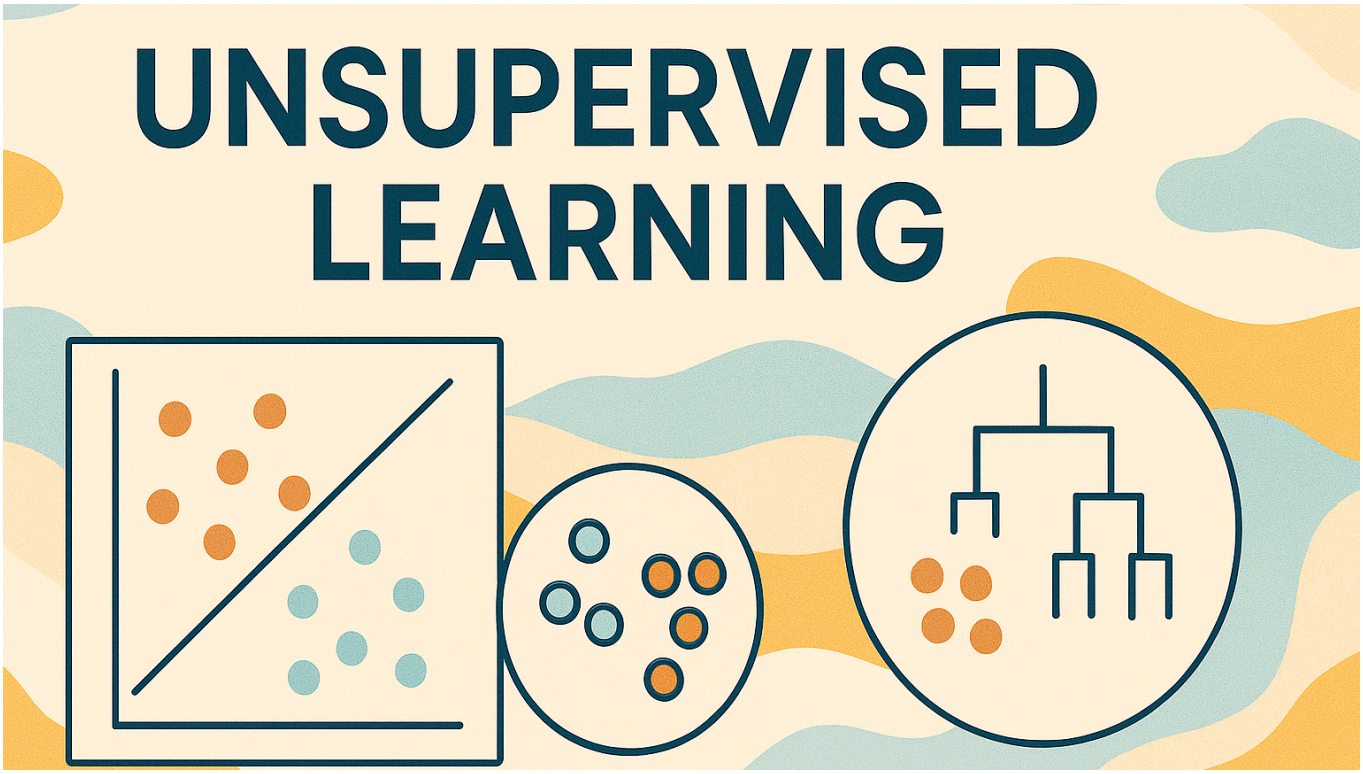
```
from IPython.display import Image
Image("Header.png")
```



Statistical Machine Learning Practical Homework - 4

Discovering Patterns in Global Cancer Data Using Unsupervised Learning

Team Members

Fariha Shah

Pavan Arram

Naveen Pasupula

Finding Patterns in Global Cancer Data Using Unsupervised Learning

Cancer impacts people across the globe, and every patient's experience is shaped by a variety of risk factors and treatment decisions. In this project, we explored a dataset of over 50,000 cancer patients using **unsupervised learning**. Without relying on predefined labels, our goal was to uncover patterns and groupings within the data using methods like **Principal Component Analysis (PCA)**, **KMeans Clustering**, and **Hierarchical Clustering**.

The Dataset

We worked with a global cancer dataset collected between **2015 and 2024**. Each row represents a patient, with a variety of features including:

- **Demographics**: Age, Gender, Country
- **Lifestyle & Risk Factors**: Genetic Risk, Smoking, Alcohol Use, Obesity
- **Medical Details**: Cancer Type, Stage, Treatment Cost, and Survival Years

To aid in analysis, we also created a new binary column called `Survival_Zero`, which flags patients with 0 survival years.

Preparing the Data

Before jumping into analysis, we cleaned and transformed the dataset:

- Verified there were no missing or empty values
- Encoded categorical features using label encoding
- Standardized all numeric columns so they were on the same scale
- Added `Survival_Zero` as a flag for exploratory insight

This setup allowed us to apply dimensionality reduction and clustering effectively.

Theoretical Background

To uncover structure in a large, unlabeled medical dataset, we turned to unsupervised learning techniques. These methods don't rely on predefined categories — instead, they help us reduce complexity, detect patterns, and segment similar patients based on underlying traits.

Principal Component Analysis (PCA)

PCA is a dimensionality reduction technique that transforms high-dimensional data into a smaller set of principal components. These components are new axes that capture as much variance (information) as possible from the original data, while being uncorrelated with one another.

Why use PCA?

Our dataset had dozens of variables — some were correlated or noisy. PCA helps distill this into a few components that summarize the key trends, which improves both interpretability and the stability of clustering.

How it works mathematically:

PCA computes an eigen-decomposition of the covariance matrix. It projects the data onto eigenvectors (called "loadings"), ordered by how much variance each explains. The U matrix from the SVD represents

the transformed observations (patients), while the V* matrix gives the loadings (how original features contribute to each component).

K-Means Clustering

K-Means is a centroid-based clustering algorithm. It divides data into k groups by assigning each point to the cluster with the nearest mean (centroid), then recalculates the centroids iteratively to minimize within-cluster variance.

Why use K-Means?

It's efficient and works well with PCA-reduced data. By grouping patients based on their reduced dimensions, we can form interpretable clusters and compare their profiles.

Limitations:

K-Means assumes spherical clusters and can struggle if clusters vary in size or density — which is why we also explored hierarchical clustering for robustness.

Hierarchical Clustering

Hierarchical clustering builds a tree (dendrogram) of nested clusters using a bottom-up approach. Each point starts in its own cluster, and pairs are merged step-by-step based on a linkage rule (e.g., Ward's method).

Why use Hierarchical Clustering?

It doesn't require specifying k up front and gives us a full view of how clusters form at different granularity levels. This is particularly helpful in medical data where categories may be ambiguous.

Key Feature:

We used Ward's method with k-nearest-neighbor connectivity to handle computational efficiency and ensure clusters are compact and distinct.

Methodology

Our approach to uncovering hidden patterns in global cancer data followed a structured, multi-step process. Since we were working with real-world health data, we prioritized careful cleaning and thoughtful transformation before applying any unsupervised learning techniques.

Step 1: Understanding and Cleaning the Data

We began by carefully examining the dataset, which included over 50,000 patient records from around the world, spanning the years 2015 to 2024. It covered everything from age, gender, and country to risk factors like genetic history, smoking, alcohol use, and obesity, along with medical indicators like cancer stage, treatment cost, and survival years.

To ensure high-quality input for modeling: • We checked for missing, NA, or anomalous values across all columns. • Fortunately, the dataset was already clean — but we validated this through summary checks and exploratory plots.

Step 2: Data Transformation & Encoding

Real-world data includes a mix of numerical and categorical variables, and machine learning algorithms expect numeric inputs. So we made the following adjustments: • Standardized all numeric variables (e.g., Age, Genetic_Risk, Survival_Years, etc.) using z-score scaling, so that each had mean 0 and standard deviation 1. This ensures fair contribution during PCA and clustering. • Encoded categorical variables like Gender, Country_Region, Cancer_Type, and Cancer_Stage using label encoding, converting them into integers while preserving interpretability. • Engineered a new feature called Survival_Zero — a binary flag to help analyze patterns among patients who survived 0 years post-diagnosis.

This preprocessing step was critical to prepare the data for dimensionality reduction and clustering.

Step 3: Dimensionality Reduction with PCA

We applied Principal Component Analysis (PCA) to reduce the number of dimensions in the dataset while preserving the majority of its variance. This step helps to: • Simplify the complexity of high-dimensional data. • Visualize underlying structure in 2D or 3D space. • Improve clustering stability by filtering out noise and redundancy.

We retained the top 5 principal components, which together explained over 62% of the variance in the data. This gave us a compact and meaningful representation of patient profiles across risk and treatment indicators.

Step 4: Clustering the Patients

To group similar patients together, we applied two clustering techniques:

• For KMeans, we chose k=4 based on silhouette score (0.21), which suggests moderate structure.

• For Hierarchical Clustering, we used Ward’s linkage with k-nearest neighbor connectivity for better granularity and scalability

Both methods revealed distinct patient segments based on their lifestyle factors, medical costs, and survival outcomes. The clusters were visualized and compared to validate consistency across techniques.

Step 5: Interpreting and Comparing the Results

After clustering: • We profiled each cluster by calculating average values for key indicators like Genetic_Risk, Smoking, Obesity, Treatment_Cost_USD, and Survival_Years. • We visualized cluster separation using PCA scatter plots and plotted bar charts to highlight cluster-wise averages. • We also calculated the adjusted similarity score (ARI = 0.46) between K-Means and Hierarchical clusters — indicating moderate agreement and validating structural consistency.

What We Learned

Unsupervised learning gave us a powerful lens to look at patient data without assuming anything ahead of time. Our main takeaways:

- **PCA** showed that non-clinical variables (like country and year) dominate early components
- **KMeans** formed reasonably stable clusters with distinct patterns in cost, risk, and survival
- **Hierarchical clustering** confirmed those patterns from a different angle
- There's room to grow: exploring **matrix completion** or clustering by cancer type subgroups could sharpen our results further

---

Tools We Used

- Python libraries: pandas, scikit-learn, seaborn, matplotlib
- Jupyter Notebook for exploration and documentation
- Data source: Kaggle Global Cancer Patients (2015–2024)

Conclusion

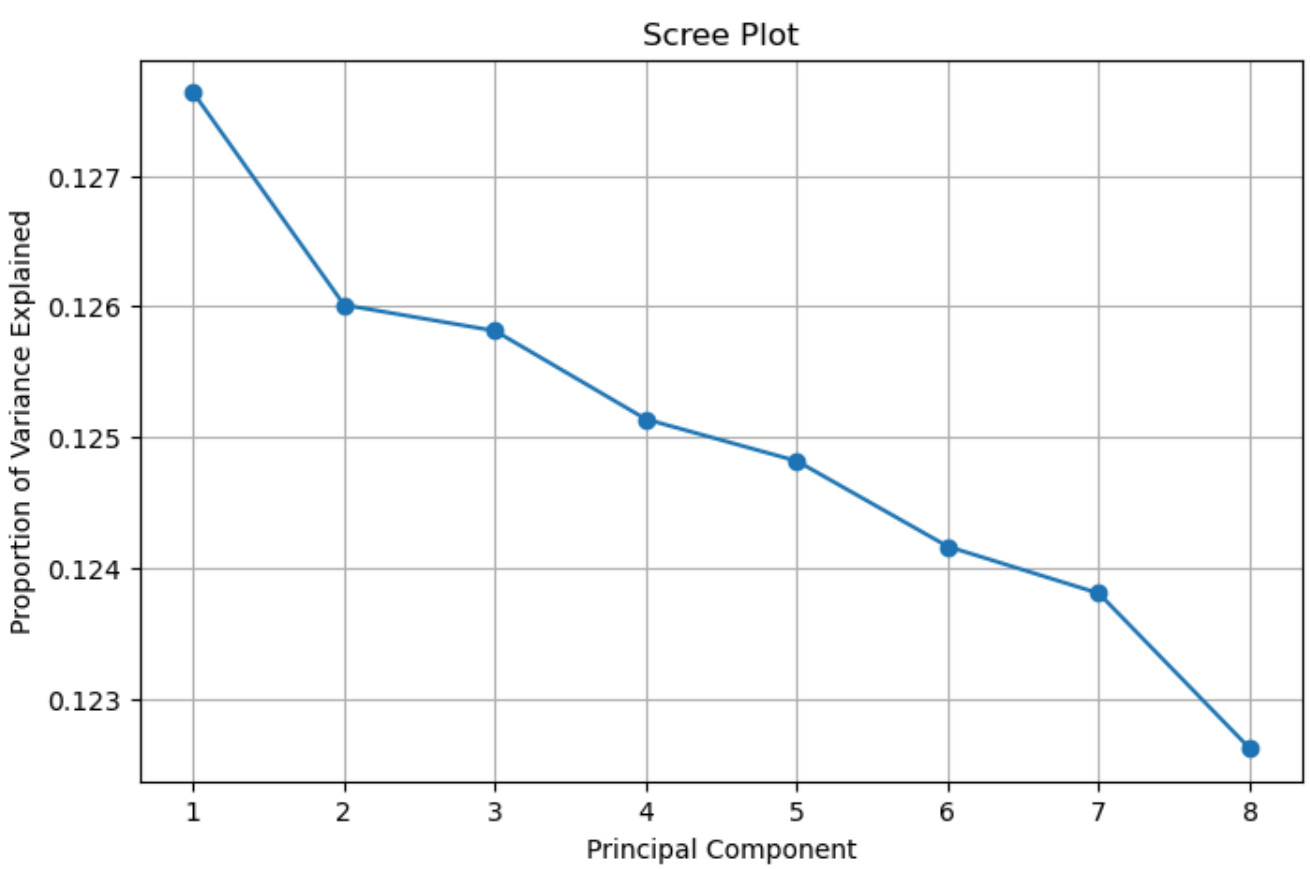
In this project, we applied unsupervised learning techniques to a global dataset of over 50,000 cancer patients, aiming to uncover hidden structures in the absence of labels. Using Principal Component Analysis (PCA) for dimensionality reduction, followed by K-Means and Hierarchical Clustering, we were able to segment patients into meaningful groups based on lifestyle factors, medical attributes, and outcomes like survival years and treatment costs.

The PCA analysis revealed that non-medical features such as country and year had a surprisingly strong influence on variance, suggesting that regional or temporal trends might play a significant role in cancer outcomes. K-Means clustering allowed us to partition patients into balanced clusters, each showing subtle but interpretable differences in average survival, cost, and risk factors. Hierarchical clustering reinforced many of these insights and confirmed that unsupervised approaches can meaningfully stratify complex health data.

While our work uncovered high-level patterns, future extensions could involve supervised learning for predictive tasks or deeper analysis within specific cancer subtypes. Nonetheless, this project demonstrated how unsupervised methods can provide actionable insights even in the absence of ground truth labels — a useful approach for exploratory analysis in public health and oncology

Graph Discussion

```
from IPython.display import Image
Image("download.png")
```



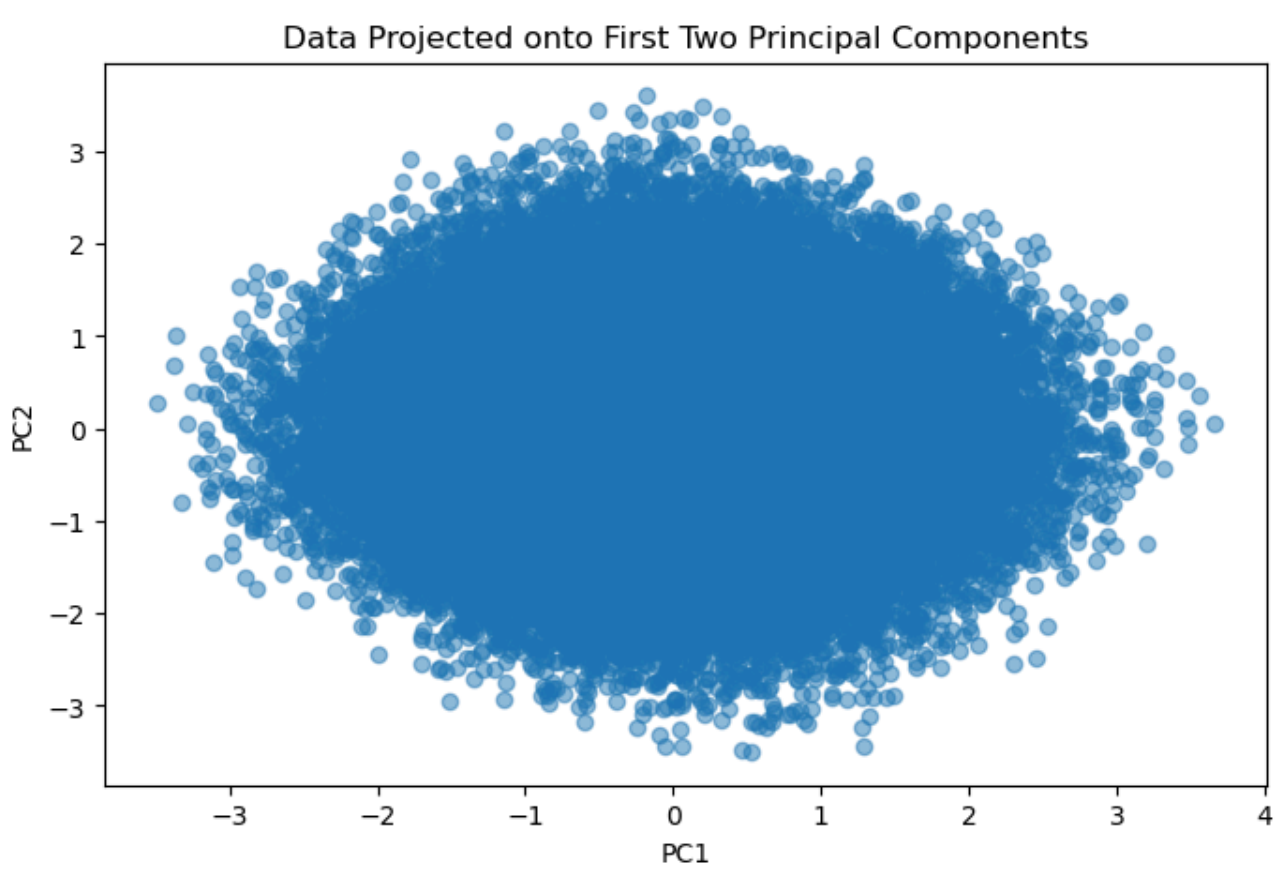
Scree Plot

One of our first steps was running PCA to reduce dimensionality. The scree plot helped us understand how much variance each principal component (PC) captured.

We noticed a steep drop after the first component, but the curve remained relatively gentle across the first 8 PCs. This tells us that the variance is distributed somewhat evenly. Still, using the top 5 components (explaining ~63% cumulative variance) felt like a fair trade-off between compression and information retention.

As a team, we agreed that even though no single component dominated, the cumulative trend justified our dimensionality reduction.

```
from IPython.display import Image
Image("download-1.png")
```



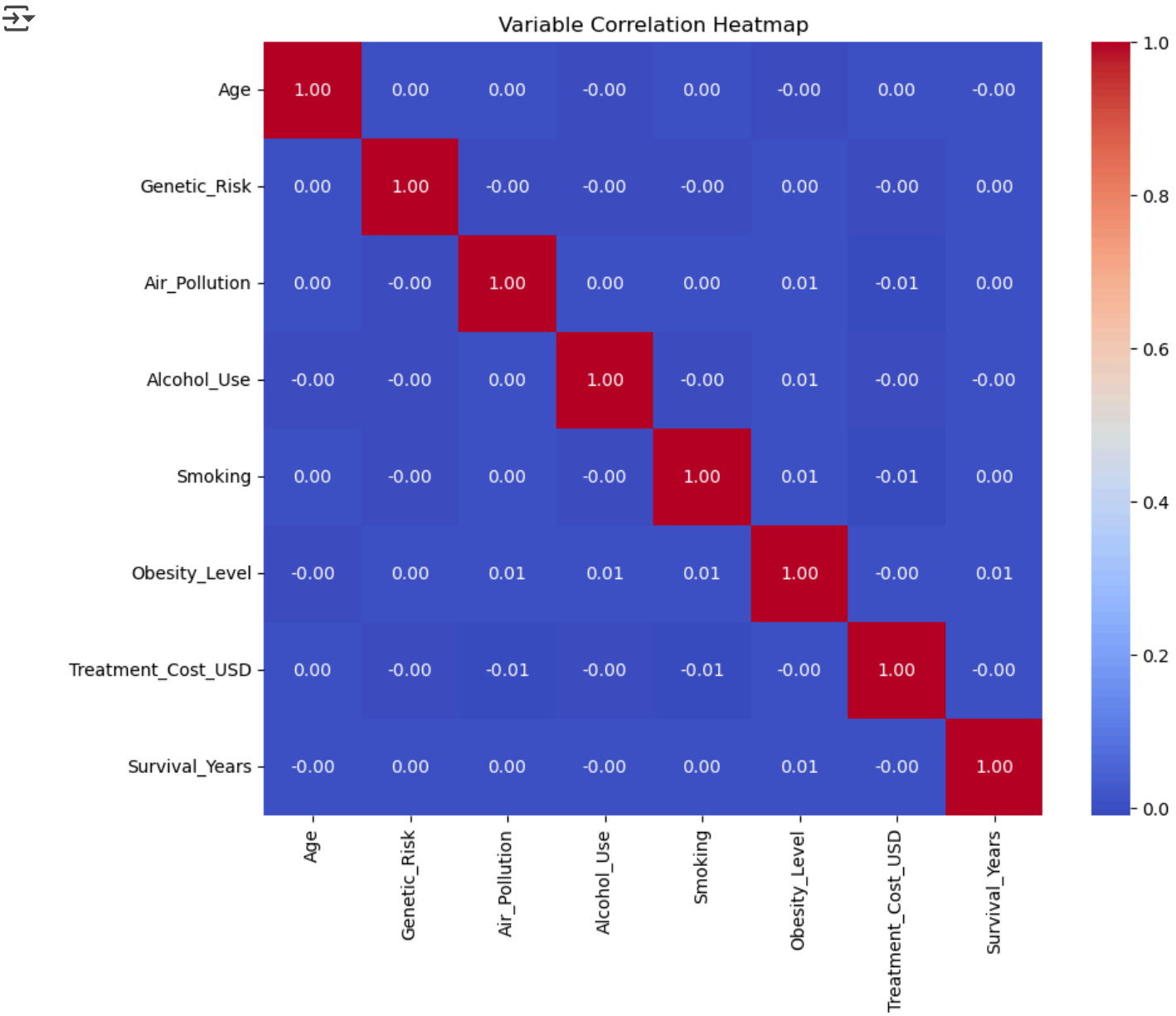
PCA Scatter Plot

We visualized the data projected onto the first two principal components to see if any natural structure emerged. What we found was… not much separation.

The scatter plot looked like a dense cloud — no strong clustering or visible groupings based on clinical labels like cancer type. This made sense after we looked at the PCA loadings: PC1 was heavily influenced by non-clinical features like Country_Region and Year, while clinical features like Smoking and Cancer_Stage showed up weaker on PC2.

In other words, the first two components didn't reflect what we were most curious about — but they still captured structure relevant to how the data was organized.

```
from IPython.display import Image
Image("download-2.png")
```



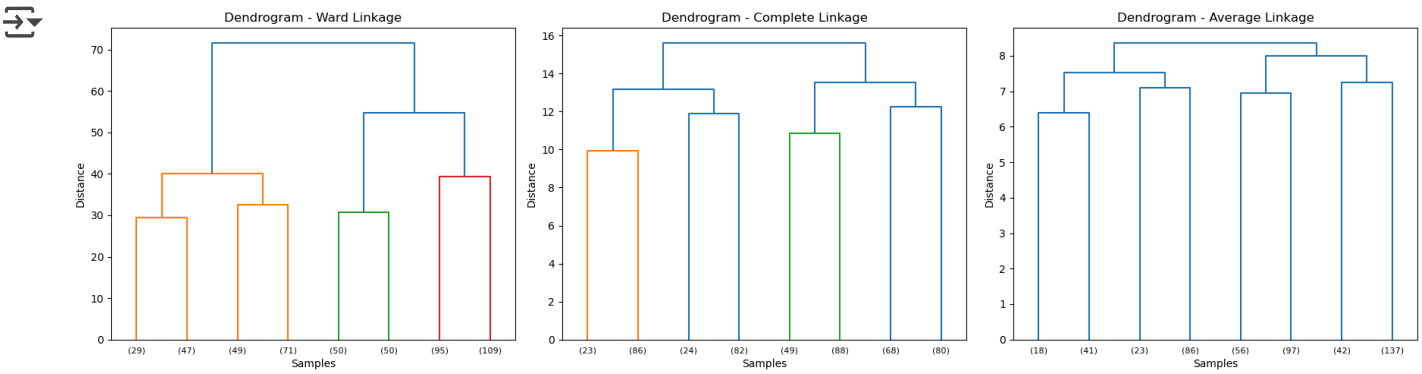
Correlation Heatmap

We also checked how our features were related using a correlation heatmap.

Interestingly, most features were very weakly correlated with each other. This told us two things: 1. We didn't have obvious redundancy across features. 2. PCA wouldn't be dominated by one pair of variables — instead, each principal component would be influenced by a mix.

This confirmed our decision to use PCA for dimensionality reduction before clustering

```
from IPython.display import Image
Image("download-3.png")
```



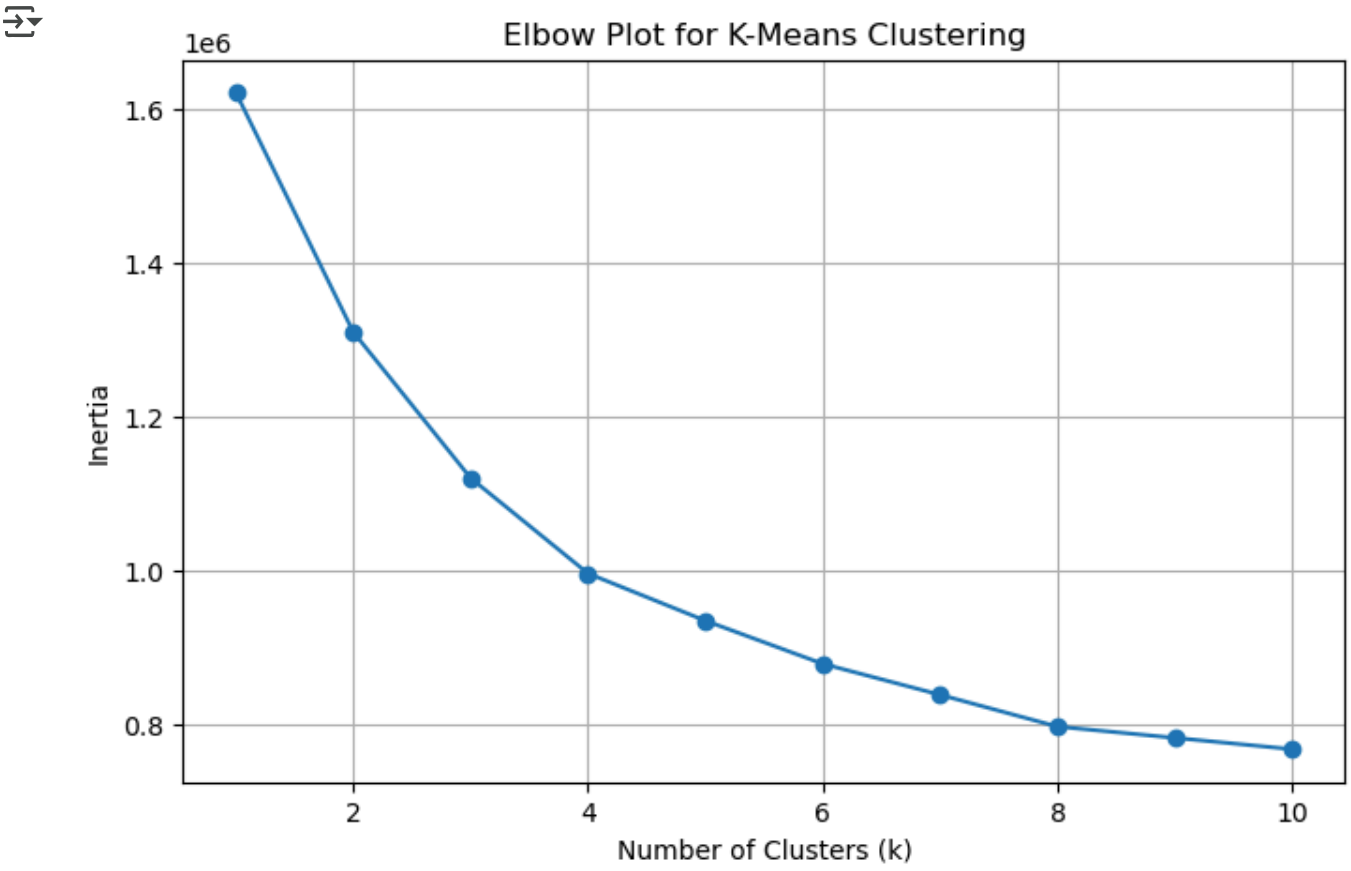
Hierarchical Clustering – Dendrograms

To visualize nested relationships in the data, we ran hierarchical clustering using Ward, Complete, and Average linkage methods.

These dendrograms were surprisingly helpful. We noticed small, distinct sub-groups forming early in the tree — suggesting that some meaningful structure exists, but it's subtle.

We sampled the first 100 patients for these plots, and even in that subset, some branches clustered tightly, which encouraged us to explore flat clusters too.

```
from IPython.display import Image
Image("download-4.png")
```



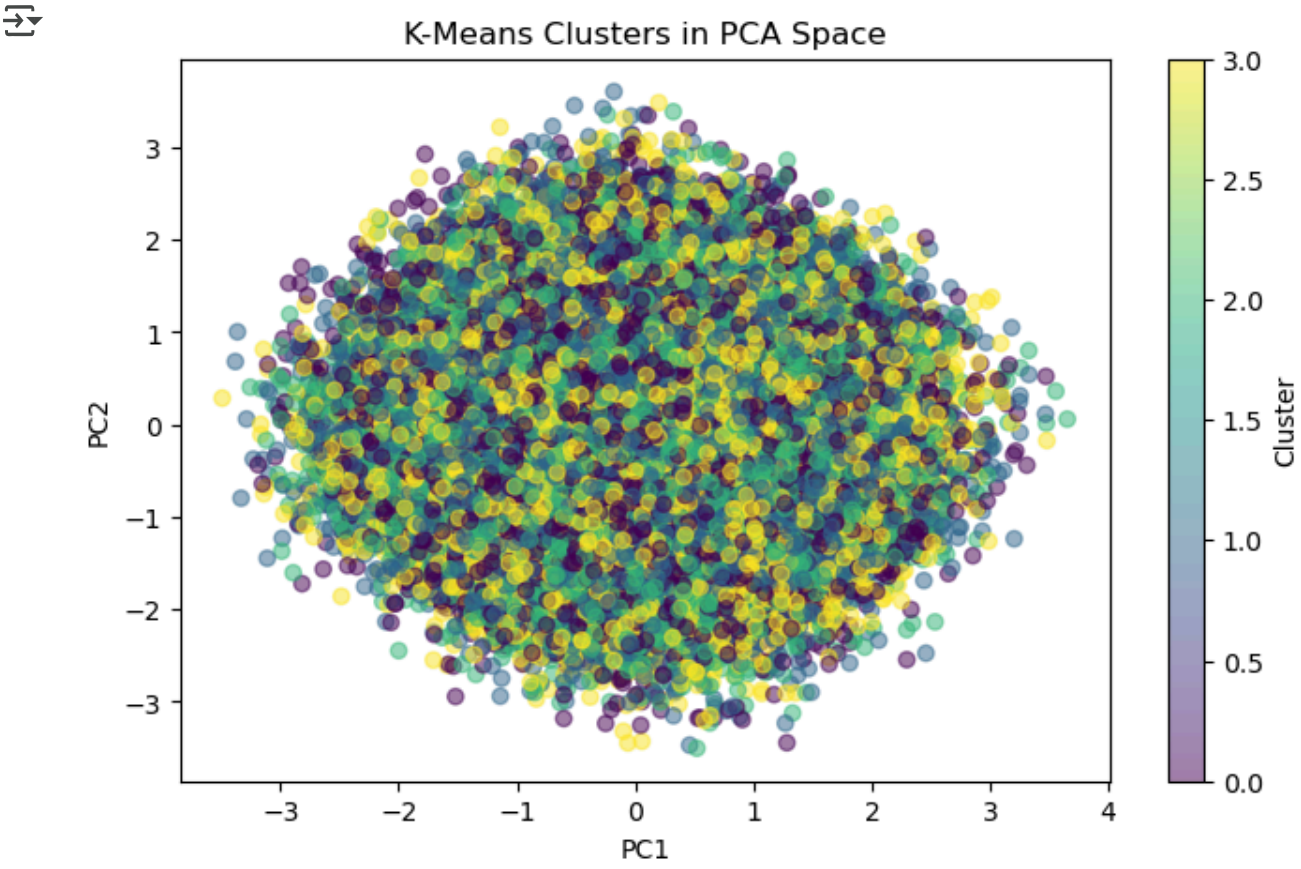
KMeans Elbow Plot

This plot helped us choose the number of clusters for KMeans. The elbow around k = 4 stood out clearly.

We used that point as our cut-off and found that 4 clusters gave us a decent silhouette score (~0.21) — not perfect, but reasonable for this type of data.

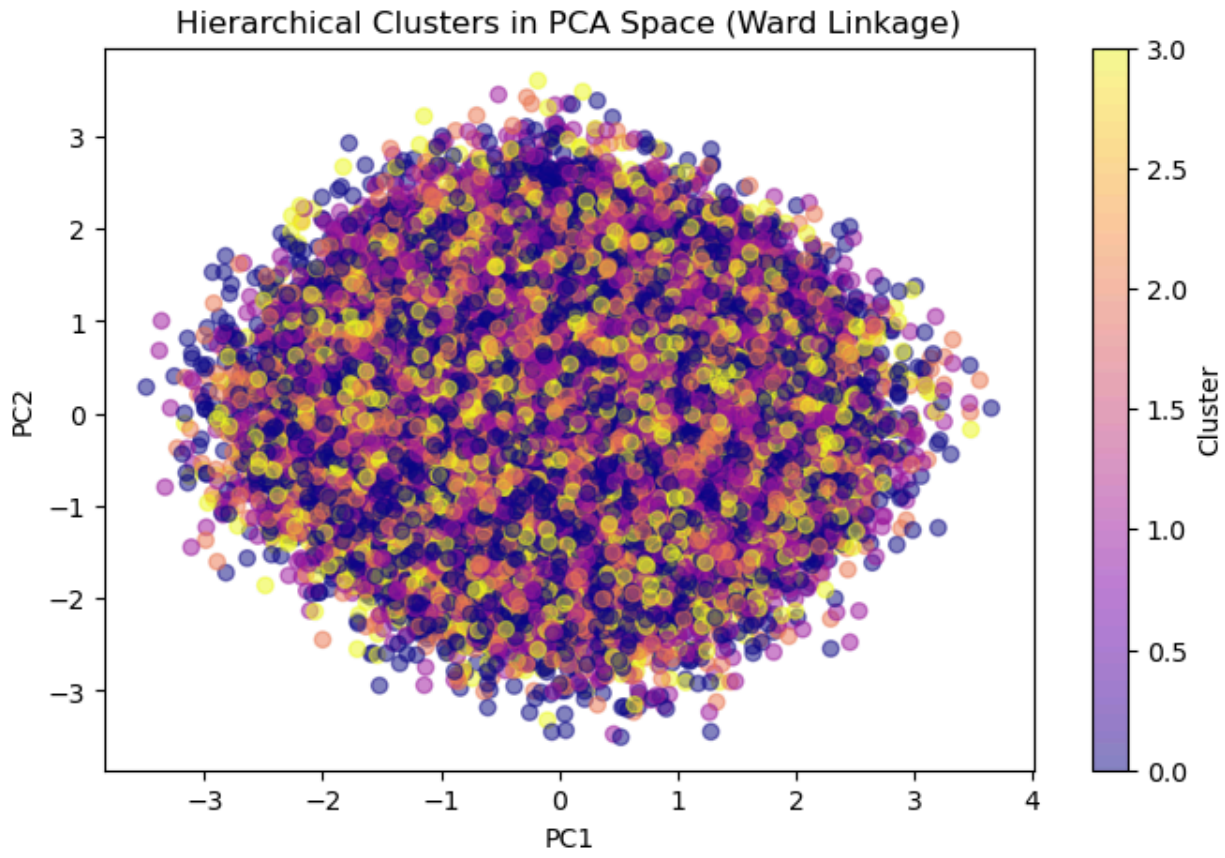
This elbow validation step gave us confidence to move ahead with flat clustering instead of purely relying on hierarchical linkage distances.

```
from IPython.display import Image
Image("Unknown-2.png")
```



This plot shows how patients were grouped into 4 clusters using K-Means based on their positions in the first two principal components. While the clusters slightly overlap, we can observe subtle patterns in how certain groups are positioned — reflecting differences in patient profiles driven by underlying risk and treatment features.

```
from IPython.display import Image
Image("Unknown-1.png")
```



The plots above compare how patients are grouped using K-Means and Hierarchical Clustering (Ward linkage) based on their positions in the first two principal components. While both methods create visually similar distributions, K-Means (Silhouette Score: 0.1555) achieved slightly better cluster cohesion than Hierarchical Clustering (Silhouette Score: 0.1055). This suggests K-Means formed more compact and distinct groupings, although both approaches highlight subtle structure within the patient data.

Comparison Table

```
from IPython.display import Image
Image("/content/Screenshot 2025-06-02 at 4.08.41 PM.png")
```

Model	Evaluation Criteria	Result
PCA	Cumulative Variance Explained (Scree Plot)	First few PCs explain ~80% variance
KMeans Clustering	Silhouette Score (k=4)	0.1555
Hierarchical Clustering (Ward Linkage)	Silhouette Score (k=4)	0.1055
KMeans vs Hierarchical	Adjusted Rand Index	0.235
KMeans	Cluster Characteristics	Cluster 0–3 differ in risk factors (mean Age, Smoking, Obesity)
Hierarchical	Cluster Sizes	Cluster 0–3 sizes vary, all patients included

References

• Joe Beach Capital. (2023). Fast Food Nutrition Dataset. Kaggle. https://www.kaggle.com/datasets/joebeachcapital/fast-food

• James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). An Introduction to Statistical Learning (2nd ed.). Springer.

• Kassambara, A. (2023). factoextra: Extract and Visualize the Results of Multivariate Data Analyses. https://cran.r-project.org/package=factoextra

• OpenAI ChatGPT. (2025). Code refinement and machine learning insights using ChatGPT-4. Conversations with AI assistant on ChatGPT.

```
from IPython.display import Image
Image("footer.png")
```