

Introduction and overview

Machine Learning – Tools and applications for policy – Lecture 1

Iman van Lelyveld – Michiel Nijhuis

DNB Data Science Hub



Introduction and overview

1. Why is this course relevant?
2. What is the structure of the course — what can you expect?
3. How is the tooling, infrastructure and the Fintech landscape developing?

Introducing the instructors - contact details

3

This course will be taught by Iman van Lelyveld – Michiel Nijhuis

Course coordinator: Iman van Lelyveld

Email: iman.van.lelyveld@vu.nl

Office hours: by appointment

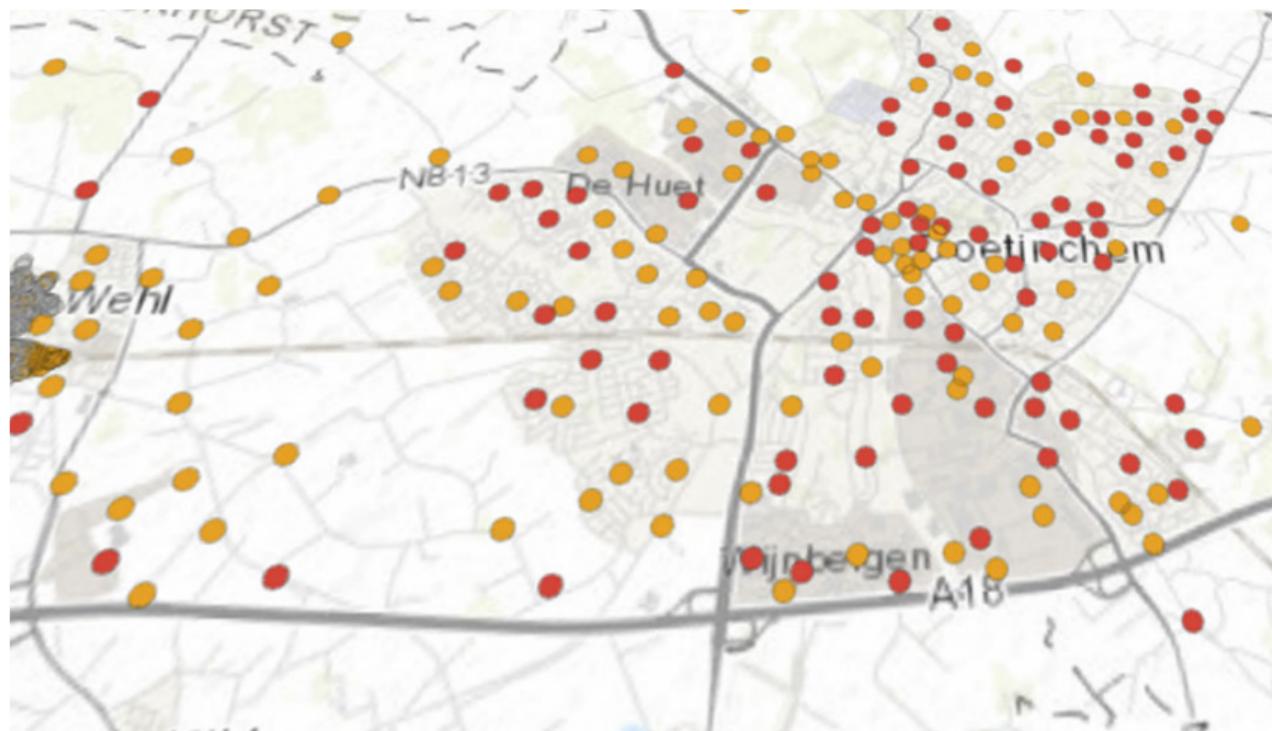
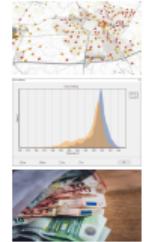
Website: [Personal page](#)

Instructor: Michiel Nijhuis

Email: michiel.nijhuis@dnb.nl

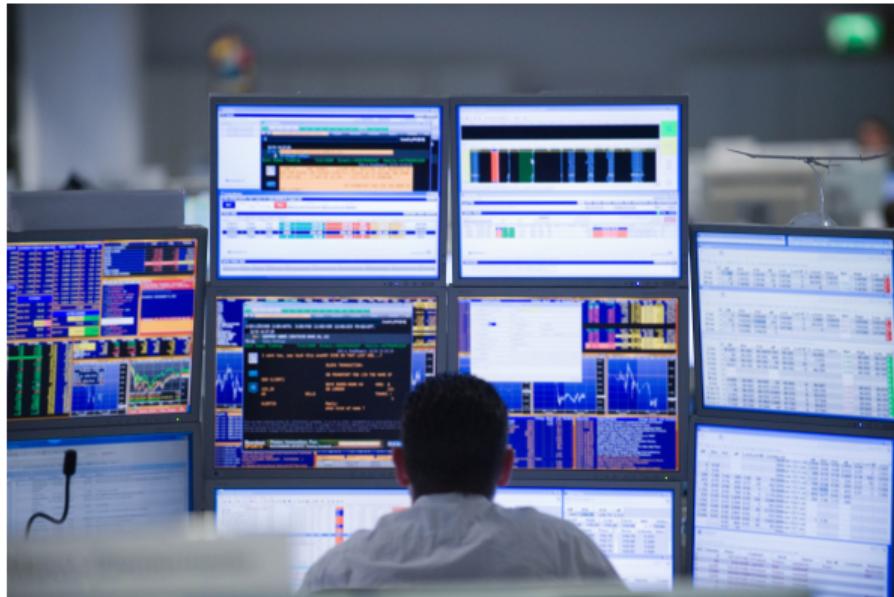
Office hours: by appointment

Michiel Nijhuis: Researching Algorithms for the Planning of Electrical Networks

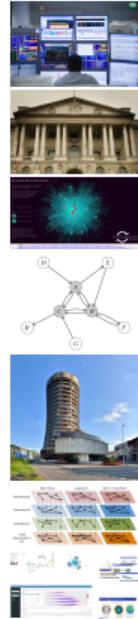


Bringing Machine Learning Algorithms in Production

Iman van Lelyveld: Forex trading at Deutsche Bank



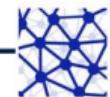
England in the 2008 crisis



Bank of



HOW
ABOUT
YOU



DataScience
Hub

- What are you expecting?
 - How are machine learning models used in practice?
 - What jobs fit Machine Learning?
 - How to apply machine learning to trading?
 - How can we use machine learning for financial research
 - Further improve code complexity
- ... but maybe something entirely different:

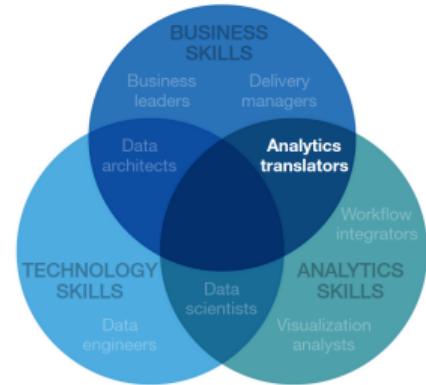
What is the objective of the course?

8

The **course goal is not to become experts** in the business, technology or analytics 'circle'.

The objective is to:

- Theory
 - provide an overview and understanding of popular Machine Learning (ML) and Artificial Intelligence (AI) techniques
 - understand the opportunities and limitations of these techniques
 - be able to interact with the experts
- Practice
 - work hands-on with ML/AI methods in Python
 - "demystify" the black box of ML/AI
 - prime you so that you can continue to learn by yourself



Source: [McKinsey](#)

Introduction and overview

Introducing the instructors

Housekeeping

Course outline

Opportunities are expanding

Should we intervene?

What is so special about Machine Learning?

The changing landscape



DataScience
Hub

- The slides will be provided
- Jupyter Notebooks and other relevant material will be posted to Github
- A good all-round book is Géron (2019)
 - Other good references are Hilpisch (2018), Raschka (2016), and van der Plas (2016)
 - Most of these authors have put their supporting material on-line: see for instance Jake van der Plas's book
- Where relevant, we cite papers and resources in the slides

L1 – Introduction and overview

1. Why is this course relevant?
2. What is the structure of the course — what can you expect?
3. How is the tooling, infrastructure and the Fintech landscape developing?

L2 – ML1 – introduction

1. What is ML? Can we see OLS as a ML problem?
2. What is ML applied to?
3. The outlines of the ML approach
 - supervised vs. unsupervised learning
 - (hyper)parameters and models
 - gradient descent and grid search
 - pre-processing features

Tutorial 1 – How to read data and use sklearn

1. Getting started with Python and data manipulation.
2. How is this different from Excel?
3. Read the data and get to know it.
4. Introduction to sklearn: where to find the buttons

L3 – ML2 – the basics

1. How do I assess success?
 - Confusion matrix, Receiver Operator Characteristic (ROC)
2. What are overfitting, bias and variance?
 - L2-regularisation
3. Support Vector Machines (SVM) and k nearest neighbours (KNN)

Tutorial 2 – Regressions versus Classifiers

1. Logit as a statistical model vs ML model
2. How to find the optimal (hyper)parameters
3. A different classifier: Support vector machines
 - Different types of kernels
 - First glimpse: Dangers of overfitting
 - Evaluating performance

L4 – ML3 – dimensionality and assessment

1. How to reduce dimensionality?
 - Principal Components Analysis (PCA)
2. Feature selection and regularization
 - How to tune model input by selecting features and beat overfitting?
 - How to select the most important features?
 - Examples RIDGE, LASSO, Elastic net

3. Is a “good” model always good? What is external validity?
 - Holdout, K-fold cross validation, Stratified K-fold. Leave-one-out (LOO)
4. What if we apply this to asset pricing?

Tutorial 3 – Data pre-processing and assessing model performance

1. How to pre-process: standardize your data
2. Pros and cons of standardization
3. Working with the confusion matrix
 - What if costs are not symmetric?
 - The trade-off between precision and recall

L5 – ML4 – improving weak learners

1. How to grow a decision tree? How to split?
 - Decision trees, purity measures
2. Can Ensemble Classifiers improve weak learners?

~~Bagging, boosting, AdaBoost, XGBoost~~

3. Can we use forests in other ways?

- Isolation forests

Tutorial 4 – Cross-validation applied to LASSO variable selection

1. Looking closer at cross-validation (CV) and holdouts
2. K-fold, Leave-one-out
3. Splitting your data into training and testing samples
4. How to use CV to tune a LASSO model

L6 – ML5 – unsupervised learning and explainability

1. Supervised versus unsupervised learning
2. What can we do with unsupervised learners?
 - K-means, t-SNE, DBSCAN, Gaussian mixtures
3. Can we transfer knowledge?
4. How to open the black box and explain results?

Tutorial 6 – Finding clusters and neighbors

1. Implementing K-means and DBSCAN
2. Hierarchical clustering: Bottom-up or Top-down?
3. Visual inspection of results

L7 – ML6 – Neural Nets

1. How to go from Perceptron to Neural Nets and beyond?
 - Multilayer Perceptrons (MLP), ADALine, backpropagation
2. What are common Neural Net architectures?
 - Convolutional NN, Recurrent NN, Long Short Term Memory (LSTM)
3. Other considerations
 - Fairness, adversarial attacks
4. Practical tips for implementation

L8 –ML7 – Natural Language Processing

1. What are the main approaches in textual analysis?
2. Going beyond simple word counts
3. The arrival of Large Language Models (LLM)

Tutorial 7 – NLTK and sentiment analysis

1. Constructing a bag of words
2. Classifying sentiments (positive/negative)
3. Example with financial news data

L9 – ML8 – generative models

1. With the release of ChatGPT, generative models have gained center stage
2. What makes them tick?
3. What are the possibilities? What the dangers?

Tutorial 8 – Explainable AI

1. Permutation importance with ELI5
2. Partial dependency plots
3. Shapley values

L10 – Summary

1. Discuss some things that can go wrong
 - survivorship bias, input errors and deceit
 - fairness and discrimination
2. What is the reaction of authorities?

What Math/Stats background do you need?

- For a refresher see a textbook like Greene (2013)
- Partial derivatives, e.g.

$$\frac{\partial(x^3 + y^2 + 1)}{\partial x}$$

- Matrix and vector operations

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 7 \\ 8 \\ 9 \end{bmatrix} = \begin{bmatrix} 1 \times 7 + 2 \times 8 + 3 \times 9 \\ 4 \times 7 + 5 \times 8 + 6 \times 9 \end{bmatrix} = \begin{bmatrix} 50 \\ 122 \end{bmatrix}$$

- Vector dot product

$$z = \mathbf{w}^T \mathbf{x} = \sum_{j=0}^m \mathbf{w}_j \mathbf{x}_j$$

$$[1 \ 2 \ 3] \times \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} = 1 \times 4 + 2 \times 5 + 3 \times 6 = 32.$$



Introduction and overview

Introducing the instructors

Housekeeping

Course outline

Opportunities are expanding

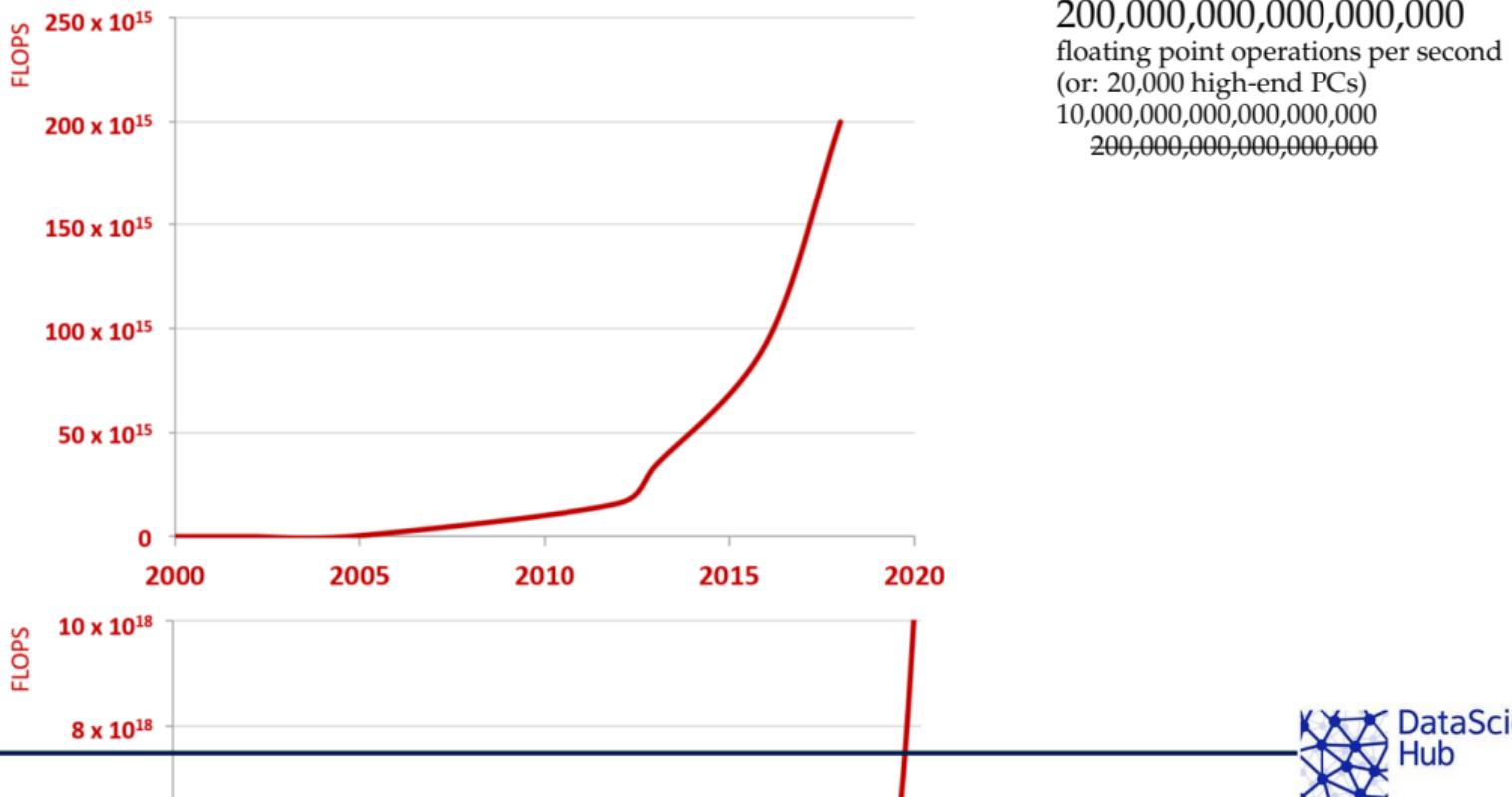
Should we intervene?

What is so special about Machine Learning?

The changing landscape

Computing power increases continuously

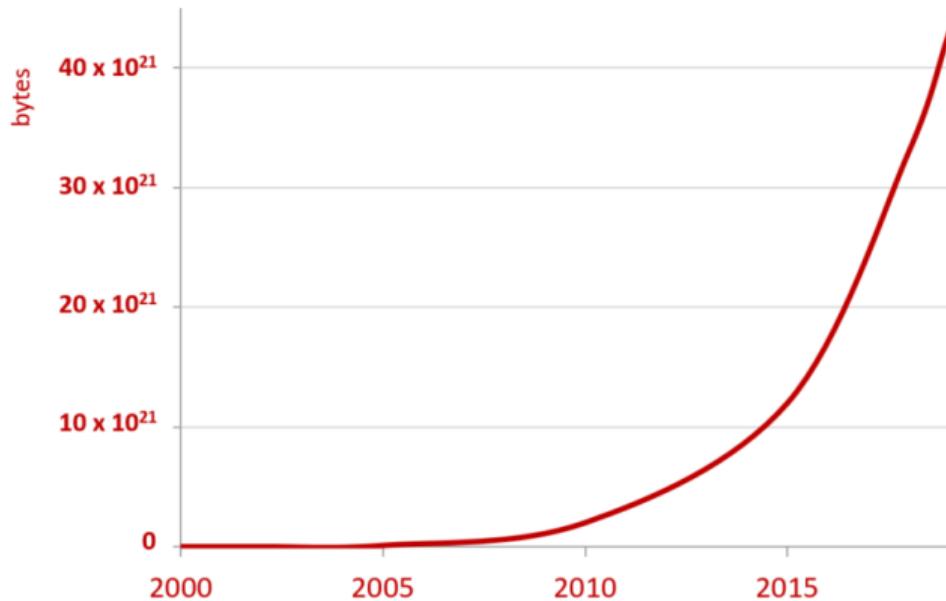
21



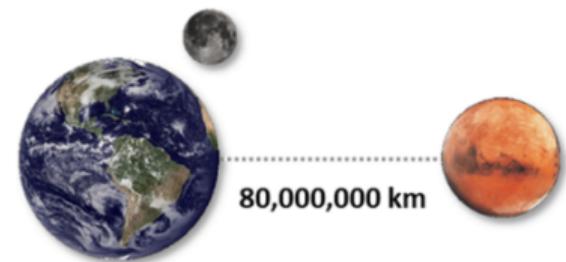
DataScience
Hub

We collect ever more data

22

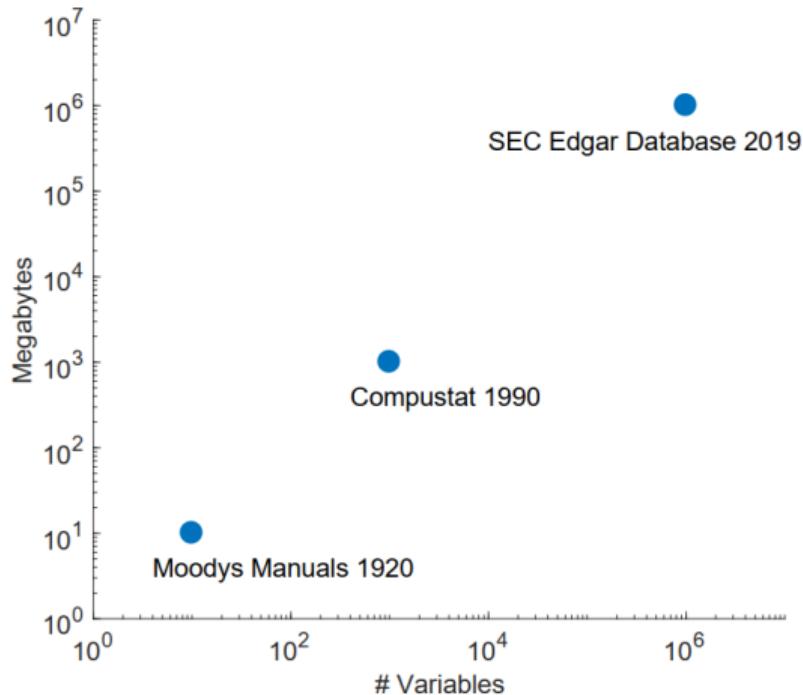


Annual in 2005 ≡ daily in 2020
Or, on CD-rom



More data available to extend the set of x_t in asset pricing

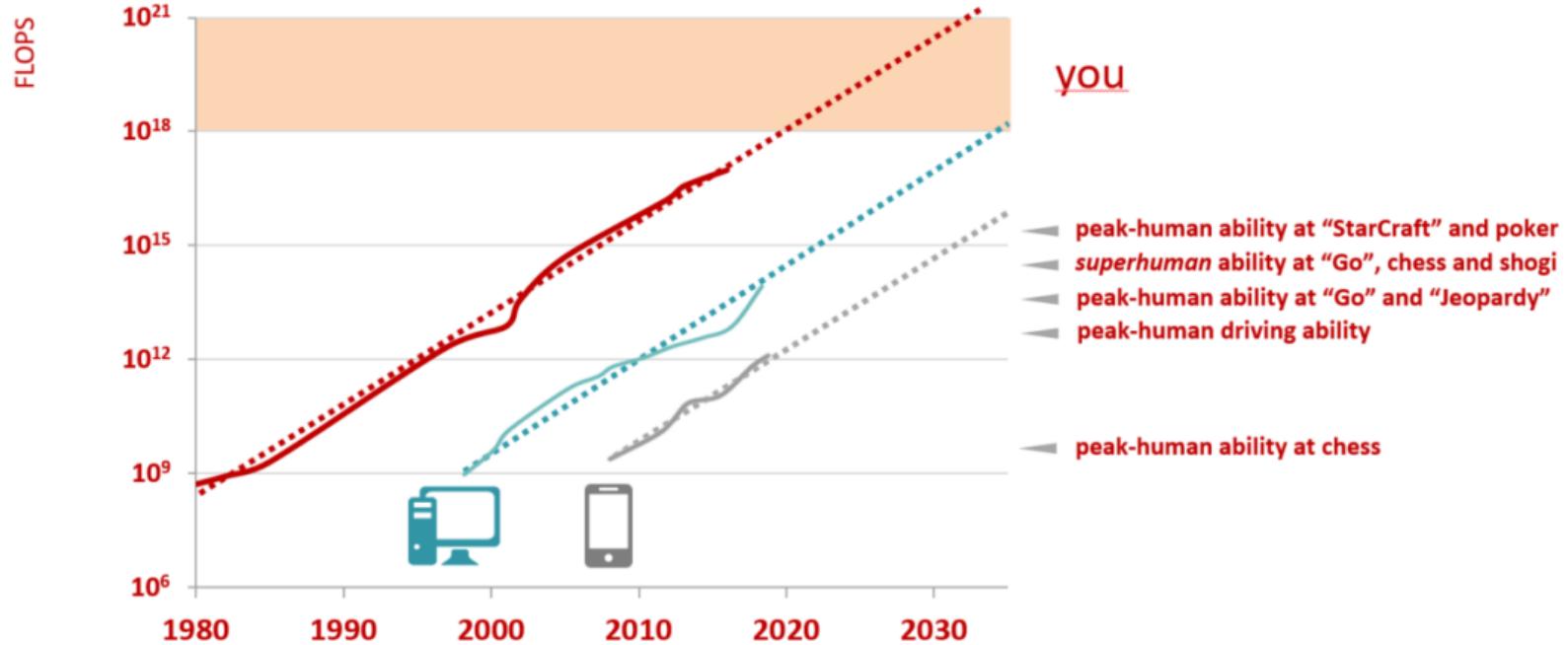
23



Source: Nagel (2019)

CPU power + algos + data has resulted in astounding progress

24



2019 *This Is What Happens In An Internet Minute*





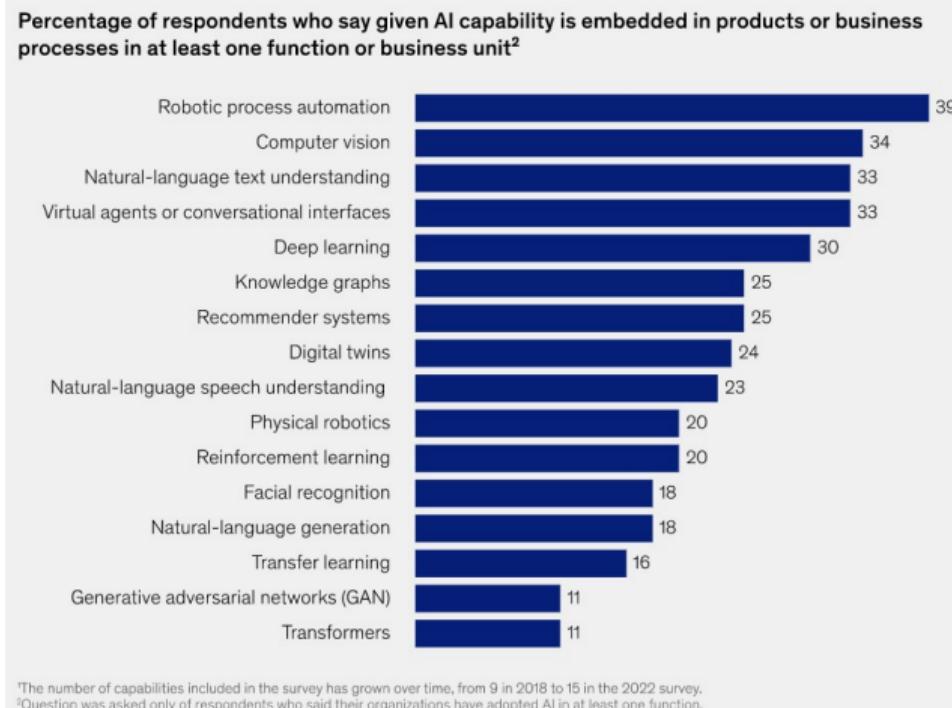
Changes in personal interests or in population characteristics (adaptive news access)

Adversary activities (avoiding spam filters; credit card fraud)

Changes in population characteristics (credit scoring)

... is used in many ways ...

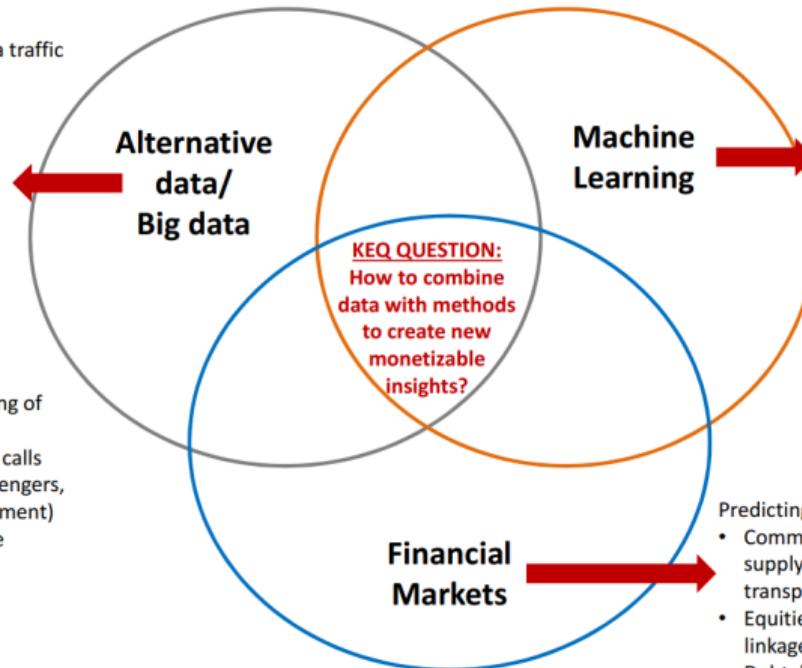
27



Source: [StateAI2022](#)

Examples

- Mobile device voice and data traffic
- Debit/credit card data
- Online shopping data
- Satellite data
- Corporate jet movements
- Weather data (crop yields,...)
- Clickstream (online ad clicks)
- Store/webpage visits
- Social media sentiment
- Foot traffic data
- Shopping center data
- Patenting data
- Job postings/hiring data
- Entertainment data (streaming of music, video, online games)
- Firm disclosures, conference calls
- Transportation data (air passengers, cargo, maritime vessel movement)
- App usage and mobile device geolocation data
- ...



Statistical methods for learning from data:

- Regressions
- Decision trees
- Clustering
- Computer vision
- Pattern recognition
- Natural language processing
- Neural networks
- ...

Predicting asset prices:

- Commodities/agriculture: supply/demand factors (weather, transp. costs, political risks,...)
- Equities: sales, competition, linkages (cust-supply chains), ...
- Debt: bankruptcy risk
- Real estate



Even has its own journal!

29

The screenshot shows the homepage of The Journal of Finance and Data Science. At the top, there is a dark blue header with the journal's name and "Open access". Below the header, there are links for "Latest issue", "All issues", and "Submit your article". A search bar with a magnifying glass icon is centered below the header. The main content area features a large image of the journal cover for Volume 5, Issue 2, which is titled "COSMOS trader – Chaotic Neuro-oscillatory multiagent financial prediction and trading system". Below the cover, the text "Volume 5, Issue 2" is displayed, along with "Pages 61-126 (June 2019)" and a link to "Download full issue". On the left side, there is a sidebar with "Actions for selected articles" (Select all / Deselect all), "Download PDFs", "Export citations", and "Show all article previews". To the right of the sidebar, there is a section for setting up alerts with a "Sign in to set up alerts" button. The main content area also includes links for "Open access", "Editorial Board", "Page ii", and "Download PDF" for the featured article. Other articles listed include "Research article • Open access COSMOS trader – Chaotic Neuro-oscillatory multiagent financial prediction and trading system" by Raymond S.T. Lee, and "Research article • Open access Can artificial intelligence enhance the Bitcoin bonanza". The bottom right corner features the "DataScience Hub" logo.

The Journal of Finance and Data Science

Open access

Latest issue All issues Submit your article ↗

Search in this journal

Volume 5, Issue 2

Pages 61-126 (June 2019)

Download full issue

Actions for selected articles

Select all / Deselect all

Download PDFs

Export citations

Show all article previews

Receive an update when the latest issues in this journal are published

Sign in to set up alerts

Open access

Editorial Board

Page ii

Download PDF

Research article • Open access
COSMOS trader – Chaotic Neuro-oscillatory multiagent financial prediction and trading system
Raymond S.T. Lee
Pages 61-82

Download PDF Article preview ▾

Research article • Open access
Can artificial intelligence enhance the Bitcoin bonanza

DataScience Hub

Source: Journal of Finance and Data Science

Introduction and overview

Introducing the instructors

Housekeeping

Course outline

Opportunities are expanding

Should we intervene?

What is so special about Machine Learning?

The changing landscape

Why Google thinks we need to regulate AI

Companies cannot just build new technology and let market forces decide how it will be used

SUNDAR PICHAI

+ Add to myFT



DataScience
Hub

Introduction and overview

Introducing the instructors

Housekeeping

Course outline

Opportunities are expanding

Should we intervene?

What is so special about Machine Learning?

The changing landscape

In 1958 the New York Times reported that the Perceptron, an early AI machine developed at Cornell University with military money, was “the embryo of an electronic computer that [the American Navy] expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence”.

(Economist, May 14th 2015)

"Every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it."

(Dartmouth Artificial Intelligence Conference (1956))

AI can refer to anything from a computer program playing a game of chess, to a voice-recognition system like Amazon's Alexa interpreting and responding to speech.

The technology can broadly be categorized into three groups:

- **narrow AI**: skilled at **one specific task**, e.g. IBM's Deep Blue (beat chess grand master Garry Kasparov – 1996), or Google DeepMind's AlphaGo (Go master Lee Sedol – 2016).
- **artificial general intelligence (AGI)** ≡ human-level
- **superintelligent AI**

"an intellect that is much smarter than the best human brains in practically every field, including scientific creativity, general wisdom and social skills"

In other words, it's **when machines have outsmarted us**.

- “Machine learning research is part of research on artificial intelligence, seeking to provide knowledge to computers through data, observations and interacting with the world. That acquired knowledge allows computers to correctly generalize to new settings.”
- “A well-posed learning problem: A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.”
- “Machine learning is the science of getting computers to act without being explicitly programmed.”
- “Machine learning algorithms can figure out how to perform important tasks by generalizing from examples.”

Machine learning = Statistics + Programming!

 **Mat Velloso**
@matvelloso

Difference between machine learning
and AI:

If it is written in Python, it's probably
machine learning

If it is written in PowerPoint, it's
probably AI

22/11/18, 5:25 PM

3,514 Retweets 10.8K Likes

Fitting **Elastic Net** regression in:

R

```
1 eNet <- glmnet(X, Y, alpha = 0.5, lambda = myLambdas)
```

MATLAB

```
1 eNet = lasso(X, Y, 'Alpha', 0.5, 'Lambda', myLambdas)
```

PYTHON/scikitLearn:

```
1 eNet = ElasticNet(alpha = myLambda, l1_ratio= 0.5)
2 eNet.fit(X, Y)
```

The crucial thing is to know what **elastic net**, **alpha** and **lambda** mean!

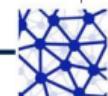
We choose Python because of the ecosystem

38

Python's Scientific Ecosystem



python™



DataScience
Hub

Introduction and overview

Introducing the instructors

Housekeeping

Course outline

Opportunities are expanding

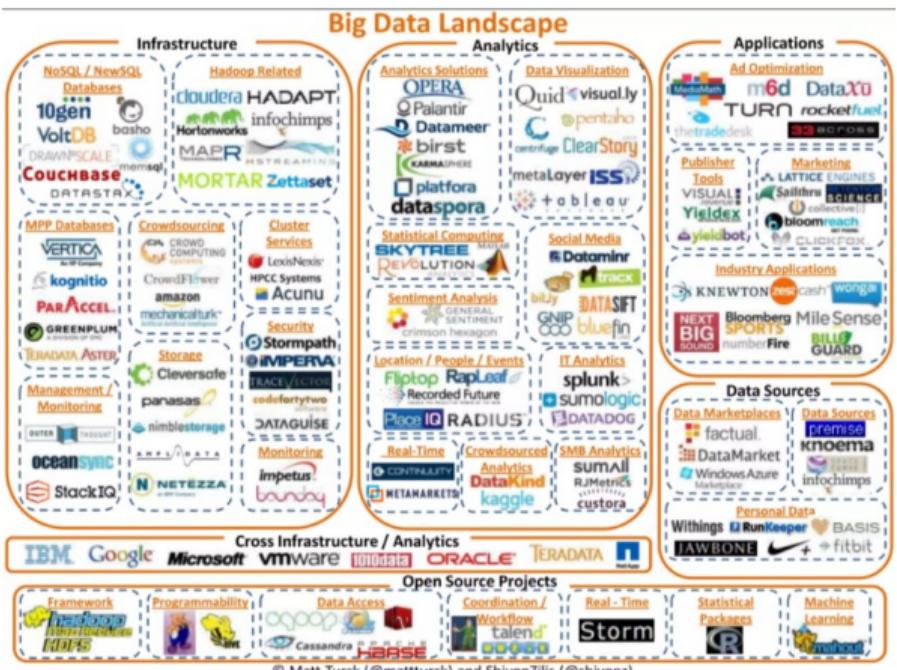
Should we intervene?

What is so special about Machine Learning?

The changing landscape

Tooling is evolving fast ...

40



2012

2023



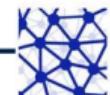
2016



2020



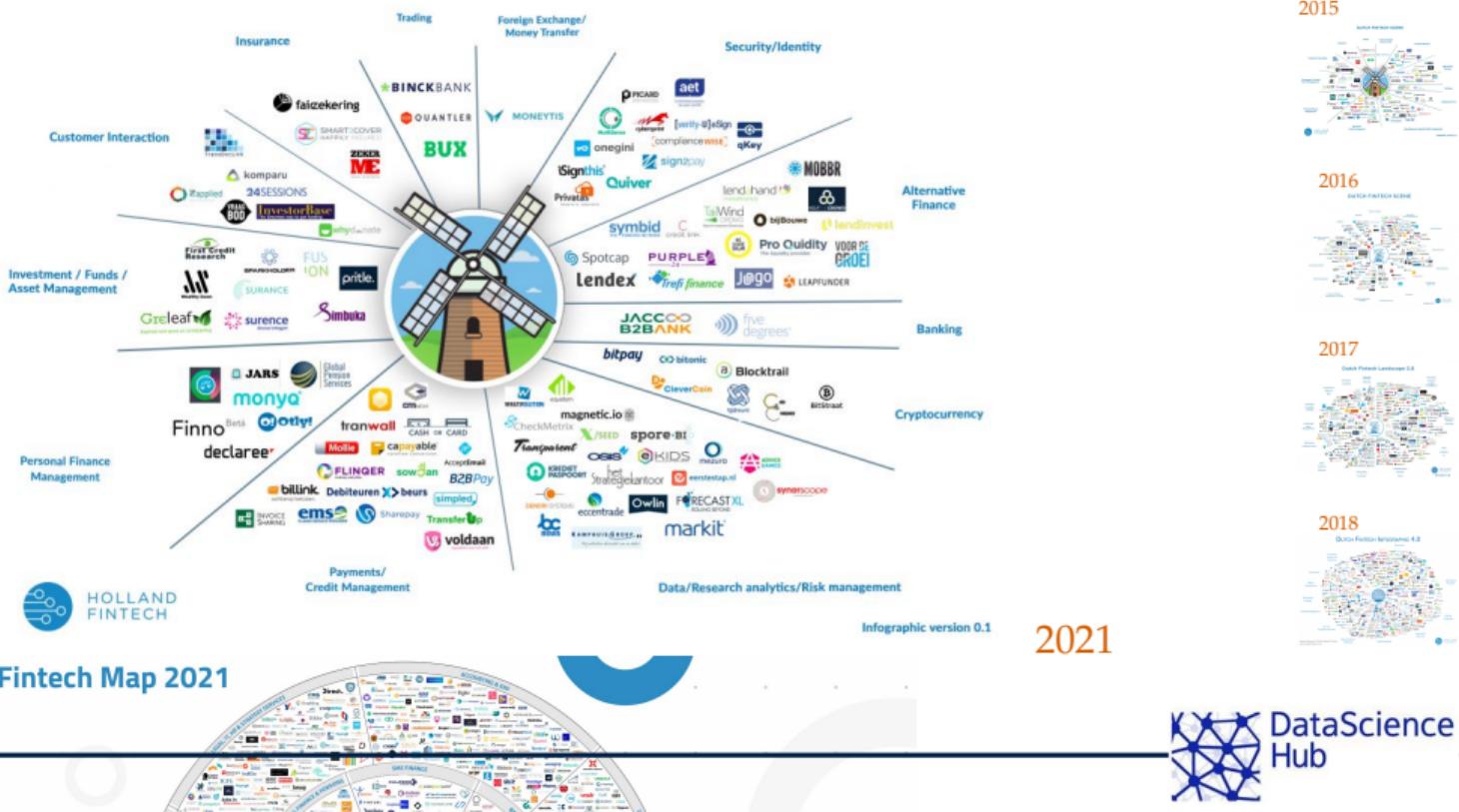
2021



DataScience
Hub

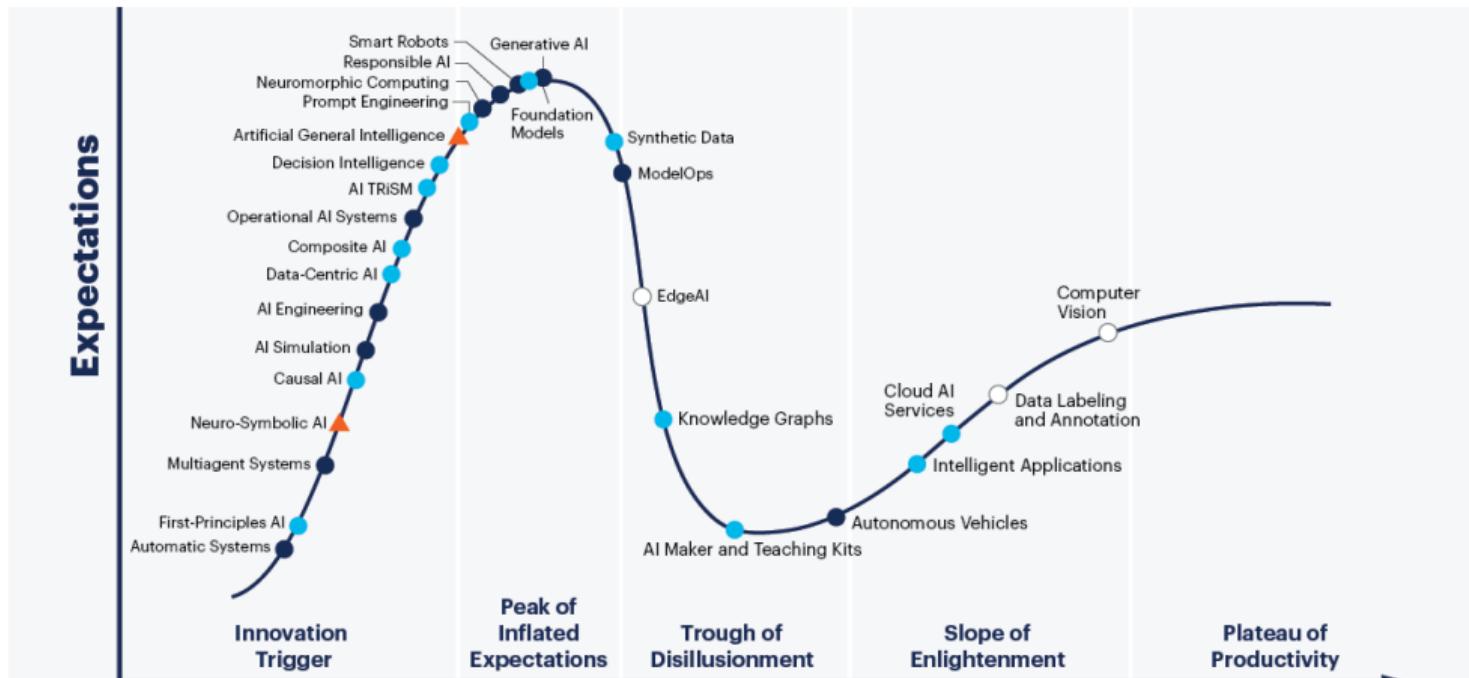
... as well as the Fintech landscape

41



Data Science Hype Cycle – 2023

42



Plateau will be reached:

○ less than 2 years

● 2 to 5 years

● 5 to 10 years

▲ more than 10 years

✖ obsolete before plateau

DataScience
Hub

In this lecture we covered:

1. Introductions and housekeeping
2. Why is this course relevant and how does it fit in the Honors programme
3. A first discussion of how ML is different
4. Sketching the landscape

- **Artificial intelligence (AI)**: A broad discipline with the goal of creating intelligent machines, as opposed to humans' and animals' natural intelligence. It is a **catch-all term** that nonetheless captures the long term ambition of the field to build machines that emulate and then exceed the full range of human cognition.
- **Machine learning (ML)**: A subset of AI that often uses statistical techniques to give machines the ability to "learn" from data without being explicitly given the instructions for how to do so. This process is known as "training" a "model" using a learning "algorithm" that progressively improves model performance on a specific task.
- **Deep learning (DL)**: An area of ML that attempts to mimic the activity in layers of neurons in the brain to learn how to recognize complex patterns in data. The "deep" in deep learning refers to the large number of layers of neurons in contemporary ML models to achieve performance gains.
- **Reinforcement learning (RL)**: An area of ML concerned with developing software agents that learn **goal-oriented behavior by trial and error** in an environment that provides rewards or penalties in response to the agent's actions (call it **reinforcement**) towards achieving that goal.

What if you haven't had enough?

45

- This course is too short to cover everything
- Luckily, there are a number of very good free courses out there
 - [Google's crash course on Machine Learning](#)
 - An excellent full length MOOC by [Andrew Ng on Coursera](#)
 - Courses from Harvard, IBM and Microsoft at [EDX](#)
 - Also see [Udemy](#), although you have to pay a little bit
 - [coding for economists](#)
- Check out the use cases at [Kaggle](#)
- Or if you are in for a game: [Code Combat](#)
- Please let us know if you have come across others!

What about softcopy resources?

46

- There are an insane amount of blogs, repo's out there. Here is wildly incomplete list
 - Some really good links about ML an Econ on Dario Sansone's [webpage](#)
 - [Economics and Data Science resources](#)
 - For code, see the website [paperswithcode.com](#)
 - A nice book on interpretability as a [website](#) or as a Pact book Molnar ([2019](#))
 - Great book (in progress) [Coding for economists](#) with great Python examples.
 - Very nice cheatsheets for folks coming from other languages: [Stata](#), R, Matlab, Excel
 - Resources that go with the [Python for Finance](#) by Hilpisch
 - More Quant resources at [O'Reilly](#)
 - Cheatsheets for folks coming from other languages: [Stata](#). Still looking for R, Matlab, Excel
 - Great newslettern [Data Mining Gaps](#)
- Please let us know if you have come across others!

What about hardcopy resources?

- There are also numerous books out there, some of them free and all of them come with a lot of supplementary material
 - McKinney, W. (2022). Python for Data Analysis, 3rd edition. <https://wesmckinney.com/book/>
 - Géron, A. (2019). Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly. Retrieved April 21, 2019, from <https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/>
 - Wentworth, P., Elkner, J., Downey, A. B., & Meyers, C. (2019). How to think like a computer scientist: Learning with Python [arXiv: 1011.1669v3 ISSN: 1098-6596].
 - Raschka, S., & Mirjalili, V. (2017). Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow. Retrieved July 27, 2019, from <https://www.packtpub.com/big-data-and-business-intelligence/python-machine-learning-second-edition>
 - James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning: With Applications in R.
 - Hastie, T., Tibshirani, R., & Friedman, J. (2017). The Elements of Statistical Learning: Data Mining, Inference, and Prediction [arXiv: 1011.1669v3 ISSN: 03436993].
 - Efron, B., & Hastie, T. (2017). Computer Age Statistical Inference.
 - Provost, F., & Fawcett, T. (2013). Data Science for Business: What you need to know about data mining and data-analytic thinking (O'Reilly, Ed.).
 - Sweigart, A. (2015). Automating the Boring Stuff with Python. <https://automatetheboringstuff.com/>

What about your tech stack

48

This course is not about research methods but if you want to take things serious you should invest in the following technologies

- Organize your thoughts: OneNote, Mindmap, Gingko App
- Organize your references: [Zotero](#), [Mendeley](#), Jabref. You might find this tedious but it will pay off by the time you have to find that one article or get the reference right.
- Use \LaTeX . [Overleaf](#) makes it really easy to start
- Pick your favourite editor – preferably suitable for multiple languages and \LaTeX .
[VSCode](#), [Sublime Text](#), [PyCharm](#).
 - Visual Studio Code (\neq Visual Studio). Has an active marketplace and integrates well with the Microsoft stack
- Check out [Open metric](#) for a much more thorough list

Things are moving fast so please let us know if you have come across interesting tools!

- Most financial data sets are under lock and key but some of it is free

Araujo 2023

Awesome list

DB.nomics

WRDS

Yahoo finance

Mockaroo

Draw my data

Kolanovic and Krishnamachari
(2017)

COVID travel data

Synthetic data (JP Morgan)

Google trend and financial time
series

Google Trend and unemploy-
ment

re3data

Open source package to access a wide range of public finance related sites

Overview of public data sets, mostly trade. Opencorporates

Amazing portal with data from BIS, ECB, OECD and many others. It has an API from
within Python

Wharton Database, See Leonard Wolk's lectures

Various APIs available (e.g., [Rapidapi](#))

If you need to make up a test data set with realistic features

Generate a data set with some properties

has a great chapter "Handbook of Alternative Data"

useful to generate realistic (but fake) data

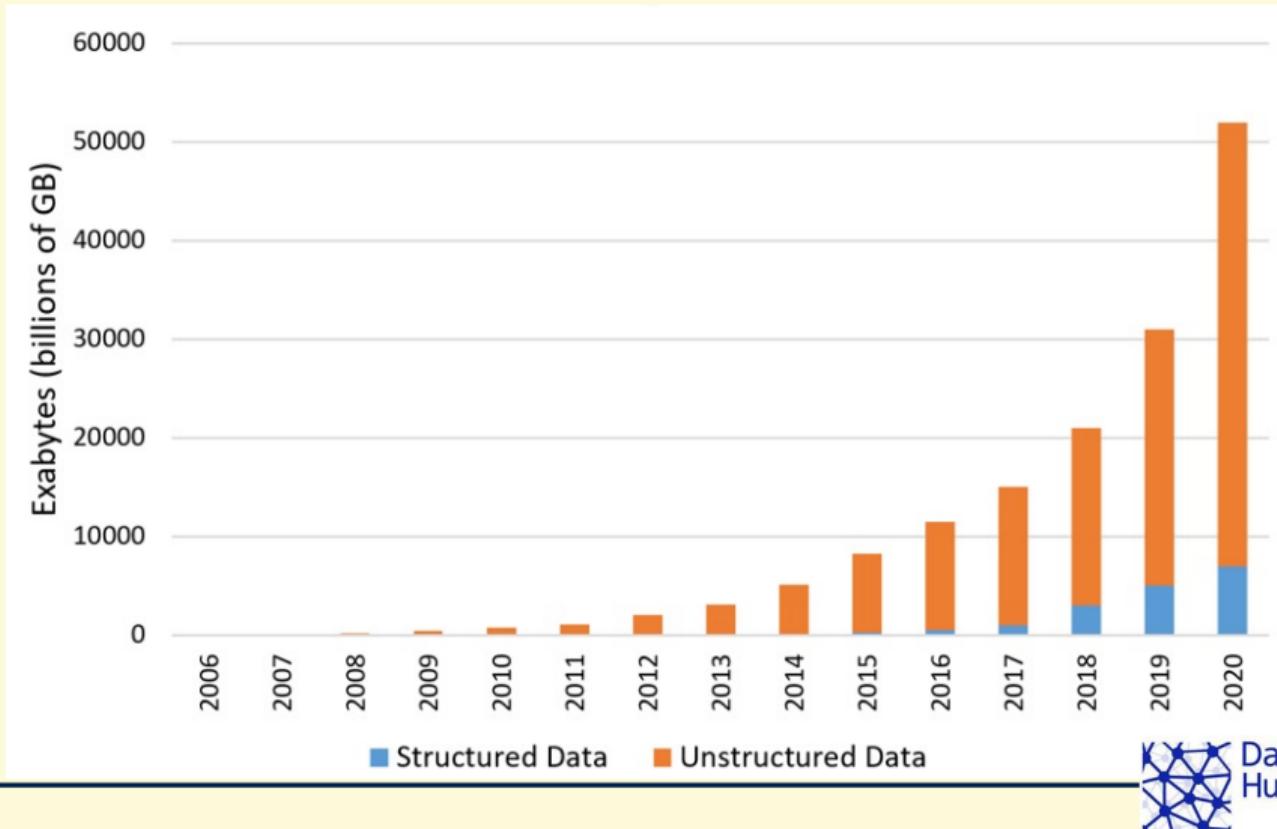
Nice example of what to do with Google Trend

Database of data sets

- See the [api-packages-overview.xls](#) with an overview of available APIs on De
Nederlandse Bank GitHub. Let us know if you have comments or
data_science@dnb.nl

The Cambrian explosion ... of data

50

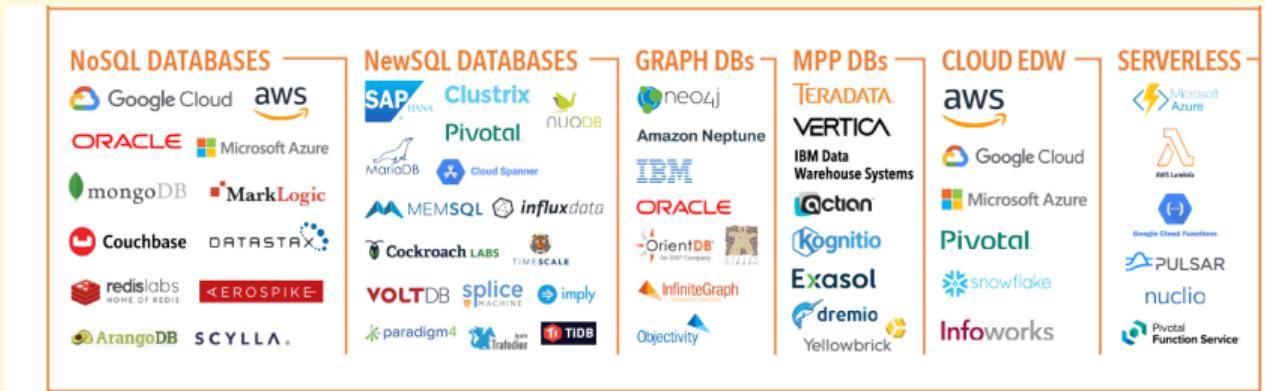


What types of database are out there

51

- Relational vs Non-relational databases

- Relational engines require consistency at the **end of each transaction**, whereas non-relational engines only require that the database be consistent at **some point in time** (i.e., eventually)



Relational Database Management Systems (RDMS)

- strictly enforce type (e.g. numeric, string, etc.)
- example: SAS

strong layout data is known in advance but use not yet

weak variable data

Source: Perkins et al. (2018)

Key-value (KV)

- Simple keys to values
- example: [key-value stores](#)

strong (horizontally) scalable and fast for unrelated data (e.g. users' session data)

weak no index hence only basic **CRUD** (Create, Read, Update, Delete)

Columnar

- Everything in a column, rows are not kept together, nice versioning
- example: [Monet db](#)

strong indexing web pages

weak it would be best if you know query structure in advance

Document

- any number of fields with unlimited nesting, JavaScript Object Notation (JSON)
- example: MongoDB, CouchDB, [document stores](#)

strong highly variable data, good match with object-oriented languages

weak verbose

Graph

- focus on the relation between nodes than on the actual node information
- example: Neo4J, [graph databases](#)

strong social network queries

weak comparing node info between 2 nodes → this is then stored elsewhere

Where to find handy tidbits?

55

- Holidays can be found using the [Holidays](#) PyPi package
- ISO codes for countries, currencies etc. can be found with the [pycountry](#) PyPi package
- Testing out your [regular expression regex101](#) site
- [Papers with code](#)
- Compute the distance between locations (API) at <https://developer.here.com>
- The list is endless ... please let me know if you find other interesting ones.

Best 12 AI Tools in 2023

Solves anything - ChatGPT

Writes anything - Writesonic

Generates Art - Midjourney

Generates Code - Replit

Generates Video - Synthesia

Generates Music - Soundraw

Generates TikToks - Fliki

Generates Avatars - Starrytars

Generates PPTs - Slides AI

Edit Pictures - Remini

Edit Videos - Pictory

Summarize Notes - Wordtune

-  Aldasoro, I., & Faia, E. (2015). Systemic Loops and Liquidity Regulation. CEPR Discussion Paper, (10918).
-  Araujo, D. (2023). Gingado: A Machine Learning Library Focused on Economics and Finance. SSRN Electronic Journal.
-  Efron, B., & Hastie, T. (2017). Computer Age Statistical Inference.
-  Géron, A. (2019). Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly. Retrieved April 21, 2019, from <https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/>
-  Greene, W. H. (2013). Econometric Analysis [Publication Title: Econometric Analysis]. MacMillan.
-  Hastie, T., Tibshirani, R., & Friedman, J. (2017). The Elements of Statistical Learning: Data Mining, Inference, and Prediction [arXiv: 1011.1669v3 ISSN: 03436993].

-  Heijmans, R., Heuver, R., Levallois, C., & van Lelyveld, I. (2016). Dynamic visualization of large financial networks. *Journal of Network Theory in Finance*, 2(2), 57–79.
-  Hilpisch, Y. (2018). Python for Finance: Mastering Data-Driven Finance. O'Reilly.
Retrieved March 1, 2019, from
<http://shop.oreilly.com/product/0636920117728.do>
-  in 't Veld, D., & van Lelyveld, I. (2014). Finding the core: Network structure in interbank markets. *Journal of Banking and Finance*, 49, 27–40.
-  James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013).
An Introduction to Statistical Learning: With Applications in R.
-  Kolanovic, M., & Krishnamachari, R. T. (2017). Big Data and AI Strategies - Machine Learning and Alternative Data Approach to Investing Quantitative and Derivatives Strategy. *J.P. Morgan report*.
-  McKinney, W. (2022). Python for Data Analysis, 3rd edition.
<https://wesmckinney.com/book/>

-  Molnar, C. (2019). Interpretable Machine Learning.
<https://christophm.github.io/interpretable-ml-book/cite.html>
-  Nagel, S. (2019). Asset Pricing and Machine Learning - Lecture 1.
Princeton Lectures in Finance.
-  Perkins, L., Redmond, E., & Wilson, J. R. (2018).
Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement.
The Pragmatic Bookshelf. Retrieved April 20, 2019, from
<http://www.pragprog.com>.
-  Provost, F., & Fawcett, T. (2013).
Data Science for Business: What you need to know about data mining and data-analytic thinking.
(O'Reilly, Ed.).
-  Raschka, S. (2016). Python Machine Learning.



Raschka, S., & Mirjalili, V. (2017).

Python machine learning: Machine learning and deep learning with Python, scikit-learn,
Retrieved July 27, 2019, from <https://www.packtpub.com/big-data-and-business-intelligence/python-machine-learning-second-edition>



Sweigart, A. (2015). Automating the Boring Stuff with Python.

<https://automatetheboringstuff.com/>



van der Plas, J. (2016). Python Data Science Handbook. O'Reilly. Retrieved October 21, 2019, from <https://jakevdp.github.io/PythonDataScienceHandbook/>



Wentworth, P., Elkner, J., Downey, A. B., & Meyers, C. (2019).

How to think like a computer scientist: Learning with Python [arXiv: 1011.1669v3
ISSN: 1098-6596].

