

# Machine Learning for Smarter Innovation

## Week 1: Foundations & Clustering

Discovering Innovation Patterns with ML

BSc Course in AI-Enhanced Innovation

# Prerequisites & What You Need

Setting You Up for Success

## What You Need to Know

- Basic Python (variables, loops, functions)
- High school math (averages, distances)
- How to use Jupyter notebooks
- Basic data concepts (tables, rows, columns)

## What We'll Provide

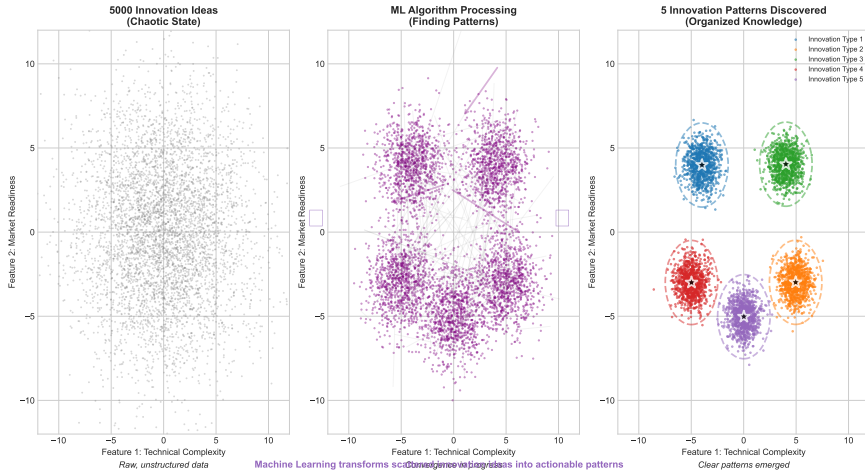
- All code templates
- Step-by-step instructions
- Visual explanations
- Practice datasets

**No machine learning experience required!**

# Machine Learning + Innovation + Design Thinking

The Power of Convergent Methodologies

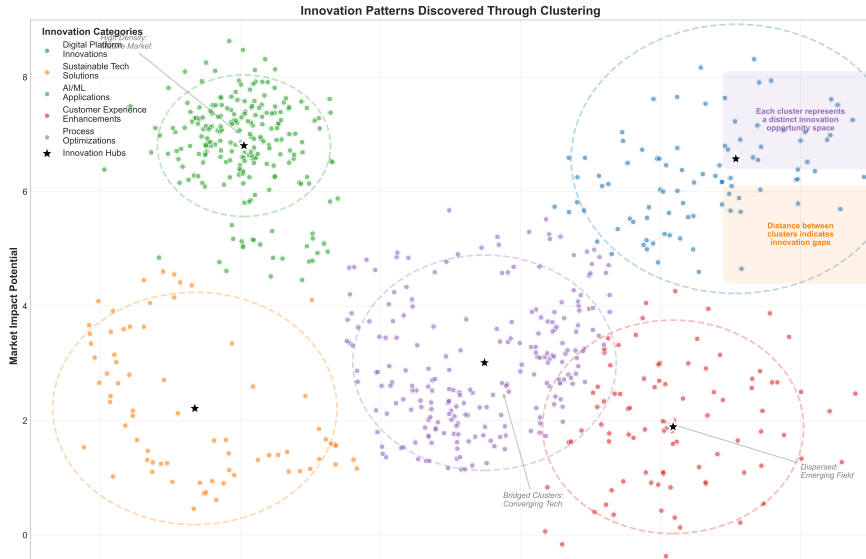
## The Convergence Flow: From Chaos to Clarity



## Where Data Science Meets Human Creativity

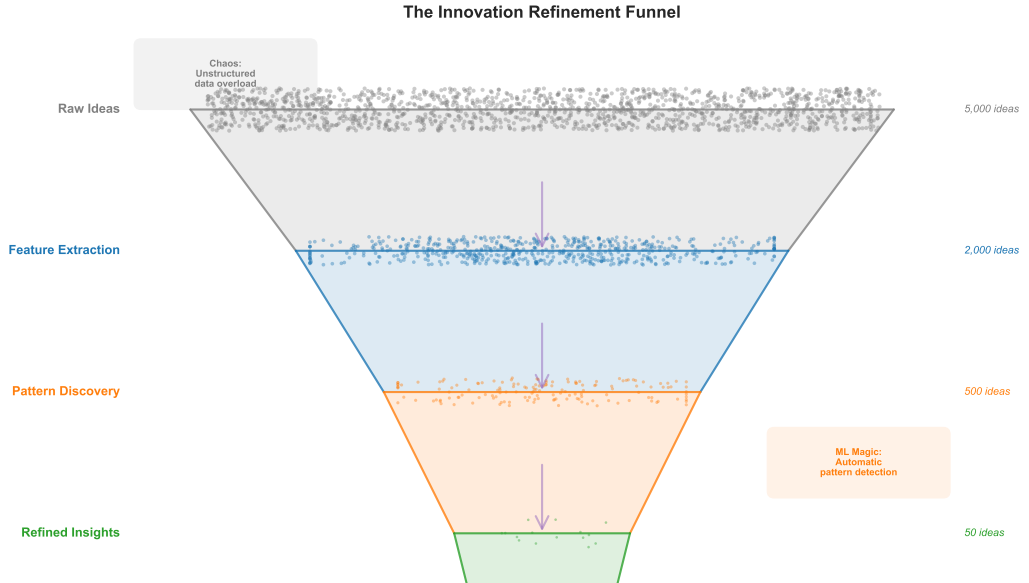
# From Data Points to Innovation Insights

Bridging the Technical-Human Gap



# The Innovation Refinement Funnel

From Chaos to Clarity Through Feature Analysis



## PART 1

### Foundation & Context

What we'll explore:

- Why traditional design hits limits
- How ML amplifies human insight
- The dual pipeline approach
- Your learning journey ahead

Setting the stage for transformation

# Part 1: Learning Objectives

What You'll Learn in This Section

By the end of Part 1, you will be able to:

- **Understand** the limitations of traditional innovation approaches
- **Recognize** how ML enhances human creativity
- **Explain** the dual pipeline methodology
- **Navigate** the 10-week learning journey
- **Identify** Week 1's role in the overall course

Success Criteria

- Can articulate 3+ traditional design limitations
- Can describe ML's value proposition
- Can map ML pipeline to design pipeline
- Understand clustering's role in innovation

# PART 1

## Foundation & Context

### Understanding the Innovation Challenge

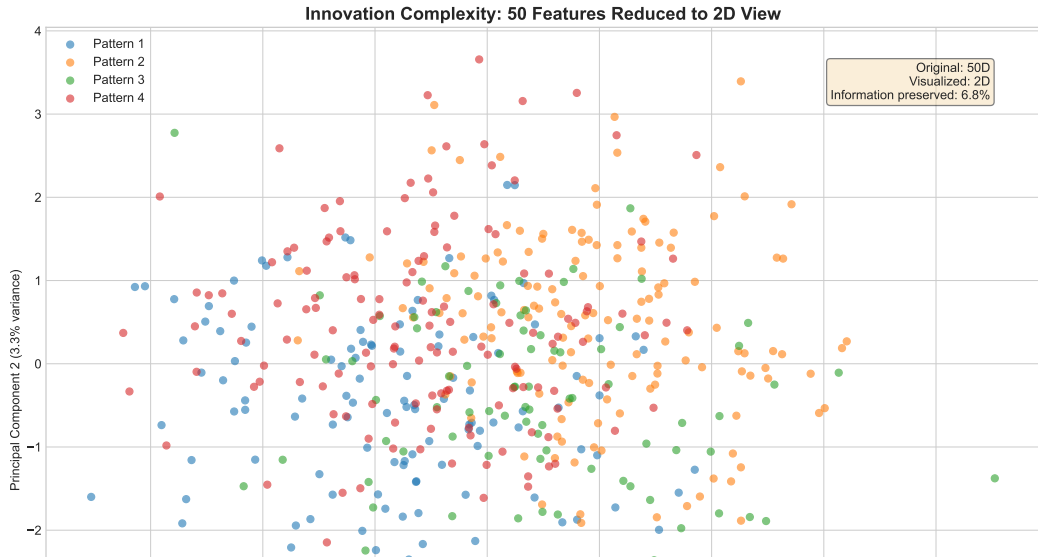


## Finding Patterns in the Chaos



# The Hidden Complexity

Each Innovation Depends on Hundreds of Features



# The Innovation Challenge

Why Traditional Design Needs AI Enhancement

## Traditional Design Limits

- **Scale:** Can analyze 50 ideas, not 50,000
- **Speed:** Months for insights
- **Bias:** Designer's perspective dominates
- **Patterns:** Miss hidden connections
- **Iteration:** Slow feedback loops

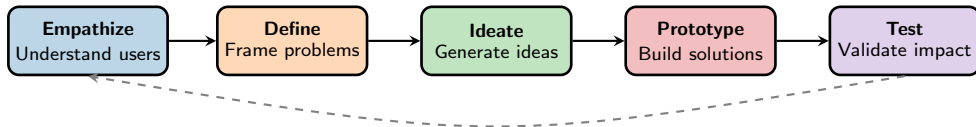
## AI-Enhanced Innovation

- **Scale:** Analyze millions of data points
- **Speed:** Real-time insights
- **Objectivity:** Data-driven discovery
- **Patterns:** Find non-obvious relationships
- **Iteration:** Continuous learning

**The Promise: 100x more insights, 10x faster innovation**

# Quick Recap: The Design Thinking Process

You've Seen This Before - Let's Connect It to ML



Iteration is key

## Traditional Approach

- Manual interviews
- Limited sample size
- Qualitative insights
- Slow iteration

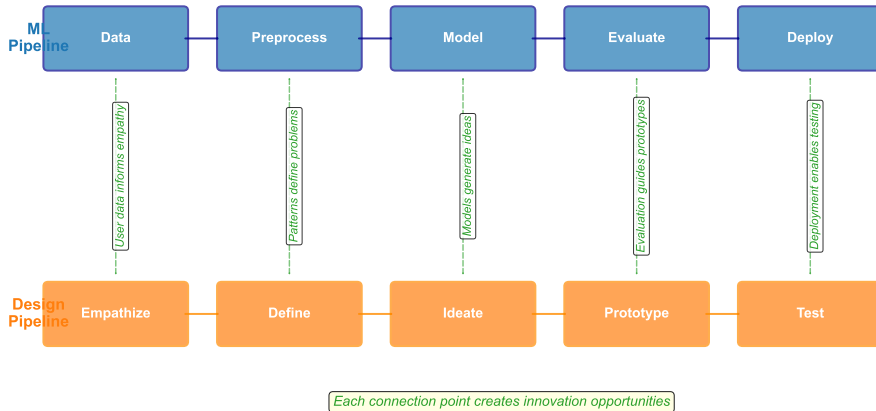
## ML-Enhanced Approach

- Data-driven discovery
- Massive scale analysis
- Quantitative patterns
- Real-time adaptation

# The Dual Pipeline

Where ML Meets Design Thinking

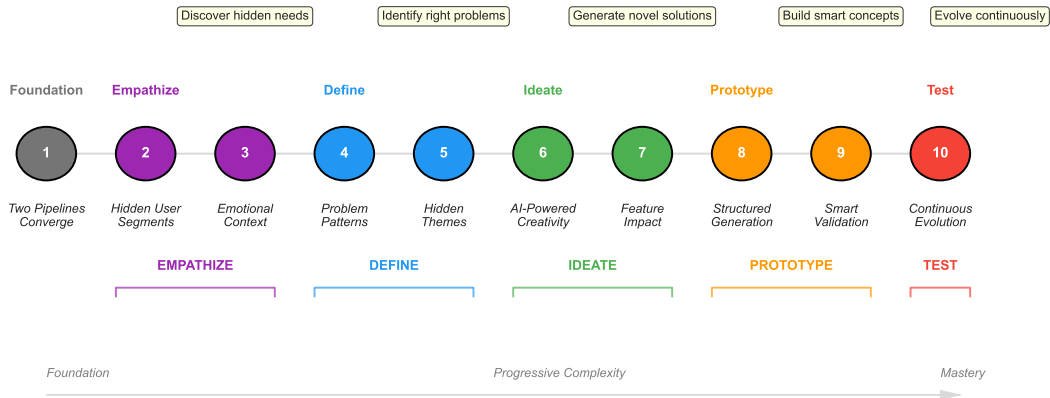
## The Convergence: ML Meets Design Thinking



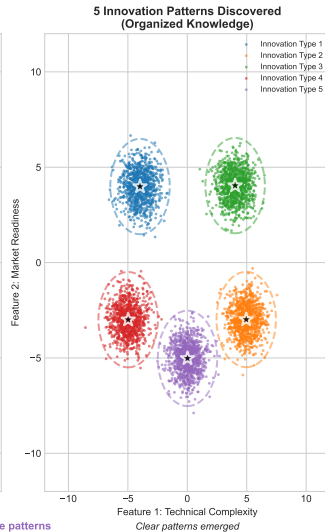
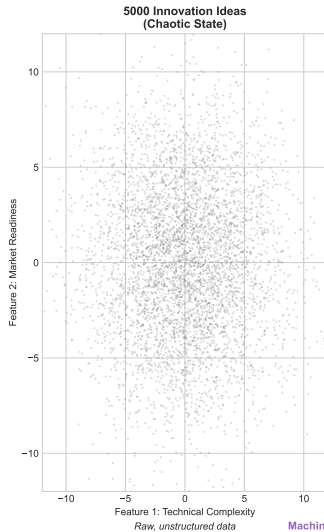
# Your Innovation Journey

10 Weeks to Understanding AI-Powered Design

## 10-Week Innovation Journey



## The Convergence Flow: From Chaos to Clarity



**The Convergence Flow: Order from Chaos**  
*Watch 5000 innovation ideas self-organize into meaningful patterns*

# PART 2

## Technical Core

### Machine Learning Algorithms & Implementation



# The Innovation Classification Problem

5000 Ideas - How Do They Connect?

## The Pain

### Current Reality:

- One-size-fits-all solutions
- Generic innovation categories
- Missed opportunities
- Unhappy edge cases

### The Cost:

- Most innovations get misclassified
- Features with low adoption rates
- Inefficient resource allocation

## The Question

### What if we could...

- Find natural innovation clusters?
- Discover innovation patterns?
- Innovate at scale?
- Identify opportunity gaps?

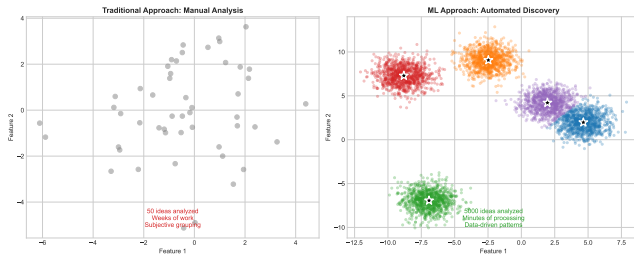
**We can!**

**Solution: Clustering**

# Current Reality: The Problem

Why One-Size-Fits-All Doesn't Work

The Current Reality: Scale Challenge in Innovation



## The Fragmentation Crisis

- **70%** of innovations fail due to misalignment
- **85%** miss their target audience
- **60%** duplicate existing solutions

## Why This Happens:

- Treating all ideas the same
- Missing subtle patterns
- No systematic categorization
- Human cognitive limits

**ML clustering reveals the hidden structure in innovation chaos**

# Discovery Exercise: Which Archetype?

Match Each Innovation to Its Type

## Innovation Examples:

- ① **Uber** - Connecting drivers with riders via app
- ② **Tesla Model 3** - Affordable electric vehicle
- ③ **Amazon Prime** - Fast delivery subscription
- ④ **iPhone Camera** - Annual improvements
- ⑤ **ChatGPT** - AI conversation interface

**Think:** What makes each one similar or different?

## Match to Type:

- A. Disruptive Innovation
- B. Incremental Innovation
- C. Platform Innovation
- D. Service Innovation
- E. Business Model Innovation

## Answers:

*(Discuss with neighbor first)*

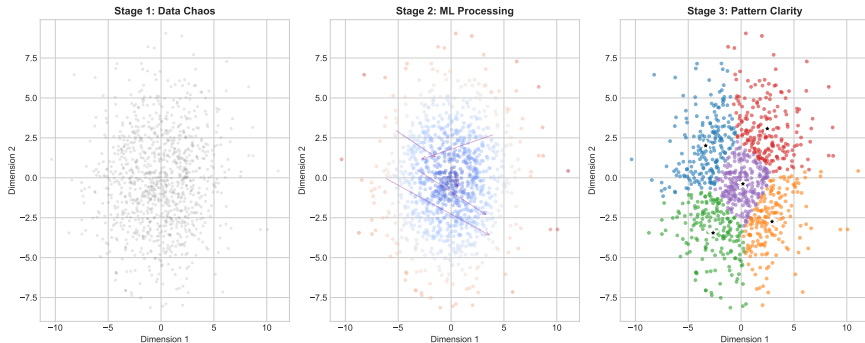
1→C (Platform), 2→A (Disruptive),  
3→E (Business Model), 4→B (Incremental),  
5→D (Service)

**ML can do this matching at scale - for thousands of innovations**

# What is Clustering?

Like Organizing a Messy Room - Finding Things That Belong Together

## From Chaos to Clarity: The ML Journey



## Clustering Finds:

- Natural groupings
- Similar approaches
- Hidden patterns
- Innovation relationships

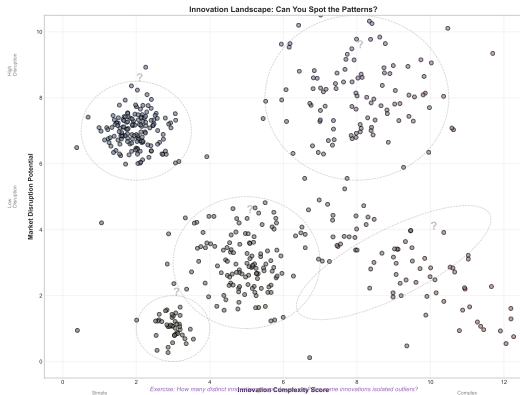
### Key Insight:

Things that look similar often belong in the same group  
*(Just like organizing books by topic on a shelf)*

# Discovery: How Many Groups Do You See?

Visual Pattern Recognition Exercise

Look at this innovation data:



## Your Observations:

- 1 Groups you see: \_\_\_\_\_
- 2 Main pattern: \_\_\_\_\_
- 3 Outliers: \_\_\_\_\_

### Key Insight:

Humans are good at 2D patterns.  
But innovation has 100+ dimensions!  
That's where ML helps.

## Questions:

- How many distinct groups?

# Discovery: What Makes Things Similar?

Understanding Features That Matter

## For Innovations, What Features Matter?

Innovation	Cost	Impact	Time
Smart Thermostat	Low	Medium	Quick
Electric Car	High	High	Long
Mobile App	Low	Low	Quick
Solar Panels	High	High	Long
AI Chatbot	Medium	Medium	Medium

### Which innovations group together?

- By cost? (Low vs High)
- By impact? (Low vs High)
- By timeline? (Quick vs Long)
- All combined?

### Discovery Exercise

#### Group these by similarity:

- 1 Smart Thermostat + ?
- 2 Electric Car + ?
- 3 Mobile App + ?

### The Challenge:

Real innovations have 100+ features!

- Market size - Technology readiness - Regulatory requirements - User demographics - Competition level - And many more...

# Discovery: Manual Clustering Exercise

Try Clustering Yourself - Then See How ML Does It

## Your Task:

Group these 12 innovations into 3 clusters:

- 1 Blockchain payment system
- 2 Voice-activated assistant
- 3 Renewable energy storage
- 4 Social media platform
- 5 Autonomous vehicle
- 6 Health tracking wearable
- 7 Cloud computing service
- 8 3D printing technology
- 9 Virtual reality training
- 10 Drone delivery system
- 11 Gene editing tool
- 12 Quantum computing

## Your Groups:

Group 1: \_\_\_\_\_

Group 2: \_\_\_\_\_

Group 3: \_\_\_\_\_

## Think About:

- What features did you consider?
- How did you decide on groups?
- Was it difficult to classify some items?
- Did any items fit multiple groups?

Let's see how ML approaches this...

# Discovery: How ML Clusters These Innovations

Based on 50+ Hidden Features

## Digital Platforms

4. Social media platform
7. Cloud computing service
2. Voice-activated assistant
1. Blockchain payment system

## Physical Innovation

5. Autonomous vehicle
10. Drone delivery system
3. Renewable energy storage
8. 3D printing technology

## Frontier Tech

11. Gene editing tool
12. Quantum computing
9. Virtual reality training
6. Health tracking wearable

### ML considers features you might not think of:

- Development complexity
- Market readiness level
- Infrastructure requirements
- Regulatory complexity
- User behavior patterns
- Technology stack similarity
- Investment requirements
- Innovation lifecycle stage

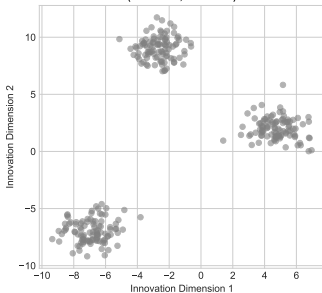


# K-Means: The Basic Clustering Method (Part 1)

Initial Setup - Like Choosing City Centers

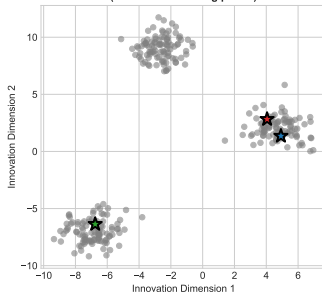
## K-Means Algorithm: Step-by-Step Innovation Clustering

Step 1: Raw Innovation Data  
(300 ideas, no labels)



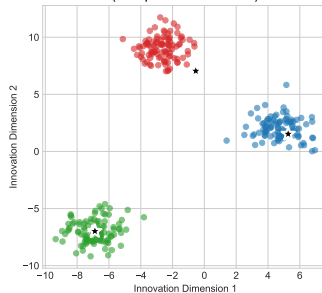
□ How many groups?

Step 2: Initialize Centers  
( $k=3$  random starting points)



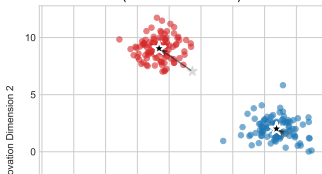
□ Start with  $k=3$  centers

Step 3: Assign to Nearest Center  
(Each point finds its cluster)

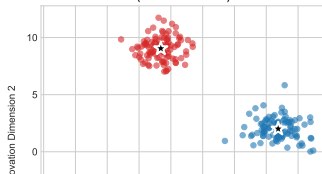


□ Distance-based assignment

Step 4: Update Centers  
(Move to cluster mean)

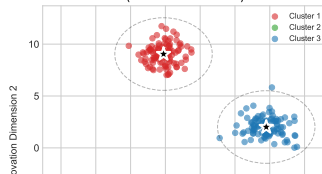


Step 5: Repeat Until Stable  
(After 5 iterations)



□ Distance-based assignment

Step 6: Convergence!  
(Stable clusters found)



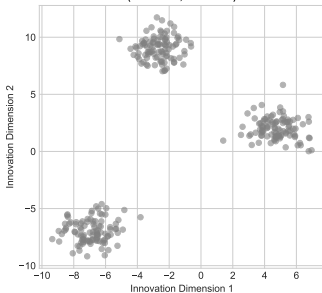
● Cluster 1  
● Cluster 2  
● Cluster 3

# K-Means: The Basic Clustering Method (Part 2)

Iteration Process - Finding Natural Groups

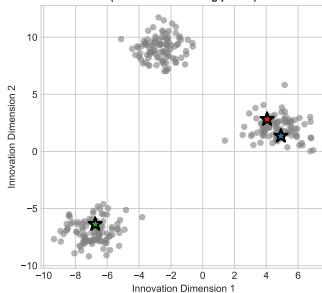
## K-Means Algorithm: Step-by-Step Innovation Clustering

**Step 1: Raw Innovation Data**  
(300 ideas, no labels)



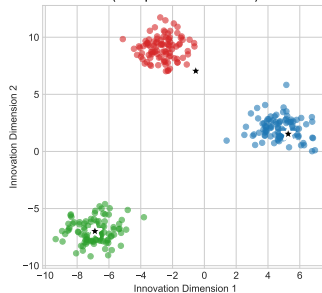
□ How many groups?

**Step 2: Initialize Centers**  
( $k=3$  random starting points)



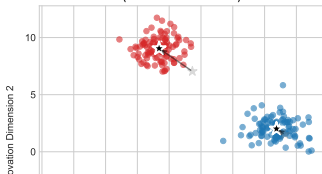
□ Start with  $k=3$  centers

**Step 3: Assign to Nearest Center**  
(Each point finds its cluster)

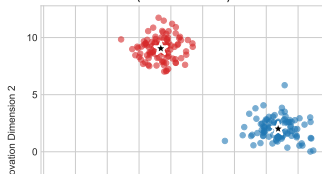


□ Distance-based assignment

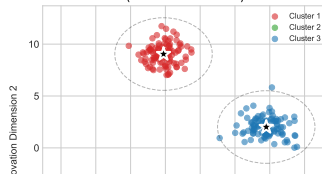
**Step 4: Update Centers**  
(Move to cluster mean)



**Step 5: Repeat Until Stable**  
(After 5 iterations)



**Step 6: Convergence!**  
(Stable clusters found)



● Cluster 1  
● Cluster 2  
● Cluster 3

# The Goldilocks Problem

Too Few vs. Too Many Groups

## Too Few ( $K=2$ )

### Oversimplification

- Mixed segments
- Lost nuance
- Generic solutions

## Just Right ( $K$ )

### Optimal Balance

- Clear segments
- Actionable insights
- Manageable complexity

## Too Many ( $K$ )

### Analysis Paralysis

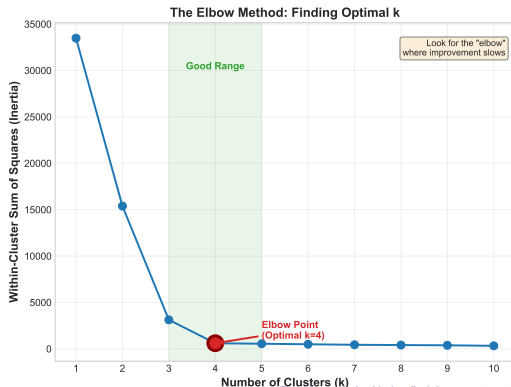
- Overfitting
- Tiny segments
- Impossible to act on

How do we find the sweet spot?

# The Elbow Method

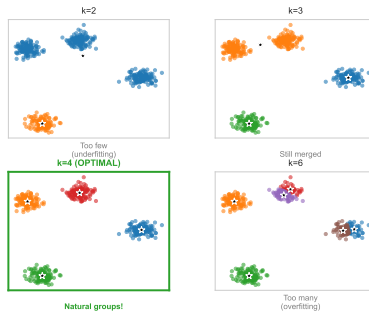
How Many Groups Should We Have? (Like Goldilocks - Not Too Few, Not Too Many)

## Choosing the Right Number of Innovation Clusters



The elbow method helps find the sweet spot between too few and too many clusters

## Visual Comparison: Different k Values



## Finding the Elbow:

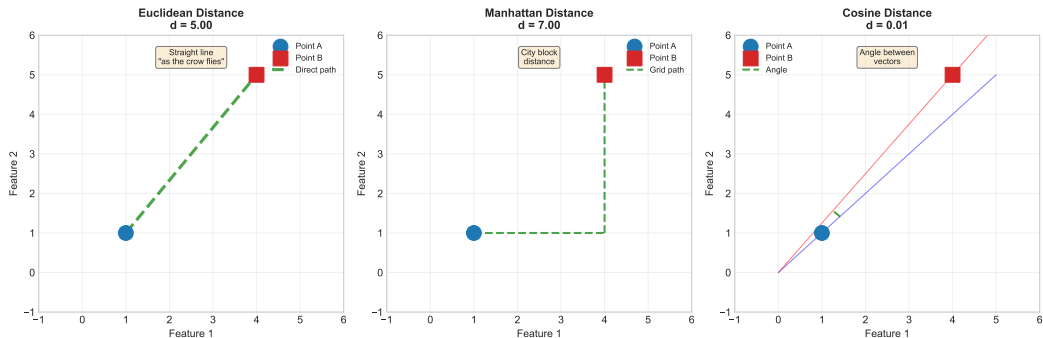
- Plot inertia vs K
- Look for the "elbow"

**Optimal K = 5**  
Best trade-off point

# Distance Metrics

Different Ways to Measure "How Close" Things Are

## Distance Metrics: Different Ways to Measure Innovation Similarity

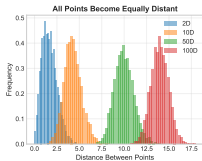
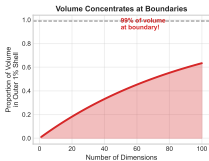


Each metric reveals different patterns in your data

# Sidestep: The Curse of Dimensionality

Why High-Dimensional Spaces Are Strange and Empty

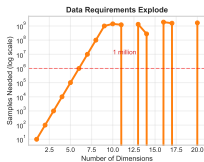
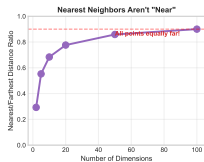
## The Curse of Dimensionality: Why High-Dimensional Spaces Are Strange



In High Dimensions:  
Everything is on the Surface



More than atoms  
in universe!



Innovation data has 100+ dimensions - that's why we need specialized ML algorithms!

## The Paradox

In 100 dimensions:

- 99.99% of space is empty
- All points are outliers
- Nearest neighbors aren't near

## Why This Matters

**Innovation has 100+ features!**

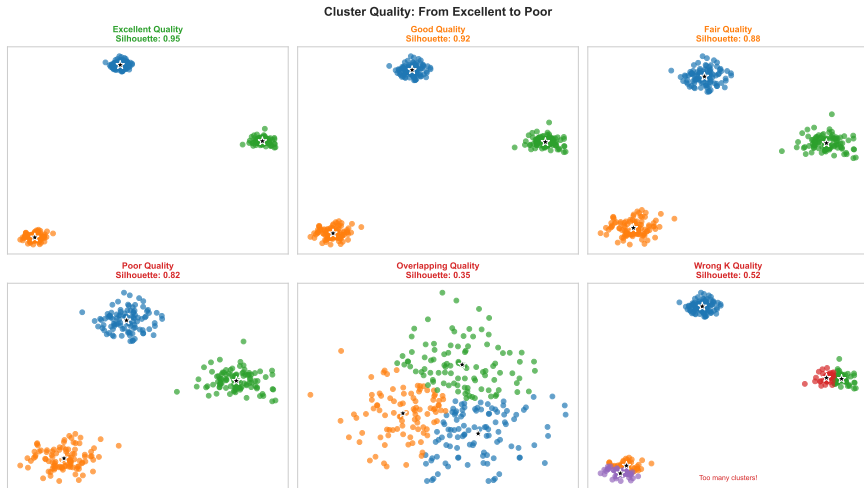
- Distance metrics break down
- Need special techniques
- Dimensionality reduction crucial
- That's why we use PCA/t-SNE

As dimensions increase:

- Points become equally distant

# Cluster Quality Metrics

Are Our Groups Any Good? (Like Checking Your Work)



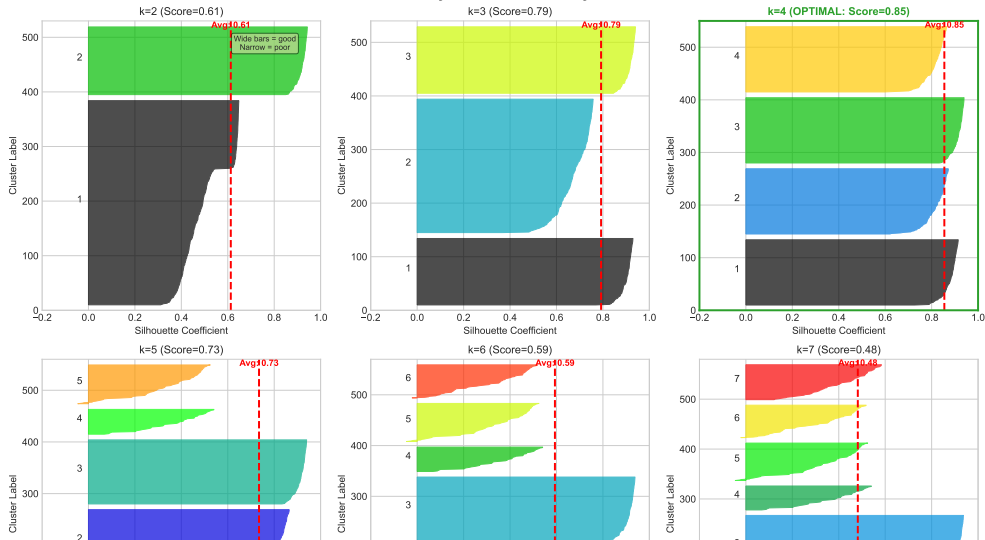
**Silhouette Score:**

**What it measures:**

# Evaluation Metric 1: Silhouette Score

Measuring Cluster Cohesion and Separation

## Silhouette Analysis: Cluster Quality Validation





# Discovery: Finding the Right K

What Happens With Different Numbers of Clusters?

## Experiment with K:

K	What Happens
K=2	Everything too mixed
K=4	Natural groups emerge
K=8	Some groups split unnecessarily
K=20	Too fragmented to use

### Your Turn:

If you have 100 customer types, what K would you choose?

- K=100? (one per type)
- K=5? (major groups)
- K=20? (detailed segments)

## The Trade-offs:

### Too Few (Under-fit)

- Mixed segments
- Lost insights
- Generic solutions

### Just Right

- Clear segments
- Actionable groups
- Meaningful patterns

### Too Many (Over-fit)

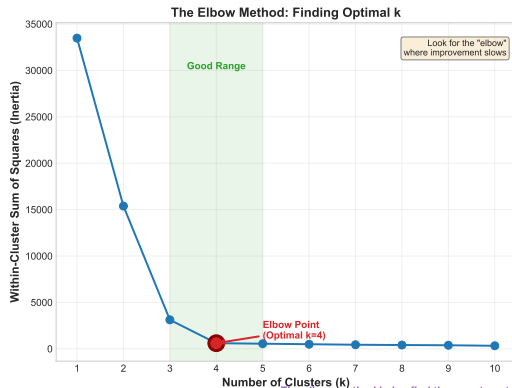
- Fragmented insights
- Hard to implement
- Statistical noise

The Elbow Method helps find the sweet spot automatically

# Evaluation Metric 2: Elbow Method

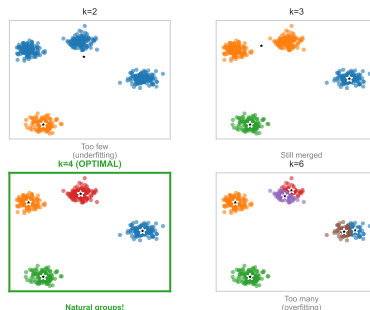
Finding the Right Number of Clusters

## Choosing the Right Number of Innovation Clusters



The elbow method helps find the sweet spot between too few and too many clusters

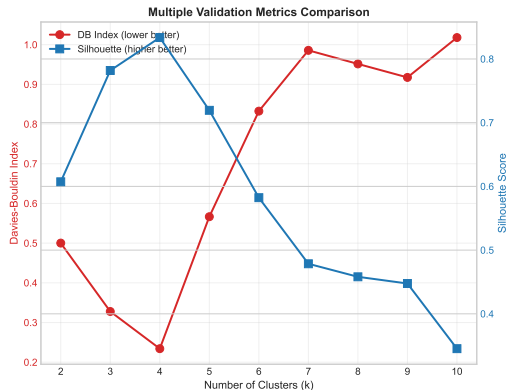
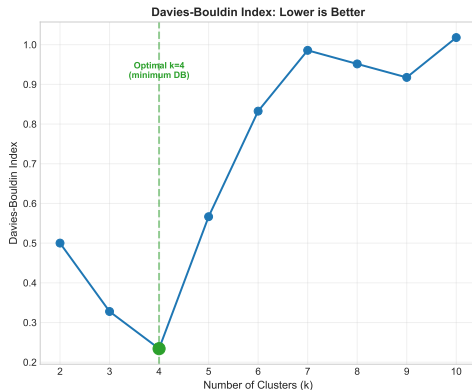
## Visual Comparison: Different k Values



# Evaluation Metric 3: Davies-Bouldin Index

Balancing Within and Between Cluster Distances

Cluster Validation: Davies-Bouldin Index



## K-Means Assumes Spherical Clusters

But what about:

- Innovations connected through technology stacks
- Domain-specific innovation clusters
- Evolution patterns (incremental, disruptive)
- Outliers and noise points

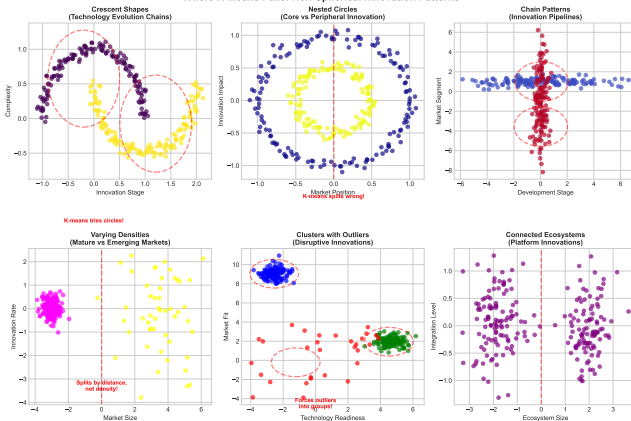
**K-Means Forces Round Pegs into Round Holes**

**Solution: Density-Based Clustering**

# Discovery: When K-Means Fails

Can You Spot Why K-Means Won't Work Here?

Where K-Means Fails: Non-Spherical Innovation Patterns



Real innovation patterns rarely form perfect circles - that's why we need advanced clustering methods! Breaks connections!

## K-Means Problems

### K-Means assumes:

- Spherical (round) clusters
- Similar sizes
- Similar densities
- No outliers

## Real Innovation Patterns

### But innovations have:

- Evolution chains
- Technology ecosystems
- Varying market sizes
- Disruptive outliers

## Exercise:

Draw clusters on the left image.  
Where does K-means fail?

Look at these patterns:

- Crescent shapes

# DBSCAN: Finding Groups Naturally

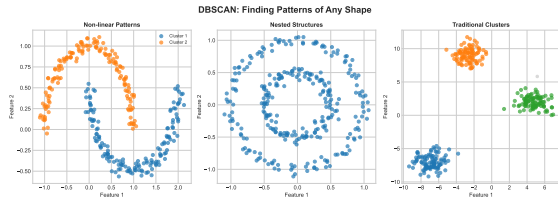
Like Finding Groups of People at a Party - Where Are the Crowds?

## DBSCAN Advantages:

- No need to specify K (*finds groups automatically*)
- Finds arbitrary shapes (*not just circles*)
- Identifies outliers (*points that don't belong*)
- Handles noise well (*robust to random points*)

### Perfect for:

- Non-spherical patterns
- Varying densities
- Outlier detection
- Exploratory analysis



# DBSCAN: Understanding Parameters

Two Simple Settings Control Everything

## Epsilon (Distance)

### What it does:

Sets the maximum distance to consider points as neighbors

### Think of it as:

How far can points be apart and still be friends?

**Too small:** Many tiny clusters

**Too large:** Everything merges

## MinPts (Density)

### What it does:

Minimum neighbors needed to form a dense region

### Think of it as:

How many friends make a group?

**Too small:** Noise becomes clusters

**Too large:** Small clusters vanish

**Rule of thumb:**  $\text{MinPts} = 2 \times \text{dimensions}$

# Clustering Algorithm Comparison

Technical Characteristics at a Glance

Algorithm	Speed	Shape	Outliers	Params	Best For
K-Means	Fast $O(nkt)$	Spherical clusters	Sensitive	K only	Quick segments
DBSCAN	Medium $O(n \log n)$	Any shape	Robust (detects)	eps, MinPts	Complex shapes
Hierarchical	Slow $O(n^2)$	Any shape	Moderate	Distance threshold	Multi-level analysis
GMM	Medium $O(nkt)$	Elliptical clusters	Moderate	K, covariance	Overlapping groups

Each algorithm has its strengths - choose wisely!



# When to Use Each Algorithm

## Practical Decision Guide

### K-Means

#### Perfect when:

- Speed is critical
- Clusters are roughly equal size
- You know K in advance
- Data has spherical patterns

### Hierarchical

#### Perfect when:

- Need multiple granularities
- Want to visualize relationships
- Small to medium datasets
- Exploring data structure

### DBSCAN

#### Perfect when:

- Clusters have irregular shapes
- Outliers need identification
- Density varies across data
- You don't know K

### GMM

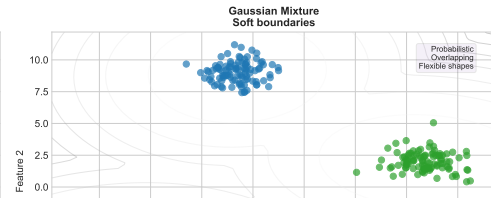
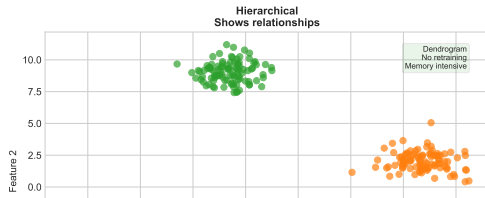
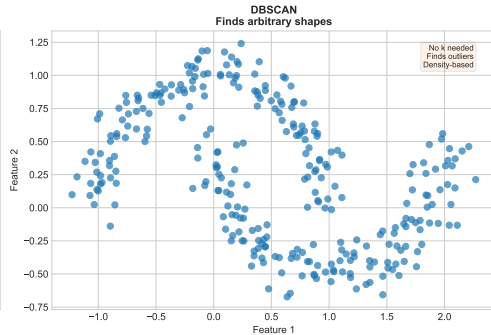
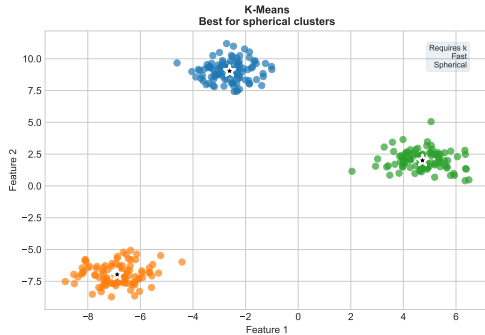
#### Perfect when:

- Groups overlap
- Need probability scores
- Elliptical cluster shapes
- Soft assignments needed

# Algorithm Visual Comparison

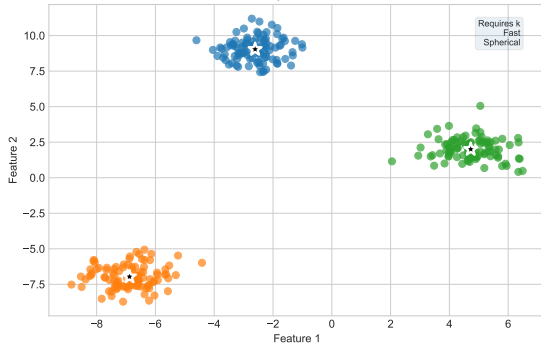
Same Data, Different Approaches

## Clustering Algorithm Comparison

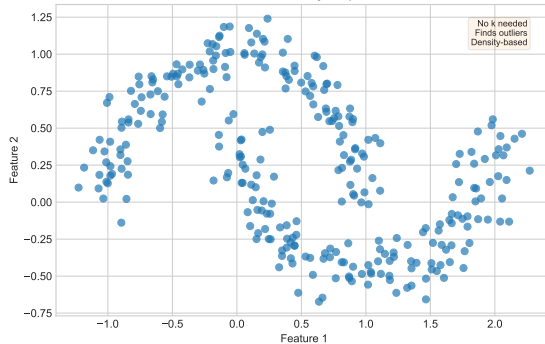


## Clustering Algorithm Comparison

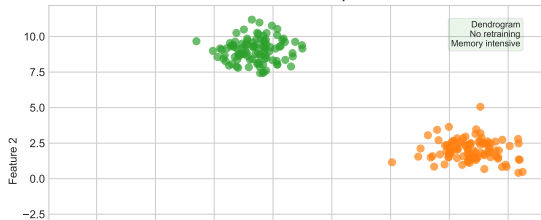
**K-Means**  
Best for spherical clusters



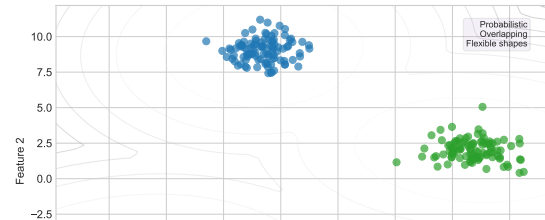
**DBSCAN**  
Finds arbitrary shapes



**Hierarchical**  
Shows relationships



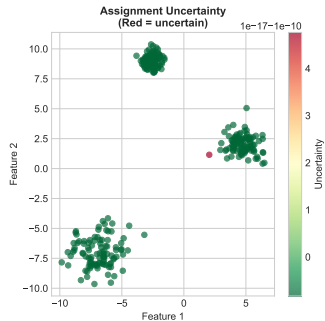
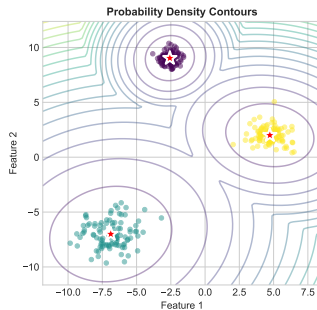
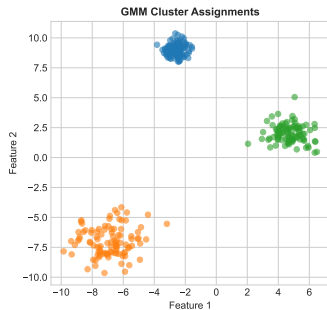
**Gaussian Mixture**  
Soft boundaries



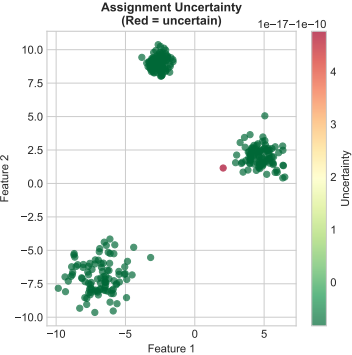
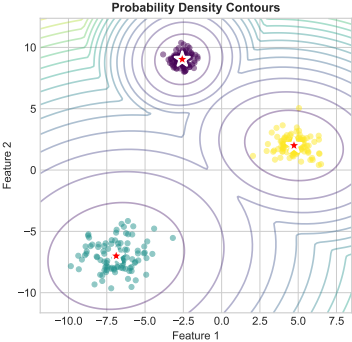
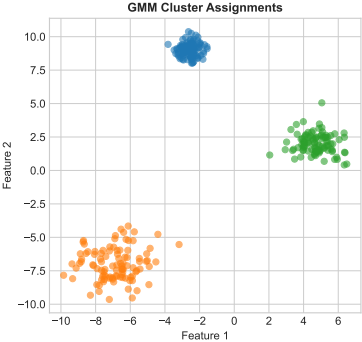
# Gaussian Mixture Models (GMM)

Soft Clustering for Overlapping Innovation Categories

## Gaussian Mixture Model: Probabilistic Clustering



Gaussian Mixture Model: Probabilistic Clustering



## Fixed K Gives One View

But real relationships are hierarchical:

- Organization: Company → Department → Team → Individual
- Geography: Country → Region → City → Neighborhood
- Products: Category → Subcategory → Brand → SKU
- Innovations: All → Categories → Sub-types → Specific solutions

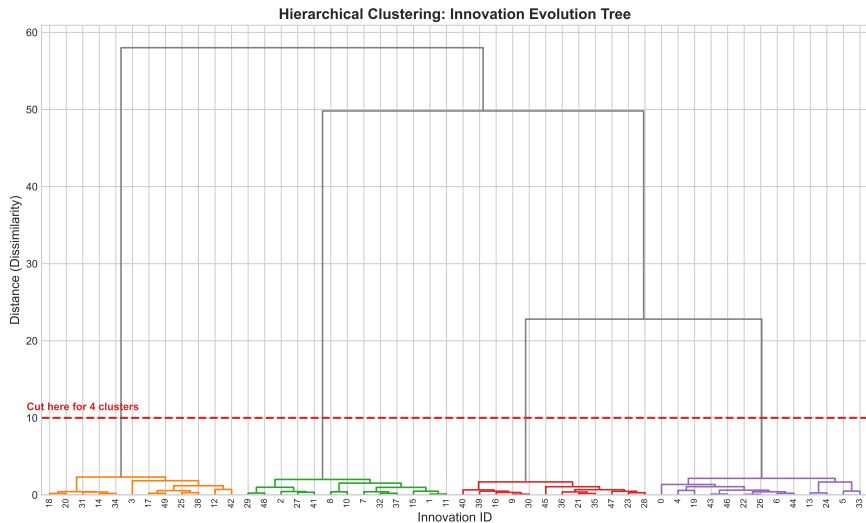
**K-means: Pick 5 groups and that's it**

**What if we need flexibility?**

Solution: See the full hierarchy, cut where needed

# Hierarchical Clustering

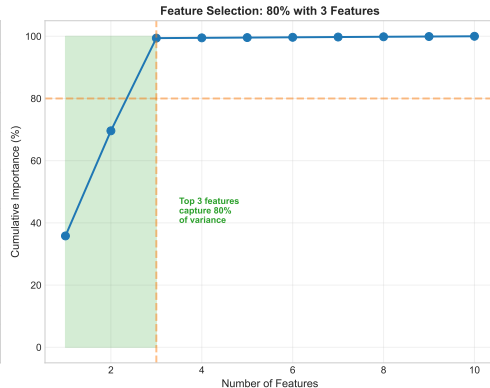
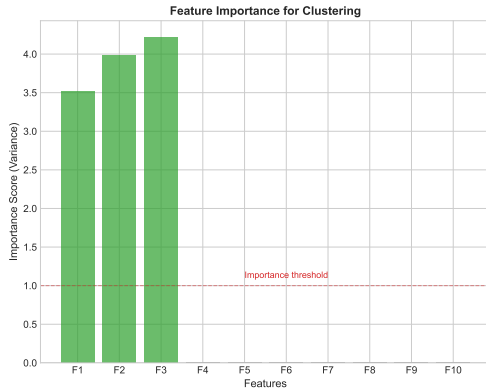
Building a Tree of Relationships



# What Drives the Clusters?

## Feature Importance Analysis

### Feature Importance Analysis



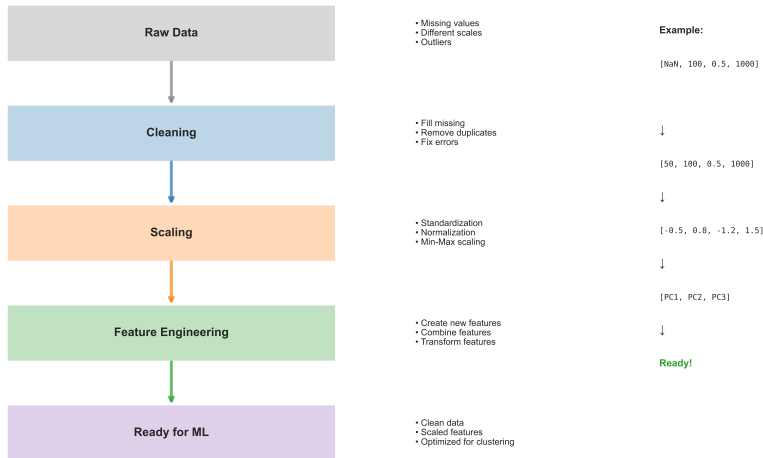
**Key Insight: Usage frequency matters most!**



# Data Preprocessing Pipeline

From Raw Data to Clustering-Ready Features

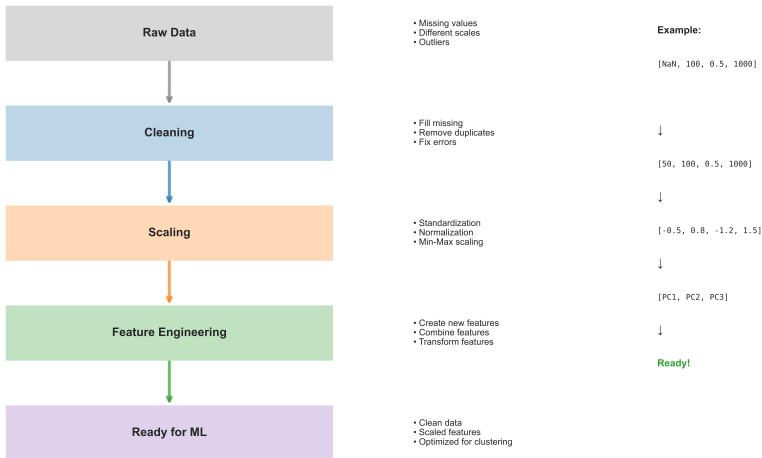
## Data Preprocessing Pipeline for Clustering



# Data Preprocessing Pipeline - Example

Real Innovation Data Transformation

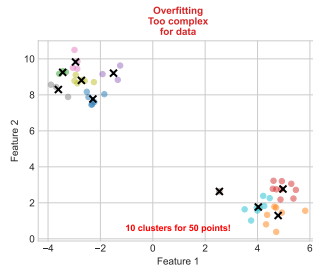
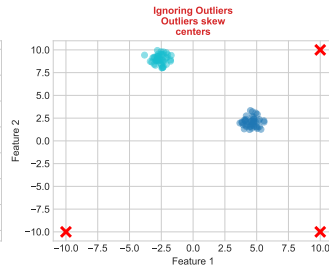
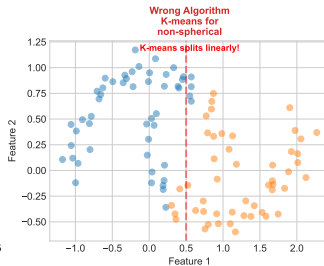
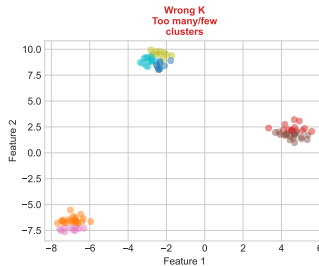
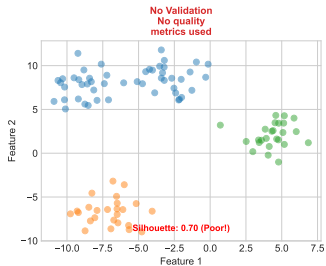
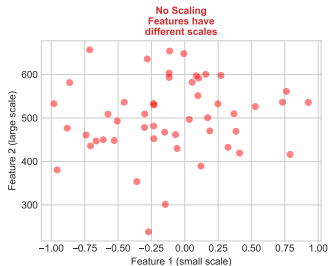
## Data Preprocessing Pipeline for Clustering



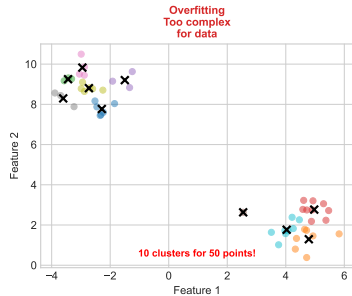
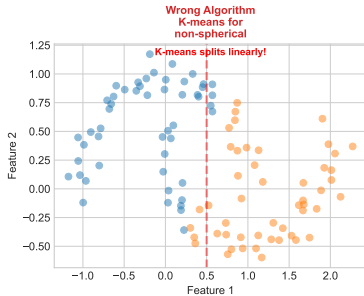
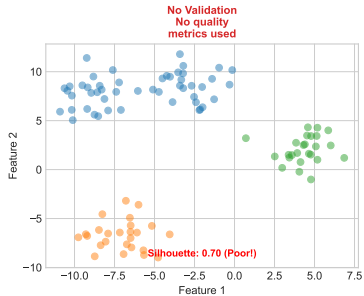
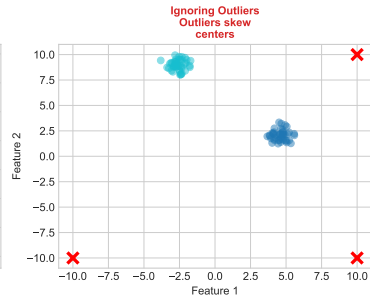
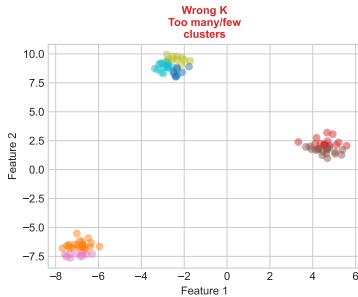
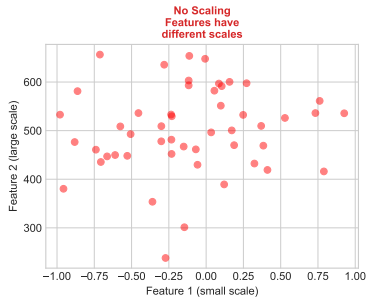
# Common Mistakes & Troubleshooting

Learn from These Pitfalls

## Common Clustering Mistakes to Avoid



## Common Clustering Mistakes to Avoid

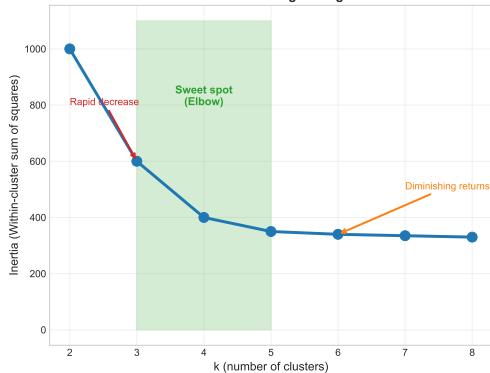


# Parameter Tuning Guidelines (Part 1)

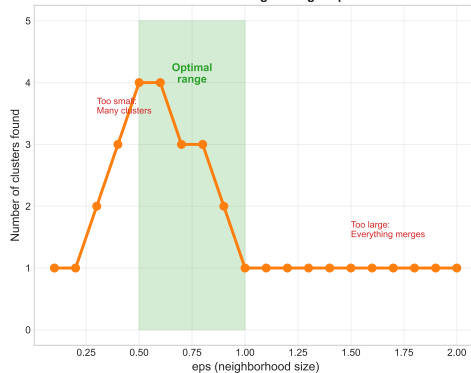
## K-Means and DBSCAN Parameters

### Parameter Tuning Guidelines - Part 1: Distance-Based Methods

K-Means: Choosing the Right k



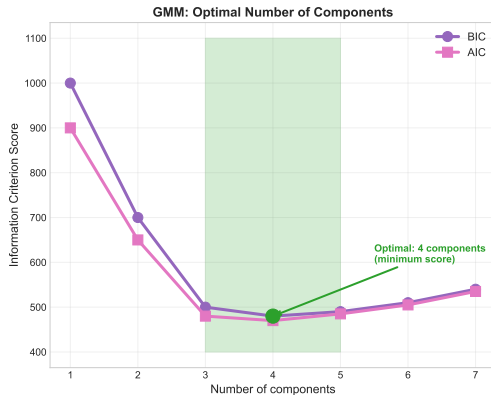
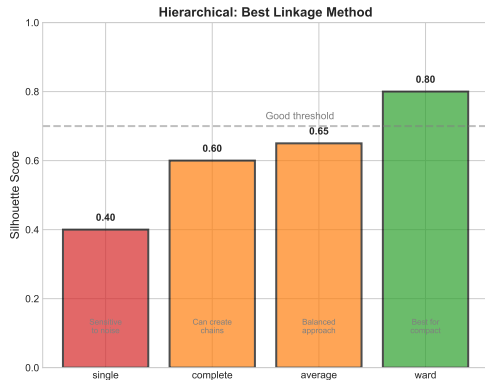
DBSCAN: Finding the Right eps



# Parameter Tuning Guidelines (Part 2)

## Hierarchical and GMM Parameters

### Parameter Tuning Guidelines - Part 2: Advanced Methods



## Quick Quiz

① K in K-means stands for:

- ☐ Kernel
- ☒ Number of clusters
- ☐ Constant

② DBSCAN finds:

- ☐ Only circles
- ☒ Any shape clusters
- ☐ Exactly K groups

## Can You Calculate?

If Silhouette Score = 0.75:

- Is this good? **Yes!**
- Range is  $[-1, 1]$
- Higher = better separation

**Remember:**

- Elbow method finds optimal K
- Scale your data first!

**Great job! Now let's apply these concepts!**

*Next: Design integration, innovation patterns, and real-world applications*

**We've learned the technical tools:**

Clustering, metrics, quality measures

**But clusters are just numbers...**

Until we connect them to innovation opportunities

**Let's transform data into innovation insights**

Each cluster represents innovation opportunities and patterns



## PART 3

### Innovation Pattern Analysis

What we'll create:

- Data-driven innovation archetypes
- Innovation pattern maps per category
- Cluster-specific journeys
- Opportunity heat maps
- Design priority matrices

Where ML reveals innovation patterns

# Part 3: Learning Objectives

Innovation Applications You'll Explore

By the end of Part 3, you will be able to:

- **Create** innovation archetypes
- **Map** innovation patterns
- **Design** opportunity matrices
- **Analyze** innovation lifecycles
- **Build** ecosystem maps
- **Prioritize** innovation efforts

Design Outcomes

- Innovation taxonomy framework
- Cluster-based strategies
- Data-driven prioritization
- Opportunity identification
- Pattern recognition skills
- Ecosystem understanding

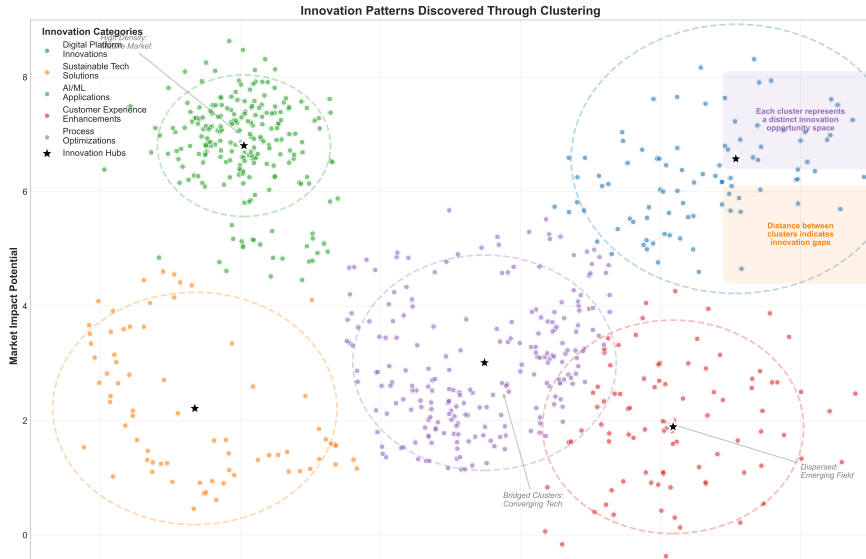
# PART 3

## Design Integration

Bridging Technology & Human Experience

# From Data Points to Innovation Insights

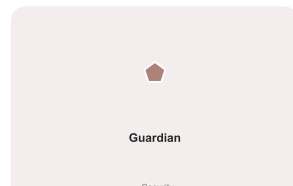
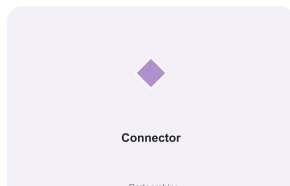
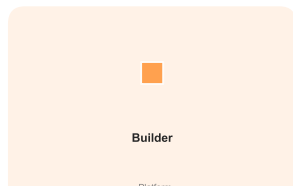
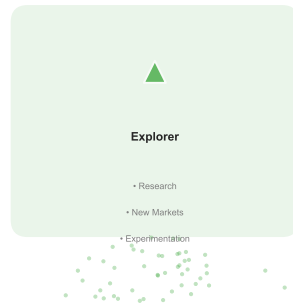
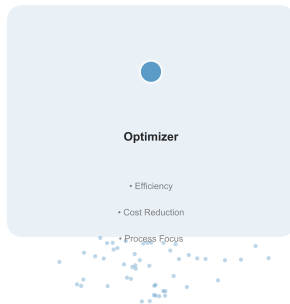
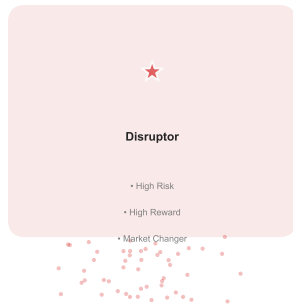
Bridging the Technical-Human Gap



# AI-Generated Innovation Archetypes

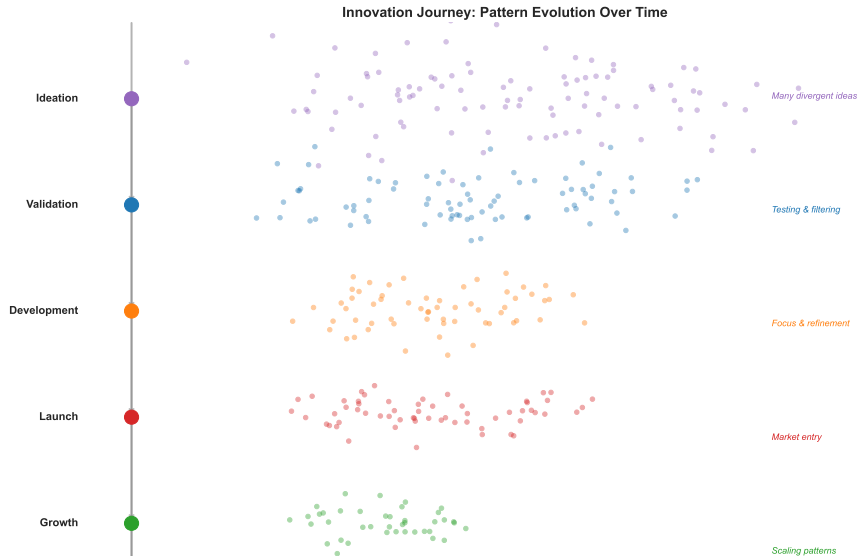
Data-Driven Character Development

## Innovation Archetypes: 6 Distinct Patterns



# Innovation Pattern Maps

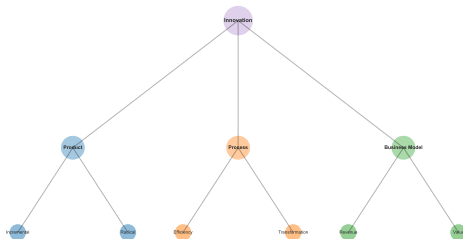
Cluster-Specific Insights



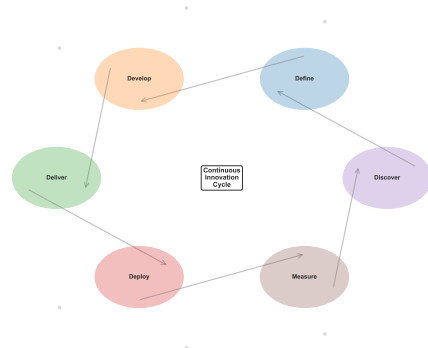
# Innovation Framework

## Taxonomy and Lifecycle Stages

Innovation Taxonomy: Hierarchical Classification



Innovation Lifecycle: Continuous Improvement Process



### Framework Levels:

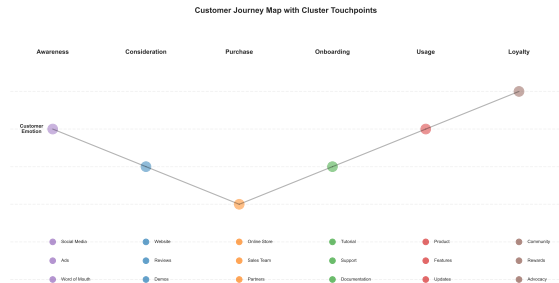
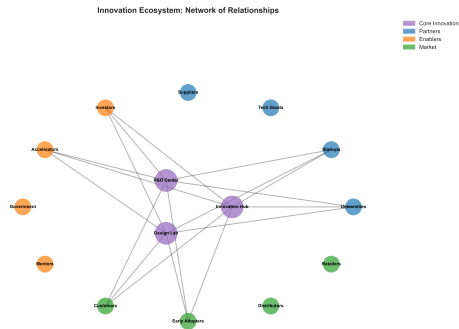
- Types & relationships
- Impact measurements
- Strategic positioning

### Lifecycle Stages:

- Ideation & discovery
- Development & testing
- Launch & scaling

# Innovation Ecosystem & Journey Mapping

From Networks to Evolution Paths



## Ecosystem Elements:

- Network connections
- Stakeholder clusters
- Value flows

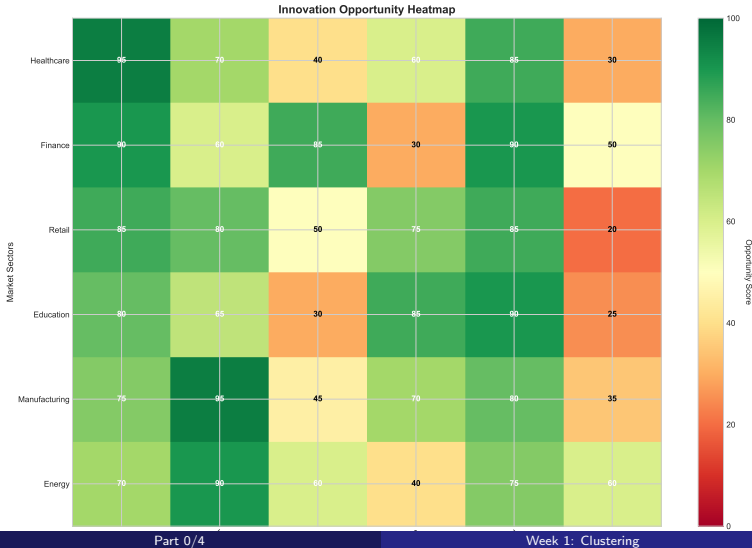
## Evolution Paths:

- Different speeds
- Varying trajectories
- Unique milestones



# Innovation Opportunities by Cluster

Where Each Category Has Potential



## Key Findings:

- Emerging tech: Early stage
- Disruptive: Scalability
- Incremental: Integration
- Platform-based: Network effects

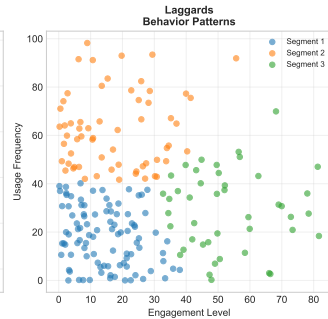
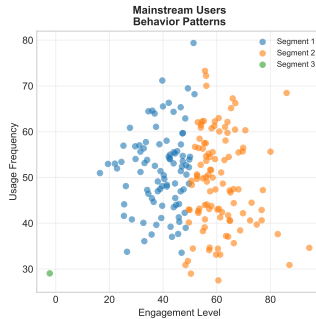
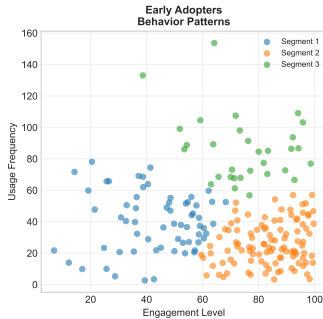
## Design implication:

One solution won't fit all!

# Innovation Patterns Revealed

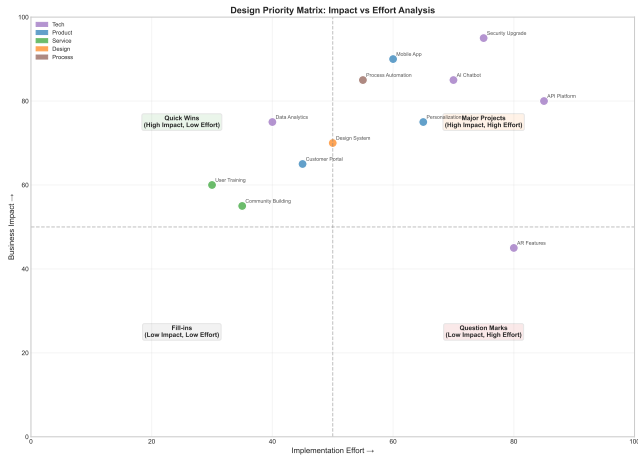
What Clusters Tell Us About Evolution

## User Behavior Pattern Clustering



# Design Priority Matrix

Where to Focus Your Efforts

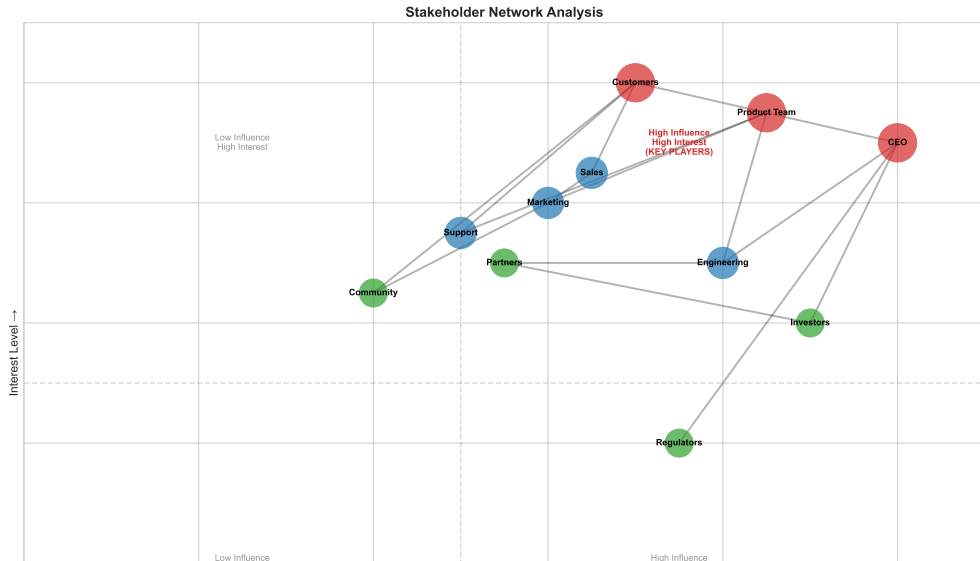


## Priority Quadrants:

- **High Impact + High Effort**  
Strategic initiatives
- **High Impact + Low Effort**  
Quick wins
- **Low Impact + Low Effort**  
Fill-ins
- **Low Impact + High Effort**  
Avoid

# Understanding Innovation Ecosystems

## Network Analysis of Innovation Connections



## Match the Application

Match algorithm to use case:

- ① Customer segmentation → **K-means**
- ② Finding outliers → **DBSCAN**
- ③ Creating taxonomy → **Hierarchical**
- ④ Overlapping groups → **GMM**

## Design Thinking

How does clustering help in:

- **Empathize**: Find user groups
- **Define**: Identify patterns
- **Ideate**: Discover opportunities
- **Prototype**: Target solutions
- **Test**: Validate segments

**Excellent! Ready to practice with real data?**

*Next: Summary, real-world case studies, and hands-on practice exercise*

## **You've learned:**

- The clustering algorithms
- How to validate quality
  - Design applications

## **Now let's see it in action**

How these techniques work in practice  
to find patterns in data