```
R version 4.0.2 (2020-06-22) -- "Taking Off Again"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin17.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.72 (7847) x86_64-apple-darwin17.0]

[History restored from /Users/andreasmurr/.Rapp.history]

> setwd('/Users/andreasmurr/Dropbox/Citizen-Forecasting-2020/replication-archive')
> # citizen forecasting 2020: a state-by-state experiment
> # andreas murr & mike lewis-beck
> # analysis
>
> # ==========
> # = set up =
> # ==========
>
> # clear working memory
>
> rm(list=ls())
>
> # write leave one out cross-validation function
>
> loo.errors = function(x){
+  e = x$residuals
+  h = lm.influence(x)$hat
+  sqrt(mean((e / (1 - h))^2))
+ }
>
> # load packages
>
> library(arm)
Loading required package: MASS
Loading required package: Matrix
Loading required package: lme4

arm (Version 1.11-1, built: 2020-4-27)

Working directory is /Users/andreasmurr/Dropbox/Citizen-Forecasting-2020/replication-archive

>
> # load data
>
> load("AnesInterestNationState.RData")
> load("AnesNation.RData")
> load("AnesState.RData")
```

```
> load("MturkInterestNationState.RData")
> load("MturkNation3.RData")
> load("MturkState3.RData")
>
> # ===========
> # = table 1 =
> # ===========
>
> data = na.omit(AnesNation)
> m.tab1 = lm(I(pop.dem*100) ~ I(p.dem*100), data=data)
> display(m.tab1, digits=1)
lm(formula = I(pop.dem * 100) ~ I(p.dem * 100), data = data)
               coef.est coef.se
(Intercept)    39.3      1.5
I(p.dem * 100)  0.2      0.0
---
n = 17, k = 2
residual sd = 2.9, R-Squared = 0.77
> loo.errors(m.tab1)
[1] 3.065488
>
> # ===========
> # = table 2 =
> # ===========
>
> # popular vote
>
> m1 = lm(pop.dem ~ p.dem, data=AnesNation[AnesNation$year < 2008,])
> m2 = lm(pop.dem ~ p.dem, data=AnesNation[AnesNation$year < 2012,])
> m3 = lm(pop.dem ~ p.dem, data=AnesNation[AnesNation$year < 2016,])
>
> y.hat = rep(NA, 3)
> y.hat[1] = predict(m1, AnesNation[AnesNation$year == 2008,])
> y.hat[2] = predict(m2, AnesNation[AnesNation$year == 2012,])
> y.hat[3] = predict(m3, AnesNation[AnesNation$year == 2016,])
>
> y = rep(NA, 3)
> y[1] = AnesNation$pop.dem[AnesNation$year == 2008]
> y[2] = AnesNation$pop.dem[AnesNation$year == 2012]
> y[3] = AnesNation$pop.dem[AnesNation$year == 2016]
>
> t1 = round(cbind(y, y.hat, y-y.hat)*100,1)
>
> # electoral vote (data)
>
> y.hat = rep(NA, 3)
> y = rep(NA, 3)
> j = 1
> for (i in c(2008, 2012, 2016)){
+  y.hat[j] = sum(with(AnesState[AnesState$year==i,], ifelse(forecast=="t" |
is.na(forecast)==TRUE, electoralv / 2, ifelse(forecast=="d", electoralv, 0))))
+  y[j] = sum(with(AnesState[AnesState$year==i,], ifelse(winner=="d", electoralv, 0)))
+  j = j + 1
+ }
>
> table(AnesState$forecast[AnesState$year==2008], useNA="always")

   d    r <NA>
  21   13   17
> table(AnesState$forecast[AnesState$year==2012], useNA="always")
```

```
   d    r <NA>
  27   24    0
> table(AnesState$forecast[AnesState$year==2016], useNA="always")

   d    r <NA>
  24   27    0
>
> with(AnesState[AnesState$year==2008,], sum(electoralv[is.na(forecast)]))
[1] 88
>
> t2 = cbind(y, y.hat, y-y.hat)
>
> # electoral vote (data)
>
> y.hat = rep(NA, 3)
> y = rep(NA, 3)
> j = 1
> for (i in c(2008, 2012, 2016)){
+  train = AnesState[AnesState$year < i,]
+  test = AnesState[AnesState$year == i,]
+  m = lm(dem.two.s ~ p.dem, data=train, weights=train$n)
+  test$y.hat = ifelse(predict(m, test) > .5, "d", "r")
+  y.hat[j] = sum(with(test, ifelse(is.na(y.hat)==TRUE, electoralv / 2, ifelse(y.hat=="d",
electoralv, 0))))
+  y[j] = sum(with(test, ifelse(winner=="d", electoralv, 0)))
+  j = j + 1
+ }
>
> t3 = cbind(y, y.hat, y-y.hat)
>
> # combined
>
> cbind(t1, t2, t3)
        y y.hat        y y.hat       y y.hat
[1,] 53.7  53.9 -0.3 364   328  36 364   305  59
[2,] 52.0  53.1 -1.1 332   343 -11 332   301  31
[3,] 51.1  52.3 -1.3 233   279 -46 233   275 -42
>
> # ===========
> # = table 3 =
> # ===========
>
> data = na.omit(AnesState)
> m.tab3 = lm(I(dem.two.s*100) ~ I(p.dem*100), data=data, weights=data$n)
> display(m.tab3, digits=1)
lm(formula = I(dem.two.s * 100) ~ I(p.dem * 100), data = data,
    weights = data$n)
               coef.est coef.se
(Intercept)    33.0      0.5
I(p.dem * 100)  0.3      0.0
---
n = 450, k = 2
residual sd = 36.1, R-Squared = 0.75
>
> J = length(unique(data$year))
> rmse = rep(NA, nrow=J)
> for (j in 1:J){
+  data.train = data[data$year%in%unique(data$year)[-j],]
+  m = lm(dem.two.s ~ p.dem, data=data.train, weights=data.train$n)
+  data.fit = data[data$year==unique(data$year)[j],]
+  data.fit$y.hat = predict(m, data.fit)
```

```
+  rmse[j] = with(data.fit, sqrt(mean(((dem.two.s - y.hat)*100)^2)))
+ }
> mean(rmse)
[1] 6.442129
>
> # ===========
> # = table 4 =
> # ===========
>
> MturkNation3$forecast.data = ifelse(MturkNation3$p.dem>.5, "Biden", "Trump")
> y.hat = predict(m.tab1, MturkNation3) / 100
> forecast.model = ifelse(y.hat > .5, "Biden", "Trump")
> cbind(MturkNation3[c(3,2,5)], y.hat, forecast.model)
     n     p.dem forecast.data    y.hat forecast.model
1 2483 0.5134917         Biden 0.495956          Trump
>
> # ===========
> # = table 5 =
> # ===========
>
> MturkState3$y.hat = predict(m.tab3, MturkState3) / 100
> MturkState3[c("state","p.dem","y.hat")]
    state      p.dem     y.hat
709    AL 0.16666667 0.3804975
710    AK 0.52941176 0.4911143
711    AZ 0.31666667 0.4262390
712    AR 0.16363636 0.3795734
713    CA 0.41666667 0.4567334
714    CO 0.61666667 0.5177222
715    CT 0.38333333 0.4465686
716    DE 0.77142857 0.5649159
717    DC 0.83333333 0.5837933
718    FL 0.41666667 0.4567334
719    GA 0.35000000 0.4364038
720    HI 1.00000000 0.6346173
721    ID 0.03125000 0.3392030
722    IL 0.61666667 0.5177222
723    IN 0.16666667 0.3804975
724    IA 0.32758621 0.4295689
725    KS 0.22641509 0.3987174
726    KY 0.18644068 0.3865274
727    LA 0.25000000 0.4059095
728    ME 0.83333333 0.5837933
729    MD 0.70000000 0.5431342
730    MA 0.85000000 0.5888757
731    MI 0.60000000 0.5126398
732    MN 0.90000000 0.6041229
733    MS 0.13043478 0.3694488
734    MO 0.20000000 0.3906623
735    MT 0.25000000 0.4059095
736    NE 0.08823529 0.3565803
737    NV 0.46666667 0.4719806
738    NH 0.72500000 0.5507578
739    NJ 0.71666667 0.5482166
740    NM 0.88235294 0.5987416
741    NY 0.73333333 0.5532990
742    NC 0.41666667 0.4567334
743    ND 0.21052632 0.3938722
744    OH 0.48333333 0.4770630
745    OK 0.13333333 0.3703327
746    OR 0.83333333 0.5837933
```

```
747    PA 0.53333333 0.4923102
748    RI 0.80000000 0.5736285
749    SC 0.15000000 0.3754151
750    SD 0.19230769 0.3883165
751    TN 0.18333333 0.3855799
752    TX 0.35000000 0.4364038
753    UT 0.12500000 0.3677915
754    VT 0.78571429 0.5692722
755    VA 0.56666667 0.5024750
756    WA 0.71666667 0.5482166
757    WV 0.27500000 0.4135330
758    WI 0.45000000 0.4668982
759    WY 0.30000000 0.4211566
>
> # ===========
> # = table 6 =
> # ===========
>
> sum(with(MturkState3, electoralv*(p.dem>.5)))
[1] 204
> sum(with(MturkState3, electoralv*(p.dem<.5)))
[1] 334
> sum(with(MturkState3, electoralv*(p.dem==.5)))
[1] 0
>
> sum(with(MturkState3, electoralv*(y.hat>.5)))
[1] 181
> sum(with(MturkState3, electoralv*(y.hat<.5)))
[1] 357
> sum(with(MturkState3, electoralv*(y.hat==.5)))
[1] 0
>
> # ============
> # = figure 1 =
> # ============
>
> # create colouring for map, needs to be pasted in the svg file
>
> paste(paste("#", MturkState3$state[MturkState3$forecast=="Biden"], collapse=",", sep=""),
"{fill: #CCCCCC;}", sep=" ")
[1] "# {fill: #CCCCCC;}"
> paste(paste("#", MturkState3$state[MturkState3$forecast=="Trump"], collapse=",", sep=""),
"{fill: #525252;}", sep=" ")
[1] "# {fill: #525252;}"
>
> # ==============
> # = discussion =
> # ==============
>
> # are the state forecasts randomly?  we can reject the hypothesis at the p<.001 level
>
> d = with(MturkState3, p.dem*n)
> r = with(MturkState3, n-d)
> summary(glm(cbind(d,r)~1,family=binomial))

Call:
glm(formula = cbind(d, r) ~ 1, family = binomial)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.4142  -3.2368  -0.4556   3.4876   7.4376
```

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.21753    0.04037  -5.388 7.13e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 697.91  on 50  degrees of freedom
Residual deviance: 697.91  on 50  degrees of freedom
AIC: 895.13

Number of Fisher Scoring iterations: 3

>
> # proportion of forecasts that are the same for state and nation
>
> mean(with(MturkInterestNationState[MturkInterestNationState$forecast.state%in%c("d","r") &
MturkInterestNationState$forecast.nation%in%c("d","r"),],  forecast.state==forecast.nation))
[1] 0.7677419
>
> mean(with(AnesInterestNationState[AnesInterestNationState$forecast.state%in%c("d","r") &
AnesInterestNationState$forecast.nation%in%c("d","r"),],  forecast.state==forecast.nation))
[1] 0.7257479
>
> # interest
>
> mean(MturkInterestNationState$interest, na.rm=TRUE)
[1] 2.514931
>
> # ============================
> # = supplementary material =
> # ============================
>
> # table a1
>
> se = with(MturkNation3, sqrt(p.dem*(1-p.dem)/n))
> round((MturkNation3$p.dem + c(-1,1)*1.96*se)*100, 1)
[1] 49.4 53.3
> round(predict(m.tab1, MturkNation3) + c(-1,1)*loo.errors(m.tab1), 1)
[1] 46.5 52.7
>
> # table a2
>
> library(binom)
> tab.a1 = with(MturkState3, binom.confint(p.dem*n, n, methods="wilson"))
> tab.a1$data.tossup = ifelse(tab.a1$lower < .5 & tab.a1$upper > .5, 1, 0)
> tab.a1$state = MturkState3$state
> tab.a1$data.lower.r = round(tab.a1$lower*100, 1)
> tab.a1$data.upper.r = round(tab.a1$upper*100, 1)
>
> tab.a1$model.lower = MturkState3$y.hat - mean(rmse)/100
> tab.a1$model.upper = MturkState3$y.hat + mean(rmse)/100
> tab.a1$model.tossup = ifelse(tab.a1$model.lower < .5 & tab.a1$model.upper > .5, 1, 0)
>
> tab.a1$model.lower.r = round(tab.a1$model.lower*100, 1)
> tab.a1$model.upper.r = round(tab.a1$model.upper*100, 1)
>
> tab.a1[c("state", "data.lower.r", "data.upper.r", "data.tossup", "model.lower.r",
"model.upper.r", "model.tossup")]
```

```
   state data.lower.r data.upper.r data.tossup model.lower.r model.upper.r model.tossup
1     AL          9.3         28.0           0          31.6          44.5            0
2     AK         31.0         73.8           1          42.7          55.6            1
3     AZ         21.3         44.2           0          36.2          49.1            0
4     AR          8.9         28.3           0          31.5          44.4            0
5     CA         30.1         54.3           1          39.2          52.1            1
6     CO         49.0         72.9           1          45.3          58.2            1
7     CT         27.1         51.0           1          38.2          51.1            1
8     DE         61.0         87.9           0          50.0          62.9            0
9     DC         60.8         94.2           0          51.9          64.8            0
10    FL         30.1         54.3           1          39.2          52.1            1
11    GA         24.2         47.6           0          37.2          50.1            1
12    HI         84.5        100.0           0          57.0          69.9            0
13    ID          0.6         15.7           0          27.5          40.4            0
14    IL         49.0         72.9           1          45.3          58.2            1
15    IN          9.3         28.0           0          31.6          44.5            0
16    IA         22.1         45.6           0          36.5          49.4            0
17    KS         13.5         35.5           0          33.4          46.3            0
18    KY         10.7         30.4           0          32.2          45.1            0
19    LA         15.8         37.2           0          34.1          47.0            0
20    ME         66.4         92.7           0          51.9          64.8            0
21    MD         57.5         80.1           0          47.9          60.8            1
22    MA         73.9         91.9           0          52.4          65.3            0
23    MI         47.4         71.4           1          44.8          57.7            1
24    MN         79.9         95.3           0          54.0          66.9            0
25    MS          6.1         25.7           0          30.5          43.4            0
26    MO         11.8         31.8           0          32.6          45.5            0
27    MT         10.2         49.5           0          34.1          47.0            0
28    NE          3.0         23.0           0          29.2          42.1            0
29    NV         34.6         59.1           1          40.8          53.6            1
30    NH         57.2         83.9           0          48.6          61.5            1
31    NJ         59.2         81.5           0          48.4          61.3            1
32    NM         73.4         95.3           0          53.4          66.3            0
33    NY         61.0         82.9           0          48.9          61.8            1
34    NC         30.1         54.3           1          39.2          52.1            1
35    ND          8.5         43.3           0          32.9          45.8            0
36    OH         36.2         60.7           1          41.3          54.1            1
37    OK          6.9         24.2           0          30.6          43.5            0
38    OR         72.0         90.7           0          51.9          64.8            0
39    PA         40.9         65.4           1          42.8          55.7            1
40    RI         62.7         90.5           0          50.9          63.8            0
41    SC          8.1         26.1           0          31.1          44.0            0
42    SD          8.5         37.9           0          32.4          45.3            0
43    TN         10.6         29.9           0          32.1          45.0            0
44    TX         24.2         47.6           0          37.2          50.1            1
45    UT          6.2         23.6           0          30.3          43.2            0
46    VT         52.4         92.4           0          50.5          63.4            0
47    VA         44.1         68.4           1          43.8          56.7            1
48    WA         59.2         81.5           0          48.4          61.3            1
49    WV         16.1         42.8           0          34.9          47.8            0
50    WI         33.1         57.5           1          40.2          53.1            1
51    WY         10.8         60.3           1          35.7          48.6            0
>
> # table a3
>
>  # rmse electoral vote (data)
>  sel = unique(AnesState$year)[tapply(AnesState$forecast, AnesState$year, function(x)
{sum(is.na(x))!=length(x)})]
>  y.hat = rep(NA, length(sel))
>  y = rep(NA, length(sel))
>  j = 1
```

```
> for (i in sel){
+    y.hat[j] = sum(with(AnesState[AnesState$year==i,], ifelse(forecast=="t" |
is.na(forecast)==TRUE, electoralv / 2, ifelse(forecast=="d", electoralv, 0))))
+    y[j] = sum(with(AnesState[AnesState$year==i,], ifelse(winner=="d", electoralv, 0)))
+    j = j + 1
+ }
> rmse.data = sqrt(mean((y-y.hat)^2))
> # rmse electoral vote (model)
> y.hat = rep(NA, length(sel)-1)
> y = rep(NA, length(sel)-1)
> j = 1
> for (i in sel[-1]){
+    train = AnesState[AnesState$year < i,]
+    test = AnesState[AnesState$year == i,]
+    m = lm(dem.two.s ~ p.dem, data=train, weights=train$n)
+    test$y.hat = ifelse(predict(m, test) > .5, "d", "r")
+    y.hat[j] = sum(with(test, ifelse(is.na(y.hat)==TRUE, electoralv / 2, ifelse(y.hat=="d",
electoralv, 0))))
+    y[j] = sum(with(test, ifelse(winner=="d", electoralv, 0)))
+    j = j + 1
+ }
> rmse.model = sqrt(mean((y-y.hat)^2))
> # table
> round(sum(with(MturkState3, electoralv*(p.dem>.5)))+c(-1,1)*rmse.data)
[1] 105 303
> round(sum(with(MturkState3, electoralv*(p.dem<.5)))+c(-1,1)*rmse.data)
[1] 235 433
> round(sum(with(MturkState3, electoralv*(y.hat>.5)))+c(-1,1)*rmse.model)
[1]  92 270
> round(sum(with(MturkState3, electoralv*(y.hat<.5)))+c(-1,1)*rmse.model)

[1] 268 446
>
>
> # ====================
> # = end source code =
> # ====================
>
```