2nd set SAS assignments

Data description for the two data sets `rwetrad` (RWE) and `tuitrad` (TUI):

Variable	Label	Explanation
`event_d`	Date of Trading	self explanatory
`event_t`	Time of Trading	self explanatory
`event_dt`	Date and Time of Transaction	self explanatory
`vol`	Number Securities Traded	self explanatory
`price`	Security Price DM	Transaction price in Euro
`bidprice`	Last recorded Bid Price	Prevailing bid price prior to the trade
`askprice`	Last recorded Ask Price	Prevailing ask price prior to the trade
`midpoint5`	Last recorded BA Midpoint 5 minutes after the trade	Prevailing Bid/Ask midpoint 5 minutes after the trade

Working with ultra high frequency data

1. Take the dataset `rwetrad` and create a new temporary data set that contains only values for one day. Use the `where` command in a datastep: `where event_d='17FEB04'd`, which retrieves all data for the 17th Februaray 2004. Plot the bid and askprice for one day into a graph. Label the axis and define a title.

2. Create a trade indicator (1 for a buy and -1 for a sell). Use the classification by the prevailing midquote and if that is inconclusive use the tick test:

i) First calculate the midquote from the bid- and askprice. Define a new variable named `inclass` that equals zero for all observations.

ii) Use an `if` statement to let `inclass` equal 1 if the transaction price is above the midquote and -1 if it is below the midquote.

iii Use `proc freq` to find out how many trades you have classified by that procedure.

iv) If not all trades are classified by the method above then use the tick test: Calculate a variable with the lagged transaction price. Then use an if statement again. If `inclass`

equals zero and the transaction price is larger than its lagged value, let `inclass` be 1 and respectively -1 if the transaction price is smaller than its lagged value.

v) Use `proc freq` to find out how many trades you have classified by that procedure.

vi) If there are still unclassified trades, look back one more lag (i.e. whether `lag1price < or > lag2price`.

3. Compute the effective spread ES, the realized spread RS and the price impact PI where

$$
\begin{aligned}
ES &= \begin{cases} 2 \cdot (price - midpoint) & \text{if buy} \\ 2 \cdot (midpoint - price) & \text{if sell} \end{cases} \\
RS &= \begin{cases} 2 \cdot (price - midpoint5) & \text{if buy} \\ 2 \cdot (midpoint5 - price) & \text{if sell} \end{cases} \\
PI &= (ES - RS)/2
\end{aligned}
$$

Again use an *if* statement in a data step (see SAS help!). Provide informative labels for the newly created spread variables.

4. The trading day starts at 9:00 and ends at 17:30. Dividing the trading day into 10 minutes intervals yields 51 intervals per day. Create with SAS two variables: an interval variable `inter1` and a count interval variable `c_int1`.

Interval	`inter1`	`c_int1`
$[9:00; 9:10[$	9:00	1
$[9:10; 9:20[$	9:10	2
	$\vdots$	
$[17:10; 17:20[$	17:10	50
$[17:20; 17:30]$	17:20	51

For that purpose, use the macro `xtenmin` in the help file on the course homepage. Try to figure out what the macro actually does. Call the macro `xtenmin` in a data step.

5. Write a macro (name it `datasampling`) that has the stock ticker (RWE, TUI) as argument. The content of the macro basically consists of a data step where the spread variables

are computed as in 3. and the ten minute indicator creation as in 4.. Call the macro for both stocks.

6. Choose again one data set. Call a sort procedure that sorts by `inter1`. Calculate means together with 90% confidence bands for the variable ES for each ten minute interval of the trading day across all trading days (Use `PROC MEANS` with the option *clm* and then *lclm* and *uclm* in the output statement (SAS help!)). Write the means to a new data set, label the means and the confidence bands, and create a variable with the stock ID in this data set (e.g. ticker="`RWE`").

7. Plot the means of the effective spread variable and its confidence bands for each ten minute interval against time of day in an appealing manner. Interpret your graphs.

/* Additional Task:*/
Repeat Task 6. and 7. for RS and PI and the second stock. For that purpose compute a macro that does the job of 6. and 7. and keeps the dataset as well as the variable flexible.