Quantitative Finance

SABR model - beta

Asked 1 year, 10 months ago Modified 1 year, 7 months ago Viewed 3k times

3

In the SABR model, the parameter beta largely controls the back-bond behaviour of the model. How do people estimate beta?

One approach is to regress atm vol vs forward, i.e.

$$\ln(\text{atm vol}) = \ln(\alpha) - (1-\beta) \times \ln(\text{forward}).$$

which comes from the SABR model (expansion). Are any other approaches used?

option-pricing stochastic-volatility volatility-surface sabr

Share Improve this question Follow

asked Dec 29, 2022 at 12:08

JohnRoper

31 1 4

2 Answers

Sorted by: Highest score (default)

5

In my experience, β is frequently pre-selected from a priori considerations because there is a large degree of redundancy between β and ρ (both affect the vol smile in similar ways).

As for example shown on P.91 of the article Managing Smile risk from Hagan et. al in the Willmott magazine, fitting α, ρ and ν for $\beta = 0$ as well as $\beta = 1$ results in no substantial difference in the quality of the fits. The screenshot below is from the article.

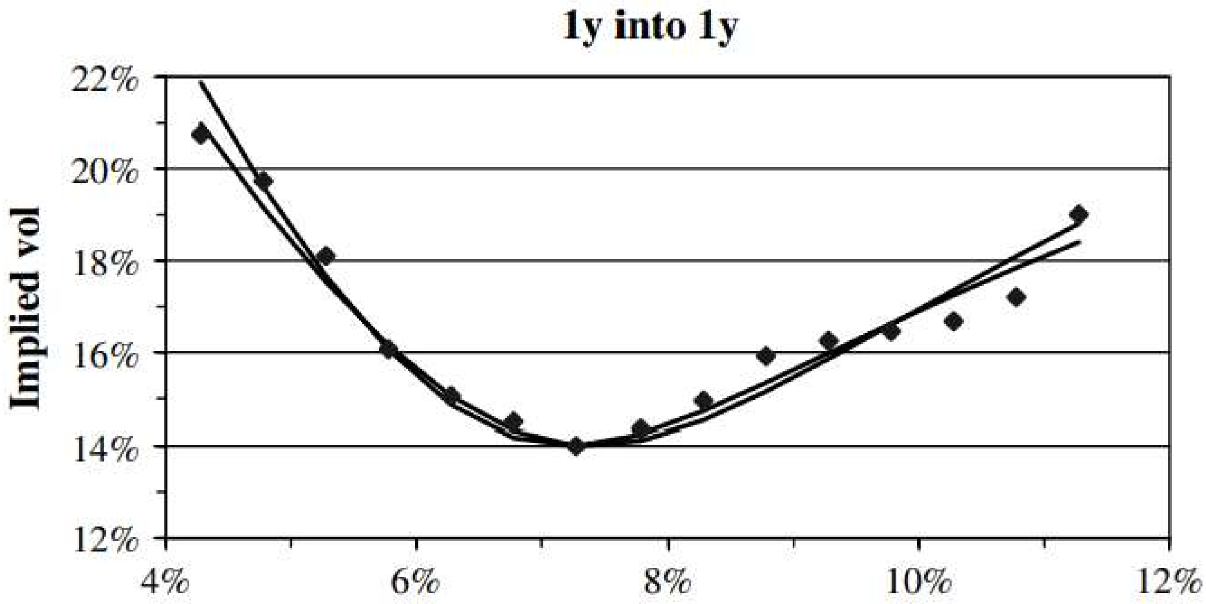


Fig. 3.3 Implied volatilities as a function of strike. Shown are the curves obtained by fitting the SABR model with exponent $\beta = 0$ and with $\beta = 1$ to the 1y into 1y swaption vol observed on 4/28/00. As usual, both fits are equally good. Data courtesy of Cantor-Fitzgerald.

The authors move on to explain that in their experience,

> market smiles can be fit equally well with any specific value of β.

The article also mentions your estimation procedure if you do not want to use a priori considerations for β. One problem is the data is usually noisy and the regression results are not very robust.

Another approach
While the choice of β may not be important for smile fitting, there are alternatives that try to minimize the hedging error. See for example [On the Estimation of the SABR Model's Beta Parameter: The Role of Hedging in Determining the Beta Parameter](). While the author works at Bloomberg, to the best of my knowledge, Bloomberg does use a hard coded value of 0.5 for their SABR implementation in `VCUB`.

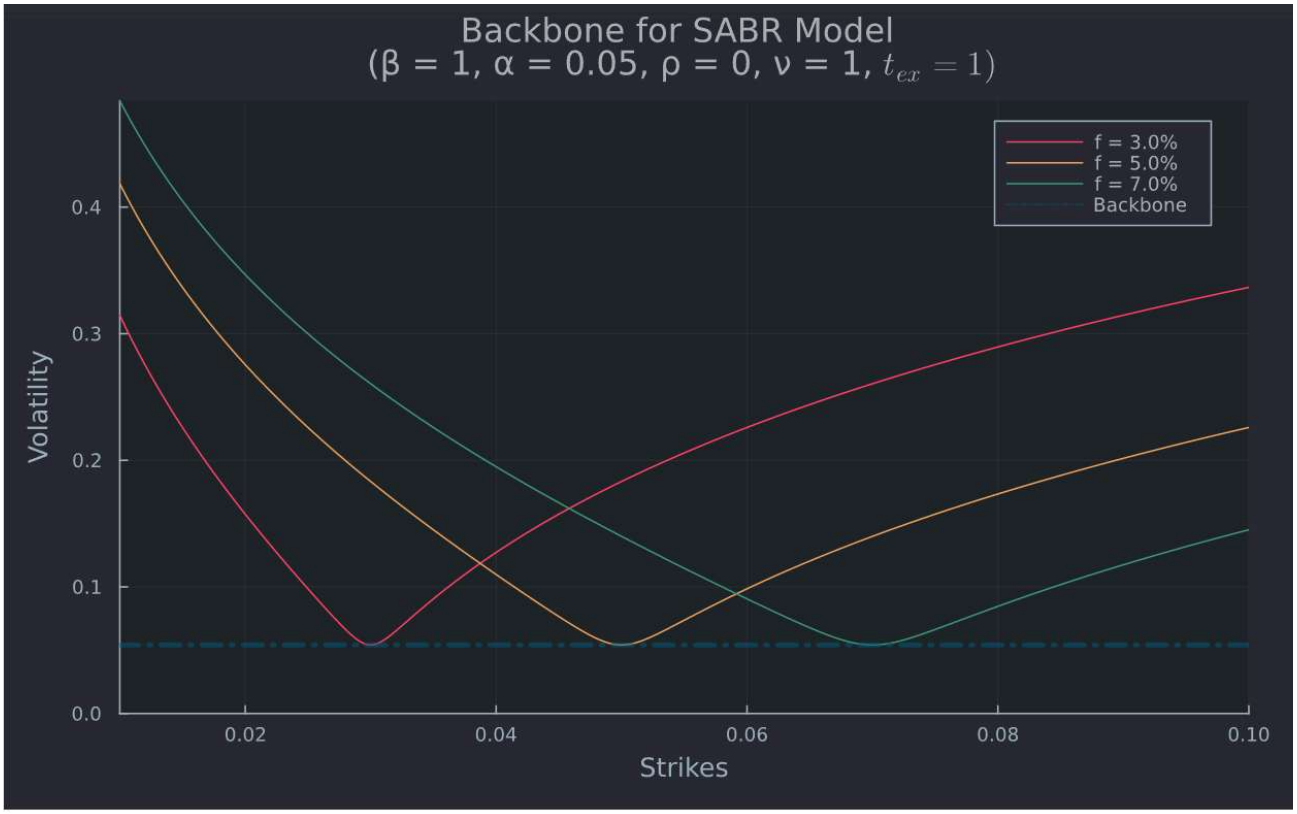
In the SABR model,

- $\sigma_{ATM} \approx \frac{\alpha}{f^{1-\beta}}$, where f is the underlying forward price.
- $\sigma_f = \alpha * f^{\beta-1}$ is the volatility of the forward rate. There are two reasons outlined why you may want to estimate β.

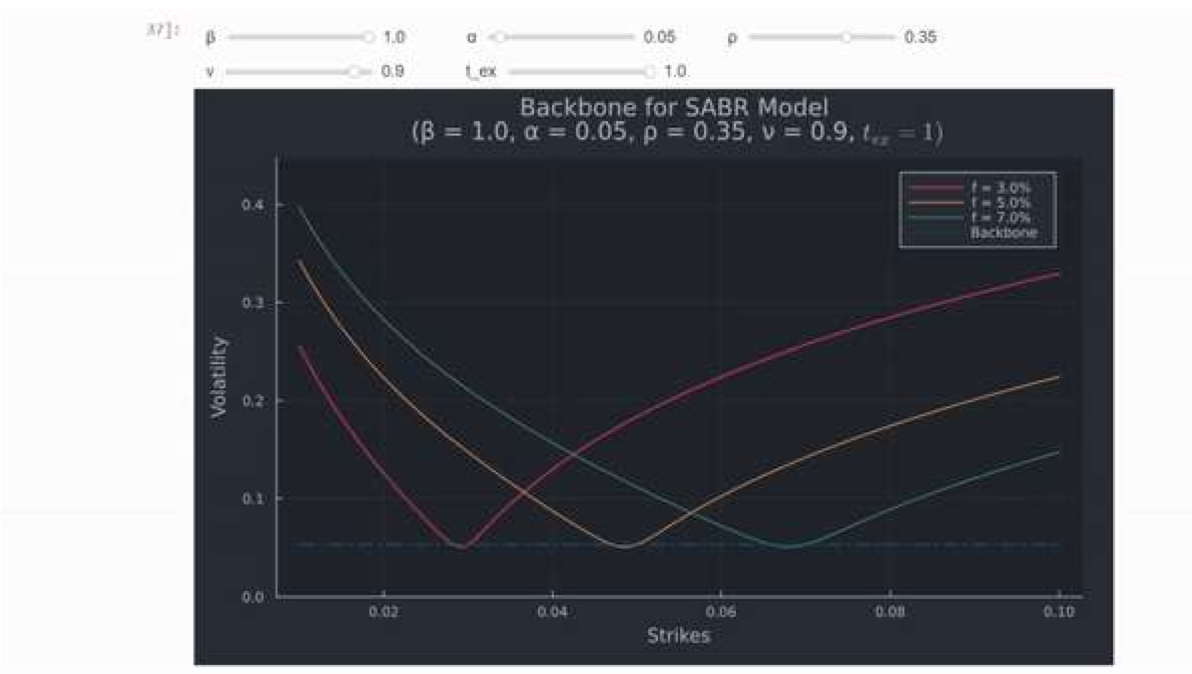
1. Given market vol, β determines how much of the volatility risk can be hedged by delta hedging (delta hedging eliminates the predictable vol change; $f^{\beta-1}$) and vega hedging (the stochastic vol risk; α)

2. β controls the trace of ATM volatilities as the underlying forward changes.
   $\Rightarrow$ If $\beta = 1$, $\alpha \approx \sigma_{ATM}$, implying σ_{ATM} does not change as *f* changes.
   $\Rightarrow$ If $\beta < 1$, σ_{ATM} will decrease if f increases because or $f^{\beta-1}$. This relationship between f and σ_{ATM} is referred to as the *backbone* and the backbone's shape shows how much additional volatility the option trader will take as the forward rate moves, as shown in the following graphic, which uses Julia. I am using hypothetical parameter values and forward rates, similiar to the paper.

```
# load packages
using Plots, PlotThemes, Interact, LaTeXStrings
theme(:juno)

#define inputs
β, α, ρ, ν, t_ex, f1, f2, f3, t_ex  = 1, 0.05, 0, 1, 1, 0.03, 0.05, 0.07, 1
K = 0.01:0.0001:0.1

#define the expression
function σ_b(β,α, ρ, ν, t_ex, f, K)
    A = α /((((f*K)^((1-β)/2))*(1+((1-β)^2)/24*log(2,(f/K))+ ((1-β)^4)/1920*log(4,
(f/K))))
    B = 1+((((1-β)^2)/24*(α^2/(f*K)^(1-β))+(1/4)*α*β*ρ*ν/((f*K)^((1-β)/2))+(2-
3*ρ^2)/24*ν^2)*t_ex
    z = ν/α*(f*K)^((1-β)/2)*log(f/K)
    χ_z = log((sqrt(1-2*ρ*z+z^2)+z-ρ)/(1-ρ))
    atm = α/(f^(1-β))*(1+(((1-β)^2)/24*(α^2/(f*K)^(1-β))+
(1/4)*α*β*ρ*ν/((f*K)^((1-β)/2))+(2-3*ρ^2)/24*ν^2)*t_ex)
    cond = f==K
  return cond ? atm : A*z/χ_z*B, atm
end

# define plots
plot(K,[x[1] for x in σ_b.(β,α, ρ, ν, t_ex, f1, K)], size =(800,500),
margin=5Plots.mm,
                                                            title = "Backbone
for SABR Model \n(β = $β, α = $α, ρ = $ρ, ν = $ν, "L"$ t_{ex}"*" = $t_ex)",
                                                            label = "f =
$(round((f1*100),digits=1))%",
                                                        xlabel = "Strikes",
                                                        ylabel = "Volatility")
plot!(K,[x[1] for x in σ_b.(β,α, ρ, ν, t_ex, f2, K)], label = "f =
$(round((f2*100),digits=1))%")
    plot!(K,[x[1] for x in σ_b.(β,α, ρ, ν, t_ex, f3, K)], label = "f =
$(round((f3*100),digits=1))%")
    plot!(K, [minimum(x[2] for x in σ_b.(β,α, ρ, ν, t_ex, fwd, K)) for fwd in K],
label = "Backbone",

linewidth = 3,

linestyle = :dashdot,

linealpha = 0.5)
    ylims!((0,maximum(x[1] for x in σ_b.(β,α, ρ, ν, t_ex, f3, K))))
    xlims!((minimum(K),maximum(K)))
```



Adding a few lines similar to this answer makes the chart interactive.



The entire implementation is a bit lengthy but well described in the referenced paper. In a nutshell, the authors come up with two solutions. The easier one is to use a two-step calibration;

- first step: use a pre-fixed beta to estimate the SABR parameters, which in turn are used to figure out the hedging error, based on delta and vega hedging, thereby computing the sum of squared relative hedging errors (which is a function of β) for all strikes from the option chain (the authors looked at Eurodollar futures options).

- second step: compute the optimal β that minimizes the sum of squared relative hedging errors

The downside of the two-step approach is that you cannot make a trade-off between hedging performance and vol smile fitting. Therefore, the authors also describe a second approach, that calibrates all parameters in one step, thereby minimizing the hedging error as well as the calibration error in one go.

Edit

I simply use the definitions and notation used in the referenced paper. Volatility risk is literally the risk associated with changes in volatility. As shown above, in the SABR model, $\sigma_f = \alpha * f^{\beta-1}$ is the volatility of the forward rate, where α is the stochastic part and $f^{\beta-1}$ is the predictable part (which implicitely affects the stochastic part via ρ). If $\beta = 1$, only the stochastic part matters. This is in essence similar to sticky delta, because the ATM vol stays the same as f moves (and the backbone is a vertical line). However, for $\beta < 1$, the predictable part can be hedged via delta hedging because it only depends on f. For more details, I recommend to read the paper I referenced.

Share Improve this answer Follow edited Dec 30, 2022 at 8:28 answered Dec 29, 2022 at 15:02

AKdemy
9,574 3 26 146

1 Thank you for the excellent and very comprehensive answer. In reason 1) above for estimating beta, you refer to "Delta" hedging. I initially assumed you mean hedging movements in the underlying forward rate, but you mention "Delta" in the context of hedging the volatility risk. Please could you elaborate? Also, how do you define "volatility risk"? – JohnRoper Dec 29, 2022 at 22:16

In the mentioned paper you can likely find the intuition you look for; depending on your choice of Beta, the forward will affect the (atm) vol differently. Hence, also your risk towards the forward (the delta) and towards the vol (the vega) in SABR are "intertwined" and contain "2nd order effects" (see e.g. equation 6 in the paper). Therefore, depending on the backbone dynamics (your choice of beta), you may loosely speaking "compensate" for some of your vega by adjusting your delta hedge and I have seen people use this as some sort of "rule of thumb" in practice when managing IR vol books – KevinT Dec 30, 2022 at 13:23

1 Extending this thought, take a look at the paper of Bartlett ("Hedging under SABR model"); he publishes an adjusted delta that "*incorporates the average change in volatility caused by changes in the underlying*". This became known as "Bartlett's Delta" even - and if you will it's sort of a vega component "merged" into the delta, which essentially boils down to sticky delta vs sticky strike dynamic considerations for your smile. – KevinT Dec 30, 2022 at 13:33

Although Bartlett (2006) provided a refined delta and vega under the SABR model, it was shown that for a portfolio that is both delta and vega hedged, the original SABR Greeks given by Hagan et al. (2002) provide essentially the same result as Bartlett's new SABR Greeks. See for example Hedging under SABR model by Bruce Bartlett in Wilmott Feb 2006. – AKdemy Dec 30, 2022 at 17:00

@JohnRoper, you can refer to [this answer] (quant.stackexchange.com/a/75169/54838) for a detailed explanation why Bartlett's delta also hedges volatility (and why it's similar to hedging with black delta and vega). – AKdemy Apr 11, 2023 at 19:41

3

A tentative answer as a practitioner.

SABR is over-specified, meaning you could fit a market smile to any arbitrary, non-pathological β. The so-called *redundancy* between β and ρ is limited to geometrical considerations though. IE $(\beta = 0, \rho_0, \nu_0)$ and $(\beta = 1, \rho_1, \nu_1)$ may both fit the smile perfectly, but they **will** produce a different delta, which matters a lot when managing a portfolio.

Remember that volatility markets in the rates space have been using normal vol for the last 15 years more or less. So make sure you look at σ_n and not σ_b in the Hagan paper, or when plotting your backbone, because rates market participants just don't speak logvol anymore (unlike their equity cousins).

Having said that, a couple possible approaches to decide on β:

- Mark It Zero (as per Walter Sobchak, *The Big Lebowski*). The idea is that you know exactly what it means to have a flat backbone and no backbone delta. It might be wrong, but there's a benefit in having a model where the practitioner (trader or portfolio manager) knows the exact quantity of delta he's getting from the backbone. And he may overlay his own finger-up-in-the-air hedging to whatever his SAAR analytics tell him. Also has the benefit of being compatible with sub-zero rates/forwards, and not a horrible assumption for a low rates environment (in 2021/2022, significant overlay needed from the trader for proper hedging).
- Infer it from the log/log vol/rate regression over a carefully chosen rolling window of time. The benefit is it will match recent market dynamics. Drawback is it moves, meaning more maintenance is needed from both the quants, to ensure the parameters don't go out of whack and are somewhat smooth on the expiry/tenor grid, AND from the trader who needs to have a sense of the dynamics implied by the current β

Quick comment on Bartlett: fine to use when you cant hedge the vol, say in illiquid markets. But be wary of changing relationships and putting too much in your delta.

Hats off to AKdemy for posting Julia code. Julia is the quant language of tomorrow, and I encourage quants to pick it up now.

Share Improve this answer Follow edited Apr 14, 2023 at 12:44 answered Apr 14, 2023 at 12:38

sob727
31 2

Julia is serving a niche market though. While Julia comes with a great mix of speed and usability, the main problem is that it still requires quite a bit of care to consistently reach C (C++ or Fortran speed) in Julia. Especially if all you knew prior to using Julia was Python. Personally, I like to use it a lot though. Especially for quick code snippets with interactive charts like the one above. – AKdemy Apr 14, 2023 at 14:54