AI 工具与效率革命——生成式 AI 实战应用

From Machine Learning Foundations to Agentic WealthTech

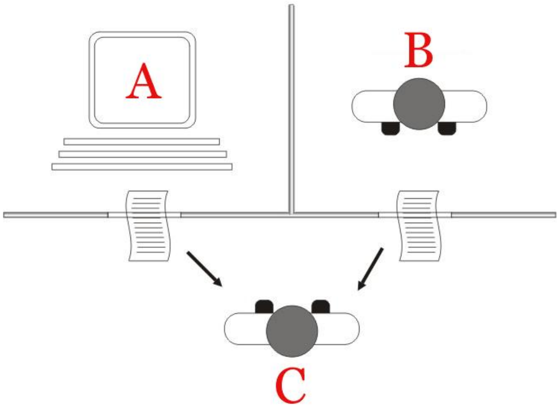
Don HUANG Yao Tung, PhD
Chief Scientist and Co-founder at AQUMON

The Original Question: Can Machines Think?

- Alan Turing reframed a philosophical question into an operational test.
- Instead of asking what thinking "really is," he asked whether a machine can behave intelligently in conversation.
- Today, AI is broader than imitation. We care about systems that can perceive, predict, reason, plan, act and learn from feedback.

Core intuition: AI is not one technology. It is a family of methods for turning information into useful behavior.

Visual Example: The Turing Test



Source: Juan Alberto Sánchez Margallo, Wikimedia Commons, CC BY 2.5.

Lecture point: Turing reframed intelligence as observable behavior in a controlled interaction.

A Practical Definition of AI

- **Artificial intelligence** refers to computational systems that perform tasks normally associated with human cognition.
- Examples: perception, language understanding, pattern recognition, decision-making and problem solving.
- In business, a practical definition is simpler: AI helps convert data and context into better decisions or actions.

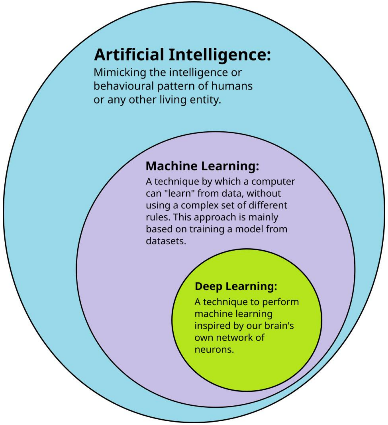
Takeaway: The central question is not "Is it intelligent?" but "Which decision process does it improve?"

AI, Machine Learning, Deep Learning and Foundation Models

Plain-language view

- AI: the big umbrella.
- Machine learning: learn patterns from data.
- Deep learning: neural networks with many layers.
- Foundation models: large general models adapted to many tasks.

Visual Example: AI, ML and Deep Learning Relationship



Source: AI-ML-DL.svg, Wikimedia Commons, CC BY-SA 4.0.

Lecture point: This hierarchy helps beginners avoid mixing up the umbrella field, learning methods and deep neural networks.

Five Abilities of Modern AI

Ability	Simple meaning	Finance example
Perceive	read inputs	parse PDFs and tables
Predict	estimate future/label	default probability
Generate	create content	draft client report
Reason	connect steps	explain portfolio risk
Act	use tools/workflows	rebalance proposal

Core intuition: Modern AI is powerful because these abilities can be combined in one workflow.

Machine Learning as Learning from Examples

Core intuition: Machine learning is like teaching by examples instead of writing every rule by hand.

- Traditional programming: human writes rules, computer follows rules.
- Machine learning: human provides examples and objectives, computer learns patterns.
- Example: instead of manually listing every fraud pattern, we train on historical transactions.

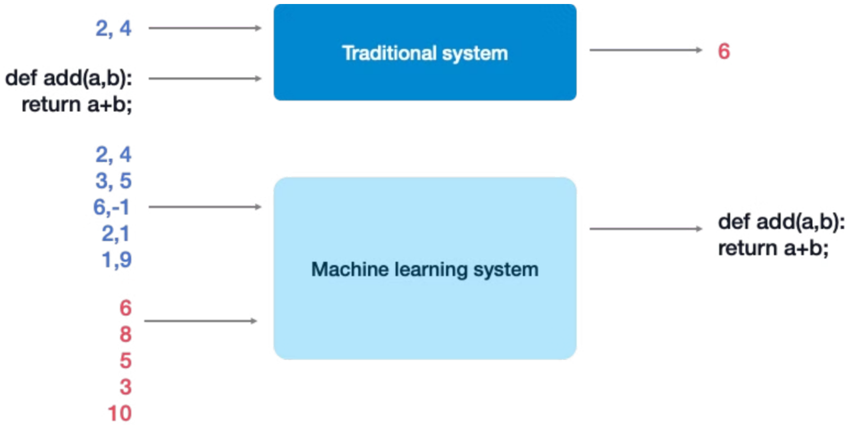
Finance lens: This is useful in finance because many patterns are too complex or too fast-changing for manual rules alone.

Machine Learning as Function Approximation

- A compact mathematical view:
$$f_\theta(x) \approx y.$$
- Input x: image, text, customer profile, transaction, market features or portfolio state.
- Output y: label, forecast, recommendation, action or generated content.
- Learning means choosing parameters θ so that f_θ works well on unseen data.

Technical lens: Most practical ML problems are about selecting the right inputs, objective function, validation design and deployment controls.

Traditional System vs ML



Data, Objective, Optimization, Validation

- A model is not trained from data alone.
- Four ingredients matter:
  1. **Data**: what the model can learn from.
  2. **Objective**: what the model is rewarded for doing.
  3. **Optimization**: how the model parameters are fitted.
  4. **Validation**: how we know it works outside training data.

$$\text{Data} + \text{Objective} + \text{Optimization} + \text{Validation} \Rightarrow \text{Useful model.}$$

Three Main Learning Paradigms

Paradigm	Everyday analogy	Typical output	Finance example
Supervised	answer key	prediction	default probability
Unsupervised	grouping	clusters/factors	client segmentation
Reinforcement	trial and feedback	policy	execution strategy

Core intuition: The paradigm depends on what kind of feedback the system receives: labels, structure, or rewards.

Supervised Learning: Learning with an Answer Key

- Training data contains both inputs and known answers:
$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}.$$

- Regression predicts a number: return, volatility, loss, revenue.
- Classification predicts a category: fraud/not fraud, churn/not churn, approve/reject.
- The model is useful only if it generalizes to new examples.

Finance lens: Credit scoring, fraud detection and customer churn prediction are classic supervised-learning problems.

Supervised Learning Examples

Task	Input x	Target y	Output type
Fraud detection	transaction amount, merchant, device, location	fraud label	probability
Credit risk	borrower profile, income, repayment history	default event	probability
Equity factor model	valuation, quality, momentum features	future return rank	score
Client churn	activity, holdings, service history	churn/not churn	class

Takeaway: The same learning framework appears in very different financial products.

Loss Function and Empirical Risk

- A model is trained by minimizing average loss on training data:

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} L(f_\theta(x_i), y_i) + \lambda\Omega(\theta).$$

- L measures prediction error.
- Ω regularizes model complexity.
- λ controls the trade-off between fit and simplicity.

Technical lens: Changing the loss function changes what the model optimizes. In finance, optimizing the wrong proxy can create the wrong behavior.

Overfitting: The Most Important Warning

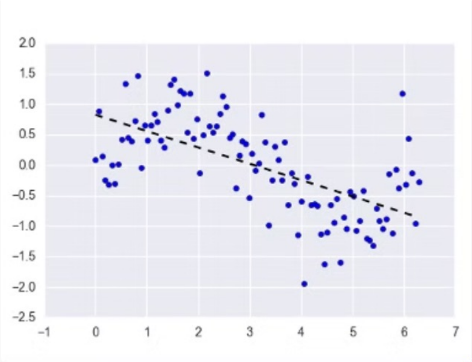
Core intuition: Overfitting means the model memorizes noise instead of learning a repeatable pattern.

- A model can look excellent on historical data and fail in the future.
- Finance is especially vulnerable because market signals are weak and noisy.
- More complexity is not always better; sometimes it simply gives noise a better suit.

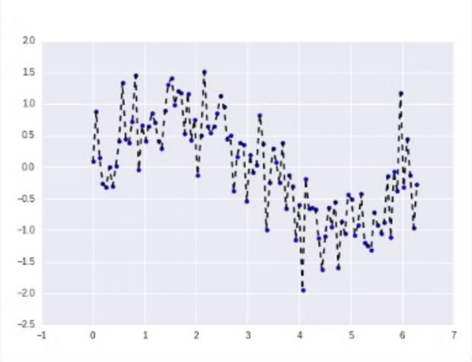
$$\mathbb{E}[(\hat{f}(x) - y)^2] = \text{Bias}^2 + \text{Variance} + \text{Noise}.$$

Takeaway: The real exam is not training performance; it is out-of-sample performance.

Visual Example: Underfitting vs. Overfitting



High bias, low variance



Low bias, high variance

Source: scikit-learn documentation, Underfitting vs. Overfitting example, BSD-3-Clause.

Lecture point: The best model is not the most complex one; it is the one that generalizes.

Train, Validation and Test Sets

- **Training set**: fit the model.
- **Validation set**: tune hyperparameters and select models.
- **Test set**: estimate final out-of-sample performance.
- In time-series finance, random splitting can leak future information.
- Use chronological split, walk-forward validation or purged cross-validation.

Demo: Machine Learning

https://playground.tensorflow.org/

Unsupervised Learning: Finding Structure Without Labels

Core intuition: Unsupervised learning is like organizing a messy library without a catalog. The system looks for natural structure.

- Data has no labels:

$$\mathcal{D} = \{x_i\}_{i=1}^{n}.$$

- Common methods: K-means, hierarchical clustering, PCA and autoencoders.
- Finance examples: market regimes, client segmentation, peer grouping and style factor extraction.

K-Means Clustering

- K-means partitions observations into K clusters.
- It minimizes within-cluster squared distance:

$$\min_{C_1,\dots,C_K} \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - \mu_k\|^2.$$

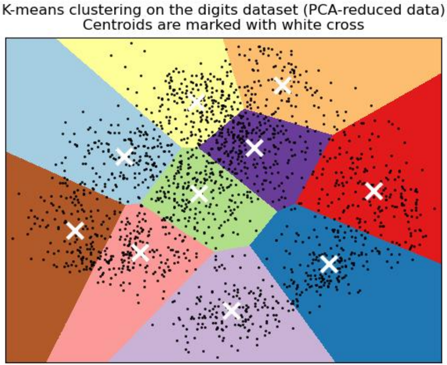
- Plain example: group customers by investment behavior.
- Finance example: cluster stocks by return correlation, valuation profile or risk behavior.

Technical lens: Clusters are descriptions, not truth. They can change when regimes change or when the feature design changes.

Demo: Machine Learning

https://playground.tensorflow.org/

Visual Example: K-Means on Handwritten Digits



Source: scikit-learn documentation, K-Means clustering on digits, BSD-3-Clause.

Lecture point: Clustering creates useful structure, but the structure depends on the representation and distance

Reinforcement Learning: Learning by Feedback

Core intuition: Reinforcement learning is learning by trying actions and receiving rewards. It is closer to training a player than fitting a spreadsheet formula.

- An agent observes state s_t, takes action a_t and receives reward r_t.
- Objective: learn a policy $\pi(a|s)$ that maximizes cumulative reward.

$$\max_{\pi} \ \mathbb{E}_{\pi}\left[\sum_{t=0}^{T} \gamma^t r_t\right].$$

Finance lens: Finance examples include order execution, market making, dynamic allocation and hedging.

Why Reinforcement Learning Is Hard in Finance

- Feedback is noisy: good decisions can have bad short-term outcomes.
- Markets are non-stationary: yesterday's policy may fail tomorrow.
- Exploration is expensive: real trading mistakes cost money.
- The agent may affect the environment through its own trades.

Takeaway: RL is conceptually attractive in finance, but deployment needs simulation, constraints and human oversight.

Demo: Reinforcement Learning

Youtube: https://www.youtube.com/watch?v=Lt-KLtkDlh8

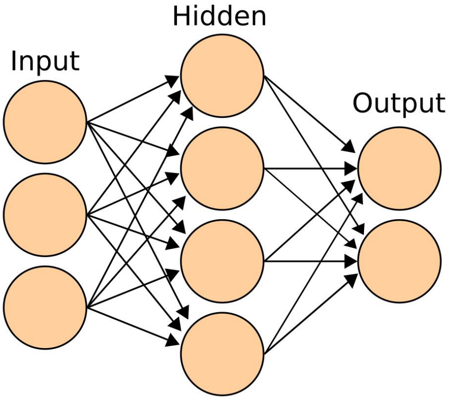
Neural Networks

Core intuition: A neural network is a flexible function made by stacking many simple transformations.

- Each layer transforms the previous representation:
$$h^{(l)} = \sigma(W^{(l)} h^{(l-1)} + b^{(l)}).$$
- Early layers learn basic patterns; later layers learn task-specific representations.
- Deep learning became powerful because of data, GPUs, better optimization and open-source tooling.

Finance lens: In finance, deep models can help with text, documents and high-dimensional data, but weak market signals still require careful validation.

Visual Example: A Simple Neural Network Topology



Source: Cburnett, Wikimedia Commons, CC BY-SA 3.0 or GFDL.

Lecture point: A neural network stacks simple units into layers so the whole system can approximate complex

CNN, RNN and LSTM: Before Transformers

- **CNN**: captures local spatial patterns; originally dominant in image tasks.
- **RNN**: processes sequences step by step; useful for language and time series.
- **LSTM**: improves long-range sequence learning through gated memory.
- These architectures prepared the ground for modern sequence models.

Takeaway: Architecture matters because it determines what patterns the model can learn efficiently.

Attention Mechanism: The Key Intuition

Core intuition: Attention lets the model decide which parts of the input are relevant to each other.

- In a sentence, a word may need information from another word far away.
- In a report, a conclusion may depend on numbers in a table several pages earlier.
- Attention creates a dynamic relevance map across tokens.

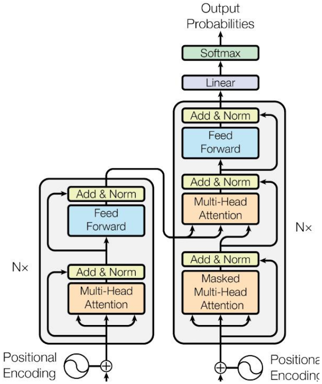
Finance lens: For financial documents, attention helps connect terms, definitions, tables, footnotes and risk disclosures.

Transformers

- Transformers are built from self-attention, feed-forward layers and normalization.
- They support parallel training over long sequences.
- They became the dominant architecture for language and increasingly for vision, audio, code and multimodal reasoning.
- Most modern LLMs are transformer-based or transformer-derived systems.

Suggested visual: A clean transformer block illustration with token embeddings, self-attention and feed-forward layers.

Visual Example: Transformer Architecture



Source: Vaswani et al. (2017), Attention Is All You Need, NeurIPS paper Figure 1.

Lecture point: Transformers combine attention, feed-forward layers, residual connections and positional

Large Language Models

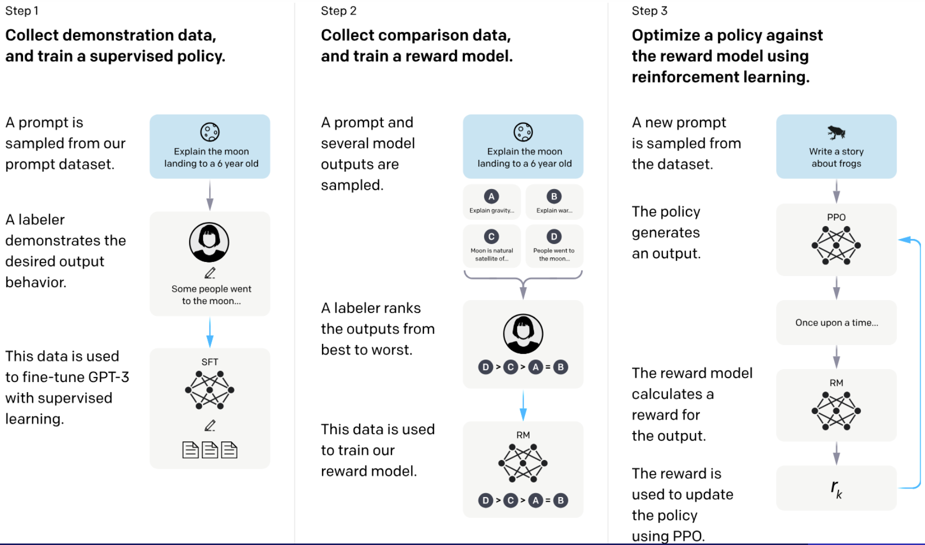
Core intuition: An LLM learns language by predicting the next token at massive scale. Simple training objective; surprisingly rich behavior.

- Pretraining objective:

$$\max_\theta \sum_t \log p_\theta(x_t | x_{<t}).$$

- Pretraining produces broad language and knowledge representations.
- Instruction tuning makes the model more helpful.
- Tool use and retrieval connect the model to external systems and fresh data.

GPT 3.5


Step 1
Collect demonstration data, and train a supervised policy.
A prompt is sampled from our prompt dataset.
Explain the moon landing to a 6 year old
A labeler demonstrates the desired output behavior.
Some people went to the moon...
This data is used to fine-tune GPT-3 with supervised learning.
SFT
Step 2
Collect comparison data, and train a reward model.
A prompt and several model outputs are sampled.
Explain the moon landing to a 6 year old
A
Explain gravity...
B
Explain war...
C
Moon is natural satellite of...
D
People went to the moon...
A labeler ranks the outputs from best to worst.
D > C > A = B
This data is used to train our reward model.
RM
D > C > A = B
Step 3
Optimize a policy against the reward model using reinforcement learning.
A new prompt is sampled from the dataset.
Write a story about frogs
The policy generates an output.
PPO
Once upon a time...
The reward model calculates a reward for the output.
RM
r_k
The reward is used to update the policy using PPO.

Foundation Models: The New AI Layer

- Foundation models are trained broadly and adapted to many downstream tasks.
- They shift development from task-specific modeling to prompt, retrieval, tool and workflow design.
- They can support language, code, image, audio, video and structured data.
- In enterprise finance, the key architecture is often a controlled workflow around a foundation model, not the raw model alone.

Takeaway: Foundation models are general engines; financial value comes from connecting them to trusted data and responsible processes.

Multimodal AI

- Multimodal models process combinations of text, image, audio, video and structured data.
- Finance examples:
  - read PDFs, charts and tables;
  - extract data from factsheets and reports;
  - analyze earnings-call audio and transcripts;
  - support advisor-client conversations with document grounding.

Suggested visual: Different inputs such as PDF, chart, table, voice and news flowing into a unified model, then producing a financial decision-support output.

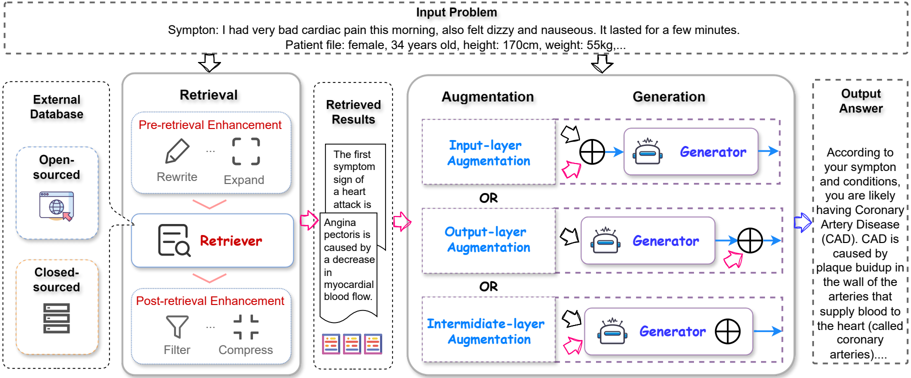
RAG: Retrieval-Augmented Generation

Core intuition: RAG gives the model a trusted library before asking it to answer.

1. Parse and index documents.
2. Retrieve relevant chunks.
3. Generate answer with source references.
4. Evaluate faithfulness and completeness.

Finance lens: RAG is central in finance because many answers must be traceable to policies, reports, contracts, fund documents or regulations.

Visual Example: RAG Architecture



Source: Fan et al. (2024), *A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models*, Figure 3.

RAG: What Can Go Wrong?

- Retrieval failure: the right document was not found.
- Chunking failure: the answer needs context split across chunks.
- Generation failure: the model misreads or over-interprets retrieved text.
- Evaluation failure: the answer sounds plausible but is not fully supported.

Technical lens: A production RAG system needs document lineage, permissions, citation checks, evaluation sets and monitoring.

AI Agents

Core intuition: An agent is not just a model that answers; it can plan, use tools, observe results and revise.

$$\text{Goal} \rightarrow \text{Plan} \rightarrow \text{Tool Use} \rightarrow \text{Observation} \rightarrow \text{Revision}.$$

- Agents are useful when a task requires multi-step execution rather than a single response.
- Examples: research assistant, coding assistant, portfolio diagnostic assistant.
- Finance needs permissioning, logging, sandboxing and approval gates.

World Models

Core intuition: A world model learns how an environment evolves, so an agent can imagine consequences before acting.

- It supports counterfactual reasoning: "What may happen if I take this action?"
- Compared with a text-only LLM, a world model focuses more on state, dynamics, action and simulation.
- It is especially important for agents that need to plan in changing environments.

Suggested visual: An agent trying actions in a simulated environment before taking real-world actions.

World Models: Mathematical View

- State: s_t; action: a_t; observation: o_t; reward: r_t.
- A world model approximates transition and observation structure:

$$\hat{p}(s_{t+1}, o_{t+1}, r_t \mid s_t, a_t).$$

- Planning can then be performed in imagination:

$$\max_{a_{0:T}} \mathbb{E}_{\hat{p}} \left[\sum_{t=0}^{T} \gamma^t r_t \right].$$

World Models in Finance: Analogy and Limits

- Finance has an implicit world-model problem: investors reason about regimes and consequences.
- Potential uses:
  - portfolio stress testing;
  - scenario generation;
  - execution simulation;
  - synthetic market environments for strategy testing.
- Major limitation: financial markets are adaptive systems; the model can change the world it tries to model.

Takeaway: A financial world model is useful only if it is treated as a scenario engine, not a crystal ball.

Evaluation of Modern AI Systems

- Classic ML evaluation: accuracy, AUC, RMSE, Sharpe ratio and cross-validation.
- Generative AI evaluation also needs:
  - factuality and citation faithfulness;
  - reasoning correctness;
  - tool-use success rate;
  - latency and cost;
  - safety and compliance.

Finance lens: For finance, evaluation should be linked to the business decision being supported.

AI Governance in Finance

- Financial AI systems need more than model accuracy.
- Key controls:
  - data lineage and access control;
  - model validation and monitoring;
  - explainability and audit trail;
  - human approval for regulated or high-impact decisions;
  - cybersecurity and privacy protection.

Takeaway: A good AI system should fail safely and leave evidence.

AI Applications in Real World

From automation to investment decision intelligence

Where AI Creates Value

- **Scale**: serve more clients at lower marginal cost.
- **Speed**: automate document reading, monitoring and reporting.
- **Consistency**: reduce manual variation in decisions.
- **Personalization**: adapt products and advice to client constraints.
- **Insight**: extract signals from unstructured and high-dimensional data.

Core intuition: AI matters when it improves a workflow, not when it is merely attached to a product name.

Major AI Use Cases in Financial Services

Area	AI task	Example output
Customer service	summarization and dialogue	advisor copilot
Risk and fraud	anomaly detection	suspicious transaction alert
Credit	supervised prediction	default probability
Compliance	document review	policy breach flag
Research	retrieval and coding	factor test memo
WealthTech	optimization and personalization	portfolio proposal
Trading	sequential decision	execution schedule

Why Finance Is Different

- Financial decisions are high-stakes and regulated.
- Data is noisy, non-stationary and often proprietary.
- Backtests can be misleading without strong controls.
- Model errors can create financial loss, conduct risk or regulatory risk.
- Therefore the AI workflow must combine automation with governance.

Takeaway: In finance, “move fast and break things” is not a strategy. It is an incident report waiting for a date.

From Digital Finance to AI-Native Finance

- **Digital finance**: online forms, dashboards, APIs and rule-based automation.
- **AI-assisted finance**: models support search, summarization, prediction and monitoring.
- **AI-native finance**: agents coordinate data, models, tools and approvals across an end-to-end process.

Core intuition: The next frontier is not just a chatbot. It is workflow intelligence.

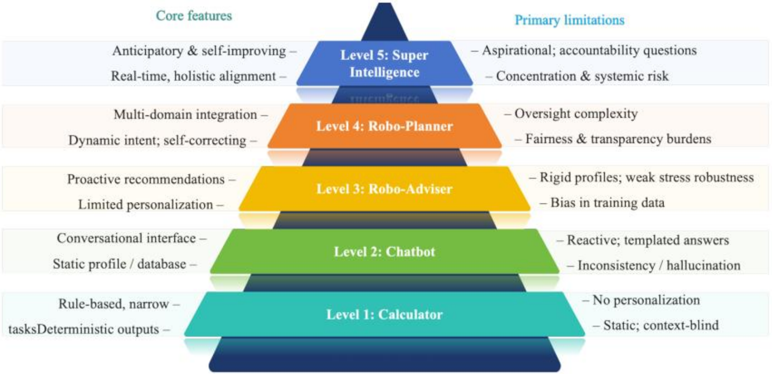
Robo-Advisory: Core Idea

Core intuition: Robo-advisory converts client information into investment recommendations through a structured, repeatable process.

1. Client onboarding and risk profiling.
2. Goal and constraint collection.
3. Strategic asset allocation.
4. Portfolio implementation.
5. Monitoring, rebalancing and reporting.

Finance lens: The value comes from systematic advice, not from pretending human judgment is unnecessary.

Visual Example: AI-Driven Robo-Advisory Roadmap



Source: Feng, Li and Liu (2025), *Robo-Advisors Beyond Automation*, Figure 2.

Risk Profiling

- Risk profiling estimates a client's ability and willingness to take risk.
- Inputs may include age, income, wealth, horizon, liquidity needs, investment knowledge and loss tolerance.
- Common output: a risk score or risk category.
- AI can assist by detecting inconsistent answers, explaining trade-offs and mapping goals to portfolios.

Technical lens: Risk profiling is partly statistical and partly behavioral. The hardest part is not the score; it is suitability and explanation.

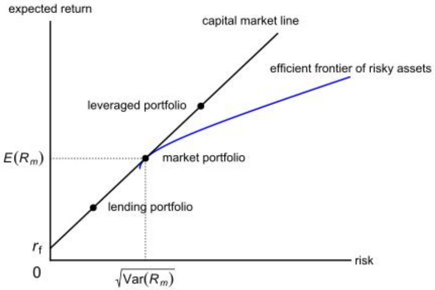
Modern Portfolio Theory: Intuition

Core intuition: Portfolio construction is about combining assets so the whole portfolio behaves better than isolated holdings.

- Return matters, but risk and correlation also matter.
- Diversification works when assets do not move perfectly together.
- A portfolio is not a shopping basket; weights determine exposures.

Finance lens: Robo-advisory turns these ideas into scalable portfolio recommendations under client constraints.

Visual Example: Efficient Frontier



Source: Wikimedia Commons, Capital market line of CAPM, used in Modern Portfolio Theory article.

Modern Portfolio Theory: Formula

- For portfolio weights w, expected return μ and covariance Σ:

$$\mathbb{E}[r_p] = w^\top \mu, \qquad \sigma_p^2 = w^\top \Sigma w.$$

- Mean-variance optimization balances return and risk:

$$\max_{w} \; w^\top \mu - \frac{\lambda}{2} w^\top \Sigma w.$$

Technical lens: Real implementation needs robust estimates, constraints, trading costs and monitoring. The elegant equation is the start, not the product.

Minimum Variance Portfolio

- The minimum variance portfolio minimizes total portfolio variance:
$$\min_{w} \; w^\top \Sigma w \quad \text{s.t.} \quad \mathbf{1}^\top w = 1.$$

- With two risky assets, the optimal weight in asset A is:
$$w_A^* = \frac{\mathrm{Var}(r_B) - \mathrm{Cov}(r_A, r_B)}{\mathrm{Var}(r_A) + \mathrm{Var}(r_B) - 2\mathrm{Cov}(r_A, r_B)}.$$

Takeaway: Diversification is driven by variance, covariance and correlation.

Practical Portfolio Constraints

- Long-only or limited shorting.
- Asset class, sector, region and issuer concentration limits.
- Turnover and transaction-cost constraints.
- Liquidity and minimum-trade-size constraints.
- Tax and account-specific restrictions.
- Compliance and suitability restrictions.

Finance lens: Constraints are not annoying details. They are how a theoretical portfolio becomes an investable portfolio.

Rebalancing

- Rebalancing restores a portfolio toward target allocation.
- Common rules:
    - calendar-based: monthly or quarterly;
    - threshold-based: rebalance if drift exceeds tolerance;
    - optimization-based: trade only when benefits exceed costs.
- AI can help forecast drift, costs and tax consequences.

Takeaway: Rebalancing is a trade-off between discipline, cost, tax and risk control.

Lab Demo: Robo Advisor Journey

Link: https://portal.aqumon.com/home/

WealthTech Product Matrix

- Digital onboarding and risk profiling.
- Portfolio proposal and suitability check.
- Goal-based investing.
- Rebalancing and tax-aware trading.
- Portfolio analytics, stress testing and performance attribution.
- Advisor copilot and client communication.
- Compliance monitoring and audit trail.

Core intuition: WealthTech is a system of connected services, not a single algorithm.

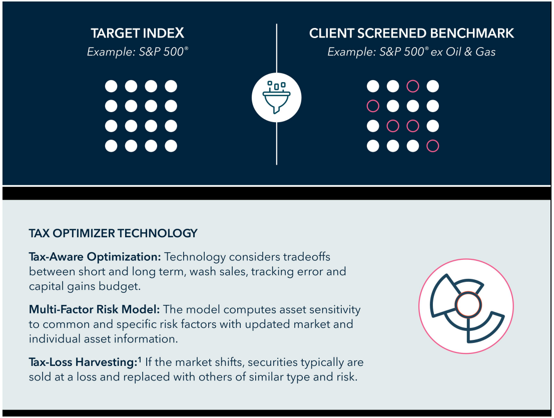
Direct Indexing: Plain-English View

Core intuition: Direct indexing means owning the individual stocks in an index-like portfolio instead of buying one index fund.

- The investor still aims to track a benchmark.
- But individual holdings allow more customization.
- Examples:
  - exclude restricted securities;
  - apply ESG or thematic tilts;
  - control sector or factor exposures;
  - harvest tax losses at individual security or lot level.

Suggested visual: Benchmark index decomposed into individual stock tiles, then personalized exclusions and tilts.

Visual Example: Direct Indexing Workflow



Source: BNY Pershing, *The Democratization of Direct Indexing*, direct-indexing workflow infographic.

Direct Indexing as an Optimization Problem

- Let w be portfolio weights and w_b benchmark weights.
- Tracking error objective:

$$\min_{w} \, (w - w_b)^\top \Sigma (w - w_b)$$

- Practical version adds costs and constraints:

$$\min_{w} \; \mathrm{TE}^2(w) + \lambda_1 \mathrm{Turnover}(w) + \lambda_2 \mathrm{TaxCost}(w)$$

$$\text{s.t. } \mathbf{1}^\top w = 1, \quad 0 \le w_i \le u_i, \quad \text{client constraints.}$$

Tax-Loss Harvesting: Plain-English View

Core intuition: Tax-loss harvesting realizes selected investment losses today to reduce tax, while trying to keep similar market exposure.

1. Identify tax lots with unrealized losses.
2. Sell selected lots.
3. Buy replacement exposures to maintain risk profile.
4. Avoid wash-sale or equivalent restrictions where applicable.
5. Track realized gains/losses and future cost basis.

Finance lens: Tax rules vary by jurisdiction; implementation must be localized and reviewed by qualified tax professionals.

Visual Example: Tax-Loss Harvesting Process

Source: Mellon Investments Corporation, *Direct Indexing: The Basics of Tax-Loss Harvesting*, process graphic.

Tax Harvesting: Decision Logic

- For tax lot i, define unrealized loss:

$$L_i = \max(0, \mathrm{CostBasis}_i - \mathrm{MarketValue}_i).$$

- A trade is attractive only if expected tax benefit exceeds costs and tracking-error impact:

$$\mathrm{NetBenefit}_i \approx \tau L_i - \mathrm{TradingCost}_i - \Delta\mathrm{TE}_i - \mathrm{OperationalPenalty}_i.$$

Technical lens: This is an optimization problem with tax lots, replacement securities, benchmark risk, trading costs and client constraints.

Direct Indexing + Tax Harvesting + AI

- Direct indexing creates more tax lots and more customization degrees of freedom.
- Tax harvesting searches across many lots, securities and replacement baskets.
- AI can support:
  - personalized constraint extraction from client documents;
  - opportunity ranking;
  - replacement-security selection;
  - explanation of after-tax trade-offs;
  - monitoring and audit trail.

Takeaway: The product value comes from personalization at scale, not from a single clever trade.

AI Agents in an Investment Firm

Core intuition: A real investment process involves many roles. Agent design can mirror this organizational structure.

Role	Main question	Typical output
Macro analyst	What is the regime?	regime view
Quant researcher	Does signal work?	research memo
Portfolio manager	How to implement?	target weights
Risk manager	What can go wrong?	risk challenge
Trader	How to trade?	execution plan
Compliance officer	Is it allowed?	approval/checklist
Committee secretary	What was decided?	audit record

Agent Role: Macro Analyst

- Monitors macro data, central bank communication, inflation, growth and liquidity conditions.
- Uses retrieval over research reports, official data and market indicators.
- Produces structured regime views:

$$\text{Regime} \in \{\text{Growth Up}, \text{Growth Down}\} \times \{\text{Inflation Up}, \text{Inflation Down}\}.$$

- Output should include evidence, confidence and changes from previous view.

Agent Role: Quant Researcher

- Converts investment hypotheses into testable factors.
- Pulls data, cleans data and checks survivorship and look-ahead bias.
- Runs IC analysis, robustness tests and parameter sweeps.
- Produces research memo with code, assumptions and performance diagnostics.

Finance lens: This is where AI can speed up research, but also where bad automation can create beautiful nonsense.

Agent Role: Portfolio Manager

- Translates signals into portfolio weights.
- Balances expected return, risk, turnover, liquidity, concentration and client constraints.
- Evaluates whether proposed trades are consistent with mandate and market view.
- Escalates unusual or high-impact changes to the investment committee.

Takeaway: The portfolio manager agent should not only ask "What has alpha?" but also "Can we implement it responsibly?"

Agent Role: Risk Manager

- Reviews exposures, drawdown, stress scenarios and tail risk.
- Challenges the strategy assumptions.
- Checks whether backtest risk is concentrated in a small number of episodes.
- Requires kill-switches, monitoring thresholds and post-trade review.

Technical lens: A strong risk agent should look for hidden beta, crowding, liquidity traps, factor concentration and regime sensitivity.

Agent Role: Compliance Officer

- Checks suitability, mandate restrictions and disclosure language.
- Reviews whether AI-generated analysis is supported by evidence.
- Prevents misleading claims such as unsupported “AI alpha” marketing.
- Maintains audit trail: prompts, retrieved sources, model outputs, approvals and final actions.

Takeaway: Compliance is not the enemy of innovation. It is the seat belt that lets the car go faster safely.

Agentic Investment Committee

1. Researcher proposes signal.
2. Risk agent challenges hidden exposures.
3. Portfolio agent proposes implementation.
4. Compliance agent checks mandate and disclosure.
5. Human committee approves, rejects or requests revision.

Core intuition: Agents do not replace accountability; they structure debate and preserve evidence.

Takeaway: The system should record dissent and rationale, not only the final output.

From Simple Investment Idea to Strategy

"Buy companies with improving fundamentals, reasonable valuation and strong cash conversion."

- Convert plain language into measurable signals:
  - earnings growth;
  - free cash flow growth;
  - return on invested capital;
  - valuation multiples;
  - balance sheet quality.
- Then test whether the signal predicts future returns after costs.

Strategy Research Pipeline

1. Define investment universe.
2. Define signal and data sources.
3. Clean data and avoid biases.
4. Analyze factor IC and monotonicity.
5. Tune parameters and rebalance frequency.
6. Construct portfolio weights.
7. Backtest with realistic costs and constraints.
8. Attribute performance and risk.
9. Prepare deployment and monitoring.

Suggested visual: Pipeline from idea to data, factor test, optimization, backtest, attribution and deployment.

AI Agent for Investment Research Implementation

- User gives simple idea in natural language.
- Agent converts idea into structured research plan.
- Data agent retrieves and validates data.
- Quant agent creates factors and runs IC analysis.
- Optimization agent builds portfolios under constraints.
- Backtest agent evaluates costs, risk and performance.
- Attribution agent explains return drivers.
- Compliance agent checks disclosures and suitability.

Summary

- AI evolved from rule-based systems to machine learning, deep learning, foundation models, agents and world models.
- FinTech applications require prediction, generation, reasoning, planning and governance.
- Robo-advisory, direct indexing and tax harvesting are examples of algorithmic personalization at scale.
- Agentic workflows can mimic the decision process of an investment firm, but must preserve accountability.
- A sound investment AI workflow goes from idea to data, IC, tuning, weighting, backtesting, attribution and monitoring.

Questions?

References and Further Reading

- Turing, A. M. (1950), "Computing Machinery and Intelligence."
- Vaswani et al. (2017), "Attention Is All You Need."
- Sutton and Barto, "Reinforcement Learning: An Introduction."
- Murphy, "Machine Learning: A Probabilistic Perspective."
- Goodfellow, Bengio and Courville, "Deep Learning."
- Research and industry materials on foundation models, RAG, AI agents and world models.
- World Bank, "Robo-Advisors: Investing through Machines."
- SEC, robo-adviser guidance and fiduciary duty materials.
- Materials from major asset managers on direct indexing and tax-loss harvesting.