entre 0.1 y 0.9, cuyo valor será aplicado a la función de mutación, mostrada en la línea 6 del Algoritmo 1, sección 2.3. Las pruebas estarán siempre realizadas con una población de 100 individuos. Los únicos factores que variarán en la ejecución de las pruebas serán: la

$K$ , que indicará el número de vecinos con los que hará la clasificación el algoritmo K-NN; y los ratios aplicados para la prueba. Primeramente se harán pruebas con todos los ratios, es decir, con los 59, y después se probarán ejecuciones con los 30 mejores ratios del T-estadístico y del *Fisher Score* para comprobar si introduciendo como entrada del programa ratios con una primera fase de filtrado, se obtienen mejores resultados. En este caso, el primer filtrado se produciría con los métodos de selección anteriormente mencionados.

Los valores que se obtienen como resultado serán siempre la media de la calidad de la población en cada generación, la mejor calidad de cada generación y la mejor instancia (ratios seleccionados) de cada ejecución. En nuestras pruebas cada ejecución siempre se corresponderá con 500 generaciones.

Se harán en todas las pruebas 10 ejecuciones independientes del algoritmo ED/K-NN, y, una vez obtenidos los resultados de cada una de ellas, se obtendrá un promedio de todos los valores de cada ejecución para mostrar valores de calidad promediados en los gráficos.

#### **4.2.1 Prueba con los 59 ratios y con $K=15$**

Esta es la primera de las pruebas que se va a analizar. El valor de  $k$  es 15, lo que quiere decir que, en principio, se debería obtener una buena clasificación, ya que el número de “vecinos” utilizados es alto. Por otro lado, hay que tener en cuenta que no se hace ningún tipo de filtrado, en cuanto a que se utiliza todo el conjunto de ratios (59) para la selección automática de los más relevantes. Los resultados obtenidos en esta prueba son los siguientes:

Los gráficos corresponden al promedio de 10 ejecuciones independientes del algoritmo evolutivo.

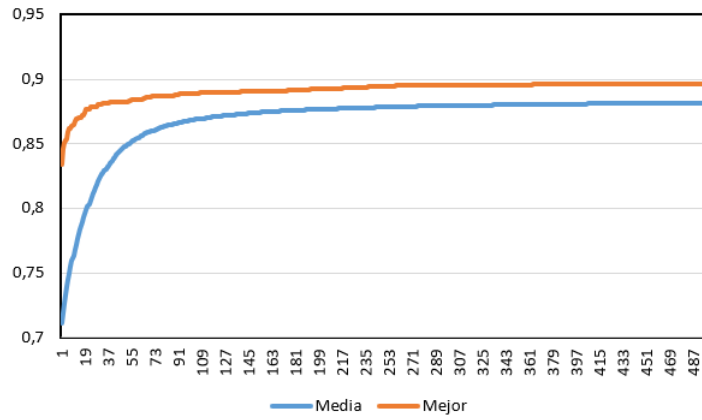


Figura 4.3: Calidad media (porcentaje de aciertos) y calidad del mejor individuo a lo largo de generaciones, usando el conjunto total de 59 ratios en la selección y  $k=15$ .

#### 4.2.2 Prueba con los 59 ratios y con $K=3$

En esta prueba se utilizan también los 59 ratios. El valor de  $k$  es igual a 3, usando, por tanto, un número reducido de vecinos. En cuanto al filtrado, no estará presente tampoco en esta, ya que la única diferencia entre esta y la anterior es el número de vecinos en el algoritmo de clasificación.

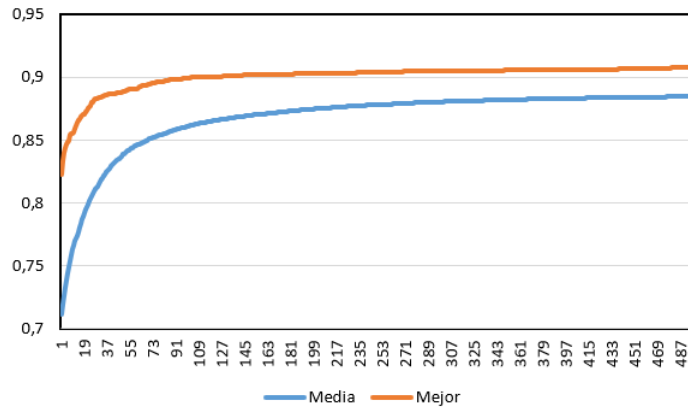


Figura 4.4: Calidad media (porcentaje de aciertos) y calidad del mejor individuo a lo largo de generaciones, usando el conjunto total de 59 ratios en la selección y  $k=3$ .

Analizando las figuras de estas dos primeras pruebas no observamos casi ningún tipo de diferencia, ni siquiera en el rápido incremento de las calidades en las generaciones iniciales del proceso evolutivo. Tanto la evolución de las mejores calidades, como la evolución de la media de las calidades, son prácticamente iguales en los dos casos. Por tanto, no se ha obtenido un



beneficio cuantitativo a la hora de aumentar la variable  $k$  en el algoritmo de clasificación.

### 4.2.3 Prueba con los 30 mejores ratios de Fisher Score y con K=15

En esta prueba el algoritmo evolutivo parte de los 30 mejores ratios obtenidos con el filtrado que proporciona *Fisher Score*. Por tanto, el algoritmo evolutivo elige el subconjunto de ratios más óptimo (de acuerdo al clasificador K-NN) de entre esos 30. Los resultados obtenidos son los siguientes:

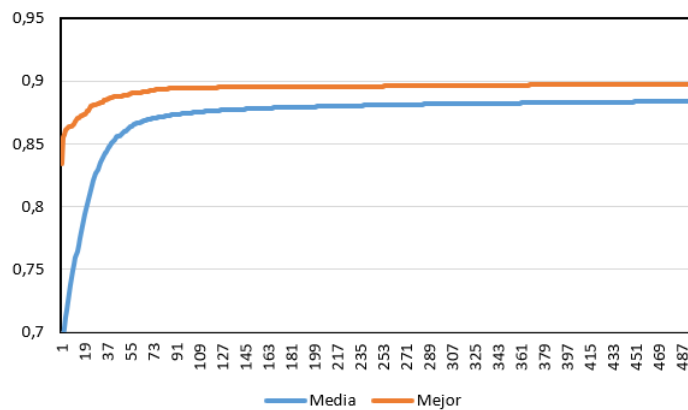


Figura 4.5: Calidad media y mejor a lo largo de las generaciones, usando los 30 mejores ratios de *Fisher Score* en la selección y  $k=15$ .

La gráfica que se obtiene al hacer esta prueba, sí cambia en comparación con la anterior con  $k=15$ . Comparando la evolución de calidades entre la figura 4.5 y 4.3, la evolución de la calidad media es mejor en las primeras generaciones. Esto es lógico, ya que el algoritmo evolutivo está seleccionando entre ratios que ya se presuponen apropiados para el clasificador. Sin embargo, la evolución similar de la calidad del mejor individuo indica que no se va a obtener una ventaja substancial.

### 4.2.4 Prueba con los 30 mejores ratios de Fisher Score y con K=3

Esta prueba es similar a la prueba 4.2.3, la única diferencia es el número de vecinos, se probará con un  $k$  más pequeño para observar si se producen diferencias en cuanto a la figura 4.5, donde  $k$  es igual a 15. Los resultados obtenidos son los siguientes:

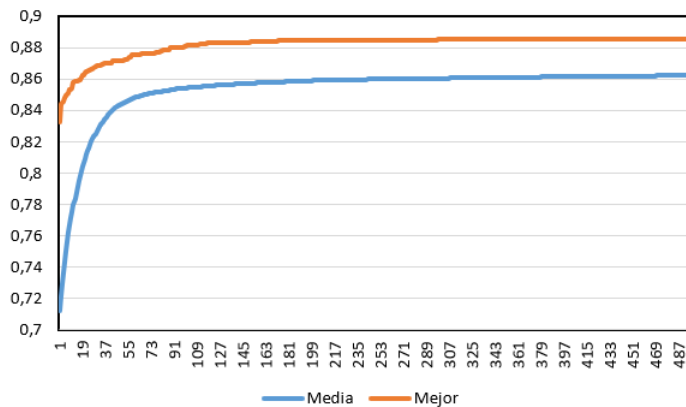


Figura 4.6: Calidad media y mejor a lo largo de las generaciones, usando los 30 mejores ratios de *Fisher Score* en la selección y  $k=3$ .

No existe gran variación entre la evolución de calidades con respecto a utilizar los 30 mejores ratios de *Fisher Score* con  $k=15$  (figura 4.5). De nuevo, la diferencia más notable entre la figura 4.6 y la 4.4 ( $k=3$  pero partiendo de 59 ratios), es la más rápida evolución de la calidad media, por la misma razón anteriormente comentada. Si se puede apreciar un pequeño detalle, ya que los valores máximos obtenidos en comparación con la prueba anterior son peores. En esta figura (4.6) se obtiene un pico máximo de 0.858 en cuanto a la media de calidad y 0.872 en el mejor individuo de la última generación; sin embargo, en la figura 4.5 los valores son de 0.876 y 0.894, en la media y en el mejor individuo respectivamente. Por tanto, el resultado que obtenemos es mejor cuando el K-NN tiene un mayor número de vecinos debido a que mejora la calidad en ambos apartados.

#### 4.2.5 Prueba con los 30 mejores ratios de T-estadístico y con $K=15$

Como ya se ha comentado en apartados anteriores, aparte del método de selección *Fisher Score*, se aplicará el método basado en las puntuaciones obtenidas del T-estadístico y se realizará el mismo proceso que en las pruebas 4.2.3 y 4.2.4. Primeramente se seleccionarán los 30 mejores valores del T-estadístico para saber qué 30 ratios se van a incluir como entrada en la ejecución del programa. Una vez ejecutado el proceso de selección evolutivo y calculada la media de las 10 ejecuciones obtenemos la siguiente gráfica:

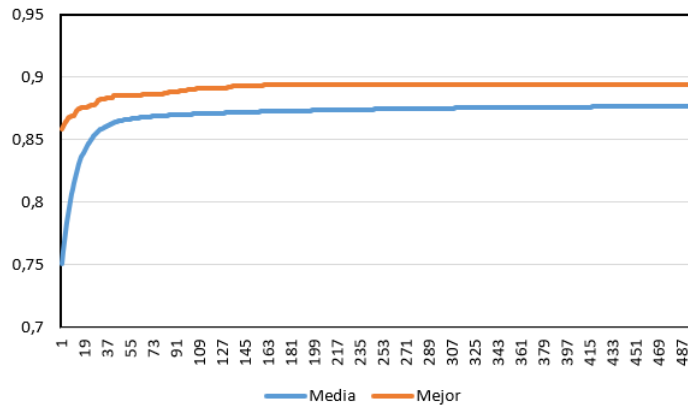


Figura 4.7: Calidad media y mejor a lo largo de las generaciones, usando los 30 mejores ratios de T-estadístico en la selección y  $k=15$ .

Esta gráfica puede ser considerada como la mejor gráfica obtenida hasta el momento, ya que supera al resto de pruebas en cuanto a las calidades obtenidas. Se muestran unos valores iniciales en la primera generación bastante más altos que en el resto de casos, lo que indica que las poblaciones iniciales son mejores. Podemos deducir, a partir de esto, que en nuestros ejemplos el T-estadístico selecciona unos ratios mejores que el *Fisher Score*, ya que las primeras poblaciones parten de una calidad más alta, lo que indica que los ratios de entrada son más relevantes.

Por otro lado, se obtiene una de las mejores puntuaciones en la poblaciones finales, solo superada por las pruebas con 59 ratios. Esto puede parecer no muy lógico, ya que en este caso no estamos obteniendo un primer filtrado para mejorar los resultados, pero puede ser posible ya que el algoritmo juega con combinaciones de un mayor número de ratios. De nuevo, la curva de crecimiento de calidad media es más pronunciada comparada con las anteriores, es decir, obtenemos el mejor crecimiento de calidad de todas las pruebas, y esto se debe al buen filtrado que se hace al comienzo de la ejecución.

#### 4.2.6 Prueba con los 30 mejores ratios de T-estadístico y con $K=3$

Como resultado de las 10 ejecuciones independientes del proceso evolutivo se muestra la siguiente figura que contiene el promedio de cada generación para todas las ejecuciones.

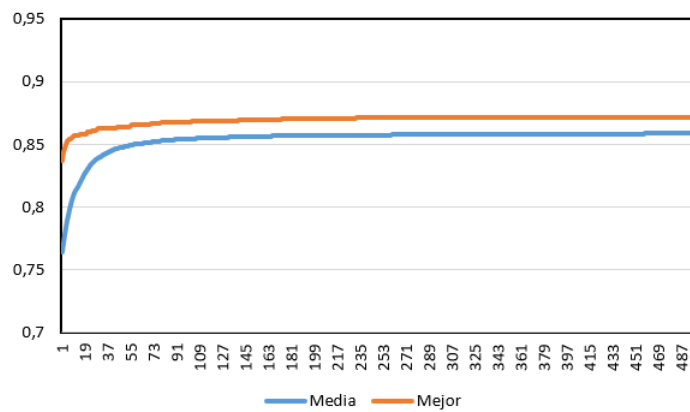


Figura 4.8: Calidad media y mejor a lo largo de las generaciones, usando los 30 mejores ratios de T-estadístico en la selección y  $k=3$ .

Debido a la mala clasificación que se produce por el K-NN, se puede observar que esta gráfica empeora en todos los detalles relevantes si la comparamos con la anterior. La última generación tiene peores resultados en la media de las calidades y en la media de los mejores individuos de cada ejecución. Además se puede observar cómo la pendiente no es tan pronunciada en las primeras generaciones de la gráfica, en comparación a la prueba 4.2.5. Esto último se debe a que se producen más errores en la clasificación, y por tanto, se obtiene un menor crecimiento en cuanto a la calidad.

### 4.3 Tabla resumen de todas las ejecuciones, generaciones y porcentajes de aciertos

En la siguiente tabla se muestra a modo de resumen los valores obtenidos en diferentes generaciones de las gráficas de las pruebas anteriores. Los valores mostrados son los porcentajes de aciertos para cada prueba. Se muestran los valores promedio (de 10 ejecuciones independientes) de la calidad media de la población y de la calidad del mejor individuo/solución. En negrita se remarcan los mejores valores obtenidos.

Tipo de prueba	Generaciones	Media	Mejor
59 ratios y k=15	Gen1	71,12	83,34
	Gen50	84,67	88,27
	Gen100	86,77	88,86
	Gen500	88,17	89,63
59 ratios y k=3	Gen1	71,1	82,27
	Gen50	83,93	88,89
	Gen100	86,11	89,96
	Gen500	<b>88,48</b>	<b>90,77</b>
30 mejores ratios Fisher Score y k=15	Gen1	68,67	83,45
	Gen50	85,97	88,89
	Gen100	87,46	89,48
	Gen500	88,4	89,7
30 mejores ratios Fisher Score y k=3	Gen1	71,22	83,27
	Gen50	84,44	87,2
	Gen100	85,44	88,08
	Gen500	86,23	88,6
30 mejores ratios T-estadístico y k=15	Gen1	78,33	86,5
	Gen50	86,57	88,52
	Gen100	87,03	88,97
	Gen500	87,69	89,44
30 mejores ratios T-estadístico y k=3	Gen1	76,36	83,75
	Gen50	84,82	86,39
	Gen100	85,45	86,8
	Gen500	85,87	87,2

Cuadro 4.3: Porcentajes de acierto de las pruebas en diferentes generaciones.

#### 4.4 Comparativa de porcentajes de aciertos entre $K=3$ y $K=15$

Para obtener una mejor vista de la comparativa de las pruebas en cuanto al número de vecinos utilizados en el K-NN se muestra la siguiente tabla:

Prueba	Número de vecinos (K-NN)			
	3NN		15NN	
	Media	Mejor	Media	Mejor
59 ratios	88,48	90,77	88,17	89,63
30 mejores ratios Fisher Score	86,23	88,6	88,4	89,7
30 mejores ratios T-estadístico	85,87	87,2	87,69	89,44

Cuadro 4.4: Porcentajes de aciertos obtenidos en las pruebas.

De los datos de esta tabla no es posible afirmar cuál es el mejor valor de  $k$ , ya que usando los 59 ratios, en  $k=3$  se obtienen unos valores ligeramente mejores. En los otros dos casos, justo al contrario.

#### 4.5 Resultado de las pruebas con el clasificador K-NN y el algoritmo ED

Además de haber realizado las pruebas anteriores para obtener la calidad media y la mejor calidad de cada generación, también se ha guardado en cada prueba el mejor individuo de cada ejecución. Así se podrá saber cuáles son los ratios más relevantes seleccionados por el algoritmo evolutivo. Al haberse hecho 10 ejecuciones de cada prueba, se obtendrán 10 instancias finales en cada una. Estas instancias (ratios seleccionados) estarán dotadas de la mayor calidad obtenida en las ejecuciones y serán guardadas en un fichero. Las instancias corresponden a los genotipos que codifican cada solución (conjunto de ratios seleccionados). Estos genotipos se muestran ya decodificados, en los que el valor 1 indica que el correspondiente ratio ha sido seleccionado para ser usado por el clasificador K-NN, y 0 lo contrario. A continuación se muestran los resultados para cada prueba en una matriz con los 10 mejores individuos de cada una, para observar y analizar los ratios seleccionados y también los coincidentes en diferentes pruebas. En las pruebas nos fijaremos en los ratios que coinciden en las 10 instancias diferentes, para asegurarse de que se están obteniendo buenos ratios y que son seleccionados en las 10 ejecuciones.

### 4.5.1 Mejores ratios para la prueba con 59 ratios y K=15

El resultado al hacer las 10 ejecuciones de la prueba es el mostrado en la siguiente tabla. Solo se incluye el ejemplo con esta prueba para mostrar el método de selección. El resto de ejecuciones para las otras pruebas estarán guardados en ficheros y no se mostrarán por el gran tamaño de cada una de ellas y la limitación de espacio.

ACT01	0	0	0	0	0	1	0	1	0	0
ACT02	0	1	1	0	0	1	0	1	0	1
ACT03	0	0	0	0	0	0	0	0	0	0
ACT04	0	0	0	1	0	0	1	0	1	0
ACT05	0	1	1	0	0	0	1	0	0	0
APL01	0	1	0	0	0	0	0	0	0	0
APL02	0	0	0	0	0	0	1	1	0	0
APL03	0	0	0	0	0	0	0	0	0	0
APL04	1	0	1	1	1	1	1	1	1	1
END01	0	0	0	0	0	0	0	0	0	0
END02	0	0	0	1	0	0	0	0	0	0
END03	1	1	1	1	1	1	1	1	1	1
EST01	0	0	0	0	0	0	0	0	0	0
EST02	0	0	0	0	0	0	0	0	0	0
EST03	1	1	0	1	0	0	0	0	0	1
EST04	0	0	0	0	0	0	0	0	0	0
EST05	0	1	1	1	0	0	0	1	0	1
EST06	1	0	0	1	0	1	0	1	1	0
EST07	0	0	0	0	0	0	0	0	0	0
EST08	1	0	1	0	0	0	1	1	0	0
EST09	1	0	0	1	0	0	1	0	0	0
LIQ01	0	0	0	0	0	0	0	0	0	0
LIQ02	0	0	0	0	0	0	0	0	0	0
LIQ03	1	1	1	0	1	0	1	0	0	0
LIQ04	1	0	1	0	1	0	1	0	1	0
LIQ05	1	1	1	1	1	1	1	1	1	1
LIQ06	1	0	0	1	1	0	0	0	0	0
LIQ07	1	0	1	0	0	1	0	0	1	1
LIQ08	0	1	0	0	1	1	1	0	1	1
LIQ09	1	1	1	0	1	0	1	0	0	1

LIQ10	0	0	0	0	0	0	0	0	0	0
LIQ11	1	0	1	1	1	0	1	0	1	0
LIQ12	1	0	0	1	1	0	0	1	1	0
LIQ13	0	1	1	0	1	0	1	0	1	1
REN01	0	1	0	0	0	0	0	0	0	1
REN02	0	0	1	1	0	0	1	0	0	0
REN03	0	1	1	0	0	1	1	1	0	1
REN04	0	0	0	0	0	0	0	0	0	0
REN05	1	1	1	1	1	1	1	1	1	1
REN06	1	1	0	1	0	0	0	0	0	1
ROT01	0	0	0	0	0	0	0	1	1	0
ROT02	0	0	0	0	0	0	0	0	0	1
ROT03	0	0	0	0	0	0	0	0	0	0
ROT04	0	0	0	0	0	0	0	0	0	0
ROT05	0	0	0	1	1	0	1	1	1	0
ROT06	0	0	0	0	0	0	0	0	0	0
ROT07	0	1	1	1	0	0	1	1	1	0
ROT08	0	0	0	1	1	1	0	1	1	0
SOL01	1	0	1	1	1	0	0	0	0	1
SOL02	1	1	0	0	0	1	0	1	1	1
SOL03	1	0	0	0	1	0	1	0	0	0
SOL04	0	0	0	1	1	1	1	1	1	0
SOL05	0	0	1	0	1	0	0	1	1	0
SOL06	1	0	0	1	0	1	1	0	0	0
SOL07	0	0	0	0	0	0	0	1	1	1
SOL08	0	0	0	0	0	0	0	0	0	0
SOL09	1	1	1	1	1	1	1	0	0	1
TES01	0	0	0	1	0	0	0	1	0	0
TES02	0	1	1	1	0	1	1	1	1	0

Para poder saber cuáles son los ratios seleccionados en las diez ejecuciones solo hace falta observar los ratios donde toda la fila esté formada por unos (cada casilla se corresponde con una ejecución). Esto indica que en las diez ejecuciones el algoritmo ha seleccionado siempre el mismo ratio en los individuos escogidos como mejores al final de las 500 generaciones. En este caso se puede observar que los ratios en los que todas las casillas tienen un valor de 1 son los siguientes: END03, LIQ05 y REN05. Observando la tabla 3.1.4, podemos obtener la fórmula que define estos ratios.

Primeramente podemos comprobar si estos 3 ratios tienen buenos valores según los filtros estadísticos. En el primero de los casos, el valor de ratio END03, es muy bajo en cuanto a los dos métodos de filtrado estadístico, con un valor de 0.11 y un  $3.75 * 10^7$  respectivamente en los dos métodos anteriormente mencionados. Esto nos hace pensar que se debería descartar ya en las primeras generaciones, ya que tiene un valor realmente malo, pero en este caso no solo no se descarta sino que incluso se selecciona en las 10 ejecuciones. Esto se debe a que el ratio es bueno dadas sus características de clasificación para el clasificador K-NN particular que se ha usado.

Siguiendo con el resto de ratios seleccionados por el programa evolutivo, podemos comprobar que los valores de los otros dos ratios (LIQ05, REN05) sí son altos: en el caso de LIQ05 los valores son de 7.94 y 2.28, y en REN05 8.12 y 2.32 (el primer valor correspondiente al T-estadístico y el segundo al *Fisher Score*). Lo que nos muestran estos resultados es que se confirma que estos ratios son muy relevantes en cuanto al fracaso o “no fracaso” de la empresa. En el caso del primer ratio no coincide con una calidad alta en los resultados de los métodos de selección usados; sin embargo, en los dos últimos ratios coincide la selección hecha por el algoritmo evolutivo con una buena calidad en los métodos de selección o filtrado estadístico.

#### 4.5.2 Mejores ratios para la prueba con 59 ratios y $K=3$

En el caso de utilizar  $k = 3$ , los ratios que han sido seleccionados por el algoritmo evolutivo en las 10 ejecuciones independientes son los siguientes: APL02, END03, EST03, LIQ02, LIQ05, REN01, REN05, REN06, ROT02. Al observar los ratios correspondientes con los valores obtenidos de las tablas 4.1 y 4.2, podemos diferenciar los ratios que con las primeras pruebas parecían poco relevantes, pero que acaban siendo seleccionados, y por otro lado tenemos los ratios que en las primeras observaciones mediante los *filter methods* ya nos indicaban un indicio de que podrían ser seleccionados. El primero está formado por END03, EST03, REN06, ROT02. Estos ratios tiene un T-estadístico y un *Fisher Score* muy bajo, lo que nos indica que



en principio el algoritmo no los debería seleccionar pero, sin embargo, los selecciona igualmente. Por tanto, es probable que aunque los métodos de filtrado no los escojan como buenos, en realidad sí lo sean, y contribuyan a proporcionar un valor alto de porcentaje de acierto en nuestro programa. O, por el contrario, sean elegidos debido a que el ratio por sus características puede ser bueno para el clasificador usado. Por otro lado, tenemos el grupo dos formado por APL02, LIQ02, LIQ05, REN01, REN05, los cuales sí cumplen las primeras previsiones (valores altos obtenidos mediante los métodos T-estadístico y *Fisher Score*) que se obtienen de ellos y son seleccionados finalmente en los individuos finales.

#### **4.5.3 Mejores ratios para la prueba con los 30 mejores ratios de Fisher Score y $k=15$**

Estas pruebas en las que se realiza un filtrado previo, deberían ser pruebas donde se obtiene un mejor resultado, debido a que se están utilizando como entrada los 30 mejores ratios basados en el *Fisher Score*. De este modo, aunque el valor obtenido a través del método de selección sea bajo comparado con el resto, siempre pertenecerá al conjunto de los 30 mejores valores de *Fisher Score*. En esta prueba obtenemos un gran número de ratios que son seleccionados como relevantes para la obtención de un alto porcentaje de aciertos. Los indicadores seleccionados en esta prueba son los siguientes: APL02, APL04, EST06, LIQ02, LIQ05, REN01, REN05, TES01. Como se puede observar en esta, y en las anteriores pruebas, hay algunos ratios coincidentes, esto da a entender que esos ratios escogidos tienen una mayor influencia en la calidad del clasificador, debido a que siempre o casi siempre son seleccionados, y por tanto, tienen una mayor relevancia en el resultado final. En el apartado de 4.6 se analizará con más detenimiento una comparativa de los ratios seleccionados en las distintas pruebas.

#### **4.5.4 Mejores ratios para la prueba con los 30 mejores ratios de Fisher Score y $k=3$**

Se repetirá la prueba anterior con un número menor de vecinos en el K-NN, para observar los ratios que son seleccionados después de las 10 ejecuciones del algoritmo. Como resultado de las 10 ejecuciones se obtienen los 10 mejores individuos. Los ratios seleccionados en todas las ejecuciones independientes del algoritmo evolutivo son los elegidos finalmente. Los ratios o indicadores seleccionados son los siguientes: EST09, LIQ05, REN01, REN05. Comparando resultados con la prueba anterior se puede observar que existen coincidencias en algunos ratios: los ratios LIQ05, REN01 y REN05 coinciden en las dos pruebas. Dichas pruebas son similares, la única diferencia entre una y otra es el número de vecinos usados por el clasificador

K-NN. Esto indica que estos tres ratios tienen una gran relevancia en la obtención de buenas calidades en la clasificación.

#### **4.5.5 Mejores ratios para la prueba con los 30 mejores ratios de T-estadístico y $k=15$**

Por último, se harán las mismas pruebas partiendo de los 30 mejores ratios proporcionados por el filtrado en el T-estadístico. De esta forma se podrá conocer qué ratios son seleccionados con las mejores 30 medidas de calidad del T-estadístico. Los siguientes indicadores son los que han sido seleccionados tras las 10 ejecuciones independientes: APL04, REN05. El número de ratios seleccionados es mucho menor que en las pruebas anteriores, ya que solo se han seleccionado dos ratios. De todas formas se puede observar que los ratios seleccionados coinciden con muchas de las anteriores pruebas. El ratio APL04 fue seleccionado en la prueba hecha anteriormente con los 30 mejores ratios *Fisher Score* y  $k = 15$ , mientras que REN05 se repite en todas las pruebas realizadas hasta el momento. Este ratio será considerado como uno de los ratios más significativos, debido al gran número de veces que se selecciona en las diferentes pruebas.

#### **4.5.6 Mejores ratios para la prueba con los 30 mejores ratios de T-estadístico y $k=3$**

Esta será la última prueba que se realizará en cuanto a la selección de los mejores ratios. En esta prueba solo se selecciona un ratio (es decir, seleccionado en las 10 ejecuciones independientes), el REN05. Este ratio se ha seleccionado en todas las pruebas, lo que nos indica la gran importancia que tiene para la obtención de buenos resultados en la clasificación. También se han observado los valores de T-estadístico y *Fisher Score*, y como se puede comprobar observamos que ese ratio tiene un alto valor en ambas medidas, exactamente un 8.12 y un 2.32 en dichas medidas, lo cual indica la gran relevancia del ratio.

### **4.6 Deducciones y observaciones a través de los ratios seleccionados por el algoritmo en las distintas pruebas**

En la figura 4.9, cada columna corresponde a un ratio y la altura de la columna dependerá del número de veces que ha aparecido un determinado ratio en las distintas pruebas. Cuantas más veces aparezca seleccionado el ratio, más alta será la columna y viceversa, ya que el eje Y indicará el número de veces que se repite.

Este gráfico nos proporciona una visión global de los ratios que han sido seleccionados en

las seis pruebas anteriores.

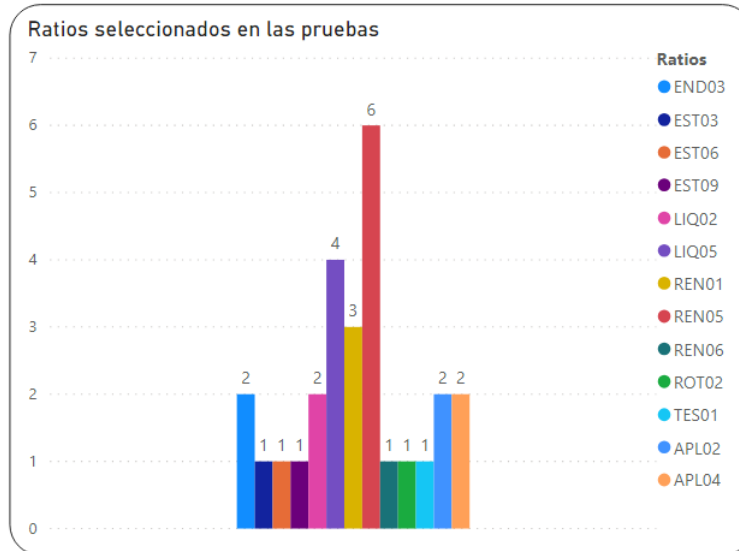


Figura 4.9: Número de veces que un ratio ha sido seleccionado en las diferentes pruebas.

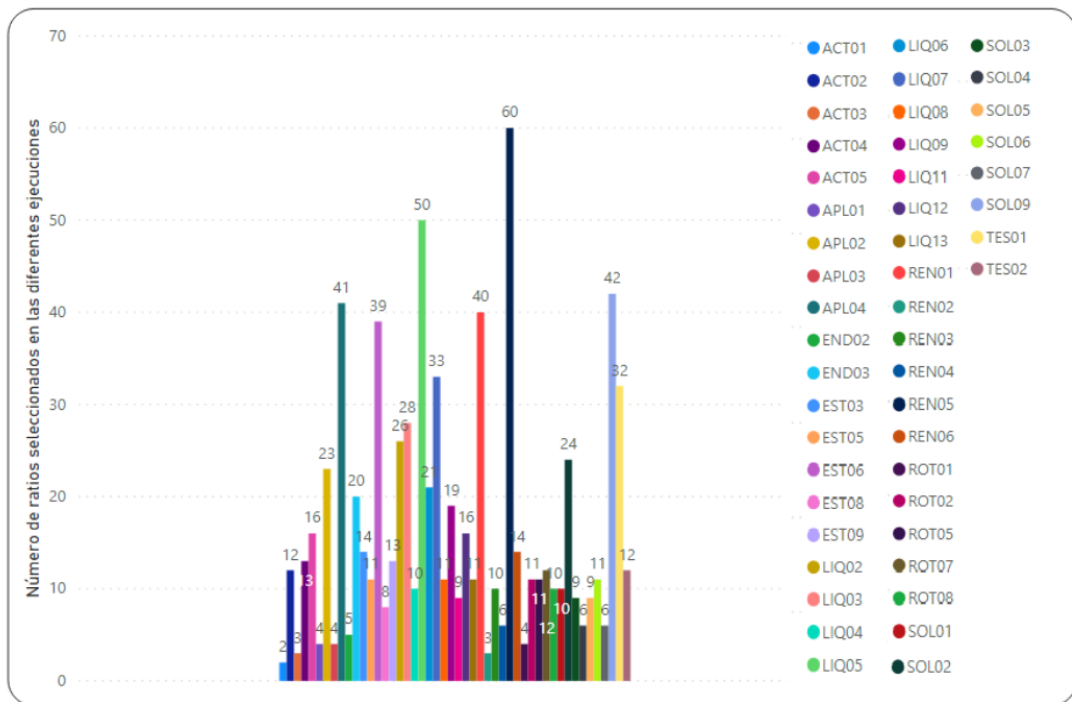
Como se denota en la figura 4.9, el ratio REN05, es el más repetido, ya que aparece en todas las pruebas, seguido de LIQ05, que se selecciona 4 veces, y por REN01, que aparece en 3 de las 6 pruebas. Si nos fijamos en sus T-estadísticos y *Fisher Scores* vemos como coinciden con unos valores altos, es decir, valores que muestran una importante relevancia del ratio en el futuro de la empresa, lo que indica que estos ratios ya habían sido indicados como buenos ratios a través de los *filters methods*, y, como es lógico, el algoritmo los selecciona en varias pruebas, afirmando su importancia en cuanto al uso de ellos para la obtención de una mejor calidad en las pruebas.

Los valores que obtienen los siguientes indicadores a través de los filtros de selección anteriormente mencionados son los siguientes: REN05 (8.12, 2.32); LIQ05 (7.94, 2.28); REN01 (9.78, 7.09); el primer valor de cada uno hace referencia al T-estadístico y el segundo al *Fisher Score*. Podemos observar cómo los 3 ratios que más veces se repiten en la figura 4.9 y 4.10, obtienen un valor alto en las figuras 4.1 y 4.2, cuyos valores son bien diferenciados en ambas figuras como picos.

De la misma forma se ha generado la figura 4.10, pero con una pequeña distinción. En vez de mostrar el número de pruebas en las que los ratios son seleccionados, se mostrará el número de veces que ha sido seleccionado el ratio a nivel de ejecución independiente del algoritmo, debido a que en la gráfica anterior puede darse el caso de que haya varios ratios que han sido

seleccionados en 9 de las 10 ejecuciones y, como fallaron en una, no han sido considerados. De esta forma nos aseguramos tener una visualización más real de las veces que ha sido seleccionado un ratio.

Por tanto, en esta gráfica (4.10) se muestra el número de veces que ha sido seleccionado cada ratio independientemente de si ha coincidido en las 10 ejecuciones de cada prueba o no.



## 4.7 Obtención de las matrices de confusión

En este apartado se muestran las matrices de confusión correspondientes al clasificador K-NN cuando este usa el conjunto optimizado de ratios proporcionado por el algoritmo. Este conjunto optimizado corresponde a aquellos ratios que fueron seleccionados en las 10 ejecuciones independientes de cada prueba, como se explicó en los apartados previos. Además, para comparar los resultados, se mostrarán los resultados de clasificación cuando se usan los 30 mejores ratios conforme a *Fisher Score* y al T-estadístico, al igual que cuando se usa el conjunto completo de los 59 ratios. Cada prueba se hará dos veces, la primera con  $k = 3$  y la segunda con  $k = 15$ , para ver la diferencia de resultados con menos o más números de vecinos en el clasificador.

Los datos de entrada que se usarán en estas pruebas serán 274 registros del fichero de entrenamiento, 137 de empresas que fracasan y otras 137 de empresas que no fracasan; por otro lado, tendremos el “fichero de test”, del cual se seleccionarán 136 empresas que se predicen como fracasadas y otras 136 que se predicen como no fracasadas. De esta forma, las clases están balanceadas, mismo número de no fracasos que de fracasos. Además, se repetirán estas pruebas también con el “fichero de test” (matTest) completo, formado por 18360 registros, donde los registros no estarán balanceados ya que habrá una mayoría de registros de empresas que se predicen como no fracasadas (18224).

A continuación se van a mostrar en los siguientes apartados las matrices de confusión obtenidas para cada una de las pruebas, una para el fichero de test completo (18360 registros), y otra con el fichero de test reducido (272 registros). En cada caso se mostrarán sus medidas más importantes, citadas ya en la sección 2.7, además de los porcentajes de acierto y error de cada una de ellas.

### 4.7.1 Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo de los 30 mejores ratios de *Fisher Score* con $K=3$

**Test reducido:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	110	26
	No fracaso	22	114

- Exactitud =  $(114+110) / (114+22+26+110) = 82.35\%$
- Precisión =  $110/(22+110) = 83.33\%$
- Sensibilidad =  $110/(110+26) = 80.88\%$
- Especificidad =  $114/(114+22) = 83.82\%$

- Porcentaje de aciertos: 82.35%
- Porcentaje de errores: 17.65%

**Test ampliado:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	110	26
	No fracaso	2278	15946

- Exactitud =  $(15946+110)/(15946+2278+26+110) = 87.45\%$
- Precisión =  $110/(110+2278) = 4.6\%$
- Sensibilidad =  $110/(110+26) = 80.88\%$
- Especificidad =  $15949/(15949+2278) = 87.5\%$

- Porcentaje de aciertos: 87.45%
- Porcentaje de errores: 12.55%

### 4.7.2 Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo de los 30 mejores ratios de *Fisher Score* con $K=15$

**Test reducido:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	120	16
	No fracaso	15	121

- Exactitud = 88.60%
- Precisión = 88.88%
- Sensibilidad = 88.23%
- Especificidad = 88.97%

- Porcentaje de aciertos: 88.60%
- Porcentaje de errores: 11.40%

**Test ampliado:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	120	16
	No fracaso	1328	16896

- Exactitud = 92.67%
- Precisión = 8.28%
- Sensibilidad = 88.23%
- Especificidad = 92.71%

- Porcentaje de aciertos: 92.67%
- Porcentaje de errores: 7.33%

**4.7.3 Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo de los 30 mejores ratios de T-estadístico con K=3**

**Test reducido:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	123	13
	No fracaso	19	117

- Exactitud = 88.23%
- Precisión = 86.61 %
- Sensibilidad = 90.44%
- Especificidad = 86.02%

- Porcentaje de aciertos: 88.23%
- Porcentaje de errores: 11.77%

**Test ampliado:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	123	13
	No fracaso	1959	16265

- Exactitud = 89.25%
- Precisión = 5.9%
- Sensibilidad = 90.44%
- Especificidad = 89.25%

- Porcentaje de aciertos: 89.25%
- Porcentaje de errores: 10.75%

#### 4.7.4 Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo de los 30 mejores ratios de T-estadístico con K=15

**Test reducido:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	125	11
	No fracaso	21	115

- Exactitud = 88.23%
- Precisión = 85.61%
- Sensibilidad = 91.92%
- Especificidad = 84.55%

- Porcentaje de aciertos: 88.23%
- Porcentaje de errores: 11.77%

**Test ampliado:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	125	11
	No fracaso	1581	16643

- Exactitud = 91.32%
- Precisión = 7.32%
- Sensibilidad = 91.92%
- Especificidad = 91.32%

- Porcentaje de aciertos: 91.32%
- Porcentaje de errores: 8.68%

#### 4.7.5 Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo del conjunto completo de los 59 ratios con K=3

**Test reducido:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	112	24
	No fracaso	18	118

- Exactitud = 84.55%
- Precisión = 86.15%
- Sensibilidad = 82.35%
- Especificidad = 86.76%

- Porcentaje de aciertos: 84.55%
- Porcentaje de errores: 15.45%



**Test ampliado:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	112	24
	No fracaso	962	17262

- Exactitud = 94.62%
- Precisión = 10.42%
- Sensibilidad = 82.35%
- Especificidad = 94.72%

- Porcentaje de aciertos: 94.62%
- Porcentaje de errores: 5.38%

**4.7.6 Matriz de confusión utilizando los mejores ratios seleccionados por el programa partiendo del conjunto completo de los 59 ratios con K=15**

**Test reducido:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	112	24
	No fracaso	12	124

- Exactitud = 86.76%
- Precisión = 90.32%
- Sensibilidad = 82.35%
- Especificidad = 91.17%

- Porcentaje de aciertos: 86.76%
- Porcentaje de errores: 13.24%

**Test ampliado:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	112	24
	No fracaso	763	17461

- Exactitud = 95.72%
- Precisión = 12.8%
- Sensibilidad = 82.35%
- Especificidad = 95.81%

- Porcentaje de aciertos: 95.72%
- Porcentaje de errores: 4.28%

#### 4.7.7 Resultados del K-NN sin selección de datos

En esta prueba se muestra es el resultado de aplicar el K-NN utilizando los 59 ratios y  $k=3$  y sin aplicar la selección evolutiva de los ratios más relevantes.

Como se ha hecho en las anteriores apartados se muestran los resultados de clasificación con un fichero test balanceado formado por 272 registros y con un fichero test de 18360 registros no balanceado.

##### Test reducido:

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	107	29
	No fracaso	55	81

- Exactitud = 69.12%
- Precisión = 66.04%
- Sensibilidad = 78.67%
- Especificidad = 59.55%

- Porcentaje de aciertos: 69.12%
- Porcentaje de errores: 30.88%

##### Test ampliado:

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	107	29
	No fracaso	2995	15229

- Exactitud = 83.52%
- Precisión = 3.44%
- Sensibilidad = 78.67%
- Especificidad = 83.56%

- Porcentaje de aciertos: 83.52%
- Porcentaje de errores: 16.48%

Si comparamos estos resultados con los obtenidos de la prueba 4.7.5 se puede observar un mejor resultado aplicando la selección evolutiva. La sensibilidad utilizando únicamente el clasificador K-NN es de 78,67%, sin embargo, aplicando el clasificador con selección evolutiva aumenta hasta un 82.35%.

Esto también sucede si hacemos una comprobación para el resto de pruebas. Los resultados de las comparativas siempre serán superiores si utilizamos el algoritmo ED con el K-NN.

## 4.8 Resultado de las pruebas con redes neuronales

En este apartado se explicarán las pruebas realizadas con el paquete de redes neuronales proporcionado por Matlab, ya comentado en la sección 2.5. Además, se analizarán los resultados obtenidos a través de estas redes.

Se han hecho numerosas pruebas con RNAs para la obtención de los resultados más óptimos. Para la realización de las siguientes pruebas, se ha creado un programa que permita la realización de estas pruebas automáticamente en Matlab, para poder realizar un mayor número de pruebas sin hacer las ejecuciones manualmente. Este programa se encarga de ejecutar redes neuronales y entrenarlas mediante unas restricciones que se explicarán a continuación. Se han hecho diferentes pruebas para la obtención de los resultados mediante el entrenamiento de las RNAs, el número de pruebas realizadas ha sido 10000 ya que se ha probado con 10 combinaciones de parámetros (modificación en el número de neuronas de cada capa de la red) diferentes y cada una se ha realizado 1000 veces (1000 repeticiones). De esta forma el programa nos permite una gran cantidad de pruebas en unos pocos minutos sin la interacción con el mismo.

Los criterios generales establecidos para la realización de las pruebas son los siguientes:

- Número de capas ocultas.
- Número de entrenamientos que se realizarán sobre la red.

En cuanto a los criterios de parada del entrenamiento de la red, se utilizarán los siguientes en cada una de las ejecuciones realizadas por el programa:

- Número máximo de iteraciones (“epochs”).
- Número máximo de incrementos (“max fail”).

El número máximo de repeticiones o *epochs* será considerado como el número de iteraciones/ciclos máximos de entrenamiento de la red, el número máximo de incrementos será considerado como el número máximo de veces que el gradiente no mejora y el número total de entrenamientos será el número total de veces que se ha ejecutado el entrenamiento de la red de manera independiente con los criterios de parada anteriormente indicados.

Para filtrar los resultados de las pruebas se han establecido otras normas para generar resultados acordes al objetivo buscado, en este caso un menor número de falsos no fracasos. Algunas de las condiciones establecidas al iniciar las pruebas con el programa fueron que el error relativo máximo en el modelo no bajase del 15% y en el test no bajase del 20,5% y se habían puesto límites en algunas otras medidas relacionadas con la matrices (especificidad,

sensibilidad, ...), pero finalmente se han realizado ejecuciones que cumplan unas reglas más estrictas y más ajustadas a la obtención del menor número de falsos no fracasos para generar un menor número de redes óptimas, pero mejores en cuanto a resultados. Las últimas pruebas realizadas para acotar mejor el número de redes seleccionadas han guardado solo las redes en las cuales el “accuracy” no bajase del 90%, tanto en el test, como en el conjunto de entrenamiento y que en sus matrices de confusión las empresas con falsos no fracasos sean menores a 12.

Las pruebas que mejores resultados mostraron fueron las hechas con dos capas ocultas. Se exponen a continuación dos pruebas en las cuales se han obtenido resultados muy relevantes en cuanto al estudio.

1. La primera prueba: Se ha realizado con 3 ratios de entrada y dos capas ocultas, una de 3 neuronas y otra de 2; por tanto, la estructura de la red sigue el patrón 3-3-2-1. Los ratios escogidos para la entrada de la red han sido los tres mejores seleccionados por el algoritmo evolutivo.
2. La segunda prueba: Se ha realizado con 10 ratios de entrada y dos capas ocultas, una de 7 neuronas y otra de 3; por tanto, la estructura de la red sigue el patrón 10-7-3-1. Los ratios escogidos para la entrada de la red han sido los diez mejores seleccionados por el algoritmo evolutivo.

Estas dos pruebas se han ejecutado con el conjunto de test ampliado (18360 registros). El objetivo de estas pruebas es obtener un resultado con el mínimo número de falsos no fracasos, para ser certero en la predicción de empresas fracasadas. A continuación se muestran dos de los ejemplos que obtienen menos “falsos no fracasos”, o lo que es lo mismo, una sensibilidad alta. Los resultados de las pruebas mostradas a continuación se obtuvieron cumpliendo las siguientes condiciones:

- Número máximo de repeticiones = 1000
- Número máximo de entrenamientos = 1000
- Número máximo de incrementos = 200
- Número de capas ocultas = 2

### 4.8.1 Resultados de la primera prueba (3-3-2-1)

En esta prueba, los 3 ratios seleccionados para la entrada de la red neuronal son los que más veces han sido seleccionados en las pruebas (4.2.1, 4.2.2, 4.2.3, 4.2.4, 4.2.5, 4.2.6) por los mejores individuos de cada una. La figura en la que se muestra el número de veces que se repitió cada ratio es la figura 4.9.

#### Resultados conjunto de entrenamiento:

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	122	15
	No fracaso	11	126

- Exactitud = 90.51%
- Precisión = 91.73%
- Sensibilidad = 89.05%
- Especificidad = 91.97%

- Porcentaje de aciertos: 90.51%
- Porcentaje de errores: 9.49%

#### Resultados test:

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	126	10
	No fracaso	1761	16463

- Exactitud = 90.35%
- Precisión = 6.67%
- Sensibilidad = 92.64%
- Especificidad = 90.33%

- Porcentaje de aciertos: 90.35%
- Porcentaje de errores: 9.65%

Los resultados obtenidos en esta prueba nos dan alrededor de un 90% de aciertos en la predicción, esto nos indica que usando tan solo los 3 mejores ratios de la empresa se obtiene un resultado considerablemente alto, además de una alta sensibilidad, ya que, solo 10 de las 136 empresas que fracasan están mal clasificadas.

### 4.8.2 Resultados de la segunda prueba (10-7-3-1)

En esta prueba, los 10 ratios seleccionados para la entrada de la red neuronal son los que más veces han sido seleccionados en las pruebas (4.2.1, 4.2.2, 4.2.3, 4.2.4, 4.2.5, 4.2.6) por los mejores individuos de cada una. La figura en la que se muestra el número de veces que se

repitió cada ratio es la figura 4.9.

**Resultados conjunto entrenamiento:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	128	9
	No fracaso	8	129

- Exactitud = 93.80%
- Precisión = 94.11%
- Sensibilidad = 93.43%
- Especificidad = 94.16%

- Porcentaje de aciertos: 93.80%
- Porcentaje de errores: 6.20%

**Resultados test:**

		Predicción	
		Fracaso	No fracaso
Clase	Fracaso	127	9
	No fracaso	1751	16473

- Exactitud = 90.41%
- Precisión = 6.76%
- Sensibilidad = 93.38%
- Especificidad = 90.39%

- Porcentaje de aciertos: 90.41%
- Porcentaje de errores: 9.59%

En cuanto al porcentaje de acierto, se puede observar una mejora relativamente pequeña respecto al porcentaje de la prueba anterior. Del mismo modo que la cuota de falsos fracasos que, como se observa, baja en una unidad. Por tanto, cuántos más ratios se seleccionen para la prueba, mejores resultados se obtienen. Podemos observar a través de estas dos pruebas que se puede reducir de una forma considerable el número de datos (ratios) de entrada sin que afecte en gran medida a los resultados.

## 4.9 Comparación de los resultados de la red neuronal con otro estudio

Se han comparado los resultados de la RNA expuesta en la prueba 4.8.2 con los resultados del estudio de A. Beade [23] [24] [25] [26], el cual está centrado en la predicción del fracaso empresarial partiendo de los 59 ratios considerados más relevantes, en cuanto a la influencia que tienen en el fracaso de una organización, es decir, los mismos con los que se parte desde el inicio en este proyecto.

Cabe destacar que los conjuntos de entrenamiento y de test usados en los dos trabajos fueron los mismos. A. Beade seleccionó los 10 ratios más relevantes a través de un análisis de sensibilidad, la selección de estos 10 ratios se realizó con aquellas variables que presentaron un mayor sumatorio de los doce índices de sensibilidad por modelo y configuración (3 configuraciones y 4 modelos). Respecto a las pruebas hechas por A. Beade, los 4 modelos corresponden con los años 1, 2, 3 y 4 previos al resultado definitivo de la empresa en cuanto a su fracaso/no fracaso, y las tres configuraciones probadas para cada modelo tienen las siguientes estructuras: 59-39-20-1, 59-30-15-1 y 59-15-4-1. Para comprobar la efectividad de este trabajo se han comparado los resultados obtenidos por A. Beade en [23] [24] [25] [26] con los obtenidos en este proyecto utilizando solo los 10 ratios considerados más relevantes en ambos casos para la entrada de la red neuronal.

		Modelo año 1 antes de fracaso	
		Red 10 - 7 - 3 - 1	
Target	Fracasadas	126	10
	No fracasadas	1.203	17.021
		Fracasadas	No fracasadas
		Output	

Figura 4.11: Matriz de confusión obtenida del trabajo de investigación de A. Beade en el conjunto de test.

Si nos fijamos en los resultados mostrados en la figura 4.11, perteneciente al proyecto de A. Beade y los comparamos con los de la prueba 4.8.2, se puede observar una similitud de resultados. Uno de los puntos en los que está centrado ese trabajo [23] [24] [25] [26] se basa en la reducción del número de errores en la predicción de las empresas que fracasarán, pero que el algoritmo las considerada como no fracasadas. Esta situación se corresponde con la

casilla superior derecha de las dos matrices. Como se puede observar en el caso de la matriz del modelo de A. Beade, su mejor resultado tiene un número de 10 empresas que han predicho mal el modelo. Sin embargo, en la matriz considerada en este proyecto el número de empresas con una predicción errónea es una unidad menor. Esto nos indica que la sensibilidad será más alta en la prueba realizada en este proyecto, es decir, se puede reducir el número de empresas mal clasificadas en esta situación.

De todas formas no se mejoran los resultados en todos los aspectos: se puede apreciar cómo el *accuracy* (exactitud), y también la especificidad, son superiores en el trabajo de A. Beade, ya que el número aciertos en cuanto a la predicción en las empresas con éxito es mayor.

En principio, el objetivo del proyecto ha sido centrarse en el número de empresas que fracasan, pero que el modelo las predice como no fracasadas. De esta forma no se le da tanta importancia al resto de medidas como a la sensibilidad. Se considera como “mal menor” que se pueda predecir un empresa como fracasada y que en realidad no fracase, ya que, la empresa en este caso no correrá ningún riesgo crítico que confluya en una quiebra o incluso en la disolución de la organización. Es decir, se eligen los modelos de predicción siempre desde el punto de vista conservador de un inversor o responsable de crédito a empresas.





# Conclusiones

---

Este proyecto se ha centrado en dos objetivos principales: obtener una reducción significativa de los parámetros de entrada en un modelo predictivo, sin que exista una pérdida de calidad en los resultados obtenidos, y en la reducción del error del modelo clasificador/predictor en las empresas clasificadas como empresas no fracasadas que, en realidad, acaban siendo empresas fracasadas.

Este estudio está diseñado para la predicción de dichos resultados un año antes del fracaso/no fracaso de la empresa. Esto podría indicar, con un año de anterioridad, el fracaso de la empresa, pudiendo considerar esto como un aviso para poder tomar medidas y cambiar la dirección económica de dicha empresa, antes de llegar a un punto crítico en la situación financiera, incluso de llegar a la quiebra.

Al inicio de este proyecto se hizo hincapié en dos métodos de selección de características (T-estadístico y *Fisher Score*), pertenecientes al grupo de los *filter methods*. Este tipo de métodos de selección se basan meramente en funciones estadísticas aplicadas a una serie de datos, como puede ser, en nuestro caso, los ratios. Estos métodos de selección no nos aportan información en relación con otros factores, se centran en el análisis univariable. Para solventar esta carencia que tienen este tipo de métodos utilizamos los *wrapper methods*. De esta forma se añade capacidad de decisión mediante un algoritmo de aprendizaje.

Para la implementación del algoritmo desarrollado en este proyecto se ha combinado el clasificador K-NN con el algoritmo de evolución diferencial. El K-NN se utiliza para la clasificación de una determinada combinación de ratios, otorgándole una calidad o bondad. Por otro lado, el algoritmo de evolución diferencial, permite obtener combinaciones de diferentes ratios, para poder mejorar los resultados obtenidos por el K-NN aplicado sobre el conjunto de datos de la anterior generación. Este proceso se repite a lo largo de las generaciones del proceso

---

evolutivo de ED, de modo que el algoritmo evolutivo optimiza, a lo largo de las generaciones, el conjunto de ratios que facultan al clasificador K-NN con el mayor poder clasificador.

Los resultados obtenidos en este trabajo muestran de una forma clara un alto porcentaje de aciertos en la predicción a un año vista. La mayoría de pruebas realizadas están en torno al 90% de aciertos, es decir, en 9 de cada diez empresas podemos obtener unos valores reales sobre los resultados de su año próximo. Este proyecto se centra principalmente en 3 conjuntos de pruebas: las pruebas realizadas sobre el conjunto inicial de los 59 ratios seleccionados a partir de su importancia en estudios previos que corroboraban su relevancia en cuanto al resultado del fracaso empresarial de una empresa u organización; los 30 mejores ratios seleccionados sobre estos 59 ratios a través del método de selección T-estadístico; y los 30 mejores ratios seleccionados sobre estos 59 ratios a través del método de selección de *Fisher Score*. Sobre estos conjuntos de 59 o 30 ratios, el algoritmo ED/K-NN selecciona a su vez los más relevantes.

En las primeras pruebas realizadas se obtuvieron los datos de la calidad media de la población y de la calidad del mejor individuo en cada generación. Estos datos se obtuvieron a través de la media de 10 ejecuciones del algoritmo ED/K-NN para evitar variaciones en cuanto a casos concretos. Los resultados obtenidos se mantienen prácticamente en todas las generaciones finales en torno al 90%, lo que indica un alto porcentaje de aciertos en todas ellas. En las gráficas incluidas en el capítulo 4 (Sección 4.2), se puede visualizar un mayor crecimiento de la calidad en las primeras generaciones de las pruebas con los 30 mejores ratios, tanto de *Fisher Score* como de T-estadístico. Esto se debe a la preselección de dichos ratios con los *filter methods*, que permiten la mejor y más rápida evolución de los individuos hacia una mayor calidad. Es decir, con un menor número de ratios iniciales podemos obtener una mayor calidad en un menor número de generaciones.

En cada una de las diez ejecuciones de las pruebas utilizadas para el cálculo del mejor individuo y de la calidad media de la población se han guardado los mejores individuos de la última generación (generación 500) de cada ejecución. Este individuo nos muestra los ratios seleccionados al final de cada prueba y, por tanto, aquellos con los que se obtienen mayor calidad.

Una vez obtenidos los “mejores individuos” (mejores conjuntos de ratios) se han hecho una serie de pruebas centradas únicamente en los ratios que más veces han sido seleccionados previamente. Se emplean como datos de entrada los ratios que han sido seleccionados en los mejores individuos escogidos para las diez ejecuciones de cada prueba. Se ha optado por utilizar solo los ratios que han sido seleccionado en los 10 mejores individuos de cada prueba,

por tanto, en el caso de que un individuo haya sido seleccionado 9 de las 10 veces posibles en una prueba, no será considerado como un ratio apto, además, los casos con un menor número de selecciones tampoco se considerarán.

Los resultados nos muestran una buena clasificación de forma general. Como observamos en todos los casos, el resultado para la misma prueba obtenido con  $k = 15$  es ligeramente mejor que el obtenido con  $k = 3$ , al utilizar un mayor número de vecinos. Por contra, el coste computacional es menor con  $k = 3$ .

También es importante la comparativa, en cuanto a los ratios seleccionados, entre los resultados obtenidos mediante los métodos de selección de características (T-estadístico y *Fisher Score*) que aportan un valor cuantitativo a la relevancia de cada ratio en el fracaso/no fracaso de una empresa, y los resultados finales aplicando el clasificador K-NN combinado con el algoritmo de evolución diferencial, aportando los ratios más relevantes a través del número de veces que se seleccionan en los mejores individuos de las 6 pruebas realizadas (Sección 4.5). Se puede apreciar una gran relación entre los resultados obtenidos por ambos, en cuanto a los ratios considerados más relevantes. La correspondencia entre el número de veces que se selecciona un ratio mediante los mejores individuos mostrados en la gráfica (4.10), con los valores altos en las tablas 4.1 y 4.2, es notable. Lo que confirma que estos métodos sirven como un buen indicador inicial sobre la relevancia de los ratios a la hora de hacer un filtrado previo, y poder así, seleccionar para el modelo clasificador/predictor una entrada con ratios depurados.

Se podrían añadir muchas comparaciones en cuanto a la utilización de los ratios seleccionados por el algoritmo que combina el clasificador K-NN y ED, y todos los ratios utilizando únicamente K-NN (sin utilizar ED). En este proyecto se ha incluido un único caso en el que se corrobora la diferencia de resultados obtenidos entre estas dos opciones. Los resultados logrados por un clasificador básico K-NN son peores que los logrados de la combinación del clasificador con evolución diferencial. En el ejemplo expuesto en este proyecto se puede apreciar un diferencia de un 15.43% de aciertos entre una prueba y otra (Sección 4.7.7).

Por último, se han hecho dos tipos de pruebas relacionadas con la clasificación con RNAs. Se han intentado conseguir buenos resultados, no tanto en cuanto al porcentaje de aciertos globales de la predicción, sino en la reducción al máximo de los fallos en los que el resultado previsto es contrario a la realidad. A través de estas dos últimas pruebas (Sección 4.8) se han podido obtener unos resultados con valores muy bajos en cuanto al error en el fracaso, lo que indica que podría utilizarse como un modelo bastante fiable de predicción del fracaso.

---

# Bibliografía

---

- [1] P. J. FitzPatrick, *A Comparison of the Ratios of Successful Industrial Enterprises With Those of Failed Companies*. Washington, 1932.
- [2] A. H. Winakor, *Financial Aspects of Corporate Net Worth*. Taylor and Francis, Ltd., 1936.
- [3] R. F. Smithr and A. H. Winakor, *Changes in the Financial Structure of Unsuccessful Industrial Corporations*. Urbana: University of Illinois, 1935.
- [4] C. L. Merwin, *Financing Small Corporations in Five Manufacturing Industries*. NBER, 1942.
- [5] W. Beaver, *Financial Ratios as predictor of failure*. *Journal of Accounting Research*. Wiley, 1966.
- [6] E. Altman, "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy," *Journal of Finance*, pp. 589–609, 1968.
- [7] L. A. Bernstein, *Análisis de estados financieros. Teoría, Aplicación e Interpretación*. Gestion 2000, 1997.
- [8] X. S. Zhao, L. L. Bao, Q. Ning, and J. C. Ji, "An improved binary differential evolution algorithm for feature selection in molecular signatures," *Molecular Informatics*, vol. 37, no. 4, 2018.
- [9] J. Gui, Z. Sun, S. Ji, D. Tao, and T. Tan, "Feature selection based on structured sparsity: A comprehensive study," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 7, 2017.
- [10] Q. Gu, Z. Li, and J. Han, "Generalized Fisher score for feature selection," *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence, UAI 2011*, 2012.
- [11] K. Price, R. Storn, and J. Lampinen, *Differential Evolution. A Practical Approach to Global Optimization*. Springer - Natural Computing Series, 2005.

- 
- [12] R. Storn and K. Price, “Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces,” *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [13] S. Das and P. Suganthan, “Differential evolution: A survey of the state-of-the-art,” *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.
- [14] V. Feoktistov, *Differential Evolution: In Search of Solutions*. NY: Springer, 2006.
- [15] R. Storn and K. Price, *Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces*. International Computer Science Institute, 1995.
- [16] D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. Cambridge, MA, USA: MIT Press, 1986.
- [17] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, “State-of-the-art in artificial neural network applications: A survey,” *Heliyon*, vol. 4, no. 11, pp. 2405–8440, 2018.
- [18] Y. Shi and X. Li, “An overview of bankruptcy prediction models for corporate firms: A systematic literature review,” *Intangible Capital*, vol. 15, no. 2, 2019.
- [19] W. Pitts, *Some Observations on the Simple Neuron Circuit*. Bulletin of Mathematical Biology, 1942.
- [20] W. Pitts, *The Linear theory of Neuron Networks: The Static Problem*. Bulletin of Mathematical Biology, 1942.
- [21] W. Pitts, *The Linear theory of Neuron Networks: The Dynamic problem*. Bulletin of Mathematical Biology, 1943.
- [22] P. I. Viñuela and I. M. Galván, *Redes de neuronas artificiales : un enfoque práctico*. Pearson Education, 2003.
- [23] A. Beade, “Análisis temporal de predicción de insolvencia empresarial,” Trabajo fin de grado, Universidade da Coruña, 2016.
- [24] A. Beade, J. Santos, and M. Rodríguez, “Selección automática de ratios financieros significativos en modelos conexionistas de predicción de la insolvencia empresarial mediante análisis de sensibilidad,” *XVII Encuentro AECA (Asociación Española de Contabilidad y Administración de Empresas)*, no. 138b, pp. 1–28, 2016.

- [25] A. Beade, J. Santos, and M. Rodríguez, “Análisis temporal de predicción de la insolvencia empresarial en pymes gallegas mediante modelos conexionistas,” *VII Jornadas de AECA (Asociación Española de Contabilidad y Administración de Empresas) sobre “Valoración, Financiación y Gestión de Riesgo”*, 2016.
- [26] A. Beade, J. Santos, and M. Rodríguez, “Modelos conexionistas de predicción de la insolvencia empresarial: Predicción en ambiente real,” *XIX Congreso AECA (Asociación Española de Contabilidad y Administración de Empresas)*, no. 40b, pp. 1–30, 2017.



