```
proc import file = "C:\Users\dryan\OneDrive\Documents\Sts499\2014s1.csv" out
= polls2014
dbms = csv replace;
datarow = 2;
GetNames = yes;
run;

proc import file = "C:\Users\dryan\OneDrive\Documents\Sts499\2016s2.csv"
out = polls2016
dbms = csv replace;
datarow =2;
GetNames = yes;
run;

proc import file = "C:\Users\dryan\OneDrive\Documents\Sts499\2018s2.csv" out
= polls2018
dbms = csv replace;
datarow =2;
GetNames = yes;
run;



data polls14;
set polls2014;
rename _538_Grade__2014_ = grade14;
rename Pollster__2014_ = name;
run;

data polls16;
set polls2016;
rename _538_Grade__2016_ = grade16;
rename Pollster__2016_ = name;
run;

data polls18;
set polls2018;
rename Pollster__2018_ = name;
rename _538_Grade__2018_ = grade18;
run;

data polls4;
length name $58;
set polls14;
if grade14 ne ' ' then gpa14 = (grade14 = "A+")*4.3 + (grade14 = "A")*4 +
(grade14 = "A-")*3.7 + (grade14 = "B+")*3.3 + (grade14 = "B")*3 + (grade14 =
"B-")*2.7 + (grade14 = "C+")*2.3 + (grade14 = "C")*2 + (grade14 = "C-")*1.7 +
(grade14 = "D+")*1.3 + (grade14 = "D")*1 + (grade14 = "F")*0;
else gpa14 = .;
run;

data polls6;
length name $58;
set polls16;
if grade16 ne ' ' then gpa16 = (grade16 = "A+")*4.3 + (grade16 = "A")*4 +
(grade16 = "A-")*3.7 + (grade16 = "B+")*3.3 + (grade16 = "B")*3 + (grade16 =
"B-")*2.7 + (grade16 = "C+")*2.3 + (grade16 = "C")*2 + (grade16 = "C-")*1.7 +
(grade16 = "D+")*1.3 + (grade16 = "D")*1 + (grade16 = "F")*0;
else gpa16 = .;
run;

data polls8;
length name $58;
set polls18;
if grade18 ne ' ' then gpa18 = (grade18 = "A+")*4.3 + (grade18 = "A")*4 +
(grade18 = "A-")*3.7 + (grade18 = "B+")*3.3 + (grade18 = "B")*3 + (grade18 =
"B-")*2.7 + (grade18 = "C+")*2.3 + (grade18 = "C")*2 + (grade18 = "C-")*1.7 +
(grade18 = "D+")*1.3 + (grade18 = "D")*1 + (grade18 = "F")*0;
else gpa18 = .;
run;


proc import file = "C:\Users\dryan\OneDrive\Documents\generalpollsa.csv" out
= general dbms = csv;
datarow =2;
getnames = Y;
run;

proc sort data = polls4;
by name;
run;

proc sort data = polls6;
by name;
run;

proc sort data = polls8;
by name;
run;

proc sort data = polls14;
by name;
run;

proc sort data = polls16;
by name;
run;

proc sort data = polls18;
by name;
run;


data test;
merge polls14 polls16 polls18;
by name;
run;



data test2 testx;
merge polls4 (in = in4)
polls6             (in = in6)
```

```
polls8            (in = in8);
by name;
output test2;
if(in4 & in6 & in8) then output testx;
run;

/*nmiss counts the number of missing inputs */
data testy;
set test2;
if nmiss( grade14, grade16, grade18) = 0;
run;


data general2;
length pollster $58;
set general;
where type = "polls-only";
rename pollster = name;
run;

proc sort data = general2;
by name;
run;

data g2t2;
merge general2 test2;
by name;
/*
raw_diff first calulated ----------------------------------------------------
-----------------------------------------------------
*/
raw_diff = rawpoll_clinton - rawpoll_trump;
run;
/*++++++++++++++++++++++++++++++++++++++++++++++++++*/
proc sort data = g2t2;
by name;
run;

ods output summary = diff_means;
proc means data = g2t2;
var raw_diff;
class state;
run;

proc sort data = g2t2 out = g2t2_state;
by state;
run;

data means_merged;
merge g2t2_state diff_means;
by state;
state_Zscore = (raw_diff - raw_diff_mean)/(raw_diff_stdDev);
run;

data means_merged;
set means_merged;
c1416 = gpa16 - gpa14;
```

```
c1618 = gpa18 - gpa16;
run;

proc import file = "C:\Users\dryan\OneDrive\Documents\Sts499\2016states.csv"
out = states_results
dbms = csv replace;
datarow = 2;
getnames = y;
run;

data states_results;
set states_results;
rename states = state;
run;

data states_results;
length state $58;
set states_results;
rename Hillary_Clinton = clinton_state_result;
rename Donald_Trump = trump_state_result;
run;

proc sort data = means_merged;
by state;
run;

proc sort data = states_results;
by state;
run;

data means_merged;
length state $58;
set means_merged;
run;

data merged_results;
merge means_merged states_results;
by state;
run;

data merged_results;
set merged_results;
/*
state_diff first calulated --------------------------------------------------
----------------
*/
state_diff = clinton_state_result - trump_state_result;
raw_state_diff = raw_diff - state_diff;
run;

/*+++++++++++++++++++++++++++++++++++++++++++*/
ods output summary = Zscore_states;
proc means data = merged_results;
var raw_state_diff;
class state;
run;
```

```
ods output summary = herding_ratio;
proc means data = merged_results mean nonobs;
var raw_diff;
class state;
run;

data herding_ratio;
length state $58;
set herding_ratio;
/*----------------
first time avg_state_diff
*/
rename raw_diff_Mean = avg_state_diff;
run;


data Zscore_states;
length state $58;
set Zscore_states;
rename NObs = NObs_1;
rename raw_state_diff_N = raw_state_diff_N_1;
rename raw_state_diff_Mean = raw_state_diff_Mean_1;
rename raw_state_diff_StdDev = raw_state_diff_StdDev_1;
rename raw_state_diff_Min = raw_state_diff_Min_1;
rename raw_state_diff_Max = raw_state_diff_Max_1;
run;





proc sort data= merged_results;
by state;
run;

data merged_results;
length state $58;
set merged_results;
run;

data means_merged_Zscore;
length state $58;
merge merged_results Zscore_states;
by state;
run;

data means_merged_Zscore;
set means_merged_Zscore;
if(raw_diff >0)  then predicted_winner = "clinton";
if(raw_diff < 0) then predicted_winner = "trump";
if(raw_diff = 0) then predicted_winner = "tie";
run;

proc sort data = means_merged_Zscore;
by state;
run;
```

```
proc freq data = means_merged_Zscore;
tables predicted_winner;
by state;
run;


data means_merged_Zscore;
length state $58;
set means_merged_Zscore;
run;

data means_merged_Zscore;
merge means_merged_Zscore herding_ratio;
by state;
abs_difference = abs(raw_diff - avg_state_diff);
run;

ods output summary = difference_in_state;
proc means data = means_merged_Zscore mean ;
class state;
var abs_difference;
run;

data means_merged_Zscore;
merge means_merged_Zscore difference_in_state;
by state;
run;


data means_merged_Zscore;
set means_merged_Zscore;
if(predicted_winner = "trump") then predicted_num = 1;
if(predicted_winner = "clinton") then predicted_num = 0;
if(predicted_winner = "tie") then predicted_num = 0.5;
/*-------------------------------- ratio_good first time --------------------
-------
raw_diff = indivudual poll result
avg_state_diff = average estimate per state
abs_difference = absolute value ((indivudual poll) - (average state
estimate))
*/
ratio_good = (raw_diff - state_diff)/abs_difference_Mean;
run;

proc sgplot data = means_merged_Zscore;
scatter x = gpa16 y = ratio_good/ colorresponse = predicted_num colormodel =
(CX0000FF CX000000 CXFF0000);
/*yaxis max = 100;*/
gradlegend / title = "Predicted Winner by Party Color";
run;
/*++++++++++++++++++++++++++++++++++*/
data time_comparison;
set means_merged_Zscore;
diff_time = forecastdate - enddate;
abs_ratio = abs(ratio_good);
run;
```

```
proc sgplot data = time_comparison;
scatter x = diff_time y = abs_ratio;
reg x = diff_time y = abs_ratio;
title "time vs ratio";
run;

data time_comparison;
set time_comparison;
/*
first time polling error ----------------------------------------------------
------------------------------
*/
polling_error = raw_diff - state_diff;
abs_polling_error = abs(polling_error);
run;

data time_comparison;
set time_comparison;
rounded_time_7 = round(diff_time, 7);
rounded_time_3 = round(diff_time, 3);
run;
/*++++++++++++++++++++++++++++++++++*/

ods output summary = time_7_all_states;
proc means data = time_comparison;
class rounded_time_7;
var abs_polling_error;
run;


proc sort data = time_comparison;
by abs_polling_error;
run;

data time_7_all_states;
set time_7_all_states;
rename abs_polling_error_Mean = abs_polling_error;
run;

data time_7_all_states;
set time_7_all_states;
rename abs_polling_error = abs_polling_error_Mean;
run;

proc means data = time_comparison;
class state;
var abs_polling_error;
run;


data west_virginia;
set time_comparison;
where (state = "West Virginia");
run;

data penn;
set time_comparison;
```

```
where (state = "Pennsylvania");
run;

proc means data= penn;
class name;
var polling_error;
run;
/*++++++++++++++++++++++++++++++++++*/
proc sgplot data = west_virginia;
scatter x = diff_time y = polling_error /colorresponse = gpa16 colormodel =
(R O Y P B G ) ;
Title  " polling error by date & color by gpa16 in WV";
run;

proc means data = west_virginia;
class gpa16;
var polling_error;
run;

data USA;
set time_comparison;
where(state = "U.S.");
run;

proc sort data = USA;
by name;
run;

proc means data = USA;
var polling_error;
class name;
run;


proc sort data = USA;
by rounded_time_7 name;
run;
/****************************************************************************
************/
data USA_7;
set USA;
by rounded_time_7 name;
if first.name;
run;

proc sort data = USA_7;
by name;
run;

proc sgplot data = USA_7;
scatter x = rounded_time_7 y = ratio_good;
yaxis max = 6;
Title ratio;
run;

proc means data = USA_7;
var polling_error;
```

```
class rounded_time_7;
run;
ods output summary = usa_abs_polling_error;
proc means data = usa_7 mean nonobs;
var abs_polling_error;
class rounded_time_7;
run;
/*+++++++++++++++++++++++++++++++++*/
data usa_abs_polling_error;
set usa_abs_polling_error;
week =rounded_time_7/7;
run;

proc reg data = usa_abs_polling_error;
model abs_polling_error_mean = week;
run;

proc sgplot data = usa_7;
scatter x = diff_time y = polling_error;
run;

proc means data = usa_7;
class name;
var polling_error;
run;



proc means data = USA_7;
class rounded_time_7;
var abs_polling_error;
run;

data usa_abs_polling_error;
length rounded_time_7 8;
set usa_abs_polling_error;
run;

proc sort data = usa_abs_polling_error;
by rounded_time_7;
run;

proc sort data = usa_7;
by rounded_time_7;
run;


data usa_merge;
merge usa_7 usa_abs_polling_error;
by rounded_time_7;
run;
/*+++++++++++++++++++++++++++++++++*/
data usa_merge;
set usa_merge;
/*difference in individual abs error versus mean abs error per week*/
diff_abs_error = abs_polling_error - abs_polling_error_Mean;
run;
```

```
data usa_merge;
set usa_merge;
abs_diff_abs_error = abs(diff_abs_error);
run;

ods output summary = usa_herding;
proc means data = usa_merge;
class rounded_time_7;
var abs_diff_abs_error;
run;

proc sgplot data = usa_herding;
scatter x = rounded_time_7 y = abs_diff_abs_error_Mean;
tItLE "please";
run;


data usa_herding;
set usa_herding;
week = rounded_time_7/7;
run;

proc reg data = usa_herding;
model abs_diff_abs_error_Mean = week;
run;


proc sgplot data = usa_7;
scatter x = rounded_time_7 y = ratio_good;
run;

/*++++++++++++++++++++++++++++++++++*/


proc sgplot data = usa_7;
scatter x = diff_time y = polling_error;
run;


ods output summary = reg;
proc means data = usa_7 mean;
class rounded_time_7;
var abs_polling_error;
run;

data reg;
set reg;
week = rounded_time_7/7;
run;


proc reg data = reg;
model abs_polling_error_mean = week;
run;
```

```
/*++++++++++++++++++++++++++++++++++*/



/*

proc means data = west_virginia;
class name;
var polling_error;
run;

data west_virginia;
set west_virginia;
if(name = "Garin-Hart-Yang Research Group") then c = 1;
if(name = "Global Strategy Group") then c = 2;
if(name = "Google Consumer Surveys") then c =3;
if(name = "Ipsos") then c =4;
if(name = "Just Win Strategies") then c =5;
if(name = "Orion Strategies") then c =6;
if(name = "Public Policy Polling") then c =7;
if(name = "R.L. Repass & Partners") then c =8;
if(name = "SurveyMonkey") then c = 9;
if(name = "YouGov") then c =10;
run;

proc sort data = west_virginia;
by name;
run;


proc sgplot data = west_virginia;
scatter x = diff_time y = polling_error /colorresponse = c colormodel = (R O
Y O B G BR WH P BL ) ;
run;



libname outdat "C:\Users\dryan\OneDrive\Documents\Sts499";
data outdat.new;
set time_comparison;
run;

data time_comparison;
set means_merged_Zscore;
diff_time = forecastdate - enddate;
abs_ratio = abs(ratio_good);
run;

/****************************************************************************
******************************
*****************************************************************************
******************************
*****************************************************************************
*****************************/


data simplify;
```

```
set means_merged_Zscore;
denominator = (rawpoll_clinton + rawpoll_trump)/100;
trump_p = rawpoll_trump/100;
clinton_p = rawpoll_clinton/100;
run;

data simplify;
set simplify;
denom = clinton_state_result + trump_state_result;
state_p = (trump_state_result/100)/(denom/100);
run;

data simplify;
set simplify;
porp_trump = trump_p/denominator;
porp_clinton = clinton_p/denominator;
run;

data simplify;
set simplify;
check = porp_clinton + porp_trump;
run;

ods output summary = p;
proc means data = simplify mean ;
class state;
var porp_trump samplesize;
run;
/*++++++++++++++++++++++++++++++++++*/
data p;
rename porp_trump_Mean = p_state_avg;
set p;
run;
/**** run below code to reset porp ****/
data porp;
merge simplify p;
by state;
run;

data porp;
set porp;
diff_state_p = porp_trump - p_state_avg;
run;

data porp;
set porp;
n_state = nobs;
run;

/*** end of the reset for porp ***/
/*data porp;
set porp;
z_den = sqrt((p_state_avg*(1- p_state_avg))/n_state);
run;
--> used the wrong sample size and true population porportion above
*/
data po;
```

```
set porp;
run;
/****************/
data po;
set po;
elec_porp = trump_state_result/(trump_state_result + clinton_state_result);
run;

data po;
set po;
z_den = sqrt((elec_porp*(1- elec_porp))/samplesize);
run;



/***************/

data po;
set po;
z_porp = (porp_trump - elec_porp)/z_den;
run;

proc means data = po;
class state;
var z_porp;
run;
ods output summary = samp;
proc means data = po mean nonobs;
class state;
var samplesize;
run;

data po2;
merge po samp;
by state;
run;
*/ below p_state_avg is the average estimate per state , and n is the average
sample size per state */
data po2;
/*++++++++++++++++++++++++++++++++++*/
data po2;
set po2;
se = sqrt((p_state_avg*(1-p_state_avg))/samplesize_mean);
run;

data po2;
set po2;
diff_time = forecastdate - enddate;
run;

data po2;
set po2;
if(gpa16 >= 3.3) then new_grade = 'A/B';
if(gpa16 >= 1.7 & gpa16 < 3.3) then new_grade = 'B/C';
run;
/*++++++++++++++++++++++++++++++++++*/
```

```
/*__________________________________________________________________________*
/
data pa;
set po2;
upper = p_state_avg + se;
lower = p_state_avg - se;
where state = 'Pennsylvania';
run;

data pa;
set pa;
if (porp_trump > lower & porp_trump < upper) then check = 1;
else check = 0;
run;

proc freq data = pa;
tables check;
title 'PA se check';
run;

data Wisc;
set po2;
upper = p_state_avg + se;
lower = p_state_avg - se;
where state = 'Wisconsin';
run;

data Wisc;
set Wisc;
if (porp_trump > lower & porp_trump < upper) then check = 1;
else check = 0;
run;

proc freq data = wisc;
tables check;
title 'Wisc se check';
run;

data mich;
set po2;
upper = p_state_avg + se;
lower = p_state_avg - se;
where state = 'Michigan';
run;

data mich;
set mich;
if (porp_trump > lower & porp_trump < upper) then check = 1;
else check = 0;
run;

proc freq data = mich;
tables check;
title 'michigan se check';
run;
/*++++++++++++++++++++++++++++++++++*/
data us;
```

```
set po2;
upper = p_state_avg + se;
lower = p_state_avg - se;
where state = 'U.S.';
run;

data us;
set us;
if (porp_trump > lower & porp_trump < upper) then check = 1;
else check = 0;
run;

proc freq data = us;
tables check;
title 'us se check';
run;

proc sgplot data = pa ;
scatter x = diff_time y = porp_trump/ group = new_grade;
band x = diff_time upper = upper lower = lower / transparency = 0.5;
lineparm x= state_p y = state_p slope = 0;
xaxis reverse;
title 'pa grades';
run;

proc ttest data = pa;
class new_grade;
var porp_trump;
run;



proc freq data = po2;
tables grade16;
run;
/*------------------------------------------------*/
/*++++++++++++++++++++++++++++++++++*/
data po3;
set po2;
where new_grade = 'A/B' | new_grade = 'B/C';
run;

data pa;
set pa;
two_week = round(diff_time,14);
run;



data pa;
set pa;
month = round(diff_time,30);
if(month > 90) then month = 90;
run;

proc freq data = pa;
tables month;
```

```
run;

ods output summary = pa_mean;
proc means data = pa mean nonobs;
class month;
var porp_trump samplesize;
run;

data pa_mean;
set pa_mean;
rename samplesize_Mean = samplesize_Mean_month;
rename porp_trump_Mean = porp_trump_Mean_month;
run;
/*+++++++++++++++++++++++++++++++++++*/
proc sort data = pa;
by month;
run;

data pa1;
merge pa pa_mean;
by month;
run;

data pa1;
set pa1;
se_month = sqrt((porp_trump_Mean_month*(1-
porp_trump_Mean_month))/samplesize_Mean_month);
upper_month = porp_trump_Mean_month + se_month;
lower_month = porp_trump_mean_month - se_month;
run;
/*
proc sgplot data = pa1 ;
scatter x = diff_time y = porp_trump/ group = new_grade;
band x = diff_time upper = upper_month lower = lower_month / transparency =
0.5;
lineparm x= state_p y = state_p slope = 0;
xaxis reverse;
title 'pa1 grades loess';
loess x = diff_time y = porp_trump/nomarkers group = new_grade;
run;
*/
proc sgplot data = pa1 ;
where new_grade in("A/B","B/C") & diff_time <101;
scatter x = diff_time y = porp_trump/group = new_grade;
band x = diff_time upper = upper_month lower = lower_month / transparency =
0.5;
lineparm x= state_p y = state_p slope = 0;
xaxis reverse;
title 'pa1 grades loess';
loess x = diff_time y = porp_trump/ nomarkers;* group = new_grade;
run;

data pa1;
set pa1;
if(porp_trump > lower_month & porp_trump < upper_month) then check_month = 1;
else check_month = 0;
run;
```

```
proc means data = pa1;
class month new_grade;
types month month*new_grade;
var check_month;
run;
/*++++++++++++++++++++++++++++++++++*/


proc means data = pa1;
class month;
var check_month;
run;

proc freq data = pa1;
tables two_week;
run;

data pa1;
set pa1;
two_week = two_week/14;
run;
ods output summary = pa_two_weeks;
proc means data = pa1 mean nonobs;
class two_week;
var porp_trump;
run;

proc sort data = pa1;
by two_week;
run;

data pa_two_weeks;
set pa_two_weeks;
rename porp_trump_mean = porp_trump_mean_two_weeks;
run;
data pa2;
merge pa1 pa_two_weeks;
by two_week;
abs_diff = abs(porp_trump_mean_two_weeks - porp_trump);
run;



ods output summary = pa2_means;
proc means data = pa2 mean nonobs;
class two_week;
var abs_diff;
run;

proc sgplot data = pa2_means;
scatter x = two_week y = abs_diff_mean;
xaxis reverse;
run;

proc loess data = pa2_means;
model  abs_diff_mean = two_week;
```

```
where two_week < 8;
run;
/*+++++++++++++++++++++++++++++++++*/


data po3;
set po3;
two_week = round(diff_time, 14);
run;

ods output summary = all;
proc means data = po3 mean nonobs;
class two_week state;
var porp_trump;
run;

data all;
set all;
rename porp_trump_Mean = porp_trump_Mean_all;
run;

proc sort data = po3;
by state two_week;
run;

proc sort data = all;
by state two_week;
run;

data all2;
merge po3 all;
by state two_week;
abs_all_diff = abs(porp_trump_mean_all - porp_trump);
run;

proc sort data = all2;
by state;
run;

proc sgplot data = all2;
scatter x = diff_time y = abs_all_diff/ group = state;
xaxis reverse;
where (state = "Pennsylvania" | state = "Wisconsin" | state = "Michigan") &
diff_time < 101;
loess x = diff_time y = abs_all_diff / nomarkers group = state;
run;

proc sgplot data = all2;
scatter x = diff_time y = abs_all_diff/ group = new_grade;
xaxis reverse;
where state = "Wisconsin"  & diff_time < 101;
loess x = diff_time y = abs_all_diff / nomarkers group = new_grade;
run;

/*+++++++++++++++++++++++++++++++++*/
proc sort data = all2;
by abs_all_diff;
```

```
run;

ods graphics on;

proc loess data = all2;
model abs_all_diff = diff_time;
where new_grade = 'A/B';
title 'A/B all states';
run;
/*ods trace on;*/
ods output position = vars;
proc contents data = all2 varnum;
run;
/*ods trace off;*/

*proc export outfile = 'C:\Users\dryan\OneDrive\Documents\Sts499\var.xlsx'
data = vars dbms = xlsx replace;
*run;


data all2;
set all2;
se_month = sqrt((porp_trump_Mean_month*(1-
porp_trump_Mean_month))/samplesize_Mean_month);
upper_month = porp_trump_Mean_month + se_month;
lower_month = porp_trump_mean_month - se_month;
run;

data all2;
set all2;
month = round(diff_time, 30);
run;
ods output summary = g;
proc means data = all2 mean nonobs;
class state two_week;
var porp_trump samplesize;
run;

data g;
set g;
rename porp_trump_mean = porp_trump_mean_g;
rename samplesize_mean = samplesize_mean_g;
run;

proc sort data = g;
by state two_week;
run;

proc sort data = all2;
by state two_week;
run;


data all3;
merge all2 g;
by state two_week;
run;
```

```
data all3;
set all3;
se_two_week = sqrt((porp_trump_Mean_g*(1-
porp_trump_Mean_g))/samplesize_Mean_g);
upper_two_week = porp_trump_Mean_g + se_two_week;
lower_two_week = porp_trump_mean_g - se_two_week;
run;

data all3;
set all3;
porp_diff_two = abs(porp_trump_mean_g - porp_trump);
run;

proc sort data = all3;
by state;
run;
*run sort above first;
*ods pdf file =
'C:\Users\dryan\OneDrive\Documents\Sts499\statePlotsColor.pdf';
*-----------------------------------------------------------
-----------------------------------------------------------
-----------------------------------------------------------
-----------------------------------------------------------
-----------------------------------------------------------;
proc sgplot data = all3 ;
where new_grade in("A/B","B/C") & diff_time <101;
scatter x = diff_time y = porp_trump/group = new_grade;
*band x = diff_time upper = upper_two_week lower = lower_two_week /
transparency = 0.5;
lineparm x= state_p y = state_p slope = 0;
xaxis reverse;
title 'all states loess';
loess x = diff_time y = porp_trump/ nomarkers group = new_grade;
by state;
run;
*ods pdf close;
*rtf - rich text file can be outputted instead of PDf - opens in word ;
data alabama;
set all3;
where state = "Alabama";
run;

data alabama;
set alabama;
if name = "SurveyMonkey" then ind = "sm" ;
else ind = "all";
run;

proc sgplot data = alabama ;
where new_grade in("A/B","B/C") & diff_time <101;
scatter x = diff_time y = porp_trump/group = ind;
*band x = diff_time upper = upper_two_week lower = lower_two_week /
transparency = 0.5;
lineparm x= state_p y = state_p slope = 0;
xaxis reverse;
title 'all states loess';
```

```
loess x = diff_time y = porp_trump/ nomarkers group = ind;
by state;
run;


proc sort data = all3;
by new_grade;
run;
*!!!!!!!!!!!!!!! the below code resets results_kansas and should only be run
in a chunk with everything below;
proc loess data = all3;
model porp_trump = diff_time/ residual;
ods output OutputStatistics = results_Kansas;
by new_grade;
where state = 'Kansas';
run;

proc sort data = results_kansas;
by new_grade diff_time;
run;

data results_Kansas;
set results_Kansas;
week = round(diff_time, 7);
run;
data kansas;
set all3;
where state = 'Kansas';
run;

data kansas_ab;
set results_Kansas;
where new_grade = 'A/B';
rename DepVar = DepVar_ab;
rename pred = pred_ab;
run;
ods output summary = kansas_ab1;
proc means data = kansas_ab mean nonobs;
class week;
var pred_ab;
run;

data kansas_bc;
set results_Kansas;
where new_grade = 'B/C';
rename DepVar = DepVar_bc;
rename pred = pred_bc;
run;
ods output summary = kansas_bc1;
proc means data = kansas_bc mean nonobs;
class week;
var pred_bc;
run;

data kansas_all;
merge kansas_ab1 kansas_bc1;
by week;
```

```
pred_ab_Mean = pred_ab_Mean *100;
pred_bc_mean = pred_bc_mean *100;
diff = pred_ab_mean - pred_bc_mean;
run;

proc sgplot data = kansas_all;
scatter x= week y= diff;
xaxis reverse values = (0 to 98 by 7);
*title 'kansas diff in loess points by grade and by week';
run;

proc glm data = kansas_all;
model diff = week week*week;
run;

proc sgplot data = kansas;
series x = diff_time y = porp_trump/group = new_grade groupLC = name;
*transparency=0.7 lineattrs=(pattern=solid);
*tip = (new_grade name);
xaxis reverse;
run;
 data kansas;
 set kansas;
 pollster = name;
 grade1 = new_grade;
 pp = porp_trump;
 week = round(diff_time,7);
 run;

 proc sort data = kansas;
 by pp;
 run;


data illinois;
set all3;
where state = 'Illinois';
run;

proc sgplot data = illinois ;
where new_grade in("A/B","B/C") & diff_time <101;
scatter x = diff_time y = porp_trump/group = new_grade;
band x = diff_time upper = upper_two_week lower = lower_two_week /
transparency = 0.5;
lineparm x= state_p y = state_p slope = 0;
xaxis reverse;
title 'all states loess';
loess x = diff_time y = porp_trump/ nomarkers group = new_grade;
run;

proc sort data = illinois;
by name;
run;

data illinois;
set illinois;
pp = porp_trump * 100;
```

```
pollster = name;
time_diff = diff_time;
porp_trump_mean_g1 = porp_trump_mean_g;
grade1 = new_grade;
run;

proc sort data = illinois;
by pollster time_diff;
run;

proc sgplot data = illinois;
scatter x = diff_time y = abs_all_diff;
where pollster = 'Ipsos';
xaxis reverse;
run;

proc ttest data = kansas;
class new_grade;
var porp_trump;
run;

proc sort data = kansas;
by pollster;
run;

*make sure to sort by pollster before running;
proc mixed data = kansas;
class new_grade pollster;
model porp_trump = new_grade/ solution;
*repeated/ subject = pollster rcorr;
random int/ subject= pollster vcorr;
run;

proc sort data = all3;
by state name;
run;

ods output summary = mixed;
proc mixed data = all3;
class new_grade name;
model porp_trump = new_grade / solution;
random int / subject = name vcorr;
by state;
where diff_time > 40 & diff_time < 80;
run;




data oregon;
set all3;
where state = "Oregon";
run;

data oregon;
set oregon;
pp = porp_trump * 100;
```

```
pollster = name;
time_diff = diff_time;
porp_trump_mean_g1 = porp_trump_mean_g;
grade1 = new_grade;
run;

proc sort data = oregon;
by pp;
run;

proc freq data = oregon;
tables pollster;
run;

data alabama;
set all3;
where state = 'Alabama';
run;

proc mixed data = alabama;
class ind name;
model porp_trump = ind / solution;
random int / subject = name vcorr;
run;

proc sort data = all3;
by state two_week name;
run;


data all4;
set all3;
by state two_week name;
if first.name;
run;
/*  doea all 50 states
ods pdf file =
'C:\Users\dryan\OneDrive\Documents\Sts499\loess_CLM_allStates.pdf';
proc sgplot data = all4;
where new_grade in("A/B","B/C") & diff_time <101;
scatter x = diff_time y = porp_trump/group = new_grade;
*band x = diff_time upper = upper_two_week lower = lower_two_week /
transparency = 0.5;
lineparm x= state_p y = state_p slope = 0;
xaxis reverse;
title 'all states loess independant observations';
loess x = diff_time y = porp_trump/clm nomarkers group = new_grade;
by state;
run;
ods pdf close;
*/

data arizona;
set all3;
where state = 'Arizona';
run;
```

```
ods output summary = arizona1;
proc means data = arizona mean nonobs;
class two_week new_grade;
var porp_trump;
run;

data arizona2;
set arizona1;
where new_grade = 'A/B';
run;

data arizona3;
set arizona1;
where new_grade = 'B/C';
rename new_grade = new_grade1;
run;

data arizona4;
merge arizona3 arizona2;
by two_week;
run;

proc sort data = all3;
by state name descending diff_time;
run;

*sphagetti plots ____________________________________________________;
*groupLC group line color - different lines and colors for pollsters by
grade;
*LP = line pattern    MC - marker color      MS - marker symbol markers (in
blue) put the points on the graph;
*ods pdf file = 'C:\Users\dryan\OneDrive\Documents\Sts499\spaghetti.pdf';
proc sgplot data = all3 ;
where new_grade in("A/B","B/C") & diff_time <101;
series x = diff_time y = porp_trump/group = name groupLC = new_grade markers
groupLP = new_grade groupMC = name groupMS = new_grade;
*band x = diff_time upper = upper_two_week lower = lower_two_week /
transparency = 0.5;
lineparm x= state_p y = state_p slope = 0;
xaxis reverse;
title 'spaghetti plot';
*loess x = diff_time y = porp_trump/ nomarkers group = new_grade;
by state;
run;
*ods pdf close;

proc means data = all3;
class state two_week new_grade;
var porp_trump;

*________________________________________________ (10/29);

proc sort data = all3;
by state new_grade;
run;
```

```
proc loess data = all3;
model porp_trump = diff_time/ residual;
ods output OutputStatistics = results_all;
by state new_grade;
run;

data results_all;
set results_all;
week = round(diff_time,7);
run;

proc sort data = results_all;
by state new_grade week;
run;

*working on getting loess point estimates for all states by week and by grade
not sure if it is worth it as the proc mixed showed there isnt a signifigant
difference in 48 of the states;
proc sort data = all3;
by state;
run;


proc sgplot data = all3;
scatter x = diff_time y = abs_all_diff;
by state;
xaxis reverse;
reg y = abs_all_diff x = diff_time;
run;

data all3;
set all3;
if porp_trump > lower_two_week & porp_trump < upper_two_week then check11 =
1;
else check11 = 0;
run;

proc sort data = all3;
by state two_week;
run;

proc means data = all3 mean;
class state two_week;
var check11;
run;

proc sort data =all3;
by state;
run;

proc ttest data = all3;
class new_grade;
by state;
var porp_trump;
run;

proc sort data = all3;
```

```
by grade16;
run;

data all5;
set all3;
where diff_time < 16;
run;

data all5;
set all5;
tt = porp_trump;
run;

proc sort data = all5;
by state diff_time;
run;

proc means data = all5 clm;
class state;
var check11;
run;

data all5;
set all5;
diff_poll = abs(porp_trump_mean_g - porp_trump);
run;

proc sort all5;
by state name;
run;


data USA_7;
set USA;
by rounded_time_7 name;
if first.name;
run;

data all6;
set all5;
by state name diff_time;





/*  --> need to sort before using the 'by' statement
proc freq data = po2
by state;
where state in (PA, mich,...);



/********************************************

data po;
set po;
```

```
m = sqrt((porp_trump*(1- porp_trump))/n_state);
moe = m*1.96;
run;

proc means data = po mean nonobs;
class state;
var moe;
run;

data se;
set po;
where state = "U.S.";
run;

proc means data = se STD;
var porp_trump;
run;

data se;
set se;
standard = 0.0286958/SQRT(n_state);
run;

data se;
set se;
upper = p_state_avg + standard;
lower = p_state_avg - standard;
run;


data se;
set se;
if(porp_trump < upper & porp_trump > lower) then check = 1;
else check = 0;
run;

proc freq data = se;
tables check;
run;

*/
```