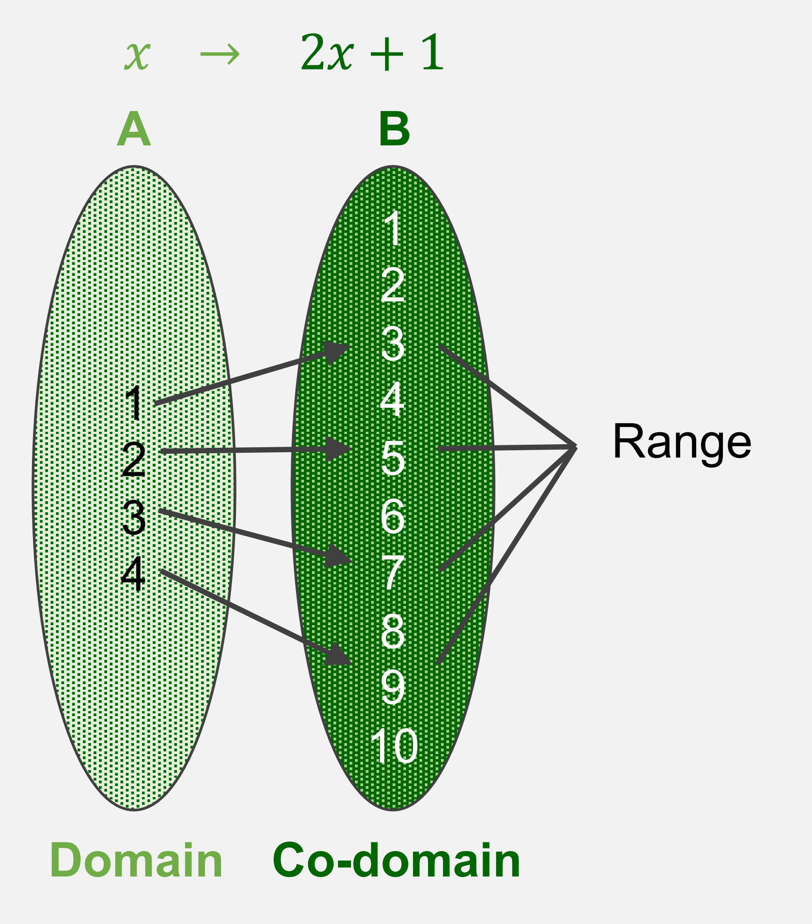
선형변환

Transformation

- A **transformation**, **function**, or **mapping**, T maps an input x to an output y
  - Mathematical notation: $T\colon x \mapsto y$
- **Domain**: Set of all the possible values of x
- **Co-domain**: Set of all the possible values of y
- **Image**: a mapped output y, given x
- **Range**: Set of all the output values mapped by each x in the domain

- **Note**: the output mapped by a particular x is uniquely determined.

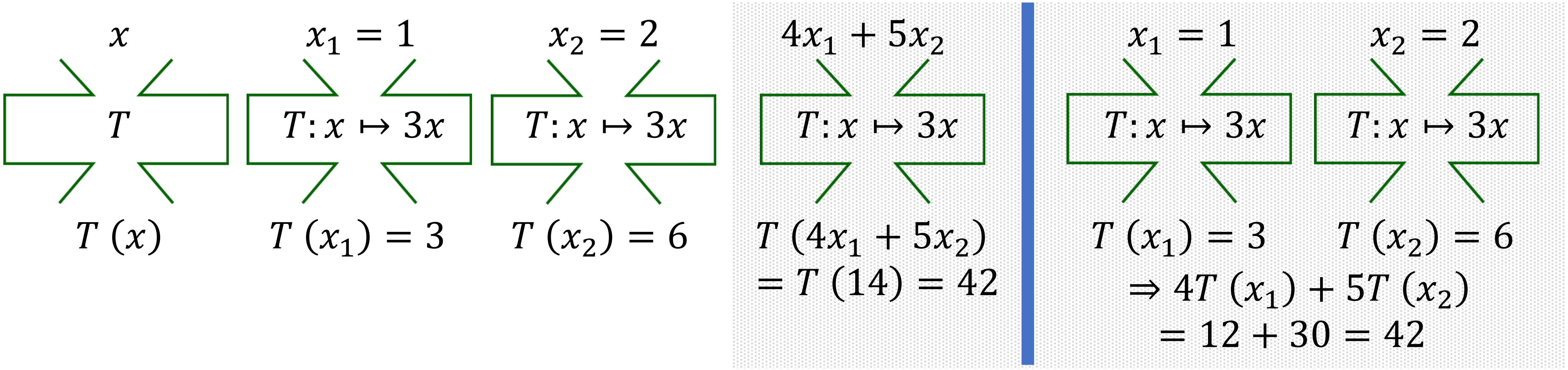


Linear Transformation

- **Definition**: A transformation (or mapping) T is **linear** if:

  I. $T\left(c\mathbf{u} + d\mathbf{v}\right) = cT\left(\mathbf{u}\right) + dT\left(\mathbf{v}\right)$ for all $\mathbf{u}, \mathbf{v}$ in the domain of T and for all scalars c and d

- Simple example: $T\colon\ x \mapsto y,\ T(x) = y\ = 3x$



Transformations between Vectors

- $T: \mathbf{x} \in \mathbb{R}^n \mapsto \mathbf{y} \in \mathbb{R}^m$: Mapping n-dim vector to m-dim vector
- Example:

$$T: \mathbf{x} \in \mathbb{R}^3 \mapsto \mathbf{y} \in \mathbb{R}^2 \quad \mathbf{x} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \in \mathbb{R}^3 \quad \mapsto \quad \mathbf{y} = T(\mathbf{x}) = \begin{bmatrix} 4 \\ 5 \end{bmatrix} \in \mathbb{R}^2$$

Matrix of Linear Transformation

- Example: Suppose T is a linear transformation from $\mathbb{R}^2$ to $\mathbb{R}^3$ such that $T\left(\begin{bmatrix}1\\0\end{bmatrix}\right) = \begin{bmatrix}2\\-1\\1\end{bmatrix}$ and $T\left(\begin{bmatrix}0\\1\end{bmatrix}\right) = \begin{bmatrix}0\\1\\2\end{bmatrix}$. With no additional information, find a formula for the image of an arbitrary **x** in $\mathbb{R}^2$

$$\mathbf{x} = \begin{bmatrix}x_1\\x_2\end{bmatrix} = x_1\begin{bmatrix}1\\0\end{bmatrix} + x_2\begin{bmatrix}0\\1\end{bmatrix}$$

$$\Rightarrow T(\mathbf{x}) = T\left(x_1\begin{bmatrix}1\\0\end{bmatrix} + x_2\begin{bmatrix}0\\1\end{bmatrix}\right) = x_1 T\left(\begin{bmatrix}1\\0\end{bmatrix}\right) + x_2 T\left(\begin{bmatrix}0\\1\end{bmatrix}\right)$$

$$= x_1\begin{bmatrix}2\\-1\\1\end{bmatrix} + x_2\begin{bmatrix}0\\1\\2\end{bmatrix} = \begin{bmatrix}2 & 0\\-1 & 1\\1 & 2\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix}$$

Matrix of Linear Transformation

- In general, let $T : \mathbb{R}^n \to \mathbb{R}^m$ be a linear transformation.
  Then T is always written as a matrix-vector multiplication, i.e.,

$$T(\mathbf{x}) = A\mathbf{x} \text{ for all } \mathbf{x} \in \mathbb{R}^n$$

- In fact, the j-th column of $A \in \mathbb{R}^{m\times n}$ is equal to the vector $T(\mathbf{e}_j)$, where $\mathbf{e}_j$ is the j-th column of the identity matrix in $\mathbb{R}^{n\times n}$:

$$A = [T(\mathbf{e}_1) \quad \cdots \quad T(\mathbf{e}_n)]$$

- Here, the matrix A is called the **standard matrix** of the linear transformation T

Matrix of Linear Transformation

- Example: Find the standard matrix A of a linear transformation from $\mathbb{R}^3$ to $\mathbb{R}^2$ such that

$$T\left(\begin{bmatrix}1\\0\\0\end{bmatrix}\right)=\begin{bmatrix}1\\2\end{bmatrix}, T\left(\begin{bmatrix}0\\1\\0\end{bmatrix}\right)=\begin{bmatrix}4\\3\end{bmatrix}, \text{ and } T\left(\begin{bmatrix}0\\0\\1\end{bmatrix}\right)=\begin{bmatrix}5\\6\end{bmatrix}.$$

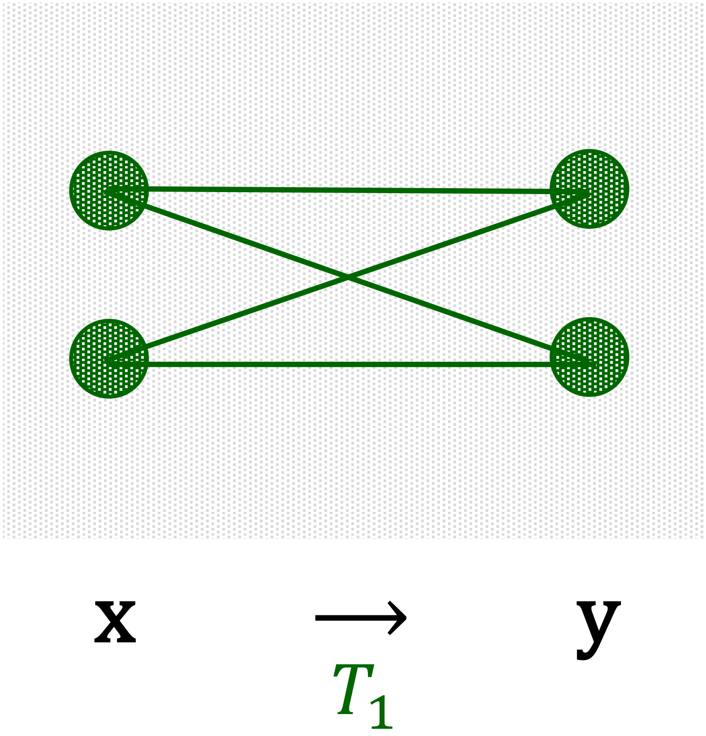
$$\mathbf{x}=\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=x_1\begin{bmatrix}1\\0\\0\end{bmatrix}+x_2\begin{bmatrix}0\\1\\0\end{bmatrix}+x_3\begin{bmatrix}0\\0\\1\end{bmatrix}$$

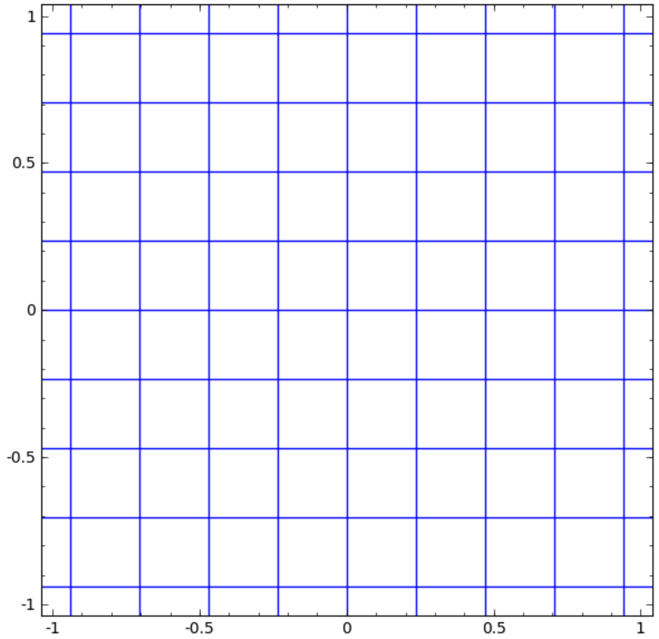
$$\Rightarrow T(\mathbf{x})=T\left(x_1\begin{bmatrix}1\\0\\0\end{bmatrix}+x_2\begin{bmatrix}0\\1\\0\end{bmatrix}+x_3\begin{bmatrix}0\\0\\1\end{bmatrix}\right)=x_1T\left(\begin{bmatrix}1\\0\\0\end{bmatrix}\right)+x_2T\left(\begin{bmatrix}0\\1\\0\end{bmatrix}\right)+x_3T\left(\begin{bmatrix}0\\0\\1\end{bmatrix}\right)$$

$$=x_1\begin{bmatrix}1\\2\end{bmatrix}+x_2\begin{bmatrix}4\\3\end{bmatrix}+x_3\begin{bmatrix}5\\6\end{bmatrix}=\begin{bmatrix}1&4&5\\2&3&6\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=A\mathbf{x}$$

Linear Transformation in Neural Networks

- Fully-connected layers (linear layer)

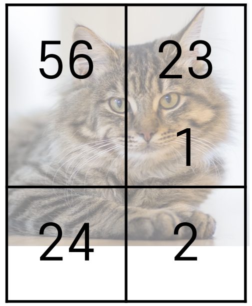


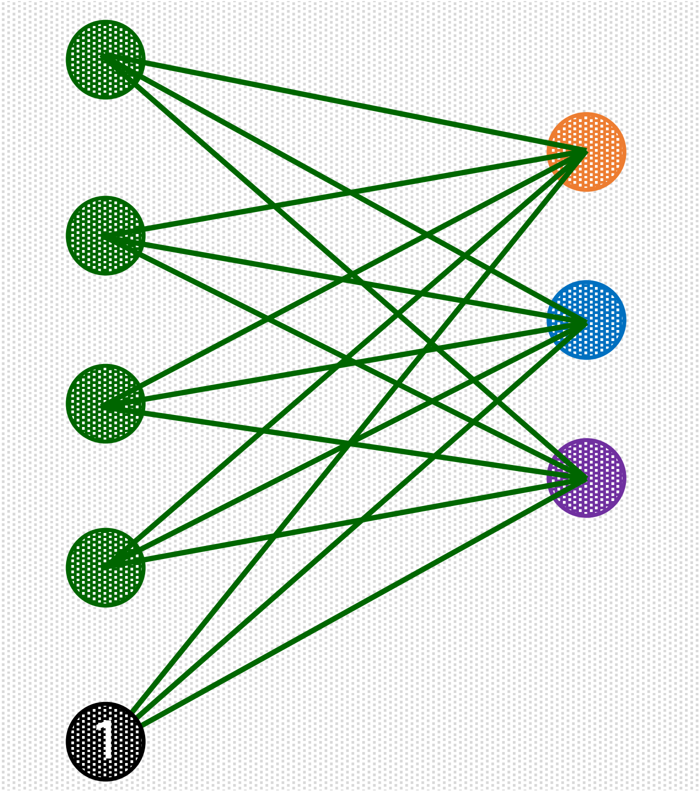


https://colah.github.io/posts/2014-03-NN-Manifolds-Topology/

Affine Layer in Neural Networks

- Fully-connected layers usually involve a bias term. That is why we call it an a ffine layer, but not a linear layer.
- Example: Image with 4 pixels and 3 classes (cat/dog/ship)





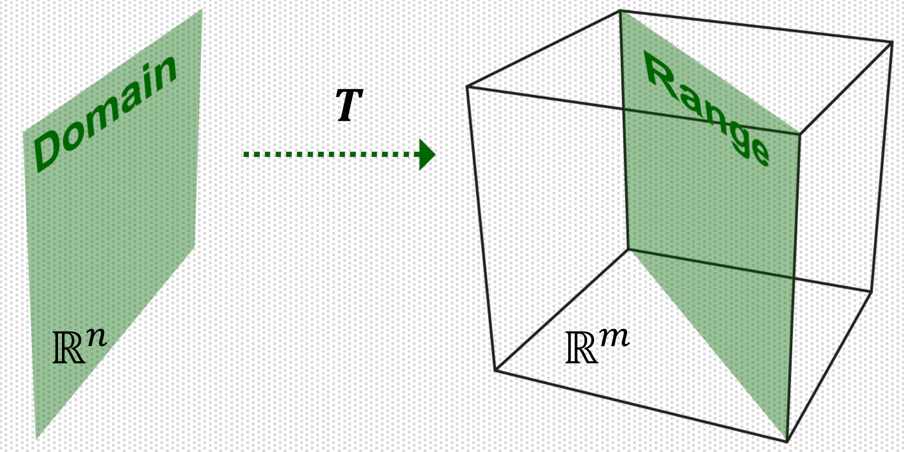
$$\begin{bmatrix} 0.2 & -0.5 & 0.1 & 2 \\ 1.5 & 1.3 & 2.1 & 1 \\ -.2 & 0.3 & 0.7 & -1.3 \end{bmatrix} \begin{bmatrix} 56 \\ 231 \\ 24 \\ 2 \end{bmatrix} + \begin{bmatrix} 1.1 \\ 3.2 \\ -1.2 \end{bmatrix} = \begin{bmatrix} -96.8 \\ 439.9 \\ 71.1 \end{bmatrix}$$

$$= 56 \begin{bmatrix} 0.2 \\ 1.5 \\ -.2 \end{bmatrix} + 231 \begin{bmatrix} -0.5 \\ 1.3 \\ 0.3 \end{bmatrix} + 24 \begin{bmatrix} 0.1 \\ 2.1 \\ 0.7 \end{bmatrix} + 2 \begin{bmatrix} 2 \\ 1 \\ -1.3 \end{bmatrix} + 1 \begin{bmatrix} 1.1 \\ 3.2 \\ -1.2 \end{bmatrix}$$

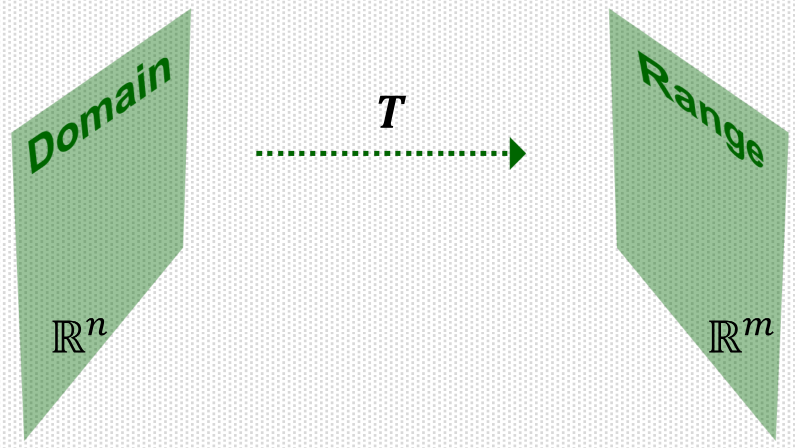
$$= \begin{bmatrix} 0.2 & -0.5 & 0.1 & 2 & 1.1 \\ 1.5 & 1.3 & 2.1 & 1 & 3.2 \\ -.2 & 0.3 & 0.7 & -1.3 & -1.2 \end{bmatrix} \begin{bmatrix} 56 \\ 231 \\ 24 \\ 2 \\ 1 \end{bmatrix}$$

ONTO and ONE-TO-ONE

- **Definition:** A mapping $T : \mathbb{R}^n \to \mathbb{R}^m$ is said to be onto $\mathbb{R}^m$ if each $\mathbf{b} \in \mathbb{R}^m$ is the image of at least one $\mathbf{x} \in \mathbb{R}^n$.
  That is, the range is equal to the co-domain.



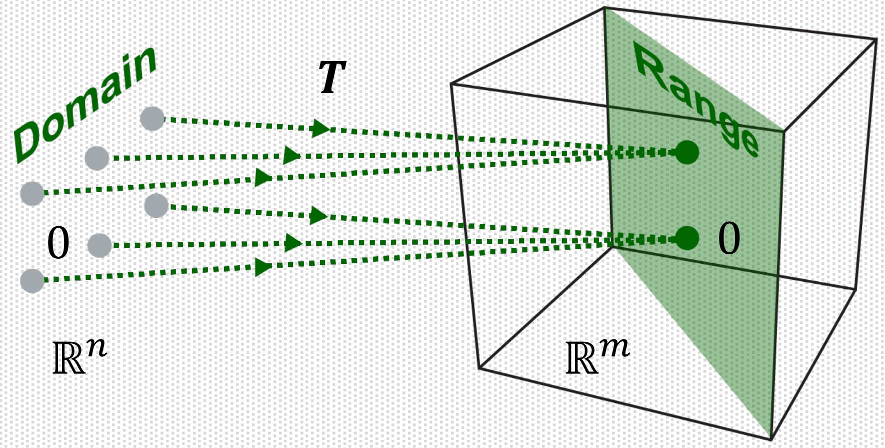
T is NOT onto $\mathbb{R}^m$



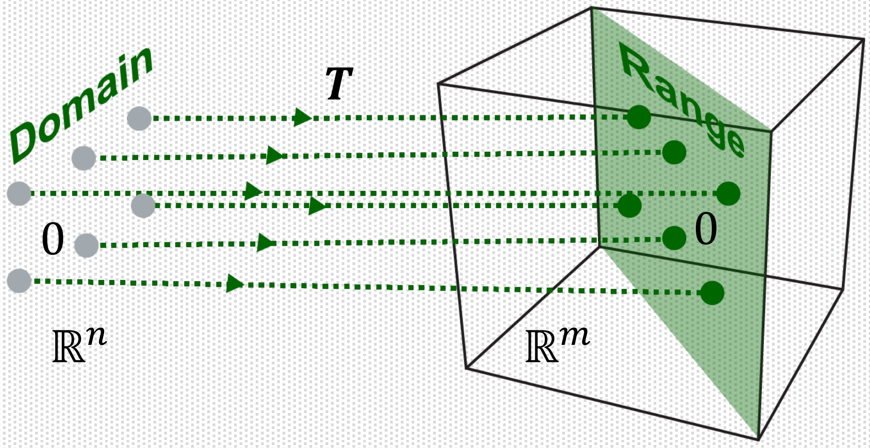
T is onto $\mathbb{R}^m$

ONTO and ONE-TO-ONE

- **Definition:** A mapping $T : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is said to be one-to-one if each $\mathbf{b} \in \mathbb{R}^m$ is the image of at most one $\mathbf{x} \in \mathbb{R}^n$.
  That is, each output vector in the range is mapped by only one input vector, no more than that.



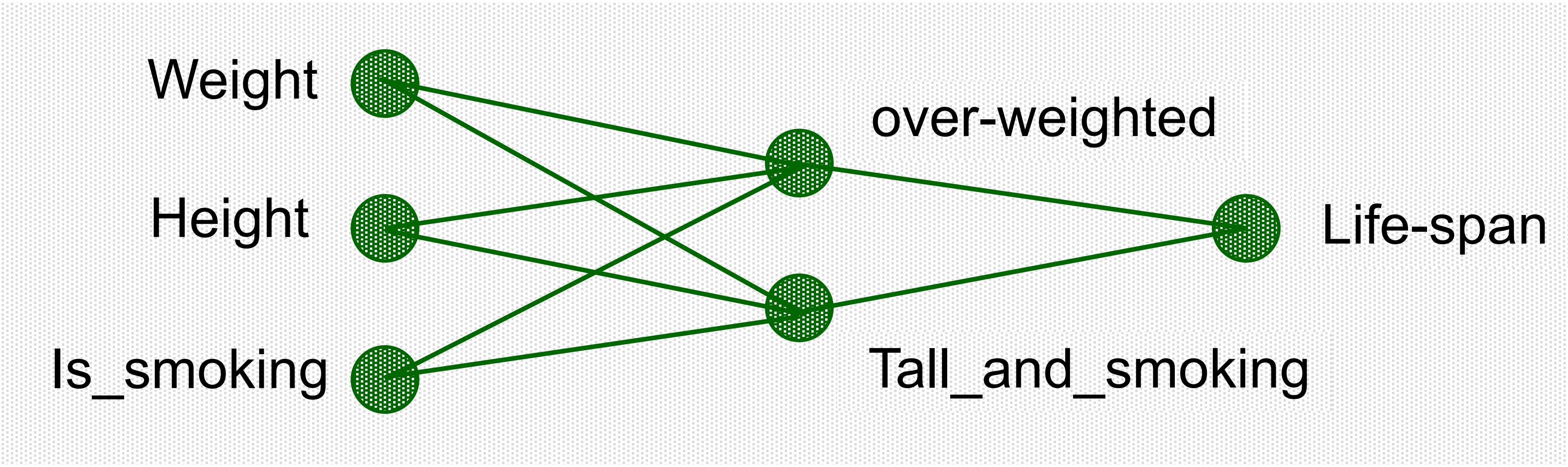
T is NOT one-to-one



T is one-to-one

Neural Network Example

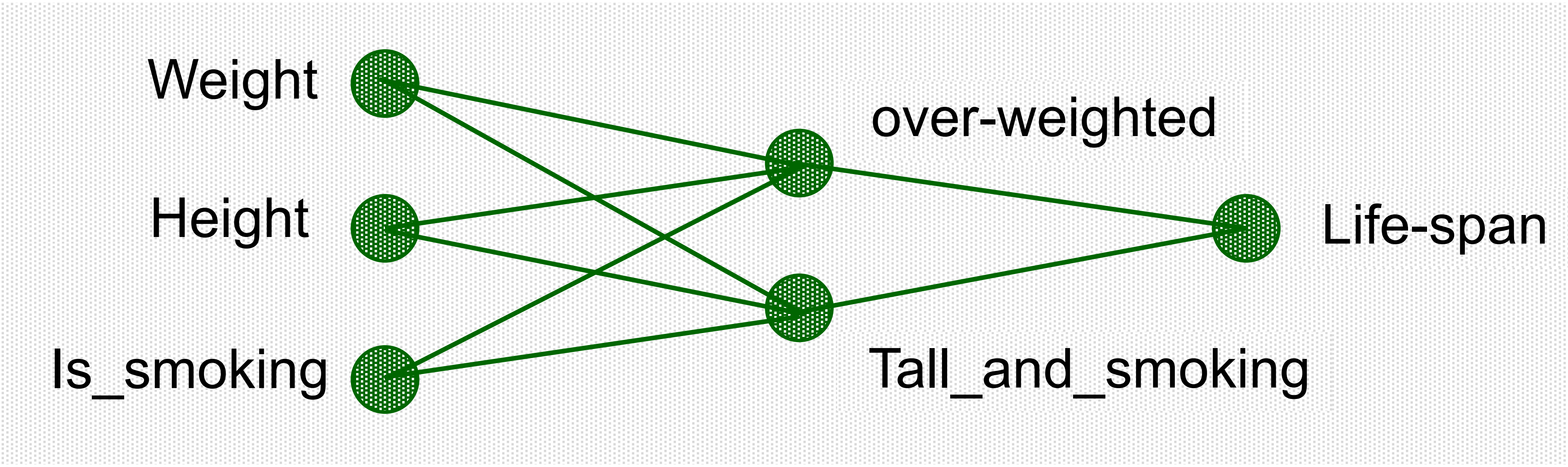
- Fully-connected layers



$$\mathbf{x} \xrightarrow[T_1]{} \mathbf{y} \xrightarrow[T_2]{} \mathbf{z}$$

Neural Network Example: ONE-TO-ONE

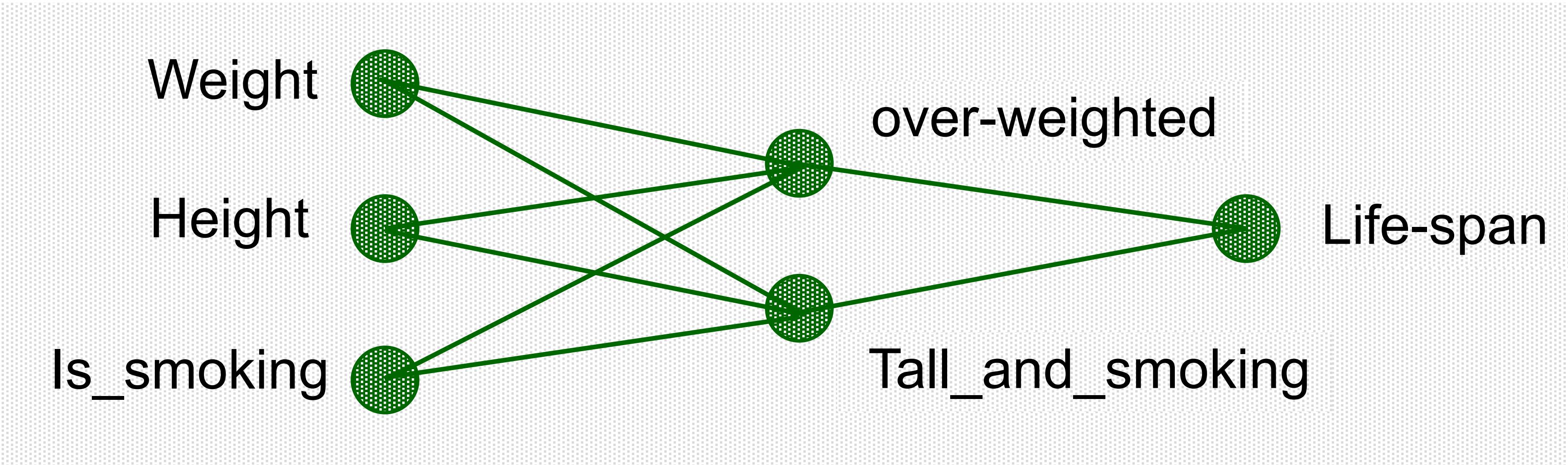
- Will there be many (or unique) people mapped to the same (over_weighted, tall_and_smoking)?



$$\mathbf{x} \xrightarrow[T_1]{} \mathbf{y} \xrightarrow[T_2]{} \mathbf{z}$$

Neural Network Example: ONTO

- Is there any (over_weighted, tall_and_smoking) that does not exist at all?



$$\mathbf{x} \underset{T_1}{\longrightarrow} \mathbf{y} \underset{T_2}{\longrightarrow} \mathbf{z}$$

ONTO and ONE-TO-ONE

- Let $T : \mathbb{R}^n \to \mathbb{R}^m$ be a linear transformation, i.e.,

$$T(\mathbf{x}) = A\mathbf{x} \text{ for all } \mathbf{x} \in \mathbb{R}^n.$$

- T is one-to-one if and only if the columns of A are linearly independent.

- T maps $\mathbb{R}^n$ onto $\mathbb{R}^m$ if and only if the columns of A span $\mathbb{R}^m$.

ONTO and ONE-TO-ONE

- **Example:**

Let $T(\mathbf{x}) = A\mathbf{x} = \begin{bmatrix} 2 & 0 \\ -1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

- Is T one-to-one?
- Does T map $\mathbb{R}^2$ onto $\mathbb{R}^3$?

ONTO and ONE-TO-ONE

- **Example:**

  Let $T(\mathbf{x}) = A\mathbf{x} = \begin{bmatrix} 1 & 4 & 5 \\ 2 & 3 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$

- Is T one-to-one?
- Does T map $\mathbb{R}^3$ onto $\mathbb{R}^2$?