

```

%global FIRST_ROW LAST_ROW PULSER_MAX_VALUE PULSER_COUNT;
*%let PULSER_COUNT=2;

%macro _analyzeNEW;
/*options nosource nosource2 nonotes;*/
%put loop: &LOOP_NUMBER;
%put first: &FIRST_ROW ;
%put last: &LAST_ROW ;

data WORK.temp;
    set WORK.QUERY_FOR_TSDTIMESERIESDATA (firstobs=&FIRST_ROW. obs=&LAST_ROW.);
run;

data _null_;
    set WORK.temp;
/*    length puls_lista $ 5000;*/
    where date="&Period_MIN"d;

    array var_list(*) puls1-puls&Postfreeze_period.;
    loop = &Postfreeze_period.;
    pulser_max_value=0;

/* Ta reda på högsta puls-värdet */
do while (loop>0);
    if var_list(loop) ne . then do;
        pulser_max_value = loop;
        LEAVE;
    end;
    else
        loop=loop-1;
end;

/* Skriv aktuellt loop-värde till makrovar */
call symput('PULSER_MAX_VALUE', put(pulser_max_value, best.));
putlog 'Max pulsvärde: ' pulser_max_value;

```

```

/* Om max puls > 0 kör vi _analyze*/
if pulser_max_value > 0 then
  call execute('%_pulse');
else putlog 'Max pulsvärde är 0. Ingen ARIMA.';

run;

%mend _analyzeNEW;

%macro _puls_lista(antal);
  %do i = 1 %to &antal;
    puls&i
  %end;
%mend _puls_lista;

%macro _pulse;
/*options nosource nosource2 nonotes;*/

ods trace on;
ods output ParameterEstimates = params_2;

proc arima data=WORK.TEMP plots=none;
  identify
    noprint
    var=Change_frequency stationarity=(adf=1 dlag=2) crosscorr=( %_puls_lista(&PULSER_MAX_VALUE) );
  estimate input=( %_puls_lista(&PULSER_MAX_VALUE) ) outest=estimat_2;
run;
%mend _pulse;
%macro _update_ARIMAX;
options nosource nosource2 nonotes;
%put &Candidate;
PROC SQL;
  CREATE TABLE WORK.QUERY_FOR_TEMP AS
  SELECT DISTINCT t1.Candidate_ID,
    t2.Variable,
    t2.Estimate,

```

```

        t2.Probt
    FROM WORK.TEMP t1
        CROSS JOIN WORK.PARAMS_2 t2;
QUIT;
%mend _update_ARIMAX;

%macro _append_ARIMAX;
options nosource nosource2 nonotes;
PROC SQL;
INSERT INTO WORK.ARIMAX
SELECT * FROM WORK.QUERY_FOR_TEMP;
QUIT;
%mend _append_ARIMAX;
%macro _create_ARIMAX;
PROC SQL noprint;
CREATE TABLE WORK.ARIMAX LIKE WORK.QUERY_FOR_TEMP;
QUIT;
%mend _create_ARIMAX;

%let startTime=%sysfunc(time(),best.);
%put Startar %sysfunc(time(),time9.4);

/*options nosource nosource2 nonotes;*/
data _null_;
    length first_row loop_number 8;

    /* RETAIN gör så att dessa fält inte blankas mellan varje radläsning */
    retain first_row loop_number 0;

    /* Läs stora tabellen med alla dagar per TicketID. */
    set WORK.QUERY_FOR_TSDSTIMESERIESDATA;

    /* Med BY kan vi använda FIRST. och LAST. */
    by Candidate_ID;

```

```

/* Om första raden i aktuell TicketID så spara undan värdet. */
if FIRST.Candidate_ID then
    first_row = _N_;

/* Om sista raden för aktuell TicketID sparar vi värden. */
if LAST.Candidate_ID then do;
    loop_number+1;

    /* Putta ut first/last och strippa bort inledande/avslutande blanka med SymputX */
    call symputX('first_row', put(first_row, best.));
    call symputX('last_row', put(_N_, best.));
    call symputX('loop_number', put(loop_number, best.));

    call symput('Candidate',Candidate_ID);

    call execute('%_analyzeNEW');

    call execute ('%_update_ARIMAX');

    if loop_number = 1 then call execute('%_create_ARIMAX');

    call execute ('%_append_ARIMAX');

end;

run;

%put Slutar %sysfunc(time()),time9.4);
%let stopTime=%sysfunc(time(),best.);
data _null_;
    t=&stopTime - &startTime;
    format t time9.4 ;
    put 'Jobbet tog: ' t;
run;

```