



FCyT
Sede Concepción
del Uruguay

Tesina de grado

Licenciatura en sistemas de información

Alumno: Iván Matías Acevedo

Directora: Daniela López De Luise

Módulo de predicción con minería de datos aplicado al mercado de divisas



Introducción

Este módulo de minería de datos es un recomendador, que como usuarios, los traders podrían utilizar como una herramienta más de confirmación para realizar una compra o venta en el mercado de divisas.



Objetivos

Objetivo general:

- Realizar recomendaciones de compras o ventas a traders que operan en el mercado de divisas Forex, mediante diferentes modelos de predicción.

Objetivos específicos:

- A partir de datos históricos, seleccionar variables relevantes y compilar datos.
- Analizar datos y desarrollar dos modelos predictivos de adaptación a series temporales.
- Mediante weka utilizar los modelos de predicción desarrollados mostrando B (si el precio baja), I (si se mantiene) o S (si sube).



Propósito

Implementar una herramienta liviana y práctica, fácilmente parametrizable y con características de autoajuste para ponerla a disposición de la industria y el público como un servicio.



Mercados financieros

Los Mercados Financieros, es el lugar donde confluye la oferta y la demanda de Activos Financieros. Si de la confluencia de la oferta y demanda existe acuerdo en el precio, cantidad y fecha de liquidación, nace formalmente una operación financiera negociando el activo financiero.

Los activos financieros que se comercializan dependen del tipo de mercado al cual pertenecen.

Mercados financieros

Mercados de Activos Tradicionales:

- Mercados de depósitos interbancarios
- Mercados de Renta Fija
- Mercados de Renta Variable
- **Mercados de Divisas**

Mercados de Instrumentos Derivados:

- Mercados de Futuros
 - Mercados de Opciones
 - Mercados de Swaps
-

Mercado de divisas

Consiste en la compra-venta de divisas (monedas), esto es, dinero. Las divisas son comerciadas a través de un broker y son comerciadas en pares, por ejemplo euro y dólar estadounidense: (par EUR/USD). Otras divisas como: GBPUSD, USDJPY, EURCHF, entre otros.



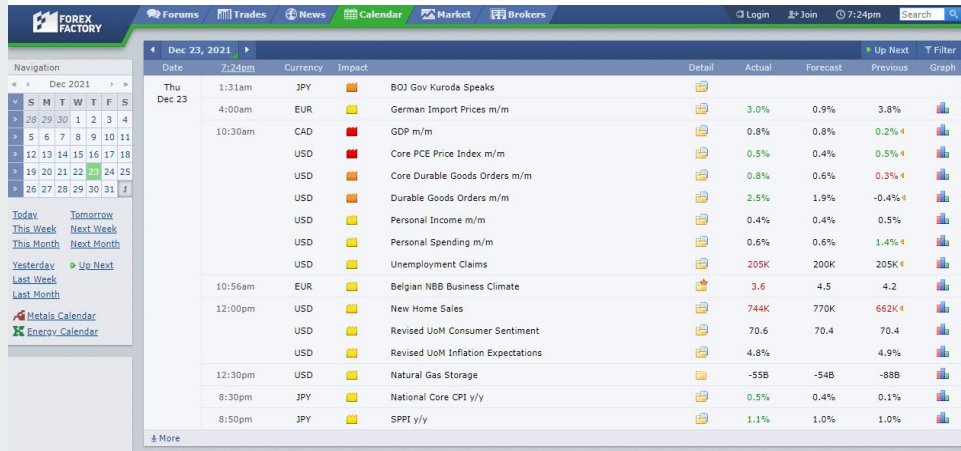
USDARS

Ejemplo de una divisa o
par exóticos.



Análisis Fundamental

Es un método que se emplea para evaluar el valor intrínseco de un activo y para analizar los factores que podrían influir en su precio en el futuro. Este tipo de análisis se basa en la evaluación de los activos a partir de hechos e influencias externos, así como de los estados financieros y de las tendencias industriales.



The screenshot displays the FOREX FACTORY website interface. The top navigation bar includes links for Forums, Trades, News, Calendar, Market, and Brokers. The main content area is titled "Dec 23, 2021" and shows a list of economic events. The table includes columns for Date, Time, Currency, Impact, Detail, Actual, Forecast, Previous, and a Graph icon. The events listed are for Thursday, December 23rd, starting at 1:31am with BOJ Gov Kuroda Speaks (JPY) and continuing through 8:50pm with SPPI y/y (JPY). Each event entry includes a color-coded impact icon (green for positive, red for negative, yellow for neutral) and a small bar chart icon.

Date	Time	Currency	Impact	Detail	Actual	Forecast	Previous	Graph
Thu Dec 23	1:31am	JPY	🟢	BOJ Gov Kuroda Speaks				
	4:00am	EUR	🟡	German Import Prices m/m	3.0%	0.9%	3.8%	
	10:30am	CAD	🔴	GDP m/m	0.8%	0.8%	0.2%	
		USD	🔴	Core PCE Price Index m/m	0.5%	0.4%	0.5%	
		USD	🟢	Core Durable Goods Orders m/m	0.8%	0.6%	0.3%	
		USD	🟢	Durable Goods Orders m/m	2.5%	1.9%	-0.4%	
		USD	🟡	Personal Income m/m	0.4%	0.4%	0.5%	
		USD	🟡	Personal Spending m/m	0.6%	0.6%	1.4%	
		USD	🟡	Unemployment Claims	205K	200K	205K	
	10:56am	EUR	🟡	Belgian NBB Business Climate	3.6	4.5	4.2	
	12:00pm	USD	🟡	New Home Sales	744K	770K	662K	
		USD	🟡	Revised UoM Consumer Sentiment	70.6	70.4	70.4	
		USD	🟡	Revised UoM Inflation Expectations	4.8%		4.9%	
	12:30pm	USD	🟡	Natural Gas Storage	-55B	-54B	-88B	
	8:30pm	JPY	🟡	National Core CPI y/y	0.5%	0.4%	0.1%	
	8:50pm	JPY	🟡	SPPI y/y	1.1%	1.0%	1.0%	

Análisis Técnico

El análisis técnico es un sistema que permite examinar y predecir los movimientos de precios en los mercados financieros a partir de datos históricos, estadísticas de mercado y patrones gráficos.





Planteo del problema

El mercado de Forex está influenciado por múltiples acontecimientos mundiales que afectan a la oferta y la demanda del mismo. De esta manera pronosticar la suba o baja del precio no solo depende de ciertas variables como, un análisis sobre patrones **técnicos** a lo largo del tiempo, sino también de otros datos **fundamentales** como sucesos sociales, decisiones políticas de un país o un banco central, el PBI de un país, inflación, etc.

Estas múltiples variables son un problema para los traders a la hora de ingresar al mercado, no solo por la cantidad sino también por la complejidad de interpretación que tienen.



Solución

- Se buscará crear un servicio que sea un recomendador para los traders que son quienes operan en los mercados de financieros.
- Este servicio no será la única herramienta utilizada por los traders sino que será un apoyo más como confirmación, a la hora de tomar decisiones de compra o ventas.
- Será un módulo inteligente desarrollado mediante datos históricos donde se pueda saber si el precio va a bajar, quedar igual o subir dentro de los próximos 15 minutos.

Metodologías y herramientas

- Inteligencia Artificial
- Minería de datos para series temporales
- Metodos de prediccion:
 - Árbol de inducción: J48
 - Red neuronal: MultilayerPerceptron
 - Metacluster: Make Density Based Clustering
- Selector de atributos:
 - Ranker
- Weka
- Python

Antecedentes



Ventas de pasajes

El objetivo principal es reducir diferentes tipos de costos fundamentalmente los producidos por el combustible. Por tal motivo, poder pronosticar la demanda de pasajeros consiste en un trabajo fundamental. Este tipo de información permite tomar acertadamente decisiones tales como aumento y/o disminución de frecuencias, cambios de equipos, inversión en equipos nuevos, aumento y/o disminución de tarifas, entre otros.



Salud

Una de las prioridades de estudio es saber las futuras proyecciones del comportamiento de la mortalidad general intrahospitalaria para prever recursos humanos, de infraestructura, equipamiento tecnológicos y financieros y así reducir el incremento de muertes, motivo por el cual se tiene como objetivo determinar el mejor modelo de predicción mensual que se ajuste a la serie original para hacer predicciones a corto plazo.

Búsqueda de datos históricos

Historial de datos (últimos 10 días)

Calidad: una de las mejores fuentes de datos gratuitas

DataSet Inicial

	A	B	C	D	E	F	G
1	<TICKER>	<DTYYYYMM>	<TIME>	<OPEN>	<HIGH>	<LOW>	<CLOSE>
2	EURUSD	20210601	0	12.230	12.230	12.230	12.230
3	EURUSD	20210601	100	12.230	12.230	12.230	12.230
4	EURUSD	20210601	200	12.229	12.229	12.229	12.229
5	EURUSD	20210601	300	12.229	12.229	12.229	12.229
6	EURUSD	20210601	400	12.230	12.230	12.230	12.230
7	EURUSD	20210601	500	12.230	12.230	12.230	12.230
8	EURUSD	20210601	600	12.231	12.231	12.231	12.231
9	EURUSD	20210601	700	12.232	12.232	12.232	12.232
10	EURUSD	20210601	800	12.233	12.233	12.233	12.233
11	EURUSD	20210601	900	12.234	12.234	12.234	12.234
12	EURUSD	20210601	1000	12.233	12.233	12.233	12.233
13	EURUSD	20210601	1100	12.233	12.233	12.233	12.233
14	EURUSD	20210601	1200	12.233	12.233	12.233	12.233

Símbolo	Rango de datos	Tamaño
AUDJPY	Enero 2001 - 31/01/2022	35.8 MB
AUDUSD	Enero 2001 - 31/01/2022	33.4 MB
CHFJPY	Enero 2001 - 31/01/2022	36.0 MB
EURCAD	Enero 2001 - 31/01/2022	39.6 MB
EURCHF	Enero 2001 - 31/01/2022	34.0 MB
EURGBP	Enero 2001 - 31/01/2022	31.6 MB
EURJPY	Enero 2001 - 31/01/2022	39.8 MB
EURUSD	Enero 2001 - 31/01/2022	36.5 MB
GBPCHF	Enero 2001 - 31/01/2022	41.5 MB
GBPJPY	Enero 2001 - 31/01/2022	43.0 MB
GBPUSD	Enero 2001 - 31/01/2022	38.0 MB
NZDJPY	Enero 2003 - 31/01/2022	31.9 MB
NZDUSD	Enero 2003 - 31/01/2022	30.1 MB
USDCAD	Enero 2001 - 31/01/2022	33.4 MB
USDJPY	Enero 2001 - 31/01/2022	35.0 MB
USDCHF	Enero 2001 - 31/01/2022	36.1 MB
XAGUSD	Enero 2001 - 31/01/2022	16.2 MB
XAUUSD	Enero 2001 - 31/01/2022	30.1 MB

Desarrollo de modelos de datos



Proceso de DataSet Inicial

- Se desarrolló dos algoritmos en python para generar dos dataset y así poder crear dos modelos diferentes:
 - **Algoritmo 1: Agrupando de a 15 elementos.**
 - **Algoritmo 2: Restando el primer elemento.**
- Utilizando precio de cierre.
- Vectores de 15 elementos.



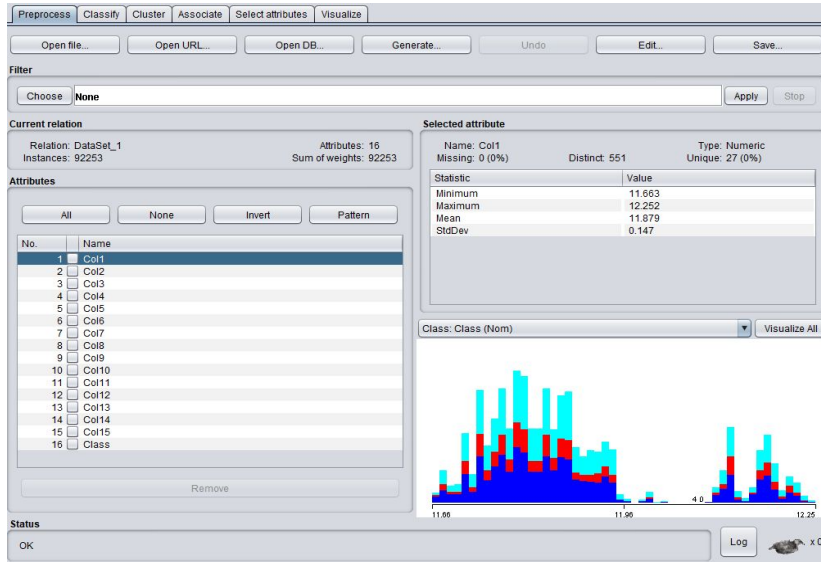
Algoritmo 1: Agrupando de a 15 elementos

Consiste en:

- 15 columnas, más la columna de variación llamada Class (B-I-S).
- La primer fila son los primeros 15 elementos partiendo del primero.
- Se segunda fila partiendo del segundo elemento los 15 elementos.
- Luego partiendo desde el tercer elemento...
- Este dataset se le asigna el nombre de *DataSet_1.csv*

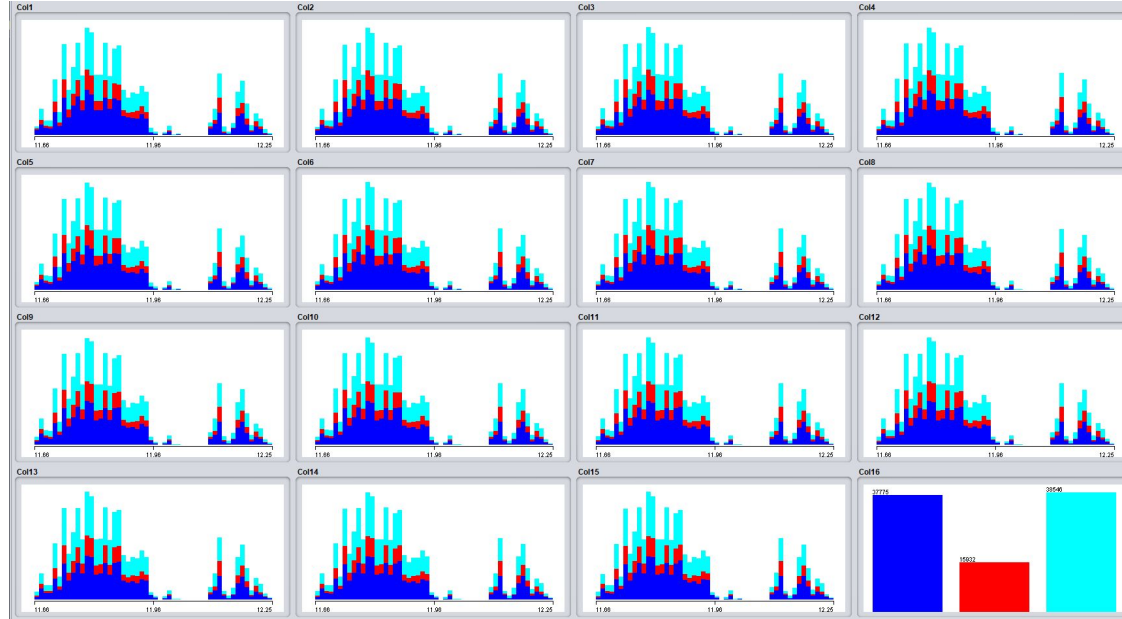
Cargando datos en weka

Informacion del dataset en weka



- 92.253 Instancias.
- Para S encontró 37.775, para I 15.932 y para B 38.546.
- Detecta un máximo de 12.252 un mínimo de 11.663 y un precio medio de 11.879.
- Datos perdidos 0% (Missing).
- Datos distintos 551 (Distinct), datos únicos 27 (Unique).
- El Name que indica el nombre de la columna.

Vista gráfica de datos en weka



Herramienta de calificación de atributos

Análisis de atributos

La herramienta Ranker retorna el ranking de columnas según cuánto podrían contribuir al momento de predecir. En el primer puesto queda la columna 1 llamada "Col1" y en el segundo puesto la columna 15 llamada "Col15".

La columna 1 y 15 se están utilizando para calcular la columna Class, que es "S" o "I" o "B" dependiendo si el precio de la columna 15 baja, sube o queda igual con respecto al precio de la columna 1.

The screenshot shows the Weka Explorer interface with the 'Attribute Evaluator' tab selected. The 'Choose' button is highlighted, and the 'InfoGainAttributeEval' method is selected. The 'Search Method' is set to 'Ranker -T 1.7976931348623157E308 -N 1'. The 'Attribute Selection Mode' is set to 'Use full training set'. The 'Attribute selection output' pane displays the results of the attribute selection process, including a list of ranked attributes and the selected attributes.

Attribute Evaluator

Choose **InfoGainAttributeEval**

Search Method

Choose **Ranker -T 1.7976931348623157E308 -N 1**

Attribute Selection Mode

☒ Use full training set
☐ Cross-validation Folds: 10 Seed: 1

(Nom) Col16

Start Stop

Attribute selection output **Resultado**

```
=== Attribute Selection on all input data ===  
  
Search Method:  
Attribute ranking.  
  
Attribute Evaluator (supervised, Class (nominal): 16 Col16):  
Information Gain Ranking Filter  
  
Ranked attributes:  
0.02932 1 Col1  
0.02253 15 Col15  
0.01488 2 Col2  
0.01117 14 Col14  
0.01084 3 Col3  
0.0099 4 Col4  
0.00846 12 Col12  
0.00817 13 Col13  
0.00737 5 Col5  
0.00697 6 Col6  
0.00675 7 Col7  
0.00674 10 Col10  
0.00615 8 Col8  
0.00605 9 Col9  
0.00583 11 Col11  
  
Selected attributes: 1,15,2,14,3,4,12,13,5,6,7,10,8,9,11 : 15
```

Status

OK Log x0

Creando modelo con Árbol de inducción y Red neuronal.

Árbol de inducción

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose **J48 -C 0.25 -M 500**

Test options

☐ Use training set

☐ Supplied test set Set...

☐ Cross-validation Folds 10

☒ Percentage split % 66

More options...

(Nom) Class

Start Stop

Result list (right-click for options)

batchSize 100

binarySplits False

collapseTree True

confidenceFactor 0.25

debug False

doNotCheckCapabilities False

doNotMakeSplitPointActualValue False

minNumObj 500

numDecimalPlaces 2

numFolds 3

reducedErrorPruning False

saveInstanceData False

seed 1

subtreeRaising True

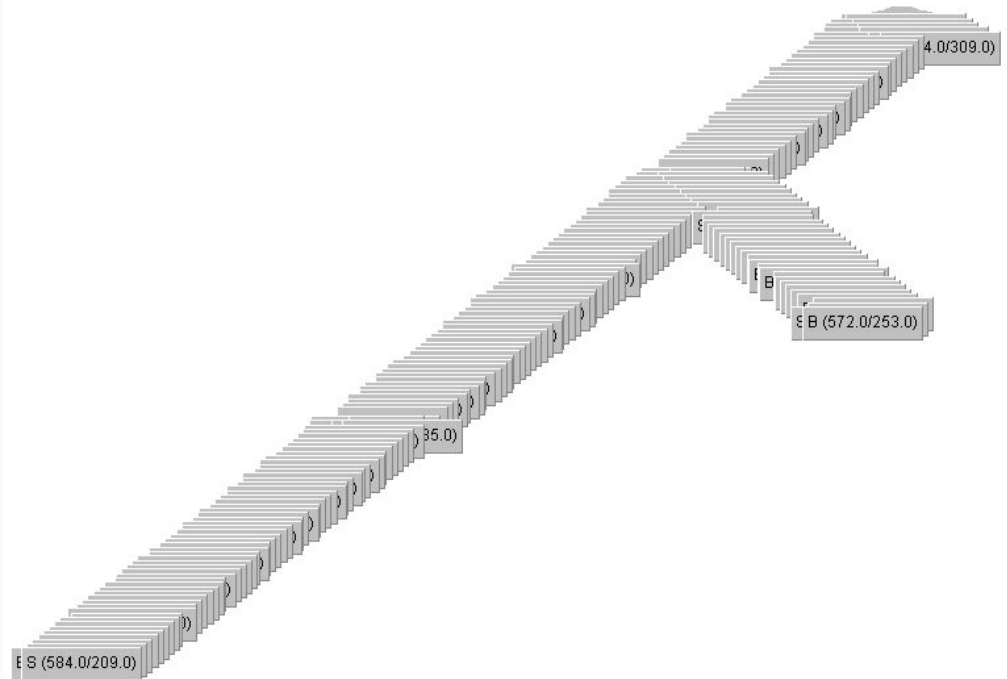
unpruned False

useLaplace False

useMDLcorrection True

Árbol de induccion

Resultado gráfico de procesar DataSet_1
con J48



Árbol de inducción

- Cuando fue “S”, clasificó como S 11665 veces (aciertos)
- Cuando Fué “B”, clasificó como “B” 11438 veces (aciertos).
- Cuando fue “I” se equivocó en todas.
- Clasifica un poco mejor las subas.

=== Summary ===

Correctly Classified Instances	23103	73.6562 %
Incorrectly Classified Instances	8263	26.3438 %
Kappa statistic	0.5514	
Mean absolute error	0.258	
Root mean squared error	0.3618	
Relative absolute error	61.6066 %	
Root relative squared error	79.0218 %	
Total Number of Instances	31366	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,904	0,239	0,725	0,904	0,805	0,654	0,895	0,807	S
	0,000	0,000	?	0,000	?	?	0,585	0,220	I
	0,880	0,209	0,748	0,880	0,809	0,661	0,897	0,819	B
Weighted Avg.	0,737	0,185	?	0,737	?	?	0,842	0,710	

=== Confusion Matrix ===

	a	b	c	<-- classified as
11665	0	1237		a = S
2865	0	2607		b = I
1554	0	11438		c = B

Red Neuronal

Preprocess Classify Cluster Associate Select

Classifier

Choose **MultilayerPerceptron** -L 0.3 -M 0.2 -N 50

Test options

☐ Use training set

☐ Supplied test set Set...

☐ Cross-validation Folds 10

☒ Percentage split % 66

More options...

(Nom) Class

GUI False

autoBuild True

batchSize 200

debug False

decay False

doNotCheckCapabilities False

hiddenLayers a

learningRate 0.3

momentum 0.2

nominalToBinaryFilter True

normalizeAttributes True

normalizeNumericClass True

numDecimalPlaces 2

reset True

resume False

seed 0

trainingTime 500

validationSetSize 0

validationThreshold 20

Red Neuronal

- Cuando fue “S”, clasificó como “S” 12900 veces (aciertos).
- Cuando Fué “B”, clasificó como “B” 11380 veces (aciertos).
- Cuando fue “I”, clasificó como “I” 3804 se equivocó en todas.
- Clasifica un poco mejor las subas.

Correctly Classified Instances	28084	89.5364 %
Incorrectly Classified Instances	3282	10.4636 %
Kappa statistic	0.8335	
Mean absolute error	0.0772	
Root mean squared error	0.2115	
Relative absolute error	18.4393 %	
Root relative squared error	46.1988 %	
Total Number of Instances	31366	

=== Detailed Accuracy By Class ===

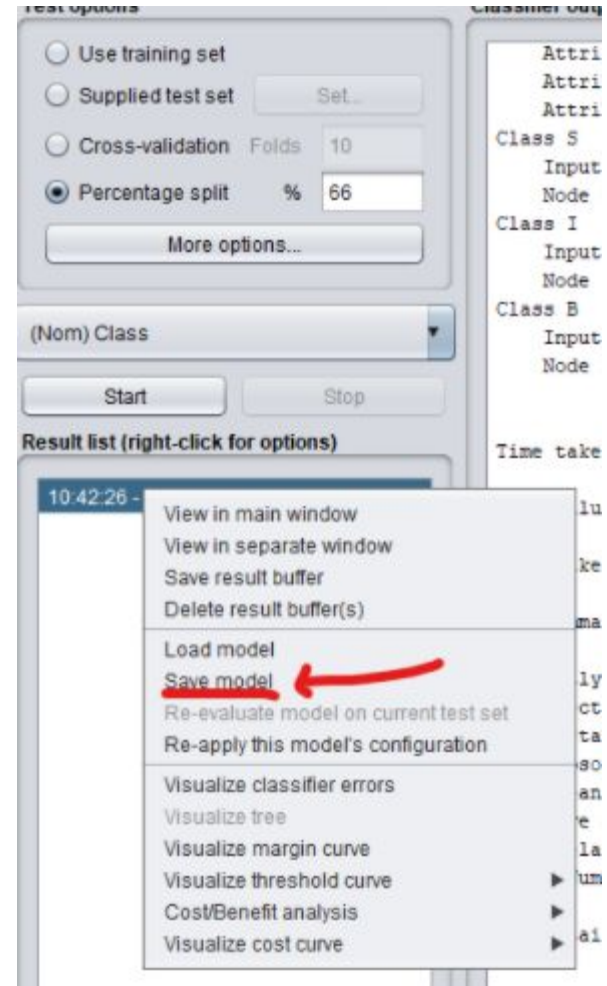
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1,000	0,069	0,910	1,000	0,953	0,920	1,000	1,000	S
	0,695	0,062	0,702	0,695	0,699	0,635	0,953	0,671	I
	0,876	0,021	0,967	0,876	0,919	0,870	0,994	0,991	B
Weighted Avg.	0,895	0,048	0,897	0,895	0,894	0,849	0,989	0,939	

=== Confusion Matrix ===

a	b	c	<-- classified as
12900	2	0	a = S
1279	3804	389	b = I
0	1612	11380	c = B

Obtener modelo

- Modelo que luego se cargará junto con los datos para realizar las predicciones.



Validación de modelo de red neuronal del DataSet_1.

Validación del modelo

- Primero se debe generar el modelo de la red neuronal (archivo .model).
- Es necesario obtener los datos que se utilizarán para realizar las pruebas (archivo arff).
- Los datos de prueba deben pasar por el mismo proceso de los datos que se utilizaron para crear el modelo.



- ```
@relation DataSet 1 Test
```

```
@attribute Col1 numeric
@attribute Col2 numeric
@attribute Col3 numeric
@attribute Col4 numeric
@attribute Col5 numeric
@attribute Col6 numeric
@attribute Col7 numeric
@attribute Col8 numeric
@attribute Col9 numeric
@attribute Col10 numeric
@attribute Col11 numeric
@attribute Col12 numeric
@attribute Col13 numeric
@attribute Col14 numeric
@attribute Col15 numeric
@attribute Class { S, I, B }
```

[illegible]



# Predicción del precio

- La columna inst, es el número de instancia del dataset.
- predicted: es lo que predijo, puede ser B, I o S.
- prediction: es la probabilidad de que haya sucedido (de 0 a 1).

| inst | predicted | prediction |
|------|-----------|------------|
| 1    | 3:B       | 1          |
| 2    | 3:B       | 1          |
| 3    | 3:B       | 1          |
| 4    | 3:B       | 1          |
| 5    | 3:B       | 1          |
| 6    | 3:B       | 1          |
| 7    | 3:B       | 1          |
| 8    | 3:B       | 1          |
| 9    | 3:B       | 1          |
| 10   | 3:B       | 0.999      |
| 11   | 1:S       | 0.999      |
| 12   | 1:S       | 1          |
| 13   | 1:S       | 0.999      |
| 14   | 1:S       | 1          |
| 15   | 1:S       | 1          |
| 16   | 1:S       | 1          |
| 17   | 1:S       | 1          |
| 18   | 1:S       | 1          |
| 19   | 1:S       | 1          |

---

**Generar modelo para el  
DataSet\_2.**





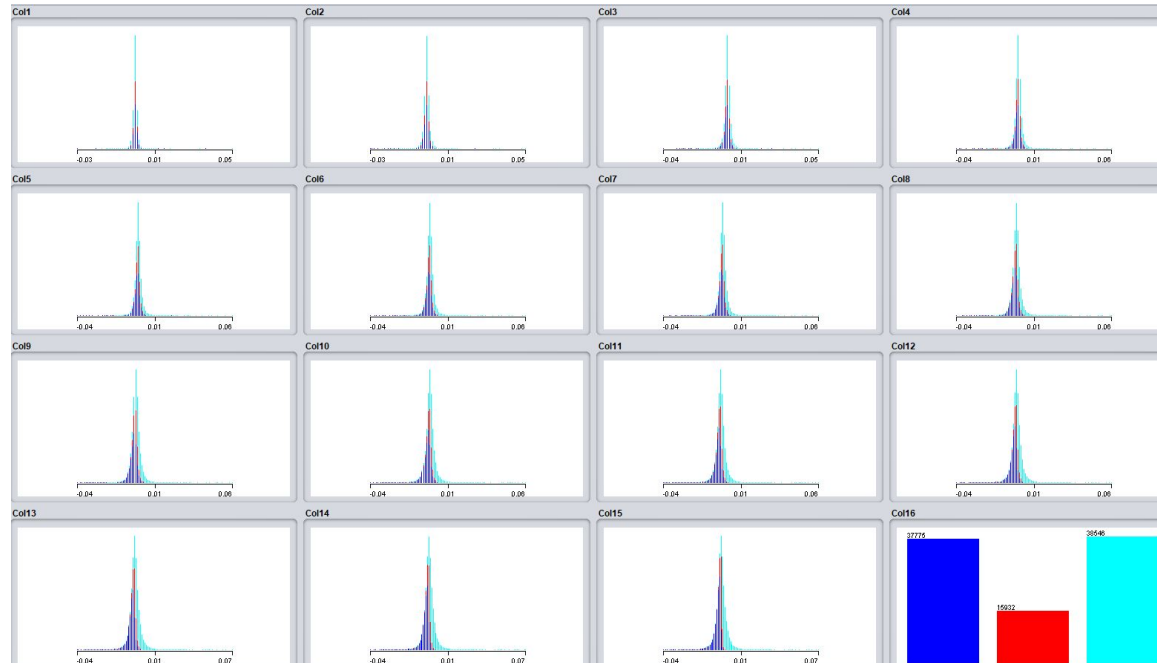
## Algoritmo 2: Restando el primer elemento

Consiste en:

- Agrupar vectores de a 16 elementos. ej. (10, 5, 45, 8, 6, 7, 12, 14, 32, 15, 11, 18, 10, 14, 15, 10) comenzando desde el primero, luego el segundo..
- A todos restarle el primero comenzando desde el segundo.
- resultado ejemplo: (5, -35, 2, 4, 3, -2, -4, -22, -5, -1, -8, 0, -4, -5, 0, B)
- Este dataset se le asigna el nombre de *DataSet\_2.csv*

DataSet 2.csv

# Vista gráfica de DataSet\_2 en weka



# Red Neuronal

- Las características de los datos no permitió que la red neuronal pueda aprender (kappa statistic siempre queda en 1).
- Datos demasiados simples (perfectos dentro de un rango), la red neuronal no puede desarrollar un modelo confiable que pueda predecir.
- Aprende de memoria.

weka.classifiers.functions.MultilayerPerceptron

About

A classifier that uses backpropagation to learn a multi-layer perceptron to classify instances. [More](#) [Capabilities](#)

GUI: False

autoBuild: True

batchSize: 100

debug: False

decay: False

doNotCheckCapabilities: False

hiddenLayers: a

learningRate: 0.3

momentum: 0.2

nominalToBinaryFilter: True

normalizeAttributes: True

normalizeNumericClass: True

numDecimalPlaces: 2

reset: True

resume: False

seed: 0

trainingTime: 500

validationSetSize: 0

validationThreshold: 20

## Resultado

=== Stratified cross-validation ===

=== Summary ===

|                                  |        |     |   |
|----------------------------------|--------|-----|---|
| Correctly Classified Instances   | 92253  | 100 | % |
| Incorrectly Classified Instances | 0      | 0   | % |
| Kappa statistic                  | 1      |     |   |
| Mean absolute error              | 0.0007 |     |   |
| Root mean squared error          | 0.0014 |     |   |
| Relative absolute error          | 0.1702 | %   |   |
| Root relative squared error      | 0.2997 | %   |   |
| Total Number of Instances        | 92253  |     |   |

=== Detailed Accuracy By Class ===

|               | TP Rate | FP Rate | Precision | Recall | F-Measure | MCC   | ROC Area | PRC Area | Class |
|---------------|---------|---------|-----------|--------|-----------|-------|----------|----------|-------|
| 1,000         | 0,000   | 1,000   | 1,000     | 1,000  | 1,000     | 1,000 | 1,000    | 1,000    | B     |
| 1,000         | 0,000   | 1,000   | 1,000     | 1,000  | 1,000     | 1,000 | 1,000    | 1,000    | I     |
| 1,000         | 0,000   | 1,000   | 1,000     | 1,000  | 1,000     | 1,000 | 1,000    | 1,000    | S     |
| Weighted Avg. | 1,000   | 0,000   | 1,000     | 1,000  | 1,000     | 1,000 | 1,000    | 1,000    |       |

=== Confusion Matrix ===

|       | a     | b     | c | <-- classified as |
|-------|-------|-------|---|-------------------|
| 37775 | 0     | 0     | 1 | a = B             |
| 0     | 15932 | 0     | 1 | b = I             |
| 0     | 0     | 38546 | 1 | c = S             |



# Proceso de normalización

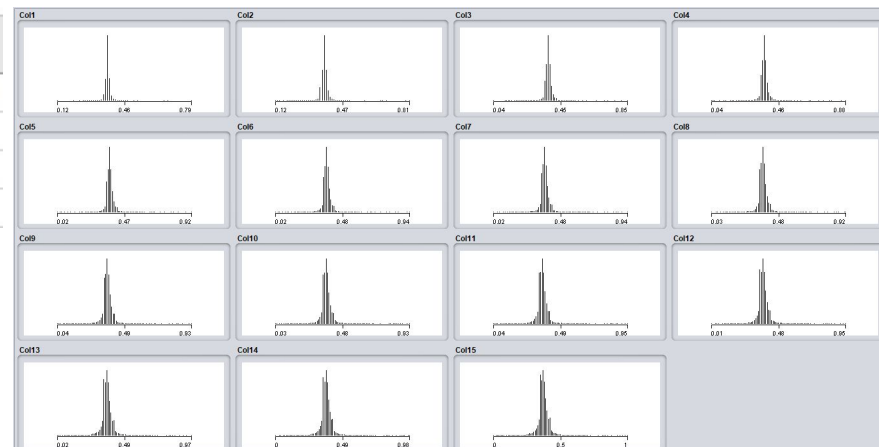
$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Consiste en:

- Proceso de normalización a DataSet\_2.
- Para que funcionen mejor muchos algoritmos de Machine Learning usados en Data Science, hay que normalizar las variables de entrada al algoritmo.
- Comprimir o extender los valores de la variable para que estén en un rango definido (0-1).
- X es el dato que ingresa a la función comenzando desde el primer elemento del dataset hasta el último, Xmin es el menor de todos los números del dataset y Xmax es el mayor elemento de todo el dataset.

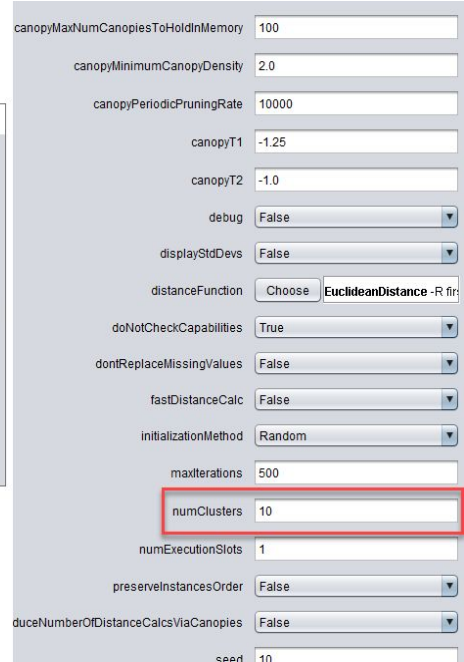
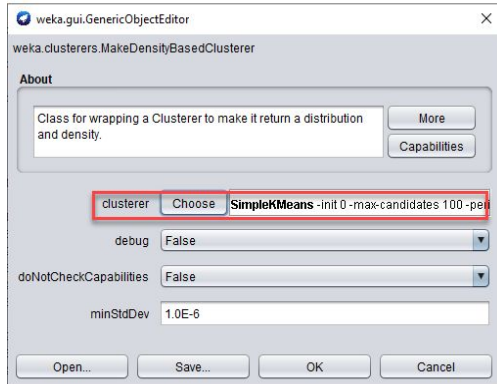
# DataSet\_3

|   | A                                                                                      | B | C | D | E | F | G |  |  |  |  |  |  |  |  |
|---|----------------------------------------------------------------------------------------|---|---|---|---|---|---|--|--|--|--|--|--|--|--|
| 1 | Col1,Col2,Col3,Col4,Col5,Col6,Col7,Col8,Col9,Col10,Col11,Col12,Col13,Col14,Col15,Class |   |   |   |   |   |   |  |  |  |  |  |  |  |  |
| 2 | 0.37,0.38,0.38,0.37,0.37,0.36,0.35,0.35,0.34,0.35,0.35,0.35,0.35,0.35,0.35,B           |   |   |   |   |   |   |  |  |  |  |  |  |  |  |
| 3 | 0.38,0.38,0.37,0.37,0.36,0.35,0.35,0.34,0.35,0.35,0.35,0.35,0.35,0.35,0.36,B           |   |   |   |   |   |   |  |  |  |  |  |  |  |  |
| 4 | 0.37,0.36,0.36,0.35,0.35,0.34,0.33,0.34,0.34,0.34,0.35,0.35,0.35,0.35,0.36,B           |   |   |   |   |   |   |  |  |  |  |  |  |  |  |



# Con el agrupador: Make Density Based Clusterer

- Para DataSet\_3
- El mejor resultado que se obtuvo fue con 10 clusters.





## Resultado: Make Density Based Clusterer

- EL “Log likelihood” es la medida para saber si el modelo es bueno.
- 44.9 es lo mejor que se logró.

### Clustered Instances

|   |             |
|---|-------------|
| 0 | 8197 ( 26%) |
| 1 | 3301 ( 11%) |
| 2 | 2213 ( 7%)  |
| 3 | 5318 ( 17%) |
| 4 | 4402 ( 14%) |
| 5 | 5300 ( 17%) |
| 6 | 42 ( 0%)    |
| 7 | 294 ( 1%)   |
| 8 | 1810 ( 6%)  |
| 9 | 490 ( 2%)   |

Log likelihood: 44.89992



# Resultados de predicción:

- Para cada cluster muestra las cantidades que predijo.
- Cada cluster se especificó en predecir B, I o S.

Number of iterations: 88  
Within cluster sum of squared errors: 411.0547036141147

Initial starting points (random):

Cluster 0: 0.38,0.38,0.38,0.39,0.38,0.38,0.39,0.39,0.38,0.38,0.38,0.38,0.38,0.37,B  
Cluster 1: 0.37,0.37,0.37,0.38,0.38,0.36,0.36,0.35,0.35,0.35,0.35,0.35,0.35,0.35,B  
Cluster 2: 0.38,0.37,0.36,0.36,0.36,0.36,0.36,0.36,0.36,0.35,0.35,0.35,0.36,0.35,B  
Cluster 3: 0.37,0.36,0.36,0.36,0.36,0.37,0.37,0.37,0.38,0.38,0.38,0.37,0.38,0.38,S  
Cluster 4: 0.38,0.38,0.36,0.37,0.36,0.37,0.37,0.37,0.37,0.37,0.38,0.37,0.37,0.35,B  
Cluster 5: 0.38,0.37,0.37,0.37,0.36,0.35,0.35,0.35,0.37,0.37,0.36,0.36,0.35,0.34,0.34,B  
Cluster 6: 0.37,0.38,0.38,0.38,0.38,0.38,0.38,0.38,0.37,0.37,0.37,0.37,0.37,0.37,I  
Cluster 7: 0.36,0.37,0.36,0.35,0.37,0.35,0.35,0.35,0.35,0.35,0.35,0.35,0.35,0.36,I  
Cluster 8: 0.39,0.38,0.39,0.39,0.39,0.38,0.39,0.4,0.4,0.4,0.39,0.39,0.37,0.36,0.36,B  
Cluster 9: 0.37,0.38,0.38,0.38,0.38,0.38,0.38,0.38,0.38,0.38,0.39,0.39,0.4,0.4,S

Missing values globally replaced with mean/mode

Final cluster centroids:

| Attribute | Cluster#               |                |              |               |                |                |               |               |               |               |               |
|-----------|------------------------|----------------|--------------|---------------|----------------|----------------|---------------|---------------|---------------|---------------|---------------|
|           | Full Data<br>(92253.0) | 0<br>(13667.0) | 1<br>(211.0) | 2<br>(4723.0) | 3<br>(28885.0) | 4<br>(14310.0) | 5<br>(1510.0) | 6<br>(9927.0) | 7<br>(6402.0) | 8<br>(3051.0) | 9<br>(9567.0) |
| Col1      | 0.3702                 | 0.3732         | 0.3539       | 0.3651        | 0.3678         | 0.3685         | 0.3623        | 0.373         | 0.3644        | 0.3837        | 0.3762        |
| Col2      | 0.3704                 | 0.3731         | 0.3341       | 0.3605        | 0.369          | 0.3652         | 0.3545        | 0.3742        | 0.3629        | 0.3874        | 0.3829        |
| Col3      | 0.3706                 | 0.3729         | 0.3132       | 0.3565        | 0.37           | 0.3623         | 0.3464        | 0.3752        | 0.3615        | 0.39          | 0.389         |
| Col4      | 0.3708                 | 0.3726         | 0.2939       | 0.3526        | 0.371          | 0.3598         | 0.3382        | 0.3761        | 0.3604        | 0.3919        | 0.3947        |
| Col5      | 0.371                  | 0.372          | 0.2754       | 0.3493        | 0.372          | 0.3577         | 0.3296        | 0.3768        | 0.3596        | 0.393         | 0.4001        |
| Col6      | 0.3711                 | 0.3712         | 0.2588       | 0.3458        | 0.3731         | 0.356          | 0.3215        | 0.3772        | 0.359         | 0.3934        | 0.405         |
| Col7      | 0.3712                 | 0.3702         | 0.2457       | 0.3425        | 0.3742         | 0.3547         | 0.3132        | 0.3775        | 0.3585        | 0.3931        | 0.4096        |
| Col8      | 0.3713                 | 0.369          | 0.2323       | 0.3393        | 0.3756         | 0.3535         | 0.3057        | 0.3776        | 0.3583        | 0.3918        | 0.4135        |
| Col9      | 0.3714                 | 0.3676         | 0.2209       | 0.3361        | 0.377          | 0.3527         | 0.299         | 0.3775        | 0.3583        | 0.3895        | 0.4172        |
| Col10     | 0.3715                 | 0.366          | 0.2099       | 0.3326        | 0.3786         | 0.352          | 0.2942        | 0.3772        | 0.3587        | 0.387         | 0.4203        |
| Col11     | 0.3716                 | 0.3643         | 0.2013       | 0.3288        | 0.3804         | 0.3514         | 0.2905        | 0.3766        | 0.3593        | 0.3834        | 0.4228        |
| Col12     | 0.3717                 | 0.3624         | 0.1964       | 0.3256        | 0.3823         | 0.3508         | 0.2875        | 0.3759        | 0.3603        | 0.3795        | 0.4248        |
| Col13     | 0.3718                 | 0.3605         | 0.1943       | 0.3228        | 0.3844         | 0.3501         | 0.2851        | 0.3751        | 0.3615        | 0.3747        | 0.4264        |
| Col14     | 0.3719                 | 0.3584         | 0.1943       | 0.3206        | 0.3867         | 0.3493         | 0.2836        | 0.3741        | 0.3628        | 0.3694        | 0.4276        |
| Col15     | 0.3719                 | 0.3558         | 0.1943       | 0.3191        | 0.3894         | 0.3482         | 0.2826        | 0.373         | 0.3644        | 0.3632        | 0.4285        |
| Class     | S                      | B              | B            | B             | S              | B              | B             | I             | I             | B             | S             |



## Resultados de predicción:

- El cluster cero predijo 13668 bajas de 13670.

Fitted estimators (with ML estimates of variance):

Cluster: 0 Prior probability: 0.1481

Attribute: Col1

Normal Distribution. Mean = 0.3732 StdDev = 0.0071

Attribute: Col2

Normal Distribution. Mean = 0.3731 StdDev = 0.009

Attribute: Col3

Normal Distribution. Mean = 0.3729 StdDev = 0.0095

Attribute: Col4

Normal Distribution. Mean = 0.3726 StdDev = 0.0098

Attribute: Col5

Normal Distribution. Mean = 0.372 StdDev = 0.0097

Attribute: Col6

Normal Distribution. Mean = 0.3712 StdDev = 0.0096

Attribute: Col7

Normal Distribution. Mean = 0.3702 StdDev = 0.0096

Attribute: Col8

Normal Distribution. Mean = 0.369 StdDev = 0.0096

Attribute: Col9

Normal Distribution. Mean = 0.3676 StdDev = 0.0096

Attribute: Col10

Normal Distribution. Mean = 0.366 StdDev = 0.01

Attribute: Col11

Normal Distribution. Mean = 0.3643 StdDev = 0.0102

Attribute: Col12

Normal Distribution. Mean = 0.3624 StdDev = 0.0106

Attribute: Col13

Normal Distribution. Mean = 0.3605 StdDev = 0.0109

Attribute: Col14

Normal Distribution. Mean = 0.3584 StdDev = 0.0111

Attribute: Col15

Normal Distribution. Mean = 0.3558 StdDev = 0.0108

Attribute: Class

Discrete Estimator. Counts = 13668 1 1 (Total = 13670)



## Porcentaje de predicción por cluster:

- Cada cluster predijo con casi un 100% de probabilidad.

**(B) Cluster 0:** Total = 13670; Aciertos = 13668; No aciertos = 2; Porcentaje= **99.98%**

**(B) Cluster 1:** Total = 214; Aciertos = 212; No aciertos = 2; Porcentaje= **98.24%**

**(B) Cluster 2:** Total = 4726; Aciertos = 4724; No aciertos = 2; Porcentaje= **99.95%**

**(S) Cluster 3:** Total = 28888; Aciertos = 28886; No aciertos = 2; Porcentaje= **99.99%**

**(B) Cluster 4:** Total = 14313; Aciertos = 14311; No aciertos = 2; Porcentaje= **99.98%**

**(B) Cluster 5:** Total = 1513; Aciertos = 1511; No aciertos = 2; Porcentaje= **99.86%**

**(I) Cluster 6:** Total = 9930; Aciertos = 9928; No aciertos = 2; Porcentaje= **99.97%**

**(I) Cluster 7:** Total = 6405; Aciertos = 6403; No aciertos = 2; Porcentaje= **99.96%**

**(B) Cluster 8:** Total = 3054; Aciertos = 3052; No aciertos = 2; Porcentaje= **99.93%**

**(S) Cluster 9:** Total = 9570; Aciertos = 9568; No aciertos = 2; Porcentaje= **99.97%**



## Resultados del Make Density Based Clusterer:

Resultados

$(99.98 + 98.24 + 99.95 + 99.99 + 99.98 + 99.86 + 99.97 + 99.96 + 99.93 + 99.97)/10 =$   
**99.78 (Promedio general de aciertos)**

**Predicción de bajas:**

$(99.98(\text{Cluster0}) + 98.24(\text{Cluster1}) + 99.95(\text{Cluster2}) + 99.98(\text{Cluster4}) + 99.86(\text{Cluster5}) + 99.93(\text{Cluster8})) / 6 (\text{Cantidad clusters}) =$  **99.65 (Promedio de aciertos de bajas)**

**Predicción de iguales:**

$(99.97(\text{Cluster6}) + 99.96(\text{Cluster7})) / 2 (\text{Cantidad de clusters}) =$  **99.965 (Promedio de aciertos de iguales)**

**Predicción de subas:**

$(99.99(\text{Cluster3}) + 99.97(\text{Cluster9})) / 2 =$  **99.98 (Promedio de aciertos de subas)**



## Comparando resultados de predicción:

```
-----Resultados de modelo MultilayerPerceptron-----
Cantidad de 1: 18582
Cantidad: 29190
Promedio: 99.87
-----Prediccion de Bajas-----
Cantidad de B: 12164
Promedio: 99.94
-----Prediccion de Iguales-----
Cantidad de I: 4847
Promedio: 99.614
-----Prediccion de Subas-----
Cantidad de S: 12179
Promedio: 99.895

Process finished with exit code 0
```

Resultados de modelo MakeDensityBasedClusterer  
Promedio general: **99.78 (Promedio general de aciertos)**  
Predicción de bajas: **99.65 (Promedio de aciertos de bajas)**  
Predicción de iguales: **99.965 (Promedio de aciertos de iguales)**  
Predicción de subas: **99.98 (Promedio de aciertos de subas)**

## Trabajos futuros:

- Desarrollo de API.
- Front-End





## Demo final:

- Datos de los últimos de enero del 2022.
- El círculo indica los del 31/01.





**Fin**

**¡Gracias!**

Matías Acevedo

Jueves 17 de Febrero 2022

