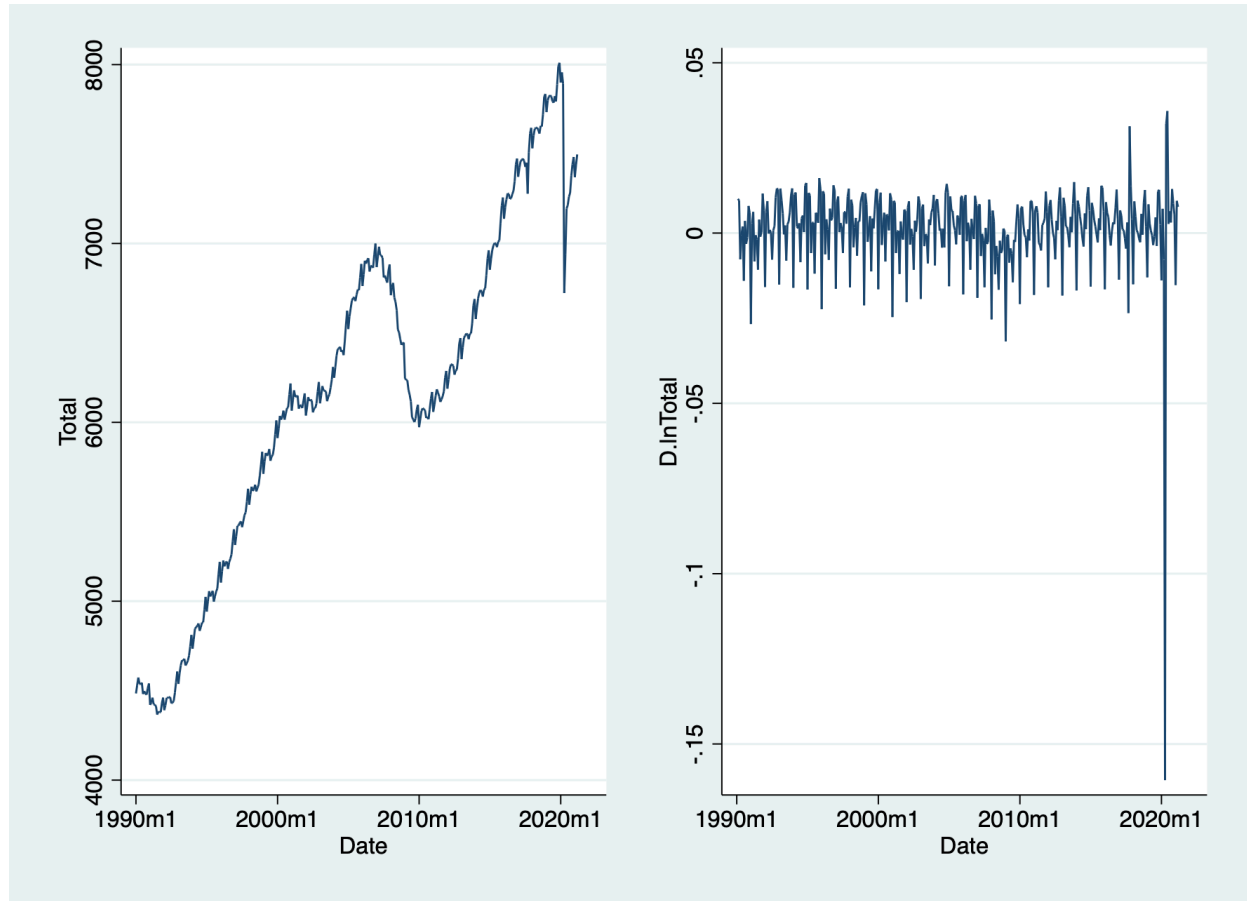


Part 1

Time Series Plots

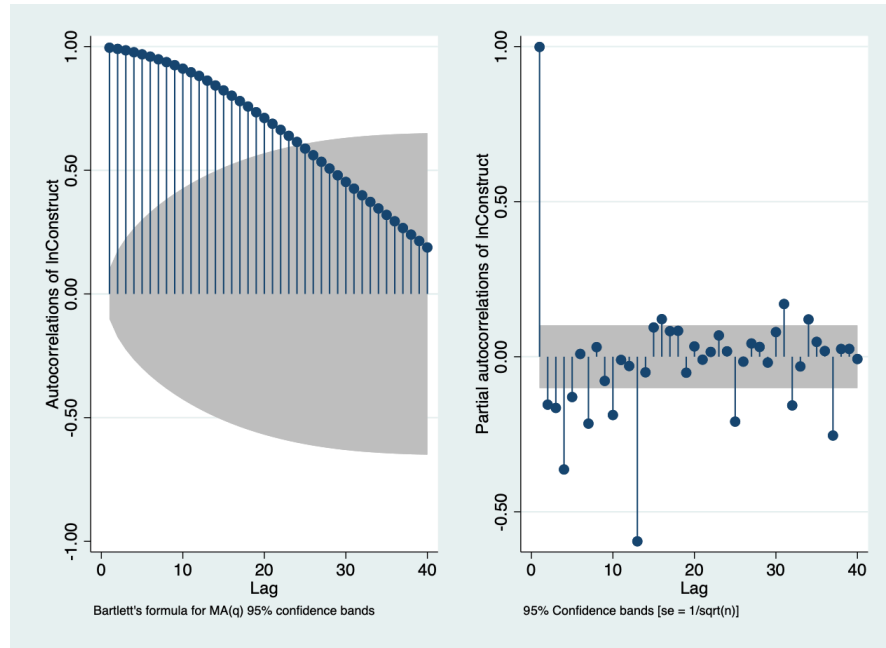


No Logs

You do not want to take logs because that adjusts the distance between values and normalizes the data. Without the distances between data being preserved, there is no good way to measure proportions unless the log transformed data has identical transforms performed (the data going in would have to be the same).

Persistent Data

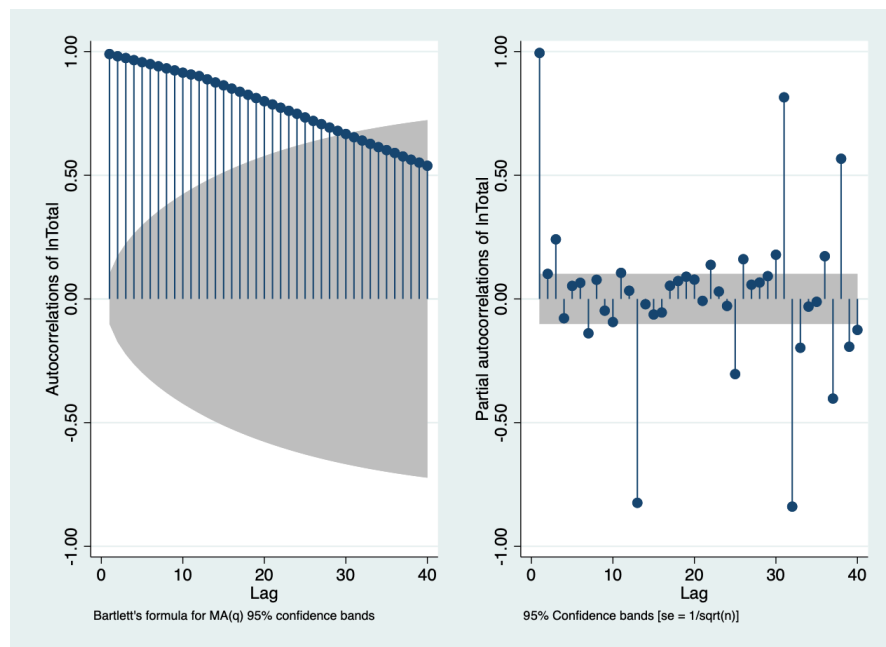
Construct



Both first lags are high which means we should difference.

Test	Result
Mackinnon approximate p-value for Z(t)	0.9796

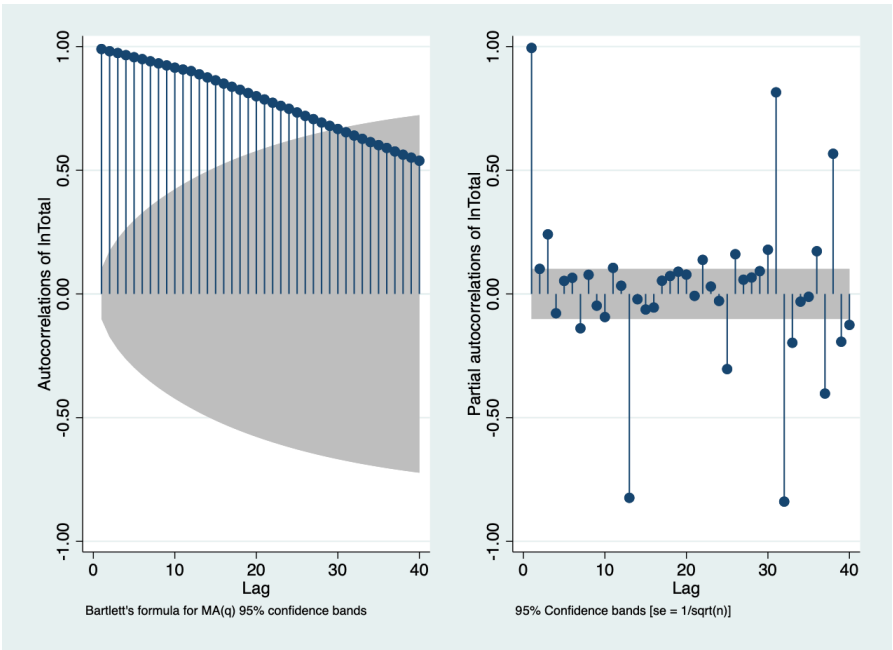
Manufacture



Both first lags are high which means we should difference.

Test	Result
Mackinnon approximate p-value for Z(t)	0.9963

Total

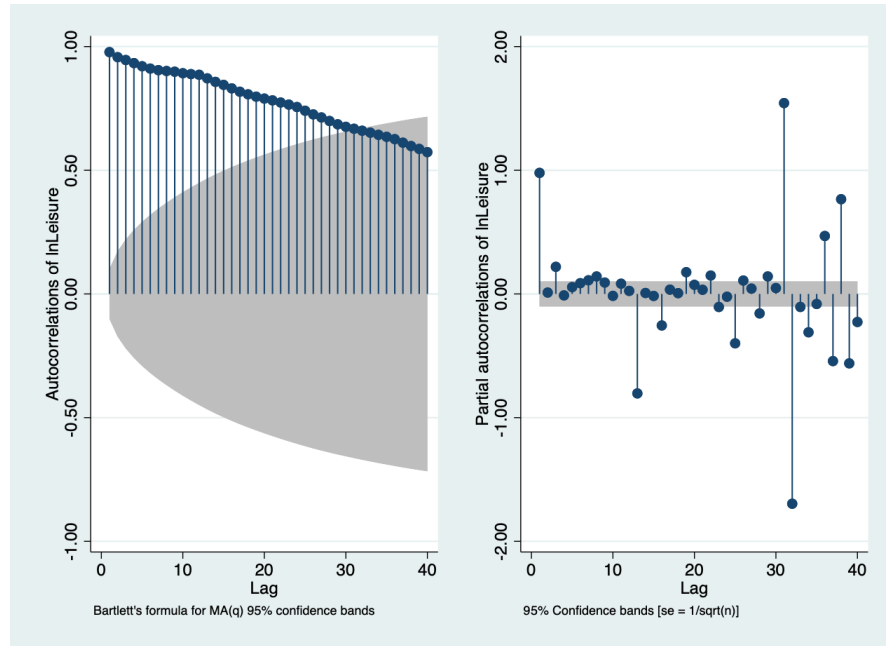


Both first lags are high which means we should difference.

Test	Result
Mackinnon approximate p-value for Z(t)	0.6570

Even though the dickey fuller test says we do not need to difference for construct, manufacture, and total, I'm still going to because I already wrote everything with it differenced the lags outside the confidence intervals in the PAC quite far apart and I can't imagine that there is really that much of an impact on current employment by employment from 30+ months ago. The only ones I can maybe understand are the 12 and 24 moth lags because employers will use historical data when hiring seasonal employees.

Leisure



Both first lags are high which means we should difference.

Test	Result
Mackinnon approximate p-value for Z(t)	0.0005

The p value for leisure is below .05 so I don't need an excuse to difference this time. However, it must be noted that lags 31 and 32 are extremely high.

Serial Correlation

Because not all lags are within the 95% confidence interval, we should worry at least a little bit about serial correlation. However, as I suggested earlier, the only lags that could have a true impact are the 12th and 24th lags and so I am not worried about serial correlation.

Dropping Lags

	Newey-West					
D.InTotal	Coef.	Std. Err.	t	P>t	[95% Conf.	Interval]
Construct						
D1.	.0004252	.0000809	5.26	0.000	.0002661	.0005842
LD.	.0000247	.000044	0.56	0.574	-.0000617	.0001112
L2D.	.0000613	.0000403	1.52	0.129	-.0000179	.0001405
L3D.	-.0000674	.0000577	-1.17	0.244	-.000181	.0000462
L12D.	-.000035	.0000641	-0.55	0.585	-.000161	.000091
L24D.	.0000494	.0000684	0.72	0.470	-.0000851	.0001839
Manufacture						
D1.	.0011938	.0001924	6.21	0.000	.0008154	.0015723
LD.	-.0002374	.0001386	-1.71	0.088	-.00051	.0000353
L2D.	-.000336	.0000718	-4.68	0.000	-.0004773	-.0001948
L3D.	-.0002689	.0001059	-2.54	0.012	-.0004772	-.0000606
L12D.	.0003308	.0001381	2.39	0.017	.000059	.0006025
L24D.	.0002403	.0001413	1.70	0.090	-.0000376	.0005182
Leisure						
D1.	.0002134	8.38e-06	25.47	0.000	.0001969	.0002299
LD.	5.34e-06	9.19e-06	0.58	0.561	-.0000127	.0000234
L2D.	-2.42e-06	.0000172	-0.14	0.888	-.0000364	.0000315
L3D.	-8.76e-06	.0000174	-0.50	0.614	-.0000429	.0000254
L12D.	.0000629	.0000361	1.74	0.082	-8.04e-06	.0001339
L24D.	-.000016	.0000428	-0.37	0.708	-.0001002	.0000681
_cons	.0011957	.0003664	3.26	0.001	.000475	.0019165

The Newey-West test suggests we should drop lags 12 and 24.

Test Number	Variable		Test Value
(1)	L24D.Construct	=	0
(2)	L12D.Leisure	=	0
(3)	L24D.Leisure	=	0
(4)	L12D.Manufacture	=	0
(5)	L24D.Manufacture	=	0

`testparm` suggests that there is no interaction between the 12th and 24th lags of each construct, leisure, and manufacture.

Equal Effects

Interaction	Prob > F
Construct ~ Manufacture	0.0161
Construct ~ Leisure	0.0008
Manufacture ~ Leisure	0.0301

The p-value for all three is less than 0.05 which means we can reject the null hypothesis and accept the alternative that the impacts of each variable are not equal.

Part 2

Differencing, Log Transforms, and Month Dummies

Differencing

See above AC and PAC charts in [Part 1](#).

Log Transforms

Log transforms make the data not have any values less than zero and forces the data into a normal distribution. It also transforms the data so it has proportional changes rather than absolute changes so that any changes over time can be reported as a percent change.

Month Dummies

There's not any reason to not include month dummies. If your data is monthly or any other form of seasonal, it will help your models because they're now identified to a particular season. If your data isn't seasonal, they won't have any effect.

Content Knowledge and Model Searches

Content Knowledge

Content knowledge can speed up the model selection process because you may already have an idea of what variables or lags have an effect on the dependent variable. For example, hourly wages and hours scheduled per week are probably a very good indication of monthly wages.

GSREG

Global search regression takes all the variables you feed it and runs a regression for any combination of the variables. This is a powerful tool to fine-tune your models, but without filtering the variables through content knowledge, it could take a very long time to run. Rather than just taking the highest scoring model, you should then examine common features of the highest scoring models on the basis of AIC, BIC, and out of sample root mean square error, and choose the most parsimonious one.

What's wrong with *stepwise* model selection?

It's prone to over fitting because it has bad predictive properties. Instead you should use out of sample fitting because it protects against over fitting. Over fitting is caused by dropping the most insignificant each step which may include variables that should be included in the model but are not relevant on their own.

Choosing Models

Model Type	Model	AIC	BIC	Root Mean Squared Errors
AR only Lags 1-3 Month dummies	<code>reg d.lnTotal 1(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11</code>	-2382.725	-2323.982	.0129605
AR only Lags 1-3,12,24 Month dummies	<code>reg d.lnTotal 1(1/3,12,24)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11</code>	-2236.218	-2170.633	.01300797
ARDL Lags 1-3 Month dummies	<code>reg d.lnTotal 1(1/3)d.lnTotal 1(1/3)d.lnConstruct 1(1/3)d.lnLeisure 1(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11</code>	-2393.008	-2299.019	.01611154
ARDL Lags 1-3,12,24 Month dummies	<code>reg d.lnTotal 1(1/3,12,24)d.lnTotal 1(1/3,12,24)d.lnConstruct 1(1/3,12,24)d.lnLeisure 1(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11</code>	-2230.781	-2115.043	.01713897

Which are the best two and why?

Model 1 has the lowest root mean squared error and model 3 has the lowest AIC and BIC. I'm also inclined to believe these are the better ones because they don't include lags 12 and 24 which is a long time for subcomponents of the total employment variable to have an effect on the total employment variable.

Rolling Window

Model 1

```
reg d.lnTotal 1(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
```

Value Type	Value
RWmaxobs12	12
RWminobs12	12
RWRmse12	.0132376

A window width of 12 had the lowest RWrmse. I thought that maybe a smaller window width would be better because the lags did not include lag 12 but I was wrong. Besides 12 months, 6 months had the second lowest.

Model 3

```
reg d.lnTotal l(1/3)d.lnTotal l(1/3)d.lnConstruct l(1/3)d.lnLeisure  
l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
```

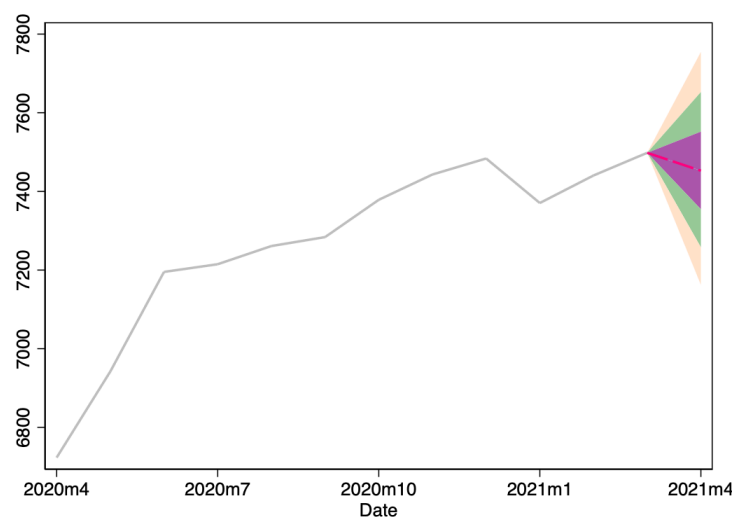
Value Type	Value
RWmaxobs12	12
RWminobs12	12
RWrmse12	.0132376

A window width of 12 had the lowest RWrmse. After my failure in model 1, I tried again hoping for better results. A window width of 12 is still the best.

Ultimately, I'm going to choose model 1 because it is autoregressive and that is what makes ARIMA work and without ARIMA I could not make my pretty fan charts. They have the same RWrmse anyways so I can't imagine the extra variables have too big a difference. And

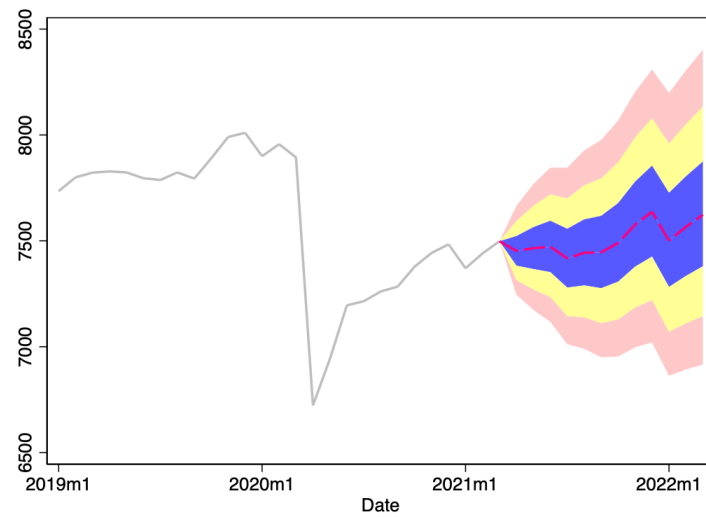
Forecasting

One month ahead



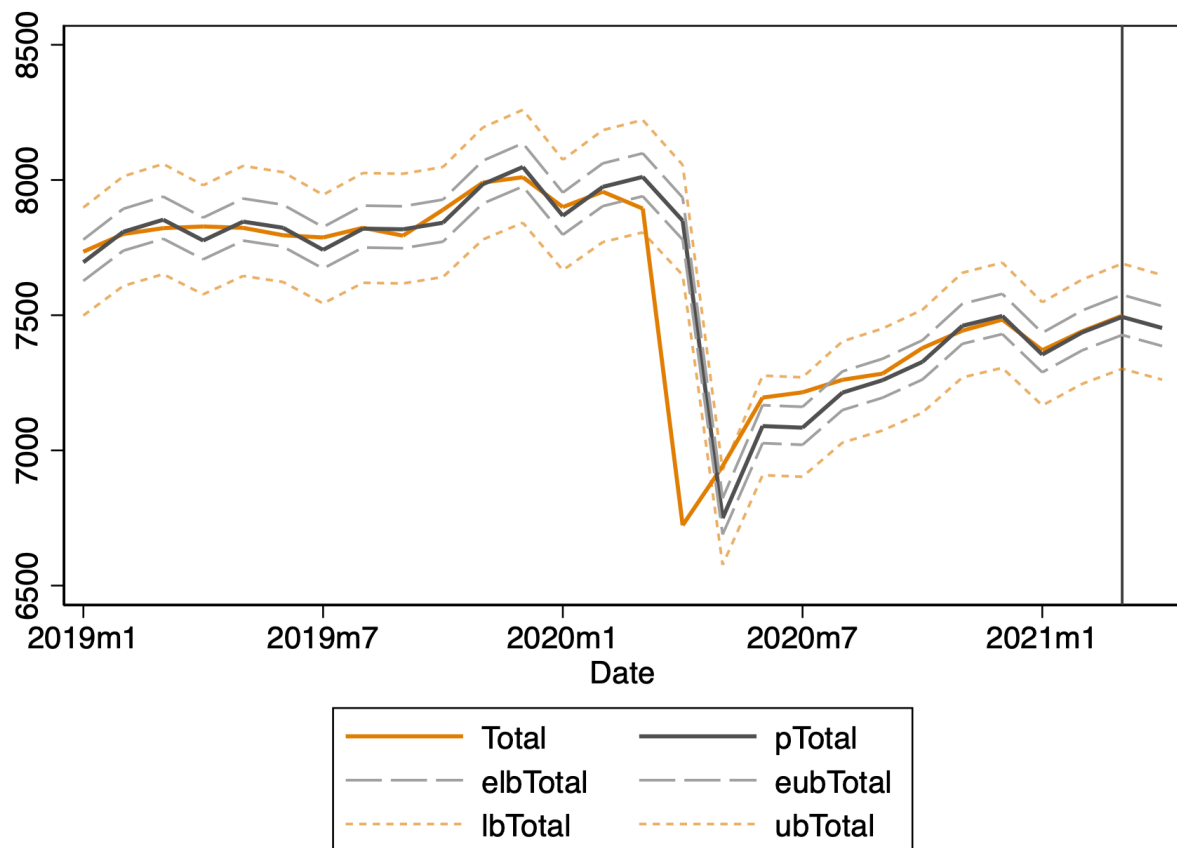
Sorry for the bad colors. Hailey peer pressured me into it.

One year out

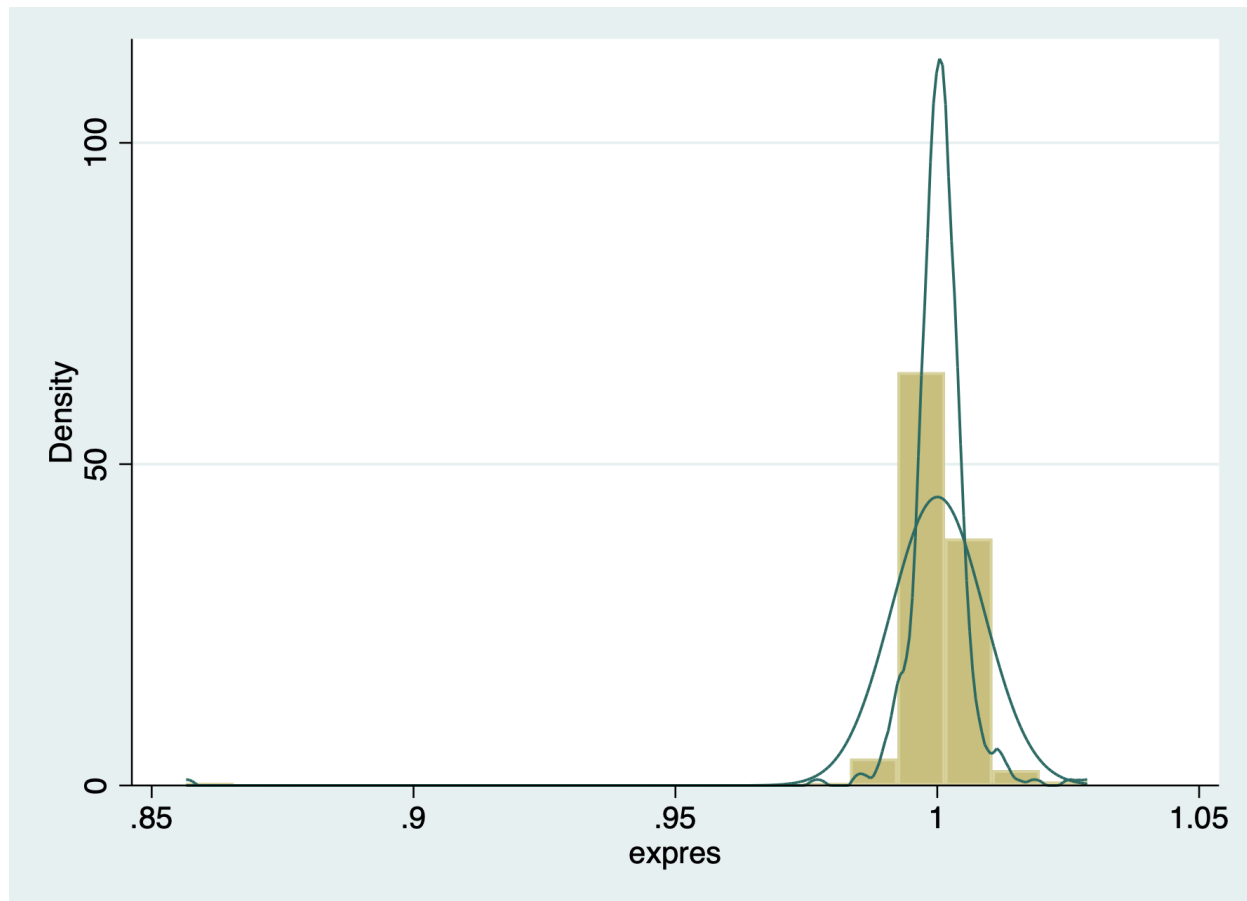


Ditto my earlier comment on the colors :)

Forecast Evaluation



The prediction is pretty good and stays well within bounds (as it should). The only real mis-step is around the pandemic because the model could not have foreseen such an event. Even so, once it did hit, the model stuck the landing and continued to forecast quite well to the present.



Residuals are good. Nice and normal.

Chebyshev

I think I have a basic understanding of Chebyshev. It's just a super general way of estimating what X portion of your population can be X distance away from the mean. In theory, this should hold even if it's not normal. That second part's key because it means we can use it on non-log transformed data.

Do File

```
1 clear
2 set more off
3
4 cd "/Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time Series/Problem
  Sets/Final Exam"
5 log using "Final Exam.smcl", replace
6 import delimited "SP21Final.csv"
```

```

7
8   gen datec=date(date, "YMD")
9   gen Date=mofd(datec)
10  gen month=month(datec)
11  format Date %tm
12  tsset Date
13
14  gen lnConstruct = ln(construct)
15  gen lnLeisure = ln(leisure)
16  gen lnManufacture = ln(manufacture)
17  gen lnTotal = ln(total)
18
19  gen Total = total
20  gen Construct = construct
21  gen Leisure = leisure
22  gen Manufacture = manufacture
23
24  /*
25  gen withMarchTotal = Total
26  replace Total=. if tin(2021m3,)
27  */
28
29  tsset Date
30  tsappend, add(12)
31  replace month=month(dofm(Date))
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
56
57 summ construct leisure manufacture total
58 summ lnConstruct lnLeisure lnManufacture lnTotal
59
60 tsline lnConstruct lnLeisure, saving(lnConstructLeisure_tsline.gph, replace)
61 tsline lnManufacture, saving(lnManufacture_tsline.gph, replace)
62 graph combine lnConstructLeisure_tsline.gph lnManufacture_tsline.gph, ///
63     saving(lnConstructLeisure, replace)
64 graph export "lnConstructLeisure-Manufacture_tsline.png", replace
65
66 tsline lnTotal, saving(lnTotal_tsline.gph, replace)
67 tsline d.lnTotal, saving(dlnTotal_tsline.gph, replace)
68 graph combine Total_tsline.gph dlnTotal_tsline.gph, saving(lnTotal-Total,
69     replace)
69 graph export "lnTotal-dlnTotal_tsline.png", replace
70
71 ac lnTotal, saving(lnTotal_ac, replace)
72 pac lnTotal, saving(lnTotal_pac, replace)
73 graph combine lnTotal_ac.gph lnTotal_pac.gph, saving(lnTotal_ac_pac, replace)
74 graph export "lnTotal_ac_pac.png", replace
75 dfuller lnTotal, trend regress
76
77 ac lnConstruct, saving(lnConstruct_ac, replace)
78 pac lnConstruct, saving(lnConstruct_pac, replace)
79 graph combine lnConstruct_ac.gph lnConstruct_pac.gph, saving(lnConstruct_ac_pac,
80     replace)
80 graph export "lnConstruct_ac_pac.png", replace
81 dfuller lnConstruct, trend regress
82
83 ac lnLeisure, saving(lnLeisure_ac, replace)
84 pac lnLeisure, saving(lnLeisure_pac, replace)
85 graph combine lnLeisure_ac.gph lnLeisure_pac.gph, saving(lnLeisure_ac_pac,
86     replace)
86 graph export "lnLeisure_ac_pac.png", replace
87 dfuller lnLeisure, trend regress
88
89 ac lnManufacture, saving(lnManufacture_ac, replace)
90 pac lnManufacture, saving(lnManufacture_pac, replace)
91 graph combine lnManufacture_ac.gph lnManufacture_pac.gph,
92     saving(lnManufacture_ac_pac, replace)
92 graph export "lnManufacture_ac_pac.png", replace
93 dfuller lnManufacture, trend regress
94
95 quietly reg l(12,24)d.Construct l(12,24)d.Leisure l(12,24)d.Manufacture
96 testparm l(12,24)d.Construct l(12,24)d.Leisure l(12,24)d.Manufacture
97

```

```

98 newey d.lnTotal l(0/3,12,24)d.Construct l(0/3,12,24)d.Manufacture
   l(0/3,12,24)d.Leisure, lag(24)
99 test ld.Construct + ld.Construct + l2d.Construct + l3d.Construct +
   l12d.Construct + l24d.Construct ///
100 == d.Manufacture + ld.Manufacture + l2d.Manufacture + l3d.Manufacture + ///
101    l12d.Manufacture + l24d.Manufacture
102 test d.Construct + ld.Construct + l2d.Construct + l3d.Construct + l12d.Construct
   + l24d.Construct ///
103 == d.Leisure + ld.Leisure + l2d.Leisure + l3d.Leisure + l12d.Leisure +
   l24d.Leisure
104 test d.Leisure + ld.Leisure + l2d.Leisure + l3d.Leisure + l12d.Leisure +
   l24d.Leisure ///
105 == d.Manufacture + ld.Manufacture + l2d.Manufacture + l3d.Manufacture + ///
106    l12d.Manufacture + l24d.Manufacture
107
108 *-----*
109 gen dlnConstruct=d.lnConstruct
110 gen l1dlnConstruct=l1d.lnConstruct
111 gen l2dlnConstruct=l2d.lnConstruct
112 gen l3dlnConstruct=l3d.lnConstruct
113 gen l12dlnConstruct=l12d.lnConstruct
114 gen l24dlnConstruct=l24d.lnConstruct
115
116 gen dlnLeisure=d.lnLeisure
117 gen l1dlnLeisure=l1d.lnLeisure
118 gen l2dlnLeisure=l2d.lnLeisure
119 gen l3dlnLeisure=l3d.lnLeisure
120 gen l12dlnLeisure=l12d.lnLeisure
121 gen l24dlnLeisure=l24d.lnLeisure
122
123 gen dlnManufacture=d.lnManufacture
124 gen l1dlnManufacture=l1d.lnManufacture
125 gen l2dlnManufacture=l2d.lnManufacture
126 gen l3dlnManufacture=l3d.lnManufacture
127 gen l12dlnManufacture=l12d.lnManufacture
128 gen l24dlnManufacture=l24d.lnManufacture
129
130 gen dlnTotal=d.lnTotal
131 gen l1dlnTotal=l1d.lnTotal
132 gen l2dlnTotal=l2d.lnTotal
133 gen l3dlnTotal=l3d.lnTotal
134 gen l12dlnTotal=l12d.lnTotal
135 gen l24dlnTotal=l24d.lnTotal
136
137 /*
138 gsreg dlnTotal dlnConstruct l1dlnConstruct l2dlnConstruct l3dlnConstruct ///
139    l12dlnConstruct l24dlnConstruct ///

```

```

140     dlnLeisure l1dlnLeisure l2dlnLeisure l3dlnLeisure l12dlnLeisure l24dlnLeisure
141     ///
142     dlnManufacture l1dlnManufacture l2dlnManufacture l3dlnManufacture ///
143     l12dlnManufacture l24dlnManufacture ///
144     if tin(1990m1,2021m3), ///
145     ncomb(1,6) aic outsample(24) fix(m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11) ///
146     samesample nindex( -1 aic -1 bic -1 rmse_out) results(gsreg_dlnTtoal) replace
147     */
148
149     loocv reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
150     quietly reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
151     estat ic
152
153     loocv reg d.lnTotal l(1/3,12,24)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
154     quietly reg d.lnTotal l(1/3,12,24)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
155     estat ic
156
157     loocv reg d.lnTotal l(1/3)d.lnTotal l(1/3)d.lnConstruct l(1/3)d.lnLeisure ///
158     l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
159     quietly reg d.lnTotal l(1/3)d.lnTotal l(1/3)d.lnConstruct l(1/3)d.lnLeisure ///
160     l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
161     estat ic
162
163     loocv reg d.lnTotal l(1/3,12,24)d.lnTotal l(1/3,12,24)d.lnConstruct ///
164     l(1/3,12,24)d.lnLeisure l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
165     m11
166     quietly reg d.lnTotal l(1/3,12,24)d.lnTotal l(1/3,12,24)d.lnConstruct ///
167     l(1/3,12,24)d.lnLeisure l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
168     m11
169     estat ic
170
171     *Lowest rmse (1)
172     reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
173     scalar drop _all
174     quietly forval w=12(12)180 {
175     gen pred=.
176     gen nobs=.
177     forval t=544/734 {
178     gen wstart=`t'-'w'
179     gen wend=`t'-1
180     reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 ///
181     if Date>=wstart & Date<=wend
182     replace nobs=e(N) if Date==`t'
183     predict ptemp
184     replace pred=ptemp if Date==`t'
185     drop ptemp wstart wend
186     }

```

```

185 gen errsq=(pred-d.lnTotal)^2
186 summ errsq
187 scalar RWrmse`w'=r(mean)^.5
188 summ nobs
189 scalar RWminobs`w'=r(min)
190 scalar RWmaxobs`w'=r(max)
191 drop errsq pred nobs
192 }
193 scalar list
194
195 /*
196 RWmaxobs12 =          12
197 RWminobs12 =          12
198 RWrmse12 =    .0132376
199 */
200
201 *lowest AIC and BIC (3)
202 reg d.lnTotal l(1/3)d.lnTotal l(1/3)d.lnConstruct l(1/3)d.lnLeisure ///
203   l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
204 scalar drop _all
205 quietly forval w=3(3)180 {
206   gen pred=.
207   gen nobs=.
208   forval t=544/734 {
209     gen wstart=`t'-'w'
210     gen wend=`t'-1
211     reg d.lnTotal l(1/3)d.lnTotal l(1/3)d.lnConstruct l(1/3)d.lnLeisure ///
212       l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 ///
213     if Date>=wstart & Date<=wend
214     replace nobs=e(N) if Date==`t'
215     predict ptemp
216     replace pred=ptemp if Date==`t'
217     drop ptemp wstart wend
218   }
219   gen errsq=(pred-d.lnTotal)^2
220   summ errsq
221   scalar RWrmse`w'=r(mean)^.5
222   summ nobs
223   scalar RWminobs`w'=r(min)
224   scalar RWmaxobs`w'=r(max)
225   drop errsq pred nobs
226 }
227 scalar list
228 /*
229 RWmaxobs12 =          12
230 RWminobs12 =          12
231 RWrmse12 =    .0132376
232 */

```



```

233
234
235 * Going with model 1 because average RWrmse is lower across window sizes
236 scalar rwrms = .0132376
237 reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 if tin(,2021m3)
238 predict pd
239 gen pflcount=exp((rwrms^2)/2)*exp(1.lnTotal+pd) if Date==tm(2021m4)
240 gen ub1=exp((rwrms^2)/2)*exp(1.lnTotal+pd+1*rwrms) if Date==tm(2021m4)
241 gen lb1=exp((rwrms^2)/2)*exp(1.lnTotal+pd-1*rwrms) if Date==tm(2021m4)
242 gen ub2=exp((rwrms^2)/2)*exp(1.lnTotal+pd+2*rwrms) if Date==tm(2021m4)
243 gen lb2=exp((rwrms^2)/2)*exp(1.lnTotal+pd-2*rwrms) if Date==tm(2021m4)
244 gen ub3=exp((rwrms^2)/2)*exp(1.lnTotal+pd+3*rwrms) if Date==tm(2021m4)
245 gen lb3=exp((rwrms^2)/2)*exp(1.lnTotal+pd-3*rwrms) if Date==tm(2021m4)
246 drop pd
247
248 replace pflcount=Total if Date==tm(2021m3)
249 replace ub1=Total if Date==tm(2021m3)
250 replace ub2=Total if Date==tm(2021m3)
251 replace ub3=Total if Date==tm(2021m3)
252 replace lb1=Total if Date==tm(2021m3)
253 replace lb2=Total if Date==tm(2021m3)
254 replace lb3=Total if Date==tm(2021m3)
255
256 twoway (tsrline ub3 ub2 if tin(2020m4,2021m4), ///
257       recast(rarea) fcolor(orange) fintensity(20) lwidth(none) ) ///
258       (tsrline ub2 ub1 if tin(2020m4,2021m4), ///
259       recast(rarea) fcolor(green) fintensity(40) lwidth(none) ) ///
260       (tsrline ub1 pflcount if tin(2020m4,2021m4), ///
261       recast(rarea) fcolor(purple) fintensity(65) lwidth(none) ) ///
262       (tsrline pflcount lb1 if tin(2020m4,2021m4), ///
263       recast(rarea) fcolor(purple) fintensity(65) lwidth(none) ) ///
264       (tsrline lb1 lb2 if tin(2020m4,2021m4), ///
265       recast(rarea) fcolor(green) fintensity(40) lwidth(none) ) ///
266       (tsrline lb2 lb3 if tin(2020m4,2021m4), ///
267       recast(rarea) fcolor(orange) fintensity(20) lwidth(none) ) ///
268       (tsline Total pflcount if tin(2020m4,2021m4) , ///
269       lcolor(gs12 pink) lwidth(medthick medthick) ///
270       lpattern(solid longdash)), scheme(slmono) legend(off)
271 graph export "TotalFan1.png", replace
272
273 * More than 1 step
274 arima d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 if
tin(1990m1,2021m3)
275 predict pnonfarm, dynamic(tm(2021m3))
276 predict mse, mse dynamic(mofd(tm(2021m4)))
277 gen totmse = mse if Date==tm(2021m4)
278 replace totmse = 1.totmse+mse if Date>tm(2021m4)
279 gen pnonfarma = Total if Date==tm(2021m3)

```

```

280 replace pnonfarma = l.pnonfarma*exp(pnonfarm+mse/2) if Date>tm(2021m3)
281
282 gen ub1a = pnonfarma*exp(totmse^.5)
283 gen ub2a = pnonfarma*exp(2*totmse^.5)
284 gen ub3a = pnonfarma*exp(3*totmse^.5)
285 gen lb1a = pnonfarma/exp(totmse^.5)
286 gen lb2a = pnonfarma/exp(2*totmse^.5)
287 gen lb3a = pnonfarma/exp(3*totmse^.5)
288
289 replace ub1a=Total if Date == tm(2021m3)
290 replace ub2a=Total if Date == tm(2021m3)
291 replace ub3a=Total if Date == tm(2021m3)
292 replace lb1a=Total if Date == tm(2021m3)
293 replace lb2a=Total if Date == tm(2021m3)
294 replace lb3a=Total if Date == tm(2021m3)
295
296 twoway (tsrline ub3a ub2a if tin(2019m1,2022m3), ///
297       recast(rarea) fcolor(red) fintensity(20) lwidth(none) ) ///
298       (tsrline ub2a ub1a if tin(2019m1,2022m3), ///
299       recast(rarea) fcolor(yellow) fintensity(40) lwidth(none) ) ///
300       (tsrline ub1a pnonfarma if tin(2019m1,2022m3), ///
301       recast(rarea) fcolor(blue) fintensity(65) lwidth(none) ) ///
302       (tsrline pnonfarma lb1a if tin(2019m1,2022m3), ///
303       recast(rarea) fcolor(blue) fintensity(65) lwidth(none) ) ///
304       (tsrline lb1a lb2a if tin(2019m1,2022m3), ///
305       recast(rarea) fcolor(yellow) fintensity(40) lwidth(none) ) ///
306       (tsrline lb2a lb3a if tin(2019m1,2022m3), ///
307       recast(rarea) fcolor(red) fintensity(20) lwidth(none) ) ///
308       (tsline Total pnonfarma if tin(2019m1,2022m3) , ///
309       lcolor(gs12 pink) lwidth(medthick medthick) ///
310       lpattern(solid longdash)) , scheme(slmono) legend(off)
311 graph export "TotalFan12.png", replace
312
313 scalar rmse_mod1 = .0132376
314 reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 if
tin(1990m1,2021m3)
315 predict plTotal
316 predict temp if tin(2021m3,2021m3)
317 replace plTotal=temp if tin(2021m3,2021m3)
318 drop temp
319 gen pTotal=exp(l.lnTotal+plTotal+(rmse_mod1^2)/2)
320 gen lbTotal=exp(l.lnTotal+plTotal-1.96*rmse_mod1+(rmse_mod1^2)/2)
321 gen ubTotal=exp(l.lnTotal+plTotal+1.96*rmse_mod1+(rmse_mod1^2)/2)
322
323 gen res=(d.lnTotal-plTotal)
324 gen expres=exp(res)
325 summ expres
326 scalar meanexpres=r(mean)

```

```

327 gen epTotal=exp(1.lnTotal+plTotal)*meanexpres
328 _pctile res, percentile(2.5,97.5)
329 return list
330 gen elbTotal=exp(1.lnTotal+plTotal+r(r1))*meanexpres
331 gen eubTotal=exp(1.lnTotal+plTotal+r(r2))*meanexpres
332
333 tsline Total pTotal elbTotal eubTotal lbTotal ubTotal if tin(2019m1,2021m4), ///
334     scheme(slmono) tline(2021m3, lcolor(gs4)) ///
335     lpattern(solid solid longdash longdash shortdash shortdash) ///
336     lcolor(dkorange gs5 gs10 gs10 dkorange%60 dkorange%60) ///
337     lwidth(medthick medthick medium medium)
338 graph export "interval_tsline.png", replace
339
340 histogram expres, normal kdensity saving(residuals.gph, replace)
341 graph export "residuals.png", replace
342
343 log close
344 translate "Final Exam.smcl" "Final Project.txt", replace

```

Log File

```

1  _____
2  ____ (R)
3  ____ /
4  / ____ /
5
6  -----
7  name: <unnamed>
8  log: /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
9  Series/Probl
10 > em Sets/Final Exam/Final Exam.smcl
11 log type: smcl
12 opened on: 29 Apr 2021, 12:07:09
13
14 1 . import delimited "SP21Final.csv"
15 (5 vars, 375 obs)
16
17 2 .
18
19 3 . gen datec=date(date, "YMD")
20
21 4 . gen Date=mofd(datec)

```

```

20
21     5 . gen month=month(datec)
22
23     6 . format Date %tm
24
25     7 . tsset Date
26           time variable:  Date, 1990m1 to 2021m3
27           delta: 1 month
28
29     8 .
30     9 . gen lnConstruct = ln(construct)
31
32    10 . gen lnLeisure = ln(leisure)
33
34    11 . gen lnManufacture = ln(manufacture)
35
36    12 . gen lnTotal = ln(total)
37
38    13 .
39    14 . gen Total = total
40
41    15 . gen Construct = construct
42
43    16 . gen Leisure = leisure
44
45    17 . gen Manufacture = manufacture
46
47    18 .
48    19 . /*
49        > gen withMarchTotal = Total
50        > replace Total=. if tin(2021m3,)
51        > */
52    20 .
53    21 . tsset Date
54           time variable:  Date, 1990m1 to 2021m3
55           delta: 1 month
56
57    22 . tsappend, add(12)
58
59    23 . replace month=month(dofm(Date))
60        (12 real changes made)
61
62    24 .
63    25 . gen m1=0
64
65    26 . replace m1=1 if month==1
66        (33 real changes made)
67

```

```
68      27 . gen m2=0
69
70      28 . replace m2=1 if month==2
71          (33 real changes made)
72
73      29 . gen m3=0
74
75      30 . replace m3=1 if month==3
76          (33 real changes made)
77
78      31 . gen m4=0
79
80      32 . replace m4=1 if month==4
81          (32 real changes made)
82
83      33 . gen m5=0
84
85      34 . replace m5=1 if month==5
86          (32 real changes made)
87
88      35 . gen m6=0
89
90      36 . replace m6=1 if month==6
91          (32 real changes made)
92
93      37 . gen m7=0
94
95      38 . replace m7=1 if month==7
96          (32 real changes made)
97
98      39 . gen m8=0
99
100     40 . replace m8=1 if month==8
101         (32 real changes made)
102
103     41 . gen m9=0
104
105     42 . replace m9=1 if month==9
106         (32 real changes made)
107
108     43 . gen m10=0
109
110     44 . replace m10=1 if month==10
111         (32 real changes made)
112
113     45 . gen m11=0
114
115     46 . replace m11=1 if month==11
```

```

116      (32 real changes made)
117
118      47 .
119      48 .
120      49 . summ construct leisure manufacture total
121
122          Variable |           Obs           Mean      Std. Dev.           Min           Max
123      -----+-----
124          construct |           375          461.0043          95.8947           323.9           696.1
125          leisure   |           375          930.2083         159.6216           660.6          1287.5
126          manufacture |           375          410.0496          63.14375           307.9           518.2
127          total     |           375         6161.164          958.2068          4366.1          8010.4
128
129      50 . summ lnConstruct lnLeisure lnManufacture lnTotal
130
131          Variable |           Obs           Mean      Std. Dev.           Min           Max
132      -----+-----
133          lnConstruct |           375          6.112377          .2043883          5.780435          6.545493
134          lnLeisure  |           375          6.820994          .1695282          6.493148          7.160458
135          lnManufact~e |           375          6.004071          .1577555          5.729775          6.250361
136          lnTotal    |           375          8.71332          .1619029          8.381625          8.988496
137
138      51 .
139      52 . tsline lnConstruct lnLeisure, saving(lnConstructLeisure_tsline.gph,
replace)
140          (file lnConstructLeisure_tsline.gph saved)
141
142      53 . tsline lnManufacture, saving(lnManufacture_tsline.gph, replace)
143          (file lnManufacture_tsline.gph saved)
144
145      54 . graph combine lnConstructLeisure_tsline.gph lnManufacture_tsline.gph,
///
146          >          saving(lnConstructLeisure, replace)
147          (file lnConstructLeisure.gph saved)
148
149      55 . graph export "lnConstructLeisure-Manufacture_tsline.png", replace
150          (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Problem Sets
151          > /Final Exam/lnConstructLeisure-Manufacture_tsline.png written in PNG
format)
152
153      56 .
154      57 . tsline lnTotal, saving(lnTotal_tsline.gph, replace)
155          (file lnTotal_tsline.gph saved)
156
157      58 . tsline d.lnTotal, saving(dlnTotal_tsline.gph, replace)
158          (file dlnTotal_tsline.gph saved)
159

```

```

160     59 . graph combine Total_tsline.gph dlnTotal_tsline.gph, saving(lnTotal-
Total, rep
161     > lace)
162     (file lnTotal-Total.gph saved)
163
164     60 . graph export "lnTotal-dlnTotal_tsline.png", replace
165     (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Problem Sets
166     > /Final Exam/lnTotal-dlnTotal_tsline.png written in PNG format)
167
168     61 .
169     62 . ac lnTotal, saving(lnTotal_ac, replace)
170     (file lnTotal_ac.gph saved)
171
172     63 . pac lnTotal, saving(lnTotal_pac, replace)
173     (file lnTotal_pac.gph saved)
174
175     64 . graph combine lnTotal_ac.gph lnTotal_pac.gph, saving(lnTotal_ac_pac,
replace)
176     (file lnTotal_ac_pac.gph saved)
177
178     65 . graph export "lnTotal_ac_pac.png", replace
179     (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Problem Sets
180     > /Final Exam/lnTotal_ac_pac.png written in PNG format)
181
182     66 . dfuller lnTotal, trend regress
183
184     Dickey-Fuller test for unit root                                Number of obs   =
374
185
186                                     ----- Interpolated Dickey-Fuller -----
187
-----
187             Test             1% Critical       5% Critical       10%
Critical
188             Statistic         Value             Value
Value
189     -----
190     Z(t)          -1.895          -3.985          -3.425
-3.130
191     -----
192     MacKinnon approximate p-value for Z(t) = 0.6570
193
194     -----
-----

```

```

195      D.lnTotal |      Coef.   Std. Err.      t    P>|t|      [95% Conf.
196      Interval]
197      -----+-----
198      lnTotal |
199      L1. |   -.0191916   .0101251   -1.90   0.059   -.0391013
200      .0007181
201      |
202      _trend |   .0000223   .0000151    1.47   0.142   -7.48e-06
203      .0000521
204      _cons |   .1644033   .085612    1.92   0.056   -.0039423
205      .3327489
206      -----+-----
207
208      67 .
209      68 . ac lnConstruct, saving(lnConstruct_ac, replace)
210      (file lnConstruct_ac.gph saved)
211
212      69 . pac lnConstruct, saving(lnConstruct_pac, replace)
213      (file lnConstruct_pac.gph saved)
214
215      70 . graph combine lnConstruct_ac.gph lnConstruct_pac.gph,
216      saving(lnConstruct_ac_p
217      > ac, replace)
218      (file lnConstruct_ac_pac.gph saved)
219
220      71 . graph export "lnConstruct_ac_pac.png", replace
221      (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
222      Series/Problem Sets
223      > /Final Exam/lnConstruct_ac_pac.png written in PNG format)
224
225      72 . dfuller lnConstruct, trend regress
226
227      Dickey-Fuller test for unit root      Number of obs   =
228      374
229
230      ----- Interpolated Dickey-Fuller -----
231
232      Critical      Test      1% Critical      5% Critical      10%
233      Value      Statistic      Value      Value
234
235      -----+-----
236
237      Z(t)      -0.586      -3.985      -3.425
238      -3.130

```



```

228 -----
229 -----
229 MacKinnon approximate p-value for Z(t) = 0.9796
230
231 -----
232 -----
232 D.          |
233 lnConstruct |      Coef.   Std. Err.      t    P>|t|      [95% Conf.
Interval]
234 -----+-----
235 lnConstruct |
236      L1. |   -.0021673   .0036962    -0.59   0.558   -.0094355
.0051009
237          |
238      _trend |    5.05e-06   6.99e-06     0.72   0.470   -8.68e-06
.0000188
239      _cons |    .0132628   .0221901     0.60   0.550   -.0303713
.0568969
240 -----
241 -----
242 73 .
243 74 . ac lnLeisure, saving(lnLeisure_ac, replace)
244     (file lnLeisure_ac.gph saved)
245
246 75 . pac lnLeisure, saving(lnLeisure_pac, replace)
247     (file lnLeisure_pac.gph saved)
248
249 76 . graph combine lnLeisure_ac.gph lnLeisure_pac.gph,
saving(lnLeisure_ac_pac, re
250     > place)
251     (file lnLeisure_ac_pac.gph saved)
252
253 77 . graph export "lnLeisure_ac_pac.png", replace
254     (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Problem Sets
255     > /Final Exam/lnLeisure_ac_pac.png written in PNG format)
256
257 78 . dfuller lnLeisure, trend regress
258
259 Dickey-Fuller test for unit root                      Number of obs   =
374
260
261 ----- Interpolated Dickey-Fuller -----
262 -----
Test          1% Critical      5% Critical      10%
Critical

```

263	Value	Statistic	Value	Value
264	-----			
265	-----			
266	-----			
267	-----			
268	-----			
269	-----			
270	-----			
271	-----			
272	-----			
273	-----			
274	-----			
275	-----			
276	-----			
277	-----			
278	-----			
279	-----			
280	-----			
281	-----			
282	-----			
283	-----			
284	-----			
285	-----			
286	-----			
287	-----			
288	-----			
289	-----			
290	-----			
291	-----			
292	-----			
293	-----			
294	-----			
295	-----			
296	-----			
374	-----			

```

297
298 ----- Interpolated Dickey-Fuller -----
299 -----
300 Test          1% Critical      5% Critical      10%
Critical
301 Statistic      Value          Value
Value
302 -----
303 Z(t)           0.313          -3.985          -3.425
-3.130
304 -----
305 MacKinnon approximate p-value for Z(t) = 0.9963
306 -----
307 -----
308 D.             |
lnManufacture |      Coef.   Std. Err.    t    P>|t|    [95% Conf.
Interval]
309 -----+-----
310 lnManufacture |
311 L1. |      .0013361   .0042655    0.31   0.754    -.0070515
.0097238
312 |
313 _trend |      9.58e-06   6.23e-06    1.54   0.125    -2.67e-06
.0000218
314 _cons |     -.0106361   .0265921   -0.40   0.689    -.0629262
.0416539
315 -----
316 -----
317 85 .
318 86 . quietly reg l(12,24)d.Construct l(12,24)d.Leisure
l(12,24)d.Manufacture
319
320 87 . testparm l(12,24)d.Construct l(12,24)d.Leisure l(12,24)d.Manufacture
321
322 ( 1) L24D.Construct = 0
323 ( 2) L12D.Leisure = 0
324 ( 3) L24D.Leisure = 0
325 ( 4) L12D.Manufacture = 0
326 ( 5) L24D.Manufacture = 0
327
328 F( 5, 356) = 146.28
329 Prob > F = 0.0000
330

```

```

331      88 .
332      89 . newey d.lnTotal l(0/3,12,24)d.Construct l(0/3,12,24)d.Manufacture
333      1(0/3,12,24
334      > )d.Leisure, lag(24)

```

```

335      Regression with Newey-West standard errors      Number of obs      =
336      350
337      maximum lag: 24      F( 18,      331) =
338      4461.04
339      Prob > F      =
340      0.0000

```

```

340      -----
341      |                               Newey-West
342      |      D.lnTotal |      Coef.   Std. Err.   t   P>|t|   [95% Conf.
343      |      Interval]
344      |-----+-----|

```

```

343      Construct |
344      D1. |      .0004252   .0000809   5.26   0.000   .0002661
345      .0005842
346      LD. |      .0000247   .0000044   0.56   0.574   -.0000617
347      .0001112
348      L2D. |      .0000613   .0000403   1.52   0.129   -.0000179
349      .0001405
350      L3D. |      -.0000674   .0000577   -1.17   0.244   -.000181
351      .0000462
352      L12D. |      -.000035   .0000641   -0.55   0.585   -.000161
353      .000091
354      L24D. |      .0000494   .0000684   0.72   0.470   -.0000851
355      .0001839

```

```

356      Manufacture |
357      D1. |      .0011938   .0001924   6.21   0.000   .0008154
358      .0015723
359      LD. |      -.0002374   .0001386   -1.71   0.088   -.00051
360      .0000353
361      L2D. |      -.000336   .0000718   -4.68   0.000   -.0004773
362      -.0001948
363      L3D. |      -.0002689   .0001059   -2.54   0.012   -.0004772
364      -.0000606
365      L12D. |      .0003308   .0001381   2.39   0.017   .000059
366      .0006025
367      L24D. |      .0002403   .0001413   1.70   0.090   -.0000376
368      .0005182

```

```

369      Leisure |

```

```

360          D1. | .0002134 8.38e-06 25.47 0.000 .0001969
      .0002299
361          LD. | 5.34e-06 9.19e-06 0.58 0.561 -.0000127
      .0000234
362          L2D. | -2.42e-06 .0000172 -0.14 0.888 -.0000364
      .0000315
363          L3D. | -8.76e-06 .0000174 -0.50 0.614 -.0000429
      .0000254
364          L12D. | .0000629 .0000361 1.74 0.082 -8.04e-06
      .0001339
365          L24D. | -.000016 .0000428 -0.37 0.708 -.0001002
      .0000681
366          |
367          _cons | .0011957 .0003664 3.26 0.001 .000475
      .0019165
368          -----
      -----
369
370          90 . test ld.Construct + ld.Construct + l2d.Construct + l3d.Construct +
      l12d.Const
371          > ruct + l24d.Construct ///
372          > == d.Manufacture + ld.Manufacture + l2d.Manufacture +
      l3d.Manufacture
373          > + ///
374          > l12d.Manufacture + l24d.Manufacture
375
376          ( 1) 2*LD.Construct + L2D.Construct + L3D.Construct + L12D.Construct +
377          L24D.Construct - D.Manufacture - LD.Manufacture - L2D.Manufacture
      -
378          L3D.Manufacture - L12D.Manufacture - L24D.Manufacture = 0
379
380          F( 1, 331) = 5.85
381          Prob > F = 0.0161
382
383          91 . test d.Construct + ld.Construct + l2d.Construct + l3d.Construct +
      l12d.Constr
384          > uct + l24d.Construct ///
385          > == d.Leisure + ld.Leisure + l2d.Leisure + l3d.Leisure +
      l12d.Leisure
386          > + l24d.Leisure
387
388          ( 1) D.Construct + LD.Construct + L2D.Construct + L3D.Construct +
389          L12D.Construct + L24D.Construct - D.Leisure - LD.Leisure -
      L2D.Leisure -
390          L3D.Leisure - L12D.Leisure - L24D.Leisure = 0
391
392          F( 1, 331) = 11.36
393          Prob > F = 0.0008

```

```

394
395     92 . test d.Leisure + ld.Leisure + l2d.Leisure + l3d.Leisure + l12d.Leisure
+ l24d
396     > .Leisure ///
397     >         == d.Manufacture + ld.Manufacture + l2d.Manufacture +
l3d.Manufacture
398     > + ///
399     >         l12d.Manufacture + l24d.Manufacture
400
401     ( 1) - D.Manufacture - LD.Manufacture - L2D.Manufacture -
L3D.Manufacture -
402     L12D.Manufacture - L24D.Manufacture + D.Leisure + LD.Leisure +
403     L2D.Leisure + L3D.Leisure + L12D.Leisure + L24D.Leisure = 0
404
405     F( 1, 331) = 4.74
406     Prob > F = 0.0301
407
408     93 .
409     94 . *-----
-----
410     > --*
411     95 . gen dlnConstruct=d.lnConstruct
412     (13 missing values generated)
413
414     96 . gen l1dlnConstruct=l1d.lnConstruct
415     (13 missing values generated)
416
417     97 . gen l2dlnConstruct=l2d.lnConstruct
418     (13 missing values generated)
419
420     98 . gen l3dlnConstruct=l3d.lnConstruct
421     (13 missing values generated)
422
423     99 . gen l12dlnConstruct=l12d.lnConstruct
424     (13 missing values generated)
425
426     100 . gen l24dlnConstruct=l24d.lnConstruct
427     (25 missing values generated)
428
429     101 .
430     102 . gen dlnLeisure=d.lnLeisure
431     (13 missing values generated)
432
433     103 . gen l1dlnLeisure=l1d.lnLeisure
434     (13 missing values generated)
435
436     104 . gen l2dlnLeisure=l2d.lnLeisure
437     (13 missing values generated)

```

```
438
439 105 . gen l3dlnLeisure=l3d.lnLeisure
440     (13 missing values generated)
441
442 106 . gen l12dlnLeisure=l12d.lnLeisure
443     (13 missing values generated)
444
445 107 . gen l24dlnLeisure=l24d.lnLeisure
446     (25 missing values generated)
447
448 108 .
449 109 . gen dlnManufacture=d.lnManufacture
450     (13 missing values generated)
451
452 110 . gen l1dlnManufacture=l1d.lnManufacture
453     (13 missing values generated)
454
455 111 . gen l2dlnManufacture=l2d.lnManufacture
456     (13 missing values generated)
457
458 112 . gen l3dlnManufacture=l3d.lnManufacture
459     (13 missing values generated)
460
461 113 . gen l12dlnManufacture=l12d.lnManufacture
462     (13 missing values generated)
463
464 114 . gen l24dlnManufacture=l24d.lnManufacture
465     (25 missing values generated)
466
467 115 .
468 116 . gen dlnTotal=d.lnTotal
469     (13 missing values generated)
470
471 117 . gen l1dlnTotal=l1d.lnTotal
472     (13 missing values generated)
473
474 118 . gen l2dlnTotal=l2d.lnTotal
475     (13 missing values generated)
476
477 119 . gen l3dlnTotal=l3d.lnTotal
478     (13 missing values generated)
479
480 120 . gen l12dlnTotal=l12d.lnTotal
481     (13 missing values generated)
482
483 121 . gen l24dlnTotal=l24d.lnTotal
484     (25 missing values generated)
485
```

```

486     122 .
487     123 . /*
488         > gsreg dlnTotal dlnConstruct l1dlnConstruct l2dlnConstruct
13dlnConstruct ///
489         >         l12dlnConstruct l24dlnConstruct ///
490         >         dlnLeisure l1dlnLeisure l2dlnLeisure l3dlnLeisure l12dlnLeisure
124dl
491         > nLeisure ///
492         >         dlnManufacture l1dlnManufacture l2dlnManufacture
13dlnManufacture ///
493         >         l12dlnManufacture l24dlnManufacture ///
494         >         if tin(1990m1,2021m3), ///
495         >         ncomb(1,6) aic outsample(24) fix(m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
m11)
496         > ///
497         >         samesample nindex( -1 aic -1 bic -1 rmse_out)
results(gsreg_dlnTtoal)
498         > replace
499         > */
500     124 .
501     125 .
502     126 . loocv reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
503
504
505         Leave-One-Out Cross-Validation Results
506         -----
507             Method          |      Value
508         -----+-----
509         Root Mean Squared Errors |      .0129605
510         Mean Absolute Errors    |      .00443719
511         Pseudo-R2              |      .14054754
512         -----
513
514     127 . quietly reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
m11
515
516     128 . estat ic
517
518         Akaike's information criterion and Bayesian information criterion
519
520         -----
521         ----
522             Model |          N    ll(null)   ll(model)      df          AIC
BIC
523             . |          371    1102.298    1206.362      15    -2382.725
-2323.982

```



```

524 -----
525 -----
526 Note: BIC uses N = number of observations. See [R] BIC note.
527
528 129 .
529 130 . loocv reg d.lnTotal l(1/3,12,24)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9
530 m10 m11
531
532 Leave-One-Out Cross-Validation Results
533 -----
534 Method | Value
535 -----+-----
536 Root Mean Squared Errors | .01300797
537 Mean Absolute Errors | .00427199
538 Pseudo-R2 | .14852182
539 -----
540
541 131 . quietly reg d.lnTotal l(1/3,12,24)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9
542 m10 m1
543 > 1
544
545 132 . estat ic
546
547 Akaike's information criterion and Bayesian information criterion
548 -----
549 -----
550 Model | N ll(null) ll(model) df AIC
551 BIC
552 -----+-----
553 . | 350 1035.49 1135.109 17 -2236.218
554 -2170.633
555 -----
556 -----
557 Note: BIC uses N = number of observations. See [R] BIC note.
558
559 133 .
560 134 . loocv reg d.lnTotal l(1/3)d.lnTotal l(1/3)d.lnConstruct
561 l(1/3)d.lnLeisure ///
562 > l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
563
564 Leave-One-Out Cross-Validation Results
565 -----
566 Method | Value
567 -----+-----

```

```

563      Root Mean Squared Errors |      .01611154
564      Mean Absolute Errors    |      .00423375
565      Pseudo-R2               |      .06722376
566      -----
567
568      135 . quietly reg d.lnTotal l(1/3)d.lnTotal l(1/3)d.lnConstruct
1(1/3)d.lnLeisure /
569      > //
570      >          l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
571
572      136 . estat ic
573
574      Akaike's information criterion and Bayesian information criterion
575
576      -----
577      ----
578      Model |          N    ll(null)    ll(model)      df          AIC
579      BIC
580      -----+-----
581      . |          371    1102.298    1220.504      24    -2393.008
582      -2299.019
583      -----
584      ----
585      Note: BIC uses N = number of observations. See [R] BIC note.
586
587      137 .
588      138 . loocv reg d.lnTotal l(1/3,12,24)d.lnTotal l(1/3,12,24)d.lnConstruct
589      ///
590      >          l(1/3,12,24)d.lnLeisure l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6
591      m7 m8
592      > m9 m10 m11
593
594      Leave-One-Out Cross-Validation Results
595      -----
596      Method |      Value
597      -----+-----
598      Root Mean Squared Errors |      .01713897
599      Mean Absolute Errors    |      .004436
600      Pseudo-R2               |      .05184396
601      -----
602
603      139 . quietly reg d.lnTotal l(1/3,12,24)d.lnTotal l(1/3,12,24)d.lnConstruct
604      ///
605      >          l(1/3,12,24)d.lnLeisure l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6
606      m7 m8
607      > m9 m10 m11

```

```

601
602     140 . estat ic
603
604     Akaike's information criterion and Bayesian information criterion
605
606     -----
607     ----
608     Model |          N    ll(null)    ll(model)      df          AIC
609     BIC
610     -----+-----
611     ----
612     . |          350    1035.49    1145.391      30    -2230.781
613     -2115.043
614     -----
615
616     Note: BIC uses N = number of observations. See [R] BIC note.
617
618
619     141 .
620     142 . *Lowest rmse (1)
621     143 . reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
622
623     Source |          SS          df          MS      Number of obs      =
624     371
625     -----+-----
626     19.13
627     Model |    .024491123          14    .001749366      Prob > F          =
628     0.0000
629     Residual |    .032550038         356    .000091433      R-squared          =
630     0.4294
631     -----+-----
632     0.4069
633     Total |    .057041161         370    .000154165      Root MSE          =
634     .00956
635
636     -----
637     D.lnTotal |          Coef.    Std. Err.      t    P>|t|      [95% Conf.
638     Interval]
639     -----+-----
640     -----
641     lnTotal |
642     LD. |    -.0016173    .0529806     -0.03    0.976    -.1058116
643     .1025769
644     L2D. |    -.1325018    .0525096     -2.52    0.012    -.2357698
645     -.0292338
646     L3D. |     .026173    .0529824     0.49    0.622    -.0780247
647     .1303708
648     |

```

```

632      m1 | -.0254857 .0024679 -10.33 0.000 -.0303391
      -.0206322
633      m2 | .0004869 .0029045 0.17 0.867 -.0052252
      .006199
634      m3 | -.003913 .0027686 -1.41 0.158 -.0093579
      .0015318
635      m4 | -.013317 .0026543 -5.02 0.000 -.0185371
      -.0080969
636      m5 | -.0064632 .0026182 -2.47 0.014 -.0116124
      -.0013141
637      m6 | -.0088465 .0025835 -3.42 0.001 -.0139273
      -.0037657
638      m7 | -.0159741 .0025331 -6.31 0.000 -.0209559
      -.0109924
639      m8 | -.0051433 .0026374 -1.95 0.052 -.0103302
      .0000435
640      m9 | -.0096793 .0025635 -3.78 0.000 -.0147208
      -.0046378
641      m10 | -.0021472 .0025489 -0.84 0.400 -.00716
      .0028657
642      m11 | .0024587 .0024684 1.00 0.320 -.0023958
      .0073131
643      _cons | .0088125 .0018473 4.77 0.000 .0051795
      .0124454
644      -----
      -----
645
646      144 . scalar drop _all
647
648      145 . quietly forval w=12(12)180 {
649
650      146 . scalar list
651          RWmaxobs180 = 180
652          RWminobs180 = 180
653          RWrmse180 = .01789015
654          RWmaxobs168 = 168
655          RWminobs168 = 168
656          RWrmse168 = .01812779
657          RWmaxobs156 = 156
658          RWminobs156 = 156
659          RWrmse156 = .0186461
660          RWmaxobs144 = 144
661          RWminobs144 = 144
662          RWrmse144 = .01935361
663          RWmaxobs132 = 132
664          RWminobs132 = 132
665          RWrmse132 = .02010893
666          RWmaxobs120 = 120

```

```

667      RWminobs120 =      120
668      RWrmse120 = .02273114
669      RWmaxobs108 =      108
670      RWminobs108 =      108
671      RWrmse108 = .0233275
672      RWmaxobs96 =      96
673      RWminobs96 =      96
674      RWrmse96 = .0244855
675      RWmaxobs84 =      84
676      RWminobs84 =      84
677      RWrmse84 = .02493232
678      RWmaxobs72 =      72
679      RWminobs72 =      72
680      RWrmse72 = .02471091
681      RWmaxobs60 =      60
682      RWminobs60 =      60
683      RWrmse60 = .02489376
684      RWmaxobs48 =      48
685      RWminobs48 =      48
686      RWrmse48 = .02475145
687      RWmaxobs36 =      36
688      RWminobs36 =      36
689      RWrmse36 = .02791922
690      RWmaxobs24 =      24
691      RWminobs24 =      24
692      RWrmse24 = .16402211
693      RWmaxobs12 =      12
694      RWminobs12 =      12
695      RWrmse12 = .0132376
696
697      147 .
698      148 . /*
699          > RWmaxobs12 =      12
700          > RWminobs12 =      12
701          > RWrmse12 = .0132376
702          > */
703      149 .
704      150 . *lowest AIC and BIC (3)
705      151 . reg d.lnTotal l(1/3)d.lnTotal l(1/3)d.lnConstruct l(1/3)d.lnLeisure
706      ///
707          >      l(1/3)d.lnManufacture m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11
708
709      371
710      13.45
711      0.0000

```

Source	SS	df	MS	Number of obs	=
Model	.026880308	23	.001168709	Prob > F	=
					0.0000

711	Residual		.030160853	347	.000086919	R-squared	=
	0.4712						
712	-----+	-----				Adj R-squared	=
	0.4362						
713	Total		.057041161	370	.000154165	Root MSE	=
	.00932						
714							
715	-----						

716	D.lnTotal		Coef.	Std. Err.	t	P> t	[95% Conf.
	Interval]						
717	-----+	-----					

718	lnTotal						
719	LD.		.2862672	.2667427	1.07	0.284	-.2383687
	.8109032						
720	L2D.		.2454317	.2661885	0.92	0.357	-.2781141
	.7689776						
721	L3D.		.4266043	.2702181	1.58	0.115	-.1048673
	.9580758						
722							
723	lnConstruct						
724	LD.		6.41e-06	.081453	0.00	1.000	-.1601974
	.1602102						
725	L2D.		.0327201	.0818081	0.40	0.689	-.128182
	.1936223						
726	L3D.		-.0371503	.0817044	-0.45	0.650	-.1978486
	.123548						
727							
728	lnLeisure						
729	LD.		-.1208348	.0688988	-1.75	0.080	-.2563467
	.0146771						
730	L2D.		-.1169829	.0684363	-1.71	0.088	-.251585
	.0176192						
731	L3D.		-.1609605	.069157	-2.33	0.021	-.2969801
	-.024941						
732							
733	lnManufacture						
734	LD.		.0485695	.1821043	0.27	0.790	-.3095975
	.4067365						
735	L2D.		-.1583242	.1862613	-0.85	0.396	-.5246673
	.208019						
736	L3D.		.1372412	.1883006	0.73	0.467	-.2331129
	.5075952						
737							
738	m1		-.0259546	.0026844	-9.67	0.000	-.0312344
	-.0206748						

739		m2	.0061874	.0054433	1.14	0.256	-.0045187
	.0168935						
740		m3	.0041315	.0052298	0.79	0.430	-.0061545
	.0144176						
741		m4	-.0019669	.0049557	-0.40	0.692	-.011714
	.0077802						
742		m5	-.0024728	.0032223	-0.77	0.443	-.0088106
	.0038649						
743		m6	-.0057242	.0032307	-1.77	0.077	-.0120785
	.00063						
744		m7	-.0128758	.0035363	-3.64	0.000	-.0198311
	-.0059206						
745		m8	-.0017029	.003742	-0.46	0.649	-.0090627
	.005657						
746		m9	-.006856	.0032793	-2.09	0.037	-.0133058
	-.0004062						
747		m10	.0015872	.0030906	0.51	0.608	-.0044915
	.007666						
748		m11	.0030825	.0026162	1.18	0.240	-.0020632
	.0082282						
749		_cons	.0039798	.0029489	1.35	0.178	-.0018201
	.0097797						
750	-----						

751							
752	152 . scalar drop _all						
753							
754	153 . quietly forval w=3(3)180 {						
755							
756	154 . scalar list						
757	RWmaxobs180 =	180					
758	RWminobs180 =	180					
759	RWrmse180 =	.06231312					
760	RWmaxobs177 =	177					
761	RWminobs177 =	177					
762	RWrmse177 =	.06310936					
763	RWmaxobs174 =	174					
764	RWminobs174 =	174					
765	RWrmse174 =	.06557282					
766	RWmaxobs171 =	171					
767	RWminobs171 =	171					
768	RWrmse171 =	.06644967					
769	RWmaxobs168 =	168					
770	RWminobs168 =	168					
771	RWrmse168 =	.0679586					
772	RWmaxobs165 =	165					
773	RWminobs165 =	165					
774	RWrmse165 =	.0691683					

775	RWmaxobs162 =	162
776	RWminobs162 =	162
777	RWrmse162 =	.06977207
778	RWmaxobs159 =	159
779	RWminobs159 =	159
780	RWrmse159 =	.07051975
781	RWmaxobs156 =	156
782	RWminobs156 =	156
783	RWrmse156 =	.0708996
784	RWmaxobs153 =	153
785	RWminobs153 =	153
786	RWrmse153 =	.07104253
787	RWmaxobs150 =	150
788	RWminobs150 =	150
789	RWrmse150 =	.07151194
790	RWmaxobs147 =	147
791	RWminobs147 =	147
792	RWrmse147 =	.0724523
793	RWmaxobs144 =	144
794	RWminobs144 =	144
795	RWrmse144 =	.07292486
796	RWmaxobs141 =	141
797	RWminobs141 =	141
798	RWrmse141 =	.07359791
799	RWmaxobs138 =	138
800	RWminobs138 =	138
801	RWrmse138 =	.07415885
802	RWmaxobs135 =	135
803	RWminobs135 =	135
804	RWrmse135 =	.07551713
805	RWmaxobs132 =	132
806	RWminobs132 =	132
807	RWrmse132 =	.07895271
808	RWmaxobs129 =	129
809	RWminobs129 =	129
810	RWrmse129 =	.08680818
811	RWmaxobs126 =	126
812	RWminobs126 =	126
813	RWrmse126 =	.08791729
814	RWmaxobs123 =	123
815	RWminobs123 =	123
816	RWrmse123 =	.0885621
817	RWmaxobs120 =	120
818	RWminobs120 =	120
819	RWrmse120 =	.0893681
820	RWmaxobs117 =	117
821	RWminobs117 =	117
822	RWrmse117 =	.09222918

823	RWmaxobs114 =	114
824	RWminobs114 =	114
825	RWrmse114 =	.09918976
826	RWmaxobs111 =	111
827	RWminobs111 =	111
828	RWrmse111 =	.10159669
829	RWmaxobs108 =	108
830	RWminobs108 =	108
831	RWrmse108 =	.1074872
832	RWmaxobs105 =	105
833	RWminobs105 =	105
834	RWrmse105 =	.11024241
835	RWmaxobs102 =	102
836	RWminobs102 =	102
837	RWrmse102 =	.11448013
838	RWmaxobs99 =	99
839	RWminobs99 =	99
840	RWrmse99 =	.11681123
841	RWmaxobs96 =	96
842	RWminobs96 =	96
843	RWrmse96 =	.12412703
844	RWmaxobs93 =	93
845	RWminobs93 =	93
846	RWrmse93 =	.13010923
847	RWmaxobs90 =	90
848	RWminobs90 =	90
849	RWrmse90 =	.12958828
850	RWmaxobs87 =	87
851	RWminobs87 =	87
852	RWrmse87 =	.13728567
853	RWmaxobs84 =	84
854	RWminobs84 =	84
855	RWrmse84 =	.14028654
856	RWmaxobs81 =	81
857	RWminobs81 =	81
858	RWrmse81 =	.14073066
859	RWmaxobs78 =	78
860	RWminobs78 =	78
861	RWrmse78 =	.14462066
862	RWmaxobs75 =	75
863	RWminobs75 =	75
864	RWrmse75 =	.14520888
865	RWmaxobs72 =	72
866	RWminobs72 =	72
867	RWrmse72 =	.14882576
868	RWmaxobs69 =	69
869	RWminobs69 =	69
870	RWrmse69 =	.16426363

871	RWmaxobs66 =	66
872	RWminobs66 =	66
873	RWrmse66 =	.16732114
874	RWmaxobs63 =	63
875	RWminobs63 =	63
876	RWrmse63 =	.1679914
877	RWmaxobs60 =	60
878	RWminobs60 =	60
879	RWrmse60 =	.18224837
880	RWmaxobs57 =	57
881	RWminobs57 =	57
882	RWrmse57 =	.19923544
883	RWmaxobs54 =	54
884	RWminobs54 =	54
885	RWrmse54 =	.2008839
886	RWmaxobs51 =	51
887	RWminobs51 =	51
888	RWrmse51 =	.2096647
889	RWmaxobs48 =	48
890	RWminobs48 =	48
891	RWrmse48 =	.20847129
892	RWmaxobs45 =	45
893	RWminobs45 =	45
894	RWrmse45 =	.2244767
895	RWmaxobs42 =	42
896	RWminobs42 =	42
897	RWrmse42 =	.25720484
898	RWmaxobs39 =	39
899	RWminobs39 =	39
900	RWrmse39 =	.24835914
901	RWmaxobs36 =	36
902	RWminobs36 =	36
903	RWrmse36 =	.24045465
904	RWmaxobs33 =	33
905	RWminobs33 =	33
906	RWrmse33 =	.26919986
907	RWmaxobs30 =	30
908	RWminobs30 =	30
909	RWrmse30 =	.49888354
910	RWmaxobs27 =	27
911	RWminobs27 =	27
912	RWrmse27 =	.32666868
913	RWmaxobs24 =	24
914	RWminobs24 =	24
915	RWrmse24 =	.66642366
916	RWmaxobs21 =	21
917	RWminobs21 =	21
918	RWrmse21 =	.34614878

```

919     RWmaxobs18 =      18
920     RWminobs18 =      18
921     RWrmse18 =    .757383
922     RWmaxobs15 =      15
923     RWminobs15 =      15
924     RWrmse15 =    .21744051
925     RWmaxobs12 =      12
926     RWminobs12 =      12
927     RWrmse12 =    .0132376
928     RWmaxobs9  =       9
929     RWminobs9  =       9
930     RWrmse9    =    .01629782
931     RWmaxobs6  =       6
932     RWminobs6  =       6
933     RWrmse6    =    .0162575
934     RWmaxobs3  =       3
935     RWminobs3  =       3
936     RWrmse3    =    .0206163
937
938     155 . /*
939         > RWmaxobs12 =      12
940         > RWminobs12 =      12
941         > RWrmse12 =    .0132376
942         > */
943     156 .
944     157 .
945     158 . * Going with model 1 because average RWrmse is lower across window
sizes
946     159 . scalar rwrmsc = .0132376
947
948     160 . reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 if
tin(,2021
949         > m3)
950
951           Source |           SS           df           MS       Number of obs   =
371
952     -----+-----
19.13
953           Model |    .024491123             14    .001749366       Prob > F           =
0.0000
954           Residual |    .032550038           356    .000091433       R-squared           =
0.4294
955     -----+-----
0.4069
956           Total |    .057041161           370    .000154165       Root MSE           =
.00956
957

```

```

958 -----
959 -----
960 -----+-----
961 -----
962      D.lnTotal |      Coef.   Std. Err.      t    P>|t|      [95% Conf.
Interval]
963 -----+-----
964      lnTotal |
965      LD. |   -.0016173   .0529806   -0.03   0.976   -.1058116
966      .1025769
967      L2D. |   -.1325018   .0525096   -2.52   0.012   -.2357698
968      -.0292338
969      L3D. |    .026173    .0529824    0.49   0.622   -.0780247
970      .1303708
971      |
972      m1 |   -.0254857   .0024679  -10.33   0.000   -.0303391
973      -.0206322
974      m2 |    .0004869   .0029045    0.17   0.867   -.0052252
975      .006199
976      m3 |   -.003913    .0027686   -1.41   0.158   -.0093579
977      .0015318
978      m4 |   -.013317    .0026543   -5.02   0.000   -.0185371
979      -.0080969
980      m5 |   -.0064632    .0026182   -2.47   0.014   -.0116124
981      -.0013141
982      m6 |   -.0088465    .0025835   -3.42   0.001   -.0139273
983      -.0037657
984      m7 |   -.0159741    .0025331   -6.31   0.000   -.0209559
985      -.0109924
986      m8 |   -.0051433    .0026374   -1.95   0.052   -.0103302
987      .0000435
988      m9 |   -.0096793    .0025635   -3.78   0.000   -.0147208
989      -.0046378
990      m10 |   -.0021472    .0025489   -0.84   0.400   -.00716
991      .0028657
992      m11 |    .0024587    .0024684    1.00   0.320   -.0023958
993      .0073131
994      _cons |    .0088125    .0018473    4.77   0.000    .0051795
995      .0124454
996 -----
997 -----
998
999      161 . predict pd
1000          (option xb assumed; fitted values)
1001          (15 missing values generated)
1002
1003      162 . gen pflcount=exp((rwrmsc^2)/2)*exp(1.lnTotal+pd) if Date==tm(2021m4)
1004          (386 missing values generated)
1005
1006

```

```

987     163 . gen ub1=exp((rwrmsse^2)/2)*exp(1.lnTotal+pd+1*rwrmsse) if
Date==tm(2021m4)
988         (386 missing values generated)
989
990     164 . gen lb1=exp((rwrmsse^2)/2)*exp(1.lnTotal+pd-1*rwrmsse) if
Date==tm(2021m4)
991         (386 missing values generated)
992
993     165 . gen ub2=exp((rwrmsse^2)/2)*exp(1.lnTotal+pd+2*rwrmsse) if
Date==tm(2021m4)
994         (386 missing values generated)
995
996     166 . gen lb2=exp((rwrmsse^2)/2)*exp(1.lnTotal+pd-2*rwrmsse) if
Date==tm(2021m4)
997         (386 missing values generated)
998
999     167 . gen ub3=exp((rwrmsse^2)/2)*exp(1.lnTotal+pd+3*rwrmsse) if
Date==tm(2021m4)
1000         (386 missing values generated)
1001
1002     168 . gen lb3=exp((rwrmsse^2)/2)*exp(1.lnTotal+pd-3*rwrmsse) if
Date==tm(2021m4)
1003         (386 missing values generated)
1004
1005     169 . drop pd
1006
1007     170 .
1008     171 . replace pflcount=Total if Date==tm(2021m3)
1009         (1 real change made)
1010
1011     172 . replace ub1=Total if Date==tm(2021m3)
1012         (1 real change made)
1013
1014     173 . replace ub2=Total if Date==tm(2021m3)
1015         (1 real change made)
1016
1017     174 . replace ub3=Total if Date==tm(2021m3)
1018         (1 real change made)
1019
1020     175 . replace lb1=Total if Date==tm(2021m3)
1021         (1 real change made)
1022
1023     176 . replace lb2=Total if Date==tm(2021m3)
1024         (1 real change made)
1025
1026     177 . replace lb3=Total if Date==tm(2021m3)
1027         (1 real change made)
1028

```

```

1029      178 .
1030      179 . twoway (tsrline ub3 ub2 if tin(2020m4,2021m4), ///
1031          >         recast(rarea) fcolor(orange) fintensity(20) lwidth(none) ) ///
1032          >         (tsrline ub2 ub1 if tin(2020m4,2021m4), ///
1033          >         recast(rarea) fcolor(green) fintensity(40) lwidth(none) ) ///
1034          >         (tsrline ub1 pflcount if tin(2020m4,2021m4), ///
1035          >         recast(rarea) fcolor(purple) fintensity(65) lwidth(none) ) ///
1036          >         (tsrline pflcount lb1 if tin(2020m4,2021m4), ///
1037          >         recast(rarea) fcolor(purple) fintensity(65) lwidth(none) ) ///
1038          >         (tsrline lb1 lb2 if tin(2020m4,2021m4), ///
1039          >         recast(rarea) fcolor(green) fintensity(40) lwidth(none) ) ///
1040          >         (tsrline lb2 lb3 if tin(2020m4,2021m4), ///
1041          >         recast(rarea) fcolor(orange) fintensity(20) lwidth(none) ) ///
1042          >         (tsline Total pflcount if tin(2020m4,2021m4) , ///
1043          >         lcolor(gs12 pink) lwidth(medthick medthick) ///
1044          >         lpattern(solid longdash)), scheme(slmono) legend(off)
1045
1046      180 . graph export "TotalFan1.png", replace
1047          (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Problem Sets
1048          > /Final Exam/TotalFan1.png written in PNG format)
1049
1050      181 .
1051      182 . * More than 1 step
1052      183 . arima d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 if
tin(199
1053          > 0m1,2021m3)
1054
1055          (setting optimization to BHHH)
1056          Iteration 0:   log likelihood =   1206.3625
1057          Iteration 1:   log likelihood =   1206.3625
1058
1059          ARIMA regression
1060
1061          Sample:   1990m5 - 2021m3
1062
1063          371
1064
1065          Wald chi2(14)   =
83.94
1066          Log likelihood =   1206.362
1067          Prob > chi2     =
0.0000
1068
1069          -----
1070          -----
1071
1072          |
1073          D.lnTotal | Coef.   Std. Err.   z   P>|z|   [95% Conf.
Interval]
1074
1075          -----+-----
1076          -----

```

```

1069      lnTotal      |
1070      lnTotal      |
1071      LD. | -.0016173 .0448009 -0.04 0.971 -.0894254
      .0861907
1072      L2D. | -.1325018 .0486537 -2.72 0.006 -.2278614
      -.0371422
1073      L3D. | .026173 .0400229 0.65 0.513 -.0522705
      .1046166
1074      |
1075      m1 | -.0254857 .0054401 -4.68 0.000 -.0361481
      -.0148232
1076      m2 | .0004869 .0068588 0.07 0.943 -.0129561
      .0139299
1077      m3 | -.003913 .0058695 -0.67 0.505 -.0154171
      .007591
1078      m4 | -.013317 .0047531 -2.80 0.005 -.0226329
      -.0040012
1079      m5 | -.0064632 .0082351 -0.78 0.433 -.0226038
      .0096773
1080      m6 | -.0088465 .0069016 -1.28 0.200 -.0223735
      .0046804
1081      m7 | -.0159741 .0056734 -2.82 0.005 -.0270938
      -.0048545
1082      m8 | -.0051433 .0074271 -0.69 0.489 -.0197002
      .0094135
1083      m9 | -.0096793 .0051505 -1.88 0.060 -.0197741
      .0004155
1084      m10 | -.0021472 .0048943 -0.44 0.661 -.0117399
      .0074456
1085      m11 | .0024587 .0066745 0.37 0.713 -.0106231
      .0155405
1086      _cons | .0088125 .0042033 2.10 0.036 .0005741
      .0170508
1087      -----+-----
      -----
1088      /sigma | .0093667 .0001378 67.96 0.000 .0090966
      .0096369
1089      -----
      -----
1090      Note: The test of the variance against zero is one sided, and the two-
      sided
1091      confidence interval is truncated at zero.
1092
1093      184 . predict pnonfarm, dynamic(tm(2021m3))
1094      (option xb assumed; predicted values)
1095      (4 missing values generated)
1096
1097      185 . predict mse, mse dynamic(mofd(tm(2021m4)))

```

```

1098
1099     186 . gen totmse = mse if Date==tm(2021m4)
1100         (386 missing values generated)
1101
1102     187 . replace totmse = l.totmse+mse if Date>tm(2021m4)
1103         (11 real changes made)
1104
1105     188 . gen pnonfarma = Total if Date==tm(2021m3)
1106         (386 missing values generated)
1107
1108     189 . replace pnonfarma = l.pnonfarma*exp(pnonfarm+mse/2) if Date>tm(2021m3)
1109         (12 real changes made)
1110
1111     190 .
1112     191 . gen ub1a = pnonfarma*exp(totmse^.5)
1113         (375 missing values generated)
1114
1115     192 . gen ub2a = pnonfarma*exp(2*totmse^.5)
1116         (375 missing values generated)
1117
1118     193 . gen ub3a = pnonfarma*exp(3*totmse^.5)
1119         (375 missing values generated)
1120
1121     194 . gen lb1a = pnonfarma/exp(totmse^.5)
1122         (375 missing values generated)
1123
1124     195 . gen lb2a = pnonfarma/exp(2*totmse^.5)
1125         (375 missing values generated)
1126
1127     196 . gen lb3a = pnonfarma/exp(3*totmse^.5)
1128         (375 missing values generated)
1129
1130     197 .
1131     198 . replace ub1a=Total if Date == tm(2021m3)
1132         (1 real change made)
1133
1134     199 . replace ub2a=Total if Date == tm(2021m3)
1135         (1 real change made)
1136
1137     200 . replace ub3a=Total if Date == tm(2021m3)
1138         (1 real change made)
1139
1140     201 . replace lb1a=Total if Date == tm(2021m3)
1141         (1 real change made)
1142
1143     202 . replace lb2a=Total if Date == tm(2021m3)
1144         (1 real change made)
1145

```



```

1146      203 . replace lb3a=Total if Date == tm(2021m3)
1147          (1 real change made)
1148
1149      204 .
1150      205 . twoway (tsrline ub3a ub2a if tin(2019m1,2022m3), ///
1151          >         recast(rarea) fcolor(red) fintensity(20) lwidth(none) ) ///
1152          >         (tsrline ub2a ub1a if tin(2019m1,2022m3), ///
1153          >         recast(rarea) fcolor(yellow) fintensity(40) lwidth(none) ) ///
1154          >         (tsrline ub1a pnonfarma if tin(2019m1,2022m3), ///
1155          >         recast(rarea) fcolor(blue) fintensity(65) lwidth(none) ) ///
1156          >         (tsrline pnonfarma lb1a if tin(2019m1,2022m3), ///
1157          >         recast(rarea) fcolor(blue) fintensity(65) lwidth(none) ) ///
1158          >         (tsrline lb1a lb2a if tin(2019m1,2022m3), ///
1159          >         recast(rarea) fcolor(yellow) fintensity(40) lwidth(none) ) ///
1160          >         (tsrline lb2a lb3a if tin(2019m1,2022m3), ///
1161          >         recast(rarea) fcolor(red) fintensity(20) lwidth(none) ) ///
1162          >         (tsline Total pnonfarma if tin(2019m1,2022m3) , ///
1163          >         lcolor(gs12 pink) lwidth(medthick medthick) ///
1164          >         lpattern(solid longdash)) , scheme(slmono) legend(off)
1165
1166      206 . graph export "TotalFan12.png", replace
1167          (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Problem Sets
1168          > /Final Exam/TotalFan12.png written in PNG format)
1169
1170      207 .
1171      208 . scalar rmse_mod1 = .0132376
1172
1173      209 . reg d.lnTotal l(1/3)d.lnTotal m1 m2 m3 m4 m5 m6 m7 m8 m9 m10 m11 if
tin(1990m
1174          > 1,2021m3)
1175
1176          Source |          SS          df          MS      Number of obs      =
371
1177          -----+-----
19.13
1178          Model |   .024491123          14   .001749366      Prob > F          =
0.0000
1179          Residual |   .032550038         356   .000091433      R-squared          =
0.4294
1180          -----+-----
0.4069
1181          Total |   .057041161         370   .000154165      Root MSE          =
.00956
1182
1183          -----
-----

```

1184	D.lnTotal	Coef.	Std. Err.	t	P> t	[95% Conf.
	Interval]					
1185	-----+-----					

1186	lnTotal					
1187	LD.	-.0016173	.0529806	-0.03	0.976	-.1058116
	.1025769					
1188	L2D.	-.1325018	.0525096	-2.52	0.012	-.2357698
	-.0292338					
1189	L3D.	.026173	.0529824	0.49	0.622	-.0780247
	.1303708					
1190						
1191	m1	-.0254857	.0024679	-10.33	0.000	-.0303391
	-.0206322					
1192	m2	.0004869	.0029045	0.17	0.867	-.0052252
	.006199					
1193	m3	-.003913	.0027686	-1.41	0.158	-.0093579
	.0015318					
1194	m4	-.013317	.0026543	-5.02	0.000	-.0185371
	-.0080969					
1195	m5	-.0064632	.0026182	-2.47	0.014	-.0116124
	-.0013141					
1196	m6	-.0088465	.0025835	-3.42	0.001	-.0139273
	-.0037657					
1197	m7	-.0159741	.0025331	-6.31	0.000	-.0209559
	-.0109924					
1198	m8	-.0051433	.0026374	-1.95	0.052	-.0103302
	.0000435					
1199	m9	-.0096793	.0025635	-3.78	0.000	-.0147208
	-.0046378					
1200	m10	-.0021472	.0025489	-0.84	0.400	-.00716
	.0028657					
1201	m11	.0024587	.0024684	1.00	0.320	-.0023958
	.0073131					
1202	_cons	.0088125	.0018473	4.77	0.000	.0051795
	.0124454					
1203	-----					

1204						
1205	210 . predict plTotal					
1206	(option xb assumed; fitted values)					
1207	(15 missing values generated)					
1208						
1209	211 . predict temp if tin(2021m3,2021m3)					
1210	(option xb assumed; fitted values)					
1211	(386 missing values generated)					
1212						
1213	212 . replace plTotal=temp if tin(2021m3,2021m3)					

```

1214         (0 real changes made)
1215
1216     213 . drop temp
1217
1218     214 . gen pTotal=exp(1.lnTotal+plTotal+(rmse_mod1^2)/2)
1219         (15 missing values generated)
1220
1221     215 . gen lbTotal=exp(1.lnTotal+plTotal-1.96*rmse_mod1+(rmse_mod1^2)/2)
1222         (15 missing values generated)
1223
1224     216 . gen ubTotal=exp(1.lnTotal+plTotal+1.96*rmse_mod1+(rmse_mod1^2)/2)
1225         (15 missing values generated)
1226
1227     217 .
1228     218 . gen res=(d.lnTotal-plTotal)
1229         (16 missing values generated)
1230
1231     219 . gen expres=exp(res)
1232         (16 missing values generated)
1233
1234     220 . summ expres
1235
1236         Variable |           Obs       Mean   Std. Dev.       Min       Max
1237     -----+-----
1238         expres |           371    1.000042    .0088821    .8565838    1.028623
1239
1240     221 . scalar meanexpres=r(mean)
1241
1242     222 . gen epTotal=exp(1.lnTotal+plTotal)*meanexpres
1243         (15 missing values generated)
1244
1245     223 . _pctile res, percentile(2.5,97.5)
1246
1247     224 . return list
1248
1249         scalars:
1250
1251                 r(r1) =   -.0089262239634991
1252                 r(r2) =   .0108835604041815
1253
1254     225 . gen elbTotal=exp(1.lnTotal+plTotal+r(r1))*meanexpres
1255         (15 missing values generated)
1256
1257     226 . gen eubTotal=exp(1.lnTotal+plTotal+r(r2))*meanexpres
1258         (15 missing values generated)
1259
1260     227 .
1260     228 . tsline Total pTotal elbTotal eubTotal lbTotal ubTotal if
tin(2019m1,2021m4),

```

```

1261      > ///
1262      >      scheme(slmono) tline(2021m3, lcolor(gs4)) ///
1263      >      lpattern(solid solid longdash longdash shortdash shortdash) ///
1264      >      lcolor(dkorange gs5 gs10 gs10 dkorange%60 dkorange%60) ///
1265      >      lwidth(medthick medthick medium medium)
1266
1267      229 . graph export "interval_tsline.png", replace
1268      (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Problem Sets
1269      > /Final Exam/interval_tsline.png written in PNG format)
1270
1271      230 .
1272      231 . histogram expres, normal kdensity saving(residuals.gph, replace)
1273      (bin=19, start=.85658383, width=.00905468)
1274      (file residuals.gph saved)
1275
1276      232 . graph export "residuals.png", replace
1277      (file /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Problem Sets
1278      > /Final Exam/residuals.png written in PNG format)
1279
1280      233 .
1281      234 . log close
1282      name: <unnamed>
1283      log: /Users/guslipkin/Documents/Spring2020/CAP 4763 ~ Time
Series/Probl
1284      > em Sets/Final Exam/Final Exam.smcl
1285      log type: smcl
1286      closed on: 29 Apr 2021, 12:09:27
1287      -----
-----

```