2/18/2010 36-402/608 ADA-II H. Seltman

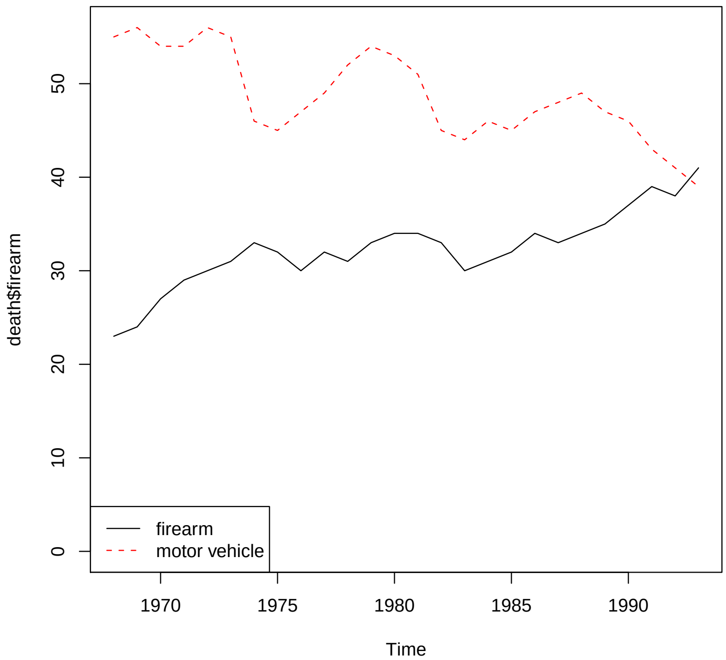
Breakout #11 Results

This dataset contains quarterly death rates from 1968 to 1993 from two causes: firearms and motor vehicles.

```
death = read.csv("ex1514.csv")
dim(death) # 26 3
sapply(death,class)
#      year      firearm motorVehicle
# "integer"    "integer"    "integer"
summary(death$year)
# Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#   1968    1974    1980    1980    1987    1993
#
#Make time series variable to hold times with values:
death$firearm = ts(death$firearm, start=1968, deltat=1)
death$motorVehicle = ts(death$motorVehicle, start=1968, deltat=1)
#
# Make a centered year variable to avoid an intercept at year 0:
death$cYr = death$year - mean(death$year)
sapply(death,class)
#   year      firearm motorVehicle          cYr
# "integer"         "ts"         "ts"    "numeric"
```

Here is some EDA:

```
plot(death$firearm, ylim=c(0,max(death[,2:3])))
lines(death$motorVehicle, col=2, lty=2)
legend("bottomleft", c("firearm","motor vehicle"), col=1:2, lty=1:2)
```

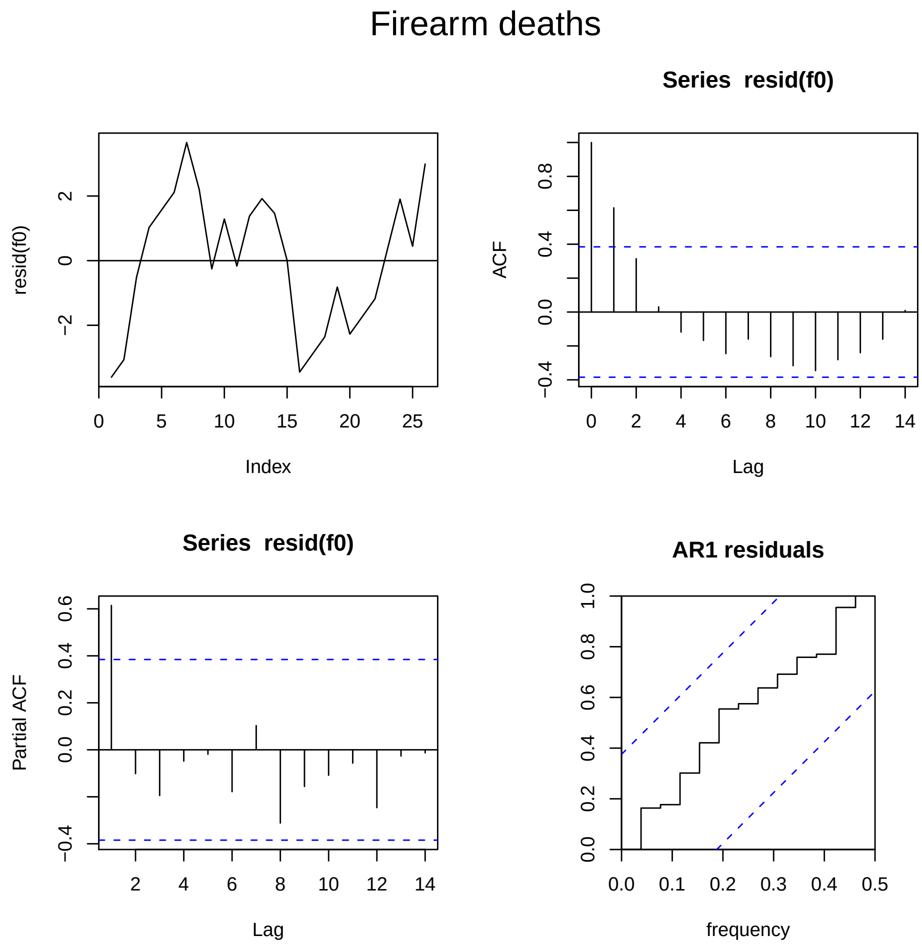


Question 1: What pattern must be fit before looking for serial correlation? Why not just look at the autocorrelation function plot of the data, instead of the residuals?

A linear change in deaths over time is seen. If you check autocorrelation directly on the data, remembering that correlation compares deviations from the mean, adjacent values will be positively correlated even when the errors are independent.

```
# A linear model over time:
f0 = lm(firearm~cYr, death)
summary(f0)
# Coefficients:
#              Estimate Std. Error t value Pr(>|t|)
# (Intercept)  32.3077     0.4148  77.896  < 2e-16 ***
# cYr           0.4561     0.0553   8.247 1.83e-08 ***
# Residual standard error: 2.115 on 24 degrees of freedom
# Multiple R-squared: 0.7392,     Adjusted R-squared: 0.7283
par(mfrow=c(2,2), oma=c(0,0,1.5,0))
plot(resid(f0), type="l"); abline(h=0)
acf(resid(f0)); pacf(resid(f0))
f1 = arima(resid(f0), order=c(1,0,0))
cpgram(f1$resid, main="AR1 residuals")
```

```
mtext("Firearm deaths", outer=T, cex=1.5)
```



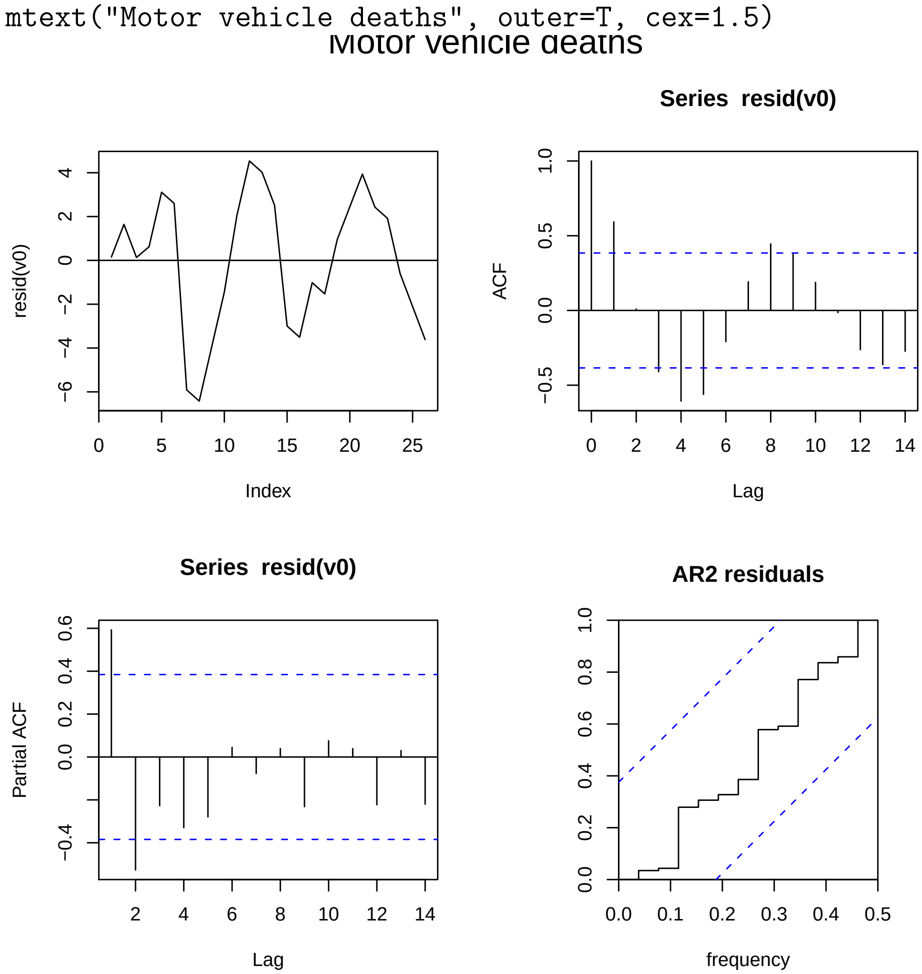
Question 2: What ARMA model is likely? Do the residuals look like white noise?

From the exponential pattern in the ACF and the single large peak in the PACF, we expect an AR(1)=ARMA(1,0,0) model. The cumulative periodogram shows white noise (all values inside the diagonal dotted lines).

Motor vehicle deaths:

```
v0 = lm(motorVehicle~cYr, death)
summary(v0)
par(mfrow=c(2,2), oma=c(0,0,1,0))
plot(resid(v0), type="l"); abline(h=0)
acf(resid(v0)); pacf(resid(v0))
v1 = arima(resid(v0), order=c(1,0,0))
v1$aic # [1] 125.3773
v2 = arima(resid(v0), order=c(2,0,0))
```

```
v2$aic
# [1] 118.1716
cpgram(v2$resid, main="AR2 residuals")
mtext("Motor vehicle deaths", outer=T, cex=1.5)
```



Question 3: What ARMA model is suggested by the plots? How does the AIC help? Do the residuals look like white noise?

With two large PACF peaks we expect AR(2)=ARMA(2,0,0) with a positive and a negative coefficient. The AIC is much smaller (7 with a “gray zone” of 2), so we really do prefer AR(2) over AR(1). The residuals show white noise.

Using the calculated AR(1) parameter, we now apply the SE correction method to testing.

```
# Pull AR(1) parameter out of arima() object:
pac=f1$coef[1] # 0.73249

# SE correction factor for autocorrelation:
SECF = sqrt((1+pac)/(1-pac))
```

```
SECF # 2.544873
```

Test $H_0 : \beta_{\mathrm{cYr}} = 0$ for firearms:

```
f0c = summary(f0)$coef
f0c
#               Estimate Std. Error   t value     Pr(>|t|)
# (Intercept) 32.3076923 0.41475366 77.896099 2.257815e-30
# cYr          0.4560684 0.05530049  8.247095 1.834314e-08
#
SEb1Adj = SECF * f0c["cYr","Std. Error"]
SEb1Adj # 0.1407327
tvalFirearm = f0c["cYr","Estimate"] / SEb1Adj
tvalFirearm # 3.24067
# adjusted p-value:
2*pt(-abs(tvalFirearm), f0$df) # 0.00348
```

Question 4: What is the general approach to getting p-values using t-tests? In what situations will the serial correlation correction factor be > 1, and what does this suggest about uncorrected tests?

If any statistic, say W, can be considered to have a Normal sampling distribution (because it is made of sums, and the data are Normal or there is enough data to use the central limit theorem), then the quantity $T = (W - \nu)/SE(W)$ follows the t distribution when the expected value of W is ν and the standard deviation of the sampling distribution of W (SE of W) is "SE(W)". The specific SE(W) formula that applies in a given situation will have σ in it, and when we substitute an estimate rather than the true value of σ we get a t rather than Z distribution. The df of the t-distribution equals the df in the estimate of σ.

The correction formula gives 1 (no change) when r=0, and values > 1 when $r > 0$, and values < 1 when $r < 0$, so when $r > 0$ uncorrected CIs will be too small, and p-values will be too small, if no correction is made.

Here is a more straightforward approach, in which arima() does the regression and calculates SE's directly. Using the `xreg=` parameter, you can `cbind()` any number of covariates.

```
dir = with(death, arima(firearm, order=c(1,0,0), xreg=cbind(cYr)))
# Coefficients:
#          ar1  intercept     cyr
#       0.7546    32.2488  0.5789
# s.e.  0.1366     1.0353  0.1260
#
```

```
# sigma^2 estimated as 2.073:  log likelihood = -46.79,  aic = 101.58

dir$coef
#       ar1  intercept        cYr
# 0.7545671 32.2488344  0.5789409

confint(dir)
#                2.5 %     97.5 %
# ar1        0.4868742  1.0222599
# intercept 30.2197781 34.2778907
# cYr        0.3319665  0.8259153

# Variance covariance matrix:
dir$var.coef
#                    ar1    intercept          cYr
# ar1        0.018654230 -0.004798695  0.007847720
# intercept -0.004798695  1.071746323 -0.002018782
# cYr        0.007847720 -0.002018782  0.015878432

# Obtaining a p-value:
tDirect = dir$coef[3] / sqrt(dir$var.coef[3,3]) # 4.59
2 * pt(-abs(tDirect), length(dir$residuals)-1) # 0.00011

## Test H_0: beta_F0 = beta_V0 (where intercept is 1980.5)
dirVehicle = with(death, arima(motorVehicle, order=c(2,0,0), xreg=cbind(cYr)))
# Variance of a difference is the sum of the variances (when independent).
SEdiff = sqrt(dir$var.coef[3,3] + dirVehicle$var.coef[4,4])
tIntDiff = (dir$coef[3]-dirVehicle$coef[4]) / SEdiff # 6.86
2 * pt(-abs(tIntDiff), 2*nrow(death)-2)
# 9.88 e-09
```

Question 5: What is a variance-covariance matrix, and how is it used here? Why did I switch from coef[3] to coef[4] and from [3,3] to [4,4]?

A V-C matrix has variances of several random variables on the diagonal and the covariances of all pairs of the random variables on the off diagonal. Here we have a sampling V-C matrix of coefficients, so the diagonals are the sampling variances of the coefficients, and the square root gives the SE.

With AR(2) instead of AR(1) there are two AR parameters to estimate so the position of the cYr variable moves up by one.