
Seminario Finanzas

Valoración IRS

Alumno

Manuel DE LA LLAVE*

Profesor

Carlos CATALÁN

7 de febrero de 2021

*manudela@ucm.es

1 Introducción

En esta práctica voy a calcular el valor presente neto (NPV) de un contrato SWAP de tipo de interés (IRS) que ya ha empezado. El contrato del IRS tiene un notional de 20 millones de USD (colateralizado también en USD), un cupón fijo de 1.2533% y el tipo de convención para contar días es un Act 360. Además, la fecha de inicio del IRS de 25 años de maduración es el 23/06/2017, siendo el 02/01/2018 la fecha de valoración y para el cupón flotante empleamos el LIBOR a 3 meses. Adicionalmente, hay que calcular una curva de forwards usando diferentes tipos de interpolación para estudiar sus efectos.

Empleo Python para realizar ambos ejercicios; los scripts y datos empleados se pueden encontrar [en este repositorio](#) para su consulta y réplica, aunque iré poniendo partes del código para explicar el ejercicio. Es la primera vez que uso Python por lo que probablemente habrá ineficiencias graves en el código, de hecho tuve bastantes problemas para trabajar con los Data Frames extraídos mediante Pandas y una parte del código está escrita para lidiar con ellos. En principio el código funciona para poder introducir cualquier fecha de valoración.

2 Valoración del IRS

Si bien en el script `main.py` están los dos ejercicios de esta práctica, he creado una clase para valorar el IRS en `irs_valuation.py`. Esta clase calcula el NPV de la pata fija y el de la pata flotante de manera simultánea, aunque arroja los resultados por separado. Primero voy a explicar el cálculo de la pata fija, luego el de la flotante y para finalizar haré el análisis global.

2.1 Pata fija

Sabemos que el NPV de la pata fija se calcula como $NPV_{fixed} = \sum_{i=0}^n C_n \Delta_i DF_i$, donde el índice i hace referencia a cada momento de pago del cupón fijo C_n . Por otro lado, Δ_i indica la fracción de días que hay entre pago y pago (número de días entre ambos dividido entre la convención; 360 en este caso) y DF_i es el factor de descuento en cada pago. Primero calculo la Δ_i :

```
aux = sum(self.valuation_date >= fixed_final_date)
delta_fixed = (fixed_final_date[aux:] - fixed_initial_date[aux:]) / self.
    act
delta_fixed[aux] = (fixed_final_date[aux] - self.valuation_date) / self.act
# From time series to float class
delta_fixed = delta_fixed.dt.seconds + delta_fixed.dt.days * (24 * 60 * 60)
delta_fixed = delta_fixed / (24 * 60 * 60)
```

Aquí la primera línea me sirve para poder cambiar la fecha de valoración, le segunda para calcular propiamente el parámetro y el resto para arreglar el problema que tuve con los pandas y que menciono en la introducción. Como consideramos que la pata fija no tiene riesgo porque está colateralizada, utilizamos para descontar flujos el tipo overnight (OIS); como no tenemos datos para todas las fechas de pagos, lo que hacemos es interpolar linealmente para obtener los descuentos necesarios:

```
df_interp = np.interp(fixed_final_date[aux:], tenor_date,
    risk_free_discount_factor)
```

Obsérvese que el `fixed_final_date` empieza más tarde porque según la fecha que elijamos para empezar a valorar es posible que hayan vencido ya algunos pagos (lo mismo haré para la pata flotante). Por último calculo el NPV empleando la fórmula expuesta al inicio del apartado:

```
fixed_npv = np.sum(self.coupon * self.notional * delta_fixed * df_interp)
print('Fixed NPV:', fixed_npv)
>>> Fixed NPV: 479194.7380326843
```

2.2 Pata flotante

En este caso, la fórmula para calcular este NPV parece muy similar a la pata fija: $NPV_{float} = \sum_{j=0}^m f_j \Delta_j DF_j$, pero tiene ciertas particularidades. Para empezar, no tenemos un cupón fijo sino que este va variando en función de la evolución del LIBOR a 3 meses. Por un lado tenemos los *fixings*, que son los tipos pasados que ya se han fijado y que usaremos para descontar ese primer pago que ya se ha fijado pero no se ha pagado; para el resto de tipos interpolaremos el factor de descuento de manera similar a como lo hicimos con el descuento de la pata fija, con la diferencia de que aquí necesitamos muchos más (la frecuencia de pago es cada tres meses, no cada año) y además usaremos el LIBOR a 3 meses, no el OIS. Una vez tengamos los descuentos podemos calcular el tipo flotante como $f_j = \left(\frac{DF_{j-1}^*}{DF_j^*} - 1 \right) / \Delta_j$, que no deja de ser el tipo forward implícito a 3 meses. Pero este factor de descuento que hemos calculado no es el que usaremos para descontar los flujos, sino que lo calcularemos de manera recursiva a través de la fórmula $DF(T_n) = \frac{1 - C_n \sum_{i=1}^{n-1} \Delta_i DF(T_i)}{1 + \Delta_n C_n}$. Esto nos garantiza un cupón fijo tal que a la hora de valorar un IRS de mercado al inicio del mismo, este valga 0, indicando que el valor esperado de los pagos de un lado y de otro son iguales, resultando así en un instrumento financieramente justo. Finalmente, la Δ_j representa lo mismo que en el caso anterior, pero esta vez la fracción de año será más fina. Aquí el código de Δ_j :

```
aux2 = sum(self.valuation_date >= float_final_date)
delta_float = (float_final_date[aux2:] - float_initial_date[aux2:]) / self.
    act
delta_float[aux2] = (float_final_date[aux2] - self.valuation_date) / self.
    act
# From time series to float class
delta_float = delta_float.dt.seconds + delta_float.dt.days * (24 * 60 * 60)
delta_float = delta_float / (24 * 60 * 60)
```

y el cálculo del tipo forward:

```
df_interp_libor = np.interp(float_final_date[aux2 - 1:], libor_date,
    libor_discount_factor)
niter = len(df_interp_libor) - 1
forward = np.zeros(niter)
for i in range(0, niter):
    forward[i] = ((df_interp_libor[i] / df_interp_libor[i+1]) - 1) /
        delta_float[aux2 + i]
index = np.zeros(aux2)
for j in range(0, aux2):
    index[j] = fixings_date[fixings_date == float_initial_date.values[j]].
        index[0]
    forward[j] = fixings_rates[index[j]]
```

como he comentado antes, primero interpolamos para sacar los factores de descuento de interés (primera línea) y los usamos para calcular el forward implícito con el primer bucle usando la fórmula mostrada anteriormente. El segundo bucle sirve para asignar los tipos ya fijados a la curva de forwards y no tener que usar los estimados mediante interpolación (los *fixings*). Algo que me llamó la atención llegado a este punto y que no había reparado hasta entonces, es que el tipo a 3 meses es muy superior al tipo que paga la pata fija, que se traduce como veremos más adelante en un pago mucho mayor. Finalmente, calculamos el factor de descuento de la pata flotante:

```
float_discount_factor = np.zeros(niter)
float_discount_factor[0] = 1 / (1 + delta_float[aux2] * self.coupon)
my_sum = np.zeros(niter - 1)
for k in range(0, niter - 1):
    my_sum[k] = delta_float[aux2 + k] * float_discount_factor[k]
```

```

float_discount_factor[k + 1] = (1 - self.coupon * sum(my_sum)) / (1 +
    delta_n[k + 1] * self.coupon)
delta_n = (float_final_date[aux2:(aux2 + k + 1)] - float_initial_date[
    aux2]) / self.act
delta_n = delta_n.dt.seconds + delta_n.dt.days * (24 * 60 * 60)
delta_n = (delta_n / (24 * 60 * 60))
# Alternative form with delta_n almost constant
# float_discount_factor[k + 1] = (1 - self.coupon * sum(my_sum)) / (1 +
    delta_float[aux2 + k] * self.coupon)

```

donde calculamos el primer factor de descuento fuera del bucle y a partir del mismo obtenemos el resto iterando. Finalmente calculo el NPV de manera análoga al anterior apartado pero teniendo en cuenta que el notional entra con el signo cambiado:

```

float_npv = sum(forward * (-self.notional) * delta_float *
    float_discount_factor)
print('Float NPV:', float_npv)
>>> Float NPV: -12132722.143793194

```

2.3 Conclusiones

Ahora tan solo queda sumar ambas cantidades, obteniendo así un NPV de -11653527.40576051 . Como estamos vendiendo el IRS, es decir, pagamos la pata fija y recibimos la flotante, esta cantidad representa lo que nosotros pagamos y el signo negativo de la misma indica que estamos recibiendo cerca de 11.5 millones de USD (en valor presente), lo que implica que podemos vender este activo por dicha cantidad en el momento justo de valoración. Sin embargo, cabe plantearse el por qué de la disparidad de las cantidades representadas aquí. Como hemos comentado por encima antes, esto tiene que ver con la diferencia de tipos que pagan ambas patas. La pata fija sólo paga 0.12533% mientras que el tipo flotante empieza en torno al 1.2% y rápidamente llega hasta el 2% convergiendo al final de la serie al 2.5%, lo que significa que la pata flotante va a estar pagando sistemáticamente mucho más que la fija. Esto puede ser debido a que cuando se calculó el cupón fijo al inicio del contrato, la expectativa de tipos de interés era mucho más baja. Ya hemos dicho que este se calcula de forma que el IRS sea justo para ambas partes, por lo que tiene sentido que al calibrar el SWAP la curva de tipos fuera mucho más baja. Para probar esto, volvemos a calcular todo suponiendo que el cupón fijo es del 2% (fijado arbitrariamente como experimento) y los resultados son mucho más “razonables”:

```

Fixed NPV: 7646927.918817272
Float NPV: -8223826.890781927
NPV IRS: -576898.9719646545

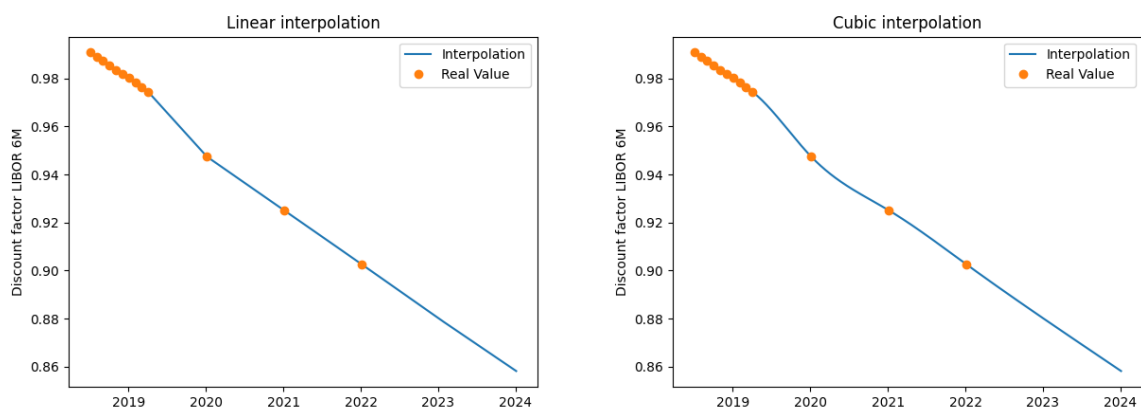
```

ahora el derivado vale cerca de medio millón, casi once millones menos que en el caso anterior. Y ahora el factor de descuento de la pata flotante también parece mucho más razonable, ya que antes del cambio estaba cercano a 1 a lo largo de todo el periodo, pero en este escenario evoluciona de manera similar al descuento de la pata fija. Si seguimos aumentando el cupón fijo al, por ejemplo, el 2.5%, el NPV resultaría positivo, teniendo así pues que pagar nosotros, lo que me indica que mi calculadora de IRS es, al menos, consistente con lo que esperaría en la teoría.

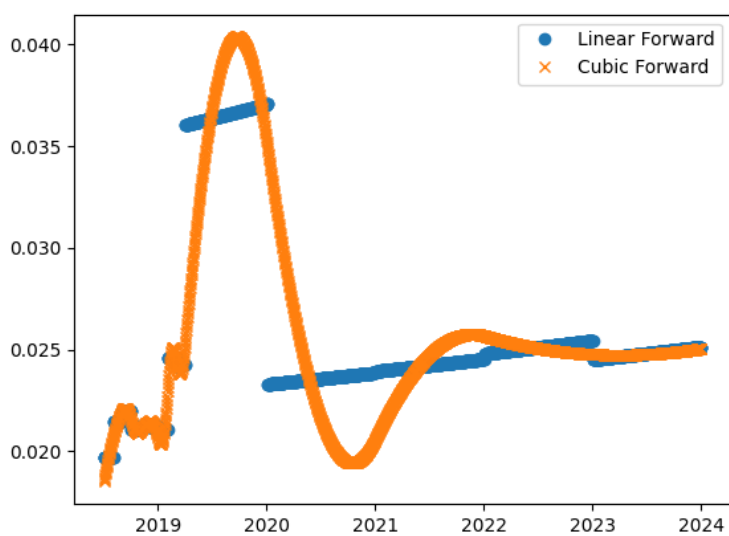
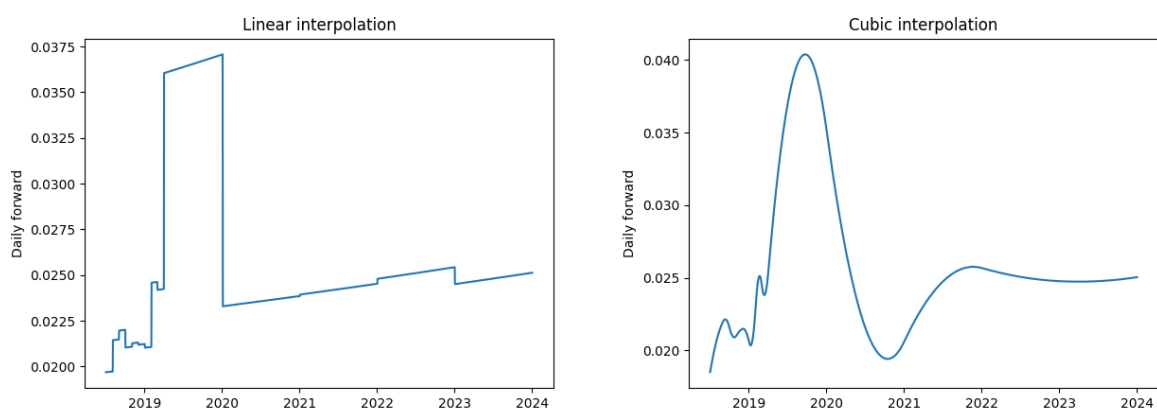
3 Cálculo de forwards

Para este ejercicio he escogido trabajar, sin ninguna razón en particular, con la curva del LIBOR a 6 meses. El cálculo del forward está hecho de la misma manera que en el ejercicio anterior, con la diferencia de que es un cálculo diario y el tenor va mes a mes al principio y luego va dando saltos más grandes, por lo que esperaríamos valores poco razonables. He empleado

la interpolación lineal que ofrece **numpy** y la cúbica unidimensional de la librería **scipy** y los resultados gráficos para el factor de descuento son los siguientes:



con el factor de descuento he calculado los forwards:



Vemos que al principio, cuando los tenores son menores, los valores son más razonables, pero cuanto más dilatada es la fecha de un tenor a otro, mayores valores atípicos observamos; en especial es notable el salto que se produce cuando los tenores pasan de ser mensuales a ser anuales. Si bien la interpolación cúbica tiene un ajuste más suave (el salto que se produce en la interpolación lineal tanto hacia arriba como hacia abajo es enorme), también toma valores más extremos. Es evidente que interpolar para una granularidad tan fina (diaria) conlleva errores más grandes que si interpoláramos por ejemplo cada 3 meses, como en el ejercicio anterior.