

```

1 *****
2 ***** Merging & Variable Creation *****
3 ***** Last Updated: 31 August 2018 *****
4 *****
5
6 * Preamble *
7 clear all
8 cd "YOUR_DIR"
9
10 *****
11 *** Bring in main dataset and merge with additional datasets ***
12 *****
13
14 sysuse Step2_final, clear
15
16 * Merge in future returns data *
17 * First size, BTM, and MOM *
18 merge 1:1 gvkey fyearqtr using sizebtmmom, nogen
19 * Now size and BTM *
20 merge 1:1 gvkey fyearqtr using sizebtm, nogen
21
22 * Finally, merge in intraday realized volatility data *
23 merge 1:1 gvkey fyearqtr using rvtouse_main, nogen
24
25
26 * Save down after merge but before creating most other variables *
27 save Prereg_data_final, replace
28
29
30
31 *****
32 *** Begin variable creation and clean up data ***
33 *****
34
35 sysuse Prereg_data_final, clear
36
37 * Fix case issues from data import *
38 foreach v of varlist all {
39     capture rename `v' `= lower("`v'")'
40 }
41
42 * Firm indicators in case gvkey causes issues *
43 egen firmid = group(gvkey)
44
45 *** Renaming variables ***
46 * Volume variables *
47 rename (abn_turn_event_0_1 abn_turn_event abn_turn_event_0_3 abn_turn_event_0_4
48 abn_turn_event_0_5) ///
49 (abn_turn_event_01 abn_turn_event_02 abn_turn_event_03 abn_turn_event_04
50 abn_turn_event_05)
51 rename (abn_turn_retail_0_1 abn_turn_retail abn_turn_retail_0_3 abn_turn_retail_0_4
52 abn_turn_retail_0_5) ///
53 (abn_turn_retail_01 abn_turn_retail_02 abn_turn_retail_03 abn_turn_retail_04
54 abn_turn_retail_05)
55 * Intraday article data *
56 foreach x in before10 before5 before2 after10 after5 after2 diff10 diff5 diff2 {
57     rename `x'a `x'
58 }
59 * Article returns variables *
60 rename (retxttm retxytd) (rett rety)
61 * UE variables *
62 rename (srw_ue zacks_ue ibes_ue analyst_ue_rank) (ue_srw ue_zacks ue_ibes ue_analyst_rank)
63 * A few AI variables *
64 rename (ai_cov ap_cov) (AI_cov AP_cov)
65
66
67 *** Clean up and replace a few variables ***
68 * Rename institutional holding variable *
69 rename inst_hold_wv inst_hold
70

```

```

67 * Replace missing values with zero for AI analyst data *
68 replace cons_earn_amt = 0 if missing(cons_earn_amt)
69
70 *** Create abnormal intraday volatility measures ***
71 gen abn_volat_intra_02 = ave_volat_intra_ea - ave_volat_intra_est
72 * Log transform of the normal abnormal volatility measure *
73 gen lnabn_volat_intra_02 = ln(ave_volat_intra_ea/ave_volat_intra_est)
74 replace lnabn_volat_intra_02 = 0 if missing(lnabn_volat_intra_02)
75
76 *** Take absolute value of pre-EA returns ***
77 foreach var in ret_pre_1week {
78     gen abs`var' = abs(`var')
79 }
80
81 *** Winsorize and demean non-binary control variables ***
82 foreach var in reporting_lag busyday lnmv btm ln_analysts inst_hold volat_pre crsp_price ///
83     log_dowflash_count ret_pre_1week absret_pre_1week AP_cov {
84     winsor2 `var', cuts(1 99) replace
85     egen `var' mean = mean(`var')
86     gen zz`var' = `var' - `var'_mean
87     drop `var'_mean
88 }
89
90 * Decile rank pre-EA momentum *
91 foreach x in absret_pre_1week {
92     egen `x'10 = xtile(`x'), nq(10)
93     replace `x'10 = `x'10 - 1
94 }
95 * Rename decile ranked variables *
96 rename (absret_pre_1week10) (absret_prelw10)
97
98 * Demean pre-EA return deciles *
99 foreach var in absret_prelw10 {
100     egen `var' mean = mean(`var')
101     gen zz`var' = `var' - `var'_mean
102     drop `var'_mean
103 }
104
105 *** Adjust window for pre-EA guidance well before the earnings announcement ***
106 replace pre_guide_earn = 0 if preempt_time < -55
107 replace pre_guide_earnq = 0 if preempt_time < -55
108 replace warn_earn = 0 if preempt_time_neg < -55
109
110 * Check if any treatment firms provided quarterly guide after treatment but not before *
111 bysort gvkey: egen guide_any_warn = max(warn_earn)
112 bysort gvkey: egen guide_any = max(pre_guide_earnq)
113
114 * Demean management forecast variables *
115 foreach x in guide_ea pre_guide_earnq warn_earn guide_any guide_any_warn {
116     egen `x' mean = mean(`x')
117     gen zz`x' = `x' - `x'_mean
118     drop `x'_mean
119 }
120
121
122 *** Multiply certain variables to be expressed in terms of basis points ***
123 foreach x in abn_turn_retail_02 abn_turn_retail_01 abn_turn_retail_03 ///
124     abn_turn_retail_04 abn_turn_retail_05 abn_volat_intra_02 {
125     replace `x' = `x' * 100
126 }
127
128 *** Winsorize abnvol, turnover, and volatility measures ***
129 foreach x in abn_turn_event_02 abn_turn_retail_02 ///
130     abn_turn_event_01 abn_turn_event_03 abn_turn_event_04 abn_turn_event_05 ///
131     abn_turn_retail_01 abn_turn_retail_03 abn_turn_retail_04 abn_turn_retail_05 ///
132     abn_buy_event abn_sell_event abn_volat_intra_02 lnabn_volat_intra_02 {
133     winsor2 `x', cuts(1 99) replace
134 }
135
136 *****

```

```

137 *** Create UE and Ret variables for main analyses ***
138 *****
139
140 * Drop UE variable since we will recreate here shortly *
141 drop ue
142
143 * Create average returns signal variable (average of signed YTD and TTM returns) *
144 egen reta = rowmean(rett rety)
145
146 * Max returns variable (e.g., higher of YTD or TTM) *
147 gen retm = rett
148 replace retm = rety if abs(rety) > abs(rett)
149
150 * Recreate UE variables *
151 egen ue_analyst = rowmean(ue_zacks ue_ibes)
152 * Winsorize analyst UE variables first, then SRW UE variables if missing analyst *
153 winsor2 ue_analyst, cuts(1 99) replace
154 winsor2 ue_srw if missing(ue_analyst), cuts(1 99) replace
155
156 * Absolute value of a few UE variables *
157 gen abs_ue_analyst = abs(ue_analyst)
158 gen abs_ue_srw = abs(ue_srw)
159 gen abs_ue_zacks = abs(ue_zacks)
160 gen abs_ue_ibes = abs(ue_ibes)
161
162 * Recreate main UE variable *
163 gen ue = ue_analyst
164     replace ue = ue_srw if ue==.
165 gen abs_ue = abs(ue)
166
167 * Absolute returns variables *
168 foreach x in rett rety reta {
169     gen abs`x' = abs(`x')
170 }
171 egen absretm = rowmax(absreth absrety)
172
173
174 *** Decile rank variables - do returns first ***
175 *** Need to do UE differently due to ordering of analyst vs. SRW ***
176 foreach x in reth absreth rety absrety reta absreta retm absretm {
177     egen `x'10 = xtile(`x'), nq(10)
178     replace `x'10 = `x'10 - 1
179 }
180
181 *****
182 *** Decile rank UE variables ***
183 *****
184
185 *** Main absolute UE measure ***
186 * First, pool Zacks and IBES surprise, then use SRW if both are missing *
187 egen absuetemp1 = xtile(abs_ue_analyst) if !missing(abs_ue_analyst), nq(10)
188 replace absuetemp1 = absuetemp1 - 1
189 egen absuetemp2 = xtile(abs_ue_srw) if missing(absuetemp1), nq(10)
190 replace absuetemp2 = absuetemp2 - 1
191 * Join these together - absue10 is the main unsigned decile rank variable *
192 egen absue10 = rowtotal(absuetemp1 absuetemp2)
193 drop absuetemp1 absuetemp2
194
195 *** Alternate UE #1 - Zacks first, then IBES, then SRW ***
196 * Deciles for different analyst providers *
197 egen azacksrank = xtile(abs_ue_zacks) if !missing(abs_ue_zacks), nq(10)
198 egen aibesrank = xtile(abs_ue_ibes) if !missing(abs_ue_ibes), nq(10)
199 egen aanrank = xtile(abs_ue_analyst) if !missing(abs_ue_analyst), nq(10)
200
201 gen absuetemp1 = azacksrank if !missing(abs_ue_zacks)
202 replace absuetemp1 = absuetemp1 - 1
203 gen absuetemp2 = aibesrank if !missing(abs_ue_ibes) & missing(abs_ue_zacks)
204 replace absuetemp2 = absuetemp2 - 1
205 egen absuetemp3 = xtile(abs_ue_srw) if missing(abs_ue_analyst), nq(10)
206 replace absuetemp3 = absuetemp3 - 1

```

```

207 * Join these together - absuealt10 is an alternative definition *
208 egen absuealt10 = rowtotal(absuetemp1 absuetemp2 absuetemp3)
209 drop absuetemp1 absuetemp2 absuetemp3
210
211 *** Alternative UE #2 - First IBES, then Zacks, then SRW ***
212 gen absuetemp1 = aibesrank if !missing(abs_ue_ibes)
213 replace absuetemp1 = absuetemp1 - 1
214 gen absuetemp2 = azacksrank if !missing(abs_ue_zacks) & missing(abs_ue_ibes)
215 replace absuetemp2 = absuetemp2 - 1
216 egen absuetemp3 = xtile(abs_ue_srw) if missing(abs_ue_analyst), nq(10)
217 replace absuetemp3 = absuetemp3 - 1
218 * Join these together *
219 egen absuealt210 = rowtotal(absuetemp1 absuetemp2 absuetemp3)
220 drop absuetemp1 absuetemp2 absuetemp3
221
222 *** Alternative UE #3 - only Zacks, then SRW (no IBES) ***
223 gen absuetemp1 = azacksrank if !missing(abs_ue_zacks)
224 replace absuetemp1 = absuetemp1 - 1
225 egen absuetemp2 = xtile(abs_ue_srw) if missing(absuetemp1), nq(10)
226 replace absuetemp2 = absuetemp2 - 1
227 * Join these together *
228 egen absuealt310 = rowtotal(absuetemp1 absuetemp2)
229 drop absuetemp1 absuetemp2
230
231 *** Alternative UE #4 - only IBES, then SRW (no Zacks) ***
232 gen absuetemp1 = aibesrank if !missing(abs_ue_ibes)
233 replace absuetemp1 = absuetemp1 - 1
234 egen absuetemp2 = xtile(abs_ue_srw) if missing(absuetemp1), nq(10)
235 replace absuetemp2 = absuetemp2 - 1
236 * Join these together *
237 egen absuealt410 = rowtotal(absuetemp1 absuetemp2)
238 drop absuetemp1 absuetemp2
239
240 *** Alternate UE - only SRW ***
241 egen absuesrw10 = xtile(abs_ue_srw), nq(10)
242 replace absuesrw10 = absuesrw10 - 1
243
244 * Drop a few unneeded variables *
245 drop azacksrank aibesrank aanrank lag_rdq_to_use
246
247
248 *****
249 *** Each of the above were for absolute (unsigned) UE ***
250 *** Do this again, except with signed UE ***
251 *****
252
253 *** Main signed UE measure ***
254 * Start with base definition of UE *
255 egen uetemp1 = xtile(ue_analyst) if !missing(ue_analyst), nq(10)
256 replace uetemp1 = uetemp1 - 1
257 egen uetemp2 = xtile(ue_srw) if missing(ue_analyst), nq(10)
258 replace uetemp2 = uetemp2 - 1
259 * Join these together = ue10 is the main signed decile rank variable *
260 egen ue10 = rowtotal(uetemp1 uetemp2)
261 drop uetemp1 uetemp2
262
263 *** Alternate UE #1 - Zacks first, then IBES, then SRW ***
264 * Deciles for different analyst providers *
265 egen zacksrank = xtile(ue_zacks) if !missing(ue_zacks), nq(10)
266 egen ibesrank = xtile(ue_ibes) if !missing(ue_ibes), nq(10)
267 egen anrank = xtile(ue_analyst) if !missing(ue_analyst), nq(10)
268
269 gen uetemp1 = zacksrank if !missing(ue_zacks)
270 replace uetemp1 = uetemp1 - 1
271 gen uetemp2 = ibesrank if !missing(ue_ibes) & missing(ue_zacks)
272 replace uetemp2 = uetemp2 - 1
273 egen uetemp3 = xtile(ue_srw) if missing(ue_analyst), nq(10)
274 replace uetemp3 = uetemp3 - 1
275 * Join these together - uealt10 is an alternative definition *
276 egen uealt10 = rowtotal(uetemp1 uetemp2 uetemp3)

```

```

277 drop uetemp1 uetemp2 uetemp3
278
279 *** Alternative UE #2 - First IBES, then Zacks, then SRW ***
280 gen uetemp1 = ibesrank if !missing(ue_ibes)
281 replace uetemp1 = uetemp1 - 1
282 gen uetemp2 = zacksrank if !missing(ue_zacks) & missing(ue_ibes)
283 replace uetemp2 = uetemp2 - 1
284 egen uetemp3 = xtile(ue_srw) if missing(ue_analyst), nq(10)
285 replace uetemp3 = uetemp3 - 1
286 * Join these together *
287 egen uealt210 = rowtotal(uetemp1 uetemp2 uetemp3)
288 drop uetemp1 uetemp2 uetemp3
289
290 *** Alternative UE #3 - only Zacks, then SRW (no IBES) ***
291 gen uetemp1 = zacksrank if !missing(ue_zacks)
292 replace uetemp1 = uetemp1 - 1
293 egen uetemp2 = xtile(ue_srw) if missing(uetemp1), nq(10)
294 replace uetemp2 = uetemp2 - 1
295 * Join these together *
296 egen uealt310 = rowtotal(uetemp1 uetemp2)
297 drop uetemp1 uetemp2
298
299 *** Alternative UE #4 - only IBES, then SRW (no Zacks) ***
300 gen uetemp1 = ibesrank if !missing(ue_ibes)
301 replace uetemp1 = uetemp1 - 1
302 egen uetemp2 = xtile(ue_srw) if missing(uetemp1), nq(10)
303 replace uetemp2 = uetemp2 - 1
304 * Join these together *
305 egen uealt410 = rowtotal(uetemp1 uetemp2)
306 drop uetemp1 uetemp2
307
308 *** Alternate UE - only SRW ***
309 egen uesrw10 = xtile(ue_srw), nq(10)
310 replace uesrw10 = uesrw10 - 1
311
312
313 *** Demean UE and Ret deciles ***
314 foreach x in rettl10 absrettl10 rety10 absrety10 reta10 absreta10 retm10 absretm10 ///
315     absue10 absuealt10 absuealt210 absuealt310 absuealt410 absuesrw10 ///
316     ue10 uealt10 uealt210 uealt310 uealt410 uesrw10 {
317     egen `x' mean = mean(`x')
318     gen zz`x' = `x' - `x'_mean
319     drop `x' mean
320 }
321
322
323 *****
324 *** Generate main AI interaction variables ***
325 *****
326
327 *** Interact AI coverage with UE and Ret ***
328 * First UE variables *
329 foreach x in ue absue absuealt absuealt2 absuealt3 absuealt4 absuesrw {
330     gen AI_`x'10 = AI_cov * zz`x'10
331 }
332 * Returns variables *
333 foreach x in rettl10 absrettl10 rety10 absrety10 reta10 absreta10 retm10 absretm10 {
334     gen AI_`x' = AI_cov * zz`x'
335 }
336
337 * Identify top and bottom extreme decile before absolute values *
338 * Returns variables *
339 foreach x in rettl rety reta retm {
340     gen abs`x'2ext = 0
341     replace abs`x'2ext = 1 if `x'10 == 0 | `x'10 == 9
342 }
343 * UE variables *
344 foreach x in ue uealt uealt2 uealt3 uealt4 uesrw {
345     gen abs`x'2ext = 0
346     replace abs`x'2ext = 1 if `x'10 == 0 | `x'10 == 9

```

```

347 }
348
349 * Demean top 2 extreme deciles *
350 foreach x in absrett2ext absrety2ext absreta2ext absretm2ext ///
351     absue2ext absuealt2ext absuealt22ext absuealt32ext absuealt42ext absuesrw2ext {
352     egen `x'__mean = mean(`x')
353     gen zz`x' = `x' - `x'__mean
354     drop `x'__mean
355 }
356
357 * Creating AI vars with extreme indicators *
358 foreach x in absrett2ext absrety2ext absreta2ext absretm2ext ///
359     absue2ext absuealt2ext absuealt22ext absuealt32ext absuealt42ext absuesrw2ext {
360     gen AI_`x' = AI_cov * zz`x'
361 }
362
363 * Extreme / Non-Extreme splits *
364 * Creating variables for use in FE/control interaction, and then AI interaction terms *
365 foreach var in ue rett rety reta retm uealt uealt2 uealt3 uealt4 uesrw {
366     gen zzabs`var'10ext = zzabs`var'10 * abs`var'2ext
367     gen zzabs`var'10nonext = zzabs`var'10 if abs`var'2ext==0
368     replace zzabs`var'10nonext = 0 if abs`var'2ext==1
369
370     gen AI_abs`var'10ext = AI_cov * zzabs`var'10ext
371     gen AI_abs`var'10nonext = AI_cov * zzabs`var'10nonext
372 }
373
374
375 *****
376 *** Cross-sectional variables (e.g., analyst provided, positive/negative) ***
377 *****
378
379 *** Positive / negative splits for buy/sell tests ***
380 * Returns variables *
381 foreach x in rett rety reta retm {
382     gen `x'__pos = (`x' >= 0)
383     gen `x'__neg = (`x' < 0)
384 }
385 * UE - only do main variable *
386 foreach x in ue {
387     gen `x'__pos = (`x' >= 0)
388     gen `x'__neg = (`x' < 0)
389 }
390
391 * Interactions *
392 foreach var in ue rett rety reta retm {
393     foreach sign in pos neg {
394         gen zzabs`var'10_`sign' = zzabs`var'10 * `var'__`sign'
395         gen zzabs`var'10ext_`sign' = zzabs`var'10ext * `var'__`sign'
396         gen zzabs`var'10nonext_`sign' = zzabs`var'10nonext * `var'__`sign'
397         gen AI_abs`var'10_`sign' = AI_cov * zzabs`var'10_`sign'
398         gen AI_abs`var'10ext_`sign' = AI_cov * zzabs`var'10ext_`sign'
399         gen AI_abs`var'10nonext_`sign' = AI_cov * zzabs`var'10nonext_`sign'
400     }
401 }
402
403 *** Articles with analyst provided estimates - for Acquisition costs tests ***
404 gen anpro = 1 if (AI_cov==1 & cons_earn_amt==1) | (AI_cov==0 & number_of_est>=3 &
number_of_est!=.)
405 replace anpro = 0 if (AI_cov==1 & cons_earn_amt==0) | (AI_cov==0 & (number_of_est<3 |
number_of_est==.))
406 gen nonanpro = (anpro == 0)
407
408 *** Loss and Non-loss firm quarters ***
409 gen lsq = (loss == 1)
410 gen nonlsq = (loss == 0)
411
412
413 *****
414 *** Generate remaining variables ***

```

```

415 *****
416
417 foreach var in ue rett rety reta retm {
418     foreach xvar in anpro nonanpro lsq nonlsq {
419         gen zzabs`var'10_`xvar' = zzabs`var'10 * `xvar'
420         gen zzabs`var'10ext_`xvar' = zzabs`var'10ext * `xvar'
421         gen zzabs`var'10nonext_`xvar' = zzabs`var'10nonext * `xvar'
422         gen AI_abs`var'10_`xvar' = AI_cov * zzabs`var'10_`xvar'
423         gen AI_abs`var'10ext_`xvar' = AI_cov * zzabs`var'10ext_`xvar'
424         gen AI_abs`var'10nonext_`xvar' = AI_cov * zzabs`var'10nonext_`xvar'
425
426         foreach sign in pos neg {
427             gen zzabs`var'10_`sign'_`xvar' = zzabs`var'10 * `var'`sign' * `xvar'
428             gen zzabs`var'10ext_`sign'_`xvar' = zzabs`var'10ext * `var'`sign' * `xvar'
429             gen zzabs`var'10nonext_`sign'_`xvar' = zzabs`var'10nonext * `var'`sign' * `xvar'
430             gen AI_abs`var'10_`sign'_`xvar' = AI_cov * zzabs`var'10_`sign'_`xvar'
431             gen AI_abs`var'10ext_`sign'_`xvar' = AI_cov * zzabs`var'10ext_`sign'_`xvar'
432             gen AI_abs`var'10nonext_`sign'_`xvar' = AI_cov * zzabs`var'10nonext_`sign'_`xvar'
433         }
434     }
435 }
436 * Create variables with alternate UE definitions *
437 foreach var in uealt uealt2 uealt3 uealt4 uesrw {
438     foreach xvar in anpro nonanpro lsq nonlsq {
439         gen zzabs`var'10_`xvar' = zzabs`var'10 * `xvar'
440         gen zzabs`var'10ext_`xvar' = zzabs`var'10ext * `xvar'
441         gen zzabs`var'10nonext_`xvar' = zzabs`var'10nonext * `xvar'
442         gen AI_abs`var'10_`xvar' = AI_cov * zzabs`var'10_`xvar'
443         gen AI_abs`var'10ext_`xvar' = AI_cov * zzabs`var'10ext_`xvar'
444         gen AI_abs`var'10nonext_`xvar' = AI_cov * zzabs`var'10nonext_`xvar'
445     }
446 }
447
448 * Create/rename variables for the future return tests *
449 * These are the indicators for positive/negative UE and TTM returns *
450 rename (ue_pos reta_pos ue_neg reta_neg) (posUE posRET negUE negRET)
451
452 * Create year-week variable for clustering *
453 gen week = week(rdq_to_use)
454 * Patch together *
455 gen year2 = year(rdq_to_use)
456 egen yearweek = concat(year2 week)
457 drop week year2
458
459 * Split into extreme and non-extremes based on signed values for UE *
460 gen posUEext = 0
461 replace posUEext = 1 if ue10 == 9
462 gen posUEnonext = 0
463 replace posUEnonext = 1 if posUEext == 0 & ue >= 0
464 gen negUEext = 0
465 replace negUEext = 1 if ue10 == 0
466 gen negUEnonext = 0
467 replace negUEnonext = 1 if negUEext == 0 & ue < 0
468
469 * Same deal with Returns *
470 gen posREText = 0
471 replace posREText = 1 if reta10 == 9
472 gen posRETnonext = 0
473 replace posRETnonext = 1 if posREText == 0 & reta >= 0
474 gen negREText = 0
475 replace negREText = 1 if reta10 == 0
476 gen negRETnonext = 0
477 replace negRETnonext = 1 if negREText == 0 & reta < 0
478
479
480 * Save file down *
481 save Regression_data_Final, replace
482
483
484 *****

```

485	***	END OF FILE - MOVE OVER TO MAIN ANALYSIS FILE	***
486	*****		
487			
488			
489			
490			