makeVolSurface:

The purpose of this function is to create a volSurface structure. The fields of this structure includes:

spots: a vector of in terms of the Ts vector calculated with fwdCurve and Ts input;

Ts: input Ts in ascending order;

fwdCurve: a replication of input

slopes: storing of precomputed

smiles: storing of precomputed smile structure in terms of input Ts

Steps of implementation:

1. Invalid input testing;
2. Sort the Ts and vols in terms of the length of term to expiry. This step is to guarantee that the Ts vector is in ascending order;
3. Calculate spot price in terms of the Ts vector calculated with fwdCurve and Ts input ;
4. Calculate slopes
5. Store Ts, fwdCurve, spots, and slopes in the struct;
6. Precompute a series of smile structs in terms of input Ts and store them in the struct;
7. Perform no-arbitrage checking to guarantee that the price of a call option must be increasing along moneyness lines at strikes of forward spot prices.

getVol

The purpose of this function is to calculate the interpolated volatility given volSurface struct, T and a vector of Ks.

Steps of implementation:

1. Invalid input testing;
2. Calculate forward spot price in term of input T. This is also the first output fwd;
3. Calculate ;
4. Check the interval the input T lies in with respect to volSurface.Ts, and, by iterating each K in vector Ks, calculate with corresponding formula:
   1. If T lies in , this is the first case to calculate;
   2. If T lies in , the script will throw an error;
   3. If T lies in , this is the last case to calculate, where (Kis variable in the code) is calculated by for equal volSurface.Ts, and = getSmileVol(volSurface.smiles(j),Kis(j)) where j denotes the interval T lies in.
5. Output fwd and vol