This white paper outlines the steps and reasoning adopted for constructing DAREC proprietary performance metrics. They are reported in the dashboard on the lab website's front page.

Mission

This project aims to provide data-driven metrics to evaluate the current performance of the cryptocurrency market. We employ quantitative modeling and machine learning to reach this goal and extract insights from market data. The first application represents the creation of market indexes like traditional finance to have a meaningful benchmark that compares the individual performance of coins and tokens.

Data Sources

We aim to blend different data sources since we acknowledge that market data are not sufficiently reflective of the actual value of a crypto project. However, our first dashboard proposal used market data from the CoinMarketCap API. As a further extension, we aim to query on-chain data from various blockchains and use the two data sources concurrently.

Data Collection

We first downloaded daily data for the top 200 coins and tokens ranked by market capitalization from the above API to create cryptocurrency indexes. The time series of each cryptocurrency can have various lengths, depending on when it started being traded on centralized exchanges and CoinMarketCap began tracking it. Since focusing on performance indicators, we filtered out all stablecoins from our pool of cryptocurrencies. They are stable by nature, and there is no meaning in analyzing a coin pegged to any other asset. We are also aware that the composition of the cryptocurrency universe selected according to the market capitalization criteria can vary over time. Therefore, we are keeping
track of the daily history of prices of more than 1000 coins to include any new entry in the top 200.

**Index Creation**

We created three different indexes using market data:

- **Market-cap-weighted index**, which intuitively resembles renowned financial indexes such as S&P500 or NASDAQ. The cryptocurrencies are weighted by their current market capitalization. The weights are obtained by dividing each coin’s market capitalization by the market’s total market capitalization. The biggest drawback of this approach is consistent over-exposure to BTC and ETH, which cover almost 70% of the market capitalization.

- **Momentum-based index**, which should capture the market’s inertia in following the current upward or downward trend. The weights are obtained through a combination of momentum indicators commonly used in quantitative financial modeling. The index is computed as follows:

  \[
  I_{\text{mom}} = \sum_{i=1}^{N} u_{i}^{\text{mom}} p_i \\
  u_{i}^{\text{mom}} = 0.5w_{i}^{\text{RSI}} + 0.5w_{i}^{\text{ADX}} \\
  w_{i}^{\text{RSI}} = \frac{RSI_i}{\sum_{j=1}^{N} RSI_j} \\
  w_{i}^{\text{ADX}} = \frac{ADX_i}{\sum_{j=1}^{N} ADX_j}
  \]

  The weights are obtained through an equal combination of two momentum indicators. The weights for each technical indicator are computed as those for the market capitalization and then combined equally into a unique weight to construct our momentum index. The relative Strength Index (RSI) considers prices at the end of the day. If the current closing price is higher than the previous one, it is regarded as a gain. Average gain averages difference between closing prices. The average loss averages those differences where the closing price is less than the previous one. But bear in mind that the average loss is a positive value.

  - Average gain = \[\frac{\text{[(previous average gain) (n-1) + current average gain]}}{n}\]
  - First average gain = total gain in the first n periods / n
  - Average loss = \[\frac{\text{[(previous average loss) (n-1) + current average loss]}}{n}\]
First average loss = total losses in the first n periods / n

\[ RSI_t = 100 - \frac{100}{1 + RS} \]
\[ RS_t = \frac{\text{averagegain}}{\text{averageloss}} \]

The other element is the Average Directional Movement Index (ADX), which highlights the trend strength. It is essential to check if the market is trending or moving sideways. The ADX is a Welles Wilder-style moving average of the Directional Movement Index (DX). The values range from 0 to 100 but rarely get above 60. To interpret the ADX, consider a high number a strong trend and a low number a weak trend.

\[ ADX = EMA \left( 100 \ast \frac{|DI^+ - DI^-|}{DI^+ - DI^-} \right), \]

where the EMA is the exponential moving average of the prices computed over a certain amount of past historical observations.

• **Volatility-based index**, which takes into account the degree of price volatility in the cryptocurrency market. The cryptocurrencies are weighted according to the Average True Range (ATR). The first step in calculating ATR is to get the True Range (TR). The true range is based on high and low values of the current period (H and L) and close values of the previous period. The second step is a calculation of ATR using values of TR. For ATR period is usually used value of 14.

The index is calculated as

\[ I_{vol} = \sum_{i=1}^{N} w_i^{ATR} p_i \]
\[ w_i^{ATR} = \frac{ATR_i}{\sum_{j=1}^{N} ATR_j} \]

where the ATR is computed as

\[ TR_t = MAX \left( |H_t - L_t|, |H_t - C_{t-1}|, |L_t - C_{t-1}| \right) \]
\[ ATR_0 = TR_0 \]
\[ ATR_0 = (n - 1) \ast ATR_{t-1} + \frac{TR_t}{n} \]
Performance measures

Once the indexes are constructed according to the previous section, it is possible to express the performance of each index constituent in terms of those several market benchmarks. The DAREC dashboard on the front page currently contains two different sections: Sharpe Ratios and Betas. The Sharpe Ratio table contains the excess Sharpe Ratio of each coin concerning the benchmark indicated in the header. Each column’s title includes the benchmark index’s name, referring to the three indexes described above, the horizon for calculating the Sharpe Ratio, and the Sharpe Ratio of the index itself. Each entry of such table is computed as

\[
SR = \frac{R_p}{\sigma_p}
\]

where:

- \( R_p \) return of coin/token/index,
- \( \sigma_p \) = standard deviation of the coin/token/index’s excess return

\[
\Delta SR_i = SR_i - SR_{\text{benchmark}}
\]

\( SR_{\text{benchmark}} \) is the number in each column’s header, while Delta \( SR_i \) is the excess Sharpe ratio for every coin/token in the rows of the table. These performance metrics are calculated concerning the three indexes constructed for three different horizons (1Y,3M,1M). All the metrics are obtained from daily data. Changing the horizon from 1Y to 3M implies that the mean and the standard deviation of returns are computed over that horizon using daily data.

The Beta table contains the results of the renowned CAPM estimate, regressing each coin/token return against the three constructed benchmarks. The regression estimate is again performed over three different time horizons. If the table contains any missing values, this specific cryptocurrency was not entirely
traded for the selected horizon (it may be because it started being traded after that date). The interpretation of the estimated beta is as follows.

- $\beta = 1$ exactly as volatile as the market
- $\beta > 1$ more volatile than the market
- $\beta < 1 > 0$ less volatile than the market
- $\beta = 0$ uncorrelated to the market
- $\beta < 0$ negatively correlated to the market

**Frequency of updates**

The dashboard can be quickly updated due to the level of automation reached in querying, modeling, and presenting the data. Our current choice is to update every week, On Monday at 8:30 am Eastern Time (ET), but this can be easily adapted to our needs and move the updates to an higher frequency.

**Further improvements**

Our project is a constant work in progress that still requires improvement:

- On-chain data as additional data sources to leverage different characteristics of each project (network size, adopters, liquidity on DeFi, etc.) and create unique performance metrics.
- Enhancing the features of the dashboard. Clickable column ordering, brief descriptions of the metrics by hovering over the header of each column, a dynamic chart for the created indexes, and additional performance metrics based on market data (absolute returns, for instance). We will also add the upload date to keep track of the timeline for the performance presented.
- Adding a predictive component by engineering a machine learning model that can output the weight of a new cryptocurrency index. We currently use traditional methods to construct our indexes that resemble traditional finance (market cap of technical indicators). Once we collect enough data on-chain, it would be interesting to produce a data-driven model that can adjust weights based on the pattern retrieved there.