

UNIVERSIDAD DEL CEMA



Maestría en Finanzas Mercado de Capitales

Autores:

- Battafarano, Maximiliano.

Tutores:

- Orazi, Pablo.
- Siri, Julián Ricardo.

Índice

Introducción	2
Marco teórico	2
Planteo propuesto	2
<i>Implementación práctica</i>	2
Especificaciones técnicas de la solución desarrollada:	3
Estructura de la base de datos utilizada para poder almacenar todos los datos fundamentales y ratios de los activos históricos del S&P500 desde su creación:	4
Input de la solución desarrollada:	6
Output de la solución desarrollada:	7
Análisis de la performance de la solución propuesta	8
Posibles mejoras por aplicar en la solución propuesta	12
<i>Aplicación de redes neuronales y algoritmos genéticos para optimizar el proceso de estimación de los retornos ante un contexto específico</i>	12
Redes neuronales con aprendizaje no supervisado	13
Redes neuronales con aprendizaje supervisado	14
<i>Riesgos de aplicar inteligencia artificial y algoritmos genéticos durante el proceso de estimación del retorno de nuestro portafolio</i>	15
<i>Aplicación del Internet Of Things - IOT para optimizar el proceso de captar señales y poder gestionar nuestro portafolio de manera coherente con nuestra estrategia</i>	15
Conclusión	16
Bibliografía	17
Autorización para publicar los trabajos finales	18

Introducción

Existen varias teorías relacionadas al hecho de poder entender y explicar los distintos factores que influyen en la valuación y las fluctuaciones de los precios de los activos financieros y, en base a ellas poder estimar las características asociados a los mismos para definir una estrategia óptima para poder administrar un portafolio de inversiones, existe un gran problema para poder determinar un nivel de riesgo estándar para un estado de mercado o activo determinado. Esto se debe a que los rendimientos y riesgos esperados no son observables, sino que se deben estimar, mientras los únicos datos observables para nosotros como profesionales del mundo de las finanzas y administración de portafolios son los rendimientos ya realizados. A su vez, más allá de que esta situación nos genera un ruido o error en nuestras estimaciones, las mismas se vuelven más complejas a medida que se incrementa el número de activos en el portafolio de inversión, volviendo sumamente complejo el proceso de re-estimación posterior rebalanceo para ajustar la estrategia de inversión al contexto actual del mercado.

Ante esta problemática, el presente trabajo presentará una propuesta que, mediante la implementación de una solución informática, le permitirá a un portfolio manager obtener estimaciones óptimas minimizando el error cuadrático de las mismas y, sobre todo, le permitirá tomar decisiones de manera rápida durante los pasos del proceso de administración de una cartera de inversión de implementar la estrategia y llevar a cabo el proceso de revisión y mejora.

Finalmente, procederemos a llevar a cabo un análisis de *back-test* para analizar qué tan efectiva fue el desarrollo implementado.

Marco teórico

Planteo propuesto

Dado que el hecho de aplicar la *Teoría Moderna de Portafolios (MTP)* actualmente implica un alto grado de complejidad para poder estimar de manera efectiva y eficiente la varianza y el rendimiento esperado de un portafolio a medida que el número de activos de este se va incrementando, desarrollamos una herramienta que le permitirá a un portfolio manager llevar a cabo el proceso de administración de carteras de una manera muy precisa y eficiente. Para esto, se desarrolló una solución informática que automatiza de manera programática los distintos modelos y métodos necesarios para poder aplicar la *MTP* en la práctica y llevar a cabo las estimaciones de nuestro portafolio óptimo.

En esta sección, procederemos a explicar cómo funciona y cómo aplicamos los distintos modelos y enfoques que fundamentan el resultado de la herramienta desarrollada. A su vez, en la sección del anexo encontrarán el output detallado de nuestra herramienta explicando en profundidad qué simboliza cada uno de ellos y cómo un portfolio manager puede sacar provecho de este.

Implementación práctica

El algoritmo desarrollado nos permitirá generar un portafolio óptimo aplicando un proceso de filtrado en tiempo real a partir de los ratios y fundamentals de los componentes históricos del S&P500. Una de las particularidades de nuestro algoritmo es que es totalmente modificable y configurable, lo que nos permite cambiar la estrategia planteada según nuestro punto de vista sobre el mercado pudiendo cambiar tanto los fundamentals utilizados como la relación de estos para con la media geométrica histórica del mercado. Para el caso del portafolio óptimo buscado en el caso práctico filtraremos solicitando que los ratios "EV/EBITDA", "P/E ratio", "P/B ratio" y "Net Debt/EBITDA" estén por debajo de la media geométrica histórica del mercado y que el "Interest Coverage" esté por encima de la misma.

Una vez que se realizó el proceso de selección de activos, se procederá a analizar los *risk-drivers* de los precios históricos de nuestros que forman parte del portafolio, dando inicio así al proceso de confección de este, el cual se realizará aplicando la *MTP*. Algo fundamental es que previo al análisis de los *risk-drivers* de los activos y la confección de las estimaciones del modelo es primordial estandarizar y normalizar las observaciones para que las estimaciones sean consistentes. Esto se aplicará tanto para las observaciones que formarán parte de las series de tiempo que utilizaremos durante el análisis de los *risk-drivers*¹, como a la cantidad y el índice de estas, logrando así asegurar que los datos son consistentes y estén efectivamente relacionados.

Algo muy importante a aclarar, es que la solución desarrollada le permite al portfolio manager utilizar distintos time frames para el análisis de los ratios y, otro completamente distinto para las proyecciones o estimaciones que el modelo llevará a cabo, permitiéndonos tener más observaciones para llevar a cabo la estimación, resultando así en una matriz de varianza y covarianza más precisa y que explica mejor el comportamiento y la correlación de los activos entre sí.

Resulta primordial, para el cálculo en cada una de las ventanas de tiempo, que todos los datos con los cuales se llevarán a cabo las estimaciones estén normalizados y estandarizados. Por ende, si bien la lógica de la solución se encarga de realizar esto en tiempo real de ejecución en base a las observaciones que la misma posee, es clave poder tener la mayor cantidad de datos e información disponible para cada uno de los posibles *time-frames* para, de esta manera, lograra las estimaciones más precisas.

Especificaciones técnicas de la solución desarrollada:

Para desarrollar la solución informática y llevar a la práctica la *MTP* y todo el marco teórico explicado en el trabajo, decidí utilizar el lenguaje de programación Python en su versión 3.8. Esto se debió a que es un lenguaje de programación gratis y de código abierto de uso general que, dada su gran comunidad tiene desarrolladas muchas funcionalidades para poder aplicar modelos y estimaciones estadísticas, al igual que módulos para poder aplicar algoritmos de redes neuronales y machine learning. Algunas de la librerías o módulos más conocido y, utilizamos en la solución planteada son:

- numpy
- scipy.stats
- statsmodels
- matplotlib
- seaborn
- pandas
- pyMC

Para poder almacenar los datos de los fundamentals y ratios de las acciones e índices que nuestra solución utiliza, decidimos utilizar un motor de base de datos NoSQL. Esto se debe al gran volumen de datos que debemos almacenar y utilizar para poder llevar a cabo nuestras estimaciones y, sobre todo, nos brinda una gran capacidad de escalabilidad a futuro en ese sentido. Esto se debe a que el hecho de utilizar un motor de base de datos NoSQL nos permite lograr una arquitectura mucho más escalable (mencionar y explicar escalabilidad horizontal), descentralizada (distribuida) y modificable (los modelos relacionales dado a las restricciones que el mismo plantea hace mucho más compleja la modificabilidad de una arquitectura o estructura del esquema de BD y las tablas que este compone).

¹ Cerciorándonos que en las comparaciones para confeccionar matrices de covarianzas o de performance contra el benchmarking definido, las observaciones estarán asociadas a un mismo time-frame.

A su vez la atomicidad de las transacciones² en el ambiente que vamos a aplicarla no es un aspecto crucial, por lo que no hay un motivo específico que nos lleve a querer considerar un motor de base de datos SQL para la solución.

Dentro de los distintos motores de base de datos NoSQL del mercado decidimos utilizar MongoDB. Este motor de base de datos almacena los mismos en “documentos”, los cuales simplemente son registros en formato JSON³, lo que permite una fácil interpretación y entendimiento de los datos para personas que no sean profesionales o trabajen en el rubro informático, lo cual era una de las premisas que tratamos de priorizar en la solución planteada. A su vez, al ser en formato JSON, los cual en el caso de los motores de bases de datos NoSQL suele ser en archivos de texto, las consultas sobre los documentos en los que persistiremos nuestros datos serán mucho más fáciles que cualquier otro motor de base de datos SQL en cuestión.

En el hipotético caso de que queramos migrar hacia una plataforma de cloud-computing podremos migrar fácilmente a MongoDB Atlas teniendo la posibilidad de tener toda una arquitectura de base de datos distribuida en la nube, logrando así una mayor disponibilidad y resiliencia para nuestra aplicación.

Ahora bien, más allá del lenguaje que utilizaremos para desarrollar la aplicación y con qué herramienta almacenar la misma, es importante definir e identificar la fuente de dicha información. Como premisa principal, las mismas deberán ser certeras, completas y verídicas. Es así como recurrimos a la información disponible en Stockrow y FRED utilizando las APIs (Application Programming Interface) que ambos brindan. De esta manera, la herramienta es capaz de obtener toda la información en forma automática y almacenarla para poder optimizar todos los cálculos de procesamiento requeridos y, sobre todo, permitirnos poder utilizar dicha información para poder desarrollar o nuevos benchmarks históricos y buscar nuevas relaciones entre los datos dado un contexto mediante algoritmos de data mining⁴.

Estructura de la base de datos utilizada para poder almacenar todos los datos fundamentales y ratios de los activos históricos del S&P500 desde su creación:

Más allá del hecho de por qué se utilizó un motor de base de datos NOSQL dado el volumen de datos que estaremos almacenando y consultando, un aspecto muy importante para tener en cuenta es el hecho de cómo almacenar la información pertinente de la manera óptima posible. A diferencia de los motores SQL en los que en estos casos se suelen utilizar tablas planas para evitar la mayor cantidad de joins⁵ posibles, aquí lo primordial dada la falta de restricciones relacionales entre cada uno de los documentos es poder contar con todos los datos posibles para poder optimizar al máximo el proceso de filtrado al momento de llevar a cabo las distintas consultas a nuestro motor de base de datos.

2 Como atomicidad de una transacción se entiende que las modificaciones en la base de datos deben seguir una regla de ejecutar todo o nada, cerciorando así que al finalizar la transacción la base de datos siempre contenga datos válidos y consistentes entre sí.

3 Es un formato que permite representar objetos de una manera fácil de interpretar, lo cual hace que sea muy aceptado y soportado por varios lenguajes de programación como Python, C/C++, entre otros.

4 Las URL de las fuentes de datos que se utilizaron en la solución desarrollada son:

- <https://stockrow.com/>
- <https://www.sec.gov>
- <https://finance.yahoo.com/>

Dicha información se encuentra adjuntada y entregada en el pendrive junto con el documento de la tesis.

5 Un Join en base de datos consiste en el proceso de tomar datos de varias tablas y colocarlos en una vista generada. Existen 4 tipos posible de joins:



Un aspecto clave a tener en cuenta en todo desarrollo informático, es definir los documentos u objetos que queremos reflejar en nuestro motor de base datos y, a su vez, identificar las posibles propiedades o clasificadores que nos permitan diferenciar distintos “tipos” dentro de los mismos. Un ejemplo de esto es, en nuestro caso práctico, el *time-frame* del ratio en cuestión, dado que el ratio Price to Book puede ser anual, semestral o trimestral, pero en sí serán distintas clasificaciones dentro del mismo documento (*Ratio*) y dentro de la misma instancia de este (*Price to Book Value*). En base a esto se definen cuatro tipos de documentos principales que conformarán nuestro modelo de base de datos. Estos son:

stock_cash_flow_hist_data:

```
"_id": {
  "$oid": "6060c369cd10bb1cf6d38747"
},
"ticker": "AAPL",
"stock_name": "Apple Inc",
"time_frame": "Q",
"cash_flow_id": "Other financial activities",
"date": "2020-12-31",
"value": "-2883000000.0"
```

Figura 1: Estructura de datos de documento de MongoDB “stock_cash_flow_hist_data”.

Fuente: Elaboración propia.

stock_growth_hist_data:

```
"_id": {
  "$oid": "6060c364cd10bb1cf6d38171"
},
"ticker": "AAPL",
"stock_name": "Apple Inc",
"time_frame": "Q",
"growth_id": "EPS Growth (basic)",
"date": "2020-12-31",
"value": "0.3492"
```

Figura 2: Estructura de datos de documento de MongoDB “stock_growth_hist_data”.

Fuente: Elaboración propia.

stock_incomes_hist_data:

```
{
  "_id": {
    "$oid": "6060c353cd10bb1cf6d36781"
  },
  "ticker": "AAPL",
  "stock_name": "Apple Inc",
  "time_frame": "Q",
  "income_id": "Shares (Basic, Weighted)",
  "date": "2020-12-31",
  "value": "16935120000.0"
}
```

Figura 3: Estructura de datos de documento de MongoDB “stock_incomes_hist_data”.

Fuente: Elaboración propia.

stock_ratios_hist_data:

```

    "_id": {
      "$oid": "6060c35acd10bb1cf6d37131"
    },
    "ticker": "AAPL",
    "stock_name": "Apple Inc",
    "time_frame": "Q",
    "ratio_id": "P/E ratio",
    "date": "2020-12-31",
    "value": "35.5023"
  }

```

Figura 4: Estructura de datos de documento de MongoDB "stock_ratios_hist_data".

Fuente: Elaboración propia.

Como se puede observar en las capturas de algunos ejemplos en las tuplas reales que almacenamos en nuestra base de datos, todos los documentos contienen el atributo "time_frame", al cual nos permitirá consultar una categoría específica dentro de un tipo de documento específico. De esta manera, podremos llevar a cabo estimaciones precisas y efectivas, pudiendo ajustar los time frames según el nivel de precisión que deseemos en nuestras estimaciones y, a su vez, ajustar el análisis de las observaciones que tenemos a un periodo histórico acotado según sea necesario. Nótese que este último aspecto es clave para poder llevar a cabo un análisis efectivo de data mining y obtener nueva información a partir de los datos que poseemos, permitiéndonos observar posibles comportamientos de los activos financieros de cierto sector, por ejemplo, dado un contexto específico. A su vez, en el caso de que creamos conveniente, tendremos la posibilidad de poder llevar a cabo reportes y tableros de control a modo de business intelligence (De esta manera podremos monitorear y comparar contra un benchmark customizado que deseemos crear como portfolio managers para poder llevar a cabo nuestra filosofía y estrategia de inversión).

Input de la solución desarrollada:

El input de la solución informática que desarrollamos son los archivos "SP_500_members.xlsx" y "SP_500_joiners_and_leavers.xlsx" que se encuentra dentro del directorio "resources" de la aplicación:

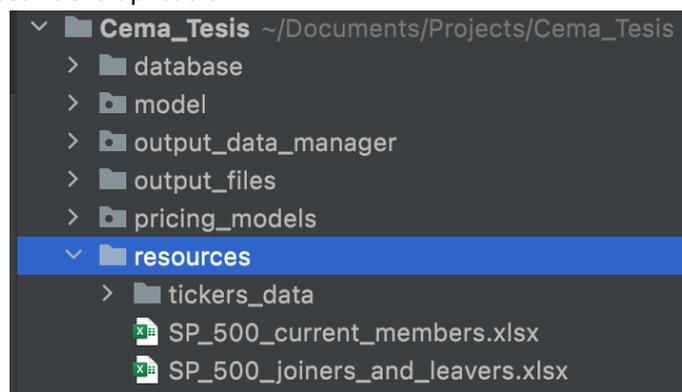


Figura 5: Ejemplificación de los archivos de input del algoritmo desarrollado.

Fuente: Elaboración propia.

En el caso de que se modifique la composición del S&P500 bastará con reemplazar ambos archivos con los datos actualizados que descargan directamente de Refinitiv Eikon.

En base a estos dos archivos nuestra aplicación seleccionará los mejores n activos que conformarán nuestro portafolio óptimo en base a los factores o ratios que forman parte de nuestro modelo multifactorial.

A su vez, el archivo de configuración de la solución desarrollada, “*config.json*”, permite que el usuario pueda configurar los distintos parámetros para ajustar el modelo según el nivel de precisión que desee, el benchmarking contra el que desee estimar el rendimiento de su portafolio y, las observaciones históricas de los retornos de los activos que formarán parte de los distintos portafolios óptimos según el modelo a aplicar:

```
{
  "factors" :
  [
    {
      "name": "EV/EBITDA",
      "enabled": true,
      "measure": "lower"
    },
    {
      "name": "P/E ratio",
      "enabled": true,
      "measure": "lower"
    },
    {
      "name": "P/B ratio",
      "enabled": true,
      "measure": "lower"
    },
    {
      "name": "Net Debt/EBITDA",
      "enabled": true,
      "measure": "lower"
    },
    {
      "name": "Interest Coverage",
      "enabled": true,
      "measure": "higher"
    }
  ],
  "time-frame" : "Q",
  "benchmark" : "SPX",
  "model-time-series-start-for-estimation" : "1990-01-01",
  "portfolio-notional" : 100000,
  "portfolio-target-return" : 0.5,
  "amount_of_stocks" : 15,
  "compute_optimal_portfolio_model" : true,
  "import_historic_ratios_database" : false,
  "download_historic_ratios_database" : false,
  "load_optimal_portfolio_stocks_from_last_run" : true,
  "hedge_portfolio_positions" : false
}
```

Figura 6: Archivo de configuración para la selección de activos y hedge del portafolio de la solución desarrollada.

Fuente: Elaboración propia.

Cabe aclarar, que cuantas más observaciones tengamos en cuenta en la estimación de nuestro modelo, más preciso será el mismo ya que podrá explicar mejor los rendimientos y las relaciones entre los distintos activos que conforman nuestro portafolio.

Output de la solución desarrollada:

La solución desarrollada nos brindará distintos outputs según lo que el usuario le solicite. Uno de los ítems fundamentales que forman parte de la salida del desarrollo realizado son los activos que formarán parte del portafolio óptimo según las preferencias del usuario. Esto se debe a que en el caso que sea necesario, puede ser utilizado como input para una nueva estimación del rendimiento de nuestro portafolio, ahorrándonos mucho tiempo de computo para poder llevar a cabo la selección de estos y, en caso de ya tener dicho portafolio, nos brindará las ponderaciones óptimas de dichos activos en el mismo de una manera eficaz y mucho menos costosa en términos de ciclos de cómputo. Esto se ve reflejado en el archivo “*Last_Run_Optimal_Stocks.json*” dentro del directorio “*output_files*”:



Figura 7: Output del algoritmo desarrollado, archivo de activos óptimos seleccionados en la última corrida del algoritmo.

Fuente: Elaboración propia.

A su vez, podremos observar toda la información y el reporte del análisis que llevó a cabo la solución desarrollada para, validar que la distribución de los retornos de los distintos activos que forman parte del portafolio óptimo es normal aplicando el test de Jarque Bera, al igual que la información de cada uno de los portafolios óptimos estimados según el modelo especificado (*PCA*, *Markowitz Mean Variance*, *Long Only* y *Equivalent Weights*). La información pertinente a cada una de las corridas para estimar el rendimiento de los portafolios se encuentra dentro del directorio “*output_files/optimization_model_results*”:

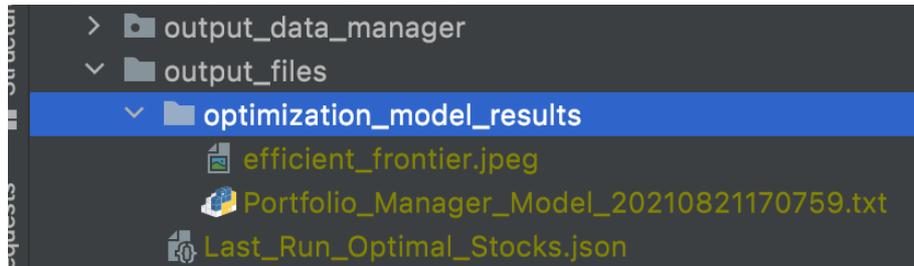


Figura 8: Output del algoritmo desarrollado, archivo de resultados del modelo de optimización y estimación del portafolio.

Fuente: Elaboración propia.

Dentro de este archivo, encontraremos información pertinente a cada uno de los portafolios que nuestra solución estimará, específicamente:

- Los activos y la ponderación de cada uno de ellos.
- El notional invertido en el portafolio.
- El P&L estimado del portafolio.
- El retorno estimado del portafolio.
- La varianza del portafolio.
- Ratio de Sharpe del portafolio.
- El objetivo de retorno del portafolio.

Análisis de la performance de la solución propuesta

Procederemos a ejecutar la solución desarrollada para, en base al input del usuario o portafolio manager, seleccionar los activos óptimos y confeccionar el portafolio óptimo junto con las estimaciones de riesgo y retorno del mismo para que luego el portafolio manager pueda ejecutar su estrategia de inversión. En base a esto y a modo comparativo, se llevarán a cabo dos estimaciones en momentos distintos, una en $T_1 = 3$ de Noviembre de 2021 y otra en $T_0 = T_1 - 60$ días para, de esta manera, evaluar la eficacia de las estimaciones y, evaluar posibles cambios o rebalanceos propuestos por la solución en base a la evolución de los *risk-drivers* de los activos que componen nuestro portafolio.

En relación con esto, el input o parámetros definidos por el portafolio manager simulado en la puesta en práctica de la solución desarrollada es:

```

{"factors" :
[
  {
    "name": "EV/EBITDA",
    "enabled": true,
    "measure": "lower"
  },
  {
    "name": "P/E ratio",
    "enabled": true,
    "measure": "lower"
  },
  {
    "name": "P/B ratio",
    "enabled": true,
    "measure": "lower"
  },
  {
    "name": "Net Debt/EBITDA",
    "enabled": true,
    "measure": "lower"
  },
  {
    "name": "Interest Coverage",
    "enabled": true,
    "measure": "higher"
  }
],
"time-frame" : "Q",
"benchmark" : "SPX",
"model-time-series-start-for-estimation" : "1990-01-01",
"portfolio-notional" : 100000,
"portfolio-target-return" : 0.5,
"amount_of_stocks" : 15,
"compute_optimal_portfolio_model" : true,
"import_historic_ratios_database" : false,
"download_historic_ratios_database" : false,
"load_optimal_portfolio_stocks_from_last_run" : true,
"hedge_portfolio_positions" : true,
"hedging_delta" : 100,
"hedging_tickers" : 100
}

```

Figura 9: Archivo de configuración para la selección de activos y hedge del portafolio de la solución desarrollada.

Fuente: Elaboración propia.

La solución definió los siguientes activos óptimos, para los cuales especifica su retorno y volatilidad esperada para cada uno de ellos de manera individual:

```

-----
Portfolio Manager details:
Tickers: ['FB', 'BRK.B', 'JPM', 'JNJ', 'UNH', 'PG', 'BAC', 'CMCSA', 'XOM', 'PFE', 'CSCO', 'VZ', 'INTC', 'TMO', 'LYV']
Returns (annualized):
[[7.07389337]
 [3.16597756]
 [4.70393527]
 [3.14899889]
 [5.32246771]
 [2.73882688]
 [5.11860572]
 [4.0777442 ]
 [0.62621115]
 [2.95029132]
 [4.13788285]
 [1.84241715]
 [2.86916689]
 [5.96837864]
 [6.00714783]]
Volatilities (annualized):
[[1.60321858]
 [0.69705075]
 [1.01191566]
 [0.68248824]
 [0.87779436]
 [0.65809297]
 [1.29445975]
 [0.89804301]
 [1.07648565]
 [0.86327717]
 [1.05324733]
 [0.73835137]
 [1.05773534]
 [0.87430815]
 [1.37189069]]

```

Figura 10: Activos óptimos que seleccionó la solución desarrollada en base a la configuración definida por el portfolio manager.

Fuente: Elaboración propia.

Una vez obtenidos los activos acordes a la estrategia definida por el portfolio manager, la solución procederá a calcular el portafolio óptimo para los mismos, en este caso, en T_0 :

```

Portfolio type: markowitz-avg
Tickers | Weights:
FB | 13330.650028960286
BRK.B | 19548.428197147427
JPM | 1680.2453986830164
JNJ | 10479.868534756
UNH | 4047.6948548245746
PG | 20505.070441571406
BAC | 8.200482931149096e-13
CMCSA | 4310.402460345671
XOM | 4.2481229356183014e-13
PFE | 2.1139854360786058e-12
CSCO | 6.671844060000056e-13
VZ | 9064.43153419994
INTC | 3997.6045699955444
TMO | 13035.60397951615
LYV | 1.3965127939704663e-12
Notional (MM USD): 100000
Delta (MM USD): 100000.00000000001
PnL annual (MM USD): 398346.30282154656
Return annual (MM USD): 3.9834630282154655
Target Return: 3.9834630282154655
Volatility annual (MM USD): 0.6228673922539485
Sharpe ratio annual: 6.395362925968314

```

Figura 11: Portafolio óptimo estimado por la solución desarrollada en T_0 .
Fuente: Elaboración propia.

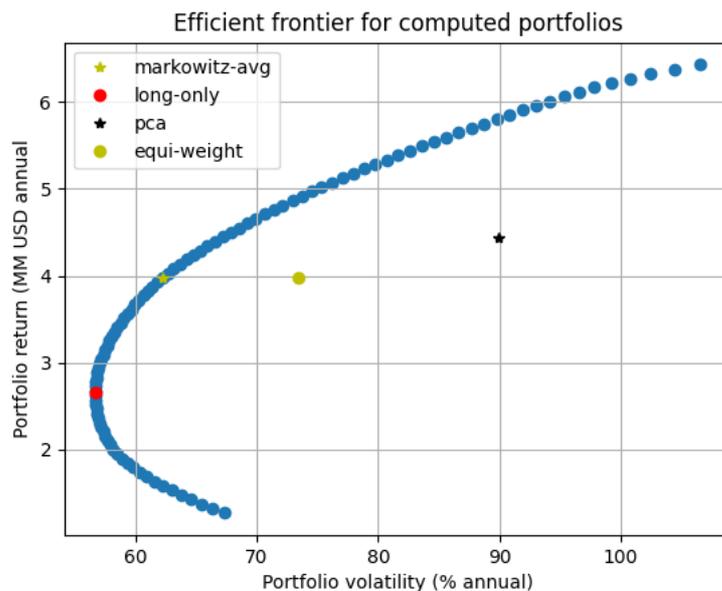


Figura 12: Comparación del portafolio óptimo estimado por la solución desarrollada contra otros tipos de portafolios en T_0 .
Fuente: Elaboración propia.

Ahora, como explicamos anteriormente, a modo de comparación, procederemos a volver a calcular estimación de nuestro portafolio óptimo con los mismos activos que en T_0 , pero en T_1 para poder evaluar y observar posibles cambios en las estimaciones y composiciones de este ante cambios en el mercado:

```

Portfolio type: markowitz-avg
Tickers | Weights:
FB | 12665.80783604495
BRK.B | 19258.026727739816
JPM | 2522.9403705454197
JNJ | 8799.868870374097
UNH | 4862.66265030772
PG | 21246.64303188832
BAC | 0.0
CMCSA | 3113.0290345669
XOM | 0.0
PFE | 3.3640841750290716e-13
CSCO | 0.0
VZ | 8936.005428171864
INTC | 3262.7233280477813
TMO | 15332.29272231314
LYV | 6.934533931534785e-13
Notional (MM USD): 100000
Delta (MM USD): 100000.00000000001
PnL annual (MM USD): 394436.9917170688
Return annual (MM USD): 3.9443699171706883
Target Return: 3.9443699172000004
Volatility annual (MM USD): 0.6278324584024486
Sharpe ratio annual: 6.282519905401732

```

Figura 13: Portafolio óptimo estimado por la solución desarrollada en T_1 .

Fuente: Elaboración propia.

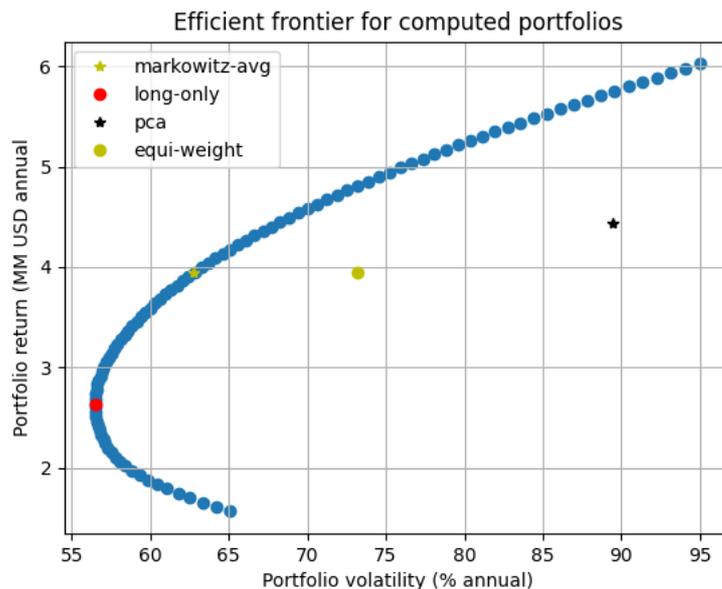


Figura 14: Comparación del portafolio óptimo estimado por la solución desarrollada contra otros tipos de portafolios en T_1 .

Fuente: Elaboración propia.

Lo primero que se observa es que en T_1 la solución nos propone un rebalanceo en nuestro portafolio dado cambios en el contexto, aconsejando que el peso de los activos de Bank of America Corp (BAC), Exxon Mobil Corporation (XOM) y Cisco Systems (CSCO) en nuestro portafolio sea de cero y, aumentar nuestra exposición en el resto de los activos salvo Facebook (FB), Berkshire Hathaway (BRK.K) y Johnson & Johnson (JNJ), activos para los cuales también nos recomienda disminuir el peso de los mismos en nuestro portafolio. A su vez, se observa que en comparación con T_0 , en T_1 el retorno estimado del mismo disminuye en USD 0.0391 MM y la volatilidad se incrementa en USD 0.005 MM, generando un decremento en el ratio de Sharpe del mismo de USD 0.1299 MM. Esto es más que esperable dado que este nos permite medir el rendimiento de nuestro portafolio en comparación con el activo libre de riesgo después de ajustar por el riesgo o volatilidad correspondientes y, actualmente en el mercado se está observando un decremento del rendimiento esperado de los activos riesgosos y un aumento considerable de la volatilidad. Esto se puede apreciar por la evolución del (VIX) entre los meses

de Septiembre y el día en el cuál se llevó a cabo la segunda estimación (3 de Noviembre de 2021):

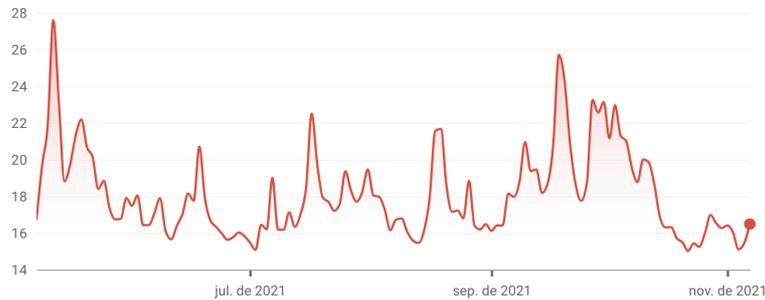


Figura 15: Evolución del Board Options Exchange Market Volatility Index (VIX) entre T_0 y T_1 .

Fuente: Google Finance(<https://www.google.com/finance/quote/VIX:INDEXCBOE?window=6M#DAdeKc-384>).

Posibles mejoras por aplicar en la solución propuesta

A continuación, mencionaré algunas herramientas que, tomando como punto de partida la solución desarrollada, nos permitirán cubrir la totalidad del proceso de administración de una cartera de inversión y, optimizar el proceso de retroalimentación para poder agilizar y obtener la mayor ventaja competitiva posible a la etapa de rebalanceo de nuestro portafolio.

Aplicación de redes neuronales y algoritmos genéticos para optimizar el proceso de estimación de los retornos ante un contexto específico

Una red neuronal es un sistema perteneciente a la rama de la inteligencia artificial dentro de la informática que, basándose en la estructura del cerebro humano, nos permite estimar comportamientos futuros y resolver problemas. Se dice que se basa en la estructura del cerebro humano ya que emula la interconexión de las neuronas mediante las dendritas y el traspaso de impulsos eléctricos o información entre ellas, transformando la salida de cada una de las neuronas en una entrada para la siguiente.

En el ámbito de la ingeniería informática, la aplicación de las redes neuronales se lleva a cabo mediante la implementación de n capas de nodos, distribuyéndolos en tres grandes grupos: la capa de entrada, la capa oculta y la capa de salida. Todos los nodos o unidades básicas de procesamiento (neuronas) son iguales y, en base a un input, entregarán una salida u output dependiendo de la lógica de este. Las capas de nodos se encuentran totalmente interconectadas entre sí, permitiendo de esta manera que las salidas de los nodos de una capa sean propagadas a todas las unidades de procesamiento de la capa siguiente, transformándose en el input de estas. A continuación, mostramos un ejemplo gráfico de cómo es la estructura de una red neuronal.

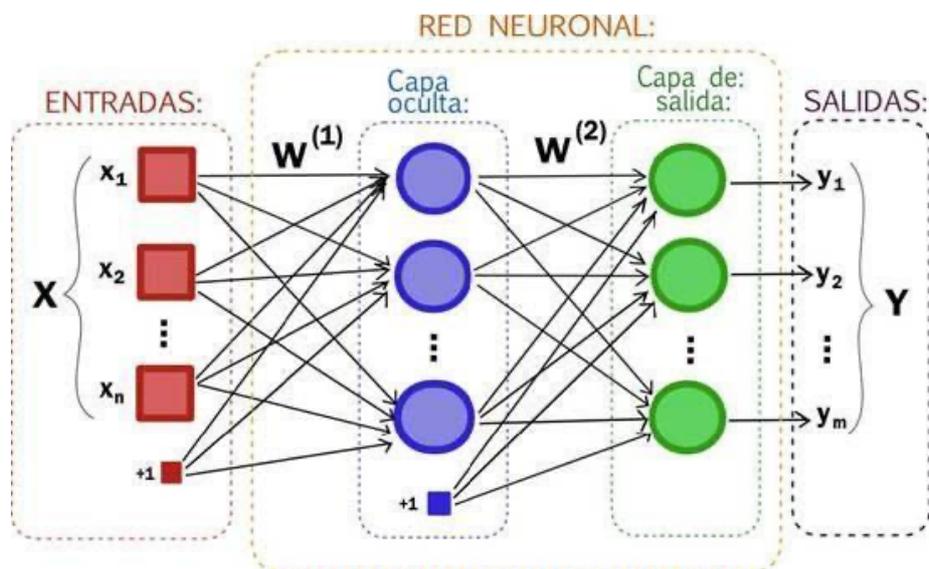


Figura 16: Estructura de una red neuronal multicapa.

Fuente: <https://sites.google.com/site/mayinteligenciartificial/unidad-4-redes-neuronales>.

Nótese, que cada una de las entradas de cada nodo posee una ponderación asociada a la misma. Esto se debe a que cada nodo recibirá n entradas, dado que los nodos de cada capa están asociados con todos los nodos de la capa anterior y, por ende, recibirán la salida de cada uno de ellos. Así, el input efectivo del nodo en la capa actual se verá reflejado por la siguiente función:

$Input = \sum_{i=1}^n x_i \times w_i$, donde x_i es el valor de la entrada i del nodo y w_i es el peso asociado a la misma.

Un aspecto muy importante para destacar en las redes neuronales es que su funcionamiento y la calidad de las predicciones depende totalmente del entrenamiento que se le dé a la red. Esto se debe a que, a medida que se vaya entrenando a la red neuronal, se irá corrigiendo la lógica de cada uno de los nodos, para que respondan de una manera determinada a los estímulos que los mismos recibirán. He aquí el aspecto más importante de las redes neuronales: el entrenamiento y el tipo de aprendizaje en el que se incurre para poder reducir el error y ajustar la estimación. Cabe aclarar que, en los casos de aplicación en finanzas, la salida de una red neuronal puede ser un valor numérico, como ser el de una regresión lineal. En este sentido, existen dos grandes tipos de redes neuronales: las redes neuronales con aprendizaje supervisado y las redes neuronales con aprendizaje no supervisado.

Redes neuronales con aprendizaje no supervisado

En este tipo de redes neuronales, el modelo se ajusta automáticamente a las observaciones. A diferencia de las redes neuronales de aprendizaje supervisado, éstas son útiles para escenarios en los que no existe un conocimiento a priori. Por ende, el aprendizaje no supervisado tratará al input o información de entrada como un conjunto de variables aleatorias, construyendo un modelo de densidad probabilística para el conjunto de datos mediante la inferencia bayesiana, para, de esta manera, asignar probabilidades condicionales para cualquiera de las variables aleatorias dadas. Esto quiere decir que es necesario que la red sea capaz de crear un código factorial de los datos; es decir, componentes estadísticamente independientes (variables aleatorias independientes).

Ahora bien, un gran problema de las redes neuronales con aprendizaje no supervisado es que las mismas se autoajustan en base al conjunto de datos de entrenamiento, por lo que en el caso de que la misma se entrene por demás, si bien arrojará soluciones muy precisas para el conjunto de datos en cuestión, será extremadamente dependiente de los mismos. Por lo que,

en caso de que se aplique la red neuronal a un conjunto de datos distinto o de cualidades diferentes, arrojará una solución con un nivel de error muy alto. A este tipo de sobreentrenamiento o sobreajuste en las redes neuronales de aprendizaje no supervisado, se lo conoce como overfitting.

Redes neuronales con aprendizaje supervisado

A diferencia de las redes neuronales de aprendizaje no supervisado, en estas, el conocimiento (información) previo nos permitirá deducir funciones y ajustarlas para que, en base a un conjunto de datos de entrenamiento, nuestra red neuronal arroje un output o solución relativamente precisa, o con un umbral de error que se encuentre dentro de los parámetros del nivel de confianza del modelo. En otras palabras, el objetivo principal de los algoritmos de aprendizaje supervisado es el de encontrar una función a la que llamaremos “g” capaz de estimar un valor en base a una muestra de variables aleatorias independientes e idénticamente distribuidas con una función de distribución de probabilidad desconocida “p”, asegurando un grado de certeza a partir del entrenamiento de ésta en base a un conjunto de datos u observaciones.

Un aspecto fundamental en lo relacionado con la certeza de la predicción que lograremos se encuentra definido por la función de pérdida de la red neuronal. Esta función, la cual llamaremos “L” ($L(z, y)$) representa la pérdida sufrida en la predicción de “z”, que representa el valor de “g” estimado (cuando su valor verdadero es “y”).

En base a lo explicado anteriormente, queda en claro que en ambos casos un aspecto fundamental para toda red neuronal es el conjunto de datos que se utiliza para entrenar a la misma y ajustar el algoritmo que aplicará cada una de sus neuronas. Esto se debe a que la estimación que llevará a cabo la red neuronal será para un conjunto de datos que esté asociado o sea similar al conjunto de datos que se utilizó para entrenar a la misma. En base a este punto clave, cabe destacar uno de los errores importantes que se suele observar cuando se implementa de manera errónea el proceso de data mining, fallas en el análisis y selección de observaciones que estén relacionadas al contexto en el cual nuestro algoritmo llevará a cabo las estimaciones. Caso contrario, el mismo no podrá explicar de manera correcta el comportamiento y la relación entre las distintas variables y, por ende, las estimaciones del modelo serán significativamente distintas a los resultados ex post.

Ahora, por qué es tan importante el proceso de data mining. Esto se debe a que a partir de un conjunto de datos determinado, nos permitirá detectar nueva información sobre los mismos en base a patrones y tendencias o reglas que expliquen su comportamiento en un contexto específico. Esto es fundamental por dos aspectos:

- Nos permitirá lograr mejores estimaciones en nuestros modelos dado que el R^2 será mayor y, por ende, explicará mejor el comportamiento de los datos.
- Nos permitirá tener en cuenta en la toma de decisiones nuevas variables o información que no esta a la vista.

Con respecto a este último punto (Nos permitirá tener en cuenta en la toma de decisiones nuevas variables o información que no esta a la vista), nótese que esta nueva información obtenida del proceso de data mining será sólo válida y aplicable al contexto de los datos sobre los cuales se llevó a cabo el proceso. Dichas características no serán significativamente explicativas con respecto al comportamiento de los datos en un contexto distinto o, en un extremo, generarán un ruido que nos llevará a tomar malas decisiones.

Riesgos de aplicar inteligencia artificial y algoritmos genéticos durante el proceso de estimación del retorno de nuestro portafolio

Si bien los beneficios en la implementación de IA en el proceso de administración de carteras de inversión son numerosos y se encuentran a la vista, no podemos depender únicamente de ella, y siempre es primordial que exista un profesional idóneo de las finanzas al mando, utilizando IA de manera complementaria a la inteligencia humana en búsqueda de mejores resultados, y no así en su reemplazo.

Como ya fue previamente mencionado, es menester considerar la problemática de que los datos difícilmente consigan diferenciar las señales del ruido de manera confiable, generando un posible overfitting que no permitirá que los algoritmos de trading puedan adaptarse correctamente a las nuevas circunstancias (con el riesgo que ello implica a la hora de administrar activos). En consecuencia, es de suma relevancia implementar la posibilidad de intervención de algoritmos alternativos en base a cambios de escenarios. La mala calidad o insuficiencia de datos se traducirá, caso contrario, en un input sobre el cual se extraerán outputs de dudosa confiabilidad. Es por ello, que es fundamental que el usuario de estas herramientas se encuentre altamente entrenado para poder utilizar los datos indicados y distinguir los modelos robustos de aquellos que no lo son. Otra herramienta que se puede utilizar para evitar esta problemática es el cross-validation o out-of-sample testing, el cual generaliza el dataset aplicando un algoritmo de resampling que nos permite utilizar de manera aleatoria distintas partes de un dataset en las diferentes iteraciones para, de esta manera, generalizar el análisis estadístico que se lleva a cabo a partir de sus observaciones que se poseen para poder entrenar a nuestra red neuronal.

Aplicación del Internet Of Things - IOT para optimizar el proceso de captar señales y poder gestionar nuestro portafolio de manera coherente con nuestra estrategia

La inteligencia artificial está siendo considerada una herramienta de suma importancia en la gestión de activos ya que constituye un recurso extremadamente útil a la hora de analizar datos a gran escala y con mayor profundidad. Varios ETFs y Hedge Funds ya utilizan modelos estadísticos de inteligencia artificial para su operatoria habitual e inclusive un informe publicado por el CFA Institute⁶ al final del año 2020, sostiene que son cada vez más los administradores de activos que utilizan la IA para operar en sus plataformas de inversión y trading, aunque también afirma que está lejos de reemplazar por completo a la actividad humana en el campo. Es fundamental en este ámbito que todo algoritmo y aplicación de inteligencia artificial sea del tipo de aprendizaje supervisado y, que esto efectivamente sea analizado y retroalimentado tanto por un ingeniero de datos como por un profesional del mundo de las finanzas.

Otro aspecto muy interesante para plantearnos es el hecho de aplicar Internet Of Things para, de esta manera, poder obtener más y mejor información aplicando los filtros correspondientes y pertinentes ante todo posible ruido. He aquí un aspecto de suma relevancia para el proceso de administración de una cartera de inversión. Esto se debe a que es una herramienta clave para identificar información que se pueda transformar en señales que nuestro agente inteligente o el administrador de la cartera de inversión pueda utilizar en la toma de decisiones en base a la estrategia del modelo financiero. En base a esto se puede utilizar como ejemplo un informe del año 2020 que la consultora norteamericana Cerulli Assoc publicó en que expone que los Hedge Funds que utilizaron este tipo de tecnologías mostraron una ventaja con respecto a los que no. En este informe, si bien el rendimiento a partir de Marzo de 2020 fue mejor en los casos de fondos sin IA vs con IA, se observa que que los Hedge Funds que

6 Artificial Intelligence in Asset Management - Söhnke M. BartramJürgen BrankeMehrshad Motahari

aplicaron IA obtuvieron resultados positivos acumulados hasta dicho momento de aproximadamente un 34% durante los tres años previos a dicho momento:

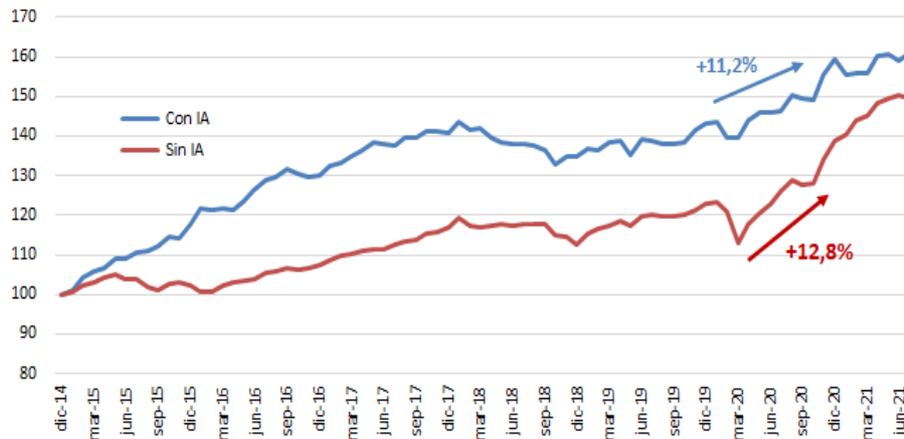


Figura 17: Comparación de Hedge Funds con IA vs sin IA.

Fuente: Elaboración propia en base a información de www.eureka hedge.com

Conclusión

En base a lo mencionado en el trabajo y a la solución desarrollada, se puede observar claramente las ventajas que ofrece el hecho de aplicar la tecnología para la administración activa de una cartera de inversión, permitiéndonos llevar a cabo estimaciones más precisas sobre el riesgo y el rendimiento de nuestro portafolio de inversión y de una manera mucho más performante teniendo en cuenta el grado de complejidad que implica este proceso a medida que la composición de activos de nuestro portafolio se va incrementando. Más allá de esto, dado el gran volumen de información que tendremos a partir de los datos históricos de los rendimientos realizados de los activos financieros a lo largo de la historia, aplicándolas de manera correcta podremos llevar a cabo técnicas de big-data que nos permitirán atacar el problema de poder explicar mejor el comportamiento y los factores que influyen en los *risk-drivers* de estos.

La pregunta entonces es, más allá de esto, como explicamos en el apartado asociado a la aplicación y utilización de redes neuronales a algoritmos para poder estimar el riesgo y retorno esperados de nuestros portafolios y activos financieros, es fundamental que para esta aplicación el aprendizaje de este sea supervisado. Además, otra consideración que se debe tener en cuenta al ahora de aplicar la inteligencia artificial a este tipo de procesos, es evaluar si mi filosofía de inversión es adaptable o no a un proceso sistemático y que pueda ser amoldable al machine learning. Del mismo modo, el profesional encargado de esta tarea también deberá evaluar la relación entre el costo y el beneficio que esto nos generará en base a la situación de partida en la cual se analiza la posible implementación de este tipo de soluciones, dado que si el alfa o ganancia que nos puede llegar a generar no es lo suficientemente conveniente para afrontar el costo de la implementación dicha decisión no será conveniente.

Bibliografía

- Bodie Kane, Marcus. Investments (Eleventh Edition): Mc GrawHill.
- Ludwing Chincarini, Daehwan Kim. Quantitative Equity Portfolio Management. An Active Approach to Portfolio Construction and Management: Mc GrawHill.
- Edwin J. Elton, Stephen J. Brown, Martin Gruber, William N. Goetzmann. Modern Portfolio Theory and Investment Analysis (Ninth Edition): Wiley.
- Yilun Chen, Student Member, IEEE, Ami Wiesel, Member, IEEE, Yonina C. Eldar, Senior Member, IEEE, and Alfred O. (October 2010). Shrinkage Algorithms for MMSE Covariance Estimation.
- Sharpe, W.F. (1964). Capital Assets Prices: A Theory of Market Equilibrium Under Conditions of Risk.
- Marty Sapp, Ed.D., Festus E. Obiakor, Ph.D., Amanda J. Gregas, and Steffanie Scholze. Mahalanobis Distance: A Multivariate Measure of Effect in Hypnosis Research.
- Scott D. Stewart, Christopher D. Piros and Jeffrey C. Heisler. Portfolio Management, Theory and Practice: Wiley.
- Aprendizaje Supervisado: Una Revisión de la Clasificación de las Técnicas de Informática, 2007: S. Kotsiantis.
- Introducción a estadística de relación de aprendizaje, 2007: Lise, Getoor y Taskar, Ben.
- Artificial Intelligence in Asset Management Söhnke M. Bartram Jürgen Branke Mehrshad Motahari.
- "Artificial intelligence could yet upend the laws of finance," Stuart Kirk, Financial Times. 22 January 2018
- Artificial intelligence in finance - Bonnie G. Buchanan, PhD, FRSA.
- El auge del científico de datos: Modelos de machine learning para el futuro. – Refinitiv.

Autorización para publicar los trabajos finales

- **Repositorio Institucional:**

SI autorizo a la Universidad del CEMA a publicar y difundir en el **Repositorio Institucional** de la Universidad de la Biblioteca con fines exclusivamente académicos y didácticos el Trabajo Final de mi autoría.

- **Catálogo en línea:**

SI autorizo a la Universidad del CEMA a publicar y difundir en el **Catálogo en línea** (acceso con usuario y contraseña) de la Biblioteca con fines exclusivamente académicos y didácticos el Trabajo Final de mi autoría.

- **Página web UCEMA:**

SI autorizo a la Universidad del CEMA a publicar y difundir en la **página web de la Universidad** como Trabajo destacado, si el mismo obtuviese la distinción correspondiente, con fines exclusivamente académicos y didácticos el Trabajo Final de mi autoría.



Battafarano, Maximiliano