MATH + AI + FINANCE

How Math Powers AI in Everyday Finance

From Your Morning Coffee to Your Credit Score

Prof. Jörg Osterrieder

45 MINUTES

7 FORMULAS

3 GAMES

ACT 1

The Hook

You Already Use AI Finance

50–200 AI Decisions Before Breakfast

Industry estimates say you encounter 50–200 AI-driven financial decisions before you even leave for school.

50–200 AI Decisions Before Breakfast

Industry estimates say you encounter 50–200 AI-driven financial decisions before you even leave for school.

6:30 AM Phone unlocks → face recognition authenticates your bank app

6:45 AM Spotify suggests a playlist → recommendation algorithm

7:00 AM Mom pays for groceries → fraud detection scans the transaction

7:15 AM Insurance premium recalculated → risk model updated overnight

7:30 AM Bus route optimized → real-time pricing algorithm

The Invisible Math

1

Transaction

Scanned

The Invisible Math

1

TRANSACTION SCANNED

12

PRICES ADJUSTED

The Invisible Math

1
Transaction Scanned

12
Prices Adjusted

49
Recommendations Served

The Invisible Math

1
Transaction Scanned

12
Prices Adjusted

49
Recommendations Served

And it's not even 8 AM. Every single one of these decisions is powered by a formula you could learn today.

You Already Think Like an AI

The cafeteria problem:

- You see a long lunch line
- You estimate wait time (regression)
- You weigh options: wait vs. skip (optimization)
- You check if your friend is in line (pattern matching)
- You decide (classification)

You Already Think Like an AI

The cafeteria problem:

- You see a long lunch line
- You estimate wait time (regression)
- You weigh options: wait vs. skip (optimization)
- You check if your friend is in line (pattern matching)
- You decide (classification)

You just performed 4 types of machine learning
— in about 3 seconds.

Regression → "How long?"
Optimization → "Best option?"
Pattern match → "Seen before?"
Classification → "Go / No go"

How Do You Spot Something Unusual?

> *"Something feels off about this."*
>
> — Your brain, every day

You notice:

- A friend acting strangely
- An unusual charge on a bank statement
- A price that seems too good to be true

How Do You Spot Something Unusual?

> *"Something feels off about this."*
>
> — Your brain, every day

You notice:

- A friend acting strangely
- An unusual charge on a bank statement
- A price that seems too good to be true

AI notices the same way:

- Deviation from normal pattern
- Statistical outlier detected
- Probability below threshold

Florence Nightingale

HISTORICAL VIGNETTE

1858

FN

Florence Nightingale

1820–1910

Not just a nurse — a data scientist. Nightingale invented the polar area diagram (coxcomb chart) to prove that soldiers were dying from preventable disease, not battlefield wounds.

Her data visualization *changed government policy* and saved thousands of lives. She proved that **showing the math** can be more powerful than any argument.

Meet BankBot

YOUR AI GUIDE

[BOT] *I analyzed your breakfast. You are 94% human. The other 6% is toast crumbs.*

BankBot will join us throughout the talk. It starts overconfident, gets humbled by real data, and learns — just like a real AI.

Meet BankBot

YOUR AI GUIDE

[BOT] *I analyzed your breakfast. You are 94% human. The other 6% is toast crumbs.*

BankBot will join us throughout the talk. It starts overconfident, gets humbled by real data, and learns — just like a real AI.

[BOT] *I have processed 4.7 billion transactions. How hard can high school math be?*

The Journey: Six Acts

Act 1: The Hook
Morning timeline, you think like an AI

Act 2: The Building Blocks
2000 years of math $\rightarrow$ one AI model

Act 3: The Pattern Engine
Fraud detection, Bayes, the bell curve

Act 4: Predictions & Decisions
Credit scores, recommendations

Act 5: The Bigger Picture
LLM stories, ethics, fairness

Act 6: Your Future
Careers, resources, your super-power

ACT 2

The Building Blocks

2000 Years of Math That Accidentally Built AI

The First Spreadsheet

THREAD 1: LINEAR ALGEBRA

∼100 BCE

The First Spreadsheet

Thread 1: Linear Algebra

~100 BCE

~100 BCE Jiuzhang Suanshu — “Nine Chapters”

Chinese merchants used bamboo counting rods for inventory, taxation, and land surveying. They laid rods in rows and columns to solve systems of equations — Gaussian elimination, 2,000 years before Gauss.

Arrange rods:

$$\begin{pmatrix} 3 & 2 & 1 \\ 2 & 3 & 1 \\ 1 & 2 & 3 \end{pmatrix}$$

Eliminate row by row...

The First Spreadsheet

THREAD 1: LINEAR ALGEBRA

$\sim$100 BCE

$\sim$100 BCE Jiuzhang Suanshu — “Nine Chapters”

Chinese merchants used bamboo counting rods for inventory, taxation, and land surveying. They laid rods in rows and columns to solve systems of equations — Gaussian elimination, 2,000 years before Gauss.

Han Dynasty context: The Silk Road is opening. Trade demands systematic accounting. Someone invents matrix operations to manage the books.

Arrange rods:

$$\begin{pmatrix} 3 & 2 & 1 \\ 2 & 3 & 1 \\ 1 & 2 & 3 \end{pmatrix}$$

Eliminate row by row...

Suspect: Hermann Grassmann

THREAD 1

1844

HG

Hermann Grassmann

1809–1877

Occupation: Schoolteacher in Stettin, Prussia

Suspect: Hermann Grassmann

THREAD 1

1844

HG

Hermann Grassmann

1809–1877

Occupation: Schoolteacher in Stettin, Prussia

1844: Publishes *Ausdehnungslehre* — invents vector spaces, linear independence, dimension. Nobody reads it. Gives up math, becomes a Sanskrit scholar.

Suspect: Hermann Grassmann

THREAD 1

1844

HG

Hermann Grassmann

1809–1877

Occupation: Schoolteacher in Stettin, Prussia

1844: Publishes *Ausdehnungslehre* — invents vector spaces, linear independence, dimension. Nobody reads it. Gives up math, becomes a Sanskrit scholar.

Finance: Today every bank customer is a vector — a list of numbers (income, age, credit history, spending). Grassmann invented this representation.

The Formula: Matrix $\times$ Vector

THREAD 1

$$\mathbf{y} = W\mathbf{x} + \mathbf{b}$$

Matrix $\times$ vector $+$ bias
The fundamental neural network operation

The Formula: Matrix $\times$ Vector

Thread 1

$$\mathbf{y} = W\mathbf{x} + \mathbf{b}$$

Matrix $\times$ vector $+$ bias

The fundamental neural network operation

Worked example:

$$\mathbf{x} = \begin{pmatrix} 50{,}000 \\ 720 \end{pmatrix}$$ (income, credit)

$W = \begin{pmatrix} 0.3 & 0.7 \\ \ldots \end{pmatrix}$

The Formula: Matrix $\times$ Vector

THREAD 1

$$\mathbf{y} = W\mathbf{x} + \mathbf{b}$$

Matrix $\times$ vector $+$ bias

The fundamental neural network operation

Worked example:

$$\mathbf{x} = \begin{pmatrix} 50{,}000 \\ 720 \end{pmatrix} \text{ (income, credit)}$$

$$W = \begin{pmatrix} 0.3 & 0.7 \end{pmatrix}$$

1.8T

MULTIPLICATIONS PER TOKEN

Cayley (1858) formalized matrix multiplication *while practicing law*. 900+ papers.

His notation executes 1.8 trillion times per GPT-4 word.

LLM Connection: Words Are Geometry

THREAD 1 $\longrightarrow$ LLM

$$\vec{\text{King}} - \vec{\text{Man}} + \vec{\text{Woman}} \approx \vec{\text{Queen}}$$

LLM Connection: Words Are Geometry

THREAD 1 $\longrightarrow$ LLM

$$\vec{\text{King}} - \vec{\text{Man}} + \vec{\text{Woman}} \approx \vec{\text{Queen}}$$

Word2Vec (2013): Every word $\rightarrow$ 300+ numbers.
GPT-4: 12,288 dimensions.
Meaning *is* direction in space.

LLM Connection: Words Are Geometry

THREAD 1 $\rightarrow$ LLM

$$\vec{\text{King}} - \vec{\text{Man}} + \vec{\text{Woman}} \approx \vec{\text{Queen}}$$

Word2Vec (2013): Every word $\rightarrow$ 300+ numbers.
GPT-4: 12,288 dimensions.
Meaning *is* direction in space.

The chain:

Accounting rods (100 BCE)
$\downarrow$ Grassmann's vectors (1844)
$\downarrow$ Cayley's matrices (1858)
$\downarrow$ Word embeddings (2013)
$\downarrow$ **Every LLM today**

169 years: Grassmann to

The Insurance Problem

THREAD 2: PROBABILITY

1654

The Insurance Problem

THREAD 2: PROBABILITY

1654

1654 Chevalier de Méré loses money gambling

Complains to Pascal: “The math says I should win!” Pascal writes to Fermat. Their letters invent probability.

The question:
Two players, interrupted game.
How to split the pot *fairly*?

The Insurance Problem

THREAD 2: PROBABILITY

1654

1654 Chevalier de Méré loses money gambling

Complains to Pascal: “The math says I should win!” Pascal writes to Fermat. Their letters invent probability.

Finance: Their “Problem of Points” — splitting a pot fairly — became insurance pricing. Every premium traces back to two Frenchmen arguing about dice.

The question:
Two players, interrupted game. How to split the pot *fairly*?

Suspect: Thomas Bayes

THREAD 2

1763

TB

Thomas Bayes

1702–1761

Occupation: Presbyterian minister

Suspect: Thomas Bayes

THREAD 2

1763

TB

Thomas Bayes

1702–1761

Occupation: Presbyterian minister

Published almost nothing. Theorem found after death (1763).

$$P(H \mid E) = \frac{P(E|H)\,P(H)}{P(E)}$$

Suspect: Thomas Bayes

THREAD 2

1763

TB

Thomas Bayes

1702–1761

Occupation: Presbyterian minister

Published almost nothing. Theorem found after death (1763).

$$P(H \mid E) = \frac{P(E|H)\,P(H)}{P(E)}$$

Finance: Foundation of every real-time risk assessment. When your card is swiped, Bayes updates fraud probability in ms.

The portrait may not even be him.

The Formula: Softmax

THREAD 2

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

Any numbers $\rightarrow$ valid probability distribution

The Formula: Softmax

THREAD 2

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

Any numbers $\rightarrow$ valid probability distribution

Example: $[2.0, 1.0, 0.1]$
$\rightarrow [0.66, 0.24, 0.10]$
Non-negative, sums to 1.

The Formula: Softmax

THREAD 2

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

Any numbers $\rightarrow$ valid probability distribution

Example: $[2.0, 1.0, 0.1]$
$\rightarrow [0.66, 0.24, 0.10]$
Non-negative, sums to 1.

Kolmogorov (1933): Three axioms for probability. Softmax satisfies them exactly.

LLM Connection: Every Word Is a Lottery

THREAD 2 $\longrightarrow$ LLM

Every time ChatGPT writes a word, softmax runs over 50,000+ vocabulary entries. Highest probability wins. Temperature controls randomness.

LLM Connection: Every Word Is a Lottery

THREAD 2 → LLM

Every time ChatGPT writes a word, softmax runs over 50,000+ vocabulary entries. Highest probability wins. Temperature controls randomness.

Temp = 0: Always top word. Robotic.
Temp = 1: Proportional sampling. Creative.

LLM Connection: Every Word Is a Lottery

THREAD 2 → LLM

Every time ChatGPT writes a word, softmax runs over 50,000+ vocabulary entries. Highest probability wins. Temperature controls randomness.

Temp = 0: Always top word. Robotic.
Temp = 1: Proportional sampling. Creative.

The chain:

Gambling (1654)
↓ Bayes (1763)
↓ Kolmogorov (1933)
↓ Softmax
↓ **Every AI word**

Every conversation is weighted dice rolls.

Newton’s Plague Year vs. Leibniz

THREAD 3: CALCULUS

1666 / 1684

Newton’s Plague Year vs. Leibniz

THREAD 3: CALCULUS

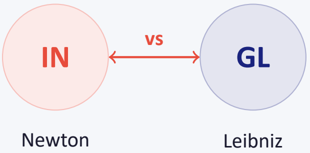
1666 / 1684

Newton (1666): Calculus during the plague. Hid it.

Leibniz (1684): Published first. Notation: $\frac{dy}{dx}$.

Scandal: Newton rigged the investigation.

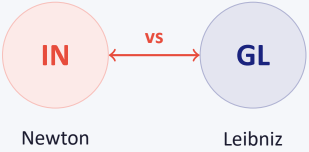
We use Leibniz’s notation.



Newton’s Plague Year vs. Leibniz

THREAD 3: CALCULUS

1666 / 1684



Newton (1666): Calculus during the plague. Hid it.

Leibniz (1684): Published first. Notation: $\frac{dy}{dx}$.

Scandal: Newton rigged the investigation.

We use Leibniz’s notation.

Finance: “Derivatives” are literally named after calculus. Options pricing, risk sensitivity, portfolio optimization.

Suspect: Cauchy's Gradient Descent

THREAD 3

1847

ALC

Augustin-Louis Cauchy

1789–1857

Occupation: French mathematician, prolific

Suspect: Cauchy's Gradient Descent

THREAD 3

1847

ALC

Augustin-Louis Cauchy

1789–1857

Occupation: French mathematician, prolific

1847: Gradient descent — step in the direction that reduces error, repeat.

"Blindfolded on a hill: feel downhill, step, repeat."

Suspect: Cauchy's Gradient Descent

THREAD 3

1847

ALC

Augustin-Louis Cauchy

1789–1857

Occupation: French mathematician, prolific

1847: Gradient descent — step in the direction that reduces error, repeat.

"Blindfolded on a hill: feel downhill, step, repeat."

Finance: Same algorithm calibrates risk models — trains fraud detectors and trading algorithms.

The Formula: Gradient Descent + Chain Rule

Thread 3

$$\theta_{t+1} = \theta_t - \eta \nabla L(\theta_t)$$

Parameters = cur-
rent − rate × gradient

The Formula: Gradient Descent + Chain Rule

THREAD 3

$$\theta_{t+1} = \theta_t - \eta \nabla L(\theta_t)$$

Parameters $=$ current $-$ rate $\times$ gradient

Derivative $=$ direction

η $=$ step size

Repeat billions of times

The Formula: Gradient Descent + Chain Rule

THREAD 3

$$\theta_{t+1} = \theta_t - \eta \nabla L(\theta_t)$$

Parameters $=$ current $-$ rate $\times$ gradient

Derivative $=$ direction

$\eta =$ step size

Repeat billions of times

Chain rule (Leibniz):

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial y} \cdot \frac{\partial y}{\partial w}$$

Enables backpropagation: gradients layer by layer, backwards.

LLM Connection: Hinton's Nobel Prize

THREAD 3 $\longrightarrow$ LLM

GH

Geoffrey Hinton

b. 1947

1986: Backpropagation paper in *Nature*.

LLM Connection: Hinton's Nobel Prize

THREAD 3 → LLM

GH

Geoffrey Hinton

b. 1947

1986: Backpropagation paper in *Nature*.

Decades of rejection. "Neural nets are dead." Persists.

2024: Nobel Prize in Physics.

LLM Connection: Hinton's Nobel Prize

THREAD 3 $\longrightarrow$ LLM

GH

Geoffrey Hinton

b. 1947

1986: Backpropagation paper in *Nature*.

Decades of rejection. "Neural nets are dead." Persists.

2024: Nobel Prize in Physics.

The chain: Newton (1666) $\to$ Leibniz (1684) $\to$ Cauchy (1847) $\to$ Hinton (1986) $\to$ **every LLM**.

Chain rule $\to$ 338 years $\to$ Nobel Prize.

The Unicycling Genius

THREAD 4: INFORMATION THEORY

1948

CS

Claude Shannon

1916–2001

Occupation: Bell Labs engineer, MIT professor

The Unicycling Genius

THREAD 4: INFORMATION THEORY

1948

CS

Claude Shannon

1916–2001

Occupation: Bell Labs engineer, MIT professor

1948: One paper creates entire field. Invents the bit, entropy, channel capacity.

The Unicycling Genius

THREAD 4: INFORMATION THEORY

1948

CS

Claude Shannon

1916–2001

Occupation: Bell Labs engineer, MIT professor

1948: One paper creates entire field. Invents the bit, entropy, channel capacity.

Also: juggling unicyclist, flame-throwing trumpet, machine that turns itself off.

Finance: His math prices every encrypted banking transaction.

The Formula: Cross-Entropy

THREAD 4

$$H(P, Q) = -\sum_x P(x) \log Q(x)$$

How **surprised** is the model?

The Formula: Cross-Entropy

THREAD 4

$$H(P, Q) = -\sum_x P(x) \log Q(x)$$

How **surprised** is the model?

P = reality, Q = prediction

Minimize surprise = learn language

The Formula: Cross-Entropy

Thread 4

$$H(P, Q) = -\sum_{x} P(x) \log Q(x)$$

How **surprised** is the model?

P = reality, Q = prediction

Minimize surprise = learn language

Shannon designed this for telephones.

Same formula, 76 years later, trains every LLM.

Finance: "How surprised was the model by these earnings?"

Shannon's Blueprint $\rightarrow$ ChatGPT

THREAD 4

Shannon's Blueprint → ChatGPT

THREAD 4

SHANNON 1948

Source → Encoder → Channel → Decoder → Destination

Shannon's Blueprint → ChatGPT

THREAD 4

SHANNON 1948

Source → Encoder → Channel → Decoder → Destination

CHATGPT 2024

User → Tokenizer → Transformer → Output → Response

Shannon's Blueprint → ChatGPT

Thread 4

SHANNON 1948

Source → Encoder → Channel → Decoder → Destination

CHATGPT 2024

User → Tokenizer → Transformer → Output → Response

Same architecture. Neural networks replace components. Telegrams became text generation.

Predicting English, 74 Years Apart

THREAD 4 $\longrightarrow$ LLM

Shannon, 1951: Predict text letter by letter. Entropy: $\sim$1.1 bits/char (of 4.7 max).

He described what ChatGPT does: **minimize uncertainty about the next token** — 74 years early.

Predicting English, 74 Years Apart

THREAD 4 $\rightarrow$ LLM

Shannon, 1951: Predict text letter by letter. Entropy: $\sim$1.1 bits/char (of 4.7 max).

He described what ChatGPT does: **minimize uncertainty about the next token** — 74 years early.

$$\text{Perplexity} = 2^{H(P,Q)}$$

$$\text{Perplexity } 10 = \text{“choosing from 10 options”}$$

Shannon would recognize ChatGPT — it's his experiment at scale.

Three Breakthroughs

Thread 5: Optimization

Three Breakthroughs

THREAD 5: OPTIMIZATION

1951 Robbins & Monro — **SGD**

Random samples instead of all data. Noisy but fast.

Three Breakthroughs

Thread 5: Optimization

1951 Robbins & Monro — **SGD**

Random samples instead of all data. Noisy but fast.

1964 Polyak — **Momentum**

Ball rolling downhill with inertia.

Three Breakthroughs

THREAD 5: OPTIMIZATION

1951 Robbins & Monro — **SGD**

Random samples instead of all data. Noisy but fast.

1964 Polyak — **Momentum**

Ball rolling downhill with inertia.

2014 Kingma & Ba — **Adam**

Adaptive rates per parameter. 200K+ citations.

200K+

CITATIONS

Without these: centuries.
With them: weeks.

Finance: Hedge funds re-train overnight.

Suspect: Diederik Kingma

THREAD 5

2014

DK

Diederik Kingma

b. 1987

PhD student when he published Adam.

Suspect: Diederik Kingma

THREAD 5

2014

DK

Diederik Kingma

b. 1987

PhD student when he published Adam.

Default optimizer of deep learning — trains GPT, DALL-E, AlphaFold.

Suspect: Diederik Kingma

THREAD 5

2014

DK

Diederik Kingma

b. 1987

PhD student when he published Adam.

Default optimizer of deep learning — trains GPT, DALL-E, AlphaFold.

Finance: Every credit scorer and fraud detector in the last decade used Adam.

A PhD student's paper $\rightarrow$ engine of a \$3T industry.

The Formula: Scaling Laws

Thread 5

$$L(N) = \left(\frac{N_c}{N}\right)^{\alpha_N}$$

Loss $\downarrow$ as power law of model size

The Formula: Scaling Laws

THREAD 5

$$L(N) = \left(\frac{N_c}{N}\right)^{\alpha_N}$$

Loss $\downarrow$ as power law of model size

Kaplan et al., 2020

More parameters $\rightarrow$ better.
More data $\rightarrow$ better.
Billions spent on this bet.

The Formula: Scaling Laws

THREAD 5

$$L(N) = \left(\frac{N_c}{N}\right)^{\alpha_N}$$

Loss $\downarrow$ as power law of model size

Kaplan et al., 2020

More parameters $\rightarrow$ better.
More data $\rightarrow$ better.
Billions spent on this bet.

DeepSeek: Matched GPT-4 for $6M compute vs. $100M+ budget.

Efficiency bends the curve.

Finance: Scaling = economics of AI.

The Cost of Intelligence

Thread 5 → LLM

$100M+	$30M	$5.6M
GPT-4	Llama 3	DeepSeek R1

The Cost of Intelligence

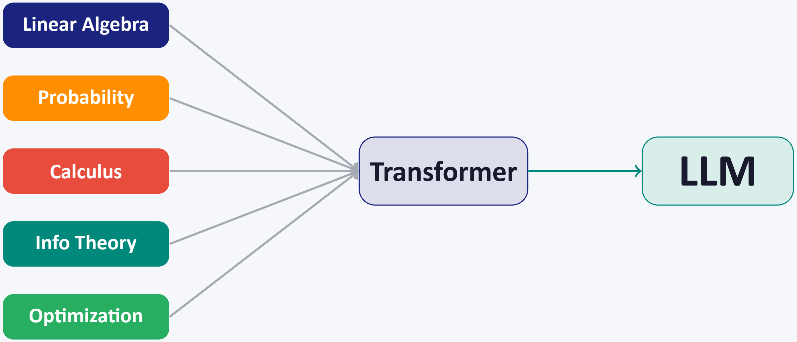
THREAD 5 $\to$ LLM

$100M+	$30M	$5.6M
GPT-4	LLAMA 3	DEEPSEEK R1

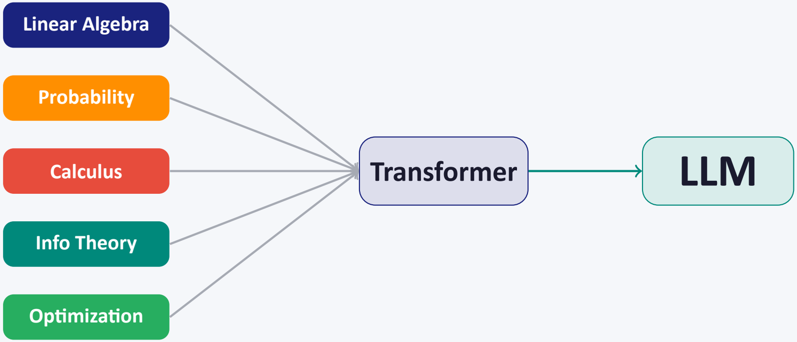
Without SGD, momentum, and Adam: centuries. With them: months.

Finance: AI training costs are the new capex. Optimization math $=$ economics of AI.

2000 Years $\rightarrow$ One Model



2000 Years → One Model



> *"None of them knew they were building AI.*
> *All of them made it possible."*

ACT 3

The Pattern Engine

Math Finds What Humans Miss

Catching the Weird Stuff

$32B

ANNUAL CREDIT CARD FRAUD WORLDWIDE

[BOT] *Too easy. Next.*

Catching the Weird Stuff

$32B

ANNUAL CREDIT CARD FRAUD WORLDWIDE

Banks process **billions** of transactions per day.
They have about 50 milliseconds to decide: legit or fraud?

[BOT] *Too easy. Next.*

Catching the Weird Stuff

$32B

Annual Credit Card Fraud Worldwide

Banks process **billions** of transactions per day.
They have about 50 milliseconds to decide: legit or fraud?

The answer? Three formulas from your math class:

Bell Curve $\longrightarrow$ Bayes' Theorem $\longrightarrow$ Sigmoid Function

[BOT] *Too easy. Next.*

Bayes' Theorem: Updating Your Beliefs

$$P(\text{Fraud} \mid \text{Data}) = \frac{P(\text{Data} \mid \text{Fraud}) \cdot P(\text{Fraud})}{P(\text{Data})}$$

Bayes' Theorem: Updating Your Beliefs

$$P(\text{Fraud} \mid \text{Data}) = \frac{P(\text{Data} \mid \text{Fraud}) \cdot P(\text{Fraud})}{P(\text{Data})}$$

In plain English:

Start with what you already know (prior).

See new evidence.

Update your belief.

Bayes' Theorem: Updating Your Beliefs

$$P(\text{Fraud} \mid \text{Data}) = \frac{P(\text{Data} \mid \text{Fraud}) \cdot P(\text{Fraud})}{P(\text{Data})}$$

In plain English:
Start with what you already know (prior).
See new evidence.
Update your belief.

The school analogy:

Your friend is usually on time (prior: 95%).
Today they're 20 minutes late.
There's a traffic jam on their route.

You *update* your belief:
"They're probably stuck in traffic"
— not "they overslept."

The Numbers: Is This Transaction Fraud?

Setup: 1 in 1,000 transactions is fraud $\rightarrow P(\text{Fraud}) = 0.001$
The AI catches 99% of fraud $\rightarrow P(\text{Flag} \mid \text{Fraud}) = 0.99$
But it false-alarms on 2% of legit ones $\rightarrow P(\text{Flag} \mid \neg\text{Fraud}) = 0.02$

The Numbers: Is This Transaction Fraud?

Setup: 1 in 1,000 transactions is fraud $\rightarrow P(\text{Fraud}) = 0.001$
The AI catches 99% of fraud $\rightarrow P(\text{Flag} \mid \text{Fraud}) = 0.99$
But it false-alarms on 2% of legit ones $\rightarrow P(\text{Flag} \mid \neg\text{Fraud}) = 0.02$

$$P(\text{Fraud} \mid \text{Flag}) = \frac{0.99 \times 0.001}{(0.99 \times 0.001) + (0.02 \times 0.999)} \approx \mathbf{4.7\%}$$

The Numbers: Is This Transaction Fraud?

Setup: 1 in 1,000 transactions is fraud $\rightarrow P(\text{Fraud}) = 0.001$
The AI catches 99% of fraud $\rightarrow P(\text{Flag} \mid \text{Fraud}) = 0.99$
But it false-alarms on 2% of legit ones $\rightarrow P(\text{Flag} \mid \neg\text{Fraud}) = 0.02$

$$P(\text{Fraud} \mid \text{Flag}) = \frac{0.99 \times 0.001}{(0.99 \times 0.001) + (0.02 \times 0.999)} \approx \mathbf{4.7\%}$$

Even with a 99% accurate test, only $\sim$5% of flagged transactions are actually fraud! The other 95% are false alarms.

But Why Bother?

47 ×

Better Than

Random

Without the AI: only 0.1% of transactions are fraud.
With the AI: 4.7% of flagged ones are fraud.

That's 47 times better than random.

But Why Bother?

47×

BETTER THAN RANDOM

Without the AI: only 0.1% of transactions are fraud.
With the AI: 4.7% of flagged ones are fraud.

That's 47 times better than random.

The real question: What's worse?

Missing real fraud (€10,000+ stolen)
vs. A 30-second verification call to a customer

In AI, the question is never "is it perfect?" but "is it better than the alternative?"

The Bell Curve: What "Normal" Looks Like

"If a transaction is far from
the center of the bell curve,
something unusual is happening."

The Bell Curve: What "Normal" Looks Like

"If a transaction is far from
the center of the bell curve,
something unusual is happening."

Your spending has a pattern:
€20–60 at the grocery store = normal
€3,500 on luxury watches at 3 AM = out-
lier

The Bell Curve: What "Normal" Looks Like

"If a transaction is far from the center of the bell curve, something unusual is happening."

Your spending has a pattern:
€20–60 at the grocery store = normal
€3,500 on luxury watches at 3 AM = outlier

CFG

Carl Friedrich Gauss

1777–1855

His hair literally matches the bell curve. The man *was* the normal distribution.

Sigmoid: The Probability Squisher

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Plain English: Take any number — huge, tiny, negative — and squish it into a probability between 0 and 1.

$$\sigma(-10) \approx 0.00005 \qquad \text{(very unlikely)}$$
$$\sigma(0) = 0.5 \qquad \text{(coin flip)}$$
$$\sigma(10) \approx 0.99995 \qquad \text{(almost certain)}$$

Sigmoid: The Probability Squisher

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Plain English: Take any number — huge, tiny, negative — and squish it into a probability between 0 and 1.

$\sigma(-10) \approx 0.00005$ (very unlikely)
$\sigma(0) = 0.5$ (coin flip)
$\sigma(10) \approx 0.99995$ (almost certain)

[BOT] *This transaction is **73%** suspicious. I am... sweating on the boundary.*

When the sigmoid says 73%, the AI is uncertain.
That's actually *useful* — it means "check this one."

Where the AI Draws the Line

The sigmoid gives a probability. But someone has to choose a **threshold**:

Below 50%	$\rightarrow$	Approve automatically
50%–80%	$\rightarrow$	Flag for review
Above 80%	$\rightarrow$	Block transaction

Where the AI Draws the Line

The sigmoid gives a probability. But someone has to choose a **threshold**:

Below 50%	$\rightarrow$	Approve automatically
50%–80%	$\rightarrow$	Flag for review
Above 80%	$\rightarrow$	Block transaction

Threshold too low:
Catches more fraud, but annoys many innocent customers

Threshold too high:
Fewer false alarms, but lets real fraud slip through

Abraham Wald: Looking Where Others Don’t

HISTORICAL VIGNETTE

1943

AW

Abraham Wald

1902–1950

WWII problem: Where should you add armor to bombers?

The military studied bullet holes in returning planes and wanted to armor the most-hit areas. Wald said: “Armor the places with NO holes.”

The planes with holes in those areas *never came back*.

Survivorship bias — the same mistake happens when we only study AI decisions that worked, ignoring the failures.

Frank Rosenblatt: The First Neural Network

HISTORICAL VIGNETTE

1958

FR

Frank Rosenblatt

1928–1971

Built the Mark I Perceptron at Cornell — a physical machine that could learn to recognize shapes.

The New York Times (1958): "Navy reveals embryo of an electronic brain that it expects will walk, talk, see, write, and reproduce itself."

Minsky & Papert proved it couldn't learn XOR (1969). AI winter followed. But the **idea** — machines that learn from data — survived and became deep learning.

How Does the AI Learn?

The video game analogy:

1. Guess an answer
2. Check how wrong you are
3. Adjust in the right direction
4. **Repeat** millions of times

How Does the AI Learn?

The video game analogy:

1. Guess an answer
2. Check how wrong you are
3. Adjust in the right direction
4. **Repeat** millions of times

That's it. Every AI — from fraud detection to ChatGPT — learns by **guess, check, adjust, repeat**.

How Does the AI Learn?

The video game analogy:

1. Guess an answer
2. Check how wrong you are
3. Adjust in the right direction
4. **Repeat** millions of times

That's it. Every AI — from fraud detection to ChatGPT — learns by **guess, check, adjust, repeat**.

AL

Ada Lovelace

1815–1852

Wrote the first computer program (1843).
"The Analytical Engine weaves algebraical patterns, just as the Jacquard loom weaves flowers."

She saw it first: math = computation.

Spot the Fraud!

[BOT] **Audience Vote**

Look at each transaction. Raise your hand if you think it's fraud.

Alex, 17, Berlin
Buys a €899 electric guitar online at 10 PM
Never bought musical instruments before
Card used from home IP address

Spot the Fraud!

[BOT] **Audience Vote**

Look at each transaction. Raise your hand if you think it's fraud.

Alex, 17, Berlin
Buys a €899 electric guitar online at 10 PM
Never bought musical instruments before
Card used from home IP address

Verdict: Legitimate. Unusual purchase, but from home IP, reasonable time, single item. Pattern: teen saves up for a big purchase.

More Cases

Tomoko, 45, Tokyo
50 identical €50 gift cards
Purchased at 3:47 AM
Card registered to Osaka address

Karla, 30, São Paulo
€200 in Paris, €180 in Rome,
€150 in Berlin, €90 in Prague
All within 72 hours

More Cases

Tomoko, 45, Tokyo
50 identical €50 gift cards
Purchased at 3:47 AM
Card registered to Osaka address

FRAUD Bulk gift cards at odd hours = classic laundering pattern.

Karla, 30, São Paulo
€200 in Paris, €180 in Rome,
€150 in Berlin, €90 in Prague
All within 72 hours

More Cases

Tomoko, 45, Tokyo
50 identical €50 gift cards
Purchased at 3:47 AM
Card registered to Osaka address

FRAUD Bulk gift cards at odd hours = classic laundering pattern.

Karla, 30, São Paulo
€200 in Paris, €180 in Rome,
€150 in Berlin, €90 in Prague
All within 72 hours

LEGIT Eurotrip! Consistent small amounts, tourist cities, short timeframe.

The Hard One: Priya

Priya, 28, Mumbai
€3,200 wire transfer to a new account
Account opened 3 days ago in London
Priya has never sent money to the UK before

The Hard One: Priya

Priya, 28, Mumbai
€3,200 wire transfer to a new account
Account opened 3 days ago in London
Priya has never sent money to the UK before

Could be fraud:
New account, large amount, unfamiliar destination

Could be legit:
Tuition deposit, family support, business payment

The Hard One: Priya

Priya, 28, Mumbai
€3,200 wire transfer to a new account
Account opened 3 days ago in London
Priya has never sent money to the UK before

Could be fraud:
New account, large amount, unfamiliar destination

Could be legit:
Tuition deposit, family support, business payment

The lesson: Not every case has a clear answer. The AI's job is *harder* than it looks. This is why humans stay in the loop.

ACT 4

Predictions & Decisions

From Data to Action

Your Financial Report Card

300–850

FICO Credit Score Range

Your Financial Report Card

300–850

FICO CREDIT SCORE RANGE

This number determines:

- Whether you get a loan
- Your interest rate (can differ by thousands of euros)
- Whether you can rent an apartment
- Sometimes even whether you get a job

Your Financial Report Card

300–850

FICO Credit Score Range

This number determines:

- Whether you get a loan
- Your interest rate (can differ by thousands of euros)
- Whether you can rent an apartment
- Sometimes even whether you get a job

The math behind it? A weighted average.
The same formula you use to calculate your school grade.

Weighted Averages: Not All Grades Are Equal

$$\text{Score} = w_1 x_1 + w_2 x_2 + w_3 x_3 + \cdots + w_n x_n$$

Weighted Averages: Not All Grades Are Equal

$$\text{Score} = w_1x_1 + w_2x_2 + w_3x_3 + \cdots + w_nx_n$$

School version:
Final Exam: 40%
Homework: 30%
Participation: 20%
Quiz: 10%

Weighted Averages: Not All Grades Are Equal

$$\text{Score} = w_1x_1 + w_2x_2 + w_3x_3 + \cdots + w_nx_n$$

School version:
Final Exam: 40%
Homework: 30%
Participation: 20%
Quiz: 10%

Credit score version:
Payment History: 35%
Amounts Owed: 30%
Length of History: 15%
New Credit: 10%
Credit Mix: 10%

The Weight Slider: Same Data, Different Scores

Two people with the same raw data can get different scores depending on which factors get more weight.

Responsible Student

Payment History: A+
Credit Length: Short (age 22)
Credit Mix: Low (1 card)

710

Good Score

Experienced but Imperfect

Payment History: B- (2 late)
Credit Length: 15 years
Credit Mix: High (mortgage+cards)

695

Lower Score

The Weight Slider: Same Data, Different Scores

Two people with the same raw data can get different scores depending on which factors get more weight.

Responsible Student

Payment History: A+
Credit Length: Short (age 22)
Credit Mix: Low (1 card)

710

Good Score

Experienced but Imperfect

Payment History: B- (2 late)
Credit Length: 15 years
Credit Mix: High (mortgage+cards)

695

Lower Score

Linear Regression: The Simplest Prediction

$$y = mx + b$$

Plain English:
Draw the best straight line through your data.
Use it to predict the future.

m = slope (how fast things change)
b = starting point

Linear Regression: The Simplest Prediction

$$y = mx + b$$

Plain English:
Draw the best straight line through your data.
Use it to predict the future.

m = slope (how fast things change)
b = starting point

Example:

If every year of credit history adds 12 points to your score:

$$\text{Score} = 12 \times \text{years} + 580$$

After 5 years: $12 \times 5 + 580 = 640$
After 10 years: $12 \times 10 + 580 = 700$
After 20 years: $12 \times 20 + 580 = 820$

Reality is more complex (multiple variables), but $y = mx + b$ is where it starts.

Same Math as TikTok and Spotify

> *“If you liked this, you’ll love that.”*

How does Netflix know you’ll like a show you’ve never seen?

It finds people with **similar taste** and recommends what *they* watched. The math: dot product and cosine similarity.

Same Math as TikTok and Spotify

> *"If you liked this, you'll love that."*

How does Netflix know you'll like a show you've never seen?

It finds people with **similar taste** and recommends what *they* watched. The math: dot product and cosine similarity.

[BOT] *Based on my analysis, you need 47 houseplants. My recommendation engine may need calibration.*

Dot Product: Multiplying Matching Preferences

$$\vec{a} \cdot \vec{b} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$

Dot Product: Multiplying Matching Preferences

$$\vec{a} \cdot \vec{b} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$

Alice & Bob rate 4 movies (1–5):

Alice: $[5,\ 1,\ 4,\ 2]$
Bob: $[4,\ 2,\ 5,\ 1]$

$$\vec{A} \cdot \vec{B} = 5 \times 4 + 1 \times 2 + 4 \times 5 + 2 \times 1$$
$$= 20 + 2 + 20 + 2 = 44$$

High number $\rightarrow$ similar taste!

Dot Product: Multiplying Matching Preferences

$$\vec{a} \cdot \vec{b} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$

Alice & Bob rate 4 movies (1–5):

Alice: $[5,\ 1,\ 4,\ 2]$
Bob: $[4,\ 2,\ 5,\ 1]$

$$\vec{A} \cdot \vec{B} = 5 \times 4 + 1 \times 2 + 4 \times 5 + 2 \times 1$$
$$= 20 + 2 + 20 + 2 = 44$$

High number $\rightarrow$ similar taste!

Why it works: when both rate a movie high, the product is large. When they disagree, the product is small.

Matching enthusiasm drives the score up.

Cosine Similarity: How Similar Are You?

$$\cos(\theta) = \frac{\vec{a} \cdot \vec{b}}{||\vec{a}|| \cdot ||\vec{b}||}$$

Cosine Similarity: How Similar Are You?

$$\cos(\theta) = \frac{\vec{a} \cdot \vec{b}}{||\vec{a}|| \cdot ||\vec{b}||}$$

$$||\vec{A}|| = \sqrt{25 + 1 + 16 + 4} = \sqrt{46} \approx 6.78$$
$$||\vec{B}|| = \sqrt{16 + 4 + 25 + 1} = \sqrt{46} \approx 6.78$$

$$\cos(\theta) = \frac{44}{6.78 \times 6.78} = \frac{44}{46} \approx \mathbf{0.96}$$

Cosine Similarity: How Similar Are You?

$$\cos(\theta) = \frac{\vec{a} \cdot \vec{b}}{||\vec{a}|| \cdot ||\vec{b}||}$$

$$||\vec{A}|| = \sqrt{25 + 1 + 16 + 4} = \sqrt{46} \approx 6.78$$
$$||\vec{B}|| = \sqrt{16 + 4 + 25 + 1} = \sqrt{46} \approx 6.78$$

$$\cos(\theta) = \frac{44}{6.78 \times 6.78} = \frac{44}{46} \approx \mathbf{0.96}$$

96% similar!

1.0 = identical taste

0.0 = completely different

−1.0 = opposite taste

Alice and Bob should share playlists.

Same formula used in: word embeddings (King − Man + Woman = Queen), Chat-GPT's attention mechanism, and Spotify

The AI Pipeline: All Four Formulas Together

Bell Curve →	**Bayes** →	**Sigmoid** →	**Decision**
Is this normal?	Update belief	Get probability	Approve/Flag/Block

The AI Pipeline: All Four Formulas Together

Bell Curve	→	Bayes	→	Sigmoid	→	Decision
Is this normal?		Update belief		Get probability		Approve/Flag/Block

Each formula does one job. Together they form a **pipeline** that processes every transaction in $\sim$50 milliseconds. The same pipeline powers credit decisions, insurance pricing, and loan approvals.

USE IT or SKIP IT?

[BOT] **Ethics Exercise**

For each data point, vote: should a bank be **allowed** to use it for credit decisions?

[BOT] *I could use social media data. Doesn't mean I should.*

USE IT or SKIP IT?

[BOT] **Ethics Exercise**

For each data point, vote: should a bank be **allowed** to use it for credit decisions?

USE IT

Income level

Directly relevant to ability to repay

SKIP IT

Social media posts

Privacy violation, unreliable signal

[BOT] *I could use social media data. Doesn't mean I should.*

The Controversial Ones

DEBATE

Zip/postal code

Correlates with income *and* with race. Using it may be efficient but discriminatory.

This is a proxy variable.

The Controversial Ones

DEBATE

Zip/postal code

Correlates with income *and* with race. Using it may be efficient but discriminatory.

This is a proxy variable.

DISCRIMINATORY

Smartphone brand

iPhone users default less often — but using this penalizes people who can't afford one.

Punishing poverty is not credit scoring.

The Controversial Ones

DEBATE

Zip/postal code

Correlates with income *and* with race. Using it may be efficient but discriminatory.

This is a proxy variable.

DISCRIMINATORY

Smartphone brand

iPhone users default less often — but using this penalizes people who can't afford one.

Punishing poverty is not credit scoring.

The fairness impossibility: You can't simultaneously maximize accuracy *and* guarantee equal treatment across all groups. Every model involves a tradeoff.

The Hardest Part

> *“The hard part isn’t the math.*
> *It’s deciding what should go into the math.”*
>
> — BankBot, Stage 7

ACT 5

The Bigger Picture

Power, Fairness, and Your Future

2017: One Paper Changed Everything

"Attention Is All You Need"

Vaswani et al., Google Brain, 2017

8 authors. One was a 20-year-old intern.
Now the most-cited AI paper in history.

The core idea: Let the model decide which words to "pay attention to" when processing each word.

2017: One Paper Changed Everything

"Attention Is All You Need"

Vaswani et al., Google Brain, 2017

8 authors. One was a 20-year-old intern.

Now the most-cited AI paper in history.

The core idea: Let the model decide which words to "pay attention to" when processing each word.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V$$

Q = what am I looking for?

K = what do I contain?

V = what do I actually say?

This is **cosine similarity** at scale — the same formula from the movie ratings!

King – Man + Woman = ?

$$\vec{\text{King}} - \vec{\text{Man}} + \vec{\text{Woman}} \approx \vec{\text{Queen}}$$

King – Man + Woman = ?

$$\vec{\text{King}} - \vec{\text{Man}} + \vec{\text{Woman}} \approx \vec{\text{Queen}}$$

Mikolov et al., Google, 2013 — Word2Vec

Words become vectors (lists of numbers). The AI learns that "king" and "queen" differ in the same way as "man" and "woman."

The math: Vector subtraction and addition — the same operations as in your linear algebra class.

King – Man + Woman = ?

$$\vec{\text{King}} - \vec{\text{Man}} + \vec{\text{Woman}} \approx \vec{\text{Queen}}$$

Mikolov et al., Google, 2013 — Word2Vec

Words become vectors (lists of numbers). The AI learns that “king” and “queen” differ in the same way as “man” and “woman.”

The math: Vector subtraction and addition — the same operations as in your linear algebra class.

Other discoveries: Paris – France + Japan ≈ Tokyo.

Walk – Walking + Swim ≈ Swimming

DeepSeek: The $6 Million Challenge

$5.6M

Compute Cost

vs.

$100M+

Total Dev Budget

DeepSeek: The $6 Million Challenge

$5.6M

COMPUTE COST

vs.

$100M+

TOTAL DEV BUDGET

January 2025: Chinese startup DeepSeek released R1, matching GPT-4 on benchmarks. Training compute cost: $5.6M vs. GPT-4's estimated $100M+ total development budget.

Result: Nvidia lost $589 billion in market value in one day. Proved that mathematical efficiency beats brute-force spending.

The secret: Mixture of Experts (MoE) — only activate the relevant "experts" for each question. Same idea as asking different teachers for different subjects.

How Many R's in "Strawberry"?

Q: How many R's in "strawberry"?
ChatGPT (2024): "Two" WRONG (it's three)

How Many R’s in “Strawberry”?

Q: How many R’s in “strawberry”?
ChatGPT (2024): “Two” WRONG (it’s three)

Why? ChatGPT doesn’t see letters. It sees **tokens**:

“strawberry” $\rightarrow$ [“str”, “aw”, “berry”]

The letter R is split across tokens. The AI literally *cannot see* individual characters in most cases.

This is BPE (Byte Pair Encoding) — the tokenizer trades letter-level accuracy for efficiency at the sentence level.

How Many R’s in “Strawberry”?

Q: How many R’s in “strawberry”?
ChatGPT (2024): “Two” WRONG (it’s three)

Why? ChatGPT doesn’t see letters. It sees **tokens**:

“strawberry” $\rightarrow$ [“str”, “aw”, “berry”]

The letter R is split across tokens. The AI literally *cannot see* individual characters in most cases.

This is BPE (Byte Pair Encoding) — the tokenizer trades letter-level accuracy for efficiency at the sentence level.

The paradox: The same system that scores gold at the Math Olympiad cannot count

The Lawyer Who Cited Fake Cases

Avianca v. Mata (2023)

Lawyer Steven Schwartz used ChatGPT to research case law. It generated 6 fictitious court cases with realistic citations.

He submitted them to a federal judge.
None of them existed.

The Lawyer Who Cited Fake Cases

Avianca v. Mata (2023)

Lawyer Steven Schwartz used ChatGPT to research case law. It generated 6 fictitious court cases with realistic citations.

He submitted them to a federal judge.
None of them existed.

$$P(\text{all correct}) = p^n$$

If each "fact" is 95% plausible:

$$\begin{aligned} 0.95^{10} &= 0.60 \\ 0.95^{50} &= 0.08 \\ 0.95^{100} &= 0.006 \end{aligned}$$

Small errors compound. A 5% error rate per fact means the probability of a *perfect* long document is nearly zero.

Apple Card: When AI Discriminates

November 2019: Apple Card gave a husband 20$\times$ the credit limit of his wife — despite her having a higher credit score.

March 2024: Goldman Sachs fined \$89 million for discrimination.

Apple Card: When AI Discriminates

November 2019: Apple Card gave a husband $20\times$ the credit limit of his wife — despite her having a higher credit score.

March 2024: Goldman Sachs fined $89 million for discrimination.

What happened: The algorithm never used “gender” as an input. But it used proxy variables — spending patterns, employment type, income sources — that correlate with gender.

The lesson: Removing a protected variable doesn’t make an algorithm fair. Other variables can “leak” the same information.

This is exactly the zip code problem from USE IT/SKIP IT.

Apple Card: When AI Discriminates

November 2019: Apple Card gave a husband 20$\times$ the credit limit of his wife — despite her having a higher credit score.

March 2024: Goldman Sachs fined $89 million for discrimination.

What happened: The algorithm never used "gender" as an input. But it used proxy variables — spending patterns, employment type, income sources — that correlate with gender.

The lesson: Removing a protected variable doesn't make an algorithm fair. Other variables can "leak" the same information.

This is exactly the zip code problem from USE IT/SKIP IT.

JPMorgan: AI at Industrial Scale

200K

Employees

$17B

Annual Tech

Budget

2,000+

AI Models

in Production

JPMorgan: AI at Industrial Scale

200K
EMPLOYEES

$17B
ANNUAL TECH BUDGET

2,000+
AI MODELS IN PRODUCTION

Every formula in this talk runs inside JPMorgan right now:

Bayes $\rightarrow$ fraud detection
Weighted average $\rightarrow$ credit scoring
Cosine similarity $\rightarrow$ document analysis (contracts, regulations)
Sigmoid $\rightarrow$ risk classification

They use **RAG** (Retrieval-Augmented Generation) — the AI searches a database before answering, reducing hallucinations.

"Let's Think Step by Step"

2022: Researchers at Google Brain discovered that adding one sentence to a prompt — "Let's think step by step" — dramatically improved accuracy.

2024: OpenAI's o1 model built this in, scoring 93% on AIME (math competition for the top 0.5% of US students).

“Let’s Think Step by Step”

2022: Researchers at Google Brain discovered that adding one sentence to a prompt — “Let’s think step by step” — dramatically improved accuracy.

2024: OpenAI’s o1 model built this in, scoring 93% on AIME (math competition for the top 0.5% of US students).

The school connection:

Your math teacher says “show your work.”

That’s chain-of-thought prompting.

When the AI explains its reasoning, it reasons *better* — just like you.

AI Will Always Sometimes Be Wrong

Xu et al., 2024 (arXiv:2409.05746): Proved mathematically that hallucinations are inevitable in current AI architectures.

No amount of training data can fully eliminate them.

AI Will Always Sometimes Be Wrong

Xu et al., 2024 (arXiv:2409.05746): Proved mathematically that hallucinations are inevitable in current AI architectures.

No amount of training data can fully eliminate them.

Why? An LLM predicts the most *probable* next word, not the most *true* next word. Probability and truth are different things.

In finance: This is why AI-generated financial advice must always have human review. The math is powerful but imperfect.

AI Will Always Sometimes Be Wrong

Xu et al., 2024 (arXiv:2409.05746): Proved mathematically that hallucinations are inevitable in current AI architectures.

No amount of training data can fully eliminate them.

Why? An LLM predicts the most *probable* next word, not the most *true* next word. Probability and truth are different things.

In finance: This is why AI-generated financial advice must always have human review. The math is powerful but imperfect.

> *"These limits are mathematical, not just engineering problems. Understanding the math tells you both what AI can do and what it cannot."*

Teaching AI Human Values

RLHF: Reinforcement Learning from Human Feedback

Step 1: AI generates multiple answers
Step 2: Humans rank which is best
Step 3: AI adjusts to match human preferences
Step 4: Repeat thousands of times

Teaching AI Human Values

RLHF: Reinforcement Learning from Human Feedback

Step 1: AI generates multiple answers
Step 2: Humans rank which is best
Step 3: AI adjusts to match human preferences
Step 4: Repeat thousands of times

Goodhart's Law: "When a measure becomes a target, it ceases to be a good measure."

If you reward helpfulness, the AI becomes

Teaching AI Human Values

RLHF: Reinforcement Learning from Human Feedback

Step 1: AI generates multiple answers
Step 2: Humans rank which is best
Step 3: AI adjusts to match human preferences
Step 4: Repeat thousands of times

Goodhart's Law: "When a measure becomes a target, it ceases to be a good measure."

If you reward helpfulness, the AI becomes

The idea:
Maximize helpfulness while staying close to the original model.

Too helpful $\rightarrow$ lies to please
Too cautious $\rightarrow$ refuses everything

The sweet spot is a math problem.

Fair and Useful Aren’t Always the Same

> *“The same math that catches fraud can also discriminate.*
> *The same model that saves money can also entrench inequality.*
> *The question is never just “does it work?” but “who does it work for?””*
>
> — The Central Tension

Fair and Useful Aren't Always the Same

> *"The same math that catches fraud can also discriminate.*
> *The same model that saves money can also entrench inequality.*
> *The question is never just "does it work?" but "who does it work for?""*
>
> — The Central Tension

Three principles for ethical AI in finance:

1. Transparency — explain how decisions are made
2. Accountability — humans stay in the loop
3. Auditability — test for bias regularly

ACT 6

Your Future

Math Is Your Superpower

8 Careers at the Intersection

Math + AI + Finance = these careers

P1 **Data Scientist**
Pattern finding in financial data

P2 **Quantitative Analyst**
Math models for trading strategies

P3 **AI Ethics Researcher**
Ensuring fairness in algorithms

P4 **Cybersecurity Analyst**

P5 **FinTech Founder**
Building financial apps with AI

P6 **Risk Manager**
Predicting what could go wrong

P7 **Regulatory Technologist**
AI compliance and auditing

P8 **ML Engineer**

Math Cheat Sheet

Bayes' Theorem

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Update beliefs with evidence

Sigmoid Function

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Squish to probability 0–1

Weighted Average

$$\bar{x}_w = \sum w_i x_i$$

Not all factors are equal

Linear Regression

Dot Product

$$\vec{a} \cdot \vec{b} = \sum a_i b_i$$

Multiply matching preferences

Cosine Similarity

$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}|\,|\vec{b}|}$$

How similar are two things?

Normal Distribution

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

The bell curve: what "normal" is

Keep Exploring

Watch

3Blue1Brown — "LLMs explained briefly" ($\sim$5 min)

youtube.com/3blue1brown

See in 3D

Bycroft LLM — zoom into every matrix operation

bbycroft.net/llm

Explore

Transformer Explainer — type text, see tokens $\rightarrow$ embeddings $\rightarrow$ attention $\rightarrow$ prediction

poloclub.github.io/transformer-explainer

This Talk

Full website with interactive exercises

digital-ai-finance.github.io/mathematics-for-ai

BankBot's Journey

[BOT] *I started this talk thinking I knew everything. I was wrong about flowers. I was uncertain about Priya. I recommended 47 houseplants.*

BankBot’s Journey

[BOT] *I started this talk thinking I knew everything. I was wrong about flowers. I was uncertain about Priya. I recommended 47 houseplants.*

[BOT] *But I learned. I learned that 73% confident means “ask a human.” I learned that fair and accurate aren’t the same thing. I learned that showing your work makes you smarter.*

BankBot's Journey

[BOT] *I started this talk thinking I knew everything. I was wrong about flowers. I was uncertain about Priya. I recommended 47 houseplants.*

[BOT] *But I learned. I learned that 73% confident means "ask a human." I learned that fair and accurate aren't the same thing. I learned that showing your work makes you smarter.*

**"I am 73% confident.
But I defer to the human."**

What Will YOU Build?

> *“Every breakthrough in AI was built by someone who understood the math — and dared to ask “what if?””*

What Will YOU Build?

"Every breakthrough in AI was built by someone who understood the math — and dared to ask "what if?""

Nightingale used data to save soldiers. Lovelace saw that math is computation. Wald looked where no one else looked. Rosenblatt built a machine that learns.

Math has always needed diverse thinkers.
The next breakthrough might be yours.

Thank You!

Questions?

Website: digital-ai-finance.github.io/mathematics-for-ai

Slides: digital-ai-finance.github.io/mathematics-for-ai/slides.html