

Final Project Deliverable 2

Number of Employees

```
1  reg d.lnCount l(12,24,36,48)d.lnCount // .01637
2  scalar drop _all
3  quietly forval w=12(12)144 {
4  gen pred=.
5  gen nob=.
6      forval t=421/733 {
7          gen wstart=`t'-'w'
8          gen wend=`t'-1
9          reg dlnCount l12dlnCount l24dlnCount l36dlnCount l48dlnCount ///
10             if Date>=wstart & Date<=wend
11             replace nob=e(N) if Date==`t'
12             predict ptemp
13             replace pred=ptemp if Date==`t'
14             drop ptemp wstart wend
15         }
16     gen errsq=(pred-d.lnCount)^2
17     summ errsq
18     scalar RWrmse`w'=r(mean)^.5
19     summ nob
20     scalar RWminobs`w'=r(min)
21     scalar RWmaxobs`w'=r(max)
22     drop errsq pred nob
23 }
24 scalar list
25 /*
26 RWmaxobs132 =          132
27 RWminobs132 =           12
28 RWrmse132 =    .0172128
29 */
```

```
1  reg d.lnCount l(12,36)d.lnCount m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
2  scalar drop _all
3  quietly forval w=12(12)144 {
4  gen pred=.
5  gen nob=.
6      forval t=409/733 {
7          gen wstart=`t'-'w'
8          gen wend=`t'-1
9          reg dlnCount l12dlnCount l36dlnCount m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 ///
10             if Date>=wstart & Date<=wend
11             replace nob=e(N) if Date==`t'
12             predict ptemp
```

```

13   replace pred=ptemp if Date==`t'
14   drop ptemp wstart wend
15   }
16   gen errsq=(pred-d.lnCount)^2
17   summ errsq
18   scalar RWrmse`w'=r(mean)^.5
19   summ nobs
20   scalar RWminobs`w'=r(min)
21   scalar RWmaxobs`w'=r(max)
22   drop errsq pred nobs
23   }
24   scalar list
25   /*
26   RWmaxobs144 =          144
27   RWminobs144 =           12
28   RWrmse144 =   .01777071
29   */

```

I'm debating between the two. I'm also not sure if I should try and find another one that has more explanatory variables.

Weekly Earnings

```

1   reg d.lnWeeklyEarnings l3d.lnWeeklyEarnings l5d.lnWeeklyEarnings l7d.lnWeeklyEarnings
2   scalar drop _all
3   quietly forval w=12(12)84 {
4   gen pred=.
5   gen nobs=.
6   forval t=622/733 {
7   gen wstart=`t'-'w'
8   gen wend=`t'-1
9   reg dlnWeeklyEarnings l3dlnWeeklyEarnings l5dlnWeeklyEarnings l7dlnWeeklyEarnings
10  ///
11  if Date>=wstart & Date<=wend
12  replace nobs=e(N) if Date==`t'
13  predict ptemp
14  replace pred=ptemp if Date==`t'
15  drop ptemp wstart wend
16  }
17  gen errsq=(pred-d.lnWeeklyEarnings)^2
18  summ errsq
19  scalar RWrmse`w'=r(mean)^.5
20  summ nobs
21  scalar RWminobs`w'=r(min)
22  scalar RWmaxobs`w'=r(max)
23  drop errsq pred nobs
24  }
25  scalar list
26  /*
27  RWmaxobs84 =          84

```

```
27 | RWminobs84 =          2
28 | RWrmse84 =  .05250414
29 | */
```

Code

```
1  clear
2  set more off
3
4  cd "/Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time Series/Problem Sets/Final
   Project"
5  *log using "Final Project.smcl", replace
6  import delimited "TS2020_Final_Project_txt2/TS2020_Final_Project_Monthly.txt"
7  rename smu12455400500000001 Count
8  rename smu12455400500000002 WeekHours
9  rename smu12455400500000003 HourlyEarnings
10 rename smu12455400500000011 WeeklyEarnings
11 rename smu12455400800000001 ServiceCount
12
13
14 label variable Count "Count"
15 label variable WeekHours "WeekHours"
16 label variable HourlyEarnings "HourlyEarnings"
17 label variable WeeklyEarnings "WeeklyEarnings"
18 label variable ServiceCount "ServiceCount"
19
20
21 gen datec=date(date, "YMD")
22 gen Date=mofd(datec)
23 gen month=month(datec)
24 format Date %tm
25 tsset Date
26
27 gen lnCount = ln(Count)
28 gen lnWeekHours = ln(WeekHours)
29 gen lnHourlyEarnings = ln(HourlyEarnings)
30 gen lnWeeklyEarnings = ln(WeeklyEarnings)
31 gen lnServiceCount = ln(ServiceCount)
32
33 gen m1=0
34 replace m1=1 if month==1
35 gen m2=0
36 replace m2=1 if month==2
37 gen m3=0
38 replace m3=1 if month==3
39 gen m4=0
40 replace m4=1 if month==4
41 gen m5=0
```

```
42  replace m5=1 if month==5
43  gen m6=0
44  replace m6=1 if month==6
45  gen m7=0
46  replace m7=1 if month==7
47  gen m8=0
48  replace m8=1 if month==8
49  gen m9=0
50  replace m9=1 if month==9
51  gen m10=0
52  replace m10=1 if month==10
53  gen m11=0
54  replace m11=1 if month==11
55  gen m12=0
56  replace m12=1 if month==12
57
58  gen dlnCount=d.lnCount
59  gen l1dlnCount=l1d.lnCount
60  gen l2dlnCount=l2d.lnCount
61  gen l3dlnCount=l3d.lnCount
62  gen l4dlnCount=l4d.lnCount
63  gen l5dlnCount=l5d.lnCount
64  gen l6dlnCount=l6d.lnCount
65  gen l7dlnCount=l7d.lnCount
66  gen l8dlnCount=l8d.lnCount
67  gen l9dlnCount=l9d.lnCount
68  gen l10dlnCount=l10d.lnCount
69  gen l11dlnCount=l11d.lnCount
70  gen l12dlnCount=l12d.lnCount
71  gen l24dlnCount=l24d.lnCount
72  gen l36dlnCount=l36d.lnCount
73  gen l48dlnCount=l48d.lnCount
74
75  gen dlnWeekHours=d.lnWeekHours
76  gen l1dlnWeekHours=l1d.lnWeekHours
77  gen l2dlnWeekHours=l2d.lnWeekHours
78  gen l3dlnWeekHours=l3d.lnWeekHours
79  gen l4dlnWeekHours=l4d.lnWeekHours
80  gen l5dlnWeekHours=l5d.lnWeekHours
81  gen l6dlnWeekHours=l6d.lnWeekHours
82  gen l7dlnWeekHours=l7d.lnWeekHours
83  gen l8dlnWeekHours=l8d.lnWeekHours
84  gen l9dlnWeekHours=l9d.lnWeekHours
85  gen l10dlnWeekHours=l10d.lnWeekHours
86  gen l11dlnWeekHours=l11d.lnWeekHours
87  gen l12dlnWeekHours=l12d.lnWeekHours
88  gen l24dlnWeekHours=l24d.lnWeekHours
89  gen l36dlnWeekHours=l36d.lnWeekHours
90  gen l48dlnWeekHours=l48d.lnWeekHours
91
92  gen dlnHourlyEarnings=d.lnHourlyEarnings
```

```
93 gen l1dlnHourlyEarnings=l1d.lnHourlyEarnings
94 gen l2dlnHourlyEarnings=l2d.lnHourlyEarnings
95 gen l3dlnHourlyEarnings=l3d.lnHourlyEarnings
96 gen l4dlnHourlyEarnings=l4d.lnHourlyEarnings
97 gen l5dlnHourlyEarnings=l5d.lnHourlyEarnings
98 gen l6dlnHourlyEarnings=l6d.lnHourlyEarnings
99 gen l7dlnHourlyEarnings=l7d.lnHourlyEarnings
100 gen l8dlnHourlyEarnings=l8d.lnHourlyEarnings
101 gen l9dlnHourlyEarnings=l9d.lnHourlyEarnings
102 gen l10dlnHourlyEarnings=l10d.lnHourlyEarnings
103 gen l11dlnHourlyEarnings=l11d.lnHourlyEarnings
104 gen l12dlnHourlyEarnings=l12d.lnHourlyEarnings
105 gen l24dlnHourlyEarnings=l24d.lnHourlyEarnings
106 gen l36dlnHourlyEarnings=l36d.lnHourlyEarnings
107 gen l48dlnHourlyEarnings=l48d.lnHourlyEarnings
108
109 gen dlnWeeklyEarnings=d.lnWeeklyEarnings
110 gen l1dlnWeeklyEarnings=l1d.lnWeeklyEarnings
111 gen l2dlnWeeklyEarnings=l2d.lnWeeklyEarnings
112 gen l3dlnWeeklyEarnings=l3d.lnWeeklyEarnings
113 gen l4dlnWeeklyEarnings=l4d.lnWeeklyEarnings
114 gen l5dlnWeeklyEarnings=l5d.lnWeeklyEarnings
115 gen l6dlnWeeklyEarnings=l6d.lnWeeklyEarnings
116 gen l7dlnWeeklyEarnings=l7d.lnWeeklyEarnings
117 gen l8dlnWeeklyEarnings=l8d.lnWeeklyEarnings
118 gen l9dlnWeeklyEarnings=l9d.lnWeeklyEarnings
119 gen l10dlnWeeklyEarnings=l10d.lnWeeklyEarnings
120 gen l11dlnWeeklyEarnings=l11d.lnWeeklyEarnings
121 gen l12dlnWeeklyEarnings=l12d.lnWeeklyEarnings
122 gen l24dlnWeeklyEarnings=l24d.lnWeeklyEarnings
123 gen l36dlnWeeklyEarnings=l36d.lnWeeklyEarnings
124 gen l48dlnWeeklyEarnings=l48d.lnWeeklyEarnings
125
126 gen dlnServiceCount=d.lnServiceCount
127 gen l1dlnServiceCount=l1d.lnServiceCount
128 gen l2dlnServiceCount=l2d.lnServiceCount
129 gen l3dlnServiceCount=l3d.lnServiceCount
130 gen l4dlnServiceCount=l4d.lnServiceCount
131 gen l5dlnServiceCount=l5d.lnServiceCount
132 gen l6dlnServiceCount=l6d.lnServiceCount
133 gen l7dlnServiceCount=l7d.lnServiceCount
134 gen l8dlnServiceCount=l8d.lnServiceCount
135 gen l9dlnServiceCount=l9d.lnServiceCount
136 gen l10dlnServiceCount=l10d.lnServiceCount
137 gen l11dlnServiceCount=l11d.lnServiceCount
138 gen l12dlnServiceCount=l12d.lnServiceCount
139 gen l24dlnServiceCount=l24d.lnServiceCount
140 gen l36dlnServiceCount=l36d.lnServiceCount
141 gen l48dlnServiceCount=l48d.lnServiceCount
142
143 /*
```

```

144 The project is to forecast the March non-seasonally adjusted estimates of average
weekly earnings and total employment for private employers (total private) for a
Florida MSA of your choice and write up a professional report on your forecast.
145 */
146 /* Count and WeeklyEarnings */
147
148 summ Count WeekHours HourlyEarnings WeeklyEarnings ServiceCount
149 summ lnCount lnWeekHours lnHourlyEarnings lnWeeklyEarnings lnServiceCount
150
151 ac lnCount, saving(lnCount_ac, replace)
152 pac lnCount, saving(lnCount_pac, replace)
153 graph combine lnCount_ac.gph lnCount_pac.gph, saving(lnCount_ac_pac, replace)
154 graph export "lnCount_ac_pac.png", replace
155 ** Probably need to difference
156
157 ac lnWeeklyEarnings, saving(lnWeeklyEarnings_ac, replace)
158 pac lnWeeklyEarnings, saving(lnWeeklyEarnings_pac, replace)
159 graph combine lnWeeklyEarnings_ac.gph lnWeeklyEarnings_pac.gph,
saving(lnWeeklyEarnings_ac_pac, replace)
160 graph export "lnWeeklyEarnings_ac_pac.png", replace
161 ** Probably need to differencen b
162
163 *starter models for count
164 *I used a pair plot to examine the rise and fall of variables with respect to each
other
165 reg d.lnCount l(12,24,36,48)d.lnCount // .01637
166 scalar drop_all
167 quietly forval w=12(12)144 {
168 gen pred=.
169 gen nobs=.
170     forval t=421/733 {
171         gen wstart=`t'-'w'
172         gen wend=`t'-1
173         reg dlnCount l12dlnCount l24dlnCount l36dlnCount l48dlnCount ///
174             if Date>=wstart & Date<=wend
175         replace nobs=e(N) if Date==`t'
176         predict ptemp
177         replace pred=ptemp if Date==`t'
178         drop ptemp wstart wend
179     }
180 gen errsq=(pred-d.lnCount)^2
181 summ errsq
182 scalar RWrmse`w'=r(mean)^.5
183 summ nobs
184 scalar RWminobs`w'=r(min)
185 scalar RWmaxobs`w'=r(max)
186 drop errsq pred nobs
187 }
188 scalar list
189 /*
190 RWmaxobs132 =          132

```

```

191 RWminobs132 =      12
192 RWrmse132 =    .0172128
193 */
194
195 reg d.lnCount l(5,12,24,36,48)d.lnCount l(5)d.lnWeekHours m5 // .01711
196 scalar drop _all
197 quietly forval w=12(12)84 {
198   gen pred=.
199   gen nob=.
200   forval t=641/733 {
201     gen wstart=`t'-'w'
202     gen wend=`t'-1
203     reg dlnCount l5dlnCount l12dlnCount l24dlnCount l36dlnCount l48dlnCount
204     15dlnWeekHours m5 ///
205     if Date>=wstart & Date<=wend
206     replace nob=e(N) if Date==`t'
207     predict ptemp
208     replace pred=ptemp if Date==`t'
209     drop ptemp wstart wend
210   }
211   gen errsq=(pred-d.lnCount)^2
212   summ errsq
213   scalar RWrmse`w'=r(mean)^.5
214   summ nob
215   scalar RWminobs`w'=r(min)
216   scalar RWmaxobs`w'=r(max)
217   drop errsq pred nob
218 }
219 scalar list
220 /*
221 RWmaxobs84 =      84
222 RWminobs84 =      23
223 RWrmse84 =    .01950911
224 */
225
226 /*
227 gsreg dlnCount l1dlnCount l2dlnCount l3dlnCount l4dlnCount l5dlnCount l6dlnCount ///
228 l7dlnCount l8dlnCount l9dlnCount l10dlnCount l11dlnCount l12dlnCount ///
229 l24dlnCount l36dlnCount l48dlnCount ///
230 if tin(1990m1,2021m1), ///
231 ncomb(1,12) aic outsample(24) fix(m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 m12) ///
232 samesample nindex( -1 aic -1 bic -1 rmse_out) results(gsreg_dlnCount) replace
233 */
234
235 *gsreg suggestions
236 reg d.lnCount l12d.lnCount m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
237 scalar drop _all
238 quietly forval w=12(12)144 {
239   gen pred=.
240   gen nob=.
241   forval t=385/733 {

```

```

241     gen wstart=`t'-'w'
242     gen wend=`t'-1
243     reg dlnCount l12dlnCount m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 ///
244     if Date>=wstart & Date<=wend
245     replace nobse=e(N) if Date==`t'
246     predict ptemp
247     replace pred=ptemp if Date==`t'
248     drop ptemp wstart wend
249     }
250 gen errsq=(pred-d.lnCount)^2
251 summ errsq
252 scalar RWrmse`w'=r(mean)^.5
253 summ nobse
254 scalar RWminobs`w'=r(min)
255 scalar RWmaxobs`w'=r(max)
256 drop errsq pred nobse
257 }
258 scalar list
259 /*
260 RWmaxobs144 =          144
261 RWminobs144 =          12
262 RWrmse144 =   .01824906
263 */
264
265 reg d.lnCount l(12,36)d.lnCount m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
266 scalar drop _all
267 quietly forval w=12(12)144 {
268     gen pred=.
269     gen nobse=.
270     forval t=409/733 {
271         gen wstart=`t'-'w'
272         gen wend=`t'-1
273         reg dlnCount l12dlnCount l36dlnCount m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 ///
274         if Date>=wstart & Date<=wend
275         replace nobse=e(N) if Date==`t'
276         predict ptemp
277         replace pred=ptemp if Date==`t'
278         drop ptemp wstart wend
279         }
280     gen errsq=(pred-d.lnCount)^2
281     summ errsq
282     scalar RWrmse`w'=r(mean)^.5
283     summ nobse
284     scalar RWminobs`w'=r(min)
285     scalar RWmaxobs`w'=r(max)
286     drop errsq pred nobse
287     }
288     scalar list
289     /*
290     RWmaxobs144 =          144
291     RWminobs144 =          12

```



```

292 RWrmsel44 = .01777071
293 */
294
295 /*
296 gsreg dlnCount l1dlnCount l2dlnCount l3dlnCount l4dlnCount l5dlnCount l6dlnCount ///
297     l7dlnCount l8dlnCount l9dlnCount l10dlnCount l11dlnCount l12dlnCount ///
298     l24dlnCount l36dlnCount ///
299     l1dlnWeekHours l2dlnWeekHours l3dlnWeekHours l4dlnWeekHours l5dlnWeekHours
l6dlnWeekHours ///
300     l7dlnWeekHours l8dlnWeekHours l9dlnWeekHours l10dlnWeekHours l11dlnWeekHours
l12dlnWeekHours ///
301     l24dlnWeekHours l36dlnWeekHours ///
302     l1dlnHourlyEarnings l2dlnHourlyEarnings l3dlnHourlyEarnings l4dlnHourlyEarnings
///
303     l5dlnHourlyEarnings l6dlnHourlyEarnings ///
304     l7dlnHourlyEarnings l8dlnHourlyEarnings l9dlnHourlyEarnings l10dlnHourlyEarnings
///
305     l11dlnHourlyEarnings l12dlnHourlyEarnings ///
306     l24dlnHourlyEarnings l36dlnHourlyEarnings ///
307     l1dlnWeeklyEarnings l2dlnWeeklyEarnings l3dlnWeeklyEarnings l4dlnWeeklyEarnings
///
308     l5dlnWeeklyEarnings l6dlnWeeklyEarnings ///
309     l7dlnWeeklyEarnings l8dlnWeeklyEarnings l9dlnWeeklyEarnings l10dlnWeeklyEarnings
///
310     l11dlnWeeklyEarnings l12dlnWeeklyEarnings ///
311     l24dlnWeeklyEarnings l36dlnWeeklyEarnings ///
312     if tin(2011m1,2021m1), ///
313     ncomb(1,4) aic outsample(24) fix(m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11) ///
314     samesample nindex( -1 aic -1 bic -1 rmse_out) results(gsreg_dlnCount_Full) replace
315 */
316
317 reg d.lnCount l4d.lnWeekHours l9d.lnWeekHours l8d.lnHourlyEarnings m1 m2 m3 m4 m5 m6
m7 m8 m9 m10 m11
318 scalar drop _all
319 quietly forval w=12(12)84 {
320     gen pred=.
321     gen nobs=.
322     forval t=624/733 {
323         gen wstart=`t'-'w'
324         gen wend=`t'-1
325         reg dlnCount l4dlnWeekHours l9dlnWeekHours l8dlnHourlyEarnings m1 m2 m3 m4 m5 m6
m7 m8 m9 m10 m11 ///
326         if Date>=wstart & Date<=wend
327         replace nobs=e(N) if Date==`t'
328         predict ptemp
329         replace pred=ptemp if Date==`t'
330         drop ptemp wstart wend
331     }
332     gen errsq=(pred-d.lnCount)^2
333     summ errsq
334     scalar RWrmsel`w'=r(mean)^.5

```

```

335 summ nobS
336 scalar RWminobs`w'=r(min)
337 scalar RWmaxobs`w'=r(max)
338 drop errsq pred nobS
339 }
340 scalar list
341 /*
342 RWmaxobs12 =          12
343 RWminobs12 =           2
344 RWrmse12 =   .0176238
345 */
346
347
348 *starter models for weekly earnings
349 reg d.lnWeeklyEarnings l1d.lnWeekHours ld.lnHourlyEarnings
350 scalar drop _all
351 quietly forval w=12(12)84 {
352   gen pred=.
353   gen nobS=.
354   forval t=616/733 {
355     gen wstart=`t'-'w'
356     gen wend=`t'-1
357     reg dlnWeeklyEarnings l1dlnWeekHours l1dlnHourlyEarnings ///
358       if Date>=wstart & Date<=wend
359     replace nobS=e(N) if Date==`t'
360     predict ptemp
361     replace pred=ptemp if Date==`t'
362     drop ptemp wstart wend
363   }
364   gen errsq=(pred-d.lnWeeklyEarnings)^2
365   summ errsq
366   scalar RWrmse`w'=r(mean)^.5
367   summ nobS
368   scalar RWminobs`w'=r(min)
369   scalar RWmaxobs`w'=r(max)
370   drop errsq pred nobS
371 }
372 scalar list
373 /*
374 RWmaxobs60 =          60
375 RWminobs60 =           2
376 RWrmse60 =   .06145693
377 */
378
379 /*
380 gsreg dlnWeeklyEarnings l1dlnWeeklyEarnings l2dlnWeeklyEarnings l3dlnWeeklyEarnings
///
381   l4dlnWeeklyEarnings l5dlnWeeklyEarnings l6dlnWeeklyEarnings ///
382   l7dlnWeeklyEarnings l8dlnWeeklyEarnings l9dlnWeeklyEarnings l10dlnWeeklyEarnings
///
383   l11dlnWeeklyEarnings l12dlnWeeklyEarnings ///

```

```

384     l24dlnWeeklyEarnings l36dlnWeeklyEarnings ///
385     if tin(2011m1,2021m1), ///
386     ncomb(1,12) aic outsample(24) ///
387     samesample nindex( -1 aic -1 bic -1 rmse_out) results(gsreg_dlnWeeklyEarnings)
replace
388 */
389
390 reg d.lnWeeklyEarnings l3d.lnWeeklyEarnings l5d.lnWeeklyEarnings m1 m2 m3 m4 m5 m6
m7 m8 m9 m10 m11
391 scalar drop _all
392 quietly forval w=12(12)84 {
393     gen pred=.
394     gen nobs=.
395     forval t=620/733 {
396         gen wstart=`t'-'w'
397         gen wend=`t'-1
398         reg dlnWeeklyEarnings l3dlnWeeklyEarnings l5dlnWeeklyEarnings m1 m2 m3 m4 m5 m6 m7
m8 m9 m10 m11 ///
399         if Date>=wstart & Date<=wend
400         replace nobs=e(N) if Date==`t'
401         predict ptemp
402         replace pred=ptemp if Date==`t'
403         drop ptemp wstart wend
404     }
405     gen errsq=(pred-d.lnWeeklyEarnings)^2
406     summ errsq
407     scalar RWrmse`w'=r(mean)^.5
408     summ nobs
409     scalar RWminobs`w'=r(min)
410     scalar RWmaxobs`w'=r(max)
411     drop errsq pred nobs
412 }
413 scalar list
414 /*
415 RWmaxobs84 =          84
416 RWminobs84 =           2
417 RWrmse84 =   .06004448
418 */
419
420 reg d.lnWeeklyEarnings l3d.lnWeeklyEarnings l5d.lnWeeklyEarnings
l7d.lnWeeklyEarnings
421 scalar drop _all
422 quietly forval w=12(12)84 {
423     gen pred=.
424     gen nobs=.
425     forval t=622/733 {
426         gen wstart=`t'-'w'
427         gen wend=`t'-1
428         reg dlnWeeklyEarnings l3dlnWeeklyEarnings l5dlnWeeklyEarnings l7dlnWeeklyEarnings
///
429         if Date>=wstart & Date<=wend

```

```

430     replace nobse=e(N) if Date==`t'
431     predict ptemp
432     replace pred=ptemp if Date==`t'
433     drop ptemp wstart wend
434 }
435 gen errsq=(pred-d.lnWeeklyEarnings)^2
436 summ errsq
437 scalar RWrmse`w'=r(mean)^.5
438 summ nobse
439 scalar RWminobs`w'=r(min)
440 scalar RWmaxobs`w'=r(max)
441 drop errsq pred nobse
442 }
443 scalar list
444 /*
445 RWmaxObs84 =          84
446 RWminObs84 =           2
447 RWrmse84 =  .05250414
448 */
449
450 /*
451 gsreg dlnCount l1dlnCount l2dlnCount l3dlnCount l4dlnCount l5dlnCount l6dlnCount ///
452     l7dlnCount l8dlnCount l9dlnCount l10dlnCount l11dlnCount l12dlnCount l24dlnCount
453     l1dlnWeekHours l2dlnWeekHours l3dlnWeekHours l4dlnWeekHours l5dlnWeekHours
454     l6dlnWeekHours ///
455     l7dlnWeekHours l8dlnWeekHours l9dlnWeekHours l10dlnWeekHours l11dlnWeekHours
456     l12dlnWeekHours l24dlnWeekHours ///
457     l1dlnHourlyEarnings l2dlnHourlyEarnings l3dlnHourlyEarnings l4dlnHourlyEarnings
458     l5dlnHourlyEarnings l6dlnHourlyEarnings ///
459     l7dlnHourlyEarnings l8dlnHourlyEarnings l9dlnHourlyEarnings l10dlnHourlyEarnings
460     l11dlnHourlyEarnings l12dlnHourlyEarnings l24dlnHourlyEarnings ///
461     l1dlnWeeklyEarnings l2dlnWeeklyEarnings l3dlnWeeklyEarnings l4dlnWeeklyEarnings
462     l5dlnWeeklyEarnings l6dlnWeeklyEarnings ///
463     l7dlnWeeklyEarnings l8dlnWeeklyEarnings l9dlnWeeklyEarnings l10dlnWeeklyEarnings
464     l11dlnWeeklyEarnings l12dlnWeeklyEarnings l24dlnWeeklyEarnings if
465     tin(2011m1,2021m1), ///
466     ncomb(1,4) aic outsample(24) fix(m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11) ///
467     samesample nindex( -1 aic -1 bic -1 rmse_out)
468     results(gsreg_dlnWeeklyEarnings_Full) replace
469 */
470
471 *log close

```

Log File

```

1      _____
2      ____ (R)      /_   /   ___/   /   ___/
3                      ___/   /   /___/   /   /___/
4
5                      Statistics/Data analysis
6
7      -----
8      --
9          name: <unnamed>
10         log: /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
11 Series/Probl
12         > em Sets/Final Project/Final Project.smcl
13         log type: smcl
14         opened on: 8 Apr 2021, 17:47:08
15
16         1 . import delimited
17           "TS2020_Final_Project_txt2/TS2020_Final_Project_Monthly.txt"
18           (6 vars, 374 obs)
19
20         2 . rename smu12455400500000001 Count
21
22         3 . rename smu12455400500000002 WeekHours
23
24         4 . rename smu12455400500000003 HourlyEarnings
25
26         5 . rename smu12455400500000011 WeeklyEarnings
27
28         6 . rename smu12455400800000001 ServiceCount
29
30         7 .
31         8 .
32         9 . label variable Count "Count"
33
34         10 . label variable WeekHours "WeekHours"
35
36         11 . label variable HourlyEarnings "HourlyEarnings"
37
38         12 . label variable WeeklyEarnings "WeeklyEarnings"
39
40         13 . label variable ServiceCount "ServiceCount"
41
42         14 .
43         15 .
44         16 . gen datec=date(date, "YMD")
45
46         17 . gen Date=mofd(datec)
47
48         18 . gen month=month(datec)

```

```
45
46     19 . format Date %tm
47
48     20 . tsset Date
49           time variable:  Date, 1990m1 to 2021m2
50           delta: 1 month
51
52     21 .
53     22 . gen lnCount = ln(Count)
54
55     23 . gen lnWeekHours = ln(WeekHours)
56           (252 missing values generated)
57
58     24 . gen lnHourlyEarnings = ln(HourlyEarnings)
59           (252 missing values generated)
60
61     25 . gen lnWeeklyEarnings = ln(WeeklyEarnings)
62           (252 missing values generated)
63
64     26 . gen lnServiceCount = ln(ServiceCount)
65
66     27 .
67     28 . gen m1=0
68
69     29 . replace m1=1 if month==1
70           (32 real changes made)
71
72     30 . gen m2=0
73
74     31 . replace m2=1 if month==2
75           (32 real changes made)
76
77     32 . gen m3=0
78
79     33 . replace m3=1 if month==3
80           (31 real changes made)
81
82     34 . gen m4=0
83
84     35 . replace m4=1 if month==4
85           (31 real changes made)
86
87     36 . gen m5=0
88
89     37 . replace m5=1 if month==5
90           (31 real changes made)
91
92     38 . gen m6=0
93
94     39 . replace m6=1 if month==6
95           (31 real changes made)
```

```
96
97     40 . gen m7=0
98
99     41 . replace m7=1 if month==7
100         (31 real changes made)
101
102     42 . gen m8=0
103
104     43 . replace m8=1 if month==8
105         (31 real changes made)
106
107     44 . gen m9=0
108
109     45 . replace m9=1 if month==9
110         (31 real changes made)
111
112     46 . gen m10=0
113
114     47 . replace m10=1 if month==10
115         (31 real changes made)
116
117     48 . gen m11=0
118
119     49 . replace m11=1 if month==11
120         (31 real changes made)
121
122     50 . gen m12=0
123
124     51 . replace m12=1 if month==12
125         (31 real changes made)
126
127     52 .
128     53 . gen dlnCount=d.lnCount
129         (1 missing value generated)
130
131     54 . gen l1dlnCount=l1d.lnCount
132         (2 missing values generated)
133
134     55 . gen l2dlnCount=l2d.lnCount
135         (3 missing values generated)
136
137     56 . gen l3dlnCount=l3d.lnCount
138         (4 missing values generated)
139
140     57 . gen l4dlnCount=l4d.lnCount
141         (5 missing values generated)
142
143     58 . gen l5dlnCount=l5d.lnCount
144         (6 missing values generated)
145
146     59 . gen l6dlnCount=l6d.lnCount
```

```
147         (7 missing values generated)
148
149     60 . gen l7dlnCount=l7d.lnCount
150         (8 missing values generated)
151
152     61 . gen l8dlnCount=l8d.lnCount
153         (9 missing values generated)
154
155     62 . gen l9dlnCount=l9d.lnCount
156         (10 missing values generated)
157
158     63 . gen l10dlnCount=l10d.lnCount
159         (11 missing values generated)
160
161     64 . gen l11dlnCount=l11d.lnCount
162         (12 missing values generated)
163
164     65 . gen l12dlnCount=l12d.lnCount
165         (13 missing values generated)
166
167     66 . gen l24dlnCount=l24d.lnCount
168         (25 missing values generated)
169
170     67 . gen l36dlnCount=l36d.lnCount
171         (37 missing values generated)
172
173     68 . gen l48dlnCount=l48d.lnCount
174         (49 missing values generated)
175
176     69 .
177     70 . gen dlnWeekHours=d.lnWeekHours
178         (253 missing values generated)
179
180     71 . gen l1dlnWeekHours=l1d.lnWeekHours
181         (254 missing values generated)
182
183     72 . gen l2dlnWeekHours=l2d.lnWeekHours
184         (255 missing values generated)
185
186     73 . gen l3dlnWeekHours=l3d.lnWeekHours
187         (256 missing values generated)
188
189     74 . gen l4dlnWeekHours=l4d.lnWeekHours
190         (257 missing values generated)
191
192     75 . gen l5dlnWeekHours=l5d.lnWeekHours
193         (258 missing values generated)
194
195     76 . gen l6dlnWeekHours=l6d.lnWeekHours
196         (259 missing values generated)
197
```



```

198 77 . gen l7dlnWeekHours=l7d.lnWeekHours
199     (260 missing values generated)
200
201 78 . gen l8dlnWeekHours=l8d.lnWeekHours
202     (261 missing values generated)
203
204 79 . gen l9dlnWeekHours=l9d.lnWeekHours
205     (262 missing values generated)
206
207 80 . gen l10dlnWeekHours=l10d.lnWeekHours
208     (263 missing values generated)
209
210 81 . gen l11dlnWeekHours=l11d.lnWeekHours
211     (264 missing values generated)
212
213 82 . gen l12dlnWeekHours=l12d.lnWeekHours
214     (265 missing values generated)
215
216 83 . gen l24dlnWeekHours=l24d.lnWeekHours
217     (277 missing values generated)
218
219 84 . gen l36dlnWeekHours=l36d.lnWeekHours
220     (289 missing values generated)
221
222 85 . gen l48dlnWeekHours=l48d.lnWeekHours
223     (301 missing values generated)
224
225 86 .
226 87 . gen dlnHourlyEarnings=d.lnHourlyEarnings
227     (253 missing values generated)
228
229 88 . gen l1dlnHourlyEarnings=l1d.lnHourlyEarnings
230     (254 missing values generated)
231
232 89 . gen l2dlnHourlyEarnings=l2d.lnHourlyEarnings
233     (255 missing values generated)
234
235 90 . gen l3dlnHourlyEarnings=l3d.lnHourlyEarnings
236     (256 missing values generated)
237
238 91 . gen l4dlnHourlyEarnings=l4d.lnHourlyEarnings
239     (257 missing values generated)
240
241 92 . gen l5dlnHourlyEarnings=l5d.lnHourlyEarnings
242     (258 missing values generated)
243
244 93 . gen l6dlnHourlyEarnings=l6d.lnHourlyEarnings
245     (259 missing values generated)
246
247 94 . gen l7dlnHourlyEarnings=l7d.lnHourlyEarnings
248     (260 missing values generated)

```

```
249
250     95 . gen l8dlnHourlyEarnings=l8d.lnHourlyEarnings
251         (261 missing values generated)
252
253     96 . gen l9dlnHourlyEarnings=l9d.lnHourlyEarnings
254         (262 missing values generated)
255
256     97 . gen l10dlnHourlyEarnings=l10d.lnHourlyEarnings
257         (263 missing values generated)
258
259     98 . gen l11dlnHourlyEarnings=l11d.lnHourlyEarnings
260         (264 missing values generated)
261
262     99 . gen l12dlnHourlyEarnings=l12d.lnHourlyEarnings
263         (265 missing values generated)
264
265    100 . gen l24dlnHourlyEarnings=l24d.lnHourlyEarnings
266         (277 missing values generated)
267
268    101 . gen l36dlnHourlyEarnings=l36d.lnHourlyEarnings
269         (289 missing values generated)
270
271    102 . gen l48dlnHourlyEarnings=l48d.lnHourlyEarnings
272         (301 missing values generated)
273
274    103 .
275    104 . gen dlnWeeklyEarnings=d.lnWeeklyEarnings
276         (253 missing values generated)
277
278    105 . gen l1dlnWeeklyEarnings=l1d.lnWeeklyEarnings
279         (254 missing values generated)
280
281    106 . gen l2dlnWeeklyEarnings=l2d.lnWeeklyEarnings
282         (255 missing values generated)
283
284    107 . gen l3dlnWeeklyEarnings=l3d.lnWeeklyEarnings
285         (256 missing values generated)
286
287    108 . gen l4dlnWeeklyEarnings=l4d.lnWeeklyEarnings
288         (257 missing values generated)
289
290    109 . gen l5dlnWeeklyEarnings=l5d.lnWeeklyEarnings
291         (258 missing values generated)
292
293    110 . gen l6dlnWeeklyEarnings=l6d.lnWeeklyEarnings
294         (259 missing values generated)
295
296    111 . gen l7dlnWeeklyEarnings=l7d.lnWeeklyEarnings
297         (260 missing values generated)
298
299    112 . gen l8dlnWeeklyEarnings=l8d.lnWeeklyEarnings
```

```

300         (261 missing values generated)
301
302     113 . gen l9dlnWeeklyEarnings=l9d.lnWeeklyEarnings
303         (262 missing values generated)
304
305     114 . gen l10dlnWeeklyEarnings=l10d.lnWeeklyEarnings
306         (263 missing values generated)
307
308     115 . gen l11dlnWeeklyEarnings=l11d.lnWeeklyEarnings
309         (264 missing values generated)
310
311     116 . gen l12dlnWeeklyEarnings=l12d.lnWeeklyEarnings
312         (265 missing values generated)
313
314     117 . gen l24dlnWeeklyEarnings=l24d.lnWeeklyEarnings
315         (277 missing values generated)
316
317     118 . gen l36dlnWeeklyEarnings=l36d.lnWeeklyEarnings
318         (289 missing values generated)
319
320     119 . gen l48dlnWeeklyEarnings=l48d.lnWeeklyEarnings
321         (301 missing values generated)
322
323     120 .
324     121 . gen dlnServiceCount=d.lnServiceCount
325         (1 missing value generated)
326
327     122 . gen l1dlnServiceCount=l1d.lnServiceCount
328         (2 missing values generated)
329
330     123 . gen l2dlnServiceCount=l2d.lnServiceCount
331         (3 missing values generated)
332
333     124 . gen l3dlnServiceCount=l3d.lnServiceCount
334         (4 missing values generated)
335
336     125 . gen l4dlnServiceCount=l4d.lnServiceCount
337         (5 missing values generated)
338
339     126 . gen l5dlnServiceCount=l5d.lnServiceCount
340         (6 missing values generated)
341
342     127 . gen l6dlnServiceCount=l6d.lnServiceCount
343         (7 missing values generated)
344
345     128 . gen l7dlnServiceCount=l7d.lnServiceCount
346         (8 missing values generated)
347
348     129 . gen l8dlnServiceCount=l8d.lnServiceCount
349         (9 missing values generated)
350

```

```

351 130 . gen l9dlnServiceCount=l9d.lnServiceCount
352      (10 missing values generated)
353
354 131 . gen l10dlnServiceCount=l10d.lnServiceCount
355      (11 missing values generated)
356
357 132 . gen l11dlnServiceCount=l11d.lnServiceCount
358      (12 missing values generated)
359
360 133 . gen l12dlnServiceCount=l12d.lnServiceCount
361      (13 missing values generated)
362
363 134 . gen l24dlnServiceCount=l24d.lnServiceCount
364      (25 missing values generated)
365
366 135 . gen l36dlnServiceCount=l36d.lnServiceCount
367      (37 missing values generated)
368
369 136 . gen l48dlnServiceCount=l48d.lnServiceCount
370      (49 missing values generated)
371
372 137 .
373 138 . /*
374     > The project is to forecast the March non-seasonally adjusted estimates of
ave
375     > rage weekly earnings and total employment for private employers (total
privat
376     > e) for a Florida MSA of your choice and write up a professional report on
you
377     > r forecast.
378     > */
379 139 . /* Count and WeeklyEarnings */
380 140 .
381 141 . summ Count WeekHours HourlyEarnings WeeklyEarnings ServiceCount
382
383      Variable |           Obs           Mean       Std. Dev.           Min           Max
384      -----+-----
385      Count |           374       14.18556       6.880684           5.3           28
386      WeekHours |           122       36.86967       3.804193           28.3          45.8
387      HourlyEarn~s |           122       19.70344       2.910126           15.01          24.6
388      WeeklyEarn~s |           122       719.7972      84.82529          503.79          916.1
389      ServiceCount |           374       10.40455       5.940013           3.9           22.8
390
391 142 . summ lnCount lnWeekHours lnHourlyEarnings lnWeeklyEarnings lnServiceCount
392
393      Variable |           Obs           Mean       Std. Dev.           Min           Max
394      -----+-----
395      lnCount |           374         2.5174       .5398403       1.667707       3.332205
396      lnWeekHours |           122         3.602049       .1041722       3.342862       3.824284
397      lnHourlyEa~s |           122         2.969891       .148565       2.708717       3.202746
398      lnWeeklyEa~s |           122         6.57194       .1198394       6.222159       6.820126

```

```

399      lnServiceC~t |          374      2.16967      .5975865      1.360977      3.12676
400
401      143 .
402      144 . ac lnCount, saving(lnCount_ac, replace)
403          (file lnCount_ac.gph saved)
404
405      145 . pac lnCount, saving(lnCount_pac, replace)
406          (file lnCount_pac.gph saved)
407
408      146 . graph combine lnCount_ac.gph lnCount_pac.gph, saving(lnCount_ac_pac,
replace)
409          (file lnCount_ac_pac.gph saved)
410
411      147 . graph export "lnCount_ac_pac.png", replace
412          (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time Series/Problem
Sets
413          > /Final Project/lnCount_ac_pac.png written in PNG format)
414
415      148 . ** Probably need to difference
416      149 .
417      150 . ac lnWeeklyEarnings, saving(lnWeeklyEarnings_ac, replace)
418          (file lnWeeklyEarnings_ac.gph saved)
419
420      151 . pac lnWeeklyEarnings, saving(lnWeeklyEarnings_pac, replace)
421          (file lnWeeklyEarnings_pac.gph saved)
422
423      152 . graph combine lnWeeklyEarnings_ac.gph lnWeeklyEarnings_pac.gph,
saving(lnWeek
424          > lyEarnings_ac_pac, replace)
425          (file lnWeeklyEarnings_ac_pac.gph saved)
426
427      153 . graph export "lnWeeklyEarnings_ac_pac.png", replace
428          (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time Series/Problem
Sets
429          > /Final Project/lnWeeklyEarnings_ac_pac.png written in PNG format)
430
431      154 . ** Probably need to differencen b
432      155 .
433      156 . *starter models for count
434      157 . *I used a pair plot to examine the rise and fall of variables with respect
to
435          > each other
436      158 . reg d.lnCount l(12,24,36,48)d.lnCount // .01637
437
438      Source |          SS          df          MS      Number of obs      =
325
439      -----+-----
16.54
440      Model |      .017539188          4      .004384797      Prob > F          =
0.0000

```

```

441      Residual | .084856979      320 .000265178  R-squared      =
0.1713
442      -----+-----
443      Total | .102396167      324 .000316038  Root MSE      =
.01628
444
445      -----
-
446      D.lnCount |      Coef.   Std. Err.      t    P>|t|      [95% Conf.
Interval]
447      -----+-----
-
448      lnCount |
449      L12D. | .3609966   .0621085     5.81   0.000     .238804
.4831893
450      L24D. | .137848   .0617615     2.23   0.026     .016338
.259358
451      L36D. | -.0160136  .0614584    -0.26   0.795    -.1369272
.1049
452      L48D. | .1265117   .0585322     2.16   0.031     .0113551
.2416683
453      |
454      _cons | .0017116   .0009853     1.74   0.083    -.0002269
.0036502
455      -----
-
456
457      159 . scalar drop _all
458
459      160 . quietly forval w=12(12)144 {
460
461      161 . scalar list
462      RWmaxobs144 =      144
463      RWminobs144 =      12
464      RWrmse144 = .0172276
465      RWmaxobs132 =      132
466      RWminobs132 =      12
467      RWrmse132 = .0172128
468      RWmaxobs120 =      120
469      RWminobs120 =      12
470      RWrmse120 = .01721825
471      RWmaxobs108 =      108
472      RWminobs108 =      12
473      RWrmse108 = .01723674
474      RWmaxobs96 =      96
475      RWminobs96 =      12
476      RWrmse96 = .01722006
477      RWmaxobs84 =      84
478      RWminobs84 =      12
479      RWrmse84 = .01726063

```

```

480      RWmaxobs72 =      72
481      RWminobs72 =      12
482      RWrmse72 = .01722377
483      RWmaxobs60 =      60
484      RWminobs60 =      12
485      RWrmse60 = .0173443
486      RWmaxobs48 =      48
487      RWminobs48 =      12
488      RWrmse48 = .01755803
489      RWmaxobs36 =      36
490      RWminobs36 =      12
491      RWrmse36 = .01805924
492      RWmaxobs24 =      24
493      RWminobs24 =      12
494      RWrmse24 = .0185871
495      RWmaxobs12 =      12
496      RWminobs12 =      12
497      RWrmse12 = .02320505
498
499      162 . /*
500          > RWmaxobs132 =      132
501          > RWminobs132 =      12
502          > RWrmse132 = .0172128
503          > */
504      163 .
505      164 . reg d.lnCount l(5,12,24,36,48)d.lnCount l(5)d.lnWeekHours m5 // .01711
506
507          Source |      SS      df      MS      Number of obs      =
116
508      -----+----- F(7, 108)      =
5.94
509          Model | .012171566      7 .001738795      Prob > F      =
0.0000
510          Residual | .03162877      108 .000292859      R-squared      =
0.2779
511      -----+----- Adj R-squared      =
0.2311
512          Total | .043800336      115 .000380872      Root MSE      =
.01711
513
514      -----
515          D.lnCount |      Coef.      Std. Err.      t      P>|t|      [95% Conf.
Interval]
516      -----+-----
517          lnCount |
518          L5D. | -.1231921      .0845717      -1.46      0.148      -.290828
.0444438
519          L12D. | .5811114      .1685831      3.45      0.001      .2469504
.9152724

```

```

520          L24D. | -.1196017 .1627467 -0.73 0.464 -.4421938
      .2029904
521          L36D. | .2532303 .1742525 1.45 0.149 -.0921684
      .5986291
522          L48D. | .1341638 .1858633 0.72 0.472 -.2342495
      .5025771
523          |
524      lnWeekHours |
525          L5D. | .0170123 .0364906 0.47 0.642 -.0553184
      .089343
526          |
527          m5 | .0067588 .0061605 1.10 0.275 -.0054524
      .0189699
528          _cons | .0004279 .0018229 0.23 0.815 -.0031854
      .0040412
529      -----
      -
530
531      165 . scalar drop _all
532
533      166 . quietly forval w=12(12)84 {
534
535          167 . scalar list
536              RWmaxobs84 =      84
537              RWminobs84 =      23
538              RWrmse84 = .01950911
539              RWmaxobs72 =      72
540              RWminobs72 =      23
541              RWrmse72 = .01949719
542              RWmaxobs60 =      60
543              RWminobs60 =      23
544              RWrmse60 = .0199438
545              RWmaxobs48 =      48
546              RWminobs48 =      23
547              RWrmse48 = .02035982
548              RWmaxobs36 =      36
549              RWminobs36 =      23
550              RWrmse36 = .02138785
551              RWmaxobs24 =      24
552              RWminobs24 =      23
553              RWrmse24 = .02268585
554              RWmaxobs12 =      12
555              RWminobs12 =      12
556              RWrmse12 = .05004898
557
558      168 . /*
559          > RWmaxobs84 =      84
560          > RWminobs84 =      23
561          > RWrmse84 = .01950911
562          > */
563      169 .

```



```

564 170 . /*
565 > gsreg dlnCount l1dlnCount l2dlnCount l3dlnCount l4dlnCount l5dlnCount
16dlnCo
566 > unt ///
567 > 17dlnCount l8dlnCount l9dlnCount l10dlnCount l11dlnCount
l12dlnCount
568 > ///
569 > 124dlnCount l36dlnCount l48dlnCount ///
570 > if tin(1990m1,2021m1), ///
571 > ncomb(1,12) aic outsample(24) fix(m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
m11
572 > m12) ///
573 > samesample nindex( -1 aic -1 bic -1 rmse_out)
results(gsreg_dlnCount)
574 > replace
575 > */
576 171 .
577 172 . *gsreg suggestions
578 173 . reg d.lnCount l12d.lnCount m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
579
580 Source | SS df MS Number of obs =
361
581 -----+----- F(12, 348) =
6.91
582 Model | .022616974 12 .001884748 Prob > F =
0.0000
583 Residual | .094871179 348 .000272618 R-squared =
0.1925
584 -----+----- Adj R-squared =
0.1647
585 Total | .117488153 360 .000326356 Root MSE =
.01651
586
587 -----
588 D.lnCount | Coef. Std. Err. t P>|t| [95% Conf.
Interval]
589 -----+-----
590 lnCount |
591 L12D. | .1748571 .0594184 2.94 0.003 .0579928
.2917215
592 |
593 m1 | -.0099477 .0043004 -2.31 0.021 -.0184056
-.0014897
594 m2 | .0009939 .0042297 0.23 0.814 -.0073251
.0093129
595 m3 | .0030247 .0042759 0.71 0.480 -.0053851
.0114345
596 m4 | -.0071933 .0042648 -1.69 0.093 -.0155814
.0011948

```

```

597          m5 |   -0.0098194   .0043178   -2.27   0.024   -0.0183118
      -.0013271
598          m6 |   -0.0133285   .0043874   -3.04   0.003   -0.0219576
      -.0046994
599          m7 |   -0.0091828   .0042967   -2.14   0.033   -0.0176336
      -.000732
600          m8 |   -0.0017998   .0042632   -0.42   0.673   -0.0101846
      .006585
601          m9 |   -0.006737   .0042824   -1.57   0.117   -0.0151597
      .0016858
602          m10 |   .0062149   .0042795    1.45   0.147   -0.0022021
      .0146319
603          m11 |   .0042124   .0042811    0.98   0.326   -0.0042078
      .0126325
604          _cons |   .0072199   .0030452    2.37   0.018   .0012306
      .0132093
605          -----
      -
606
607      174 . scalar drop _all
608
609      175 . quietly forval w=12(12)144 {
610
611      176 . scalar list
612          RWmaxobs144 =          144
613          RWminobs144 =           12
614          RWrmse144 =   .01824906
615          RWmaxobs132 =          132
616          RWminobs132 =           12
617          RWrmse132 =   .01832173
618          RWmaxobs120 =          120
619          RWminobs120 =           12
620          RWrmse120 =   .01833557
621          RWmaxobs108 =          108
622          RWminobs108 =           12
623          RWrmse108 =   .01841089
624          RWmaxobs96 =           96
625          RWminobs96 =           12
626          RWrmse96 =   .01836974
627          RWmaxobs84 =           84
628          RWminobs84 =           12
629          RWrmse84 =   .01849267
630          RWmaxobs72 =           72
631          RWminobs72 =           12
632          RWrmse72 =   .01861349
633          RWmaxobs60 =           60
634          RWminobs60 =           12
635          RWrmse60 =   .01911515
636          RWmaxobs48 =           48
637          RWminobs48 =           12
638          RWrmse48 =   .01922268

```

```

639      RWmaxobs36 =      36
640      RWminobs36 =      12
641      RWrmse36 = .01991683
642      RWmaxobs24 =      24
643      RWminobs24 =      12
644      RWrmse24 = .02022186
645      RWmaxobs12 =      12
646      RWminobs12 =      12
647      RWrmse12 = .02009249
648
649      177 . /*
650          > RWmaxobs144 =      144
651          > RWminobs144 =      12
652          > RWrmse144 = .01824906
653          > */
654      178 .
655      179 . reg d.lnCount l(12,36)d.lnCount m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
656
657          Source |           SS           df           MS       Number of obs   =
337
658      -----+-----
5.68
659          Model |   .019946057           13   .001534312       Prob > F           =
0.0000
660          Residual |   .087185203          323   .000269923       R-squared           =
0.1862
661      -----+-----
0.1534
662          Total |   .107131259          336   .000318843       Root MSE           =
.01643
663
664      -----
-
665          D.lnCount |           Coef.   Std. Err.      t    P>|t|     [95% Conf.
Interval]
666      -----+-----
-
667          lnCount |
668          L12D. |   .1849403   .0636401     2.91   0.004     .059739
.3101417
669          L36D. |  -.049332   .0606582    -0.81   0.417    -.1686671
.0700031
670          |
671          m1 |  -.0073418   .0044769    -1.64   0.102    -.0161493
.0014658
672          m2 |   .0022711   .0043559     0.52   0.602    -.0062984
.0108407
673          m3 |   .0043593   .004416     0.99   0.324    -.0043285
.0130471
674          m4 |  -.0065438   .0043922    -1.49   0.137    -.0151847
.002097

```

```

675          m5 |   -.0089215   .0045194   -1.97   0.049   -.0178126
        -.0000304
676          m6 |   -.0133453   .0046241   -2.89   0.004   -.0224425
        -.004248
677          m7 |   -.0085154   .004457   -1.91   0.057   -.0172839
        .0002531
678          m8 |   -.0004554   .004392   -0.10   0.917   -.0090959
        .0081852
679          m9 |   -.0056625   .0044299   -1.28   0.202   -.0143775
        .0030526
680          m10 |   .0071688   .0044386   1.62   0.107   -.0015635
        .0159011
681          m11 |   .0042074   .0044259   0.95   0.343   -.0044998
        .0129146
682          _cons |   .0067355   .0031722   2.12   0.034   .0004948
        .0129762
683          -----
        -
684
685      180 . scalar drop _all
686
687      181 . quietly forval w=12(12)144 {
688
689      182 . scalar list
690          RWmaxobs144 =          144
691          RWminobs144 =           12
692          RWrmse144 =   .01777071
693          RWmaxobs132 =          132
694          RWminobs132 =           12
695          RWrmse132 =   .01782557
696          RWmaxobs120 =          120
697          RWminobs120 =           12
698          RWrmse120 =   .01785253
699          RWmaxobs108 =          108
700          RWminobs108 =           12
701          RWrmse108 =   .01794692
702          RWmaxobs96 =           96
703          RWminobs96 =           12
704          RWrmse96 =   .01793358
705          RWmaxobs84 =           84
706          RWminobs84 =           12
707          RWrmse84 =   .01803355
708          RWmaxobs72 =           72
709          RWminobs72 =           12
710          RWrmse72 =   .01807408
711          RWmaxobs60 =           60
712          RWminobs60 =           12
713          RWrmse60 =   .01843535
714          RWmaxobs48 =           48
715          RWminobs48 =           12
716          RWrmse48 =   .01835092

```

```

717     RWmaxobs36 =      36
718     RWminobs36 =      12
719     RWrmse36 = .01863303
720     RWmaxobs24 =      24
721     RWminobs24 =      12
722     RWrmse24 = .0196745
723     RWmaxobs12 =      12
724     RWminobs12 =      12
725     RWrmse12 = .01880291
726
727     183 . /*
728         > RWmaxobs144 =      144
729         > RWminobs144 =      12
730         > RWrmse144 = .01777071
731         > */
732     184 .
733     185 . /*
734         > gsreg dlnCount l1dlnCount l2dlnCount l3dlnCount l4dlnCount l5dlnCount
16dlnCo
735         > unt ///
736         >      17dlnCount l8dlnCount l9dlnCount l10dlnCount l11dlnCount
l12dlnCount
737         > ///
738         >      124dlnCount l36dlnCount ///
739         >      11dlnWeekHours l2dlnWeekHours l3dlnWeekHours l4dlnWeekHours
l5dlnWeek
740         > Hours l6dlnWeekHours ///
741         >      17dlnWeekHours l8dlnWeekHours l9dlnWeekHours l10dlnWeekHours
l11dlnWe
742         > ekHours l12dlnWeekHours ///
743         >      124dlnWeekHours l36dlnWeekHours ///
744         >      11dlnHourlyEarnings l2dlnHourlyEarnings l3dlnHourlyEarnings
l4dlnHour
745         > lyEarnings ///
746         >      15dlnHourlyEarnings l6dlnHourlyEarnings ///
747         >      17dlnHourlyEarnings l8dlnHourlyEarnings l9dlnHourlyEarnings
l10dlnHou
748         > rlyEarnings ///
749         >      111dlnHourlyEarnings l12dlnHourlyEarnings ///
750         >      124dlnHourlyEarnings l36dlnHourlyEarnings ///
751         >      11dlnWeeklyEarnings l2dlnWeeklyEarnings l3dlnWeeklyEarnings
l4dlnWeek
752         > lyEarnings ///
753         >      15dlnWeeklyEarnings l6dlnWeeklyEarnings ///
754         >      17dlnWeeklyEarnings l8dlnWeeklyEarnings l9dlnWeeklyEarnings
l10dlnWee
755         > klyEarnings ///
756         >      111dlnWeeklyEarnings l12dlnWeeklyEarnings ///
757         >      124dlnWeeklyEarnings l36dlnWeeklyEarnings ///
758         >      if tin(2011m1,2021m1), ///

```

```

759 > ncomb(1,4) aic outsample(24) fix(m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
m11)
760 > ///
761 > samesample nindex( -1 aic -1 bic -1 rmse_out)
results(gsreg_dlnCount_
762 > Full) replace
763 > */
764 186 .
765 187 . reg d.lnCount l4d.lnWeekHours l9d.lnWeekHours l8d.lnHourlyEarnings m1 m2
m3 m
766 > 4 m5 m6 m7 m8 m9 m10 m11
767
768 Source | SS df MS Number of obs =
112
769 -----+----- F(14, 97) =
3.24
770 Model | .013917393 14 .0009941 Prob > F =
0.0003
771 Residual | .029798432 97 .0003072 R-squared =
0.3184
772 -----+----- Adj R-squared =
0.2200
773 Total | .043715825 111 .000393836 Root MSE =
.01753
774
775 -----
-
776 D.lnCount | Coef. Std. Err. t P>|t| [95% Conf.
Interval]
777 -----+-----
-
778 lnWeekHours |
779 L4D. | -.0013847 .0384012 -0.04 0.971 -.0776005
.074831
780 L9D. | .0397686 .0385964 1.03 0.305 -.0368346
.1163718
781 |
782 lnHourlyEa~s |
783 L8D. | -.039029 .0414024 -0.94 0.348 -.1212014
.0431433
784 |
785 m1 | -.0097045 .0078517 -1.24 0.219 -.0252879
.0058789
786 m2 | .0000949 .0079445 0.01 0.990 -.0156727
.0158626
787 m3 | -.004712 .0083585 -0.56 0.574 -.0213013
.0118773
788 m4 | -.0273667 .0081729 -3.35 0.001 -.0435876
-.0111459
789 m5 | -.0076836 .0081259 -0.95 0.347 -.0238112
.008444

```

```

790          m6 |   -.020254   .0081465   -2.49   0.015   -.0364227
      -.0040854
791          m7 |   -.0130812   .0081852   -1.60   0.113   -.0293265
      .0031642
792          m8 |    .0041701   .0081051    0.51   0.608   -.0119164
      .0202565
793          m9 |   -.0089171   .0082764   -1.08   0.284   -.0253435
      .0075093
794         m10 |    .0153608   .0081153    1.89   0.061   -.0007459
      .0314674
795         m11 |    .0040463   .0079619    0.51   0.612   -.0117559
      .0198485
796        _cons |    .0094122   .0056462    1.67   0.099   -.0017939
      .0206183
797        -----
798        -
799      188 . scalar drop _all
800
801      189 . quietly forval w=12(12)84 {
802
803      190 . scalar list
804          RWmaxobs84 =          84
805          RWminobs84 =           2
806          RWrmse84 =   .01847546
807          RWmaxobs72 =          72
808          RWminobs72 =           2
809          RWrmse72 =   .01855448
810          RWmaxobs60 =          60
811          RWminobs60 =           2
812          RWrmse60 =   .01850723
813          RWmaxobs48 =          48
814          RWminobs48 =           2
815          RWrmse48 =   .01850217
816          RWmaxobs36 =          36
817          RWminobs36 =           2
818          RWrmse36 =   .01942535
819          RWmaxobs24 =          24
820          RWminobs24 =           2
821          RWrmse24 =   .02208272
822          RWmaxobs12 =          12
823          RWminobs12 =           2
824          RWrmse12 =   .0176238
825
826      191 . /*
827          > RWmaxobs12 =          12
828          > RWminobs12 =           2
829          > RWrmse12 =   .0176238
830          > */
831      192 .

```

```

832 193 . /*-----
833 ---
834 > -*/
835 194 .
836 195 . *starter models for weekly earnings
837 196 . reg d.lnWeeklyEarnings l1d.lnWeekHours ld.lnHourlyEarnings
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868

```

Source	SS	df	MS	Number of obs	=
Model	.00720643	2	.003603215	Prob > F	=
Residual	.284697808	117	.002433315	R-squared	=
Total	.291904237	119	.002452977	Root MSE	=

	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
lnWeeklyEa~s					
LD.	-.1344969	.105932	-1.27	0.207	-.3442897
lnHourlyEa~s					
LD.	.0732899	.1167768	0.63	0.531	-.1579805
_cons	.0016809	.0045103	0.37	0.710	-.0072515

```

197 . scalar drop _all
198 . quietly forval w=12(12)84 {
199 . scalar list
RWmaxobs84 =      84
RWminobs84 =       2
RWrmse84 = .06184586
RWmaxobs72 =      72
RWminobs72 =       2
RWrmse72 = .06163593

```



```

869     RWmaxobs60 =      60
870     RWminobs60 =       2
871     RWrmse60 = .06145693
872     RWmaxobs48 =      48
873     RWminobs48 =       2
874     RWrmse48 = .06185207
875     RWmaxobs36 =      36
876     RWminobs36 =       2
877     RWrmse36 = .06201342
878     RWmaxobs24 =      24
879     RWminobs24 =       2
880     RWrmse24 = .06223845
881     RWmaxobs12 =      12
882     RWminobs12 =       2
883     RWrmse12 = .06581989
884
885     200 . /*
886         > RWmaxobs60 =      60
887         > RWminobs60 =       2
888         > RWrmse60 = .06145693
889         > */
890     201 .
891     202 . /*
892         > gsreg dlnWeeklyEarnings l1dlnWeeklyEarnings l2dlnWeeklyEarnings
13dlnWeeklyEa
893         > rnings ///
894         >         14dlnWeeklyEarnings 15dlnWeeklyEarnings 16dlnWeeklyEarnings ///
895         >         17dlnWeeklyEarnings 18dlnWeeklyEarnings 19dlnWeeklyEarnings
110dlnWee
896         > klyEarnings ///
897         >         111dlnWeeklyEarnings 112dlnWeeklyEarnings ///
898         >         124dlnWeeklyEarnings 136dlnWeeklyEarnings ///
899         >         if tin(2011m1,2021m1), ///
900         >         ncomb(1,12) aic outsample(24) ///
901         >         samesample nindex( -1 aic -1 bic -1 rmse_out)
results(gsreg_dlnWeekly
902         > Earnings) replace
903         > */
904     203 .
905     204 . reg d.lnWeeklyEarnings l3d.lnWeeklyEarnings l5d.lnWeeklyEarnings m1 m2 m3
m4
906         > m5 m6 m7 m8 m9 m10 m11
907
908         Source |      SS      df      MS      Number of obs      =
116
909         -----+-----
2.22
910         Model | .062824046      13   .004832619      Prob > F      =
0.0134
911         Residual | .221893915     102   .002175431      R-squared      =
0.2207

```

```

912 -----+----- Adj R-squared =
0.1213
913 Total | .284717961 115 .002475808 Root MSE =
.04664
914
915 -----
-
916 D. |
917 lnWeeklyEa~s | Coef. Std. Err. t P>|t| [95% Conf.
Interval]
918 -----+-----
-
919 lnWeeklyEa~s |
920 L3D. | -.230666 .0948354 -2.43 0.017 -.4187716
-.0425604
921 L5D. | -.1701094 .0948714 -1.79 0.076 -.3582864
.0180676
922 |
923 m1 | -.0129295 .0212711 -0.61 0.545 -.0551207
.0292616
924 m2 | .0230655 .0211959 1.09 0.279 -.0189765
.0651074
925 m3 | -.0120234 .0220335 -0.55 0.586 -.0557268
.03168
926 m4 | .0106776 .0218502 0.49 0.626 -.0326621
.0540173
927 m5 | .0379684 .0215915 1.76 0.082 -.0048583
.0807952
928 m6 | .0272323 .0215653 1.26 0.210 -.0155424
.0700069
929 m7 | -.0216364 .0213979 -1.01 0.314 -.064079
.0208063
930 m8 | .0154331 .0210082 0.73 0.464 -.0262366
.0571028
931 m9 | .0206819 .021546 0.96 0.339 -.0220544
.0634182
932 m10 | .0210709 .0217906 0.97 0.336 -.0221507
.0642924
933 m11 | .0089656 .0216561 0.41 0.680 -.0339891
.0519203
934 _cons | -.0075658 .0151177 -0.50 0.618 -.0375517
.0224201
935 -----
-
936
937 205 . scalar drop _all
938
939 206 . quietly forval w=12(12)84 {
940
941 207 . scalar list
942 RWmaxobs84 = 84

```

```

943      RWminobs84 =          2
944      RWrmse84 = .06004448
945      RWmaxobs72 =          72
946      RWminobs72 =          2
947      RWrmse72 = .0605325
948      RWmaxobs60 =          60
949      RWminobs60 =          2
950      RWrmse60 = .06068574
951      RWmaxobs48 =          48
952      RWminobs48 =          2
953      RWrmse48 = .06037733
954      RWmaxobs36 =          36
955      RWminobs36 =          2
956      RWrmse36 = .06130591
957      RWmaxobs24 =          24
958      RWminobs24 =          2
959      RWrmse24 = .06544875
960      RWmaxobs12 =          12
961      RWminobs12 =          2
962      RWrmse12 = .07012904
963
964      208 . /*
965          > RWmaxobs84 =          84
966          > RWminobs84 =          2
967          > RWrmse84 = .06004448
968          > */
969      209 .
970      210 . reg d.lnWeeklyEarnings l3d.lnWeeklyEarnings l5d.lnWeeklyEarnings
17d.lnWeekly
971          > Earnings
972
973          Source |          SS          df          MS      Number of obs      =
114
974      -----+----- F(3, 110)      =
4.04
975          Model | .027592174          3 .009197391      Prob > F      =
0.0091
976          Residual | .250484433        110 .002277131      R-squared      =
0.0992
977      -----+----- Adj R-squared      =
0.0747
978          Total | .278076607        113 .002460855      Root MSE      =
.04772
979
980      -----
-
981      D.          |
982      lnWeeklyEa~s |          Coef.      Std. Err.      t      P>|t|      [95% Conf.
Interval]
983      -----+-----
-

```

```

984      lnWeeklyEa~s |
985          L3D. | -.2446256 .0905231 -2.70 0.008 -.4240211
-.0652301
986          L5D. | -.1957955 .0901769 -2.17 0.032 -.3745049
-.0170861
987          L7D. | .0649411 .0912448 0.71 0.478 -.1158846
.2457668
988          |
989          _cons | .0027599 .0044776 0.62 0.539 -.0061138
.0116335
990      -----
-
991
992      211 . scalar drop _all
993
994      212 . quietly forval w=12(12)84 {
995
996      213 . scalar list
997          RWmaxobs84 =      84
998          RWminobs84 =      2
999          RWrmse84 = .05250414
1000          RWmaxobs72 =      72
1001          RWminobs72 =      2
1002          RWrmse72 = .05277312
1003          RWmaxobs60 =      60
1004          RWminobs60 =      2
1005          RWrmse60 = .05308846
1006          RWmaxobs48 =      48
1007          RWminobs48 =      2
1008          RWrmse48 = .05299891
1009          RWmaxobs36 =      36
1010          RWminobs36 =      2
1011          RWrmse36 = .05349229
1012          RWmaxobs24 =      24
1013          RWminobs24 =      2
1014          RWrmse24 = .05283688
1015          RWmaxobs12 =      12
1016          RWminobs12 =      2
1017          RWrmse12 = .06056213
1018
1019      214 . /*
1020          > RWmaxobs84 =      84
1021          > RWminobs84 =      2
1022          > RWrmse84 = .05250414
1023          > */
1024      215 .
1025      216 . /*
1026          > gsreg dlnCount l1dlnCount l2dlnCount l3dlnCount l4dlnCount l5dlnCount
l6dlnCo
1027          > unt ///

```

```

1028         >          17dlnCount 18dlnCount 19dlnCount 110dlnCount 111dlnCount
112dlnCount
1029         > 124dlnCount ///
1030         >          11dlnWeekHours 12dlnWeekHours 13dlnWeekHours 14dlnWeekHours
15dlnWeek
1031         > Hours 16dlnWeekHours ///
1032         >          17dlnWeekHours 18dlnWeekHours 19dlnWeekHours 110dlnWeekHours
111dlnWe
1033         > ekHours 112dlnWeekHours 124dlnWeekHours ///
1034         >          11dlnHourlyEarnings 12dlnHourlyEarnings 13dlnHourlyEarnings
14dlnHour
1035         > lyEarnings ///
1036         >          15dlnHourlyEarnings 16dlnHourlyEarnings ///
1037         >          17dlnHourlyEarnings 18dlnHourlyEarnings 19dlnHourlyEarnings
110dlnHou
1038         > rlyEarnings ///
1039         >          111dlnHourlyEarnings 112dlnHourlyEarnings 124dlnHourlyEarnings ///
1040         >          11dlnWeeklyEarnings 12dlnWeeklyEarnings 13dlnWeeklyEarnings
14dlnWeek
1041         > lyEarnings ///
1042         >          15dlnWeeklyEarnings 16dlnWeeklyEarnings ///
1043         >          17dlnWeeklyEarnings 18dlnWeeklyEarnings 19dlnWeeklyEarnings
110dlnWee
1044         > klyEarnings ///
1045         >          111dlnWeeklyEarnings 112dlnWeeklyEarnings 124dlnWeeklyEarnings if
tin
1046         > (2011m1,2021m1), ///
1047         >          ncomb(1,4) aic outsample(24) fix(m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
m11)
1048         > ///
1049         >          samesample nindex( -1 aic -1 bic -1 rmse_out)
results(gsreg_dlnWeekly
1050         > Earnings_Full) replace
1051         > */
1052     217 .
1053     218 . log close
1054         name: <unnamed>
1055         log: /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Probl
1056         > em Sets/Final Project/Final Project.smcl
1057         log type: smcl
1058         closed on: 8 Apr 2021, 17:48:56
1059         -----
--

```

