Forecasting and Time Series Analysis

San Cannon

Rockhurst University

Week 7

External information

Last week Dr. Smalter Hall talked about how he built a model to make a prediction about GDP. To do so, he didn't just rely on past values of GDP to do that forecast but instead added external information to the model.

For most of the really interesting questions, this is something that we'll want to be able to do.

You already know how to do linear regression without considering time. We've worked to incorporate time as we've modeled our time series as with linear components. And we've hinted at how to use one time series to help explain another.

Now we look at it in more detail.

Reminder: without time

We looked at how a few independent variables affected the mileage of automobiles.

Say we looked at:
$y = \beta_0 + \beta_1 * x_1 + \beta_2 * x_+ \beta_3 * x_3 + \ldots + \beta_k * x_k + \epsilon$

Where y is miles per gallon and the x's might be horsepower, weight, cylinders, etc.

We have always said that the expectation for ϵ is that they are uncorrelated. And for this simple model, that might be a good assumption.

But what if we consider time? Do you get the same miles per gallon every time you fill your tank?

Probably not. But the horsepower, weight, cyclinders, etc. don't change so why not?

Adding time

Say we modeled your mileage today:
$y_t = \beta_0 + \beta_1 * x_1 + \beta_2 * x_+ \beta_3 * x_3 + \ldots + \beta_k * x_k + \epsilon_t$

Just like we would for your mileage yesterday:
$y_{t-1} = \beta_0 + \beta_1 * x_1 + \beta_2 * x_+ \beta_3 * x_3 + \ldots + \beta_k * x_k + \epsilon_{t-1}$

The estimated relationship is the same:
$\hat{y}_t = b_0 + b_1 * x_1 + b_2 * x_2 + b_3 * x_3 + \ldots + b_k * x_k$

The differences between our actual values of y_t and y_{t-1} is due to the residuals.

What if they change over time? Why might that be?

Changing over time

What if some of those x's were things that did change over time? Maybe miles driven per trip, type of driving (highway/city), speed, weather, number of passengers, etc?

Then our relationship might look like this:
$y_t = \beta_0 + \beta_1 * x_{1,t} + \beta_2 * x_{2,t} + \beta_3 * x_3 + \ldots + \beta_k * x_k + \epsilon$
Where x_1, and x_2 are variables that change over time and x_3 and x_k might be things like cylinders which do not. What does this mean?

Now the difference between your mileage today y_t and your mileage yesterday y_{t-1} - y'_t is more than just the errors:

$y'_t = \beta_0 + \beta_1 * x'_{1,t} + \beta_2 * x'_{2,t}$

What is the implication here?

Errors and residuals

We have used the ARIMA framework to model the relationship between y_t and past values of y_t. Here we are talking about modeling y_t as a linear combination of some x's.

- The difference between y_t and $f(y_{t-k})$ is a *residual*. We used ACFs and test statistics to see if they were white noise because we want to know that we have all the information from the history of y that we can use to explain today's y.
- The difference between y_t and $\hat{y}_t$ is a *regression error*. If we believe there is information in past values of y (or x) that we haven't accounted for, it will show up in the regression error.

Estimation

If we don't account for this information, it affects our estimation:

- Our regression methodologies try to minimize the sum of squared errors. Estimating the coefficients (βs) without taking into account the information from the past means that our least squares approach will not give us the best estimators.
- Like our problem with stationarity, ignoring autocorrelation in the errors means our statistical tests aren't correct.
- AIC will not be a good measure for choosing a forecasting model

So what do we do?

So we want *residuals* to be white noise - the FPP book uses e_t to denote this - but we can model the *regression errors* to follow an ARIMA process. FPP denotes regression errors as n_t.

Let's put this in a familar form but substitute n_t for y_t:
$n'_t = c + \phi_1 n'_{t-1} + \cdots + \phi_p y'_{t-p} + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} + e_t$

Which says that the regression errors are related to the residuals by an ARIMA(p,d,q).

What does that mean really?

First: check for stationarity

We already know that non-stationary series are a pain. Using them in linear regressions is no exception.

First we check to see if things are stationary. If all our time series are stationary, we can proceed with just considering an ARMA process.

If any of the series is non-stationary, the estimated coefficients will be incorrect. [1]

It is common to treat all the variables in the model the same way to maintain interpretability. So if we need to difference one series to get stationarity, we should difference them all.

If there is trend and seasonality in multiple series, get rid of them in multiple series.

[1] Except if the series are cointegrated. We'll come back to this later.

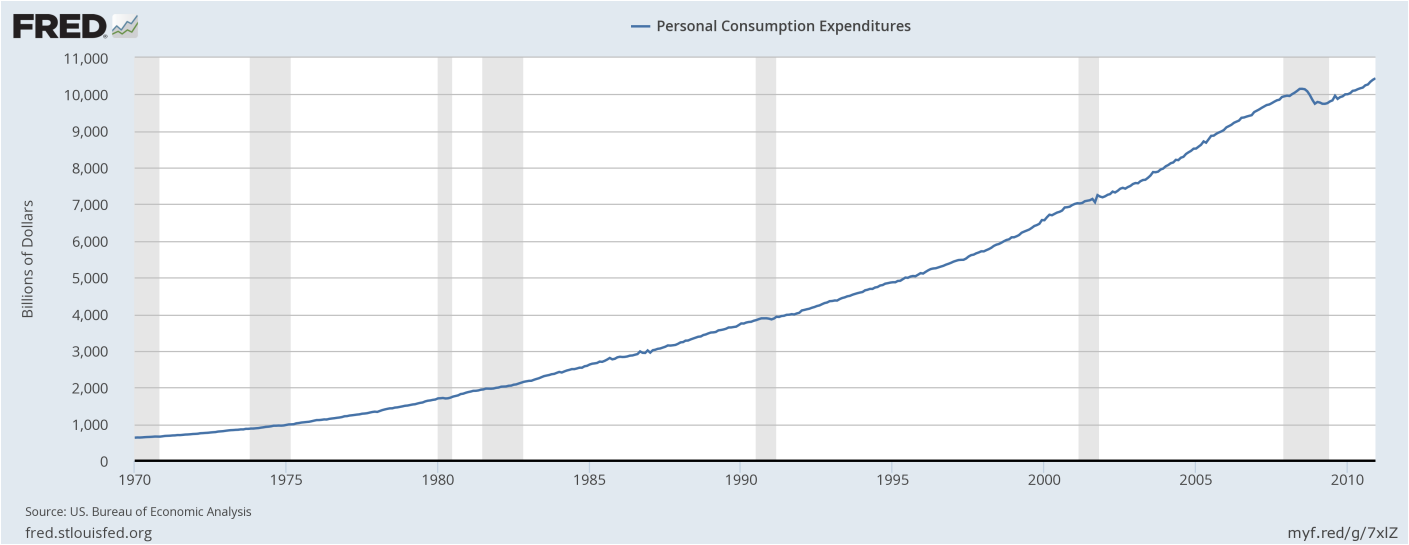
Getting started

Suggested approach:

- Make things stationary
- Begin with a proxy for your model errors. We don't know the error structure so let's start by guessing ARIMA(2,0,0) for non seasonal data and ARIMA(2,0,0)(1,0,0) for seasonal data.
- Estimate the regression coefficients, calculate the values of n_t (the errors) and identify a better ARMA model for the errors.
- Re-fit the whole model using your new ARMA structure.
- Make sure that the e_t look like white noise

Example: Forecasting consumption

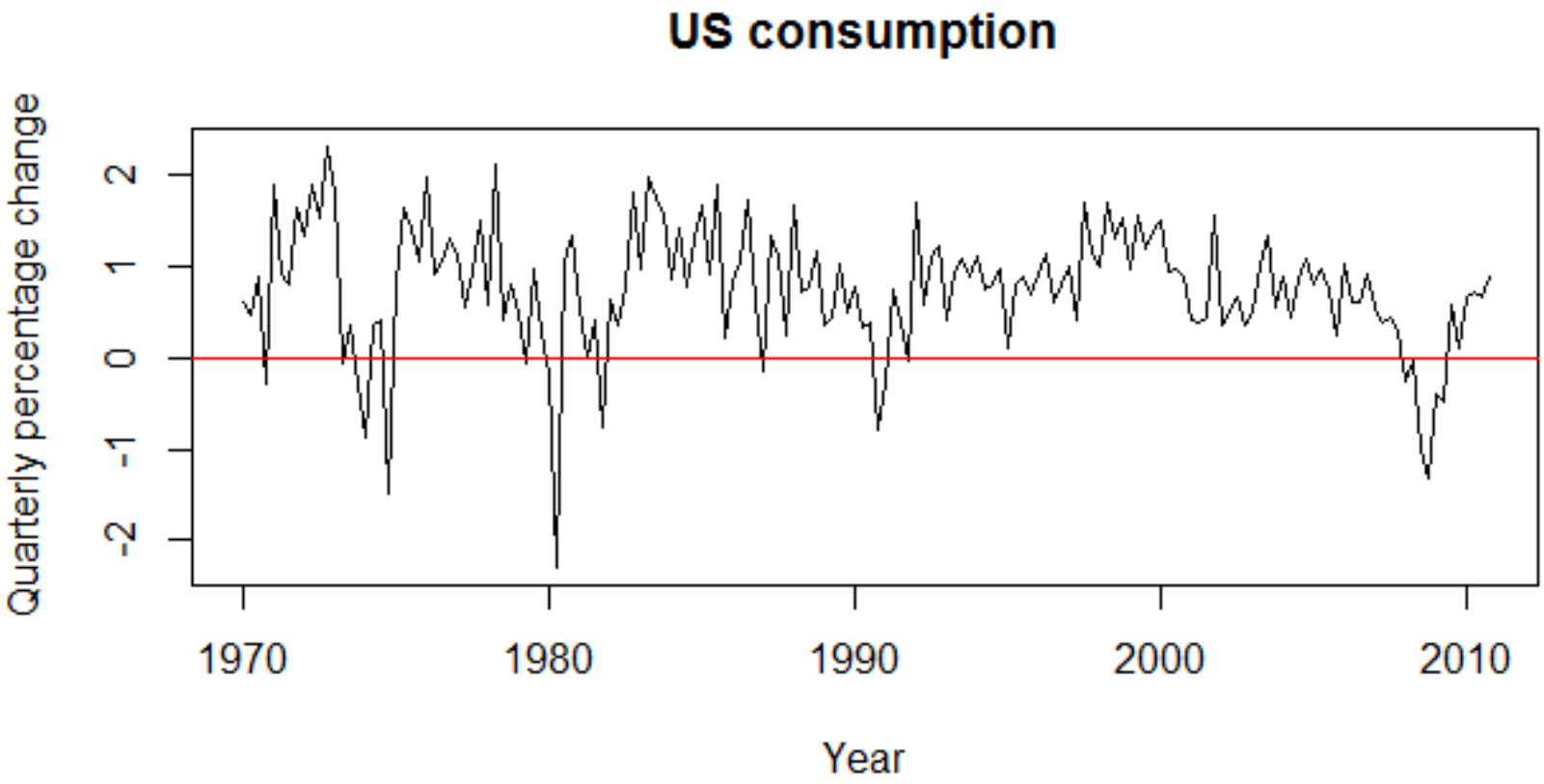
How might we forecast what people are going to buy in the future?



Clearly non-stationary so let's look at differences

Stationary consumption

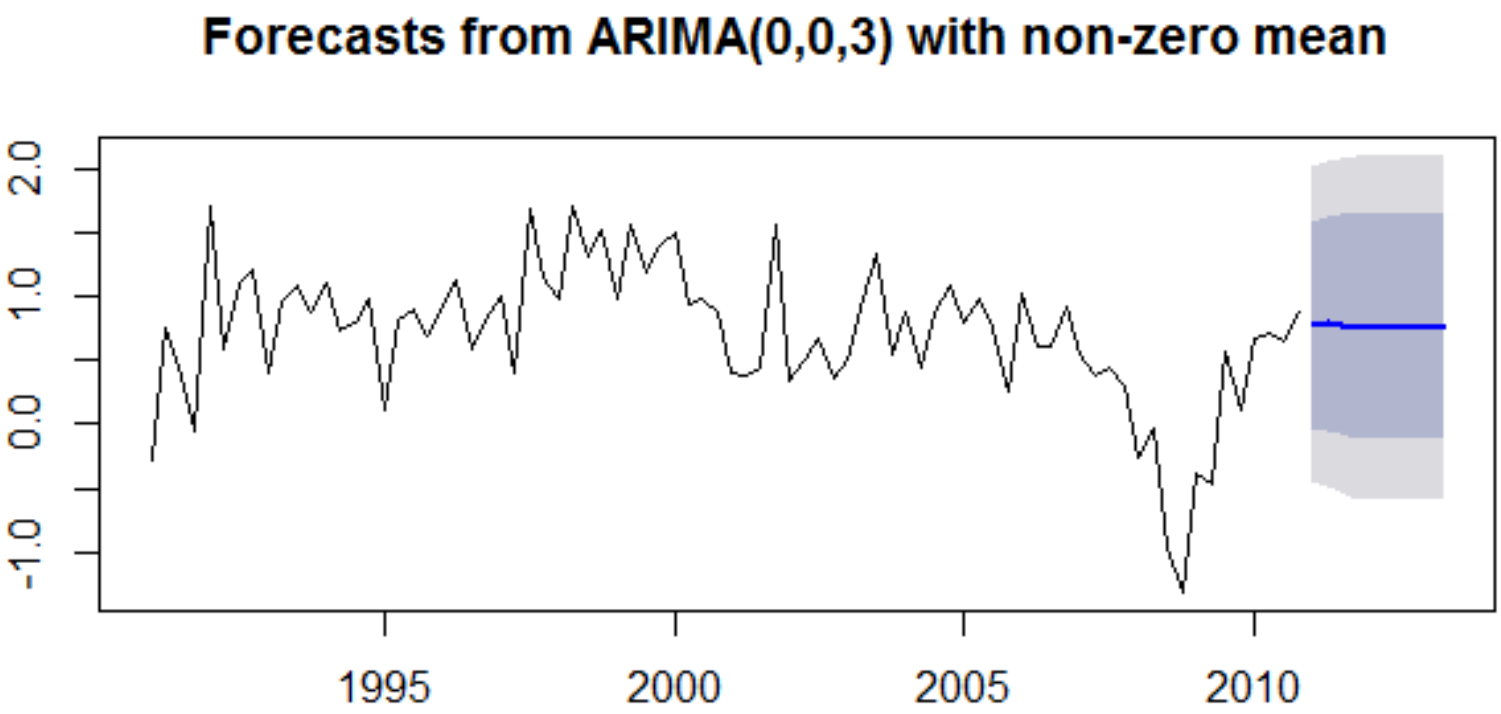
Is it stationary now?



Check our ARIMA model

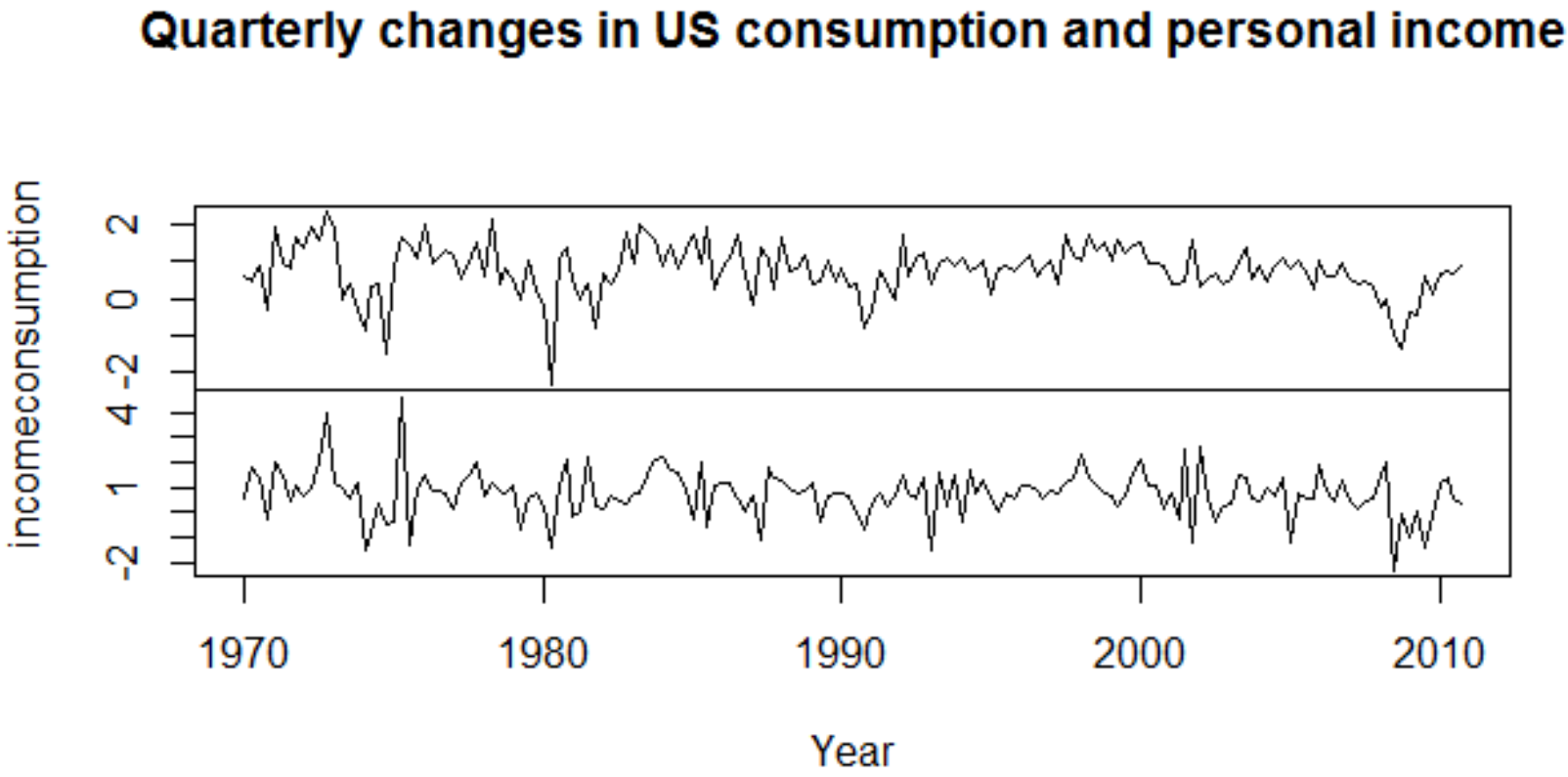
```
Series: USconsumption
ARIMA(0,0,3) with non-zero mean

Coefficients:
          ma1     ma2     ma3  intercept
      0.2542  0.2260  0.2695     0.7562
s.e.  0.0767  0.0779  0.0692     0.0844

sigma^2 estimated as 0.3953:  log likelihood=-154.73
AIC=319.46   AICc=319.84   BIC=334.96

Training set error measures:
                       ME      RMSE       MAE  MPE MAPE
Training set -4.013475e-05 0.6209988 0.4578466 -Inf  Inf
                    MASE       ACF1
Training set 0.6590496 0.0101825
```

And the forecast?



What if we use outside information?

Like income? Because consumption is differenced, we need to difference income as well.



The model

$$y'_t = b_1 x'_{1,t} + n_t$$

where

$y'_t =$ difference in consumption

$x'_{1,t} =$ difference in income

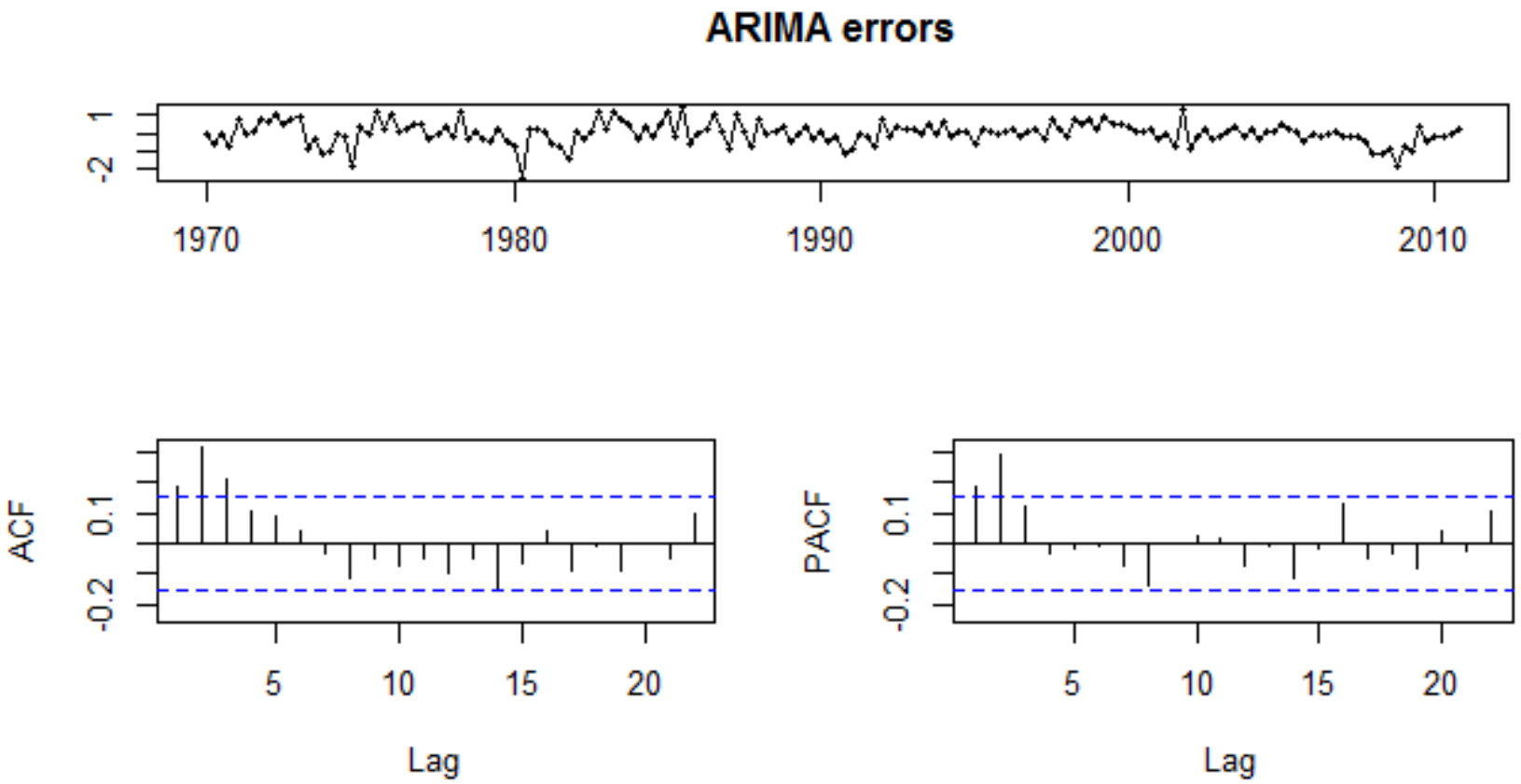
We have already seen that consumption is an MA(3) process so we know that we can't ignore the e_t or the n_t.

Modeling ARIMA errors

- Are they stationary? Check.
- Try the initial model with AR(2)

```
Series: USconsumption
ARIMA(0,0,3) with non-zero mean

Coefficients:
         ma1     ma2     ma3  intercept
      0.2542  0.2260  0.2695     0.7562
s.e.  0.0767  0.0779  0.0692     0.0844

sigma^2 estimated as 0.3953:  log likelihood=-154.73
AIC=319.46   AICc=319.84   BIC=334.96

Training set error measures:
                       ME      RMSE       MAE  MPE MAPE
Training set -4.013475e-05 0.6209988 0.4578466 -Inf  Inf
                    MASE      ACF1
Training set 0.6590496 0.0101825
```

How do those errors look?



What ARMA should the errors be?

We can use auto.arima() on the error series from the last estimated model.

```
Series: arima.errors(fit)
ARIMA(1,0,2) with zero mean

Coefficients:
          ar1      ma1     ma2
      0.6443  -0.5456  0.2224
s.e.  0.1507   0.1628  0.0774

sigma^2 estimated as 0.3461:  log likelihood=-144.32
AIC=296.64   AICc=296.89   BIC=309.04
```

Step 2

Now let's use the model specified by our last step: ARIMA(1,0,2):

```
Series: usconsumption
ARIMA(1,0,2) with non-zero mean

Coefficients:
         ar1      ma1     ma2  intercept   income
      0.6516  -0.5440  0.2187     0.5750   0.2420
s.e.  0.1468   0.1576  0.0790     0.0951   0.0513

sigma^2 estimated as 0.3502:  log likelihood=-144.27
AIC=300.54   AICc=301.08   BIC=319.14
```

So what did we get from that?

Why bother with all this? Why not just regress the change in consumption on the change in income and be done with it?

```
Call:
tslm(formula = usconsumption ~ income)

Residuals:
    Min      1Q  Median      3Q     Max
-2.3681 -0.3237  0.0266  0.3436  1.5581

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.52062    0.06231   8.356 2.79e-14 ***
income        0.31866    0.05226   6.098 7.61e-09 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.6274 on 162 degrees of freedom
Multiple R-squared:  0.1867,Adjusted R-squared:  0.1817
```

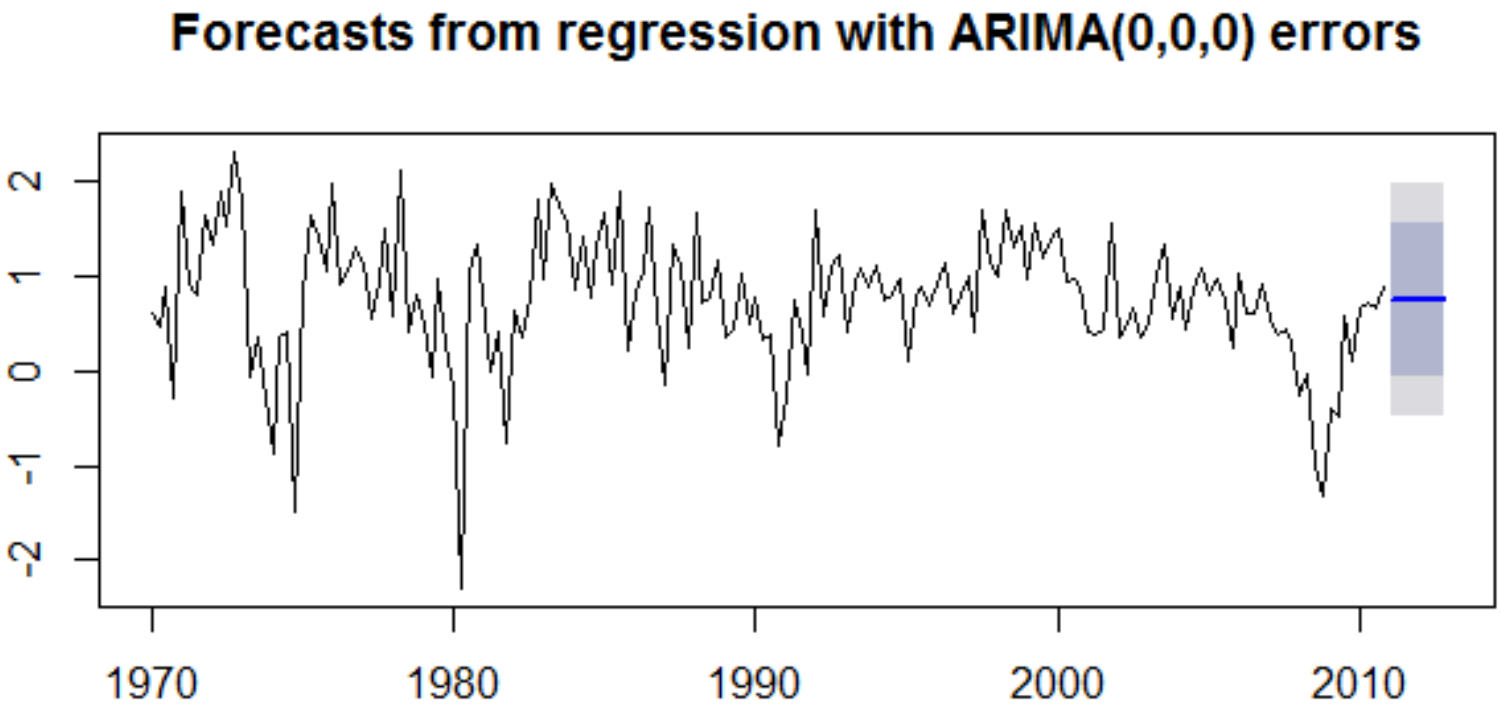
Well that's hard to compare

Let's use the same Arima framework:

```
Series: usconsumption
ARIMA(0,0,0) with non-zero mean

Coefficients:
      intercept      income
         0.5206      0.3187
s.e.     0.0619      0.0519

sigma^2 estimated as 0.3937:  log likelihood=-155.26
AIC=316.52   AICc=316.67   BIC=325.82

Training set error measures:
                      ME      RMSE       MAE  MPE MAPE
Training set 2.219122e-14 0.6236113 0.4611877 -Inf  Inf
                    MASE      ACF1
Training set 0.6638589 0.1295482
```

Did it help the forecast?

Somewhat:



But of course there is a short cut

Because the auto.arima() function will do all this for you:

```
Series: usconsumption
ARIMA(1,0,2) with non-zero mean

Coefficients:
         ar1      ma1     ma2  intercept     income
      0.6516  -0.5440  0.2187     0.5750     0.2420
s.e.  0.1468   0.1576  0.0790     0.0951     0.0513

sigma^2 estimated as 0.3502:  log likelihood=-144.27
AIC=300.54   AICc=301.08   BIC=319.14

Training set error measures:
                     ME      RMSE       MAE  MPE MAPE
Training set 0.001835782 0.5827238 0.4375789 -Inf  Inf
                    MASE        ACF1
Training set 0.6298752 0.000656846
```

What about other data?

There are *stochastic regressors* - explanatory variables that have a time dependent data generating process - and *deterministic regressors* - explanatory variables that capture the state of the world at a particular time.

- Examples of stochastic regressors
  - Interest rates
  - Consumer price index
  - Unemployment rate
  - Rate per 1000 households of television viewership
- Examples of deterministic regressors
  - Dummy coding for holiday events
  - Settings on a machine, for example, electric current, temperature, and pressure on production equipment

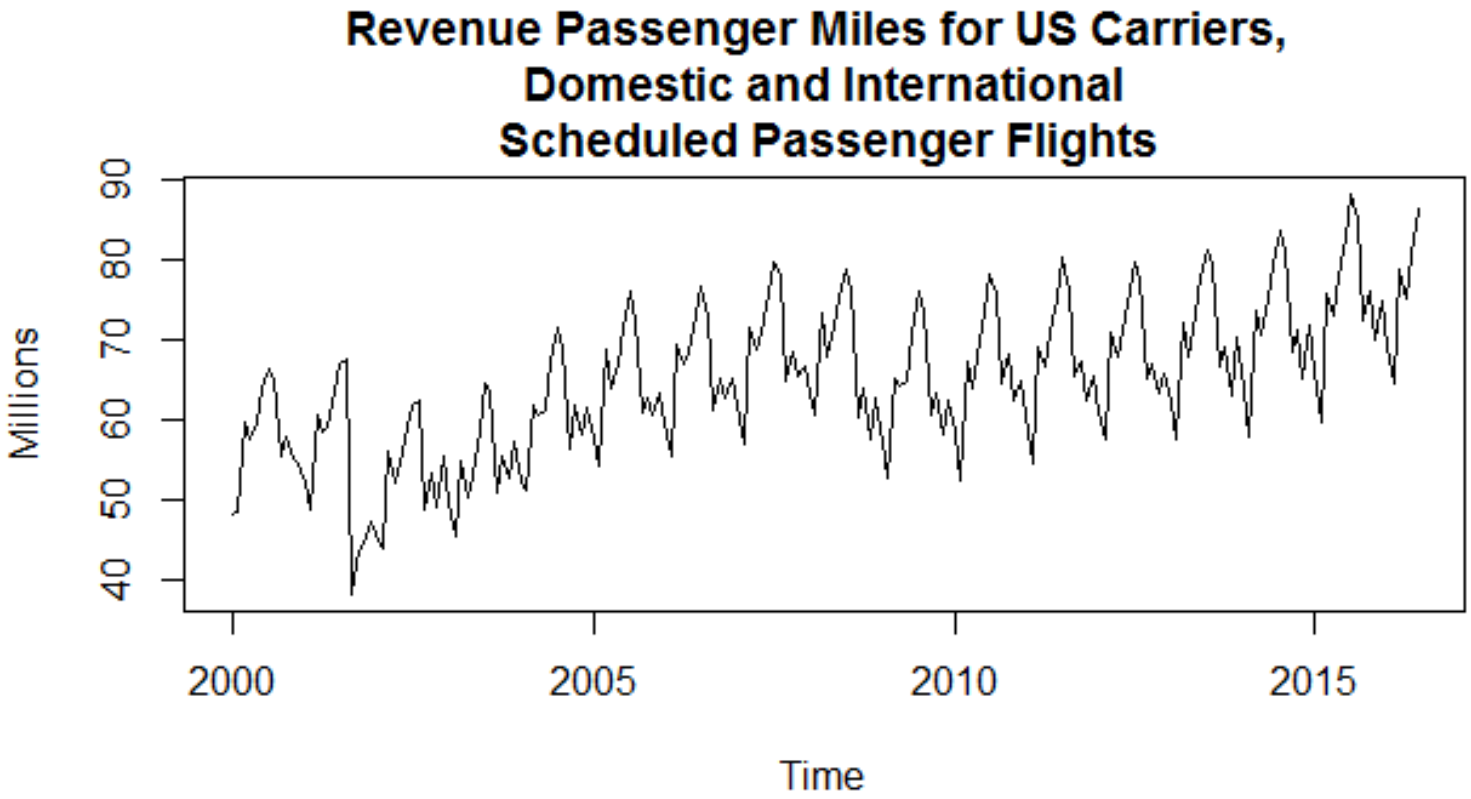
How to incorporate other data

Follow the path:

- EDA on dependent variable (thing you want to explain or forecast)
- Relevant transformation (if any) on dependent variable
- Forecasting: start with modeling individual series to see what gives you "best" forecast
- Explaining: what other information do you need? What do you need to do to it?

Example: Air miles

We can look at miles flown besides our nice Air Passenger data. Pulling data from FRED:



Source:Bureau of Transporation Statistics.

Adding dummies

Can we capture the effect of September 11 as part of the model?

Yup. Create a dummy for when it happened.

Or create a dummy for post 9/11 - what's the difference and why would you do that?

ARIMA without dummy

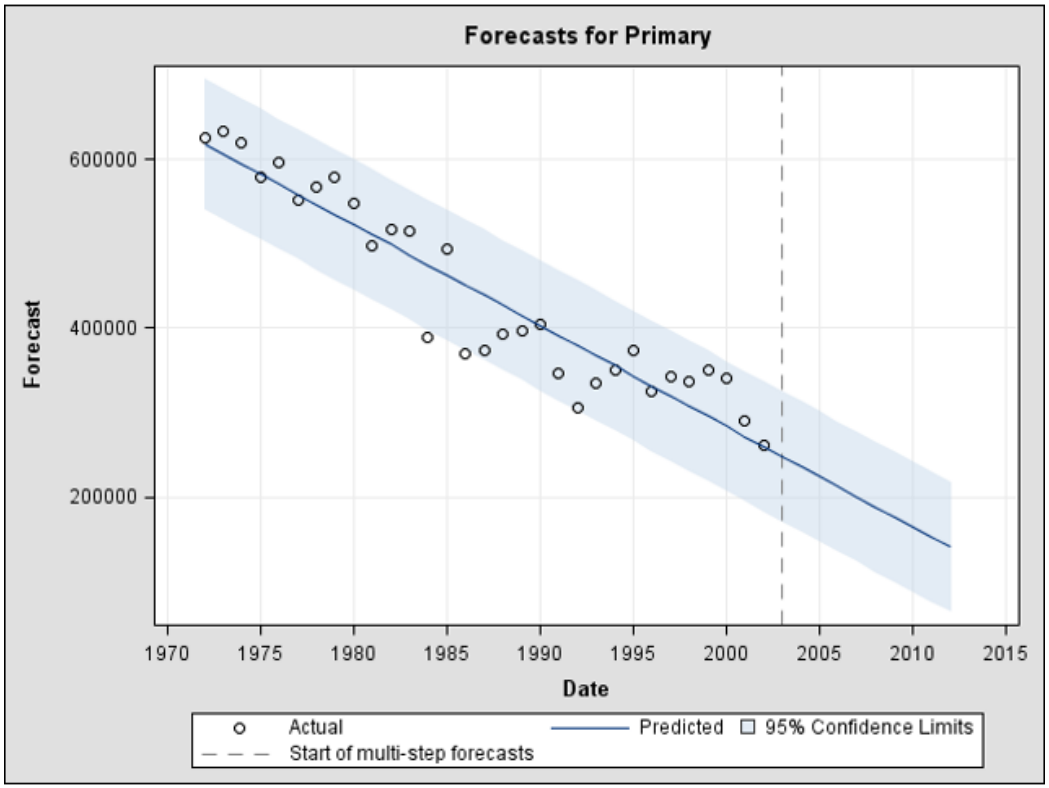
```
Series: AirMiles
ARIMA(4,1,2)(0,0,2)[12]

Coefficients:
          ar1     ar2     ar3      ar4      ma1      ma2
      -0.2749  0.6760  0.1704  -0.1448  -0.0317  -0.8966
s.e.   0.0865  0.0851  0.0775   0.0755   0.0417   0.0412
        sma1    sma2
      0.8265  0.7019
s.e.  0.0754  0.0985

sigma^2 estimated as 1.022e+13:  log likelihood=-3235.65
AIC=6489.3   AICc=6490.27   BIC=6518.85

Training set error measures:
                    ME    RMSE     MAE       MPE     MAPE
Training set 383757.9 3123607 2340498 0.2709013 3.744216
                   MASE         ACF1
```

ARIMA with dummy

```
Series: train_miles
ARIMA(1,0,2)(1,1,0)[12] with drift

Coefficients:
         ar1      ma1     ma2     sar1     drift  train_terror
      0.9030  -0.1546  0.1008  -0.5271  89934.93     -10022016
s.e.  0.0428   0.0954  0.0852   0.0812  75932.96       1318832

sigma^2 estimated as 3.519e+12:  log likelihood=-2474.27
AIC=4962.54   AICc=4963.29   BIC=4983.88

Training set error measures:
                    ME    RMSE     MAE         MPE     MAPE
Training set -6452.094 1772622 1193552 -0.1151354 2.041207
                   MASE        ACF1
Training set 0.3838268 -0.01073464
```

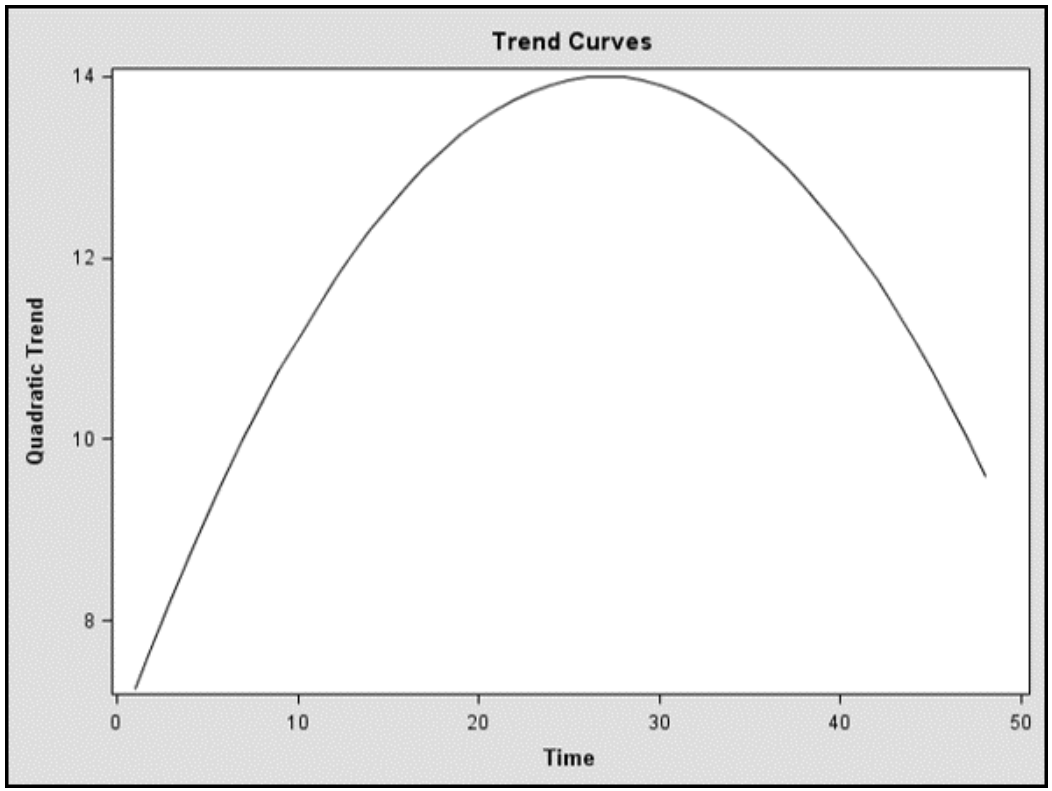
Now trends

Remember we usually use a linear trend: $y_t = \beta_0 + \beta_1 t$



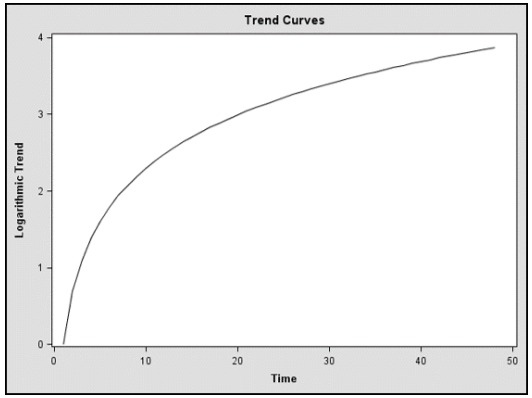
Quadratic trends

But there are quadratic trends: $y_t = \beta_0 + \beta_1 t + \beta_2 t^2$

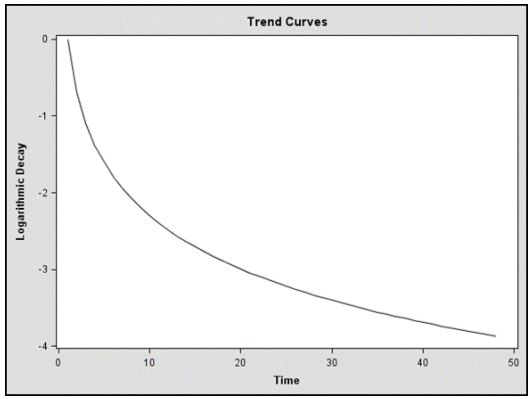


Log trends

And Logarithmic trends: $y_t = \beta_0 + \beta_1 log(t), \beta_1 > 0$



And : $y_t = \beta_0 + \beta_1 log(t), \beta_1 < 0$



Adding them to R code

In R: use arima() function:
$model = arima(yvar, xreg = xvar, order = c(p, d, q))$
Where

- *yvar* contains numeric vector of the time series process
- $xreg = xvar$ is a dataframe containing the regressors
- $order = c(p, d, q)$ defines the order of the ARIMA(p,d,q) model for the error process

Code examples

- Our PCE example: usconsumption is a data frame with two columns: consumption and income:
  $fit <- Arima(usconsumption[,1], xreg = usconsumption[,2], order = c(2,0,0))$
- Our PCE example with a linear trend:
  time = (1:length(unconsumption[,1]))
  $fit <- Arima(usconsumption[,1], xreg = data.frame(time, usconsumption[,2]), order = c(2,0,0))$
- Our Airmiles example with a dummy variable for Sept 11:
  $fit_plane <- arima(AirMiles.mill, xreg = terror, order = c(1,0,2), seasonal = c(1,1,0))$