Data Science at ING / WBAA

Max Baak — 9 Dec 2020

WHOLESALE BANKING advanced analytics

ING

Max Baak - data scientist / statistician / physicist

Background in HEP (2002 - 2015)

- 2007: PhD @ Nikhef, on research at BaBar experiment (SLAC)
- 2007 - 2015: ATLAS experiment
    - 2008 - 2015: research fellow then staff at CERN
    - DQ, SUSY searches, Gfitter, statistics
- Soft spot for statistical methodology

KPMG Advanced Analytics & Big Data team (2015 - 2018)

- Data science consultancy
- Chief Data Scientist, then team lead (2017).

ING Wholesale Banking Advanced Analytics (2019 - now)

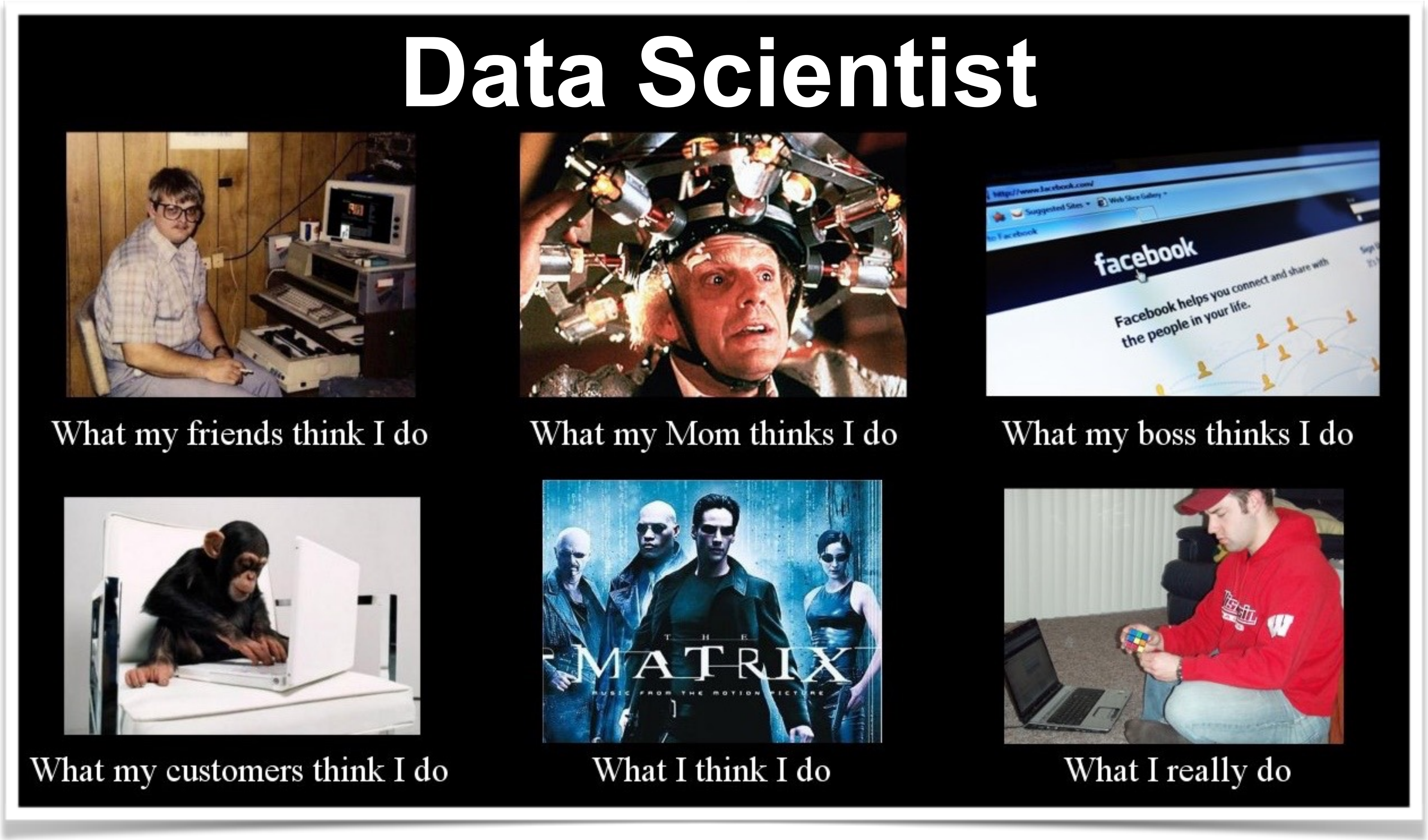
- Chapter lead data science (18 data scientists)

ING WBAA

- WBAA: Wholesale Banking Advanced Analytics
  - Wholesale Banking: caters to (big) companies.
  - We build data-driven AI products that benefit these clients and generate revenue / save costs for the bank.
  - In terms of data science, our projects are very diverse.
- Our team:
  - 120 people, 18 data scientists.
  - 4 countries (NL, UK, Poland, Romania)
  - 11 project teams

WHOLESALE BANKING
advanced
analytics

Harvard: "data scientist the sexiest job of the 21st century"


Data Scientist
facebook
Facebook helps you connect and share with the people in your life.
What my friends think I do
What my Mom thinks I do
What my boss thinks I do
THE MATRIX
What my customers think I do
What I think I do
What I really do

"We solve complex data puzzles"

Executive summary

In case you're considering to become a data scientist:

- Computer science: focus is often on highest performing algorithm.
- In physics you learn to extract insights from (complex) data.
  - Thorough: data understanding, covering systematic effects, completion of evidence gathering, defending your results.
- Complementary. Both are important!

Strong background in statistics & methodology is very useful!

Project examples

The bond market

- Bonds are loans issued by companies. Bonds can be traded.
  - Every company can issue multiple bonds with different characteristics (like maturity, rate, seniority).
- The bond market is a traditional market
  - $700 billion traded daily, 3 times more than stocks
  - 50% of the value is voice traded.
  - Market can be inefficient and illiquid
- Bonds are grouped into “universes”.

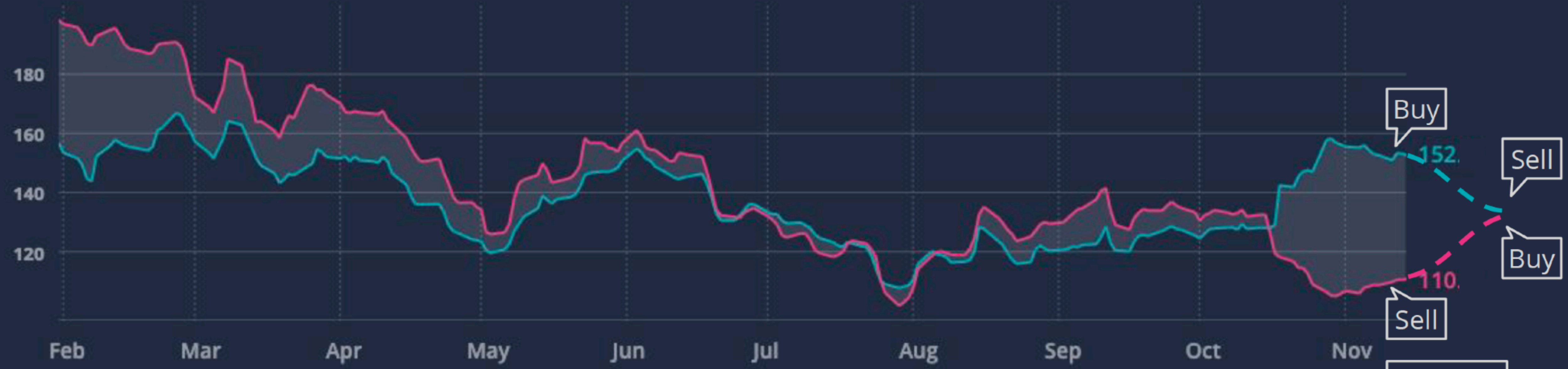
Bond pair trading

- Bonds can move together
    - Therefore market anomalies can appear between bonds.
- Look for a pair of bonds that are correlated and/or related to each other.
- Wait for an anomaly to appear, meaning that suddenly the difference between the two bonds increases.
- Assumption is that bonds will converge to each other again.
    - Called: mean reversion.

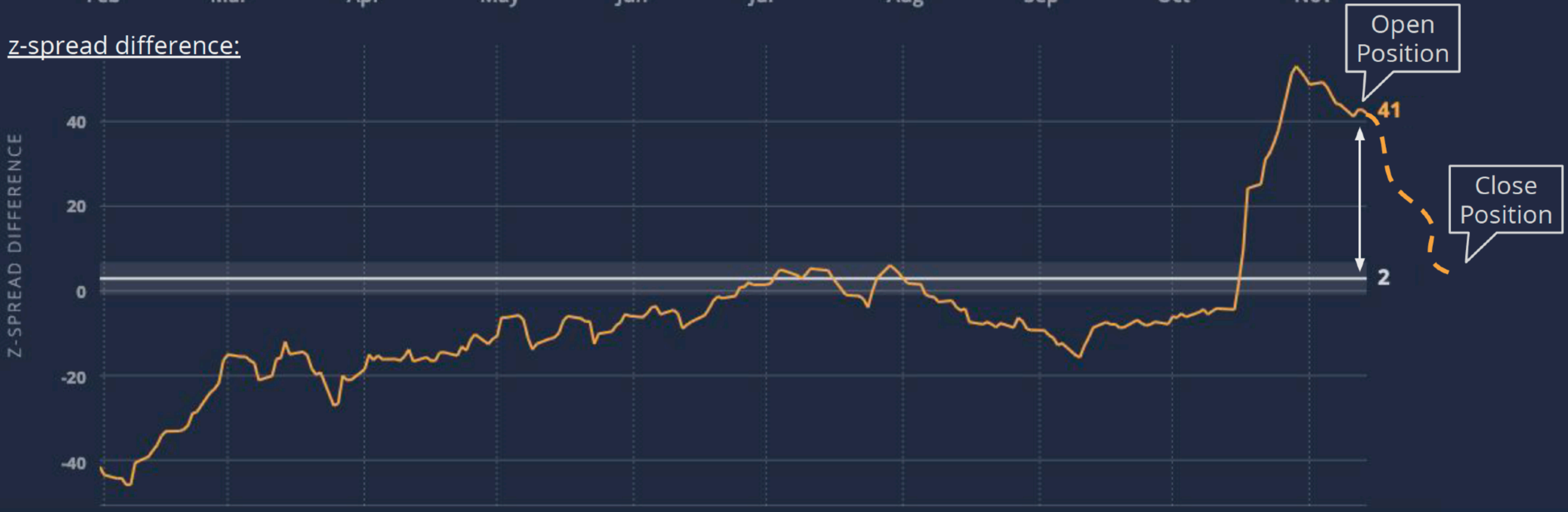
RENAUL 1 5/8 26/05/2026
FR0013334695 · France · BBB+

VW 2 1/2 31/07/2026
XS1944390597 · Germany · A-

z-spread of the 2 bonds:



z-spread difference:

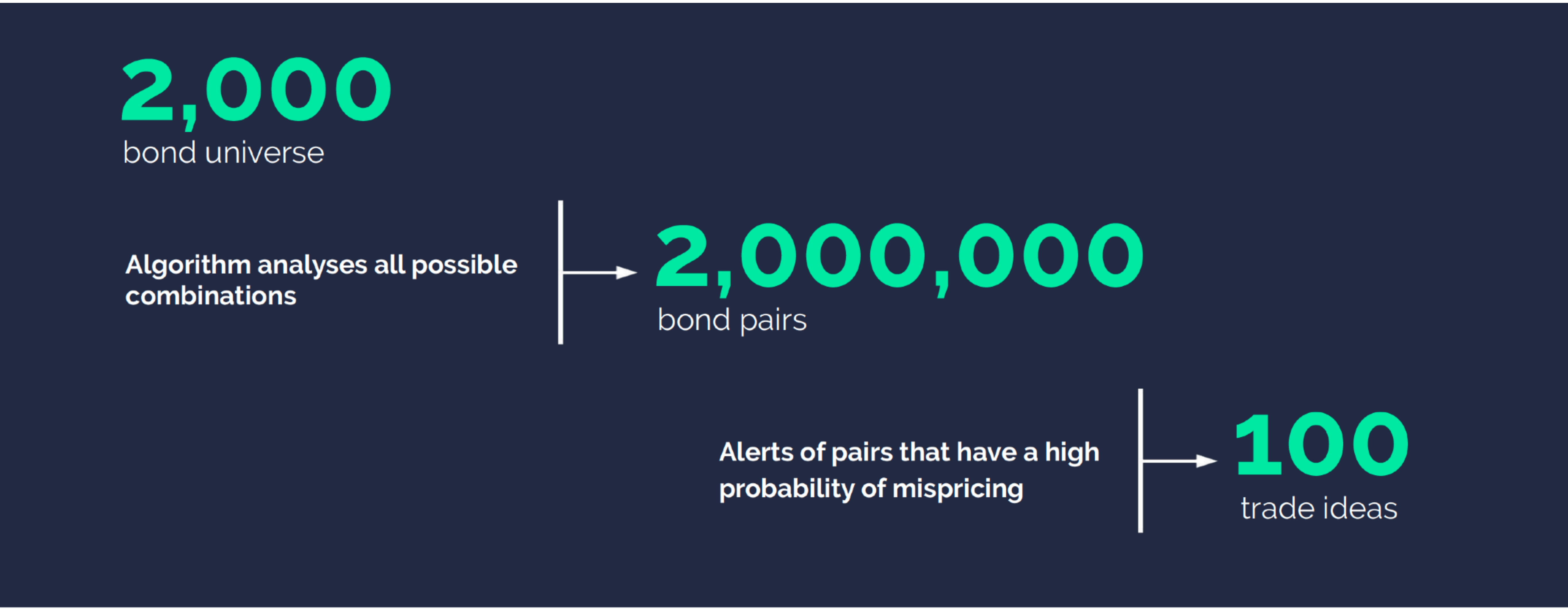


Katana Labs: detecting anomalies in relative value

- We built a tool that gives trading suggestions to asset managers based on relative value of bond pairs.
  - It is not (!) a pure auto trading.
- There is trade-off between the number of alerts and profitability
- The trick is: alerting enough in advance.

- Katana Labs spun out into a separate company last year.

A filtering exercise



2,000
bond universe

Algorithm analyses all possible combinations

$\longrightarrow$ **2,000,000**
bond pairs

Alerts of pairs that have a high probability of mispricing

$\longrightarrow$ **100**
trade ideas

Data science problem statement

How to perform:

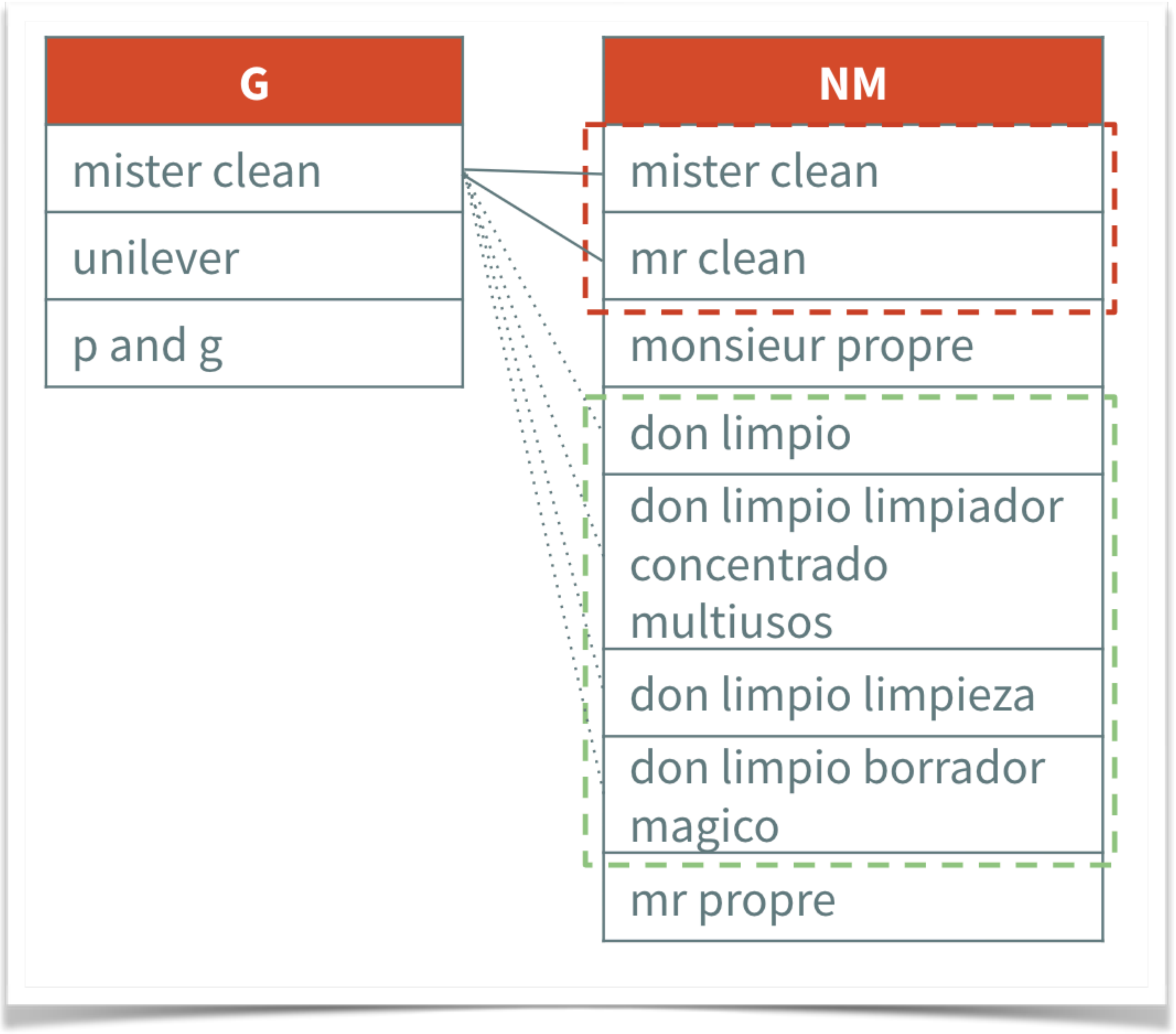
1. Identifying relevant pairs of bonds
2. Detecting anomalies
3. When to open position
4. When to close position, after mean reversion happened

Name matching

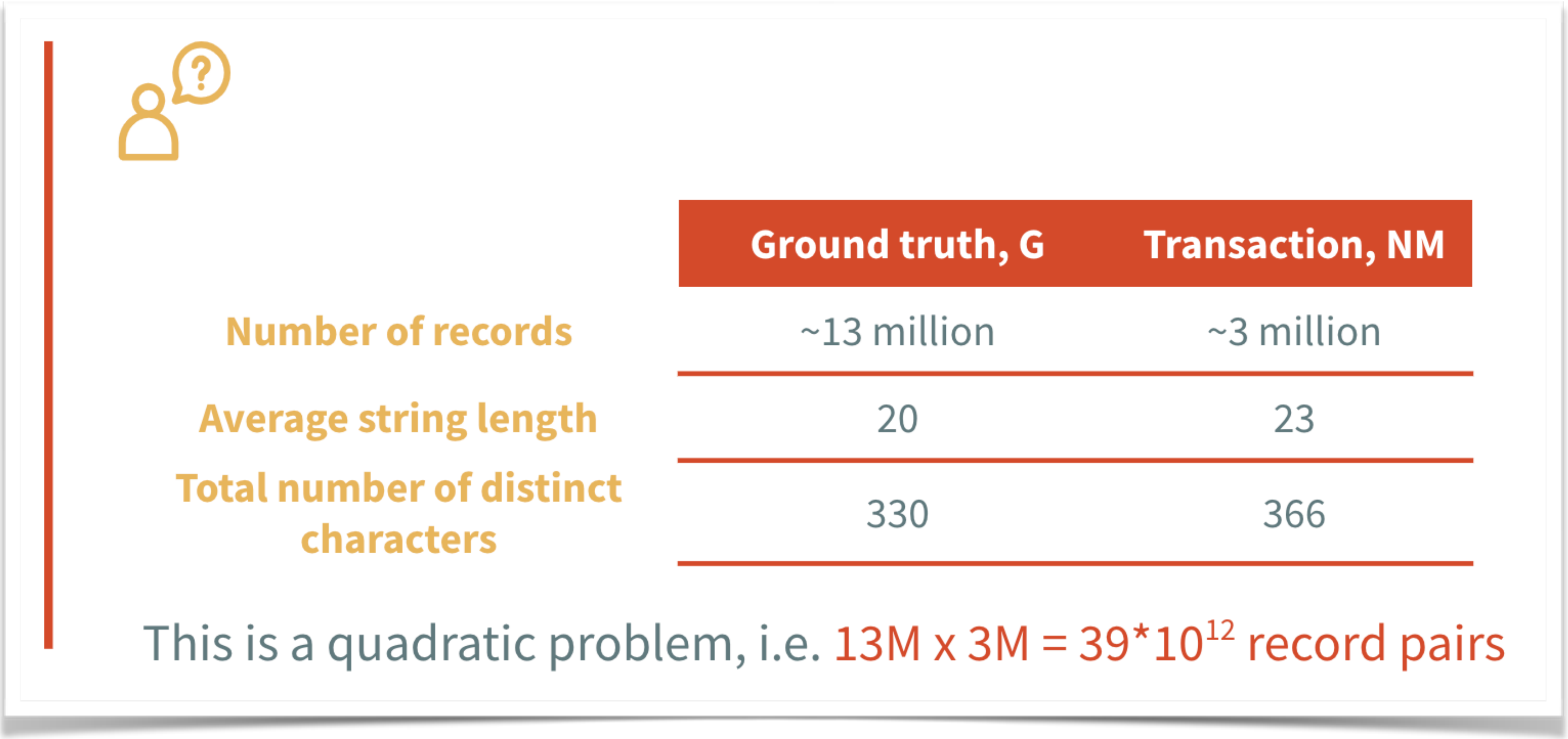
- Wholesale banking deals with corporate clients.
    - Bank wants a holistic view of these clients.
- **One way to enrich: look at names on in-/outgoing transactions**
- Two use cases:
    - Match external bank accounts to ING accounts
    - Match (high-risk) names to international watch list(s)

Reality

- Significantly different names used for same entity
  - company names / owner names
  - abbrev. / wrong field / generic name
  - entities in different countries
  - different companies under same entity
- Mismatch with ground-truth names when doing entity matching on full set.



Why name matching at scale?



	Ground truth, G	Transaction, NM
Number of records	~13 million	~3 million
Average string length	20	23
Total number of distinct characters	330	366

This is a quadratic problem, i.e. 13M x 3M = $39*10^{12}$ record pairs

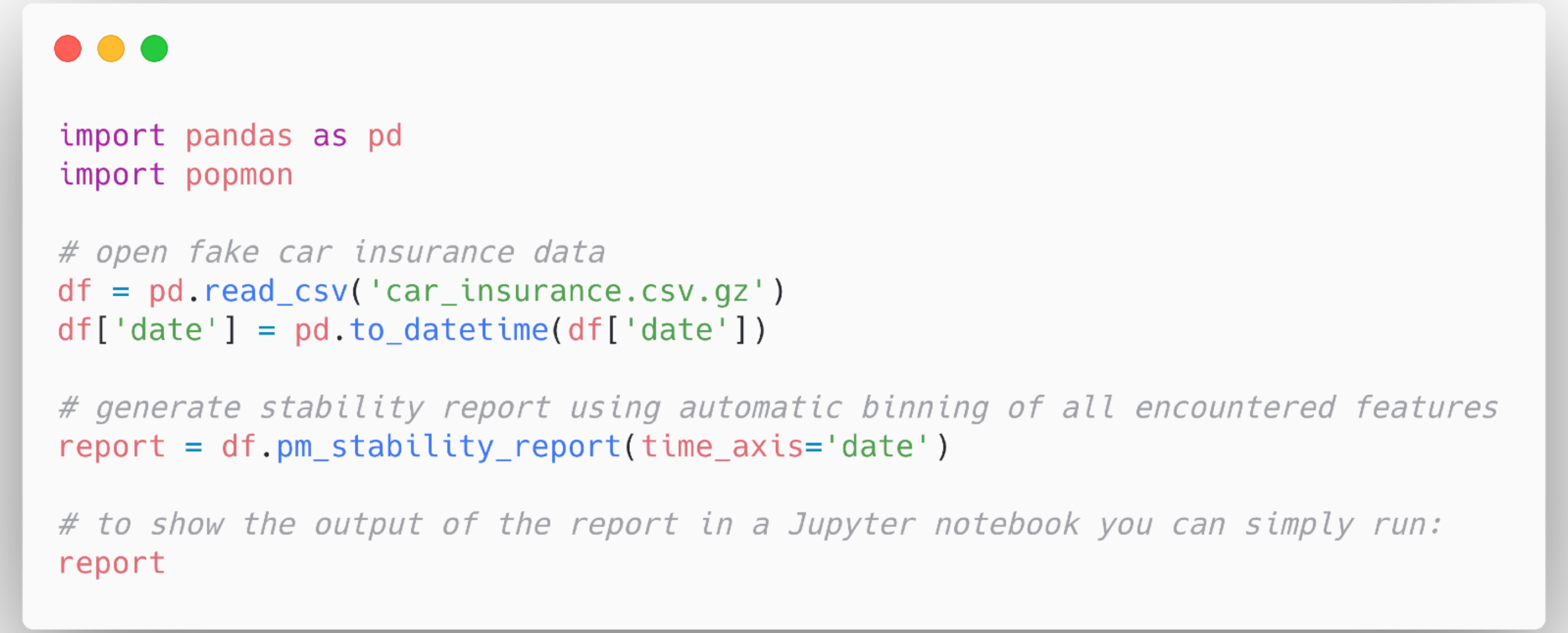
Data science problem statement

How to perform:

1. Preselection of relevant name pairs - lot of filtering.
2. Identifying *correct* name pairs.
3. Correct for differences between ING and non-ING data.

popmon - population shift monitoring made easy

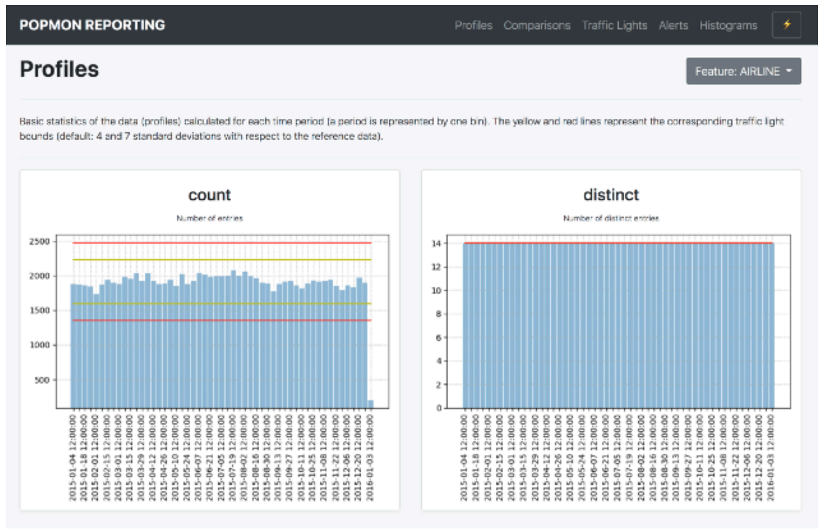
- pip install popmon
- Use popmon to monitor the stability of a pandas or spark dataset
- Automatically detect changes over time from trends, shifts, peaks, outliers, anomalies, correlations, etc.
    - Supports numerical, ordinal, categorical features.
- Alerting based on static or dynamic business rules.



```python
import pandas as pd
import popmon

# open fake car insurance data
df = pd.read_csv('car_insurance.csv.gz')
df['date'] = pd.to_datetime(df['date'])

# generate stability report using automatic binning of all encountered features
report = df.pm_stability_report(time_axis='date')

# to show the output of the report in a Jupyter notebook you can simply run:
report
```

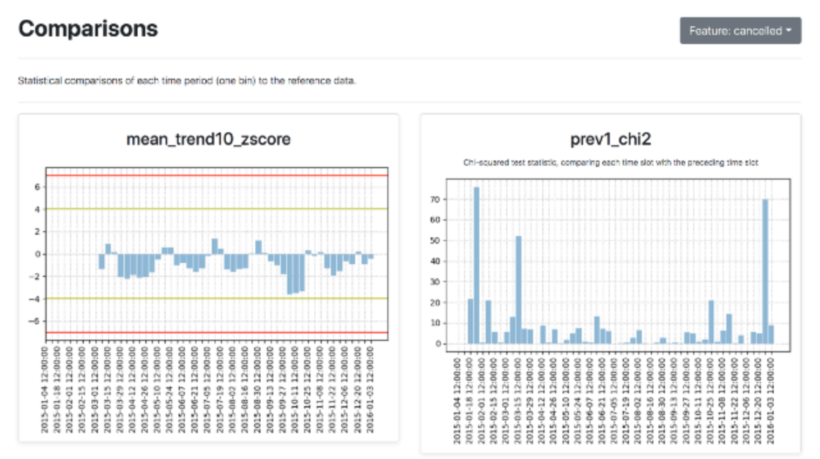
Example

https://crclz.com/popmon/reports/flight_delays_report.html



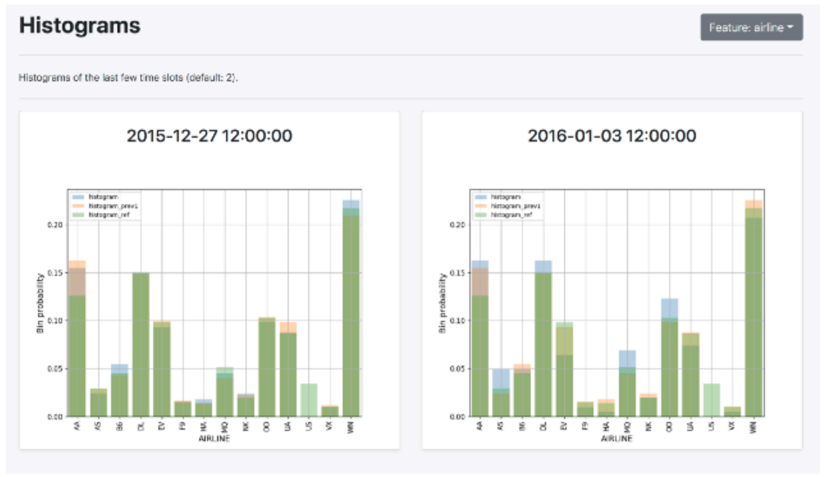
Profiles

Basic statistics of the data (profiles) calculated for each time period (a period is represented by one bin). The yellow and red lines represent the corresponding traffic light bounds (default: 4 and 7 standard deviations with respect to the reference data).


Feature: cancelled
mean_trend10_zscore
prev1_chi2
Chi-squared test statistic, comparing each time slot with the preceding time slot

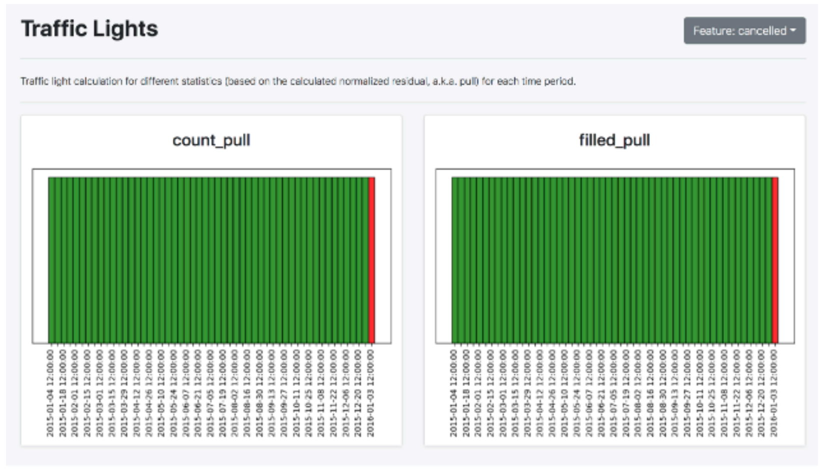
Comparisons

Statistical comparisons of each time period (one bin) to the reference data.



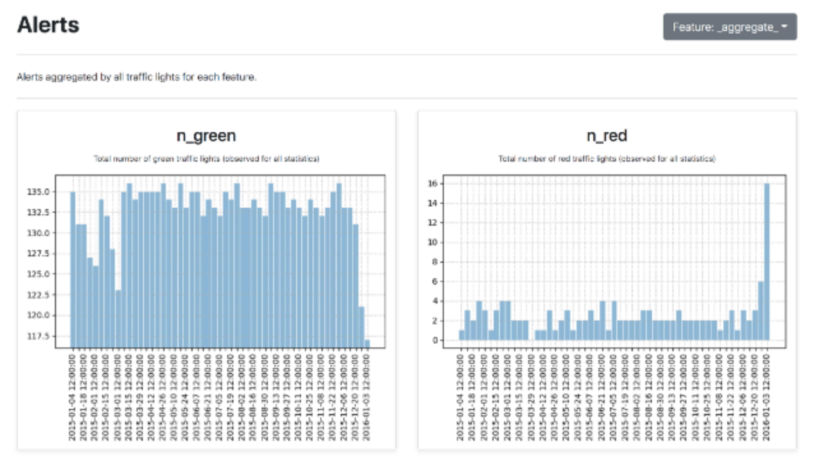
Histograms

Histograms of the last few time slots (default: 2).



Traffic Lights

Traffic light calculation for different statistics (based on the calculated normalized residual, a.k.a. pull) for each time period.


Feature: _aggregate_
n_green
Total number of green traffic lights (observed for all statistics)
n_red
Total number of red traffic lights (observed for all statistics)

Alerts

Alerts aggregated by all traffic lights for each feature.

popmon

Data modelling and synthesis

Data science problem statement

- Want to synthesize fake data that is indistinguishable from real data.
  - Generating pseudo-experiments
  - Sharing sensitive data.
- In order to synthesize fake data, we need to model the actual data first.

Modelling problem

- Modelling of high-dimensional datasets is complex due to correlations.
  - High-dimensional: 3+ features.
- Several techniques for that, e.g. histograms, tabular-GAN, bayesian networks, but not directly applicable.
  - For example, hard to train, very slow.
- We had an interesting new idea we wanted to try out …

General proposition

A. Transform all features to a “uniformly” filled hypercube.

- All transformations are inverse-able.
- As much structure/modes/correlation is taken out of data already, put into transformers.

B. Then, the residual structure in data is modelled.

Variable types

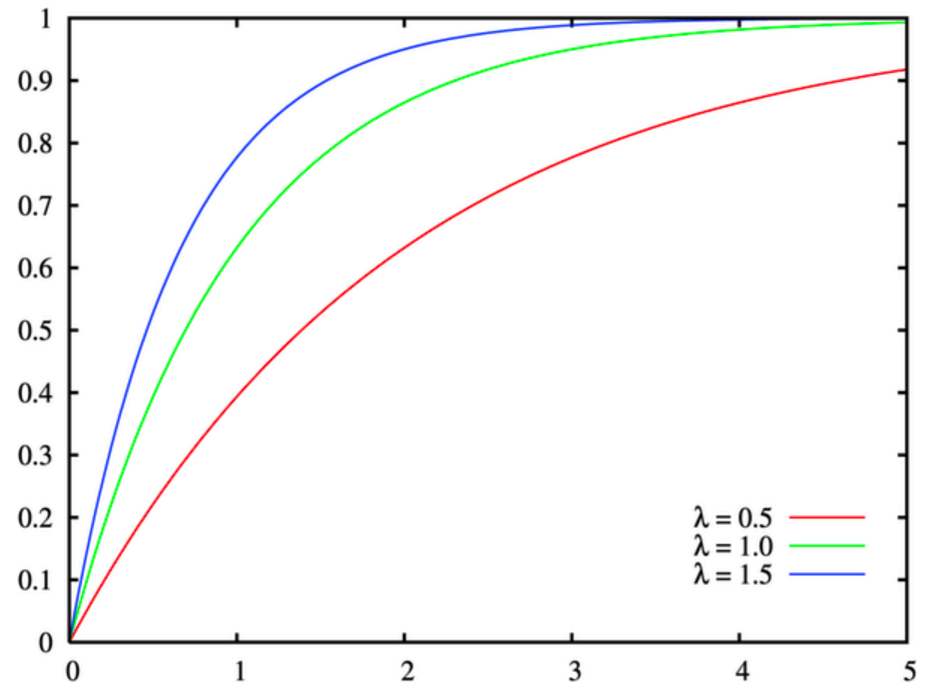
- Modelling method works for all variable types.
- Will focus here on continuous variables only.

The method

1. Quantile transformation to normal distribution
2. Linear decorrelation (PCA)
3. Quantile transformation to uniform distribution
4. Non-linear feature ordering
5. Machine learning classifier

Quantile transformation

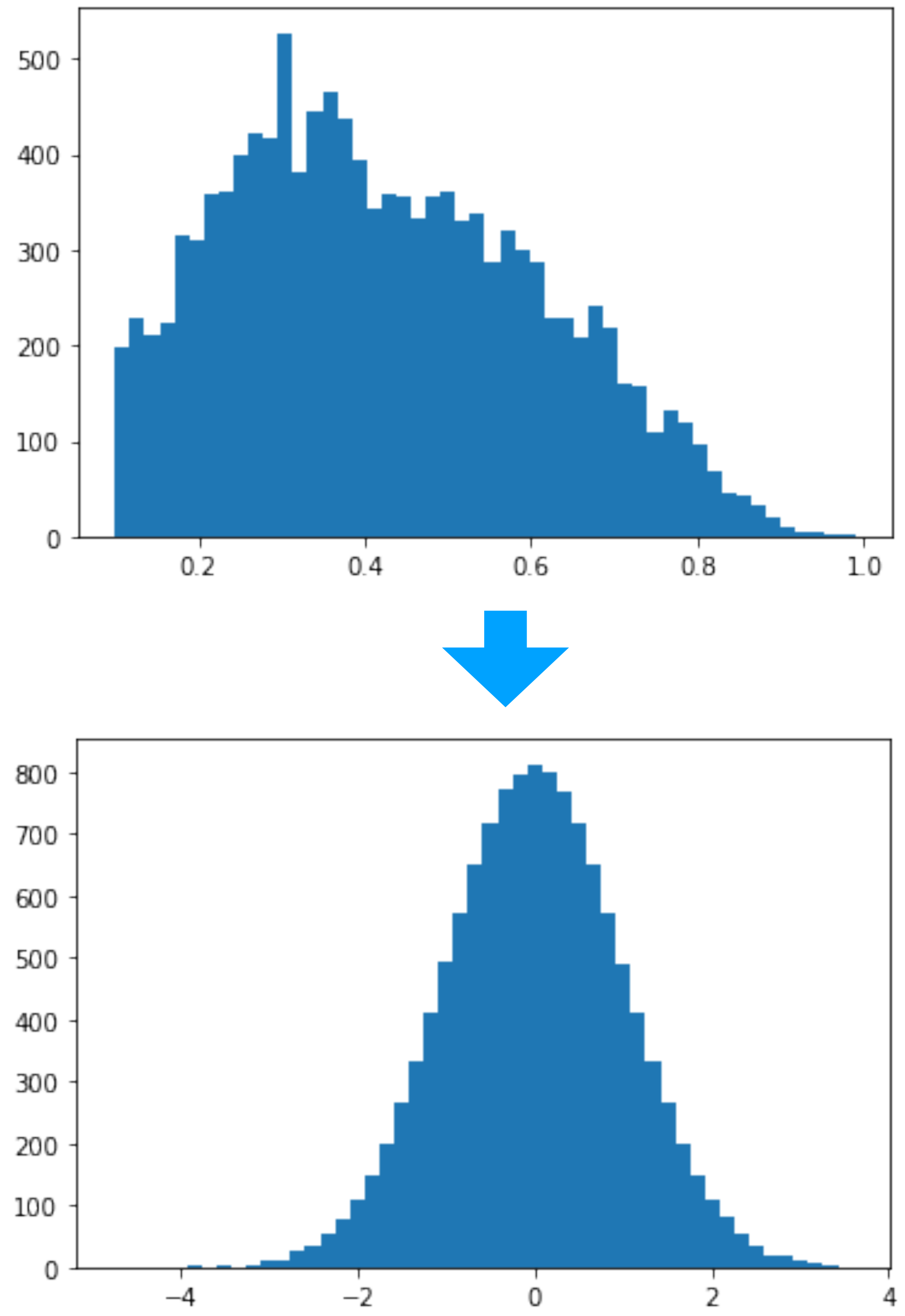
- The **cumulative distribution function** (**CDF**) of a real-valued random variable X evaluated at x, is the probability that X will take a value less than or equal to x.
- I.e. the CDF is the integral of the PDF.
- When transforming each value X to its CDF value, one gets a uniform distribution b/n 0 and 1.
- Transformation is invertible.



CDF of exponential distribution.

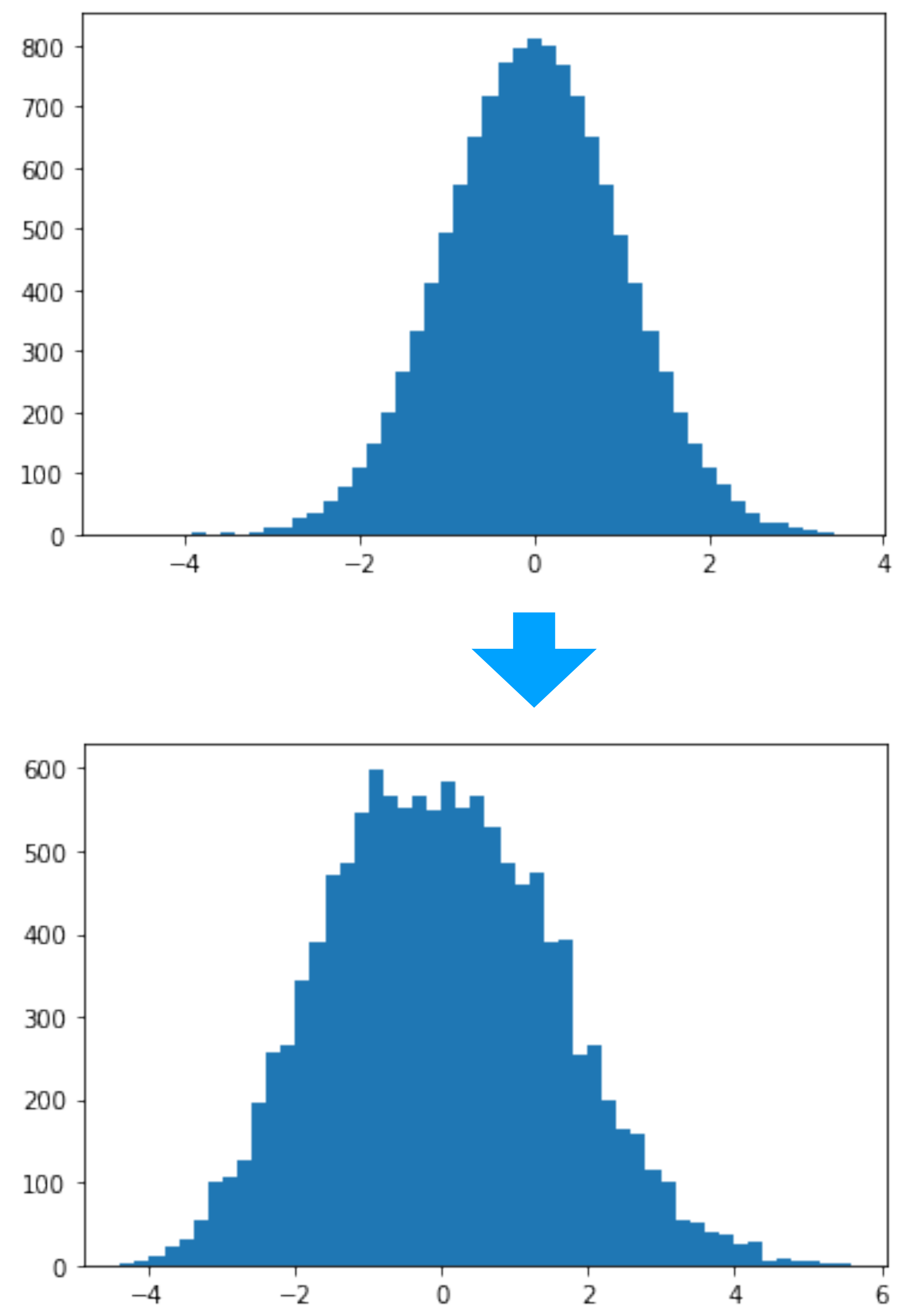
1. quantile transformation to normal

- Each feature is quantile transformed to a normal distribution.
  - Can apply smoothing is needed.
- Per feature, describe the transformation mathematically with a Jacobian:
  - Inverse Jacobian = (marginalized 1-dim pdf) / (normal distribution)
- We have now entered the “Copula” space.



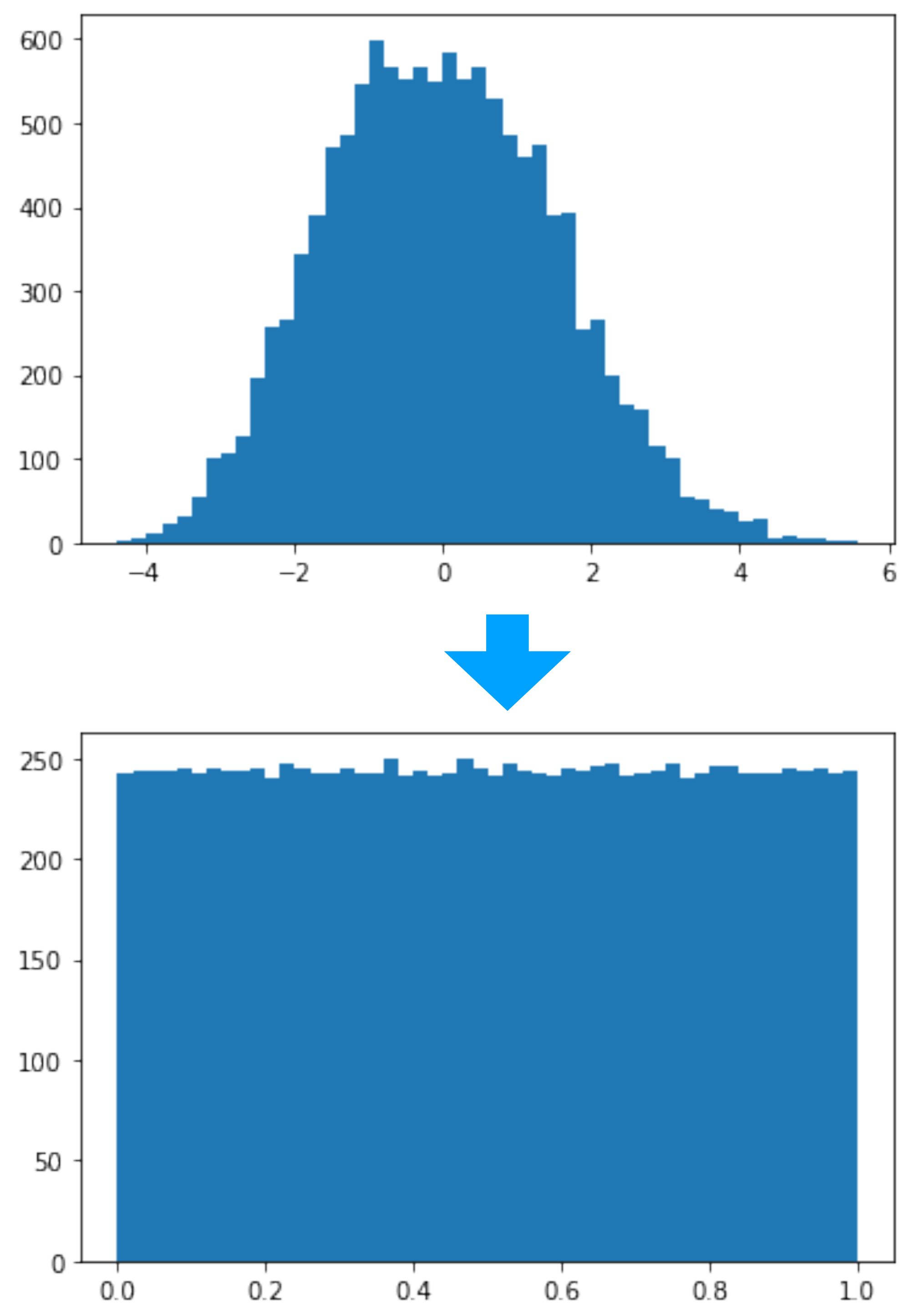
2. Linear decorrelation (PCA)

- In theory, PCA works best for normal distributions.
- Apply PCA to Copula distributions.
- # components = # features
- This takes out the linear correlations.
- PCA applies a rotation, so (inverse) Jacobian = 1.



3. quantile transformation to uniform

- For normally distributed input features with non-linear correlations, applying PCA rotation gives as output normal-like, “distorted” distributions.
- I apply another KDE quantile transformation to make each PCA-feature uniformly distributed.
- Inverse Jacobian = (marginalized 1-dim pdf after PCA)



4. non-linear feature ordering

- We can order the transformed features in two ways:

1. Based on PCA importance (eg. > 99%).

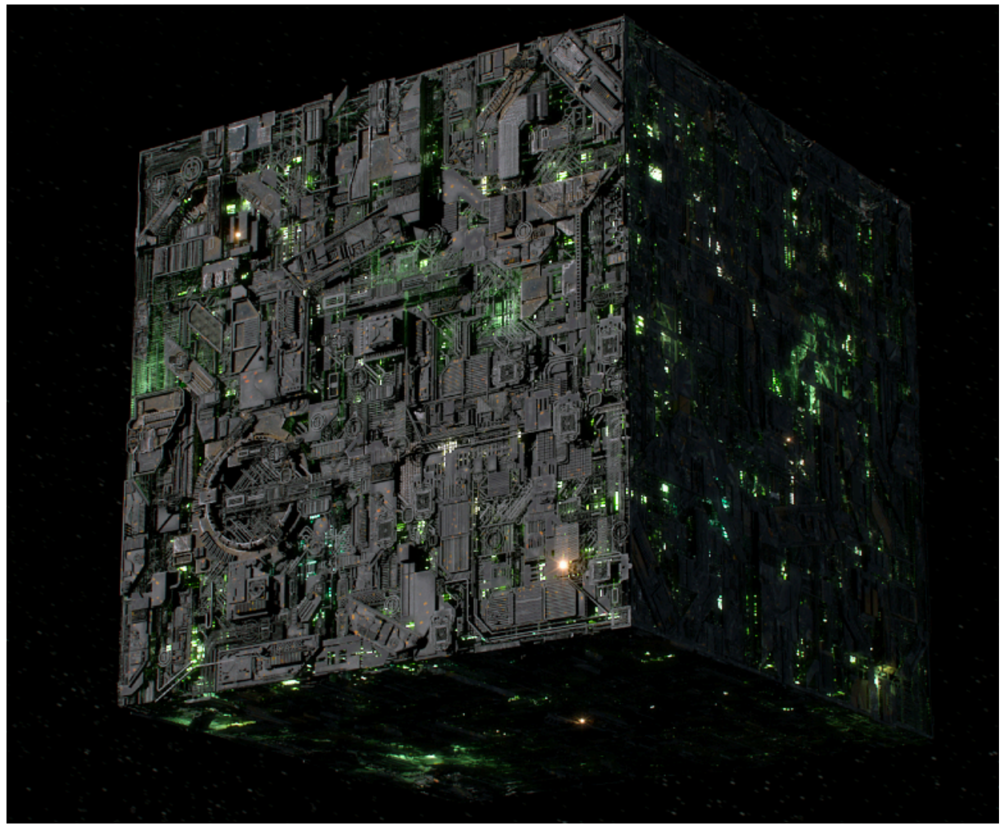
2. Based on ranking of mutual information of feature pairs. (eg. MI > 0.05)

- If so desired, only consider top-n features for further non-linear modelling.

- Or keep all of them.

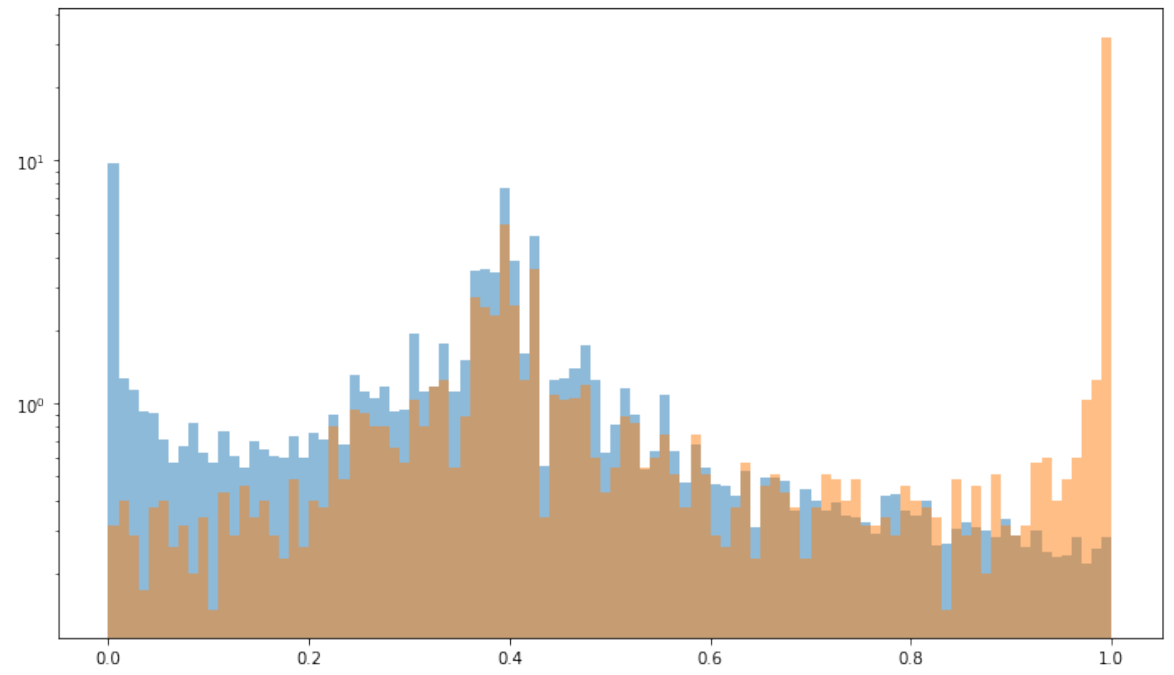
Intermezzo

- Dataset is transformed into "uniformly" filled hypercube.
    - Each marginalized distribution is uniform.
    - No more linear correlations.
- Dataset still contains residual non-linear correlations.
- Seen as local pockets of non-uniform density.
    - How to describe these?



5. Adversarial training

- Generate uniform sample of same size (= class 0).
- Train a ML classifier to distinguish between the transformed dataset (class 1) and a uniform dataset (class 0).
    - Many non-linear classifiers will do.
        - Important not to overtrain — need well-calibrated probabilities.
- cube density(x) = prob(x|1) / prob(x|0)
- Note that classifier focusses *fully* on non-linear structures.



KDECopulaNNPdf

- Sklar's theorem:

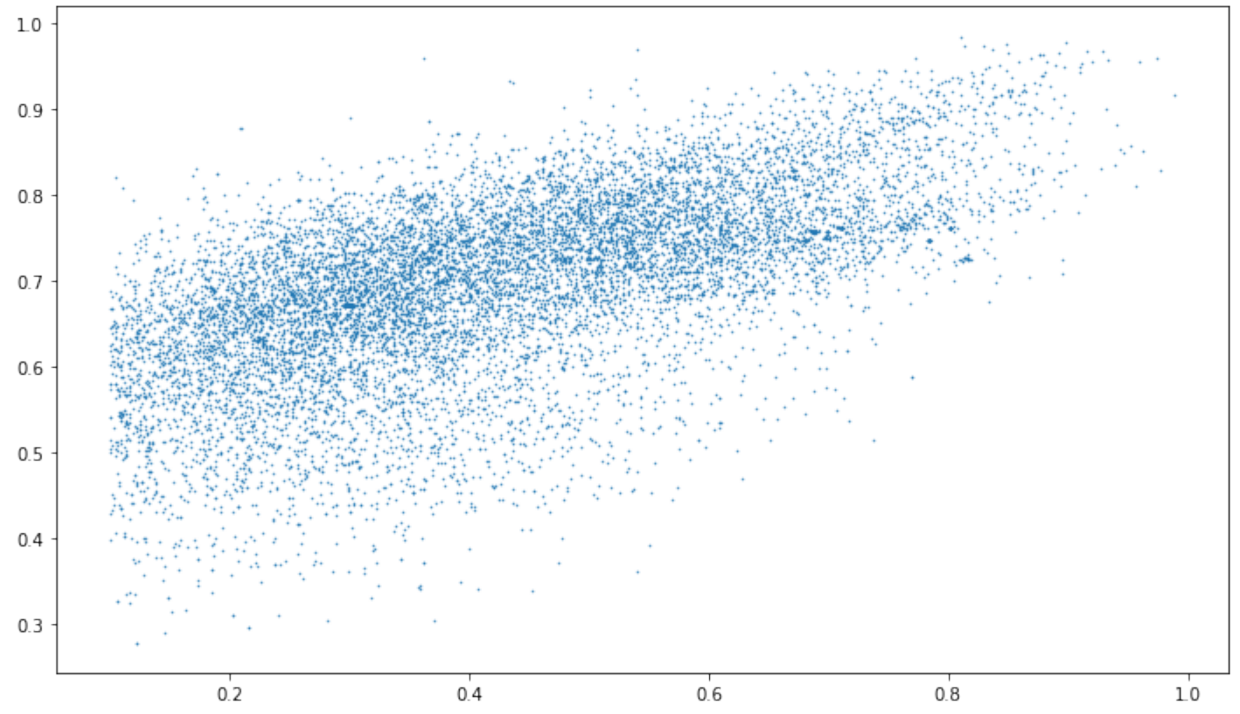
- $$\mathrm{pdf}(x) = \left[\prod_i \frac{kq1(x_i)}{G(x_{\mathrm{normal},i})} \cdot 1 \cdot kq2(x_{\mathrm{pca},i}) \right] \frac{p(x_{\mathrm{uniform}}|\text{transformed data})}{p(x_{\mathrm{uniform}}|\text{uniform data})}$$

Synthesizing fake data

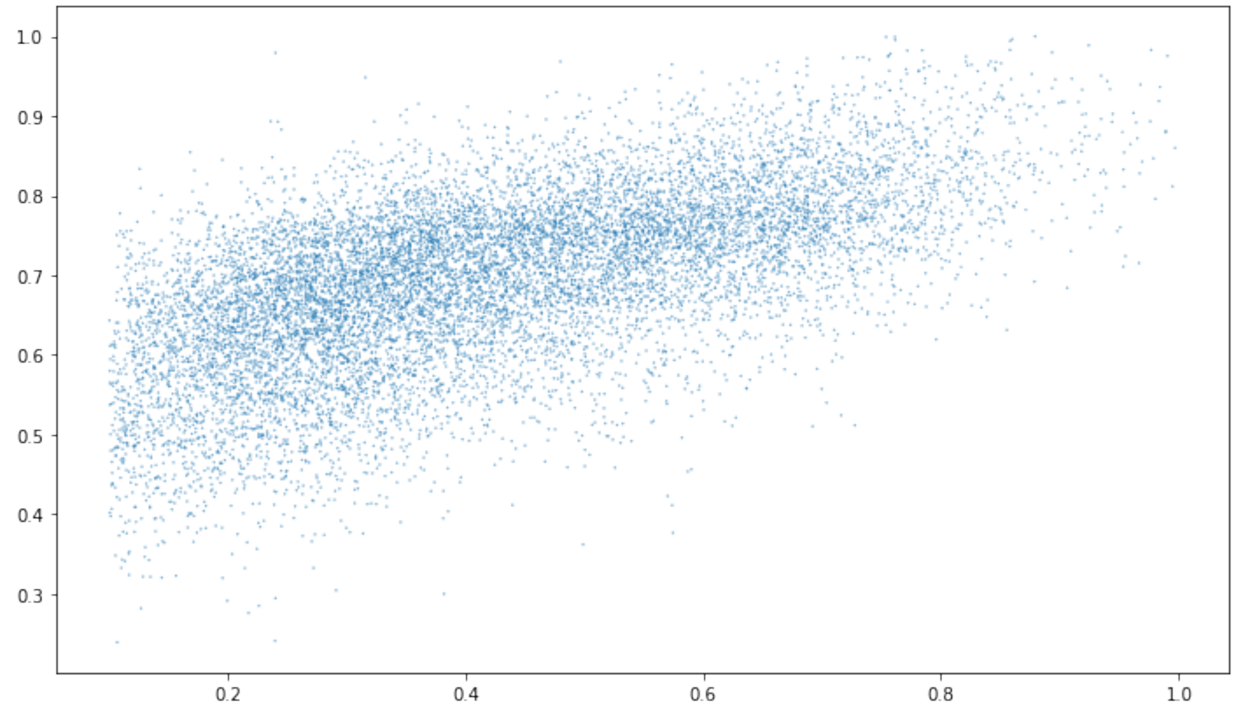
1. Generate data points {x} uniformly in hypercube.
2. Give weight to each data point: $w(x) = \text{prob}(x|1) / \text{prob}(x|0)$
3. Apply inverse transformations to each data point.

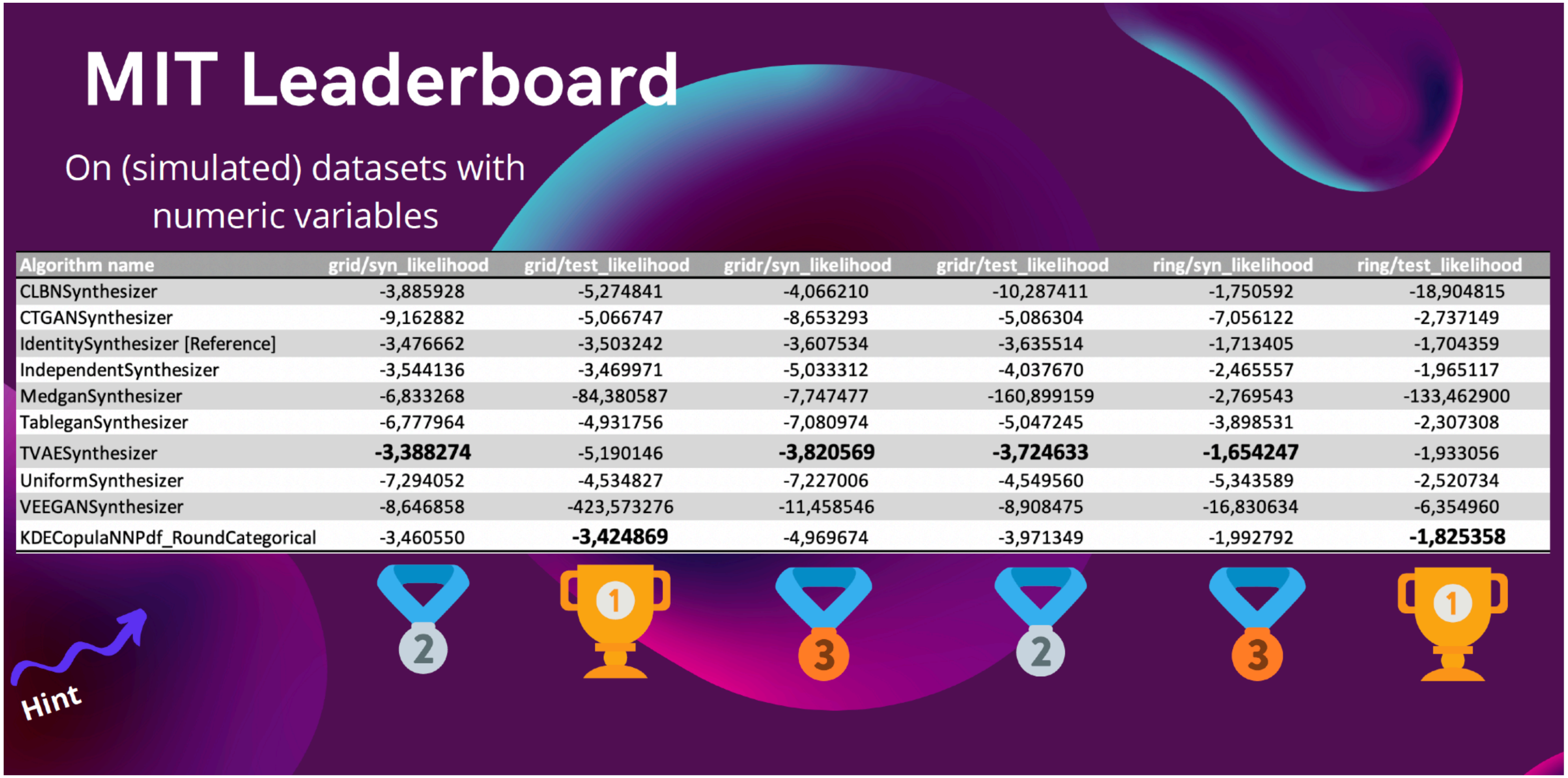
Non-linear correlations

Real data



Synthetic data





MIT Leaderboard

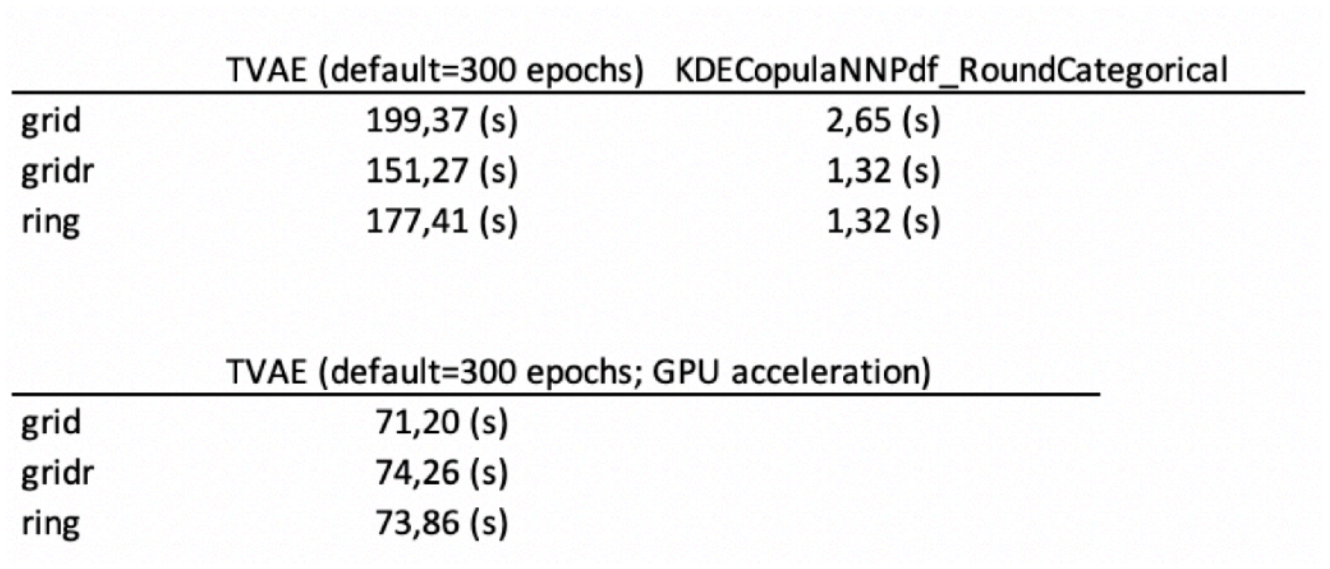
On (simulated) datasets with numeric variables

Algorithm name	grid/syn_likelihood	grid/test_likelihood	gridr/syn_likelihood	gridr/test_likelihood	ring/syn_likelihood	ring/test_likelihood
CLBNSynthesizer	-3,885928	-5,274841	-4,066210	-10,287411	-1,750592	-18,904815
CTGANSynthesizer	-9,162882	-5,066747	-8,653293	-5,086304	-7,056122	-2,737149
IdentitySynthesizer [Reference]	-3,476662	-3,503242	-3,607534	-3,635514	-1,713405	-1,704359
IndependentSynthesizer	-3,544136	-3,469971	-5,033312	-4,037670	-2,465557	-1,965117
MedganSynthesizer	-6,833268	-84,380587	-7,747477	-160,899159	-2,769543	-133,462900
TableganSynthesizer	-6,777964	-4,931756	-7,080974	-5,047245	-3,898531	-2,307308
TVAESynthesizer	**-3,388274**	-5,190146	**-3,820569**	**-3,724633**	**-1,654247**	-1,933056
UniformSynthesizer	-7,294052	-4,534827	-7,227006	-4,549560	-5,343589	-2,520734
VEEGANSynthesizer	-8,646858	-423,573276	-11,458546	-8,908475	-16,830634	-6,354960
KDECopulaNNPdf_RoundCategorical	-3,460550	**-3,424869**	-4,969674	-3,971349	-1,992792	**-1,825358**
	2	1	3	2	3	1

(also doing very well on categorical variables)

Algorithm speed

- Very fast compared with competition!

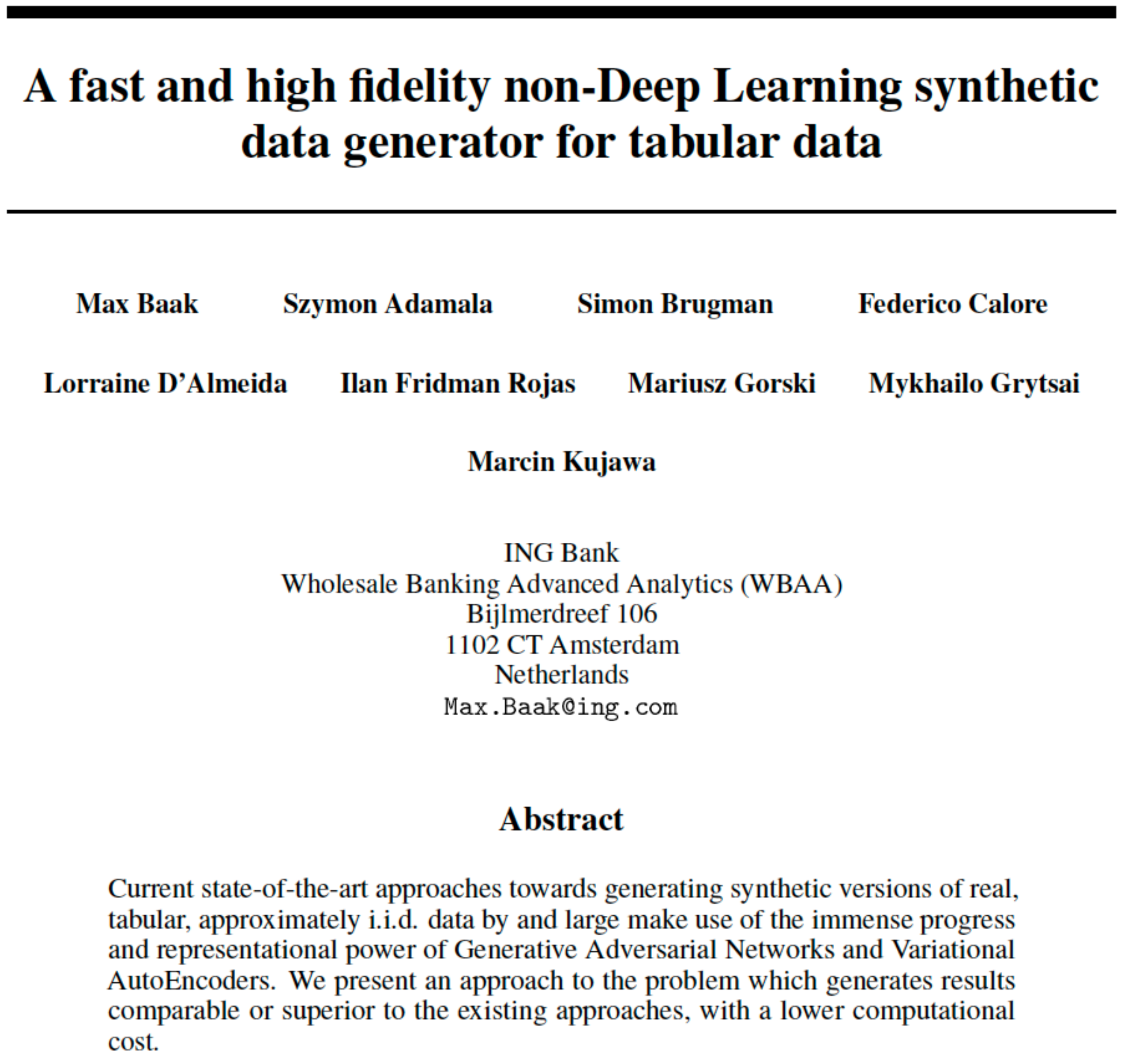


	TVAE (default=300 epochs)	KDECopulaNNPdf_RoundCategorical
grid	199,37 (s)	2,65 (s)
gridr	151,27 (s)	1,32 (s)
ring	177,41 (s)	1,32 (s)

	TVAE (default=300 epochs; GPU acceleration)
grid	71,20 (s)
gridr	74,26 (s)
ring	73,86 (s)

Research paper

- Algorithm: transformations & a binary classifier.
- Would have been happy to share the code, but not yet in a good enough state :-)
- Hope you can use it!



A fast and high fidelity non-Deep Learning synthetic data generator for tabular data

Max Baak **Szymon Adamala** **Simon Brugman** **Federico Calore**
Lorraine D'Almeida **Ilan Fridman Rojas** **Mariusz Gorski** **Mykhailo Grytsai**
Marcin Kujawa

ING Bank
Wholesale Banking Advanced Analytics (WBAA)
Bijlmerdreef 106
1102 CT Amsterdam
Netherlands
Max.Baak@ing.com

Abstract

Current state-of-the-art approaches towards generating synthetic versions of real, tabular, approximately i.i.d. data by and large make use of the immense progress and representational power of Generative Adversarial Networks and Variational AutoEncoders. We present an approach to the problem which generates results comparable or superior to the existing approaches, with a lower computational cost.

Contact

- max.baak@ing.com
- Or contact me on LinkedIn