07: Basis, Vector Space, and Applications

Chapter Goal: To formalize the concepts of **Basis**, Vector Space, and **Linear Independence**, and to understand **why** changing basis is such a powerful tool in machine learning.

---

1. Formal Definition: What is a "Basis"?

- **Basis:** A set of `n` vectors that are used as the "rulers" or "framework" to define a space.
- **Two Mandatory Conditions to be a Basis:**
    1. **Linearly Independent:** No single basis vector can be created from a linear combination (a mix) of the other basis vectors. They must all be "original" and point in fundamentally different directions.
    2. **Span the Space:** The Span (all possible linear combinations) of the basis vectors must be able to reach every single point within that space.
- **Dimension:** The number of linearly independent vectors (`n`) in the basis defines the **dimension** of the vector space.
    - 2 independent vectors → 2D Space.
    - 3 independent vectors → 3D Space.

---

2. Linear Independence

- **Intuitive Idea:** A new vector ($\vec{b}_3$) is **linearly independent** of $\vec{b}_1$ and $\vec{b}_2$ if it "escapes" the space that $\vec{b}_1$ and $\vec{b}_2$ have already formed.
    - If $\vec{b}_1$ forms a line (1D), $\vec{b}_2$ is independent if it is **not** on that same line.
    - If $\vec{b}_1$ and $\vec{b}_2$ form a plane (2D), $\vec{b}_3$ is independent if it is **not** on that plane (it points "out" of the plane).
- **Formal Definition:** A set of vectors $\{\vec{b}_1, \vec{b}_2, \ldots, \vec{b}_n\}$ is linearly independent if it's impossible to find scalars $a_1, a_2, \ldots$ (other than all zeros) such that one vector can be written as a combination of the others.
    - **Example:** $\vec{b}_3$ is independent of $\vec{b}_1, \vec{b}_2$ if the equation $\vec{b}_3 = a_1\vec{b}_1 + a_2\vec{b}_2$ has no solution.
- **Linearly Dependent:** If one vector **can** be created from a mix of the others (e.g., $\vec{b}_3$ lies on the plane of $\vec{b}_1, \vec{b}_2$), then the set is linearly dependent. That vector is "redundant" and does not add a new dimension.

---

3. Properties of a Basis (It Doesn't Have to Be Perfect)

The vectors in a basis **DO NOT** have to be:

- **Unit Vectors** (length of one).
- **Orthogonal** (perpendicular to each other).

BUT, if a basis satisfies both of these properties (making it an **orthonormal basis**), all calculations (like changing basis) become much easier, because we can use the [Projection] trick with the [Dot Product].

---

4. Application: Why Do We Care About Changing Basis?

- **Core Idea:** The original axes (`x` , `y`) of our data are often not the most informative. By changing the basis, we can view the data from a more useful perspective.

Example 1: Nearly Co-linear Data (like in PCA)

- **Problem:** We have a cloud of 2D data points that appear to form a slanted line.
- **Change of Basis Idea:** Instead of using the `x` and `y` axes, let's create a new basis:
    - **Basis 1 ($\vec{b}_1$):** Aligned with the "best-fit" line that represents the data. This axis captures the primary information or **"signal"**.
    - **Basis 2 ($\vec{b}_2$):** Perpendicular to $\vec{b}_1$. This axis captures the distance of each point from the line, which we can consider **"noise"** or error.
- **Usefulness:**
    - We can "discard" the $\vec{b}_2$ dimension to reduce the dimensionality of the data without losing much information.
    - The magnitude of the coordinates in the $\vec{b}_2$ dimension tells us how well our line model fits the data.

Example 2: Facial Recognition (Neural Networks)

- **Problem:** The initial input is thousands of raw pixel values. This is not an informative basis.
- **Change of Basis Idea:** A neural network learns to transform this pixel basis into a new, more meaningful basis:
    - **Basis 1:** "Degree of nose curvature".
    - **Basis 2:** "Skin tone".
    - **Basis 3:** "Distance between eyes".
    - ...and so on.
- **Goal:** The "learning" process is the search for the most informative basis that can distinguish one face from another.

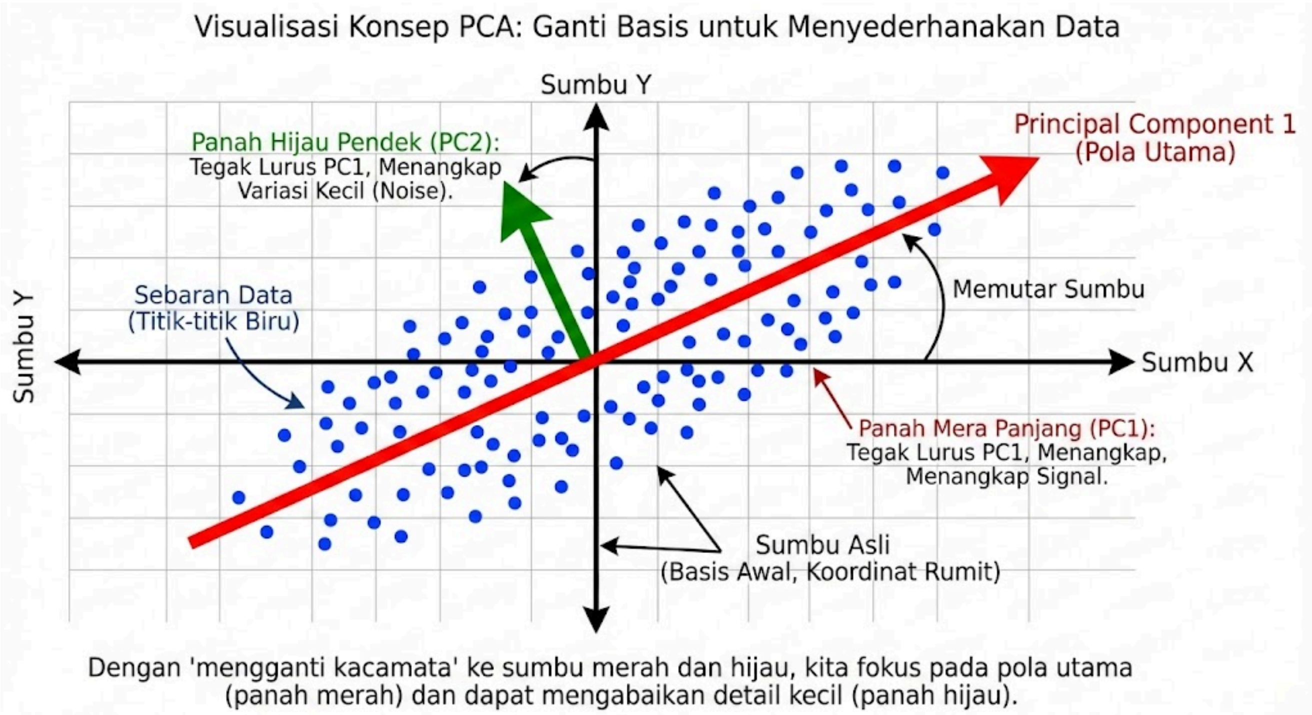
Final Conclusion: Change of Basis is one of the most powerful tools in data science and machine learning. It is the method for extracting meaningful **features** and simplifying data by viewing it from the right perspective.

> **Note!**
>
> Penekanan biar lebih paham lagi :

5. Intuitive Visualizations for Applications

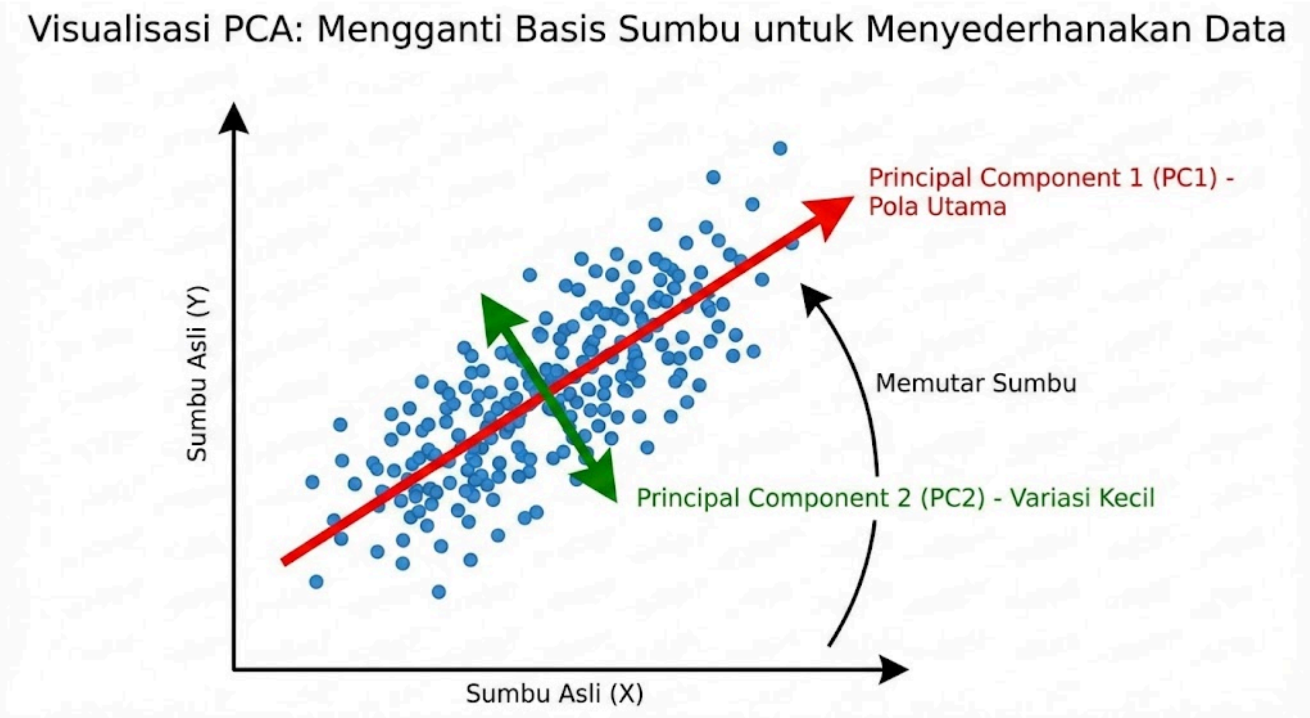
Visualization 1: "Tilting Your Head" at Data (Signal vs. Noise)

This is a classic example from Data Science (often related to PCA or Regression).



- **Old Basis (Standard X and Y axes):**
  - Look at the cloud of blue data points. They are scattered in a diagonal, slanted pattern.
  - If we use the standard X and Y axes, the data looks "messy". The coordinates are large in both X and Y, making it hard to see the core pattern.
- **Changing the Basis (New Red & Green Axes):**
  - Imagine we "tilt our head" or rotate the axes to better align with the data.
  - **New Basis Vector 1 (The "Signal" Axis):** We place this axis right through the middle of the data cloud, following its main direction.
    - This axis captures the **"main trend"** or the most important information in the data. It says: "Oh, the data mainly varies along this direction."

    - **New Basis Vector 2 (The "Noise" Axis):** We place this axis perpendicular to the signal axis.
        - This axis now only measures how much each point **deviates** from the main trend line. The smaller the values on this axis, the cleaner the data.
- **Why is this useful?** The computer becomes smarter. It can say: "Keep the data along the Signal axis (important), and discard the data along the Noise axis (it's just random variation)." This allows for **dimensionality reduction**—making files smaller while keeping the most crucial information intact.



Visualization 2: Facial Recognition - "From Pixels to Features"

Now, imagine how a computer sees a face.

- **Old Basis (Pixels):** The computer sees a face as thousands of individual color dots. "Pixel 1 is brown, Pixel 2 is beige..." This is **NOT USEFUL**. The computer doesn't know what a nose or an eye is.
- **Changing the Basis (Eigenfaces):** The computer can learn to change the basis of the face.
- **Analogy: A Cooking Recipe**
    - **Old Way (Pixel Basis):** "Put a grain of salt at coordinate (1,1), put a grain of sugar at coordinate (1,2)..." (Millions of tedious instructions).
    - **New Way (Feature Basis):** "Take 1 scoop of the 'Base Face Shape' (Basis 1) + add a little bit of 'Eye Shadow Pattern' (Basis 2) - subtract some 'Cheek Shape' (Basis 3)."

With a **Change of Basis**, the computer no longer sees "color dots"; it sees **FEATURES** (Nose, Eyes, Cheeks). This is why it can recognize your face even if the lighting in a photo changes slightly.

Visual Conclusion: Changing Basis is Just Changing Your Glasses

- In the **Data Example**, we changed our glasses to see the messy cloud as a straight line.
- In the **Face Example**, we switched from "pixel microscope glasses" to "facial feature glasses".

This illustrates the core purpose of Changing Basis: **to separate important information (signal) from unimportant variation (noise)** by finding the right perspective.

Tags: #mml-specialization #linear-algebra #basis-vectors #vector-space #linear-independence #pca ````