An Introduction to Experimental Software Engineering

Fernando Brito e Abreu (fba@di.fct.unl.pt)
Universidade Nova de Lisboa (http://www.unl.pt)
QUASAR Research Group (http://ctp.di.fct.unl.pt/QUASAR)

Português técnico cá e lá …

<table>
<thead>
<tr>
<th>Brasil</th>
<th>Portugal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planejamento</td>
<td>Planeamento</td>
</tr>
<tr>
<td>Correlado</td>
<td>Correlacionado</td>
</tr>
<tr>
<td>Doutorado</td>
<td>Doutoramento</td>
</tr>
<tr>
<td>Gerenciamento</td>
<td>Gestão</td>
</tr>
<tr>
<td>Treinamento</td>
<td>Formação profissional</td>
</tr>
<tr>
<td>Pesquisa / Pesquisador</td>
<td>Investigação / Investigador</td>
</tr>
<tr>
<td>Usuário</td>
<td>Utilizador</td>
</tr>
<tr>
<td>Planilha</td>
<td>Folha de cálculo</td>
</tr>
<tr>
<td>Banco de dados</td>
<td>Base de dados</td>
</tr>
<tr>
<td>Escopo</td>
<td>Âmbito</td>
</tr>
<tr>
<td>Experimento</td>
<td>Experiência</td>
</tr>
</tbody>
</table>

Tudo o resto é igual, à parte a pronúncia 😊
What is ... Experimental Software Engineering?

- Is a branch of Software Engineering where, by means of experimentation we want to validate hypotheses raised by induction (and abduction), aiming at building theories that will allow us to:

  - help understand the virtues and limitations of methods, techniques and tools, namely by assessing current SE claims

  - express quantitatively the cause–effect relationships among sw process characteristics (resources and activities) and sw product characteristics

Sw development as a feedback loop ...
Supports arguments concerning the suitability, limits, costs and risks, inherent to software engineering tools and techniques, with experimental evidence

Facilitator of the evolution of the Software Engineering body of knowledge as a quality driven process

In line with the requirements of the highest levels of maturity models (e.g. CMMI)

There are no “Laws” in Software Engineering

Software Engineering is traditionally qualitative

Software engineering is full of:
- “Theories” about effectiveness of software engineering practices, methods and techniques
- Unsubstantiated claims about efficiency of engineering practices, methods and techniques
Some Software Engineering “theories”

ABOUT THE PRODUCT:

- Cohesion should be maximized and coupling should be minimized
  - Is this what practitioners are doing in practice?

- The complexity of a software system increases non linearly with its age (“Lehman’s “Law” of Software Evolution)

- A decade ago: OOSD improves modularity upon PSD (Procedural Software Development)

- Nowadays: AOSD improves systems modularity upon OOP

More Software Engineering “theories”

ABOUT THE PROCESS:

- Accurate effort estimates can be produced without a detailed design (e.g. using Function Points analysis)

- Software inspections are more efficient than testing

- Agile processes lead to shorter development cycles in the long term (until final deployment is fully achieved)

- More controlled processes lead to an higher effectiveness in defect removal
More Software Engineering “theories”

ABOUT IT SERVICE MANAGEMENT:
- Some defect types take longer to correct than others
- The distribution of defect types is not uniform
- Aspects such as culture/nation, business area, licensing level (customer value) or size of the user community influence the bug solving process
  - Is there a concordance in the ordering between fault impact (user perspective) and urgency (support perspective)?
  - Does that concordance vary from country to country?
  - Does business area (e.g., public administration, banking, military, education, utilities) have an influence on incident priority?
  - Is there a relation between licensing levels and the volume of incidents? (by country, by incident type, by business area)
- How can we assess these “theories” or evaluate these claims?

Evaluating the claims

- **Engineering method** – Prototype, test, and improve the solution until it requires no further improvement!
  - Trial and error approach 😏
  - When should we stop? 😐
- **Empirical method**
- **Analytical method**
Evaluating the claims

- Engineering method

- Empirical method – Model proposed and evaluated through empirical studies
  - The evaluation can follow different strategies, such as surveys, case studies or controlled experiments
  - Criteria for stopping evaluation can be defined in advance 😊
  - Systematic approaches (with a well-defined process) to validation exist 😊
  - Results preciseness depends on collecting a large sample 😊

- Analytical method

- Analytical method - Develop a formal theory and compare its derived results with empirical observations
  - Formal theories are often hard to conceive and express 😐
  - Requires less observations than in the empirical method 😊
Empirical strategies

Survey

- **Objectives**: Descriptive (of population), explorative

- **Process**:
  - Document possible relationships (e.g. build taxonomy)
  - Poll after events have occurred (post-mortem)
  - Collect data through interviews or questionnaires

- **Example**:
  - Attempt to determine major problems encountered in doing program maintenance by sending a survey to subscribers of a practitioners’ journal asking respondents to rank a list of problem types by order of importance

Empirical strategies

Case study / action research

- **Objectives**: Comparison, phenomena interpretation

- **Process**:
  - Identify key factors that affect outcome of activity and then document the activity (inputs, constraints, resources, and outputs)
  - Collect data in work environment or real world situation

- **Example**:
  - For one project observe how practitioners use a given tool. Train then in the use of a tool. Observe how proficiency improved and build an interpretation for the phenomenon
Empirical strategies

Controlled experiment

- **Objectives:** Confirm theories or conventional wisdom, explore relationships, evaluate accuracy of models, validate measures

- **Process:**
  - Controlled investigation of an activity - identify the key factors and manipulate them in order to document their effects on outcomes
  - Data collected from subject performance

- **Example:**
  - Indentation study, where different groups are assigned maintenance activities based upon code indented differently

---

Comparison of empirical strategies

<table>
<thead>
<tr>
<th>FACTOR</th>
<th>TYPE</th>
<th>Survey</th>
<th>Case Study</th>
<th>Experiment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execution control</td>
<td>None/Low</td>
<td>Medium</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>Measurement control</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>Investigation cost</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>Ease of Replication</td>
<td>Medium</td>
<td>Low</td>
<td>Medium ...</td>
<td></td>
</tr>
</tbody>
</table>

- **Execution control** describes how much control the researcher has over the study
- **Measurement control** is the degree to which the researcher can decide upon which measures to be collected and to include or exclude during the study
The Scientific method

- It is a fundamental technique used by scientists to raise hypothesis and produce theories

- **Assumption**: world is a cosmos not a chaos
  - Scientific knowledge is **predictive** (positivist philosophy)
  - **Cause and effect** relationship exist
  - Knowledge in an area is expressed as a set of **theories**
  - Theories are raised based upon **hypothesis**
  - Hypothesis are tested through **experiments**

- The scientific method progresses through a series of steps

---

Steps in the Scientific Method (i/iii)

- **1 - Observe facts**
  - **Fact** means the "quality of being actual" or "a piece of information presented as having objective reality"

- **2 - Formulate hypothesis**
  - An **hypothesis** is a tentative theory that has not been tested (knowledge before experimental work performed)
  - Formulation can be performed through:
    - **induction** (generalization of observed facts)
    - **abduction** (suggestion that something could be)
3 - Test the hypothesis

- Build experiments to see if the hypothesis holds
  - Collect data
  - Check that measurements are valid and eliminate outliers
  - Perform statistical tests
  - Document results to communicate to colleagues (results packaging)

- Use hypothesis to make predictions and compare them with newly observed facts
- Experiments can only prove that an hypothesis is false
- If unsure revise hypothesis (step 2) in light of new experiments or observations

4 - Raise a theory

- After extensive experimentation corroborating the hypothesis
- A theory is a conceptual framework that explains existing facts or predicts new facts

5 – Express a law

- Law is a theory or group of theories that has been widely confirmed
- Confirmation can be obtained with intensive “in vivo” evidence
- A law should delimit its own application scope
  - e.g. Newton’s laws (holds for velocities much less than speed of light)
- Laws (as well as theories) are open to rebuttal

Question:

- Do we do experimentation in Software Engineering?
Survey on experimental practices in Software Engineering  [Sjøberg et al., 2005]

- 5453 articles in 12 major journals and conferences in software engineering (1993-2002)
- Only 1.9% (103) reported controlled experiments
- 14 series of experiments (only 6 performed independently)
- 5 series included partial rejection of the claims of the initiating experiment
- Only 1 of the rejections was reported by the initial authors

Other surveys report less than 10% of experiment-based validation papers in software engineering and computer science [Glass et al., 2002; Ramesh et al., 2004]

Why is experimentation not common practice in Software Engineering?

The five fears of experimental validation

- Shareholders - Commercial fear
- Project managers - Budgeting fear
- Team members - Evaluation fear
- Software engineers - Misinterpretation fear
- Researchers - Apathy fear (not any more 😊)

(F. Brito e Abreu, 1997)
Why is experimentation not common practice in Software Engineering?

- Traditional scientific method isn't applicable
- Current Level of experimentation is good enough
- Experiments cost too much
- Demonstrations will suffice
- There's too much noise in the way
- Experimentation will slow progress
- Technology changes too fast
- Software developers not trained in importance of scientific method are not sure how to analyze data

(Walter Tichy, 1998)

Replication

- **Experiments replication** is required for wide acceptance of theories (e.g. F&DA). Why?

- A single experiment can't be expected to provide definitive evidence on an issue, namely because threats to validity may bias the results
  - Replication can be used to remove those biases

- Fear of lack of originality of the replicated experiment?
  - Major venues of experimentation encourage replication 😊
Replication in Empirical Software Engineering Research

Many fundamental results in Software Engineering suffer from threats to validity that can be addressed by replication studies. The primary goal of this workshop is to raise the perceived value of replication work by creating both recognition for, and awareness of, replication studies. The workshop aims to encourage revisiting results, including those that have long been accepted but which in fact have only weak empirical support. In addition, the workshop seeks to identify and suggest solutions for recurring practical problems in selecting, designing, and performing replication studies. The workshop also seeks to advance the state of research reporting techniques and tool development and deployment, with a focus on making experiments repeatable and tools more reusable. By providing a venue in which researchers can discuss tools, methods, results and philosophical foundations of replication, this workshop will help to advance the empirical methods and scientific rigor of the Software Engineering community.

More info in:
http://sequoia.cs.byu.edu/reser2010

Replication

- Replicating experiments is harder than it seems (the tacit knowledge problem), specially if people is involved
  - Not easy to repeat an experiment under the same conditions, if the human factor has a strong influence
  - However, if the sample is large, the individual influences get averaged and then cancelled

- Guidelines for experimentation could certainly help mitigating those difficulties!
  - You can find a set of proposed software engineering experiment reporting guidelines in the ESE course page at FCT/UNL: http://ctp.di.fct.unl.pt/phd/ese
Replication requirements

- To perform replication we need access to the experiments’ data samples
  - Early attempts to build up experimental data databases in the software field had limited success
  - a good exception is:

- Experimentation in general requires samples of considerable size relative to real world software development projects namely including process data (efforts, schedules, defect data, etc).
  - This often implies a relation among university and software companies, often encompassing non-disclosure agreements …

Who’s teaching ESE?

Several CS departments have recently started dedicated ESE courses or include this topic very strongly in their SE courses:

- Colorado State University (USA)
- George Mason University (USA)
- University of Texas at Austin (USA)
- University of Maryland (USA)
- Walden University (USA)
- Worcester Polytechnic Institute (USA)
- University of Calgary (Canada)
- Oregon State University (USA)
- Florida Atlantic University (USA)
- University of Otago (New Zealand)
- University of Sannio (Italy)
- University of Oulu (Finland)
- Linköpings Universitet (Sweden)
- Lund University (Sweden)
- University of Skövde (Sweden)
- Kaiserslautern University (Germany)
- NTNU University (Norway) – 2005 version
- Technical University of Sydney (Australia)
- Universidade Nova de Lisboa (Portugal)
- …
Some landmark references?

- A journal ...
  - Empirical Software Engineering: An International Journal

- ... and a book:
  - Experimentation in Software Engineering: An Introduction

Competence centers in ESE

- Lund University (Sweden)
  - SERG - Software Engineering Research Group

- UNIVERSITY OF MARYLAND (USA)
  - ESEG – Experimental Software Engineering Group

- Fraunhofer Institut Experimentelles Software Engineering
  - Founded in 1996, is directed by Prof. Dieter Rombach
USA Government support

- CeBASE – sponsored initiative by the National Science Foundation

- University of Maryland College Park
- University of Southern California
- Fraunhofer Center for Experimental Software Engineering - Maryland
- University of Nebraska-Lincoln
- Mississippi State University

Tutorial outline

- Requirements Definition
- Design Planning
- Experiment Execution
- Data Analysis
- Results Packaging
Experimental Process

- Present a process model that acts as:
  - a guideline for conducting experiments
  - a framework for supporting experiments comparison

- Integration of contributions from:
  - experiment conduction guidelines
  - experiment reporting guidelines
  - models for representing experimental data

- Proposed in:
This is the whole process model, but we also need to understand the ontology behind it 😊

Overview of the experimental process
Experimental steps

- **Requirements definition**
  - Which are the goals of the experiment?

- **Design planning**
  - Formalize the goals in research hypotheses
  - Define who, when and how the experiment will be conducted

- **Experiment execution**
  - Prepare and collect data in a controlled way

- **Analysis and Interpretation**
  - Check measurements and analyze data to test hypotheses

- **Results Packaging**
  - Interpret and document results to communicate to colleagues

Problem statement

- **what** is the problem that the experiment will address
- **where** can it be observed
- **when** can it be observed
- **who** can observe it or is concerned with it
- **how** does the problem impact those experiencing it
- **why** solving the identified problem is important

*I keep six honest serving-men
(They taught me all I knew);
Their names are What and Why and When
And How and Where and Who.*

*Rudyard Kipling*, Literature Nobel Prize, 1907
Objectives definition

- Those objectives can be summarized using an abstract template with just five items (next slide)

- This systematic description:
  - promotes an explicit goal-driven approach for experimental work
    - helps the researcher delimiting the experiment's boundaries and focusing on its essential goals
  - provides an abstract matching mechanism
    - helps other researchers in searching experiments that are relevant in their own area of concern
    - this is common practice in other sciences (e.g. medicine)

Objectives definition template

Analyze <Object of study>
- Object of study – entity being studied

For the purpose of <Purpose>
- Purpose – intention of study

With respect to their <Quality focus>
- Quality focus – primary effect under study

From the point of view of the <Perspective>
- Perspective – viewpoint of interpretation of results

In the context of <Context>
- Context – environment in which experiment being run
Objectives definition template

Object of study
- e.g. product, process, resource, model, theories

Purpose
- e.g. compare two different techniques, characterize a learning curve, analyze the impact of using a tool, replicate a study

Quality focus
- e.g. effectiveness, cost, reliability, maintainability, portability

Perspective
- e.g. developer, program manager, customer, user, researcher

Context
- e.g. practitioners in a softwarehouse, students in a course

Example 1: (Indentation study)

- Analyze program indentation levels
- For the purpose of evaluation
- With respect to their effectiveness in program comprehension
- From the point of view of the researcher
- In the context of undergraduate students in a programming course
Example 2: (ITSM study)

- **Analyze** SLAs in providing IT services
- **For the purpose of** comparing system and model-based compliance verification
- **With respect to their** effectiveness and efficiency
- **From the point of view of the** service provider
- **In the context of** a network of financial self-service terminals

Context definition

The context includes:

- where will the experiment take place (environment)
  - university course, softwarehouse, client company

- who will be involved in the experiment
  - those subject should be characterized (e.g. number, experience, workload)

- which software artifacts are used in the experiment
  - Those artifacts should be characterized (e.g. type, size, complexity, application domain)
Context definition

- A specified context:
  - has its own benefits, costs, and risks
  - facilitates the comparability among different studies
  - allow practitioners evaluating to which extent results obtained in a study apply to their specific needs
  - determines our ability to generalize from the experimental results

- At this phase, an informal assessment of the context is sufficient
  - The context will be detailed during the design phase
Experiment steps

- **Requirements definition**
  - Which are the goals of the experiment?

- **Design planning**
  - Formalize the goals in research hypotheses
  - Define who, when and how the experiment will be conducted

- **Experiment execution**
  - Prepare and collect data in a controlled way

- **Analysis and Interpretation**
  - Check measurements and analyze data to test hypotheses

- **Results Packaging**
  - Interpret and document results to communicate to colleagues

---

**BASIC CONCEPTS**
Population and sample

- A **population** is the universe of all the elements from which a sample can be drawn for an experiment
  - Examples: Java practitioners, UML 2.0 models, programs produced in C#

- A **sample** is a number of elements drawn from a population
  - A sample is usually a subset of the population
  - Examples: My company’s programmers, internal projects developed, …

What are samples for?

- Sample items are used to test hypotheses about the population

- A sample is wished to be representative of the population
  - E.g. in digital communications the analog signal (voice) is sampled, that is, its frequency spectrum (amplitude at several frequencies) is sampled at regular time intervals in order to convert it to digital form
**Subjects**

- **Subject (aka case, aka entity)** is a population or sample member, from which we collect data in the experiment
  - **Examples**: a specific practitioner, a given UML model, a given program

- Subjects in ESE are usually **persons** or **artifacts**
  - E.g. developers, users, tools, project deliverables

- Subjects of the same kind are characterized by a common set of **variables**

**Variables**

- Variables are features (descriptors) of subjects that we measure, control, or manipulate in research
  - Each variable is expected to represent a given subject characteristic or “quality”

- A variable has:
  - an **identifier**
  - a **role** in our experimental research (independent or dependent) – to be discussed later
  - a **measurement scale type**
  - a **statistical distribution**
Factors and Treatments

- A **factor level** is a distinct value of a given discrete independent variable (aka factor)
  - E.g. 3 factor levels for the **language** variable \{C++, Java, C#\}

- A **treatment** is a tuple of factor levels to be considered in the experiment
  - e.g. for factors **language** and **IDE** we could have the following treatments
    - \{(Java, Eclipse), (Java, NetBeans), (C#, VisualStudio), … \}

Types of empirical studies

- **Correlation studies**
  - Attempts to determine how much of a relationship (association or collinearity) exists among variables
    - It can not establish cause & effect (e.g., life-time expectancy vs. literacy)
    - To measure the association strength we use a correlation coefficient

- **Regression studies**
  - Attempts to produce an estimation model that allows to determine the values of a dependent (outcome) variable based upon a set of independent variables (aka predictors or regressors)
    - Model parameters are determined by calibration (regression)
    - Models can be linear or non-linear
    - Outcome variables can be continuous or discrete (e.g., logistic regression)
Types of empirical studies

- **Controlled experiments** (*in vitro*)
  - Attempts to establish cause & effect
  - Independent variables are manipulated (controlled / blocked)
  - Subjects are randomly assigned to groups

- **Quasi-experiments** (or field or natural experiment)
  - The values of independent variables are usually predefined (are “found” in the sample)
  - Subjects cannot be randomly assigned to groups and / or independent variables cannot be fully controlled

In this course we will concentrate in these latter two types of empirical studies

Independent variables
(aka explanatory variables or factors)

- These are the ones whose effect on the dependent variable(s) we want to investigate
  - Are those that are (expected to be) manipulated in experimental research

- **Examples:**
  - Programming language, Development environment, Design size, Experience/background of subjects
Dependent variables
(aka outcome or response variables)

- Those whose effect of the independent ones we want to assess in experimental research
- For each hypothesis we usually only have one dependent variable

Examples:
- Productivity, Design complexity, Effort to produce a given deliverable, Project schedule, Defects found in code inspection, System faults in operation (e.g. MTBF, MTTR)

Exercise: Identify the independent and dependent variables …
HYPOTHESES FORMULATION

Bringing up hypotheses

- Observation is required
  - Informal
    - E.g. looking around and asking questions about possible causes for known problems or for noticeable successes in previous projects
  - Formal
    - Survey related papers or books (e.g. start on this course’s bibliography)
    - Use some qualitative approaches such as the laddering technique or cognitive maps to derive some hypothetical causalities perceived by domain experts

- Top-down decomposition
  - Often the hypotheses formulation process progresses from an abstract to a concrete version, whose level of detail is appropriate for clearly identifying the adequate variables
Abstract vs. Concrete hypotheses

Example 1:
- Method A produces higher quality code than Method B (abstract)
- Using Method A will result in fewer defects being discovered during integration testing than using Method B. (concrete)

Example 2:
- Programmer will create better test suites using Tool A than using Tool B. (abstract)
- Test suites created by programmers using Tool A will have higher branch cover than test suites created by programmers using Tool B. (concrete)

Example Hypotheses

Abstract or Concrete?
1. Programmers using object-oriented programming will produce high quality programs?
2. Testers will obtain better code coverage using the C++Test tool from Parasoft than testers not using this tool.
3. Programmers using a visual language are more productive than programmers using a procedural language.
4. Functional programs are more understandable than procedural programs.
5. Java programs are easier to maintain than C programs.
6. Object-oriented languages encourage more reuse than procedural programs.
Hypotheses formulation

This formulation is formally defined by two complementary statements:

- **Null hypothesis** ($H_0$)
  - No cause-effect can be observed

- **Alternative hypothesis** ($H_1$)
  - Some effect appears to be present

When we accept the null hypothesis, we reject the alternative one and vice-versa!

Null hypothesis ($H_0$)

- States that there is no statistically significant difference between treatments (tools, techniques, methods, …) or there is no underlying trend or pattern in the outcome variable due to factors

- The only reasons for differences in the observations are coincidental (due to random error)
Alternative hypothesis ($H_1$)

- States that there is a statistically significant difference between treatments or an underlying trend or pattern in the outcome variable due to factors.

False positives

- If we inadequately reject a null hypothesis (accept the alternative one), we say we have a **false positive**
  - *This is called a Type I error (to be seen again later)*
  - *False positives occur when you think you have observed an effect when in fact that is not sustainable*

\[
P(\text{Type I error}) = P(\text{reject } H_0 \mid H_0 \text{ true})
\]

- **False positives may be due to inadequate definition of the confidence level, sampling bias, coincidental data,** …
False negatives

- If we inadequately accept a null hypothesis (reject the alternative one), we say we have a **false negative**
  - This is called a **Type II error** (to be seen again later)
  - False negatives occur when you could not observe an effect that in fact occurs

\[
P(\text{Type II error}) = P(\text{accept } H_0 \mid H_0 \text{ false})
\]

- **False negatives are due to reduced test power and are usually less “harmful” than false positives**

Test power

- Many different statistical tests can be used to test hypotheses
  - The result of a test evaluates the outcome of an experiment

- **Statistical tests have several characteristics such as pre-conditions for application and test power**
  - The **power of a statistical test** is the probability that it will reveal a true effect

\[
\text{Power} = 1 - P(\text{Type II error}) = P(\text{reject } H_0 \mid H_0 \text{ false})
\]
Purpose of experiment

- Usually, the experimenter would like to reject the null hypothesis with as high confidence as possible
  - In other words, he wants to find out some causality in a given phenomenon
  - This is what the positivist approach to Science is all about (believing the world is not a chaos)

- To reject $H_0$ the researcher must conduct an experiment or quasi-experiment where he obtains subjects data that show (by applying an adequate statistical test) that there is a significant difference between the treatments

VARIABLES SELECTION
GQM (Goal-Question-Metric)

- Often variables of interest (e.g., *ability) are not directly measurable and we have to identify indirect metrics
  - We can use the Goal-Question-Metric approach to find them

1. List major research goals (usually expressed upon characteristics of the object of study)
2. Derive questions from each goal that must be answered to determine if the goals are being met
3. Decide what to measure to answer questions adequately

Two examples follow:

CS Graduate Student Admission

Major goals
- Assess quality of graduate students admitted
- Evaluate effectiveness of admission process

Questions
- What is the admission criteria?
- Are the criteria working?
- What is graduate student quality?
CS Graduate Admission

Measures
- CS degree, GRE score > 1300, positive letters of recommendation
- % students admitted who complete degree (MSc and PhD)
- Academic record: Comprehensive exam performance, course grades
- Scientific record: number of published papers, theses, citation index

Software Testing Process

Major goals
- Improve the design testing process

Questions
- How is design testing done currently?
- How much time does design testing take?
- How much does design testing cost?
- How efficient is the design testing process?
- How effective is the design testing process?
Software testing process

Measures

- Number of tests (reviews) per module
- Time spent testing/module
- Cost of design testing
- Weighted errors found / cost of design testing
- % of specification errors found in design testing
- % of design errors found during integration testing

SUBJECTS SELECTION
Choosing material subjects (artifacts)

Materials should:
- Vary in experimental difference being tested
- Be representative
- Be of appropriate level of difficulty – not too hard or too easy
  - Do not use toy programs (e.g., < 100 lines)
- Be comparable across different experimental conditions

Choosing human subjects

- Should be both representative and relatively uniform
- How to select a representative sample?
  - Subjects should be uniform in characteristics and abilities
    - E.g. there are large individual differences between programmers
  - How can a homogeneous set be selected from a heterogeneous population?
- Should reflect characteristics of population
  - Students often selected because of convenience (class, grade)
  - What is a typical professional programmer? A software engineer?
  - Do students reflect these characteristics?
  - Can we assess abilities before selection?
Choosing human subjects

- Should have enough subjects to reflect diversity
- Programmers characterized by diversity requires a large number of subjects
  - Solution:
    1. Group by abilities in some manner
    2. Use a within groups design – each subject exposed to several treatments

Sampling types

- Sampling is the act of constructing a sample
  - This can be performed in distinct ways

- Probability Sampling
  - For each population item there is a known selection probability
  - Subtypes:
    - Simple Random Sampling, Stratified Random Sampling, Cluster Sampling, Multistage Sampling, Systematic Sampling

- Judgment Sampling
  - We don’t known the selection probability of population items
  - Subtypes:
    - Quota Sampling, Purposive Sampling
Probability sampling types

- Simple Random Sampling
  - Subjects are selected randomly from the population

- Stratified Random Sampling
  - The population is divided into a number of groups (strata) with a known distribution between the groups
  - Random sampling is then applied within each strata

- Systematic Sampling
  - Every n\textsuperscript{th} member of the population is selected as a subject
    - "n" is the rounded integer of the ratio population size / sample size

- Other: Cluster Sampling, Multistage Sampling

Judgment sampling types

- Convenience sampling
  - The nearest/available and most convenient subjects are selected

- Quota Sampling
  - The population is divided into strata
  - Convenience sampling is used within each strata

- Other: Purposive Sampling, …
Case Study: Components factory

Introduction

- Cleanroom software development
- Almost 100 post-graduate (2nd cycle – Bologna model) students involved with formal roles assignment
  - Software Engineering course labs assignment at FCT/UNL
- Several stages
  1. Collaborative requirements specification
  2. Components interfaces standardization
  3. Prototype production
  4. Components production (Java)
  5. Code inspection and repair
  6. Product assembly (components integration)
  7. Integration testing
  8. Product assessment (client)
  9. Product documentation and delivery

Case Study: Components factory

Collected data

- Practitioners’ expertise
  - Based upon academic record (around 30 independent assessments for each practitioner)
- Individual components complexity (internal and external)
  - Collected with Eclipse Metrics Plugin
- Defects found in components’ inspection
  - Categorized with the help of a checklist
- Failures found in component integration (product assembly)
  - Professor played the client’s role
- Components assembly complexity (internal and external)
  - Collected with Eclipse Metrics Plugin
Forces influencing quality in component development and integration

Research Hypothesis

<table>
<thead>
<tr>
<th>Context</th>
<th>Component development, during design</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hypothesis</td>
<td>Developers expertise level conditions their abstractions utilization choices</td>
</tr>
<tr>
<td>Collected data</td>
<td>Software Engineering labs project data, combined with subjects’ grades</td>
</tr>
<tr>
<td>Rationale</td>
<td>Do the best programmers tend to use different abstractions, when compared to “less talented ones”? How does competence relate to the complexity of their design options?</td>
</tr>
</tbody>
</table>
Research hypothesis

<table>
<thead>
<tr>
<th>Context</th>
<th>Component development, during code inspections</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hypothesis</td>
<td>Code reviewers expertise conditions the success of code reviews, during components development process</td>
</tr>
<tr>
<td>Case study</td>
<td>Software Engineering labs project data, combined with subjects’ grades</td>
</tr>
<tr>
<td>Rationale</td>
<td>Expert developers make better reviews, as the essential skills that make them good developers are basically the same as the ones required for being good reviewers</td>
</tr>
</tbody>
</table>

EXPERIMENT DESIGN
Application of treatments (between groups)

- We can split our sample and apply different treatments to different subjects
  - This is called the **between groups experimental design**
  - There are several concerns that should be considered when deciding how to conduct subjects' split into groups
  - One group can correspond to a group of subjects to which any treatment at all is applied – this is called the **control group**

**Example** *(hypothesis: CASE tools improve productivity)*

- **Experimental subjects** (subjects applying treatments)
  - Groups of projects using CASE tool X, CASE tool Y, CASE tool Z
- **Control subjects** (subjects not using treatment)
  - Group of projects not using CASE tools

Application of treatments (within groups)

- We apply different treatments to the same subjects
  - This is called the **within groups experimental design**
  - There are several concerns that should be considered when deciding **when to apply** treatments
  - Still, we can split our sample in different groups, so the concerns regarding group splitting still hold

**Example**: *(hypothesis: environment impacts reliability)*

- *Group A first applies treatment X, then applies treatment Y*
- *Group B first applies treatment Y, then applies treatment X*
  - Treatment X = {Java, Eclipse}
  - Treatment Y = {C#, Visual Studio}
Within-groups design
(aka repeated measures design)

- Several treatments are applied to the same subject
  - We then take repeated measures from the same subject

- Good aspects:
  - Requires less subjects
  - Reduces the variability due to subject differences
  - Reduces background “noise”

Within-groups design
(aka repeated measures design)

- Bad aspects:
  - Often it is not feasible to apply several treatments to the same subject (often the case for Software Engineering projects)
  - Results may be affected by ordering: learning/practice effects, fatigue, etc

- Mitigation solutions:
  - Counterbalance order (can be difficult)
  - Train until asymptote on learning curve is reached (time-consuming)
  - Test on different days
**Between-groups design**  
(aka independent samples design)

- Only one treatment is applied to each subject
  - But the same treatment is applied to several subjects

- Good aspects:
  - More feasible to apply only one per subject in the case of Software Engineering projects
  - No side effects due to ordering or learning effects

<table>
<thead>
<tr>
<th>Bad aspects:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requires more subjects</td>
</tr>
<tr>
<td>Augments the variability due to subject differences</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mitigation solutions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eliminate confound subjects with pre-tests to guarantee that variations in results are simply due to individual differences between subjects</td>
</tr>
<tr>
<td>Do random sampling and random allocation to treatments</td>
</tr>
</tbody>
</table>
Within versus between groups designs

<table>
<thead>
<tr>
<th>Within Groups Design (aka repeated measures design)</th>
<th>Between Groups Design (aka independent samples design)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pros</td>
<td>Cons</td>
</tr>
<tr>
<td>Smaller samples can be used</td>
<td>Validity threats due to learning effects</td>
</tr>
<tr>
<td>No learning effects</td>
<td>Require larger samples</td>
</tr>
</tbody>
</table>

Other experimental designs

- Matched design
  - Subjects are different, but matched (treated as repeated measures)

- Mixed designs
  - for complex designs with more than one independent variable
  - some conditions repeated, some not.
Simple Designs

- 1 factor with 2 treatments
- 1 factor with more than 2 treatments
- 2 factors with 2 treatments
- More than 2 factors with 2 treatments

1 Factor with 2 Treatments

- Completely randomized design (between groups)
  - Example
    - Subjects test a program
    - Each subject either uses or does not use a coverage tool

- Paired comparison design (within groups)
  - Example
    - Subjects test two programs using a coverage tool
    - Each subject tests one program using the tool and one program not using the tool
1 Factor with > 2 Treatments

- Completely randomized (between groups)
  - Example
    - Each subject is given a comprehension quiz upon a version of a program indented in one of 3 levels (0, 3, 6 spaces)

- Randomized complete block design (within groups)
  - Example
    - Each subject is given a comprehension quiz upon 3 programs each indented differently in one of 3 levels (0, 3 or 6 spaces)

2*2 factorial design

- 2 factors each with 2 treatments

- Completely randomized (between groups)
  - Each subject sees only one level of each factor (one tuple)

- Incomplete block design (within groups)
  - More than one treatment (e.g. 2) is applied to each subject

- Complete block design (within groups)
  - All treatments are applied to each subject
Experiment Execution

Fernando Brito e Abreu (fba@di.fct.unl.pt)
Universidade Nova de Lisboa (http://www.unl.pt)
QUASAR Research Group (http://ctp.di.fct.unl.pt/QUASAR)

Overview of the experimental process
Experiment steps

- **Requirements definition**
  - Which are the goals of the experiment?

- **Design planning**
  - Formalize the goals in research hypotheses
  - Define who, when and how the experiment will be conducted

- **Experiment execution**
  - Prepare and collect data in a controlled way

- **Analysis and Interpretation**
  - Check measurements and analyze data to test hypotheses

- **Results Packaging**
  - Interpret and document results to communicate to colleagues

Abstract

- **Requirements definition**
- **Design planning**
- **Experiment execution**
  - Collection clearance
  - Motivation of participants
  - Data collection
  - Data validation
  - Problem reporting
  - Data analysis
  - Results packaging
COLLECTION CLEARANCE

Collection clearance

- Stands for the permission to proceed with data collection activities

- Researchers want to:
  - use real-world data (e.g. for representativeness sake) and therefore try to get it from companies
  - publish their research findings (publish or perish!)

- Companies may be interested in experiment results but:
  - collected data may be considered sensitive and potentially harm their commercial interests if publicly disclosed
Non-disclosure agreement

- The previous conflict of interests may be solved if researchers sign a **non-disclosure agreement**

- Usually these agreements state how researchers are authorized to disclose experimental results externally; this may include the obligation to:
  - not cite the name of the company where the collection took place (source confidentiality)
  - mask details regarding the process that may lead to source identification (e.g. granting the anonymity of participants)
  - only publish grouped data, not raw ones

Non-disclosure agreement

- Non disclosure agreements may be the only way to proceed, but they have a very perverse effect
  - Since raw data is not made available to other researchers, experiments cannot be independently replicated 😞

- Experiments replication is pointed out as the Achilles’ heel in Experimental Software Engineering …
Recruit participants

- It is difficult to recruit professional practitioners to participate in experiments
- This situation often leads to the usage of students as surrogates for those practitioners
  - A systematic review showed that, out of 5488 participants in 113 controlled experiments in software engineering, less than 10% were professionals:
  
<table>
<thead>
<tr>
<th>Professional practitioners</th>
<th>Students (mostly undergraduates)</th>
<th>Faculty members, post-docs or unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td>9,4%</td>
<td>86,8%</td>
<td>3,8%</td>
</tr>
</tbody>
</table>

### Motivate students
- The experimental work is usually carried out within a course they follow
- Their reward is often of an academic nature, such as part of the course grade
- If the experiment is not smartly hidden from them, then:
  - Perform collection as non-intrusive as possible to their work
  - Make them aware of the benefits of research, since later some of them may be recruited as junior researchers

### Motivate professionals
- Most probably they will participate in the experiments as part of their job or as part of a training course
  - E.g. a project being used as a pilot for the introduction of a new development tool
- It is important that their participation in the experiment
  - Is as non-intrusive as possible to their normal work
  - The results of the experiment offer some sort of perceived added value to the organization they work for

Suggested reading:

DATA COLLECTION

Data collection

- Regarding automation, data collection may be:
  - **manual**
    - e.g. subjects fill questionnaires or observers record observations by hand
  - **semi automated**
    - e.g. filling electronic forms, that then perform some kind of aggregation and cross validation
  - **fully automated**
    - e.g. some variables may be extracted with tools (configuration management tools, trouble-ticket systems, etc)

- The collection protocol should be kept constant across subjects and sampling period
  - Fully automated collection is perhaps the only way to get this
Data collection

- Experimenters should record
  - the schedule and effort used in the experiment by participants, to compare with planned values
  - any problems detected on the experiment guidelines, so that it can be improved in further replications of the experiment
  - subject mortality

DATA VALIDATION
Data validation

- This activity ensures that experimental data has been collected correctly

- Problems with data collection can result from:
  - erroneous performance of collection tools
  - misinterpretation of data collection forms by the experiment participants
  - deviations from the planned experimental protocol

Data validation

- Data quality is essential for inference purposes
  - Garbage in -> garbage out!

- This validation activity may involve not only the researchers conducting the experiment, but also the participants
  - The latter can help clarifying data that is found likely to be incomplete, or incorrect.
PROBLEM REPORTING

Reporting problems

- Recording should be performed as problems arise, since postponing may soon lead to oblivion.

- Uncovered or insufficiently covered details or deviations from the original plan should be recorded, such as:
  - inability to use a given tool effectively
  - schedule and effort slippages
  - subjects mortality (next slide)

- Identifying those problems, and how they were mitigated:
  - facilitates experiments replication
  - helps identifying potential validity threats in the results
Subjects mortality

- Information regarding the subjects that are removed from the experiment should be recorded, such as:
  - the cause of removal
  - the potential impact on experiment results

- Patterns of subjects mortality may help uncovering factors which are important to the experiment but are not addressed by the adopted design
  - These factors should be considered, when analyzing the threats to the experiment validity

---

Data Analysis

Fernando Brito e Abreu (fba@di.fct.unl.pt)
Universidade Nova de Lisboa (http://www.unl.pt)
QUASAR Research Group (http://ctp.di.fct.unl.pt/QUASAR)
Overview of the experimental process

Experiment steps

- **Requirements definition**
  - Which are the goals of the experiment?

- **Design planning**
  - Formalize the goals in research hypotheses
  - Define who, when and how the experiment will be conducted

- **Experiment execution**
  - Prepare and collect data in a controlled way

- **Analysis and Interpretation**
  - Check measurements and analyze data to test hypotheses

- **Results Packaging**
  - Interpret and document results to communicate to colleagues
Data analysis taxonomy

- Number of independent variables:
  - One-factorial (aka unifactorial) analysis
    - one independent variables
    - The number of treatments is the number of levels of that factor
  - Multifactorial analysis
    - Several independent variables (factors) are used
    - The number of treatments is the product of the number of levels of each factor (plus 1, if a control group is used)

- Number of dependent variables:
  - Univariate analysis – one dependent variables
  - Multivariate analysis – several dependent variables

Data analysis methods

- Correlation studies
- Proportion testing
- Inference tests for categorical and continuous data
  - Parametric testing
  - Non-parametric testing
- Regression analysis
  - Linear regression modeling
  - Nonlinear regression modeling
  - Logistic regression analysis
- Multivariate data analysis
  - Factor analysis
  - Cluster analysis
  - Discriminant analysis
Measurement scale types

The amount of information that can be provided by a variable is determined by its measurement scale type

- Certain operations (transformations or tests) are not valid for all types

The most common types of measurement scale are:

- Qualitative variables:
  - Nominal scale
  - Ordinal (aka Ranking) scale

- Quantitative variables
  - Absolute scale
  - Interval scale
  - Ratio scale

Nominal scale

- Entities are classified in a *unordered* discrete set of categories

- Nominal variables allow for only qualitative classification
  - That is, they can be measured only in terms of whether the individual items belong to some distinctively different categories, but we cannot quantify or even rank order those categories

- Examples
  - Programming language in which a program is written
  - Model author
  - Defect cause (origin)
Ordinal scale

- Entities are classified in a **ordered** discrete set of categories
  - Ordinal variables allow us to rank order the items we measure in terms of which has less and which has more of the quality represented by the variable, but still they do not allow us to say "how much more."

- Arithmetic operations are **not** supported for this scale
  - But relational operators are (>, >=, =, !=, <=, <)

- Examples
  - Fault impact on operation (Very low, Low, ... Very large)
  - Defect removal difficulty (Simple, Normal, Difficult)
  - Execution time categories (Batch, JIT, Interactive, Critical)
  - Scale type (nominal, ordinal, interval)
    - For example, we can say that nominal measurement provides less information than ordinal measurement, but we cannot say "how much less" or how this difference compares to the difference between ordinal and interval scales.

Interval scale

- Values are discrete and ordered like in the ordinal type, but there is a distance among distinct categories
  - The distance between consecutive categories is constant
  - The scale has no absolute zero

- We can **rank the items** measured, as well as quantify and compare the **sizes of differences** between them
  - Example: Instants of system faults occurred in operation
  - A fault occurred on the 5th April is 2 days greater than another on the 3rd April, and the time elapsed between these two instants is half as much as the one between faults occurred in the 10th and 14th April
Absolute scale

- Similar to the interval type but with an absolute zero
  - Corresponds to simple counting of entities

- All arithmetic operations are supported

Examples
- Number of project participants
- Number of defects found
- Program size (KLOC, number of classes)
  - Example: a program with 800 KLOC is twice as large as one with 400 KLOC

Ratio scale

- They may correspond to:
  - continuous properties (e.g. most characteristics of nature)
  - ratios among entities (usually absolute)

- All arithmetic operations are supported
  - Most statistical data analysis procedures do not distinguish between the interval, absolute and ratio properties of the measurement scales

Examples
- MTBF, MTTR
- Defect density
- Reviewer productivity
Descriptive statistics

- These are usually taken from the available sample
  - They are important to the extent to which they can infer information about the population

- Descriptive statistics types:
  - Measures of Central Tendency
  - Measures of Dispersion

Measures of Central Tendency

- Mode
  - The most frequent value of a set of values

- Median
  - Middle value of an ordered set of values (or the average of the middle two in an even-numbered set)

- Mean (aka Arithmetic Mean)
  - An average of n numbers computed by adding some function of the numbers and dividing by some function of n
  - This is probably the most often used descriptive statistic
  - The mean is a particularly informative measure of the “central tendency” of the variable if it is reported along with its confidence intervals

- Geometric mean
  - This statistic is useful when the measurement scale is not linear; it is computed as:
  - \( G = (x_1 \times x_2 \times \ldots \times x_n)^{1/n} \), where \( n \) is the sample size.
Confidence Interval for the Mean

- Gives us a range of values around the mean where we expect the "true" (population) mean is located, with a given level of certainty or significance ($p$)

- **Example**
  - Sample mean = 23; $p = 0.05$ confidence interval = [19, 27]
  - **Conclusion**: there is a 95% probability that the population mean is greater than 19 and lower than 27

- The calculation of confidence intervals is based on the assumption that the variable is normally distributed in the population
  - The estimate may not be valid if this assumption is not met, unless the sample size is large, say $n=100$ or more

- The width of the confidence interval depends on
  - **Value of $p$** – smaller $p$-level leads to wider confidence intervals thereby increasing the "certainty" of the estimate, and vice versa
  - **Sample size** – The larger the sample size, the more reliable its mean (smaller interval)
  - **Variation of data values** – The larger the variation, the less reliable the mean (larger interval)

Measures of Dispersion

- **Quartile**
  - **Definition**: any of three points that divide an ordered distribution into four parts, each containing one quarter of the total cases
  - The 2nd quartile is de **median**

- **Decile**
  - **Definition**: any of nine points that divide an ordered distribution into equal intervals, where each interval contains one-tenth of the total cases
  - The 5th decile is de **median**

- **Percentile**
  - **Definition**: any of the 99 numbered points that divide an ordered distribution into 100 parts, each of which contains one-hundredth of the total cases
  - The **median** is the 50 percentile
### Measures of Dispersion (around the mean)

- **Interval of variation (aka range)**
  - Difference between maximum and minimum values of the variable within the considered cases

- **Variance range**
  - Is the sum of squared deviations from the mean divided by one less than the number of cases
  - Is measured in units that are the square of those of the variable itself (the square of the standard deviation)

- **Standard deviation**
  - Is the square root of the variance
  - Is measured in the same units as the variable itself
  - A measure of dispersion around the mean. In a normal distribution, 68% of the cases fall within one standard deviation of the mean and 95% of the cases fall within two standard deviations
    - Example, if the mean age is 45, with a standard deviation of 10, 95% of the cases would be between 25 and 65 in a normal distribution

### Measures of Dispersion

- **Skewness**
  - Measures the deviation of the distribution from symmetry

- **Kurtosis**
  - Measures the "peakedness" of the distribution relative to the normal distribution
### Statistics for each scale

<table>
<thead>
<tr>
<th>Scale Type</th>
<th>Measure of Central Tendency</th>
<th>Measure of Dispersion</th>
<th>Measure of Association (correl. coef.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominal</td>
<td>Mode</td>
<td>Frequency (as in histograms)</td>
<td>Pearson $\chi^2$ (Chi-Square)</td>
</tr>
<tr>
<td>Ordinal</td>
<td>Median, Percentile</td>
<td>Interval of variation</td>
<td>Spearman rho, Kendall tau</td>
</tr>
<tr>
<td>Interval &amp; Absolute</td>
<td>Mean</td>
<td>Standard deviation, variance range</td>
<td>Pearson r</td>
</tr>
<tr>
<td>Ratio</td>
<td>Geometric mean</td>
<td>Coefficient of variation</td>
<td></td>
</tr>
</tbody>
</table>

### Statistical distribution

- The distribution shape tells us the frequency of values from different ranges of the variable

- **distribution function:**
  - A distribution function (also known as the probability distribution function) of a continuous random variable $X$ is a mathematical relation that gives for each number $x$, the probability that the value of $X$ is less than or equal to $x$.
  - For example, a distribution function of height gives, for each possible value of height, the probability that the height is less than or equal to that value.
  - For discrete random variables, the distribution function is often given as the probability associated with each possible discrete value of the random variable; for instance, the distribution function for a fair coin is that the probability of heads is 0.5 and the probability of tails is 0.5.
Uniform distribution

- This distribution takes two parameters, a and b (a<=b)
  - A uniform variate takes values between these two parameters with equal probability
- The density function is flat between a and b
- **Example**: dice or coin toss result

Poisson distribution

Is defined as:

- \( f(x) = \frac{x \cdot e^{-}}{x!} \)
  
  for \( x = 0, 1, 2, ..., \ 0 < \)

- Where:
  - \( \lambda \) is the expected value of \( x \) (the mean)
  - \( e \) is the base of the natural logarithm, sometimes called Euler's \( e \) (2.71...)
- Has an asymmetric curve
- It is often used to represent arrival events
  - e.g. in models encompassing network simulation or queues in general
Normal (Gaussian) distribution

- Normal or Gaussian
  - Continuous symmetric bell curve
  - The distribution is uniquely determined by its mean and standard variance

- A characteristic property of the Normal distribution is that 68% of all of its observations fall within a range of ±1 standard deviation from the mean, and a range of ±2 standard deviations includes 95% of the cases
  - In other words, in a Normal distribution, observations that have a standardized value of less than -2 or more than +2 have a relative frequency of 5% or less
  - Standardized value means that a value is expressed in terms of its difference from the mean, divided by the standard deviation

Normal (Gaussian) distribution

- It has been noted empirically that many measurement variables have distributions that are at least approximately normal
  - Even when a distribution is non-normal, the distribution of the mean of many independent observations from the same distribution becomes arbitrarily close to a normal distribution as the number of observations grows large

Central limit theorem

- As the sample size (of samples used to create the sampling distribution of the mean) increases, the shape of the sampling distribution becomes normal
  - Note: for n=30, the shape of that distribution is "almost" perfectly normal
Why is the "Normal distribution" important?

- … because in most cases, it approximates well the function that represents the relationship between "magnitude" and "significance" of relations between two variables, depending on the sample size.

- The distribution of many test statistics is normal or follows some form that can be derived from the normal distribution
  - Many frequently used statistical tests make the assumption that the data come from a normal distribution.

Distribution adherence

- The distribution type conditions the kind of statistical tests we can apply.

- Therefore we want to know if a variable follows (adheres to) a given statistical distribution
  - Often we are interested in how well the distribution can be approximated by the normal distribution.

- We can take several, increasingly powerful, approaches:
  - Use descriptive statistics (seen previously)
  - Use plots
  - Use distribution adherence tests
Assessing Normal distribution

- Plots
  - Q-Q (Quartile-Quar..tile) plots
  - P-P Plots

Testing distribution adherence

Most common normality tests

- **Kolmogorov-Smirnov** one-sample test
- **Lilliefors** test (correction upon the previous)
- **Shapiro-Wilks' W** test
- **Royston test** (correction upon the previous)

These tests are also known as **goodness-of-fit** ones since they test whether the observations could reasonably have come from the specified distribution.
Testing distribution adherence
Kolmogorov-Smirnov one-sample test

- The Kolmogorov-Smirnov one-sample test for normality is based on the maximum difference between the sample cumulative distribution and the hypothesized cumulative distribution.

- $H_0$: $X \sim N(\mu; \sigma)$
- $H_1$: $\neg X \sim N(\mu; \sigma)$

Notes:
- For many software programs, the probability values that are reported are based on those tabulated by Massey (1951); those probability values are valid when the mean and standard deviation of the normal distribution are known a-priori and not estimated from the data
- This test can be used to verify goodness of fit for other distributions (e.g. uniform, Poisson, exponential)

Interpretation:
- If the Z statistic is significant, then the hypothesis that the respective distribution is normal ($H_0$) should be rejected
  - “Significant” means that the statistical significance $p$ of the result is not inferior to the test significance $\alpha$ (required level)

Example:
- Consider the test significance $\alpha = 0.05$
- Probability of Type I error = $0.05 \times 100\% = 5\%$
  - (probability of rejecting $H_0$, the null hypothesis, when it is true)
- If $p \leq \alpha$ (significant Z statistic):
  - Reject $H_0$ and accept $H_1$ (sample cannot come from a Normal population)
- If $p > \alpha$ (not significant Z statistic):
  - Accept $H_0$ and therefore reject $H_1$ (sample may come from a Normal population)
Example:

<table>
<thead>
<tr>
<th>N</th>
<th>Functional Size</th>
<th>Mean</th>
<th>Std. Deviation</th>
<th>Norm. Parameters a,b</th>
<th>Absolute Differences</th>
<th>Positive Differences</th>
<th>Negative Differences</th>
<th>Kolmogorov-Smirnov Z</th>
<th>Asymp. Sig. (2-tailed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3310</td>
<td></td>
<td>444.17</td>
<td>926.623</td>
<td>0.317</td>
<td>0.262</td>
<td>-0.317</td>
<td>18.238</td>
<td>0.000</td>
<td></td>
</tr>
</tbody>
</table>

- Even for a test significance $\alpha = 0.01$ (99% confidence interval), since $p=0.000 \leq \alpha$ (significant Z statistic):
  - We reject $H_0$ and accept $H_1$ (neither Size nor Effort can come from a Normal population)

SPSS:

- Analyze
- Nonparametric Tests
- 1-Sample K-S…

Statistical significance (p-value) of a result

- The p-value represents the probability of error that is involved in accepting our observed result as valid, that is, as "representative" of the population
  - A p-value of 5% (i.e., 1/20) indicates that there is a 5% probability that the relation between the variables found in our sample is a "fluke" (stroke of luck)

- For adherence tests, the p-value is the probability that the observed difference between the sample cumulative distribution and the hypothesized cumulative distribution occurred by pure chance ("luck of the draw")
  - In other words, that in the population from which the sample was drawn, no such difference exists
Common p-values
(conventions in many research areas)

- Borderline statistically significant
  - \( \alpha = 5\% (1/20) \)

- Statistically significant
  - \( \alpha = 1\% (1/100) \)

- Highly statistically significant
  - \( \alpha = 0.5\% (1/200) \) or even \( 0.1\% (1/1000) \)

Hypothesis testing

- Suppose that a CIO is interested in showing that in his software-house the projects have an average defect density (ADD) below \( 5[KLOC]^{-1} \). This question, in statistical terms: “Is ADD < 5?”

- **STEP 1:** State as a "statistical null hypothesis" (hypothesis \( H_0 \)) something that is the logical opposite of what you believe.
  - \( H_0: ADD > 5 \)

- **STEP 2:** Collect data (build a sample)

- **STEP 3:** Using statistical theory, show from the data that it is likely \( H_0 \) is false, and should be rejected.
  - By rejecting \( H_0 \), you support what you actually believe.

- This kind of situation, which is typical in many fields of research, is called "Reject-Support testing," (RS testing) because rejecting the null hypothesis supports the experimenter's theory.
Hypothesis testing

- The null hypothesis is either true or false
  - The statistical decision should be set up so that no "ties" occur
- The null hypothesis is either rejected or not rejected

Two kinds of errors
- $\alpha$ – Type I error rate, must be kept at or below .05
- $\beta$ – Type II error rate, must be kept low as well (the conventions are much more rigid with respect to $\alpha$ than with respect to $\beta$)

- The "Statistical power," $(1-\beta)$, must be kept high
  - Ideally, power should be at least .80 to detect a reasonable departure from the null hypothesis

<table>
<thead>
<tr>
<th>Decision</th>
<th>State of the World</th>
</tr>
</thead>
<tbody>
<tr>
<td>H0</td>
<td>$\alpha$ – Type I Error (($1 - \alpha$))</td>
</tr>
<tr>
<td>H1</td>
<td>$\beta$ – Type II Error (1 - $\beta$)</td>
</tr>
</tbody>
</table>

Hypothesis testing (expanded)

<table>
<thead>
<tr>
<th>STATE OF THE WORLD</th>
<th>DECISION</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H_0$ is True</td>
<td>Accept $H_0$</td>
</tr>
<tr>
<td>$H_1$ is False</td>
<td>Reject $H_1$</td>
</tr>
<tr>
<td>$H_0$ = False</td>
<td>OK</td>
</tr>
<tr>
<td>$H_1$ is True</td>
<td>Type I Error</td>
</tr>
<tr>
<td>$H_0$ = True</td>
<td>OK</td>
</tr>
<tr>
<td>$H_1$ = False</td>
<td>Type II Error</td>
</tr>
</tbody>
</table>
Statistical Tests

- Parametric vs. Non-parametric
  - Assumptions
    - Distribution - normal?
    - Independence
  - Power
    - Parametric require fewer data points
  - Robustness
    - Tests may be applied even if assumptions are not satisfied

Statistical Tests

- Parametric tests
  - Assure stronger validity than the non-parametric counterparts
    - Their statistical power is greater

- Non-parametric tests
  - Weaker validity than the parametric counterparts
    - Their statistical power is smaller
Statistical Tests for Scales

Measurement scale of the variable under consideration

- Nominal
- Ordinal
- Interval
- Ratio

- Non-parametric test
- Normal distribution

- No
- Yes

Non-parametric methods
- Parametric methods

Designs and Tests

<table>
<thead>
<tr>
<th>Design</th>
<th>Parametric</th>
<th>Non-parametric</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 factor, 1 treatment</td>
<td></td>
<td>Chi-square Binomial</td>
</tr>
<tr>
<td>1 factor, 2 treatments, randomized</td>
<td>t-Student (aka t-test), F-test</td>
<td>Mann-Whitney (aka U-test)</td>
</tr>
<tr>
<td>1 factor, 2 treatments, paired</td>
<td>Paired t-test</td>
<td>Wilcoxon, Sign test</td>
</tr>
<tr>
<td>1 factor &gt; 2 treatments, 2 or more</td>
<td>ANOVA</td>
<td>Kruskal-Wallis (aka H-test)</td>
</tr>
<tr>
<td>factors</td>
<td>MANOVA</td>
<td></td>
</tr>
</tbody>
</table>
### Parametric tests (between groups)

<table>
<thead>
<tr>
<th>Name</th>
<th>Factors / Treat.</th>
<th>Outcome scale</th>
<th>Null hypotheses</th>
</tr>
</thead>
<tbody>
<tr>
<td>t-Student (aka t-test) [one sample]</td>
<td>NA</td>
<td>Numeric (absolute, interval or ratio)</td>
<td>The mean of a variable is equal to a specified constant?</td>
</tr>
<tr>
<td>t-Student (aka t-test) [2 independent samples]</td>
<td>1/2</td>
<td>Numeric (absolute, interval or ratio)</td>
<td>The means of a variable on each group (treatment) are the same?</td>
</tr>
<tr>
<td>One-Way ANOVA</td>
<td>1/2+</td>
<td>Numeric (absolute, interval or ratio)</td>
<td>The means of a variable on each group (treatment) are the same?</td>
</tr>
</tbody>
</table>
| Factorial ANOVA                     | 2+/2+            | Numeric (absolute, interval or ratio)      | i) The means of a variable on each group (treatment) are the same?  
 ii) There is no interaction among the factors? |

### Nonparametric tests (between groups)

<table>
<thead>
<tr>
<th>Name</th>
<th>Factors / Treat.</th>
<th>Outcome scale</th>
<th>Null hypotheses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binomial test (test of proportions)</td>
<td>1/2</td>
<td>NA</td>
<td>The expected proportions are the ones being tested?</td>
</tr>
<tr>
<td>Chi-Square (test of proportions)</td>
<td>1/2+</td>
<td>NA</td>
<td>The expected proportions in the groups are similar?</td>
</tr>
<tr>
<td>Mann-Whitney test (aka U-test)</td>
<td>1/2</td>
<td>At least ordinal scale</td>
<td>The two groups have similar central tendency?</td>
</tr>
<tr>
<td>Kruskal-Wallis test (aka H-test)</td>
<td>1/2+</td>
<td>At least ordinal scale</td>
<td>The several groups have a similar localization parameter?</td>
</tr>
</tbody>
</table>
| Nonparametric Factorial ANOVA       | 2+/2+            | At least ordinal scale                     | i) The several groups have a similar localization parameter?  
 ii) There is no interaction among the factors? |
Abstract

- Requirements definition
- Design planning
- Experiment execution
- Data analysis
- Results packaging
VALIDITY THREATS IDENTIFICATION

Validity threats

- Threats to validity are influences that may limit our ability to interpret or draw conclusions from the study's data

- There are several kinds of validity that must be protected from such threats:
  - Construct validity
  - Internal validity
  - External validity
  - Conclusion validity
Construct validity

- This validity occurs when independent and dependent variables accurately model the abstract hypotheses
  - This means we have chosen adequate metrics

- Example:
  - "We propose subjective metrics for measuring each of the dependent variables (maintainability sub-characteristics) based on the judgment of the subjects. As the subjects involved in this experiment have medium experience in UML class diagram design we think their ratings can be considered significant.
  - The independent variables that measure the structural complexity of class diagrams can also be considered constructively valid, because from a system theory point of view, a system is called complex if it is composed of many (different types of elements), with many (different types of) (dynamically changing) relationships between them (Poels and Dedene, 2000a)."

Internal validity

- This validity occurs when changes in the dependent variables can be safely attributed to changes in the treatment (independent variables)
  - This means we have the right data

- Example:
  - “Seeing the results of the experiment we can conclude that empirical evidence of the existing relationship between the independent and the dependent variables exists. We have tackled different aspects that could threaten the internal validity of the study, such as: differences among subjects, knowledge of the universe of discourse among class diagrams, accuracy of subject responses, learning effects, fatigue effects, persistence effects and subject motivation.”
External validity

- This validity occurs when the study's results generalize to settings outside the study
  - This means we have the right respondents/sample

- Example:
  - If students are subjects do results generalize to professional programmers?
  - If the programs used in the experiment are 100 lines or less, do the results generalize to 500,000 line programs?
  - “In general in order to extract a final conclusion that can be generalized, we need to replicate this experiment with a greater number of subjects, including practitioners. After doing replication we will have a cumulative body of knowledge; which will lead us to confirm if the presented metrics could really be used as early quality indicators, and could be used to predict class diagram maintainability.”

External validity

- Refers to the generalization of research findings, either from a sample to a larger population or to settings and populations other than those studied.

- While definitions vary, discussions generally agree that experiments are lower in external validity than other methodological approaches.

- Further, external validity is widely treated as an issue to be addressed through methodological procedures.

- When testing theories, all measures are indirect indicators of theoretical constructs, and no methodological procedures taken alone can produce external validity.
**External validity**

- External validity can be assessed through determining:
  - (1) the extent to which empirical measures accurately reflect theoretical constructs
  - (2) whether the research setting conforms to the scope of the theory under test
  - (3) our confidence that findings will repeat under identical conditions
  - (4) whether findings support the theory being tested
  - (5) the confirmatory status of the theory under test.

- In these ways, external validity is foremost a theoretical issue and can only be addressed by an examination of the interplay between theory and methods.

---

**Conclusion validity**

- Conclusion validity is the degree to which conclusions we reach about relationships in our data are reasonable (credible or believable).

- In many ways, conclusion validity is the most important of the four validity types because it is relevant whenever we are trying to decide if there is a relationship in our observations (and that's one of the most basic aspects of any analysis)
PRODUCING A REPORT

Reasons for Report

- Report results of experiment
  - Dissemination of what you did and what you found out
  - Confirm or reject improves state of software engineering research and practice

- Encourage replication
  - Confirm or alter hypothesis

- Experience base
  - Avoid re-inventing the wheel
  - Analysis
  - Suggest further investigations
Parts of a Report

1. Title page
2. Abstract
3. Introduction
4. Background
5. Experiment description
6. Results
7. Discussion
8. Conclusions
9. References
10. Optional

1. Title page

- Mandatory
  - Descriptive and informative without being overly short or long or full a jargon.
  - Authors and affiliations/addresses.
- Optional
  - Type of report – preliminary report, technical report, manuscript submitted for publication
- Disclaimer
- Date
2. Abstract

- Short description (50-200 words) of the contents
- Should briefly provide a context for the work
  - Describe what was done
  - Summarize what was found
- Important – indexing, help reader

3. Introduction

- Non-technical
- Describes
  - Provides a context for the problem or investigation
  - What others have done
  - What approach the paper took
  - Describes the organization of the rest of the paper
- Must provide sufficient information for read decision
4. Background

- Provides the background necessary to understand
  - What was done
  - What or why a particular approach was taken.
- Historical context information about the problem
  - Origin
  - What others have done
  - What questions still remain to be answered
  - What questions this work focuses on
  - How this experiment is related to or different from previous work.
- Introduce and explain terminology

5. Experiment description

- Overview of experiment purpose, design and procedure
- Enable reader to thoroughly understand what was done and could replicate it
- Describes hypotheses
- Identifies dependent and independent variables
- Describes subjects, materials, and subject task
6. Results

- Describe
  - Data collected
  - How it was analyzed
  - Results of the analysis
- Use hypotheses to organization presentation
- Probably the most common way to organize
- Liberal use of figures, charts, and tables greatly aid the presentation.
- Clear indication of types of analysis done, significant level used and whether statistically significant results were found

7. Discussion

- General description of
  - Significance of the results
  - What results seem to indicate
- Point out extensions or limitations of the study and findings
- Relate the results back to the original hypotheses
- Discuss the threats to validity (construct, internal and external) of the experiment
- Often not a separate section but is included in the Results section
8. Conclusions

- Contents of this section can vary greatly
  - Summary of results
  - Tie results back to original hypotheses
  - Degree to which goals of experiment attained
  - Discuss practical significance of results
  - Mention of ongoing or suggestions for future investigations in the area
  - Suggested extensions of the work

9. References

- List of research cited in paper
- Rule to remember is to cite everything you use from another source in the report
- Each reference should be sufficiently complete so that reader could locate the source
10. Optional

- Acknowledgements
  - People who helped with the investigation and who are not listed among the authors
  - Sources of financial support for the investigation.
  - Placed immediately before the Reference section

- Appendices
  - Contains some of the detailed information that would interrupt the flow of the report and that are not necessary to understand the report
    - Copies of a questionnaire administered
    - Programs used in the experiment

---

Fernando Brito e Abreu (fba@di.fct.unl.pt)
Universidade Nova de Lisboa (http://www.unl.pt)
QUASAR Research Group (http://ctp.di.fct.unl.pt/QUASAR)
Statistical advisors

- **Statistical Advisor** *(StatSoft)*
- **The Decision Tree for Statistics** *(MicrOsiris)*
- **Statistical Test Advisor** *(Analyse-it)*
- **Selecting Statistics** *(Social Research Methods)*

Statistical tools

- **R System** *(open source tool)*
  - The R Project for Statistical Computing
  - R manuals

- **SPSS**
  - Help for SPSS users

- **PROPHET**
  - A Glossary on Statistics
  - StatGuide: Expanded List of Topics

- **SAS**
  - SAS Statistics Tutorials

- **STATISTICA** *(Statsoft)*
  - Electronic Statistics Textbook
Other statistical resources

- NIST/SEMATECH e-Handbook of Statistical Methods
- A New View of Statistics (Will G. Hopkins)
- Interactive Statistical Demonstrations and Tutorials
- Statistical Thinking for Managerial Decisions
- Hossein Arsham list of Probability and Statistics Resources

Bibliography

BOOKS / EDITED BOOKS

- Experimentation in Software Engineering: An Introduction
- Basics of Software Engineering Experimentation
- Empirical Studies in Software Engineering
- Empirical Methods and Studies in Software Engineering
- Lecture Notes on Empirical Software Engineering
- Guide to Advanced Empirical Software Engineering
  - Forrest Shull, Janice Singer, and Dag I.K. Sjøberg (editors), Springer, 2007
Bibliography

- **Experiment conduction and reporting guidelines**
  - [Kitchenham et al., 2002] Evidence software engineering
  - [Jedlitschka and Pfahl, 2005] Evidence in software engineering
  - [Jedlitschka and Ciolkowski, 2005] Guidelines for empirical work in software engineering
  - [Goulão and Brito e Abreu, 2007] Modeling the Experimental Software Engineering Process

- **Models for representing experimental data**
  - [Kitchenham and Hughes, 2001] Software measurement data model
  - [Garcia et al., 2004] Software measurement ontology
  - [Jeusfeld et al., 1998] Design and Analysis of Quality Information for Data Warehouses (inspired by [Basili et al., 1994]'s GQM)

Bibliography

- **Surveys**