



BITS Pilani
Pilani Campus

Applied Machine Learning

Dr. Harikrishnan N B
Computer Science and Information Systems



SE ZG568 / SS ZG568, Applied Machine Learning Lecture No. 2 [02- Feb-2025]

Recap

‘Learning’ in Machine Learning

Types of Learning

Supervised Learning Setup

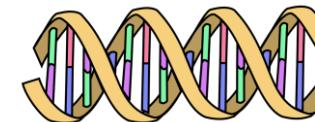
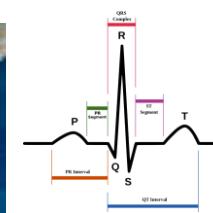
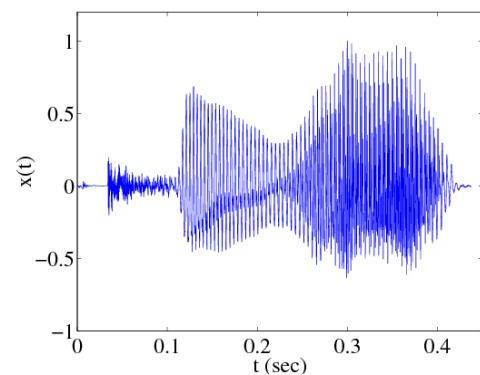
k-NN

Training, Crossvalidation, Testing

Performance Metric

Data is the king - AI Community

What is **Data?**

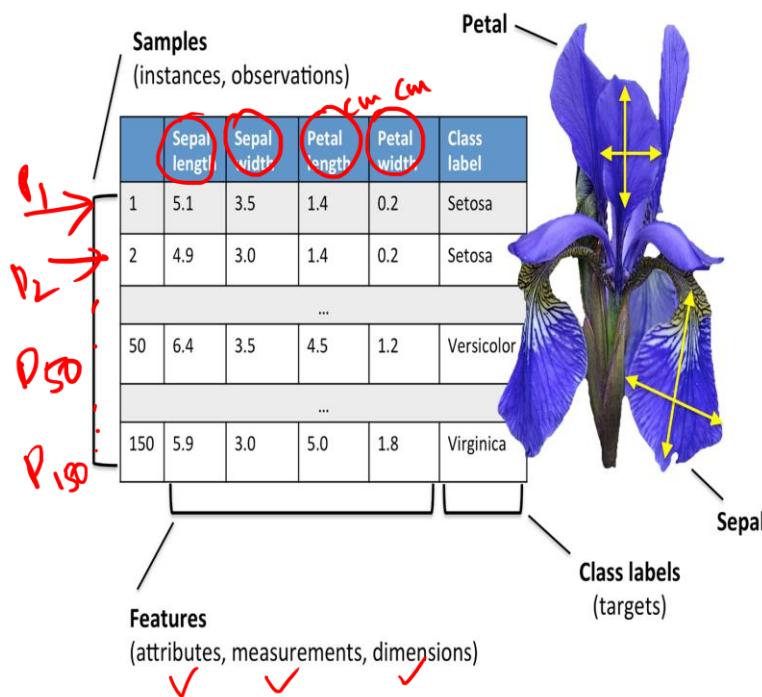


Data is everywhere!

Data Format

	$\downarrow \text{col1}$	$\downarrow \text{col2}$	\dots	$\downarrow \text{coln} \downarrow$	$\begin{matrix} \text{Row1} & f_1, f_2, \dots, f_n \\ \text{Row2} & \vdots \\ \text{RowM} & \end{matrix}$	$M \times N$ matrix
①	f_1 feature ₁	f_2 feature ₂	\dots	f_n feature _n	Output (Y)	
②	a_1	a_2	\dots	a_n	y_1	$M = \# \text{ of data instances}$
③	b_1	b_2	\dots	b_n	y_2	$N = \# \text{ of features}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮
M	⋮	⋮	⋮	⋮	⋮	0

- f_1, f_2, \dots, f_n are called features
- Each row is a data instance or a feature vector
- Y is the output variable.



Iris Plant

Iris Setosa

Iris Versicolor

Iris Virginica

Three class classification

f_1, f_2, f_3, f_4

150×4

$1 < 2 < 3 < 4$
 Blue $>$ Brown

ID	Name	Eye Color	Blood Type	Favorite Sport
1	Alice	Blue <u>1</u>	O	Football
2	Bob	Brown <u>2</u>	A	Basketball
3	Charlie	Green <u>3</u>	B	Cricket
4	David	Hazel <u>4</u>	AB	Tennis

Nominal Data

What is Nominal Data?

- **Definition:** Nominal data is categorical data that represents names or labels with no inherent order or ranking.
- **Key Characteristics:**
 - Discrete categories
 - No numerical meaning
 - Cannot be logically ordered
 - Used for classification & identification

Examples of Nominal Data

- **Personal Information:** Eye color (Brown, Blue, Green), Blood type (A, B, AB, O)
- **Demographics:** Marital Status (Single, Married, Divorced)
- **Technology:** Web Browsers (Chrome, Firefox, Safari)

Preprocessing Nominal Data

ID	Color	Shape	Category
1	Red	Circle	A
2	Blue	Square	B
3	Green	Triangle	A
4	Red	Square	C
5	Blue	Circle	B

Why label encoding is not preferred for Nominal Data

ID	Color	Shape	Category
1	Red	Circle	A
2	Blue	Square	B
3	Green	Triangle	A
4	Red	Square	C
5	Blue	Circle	B

Here, the `LabelEncoder` assigns:

- `Color` : Red → 2, Blue → 0, Green → 1
- `Shape` : Circle → 0, Square → 1, Triangle → 2
- `Category` : A → 0, B → 1, C → 2

ID	Color	Shape	Category
1	2	0	0
2	0	1	1
3	1	2	0
4	2	1	2
5	0	0	1

Issue with Label Encoding for Nominal Data

Since label encoding assigns numbers, models might mistakenly interpret them as ordered (e.g., thinking **Red > Green > Blue**). But in reality, **nominal data has no inherent order**.

One Hot Encoding

ID	Color	Shape	Category
1	Red	Circle	A
2	Blue	Square	B
3	Green	Triangle	A
4	Red	Square	C
5	Blue	Circle	B



Key Takeaways

- **Label Encoding** is not ideal for nominal data because it may create a false sense of order.
- **One-Hot Encoding** is preferred since it represents each category as a separate binary column.

ID	Color_Blue	Color_Green	Color_Red	Shape_Circle	Shape_Square	Shape_Triangle	Category_A	Category_B	Category_C
1	0	0	1	1	0	0	1	0	0
2	1	0	0	0	1	0	0	1	0
3	0	1	0	0	0	1	1	0	0
4	0	0	1	0	1	0	0	0	1
5	1	0	0	1	0	0	0	1	0

Ordinal Data

What is Ordinal Data?

- Ordinal data is categorical data with a meaningful order but no fixed distance between categories.
- It represents **ranked** information.
- Example: **Customer Satisfaction Levels** (Poor, Average, Good, Excellent)

Poor < Average < Good < Excellent

Characteristics of Ordinal Data

- Categories are **ordered** but not necessarily equidistant.
- **Comparison** is possible (e.g., "Good" is better than "Average").
- Arithmetic operations (addition, subtraction) **are not valid**. Why?
- Used in **surveys, grading systems, and Likert scales**.

Examples of Ordinal Data

- **Educational Level:** High School < Bachelor's < Master's < Ph.D.
- **Pain Severity:** No pain < Mild < Moderate < Severe
- **Hotel Ratings:** 1-star < 2-star < 3-star < 4-star < 5-star
- **Agreement Scale:** Strongly Disagree < Disagree < Neutral < Agree < Strongly Agree

$$\text{Bachelors - High School} = 1$$

$$\text{Ph.D - Masters} = 1$$

1 2 3 4

Key Take Away

Tree-based models (e.g., Decision Trees, Random Forests) handle ordinal data and nominal data well without assuming a linear relationship.



Learning - Function Approximation

Problem Setting

- Set of possible instances X .
- Unknown target function $f: X \rightarrow Y$
- Set of function hypotheses $\boxed{H = \{h \mid h: X \rightarrow Y\}}$

Input

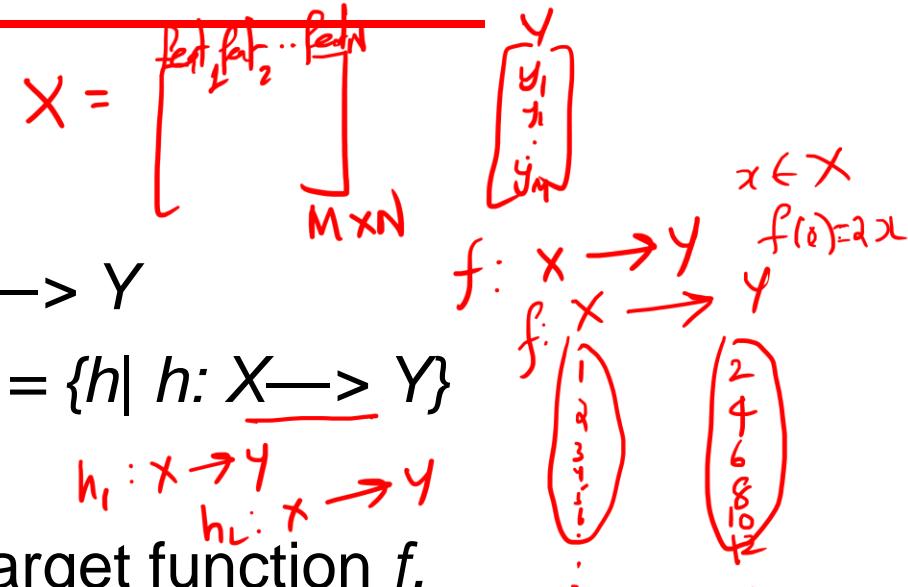
$$h \approx f \quad h_1: X \rightarrow Y \quad h_L: X \rightarrow Y$$

Training Examples of unknown target function f .

Output

$$H = \{h_1; h_2, \dots, h_s, \dots, h_n\} \quad h_n: X \rightarrow Y$$

Hypothesis $\boxed{h \in H}$ that best approximates target function \underline{f} .



Probabilistic Interpretation

I ₁	I ₂	f
0	0	0
0	1	1
1	0	1
1	1	1

logical OR

$$\begin{matrix} I_1 \\ 0 \\ 0 \\ \diagdown \\ 0.5 \\ \diagup \\ 1 \end{matrix}$$

h : Decision tree

Let $H = \{h \mid h : X \rightarrow Y\}$.

Maximum A Posterior Hypothesis:

Decision Tree entropy Impurity function:
Using Shannon Entropy

$$\arg \max_h P(h \mid D) = \arg \max_h \frac{p(D \mid h)P(h)}{P(D)}$$

Since $P(D)$ is constant with respect to h , we have:

$$\arg \max_h P(D \mid h)P(h)$$

Applying the logarithm to both sides:

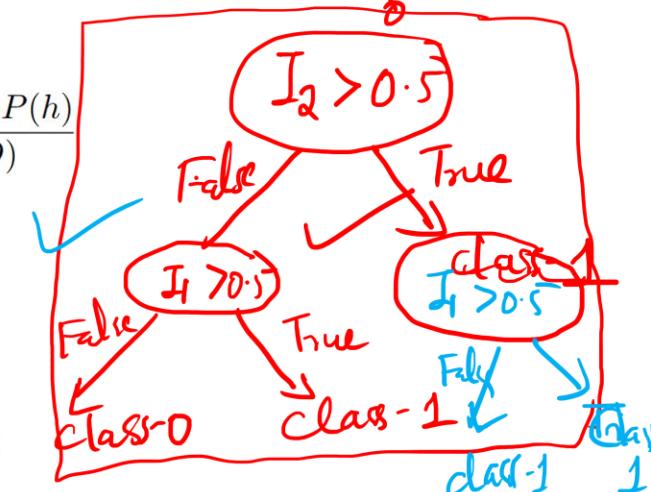
$$\arg \max_h \log P(D \mid h) + \log P(h)$$

Finally, to convert the maximization to minimization, we negate the expression:

$$\arg \min_h -\log P(D \mid h) - \log P(h)$$

Gini impurity

max Info gain
reducing the entropy



D₁ →

Data Compression Interpretation

Minimum Description Length (MDL)

The MDL Principle states that the best hypothesis is the one that minimizes the total description length of the data and the model, which leads to a trade-off between fit and complexity.

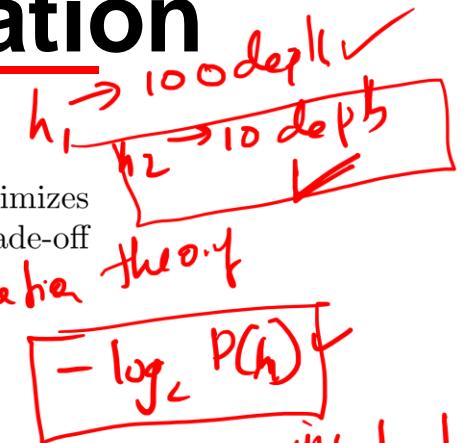
Thus, the minimization problem:

$$\arg \min_h -\log P(D | h) + \log P(h)$$

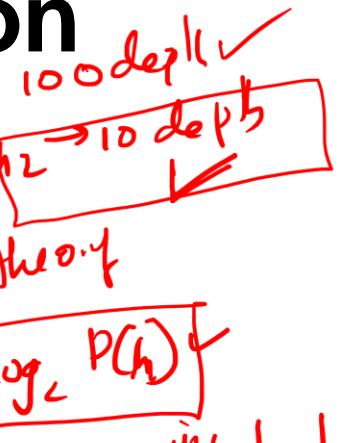
corresponds to selecting the hypothesis h that compresses the data D efficiently (good data fit) and describes the hypothesis itself in a minimal number of bits (simple model).

Summary: The MDL principle balances two competing factors:

- **Fitting the data:** You want a model that fits the data well, but not too well (which could lead to overfitting).
- **Simplicity of the model:** You want a model that is as simple as possible to avoid overfitting and ensure generalization to unseen data.



length in bits of
the description of
the model "h"

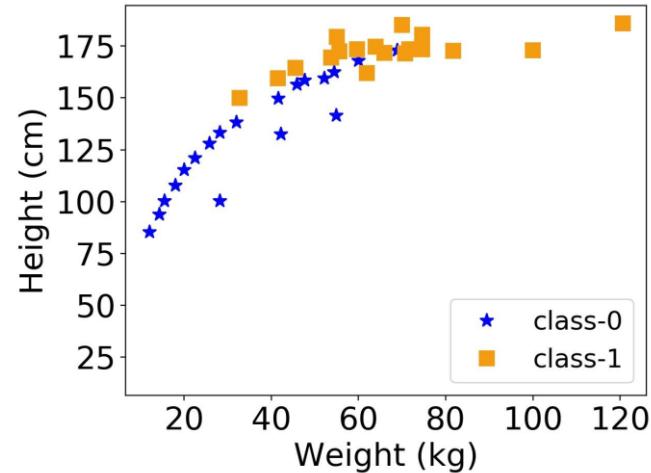


Principled Way of Doing Supervised Learning

Understanding Classification Via Example

A	B	C
Weight (kg)	Height (cm)	Class
12	85.5	0
14.2	94	0
15.4	100.3	0
17.9	107.9	0
199	115.5	0
22.4	121.1	0
25.8	128.2	0
28.1	133.3	0
28.1	100.4	0
31.9	138.4	0
74.6	173.6	1
71.6	173.6	1
59.6	173.6	1
55.6	172.6	1
74.6	180.5	1
70.6	171.6	1
53.6	169.6	1
45.5	164.6	1
41.44	159.65	1
32.68	149.99	1
54.93	179.62	1

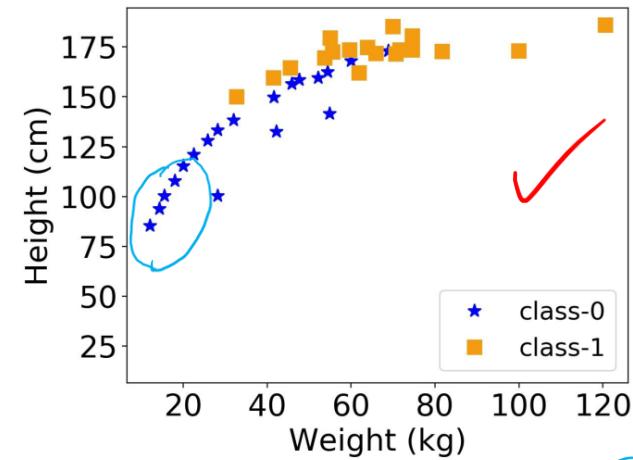
Class-0 below 15 years
Class-1 above 15 years



k-Nearest Neighbours

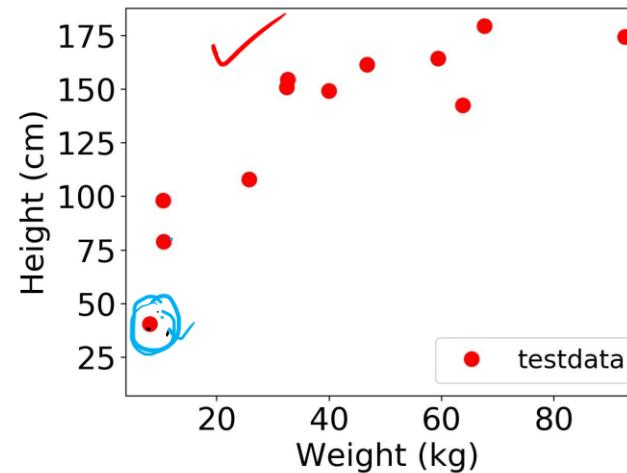
K-Nearest Neighbour Classifier

TRAINDATA



$k=5$

TESTDATA



$$dist(x, z) = \left(\sum_{j=1}^r |x_j - z_j|^p \right)^{\frac{1}{p}}$$

Mathematical Description

Let the training data $x_i \in R^2$ and its corresponding labels $y_i \in \{0, 1\}$ where $i \in \{0, 1, \dots, N\}$

$$D = \{(x_0, y_0), (x_1, y_1), \dots, (x_N, y_N)\}$$

Let the set of k nearest neighbours of z be S_x where $S_x \subseteq D$ and $|S_x| = k$ (no. of elements in S_x is k).

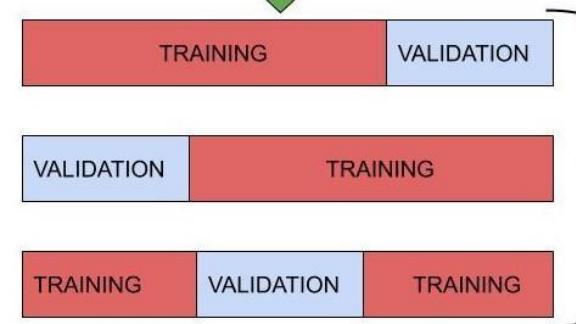
Now $\forall (x', y') \in D \setminus S_x$
 (for all data instances in D but not in S_x) the following is true.

$$dist(z, x') \geq \max_{x'', y'' \in S_x} dist(z, x'')$$

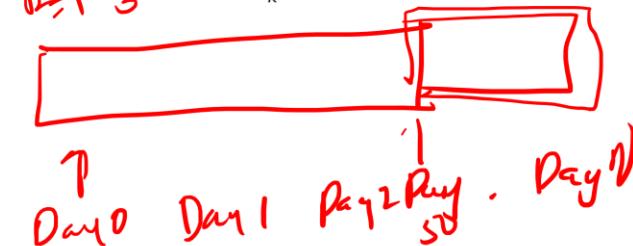
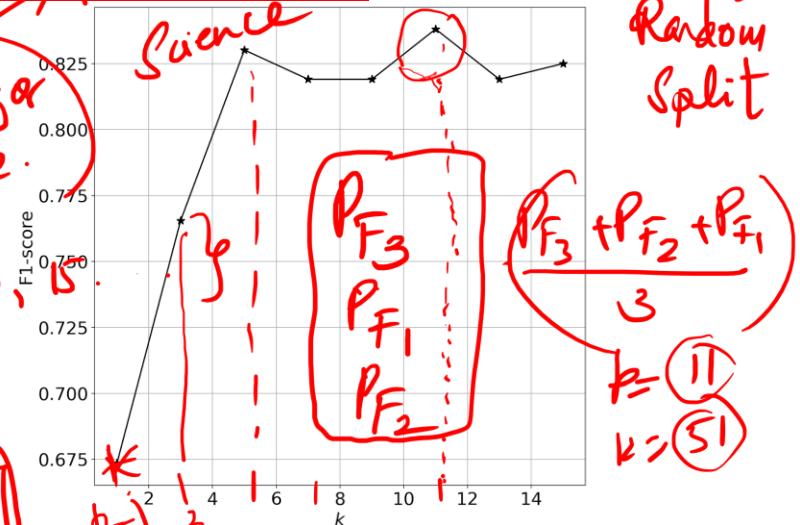
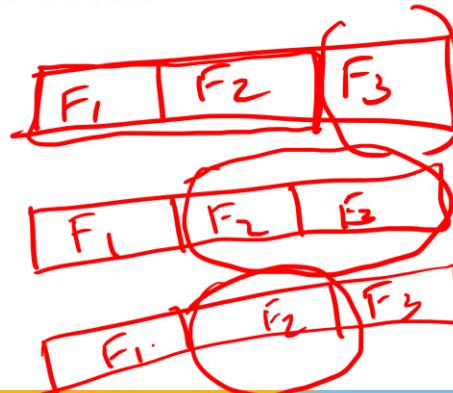
$$dist(x, z) = \left(\sum_{j=1}^r |x_j - z_j|^p \right)^{\frac{1}{p}}$$

How to choose K? Principled way of doing ML

$D = \{(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)\}$ → Philosophy of TimeSplit

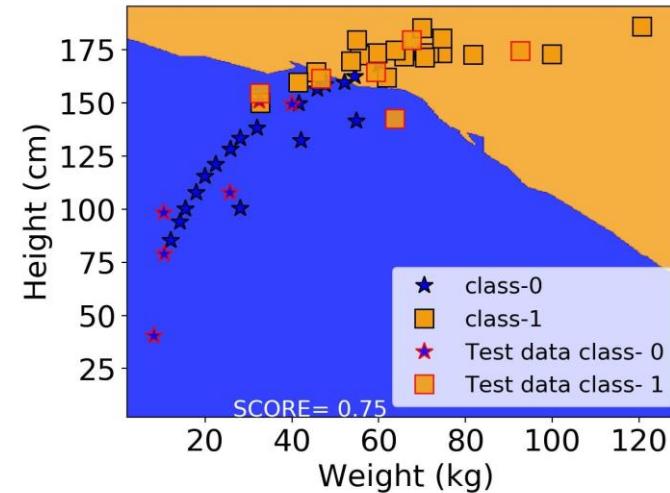
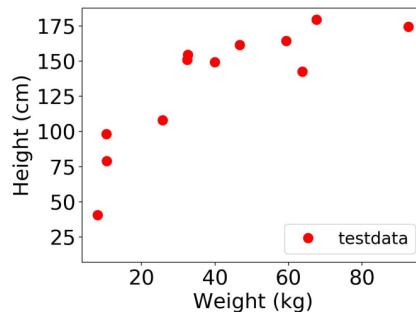
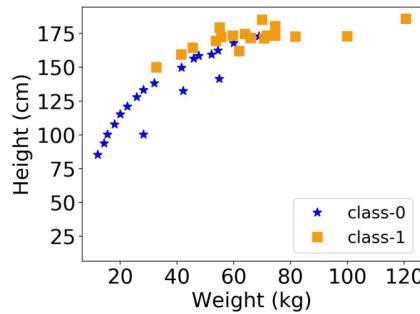


(Occam's Razor Principle)

$$k = \{1, 5, 7, 9, 11, 13, 15\}$$


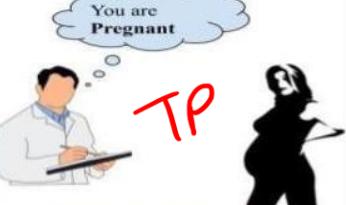
Let us see how the model performs for Test data

data



Confusion Matrix

Performance Measures

		PREDICTED LABEL	
		NEGATIVE	POSITIVE
TRUE LABEL	NEGATIVE	TN 	FP 
	POSITIVE	FN 	TP 

Test data

Model (M)

$$M(\text{testdata}_1) = 0/P_1$$

$$M(\text{testdata}_2) = 0/P_2$$

$$M(\text{testdata}_n) = 0/P_N$$

		PREDICTED LABEL	
		NEGATIVE	POSITIVE
TRUE LABEL	NEGATIVE	TRUE NEGATIVE	FALSE POSITIVE
	POSITIVE	FALSE NEGATIVE	TRUE POSITIVE

Rediff True label

O O

1 O

1 D

0 1

0 1

1 1

1 1

0 1

Performance Measures

		PREDICTED LABEL	
		NEGATIVE	POSITIVE
TRUE LABEL	NEGATIVE	 You are Not Pregnant	 You are Pregnant!
	POSITIVE	 You are Not Pregnant	 You are Pregnant

If we consider a binary classification problem,
 C_{00} represents the count of true negative
 C_{01} represents the count of false positive
 C_{10} represents the count of false negative and
 C_{11} represents the count of true positive.

$$\frac{P}{P+R} = \frac{P}{P+R}$$

$$\Rightarrow \frac{2+P+R}{P+R} \quad \left[Accuracy = \frac{TN + TP}{TN + FP + TP + FN} \right]$$

$$\left[Precision = \frac{TP}{TP + FP} \rightarrow FP = 0 \right]$$

$$\left[Recall = \frac{TP}{TP + FN} \rightarrow FN = 0 \right]$$

$$\Rightarrow F1 Score = 2 * \frac{Precision * Recall}{Precision + Recall} \Rightarrow \text{NA}$$

Example

	Predicted Positive +	Predicted Negative -
Actual Positive +	50 TP	10 FN
Actual Negative -	5 FP	35 TN

$\text{Accuracy} = 0.85$
 $\text{Precision} = \frac{TP}{TP+FP} \Rightarrow 0.9$
 $\text{Recall} = \frac{TP}{TP+FN} \Rightarrow 0.83$
 $\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \Rightarrow 0.85$

Example

	Predicted Positive	Predicted Negative
Actual Positive	50	10
Actual Negative	5	35

Model

- Precision: $\frac{TP}{TP+FP} = \frac{50}{50+5} = 0.909$
- Recall: $\frac{TP}{TP+FN} = \frac{50}{50+10} = 0.833$
- F1-Score: $\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 \times 0.909 \times 0.833}{0.909 + 0.833} = 0.87$

Class Imbalance

Suppose we have 1000 samples, and the confusion matrix (with respect to the original labels) is given by:

	Predicted Positive	Predicted Negative
Actual Positive	TP = 940	FN = 10
Actual Negative	FP = 30	TN = 20

50 -ve
→ 20

$$\text{Accuracy} = 0.96$$

$$\text{Precision} = 0.969$$

$$\text{Recall} = 0.981$$

$$\text{F1 Score} = 0.978$$

Class Imbalance - Misinterpretation of Results

innovate

achieve

lead

Results

Suppose we have 1000 samples, and the confusion matrix (with respect to the original labels) is given by:

	Predicted Positive	Predicted Negative
Actual Positive	TP = 940	FN = 10
Actual Negative	FP = 30	TN = 20

Accuracy measures the overall proportion of correctly classified instances:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{Total}} = \frac{940 + 20}{1000} = \frac{960}{1000} = 0.96.$$

- Precision₊:

The fraction of predicted positives that are truly positive.

$$\text{Precision}_+ = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{940}{940 + 30} = \frac{940}{970} \approx 0.969.$$

- Recall₊:

The fraction of actual positives that are correctly predicted.

$$\text{Recall}_+ = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{940}{940 + 10} = \frac{940}{950} \approx 0.989.$$

- F1-score₊:

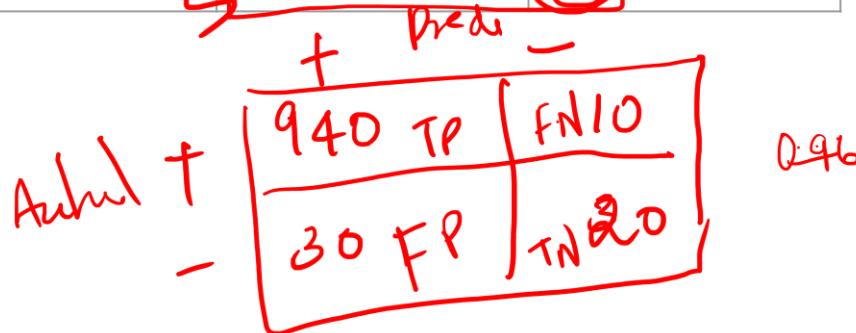
The harmonic mean of precision and recall.

$$\text{F1}_+ = \frac{2 \times \text{Precision}_+ \times \text{Recall}_+}{\text{Precision}_+ + \text{Recall}_+} \approx \frac{2 \times 0.969 \times 0.989}{0.969 + 0.989} \approx \frac{1.916}{1.958} \approx 0.979.$$

For class imbalance data, **accuracy leads to misleading interpretation**, precision, recall, f1-score computed using positive class alone is not enough to interpret the results.

Macro Precision, Recall F1-Score

	Predicted Positive	Predicted Negative
Actual Positive	50	10
Actual Negative	5	35



Class-wise Metrics:

1. Positive Class (Class 1):

- Precision: $\frac{TP}{TP+FP} = \frac{50}{50+5} = 0.909$
- Recall: $\frac{TP}{TP+FN} = \frac{50}{50+10} = 0.833$
- F1-Score: $\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 \times 0.909 \times 0.833}{0.909 + 0.833} = 0.87$

$$\frac{940}{9}$$

2. Negative Class (Class 0):

- Precision: $\frac{TN}{TN+FP} = \frac{35}{35+5} = 0.875$
- Recall: $\frac{TN}{TN+FN} = \frac{35}{35+10} = 0.778$
- F1-Score: $\frac{2 \times 0.875 \times 0.778}{0.875 + 0.778} = 0.824$

$$\frac{35}{35+5}$$

$$\frac{35}{55+10}$$

Macro-Averaged Metrics:

Macro-averaging treats each class equally by computing the average of the metrics across all classes.

- Macro Precision: $\frac{0.909+0.875}{2} = 0.892$
- Macro Recall: $\frac{0.833+0.778}{2} = 0.806$
- Macro F1-Score: $\frac{0.87+0.824}{2} = 0.847$

$$\begin{aligned} &= 0.78 \\ &= 0.694 \\ &= 0.738 \end{aligned}$$

Compute Macro Precision, Recall, F1-score

Suppose we have 1000 samples, and the confusion matrix (with respect to the original labels) is given by:

	Predicted Positive	Predicted Negative
Actual Positive	TP = 940	FN = 10
Actual Negative	FP = 30	TN = 20

Macro Precision
Macro F1 score
Macro Recall



$$\text{Macro Precision} = 0.83$$

Macro Recall
Macro F1 score

So far...

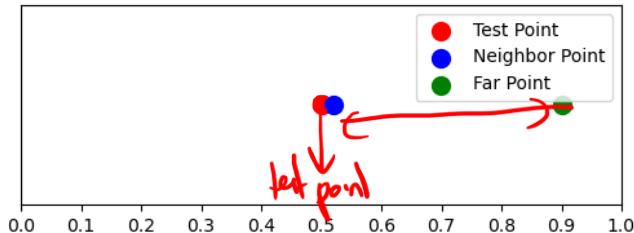


- Knowledge of the problem that you are trying to solve →
- Load the data correctly
- Analyze the data
 - How many classes?
 - Number of data instances per class
 - Train-test split → *RandomSplit*, *TimeSplit*
- Algorithm used for solving the problem
 - What are the hyperparameters of the algorithm?
 - Crossvalidation to find the best hyperparameters.
- Retrain the training data with the tuned hyperparameters.
- Used the trained model (with best hyperparameters obtained after crossvalidation) on test data to evaluate the performance of unseen test data. This should be done only once.

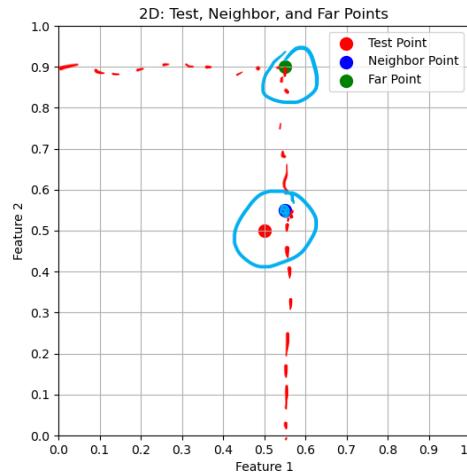
DT = {
kNN = {
SVM = {
NB = {
LR = {

Notion of closeness of points

1D: Test, Neighbor, and Far Points



3D Plot with Axes Meeting at the Origin (0,0,0)



Train Feature 1 \in $(\text{Test Feature}_1 - \epsilon, \text{Test Feature}_1 + \epsilon)$

Train Feature 2 \in $(\text{Test Feature}_2 - \epsilon, \text{Test Feature}_2 + \epsilon)$

Test Feature 1 \in $(\text{Test Feature}_1 - \epsilon, \text{Test Feature}_1 + \epsilon)$

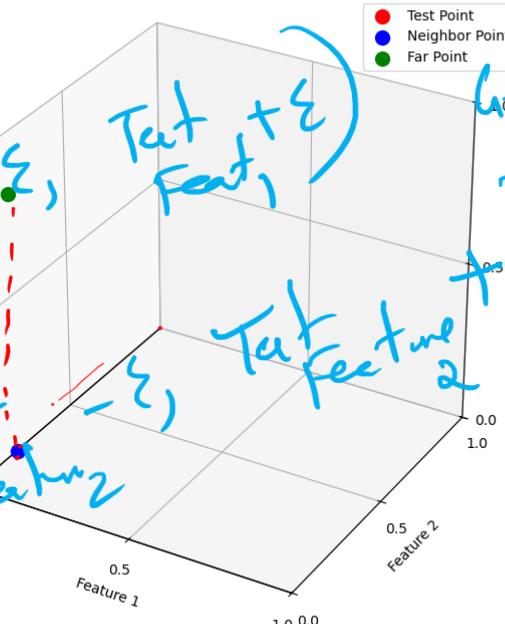
Test Feature 2 \in $(\text{Test Feature}_2 - \epsilon, \text{Test Feature}_2 + \epsilon)$

Far Feature 1 \in $(\text{Test Feature}_1 - \epsilon, \text{Test Feature}_1 + \epsilon)$

Far Feature 2 \in $(\text{Test Feature}_2 - \epsilon, \text{Test Feature}_2 + \epsilon)$

Neighbor Feature 1 \in $(\text{Test Feature}_1 - \epsilon, \text{Test Feature}_1 + \epsilon)$

Neighbor Feature 2 \in $(\text{Test Feature}_2 - \epsilon, \text{Test Feature}_2 + \epsilon)$



In N dimensions

What is the notion of closeness in high dimension?

$$X_{\text{test}} = [f_1^{\text{test}}, f_2^{\text{test}}, f_3^{\text{test}}, f_4^{\text{test}}, \dots, f_N^{\text{test}}]$$

$$f_1^{\text{train}} \quad f_2^{\text{train}} \quad f_3^{\text{train}} \quad f_4^{\text{train}} \quad \dots \quad f_N^{\text{train}}$$

A point in X_{train} is close to X_{test} when
for all i from 1 to N ,

$$f_i^{\text{train}} \in [f_i^{\text{test}} - \epsilon, f_i^{\text{test}} + \epsilon]$$

$$i=1 \text{ to } N$$

$$f_1^{\text{train}} \in (f_1^{\text{test}} - \epsilon, f_1^{\text{test}} + \epsilon)$$

$$f_2^{\text{train}} \in (f_2^{\text{test}} - \epsilon, f_2^{\text{test}} + \epsilon)$$

Assumptions

Setup:

- Let there be N independent features:

iid

$$(f_{i-\epsilon}, f_{i+\epsilon})$$

$$f_1, f_2, \dots, f_N,$$

where each feature is uniformly distributed on the interval $[0, 1]$.

- Consider a fixed test point

$$X_{\text{test}} = (f_1, f_2, \dots, f_N) \in [0, 1]^N.$$

- For a given small $\epsilon > 0$ (assumed to be small enough so that for every i the interval remains

within $[0, 1]$), define for each dimension i the interval:

$$\int_a^b \underbrace{1}_{\text{constant}} dx = 1$$

$$I_i = [f_i - \epsilon, f_i + \epsilon].$$

$$\int_a^b \underbrace{1}_{\text{constant}} dx = 1$$

$$\int_a^b f(x) dx = 1$$

$$c \times [b-a] = 1$$

$$\int_{\Omega} 1 d\omega \Rightarrow \int_{\Omega} z^i dz = 1 - 0 = 1$$

innovate

achieve

lead

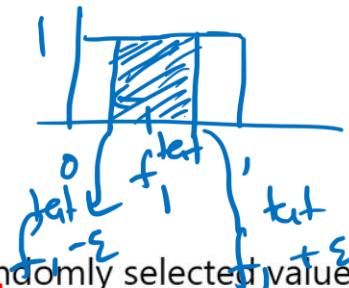
$$P(f_1, f_2, \dots, f_N) = P(f_1) \cdot P(f_2) \cdots P(f_N)$$

Probability Density

$f(x) \leq \frac{1}{b-a}$

$$\int_{-\infty}^{\infty} f(x) dx = 1$$

$$P(f_i^{\text{test}} \in (f_i^{\text{train}} - \epsilon, f_i^{\text{train}} + \epsilon))$$



Probability in One Dimension:

Since each feature is uniformly distributed over $[0, 1]$, the probability that a randomly selected value for feature i falls within I_i is given by the length of the interval:

$$P(\text{feature } i \in I_i) = 2\epsilon.$$

This calculation assumes that $I_i \subset [0, 1]$.

$$\int_{f_i^{\text{test}} - \epsilon}^{f_i^{\text{test}} + \epsilon} 1 dx = 2\epsilon$$

$$[x]_{f_i^{\text{test}} - \epsilon}^{f_i^{\text{test}} + \epsilon} = 2\epsilon$$



$$P(f_1, f_2, \dots, f_n) = P(f_1)P(f_2) \cdots P(f_n)$$

Joint Probability (All Dimensions):

$$P(f_1^{\text{test}} - \epsilon < f_1^{\text{train}}, f_1^{\text{test}} + \epsilon) = 2\epsilon$$

Assuming the features are independent, the probability that a randomly selected point

$$\begin{aligned} P(f_2^{\text{test}} - \epsilon < f_2^{\text{train}} < f_2^{\text{test}} + \epsilon) &= 2\epsilon \\ \mathbf{x} = (x_1, x_2, \dots, x_N) &= (2\epsilon)^N \end{aligned}$$

falls within the "close" region around X_{test} (i.e., $x_i \in I_i$ for every i) is the product of the individual probabilities:

$$P(\mathbf{x} \text{ is close to } X_{\text{test}}) = \prod_{i=1}^N P(x_i \in I_i) = (2\epsilon)^N.$$

$$\epsilon < 0.5$$

$$\begin{aligned} \text{As } N \rightarrow \infty \\ (2\epsilon)^N &= 0 \end{aligned}$$

Curse of Dimensionality

The Curse of Dimensionality:

As the number of dimensions N increases, the probability

$$P(\text{close}) = (2\epsilon)^N$$

decreases exponentially. This exponential decay means that—even if each individual feature has a relatively high chance (i.e., 2ϵ) of being within ϵ of the corresponding value in X_{test} —the joint probability that *all* features are simultaneously close is extremely small.

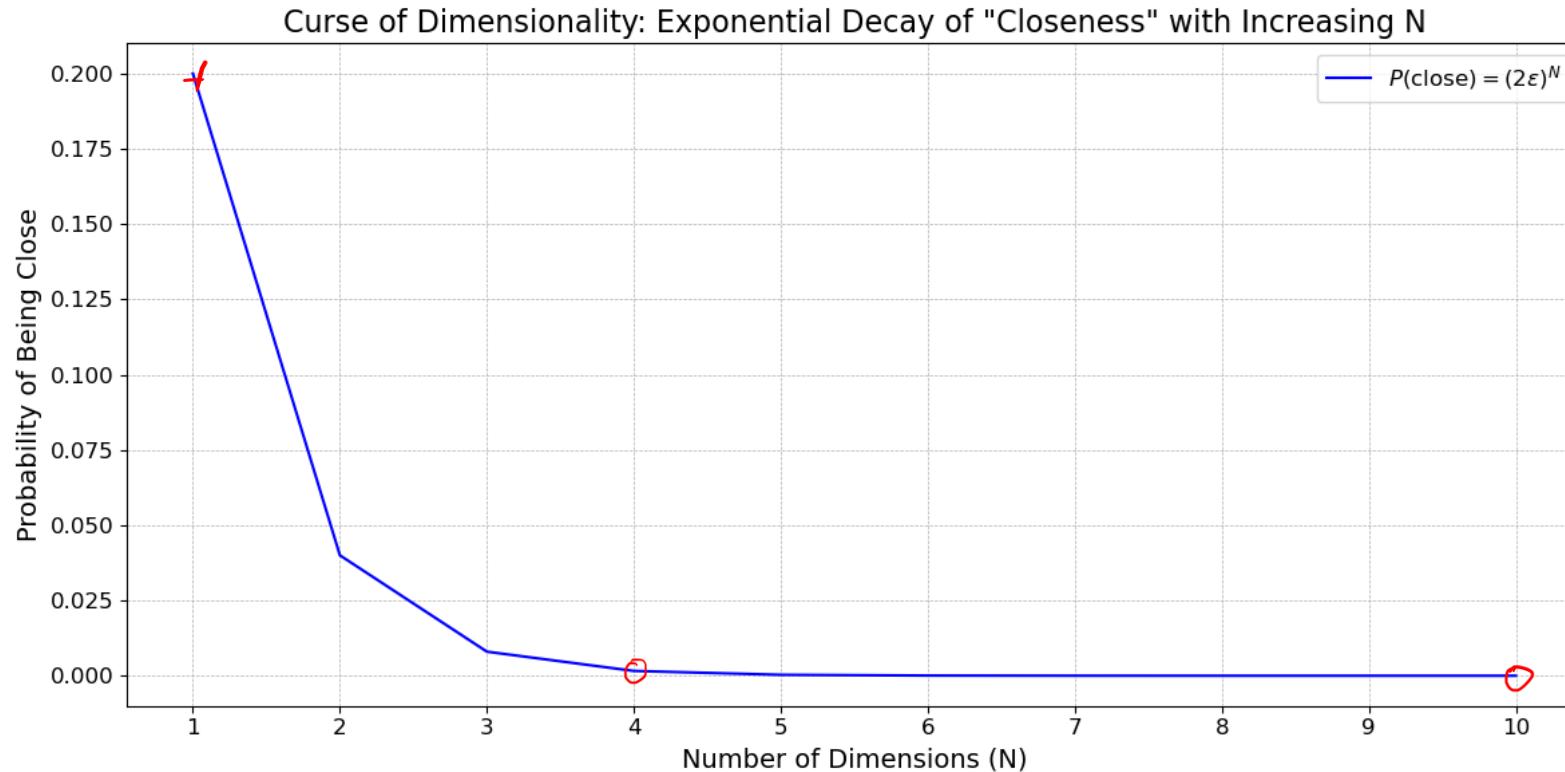
Example:

If $N = 10$ and $\epsilon = 0.1$, then:

$$P(\text{close}) = (2 \times 0.1)^{10} = (0.2)^{10} \approx 1.024 \times 10^{-7}.$$

This extremely small probability illustrates that in a 10-dimensional space, it is highly unlikely for a randomly chosen point to be close to X_{test} in every feature, which is a clear demonstration of the **curse of dimensionality**.

Curse of Dimensionality



Dimensionality Reduction Techniques

Principal Component Analysis

Singular Value Decomposition

t-Distributed Stochastic Neighbor Embedding
(t-SNE)

Why should I learn Linear Algebra?



Fractals using Iterated Function Systems

Barnsley Fern Leaf

Why should I learn Linear Algebra?



Barnsley Fern Leaf

Are you curious to **write a computer program** to generate this fern leaf?

w	a	b	c	d	e	f	p	Portion generated
f_1	0	0	0	0.16	0	0	0.01	Stem
f_2	0.85	0.04	-0.04	0.85	0	1.60	0.85	Successively smaller leaflets
f_3	0.20	-0.26	0.23	0.22	0	1.60	0.07	Largest left-hand leaflet
f_4	-0.15	0.28	0.26	0.24	0	0.44	0.07	Largest right-hand leaflet

These correspond to the following transformations:

$$f_1(x, y) = \begin{bmatrix} 0.00 & 0.00 \\ 0.00 & 0.16 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$f_2(x, y) = \begin{bmatrix} 0.85 & 0.04 \\ -0.04 & 0.85 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.00 \\ 1.60 \end{bmatrix}$$

$$f_3(x, y) = \begin{bmatrix} 0.20 & -0.26 \\ 0.23 & 0.22 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.00 \\ 1.60 \end{bmatrix}$$

$$f_4(x, y) = \begin{bmatrix} -0.15 & 0.28 \\ 0.26 & 0.24 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.00 \\ 0.44 \end{bmatrix}$$

Why should I learn Linear Algebra?

Google search results for "fourier":

Joseph Fourier
French mathematician and physicist

Joseph Fourier
Jean-Baptiste Joseph Fourier was a French mathematician and physicist born in Auxerre and best known for initiating the investigation of Fourier series, ...

Fourier transform
In physics, engineering and mathematics, the Fourier transform (FT) is an integral transform that takes a function as input and outputs another function ...

People also ask

What is Fourier used for?

Joseph Fourier | Biography & Facts
Joseph Fourier, French mathematician, known also as an Egyptologist and administrator, who exerted strong influence on mathematical physics.

Joseph Fourier (1768 - 1830) - Biography
Joseph Fourier studied the mathematical theory of heat conduction. He established the partial differential equation governing heat diffusion and solved it by ...

But what is the Fourier Transform? A visual introduction.

Why should I learn Linear Algebra?

Google search results for "fourier":

Joseph Fourier
French mathematician and physicist

Wikipedia
https://en.wikipedia.org/wiki/Joseph_Fourier

Jean-Baptiste Joseph Fourier was a French mathematician and physicist born in Auxerre and best known for initiating the investigation of Fourier series, ...

Fourier transform
In physics, engineering and mathematics, the Fourier transform (FT) is an integral transform that takes a function as input and outputs another function ...

People also ask

What is Fourier used for?

Britannica
<https://www.britannica.com.../Sociology-&Society>

Joseph Fourier | Biography & Facts
Joseph Fourier, French mathematician, known also as an Egyptologist and administrator, who exerted strong influence on mathematical physics.

MacTutor History of Mathematics
<https://mathshistory.st-andrews.ac.uk/Biographies/F...>

Joseph Fourier (1768 - 1830) - Biography
Joseph Fourier studied the mathematical theory of heat conduction. He established the partial differential equation governing heat diffusion and solved it by ...

YouTube • 3Blue1Brown
10+ views • 6 years ago

But what is the Fourier Transform? A visual introduction.

How does Google Search Engine works? The secret lies in linear algebra



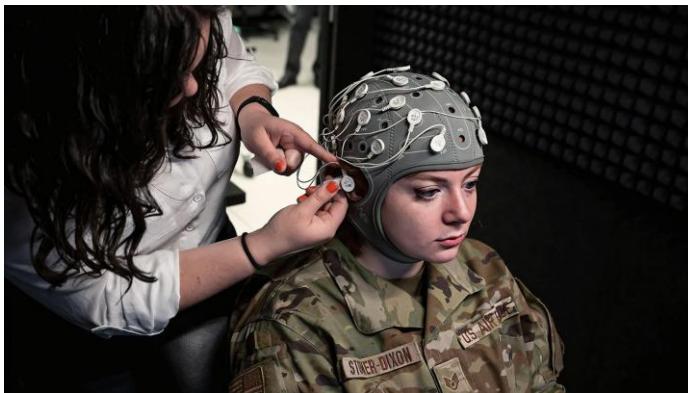
Page Rank Algorithm

Larry Page and Sergey Brin

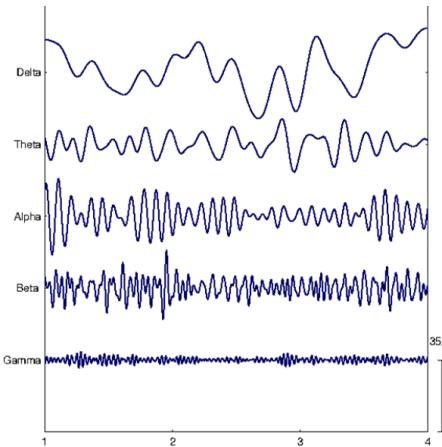
Image Source: <https://www.theverge.com/2019/12/4/20994361/google-alphabet-larry-page-sergey-brin-sundar-pichai-co-founders-ceo-timeline>

Why should I learn Linear Algebra?

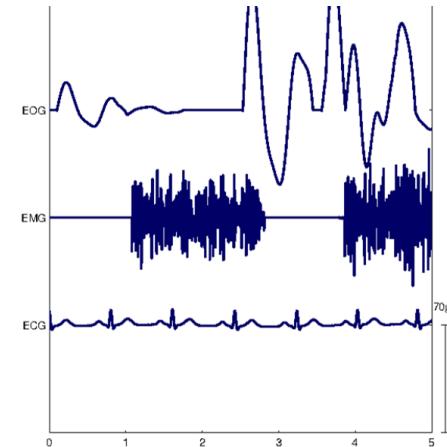
Ever measured the electrical activity of your brain?



Independent Component Analysis (ICA), Principal Component Analysis (PCA) to remove EEG artifacts



(a) Brain Rhythms



(b) Artifacts

Figure 1. (a) Five normal brain rhythms, from low to high frequencies. Delta, Theta, Alpha, Beta and Gamma rhythms comprise the background EEG spectrum. (b) Three different types of artifacts. Ocular, muscular and cardiac artifacts are the most frequent physiological contaminants in the literature on EEG artifact removal.

1. Reference: Urigüen, J. A., & Garcia-Zapirain, B. (2015). EEG artifact removal—state-of-the-art and guidelines. *Journal of neural engineering*, 12(3), 031001.

Goal of Linear Algebra

- The central problem of Linear Algebra is to **understand** a system of linear equations.

Goal of Linear Algebra

- The central problem of Linear Algebra is to **understand** a system of linear equations.

$$\begin{array}{l} -x + y = 0 \\ 2x + y = 3 \end{array} \quad \left. \begin{array}{l} \\ \end{array} \right\} \text{System of Linear Equations}$$

Goal of Linear Algebra

- The central problem of Linear Algebra is to **understand** a system of linear equations.

$$\begin{array}{l} -x + y = 0 \\ 2x + y = 3 \end{array} \quad \left. \begin{array}{l} \\ \end{array} \right\} \text{System of Linear Equations}$$

Does this system of linear equations has a solution?



What is the value of the **unknown variables x and y** that satisfies this system of linear equations?

Goal of Linear Algebra

- The central problem of Linear Algebra is to **understand** a system of linear equations.

$$\begin{aligned} -x + y &= 0 \\ 2x + y &= 3 \end{aligned} \quad \left. \begin{array}{l} \\ \end{array} \right\} \text{System of Linear Equations}$$

Does this system of linear equations has a solution?



What is the value of the **unknown variables x and y** that satisfies this system of linear equations?

- Understanding involves**
 - Insights about row picture and column picture.
 - Explore the existence of solution to the system of linear equations.
 - Insights about column space, row space, right null space, left null space.
 - What new can we say about the system?**



Matrix- vector multiplication as dot product

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_{11} \times x + a_{12} \times y \\ a_{21} \times x + a_{22} \times y \end{bmatrix}$$

Dot Product

$$a = \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} \quad b = \begin{bmatrix} b_{11} \\ b_{21} \end{bmatrix}$$

$$a \cdot b = a^T b$$

Matrix- vector multiplication as dot product

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_{11} \times x + a_{12} \times y \\ a_{21} \times x + a_{22} \times y \end{bmatrix}$$

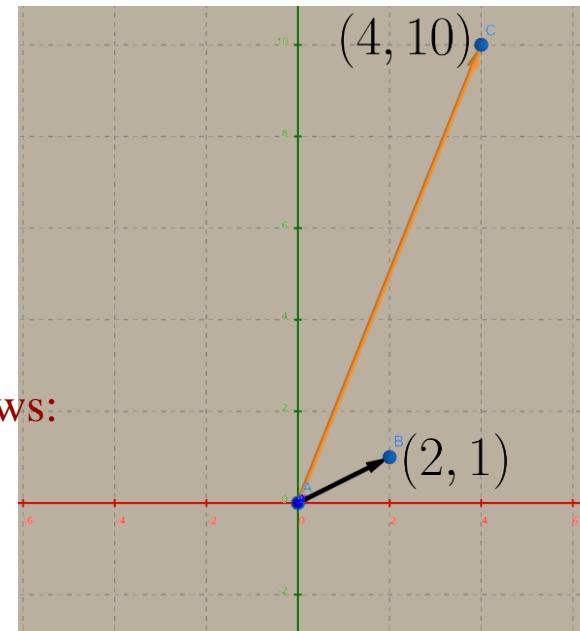
Geometric Interpretation of Matrix Vector Multiplication

Intuition for Matrix vector multiplication for Square Matrices

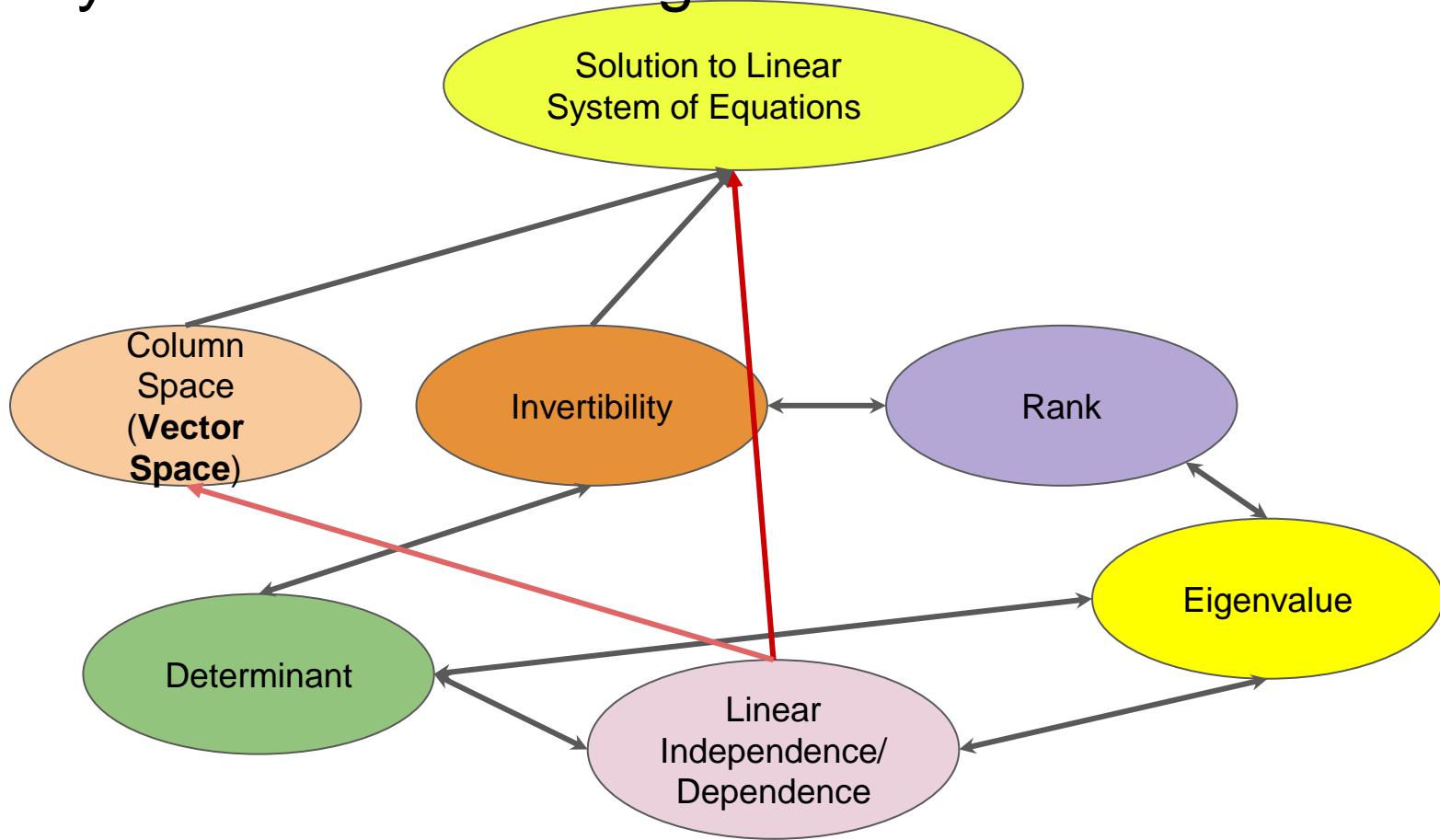
$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 10 \end{bmatrix}$$

Matrix(Square Matrix) vector multiplication can be seen as follows:

- Rotation
- Stretching or Shrinking



Beauty Lies in Connecting Ideas



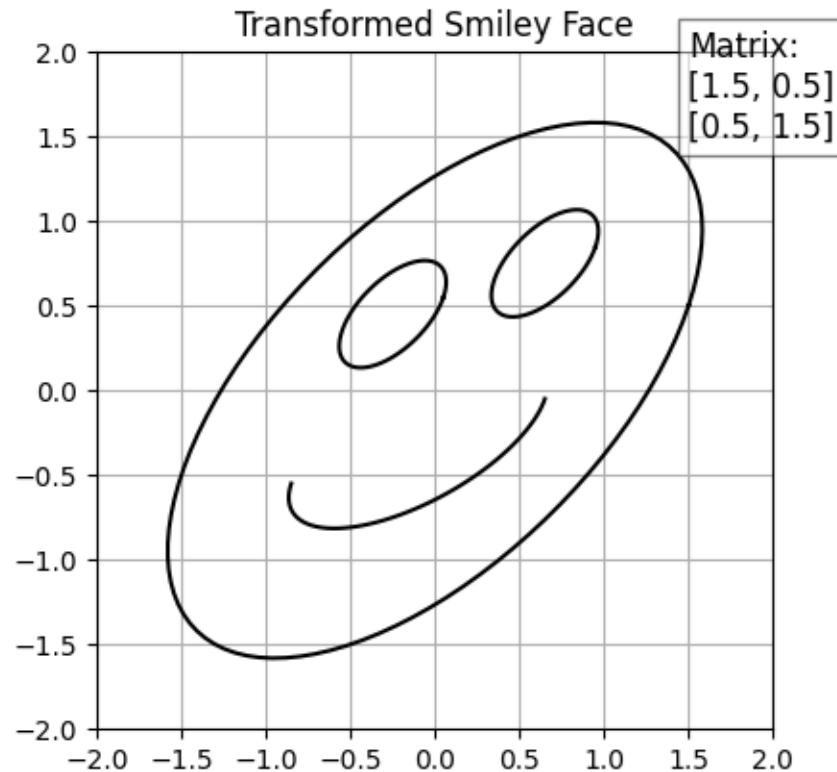
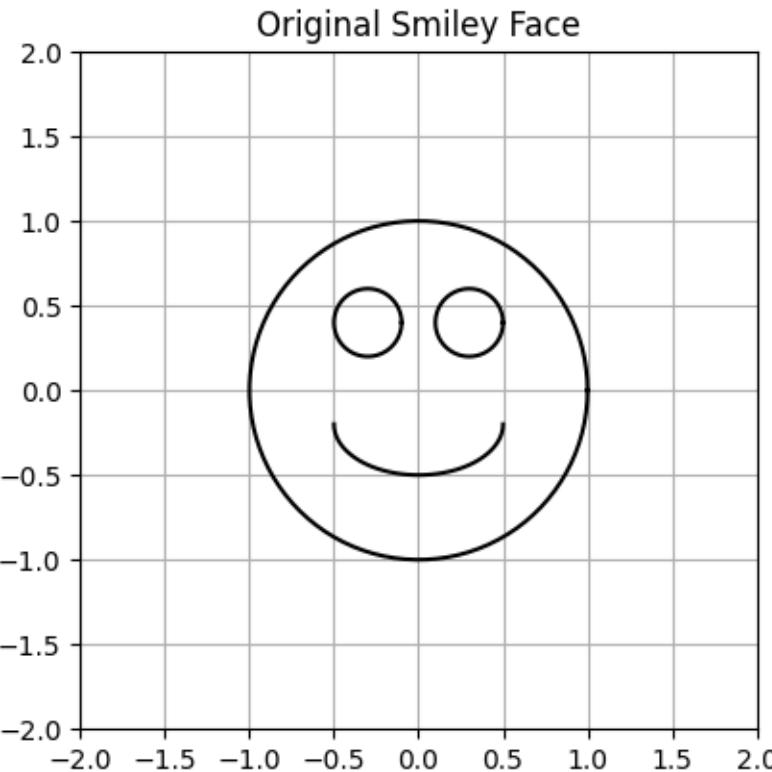
Matrix- vector multiplication

$$-x + y = 0$$

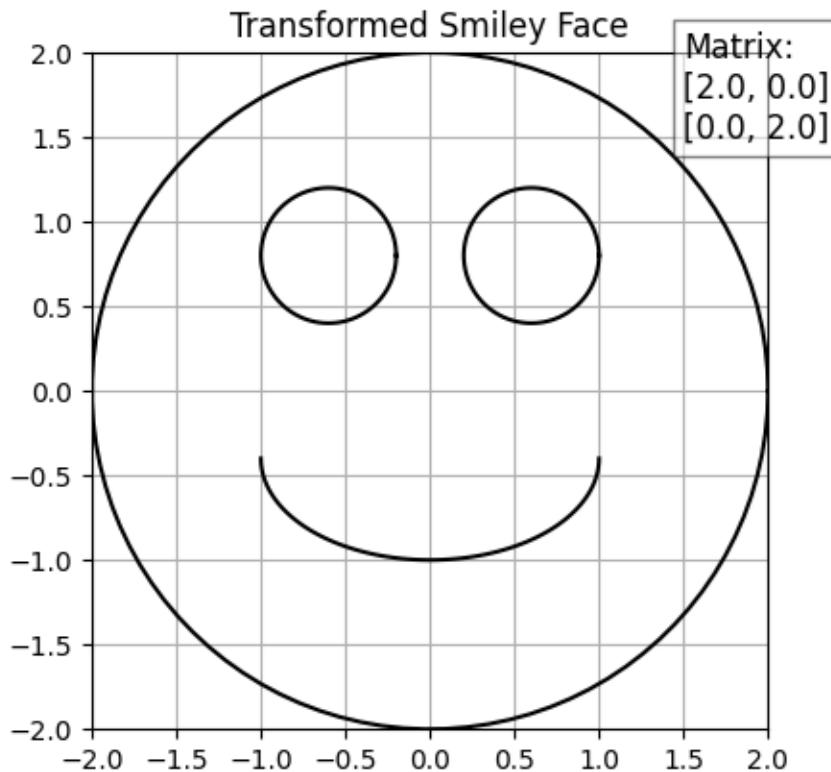
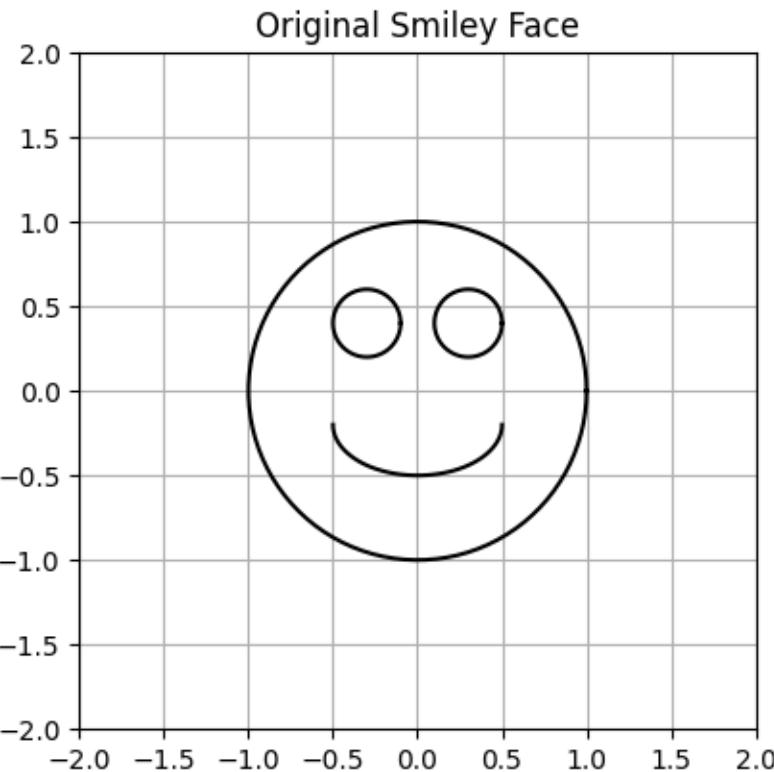
$$2x + y = 3$$

Dot Product

Geometric Interpretation of Matrix Multiplication

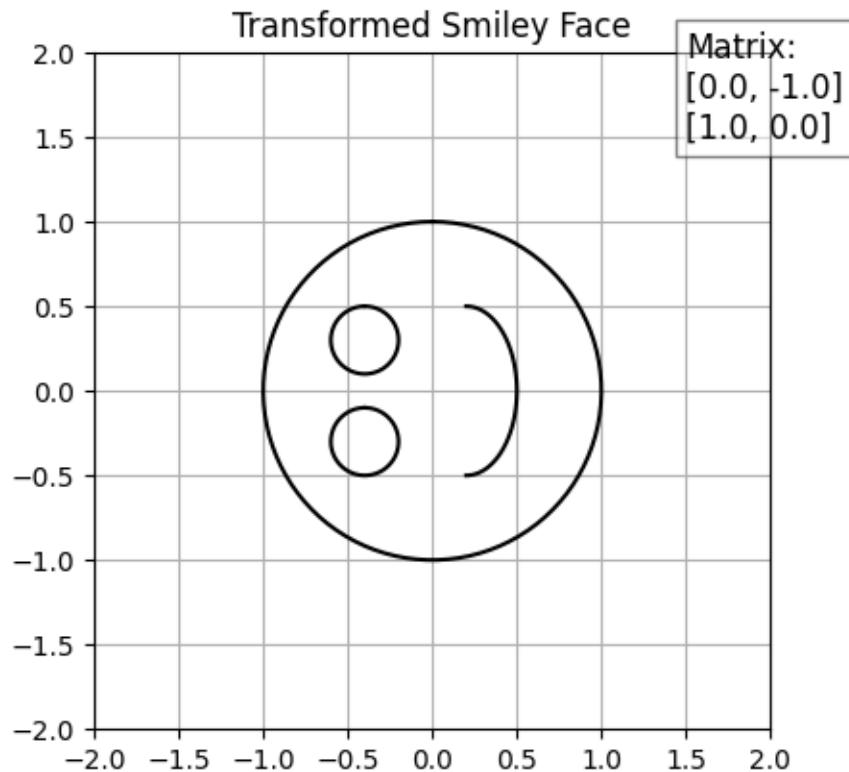
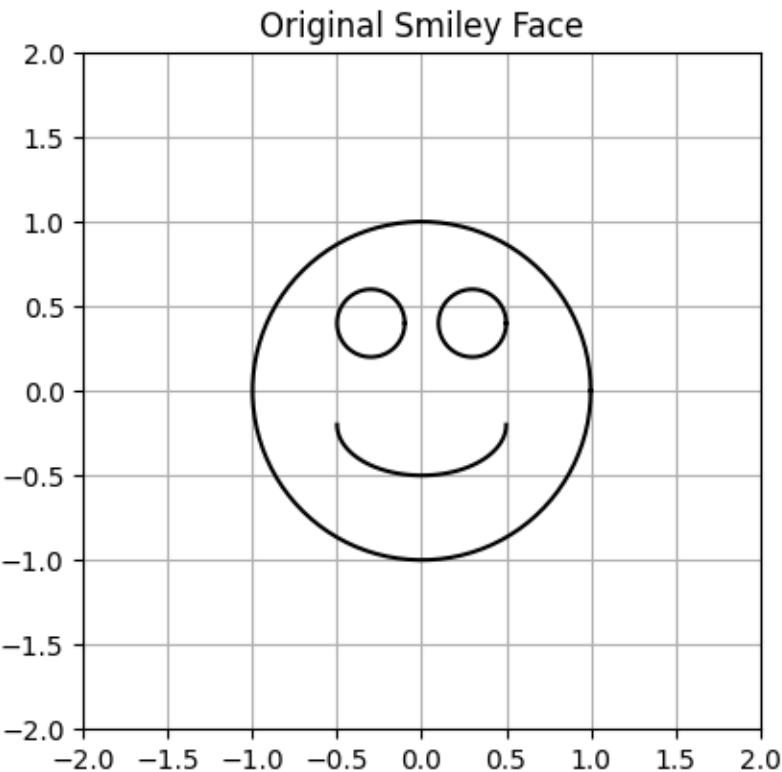


Geometric Interpretation of Matrix Multiplication



What should I do if I want to rotate the smiley by 90 degree?

What should I do if I want to rotate the smiley by 90 degree?



What can you say about the following transformation?

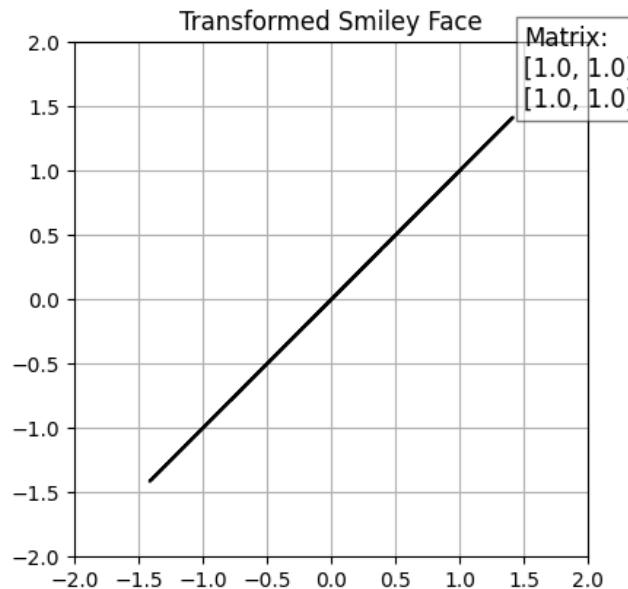
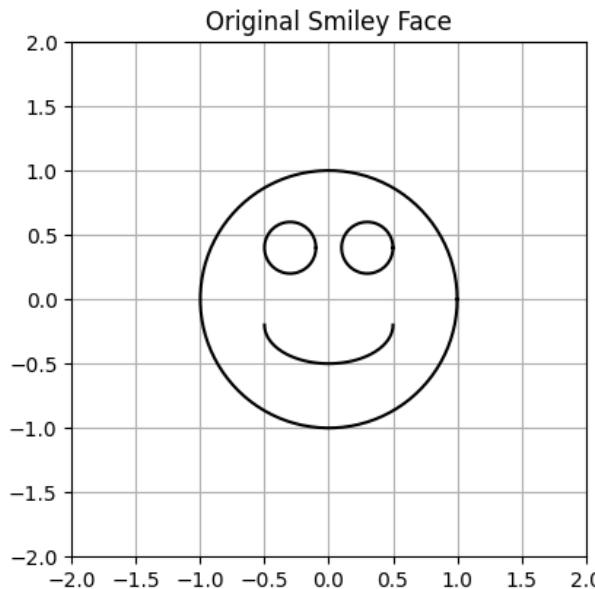
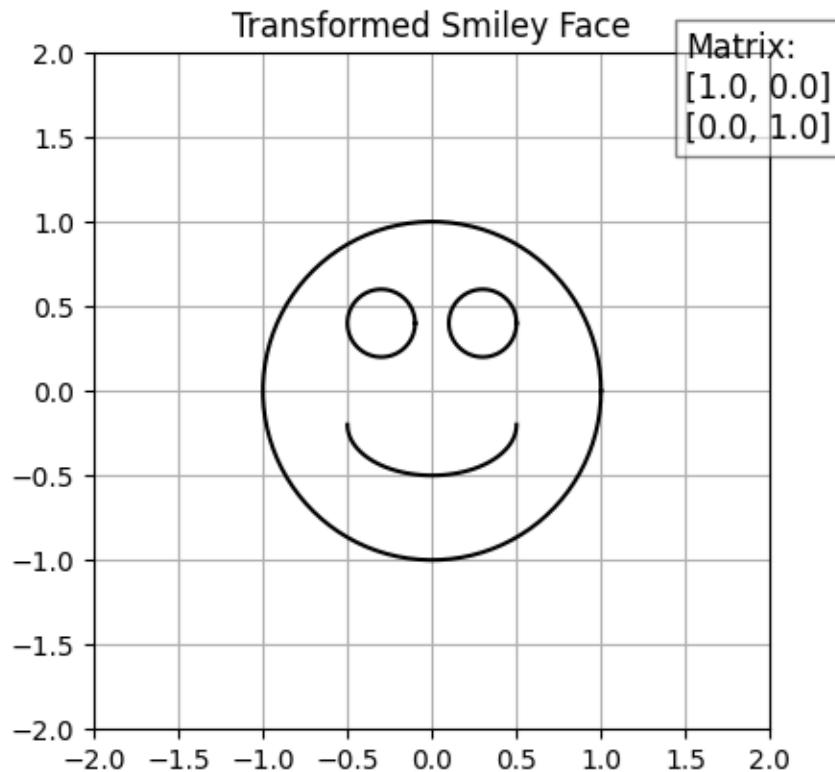
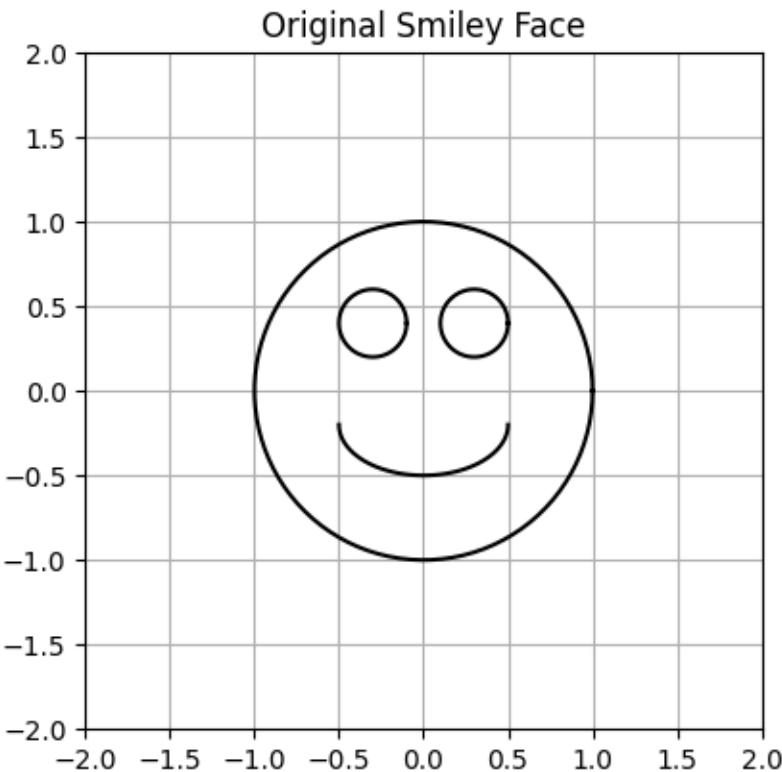


Figure out the transformation (matrix) that will preserve the smiley as it is!

Figure out the transformation (matrix) that will preserve the smiley as it is!



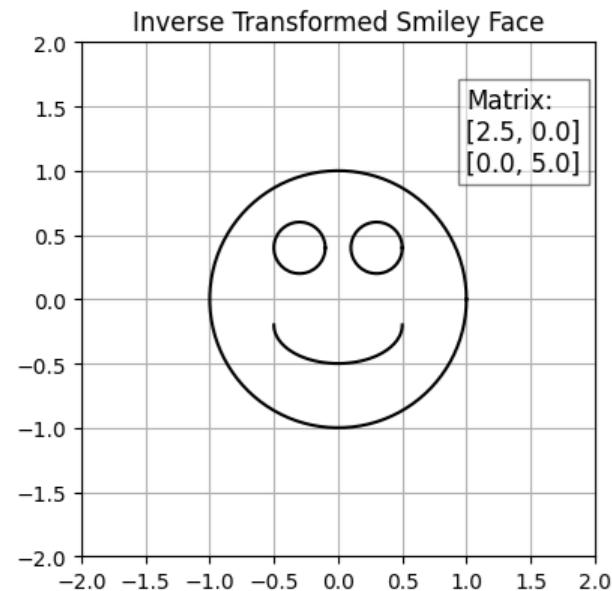
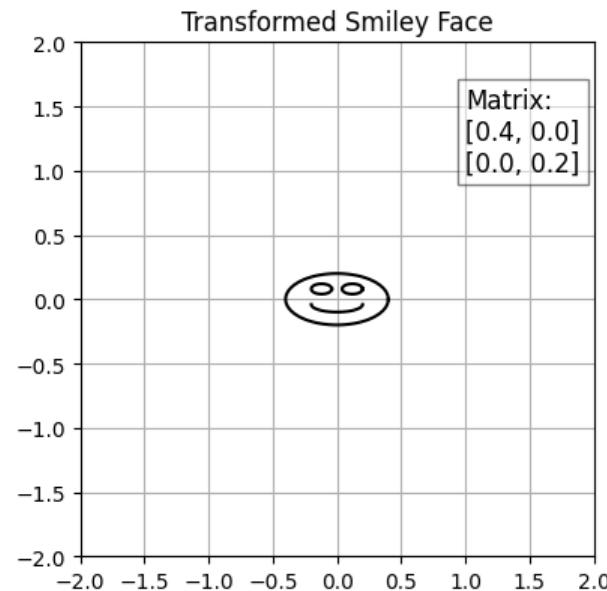
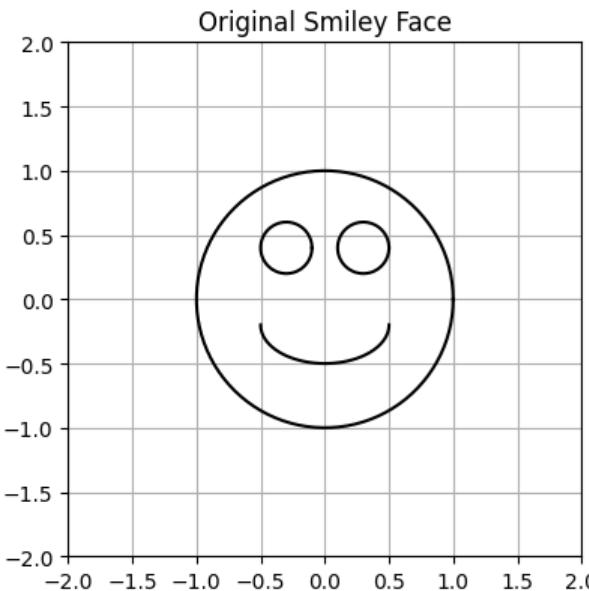
Identity Matrix

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

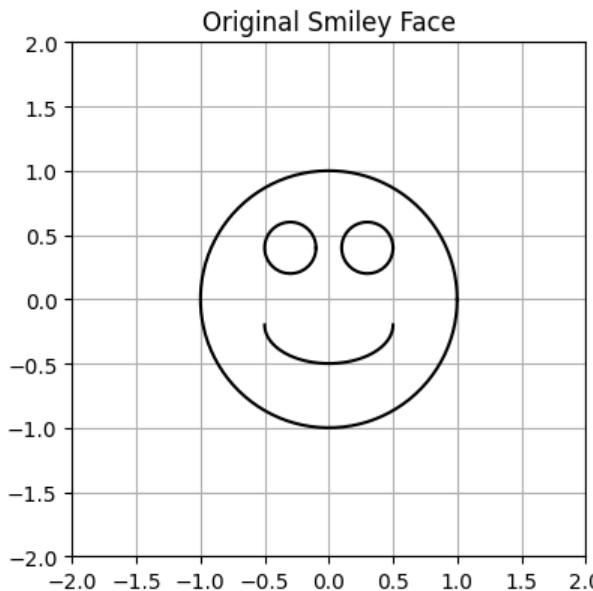
Inverse of a Matrix

$$AA^{-1} = A^{-1}A = I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

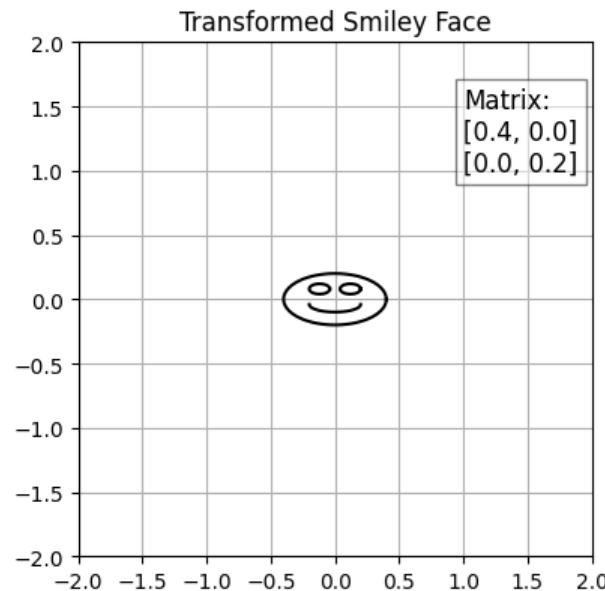
How do we understand matrix inverse geometrically?



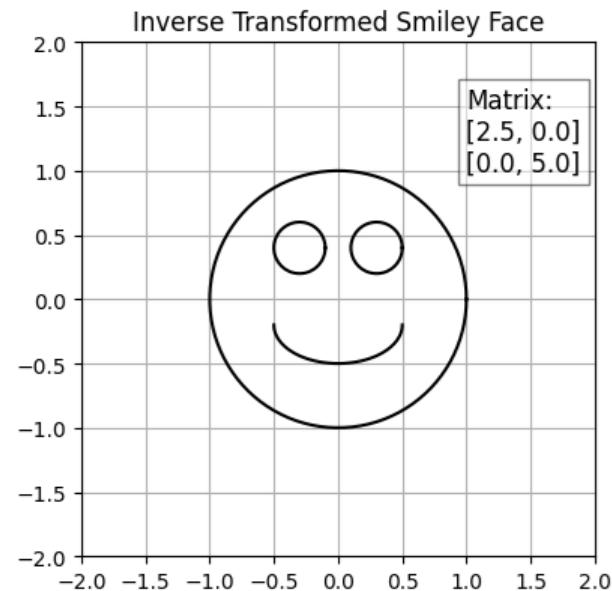
In a Nutshell



$$\vec{x}_i$$

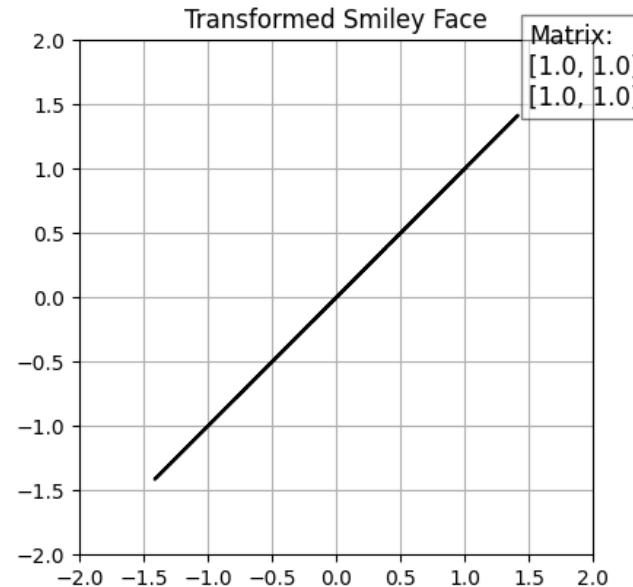
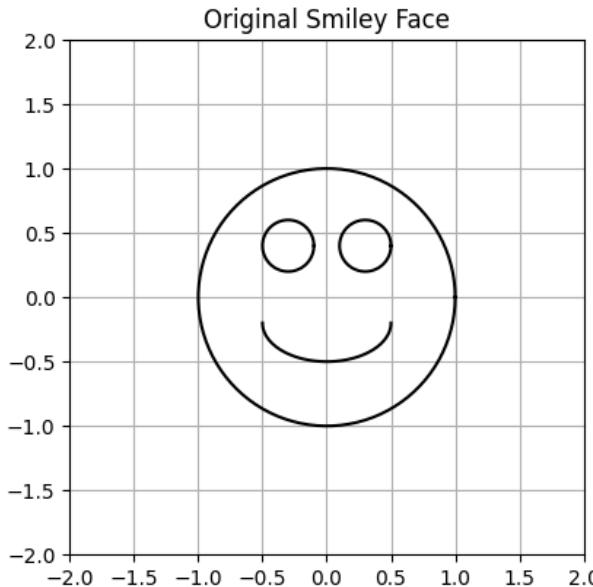


$$A\vec{x}_i$$



$$A^{-1}A\vec{x}_i$$

Can you get back the original smiley?



$$\vec{x}_i$$

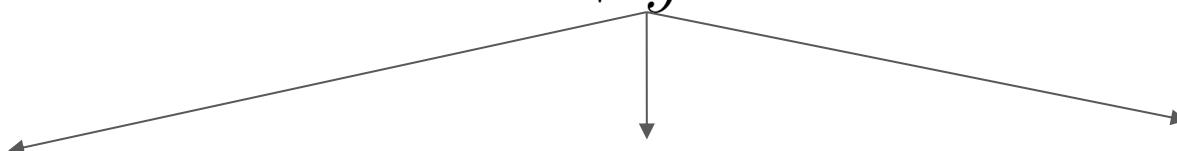
$$A\vec{x}_i$$

?

$$A^{-1}A\vec{x}_i$$

Solutions to System of Linear Equations

$$\begin{aligned}-x + y &= 0 \\ 2x + y &= 3\end{aligned}$$



Algebraic Interpretation

Row Picture

Column Picture

