CSSS 512 HW 2

Chris Hess

April 19, 2018

Contents

# Problem 1 - U.S. House of Representatives

Part a

The time-series does show evidence of a substantial shock in 1994 where the average through the first portion of the series ($\approx$ 46 seat majority) does not approximate the short-run average post-1994 ($\approx$ -7). The ACF and PACF suggest that the time-series may be autoregressive order 1 before demeaning to the series, though there is not a strict pattern of geometric decline from L1. The Phillips-Perron and Augmented Dickey-Fuller tests disagree with each other, and this lack of conclusive evidence regarding stationarity likely is a function of the short series. Demeaning the series according to the 1994 structural break (i.e. pre and post) shows much stronger evidence in favor of stationarity, and also indicates that the initial evidence of autocorrelation was largely related to pre/post 1994 variation.

```
#plot ts, ACF, PACF
#run ADF and PP tests
#demean by pre/post 1994 periods
#plot ts, ACF, PACF
#what effect does 1994 structural break have on ts?

#pull ts, every other year frequency 1963-2017
ts_house <- ts(congress$DemHouseMaj,
               frequency = 1/2,
               start = 1963, end = 2017)

#plot as observed
autoplot(ts_house) +
  geom_hline(yintercept = 0, linetype = 3, color = "grey60") +
  geom_vline(xintercept = 1994, color = "Blue") +
  theme_minimal() +
  labs(title = "Democratic House Majority Time-series")
```
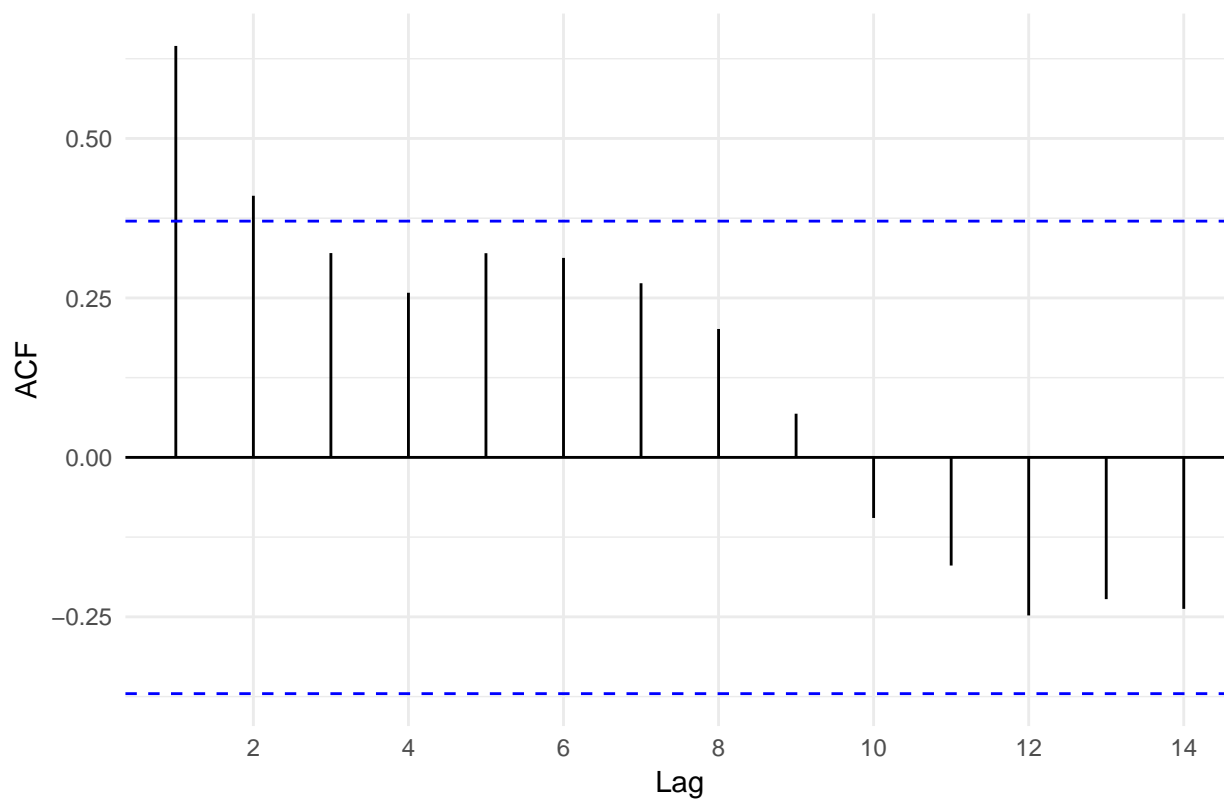
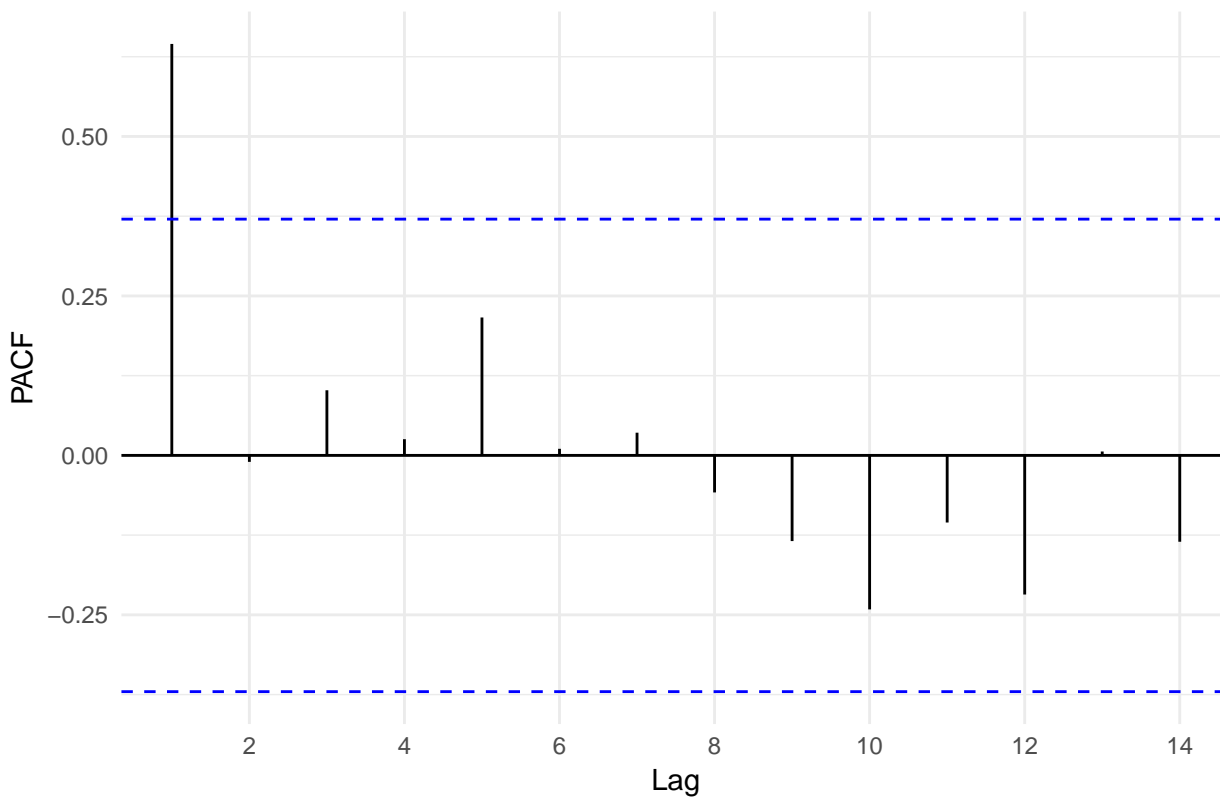# Democratic House Majority Time−series
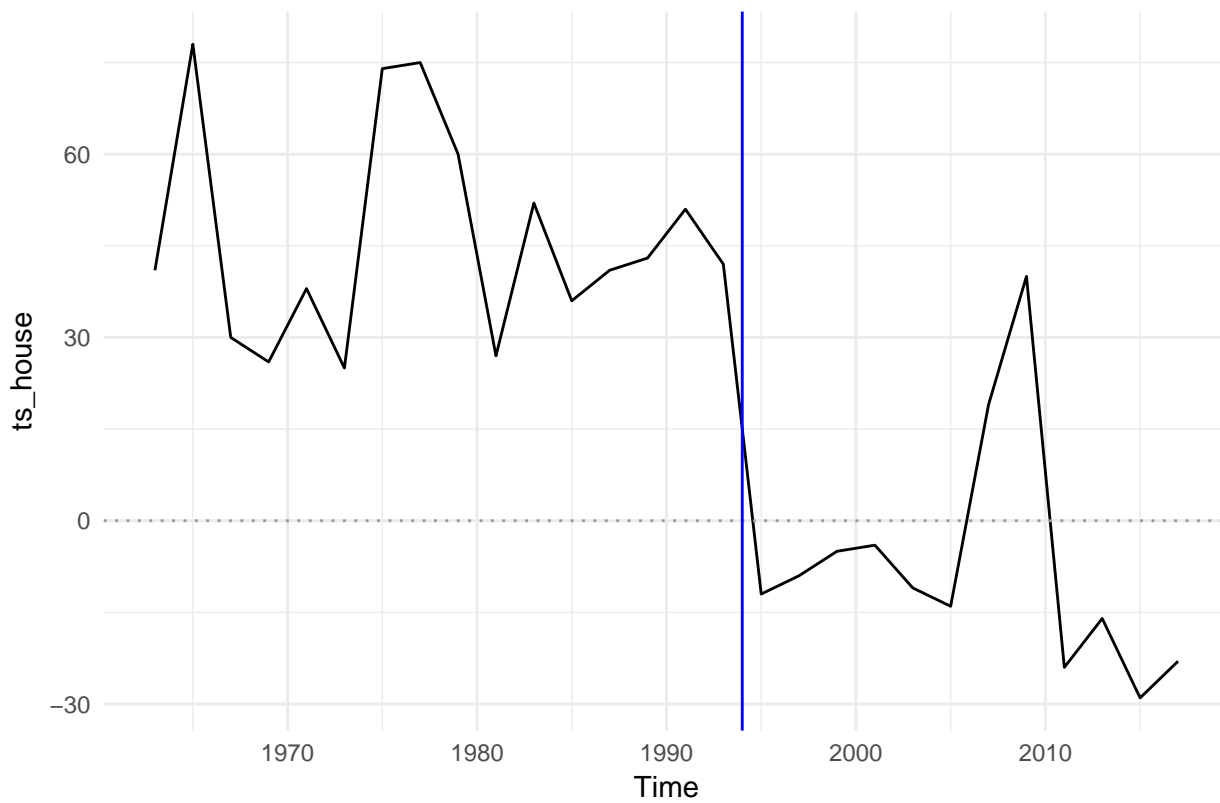

```r
#observed ACF
ggAcf(ts_house) + #substantial autocorrelation but not quite geometric decline
  theme_minimal() +
  labs(title = "Democratic House Majority ACF")
```

Democratic House Majority ACF


```r
#observed PACF
ggPacf(ts_house) + #AR(1), phi = .65 ish
  theme_minimal() +
  labs(title = "Democratic House Majority PACF")
```

Democratic House Majority PACF


```r
PP.test(ts_house)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  ts_house
## Dickey-Fuller = -3.6974, Truncation lag parameter = 2, p-value =
## 0.04239
```

```r
tseries::adf.test(ts_house)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_house
## Dickey-Fuller = -2.7382, Lag order = 3, p-value = 0.2901
## alternative hypothesis: stationary
```

```r
#pre-1994 mean for ts
preMean <- congress %>%
  filter(StartYear < 1994) %>%
  summarize(mean = mean(DemHouseMaj)) %>%
  pull(mean)

#post-1994 mean for ts
postMean <- congress %>%
  filter(StartYear >= 1994) %>%
  summarize(mean = mean(DemHouseMaj)) %>%
  pull(mean)

#demean based on pre/post 1994 (i.e. 1-16th obs vs 17-28th obs)
```

```
ts_house[1:16] <- ts_house[1:16] - preMean
ts_house[-1:-16] <- ts_house[-1:-16] - postMean

#demeaned ts - looks stationary now
autoplot(ts_house) +
  geom_hline(yintercept = 0, linetype = 3, color = "grey60") +
  geom_vline(xintercept = 1994, color = "Blue") +
  theme_minimal() +
  labs(title = "Democratic House Majority Time-series")
```

Democratic House Majority Time−series
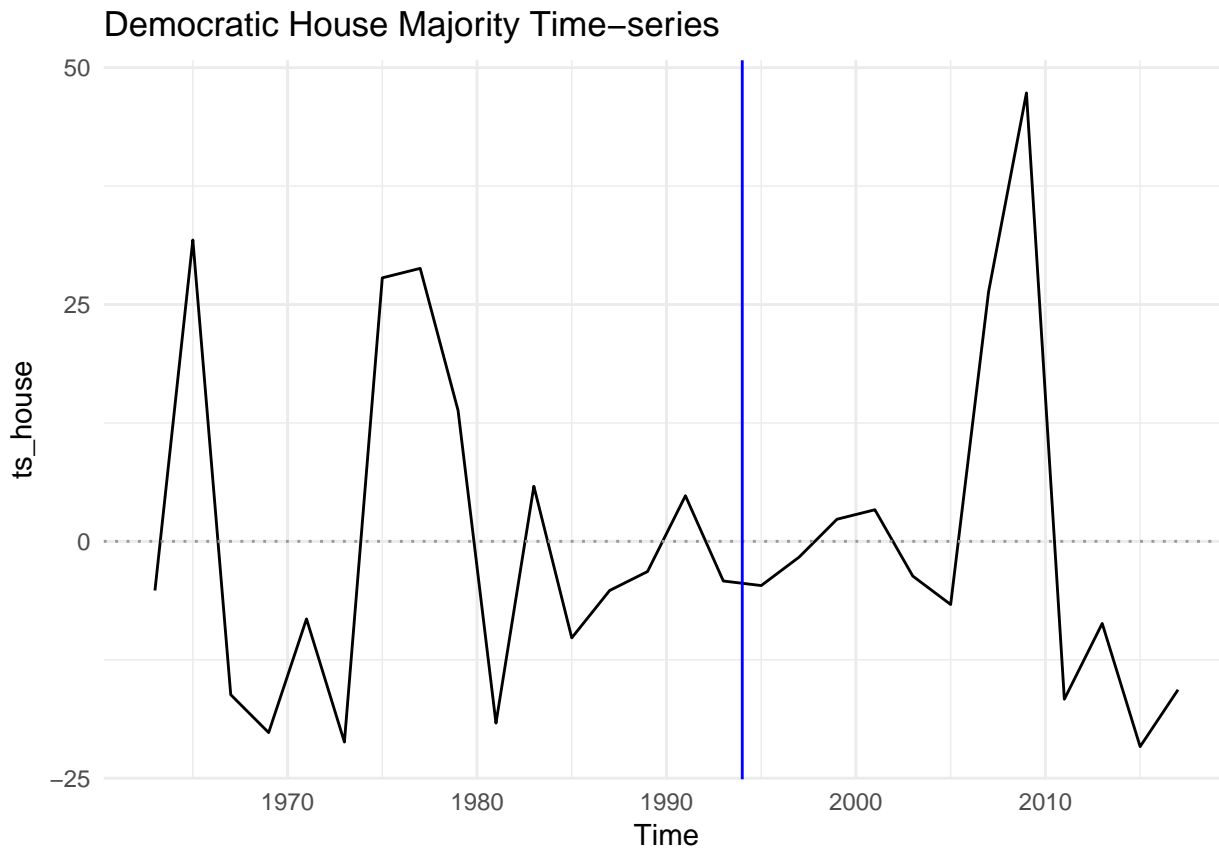

```
#demeaned ACF
ggAcf(ts_house) + #no autocorrelation
  theme_minimal() +
  labs(title = "Democratic House Majority ACF")
```

Democratic House Majority ACF
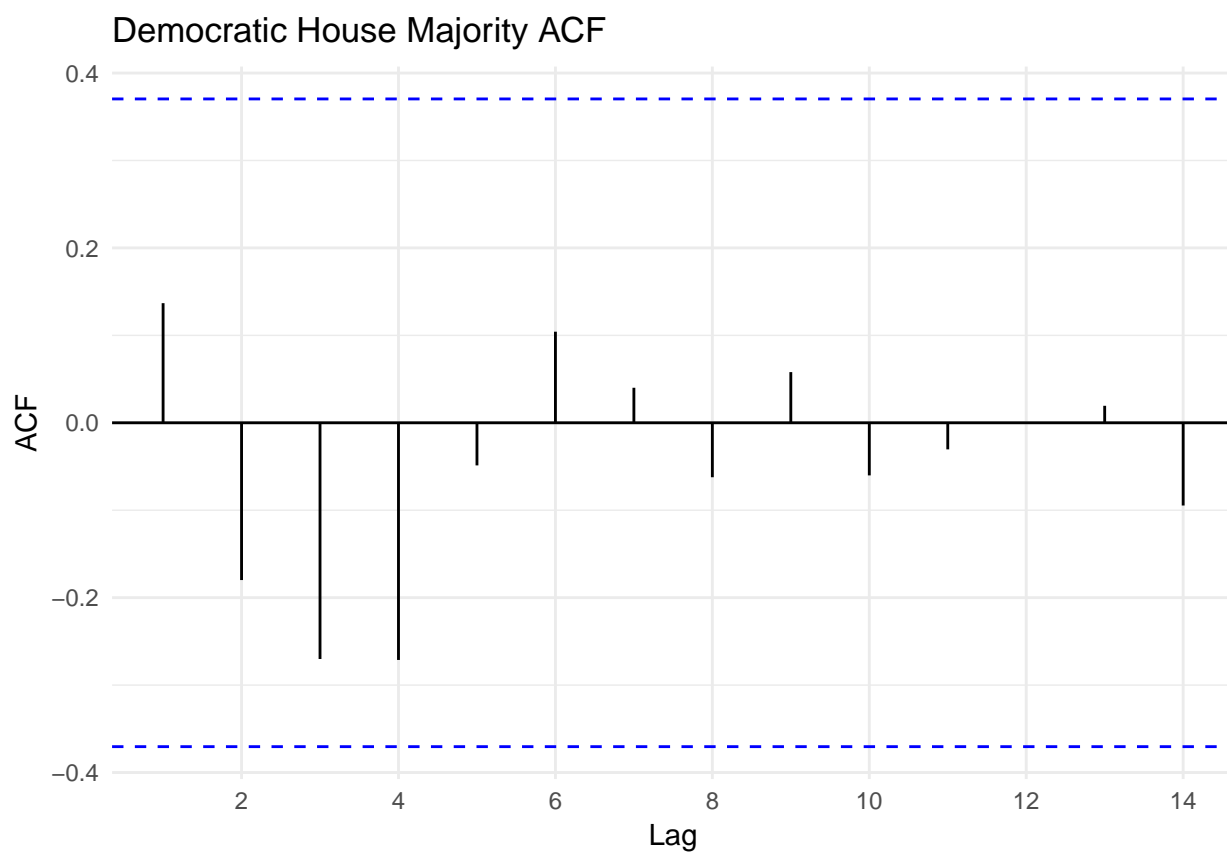

```r
#demeaned PACF
ggPacf(ts_house) + #no need for lags
  theme_minimal() +
  labs(title = "Democratic House Majority PACF")
```

Democratic House Majority PACF
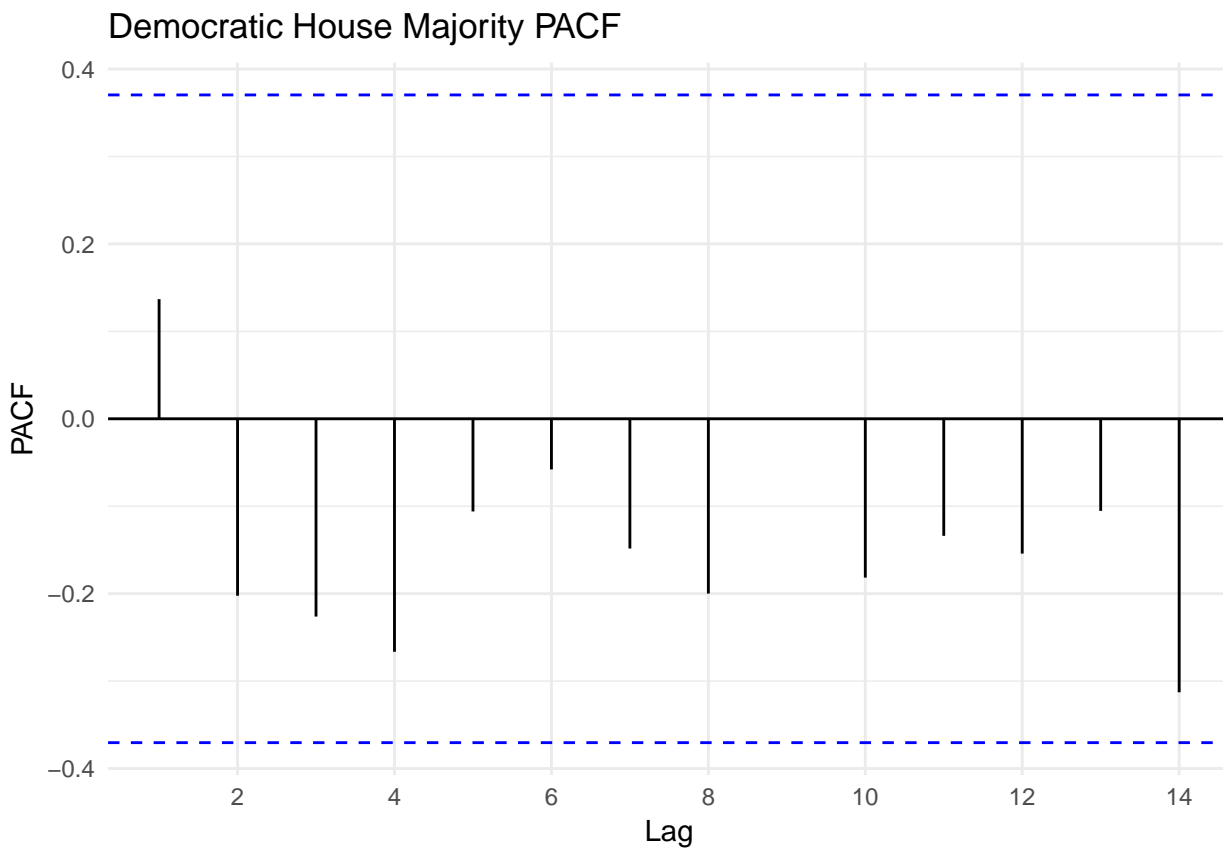
Part b

The AR(0)/OLS model of the Democratic House majority time-series shows support for the coattails theory and for the re-aligment shock in 1994. The clearest relationships are that the Southern re-alignment switched the make-up of the House towards majority-Republican (a difference of nearly 48 seats for pre 1994 elections), and that the party of the newly-elected president (in presidential election years) typically sees a proportional boost in the House. The evidence for the partisan midterm resurgence is a bit uncertain but still suggests that midterms tend to associate with the party not holding the presidency taking back some seats. There doesn't seem to be much salience to unemployment for the Democratic House majority, net of the other covariates. The model tended to miss the true value of the time-series by about 12-14 seats depending on the specification.

```
#pull ts, every other year frequency 1963-2017
ts_house <- ts(congress$DemHouseMaj,
               frequency = 1/2,
               start = 1963, end = 2017)

#create df of covariates
covar <- congress %>% select(PartisanMidterm, PartisanUnem, Coattails, Pre1994)

#function to provide sum stats that can be passed to xtable()
armaFit <- function(ts, order = c(0, 0, 0), seasonal.order = c(1, 0, 0),
                    seasonal.period = NA, xdf = NULL){
  if(is.na(seasonal.period)){
    mod <- Arima(ts, order = order, xreg = xdf)
  } else{
    mod <- Arima(ts, order = order,
                 seasonal = list(order = seasonal.order,
                                 period = seasonal.period), xreg = xdf)
  }
```

```r
    lab.order <- paste0("(", order[1], ",", order[2], ",", order[3], ")")
    aic <- round(mod[['aic']], 3)
    rmse <- round(sqrt(mean((ts - mod$fitted)^2)), 3)
    lenT <- length(ts)

    phi1 <- round(mod$coef["ar1"], 3)
    phi2 <- round(mod$coef["ar2"], 3)
    psi1 <- round(mod$coef["ma1"], 3)
    sar1 <- round(mod$coef["sar1"], 3)
    phi1 <- ifelse(is.na(phi1), "", phi1)
    phi2 <- ifelse(is.na(phi2), "", phi2)
    psi1 <- ifelse(is.na(psi1), "", psi1)
    sar1 <- ifelse(is.na(sar1), "", sar1)

    coef <- round(mod[['coef']], 3)
    coef <- round(coef[c((length(coef)-3),
                         (length(coef)-2),
                         (length(coef)-1),
                         length(coef))], 3)

    se <- round(sqrt(diag(mod[['var.coef']])), 3)
    se_phi1 <- round(se["ar1"], 3)
    se_phi2 <- round(se["ar2"], 3)
    se_psi1 <- round(se["ma1"], 3)
    se_sar1 <- round(se["sar1"], 3)
    se_phi1 <- ifelse(is.na(se_phi1), "", se_phi1)
    se_phi2 <- ifelse(is.na(se_phi2), "", se_phi2)
    se_psi1 <- ifelse(is.na(se_psi1), "", se_psi1)
    se_sar1 <- ifelse(is.na(se_sar1), "", se_sar1)
    se <- round(se[c((length(se)-3),
                     (length(se)-2),
                     (length(se)-1),
                     length(se))], 3)

    names(lab.order) <- "Order"
    names(aic) <- "AIC"
    names(rmse) <- "RMSE"
    names(lenT) <- "$T$"
    names(phi1) <- "$\\phi_1$"
    names(phi2) <- "$\\phi_2$"
    names(psi1) <- "$\\psi_1$"
    names(sar1) <- "S$\\phi_1$"

    oneline <- c(lab.order, aic, rmse, lenT, phi1, phi2, psi1, sar1, coef)
    twoline <- c("", "", "", "",
                 ifelse(se_phi1 == "", "", paste0("(", se_phi1, ")")),
                 ifelse(se_phi2 == "", "", paste0("(", se_phi2, ")")),
                 ifelse(se_psi1 == "", "", paste0("(", se_psi1, ")")),
                 ifelse(se_sar1 == "", "", paste0("(", se_sar1, ")")),
                 paste0("(", se, ")"))
    lines <- rbind(oneline, twoline)
    rownames(lines) <- NULL

    return(lines)
}
```

```r
#AR(0) with covariates
ar0 <- armaFit(ts_house, order = c(0, 0, 0), xdf = covar)

print(xtable(ar0, digits = 3), booktabs = T, include.rownames = FALSE,
      sanitize.colnames.function=function(x){x})
```

Order	AIC	RMSE	T	ϕ_1	ϕ_2	ψ_1	Sϕ_1	PartisanMidterm	PartisanUnem	Coattails	Pre1994
(0,0,0)	239.243	13.999	28					-7.27	-2.053	18.396	47.994
								(3.823)	(1.733)	(5.331)	(5.703)

Part c

Across the 5 specifications, there is fairly robust support for the 1994 structural break and the importance of presidential coattails. The partisan unemployment theory receives pretty weak support, since the coefficient is uncertain through 4 out of 5 specifications. The 5 models are largely indistinguishable in terms of AIC, and both the RMSE and SE of the regression are all relatively close across models too. The ARMA(1, 1) performs best in-sample, though this is somewhat expected given the additional model parameters. Accordingly, the improvement in fit for including both autoregressive and moving-average term is weaker than would be expected if the two parameters were both necessary (the AR(2) is similar in this respect). All models miss the true values of the time-series they were trained on by about 14 or 15 seats.

```r
#AR(1)
ar1 <- armaFit(ts_house, order = c(1, 0, 0), xdf = covar)

#AR(2)
ar2 <- armaFit(ts_house, order = c(2, 0, 0), xdf = covar)

#MA(1)
ma1 <- armaFit(ts_house, order = c(0, 0, 1), xdf = covar)

#ARMA(1,1)
arma11 <- armaFit(ts_house, order = c(1, 0, 1), xdf = covar)

#bind the sum stat rows together
sums <- rbind(ar0, ar1, ar2, ma1, arma11)

print(xtable(sums, digits = 3), booktabs = T, include.rownames= FALSE,
      sanitize.colnames.function=function(x){x})
```

Order	AIC	RMSE	T	ϕ_1	ϕ_2	ψ_1	Sϕ_1	PartisanMidterm	PartisanUnem	Coattails	Pre1994
(0,0,0)	239.243	13.999	28					-7.27	-2.053	18.396	47.994
								(3.823)	(1.733)	(5.331)	(5.703)
(1,0,0)	240.221	13.732	28	0.234				-8.848	-2.42	15.364	46.656
				(0.223)				(3.86)	(1.75)	(5.828)	(7.023)
(2,0,0)	239.662	13.034	28	0.448	-0.377			-10.726	-2.858	10.279	44.729
				(0.238)	(0.215)			(3.158)	(1.744)	(5.862)	(6.038)
(0,0,1)	239.495	13.529	28			0.389		-9.866	-2.761	13.121	45.386
						(0.3)		(3.743)	(1.841)	(6.315)	(7.58)
(1,0,1)	238.096	12.189	28	-0.382		1		-11.673	-3.846	14.522	42.887
				(0.254)		(0.108)		(3.204)	(1.484)	(6.058)	(6.623)

Part d

The best model given the in-sample and out-of-sample goodness of fit indicators is the AR(1). The in-sample indicators would suggest (somewhat weakly) that the ARMA(1,1) is the best model, however this more complicated model actually predicts

worse out-of-sample compared to the simpler AR(1) specification. Given that the two fits between AR(1) and ARMA(1,1) are not all that different, I'd opt for the more parsimonious model. The ARMA(1,1)'s psi of 1.0 is suspect too, though I'm less certain that this is a fatal flaw so much as a sign of odd fit.

```r
#ar0
f_ar0 <- function(x, h){forecast(Arima(x, order=c(0,0,0)), h=h)}
e_ar0 <- tsCV(ts_house, f_ar0, h = 3,
              window = 20)
mae_ar0 <- round(apply(e_ar0, 2, function(x){mean(abs(ts_house - x), na.rm = T)}), 3)
avg_mae_ar0 <- round(mean(mae_ar0), 3)
names(avg_mae_ar0) <- "avgMAE"
e_ar0 <- c(ar0[1, 1], ar0[1, 2], ar0[1, 3], mae_ar0, avg_mae_ar0)

#ar1
f_ar1 <- function(x, h){forecast(Arima(x, order=c(1,0,0)), h=h)}
e_ar1 <- tsCV(ts_house, f_ar1, h = 3,
              window = 20)
mae_ar1 <- round(apply(e_ar1, 2, function(x){mean(abs(ts_house - x), na.rm = T)}), 3)
avg_mae_ar1 <- round(mean(mae_ar1), 3)
names(avg_mae_ar1) <- "avgMAE"
e_ar1 <- c(ar1[1, 1], ar1[1, 2], ar1[1, 3], mae_ar1, avg_mae_ar1)

#ar2
f_ar2 <- function(x, h){forecast(Arima(x, order=c(2,0,0)), h=h)}
e_ar2 <- tsCV(ts_house, f_ar2, h = 3,
              window = 20)
mae_ar2 <- round(apply(e_ar2, 2, function(x){mean(abs(ts_house - x), na.rm = T)}), 3)
avg_mae_ar2 <- round(mean(mae_ar2), 3)
names(avg_mae_ar2) <- "avgMAE"
e_ar2 <- c(ar2[1, 1], ar2[1, 2], ar2[1, 3], mae_ar2, avg_mae_ar2)

#ma1
f_ma1 <- function(x, h){forecast(Arima(x, order=c(0,0,1)), h=h)}
e_ma1 <- tsCV(ts_house, f_ma1, h = 3,
              window = 20)
mae_ma1 <- round(apply(e_ma1, 2, function(x){mean(abs(ts_house - x), na.rm = T)}), 3)
avg_mae_ma1 <- round(mean(mae_ma1), 3)
names(avg_mae_ma1) <- "avgMAE"
e_ma1 <- c(ma1[1, 1], ma1[1, 2], ma1[1, 3], mae_ma1, avg_mae_ma1)

#arma1,1
f_arma11 <- function(x, h){forecast(Arima(x, order=c(1,0,1)), h=h)}
e_arma11 <- tsCV(ts_house, f_arma11, h = 3,
              window = 20)
mae_arma11 <- round(apply(e_arma11, 2, function(x){mean(abs(ts_house - x), na.rm = T)}), 3)
avg_mae_arma11 <- round(mean(mae_arma11), 3)
names(avg_mae_arma11) <- "avgMAE"
e_arma11 <- c(arma11[1, 1], arma11[1, 2], arma11[1, 3], mae_arma11, avg_mae_arma11)

#compile model fit stats
forecastMAE <- rbind(e_ar0, e_ar1, e_ar2, e_ma1, e_arma11)

print(xtable(forecastMAE), booktabs = T, include.rownames=FALSE)
```

Order	AIC	RMSE	h=1	h=2	h=3	avgMAE
(0,0,0)	239.243	13.999	30.975	41.386	42.208	38.19
(1,0,0)	240.221	13.732	26.078	39.36	37.721	34.386
(2,0,0)	239.662	13.034	26.944	39.481	37.505	34.643
(0,0,1)	239.495	13.529	30.738	41.285	41.653	37.892
(1,0,1)	238.096	12.189	27.598	39.671	37.665	34.978

Part e

The coattails effect is what the scenarios highlight more than anything—i.e. assuming that the Democrats beat Trump in 2020, they will see the largest gains in their share of House seats relative to the unemployment-related differences. This boost due to coattails will however be approximately halved by a partisan midterm wave. We should have the most certainty about the forecast one election ahead, since this is not adding in forecast error due to multiple forecasts.

```r
#selecting AR(1) as final model
ar1 <- Arima(ts_house, order = c(1, 0, 0), xreg = covar)

#counterfactual 1 - unemployment stays at 4.6% for all three elections
cf1 <- data.frame(
  "Pre1994" = rep(0, 3), #all forecasts are post1994
  "ParisanUnemp" = rep(4.6-6.075, 3),
  "PartisanMidterm" = c(-1, 0, 1),
  "Coattails" = c(0, 1, 0)
)
cf1
```

```
##   Pre1994 ParisanUnemp PartisanMidterm Coattails
## 1       0      -1.475              -1         0
## 2       0      -1.475               0         1
## 3       0      -1.475               1         0
```

```r
#forecast forward three periods based on cf1 X's
pred_cf1 <- predict(ar1, newxreg = cf1)

##counterfactual 2 - unemployment falls to 3.6% for all three elections
cf2 <- data.frame(
  "Pre1994" = rep(0, 3), #all forecasts are post1994
  "ParisanUnemp" = rep(3.6-6.075, 3),
  "PartisanMidterm" = c(-1, 0, 1),
  "Coattails" = c(0, 1, 0)
)
cf2
```

```
##   Pre1994 ParisanUnemp PartisanMidterm Coattails
## 1       0      -2.475              -1         0
## 2       0      -2.475               0         1
## 3       0      -2.475               1         0
```

```r
#forecast forward three periods based on cf2 X's
pred_cf2 <- predict(ar1, newxreg = cf2)

##counterfactual 2 - unemployment rises to 5.6% for all three elections
cf3 <- data.frame(
  "Pre1994" = rep(0, 3), #all forecasts are post1994
  "ParisanUnemp" = rep(5.6-6.075, 3),
  "PartisanMidterm" = c(-1, 0, 1),
  "Coattails" = c(0, 1, 0)
```

```
)
cf3
```

```
##   Pre1994 ParisanUnemp PartisanMidterm Coattails
## 1       0       -0.475              -1         0
## 2       0       -0.475               0         1
## 3       0       -0.475               1         0
```

```r
#forecast forward three periods based on cf2 X's
pred_cf3 <- predict(ar1, newxreg = cf3)

#construct tidy matrix for value, upper and lower
pred_vals <- data.frame(
  cf = c(rep("Scenario 1", 3), rep("Scenario 2", 3), rep("Scenario 3", 3)),
  time = rep(c(2019, 2021, 2023), 3),
  values = c(pred_cf1$pred, pred_cf2$pred, pred_cf3$pred),
  upper = c(pred_cf1$pred[1]+1.96*pred_cf1$se,
            pred_cf1$pred[2]+1.96*pred_cf1$se,
            pred_cf1$pred[3]+1.96*pred_cf1$se,
            pred_cf2$pred[1]+1.96*pred_cf2$se,
            pred_cf2$pred[2]+1.96*pred_cf2$se,
            pred_cf2$pred[3]+1.96*pred_cf2$se,
            pred_cf3$pred[1]+1.96*pred_cf3$se,
            pred_cf3$pred[2]+1.96*pred_cf3$se,
            pred_cf3$pred[3]+1.96*pred_cf3$se),
  lower = c(pred_cf1$pred[1]-1.96*pred_cf1$se,
            pred_cf1$pred[2]-1.96*pred_cf1$se,
            pred_cf1$pred[3]-1.96*pred_cf1$se,
            pred_cf2$pred[1]-1.96*pred_cf2$se,
            pred_cf2$pred[2]-1.96*pred_cf2$se,
            pred_cf2$pred[3]-1.96*pred_cf2$se,
            pred_cf3$pred[1]-1.96*pred_cf3$se,
            pred_cf3$pred[2]-1.96*pred_cf3$se,
            pred_cf3$pred[3]-1.96*pred_cf3$se))

ggplot() +
  geom_line(data=pred_vals, aes(x = time, y = values,
                                group = cf, color = cf)) +
  geom_point(data = pred_vals, aes(x = time, y = values,
                                   group = cf, shape = cf, color = cf),
             size = 3) +
  geom_ribbon(data = pred_vals, aes(x = time, ymin = lower, ymax = upper,
                                    group = cf, fill = cf),
              color = NA, alpha = .25) +
  geom_line(data = congress %>% filter(StartYear >= 2000), aes(x = StartYear, y = DemHouseMaj),
            color = "black") +
  scale_x_continuous() +
  xlab("\nStart Year of Congress") +
  ylab("Predicted House Dem Share\n") +
  labs(subtitle = "Shaded area denotes 95% predictive interval",
       fill = "Counterfactual\nForecast",
       color = "Counterfactual\nForecast",
       shape = "Counterfactual\nForecast") +
  theme_minimal()
```

Shaded area denotes 95% predictive interval
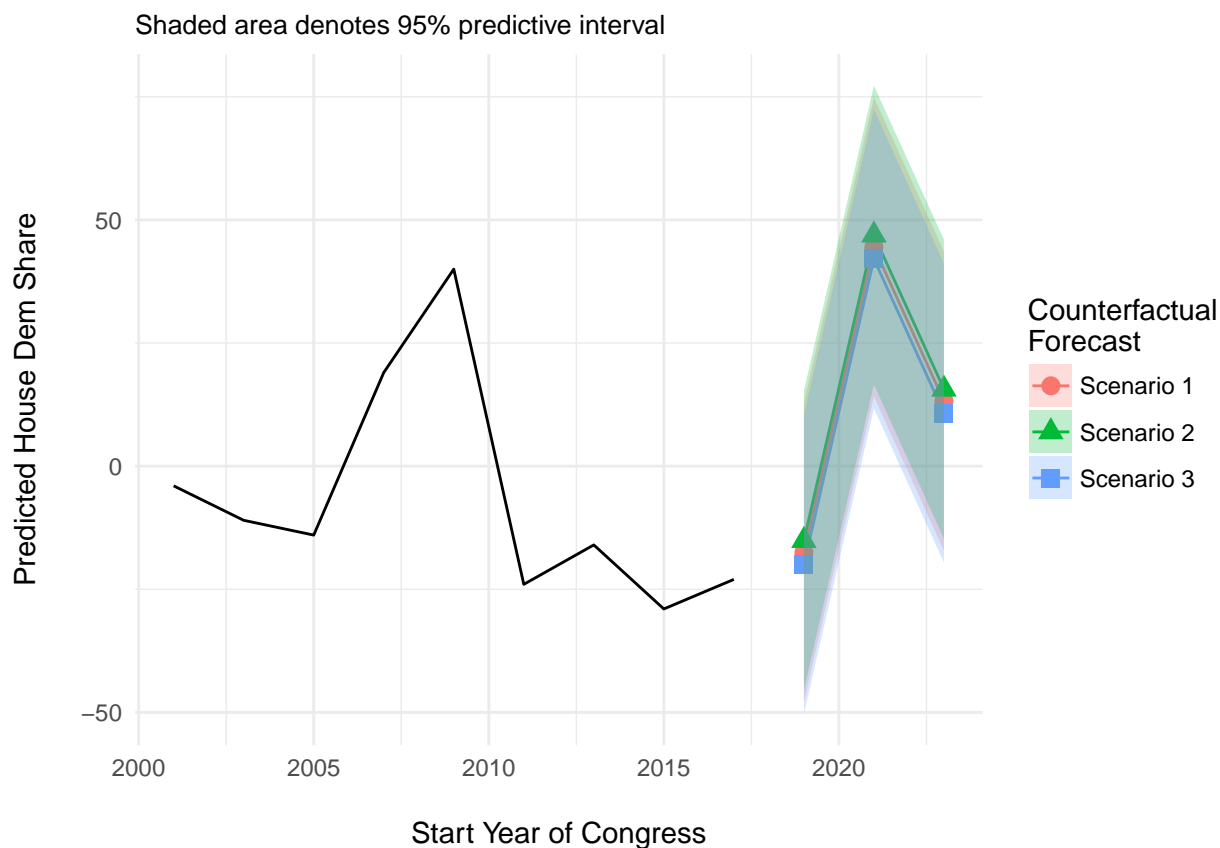
## Problem 2 - U.S. Senate

Part a

The observed time-series has similar evidence of the 1994 structural break, and otherwise the ACF and PACF does show evidence suggestive of an AR(1) process. The PP test and Dickey-Fuller tests both indicate non-stationarity in the observed time-series, though demeaning the data according to pre/post 1994 periods creates a time-series that looks fairly stationary. Both the ACF and PACF suggest autocorrelation that an AR(1) process with phi of .6 would capture well.

```r
#plot ts, ACF, PACF
#run ADF and PP tests
#demean by pre/post 1994 periods
#plot ts, ACF, PACF
#what effect does 1994 structural break have on ts?

#pull ts, every other year frequency 1963-2017
ts_senate <- ts(congress$DemSenateMaj,
                frequency = 1/2,
                start = 1963, end = 2017)

#plot as observed
autoplot(ts_senate) +
  geom_hline(yintercept = 0, linetype = 3, color = "grey60") +
  geom_vline(xintercept = 1994, color = "Blue") +
  theme_minimal() +
  labs(title = "Democratic Senate Majority Time-series")
```
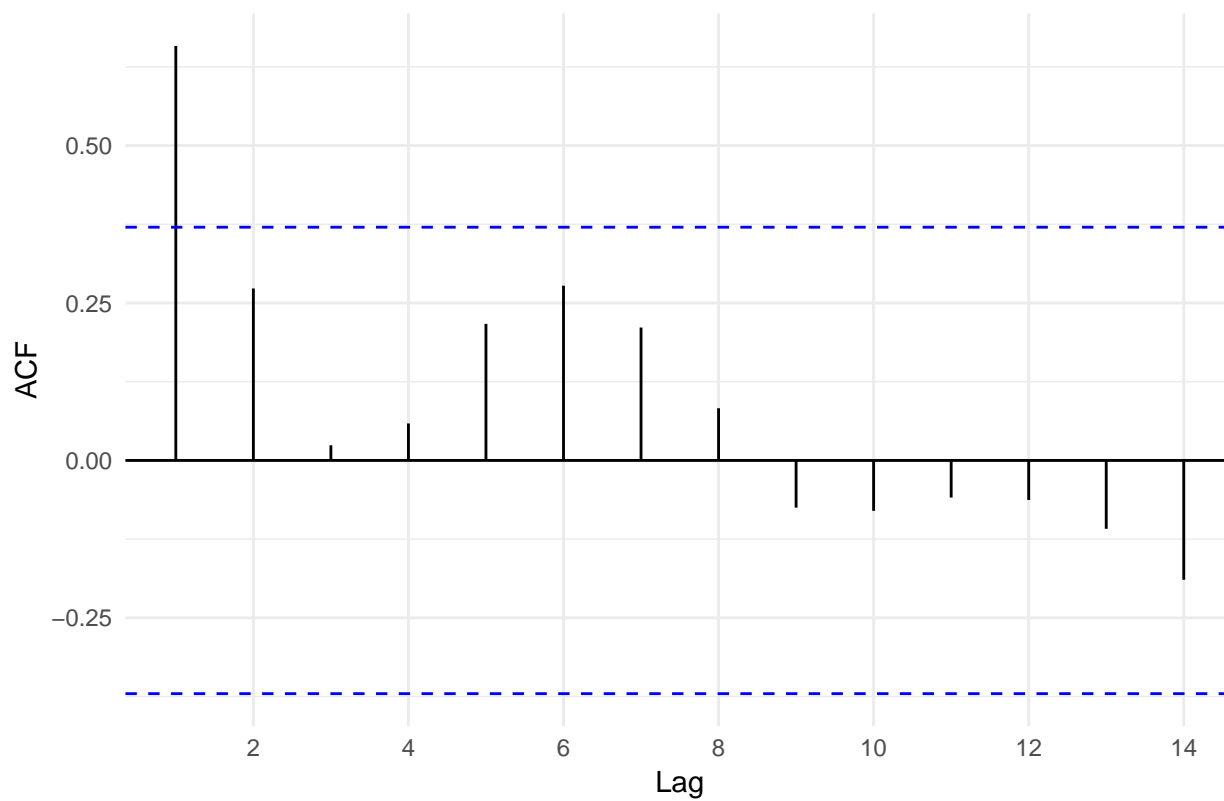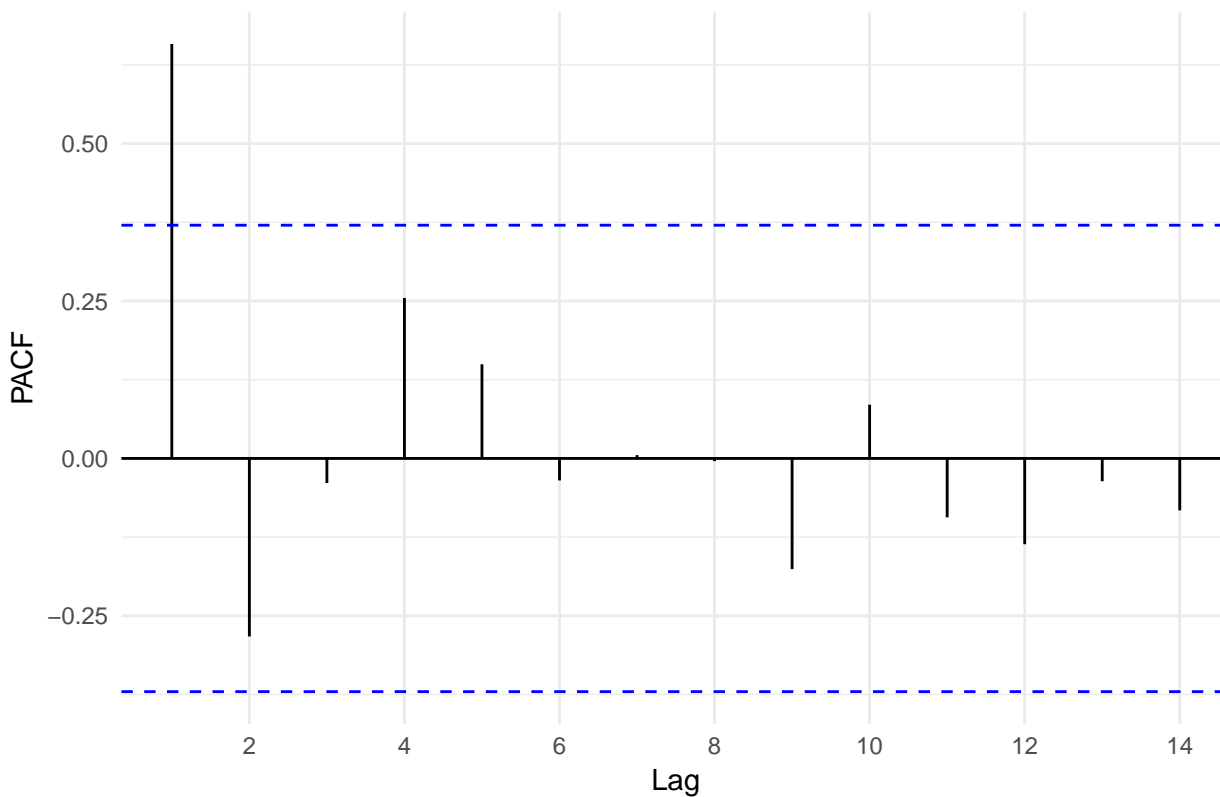
Democratic Senate Majority Time−series
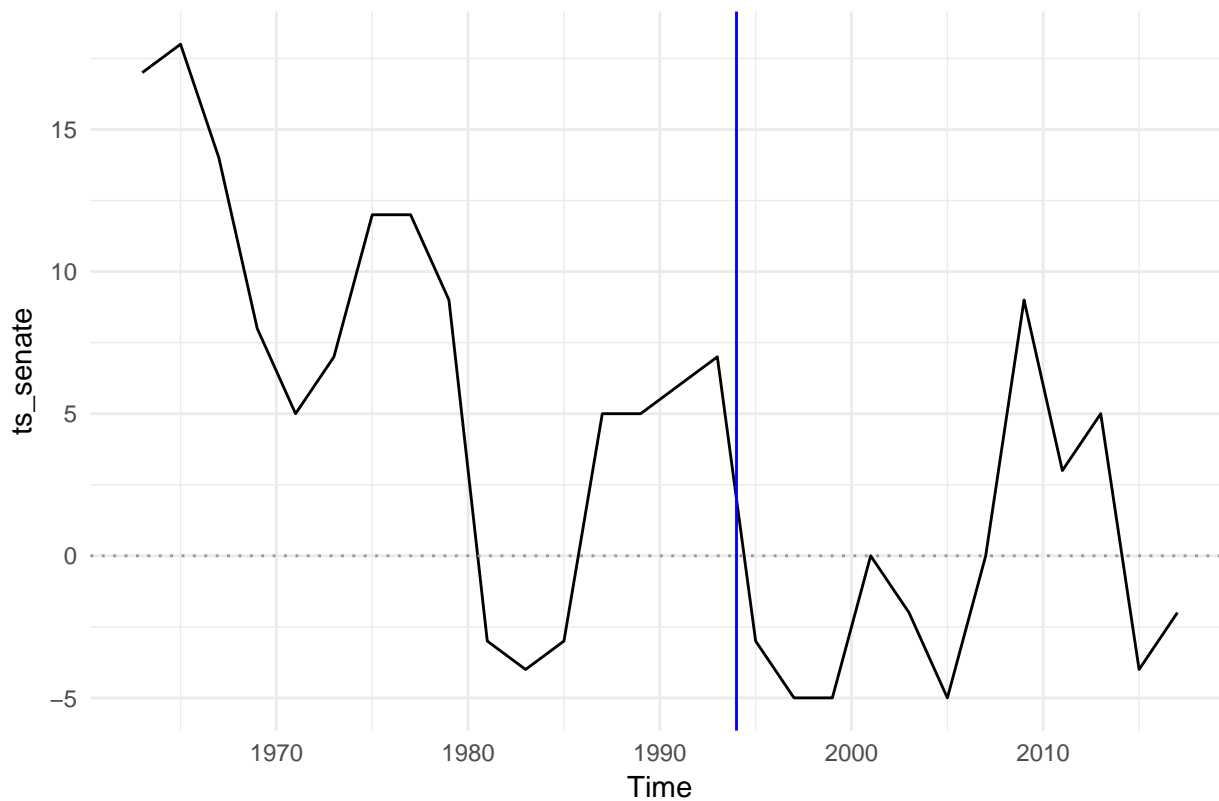
```r
#observed ACF
ggAcf(ts_senate) + #could be AR(1)
  theme_minimal() +
  labs(title = "Democratic Senate Majority ACF")
```

Democratic Senate Majority ACF


```r
#observed PACF
ggPacf(ts_senate) + #AR(1), phi = .65 ish
  theme_minimal() +
  labs(title = "Democratic Senate Majority PACF")
```

Democratic Senate Majority PACF


```r
PP.test(ts_senate)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  ts_senate
## Dickey-Fuller = -2.8296, Truncation lag parameter = 2, p-value =
## 0.2552
```

```r
tseries::adf.test(ts_senate)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_senate
## Dickey-Fuller = -2.1751, Lag order = 3, p-value = 0.5055
## alternative hypothesis: stationary
```

```r
#pre-1994 mean for ts
preMean <- congress %>%
  filter(StartYear < 1994) %>%
  summarize(mean = mean(DemSenateMaj)) %>%
  pull(mean)

#post-1994 mean for ts
postMean <- congress %>%
  filter(StartYear >= 1994) %>%
  summarize(mean = mean(DemSenateMaj)) %>%
  pull(mean)

#demean based on pre/post 1994 (i.e. 1-16th obs vs 17-28th obs)
```
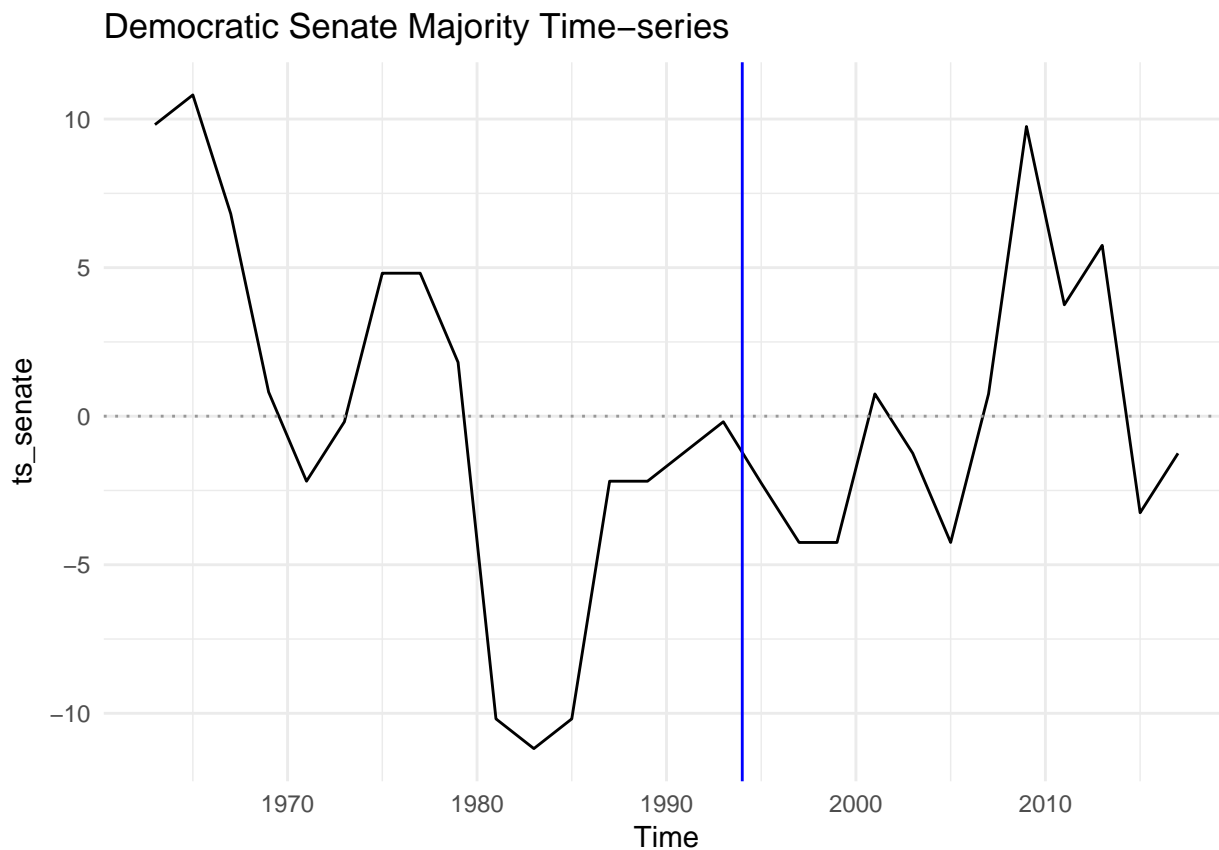
```
ts_senate[1:16] <- ts_senate[1:16] - preMean
ts_senate[-1:-16] <- ts_senate[-1:-16] - postMean

#demeaned ts - looks stationary now
autoplot(ts_senate) +
  geom_hline(yintercept = 0, linetype = 3, color = "grey60") +
  geom_vline(xintercept = 1994, color = "Blue") +
  theme_minimal() +
  labs(title = "Democratic Senate Majority Time-series")
```



```
#demeaned ACF
ggAcf(ts_senate) + #still shows autocorrelation
  theme_minimal() +
  labs(title = "Democratic Senate Majority ACF")
```

Democratic Senate Majority ACF
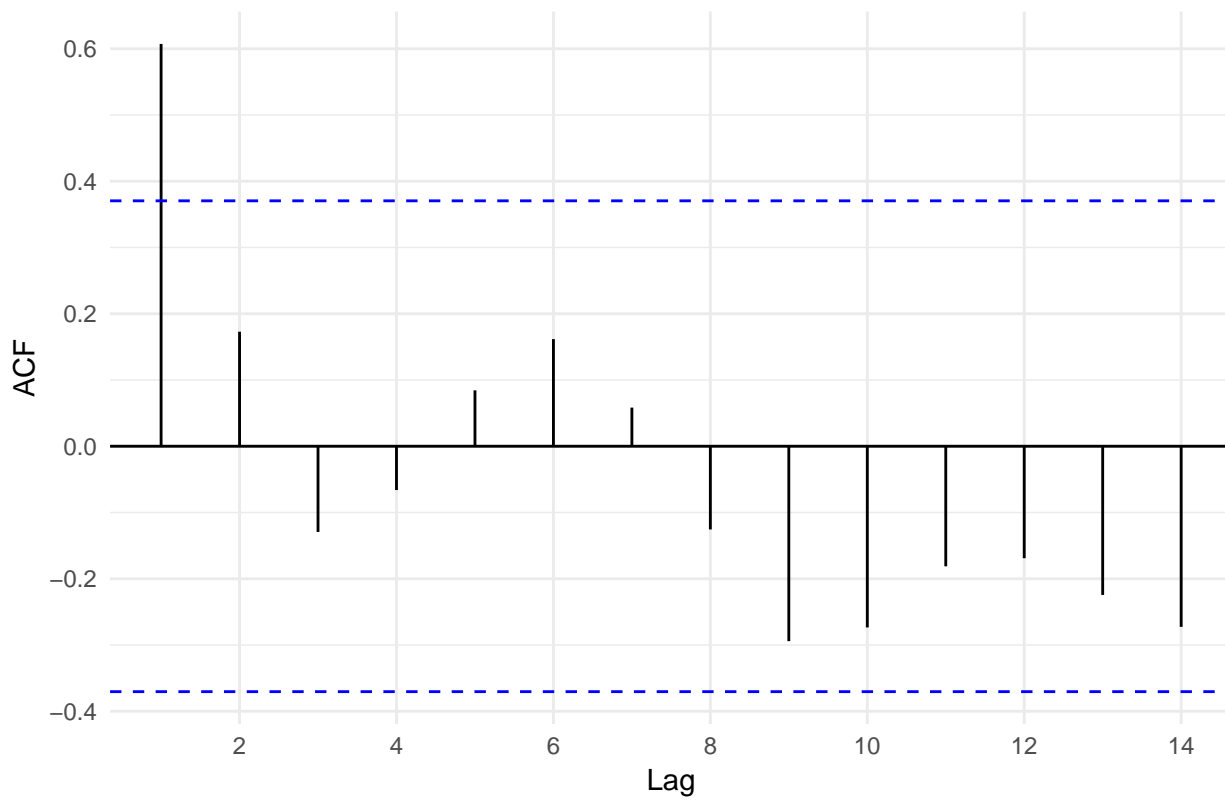

```
#demeaned PACF
ggPacf(ts_senate) + #need an AR(1) with phi = .6 for this ts
  theme_minimal() +
  labs(title = "Democratic Senate Majority PACF")
```
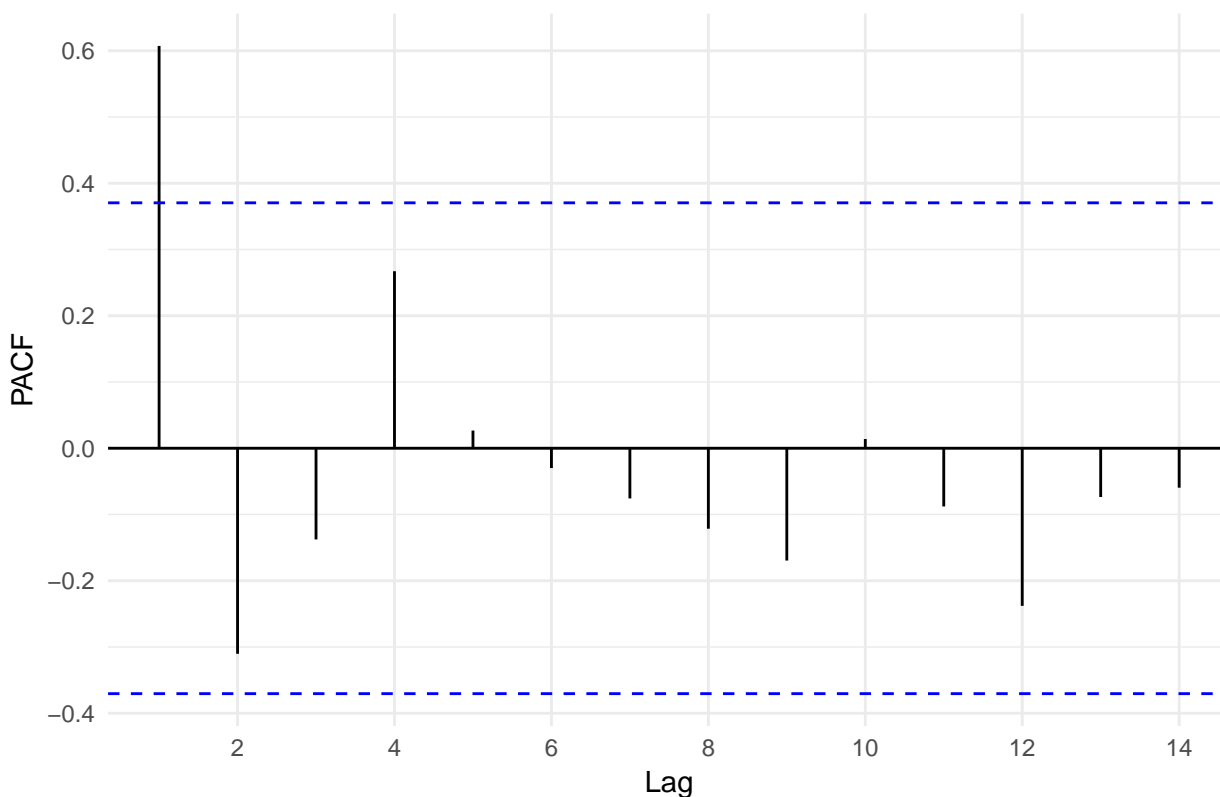
Democratic Senate Majority PACF

Part b

The best fitting model using AIC and FMSE is the ARMA(1, 1) since this achieves the lowest average model error and is comparable to the MA(2) in terms of AIC. The table of coefficients for the U.S. Senate majority time-series suggests less salience to the coattails theory for the upper chamber of Congress. Instead, the only robust features predicting substantive differences in the Democratic Senate majority over the time-series are the unemployment rate and the structural break for 1994. The evidence for Partisan midterm swings is fairly weak.

The overall takeaways relative to the House models are that the direction the presidency goes is less predictive for senate races and that there is some significance to higher than average unemployment levels as predicting worse outcomes for Democrats.

```r
#pull ts, every other year frequency 1963-2017
ts_senate <- ts(congress$DemSenateMaj,
                frequency = 1/2,
                start = 1963, end = 2017)

#create df of covariates
covar <- congress %>% select(PartisanMidterm, PartisanUnem, Coattails, Pre1994)

#AR(0) with covariates
ar0 <- armaFit(ts_senate, order = c(0, 0, 0), xdf = covar)

#AR(1)
ar1 <- armaFit(ts_senate, order = c(1, 0, 0), xdf = covar)
tab.ar1 <- xtable(ar1)

#AR(2)
ar2 <- armaFit(ts_senate, order = c(2, 0, 0), xdf = covar)
tab.ar2 <- xtable(ar2)
```

```r
#MA(1)
ma1 <- armaFit(ts_senate, order = c(0, 0, 1), xdf = covar)
tab.ma1 <- xtable(ma1)

#ARMA(1,1)
arma11 <- armaFit(ts_senate, order = c(1, 0, 1), xdf = covar)
tab.arma11 <- xtable(arma11)

#bind the sum stat rows together
sums <- rbind(ar0, ar1, ar2, ma1, arma11)

print(xtable(sums), booktabs = T, include.rownames = FALSE,
      sanitize.colnames.function=function(x){x})
```

Order	AIC	RMSE	T	ϕ_1	ϕ_2	ψ_1	$S\phi_1$	PartisanMidterm	PartisanUnem	Coattails	Pre1994
(0,0,0)	183.044	5.132	28					1.792	0.358	3.403	8.503
								(1.402)	(0.635)	(1.954)	(2.091)
(1,0,0)	172.3	4.039	28	0.696				-0.341	-0.268	1.892	8.151
				(0.145)				(0.945)	(0.459)	(1.392)	(3.507)
(2,0,0)	168.047	3.563	28	1.087	-0.565			-0.806	-0.884	0.103	6.611
				(0.16)	(0.174)			(0.635)	(0.367)	(1.091)	(2.581)
(0,0,1)	170.826	3.749	28			1		-0.464	-1.224	1.784	7.377
						(0.122)		(0.767)	(0.339)	(1.467)	(2.204)
(1,0,1)	168.501	3.413	28	0.44		1		-0.72	-1.35	0.807	9.131
				(0.187)		(0.148)		(0.576)	(0.24)	(1.134)	(2.171)

Part c

The AR(1,)(1)$_3$ model does not perform as well as the ARMA(1,1) in terms of overall fit, but does find a fairly precise cycle for every three periods that coincides with the three classes of senators and their respective order of elections. The negative phi for multiplicative seasonality suggests that historical levels of Democratic Senate majority for first-class, second-class and third-class Senate elections predict a 50% inverse for the respective elections in the next cycle (holding the other covariates constant). For instance, if a first-class senator is elected at a high point of Democratic Senate Majority, the next time this class is up again the effect of seasonal history will be to bring the predictions back towards 0 (i.e. 50-50 Dem:Rep).

```r
ar1_sar1 <- armaFit(ts_senate,
                order = c(1, 0, 0),
                seasonal.order = c(1, 0, 0),
                seasonal.period = 3,
                xdf = covar)
ar1_sar1[1,1] <- "(1, 0, 0)(1, 0, 0)"
sums <- rbind(ar0, ar1, ar2, ma1, arma11, ar1_sar1)
```

Order	AIC	RMSE	T	ϕ_1	ϕ_2	ψ_1	$S\phi_1$	PartisanMidterm	PartisanUnem	Coattails	Pre1994
(0,0,0)	183.044	5.132	28					1.792	0.358	3.403	8.503
								(1.402)	(0.635)	(1.954)	(2.091)
(1,0,0)	172.3	4.039	28	0.696				-0.341	-0.268	1.892	8.151
				(0.145)				(0.945)	(0.459)	(1.392)	(3.507)
(2,0,0)	168.047	3.563	28	1.087	-0.565			-0.806	-0.884	0.103	6.611
				(0.16)	(0.174)			(0.635)	(0.367)	(1.091)	(2.581)
(0,0,1)	170.826	3.749	28			1		-0.464	-1.224	1.784	7.377
						(0.122)		(0.767)	(0.339)	(1.467)	(2.204)
(1,0,1)	168.501	3.413	28	0.44		1		-0.72	-1.35	0.807	9.131
				(0.187)		(0.148)		(0.576)	(0.24)	(1.134)	(2.171)
(1, 0, 0)(1, 0, 0)	169.047	3.622	28	0.818			-0.51	-0.494	-0.301	0.728	6.887
				(0.12)			(0.185)	(0.641)	(0.388)	(1.029)	(3.045)