Institute for Advanced Computing And Software Development (IACSD)
Akurdi, Pune

Data Visualization

Dr. D.Y. Patil Educational Complex, Sector 29, Behind Akurdi Railway Station,
Nigdi Pradhikaran, Akurdi, Pune - 411044.

Contents

Part 1: Introduction to Business Intelligence and Basics

What is business intelligence?

Transforming data into business insights

Business intelligence (BI) leverages software and services to transform data into actionable insights that inform an organization's business decisions.

Business intelligence definition

Business intelligence (BI) leverages software and services to transform data into actionable insights that inform an organization's strategic and tactical business decisions. BI tools access and analyze data sets and present analytical findings in reports, summaries, dashboards, graphs, charts and maps to provide users with detailed intelligence about the state of the business.

The term business intelligence often also refers to a range of tools that provide quick, easy-to-digest access to insights about an organization's current state, based on available data.

Business intelligence examples

Reporting is a central facet of business intelligence and the dashboard is perhaps the archetypical BI tool. Dashboards are hosted software applications that automatically pull together available data into charts and graphs that give a sense of the immediate state of the company.

Although business intelligence does not tell business users what to do or what will happen if they take a certain course, neither is BI solely about generating reports. Rather, BI offers a way for people to examine data to understand trends and derive insights by streamlining the effort needed to search for, merge and query the data necessary to make sound business decisions.

For example, a company that wants to better manage its supply chain needs BI capabilities to determine where delays are happening and where variabilities exist within the shipping process, says Chris Hagans, vice president of operations for WCI Consulting, a consultancy focused on BI. That company could also use its BI capabilities to discover which products are most commonly delayed or which modes of transportation are most often involved in delays.

The potential use cases for BI extend beyond the typical business performance metrics of improved sales and reduced costs, says Cindi Howson, research vice president at Gartner, an IT research and advisory firm. She points to the Columbus, Ohio, school system and its success using BI tools to examine numerous data points — from attendance rates to student performance — to improve student learning and high school graduate rates.

Please read https://learn.g2.com/business-intelligence-examples

Business intelligence vs. business analytics

One thing you will have noticed from those examples is that they provide insights into the *current* state of the business or organization: where are sales prospects in the pipeline *today?* How many members have we lost or gained *this month?* This gets to the key distinction between business intelligence and another, related term, business *analytics.*

Business intelligence is *descriptive*, telling you what's happening *now* and what happened in the past to get us to that state. **Business analytics,** on the other hand, is an umbrella term for data analysis techniques that are *predictive* — that is, they can tell you what's *going* to happen in the future — and *prescriptive* — that is, they can tell you what you *should* be doing to create better outcomes. (Business analytics are usually thought of as that subset of the larger category of *data analytics* that's specifically focused on business.)

The distinction between the descriptive powers of BI and the predictive or descriptive powers of business analytics goes a bit beyond just the timeframe we're talking about. It also gets to the heart of the question of *who* business intelligence is for.

BI aims to deliver straightforward snapshots of the current state of affairs to business managers. While the predictions and advice derived from business analytics requires data science professionals to analyze and interpret, one of the goals of BI is that it should be easy for relatively non-technical end users to understand, and even to dive into the data and create new reports.

Business intelligence strategy

In the past, IT professionals had been the primary users of BI applications. However, BI tools have evolved to be more intuitive and user-friendly, enabling a large number of users across a variety of organizational domains to tap the tools.

Gartner's Howson differentiates two types of BI. The first is traditional or classic BI, where IT professionals use in-house transactional data to generate reports. The second is modern BI, where business users interact with agile, intuitive systems to analyze data more quickly.

Howson explains that organizations generally opt for classic BI for certain types of reporting, such as regulatory or financial reports, where accuracy is paramount and the questions and data sets used are standard and predicable. Organizations typically use modern BI tools when business users need insight into quickly changing dynamics, such as marketing events, in which being fast is valued over getting the data 100 percent right.

But while solid business intelligence is essential to making strategic business decisions, many organizations struggle to implement effective BI strategies, thanks to poor data practices, tactical mistakes and more.

Self-service business intelligence

The drive to make it possible for just about anyone to get useful information out of business intelligence tools has given rise to self-service business intelligence, a category of BI tools aimed at abstracting away the need for IT intervention in generating reports. Self-service BI tools enable organizations to make the company's internal data reports more readily available to managers and other nontechnical staff.

Among the keys to self-service BI success are business intelligence dashboards and UIs that include pull-down menus and intuitive drill-down points that allow users to find and transform data in easy-to-understand ways. A certain amount of training will no doubt be required, but if the advantages of the tools are obvious enough, employees will be eager to get on board.

Keep in mind, though, that there are pitfalls to self-service BI as well. By steering your business users into becoming *ad hoc* data engineers, you can end up with a chaotic mix of metrics that vary across departments, run into data security problems, and even run up big licensing or SaaS bills if there's no centralized control over tool rollout. So even if you are committing to self-service business intelligence within your organization, you can't just buy an off-the-shelf product, point your staff to the UI, and hope for the best.

Business intelligence software and systems

A variety of different types of tools fall under the business intelligence umbrella. The software selection service SelectHub breaks down some of the most important categories and features:

- Dashboards
- Visualizations
- Reporting
- Data mining
- ETL (extract-transfer-load —tools that import data from one data store into another)
- OLAP (online analytical processing)

Of these tools, SelectHub says the dashboards and visualization are by far the most popular; they offer the quick and easy-to-digest data summaries that are at the heart of BI's value proposition.

There are tons of vendors and offerings in the BI space, and wading through them can get overwhelming. Some of the major players include:

- Tableau, a self-service analytics platform provides data visualization and can integrate with a range of data sources, including Microsoft Azure SQL Data Warehouse and Excel
- Splunk, a "guided analytics platform" capable of providing enterprise-grade business intelligence and data analytics
- Alteryx, which blends analytics from a range of sources to simplify workflows as well as provide a wealth of BI insights
- Qlik, which is grounded in data visualization, BI and analytics, providing an extensive, scalable BI platform
- Domo, a cloud-based platform that offers business intelligence tools tailored to various industries (such as financial services, health care, manufacturing and education) and roles (including CEOs, sales, BI professionals and IT workers)
- Dundas BI, which is mostly used for creating dashboards and scorecards, but can also do standard and ad-hoc reporting
- Google Data Studio, a supercharged version of the familiar Google Analytics offering
- Einstein Analytics, Salesforce.com's attempt to improve BI with AI
- Birst, a cloud-based service in which multilple instances of the BI software share a common data backend.

Business intelligence analyst

Any company that's serious about BI will need to have business intelligence analysts on staff. in general, they aim to use all the features of BI tools to get the data that companies need, the most important being discovering areas of revenue loss and identifying where improvements can be made to save the company money or increase profits.

Even if your company relies on self-service BI tools on a day-to-day basis, business intelligence analysts have an important role to play, as they are necessary for managing and maintaining those tools and their vendors. They also set up and standardize the reports that managers are going to be generating to make sure that results are consistent and meaningful across your organization. And to avoid garbage in/garbage out problems, business intelligence analysts need to make sure the data going into the system is correct and consistent, which often involves getting it out of other data stores and cleaning it up.

The future of business intelligence

Moving ahead, Howson says Gartner sees a third wave of disruption on the horizon, something the research firm calls "augmented analytics," where machine learning is baked into the software and will guide users on their queries into the data.

"It will be BI and analytics, and it will be smart," she says.

The combinations included in these software platforms will make each function more powerful individually and more valuable to the businesspeople using them, Gorman says.

"Someone will look at reports from, for example, last year's sales — that's BI — but they'll also get predictions about next year's sales — that's business analytics — and then add to that a what-if capability: What would happen if we did X instead of Y," Gorman says, explaining that software makers are moving to develop applications that will provide those functions within a single application rather than delivering them via multiple platforms as is now the case.

"Now the system delivers higher-value recommendations. It makes the decision-maker more efficient, more powerful and more accurate," he adds.

And although BI will remain valuable in and of itself, Howson says organizations can't compete if they're not moving beyond only BI and adopting advanced analytics as well.

In fact, Gartner's Magic Quadrant report predicts that by 2020 organizations offering "users access to a curated catalog of internal and external data will realize twice the business value from analytics investments than those that do not."

Howson adds: "There is a need for reporting, but reporting alone is not enough. If you're only doing reporting you're behind already. Unless your reporting is smart and agile, you're behind. You're a laggard."

Requirement Gathering

Requirements engineering provides the appropriate mechanism for understanding what the customer wants, analyzing need, assessing feasibility, negotiating a reasonable solution, specifying the solution unambiguously, validating the specification.

Requirement engineering consists of seven different tasks as follow:

1. Inception

Inception is a task where the requirement engineering asks a set of questions to establish a software process. In this task, it understands the problem and evaluates with the proper solution. It collaborates with the relationship between the customer and the developer. The developer and customer decide the overall scope and the nature of the question.

2. Elicitation

Elicitation means to find the requirements from anybody. The requirements are difficult because the **following problems occur in elicitation**.

Problem of scope: The customer give the unnecessary technical detail rather than clarity of the overall system objective.
Problem of understanding: Poor understanding between the customer and the developer regarding various aspect of the project like capability, limitation of the computing environment.
Problem of volatility: In this problem, the requirements change from time to time and it is difficult while developing the project.

3. Elaboration

In this task, the information taken from user during inception and elaboration and are expanded and refined in elaboration. Its main task is developing pure model of software using functions, feature and constraints of a software.

4. Negotiation

In negotiation task, a software engineer decides the how will the project be achieved with limited business resources. To create rough guesses of development and access the impact of the requirement on the project cost and delivery time.

5. Specification

In this task, the requirement engineer constructs a final work product. The work product is in the form of software requirement specification. In this task, formalize the requirement of the proposed software such as informative, functional and behavioral. The requirement are formalize in both graphical and textual formats.

6. Validation

The work product is built as an output of the requirement engineering and that is accessed for the quality through a validation step. The formal technical reviews from the software engineer, customer and other stakeholders helps for the primary requirements validation mechanism.

7. Requirement management

It is a set of activities that help the project team to identify, control and track the requirements and changes can be made to the requirements at any time of the ongoing project.

These tasks start with the identification and assign a unique identifier to each of the requirement. After finalizing the requirement traceability table is developed.

REQUIREMENT ANALYSIS

Requirements analysis results in the specification of software's operational characteristics, indicates software's interface with other system elements, and establishes constraints that software must meet. Requirements analysis allows you to elaborate on basic requirements established during the inception, elicitation, and negotiation tasks that are part of requirements engineering.
The requirements modeling action results in one or more of the following types of models:

- Scenario-based models of requirements from the point of view of various system "actors"
- Data models that depict the information domain for the problem
- Class-oriented models that represent object-oriented classes (attributes and operations) and the manner in which classes collaborate to achieve system requirements
- Flow-oriented models that represent the functional elements of the system and how they transform data as it moves through the system
- Behavioral models that depict how the software behaves as a consequence of external "events"

These models provide a software designer with information that can be translated to architectural, interface, and component-level designs.

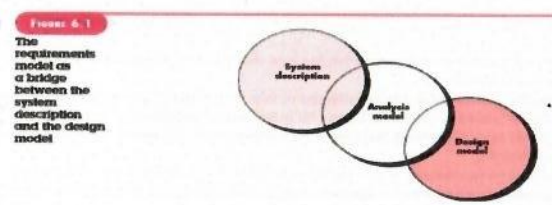


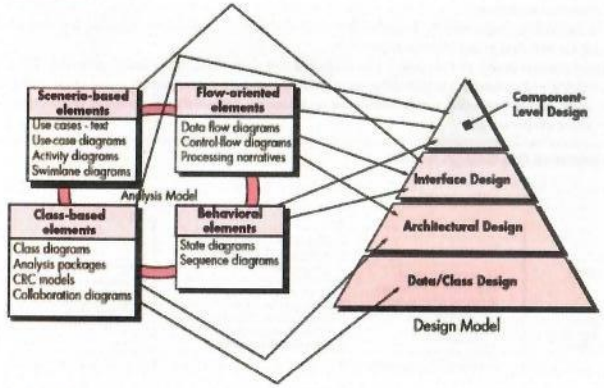
FIGURE 6.1 The requirements model as a bridge between the system description and the design model

Here focus is on scenario-based modeling—a technique that is growing increasingly popular throughout the software engineering community;
scenario-based modeling—a technique that is growing increasingly popular throughout the software engineering community;
data modeling—a more specialized technique that is particularly appropriate when an application must create or manipulate a complex information space;
class modeling—a representation of the object-oriented classes and the resultant collaborations that allow a system to function.

DESIGN WITHIN CONTEXT OF SOFTWARE ENGINEERING

Software design begins once software requirements have been analyzed and modeled, software design is the last software engineering action within the modeling activity and sets the stage for construction (code generation and testing).

FIGURE 8.1 Translating the requirements model into the design model



Each of the elements of the requirements model provides information that is necessary to create the four design models required for a complete specification of design. The flow of information during software design is illustrated in Figure 8.1.
The requirements model, manifested by scenario-based, class-based, flow-oriented, and behavioral elements, feed the design task.
The data/class design transforms class models data structures required to implement the software. The objects and relationships defined in the CRC diagram and the detailed data content depicted by class attribute.
The architectural design defines the relationship between major structural elements of the software, the architectural styles and design patterns that can be used to achieve the requirements defined for the system, and the constraints that affect the way in which architecture can be implemented.
The interface design describes how the software communicates with systems that interoperate with it, and with humans who use it. An interface implies a flow of information and a specific type of behavior.
The component-level design transforms structural elements of the software architecture into a procedural description of software components.

THE DESIGN PROCESS

Software design is an iterative process through which requirements are translated into a "blueprint" for constructing the software.The design is represented a high level of abstraction— a level that can be directly traced to the specific system objective and more detailed data, functional, and behavioral requirements.

Software Quality Guidelines and Attributes

Throughout the design process, the quality of the evolving design is assessed. Three characteristics that serve as a guide for the evaluation of a good design: Each of these characteristics is actually a goal of the design process.

- The design must implement all of the explicit requirements.
- The design must be a readable, understandable guide for those who generate code and for those who test.

- The design should provide a complete picture of the software, addressing the data, function and behavior.

Quality Guidelines

- A design should exhibit an architecture that has been created using recognizable architectural styles or patterns.
- A design should be modular.
- A design should contain distinct representations of data, architecture, interfaces, and component.
- A design should lead to components that exhibit independent functional characteristics.
- A design should lead to interfaces that reduce the complexity of connections between components and with the external environment.
- A design should be represented using a notation that effectively communicates its meaning.
- A design should be derived using a repeatable method that is driven by information obtained during software requirements analysis.

Quality Attributes

Quality attributes represent a target for all software design:

- Functionality - It is assessed by evaluating the feature set and capabilities of the program.
- Usability – It is assessed by considering human factors
- Reliability - It is evaluated by measuring the frequency and severity of failure, the accuracy of output results, the mean-time-to-failure (MTTF), the ability to recover from failure, and the predictability of the program.
- Performance - Itis measured by considering processing speed, response time, resource consumption, throughput, and efficiency.
- Supportability – It combines the ability to extend the program.

THE DESIGN MODEL

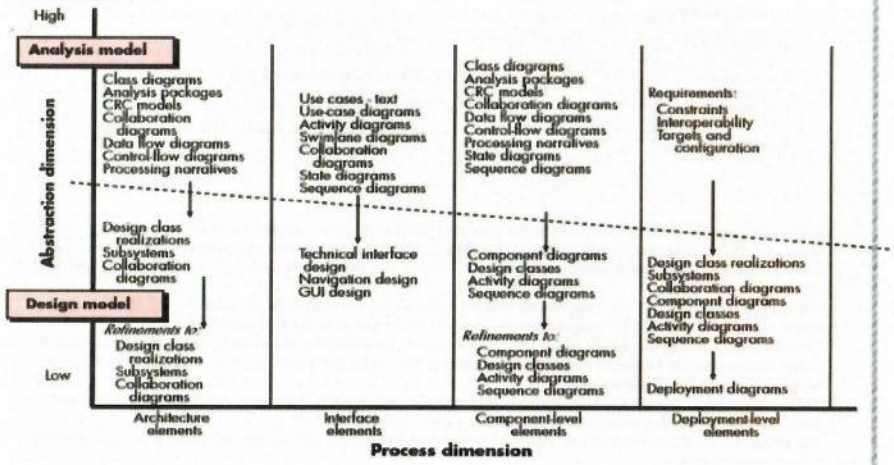
The design model can be viewed in two different dimensions as illustrated in Figure 8.4.

The process dimension indicates the evolution of the design model as design tasks are executed as part of the software process. The abstraction dimension represents the level of detail as each element of the analysis model is transformed into a design equivalent and then refined iteratively.

Referring to Figure 8.4, the dashed line indicates the boundary between the analysis and design models. In some cases, a clear distinction between the analysis and design models is possible.

The elements of the design model use many of the same UML diagrams that were used in the analysis model. The difference is that these diagrams are refined and elaborated as part of design; more implementation-specific detail is provided, and architectural structure and style, components that reside within the architecture, and interfaces between the components and with the outside world are all emphasized.

FIGURE 8.4 Dimensions of the design model



You should note, however, that model elements indicated along the horizontal axis are not always developed in a sequential fashion. The deployment model is usually delayed until the design has been fully developed.

You can apply design patterns at any point during design. These patterns enable you to apply design knowledge to domain-specific problems that have been encountered and solved by others.

SOFTWARE ARCHITECTURE

The software architecture of a program is the structure, which comprise software components, the externally visible properties of those components, and the relationships among them.

The architecture is not the operational software. Rather, it is a representation that enables you to

(1) analyze the effectiveness of the design in meeting its stated requirements,
(2) consider architectural alternatives at a stage when making design changes is still relatively easy, and
(3) reduce the risks associated with the construction of the software.

This definition emphasizes the role of "software components" in any architectural representation. In the context of architectural design, the properties of components are those characteristics that are necessary for an understanding of how the components interact with other components. At the architectural level, internal properties (e.g., details of an algorithm) are not specified. The relationships between components can be as simple as a procedure call from one module to another or as complex as a database access protocol.

Software architecture considers two levels of the design pyramid (Figure 8.1)—data design and architectural design.

Data design enables you to represent the data component of the architecture in conventional systems and class definitions in object-oriented systems.

Architectural design focuses on the representation of the structure of software components, their properties, and interactions.

ELEMENTS OF SOFTWARE QUALITY ASSURANCE

Software Quality Assurance(SQA) encompasses a broad range of concerns(elements) and activities that focus on the management of software quality.

Standards - The IEEE, ISO, and other standards organizations have produced a broad array of software engineering standards and related documents. The job of SQA is to ensure that standards that have been adopted are followed and that all work products conform to them.

Reviews and audits - Technical reviews are a quality control activity performed by software engineers, Their intent is to uncover errors. Audits are a type of review performed by SQA with the intent of ensuring that quality guidelines are being followed for software engineering work.

Testing - Software testing primary goal is to find errors. The job of SQA is to ensure that testing is properly planned and efficiently conducted.

Error/defect collection and analysis - SQA collects and analyzes error and defect data to better understand how errors are introduced and what software engineering activities are best suited to eliminating them.

Change management - change is not properly managed, change can lead to confusion, and confusion almost always leads to poor quality. SQA ensures that adequate change management have been instituted.

Education - Every software organization wants to improve its software engineering practices. A key contributor to improvement is education of software engineers, their managers, and other stakeholders. The SQA organization takes the lead in software process improvement and is a key proponent andsponsor of educational programs.

Security management - SQA ensures that appropriate process and technology are used to achieve software security.

Safety - SQA may be responsible for assessing the impact of software failure and for initiating those steps required to reduce risk

Risk management - SQA organization ensures that risk management activities are properly conducted and that risk-related contingency plans have been established.

Vendor management - The job of the SQA organization is to ensure that high-quality software results by suggesting specific quality practices that the vendor should follow, and incorporating quality mandates as part of any contract with an external vendor.

SOFTWARE TESTING FUNDAMENTALS

The goal of testing is to find errors, and a good test is one that has a high probability of finding an error.

Testability - "Software testability is simply how easily can be tested." The following characteristics lead to testable software.

Operability - "The better it works, the more efficiently it can be tested." If a system is designed and implemented with quality in mind, relatively few bugs will block the execution of tests, allowing testing to progress without fits and starts.

Observability - "What you see is what you test."Inputs provided as part of testing produce distinct outputs. System states and variables are visible or queriable during execution. Incorrect output is easily identified. Internal errors are automatically detected and reported. Source code is accessible.

Controllability - "The better we can control the software, the more the testing can be automated and optimized."

Decomposability - "By controlling the scope of testing, we can more quickly isolate problems and perform smarter retesting." The software system is built from independent modules that can be tested independently.

Simplicity - "The less there is to test, the more quickly we can test it." The program should exhibit functional simplicity, structural simplicity and code simplicity .

Stability - "The fewer the changes, the fewer the disruptions to testing." Changes to the software are infrequent, controlled when they do occur, and do not invalidate existing tests. The software recovers well from failures.

Understandability - "The more information we have, the smarter we will test." The architectural design and the dependencies between internal, external, and shared components are well understood. Technical documentation is instantly accessible, well organized, specific and detailed, and accurate. Changes to the design are communicated to tester.

Test Characteristics

- A good test has a high probability of finding an error. ➢A good test is not redundant.
- A good test should be "best of breed".
- A good test should be neither too simple nor too complex.

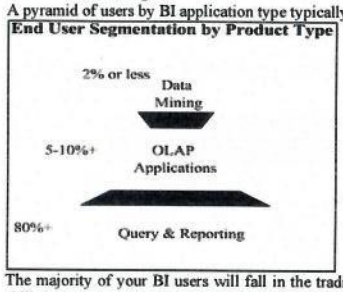
BI User Segmentation

There are user types delineated by application requirements and by skill. If we look at the users in light of their application requirements, we normally see three distinct types:

- Traditional query and reporting

- OLAP
- Data mining

A pyramid of users by BI application type typically looks like the one depicted in Figure 6-1.

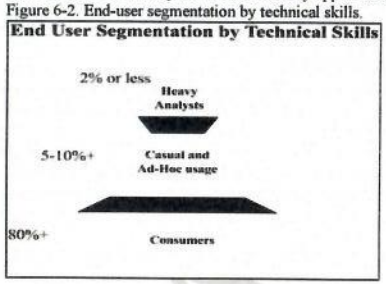


The majority of your BI users will fall in the traditional query and reporting area. This has nothing to do with their level of skill or their position within the enterprise. The most important user of all may receive one small report once a month. This report may be responsible for the majority of territory assignments and critical sales efforts throughout the company. This user may have minimal computer skills and no involvement in the creation or maintenance of the output.

A common error in the BI space is to assign business value to the end user's skills in handling a tool. No one is negating or minimizing the value that a power user may provide, but there is no correlation between the power user's ability to perform technical work and the impact upon the business.

The majority of BI users identified by degree of skill and involvement in the processes will be casual users at best. The profile of the number of users by technical skills may appear as shown in Figure 6.2

Figure 6-2. End-user segmentation by technical skills.



Note that we see a large number of users who have little skill or involvement in the technical end of BI. We also have a large number of users who interoperate with BI applications at the query and reporting level and not in the more sophisticated types such as data mining.

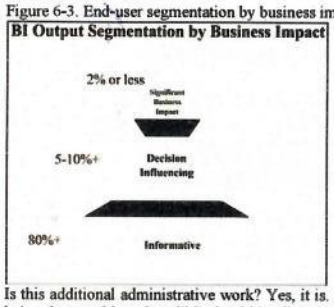
I receive copies of reports and articles regarding many facets of BI, including a recent one from a major industry council discussing end-user segmentation. The research seemed sound, and the conclusions were solid. They suggested that most corporations have a far larger population of passive BI tools users than active, probing ones. I don't think a large, complex survey was required to come to this conclusion.

What information should be collected about our end users? Here is a partial list of the user metadata. In the Appendix, I offer a series of checklists for various aspects of BI solutions. For now, I would suggest the following:

- What is their functional BI category? (consumer, casual and ad hoc, heavy analyst)
- What is their departmental and business information?
- What decision-making level do they provide within the enterprise?
- What BI application and processing requirements do they have?—(query, OLAP, data mining)
- What tools and data map to these requirements?
- What skill level profile (casual, ad hoc, and analyst) are they?
- If a provider of BI output, what key output do they provide?
- What is the business impact of their output if they create BI deliverables?
- What is their backup and plan for skills transfer if they move on or get promoted?
- If a creator, whom do they support?
- How and where will they document their BI activities?

User segmentation by business impact populations is shown in Figure 6.3 Using business impact by user more accurately reflects the overall benefit to the enterprise than cranking out thousands of reports at all levels and assuming the tonnage will produce gold.

Figure 6-3. End-user segmentation by business impact.



Is this additional administrative work? Yes, it is. However, my response to this is "Do you know where your BI analyses are being done and by whom?" It should be disturbing to think that a significant investment has been made in creating new data and in implementing new technologies, but we have no idea what the heck we're doing with them!

By performing a reasonable user survey, we can also determine if much of what we are doing has any business impact. Might it not be helpful to categorize BI output itself? If we are creating 50 new reports with our latest query and reporting tool and they are all simply informational in usage, are we getting a decent return on our investment? We can use our pyramid of users by skill level to construct a view of the BI output that we create, disseminate, and act upon.

Let's tie together the users, their skills, their applications, and their business impact. Would it not make perfect sense to begin with Figure 6-3? Where will we get the greatest business benefit?

If we identify our top ten BI application outputs (or whatever our selections), then we can work backward. Who produces them? How do we track, measure, and back up the applications? What if we discover that we really aren't having much of an impact at all from our efforts?

By assigning some sane measurement of the importance of the output, we also develop a better corporate culture for delineating our users and in making decisions about tools, data, and so on. Does it make any sense to purchase a suite of new query tools for $700,000 because our users tell us they need more information? What do you think?

End-User Attributes

Let's look at the three types of users we defined above: consumers, casual and ad hoc, and heavy analysts. We have segmented them by skill. Sometimes, someone will emerge to fit a category simply by sheer interest in getting a job done. You have probably seen departmental specialists who simply seem to have a knack for working out business problems with tools they were not trained in. Somehow, they just seem to "get it."

In the early days of end-user computing where most of the tools were "green-screen" based, there were always users within an account who could produce results that the others could not. Today, we have tools that allow less sophisticated users to produce results thanks to the set of GUI objects (mini-icons, actions, etc.) that many tools deliver. However, there will still be those analysis requirements that need someone who "gets it" to complete the task. All the icons on the planet cannot make certain tasks easy. Someone will have to produce the result.

We defined these groups earlier in our section about the impact of business intelligence. Now we'll refine these definitions a bit:

- **Consumers**: These are the individuals who simply use the fruits of another's labor. They may be far down on the food chain, or they may be executives. They either cannot, will not, or have no reason to learn much about a tool and how analysis takes place. They should have no bearing on any choices in BI technology, but their needs should be thoroughly understood because many of them may fall into the top layer of the pyramid shown above—the ones who may have significant business impact responsibilities.
- **Casual and Ad Hoc Users**: These are the people who "get it." They either have a skill or seem to be able to learn a tool and apply it to a business problem. They tend to be functional area specialists and may be among the individuals most likely to succeed in the enterprise. They are also the ones you need to listen to outside of the executive levels. They know what needs to be produced to react to business data requests and either figure out some way to do it or find someone that can.
- **Heavy Analysts**: These are often a superset of the casual and ad hoc group or pure IT types. Sometimes, these are the statistical gurus or those with significant skills responsible for producing the extremely complex analytics results. They are often assigned to projects such as the ERP or CRM solutions for the enterprise. They will wallow in complex data issues and in the areas where multiple levels within the enterprise must share information. There isn't a single tool or even a suite of tools in the universe that will make their tasks easy. If one could harness the skills of these users with business analytics targeted to making sweeping changes, you would see the true potential of BI leap out.

A Holistic View of the Users

Do you agree with the assumptions and categories of users described? If not, then jot down your own definitions. What should concern you is any BI initiative in which there is no clear-cut targeted audience. The degree of naiveté surrounding most BI solution decisions is astounding. If you already have three tools installed, for example, do you really think adding a fourth is the key to your success?

May I suggest that the definitions shown as a matrix in Figure 6-4 would be a more accurate and meaningful way of assigning BI-based attributes to your end users? There is no correlation across the rows of the table; these are simply segments within the columns. For example, an OLAP consumer could have only informative impact on the business.

Figure 6-4. End-user profiles.

End User Profiles

User Attributes	BI Application Type	User Business Influence
Consumer	Query & Reporting	Significant Business Impact
Casual & Ad-Hoc	OLAP	Decision Influencing
Heavy Analyst	Data Mining	Informative

If we find that our targeted user set falls very heavily into categories that seem to be information-heavy but influence-light, we may not be utilizing BI applications to our greatest potential.

The matrix shown in Figure 6-5 depicts one user's or group's BI profile. The user attributes for this individual or set of users suggests they are the ones who have some product skills (they fall in the casual and ad hoc category).

Figure 6-5. One end-user's profile.

One End User's Profile ...

User Attributes	BI Application Type	User Business Influence
Consumer	Query & Reporting	Significant Business Impact
Casual & Ad-Hoc	OLAP	Decision Influencing
Heavy Analyst	Data Mining	Informative

Because the application type is OLAP they are probably the ones who attach to an OLAP data server source and "slice-'n-dice," drill up and down, and generally torture the OLAP values to get some results. If they were OLAP application builders, they would certainly not rate as casual. Because they fall into the informative category, is it safe to assume that they produce a bunch of busy-work for others or for themselves?

This is where taking an enterprise-wide view and learning more about the infrastructure of our organization is important. What about the matrix shown in Figure6-6? I worked with one customer with an executive who was relatively savvy about using BI output. The executive was a consumer-level user and had little influence on the creation stages of BI analysis. I am going to take a little license here with the example to make a point about tying all facets of BI usage together.

Figure 6-6. Executive user support profiles.

Executive Sponsor

User Attributes	BI Application Type	User Business Influence
Consumer	Query & Reporting	Significant Business Impact
Casual & Ad-Hoc	OLAP	Decision Influencing
Heavy Analyst	Data Mining	Informative

Support User A

User Attributes	BI Application Type	User Business Influence
Consumer	Query & Reporting	Significant Business Impact
Casual & Ad-Hoc	OLAP	Decision Influencing
Heavy Analyst	Data Mining	Informative

Support User B

User Attributes	BI Application Type	User Business Influence
Consumer	Query & Reporting	Significant Business Impact
Casual & Ad-Hoc	OLAP	Decision Influencing
Heavy Analyst	Data Mining	Informative

The five user types of BI reporting & analytics

In an enterprise environment there are various ways of how users can interact with data using BI. Some users have a need for consuming pre-defined management reports, while other users wish to build their own datasets and create their own analysis. We came up with five types of business users in reporting and analytics: the report consumer, report analyst, self-service data analyst, basic data analyst and advanced data analyst.

Every type of user has its own data needs and requires different types of skills. Understanding this can help you come up with the right approach for guidance and support.

Types of users

The types of BI users

On the far left users solely use data that is prepared and served by others (managed BI). The more we move to the right the more the user does on their own. This means creating own reports, linking the data with own spreadsheet data or even creating their own data models on information that is not (yet) available in a managed BI environment.

Level of experience

User on the left require a basic understanding of BI. The more we move to the right the more data skills and knowledge is required.

Level of control

User on the far left only consume data, the more we move to the right the less organizational control there is on created reports or dashboards.

A brief introduction of each user type and required guidance and support

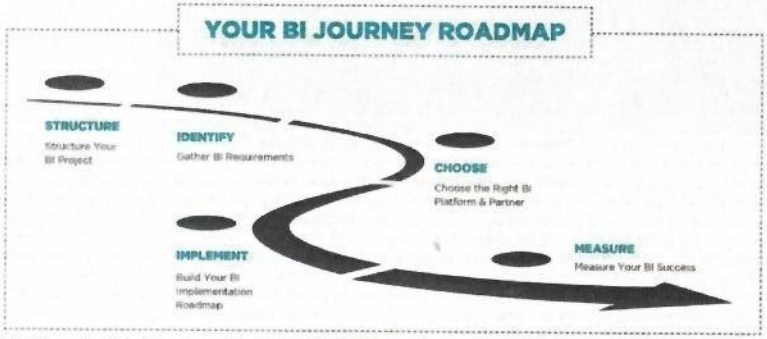
Report Consumer *Consumes report data*	The report consumer uses the Power BI service to consume reports. Has a specific information need. Has no time and or knowledge to filter data or a need to deep dive in details. Needs a brief introduction to the basic of the Power BI service.
Report Analyst *Analyses report data*	The report analyst uses the BI service to get more insights. Has a feeling for data and is able to handle spreadsheet like data volumes. Is able to use report filters, cross filters and drill down on details.
Self Service Data Analyst *Creates and shares new insights on existing data models*	The self-service data analyst is a more experienced business data user. Is able to interpret a pre-build data model. Understands how measures work and has a good understanding of relationships and the impact of handling larger data volumes. Needs an advanced BI walkthrough
Basic Data Analyst *Adds own data to enrich existing data models*	The basic data analyst has experience building his own datasets and reports using a tool like Excel. Misses essential data in existing models and enriches these with his own spreadsheet data. Needs an advanced BI walkthrough
Advanced Data Analyst *Creates new insights while combining existing and new data*	The advanced data analyst is able to build his own data models in BI tool using data from managed and unmanaged data sources. Has a good understanding of data structures, data modelling and data analysis. Knows how to write SQL queries. Depending in the skill level this user could use an advanced BI walkthrough with advanced scripting training and data modelling.

Every type of user has its own needs and requires different types of skills. Understanding this can help you come up with the right approach for guidance and support. The brief introduction above can be a good start.

How to Gather Your Business Intelligence Requirements

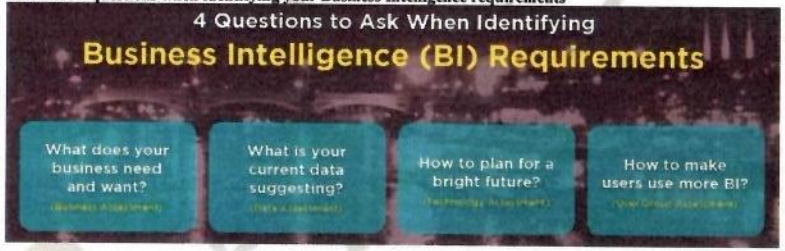
Like any other projects, when you start your Business Intelligence (BI) project, understanding your Business Intelligence requirements is one of the important steps. As the graphic below outlines, there are five major steps on the BI journey roadmap:

1) Structure
2) Identify
3) Choose
4) Implement
5) Measure



This blog post will be focusing on Step 2: Identify. We'll consider some tips and suggestions for how to identify and gather your BI Requirements.

Ask these 4 questions when identifying your Business Intelligence requirements



Each department at your organization has different Business Intelligence requirements and needs. The following four questions can serve as guidelines when you gather your organization's Business Intelligence requirements:

Q1. What does your business need and want? (Business Assessment)
When you answer this question, consider the past, present, and the future. BI needs to address pains experienced in the past. BI needs to deliver on the current needs. More importantly, BI will scale out to cover future wants.

Q2. What is your current data suggesting? (Data Assessment)
Data discovery uncovers the data assets in your current environment. It helps you understand what data you have. It can also suggest data-driven requirements to BI.

Q3. How to plan for a bright future? (Technology Assessment)
Technology capability planning allows architects and other visionaries to envision the future and retrospectively plan for a BI solution that leads you there.

Q4. How to make your end users engage more with BI? (User Group Assessment)
Profiling users into user groups can help visualize their consumption and usage patterns. In addition, it ensures a single BI solution is selected to suit the needs of the different groups of customers.

Results: Asking the questions above will help you find the following results:

- Address business, data, technical, and usage needs.
- Address current and future needs. You will be able to take advantage of the "land and expand" approach.
- Save time by gathering these requirements only once. The requirements package will be used in both the selection phase, as well as in the implementation phase.

3 Methods to gather your Business Intelligence requirements
During the process of gathering your Business Intelligence requirements, you should consider the following three methods.

1. Pain Method (Covers the past)

- **Mentality:** Users are currently experiencing pains related to information needs.
- **How it works:** Vent the pains. Let the end users vent out their information pains, and ask them how they can be relieved. Business Analytics can also convert their pains into requirements.
- **Limitations:** Users are limited by the current situation.

2. Need Method (Covers the present)

- **Mentality:** I need XYZ to perform my job. XYZ is a must-have and I can't live without it.
- **How it works:** Ask the following questions. "What information and information artifacts do you need to perform your functions? What if those pieces are taken away?"
- **Limitations:** Requirements generated from the need mode tend to be very operational in nature. They do not provide out-of-the-box thinking.

- **3. Dream Method** (Nice to have; covers future needs)
- **Mentality:** Let users' imaginations go wild. The sky is the limit.
- **How it works:** Ask users to dream of the ideal future state and how analytics can support their dreams.
- **Limitations:** Not all dreams can be fulfilled. Constraints may prevent the dreams from becoming reality.

Perform data discovery
As you gather your organization's Business Intelligence requirements, you will realize a large amount of data is already available and it has a story to tell. It is helpful to get to know your data and understand the gaps and opportunities. This will help leverage the opportunities in the select and implement BI project and spin off side projects to fix some of your current data issues. Data discovery can be broken down into data inventory and data quality.

Data Quality Assessment

- **What:** Data quality assessment is a test that helps organizations uncover data quality issues in terms of incorrect data, incomplete data, duplicated data, and stale data.
- **Why:** BI success depends on good quality data. Uncovering data quality at an early stage allows you to fix some of the issues before implementing the tool.

Data Inventory

- **What:** Data inventory is an inventory documentation of your key data assets.
- **Why:** Many organizations have a complicated data landscape. Creating a data inventory helps IT to understand what data assets are out there. Knowing the data assets in turn helps the BI project to identify requirements in terms of gaps and opportunities.

Here are our 2 helpful tips for you!
Are you ready to gather your Business Intelligence requirements? Before you get started, here are two last tips from us:

1. Focus on your business needs
Don't focus on what the tool will look like at the beginning. Instead, focus on what your business needs BI to do and the analytics it must be able to support. Don't use these opportunities to create a long list of requirements; instead, use them to understand the motivations and functional uses; requirements will come naturally after that.

2. Ask the right questions
In some situations, your executive team might not have all the answers right away. Help them to focus their thoughts by asking good questions. Since you might not be able to gather all the requirements you need in a single session, give your executive team some time to think through the questions and schedule a follow up meeting to gather the results.

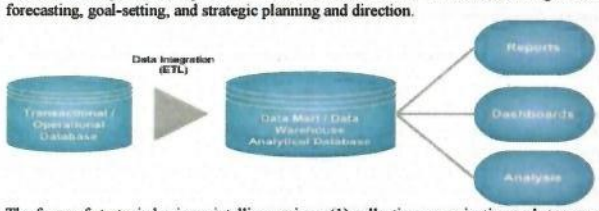
Strategic Approach to BI

Operational business intelligence is often associated with reporting from a transactional or operational data source, and typically is consistent with reporting of data within or during an organizational business process. Further, operational business intelligence can be defined as analytics that is tightly connected or embedded within common business processes with the twin goals of supporting operational decision making and monitoring organizational operations.

In general, operational business intelligence provides time-sensitive, relevant information to operations managers, business professionals, and front-line, customer-facing employees to support daily work processes. Additionally if the data retrieved from the analysis directly supports or helps complete an operational tasks, then the intelligence is operational in nature.
Tangible results of **operational business intelligence** can include:

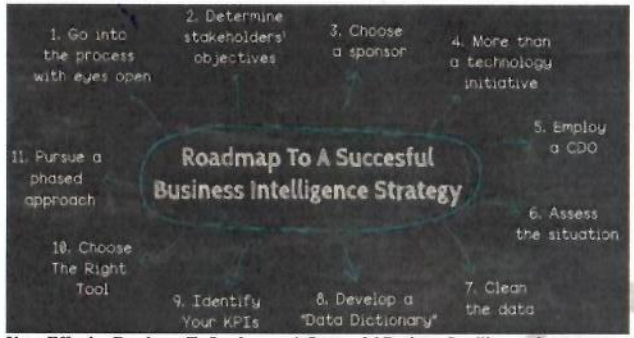
- Invoices
- Meeting Schedules and Badges
- Receipts
- Shipping Documents
- Financial Statements
- Marketing Mailing Lists

Strategic business intelligence is often associated with reporting from an analytical data source, data mart, or data warehouse. Fundamentally, strategic business intelligence improves a business process by analyzing a predetermined set of metrics relevant to that process and provides historical context of data. In addition, strategic intelligence provides the basis for forecasting, goal-setting, and strategic planning and direction.



The focus of strategic business intelligence is on (1) collection, organization and storage of huge amounts of data, (2) optimization of that data for rapid reporting and analysis, (3) and identification of key business drivers through the analysis of historical facts, (4) assistance with answering key business questions.
Questions answered by **strategic business intelligence** can include:

- Who are the most valuable customers?
- Which customers are most likely to buy additional products / services?
- Which products can be bundled together?
- Which territories or regions have the highest project growth?
- What is the optimal price of our products?
- What is the total cost associated with customer acquisition?



Your Effective Roadmap To Implement A Successful Business Intelligence Strategy

Odds are you know your business needs business intelligence (BI). Over the past 5 years, big data and BI became more than just data science buzzwords. Without real time insight into their data, businesses remain reactive, miss strategic growth opportunities, lose their competitive edge, fail to take advantage of cost savings options, don't ensure customer satisfaction... the list goes on. In response to this increasing need for data analytics, business intelligence software have flooded the market. With the benefits being numerous and the costs of not having good BI growing, it is easy to want to quickly adopt a solution. Unfortunately this approach could be disastrous. Investing in BI shouldn't be taken lately. Whether you are starting from scratch, moving past spreadsheets, or looking to migrate to a new platform: you need a business intelligence strategy and roadmap in place. We previously discussed business intelligence for small business. Now we are going to take that a step further with the following 11 steps to a better business intelligence strategy. These steps are imperative for businesses, of all sizes, looking to successfully launch and manage their business intelligence.

What Is A Business Intelligence Strategy?

A business intelligence strategy refers to all the steps you undertake in order to implement business intelligence in your company. It goes all the way to diving in the BI process, defining the stakeholders and main actors, to assessing the situation, defining the goals and finding the performance indicators that will help you measure your efforts to achieve these goals. You define the strategy in terms of vision, organization, processes, architecture and solutions, and then draw a roadmap based on the assessment, the priority and the feasibility.
Business intelligence implementation is not an easy task, as it requires a lot of preparation work beforehand, gathers many different actors and will involve expenses. But the rewards outperform by far its costs, and it is well know that business intelligence ROI is real even if it is sometimes hard to quantify. The costs of *not* implementing it are more damaging especially in the long-term.

Why Do I Need One?

Applying business intelligence is important – but the way you do it matters just as much. This is why having a BI strategy is extremely important: no sailor ever threw their ship in the sea without a map, a telescope and a compass. Think of your strategy just as that: defining the steps on your BI roadmap, following your goals as a compass to stay in the right direction, and investing and using the right tools to get a deep view in your information and understand it.
Everyday a business' information is increasing, and the amount of data dealt with and stored gets way out of proportion to treat it manually – adding on top of that all of the unstructured data that needs to be processed first in order to be understood and later used. It takes time and knowledge to make the best out of such asset, as well as a solid planification. The information a business gathers is filled with precious insights that will help it measure its performance, understand their customers, identify competitive advantages, and much more. A strategy will give your solution a direction, and a goal. Business intelligence without strategy might bring up some insights, it will not lead you where you want to head to.
Having a BI strategy in place before implementing – or just selecting – a system lets you find the perfect match for your needs. It will also facilitate and unclutter the decision-making process, which usually is the goal number one of BI.
The benefits of business intelligence are numerous and undeniable; now you just need to get there and reap them!

11 Steps on Your Business Intelligence Roadmap

1. Go into the process with eyes wide open

When you have the right business intelligence solution, it is easy to identify trends, pitfalls and opportunities early on. But implementing the right solution isn't always easy. Actually, it usually isn't. We are going to be honest here, even the best software needs some initial heavy lifting to maximize its potential. If you go in with the right mindset you will be prepared to address issues like complicated data problems, change management resistance, waning sponsorship, IT reluctance and user adoption challenges. Reminding stakeholders, and yourself, of the pain points that necessitated it will encourage the process forward. It will be worth it.

2. Determine stakeholder objectives

Odds are everyone at your organization could benefit from increased data access and insights. That doesn't mean they are all key stakeholders. Right off the bat you must determine who your key stakeholders are. Then find out what they need: visible and vocal executive sponsorship is a must. Gathering and setting executive team expectations early is paramount. Then move past the executive team. They often don't have the same front-line knowledge that other staff do. Collect and prioritize pain points and key performance indicators (KPIs) across the organization. They might not all make it into the initial rollout, but it is better to start big and roll back.

3. Choose a sponsor

While a business intelligence strategy should include multiple stakeholders, it is imperative to have a sponsor to spearhead the implementation. It may be tempting to place the Chief Information Officer (CIO) or Chief Technical Officer (CTO). This is usually not the best approach. It should be sponsored by an executive who has bottom-line responsibility, a broad picture of the organization's strategy and goals and knows how to translate the company mission into mission focused KPIs.

CFOs and CMOs are good fits. They can govern the implementation with a documented business case and be responsible for changes in scope. Of course, whoever the chosen sponsor is, they will need to be in constant communication with the CIO/CTO. Which brings us to the next step...

4. BI is not just a technology initiative

We are going to repeat ourselves a bit here. Because it is *that* important. To succeed, a deployment must have the support of key business areas, from the get-go. IT should be involved to ensure governance, knowledge transfer, data integrity, and the actual implementation. But every stakeholder and their respective business areas should also be involved throughout the process.

By involving a range of stakeholders you can ensure you cover the three broad classes of business intelligence users: strategic, tactical and operational. These different users types will need customized solutions. Understanding who will use the data and for what purposes can show the type of information needed and its frequency, and help guide your decision making.

The business as a whole must be willing to dedicate the necessary resources: staff, IT resources, costs, etc. BI implementation doesn't just come out of the IT budget. The best business intelligence strategy lays out these resources in the beginning, with additional wiggle room.

5. Employ a Chief Data Officer (CDO)

Big data guru Bernard Marr wrote about *The Rise of Chief Data Officers*. In the article, he pointed to a pretty fascinating trend: "Experian has predicted that the CDO position will become a standard senior board level role by 2020, bringing the conversation around data gathering, management, optimization, and security to the C-level." We love that data is moving permanently into the C-Suite. While, like the CIO, the CDO probably shouldn't be the main sponsor for BI implementation: they (or a similar role) are a great key stakeholder to involve. They will also most likely own the project after the initial implementation is complete.

6. Assess the current situation

As we have already stated: usually a deployment isn't quick or easy. There is a lot of work to do on the front end. One of the biggest sections on a business intelligence roadmap should be assessing the current situation. Now that you have all the right stakeholders at the table the next step is analyzing the current software stack, and the processes and organizational structures surrounding it (or lack thereof). Find out what is working, as you don't want to totally scrap an already essential report or process. Find a way to integrate it into the new strategy, or you will have upset employees. On the flip side, document everything that isn't working. What data analysis questions are you unable to currently answer? Which processes are inefficient or broken?

On top of all this you need to compile which data sources you currently have and how they are being stored. Decide which are necessary to your business intelligence strategy. This should also include creating a plan for data storage services. Are the data sources going to remain disparate? Or does building a data warehouse make sense for your organization?

As with all these steps, both IT and the various business stakeholders should be involved throughout this hefty step.

7. Clean the data

Clean data in, clean analytics out. It's that simple. Cleaning your data may not be quite as simple, but it will ensure the success of your BI. It is crucial to guarantee a solid data quality management, as it will help you maintain the cleanest data possible for better operational activities and decision-making made relying on that data.

Indeed, every year low-quality data is estimated to cost over $9.7 million to American business only, as it impacts the bottom-line, the productivity and ultimately the overall ROI. Of course, one shouldn't become overly obsessed with 100% pure data quality, as perfection doesn't exists, and especially because the purpose is not to create subjective notions of what high quality data is or isn't. The goal is to boost the ROI of your department – and any other – that are relying on this data.

Structured Data	Unstructured Data
Organized information	Diverse structure for information
Quantitative	Qualitative
Requires less storage	Requires more storage
Not flexible	Flexible
Ex. ID, codes, databases, etc	Ex. Text files, SMS, Emails, Videos ,images,etc

8. Develop a "Data Dictionary"

With Agile development, extensive documentation has become a faux-pas. Large data dictionaries can be cumbersome and hard to keep updated. That said, for business intelligence to succeed there needs to be at least a consensus on data definitions and business calculations. The lack of agreement on definitions is a widespread problem in companies today. For example, finance and sales may define "gross margin" differently, leading to their numbers not matching. To nip this in the bud, get all the SMEs at the same table to hammer the definitions out. Then for knowledge transfer choose the repository, best suited for your organization, to host this information.

9. Identify key performance indicators (KPIs)

KPIs are measurable values that show how effectively a company is achieving their business objectives. They sit at the core of a good BI strategy. KPIs indicate areas businesses are on the right track and where improvements are needed. When implementing a BI strategy, it is crucial to consider the company's individual strategy and align KPIs to company's objectives. It may be tempting to create KPIs for everything. This can be a runaway train. It is best to start with the most important KPIs; then create standards and governance with KPI examples in mind. You can always expand on these later.

10. Choose the right tool / partner for your business

At step 10 we finally get to choosing a BI software/partner. Yes, you are this far along in your business intelligence roadmap and you don't even have a tool yet. By preparing properly through steps 1-9 you will be best suited to find the right tool and implement it successfully. During this process you will need to choose and perform a cloud vs on-premise comparison. You also need to make sure to choose a solution that can start small but easily scale as your company and needs grow. Look for flexible solutions that address the needs of all your user.

11. Pursue a phased approach

Rome wasn't built in a day: neither will your BI. A successful BI strategy takes an iterative approach. Think "actionable" and take baby steps. Choose a few KPIs and build a few business dashboards as examples. Gather feedback. Repeat again with new releases every few weeks. Continuously ask yourself what is working and what stakeholders are benefiting.

A good BI roadmap doesn't have an end date. Your organization should be invested in it for the long term. You should be continually measuring and refining your processes, data and reports. Don't let it become stagnate: continually raise the bar.

How To Create A Business Intelligence Strategy

As we have seen all along this article, there's a lot to consider when you want to create and implement a new BI strategy. Let's summarize here all that you need to think beforehand:

- **Assess the situation:** analyze the organizational structure, processes and software stack – or the absence of such. Find out what is working and what isn't, to save you time on already functioning processes. Ask yourself the right business questions and define the strategic goals you want to achieve.
- **Building the BI roadmap:** establishing the steps to follow is like looking on your itinerary before hitting the road. You are aware of everything that will come up and more prepared in front of surprises and problems to handle.
- **Defining your team:** from the head of BI to the business analyst to the developer, you need a solid team with clear roles that will be able to carry out the different tasks on your roadmap.
- **Organizing your BI system:** the data warehouse, the data sources, the software drawing out insights... There's a lot of thinking behind this that shouldn't be neglected, as it will be your central tool to navigate your data and bring out insightful analytics. Once you know where you go and with whom, you shouldn't pick the mount at random!
- **Get ready to hit the road, Jack!** As one would say, you are now ready to rumble! You have all the keys in hands to start the first step of your roadmap and launch your new BI strategy. Good luck in your business intelligence implementation !

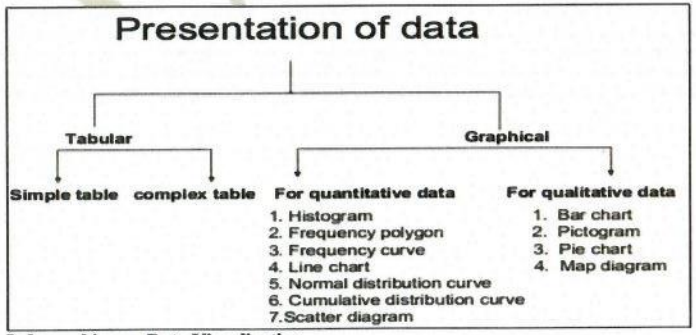
The power a strong BI strategy can bring to your business is compelling – if done correctly. With these 11 steps, your business intelligence roadmap may look a bit daunting; but without them you will end up with an even bigger headache. When done right, BI implementation is the gift that keeps giving. You just need to stick to your business intelligence strategy to get there.

Significance of visual analytics Information Visualization

Visualization is needed in many cases. Depending on the use case, we may need different tools with different features.
• Major use cases
– Presentation: static presentations in meetings – PowerPoint
– Reporting
• Regular/seasonal reports for casual business users - reports
• Real-time or near real-time reporting - dashboard
• Interactive reporting and exploration by power users – interactive reports or dashboard.
• Executive reporting and decision making - dashboard
– Analytical
• Used in the process of analysis, accompanying queries and calculations - Excel
• Advanced visual driven analysis, often used for research – Power BI/Tableau
– Monitoring: real-time operational monitoring (driving, manufacturing) - dashboard
– Public communication/journalism
• Tell a story to the public https://www.vox.com/2018/1/8/16822374/school-segregationgerrymander-map - web
– Demonstration/simulation: interactive demonstration for complex scenarios - http://setosa.io/bus

Why Data Visualization?

• Visualizing is basically a human physiological and psychological capability, and plays an important role in human information behavior and decision making
– Recall or memorize data more effectively
– Enable fast perception based on instinct (see the figure on the right)
– Helps data comprehension and enhance problem solving capabilities (cognition)
– Extract/provoke additional (implicit) perspectives and meanings
– Ease the cognitive load of information processing and exploration
– Help to shape the attention and focus
– Effective communication (story telling)
• More specifically (see examples in the following slides)
– Identify patterns and trends
– Quickly focus on area of interest or area of difference
Identify structures or relationships
– More comprehendible with familiar visual context
– Identify structures and relationships that are hard to express in words



Infographics vs. Data Visualization

• **Major differences ::**
– One time creation and use; mostly created using graphic design

tools rather than using data processing tools
– Information often is more general and can be more qualitative.
– Utilizes more free forms (non standard) of visual diagrams or illustrations (illustrational diagrams); emphasizes creativity and artistically expression to communicate or impress casual viewers
– Often hand-crafted instead of automatically populated from a data source.
– Not for interactive exploration or decision making; intended for more casual use (informational) for general people.
• More readings: infographics vs. data visualization
– https://visage.co/throwdown-data-visualization-vs-infographics/
– http://www.jackhagley.com/What-s-the-difference-between-an-Infographic-and-a-Data-Visualisation

Information design is the practice of presenting information in a way that fosters an efficient and effective understanding of the information.
– These include elements like layout, flow, use of text style, bullets, spacing, etc.
– https://en.wikipedia.org/wiki/Information_design

• **Information visualization** is the study of visual representations of information or data to reinforce human cognition. The data include both numerical and non-numerical data, such as text and geographic information.
– A very close field, and very often used as the synonym for, or even include, data visualization
– Often in the form of illustrations and infographics
– https://en.wikipedia.org/wiki/Information_visualization

• **Infographics** is a specific type of information visualization that are usually a mixture of texts, graphics, and data visual forms (charts, diagrams, tables, maps, etc.) to quickly and vividly communicate complex information (multiple variables or dimensions).
– http://en.wikipedia.org/wiki/Information_graphics
– https://visual.ly/blog/11-infographics-about-infographics/
– Often used in mass communication (e.g. journalism) and marketing
• https://www.business2community.com/digital-marketing/visual-marketing-pictures-worth-60000-words-01126256
• https://www.interaction-design.org/literature/article/information-visualization-who-needs-it
• See more resource about information visualization
– https://www.interaction-design.org/literature/topics/information-visualization

Scientific Visualization

• Physical science visualization
– "Primarily concerned with the visualization of threedimensional phenomena (architectural, meteorological, medical, biological, etc.), where the emphasis is on realistic renderings of volumes, surfaces, illumination sources, and so forth"
– Visualization (simulation) of reality (universe, sun, explosion, atom, climate, etc.)
– https://en.wikipedia.org/wiki/Scientific_visualization
• Mathematical model/algorithm visualization – the visualization created based on math calculations and models
– http://acko.net/blog/how-to-fold-a-julia-fractal/

Business Data/Information Visualization

• Business is a general term to describe activities, events, and operations that make an system running (more like the term field or domain)
– Business includes many activities directly associated with human, like commerce, government, education, sports, charity, entertainment, etc.
– Or events that impact human, such as weather, earthquake, etc.
– Business data or information records various aspects of these activities.
• Main features of business data
– Abstract: data is not directly defining or visualizing (simulating) a real world phenomenon as close as possible, but just representing abstractly an activity, patterns, trends, clusters, outliers, and gaps

– Often quantitative
– Often structured or semi-structured, repeated
– Multidimensional
– Directly comprehendible by average human (in a particular "business")

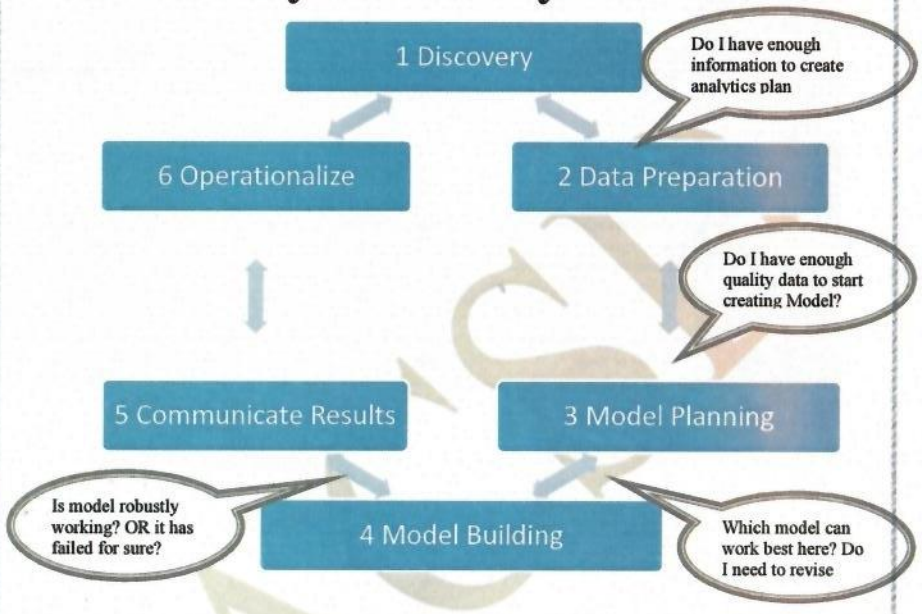
• **Business data visualization features**
– Main purposes are information seeking, analysis, decision support, monitoring, and communication.
– Using simple, standard, and abstract images (symbol/chart/diagram/map)
– Highly reused and commonly accepted visualization forms – following standard practices https://www.ibcs.com/standards
– Utilizes data binding techniques to generate visualizations in an automated way (as part of an analytics software application)
• Where is data visualization used in businesses?
– Part of a BI or analytics process especially in self-service
– Communication of results all kinds of reports (periodical/seasonal or real time) and presentations (e.g. PowerPoint)
– Presentation of results in statistical analysis, data mining or other advanced analytics.
– Visual analytics
– Operational or administrative monitoring

Business Data Visualization
• Periodical reports
– https://myit-2019.itdashboard.gov
• Performance dashboards
– https://www.geckoboard.com/learn/dashboard-examples/
• Visual data exploration and seeking
– https://www.productchart.com/smartphones/
– https://finviz.com/map.ashx
– https://www.census.gov/dataviz/
• Visual analytics
– https://www.google.com/publicdata/directory
• Real time monitoring
– https://www.nytimes.com/interactive/2018/11/06/us/elections/results-dashboard-live.html

What are NOT Good Business Data Visual
These examples are not really considered to be business data visualization – not the focus of this class
• Not meaningful business data
– http://classes.dma.ucla.edu/Spring13/161/projects/students/david/project-5/html/?/image-gallery/
• Mathematical visuals
– https://mathigon.org/world/Fractals
– https://en.wikipedia.org/wiki/Fractal
• Scientific visualization
– https://en.wikipedia.org/wiki/Scientific_visualization
• Reality simulation
– https://weather.com/weather/radar/interactive/l/USGA0028:1:US
• Infographics/data graphics
– https://visual.ly/m/design-portfolio/
– https://informationisbeautiful.net (not all but many are)
– http://www.visualizingdata.com (not all but many are)
– http://courses.ischool.berkeley.edu/i247/s18/ (not all but many are)
• Too much artistic (visual embellishment)
– http://hci.usask.ca/uploads/173-pap0297-bateman.pdf

	Content	Visual Forms/Tools	Purpose
Business data visualization	Quantitative data related to business activities; metrics, key performance indicators (KPIs)	Charts, diagrams dashboards	Data exploration, analysis, decision making
General data visualization	General quantitative data	Charts, diagrams dashboards	Data exploration, analysis, decision making
Information visualization	All kinds of information, quantitative and qualitative	Infographics, illustrational diagrams	Information seeking, artistic illustration, casual communication, story telling
Illustration	Processes, structures concepts, ideas	Diagram, image, graphics	Making the content more vivid and engaging, easier to understand the complexity.
Scientific visualization	Real world object or phenomenon, mathematical functions and formulas	Computer generated graphics, 3D virtual reality	Recreate or simulate the real-world object or phenomenon, or visualize an algorithm effect.
Simulation	Calculated data based on formulas or rules	Animated diagram or virtual reality	Demonstrate the effect of scenarios under certain rules

Part 2
Data Analytics Life Cycle



Phase 1 : Discovery

In this phase,

- The data science team must learn and investigate the problem,
- Develop context and understanding, and Learn about the data sources needed and available for the project.
- In addition, the team formulates initial hypotheses that can later be tested with data.

The team should perform **five main** activities during this step of the discovery phase:

Identify data sources: Make a list of data sources the team may need to test the initial hypotheses outlined in this phase. *Make an inventory of the datasets currently available and those thatcan be purchased or otherwise acquired for the tests the team wants to perform.*

Capture aggregate data sources: This is for previewing the data and providing high-level understanding. *It enables the team to gain a quick overview of the data and performfurther exploration on specific areas.*

Review the raw data: Begin understanding the interdependencies among the data attributes. *Become familiar with the content of the data, its quality, and its limitations.*

Evaluate the data structures and tools needed: The data type and structure dictate which tools the team can use to analyze the data.

Scope the sort of data infrastructure needed for this type of problem: In addition to the tools needed, the data influences the kind of infrastructure that's required, such as disk storage and network capacity.

Unlike many traditional stage-gate processes, in which the team can advance only when specific criteria are met, the Data Analytics Lifecycle is intended to accommodate more ambiguity.

For each phase of the process, it is recommended to pass certain checkpoints as a way of gauging whether the team is ready to move to the next phase of the Data Analytics Lifecycle.

Phase 2: Data Preparation

It has Steps to explore, Preprocess, and condition data prior to modeling and analysis.

It requires the presence of an analytic **sandbox (workspace)**, in which the team can work with data and perform analytics for the duration of the project.

The team needs to execute **Extract, Load, and Transform (ELT) or extract, transform and load (ETL)** to get data into the sandbox.In ETL, users perform processes to **extract data from a datastore, perform data transformations, and load the data back into the datastore.**

The ELT and ETL are sometimes abbreviated as ETLT. Data should be transformed in the ETLT process so the team can work with it and analyze it.

**** Rules for Analytics Sandbox**

When developing the analytic sandbox, collect all kinds of data there, as team members need access to high volumes and varieties of data for a Big Data analytics project.

This can include everything from **summary-level aggregated data, structured data , raw data feeds, and unstructured text data from call logs or web logs,** depending on the kindof analysis the team plans to undertake.

A good rule is to plan for the sandbox to be at least 5– 10 times the size of the original datasets, partly because copies of the data may be created that serve as specific tables or data stores for specific kinds of analysis in the project.

**** Structured vs Unstructured Data**

Structured Data	Unstructured Data
Organized information	Diverse structure for information
Quantitative	Qualitative
Requires less storage	Requires more storage
Not flexible	Flexible
Ex. ID, codes, databases, etc	Ex. Text files, SMS, emails, Videos, images, etc

What is Structured Data?

Structured data usually resides in relational databases (RDBs). Fields store length-delimited data like phone numbers, Social Security numbers, or ZIP codes, and records even contain text strings of variable length like names, making it a simple matter to search.

Data may be human- or machine-generated, as long as the data is created within an RDB structure. This format is eminently searchable, both with human-generated queries and via algorithms using types of data and field names, such as alphabetical or numeric, currency, or date.

Common relational database applications with structured data include airline reservation systems, inventory control, sales transactions, and ATM activity. Structured Query Language (SQL) enables queries on this type of structured data within relational databases.

Some relational databases store or point to unstructured data, such as customer relationship management (CRM) applications. The integration can be awkward at best since memo fields do not lend themselves to traditional database queries. Still, most of the CRM data is structured.

Benefits of Using Structured Data

Easy to Use

Business users who understand what the subject matter of the data is and how it is related to their infrastructure can easily understand how to structure their data. Tools such as Excel or Google Sheets make structured data easy, or more advanced users can lean further into SQL or business intelligence tools.

Convenient Storage

Because structured data is organized, it is commonly stored in data centers for easy access of the data. The data warehouses hold their own space for businesses that choose to use it.

Instant Usability

Structured data is organized, making it easy for a company to find exactly what they are looking for. With this method, a company can begin using the data instantly.

Disadvantages of Structured Data

Limitations on Use

Due to the organization style of structured data, it is more difficult to have flexibility or varied use cases.

Limited Storage

Structured data is stored in specific spaces of data warehouses. While accessing the data is easy, scalability can be difficult. Changes within data warehouses can become hard to manage. Using cloud data centers help with the storage problems.

High Overhead

Data centers or other storage for structured data can become expensive and be part of the structured data ordeal. Again, cloud data centers are recommended, but the storage can still require significant work to keep the data maintained properly.

Structured Data Examples

- ZIP codes
- Phone numbers
- Email addresses
- ATM activity
- Inventory control
- Student fee payment databases
- Airline reservation and ticketing

What is Unstructured Data?

Unstructured data is essentially everything else. Unstructured data has an internal structure but is not structured via predefined data models or schema. It may be textual or non-textual and human- or machine-generated. It may also be stored within a non-relational database like NoSQL.

Typical human-generated unstructured data includes:

Text Files: Word processing, spreadsheets, presentations, emails, and logs.
Email: Message field
Social Media: Data from Facebook, Twitter, and LinkedIn.
Websites: YouTube, Instagram, and photo sharing sites.
Mobile Data: Text messages and locations.
Communications: Chat, IM, phone recordings, and collaboration software.
Media: MP3, digital photos, and audio and video files.
Business Applications: Microsoft Office documents and productivity applications.

Typical machine-generated unstructured data includes:

Satellite Imagery: Weather data, landforms, and military movements.
Scientific Data: Oil and gas exploration, space exploration, seismic imagery, and atmospheric data.
Digital Surveillance: Surveillance photos and video.
Sensor Data: Traffic, weather, and oceanographic sensors.

Benefits of Using Unstructured Data

Limitless Use

Use cases for unstructured data are significantly larger than structured data due to its flexibility. From social media posts to scientific data, unstructured data gives companies the flexibility to use the data how they want.

Greater Insights

When a company has more unstructured data than structured data, there is more data to work with. Unstructured data may be difficult to analyze, but through processing, a company can benefit from the data.

Low Overhead

Because of the ability to store unstructured data at data lakes, a business can save money with how they choose to store the data.

Disadvantages of Unstructured Data

Hard to Analyze

If a company uses unstructured data, it is more difficult to take the raw data and analyze it despite its flexibility.

Data Analytic Tools

Unstructured data cannot be managed by business tools. Its inconsistent nature makes it more difficult than structured data.

Numerous Formats

Unstructured data comes in many different forms, such as medical records, social media posts, and emails. This information may be challenging with analysis.

****Performing ETLT**

As part of the ETLT step, it is advisable to make an inventory of the data and compare the data currently available with datasets the team needs.

Performing this sort of gap analysis provides a framework for understanding which datasets the team can take advantage of today and where the team needs to initiate projects for data collection or access to new datasets currently unavailable.

A component of this sub-phase involves extracting data from the available sources and determining data connections for raw data, **online transaction processing (OLTP) databases, online analytical processing (OLAP) cubes**, or other data feeds.

Data conditioning refers to the process of cleaning data, normalizing datasets, and performing transformations on the data.

****Tools for Data Preparation**

Several tools are commonly used for this phase:

Python Pandas has a complete set of modeling data wrangling and cleaning capabilities for any Tabular data coming from any files like excel, database , XML, or JSON.

Hadoop can perform massively parallel ingest and custom analysis for web traffic analysis, GPS location analytics, and combining of massive unstructured data feeds from multiple sources.

Alpine Miner provides a graphical user interface (GUI) for creating analytic workflows, including data manipulations and a series of analytic events such as staged data-mining techniques (for example, first select the top 100 customers, and then run descriptive statistics and clustering).

Open Refine (formerly called Google Refine) is "a free, open source, powerful tool for working with messy data. A GUI-based tool for performing data transformations, and it's one of the most robust free tools currently available.

Data Wrangler is an interactive tool for data cleaning and transformation. Wrangler was developed at Stanford University and can be used to perform many transformations on a given dataset.

Phase 3 : Model Planning

It determines the methods, techniques, and workflow it intends to follow for the subsequent model building phase.

The team explores the data to learn about the relationships between variables and subsequently selects key variables and the most suitable models.

During this phase that the team refers to the hypotheses developed in Phase 1, when they first became acquainted with the data and understanding the business problems or domain area.

Common Tools for the Model Planning Phase

Python has a complete set of modeling capabilities and provides a good environment for building interpretive models with high-quality code. In addition, it has the ability to interface with databases via an ODBC connection and execute statistical tests.

SQL Analysis services can perform in-database analytics of common data mining functions, involved aggregations, and basic predictive models.

SAS/ ACCESS provide integration between SAS and the analytics sandbox via multiple data connectors such as OBDC, JDBC, and OLE DB. SAS itself is generally used on file extracts, but with SAS/ ACCESS, users can connect to relational databases (such as Oracle or Teradata).

Phase 4 : Model Building

In this phase the data science team needs to develop data sets for training, testing, and production purposes. These data sets enable the data scientist to develop the analytical model and train it ("training data"), while holding aside some of the data ("holdout data" or "test data") for testing the model. The team develops datasets for testing, training, and production purposes.

In addition, in this phase the team builds and executes models based on the work done in the model planning phase. The team also considers whether its existing tools will sufficient for running the models, or if it will need a more robust environment for executing models and workflows (for example, fast hardware and parallel processing, if applicable).

Free or Open Source tools: Python, Rand PL, Octave
Commercial Tools: Matlab, STATISTICA.

Phase 5 : Communicate Results

After executing the model, the team needs to compare the outcomes of the modeling to the criteria established for success and failure.

The team considers how best to articulate the findings and outcomes to the various team members and stakeholders, taking into warning, assumptions, and any limitations of the results account.

The team should identify key findings, quantify the business value, and develop a narrative to summarize and convey findings to stakeholders.

Phase 6 : Operationalize

Here, the team communicates the benefits of the project more broadly and sets up a pilot project to deploy the work in a controlled way before broadening the work to a full enterprise or ecosystem of users.

This approach enables the team to learn about the performance and related constraints of the model in a production environment on a small scale and make adjustments before a full deployment.

The team delivers final reports, briefings, code, and technical documents. In addition, the team may run a pilot project to implement the models in a production environment.

Key outputs of Data Analytics for each of the main stakeholders

Key outputs for each of the main stakeholders of an analytics project and what they usually expect at the conclusion of a project.

Business User typically tries to determine the benefits and implications of the findings to the business.

Project Sponsor typically asks questions related to the business impact of the project, the risks and return on investment (ROI), and the way the project can be evangelized within the organization (and beyond).

Project Manager needs to determine if the project was completed on time and within budget and how well the goals were met.

Business Intelligence Analyst needs to know if the reports and dashboards he manages will be impacted and need to change.

Data Engineer and **Database Administrator (DBA)** typically need to share their code from the analytics project and create a technical document on how to implement it.

Data Scientist needs to share the code and explain the model to her peers, managers, and other stakeholders.

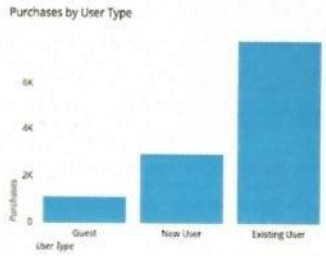
Part 3
Essential Chart Types for Data Visualization

Charts are an essential part of working with data, as they are a way to condense large amounts of data into an easy to understand format. Visualizations of data can bring out insights to someone looking at the data for the first time, as well as convey findings to others who won't see the raw data. There are countless chart types out there, each with different use cases. Often, the most difficult part of creating a data visualization is figuring out which chart type is best for the task at hand.
Your choice of chart type will depend on multiple factors. What are the types of metrics, features, or other variables that you plan on plotting? Who is the audience that you plan on presenting to – is it just an initial exploration for yourself, or are you presenting to a broader audience? What is the kind of conclusion that you want the reader to draw?
In this article, we'll provide an overview of essential chart types that you'll see most frequently offered by visualization tools. With these charts, you will have a broad toolkit to be able to handle your data visualization needs. Guidance on when to select each one based on use case is covered in a follow-up article.

The Foundational Four

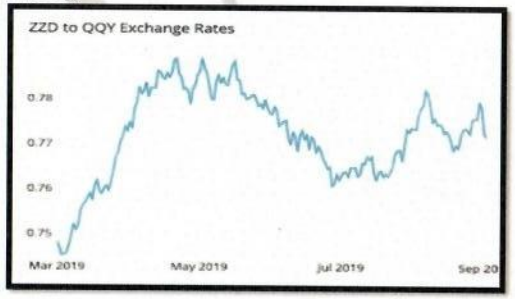
In his book Show Me the Numbers, Stephen Few suggests four major encodings for numeric values, indicating positional value via bars, lines, points, and boxes. So we'll start off with four basic chart types, one for each of these value-encoding means.

Bar chart



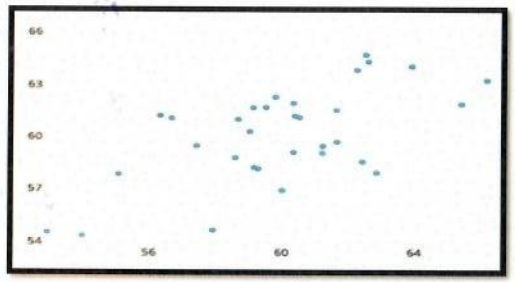
In a bar chart, values are indicated by the length of bars, each of which corresponds with a measured group. Bar charts can be oriented vertically or horizontally; vertical bar charts are sometimes called column charts. Horizontal bar charts are a good option when you have a lot of bars to plot, or the labels on them require additional space to be legible.

Line chart



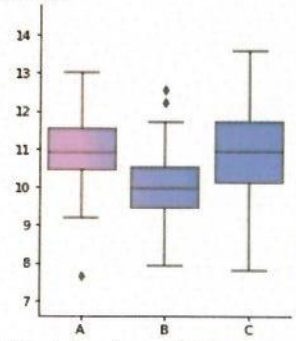
Line charts show changes in value across continuous measurements, such as those made over time. Movement of the line up or down helps bring out positive and negative changes, respectively. It can also expose overall trends, to help the reader make predictions or projections for future outcomes. Multiple line charts can also give rise to other related charts like the sparkline or ridgeline plot.

Scatter plot



A scatter plot displays values on two numeric variables using points positioned on two axes: one for each variable. Scatter plots are a versatile demonstration of the relationship between the plotted variables—whether that correlation is strong or weak, positive or negative, linear or non-linear. Scatter plots are also great for identifying outlier points and possible gaps in the data.

Box plot



A box plot uses boxes and whiskers to summarize the distribution of values within measured groups. The positions of the box and whisker ends show the regions where the majority of the data lies. We most commonly see box plots when we have multiple groups to compare to one another; other charts with more detail are preferred when we have only one group to plot.

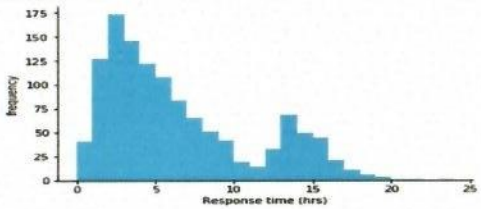
Tables and single values

Before moving on to other chart types, it's worth taking a moment to appreciate the option of just showing the raw numbers. In particular, when you only have one number to show, just displaying the value is a sensible approach to depicting the data. When exact values are of interest in an analysis, you can include them in an accompanying table or through annotations on a graphical visualization.

Common Variations

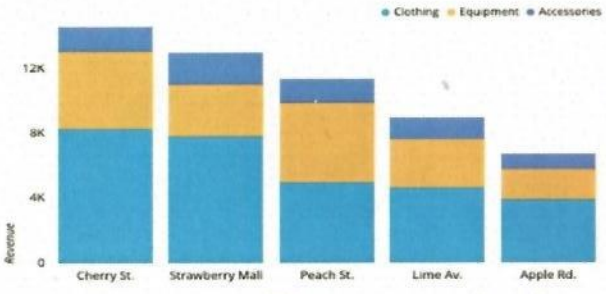
Additional chart types can come about from changing the ways encodings are used, or by including additional encodings. Secondary encodings like area, shape, and color can be useful for adding additional variables to more basic chart types.

Histogram



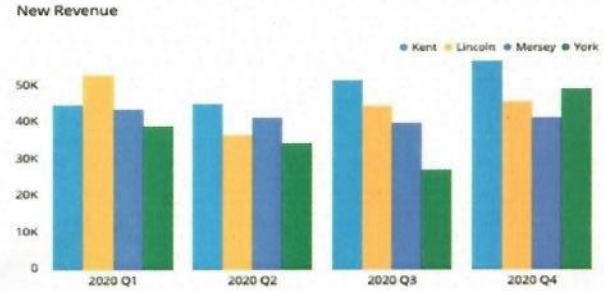
If the groups depicted in a bar chart are actually continuous numeric ranges, we can push the bars together to generate a histogram. Bar lengths in histograms typically correspond to counts of data points, and their patterns demonstrate the distribution of variables in your data. A different chart type like line chart tends to be used when the vertical value is not a frequency count.

Stacked bar chart



One modification of the standard bar chart is to divide each bar into multiple smaller bars based on values of a second grouping variable, called a stacked bar chart. This allows you to not only compare primary group values like in a regular bar chart, but also illustrate a relative breakdown of each group's whole into its constituent parts.

Grouped bar chart

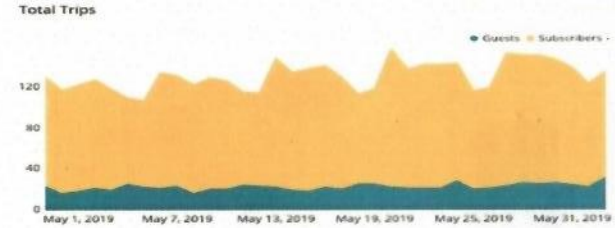


If, on the other hand, the sub-bars were placed side-by-side into clusters instead of kept in their stacks, we would obtain the grouped bar chart. The grouped bar chart does not allow for comparison of primary group totals, but does a much better job of allowing for comparison of the sub-groups.

Dot plot

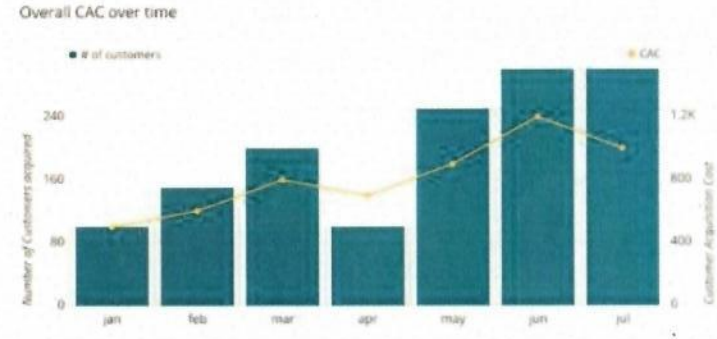
A dot plot is like a bar chart in that it indicates values for different categorical groupings, but encodes values based on a point's position rather than a bar's length. Dot plots are useful when you need to compare across categories, but the zero baseline is not informative or useful. You can also think of a dot plot as like a line plot with the line removed, so that it can be used with variables with unordered categories rather than just continuous or ordered variables.

Area chart



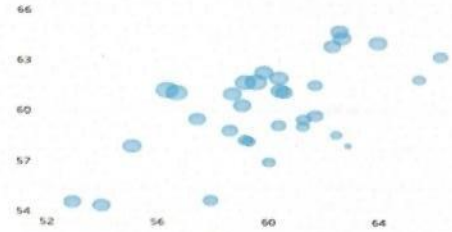
An area chart starts with the same foundation as a line chart – value points connected by line segments – but adds in a concept from the bar chart with shading between the line and a baseline. This chart is most often seen when combined with the concept of stacking, to show how both how a total has changed over time, but also how its components' contributions have changed.

Dual-axis chart



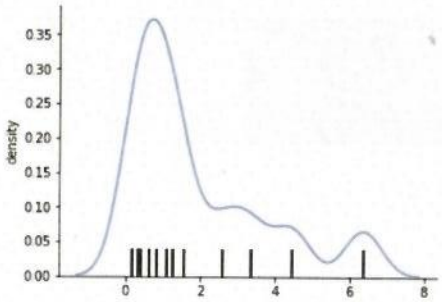
Dual-axis charts overlay two different charts with a shared horizontal axis, but potentially different vertical axis scales (one for each component chart). This can be useful to show a direct comparison between the two sets of vertical values, while also including the context of the horizontal-axis variable. It is common to use different base chart types, like the bar and line combination, to reduce confusion of the different axis scales for each component chart.

Bubble chart



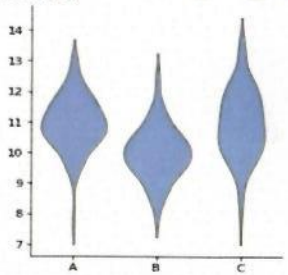
Another way of showing the relationship between three variables is through modification of a scatter plot. When a third variable is categorical, points can use different shapes or colors to indicate group membership. If the data points are ordered in some way, points can also be connected with line segments to show the sequence of values. When the third variable is numeric in nature, that is where the bubble chart comes in. A bubble chart builds on the base scatter plot by having the third variable's value determine the size of each point.

Density curve



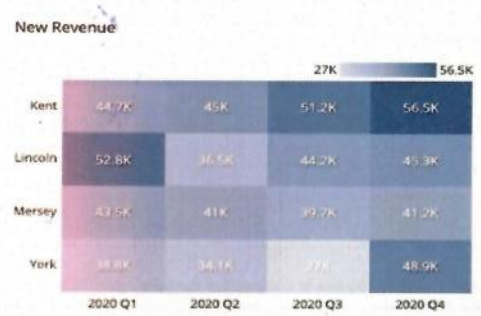
The density curve, or kernel density estimate, is an alternative way of showing distributions of data instead of the histogram. Rather than collecting data points into frequency bins, each data point contributes a small volume of data whose collected whole becomes the density curve. While density curves may imply some data values that do not exist, they can be a good way to smooth out noise in the data to get an understanding of the distribution signal.

Violin plot



An alternative to the box plot's approach to comparing value distributions between groups is the violin plot. In a violin plot, each set of box and whiskers is replaced with a density curve built around a central baseline. This can provide a better comparison of data shapes between groups, though this does lose out on comparisons of precise statistical values. A frequent variation for violin plots is to include box-style markings on top of the violin plot to get the best of both worlds.

Heatmap

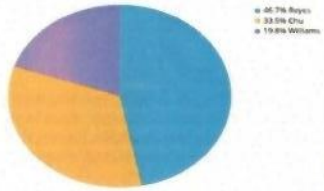


The heatmap presents a grid of values based on two variables of interest. The axis variables can be numeric or categorical; the grid is created by dividing each variable into ranges or levels like a histogram or bar chart. Grid cells are colored based on value, often with darker colors corresponding with higher values. A heatmap can be an interesting alternative to a scatter plot when there are a lot of data points to plot, but the point density makes it difficult to see the true relationship between variables.

Specialist Charts

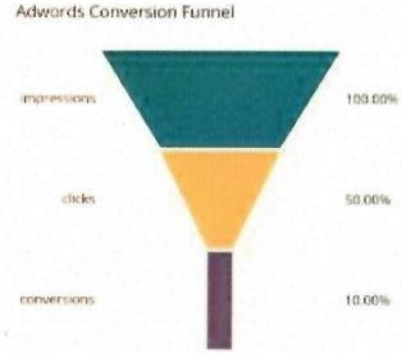
There are plenty of additional charts out there that encode data in other ways for particular use cases. Xenographics includes a collection of some fanciful charts that have been driven by very particular purposes. Still, some of these charts have use cases that are common enough that they can be considered essential to know.

Pie chart



You might be surprised to see pie charts being sequestered here in the 'specialist' section, considering how commonly they are utilized. However, pie charts use an uncommon encoding, depicting values as areas sliced from a circular form. Since a pie chart typically lacks value markings around its perimeter, it is usually difficult to get a good idea of exact slice sizes. However, the pie chart and its cousin the donut plot excel at telling the reader that the part-to-whole comparison should be the main takeaway from the visualization.

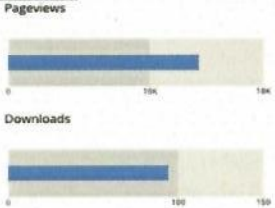
Funnel chart



A funnel chart is often seen in business contexts where visitors or users need to be tracked in a pipeline flow. The chart shows how many users make it to each stage of the tracked process from the width of the funnel at each stage division. The tapering

of the funnel helps to sell the analogy, but can muddle what the true conversion rates are. A bar chart can often fulfill the same purpose as a funnel chart, but with a cleaner representation of data.

Bullet chart



The bullet chart enhances a single bar with additional markings for how to contextualize that bar's value. This usually means a perpendicular line showing a target value, but also background shading to provide additional performance benchmarks. Bullet charts are usually used for multiple metrics, and are more compact to render than other types of more fanciful gauges.

Map-based plots

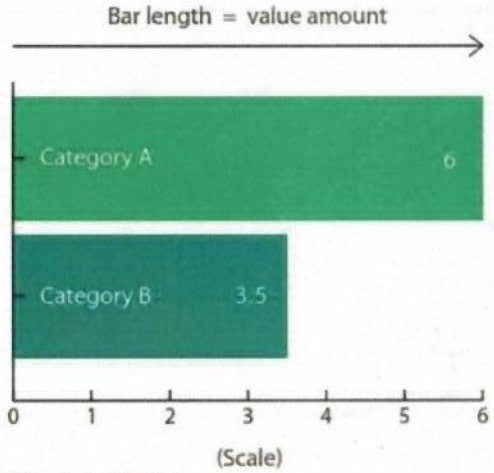
There are a number of families of specialist plots grouped by usage, but we'll close this article out by touching upon one of them: map-based or geospatial plots. When values in a dataset correspond to actual geographic locations, it can be valuable to actually plot them with some kind of map. A common example of this type of map is the choropleth like the one above. This takes a heat map approach to depicting value through the use of color, but instead of values being plotted in a grid, they are filled into regions on a map.

When to use which plot?

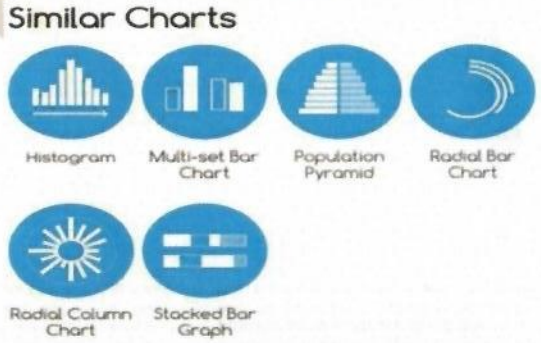
Bar Chart

- Classic Bar chart uses either horizontal or vertical bars (column chart) to show discrete, numerical comparisons across categories
- One axis shows categories being compared
- Other axis shows discrete value scale
- They are **different from Histogram** , because categories being compared are not continuous intervals
- Categories are discrete so Bar Charts answer question "how many?" for each category
- **Limitation:** When too many categories to be compared then labeling to bars is difficult
- **Functions / Usage:** Comparison , Patterns
- **Other names** : Bar Graph, Column Graph

Anatomy of Bar Chart



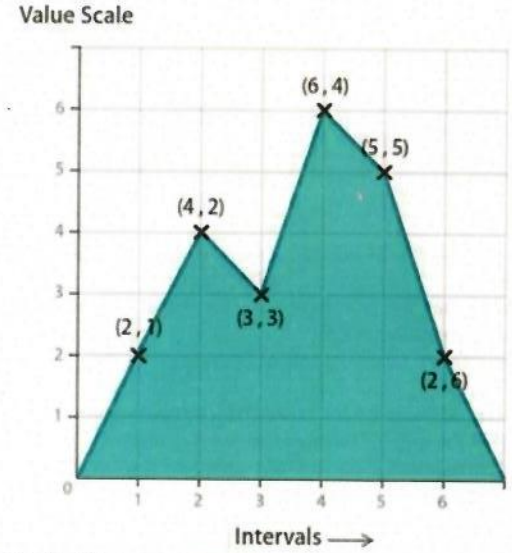
Similar Charts to Bar Chart



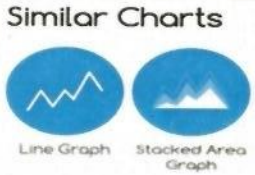
Area Graph

- They are line graphs with area under the line filled
- Like line graphs these are used to explain pattern of quantitative values over an interval or time period
- They highlight trend than specific values
- **Two variations** : Grouped Area Plots , Stacked Area Plots
- **Functions / Usage** : Show pattern over time, data over time, relationships

Anatomy of Area Graph



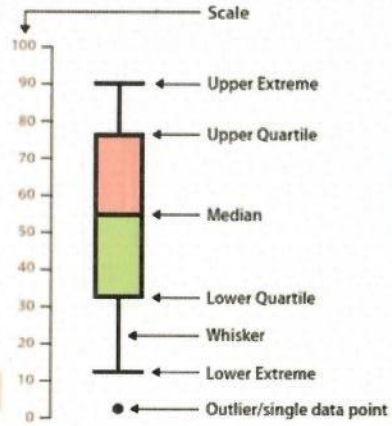
Similar Plots



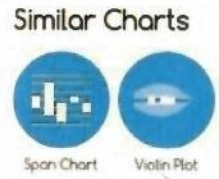
Box and Whisker Plot

- Displays quartiles of numeric data
- Specially used for **detecting presence of outlier**
- Lines extending parallel from the boxes are known as whiskers
- Whisker represents variability outside the upper and lower quartiles
- Outliers are plotted as individual dots
- They take up less space
- Gives values of Q1, Q3 , median and presence of outliers
- **Important Equations:**
- IQR = Q3- Q1
- Lower Quartile = Q1
- Upper Quartile = Q3
- Whisker lower extreme = Q1 – 1.5 * IQR
- Whisker upper extreme = Q3 + 1.5 * IQR
- **Functions / Usage** : Find Distribution , Find Range, Compare distributions, Check presence of outliers ,skewness
- **Other names** : Box Plot

Anatomy of Box Plot



Similar Plots

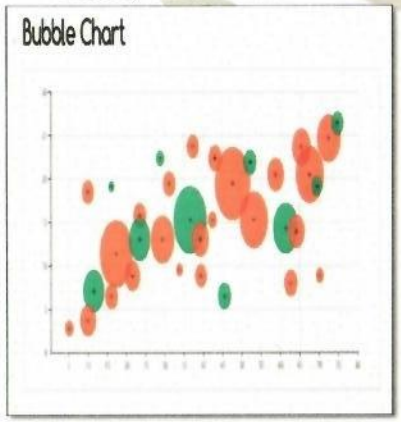


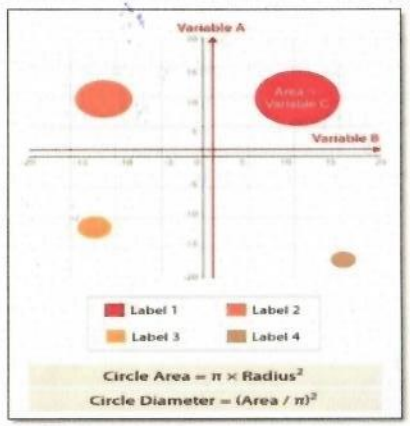
Bubble Chart

- It combination of scatter plot and proportional area chart
- It is multi variable graph
- Like scatter plot each bubble is placed at (x,y) point and then **size of bubble** is given by third variable
- Every bubble is assigned a **category / label**
- **Color** can represent another variable
- **Time** can be shown on axisor show as animation
- **Functions / Usage:**
- Compare categorical circles by using positioning and proportions.
- Can be used to detect patterns or correlations
- Used to check distribution of data
- Used to get proportions
- Used to find relationships
- **Limitations** : Too many bubbles can make it hard to read

Similar Plots

- Anatomy of Bubble Charts

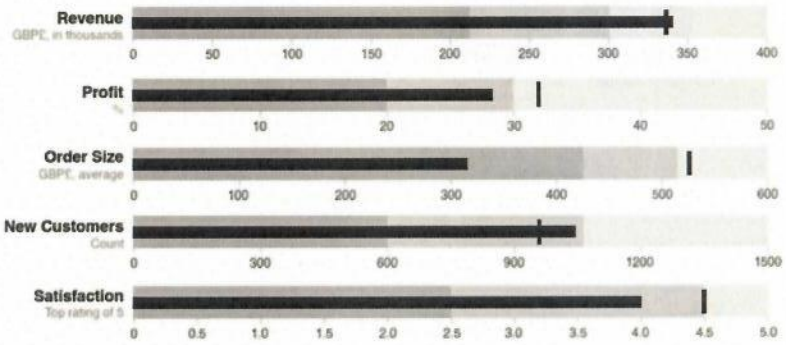


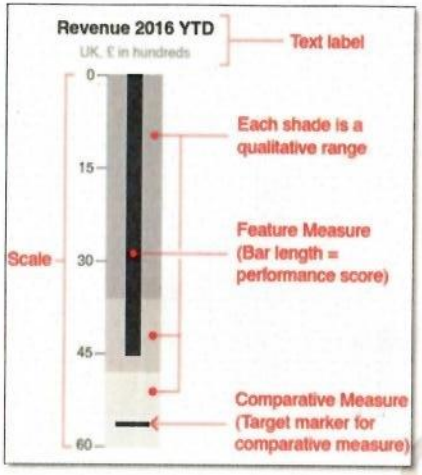


Bullet Graph

- Packed representation of information and context, which appears similar to bar chart
- Developed by Stephen Few, to reduce "chartjunk" and make more information presentable simply
- **Feature measure** : Main data value is encoded by length in main bar in middle of the chart
- **Comparative Measure**: Line maker which runs perpendicular to the orientation of the graph. It denote the target marker. If this line is passed by the feature measure then you have achieved the target.
- **Qualitative Range**: Segmented color bars behind the feature measure are used to display qualitative range scores. Each color shade assigns performance range rating. Ideally keep maximum number of ranges to 5.
- **Functions** : Comparisons and Range

Anatomy of Bullet charts





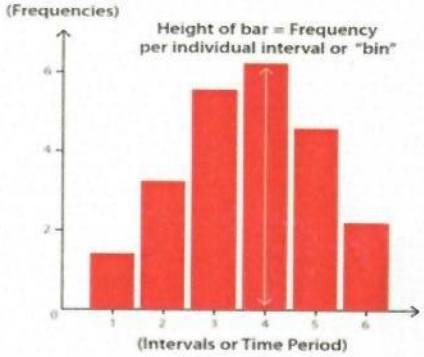
Similar Charts

Bar Chart

Histograms

- Histograms visualize the data over a continuous interval or certain time period
- Each bar represents frequency of each bin / interval
- Total Area of Histogram is number of data points/ samples
- Gives the estimation of dense parts of data
- Can see gaps in data if present
- Can understand min and max values
- **Usage / Functions** : Shows distribution of data, Roughly understand probability distribution
- Can understand range and patterns

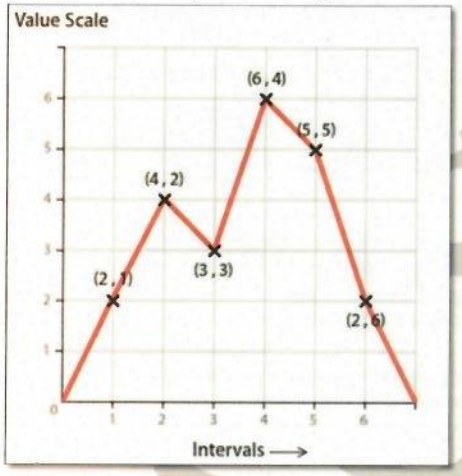
Anatomy of histogram



Line Graphs

- It displays quantitative value over a continuous interval or time span
- Gives an overview / big picture of an interval
- First points are drawn on Cartesian coordinate system and then points are connected by line
- Y-axis : gives quantitative value
- X-axis gives sequence or intervals (categories)
- **Usage / Functions** :Used to show trends in data and compare with other patterns

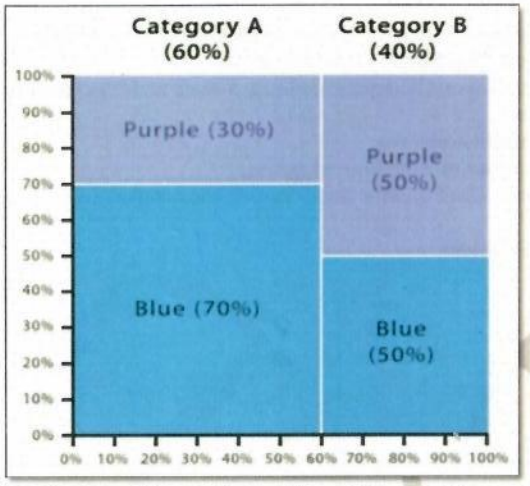
Anatomy of Line Graphs



Marimekko Plots (Mossaic Plots)

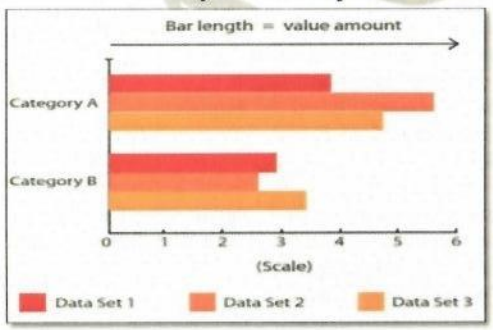
- Shows distribution of categorical variables over pair of variables
- They are like two way 100% stacked bar plots
- Here all bars are of same total height and length
- These bars are divided in segments based on categories
- ONLY HERE : **both axis are SCALE**
- **Usage / Functionality** :
- Gives general overview of data
- find pattern from categorical variables , and compare subcategories
- **Limitation** : When too many categories / segments then difficult to read
- Also difficult to compare various categories

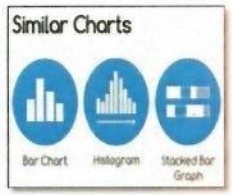
Anatomy of Marimekko Charts (Mosaic Plots)



Multiset Bar Chart (Grouped Bar Chart or Clustered bar Chart)

- Multipledatasets can be plotted together
- Each data series is assigned a color / shade
- Each group of bars are placed separately
- **Usage / Functionality** :
- Used to compare multiple datasets on common parameters
- Sometimes used to compare small histograms
- Can be used to find relation between multiple datasets
- **Limitation**
- Too many datasets comparison becomes tedious





Pictogram Chart

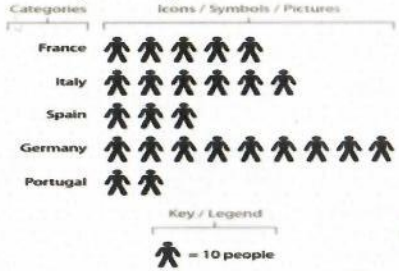
Also known as Pictograph Chart, Pictorial Chart, Pictorial Unit Chart, Picture Graph

- Uses icons to represents small set of discrete data
- Icon represents subject / category of data
- Ex. Population graph icon will be 'person'
- Each icon may represent one or 'n' number of units
- Comparison happens column wise or row wise

- This graph goes beyond language / education / background
- This graph can communicate with masses / general populous

- **Limitation** :
- Cannot represent large datasets
- Partial icon display is confusing to interpret

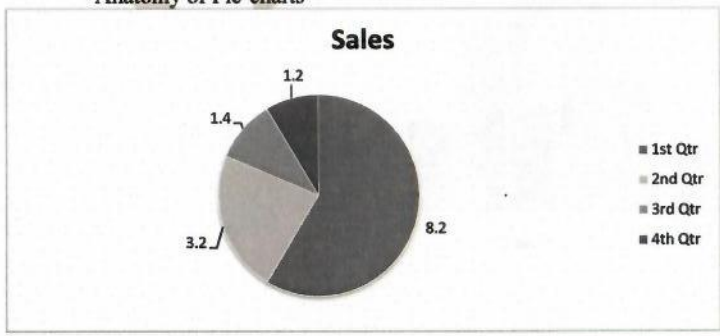
Anatomy of Pictogram Charts



Pie Chart

- Uses a circular representation to represent complete data
- Explains proportions and percentages
- Circle is divided into proportional segments as per the categories
- Full circle represents full data
- Length of the arc represents proportion of that category
- **Usage / Functionality**
- In offices for presentations
- Overall comparison of categories
- To give quick understanding of proportion of data
- Can be used to compare a part with complete data
- **Limitations**
- Can't show many values
- Requires more space
- Can't be use to have accurate comparisons

Anatomy of Pie-charts

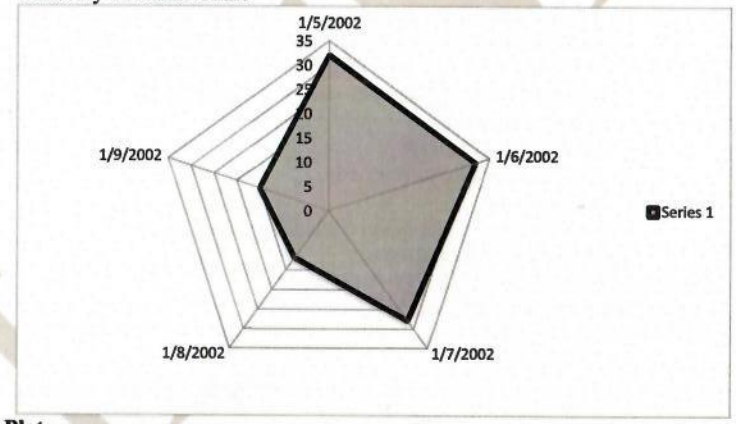


Radar Chart

Also known as Spider chart, Web Chart, Polar Chart, Star Plots

- Represents value of multiple variables
- Each variable is given an axis starting from circle of the center
- All axis are arranged radially with equal distances between each other
- Generally grid lines/circles are persent to be able to compare values of different variables on various axis
- Finally all variable values from same dataset are connected together to form a polygon/star
- **Usage / Functions**
- Compares multiple quantitative variables
- Check if a variable has outlier value
- **Limitations**
- Multiple star charts within a single plot may be confusing
- Can show only limited variables
- Comparing all variables with each other is bit difficult

Anatomy of Radar Chart

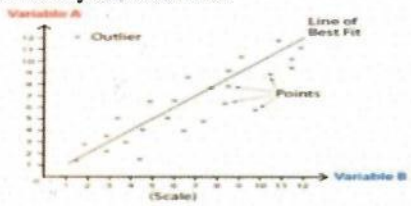


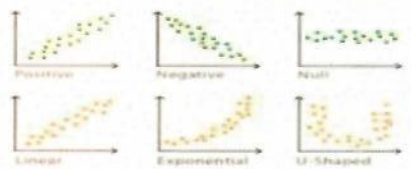
Scatter Plot

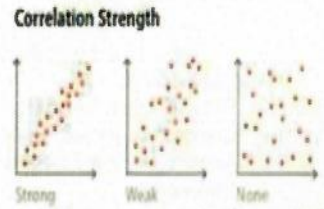
Also known as scatter graph, point graph, X-Y plot, Scatter chart, Scattergram

- Shows each data point as a single dot
- Ideal when Both axis have continuous data
- **Usage / Functions**
- To detect various correlations – linear, exponential, U-shaped, etc
- Strength of correlation is determined by how closely the points are packed
- Line of best fit / trend lines are fitted to show the pattern in the data
- **Limitation**
- Correlation between two variables may not be causal and other variables may be influencing the relation

Anatomy of Scatter Plot



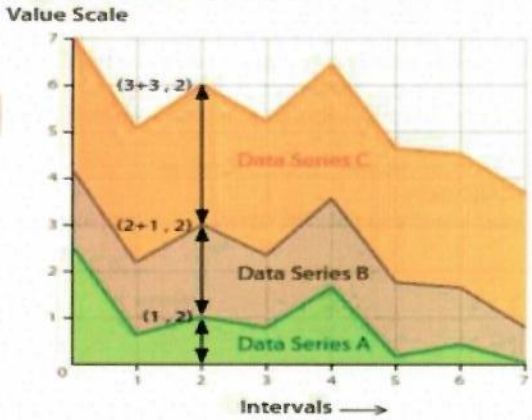




Stacked Area Charts

- Multiple series will have their area charts stacked here
- First series starts at the bottom then next one above it and so on
- Complete plot represents whole data
- **Usage / Functions**
- Used to compare various series
- Patterns in various series
- How data over time is changing
- Used for finding relationships in series
- **Limitations**
- Don't work for negative values

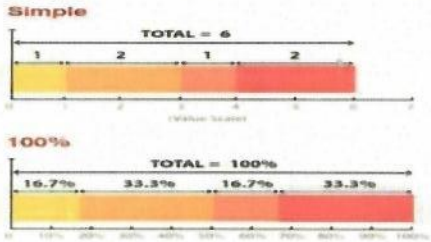
Anatomy of Stacked Area Charts



Stacked Bar Chart

- Here Multiple bars are stacked over each other
- **Usage / Functions**
- How a major category has distribution in miner categories
- **2 Types**
- **Simple Stacked**
  - Here all bars are stacked simply
- **100% Stacked**
  - Here together all stacked bar char is make 100%
  - Each part will show percentage of distribution
- **Limitations**
- Difficult show too many categories
- Compare every segment to the other is difficult here

Anatomy of Stacked bar Charts

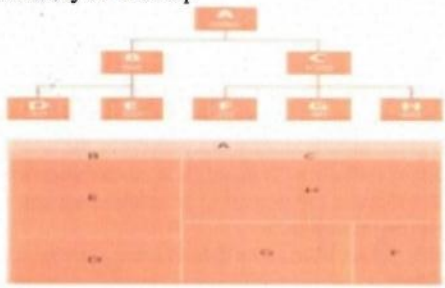


Treemap

Alternative for tree diagrams

- It displays hierarchy as well as quantity using area size
- Each category is given a rectangle and subcategories are put inside that rectangle
- Area size of a category is in proportion to the quantity of that category
- Area size of parent is total of all subcategories
- If there are no quantity values given then all categories and subcategories within them are given same size of blocks
- Commonly "squarified algorithm" is used to create the blocks
- Usage / Functions
- Can be used to compare various categories
- Shows hierarchy
- Requires less size to show the hierarchy than tree diagrams

Anatomy of Treemap

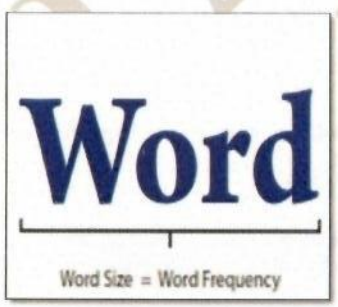


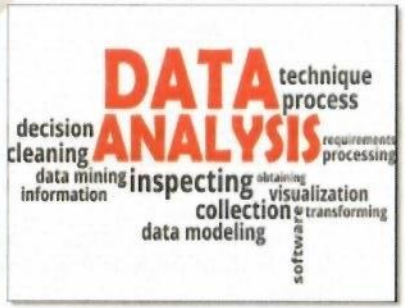
Word Cloud

Also known as TagCloud

- It displays words based on their frequency in the given data
- Word Size = Word frequency
- Then all words are arranged to show a cluster like cloud or rectangle shape
- Words are arranged randomly in rows or columns or angular to fit the shape
- Another use can be to show words based on Meta-data. For example show names of countries based on their population or GDP, etc
- Generally color of words doesn't signify anything, but it can be used to highlight some variable like category of words, etc
- Usage / Functionality
- Used for text analysis
- Used to check frequent words
- **Limitations**
- Long words are emphasized more over short words
- Not accurate for analysis
- More for beautification / aesthetics

Anatomy of Word Cloud



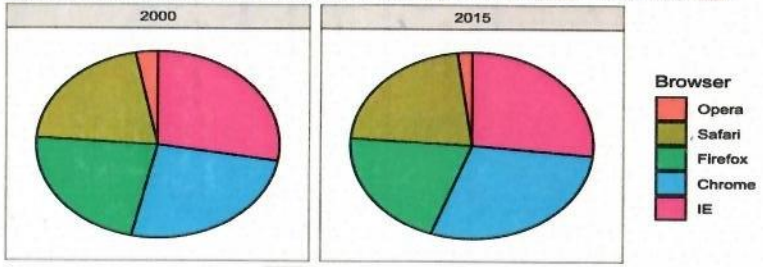


Data Visualization principles (12 principles)

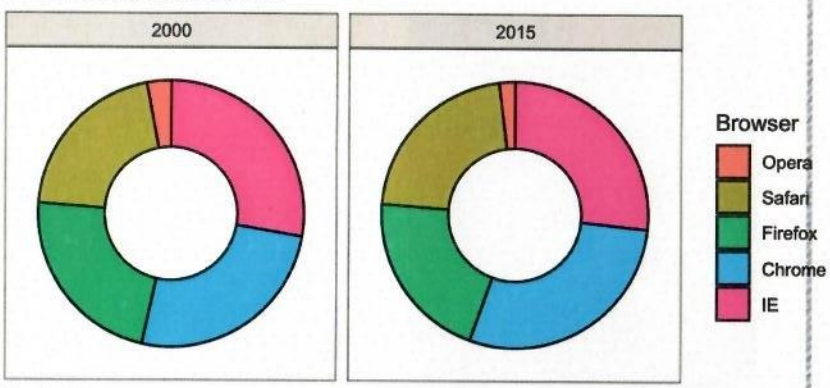
We have already provided some rules to follow as we created plots for our examples. Here, we aim to provide some general principles we can use as a guide for effective data visualization. Much of this section is based on a talk by Karl Broman[20] titled "Creating Effective Figures and Tables"[21] and includes some of the figures which were made with code that Karl makes available on his GitHub repository[22], as well as class notes from Peter Aldhous' Introduction to Data Visualization course[23]. Following Karl's approach, we show some examples of plot styles we should avoid, explain how to improve them, and use these as motivation for a list of principles. We compare and contrast plots that follow these principles to those that don't.
The principles are mostly based on research related to how humans detect patterns and make visual comparisons. The preferred approaches are those that best fit the way our brains process visual information. When deciding on a visualization approach, it is also important to keep our goal in mind. We may be comparing a viewable number of quantities, describing distributions for categories or numeric values, comparing the data from two groups, or describing the relationship between two variables. As a final note, we want to emphasize that for a data scientist it is important to adapt and optimize graphs to the audience. For example, an exploratory plot made for ourselves will be different than a chart intended to communicate a finding to a general audience.

11.1 Encoding data using visual cues

We start by describing some principles for encoding data. There are several approaches at our disposal including position, aligned lengths, angles, area, brightness, and color hue.
To illustrate how some of these strategies compare, let's suppose we want to report the results from two hypothetical polls regarding browser preference taken in 2000 and then 2015. For each year, we are simply comparing five quantities – the five percentages. A widely used graphical representation of percentages, popularized by Microsoft Excel, is the pie chart:



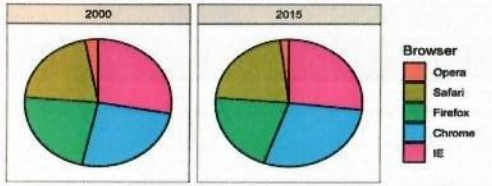
Here we are representing quantities with both areas and angles, since both the angle and area of each pie slice are proportional to the quantity the slice represents. This turns out to be a sub-optimal choice since, as demonstrated by perception studies, humans are not good at precisely quantifying angles and are even worse when area is the only available visual cue. The donut chart is an example of a plot that uses only area:

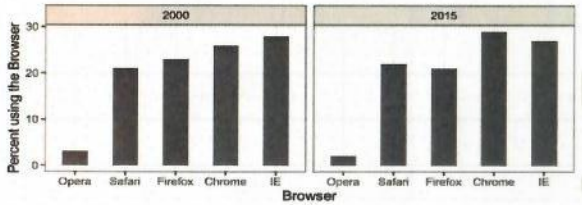


To see how hard it is to quantify angles and area, note that the rankings and all the percentages in the plots above changed from 2000 to 2015. Can you determine the actual percentages and rank the browsers' popularity? Can you see how the percentages changed from 2000 to 2015? It is not easy to tell from the plot. In fact, the pie R function help file states that:
Pie charts are a very bad way of displaying information. The eye is good at judging linear measures and bad at judging relative areas. A bar chart or dot chart is a preferable way of displaying this type of data.
In this case, simply showing the numbers is not only clearer, but would also save on printing costs if printing a paper copy:

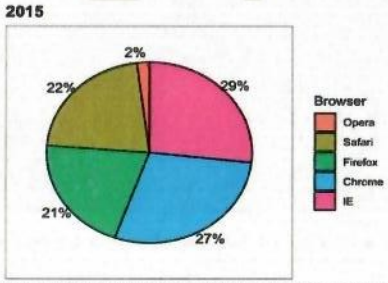
Browser	2000	2015
Opera	3	2
Safari	21	22
Firefox	23	21
Chrome	26	29
IE	28	27

The preferred way to plot these quantities is to use length and position as visual cues, since humans are much better at judging linear measures. The barplot uses this approach by using bars of length proportional to the quantities of interest. By adding horizontal lines at strategically chosen values, in this case at every multiple of 10, we ease the visual burden of quantifying through the position of the top of the bars. Compare and contrast the information we can extract from the two figures.





Notice how much easier it is to see the differences in the barplot. In fact, we can now determine the actual percentages by following a horizontal line to the x-axis.
If for some reason you need to make a pie chart, label each pie slice with its respective percentage so viewers do not have to infer them from the angles or area:



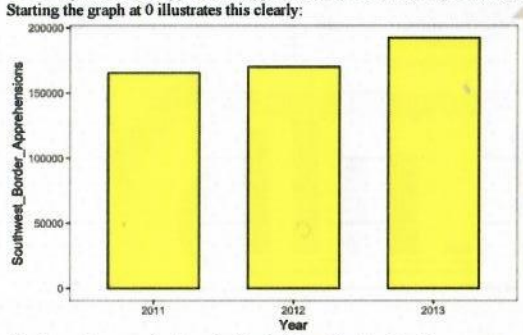
In general, when displaying quantities, position and length are preferred over angles and/or area. Brightness and color are even harder to quantify than angles. But, as we will see later, they are sometimes useful when more than two dimensions must be displayed at once.

11.2 Know when to include 0

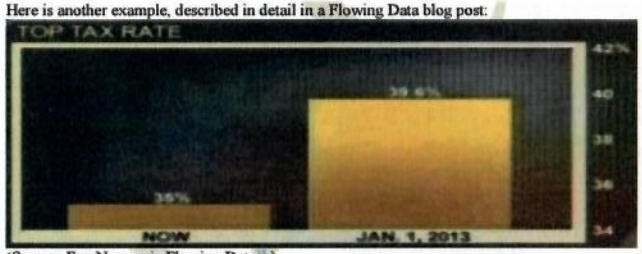
When using barplots, it is misinformative not to start the bars at 0. This is because, by using a barplot, we are implying the length is proportional to the quantities being displayed. By avoiding 0, relatively small differences can be made to look much bigger than they actually are. This approach is often used by politicians or media organizations trying to exaggerate a difference. Below is an illustrative example used by Peter Aldhous in this lecture:
http://paldhous.github.io/ucb/2016/dataviz/week2.html.

(Source: Fox News, via Media Matters[34].)

From the plot above, it appears that apprehensions have almost tripled when, in fact, they have only increased by about 16%. Starting the graph at 0 illustrates this clearly:

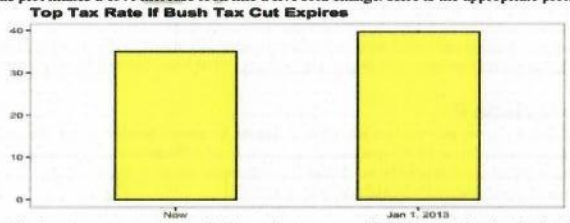


Here is another example, described in detail in a Flowing Data blog post:



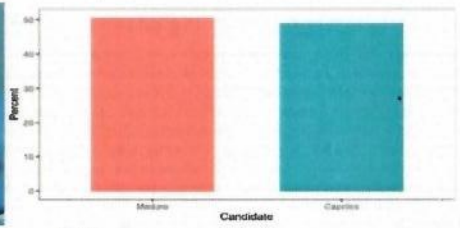
(Source: Fox News, via Flowing Data[35])

This plot makes a 13% increase look like a five fold change. Here is the appropriate plot:



Finally, here is an extreme example that makes a very small difference of under 2% look like a 10-100 fold change in Plot a:

Plot a

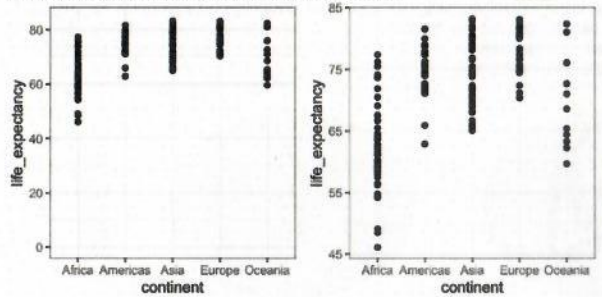


Plat b

(Source: Venezolana de Televisión via Pakistan Today[36] and Diego Mariano.)

Here is the appropriate plot, plot b.

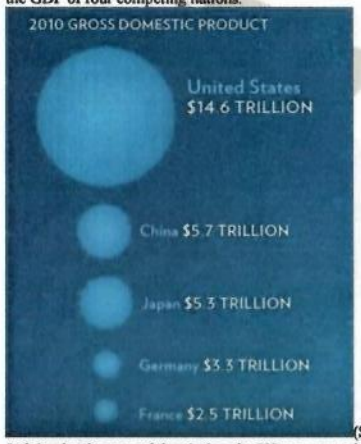
When using position rather than length, it is then not necessary to include 0. This is particularly the case when we want to compare differences between groups relative to the within-group variability. Here is an illustrative example showing country average life expectancy stratified across continents in 2012:



Note that in the plot on the left, which includes 0, the space between 0 and 43 adds no information and makes it harder to compare the between and within group variability.

11.3 Do not distort quantities

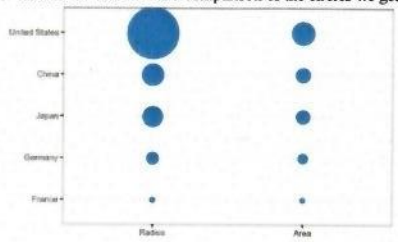
During President Barack Obama's 2011 State of the Union Address, the following chart was used to compare the US GDP to the GDP of four competing nations:



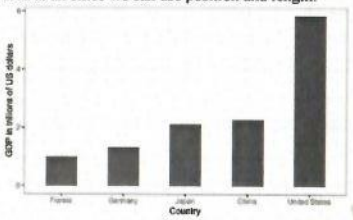
(Source: The 2011 State of the Union Address[37])

Judging by the area of the circles, the US appears to have an economy over five times larger than China's and over 30 times larger than France's. However, if we look at the actual numbers, we see that this is not the case. The actual ratios are 2.6 and 5.8 times bigger than China and France, respectively. The reason for this distortion is that the radius, rather than the area, was made to be proportional to the quantity, which implies that the proportion between the areas is squared: 2.6 turns into 6.5 and

5.8 turns into 34.1. Here is a comparison of the circles we get if we make the value proportional to the radius and to the area:



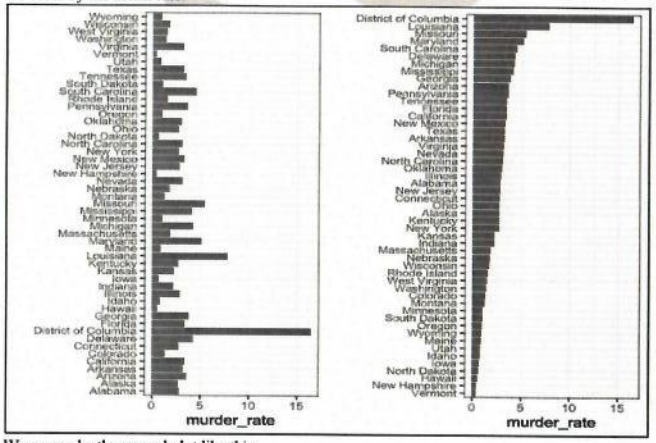
Not surprisingly, some libraries defaults to using area rather than radius. Of course, in this case, we really should not be using area at all since we can use position and length:



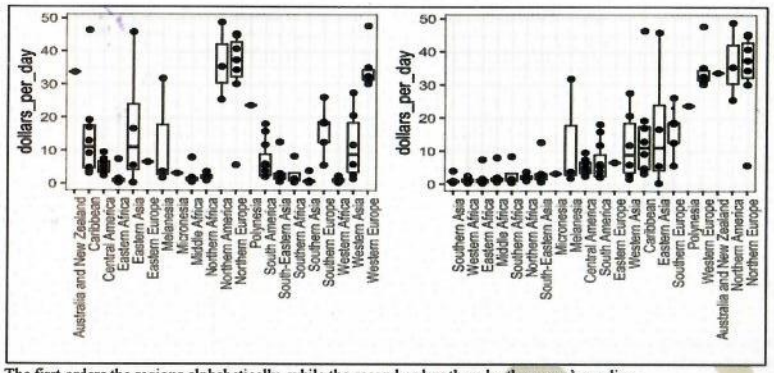
11.4 Order categories by a meaningful value

When one of the axes is used to show categories, as is done in barplots, the default **ggplot2** behavior is to order the categories alphabetically when they are defined by character strings. If they are defined by factors, they are ordered by the factor levels. We rarely want to use alphabetical order. Instead, we should order by a meaningful quantity. In all the cases above, the barplots were ordered by the values being displayed. The exception was the graph showing barplots comparing browsers. In this case, we kept the order the same across the barplots to ease the comparison. Specifically, instead of ordering the browsers separately in the two years, we ordered both years by the average value of 2000 and 2015.

We previously learned how to use the `reorder` function, which helps us achieve this goal. To appreciate how the right order can help convey a message, suppose we want to create a plot to compare the murder rate across states. We are particularly interested in the most dangerous and safest states. Note the difference when we order alphabetically (the default) versus when we order by the actual rate:



We can make the second plot like this:

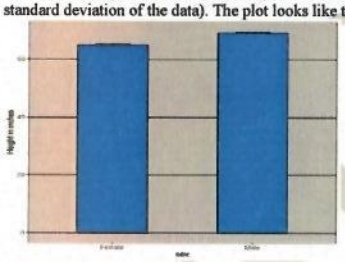


The first orders the regions alphabetically, while the second orders them by the group's median.

11.5 Show the data

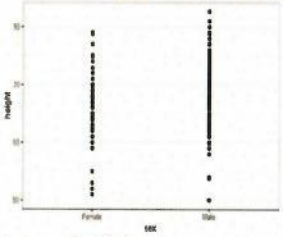
We have focused on displaying single quantities across categories. We now shift our attention to displaying data, with a focus on comparing groups.

To motivate our first principle, "show the data", we go back to our artificial example of describing heights to ET, an extraterrestrial. This time let's assume ET is interested in the difference in heights between males and females. A commonly seen plot used for comparisons between groups, popularized by software such as Microsoft Excel, is the dynamite plot, which shows the average and standard errors (standard errors are defined in a later chapter, but do not confuse them with the standard deviation of the data). The plot looks like this:



The average of each group is represented by the top of each bar and the antennae extend out from the average to the average plus two standard errors. If all ET receives is this plot, he will have little information on what to expect if he meets a group of human males and females. The bars go to 0: does this mean there are tiny humans measuring less than one foot? Are all males taller than the tallest females? Is there a range of heights? ET can't answer these questions since we have provided almost no information on the height distribution.

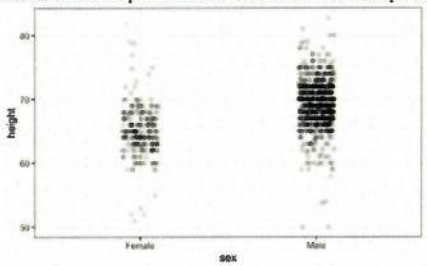
This brings us to our first principle: show the data. This simple **ggplot2** code already generates a more informative plot than the barplot by simply showing all the data points:



For example, this plot gives us an idea of the range of the data. However, this plot has limitations as well, since we can't really see all the 238 and 812 points plotted for females and males, respectively, and many points are plotted on top of each other. As we have previously described, visualizing the distribution is much more informative. But before doing this, we point out two ways we can improve a plot showing all the points.

The first is to add *jitter*, which adds a small random shift to each point. In this case, adding horizontal jitter does not alter the interpretation, since the point heights do not change, but we minimize the number of points that fall on top of each other and,

therefore, get a better visual sense of how the data is distributed. A second improvement comes from using *alpha blending*: making the points somewhat transparent. The more points fall on top of each other, the darker the plot, which also helps us get a sense of how the points are distributed. Here is the same plot with jitter and alpha blending:

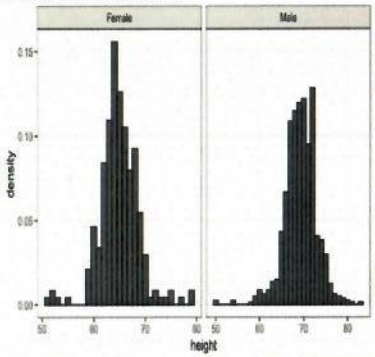


Now we start getting a sense that, on average, males are taller than females. We also note dark horizontal bands of points, demonstrating that many report values that are rounded to the nearest integer.

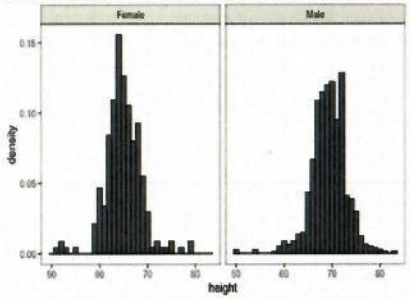
11.6 Ease comparisons

11.6.1 Use common axes

Since there are so many points, it is more effective to show distributions rather than individual points. We therefore show histograms for each group:

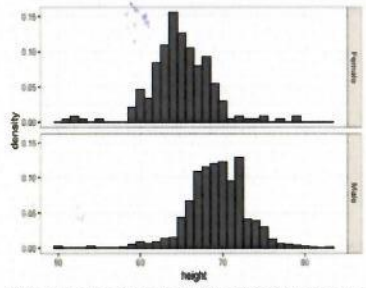


However, from this plot it is not immediately obvious that males are, on average, taller than females. We have to look carefully to notice that the x-axis has a higher range of values in the male histogram. An important principle here is to **keep the axes the same** when comparing data across two plots. Below we see how the comparison becomes easier:



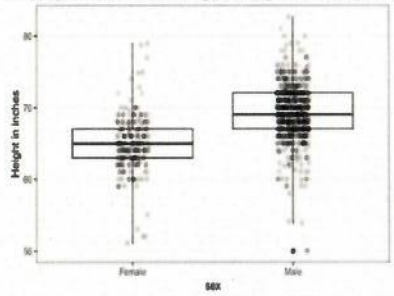
11.6.2 Align plots vertically to see horizontal changes and horizontally to see vertical changes

In these histograms, the visual cue related to decreases or increases in height are shifts to the left or right, respectively: horizontal changes. Aligning the plots vertically helps us see this change when the axes are fixed:

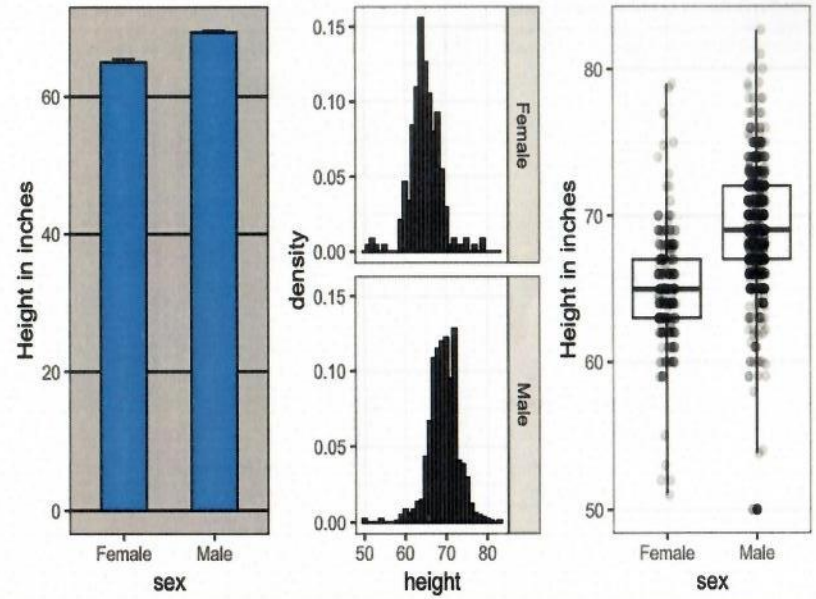


This plot makes it much easier to notice that men are, on average, taller.

If , we want the more compact summary provided by boxplots, we then align them horizontally since, by default, boxplots move up and down with changes in height. Following our *show the data* principle, we then overlay all the data points:



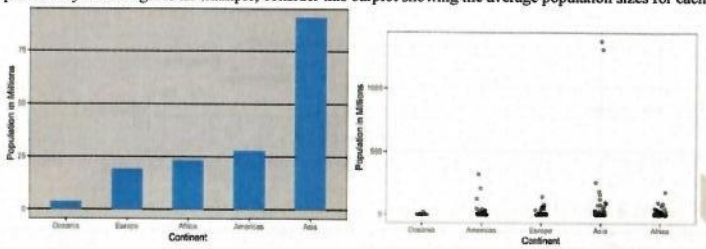
Now contrast and compare these three plots, based on exactly the same data:



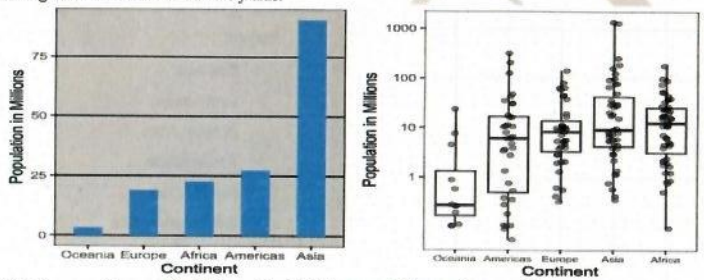
Notice how much more we learn from the two plots on the right. Barplots are useful for showing one number, but not very useful when we want to describe distributions.

11.6.3 Consider transformations

We have motivated the use of the log transformation in cases where the changes are multiplicative. Population size was an example in which we found a log transformation to yield a more informative transformation.
The combination of an incorrectly chosen barplot and a failure to use a log transformation when one is merited can be particularly distorting. As an example, consider this barplot showing the average population sizes for each continent in 2015:



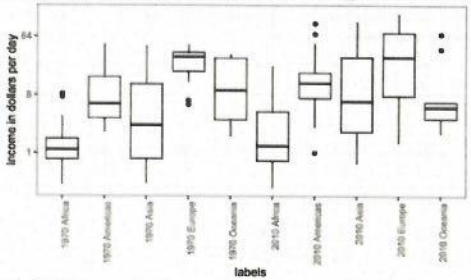
From this plot, one would conclude that countries in Asia are much more populous than in other continents. Following the *show the data* principle, we quickly notice that this is due to two very large countries, which we assume are India and China. Using a log transformation here provides a much more informative plot. We compare the original barplot to a boxplot using the log scale transformation for the y-axis:



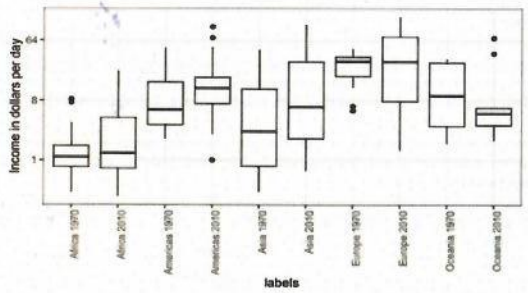
With the new plot, we realize that countries in Africa actually have a larger median population size than those in Asia.
Other transformations you should consider are the logistic transformation (logit), useful to better see fold changes in odds, and the square root transformation (sqrt), useful for count data.

11.6.4 Visual cues to be compared should be adjacent

For each continent, let's compare income in 1970 versus 2010. When comparing income data across regions between 1970 and 2010, we made a figure similar to the one below, but this time we investigate continents rather than regions.

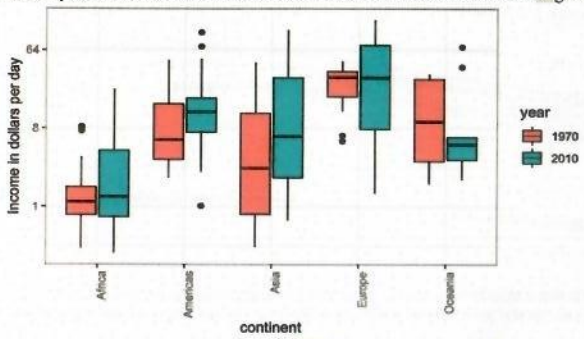


The default in **ggplot2** is to order labels alphabetically so the labels with 1970 come before the labels with 2010, making the comparisons challenging because a continent's distribution in 1970 is visually far from its distribution in 2010. It is much easier to make the comparison between 1970 and 2010 for each continent when the boxplots for that continent are next to each other:



11.6.5 Use color

The comparison becomes even easier to make if we use color to denote the two things we want to compare:

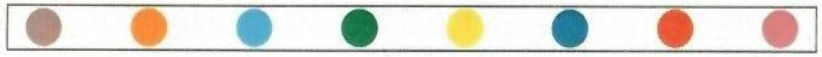


11.7 Think of the color blind

About 10% of the population is color blind. Unfortunately, the default colors used in **ggplot2** are not optimal for this group. However, **ggplot2** does make it easy to change the color palette used in the plots. An example of how we can use a color blind friendly palette is described here: http://www.cookbook-r.com/Graphs/Colors_(ggplot2)/#a-colorblind-friendly-palette:

```
color_blind_friendly_cols <-
  c("#999999", "#E69F00", "#56B4E9", "#009E73",
    "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
```

Here are the colors



There are several resources that can help you select colors, for example this one: http://bconnelly.net/2013/10/creating-colorblind-friendly-figures/.

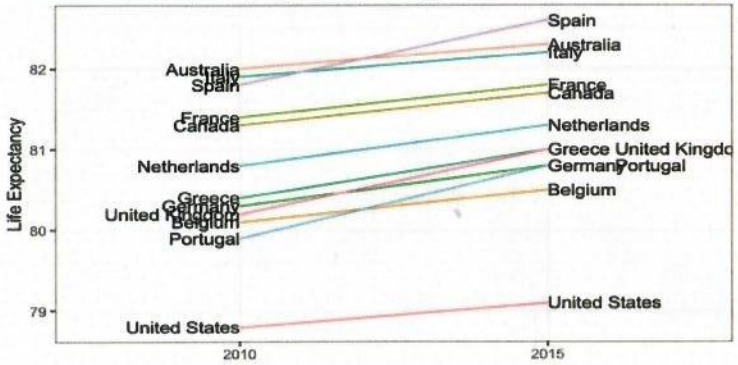
11.8 Plots for two variables

In general, you should use scatterplots to visualize the relationship between two variables. In every single instance in which we have examined the relationship between two variables, including total murders versus population size, life expectancy versus fertility rates, and infant mortality versus income, we have used scatterplots. This is the plot we generally recommend. However, there are some exceptions and we describe two alternative plots here: the *slope chart* and the *Bland-Altman plot*.

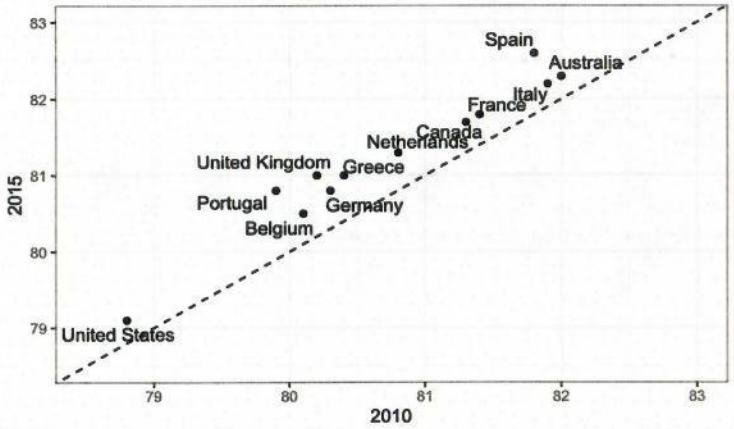
11.8.1 Slope charts

One exception where another type of plot may be more informative is when you are comparing variables of the same type, but at different time points and for a relatively small number of comparisons. For example, comparing life expectancy between 2010 and 2015. In this case, we might recommend a *slope chart*.

There is no geometry for slope charts in **ggplot2**, but we can construct one using `geom_line`. We need to do some tinkering to add labels. Below is an example comparing 2010 to 2015 for large western countries:



An advantage of the slope chart is that it permits us to quickly get an idea of changes based on the slope of the lines. Although we are using angle as the visual cue, we also have position to determine the exact values. Comparing the improvements is a bit harder with a scatterplot:



In the scatterplot, we have followed the principle *use common axes* since we are comparing these before and after. However, if we have many points, slope charts stop being useful as it becomes hard to see all the lines.

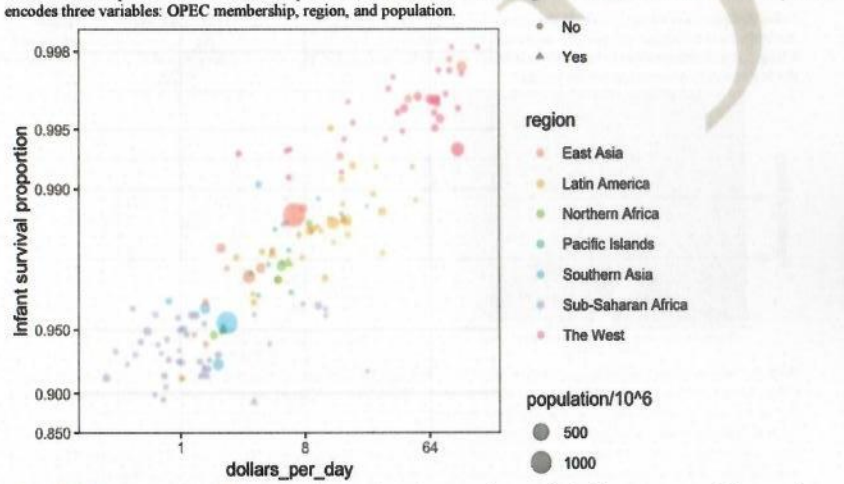
11.8.2 Bland-Altman plot

Since we are primarily interested in the difference, it makes sense to dedicate one of our axes to it. The Bland-Altman plot, also known as the Tukey mean-difference plot and the MA-plot, shows the difference versus the average:

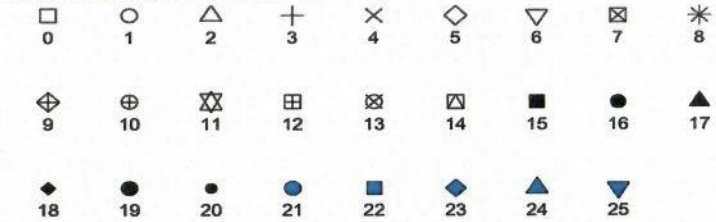
Here, by simply looking at the y-axis, we quickly see which countries have shown the most improvement. We also get an idea of the overall value from the x-axis.

11.9 Encoding a third variable

An earlier scatterplot showed the relationship between infant survival and average income. Below is a version of this plot that encodes three variables: OPEC membership, region, and population.



We encode categorical variables with color and shape. These shapes can be controlled with `shape` argument. Below are the shapes available for use in R. For the last five, the color goes inside.

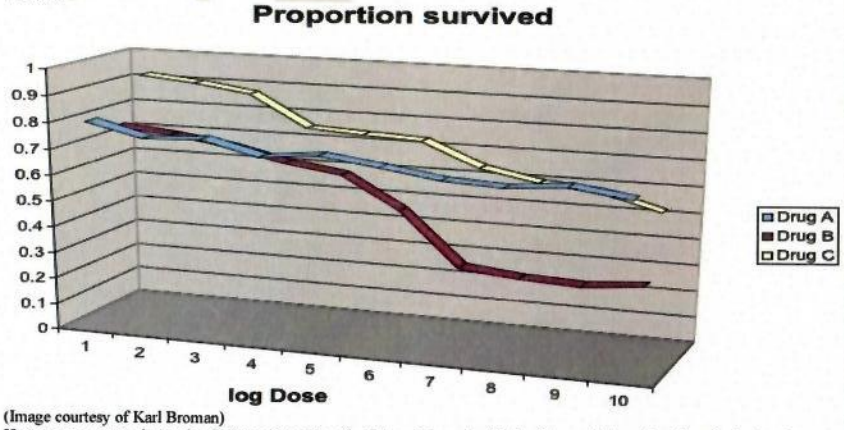


For continuous variables, we can use color, intensity, or size. We now show an example of how we do this with a case study. When selecting colors to quantify a numeric variable, we choose between two options: sequential and diverging. Sequential colors are suited for data that goes from high to low. High values are clearly distinguished from low values. Here are some examples offered by the package

Diverging colors are used to represent values that diverge from a center. We put equal emphasis on both ends of the data range: higher than the center and lower than the center. An example of when we would use a divergent pattern would be if we were to show height in standard deviations away from the average. Here are some examples of divergent patterns:

11.10 Avoid pseudo-three-dimensional plots

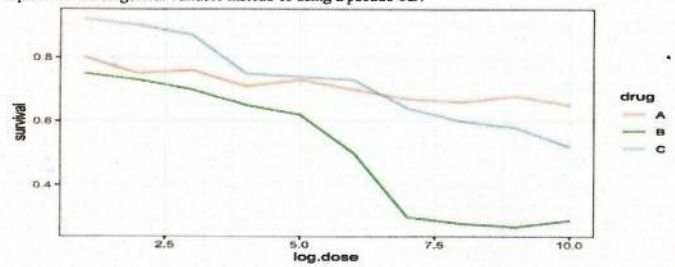
The figure below, taken from the scientific literature[1], shows three variables: dose, drug type and survival. Although your screen/book page is flat and two-dimensional, the plot tries to imitate three dimensions and assigned a dimension to each variable.



(Image courtesy of Karl Broman)

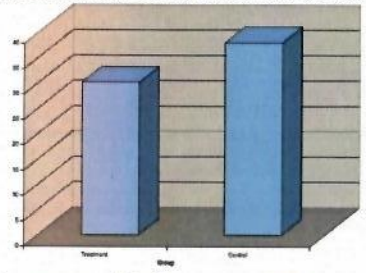
Humans are not good at seeing in three dimensions (which explains why it is hard to parallel park) and our limitation is even worse with regard to pseudo-three-dimensions. To see this, try to determine the values of the survival variable in the plot

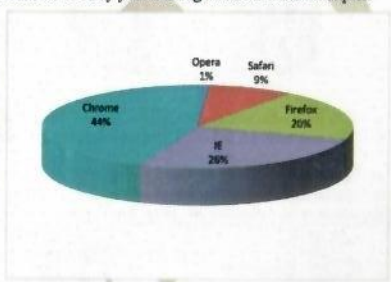
above. Can you tell when the purple ribbon intersects the red one? This is an example in which we can easily use color to represent the categorical variable instead of using a pseudo-3D:



Notice how much easier it is to determine the survival values.

Pseudo-3D is sometimes used completely gratuitously: plots are made to look 3D even when the 3rd dimension does not represent a quantity. This only adds confusion and makes it harder to relay your message. Here are two examples:





(Images courtesy of Karl Broman)

11.11 Avoid too many significant digits

By default, statistical software like R returns many significant digits. The default behavior in R is to show 7 significant digits. That many digits often adds no information and the added visual clutter can make it hard for the viewer to understand the message. As an example, here are the per 10,000 disease rates, computed from totals and population in R, for California across the five decades:

state	year	Measles	Pertussis	Polio
California	1940	37.8826320	18.3397861	0.8266512
California	1950	13.9124205	4.7467350	1.9742639
California	1960	14.1386471	NA	0.2640419
California	1970	0.9767889	NA	NA
California	1980	0.3743467	0.0515466	NA

We are reporting precision up to 0.00001 cases per 10,000, a very small value in the context of the changes that are occurring across the dates. In this case, two significant figures is more than enough and clearly makes the point that rates are decreasing:

state	year	Measles	Pertussis	Polio
California	1940	37.9	18.3	0.8
California	1950	13.9	4.7	2.0
California	1960	14.1	NA	0.3
California	1970	1.0	NA	NA
California	1980	0.4	0.1	NA

Useful ways to change the number of significant digits or to round numbers are `signif` and `round`. You can define the number of significant digits globally by setting options like this: `options(digits = 3)`. Another principle related to displaying tables is to place values being compared on columns rather than rows. Note that our table above is easier to read than this one:

state	disease	1940	1950	1960	1970	1980
California	Measles	37.9	13.9	14.1	1	0.4
California	Pertussis	18.3	4.7	NA	NA	0.1
California	Polio	0.8	2.0	0.3	NA	NA

11.12 Know your audience

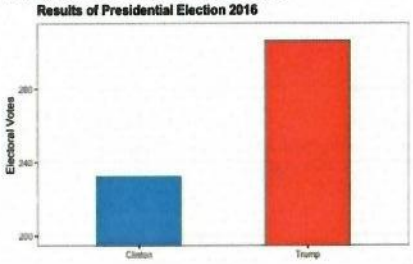
Graphs can be used for 1) our own exploratory data analysis, 2) to convey a message to experts, or 3) to help tell a story to a general audience. Make sure that the intended audience understands each element of the plot.

As a simple example, consider that for your own exploration it may be more useful to log-transform data and then plot it.

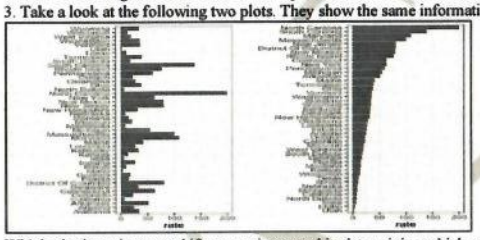
However, for a general audience that is unfamiliar with converting logged values back to the original measurements, using a log-scale for the axis instead of log-transformed values will be much easier to digest.

11.13 Exercises

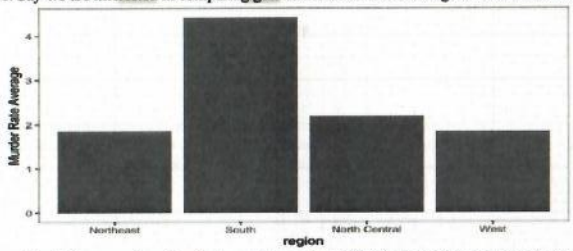
1. Pie charts are appropriate:
   a. When we want to display percentages.
   b. When **ggplot2** is not available.
   c. When I am in a bakery.
   d. Never. Barplots and tables are always better.
2. What is the problem with the plot below:



   a. The values are wrong. The final vote was 306 to 232.
   b. The axis does not start at 0. Judging by the length, it appears Trump received 3 times as many votes when, in fact, it was about 30% more.
   c. The colors should be the same.
   d. Percentages should be shown as a pie chart.
3. Take a look at the following two plots. They show the same information: 1928 rates of measles across the 50 states.

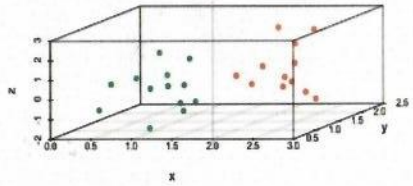


Which plot is easier to read if you are interested in determining which are the best and worst states in terms of rates, and why?
   a. They provide the same information, so they are both equally as good.
   b. The plot on the right is better because it orders the states alphabetically.
   c. The plot on the right is better because alphabetical order has nothing to do with the disease and by ordering according to actual rate, we quickly see the states with most and least rates.
   d. Both plots should be a pie chart.
6. Say we are interested in comparing gun homicide rates across regions of the US. We see this plot:

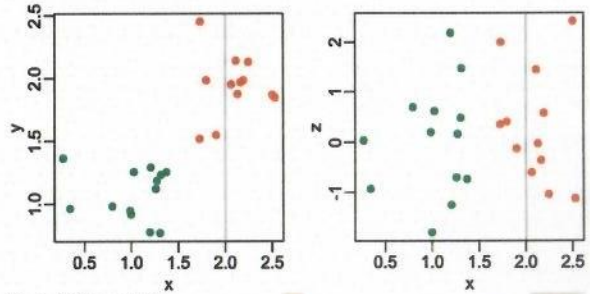


and decide to move to a state in the western region. What is the main problem with this interpretation?
   a. The categories are ordered alphabetically.
   b. The graph does not show standarad errors.
   c. It does not show all the data. We do not see the variability within a region and it's possible that the safest states are not in the West.

   d. The Northeast has the lowest average.
7. Make a boxplot of the murder rates by region, showing all the points and ordering the regions by their median rate.
8. The plots below show three continuous variables.



The line x=2
appears to separate the points. But it is actually not the case, which we can see by plotting the data in a couple of two-dimensional points.



Why is this happening?
   a. Humans are not good at reading pseudo-3D plots.
   b. There must be an error in the code.
   c. The colors confuse us.
   d. Scatterplots should not be used to compare two variables when we have access to 3.

Reference : http://rafalab.dfci.harvard.edu/dsbook/data-visualization-principles.html

Understanding Visual Encoding and the Way We Process Information

If you are wondering, daydreaming, and thinking: how will others process my information? At the launch of a new product or marketing campaign, you need to feel like your user.
When you open up a computer application or web pages such as Twitter or this one, you are instantly greeted with a variety of images, colors, and symbols that look like images but could be....anything. The Twitter bird or hashtag emojis are examples of one kind of visual encoding, while the color in a block header on a website such as this one is another example.
Knowing why we feel a certain way when we see certain things is half the battle when launching a new campaign.
We are here today to answer the question: "how do we process information?" Let's begin seeing through the eyes of your users today.

How do humans Process information?

In a nutshell, we process information from our primary senses. Those are sound, sight, scent, touch, and taste. Thoughts are a form of senses as well but are more part of our processing center as opposed to our intake units.
Our five senses are our intake units. Simply put, when we "sense" something, it's an input method to send a signal to the brain.
The brain then processes that in a way that we can use. Sometimes a scent or a taste can make us both enjoy and remember something at the same time.
That's called encoding.

What is Encoding?

So when we intake information through our senses, multiple responses at the brain level can occur. The number of responses to each sensation is infinite to each human.
There are three kinds of encoding: visual encoding, acoustic encoding, and semantic encoding.
Acoustic encoding processes our sound experiences. And semantic processes how we feel things as well as how we communicate and rationalize things.
So when you read the news and respond emotionally, for example, your brain is engaging in semantic encoding.

This article is organized a certain way visually because the visual encoding that your brain is processing right now prefers to read things this way over any other way.Many human responses are universal across cultures and across the wide globe. For example, we all know what excitement and anger are — they are emotions that don't discriminate in any land.We all encode things similarly across the world. Even if two different countries have two different reactions to the same piece of news, the way we encode the information is the same.

Because those feelings are relatable, you can use that information in your business. Change the way people feel about things by changing the way they see it.Discover these six mind-blowing ways that AI can supercharge your business infrastructure and begin to get a feel for how this works. It's easier than you think.

Why Understand Visual Encoding?

Understanding how those responses occur is essential in the world of AI, marketing, and business. When you understand this, you take your business to the next level. When you know this, you can use scents and tastes in your products, audios in your marketing, emotive visual presentations, or tactile experience. And all of the above. When asking how do we process information, start with visual encoding, and then you can incorporate sensational experiences in the rest of your brand building.

Simply put, we intake something through one of our senses, it gets directed to the appropriate "department" in the brain, and sometimes there is a crossover. The most common crossover in the brain is when our limbic system (our emotional center) crosses over our language center.
For example, we see something in a store window and remember the day we got engaged and tell someone right then and there that we love them. Sometimes when this crossover process happens, we buy something. In fact, for every single purchase we make as humans, some form of crossover must happen. Our limbic system might be happy because we bought it, or not so much because we had to buy it and didn't want to. That's why understanding how we process information visually is important. It can help make your end-user find that magic happy button, so they get excited when they purchase your product. Pursuing that specific response is the Gestaltian method of business that is winning for many Fortune 500s today.

Gestalt Theory on Visual Encoding

The Gestalt theory on visual encoding goes back to German scientists who, in the 1920s, came up with multiple principles on visual encoding. They wanted to know what people see when processing visual cues. It was called "Gestalt" psychology because the word "Gestalt" in German means "putting together." Gestaltian psychology and perception theories run on the notion that the whole is greater than the parts. If you see something Gestaltian, you see it as one object, and not a composition of the multiple objects that are in it. For example, you've probably seen optical illusions in art form or online. There is a famous drawing of something that looks like the face of an older woman one way, and the shape of a younger woman the other way. When the average human sees that, they see one obvious thing, at the Gestaltian level. But once we are told there is a series of parts in any given image, we process the same illusion in a broken down way.

A Business Example of Visual Encoding

Let's take this to an example in the business world – the Mastercard logo. When you see that little blend of orange and red interlocking circles, you know what it is. You see the grouping of interlocking circles and think. "Mastercard." You don't process an orange circle here, a red circle here, the text there, and then think about pulling it together.
You see the sum, not the parts.That is the Gestaltian perception because that is all you know.
This kind of visual encoding we have as a commercial society presents a problem to a business like Mastercard that wants to change their logo. And this happened.
News broke that Mastercard wanted to change the logo but wasn't sure how people would react. In the end, they chose to keep the orange and red circles, but took out the word "Mastercard."
So the logo stayed the same, but for the text component of "Mastercard" in between the orange and red circles. That's because of Gestaltian perception principles of how we undergo visual encoding in our brains. If they had marketed the word alone "Mastercard" it would not be as recognizable to you like the simple orange and red interlocking circles are. That kind of brand recognition is based on understanding visual encoding. So the next person that asks you, how do we process information, the answer is: It's Gestaltian.
Learn more about the 6 fascinating effects data visualization has on decision making.
The three most important Gestalt principles in visual encoding are continuity, proximity, and enclosed. Let's have a look at those. Let's use a random example of a New York deli counter to illustrate how consumers make decisions based on information processed through visual encoding.

Proximity Matters

Proximity is something that our brains register when we're looking at something. Let's say you're short on counter space and want to figure out how to organize your sandwiches, salads, and donuts.
Do so in a way where birds of a feather stick together. In Gestalt principles, our brain tells us that when we see things close together we think they are related.
Give your diners a little nudge to buy some salad with their sandwich by placing sandwiches and salads together in the window or counter. They may never have thought of getting a salad today if you hadn't positioned them as such, and suddenly realize they are hungrier than they thought.

Continuity Principle

The continuity principle stipulates that our brains compute things as being together or related when they are in a line. If you have ever shopped on Amazon or eBay, or even any other major retailer online, you have seen continuity when you see "Related Projects" bars.
You can use your actual line in a New York Deli to provide visual encoding clues to help your consumers buy more. If they need to line up right beside the counter with all of the yummy pastries and desserts before the cash register, so be it.
Another way to bring continuity into your presentation is how you line up things in the window. Keep all lines going either vertical or horizontal, and keep that consistent in your store.
If any of those lines break, the continuity of the visual encoding stops.

Enclosure

In the example above, of the art of the older, versus younger women, enclosure is the Gestalt perception principle we use to see what we see.
The enclosure is a form of visual encoding where our eyes look at something and we want to see one thing and one thing alone. You may go to a New York Deli one day with nothing but a cheeseburger on your mind, and you won't see anything else in the store until you find that menu item.
So use enclosure in your presentations to upsell your customers in your deli as well. A fancy Thanksgiving display is an eye-catching thing that will draw attention as the "one big thing."
But then, your customers will begin to see the parts that make up that sum and perhaps make some additional purchase decisions from there.
And just remember, enclosure encoding doesn't stop there.
You will always be using multiple forms of perception and visual encoding at once. Knowing them will have you seeing what your customer sees and seeing your bottom line increase as a result.

Start Seeing Differently Today

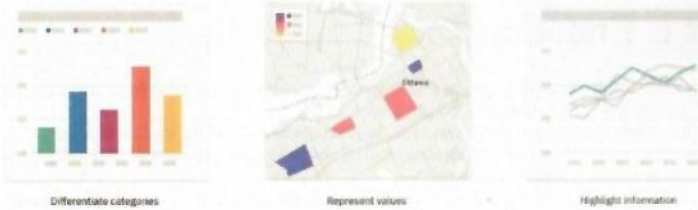
Once you understand how to see what your clients, readers, or customers are seeing, and what they are feeling, you can create any marketing campaign that you want. Or even if you are just looking to bring some artificial intelligence into your staff room, reception hall, or board room, you want to ask yourself, "how do we process information" from the perspective of all who will see this space.
Maybe changing the way your employees feel when they are in the staff room will result in a more cohesive and unified team? You can achieve this by studying how they process the information of the visual cues they are currently working with.
Do they like ugly green walls? Or do they seem in better moods in the board room with the windows?
Enclosures, proximity, and continuity are all Gestaltian principles of visual encoding that can help you make everybody win and elevate your business today. When you have a big message to send or feeling to inspire, discover the answers to the age-old question, "how do we process information" today.

Choose Right Colors

For data visualization, **colour is more than aesthetic**. Depending on your audience and the story you want to tell, the colours you select and the way you use them can have a huge impact on how the information is perceived. Colour can be used to:

- differentiate categories of data, like different ice cream flavours.
- represent values, like ice cream sales by neighbourhood.
- call attention to specific values, like highlighting your neighbourhood's ice cream sales.

When used with purpose, **colour tells a story and brings clarity to your data**, but when used without consideration, colour can work against you.
We saw this firsthand while exploring custom colour palettes for visualizations in IBM Cognos Analytics. You might be thinking, "How could creating custom palettes be a problem? What's the worst that could happen?" Well, after talking to customers, we realized that while they're business and/or analytics experts, they're **not necessarily colour experts**. The freedom to create your own palette, while liberating to our customers, also opened them up to the potential risks involved in using colour if they don't have a solid understanding of how to apply it to data. A poor selection of colours can unknowingly compromise visual clarity or even skew the data.
I'm not pointing this out to scare you. If it's difficult to distinguish the colours on your dashboard, profits aren't going to instantly plummet… or at least it's safe to assume colour palettes weren't the reason. I'm pointing this out so that we can give colour the weight it



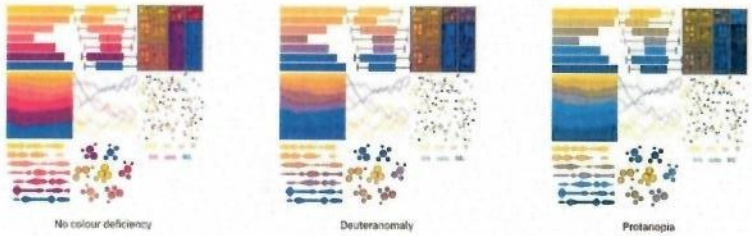
deserves, especially when using it to tell stories with our data. Colour shouldn't be something we unconsciously apply to "make it look pretty" because **colour conveys meaning whether it is intentional or not.**

Factors that complicate colour

On the surface colour seems relatively straightforward, but there are underlying elements at play that add to the complexity of selecting colours for data visualizations and create potential issues for correctly interpreting the information. Common issues with colour include:

Perceptual factors

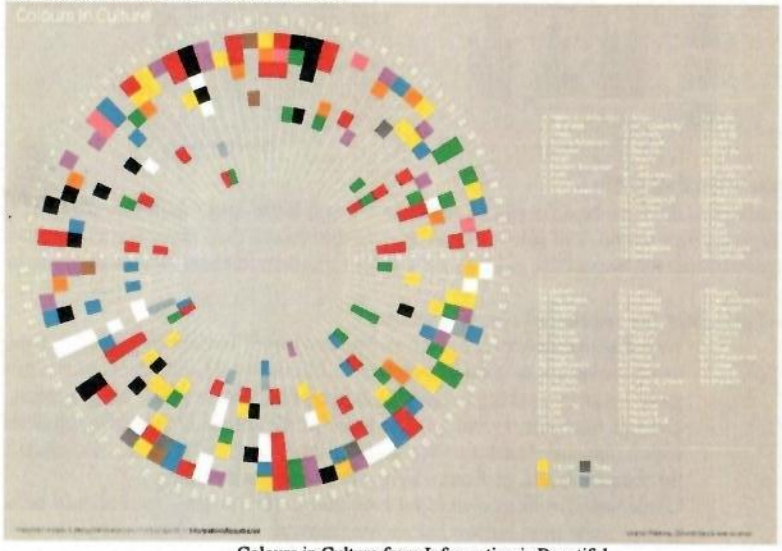
These factors include our ability to see colours and distinguish them from one another. Visual impairments like colour blindness or cataracts, colour differences on a phone compared to a TV or a laptop, or even the size and shape of what you're looking at can effect how easily one colour is identified from another.



Differences in colour perception via Viz Palette

Semantic factors

This is how we interpret meaning from colour. The associations we have with colour depends on the cultural, environmental and personal contexts that we've been exposed to. These differences can be seen in "Colours in Culture" from Information is Beautiful (below). The colour green, for example, is associated with good luck in Arab, Japanese, and Western cultures — while in African, Chinese, and Eastern European cultures, good luck is associated with the colour red.



Colours in Culture from Information is Beautiful

By being aware of these perceptual and semantic factors, you can select colour combinations that avoid common constraints and help focus attention on the message you want to share.

Questions to ask before picking colours

Let's imagine that you're a part owner in the Udderly Delicious Ice Cream Company, a small fleet of ice cream carts in Ottawa, Canada. Here are a few questions you can ask yourself when picking colours for data visualizations.

Who's the audience?

Are you making visualizations for a specific group of people? Are there cultural or industry-specific conventions they use for colour?

Let's say you thought it would be helpful to put together a few visualizations to share with your business partners to help inform budget, sales targets, and other areas to focus on this year.

You know that, when it comes to cultural or industry-specific conventions in Canada, green means good and red means bad. You've also seen blue and red used together when referencing temperature.

By taking a few moments to write down what you know about your audience, their goals & motivations, your story becomes clearer.

What's the story?

Ok, now that we know who the audience is, it's important to focus on the story we want to tell before thinking about colour. What are you trying to explain to your audience? For example, at the Udderly Delicious Ice Cream Company, what are the three most important questions that you and your business partners need answered to inform the business this year?

Maybe you're wondering what the annual sales for the top 5 ice cream flavours are because it will affect what you choose to order for your first batch this year. Or maybe you're wanting to see sales volume by neighbourhood, so that you can strategically distribute the ice cream carts to meet customer demands. Or perhaps you want to visualize customer satisfaction per cart to see if any trends become obvious.

By knowing the story and how it relates to your audience, you'll know how to check if you've achieved what you set out to do. It means you know why sharing this information matters, what data you'll be using — and often this will lead you to the best way to visualize this information.

When to use colour in data visualizations

Now that we know the audience and have figured out what information we want to share, we can focus on colouring the data.

It's important to remember that **colour needs a purpose**. If the information can be understood without it, then don't add colour. As a general rule, if the visualization only has two dimensions of data, like gross profit over the years, then you don't even need a palette, a single colour is perfect.

Keep it simple — only use a palette with 3 dimensions of data

When colouring data visualizations, the type of data being coloured determines the type of colour palette that will be used. We will be focusing on the two most common types of data —qualitative (categories) and quantitative (values).

Qualitative data is information that has no logical order and can't be translated into numeric order. Different ice cream flavours is one example, since "cookie dough" isn't higher or lower than "mint chocolate chip".

Quantitative data is information that implies an order. Daily temperature during the month of August is one example, since the temperature of 32°C on one day is higher than 16°C on another day.

Below, I've aggregated some best practices for creating data visualization colour palettes. To see how colour can be applied to different data, let's use the business questions we've identified for the Udderly Delicious Ice Cream Company.

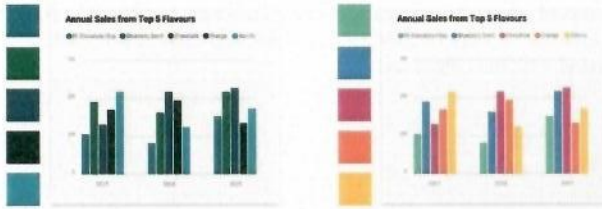
Best practices for creating data viz palettes

Qualitative data

Qualitative data uses categorical palettes with distinct colour swatches to differentiate the categories.

Best practices for categorical colour palettes:

- Swatches should be distinct hues (colours). When colours are too similar it can cause visual clusters, groups, or give a perception of order.
- No one colour should stand out relative to the other colour swatches. The colours should be visually equal in luminance (brightness) and chroma (saturation).
- Palettes should use a small number of colours. People are only able to reliably distinguish 5–8 colours simultaneously.



Quantitative data

Quantitative data can be coloured in one of two ways. If the data follows a path from low to high (or vice versa), like sales volume by neighbourhood, then it can be coloured using a sequential palette. Sequential palettes use a colour gradient to show when values are low vs. high.

Best practices for sequential colour palettes:

- Two distinct colours (i.e. blue and yellow) work better in a gradation than using variations of a single colour (i.e. blue only). Having two colours gives more contrast, making it easier to differentiate colour along the gradation.
- Colours must clearly indicate the order of the palette. It should be obvious which data values are smaller & larger compared to the other data values (i.e. the palette should go from a light colour to a dark colour).
- Gradation of colours should be even across the palette — you should be able to determine how far two values are from one another from any point on the palette.
- Lastly, gradients should be divided into 5 equal steps (or bins). Stepping a gradient makes it easier to distinguish where values are distributed along the scale (if you're looking for an awesome tool to do this for you, I've provided a link at the end).

Not great Follows best practices

Some types of quantitative data, such as temperature or customer satisfaction, are better visualized by divergent palettes. Divergent palettes use a third neutral colour in the middle of a sequential palette to show the change in data values from two directions relative to a neutral midpoint (often zero).

Best practices for divergent colour palettes:

- A neutral colour should be placed at the midpoint of the palette. Light greys, yellows or even white will work. Just make sure the neutral colour is still visible on the visualization's background.
- End colours must be balanced in terms of luminance (brightness) and chroma (saturation), so that the perception of colour progression from dark to light (towards the midpoint) is equal on both ends.
- Gradation of colours should be even across the palette — you should be able to determine how far two values are from one another from any point on the palette.
- Lastly, gradients should be divided into 5 equal steps (or bins). Stepping a gradient makes it easier to distinguish where values are distributed along the scale.

Not great Follows best practices

Final thoughts

As we've learned, when it comes to data visualization, colour can either help or hinder the message you're trying to communicate. My hope is that you've come away with a new appreciation for the role it plays in your story, as well as a frame of reference for applying colour with meaning.

We can bring clarity to the data when we take a moment to think about our audience, the story we want to tell, and which colours carry meaning for them.

I'll leave you with some of my favourite tools for creating data visualization palettes:

- Viz palette: a tool to visualize palettes across visualization types and test for visually similar colours and names
- Chroma.js: a tool for helping create sequential and divergent palettes
- Colorbrewer: a library for sequential and divergent palettes

Data Visualization: Rules for Encoding Values in Graph

When data is communicated graphically, just like verbal communication using language, certain rules of syntax and semantics apply. If you disobey the rules, you run the risk of being misunderstood. The rules of graphical communication are rarely arbitrary, but are usually based on an understanding of visual perception—how we see and the ways in which information can be visually encoded for easy and accurate decoding by our audience.

Most graphs that are used to present quantitative business data are two-dimensional with two axes (one horizontal, called the X axis, and one vertical, called the Y axis), and use one or more of three particular objects to encode values: *points, lines and bars*. The choice of which one or more of these three objects to use in a graph should never be arbitrary, and need never be, because the rules are simple to understand and follow. Why bother? It all boils down to *communication*. If your data is worth reporting, it is worth reporting well.

Points and Their Uses

Points are the data-encoding objects with the simplest possible shape. They pinpoint a specific location on a graph in a way that no other object can, due to the fact that they have negligible height and width. In the context of a 2-D, XY axis graph, points encode values by their location in association with the scale along each axis. In Figure 1, the left-most data point encodes a value of two on the Y axis and is associated with the categorical label "A" on the X axis.

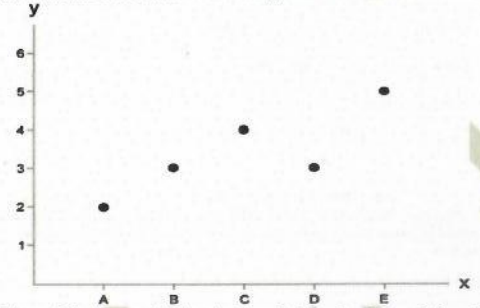


Figure 1: Points encode values based on their location in association with scales along the axes.

Points can take any simple shape, including dots, squares, triangles, diamonds, x's, plus signs, and dashes. When the points in a graph only need to encode a single set of values, and therefore require only a single shape, I prefer to use dots, because they are the simplest visually.

There are only a couple of circumstances when points by themselves (that is, without lines to connect them) are the most effective option. Figure 2 illustrates the primary circumstance. Take a look at it and see if you can determine why points alone in this example are superior to lines or bars.

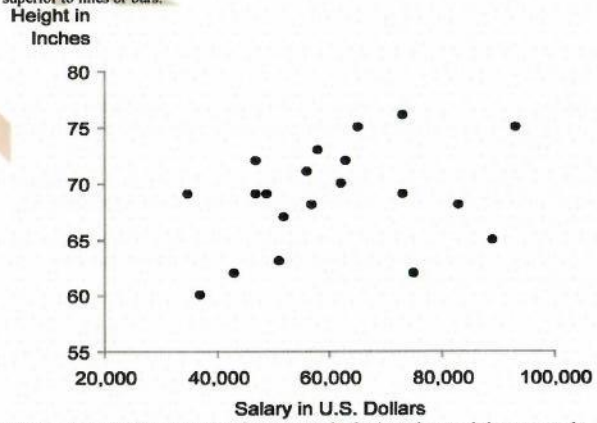


Figure 2: This graph illustrates the primary strength of points when used alone to encode data.

If the reason isn't clear, let me give you a hint. Notice that the scales on both axes are quantitative. Except in the case of this kind of graph, called a *scatter plot*, one of the axes on a

graph has a categorical scale—one that labels what is being measured (for example, departments or regions) rather than quantitative values. Every point on the graph in Figure 2 encodes two quantitative values: one along the X axis (in this case a person's salary in dollars) and one along the Y axis (in this case a person's height in inches). Scatter plots include two quantitative scales because they are specifically designed to display the correlation (or lack of one) between two sets of measures, in this case whether there is a correlation between how tall someone is and the salary that person earns. Imagine using bars to encode this data. Bars extending into the graph from both axes would produce a cluttered mess. Now try to imagine using lines. Lines connecting each of these data points would produce a meaningless, meandering squiggle. Only a point can be used to simultaneously encode two quantitative values based on a horizontal and vertical location, because points alone, without height and width, occupy space in the minimum way that is needed to do this.

There is one other circumstance where points alone work better than lines or bars, but we'll come back to that a little later. Before moving on, however, I want to make it clear that points alone are never a good option for encoding a series of values through time (that is, *time-series data*). It is too hard to follow the chronological sequence of the values when points alone are used, as you can see in Figure 3.

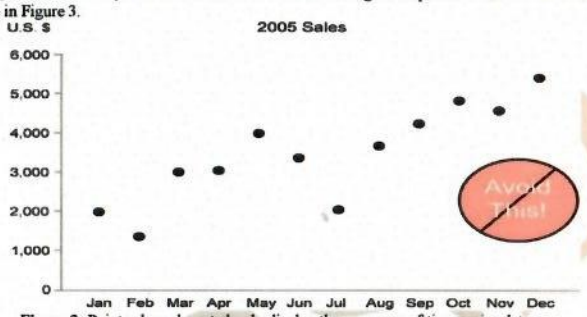


Figure 3: Points alone do not clearly display the sequence of time-series data.

When points are connected by lines, however, the sequence of time-series data becomes easy to follow, illustrated in Figure 4.

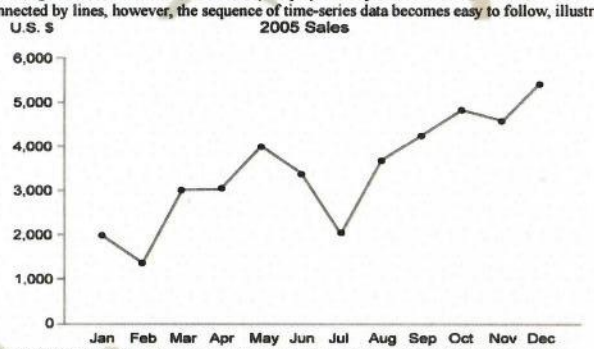


Figure 4: By adding lines to connect the points, the sequence of time-series data can be clarified.

Lines and Their Uses

Lines can be thought of as points extended through space from one value to the next to connect them. Similar to points, lines encode values by means of their location in relation to the scales along the axes; unlike points, only the ends of each segment of the line mark the locations of values.

The strength of lines is more obvious than points or bars. Looking at the graph in Figure 4 above, it is easy to see that lines, by connecting each value to the next, show the overall shape of the values in a way that points or bars cannot. Figure 5 shows the same data as Figure 4, this time using only lines.

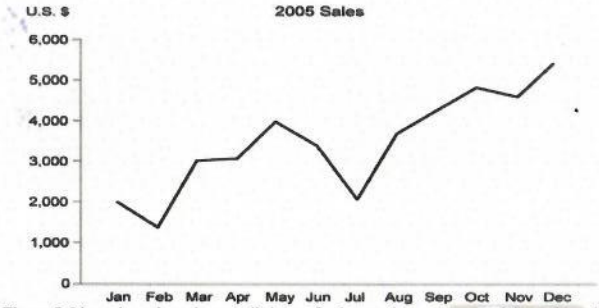


Figure 5: Lines alone show the overall shape of values as they move up and down through time.

As you can see, lines alone display the overall shape of data as it changes through time more clearly than any other encoding method. Nothing distracts from the pure movement of the data up and down from one value to the next. Lines excel in their ability to show trends and patterns of change.

When you wish to emphasize the overall shape of time-series data but must also place additional focus on the individual values, the combination of points and lines works well. This is especially useful when you display multiple data sets across a time series, each as a separate line, and want to make it easy to compare individual values at particular points in time to one another. Including points in Figure 6 makes it a little easier to compare the three values for the month of May, for example, than it would be if the data were encoded using lines alone.

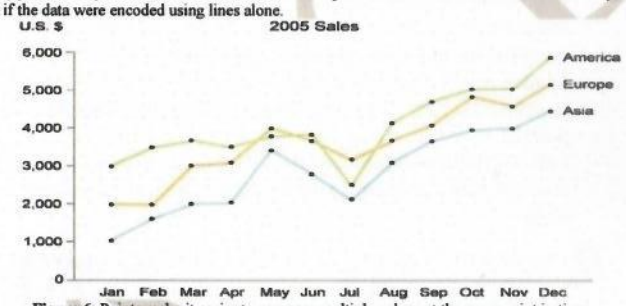


Figure 6: Points make it easier to compare multiple values at the same point in time.

One of the most important guidelines to keep in mind about lines is that they should only be used to connect values that are themselves intimately connected to one another. Changes in the amount of sales from one month, quarter, or year to the next are intimately connected to one another. On the other hand, a set of values that measure the expenses of different departments are not intimately connected; they are discrete and should be displayed in a manner that visually suggests their discreteness. Connecting discrete values with a line does not properly depict the relationship between those values, as shown in Figure 7.

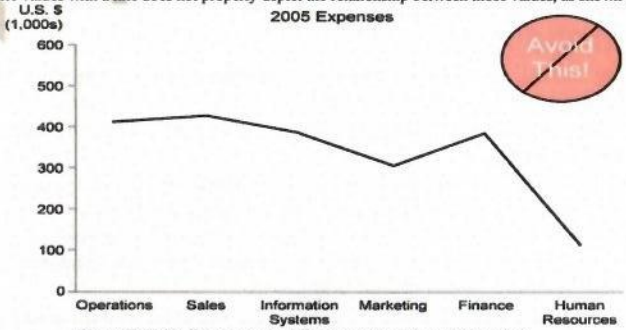


Figure 7: Using lines to connect discrete values doesn't make sense.

When lines are used properly in graphs, their slope is meaningful. For instance, with time-series data, the slope of the line from one value to the next represents the rate of change—the steeper the slope the greater the rate—but with discrete values,

such as those for each of the departments above, the pattern formed by the line and the slopes from one value to the next have no meaning.

Bars and Their Uses

Bars are visually the most weighty and dominant of the three objects that we commonly use to encode data in graphs. Unlike points and lines, which encode values as location relative to a scale along an axis, bars encode quantitative values in two ways. Similar to points and lines, the location of a bar's endpoint encodes its value. Unlike them, a bar's length (or height, depending on how you look at it) also encodes its quantitative value. When bars are properly used, you can compare their values by comparing their lengths. Because bars have a great deal of visual weight and stand out so clearly as separate from one another, it is very easy to compare their lengths to determine the relative magnitudes of the values they encode. Take a look at Figure 8 to see how well this works.

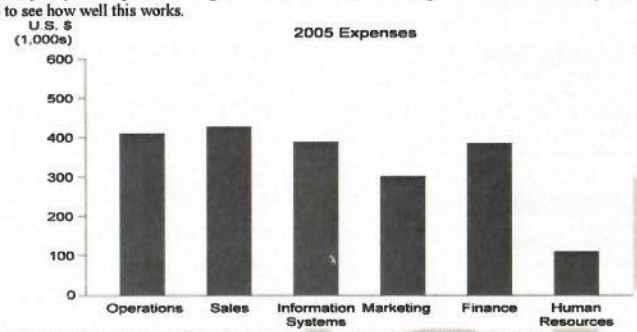


Figure 8: Bars do a great job of representing discrete values and supporting their comparisons.

When you wish to help your audience focus on individual values and compare individual values to one another, bars are ideal. This works especially well for discrete values that are not intimately connected to one another. Bars can be used, however, to encode values along a series of intimately connected values, such as a time series, when you're less concerned about showing the overall shape of the values (which a line would do better) and more concerned about helping people examine and compare individual values. Figure 9 illustrates an occasion when bars work well for time-series data, because the primary purpose is to help people compare actual to budgeted expenses at a particular point in time rather than to see the overall trend of those expenses through time.

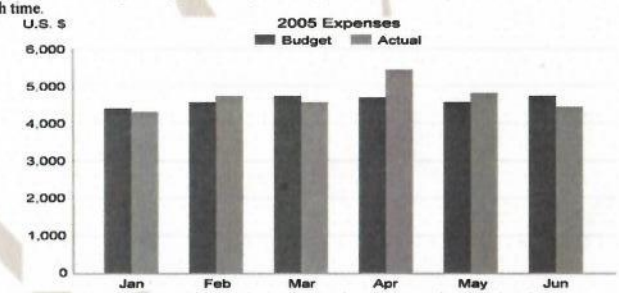


Figure 9: Bars can be used to encode time-series data when you intend primarily to support the examination and comparison of individual values.

Figure 9 can help us learn another lesson about bars. Notice how all of the bars fall within the range of $4,250 and $5,500. They are also all fairly tall, and, with only one exception, fairly close in size. Sometimes when values fall within a relatively narrow range and are all far from zero, it is useful to narrow the quantitative scale such that it begins just below the lowest value and ends just above the highest value. This spreads the values over more space in the graph, enhancing their differences so they can be examined in greater detail. Look what happens, however, when we do this with bars.

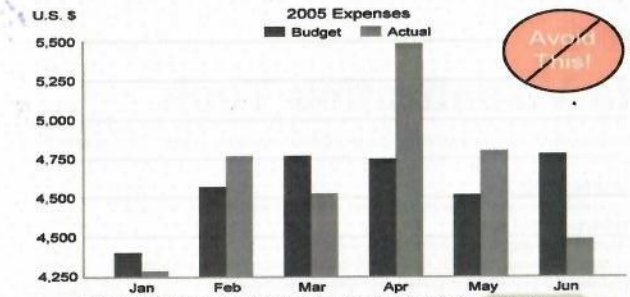


Figure 10: Bars don't work unless the quantitative scale begins at zero.

We now have a problem, because the length of each bar is supposed to represent its value such that comparisons of their lengths support accurate comparisons of their values. Based on a comparison of bar lengths, actual expenses in January appear to be about 1/15th the amount of those in February, but this is hardly the case. For bars to work properly, they can only be used with a quantitative scale that begins at zero. What do you do, then, when you wish to narrow the quantitative scale to make more fine-tuned comparisons of the values? With the data in Figure 10, we could use lines and points, because lines can be used to display time series, but what if the values are discrete and not intimately related? The answer is to take advantage of the discrete appearance of points, which can be used with a narrowed quantitative scale that does not begin at zero, because they encode values using location alone, not length as bars do. Figure 11 provides an example of a point graph that solves the problem.

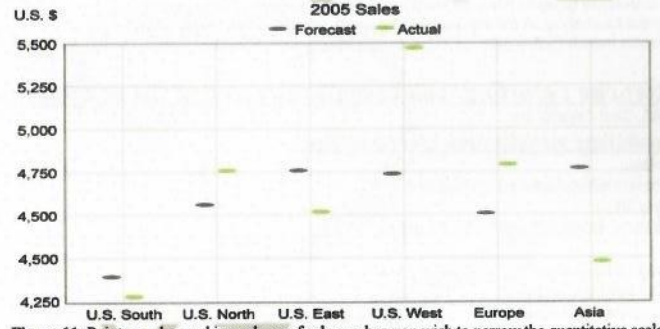


Figure 11: Points can be used in exchange for bars when you wish to narrow the quantitative scale, but still focus on individual discrete values and their comparisons.

These guidelines for the effective use of points, lines and bars to display quantitative data in graphs are quite easy to understand and follow using just about any software, from *Microsoft Excel* to the most sophisticated business intelligence software available. As you can see, these guidelines aren't arbitrary, and neither is the effective communication that results when you follow them. Happy graphing!

Part 4
Data Visualization Tools

• Visualization products have been evolving fast, and there is increasing overlap. But they generally fall into three major categories.

• Standalone tools
– They are specifically designed to produce stunning visualizations, and can work with multiple platforms and data sources.
– Some of them are growing to more full stack analytics tools.
– Examples include Tableau, Power BI, Qlik, SpotFire, and others. They can be desktop based or cloud based (http://idashboards.com, http://www.klipfolio.com).

• Embedded tools
– Broader analytics, business intelligence, and reporting platforms that often incorporate visualization capabilities. These products can address more complex data platform needs and often provide wide-ranging capabilities but may require more training in order to exploit their full potential. In some cases, IT may need to be looped in to assist in integrating these tools with underlying data and related applications.
– Examples like SSRS, IBM, Oracle, MicroStrategy, SAP Crystal, and others.

• Visualization libraries or services
– These tools are offered as programming libraries or services for general applications (web, mobile, etc.).
– These tools can be useful when the visualization requires complete customization, substantial interactivity, or for developing a framework that allows you to reuse code.
– Examples include Python libraries, D3.js, Google Charts, dotNetCharting, Telerik, Nevron, amCharts, etc.

Enterprise reporting tools (usually as a part of the complete BI system)
– SSRS, SAP Crystal, etc.

• Standalone visualization tool (desktop)
– Tableau:
http://www.tableausoftware.com/public/
– Power BI
– QlikView, Dundas, Spotfire, SAP Lumira,
etc.
• Cloud (web) based
– http://idashboards.com
– http://www.klipfolio.com/

• Embedded tools
– Microsoft Excel, Visio
– Google Docs Spreadsheet
http://www.benlcollins.com/spreadsheets/dynamic-dashboard-in-googlespreadsheets/

• Developer oriented libraries and APIs
– Programming library: Python Libraries, D3, dotNetCharting, Telerik, Nevron, amCharts, etc.
– Web API: Google Charts(https://developers.google.com/chart/)

• Casual charting tools
– Google Chart creators:http://dexautomation.com/googlechartgenerator.php
– Other free online charting tools
• http://www.onlinecharttool.com/
• http://nces.ed.gov/nceskids/createagraph (forkids)

• More
– http://selection.datavisualization.ch
– https://www.g2crowd.com/categories/datavisualization
– http://www.creativebloq.com/designtools/data-visualization-712402

Skills in Data Visualization Development

• Data visualization draws knowledge and experience from multiple fields including: computing, business, and design.

• Most important Skills
– Visualization design: charts, diagrams, maps, etc.
– UI and interaction design
– Business domain knowledge

• Highly useful Skills
– Programming/scripting
– Familiarity of the tool
– Data models
– Data preparation
– Analytics methods

• Very helpful Skills
– Artistic design
– Communication, story telling
– Information behavior