

Selección automática de una cartera óptima a partir de datos importados de Internet. Una aplicación VBA¹

Pérez-Salamero González, Juan Manuel y Ventura Marco, Manuel
juan.perez-salamero@uv.es (contacto), manuel.ventura@uv.es
*Departament d'Economia Financera i Actuarial
Universitat de València*

RESUMEN

Según el Programa Básico de Estudios (“Core Syllabus”) para la formación actuarial en Europa, la formación de un actuario de seguros exige que los estudios conducentes a la capacitación profesional otorguen al estudiante “un conocimiento práctico, aplicado, de las modernas tecnologías de la información y la comunicación”.

En esta comunicación se presenta una aplicación docente realizada con Visual Basic Aplicado en la que se ha implementado un modelo de selección de cartera óptima clásico, utilizando las herramientas de optimización y de importación de datos que proporciona Excel. La cartera contempla a los activos del IBEX-35 y se ha programado el cálculo automático de la composición óptima conforme se actualicen periódicamente las cotizaciones de dichos activos.

La aplicación se aborda en la asignatura “Programación en Visual Basic Aplicado” del Máster Universitario en Ciencias Actuariales y Financieras de la Universitat de València.

Esta aplicación integra conceptos tratados en distintas asignaturas, requiere un software relativamente disponible y muestra al estudiante las potencialidades prácticas del uso de la programación en su futura carrera profesional.

¹ Los autores agradecen la asistencia financiera recibida del Ministerio de Economía y Competitividad, proyecto ECO2012-36685. Gracias igualmente a los/as evaluadores/as anónimos/as por sus enriquecedores comentarios. Cualquier error es completamente responsabilidad de los autores.

ABSTRACT

According to the "Core Syllabus" for actuarial training in Europe, the formation of an insurance actuary requires studies leading to professional qualification grant the student "practical knowledge, applied modern technologies information and communication".

This paper shows an educational application performed with Visual Basic Applied where we have implemented a model of classical optimal portfolio selection, using optimization tools and import data provided by Excel. The portfolio includes the IBEX-35 assets and is scheduled automatic calculation of the optimal composition as the prices of these assets are periodically updated.

The application is addressed in the subject "Programming in Visual Basic Applied", a course of the Master in Actuarial Science of University of Valencia.

This application integrates concepts discussed in different subjects, requires relatively available software and shows the student the practical potential of using programming in their future career.

Palabras claves:

Visual Basic para Aplicaciones; Microsoft Office Excel; objeto de aprendizaje; Proyecto Orientado al Aprendizaje; selección de cartera; frontera eficiente; captura automática de datos.

Área temática:

A1. Metodología y Docencia.

1. INTRODUCCIÓN

En la versión del “Core Syllabus” (Groupe Consultatif Actuariel Europeen, 2011, pág. 3), para la formación de actuarios en Europa a implantar en el curso 2014-2015, se establece como competencia básica:

“1. Computing.

Aim: To provide a grounding in modern computing methods necessary for the work of an Actuary.

The student is expected to have a working knowledge of modern information communication and technology (ICT) as appropriate for the work of an actuary.”

Así pues, se mantiene la exigencia de conocimientos sobre Informática de las anteriores versiones del “Core Syllabus, cuyo cumplimiento se ve reflejado en los planes de estudio del Máster en Ciencias Actariales y Financieras impartidos en la Universidad de Alcalá de Henares, Universidad de Barcelona, Universidad Carlos III, Universidad Complutense, Universidad de Málaga y Universidad de Valencia (Instituto de Actuarios Españoles, 2015). La **Tabla 1** relaciona las materias que incorporan en su temario, en mayor o menor medida, contenidos relacionados con Visual Basic Aplicado (VBA) a Excel.

Tabla 1. Asignaturas con contenido de VBA. Máster en Ciencias Actariales y Financieras

Universidad	Asignatura
Alcalá de Henares	Introducción al Visual Basic y Modelos de Cartera.
Barcelona	Programación y Aplicaciones.
Carlos III	Herramientas de Cálculo Actuarial.
Complutense	Informática y Cálculo Numérico.
Málaga	Métodos y Modelos de Cálculo para Aplicaciones Actariales I.
Valencia	Programación en Visual Basic Aplicado.

Fuente: página Web del Máster de cada Universidad

En los distintos títulos que han ido formando a los actuarios en la Universidad de Valencia, el tratamiento del requisito de conocimientos en Informática ha evolucionado y esta evolución está descrita en el trabajo de Ventura Marco (2013b). Según este trabajo, la última etapa de la docencia de Informática, en concreto de Programación en VBA, está orientada al diseño de aplicaciones por parte de los estudiantes como una metodología que ha proporcionado mejores resultados académicos que las anteriores.

De acuerdo con esta metodología, en los tres últimos cursos académicos del Máster en Ciencias Actuariales y Financieras, se ha incorporado al sistema de enseñanza-aprendizaje la elaboración de un Proyecto por parte de los estudiantes (Project Oriented Learning, POL). La descripción de cómo se ha implementado el POL hasta el presente curso se recoge en Ventura Marco (2013a).

En las instrucciones facilitadas a los estudiantes para elaborar el “Proyecto VBA Excel para modelos financiero/actuariales” se facilita unas referencias bibliográficas para que se pueda escoger el proyecto a realizar². Estas referencias son genéricas sobre el diseño de aplicaciones en Excel y/o VBA o se centran en el diseño de aplicaciones financieras. En materia estrictamente actuarial no existen referencias de desarrollo y diseño de aplicaciones en Excel y/o VBA, pero sí se dispone de referencias del uso de Excel como soporte de ayuda o de códigos de programación en BASIC para la realización de determinadas tareas y/o para su resolución concreta, motivo por el que en el sistema de evaluación por rúbricas del Proyecto se valora en mayor medida una aplicación de tipo actuarial basada en cálculos o modelos aprendidos por el/la estudiante en otras materias del Máster.

Además de las referencias bibliográficas para orientar la elección del POL por los estudiantes, para elaborar dicho proyecto se requiere del uso y disponibilidad de objetos de aprendizaje por su parte, con mayor o menor calidad. Desde ficheros Excel sencillos, tipo ejemplo auto-contenido, que plasmen un método o un caso específico hasta complejas aplicaciones, pasando por el empleo de los proyectos elaborados por estudiantes de cursos anteriores.

Según Velázquez et al. (2005, pág. 329) y Parada Gandos (2010) no existe una definición completamente aceptada de “Objeto de Aprendizaje” (OA). La definición de un OA más utilizada y que estos autores citan es la que ha proporcionado el Institute of Electrical and Electronics Engineers (IEEE): “Es una entidad digital o no digital, que puede ser usada, reusada o referenciada durante el aprendizaje soportado por la tecnología”.

² Una buena parte de referencias se encuentran en Ventura Marco (2013b, págs. 13-14).

La Universidad Politécnica de Valencia (ASIC-ICE, 2008) opta por adaptar la definición de OA que recoge Wiley (2000), a partir de la definición de trabajo del Learning Technology Standards Committee (LTSC) creado por el IEEE, y propone la siguiente definición: “la unidad mínima de aprendizaje, en formato digital, que puede ser reusada y secuenciada”. Según esta definición, para ASIC. ICE (2008, pág. 4), las características que debe tener un OA son las siguientes: formato digital; propósito pedagógico; contenido interactivo; indivisible e independiente y reutilizable.

Según Velázquez et al. (2005, pág. 331) la “calidad de un objeto de aprendizaje se conforma de 3 elementos, que son los tecnológicos, los pedagógicos y los de contenido”. Los primeros hacen referencia a la reutilización y la adaptabilidad, los segundos se refieren a los elementos que facilitan el proceso enseñanza-aprendizaje (número de ejemplos, experimentación/interacción, evaluación,...) y respecto a los terceros, se valora el nivel de detalle y de complejidad con que se aborda el tema.

Con la finalidad de proporcionar OA a los estudiantes de “Programación en VBA” del Máster en CC. Actuariales y Financieras, en esta comunicación se presenta una aplicación docente realizada con Visual Basic Aplicado en Excel en la que se ha implementado un modelo clásico de selección de cartera óptima, utilizando las herramientas de optimización y de importación de datos que proporciona Excel.³ La cartera contempla a los activos del IBEX-35, además de un activo sin riesgo ni rentabilidad (dinero en efectivo o cuenta corriente sin remuneración) y se ha programado el cálculo automático de la composición óptima conforme se actualicen periódicamente las cotizaciones de dichos activos.

Muchas de las características de esta aplicación se ajustan a las requeridas para disponer de un OA de calidad, pero otras impedirían catalogarla como tal atendiendo a la definición y requisitos que se exijan. Sin embargo, convenientemente adaptada⁴ puede usarse como un OA tanto para el aprendizaje de la teoría clásica de Gestión de Cartera como para la Programación en VBA de una aplicación financiera. Si se complementa con distintos OA de acoplamiento como son una “Guía Introductoria”,

³ La versión que se presenta se ha desarrollado y ampliado desde el año 2002 y diversas ediciones del curso de VBA en Excel.

⁴ Se podrían seguir los consejos de Rodríguez Bernal (2014) para los aspectos de optimización.

una “Actividad práctica” y una “Evaluación”, así como un “Resumen” (ASIC-ICE, 2008, pág. 9) dará como resultado un Módulo de Aprendizaje (MA).

Esta aplicación integra conceptos tratados en otra asignatura de los Másteres en Ciencias Actuariales y Financieras en relación con la Gestión de Carteras o, como es el caso de la Universidad de Alcalá de Henares, en la propia asignatura que imparte también VBA, por lo que tras una adaptación podría utilizarse también como OA en dichas materias para ilustrar la teoría clásica de gestión de cartera y el concepto de frontera eficiente.

La aplicación requiere un software relativamente disponible, como es Microsoft Excel, y muestra las potencialidades prácticas del uso de la programación en su futura carrera profesional, por lo que pretende animar a su uso.

Tras esta introducción, el segundo apartado describe el planteamiento básico específico de selección de cartera que se ha contemplado en el diseño de la aplicación. En el tercer apartado se detalla la estructura del libro de Excel así como el diseño del proyecto de VBA para determinar la composición de la cartera óptima de forma automatizada importando datos desde la Web. Las conclusiones valoran los resultados que proporciona la aplicación presentada como OA, así como sus potenciales usos de cara al desarrollo dirigido de un POL o a la creación de un MA.

2. PLANTEAMIENTO DEL MODELO DE SELECCIÓN DE CARTERA

Hay muchos trabajos que han tratado la gestión de cartera o conjunto de inversiones o activos, pero en esta aplicación se va a considerar el problema de selección de la cartera óptima⁵. El modelo básico más conocido es el modelo de optimización media-varianza de Markowitz (1952, 1959 y 1987), un modelo estático o de un solo período que se sustenta en el binomio rentabilidad-riesgo, en el que dado un presupuesto se trata de seleccionar la combinación óptima de n posibilidades de

⁵ Se pueden citar algunos ejemplos de trabajos que analizan la gestión de cartera mediante el uso de Excel como el de Piñol Espasa (2013) para carteras de opciones; el de Fogués Zornora y Jiménez Fernández (2012) con un modelo distinto al de Markowitz y otro con más similitudes al presentado en estas páginas, Jimeno Moreno (2004, págs. 231-245).

inversión, teniendo en cuenta dos índices en el criterio de óptimo: el rendimiento esperado de la cartera y el riesgo asociado a la misma obtenido como la varianza o desviación estándar de dicho rendimiento.

El problema de selección con ese binomio puede formularse de varias maneras⁶, pero para la aplicación a desarrollar en Excel se ha optado por el siguiente planteamiento matemático general del problema:

	$Min \{V(x_1, x_2, \dots, x_n) - \beta \cdot R(x_1, x_2, \dots, x_n)\}$	(1)
sujeto a:		
	$\sum_{j=1}^n x_j = 1$	(2)
	$0 \leq d_j \leq x_j \leq u_j \quad \forall j = 1, 2, \dots, n$	(3)

donde

n : es el número de inversiones distintas posibles.

x_j : porcentaje de la inversión total o presupuesto dedicado a la inversión j -ésima. (4)

$$V(x_1, x_2, \dots, x_n) = (x_1, x_2, \dots, x_n) \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad (5)$$

Varianza total de la cartera. Mide su riesgo. Se trata de una forma cuadrática semi-definida positiva y, por tanto, una función convexa.

a_{ij} : covarianza entre las rentabilidades esperadas de las inversiones i -ésima y j -ésima. (6)

$$\beta \geq 0 \quad (7)$$

Ponderación de la rentabilidad respecto al riesgo. Si $\beta = 0$, al inversor le es indiferente la rentabilidad obtenida. A mayor valor de β se prefiere mayor rentabilidad frente a menor riesgo, actuando a modo de coeficiente de aversión al riesgo.

⁶ En Ventura; Meneu y Pérez-Salamero (2000, epígrafe 2.7 y Anexo 3) hay una breve descripción de los distintos planteamientos del problema.

$$R(x_1, x_2, \dots, x_n) = \sum_{j=1}^n E[r_j] \cdot x_j \quad (8)$$

Rendimiento esperado total de la cartera.

$$E[r_j]: \text{rentabilidad esperada de la inversión } j\text{-ésima.} \quad (9)$$

$$u_j > 0: \text{cota superior admisible para la inversión } j\text{-ésima.} \quad (10)$$

$$d_j \geq 0: \text{cota inferior admisible para la inversión } j\text{-ésima.} \quad (11)$$

Dadas las propiedades de las funciones que intervienen en la función objetivo (1) como en las restricciones, se tiene un problema de programación cuadrática cuyas soluciones serán óptimos globales. Resolviendo este mismo problema para distintos valores de β , se obtendrá un par (R^*, V^*) , rendimiento y varianza óptimas. Representando en un gráfico esos distintos pares óptimos (en el eje de ordenadas el rendimiento óptimo y en el eje de abscisas el riesgo de la cartera), se obtiene la llamada “Frontera eficiente”, Jimeno Moreno (2004, pág. 240).

En la aplicación que se ha creado se cuenta con $n = 36$, resultado de los 35 títulos del IBEX más un activo sin riesgo: efectivo o dinero en cuenta corriente sin remunerar, aunque se puede modificar esta última circunstancia. No se contempla posiciones cortas, exigiendo la no negatividad de los porcentajes de participación. Por (2) y (3), si se omite el dato para u_j para algún activo, actuará la cota superior general de la unidad (100%); y si se omite el dato para d_j actuará la cota inferior general de 0.

3. PROYECTO DE GESTIÓN DE CARTERA AUTOMATIZADA

A continuación se describe la estructura del fichero Excel “Cartera_IBEX35.xlsm”, las herramientas de Excel utilizadas y el código de VBA utilizado para su manejo, así como los controles ActiveX incorporados.

3.1. Estructura del libro.

El libro se ha organizado en las siguientes hojas: “Principal”; “Datos del modelo”; “Web IBEX”; 35 hojas de cotizaciones históricas de empresas: “Empresa1” a “Empresa35” y “Resultados”.

3.1.1. Hoja "Principal".

En la hoja principal se ha dispuesto la información necesaria para el planteamiento del problema y posterior resolución mediante Solver. **Tabla 2.**

Tabla 2. Estructura del problema de gestión de cartera. Hoja "Principal".

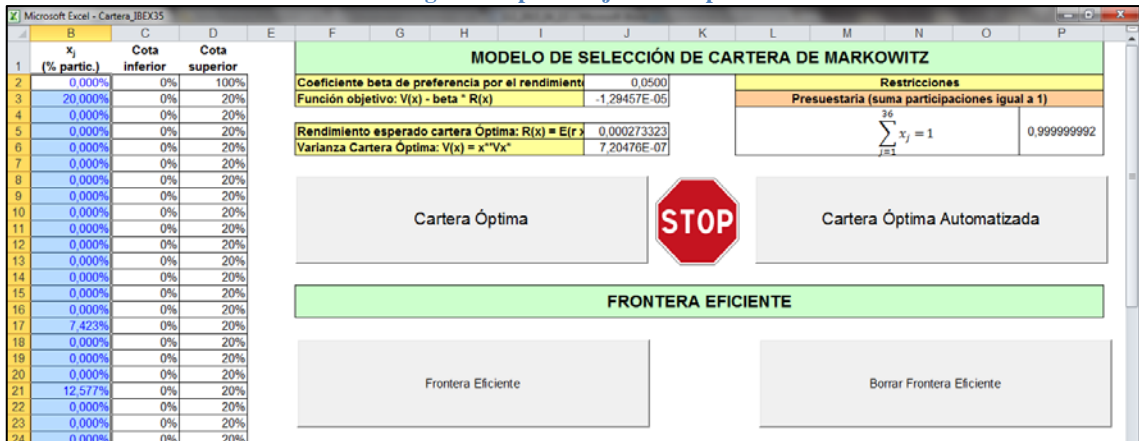
Expresión	Rango	Nombre	Fórmula
(1)	J3	Funcion_Objetivo	var_cartera-beta*rent_cartera
(2)	P4	Presupuesto	SUMA(Participacion)
(4)	B2:B37	Participacion	Valores variables independientes
(5)	J6	var_cartera	MMULT(MMULT(TRANSPONER(Participacion); var_covar);Participacion)
(7)	J2	beta	Dato
(8)	J5	rent_cartera	MMULT(rent_media;Participacion)
(10)	C2:C37	d_cotas_inf	Datos
(11)	D2:D37	u_cotas_sup	Datos

Se ha incluido un botón de comando "Cartera Óptima" que ejecuta manualmente la rutina en VBA que resuelve mediante Solver el problema y asigna el valor óptimo a las celdas de las variables independientes, x_j , "Participación".

En una segunda zona de la hoja se han insertado dos botones de comando para generar y eliminar el gráfico de la "Frontera Eficiente". Se ha tenido que insertar en la misma hoja, ya que Solver debe ejecutarse desde la hoja en donde se encuentren los datos del planteamiento (Moreno Bonilla y Cano Rodríguez, 2010, pág. 332). Además, se ha preparado la cabecera de la tabla de datos de rendimiento y varianza óptimas para distintos valores de β .

Se ha incorporado dos botones para la gestión automatizada de la cartera. Uno la activa y almacena los resultados en la hoja correspondiente. El otro detiene la rutina de VBA si se pulsa antes de finalizar el proceso. El aspecto se aprecia en la **Imagen 1.**

Imagen 1. Aspecto hoja "Principal".



3.1.3. Hoja “Datos del modelo”.

La hoja de Datos está organizada en tres zonas:

- Zona de datos de cotizaciones de cada empresa recogidos en las hojas “Empresa#”; cuyo aspecto se aprecia en la **Imagen 2**. Se recoge la información de las últimas 52 cotizaciones de cada empresa del IBEX que estén almacenadas en las hojas de “Empresa#”. La razón de ese número es doble, ya que es el número aproximado de semanas del año y el número de datos de cotizaciones que se vienen a actualizar en la Web de BME (2015) a lo largo de una sesión diaria. En el rango A2:A53 se almacena el día del dato de cotización (el mismo para los 35 títulos), mientras que en el rango B2:B53 se almacena la hora y minuto de la cotización, que no siempre es la misma cuando se actualiza la Web. Se ha tomado la última hora de los 35 títulos en cada actualización mediante el uso en el procedimiento VBA de la función MAX() de Excel. Los datos de cotizaciones se almacenan en el rango D2:AL53.

Imagen 2. Aspecto zona de cotizaciones. Hoja “Datos del modelo”.

	A	B	C	D	E	AK	AL
1	Fecha	Hora		Empresa 1	Empresa 2	Empresa 34	Empresa 35
2	10/04/2015	17:38:00		3,37	17,47	40,68	13,185
3	10/04/2015	17:29:00		3,367	17,44	40,57	13,17
4	10/04/2015	17:20:00		3,367	17,435	40,56	13,17
52	10/04/2015	9:25:00		3,32	17,545	40,545	13,07
53	10/04/2015	9:10:00		3,332	17,5	40,68	13,03

- Zona de cálculo de rendimientos esperados por intervalo de tiempo. Para el cálculo de los rendimientos se hace la hipótesis de que el intervalo de tiempo entre dato de cotización y su consecutivo es siempre el mismo. Dicha hipótesis supone que el tiempo establecido para actualizar los datos es el mismo durante la ejecución del programa de gestión automatizada, pero eso no garantiza que en una misma actualización los 35 títulos tengan el mismo dato de minuto⁷. Además, si se tiene en cuenta la presencia de datos de cotizaciones de dos jornadas diferentes, es más evidente que el procedimiento de cálculo del rendimiento que se ha seguido en nuestra aplicación es erróneo al suponer el mismo período de devengo del rendimiento. El rendimiento del activo “Efectivo” se ha establecido en el 0%, mientras que el de las 35 acciones se ha

⁷ En una futura versión, se trataría de calcular automáticamente el rendimiento anual equivalente, teniendo en cuenta el verdadero período de devengo del rendimiento.

determinado sin contemplar el posible cobro de dividendos como el cociente entre la variación del precio entre dos instantes consecutivos (posterior menos el anterior), y el precio del momento anterior. Se obtienen, pues, 51 rendimientos para cada uno de los 35 activos del IBEX, se ha asignado un nombre al rango de celdas que contiene los 51 rendimientos para cada empresa (“rend_#”), así como para el “Efectivo” (“rend_0”). Para el correcto funcionamiento, se requiere contar con los 52 datos previos de cotizaciones para poder iniciar la búsqueda de la cartera óptima. En la **Imagen 3** se aprecia la disposición de los datos de rendimiento por instante o momento.

Imagen 3. Aspecto zona de rendimientos. Hoja “Datos del modelo”.

	AM	AN	AO	BU	BV	BW
1	Rendimientos					
2	Efectivo	Empresa 1	Empresa 2	Empresa 34	Empresa 35	
3	0,0000000%	0,0891001%	0,1720183%	0,2711363%	0,1138952%	
4	0,0000000%	0,0000000%	0,0286779%	0,0246548%	0,0000000%	
52	0,0000000%	0,0000000%	-0,1424907%	0,5056110%	0,1912777%	
53	0,0000000%	-0,3601441%	0,2571429%	-0,3318584%	0,3069839%	

- Zona de cálculo del riesgo de la cartera a través de la matriz de varianzas-covarianzas, así como del rendimiento medio de cada activo. Las celdas con parte de información del planteamiento del problema de gestión de cartera se relacionan en la **Tabla 3**.

Tabla 3. Estructura del problema de gestión de cartera. Hoja "Datos del modelo".

Expresión	Rango	Nombre	Fórmula
(5) y (6)	C57:AL92	var_covar	COVARIANZA.M(rend_#;rend_##)
(8)	C93:AL93	rent_media	PROMEDIO(rend_#)

El aspecto de esta zona se recoge en la **Imagen 4**.

Imagen 4. Aspecto zona de cálculo del riesgo y rendimiento de la cartera. Hoja “Datos del modelo”.

	B	C	D	E	AK	AL	AM
55	Matriz de varianzas covarianzas						
56	Activos	Efectivo	Empresa 1	Empresa 2	Empresa 34	Empresa 35	
57	Efectivo	0,0000000000	0,0000000000	0,0000000000	0,0000000000	0,0000000000	
58	Empresa 1	0,0000000000	0,0000046388	0,0000003981	-0,0000008474	0,0000003632	
59	Empresa 2	0,0000000000	0,0000003981	0,0000010989	-0,0000005476	0,0000003581	
91	Empresa 34	0,0000000000	-0,0000008474	-0,0000005476	0,0000035358	0,0000005187	
92	Empresa 35	0,0000000000	0,0000003632	0,0000003581	0,0000005187	0,0000029896	
93	Rend. Medio	0,0000000%	0,0224652%	-0,0033103%	0,0001732%	0,0233363%	

3.1.4. Hoja “Web IBEX”.

En esta hoja se ha establecido una conexión a la página Web que publica los datos de cotizaciones de los activos del índice IBEX con un intervalo de 10 minutos (BME, 2015), aproximadamente, entre dato y dato. Además, se da un retraso en torno a 20 minutos entre la hora de cotización del dato publicado y la hora real en el que se dispone de él en la Web. Pese a esas deficiencias, se ha optado por establecer esa conexión por tratarse de un índice conocido por la práctica totalidad de los estudiantes del Máster en CC. AA. Y FF.

Para establecer la conexión se ha utilizado la herramienta de Excel de la pestaña de Datos: “Obtener datos externos Web” explicada en Charte Ojeda (2010, pp. 334-338). Tecleada la dirección de la página Web del IBEX35 en la ventana “Nueva consulta web”, se configuran las “Opciones...” para que se importe con formato HTML completo. Las propiedades de conexión se fijan para que se actualice cada 10 minutos en segundo plano, sin actualizar automáticamente al abrir el fichero. No tiene sentido sobrescribir sobre la conexión, ya que al actualizarse se reemplazan los cambios realizados por los datos de origen.

Imagen 5. Aspecto Hoja “Web IBEX”.

	D	E	F	J	K	L	M
13	Mercado		Indice				
16	IBEX 35®						
17	Nombre	Anterior	Último	Fecha	Hora	% Dif.	
18						Año 2015	
19	IBEX 35®	11.734,60	11.749,30	10/04/2015	17:38	14,3	
20	Nombre	Últ.	% Dif.	Efectivo (miles €)	Fecha	Hora	
21	ABENGOA B	3,37	0,66	103.834,96	10/04/2015	Cierre	
22	ABERTIS SE.A	17,47	0,49	90.244,71	10/04/2015	Cierre	
54	TEC.REUNIDAS	40,68	0,69	10.682,53	10/04/2015	Cierre	
55	TELEFONICA	13,185	0,5	787.327,13	10/04/2015	Cierre	
56	Precios expresados en euros						

3.1.5. Hojas “Empresa#”.

Estas hojas se crearon inicialmente con la cabecera de la tabla de datos de cotizaciones (nombre de la empresa, precio, día y hora de cotización) para las 35 empresas. Mediante un procedimiento de VBA se va copiando periódicamente la información de la hoja “Web IBEX” a cada una de las 35 hojas. Esto requiere que la disposición de la información de la Web en su correspondiente hoja no cambie, pues el programa en VBA da por supuesto que los datos a copiar se encuentran siempre en las mismas celdas. Esta hipótesis no es exagerada cuando se está procediendo a obtener

carteras óptimas con breves intervalos de tiempo, pero transcurridas unas semanas es muy posible que la distribución de los datos de cotización en la Web difiera de la existente al inicio. Los datos que aparecen en las distintas imágenes se han obtenido mediante una actualización de la página Web cada 10 minutos y la ejecución del procedimiento de VBA para su captura en estas hojas también cada 10 minutos.

Imagen 6. Aspecto Hoja "Empresa#".

	A	B	C	D
1	ABENGOA B			
2	Precio	Día	Hora	
3	3,370	10/04/2015	Cierre	
4	3,367	10/04/2015	17:29	
5	3,367	10/04/2015	17:20	

3.1.6. Hoja "Resultados".

La hoja de resultados se ha incluido en el proyecto como una manera de comunicar al usuario las diversas carteras óptimas que el proceso de automatización va a generar. En una versión más depurada de la aplicación, se trataría de comunicar al usuario mediante formularios, cuadros de diálogo u otro tipo de mensajes, qué órdenes de compra o de venta debería realizar para realizar la transición de una cartera óptima a la nueva cartera según la evolución de las cotizaciones de los activos. También servirá esa información para analizar su evolución en el tiempo y comparar los resultados con otro tipo de estrategias.

Después de determinar la composición óptima, el procedimiento copiará en una nueva columna B insertada en la hoja "Resultados" lo siguiente: el rango de celdas "Participación", el rendimiento ("rent_cartera") y la varianza ("var_cartera") de la Hoja "Principal", además de la fecha y hora de la hoja de "Datos del modelo". Véase **Imagen 7**.

Imagen 7. Aspecto Hoja "Resultados".

	A	B	C	D	E	F	G	H	I	J
1	Fecha	07/04/2015	07/04/2015	07/04/2015	07/04/2015	07/04/2015	07/04/2015	07/04/2015	07/04/2015	
2	Hora	11:35:00	11:25:00	11:15:00	10:55:00	10:45:00	10:35:00	9:50:00	9:25:00	
3	V*	1,172E-05	1,134E-05	1,132E-05	1,088E-05	1,082E-05	0,063205	0,0631995	1969420,4	
4	R*	0,0006654	0,0005793	0,0005863	0,0005531	0,0005702	0,0316552	0,0317385	196,51003	
5	Objetivo	-2,155E-05	-1,763E-05	-1,799E-05	-1,678E-05	-1,769E-05	0,0616222	0,0616125	1969410,6	
6	Efectivo	0,000%	0,000%	0,000%	12,913%	12,913%	0,000%	0,000%	0,000%	
7	Empresa1	0,000%	0,001%	0,001%	0,205%	0,205%	20,000%	20,000%	20,000%	
8	Empresa2	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	
9	Empresa3	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	
10	Empresa4	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	

3.2. Resolución de la cartera óptima.

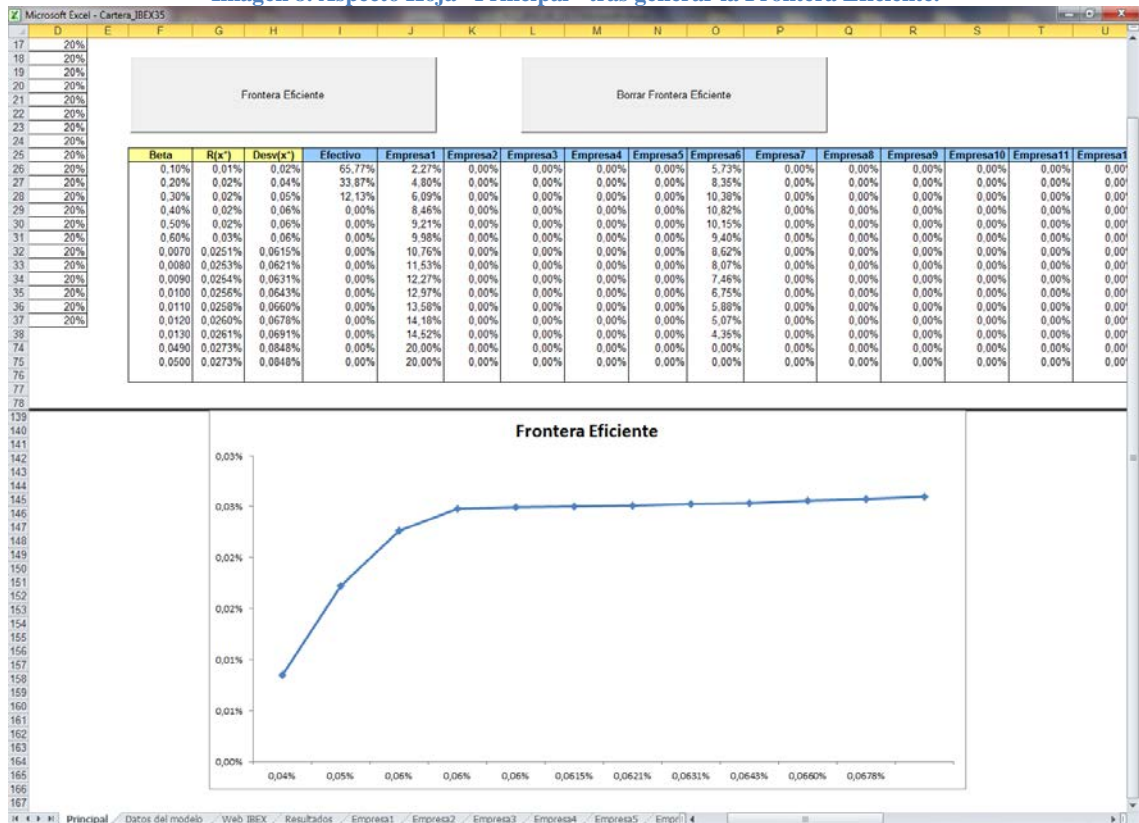
Se ha programado un procedimiento en VBA (Véase Anexo A, Código VBA 1) para la resolución del problema que pasa por reiniciar Solver y añadir cada vez todos los elementos. Si no se hace así y el código de VBA se limita a ejecutar Solver con las instrucciones mínimas que proporciona el grabador de macros, se corre el riesgo de que la solución de Solver tome los elementos que por defecto estén almacenados, pudiendo cambiar el planteamiento que inicialmente se formuló. El procedimiento está pensado para ejecutar Solver con datos nuevos de forma repetida como son los casos de la generación de la “Frontera Eficiente” y el del proceso automático de gestión de cartera. Se activa una vez haciendo clic en el botón de “Cartera Óptima” de la hoja “Principal”, y será llamado a ejecutarse varias veces al hacer clic en el botón de “Frontera Eficiente”, resolviendo el problema para distintos valores del coeficiente β de preferencia por el rendimiento; o haciendo clic en el botón de “Cartera Óptima automatizada”, resolviéndose con datos de cotizaciones actualizados, según la frecuencia establecida.

Se han modificado las Opciones de Solver por defecto, ampliando la precisión, el tiempo y número de iteraciones en la resolución, así como la selección del escalado automático.

3.3. Generación de la Frontera eficiente

Para generar el gráfico de la Frontera eficiente se pregunta al usuario el número total de puntos a representar, lo que sirve de base para calcular valores equidistantes del coeficiente β . Se ha considerado en el ejemplo del código VBA que el valor máximo para este coeficiente es de 0,05, pero como muchos otros parámetros de la aplicación se podría cambiar. Para cada uno de ellos se resuelve el problema de cartera óptima obteniendo la composición de la misma, el rendimiento y el riesgo. El procedimiento de VBA (Véase Anexo B, Código VBA 2) irá copiando los resultados de cada problema de gestión de cartera en un rango de celdas. Esta tabla de datos la utiliza para generar un gráfico que represente la “Frontera Eficiente” debajo de dicha tabla. Véase la **Imagen 8**.

Imagen 8. Aspecto Hoja "Principal" tras generar la Frontera Eficiente.



Por último, se ha creado un código VBA (Véase Anexo B, Código VBA 3) para eliminar tanto los datos de la tabla que ha generado la Frontera Eficiente como el gráfico y se ha vinculado al botón “Borrar Frontera Eficiente”.

3.4. Importación de datos Web y resolución de la cartera óptima automatizada

A partir de la hoja “Web IBEX” se establece una conexión con BME (2015), mediante el procedimiento `Gestion_Cartera_Automatizada()`, que permite capturar automáticamente los datos de cotización de los títulos del IBEX (Véase Anexo C, Código VBA 4). El primer bloque de instrucciones ordena la ejecución de una rutina, `CopiarDatos()`, transcurridos 10 minutos desde el momento actual (Véase Anexo C, Código VBA 5). La automatización se efectúa gracias al comando “`Application.OnTime`” de VBA como sugiere Ycaza (2013). Hay una variable contador que controla que no se realicen más de 60 iteraciones.

El procedimiento `CopiarDatos()` almacena los datos de la hoja “Web IBEX” en las hojas “`Empresa#`” insertando una primera fila con los datos de la nueva cotización.

Una vez copiados los nuevos datos en “Empresa#”, con la rutina `QuitarDuplicados()` se depuran duplicidades (Véase Anexo C, Código VBA 7). En la hoja “Datos del modelo”, los rendimientos medios y sus varianzas-covarianzas se calculan automáticamente a partir de la copia de los últimos 51 registros de cotizaciones sobre sí mismos con un retardo temporal de un período. El dato recién actualizado ocupa la primera fila en la columna correspondiente. Después, el procedimiento hace una llamada a la rutina de `CarteraOptima()` devolviendo los datos en la Hoja Principal. Para evitar que los resultados se pierdan en cada actualización al ser sustituidos por la nueva solución, se ha creado otra rutina, `Copia_Resultados()`, cuya ejecución copia los resultados en la hoja “Resultados” (Véase Anexo C, Código VBA 6).

En la versión presentada en estas páginas, se repite la actualización cada 10 minutos hasta un máximo de 60 iteraciones, pero estos parámetros se podrían modificar.

Pudiera ser que los últimos datos de cotización de la jornada se repitan, generando rendimientos nulos, pero por eso se ha creado un procedimiento auxiliar para eliminar los valores duplicados que pudieran capturarse por algún tipo de incidencia (Véase Anexo C, Código VBA 7).

Se podía implementar un mecanismo de tiempo para la finalización de la gestión automatizada, pero se ha optado por añadir un procedimiento de interrupción voluntaria, `Stop_Automatizar()`, que permita detener el programa de gestión automática de cartera antes de su finalización (Véase Anexo C, Código VBA 8).

4. CONCLUSIONES

Se ha creado una aplicación que puede adaptarse a un OA para ayudar en el proceso de enseñanza-aprendizaje, tanto en una asignatura de Programación en VBA, como en otra que trate la Gestión de Carteras.

La gestión de Solver mediante VBA, la importación automática de datos y la generación automática de la Frontera Eficiente proporcionan a la aplicación unas características diferenciales dinámicas respecto a otras aplicaciones existentes, que se limitan a usar las herramientas de Excel de forma estática, sin automatizar.

El desarrollo futuro de la aplicación va orientado a crear un MA y a servir de apoyo para la elaboración de su POL sugiriendo cambios o mejoras respecto a:

- Planteamiento del modelo y los supuestos en los que se basa: considerar otro tipo de activos en la cartera (fondos de inversión, renta fija, materias primas,...); existencia de costes de transacción; nuevas restricciones sobre los activos (lotes, integridad del número de activos,...); permitir posiciones a corto ($x_j < 0$); existencia de arbitraje; planteamientos matemáticos alternativos; otros modelos teóricos de gestión de cartera; métodos alternativos en la determinación de los rendimientos teniendo en cuenta, entre otras cuestiones, el distinto período de generación; rentabilidades alternativas del activo sin riesgo; selección de valores de beta; etc.

- Análisis ex-post de la evolución de la composición de la cartera y de su rentabilidad en comparación con la rentabilidad del mercado o de otras carteras conocidas; análisis de los resultados según la frecuencia en la determinación de la composición óptima (minutos, horas, días, semanas, meses, etc.);

- Mejoras en el interface de la aplicación: control de errores, almacenamiento de datos, comunicación con el usuario (órdenes de compra/venta).

5. REFERENCIAS BIBLIOGRÁFICAS

- ASIC-ICE. (2008). “Plan de Acciones para la Convergencia Europea. Los objetos de aprendizaje como recurso para la docencia universitaria: criterios para su elaboración”. Vicerrectorado de Estudios y Convergencia Europea. Vicerrectorado de Tecnologías de la Información y de las Comunicaciones. Universidad Politécnica de Valencia. Valencia: Universidad Politécnica de Valencia. Consultado el 11 de abril de 2015, de http://www.aqu.cat/doc/doc_22391979_1.pdf
- BME. (10 de abril de 2015). “Índices. BME Renta Variable. Índices IBEX. Precios sesión”. Bolsas y Mercados Españoles. Consultado en 2015, de: <http://www.bmerv.es/esp/asp/Mercados/Precios.aspx?indice=ESI100000000&punto=indice>
- CHARTE OJEDA, F. (2010). *Guía Práctica: Excel 2010*. Anaya Multimedia, Madrid.

- FOGUÉS ZORNOZA, P., y JIMÉNEZ FERNÁNDEZ, E. (2012). “Modelo de selección de cartera con Solver”. *Modelling in Science Education and Learning*, 5(4), pp. 57-63.
- GROUPE CONSULTATIF ACTUARIEL EUROPEEN. (2011). “Core Syllabus for Actuarial Training in Europe”. Consultado el 10 de abril de 2015, de Programa básico de estudios para la formación actuarial en Europa. Formación. Instituto de Actuarios Españoles.: http://www.actuarios.org/espa/web-nueva/formacion/escuela/SYLLABUS_2014.pdf
- INSTITUTO DE ACTUARIOS ESPAÑOLES. (2015). “Formación”. Consultado el 10 de abril de 2015, de Instituto de Actuarios Españoles. Colegio Profesional: <http://www.actuarios.org/espa/web-nueva/formacion/formacion.htm>
- JIMENO MORENO, J. P. (2004). *Los mercados financieros y sus matemáticas. Una guía teórica y práctica para comprender las matemáticas de los mercados*. Barcelona: Ariel.
- MARKOWITZ, H. M. (1952). “Portfolio Selection”. *The Journal of Finance*, 7(1), pp. 77-91.
- MARKOWITZ, H. M. (1959). *Portfolio Selection: Efficient Diversification of Investment*. Cowles Foundation Monograph 16, Wiley, Yale University Press, New Haven (CT).
- MARKOWITZ, M. H. (1987). *Mean-variance Analysis in Portfolio Choice and Capital Markets*. Basil Blackwell, Oxford.
- MORENO BONILLA, F. y CANO RODRÍGUEZ, M. (2010). *Excel 2010. Modelos económicos y financieros*. Anaya Multimedia, Madrid.
- PARADA GANDOS, C. (2010). “Definición de objeto de aprendizaje (oa)”. Consulta 11 de abril de 2015. E-colección. Colección de metodologías y recursos para el e-learning: <https://ecoleccion.wordpress.com/2010/09/02/definicion-de-objeto-de-aprendizaje-oa/>

- PIÑOL ESPASA, J. A. (2013). “Diseño de carteras de opciones con Excel”. *Rect@*, 14, pp. 169-180.
- RODRÍGUEZ BERNAL, Y. (2014). “Aprendizaje Orientado a la Resolución de Problemas en Programación Lineal, Solver en MS-Excel”. IX Conferencia Latinoamericana de Objetos y Tecnologías de Aprendizaje (pp. 707-710). LACLO, Manizales (Colombia).
- VELÁZQUEZ A., C., MUÑOZ A., J., ÁLVAREZ R., F., y ARÉVALO M., C. (2005). “La importancia de la Definición de la Calidad del Contenido de un Objeto de Aprendizaje”. VI Encuentro Internacional de Computación ENC'2005 (pp. 329-333). Soc. Mexicana de Ciencias de la Computación, Mexico- D. F.
- VENTURA MARCO, M. (2013a). “Una experiencia de Project Oriented Learning en finanzas y seguros mediante un proyecto VBA Excel”. III Jornadas de Intercambio de Experiencias de Innovación Educativa en Finanzas dentro del ámbito del Espacio Europeo de Educación Superior. Tirant lo Blanch, Valencia.
- VENTURA MARCO, M. (2013b). “Informática en CAF: de la programación al diseño de aplicaciones pasando por la modelización”. *Anales de ASEPUMA*, 21, 533, 20 páginas.
- VENTURA MARCO, M., MENEU GAYA, R., & PÉREZ-SALAMERO GONZÁLEZ, J. M. (2000). *Modelización y resolución de problemas de optimización en Economía*. Valencia: Repro-Exprés, S. L.
- WILEY, D. A. (2000). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. (D. A. WILEY, Ed.) Consulta el 11 de abril de 2015. The Instructional Use of Learning Objects. Online Version: <http://reusability.org/read/chapters/wiley.doc>
- YCAZA, R. (2013). “Cómo ejecutar una macro periódicamente (cada cierto tiempo)”. Consulta 20 de enero de 2015. Aprende a poner a la tecnología de tu lado. Raymundo Ycaza: www.raymundoycaza.com/ejecutar-una-macro-periodicamente/

ANEXOS

A. Resolución de la cartera óptima

Código VBA 1. Cartera óptima.

```
Sub CarteraOptima()  
'Añadido para borrar datos y no añadir restricciones cada iteración  
SolverReset  
' Solucion Macro  
' Macro creada el 15/04/2002 y modificada el 12/04/2015. Manuel Ventura  
' y Juan M. Pérez-Salamero  
' Configuración Solver  
SolverOk SetCell:="Funcion_Objetivo", MaxMinVal:=2, ValueOf:="0", _  
ByChange:="Participacion", Engine:=1, EngineDesc:="GRG Nonlinear"  
SolverOptions MaxTime:=1000, Iterations:=10000, Precision:=0.0000001, _  
Convergence:=0.0001, StepThru:=False, Scaling:=True, AssumeNonNeg:=True, _  
Derivatives:=1  
SolverAdd CellRef:="presupuesto", Relation:=2, FormulaText:="=1"  
SolverAdd CellRef:="Participacion", Relation:=1, FormulaText:="u_cotas_sup"  
SolverAdd CellRef:="Participacion", Relation:=3, FormulaText:="d_cotas_inf"  
'Añadido para no mostrar mensaje  
SolverFinish  
End Sub
```

B. Frontera Eficiente

Código VBA 2. Generación Frontera Eficiente.

```
Sub FronteraEficiente()  
Dim contador As Integer  
Dim maxcontador As Integer  
Application.ScreenUpdating = False  
Application.CutCopyMode = False  
Range("ResultFrontEficiente").Clear  
maxcontador = _  
Val(InputBox("Nº de puntos para Frontera Eficiente:", LIMITE_SUPERIOR_CONTADOR))  
Range(Cells(26, 6), Cells(26 + maxcontador, 44)).Name = "ResultFrontEficiente"  
For contador = 1 To maxcontador  
'El valor máximo de beta se ha considerado 0.05  
Range("beta").Value = contador * (0.05 / maxcontador)  
CarteraOptima  
Application.SendKeys ("{Enter}")  
Cells(25 + contador, 6) = Range("beta")  
Cells(25 + contador, 7) = Range("rent_cartera")  
Cells(25 + contador, 8) = Sqr(Range("var_cartera"))  
For j = 1 To 36  
Cells(25 + contador, 8 + j).Value = Cells(1 + j, 2).Value  
Next j  
Cells(25 + contador, 6).Select  
Selection.NumberFormat = "0.0000"  
Cells(25 + contador, 7).Select  
Selection.NumberFormat = "0.0000%"  
Cells(25 + contador, 8).Select  
Selection.NumberFormat = "0.0000%"  
Range(Cells(25 + contador, 9), Cells(10 + contador, 44)).Select  
Selection.NumberFormat = "0.00%"  
Next contador  
Formato  
Application.ScreenUpdating = True  
For Each chobj In ActiveSheet.ChartObjects  
chobj.Delete  
Next chobj  
Range(Cells(26, 7), Cells(26 + maxcontador, 7)).Select  
Set frontEfChart = ActiveSheet.Shapes.AddChart(xlLineMarkers, 350, 550 + _  
12 * maxcontador, 700, 350).Chart  
frontEfChart.SetSourceData Source:=Range(Cells(26, 7), Cells(26 + maxcontador, 7))  
frontEfChart.SeriesCollection(1).XValues = Range(Cells(26, 8), Cells(26 +
```

```
maxcontador, 8))
frontEfChart.SetElement (msoElementChartTitleAboveChart)
frontEfChart.ChartTitle.Text = "Frontera Eficiente"
frontEfChart.Axes(xlValue).MajorGridlines.Select
Selection.Delete
frontEfChart.Legend.Select
Selection.Delete
ActiveSheet.ChartObjects.Select
End Sub
```

Código VBA 3. Borrar Frontera Eficiente.

```
Private Sub CmdBorrar_Click()
For Each chobj In ActiveSheet.ChartObjects
    chobj.Delete
Next chobj
Range("ResultFrontEficiente").Select
Selection.Clear
Range("b4").Select
End Sub
```

C. Importación de datos Web y resolución de la cartera óptima automatizada

Código VBA 4. Declaración de variables del módulo. Gestión de cartera automatizada.

```
Dim contador As Integer
Dim Tiempo As Variant
Dim ENCENDIDO As Boolean
'Módulo AUTOMATIZACIÓN
Public Sub Gestion_Cartera_Automatizada()
' Código inspirado en Raymundo Ycaza en
' www.raymundoycaza.com/ejecutar-una-macro-periodicamente/
' Consultado el 20-1-2015.

' Programa para generar cada cuánto debe ejecutar una macro Excel
' se ha puesto cada 10 minutos que ejecute la copia de datos de
' cotizaciones del IBEX en las correspondientes hojas.
' Se realizan hasta 60 capturas distantes 10 minutos
If contador <= 60 Then
    Tiempo = Now + TimeValue("00:10:00")
    Application.OnTime Tiempo, "CopiarDatos", , True
    ENCENDIDO = True
Else
    ENCENDIDO = False
    Exit Sub
End If
End Sub
```

Código VBA 5. Copiar Datos.

```
Public Sub Copiardatos()
Dim i As Integer
'Declaración de matriz que almacena datos de precio, día y hora para las 35
'empresas del IBEX
Dim datos(3, 35)
Sheets("Web IBEX").Select
contador = contador + 1
'Asignación datos a matriz
For i = 1 To 35
    datos(1, i) = Cells(20 + i, 5).Value
    datos(2, i) = Cells(20 + i, 11).Value
    datos(3, i) = Cells(20 + i, 12).Value
Next i
'Copia datos a hojas
For i = 1 To 35
    Sheets("Empresa" & i).Select
    Rows("3:3").Select
    Selection.Insert Shift:=xlUp
    For j = 1 To 3
```

```

        Cells(3, j).Value = datos(j, i)
        Cells(3, j).Value = datos(j, i)
        Cells(3, j).Value = datos(j, i)
    Next j
Next i
'Elimina datos duplicados
Call QuitarDuplicados
'Copia a "Datos del modelo"
'Asignación fecha de la cartera
Sheets("Datos del Modelo").Select
For j = 53 To 3 Step -1
    Cells(j, 1).Value = Cells(j - 1, 1).Value
Next j
Cells(2, 1) = Sheets("Empresal").Range("B3").Value
'Asignación hora de la cartera
Sheets("Datos del Modelo").Select
For j = 53 To 3 Step -1
    Cells(j, 2).Value = Cells(j - 1, 2).Value
Next j
'Hora más tardía
Range("C3").Select
ActiveCell.FormulaR1C1 = "=MAX('Web IBEX'!R[19]C[9]:R[53]C[9])"
Cells(2, 2) = Range("C3").Value
Range("C3").Clear
'Asignación de cotizaciones
Sheets("Datos del Modelo").Select
For i = 1 To 35
    For j = 53 To 3 Step -1
        Cells(j, i + 3).Value = Cells(j - 1, i + 3).Value
    Next j
    Cells(2, i + 3) = Sheets("Empresa" & i).Range("A3").Value
Next i
Call CarteraOptima
Call Copia_Resultados
Call Gestion_Cartera_Automatizada
End Sub

```

Código VBA 6. Copia Resultados.

```

Sub Copia_Resultados()
    Sheets("Resultados").Select
    Columns("B:B").Select
    Selection.Insert Shift:=xlToRight, CopyOrigin:=xlFormatFromLeftOrAbove
    Sheets("Principal").Select
    Range("J3").Select
    Selection.Copy
    Sheets("Resultados").Select
    Range("B5").Select
    Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
        :=False, Transpose:=False
    Sheets("Principal").Select
    Range("J5").Select
    Application.CutCopyMode = False
    Selection.Copy
    Sheets("Resultados").Select
    Range("B4").Select
    Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
        :=False, Transpose:=False
    Sheets("Principal").Select
    Range("J6").Select
    Application.CutCopyMode = False
    Selection.Copy
    Sheets("Resultados").Select
    Range("B3").Select
    Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
        :=False, Transpose:=False
    Sheets("Datos del modelo").Select
    Range("A2").Select
    Application.CutCopyMode = False
    Selection.Copy
    Sheets("Resultados").Select

```

```
Range("B1").Select
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
:=False, Transpose:=False
Application.CutCopyMode = False
Selection.NumberFormat = "m/d/yyyy"
Sheets("Datos del modelo").Select
Range("B2").Select
Selection.Copy
Sheets("Resultados").Select
Range("B2").Select
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
:=False, Transpose:=False
Application.CutCopyMode = False
Selection.NumberFormat = "[$-F400]h:mm:ss AM/PM"
Sheets("Principal").Select
Range("B2:B37").Select
Selection.Copy
Sheets("Resultados").Select
Range("B6").Select
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _
:=False, Transpose:=False
Columns("B:B").Select
Selection.Font.Bold = False
Range("B6").Select
Range(Selection, Selection.End(xlDown)).Select
Selection.Style = "Percent"
Selection.NumberFormat = "0.000%"
End Sub
```

Código VBA 7. Depurar registros duplicados.

```
Sub QuitarDuplicados()
Dim i As Integer
For i = 1 To 35
    Range("Tabla_Empresa" & i).RemoveDuplicates Columns:=Array(1, 2, 3), _
        Header:=xlYes
Next i
End Sub
```

Código VBA 8. Detener la Gestión de Cartera Óptima Automatizada.

```
Sub Stop_Automatizar()
If ENCENDIDO = True Then
    Application.OnTime Tiempo, "CopiarDatos", , False
    ENCENDIDO = False
Else
    MsgBox "No está en marcha Cartera Óptima Automatizada"
End If
End Sub
```