Objetivo:

Si bien el dataset Iris es usado con frecuencia para problemas de clasificación, en este caso decidí abordar un problema de agrupamiento usando este mismo. Por lo tanto, el objetivo principal del problema de clustering en este contexto es agrupar las observaciones en función de similitudes en las características morfológicas, sin tener en cuenta las etiquetas de clase. Es decir, explorar y descubrir patrones de agrupamiento que pueden ayudar en la comprensión de la variabilidad de las características en el conjunto de datos.

Tipo de problema:

Este enfoque, cuando se utiliza el conjunto de datos Iris enfocado en el clustering, pertenece al tipo de problema de clustering no supervisado. Es importante tener en cuenta que:

- En este enfoque, el objetivo no es predecir una etiqueta de clase específica, sino descubrir patrones y estructuras ocultas en los datos mediante la formación de grupos (clusters) de observaciones similares.

- Los datos conservan su naturaleza original, que consta de cuatro características numéricas continuas: longitud y ancho del sépalo, longitud y ancho del pétalo. Sin embargo, las etiquetas de clase (especies de iris) se eliminan, ya que en un problema de clustering no se utilizan.

Entorno de desarrollo:

En este caso optaremos por el uso de la herramienta RapidMiner. Esta nos permitirá llevar a cabo el análisis de datos y la implementación de diversos modelos. Esto con el fin de obtener una solución lo más acertada posible a nuestro objetivo.

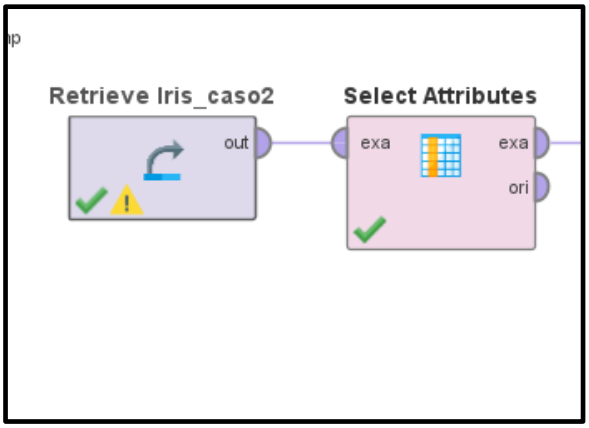
Sobre RapidMiner:

RapidMiner es utilizado en una variedad de industrias, desde la salud y las finanzas hasta la fabricación y la investigación académica, para abordar una amplia gama de problemas de análisis de datos y aprendizaje automático. Es una plataforma de código abierto y una suite de software para la ciencia de datos, el aprendizaje automático y el análisis avanzado de datos. Fue desarrollada para facilitar y acelerar el proceso de análisis de datos y la construcción de modelos predictivos, lo que la convierte en una herramienta valiosa para científicos de datos, analistas y profesionales en el campo de la inteligencia empresarial. Cabe destacar, que además de su edición de código abierto, también existe una versión comercial con características adicionales y soporte profesional.

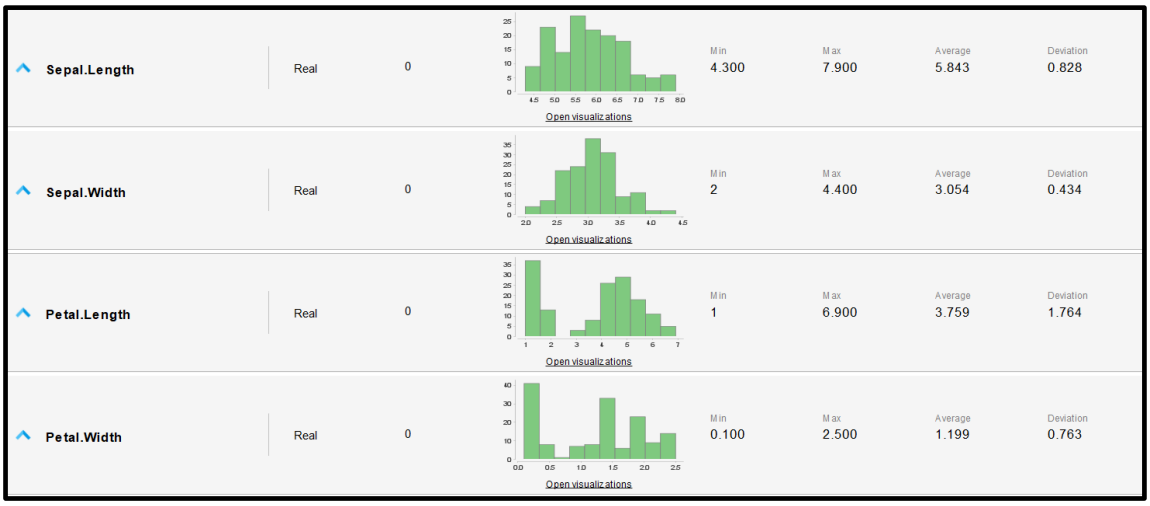
Para más información puede visitar el sitio oficial: https://rapidminer.com/

Análisis del dataset, preparación de datos y selección de atributos:

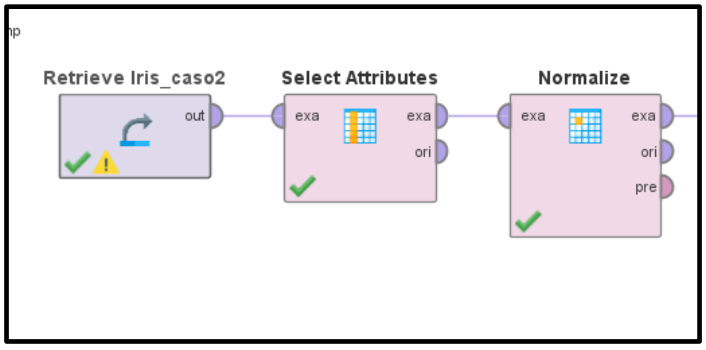
Antes que nada, es vital saber, que para abordar el conjunto de datos Iris como un problema de agrupamiento, se deben suprimir las etiquetas de clase (las especies de iris) en el conjunto de datos para que el algoritmo de agrupamiento no tenga acceso a esta información. Para eso usaremos el operador “Select Atributes”



Algo que siempre queremos hacer al momento de trabajar con análisis de clustering o cualquier otro método que trabaje con distancias es normalizar los datos. Esto nos ayuda a evitar que se dé un “peso” no correspondiente en el análisis a las variables que se miden en una escala mucho mayor. Un buen ejemplo de esto es la variable “Petal.Length” (el cual se mide en la escala más grande).



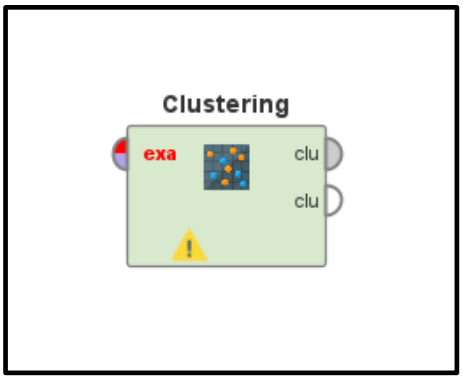
En este caso usaremos el operador “Normalize” (Z-transformation) para solucionar este problema.



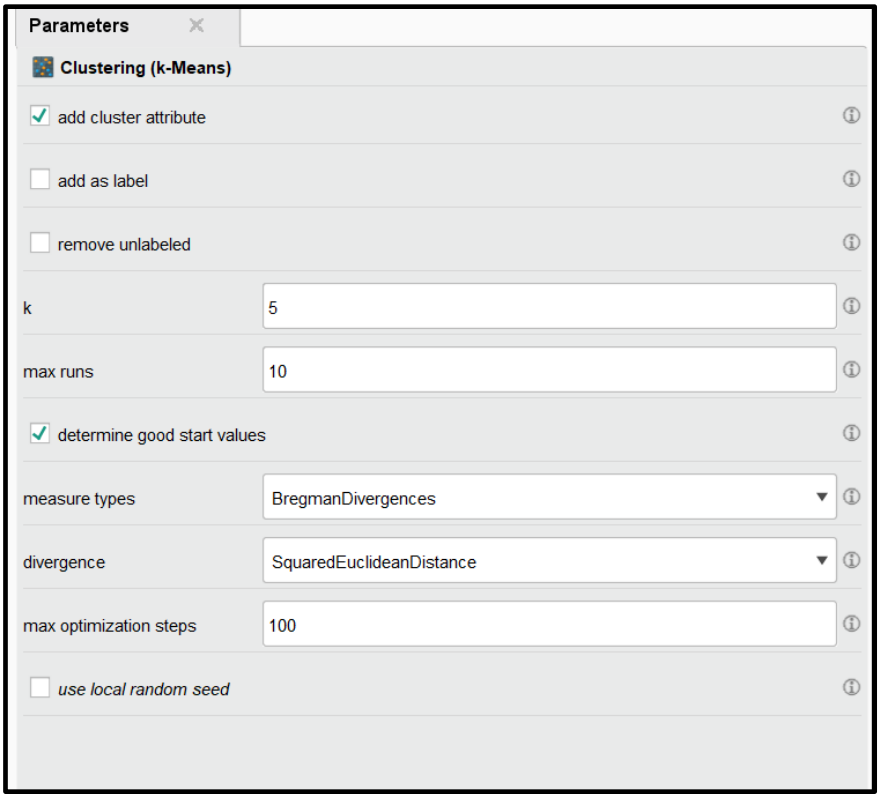
Otra observación relevante dentro de esta sección del caso de prueba es que se puede observar, que este dataset no contiene ningún Missing Value. Por lo tanto, usar algún operador relacionado a los mismos no es necesario, pues no jugaría un papel relevante en el análisis.

Modelo elegido:

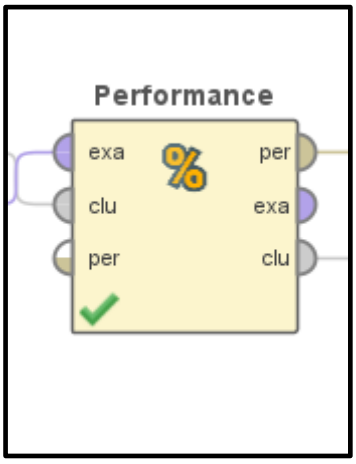
Usaremos el operador enfocado en clustering llamado “k-Means”. Este operador, tiene como objetivo principal agrupar un conjunto de datos en "K" clusters, donde "K" es un número predefinido. Cada cluster representa un grupo de puntos de datos que son similares entre sí en función de las características utilizadas en el análisis.



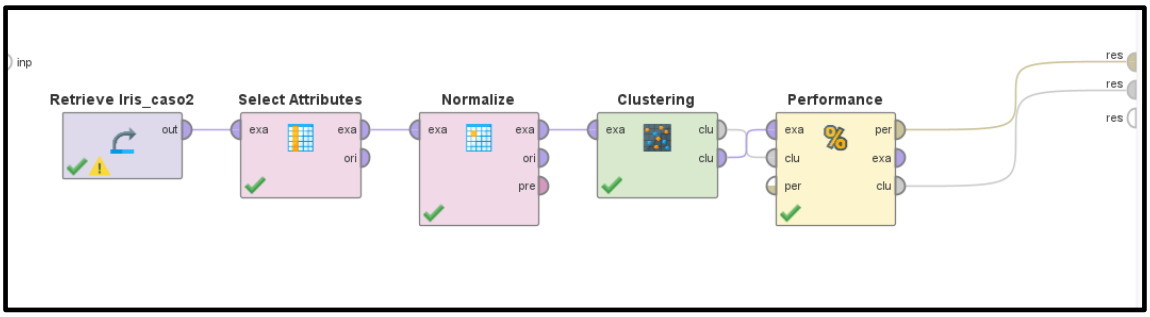
Con este nos centraremos en probar con diversos valores de k (modificable en los parámetros del operador).



También recurriremos al operador “Cluster Distance Performance”, este se utiliza para evaluar el rendimiento de un modelo de clustering, como K-Means, en función de cómo los datos se agrupan en diferentes clusters. Proporciona métricas que evalúan la calidad de los clusters formados por el modelo.

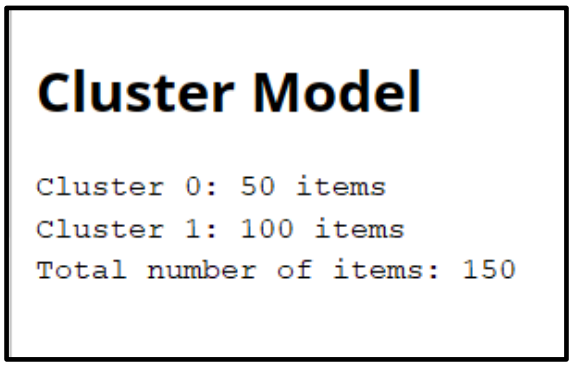


La estructura final del proceso debería ser la siguiente:

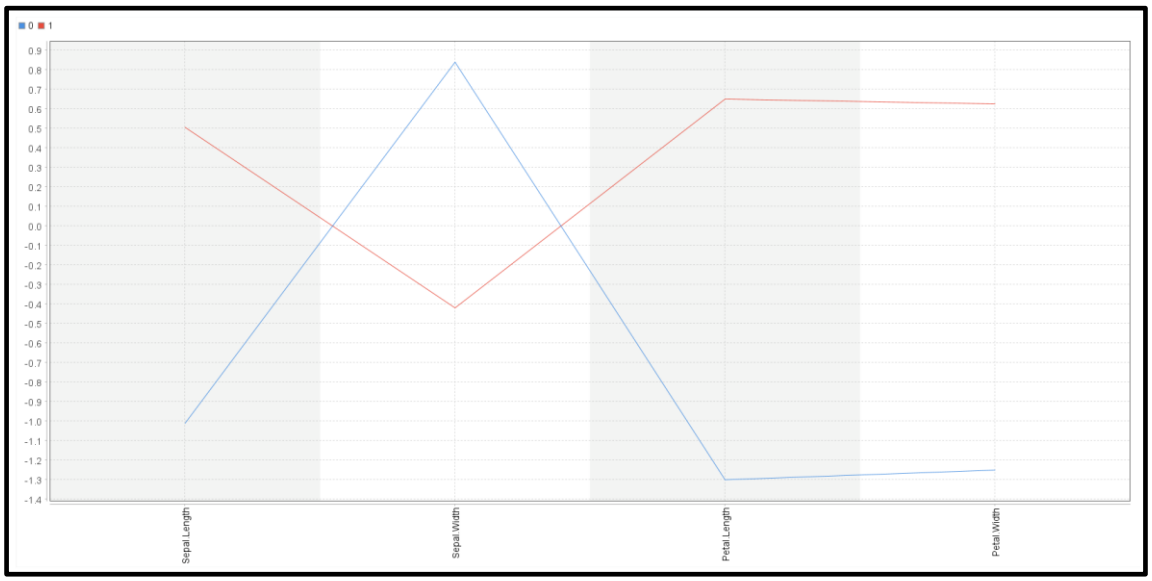


Probaremos el proceso con diversos valores de k, esto con la finalidad de ver para que valor del mismo se ajusta mejor en relación a nuestro objetivo.

- k = 2:



Attribute	cluster_0	cluster_1
Sepal.Length	-1.011	0.506
Sepal.Width	0.839	-0.420
Petal.Length	-1.301	0.650
Petal.Width	-1.251	0.625



PerformanceVector

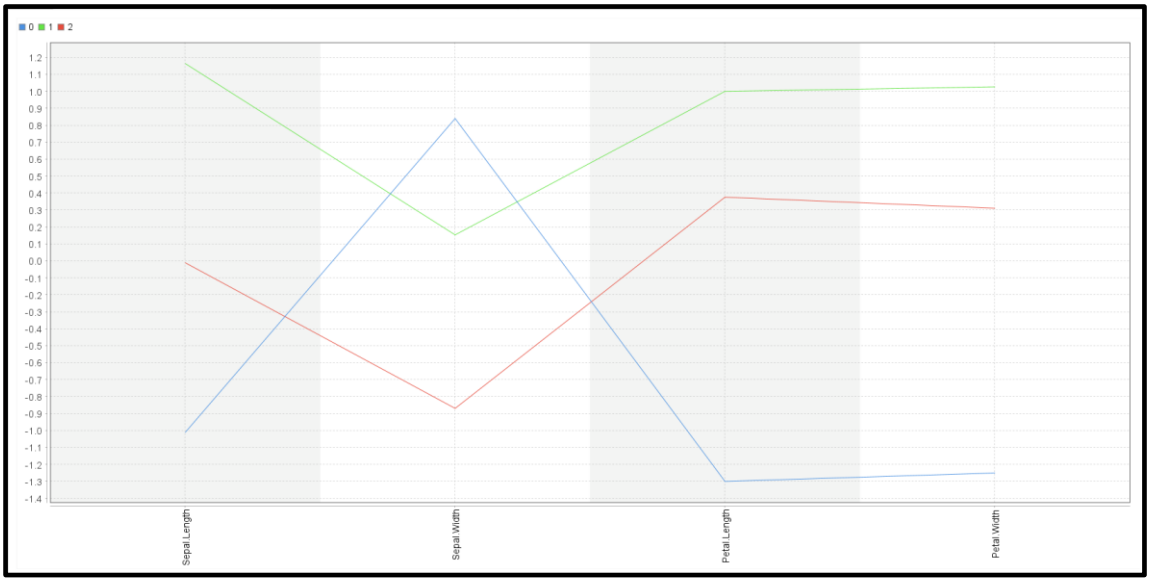
```
PerformanceVector:
Avg. within centroid distance: -1.482
Avg. within centroid distance_cluster_0: -0.963
Avg. within centroid distance_cluster_1: -1.741
Davies Bouldin: -0.598
```

- k = 3:

Cluster Model

```
Cluster 0: 50 items
Cluster 1: 44 items
Cluster 2: 56 items
Total number of items: 150
```

Attribute	cluster_0	cluster_1	cluster_2
Sepal.Length	-1.011	1.164	-0.011
Sepal.Width	0.839	0.153	-0.870
Petal.Length	-1.301	1.000	0.376
Petal.Width	-1.251	1.026	0.311



PerformanceVector

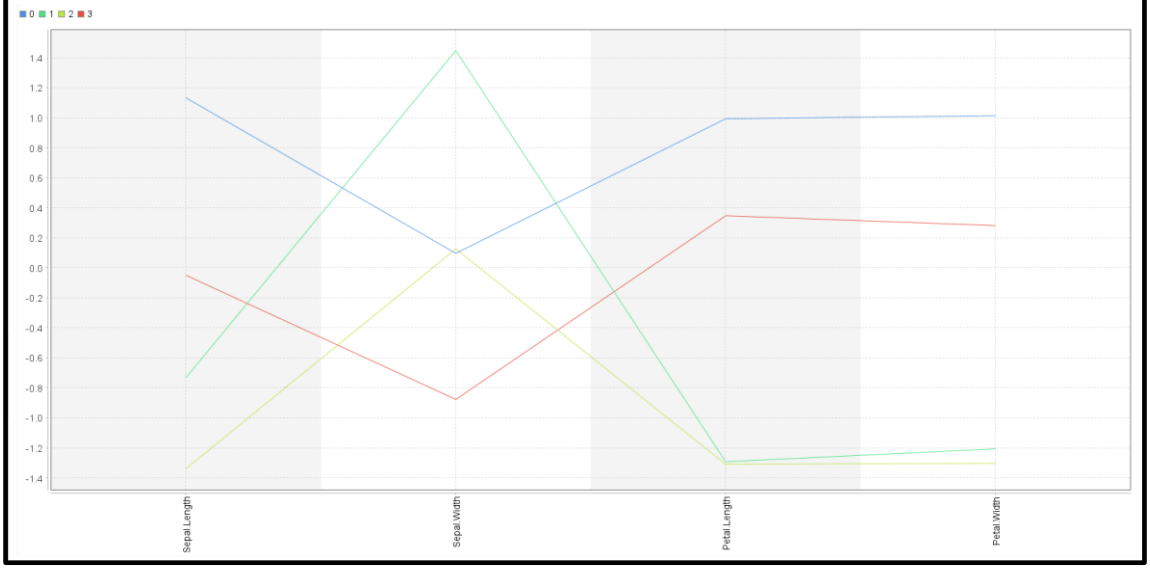
```
PerformanceVector:
Avg. within centroid distance: -0.935
Avg. within centroid distance_cluster_0: -0.963
Avg. within centroid distance_cluster_1: -0.988
Avg. within centroid distance_cluster_2: -0.867
Davies Bouldin: -0.834
```

- k = 4:

Cluster Model

```
Cluster 0: 47 items
Cluster 1: 27 items
Cluster 2: 23 items
Cluster 3: 53 items
Total number of items: 150
```

Attribute	cluster_0	cluster_1	cluster_2	cluster_3
Sepal.Length	1.132	-0.732	-1.339	-0.050
Sepal.Width	0.096	1.447	0.126	-0.877
Petal.Length	0.993	-1.293	-1.310	0.346
Petal.Width	1.014	-1.207	-1.303	0.281



PerformanceVector

```
PerformanceVector:
Avg. within centroid distance: -0.758
Avg. within centroid distance_cluster_0: -1.013
Avg. within centroid distance_cluster_1: -0.504
Avg. within centroid distance_cluster_2: -0.356
Avg. within centroid distance_cluster_3: -0.835
Davies Bouldin: -0.866
```

Análisis de Resultados:

En el caso del dataset Iris, sabemos de antemano que tienes tres tipos de flores (setosa, versicolor y virginica) como etiquetas o clases. Si realizamos un análisis de clustering sin utilizar esta información de clase y tratar de descubrir agrupamientos naturales basados solo en las características, no tendría sentido probar con un número de clusters "k" mayor a 3.

El algoritmo k-Means agrupa datos en un número específico de clusters, y en este caso, lo lógico sería configurar "k" en 3, ya que sabes que hay tres clases reales en el conjunto de datos. Utilizar un número mayor de clusters solo dividiría los datos en grupos adicionales que no tienen un significado intrínseco en el contexto de las especies de iris (incluso los resultados para k= 3 son muy acordes a lo que refleja el dataset original sin eliminar las clases).

Sin embargo hacer esto se podría considerar “trampa”, o una ventaja un tanto injusta, pues no siempre sabremos de antemano cuantas clases tendremos en el dataset. Por lo tanto, para ser objetivos tomaremos dos métricas relevantes: el promedio de la distancia dentro del centroide (Avg. within centroid distance) y el índice Davies-Bouldin (Davies Bouldin):

- Avg. within centroid distance: Un promedio de la distancia dentro del centroide más bajo significa que los puntos dentro de cada cluster están más cerca entre sí, lo que sugiere que los clusters son más compactos y mejor definidos, lo cual se considera una característica deseable en el análisis de clustering.

- Davies-Bouldin: El índice Davies-Bouldin mide la calidad de los clusters formados por un algoritmo de clustering. Cuanto menor sea el valor del índice, mejor será la separación y definición de los clusters. En otras palabras, un valor bajo del índice Davies-Bouldin sugiere que los clusters son más distintos y mejor definidos entre sí, lo que es un objetivo deseado en el análisis de clustering.

Teniendo esto en cuento y que las distancias deben ser medidas en valor absoluto (pues no tiene sentido que sean negativas):

Dado estos tres vectores de rendimiento:

En términos del índice Davies-Bouldin, un valor más bajo en términos absolutos indica un mejor rendimiento del modelo de clustering. Por lo tanto, "PerformanceVector(k=2)" con un valor absoluto de 0.598 es el mejor, seguido por "PerformanceVector(k=3)" con un valor absoluto de 0.834 y luego "PerformanceVector(k=4)" con un valor absoluto de 0.866.

En cuanto al promedio de la distancia dentro del centroide, un valor más bajo en términos absolutos indica un mejor rendimiento. En este aspecto, "PerformanceVector(k=4)" con un valor absoluto de 0.758 es el mejor, seguido por "PerformanceVector(k=3)" con un valor absoluto de 0.935 y luego "PerformanceVector(k=2)" con un valor absoluto de 1.482.

Posibles mejoras por evaluar:

- Uso de métricas múltiples: En lugar de depender de una sola métrica, considera utilizar varias métricas de evaluación para obtener una imagen más completa del rendimiento de tu modelo. Algunas métricas pueden ser más apropiadas para ciertos tipos de datos o problemas.

- Validación cruzada: La validación cruzada, como la validación cruzada k-fold o la validación cruzada leave-one-out, puede ayudarte a evaluar la estabilidad del modelo y reducir el riesgo de sobreajuste. Esto es especialmente útil cuando tienes un conjunto de datos pequeño.

- Selección de características: Si estás trabajando con conjuntos de datos de alta dimensionalidad, considera técnicas de selección de características o reducción de dimensionalidad, como PCA (Análisis de Componentes Principales) o LDA (Análisis de Discriminante Lineal), para mejorar la precisión del modelo y reducir el tiempo de entrenamiento.