TIME SERIES ANALYSIS:
FORECASTING PRODUCT DEMAND AND REVENUE

ECONOMETRICS LABORATORY AND
CENTER FOR REGULATORY POLICY
UNIVERSITY OF CALIFORNIA AT BERKELEY

JANUARY 7-9, 1998

LECTURE / DISCUSSION

INTRODUCTION TO FORECASTING

OBJECTIVES OF THIS COURSE

- Learn concepts (and technical jargon) for forecasting problems.
- Get hands-on experience with leading time series methods, especially *Box-Jenkins ARIMA* methods and *vector autoregressions (VARs)*.
- See how alternative methods differ in their out-of-sample predictions.

TYPES OF FORECASTING PROBLEMS

Pure Prediction Problem:

Current decisions depend upon current expectations of future values of variables of interest, but those variables are not affected by current decisions.

Example:

A firm needs predictions of future values of commodity prices (e.g., oil, coal) to determine pricing policy of future production. If the firm is relatively small customer in market, its decisions do not "feed back" into commodity prices.

Control Problem:

Future values of variables of interest depend, in part, upon variables under control of decision maker.

Example:

A firm’s future sales and revenue from its product depend upon control variables like price and advertising expenditure, as well as variables outside its control (e.g., weather, prices of competing products, etc.). Predictions must account for "feedback" from control variables to sales and revenue.

MATHEMATICAL NOTATION AND FRAMEWORK

Quantity (or quantities) to be forecasted: y_t .

- Subscript "t" denotes observation number; also involves unit of measurement of data (hours, days, weeks, months, quarters, years, etc.)

- Variable y_t can denote a single quantity of interest, or can represent a collection of N variables, in vector form:

$$y_t = (y_{1t}, y_{2t}, \ldots, y_{Nt}) .$$

Sometimes we use different letter names, rather than double subscripts, to denote different variables, e.g., "y_t and x_t" rather than "y_{1t} and y_{2t} ."

Sample of observations from $t = 1$ to $t = T$:

$$\text{Sample} = \{y_1, \ldots, y_t, \ldots, y_T\} .$$

Predicted value for future value at time period s:

$$\hat{y}_s = f_s(y_1, \ldots, y_t, \ldots, y_T) ,$$

some function of the sample of observations. Often "linear", e.g.,

$$\hat{y}_{T+1} = \hat{\pi}_0 \, y_T + \hat{\pi}_1 \, y_{T-1} + \ldots + \hat{\pi}_k \, y_{t-k} ,$$

where π coefficients might be estimated using entire sample.

EMPIRICAL EXAMPLE

A public utility has quarterly data on its total revenues (REV), total costs (COST), and an index of the price of its output (PRICE) from the early 60’s to early 90’s, and wants forecasts these variables (especially REV and COST) to determine its long-term investment strategy. Here

$$y_t = (REV_t, COST_t, PRICE_t) ,$$

where t runs from 1 (quarter #1 of 1960) to $T = 132$ (quarter #4 of 1992).

FIGURE - REVENUES, COSTS, PRICE DATA

OBJECT OF FORECASTING

For a single variable, want to minimize *mean squared error* of forecasts, i.e., minimize

$$MSE = E(\hat{y}_s - y_s)^2$$

over choice of forecasting functions $f_s(y_1, \ldots, y_s)$.

Remarks:

- Mean squared error is mathematically convenient approximation for cost of incorrect forecast.

- Could give different weights to positive and negative errors, or use non-quadratic objective function; this would complicate mathematical derivations, estimation procedures, etc.

- Treating future values as *random*, can only hope to build procedures that work well "on average," not "always." Also, quality of forecasting procedure best judged by *out-of-sample forecasts* - that is, sample MSE for observations which are *not* used to estimate unknown parameters.

ESTIMATED MEAN SQUARED ERROR

Since mean squared error (MSE) involves unobservable population expectations, we use sample MSE, or its square root (RMSE), to evaluate predictions,

$$\widehat{MSE} = \frac{1}{T} \sum_{t=1}^{T} (y_t - \hat{y}_t)^2 = \frac{1}{T} \sum_{t=1}^{T} (y_t - f_t(y_1, \ldots, y_T))^2 .$$

Better yet, can use subsample of observations from $t = 1$ to $t = S$ to estimate model, then use observations from $t = S + 1$ to $t = T$ to estimate mean squared error,

$$\widetilde{MSE} = \frac{1}{T-S} \sum_{t=S+1}^{T} (y_t - f_t(y_1, \ldots, y_S))^2,$$

which gives more realistic "out-of-sample" measure of goodness of fit of forecasting equations.

"CHAIN RULE OF FORECASTING"

To forecast several periods ahead, can often recursively use a one-step-ahead forecasting relation. That is, if we have

$$\hat{y}_{T+1} = f(y_T, y_{T-1}, y_{T-2}, \ldots)$$

for some estimated function $\hat{f}$, get the prediction for time T+2 by using same formula, adding "1" to every subscript and substituting forecasted $\hat{y}_{T+1}$ for y_{T+1}:

$$\hat{y}_{T+2} = f(\hat{y}_{T+1}, y_T, y_{T-1}, \ldots) .$$

Can use same approach ("chain rule of forecasting") to get $\hat{y}_{T+3}$, $\hat{y}_{T+4}$, etc., once we have one-step forecasting rule.

AD-HOC FORECASTING METHODS

Features:

- Simple rules for extrapolating data into future.

- No explicit model for how data is generated, and no systematic adjustment for past forecasting errors.

Examples:

- Subjective Forecasts
- Deterministic Models
- Moving Averages
- Exponentially-Weighted Moving Averages

Subjective Forecasts:

Experienced manager makes predictions using individual-specific approach (e.g., "bend the ruler").

Advantages:

- Manager might have non-quantifiable prior information ("experience") which aids in prediction.

Disadvantages:

- Assessment of quality of forecasts is difficult (need standard for comparison).

- Forecasting procedure not replicable across different managers.

Deterministic Models:

Assume y_t generated from fixed function,

$$y_t = f(t) ,$$

so forecasts are of the form $\hat{y}_s = f(s)$. Common choices of functions are polynomials or exponentials,

$$f(t) = \alpha_0 + \alpha_1 \cdot t + \alpha_2 \cdot t^2 + \ldots + \alpha_K \cdot t^K, \quad \text{or}$$

$$f(t) = A\, e^{b \cdot t} .$$

Parameters chosen to exactly fit most recent observations.

Advantages:

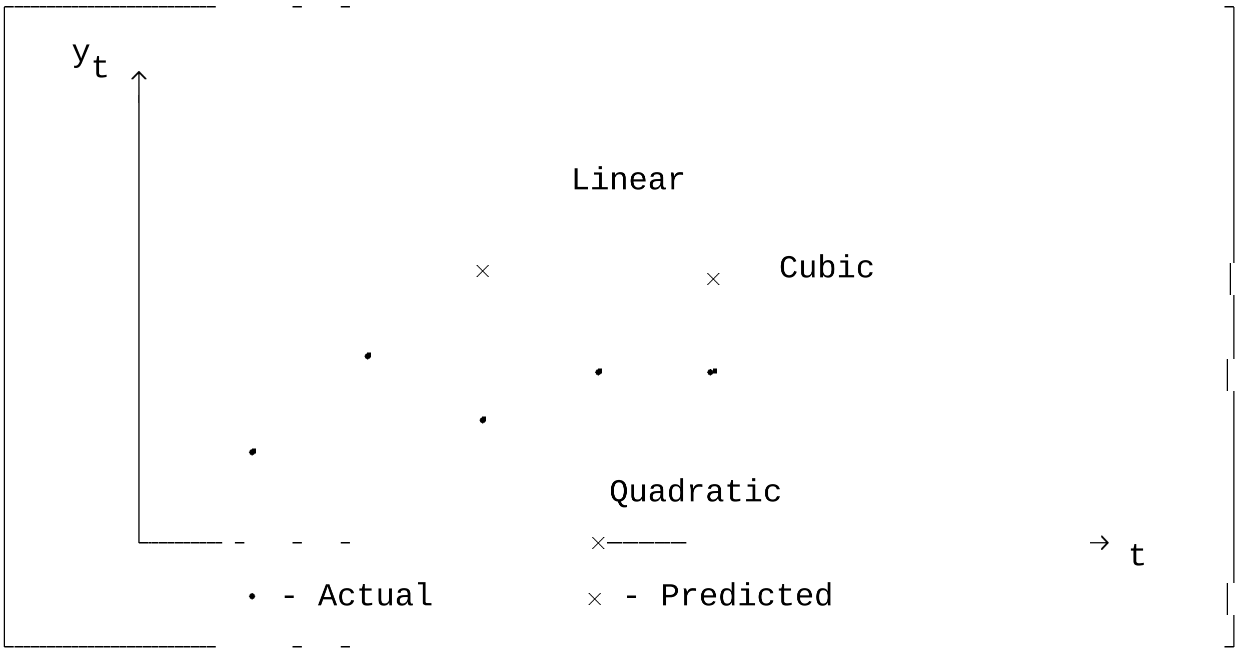
- Simple approach; requires few observations (equal to number of parameters).

Disadvantages:

- Models generally don’t fit perfectly if T exceeds number of parameters.

- Forecasts $\hat{y}_s$ very sensitive to choice of number of parameters, and can vary widely if new observations incorporated in forecasting procedure.

FIGURE - SEQUENTIAL POLYNOMIAL FITS



Moving Averages:

Simple (equally weighted) average of most recent K observations on y_t:

$$\hat{y}_{T+1} = \frac{1}{K}(y_T + y_{T-1} + \dots + y_{T-K+1}) ,$$

$$\hat{y}_{T+2} = \frac{1}{K}(\hat{y}_{T+1} + y_T + \dots + y_{T-K+2}) , \quad \text{etc.}$$

Advantages:

- Will be optimal forecasting method for a particular time series (autoregressive) model.

- Simple to compute; no unknown parameters to estimate.

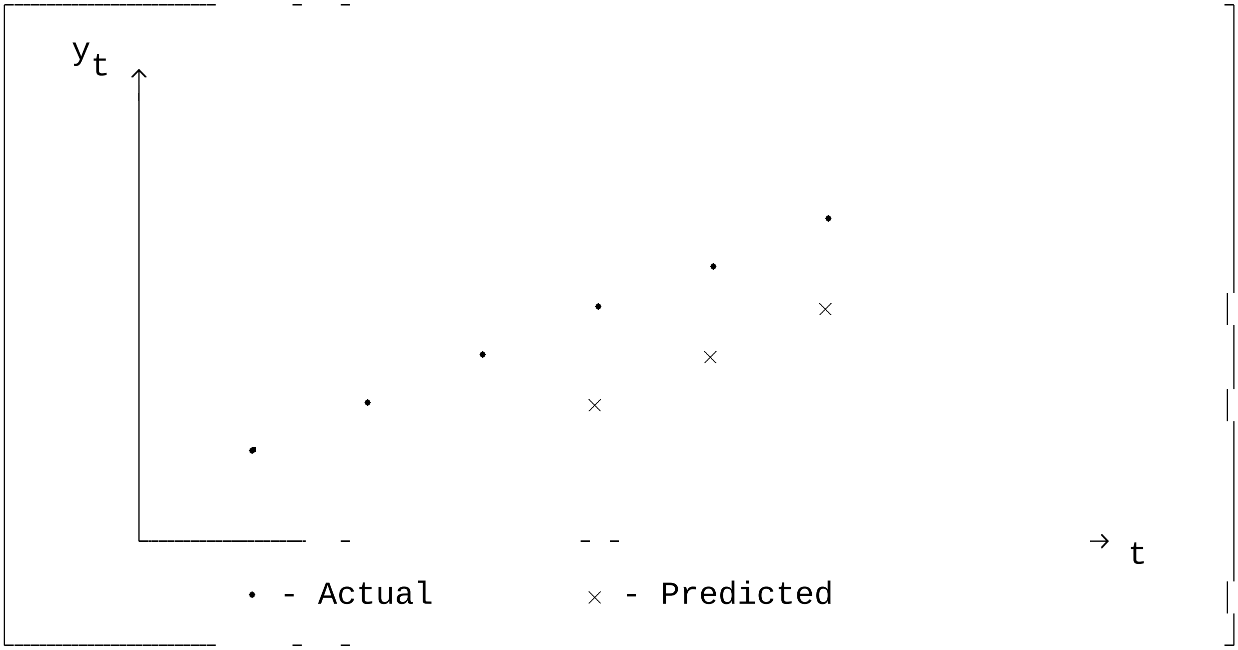
Disadvantages:

- Can get better predictor in general by estimating a particular time series (autoregressive) model.

- For trending data, can systematically under- or over-predict y_t. For example, if

$$
\begin{aligned}
y_t &= \alpha + \beta\cdot t\,, && \text{then} \\
\hat{y}_{T+1} &= \alpha + \beta\cdot(T - (K + 1)/2) && \\
&< y_{T+1} && \text{if } \beta > 0.
\end{aligned}
$$

FIGURE - SIMPLE MOVING AVERAGE PREDICTOR (K = 3)



Exponentially Weighted Moving Average (EWMA):

Starting with "current predicted value" $\hat{y}_T$, next period predictor is weighted average of current value and current prediction:

$$\begin{aligned} \hat{y}_{T+1} &= p \cdot y_T + (1 - p) \cdot \hat{y}_T \ , \\ &= y_T + (1 - p) \cdot (y_T - \hat{y}_T) \ , \end{aligned}$$

$$\hat{y}_s = \hat{y}_{s+1} \quad \text{for} \quad s = T + 2, T + 3, \ldots \ ,$$

for some π between zero and one.

Advantages:

- Will be optimal forecasting method for a particular time series (moving average) model.

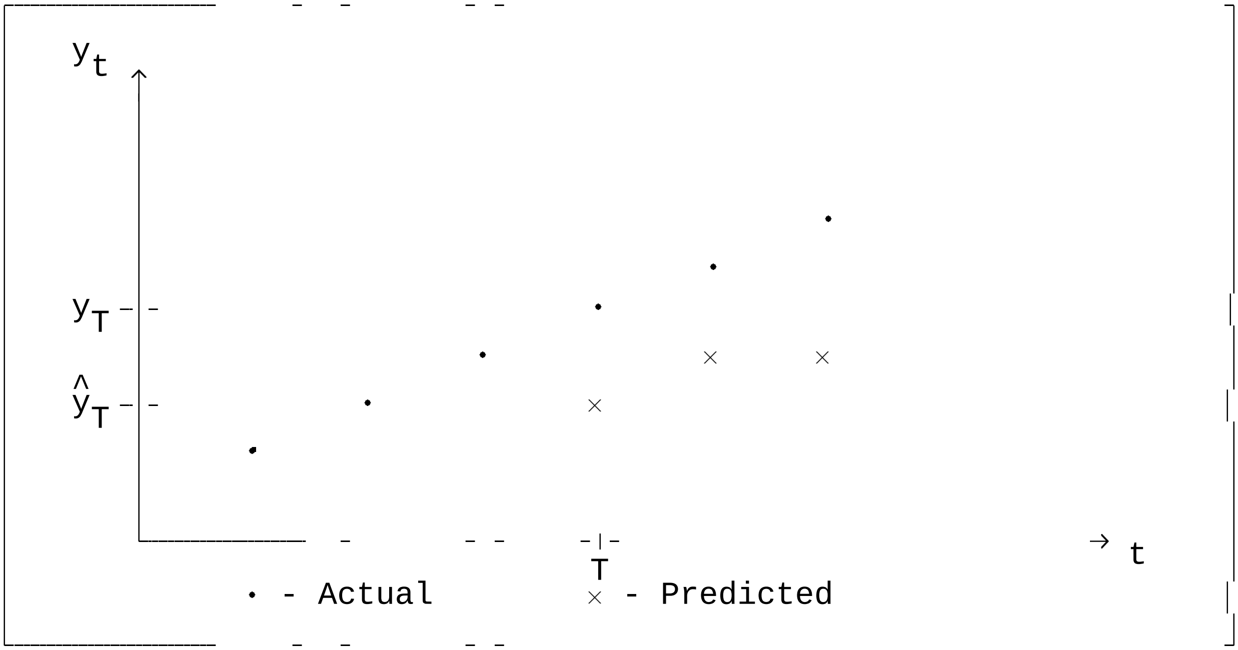
Disadvantages:

- Arbitrary value of p (e.g., $p = 1/2$) isn’t optimal; can get better predictor in general by estimating "best" coefficient.

- Forecasted values $\hat{y}_s$ are constant for all future time periods.

- Can systematically under- or over-predict y_t for trending data, just like simple moving average.

FIGURE - EWMA PREDICTOR (P = 3)



TRADITIONAL ECONOMETRIC METHODS

Features:

- Based upon linear regression and related methods; often derive models from economic theory.

- Emphasis is on estimates of model parameters (like elasticities), not on forecasted values of y_s.

- Useful for analysis of *structural (policy) change*, since parameters assumed constant when policy changes.

Examples:

- Deterministic Regression Models.
- Distributed Lag Models.
- Simultaneous Equations Models.

Deterministic Regression Models:

Add error term to deterministic prediction model; for example,

$$y_t = \alpha_0 + \alpha_1 \cdot t + \alpha_2 \cdot t^2 + \ldots + \alpha_K \cdot t^K + \varepsilon_t$$

or

$$y_t = A \cdot e^{b \cdot t} + \varepsilon_t \,.$$

Estimate unknown parameters by (linear or nonlinear) least squares.

- Model might include seasonal dummies for recurring intercept shifts.

Advantages:

- Permits deviations of data from deterministic trend.

- Useful first step in building time series model for residuals ("detrending").

Disadvantages:

- No direct interaction of forecasted values with current observations, once parameters are estimated. No effect of past residuals.

- Predictions are sensitive to specification of trend function, particularly for long-term forecasts.

- Doesn’t allow for interactions of forecasted values with other observable variables.

Distributed Lag Models:

Regression model for y_t which has trends and current and lagged values of other predictor variables on right-hand side. For example,

$$y_t = \alpha + \beta \cdot t + \gamma \cdot x_t + \delta \cdot x_{t-1} + \varepsilon_t .$$

Use standard least squares methods to estimate unknown coefficients.

Advantages:

- Permits deviations of data from deterministic trend, and allows for interactions with other variables. Useful for control problems.

Disadvantages:

- Same as for deterministic regression model; in addition, non-deterministic regressors x_t must be forecasted to obtain forecasts of y_t.

Simultaneous Equations Models:

For multivariate dependent variables, system of equations which simultaneously determines components of y_t given current and lagged regressors x_t. For example,

$$y_{1t} = \alpha_1 + \beta_1 \cdot y_{2t} + \gamma_1 \cdot x_t + \delta_1 \cdot x_{t-1} + \varepsilon_{1t},$$
$$y_{2t} = \alpha_2 + \beta_2 \cdot y_{1t} + \gamma_2 \cdot x_t + \delta_2 \cdot x_{t-1} + \varepsilon_{2t},$$

which simultaneously determine y_{1t} and y_{2t} .

Advantages:

- If model correctly specified, can use model to forecast effects of policy interventions (e.g., introduction of sales tax) without historical data.

- Permits complex interactions among dependent variables.

Disadvantages:

- Results very sensitive to model specification; in particular, identification of parameters requires extra assumptions (e.g., $\gamma_1 = \delta_2 = 0$) which might be suspect.

TIME SERIES MODELS

Features:

- Emphasis on *forecasts*, not *model parameters*; preference for simple models, not necessarily economically interpretable models.

- Approach expesses y_t as a (possibly time-dependent) function of past values of y_t and current and past error terms ε_t:

$$y_t = f(y_{t-1}, y_{t-2}, \ldots, y_{t-p}, \varepsilon_t, \varepsilon_{t-1}, \ldots, \varepsilon_{t-q}, t) .$$

- The function f is typically assumed linear; parameters are estimated using variants of least squares. Forecasts of y_t are obtained by recursively fitting model into future, replacing future errors by their predicted (zero) values.

Examples (will be covered in detail later):

- ARIMA(1,1,1):

$$y_t = y_{t-1} + \alpha + \beta\cdot(y_{t-1} - y_{t-2}) + \varepsilon_t + \theta\cdot\varepsilon_{t-1}.$$

- VAR(1):

$$y_{1t} = \alpha_1 + \beta_1 y_{1,t-1} + \gamma_1 y_{2,t-1} + \varepsilon_{1t},$$

$$y_{2t} = \alpha_2 + \beta_2 y_{1,t-1} + \gamma_2 y_{2,t-1} + \varepsilon_{2t}.$$

- GARCH(1,1):

$$y_t = y_{t-1} + \varepsilon_t ,$$

$$\varepsilon_t = h_t\cdot u_t , \qquad u_t \text{ i.i.d.},$$

$$h_t = \alpha + \beta\cdot h_{t-1} + \gamma\cdot(\varepsilon_{t-1})^2.$$

[Here, the variance of y_t has complex dynamic behavior, rather than the mean.]

LECTURE / DISCUSSION

INTRODUCTION TO TSP STATISTICAL PACKAGE

STEPS TO RUN A TSP JOB

1. Create or modify a file of TSP commands.
2. Run the TSP command file.
3. Print the output.

TSP COMMAND FILE:

EXAMPLE.TSP

```
options crt limwarn=0;

freq q;
smpl 60:1 92:4;
load(file="utildata.xls", format=excel)
                                rev cost price;
trend t;
y=rev;

yma=0.2*(y(-1)+y(-2)+y(-3)+y(-4)+y(-5));

olsq y c t;
yhat=@fit;

plot y * yhat ^ yma +;

stop;
```

TO RUN THE PROGRAM:

1. Login. You will be in the Sun Open Windows graphical environment, with the following display:

   Upper left corner - file manager window
   Right side - shelltool window with Unix prompt (% sign)
   A few miscellaneous icons (clock, mailtool) that can be ignored.

2. The file manager is a window showing your directories and files; these look like pictures of file folders. Open the **wedmorn** directory by clicking twice with the left mouse button while the pointer is on the **wedmorn** icon. You should now see a folder icon labeled **example.tsp**.

3. Bring this file into the text editor by pointing to the icon labeled **example.tsp** and clicking twice with the left mouse button. You will get a text editor window that shows the contents of the file.

4. Edit the file as you want. The editor is fairly straightforward (the arrows work as on a PC, and backspacing will delete characters as you go backwards). You can position the cursor anywhere by moving the pointer and clicking once with the left mouse button.

5. Save the edited file under a new name. To do this, point to the **File** menu button (upper left corner of the text editor window) and click once with the right mouse button. This will bring up a menu from which you should choose **Store as a New File** by clicking once on it with the left button. You will be prompted for a new file name. Type in the name of the new file, such as **myprog.tsp**, or whatever name you want to give your command file. The name must end in **.tsp**, however.

6. (OPTIONAL) To exit the text editor window, point to any part of the upper frame (darker gray) and click once with the right button. Select **Quit** (bottom of the menu) and click once with the left button. This will make the text editor window go away. Alternatively, you can just close (iconify) the text editor window by pointing to the upside-down triangle in the upper left corner of the window and clicking once with the left button.

7. In the window with the Unix prompt (%), change to the **wedmorn** directory and then run the program in TSP by typing, for example:

   ```
   cd wedmorn
   tsp file1 file2
   ```

   at the percent prompt. The name of the TSP command file you want to run is **file1** (you can omit the **.tsp** extension of the filename when you use the "tsp" command, since the program automatically looks for command files ending in **.tsp**). The name of the output file is **file2** (TSP will add the extension **.out** to this file name, e.g., **file2.out**). In this example, we will replace **file1** (and, optionally, **file2**) with **example**.

NOTE: Do not worry about multiple icons or having multiple windows open at the same time. When you log out, all windows (open and iconified) will be closed properly.

To LOGOUT of the workshop session, move the pointer to a blank (non-window) area of the scressn. CLick the right button of the mouse, point to EXIT, and click on EXIT with the left button. The pop-up menu that appears will ask for confirmation of the EXIT command by offering you an EXIT or CANCEL choice. Move the pointer to the EXIT option and click on it with the left button. This officially logs you out of the Open Windows environment and ends your current session.

Key TSP Commands for This Tutorial

help	(gives help on various commands - use in interactive mode)
smpl	(sets sample size for input, estimation, prediction)
freq	(sets the data frequency, "quarterly" or "monthly")
trend	(creates a trend variable)
y1=y(-1)	(creates the lagged variable y_{t-1} from y_t)
olsq	(least squares regression procedure)
plot	(plots data series)
print	(prints specific variables, like regression coefficients)
msd	(prints out summary statistics, like means and std. dev’s)
forcst	(forecasts using most recent **olsq** coefficients)
bjident	(prints out autocorrelation functions for Box-Jenkins identification methods)
bjest	(estimates coefficients for Box-Jenkins ARIMA models)
bjfrcst	(like **forcst**, but for ARIMA models)
unit	(procedure to test for "unit roots")
var	(estimates vector autoregression models)
siml	(used for forecasting using vector autoregression estimates)
coint	(procedure to test for cointegrating relationship)
arch	(estimation method for ARCH models)

WORKSHOP

SIMPLE FORECASTING PROCEDURES

SIMPLE FORECASTING METHODS FOR UTILITY DATA

The **wedmorn** subdirectory contains the file "utildata.xls," which is a Microsoft Excel-compatible file containing the revenue, cost, and price data for the utility in the cited empirical example. The sample ranges from the first quarter of 1960 (60:1) to the last quarter of 1992 (92:4). It also has the files **example.tsp** and **example2.tsp** for this workshop.

1. Examine the file **example.tsp.** The first few lines set some options (like turning off non-fatal warning messages) and loads the variables, "rev," "cost," and "price." The line

   ```
   freq q;
   ```

   sets the data frequency to "quarterly" (instead of "m," for "monthly"), and the statement

   ```
   smpl 60:1 92:4
   ```

   sets the sample size. Then the program defines the trend variable "t", and defines "y" (the current dependent variable) as the "rev" variable. The next line defines "yma" as the moving-average predictor of "y", using a moving average of the most recent K=5 values of y. Then the program regresses y on a constant "c" and the trend variable "t," saves the fitted values as "yhat," and plots the results.

2. Run the program **example.tsp** in the window with the Unix % prompt by typing

```
tsp example util
```

Your output will have the name "util.out." You can look at the output by typing

```
more util.out
```

or print the output by typing

```
qlpr util.out
```

3. Do the different fitting methods track the "rev" (revenue) series very well within the sample period (using the "eyeball metric")? Change the dependent variable from "y=rev" to "y=cost" and "y=price" and see if your answers change. (See the files **example2.tsp** and **example3**.)

4. For the "price" variable, for which the OLS fitted values don’t fit well, try adding a quadratic trend term, by defining a squared trend variable as

```
t2=t**2;
```

before the OLSQ procedure, and modifying that program line to

```
olsq y c t t2;
```

to make the deterministic regression a quadratic specification in time. How do the results look with this specification? (**example4.tsp**)

5. Now examine the file **example5.tsp** in the **wedmorn** subdirectory. This program extends the previous program in several ways:

    1) Constructs the EWMA predictor, "yewma," using a weighting factor of p = 0.5;
    2) Reserves the last 12 observations (90:1 through 92:4) for out-of-sample forecasting and model evaluation;
    3) Only fits the deterministic regression up to 89:4, and uses the TSP command **forcst** to predict out-of-sample;
    4) Creates a new TSP procedure, **mspe**, to evaluate the mean-squared prediction error in the post-estimation period.

6. Run the **example5.tsp** program, and observe the differences in mean-squared prediction errors for each of the three forecasting methods (moving average, EWMA, and deterministic regression). Also, try changing the dependent variable to "price," and adding the quadratic term to the regression (**example6.tsp**).

7. How does the mean-squared prediction error vary with K, the number of terms in the moving average (currently K=5) and p, the weight factor in the EWMA? Does the MSPE increase or decrease as K and p increase? What happens as p approaches 1?

DISCUSSION OF WORKSHOP RESULTS

LECTURE / DISCUSSION

PRELIMINARY DATA TRANSFORMATIONS

"PREFILTERING"

Typically, original dependent variable y_t must be transformed ("prefiltered") to fit into linear time series framework. Such transformations include:

Logarithmic Transformation:

For (nonnegative) economic data, typically take

$$y_t = \log(Y_t) \ ,$$

where Y_t is original (level) variable.

• Changes in $\log(Y_t)$ over time are (approximately) percentage changes in original variable Y_t, so forecasts concern growth *rates* rather than *levels*. That is

$$y_t - y_{t-1} = \log(Y_t) - \log(Y_{t-1}) \simeq \frac{Y_t - Y_{t-1}}{Y_t} \ .$$

Detrending and Seasonal Adjustment:

The classical decomposition of a time series variable y_t assumes it is composed of trend, seasonal (if measurement interval is less than one year), and "irregular" components:

$$y_t = \text{trend}_t + \text{seasonal}_t + u_t ,$$

where the trend and seasonal are modeled with deterministic functions and u_t is assumed to follow a time series model. So "prefilter" involves preliminary least-squares fit of determinstic regression function, e.g.,

$$y_t = \alpha + \delta \cdot t + \gamma_2 \cdot s_{2t} + \ldots + \gamma_K \cdot s_{Kt} + u_t ,$$

where s_{jt} is indicator (dummy) for season "j". Time series model is applied to residuals

$$\hat{u}_t = y_t - \hat{\alpha} + \hat{\delta} \cdot t + \hat{\gamma}_2 \cdot s_{2t} + \ldots + \hat{\gamma}_K \cdot s_{Kt} .$$

EXAMPLE OF SEASONAL DATA

Monthly housing starts in the United States, 1986-1995.

Differencing:

For non-seasonal but trending (growing) data, an alternative to detrending is *first differencing*, i.e., "prefiltering" y_t by modelling

$$w_t \equiv \Delta y_t \equiv y_t - y_{t-1}$$

by time series model. This removes linear trends, since if

$$y_t = \alpha + \delta \cdot t + u_t , \qquad \text{then}$$

$$\begin{aligned}\Delta y_t &= (\alpha + \delta \cdot t + u_t) - (\alpha + \delta \cdot (t-1) + u_{t-1}) \\ &= \delta + u_t - u_{t-1} ,\end{aligned}$$

which no longer depends directly on time t.

Higher-Order Differences:

If trend is quadratic in time t, can use *second difference*: if

$$y_t = \alpha + \delta \cdot t + \gamma \cdot t^2 + u_t, \quad \text{then}$$

$$\begin{aligned} \Delta^2 y_t &= \Delta(\Delta y_t) = (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) \\ &= y_t - 2\, y_{t-1} + y_{t-2} \\ &= 2\,\gamma + u_t - 2\, u_{t-1} + u_{t-2}\,. \end{aligned}$$

- First-difference Δy_t denotes *change* in y_t; second difference $\Delta^2 y_t$ denotes *acceleration* (i.e., "change in change") of y_t.

- Could have higher-order differences for more complicated (polynomial) trends.

Seasonal Differences:

For data with linear trend and K seasons (with separate intercept term for each season), use *seasonal difference*: if

$$y_t = \alpha + \delta \cdot t + \gamma_2 \cdot s_{2t} + \ldots + \gamma_K \cdot s_{Kt} + u_t,$$ then

$$\begin{aligned} \Delta_K y_t &\equiv y_t - y_{t-K} \\ &= K \cdot \delta + u_t - u_{t-K} . \end{aligned}$$

- For seasonal differences, lose first K observations, but don’t need to estimate K seasonal coefficients.

LAG OPERATOR NOTATION

The TSP statistical package (and others) uses *lag operator notation* to represent differencing. This notation uses the "backshift" or "lag" operator symbol B, which, when premultiplying a variable with a subscript, reduces its subscript by one:

$$B\ y_t \equiv y_{t-1}\ ,\quad B\ y_{t-1} = y_{t-2}\ ,\quad \text{etc.}$$

(In some texts, "$\mathscr{L}$" is used rather than "B" for this "lag" operator.) With this notation, the first-difference of y_t is written as

$$\begin{aligned} \Delta y_t &= y_t - y_{t-1} \\ &= y_t - B\ y_t \\ &= (1 - B)\ y_t\ . \end{aligned}$$

Similar arguments give second- and seasonal differences as

$$\Delta^2 y_t = (1 - B)^2 y_t\ ,\qquad \Delta_s y_t = y_t - B^s\ y_t = (1 - B^s)\ y_t\ .$$

DETRENDING VS. DIFFERENCING

Choice of "prefilter" depends upon assumptions about unobserved error terms. For simple trend model

$$y_t = \alpha + \delta \cdot t + u_t ,$$

detrending yields

$$u_t = y_t - \alpha + \delta \cdot t ,$$

while differencing yields

$$w_t = \Delta y_t = \delta + u_t - u_{t-1} .$$

Must decide whether time series model is more applicable to u_t or w_t. (Will be discussed later.)

"RECOLORING"

Once time-series model is fit to "prefiltered" version of y_t, then model must be retransformed to return to dependent variable of interest.

"Re-trending":

If detrended data satisfies, for example,

$$u_t = \beta\, u_{t-1} + \varepsilon_t + \theta\, \varepsilon_{t-1},$$

then "retrended" variable $y_t = \alpha + \delta\, t + u_t$ satisfies

$$y_t = [\alpha \cdot (1-\beta)+\delta] + \delta \cdot (1-\beta) \cdot t + \beta\, y_{t-1} + \varepsilon_t + \theta\, \varepsilon_{t-1}$$

which is used recursively to forecast y_s.

Integrating:

If data are differenced, e.g.,

$$\Delta y_t \equiv w_t = \beta w_{t-1} + \varepsilon_t + \theta \varepsilon_{t-1},$$

then "integrated" level variable y_t satifies

$$y_t = y_0 + \sum_{s=1}^{t} \Delta y_s$$

$$= y_0 + \beta \sum_{s=2}^{t} \Delta y_{s-1} + \sum_{s=1}^{t} \varepsilon_s + \theta \sum_{s=2}^{t} \varepsilon_{t-1},$$

which can be used to forecast recursively, once the starting value y_0 is known.

LECTURE / DISCUSSION

STATIONARY PROCESSES

TIME SERIES DATA AS STOCHASTIC PROCESSES

To construct procedures which work well "on average", we treat the observations on y_t as realizations of a collections of *random variables*, with a probability distribution of possible outcomes. We try to learn about the *population parameters* of this collection of random variables (called a *stochastic process*), such as the mean values, variances, and covariances, to construct prediction equations.

POPULATION MOMENTS AND STATIONARITY

Assuming y_t is univariate (i.e., a scalar, not a vector), its distribution can be summarized by the expectations (mean values), variances and covariances of the random variables.

Expectation (Mean) of y_t:

$$E[y_t] = \mu_t \,.$$

Variance of y_t:

$$\begin{aligned} Var(y_t) &= E[(y_t - \mu_t)^2] = E[(y_t - \mu_t)(y_t - \mu_t)] \\ &= \sigma_t^2 \equiv \sigma_{tt} \,. \end{aligned}$$

Covariance of y_t and y_{t-s}:

$$\begin{aligned} Cov(y_t, y_{t-s}) &= E[(y_t - \mu_t)(y_{t-s} - \mu_{t-s})] \\ &\equiv \sigma_{t,t-s} \,. \end{aligned}$$

(Weak or Covariance) Stationarity of y_t:

The time series variable y_t is called *stationary* (more precisely, *weakly stationary* or *covariance stationary*) if the mean of y_t is constant,

$$\mu_t = \mu \quad \text{for all integer } t ,$$

and the covariance of y_t and y_{t-s} only depends on s ,

$$\sigma_{t,t-s} = \gamma(s)$$

for all integers t and s, some function $\gamma(\cdot)$.

Strong Stationarity of y_t:

The time series variable y_t is called *strongly stationary* if the joint distribution of

$$\{y_t, y_{t+1}, \ldots, y_{t+s}\}$$

is the same as the joint distribution of

$$\{y_1, y_2, \ldots, y_{s+1}\}$$

for all integers t and s. Clearly, if y_t is strongly stationary and the mean and variance of y_t exists, then y_t is (weakly) stationary. Strong stationarity is sometimes assumed if higher moments of y_t (reflecting skewness or kurtosis) are of interest.

Meaning and Importance of "Stationarity":

- The means, variances, and covariances of the y_t process (or the entire joint distribution of the process, for strong stationarity) don’t depend upon its "starting point."

- With stationarity, past data can be used to determine relationships across time periods, and those relationships will be valid for future observations.

- Without stationary, need strong conditions changes in mean, variance and covariances over time - otherwise, too many unknown parameters per observation.

- Nonstationarity in mean easier to deal with (via detrending or other linear regression methods) than nonstationarity in autocorrelation structure.

AUTOCOVARIANCE AND AUTOCORRELATION FUNCTION

Autocovariance Function:

$\gamma(s) = \mathrm{Cov}(y_t, y_{t-s})$

$= \mathrm{Cov}(y_{t-s}, y_t)$, by usual properties of covariances

$= \mathrm{Cov}(y_t, y_{t+s})$, by stationarity of y_t

$= \gamma(-s)$, so

$\gamma(s) = \gamma(-s) = \gamma(|s|)$.

Also,

$\gamma(0) = \mathrm{Cov}(y_t, y_t)$

$= \mathrm{Var}(y_t)$

$\equiv \sigma^2$

$= \mathrm{Var}(y_{t-s})$, again by stationarity of y_t.

Autocorrelation Function (Theoretical ACF):

$$\rho(s) = \text{Correlation}(y_t, y_{t-s}) = \text{Correlation}(y_t, y_{t+s})$$

$$= \frac{\text{Cov}(y_t, y_{t-s})}{\sqrt{\text{Var}(y_t)\cdot\text{Var}(y_{t-s})}}$$

$$= \frac{\gamma(s)}{\sqrt{\gamma(0)\cdot\gamma(0)}}$$

$$= \frac{\gamma(s)}{\gamma(0)} \; .$$

Remarks:

- By the usual properties of correlations, $|\rho(s)| \leq 1$.

- The mean μ, variance $\sigma^2 = \gamma(0)$, and autocorrelation function $\rho(s)$ capture the essential information about typical value, dispersion, and association over time for the y_t process, and are the basis for predicting future y_t values given their current values.

ESTIMATION OF MOMENTS

Sample:

$\{y_1, y_2, \ldots, y_T\}$, observed.

Sample Mean:

$$\bar{y} = \frac{1}{T} \sum_{t=1}^{T} y_t \, .$$

Sample Autocovariance Function:

$$c(s) = \frac{1}{T} \sum_{t=s+1}^{T} (y_t - \bar{y})(y_{t-s} - \bar{y}) \quad \text{for } s = 0, 1, \ldots, T-1$$

$$c(s) = 0 \qquad \text{for } s = T, T+1, \ldots.$$

Sample ACF (Autocorrelation Function):

$$r(s) = \frac{c(s)}{c(0)} \qquad \text{for} \quad s = 0, 1, \ldots, T - 1$$

$$r(s) = 0 \qquad \text{for} \quad s = T, T + 1, \ldots.$$

EXAMPLE - TYPICAL COMPUTER GRAPH OF SAMPLE ACF

```
  -1.0      -0.6      -0.2       0.2       0.6       1.0
    +---------+---------+----0----+---------+---------+
 1  |                  +     |     +        *         |0.60
 2  |                  +     |     +  *               |0.36
 3  |                +       |    *  +                |0.22
 4  |                +       |  *    +                |0.13
 5  |               +        | *      +               |0.07
 6  |               +        |*       +               |0.04
 7  |               +        *        +               |0.02
 8  |               +        *        +               |0.01
```

Vertical axis (left-hand side) = Values of s

Horizontal axis (top) = Possible values of r(s)

Asterisks (*) = Actual values of r(s) for data set

(Also given numerically down right-hand side of graph)

Plus-signs (+) = Standard error bounds for r(s)

ERGODICITY

Large-Sample Behavior of Sample Moments:

We want the estimators $\bar{y}$, $\hat{\sigma}^2 = c(0)$, and $r(s)$ to tend to the values they estimate (respectively, μ, σ^2, and $\rho(s)$) in probabilty as the sample size increases, i.e., we want

$$\mathrm{plim}(\bar{y}) = \mu \, ,$$

$$\mathrm{plim}(\hat{\sigma}^2) = \sigma^2 \, , \quad \text{and}$$

$$\mathrm{plim}(r(s)) = \rho(s)$$

as $T \to \infty$. This will happen if the dependence between y_t and y_{t+s} vanishes as $s \to \infty$; if it does happen, then the y_t process is called *ergodic.*

Examples of Non-Ergodic Processes:

(1) $y_t = z$ for all t, where $z \sim \mathcal{N}(\mu, \sigma^2)$. Here

$$\rho(s) = 1 \quad \text{for all } s.$$

(2) $y_t = z_1$ if t is even,
$y_t = z_2$ if t is odd,

where z_1 and z_2 are independently distributed $\mathcal{N}(\mu, \sigma^2)$. For this process,

$$\rho(s) = 1 \quad \text{if } s \text{ is even,}$$

$$\rho(s) = 0 \quad \text{if } s \text{ is odd.}$$

Necessary Condition for Ergodicity:

$$\rho(s) \to 0 \quad \text{as} \quad s \to \infty \,.$$

- Most of the models considered below will be ergodic, though some will require data transformations for this.

WORKSHOP

DETRENDING, SEASONAL ADJUSTMENT, AUTOCORRELATION

We start with the utility data set of the previous workshop; we will check to see if combining detrending or differencing with the previous simple forecasting approaches improves the performance of the forecasts.

1. In the **wedaft** subdirectory are copies of the **utildata.xls** data set and a copy of the previous **example5.tsp** program, now called **utility.tsp**. We'll first try logarithmic transformations for variables, by changing the definition of the dependent variable to

   ```
   y = log(rev);
   ```

   which makes the log of revenues (or cost or price) the dependent variable for analysis. Before calculating the MSEs for the various forecasting methods, we should transform the data back from logs to levels, by inserting commands like

   ```
   yhat = exp(yhat);
   ```

   with similar definitions for "yma" and "yewma" (**utility1.tsp**).

   How do the MSE's here compare with their (non-logarithm) counterparts?

2. Next, we’ll apply the moving-average and EWMA procedures to detrended versions of the dependent variables. After the **olsq** command, insert the lines

   ```
   u = @res;
   uewma = u;
   ```

   This saves the detrended residuals in the variable "u". Then duplicate the lines for "yma" and "yewma" using "uma", etc. Before plotting the forecasts and calculating MSEs, "retrend" by inserting commands like

   ```
   yma2 = yhat + uma;
   ```

   to put the trend term back in (same for "yewma2" - see **utility2.tsp)**

   Does this "prefiltering" approach improve the fit for the moving-average procedures, as measured by the MSE or the graphs of the forecasts?

3. An alternative to detrending is *first-differencing*; given the results of the previous workshop (where the "best" value of the EWMA weight p was around one, corresponding to predicting y_t by its lagged value y_{t-1}), differencing might be a better approach to achieve stationarity. To perform the statistical analysis on the differences of y_t, add a line defining

   ```
   dy = res - res(-1);
   ```

   with analogous changes for the other variables. Before plotting the forecasts and calculating MSEs, you can "integrate" the differenced variables over the post-sample period, using commands like

   ```
   smpl 60:1 89:4;
   yma=y;
   smpl 90:1 92:4;
   yma=yma(-1)+dyma;
   ```

   for the moving-average procedure (**utility3.tsp**). As always, the criterion for evaluating the success of the prefiltering approach will be the post-estimation MSE of the forecasts.

4. By this time, we’ve applied a lot of preliminary data transformations and smoothing procedures to the revenue, cost, and price variables from the utility data set. If you had to pick a single prefiltering/forecasting method combination for all three of the series, what would it be? Your choice might be based only on the MSE results, or can involve other considerations like simplicity of the approach or interpretability of the plotted values. In any case, a good way to finish up would be to use your "favorite" combination to get post-estimation forecasts and MSEs for each of the three variables.

5. Now we'll look at a very seasonal data set on total housing starts in the U.S., a monthly series which runs from January 1986 (86:1) to October 1995 (95:10). The data are in the Excel-file **housing.xls** in the **wedaft** directory, in a variable named **starts**. After changing the **freq** (to **freq m)**, **smpl**, and **load** commands, you can create monthly dummies by inserting

   dummy;

   after the **trend** command; this generates the TSP variables **m1-m12**, monthly dummy variables which can be substituted for the constant term **c** in the **olsq** and **forcst** commands to control for systematic seasonal variation in housing starts.

   We will still want to reserve some observations for post-sample forecasting; to judge the effect of the seasonal dummies, we should have at least a year's worth of post-estimation observations (e.g., **smpl 94:11 95:10;**).

   The modified version of the program for this new data set is **utility5.tsp**, in case you're having trouble getting all the **smpl** statements straight. You might try the logarithmic transformation if you have extra time.

6. Finally, we’ll inspect the autocorrelation function of the housing start series, using the **bjident** command (which is an acronym for "Box-Jenkins Time Series Identification", to be discussed more fully in the next sessions). After loading in the data, the TSP command

   ```
   bjident(ndiff=1,nsdiff=1) starts;
   ```

   will plot autocorrelations for the "starts" variable and up to one regular and one seasonal difference (**utility6.tsp**). (We could look at higher differences by adjusting the options in the parentheses after the **bjident** command.)

   The estimated autocorrelation functions should vary a lot across different differencing procedures; for example, without seasonal differences, the estimated autocorrelations should be pretty large at the "seasonal lags" (around $s = 12$ months), but not with the seasonal differences. Note that TSP uses the *lag operator notation* to indicate the degree of differencing.

   The **bjident** command will be used extensively in the upcoming discussions on fitting autoregressive and ARIMA models.

DISCUSSION OF WORKSHOP RESULTS

LECTURE / DISCUSSION

UNIVARIATE AUTOREGRESSIONS

KEY FEATURES

- Model current value of y_t as a linear combination of its own past values (maybe including a deterministic trend) plus an unpredictable error term (called *white noise*).

- As a forecasting model, combines best features of moving average and EWMA approaches (namely, adaptation of forecasts to recent values of y_t) with best features of deterministic regression approaches (matching prediction equation to historical data using least squares).

- Distinction between stationary models (without trends or differencing) and nonstationary versions - can test for *trend stationarity* versus *difference stationarity* ("unit roots").

EXAMPLES OF AUTOREGRESSIVE MODELS

Stationary Models:

- White Noise Process ("WN(σ^2)").
- First-Order Autoregressive Process ("AR(1)").
- Higher-Order Autoregressive Process ("AR(p)").

Nonstationary Models:

- Random Walk (with drift).
- First-Order Autoregressive with Trend.
- Higher-Order Autoregressive with Trend.

WHITE NOISE PROCESS

A process y_t is called a *white noise* process, denoted

$y_t \sim WN(\sigma^2)$, if

$E[y_t] = 0$,

$Var[y_t] = \sigma^2$,

and

$\gamma(s) = \rho(s) = 0$ if $s \neq 0$.

Remarks:

- White noise is simplest example of stationary, ergodic process; basic "building block" of time series models.

- If y_t is independently and identically distributed (i.i.d.) across t, not just uncorrelated across t, process is strongly stationary, not just covariance stationary.

- White noise is unpredictable given the past; typically used to represent unpredictable component ("noise") of observable time series.

- Typically use "ε_t" (or other Greek letters) for white noise, since it usually represents unobservable error term.

FIRST-ORDER AUTOREGRESSIVE PROCESS ("AR(1)")

A *first-order autoregressive process*, denoted

$$y_t \sim \mathrm{AR}(1)$$

satisfies the equation

$$y_t = \alpha + \beta\, y_{t-1} + \varepsilon_t\,,$$

where

$$\varepsilon_t \sim \mathrm{wn}(\tau^2)$$

for some fixed (usually unknown) α, β, and τ^2.

Mean of AR(1):

$$E[y_t] = E[\alpha + \beta\, y_{t-1} + \varepsilon_t]$$

$$= \alpha + \beta\, E[y_{t-1}] + E[\varepsilon_t]$$

$$= \alpha + \beta\, E[y_t] \qquad \text{by stationarity,}$$

so

$E[y_t] \equiv \mu = \alpha/(1 - \beta)$, assuming $\beta \neq 1$.

Variance of AR(1):

$$\mathrm{Var}(y_t) = \mathrm{Var}(\alpha + \beta\, y_{t-1} + \varepsilon_t)$$

$$= \beta^2\, \mathrm{Var}(y_{t-1}) + \mathrm{Var}(\varepsilon_t) + 2\beta\, \mathrm{Cov}(y_{t-1}, \varepsilon_t)$$

$$= \beta^2\, \mathrm{Var}(y_t) + \mathrm{Var}(\varepsilon_t)\ ,$$

since $\mathrm{Cov}(y_{t-1}, \varepsilon_t) = 0$ and $\mathrm{Var}(y_t) = \mathrm{Var}(y_{t-1})$ by stationarity. So

$$\mathrm{Var}(y_t) \equiv \sigma^2 = \beta^2 \sigma^2 + \tau^2 = \tau^2/(1 - \beta^2)\ ,$$

assuming $|\beta| < 1$.

Autocovariance Function of AR(1):

$$\mathrm{Cov}(y_t, y_{t-s}) = \mathrm{Cov}(\alpha + \beta\, y_{t-1} + \varepsilon_t, y_{t-s})$$

$$= \beta\, \mathrm{Cov}(y_{t-1}, y_{t-s}) + \mathrm{Cov}(\varepsilon_t, y_{t-s})$$

$$= \beta\, \mathrm{Cov}(y_t, y_{t-s+1}) ,$$

so

$$\gamma(s) \equiv \mathrm{Cov}(y_t, y_{t-s})$$

$$= \beta\, \gamma(s-1)$$

$$= \beta^2\, \gamma(s-2) \quad \ldots$$

$$= \beta^s\, \gamma(0) = \beta^s \sigma^2.$$

ACF of AR(1):

$\rho(s) = \beta^{s}$,   s = 0, 1, 2, ...

Remarks:

- ACF declines geometrically to zero as lag length s increases; dependence lasts across all time periods.

- If sample ACF "tails off" geometrically, AR(1) may be appropriate model.

- In empirical work, AR(1) is common model for change in aggregate GNP, consumption, capital stock, etc.

FIGURE - ACF FOR AR(1), POSITIVE COEFFICIENT

Model: $y_t = 0.6\ y_{t-1} + \varepsilon_t$

```
    -1.0      -0.6      -0.2       0.2       0.6       1.0
      +---------+---------+----0----+---------+---------+
  1   |                        |              *         |0.60
  2   |                        |        *               |0.36
  3   |                        |    *                   |0.22
  4   |                        |  *                     |0.13
  5   |                        | *                      |0.07
  6   |                        |*                       |0.04
  7   |                        *                        |0.02
  8   |                        *                        |0.01
```

FIGURE - ACF FOR AR(1), NEGATIVE COEFFICIENT

Model: $y_t = -0.6\ y_{t-1} + \varepsilon_t$

```
  -1.0      -0.6      -0.2       0.2       0.6       1.0
    +---------+---------+----0----+---------+---------+
  1 |                        |              *         |-0.60
  2 |               *        |                        | 0.36
  3 |                        |    *                   |-0.22
  4 |                     *  |                        | 0.13
  5 |                        | *                      |-0.07
  6 |                       *|                        | 0.04
  7 |                        *                        |-0.02
  8 |                        *                        | 0.01
```

HIGHER-ORDER AUTOREGRESSIVE PROCESSES ("AR(P)")

A more complex autocorrelation pattern is generated from a p^{th}-*order autoregressive process*, which is denoted

$$y_t \sim AR(p)$$

and satisfies

$$y_t = \alpha + \beta_1 y_{t-1} + \beta_2 y_{t-2} + \dots \beta_p y_{t-p} + \varepsilon_t,$$

where

$$\varepsilon_t \sim wn(\tau^2)$$

for some α, β_1, ..., β_q and τ^2.

Mean and Variance of AR(p):

Simple to calculate formula for mean of y_t:

$$
\begin{aligned}
E[y_t] &\equiv \mu \\
&= \alpha + \beta_1 E[y_{t-1}] + \beta_2 E[y_{t-2}] + \ldots \beta_p E[y_{t-p}] + E[\varepsilon_t] \\
&= \alpha + (\beta_1 + \beta_2 + \ldots \beta_p)\, \mu
\end{aligned}
$$

assuming y_t is stationary, which implies

$$\mu \equiv E[y_t] = \frac{\alpha}{1 - \beta_1 - \ldots - \beta_p} .$$

More difficult to calculate variance and autocovariances of AR(p) process, since y_t values are correlated. However, can show that the ACF of y_t satisfies the following formula (known as the *Yule-Walker* equations), which holds for all lag lengths $s = 1, 2, \ldots$:

$$\rho(s) = \beta_1 \rho(s-1) + \beta_2 \rho(s-2) + \ldots + \beta_p \rho(s-p) .$$

Remarks:

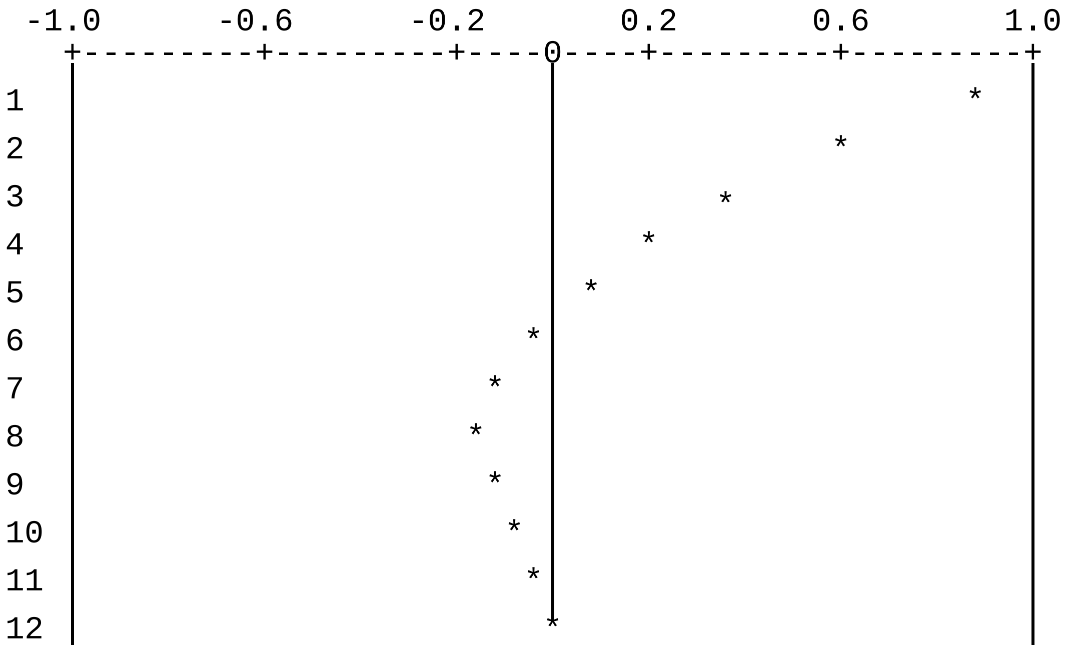
- Stationarity of AR(p) (i.e., constant mean, variance, and autocovariances) requires $|\beta_1 + ... + \beta_p| < 1$, but this condition is not sufficient (unless p = 1).

- Magnitude of ACF for AR(p) declines exponentially to zero; ACF can either decline monotonically or cyclically, depending on signs and relative magnitudes of $\beta_1, ..., \beta_p$ (see example below).

- Yule-Walker equations say that ACF for AR(p) follows the same autoregressive pattern that the time series y_t does - obtained by calculating $\mathrm{Cov}(y_t, y_{t-s})$ after plugging in formula for y_t.

FIGURE - ACF FOR AR(P), P = 2

Model: $y_t = 1.3\ y_{t-1} - 0.5\ y_{t-1} + \varepsilon_t$



ESTIMATION OF AUTOREGRESSIVE MODELS

Since a p^{th}-order autoregression expresses y_t as a linear function of $y_{t-1}, \ldots, y_{t-p}$ plus an error term that is uncorrelated over time (and therefore uncorrelated with $y_{t-1}, \ldots, y_{t-p}$), we can use *ordinary least squares* (OLS) to estimate the unknown coefficients: we minimize

$$\text{SSE} = \sum_{t=p+1}^{T} (y_t - a + b_1\, y_{t-1} + b_2\, y_{t-2} + \ldots\, b_p\, y_{t-p})^2$$

over $a, b_1, \ldots, b_p$ to get the estimated coefficients $\hat{\alpha}, \hat{\beta}_1, \ldots, \hat{\beta}_p$.

Remarks:

- The least-squares estimates will be consistent - that is, their distribution shrinks to the true values as the sample size increases.

- Since the true parameters of the process are unknown, we use the estimated parameters in any prediction formulas.

CHOICE OF AUTOREGRESSION ORDER

Given p, can fit coefficients of AR(p) model by least squares. Need *model selection criterion*, i.e., need procedure to determine best choice of p. Several different approaches have been proposed:

Information Criteria:

Given fitted model for particular p, can construct summary statistics which reflect trade-offs between higher goodness-of-fit (i.e., smaller MSE) and lower number of parameters. Two such summary measures are *Akaike's Information Criterion* (AIC) and *Schwarz's Bayesian Information Criterion* (BIC). We prefer models with higher "information" - thus, we can choose p to maximize AIC and/or BIC.

Stepwise Autoregressions:

Starting from $p = 1$, fit AR(1), AR(2), ..., AR(p+1), etc., until highest-order coefficient $\hat{\beta}_p$ is insignificant using standard t-test. This is closely related to use of *sample partial autocorrelation function* (PACF) to determine p.

PARTIAL AUTOCORRELATION FUNCTION (PACF)

For *any* stationary process y_t (not just autoregressive process), could define "approximate AR(s) coefficients" $\beta_1^*, \beta_2^*, \ldots, \beta_s^*$ to solve first s Yule-Walker equations:

$$\rho(1) = \beta_1^* \rho(0) + \beta_2^* \rho(-1) + \ldots + \beta_p^* \rho(1-s) ,$$
$$\rho(2) = \beta_1^* \rho(1) + \beta_2^* \rho(0) + \ldots + \beta_p^* \rho(2-s) ,$$
$$\ldots$$
$$\rho(s) = \beta_1^* \rho(s-1) + \beta_2^* \rho(s-1) + \ldots + \beta_p^* \rho(0) .$$

For every possible s = 1, 2, 3, ..., the value of β_s^* that solves these equations is called the *theoretical partial autocorrelation function* ("PACF"):

$$\pi(s) \equiv \beta_s^* .$$

Behavior of Partial Autocorrelation Function:

If y_t is an AR(p) process, then the PACF "cuts off" after p lags, i.e.,

$$\pi(s) \equiv 0 \quad \text{for} \quad s = p + 1,\ p + 2,\ .....$$

Estimation of Partial Autocorrelation Function:

Either plug estimated autocorrelations r(s) into Yule-Walker equations and solve, or run stepwise autoregressions, and set

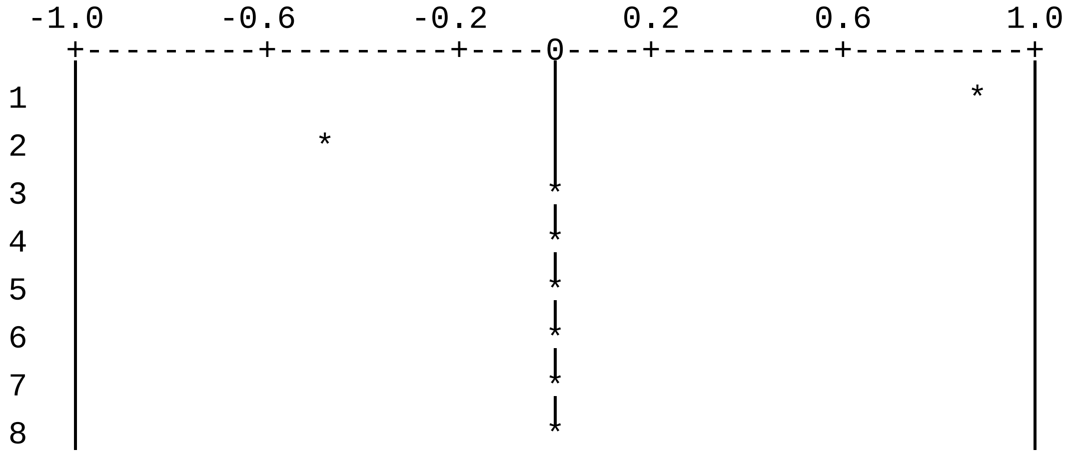
$$\hat{\pi}(s) = \hat{\beta}_s\ .$$

Selection of Autoregressive Order using PACF:

Can choose p corresponding to largest statistically significant value of PACF - this is same approach as stepwise regression, but uses canned PACF program rather than series of autoregressions.

FIGURE - PACF FOR AR(P), P = 2

Model: $y_t = 1.3\ y_{t-1} - 0.5\ y_{t-1} + \varepsilon_t$



Remark:

This is the *true* PACF - estimates would differ, depending upon the particular data observed.

PREDICTION USING AUTOREGRESSIONS

Prediction with an autoregressive process is straightforward; given the parameters α, β_1, ..., β_p (or estimates of them), can predict next period's value y_{T+1} by using autoregression formula, with predicted error $\hat{\varepsilon}_{T+1} = E[\varepsilon_{T+1}] = 0$:

$$\hat{y}_{T+1} = \alpha + \hat{\beta}_1 y_{T-1} + \hat{\beta}_2 y_{T-2} + \ldots \hat{\beta}_p y_{T-p} .$$

For longer-term predictions (more than one time-period ahead), this formula can be used recursively, plugging in shorter-term forecasted values wherever necessary (the "chain rule of forecasting").

NON-STATIONARY EXAMPLE: RANDOM WALK (WITH DRIFT)

The process y_t is called a *random walk* if the first difference of y_t is a white noise process; it is a *random walk with drift* if the change in y_t is the sum of a constant and a white noise process.

Random Walk:

$$y_t = y_{t-1} + \varepsilon_t , \quad \text{for} \quad \varepsilon_t \sim WN(\tau^2) , \quad \text{i.e.,}$$

$$y_t - y_{t-1} = \Delta y_t \sim WN(\tau^2) .$$

Random Walk with Drift:

$$y_t = \alpha + y_{t-1} + \varepsilon_t , \quad \text{for} \quad \varepsilon_t \sim WN(\tau^2) , \quad \text{i.e.,}$$

$$y_t - y_{t-1} - \alpha = (\Delta y_t - \alpha) = WN(\tau^2) .$$

Nonstationarity of Random Walk:

As for the AR(1) case, we can calculate

$$E[y_t] = E[\alpha + y_{t-1} + \varepsilon_t]$$

$$= \alpha + E[y_{t-1}]$$

and

$$\mathrm{Var}(y_t) = \mathrm{Var}[\alpha + y_{t-1} + \varepsilon_t]$$

$$= \mathrm{Var}(y_{t-1}) + \mathrm{Var}(\varepsilon_t)$$

$$= \mathrm{Var}(y_{t-1}) + \tau^2 ,$$

so it is impossible to have $E[y_t] = E[y_{t-1}]$ and $\mathrm{Var}(y_t) = \mathrm{Var}(y_{t-1})$ if α and/or τ^2 are nonzero.

"Initial Condition" Representation:

If starting value at $t = 0$, y_0, taken as fixed, then repeated substitution yields

$$y_t = \alpha + (\alpha + y_{t-2} + \varepsilon_{t-1}) + \varepsilon_t \quad \ldots$$

$$= y_0 + \alpha \cdot t + \left\{ \sum_{s=1}^{t} \varepsilon_s \right\} .$$

Moments of Random Walk with Drift:

$$E[y_t] = y_0 + \alpha \cdot t \ ,$$

$$\mathrm{Var}(y_t) = \tau^2 \cdot t \ ,$$

$$\mathrm{Cov}(y_t, y_s) = \tau^2 \cdot \min\{t, s\} \ .$$

Remarks:

- Random walk can be viewed as "limiting case" of AR(1) when autoregressive parameter β approaches one.

- Theoretical ACF of random walk doesn’t exist; sample ACF $r(s)$ for random walk will decline *slowly* (that is, approximately linearly, not geometrically) to zero as lag length s increases.

- In empirical work, random walk is common model for level of stock prices, commodity prices, etc., for high frequency (e.g., daily) data. For such short time intervals (e.g., daily data), drift parameter approximately zero.

EXAMPLE - STOCK PRICES

Standard and Poor’s monthly average index of 500 stock prices, January 1960 through February 1996.

NONSTATIONARY EXAMPLE: AR(1) WITH TREND

$$y_t = \alpha + \beta y_{t-1} + \delta t + \varepsilon_t$$

Remarks:

- Combination of deterministic regression model and AR(1); model is called *trend stationary* if $\beta < 1$.

- Includes standard AR(1) ($\delta = 0$) and random walk with drift ($\delta = 0$, $\beta = 1$) as special cases.

- Could also include "seasonal dummies" if data were monthly or quarterly.

- Like random walk, sample ACF will decline slowly (linearly in lag length s) if trend parameter δ is nonzero.

NONSTATIONARY EXAMPLE: AR(P) WITH TREND

$$y_t = \alpha + \beta_1 y_{t-1} + \ldots + \beta_p y_{t-p} + \delta t + \varepsilon_t$$

Remarks:

- Can rewrite model as

$$\Delta y_t = \delta + \gamma_0 y_{t-1} + \gamma_1 \Delta y_{t-1} + \ldots + \gamma_p \Delta y_{t-p} + \varepsilon_t,$$

where $\gamma_1 = \beta_1 + \ldots + \beta_p - 1$. That is, the change in y_t is p^{th}-order autoregressive, but with the *level* of y_{t-1} also included as a regressor.

GRAPHICAL DETECTION OF NONSTATIONARITY

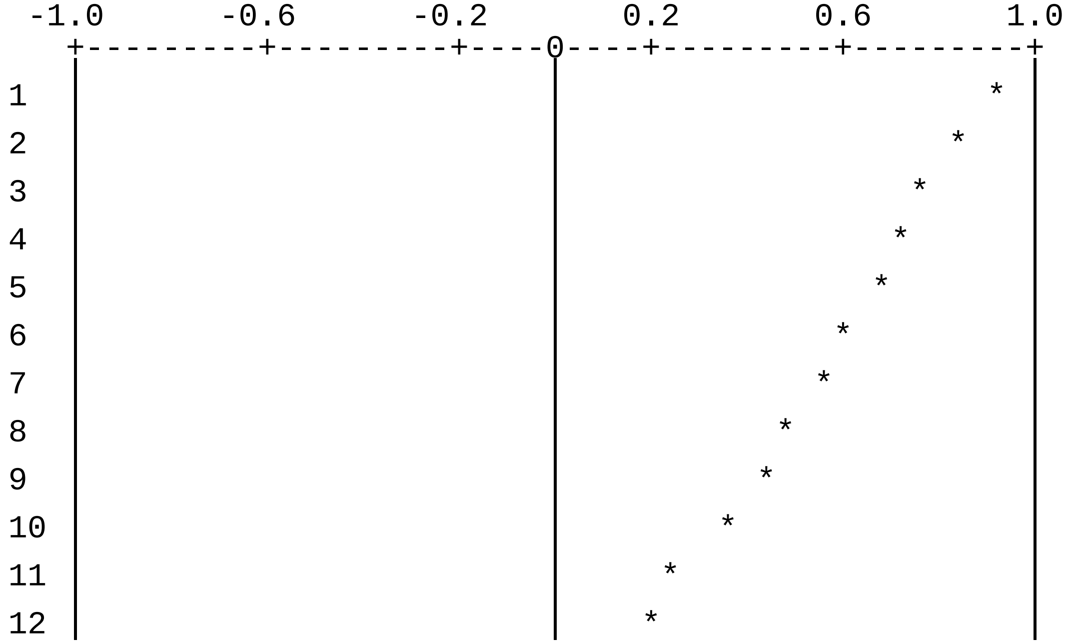
"Eyeball Metric":

If graph of y_t against t "starts low and ends high," or vice versa, then detrending or differencing is needed. If same holds true for detrended/differenced data, then higher-order detrending or differencing required.

Inspection of Sample ACF:

If sample ACF for data declines slowly (linearly) to zero, then detrending/differencing needed. If it converges to zero quickly, no trend removal is indicated.

FIGURE - SAMPLE ACF FOR NONSTATIONARY SERIES



DICKEY-FULLER TEST OF RANDOM WALK

To distinguish between a random walk (which requires differencing) and a trend-stationary a formal test of the null hypothesis that the variable y_t is a random walk,

$$y_t = y_{t-1} + \varepsilon_t, \qquad \varepsilon_t \sim WN(\sigma^2) ,$$

versus the alternative hypothesis that the data is AR(1) plus trend,

$$y_t = \alpha + \beta y_{t-1} + \delta t + \varepsilon_t .$$

The null hypothesis is a special case of this model, with $\beta = 1$ (and $\delta = 0$).

Null and Alternative Hypotheses:

H_o: $\beta = 1$ versus H_A: $\beta < 1$ (trend stationary).

Test Statistic:

Can use usual t-test: reject H_o if

$$\mathcal{T} = \frac{\hat{\beta} - 1}{SE(\hat{\beta})} < \text{critical value.}$$

Critical Values:

Because y_t is nonstationary under null hypothesis, cannot use usual t-table to get critical values. Appropriate critical values are given in the following table.

	1%	2.5%	5%	10%
Random walk	-3.96	-3.66	-3.41	-3.13
Standard	-2.57	-2.33	-1.96	-1.65

Remarks:

- Dickey-Fuller critical values are bigger than standard t critical values; need more negative $\mathcal{T}$ to reject null hypothesis.

- Rejection of null hypothesis indicates detrending, not first-differencing, is appropriate trend-removal method; failure to reject H_o indicates opposite approach.

- Could also construct F-test of joint null hypothesis that $\beta = 1$ and $\delta = 0$, with a similar inflation of critical values.

- To test H_o: $\Delta y_t \sim AR(p)$ versus H_A: $y_t \sim AR(p)$ plus trend, can add p lags of Δy_t as regressors and do the same t-test. (This is called the *augmented Dickey-Fuller test*).

WORKSHOP

UNIVARIATE AUTOREGRESSIONS

Again, we start with the utility data set of the previous workshop; we will now combine detrending or differencing with autoregressive fits, which might be viewed as "weighted moving averages," with weights that are estimated from the data.

The **utildata.xls** data set has been copied into the **thumorn** subdirectory.

1. To start, apply the **bjident** command to each of the revenue, cost, and price variables (or their logarithms, if you prefer) to view the estimated ACFs and PACFs for the levels (logs) of the variables and their differences (**autoreg.tsp**).

   By inspecting the ACFs and PACFs for the variables, does it seem that differencing/detrending is needed? What order of autoregression is indicated for the (detrended or differenced) variable?

2. Try using least squares to estimate trend-stationary AR(p) models for the variables, using the **olsq** command. For example, a third-order autoregression, with trend, for the **y** variable would use

   ```
   olsq y c t y(-1)-y(-3);
   ```

   with a similar modification of the **forcst** command. You can use the earlier PACF information to determine the order p of the autoregression, or, alternatively, can fit successively-higher order models until the t-statistics for the additional terms become insignificant. (**autoreg2.tsp**)

   How do the autoregressive forecasts compare with the moving-average and EWMA predictors for these variables, in terms of predictive mean squared error?

3. Again, an alternative to detrending is *first-differencing*; by defining the dependent variable as, say,

```
dy = res - res(-1);
```

you can run similar autoregressions for the differenced data, omitting the trend variable "t". And again, before evaluating the forecasting performance of the procedures, you should "integrate" to get predictions for the un-differenced variable by inserting

```
smpl 89:4 89:4;
yint=y;
smpl 90:1 92:4;
yint=yint(-1)+dyhat;
```

before the **plot** command. (**autoreg3.tsp**)

4. To test the null hypothesis that first-differencing is appropriate, against the alternative of trend stationarity, you can use the TSP command **unit** command,

```
unit y;
```

which prints out a number of summary statistics, but concludes with the Dickey-Fuller test and an alternative test (called the *weighted symmetric* test) for the "y" variable. By applying this procedure to differenced versions of the revenue, cost, and price variables, you can determine how many differences are needed before rejecting the null hypothesis that yet another difference is needed to achieve stationarity. (**autoreg4.tsp**)

5. Now let's look at some data from financial markets. In the file **returnsp.xls** in the **thumorn** directory are some monthly financial variables from January, 1960 (60:1) to February, 1996 (96:2). Two variables which are in the data set, and are of obvious interest to financial planners, are "snp", the monthly Standard and Poor's 500 stock index, and "r3", the three-month Treasury bill rate. Our object is to see whether these behave like trend-stationary variables, random walks, or difference-stationary autoregressions.

   For both these variables, we can use **bjident(ndiff=1,nsdiff=1)** to plot the ACFs and PACFs for their levels, first-, and seasonal-differences. Then, constructing the first-difference

   ```
   dy = y - y(-1);
   ```

   we do Dickey-Fuller tests on "y" and "dy" using the **unit** command.

   Do the results suggest whether each variable is trend-stationary, a random walk, or autoregressive in first differences? (A program which does this analysis is **autoreg5.tsp**).

6. Finally, we can fit trend-stationary and difference-stationary AR models to "snp", and then compare their post-sample forecasting performance, as we did in part 3 above (**autoreg6.tsp**).

DISCUSSION OF WORKSHOP RESULTS

LECTURE / DISCUSSION

UNIVARIATE ARIMA MODELS

KEY FEATURES

- Combine autoregressive models with more complex error terms, called *moving average processes*.

- Object is to get models with fewer parameters than pure autoregressions, but with equally-complex dynamic patterns.

- Model selection methods, called *time series identification* methods, use autocorrelation and partial autocorrelation functions (ACF and PACF).

- After model identification and fitting, adequacy of model is checked using diagnostic tests.

EXAMPLES OF ARIMA MODELS

- First-order Moving Average Process ("MA(1)")
- Higher-order Moving Average Process ("MA(q)")
- Mixed Autoregressive-Moving Average Process ("ARMA(p,q)")
- Integrated Processes ("ARIMA(p,d,q)")

FIRST-ORDER MOVING AVERAGE PROCESS ("MA(1)")

$$y_t \sim \mathrm{MA}(1)$$

if

$$y_t = \mu + \varepsilon_t + \theta\,\varepsilon_{t-1},$$

where

$$\varepsilon_t \sim \mathrm{wn}(\tau^2)$$

for some fixed (usually unknown) μ, θ, and τ^2.

Mean and Variance of MA(1):

$$E[y_t] = E[\mu + \varepsilon_t + \theta \varepsilon_{t-1}]$$

$$= \mu + E[\varepsilon_t] + \theta E[\varepsilon_{t-1}]$$

$$= \mu .$$

$$Var(y_t) = Var(\mu + \varepsilon_t + \theta \varepsilon_{t-1})$$

$$= Var(\varepsilon_t) + \theta^2 Var(\varepsilon_{t-1}) ,$$

since $Cov(\varepsilon_t, \varepsilon_{t-1}) = 0$. So

$$\sigma^2 = \tau^2 + \theta^2 \tau^2 = \tau^2 (1 + \theta^2) .$$

Autocovariance Function of MA(1):

$$\mathrm{Cov}(y_t, y_{t-1}) = \mathrm{Cov}(\mu + \varepsilon_t + \theta\,\varepsilon_{t-1}, \mu + \varepsilon_{t-1} + \theta\,\varepsilon_{t-2})$$

$$= \theta\,\mathrm{Var}(\varepsilon_{t-1})$$

$$= \theta\,\tau^2 .$$

For all other lags,

$$\mathrm{Cov}(y_t, y_{t-s}) = 0 \quad \text{if} \quad s = 2, 3, \ldots.$$

ACF of MA(1):

$$\rho(s) = 1 \qquad \text{if} \quad s = 0$$

$$= \theta/(1 + \theta^2) \quad \text{if} \quad s = 1$$

$$= 0 \qquad \text{if} \quad s = 2, 3, 4, \ldots.$$

Remarks:

- ACF declines to zero after first time period; dependence only for adjacent periods.

- If sample ACF "cuts off" after one lag, MA(1) may be appropriate model.

- Partial ACF "tails off" to zero exponentially after first lag.

FIGURES - ACF AND PACF FOR MA(1)

Model: $y_t = \varepsilon_t + 0.6\ \varepsilon_{t-1}$

ACF:

```
  -1.0      -0.6      -0.2       0.2       0.6       1.0
    +---------+---------+----0----+---------+---------+
 1  |                        |              *         |
 2  |                        *                        |
 3  |                        *                        |
 4  |                        *                        |
 5  |                        *                        |
 6  |                        *                        |
 7  |                        *                        |
 8  |                        *                        |
```

PACF:

```
  -1.0      -0.6      -0.2       0.2       0.6       1.0
    +---------+---------+----0----+---------+---------+
1   |                        |              *         |
2   |                        |        *               |
3   |                        |    *                   |
4   |                        |  *                     |
5   |                        | *                      |
6   |                        |*                       |
7   |                        *                        |
8   |                        *                        |
```

HIGHER-ORDER MOVING AVERAGE PROCESSES ("MA(Q)")

$$y_t \sim \mathrm{MA}(q)$$

if

$$y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots \theta_q \varepsilon_{t-q},$$

where

$$\varepsilon_t \sim \mathrm{wn}(\tau^2)$$

for some fixed (usually unknown) μ, θ_1, ..., θ_q and τ^2.

Mean and Variance of MA(q):

With similar reasoning to MA(1) case, can show

$E[y_t] = \mu$, and

$$\mathrm{Var}(y_t) = \mathrm{Var}(\varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots \theta_q \varepsilon_{t-q})$$

$$= \tau^2 \cdot (1 + \sum_{j=1}^{q} \theta_j^2) \ .$$

ACF of MA(1):

$$\rho(s) = \frac{(\theta_s + \sum_{j=s+1}^{q} \theta_j \cdot \theta_{j-s})}{(1 + \sum_{j=1}^{q} \theta_j^2)} \quad \text{if} \quad s = 0, 1, \dots, q$$

$$= 0 \quad \text{if} \quad s = q + 1, q + 2, \dots .$$

Remarks:

- ACF equals zero after q lags; dependence only for q adjacent periods.

- Partial ACF "tails off" to zero exponentially after q lags.

- If sample ACF "cuts off" after q lags, MA(q) may be appropriate model.

FIGURE - ACF FOR MA(Q), Q = 2

Model: $y_t = \varepsilon_t + 1.3\ \varepsilon_{t-1} - 0.5\ \varepsilon_{t-1}$

```
  -1.0      -0.6      -0.2       0.2       0.6       1.0
    +---------+---------+----0----+---------+---------+
1   |                        |           *            |  0.53
2   |                     *  |                        | -0.13
3   |                        *                        |  0.00
4   |                        *                        |  0.00
5   |                        *                        |  0.00
6   |                        *                        |  0.00
7   |                        *                        |  0.00
8   |                        *                        |  0.00
```

INVERTIBILITY OF MOVING AVERAGE PROCESS

We want to be able to write the MA(q) process

$$y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots \theta_q \varepsilon_{t-q}$$

in term of a (possibly infinite order) autoregression, i.e.,

$$y_t = \alpha + \varepsilon_t + \beta_1 y_{t-1} + \beta_2 y_{t-2} + \cdots\cdot$$

This is useful for prediction; given past values of y_t, this formula can be used recursively to obtain predicted future values $\hat{y}_s$ of the series, setting predicted future error terms to zero.

EXAMPLE: "INVERTING" MA(1)

Want to represent y_t in terms of past values and current error term. Since $y_t = \mu + \varepsilon_t + \theta\, \varepsilon_{t-1}$ implies

$$\varepsilon_t = (y_t - \mu) - \theta\, \varepsilon_{t-1},$$

repeated substitution yields

$$\varepsilon_t = (y_{t-s} - \mu) - \theta\, (y_{t-1} - \mu) + \theta^2 (y_{t-2} - \mu) + \ldots. , \quad \text{i.e.}$$

$$y_t = \varepsilon_t + \frac{\mu}{(1 + \theta)} + \theta\, y_{t-1} - \theta^2\, y_{t-2} + \theta^3\, y_{t-3} + \ldots.$$

Predicted Value at time T + 1:

Given y_T, y_{T-1}, ..., would predict future value y_{T+1} by

$$\hat{y}_{T+1} = \frac{\mu}{(1 + \theta)} + \theta\, y_T - \theta^2 y_{T-1} + \theta^3 y_{T-2} + \ldots ,$$

an exponentially-weighted sum of all current and past values of y_t. This representation requires $|\theta| < 1$; otherwise, infinite sum diverges.

Updating Formula for Forecasts:

If forecast for y_T given y y_{T-1}, y_{T-2}, ..., was

$$\hat{y}_T = \frac{\mu}{(1 + \theta)} + \theta\, y_{T-1} - \theta^2\, y_{T-2} + \theta^3\, y_{T-3} + \dots ,$$

then forecast of next period’s value is

$$\hat{y}_{T+1} = \theta\, y_T + (1 - \theta)\, \hat{y}_T ,$$

so forecasting formula has exponentially-weighted moving average form.

Remarks:

- For higher-order MA(q) processes, obtaining autoregressive representation more difficult, but it is built in to statistical package routines (like BJFRCST in TSP).

MIXED AUTOREGRESSIVE-MOVING AVERAGE ("ARMA(P,Q)") PROCESSES

$$y_t \sim \text{ARMA}(p,q)$$

if

$$y_t = \alpha + \beta_1 y_{t-1} + \dots \beta_p y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-1},$$

where

$$\varepsilon_t \sim \text{wn}(\tau^2) \ .$$

Remarks:

- This is a process that combines an autoregressive model for y_t with a moving average model for the error terms.

- Can generate very complex time-dependence patterns in data with very simple ARMA models. In practice, p and q are typically 2 or less.

Moments of ARMA(p,q):

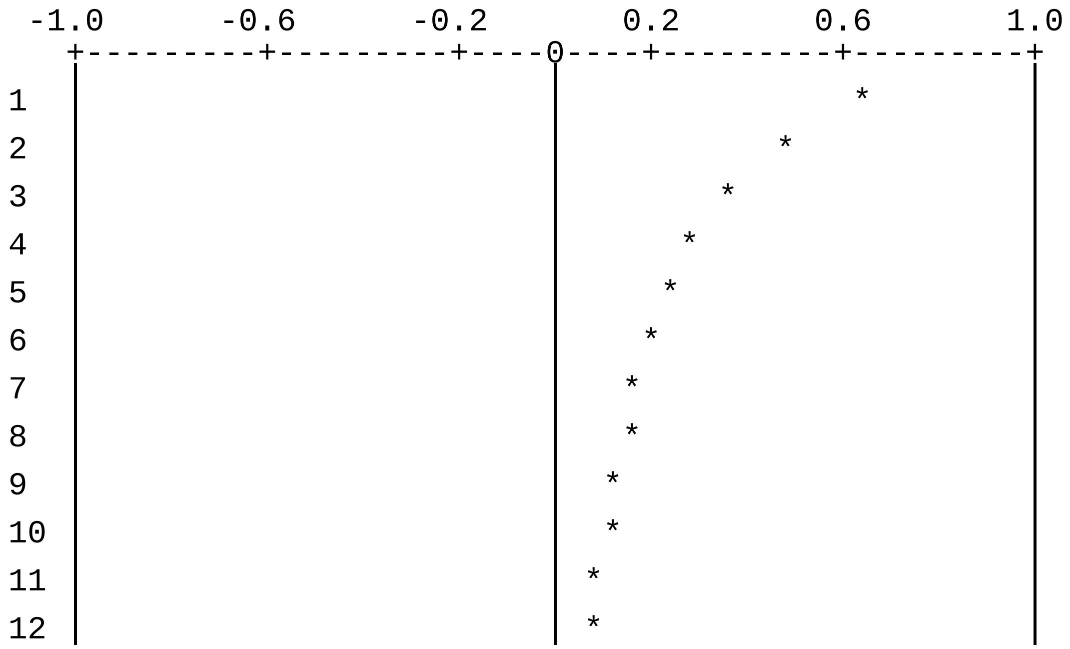
Like the AR(p) case, it is simple to calculate formula for mean of y_t, which has the same form:

$$\mu \equiv E[y_t] = \frac{\alpha}{1 - (\beta_1 + ... + \beta_p)} .$$

And it is even more difficult to calculate variance and autocovariances of ARMA(p) process than for the pure AR(p). However, the ACF for this process "tails off" after the first (q - p) lags.

Figure - ACF for AR(p), p = 2

Model: $y_t = 0.8\ y_{t-1} + \varepsilon_t - 0.3\ \varepsilon_{t-1}$



Remarks:

- Stationarity of ARMA(p,q) (i.e., constant mean, variance, and autocovariances) requires $|\beta_1 + ... + \beta_p| < 1$, and invertibility requires $|\theta_1 + ... + \theta_p| < 1$.

- ACF of ARMA(p, q) "tails off" exponentially after (q - p) lags; PACF "tails off" after (p - q) lags.

- Even for p = q = 1, ACF and PACF of mixed process can have complicated oscillatory patterns.

"UNIT ROOTS" AND INTEGRATED ("ARIMA(P,D,Q)") PROCESSES

If data are nonstationary, but the d^{th} difference of y_t is stationary with an ARMA representation, i.e.,

$$\Delta^d y_t = (1 - B)^d y_t \sim \text{ARMA}(p,q) \ ,$$

then the original (un-differenced, or integrated) process y_t is called an *integrated ARMA* (or ARIMA) process.

Remarks:

- Choice of number of differences d same as discussed in previous sections (i.e., use "eyeball metric" or check that estimated ACF of differenced series declines quickly to zero).

- Given model for differenced series, can forecast original y_t by "integrating," as discussed earlier.

- For economic data, typically have $d = 0, 1,$ or (at most) 2.

- Examples include longer-term commodity price and financial variables, like the three-month Treasury bill rate (plotted below).

EXAMPLE - 3-MONTH U.S. TREASURY BILL RATE

Sample runs from January, 1960, through February, 1996.

LECTURE / DISCUSSION

BOX-JENKINS IDENTIFICATION, ESTIMATION AND FORECASTING

"COMMON FACTOR" PROBLEM

If y_t is a low-order ARMA process, (say $y_t \sim \text{ARMA}(p, q)$ for p and q small) it can always be written as a higher-order process (e.g., $y_t \sim \text{ARMA}(p+1, q+1)$) by introducing "common factors" in the equation for y_t.

Example:

If y_t is white noise - that is, $y_t = \varepsilon_t$, then subtracting "$\beta\, y_{t-1}$" from y_t gives

$$y_t = \beta\, y_{t-1} + \varepsilon_t - \beta\, \varepsilon_{t-1},$$

which is of the ARMA(1, 1) form with $\theta = -\beta$.

- This applies to differencing as well - in general, can rewrite ARIMA(p,d,q) as $\text{ARIMA}(p+p^*, d+d^*, q+p^*+d^*)$ for any p^* and d^*.

"PRINCIPLE OF PARSIMONY"

Since any ARIMA process can be written as a higher-order ARIMA process by introducing common factors, we try to find the *lowest-order* ARIMA process that adequately fits the data. That is, we favor models with fewer parameters, since they involve estimation of fewer coefficients.

- Estimation of higher-order models typically yields "near common factors", that is, autoregressive and moving average coefficients that are nearly equal. This is a symptom of "overfitting," i.e., that a lower-order model should be used.

- General fitting strategy - starting with lowest-order model (say, ARIMA(0,0,0)), sequentially add parameters until model "fits well," i.e., has residuals which look uncorrelated across time periods. This is known as *time series identification* (where best p, d, and q are being "identified" by data).

ESTIMATION OF ARMA PROCESSES

Since residuals ε_t depend nonlinearly on unknown parameters because of recursion relationship

$$\varepsilon_t = y_t - \alpha - \beta_1 y_{t-1} - \cdots - \beta_p y_{t-p} - \theta_1 \varepsilon_{t-1} - \cdots - \theta_q \varepsilon_{t-1},$$

must use nonlinear least squares (which is maximum likelihood for normal errors) to estimate parameters of model. That is, choose $\hat{\alpha}$, $\hat{\beta}_1$, ..., $\hat{\beta}_p$, $\hat{\theta}_1$, ..., $\hat{\theta}_q$ to minimize

$$\text{SSE} = \sum_{t=1}^{T} (\hat{\varepsilon}_t)^2 ,$$

where the residuals $\hat{\varepsilon}_t$ are defined using the recursion relationship and the estimated parameters.

Remarks:

- Different estimation routines use different approximations for *unknown starting values*

$$y_0, \ y_{-1}, \ \ldots, \ y_{-p+1} \ \text{and} \ \varepsilon_0, \ \varepsilon_{-1}, \ \ldots, \ \varepsilon_{-q+1},$$

which are needed to evaluate first few residuals (in recursion above). These differences in approximations are negligible if the sample size T is very large, but for moderate sample sizes, different algorithms can give very different estimated coefficients.

- Fortunately, even though estimated equations are sensitive to "initial condition" assumptions, the forecasts $\hat{y}_s$ of future y_s tend to be less sensitive.

STEPS IN BOX-JENKINS ARIMA MODELING

1. Determine order of differencing, d, by inspecting estimated ACF; pick lowest order for which ACF dies out quickly.

2. Pick preliminary orders p and q of ARMA(p,q) by inspecting ACF and PACF for $\Delta^d y_t$; for example, pick p to match highest significant lag in PACF, and use ACF analogously to pick q, or start with low values of p and q (say p = q = 1).

3. Fit ARMA(p,q) model to data using maximum likelihood (nonlinear least squares).

4. Use residuals to determine whether they are serially uncorrelated, using *diagnostic tests* - in particular, the *portmanteau t*est based on the "Q" statistic (described below).

5. If null hypothesis of uncorrelated residuals is rejected, increase p and/or q and return to step 3.

6. If Q-statistic indicates residuals are uncorrelated, try reducing p and/or q in steps 3 and 4. Continue until further reductions yield models that fail portmanteau test.

PORTMANTEAU ("Q") STATISTIC

If the orders of an ARIMA model are correctly specified, the residuals $\hat{\varepsilon}_t$ are estimates of white-noise error terms ε_t, which has true ACF equal to zero for all positive lags. Thus, the estimated ACF of the residuals, $r_\varepsilon(s)$, should estimate zero when $s = 1, 2, \ldots.$ The statistic

$$Q \equiv \sum_{s=1}^{k} (r_\varepsilon(s))^2$$

should be smaller for a correctly-specified ARIMA than for a misspecifed ARIMA (with p, q, or d too small).

For k sufficiently large, Q is approximately distributed as a chi-squared random variable with $k - p - q$ degrees of freedom under the null hypothesis that the model is correctly specified.

WORKSHOP

ARIMA MODELS

In this workshop, the object is to fit ARIMA models to some of the data sets considered in the earlier workshops, and to evaluate their post-sample forecasting performance. Three of the data sets are particularly amenable to ARIMA modelling:

1. The three month T-bill interest rate variable, "r3," introduced in the previous workshop. This series, from 60:1 to 96:2, is in the **returnsp.xls** file in the **thuaft** subdirectory. As for the earlier workshop, we will take the sample period for estimation from 60:1 to 94:12 (180 observations), and use the period from 95:1 through 96:2 for post-sample evaluation.

2. The Standard and Poor’s stock price variable, "snp," in the same data file and with the same sampling ranges as the "r3" variable.

3. The U.S. housing start series from the workshop on seasonal adjustment; the data are in **housing.xls**, the sample period is from 86:1 to 94:10, and the *ex-post* forecasting period is from 94:11 to 95:10.

The first of the three series is probably the most challenging in terms of model specification, and should be considered first; however, the others are also worthy of consideration, if time permits. (For example, the housing start variable will probably require seasonal differencing to achieve stationarity.)

For the "r3" series (and possibly the others), work through the Box-Jenkins model specification cycle (selection of orders, model fitting, and diagnostic checking) using the TSP command **bjident**, which was discussed earlier, and the TSP command for estimating ARIMA models, **bjest**. The format for this command is

```
bjest(constant, nar=p, ndiff=d,
             nma=q, nsdiff=d) series name;
```

there are other options available, as described in the TSP manual, but these should be enough for the present purposes.

After determining the "best" model and estimating the parameters using **bjest**, we can use the model to produce forecasts using **bjforcst**. For the T-bill rate series, the command

```
bjforcst(orgbeg=95:1, nhoriz=14) r3;
```

should be inserted immediately after the **bjest** command; since the estimation routine keeps track of the number of differences, etc., these do not need to be respecified for the forecasting routine, and the original "r3" variable, not its difference, can be used. (The forecasting routine does all the "integrating" needed to get back to the original data series, which is pretty convenient).

Again, graph the "r3" variable and its forecasted value over the post-sample period, and evaluate the sample MSE of the forecast.

DISCUSSION OF WORKSHOP RESULTS