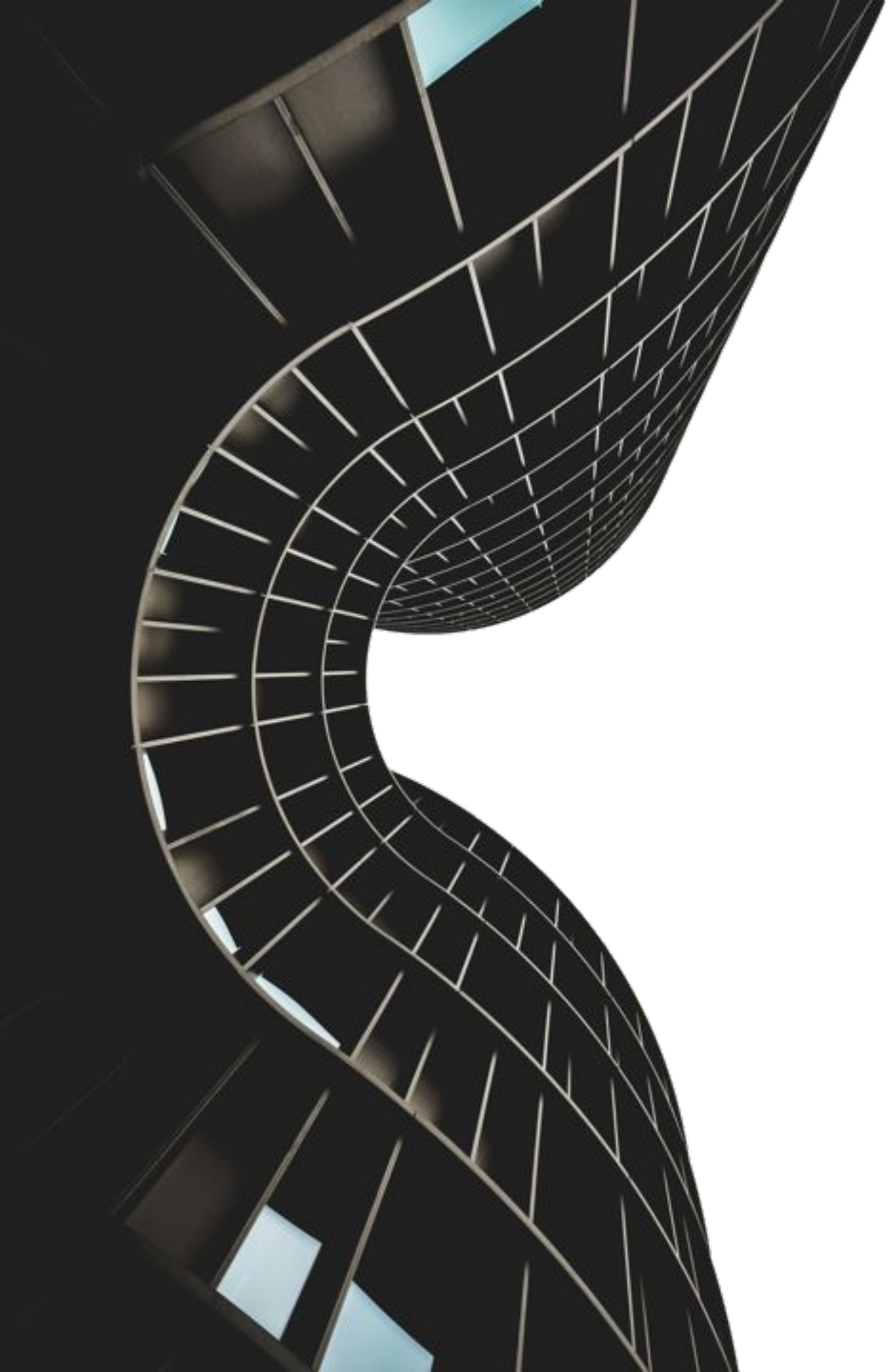


feedzai

The Forefront of Fraud Fighting

Miguel Araújo
Senior Data Scientist
miguel.araujo@feedzai.com

2018-09-04

feedzai

ABOUT FEEDZAI

QUICK FACTS

- 320+ employees and growing
- Founded by data scientists and aerospace engineers
- 20% of top 25 world banks (excl. China)
- Series C funded: $82M raised to date
- Headquarters in Portugal with offices in Silicon Valley, New York City, Atlanta, Hong Kong, London, Lisbon, Coimbra and Porto.

MISSION

KEEP COMMERCE SAFE & CREATE A BETTER CUSTOMER EXPERIENCE THROUGH MACHINE LEARNING

INVESTORS

OAK HC/FT | Financial Technology Partners | Capital One | Sapphire Ventures | citi Ventures

WHAT OTHERS SAY

The U.S. market fraud prevention just got a new player.	Ranked as a cool technology to watch.	Startups that are owning the data game.	Feedzai's machine learning is the next wave.	Payment Card Management: Essential tools for U.S. card issuers
Forbes	Gartner	Entrepreneur Magazine	Forrester	Aite

Source: itgovernance.co.uk

How fraud starts: Skimming

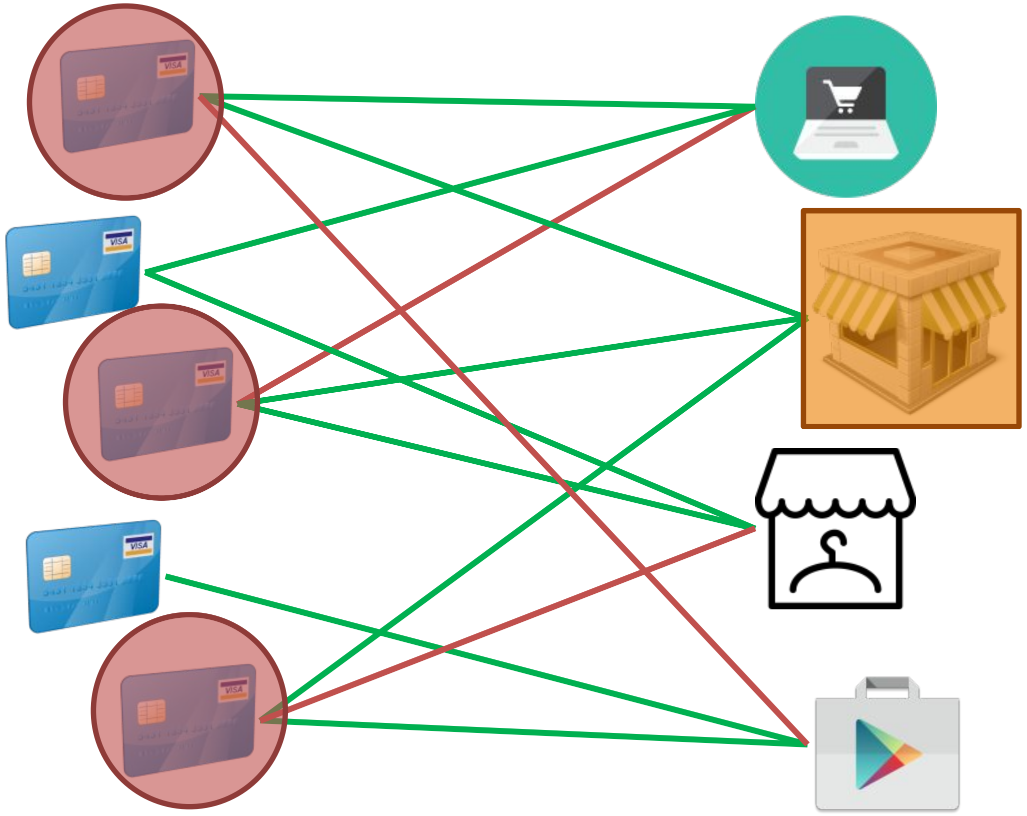
https://www.youtube.com/watch?v=DIKi1URjiwA

AGENDA

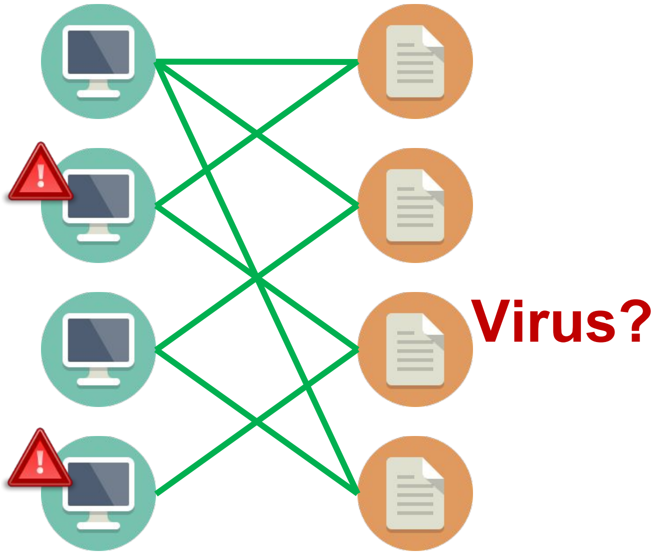
1. Detecting Points-of-Compromise
2. Feature Engineering at Scale
3. Automatic Machine Learning
4. Deep Learning
5. Reject Inference and Counterfactual Analysis

BreachRadar: Detecting Points-of-Compromise

Imagine you work at Feedzai



Given: Bipartite graph and "victim nodes".
Find: "Infectious nodes".

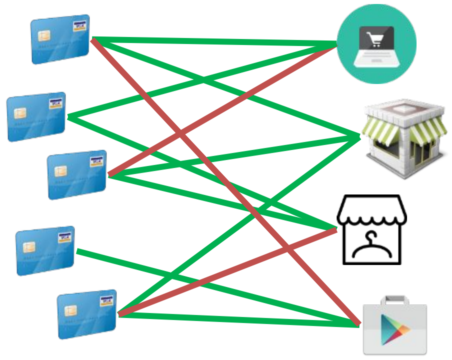


Given:

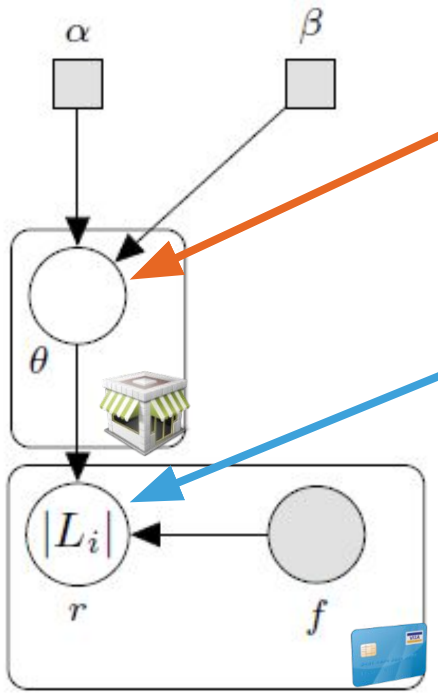
- > 100M credit cards, > 1B transactions;
- Fraud labels.

Find:

The most likely Points-of-Compromise.



Problem: Cards/locations with many transactions influence results disproportionately.

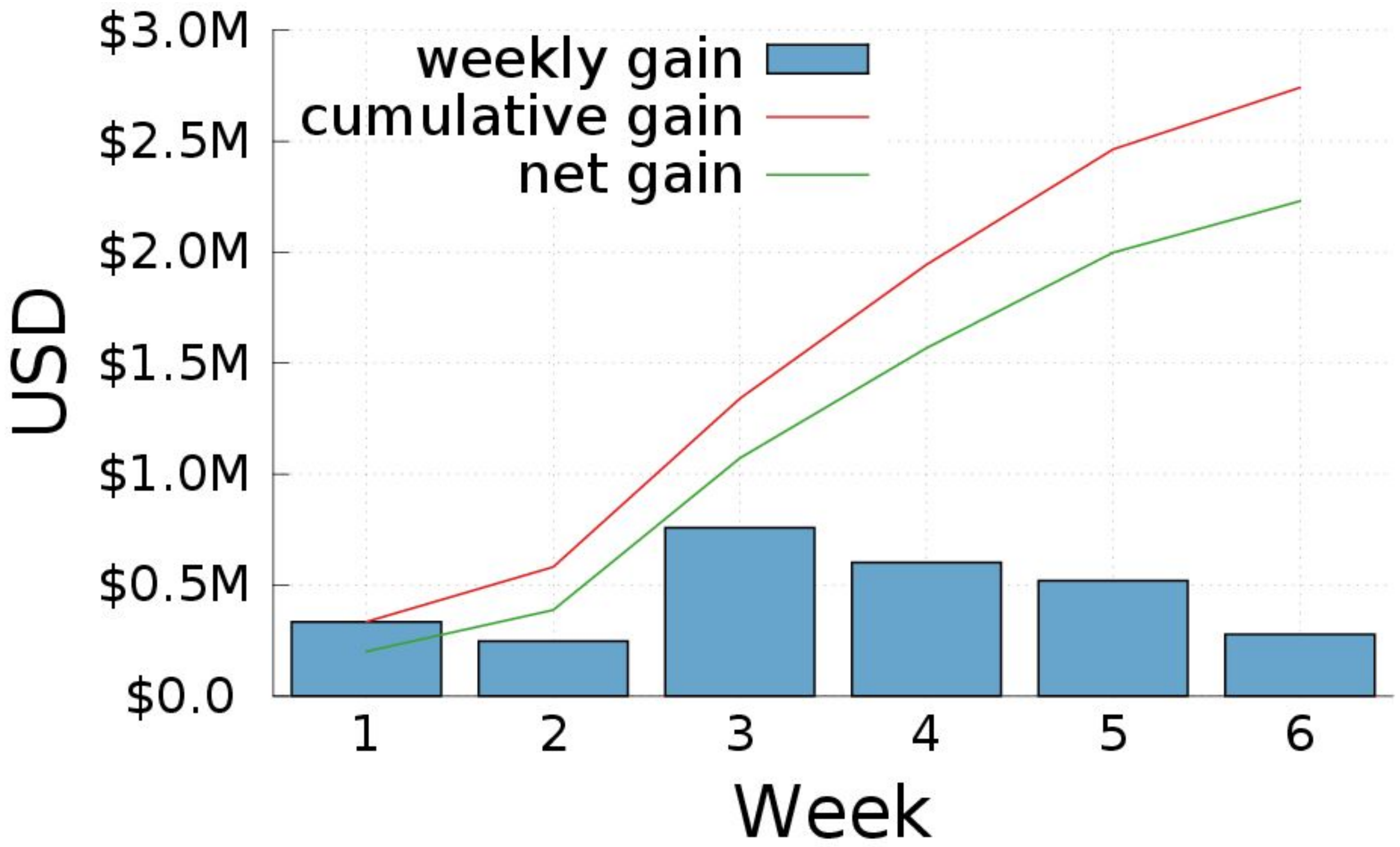


Probability of being compromised.

Blame each compromised card assigns to each merchant.

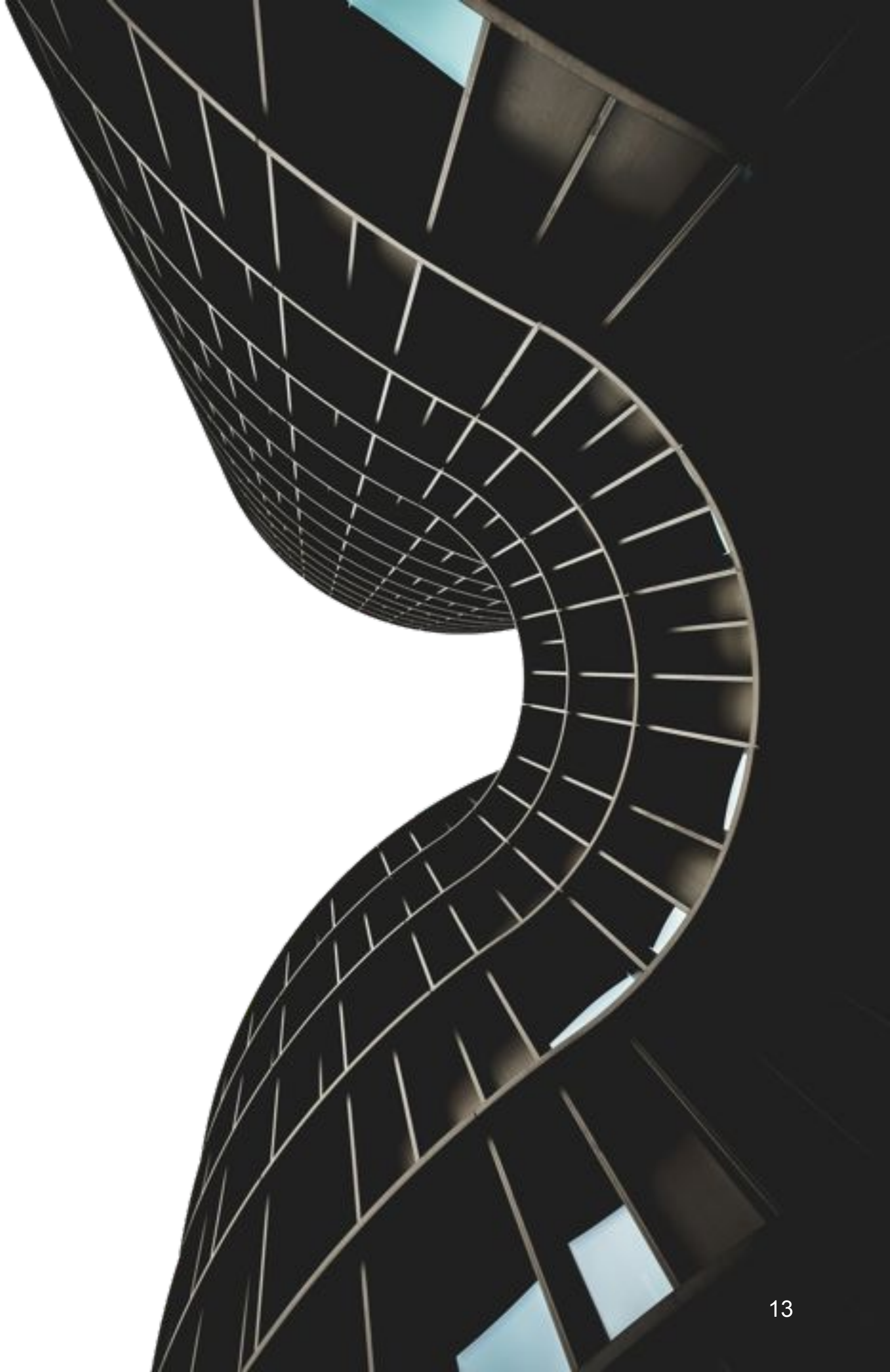
Alternating Optimization: cards *agree* on most likely mutual Points-of-Compromise.

feedzai

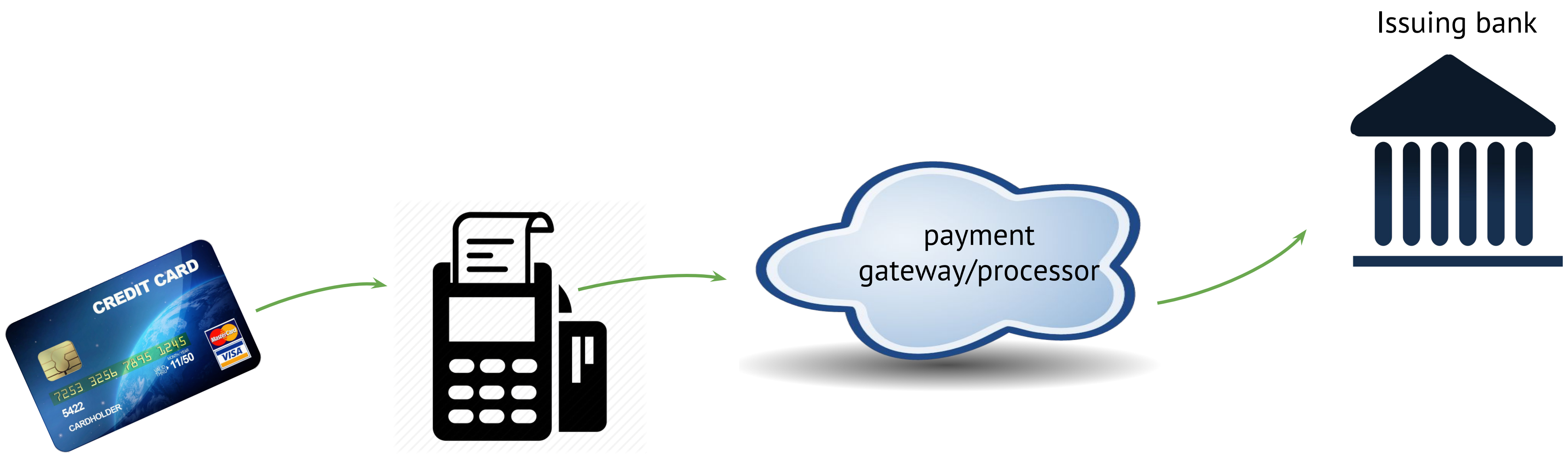

weekly gain
cumulative gain
net gain
USD
$3.0M
$2.5M
$2.0M
$1.5M
$1.0M
$0.5M
$0.0
1
2
3
4
5
6
Week

feedzai

Lightweight Profiles

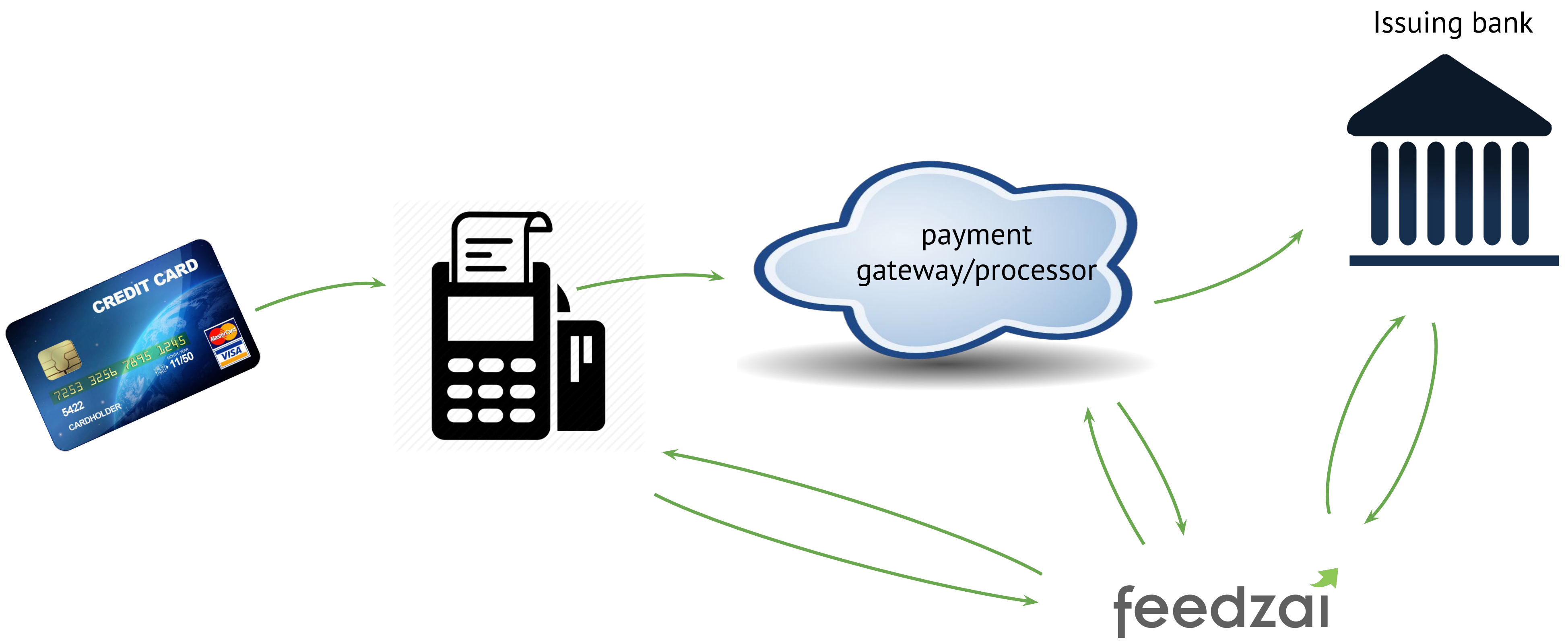


Life of a transaction



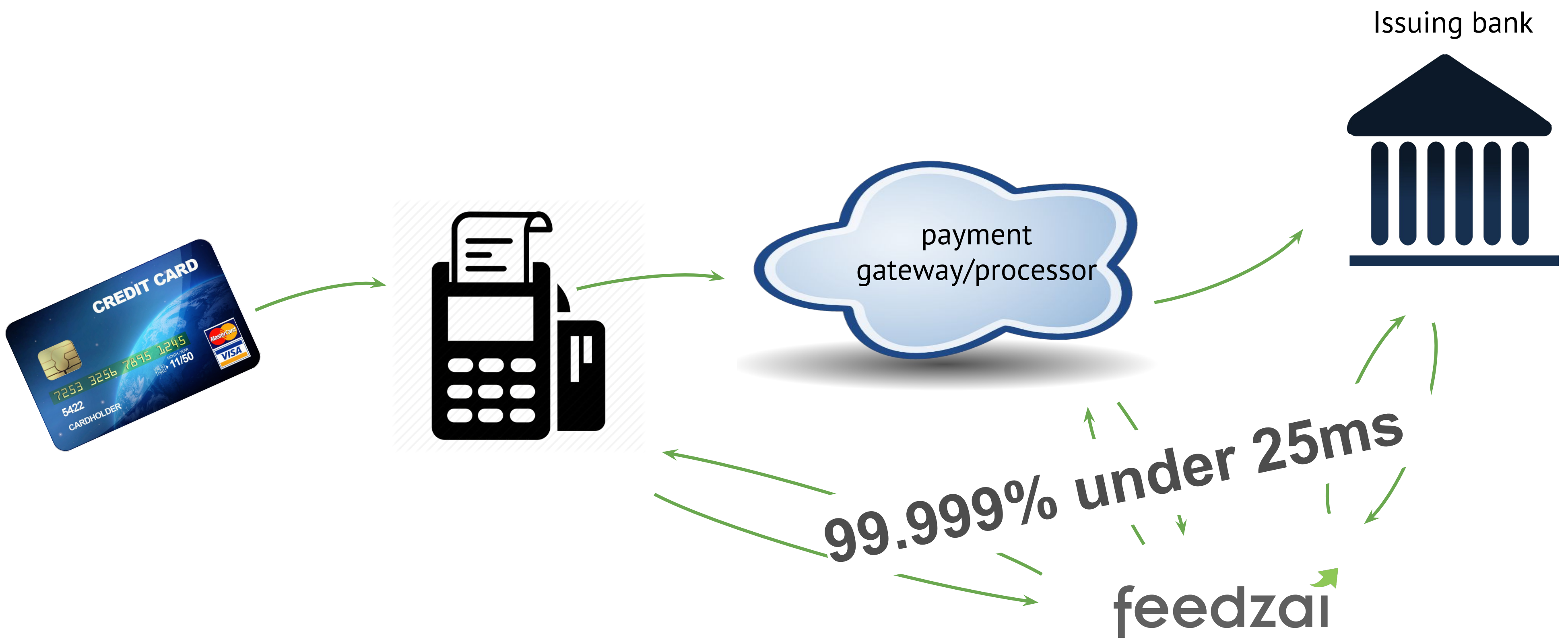
feedzai

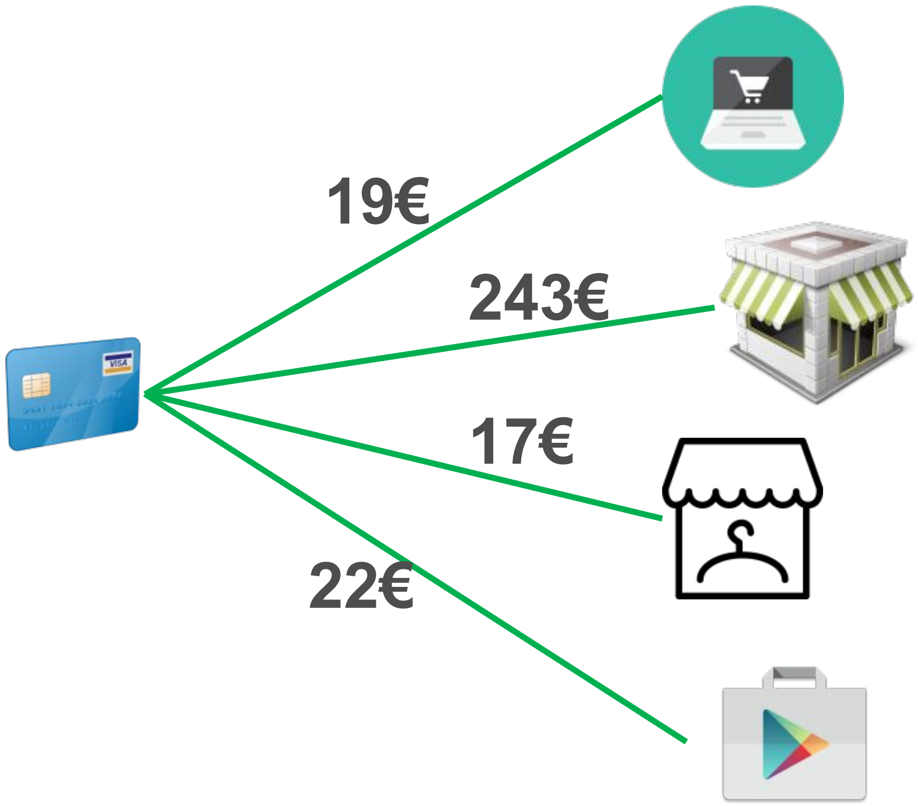
Life of a transaction



feedzai

Life of a transaction





Profiles are individual aggregations:

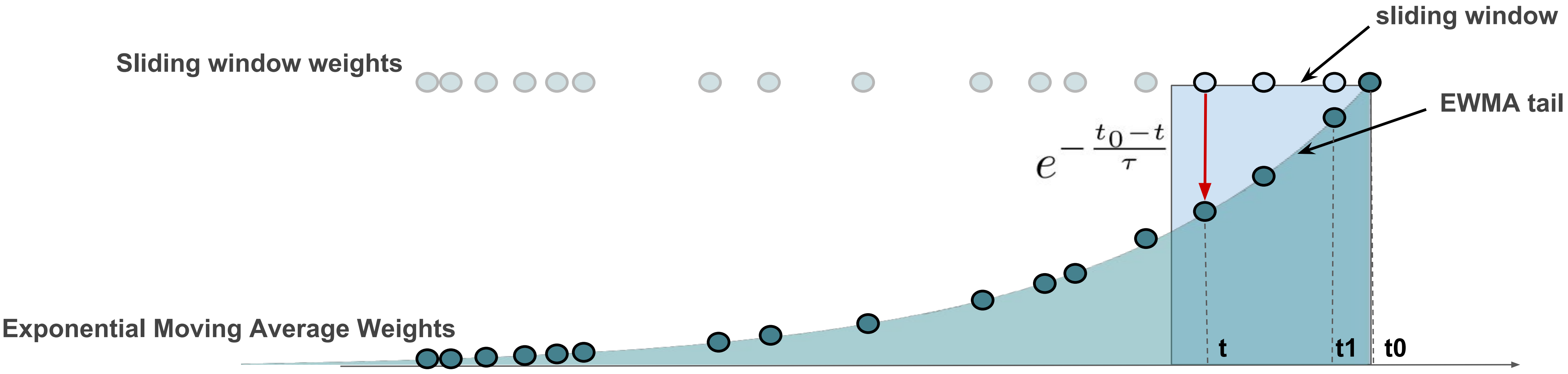
- Average spending in the last week.
- Distance to average location.
- Number of transactions in the last 10 minutes, ...

Context/Motivation

- Overhead of storing events.
- Overhead of expiring events.

feedzai

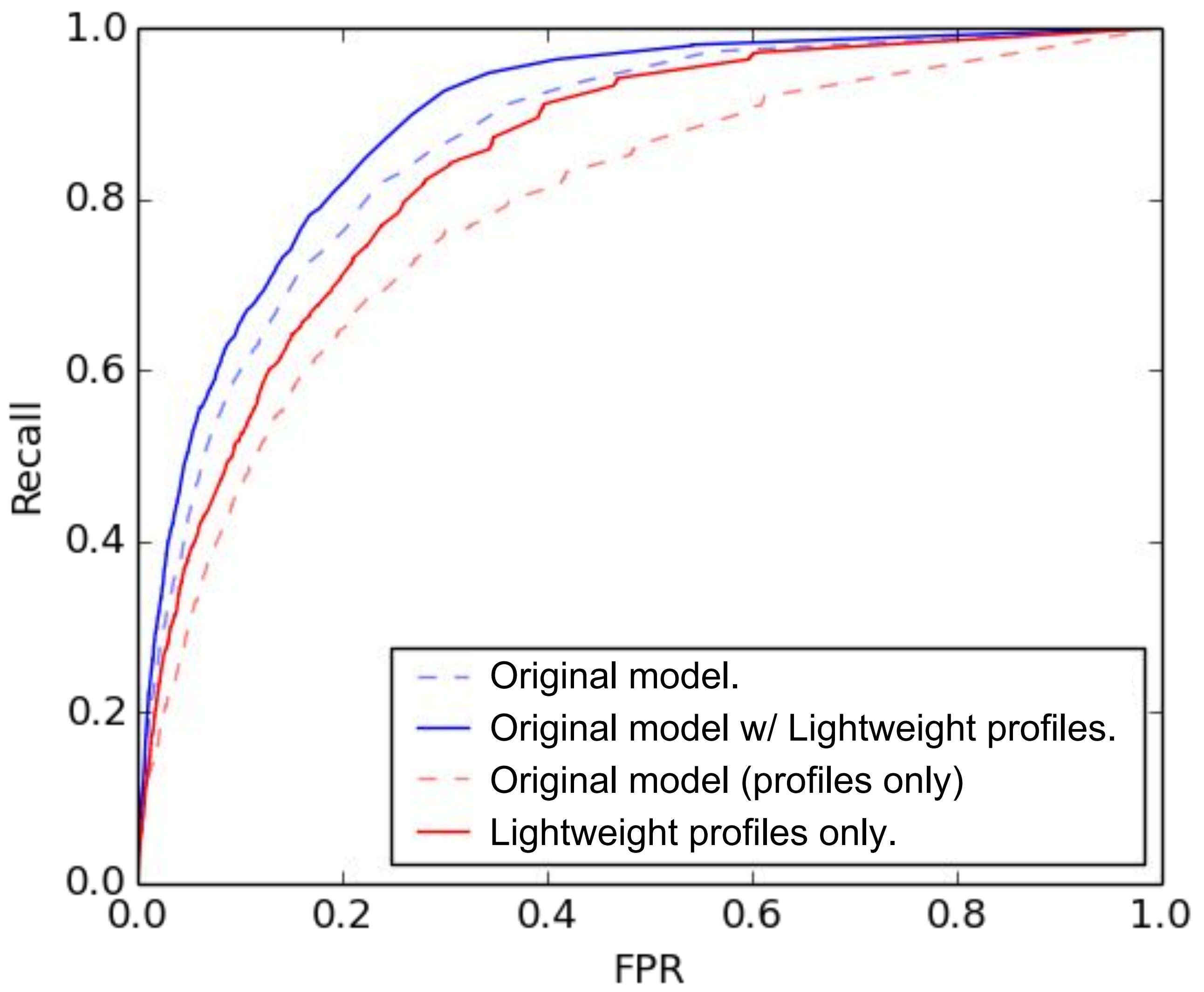
Sliding window VS EWMA



$$EMA(t_0) = \sum_{i=0}^{+\infty} z_i e^{-\frac{t_0 - t_i}{\tau}}$$

$$= EMA(t_1) e^{-\frac{t_0 - t_1}{\tau}} + z_0$$

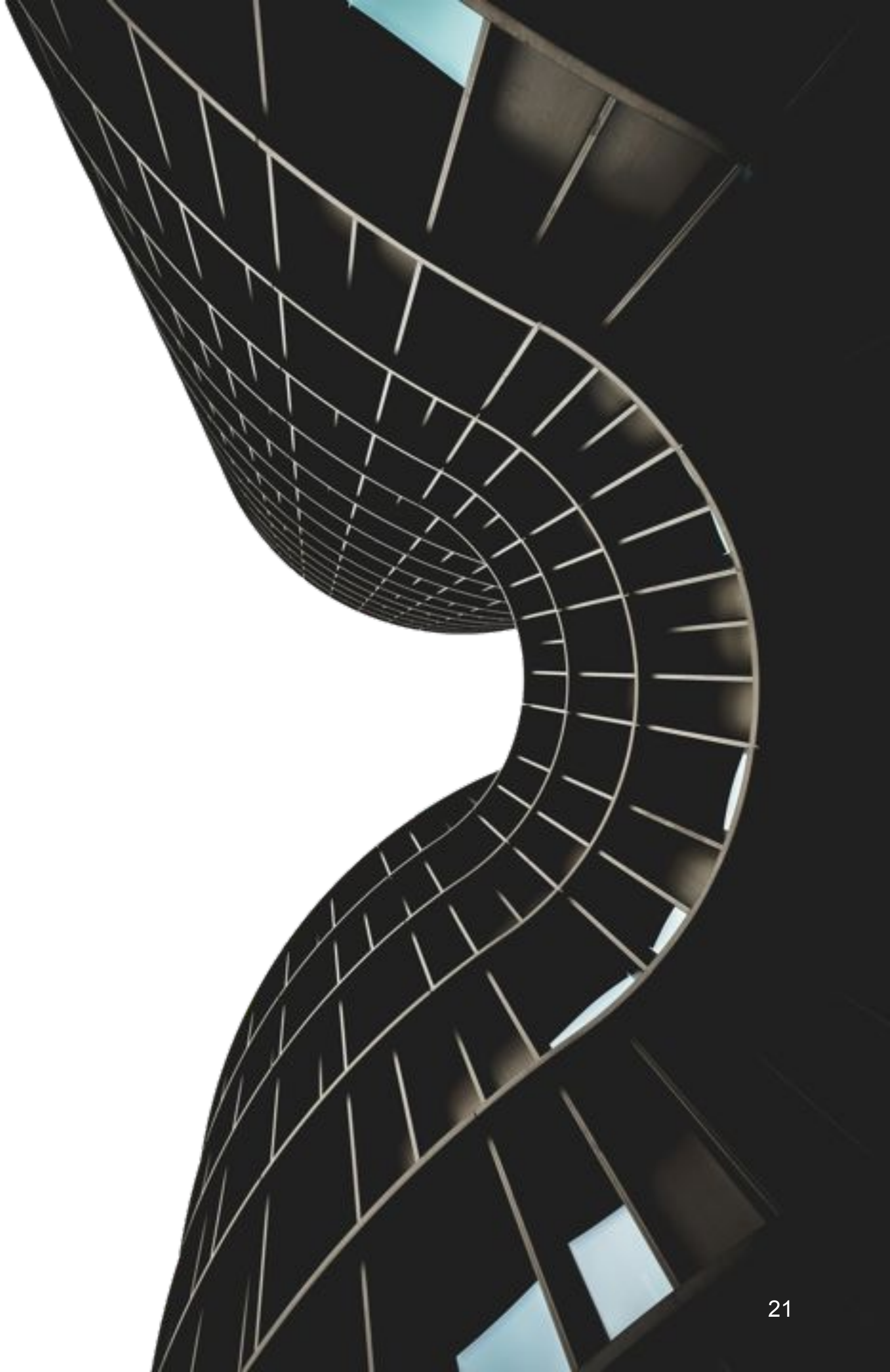
Huge memory savings!


1.0
0.8
0.6
0.4
0.2
0.0
Recall
0.0
0.2
0.4
0.6
0.8
1.0
FPR
Original model.
Original model w/ Lightweight profiles.
Original model (profiles only)
Lightweight profiles only.

feedzai

feedzai

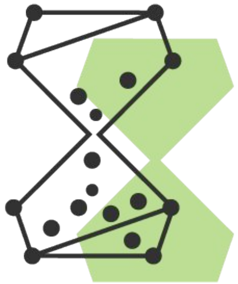
AutoML



AutoML

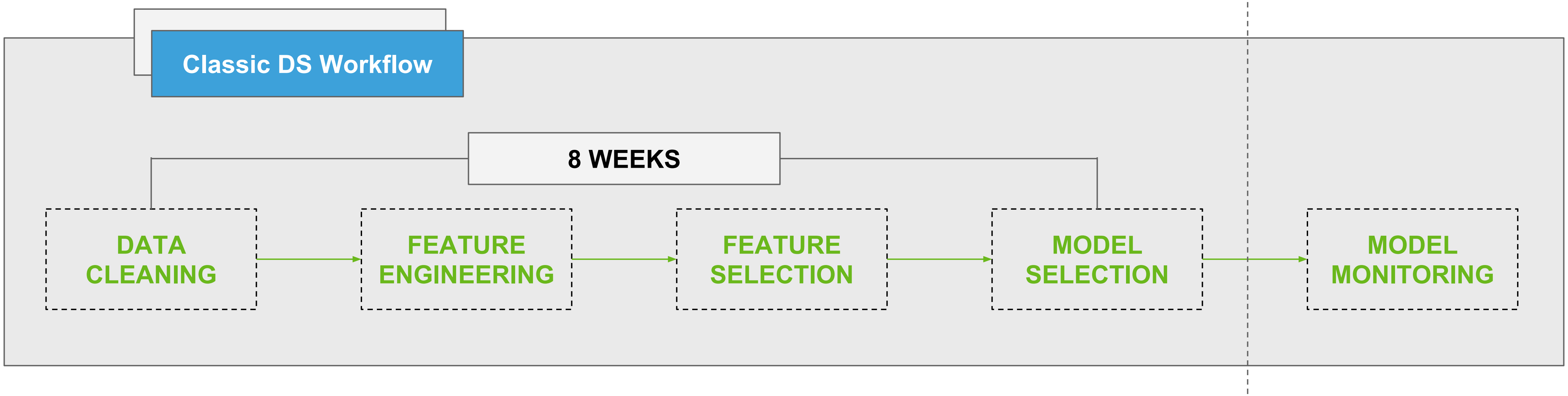
> 50 Data Scientists

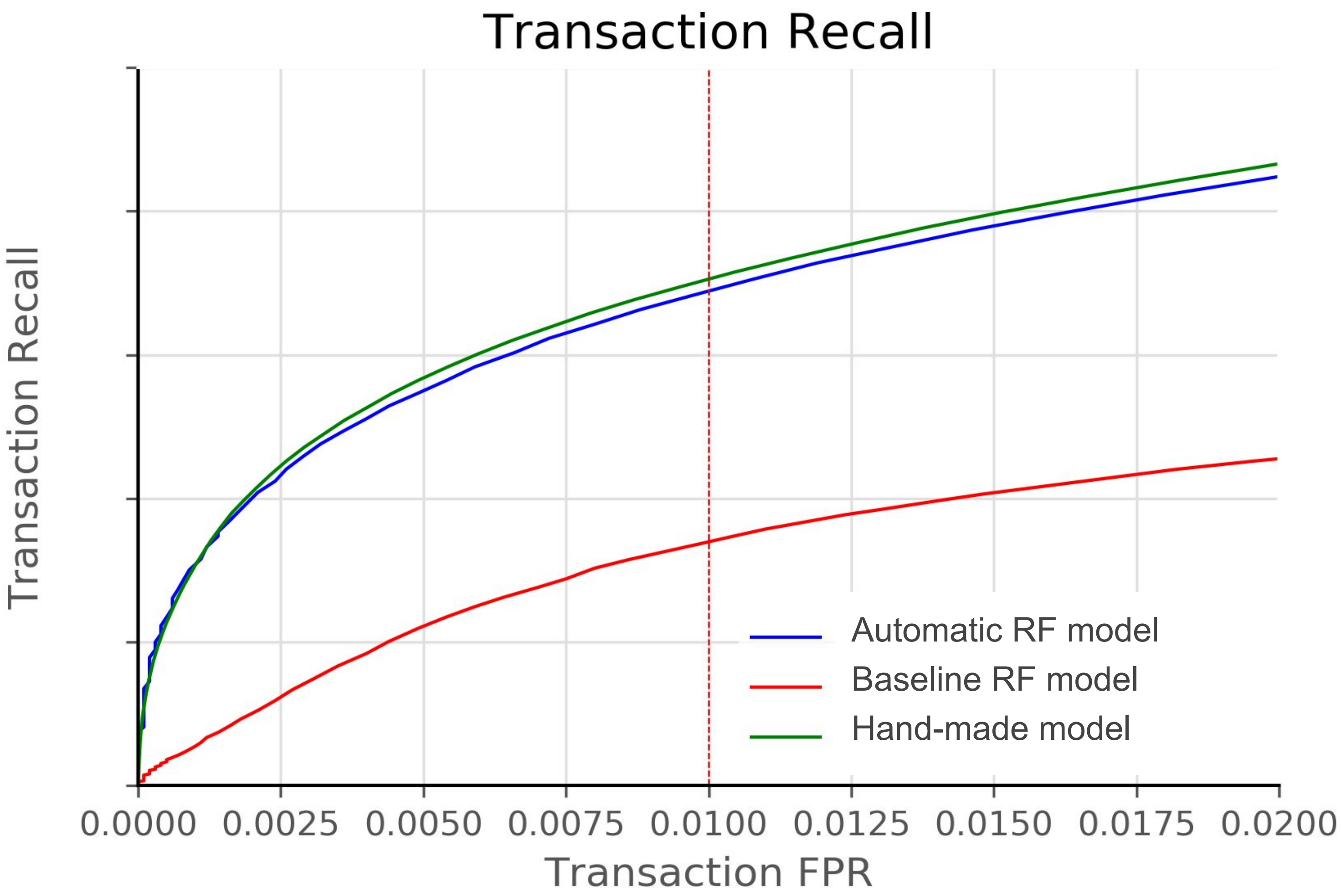
Tens of projects



Time to production

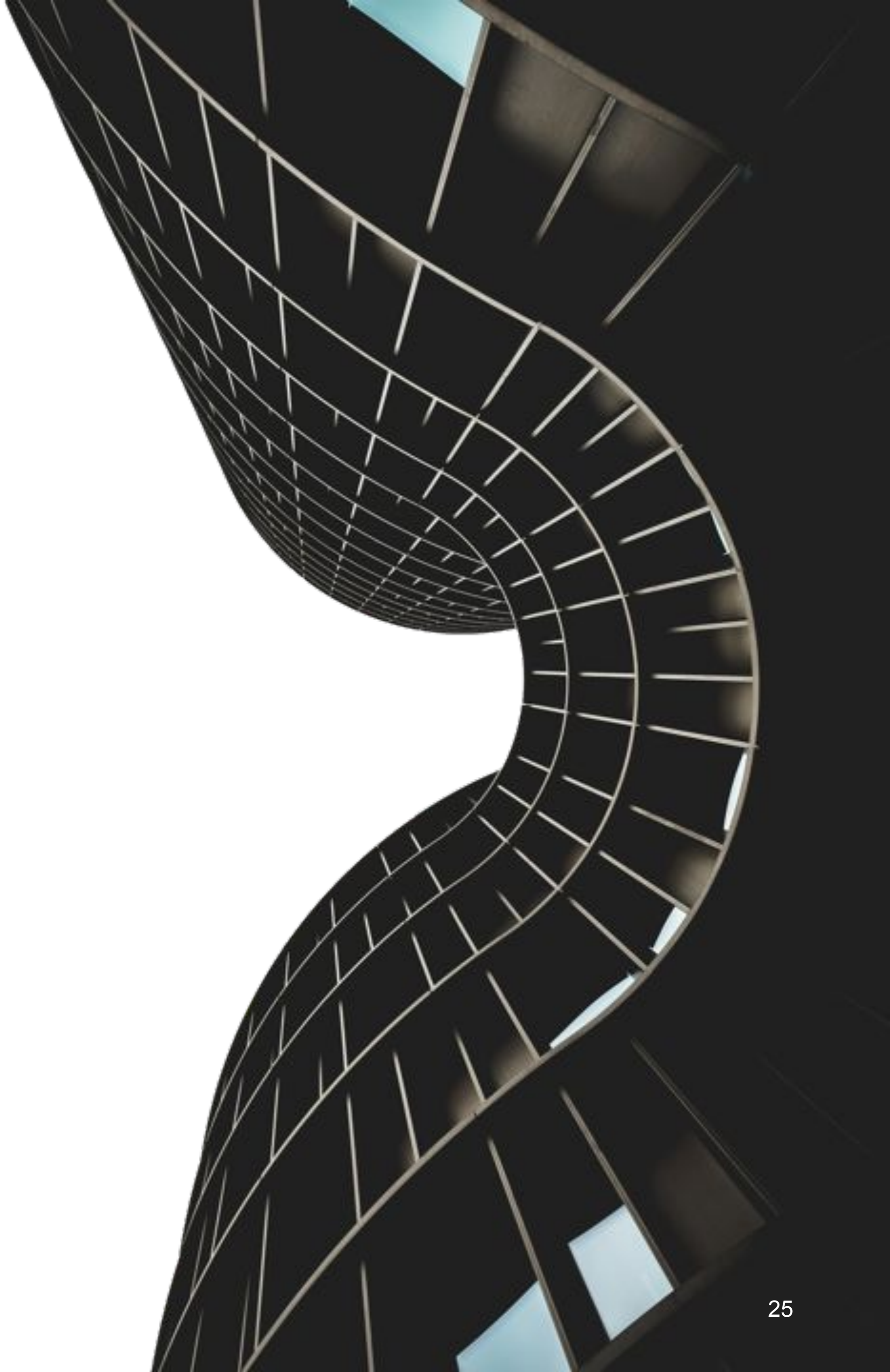
AutoML


Classic DS Workflow
8 WEEKS
DATA CLEANING
FEATURE ENGINEERING
FEATURE SELECTION
MODEL SELECTION
MODEL MONITORING


Transaction Recall
Transaction Recall
0.0000
0.0025
0.0050
0.0075
0.0100
0.0125
0.0150
0.0175
0.0200
Transaction FPR
Automatic RF model
Baseline RF model
Hand-made model

feedzai

Deep Learning



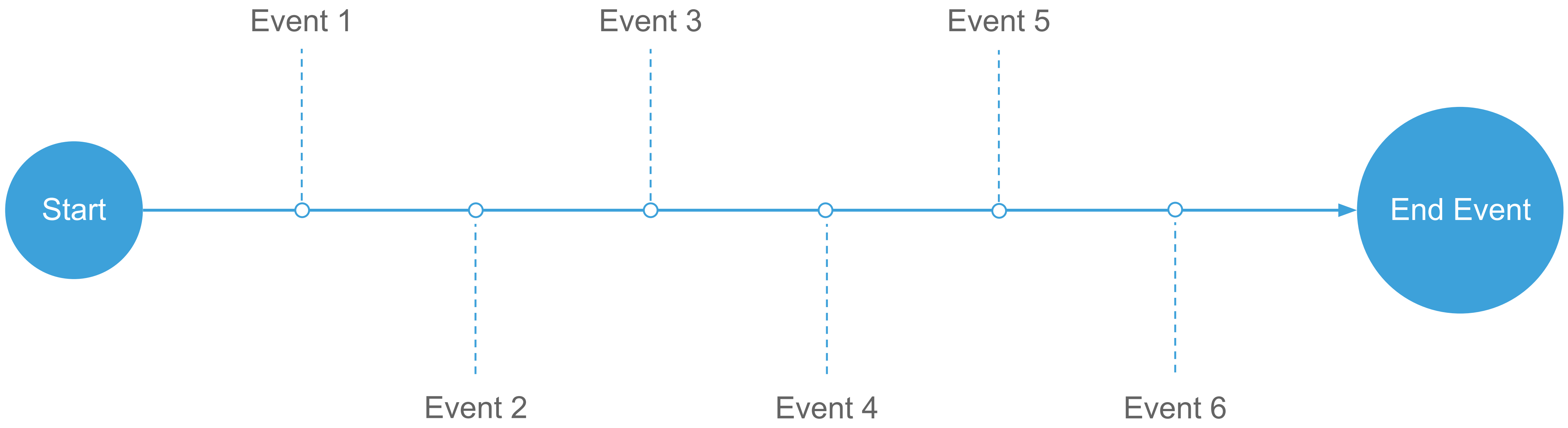
Why Deep Learning?

Automatic Feature Engineering is **good**.

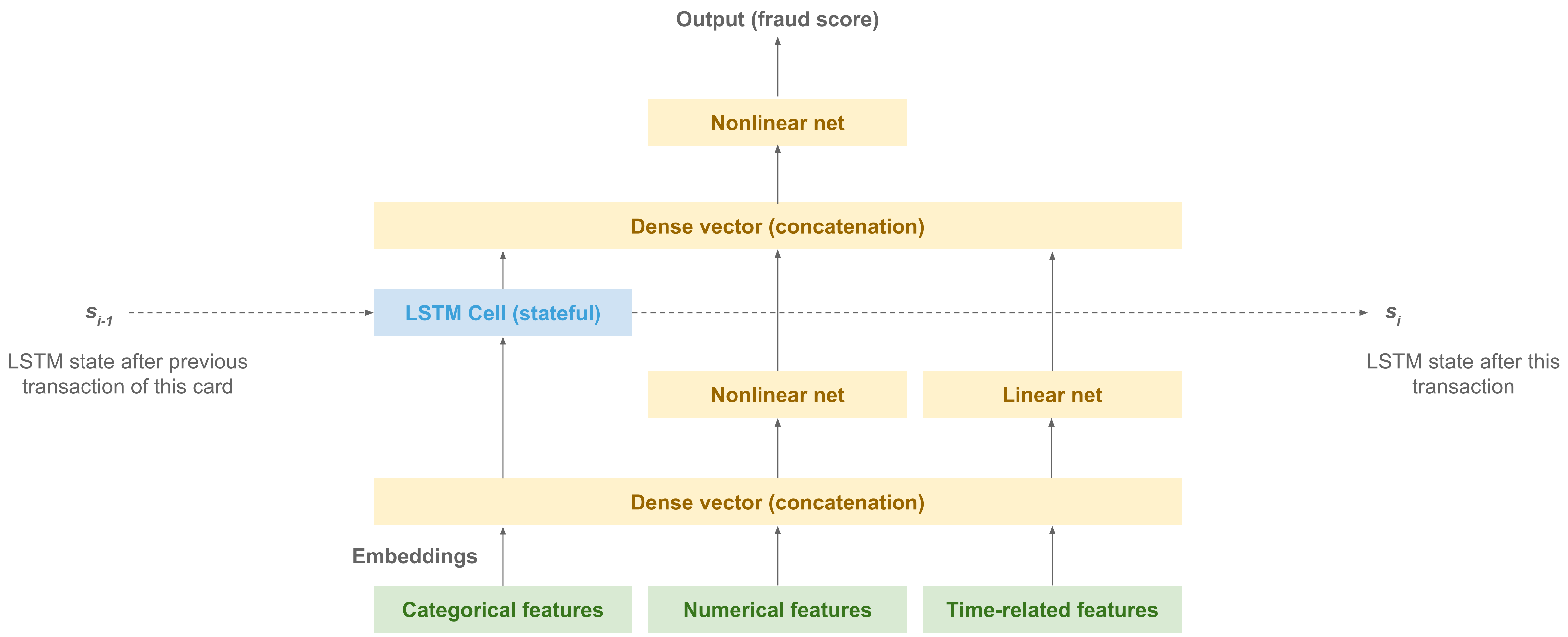
No Feature Engineering is **better**.

Sequence Network

- **Idea:** instead of scoring a transaction, score the **sequence of transactions of the card.**



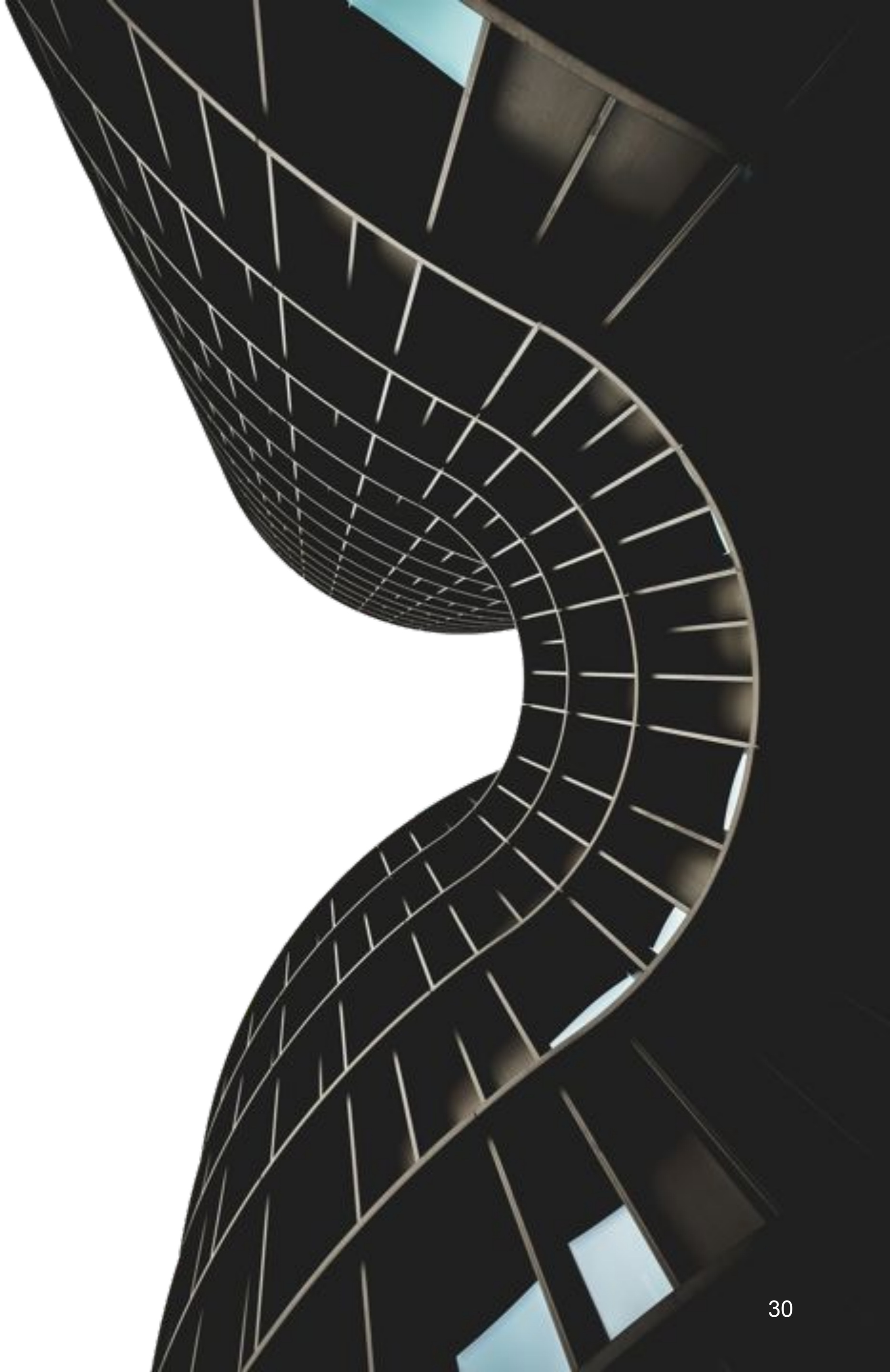
The sequence network



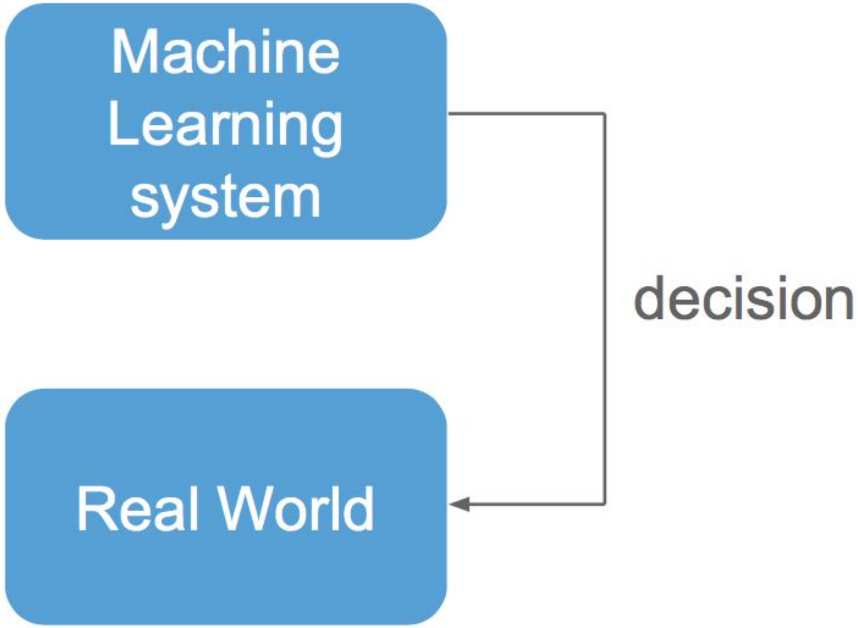
Deep Learning

Method	Dataset	Recall @ 1% FPR
Random Forest	**engineered data**	reference
XGBoost	**engineered data**	+5.4%
LightGBM	**raw data**	+5.8%
Sequence network	**raw data**	+12.5%

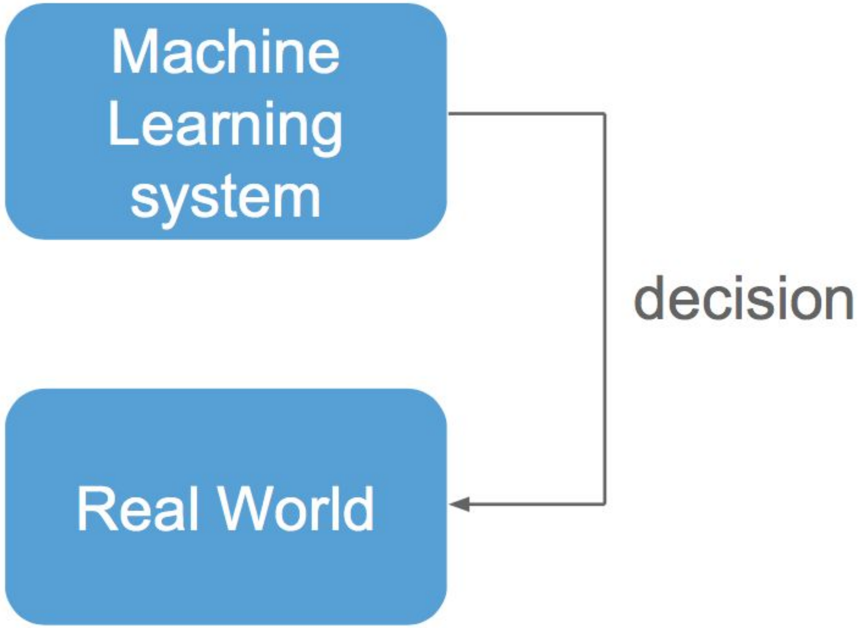
Reject Inference



Algorithms impact the real world

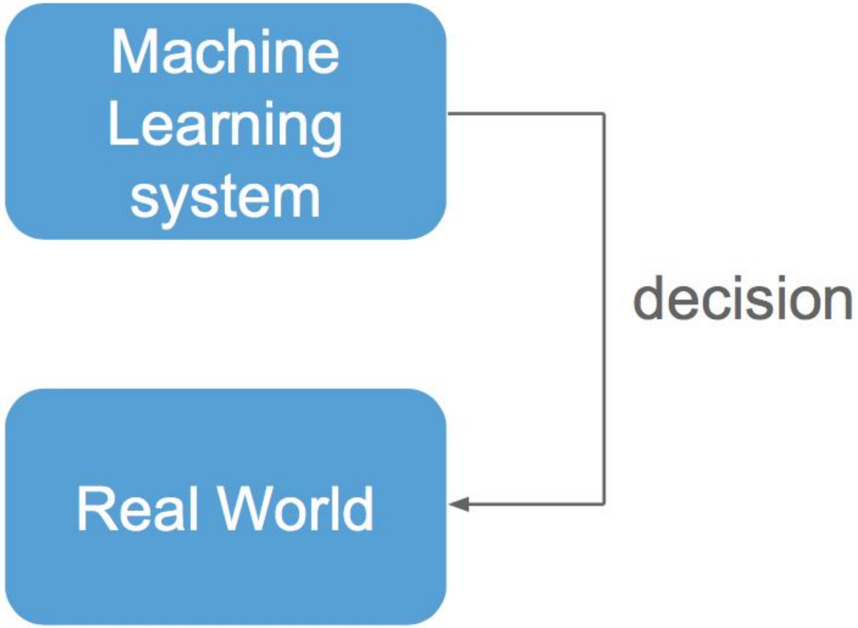


Algorithms impact the real world



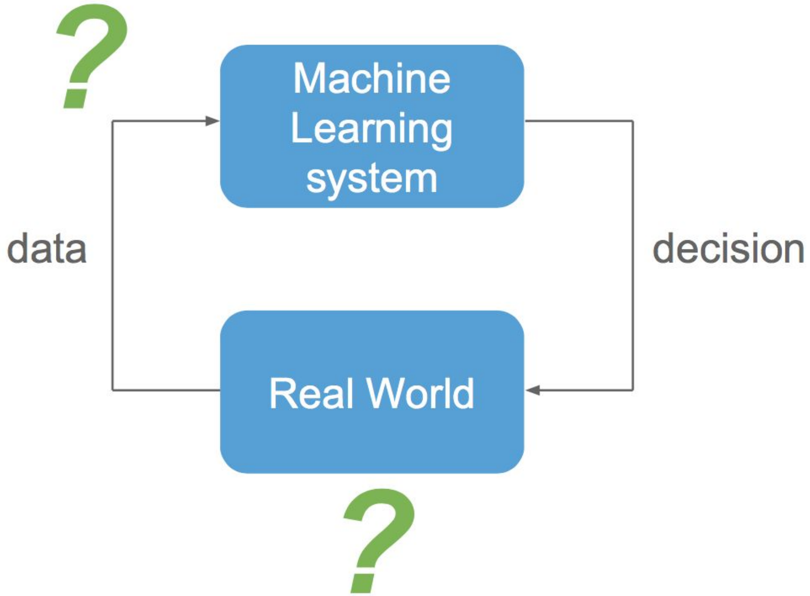
feedzai

Algorithms impact the real world



Counterfactual estimation

Many machine learning systems have direct influence in the real world



1. How to evaluate the production model?

Counterfactual Estimation

2. How to retrain a model, without bias?

Counterfactual estimation

	APPROVES	DECLINES
Actual Positives (fraud)	False Negatives (FN)	True Positives (TP)
Actual Negatives (legitimate)	True Negatives (TN)	False Positives (FP)

Counterfactual estimation

APPROVES | **DECLINES**

Label Observed | Label Not Observed

Counterfactual estimation

APPROVES

DECLINES

Label Observed	Label Not Observed

How to treat declines?	Model Evaluation	Model Re-Training
Ignore declines	0% Recall, NaN% Precision	Misses easy fraud

Counterfactual estimation

APPROVES — Label Observed

DECLINES — Label Not Observed

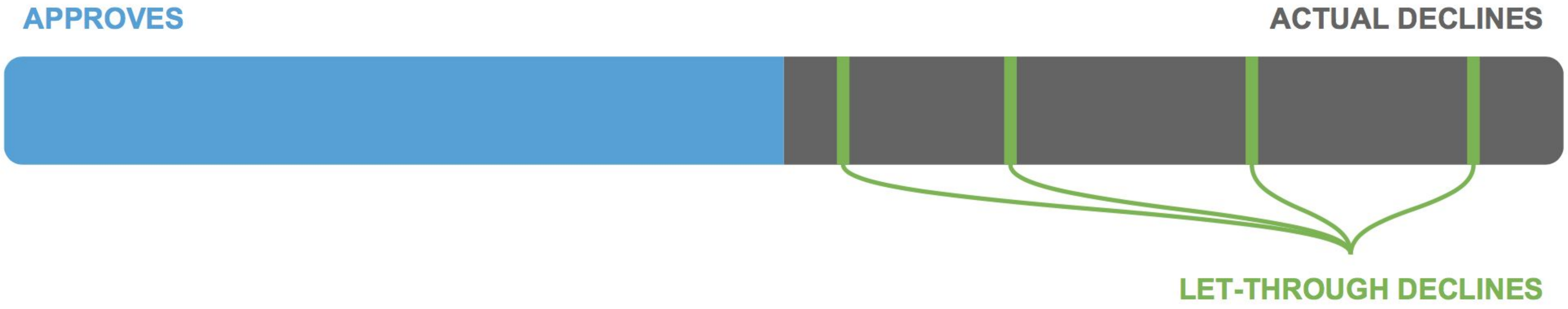
How to treat declines?	Model Evaluation	Model Re-Training
Ignore declines	0% Recall, NaN% Precision	Misses easy fraud
Treat declines as fraud	100% Precision, 0% FPR	Biased model

Counterfactual estimation

APPROVES | DECLINES
Label Observed | Label Not Observed

How to treat declines?	Model Evaluation	Model Re-Training
Ignore declines	0% Recall, NaN% Precision	Misses easy fraud
Treat declines as fraud	100% Precision, 0% FPR	Biased model
Use expert information	Costly + analyst bias	Costly + analyst bias

Evaluating Models



1. Randomly (with small probability, called *propensity*) **let through transactions classified as fraud**

2. After we get labels: estimate precision, recall, FPR **without significant bias**

feedzai

Evaluating Models

For all metrics:

1. Ignore actually declined transactions
2. Weight each approved transaction by *1/propensity*

Score	P(Approving)	Declined?	Fraudulent?	Weight
0.3	100%	No	Yes	1
0.4	100%	No	No	1
0.6	10%	Yes	?	0
0.7	10%	Yes	?	0
0.8	10%	No	Yes	10
0.9	10%	Yes	?	0

$$\text{Recall} = TP / (TP + FN)$$
$$= 10 / (10 + 1)$$
$$\approx 99.1\%$$

THANK YOU

we are hiring