Chapter 10
Verification and Validation of Simulation Models

Banks, Carson, Nelson & Nicol

Discrete-Event System Simulation

Purpose & Overview

- Important and difficult task of a model developer is the verification and validation of the model developed!!
- **The goal** of the validation process is:
  - To produce a model **that represents true behavior** closely enough for decision-making purposes
    - Model is used as a substitute for the actual system for the purpose of experimenting with the system, analyzing system behavior and predicting system performance
  - To increase the model’s **credibility to an acceptable level**
    - So that the model is used by managers and other decision makers
    - Manager’s make decisions based on the recommendations from the model simulations!!!
    - Justified to look upon a model with some degree of skepticism

Purpose & Overview

- **Validation** is an integral part of model development and should not be seen as an isolated set of procedures that follow model development.
- **Verification** is concerned with building the model correctly
  - Is the model implemented correctly in the simulation software?
  - Are the input parameters and logical structures represented correctly?
- Validation is concerned with building the correct model
  - Is the model an accurate representation of the real system?
  - Achieved through the calibration of the model, an iterative process of comparing the model to actual system behavior.
- Most methods are informal subjective comparisons while a few are formal statistical procedures

Modeling-Building, Verification & Validation

- **First step** consists of observing the real system and the interactions among the various components and of collecting data on their behavior
  - Persons familiar with the system, or any other subsystem should be questioned to take advantage of their special knowledge
  - Operators, technicians, repair and maintenance personnel, engineers, supervisors and managers understand certain aspects of the system that might be unfamiliar to others
  - As model development proceeds, new questions may arise and the model developers will return to this step of learning true system structure and behavior

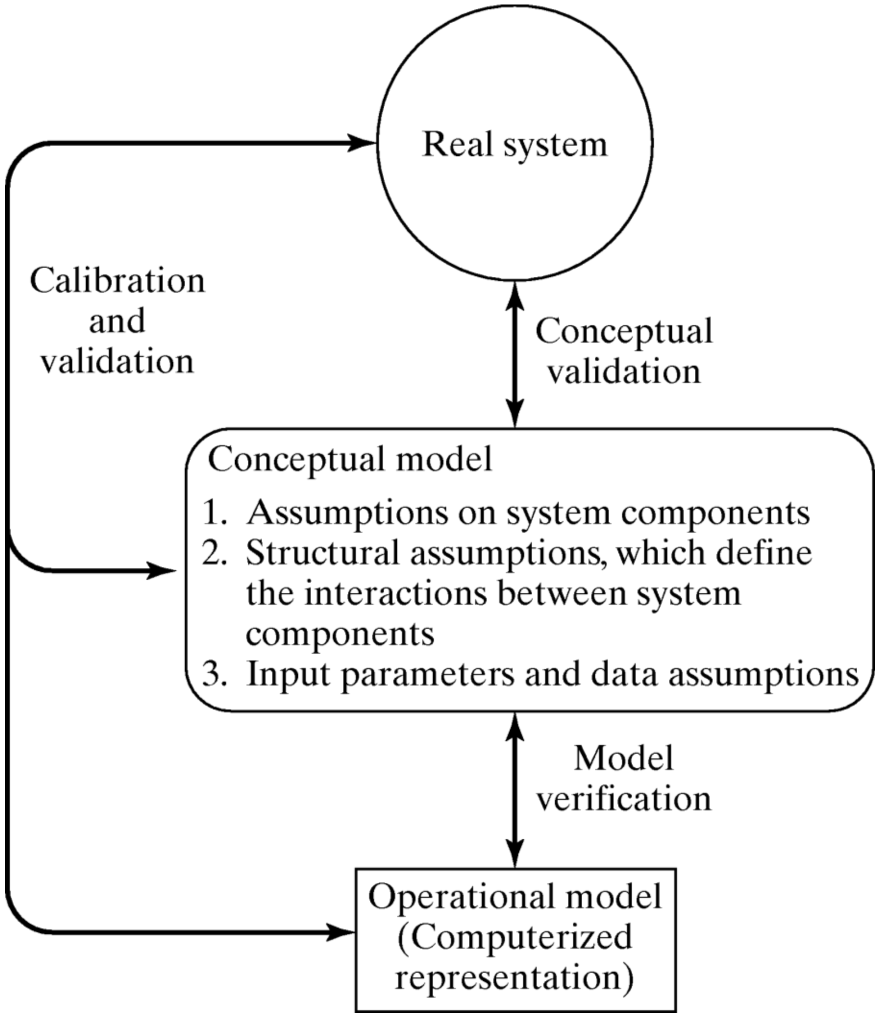
Modeling-Building, Verification & Validation

- **The second step** in model building is the construction of conceptual model
  - A collection of assumptions about the components and the structure of the system
  - Plus hypotheses about the values of model input parameters
- Conceptual validation is the comparison of the real system to the conceptual model

Modeling-Building, Verification & Validation

- **The third step** is the implementation of an operation model, usually by using simulation software and incorporating the assumptions of the conceptual model into the worldview and concepts of the simulation software
- In actuality, model building is not a linear process with three steps.
- Instead, the model builder will return to each of these steps many times while building, verifying and validating the model.

Modeling-Building, Verification & Validation



Verification

- ***Purpose:*** ensure the conceptual model is reflected accurately in the computerized representation.
- Many common-sense suggestions, for example:
  - Have someone else check the model.
  - Make a flow diagram that includes each logically possible action a system can take when an event occurs.
  - Closely examine the model output for reasonableness under a variety of input parameter settings. (Often overlooked!)
  - Print the input parameters at the end of the simulation, make sure they have not been changed inadvertently.
  - If the model is animated, verify what is seen in the animation imitates the actual system

Verification

- Common-sense suggestions (continued ..):
  - Make the operational model as self-documenting as possible. Give precise definition of every variable used and a general description of the purpose of each sub model, procedure, component
  - The interactive run controller (IRC) (or debugger) is an essential component of successful simulation model building.
  - Graphical interfaces are recommended for accomplishing verification and validation.

Other Important Tools

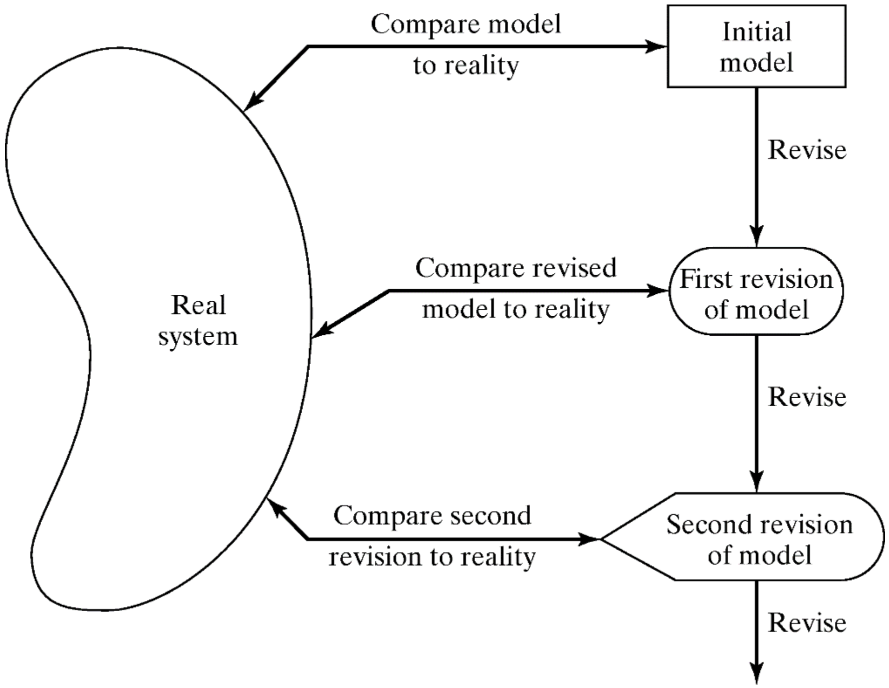
- **Documentation**
  - A means of clarifying the logic of a model and verifying its completeness
  - If a model builder writes brief comments in the operational model, plus definitions of all variables and parameters, plus descriptions of each major section of the operational model, it becomes much simpler for someone else, or the model builder at a later date, to verify the model logic
- **Use of a trace**
  - A detailed printout of the state of the simulation model over time.

Examination of Model Output for Reasonableness [Verification]

- ***Example:*** A model of a complex network of queues consisting many service centers.
  - Response time is the primary interest, however, it is important to collect and print out **many statistics in addition to response time**.
    - Two statistics that give a quick indication of model reasonableness are **current contents** and **total counts**, for example:
      - If the current content grows in a more or less linear fashion as the simulation run time increases, it is likely that a queue is unstable
      - If the total count for some subsystem is zero, indicates no items entered that subsystem, a highly suspect occurrence
      - If the total and current count are equal to one, can indicate that an entity has captured a resource but never freed that resource.
    - Compute certain long-run measures of performance, e.g. compute the long-run server utilization and compare to simulation results

Calibration and Validation

- **Validation:** the overall process of comparing the model and its behavior to the real system.
- **Calibration:** the iterative process of comparing the model to the real system and making adjustments.



Calibration and Validation

- **A possible criticism** of the calibration phase is that the model has been validated only for one data set, that the model has been “fitted” to one data set.
- **One way to alleviate** this is to collect a new set of system data to be used at the final stage of validation, after the model has been calibrated by using the original system data set, a “final” validation is conducted using the second system data set.
- **The modeler returns to calibration** if unacceptable discrepancies between the model and real system are discovered in the final validation effort
- Validation is not an either/or proposition, no model is every totally representative of the system under study.

Calibration and Validation

- No model is ever a perfect representation of the system
  - The modeler must weigh the possible, but not guaranteed, increase in model accuracy versus the cost of increased validation effort.
- **Three-step approach, Naylor and Finger** [1967]:
  - Build a model that has high face validity.
  - Validate model assumptions.
  - Compare the model input-output transformations with the real system's data.

High Face Validity

- **First goal** is to construct a model that appears reasonable on its face to model users
- **Ensure a high degree of realism:** Potential users should be involved in model construction (from its conceptualization to its implementation).
- Potential users and knowledgeable persons can identify model readiness and deficiencies. They can also be involved in the calibration process.
- **Sensitivity analysis** can also be used to check a model's face validity, i.e., model behaves as expected or not.
  - *Example:* In most queueing systems, if the arrival rate of customers were to increase, it would be expected that server utilization, queue length and delays would tend to increase.

Validate Model Assumptions

[Calibration & Validation]

- Two General classes of model assumptions:
  - **Structural assumptions:** how the system operates and involves simplifications and abstraction of reality.
  - **Data assumptions:** reliability of data and its statistical analysis.
- *Bank example:* customer queueing and service facility in a bank.
  - *Structural assumptions*, e.g., customer waiting in one line versus many lines, served FCFS versus priority.
  - *Data assumptions*, e.g., inter-arrival time of customers, service times for commercial accounts.
    - Verify data reliability with bank managers.
    - Identify the distribution, estimate the parameters, test correlation and goodness of fit for data (see Chapter 9 for more details).

Validate Input-Output Transformation

[Calibration & Validation]

- **Goal:** Validate the model's ability to predict future behavior
  - The ultimate test of a model and the only objective test of the model.
  - The structure of the model should be accurate enough to make good predictions for the range of input data sets of interest, not just for one input data set.
- In this phase of the validation process, the model is viewed as an input-output transformation
- One possible approach: use historical data that have been reserved for validation purposes only.
- **Criteria:** use the main responses of interest for validating the model.

Bank Example [Validate I-O Transformation]

- **Example**: A bank currently has one drive-in window serviced by one teller, only one or two transactions are allowed. It is planning to expand this service
  - Data collection: 90 customers during 11 am to 1 pm (typical rush hour)
    - Observed service times $\{S_i, i = 1,2, \ldots, 90\}$.
    - Observed interarrival times $\{A_i, i = 1,2, \ldots, 90\}$.
  - Data analysis led to the conclusion that:
    - Interarrival times are exponentially distributed with rate $\lambda = 45$
    - Service times are assumed to be $N(1.1, 0.2^2)$

The Black Box

[Bank Example: Validate I-O Transformation]

- A model was developed in close consultation with bank management and employees
- Model assumptions were validated
- Resulting model is now viewed as a “black box”:

Input Variables

Uncontrolled variables, X

Possion arrivals
$\lambda = 45/hr$: $X_{11}, X_{12}, \ldots$
Services times,
$N(D_2, 0.2^2)$: $X_{21}, X_{22}, \ldots$

Controlled Decision variables, D

$D_1 = 1$ (one teller)
$D_2 = 1.1$ min
(mean service time)
$D_3 = 1$ (one line)

Model “black box”
$f(X,D) = Y$

Model Output Variables, *Y*

Primary interest:
Y_1 = teller’s utilization
Y_2 = average delay
Y_3 = maximum line length

Secondary interest:
Y_4 = observed arrival rate
Y_5 = average service time
Y_6 = sample std. dev. of service times
Y_7 = average length of time

Comparison with Real System Data

[Bank Example: Validate I-O Transformation]

- Real system data are necessary for validation.
  - System responses should have been collected during the same time period (from *11*am to *1*pm on the same Friday.)
- Compare the average delay from the model Y_2 with the actual delay Z_2:
  - Average delay observed, $Z_2 = 4.3$ minutes, consider this to be the true mean value $\mu_0 = 4.3$.
  - When the model is run with generated random variates X_{1n} and X_{2n}, Y_2 should be close to Z_2.
  - Six statistically independent replications of the model, each of *2*-hour duration, are run.

Hypothesis Testing

[Bank Example: Validate I-O Transformation]

- Compare the average delay from the model Y_2 with the actual delay Z_2 (continued):
  - Null hypothesis testing: evaluate whether the simulation and the real system are *the same* (w.r.t. output measures):

$$H_0: E(Y_2) = 4.3 \text{ minutes}$$

$$H_1: E(Y_2) \neq 4.3 \text{ minutes}$$

    - If H_0 is not rejected, then, there is no reason to consider the model invalid
    - If H_0 is rejected, the current version of the model is rejected, and the modeler needs to improve the model

Hypothesis Testing

[Bank Example: Validate I-O Transformation]

- Results of six replications of the bank model

Replication	Y_4 Arrivals/Hr	Y_5 Minutes	Y_2 Avg. Delay
1	51	1.07	2.79
2	40	1.12	1.12
3	45.5	1.06	2.24
4	50.5	1.10	3.45
5	53	1.09	3.13
6	49	1.07	2.38
		Sample Mean	2.51
		Standard Dev	0.82

Hypothesis Testing

[Bank Example: Validate I-O Transformation]

- Conduct the "*t* test" (Uses Student –t distribution)
  - Chose level of significance ($\alpha = 0.05$) and sample size ($n = 6$), see result in Table.
  - Compute the same mean and sample standard deviation over the n replications:

$$\bar{Y}_2 = \frac{1}{n}\sum_{i=1}^{n} Y_{2i} = 2.51 \text{ minutes} \qquad S = \sqrt{\frac{\sum_{i=1}^{n}(Y_{2i} - \bar{Y}_2)^2}{n-1}} = 0.82 \text{ minutes}$$

  - Compute test statistics:

$$|t_0| = \left|\frac{\bar{Y}_2 - \mu_0}{S/\sqrt{n}}\right| = \left|\frac{2.51 - 4.3}{0.82/\sqrt{6}}\right| = 5.34 \;>\; t_{critical} = 2.571 \quad \text{(for a 2-sided test)}$$

  - Hence, reject H_0. Conclude that the model is inadequate.
  - Check: the assumptions justifying a *t* test, that the observations (Y_{2i}) are normally and independently distributed.

Hypothesis Testing

[Bank Example: Validate I-O Transformation]

- The modeler realized that the original model contained two unstated assumptions:
  - When a car arrived to find the window immediately available, the teller began service immediately
  - There is no delay between one service ending and the next beginning, when a car is waiting
- It was found that these assumptions were not exactly correct due to:
  - The teller attending other duties when there is no car waiting
  - The cars have to physically move to the teller position that was not accounted for

Hypothesis Testing (Revisited)

[Bank Example: Validate I-O Transformation]

- Results of six replications of the bank model (Revisited)

Replication	Y_4 Arrivals/Hr	Y_5 Minutes	Y_2 Avg. Delay
1	51	1.07	5.37
2	40	1.12	1.98
3	45.5	1.06	5.29
4	50.5	1.10	3.82
5	53	1.09	6.74
6	49	1.07	5.49
		Sample Mean	4.78
		Standard Dev	1.66

Hypothesis Testing (Revisited)

[Bank Example: Validate I-O Transformation]

- Conduct the "*t* test" (Uses Student –t distribution) with new values
  - Chose level of significance (α = *0.05*) and sample size (*n* = *6*), see result in Table.
  - Compute the same mean and sample standard deviation over the n replications:

$$\bar{Y}_2 = \frac{1}{n}\sum_{i=1}^{n} Y_{2i} = 4.78 \text{ minutes} \qquad S = \sqrt{\frac{\sum_{i=1}^{n}(Y_{2i} - \bar{Y}_2)^2}{n-1}} = 1.66 \text{ minutes}$$

  - Compute test statistics:

$$|t_0| = \left|\frac{\bar{Y}_2 - \mu_0}{S/\sqrt{n}}\right| = \left|\frac{4.78 - 4.3}{1.66/\sqrt{6}}\right| = 0.710 \; < \; t_{critical} = 2.571 \;\; \text{(for a 2-sided test)}$$

  - Hence, accept H_0. Conclude that the model is adequate

Hypothesis Testing

[Bank Example: Validate I-O Transformation]

- Similarly, compare the model output with the observed output for other measures:
  $Y_4 \leftrightarrow Z_4$, $Y_5 \leftrightarrow Z_5$, and $Y_6 \leftrightarrow Z_6$

Student –t Distribution

- Student's distribution arises when (as in nearly all practical statistical work) the population standard deviation is unknown and has to be estimated from the data.
- Confidence intervals and hypothesis tests rely on Student's *t*-distribution to cope with uncertainty resulting from estimating the standard deviation from a sample, whereas if the population standard deviation were known, a normal distribution would be used.
- Published in 1908 by William Gosset under the pseudonym Student

Type I and II Error

[Validate I-O Transformation]

- Type I error (α): (Error of first kind)
  - Rejecting H0 when H0 is true (False Positive)(False Alarm)
  - **Error of rejecting a valid model**.
  - Controlled by specifying a small level of significance α.
- Type II error (β): (Error of Second Kind)
  - Failure to reject H0 when H1 is true (False Negative)(A Miss)
  - **Failure to reject an invalid model** (Error of accepting a model as valid when it is invalid)
  - Controlled by specifying critical difference and find the *n*.
- For a fixed sample size *n*, increasing α will decrease β.

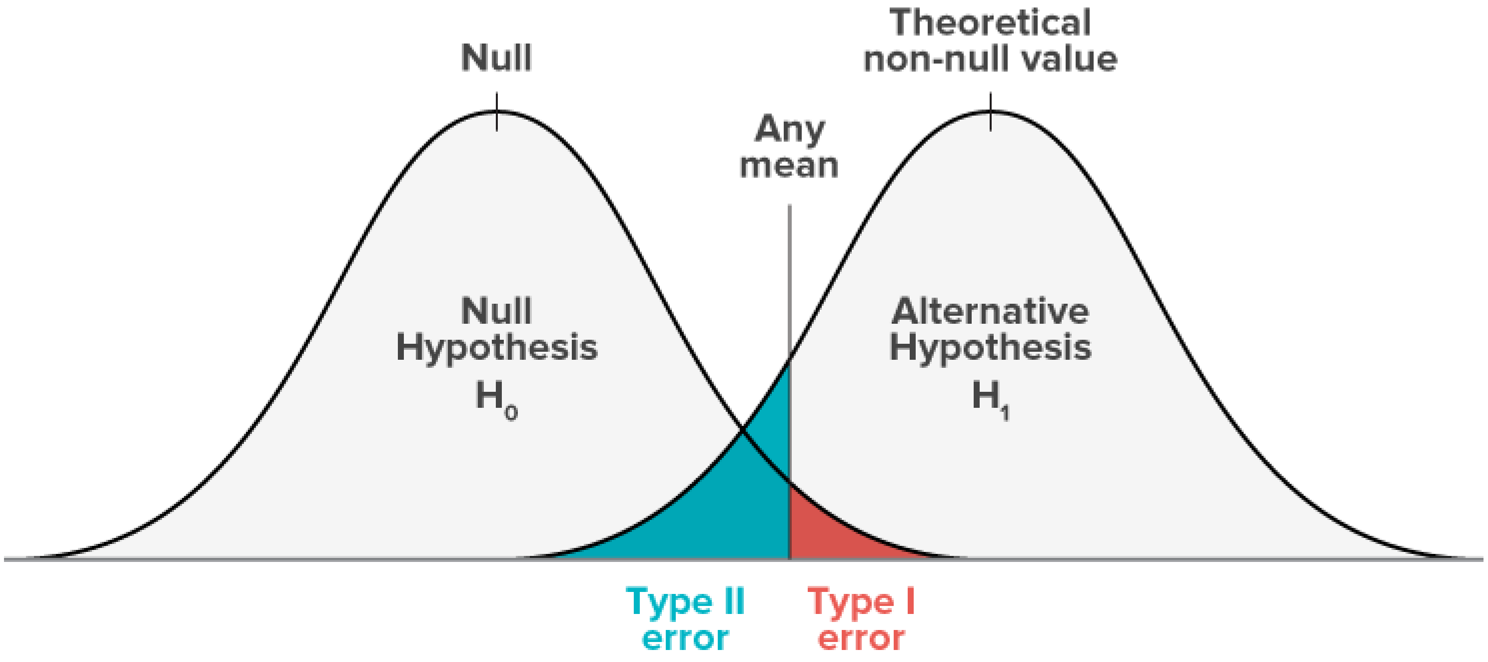
Type II Error

- For validation, the **power of the test** is the probability of detecting a departure from H_0:
  - **Power of the test** = Probability[detecting an invalid model] = $1 - \beta$
  - β = P(Type II error) = P(failing to reject $H_0 | H_1$ is true)
  - Consider failure to reject H_0 as a strong conclusion, the modeler would want β to be small.
  - Value of β depends on:
    - Sample size, *n*
    - The true difference, δ, between *E(Y)* and μ: $\delta = \dfrac{|E(Y) - \mu|}{\sigma}$
- In general, the best approach to control β error is:
  - Specify the critical difference, δ.
  - Choose a sample size, *n*, by making use of the operating characteristics curve (OC curve), which are graphs of the probability of Type II error β versus δ for a given sample size *n.*

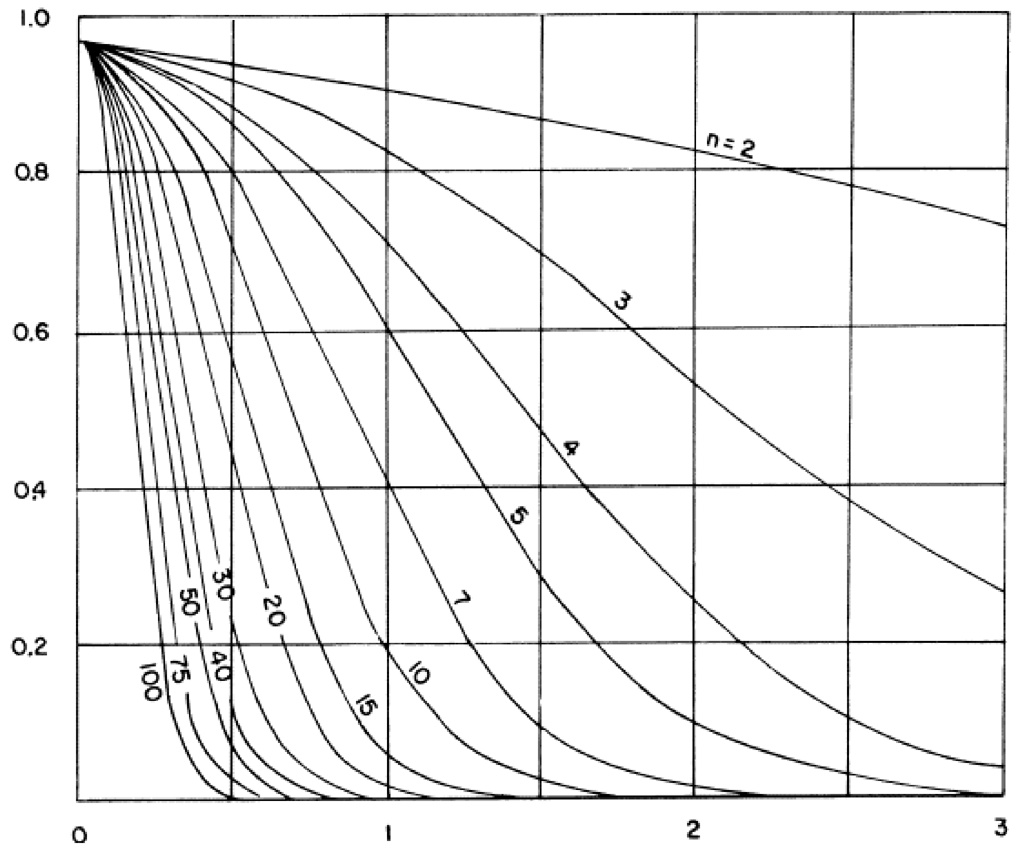
Hypothesis Testing

Statistics: Hypothesis Test	Null Hypothesis is True	Null Hypothesis is False
Reject Null Hypothesis	Type I Error	Correct
Fail to Reject Null Hypothesis	Correct	Type II Error

Type I and II errors



Operating Characteristic (OC) Curves



Probability of not detecting a difference (β)

Standardized difference (δ) for α=0.05

Confidence Interval (CI) Testing

[Validate I-O Transformation]

- **Confidence interval testing:** evaluate whether the simulation and the real system are close enough (ε).
- If Y is the simulation output, let $\mu = E(Y)$ and μ_0 the actual performance measure. The confidence interval (C.I.) for μ is:

$$\bar{Y} \pm t_{\alpha/2,n-1} S / \sqrt{n}$$

- Validating the model:
  - Suppose the C.I. does not contain μ_0:
    - If the best-case error is > ε, model needs to be refined.
    - If the worst-case error is ≤ ε, accept the model.
    - If best-case error is ≤ ε, but worst-case > ε, additional replications are necessary.
  - Suppose the C.I. contains μ_0:
    - If either the best-case or worst-case error is > ε, additional replications are necessary.
    - If the worst-case error is ≤ ε, accept the model.

Confidence Interval Testing

[Validate I-O Transformation]

- Bank example: $\mu_0 = 4.3$, and "close enough" is ε = *1* minute of expected customer delay.
  - A *95%* confidence interval, based on the *6* replications (Slide 22) is *[1.65, 3.37]* because:

$$\bar{Y} \pm t_{0.025,5} S / \sqrt{n}$$

$$2.51 \pm 2.571(0.82 / \sqrt{6})$$

  - Falls outside the confidence interval, the best case |*3.37 – 4.3*| = *0.93 < 1 (=* ε*)*, but the worst case |*1.65 – 4.3*| = *2.65 > 1(=* ε*)*, additional replications are needed to reach a decision.

Confidence Interval Testing

[Validate I-O Transformation]

- Bank example: $\mu_0 = 4.3$, and "close enough" is ε = *1* minute of expected customer delay.
  - A *95%* confidence interval, based on the *6* replications (Slide 25) is *[3.03, 6.52]* because:

$$\bar{Y} \pm t_{0.025,5} S / \sqrt{n}$$

$$4.78 \pm 2.571(1.66 / \sqrt{6})$$

  - CI contains μ_0
  - The worst case |*6.52 – 4.3*| = *2.22 > 1 (=* ε*)*, and the best case |*3.03 – 4.3*| = *1.27 > 1(=* ε*)*, additional replications are needed to reach a decision.

Using Historical Output Data

[Validate I-O Transformation]

- An alternative to generating input data:
  - Use the actual historical record.
  - Drive the simulation model with the historical record and then compare model output to system data.
  - In the bank example, use the recorded interarrival and service times for the customers $\{A_n, S_n, n = 1,2,\ldots\}$.
- Procedure and validation process: similar to the approach used for system generated input data.

Using a Turing Test

[Validate I-O Transformation]

- Use in addition to statistical test, or when no statistical test is readily applicable.
- Utilize persons' knowledge about the system.
- For example:
  - Present 10 system performance reports to a manager of the system. Five of them are from the real system and the rest are "fake" reports based on simulation output data.
  - If the person identifies a substantial number of the fake reports, interview the person to get information for model improvement.
  - If the person cannot distinguish between fake and real reports with consistency, conclude that the test gives no evidence of model inadequacy.

Summary

- Model validation is essential:
  - Model verification
  - Calibration and validation
  - Conceptual validation
- Best to compare system data to model data, and make comparison using a wide variety of techniques.
- Some techniques that we covered (in increasing cost-to-value ratios):
  - Insure high face validity by consulting knowledgeable persons.
  - Conduct simple statistical tests on assumed distributional forms.
  - Conduct a Turing test.
  - Compare model output to system output by statistical tests.