

Крипто прогнозиране & изкуствен интелект

„Прогнозирането на финансовите пазари е като да караш кола със закрито предно стъкло, гледайки само в огледалото за задно виждане.“

猿猿猿猿猿猿 **Изготвили** 猿猿猿:

Велизар Митов

Десислава Недялкова

Иван Паспалджиев

Иван Резняков

Камелия Стефанова

Николай Пенчев

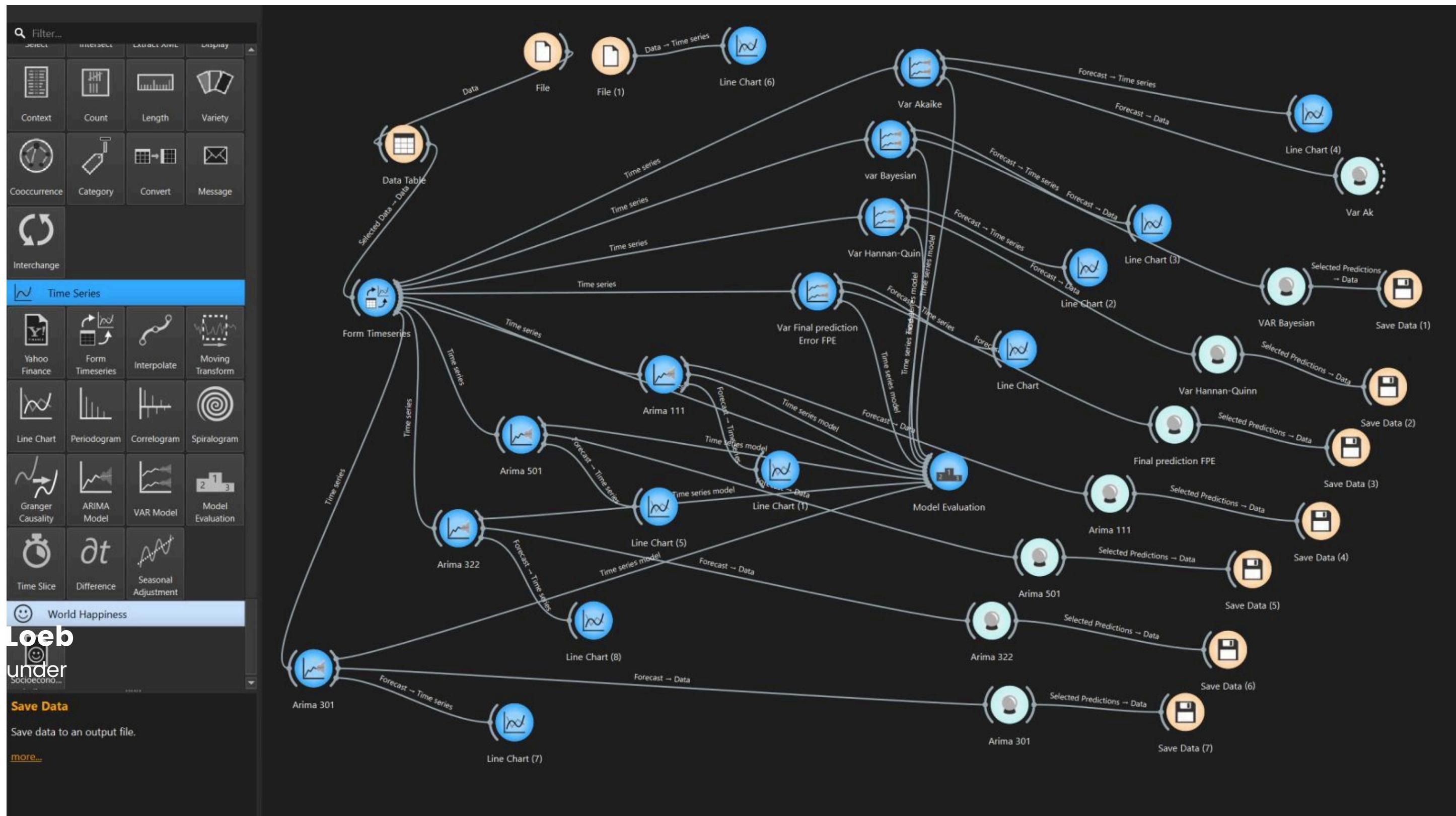


План на работа

- Архитектура на проекта
- Използвани модели за предсказване
- Описателна статистика
- Feature engineering ARIMA & VAR via Orange
- Използвани модели (ML: LSTM и хибриден)
- Визуализация (Plotly, Matplotlib)
- Оптимизация на портфейл с генетичен алгоритъм
- Backtest (модул метрики(МАРЕ, DS, R, G))
- Финален код review

Descriptive Statistics

Онусамелна стамисмуга



ВЪВЕДЕНИЕ

Архитектура на проекта В контекста на CRISP-DM

Cross Industry Standard Process for Data Mining

(Стандартизиран междусекторен процес за извличане на знания от данни)

CRISP-DM е често използван в проекти с данни, включително и такива за предсказване на цени на криптовалути.

👉 Целта ни е да изградим модел за прогноза на BTC цената на биткойн и Rule-based правила-базирани на AI трейдър за BTC (начално ниво).

👉 Проблемът е как да прогнозираме BTC цената една стъпка напред и да вземаме базови автоматизирани инвестиционни решения

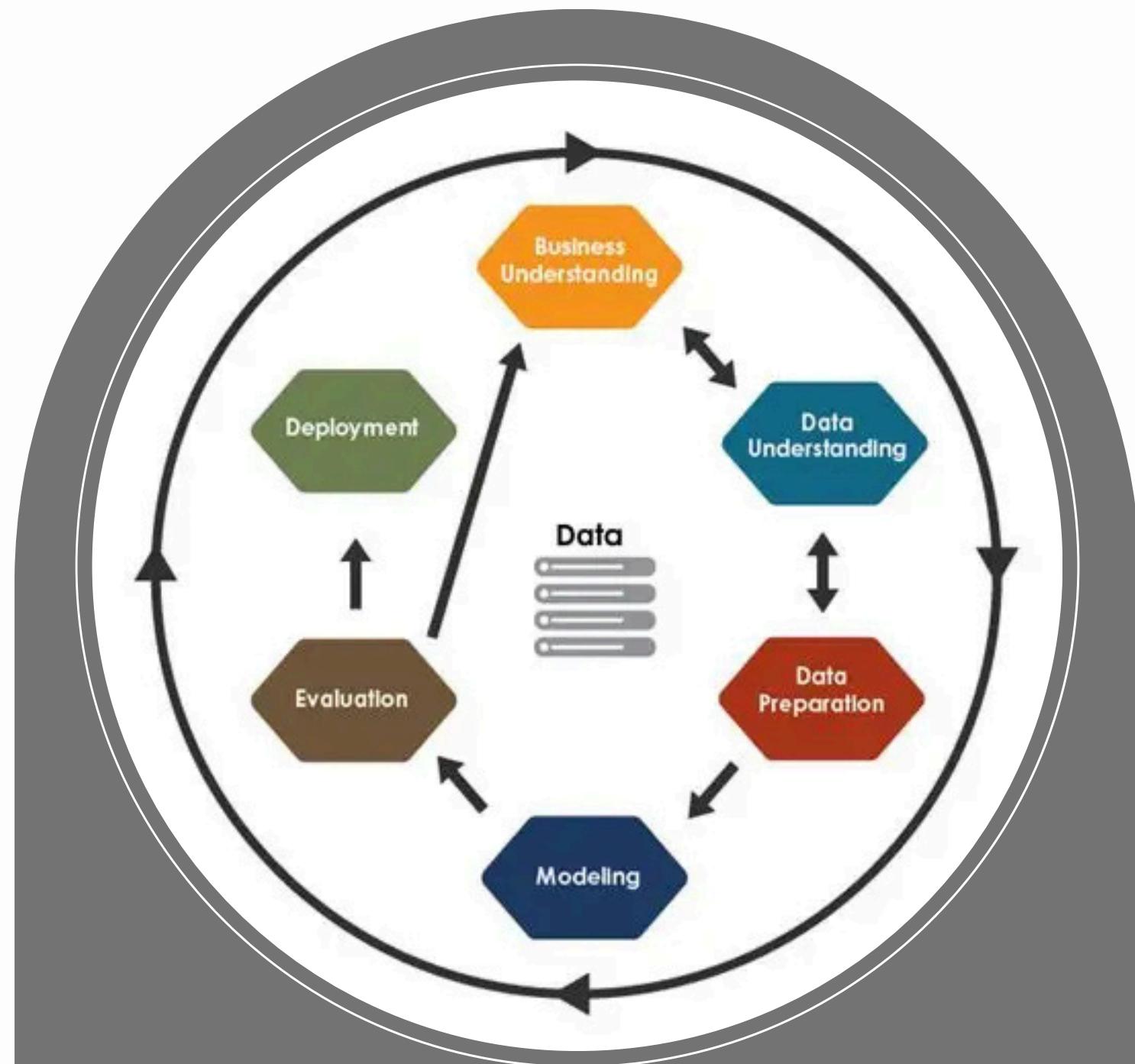
👉 Разбиране на данните. Исторически данни на последните 10 години от Investing.com

👉 Подготовка на данните. Описателна статистика и хистограма

👉 Моделиране. Моделите са ARIMA via Orange & VAR via Orange

👉 Оценка. Валидация на база различни статистики.

👉 Разгръщане. Model training



Използван модел LSTM

LSTM (Long Short-Term Memory)

(Дългосрочна краткосрочна памет)
е тип рекурентна невронна мрежа
(RNN) recurrent neural network

Моделът, който използваме е за да направим ценово прогнозиране за пазара на биокайн. Фокусираме се върху хибриден модел за дълбоко обучение, който комбинира CNN, части от трансформатори и LSTM. Нашата цел е да направим прогноза за следващата цена, използвайки само минали и текущи данни, без изтичане на бъдещата информация.

```
#Causal Conv1D: Avoids future data, good for fast signal extraction.
#Multi-Head Attention: Transformer-style attention to focus on important parts of the sequence (no future leak).
#LSTM: Captures temporal dependencies.
#Dense: Outputs 1 value (predicted delta).
#Loss is MSE, but the model uses sample_weight=confidence, giving more influence to high-confidence examples.

def build_causal_price_predictor(input_shape):
    from tensorflow.keras.models import Model
    from tensorflow.keras.layers import Input, Dense, Conv1D, Add, LayerNormalization, MultiHeadAttention, LSTM, Dropout

    inp = Input(shape=input_shape)

    # Causal convolution: looks only at current and past
    x = Conv1D(32, kernel_size=1, padding='causal', activation='relu')(inp)

    # Transformer-style attention (no future access)
    attn = MultiHeadAttention(num_heads=2, key_dim=32)(x, x, attention_mask=None)
    x = Add()([x, attn])
    x = LayerNormalization()(x)
    x = Dropout(0.3)(x)

    # Single-direction LSTM (causal)
    x = LSTM(64, return_sequences=False)(x)
    x = Dropout(0.3)(x)

    # Final linear regression output
    out = Dense(1, activation='linear')(x)

    model = Model(inputs=inp, outputs=out)
    model.compile(optimizer='adam', loss='mse', metrics=['mae'])
    return model
```

Защо LSTM е подходящ за криптовалути?

- Моделът може да запомня пазарни движения и модели във времето.
- Устойчив е на шум в данните, характерен за криптоиздадите.
- Подходящ е за краткосрочни и дългосрочни предсказания.

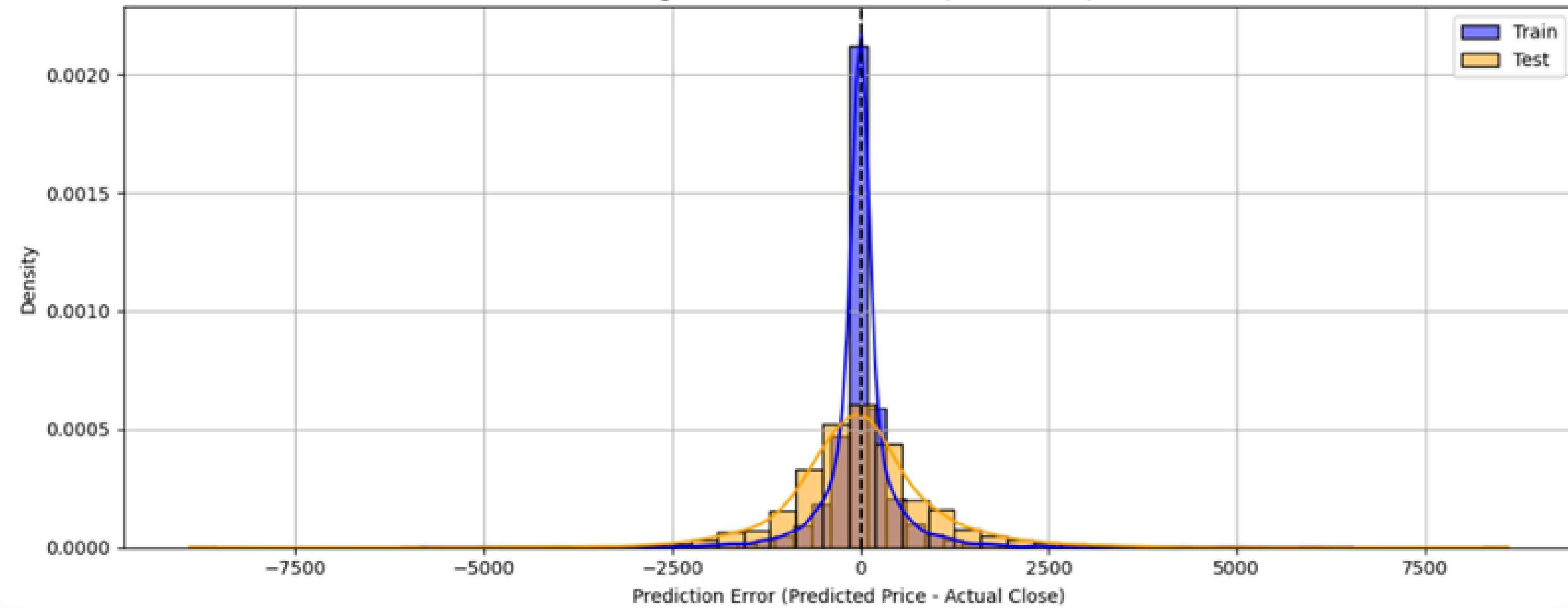
Прогнозирана срещу реална цена на замваряне



Тук виждаме сравнение между реалната цена и прогнозираната. Моделът се справя добре – линиите почти се припокриват. Това показва, че моделът успява да следва тренда. Но за реална търговия не е достатъчно, защото не улавя резките промени толкова добре.

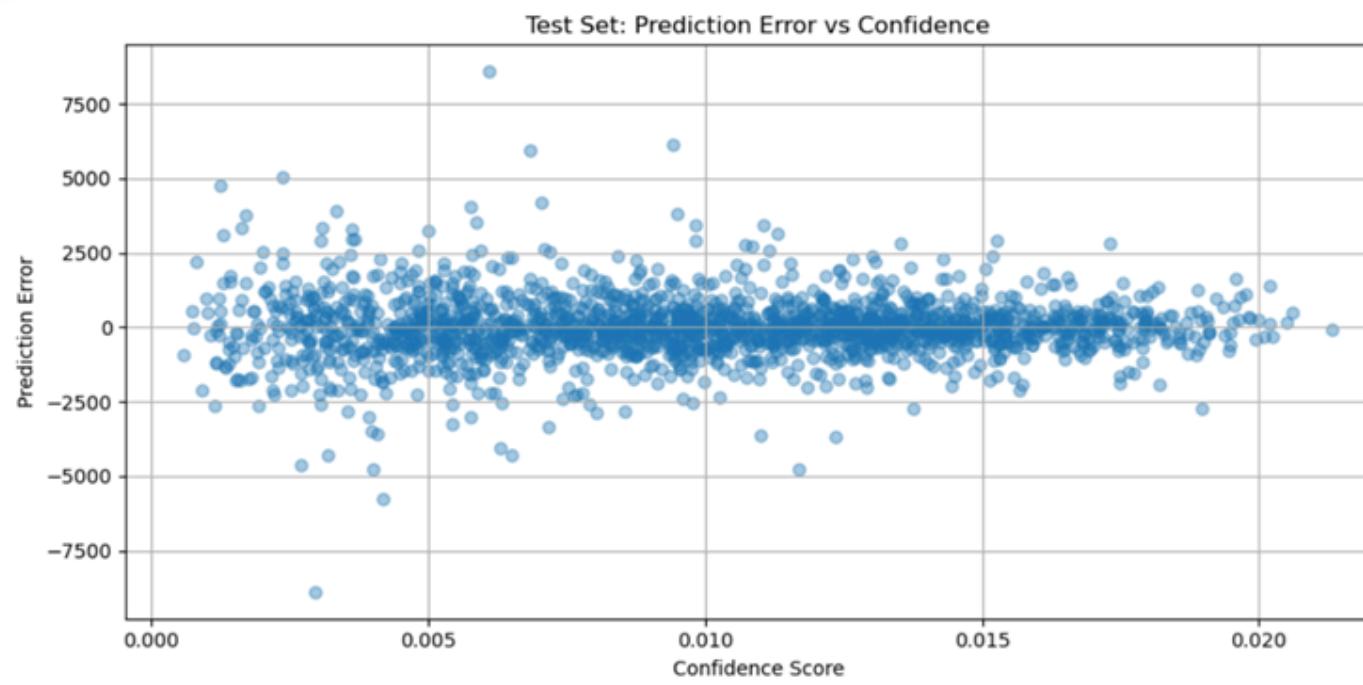
Грешки в прогнозирането

Histogram of Prediction Errors (Train vs Test)

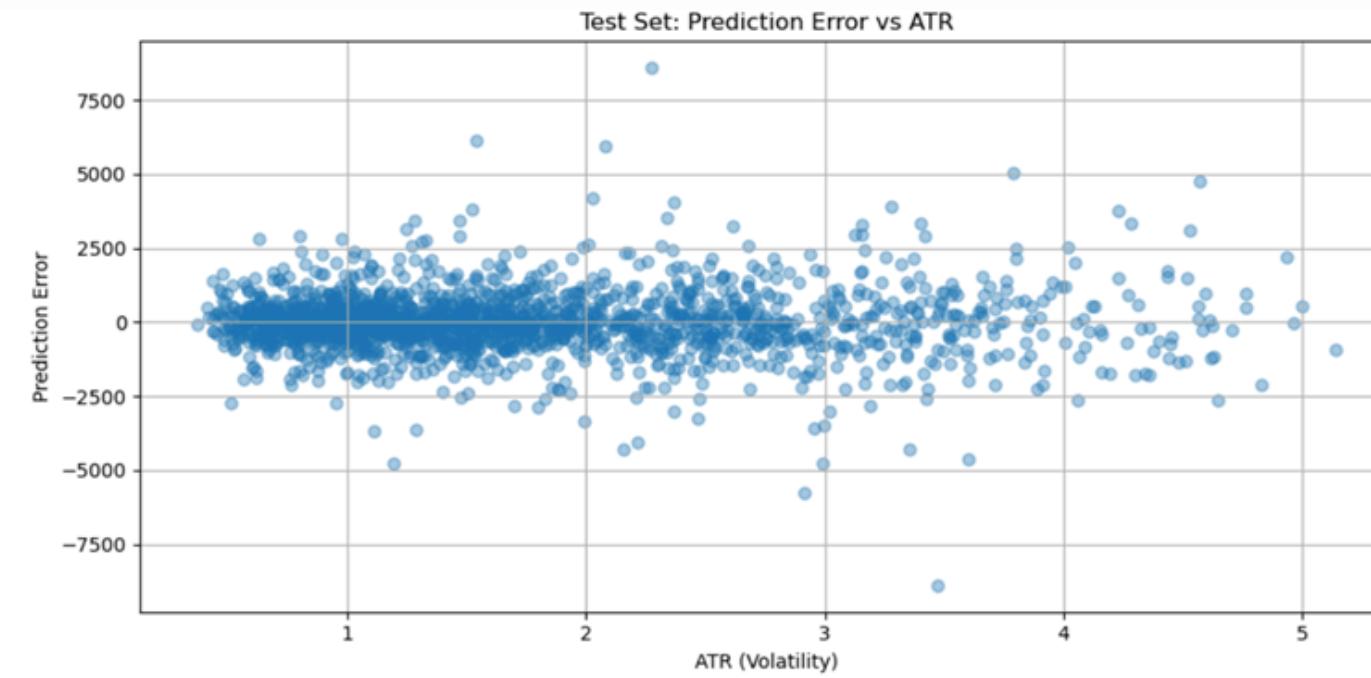


Хистограмата показва разпределението на грешките при прогнозирането. Вижда се, че повечето грешки са съсредоточени около нулата, което означава, че предсказанията са близки до реалността както в тренировъчния, така и в тестовия сет.

Тестова прогнозна грешка vs увереност

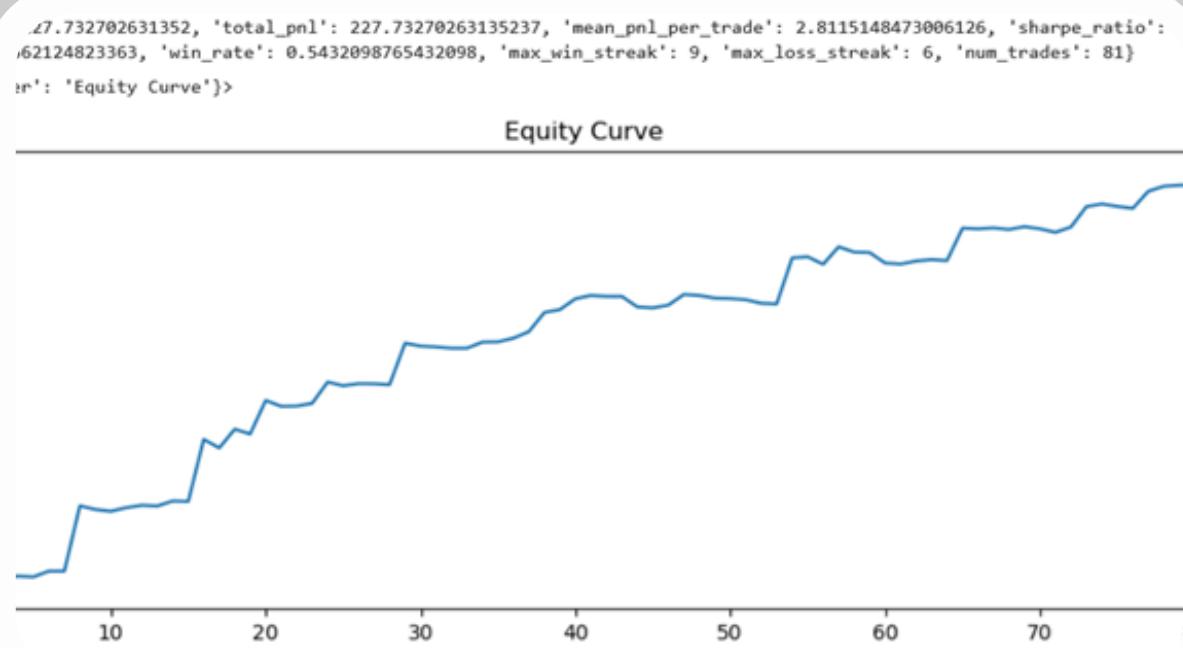


Графиката показва грешка vs увереност. По-ниска увереност води до по-големи грешки — логично.

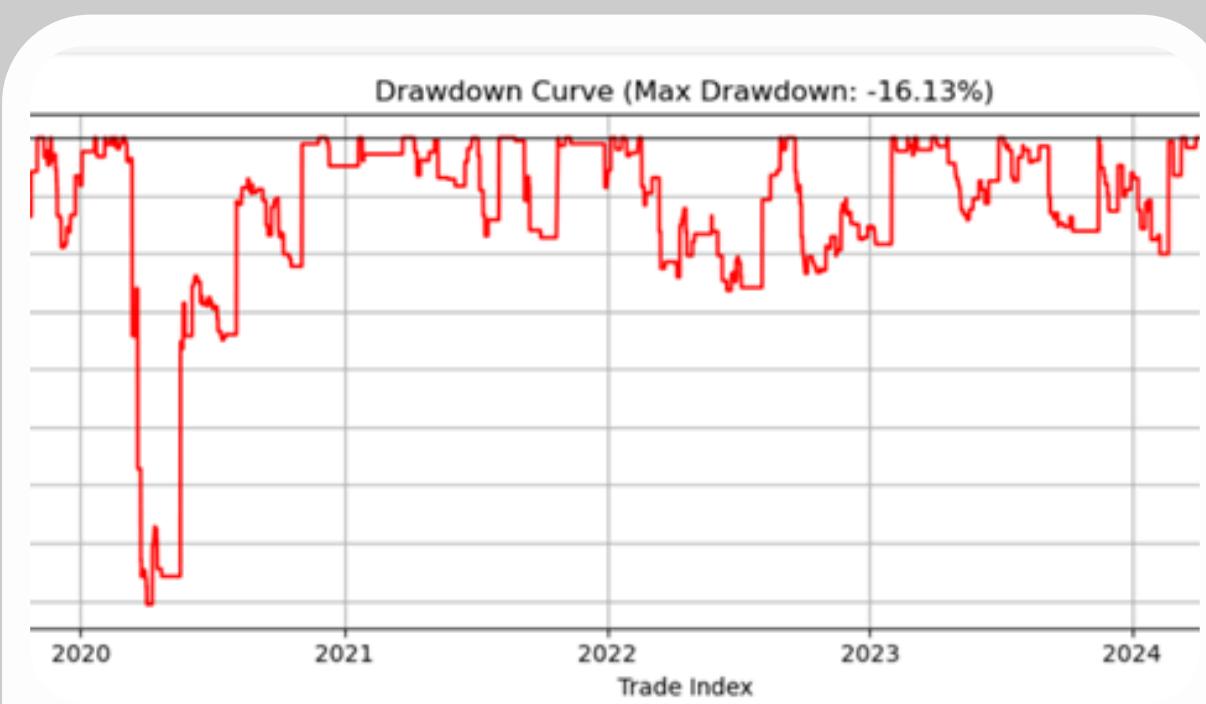


Графиката показва грешка спрямо ATR (волатилност). При ниска волатилност моделът е по-точен. Тоест – можем да филтрираме сигналите според пазарната турбулентност.

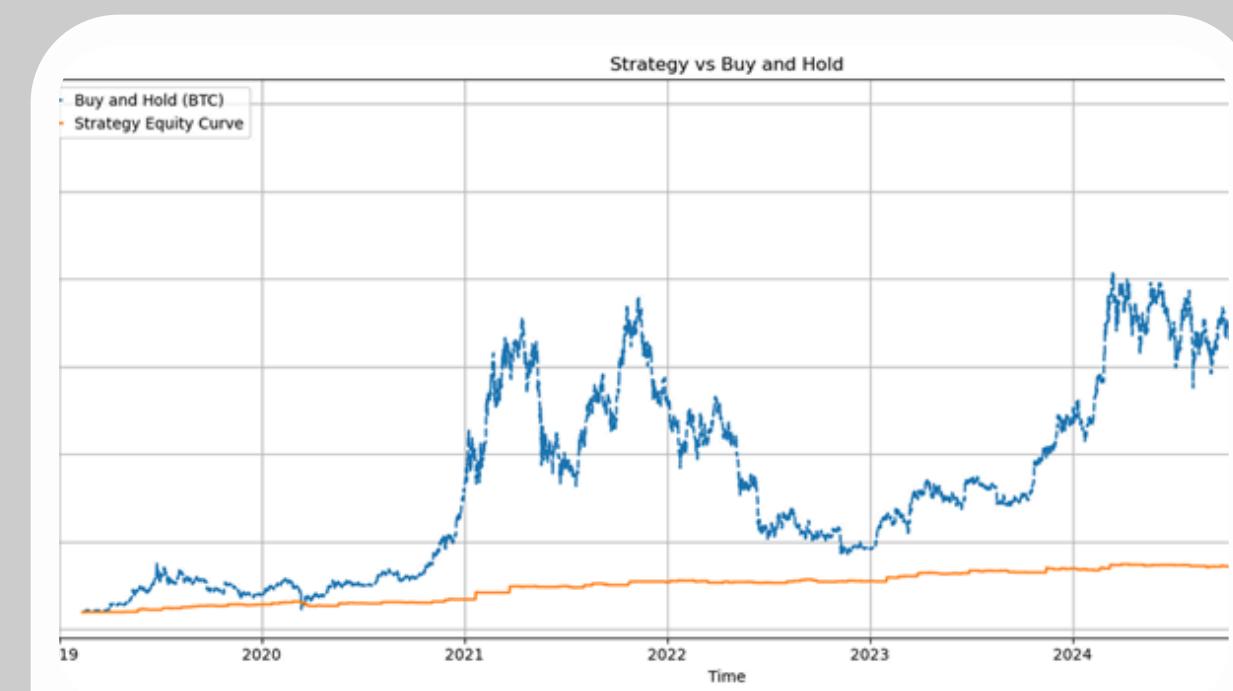
Първи опит за предсказания на пазара (Back Testing)



Графиката показва стабилен растеж.

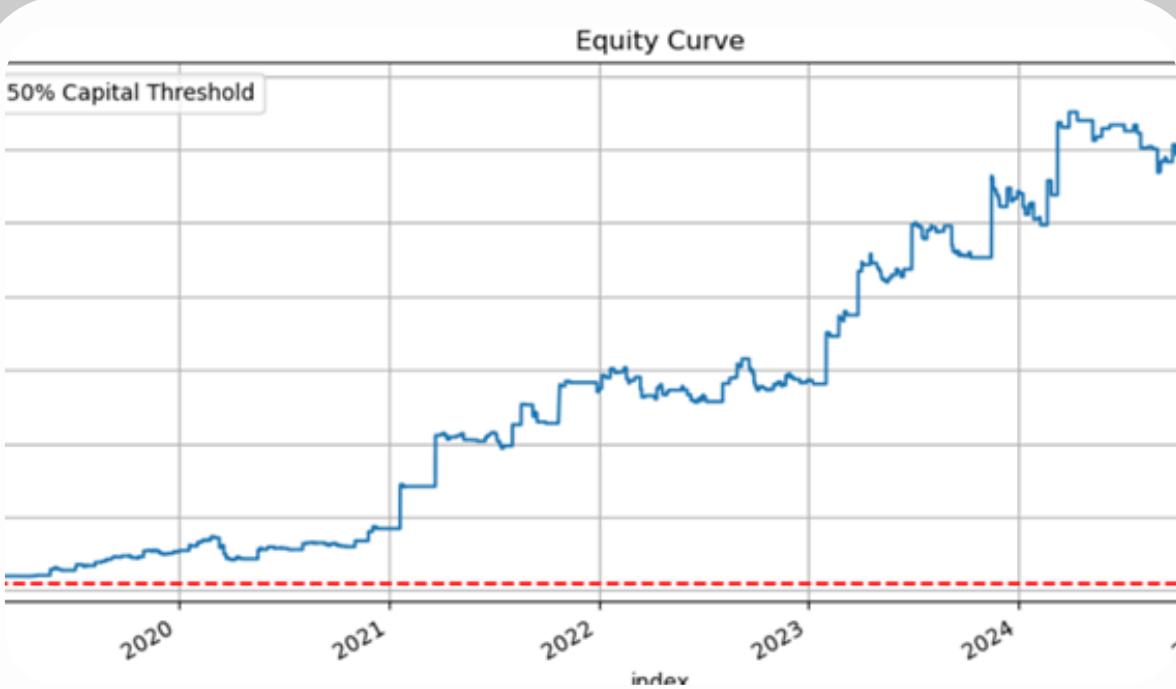


Средната максимална загуба (drawdown) е около 16%, което е приемливо.



На тази графика сравняваме с buy & hold стратегия – нашата се справя по-добре в някои периоди.

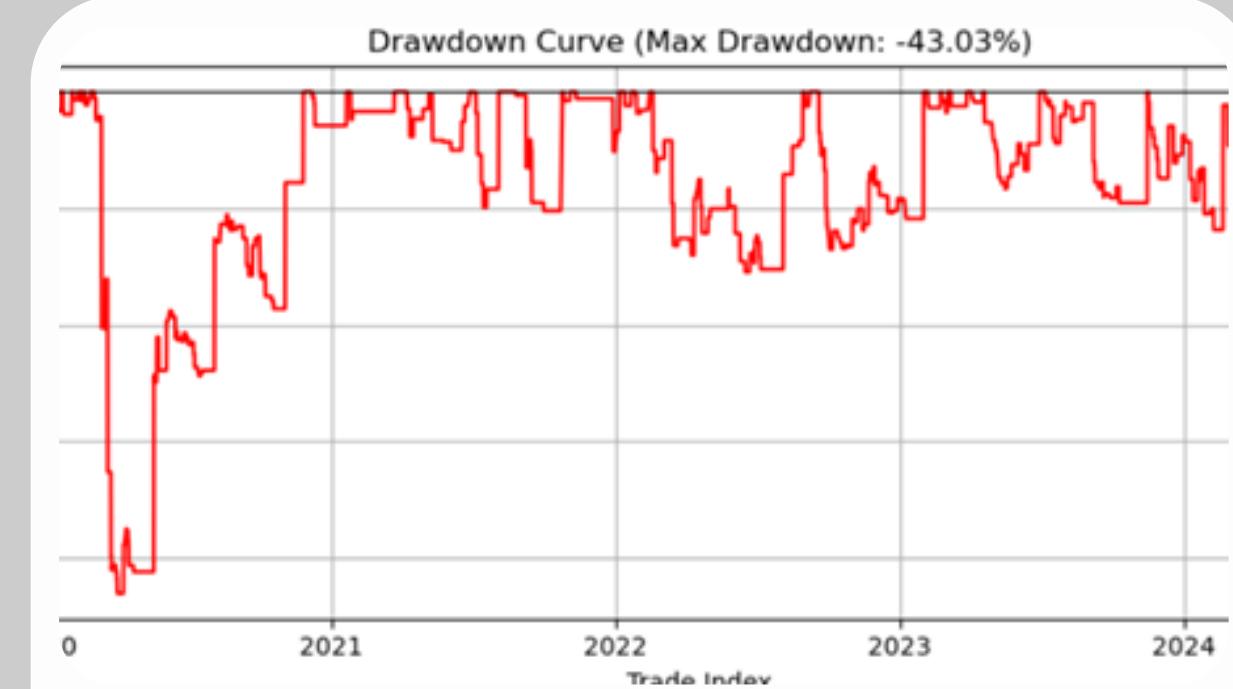
Тестване на стратегията след повишаване на риска



След като увеличихме риска, резултатите останаха положителни.

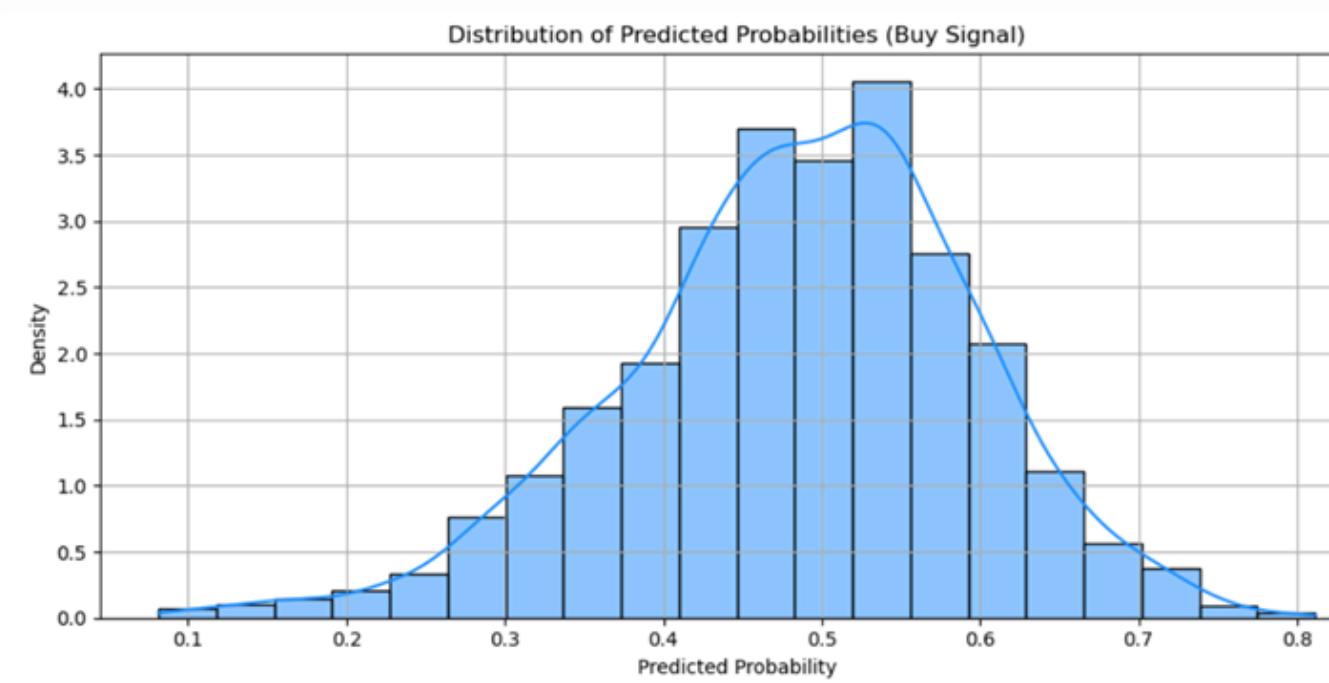


Виждаме, че стратегията се представя стабилно дори при по-агресивна търговия.



Загубите не надхвърлиха лимита от 50% капитал.

Модел за бинарна класификация на база на същата архитектура на модела



Сравнението с BTC buy & hold показва, че сме по-добре в нестабилни периоди.

графика на разпределението на вероятностите, които моделът връща за „сигнал за купуване“. За да избегнем случайни колебания, филтрирахме така:

- ◆ Ако вероятността е над 52%, приемаме сигнал за покупка
- ◆ Под 39% – неутрална позиция

Така изчистихме шума от несигурните случаи.

На графиката се вижда как стратегията се представя срещу класическото „купи и задръж“.

Добрата новина е, че успяхме да надминем тази базова стратегия.

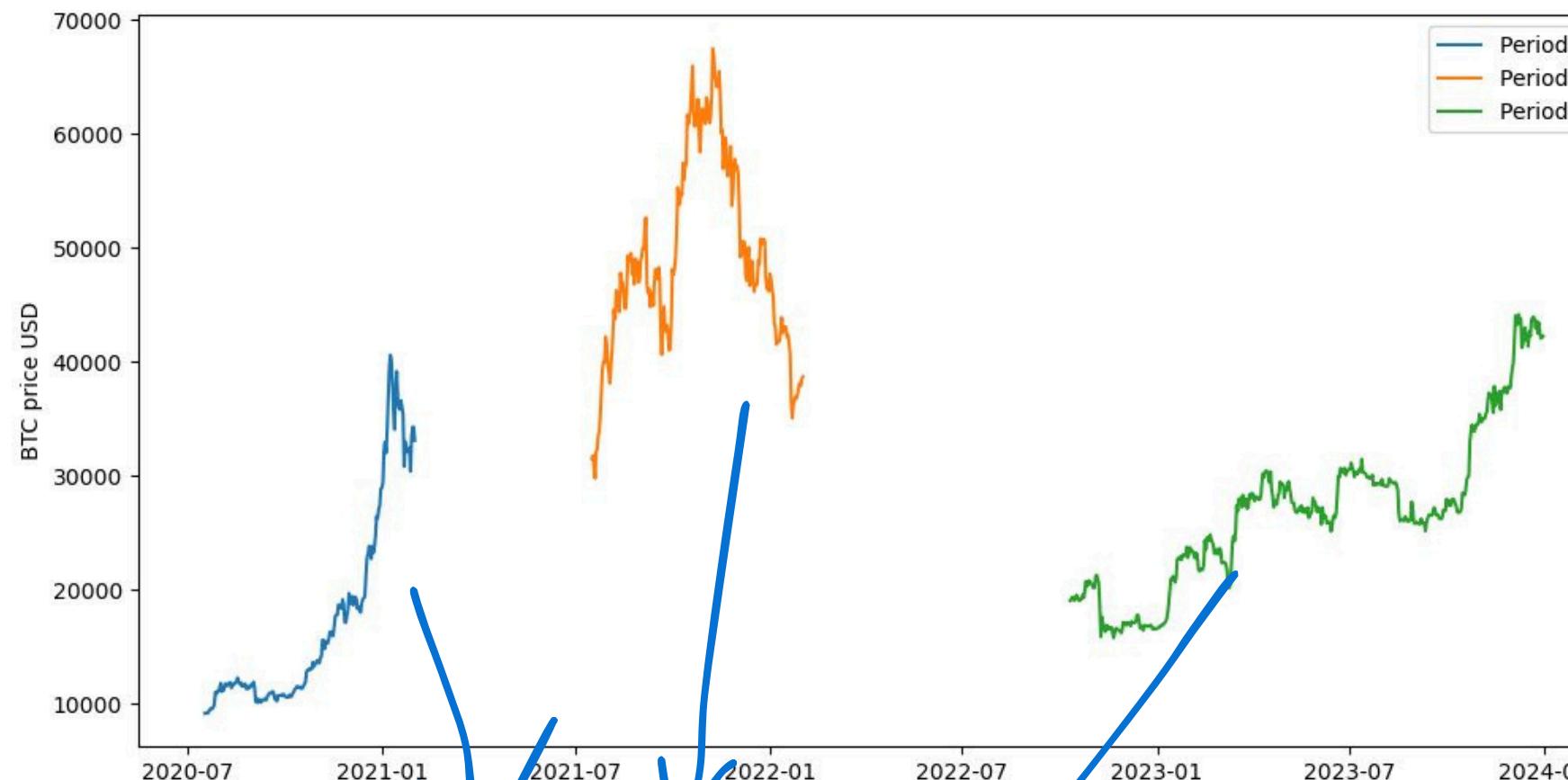
Лошата е, че това е свързано с висока променливост и рисков.

Това ни отвежда до следващата ни цел:

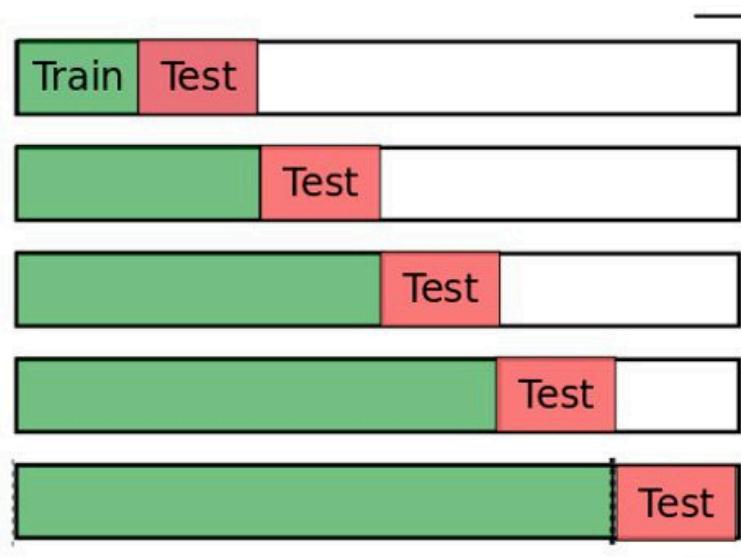
Как да направим по-умен модел, който не само надминава „купи и задръж“, но и го прави с контролирана експозиция към риска.“

Валидация на моделите

Подаваме произволни периоди за валидация.

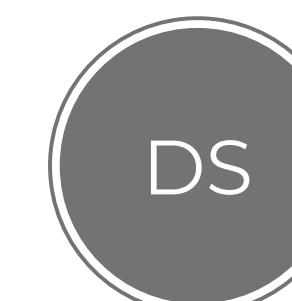


Валидация с един такт напред.



MEAN ABSOLUTE PERCENTAGE ERROR

Каква е грешката в модела?



DIRECTIONAL SYMMETRY

Колко често познаваме посоката на промяна?



STABILITY COEFFICIENT

Съотношение на средната грешка спрямо стандартното отклонение. Т.е. гоколко са стабилни прогнозите ни между отделните валидационни периоди?



COMPUTATIONAL EFFICIENCY

Колко бързо тренираме модела и прогнозираме с него?



ОБОБЩЕН ИНДИКАТОР

Валидация на моделите

MAPE Directional Symmetry Model stability Computational efficiency Combined score

model

ARIMA_212

ARIMA_111

2.253042

2.209730

48.179237

46.462127

0.246164

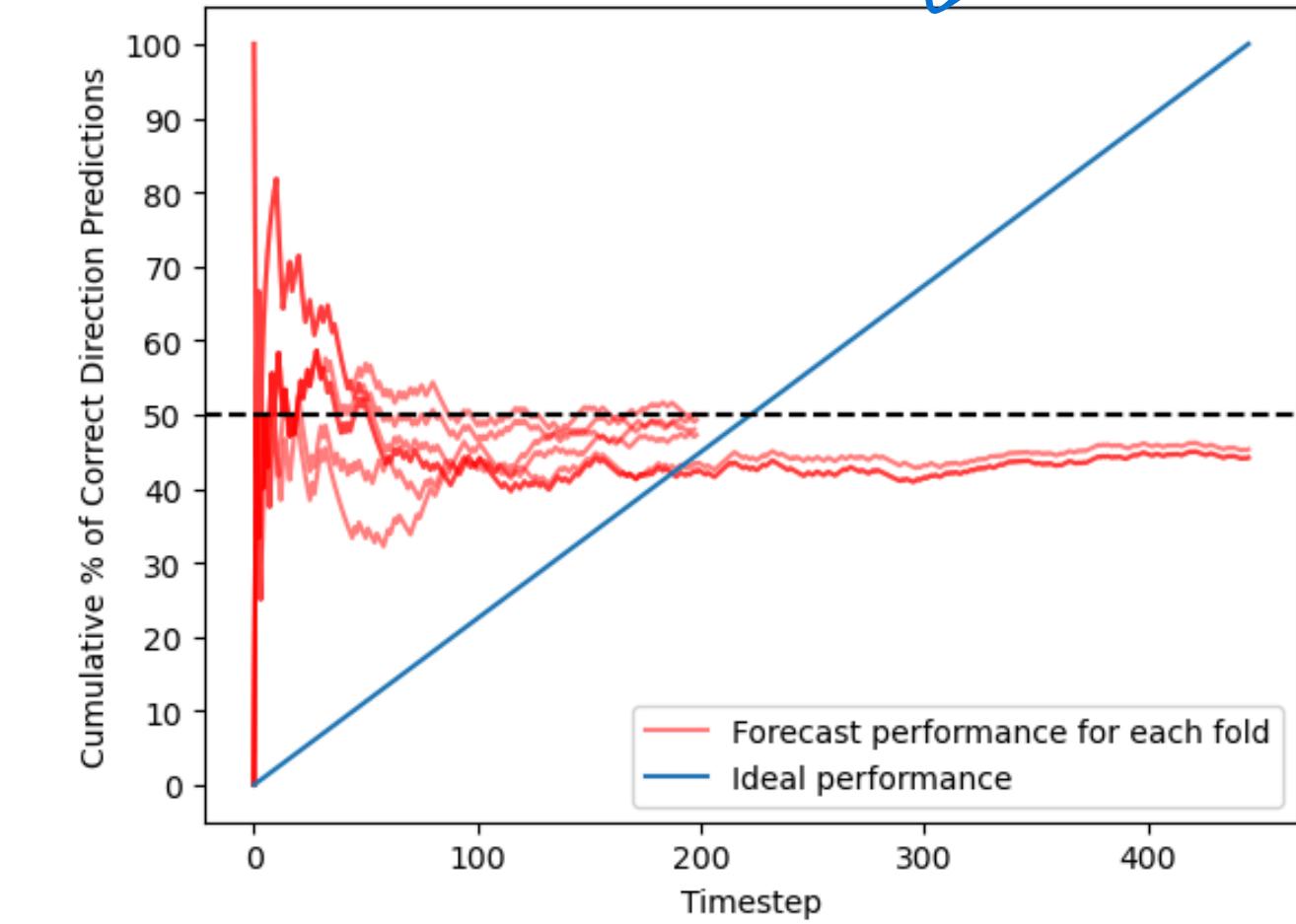
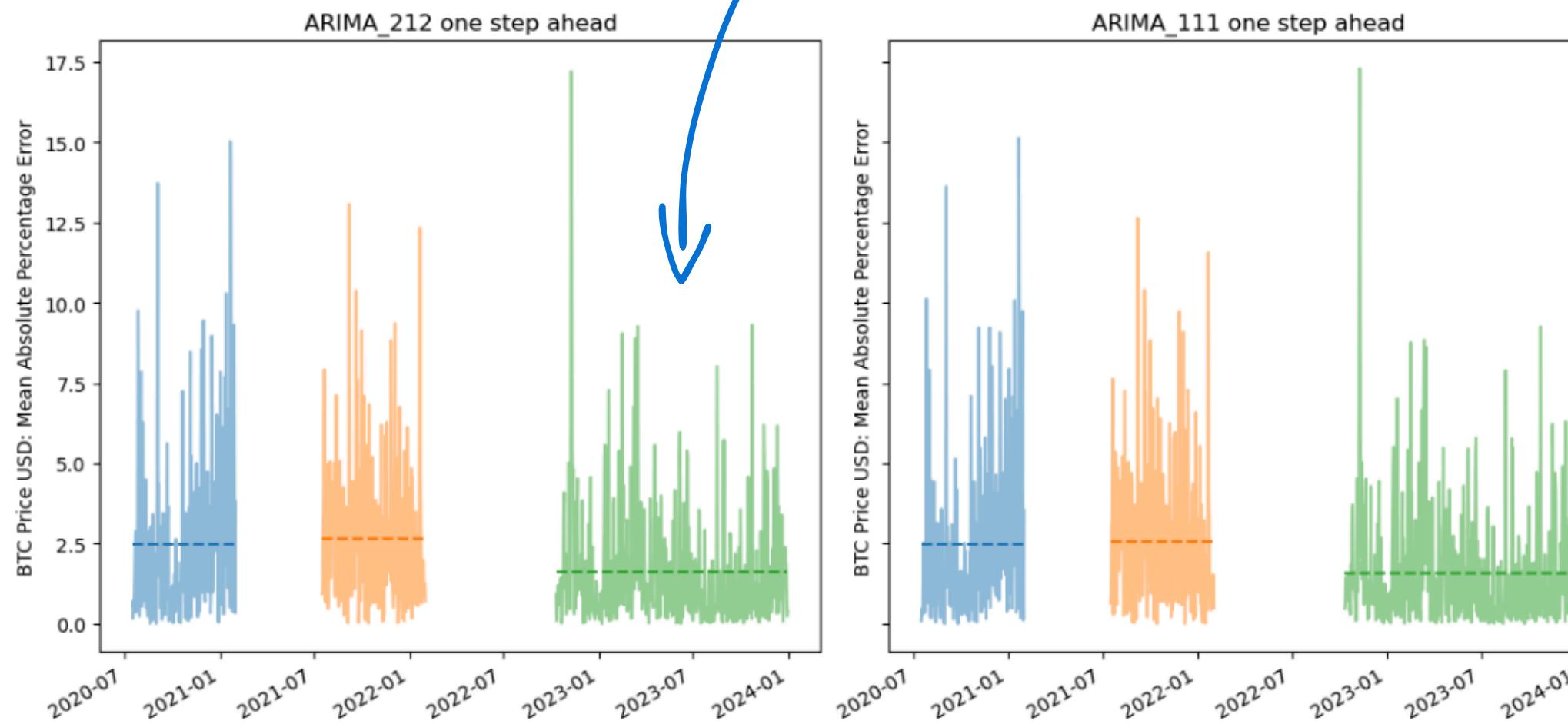
0.247014

2.427117e-06

6.047361e-07

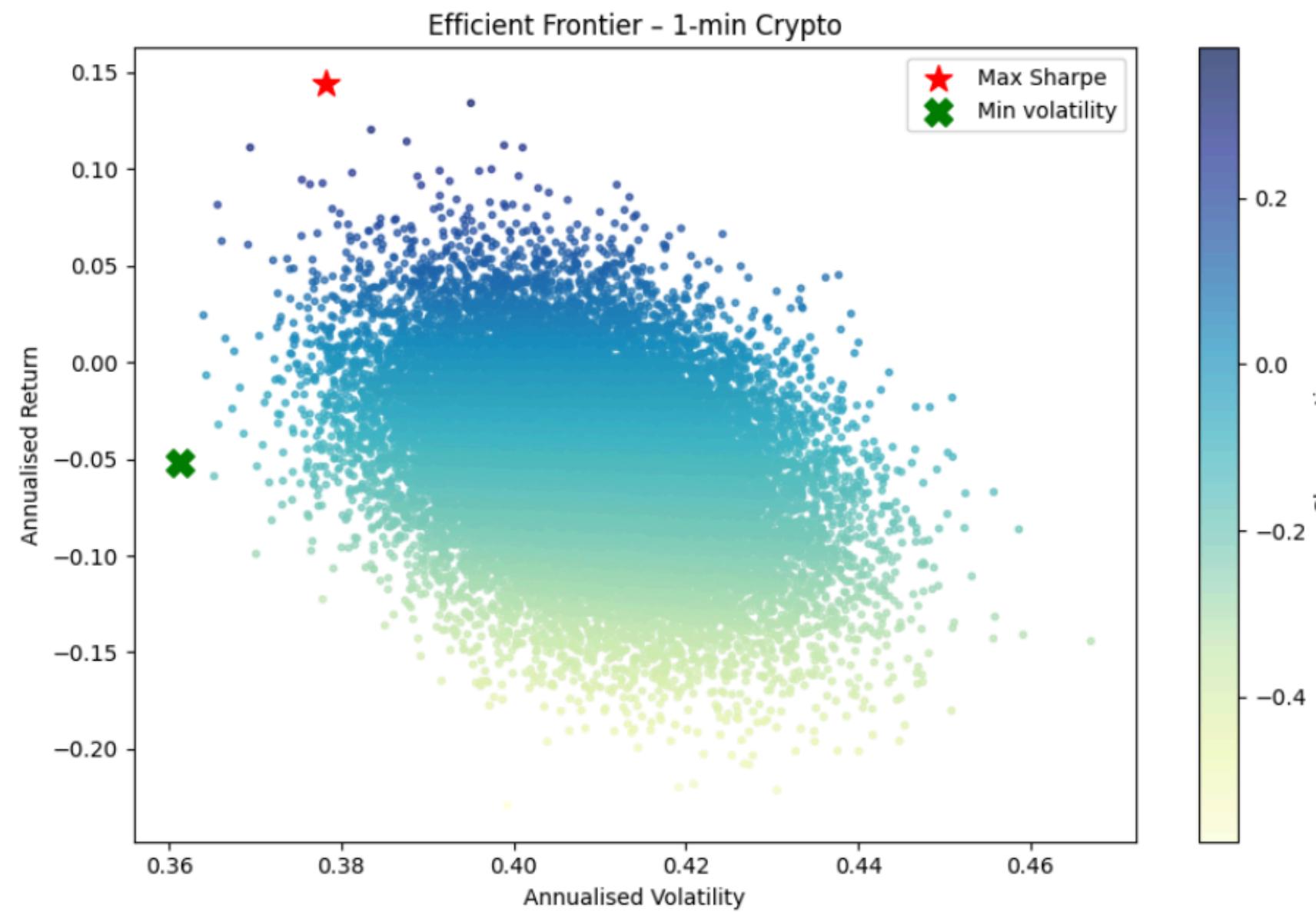
45.680029

44.005383



Портфейлна оптимизация

Монте Карло генериране на портфейлни тегла за оценка на Efficient Frontier.



ПОРТФЕЙЛ С МАКС. ШАРПЕ
Год. доходност : 14.40%
Год. волатилност : 37.82%

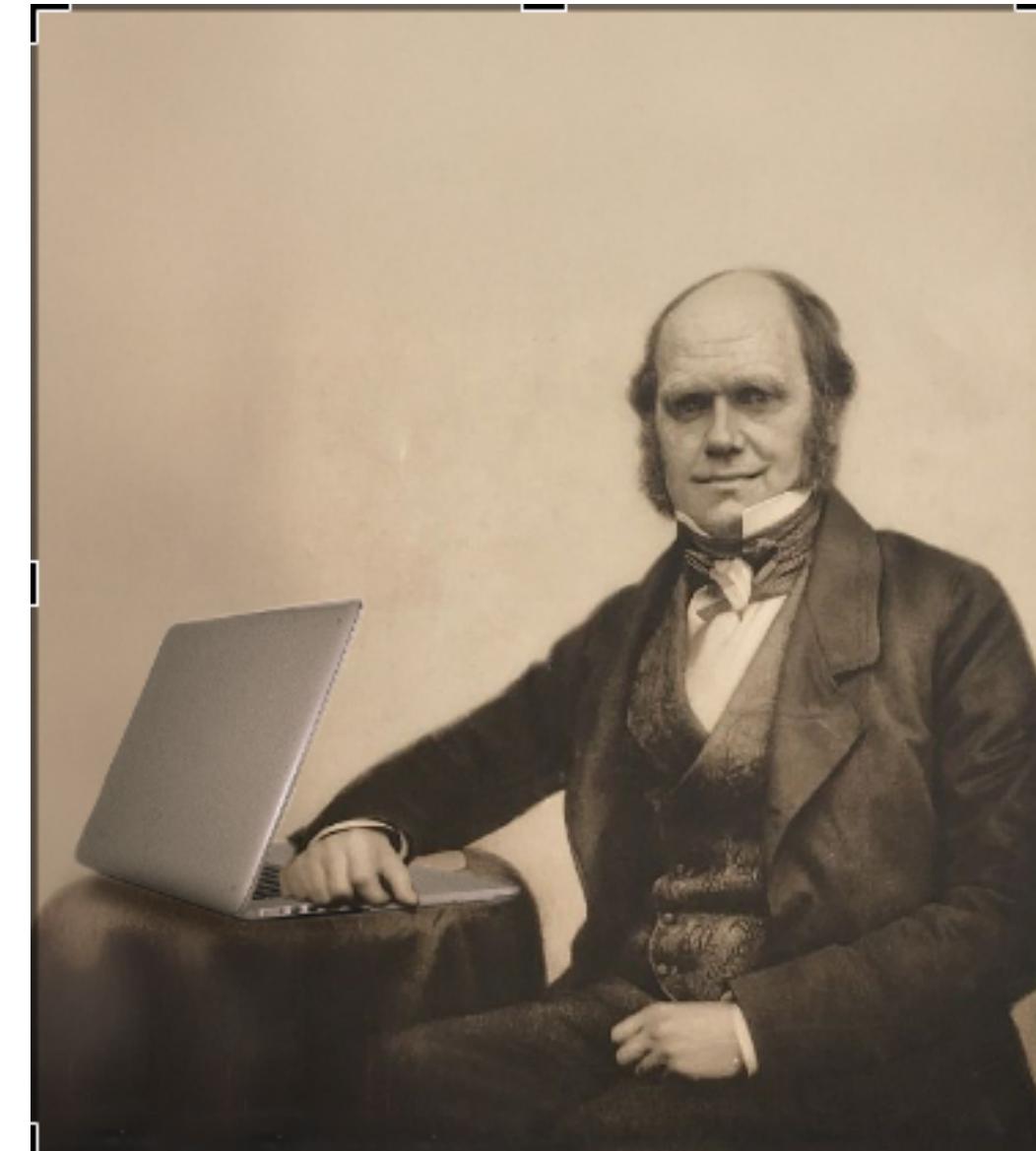
	ADA	ALGO	ANKR	ATOM	BAT	BNB	BTC	CELR	CHZ	COS	...	VET	WAN	WAVES	WIN	XLM
Weight	0.0	0.02	0.02	0.03	0.03	0.02	0.02	0.01	0.01	0.01	...	0.03	0.03	0.03	0.02	0.03

ПОРТФЕЙЛ С МИН. ВОЛАТИЛНОСТ
Год. доходност : -5.24%
Год. волатилност : 36.12%

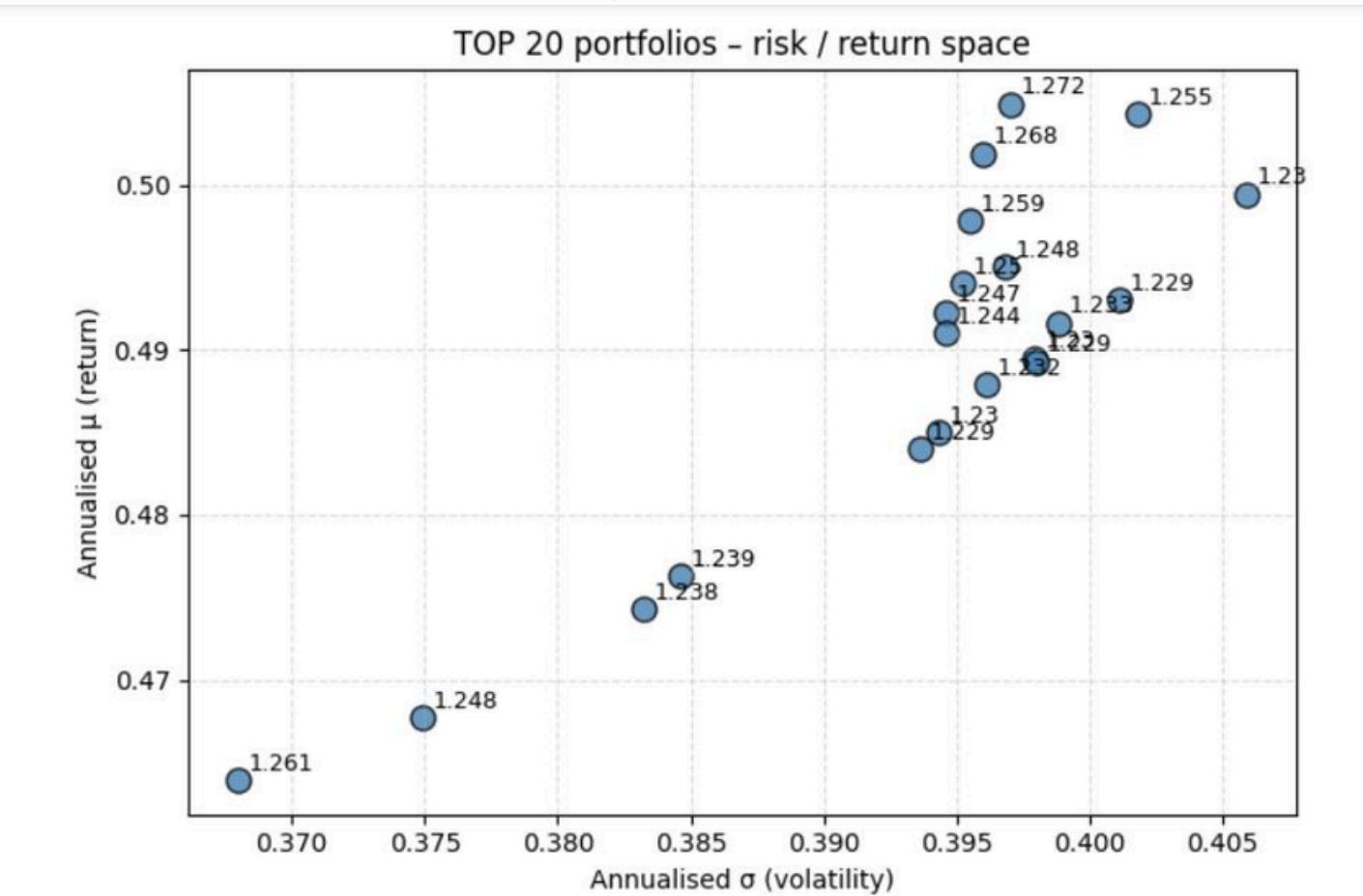
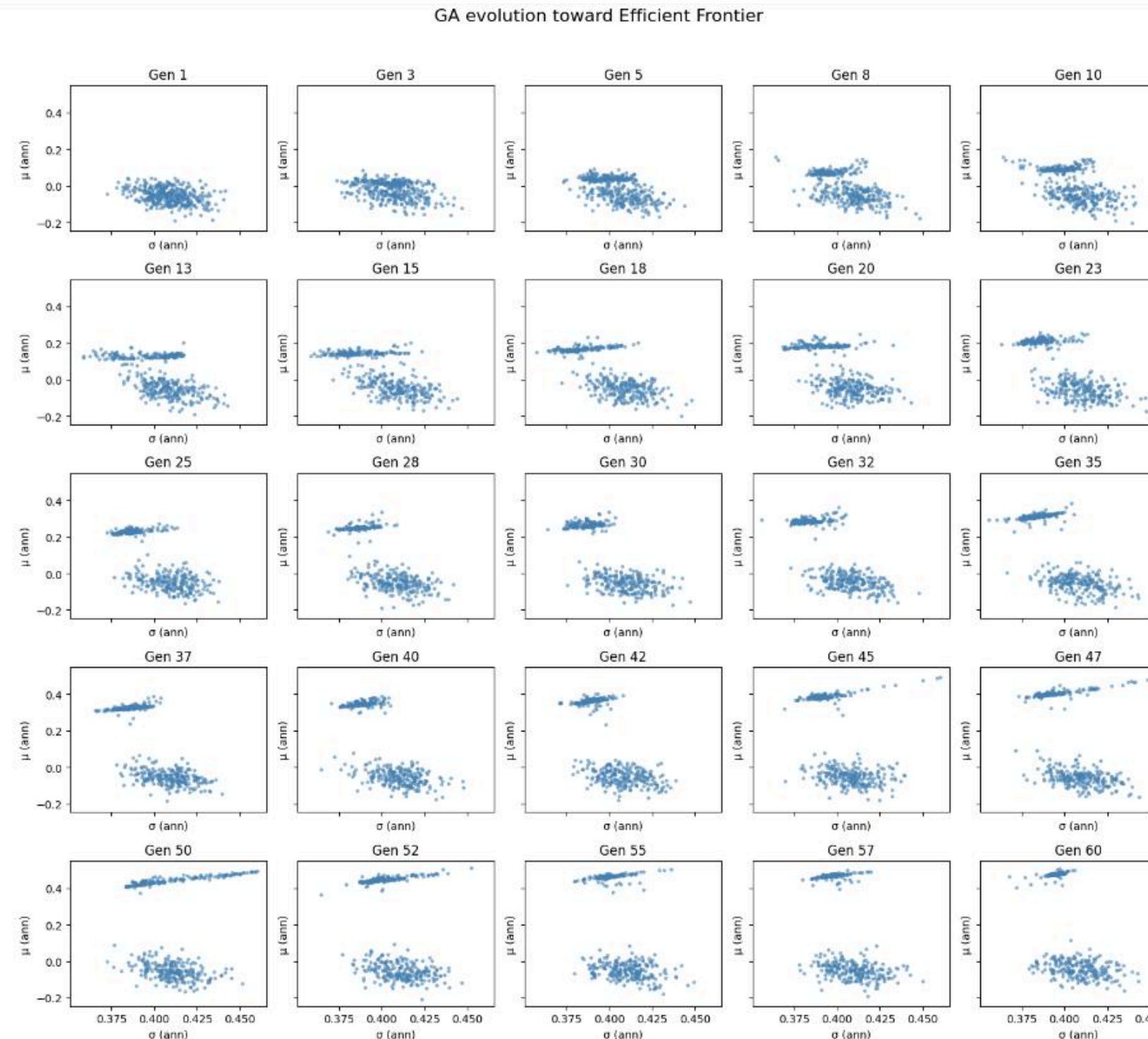
	ADA	ALGO	ANKR	ATOM	BAT	BNB	BTC	CELR	CHZ	COS	...	VET	WAN	WAVES	WIN	XLM
Weight	0.01	0.02	0.0	0.03	0.02	0.04	0.02	0.03	0.03	0.01	...	0.0	0.0	0.0	0.04	0.01

Еволюционни алгоритми (EA)

- Ключова идея: Вдъхновени от естествения отбор.
- Основни стъпки:
 - Инициализация на популация (случайни решения).
 - Оценка (фитнес) на всяко решение (нормфейл).
 - Селекция на най-добрите решения.
 - Рекомбинация (кръстосване) и мутация за създаване на нови решения.
 - Повтаряне, докато бъдат изпълнени критериите за спиране.



Еволюционна порфейлна оптимизация



GA-best Final
\$137,679 CAGR 0.25%



MC-maxSharpe Final
\$132,597 CAGR 0.22%

Еволюцията - как всяко поколение се приближава до границата на ефективност.

**БЛАГОДАРИМ ЗА
ВНИМАНИЕТО!**

**Summer school
on Modeling, AI & Complex Systems 2025**

