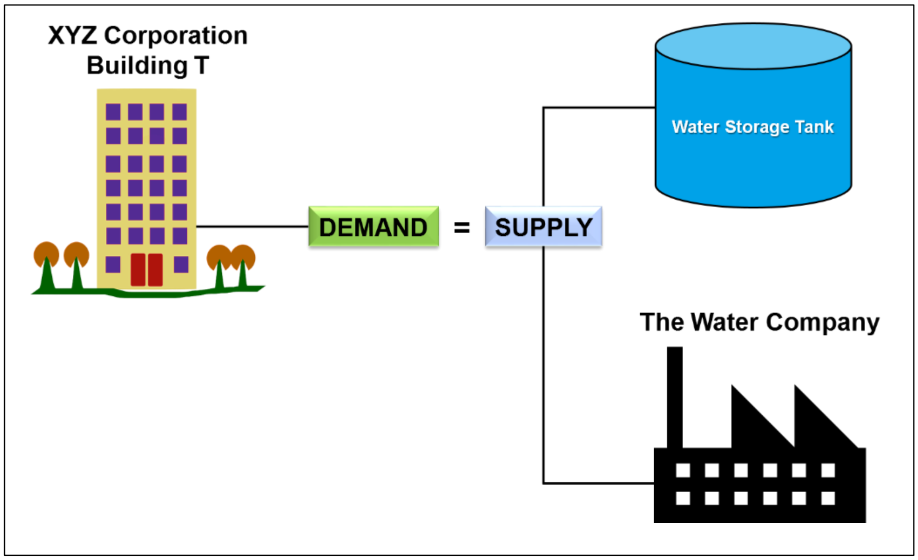
Chapter 4 Forecasting, Experimental Design, and Optimization

4.1 Forecasting, Experimental Design, and Optimization: Background and Project Overview Case Study

This case study reinforces the concepts that you learned in the following courses: Time Series Modeling Essentials, Experimentation in Data Science, and Optimization Concepts for Data Science. Prerequisite concepts in the SAS Academy for Data Science might be required to successfully complete this case study.



Building T is the corporate headquarters of the XYZ Corporation. The XYZ Corporation, like other businesses in the area, has a contract in place with The Water Company to provide water to the building at a contracted price per gallon.

What differentiates XYZ Corporation from other businesses is that it does not depend solely on The Water Company to supply the building with water. In fact, XYZ Corporation has a second source: its own water storage tank. Precipitation is collected, treated, stored, and used to supply water to Building T, and there is a per-gallon cost associated with this process.

XYZ Corporation hired your team of consultants to provide solutions to three main objectives.

1. Forecast weekly water demand in Building T for the next four weeks.
2. Design an experiment around the water treatment process to lower the per-gallon cost associated with supplying water to Building T from the water storage tank.
3. Using the weekly water demand forecasts, build an optimization model to choose favorable contract rates with The Water Company.

More detail about each objective is provided below.

Objective 1: Forecasting

Your team is given a SAS data set, **rawdata_BldgT**. It contains more than a year and a half of historical weekly gallon data from Building T.

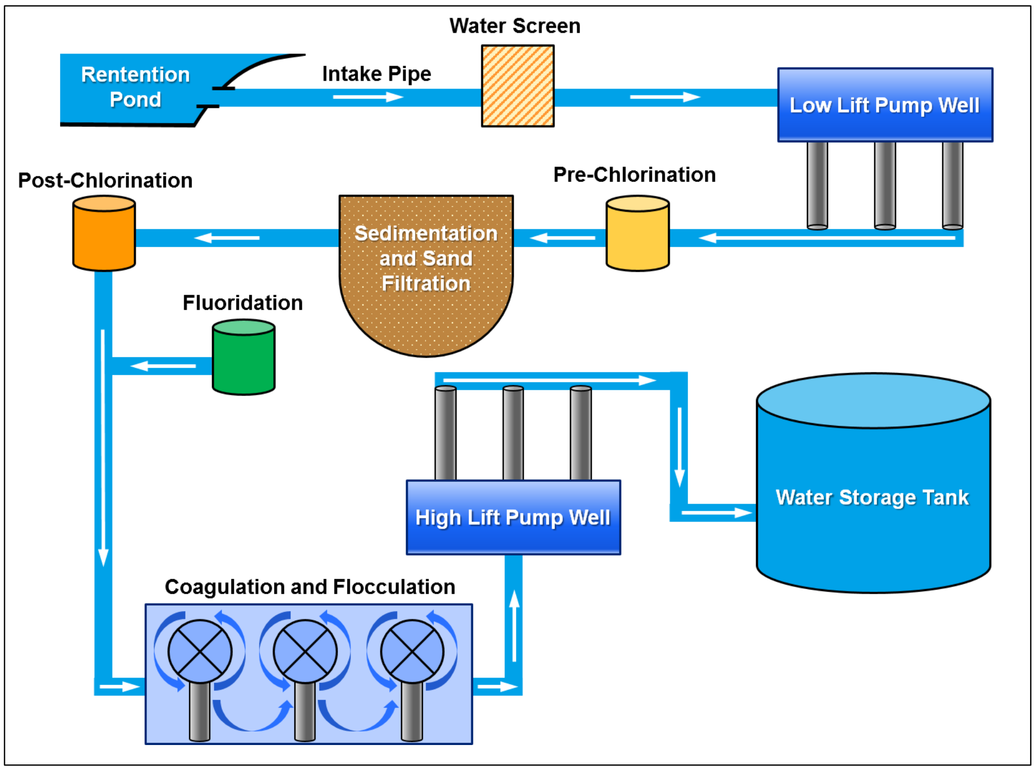
The gallons used per week are divided into two categories: Cooling and Main. The *Cooling category* represents the gallons used to regulate the temperature in the building. The *Main category* represents the gallons used by the employees. Together, they are the total amount of water, in gallons, that is used per week in Building T. There is no difference in the quality of the water between the two categories.

The date range begins on the week of July 7, 2013, and runs through the end of the week beginning April 12, 2015. There is no partial or missing data for the weeks.

XYZ Corporation needs ***total*** water demand forecasts for each week over the next four weeks.

For Objective 1, provide answers to the following questions:

1. How many total gallons of water is Building T expected to use in each of the next four weeks?
2. When you developed the forecast models, were there any identifiable patterns in water usage over time? If so, how would your team characterize these patterns?



Objective 2: Experimental Design

Examine the water treatment process flow chart and design an experiment to lower the per-gallon cost that is associated with this process. Currently, it costs XYZ Corporation 18 cents ($0.18) per gallon to supply Building T with treated water from the water storage tank.

For Objective 2, provide the following:

1. Make a proposal for how a designed experiment can be performed on the water treatment process to lower the per-gallon cost that is associated with supplying water to Building T via the water storage tank.
2. Identify the experimental units for this experiment.
3. In your proposal, include at least two experimental factors, two covariates, and one blocking variable. Describe each and how you can incorporate them into the experimental design.

Objective 3: Optimization

Contracts between XYZ Corporation and The Water Company are renewed every four weeks. The current four weeks ended, and XYZ Corporation received two contract proposals from The Water Company to supply water for the next four weeks.

The ***first contract*** from The Water Company supplies water at 15 cents ($0.15) per gallon with a minimum of 25,000 gallons purchased per week.

The ***second contract*** supplies water at 12 cents ($0.12) per gallon with a minimum of 35,000 gallons purchased per week.

Both contracts do not have a capped amount. Therefore, XYZ Corporation can purchase as many gallons as needed from The Water Company at or above the respective minimum gallon thresholds at the contracted price.

Currently, 62,500 gallons are in the water storage tank. The president of XYZ Corporation is insistent that the water storage tank does not drop below 30,000 gallons during any week over the next four weeks. For the next two weeks, the treatment cost from water storage is 18 cents ($0.18) per gallon. Due to expected gains in water treatment efficiency from your team's proposed experimental design, the treatment cost will drop to 10 cents ($0.10) per gallon in weeks three and four.

The number of gallons in the water storage tank is dependent on the number of gallons remaining from the prior week, plus the difference between how many gallons of precipitation accumulated in the current week and how many gallons are used to supply Building T from the water storage tank in the current week. The amount of expected precipitation in that region over the next four weeks was already estimated. Based on expected precipitation and the accumulation methods used by XYZ Corporation, they expect to add 12,000 gallons to the water storage tank in the first week, 18,000 gallons in the second week, 20,000 gallons in the third week, and 22,000 gallons in the fourth week. There is no cap on the amount of water that can be stored at any one time.

XYZ Corporation is also a proud member of the Elite Environmental Corporate Sustainer Initiative (EECSI). The primary environmental sustainability project that granted XYZ Corporation membership into EECSI is its investment in renewable water usage (via the water storage tank). One of the requirements for XYZ to remain a member of EECSI is that at least 25% of all water supplied to Building T each week must come from the water storage tank.

Use your team's total water demand forecasts from Objective 1, along with the information above, to construct an optimization model that minimizes total water cost during the next four weeks. The optimal combination of gallons purchased from The Water Company and used from the water storage tank each week must be at least equal to the forecasted water demand for each week. The objective function, total water cost over the next four weeks, is equal to the sum of (Price*Quantity bought from The Water Company) + (Treatment Cost*Quantity used from the water storage tank) for each week.

For Objective 3, provide answers to the questions below.

1. Which of the two proposed contracts provides XYZ Corporation with the lowest total water cost over the next four weeks?
   - 15 cents ($0.15) per gallon with a minimum of 25,000 gallons purchased per week
   - 12 cents ($0.12) per gallon with a minimum of 35,000 gallons purchased per week

After you choose the contract recommended by your team, answer the following questions:

2. How many gallons will XYZ buy from The Water Company each week?
3. How many gallons will XYZ use from the water storage tank each week?

4. What is XYZ's projected total water cost at the end of the next four weeks?
5. What is XYZ's projected ending water storage tank inventory at the end of each week?
6. How much money can XYZ save by choosing the recommended contract over the alternative contract?
7. How many more or fewer gallons are projected to be in the water storage tank at the end of the four-week period compared to a choice for the alternative contract?

Competition Guidelines

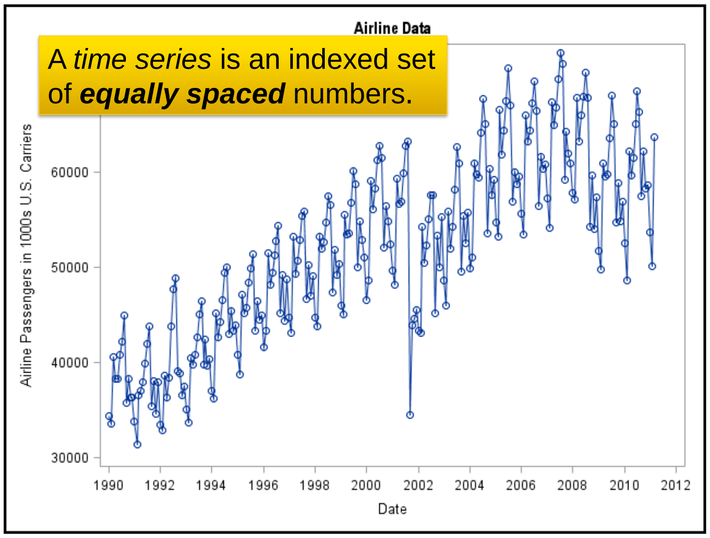
Your team is competing with the other teams in the class. Your team is responsible for two things. First, each team submits to the instructor answers to the questions from Objectives 1, 2, and 3. Second, each team presents their answers, ***in a concise manner***, to the class. Structure your presentation as if you were presenting to a high-level, non-technical audience.

The instructor has the actual (correct) answers for the gallon demand in Building T over the four weeks that your team is forecasting. The correct answers to the optimization questions outlined in Objective 3 are also available. These are used to evaluate the accuracy of your team's responses to the questions above. Because your team is not responsible for generating weekly precipitation forecasts, for consistency, assume that the provided estimates are 100% accurate.

4.2 Forecasting, Experimental Design, and Optimization: Background and Project Overview Review

Time Series Modeling Essentials: Review

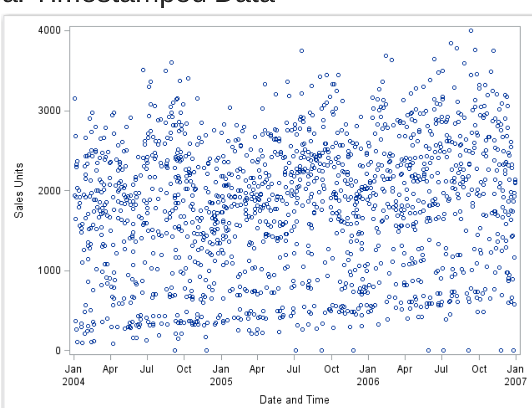
A Time Series



5

Transactional Data: Example

Historical Timestamped Data



No obvious patterns exist in the transactional data.

6

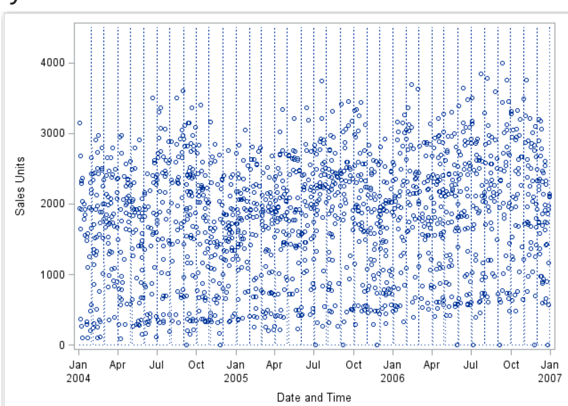
Transactional Analysis

- Given a timestamped data set, each observation of the data set can be assigned an observation (raw) index, a time index, and a season index.
- Count and frequency analysis can be applied to the timestamped data set based on these indices.
- Each of these indices does not depend on the data under analysis. These indices only structure the data for subsequent analysis.

7

Transactional Data Time Binning

Monthly Time Bins



A monthly interval time series has one observation per interval or bin.

8

Transactional Data Time Binning

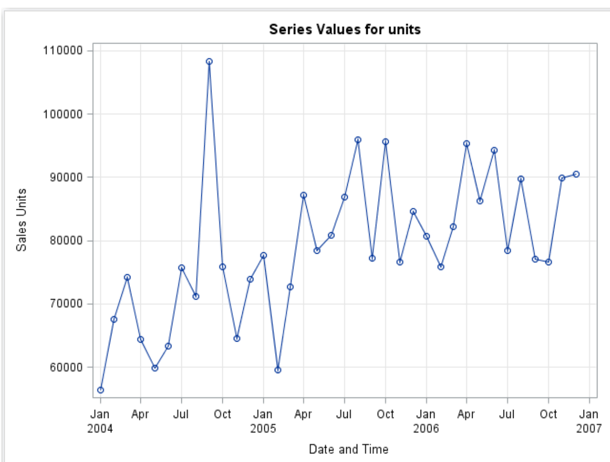
Perform time binning with the TIMESERIES procedure's INTERVAL= option. Use accumulated totals.

```
proc timeseries data=transactions out=outsum;
   id date interval=month
           accumulate=total;
   var units;
run;
```

9

Transactional Data Accumulation

Accumulated on a Monthly ***Total*** Basis



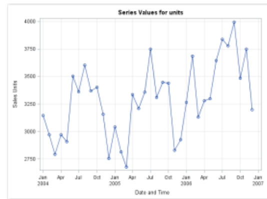
10

Transactional Data Accumulation

For this timestamped data, accumulating on a monthly ***average*** or ***maximum*** basis might be better than accumulating on a ***total*** basis.

ACCUMULATE=TOTAL

ACCUMULATE=AVERAGE

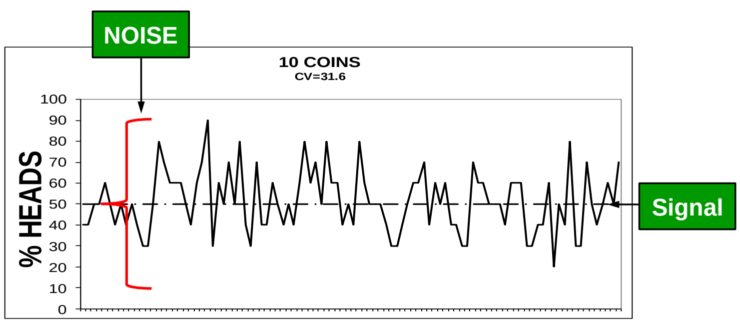


ACCUMULATE=MAXIMUM

11

Variation in Time Series Data: Two Main Parts

- signal
- noise



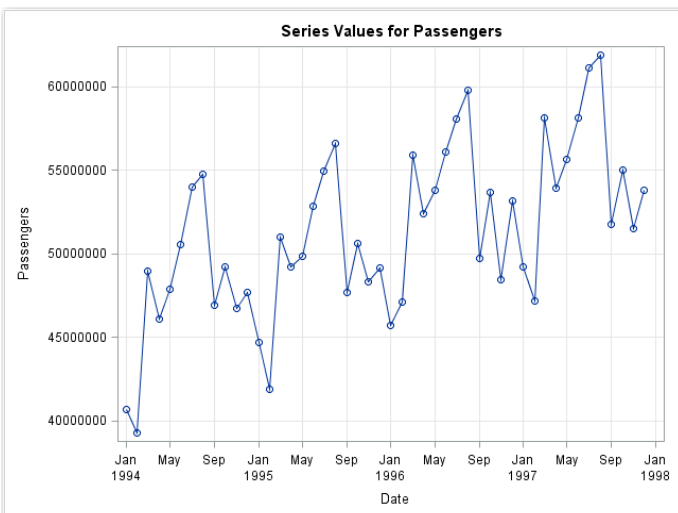
12

Signal Components

- level
- seasonality
- trend
- irregular
- exogenous
  (also known as *explanatory variable effects*)
- cycle

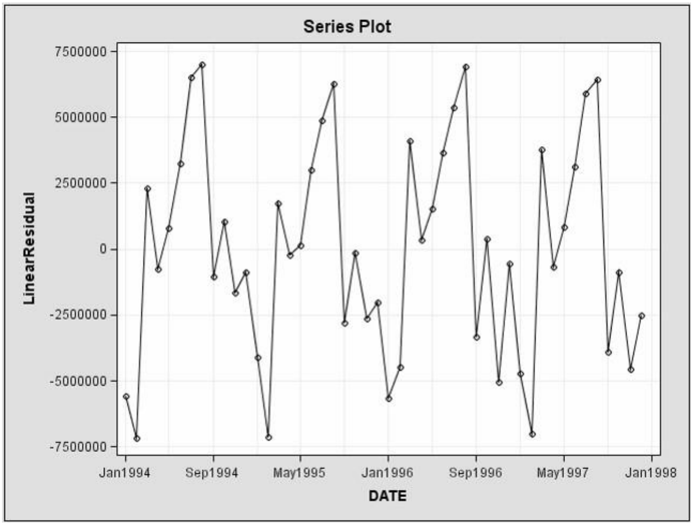
13

The Airline Data



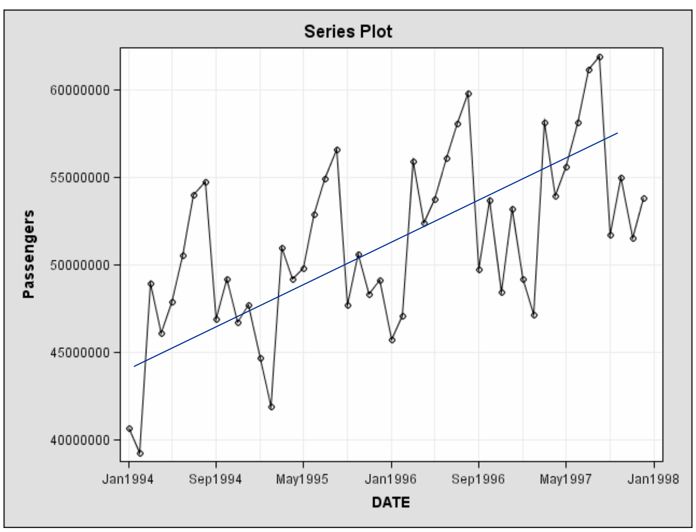
14

Signal Components: Seasonality



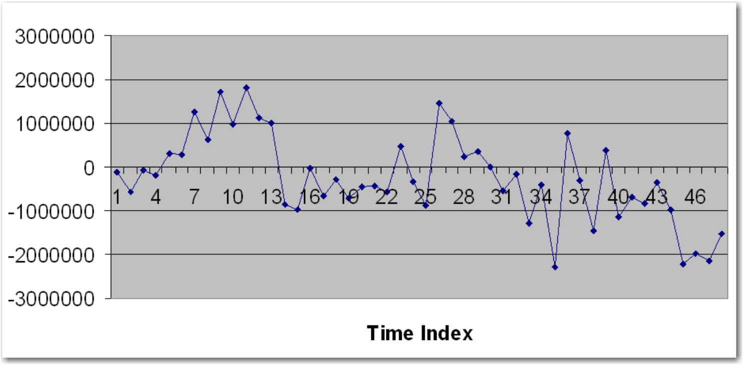
15

Signal Components: Trend



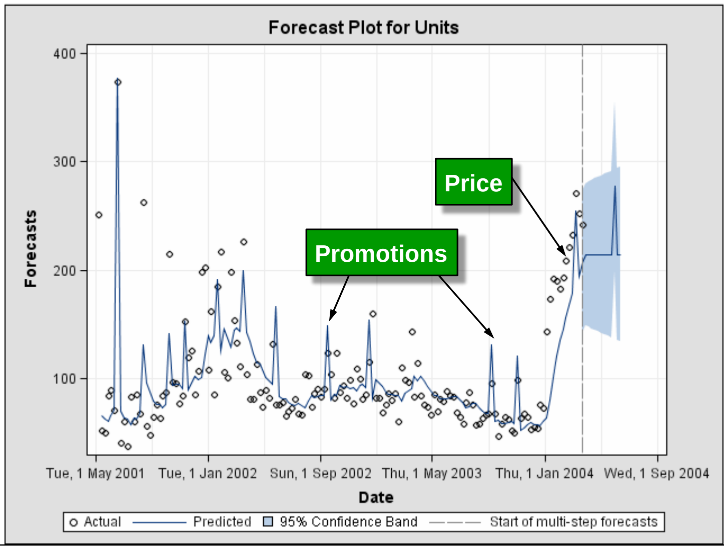
16

Signal Components: Irregular



17

Signal Components: Exogenous Effects



18

Necessary Conditions for Good Forecasts

- The identified signal continues into the future.
- Forecasting model complexity should be adequate to capture signal components.
- Forecasting models should not be overly complex.
- The best forecasting model is the one that captures and extrapolates the most signal, and that also ignores the noise.

19

Time Series Modeling Essentials – Review

- ARMA and ARMAX Model Review
- Exponential Smoothing Models Review
- UCM Models Review

20

Correlation of Y with Past Y: *Autocorrelation*

Autocorrelation (Order 1):

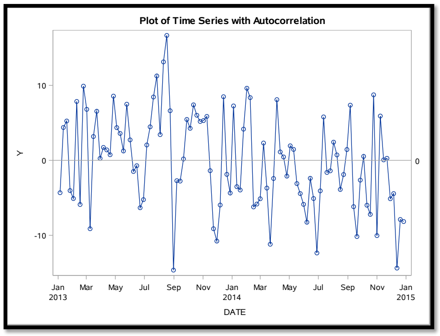
$$Y_t \text{ is correlated with } Y_{t-1}$$

Time Series at Time t:

$$Y_t = Y(t)$$

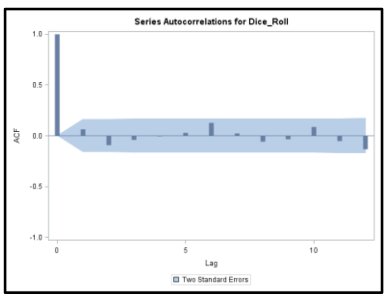
First Lag:

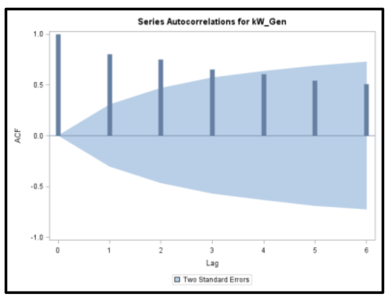
$$Y_{t-1} = Y(t-1)$$



21

Autocorrelation Plots





- The autocorrelation plot enables you to see the autocorrelation at multiple lags.
- The blue range indicates 95% confidence intervals for each lag.

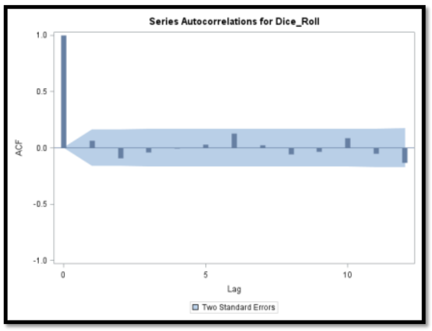
22

White Note

- White noise is a series that varies randomly around its mean.
- It has no systematic variation.
- It consists of only random variation.

Forecasting a white noise process reverts to the mean.

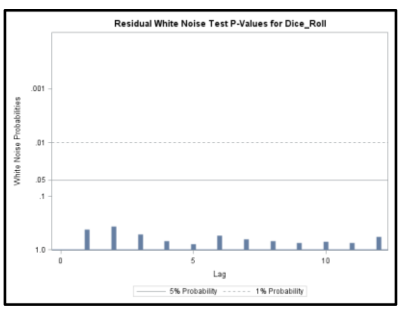
Mean
7.02

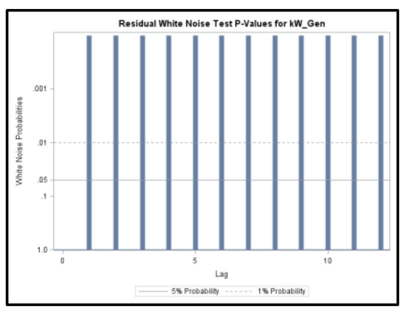


Yt	Count
2	5
3	3
4	15
5	21
6	21
7	21
8	21
9	18
10	11
11	12
12	2

23

The Ljung-Box Chi-Square Test for White Noise





“White means white.”

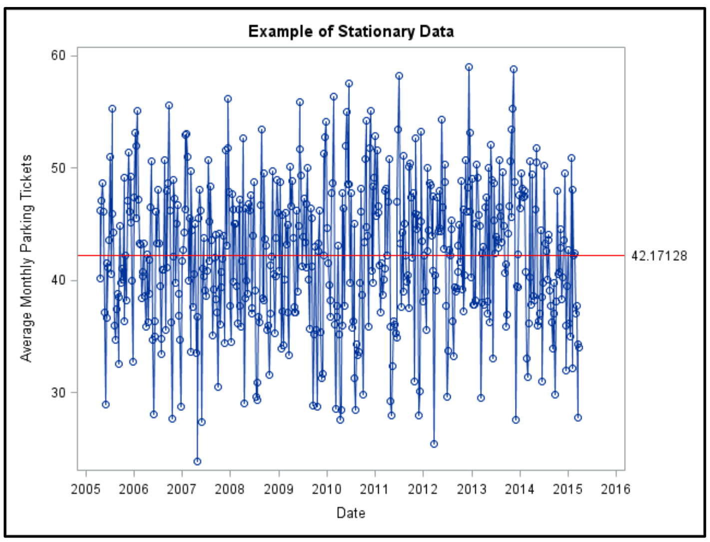
24

Stationarity

- A *stationary* time series is defined as having a constant mean, constant variance, and that any autocorrelation between adjacent terms is constant across all time periods.
- A *nonstationary* time series does not have a constant mean and variance, and tends to exhibit a discernable pattern in the data across time.
- A time series with long-term trend or seasonal components cannot be stationary because the mean of the series depends on the time that the value is observed.

25

Visualizing Stationary Data



26

ARMA versus ARIMA Models

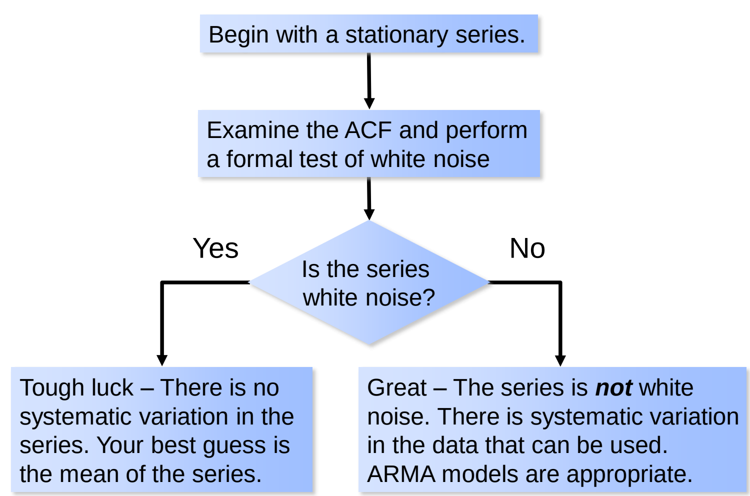
The "I" in ARIMA stands for *integrated*, and tells you in what *order* the data was differenced to convert it to a stationary process.

- starting with a stationary process: ARMA model
- starting without a stationary process: Transforming the data in order to create a stationary process warrants using an ARIMA model.

This class works with stationary time series and thus uses ARMA models.

27

ARMA Models: Initial Process Flow



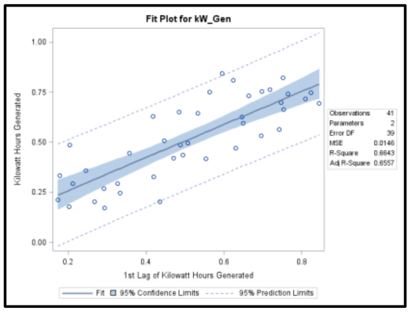
28

Regression of Y on Past Y: *Autoregression*

Reminder:

OLS Regression Model: $$Y = \beta_0 + \beta_1 X + \varepsilon$$

Autoregressive (Order 1) Model:



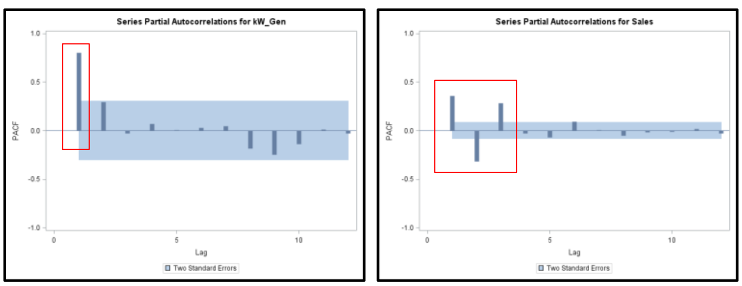
$$Y_t = \beta_0 + \beta_1 Y_{t-1} + \varepsilon_t$$

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \varepsilon_t$$

29

Partial Autocorrelation Function Plot (PACF)

- Significant spikes in the PACF are the most important source of information in identifying an autoregressive series.



- Unlike the ACF, autocorrelations do not “spill over” from lag to lag, but rather hold constant all lower order lags.

30

Autoregressive versus Moving Average Models

First Order Autoregressive Model AR(1)

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \varepsilon_t$$

– Y_t is a function of the previous value plus some error.

First Order Moving Average MA(1)

$$Y_t = \theta_0 - \theta_1 \varepsilon_{t-1} + \varepsilon_t$$

– Y_t is a function of its immediately previous shock plus error (significant autocorrelation between Y_t and ε_{t-1}).

31

ML Estimation of an AR(1) Model

Maximum Likelihood Estimation					
Parameter	Estimate	Standard Error	*t* Value	Approx Pr > |t|	Lag
MU	42.59964	0.46518	91.58	<.0001	0
AR1,1	0.45529	0.03909	11.65	<.0001	1

- **MU** is the estimated mean of the series, μ.
- **AR1,1** at **Lag=1** is the estimated first order autoregression parameter, ϕ_1.
- *P*-values test H_0: parameter=0.
- If the series is white noise, then all parameter estimates, other than that for MU, should be non-significant.

32

Model Goodness-of-Fit Statistics

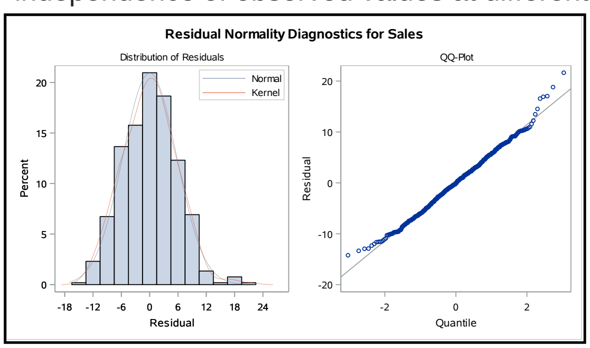
A diagnostic statistic calculated using the same sample that was used to fit the model is a *goodness-of-fit* statistic.

Constant Estimate	23.20426
Variance Estimate	33.5095
Std Error Estimate	5.788739
AIC	3304.076
SBC	3312.583
Number of Residuals	520

33

Check of Residuals

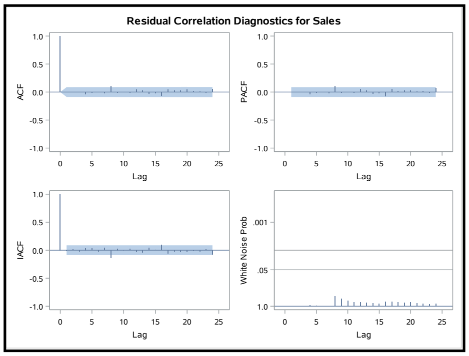
- The residuals are $(Y_t - Y_t)$.
- White Noise Assumption
  - **normal distribution with a mean of 0 and constant variance σ^2**
  - independence of observed values at different times



34

Assumptions for Residuals

- White Noise Assumption
  - normal distribution with a mean of 0 and constant variance σ^2
  - **independence of observed values at different times**

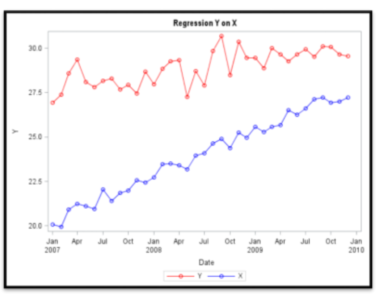


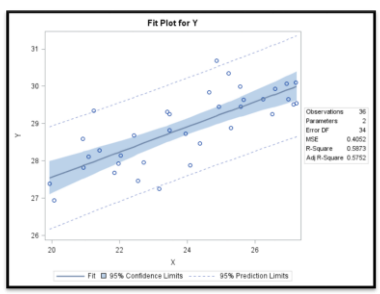
35

Regression of Y on X

Linear Regression Model: $Y_t = \beta_0 + \beta_1 X_t + \varepsilon_t$ *

* X_t is an external or *exogenous* predictor of Y_t.





36

Time Series Regression Terminology

Ordinary Regressor

- an input variable that has only a concurrent influence on the target variable
  - X at time *t* is correlated with Y at time *t*.
    X at times before *t* is uncorrelated with Y at time *t*.

Dynamic Regressor

- an input variable that influences the target variable at current and past values
  - X at times *t*, *t*-1, *t*-2, …, influences Y at time *t*.

Transfer Function

- a function that provides the mathematical relationship between a dynamic regressor and the target variable

37

Types of Regressors—Measurement Scale

Binary (dummy) variables

- take the value zero or one
- can be used to quantify nominal data

Categorical variables

- nominal scaled ⇨ nonquantitative categories
- Ordinal scaled variables can be treated as categorical.
- They must be coded into a quantitative input, usually using a form of dummy coding for each level (less one if a constant term is used in the model).

Quantitative variables

- interval or ratio scaled
- can be transformed

38

Types of Regressors—Randomness

Deterministic

- controlled by experimenter
- alternatively, can be perfectly predicted without error

Stochastic

- governed by unknown probability distributions
- cannot be perfectly predicted

39

The Cross-Correlation Function (CCF)

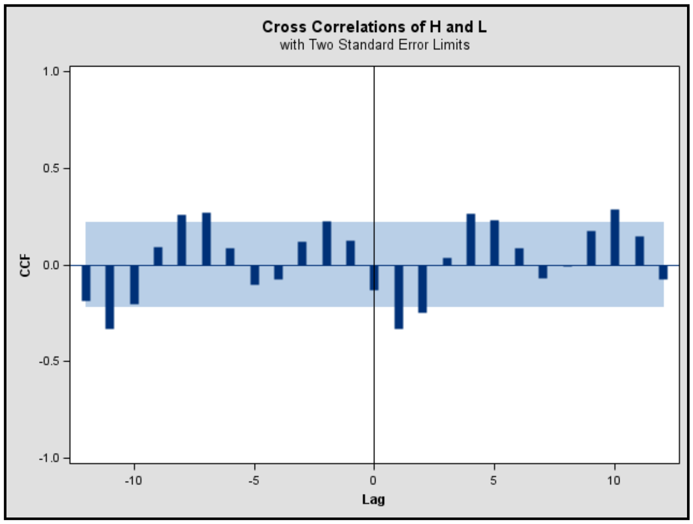
CCF(k) is the cross-correlation of target Y with input X at lag k.

- A significant value at lag k implies that Y_t and X_{t-k} are correlated.
- Spikes and decay patterns in the cross-correlation function can help determine the form of the transfer function.
- The sample CCF estimates an unknown population CCF.

40

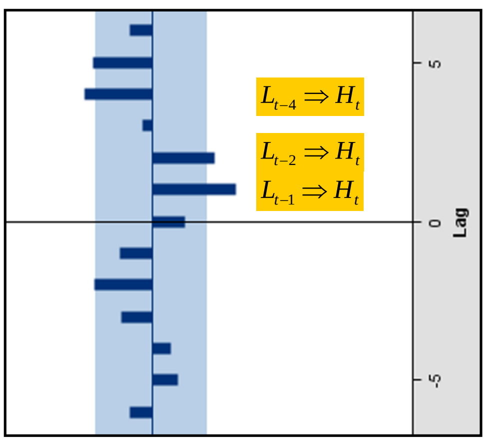
continued...

The Cross-Correlation Function (CCF)



41

The Cross-Correlation Function (CCF)



L	H
Jan	Jan
Feb	Feb
Mar	Mar
Apr	Apr
May	May

42

Events

- An *event* is anything that changes the underlying process that generates time series data.
- The analysis of events includes two activities:
  - exploration to identify the functional form of the effect of the event
  - inference to determine whether the event has a statistically significant effect
- Other names for the analysis of events are the following:
  - ***intervention analysis***
  - interrupted time series analysis

43

Intervention Analysis

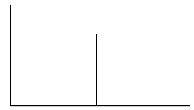
- Special case of *transfer function modeling* in which the predictor variable is a deterministic categorical variable
- Derived from the concept of a public policy *intervention* having an effect on a socio-economic variable
  - Example: Raising the minimum wage increases the unemployment rate.
  - Example: Implementing a severe drunk-driving law reduces automobile fatalities.

44

Examples of Input Variables

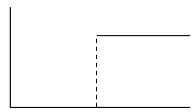
Point/Pulse

X	0	0	0	1	0	0	0	….
Y	0	0	0	8	0	0	0	
t	1	2	3	4	5	6	7	



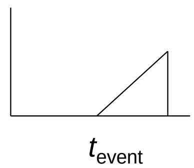
Step

X	0	0	0	1	1	1	1	….
Y	0	0	0	8	8	8	8	
t	1	2	3	4	5	6	7	



Ramp

X	0	0	0	1	2	3	0	….
Y	0	0	0	2	4	6	0	
t	1	2	3	4	5	6	7	



45

Forecasting

If someone asks you if you can forecast something, your answer should always be "Yes."

If someone asks you if you can forecast something ***accurately***, you cannot answer until you establish what accuracy means and until you perform preliminary modeling of the data.

46

Liability

- You have no control over future events such as catastrophes, economic downturns, war, the integrity of key players, the survival of key players, and so on.
- You need to assume that the underlying future behavior remains consistent with past behavior.

47

Forecasting Before You Forecast

	Quarter	*t*	
Ultimate Goal: Forecast the next four quarters.	4Q2015	Y_{t+4}	
	3Q2015	Y_{t+3}	
	2Q2015	Y_{t+2}	
	1Q2015	Y_{t+1}	
How well can you forecast these four most recent observed quarters?	4Q2014	Y_t	Holdout Sample
	3Q2014	Y_{t-1}	
	2Q2014	Y_{t-2}	
	1Q2014	Y_{t-3}	
Forecasting observed values with the remaining observed series	4Q2013	Y_{t-4}	Fit Sample
	…	…	

48

Holdout Sample: Simulating a Retrospective Study

1. Divide the time series data into two segments. The *fit sample* is used to derive a forecast model. The *holdout sample* is used to evaluate forecast accuracy.
2. Derive a set of candidate models.
3. Calculate the chosen model accuracy statistic by forecasting the holdout sample.
4. Choose the model with the best accuracy statistic.

49

Choosing the Holdout Sample

- Choose enough time points to cover a complete seasonal period. For example, for monthly data, hold out at least 12 observations.
- The holdout sample is always at the end of the series.
- If unique behavior occurs within the holdout sample, do not use a holdout sample. Instead, base accuracy calculations on the entire series.
- If there is insufficient data to fit a model without the holdout sample, then do not use a holdout sample. Again, base accuracy calculations on the entire series.

50

Rules of Thumb

- At least four time points are required for every parameter to be estimated in a model.
- Anything above the minimum series length can be used to create a holdout sample.
- Holdout samples should rarely contain more than 25% of the series.

51

Model Fit Statistics

Mean Absolute Percent Error:

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n} | Y_t - \hat{Y}_t | / Y_t$$

Mean Absolute Error:

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n} | Y_t - \hat{Y}_t |$$

52

continued...

Model Fit Statistics

R-Square: $R^2 = 1 - \sum_{t=1}^{n} (Y_t - \hat{Y}_t)^2 \Big/ \sum_{t=1}^{n} (Y_t - \overline{Y})^2$

Root Mean Square Error:

$$\mathrm{MSE} = \frac{1}{n-k} \sum_{t=1}^{n} (Y_t - \hat{Y}_t)^2 \quad *$$

$$\mathrm{RMSE} = \sqrt{\mathrm{MSE}}$$

* For holdout samples, use divisor *n* rather than *n-k*.

53

The Stochastic Input Variable Conundrum

- Future values of the input variable are the following values:
  - deterministic (known)
  - stochastic (unknown, and therefore, estimated)
- A stochastic input, X_t, must be forecast for *T* periods so that Y_t can be forecast for *T* periods.
- The forecast accuracy of Y_t depends on the forecast accuracy of the stochastic input variable.

54

Scenario Analysis / What-if Analysis

- Choose future values of the stochastic input variable to generate different forecasts for Y_t.
- Run the same model, and replace the chosen future value each time.
- This reduces a complex process into a series of simple Boolean logic steps.
  - For example, for period *t*+2:
    - if $X_{t+2} = X_1$, then $Y_{t+2} = Y_1$
    - if $X_{t+2} = X_2$, then $Y_{t+2} = Y_2$
    - …
    - if $X_{t+2} = X_k$, then $Y_{t+2} = Y_k$
- For *k* chosen future values of X_{t+2}

55

Choosing a Winning Set of Forecasts

Good forecasts should have the following characteristics:

- be highly correlated with the actual series values
- exhibit small forecast errors
- capture the prominent features of the original time series

In addition, forecast quality should be based on the business, engineering, or scientific problem that is being addressed.

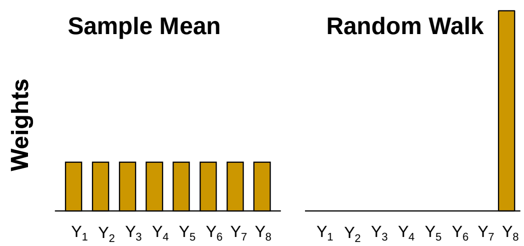
56

Time Series Modeling Essentials – Review

- ARMA and ARMAX Model Review
- Exponential Smoothing Models Review
- UCM Models Review

57

Weighted Average Examples



Weights applied to past values to predict Y_9

58

Simple Moving Average

Disadvantages

- cannot be used on the first *n*-1 terms of the time series without adding other terms by some other means
- can be influenced by extreme values within the window
- requires the retaining of the most recent *n* observations to produce forecasted value

59

Exponential Smoothing Models: Premise

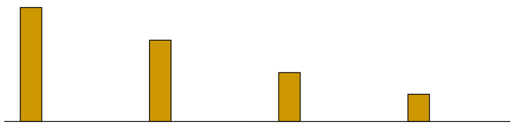
- Weighted averages of past values can produce good forecasts of the future.
- The weights should emphasize the most recent data.
- Forecasting should require only a few parameters.
- Forecast equations should be simple and easy to implement.

60

The Exponential Smoothing Coefficient

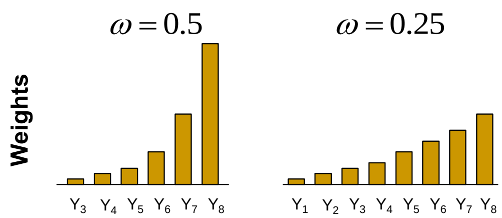
Forecast Equation

$$
\begin{aligned}
\hat{Y}_{t+1} &= \omega Y_t + (1-\omega)\hat{Y}_t \\
&= \omega Y_t + (1-\omega)[\omega Y_{t-1} + (1-\omega)\hat{Y}_{t-1}] \\
&= \omega Y_t + \omega(1-\omega)Y_{t-1} + (1-\omega)^2 \hat{Y}_{t-1} \\
&= \omega Y_t + \omega(1-\omega)Y_{t-1} + (1-\omega)^2[\omega Y_{t-2} + (1-\omega)\hat{Y}_{t-2}] \\
&= \omega Y_t + \omega(1-\omega)Y_{t-1} + \omega(1-\omega)^2 Y_{t-2} + \omega(1-\omega)^3 Y_{t-3} + \cdots
\end{aligned}
$$



61

Simple Exponential Smoothing

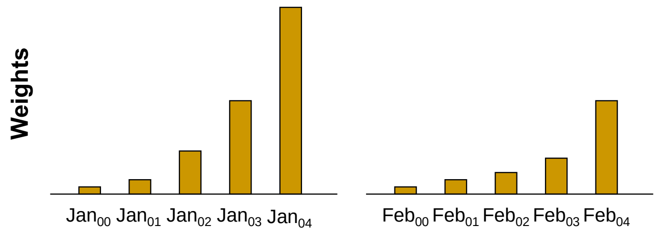


Weights applied to past values to predict Y_9

As the parameter increases, the emphasis on the most recent values increases.

62

Exponential Smoothing for Seasonal Data



Weights decay with respect to the seasonal factor.

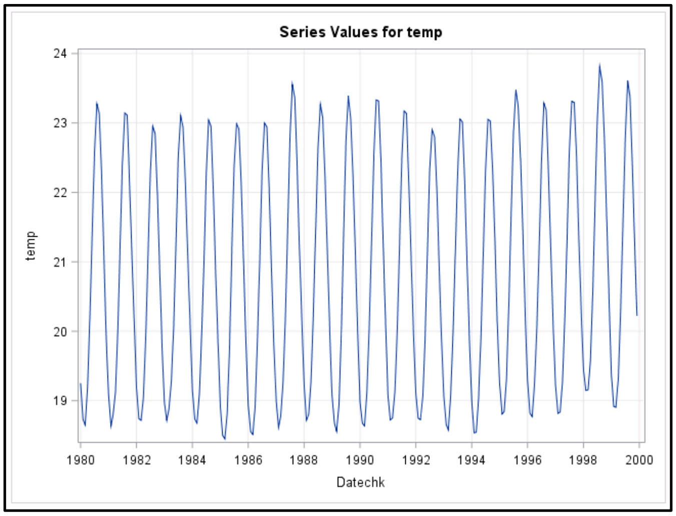
63

ESM Parameters and Keywords

ESM	Parameters	Model=*Keyword*
Simple	ω	SIMPLE
Double	ω	DOUBLE
Linear (Holt)	ω, γ	LINEAR
Damped-Trend	ω, γ, ϕ	DAMPTREND
Seasonal	ω, δ	SEASONAL
Additive Winters	ω, γ, δ	ADDWINTERS
Multiplicative Winters	ω, γ, δ	WINTERS

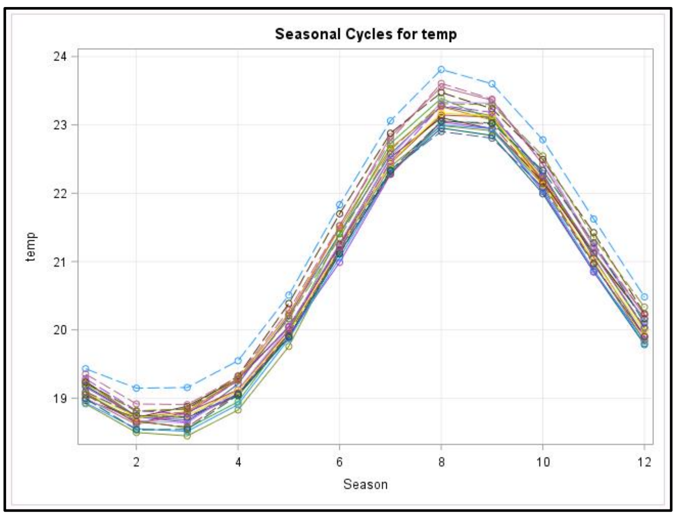
64

ESM ODS Output



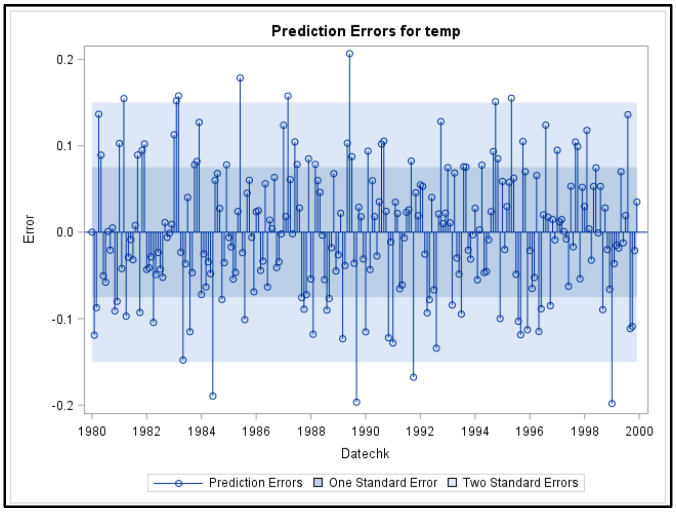
65

ESM ODS Output



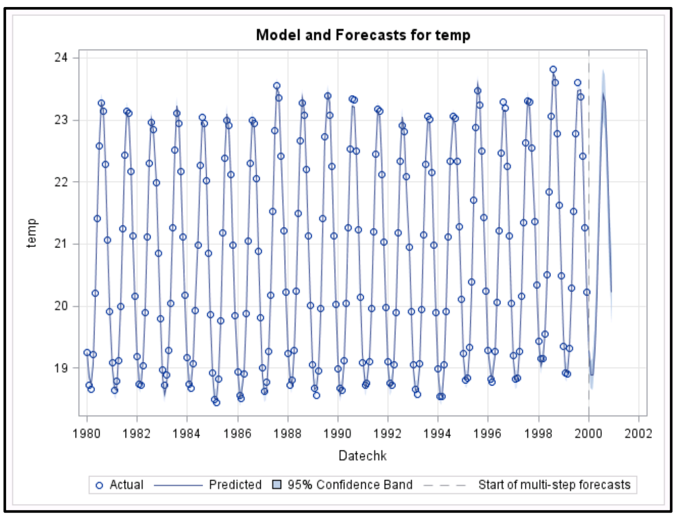
66

ESM ODS Output



67

ESM ODS Output



68

PROC ESM Syntax

```
PROC ESM DATA=SAS-data-set OUT=SAS-data-set
       OUTEST=SAS-data-set
       OUTFOR=SAS-data-set
       OUTSTAT=SAS-data-set
       OUTSUM=SAS-data-set
       SEASONALITY=n
       PLOT=option|(options)
       PRINT=option|(options)
       LEAD=n
       <options>;
       BY variables;
       ID variable INTERVAL=interval;
       FORECAST variables / MODEL=model <options>;
RUN;
```

69

Time Series Modeling Essentials – Review

- ARMA and ARMAX Model Review
- Exponential Smoothing Models Review
- UCM Models Review

70

Unobserved Components Models (UCMs)

- Also known as *structural time series models*
- Decompose time series into components:
  - trend
  - season
  - cycle
  - irregular
  - regressors
- General form:

$$Y_t = \text{Trend} + \text{Season} + \text{Cycle} + \text{Regressors}$$

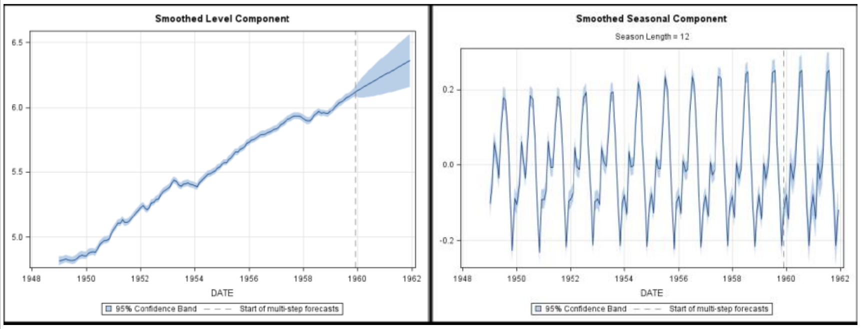
71

UCMs

- Each component captures some important feature of the series dynamics.
- Components in the model have their own models.
- Each component has its own source of error.
- The coefficients for trend, season, and cycle are dynamic.
- The coefficients are testable.
- Each component has its own forecasts.

72

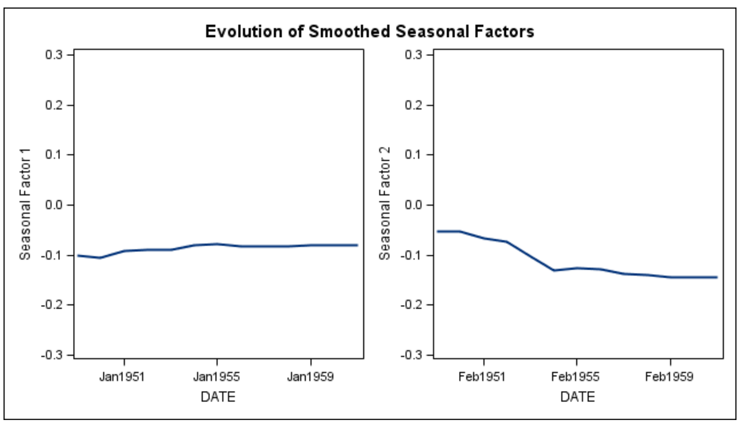
Airline Data: Component Estimates



Estimated slope of the trend curve ~ 0.01/month
(in the log scale)

73

Annual Variation in the Seasonal Effects



74

Trend Component Example

Two models for trend:

- The Random Walk Trend (RW) represents a slowly varying level without a drift in any particular direction.
- The Local Linear Trend (LL) represents a locally linear pattern with slowly varying intercept and slope.

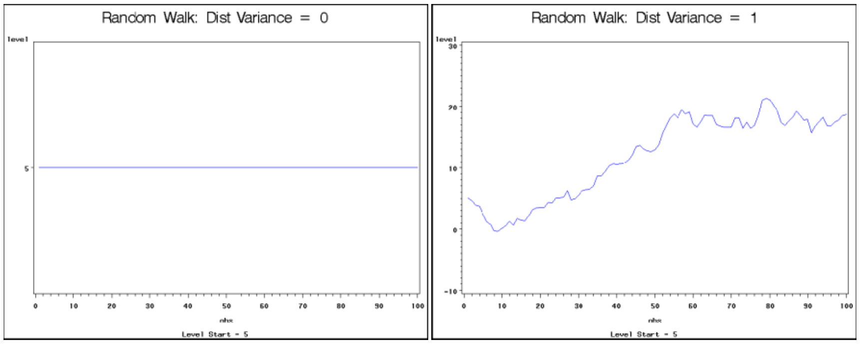
75

Random Walk Trend

$$\mu_t = \mu_{t-1} + \eta_t \qquad \eta_t \sim N(0, \sigma^2_{\mu})$$

76

Random Walk Simulation



77

Local Linear Trend

Deterministic linear trend:

$$\mu_t = \mu_0 + \beta_0 * t$$

Recursive form:

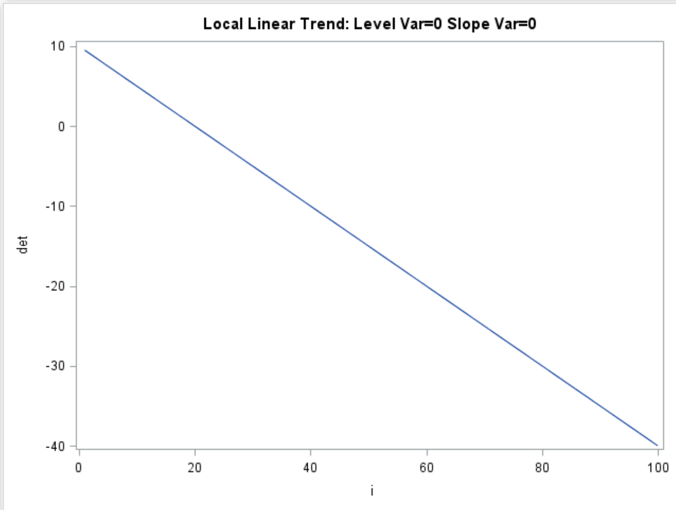
$$\mu_t = \mu_{t-1} + \beta_{t-1}$$
$$\beta_t = \beta_{t-1}$$

Local linear trend:

$$\mu_t = \mu_{t-1} + \beta_{t-1} + \eta_t \qquad \eta_t \sim \mathrm{N}(0, \sigma^2_\mu)$$
$$\beta_t = \beta_{t-1} + \xi_t \qquad \xi_t \sim \mathrm{N}(0, \sigma^2_\beta)$$

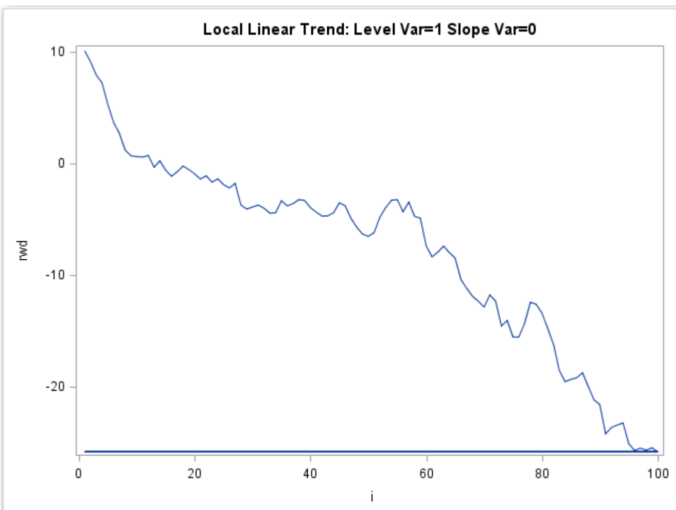
78

Local Linear Trend Simulation



79

Local Linear Trend Simulation



80

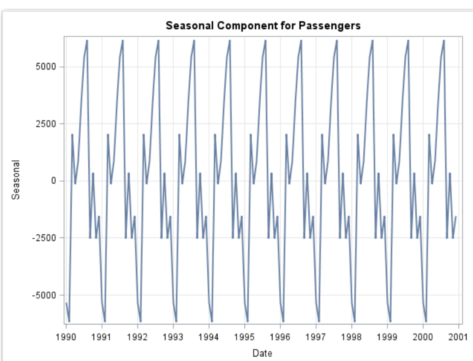
Season Component Example

1. The seasonal fluctuations are a common source of variation in the time series data.
2. The seasonal effects are regarded as corrections to the general trend of the series due to seasonal variations, and these effects sum to zero when summed over the full season cycle.
3. Therefore, a (deterministic) seasonal component γ_t is modeled as a periodic pattern of an integer period s so that the sum is as follows:

$$\sum_{i=0}^{s-1} \gamma_{t-i} = 0$$

81

Example of a (Deterministic) Seasonal Pattern (Period=12)



Seasonal Index	Seasonal Component
1	-5338.903472
2	-6177.449306
3	2037.0590278
4	-128.6951389
5	882.51736111
6	3367.1923611
7	5416.2756944
8	6168.7340278
9	-2525.282639
10	344.73819444
11	-2519.265972
12	-1526.920139

82

A General UCM

A general UCM can be described as follows:

$$y_t = \mu_t + \gamma_t + \psi_t + r_t + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{j=1}^{m} \beta_j x_{jt} + \varepsilon_t$$

$$\varepsilon_t \sim i.i.d.\, N\left(0, \sigma_\varepsilon^2\right)$$

- $\varepsilon_t, \mu_t, \gamma_t, \psi_t$, and r_t represent different stochastic components.
- The model can contain multiple seasons and cycles.
- The term $\sum_{j=1}^{m} \beta_j x_{jt}$ represents the effects of predictors.
- The term $\sum_{i=1}^{p} \phi_i y_{t-i}$ is a regression term involving the lags of the dependent variable.

83

Model Specification Syntax

A UCM is specified by describing the components in the model. For example, consider the following model:

$$y_t = \mu_t + \gamma_t + \varepsilon_t$$

It consists of the LL trend μ_t, monthly trigonometric season γ_t, and an irregular component ε_t. The corresponding syntax is as follows:

```
MODEL y;
      IRREGULAR;
      LEVEL;
      SLOPE;
      SEASON LENGTH=12 TYPE=TRIG;
```

84

continued...

A General Model Building Approach

A general modeling approach can be described as follows:

- Identify systematic components of variation in the data.
- Specify a general UCM that accommodates these components.
- Identify components that are non-stochastic. The variance of these components can be fixed at 0.
- Identify components that are not significant in explaining variation in the target. These components are candidates for removal from the model.

85

Experimentation in Data Science: Review

What Is an Experiment?

According to Merriam-Webster, this is an experiment:

1. a scientific test in which ***you perform a series of actions*** and carefully ***observe their effects*** in order to learn about something.
2. something that is done as a test
3. something that you do to see how well or how badly it works

Experiments differ from observational data in the types of conclusions that you can draw from them.

87

Observational versus Experimental

Observation: “As my son gets taller, the national debt increases.”

Incorrect conclusion: “Stop feeding your son!”

Correct conclusion: “A child’s height and the national debt increased over time.”

Correlation does not imply causation.

88

Observational versus Experimental

Experiment: “As I increase the dose of a drug by 5 mg, the symptoms decrease by 12%, on average, compared to a control group.”

With an experiment, you can evaluate the cause-effect relationship between the things that you change in the experiment and the variable that you are measuring.

Experiments enable you to identify causal drivers.

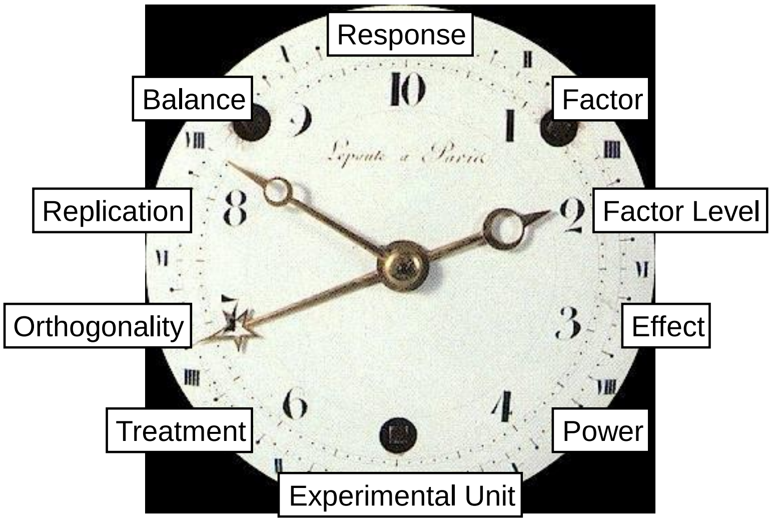
89

Experiments Enable Small-Scale Deployments

- Many business decisions should be fact-checked to assess whether they have the intended consequences and no unintended ones.
- Testing many possible scenarios on a small scale enables you to compare which is the most profitable.
- Small demonstrations of success make it easier to communicate the value to stakeholders.
- Experiments answer questions of causation.

90

Basic Terms in Design of Experiments (DOE)



91

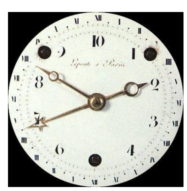
Basic Terms in DOE: Treatment

A *treatment* is a combination of all of the factors, each at one level. In a typical marketing context, a treatment constitutes a unique *offer*.

Examples:

- 2.99% Intro Rate, in a White Envelope, 4.99% Goto rate
- 0% Intro Rate, in a Gray Envelope, 7.99% Goto rate
- 2.99% Intro Rate, in a Gray Envelope, 7.99% Goto rate
- 0% Intro Rate, in a White Envelope, 4.99% Goto rate

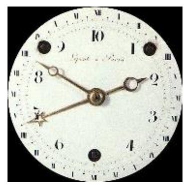
There are eight possible ***treatments*** when you have three ***factors***, each at two ***levels***.



92

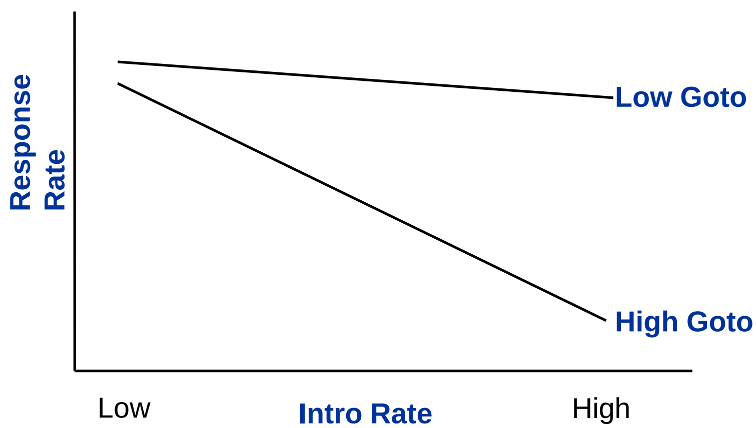
Other Terms in DOE

- An *experimental unit* is the smallest unit to which a ***treatment*** can be applied.
- *Replication* occurs when more than one ***experimental unit*** receives the same ***treatment***.
- *Power* is the probability that you can detect an ***effect***, if one exists.



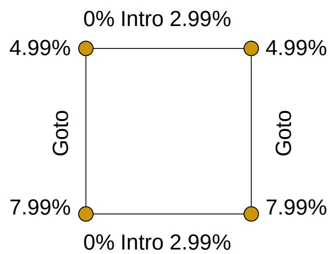
93

Detecting Interactions between Factors



94

Factorial Arrangement of the Treatments



- Permits the testing and estimation of an interaction term
- Increases the precision of estimates for the same test volumes
- Can use every individual in every test

Combinations of factor levels provide replication for individual factors.

95

Randomization

After an experiment is defined, the next step is to randomly assign treatments to experimental units.

A typical approach to randomization of *N* customers to *k* treatments involves the following steps:

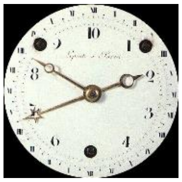
1. Define the population of interest.
2. Select a simple random sample from the population equal to the total sample size, *N* (for example, *N*=100,000).
3. Randomly partition the sample into *k* equal groups (for example, *k*=4 groups of 25,000 each).
4. Assign each group to one of the four treatments.

96

Orthogonality

Another ideal property of an experimental design is orthogonality among the elements of interest. There are at least three ways to think about the importance of this property:

- Algebraic interpretation – Matrices behave well.
- Geometric interpretation – Pictures look nice.
- Statistical interpretation – Estimates have low variance.



97

Factorial Arrangement versus OFAT

Factorial Treatment Structure

Pros

+ ***Reuses*** observations (more power for fewer exp units)
+ Tests for ***interactions***
+ Guarantees balanced and orthogonal treatment plans
+ Is an efficient way to test many factors

Cons

- Can be more complicated to set up
- Can be more complicated to sell to a nontechnical audience

98

Factorial Arrangement versus OFAT

One-Factor-at-a-Time Tests

Pros

+ Are easy to set up (A/B and Champion/Challenger tests are typical in many industries.)
+ Might yield lower per-unit printing costs
+ Have clear “control” offer, clear test offers
+ Do not require users to learn new words such as “balance” and “orthogonality”!

Cons

+/- Permit simple analysis that could be done with a pencil and paper!
- Do not allow a test for interactions
- Represent an ***inefficient*** use of experimental units

99

Blocking

Blocks are groups of experimental units that are homogeneous in some way. Typically, they represent nuisance variability.

- Examples: region, school, company
- Blocks might or might not be randomly selected.
- Because units exist in blocks, rather than being assigned to them, blocks reflect a restriction on the randomization in an experiment.

100

Covariates

Covariates are characteristics of the experimental units that are measured but cannot be assigned or imposed upon them.

- Examples: age, gender, income
- Covariates are usually not selected, but are characteristics of the units that were selected.
- Covariates often represent nuisance variability.

101

What Is the Difference?

Blocks and covariates are sometimes treated interchangeably, and the distinction is not always clear.

- Blocks are typically categorical and can be thought of as groups of experimental units. Blocks are often used as random effects in models.
  - For example, a neighborhood is a group of households.
- Covariates can be continuous or categorical and are characteristics of the individual experimental unit. Covariates are often used as (fixed) predictors in models.
  - For example, a person is male or female and is of a particular age.

102

Deploying a Model

- You know how to build a predictive model.
- You know how to score new data.
- Now you must take action based on those scores.
- How do you know whether the deployment is effective?
  - Did you identify the right cases?
  - Did the action have the desired consequence?

Experiments are an effective way to separate the impact of the model from the impact of the message.

103

Evaluate Model Deployment

- Compare actual results against expectations.
- Compare the challenger's results against the champion's.
- Did the model find the right people?
- Did the action affect their behavior?
- What are the characteristics of the cases most affected by the intervention?

104

The Data Science Lab

Carefully designed experiments enable you to evaluate many factors, interactions, and scenarios to identify key drivers.

These experiments are successful by virtue of the data scientist's ability to carefully control the factors, similar to a laboratory setting.

Data science labs can be rich with many experiments, trying out ideas on a small scale. Successful experiments can lead to changes in how the business is run.

105

Out of the Lab

Although the deployment has left the lab, it is important to continue to learn from any decisions made as a result of experiments in the lab.

Justification to continue to experiment outside the data science lab:

- unintended consequences
- population drift
- incremental response

106

Unintended Consequences

Plan:	Product recommendation to generate new sales.
Unintended consequence:	Cannibalized sales of another product, net loss of profit.
Plan:	Solicit customers that the model predicts are likely to respond.
Unintended consequence:	Your marketing approach turns off customers and drives them away.

107

Deploy on a Small Scale

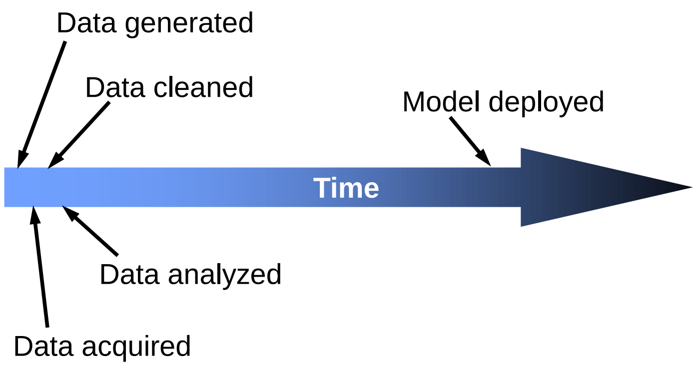
Smart move: Phase in new actions to evaluate unintended consequences without compromising the business.

One example of phasing in an intervention:

Week	Old Action	New Action
1	99%	1%
2	95%	5%
3	90%	10%
4	75%	25%
5	50%	50%

108

Scoring Pitfalls: Population Drift



109

Control Group

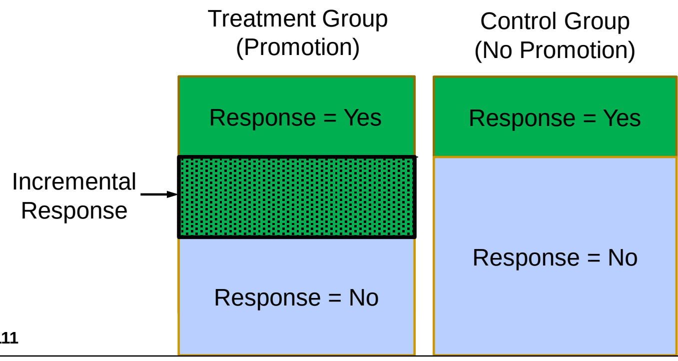
A control group is a randomly sampled group of candidate cases (customers, prospects, store locations, transactions, and so on) that are held back from a policy change, offer, or action. In medical research, the control group often receives a placebo.

This might seem unfair, at first, to hold back a potentially beneficial action. However, the control group is critical to fact-based decision making and smart business practice.

110

Example

- Treatment: the group that receives the offer.
  You assume that an incremental response exists,
  but you do not know for which customers.
- Control: the group that receives no offer. You assume that there is no known incremental response.



111

What Is an Incremental Response?

- A measure of the true effect of an action taken.
- The ***additional*** value that, in absence of the action, would not be realized.
- The most common application of incremental response modeling is in targeted marketing campaigns.
- Many other application areas are possible.

112

Four Types

Persuadables: Respond only with offer
Loyal Customers: Offer irrelevant, Likely to respond
Lost Causes: Offer irrelevant, Unlikely to Respond
Do Not Disturbs: Less likely to respond with offer

RESPONSE:	YES	NO
Offer = YES	Persuadables + Sure things	Do Not Disturbs
Offer = NO	Sure things	Lost Causes + Do Not Disturbs

113

Incremental Response / Sales Modeling

Only binary target variable:

- Probability of response for both scenarios predicted

Both binary and interval target variables:

- Probability of response for both scenarios predicted
- Difference in amount of revenue for the two scenarios predicted

114

Data Structure

- Treatment variable (required):
  - Binary
  - Does observation belong to treatment or control group?
- Response variable (required):
  - Binary target variable
  - Does the customer purchase or not?

115

Data Structure

- Outcome variable (optional):
  - Interval target variable
  - Amount of purchase
- Input variables

116

Variable Selection

- Important because incremental response modeling is relying on a double calculation
- Incremental effect = Treatment result – Control result
- Net Information Value (NIV) method by Larsen (2010):
  - intuitive
  - easy to implement
  - flexible
- Modification of the concepts of weight of evidence and information value

117

Weight of Evidence

- Y is binary (0/1) and Y=1 is the event
- Input variable X partitioned into I bins
- Weight of Evidence (WoE):

$$WOE_i = \log \frac{P(X = x_i | Y = 1)}{P(X = x_i | Y = 0)} \quad \text{for } i = 1,2,\ldots,I$$

118

Net WOE and Net IV

- Treatment = T
- Control = C

$$NWOE = \log \frac{P(X = x_i|Y = 1)_T / P(X = x_i|Y = 0)_T}{P(X = x_i|Y = 1)_C / P(X = x_i|Y = 0)_C}$$

$$NIV = \sum_i (P(X = x_i|Y = 1)_T P(X = x_i|Y = 0)_C - P(X = x_i|Y = 0)_T P(X = x_i|Y = 1)_C) \cdot NWOE_i$$

119

Penalized Net Information Value

- Penalty for the difference in NWOE between training and validation data
- Variables that improve model stability are selected
- Difference:

$$\omega = |NWOE_{train} - NWOE_{valid}|$$

- PNIV = NIV – Penalty
- PNIV is automatically used in SAS Enterprise Miner if a validation data set exists.

120

Difference Score Model

- The difference between the predictions for the treatment group model and the control group model is measured.
- The differences are called ***difference scores***.
- The difference scores are binned and ranked.
- The top subset (for example, 10%) is used as Persuadables.

121

Construction of the Predictive Model

Two alternatives exist:

- two separate models for the treatment and the control group
- one combined model for both groups

122

Two Separate Models

- Difference scores:

$$\widehat{DS}_{\,i} = (\hat{Y}_T - \hat{Y}_C)_{\,i} \;\; \text{for } i = 1,2,\dots,n$$

- Two alternatives to determine the Persuadables:
  - At least, the value of the difference score should be positive
  - More careful considerations with ranks:

$$\widehat{DS}_{\,(i)} = (\hat{Y}_T - \hat{Y}_C)_{\,(i)} \;\; \text{for } i = 1,2,\dots,n$$

123

The Combined Model

- Introduce an indicator variable T:
  - Treatment group: T=1
  - Control group: T=0
- Combined model:

$$Y = X\beta + T\gamma + (XT)\varphi$$

- Difference scores:

$$\widehat{DS}_i = \hat{Y}_T - \hat{Y}_C = \hat{\gamma} + X\hat{\varphi} \;\; \text{for } i = 1,2,\dots n$$

124

The Incremental Sales Model

- An incremental response model taking care of two targets.
- One target binary:
  - Response or not
  - Response model
- One target interval:
  - Amount of response
  - Outcome model
- Identify customers likely to spend incrementally ***with*** marketing action.

125

The Incremental Sales Model

- Selection bias due to unobserved target of amount if no response.
- Correction by use of the Heckman two-step method based on the inverse Mills ratio.

126

Model Result for the Incremental Response Model

- The treatment and control data sets are ranked in descending order by the difference scores.
- The ranked observations are partitioned into bins.
- The average predicted interval response is predicted for each bin, once for the treatment model (t_ave) and once for the control model (c_ave).
- Predicted incremental response: t_ave - c_ave.

127

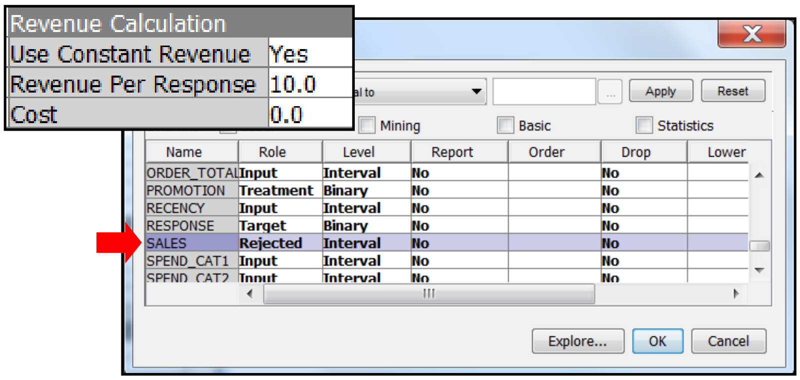
Incremental Revenue Analysis

- Constant revenue and constant cost
- Variable revenue and constant cost
- Variable revenue and variable cost

128

Constant Revenue and Constant Cost

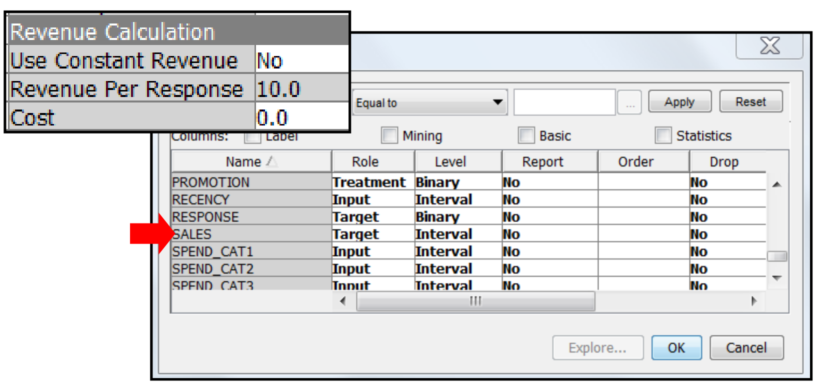
- No use of a cost variable
- No use of an interval target variable (but the property **Use Constant Revenue** would override it)



129

Variable Revenue and Constant Cost

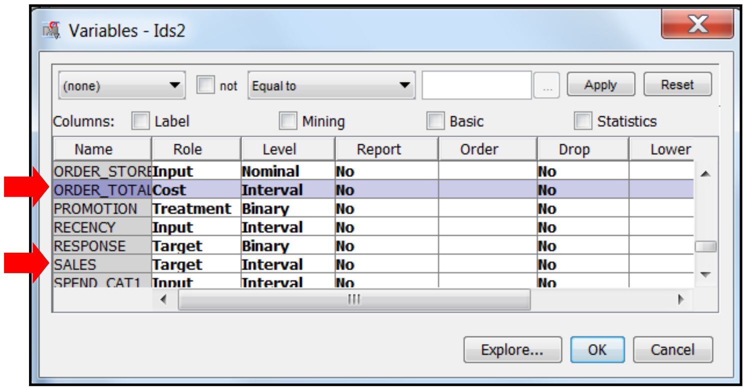
- No use of a cost variable
- Use of an interval target variable



130

Variable Revenue and Variable Cost

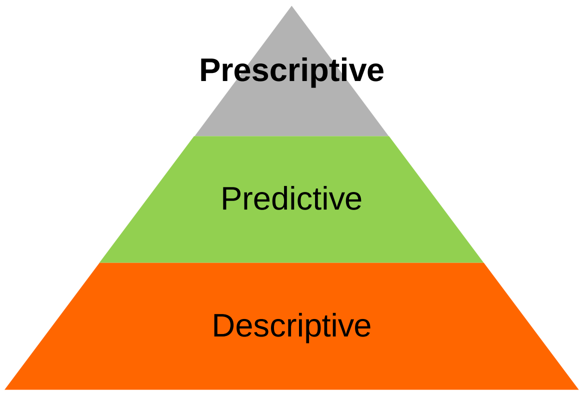
Set the role of the cost variable to **Cost** in the metadata definition.



131

Optimization Concepts for Data Science: Review

Analytical Categories and Optimization



133

Defining Mathematical Optimization Problems

General form of an optimization problem:

$$\begin{array}{ll} \min|\max & f(\mathbf{x}) \\ \text{subject to} & \boxed{\begin{array}{l} c_i(\mathbf{x}) \ \{\le,=,\ge\} \ b_i \ \ (i=1,2,\ldots,m) \\ \quad l_j \le x_j \le u_j \ \ (j=1,2,\ldots,n) \end{array}} \end{array}$$

where

- **x** is an array of *decision variables*.
- f(**x**) is an *objective function*.
- $c_1(\mathbf{x}), \ldots, c_m(\mathbf{x})$ are functions that together with the limits and bounds on **x** determine the *constraints*.

A *feasible solution* **x*** is a set of values that satisfy the constraints.

134 ...

Defining Mathematical Optimization Problems

General form of an optimization problem:

$$\begin{array}{ll} \min|\max & f(\mathbf{x}) \\ \text{subject to} & \boxed{\begin{array}{l} c_i(\mathbf{x})\ \{\le,=,\ge\}\ b_i\ \ (i=1,2,\ldots,m) \\ \quad l_j \le x_j \le u_j\ \ (j=1,2,\ldots,n) \end{array}} \end{array}$$

where

- $\mathbf{x}$ is an array of *decision variables*.
- $f(\mathbf{x})$ is an *objective function*.
- $c_1(\mathbf{x})$, ... ,$c_m(\mathbf{x})$ are functions that together with the limits and bounds on $\mathbf{x}$ determine the *constraints*.

An *optimal solution* $\mathbf{x}^*$ is a feasible solution with the minimum or maximum objective function value.

135

Defining Mathematical Optimization Problems

General form of an optimization problem:

$$\begin{array}{ll} \min|\max & f(\mathbf{x}) \\ \text{subject to} & c_i(\mathbf{x})\ \{\le,=,\ge\}\ b_i\ \ (i=1,2,\ldots,m) \\ & \quad l_j \le x_j \le u_j\ \ (j=1,2,\ldots,n) \end{array}$$

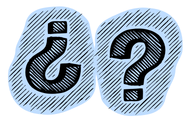
where

- $\mathbf{x}$ is an array of *decision variables*.
- $f(\mathbf{x})$ is an *objective function*.
- $c_1(\mathbf{x})$, ... ,$c_m(\mathbf{x})$ are functions that together with the limits and bounds on $\mathbf{x}$ determine the *constraints*.

An *optimal solution* $\mathbf{x}^*$ is a set of values that minimize or maximize the objective function subject to the constraints.

136

Classification of Mathematical Optimization Problems

	All continuous variables	Some integer variables	All integer variables
All linear functions	**Linear Programming (LP)**	**Mixed-Integer Linear Programming (MILP)**	**Integer Linear Programming (ILP)**
Some nonlinear functions	**Nonlinear Programming (NLP)**		

137

A Linear Programming Problem

$$\begin{aligned} &\min|\max && f_1x_1 + \ldots + f_nx_n \\ &\text{subject to} && \mathbf{Ax}\ \{\le,=,\ge\}\ \mathbf{b} \\ & && l_j \le x_j \le u_j \quad (j = 1,2,\ldots,n) \end{aligned}$$

- Each linear constraint can be either an inequality or an equation.
- Bounds can be $\pm\infty$, so x_j can be restricted to be nonnegative ($l_j = 0$ and $u_j = +\infty$) or free ($l_j = -\infty$ and $u_j = +\infty$).

138

Lagrangian Function

$$L(\mathbf{x},\boldsymbol{\lambda}) = F(\mathbf{x}) + \boldsymbol{\lambda}[\mathbf{b} - A(\mathbf{x})]$$

139

Lagrangian Function

$$L(\mathbf{x},\boldsymbol{\lambda}) = F(\mathbf{x}) + \boldsymbol{\lambda}[\mathbf{b} - A(\mathbf{x})]$$

$$\text{optimize} \rightarrow$$

$$\frac{dL}{dx_i} = 0$$

$$\frac{dL}{d\lambda_i} = 0$$

140

A Furniture-Making Problem

The cost and availability of labor, metal, and wood are as follows:

	Labor (hrs)	Metal (lbs)	Wood (ft^3)
Cost ($)	14	20	11
Availability	225	117	420

Assuming all furniture can be sold, how many **desks**, **chairs**, **bookcases**, and **bedframes** should the company produce per day so that its profit is as large as possible?

- **What should the decision variables be?**

141

Furniture-Making Problem Data

	Labor (hrs)	Metal (lbs)	Wood (ft^3)	Selling Price ($)
Desks	2	1	3	94
Chairs	1	1	3	79
Bookcases	3	1	4	125
Bedframes	2	1	4	109
Cost ($)	14	20	11	
Availability	225	117	420	

- **What should the objective be?**

Maximize NetProfit = Revenue – Cost

142 ...

Mathematical Optimization Formulations

The basic structure of a typical mathematical optimization problem formulation is shown here:

min|max objective function

subject to constraints
variable bounds

The formulation is easier to understand if it is followed by definitions for the decision variables, sets, and parameters.

143

PROC OPTMODEL Programs

The basic structure of a PROC OPTMODEL program that solves a mathematical optimization problem is shown here:

```
proc optmodel;
   /* declare sets and parameters */
   /* declare variables */
   /* declare constraints */
   /* declare objective */
   solve;
   /* print solution */
quit;
```

144

Furniture-Making Problem Index Sets

	Labor (hrs)	Metal (lbs)	Wood (ft^3)	Selling Price ($)
Desks	2	1	3	94
Chairs	1	1	3	79
Bookcases	3	1	4	125
Bedframes	2	1	4	109
Cost ($)	14	20	11	
Availability	225	117	420	

PRODUCTS

RESOURCES

```
set RESOURCES = /labor metal wood/;
set PRODUCTS = /desks chairs bookcases bedframes/;
```

145 ...

Furniture-Making Problem Parameters

	Labor (hrs)	Metal (lbs)	Wood (ft^3)	Selling Price ($)
Desks	2	1	3	94
Chairs	1	1	3	79
Bookcases	3	1	4	125
Bedframes	2	1	4	109
Cost ($)	14	20	11	
Availability	225	117	420	

PRODUCTS

RESOURCES

```
set PRODUCTS = /desks chairs bookcases bedframes/;
num selling_price {PRODUCTS} = [94 79 125 109];
```

146 ...

Furniture-Making Problem Parameters

	Labor (hrs)	Metal (lbs)	Wood (ft^3)	Selling Price ($)
Desks	2	1	3	94
Chairs	1	1	3	79
Bookcases	3	1	4	125
Bedframes	2	1	4	109
Cost ($)	14	20	11	
Availability	225	117	420	

PRODUCTS

RESOURCES

```
set RESOURCES = /labor metal wood/;
num cost {RESOURCES} = [14 20 11];
```

147

...

Furniture-Making Problem Parameters

	Labor (hrs)	Metal (lbs)	Wood (ft^3)	Selling Price ($)
Desks	2	1	3	94
Chairs	1	1	3	79
Bookcases	3	1	4	125
Bedframes	2	1	4	109
Cost ($)	14	20	11	
Availability	225	117	420	

PRODUCTS

RESOURCES

```
set RESOURCES = /labor metal wood/;
num availability {RESOURCES} = [225 117 420];
```

148

...

Furniture-Making Problem Parameters

	Labor (hrs)	Metal (lbs)	Wood (ft^3)
Desks	2	1	3
Chairs	1	1	3
Bookcases	3	1	4
Bedframes	2	1	4

PRODUCTS

RESOURCES

```
set RESOURCES = /labor metal wood/;
set PRODUCTS = /desks chairs bookcases bedframes/;
num required {PRODUCTS, RESOURCES} =
  [2 1 3 1 1 3 3 1 4 2 1 4];
```

149

Declaring an Array of Decision Variables

The syntax for declaring an array of decision variables follows the same pattern as declaring a parameter array.

```
var NumProd {PRODUCTS} >= 0;
```

Variable Name

Index Set

Bounds or Options

In the formulation that uses arrays and index sets, the decision variable **Desks** becomes **NumProd['desks']**.

150

Declaring Decision Variables

VAR *var-declaration [, … , var-declaration]* **;**

The syntax for a *var-declaration* is

name [**{** *index-set* **}** *] [var-option(s)]*

The available *var-options* are as follows:

- *>= expression*
- *<= expression*
- **INIT** *expression*
- **INTEGER**
- **BINARY**

151

Declaring an Array of Implicit Variables

The syntax for declaring an array of implicit variables is similar to the syntax for declaring a parameter array.

Variable Name

Local Dummy Parameter

Index Set

```
impvar AmountUsed {r in RESOURCES} =
   sum {p in PRODUCTS} NumProd[p]
      * required[p,r];
```

Expression

In this expression, **r** and **p** are local dummy parameters.

152 ...

Declaring an Array of Implicit Variables

```
impvar AmountUsed {r in RESOURCES} =
   sum {p in PRODUCTS} NumProd[p]
      * required[p,r];
```

```
impvar AmountUsed['labor'] =
   sum {p in PRODUCTS} NumProd[p]
      * required[p,'labor'];
impvar AmountUsed['metal'] =
   sum {p in PRODUCTS} NumProd[p]
      * required[p,'metal'];
impvar AmountUsed['wood'] =
   sum {p in PRODUCTS} NumProd[p]
      * required[p,'wood'];
```

RESOURCES

153

Declaring an Objective

The objective function for the furniture-making problem can be declared as the difference between the implicit variables **Revenue** and **TotalCost**.

Sense

Objective Name

Expression

```
max NetProfit = Revenue - TotalCost;
```

154

Declaring a Family of Constraints

Constraint Name

Local Dummy Parameter

Index Set

```
con Usage {r in RESOURCES}:
   AmountUsed[r] <= availability[r];
```

Expression

Relation

Expression

In the formulation using arrays and index sets, the constraint **Labor** becomes **Usage['labor']**.

155

Declaring a Family of Constraints

```
con Usage {r in RESOURCES}:
   AmountUsed[r] <= availability[r];
```

```
con Usage['labor']:
   AmountUsed['labor']
      <= availability['labor'];
con Usage['metal']:
   AmountUsed['metal']
      <= availability['metal'];
con Usage['wood']:
   AmountUsed['wood']
      <= availability['wood'];
```

RESOURCES

156

Data Envelopment Analysis

The Motivation Behind the Model

"To improve performance, one needs to constantly evaluate operations or processes related to producing products, providing services, and marketing and selling products."

- Joe Zhu

157

Define Efficiency

$$E_r = \frac{y_r}{y_R}$$

$y_r =$ Measurement under evaluation

$y_R =$ Possible maximum

158

Comparing Efficiency

$$E_r = \frac{z_r}{z_R}$$

z_r = Unit under evaluation

z_R = Comparison unit

159

Perspectives for Measuring Efficiency

- Output-oriented
- Input-oriented

160

Perspectives for Measuring Efficiency

- **Output-oriented** — **Expand production.**
- **Input-oriented** — **Minimize resource usage.**

161

Data Envelopment Analysis

- **Efficiency**

$$\text{Max } E = \frac{\textit{weighted sum of outputs}}{\textit{weighted sum of inputs}}$$

Linear Form

Max $E =$ *weighted sum of outputs*

s.t.

- *weighted sum of inputs = 1*
- *weighted sum of outputs - weighted sum of inputs* ≤ 0

162

...

Data Envelopment Analysis

Efficiency

Comparing Multiple Decision Making Units

Input-Oriented

$$\begin{aligned} &\min \theta \\ &\text{s.t.} \quad X\lambda \le \theta X_o \\ &\qquad\ \ Y\lambda \ge Y_o \\ &\qquad\ \ \theta \text{ free},\ \lambda \ge 0 \end{aligned}$$

Output-Oriented

$$\begin{aligned} &\max \phi \\ &\text{s.t.} \quad X\lambda \le X_o \\ &\qquad\ \ Y\lambda \ge \phi Y_o \\ &\qquad\ \ \phi \text{ free},\ \lambda \ge 0 \end{aligned}$$

163 ...

Data Envelopment Analysis

Efficiency

Input-oriented example

Comparing Multiple Decision Making Units

	DMU A	DMU B	DMU C	DMU D	Hybrid DMU
λ	?	?	?	?	?
Staff	375	300	436	310	?
Managers	4	2	5	4	?
Quality	275	300	436	280	?
Quantity	1060	975	1070	1009	?

- How efficient is DMU A?

164 ...

Data Envelopment Analysis

Input-Oriented

```
proc optmodel;

    /* Declare Variables */
    var DMU_A >=0, DMU_B >=0, DMU_C >=0, DMU_D >=0;
    var H;

    /* Declare Objective Function */
    min Resources=H;

    /* Declare Constraints */
    con Staff_IP: DMU_A*375 + DMU_B*300 + DMU_C*436 + DMU_D*310 <= 375*H;
    con Managers_IP: DMU_A*4 + DMU_B*2 + DMU_C*5 + DMU_D*4 <= 4*H;

    con Quality_OP: DMU_A*275 + DMU_B*300 + DMU_C*436 + DMU_D*280 >= 275;
    con Quantity_OP: DMU_A*1060 + DMU_B*975 + DMU_C*1070 + DMU_D*1009 >= 1060;

    /* Call the Solver */
    solve;

    /* Print Solutions */
    print DMU_A DMU_B DMU_C DMU_D;

quit;
```

$$\begin{aligned} &\min \theta \\ &\text{s.t.} \quad X\lambda \le \theta X_o \\ &\qquad\ \ Y\lambda \ge Y_o \\ &\qquad\ \ \theta \text{ free}, \lambda \ge 0 \end{aligned}$$

165 ...

Data Envelopment Analysis

Input-Oriented

```
proc optmodel;

    /* Declare Variables */
    var DMU_A >=0, DMU_B >=0, DMU_C >=0, DMU_D >=0;
    var H;

    /* Declare Objective Function */
    min Resources=H;

    /* Declare Constraints */
    con Staff_IP: DMU_A*375 + DMU_B*300 + DMU_C*436 + DMU_D*310 <= 375*H;
    con Managers_IP: DMU_A*4 + DMU_B*2 + DMU_C*5 + DMU_D*4 <= 4*H;

    con Quality_OP: DMU_A*275 + DMU_B*300 + DMU_C*436 + DMU_D*280 >= 275;
    con Quantity_OP: DMU_A*1060 + DMU_B*975 + DMU_C*1070 + DMU_D*1009 >= 1060;

    /* Call the Solver */
    solve;

    /* Print Solutions */
    print DMU_A DMU_B DMU_C DMU_D;

quit;
```

$$\begin{aligned} &\min \theta \\ &\text{s.t.} \quad X\lambda \le \theta X_o \\ &\qquad\ \ Y\lambda \ge Y_o \\ &\qquad\ \ \theta \text{ free}, \lambda \ge 0 \end{aligned}$$

166 ...

Data Envelopment Analysis

Input-Oriented

```
proc optmodel;

    /* Declare Variables */
    var DMU_A >=0, DMU_B >=0, DMU_C >=0, DMU_D >=0;
    var H;

    /* Declare Objective Function */
    min Resources=H;

    /* Declare Constraints */
    con Staff_IP: DMU_A*375 + DMU_B*300 + DMU_C*436 + DMU_D*310 <= 375*H;
    con Managers_IP: DMU_A*4 + DMU_B*2 + DMU_C*5 + DMU_D*4 <= 4*H;

    con Quality_OP: DMU_A*275 + DMU_B*300 + DMU_C*436 + DMU_D*280 >= 275;
    con Quantity_OP: DMU_A*1060 + DMU_B*975 + DMU_C*1070 + DMU_D*1009 >= 1060;

    /* Call the Solver */
    solve;

    /* Print Solutions */
    print DMU_A DMU_B DMU_C DMU_D;

quit;
```

$$\begin{aligned} &\min \theta \\ &\text{s.t.}\quad X\lambda \le \theta X_o \\ &\qquad\quad Y\lambda \ge Y_o \\ &\qquad\quad \theta \text{ free, } \lambda \ge 0 \end{aligned}$$

167 ...

Data Envelopment Analysis

Input-Oriented

```
proc optmodel;

    /* Declare Variables */
    var DMU_A >=0, DMU_B >=0, DMU_C >=0, DMU_D >=0;
    var H;

    /* Declare Objective Function */
    min Resources=H;

    /* Declare Constraints */
    con Staff_IP: DMU_A*375 + DMU_B*300 + DMU_C*436 + DMU_D*310 <= 375*H;
    con Managers_IP: DMU_A*4 + DMU_B*2 + DMU_C*5 + DMU_D*4 <= 4*H;

    con Quality_OP: DMU_A*275 + DMU_B*300 + DMU_C*436 + DMU_D*280 >= 275;
    con Quantity_OP: DMU_A*1060 + DMU_B*975 + DMU_C*1070 + DMU_D*1009 >= 1060;

    /* Call the Solver */
    solve;

    /* Print Solutions */
    print DMU_A DMU_B DMU_C DMU_D;

quit;
```

$$\begin{aligned} &\min \theta \\ &\text{s.t.}\quad X\lambda \le \theta X_o \\ &\qquad\quad Y\lambda \ge Y_o \\ &\qquad\quad \theta \text{ free, } \lambda \ge 0 \end{aligned}$$

168 ...

Data Envelopment Analysis

Input-Oriented

```
proc optmodel;

   /* Declare Variables */
   var DMU_A >=0, DMU_B >=0, DMU_C >=0, DMU_D >=0;
   var H;

   /* Declare Objective Function */
   min Resources=H;

   /* Declare Constraints */
   con Staff_IP: DMU_A*375 + DMU_B*300 + DMU_C*436 + DMU_D*310 <= 375*H;
   con Managers_IP: DMU_A*4 + DMU_B*2 + DMU_C*5 + DMU_D*4 <= 4*H;

   con Quality_OP: DMU_A*275 + DMU_B*300 + DMU_C*436 + DMU_D*280 >= 275;
   con Quantity_OP: DMU_A*1060 + DMU_B*975 + DMU_C*1070 + DMU_D*1009 >= 1060;

   /* Call the Solver */
   solve;

   /* Print Solutions */
   print DMU_A DMU_B DMU_C DMU_D;

quit;
```

$$\begin{aligned} &\min \theta \\ &\text{s.t.} \quad X\lambda \le \theta X_o \\ &\qquad\; Y\lambda \ge Y_o \\ &\qquad\; \theta \text{ free}, \lambda \ge 0 \end{aligned}$$

169 ...

Data Envelopment Analysis

Input-Oriented

```
proc optmodel;

   /* Declare Variables */
   var DMU_A >=0, DMU_B >=0, DMU_C >=0, DMU_D >=0;
   var H;

   /* Declare Objective Function */
   min Resources=H;

   /* Declare Constraints */
   con Staff_IP: DMU_A*375 + DMU_B*300 + DMU_C*436 + DMU_D*310 <= 375*H;
   con Managers_IP: DMU_A*4 + DMU_B*2 + DMU_C*5 + DMU_D*4 <= 4*H;

   con Quality_OP: DMU_A*275 + DMU_B*300 + DMU_C*436 + DMU_D*280 >= 275;
   con Quantity_OP: DMU_A*1060 + DMU_B*975 + DMU_C*1070 + DMU_D*1009 >= 1060;

   /* Call the Solver */
   solve;

   /* Print Solutions */
   print DMU_A DMU_B DMU_C DMU_D;

quit;
```

$$\begin{aligned} &\min \theta \\ &\text{s.t.} \quad X\lambda \le \theta X_o \\ &\qquad\; Y\lambda \ge Y_o \\ &\qquad\; \theta \text{ free}, \lambda \ge 0 \end{aligned}$$

170 ...

Data Envelopment Analysis

The SAS System

The OPTMODEL Procedure

Solution Summary	
Solver	LP
Algorithm	Dual Simplex
Objective Function	eff
Solution Status	Optimal
Objective Value	0.8689138577
Primal Infeasibility	0
Dual Infeasibility	0
Bound Infeasibility	0
Iterations	7
Presolve Time	0.00
Solution Time	0.00

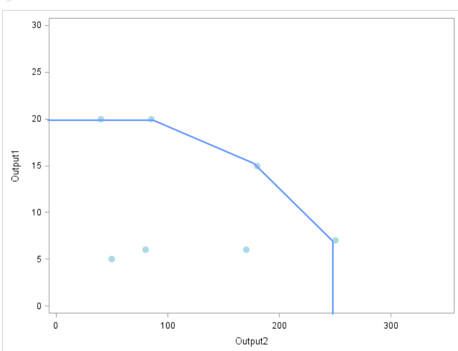
DMU_A	DMU_B	DMU_C	DMU_D
0	0.38951	0	0.67416

- The objective value provides the efficiency percentage.
- The DMU variable values represent the percentage from each DMU that is allocated to the hybrid DMU.

171 ...

Data Envelopment Analysis

Frontiers



- Data envelopment analysis creates an empirical piecewise linear frontier.
- DMUs that reside on the frontier are considered to be efficient.

172 ...

Data Envelopment Analysis

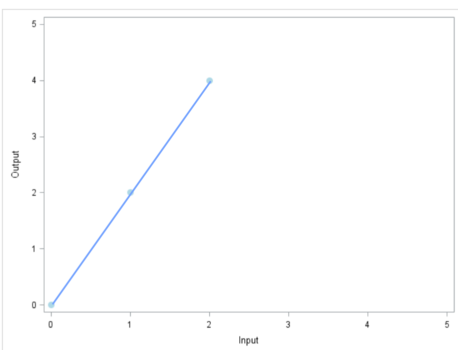
Returns to Scale

- Constant returns to scale.
- Variable returns to scale.
  - Increasing returns to scale.
  - Constant returns to scale.
  - Decreasing returns to scale.

173 ...

Data Envelopment Analysis

Returns to Scale



- *Constant returns to scale envelopment* assumes that output and input change by equivalent proportions.
  - For example, if inputs double, you expect outputs to double.

174 ...

Nonlinear Programming Problems

The general form of a nonlinear programming problem is

$$\begin{array}{lll} \min|\max & f(\mathbf{x}) & \\ \text{subject to} & c_i(\mathbf{x}) \ \{\leq,=,\geq\}\ b_i & (i=1,2,\ldots,m) \\ & l_j \leq x_j \leq u_j & (j=1,2,\ldots,n) \end{array}$$

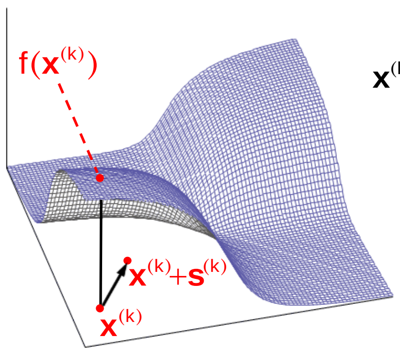
where $f(\mathbf{x})$ and $c_1(\mathbf{x})$, ... ,$c_m(\mathbf{x})$ are continuous functions.
NLPs can be classified according to their constraints.

- **unconstrained**: no constraints or bounds
- **bound constrained**: no constraints other than bounds
- **linearly constrained**: $c_1(\mathbf{x})$, ... ,$c_m(\mathbf{x})$ are linear functions.
- **nonlinearly constrained**: One or more of the functions $c_i(\mathbf{x})$ is nonlinear.

175

General Strategy for Solving NLPs

Most of the algorithms for solving NLPs are iterative.



$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)}\mathbf{s}^{(k)}$$

$\mathbf{x}^{(k)}$ iterate

$\mathbf{s}^{(k)}$ search direction

$\alpha^{(k)}$ step length

Local information at $\mathbf{x}^{(k)}$ is used to find the direction $\mathbf{s}^{(k)}$, so essentially $f(\mathbf{x})$ is approximated by a quadratic at $\mathbf{x}^{(k)}$.

176

Two-Dimensional Constrained Example

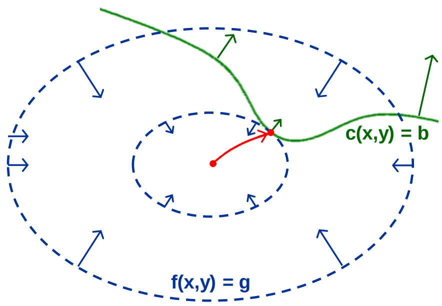
maximize $f(x,y) - \lambda^*[c(x,y) - b]$

subject to $\cancel{c(x,y) = b}$

The solid line is the set of feasible solutions.

Dashed lines are contours of the objective function.

Arrows are gradients of the objective function and constraint.



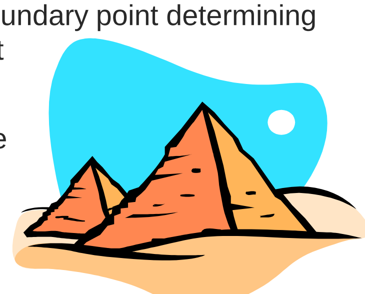
177

Interpretation of Lagrange Multipliers

- Like dual values, the (nonzero) Lagrange multiplier of a constraint can be interpreted as follows:

$$\lambda_i^* = \frac{\textbf{Change in optimal objective}}{\textbf{Unit increase in constraint limit}}$$

- This assumes that the boundary point determining the optimal solution is not over-determined.
- Exercise caution because boundaries can now be curved.



178