Artificial Intelligence: *Modeling Human Intelligence with Networks*

Jeová Farias Sales Rocha Neto
jeova_farias@brown.edu

What is Intelligence?

What is Intelligence? What is Learning?

Which face is real?

(a)

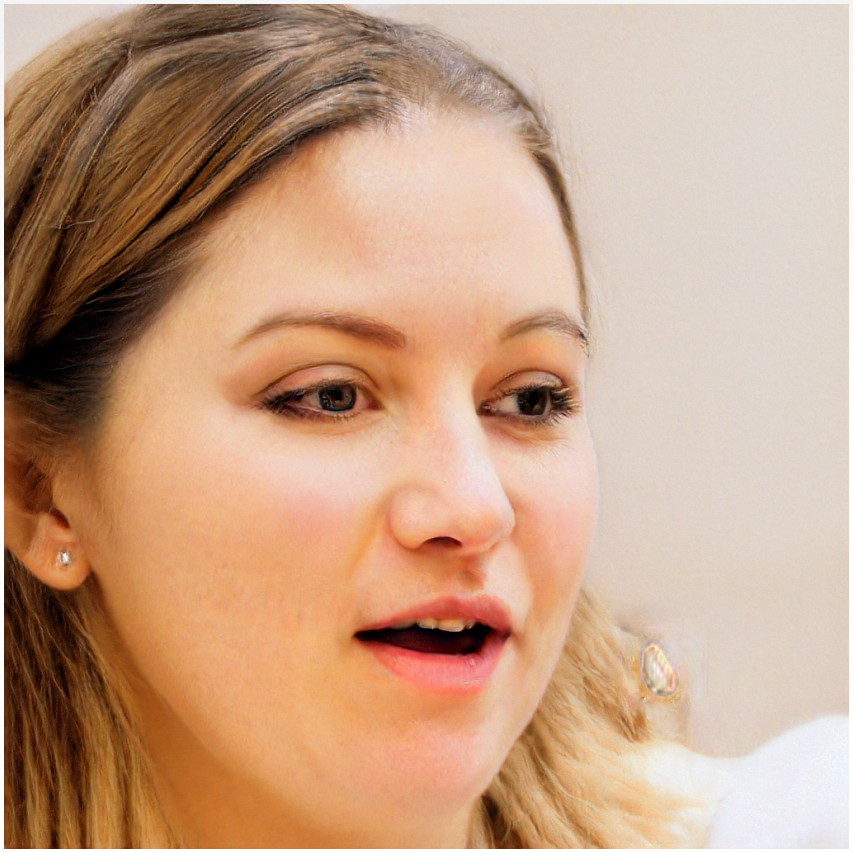
(b)

Which face is real?

(a) FAKE

(b) REAL

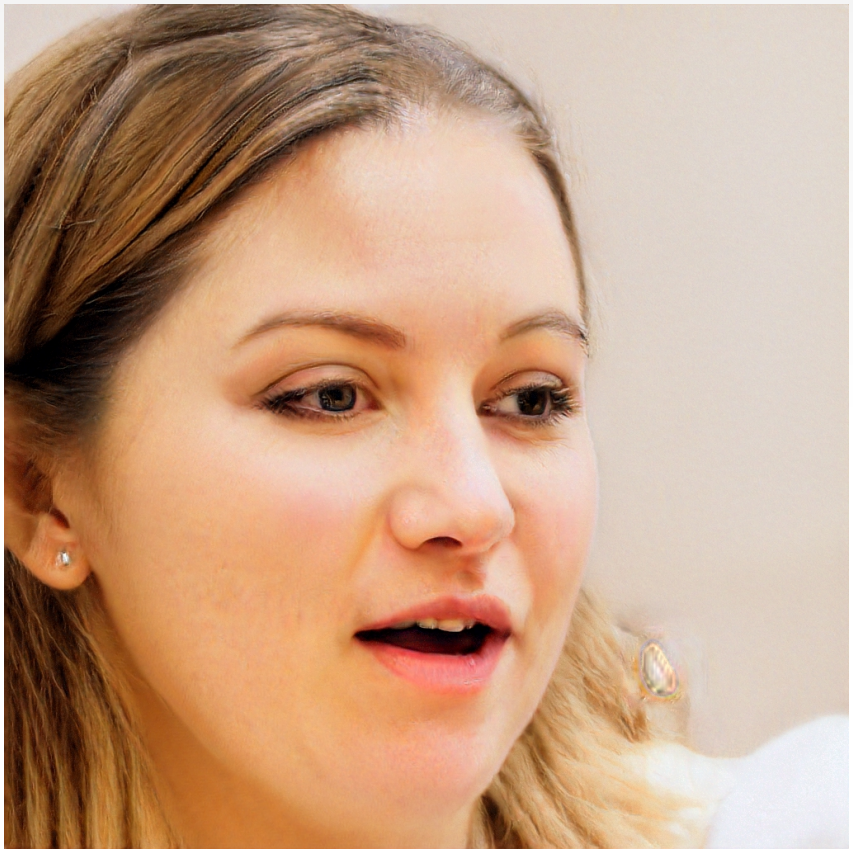
Which face is real?



(a)

(b)

Which face is real?



(a) FAKE

(b) REAL

Which face is real?

(a)

(b)

Which face is real?

(a) REAL

(b) FAKE

Which animal is this?

Class: beaver (0.38 % of confidence)

Class: ice_bear (1.00 % of confidence)

Class: ant (1.00 % of confidence)

Which thing is this?

Class: beer_glass (0.55 % of confidence)

Class: broccoli (1.00 % of confidence)

Class: alp (0.85 % of confidence)

How do you think they do it?

Goals of this class

- Undestand how they do it!

Goals of this class

- Undestand how they do it!
- Have a better idea of what AI truly means.

Goals of this class

- Undestand how they do it!
- Have a better idea of what AI truly means.
- Learn some cool applications of AI using networks!

Goals of this class

- Undestand how they do it!
- Have a better idea of what AI truly means.
- Learn some cool applications of AI using networks!
- Learn the basics of many cool topics:
  - Programming / Algorithms / Data structures + Python
  - Linear Algebra/ Calculus.
  - Probability Theory.
  - Learning Theory.
  - Machine Learning / Deep Learning.

Goals of this class

- Undestand how they do it!
- Have a better idea of what AI truly means.
- Learn some cool applications of AI using networks!
- Learn the basics of many cool topics:
  - Programming / Algorithms / Data structures + Python
  - Linear Algebra/ Calculus.
  - Probability Theory.
  - Learning Theory.
  - Machine Learning / Deep Learning.
- And

Goals of this class

- Undestand how they do it!
- Have a better idea of what AI truly means.
- Learn some cool applications of AI using networks!
- Learn the basics of many cool topics:
  - Programming / Algorithms / Data structures + Python
  - Linear Algebra/ Calculus.
  - Probability Theory.
  - Learning Theory.
  - Machine Learning / Deep Learning.
- And

How is it gonna work?

Monday-Thursday:

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.
- Time allowing: texts/videos + discussions.

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.
- Time allowing: texts/videos + discussions.
- Daily homeworks (sometimes during class).

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.
- Time allowing: texts/videos + discussions.
- Daily homeworks (sometimes during class).

Fridays:

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.
- Time allowing: texts/videos + discussions.
- Daily homeworks (sometimes during class).

Fridays: Group Presentations

- Teams of 3 or 4

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.
- Time allowing: texts/videos + discussions.
- Daily homeworks (sometimes during class).

Fridays: Group Presentations

- Teams of 3 or 4 (I'll choose them ☺).
- Same team always!

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.
- Time allowing: texts/videos + discussions.
- Daily homeworks (sometimes during class).

Fridays: Group Presentations

- Teams of 3 or 4 (I'll choose them ☺).
- Same team always!
- Different Topics about AI.

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.
- Time allowing: texts/videos + discussions.
- Daily homeworks (sometimes during class).

Fridays: Group Presentations

- Teams of 3 or 4 (I'll choose them ☺).
- Same team always!
- Different Topics about AI.
- 15/20 min presentations + 5 min questions.

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.
- Time allowing: texts/videos + discussions.
- Daily homeworks (sometimes during class).

Fridays: Group Presentations

- Teams of 3 or 4 (I'll choose them ☺).
- Same team always!
- Different Topics about AI.
- 15/20 min presentations + 5 min questions.
- Last one is a project.

How is it gonna work?

Monday-Thursday: CLASSES!!

- Theory + Programming Topic *(We will take our time)*.
- Time allowing: texts/videos + discussions.
- Daily homeworks (sometimes during class).

Fridays: Group Presentations

- Teams of 3 or 4 (I'll choose them ☺).
- Same team always!
- Different Topics about AI.
- 15/20 min presentations + 5 min questions.
- Last one is a project.
- More details later today.

How is it gonna work?

Grading

- Grade breakdown: HM (20%) + 1st and 2nd presentations (40%) + last presentation (40%).

How is it gonna work?

Grading

- Grade breakdown: HM (20%) + 1st and 2nd presentations (40%) + last presentation (40%).
- Homeworks (Let's see if it works):
  - Theory + Programming.

How is it gonna work?

Grading

- Grade breakdown: HM (20%) + 1st and 2nd presentations (40%) + last presentation (40%).
- Homeworks (Let's see if it works):
  - Theory + Programming.
  - I'll ask you something about it the next day, **individually**.

How is it gonna work?

Grading

- Grade breakdown: HM (20%) + 1st and 2nd presentations (40%) + last presentation (40%).
- Homeworks (Let's see if it works):
  - Theory + Programming.
  - I'll ask you something about it the next day, **individually**.
  - One question about the class from the day before + 1 or 2 questions about HW.

How is it gonna work?

Grading

- Grade breakdown: HM (20%) + 1st and 2nd presentations (40%) + last presentation (40%).
- Homeworks (Let's see if it works):
  - Theory + Programming.
  - I'll ask you something about it the next day, **individually**.
  - One question about the class from the day before + 1 or 2 questions about HW.
  - Full grade, half grade or none.

How is it gonna work?

Grading

- Grade breakdown: HM (20%) + 1st and 2nd presentations (40%) + last presentation (40%).
- Homeworks (Let's see if it works):
  - Theory + Programming.
  - I'll ask you something about it the next day, **individually**.
  - One question about the class from the day before + 1 or 2 questions about HW.
  - Full grade, half grade or none.
- Presentations:
  - Teams should turning in papers (5-7 pages) for the last presentation and prepare slides for all presentations.

How is it gonna work?

Grading

- Grade breakdown: HM (20%) + 1st and 2nd presentations (40%) + last presentation (40%).
- Homeworks (Let's see if it works):
  - Theory + Programming.
  - I'll ask you something about it the next day, **individually**.
  - One question about the class from the day before + 1 or 2 questions about HW.
  - Full grade, half grade or none.
- Presentations:
  - Teams should turning in papers (5-7 pages) for the last presentation and prepare slides for all presentations.
  - Grading: 0-10 (paper + slides + presentation).

How is it gonna work?

Grading

- Grade breakdown: HM (20%) + 1st and 2nd presentations (40%) + last presentation (40%).
- Homeworks (Let's see if it works):
  - Theory + Programming.
  - I'll ask you something about it the next day, **individually**.
  - One question about the class from the day before + 1 or 2 questions about HW.
  - Full grade, half grade or none.
- Presentations:
  - Teams should turning in papers (5-7 pages) for the last presentation and prepare slides for all presentations.
  - Grading: 0-10 (paper + slides + presentation).
  - Grades are given individually, but its considers team work.

How is it gonna work?

Grading

- Grade breakdown: HM (20%) + 1st and 2nd presentations (40%) + last presentation (40%).
- Homeworks (Let's see if it works):
  - Theory + Programming.
  - I'll ask you something about it the next day, **individually**.
  - One question about the class from the day before + 1 or 2 questions about HW.
  - Full grade, half grade or none.
- Presentations:
  - Teams should turning in papers (5-7 pages) for the last presentation and prepare slides for all presentations.
  - Grading: 0-10 (paper + slides + presentation).
  - Grades are given individually, but its considers team work.
  - Other teams improve their grades if they ask questions.
- *Things can change, if necessary.*

How is it gonna work?

Textbook

- There is no textbook for this course. The readings are provided.
- Laptops will be extremely useful! *(Let me know if you didn't bring one).*

How is it gonna work?

Textbook

- There is no textbook for this course. The readings are provided.
- Laptops will be extremely useful! *(Let me know if you didn't bring one).*

Rules here

- During lectures, no talk and no cellphone!

How is it gonna work?

Textbook

- There is no textbook for this course. The readings are provided.
- Laptops will be extremely useful! *(Let me know if you didn't bring one).*

Rules here

- During lectures, no talk and no cellphone!
- No laptop lid open, unless for programming or specified otherwise (take notes on paper).

How is it gonna work?

Textbook

- There is no textbook for this course. The readings are provided.
- Laptops will be extremely useful! *(Let me know if you didn't bring one).*

Rules here

- During lectures, no talk and no cellphone!
- No laptop lid open, unless for programming or specified otherwise (take notes on paper).
- Bad etiquette will reflect on the HW grade (for the worse).

How is it gonna work?

Textbook

- There is no textbook for this course. The readings are provided.
- Laptops will be extremely useful! *(Let me know if you didn't bring one).*

Rules here

- During lectures, no talk and no cellphone!
- No laptop lid open, unless for programming or specified otherwise (take notes on paper).
- Bad etiquette will reflect on the HW grade (for the worse).
- Asking questions and participating in class will also reflect on grade (for the good ☺).

How is it gonna work?

Textbook

- There is no textbook for this course. The readings are provided.
- Laptops will be extremely useful! *(Let me know if you didn't bring one).*

Rules here

- During lectures, no talk and no cellphone!
- No laptop lid open, unless for programming or specified otherwise (take notes on paper).
- Bad etiquette will reflect on the HW grade (for the worse).
- Asking questions and participating in class will also reflect on grade (for the good ☺).

How is it gonna work?

Final thoughts

- This course is challenging (for me and for you)!

How is it gonna work?

Final thoughts

- This course is challenging (for me and for you)!
- I don't mind if we don't follow the schedule strictly.

How is it gonna work?

Final thoughts

- This course is challenging (for me and for you)!
- I don't mind if we don't follow the schedule strictly.
- Office Hours – 1pm to 2:30pm, Room A9/The Rock (Kinda TBD...).

How is it gonna work?

Final thoughts

- This course is challenging (for me and for you)!
- I don't mind if we don't follow the schedule strictly.
- Office Hours – 1pm to 2:30pm, Room A9/The Rock (Kinda TBD...).
- Personal Interests/Small Research Project?

How is it gonna work?

Final thoughts

- This course is challenging (for me and for you)!
- I don't mind if we don't follow the schedule strictly.
- Office Hours – 1pm to 2:30pm, Room A9/The Rock (Kinda TBD...).
- Personal Interests/Small Research Project? – Come talk to me outside the class.

How is it gonna work?

Final thoughts

- This course is challenging (for me and for you)!
- I don't mind if we don't follow the schedule strictly.
- Office Hours – 1pm to 2:30pm, Room A9/The Rock (Kinda TBD...).
- Personal Interests/Small Research Project? – Come talk to me outside the class.
- My only goal here is to have you understand the content and enjoy the class!

Course Schedule

Week 1	Theoretical Topic	Programming Topic
Monday	- Introduction to AI - Neuron model - Vectors, dot product, geometry	- Intro to programming - For loops, while loops, if statements - Data structures
Tuesday	- Classification task, Linear separators - Loss functions - What is learning?	- Uploading data - Data visualization - Plotting functions
Wednesday	- Perceptron algorithm - Geometry of algorithm - Convergence results	- Implement perceptron algorithm
Thursday	- Nonlinearly separable data - Multi-class classification, Matrices - MNIST challenge	- Implement perceptron algorithm - Matrix data structure, Matrix operations - Programming MNIST forward pass
Friday	Group presentations	

Course Schedule

Week 2	Theoretical Topic	Programming Topic
Monday	- Limits - Definition of derivative - Extrema of a differentiable function	- Plotting single variable functions - Plotting single multivariable functions - Numerical limits
Tuesday	- Loss function - Gradient descent - Convexity	- Numerical derivatives - Plotting tangent lines - Gradient descent
Wednesday	- Algorithm for MNIST, - Cross correlation - Multilayer neural network	- MNIST algorithm
Thursday	- Convolutions - Relu, Sigmoids - Max pooling	- Convolutions in images
Friday	Group presentations	

Course Schedule

Week 3	Theoretical Topic	Programming Topic
Monday	- AlexNet, RCNN - Dropout - Batch size	- TensorFlow / Keras
Tuesday	- Sample space - Probability distribution, Random variable - Expectation, Independence	- Generating random numbers - Plot distribution and histograms - Law of large number simulation
Wednesday	- Conditional probability - Markov chains - Graphs, Random walk	- Construct and visualize network - Simulate random walk - Implement random walk on map
Thursday	- Markov model for sentence generation - Applications of markov chains i.e. QMR network, page rank, games	- Random sentence generator
Friday	Group presentations / Programming project due	

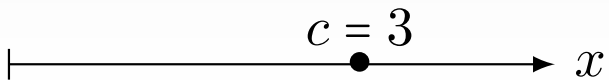
Now, let the classes start!

Basics of Linear Algebra

Basics of Linear Algebra

Scalars

- Any real number, $c \in \mathbb{R}$
- Operations: Addition, multiplication (subtraction, division).
- Graphical representation:



Basics of Linear Algebra

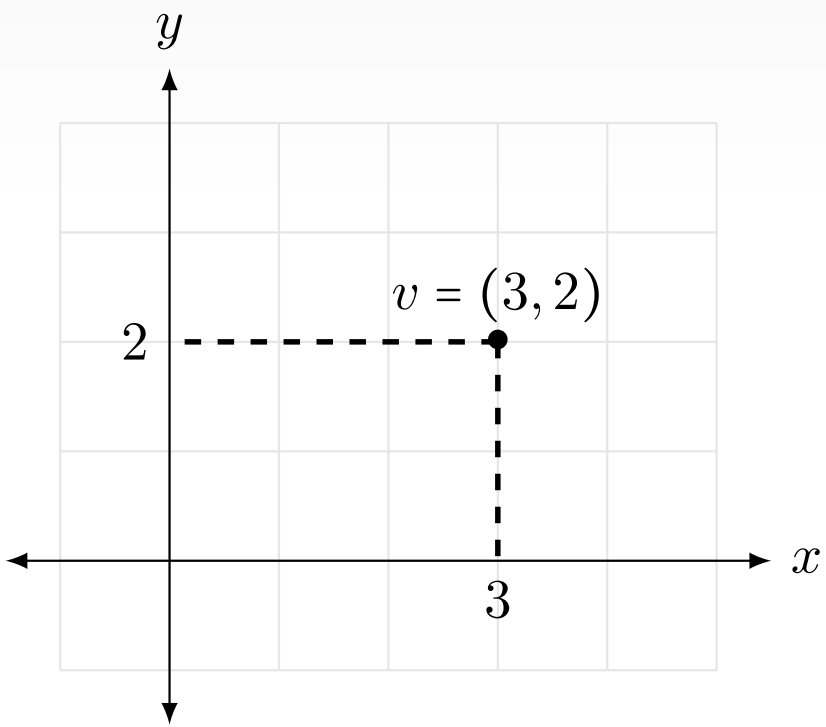
Vectors

- An ordered set of n real numbers, $v \in \mathbb{R}^n$

Basics of Linear Algebra

Vectors

- An ordered set of n real numbers, $v \in \mathbb{R}^n$ (or integers $v \in \mathbb{Z}^n$).
- Graphical representation:

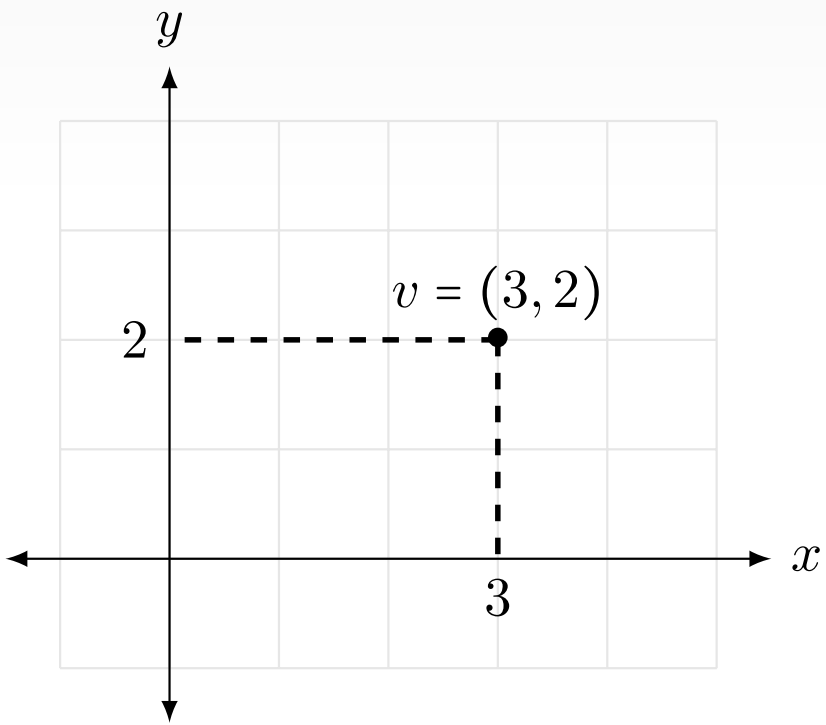


(a) A point in $\mathbb{R}^2$

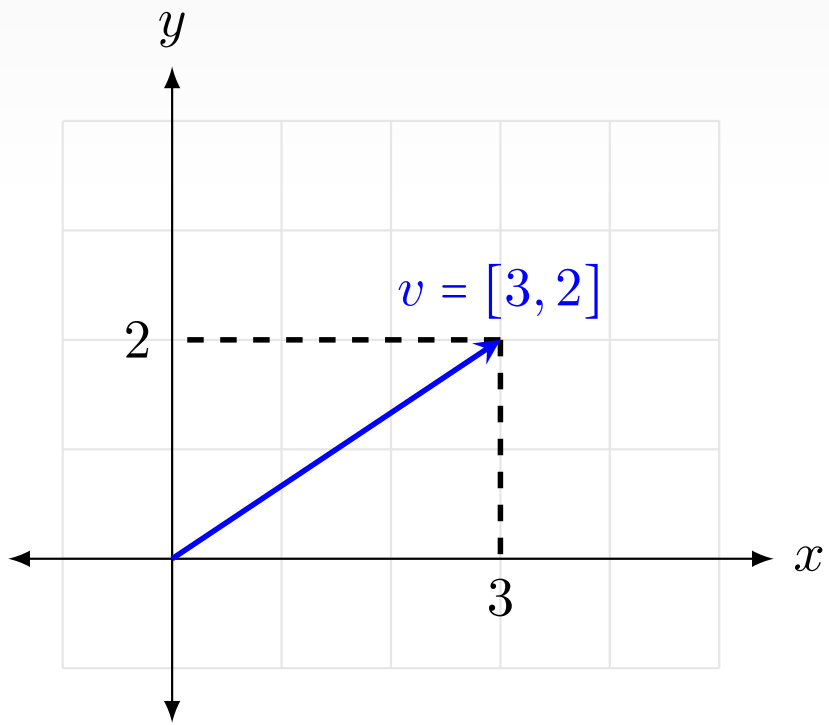
Basics of Linear Algebra

Vectors

- An ordered set of n real numbers, $v \in \mathbb{R}^n$ (or integers $v \in \mathbb{Z}^n$).
- Graphical representation:



(a) A point in $\mathbb{R}^2$



(b) Direction and magnitude in $\mathbb{R}^2$

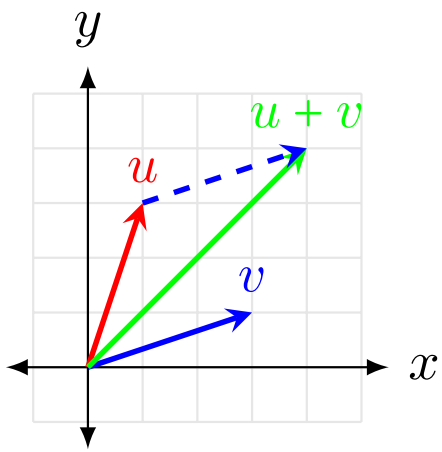
Vectors

- Operations:
  - Addition: if $v = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^n$ and $u = [u_1, u_2, \ldots, u_n] \in \mathbb{R}^n$,

$$v + u = [v_1 + u_1, v_2 + u_2, \ldots, v_n + u_n] \in \mathbb{R}^n$$

  - Multiplication by scalar: if $v = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^n$ and $c \in \mathbb{R}$,

$$cv = [cv_1, cv_2, \ldots, cv_n] \in \mathbb{R}^n$$



(a) $u + v$

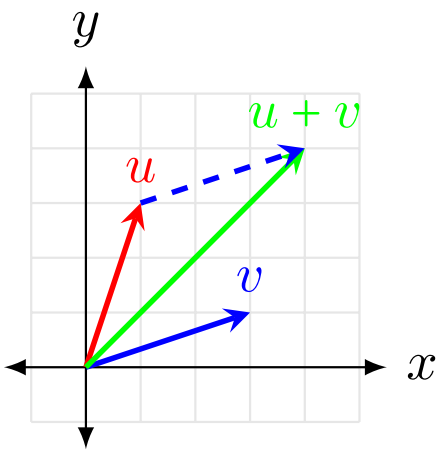
Vectors

- Operations:
  - Addition: if $v = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^n$ and $u = [u_1, u_2, \ldots, u_n] \in \mathbb{R}^n$,

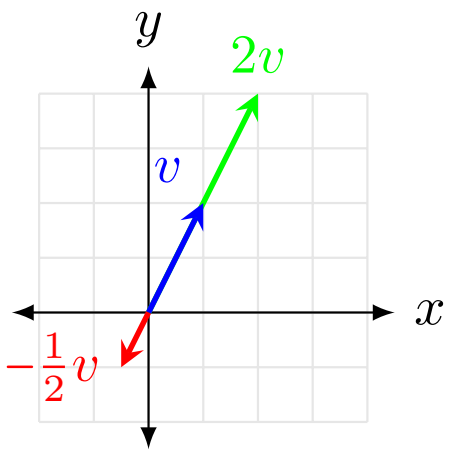
$$v + u = [v_1 + u_1, v_2 + u_2, \ldots, v_n + u_n] \in \mathbb{R}^n$$

  - Multiplication by scalar: if $v = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^n$ and $c \in \mathbb{R}$,

$$cv = [cv_1, cv_2, \ldots, cv_n] \in \mathbb{R}^n$$



(a) $u + v$



(b) cv

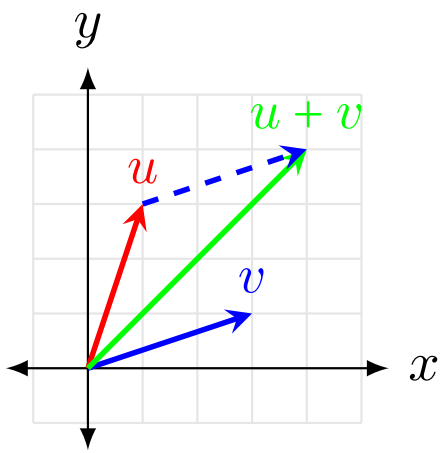
Vectors

- Operations:
  - Addition: if $v = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^n$ and $u = [u_1, u_2, \ldots, u_n] \in \mathbb{R}^n$,

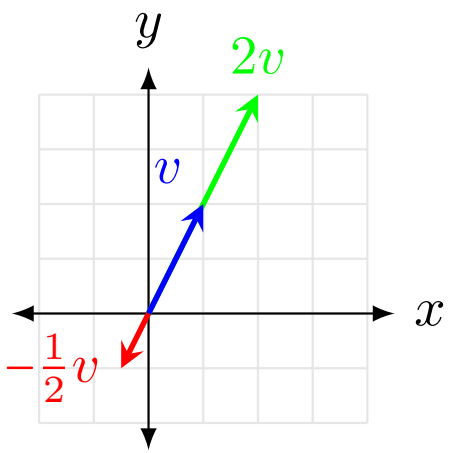
$$v + u = [v_1 + u_1, v_2 + u_2, \ldots, v_n + u_n] \in \mathbb{R}^n$$

  - Multiplication by scalar: if $v = [v_1, v_2, \ldots, v_n] \in \mathbb{R}^n$ and $c \in \mathbb{R}$,

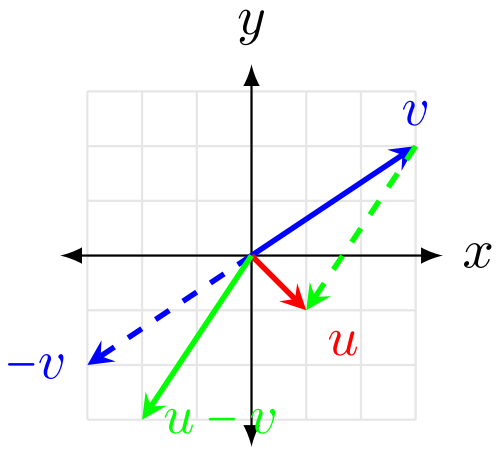
$$cv = [cv_1, cv_2, \ldots, cv_n] \in \mathbb{R}^n$$



(a) $u + v$



(b) cv



(c) $u - v$

Vectors

- The norm of a vector: $v = [v_1, v_2, \ldots, v_n]$,

$$\|\cdot\|\colon \mathbb{R}^n \longrightarrow \mathbb{R}, \quad v \longmapsto \sqrt{\sum_{i=1}^{n} v_i^2} = \sqrt{v_1^2 + v_2^2 + \ldots + v_n^2}$$

Vectors

- The norm of a vector: $v = [v_1, v_2, \ldots, v_n]$,

$$\|\cdot\|: \mathbb{R}^n \longrightarrow \mathbb{R}, \quad v \longmapsto \sqrt{\sum_{i=1}^{n} v_i^2} = \sqrt{v_1^2 + v_2^2 + \ldots + v_n^2}$$

Ex.: $v = [3, 4]$, $\|v\| = \sqrt{3^2 + 4^2} = 5$

Vectors

- The norm of a vector: $v = [v_1, v_2, \ldots, v_n]$,

$$\|\cdot\|: \mathbb{R}^n \longrightarrow \mathbb{R}, \quad v \longmapsto \sqrt{\sum_{i=1}^{n} v_i^2} = \sqrt{v_1^2 + v_2^2 + \ldots + v_n^2}$$

Ex.: $v = [3, 4]$, $\|v\| = \sqrt{3^2 + 4^2} = 5$
Obs.: $\|cv\| = \sqrt{\sum_i (cv_i)^2} = |c|\sqrt{\sum_i v_i^2} = |c|\,\|v\|$

Vectors

- The norm of a vector: $v = [v_1, v_2, \ldots, v_n]$,

$$\|\cdot\|: \mathbb{R}^n \longrightarrow \mathbb{R}, \quad v \longmapsto \sqrt{\sum_{i=1}^{n} v_i^2} = \sqrt{v_1^2 + v_2^2 + \ldots + v_n^2}$$

  Ex.: $v = [3, 4]$, $\|v\| = \sqrt{3^2 + 4^2} = 5$
  Obs.: $\|cv\| = \sqrt{\sum_i (cv_i)^2} = |c|\sqrt{\sum_i v_i^2} = |c|\,\|v\|$

- Vector Normalization:

$$v \longmapsto \frac{v}{\|v\|}$$

  Ex.: $v = [3, 4]$, $v' = \left[\frac{3}{5}, \frac{4}{5}\right]$

Vectors

- The norm of a vector: $v = [v_1, v_2, \ldots, v_n]$,

$$\|\cdot\|: \mathbb{R}^n \longrightarrow \mathbb{R}, \quad v \longmapsto \sqrt{\sum_{i=1}^{n} v_i^2} = \sqrt{v_1^2 + v_2^2 + \ldots + v_n^2}$$

Ex.: $v = [3, 4]$, $\|v\| = \sqrt{3^2 + 4^2} = 5$
Obs.: $\|cv\| = \sqrt{\sum_i (cv_i)^2} = |c|\sqrt{\sum_i v_i^2} = |c|\,\|v\|$

- Vector Normalization:

$$v \longmapsto \frac{v}{\|v\|}$$

Ex.: $v = [3, 4]$, $v' = \left[\frac{3}{5}, \frac{4}{5}\right]$
Obs.:

$$\left\| \frac{v}{\|v\|} \right\| = \frac{\|v\|}{\|v\|} = 1$$

Vectors

- The norm of a vector: $v = [v_1, v_2, \ldots, v_n]$,

$$\|\cdot\|: \mathbb{R}^n \longrightarrow \mathbb{R}, \quad v \longmapsto \sqrt{\sum_{i=1}^{n} v_i^2} = \sqrt{v_1^2 + v_2^2 + \ldots + v_n^2}$$

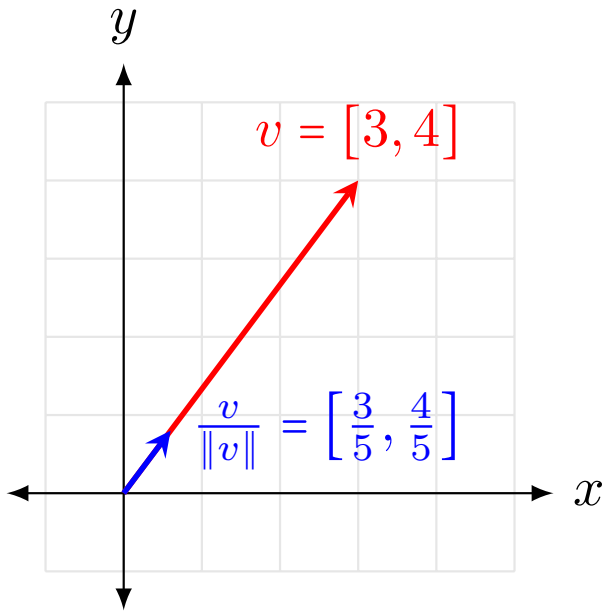
Ex.: $v = [3, 4]$, $\|v\| = \sqrt{3^2 + 4^2} = 5$
Obs.: $\|cv\| = \sqrt{\sum_i (cv_i)^2} = |c|\sqrt{\sum_i v_i^2} = |c|\,\|v\|$

- Vector Normalization:

$$v \longmapsto \frac{v}{\|v\|}$$

Ex.: $v = [3, 4]$, $v' = \left[\frac{3}{5}, \frac{4}{5}\right]$
Obs.:

$$\left\| \frac{v}{\|v\|} \right\| = \frac{\|v\|}{\|v\|} = 1$$



Vectors

- Vector Normalization to (sum) to 1:

$$v \longmapsto \frac{v}{\sum_i v_i}$$

Ex.: $v = [1, 2, 3, 4]$, $v' = [0.1, 0.2, 0.3, 0.4]$

Vectors

- Vector Normalization to (sum) to 1:

$$v \longmapsto \frac{v}{\sum_i v_i}$$

Ex.: $v = [1, 2, 3, 4]$, $v' = [0.1, 0.2, 0.3, 0.4]$
Obs.: $\sum_i v'_i = 1$.

Vectors

- Vector Normalization to (sum) to 1:

$$v \longmapsto \frac{v}{\sum_i v_i}$$

Ex.: $v = [1, 2, 3, 4]$, $v' = [0.1, 0.2, 0.3, 0.4]$
Obs.: $\sum_i v'_i = 1$. *Where do we see it?*

Vectors

- Vector Normalization to (sum) to 1:

$$v \longmapsto \frac{v}{\sum_i v_i}$$

Ex.: $v = [1, 2, 3, 4]$, $v' = [0.1, 0.2, 0.3, 0.4]$
Obs.: $\sum_i v'_i = 1$. *Where do we see it?*

- **Dot product**

$$(v, u) \in \mathbb{R}^n \longmapsto v \cdot u = \sum_{i=1}^{n} v_i u_i = v_1 u_1 + v_2 u_2 + \ldots + v_n u_n$$

Vectors

- Vector Normalization to (sum) to 1:

$$v \longmapsto \frac{v}{\sum_i v_i}$$

Ex.: $v = [1, 2, 3, 4]$, $v' = [0.1, 0.2, 0.3, 0.4]$
Obs.: $\sum_i v'_i = 1$. *Where do we see it?*

- **Dot product**

$$(v, u) \in \mathbb{R}^n \longmapsto v \cdot u = \sum_{i=1}^{n} v_i u_i = v_1 u_1 + v_2 u_2 + \ldots + v_n u_n$$

Ex.: $v = [1, 2, 3]$, $u = [4, 5, 6]$, then $v \cdot u = 4 + 10 + 18 = 32$.

Vectors

- Vector Normalization to (sum) to 1:

$$v \longmapsto \frac{v}{\sum_i v_i}$$

Ex.: $v = [1, 2, 3, 4]$, $v' = [0.1, 0.2, 0.3, 0.4]$
Obs.: $\sum_i v'_i = 1$. *Where do we see it?*

- **Dot product**

$$(v, u) \in \mathbb{R}^n \longmapsto v \cdot u = \sum_{i=1}^{n} v_i u_i = v_1 u_1 + v_2 u_2 + \ldots + v_n u_n$$

Ex.: $v = [1, 2, 3]$, $u = [4, 5, 6]$, then $v \cdot u = 4 + 10 + 18 = 32$.
Obs. 1: Important properties:

$$v \cdot u = u \cdot v, \quad (cv) \cdot u = c(v \cdot u), \quad (v + u) \cdot w = v \cdot w + u \cdot w \cdot w$$

Vectors

- Vector Normalization to (sum) to 1:

$$v \longmapsto \frac{v}{\sum_i v_i}$$

Ex.: $v = [1, 2, 3, 4]$, $v' = [0.1, 0.2, 0.3, 0.4]$
Obs.: $\sum_i v'_i = 1$. *Where do we see it?*

- **Dot product**

$$(v, u) \in \mathbb{R}^n \longmapsto v \cdot u = \sum_{i=1}^{n} v_i u_i = v_1 u_1 + v_2 u_2 + \ldots + v_n u_n$$

Ex.: $v = [1, 2, 3]$, $u = [4, 5, 6]$, then $v \cdot u = 4 + 10 + 18 = 32$.
Obs. 1: Important properties:

$$v \cdot u = u \cdot v, \quad (cv) \cdot u = c(v \cdot u), \quad (v + u) \cdot w = v \cdot w + u \cdot w \cdot w$$

Obs. 2: Using the Law of Cosines, we also have:

$$v \cdot u = \|v\| \|u\| \cos(\theta)$$

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$:

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v:

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

$$\|v\| = \sqrt{\sum_i v_i^2} = \sqrt{v \cdot v}$$

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

$$\|v\| = \sqrt{\sum_i v_i^2} = \sqrt{v \cdot v}$$

  - Weighted Average of $v \in \mathbb{R}^n$ with weights $w \in \mathbb{R}^n$.

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

$$\|v\| = \sqrt{\sum_i v_i^2} = \sqrt{v \cdot v}$$

  - Weighted Average of $v \in \mathbb{R}^n$ with weights $w \in \mathbb{R}^n$.

$$\frac{1}{\sum_i w_i} \sum_i w_i v_i = \frac{w \cdot v}{[1, 1, \ldots, 1] \cdot w}$$

- Orthogonality
  - When is $u \cdot v = 0$?

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

$$\|v\| = \sqrt{\sum_i v_i^2} = \sqrt{v \cdot v}$$

  - Weighted Average of $v \in \mathbb{R}^n$ with weights $w \in \mathbb{R}^n$.

$$\frac{1}{\sum_i w_i} \sum_i w_i v_i = \frac{w \cdot v}{[1, 1, \ldots, 1] \cdot w}$$

- Orthogonality
  - When is $u \cdot v = 0$?
  - Two cases:
    - $u = [0, 0, \ldots, 0]$ or $v = [0, 0, \ldots, 0]$

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

$$\|v\| = \sqrt{\sum_i v_i^2} = \sqrt{v \cdot v}$$

  - Weighted Average of $v \in \mathbb{R}^n$ with weights $w \in \mathbb{R}^n$.

$$\frac{1}{\sum_i w_i} \sum_i w_i v_i = \frac{w \cdot v}{[1, 1, \ldots, 1] \cdot w}$$

- Orthogonality
  - When is $u \cdot v = 0$?
  - Two cases:
    - $u = [0, 0, \ldots, 0]$ or $v = [0, 0, \ldots, 0]$ (not interesting)

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

$$\|v\| = \sqrt{\sum_i v_i^2} = \sqrt{v \cdot v}$$

  - Weighted Average of $v \in \mathbb{R}^n$ with weights $w \in \mathbb{R}^n$.

$$\frac{1}{\sum_i w_i} \sum_i w_i v_i = \frac{w \cdot v}{[1, 1, \ldots, 1] \cdot w}$$

- Orthogonality
  - When is $u \cdot v = 0$?
  - Two cases:
    - $u = [0, 0, \ldots, 0]$ or $v = [0, 0, \ldots, 0]$ (not interesting)
    - $u \cdot v = \|v\| \|u\| \cos(\theta) = 0$

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

$$\|v\| = \sqrt{\sum_i v_i^2} = \sqrt{v \cdot v}$$

  - Weighted Average of $v \in \mathbb{R}^n$ with weights $w \in \mathbb{R}^n$.

$$\frac{1}{\sum_i w_i} \sum_i w_i v_i = \frac{w \cdot v}{[1, 1, \ldots, 1] \cdot w}$$

- Orthogonality
  - When is $u \cdot v = 0$?
  - Two cases:
    - $u = [0, 0, \ldots, 0]$ or $v = [0, 0, \ldots, 0]$ (not interesting)
    - $u \cdot v = \|v\| \|u\| \cos(\theta) = 0 \rightarrow \cos(\theta) = 0$

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

$$\|v\| = \sqrt{\sum_i v_i^2} = \sqrt{v \cdot v}$$

  - Weighted Average of $v \in \mathbb{R}^n$ with weights $w \in \mathbb{R}^n$.

$$\frac{1}{\sum_i w_i} \sum_i w_i v_i = \frac{w \cdot v}{[1, 1, \ldots, 1] \cdot w}$$

- Orthogonality
  - When is $u \cdot v = 0$?
  - Two cases:
    - $u = [0, 0, \ldots, 0]$ or $v = [0, 0, \ldots, 0]$ (not interesting)
    - $u \cdot v = \|v\| \|u\| \cos(\theta) = 0 \rightarrow \cos(\theta) = 0 \rightarrow \theta =$

Vectors

- Examples of dot product:
  - Sum of the elements in $v \in \mathbb{R}^n$: $\sum_i v_i = [1, 1, \ldots, 1] \cdot v$
  - Average of the elements in v: $\frac{1}{N} \sum_i v_i = \frac{1}{N}([1, 1, \ldots, 1] \cdot v)$
  - Norm of v.

$$\|v\| = \sqrt{\sum_i v_i^2} = \sqrt{v \cdot v}$$

  - Weighted Average of $v \in \mathbb{R}^n$ with weights $w \in \mathbb{R}^n$.

$$\frac{1}{\sum_i w_i} \sum_i w_i v_i = \frac{w \cdot v}{[1, 1, \ldots, 1] \cdot w}$$

- Orthogonality
  - When is $u \cdot v = 0$?
  - Two cases:
    - $u = [0, 0, \ldots, 0]$ or $v = [0, 0, \ldots, 0]$ (not interesting)
    - $u \cdot v = \|v\| \|u\| \cos(\theta) = 0 \rightarrow \cos(\theta) = 0 \rightarrow \theta = 90°$ (or $\frac{c\pi}{4}, \forall c \in \mathbb{Z}$).

Vectors

Matrices

- A set of real vectors of the same length ($M \in \mathbb{R}^{n \times m}$, n rows, m columns),

Vectors

Matrices

- A set of real vectors of the same length ($M \in \mathbb{R}^{n \times m}$, n rows, m columns), *for now*.

Vectors

Matrices

- A set of real vectors of the same length ($M \in \mathbb{R}^{n \times m}$, n rows, m columns), *for now*.
- The composition of vectors $\left\{v^1, v^2, \ldots, v^m\right\}$, $v^i \in \mathbb{R}^n$, to form a matrix can be done in two ways:
  - By stacking them:

$$M = \begin{bmatrix} — & v^1 & — \\ — & v^2 & — \\ — & \vdots & — \\ — & v^m & — \end{bmatrix} = \begin{bmatrix} v_1^1 & v_2^1 & \ldots & v_n^1 \\ v_1^2 & v_2^2 & \ldots & v_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ v_1^m & v_2^m & \ldots & v_n^m \end{bmatrix} \in \mathbb{R}^{m \times n}$$

Vectors

Matrices

- A set of real vectors of the same length ($M \in \mathbb{R}^{n \times m}$, n rows, m columns), *for now*.
- The composition of vectors $\{v^1, v^2, \ldots, v^m\}$, $v^i \in \mathbb{R}^n$, to form a matrix can be done in two ways:
  - By stacking them:

$$M = \begin{bmatrix} — & v^1 & — \\ — & v^2 & — \\ — & \vdots & — \\ — & v^m & — \end{bmatrix} = \begin{bmatrix} v_1^1 & v_2^1 & \ldots & v_n^1 \\ v_1^2 & v_2^2 & \ldots & v_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ v_1^m & v_2^m & \ldots & v_n^m \end{bmatrix} \in \mathbb{R}^{m \times n}$$

  - By concatenating them:

$$H = \begin{bmatrix} | & | & | & | \\ v_1 & v_2 & \cdots & v_2 \\ | & | & | & | \end{bmatrix} = \begin{bmatrix} v_1^1 & v_1^2 & \ldots & v_1^m \\ v_2^1 & v_2^2 & \ldots & v_2^m \\ \vdots & \vdots & \ddots & \vdots \\ v_n^1 & v_n^2 & \ldots & v_n^m \end{bmatrix} \in \mathbb{R}^{n \times m}$$

Vectors

Matrices

- A set of real vectors of the same length ($M \in \mathbb{R}^{n \times m}$, n rows, m columns), *for now*.
- The composition of vectors $\{v^1, v^2, \ldots, v^m\}$, $v^i \in \mathbb{R}^n$, to form a matrix can be done in two ways:
  - By stacking them:

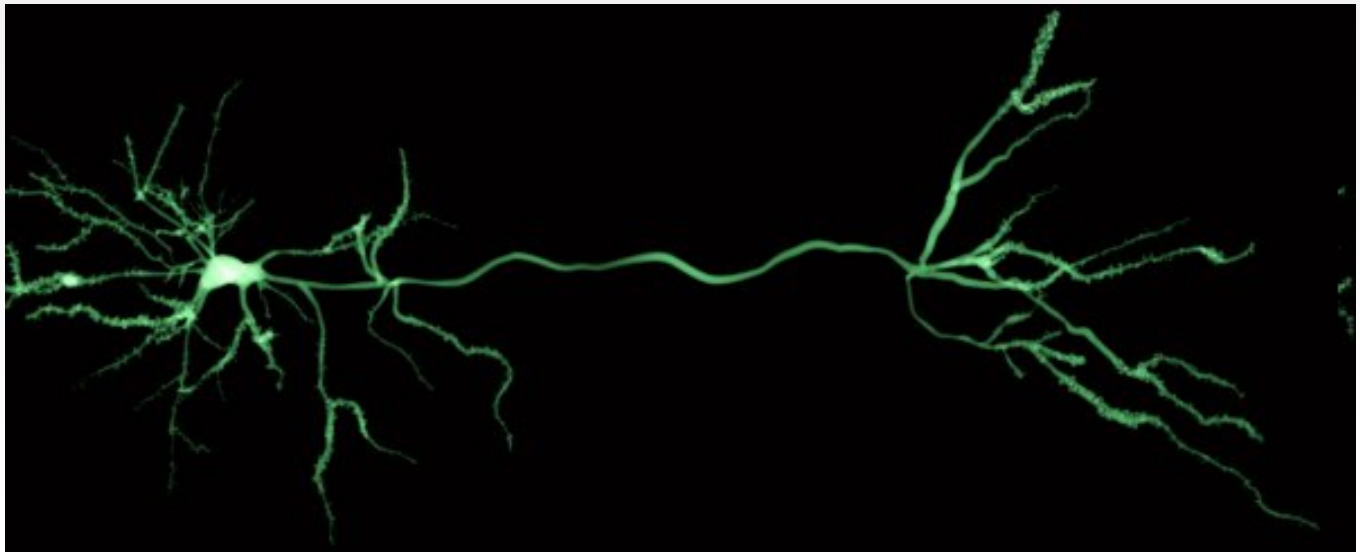
$$M = \begin{bmatrix} — & v^1 & — \\ — & v^2 & — \\ — & \vdots & — \\ — & v^m & — \end{bmatrix} = \begin{bmatrix} v_1^1 & v_2^1 & \ldots & v_n^1 \\ v_1^2 & v_2^2 & \ldots & v_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ v_1^m & v_2^m & \ldots & v_n^m \end{bmatrix} \in \mathbb{R}^{m \times n}$$

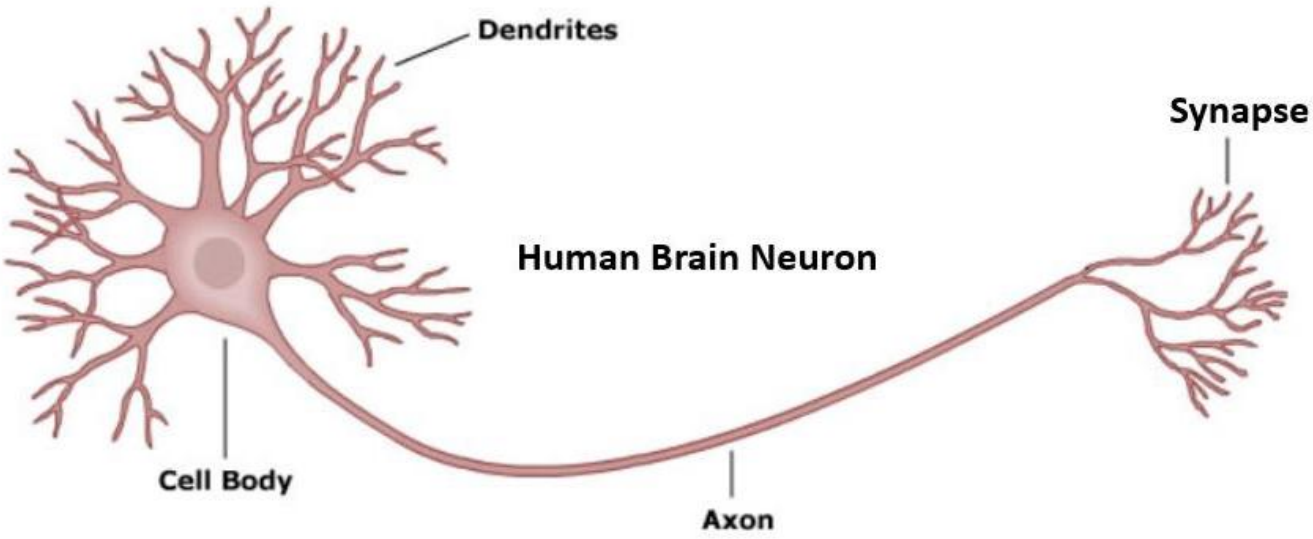
  - By concatenating them:

$$H = \begin{bmatrix} | & | & | & | \\ v_1 & v_2 & \cdots & v_2 \\ | & | & | & | \end{bmatrix} = \begin{bmatrix} v_1^1 & v_1^2 & \ldots & v_1^m \\ v_2^1 & v_2^2 & \ldots & v_2^m \\ \vdots & \vdots & \ddots & \vdots \\ v_n^1 & v_n^2 & \ldots & v_n^m \end{bmatrix} \in \mathbb{R}^{n \times m}$$

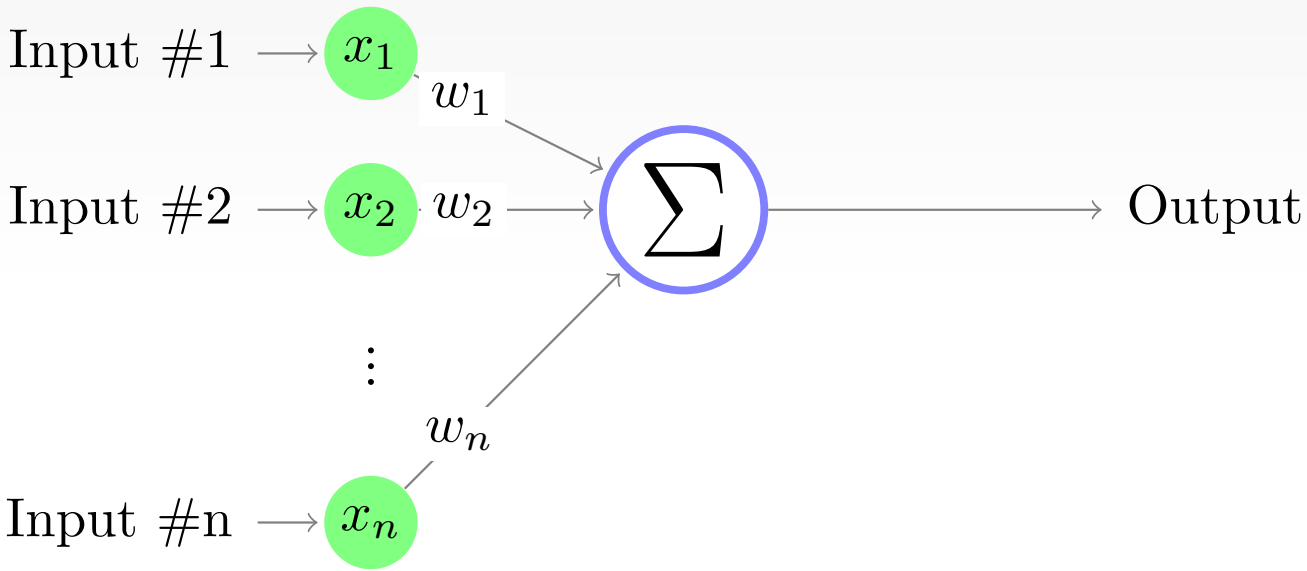
- Transposition: $M^\top = H$.

The Neuron!

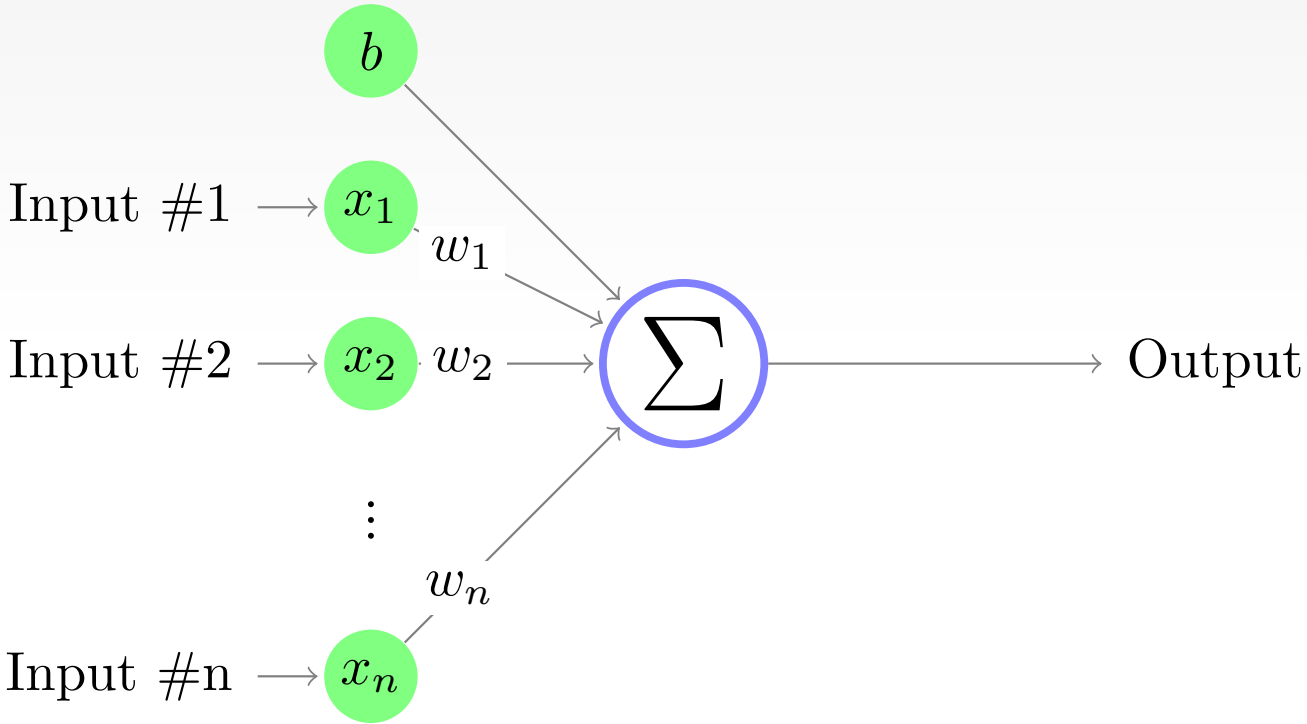




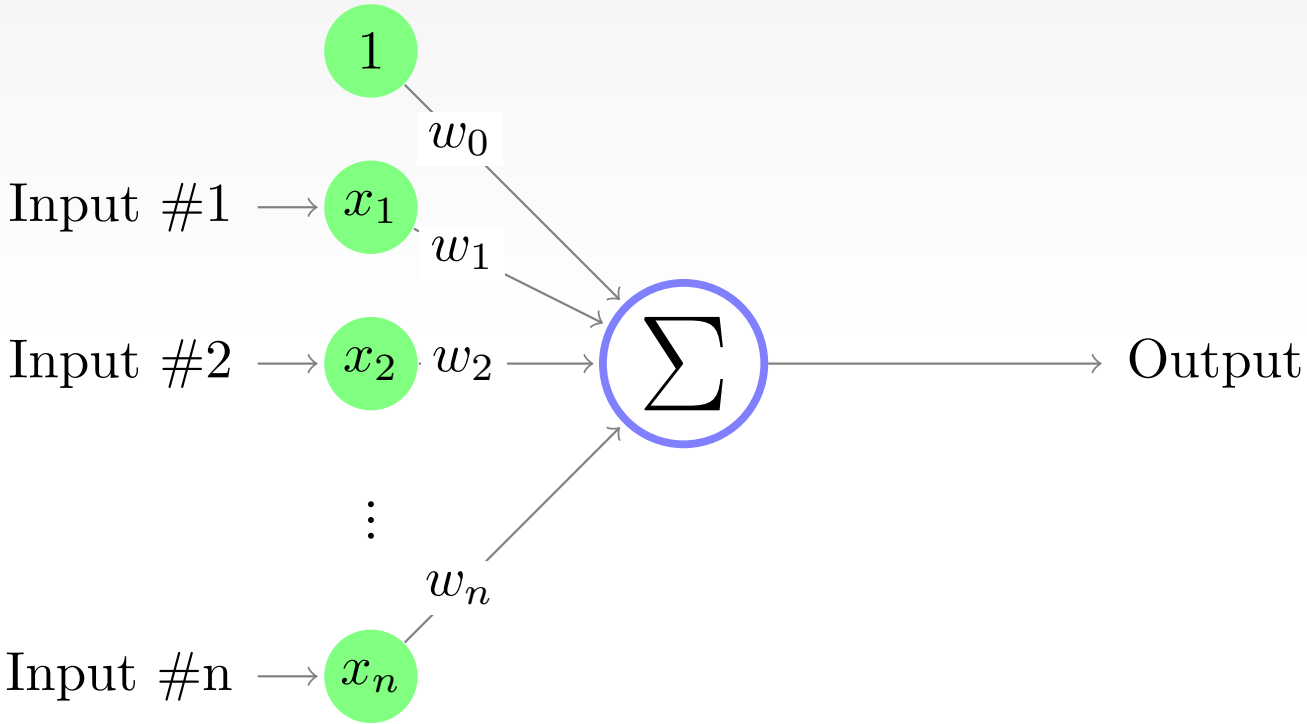
The Neuron Model!



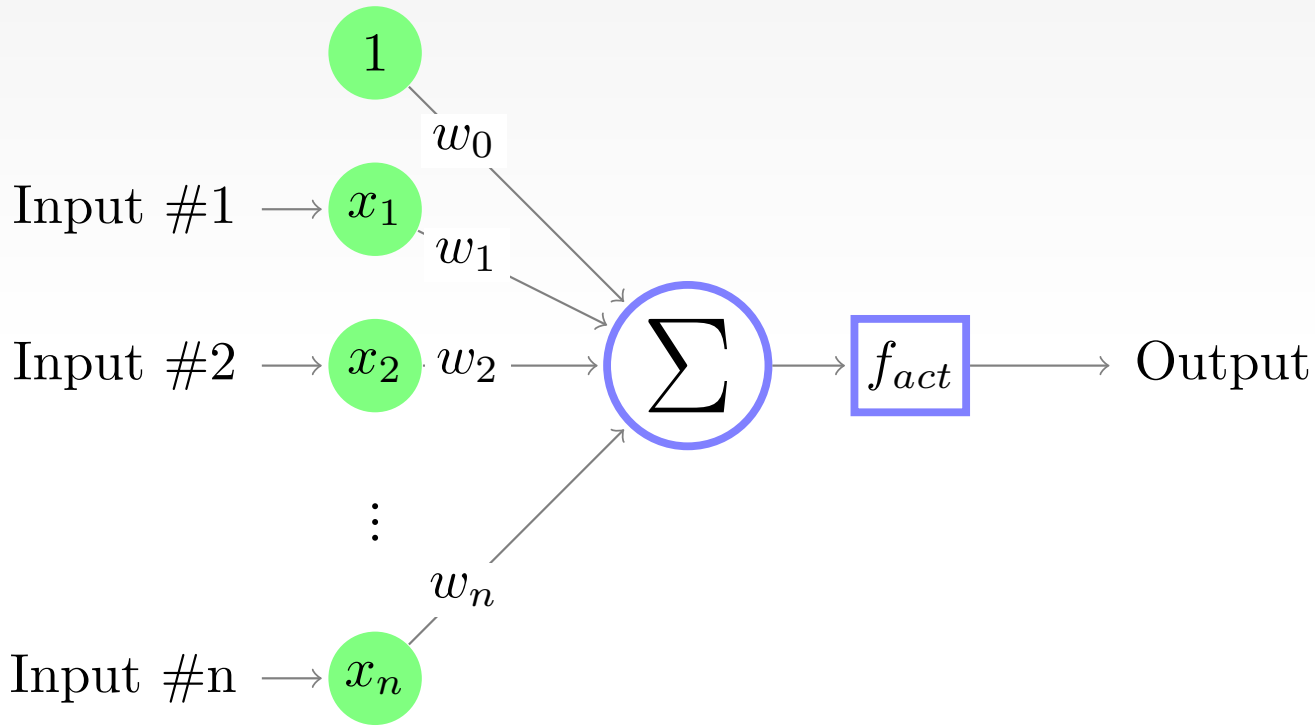
The Neuron Model!



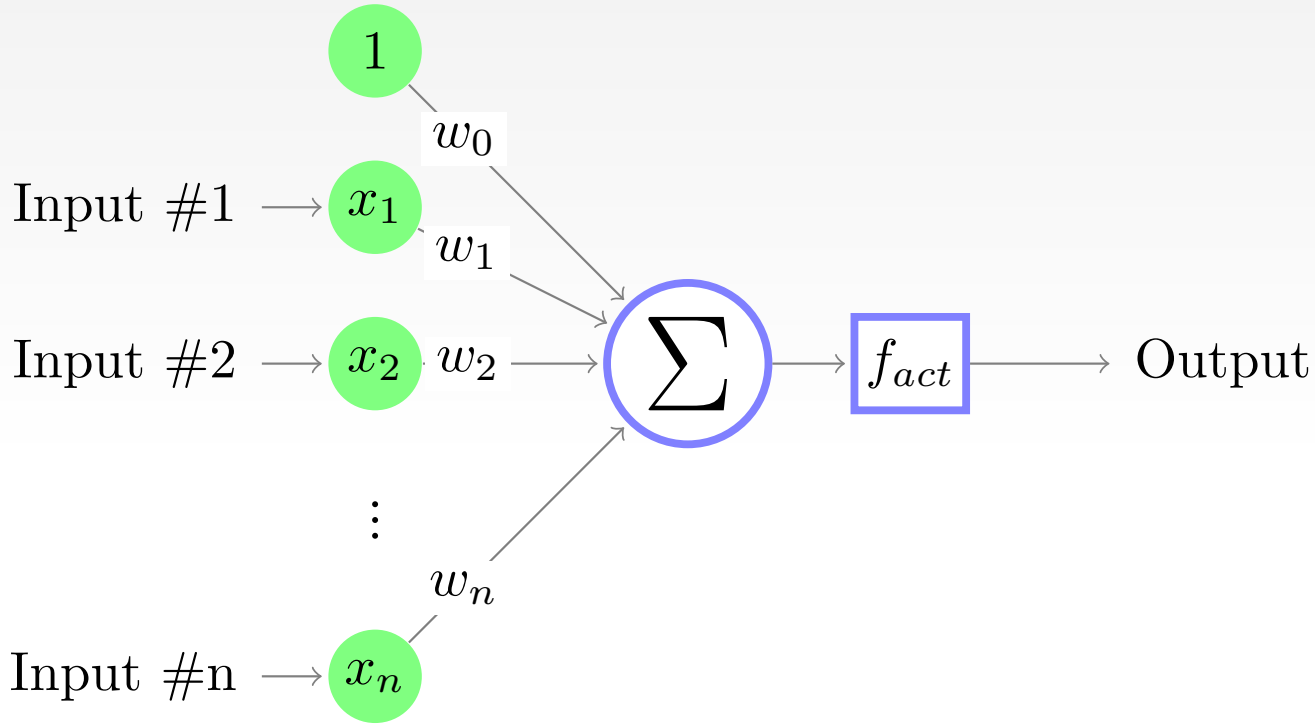
The Neuron Model!



The Neuron Model!

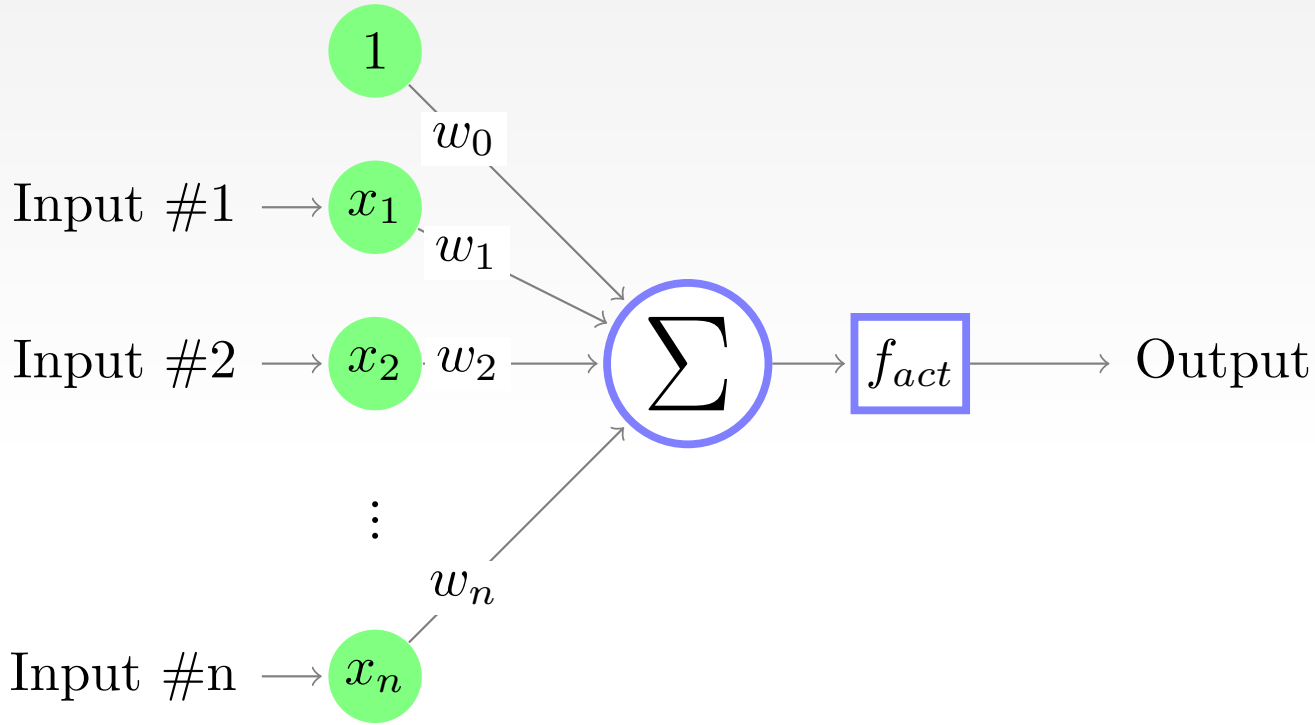


The Neuron Model!



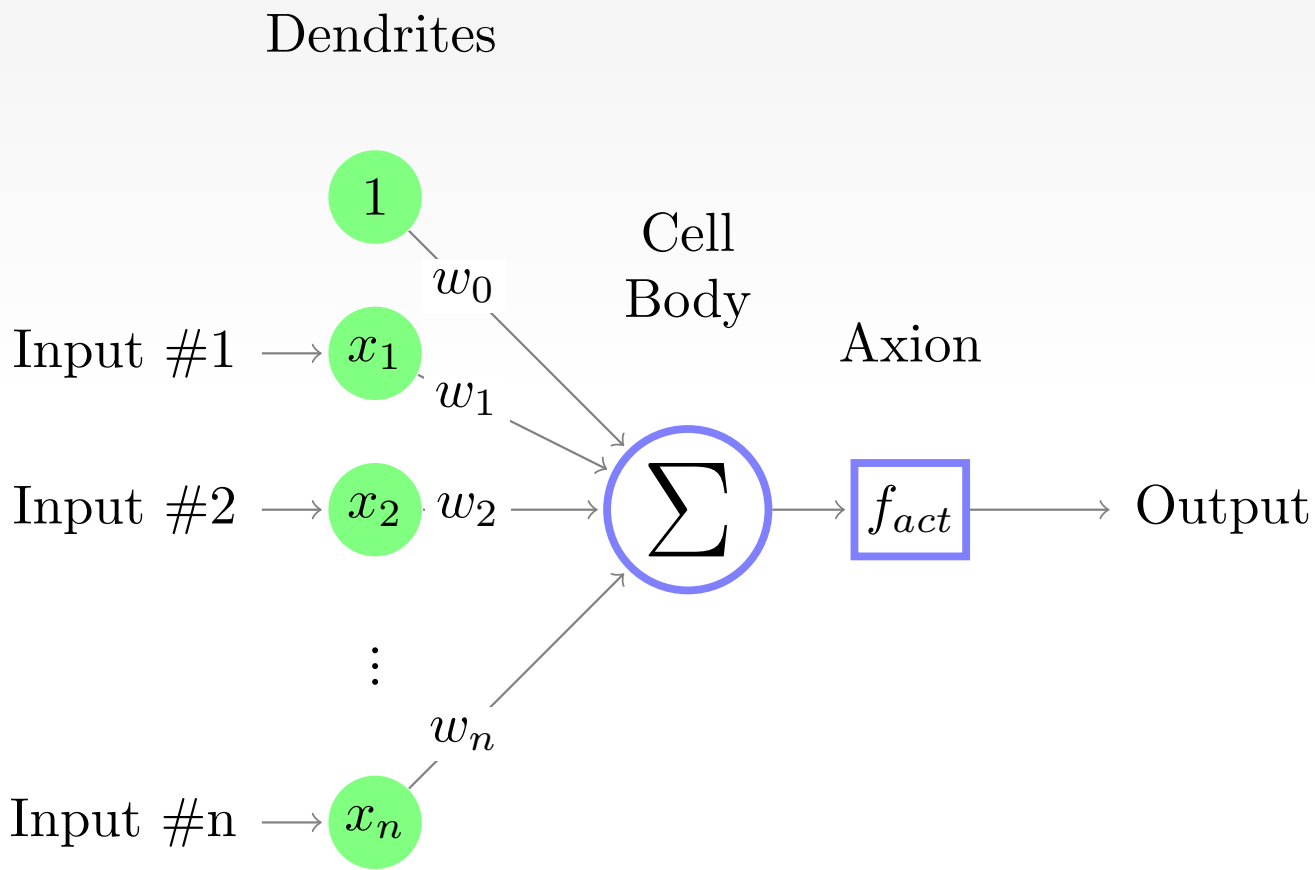
$$f_{act}(x) = \begin{cases} +1, & x > 0 \\ -1, & x < 0 \end{cases}$$

The Neuron Model!

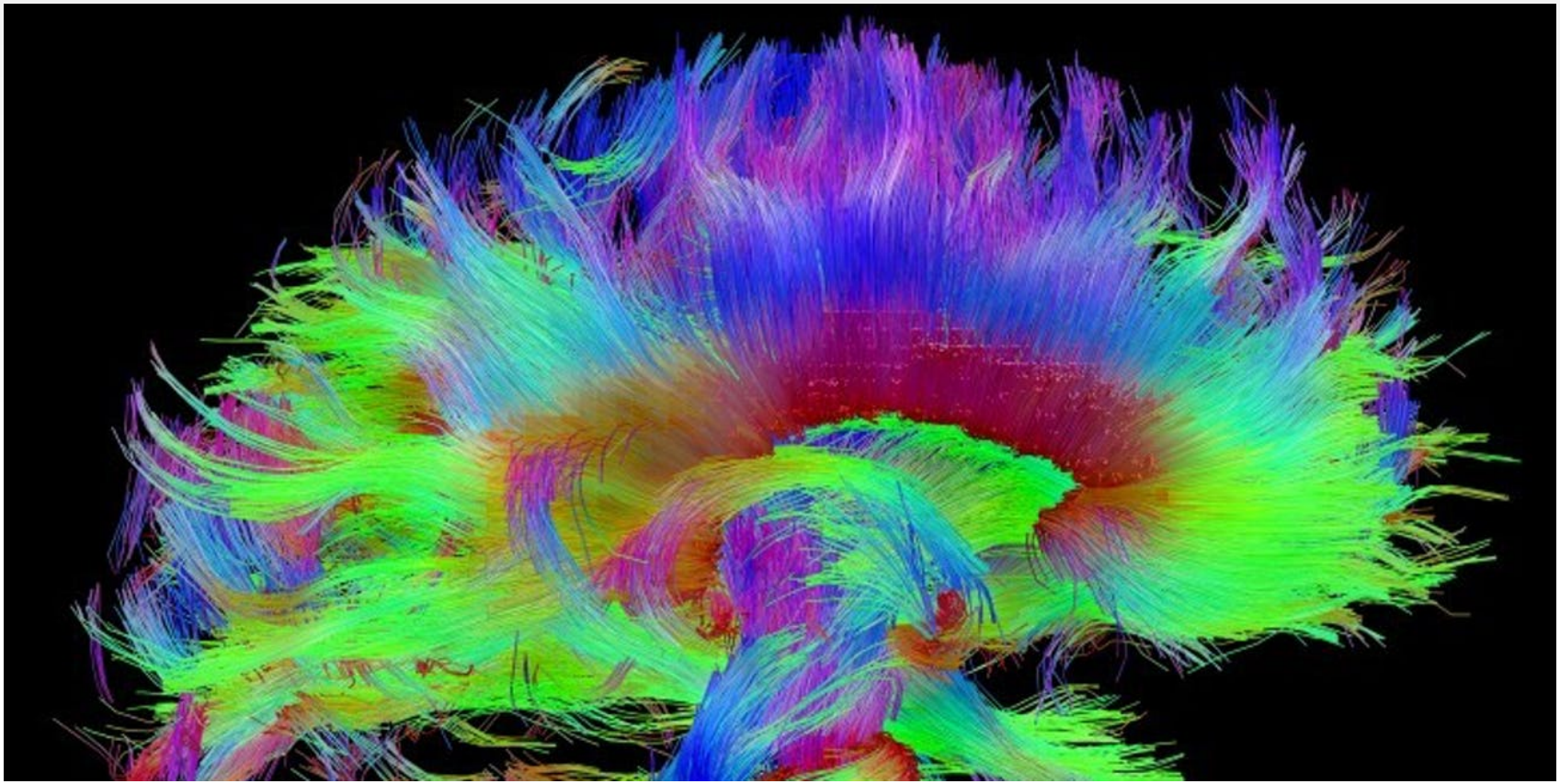


$$f_{act}(x) = \begin{cases} +1, & x > 0 \\ -1, & x < 0 \end{cases} = \text{sign}(x)$$

The Neuron Model!



The Connectome Project



"Human Connectome Project is to build a 'network map' (connectome) that will shed light on the anatomical and functional connectivity within the healthy human brain."

Now, let's use Python!

First Project:

First Project:
The Gray Zone of Artificial Intelligence

First Project

Premise and Goal

- AI revolutionized modern society!
- The goal of this project is to research a recent AI technology and present your findings to the class.

First Project

Premise and Goal

- AI revolutionized modern society!
- The goal of this project is to research a recent AI technology and present your findings to the class.

What you should do

1. Choose a modern application of AI.

First Project

Premise and Goal

- AI revolutionized modern society!
- The goal of this project is to research a recent AI technology and present your findings to the class.

What you should do

1. Choose a modern application of AI.
2. Provide a basic understanding of how this technology works and some beneficial applications of it.

First Project

Premise and Goal

- AI revolutionized modern society!
- The goal of this project is to research a recent AI technology and present your findings to the class.

What you should do

1. Choose a modern application of AI.
2. Provide a basic understanding of how this technology works and some beneficial applications of it.
3. Discuss the implications of that technology in terms of:
   - What are the deeper implications of how this technology affects society?
   - In what ways does this technology lie in a zone of ethical ambiguity.

First Project

Topics

- You should choose a topic among the follwoing:
  - AI and Medicine.
  - AI and Arts.
  - Driverless cars.
  - Modern Robotics.
  - Games and virtual Reality/augmented reality.
  - AI beating humans: Deep Blue (chess), IBM Watson (Jeopardy), Alpha Go (go), etc.
  - Language, automated translation, human-like speech.
  - Self-reconfiguring modular robots.
  - Exoskeleton/wearable robotics.
  - Military technology.
- But you can also suggest a topic

First Project

Topics

- You should choose a topic among the follwoing:
  - AI and Medicine.
  - AI and Arts.
  - Driverless cars.
  - Modern Robotics.
  - Games and virtual Reality/augmented reality.
  - AI beating humans: Deep Blue (chess), IBM Watson (Jeopardy), Alpha Go (go), etc.
  - Language, automated translation, human-like speech.
  - Self-reconfiguring modular robots.
  - Exoskeleton/wearable robotics.
  - Military technology.
- But you can also suggest a topic, but under my approval ☺.
- Different teams, different topics. First come, first chose.
- The deadline to inform the topic is **tommorrow**.

Teams

Team 1:	Ahmad, Allison Y., Allison Z..
Team 2:	Aprille, Arthur, Aryan.
Team 3:	Badr, Camellia, Connor.
Team 4:	Domenic, Iris, Josefine.
Team 5:	Justin, Kabeera, Olivia.
Team 6:	Parth, Ray, Roshni.
Team 7:	Sarah, Sean, Sophia, Zitao.

One last thing! Install Python and Jupyter!