

Inteligencia artificial

Alejandro López Gómez

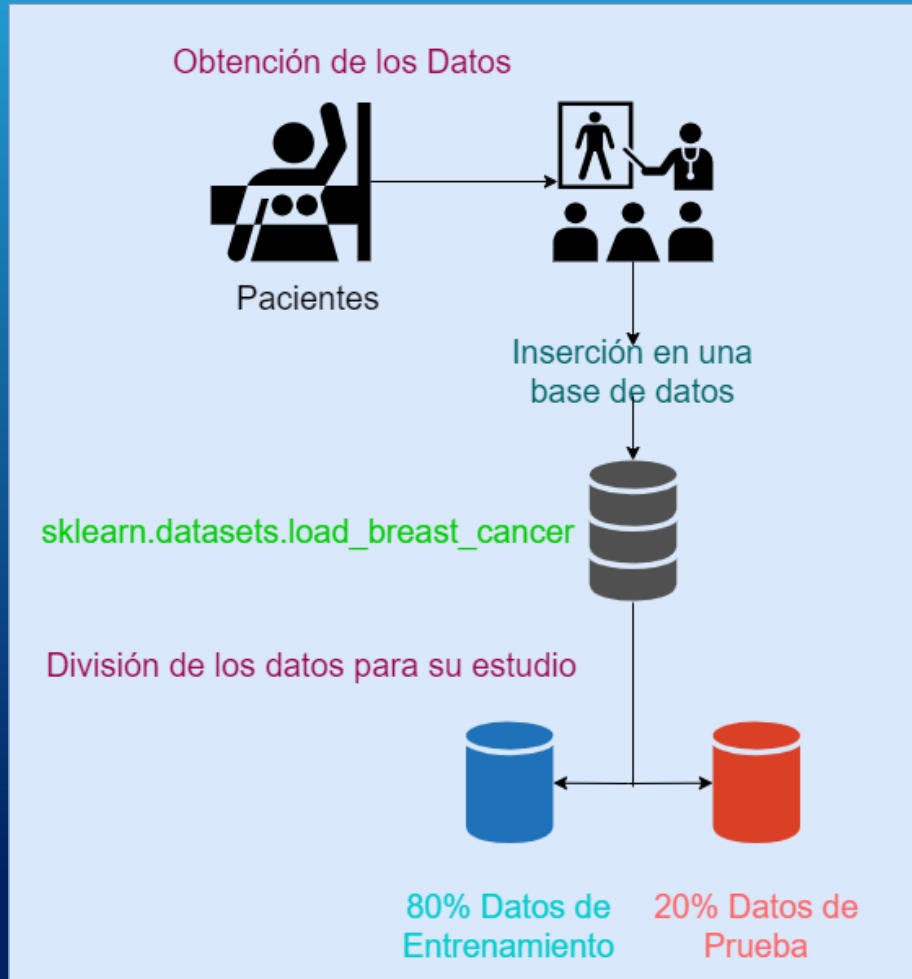
Emilio Macías Do Santos

Cristina Del Pilar Mallaupoma Cáceres

3º del GIC

Cargar los datos y librerías:





Explicación división de la muestra:

Se ha realizado la división de los datos de la siguiente forma:

- 80% datos de entrenamiento.
- 20% datos de prueba.

Falsos Negativos en problemas de clasificación:

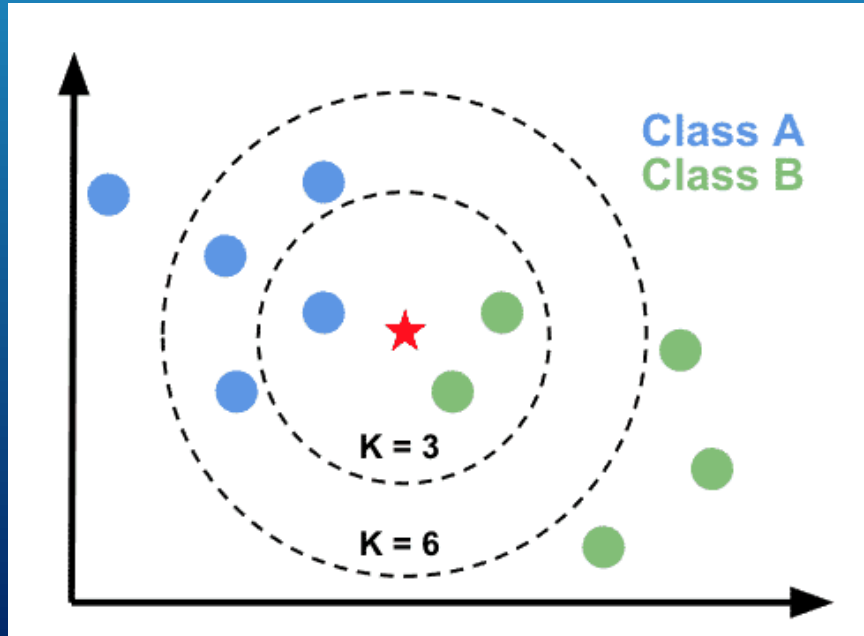
¿Cómo se calculan los Falsos Negativos?

VALORES PREDICCIÓN	VALORES REALES	
	Positivo	Negativo
Positivo	Verdaderos positivos	Falsos Positivos
Negativo	Falsos Negativos	Verdaderos Negativos

Utilizando el modelo anterior y habiendo dividido los datos de entrenamiento y prueba en trozos de 5 elementos:

1. Ajustamos el modelo con los datos de entrenamiento.
2. Realizamos una predicción de los datos utilizando una aproximación.
3. Creamos una matriz de confusión con la predicción del modelo y los datos de prueba.
4. Obtenemos la media de la puntuación obtenida.

Explicación del modelo K-nn:



1. Nos creamos el modelo Knn.
2. Lo entrenamos con los datos de prueba.
3. Calculamos los Falsos Negativos y la puntuación obtenida de la validación cruzada.

Representación del modelo Knn:

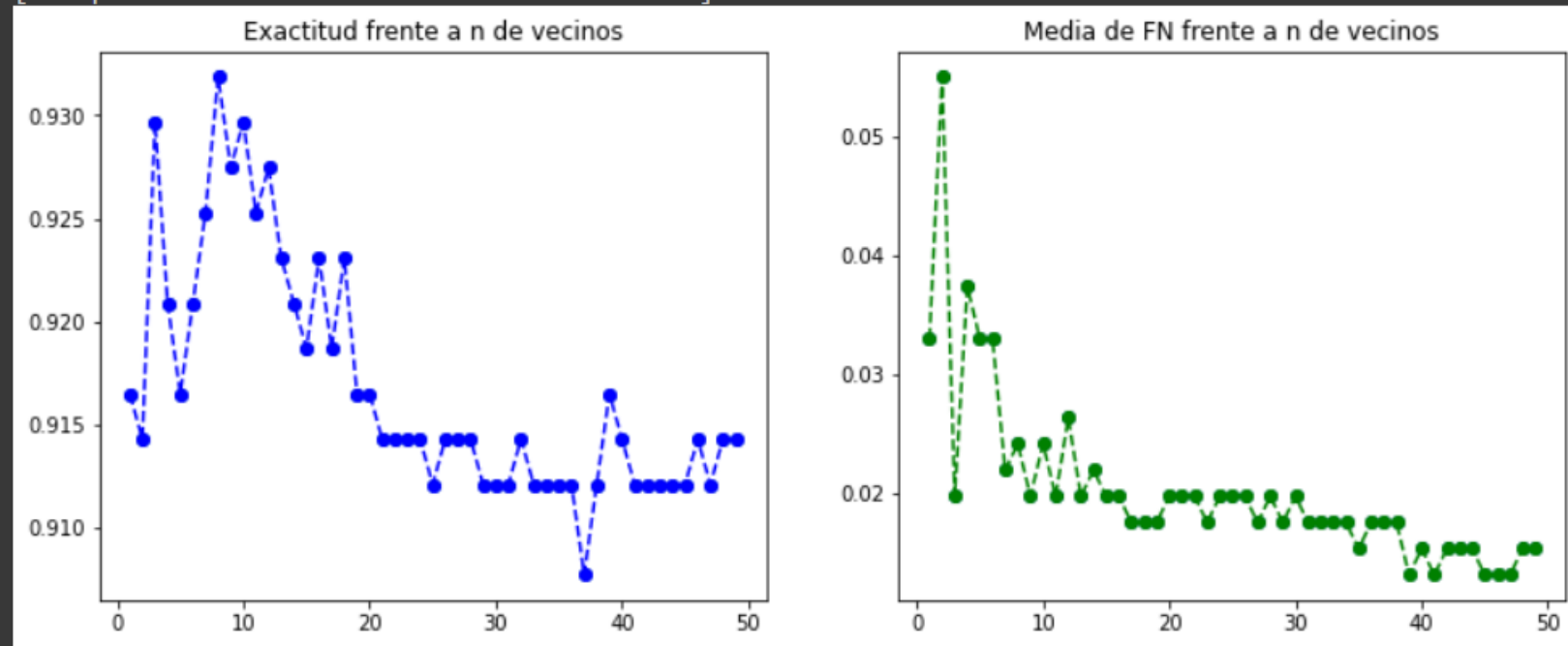
Train set accuracy: 0.931868

El mejor numero de vecinos para la Exactitud es: 8

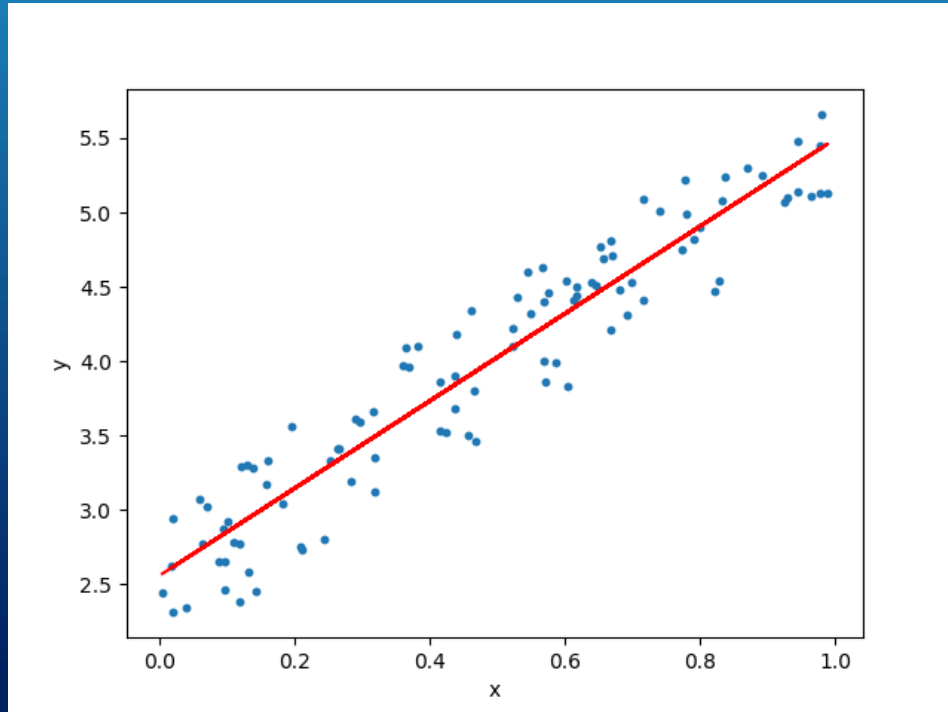
FN set accuracy: 0.013187

El mejor numero de vecinos para los FN es: 39

[<matplotlib.lines.Line2D at 0x7f0c3e6432e0>]



Explicación del modelo lineal:



1. Nos creamos el modelo lineal.
2. Adaptamos el conjunto de entrenamiento.
3. Ajustamos el modelo con los datos anteriores.
4. Calculamos el AIC (Criterio de Información de Akaike) y los Falsos Negativos.

Representación del modelo lineal:

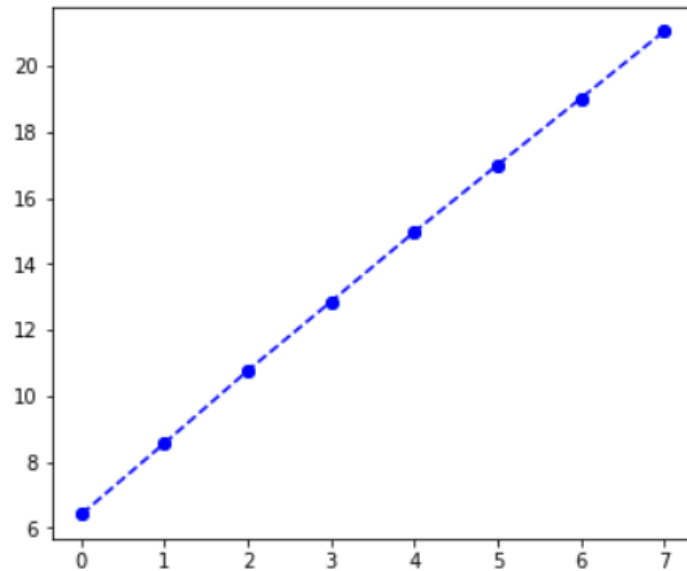
El mejor valor de AIC es: 6.416903

El mejor numero de parametros es: 1 , los cuales son $[-0.10023176]$

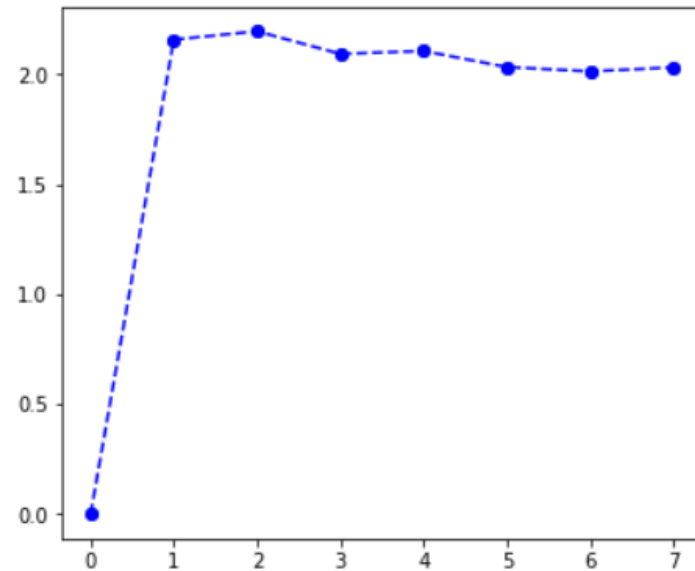
El mejor valor de FN es: 0.342857

El mejor numero de parametros es: 2 , los cuales son $[-0.09195942 \ -0.02254531]$

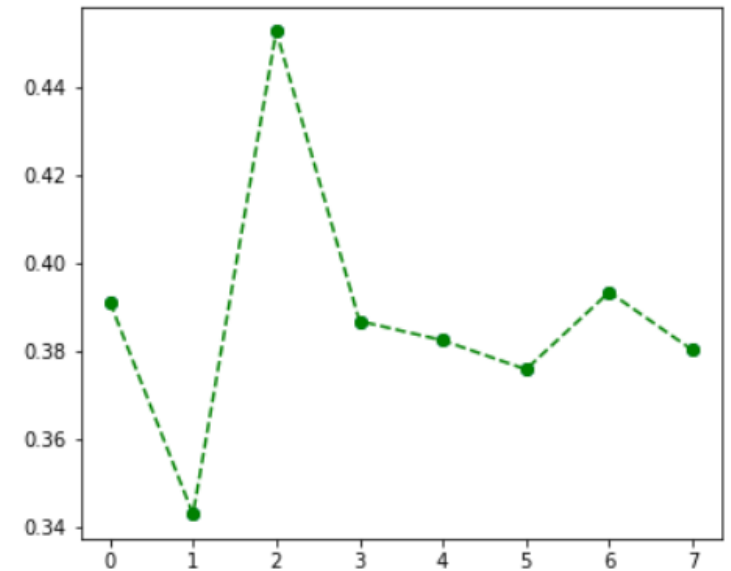
Valores de AIC:



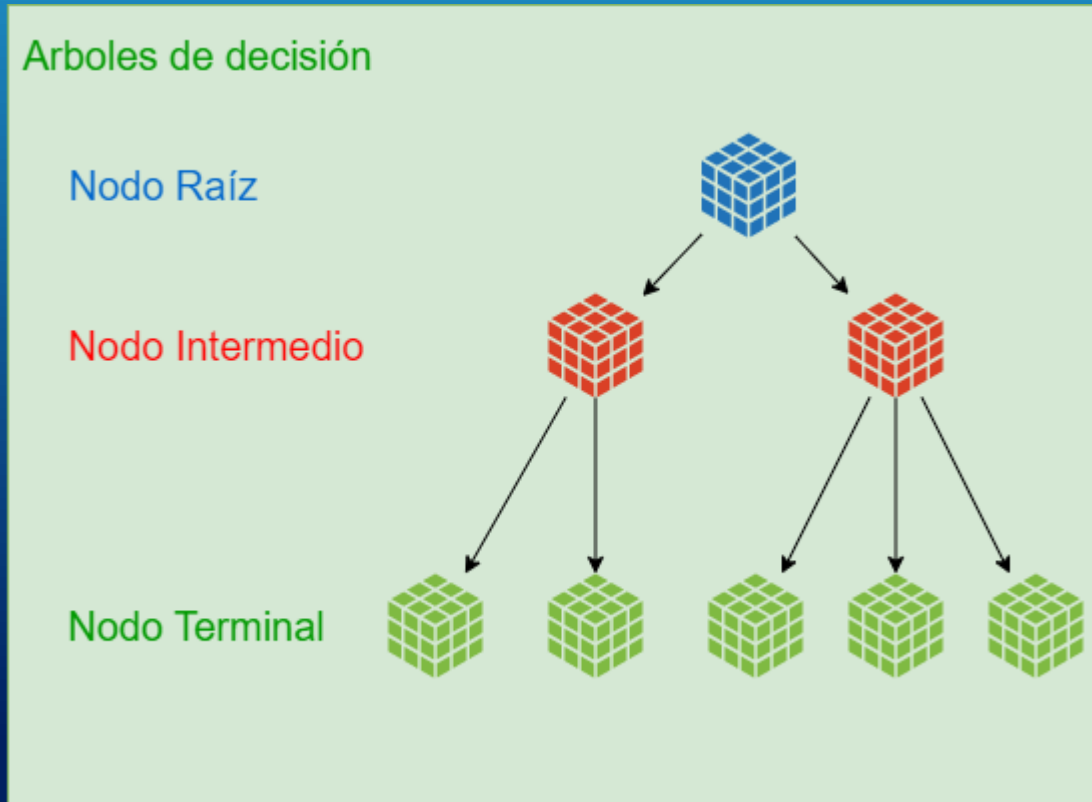
Pendiente del modelo



Media de FN frente a



Explicación del modelo de árboles de decisión:



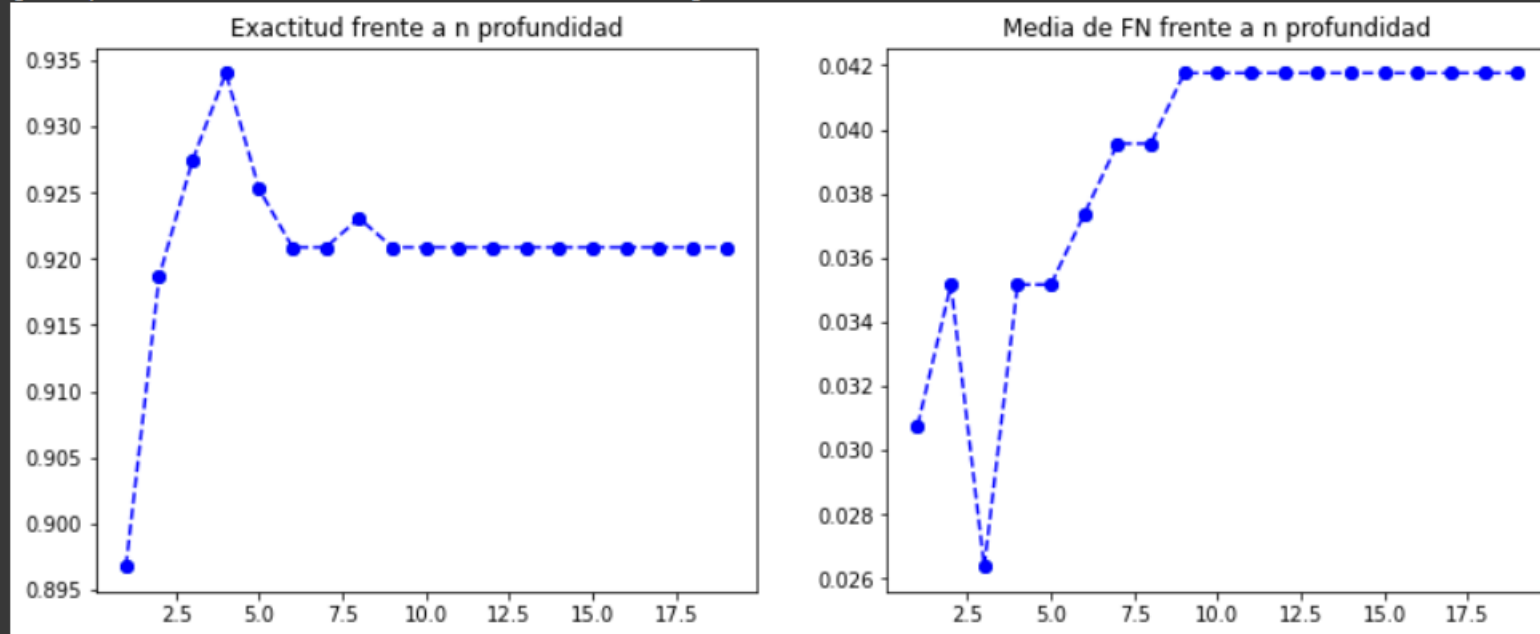
1. Nos creamos el modelo de árboles de decisión.
2. Entrenamos el modelo.
3. Obtenemos una predicción de los datos.
4. Calculamos los Falsos Negativos.
5. Calculamos su precisión utilizando la media de la validación cruzada de los datos obtenidos.

Representación del modelo de árboles de decisión:

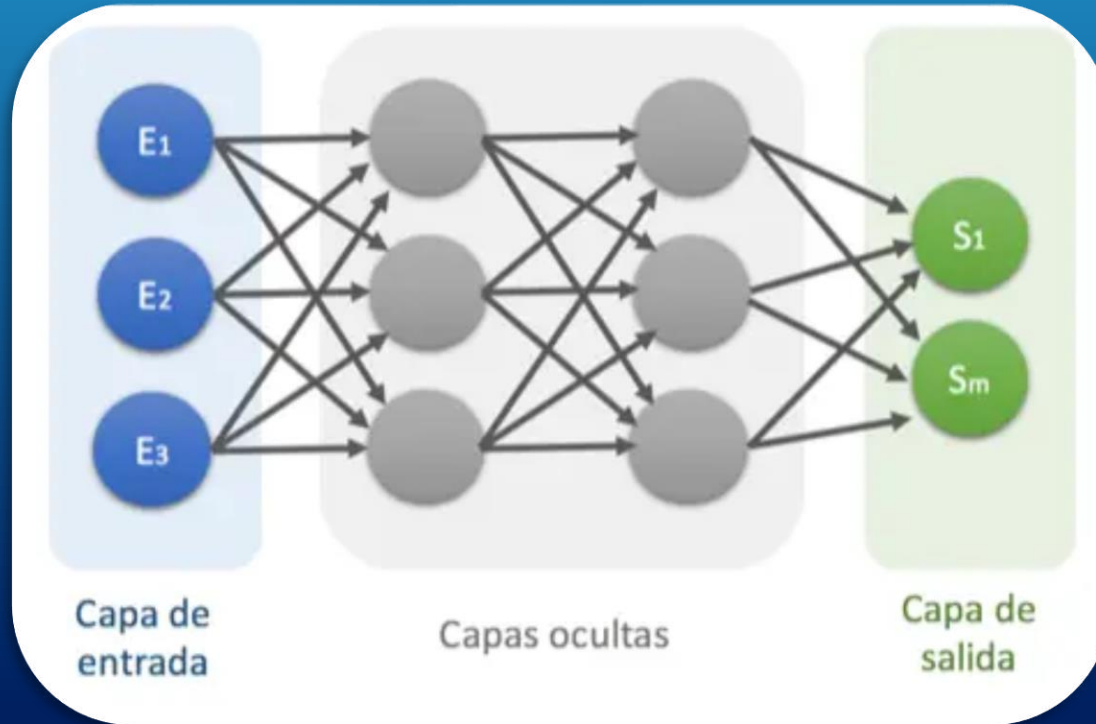
```
Train set accuracy: 0.934066  
El mejor valor de profundidad es: 4
```

```
FN set accuracy: 0.026374  
El mejor valor de profundidad para los FN es: 3
```

```
[<matplotlib.lines.Line2D at 0x7f0c3e99dc70>]
```



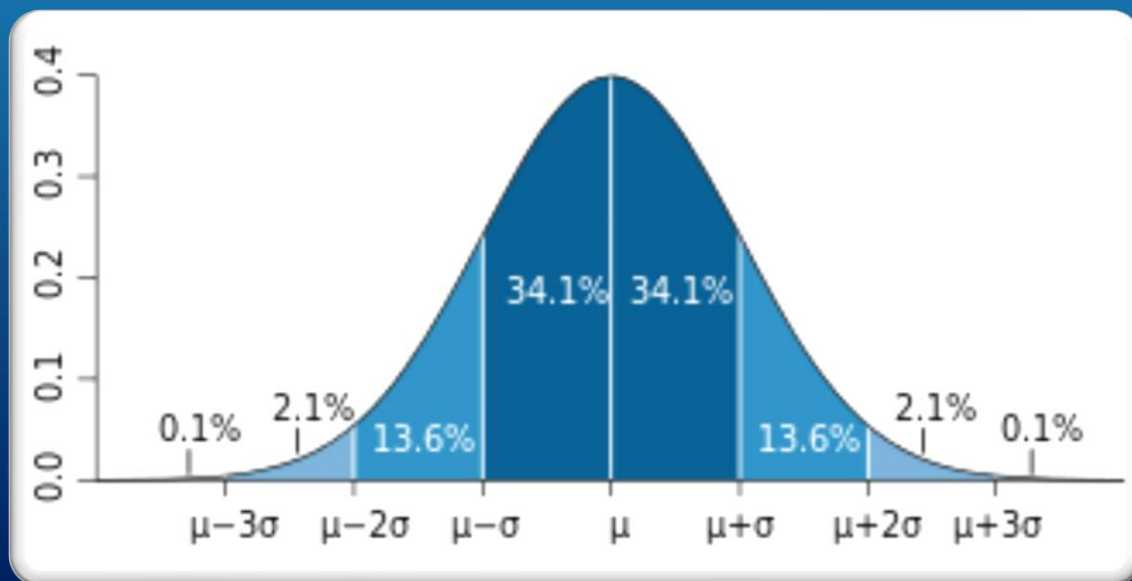
Explicación del modelo RNA Multicapa (Parte 1):



1. Nos creamos el modelo RNA Multicapa.
2. Entrenamos los modelos utilizando datos de entrenamiento estandarizados.
3. Obtenemos la puntuación del modelo utilizando los datos de prueba.
4. Obtenemos la precisión del modelo utilizando la validación cruzada de los datos de entrenamiento.

Explicación del modelo RNA Multicapa (Parte 2):

¿Por qué estandarizamos/escalamos los datos?



- Se realiza el escalado de los datos debido a que aumenta la precisión del modelo.
- El escalado se realiza con los datos de entrenamiento y prueba de X.

Representación del modelo RNA Multicapa:

100%|██████████| 3/3 [01:24<00:00, 28.18s/it]

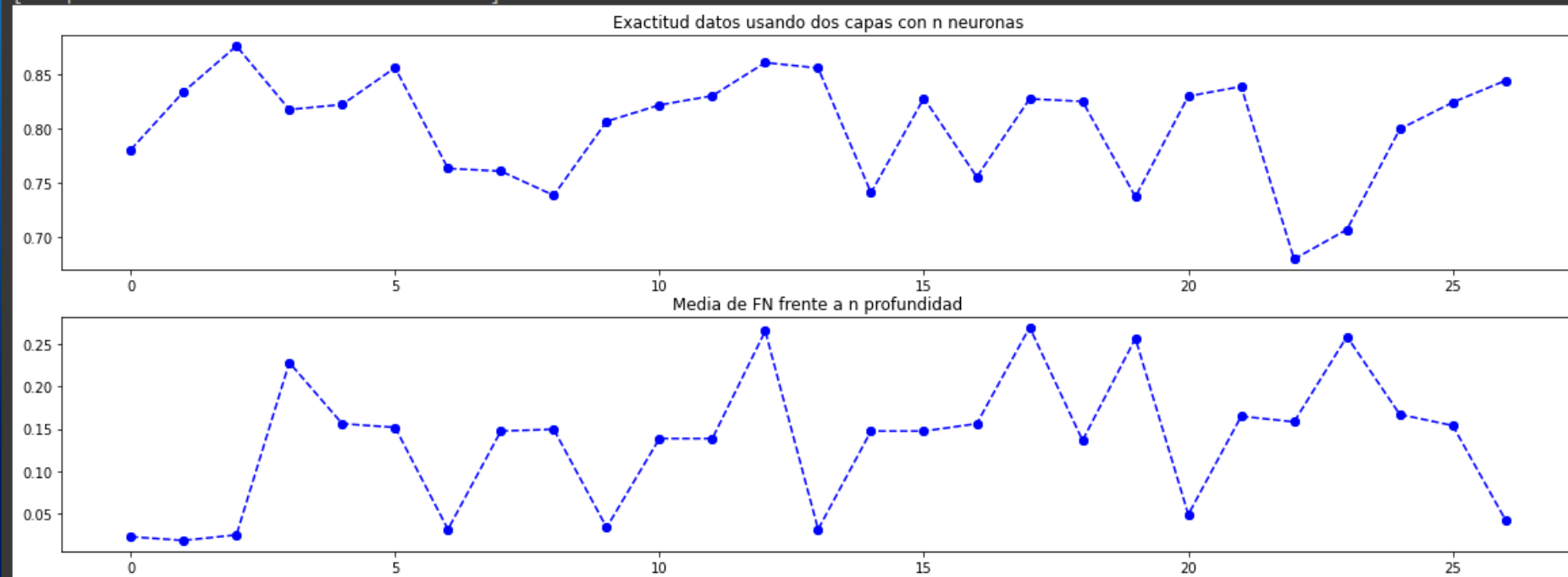
Train set accuracy: 0.875931

El mejor valor de numero de capas ocultas es: [5, 5] y tda: 0.0005

FN set accuracy: 0.017582

El mejor valor de numero de capas ocultas es: [5, 5] y tda: 0.0003

[<matplotlib.lines.Line2D at 0x7fa738144d90>]



Comparación de modelos (Parte 1):

Exactitud de los modelos:

```
Exactitud:  
Exactitud knn: 93.187 %  
Exactitud lineal: 60.0 %  
Exactitud árbol 93.407 %  
Exactitud rna: 96.923 %
```

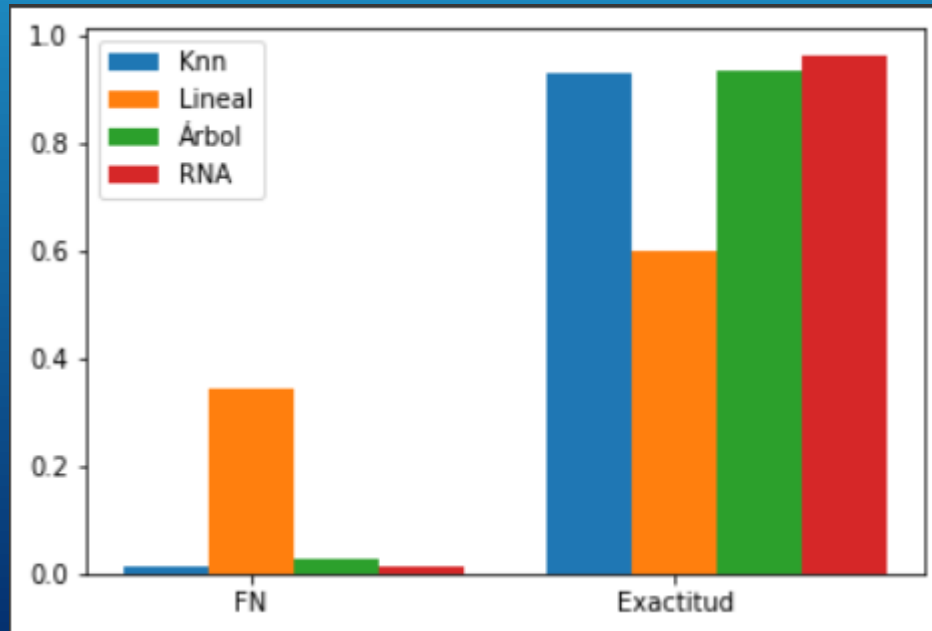
Falsos Negativos en los modelos:

```
FN:  
Probabilidad de FN en knn: 0.013  
Probabilidad de FN en lineal: 0.343  
Probabilidad de FN en árbol 0.026  
Probabilidad de FN en redes neuronales: 0.015
```

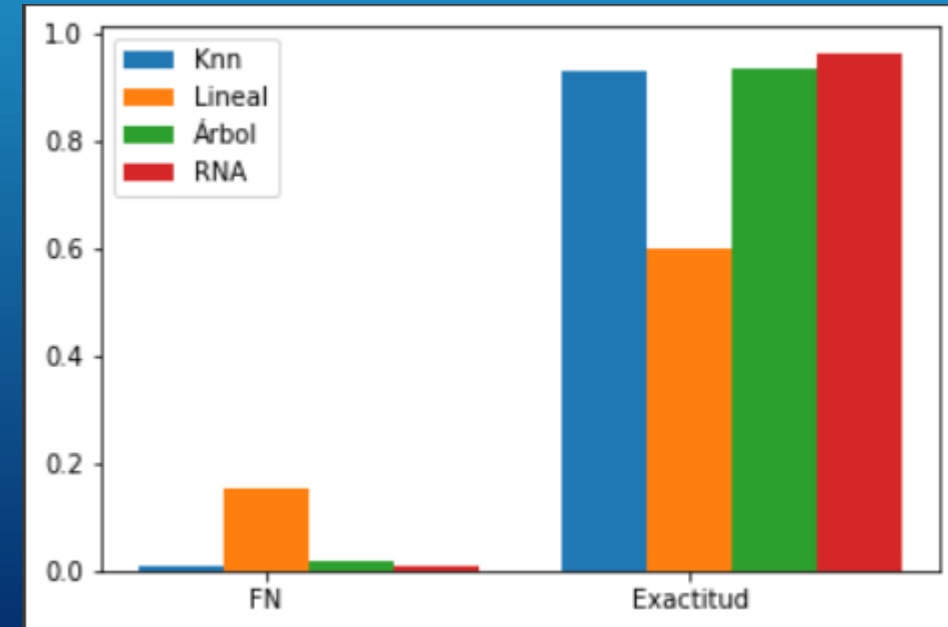
Se puede observar que el modelo Lineal es el modelo más ineficiente debido a que genera mayor probabilidad de Falsos Negativos y tiene la menor exactitud.

Comparación de modelos (Parte 2):

Datos de entrenamiento



Datos de test



A continuación podemos observar que el peor modelo en función a los Falsos Negativos y la exactitud es el *modelo lineal*. El mejor modelo son las *redes neuronales*, ya que tienen la menor probabilidad y la mayor exactitud.

Extensión de los modelos a otros problemas médicos (Parte 1):

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

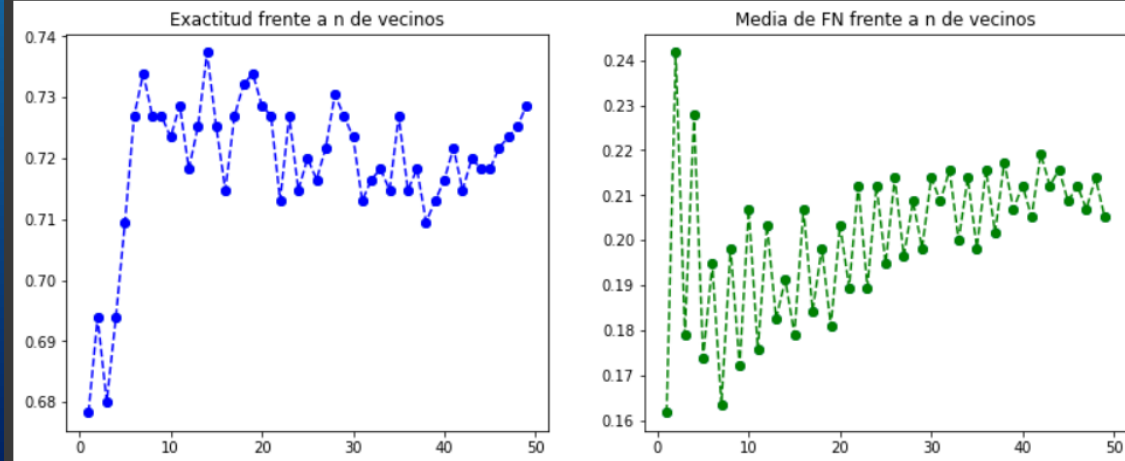
Se ha utilizado una base de datos con la siguiente estructura.

Extensión de los modelos a otros problemas médicos (Parte 2):

```
Test set accuracy: 0.737391  
El mejor numero de vecinos para la Exactitud es: 14
```

```
Test set accuracy: 0.161739  
El mejor numero de vecinos para los FN es: 1
```

```
[<matplotlib.lines.Line2D at 0x7f0c3dd2a400>]
```

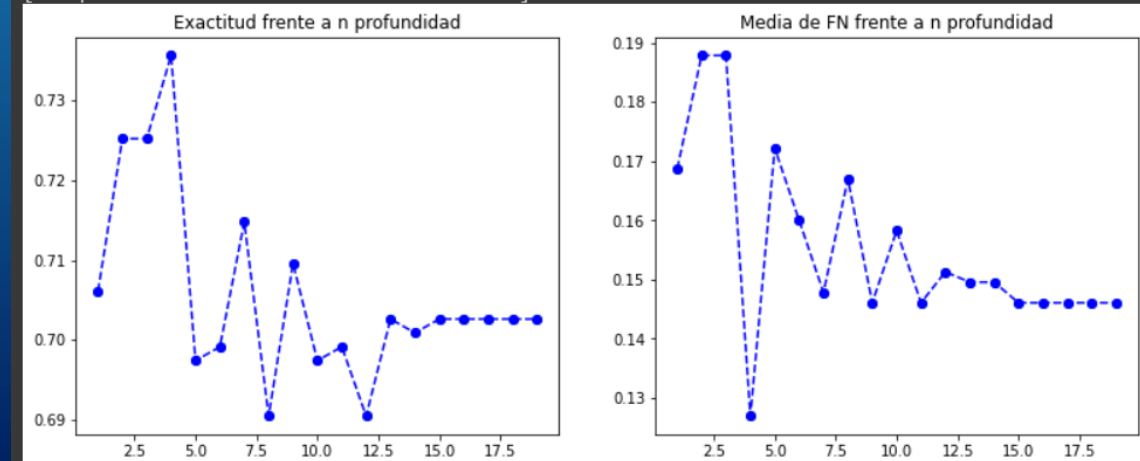


Modelo Knn

```
Train set accuracy: 0.735652  
El mejor valor de profundidad es: 4
```

```
FN set accuracy: 0.126957  
El mejor valor de profundidad para los FN es: 4
```

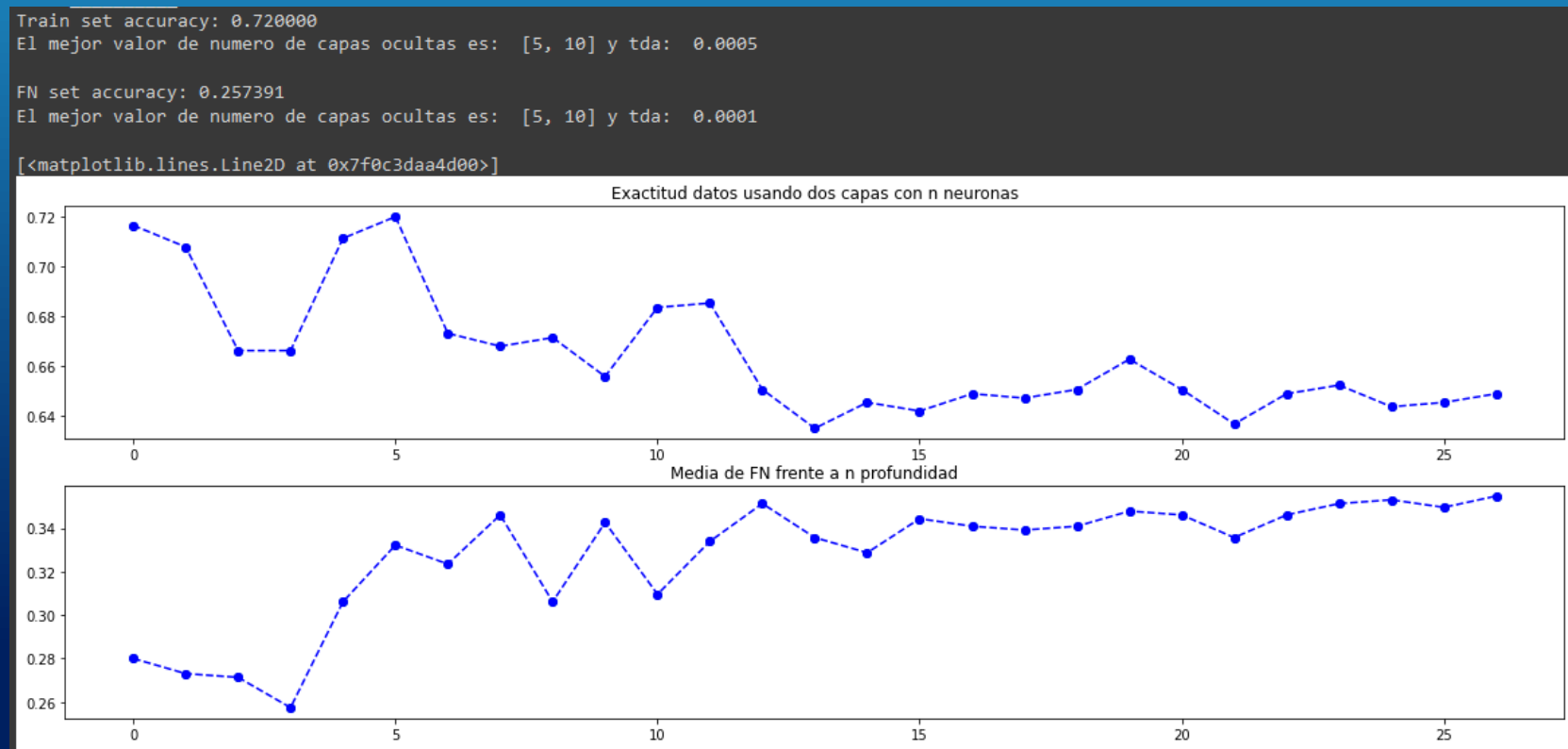
```
[<matplotlib.lines.Line2D at 0x7f0c3db68940>]
```



Modelo de Árboles de Decisión

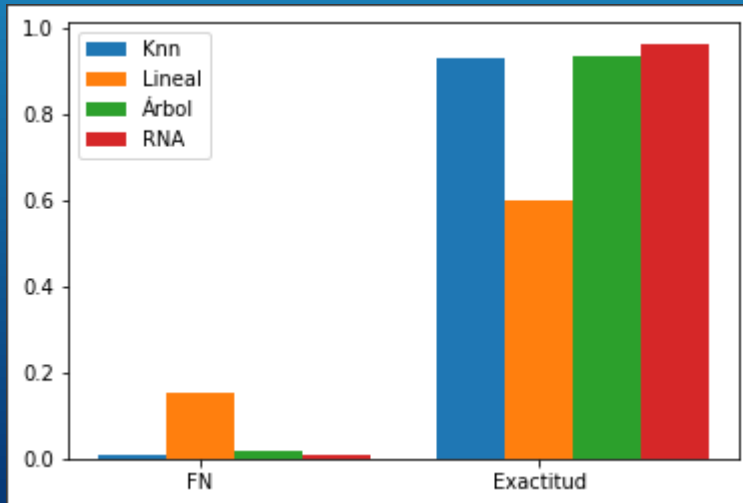
Extensión de los modelos a otros problemas médicos (Parte 3):

- La base de datos utilizada es sobre un estudio para determinar si una paciente embarazada tiene o no diabetes.
- Lo que comprobamos, con esta nueva base de datos, los resultados se mantienen alrededor del 70%. En cambio, con la primera base de datos la exactitud no baja del 80%.
- Lo único en lo que se diferencia es el número de parámetros. En esta nueva base de datos solo hay 8 parámetros mientras en la anterior hay 30.

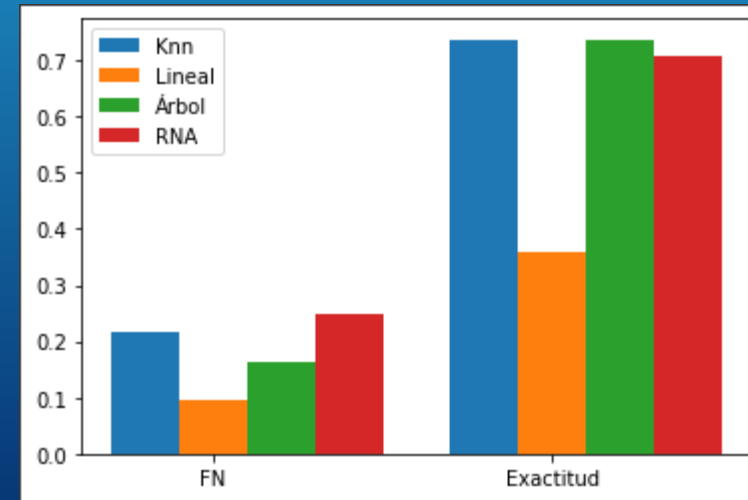


Modelo RNA Multicapa

Conclusiones :



1º Base de Datos



2º Base de Datos

- A la hora de predecir en ambos conjuntos utilizamos los conjuntos de test.
- Nuestros modelos creados se ajustan mejor a la 1º base de datos que a la 2º.
- En la 2º Base de Datos tenemos menos precisión y mayor probabilidad de Falsos Negativos.

Fin

Alejandro López Gómez

Emilio Macías Do Santos

Cristina Del Pilar Mallaupoma Cáceres