

# Smarter Solutions with Hybrid Rule-ML Systems

Ivan Reznikov  
@hyke @mdxdubai

# Shameless Self-Promotion

- PhD in Computational Sciences
- 12+ years of Python and Data Science experience
- Worked for small/medium/large enterprise companies, startups, etc
- Lead Data Scientist at Hyke
- Kaggle Competition Expert
- TEDx Speaker (2017), GITEX/PyCON MEA (2021)

<https://www.linkedin.com/in/reznikovivan/>  
<https://github.com/IvanReznikov>

# Roadmap

- Rule-Based Systems
- ML-Based Systems
- Hybrid Systems
- Practical Cases
- Experience Using

# Rule-based systems

Set of if-then statements, also known as rules, to make decisions or perform specific tasks

Key: domain knowledge

Examples:

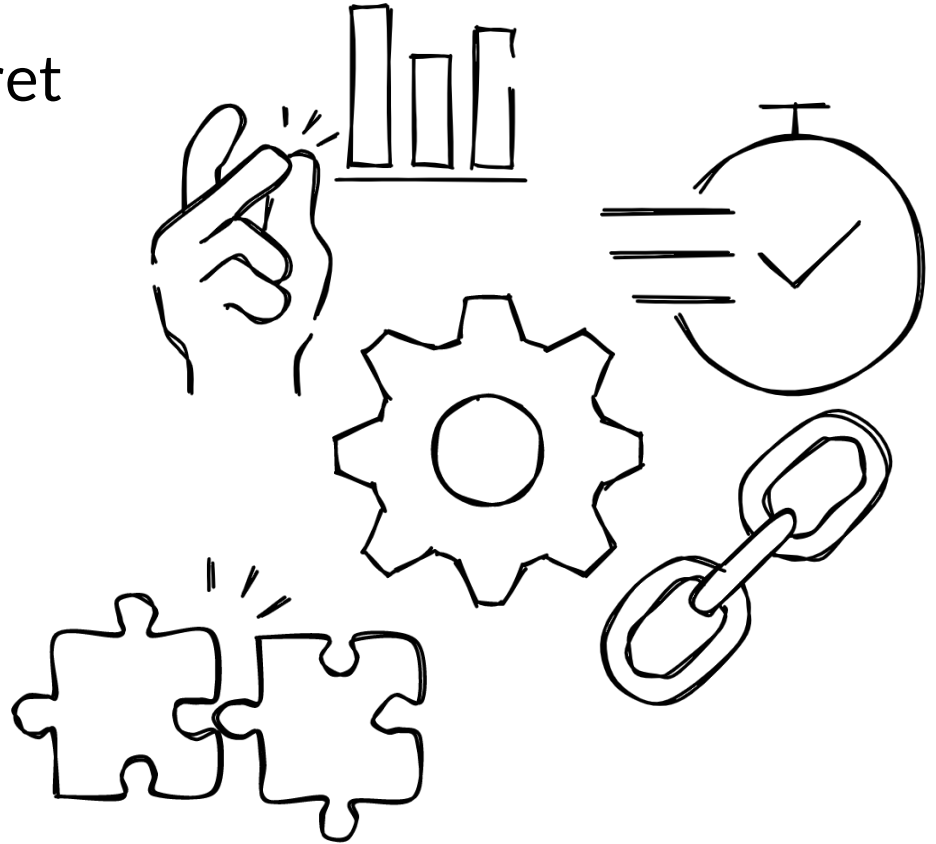
Finance: identify fraud by some common transaction patterns

Healthcare: assign doctors, based on medical history and analysis

Supply Chain: demand forecasting for new product introduction (NPI)

# Rule-based systems: Advantages

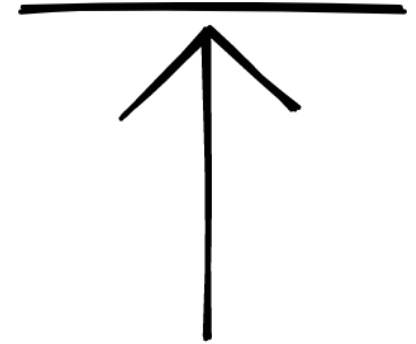
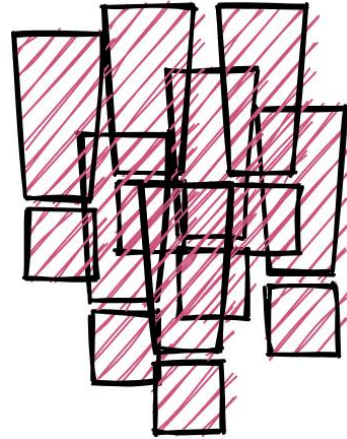
- + Simple to comprehend and interpret
- + Fast to implement
- + Easy to modify
- + Robust
- + Compatible with ML



# Rule-based systems: Disadvantages

- Issues involving a vast number of variables
- Problems with numerous constraints
- Limited to existing rules

_id	f1	f2	f3	f4	...	f <sub>n-1</sub>	f <sub>n</sub>
1	1.3	2.1	0.99	1.1	...	8.7	11.2
2	24.5	4.1	3.5	2.3	...	9.5	3.1
3	2.2	45.1	NaN	0.2	...	11.7	NaN
4	0.3	12.9	4.4	1.4	...	0	23.1



# ML-based systems

Set of algorithms to learn from data and make predictions or take actions without being explicitly programmed to do so

Key: patterns from historical data

Examples:

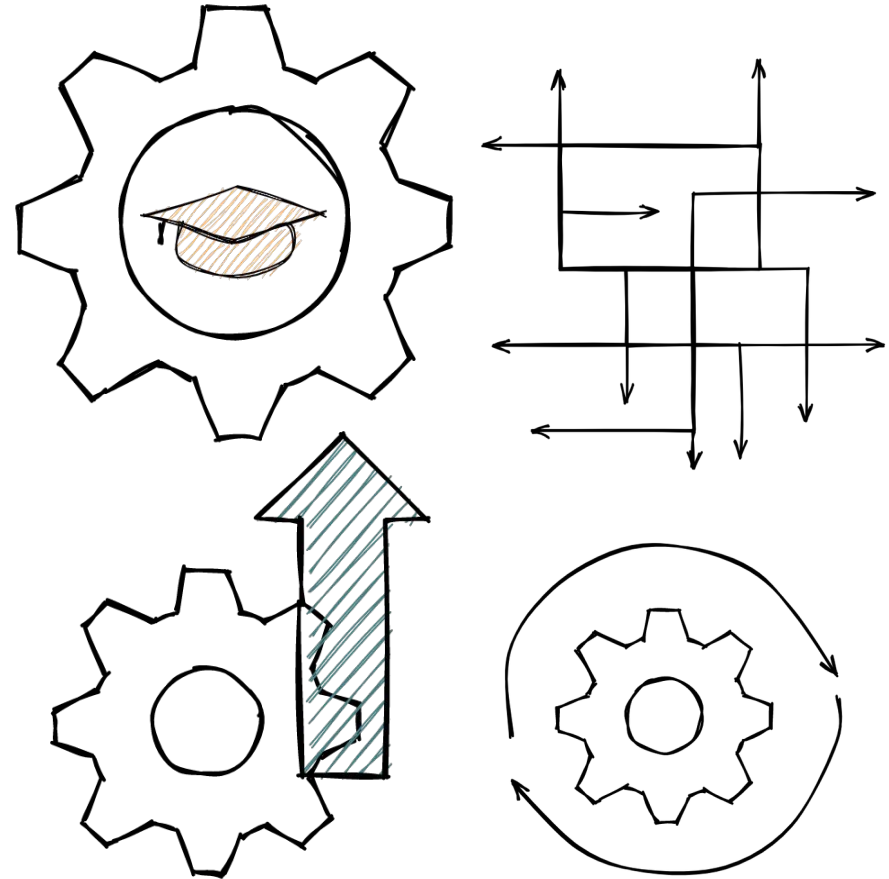
Finance: analyze large amounts of financial data to detect outlier fraudulent transactions

Healthcare: analyze patient data and predict the likelihood of a patient developing a certain disease

Supply Chain: demand forecasting based on historical sales

# ML-based systems: Advantages

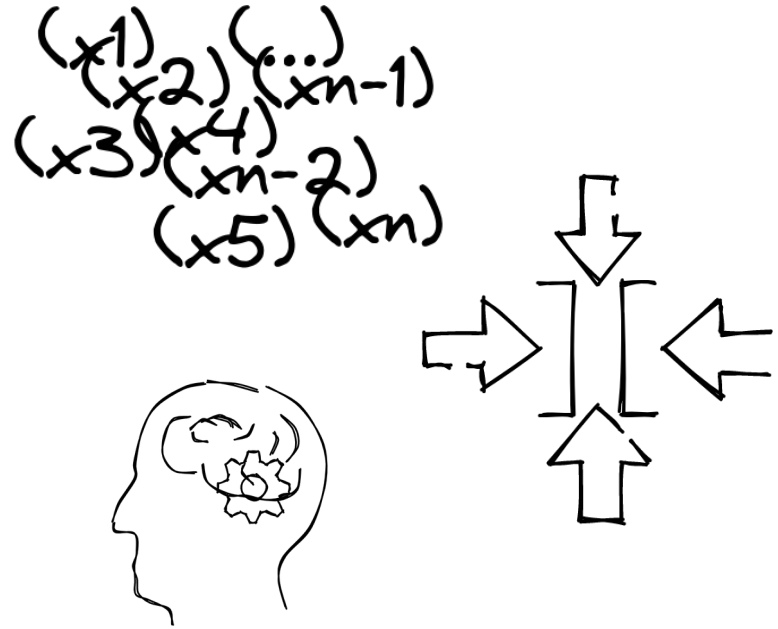
- + Autonomous learning systems
- + Ability to tackle more intricate problems
- + Higher efficiency with reduced human intervention compared to rule-based systems
- + Flexibility to adapt to changes in data and environment over time through continuous learning





# ML-based systems: Disadvantages

- Issues involving a vast number of variables
- Problems with numerous constraints
- Limited cognitive ability



# Rule vs ML-based systems

## What shall I choose?

# Rule vs ML-based systems

## Rule-Based:

- Danger of error
- Fast delivery
- Not enough data



## ML-Based:

- Complex problem
- Self-learning
- More flexible
- Better results



# Hybrid rule-ML systems

A hybrid rule-ML systems is a combination of the two approaches, which aims to leverage the strengths of both to achieve better results.

Sometimes even better than either system can on its own.



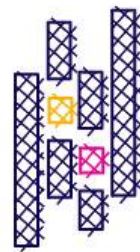
# Hybrid-based systems

id	n_bedrooms	area	price	...	sold_7	price_drop
1	1	800	1.1	...	0	0
2	1	750	0.9	...	1	0
3	2	1100	1.6	...	0	1
4	0	600	0.7	...	1	1
n	1	900	1.2	...	0.7	0.45

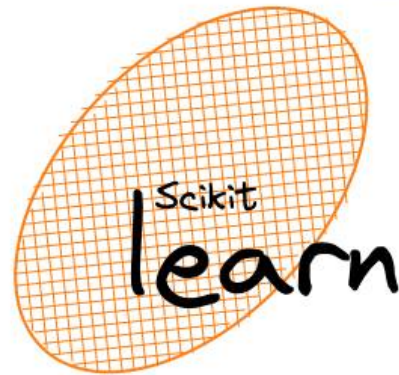
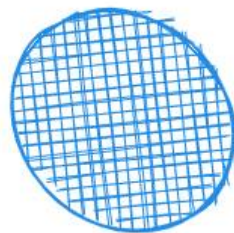
1.  $\text{Rule}(x_i) \rightarrow \text{ML}(x_i)$ : Feature engineering
2.  $\text{ML}(y_i) \rightarrow \text{Rule}(y_i)$ : Post processing
3.  $\text{ML}(y_i) \rightarrow \text{Rule}(z_i)$ : Use ml predictions for other decisions. For example, should we place a bid for a flat.
4.  $\text{ML}(y_i) + \text{Rule}(y_i) \rightarrow \text{aggregated forecast}$

The Jupyter logo consists of a large orange circle with a cross-hatch pattern. Inside the circle, the word "Jupyter" is written in a dark gray, handwritten-style font. Four small gray circles with cross-hatch patterns are positioned around the main orange circle: one at the top-left, one at the top-right, one at the bottom-left, and one at the bottom-right.

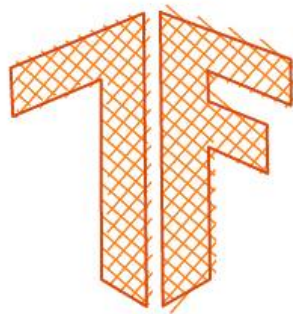
Jupyter



pandas



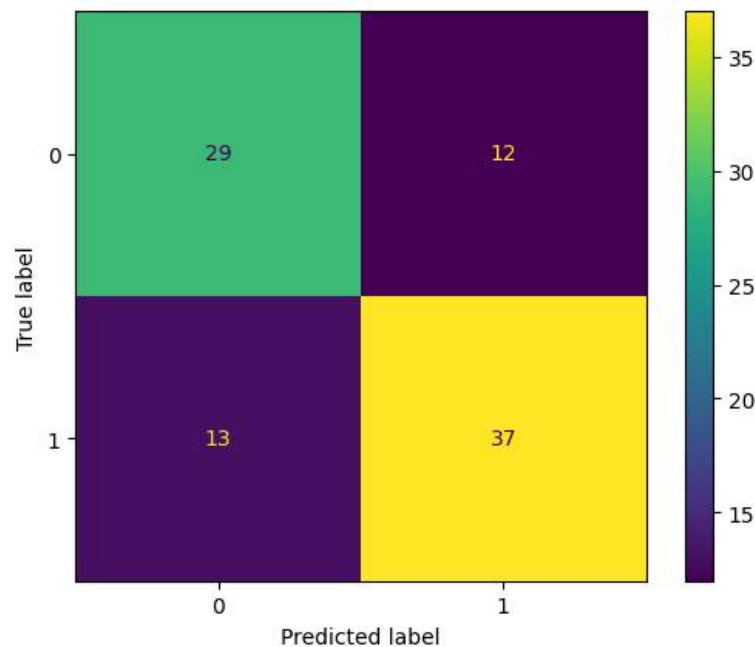
Scikit  
learn



TensorFlow

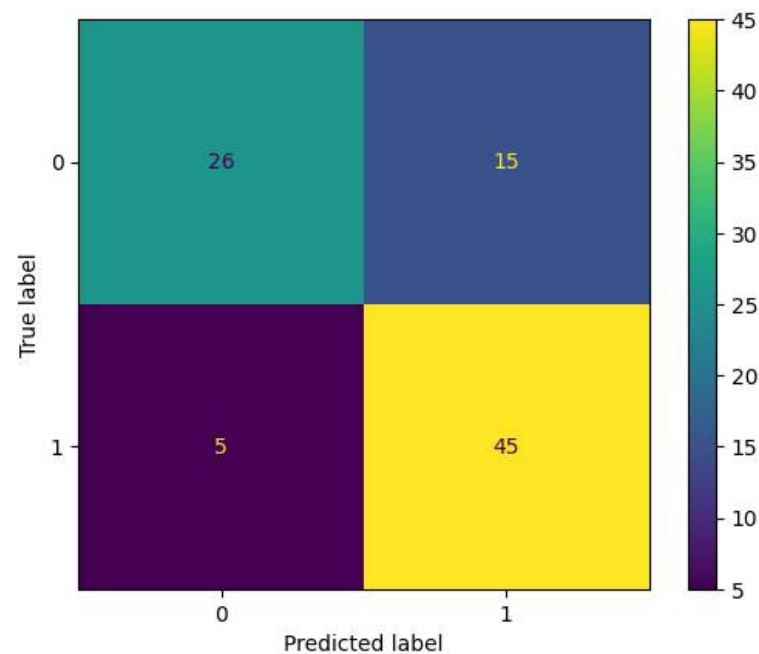
# Hybrid Systems: Missed Feature

Pure ML (Random Forest)



$$F1 = 0.747$$

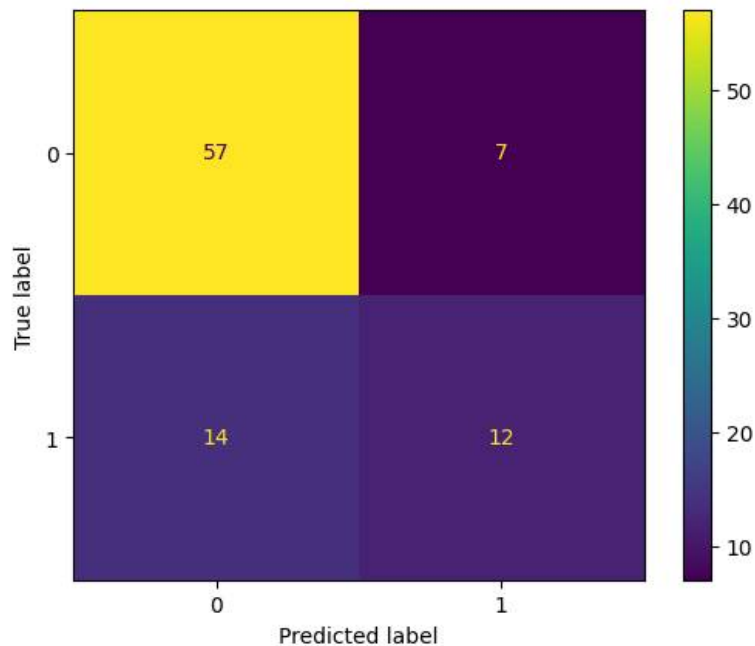
Hybrid Based



$$F1 = 0.818$$

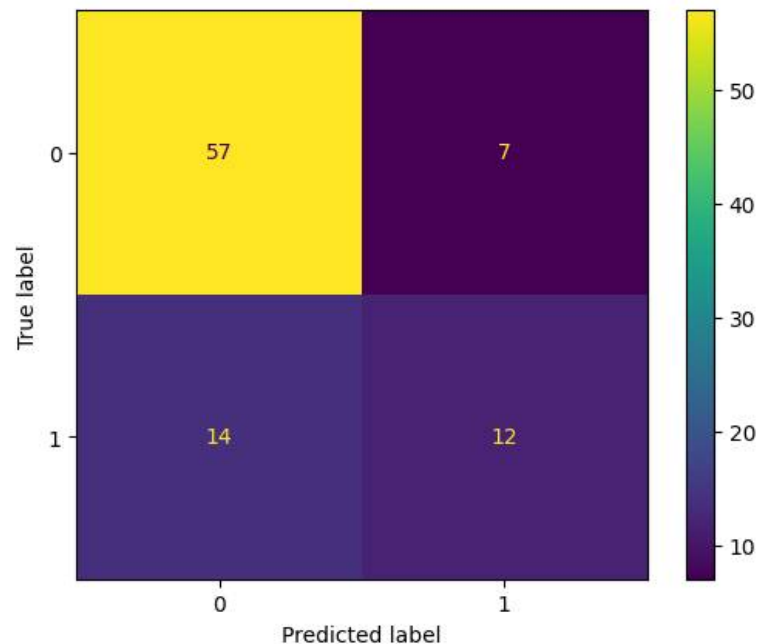
# Hybrid Systems: All Feature Caught

Pure ML (Random Forest)



$$F1 = 0.533$$

Hybrid Based

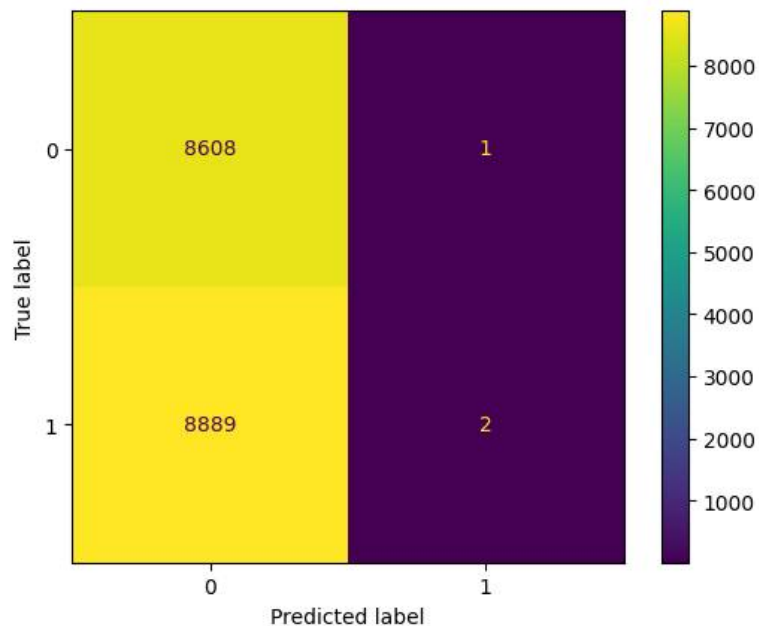


$$F1 = 0.533$$



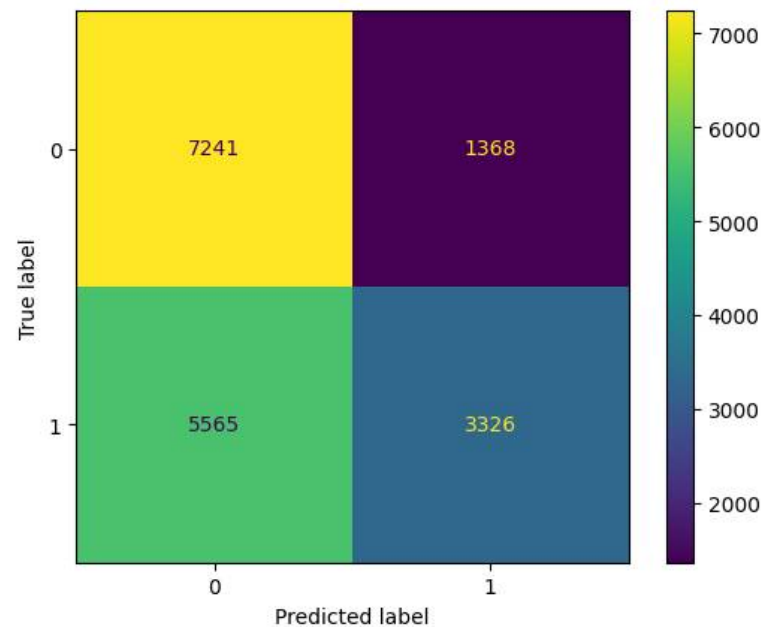
# Hybrid Systems: Neural Network

Pure ML (NN)



$$F1 = 0.0004$$

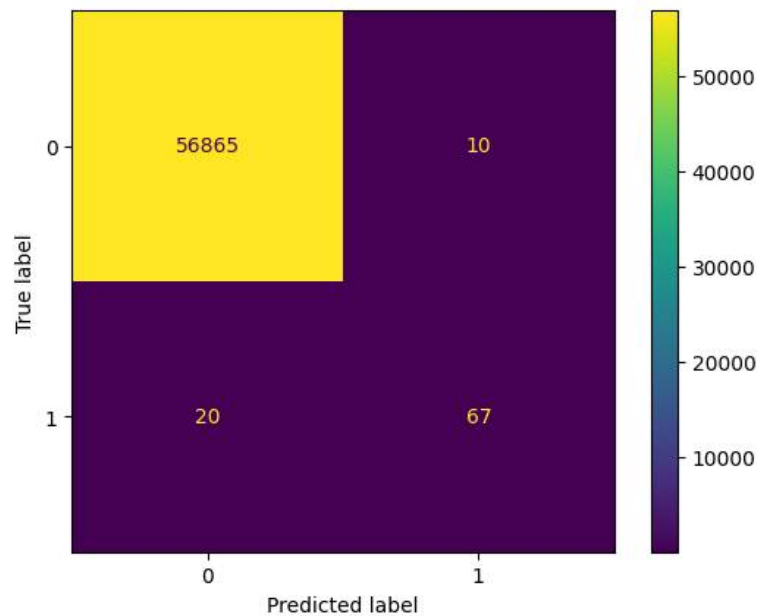
Hybrid Based



$$F1 = 0.489$$

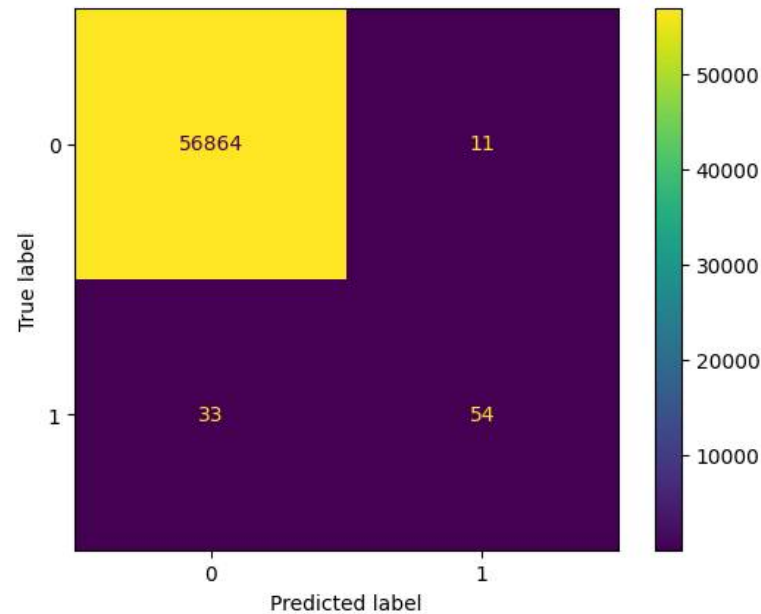
# Hybrid Systems: Unbalanced Dataset

Pure ML (kNN)



$$F1 = 0.817$$

Hybrid Based




$$F1 = 0.711$$

# Hybrid Systems: Demand Forecasting

## 1. Format Outputs (without Data Filtering)

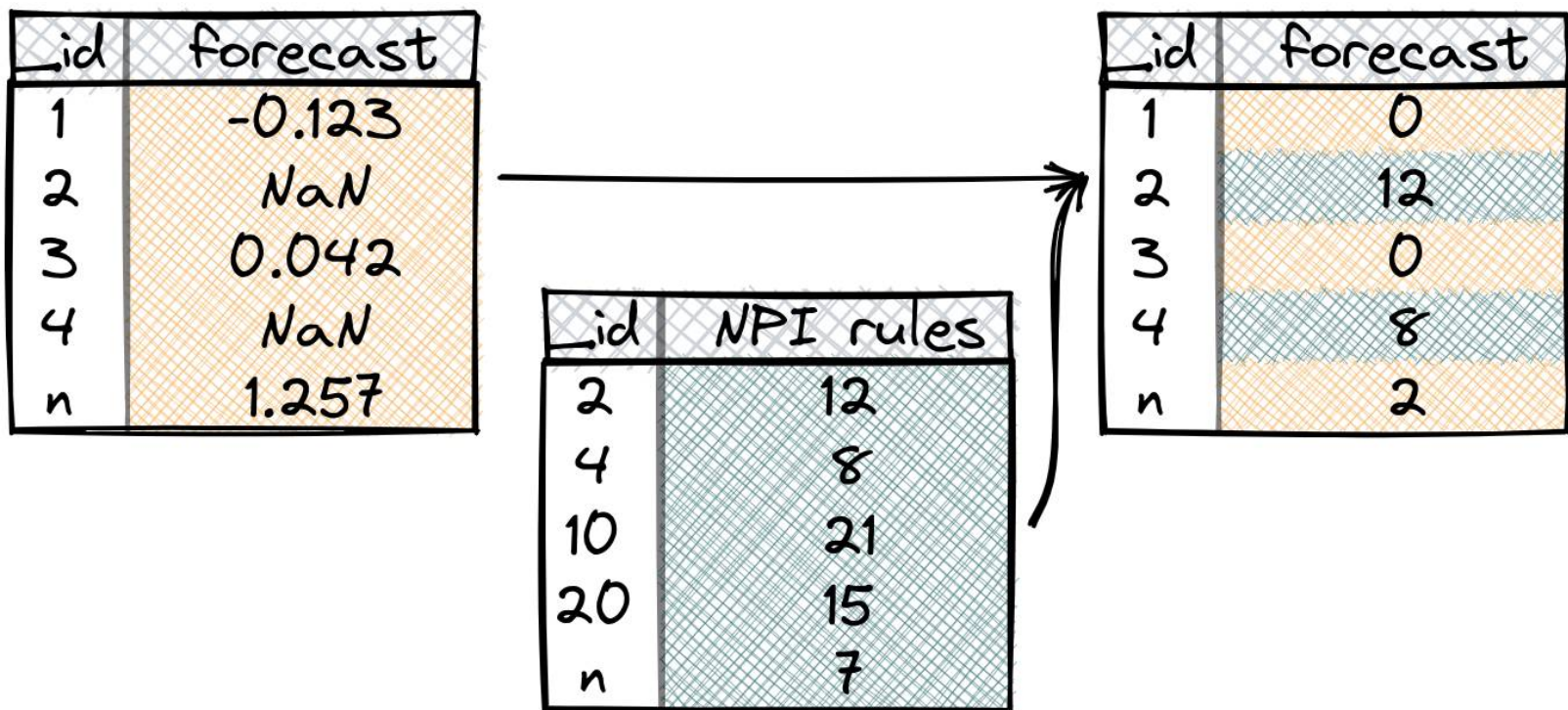
id	forecast
1	-0.123
2	-1.203
3	0.042
4	0.718
n	1.257



forecast
0
0
0
1
2

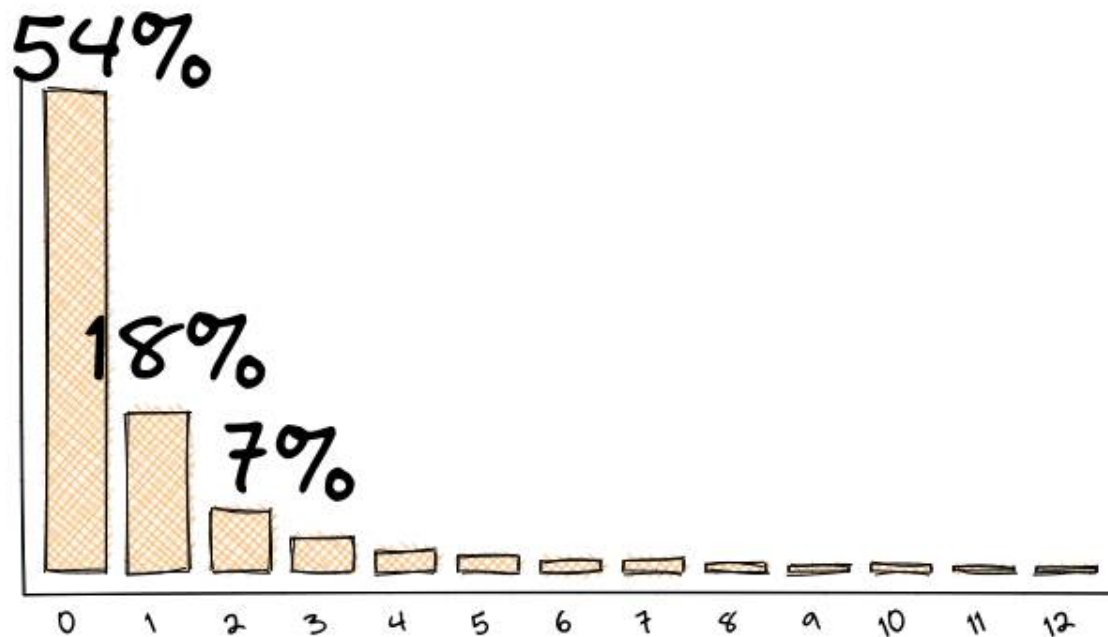
# Hybrid Systems: Demand Forecasting

## 2. Format NPI Outputs (without Data Filtering)



# Hybrid Systems: Demand Forecasting

## 3. Select Threads (with Data Filtering) for Time Series



# Conclusions

Hybrid rule-ML systems offer practical benefits such as fast implementation, robustness to outliers, and increased transparency.

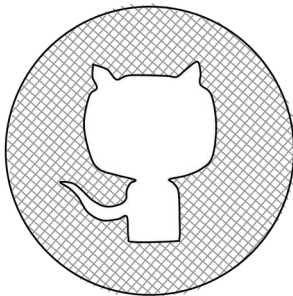
They are handy when combining business logic with machine learning. For instance, hybrid rule-ML systems in healthcare can diagnose diseases by combining clinical rules and machine learning algorithms that analyze patient data.

As business and data science continue to incorporate, hybrid systems can be crucial in smoother integration.

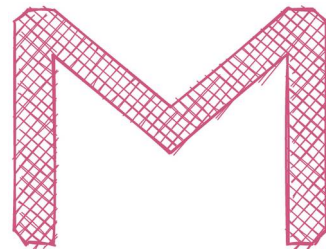
# Time for Questions



<https://www.linkedin.com/in/reznikovivan/>



<https://github.com/IvanReznikov>



[ivanreznikov@gmail.com](mailto:ivanreznikov@gmail.com)