Unit 2C: Visualizing elements of $\mathbb{R}^n$

Prof.Dr.P.M.Bajracharya

School of Mathematical Sciences
T.U., Kirtipur

March 1, 2025

Summary

1. Ways to label points, elements, vectors, matrices
2. Visualizing elements
3. Vector addition and scalar multiplication

Ways to label points, elements, vectors, matrices

Consider three points on the plane. It may seem natural or more convenient to label them as (x, y) pairs, with a number indicating which point it is, as in:

$$\begin{array}{ll} (x_1, y_1) & \text{is the first point} \\ (x_2, y_2) & \text{is the second point} \\ (x_3, y_3) & \text{is the third point.} \end{array}$$

For three points in space ($\mathbb{R}^3$),

$$\begin{array}{ll} (x_1, y_1, z_1) & \text{is the first point} \\ (x_2, y_2, z_2) & \text{is the second point} \\ (x_3, y_3, z_3) & \text{is the third point.} \end{array}$$

That convention is fine, but we will adopt a numbering convention that will help us generalize to many dimensions, e.g., to $\mathbb{R}^n$, for arbitrary n. In general, given a vector $x \in \mathbb{R}^n$, we will label the different elements of x with subscripts, as in

$$\begin{aligned} x &= (x_1, x_2, ..., x_n) \quad \text{for } x \in \mathbb{R}^n \\ y &= (y_1, y_2, ..., y_n) \quad \text{for } y \in \mathbb{R}^n. \end{aligned}$$

This system works well if there are a small number of points. However, how to denote points if there are , say, 100 points? The above system is difficult to implement. So, we need a more *powerful system of notations* to denote many points in high dimension.

More powerful system of notations:
We can label the elements in a way that makes the interpretation of matrices as consisting of high-dimensional vectors more immediate.

$$\begin{aligned} x_1 &= (x_{11}, x_{12}) \quad \text{is the first point} \\ x_2 &= (x_{21}, x_{22}) \quad \text{is the second point} \\ x_3 &= (x_{31}, x_{32}) \quad \text{is the third point.} \end{aligned}$$

Given this notational convention, we could write this as a matrix:

$$A = \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{31} & x_{32} \end{pmatrix}$$

Alternatively, we could change the order of the subscripts, letting the first subscript denote the component and the second subscript denote the point, to obtain

$x_1 = (x_{11}, x_{21})$ is the first point

$x_2 = (x_{12}, x_{22})$ is the second point

$x_3 = (x_{13}, x_{23})$ is the third point.

In this case, we obtain the following matrix:

$$A = \begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \end{pmatrix}$$

Here, we have still used the notation that element ij refers to the element that is in the ith horizontal row and the jth vertical column, but we have swapped what rows and columns mean. In this case, the columns of A are data points.

Visualizing elements

If, instead of working with a small number of points in $\mathbb{R}^2$ or $\mathbb{R}^3$, we have 10^9 points, each of which is described by a set of 10^6 numbers, then we have a much larger matrix. Not only would it be difficult for a human to write out that many points, but it is not possible to visualize points in that many dimensions.

Question: What do 10^9 points in 10^6 "look like"? For example, do they look similar to or different than on $\mathbb{R}^2$ or $\mathbb{R}^3$?

This is very common in data science, but it can be a very difficult question to answer.

Another question: What does it even mean to ask that previous question? That is, how can we quantify what "look like" means?

That ambiguity being noted, it can be difficult to tell what such a data set "looks like," and indeed one of the main challenges here is going to be that high dimensional spaces are very different that low dimensional spaces.

Data visualization is one of the main pillars supporting data analysis. It has been a powerful tool and has been widely adopted by organizations owing to its effectiveness in abstracting out the right information, understanding and interpreting results clearly and easily.
For $n = 1, 2, 3$, this is easy – it simply corresponds to the familiar one-dimensional line, two-dimensional plane, and three dimensional space, with which we are familiar, as well as familiar subsets of these spaces.

However, dealing with multi-dimensional datasets with typically more than two attributes start causing problems, since our medium of data analysis and communication is typically restricted to two dimensions.

Here, we will describe one way to "visualize" higher dimensional vectors.

We will call this the **parallel coordinates method** (or **augmented visualization method**).

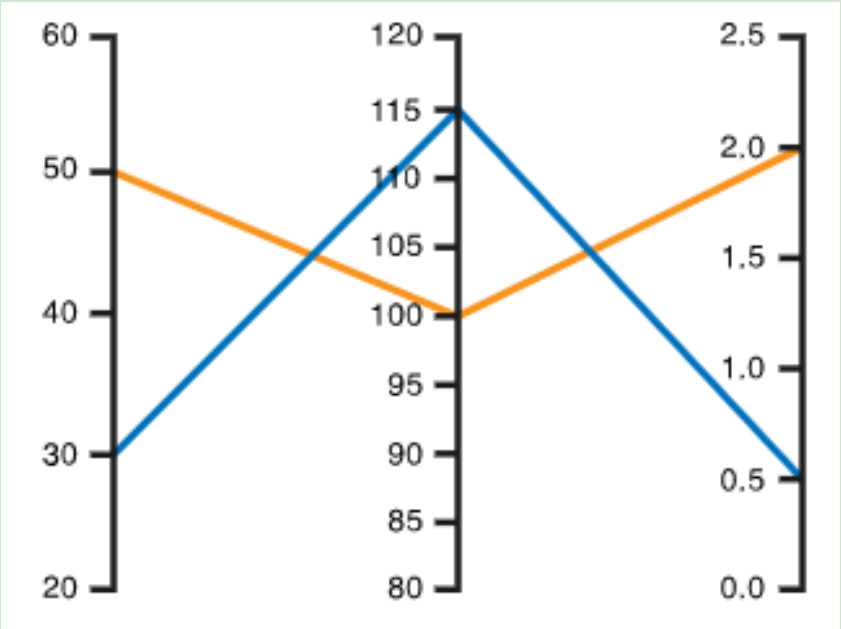
Parallel coordinates method:

- To show a set of points in an n-dimensional space, n equally spaced vertical lines are drawn.
- A point in n-dimensional space is represented as a polyline (broken lines) with vertices on the parallel axes.
- The position of the vertex on the i-th axis corresponds to the i-th coordinate of the point.

Thus, the basic idea of the augmented visualization method is to view a vector $x \in \mathbb{R}^n$ as n separate numbers $x_i \in \mathbb{R}$, which can be represented on a two dimensional piece of paper by n points on n separate number lines, plotted vertically and parallel to each other. On the ith vertical line, we plot the point x_i.

Example

Let $x = (30, 115, 0.5)$ and $y = (50, 100, 2)$ be two vectors in $(R)^3$. Using parsllel coordinates method, we plot them as in



- This type of visualisation is used for plotting multivariate, numerical data.
- Parallel Coordinates Plots are ideal for comparing many variables together and seeing the relationships between them.
- For example, if you had to compare an array of products with the same attributes (comparing computer or cars specifications across different models).

- In a Parallel Coordinates Plot, each component is given its own axis and all the axes are placed in parallel to each other.
- Each axis can have a different scale, as each variable works off a different unit of measurement.
- Values are plotted as a series of lines that connected across all the axes.
- This means that each line is a collection of points placed on each axis, that have all been connected together.

Ordering the axes:

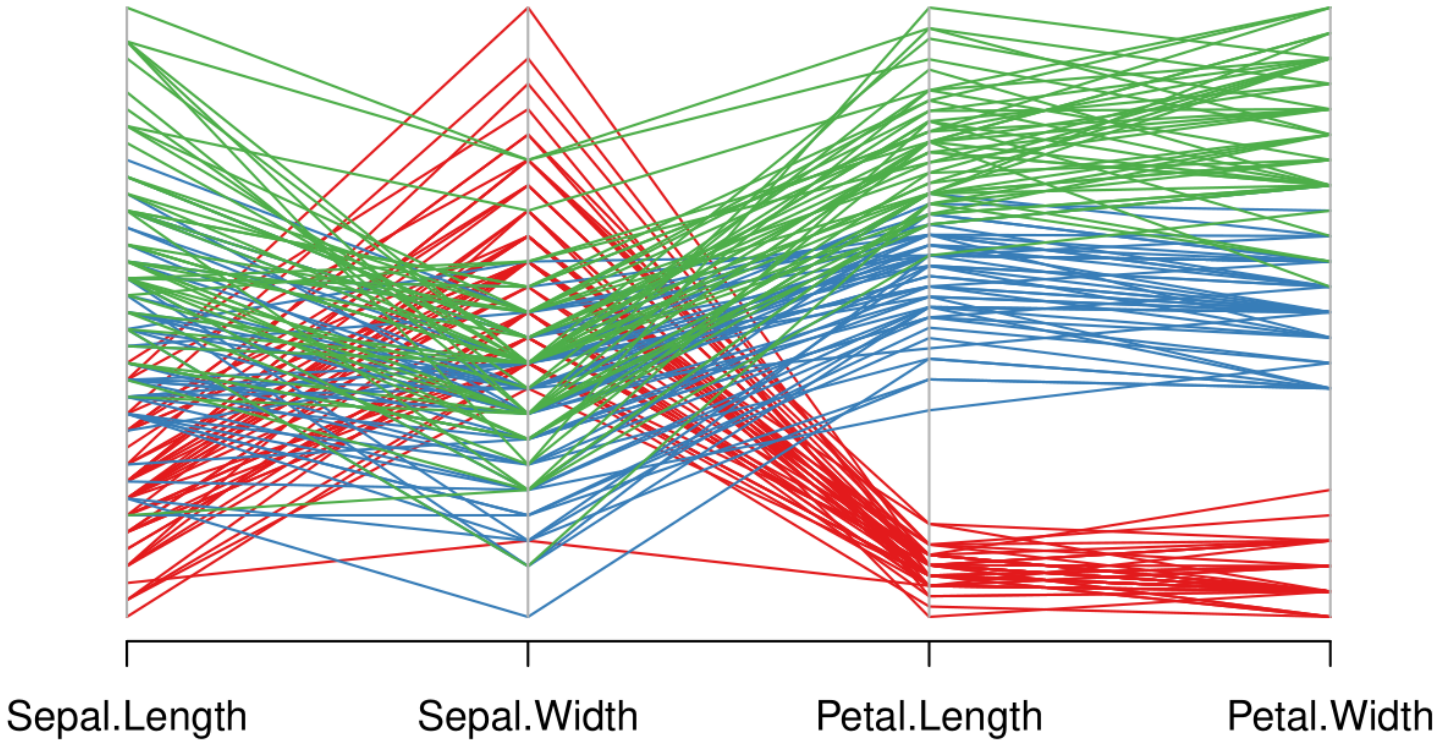
- The order the axes are arranged in can impact the way how the reader understands the data.
- One reason for this is that the relationships between adjacent variables are easier to perceive, than for non-adjacent variables.

So re-ordering the axes can help in discovering patterns or correlations across variables. Thus, it is important to find the best variable order in your parallel coordinates chart.

The downside to Parallel Coordinates

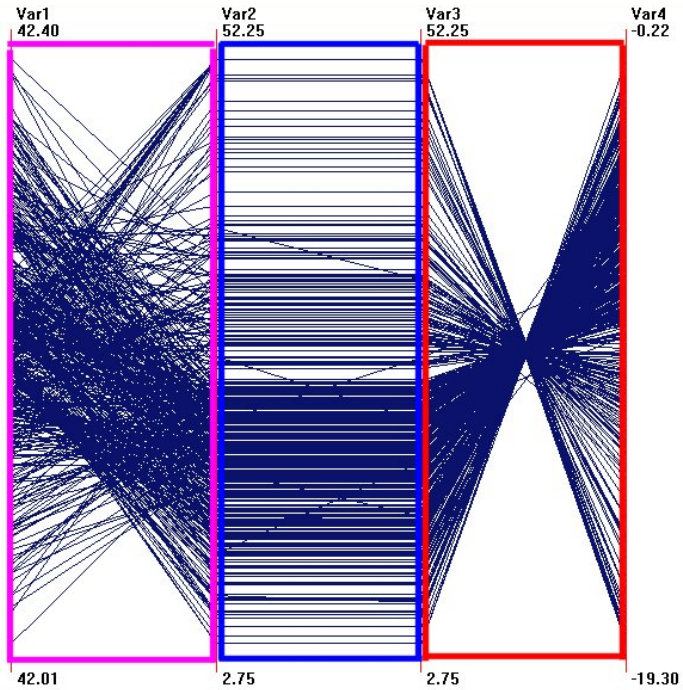
- Plots, is that they can become over-cluttered and therefore, illegible when they're very data-dense.
- The best way to remedy this problem is through interactivity and a technique known as "Brushing".
- Brushing highlights a selected line or collection of lines while fading out all the others.
- This allows you to isolate sections of the plot you're interested in while filtering out the noise.

Example.



Example.

Parallel Coordinates and Scatter Plot Matrix: This example shows that Var1-Var2 has no correlation; Var2-Var3 has very strong positive correlation; Var3-Var4 has very strong negative (inverse) correlation.



Vector addition and scalar multiplication

Vector addition: If $x, y \in \mathbb{R}^n$, then

$$z = x + y = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} + \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \vdots \\ x_n + y_n \end{pmatrix} \in \mathbb{R}^n.$$

Scalar multiplication: If $x, y \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$, then

$$y = \alpha x = \alpha \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \alpha x_1 \\ \alpha x_2 \\ \vdots \\ \alpha x_n \end{pmatrix} \in \mathbb{R}^n.$$

Both of the above operations give as output a vector in the same vector space, and thus we can apply them recursively.

Let $x, y \in \mathbb{R}^n$. The distance between x and y is given by $\|x - y\|$.

Example. If $x = (0, 0)$ and $y = (5, 5)$, then $x - y = (0 - 5, 0 - 5) = (5, 5)$.

1. $\|x - y\|_1 = \|(5, 5)\|_1 = 5 + 5 = 10.$
2. $\|x - y\|_2 = \|(5, 5)\|_2 = \sqrt{5^2 + 5^2} = 7.07.$
3. $\|x - y\|_\infty = \|(5, 5)\|_\infty = \max\{5, 5\} = 5.$

Here, we have considered two vectors in $\mathbb{R}^2$ and computed the norm of the difference vector which is also a vector in $\mathbb{R}^2$. Again,

$$\|x-y\|_\infty \leq \|x-y\|_2 \leq \|x-y\|_1,$$
$$\|x-y\|_1 \leq \sqrt{2}\|x-y\|_2 \leq 2\|x-y\|_\infty,$$

Two things should be noted.

- First, the distance between two vectors depends on the norm you use to measure distance.
- Second, the relationship between those distances is the same as the relationship between the corresponding norms (that we discussed eariler).

Clearly, these are true, since the distance is just a norm of the difference vector.

Another way in which norms are useful is that they can be used to normalize vectors.

Normalize : Divide a vector by its norm (or size) to get a unit vector (or normalized vector) in the direction of the original vector.

For example, the unit vector (or normalized vector) $u \in \mathbb{R}^n$ in the direction of the vector $x \in \mathbb{R}^n$ is given by

$$u = \frac{x}{\|x\|}.$$

This is an example of multiplying a vector by a scalar to construct another vector. Any vector, except for a vector in which each entry equals 0 and which thus has norm equal to 0, can be normalized.

Example. Let $x = (1, 2)$. In this case:

1. Since $\|x\|_1 = 3$, the normalized vector is $x' = \frac{1}{3}(1, 2) = (1/3, 2/3)$.
2. Since $\|x\|_2 = \sqrt{5}$, the normalized vector is $x' = \frac{1}{\sqrt{5}}(1, 2) = (1/\sqrt{5}, 2/\sqrt{5})$.
3. Since $\|x\|_\infty = 2$, the normalized vector is $x' = \frac{1}{2}(1, 2) = (1/2, 1)$.

Example. Let $x = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$. In this case:

Since $\|x\|_1 = |\cos\theta| + |\sin\theta|$, the normalized vector is

$$x' = \frac{1}{|\cos\theta| + |\sin\theta|} \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$$

$$= \begin{pmatrix} \cos\theta/(|\cos\theta| + |\sin\theta|) \\ \sin\theta/(|\cos\theta| + |\sin\theta|) \end{pmatrix}$$

Since $\|x\|_2 = 1$, the normalized vector is $x' = x$. That is, for every value of θ, the vector x is normalized, with respect to the Euclidean norm.

Since $\|x\|_\infty = \max\{|\cos\theta|, |\sin\theta|\}$, the normalized vector is

$$x' = \tfrac{1}{2}(1,2) = \begin{pmatrix} \cos\theta/\|x\|_\infty \\ \sin\theta/\|x\|_\infty \end{pmatrix}.$$

Example. Let $x = (1, 0, ..., 0) \in \mathbb{R}^n$. In this case:

1. Since $\|x\|_1 = 1$, the normalized vector is $x' = x$.
2. Since $\|x\|_2 = 1$, the normalized vector is $x' = x$.
3. Since $\|x\|_\infty = 1$, the normalized vector is $x' = x$.