Fixed Income Derivatives Final Exam Fall 2024(18.01.2025)

— SOLUTION —

Imagine that you are working for a major financial institution and that you have been approached by a client who has a 10 year floating rate loan on which he pays 6M Euribor semi-annually plus a spread. The client is worried that future interest rates will be high and that his future loan payments will rise beyond what is acceptable to him. For simplicity and without loss of generality, you will work as if the notional of the clients loan obligation is one Euro and all your answers will be per 1 Euro of total debt. Your task will be to present to the client three different solutions to manage his interest rate risk and weigh the pros and cons of each of the three solutions. The three different solutions you will offer are:

1) To enter into a 10Y payer interest rate swap in which the client receives 6M floating Euribor and pays a fixed rate instead. That way, the client will turn his future unknown floating rate payments into a known fixed rate.

2) To construct a 10Y interest rate cap with a strike of $K = 0.06$ to prevent that the clients future floating rate payments will exceed the strike of $K = 0.06$.

3) To buy a 3Y7Y payer swaption with a strike as close to $K = 0.06$ as possible so that if interest rates have risen at the time of exercise in 3 years, the client can enter into the underlying 7Y payer interest rate swap and convert his floating rate payments into fixed rate payments of size equal to the strike of the swaption.

In Problem 1, we will examine the first of these three options, in Problem 2 the second, in Problem 3 the third. In Problem 4, you will compare the three different ways of managing interest rate risk and assess the pros and cons of each of the three. Problem 5 is independent of the first four problems.

SOLUTION

For problems 1, 2 and 3, we will use market data that includes the 6M Euribor fixing recently announced, Euribor Forward Rate Agreements and Euribor denominated interest rate swaps constructed at the exact same time as the Euribor rate announcement. The swap rates represent the par swap rates for a range of swap agreements with different maturities in which the floating leg pays 6M Euribor and the fixed leg pays an annual fixed rate. All interest rate swaps throughout this exam pay 6M floating Euribor against a fixed rate paid annually. The Euribor rate, the FRA rates and par swap rates are all, as is usually the case, to be understood as "simple" interest rates. The data for these rates is shown in the table below.

Table 1: Euribor, FRA and Swap Market Data

EURIBOR	Fixing	FRA	Midquote	IRS	Midquote
6M	0.03772	1X7	0.04026	2Y	0.05228
		2X8	0.04261	3Y	0.05602
		3X9	0.04477	4Y	0.05755
		4X10	0.04677	5Y	0.05791
		5X11	0.04860	7Y	0.05718
		6X12	0.05029	10Y	0.05539
		7X13	0.05183	15Y	0.05324
		8X14	0.05324	20Y	0.05205
		9X15	0.05452	30Y	0.05087

In addition, we also observe market prices for caplets on 6M forward Euribor rates for a strike of $K = 0.055$ and a notional of 1 Euro. The caplet price on the future Euribor rate $L(T_{i-1}, T_i)$ announced at time T_{i-1} is denoted by π_i and the payment on the caplet contract thus occurs at time T_i. Caplet prices are given in basispoints, so to transfer to monetary units, simply divide by 10,000.

Table 2: Caplet Market Prices

T_i	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
π_i(bps.)	-	3.5920	19.2679	32.1887	37.2136	36.4750	32.2678	26.9031	21.2176	16.2022
T_i	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
π_i(bps.)	12.0628	8.8952	6.5191	4.8435	3.6485	2.8098	2.2067	1.7814	1.4707	1.2443

Finally, we also have data for the Black implied volatility of 3Y7Y payer swaptions for a range of different strikes. These swaptions give the owner the right to enter into a 7Y payer swap at a strike K exactly 3 years from now. The ATMF is simply the 3Y7Y par swap rate observed right now and the offsets, denoted K_{offset} and measured in basispoints, are relative to the 3Y7Y forward par swap rate.

Table 3: $3Y7Y$ Swaption Market Prices

$K_{offset}(bp)$	-300	-250	-200	-150	-100	-50	ATMF
$\Pi_{swaption}$	0.220675	0.183310	0.155103	0.129001	0.108120	0.084411	0.071866
$K_{offset}(bp)$	50	100	150	200	250	300	-
$\Pi_{swaption}$	0.066535	0.073942	0.082751	0.093605	0.098971	0.108909	-

At the end of this document, a Python script is presented containing the data from the tables above. You do not have to use this piece of code but it might save you some time.

SOLUTION

Problem 1

We will begin by fitting a term structure of continuously compounded zero coupon bond rates to the market data presented in Table 1 above. Please perform the fit and answer the following questions:

a) Plot the term structures of spot- and forward rates from your fit for all maturities up to 30 years and also report the 6M, 1Y, 2Y, 5Y, 10Y, 15Y, 20Y and 30Y continuously compounded spot rates from your fit. Does your fit match the data? If yes, provide evidence that this is so for example by reporting the SSE(Sum-of-Squared-Errors) of the fit.

b) Discuss which properties the term structures of spot- and forward rates should have and also discuss if your fit has those properties. Also explain which type of curve you have used when fitting the market data and why you chose that type of fit.

c) Compute 6M forward Euribor rates $L(0, T_{i-1}, T_i)$ up to $T_i = 10$ and the 10Y par swap rate. Plot the term structure of 6M forward Euribor rates and the 10Y par swap rate as a horizontal line in a separate plot. Report the 10Y par swap rate and explain how it is related to 6M forward Euribor rates?

Problem 1 - Solution

a) The fitted spot rates depend slightly on the method used to perform the fit but the fit should be near perfect. The term structures of spot- and forward rates are shown in Figure 1 below and the 6M, 1Y, 2Y, 5Y, 10Y, 15Y, 20Y and 30Y continuously compounded spot rates should roughly be

T	6M	1Y	2Y	5Y	10Y	15Y	20Y	30Y
R	0.03737	0.04352	0.05115	0.05666	0.05361	0.05102	0.04955	0.04804

The fit is close to perfect as the SSE from the swap portion of the fit can be found to be of order 10^{-25} and from the FRA portion to be of order 10^{-10}.

b) As a minimum, spot rates should be continuous and forward rates should be positive. It seems that a polynomial of degree 3 produces the most plausible fit. Choosing a higher degree polynomial makes forward rates for large maturities vary a bit much and choosing a lower degree polynomial makes the curves a bit rough. The Nelson-Siegel fit also seems a bit rough. Choosing a "smooth" transition, as was done here, implies that forward rates are differential and so certainly, spot rates are continuous as well.

c) The 10Y par swap rate is roughly $R_{10Y} = 0.05539$. The 10Y par swap rate is a weighted average of the forward rates corresponding to the floating leg. That is, the 10Y par swap rate is a weighted average of 6M forward Euribor rates.

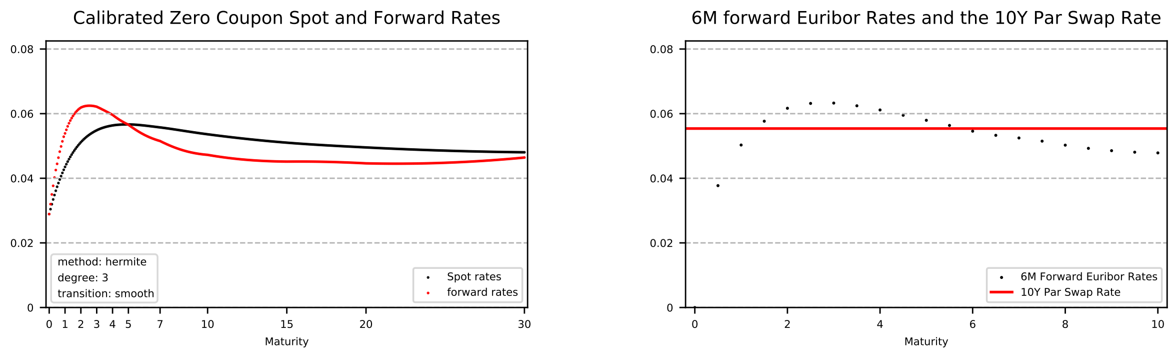

Figure 1: Calibrated Yield Curves, the 10Y Swap Rate and 6M Forward Euribor

SOLUTION

Problem 2

Now, we will focus on pricing the strike $K = 0.06$ 10Y interest rate cap that the investor considers as a tool to protect himself against interest rate increases. We will do so using two different models. First, we will use the Vasicek model in which the dynamics of the short rate r_t are given by

$$dr_t = (b - ar_t)dt + \sigma dW_t, \qquad r(0) = r_0 \tag{1}$$

where $a > 0$, $b > 0$ and σ are all constant parameters and W_t is a Brownian motion. Second, we will use the Hull-White Extended Vasicek(HWEV) model in which the dynamics of the short rate are

$$dr_t = \big(\Theta(t) - ar_t\big)dt + \sigma dW_t, \qquad r(0) = r_0 \tag{2}$$

where $a > 0$ and σ are constant parameters, $\Theta(t)$ is a deterministic function and W_t is a Brownian motion.

a) Set $\sigma = 0.02$ and use the initial values $r_0 = 0.035$, $a = 6$ and $b = 0.25$ to fit a Vasicek model to the ZCB spot rates you found in Problem 1 and answer the following questions:

   i) Report the parameters of the Vasicek model from your fit and plot the term structure of spot rates implied by the fitted parameters along with the term structure of spot rates you found in Problem 1a. Is the Vasicek model able to fit the term structure of market spot rates?

   ii) Could it be expected that a Vasicek model is able to fit this particular term structure of observed spot rates? Please explain your reasoning.

   iii) Suggest another model that might produce a better fit and explain why you expect that method to work better.

b) Now we will fit the Hull-White Extended Vasicek (HWEV) model to the prices of caplets for a strike $K = 0.055$ given in Table 2. That is, please perform the following steps:

   i) Write a function that takes as its first argument the parameters a and σ and returns the SSE between observed caplet prices from the table above and computed caplet prices as a function of a and σ.

   ii) Using an initial values for a of 2.5 and an initial value for σ of 0.018, fit the HWEV model to caplet prices using "nelder-mead" and report the fitted values $\hat{a}$ and $\hat{\sigma}$ you obtain.

   iii) Plot the term structure of caplet Black implied volatility corresponding to both market- and fitted caplet prices in the same plot and use this plot to assess if the HWEV model is able to fit observed caplet prices well.

   iv) Now that we have fitted the HWEV model to caplet prices, how do we ensure that the model also fits ZCB spot rates and do we even have to? Please explain your reasoning.

c) Next, we will simulate trajectories of the short rate in both the Vasicek model and in the HWEV model with the parameters you obtained when fitting the models. If for some reason you have been unable to fit one or both of the models, simply proceed assuming the values of the parameters that were given as the initial parameter values.

   i) Simulate the short rate in both the Vasicek and HWEV model by taking $M = 500$ steps for values of t from $t = 0$ to $t = 10$ so that the step-size is $\delta = 0.02$. Plot the two trajectories in separate plots in which you also include the mean and a 95% two-sided confidence interval.

   ii) Interpret the behavior of the short rate in these two models and relate your findings to the parameters of the models as well as the initial fit of the ZCB spot rates you obtained in 1a.

   iii) Briefly explain which of the two models you think fits the market data best and which is more realistic. Please also explain which of the two models is more likely to produce a reliable price of a 10Y caplet with a strike that is different from the one for which we have caplet price data.

SOLUTION

d) Finally, we will compute the price of the 10Y interest rate cap with a strike of $K = 0.06$ on a series of 6M Euribor payments using the model you found to be the best in problem 2c. Please explain how you compute first caplet prices and then the cap price. Also, explain which parameter values you will use. Note that the caplet with a maturity of T_i, pays the amount $\big(L(T_{i-1}, T_i) - K\big)_+$ at time T_i where $L(T_{i-1}, T_i)$ is the Euribor fixing announced 6 months prior at time T_{i-1}.

i) Compute prices of 6M Euribor caplets for maturities ranging from $T_i = 1$ to $T_i = 10$ and report these for $T_i \in [1, 2, 4, 6, 8, 10]$

ii) Find the price of a 10Y interest rate cap with a strike of $K = 0.06$ on 6M Euribor that begins as early as possible and ends in exactly $T = 10$ years. Report the price in basispoints, both as an amount paid upfront as well as spread on top of regular semi-annual interest rate payments.

Problem 2 - Solution

a) The term structure of spot rates in the Fitted Vasicek model can be found in Figure 2.

i) Fitting the Vasicek model with $\sigma = 0.02$ gives us roughly $\hat{r_0} = 0.02498$, $\hat{a} = 5.57422$ and $\hat{b} = 0.28856$. From the plot in Figure 2, we see that the Vasicek model is not able to fit the observed term structure of spot rates very well. In particular, the model is not able to capture the 'hump' in market spot rates.

ii) In the Vasicek model, the term structure of spot rates will be either strictly upward sloping or strictly downward sloping, so therefore the Vasicek model can by construction not fit the hump-shaped term structure of market spot rates. The model tries its best by choosing $\hat{a}$ to take a large value making the short rate very fast mean-reverting, but it does not work well and choosing a very high in the Vasicek model also has other unfortunate consequences.

iii) To fit the term structure of market spot rates, we have a number of models available. The CIR model would unfortunately suffer the same fate as the Vasicek model since the term structure of spot rates will also be either strictly upward or strictly downward sloping in this model. However, we could also use either the Ho-Lee model or the Hull-White Extended Vasicek model and in particular use that if $\Theta(t)$ is chosen appropriately in these models, they will by construction fit the initial term structure of interest rates perfectly.

b) The term structure of spot rates in the HWEV model can be found in Figure 2.

i) The code used to fit the HWEV model can be found in Appendix A

ii) The parameter values from the HWEV model fit become $\hat{a} = 2.03298$ and $\hat{\sigma} = 0.02031$ and the model seems to fit the market caplet prices quite well. The SSE from the fit is of order 10^{-10}.

iii) The term structures of market- and fitted caplet implied volatility are also shown in Figure 2 and they confirm that the HWEV model is able to fit market prices of 6M Euribor caplets.

iv) When fitting the HWEV model, we 'only' needed to use the parameters a and σ and not $\Theta(t)$ from the drift of the short rates in the HWEV model. This is of course because European option prices on ZCB bonds and thus caplet prices in the HWEV model are independent of $\Theta(t)$. Thus even after fitting the HWEV model to caplet prices, we are still free to choose $\Theta(t)$ such that the model fits the initial term structure of spot rates perfectly.

c) The simulalted trajectories of the Vasicek and HWEV models can be seen in Figure 3

i) The code used to simulate the short rate in the two models can be found in Appendix A.

ii) In the Vasicek model fit, we found that the parameter a governing the speed of mean-reversion from the fit was as high as $\hat{a} = 5.57422$ which results in the model mean-reverting very fast and the 95% the confidence intervals becoming very narrow settling quickly to a range from only 0.0400 to 0.0635 under the stationary distribution. Also, the trajectory of the mean in the Vasicek model is strictly increasing which is inconsistent with the hump-shaped term

SOLUTION

structure of forward rates in market data. In the HWEV model fit, $\hat{a} = 2.03298$ and the model mean-reverts at a slower and more realistic pace. Under the stationary distribution, the 95% confidence interval is roughly 0.02756 to 0.06704 which is also much more realistic. Finally, the mean trajectory of the short rate in the HWEV model is hump-shaped and consistent with the trajectory of observed forward rates.

iii) The HWEV model fits the market data much better and also results in a more realistic process for the short rate. This is hardly surprising as the HWEV model is much more flexible. Also, the HWEV model has the advantage that you can fit the yield curve using $\Theta(t)$ and then still have a and σ available to fit caplet prices. In the Vasicek model, you 'use up' all parameters to fit try to fit the initial term structure and have no free parameters left to fit caplet prices.

d) We will use the HWEV model as it proved superior to the Vasicek model with the fitted parameters $\hat{a} = 2.03298$ and $\hat{\sigma} = 0.02031$. Caplet prices can be found explicitly using Blacks formula in the HWEV model so first, we compute these for $K = 0.06$ and then we will find the price of the interest rate cap with same strike as the sum of the individual caplets.

i) Caplet prices for $T_i \in [1, 2, 4, 6, 8, 10]$ become:

Table 4: Caplet Prices for $K = 0.06$ in the HWEV Model

T_i	1	2	4	6	8	10
π_i(bps.)	0.6687	15.8546	12.8065	2.6277	0.6122	0.2263

ii) The price of the strike $K = 0.06$ interest rate cap if paid upfront becomes $\pi_{cap} = 120.37$ basispoints corresponding to a semi-annual premium of 8.0333 basispoints.

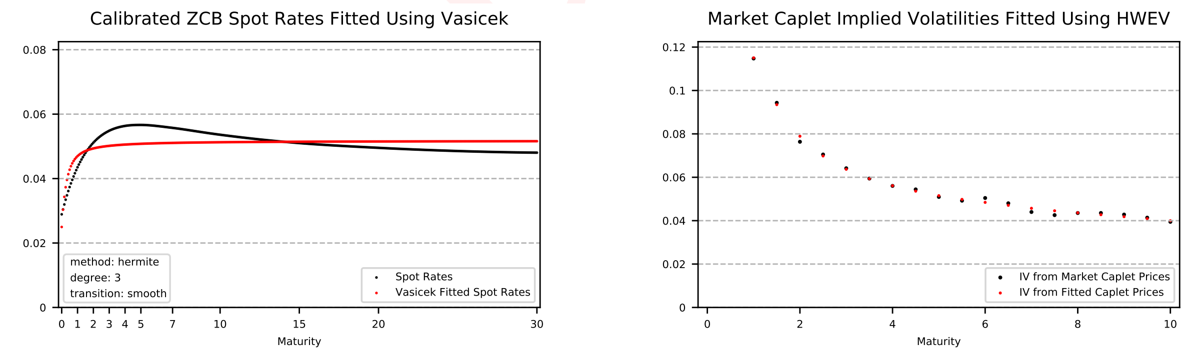

Figure 2: The Fitted Values of the Vasicek and HWEV Models

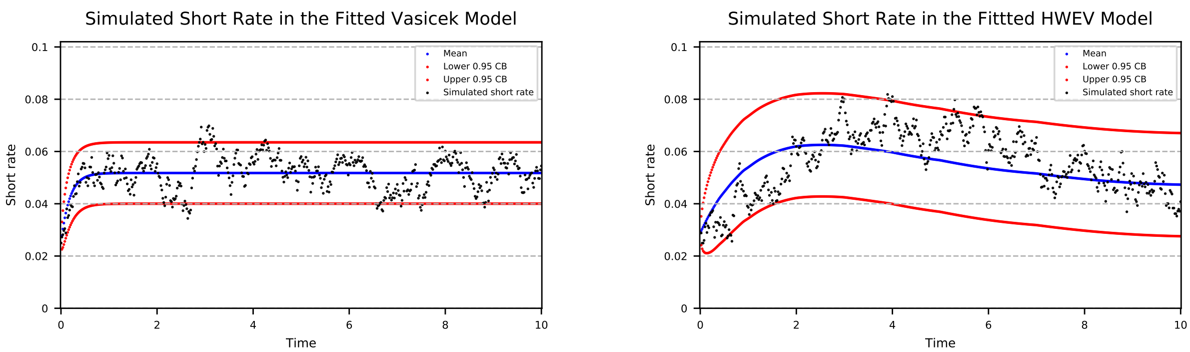

Figure 3: Simulated trajectories of the short rate Fitted Vasicek and HWEV Models

Problem 3

Next, we will focus on the 3Y7Y payer swaption that the investor also considers as a possible tool to protect himself against interest rate increases. The underlying asset of the 3Y7Y payer swaption is the 3Y7Y forward payer swap and we will model the forward 3Y7Y par swap rate F_t and its volatility σ_t using the SABR model. That is, we assume that F_t and σ_t have the following joint dynamics

$$\begin{aligned} dF_t &= \sigma_t F_t^{\beta} dW_t^{(1)}, \qquad && F(0) = F_0 \\ d\sigma_t &= \upsilon \sigma_t dW_t^{(2)}, \qquad && \sigma(0) = \sigma_0 \\ dW_t^{(1)} dW_t^{(2)} &= \rho dt && \end{aligned} \tag{3}$$

where $0 \leq \beta \leq 1$, $0 < \upsilon$ and $-1 < \rho < 1$ are constants and $W_t^{(1)}$ and $W_t^{(2)}$ are correlated Brownian motions with correlation coefficient ρ. Also recall that we have data for market implied volatilities from market prices of 3Y7Y payer swaptions in Table 3.

a) First, we examine the market implied volatilities of swaption prices and find the price of the swaption that the client considers as a tool to manage interest rate risk.

   i) Plot the market implied volatilities of the payer swaptions as a function of their strike. Interpret the plot and in particular assess if the shape of the graph is common for market implied volatilities. What does the implied volatility curve tell us about the distribution of the underlying par swap rate implied by swaption prices.

   ii) Find the ATMF forward rate corresponding to the 3Y7Y payer swaption. That is, simply find the 3Y7Y forward par swap rate.

   iii) Find the price of a the 3Y7Y payer swaption that has a strike closest to $K = 0.06$. Report the price of this swaption in basispoints as well as its exact strike. This particular swaption will be the one that the client will consider as a tool to hedge against rising interest rates.

b) Assuming we know that $\beta = 0.55$, fit the SABR model to the 3Y7Y payer swaption market implied volatilities given in Table 3 and solve the following questions:

   i) Report the fitted parameter values of σ_0, υ and ρ and plot fitted implied volatilities in the same plot as implied volatilities from observed market prices.

   ii) Do fitted implied volatilities match market implied volatilities? Report the SSE of your fit.

c) Now we will consider a so called strangle consisting of a long position in one ATMF + 100 basispoints payer swaption and a long position in one ATMF - 100 basispoints receiver swaption. We will use the parameter values from the fit of the SABR model. If you were not able to get sensible parameter values when fitting the SABR model, simply use the initial values suggested in Problem 3b.

   i) Argue that a payer swaption can be seen as a call option on the par swap rate of the underlying interest rate swap and that a receiver swaption can be seen as a put option on the par swap rate of the underlying interest rate swap.

   ii) Find the value of the strangle in the SABR model using the fitted parameter values. Bump the initial value F_0 of underlying swap rate by one basispoint up and down and report the loss/gain to the value of the strangle in both cases. Also bump the initial volatility σ_0 up and down by 0.001 and report the loss/gain in both cases.

   iii) Based on your answer to ii) assess the nature of the exposure you get from a strangle. Is the strangle very sensitive to changes in the forward par swap rate and what is the direction of the exposure? Also, is the strangle very sensitive to changes in volatility and what is the direction of the exposure? Please provide some intuition for your conclusions.

SOLUTION

Problem 3 - Solution

a) The implied volatilities corresponding to market prices of swaptions are displayed in Figure .

   i) Swaption implied volatilities have the shape of an asymmetric 'smile' or perhaps 'smirk' which is very common for observed option implied volatilities. The higher implied volatilities for deep in-the-money and deep out-of-the-money swaptions reflect that the implied distribution of the underlying par swap rate under the pricing measure chosen by the market generally has more probability mass in the tales of the distribution than that of the log-normal distribution. The asymmetry and much higher implied volatilities for small strikes reflects that the distribution under the pricing measure is not symmetric and has even more probability mass in the left tail of the distribution. Market participants are thus pricing swaptions as if the underlying par swap rate is more likely to rise sharply and much more likely to fall sharply than under the log-normal distribution.

   ii) The 3Y7Y forward par swap rate is $R_{3Y7Y} = 0.05503$.

   iii) The 3Y7Y payer swaption with a strike closest to $K = 0.06$ has a strike of 0.06004 and a price of 41.20 basispoints.

b) The implied volatility smile from the fitted SABR model is also displayed in Figure .

   i) The fitted parameter values become roughly $\hat{\sigma}_0 = 0.01788$, $\hat{\upsilon} = 0.5840$ and $\hat{\rho} - 0.3440$ and the algorithm converges very quickly.

   ii) From the plot in Figure , we see that the fitted SABR model fits market implied volatilities quite well. This is also confirmed by the SSE which was only $2.8 * 10^{-5}$.

c) The strangle will consist of a long position in the payer swaption with strike $K = 0.06503$ and a long position in a receiver swaption with a strike of $K = 0.04503$

   i) Denoting the par swap rate at time of exercise by T_n and maturity T_N by R_n^N and denoting the corresponding accrual factor by S_n^N, we get that the payoffs $\chi(T_n)$ at exercise of the payer and receiver swaptions are respectively

$$\begin{aligned} \chi_{payer}(T_n) &= S_n^N \Big(R_n^N(T_n) - K \Big)_+ \\ \chi_{receiver}(T_n) &= S_n^N \Big(K - R_n^N(T_n) \Big)_+ \end{aligned} \tag{4}$$

   A receiver swaption is thus simply a put option on the par swap rate of the underlying swap and we can find the price of the receiver swaption directly from Blacks formula for a put option.

   ii) The value of the strike $K = 0.06503$ payer swaption is approximately 16.053 basispoints and the value of the strike $K = 0.04503$ receiver swaption is approximately 31.424 basispoints. The total initial price of the strangle therefore becomes 47.477 basispoints. Increasing the par swap rate by one basispoint results in a value of the strangle of 47.485 and decreasing the par swap rate by one basispoint results in a value of the strangle of 47.481 basispoints. The value of the long position in the strangle is thus in-sensitive to changes in the underlying swap rate and we can say that this position is delta-neutral. Bumping the σ_0 by 0.001 up gives us that the value of the strangle becomes 53.605 basispoints and bumping the σ_0 by 0.001 down gives us that the value of the strangle becomes 41.688 basispoints. Hence, he value of the strangle goes up by 6.128 basispoints when σ_0 rises by 0.001 and drops by 5.789 basispoints when σ_0 falls by 0.001.

   iii) From ii), we conclude that entering into a strangle, we are not exposed to changes in the underlying par swap rate and thus not exposed to changes in interest rates. However, we are positively exposed to changes in volatility as measured by σ_0. Now, the strangle position will be profitable if either the payer swaption or the receiver swaption expires sufficiently in-the-money and that is likely to happen if the distribution of the underlying has sufficiently fat tails. This in turn will be the case if the volatility of par swap rate increases and hence entering into a strangle gives the investor positive exposure to volatility and in turn turmoil in interest rate markets. A strangle can therefore be seen as way to hedge against interest rate uncertainty.

SOLUTION

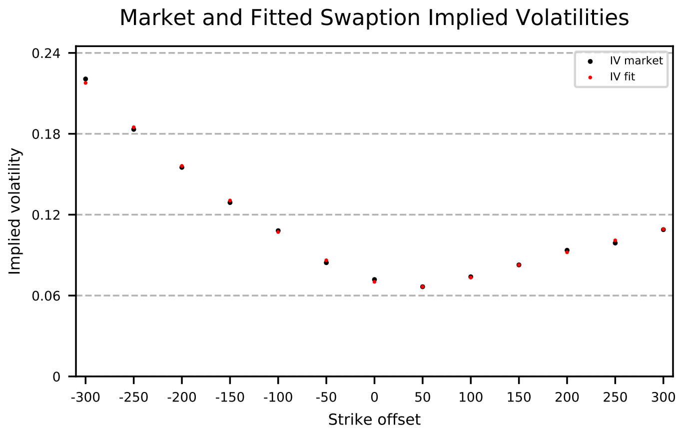

Figure 4: Market and Fitted Swaption Implied Volatilities

SOLUTION

Problem 4

Finally, in this problem we will compare the three options the client is offered to manage the risk of his 10Y floating rate obligation on which he must pay 6M Euribor semi-annually.

a) We will begin by reiterating the results we have for the three different options offered to the client.

   i) Option 1 was to enter into a 10Y payer swap immediately. Report the 10Y par swap rate.

   ii) Option 2 was to buy a 10Y interest rate cap with a strike of $K = 0.06$. Report the price of the interest rate cap both in terms of an upfront payment and in terms of a spread paid on top of the clients regular semi-annual payments.

   iii) Option 3 was to buy a 3Y7Y payer swaption with a strike as close to $K = 0.06$ as possible. Report the price and strike of the 3Y7Y payer swaption.

b) Imagine that the client will choose exactly one of the three options 1, 2 or 3 and consider the total payments implied by each of the three options. Please describe which future evolution of interest rates that would *minimize* the total payments of the client and how low these payments would they be in this scenario. Also, please describe which future evolution of interest rates that would *maximize* the total payments of the client and how high these payments would they be in this scenario. Please do so for

   i) The 10Y interest rate swap

   ii) The 10Y interest rate cap

   iii) The 3Y7Y payer swaption

c) Explain which of the three options you deem the most risky and which do you deem the safest. In particular, explain which of the three options that gives the client exposure to the worst outcome and outline this outcome.

d) The 10Y interest rate swap costs nothing upfront and eliminates all future interest rate uncertainty. Options 2 and 3, on the other hand, involve an upfront payment, possibly spread out over the future. Yet, these options do not eliminate all uncertainty about future interest rate payments. Explain why, nonetheless, options 2 and 3 are valid alternatives and that it is fair that these options come at a cost.

SOLUTION

Problem 4 - Solution

a) Looking at the clients three options to manage interest rate risk gave us the following result:

   i) The 10Y par swap rate is 0.05539.

   ii) The upfront price of the strike $K = 0.06$ interest rate cap is 120.37 bps. corresponding to a semi-annual spread of 8.0333 bps.

   iii) The payer swaption with a strike closest to $K = 0.06$ has a strike of 0.06003 and a price of 41.20 bps.

b) The worst and best scenarios for the clients three options will be described below.

   i) If the client chooses to enter into the 10Y interest rate swap, his future interest rate payments are all predetermined and independent of how interest rates evolve.

   ii) If the client chooses the interest rate cap with a strike of $K = 0.06$, the best possible scenario occurs if interest rates fall sharply and as mush as possible. The client will have to pay for the interest rate cap, likely in the form of a spread applied along his regular interest rate payments, but in principle, his semi-annual interest payments could become arbitrarily small. The worst scenario involves future Euribor rates increasing to be permanently higher than 0.06 for the next 10 years in which case the clients interest rate payments would 0.06 plus the spread.

iii) The future scenario that would minimize the clients future interest payments is one in which interest rates fall and stay low throughout the next 10 years. Now, this implies that the swaption will not be exercised in the most favorable scenario. If interest rates were high initially, such that the payer swaption is exercised, and only to fall beyond 3 years, the interest payments of the client would effectively be fixed at the level of the swaption strike and he would not benefit from interest rates falling after the exercise date. The future scenario in which the clients interest payments are the highest is one in which interest rates are high initially but not high enough for the swaption to be in-the-money at time of exercise of the swaption. The worst scenario also involves interest rates rising sharply after the time of exercise of the swaption such that the clients floating rate payments beyond three years become very high. In this scenario, the total interest payments of the client could in principle become arbitrarily large.

c) The most risky option is for the client to buy the 3Y7Y swaption since, if he chooses this option, he is not protected before 3 years and also, there is a chance that interest rates rise after the time of exercise of the swaption. This scenario could be catastrophic and he could end up paying a very high interest rate on his obligation after the time of exercise. The safest option is to enter into the a 10Y swap immediately and fix his interest rate for the next 10 years.

d) Entering into the 10Y interest rate swap and eliminating all uncertainty does not come with an upfront cost but this strategy also does not allow for any upside. Choosing either the interest rate cap or the 3Y7Y payer, the client will benefit from declining future interest rates and the potential upside is substantial in the sense that future interest rates could fall drastically.

SOLUTION

Problem 5

We will now consider a model in which the short rate at time t denoted r_t is driven by two factors X_t and Y_t. The dynamics of the two factors are given by

$$\begin{aligned} dX_t &= -\gamma X_t dt + \phi dW_t^{(1)}, & X_0 &= x_0, \\ dY_t &= \big(b - aY_t\big)dt + \sigma dW_t^{(2)}, & Y_0 &= y_0, \end{aligned} \tag{5}$$

where $\gamma > 0$, $\phi \in \mathbb{R}$, $a > 0$, $b > 0$ and $\sigma \in \mathbb{R}$ are known constants, the initial values x_0 and y_0 are known, and $W_t^{(1)}$ and $W_t^{(2)}$ are independent Brownian motions. The short rate for all $t \geq 0$ is the sum of the two factors.

$$r_t = X_t + Y_t \text{ for } t > 0 \quad \text{ and } \quad r_0 = x_0 + y_0 \text{ for } t = 0 \tag{6}$$

a) Consider the function

$$f(t, X, Y) = e^{-\gamma(T-t)}X + e^{-a(T-t)}Y \tag{7}$$

for $T > 0$ fixed where X and Y have dynamics as given in (5). Use the function $f(t, X, Y)$ to find a solution for $r_T|x_0, y_0$. Alternatively, carefully solve the SDE's for X_t and Y_t individually to find expressions for $X_T|x_0$ and $Y_T|y_0$ and argue why you can use these to find the solution for $r_T|x_0, y_0$.

b) Show that the distribution of r_T given x_0 and y_0 is Gaussian, $r_T|x_0, y_0 \sim N\big(M, V\big)$, with mean $M = M(T; x_0, y_0)$ and variance $V = V(T)$ where

$$\begin{aligned} M(T; x_0, y_0) &= x_0 e^{-\gamma T} + y_0 e^{-aT} + \frac{b}{a}\Big[1 - e^{-aT}\Big] \\ V(T) &= \frac{\phi^2}{2\gamma}\Big[1 - e^{-2\gamma T}\Big] + \frac{\sigma^2}{2a}\Big[1 - e^{-2aT}\Big] \end{aligned} \tag{8}$$

and also give the stationary distribution of r_∞.

c) Now assume that $x_0 = 0$, $\gamma = 32$, $\phi = 0.03$, $y_0 = 0.03$, $a = 0.5$, $b = 0.025$, $\sigma = 0.015$ and simulate one trajectory of each of X_t, Y_t and r_t using $N = 100$ steps from time $t = 0$ to $t = 1$ such that the step-size is $\delta = 0.01$ and plot the three trajectories along with the mean of r_t and a 95 per cent two-sided confidence interval for r_t. Also, simulate one trajectory of each of X_t, Y_t and r_t using $N = 1000$ steps from time $t = 0$ to $t = 10$ such that the step-size is still $\delta = 0.01$. In a separate plot show these three trajectories of X_t, Y_t and r_t along with the mean of r_t and a 95 per cent two-sided confidence interval for r_t. Report the two-sided upper- and lower confidence bounds for both $T = 1$ and $T = 10$.

d) Please briefly explain the role that X_t and Y_t play in the overall dynamics of the short rate r_t based on the parameter-values of the processes of X_t and Y_t as well as the plots of their simulated trajectories. In doing so, please consider the following questions:

   i) Which of the two factors mean-revert the fastest and how is your answer related to the relevant parameter(s) from the dynamics of X_t and Y_t in (5)

   ii) Explain how the fluctuations in X_t and Y_t influence the fluctuations in r_t differently and relate your answer to i) above.

   iii) Determine roughly when r_T settles to its stationary distribution and relate your answer to your answer to i) and ii) above.

Problem 5 - Solution

a) Applying Ito's formula to $f(t, X, Y)$ gives us that

$$\begin{aligned} df(t, X_t, Y_t) &= d\big(e^{-\gamma(T-t)}X_t + e^{-a(T-t)}Y_t\big) \\ &= \gamma e^{-\gamma(T-t)}X_t dt + e^{-\gamma(T-t)}\big[-\gamma X_t dt + \phi dW_t^{(1)}\big] + ae^{-a(T-t)}Y_t dt + e^{-a(T-t)}\big[\big(b - aY_t\big)dt + \sigma dW_t^{(2)}\big] \\ &= be^{-a(T-t)}dt + \phi e^{-\gamma(T-t)}dW_t^{(1)} + \sigma e^{-a(T-t)}dW_t^{(2)} \end{aligned} \tag{9}$$

SOLUTION

Integrating from $t = 0$ to $t = T$ gives us that

$$
\begin{aligned}
\int_0^T df(s, X_s, Y_s) &= X_T + Y_T - e^{-\gamma T} x_0 - e^{-aT} y_0 \\
&= \int_0^T b e^{-a(T-s)} ds + \int_0^T \phi e^{-\gamma(T-s)} dW_s^{(1)} + \int_0^T \sigma e^{-a(T-s)} dW_s^{(2)} \qquad \Rightarrow \\
r_T = X_T + Y_T &= e^{-\gamma T} x_0 + e^{-aT} y_0 + \frac{b}{a}\big[1 - e^{-aT}\big] + \phi \int_0^T e^{-\gamma(T-s)} dW_s^{(1)} + \sigma \int_0^T e^{-a(T-s)} dW_s^{(2)}
\end{aligned} \tag{10}
$$

Alternatively, we can solve for $X_T|x_0$ and $Y_T|y_0$ separately and use that r_t is simply the sum of X_t and Y_t. Here, we also need to observe that the SDE for X_t is independent of Y_t and the SDE for Y_t is independent of X_t. Therefore, we can conclude that the solution for $r_t|r_0$ is simply the sum of the solutions for $X_t|x_0$ and $Y_t|y_0$. Solving for $X_T|x_0$ can be done by applying Ito to $f(t, X) = e^{-\gamma(T-t)} X$ and integrating from $t = 0$ to $t = T$. Solving for $Y_T|y_0$ can be done by applying Ito to $f(t, Y) = e^{-a(T-t)} Y$ and integrating from $t = 0$ to $t = T$. We get that

$$
\begin{aligned}
X_T &= e^{-\gamma T} x_t + \phi \int_0^T e^{-\gamma(T-s)} dW_s^{(1)}, \\
Y_T &= e^{-aT} y_t + \frac{b}{a}\big[1 - e^{-aT}\big] + \sigma \int_0^T e^{-a(T-s)} dW_s^{(2)} \qquad \Rightarrow \\
r_T|x_0, y_0 = X_T + Y_T &= e^{-\gamma T} x_t + e^{-aT} y_t + \frac{b}{a}\big[1 - e^{-aT}\big] + \phi \int_0^T e^{-\gamma(T-s)} dW_s^{(1)} + \sigma \int_0^T e^{-a(T-s)} dW_s^{(2)} \\
&= e^{-\gamma T} x_t + e^{-aT} y_t + \frac{b}{a}\big[1 - e^{-aT}\big] + I_1 + I_2
\end{aligned} \tag{11}
$$

b) Since the ito integrals in (10), $I_1 =$ and I_2, have deterministic integrands, they are both $\mathcal{F}_T$ adapted Gaussian random variables and have mean 0. Also, since the two Brownian motions $W_t^{(1)}$ and $W_t^{(2)}$ are independent, these two integrals are independent implying that the variance of $r_T|x_0, y_0$ is the sum of the variances of I_1 and I_2. The variances of I_1 and I_2 can be computed using Ito isometry

$$
\begin{aligned}
\mathbb{E}\big[r_T|x_0, y_0\big] &= e^{-\gamma T} x_t + e^{-aT} y_t + \frac{b}{a}\big[1 - e^{-aT}\big] \\
\mathrm{Var}\big[X_T|x_0, y_0\big] &= \mathrm{E}\Big[\Big(\phi \int_0^T e^{-\gamma(T-s)} dW_s^{(1)}\Big)^2 \Big| x_0, y_0\Big] = \phi^2 \int_0^T e^{-2\gamma(T-s)} ds = \phi^2 \Big[\frac{1}{2\gamma} e^{-2\gamma(T-s)}\Big]_0^T = \frac{\phi^2}{2\gamma}\big[1 - e^{-2\gamma T}\big] \\
\mathrm{Var}\big[Y_T|x_0, y_0\big] &= \mathrm{E}\Big[\Big(\sigma \int_0^T e^{-a(T-s)} dW_s^{(2)}\Big)^2 \Big| x_0, y_0\Big] = \sigma^2 \int_0^T e^{-2a(T-s)} ds = \sigma^2 \Big[\frac{1}{2a} e^{-2a(T-s)}\Big]_0^T = \frac{\sigma^2}{2a}\big[1 - e^{-2aT}\big]
\end{aligned} \tag{12}
$$

So we have that

$$
\begin{aligned}
r_T|x_0, y_0 \;&\sim\; N\big(M, V\big) \\
M \;&=\; M(T; x_0, y_0) = x_0 e^{-\gamma T} + y_0 e^{-aT} + \frac{b}{a}\Big[1 - e^{-aT}\Big] \\
V \;&=\; V(T) = \frac{\phi^2}{2\gamma}\Big[1 - e^{-2\gamma T}\Big] + \frac{\sigma^2}{2a}\Big[1 - e^{-2aT}\Big]
\end{aligned} \tag{13}
$$

Sending $T \nearrow \infty$ gives us that the stationary distribution of r_t is

$$
r_\infty \sim N\Big(\frac{b}{a}, \frac{\phi^2}{2\gamma} + \frac{\sigma^2}{2a}\Big) \tag{14}
$$

c) Plots of the simulated trajectories of X_t, Y_t and r_t are shown in Figure 5. The 5% two-sided confidence interval of r_1 is $\big[0.0133667, 0.0623720\big]$ and the 95% two-sided confidence interval of r_{10} is $\big[0.0195616, 0.0801689\big]$.

d) The roles of X_t and Y_t in the dynamics of r_t can be deduced from the plots in Figure 5.

   i) It is quite clear that X_t mean reverts beck to it's mean of 0 much faster then Y_t mean reverts back to it's mean $\frac{b}{a}\big[1 - e^{-at}\big]$. This is because γ relative to ϕ^2 governing the mean reversion of X_t is much larger than a relative to σ^2 governing the mean reversion of Y_t.

SOLUTION

ii) The value of r_t is mostly determined by the slow mean reverting factor Y_t and hence, the long-term fluctuations of r_t are largely driven by Y_t. The fast mean-reverting factor X_t on the other hand generates short-term fluctuations in r_t that vanish relatively quickly.

iii) The rate of conversion of r_t is almost entirely determined by the slow mean-reverting factor and from the plots it seems that the distribution of r_t settles to its stationary distribution roughly after 8 years.

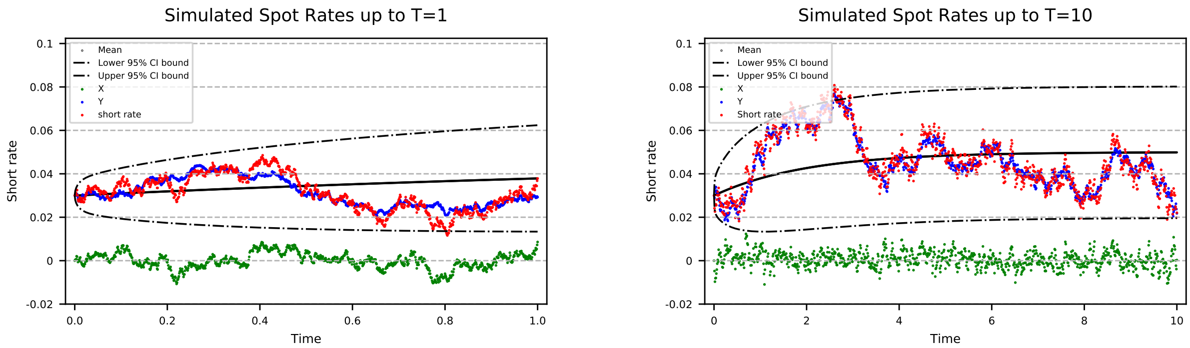

Figure 5: Simulated Trajectories of the Factors and the Short Rate

SOLUTIO

A Python Code

```
import numpy as np
from scipy.stats import norm
from scipy.optimize import minimize
import fixed_income_derivatives_E2024 as fid
import matplotlib.pyplot as plt

alpha_caplet = 0.5
N_caplet = 21
T_caplet = np.array([i*alpha_caplet for i in range(0,N_caplet)])
strike_caplet_market = 0.055
price_caplet_market = np.array([0, 0, 3.592, 19.2679, 32.1887, 37.2136, 36.475, 32.2678, 26.9031, 21.2176, 16.2022, 12.0628,
8.8952, 6.5191, 4.8435, 3.6485, 2.8098, 2.2067, 1.7814, 1.4707, 1.2443])
price_caplet_market = price_caplet_market/10000

K_swaption_offset = np.array([-300,-250,-200,-150,-100,-50,0,50,100,150,200,250,300])
iv_swaption_market = np.array([0.220675, 0.18331, 0.155103, 0.129001, 0.10812, 0.084411, 0.071866, 0.066535, 0.073942, 0.082751, 0.093605, 0.098971, 0.108909])

EURIBOR_fixing = [{"id": 0,"instrument": "libor","maturity": 1/2, "rate":0.03772}]
fra_market = [{"id": 1,"instrument": "fra","exercise": 1/12,"maturity": 7/12, "rate": 0.04026},
{"id": 2,"instrument": "fra","exercise": 2/12,"maturity": 8/12, "rate": 0.04261},
{"id": 3,"instrument": "fra","exercise": 3/12,"maturity": 9/12, "rate": 0.04477},
{"id": 4,"instrument": "fra","exercise": 4/12,"maturity": 10/12, "rate": 0.04677},
{"id": 5,"instrument": "fra","exercise": 5/12,"maturity": 11/12, "rate": 0.0486},
{"id": 6,"instrument": "fra","exercise": 6/12,"maturity": 12/12, "rate": 0.05029},
{"id": 7,"instrument": "fra","exercise": 7/12,"maturity": 13/12, "rate": 0.05183},
{"id": 8,"instrument": "fra","exercise": 8/12,"maturity": 14/12, "rate": 0.05324},
{"id": 9,"instrument": "fra","exercise": 9/12,"maturity": 15/12, "rate": 0.05452}]
swap_market = [{"id": 10,"instrument": "swap","maturity": 2, "rate": 0.05228, "float_freq": "semiannual", "fixed_freq": "annual","indices": []},
{"id": 11,"instrument": "swap","maturity": 3, "rate": 0.05602, "float_freq": "semiannual", "fixed_freq": "annual","indices": []},
{"id": 12,"instrument": "swap","maturity": 4, "rate": 0.05755, "float_freq": "semiannual", "fixed_freq": "annual","indices": []},
{"id": 13,"instrument": "swap","maturity": 5, "rate": 0.05791, "float_freq": "semiannual", "fixed_freq": "annual","indices": []},
{"id": 14,"instrument": "swap","maturity": 7, "rate": 0.05718, "float_freq": "semiannual", "fixed_freq": "annual","indices": []},
{"id": 15,"instrument": "swap","maturity": 10, "rate": 0.05539, "float_freq": "semiannual", "fixed_freq": "annual","indices": []},
{"id": 16,"instrument": "swap","maturity": 15, "rate": 0.05324, "float_freq": "semiannual", "fixed_freq": "annual","indices": []},
{"id": 17,"instrument": "swap","maturity": 20, "rate": 0.05205, "float_freq": "semiannual", "fixed_freq": "annual","indices": []},
{"id": 18,"instrument": "swap","maturity": 30, "rate": 0.05087, "float_freq": "semiannual", "fixed_freq": "annual","indices": []}]
data_zcb = EURIBOR_fixing + fra_market + swap_market

# Problem 1 - Computing the 10Y swap par swap rate
# 1a) Fitting the yield curve
mesh = 1/12
M = 360
interpolation_options = {"method":"hermite","degree":3,"transition": "smooth"}
T_fit, R_fit = fid.zcb_curve_fit(data_zcb,interpolation_options = interpolation_options)
T_inter = np.array([i*mesh for i in range(0,M+1)])
p_inter, R_inter, f_inter, T_inter = fid.zcb_curve_interpolate(T_inter,T_fit,R_fit,interpolation_options = interpolation_options)
R_output = fid.for_values_in_list_find_value_return_value([0.5,1,2,5,10,15,20,30],T_inter,R_inter)
r0 = R_inter[0]
print(f"Problem 1a - 6M,1Y,2Y,5Y,10Y,15Y,20Y,30Y spot rates from the fit: {np.round(R_output,5)}")
# 1c) Computing the 10Y par swap rate
alpha_floating_leg = 0.5
T_10Y_swap = np.array([i*alpha_floating_leg for i in range(0,21)])
p_10Y_swap = fid.for_values_in_list_find_value_return_value(T_10Y_swap,T_inter,p_inter)
L_6M = fid.forward_libor_rates_from_zcb_prices(T_10Y_swap,p_10Y_swap,horizon = 1)
R_10Y_swap, S_10Y_swap = fid.swap_rate_from_zcb_prices(0,0,10,"annual",T_10Y_swap,p_10Y_swap)
print(f"Problem 1c - 10Y par swap rate: {R_10Y_swap}")

# Problem 2 - Pricing the interest rate cap
def fit_hwev_caplet_prices(param,price,strike_observed,T,p,scaling = 1):
    a, sigma = param
    caplet_price_fit = fid.caplet_prices_hwev(strike_observed,a,sigma,T,p)
    M = len(price)
    sse = 0
    for m in range(0,M):
        sse += scaling*(price[m] - caplet_price_fit[m])**2
    return sse

# 2a) Fitting a Vasicek model to the yield curve
sigma_vasicek = 0.02
param_0 = 0.035, 6, 0.25
result = minimize(fid.fit_vasicek_no_sigma_obj,param_0,method = 'nelder-mead',args = (sigma_vasicek,R_inter,T_inter),options={'xatol': 1e-20,'disp': False})
r0_vasicek, a_vasicek, b_vasicek = result.x
print(f"Problem 2a - Vaseicek parameters: r0: {r0_vasicek}, a: {a_vasicek}, b: {b_vasicek}, sigma: {sigma_vasicek}, SSE: {result.fun}")
p_vasicek = fid.zcb_price_vasicek(r0_vasicek,a_vasicek,b_vasicek,sigma_vasicek,T_inter)
f_vasicek = fid.forward_rate_vasicek(r0_vasicek,a_vasicek,b_vasicek,sigma_vasicek,T_inter)
R_vasicek = fid.spot_rate_vasicek(r0_vasicek,a_vasicek,b_vasicek,sigma_vasicek,T_inter)
# 2b) Fitting the HWEV model to caplet prices
param_0 = 2.5, 0.018
result = minimize(fit_hwev_caplet_prices,param_0,method = 'nelder-mead',args = (price_caplet_market,strike_caplet_market,T_10Y_swap,p_10Y_swap),options={'xatol': 1e-20,'disp': False})
a_hwev, sigma_hwev = result.x
print(f"Problem 2b - HWEV parameters: a: {a_hwev}, sigma: {sigma_hwev}, SSE: {result.fun}")
caplet_price_fit = fid.caplet_prices_hwev(strike_caplet_market,a_hwev,sigma_hwev,T_10Y_swap,p_10Y_swap)
sigma_market, sigma_fit = np.nan*np.ones(N_caplet), np.nan*np.ones(N_caplet)
for i in range(2,N_caplet):
    sigma_market[i] = fid.black_caplet_iv(price_caplet_market[i],T_10Y_swap[i],strike_caplet_market,0.5,p_10Y_swap[i],L_6M[i],type = "call",prec = 1e-10)
    sigma_fit[i] = fid.black_caplet_iv(caplet_price_fit[i],T_10Y_swap[i],strike_caplet_market,0.5,p_10Y_swap[i],L_6M[i],type = "call",prec = 1e-10)
# 2c) - Simulated trajectories of the Vasick and HWEV models
size_ci = 0.95
M_simul, T_simul = 500, 10
mesh_simul = T_simul/M_simul
t_simul = np.array([i*mesh_simul for i in range(0,M_simul+1)])
r_simul_vasicek = fid.simul_vasicek(r0_vasicek,a_vasicek,b_vasicek,sigma_vasicek,M_simul,T_simul,method = "euler")
mean_vasicek = fid.mean_vasicek(r0_vasicek, a_vasicek, b_vasicek, sigma_vasicek,t_simul)
lb_vasicek, ub_vasicek = fid.ci_vasicek(r0_vasicek, a_vasicek, b_vasicek, sigma_vasicek,t_simul,size_ci,type_ci = "two_sided")
lb_sd_vasicek, ub_sd_vasicek = fid.ci_vasicek(r0_vasicek, a_vasicek, b_vasicek, sigma_vasicek,np.inf,size_ci,type_ci = "two_sided")
print(f"Problem 2c - Vasicek model 2-sided CI under the stationary distribution: {lb_sd_vasicek}, {ub_sd_vasicek}")
f_simul, f_T_simul = fid.interpolate(t_simul,T_inter,f_inter,interpolation_options)
theta_hwev = fid.theta_hwev(t_simul,f_simul,f_T_simul,a_hwev,sigma_hwev)
r_simul_hwev = fid.simul_hwev(r0,t_simul,theta_hwev,a_hwev,sigma_hwev,method = "euler")
mean_hwev, var_hwev = fid.mean_var_hwev(a_hwev,sigma_hwev,t_simul,f_simul,f_T_simul)
lb_hwev, ub_hwev = fid.ci_hwev(a_hwev,sigma_hwev,t_simul,f_simul,f_T_simul,size_ci,type_ci = "two_sided")
```

SOLUTION

```
print(f"HWEV model 2-sided CI under the stationary distribution: {lb_hwev[-1]}, {ub_hwev[-1]}")
# 2d) - Price of the interest cap
strike_cap = 0.06
caplet_price_cap = fid.caplet_prices_hwev(strike_cap,a_hwev,sigma_hwev,T_10Y_swap,p_10Y_swap)
caplet_price_report = []
for i in [2,4,8,12,16,20]:
    caplet_price_report.append(np.round(10000*caplet_price_cap[i],4))
price_cap = sum(caplet_price_cap[2:])
premium_cap = alpha_floating_leg*price_cap/S_10Y_swap
print(f"Problem 2d - Caplet prices for T=1,2,4,6,8,10: {caplet_price_report}")
print(f"Problem 2d - price_cap: {10000*price_cap}, premium_cap: {10000*premium_cap}")

# Problem 3 - Computing the price of the 3Y7Y payer swaption
# 3a) 3Y7Y swaption price
T_n, T_N = 3, 10
beta = 0.55
R_swaption, S_swaption = fid.swap_rate_from_zcb_prices(0,T_n,T_N,"annual",T_10Y_swap,p_10Y_swap)
print(f"ATMF 3Y7Y par swap rate: {R_swaption}")
N_swaption = len(K_swaption_offset)
K_swaption = K_swaption_offset/10000 + R_swaption*np.ones(N_swaption)
price_swaption_market = np.zeros([N_swaption])
for i in range(0,N_swaption):
    price_swaption_market[i] = fid.black_swaption_price(iv_swaption_market[i],T_n,K_swaption[i],S_swaption,R_swaption)
print(f"Payer swaption with strike closest to K=0.06. strike: {K_swaption[7]}, price: {price_swaption_market[7]*10000} bps")
# 3b) Fitting the SABR model
param_0 = 0.025, 0.48,-0.25
result = minimize(fid.fit_sabr_no_beta_obj,param_0,method = 'nelder-mead',args = (beta,iv_swaption_market,K_swaption,T_n,R_swaption),options={'xatol': 1e-8,'disp': False})
sigma_0, upsilon, rho = result.x
print(f"Parameters from the SABR fit where beta: {beta} are: sigma_0: {sigma_0}, upsilon: {upsilon}, rho: {rho}, SSE: {result.fun}")
iv_fit, price_fit = np.zeros([N_swaption]), np.zeros([N_swaption])
for i in range(0,N_swaption):
    iv_fit[i] = fid.sigma_sabr(K_swaption[i],T_n,R_swaption,sigma_0,beta,upsilon,rho)
    price_fit[i] = fid.black_swaption_price(iv_fit[i],T_n,K_swaption[i],S_swaption,R_swaption,type = "call")
# 3c) Price and risk management of a strangle
iv_payer_swaption = fid.sigma_sabr(K_swaption[8],T_n,R_swaption,sigma_0,beta,upsilon,rho)
price_payer_swaption_init = fid.black_swaption_price(iv_payer_swaption,T_n,K_swaption[8],S_swaption,R_swaption,type = "call")
iv_receiver_swaption = fid.sigma_sabr(K_swaption[4],T_n,R_swaption,sigma_0,beta,upsilon,rho)
price_receiver_swaption_init = fid.black_swaption_price(iv_receiver_swaption,T_n,K_swaption[4],S_swaption,R_swaption,type = "put")
print(f"Initially. Payer swaption: {price_payer_swaption_init*10000}, Receiver swaption: {price_receiver_swaption_init*10000}, Strangle: {price_payer_swaption_init*10000+price_receiver_swaption_init
bps = 0.0001
iv_payer_swaption = fid.sigma_sabr(K_swaption[8],T_n,R_swaption + bps,sigma_0,beta,upsilon,rho)
price_payer_swaption = fid.black_swaption_price(iv_payer_swaption,T_n,K_swaption[8],S_swaption,R_swaption + bps,type = "call")
iv_receiver_swaption = fid.sigma_sabr(K_swaption[4],T_n,R_swaption + bps,sigma_0,beta,upsilon,rho)
price_receiver_swaption = fid.black_swaption_price(iv_receiver_swaption,T_n,K_swaption[4],S_swaption,R_swaption + bps,type = "put")
print(f"Par swap rate UP 1 bps. Payer swaption: {price_payer_swaption*10000}, Receiver swaption: {price_receiver_swaption*10000}, Strangle: {price_payer_swaption*10000+price_receiver_swaption*10000}
price_payer_swaption_init-price_receiver_swaption_init)}")
iv_payer_swaption = fid.sigma_sabr(K_swaption[8],T_n,R_swaption - bps,sigma_0,beta,upsilon,rho)
price_payer_swaption = fid.black_swaption_price(iv_payer_swaption,T_n,K_swaption[8],S_swaption,R_swaption - bps,type = "call")
iv_receiver_swaption = fid.sigma_sabr(K_swaption[4],T_n,R_swaption - bps,sigma_0,beta,upsilon,rho)
price_receiver_swaption = fid.black_swaption_price(iv_receiver_swaption,T_n,K_swaption[4],S_swaption,R_swaption - bps,type = "put")
print(f"Par swap rate DOWN 1 bps. Payer swaption: {price_payer_swaption*10000}, Receiver swaption: {price_receiver_swaption*10000}, Strangle: {price_payer_swaption*10000+price_receiver_swaption*1000
price_payer_swaption_init-price_receiver_swaption_init)}")
iv_payer_swaption = fid.sigma_sabr(K_swaption[8],T_n,R_swaption,sigma_0 + 0.001,beta,upsilon,rho)
price_payer_swaption = fid.black_swaption_price(iv_payer_swaption,T_n,K_swaption[8],S_swaption,R_swaption,type = "call")
iv_receiver_swaption = fid.sigma_sabr(K_swaption[4],T_n,R_swaption,sigma_0 + 0.001,beta,upsilon,rho)
price_receiver_swaption = fid.black_swaption_price(iv_receiver_swaption,T_n,K_swaption[4],S_swaption,R_swaption,type = "put")
print(f"Volatility 0.001 UP. Payer swaption: {price_payer_swaption*10000}, Receiver swaption: {price_receiver_swaption*10000}, Strangle: {price_payer_swaption*10000+price_receiver_swaption*10000}, Di
price_payer_swaption_init-price_receiver_swaption_init)}")
iv_payer_swaption = fid.sigma_sabr(K_swaption[8],T_n,R_swaption,sigma_0 - 0.001,beta,upsilon,rho)
price_payer_swaption = fid.black_swaption_price(iv_payer_swaption,T_n,K_swaption[8],S_swaption,R_swaption,type = "call")
iv_receiver_swaption = fid.sigma_sabr(K_swaption[4],T_n,R_swaption,sigma_0 - 0.001,beta,upsilon,rho)
price_receiver_swaption = fid.black_swaption_price(iv_receiver_swaption,T_n,K_swaption[4],S_swaption,R_swaption,type = "put")
print(f"Volatility 0.001 DOWN. Payer swaption: {price_payer_swaption*10000}, Receiver swaption: {price_receiver_swaption*10000}, Strangle: {price_payer_swaption*10000+price_receiver_swaption*10000},
price_payer_swaption_init-price_receiver_swaption_init)}")

# problem 5
def mean_var(x0,a,b,sigma,y0,gamma,phi,T):
    N = len(T)
    mean = np.zeros(N)
    var = np.zeros(N)
    for n, t in enumerate(T):
        mean[n] = x0*np.exp(-gamma*T[n]) + y0*np.exp(-a*T[n]) + b/a*(1-np.exp(-a*T[n]))
        var[n] = phi**2/(2*gamma)*(1-np.exp(-2*gamma*T[n])) + sigma**2/(2*a)*(1-np.exp(-2*a*T[n]))
    return mean, var

# 5c) Simulating the factors driving the short rate and the short rate itself
x0, gamma, phi = 0, 32, 0.03
y0, a, b, sigma = 0.03, 0.5, 0.025, 0.015
M = 1000
T_st = 1
T_lt = 10
size_ci = 0.95
z = norm.ppf(size_ci + 0.5*(1-size_ci),0,1)
delta_st = T_st/M
t_simul_st = np.array([i*delta_st for i in range(0,M+1)])
X_st = fid.simul_vasicek(x0,gamma,0,phi,M,T_st)
Y_st = fid.simul_vasicek(y0,a,b,sigma,M,T_st)
r_st = X_st + Y_st
mean_st, var_st = mean_var(x0,a,b,sigma,y0,gamma,phi,t_simul_st)
lb_st, ub_st = mean_st - z*np.sqrt(var_st), mean_st + z*np.sqrt(var_st)
delta_lt = T_lt/M
t_simul_lt = np.array([i*delta_lt for i in range(0,M+1)])
X_lt = fid.simul_vasicek(x0,gamma,0,phi,M,T_lt)
Y_lt = fid.simul_vasicek(y0,a,b,sigma,M,T_lt)
r_lt = X_lt + Y_lt
mean_lt, var_lt = mean_var(x0,a,b,sigma,y0,gamma,phi,t_simul_lt)
lb_lt, ub_lt = mean_lt - z*np.sqrt(var_lt), mean_lt + z*np.sqrt(var_lt)


# Plot of the fitted ZCB term structures of spot and forward rates
fig = plt.figure(constrained_layout=False, dpi = 300, figsize = (5,3))
fig.suptitle(f"Calibrated Zero Coupon Spot and Forward Rates", fontsize = 10)
gs = fig.add_gridspec(nrows=1,ncols=1,left=0.12,bottom=0.2,right=0.88,top=0.90,wspace=0,hspace=0)
```

SOLUTION

```
ax = fig.add_subplot(gs[0,0])
xticks = [0,1,2,3,4,5,7,10,15,20,30]
ax.set_xticks(xticks)
ax.set_xticklabels(xticks,fontsize = 6)
ax.set_xlim([xticks[0]+-0.2,xticks[-1]+0.2])
plt.xlabel(f"Maturity",fontsize = 6)
ax.set_yticks([0,0.02,0.04,0.06,0.08])
ax.set_yticklabels([0,0.02,0.04,0.06,0.08],fontsize = 6)
ax.set_ylim([0,0.0825])
plt.grid(axis = 'y', which='major', color=(0.7,0.7,0.7,0), linestyle='--')
p1 = ax.scatter(T_inter[0:], R_inter[0:], s = 1, color = 'black', marker = ".",label="Spot rates")
p2 = ax.scatter(T_inter[0:], f_inter[0:], s = 1, color = 'red', marker = ".",label="forward rates")
plots = [p1,p2]
labels = [item.get_label() for item in plots]
ax.legend(plots,labels,loc="lower right",fontsize = 6)
bbox = {"facecolor": (1,1,1,0.8),"edgecolor": (0.7,0.7,0.7,0.5),"boxstyle": "Round"}
if interpolation_options["method"] == "hermite":
    ax.text(0.32,0.0032,f" method: {interpolation_options['method']} \n degree: {interpolation_options['degree']} \n transition: {interpolation_options['transition']}", fontsize = 6,linespacing = 1.7
else:
    ax.text(0.32,0.0032,f" method: {interpolation_options['method']} \n transition: {interpolation_options['transition']}", fontsize = 6,linespacing = 1.7, bbox = bbox)
# plt.show()

# Plot of 6M forward Euribor rates and the 10Y par swap rate
fig = plt.figure(constrained_layout=False, dpi = 300, figsize = (5,3))
fig.suptitle(f"6M forward Euribor Rates and the 10Y Par Swap Rate", fontsize = 10)
gs = fig.add_gridspec(nrows=1,ncols=1,left=0.12,bottom=0.2,right=0.88,top=0.90,wspace=0,hspace=0)
ax = fig.add_subplot(gs[0,0])
xticks = np.array([0,2,4,6,8,10])
ax.set_xticks(xticks)
ax.set_xticklabels(xticks,fontsize = 6)
ax.set_xlim([xticks[0]+-0.2,xticks[-1]+0.2])
plt.xlabel(f"Maturity",fontsize = 6)
ax.set_yticks([0,0.02,0.04,0.06,0.08])
ax.set_yticklabels([0,0.02,0.04,0.06,0.08],fontsize = 6)
ax.set_ylim([0,0.0825])
plt.grid(axis = 'y', which='major', color=(0.7,0.7,0.7,0), linestyle='--')
p1 = ax.scatter(T_10Y_swap, L_6M, s = 2, color = 'black', marker = ".",label="6M Forward Euribor Rates")
p2 = ax.axhline(y=R_10Y_swap, linewidth = 1.5,color = 'red',label="10Y Par Swap Rate")
plots = [p1,p2]
labels = [item.get_label() for item in plots]
ax.legend(plots,labels,loc="lower right",fontsize = 6)
# plt.show()

# Plot of Calibrated ZCB Spot Rates Fitted Using Vasicek
fig = plt.figure(constrained_layout=False, dpi = 300, figsize = (5,3))
fig.suptitle(f"Calibrated ZCB Spot Rates Fitted Using Vasicek", fontsize = 10)
gs = fig.add_gridspec(nrows=1,ncols=1,left=0.12,bottom=0.2,right=0.88,top=0.90,wspace=0,hspace=0)
ax = fig.add_subplot(gs[0,0])
xticks = [0,1,2,3,4,5,7,10,15,20,30]
ax.set_xticks(xticks)
ax.set_xticklabels(xticks,fontsize = 6)
ax.set_xlim([xticks[0]+-0.2,xticks[-1]+0.2])
plt.xlabel(f"Maturity",fontsize = 6)
ax.set_yticks([0,0.02,0.04,0.06,0.08])
ax.set_yticklabels([0,0.02,0.04,0.06,0.08],fontsize = 6)
ax.set_ylim([0,0.0825])
plt.grid(axis = 'y', which='major', color=(0.7,0.7,0.7,0), linestyle='--')
p1 = ax.scatter(T_inter[0:], R_inter[0:], s = 1, color = 'black', marker = ".",label="Spot Rates")
p2 = ax.scatter(T_inter, R_vasicek, s = 1, color = 'red', marker = ".",label="Vasicek Fitted Spot Rates")
plots = [p1,p2]
labels = [item.get_label() for item in plots]
ax.legend(plots,labels,loc="lower right",fontsize = 6)
bbox = {"facecolor": (1,1,1,0.8),"edgecolor": (0.7,0.7,0.7,0.5),"boxstyle": "Round"}
if interpolation_options["method"] == "hermite":
    ax.text(0.32,0.0032,f" method: {interpolation_options['method']} \n degree: {interpolation_options['degree']} \n transition: {interpolation_options['transition']}", fontsize = 6,linespacing = 1.7
else:
    ax.text(0.32,0.0032,f" method: {interpolation_options['method']} \n transition: {interpolation_options['transition']}", fontsize = 6,linespacing = 1.7, bbox = bbox)
# plt.show()

# Plot of fitted and observed caplet implied volatility
fig = plt.figure(constrained_layout=False, dpi = 300, figsize = (5,3))
fig.suptitle(f"Market Caplet Implied Volatilities Fitted Using HWEV", fontsize = 10)
gs = fig.add_gridspec(nrows=1,ncols=1,left=0.12,bottom=0.2,right=0.88,top=0.90,wspace=0,hspace=0)
ax = fig.add_subplot(gs[0,0])
xticks = np.array([0,2,4,6,8,10])
ax.set_xticks(xticks)
ax.set_xticklabels(xticks,fontsize = 6)
ax.set_xlim([xticks[0]+-0.2,xticks[-1]+0.2])
plt.xlabel(f"Maturity",fontsize = 6)
ax.set_yticks([0,0.02,0.04,0.06,0.08,0.1,0.12])
ax.set_yticklabels([0,0.02,0.04,0.06,0.08,0.1,0.12],fontsize = 6)
ax.set_ylim([0,0.1225])
plt.grid(axis = 'y', which='major', color=(0.7,0.7,0.7,0), linestyle='--')
p1 = ax.scatter(T_10Y_swap, sigma_market, s = 6, color = 'black', marker = ".",label="IV from Market Caplet Prices")
p2 = ax.scatter(T_10Y_swap, sigma_fit, s = 2, color = 'red', marker = ".",label="IV from Fitted Caplet Prices")
plots = [p1,p2]
labels = [item.get_label() for item in plots]
ax.legend(plots,labels,loc="lower right",fontsize = 6)
# plt.show()

# PLot of simulated short rates in the Vasicek model
fig = plt.figure(constrained_layout=False,dpi=300,figsize=(5,3))
fig.suptitle(f"Simulated Short Rate in the Fitted Vasicek Model",fontsize=10)
gs = fig.add_gridspec(nrows=1,ncols=1,left=0.12,bottom=0.2,right=0.88,top=0.90,wspace=0,hspace=0)
ax = fig.add_subplot(gs[0,0])
xticks = [0,2,4,6,8,10]
ax.set_xticks(xticks)
ax.set_xticklabels(xticks,fontsize = 6)
ax.set_xlim([xticks[0]-0.01,xticks[-1]+0.01])
plt.xlabel(f"Time",fontsize = 7)
yticks1 = [0,0.02,0.04,0.06,0.08,0.1]
ax.set_yticks(yticks1)
ax.set_yticklabels(yticks1,fontsize = 6)
```

SOLUTION

```
ax.set_ylim([yticks1[0],yticks1[-1] + (yticks1[-1]-yticks1[0])*0.02])
plt.grid(axis = 'y', which='major', color=(0.7,0.7,0.7,0), linestyle='--')
ax.set_ylabel(f"Short rate",fontsize = 7)
p1 = ax.scatter(t_simul, mean_vasicek, s = 1, color = 'blue', marker = ".",label=f"Mean")
p2 = ax.scatter(t_simul, lb_vasicek, s = 1, color = 'red', marker = ".",label=f"Lower {size_ci} CB")
p3 = ax.scatter(t_simul, ub_vasicek, s = 1, color = 'red', marker = ".",label=f"Upper {size_ci} CB")
p4 = ax.scatter(t_simul, r_simul_vasicek, s = 1, color = 'black', marker = ".",label="Simulated short rate")
plots = [p1,p2,p3,p4]
labels = [item.get_label() for item in plots]
ax.legend(plots,labels,loc="upper right",fontsize = 5)
# plt.show()

# PLot of simulated short rates in the Hull-White Extended Vasicek model
fig = plt.figure(constrained_layout=False,dpi=300,figsize=(5,3))
fig.suptitle(f"Simulated Short Rate in the Fittted HWEV Model",fontsize=10)
gs = fig.add_gridspec(nrows=1,ncols=1,left=0.12,bottom=0.2,right=0.88,top=0.90,wspace=0,hspace=0)
ax = fig.add_subplot(gs[0,0])
xticks = [0,2,4,6,8,10]
ax.set_xticks(xticks)
ax.set_xticklabels(xticks,fontsize = 6)
ax.set_xlim([xticks[0]-0.01,xticks[-1]+0.01])
plt.xlabel(f"Time",fontsize = 7)
yticks1 = [0,0.02,0.04,0.06,0.08,0.1]
ax.set_yticks(yticks1)
ax.set_yticklabels(yticks1,fontsize = 6)
ax.set_ylim([yticks1[0],yticks1[-1] + (yticks1[-1]-yticks1[0])*0.02])
plt.grid(axis = 'y', which='major', color=(0.7,0.7,0.7,0), linestyle='--')
ax.set_ylabel(f"Short rate",fontsize = 7)
p1 = ax.scatter(t_simul, mean_hwev, s = 1, color = 'blue', marker = ".",label=f"Mean")
p2 = ax.scatter(t_simul, lb_hwev, s = 1, color = 'red', marker = ".",label=f"Lower {size_ci} CB")
p3 = ax.scatter(t_simul, ub_hwev, s = 1, color = 'red', marker = ".",label=f"Upper {size_ci} CB")
p4 = ax.scatter(t_simul, r_simul_hwev, s = 1, color = 'black', marker = ".",label="Simulated short rate")
labels = [item.get_label() for item in plots]
ax.legend(plots,labels,loc="upper right",fontsize = 5)
# plt.show()

# Plot of swaption market implied volatilities
fig = plt.figure(constrained_layout=False, dpi = 300, figsize = (5,3))
fig.suptitle(f"Market and Fitted Swaption Implied Volatilities", fontsize = 10)
gs = fig.add_gridspec(nrows=1,ncols=1,left=0.12,bottom=0.2,right=0.88,top=0.90,wspace=0,hspace=0)
ax = fig.add_subplot(gs[0,0])
xticks = K_swaption_offset
ax.set_xticks(xticks)
ax.set_xticklabels(xticks,fontsize = 6)
ax.set_xlim([xticks[0]-10,xticks[-1]+10])
plt.xlabel(f"Strike offset",fontsize = 7)
ax.set_yticks([0,0.06,0.12,0.18,0.24])
ax.set_yticklabels([0,0.06,0.12,0.18,0.24],fontsize = 6)
ax.set_ylim([0,0.245])
ax.set_ylabel(f"Implied volatility",fontsize = 7)
plt.grid(axis = 'y', which='major', color=(0.7,0.7,0.7,0), linestyle='--')
p1 = ax.scatter(K_swaption_offset, iv_swaption_market, s = 6, color = 'black', marker = ".",label="IV market")
p2 = ax.scatter(K_swaption_offset, iv_fit, s = 2, color = 'red', marker = ".",label="IV fit")
plots = [p1,p2]
labels = [item.get_label() for item in plots]
ax.legend(plots,labels,loc="upper right",fontsize = 5)
# plt.show()

# PLot of Short term fluctuations
fig = plt.figure(constrained_layout=False,dpi=300,figsize=(5,3))
fig.suptitle(f"Simulated Spot Rates up to T=1",fontsize=10)
gs = fig.add_gridspec(nrows=1,ncols=1,left=0.12,bottom=0.2,right=0.88,top=0.90,wspace=0,hspace=0)
ax = fig.add_subplot(gs[0,0])
xticks = np.array([0,0.2,0.4,0.6,0.8,1])
ax.set_xticks(xticks)
ax.set_xticklabels(xticks,fontsize = 6)
ax.set_xlim([xticks[0]-0.02*(xticks[-1]-xticks[0]),xticks[-1]+0.02*(xticks[-1]-xticks[0])])
plt.xlabel(f"Time",fontsize = 7)
yticks = [-0.02,0,0.02,0.04,0.06,0.08,0.1]
ax.set_yticks(yticks)
ax.set_yticklabels(yticks,fontsize = 6)
ax.set_ylim([yticks[0],yticks[-1] + (yticks[-1]-yticks[0])*0.02])
plt.grid(axis = 'y', which='major', color=(0.7,0.7,0.7,0), linestyle='--')
ax.set_ylabel(f"Short rate",fontsize = 7)
p1 = ax.scatter(t_simul_st, mean_st, s = 0.25, color = 'black', marker = ".",label=f"Mean")
p2 = ax.plot(t_simul_st, lb_st, color = 'black',linestyle = "dashdot", linewidth = 1,label=f"Lower 95% CI bound")
p3 = ax.plot(t_simul_st, ub_st, color = 'black',linestyle = "dashdot", linewidth = 1,label=f"Upper 95% CI bound")
p4 = ax.scatter(t_simul_st, X_st, s = 1, color = 'green', marker = ".",label=f"X")
p5 = ax.scatter(t_simul_st, Y_st, s = 1, color = 'blue', marker = ".",label=f"Y")
p6 = ax.scatter(t_simul_st, r_st, s = 1, color = 'red', marker = ".",label=f"short rate")
plots = [p1,p2[0],p3[0],p4,p5,p6]
labels = [item.get_label() for item in plots]
ax.legend(plots,labels,loc="upper left",fontsize = 5)
# plt.show()

# PLot of long term fluctuations
fig = plt.figure(constrained_layout=False,dpi=300,figsize=(5,3))
fig.suptitle(f"Simulated Spot Rates up to T=10",fontsize=10)
gs = fig.add_gridspec(nrows=1,ncols=1,left=0.12,bottom=0.2,right=0.88,top=0.90,wspace=0,hspace=0)
ax = fig.add_subplot(gs[0,0])
xticks = np.array([0,2,4,6,8,10])
ax.set_xticks(xticks)
ax.set_xticklabels(xticks,fontsize = 6)
ax.set_xlim([xticks[0]-0.02*(xticks[-1]-xticks[0]),xticks[-1]+0.02*(xticks[-1]-xticks[0])])
plt.xlabel(f"Time",fontsize = 7)
yticks = [-0.02,0,0.02,0.04,0.06,0.08,0.1]
ax.set_yticks(yticks)
ax.set_yticklabels(yticks,fontsize = 6)
ax.set_ylim([yticks[0],yticks[-1] + (yticks[-1]-yticks[0])*0.02])
plt.grid(axis = 'y', which='major', color=(0.7,0.7,0.7,0), linestyle='--')
ax.set_ylabel(f"Short rate",fontsize = 7)
p1 = ax.scatter(t_simul_lt, mean_lt, s = 0.25, color = 'black', marker = ".",label=f"Mean")
p2 = ax.plot(t_simul_lt, lb_lt, color = 'black',linestyle = "dashdot", linewidth = 1,label=f"Lower 95% CI bound")
```

SOLUTION

```
p3 = ax.plot(t_simul_lt, ub_lt, color = 'black',linestyle = "dashdot", linewidth = 1,label=f"Upper 95% CI bound")
p4 = ax.scatter(t_simul_lt, X_lt, s = 1, color = 'green', marker = ".",label=f"X")
p5 = ax.scatter(t_simul_lt, Y_lt, s = 1, color = 'blue', marker = ".",label=f"Y")
p6 = ax.scatter(t_simul_lt, r_lt, s = 1, color = 'red', marker = ".",label=f"Short rate")
plots = [p1,p2[0],p3[0],p4,p5,p6]
labels = [item.get_label() for item in plots]
ax.legend(plots,labels,loc="upper left",fontsize = 5)
plt.show()
```

SOLUTION