STAT 331 FINAL PROJECT: CASE STUDY

Strike Activity in Organization for Economic Co-operation and Development during the postwar period (1951-1985)

University of Waterloo
Faculty of Mathematics

Instructor: Martin Lysy

Prepared by Group 24:

Xitao Zhou	*20549853*
Yuanjing Cai	*20549267*
Yidan Chen	*20568620*

1. Summary

The goal of this project is to explore the relation between strike activity and some explanatory macroeconomic variables and find a linear model which fits the data. The given data provides the information about strike activities in 18 countries belonging to the Organization for Economic Co-operation and Development during the postwar period, and 7 possible factors which may have influence on strike activity. In this project, the strike activity is expressed by the number of days in the year lost per 1000 workers due to strikes, the variables *Unemp* and *Infl* indicates the unemployment rate (%) and inflation rate (%) respectively. The democracy, denoted as Demo, is expressed as the proportion of left-party parliamentary representation. The variable *Centr* and *Dens* indicate the measure of union centralization and trade union density respectively. After we checked collinearity and applied variable transformation and automated model selection, two candidate models are selected for further comparison. After several model diagnostics assessments, the model which gives the best interpretation of the relationship between strike activity and significant factors is selected as our final result.

2. Model selection

2.1 Multi-linearity

```
> alias(lm(Strike~.-1,data=strikes))
Strike ~ (Country + Year + Unemp + Infl + Demo + Centr + Dens) - 1
Complete :
      CountryAustralia  CountryAustria CountryBelgium CountryCanada CountryDenmark
Centr       3/8               1            3/4              0            1/2
      CountryFinland  CountryFrance CountryGermany CountryIreland CountryItaly CountryJapan
Centr       3/4             0            1/4            1/2          1/4        1/8
      CountryNetherlands CountryNewZealand CountryNorway CountrySweden CountrySwitzerland
Centr       3/4               3/8             7/8            7/8            1/2
      CountryUnitedKingdom CountryUnitedStates Year Unemp Infl Demo Dens
Centr       3/8                  0               0    0     0    0    0
```

Before applying auto-selection, the multicollinearity between explanatory variables is checked and It turns out that *Centr* is completely linearly dependent on Country. The variable Country may contain more information than *Centr* by common sense. Therefore, *Centr* is removed from the model.

2.2 Variable Transformation

Firstly, we obtained 2 different models using the three automated model selection methods. The residual plots of the additive models have a fanning-out pattern as the predicted values become larger. So we make an assumption that log-transformation may solve this problem. Since Strike variable may be zero, instead of taking log on Strike directly, taking log on (strike+1) can avoid getting NA's or infinities. After the log transformation, the residual plots seem much more random. We display the residual plots of M1 before and after log transformation, where M1 is the result we get from forward model selection:

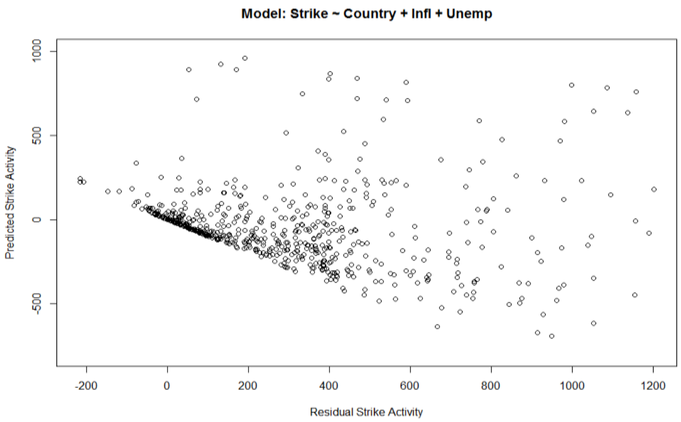


Figure 2.2.1 Residual plot of Model: Strike ~ Country+Infl+Unemp

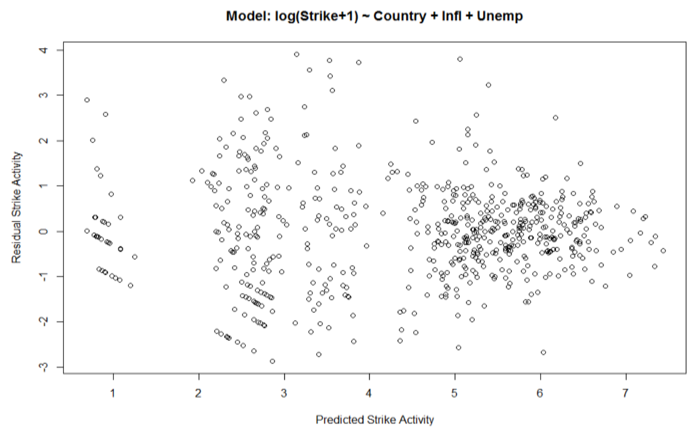


Figure 2.2.2 Residual plot of Model: log(Strike+1)~ Country+Infl+Unemp

2.3 Auto-Selection

For automated model selection, two scenarios are considered, whether *Country* is included in the interaction effects or not.

2.3.1 *Country* not included in interaction effects

A base model and a full model are selected for automated model selection, where *Country* is excluded in interaction terms, and *Centr* is removed to avoid collinearity

-Base model: *log(Strike+1)~1*

-Full model: *log(Strike+1)~(.-Centr-Country)^2+Country*

The models returned by forward, backward and stepwise model selection are listed as follows:

-Forward: *log(Strike + 1) ~ Country + Dens + Infl + Year + Infl : Year + Dens : Infl*

-Backward: *log(Strike + 1) ~ Year + Unemp + Infl + Dens + Country + Year : Unemp + Year : Infl + Unemp : Infl + Infl : Dens*

-Stepwise: *log(Strike + 1) ~ Year + Infl + Unemp + Dens + Country + Year:Unemp + Year:Infl + Infl:Unemp + Infl:Dens*

The backward and stepwise selection returns the same model. Since it has larger adjusted R-squared value more significant variables than the model obtained by forward selection, we choose the model obtained by backward(stepwise) selection to be our candidate for further examination, which we refer to as

M1: Year + Unemp + Infl + Dens + Country + Year : Unemp + Year : Infl + Unemp : Infl + Infl : Dens

2.3.2 Country included in interaction effects

A base model and a full model are selected for automated model selection, where *Country* is included in interaction terms, and *Centr* is removed to avoid collinearity.

-Base model: *log (Strike+1)~1*

-Full model: *log (Strike+1)~(.-Centr)^2*

The models returned by forward, backward and stepwise model selection are listed as follows:

-Forward: *log (Strike + 1) ~ Country + Dens + Infl + Year + Country : Year + Country : Infl + Country : Dens*

-Backward: *log (Strike + 1) ~ Country + Dens + Infl + Year + Unemp + Country : Year + Country : Unemp + Year : Dens + Unemp : Dens*

-Stepwise: *log (Strike + 1) ~ Country + Dens + Infl + Year + Unemp + Country : Year + Country : Dens + Infl : Unemp*

The three auto-selection procedures produce different results. Backward selection method is known as a better method compared with forward selection. Moreover, the adjusted R-squared is the highest among the three results, indicating stronger predictive power. So we choose the model obtained by backward selection as another candidate model for further examination, denoted as *M2: log (Strike + 1) ~ Country + Dens + Infl + Year + Unemp + Country:Year + Country:Unemp + Year :Dens +Unemp: Dens*

3. Model Diagnostics

In the model selection step, we obtained two competing models M1 and M2. In this section, we will take a closer inspection on the two models, where leverages and influences, residual plots, and cross-validation results are examined and compared. A model that performs better based on those criteria is selected as our final model.

3.1 Leverage and Cook's Distance

To avoid the inaccurate estimation caused by uncertain data, leverages and Cook's Distance would be of great use to diagnose potential outliers or influential observations.
We draw the plot of Cook's Distance against Leverage for each model to examine the leverage and influence of the points based on them. The points of which the leverage is more than twice the average of the leverage are marked in blue as "high leverage points", and the point with the largest Cook's Distance is marked in red as the most influential point. In the plots, it can be observed that the points in *Figure 3.1.1* are more pushed to the bottom left corner (i.e. points have lower leverage in general), and there are much less high leverage points than in *Figure 3.1.2*. The most influential points of both plots correspond to the same observation, but it has more influence in M1 than in M2. From the Residuals vs Leverages plots, we also see that in *Figure 3.1.4*, there are more points with DIFFTS far away from other residuals, and the residuals are more spread out along the horizontal line. Therefore, both types of plots suggest that M1 has fewer outliers and are less affected by high leverage and influential points, so it is a model that better describes the behavior of most points in the dataset.

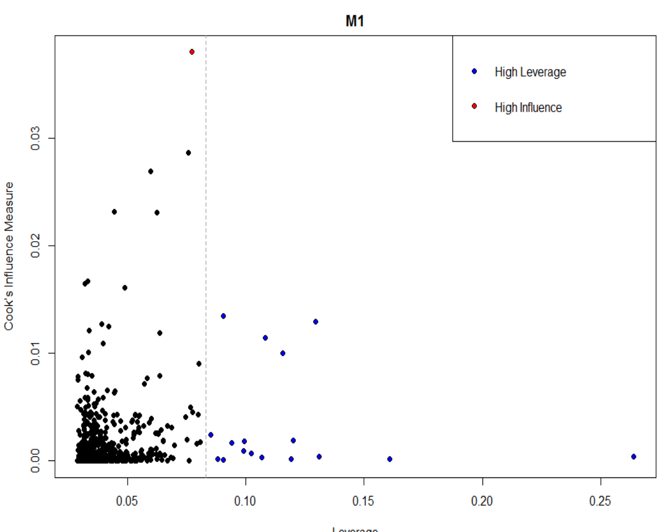


Figure 3.1.1 Cook's influence measure vs. Leverage(M1)

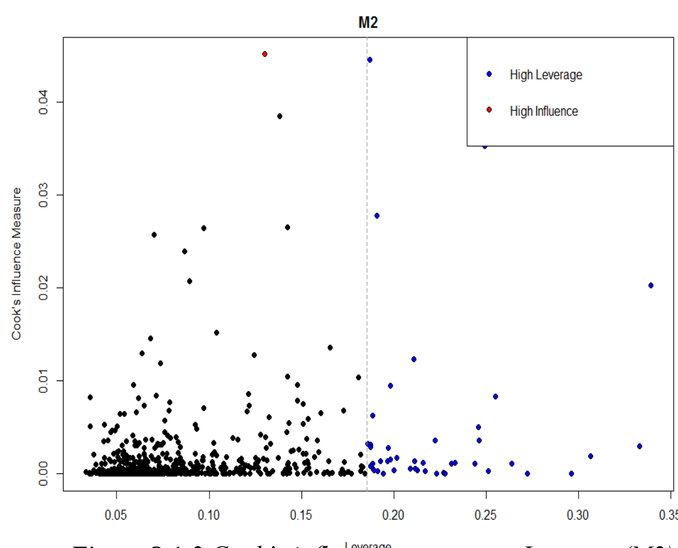
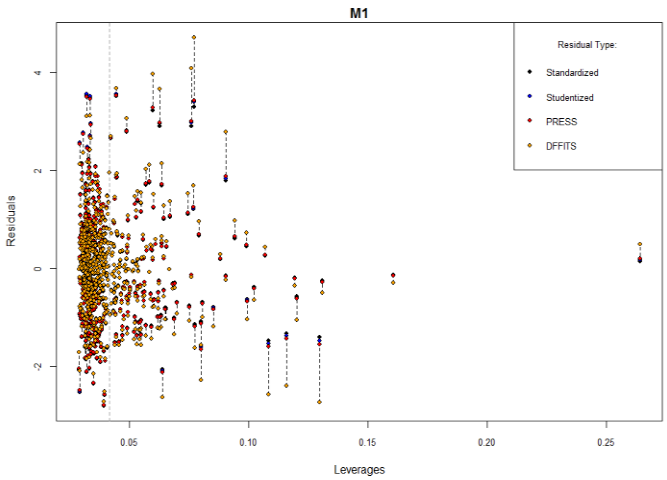


Figure 3.1.2 Cook's influence measure vs Leverage(M2)



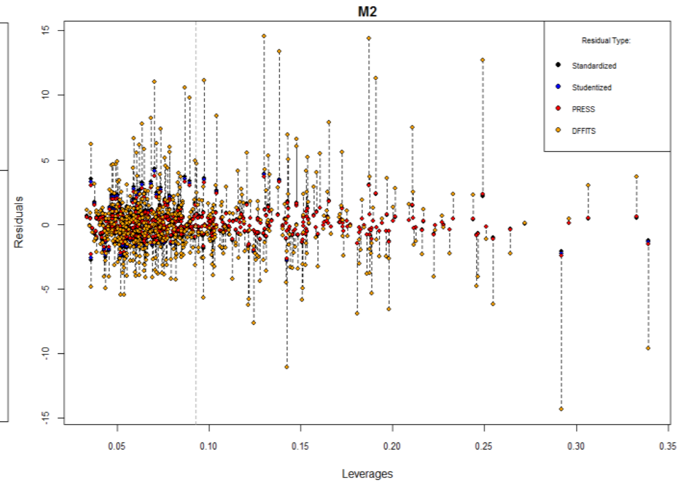
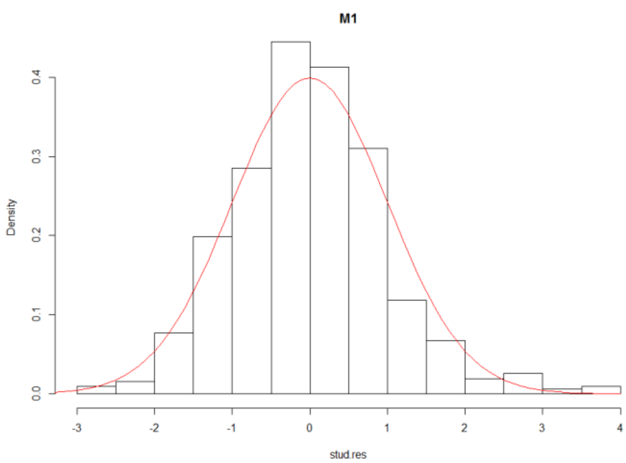


Figure 3.1.3 Residuals vs. Leverage(M1)

Figure 3.1.4 Residuals vs. Leverage(M2)

3.2 Residual plots

Different types of residual plots are helpful instrument to explore the relationship between the model and normal distribution. The model which fits normal distribution better may be able to give more precise and regular estimation and prediction for the data. One important assumption of linear regression is that the residuals form a standard Normal distribution. We have drawn different types of residual plots to check the Normality. *Figure 3.2.1*, *Figure 3.2.2*, *Figure 3.2.3* and *Figure 3.2.4* show that the residuals of M1 better form a standard Normal distribution. *Figure 3.2.5 and Figure 3.2.6* indicate that the residuals of M1 are more randomly and evenly distributed on two sides of the horizontal line y=0. Therefore, M1 better meets the Normality assumption of the residuals.



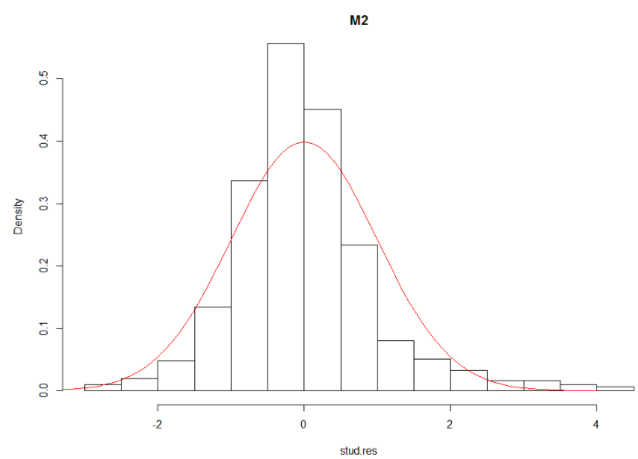


Figure 3.2.1 Density vs. Studentized Residuals(M1)

Figure 3.2.2 Density vs. Studentized Residuals(M2)

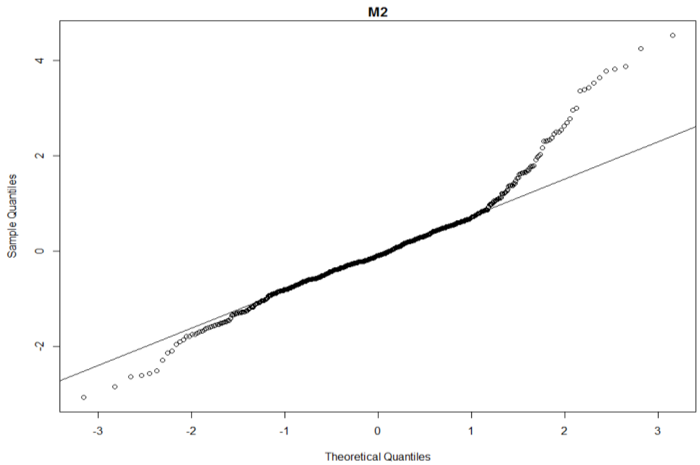


Figure 3.2.3 Q-Q Plot (M1)

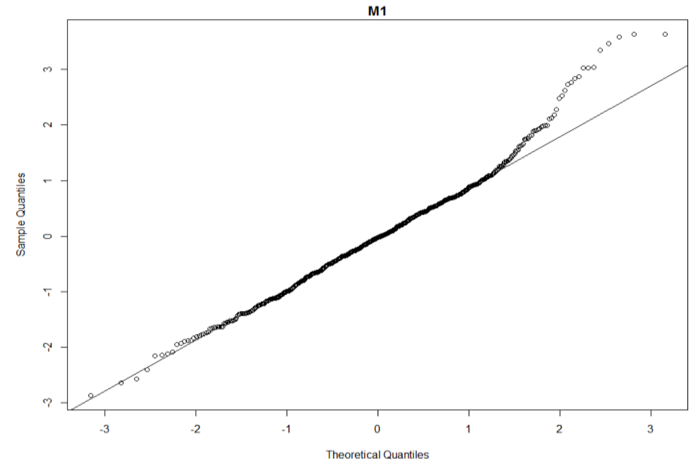


Figure 3.2.4 Q-Q Plot (M2)

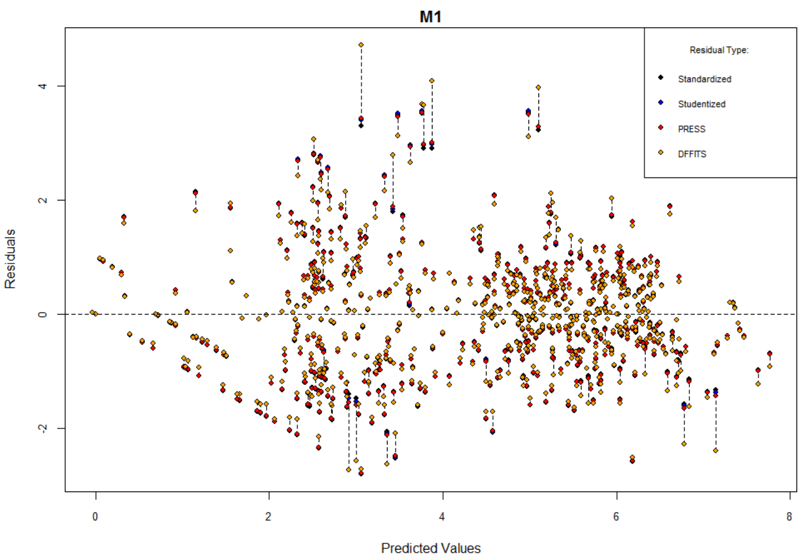


Figure 3.2.5 Residuals vs. Predicted Values(M1)

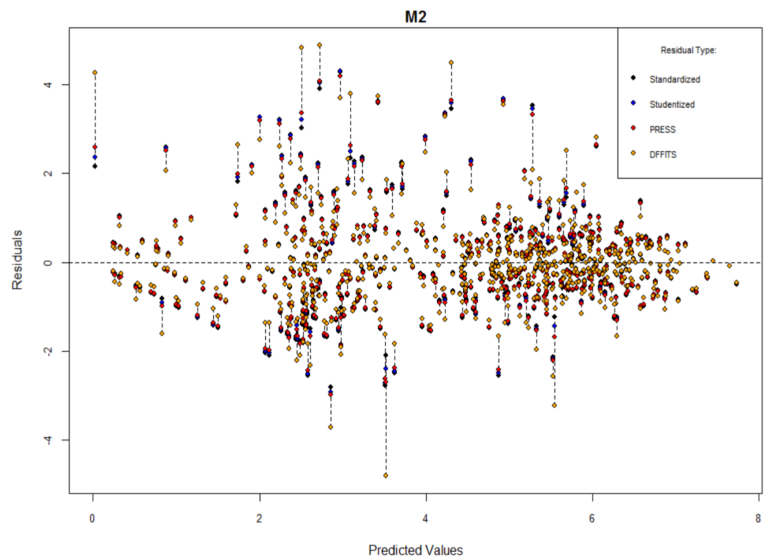


Figure 3.2.6 Residuals vs. Predicted Values(M2)

3.3 Cross-Validation

Cross-Validation helps determine which model performs better in terms of prediction. A subset of 500 observations is chosen as our training set among 625 observations in the dataset and the rest of the dataset is set as our test set. The predicting process has been replicated two thousand times. Note that SSE measures the sum of squared error of log (*Strike+1*). The result is shown as below:

	AIC	PRESS statistics	SSE	Likelihood Ratio
M_b1	1913.706	781.6154	1216.254	135.3358
M_b2	1843.654	706.7085	1234.854	

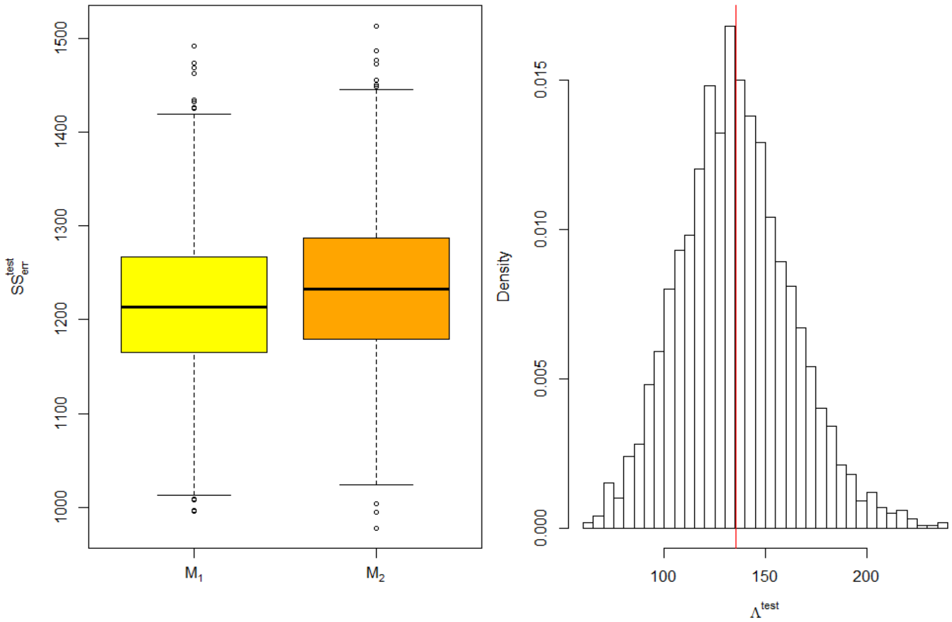


Figure 3.3.1 Cross-validation model comparison results. Left: testing sum-of-square residuals. Right: testing likelihood ratio statistic.

AIC and PRESS statistics both pick M2 over M1, and the Sum of Square Error and Likelihood Ratio indicate that M1 is a better model. However, we should notice that cross-validation is a more precise measurement of predictive power than AIC, and AIC is just a less time-consuming shortcut of cross-validation. Therefore, we pick the model that cross-validation prefers as a better model, which is M1.

3.4 Parameter estimates and Confidence Intervals

According to the analysis above, M1 is chosen as our final model.

M1: Year + Unemp + Infl + Dens + Country + Year : Unemp + Year : Infl + Unemp : Infl + Infl : Dens

The estimates of coefficients, p-values and 95% confidence intervals are displayed below.

	Estimate	Pr(>|t|)	95% Confidence interval
(Intercept)	129.4782342	<0.001 ***	[94.076, 164.881]
Year	-0.0638421	<0.001 ***	[-0.082, -0.046]
Unemp	-10.6450785	<0.001 ***	[-15.518, -5.772]
Infl	-5.7891600	0.005492 **	[-9.211, -2.368]
Dens	0.0222431	0.025750 *	[0.006, 0.039]
CountryAustria	-3.4439655	<0.001 ***	[-3.896, -2.992]
CountryBelgium	-0.4701281	0.087023 .	[-0.922, -0.018]
CountryCanada	1.2205081	<0.001 ***	[0.682, 1.759]
CountryDenmark	-2.4501125	<0.001 ***	[-2.920, -1.980]
CountryFinland	-0.8194241	0.002170 **	[-1.258, -0.381]
CountryFrance	0.3404365	0.346705	[-0.255, 0.936]
CountryGermany	-2.4133212	<0.001 ***	[-2.883, -1.943]
CountryIreland	0.3476904	0.230937	[-0.130, 0.825]
CountryItaly	1.2544640	<0.001 ***	[0.763, 1.746]
CountryJapan	-0.5472830	0.069757 .	[-1.043, -0.051]
CountryNetherlands	-2.7392185	<0.001 ***	[-3.207, -2.272]
CountryNewZealand	-0.6152693	0.029452 *	[-1.080, -0.151]
CountryNorway	-2.4983870	<0.001 ***	[-2.942, -2.054]
CountrySweden	-3.6733157	<0.001 ***	[-4.179, -3.167]
CountrySwitzerland	-4.0707646	<0.001 ***	[-4.567, -3.574]
CountryUnitedKingdom	-0.2773643	0.301546	[-0.719, 0.164]
CountryUnitedStates	1.0484533	0.002147 **	[0.488, 1.609]
Year:Unemp	0.0054371	0.000324 ***	[0.003, 0.008]
Year:Infl	0.0029656	0.005294 **	[0.001, 0.005]
Unemp:Infl	-0.0114277	0.005317 **	[-0.018, -0.005]
Infl:Dens	0.0012156	0.105258	[0.000, 0.002]

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1			

Table 3.4.1 Table of coefficient estimates, p-values and 95% confidence intervals

4. Discussion:

4.1 Most important factors

Based on *Table 3.4.1*, *Unemp* and *Infl* have both small p-values, and the interaction terms involving these two variables also have p-values less than 0.01, so they are considered the most important macroeconomics factors. Meanwhile, both *Year* and *Country* have small p-values and therefore are considered significant. To find out which of them are more important, we use anova () in R. The results are displayed as following:

```
Model 1: log(Strike + 1) ~ Unemp + Infl
Model 2: log(Strike + 1) ~ Unemp + Infl + Country + Year
  Res.Df    RSS Df Sum of Sq     F                   Pr(>F)
1    622 2060.0
2    604  772.2 18    1287.8 55.96 < 0.0000000000000022 ***

Model 1: log(Strike + 1) ~ Country + Year + Unemp + Infl
Model 2: log(Strike + 1) ~ Country + Year
  Res.Df    RSS Df Sum of Sq      F        Pr(>F)
1    604 772.20
2    606 820.56 -2   -48.366 18.916 0.0000001077 ***
```

Figure 4.1.1 anova() test results

Country and *Year* turn out to be more significant in the presence of *Unemp* and *Infl* than *Unemp* and *Infl* to be in the presence of *Country* and *Year*. This result also makes sense if we consider the interpretation of the variables. *Country* and *Year* are both fundamental determinants in the sense that the effect of the macroeconomics variables, such as unemployment rate and inflation rate, can heavily depend on which country we are talking about and what year we are in. In addition, *Country* and *Year* both contain more information other than macroeconomics factors, such as culture and educational levels, which can also affect strike activity in an indirect way. Therefore, we conclude that *Country* and *Year* are the most important factors that determine strike activity.

4.2 High-p-values in the final model

There is still a variable with high p-value (> 0.05), *Infl:Dens*, in our final model. We still keep it because in auto-selection, p-values of variables depend on the variables determined by previous steps. The order in which the variables are added to the model can also affect their p-values. It's likely that we get this high p-value by chance. We choose to be conservative in terms of deleting variables from the model obtained by auto-selection process, and the p-value of *Infl:Dens* (0.105258) is just a little higher than 0.1, so we decide to keep it in our model.

4.3 Outlying observations

We did not remove any observations during the data analysis process. We did find some points with high influence or high leverage, but based on our analysis in *Section 3.1*, those points do not have large effect on our final model M1. Also, due to lack of the background information, we did not find any good reason to remove any of the existing variables without obscuring the underlying real relationship among the variables.

4.4 Regression Assumptions

There are mainly two deficiencies of our final model--the violation of one linear regression assumption, and the not-so-good performance in prediction. The linear regression assumption A6 is violated. By looking at the *Figure 3.2.5*, the residuals do not perfectly form a standard Normal distribution. The Q-Q plot (*Figure 3.2.5*) and the histogram (*Figure 3.2.1*) show that they are little skewed to the right. In the plot of Residual vs Predicted values (*Figure 3.2.2*), we can also see diagonal lines in bottom left corner, where the residuals appear to be "linear" in predicted values and not so random. Hence, the assumption which requires the residuals to be independently identically following Normal distribution is clearly violated. The adjusted R-squared value is only 0.6922 in our final model, which we think is not large enough to make good predictions. We tried to put more covariates in the model, but that helped very little. So eventually we just retained the model as it is.

Appendix

```
strikes<-read.csv("strikes_clean.csv")
# Country excluded in interaction terms
M0<-lm(log(Strike+1)~1,data=strikes)
Mfull<-lm(log(Strike+1)~(.-Centr-Country)^2+Country,data=strikes)
Mstart<-lm(log(Strike+1)~Year + Infl + Unemp+Dens+Demo+Country,data=strikes)
Mfwd<-step(object = M0, scope = list(lower = M0, upper = Mfull),
        direction = "forward", trace = FALSE)
Mbwd<-step(object = Mfull, scope = list(lower = M0, upper = Mfull),
        direction = "backward", trace = FALSE)
Mstep <- step(object = Mstart, scope = list(lower = M0, upper = Mfull),
          direction = "both", trace = FALSE)

#Country included in interaction terms
M01<-lm(log(Strike+1)~1,data=strikes)
Mfull1<-lm(log(Strike+1)~(.-Centr)^2,data=strikes)
Mstart1<-lm(log(Strike+1)~Year + Infl + Unemp,data=strikes)
Mfwd1<-step(object = M01, scope = list(lower = M01, upper = Mfull1),
        direction = "forward", trace = FALSE)

#Residuals
y.hat<-predict(M)
h<-hatvalues(M)
res<-resid(M)
Msum <- summary(M)
sigma.hat <- Msum$sigma
#studentized residuals
stan.res <- res/sigma.hat
stud.res<-studres(M)
#press residuals
press<-res/(1-h)
#DIFFTS
dfts<-dffits(M)

p <- length(coef(M))
n <- nobs(M)
hbar <- p/n # average leverage
stud.res <- stud.res*sqrt(1-hbar) # at h = hbar, stud.res = stan.res
press <- press*(1-hbar)/sigma.hat # at h = hbar, press = stan.res
```

```
dfts <- dfts*(1-hbar)/sqrt(hbar) # at h = hbar, dfts = stan.res
#plot all residuals against predicted values
par(mfrow = c(1,1), mar = c(4,4,1.5,1))
cex <- .8
plot(y.hat, rep(0, length(y.hat)), type = "n", # empty plot to get the axis range
     ylim = range(stan.res, stud.res, press, dfts), cex.axis = cex,
     xlab = "Predicted Values", ylab = "Residuals",main="M2")
# dotted line connecting each observations residuals for better visibility
segments(x0 = y.hat,
         y0 = pmin(stan.res, stud.res, press, dfts),
         y1 = pmax(stan.res, stud.res, press, dfts),
         lty = 2)
points(y.hat, stan.res, pch = 21, bg = "black", cex = cex)
points(y.hat, stud.res, pch = 21, bg = "blue", cex = cex)
points(y.hat, press, pch = 21, bg = "red", cex = cex)
points(y.hat, dfts, pch = 21, bg = "orange", cex = cex)
abline(h=0,lty=2)

# Plot residuals against leverages
plot(h, rep(0, length(y.hat)), type = "n", cex.axis = cex,
     ylim = range(stan.res, stud.res, press, dfts),
     xlab = "Leverages", ylab = "Residuals",main="M2")
segments(x0 = h,
         y0 = pmin(stan.res, stud.res, press, dfts),
         y1 = pmax(stan.res, stud.res, press, dfts),
         lty = 2)
points(h, stan.res, pch = 21, bg = "black", cex = cex)
points(h, stud.res, pch = 21, bg = "blue", cex = cex)
points(h, press, pch = 21, bg = "red", cex = cex)
points(h, dfts, pch = 21, bg = "orange", cex = cex)
abline(v = hbar, col = "grey60", lty = 2)
legend("topright", legend = c("Standardized", "Studentized", "PRESS", "DFFITS"),
       pch = 21, pt.bg = c("black", "blue", "red", "orange"), title = "Residual Type:",
      cex = 0.7, pt.cex = 1)

#residual histograms
hist(stud.res,prob=TRUE,main ="M2")
lines(seq(-4,4,by=0.1),dnorm(seq(-4,4,by=0.1)),col="red")
#Residual QQ-plots
qqnorm(studres(M),main="M2")
```

```
qqline(studres(M))
# cook's distance vs. leverage
D <- cooks.distance(M)
# flag some of the points
infl.ind <- which.max(D) # top influence point

lev.ind <- h > 2*hbar # leverage more than 2x the average
clrs <- rep("black", len = n)
clrs[lev.ind] <- "blue"
clrs[infl.ind] <- "red"
par(mfrow = c(1,1), mar = c(4,4,2,2))
cex <- 1
plot(h, D, xlab = "Leverage", ylab = "Cook's Influence Measure",
    pch = 21, bg = clrs, cex = cex, cex.axis = cex,main="M2")
p <- length(coef(M))
n <- 625
hbar <- p/n # average leverage
abline(v = 2*hbar, col = "grey60", lty = 2) # 2x average leverage
legend("topright", legend = c("High Leverage", "High Influence"), pch = 21,
      pt.bg = c("blue", "red"), cex = 0.75, pt.cex = cex)

#Cross-validation
M1 <- Mbwd
M2 <- Mbwd1
Mnames <- expression(M[1], M[2])

AIC1 <- AIC(M1)
AIC2 <- AIC(M2) # for M2
c(AIC1 = AIC1, AIC2 = AIC2)

# PRESS
press1 <- resid(M1)/(1-hatvalues(M1)) # M1
press2 <- resid(M2)/(1-hatvalues(M2)) # M2
c(PRESS1 = sum(press1^2), PRESS2 = sum(press2^2))

# plot PRESS statistics
par(mar = c(3, 6, 1, 1))
boxplot(x = list(press1^2, press2^2), names = Mnames,
       ylab = expression(PRESS[i]^2), col = c("yellow", "orange"))
```

```
nreps <- 2e3 # number of replications
ntot <- nrow(strikes) # total number of observations
ntrain <- 500 # size of training set
ntest <- ntot-ntrain # size of test set
sse1 <- rep(NA, nreps) # sum-of-square errors for each CV replication
sse2 <- rep(NA, nreps)
Lambda <- rep(NA, nreps) # likelihod ratio statistic for each replication
system.time({
  for(ii in 1:nreps) {
    if(ii%%400 == 0) message("ii = ", ii)
    # randomly select training observations
    train.ind <- sample(ntot, ntrain) # training observations

    M1.cv <- update(M1, subset = train.ind)
    M2.cv <- update(M2, subset = train.ind)
    # testing residuals for both models
    # that is, testing data - predictions with training parameters
    M1.res <- log(Strike[-train.ind]+1) - predict(M1.cv, newdata = strikes[-train.ind,])
    M2.res <- log(Strike[-train.ind]+1) - predict(M2.cv, newdata = strikes[-train.ind,])
    # total sum of square errors
    sse1[ii] <- sum((M1.res)^2)
    sse2[ii] <- sum((M2.res)^2)
    # testing likelihood ratio
    M1.sigma <- sqrt(sum(resid(M1.cv)^2)/ntrain) # MLE of sigma
    M2.sigma <- sqrt(sum(resid(M2.cv)^2)/ntrain)
    Lambda[ii] <- sum(dnorm(M1.res, mean = 0, sd = M1.sigma, log = TRUE))
    Lambda[ii] <- Lambda[ii] - sum(dnorm(M2.res, mean = 0, sd = M2.sigma, log = TRUE))
  }
})

# plot cross-validation SSE and Lambda
par(mfrow = c(1,2))
par(mar = c(4.5, 4.5, .1, .1))
boxplot(x = list(sse1, sse2), names = Mnames, cex = .7,
        ylab = expression(SS[err]^{test}), col = c("yellow", "orange"))
hist(Lambda, breaks = 50, freq = FALSE, xlab = expression(Lambda^{test}),
     main = "", cex = .7)
abline(v = mean(Lambda), col = "red") # average value
```