SECTION III: SAS CODE TO CREATE TABLE III, TRADING COSTS FOR CUSTOMER TRADES

```
*read BARCLAYS index values for benchmark;
data Barclay; set cb2.barclayindex_120116; run;
proc sort data=barclay; by date; run;
data barclay;
      set barclay;
      drop B_144A B_MBS B_CMBS B_ABS B_TShort S_P_500;
      SPINDX=S_P_500;
      B_Tlong_pre = lag1(B_Tlong);
      B_Corp_pre = lag1(B_Corp);
      B_T3M_pre = lag1(B_T3M);
      B_HY_pre = lag1(B_HY);
      SPINDX_pre = lag1(SPINDX);
      rename date=trd_exctn_dt;
run;

*read trade data;
data tradeall;
      set bond.aggtradedata121516adjJF;
      if TRDG_MKT_CD="P1" then delete; *primary market;
      if cust=1;                    *customer trades;
      if rsc="P";                   *dealer acts as principal;
      if offering_amt*1000 lt 500000000 then sz=1;
      if offering_amt*1000 ge 500000000 and offering_amt*1000 lt 1000000000
      then sz=2;
      if offering_amt*1000 ge 1000000000 then sz=3;
      if TRD_EXCTN_DT gt dessdate then trans=1; else trans=0;
      drop trdsec trdhr trdmin trdday trdmonth trdyr trdg_mkt_cd _freq_
      issue_id Issuer_id rating rating2 X tempid effective_date
      amount_outstanding finOS enddt cus6 industry_group quant_sgn year;
run;

proc sort data=tradeall; by trd_exctn_dt; run;
data tradeall;
      merge tradeall (in=a) barclay (in=b);
      by trd_exctn_dt;
      if a and b;
run;

proc sort data=tradeall nodupkey out=tradeall;
      by cusip_id TRD_EXCTN_DT trdtime trid; run;
proc sort data=tradeall; by cusip_id TRD_EXCTN_DT exctn_tm; run;

*create variables for indicator variable regressions following Bessembinder,
Maxwell and Venkataraman (2006, JFE);
data trade1;
      set tradeall;
      by cusip_id trd_exctn_dt;
      *previous trade;
      lcusip_id=lag1(cusip_id);
      ltrd_exctn_dt=lag1(trd_exctn_dt);
      lexctn_tm = lag1(exctn_tm);
      lentrd_pr=lag1(entrd_pr);
```

```
      lentrd_vol_qt=lag1(entrd_vol_qt);
      lcust=lag1(cust);
      ltxn=lag(txn);
      lvolume_dol=lag1(volume_dol);
      lB_Tlong=lag1(B_Tlong);
      lB_Corp=lag1(B_corp);
      lB_T3M=lag1(B_T3m);
      lB_HY=lag1(B_HY);
      lSPINDX=lag1(SPINDX);
   *one-day lag of previous trade;
      lB_Tlong_pre=lag1(B_Tlong_pre);
      lB_Corp_pre=lag1(B_Corp_pre);
      lB_T3M_pre=lag1(B_T3m_pre);
      lB_HY_pre=lag1(B_HY_pre);
      lspindx_pre=lag1(SPINDX_pre);
      *elapsed day;
      if cusip_id =lcusip_id then elapdays = 1 + trd_exctn_dt -
      ltrd_exctn_dt;
      weightvar1 = 1/sqrt(elapdays);
      *new cusip;
      if first.cusip_id then do;
            ltrd_exctn_dt=.;
            lexctn_tm =.;
            lentrd_pr=.;
            lentrd_vol_qt=.;
            lcust=lag1(cust);
            ltxn=.;
            lvolume_dol=.;
            lB_Tlong=.;
            lB_corp=.;
            lB_T3M=.;
            lB_HY=.;
            lspindx=.;
            elapdays=.;
            weightvar1=.;
            lB_Tlong_pre=.;
            lB_Corp_pre=.;
            lB_T3M_pre=.;
            lB_HY_pre=.;
            lspindx_pre=.;
      end;
      format ltrd_exctn_dt date10.;
run;
data trade2;
      set trade1;
      t_ret = ((entrd_pr-lentrd_pr)/lentrd_pr)*100;
      t_chg = (entrd_pr-lentrd_pr);
      *adjacent trades on the same day then use one-day return;
      if (cusip_id=lcusip_id) and (trd_exctn_dt=ltrd_exctn_dt) then do;
            stockret = ((spindx-spindx_pre)/spindx_pre)*100;
            trsyret = ((B_Tlong-B_Tlong_pre)/B_Tlong_pre)*100;
            bondret = ((B_corp-B_corp_pre)/B_corp_pre)*100;
            Tbillret = ((B_T3m-B_T3m_pre)/B_T3m_pre)*100;
            HYret = ((B_HY-B_HY_pre)/B_HY_pre)*100;
      end;
 *adjacent trades are NOT on the same day then multiday return including both
 day s and day t;
```

```
      if (cusip_id=lcusip_id) and (trd_exctn_dt^=ltrd_exctn_dt) then do;
            stockret = ((spindx-lspindx_pre)/lspindx_pre)*100;
            trsyret = ((B_Tlong-lB_Tlong_pre)/lB_Tlong_pre)*100;
            bondret = ((B_corp-lB_corp_pre)/lB_corp_pre)*100;
            Tbillret = ((B_T3m-lB_T3m_pre)/lB_T3m_pre)*100;
            HYret = ((B_HY-lB_HY_pre)/lB_HY_pre)*100;
      end;
      TSret = trsyret - Tbillret;
      CSret = HYret - bondret;
      if (cusip_id^=lcusip_id) then do;
            stockret = .;
            trsyret = .;
            bondret = .;
            Tbillret = .;
            HYret = .;
            t_ret = .;
            t_chg = .;
            TSret = .;
            HYret = .;
      end;
      drop B_Tlong_pre B_Tlong B_Corp_pre B_Corp B_T3M_pre B_T3M B_HY_pre
      B_HY SPINDX_pre SPINDX lB_Tlong lB_corp lB_T3M lB_HY lspindx
      lB_Tlong_pre lB_Corp_pre lB_T3M_pre lB_HY_pre lspindx_pre HYret
      Tbillret;
run;

data trade3;
      set trade2;
      * Q(t);
      *******;
      Q = 0;
      if txn=1 then Q=-1;
            *if (RPT_SIDE_CD="B") and (cust=1) then Q=-1;
      if txn=-1 then Q=1;
            *if (RPT_SIDE_CD="S") and (cust=1) then Q=1;
      preQ = 0;
      if ltxn=1 then preQ=-1;
            *if (lRPT_SIDE_CD="B") and (lcust=1) then preQ=-1;
      if ltxn=-1 then preQ=1;
            *if (lRPT_SIDE_CD="S") and (lcust=1) then preQ=1;
      deltaQ = Q - preQ;
      if (cusip_id^=lcusip_id) then deltaQ=.;
            *deltaQtrace = deltaQ*trace;
      if (cusip_id^=lcusip_id) then preQ=.;
      * I(t);
      ******;
      if cust=0 then D=1; else D=0;
      if lcust=0 then preD=1; else preD=0;
      if (cusip_id^=lcusip_id) then preD=.;
      abslret=abs(t_ret);
      if abslret>10 then delete;
      *price change unlikely to reflect trading cost bounce, follows
         Bessembinder, Maxwell and Venkataraman (2006);
run;

proc means data=trade3 noprint;
      var cust;
```

```
      by cusip_id;
      output out=tradfreq n=numtrad;
run;

data trade4;
      merge trade3 (in=a) tradfreq (in=b);
      by cusip_id;
      if a and b;
      if numtrad>20; *keep bonds with more than 20 trades over sample period;
      keep cusip_id exctn_tm trd_exctn_dt entrd_pr entrd_vol_qt volume_dol
      txn dc rsc cust offering_amt highyield dessdate r144a trdsz sz trans
      stockret t_ret trsyret CSret TSret bondret deltaQ elapdays weightvar1
      age young offering_date;
run;

*create period variables;
data trade5;
      set trade4;
      trdyrmonth = year(TRD_EXCTN_DT)*12 + month(TRD_EXCTN_DT);
      MDY=mdy(month(TRD_EXCTN_DT),1,year(TRD_EXCTN_DT));format MDY MMYYn4.;
      yr=year(MDY);mo=month(MDY);
      *definition;
      if (yr=2006) or (yr=2007 and mo le 6) then pre=1;else pre=0;
      if (yr=2007 and mo ge 7) or (yr=2008) or (yr=2009 and mo le 4) then
      crisis=1;else crisis=0;
      if (yr=2009 and mo ge 5) or (yr=2010 and mo le 6) then post=1; else
      post=0;
      if (yr=2010 and mo ge 7) or (yr=2011) or (yr=2012) or (yr=2013) or
      (yr=2014 and mo le 3) then reg=1; else reg=0;
      if (yr=2014 and mo ge 4) or (yr in (2015,2016)) then volc=1; else
      volc=0;
      *deltaQ interaction;
            deltaq_volume_dol = deltaq*log(volume_dol);
            deltaq_1 = deltaq*pre;
            deltaq_2 = deltaq*crisis;
            deltaq_3 = deltaq*post;
            deltaq_4 = deltaq*reg;
            deltaq_5 = deltaq*volc;
      period=0;
      if (yr=2006) or (yr=2007 and mo le 6) then period=1;
      if (yr=2007 and mo ge 7) or (yr=2008) or (yr=2009 and mo le 4) then
      period=2;
      if (yr=2009 and mo ge 5) or (yr=2010 and mo le 6) then period=3;
      if (yr=2010 and mo ge 7) or (yr=2011) or (yr=2012) or (yr=2013) or
      (yr=2014 and mo le 3) then period=4;
      if (yr=2014 and mo ge 4) or (yr in (2015,2016)) then period=5;
      if period=0 then delete;
run;

*trading cost by period;
proc model data = trade5 noprint;
 parms b0 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11;
 t_ret = b0 + b2*trsyret + b3*stockret + b4*bondret + b5*TSret + b6*CSret +
         b7*deltaq + b8*deltaq_2 + b9*deltaq_3 + b10*deltaq_4 + b11*deltaq_5;
        fit t_ret/ gmm kernel = (bart,1,0) outest = temp1;
        weight weightvar1;
```

```
         instruments trsyret stockret bondret TSret CSret deltaq deltaq_2
        deltaq_3 deltaq_4 deltaq_5;
quit;

*trade size;
data trade5;
      set trade5;
      period=0;
      if volume_dol<100000 then trdsz=1;
      if volume_dol>=100000 and volume_dol<=1000000 then trdsz=2;
      if volume_dol>1000000 and volume_dol<=10000000 then trdsz=3;
      if volume_dol>10000000 then trdsz=4;
      if volume_dol<1000 then delete;
run;
proc sort data=trade5; by trdsz; run;
proc model data = trade5 noprint;
 parms b0 b2 b3 b4 b5 b6 b7 b8 b9 b10 b11;
 t_ret = b0 + b2*trsyret + b3*stockret + b4*bondret + b5*TSret + b6*CSret +
              b7*deltaq + b8*deltaq_2 + b9*deltaq_3 + b10*deltaq_4 +
              b11*deltaq_5;
             fit t_ret/ gmm kernel = (bart,1,0) outest = temp1;
             weight weightvar1;
             instruments trsyret stockret bondret TSret CSret deltaq deltaq_2
                deltaq_3 deltaq_4 deltaq_5;
              by trdsz;
quit;

* volume distribution;
proc sort data=trade5; by period trdsz; run;
proc univariate data=trade5 noprint;
      var volume_dol;
      by period;
      output out=test1 sum=volume_all;
run;
proc univariate data=trade5 noprint;
      var volume_dol;
      by period trdsz;
      output out=test2 mean=trdsz_mn median=trdsz_md sum=volume_trdsz;
run;
proc sort data=test2; by period; run;
proc sort data=test1; by period; run;
data test3;
      merge test2 (in=a) test1 (in=b);
      by period;
      if a and b;
      volume_pct=(volume_trdsz/volume_all)*100;
run;

* By Credit rating - use highyield;
* Issue Size use sz;
*Bond Age - use age;
      age=trd_exctn_dt - offering_date;
      if age <= 365 then young=1; else young=0;
* Click versus call;
      if ((highyield=0) and (sz=3) and (young=1) and (trdsz^=4)) then elec=1;
      else elec=0;
      if highyield=. or sz=. or young=. or trdsz=. then delete;
```