

## Introduction to Artificial Intelligence in Finance and Quantitative Analysis

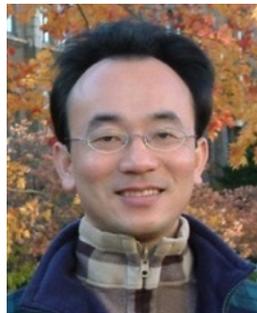
1121AIFQA01

MBA, IM, NTPU (M5276) (Fall 2023)

Tue 2, 3, 4 (9:10-12:00) (B3F17)



<https://meet.google.com/paj-zhhj-mya>



Min-Yuh Day, Ph.D,  
Associate Professor

Institute of Information Management, National Taipei University

<https://web.ntpu.edu.tw/~myday>





# Min-Yuh Day, Ph.D.



**Associate Professor, Information Management, NTPU**

**Visiting Scholar, IIS, Academia Sinica**

**Ph.D., Information Management, NTU**

**Director, Intelligent Financial Innovation Technology, IFIT Lab, IM, NTPU**

**Associate Director, Fintech and Green Finance Center, NTPU**

**Artificial Intelligence, Financial Technology, Big Data Analytics,  
Data Mining and Text Mining, Electronic Commerce**

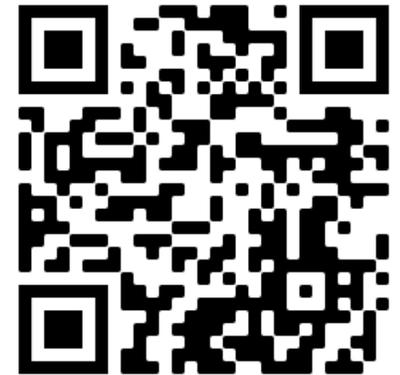


# Course Syllabus

## National Taipei University

### Academic Year 112, 1<sup>st</sup> Semester (Fall 2023)

- **Course Title: Artificial Intelligence in Finance and Quantitative Analysis**
- **Instructor: Min-Yuh Day**
- **Course Class: MBA, IM, NTPU (3 Credits, Elective)**
- **Details**
  - **EMI Course**  
**(3 Credits, Elective, One Semester) (M5276)**
- **Time & Place: Tue, 2, 3, 4, (9:10-12:00) (B3F17)**
- **Google Meet: <https://meet.google.com/paj-zhhj-mya>**



[https://meet.google.com/  
paj-zhhj-mya](https://meet.google.com/paj-zhhj-mya)



# Course Objectives

- 1. Understand the fundamental concepts and research issues of Artificial Intelligence in Finance and Quantitative Analysis.**
- 2. Equip with Hands-on practices of Artificial Intelligence in Finance and Quantitative Analysis.**
- 3. Conduct information systems research in the context of Artificial Intelligence in Finance and Quantitative Analysis.**

# Course Outline

- This course introduces the **fundamental concepts, research issues, and hands-on practices of AI in Finance and Quantitative Analysis.**
- **Topics include:**
  1. Introduction to Artificial Intelligence in Finance and Quantitative Analysis
  2. AI in FinTech: Metaverse, Web3, DeFi, NFT, Financial Services Innovation and Applications
  3. Investing Psychology and Behavioral Finance
  4. Event Studies in Finance
  5. Finance Theory
  6. Data-Driven Finance
  7. Financial Econometrics
  8. AI-First Finance
  9. Deep Learning in Finance
  10. Reinforcement Learning in Finance
  11. Algorithmic Trading, Risk Management, Trading Bot and Event-Based Backtesting
  12. Case Study on AI in Finance and Quantitative Analysis.

# Core Competence

- **Exploring new knowledge in information technology, system development and application 80 %**
- **Internet marketing planning ability 10 %**
- **Thesis writing and independent research skills 10 %**

# Four Fundamental Qualities

- **Professionalism**
  - **Creative thinking and Problem-solving 40 %**
  - **Comprehensive Integration 40 %**
- **Interpersonal Relationship**
  - **Communication and Coordination 10 %**
  - **Teamwork 5 %**
- **Ethics**
  - **Honesty and Integrity 0 %**
  - **Self-Esteem and Self-reflection 0 %**
- **International Vision**
  - **Caring for Diversity 0 %**
  - **Interdisciplinary Vision 5 %**

# College Learning Goals

- **Ethics/Corporate Social Responsibility**
- **Global Knowledge/Awareness**
- **Communication**
- **Analytical and Critical Thinking**

# Department Learning Goals

- **Information Technologies and System Development Capabilities**
- **Internet Marketing Management Capabilities**
- **Research capabilities**

# Syllabus

**Week Date Subject/Topics**

- 1 2023/09/12 Introduction to Artificial Intelligence in Finance and Quantitative Analysis**
- 2 2023/09/19 AI in FinTech: Metaverse, Web3, DeFi, NFT, Financial Services Innovation and Applications**
- 3 2023/09/26 Investing Psychology and Behavioral Finance**
- 4 2023/10/03 Event Studies in Finance**
- 5 2023/10/10 National Day (Day off)**
- 6 2023/10/17 Case Study on AI in Finance and Quantitative Analysis I**

# Syllabus

Week	Date	Subject/Topics
7	2023/10/24	Finance Theory and Data-Driven Finance
8	2023/10/31	Midterm Project Report
9	2023/11/07	Financial Econometrics
10	2023/11/14	AI-First Finance
11	2023/11/21	Industry Practices of AI in Finance and Quantitative Analysis
12	2023/11/28	Case Study on AI in Finance and Quantitative Analysis II

# Syllabus

<b>Week</b>	<b>Date</b>	<b>Subject/Topics</b>
<b>13</b>	<b>2023/12/05</b>	<b>Deep Learning in Finance; Reinforcement Learning in Finance</b>
<b>14</b>	<b>2023/12/12</b>	<b>Algorithmic Trading; Risk Management; Trading Bot and Event-Based Backtesting</b>
<b>15</b>	<b>2023/12/19</b>	<b>Final Project Report I</b>
<b>16</b>	<b>2023/12/26</b>	<b>Final Project Report II</b>

# Teaching Methods and Activities

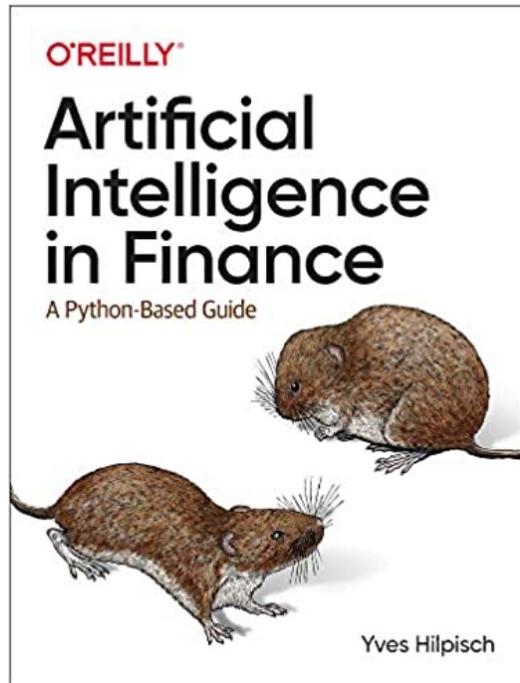
- **Lecture**
- **Discussion**
- **Practicum**

# Evaluation Methods

- **Individual Presentation 60 %**
- **Group Presentation 10 %**
- **Case Report 10 %**
- **Class Participation 10 %**
- **Assignment 10 %**

# Required Texts

- **Yves Hilpisch (2020),  
Artificial Intelligence in Finance: A Python-Based Guide,  
O'Reilly Media.**



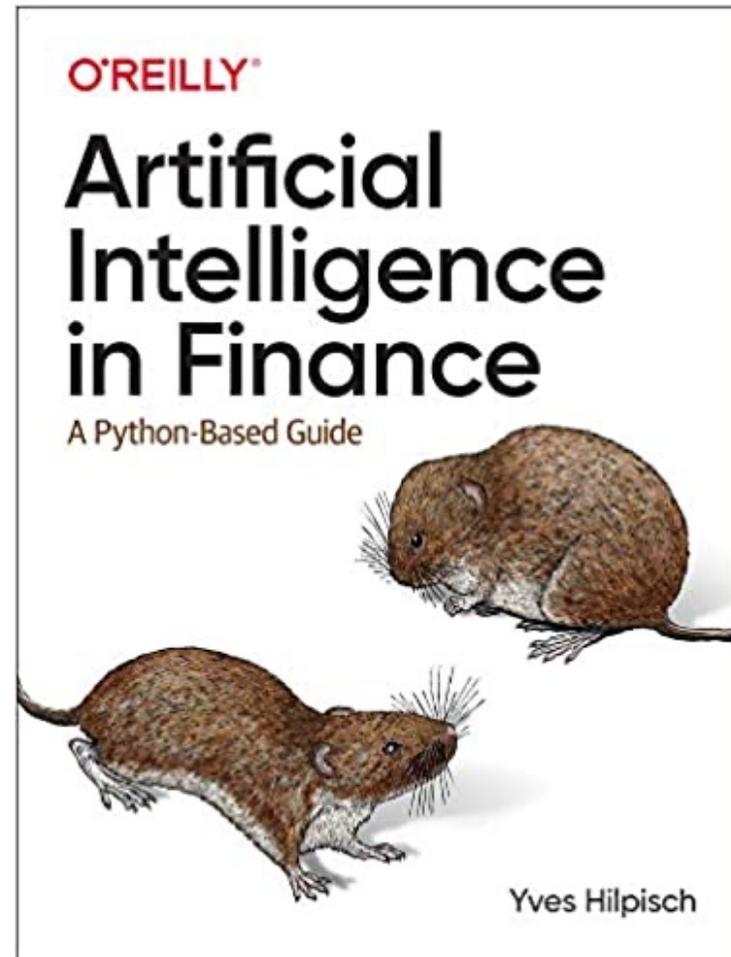
# Reference Books

- **Stefan Jansen (2020), Machine Learning for Algorithmic Trading: Predictive models to extract signals from market and alternative data for systematic trading strategies with Python, 2nd Edition, Packt Publishing.**
- **Aurélien Géron (2022), Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 3rd Edition, O'Reilly Media.**
- **Hariom Tatsat, Sahil Puri, Brad Lookabaugh (2020), Machine Learning and Data Science Blueprints for Finance: From Building Trading Strategies to Robo-Advisors Using Python, O'Reilly Media**
- **Chris Kelliher (2022), Quantitative Finance With Python: A Practical Guide to Investment Management, Trading, and Financial Engineering, Chapman and Hall/CRC.**
- **Simon Thompson (2023), Green and Sustainable Finance: Principles and Practice in Banking, Investment and Insurance, 2nd Edition, Kogan Page.**
- **Cino Robin Castelli, Cyril Shmatov (2022), Quantitative Methods for ESG Finance, Wiley**
- **Abdullah Karasan (2021), Machine Learning for Financial Risk Management with Python: Algorithms for Modeling Risk, O'Reilly Media.**
- **Yves Hilpisch (2018), Python for Finance: Mastering Data-Driven Finance, 2nd Edition, O'Reilly Media.**

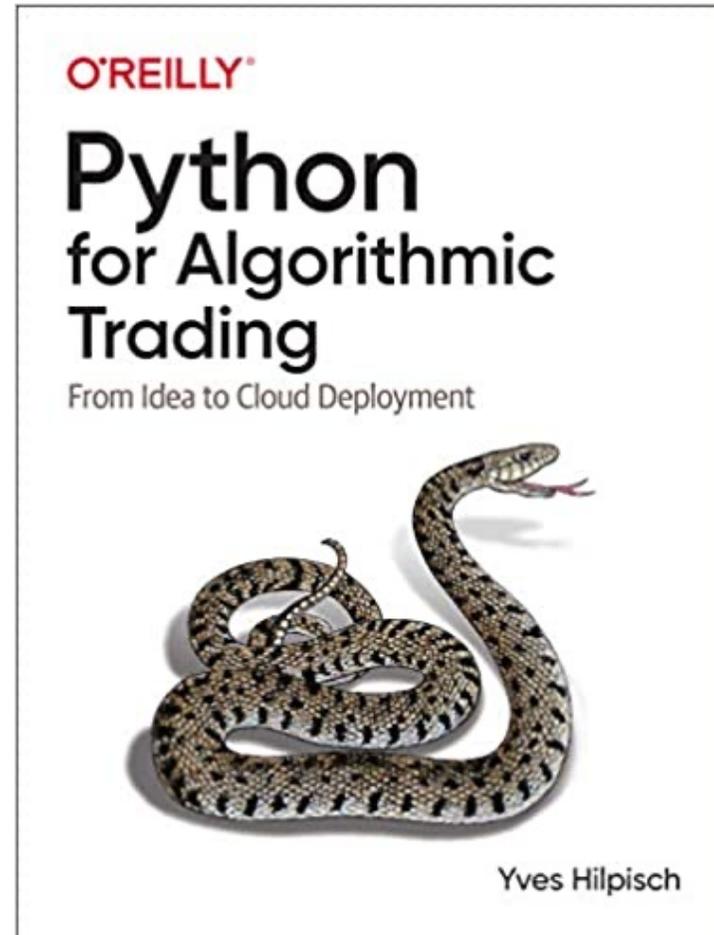
# Other References

- **Paolo Sironi (2016), FinTech Innovation: From Robo-Advisors to Goal Based Investing and Gamification, Wiley.**
- **Yves Hilpisch (2020), Financial Theory with Python: A Gentle Introduction, O'Reilly Media.**
- **Yves Hilpisch (2020), Python for Algorithmic Trading: From Idea to Cloud Deployment, O'Reilly Media.**
- **Yuxing Yan (2017), Python for Finance: Apply powerful finance models and quantitative analysis with Python, Second Edition, Packt Publishing.**

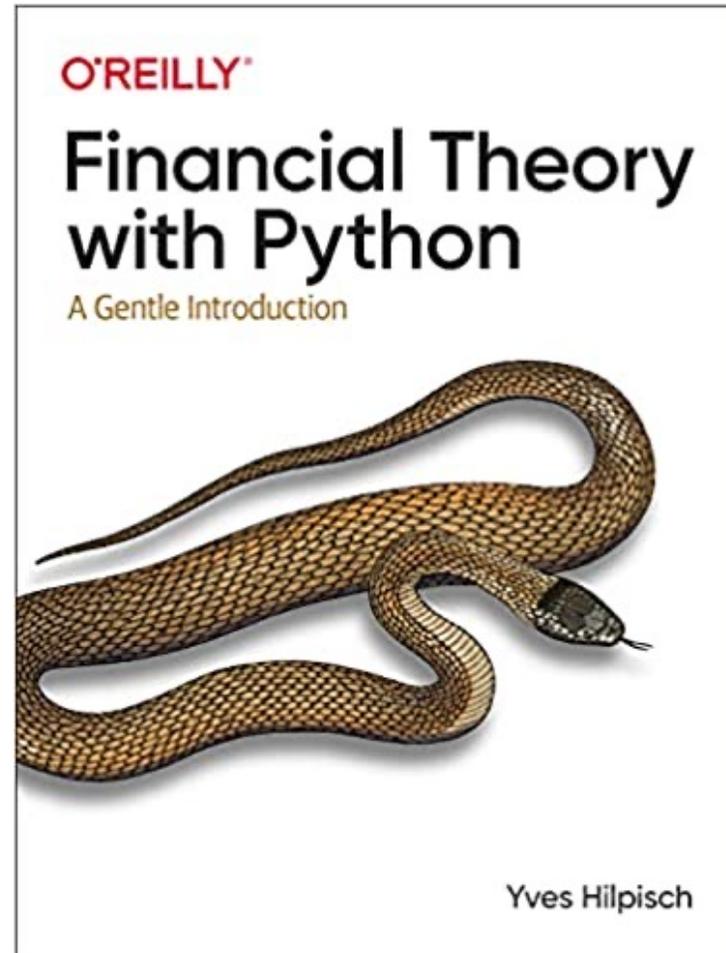
Yves Hilpisch (2020),  
**Artificial Intelligence in Finance:**  
**A Python-Based Guide,**  
O'Reilly



Yves Hilpisch (2020),  
**Python for Algorithmic Trading:**  
From Idea to Cloud Deployment,  
O'Reilly



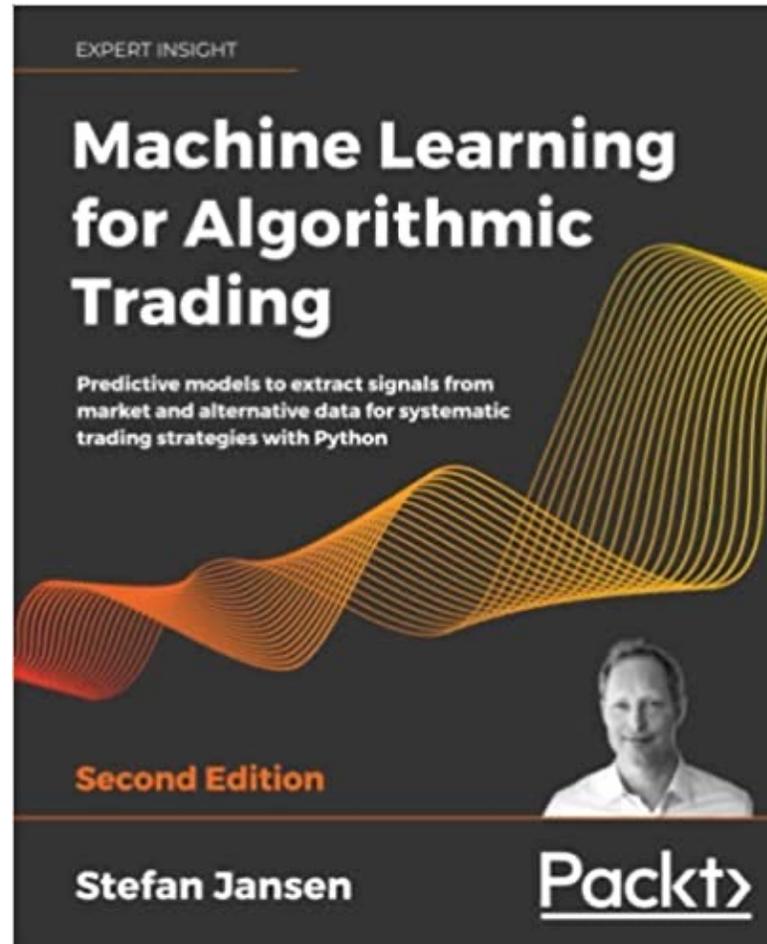
Yves Hilpisch (2021),  
**Financial Theory with Python:**  
**A Gentle Introduction,**  
O'Reilly



Stefan Jansen (2020),

# Machine Learning for Algorithmic Trading:

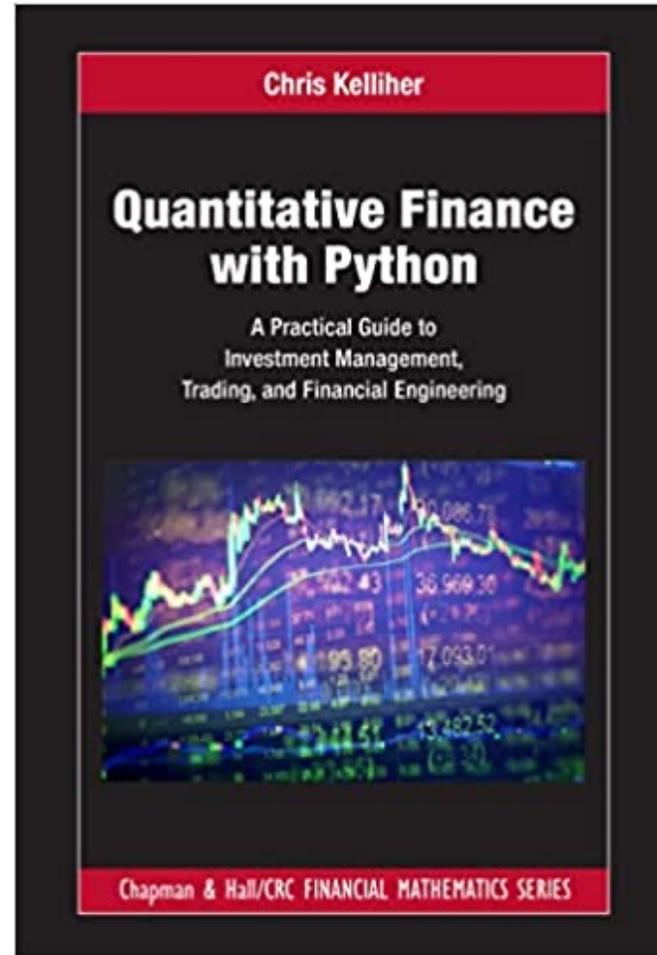
Predictive models to extract signals from market and alternative data for systematic trading strategies with Python, 2nd Edition,  
Packt Publishing.



**Chris Kelliher (2022),**

**Quantitative Finance With Python:**

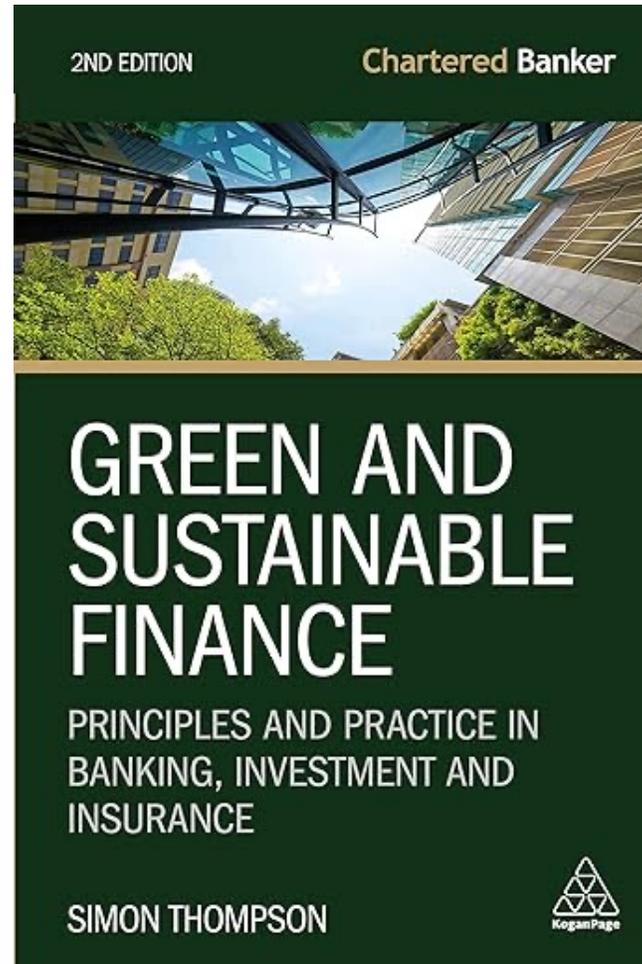
**A Practical Guide to Investment Management, Trading, and Financial Engineering,  
Chapman and Hall/CRC.**



Simon Thompson (2023),

# Green and Sustainable Finance:

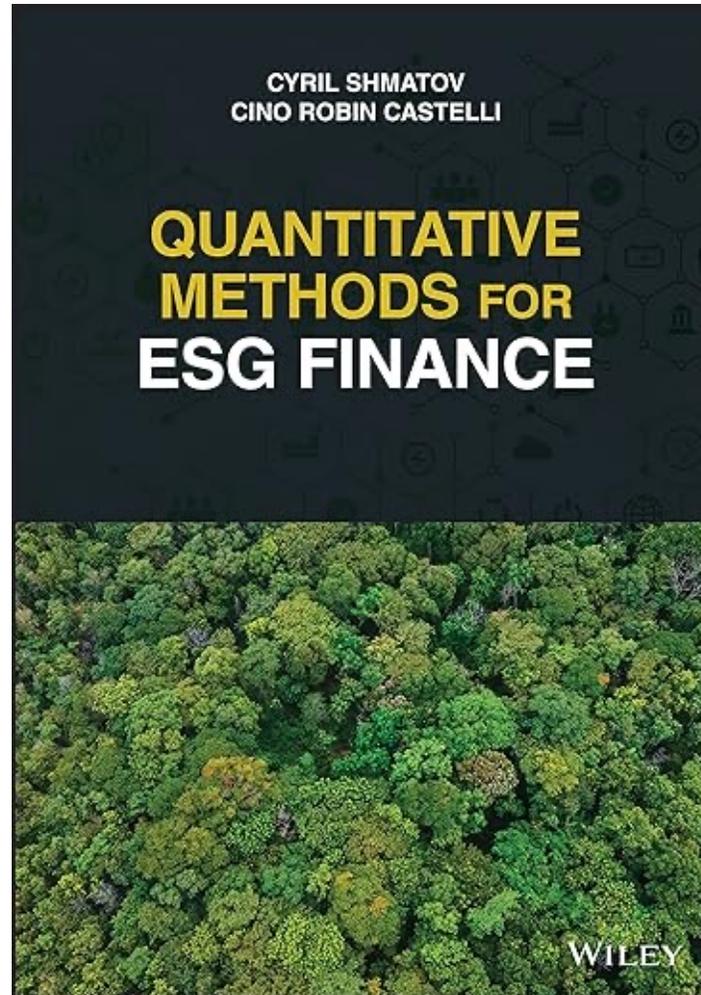
Principles and Practice in Banking, Investment and Insurance, 2nd Edition,  
Kogan Page



Cino Robin Castelli, Cyril Shmatov (2022),

# Quantitative Methods for ESG Finance,

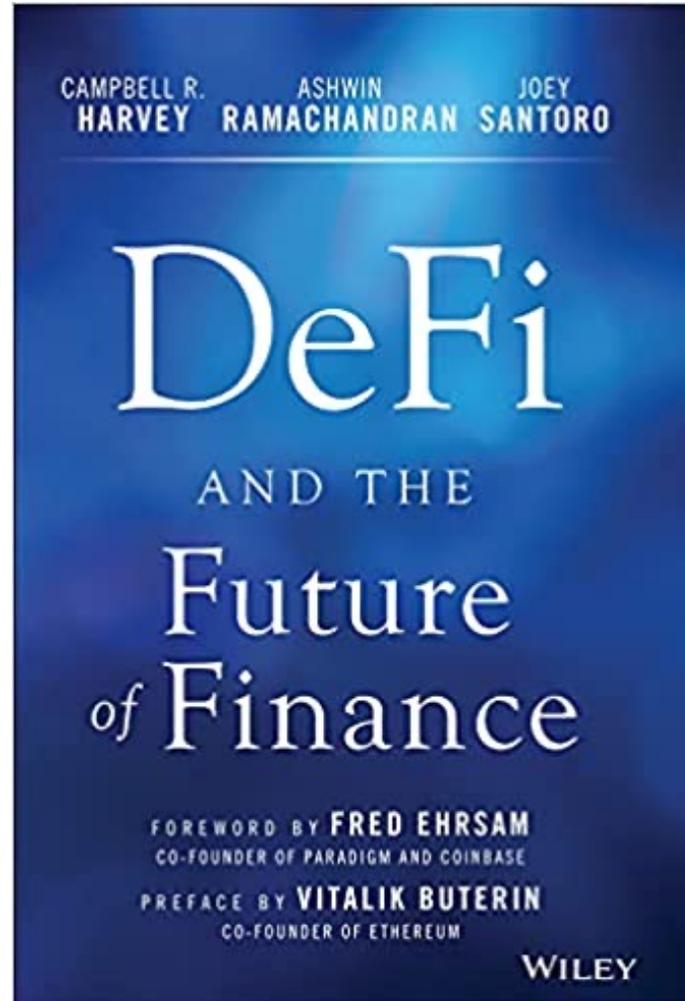
Wiley



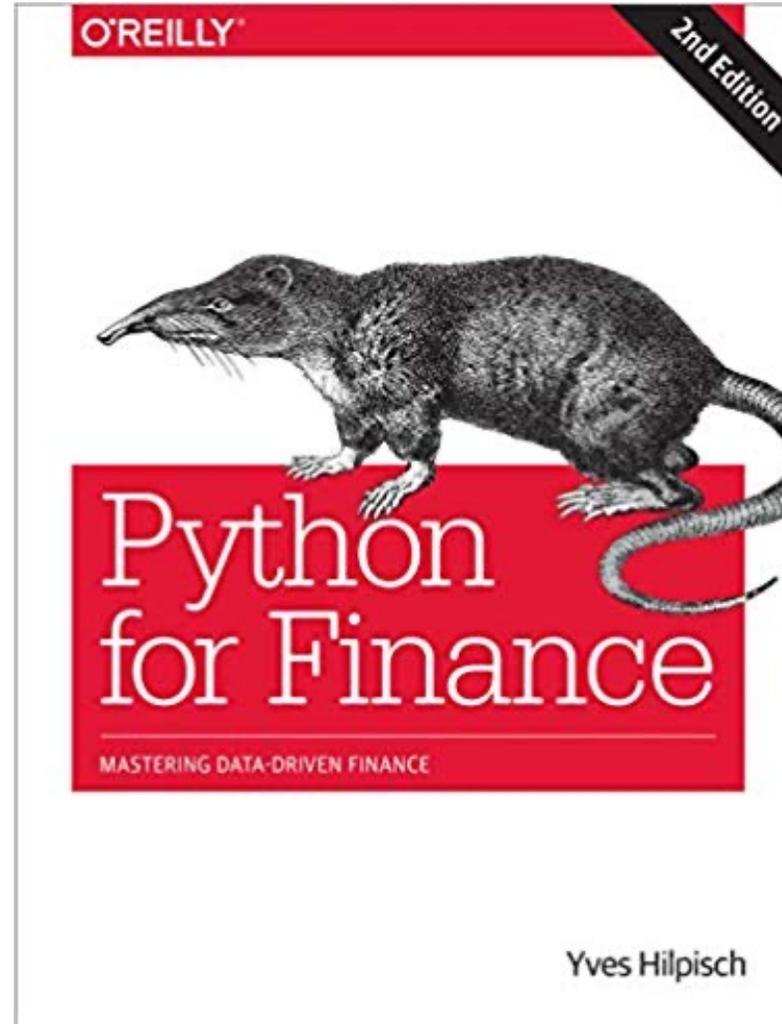
Campbell R. Harvey, Ashwin Ramachandran, Joey Santoro, Fred Ehrsam (2021),

# DeFi and the Future of Finance,

Wiley



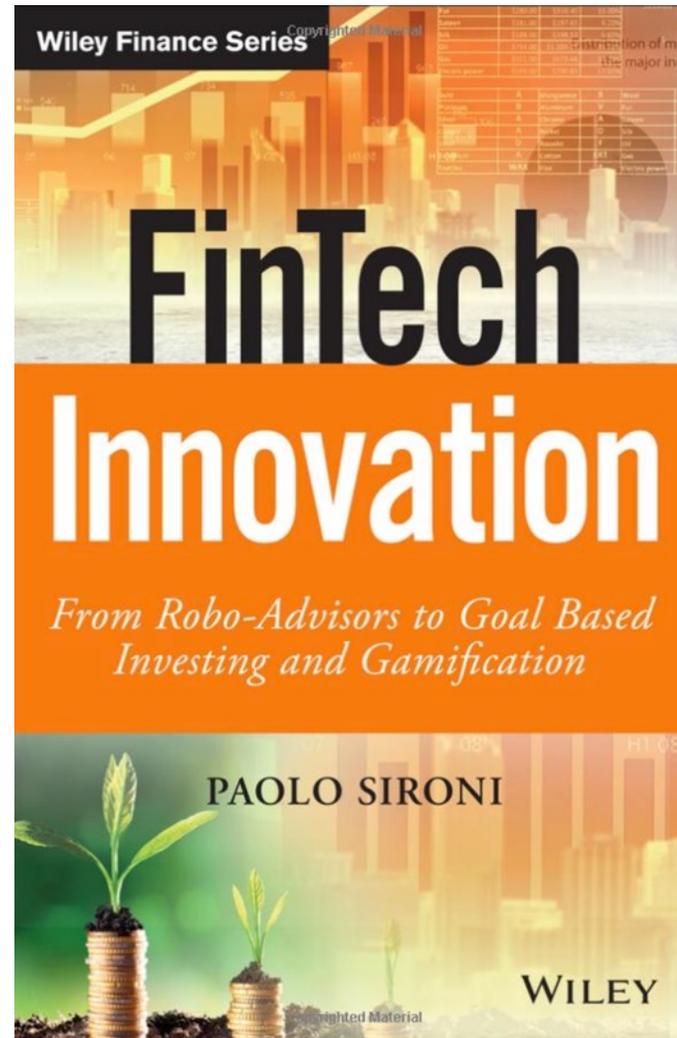
Yves Hilpisch (2018),  
**Python for Finance: Mastering Data-Driven Finance,**  
O'Reilly



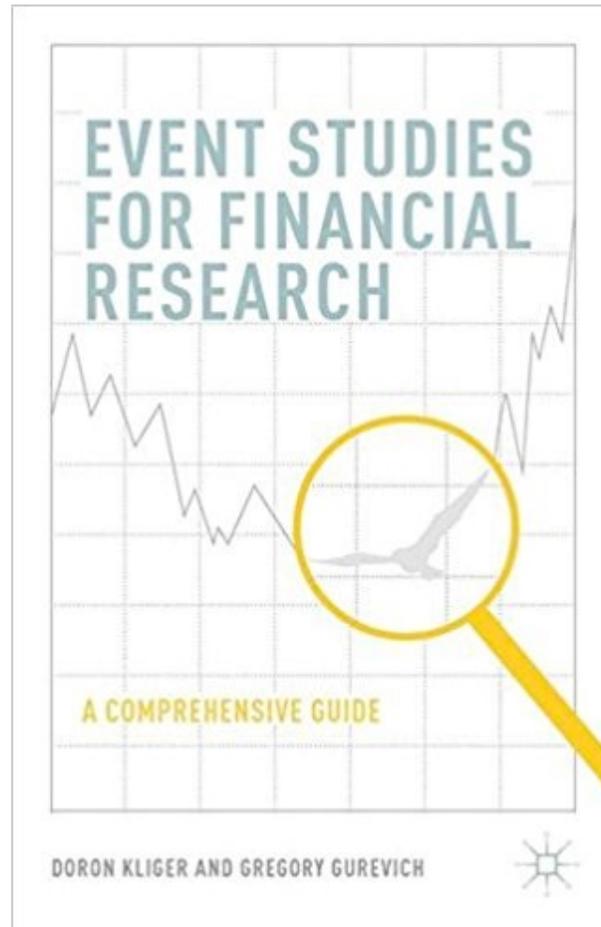
Paolo Sironi (2016)

# FinTech Innovation:

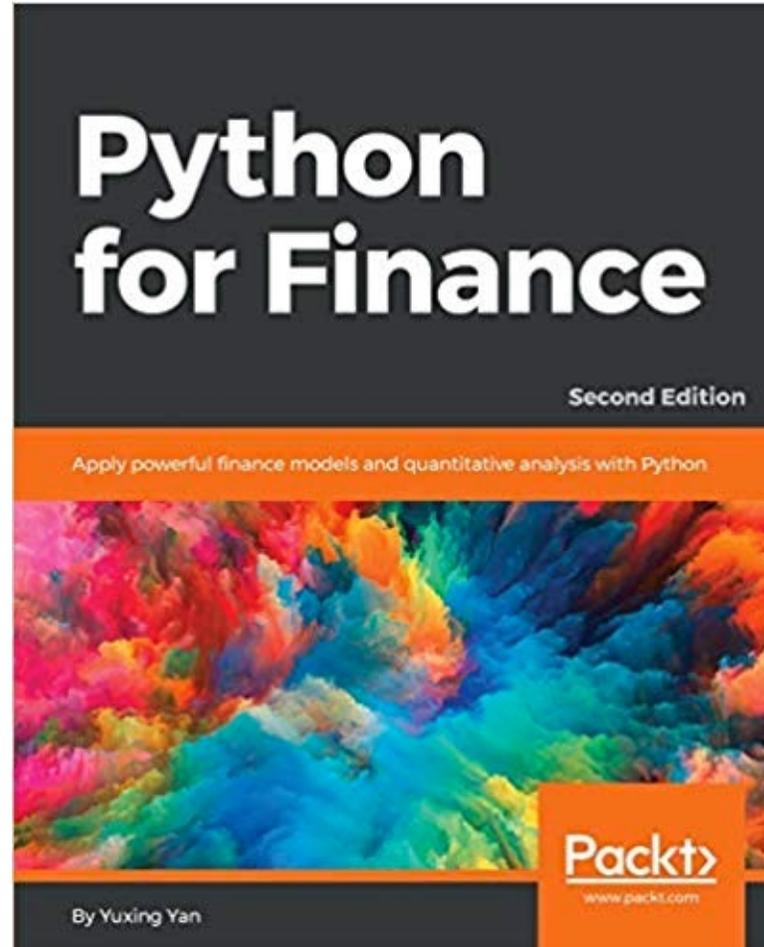
From Robo-Advisors to Goal Based Investing and Gamification,  
Wiley



Doron Kliger and Gregory Gurevich (2014),  
**Event Studies for Financial Research:**  
**A Comprehensive Guide,**  
Palgrave Macmillan



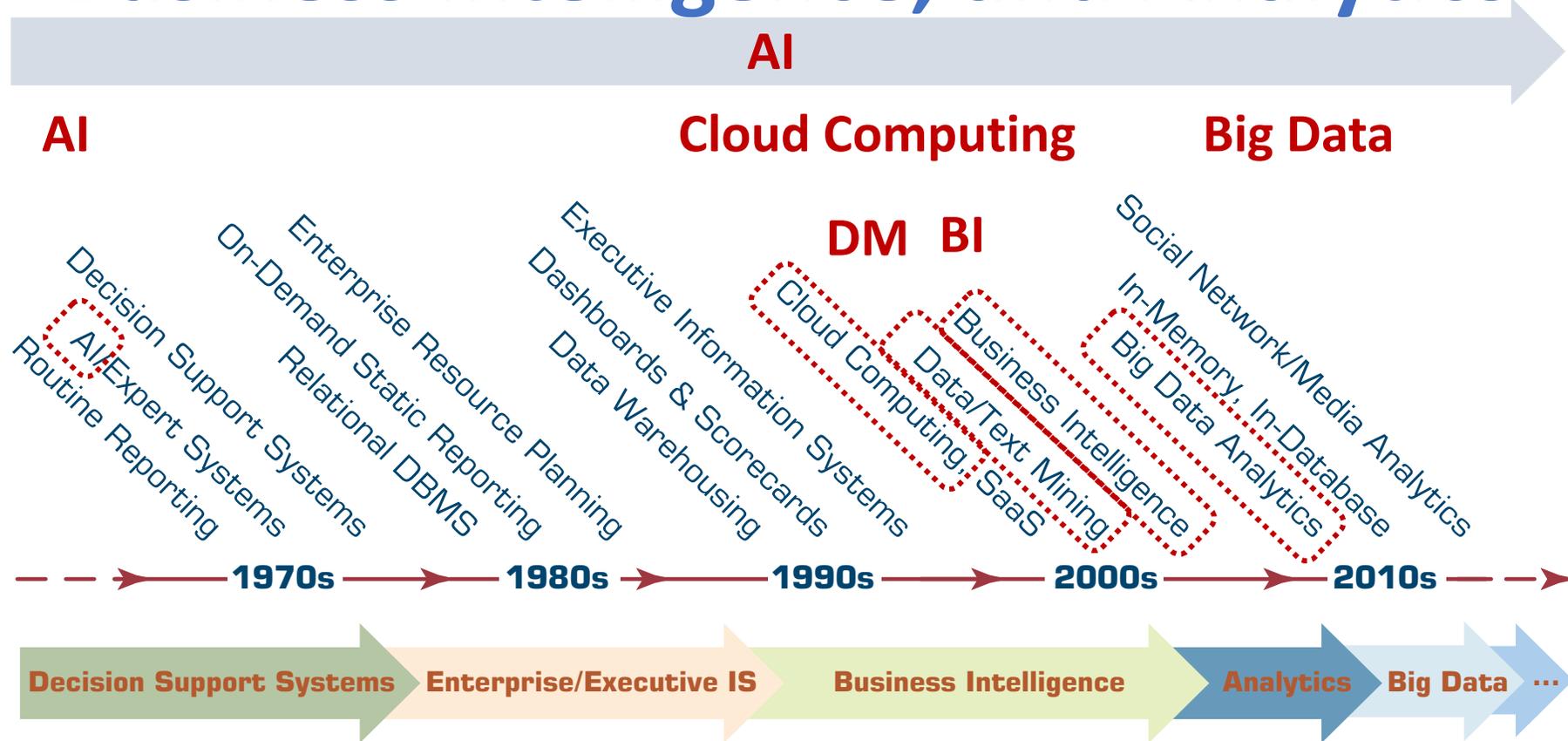
Yuxing Yan (2017),  
**Python for Finance: Apply powerful finance models  
and quantitative analysis with Python**, Second Edition,  
Packt Publishing



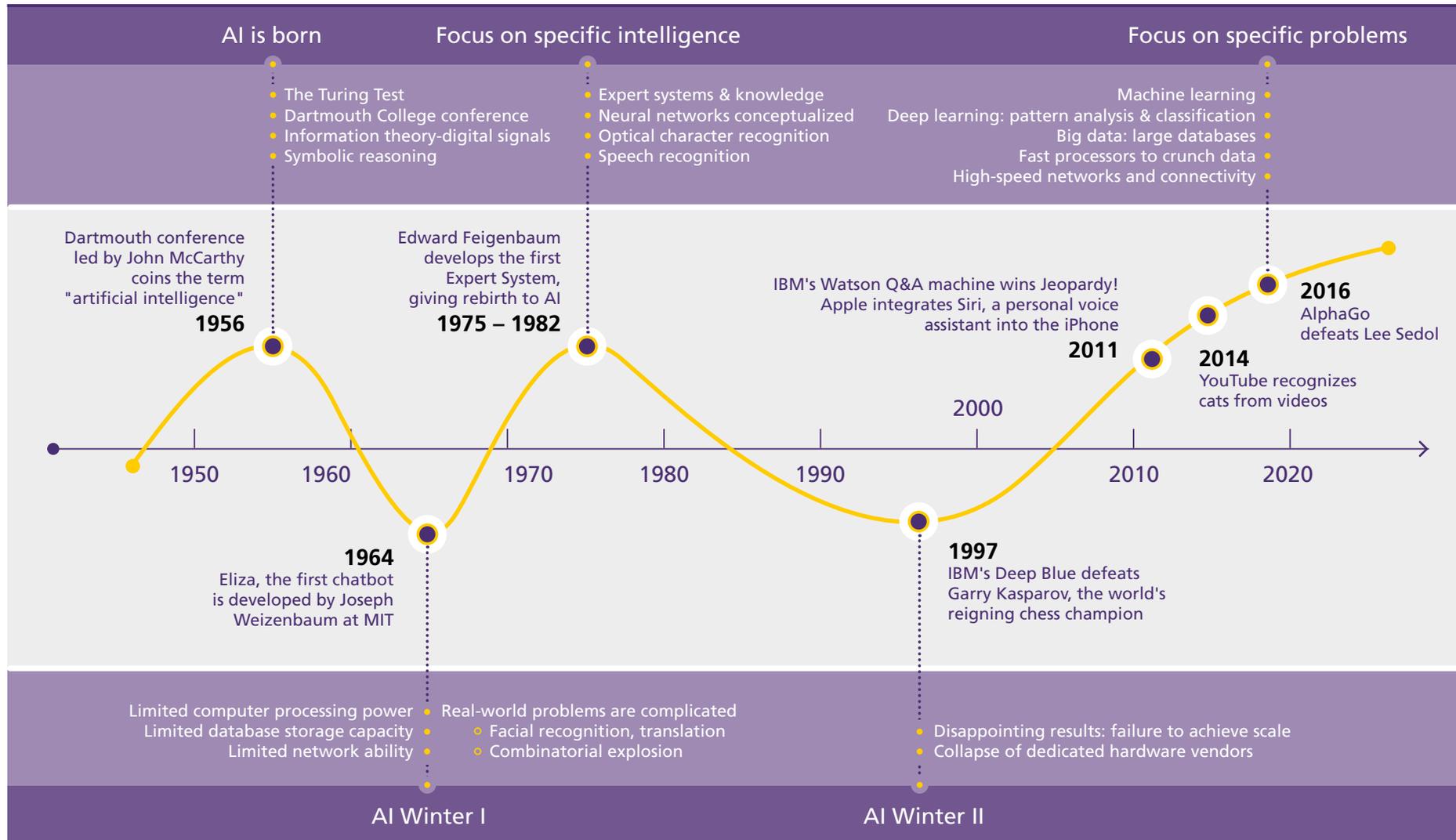
# Artificial Intelligence (AI)

# AI, Big Data, Cloud Computing

## Evolution of Decision Support, Business Intelligence, and Analytics



# The Rise of AI



# **Definition of Artificial Intelligence (A.I.)**

# Artificial Intelligence

**“... the science and  
engineering  
of  
making  
intelligent machines”**

**(John McCarthy, 1955)**

# Artificial Intelligence

**“... technology that  
thinks and acts  
like humans”**

# Artificial Intelligence

**“... intelligence  
exhibited by machines  
or software”**

# 4 Approaches of AI

<b>Thinking Humanly</b>	<b>Thinking Rationally</b>
<b>Acting Humanly</b>	<b>Acting Rationally</b>

# 4 Approaches of AI

<p><b>2.</b> <b>Thinking Humanly: The Cognitive Modeling Approach</b></p>	<p><b>3.</b> <b>Thinking Rationally: The “Laws of Thought” Approach</b></p>
<p><b>1.</b> <b>Acting Humanly: The Turing Test Approach</b> (1950)</p>	<p><b>4.</b> <b>Acting Rationally: The Rational Agent Approach</b></p>

# AI Acting Humanly: The Turing Test Approach

(Alan Turing, 1950)

- Knowledge Representation
- Automated Reasoning
- Machine Learning (ML)
  - Deep Learning (DL)
- Computer Vision (Image, Video)
- Natural Language Processing (NLP)
- Robotics

# FinTech

# Financial Technology



# Financial Technology

## FinTech

**“providing  
financial services  
by making use of  
software and  
modern technology”**

# Financial Revolution with Fintech

## A financial services revolution

### Consumer Trends



1. Simplification



2. Transparency



3. Analytics



4. Reduced Friction

# FinTech: Financial Services Innovation



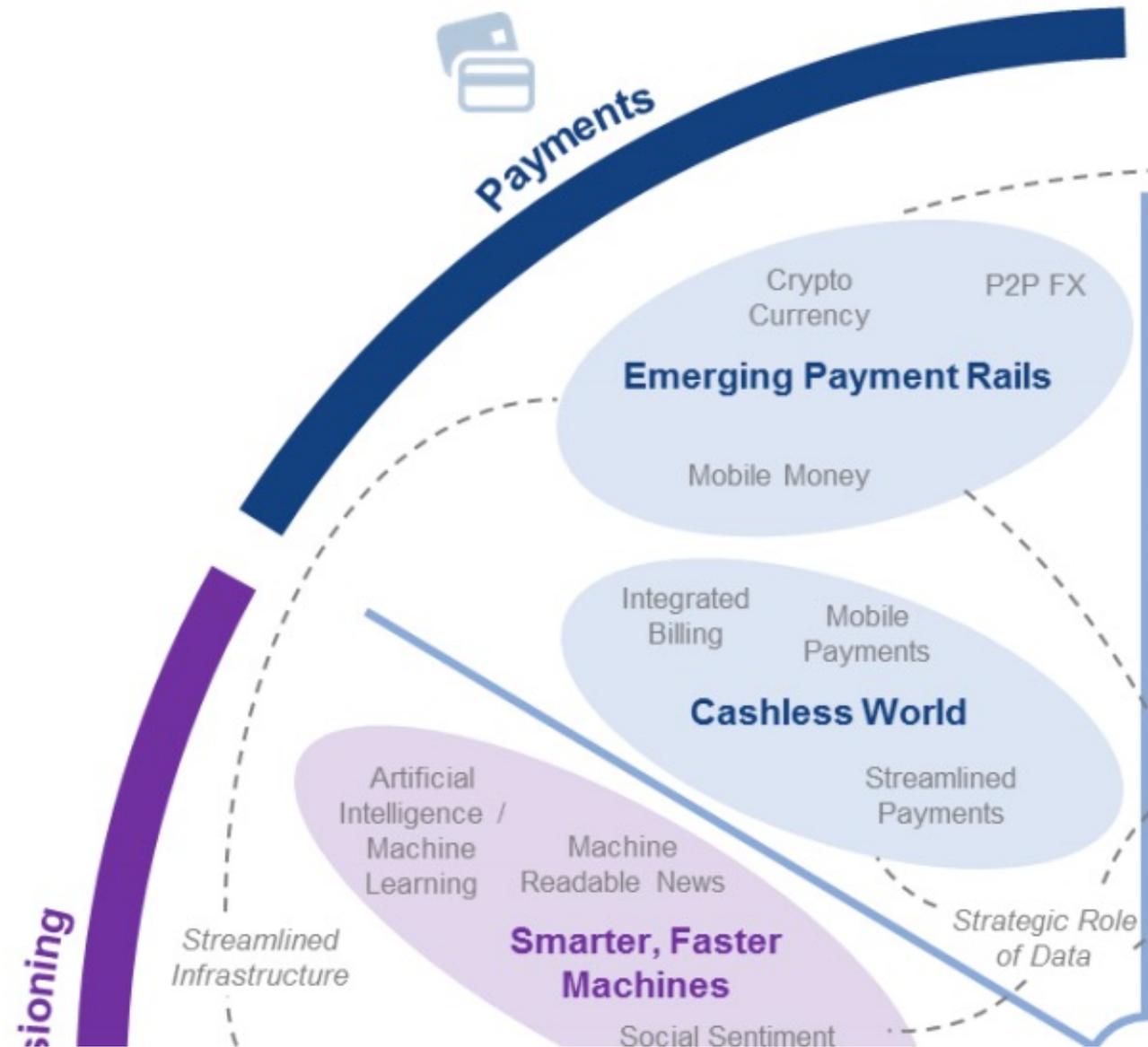
# **FinTech:**

## **Financial Services Innovation**

- 1. Payments**
- 2. Insurance**
- 3. Deposits & Lending**
- 4. Capital Raising**
- 5. Investment Management**
- 6. Market Provisioning**

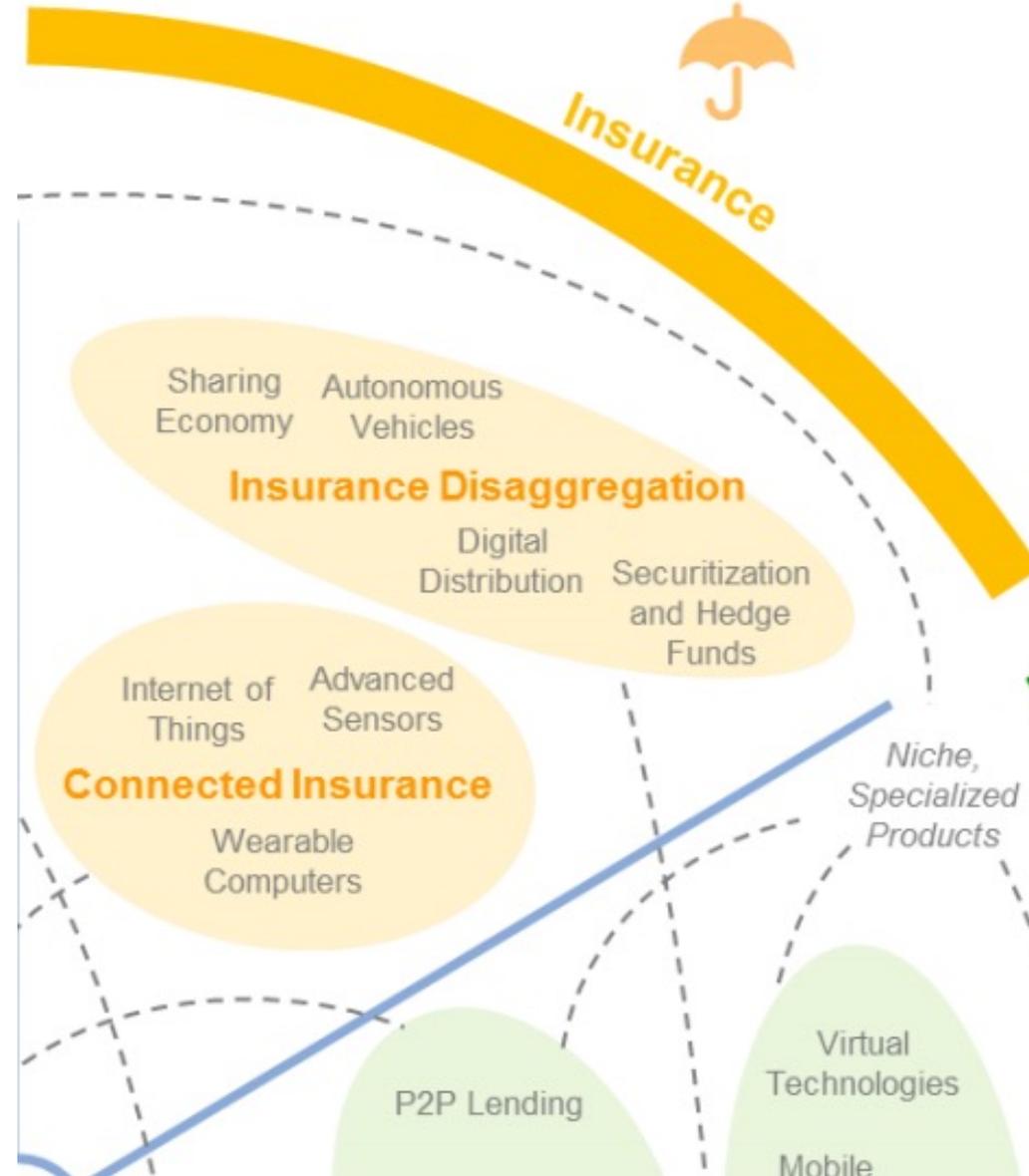
1

# FinTech: Payment



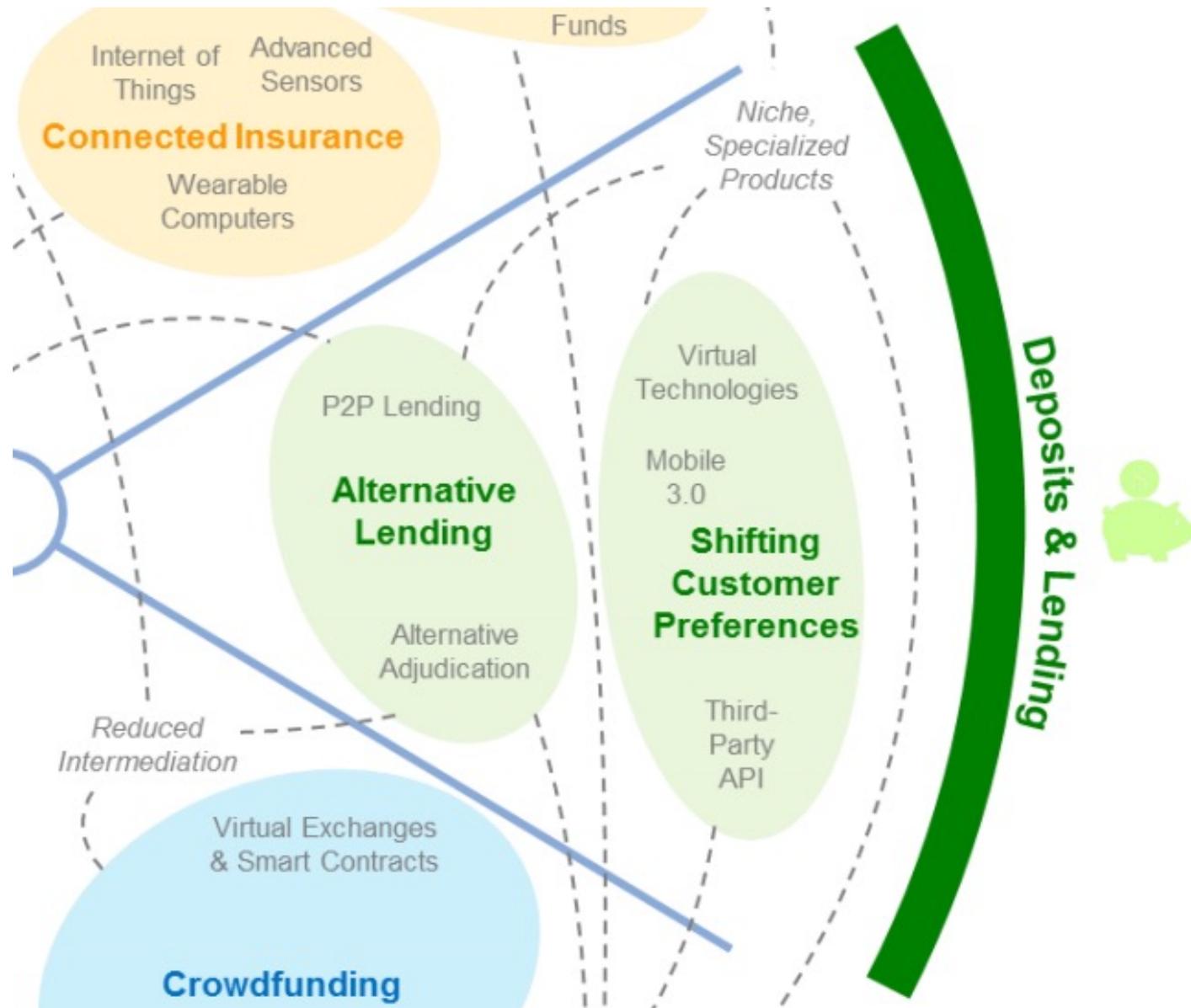
# 2

# FinTech: Insurance



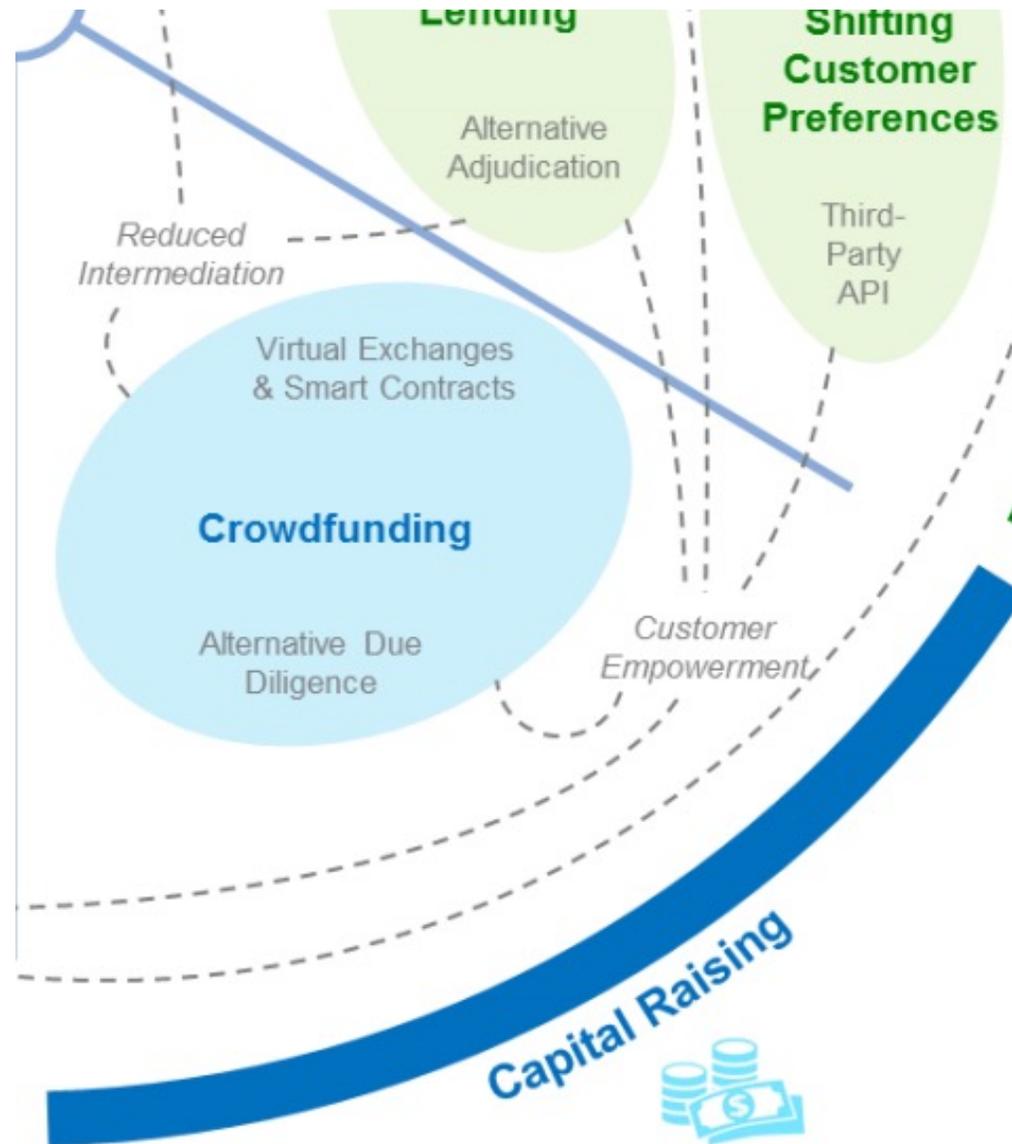
# 3

## FinTech: Deposits & Lending

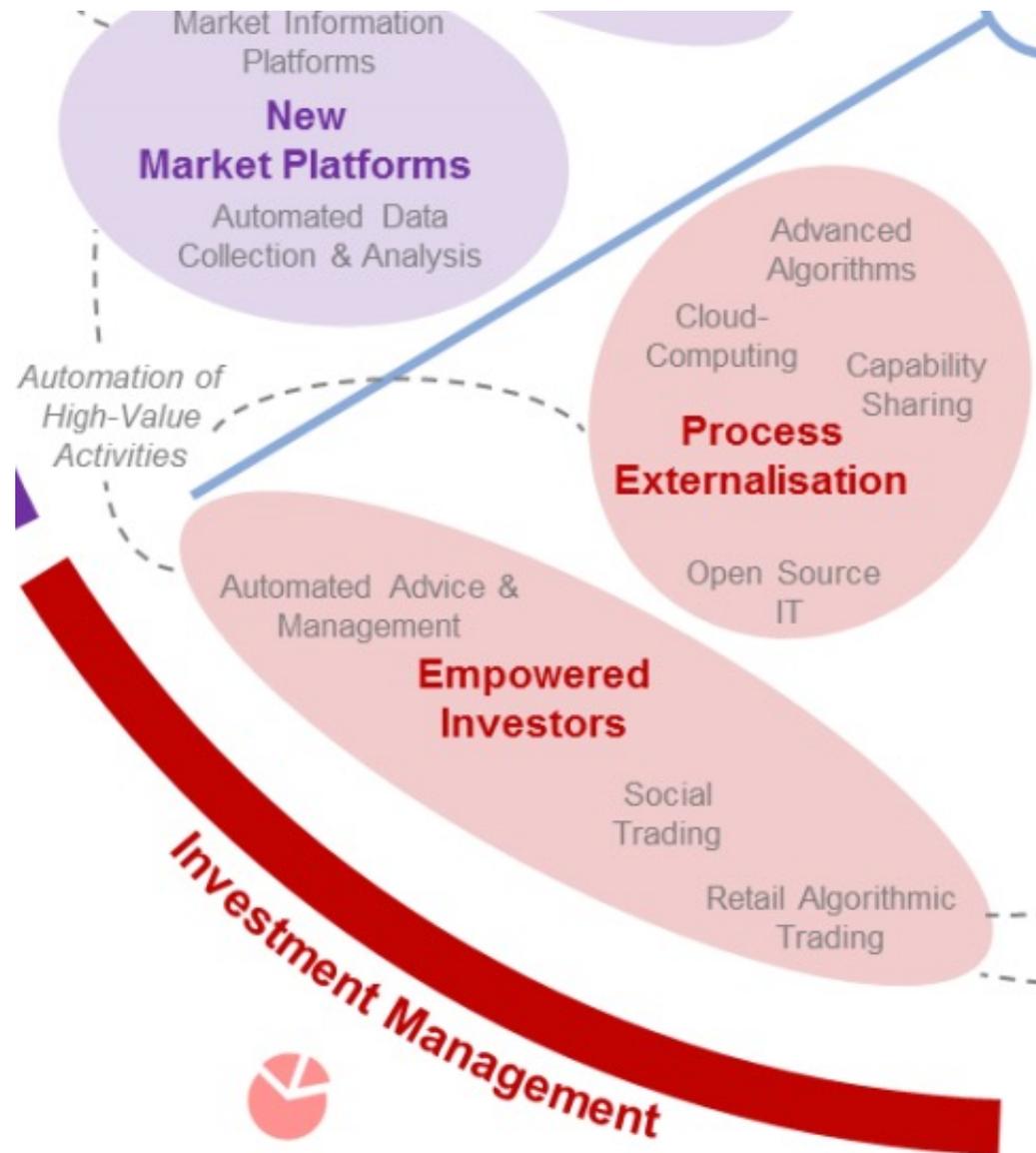


# 4

# FinTech: Capital Raising

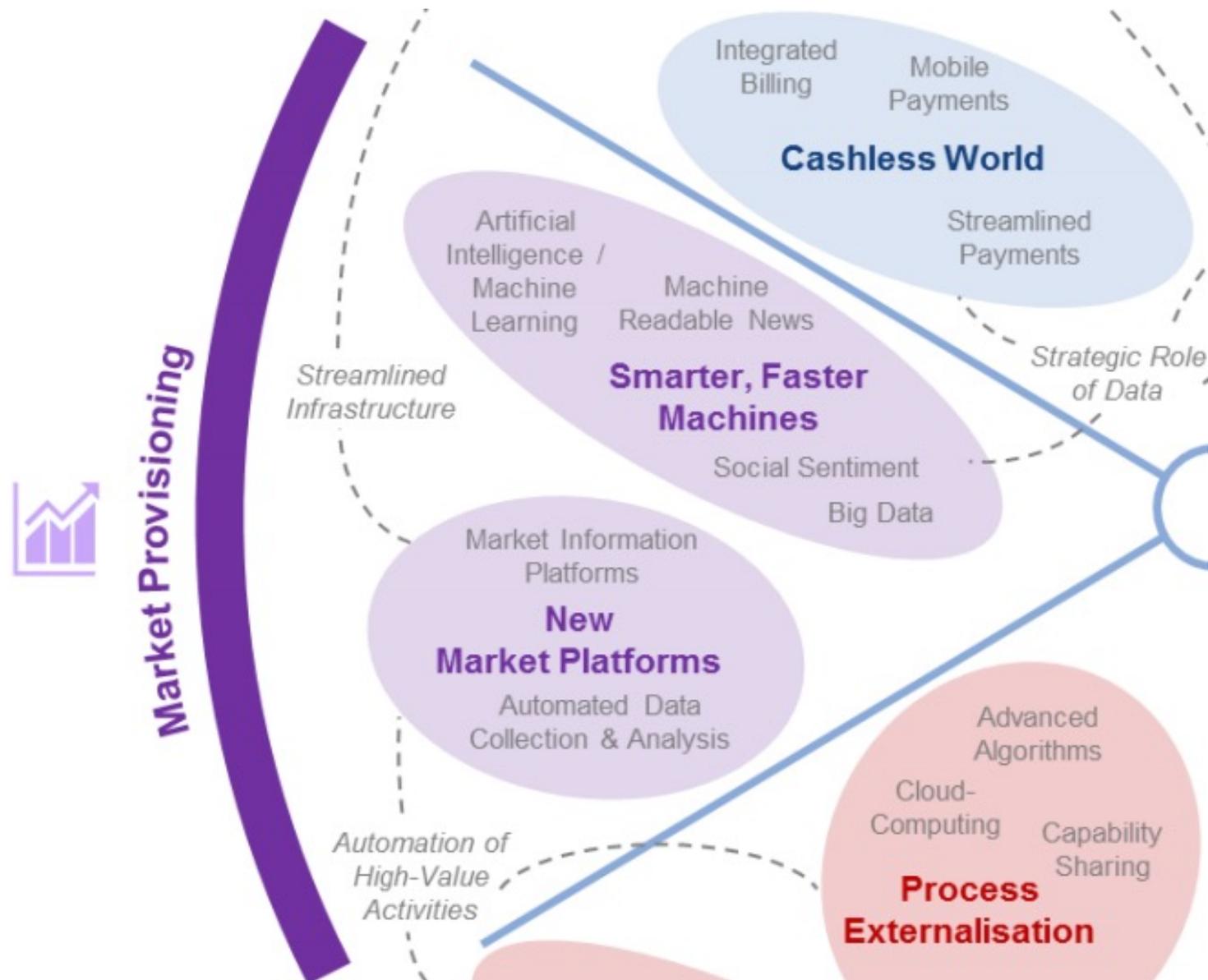


# 5 FinTech: Investment Management



# 6

# FinTech: Market Provisioning



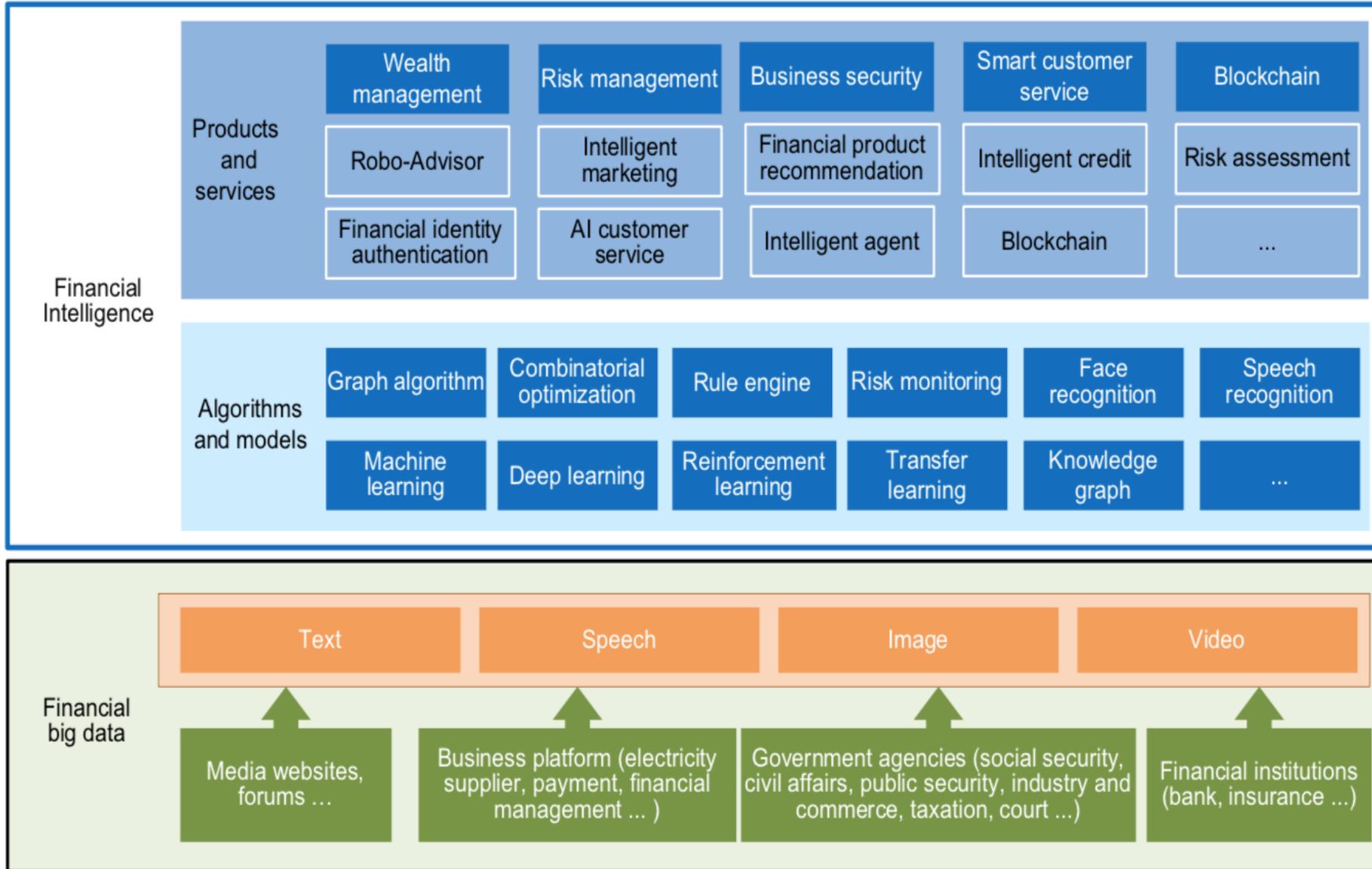
**AI**

**in**

**FinTech**

# FinBrain: when Finance meets AI 2.0

(Zheng et al., 2019)



Source: Xiao-lin Zheng, Meng-ying Zhu, Qi-bing Li, Chao-chao Chen, and Yan-chao Tan (2019), "Finbrain: When finance meets AI 2.0." Frontiers of Information Technology & Electronic Engineering 20, no. 7, pp. 914-924

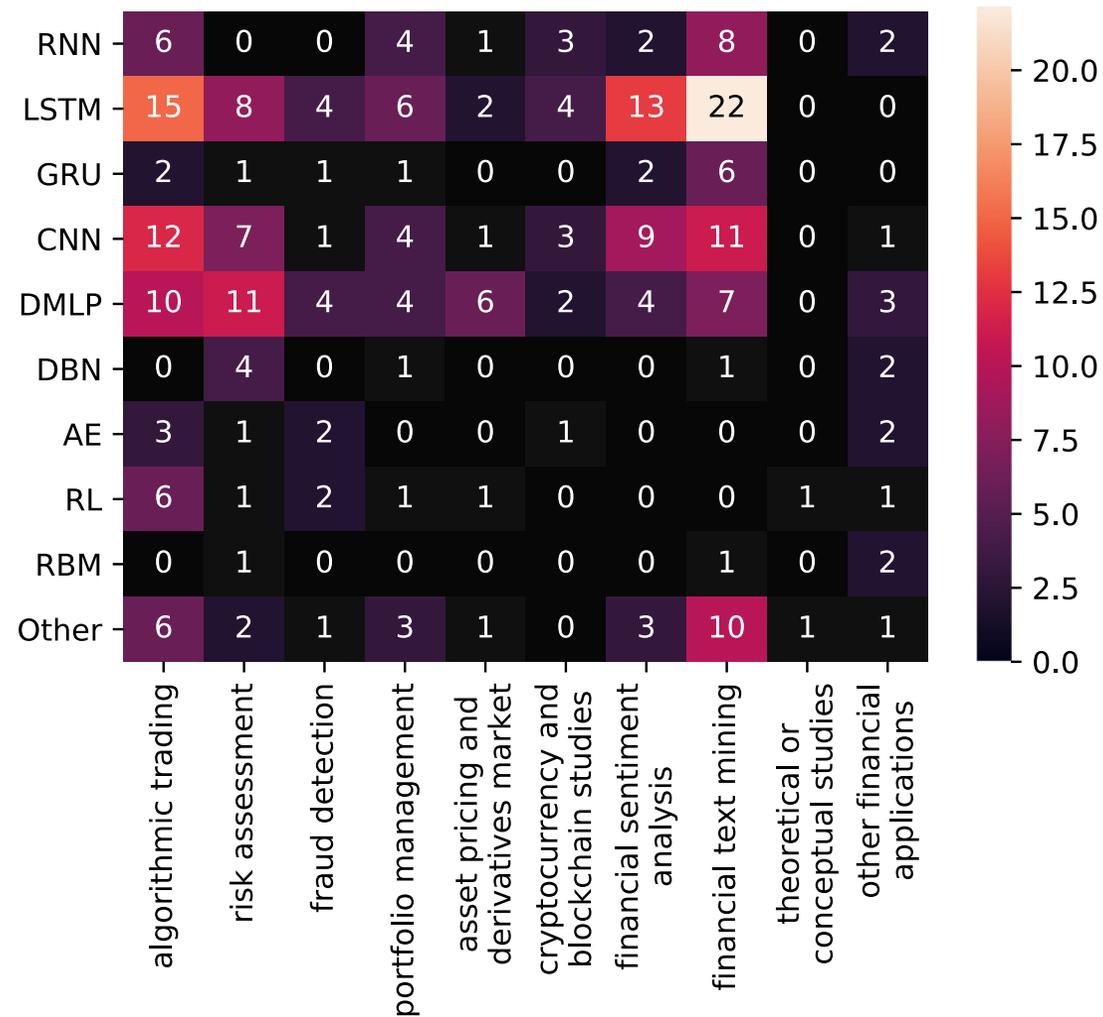
# AI 2.0

**a new generation of AI  
based on the  
novel information environment of  
major changes and  
the development of  
new goals.**

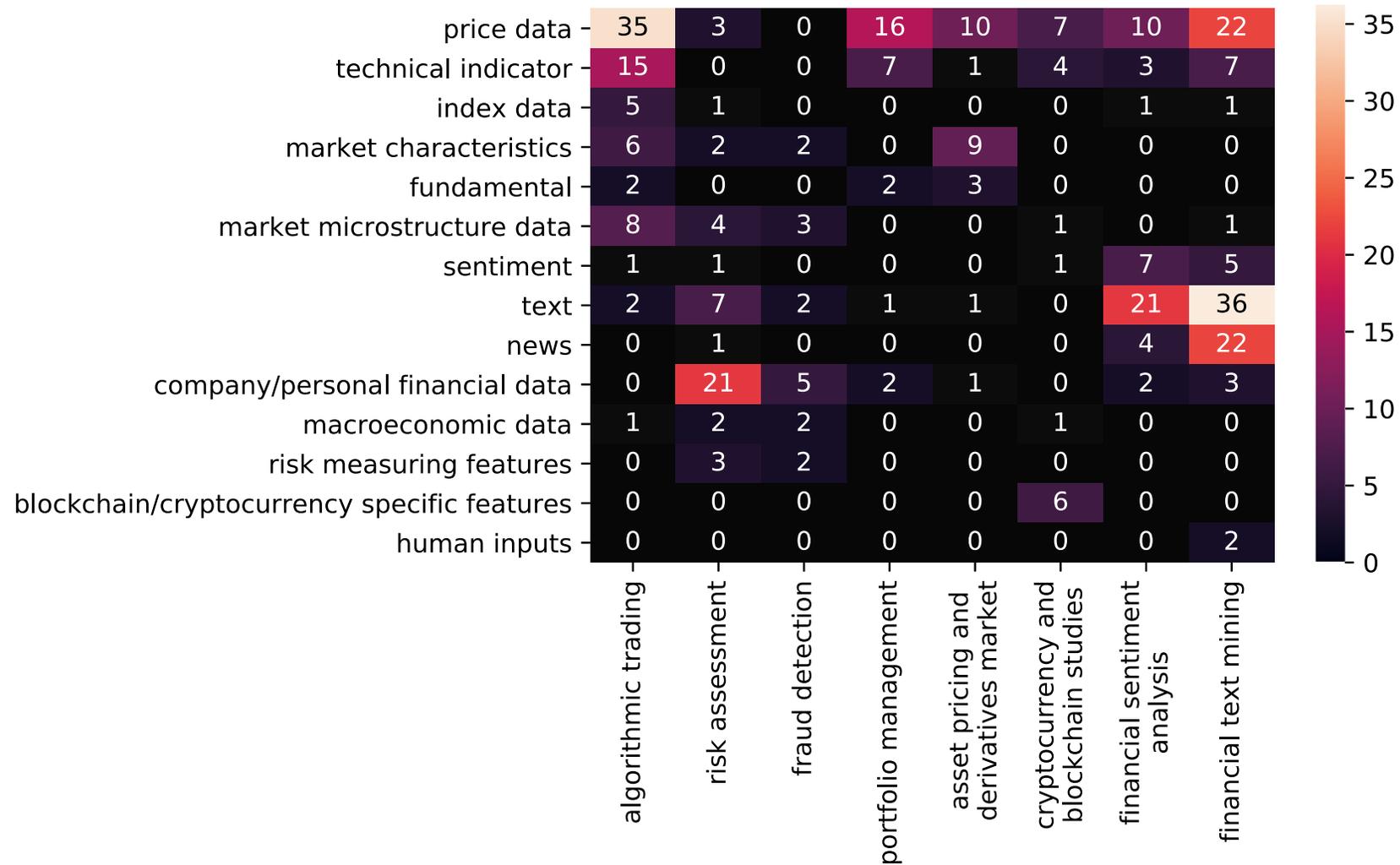
# Technology-driven Financial Industry Development

Development stage	Driving technology	Main landscape	Inclusive finance	Relationship between technology and finance
Fintech 1.0 (financial IT)	Computer	Credit card, ATM, and CRMS	Low	Technology as a tool
Fintech 2.0 (Internet finance)	Mobile Internet	Marketplace lending, third-party payment, crowdfunding, and Internet insurance	Medium	Technology- driven change
Fintech 3.0 (financial intelligence)	AI, Big Data, Cloud Computing, Blockchain	Intelligent finance	High	Deep fusion

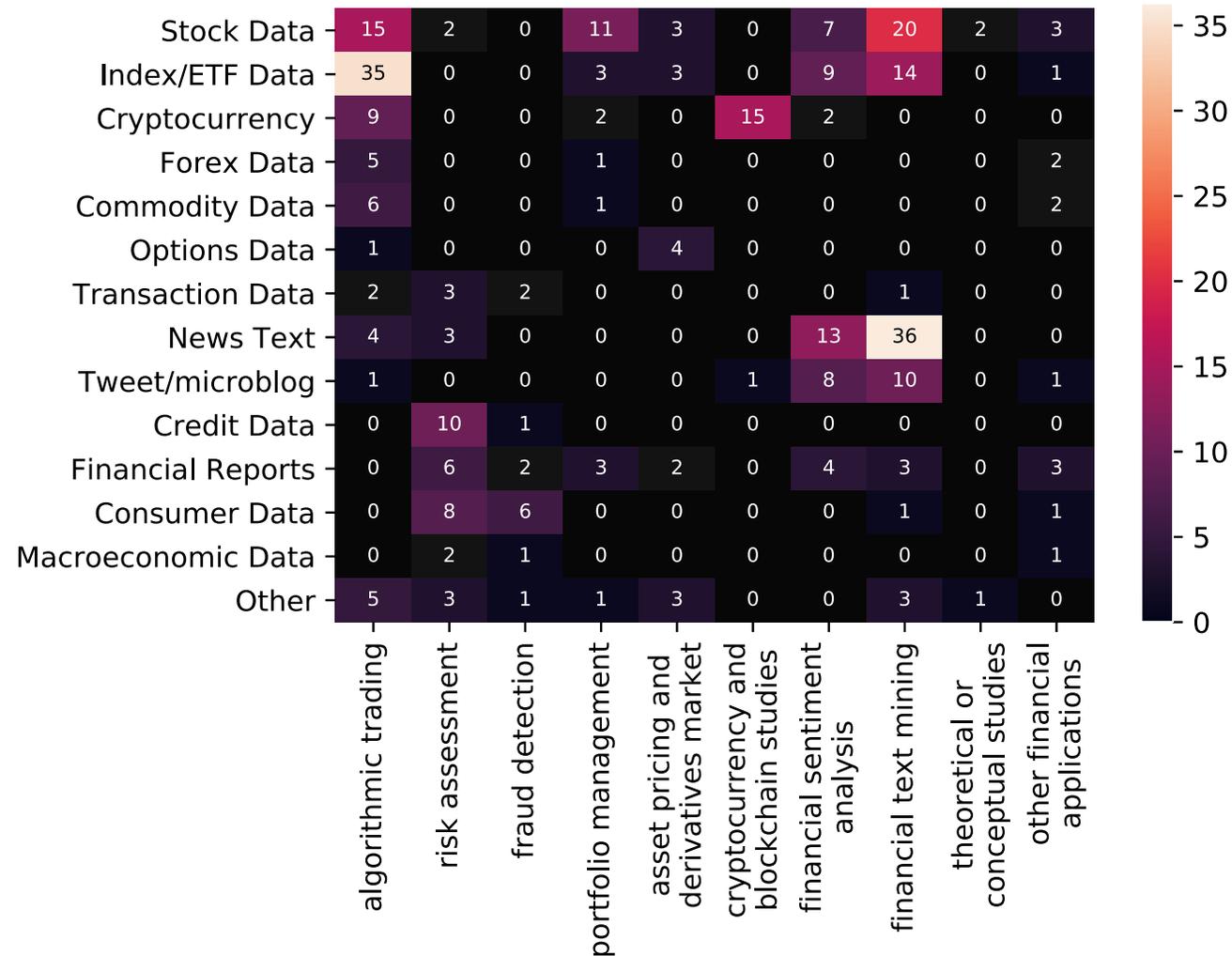
# Deep learning for financial applications: Topic-Model Heatmap



# Deep learning for financial applications: Topic-Feature Heatmap

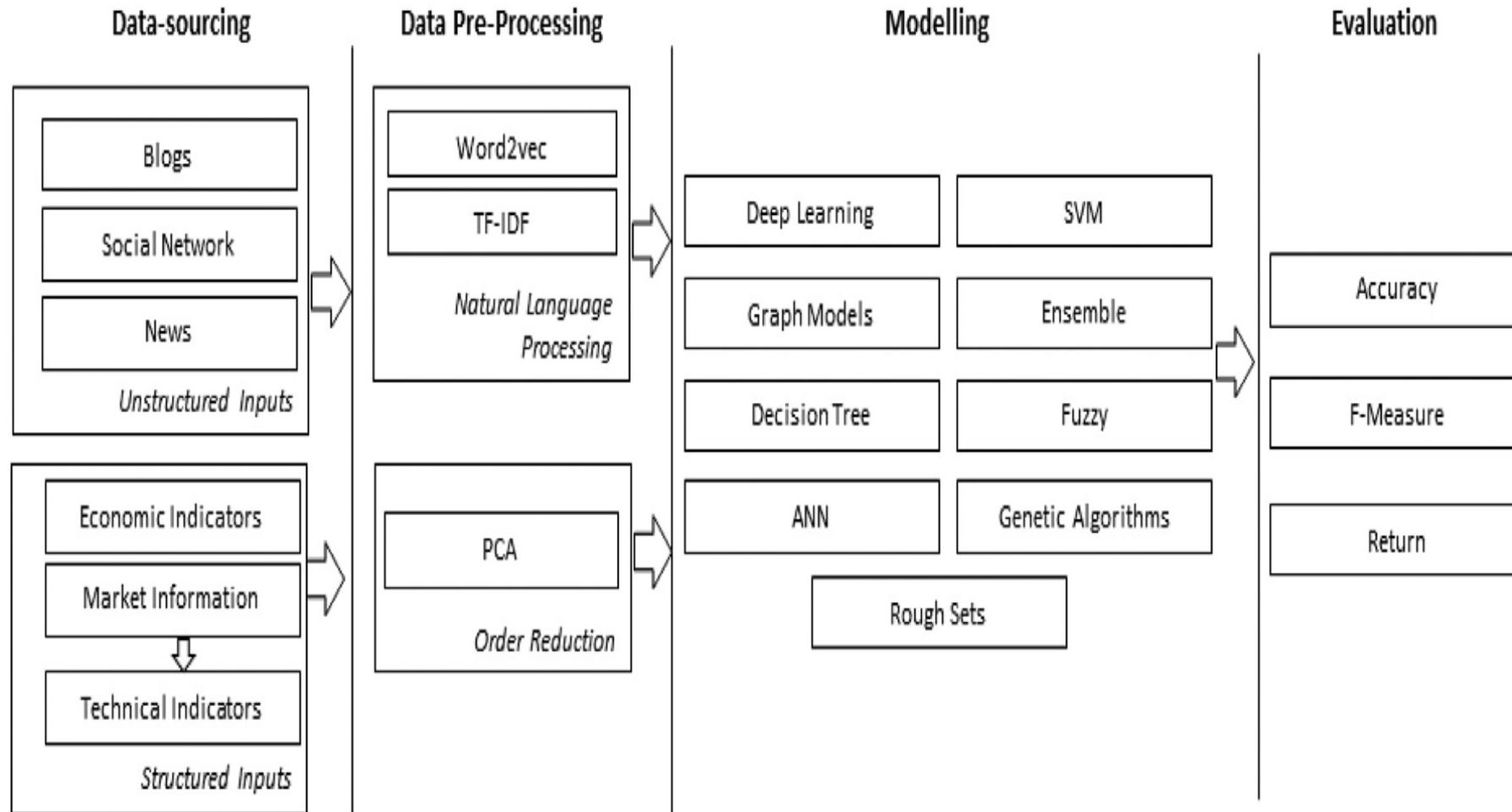


# Deep learning for financial applications: Topic-Dataset Heatmap



# Stock Market Movement Forecast:

## Phases of the stock market modeling



# **Green Finance and Sustainable Finance**

# Evolution of Sustainable Finance Research



Source: Kumar, S., Sharma, D., Rao, S., Lim, W. M., & Mangla, S. K. (2022). Past, present, and future of sustainable finance: Insights from big data analytics through machine learning of scholarly research. *Annals of Operations Research*, 1-44.

# **AI for Environmental, Social, and Governance (AI4ESG)**

# AI for Social Good (AI4SG)

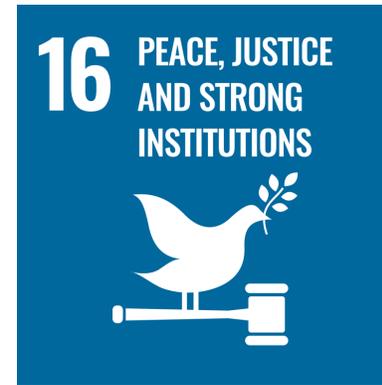
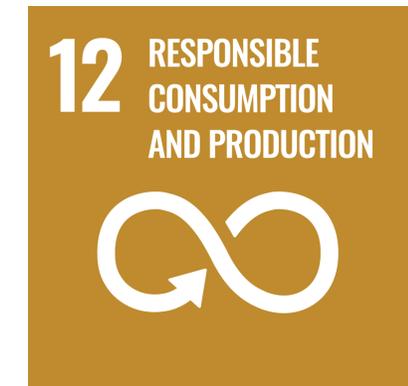
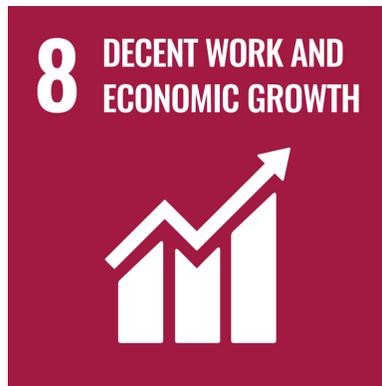
# Sustainability

SDGs

CSR

ESG

# Sustainable Development Goals (SDGs)



# Sustainable Development Goals (SDGs) and 5P

Partnership

Peace

Prosperity

People

Planet



# Green Finance

Generic term

implying use or diversion

of **financial resources**

to deploy and support projects

with **long term positive impact**

on the **environment**

# **Sustainable Finance**

## **Finances**

**deployed in support of projects  
that ensure just, sustainable and  
inclusive growth  
or attainment of one or more  
sustainable development goals**

# Carbon Finance

Financial instruments

based on

**economic value of carbon emissions**

which an organization cannot avoid but which it offsets by funding other compensatory projects that contribute to **carbon emissions reduction**

# Climate Finance

Finances deployed  
in support of low carbon and  
climate resilient projects  
that help in **climate change mitigation** and  
adaptation efforts,  
particularly in the  
**energy and infrastructure sectors**

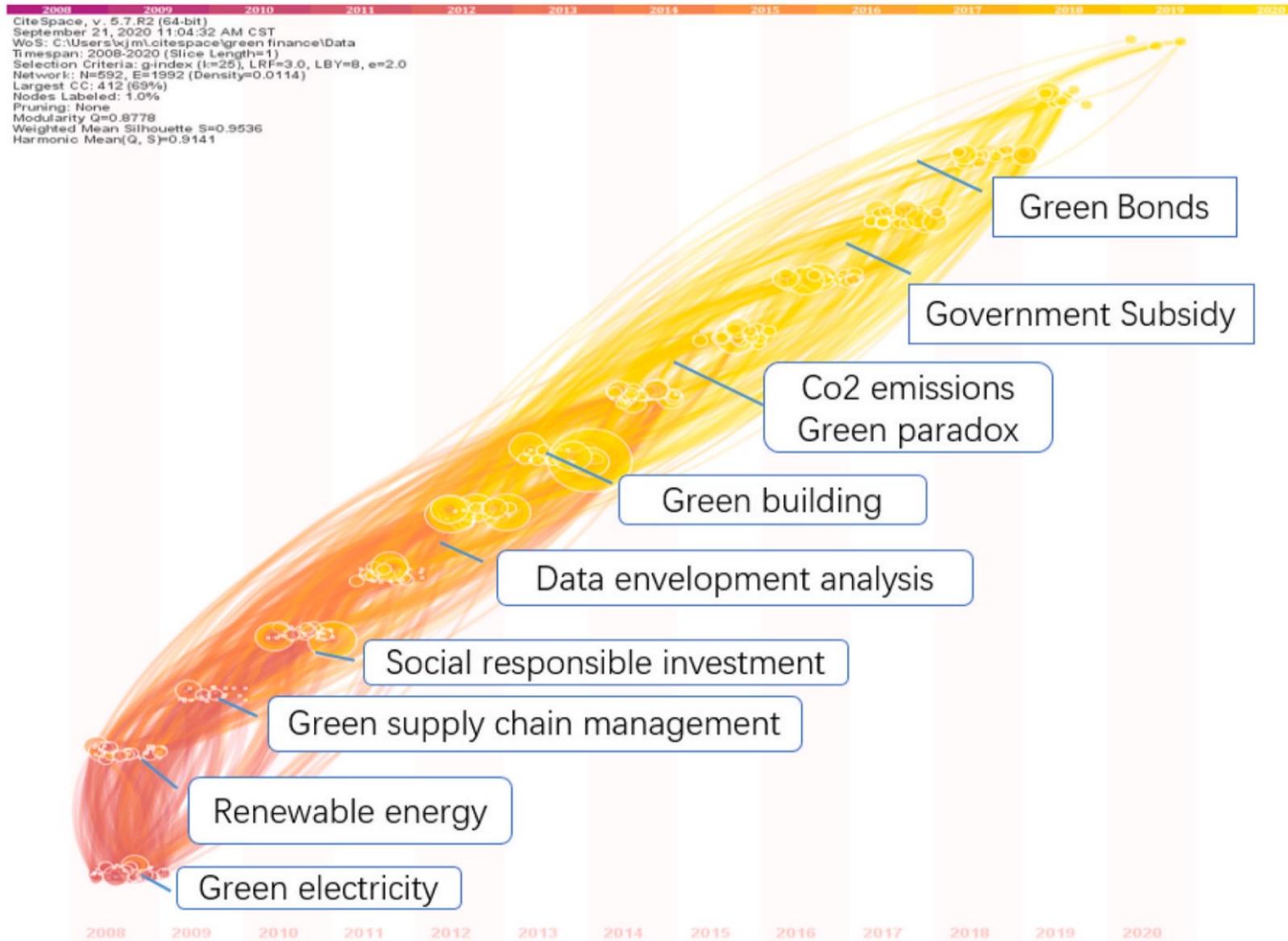
# ESG Investing

**Investments** considering the broad range of **environmental** (e.g. climate change, pollution biodiversity loss), **social** (e.g. working conditions, human rights, salary or compensation structures) and **governance** (e.g. board composition, diversity and inclusion, taxes) characteristics of the projects or companies being invested in; **ethical and business sustainability** considerations are **integral part of financing**

# Impact Investing

**Investing** in projects  
that solve a **social or environmental problem**;  
the focus is on the **positive impact**  
rather than the  
means used to produce that impact

# Dynamic Trends of Green Finance and Energy Policy



**ESG:**

**Environmental**

**Social**

**Governance**

**CSR:**  
**Corporate**  
**Social**  
**Responsibility**

# ESG to 17 SDGs

## ENVIRONMENT



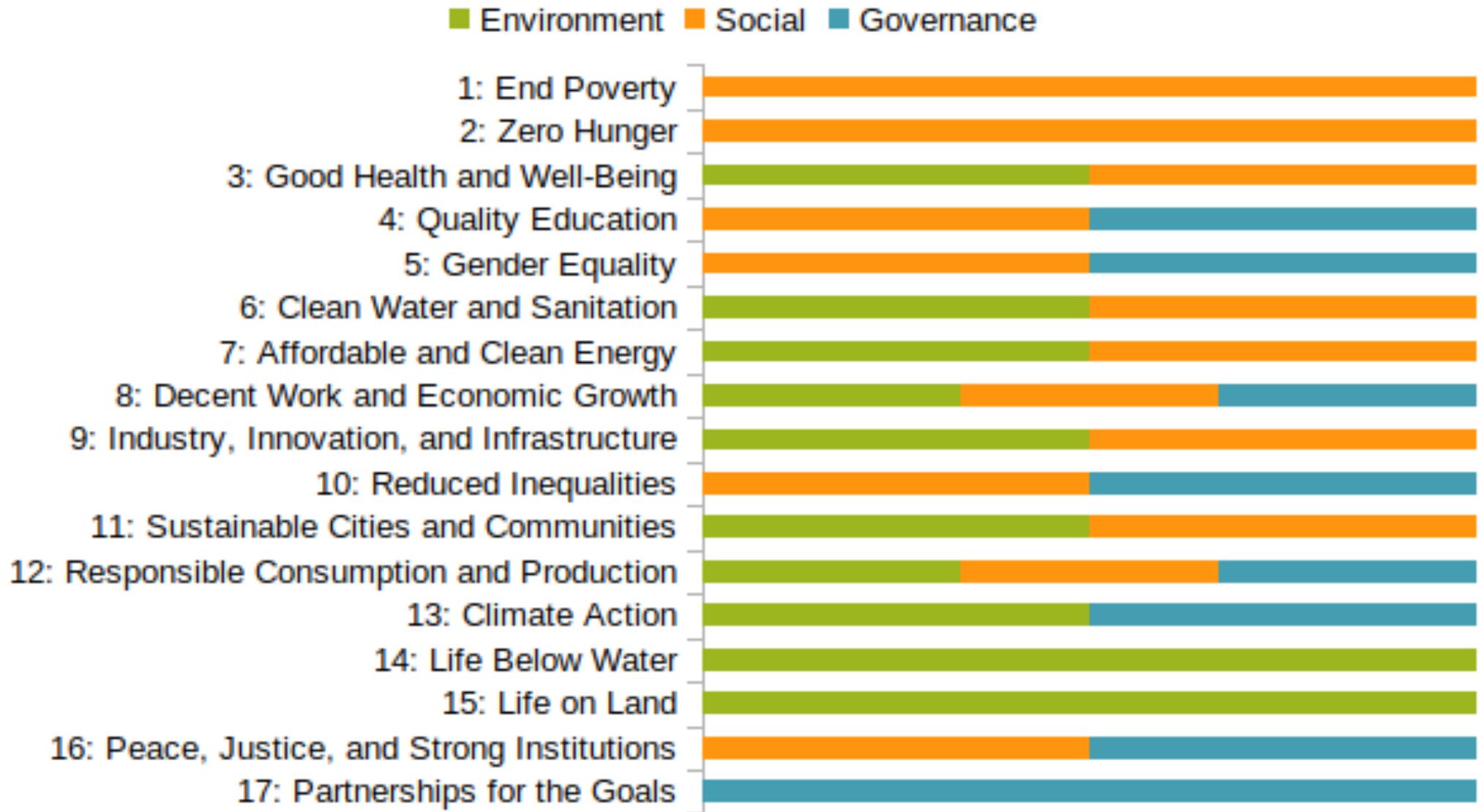
## SOCIAL



## GOVERNANCE



# ESG to 17 SDGs



# **Generative AI for ESG Applications**

# AI and Sustainability Development Goals (SDGs)

SDGs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
	No poverty	Zero hunger	Good health and well-being	Quality education	Gender equality	Clean water and sanitation	Affordable and clean energy	Decent work and economic growth	Industry, innovation and infrastructure	Reduces inequalities	Sustainable cities and communities	Responsible consumption and production	Climate action	Life below water	Life on land	Peace, justice and strong institutions	Partnerships for the goals
<b>Economic</b>								●	●	●	○						●
<b>Ecological</b>		○					○				○	○	●	●	●		
<b>Social</b>	●	●	●	●	●	●	●				●	●				●	
<b>Positive impact of AI*</b>	100%	76%	69%	10%0	56%	100%	100%	92%	100%	90%	100%	82%	80%	90%	100%	58%	26%

Note: ● adopted from Vinuesa et al. (2020), ○ added based on our analysis.

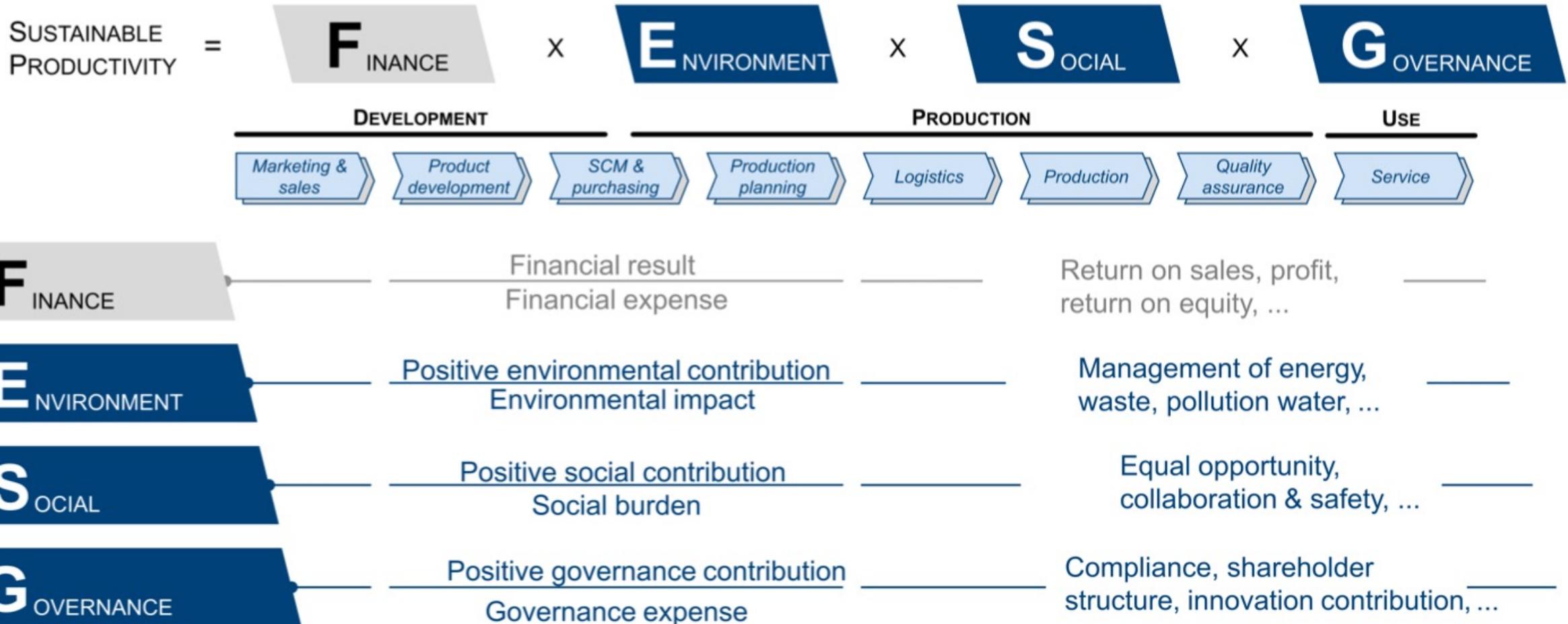
\*The assessment of AI's possible positive impact is based on a consensus-based expert elicitation process (Vinuesa et al., 2020).

# AI for Sustainability

Dimension	Code characteristics					
Primary objective <sup>1</sup>	Develop new (AI) methods (11/95)	Compare (AI) methods (39/95)	Apply (AI) methods (53/95)	Develop new system (20/95)	Other objective (4/95)	
Sustainability dimension	Economic (23/95)		Ecological (17/95)		Social (72/95)	
Sustainable Development Goals (SDGs)	SDG 1 (0/95)	SDG 2 (2/95)	SDG 3 (55/95)	SDG 4 (6/95)	SDG 5 (0/95)	SDG 6 (0/95)
	SDG 7 (9/95)	SDG 8 (7/95)	SDG 9 (8/95)	SDG 10 (1/95)	SDG 11 (9/95)	SDG 12 (8/95)
	SDG 13 (2/95)	SDG 14 (0/95)	SDG 15 (2/95)	SDG 16 (11/95)	SDG 17 (0/95)	
Data source	Reviews (12/95)	Social media/ Online forums (31/95)	Health records (21/95)	Environment/ Weather (10/95)	Energy (5/95)	
Data source plurality	Single source (50/95)		Multiple sources (44/95)		N/A (1/95)	
Data sensitivity	Publicly available data (64/95)	Internal data (16/95)		Other (11/95)	N/A (9/95)	
Manual labeling	Yes (32/95)			No (63/95)		
Technology	ML (91/95)	NLP (42/95)		CV (12/95)	Other (21/95)	
Type of learning for ML approach	Supervised learning (85/95)			Unsupervised learning (23/95)		
Neural vs. non-neural	Non-neural (45/95)		Neural (50/95)		Deep learning (38/95)	
Evaluation	Technical evaluation (83/95)			Domain evaluation (25/95)		
Paradigm	DSR/ADR (30/95)			Non-DSR/ADR (64/95)		
			0-9	10-29	30-54	55-69
						70-95
Notes: Code dimensions are not mutually exclusive; one article can be classified into one or more code characteristics; <sup>1</sup> 'Compare' does include 'apply'.						

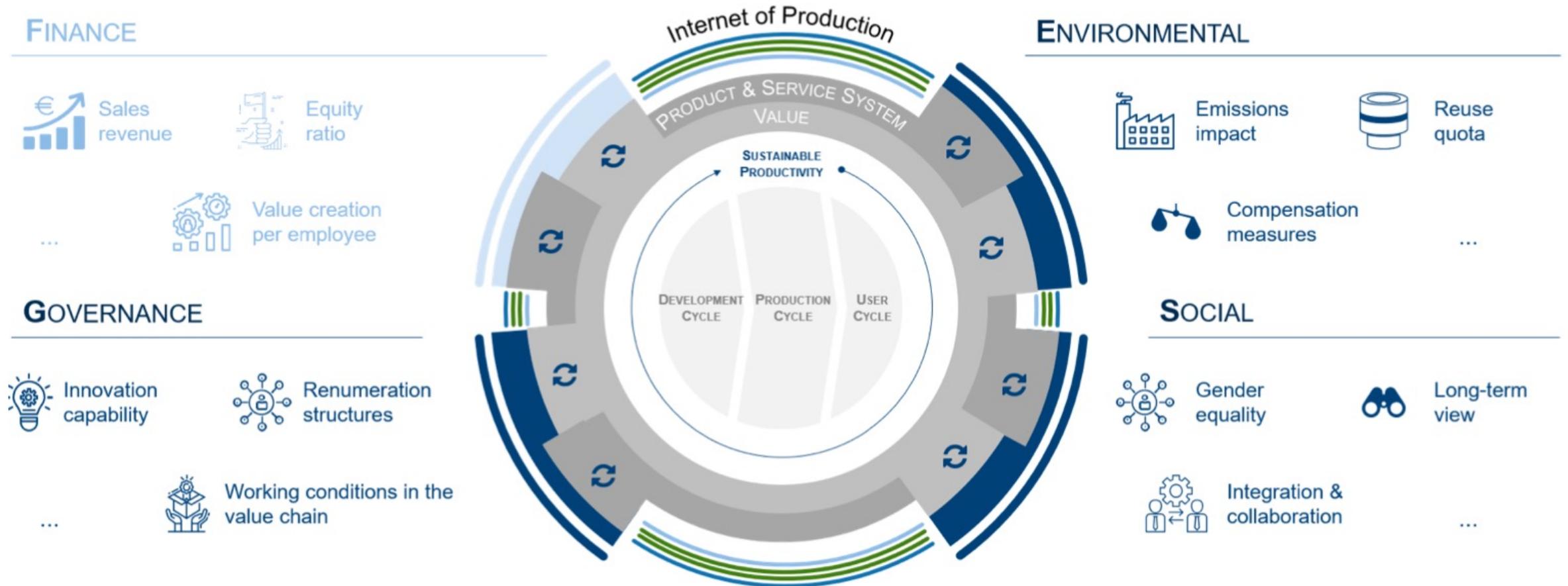
Source: Schoormann, T., Strobel, G., Möller, F., Petrik, D., & Zschech, P. (2023).

# Sustainable Productivity: Finance ESG



# Sustainable Resilient Manufacturing

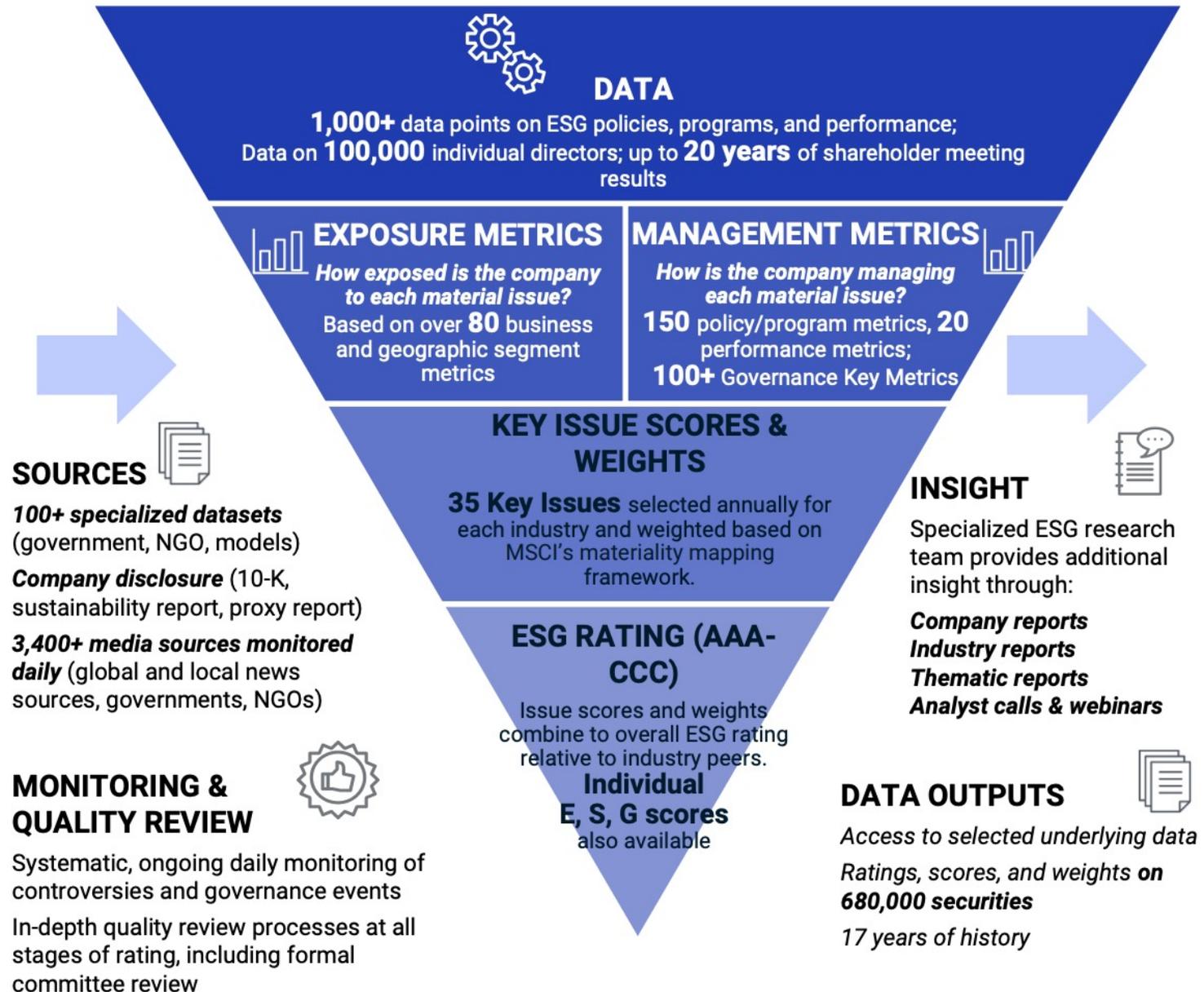
## ESG



# ESG Indexes

- **MSCI ESG Index**
- **Dow Jones Sustainability Indices (DJSI)**
- **FTSE ESG Index**

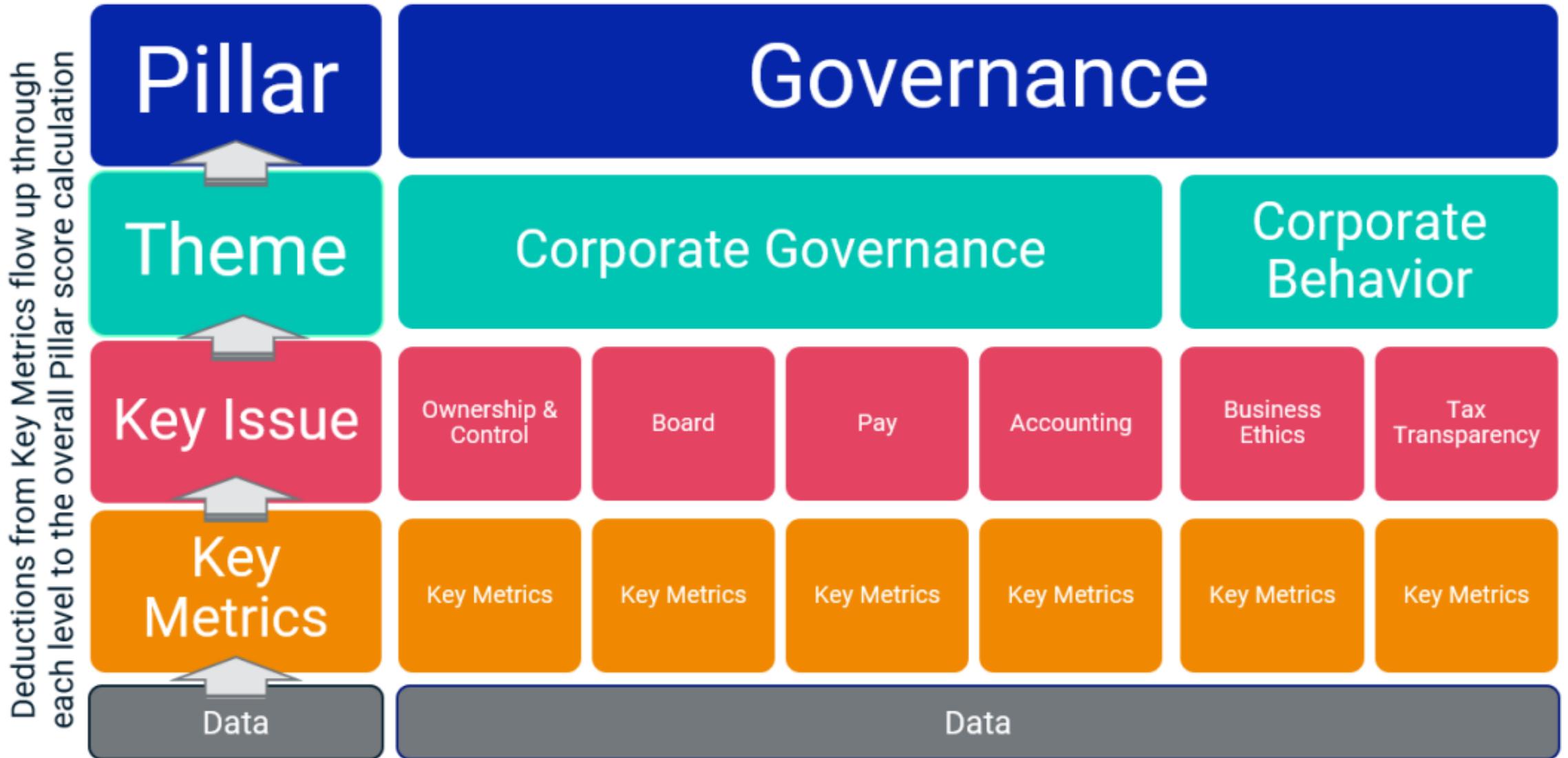
# MSCI ESG Rating Framework



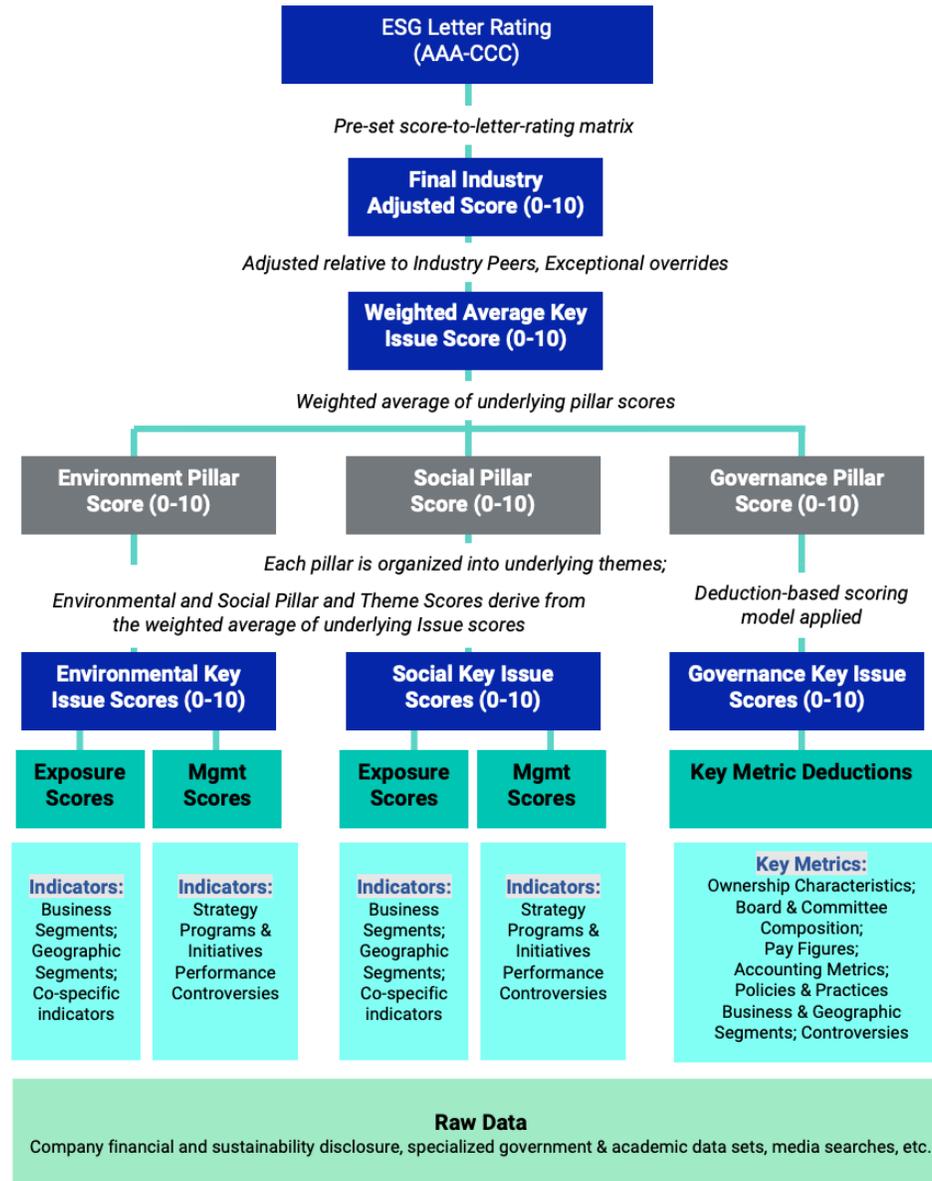
# MSCI ESG Key Issue Hierarchy

3 Pillars	10 Themes	35 ESG Key Issues	
<b>Environment</b>	<b>Climate Change</b>	Carbon Emissions Product Carbon Footprint	Financing Environmental Impact Climate Change Vulnerability
	<b>Natural Capital</b>	Water Stress Biodiversity & Land Use	Raw Material Sourcing
	<b>Pollution &amp; Waste</b>	Toxic Emissions & Waste Packaging Material & Waste	Electronic Waste
	<b>Environmental Opportunities</b>	Opportunities in Clean Tech Opportunities in Green Building	Opportunities in Renewable Energy
<b>Social</b>	<b>Human Capital</b>	Labor Management Health & Safety	Human Capital Development Supply Chain Labor Standards
	<b>Product Liability</b>	Product Safety & Quality Chemical Safety Consumer Financial Protection	Privacy & Data Security Responsible Investment Health & Demographic Risk
	<b>Stakeholder Opposition</b>	Controversial Sourcing Community Relations	
	<b>Social Opportunities</b>	Access to Communications Access to Finance	Access to Health Care Opportunities in Nutrition & Health
<b>Governance</b>	<b>Corporate Governance</b>	Ownership & Control Board	Pay Accounting
	<b>Corporate Behavior</b>	Business Ethics Tax Transparency	

# MSCI Governance Model Structure



# MSCI Hierarchy of ESG Scores



# DJSI S&P Global ESG Score

**8,000**  
Companies

**90%**  
Global market capitalization

**340,000+**  
Current Research Universe and Active Securities



Approx.  
**1,000**  
Datapoints

**Assessed values, text, checkboxes, documents**  
Sources: Web-based questionnaire and company documents

**130+**  
Questions

**Weighted data point scores**  
Up to 50% industry-specific

Ave.  
**30+**  
Criteria scores

**Weighted question scores**  
61 industry specific approaches, with tailored questions, criteria and related weightings

**3**  
Dimension scores

**Weighted criteria scores**  
Adjusted for corporate ESG controversies where applicable

**1**  
S&P Global ESG Score

**Sum of weighted dimension scores**

# FTSE Russell ESG Ratings



# Sustainalytics

## ESG Risk Ratings

Analyst-based  
approach

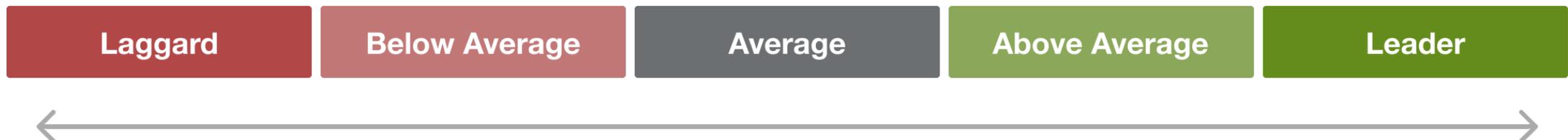
Sustainalytics' ESG Risk Ratings measure a company's exposure to industry-specific material ESG risks and how well a company is managing those risks.



# Truvalue ESG Ranks

Machine-based  
approach

- **Truvalue Labs** applies **AI** to analyze over **100,000 sources** and uncover **ESG risks** and opportunities hidden in **unstructured text**.
- The ESG Ranks data service produces an overall company rank based on industry percentile leveraging the **26 ESG categories** defined by the **Sustainability Accounting Standards Board (SASB)**.
- The data feed covers **20,000+** companies with more than **13 years** of history.



# Analyst-driven vs. AI-driven ESG

## Analyst-driven ESG research

Derives ratings in a structured data model

## Sustainalytics



*Analyst role at the end of the process allows subjectivity to color results*

## AI-driven ESG research

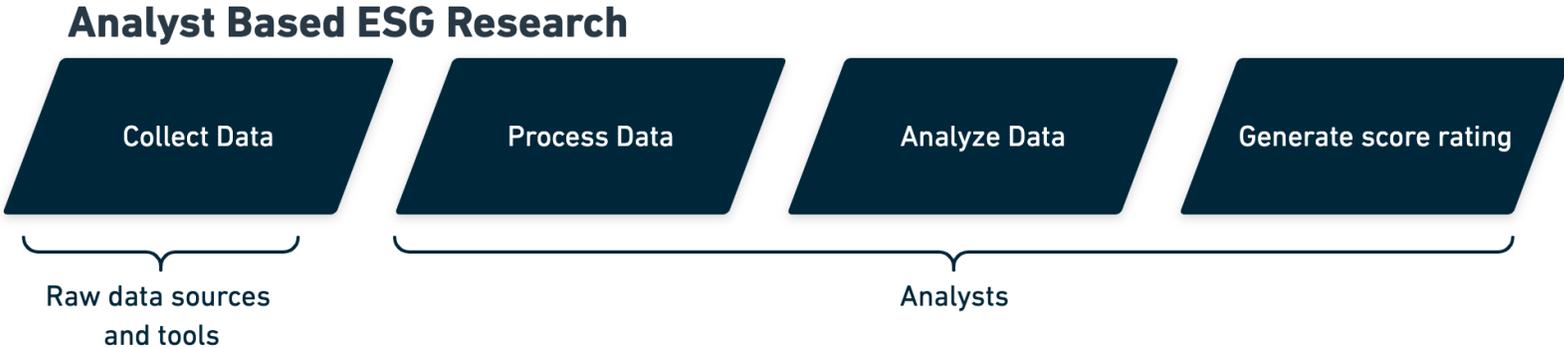
Derives signals from unstructured data

## Truvalue Labs

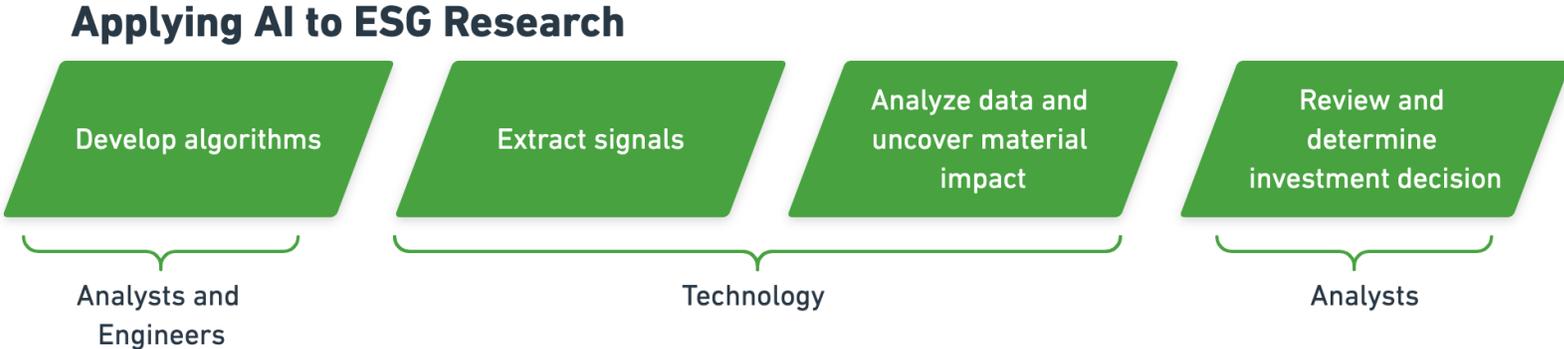


*Analyst expertise at the beginning of the process produces consistent results*

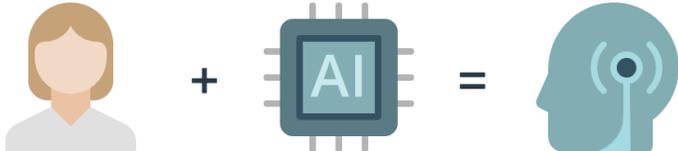
# Analyst based ESG Research



# AI based ESG Research

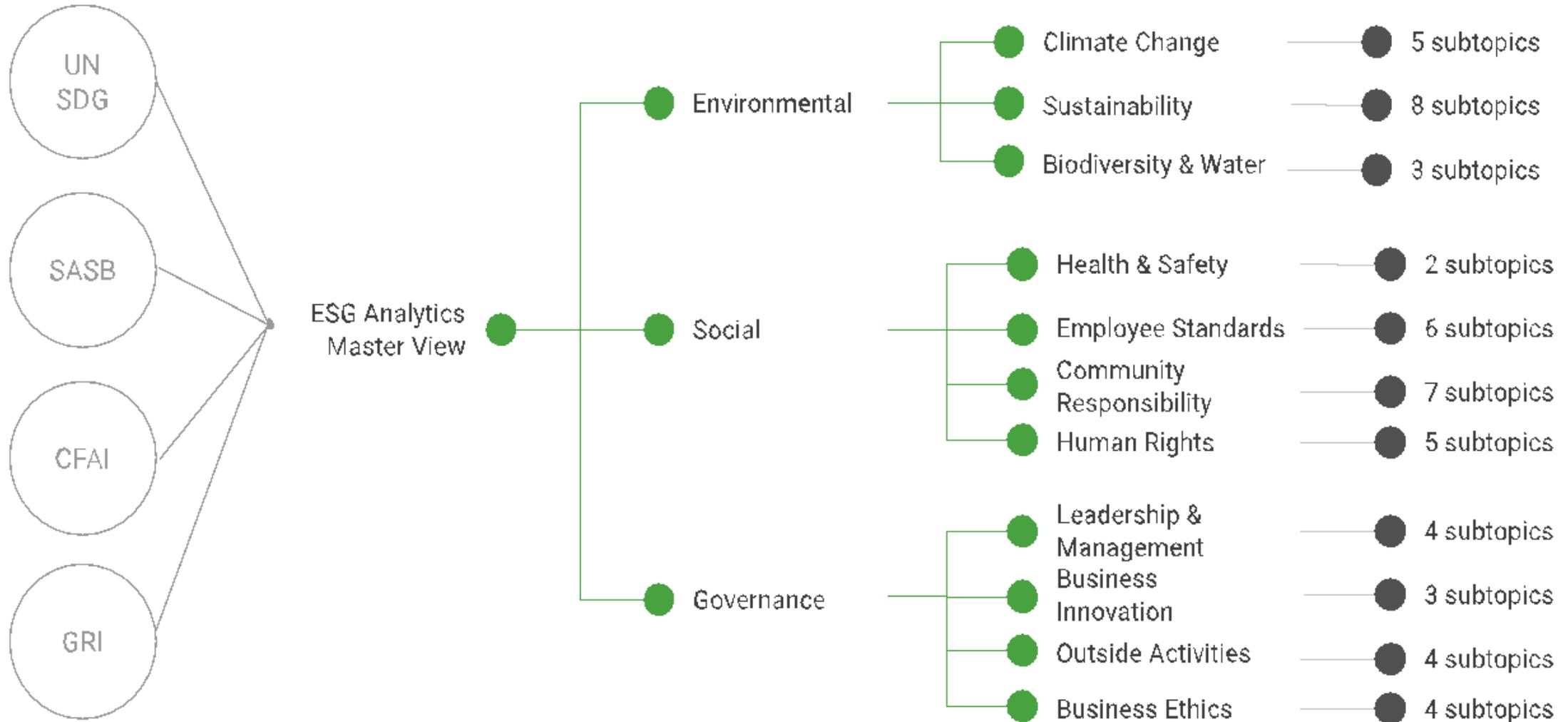


**It would take an analyst over 5 years to do what our AI can in 1 week**  
Combining analysts with AI creates gives you the full picture



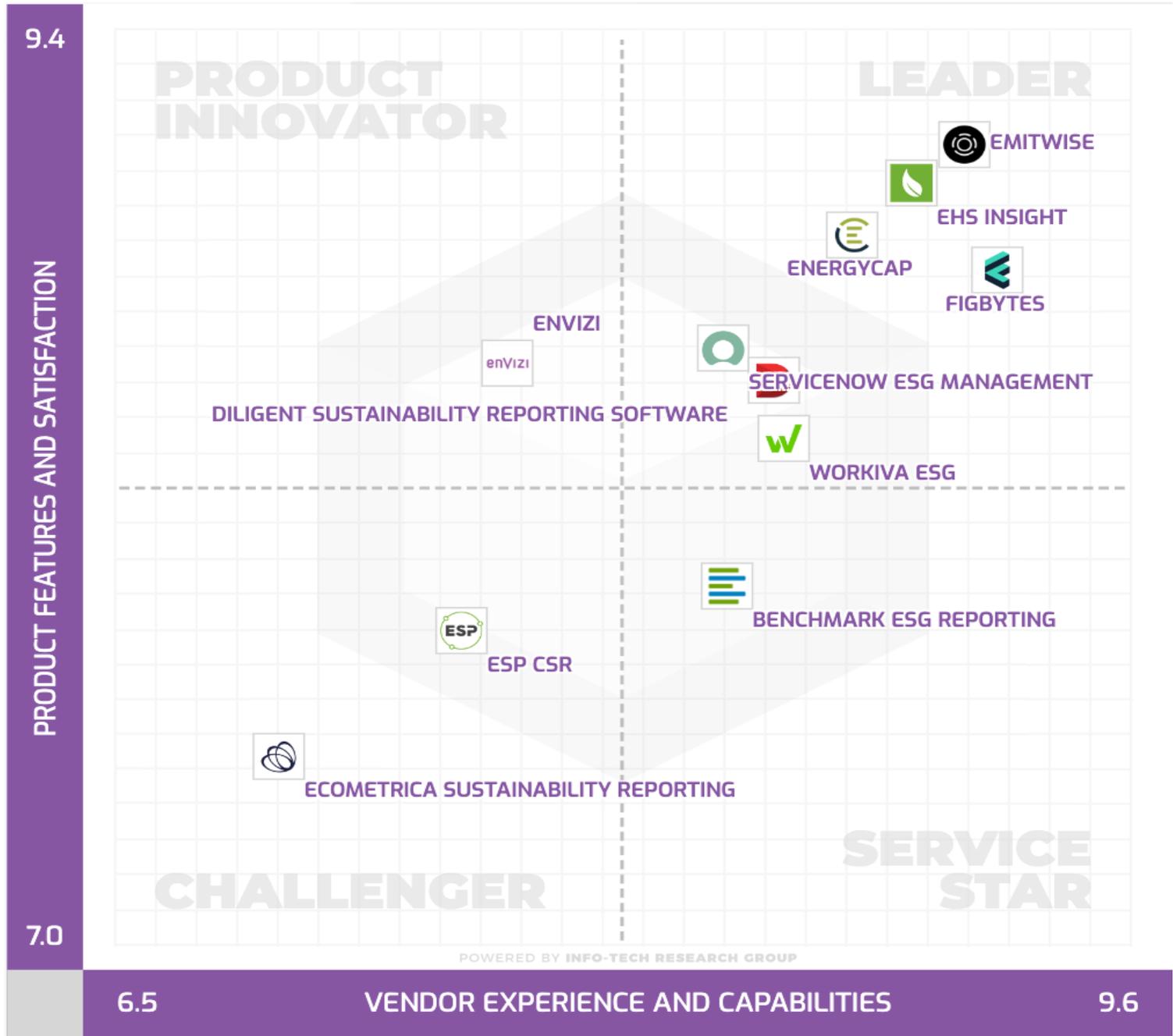
**ESG ANALYTICS**  
Invest where it matters.

# ESG Analytics: NLP Taxonomy



# Top ESG Reporting Software

**Environmental, Social and Governance (ESG) Reporting software or Sustainability software helps organizations manage their operational data, evaluate their impact on the environment and provide reporting to perform audits.**



# ESG Reporting Software: Emitwise

- Emitwise is the carbon management platform for companies with complex manufacturing supply chains to confidently understand, track and reduce their complete carbon footprint.
- Combining 100 years of carbon accounting experience and machine learning technology, we accelerate climate action by increasing the accuracy of scope 3 emissions.
- The platform empowers manufacturers and their supply chains to make carbon-led business decisions that lower risk, increase profitability and deliver ambitious climate action.

**9.2**

COMPOSITE  
SCORE

**9.3**

CX SCORE

**+99**

EMOTIONAL  
FOOTPRINT

**94%**

LIKELINESS TO  
RECOMMEND

# ESG Reporting Software: Workiva ESG

- Workiva is a cloud native platform that simplifies the complexities of reporting and compliance.
- Workiva ESG is the end-to-end platform that allows you to integrate financial data, nonfinancial data, and XBRL.
- Workiva, the platform that streamlines your entire ESG process.
- Automate data collection, utilize frameworks, and directly connect to all your ESG reports. in meaningful glossy reports, accurate survey responses, and regulatory filings with integrated XBRL tagging.

**8.4**

COMPOSITE  
SCORE

**8.7**

CX SCORE

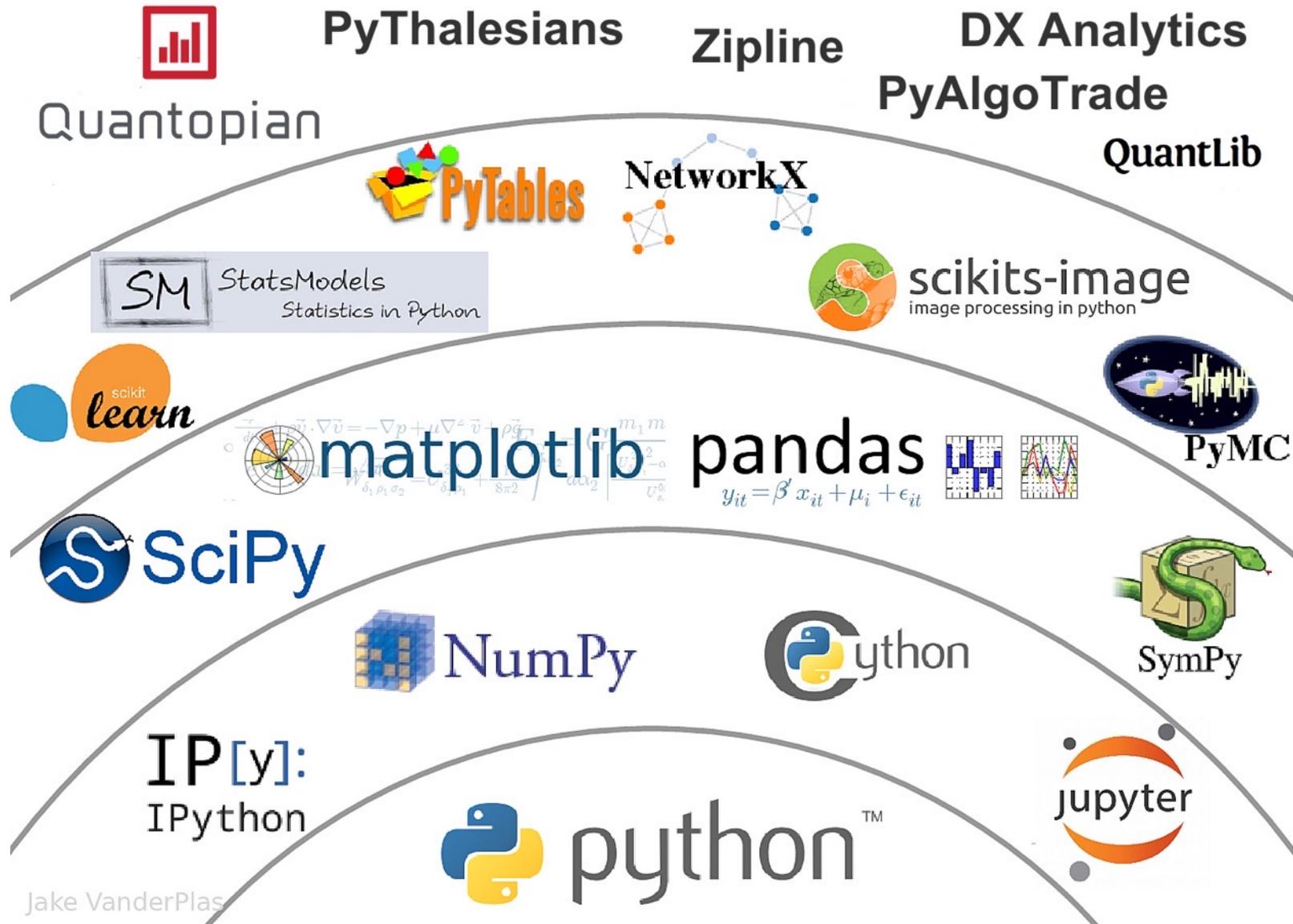
**+92**

EMOTIONAL  
FOOTPRINT

**89%**

LIKELINESS TO  
RECOMMEND

# The Quant Finance PyData Stack



Jake VanderPlas

Source: [http://nbviewer.jupyter.org/format/slides/github/quantopian/pyfolio/blob/master/pyfolio/examples/overview\\_slides.ipynb#/5](http://nbviewer.jupyter.org/format/slides/github/quantopian/pyfolio/blob/master/pyfolio/examples/overview_slides.ipynb#/5)

# Yves Hilpisch (2020), **Artificial Intelligence in Finance: A Python-Based Guide**, O'Reilly

yhilpisch / aiif Public <https://github.com/yhilpisch/aiif> Notifications Star 98 Fork 77

Code Issues Pull requests Actions Projects Wiki Security Insights

main 1 branch 0 tags Go to file Code

yves Code updates for TF 2.3. e334251 on Dec 8, 2020 4 commits

code	Code updates for TF 2.3.	11 months ago
.gitignore	Code updates for TF 2.3.	11 months ago
LICENSE.txt	Code updates.	11 months ago
README.md	Code updates.	11 months ago

README.md

## Artificial Intelligence in Finance

### About this Repository

This repository provides Python code and Jupyter Notebooks accompanying the **Artificial Intelligence in Finance** book published by [O'Reilly](#).



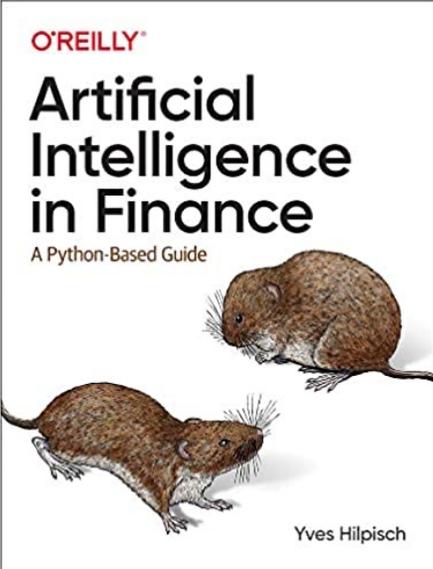
**About**  
Jupyter Notebooks and code for the book **Artificial Intelligence in Finance** (O'Reilly) by Yves Hilpisch.  
[home.tpq.io/books/aiif](http://home.tpq.io/books/aiif)  
Readme  
View license

**Releases**  
No releases published

**Packages**  
No packages published

**Languages**

Jupyter Notebook	97.4%
Python	2.6%



# Yves Hilpisch (2020), **Artificial Intelligence in Finance: A Python-Based Guide**, O'Reilly

yhilpisch / aiif Public

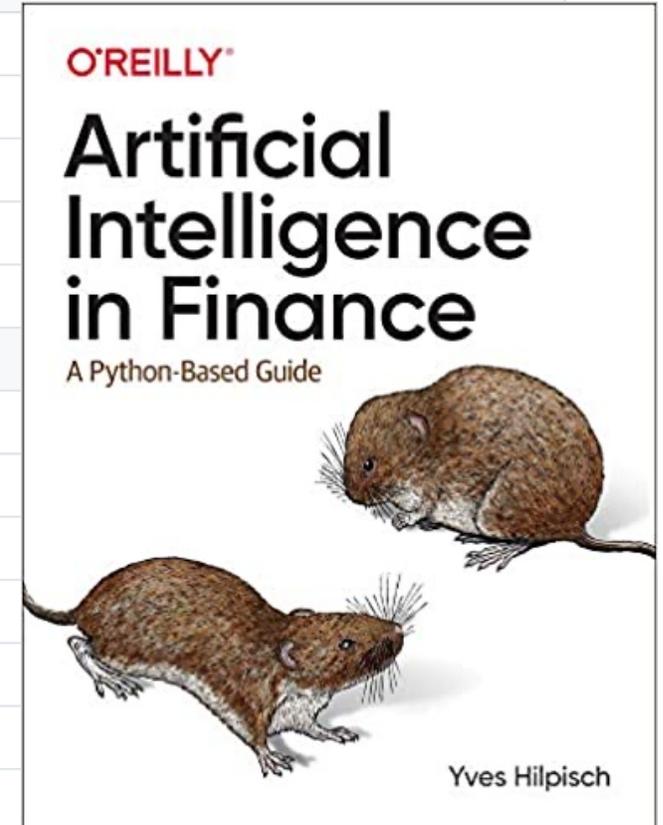
Notifications Star 98 Fork 77

Code Issues Pull requests Actions Projects Wiki Security Insights

main aiif / code / <https://github.com/yhilpisch/aiif/tree/main/code> Go to file

yves Code updates for TF 2.3. e334251 on Dec 8, 2020 History

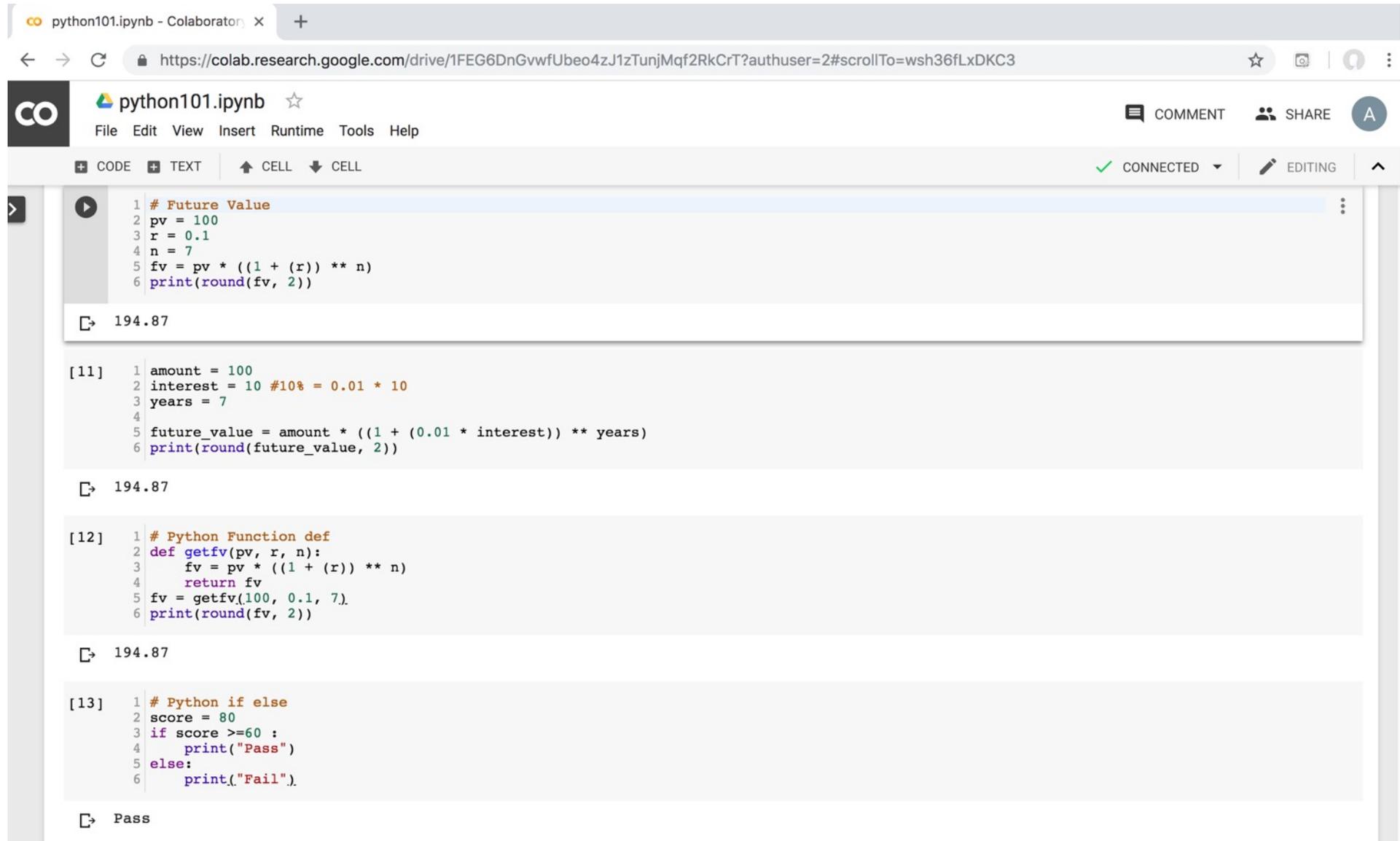
..	
oanda	Code updates for TF 2.3.
01_artificial_intelligence.ipynb	Code updates for TF 2.3.
02_superintelligence.ipynb	Code updates for TF 2.3.
03_normative_finance.ipynb	Code updates for TF 2.3.
04_data_driven_finance_a.ipynb	Initial commit.
04_data_driven_finance_b.ipynb	Initial commit.
05_machine_learning.ipynb	Code updates for TF 2.3.
06_ai_first_finance.ipynb	Code updates for TF 2.3.
07_dense_networks.ipynb	Code updates for TF 2.3.
08_recurrent_networks.ipynb	Code updates for TF 2.3.
09_reinforcement_learning_a.ipynb	Code updates.
09_reinforcement_learning_b.ipynb	Code updates for TF 2.3.



Source: <https://github.com/yhilpisch/aiif/tree/main/code>

# Python in Google Colab (Python101)

<https://colab.research.google.com/drive/1FEG6DnGvwfUbeo4zJ1zTunjMqf2RkCrT>



The screenshot shows a Google Colab notebook titled "python101.ipynb". The interface includes a browser address bar with the URL <https://colab.research.google.com/drive/1FEG6DnGvwfUbeo4zJ1zTunjMqf2RkCrT?authuser=2#scrollTo=wsh36fLxDKC3>. The notebook has a menu bar with "File", "Edit", "View", "Insert", "Runtime", "Tools", and "Help". The top right shows "COMMENT", "SHARE", and a user profile icon. The notebook is in "EDITING" mode and is "CONNECTED".

The notebook contains four code cells:

- Cell 1:** A code cell with the following Python code:

```
1 # Future Value
2 pv = 100
3 r = 0.1
4 n = 7
5 fv = pv * ((1 + (r)) ** n)
6 print(round(fv, 2))
```

The output is "194.87".
- Cell [11]:** A code cell with the following Python code:

```
1 amount = 100
2 interest = 10 #10% = 0.01 * 10
3 years = 7
4
5 future_value = amount * ((1 + (0.01 * interest)) ** years)
6 print(round(future_value, 2))
```

The output is "194.87".
- Cell [12]:** A code cell with the following Python code:

```
1 # Python Function def
2 def getfv(pv, r, n):
3     fv = pv * ((1 + (r)) ** n)
4     return fv
5 fv = getfv(100, 0.1, 7)
6 print(round(fv, 2))
```

The output is "194.87".
- Cell [13]:** A code cell with the following Python code:

```
1 # Python if else
2 score = 80
3 if score >=60 :
4     print("Pass")
5 else:
6     print("Fail").
```

The output is "Pass".

<https://tinyurl.com/aintpupython101>

# Python in Google Colab (Python101)

<https://colab.research.google.com/drive/1FEG6DnGvwfUbeo4zJ1zTunjMqf2RkCrT>

The screenshot shows a Google Colab notebook titled "python101.ipynb". The interface includes a top navigation bar with "File", "Edit", "View", "Insert", "Runtime", "Tools", and "Help" menus, along with "Comment", "Share", and "Settings" icons. A "Table of contents" sidebar on the left lists various topics, with "Uncertainty and Risk" currently selected. The main content area displays a table of contents with expandable sections: "AI in Finance", "Normative Finance and Financial Theories", and "Uncertainty and Risk". Below the table of contents, a code cell is visible, containing Python code that imports numpy and defines variables for stock and bond prices and market price vectors.

python101.ipynb ☆

File Edit View Insert Runtime Tools Help [All changes saved](#)

Comment Share Settings A

RAM Disk Editing

Table of contents

- AI in Finance
  - Normative Finance and Financial Theories
    - Uncertainty and Risk**
    - Expected Utility Theory (EUT)
    - Mean-Variance Portfolio Theory (MVPT)
    - Capital Asset Pricing Model (CAPM)
    - Arbitrage Pricing Theory (APT)
  - Deep Learning for Financial Time Series Forecasting
  - Portfolio Optimization and Algorithmic Trading
    - Investment Portfolio Optimisation with Python
    - Efficient Frontier Portfolio Optimisation in Python
    - Investment Portfolio Optimization

AI in Finance

- Source: Yves Hilpisch (2020), Artificial Intelligence in Finance: A Python-Based Guide, O'Reilly Media.
- Github: <https://github.com/yhilpisch/aiif/>

Normative Finance and Financial Theories

Uncertainty and Risk

```
1 import numpy as np
2
3 #The prices of the stock and bond today.
4 S0 = 10
5 B0 = 10
6 print('S0', S0)
7 print('B0', B0)
8
9 #The uncertain payoff of the stock and bond tomorrow.
10 S1 = np.array((20, 5))
11 B1 = np.array((11, 11))
12 print('S1', S1)
13 print('B1', B1)
14
15 #The market price vector
16 M0 = np.array((S0, B0))
```

<https://tinyurl.com/aintpupython101>

# Python in Google Colab (Python101)

python101.ipynb ☆

File Edit View Insert Runtime Tools Help [All changes saved](#)

Comment Share Settings A

RAM Disk Editing

## Table of contents

- Data Driven Finance
  - Financial Econometrics and Regression**
  - Data Availability
  - Normative Theories Revisited
    - Mean-Variance Portfolio Theory
    - Capital Asset Pricing Model
    - Arbitrage-Pricing Theory
  - Debunking Central Assumptions
  - Normality
    - Sample Data Sets
    - Real Financial Returns
  - Linear Relationships
- Deep Learning for Financial Time Series Forecasting
- Portfolio Optimization and Algorithmic Trading
  - Investment Portfolio Optimisation with Python
  - Efficient Frontier Portfolio Optimisation in Python
  - Investment Portfolio Optimization

## Data Driven Finance

### Financial Econometrics and Regression

```
[18] 1 import numpy as np
      2
      3 def f(x):
      4     return 2 + 1 / 2 * x
      5
      6 x = np.arange(-4, 5)
      7 x

array([-4, -3, -2, -1, 0, 1, 2, 3, 4])
```

```
1 y = f(x)
2 y

array([ 0.00,  0.50,  1.00,  1.50,  2.00,  2.50,  3.00,  3.50,  4.00])
```

```
1 print('x', x)
2
3 print('y', y)
4
5 beta = np.cov(x, y, ddof=0)[0, 1] / x.var()
6 print('beta', beta)
```

# Python in Google Colab (Python101)

The screenshot shows a Google Colab notebook with the following components:

- Header:** "python101.ipynb" with a star icon, and navigation links for File, Edit, View, Insert, Runtime, Tools, and Help. A status bar indicates "All changes saved".
- Table of Contents (Left Panel):** A list of topics including "Financial Econometrics and Regression", "Data Availability", "Normative Theories Revisited", "Mean-Variance Portfolio Theory", "Capital Asset Pricing Model", "Arbitrage-Pricing Theory", "Debunking Central Assumptions", "Normality", "Sample Data Sets", "Real Financial Returns", "Linear Relationships", "Financial Econometrics and Machine Learning", "Machine Learning", "Data", "Success", "Capacity", "Evaluation", "Bias & Variance", and "Cross-Validation".
- Code Editor (Center):** A code cell titled "Machine Learning" containing the following Python code:

```
1 import numpy as np
2 import pandas as pd
3 from pylab import plt, mpl
4 np.random.seed(100)
5 plt.style.use('seaborn')
6 mpl.rcParams['savefig.dpi'] = 300
7 mpl.rcParams['font.family'] = 'serif'
8
9 url = 'http://hilpisch.com/aiif_eikon_eod_data.csv'
10
11 raw = pd.read_csv(url, index_col=0, parse_dates=True)['EUR=']
12 raw.head()
```
- Runtime Output (Bottom):** The output of the code cell shows a DataFrame with the following data:

Date	
2010-01-01	1.4323
2010-01-04	1.4411
2010-01-05	1.4368
2010-01-06	1.4412
2010-01-07	1.4318

Name: EUR=, dtype: float64
- Code Editor (Bottom):** A second code cell with the code: `[2] 1 raw.tail()`



# Python in Google Colab (Python101)



python101.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment

Share



## Table of contents

### Deep Learning (DL) in Finance

Dense Neural Networks (DNN)

Baseline Prediction

Normalization

Dropout

Regularization

Bagging

Optimizers

Recurrent Neural Networks (RNN)

First Example

Second Example

Financial Price Series

Financial Return Series

Financial Features

Deep RNNs

Convolutional Neural Networks (CNN)

Reinforcement Learning (RL) in Finance

+ Code + Text

Connect

Editing

## Deep Learning (DL) in Finance

- Source: Yves Hilpisch (2020), Artificial Intelligence in Finance: A Python-Based Guide, O'Reilly Media.
- Github: <https://github.com/yhilpisch/aiif/>

## Dense Neural Networks (DNN)

```
1 import os
2 import numpy as np
3 import pandas as pd
4 from pylab import plt, mpl
5 plt.style.use('seaborn')
6 mpl.rcParams['savefig.dpi'] = 300
7 mpl.rcParams['font.family'] = 'serif'
8 pd.set_option('precision', 4)
9 np.set_printoptions(suppress=True, precision=4)
10 os.environ['PYTHONHASHSEED'] = '0'

[ ] 1 url = 'http://hilpisch.com/aiif_eikon_id_eur_usd.csv'
     2 symbol = 'EUR_USD'
     3 raw = pd.read_csv(url, index_col=0, parse_dates=True)
     4 raw.head()
```

HIGH LOW OPEN CLOSE

<https://tinyurl.com/aintpupython101>

# Python in Google Colab (Python101)



python101.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment

Share



A

## Table of contents

- Financial Features
- Deep RNNs
- Convolutional Neural Networks (CNN)
- Reinforcement Learning (RL) in Finance**
- Reinforcement Learning (RL)
  - CartPole Environment
  - Dimensionality Reduction
  - Action Rule
  - Total Reward per Episode
  - Simple Learning
  - Testing the Results
  - DNN Learning
  - Q Learning
  - Finance Environment
  - Improved Finance Environment
  - Improved Financial QL Agent

+ Code + Text

Connect

Editing

## Reinforcement Learning (RL) in Finance

- Source: Yves Hilpisch (2020), Artificial Intelligence in Finance: A Python-Based Guide, O'Reilly Media.
- Github: <https://github.com/yhilpisch/aiif/>

## Reinforcement Learning (RL)

```
1 import os
2 import math
3 import random
4 import numpy as np
5 import pandas as pd
6 from pylab import plt, mpl
7 plt.style.use('seaborn')
8 mpl.rcParams['savefig.dpi'] = 300
9 mpl.rcParams['font.family'] = 'serif'
10 np.set_printoptions(precision=4, suppress=True)
11 os.environ['PYTHONHASHSEED'] = '0'
```

## CartPole Environment

```
[ ] 1 import gym
     2
```

# Python in Google Colab (Python101)



python101.ipynb ☆

File Edit View Insert Runtime Tools Help [All changes saved](#)

Comment

Share



## Table of contents

### Algorithmic Trading

Vectorized Backtesting

Backtesting an SMA-  
Based Strategy

Backtesting a Daily DNN-  
Based Strategy

Backtesting an Intraday  
DNN-Based Strategy

Risk Management

Trading Bot

Vectorized Backtesting

Event-Based Backtesting

Assessing Risk

Backtesting Risk  
Measures

Stop Loss

Trailing Stop Loss

Take Profit

Combinations

Backtesting Cryptocurrency  
Bitcoin

+ Code + Text

RAM   
Disk

Editing

## Algorithmic Trading

- Source: Yves Hilpisch (2020), Artificial Intelligence in Finance: A Python-Based Guide, O'Reilly Media.
- Github: <https://github.com/yhilpisch/aiif/>

## Vectorized Backtesting

```
1 import os
2 import math
3 import numpy as np
4 import pandas as pd
5 from pylab import plt, mpl
6 plt.style.use('seaborn')
7 mpl.rcParams['savefig.dpi'] = 300
8 mpl.rcParams['font.family'] = 'serif'
9 pd.set_option('mode.chained_assignment', None)
10 pd.set_option('display.float_format', '{:.4f}'.format)
11 np.set_printoptions(suppress=True, precision=4)
12 os.environ['PYTHONHASHSEED'] = '0'
```

## Backtesting an SMA-Based Strategy

# Python in Google Colab (Python101)

The screenshot shows a Google Colab notebook titled "python101.ipynb". The interface includes a top navigation bar with "File", "Edit", "View", "Insert", "Runtime", "Tools", and "Help" menus, along with "Comment", "Share", and "Settings" icons. A "Table of contents" sidebar on the left lists various topics under "Algorithmic Trading", "Risk Management", and "Backtesting Cryptocurrency". The main workspace displays two code cells. The first cell, titled "Vectorized Backtesting", contains Python code for setting up the environment with imports for os, math, numpy, pandas, and matplotlib, and configuring plot styles. The second cell, titled "Backtesting an SMA-Based Strategy", contains code to fetch data from a CSV file and display its information.

python101.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share ⚙️ A

RAM  Disk  Editing ^

Table of contents

- Algorithmic Trading
  - Vectorized Backtesting**
  - Backtesting an SMA-Based Strategy
  - Backtesting a Daily DNN-Based Strategy
  - Backtesting an Intraday DNN-Based Strategy
- Risk Management
  - Trading Bot
  - Vectorized Backtesting
  - Event-Based Backtesting
  - Assessing Risk
  - Backtesting Risk Measures
    - Stop Loss
    - Trailing Stop Loss
    - Take Profit
    - Combinations
- Backtesting Cryptocurrency
  - Bitcoin

+ Code + Text

Vectorized Backtesting

```
1 import os
2 import math
3 import numpy as np
4 import pandas as pd
5 from pylab import plt, mpl
6 plt.style.use('seaborn')
7 mpl.rcParams['savefig.dpi'] = 300
8 mpl.rcParams['font.family'] = 'serif'
9 pd.set_option('mode.chained_assignment', None)
10 pd.set_option('display.float_format', '{:.4f}'.format)
11 np.set_printoptions(suppress=True, precision=4)
12 os.environ['PYTHONHASHSEED'] = '0'
```

Backtesting an SMA-Based Strategy

```
[ ] 1 url = 'http://hilpisch.com/aiif_eikon_eod_data.csv'
     2 symbol = 'EUR='
     3 data = pd.DataFrame(pd.read_csv(url, index_col=0,
     4                               parse_dates=True).dropna()[symbol])
     5 data.info()
```

# Python in Google Colab (Python101)



File Edit View Insert Runtime Tools Help [All changes saved](#)

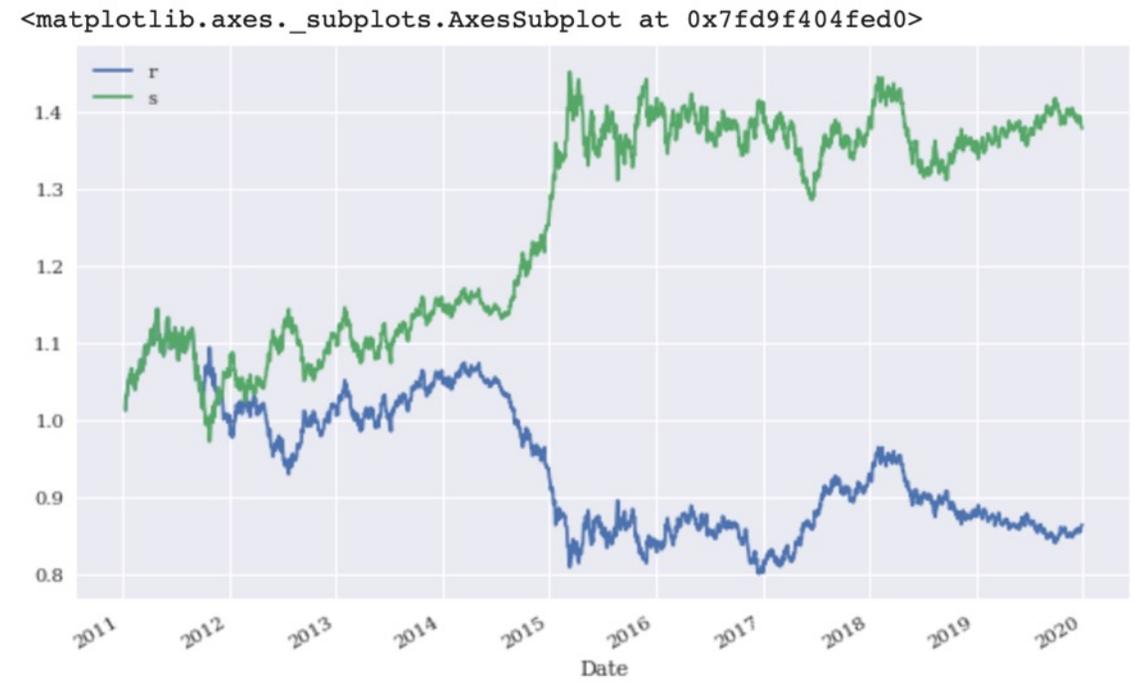
Comment Share Settings Profile

RAM Disk Editing

- Table of contents
- Algorithmic Trading
  - Vectorized Backtesting**
  - Backtesting an SMA-Based Strategy
  - Backtesting a Daily DNN-Based Strategy
  - Backtesting an Intraday DNN-Based Strategy
- Risk Management
  - Trading Bot
  - Vectorized Backtesting
  - Event-Based Backtesting
  - Assessing Risk
  - Backtesting Risk Measures
    - Stop Loss
    - Trailing Stop Loss
    - Take Profit
  - Combinations
- Backtesting Cryptocurrency
  - Bitcoin

+ Code + Text

```
[ ] 1 data['r'] = np.log(data[symbol] / data[symbol].shift(1))
     2 data.dropna(inplace=True)
     3 data['s'] = data['p'] * data['r']
     4 data[['r', 's']].sum().apply(np.exp) # gross performance
     5 data[['r', 's']].sum().apply(np.exp) - 1 # net performance
     6 data[['r', 's']].cumsum().apply(np.exp).plot(figsize=(10, 6))
```



# Python in Google Colab (Python101)

python101.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share ⚙️ A

RAM ✓ Disk

Editing ^

+ Code + Text

Table of contents

- Algorithmic Trading
  - Vectorized Backtesting
    - Backtesting an SMA-Based Strategy
    - Backtesting a Daily DNN-Based Strategy**
    - Backtesting an Intraday DNN-Based Strategy
  - Risk Management
    - Trading Bot
    - Vectorized Backtesting
    - Event-Based Backtesting
    - Assessing Risk
    - Backtesting Risk Measures
      - Stop Loss
      - Trailing Stop Loss
      - Take Profit
      - Combinations
  - Backtesting Cryptocurrency
    - Bitcoin

```
1 test['s_'] = np.where(test['p'].diff() != 0,
2                       test['s'] - pc, test['s'])
3 # test['s_'].iloc[0] -= pc
4 test['s_'].iloc[-1] -= pc
5 test[['r', 's', 's_']].sum().apply(np.exp)
6 test[['r', 's', 's_']].sum().apply(np.exp) - 1
7 test[['r', 's', 's_']].cumsum().apply(np.exp).plot(figsize=(10, 6))
```

<matplotlib.axes.\_subplots.AxesSubplot at 0x7fd901d89910>

Date	r	s	s_
2018-01	1.00	1.00	1.00
2018-04	1.03	1.08	1.07
2018-07	0.98	1.15	1.14
2018-10	0.95	1.22	1.21
2019-01	0.94	1.30	1.28
2019-04	0.94	1.25	1.24
2019-07	0.93	1.28	1.27
2019-10	0.92	1.30	1.28
2020-01	0.93	1.28	1.27

# Python in Google Colab (Python101)

The screenshot shows a Google Colab notebook titled "python101.ipynb". The interface includes a top navigation bar with "File", "Edit", "View", "Insert", "Runtime", "Tools", and "Help" menus, along with "Comment", "Share", and "Settings" icons. A "Table of contents" sidebar on the left lists various topics, with "Risk Management" highlighted. The main workspace contains two code cells. The first cell, under the "Risk Management" heading, contains Python code for setting up the environment with imports for os, numpy, pandas, and matplotlib, and configuring plot styles. The second cell, under the "Trading Bot" heading, contains a comment block for a finance-related project.

python101.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment Share ⚙️ A

RAM  Disk  Editing ^

+ Code + Text

↑ ↓ 🔗 🗨️ ✎ 📄 🗑️ ⋮

**Table of contents** ✕

- Algorithmic Trading
  - Vectorized Backtesting
    - Backtesting an SMA-Based Strategy
    - Backtesting a Daily DNN-Based Strategy
    - Backtesting an Intraday DNN-Based Strategy
  - Risk Management**
    - Trading Bot
    - Vectorized Backtesting
    - Event-Based Backtesting
    - Assessing Risk
    - Backtesting Risk Measures
      - Stop Loss
      - Trailing Stop Loss
      - Take Profit
      - Combinations
  - Backtesting Cryptocurrency Bitcoin

**▼ Risk Management**

```
[ ] 1 import os
     2 import numpy as np
     3 import pandas as pd
     4 from pylab import plt, mpl
     5 plt.style.use('seaborn')
     6 mpl.rcParams['savefig.dpi'] = 300
     7 mpl.rcParams['font.family'] = 'serif'
     8 pd.set_option('mode.chained_assignment', None)
     9 pd.set_option('display.float_format', '{:.4f}'.format)
    10 np.set_printoptions(suppress=True, precision=4)
    11 os.environ['PYTHONHASHSEED'] = '0'
```

**▼ Trading Bot**

```
[ ] 1 # import finance
     2 # finance.py
     3 # Finance Environment
     4 #
     5 # (c) Dr. Yves J. Hilpisch
     6 # Artificial Intelligence in Finance
     7 #
```

# Python in Google Colab (Python101)



python101.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Comment

Share



## Table of contents



- Algorithmic Trading
  - Vectorized Backtesting
    - Backtesting an SMA-Based Strategy
    - Backtesting a Daily DNN-Based Strategy
    - Backtesting an Intraday DNN-Based Strategy
  - Risk Management
    - Trading Bot
    - Vectorized Backtesting
    - Event-Based Backtesting**
    - Assessing Risk
    - Backtesting Risk Measures
      - Stop Loss
      - Trailing Stop Loss
      - Take Profit
      - Combinations
    - Backtesting Cryptocurrency Bitcoin

+ Code + Text

## Event-Based Backtesting

RAM   
Disk

Editing



```
1 #import backtesting as bt
2
3 # backtesting.py
4 # Event-Based Backtesting
5 # --Base Class (1)
6 #
7 # (c) Dr. Yves J. Hilpisch
8 # Artificial Intelligence in Finance
9 #
10
11 class BacktestingBase:
12     def __init__(self, env, model, amount, ptc, ftc, verbose=False):
13         self.env = env
14         self.model = model
15         self.initial_amount = amount
16         self.current_balance = amount
17         self.ptc = ptc
18         self.ftc = ftc
19         self.verbose = verbose
20         self.units = 0
21         self.trades = 0
22
23     def get_date_price(self, bar):
24         ''' Returns date and price for a given bar.
25         ...
```

<https://tinyurl.com/aintpupython101>

# Python in Google Colab (Python101)

Table of contents

- Algorithmic Trading
  - Vectorized Backtesting
    - Backtesting an SMA-Based Strategy
    - Backtesting a Daily DNN-Based Strategy
    - Backtesting an Intraday DNN-Based Strategy
  - Risk Management
    - Trading Bot
    - Vectorized Backtesting
    - Event-Based Backtesting
    - Assessing Risk
    - Backtesting Risk Measures
      - Stop Loss
      - Trailing Stop Loss
      - Take Profit
      - Combinations**
  - Backtesting Cryptocurrency Bitcoin

+ Code + Text RAM Disk Editing

## Combinations

```
1 tb.backtest_strategy(sl=0.015, tsl=None,  
2                       tp=0.0185, wait=5)
```

```
=====  
2018-01-17 | *** START BACKTEST ***  
2018-01-17 | current balance = 10000.00  
=====  
-----  
*** STOP LOSS (SHORT | -0.0203) ***  
-----  
*** STOP LOSS (SHORT | -0.0152) ***  
-----  
*** TAKE PROFIT (SHORT | 0.0189) ***  
-----  
*** TAKE PROFIT (SHORT | 0.0219) ***  
-----  
*** TAKE PROFIT (SHORT | 0.0192) ***  
-----  
*** STOP LOSS (LONG | -0.0154) ***  
-----  
*** TAKE PROFIT (SHORT | 0.0214) ***  
-----  
*** STOP LOSS (SHORT | -0.0158) ***  
-----  
*** TAKE PROFIT (SHORT | 0.0223) ***  
-----  
*** STOP LOSS (SHORT | -0.0162) ***
```

# Python in Google Colab (Python101)



python101.ipynb ☆

File Edit View Insert Runtime Tools Help [All changes saved](#)

Comment Share Settings A

RAM Disk Editing

Navigation icons: up, down, refresh, comment, edit, copy, trash, menu

## Table of contents

- Algorithmic Trading
  - Vectorized Backtesting
    - Backtesting an SMA-Based Strategy
    - Backtesting a Daily DNN-Based Strategy
    - Backtesting an Intraday DNN-Based Strategy
  - Risk Management
    - Trading Bot
    - Vectorized Backtesting
    - Event-Based Backtesting
    - Assessing Risk
    - Backtesting Risk Measures
      - Stop Loss
      - Trailing Stop Loss
      - Take Profit
      - Combinations
- Backtesting Cryptocurrency Bitcoin**

+ Code + Text

## Backtesting Cryptocurrency Bitcoin

- Financial Functions (ffn): <https://pmorrisette.github.io/ffn/>
- backtesting.py: <https://kernc.github.io/backtesting.py/>

15s

```
1 !pip install ffn
2 import ffn
3 import plotly.express as px
4 %pylab inline
5 #BTC-USD Bitcoin USD
6 df = ffn.get('btc-usd', start='2016-01-01', end='2021-12-31')
7 print('df')
8 print(df.head())
9 print(df.tail())
10 print(df.describe())
11 df.plot(figsize=(14,10))
12
13 returns = df.to_returns().dropna()
14 print('returns')
15 print(returns.head())
16 print(returns.tail())
17 print(returns.describe())
18 #ax = df.plot(figsize=(12,9))
19
20 perf = df.calc_stats()
21 perf.plot(figsize=(14, 10))
```

# Python in Google Colab (Python101)



python101.ipynb ☆

File Edit View Insert Runtime Tools Help [All changes saved](#)

Comment Share Settings A

## Table of contents

### Algorithmic Trading

#### Vectorized Backtesting

Backtesting an SMA-Based Strategy

Backtesting a Daily DNN-Based Strategy

Backtesting an Intraday DNN-Based Strategy

### Risk Management

Trading Bot

Vectorized Backtesting

Event-Based Backtesting

Assessing Risk

Backtesting Risk Measures

Stop Loss

Trailing Stop Loss

Take Profit

Combinations

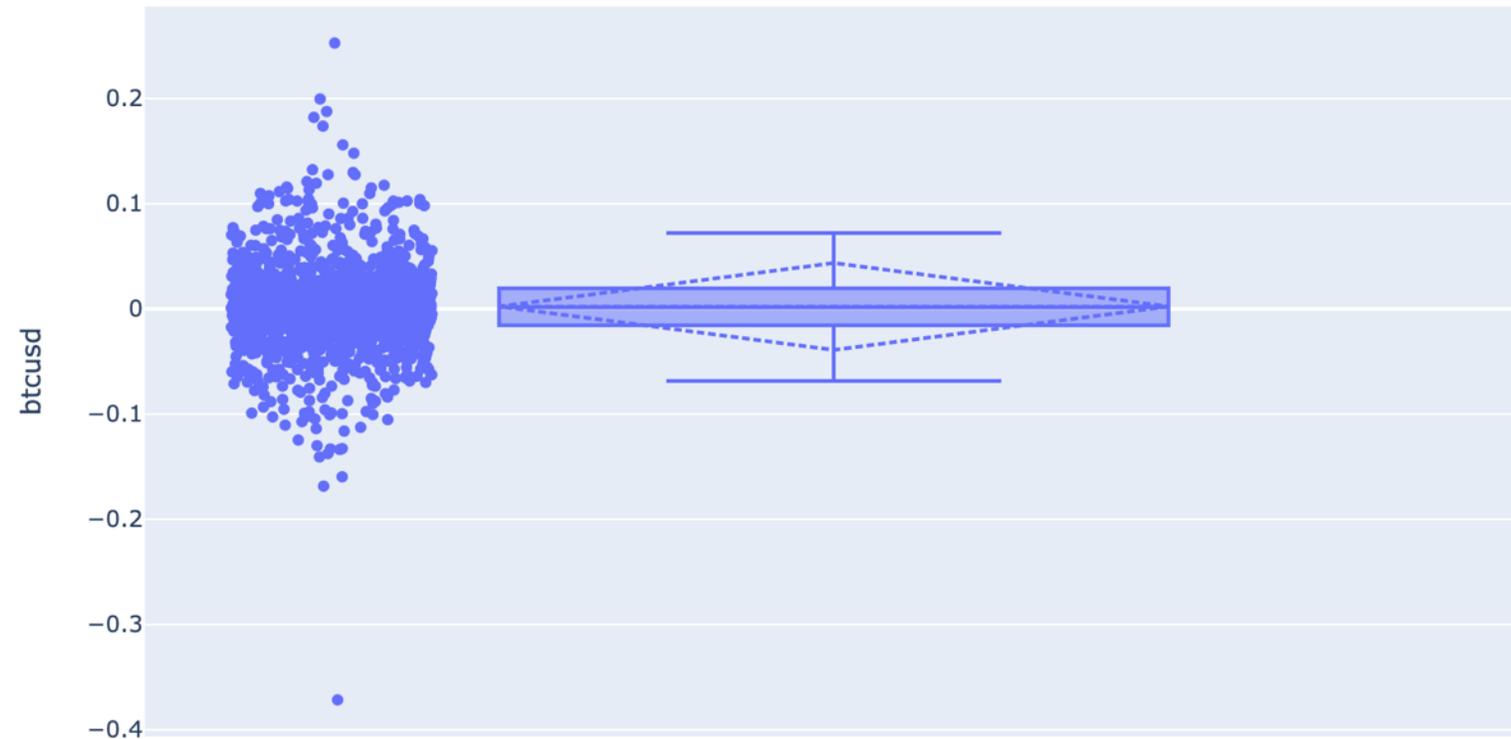
**Backtesting Cryptocurrency  
Bitcoin**

+ Code + Text

RAM  Disk  Editing

↑ ↓ 🔗 💬 ⚙️ 📄 🗑️ ⋮

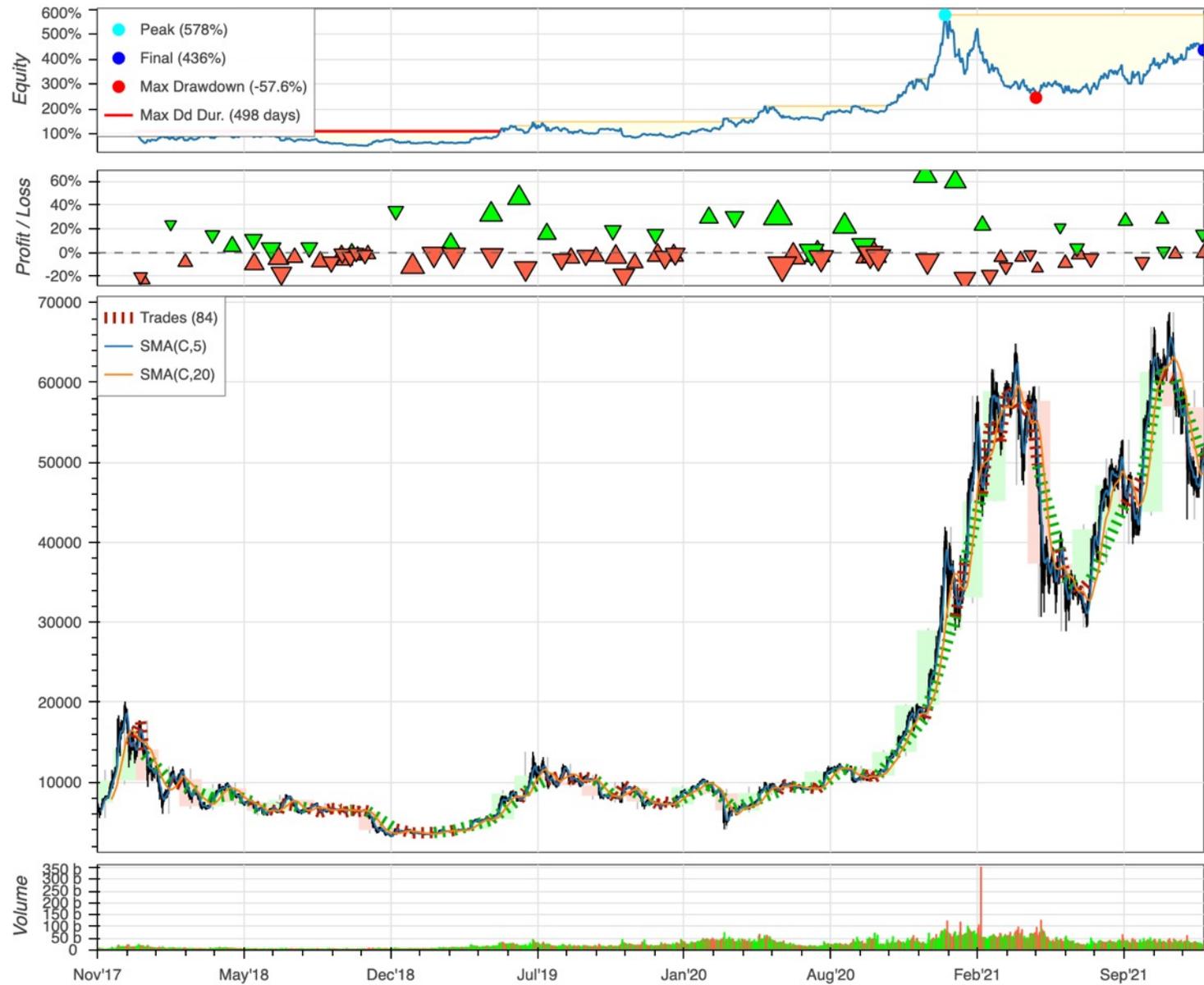
btcusd returns box



# Python in Google Colab (Python101)

- Algorithmic Trading
  - Vectorized Backtesting
    - Backtesting an SMA-Based Strategy
    - Backtesting a Daily DNN-Based Strategy
    - Backtesting an Intraday DNN-Based Strategy
  - Risk Management
    - Trading Bot
    - Vectorized Backtesting
    - Event-Based Backtesting
    - Assessing Risk
    - Backtesting Risk Measures
      - Stop Loss
      - Trailing Stop Loss
      - Take Profit
      - Combinations
- Backtesting Cryptocurrency
  - Bitcoin





# Teaching



- **Artificial Intelligence in Finance and Quantitative**
  - Fall 2021, Fall 2022, Fall 2023
- **Artificial Intelligence for Text Analytics**
  - Spring 2022, Fall 2023
- **Big Data Analytics**
  - Fall 2020, Spring 2023
- **Software Engineering**
  - Fall 2020, Fall, 2021, Spring 2022, Spring 2023
- **Artificial Intelligence**
  - Spring 2021, Fall 2022
- **Data Mining**
  - Spring 2021
- **Foundation of Business Cloud Computing**
  - Spring 2021, Spring 2022, Spring 2023
- **Python for Accounting Applications**
  - Fall 2023

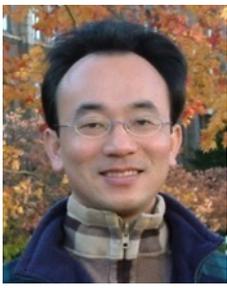
# Research Projects



1. **Applying AI technology to construct knowledge graphs of cryptocurrency anti-money laundering: a few-shot learning model**
  - MOST, 110-2410-H-305-013-MY2, 2021/08/01~2023/07/31
2. **Fintech Green Finance for Carbon Market Index, Corporate Finance, and Environmental Policies. Carbon Emission Sentiment Index with AI Text Analytics**
  - NTPU, 112-NTPU\_ORDA-F-003 , 2023/01/01~2024/12/31
3. **Digital Support, Unimpeded Communication: The Development, Support and Promotion of AI-assisted Communication Assistive Devices for Speech Impairment. Multimodal Cross-lingual Task-Oriented Dialogue System for Inclusive Communication Support**
  - NSTC 112-2425-H-305-002-, 2023/05/01-2026/04/30
4. **Establishment and Implement of Smart Assistive Technology for Dementia Care and Its Socio-Economic Impacts. Intelligent, individualized and precise care with smart AT and system integration**
  - NSTC, NSTC, 112-2627-M-038-001-, 2023/08/01~2024/07/31
5. **Use deep learning to identify commercially dental implant systems - observational study**
  - USTP-NTPU-TMU, USTP-NTPU-TMU-112-01, 2023/01/01~2023/12/31
6. **Metaverse AI Multimodal Cross-Language Task-Oriented Dialogue System**
  - ATEC Group x NTPU, NTPU-112A413E01, 2023/05/01~2026/04/30
7. **Metaverse Avatar Automatic Metadata Generation Module**
  - FormosaVerse x NTPU, NTPU-111A413E01, 2022/12/01~2023/11/30
8. **Pilot Study on Universal Data Processing for Code Generation Engine**
  - III x NTPU, NTPU-112A513E01, 2023/08/01~2023/12/22

# Summary

- This course introduces the **fundamental concepts, research issues, and hands-on practices of AI in Finance and Quantitative Analysis.**
- **Topics include:**
  1. Introduction to Artificial Intelligence in Finance and Quantitative Analysis
  2. AI in FinTech: Metaverse, Web3, DeFi, NFT, Financial Services Innovation and Applications,
  3. Investing Psychology and Behavioral Finance
  4. Event Studies in Finance
  5. Finance Theory
  6. Data-Driven Finance
  7. Financial Econometrics
  8. AI-First Finance
  9. Deep Learning in Finance
  10. Reinforcement Learning in Finance
  11. Algorithmic Trading, Risk Management, Trading Bot and Event-Based Backtesting
  12. Case Study on AI in Finance and Quantitative Analysis.



# AI in Finance and Quantitative Analysis



2020 Cohort



Accredited  
Educator



Solutions  
Architect  
Associate



Cloud  
Practitioner

## Contact Information

**Min-Yuh Day, Ph.D.**

Associate Professor

[Institute of Information Management, National Taipei University](#)

Tel: 02-86741111 ext. 66873

Office: B8F12

Address: 151, University Rd., San Shia District, New Taipei City, 23741 Taiwan

Email: [myday@gm.ntpu.edu.tw](mailto:myday@gm.ntpu.edu.tw)

Web: <http://web.ntpu.edu.tw/~myday/>

