

Estrategia de Pairs Trading con Filtro de Kalman

Nicolás Ramírez Castro

Maximiliano Hernández Aparicio

1. Introducción

Este reporte busca dar a entender el desarrollo e implementación de una estrategia de Pairs Trading aplicada a los activos **DAL** (Delta Air Lines) y **UAL** (United Airlines). Esta estrategia es utilizada en el ámbito de la gestión de portafolios y trading cuantitativo, y se basa en identificar dos activos cuyo comportamiento histórico muestra un alto grado de comovimiento o cointegración.

La hipótesis fundamental detrás de Pairs Trading es que, ante desviaciones de la relación histórica entre los dos activos, existe una oportunidad de arbitraje: cuando el spread se aleja de su media histórica, el trader puede abrir una posición de compra (long) en el activo infravalorado y una posición de venta (short) en el activo sobrevalorado, con la expectativa de que el spread regresará a su media.

En este proyecto, además de utilizar las pruebas tradicionales de cointegración (Engle-Granger y Johansen), se implementó un Filtro de Kalman para estimar el hedge ratio dinámico y ajustar la estrategia de cobertura en tiempo real. Finalmente, se diseñó un backtesting para evaluar el desempeño histórico de la estrategia, calculando métricas clave como el Sharpe Ratio, el Sortino Ratio y el Max Drawdown.

2. Desglose de los Pasos

2.1. Descarga y Normalización de Datos (Descarga_Activos.py)

- Se descargaron los datos históricos de precios de cierre ajustados para **DAL** y **UAL** desde Yahoo Finance.
- Se aplicó una normalización para facilitar la comparación visual de los activos.

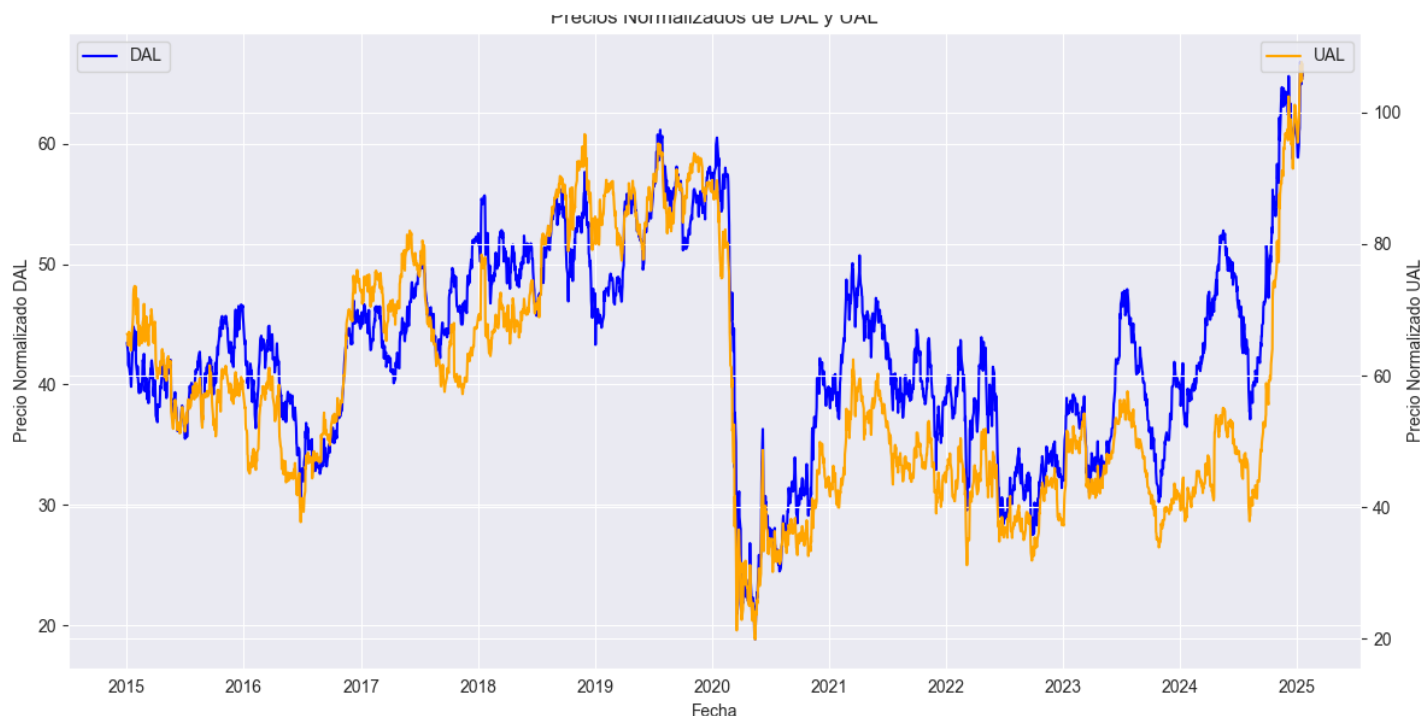


Figura 2.1 Grafico de precios de activos.

2.2 Pruebas de Estacionaridad (Prueba_DickeyFuller.py)

- Se realizó la Prueba de Dickey-Fuller Aumentada (ADF) a cada serie individual, con el objetivo de verificar si son series no estacionarias, una condición clave para evaluar cointegración.

```
Prueba ADF para DAL:
- Estadístico ADF: -2.3903
- p-valor: 0.1445
- Umbrales críticos: {'1%': np.float64(-3.4329507078222634), '5%': np.float64(-2.8626889823128554), '10%': np.float64(-2.567381647203466)}

P-value es mayor a 0.05, por lo que la serie DAL NO es estacionaria.

Prueba ADF para UAL:
- Estadístico ADF: -1.3776
- p-valor: 0.5929
- Umbrales críticos: {'1%': np.float64(-3.4329424595562963), '5%': np.float64(-2.862685339846278), '10%': np.float64(-2.5673797079109892)}

P-value es mayor a 0.05, por lo que la serie UAL NO es estacionaria.

Ambas series no son estacionarias. Son aptas para una estrategia de pairs trading.
```

Figura 2.2 Resultados de pruebas ADF para cada activo.

2.3 Pruebas de Cointegración (Prueba_EngleGranger, Prueba_Johanssen.py)

- **Engle-Granger:** Se realizó una regresión OLS entre ambos activos y se evaluaron los residuos mediante ADF para verificar si son estacionarios.
- **Johansen:** Se utilizó el método multivariado de Johansen para identificar vectores de cointegración y obtener un **modelo de spread**.

OLS Regression Results						
=====						
Dep. Variable:	DAL	R-squared:	0.773			
Model:	OLS	Adj. R-squared:	0.773			
Method:	Least Squares	F-statistic:	8586.			
Date:	Fri, 28 Feb 2025	Prob (F-statistic):	0.00			
Time:	19:42:50	Log-Likelihood:	-7089.1			
No. Observations:	2527	AIC:	1.418e+04			
Df Residuals:	2525	BIC:	1.419e+04			
Df Model:	1					
Covariance Type:	nonrobust					
=====						
	coef	std err	t	P> t	[0.025	0.975]

const	18.5917	0.273	68.084	0.000	18.056	19.127
UAL	0.4146	0.004	92.661	0.000	0.406	0.423
=====						
Omnibus:	47.830	Durbin-Watson:	0.018			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	36.413			
Skew:	0.203	Prob(JB):	1.24e-08			
Kurtosis:	2.575	Cond. No.	209.			
=====						

Figura 2.3 Resultados de la regresión OLS

En base a los resultados obtenidos, si es posible continuar con la prueba Johansen. Se observó que el valor de **Rcuadrado** si es mayor a 0.7, que la prueba **Omnibus** es igual a 0, la prueba **JB** es menor que 0, por lo que si cumple con los requerimientos para proseguir como Trade de pares.

```

El Hedge Ratio es 0.4146 UAL por 1 de DAL

Prueba de Estacionariedad (ADF) sobre los residuos:
- ADF Statistic: -3.5961
- P-value: 0.0058
- Critical vals: {'1%': np.float64(-3.4329424595562963), '5%': np.float64(-2.862685339846278), '10%': np.float64(-2.5673797079109892)}

P-value es menor a 0.05 por lo que los residuos son estacionarios y se CONFIRMA la relación de cointegración.

```

Figura 2.3 Resultado de prueba ADF sobre residuales de la regresión OLS

En este caso en base a los resultados de la prueba ADF sobre los residuales de la regresión OLS, se confirma la cointegración de estos dos activos por lo que la continuidad del modelo puede seguir. De igual manera obtuvimos el Hedge Ratio con los resultados de la regresión OLS, sin embargo, este no será el que usemos, mas adelante usaremos el Filtro de Kalman para obtener nuestro Hedge Ratio. Despues de obtener estos resultados pasamos a la prueba de Johansen.

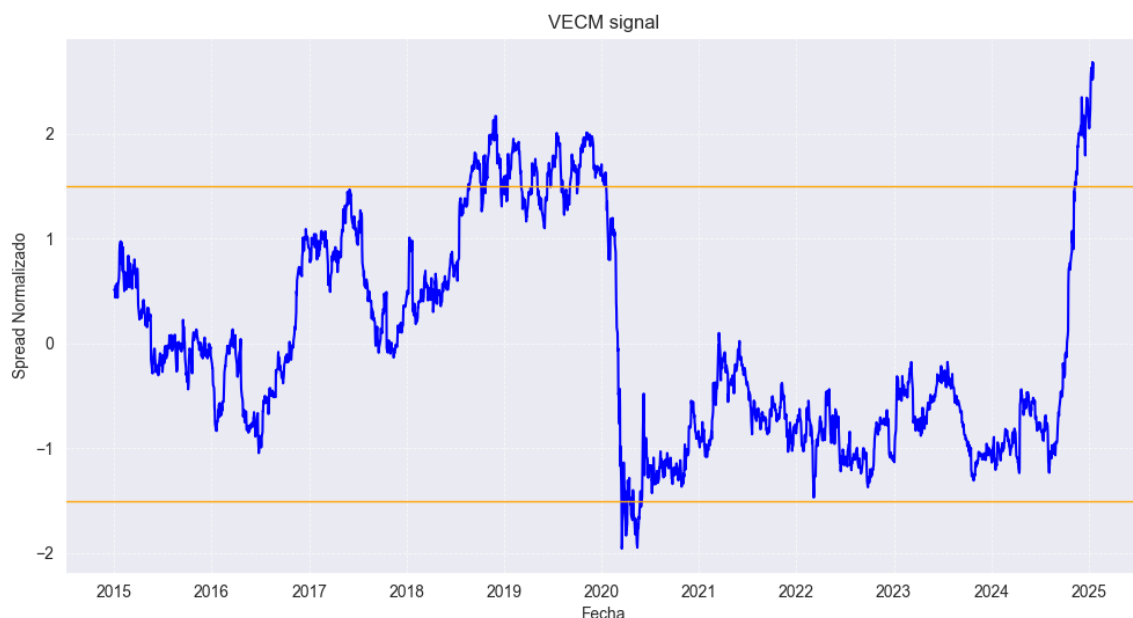
```
Eigenvalue Statistics: [15.23371309  1.72844104]
Trace Statistics: [13.50527204  1.72844104]
Eigenvalues: [0.00533434 0.0006843 ]
Eigenvectors:
[[ 0.24782256 -0.03662185]
 [-0.09437834  0.07092133]]

Nuestro modelo de spread es:
u_t = 0.24782 * UAL -0.09438 * DAL
```

Figura 2.3 Resultado de Prueba Johansen

Una vez teniendo los resultados de la prueba Johansen, podemos armar nuestro modelo de spread usando los Eigenvectores, y se hace la ecuación que se muestra como u_t .

Este sería el spread modelo normalizado que se generó usando el hedge ratio de la regresión OLS.



2.4 Estimación Dinámica del Hedge Ratio

- Se implementó un **Filtro de Kalman**, lo que permitió estimar un **hedge ratio dinámico**, adaptándose a posibles cambios en la relación entre los activos a lo largo del tiempo.
- Se graficó la evolución temporal del hedge ratio estimado.

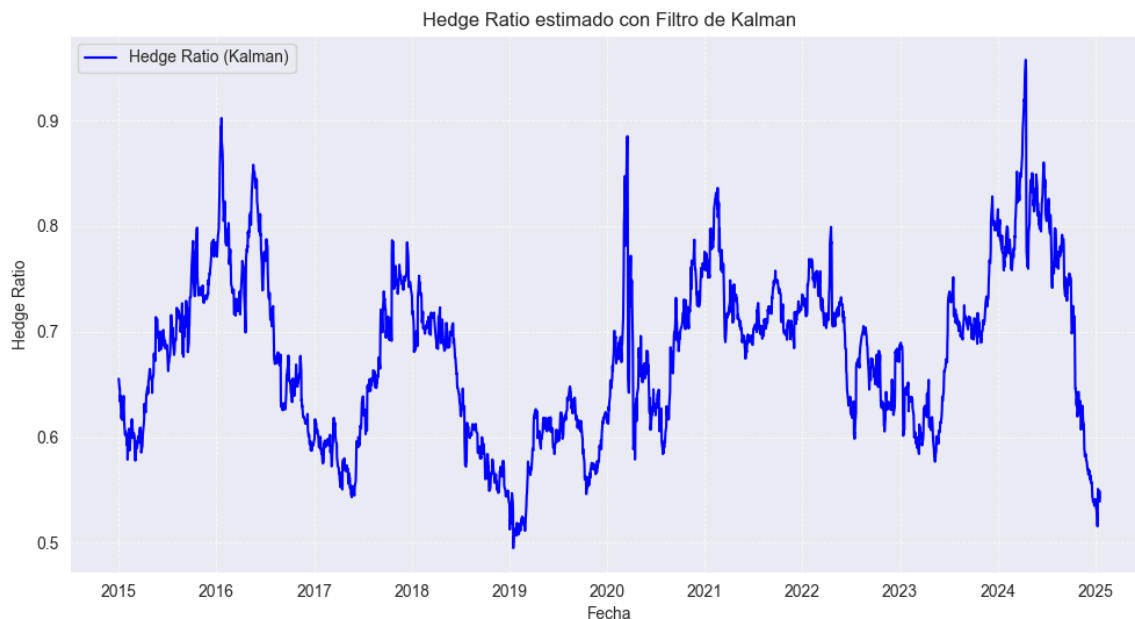


Figura 2.4 Hedge Ratio Dinámico con Filtro de Kalman

En este paso, se usó el filtro de Kalman para calcular y poder ajustar la cobertura en tiempo real dándonos una ventaja sobre los métodos de regresión OLS que suponen una relación fija entre los activos.

2.5 Generación de Señales de Trading

- Con el **spread normalizado** obtenido, se definieron **reglas de trading** basadas en umbrales de **1.5 desviaciones estándar**.
- **Reglas de trading:**
 - Cuando el spread **supera +1.5 sigma**: **Vender (short)** el activo sobrevalorado y **comprar (long)** el infravalorado.
 - Cuando el spread **cae por debajo de -1.5 sigma**: **Comprar (long)** el activo sobrevalorado y **vender (short)** el infravalorado.

- Se generaron **gráficos** que muestran las señales de compra/venta sobre los **precios de los activos** y sobre el **spread**.

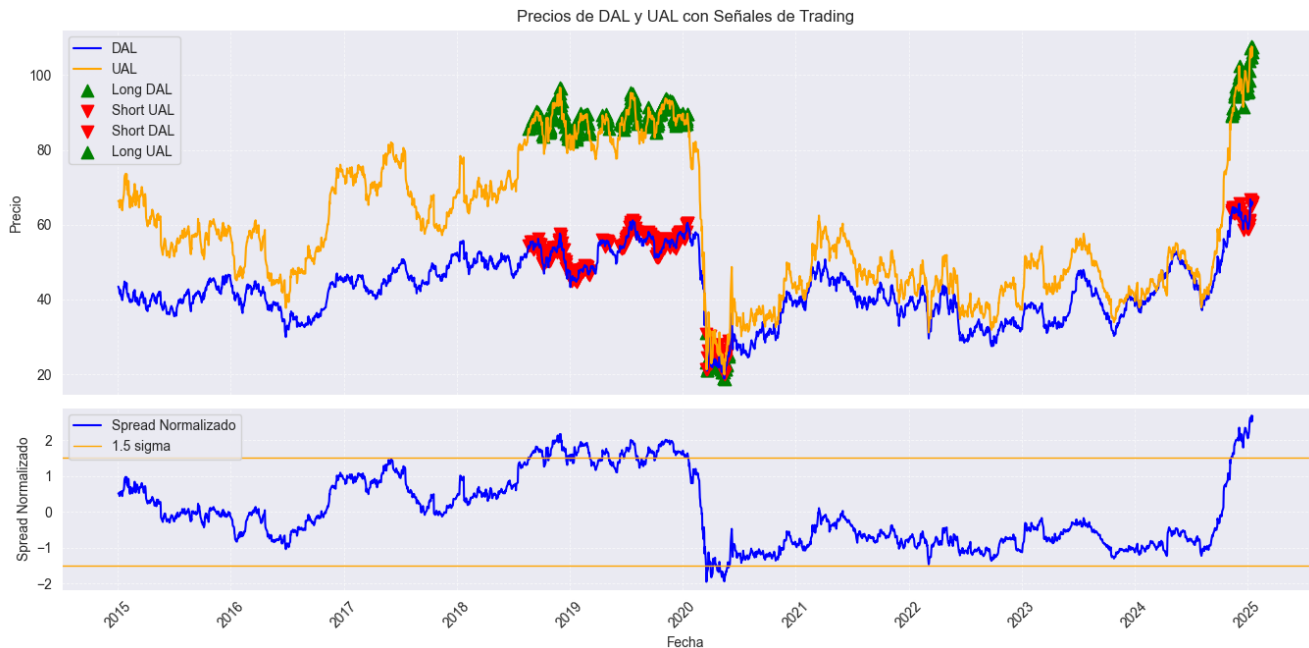


Figura 2.5 Señales de Trading en los activos

2.6 Backtesting de la Estrategia

- Se simuló la evolución de un portafolio inicial de **\$1,000,000**.
- Se incorporaron **costos de transacción de 0.125%** por cada operación.
- Se calculó la evolución histórica del capital (**equity curve**).
- Monto por trade: **\$6,000**
- Si al hacer el trade el capital baja de **\$250,000** no se ejecuta el trade

2.6.1 Descripción de la Funcionalidad del código

La función `ejecutar_backtesting` realiza el backtesting de la estrategia, evaluando cada operación basada en señales de trading y verificando la disponibilidad de capital. Se inicia con un capital inicial y se ejecutan trades según las reglas definidas.

2.6.2 Parámetros de Entrada

- **df_trades:** DataFrame con las operaciones sugeridas por la estrategia.
- **tickers:** Lista de dos activos involucrados en el trading de pares.
- **spread_norm:** Spread normalizado utilizado para definir las señales de trading.
- **hedge_ratios:** Relación de cobertura entre los dos activos.
- **capital_inicial:** Capital con el que se inicia el backtesting.
- **comision:** Costo por transacción.
- **monto_trade:** Monto fijo utilizado en cada operación.
- **capital_minimo:** Capital mínimo necesario para abrir nuevas posiciones.

2.6.3 Estructura del DataFrame df_backtesting

Se inicializa el DataFrame df_backtesting copiando df_trades y añadiendo varias columnas para registrar el estado del portafolio:

- **Capital Disponible:** Estado del capital actual.
- **Posiciones Long y Short:** Cantidad de acciones compradas y vendidas.
- **Costos y Ganancias:** Gasto en compras, ingresos por ventas y comisiones.
- **Valor del Portafolio:** Suma del capital disponible, activos en cartera y ganancias/pérdidas latentes.

2.6.4 Lógica del Backtesting

El backtesting recorre cada fila del DataFrame y ejecuta operaciones basadas en la columna Sigma, que representa la señal de trading:

Ejecución de Operaciones

1. **Short Trade (Sigma = +1.5):**
 - Se compra el activo tickers[0] (long) y se vende tickers[1] (short).
 - Se calcula la cantidad de acciones a comprar y vender basado en monto_trade.
 - Se verifica que el capital sea suficiente antes de abrir la posición.
 - Se descuentan costos y comisiones, y se actualiza el capital restante.

2. Long Trade (Sigma = -1.5):

- Se vende tickers[0] (short) y se compra tickers[1] (long).
- Se calculan las cantidades de acciones y costos de operación.
- Se actualizan las posiciones y el capital disponible.

3. Cierre de Posiciones (Sigma = 0):

- Se venden todas las posiciones en largo.
- Se liquidan todas las posiciones en corto.
- Se suman los ingresos de ventas al capital y se cierran las posiciones.

Cálculo del Valor del Portafolio

En cada iteración, se calcula el valor del portafolio como:

Valor Portafolio = Nuevo Capital + Dinero en activos Long + Ganancia/Pérdida activo short

Este valor representa el estado actual de la estrategia en cada momento del backtesting.

La manera de trabajar esta parte del código sabemos que tal vez no fue la mas adecuada por trabajar directamente sobre dataframes manualmente, pero decidimos hacerlo de esta manera porque queríamos ir verificando que los cálculos se hicieran bien en el dataframe, el punto de trabajarlo de esta manera fue para trabajar bajo la lógica de la estrategia de pares de trading, una vez conociendo y haciendo bien la lógica del backtesting, se planeaba pulir el código para dejarlo mas claro y entendible pero el tiempo no nos lo permitió.

Resultado del calculo del valor de portafolio a lo largo del tiempo



Figura 2.6 Resultado del backtesting mostrando el valor del portafolio en el tiempo

En base a la gráfica podemos darnos cuenta de que el rendimiento fue de aproximadamente 12%-13%. Para ser 10 años durante los cuales se trabajó la estrategia, el rendimiento es bajo, sin embargo, si vemos la evolución del valor del portafolio, podemos darnos cuenta de que el riesgo de perder dinero fue realmente bajo, lo máximo que llegó a perder la estrategia fue de aproximadamente \$60,000, por lo que obviamente no consideramos que sea la mejor estrategia para invertir, pues hay opciones que durante esos 10 años otorgan más rendimiento con un riesgo igual o menor. Creemos que la estrategia podría pulirse aún más si le metemos más condiciones antes de que se ejecute cualquier trade.

Conclusión del Proyecto de Pairs Trading

En este proyecto, se desarrolló y evaluó una estrategia de Pairs Trading aplicada a las acciones de Delta Air Lines (DAL) y United Airlines (UAL). La estrategia se basó en identificar una relación de cointegración entre estos activos y aprovechar oportunidades de arbitraje cuando su spread se desviaba de su media histórica.

Los resultados mostraron que la estrategia generó un rendimiento del 12.27% en un período de aproximadamente 10 años. Si bien este rendimiento no es el más alto en comparación con otras alternativas de inversión, el riesgo de pérdida fue relativamente bajo, con un drawdown máximo de \$60,000. Esto indica que la estrategia fue estable y conservadora, aunque con margen de optimización.

Mejoras Futuras

- Se identificó que el código de backtesting podría optimizarse para mejorar su claridad y eficiencia.
- Se podrían incluir más filtros o condiciones antes de ejecutar trades, reduciendo posibles operaciones ineficientes.
- El uso de un modelo de optimización del tamaño de las posiciones podría mejorar la rentabilidad sin aumentar el riesgo.
- Se podría explorar la inclusión de factores macroeconómicos o indicadores adicionales para fortalecer la toma de decisiones.

En conclusión, este proyecto permitió comprender y aplicar los principios del Pairs Trading, integrando herramientas cuantitativas avanzadas como el Filtro de Kalman y pruebas de cointegración. Aunque la estrategia mostró estabilidad y un bajo nivel de riesgo, se recomienda continuar su refinamiento para maximizar su rentabilidad en escenarios reales de trading.

