ANALYTICS PULSE

Aug - Sep 2025

Predictive Analytics Solutions Pvt. Ltd.

Picking the **Correct Data Visualizaton in Research Analysis**

Introduction

In modern research, data is both the foundation of evidence and the driver of decisionmaking. Yet raw data in spreadsheets or databases rarely speaks for itself; it must be transformed into a format that highlights key insights, relationships, and patterns. This is where visualization plays a critical role. A correct visualization enhances comprehension, fosters accurate interpretation, and supports the credibility of research findings. Conversely, an inappropriate or misleading visualization can obscure truths, distort meaning, or lead to erroneous conclusions.

The infographic designed by Dr. Andrew Abela and Nicolas, refer Figure – 1, provides a structured framework for visualization selection by posing a simple but powerful question: *"What would you like to show?"* It then categorizes visualizations into four fundamental purposes—comparison, relationship, distribution, and composition—each aligned with specific chart types. This framework not only provides practical guidance but also trains researchers to think critically about their objectives before choosing a chart. In the following sections, we explore these categories in depth, contextualize them with examples, and highlight their value for effective research communication. Data visualization isn't just about charts—it's about telling a story with clarity and impact.

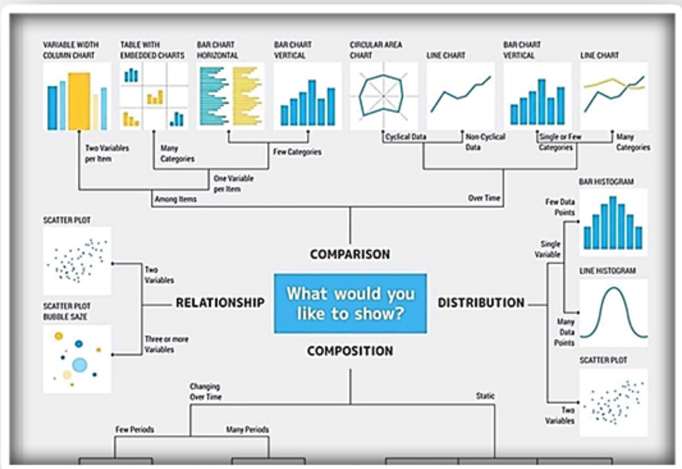


Figure – 1: What would you like to show?

Theoretical Framework for Visualization Selection

The theoretical basis for visualization selection lies in the alignment between research intent and data characteristics. Every dataset can be approached from different analytical angles, but clarity emerges only when the visualization is tailored to the central question. The framework provided in the infographic builds on this principle by distinguishing four distinct purposes:

- Comparison:
  Highlighting differences between groups, categories, or time periods.
- Relationship:
  Exploring how two or more variables interact with one another.
- Distribution:
  Showing how data points spread across a range of values.
- Composition:
  Demonstrating how parts contribute to a whole, either at a moment in time or across periods.

These categories are not arbitrary—they reflect fundamental modes of reasoning in data analysis. Each mode addresses a different type of research question, and the chosen visualization must reveal insights with minimal distortion. This theoretical structure empowers researchers to move beyond aesthetics and base visualization decisions on analytical logic.

Comparison Visualizations

Comparison is the most common visualization purpose in research because scholars often wish to examine differences across categories, conditions, or groups. For example, a public health researcher might want to compare vaccination rates across countries or a computer scientist might compare algorithm performance across datasets.

For few categories, bar charts (vertical or horizontal) are ideal since they clearly display differences in magnitude. When many categories are involved, a table with embedded charts or grouped bar charts may work better to prevent clutter. Line charts are particularly powerful when the comparison involves changes over time, especially with many observations. For cyclical patterns—such as seasonal sales or weather fluctuations—circular area charts can visualize repetition effectively.

The key insight here is that comparisons must remain readable: too many categories in a bar chart or too many lines in a line chart overwhelm the viewer. Researchers must balance clarity with completeness by selecting the chart type that emphasizes the most relevant differences.

Relationship Visualizations

Relationship visualizations aim to reveal whether and how variables are related. They are particularly useful in hypothesis testing and exploratory research. For instance, economists may ask whether education level correlates with income, while computer scientists might test whether model accuracy improves with more training data.

The scatter plot is the cornerstone of relationship visualizations. With two variables, it allows researchers to detect linear, curvilinear, or even non-existent relationships. Adding a third variable through bubble size or color turns the scatter plot into a multi-dimensional tool, making it possible to reveal richer interactions. For example, in medical studies, scatter plots can compare body mass index (BMI) and blood pressure while using bubble size to represent age group.

By focusing on relationships, these visualizations enable researchers to validate correlations, identify outliers, and generate insights that are not immediately apparent from summary statistics alone.

Distribution Visualizations

Distribution visualizations help researchers understand how values are spread across a dataset, identifying ranges, concentrations, and anomalies. They are particularly important in disciplines such as education, epidemiology, and finance where variability matters as much as averages.

For a single variable, the histogram is the primary tool. It shows frequency distribution across intervals, revealing whether data is normally distributed, skewed, or multimodal. Line histograms may also be used for smoother representation of distributions with many data points. When examining two variables simultaneously, scatter plots are effective in illustrating how data points are dispersed.

Consider an education study measuring exam scores: a histogram would reveal whether most students scored near the average or if there were sharp divides between high and low performers. By showing how data is distributed, researchers gain a deeper understanding that raw averages alone cannot provide.

Composition Visualizations

Composition visualizations illustrate how parts contribute to a whole, either in a snapshot or over time. This is especially useful in policy, business, and sustainability research where proportional analysis is central.

For static compositions, pie charts work when showing simple proportions (e.g., percentage of energy sources in national consumption). However, when proportions change over time, stacked bar or stacked area charts become more effective since they show both relative and absolute contributions. Waterfall charts are useful when analyzing sequential additions or subtractions, such as tracking budget inflows and outflows. For hierarchical data, tree maps can reveal nested structures of composition.

The strength of composition visualizations lies in their ability to connect detail to the larger picture, ensuring that research communicates not just the whole but also its meaningful components.

Dynamic vs. Static Visualization

A critical consideration in visualization design is whether the data is dynamic (changing over time) or static (fixed snapshot). Static visualizations such as pie charts or bar charts are useful for presenting one-time comparisons. Dynamic data, however, requires visualization methods that can show change—line charts, stacked area charts, or animated dashboards.

For example, a single pie chart can show the energy mix in India in 2023, but a line chart or stacked area chart is necessary to reveal how renewable energy share has grown across a decade. Choosing between static and dynamic visualization ensures the representation matches the temporal nature of the dataset.

Practical Guidelines for Researchers

To effectively apply the framework, researchers should follow these guidelines:

1. Start with the research question. Clarify whether the aim is to compare, show relationships, display distributions, or illustrate composition.
2. Match variables to chart types. Account for the number of variables, categories, and whether time is a factor.
3. Prioritize clarity. Avoid complex visuals if a simpler chart communicates the message.
4. Prevent misrepresentation. Ensure axes, scales, and proportions are accurate and not misleading.
5. Design for the audience. Test the visualization with peers or non-expert readers to ensure comprehension.
6. Balance detail with readability. Overly detailed visuals can overwhelm, while overly simple ones may omit critical insights.

By following these steps, researchers can maximize the communicative power of their data. For more details refer, Figure – 2 which indicate choices related that is the data we want to represent continuous or categorical.

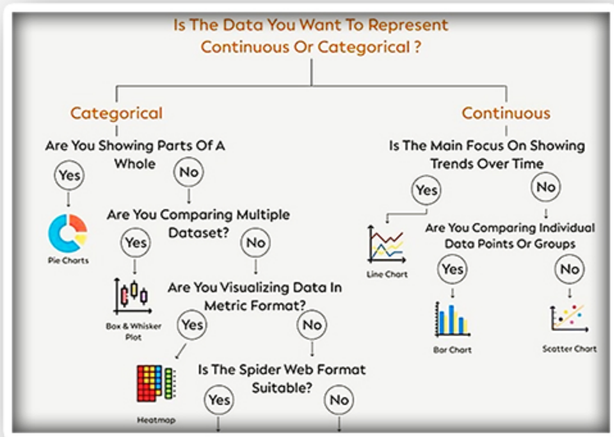


Figure – 2: For Data Visualization

Limitations of the Framework

While comprehensive, the framework is not without limitations. First, it primarily covers basic visualization types, but does not extend to advanced or interactive graphics like heatmaps, network graphs, or geospatial maps, which are increasingly important in big data research. Second, it assumes visual literacy on the part of the researcher and audience, which may not always be the case. Third, cultural and contextual differences in interpreting visuals are not addressed—for example, color perceptions differ across regions. Finally, with the rise of real-time data and dashboards, static chart-based frameworks may not fully capture the needs of dynamic, streaming data visualization. Researchers must therefore use this framework as a foundation but adapt it to evolving tools and contexts.

Conclusion

Selecting the correct visualization is a critical step in research data analysis, bridging the gap between numbers and narrative. The framework by Abela and Nicolas provides a logical, question-driven pathway that encourages researchers to align visualization choices with their analytical goals. By categorizing visualizations into comparison, relationship, distribution, and composition, and by considering the static or dynamic nature of data, the framework equips researchers with a practical decision-making toolkit.

Ultimately, the right visualization clarifies, simplifies, and persuades—it allows researchers to tell a story that is both accurate and impactful. In the age of information overload, this ability is not just a technical skill but a scholarly responsibility.

Article By

Prof. (Dr.) Rudra Rameshwar, India

International and National Level Workshops Trainer - FDP, SDP and MDP Resource Person (Quantitative & Qualitative Research), Research Data Analysis Expert, Member - Asia-Africa Development Council, Member, Editor in Chief- World Academy of Informatics & Management Sciences, Technical Committee Member - National and International Conferences, Accredited and Certified International Master Trainer from FITS, USA, CEDx Certified Trainer, Motivational Speaker, Researcher, Academician, Writer, Career Counselor, Master Trainer & Consultant - Soft/Essential Skills, Ph.D. Guide/Examiner, UG and PG Courses, Examination Paper Setter, Inspirational Coach, Life Coach and Incident Teller, Author - Reviewer and Guest Editor of various reputed International Journals - IGI Global, Emerald, Inderscience, Taylor and Francis, Elsevier, Springer, AOM, BAM, EURAM etc.

----X----

Unlocking Consumer Behavior:
The Power of Market Basket Analysis

Introduction: What's in Your Basket?

Imagine walking into a grocery store to buy milk — and walking out with cereal, eggs, and chocolate. You didn't plan it, but the store did. Behind this intuitive alignment lies a powerful analytical technique called Market Basket Analysis (MBA) — a cornerstone of modern retail strategy. It's the reason "Customers who bought this also bought..." has become a hallmark of your Amazon and Flipkart experience.

In today's age of data-driven decision-making, Market Basket Analysis helps businesses not just understand what customers buy — but what they buy together.

A Brief History: From Grocery Stores to Big Data Giants

The origins of MBA trace back to the early 1990s, when large grocery chains began mining point-of-sale data to uncover product purchase patterns. One popular (though likely apocryphal) tale from this era tells how a retail chain discovered men buying diapers often also bought beer — an insight that led to them being placed together, boosting sales.

This anecdote highlights the essence of MBA: finding association rules between items in large datasets. Since then, MBA has evolved into a core application of unsupervised learning within the field of data mining and is widely used across industries like retail, banking, healthcare, and e-commerce (Tan, Steinbach, & Kumar, 2006).

The Math Behind the Magic:
Support, Confidence, and Lift

Before diving into an example, it's important to understand the core metrics that drive MBA :

Support – How frequently an itemset appears in the dataset
Support(A) = Number of transactions containing A / Total number of transactions Range: 0 to 1

Confidence – The likelihood that item B is bought when item A is bought
Confidence(A → B) = Support(A∩B) / Support(A)
Range: 0 to 1

Lift – How much more likely B is to be bought with A than without A.
Lift(A → B) = Confidence(A → B) / Support(B)
Range: 0 to infinity

where,
Lift > 1: Positive association among products in baskets
Lift = 1: Independent in baskets
Lift < 1: Negative association among products in baskets

Let's consider a small dataset of 5 transactions:

Transactions	Item Purchased			
T1	Bread	Milk		
T2	Bread	Diaper	Beer	Eggs
T3	Milk	Diaper	Beer	Coke
T4	Bread	Milk	Diaper	Beer
T5	Bread	Milk	Diaper	Coke

Here we want to analyze the rule: {Diaper} → {Beer}

Step 1: Count Support
- Total Transactions = 5
- Transactions with Diaper = T2, T3, T4, T5 → 4
- Transactions with Diaper and Beer = T2, T3, T4 → 3
- Transactions with Beer = T2, T3, T4 → 3

Support(Diaper) = 4/5 = 0.8
Support(Diaper → Beer) = 3/5 = 0.6

Step 2: Calculate Confidence
Confidence(Diaper → Beer) = 3/4 = 0.75

Step 3: Calculate Lift
Support(Beer) = 3/5 = 0.6
Lift = 0.75 / 0.6 = 1.25

Interpretation:
- The support of 0.6 means 60% of transactions included both Diaper and Beer.
- The confidence of 0.75 tells us that in 75% of the cases where Diaper is bought, Beer is also bought.
  A lift of 1.25 implies a positive association — customers who buy diapers are 25% more likely to also buy beer than the average buyer.

Where Is Market Basket Analysis Used?

Retail & E-commerce: From Amazon's recommendation engine to supermarket layouts, MBA helps tailor the customer journey and drive upselling (Agrawal etal., 1993).
2. Banking & Finance: Understanding service bundling — customers who opt for savings accounts may also be interested in credit cards or investment accounts.
3. Healthcare: Detecting drug interactions or co-prescription patterns.
4. Hospitality & Food Delivery: Food chains like Domino's use MBA to recommend sides and drinks based on the pizza ordered.
For instance, Amazon and Flipkart have recently optimized their recommendation engines using this approach,

leading to significant growth in average order values. Flipkart implemented dynamic product bundling for mobile phones and accessories, resulting in a noticeable increase in cross-category conversions.

Challenges and Cautions

While MBA is insightful, it must be interpreted carefully:

- Spurious correlations: High lift doesn't always mean causation.
- Data sparsity: In large catalogs, associations may become too granular.
- Privacy concerns: Deep mining of consumer data needs to balance ethical use and consent.

Conclusion: Insights in Every Cart

Market Basket Analysis has transformed the way businesses understand consumers. What began as a grocery store tactic is now a strategic tool across industries. In a world increasingly driven by personalized experiences, knowing what's in the basket may just be the key to what's in the mind.

Next time you find yourself buying unexpected combos online, know that somewhere, an algorithm predicted it — and it all started with Market Basket Analysis.

Sometimes, customers may not even realize what they need until it is suggested to them.

That's where Market Basket Analysis steps in — by creating meaningful product bundles or recommending add-ons, businesses can gently nudge consumers toward a more satisfying shopping experience while simultaneously increasing sales.

MBA is also popularly known as the Apriori algorithm, based on the classic method used to identify frequent item sets by leveraging previously found patterns in a level-wise search. Additionally, it is also commonly referred to as:

- Association Rule Mining
- Affinity Analysis

These terms are often used interchangeably, especially in the context of retail analytics and recommendation systems. Here's a quick explanation of each:

Market Basket Analysis: Focuses on finding product combinations frequently bought together.

Apriori Algorithm: A popular algorithm used to perform market basket analysis by identifying frequent itemsets.

Association Rule Mining: The broader data mining technique used to uncover relationships between variables (e.g., "If a customer buys bread, they are likely to buy butter").

Affinity Analysis: A marketing term used to describe the same process in customer behavior studies.

References

Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. ACM SIGMOD Record, 22(2), 207–216.

Tan, P. N., Steinbach, M., & Kumar, V. (2006). Introduction to Data Mining. Boston: Pearson/Addison Wesley.

Han, J., Pei, J., & Kamber, M. (2011). Data Mining: Concepts and Techniques. Elsevier.

Article By

Prof. Rohit Lala
Assistant Professor, Program Coordinator -BBA BA,
Faculty of Management Studies,
Marwadi University, Rajkot

Prof. Paras Rughani
HoD -BBA,
Faculty of Management Studies,
Marwadi University, Rajkot

----x----

XLSTAT:
A Comprehensive Tool Balancing Accessibility and Analytical Power

Introduction

XLSTAT is a powerful Microsoft Excel add-on that integrates over 220 statistical tools ranging from basic descriptive analysis to advanced modelling, machine learning, and data mining into a familiar, no-coding interface. It supports data preparation, cleaning, customizable visualizations, and automated reporting, enabling more than 150,000 users in 120 countries to efficiently analyze, visualize, and share insights directly within Excel.

Why Choose XLSTAT?

Seamless integration with Microsoft Excel: XLSTAT works directly within Excel, allowing users to analyze, visualize, and model data without leaving the familiar spreadsheet environment.

Comprehensive statistical toolkit: Offers over 220 statistical tools, including descriptive statistics, regression, ANOVA, multivariate analysis, forecasting, and machine learning—all in one add-on.

User-friendly and no coding required: Designed for ease of use, XLSTAT features an intuitive interface and requires no programming knowledge, making advanced analyses accessible to all skill levels.

Efficient data visualization: Provides customizable charts and advanced data visualization options directly in Excel to enhance interpretation and reporting.

Fast setup and quick learning curve: Easy installation and extensive tutorials, documentation, and customer support help users get started quickly and work efficiently.

Wide applicability: Suitable for business analysts, researchers, students, and professionals across diverse industries who need reliable, in-depth data analysis.

Affordable and scalable: Offers a cost-effective solution with options tailored for different needs and organization sizes, capable of handling small to large datasets.

Frequent updates and strong community support: Regular feature enhancements and a global user base ensure ongoing improvements and helpful resources

Some Key Features of XLSTAT over Other Statistical Tools

Descriptive Statistics: Provides tools to summarize, visualize, and describe datasets through measures such as mean, median, variance, and distribution charts.

Hypothesis Testing: Enables users to perform a wide range of parametric and non-parametric tests to assess statistical hypotheses on their data.

ANOVA/MANOVA: The software supports Analysis of Variance (ANOVA) and Multivariate Analysis of Variance (MANOVA), allowing for comparison of means across multiple groups or variables.

Multivariate Analysis: Offers techniques like principal component analysis and cluster analysis to explore patterns and relationships among multiple variables simultaneously.Time Series Analysis: The software includes methods for decomposing, forecasting, and analyzing temporal trends and seasonality in time-sequenced data.

Machine Learning: Integrates algorithms for supervised and unsupervised machine learning, such as regression, classification, and clustering, enabling data-driven predictions and modelling.

Variants: XLSTAT offers 8 tailored solutions to suit diverse analysis needs: Basic, Basic+, Sensory, Life Sciences, Marketing, Forecasting, Quality, and Premium.

Case Study: XLSTAT for Customer Satisfaction Analysis

A retail chain surveyed 120 customers on five service dimensions: Product Quality, Price Fairness, Store Cleanliness, Staff Friendliness, and Checkout Speed—plus an Overall Satisfaction score (1–10 scale). The aim was to identify the strongest satisfaction drivers

Method: Data were entered into Excel and analyzed with XLSTAT

1. Correlation Matrix visualized as a heatmap to assess relationships (Figure 1).
2. Multiple Linear Regression (standardized betas) to determine predictors (Figure 2).
3. Scatter Plot with Regression Line to show the strongest bivariate relationship (Figure 3).

Figure 1. The correlation matrix presents the pairwise relationships among the study variables. In this dataset, the highest correlation is observed between Staff Friendliness and Overall Satisfaction, suggesting that improvements in staff friendliness are strongly associated with increased overall satisfaction. Other variables show weaker or moderate correlations, indicating that while they may contribute to satisfaction, their influence is less direct compared to staff friendliness.

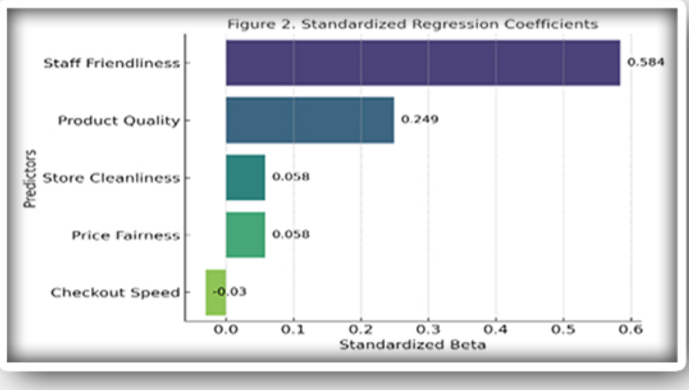


Figure 2 reveals that staff friendliness (β = 0.584) is by far the most influential factor driving satisfaction, followed by product quality (β = 0.249). Store cleanliness and price fairness (both β = 0.058) have minimal positive effects, while checkout speed (β = −0.030) shows a negligible negative impact. This suggests that enhancing staff interactions and maintaining high product quality should be the primary focus for improving overall satisfaction.

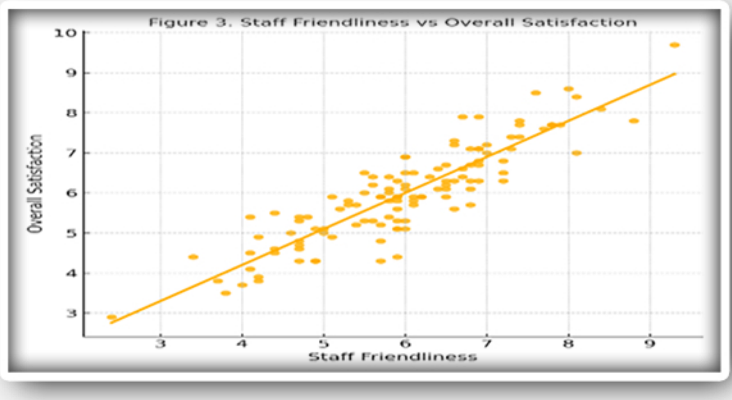


Figure 3. Scatter plot with fitted regression line illustrating the strongest bivariate relationship between Staff Friendliness and Overall Satisfaction. The strong positive correlation confirms Staff Friendliness as the most influential single predictor in the model.

Key Findings

- Staff Friendliness (β = 0.58) and Product Quality (β = 0.25) had the strongest impact.
- Checkout Speed showed negligible influence (β = −0.03).
- Model fit: $R^2 = 0.88$, no multicollinearity issues (all VIF < 2).

Conclusion

XLSTAT seamlessly integrates with Microsoft Excel, enabling easy data preparation, processing, and reporting through customizable charts and over 220 analytical tools. Its no-code interface, special discounts for students and academics, and robust customer support make it accessible and user-friendly. Serving industries from marketing and healthcare to manufacturing, finance, and academia, XLSTAT empowers users of all skill levels to transform raw data into actionable insights, streamlining workflows, enhancing decision-making, and offering a versatile, cost-effective alternative to complex standalone statistical software.

Article By

Dr. Poornima M. Charantimath, B E, FIE

Dr. Poornima M. Charantimath is a distinguished academician, consultant, and thought leader with over 30 years of experience in administration, teaching, research, and consultancy in engineering and management education. She serves as Professor of Eminence at KES Institute of Excellence in Management Science, Hubli, and is a Board of Studies Member at RVIMS, Bengaluru. Her leadership roles include Director at KLS IMER, Belagavi, and Deputy Director at the Indian Institute of Materials Management, Bangalore. A renowned corporate trainer and consultant, she has worked with organizations like Aditya Birla Group, Tata Motors, and HLL Lifecare Ltd. She has authored several acclaimed books, including Entrepreneurship Development and Small Business Enterprises and Total Quality Management, and has guided five Ph.D. scholars. Recognized with multiple national awards, including the Women Achievers Award and the Distinguished Engineer's Award, Dr. Charantimath continues to make significant contributions to entrepreneurship, quality management, and higher education.

Self-Hosted AI with Ollama

Why Self-Host?

Self-hosting means running applications, services, or websites on your own server infrastructure instead of using third-party hosting providers.

In today's cloud-driven world, most businesses and individuals rely on external providers for hosting. While cloud solutions offer convenience and scalability, they also bring recurring costs, limited customization, and data privacy concerns. Self-hosting is a powerful alternative that allows organizations and individuals to regain control of their digital infrastructure.

Self-hosting empowers you to take ownership of your digital assets. Though it requires effort and technical expertise, the benefits of data privacy, independence, and cost efficiency make it an attractive option. With proper setup and maintenance, self-hosting provides unmatched flexibility and security.

Key Benefits

Complete Control: Full authority over your applications, data, and infrastructure.

Enhanced Privacy & Security: Sensitive data remains on your servers, minimizing risks of third-party access.

Customization: Tailor software and configurations to your specific needs.

Independence from Third Parties: Avoid unexpected service changes, price hikes, or shutdowns.

Cost Efficiency (Long-Term): While initial setup costs exist, ongoing expenses are typically lower than cloud subscriptions.

Performance Optimization: Choose hardware suited to your applications for better performance.

Skill Development: Gain hands-on experience in networking, server administration, and command-line tools.

What is Ollama?

Ollama is a free, open-source tool that allows you to run large language models (LLMs) locally on your computer. It provides data privacy, security, and offline access to AI capabilities such as code assistance, writing support, and content generation.

By running Ollama locally, you maintain full data ownership and avoid risks tied to cloud storage. Offline execution also reduces latency and ensures reliability.

Ollama Features

Local & Private Execution: Models run fully on your machine; no data leaves your system.

Curated Model Library: Download and run variety of models (e.g., Llama 3, Mistral) with a simple 'ollama pull' command.

Easy Model Management: Use a Docker-like CLI to list, download, run, and remove models.

Customization: Fine-tune models with a 'Modelfile' to adjust behavior.

Cross-Platform Support: Works on macOS, Linux, and Windows.

API Integration: Offers an OpenAI-compatible REST API for application workflows.

Hardware Acceleration: Optimizes GPU/CPU usage (supports NVIDIA, Apple Silicon/Metal, AMD).

Multimodal Support: Run models like LLaVA that handle both text and images.

How Ollama Helps

Simplifies Setup: Manages dependencies like Docker, easing AI deployment.

Local Development & Testing: Build and test AI apps without cloud costs or latency.

Customization: Fine-tune models for specific needs via 'Modelfile'.

Framework Compatibility: Works with programming languages and frameworks such as LangChain.

Security & Compliance: Supports industries with strict data regulations (e.g., healthcare, finance).

Why Self-Host?

Personal Chatbot: Create a chatbot with a defined personality.

Offline Writing Assistant: Generate blog posts, marketing copy, or personal writing without internet

Interactive Q&A: Use 'ollama run llama3' for Q&A sessions from the command line.

Educational Exploration: Experiment with LLMs to learn NLP concepts.

Why Self-Host?

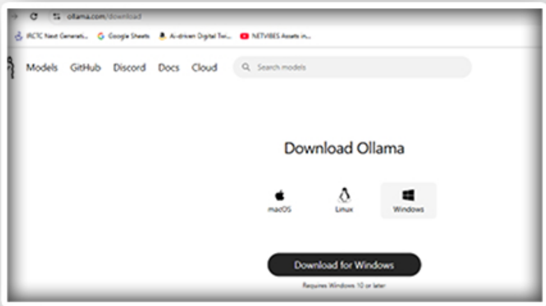
For most current and exact system requirement, please refer Ollama website.

Category	macOS	Windows	Linux
Operating System (OS)	macOS 13.0 (Ventura) or later	Windows 10/11 64-bit	Ubuntu 20.04+ or other modern distros
CPU	Apple Silicon (M1 or later recommended) or Intel CPU	Intel/AMD 64-bit processor	Intel/AMD 64-bit processor
RAM	Minimum 8 GB	Minimum 8 GB	Minimum 8 GB
Storage	At least 10 GB free disk space	At least 10 GB free disk space	At least 10 GB free disk space
Network	Internet required for setup and downloads	Internet required for setup and downloads	Internet required for setup and downloads

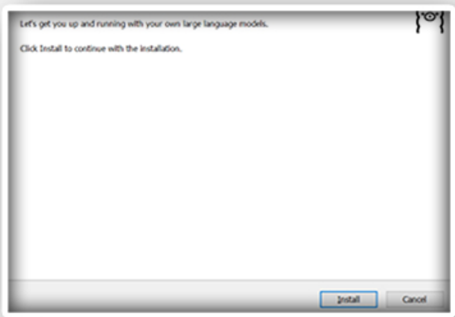
How to Install and Configure Ollama

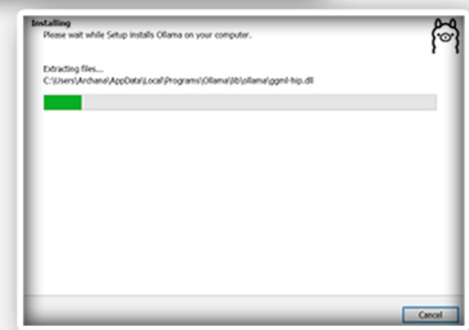
Ollama supports Linux, macOS, and Windows. Below is the process for Windows operating system:

1. Download the Ollama installer from the official site : https://ollama.com/download

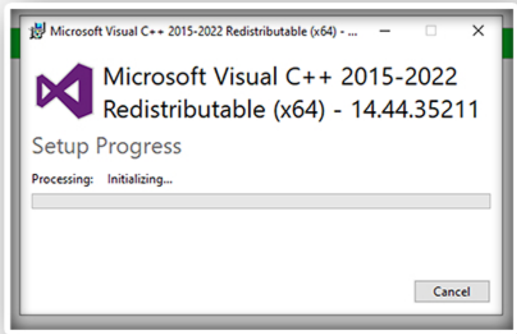


2. Run the setup file from your local drive.
3. Click Install and wait for extraction.





4. Allow Microsoft Visual C++ to download and install automatically (if required).



5. Let the installer complete installation — Ollama will launch automatically.

Note: For macOS or Linux, follow the official installation guide on the Ollama website.

Conclusion:

Popular AI models and tools available today are mostly subscription-based and can become expensive beyond the free tier—especially for organizations with a larger number of users, or those just beginning their initial and experimental foray into AI implementation. In comparison, self-hosted AI provides an inexpensive alternative to jump-start the AI implementation process.

----X----

Article By
Madhusudhan
Predictive Analytics Solutions Pvt Ltd

Linear programming in Python using **PuLP**

Linear programming (LP) is a method used to solve problems where you want to find the best outcome (like maximizing profit or minimizing cost) while following certain rules (linear equations and inequalities). It's used in many fields, such as economics, manufacturing, transportation, and management, to make decisions that involve resources, costs, or production processes.

The general process involves the following steps:

- Initializing the model
- Setting up the objective function
- Defining the decision variables
- Specifying the constraints

and then solving the problem.

To solve linear programming problems in Python, PuLP library can be used. PuLP helps you model optimization problems (such as linear or integer programming) but doesn't directly solve them. Instead, it relies on external solvers to find the solution. This library is a part of the COIN-OR project.

A solver is the tool that performs the actual optimization. It takes the model you've created — including the decision variables, objective function, and constraints — and computes the best solution.

This simplifies the process of solving optimization problems by letting you focus on modelling while relying on the right solver to handle the computations.

Example

A company makes two products, A and B. The company wants to make as much profit as possible but has limited resources.

- Product A sells for Rs 700 and takes 4 hours of labour and 6 units of raw material to make.
- Product B sells for Rs 500 and takes 5 hours of labour and 2 units of raw material to make.

The company has:

- 200 hours of labour available each week.
- 150 units of raw material available each week.

How many units of Product A and Product B should the company make to get the highest profit?

Let's explore how to solve the above optimization problem using Python and the PuLP library

1. Install the PuLP Library

First, you need to install the PuLP library, which is used to solve linear programming problems.

```
pip install PuLP
```

2. Import the Library

Once installed, you can import PuLP into your Python script. You can use a short alias (p) instead of writing pulp every time.

```
import pulp as p
```

3. Create the Optimization Problem

Here, we're creating a maximization problem. This means we want to maximize the value of the objective function (e.g., profit).

```
prob = p.LpProblem("Maximization_Problem", p.LpMaximize)
```

- "Maximization_Problem" is the name of the problem.

  p.LpMaximize indicates that we're maximizing the objective function.

4. Define Decision Variables

The decision variables are the quantities we want to optimize. In this case, x1 is the number of units of Product A, and x2 is the number of units of Product B. We also specify that these are non-negative integers

```
x1 = p.LpVariable('x1', lowBound=0, cat='Integer') # Product A
x2 = p.LpVariable('x2', lowBound=0, cat='Integer') # Product B
```

- lowBound=0 ensures that both variables are non-negative (you can't produce a negative number of products).

  cat='Integer' ensures that the variables can only take integer values (no fractional products).

5. Set Up the Objective Function

Now we define the objective function. The goal is to maximize the profit, which is given by: 700x1 + 500 x2

This means:

- Each unit of Product A gives 700 units of profit.
- Each unit of Product B gives 500 units of profit.

```
prob += 700 * x1 + 500 * x2 # Maximizing profit
```

This expression is added to the problem, prob, using +=

6. Add Constraints

Next, we add the constraints to the problem. These are the limits we have to work within:

Labor constraint: Each unit of Product A requires 4 hours of labour, and each unit of Product B requires 5 hours. We can't exceed 200 total labour hours.

```
prob += 4 * x1 + 5 * x2 <= 200 # Labour hours constraint
```

Raw material constraint: Each unit of Product A requires 6 units of raw material, and each unit of Product B requires 2 units. We can't exceed 150 total units of raw material.

```
prob += 6 * x1 + 2 * x2 <= 150 # Raw material constraint
```

These constraints are also added to the problem using +=

7. Display the Problem

This line prints out the problem with all the objective function and constraints. It's useful for checking your setup.

```
print(prob)
```

```
Maximization_Problem:
MAXIMIZE
700*x1 + 500*x2 + 0
SUBJECT TO
_C1: 4 x1 + 5 x2 <= 200

_C2: 6 x1 + 2 x2 <= 150

VARIABLES
0 <= x1 Integer
0 <= x2 Integer
```

8. Solve the Problem

Now we tell PuLP to solve the problem using the solve() method.

```
status = prob.solve()
```

This will find the optimal values for x1 and x2 that maximize the objective function while satisfying the constraints.

9. Display the Results

Finally, we display the results:

- The status of the solution (whether it was successful or not).
- The optimal number of units of Product A and Product B to produce.
- The maximum profit we can achieve.

```
print(f"Status: {p.LpStatus[status]}”) # Status of the solution
print(f"Optimal number of Product A to produce: {p.value(x1)}”)
print(f"Optimal number of Product B to produce: {p.value(x2)}”)
print(f"Maximum Profit: Rs {p.value(prob.objective)}”)
```

```
Status: Optimal
Optimal number of Product A to produce: 16.0
Optimal number of Product B to produce: 27.0
Maximum Profit: Rs 24700.0
```

Summary:

1. Install and import the PuLP library.
2. Create a maximization problem using p.LpProblem().
3. Define decision variables that represent the number of products to produce.
4. Set up the objective function (maximize profit).

   Add constraints for labour and raw materials.

6. Display the problem to verify setup.
7. Solve the problem using the solve() method.
8. Display the results, including the optimal quantities and maximum profit.

Article By

Manjunatha M.C

Predictive Analytics Solutions Pvt Ltd

----X----