

## SECTION I: SAS CODE TO Read Enhanced Academic TRACE data

```
**** Read the Corp Trade File ****;

* Some fields in the Enhanced TRACE dataset used by Bessembinder,
Maxwell, Jacobsen and Venkataraman (2018, JF) differ from those
received by academic researchers under Academic TRACE subscription;

* local directory;
libname cblib 'folder with text data';

%macro readcbtrd(a,b);

  data WORK.testx;
    %let _EFIERR_ = 0; /* set the ERROR detection macro variable */
    infile "folder information/&a"
           delimiter = '|' MISSOVER DSD lrecl=32767
    firstobs=2 ;
    informat FINRA_SCRTY_ID $264. ;
    informat MSG_SEQ_NB best32. ;
    informat MSG_TYPE_CD $1. ;
    informat REC_TYPE_NM $16. ;
    informat TRD_RPT_TS anydtdtm40. ;
    informat TRD_EXCTN_TS anydtdtm40. ;
    informat TRD_STLMT_DT yymmdd10. ;
    informat ISSUE_SYM_ID $8. ;
    informat CUSIP_ID $9. ;
    informat RPT_SIDE_CD $1. ;
    informat ENTRD_VOL_QT best32. ;
    informat ENTRD_PR best32. ;
    informat PR_OVRRD_CD $1. ;
    informat CALCD_YLD_PT best32. ;
    informat SBMTG_FIRM_ID $1. ;
    informat RPTG_PARTY_ID $40. ;
    informat RPTG_PARTY_GVP_ID $40. ;
    informat CNTRA_PARTY_ID $40. ;
    informat CNTRA_PARTY_GVP_ID $40. ;
    informat RPTG_CPCTY_CD $1. ;
    informat CNTRA_CPCTY_CD $1. ;
    informat LCKD_IN_FL $1. ;
    informat TRD_MDFR_LATE_CD $1. ;
    informat TRD_MDFR_SRO_CD $1. ;
    informat PBLSH_FL $1. ;
    informat SLLR_CMSN_AMT best32. ;
    informat BUYER_CMSN_AMT best32. ;
    informat SLLR_FEES_AMT $1. ;
    informat BUYER_FEES_AMT $1. ;
    informat ASOF_FL $1. ;
    informat RVRSL_FL $1. ;
    informat SPCL_PR_FL $1. ;
    informat SPCL_PR_MEMO_TX $19. ;
```

```
informat SYSTM_CNTRL_NB best32. ;
informat SYSTM_CNTRL_DT yymmdd10. ;
informat PREV_TRD_CNTRL_NB best32. ;
informat PREV_TRD_CNTRL_DT yymmdd10. ;
informat FIRST_TRD_CNTRL_NB best32. ;
informat FIRST_TRD_CNTRL_DT yymmdd10. ;
informat TRD_ST_CD $1. ;
informat HIGH_PR $1. ;
informat HIGH_CNTRL_NB $1. ;
informat LOW_PR $1. ;
informat LOW_CNTRL_NB $1. ;
informat LSALE_PR $1. ;
informat LSALE_CNTRL_NB $1. ;
informat TRDG_MKT_CD $2. ;
informat SPCL_PRCSG_CD $1. ;
informat ATS_FL $1. ;
informat NO_RMNRN_CD $1. ;
format FINRA_SCRTY_ID $264. ;
format MSG_SEQ_NB best12. ;
format MSG_TYPE_CD $1. ;
format REC_TYPE_NM $16. ;
format TRD_RPT_TS datetime. ;
format TRD_EXCTN_TS datetime. ;
format TRD_STLMT_DT yymmdd10. ;
format ISSUE_SYM_ID $8. ;
format CUSIP_ID $9. ;
format RPT_SIDE_CD $1. ;
format ENTRD_VOL_QT best12. ;
format ENTRD_PR best12. ;
format PR_OVRRD_CD $1. ;
format CALCD_YLD_PT best12. ;
format SBMTG_FIRM_ID $1. ;
format RPTG_PARTY_ID $40. ;
format RPTG_PARTY_GVP_ID $40. ;
format CNTRA_PARTY_ID $40. ;
format CNTRA_PARTY_GVP_ID $40. ;
format RPTG_CPCTY_CD $1. ;
format CNTRA_CPCTY_CD $1. ;
format LCKD_IN_FL $1. ;
format TRD_MDFR_LATE_CD $1. ;
format TRD_MDFR_SRO_CD $1. ;
format PBLSH_FL $1. ;
format SLLR_CMSN_AMT best12. ;
format BUYER_CMSN_AMT best12. ;
format SLLR_FEES_AMT $1. ;
format BUYER_FEES_AMT $1. ;
format ASOF_FL $1. ;
format RVRSL_FL $1. ;
format SPCL_PR_FL $1. ;
format SPCL_PR_MEMO_TX $19. ;
format SYSTM_CNTRL_NB best12. ;
format SYSTM_CNTRL_DT yymmdd10. ;
```

```

format PREV_TRD_CNTRL_NB best12. ;
format PREV_TRD_CNTRL_DT yymmdd10. ;
format FIRST_TRD_CNTRL_NB best12. ;
format FIRST_TRD_CNTRL_DT yymmdd10. ;
format TRD_ST_CD $1. ;
format HIGH_PR $1. ;
format HIGH_CNTRL_NB $1. ;
format LOW_PR $1. ;
format LOW_CNTRL_NB $1. ;
format LSALE_PR $1. ;
format LSALE_CNTRL_NB $1. ;
format TRDG_MKT_CD $2. ;
format SPCL_PRCSG_CD $1. ;
format ATS_FL $1.;
format NO_RMNRN_CD $1.;
input

```

```

FINRA_SCRTY_ID
MSG_SEQ_NB
MSG_TYPE_CD $
REC_TYPE_NM $
TRD_RPT_TS
TRD_EXCTN_TS
TRD_STLMT_DT
ISSUE_SYM_ID $
CUSIP_ID $
RPT_SIDE_CD $
ENTRD_VOL_QT
ENTRD_PR
PR_OVRRD_CD $
CALCD_YLD_PT
SBMTG_FIRM_ID $
RPTG_PARTY_ID $
RPTG_PARTY_GVP_ID $
CNTRA_PARTY_ID $
CNTRA_PARTY_GVP_ID $
RPTG_CPCTY_CD $
CNTRA_CPCTY_CD $
LCKD_IN_FL $
TRD_MDFR_LATE_CD $
TRD_MDFR_SRO_CD $
PBLSH_FL $
SLLR_CMSN_AMT
BUYER_CMSN_AMT
SLLR_FEES_AMT $
BUYER_FEES_AMT $
ASOF_FL $
RVRSL_FL $
SPCL_PR_FL $
SPCL_PR_MEMO_TX $
SYSTEM_CNTRL_NB
SYSTEM_CNTRL_DT
PREV_TRD_CNTRL_NB

```

```

PREV_TRD_CNTRL_DT
FIRST_TRD_CNTRL_NB
FIRST_TRD_CNTRL_DT
TRD_ST_CD $
HIGH_PR $
HIGH_CNTRL_NB $
LOW_PR $
LOW_CNTRL_NB $
LSALE_PR $
LSALE_CNTRL_NB $
TRDG_MKT_CD $
SPCL_PRCSG_CD $
ATS_FL $
NO_RMNRN_CD $
;

    if FINRA_SCRTY_ID='Confidential Treatment Requested by FINRA'
then do;
    %put "Footer found";
    end;
    else output;
    if _ERROR_ then call symputx('_EFIERR_',1); /* set ERROR
detection macro variable */
    run;

    proc append base=cbtrd data=testx force; run;
    proc delete data=testx; run;

%mend;

*macro reads each text file and appends to the base dataset;
%macro readdatal;
%let filrf=mydir;
%let rc=%sysfunc(filename(filrf," folder info ")); /* assign dir name
*/
%let did=%sysfunc(dopen(&filrf)); /* open directory */
%let lstname=; /* clear filename macro var */
%let memcount=%sysfunc(dnum(&did)); /* get # files in directory */
%if &memcount > 0 %then /* check for blank directory */
    %do i=1 %to &memcount; /* start loop for files */
        %let lstname=%sysfunc(dread(&did,&i)); /* get file name to
process */

        %readcbtrd(&lstname,&i);
        %put &memcount &i &lstname;

    %end;
%let rc=%sysfunc(dclose(&did)); /* close directory */
%mend readdatal;
%readdatal;

```

---

- \*\*\*\* Clean the Corp Trade File \*\*\*\*;

```

data trade;
  set cbfl3.cbtrades_2017_18_g1new;
  format ATS_MP_ID $4.0;
  ATS_MP_ID=" ";
  TRD_EXCTN_DT = datepart(TRD_EXCTN_TS);
  TRD_EXCTN_TM = timepart(TRD_EXCTN_TS);
  TRD_RPT_DT = datepart(TRD_RPT_TS);
  TRD_RPT_TM = timepart(TRD_RPT_TS);
  rename ISSUE_SYM_ID = BOND_SYM_ID CALCD_YLD_PT = YLD_CALC_PT
RPTG_CPCTY_CD=MM_CPCTY_CD LCKD_IN_FL=AGU_TRD_ID ASOF_FL=ASOF_CD
RVRSL_FL=RVRSL_CD SPCL_PR_FL=SPCL_TRD_CD SPCL_PR_MEMO_TX=SPCL_MEMO_TX
TRD_ST_CD=TRC_ST;
run;

proc sort data=trade; by cusip_id trd_exctn_dt trd_exctn_tm; run;

data trade;
  set trade;
  if cusip_id=' ' then delete; *remove trades without valid cusip;
run;

proc freq data=trade; tables trc_st; run;

/** corrected trades **/

* remove cancelled trades;
data trade_correctsd;
  set trade;
  if trc_st="C" and rvrsl_cd=" ";
  keep cusip_id systm_cntrl_nb systm_cntrl_dt;
run;
proc sort data=trade_correctsd nodupkey; by cusip_id systm_cntrl_nb
systm_cntrl_dt; run;

data trade_correct1;
  set trade;
  if trc_st="C" and rvrsl_cd=" " then delete; *remove message;
run;

proc sort data=trade_correctsd nodupkey;
  by cusip_id systm_cntrl_nb systm_cntrl_dt; run;
proc sort data=trade_correct1;
  by cusip_id systm_cntrl_nb systm_cntrl_dt; run;

data trade_correct;
  merge trade_correct1 (in=a) trade_correctsd (in=b);
  by cusip_id systm_cntrl_nb systm_cntrl_dt;

```

```

        if b then delete; *remove original trade that is corrected;
run;

proc delete data=trade_correctsd; run;
proc delete data=trade_correct1; run;
proc delete data=trade; run;

/** cancelled trades **/
data trade_cancelisd;
    set trade_correct;
    if trc_st="X" and rvrsl_cd=" ";
    keep cusip_id systm_cntrl_nb systm_cntrl_dt;
run;
proc sort data=trade_cancelisd nodupkey; by cusip_id systm_cntrl_nb
systm_cntrl_dt; run;

data trade_correct2;
    set trade_correct;
    if trc_st="X" and rvrsl_cd=" " then delete; *remove message;
run;

proc sort data=trade_cancelisd nodupkey;
    by cusip_id systm_cntrl_nb systm_cntrl_dt; run;
proc sort data=trade_correct2;
    by cusip_id systm_cntrl_nb systm_cntrl_dt; run;

data trade_correct;
    merge trade_correct2 (in=a) trade_cancelisd (in=b);
    by cusip_id systm_cntrl_nb systm_cntrl_dt;
    if b then delete; *remove cancelled trade;
run;

proc delete data=trade_cancelisd; run;
proc delete data=trade_correct2; run;

/** cancelling reversals **/
data trade_cancelrv;
    set trade_correct;
    if trc_st="X" and rvrsl_cd="R" and ASOF_CD="R";
        *message canceling the reversal;
    keep cusip_id systm_cntrl_nb systm_cntrl_dt;
run;
proc sort data=trade_cancelrv nodupkey; by cusip_id systm_cntrl_nb
systm_cntrl_dt; run;

data trade_correct3;
    set trade_correct;
    if trc_st="X" and rvrsl_cd="R" and ASOF_CD="R" then delete;
        *remove message;
run;

```

```

proc sort data=trade_cancelrv nodupkey;
  by cusip_id system_cntrl_nb system_cntrl_dt; run;
proc sort data=trade_correct3;
  by cusip_id system_cntrl_nb system_cntrl_dt; run;

data trade_correct;
  merge trade_correct3 (in=a) trade_cancelrv (in=b);
  by cusip_id system_cntrl_nb system_cntrl_dt;
  if b then delete; *remove reversal that is cancelled;
run;
proc delete data=trade_cancelrv; run;
proc delete data=trade_correct3; run;

/** group 8 - color code dark blue - cancelling reversals **/
data trade_cancelrv;
  set trade_correct;
  if trc_st="C" and rvrsl_cd="R" and ASOF_CD="R"; *message
canceling the reversal;
  keep cusip_id system_cntrl_nb system_cntrl_dt;
run;
proc sort data=trade_cancelrv nodupkey; by cusip_id system_cntrl_nb
system_cntrl_dt; run;

data trade_correct4;
  set trade_correct;
  if trc_st="C" and rvrsl_cd="R" and ASOF_CD="R" then delete;
  *remove message;
run;

proc sort data=trade_cancelrv nodupkey;
  by cusip_id system_cntrl_nb system_cntrl_dt; run;
proc sort data=trade_correct4;
  by cusip_id system_cntrl_nb system_cntrl_dt; run;

data trade_correct;
  merge trade_correct4 (in=a) trade_cancelrv (in=b);
  by cusip_id system_cntrl_nb system_cntrl_dt;
  if b then delete; *remove reversal that is cancelled;
run;

proc delete data=trade_cancelrv; run;
proc delete data=trade_correct4; run;

/** cancelling trades after 20 business days **/
data trade_cancelrv;
  set trade_correct;
  if trc_st="Y" and rvrsl_cd="R" and ASOF_CD="R"; *message
canceling the trade;

```

```

        rename prev_trd_cntrl_nb=system_cntrl_nb
prev_trd_cntrl_dt=system_cntrl_dt;
        keep cusip_id prev_trd_cntrl_nb prev_trd_cntrl_dt;
run;

proc sort data=trade_cancelrv nodupkey; by cusip_id system_cntrl_nb
system_cntrl_dt; run;

data trade_correct4;
        set trade_correct;
        if trc_st="Y" and rvrsl_cd="R" and ASOF_CD="R" then delete;
*remove message;
run;

proc sort data=trade_cancelrv nodupkey;
        by cusip_id system_cntrl_nb system_cntrl_dt; run;
proc sort data=trade_correct4;
        by cusip_id system_cntrl_nb system_cntrl_dt; run;

data trade_correct;
        merge trade_correct4 (in=a) trade_cancelrv (in=b);
        by cusip_id system_cntrl_nb system_cntrl_dt;
        if b then delete; *remove trade that is cancelled;
run;

proc delete data=trade_cancelrv; run;
proc delete data=trade_correct4; run;

/** corrected reversals **/

data trade_cancelrv;
        set trade_correct;
        if trc_st="R" and rvrsl_cd="R" and ASOF_CD="R"; *message
reversing the trade;
        rename first_trd_cntrl_nb=system_cntrl_nb
first_trd_cntrl_dt=system_cntrl_dt;
        keep cusip_id first_trd_cntrl_nb first_trd_cntrl_dt;
run;
proc sort data=trade_cancelrv nodupkey; by cusip_id system_cntrl_nb
system_cntrl_dt; run;

data trade_correct4;
        set trade_correct;
        if trc_st="R" and rvrsl_cd="R" and ASOF_CD="R" then delete;
*remove message;
run;

proc sort data=trade_cancelrv nodupkey;
        by cusip_id system_cntrl_nb system_cntrl_dt; run;
proc sort data=trade_correct4;
        by cusip_id system_cntrl_nb system_cntrl_dt; run;

```



```

data trade_correct;
  merge trade_correct4 (in=a) trade_cancelrv (in=b);
  by cusip_id_sysm_cntrl_nb sysm_cntrl_dt;
  if b then delete;    *remove trade that is reversed;
run;
proc delete data=trade_correct4; run;

data trade_reversal; set trade_correct; run;

proc delete data=trade_correct; run;

options nocenter ls=180;
*options nomlogic nosymbolgen dquote nomprint nomacrogen nonotes;
*options nomlogic nosymbolgen dquote nomprint nomacrogen notes;
options mlogic symbolgen dquote mprint macrogen notes;

* local directory;
libname cb '/home/Bonds/CB/T2016to2018';
libname cblib '/backup/Bonds/CB/GlobalDealerCode';

data tradel; set cb.cbtrades_clean_2017_18_glnew; run;

*global dealer codes;
*-----;
data bdnet;
  set cblib.dealercode_global_2018_reallynew;
  rptg_new=global_dnum;
  cntra_new=global_dnum;
  rptg_gvp_new=global_dnum;
  cntra_gvp_new=global_dnum;
  drop global_dnum year group academic bdcodes;
run;

/* create a GLOBAL dealer code
Global_dnum is a unique dealer code created by the authors. This code
replaces the Masked ID with a simple number for each dealer.
This portion of the code is not included below.
*/

* 1. Customer Trades;
*-----;
data trade_cust; *final; *CHECK;
  set trade;
  if cust=1;
  * a unique dealer code for all customers is reported on TRACE;
  if RPT_SIDE_CD="B" then do;
    BuyDC=rptg;

```

```

        BuyRSC=RPTG_CPCTY_CD;
        SellDC=cntra;
        SellRSC=CNTRA_CPCTY_CD;
    end;
    if RPT_SIDE_CD="S" then do;
        SellDC=rptg;
        SellRSC=RPTG_CPCTY_CD;
        BuyDC=cntra;
        BuyRSC=CNTRA_CPCTY_CD;
    end;
run;

* 2. INTERDEALER TRADES;
*-----;
data trade_dealer;
    set trade;
    if cust=0;
run;

* 2.1 when combinations of rptg_gvp and cntra_gvp are missing;
*-----;
data test1_1; *final;
    set trade_dealer;
    if rptg_gvp=. and cntra_gvp=.;
    if RPT_SIDE_CD="B" then do;
        BuyDC=rptg;
        BuyRSC=RPTG_CPCTY_CD;
        SellDC=cntra;
        SellRSC=CNTRA_CPCTY_CD;
    end;
    if RPT_SIDE_CD="S" then do;
        SellDC=rptg;
        SellRSC=RPTG_CPCTY_CD;
        BuyDC=cntra;
        BuyRSC=CNTRA_CPCTY_CD;
    end;
run;

data test1_2; *final;
    set trade_dealer;
    if rptg_gvp^=. and cntra_gvp^=.;
    if RPT_SIDE_CD="B" then do;
        BuyDC=rptg_gvp;
        BuyRSC=RPTG_CPCTY_CD;
        SellDC=cntra_gvp;
        SellRSC=CNTRA_CPCTY_CD;
    end;
    if RPT_SIDE_CD="S" then do;
        SellDC=rptg_gvp;
        SellRSC=RPTG_CPCTY_CD;
    end;
run;

```

```

        BuyDC=cntra_gvp;
        BuyRSC=CNTRA_CPCTY_CD;
    end;
run;

data trade2;
    set trade_dealer;
    if rptg_gvp=. and cntra_gvp=. then delete;
    if rptg_gvp^=. and cntra_gvp^=. then delete;
run;

data test1_3;
    set trade2;
    if cntra_gvp^=. and rptg_gvp=.;
    if RPT_SIDE_CD="B" then do;
        BuyDC=rptg;
        BuyRSC=RPTG_CPCTY_CD;
        SellDC=cntra_gvp;
        SellRSC=CNTRA_CPCTY_CD;
    end;
    if RPT_SIDE_CD="S" then do;
        SellDC=rptg;
        SellRSC=RPTG_CPCTY_CD;
        BuyDC=cntra_gvp;
        BuyRSC=CNTRA_CPCTY_CD;
    end;
run;

data test1_4;
    set trade2;
    if cntra_gvp=. and rptg_gvp^=.;
    if RPT_SIDE_CD="B" then do;
        BuyDC=rptg_gvp;
        BuyRSC=RPTG_CPCTY_CD;
        SellDC=cntra;
        SellRSC=CNTRA_CPCTY_CD;
    end;
    if RPT_SIDE_CD="S" then do;
        SellDC=rptg_gvp;
        SellRSC=RPTG_CPCTY_CD;
        BuyDC=cntra;
        BuyRSC=CNTRA_CPCTY_CD;
    end;
run;

data trade_all; set trade_cust test1_1 test1_2 test1_3 test1_4; run;

```

```

/* double counting of inter-dealer trades & Affiliate Trades*/

*identify affiliate trades - interdealer trades with dc=3763. These
trades are identified in TRACE by a unique dealer code;

data trade_all;
    set trade_all;
    if selldc=3763 or buydc=3763 then aff=1; else aff=0;
run;

* adjust for double reporting of ID trades that are not affiliate
trades;
data trade_all;
    set trade_all;
    if cust=0 and aff=0 and rpt_side_cd="B" then delete;
run;

*some affiliate trades are simply transfer of positions to non-member
affiliates, reported by broker dealer with spcl_prmsg_cd=A. These
trades are not disseminated on TRACE and should be deleted;

data trade_all;
    set trade_all;
    if aff=1 and spcl_prmsg_cd="A" then delete;
run;

*some affiliate trades are legitimate trades with non-member
affiliates, reported by broker dealer with special code=" ". These
trades are disseminated on TRACE and can be treated as customer
trades (global_dnum=544);

data trade_all;
    set trade_all;
    if aff=1 and spcl_prmsg_cd^="A" then cust=1;
    if aff=1 and buydc=3763 then buydc=544;
    if aff=1 and selldc=3763 then selldc=544;
run;

data cb.trades_w_dealercodes_2018_rnew; set trade_all; run;

```