GREEN TRANSITION

GrEnFIn Summerschool 2024

Nursat Kulumkanov, Bob Thomas, Maria Luna Ariano, Andrea Dosi, Orsola Ongarato, Matteo Ravagni, Coline Viratelle, Franziska Spitzauer, Siyu Wu, Miguel Armayor Martínez, Marius Korthals Altes,
Roberto Baricchia and Francesco Pascullo.

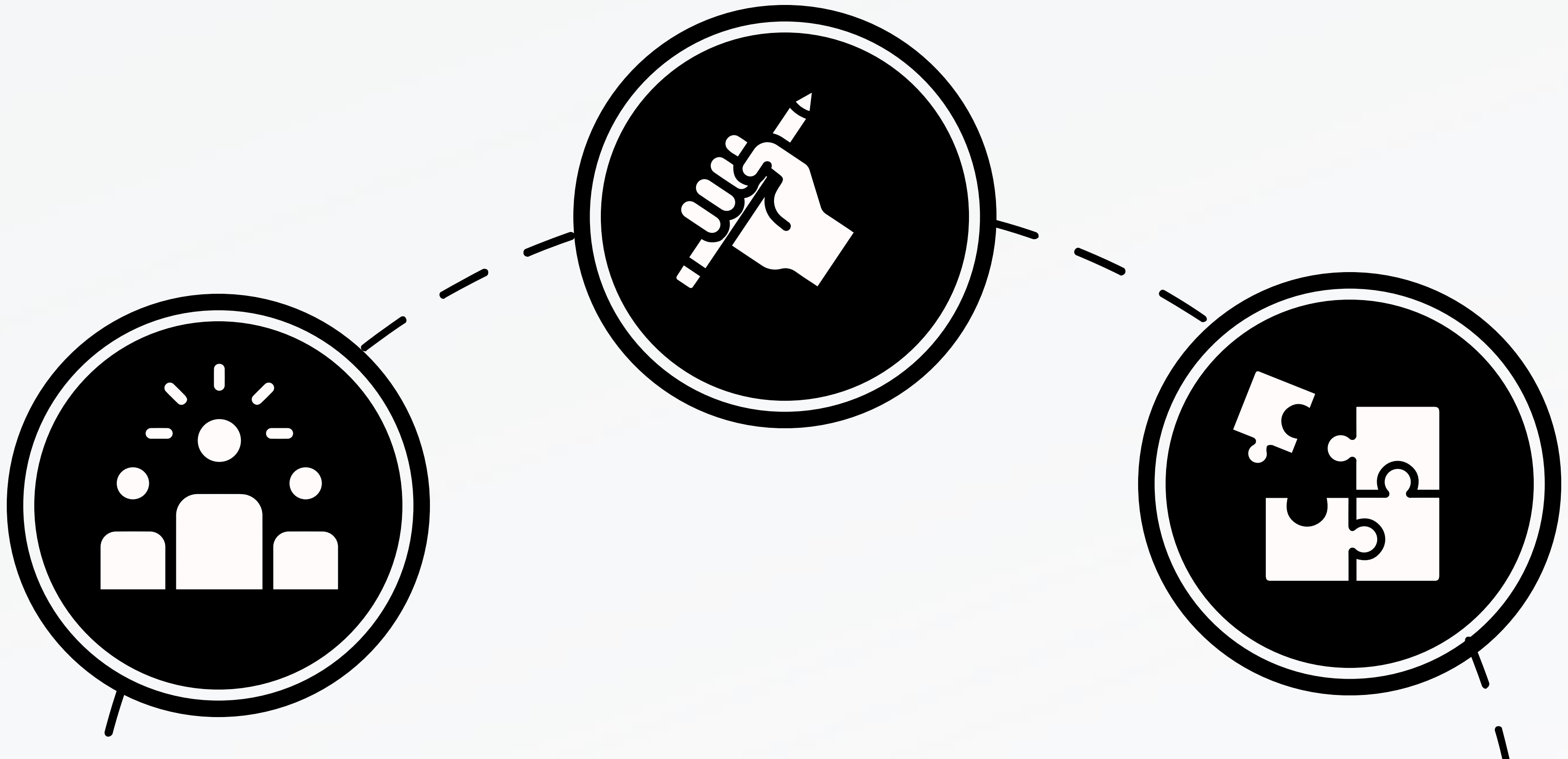
,

CONTENT

01) PROJECT “GREEN TRANSITION”

Analysis of „greenness“ – **green bonds** versus brown bonds

Objective n° 1	What is the role of greenium in **credit risk**?
Objective n° 2	What about **physical risk** and **transition risk**?



02) MERTON'S MODEL

Credit risk is defined as the risk that a borrower is not able to meet its financial obligations on time.

MERTON'S MODEL:

- Used to judge a corporation's risk of credit default
- Two possible scenarios at time T:

  1) $\boldsymbol{A_t > K}$: no default

  (t = T)

  2) $\boldsymbol{A_t \leq K}$**:** default

- Thus the value of a firm's equity at time T equals the payoff of a European call option on A_t:

  $E_t = max(A_t - K, 0) = (A_t - K)^+$. **(t = T)**

ASSUMPTIONS OF MERTON'S MODEL:

- Perfect and frictionless market
- Firm finances itself by equity and debt, debt consists of zero-coupon bonds with maturity T
- Values at time t of equity and debt are denoted by E_t and K_t
- A_t : Value of the firm's asset at time t
- K: Nominal value of debt

- A_t follows a geometric Brownian motion, thus

$$dA_t = \mu A_t dt + \sigma A_t dz_t, \quad A_0 > 0 \quad \text{and} \quad \ln A_T \sim N\left(\ln A_t + \left(\mu - \frac{\sigma^2}{2}\right)(T-t), \sigma^2(T-t)\right)$$

- Probability of default at time t: $P_t = \Pr(A_T \leq K) = \Phi\left(-\frac{\ln(A_t) + \left(\mu - \frac{\sigma^2}{2}\right)(T-t) - \ln(K)}{\sigma\sqrt{T-t}}\right)$

- The **distance to default** is the number of standard deviations that the firm‘s asset value is away from the default and can be defined as:

$$DD = \frac{\ln\left(\frac{A_t}{K}\right) + \left(\mu - \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$$

- <u>Limit</u>: market value of assets and volatility of assets are not observable, but thanks to Merton‘s Model we can use the standard Black Scholes call formula:

$$E_t = A_t\Phi(d_1) - Ke^{-r(T-t)}\Phi(d_2) \quad \text{with} \quad d_1 = \frac{\log\left(\frac{A_t}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}} \qquad d_2 = d_1 - \sigma\sqrt{T-t}$$

03) APPLYING MERTON'S MODEL

Aim: Calibrate the quadratic loss function

$$minarg_{\{A_t,\sigma\}} L(A_t,\sigma) = \sum_{t=1}^{N} (E_t^{Market} - E_t^{BS}(A_t,\sigma))^2$$

Problem: - $\{E_t\}_{t=1,\dots,N}$: observed equity values, thus *N* known variables ↯

- $L(A_t,\sigma)$: *N + 1* unknown variables

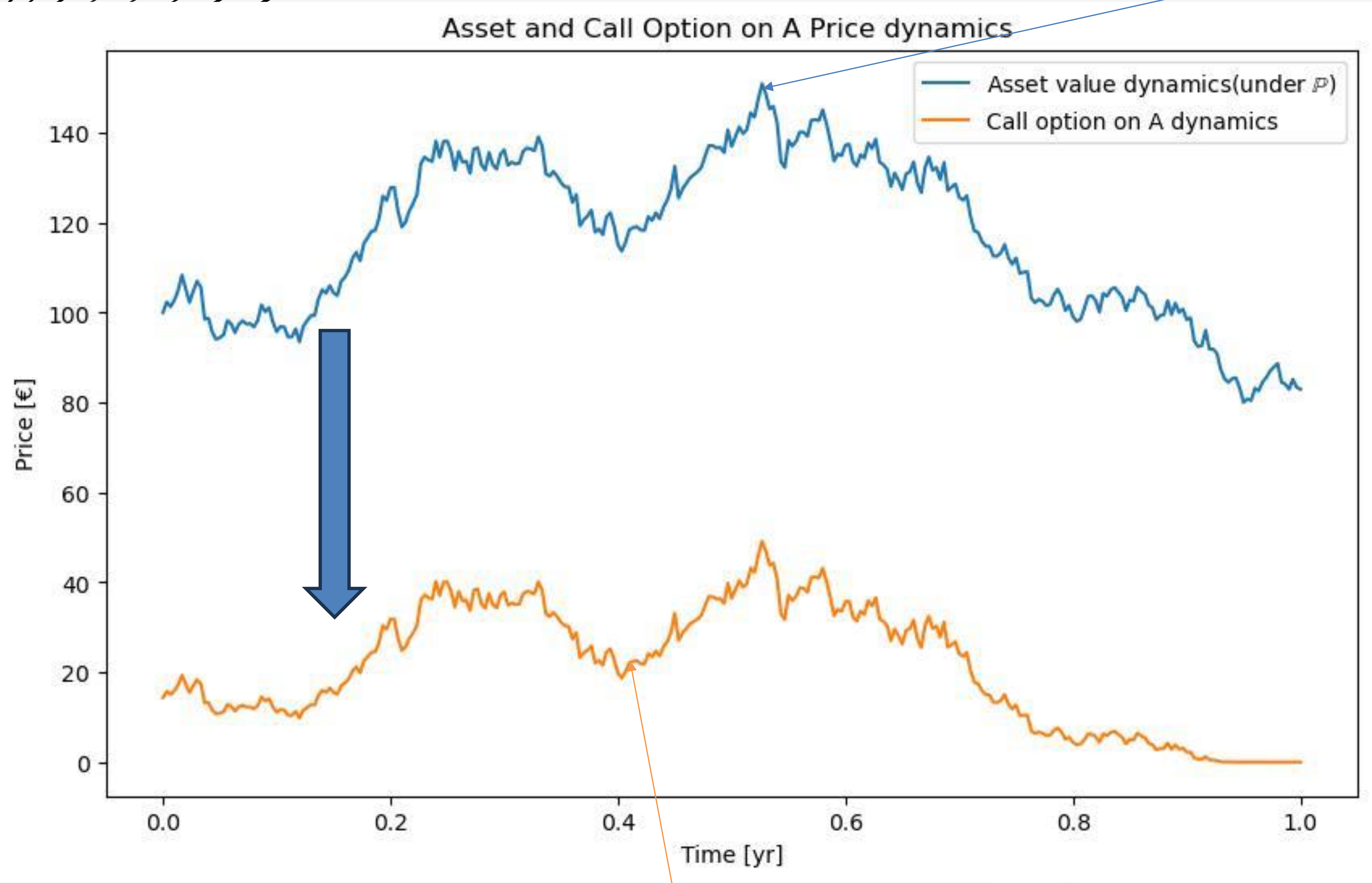
Simplification: Set $A_t = \alpha A_t^{Book} + \beta,\ \alpha,\beta \in \mathbb{R}$.

Determine α,β,σ via minimizing the loss function:

$$L(\alpha,\beta,\sigma) = \sum_{t=1}^{N} (E_t^{Market} - E_t^{BS}(\alpha,\beta,\sigma))^2$$ with

$$E_t^{BS}(\alpha,\beta,\sigma) = (\alpha A_t^{Book} + \beta)\Phi(d_{1,t}) - K e^{-r_t(T-t)}\Phi(d_{2,t})$$

$$dA_t = A_t(\mu dt + \sigma dW_t^{\mathbb{P}})$$

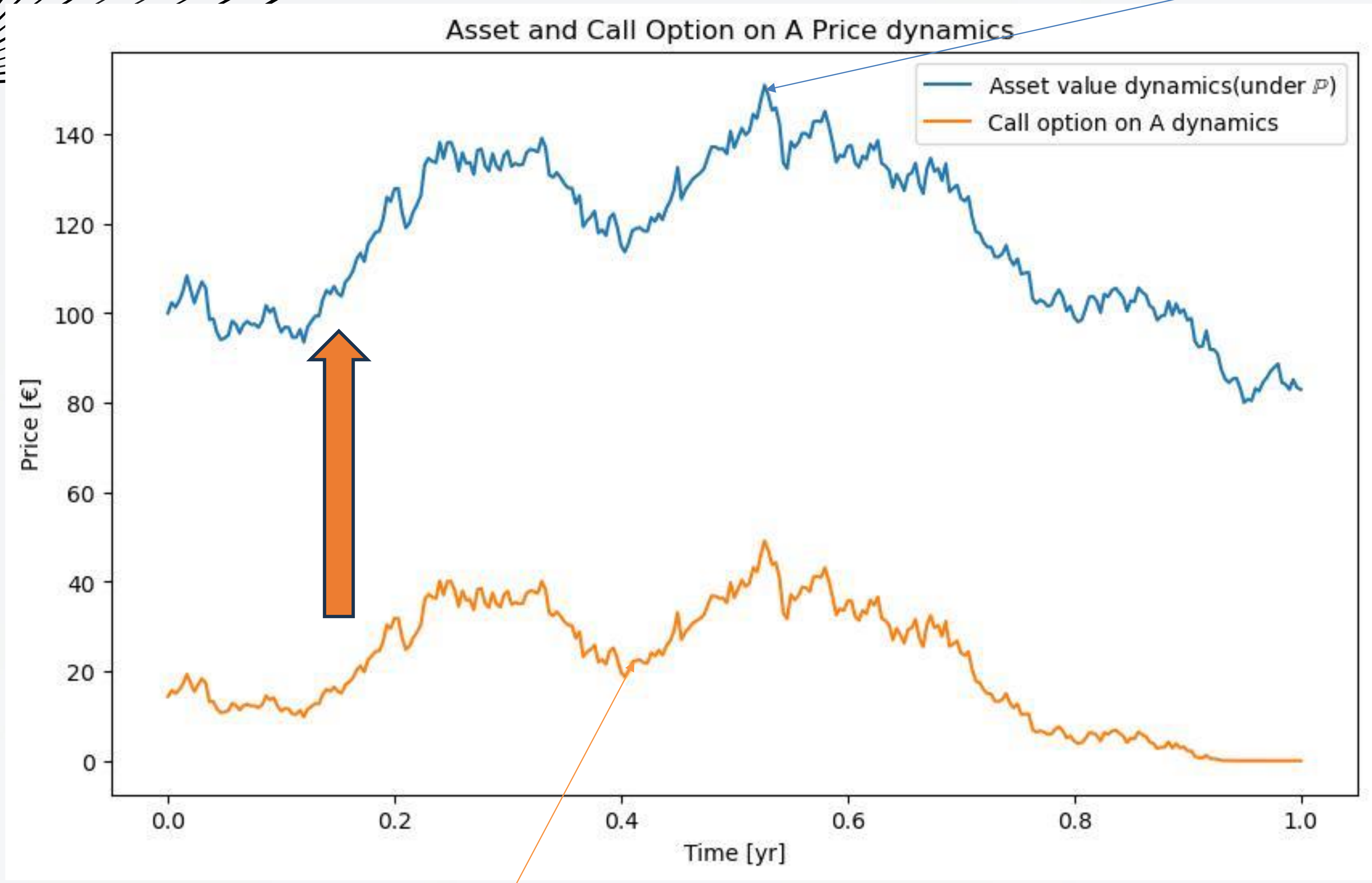


$$d_1 = \frac{\log\left(\frac{A_t}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$$

$$d_2 = d_1 - \sigma\sqrt{T-t}$$

$$BS_t = A_t\Phi(d_1) - Ke^{-r(T-t)}\Phi(d_2)$$

$$dA_t = A_t(\mu^* dt + \sigma^* dW_t^{\mathbb{P}})$$



$$A_t = \alpha^* A_t^{book} + \beta^*$$

$$d_1(\alpha,\beta,\sigma) = \frac{\log\left(\frac{\alpha A_t^{\text{book}} + \beta}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$$

$$d_2(\alpha,\beta,\sigma) = d_1(\alpha,\beta,\sigma) - \sigma\sqrt{T-t}$$

$$E_t \approx BS_t(\alpha^*, \beta^*, \sigma^*)$$

$$BS_t(\alpha,\beta,\sigma) = (\alpha A_t^{\text{book}} + \beta)\Phi(d_1(\alpha,\beta,\sigma)) - Ke^{-r(T-t)}\Phi(d_2(\alpha,\beta,\sigma))$$

04) DATA FOR MERTON'S MODEL

Input Data:

- E_t^{Market}: Market value of equities
- K_t: Book value of liabilities
- r: Risk free rate per country
- $A_t^{Book:}$ Book value of asset
- T – t : Difference between the redemtption date and the current date

04) DATA FOR MERTON'S MODEL

$$\left(\frac{\left(\frac{\text{Total Debt}}{\text{Equity}}\%\right)}{100}\right)^{-1} \times \text{Book Value per Share} = D$$

$$\left(\left(\frac{\text{Total Debt}}{\text{Total Assets}}\%\right)/100\right)^{-1} \times D = A_t$$

05) IMPLEMENTATION OF MERTON'S MODEL

The class was created in **Python** to create a Merton's model. It defines several functions, which are asset_value, d1, d2, calculate_equity_value

Following variables were defined: D, T, t, r, Vb. The class was used for calculation of the objective function (that calculates the squared difference between real and calculated equity values using the Merton model.)

The objective function was defined and minimized to find the optimal value of alpha, beta, and sigma values. The objective function result needs to be around 0 for a proper optimization

```python
class Model:
    def __init__(self, D, T, r, t, Vb):
        self.D = D
        self.T = T
        self.t = t
        self.r = r
        self.Vb = Vb

    def asset_value(self, alpha, beta):
        V = alpha * self.Vb + beta
        return max(V, 1e-5)  # Ensure V is positive
```

```python
def d1(self, alpha, beta, sigma):
    V = self.asset_value(alpha, beta)
    return (np.log(V / self.D) + (self.r + 0.5 * sigma ** 2) * (self.T - self.t)) / (sigma * np.sqrt(self.T - self.t))

def d2(self, alpha, beta, sigma):
    return self.d1(alpha, beta, sigma) - sigma * np.sqrt(self.T - self.t)

def calculate_equity_value(self, alpha, beta, sigma):
    V = self.asset_value(alpha, beta)
    d1 = self.d1(alpha, beta, sigma)
    d2 = self.d2(alpha, beta, sigma)
    equity_value = V * norm.cdf(d1) - self.D * np.exp(-self.r * (self.T - self.t)) * norm.cdf(d2)
    return equity_value
```

1. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html

OPTIMIZATION

The minimization function was used for minimizing objective_function.
The optimzed parameters are alpha, beta, and sigma

The loop was made for objective function is to sum up the values for every period.

Because of data incosistency, data processing was done (Min-Max scaling). This process ensures that the features contribute equally to the analysis or model, preventing features with larger magnitudes from dominating those with smaller magnitudes. It provides a convenient way to normalize your data, ensuring all features contribute equally to the model's performance (1).

```python
# Define the objective function for optimization
def objective_function(params, df):
    alpha, beta, sigma = params
    total_error = 0
    count = 0  # To keep track of valid data points
```

```python
# Using minimize with L-BFGS-B method for optimization
result = minimize(objective_function, initial_guess, args=(df,), method='L-BFGS-B', bounds = [(0.001, None), (0.001, None), (0.001, None)])

optimal_alpha, optimal_beta, optimal_sigma = result.x
```

```python
for _, row in df.iterrows():
    D = row['Total Liabilities']
    Vb = row['Total Assets']
    r = row['r']
    T = row['T']
    real_equity_value = row['E']  # Assuming this is the real equity value

    if D <= 0 or Vb <= 0:
        continue  # Skip invalid debt or book value

    model = Model(D, T, r, 0, Vb)  # Assuming t is 0 as we don't have a separate column for it
    calculated_equity_value = model.calculate_equity_value(alpha, beta, sigma)
    total_error += (real_equity_value - calculated_equity_value) ** 2
    count += 1

return total_error / max(count, 1)  # Normalize by count to avoid division by zero
```

DD

DD value was calculated for three time horizons, which are 1, 5, and 10 years. Thus, we created loop for every time horizon.

It uses the asset_value for every step which were retrieved using optimized valus of beta, alpha, and sigma.

The function creates the results of data frame of 24 values for each time horizon.

```python
def calculate_asset_and_dd(row, alpha, beta, sigma, T_horizon):
    D = row['Total Liabilities']
    Vb = row['Total Assets']
    r = row['r']
    t = 0  # Assuming t is 0 as we don't have a separate column for it

    if sigma <= 0 or T_horizon - t <= 0:
        return np.nan  # Return NaN if sigma or T - t is non-positive

    model = Model(D, T_horizon, r, t, Vb)
    asset_value = model.asset_value(alpha, beta)

    if D <= 0 or asset_value <= 0:
        return np.nan  # Return NaN if D or asset_value is non-positive

    DD = (np.log(asset_value / D) + (r + 0.5 * sigma ** 2) * (T_horizon - t)) / (sigma * np.sqrt(T_horizon - t))
    return DD
```

CONVIVO

```
Optimal alpha: 1.415767
Optimal beta: 0.079179
Optimal sigma: 0.017935
Objective function value: 0.037263082228493796
```

	Total Liabilities	Total Assets	E	DD_1_year	DD_5_years
0	0.723679	0.741736	1.000000	26.218208	14.239714
1	0.800216	0.845949	0.445304	27.342673	14.534129
2	0.806033	0.925528	0.525369	31.839291	16.877219
3	0.945413	1.000000	0.474176	27.007446	14.686429
4	0.955017	0.964409	0.543055	24.465225	13.429823
5	1.000000	0.943046	0.695138	20.644936	11.588681
6	0.984460	0.906711	0.173159	19.577830	11.335876
7	0.949291	0.885364	0.535582	20.378080	11.733656
8	0.776958	0.793902	0.512491	25.787836	14.044253
9	0.761655	0.744505	0.554941	23.470413	12.851276
10	0.727488	0.702435	0.626909	23.174142	12.996062
11	0.650720	0.640615	0.524890	24.604040	13.551748
12	0.518942	0.522448	0.629799	26.783780	14.397890
13	0.515194	0.481343	0.730648	23.048287	12.675462
14	0.443641	0.371146	0.443330	18.650260	10.822310
15	0.521771	0.365614	0.150523	8.917708	6.539600
16	0.447052	0.322478	0.012078	11.527868	7.727845
17	0.477558	0.320016	0.114379	7.354479	5.631045
18	0.323034	0.234450	0.149544	14.752997	8.941757
19	0.335204	0.214283	0.100653	8.751770	6.390583
20	0.280029	0.208281	0.178333	17.452853	10.149169

	DD_10_years
0	12.291589
1	12.315518
2	14.265905
3	12.690335
4	11.695988
5	10.276852
6	10.296452
7	10.612988
8	12.150733
9	11.168759
10	11.516223
11	11.835099
12	12.319685
13	11.055900
14	9.846027
15	6.879406
16	7.738135
17	6.051824
18	8.394614
19	6.707915
20	9.248384
21	18.479250
22	48.393423
23	18.479250

EON

```
Optimal alpha: 1.415767
Optimal beta: 0.079179
Optimal sigma: 0.017935
Objective function value: 0.037263082228493796
```

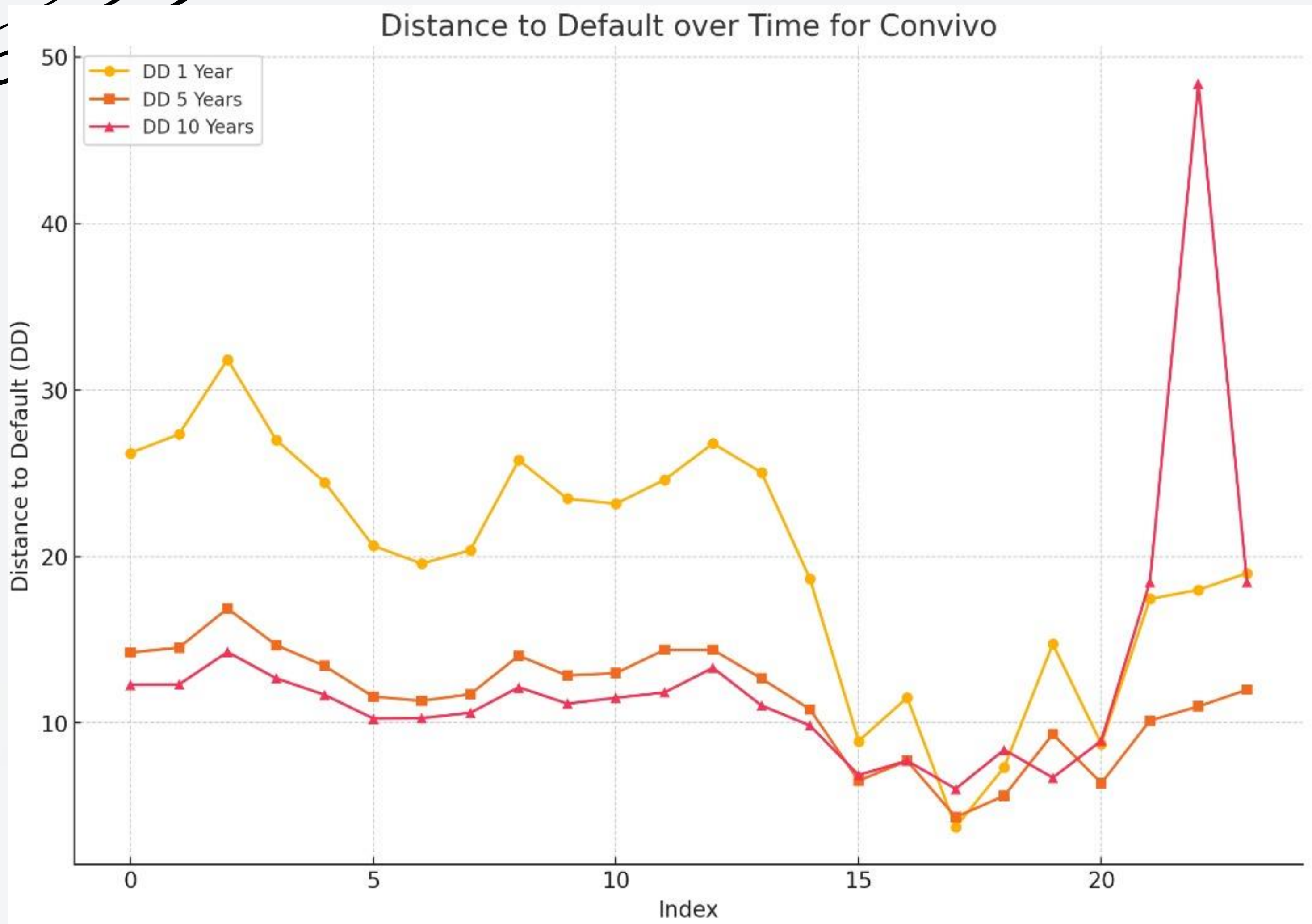
	Total Liabilities	Total Assets	E	DD_1_year	DD_5_years
0	0.723679	0.741736	1.000000	26.218208	14.239714
1	0.800216	0.845949	0.445304	27.342673	14.534129
2	0.806033	0.925528	0.525369	31.839291	16.877219
3	0.945413	1.000000	0.474176	27.007446	14.686429
4	0.955017	0.964409	0.543055	24.465225	13.429823
5	1.000000	0.943046	0.695138	20.644936	11.588681
6	0.984460	0.906711	0.173159	19.577830	11.335876
7	0.949291	0.885364	0.535582	20.378080	11.733656
8	0.776958	0.793902	0.512491	25.787836	14.044253
9	0.761655	0.744505	0.554941	23.470413	12.851276
10	0.727488	0.702435	0.626909	23.174142	12.996062
11	0.650720	0.640615	0.524890	24.604040	13.551748
12	0.518942	0.522448	0.629799	26.783780	14.397890
13	0.515194	0.481343	0.730648	23.048287	12.675462
14	0.443641	0.371146	0.443330	18.650260	10.822310
15	0.521771	0.365614	0.150523	8.917708	6.539600
16	0.447052	0.322478	0.012078	11.527868	7.727845
17	0.477558	0.320016	0.114379	7.354479	5.631045
18	0.323034	0.234450	0.149544	14.752997	8.941757
19	0.335204	0.214283	0.100653	8.751770	6.390583
20	0.280029	0.208281	0.178333	17.452853	10.149169

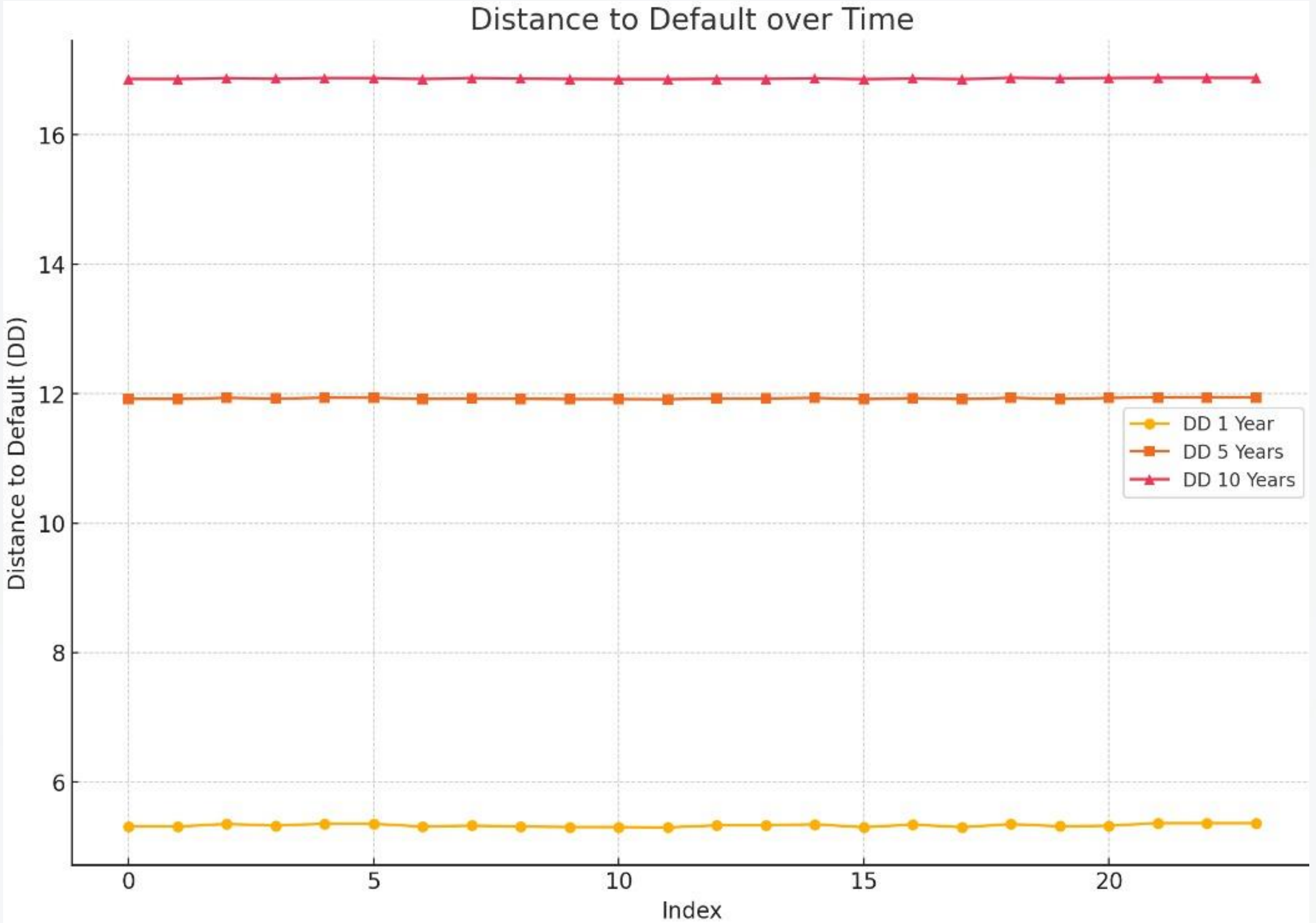
	DD_10_years
0	14.619421
1	14.623761
2	14.619740
3	14.620762
4	14.622386
5	14.630160
6	14.628663
7	14.625199
8	14.624651
9	14.629879
10	14.629379
11	14.629807
12	14.636075
13	14.689116
14	14.629807
15	14.710572
16	14.652394
17	14.632611
18	14.636907
19	14.651037
20	14.641384
21	14.641331
22	14.638799
23	14.638986

HERA

Optimal alpha: 0.365812
Optimal beta: 0.372403
Optimal sigma: 10.663355
Objective function value: 0.08488127025108479

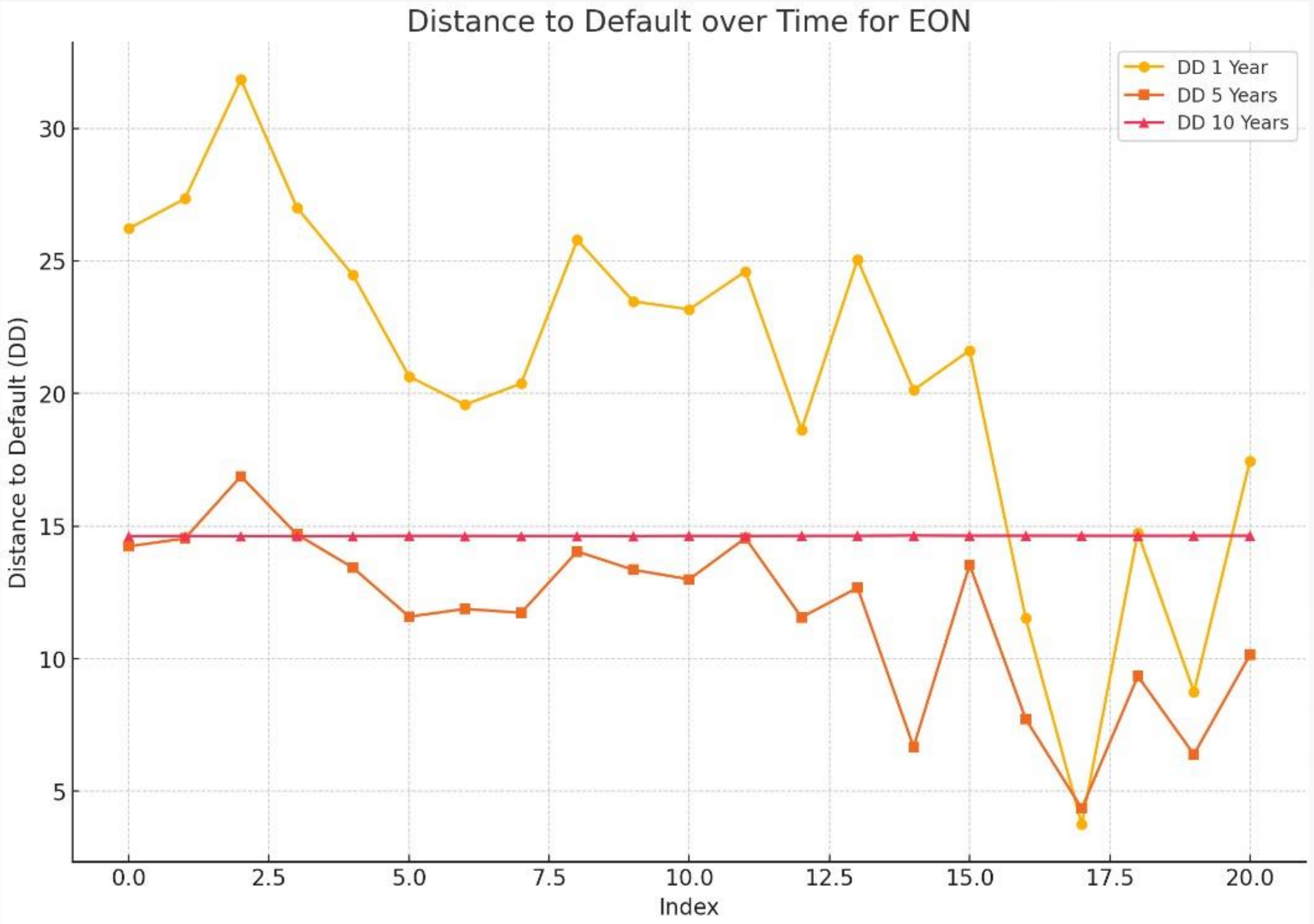
	Total Liabilities	Total Assets	E	DD_1_year	DD_5_years	DD_10_years
0	0.557689	0.237037	1.000000	5.315626	11.918730	16.861398
1	0.655445	0.426347	0.880108	5.313666	11.917874	16.860810
2	0.325233	0.081090	0.751987	5.353753	11.935778	16.873449
3	0.506316	0.271518	0.764729	5.327253	11.923970	16.865139
4	0.316011	0.092944	0.886735	5.357458	11.937438	16.874626
5	0.359194	0.216388	0.827039	5.355319	11.936466	16.873926
6	0.316011	0.092944	0.886735	5.357458	11.937438	16.874626
7	0.665293	0.691125	0.543345	5.328005	11.924203	16.865211
8	0.692557	0.504812	0.790725	5.313522	11.917917	16.860935
9	1.000000	1.000000	0.624094	5.305532	11.914447	16.858574
10	0.689967	0.331940	0.745575	5.302716	11.913341	16.857927
11	0.847815	0.592388	0.526562	5.299996	11.912225	16.857227
12	0.428967	0.135551	0.538186	5.332551	11.926697	16.867383
13	0.557301	0.314124	0.402670	5.321442	11.921618	16.863693
14	0.359346	0.098076	0.314296	5.345964	11.932521	16.871347
15	0.513154	0.304554	0.247798	5.328481	11.924722	16.865850
16	0.377256	0.141977	0.313852	5.345079	11.932229	16.871232
17	0.621413	0.432659	0.236214	5.319323	11.920848	16.863307
18	0.320754	0.093306	0.153471	5.356340	11.937383	16.874980
19	0.347492	0.138169	0.094434	5.352571	11.935747	16.873868
20	0.421367	0.182316	0.194593	5.337924	11.929046	16.868996
21	0.451454	0.220649	0.117050	5.334369	11.927394	16.867773
22	0.340467	0.115488	0.073193	5.352547	11.935591	16.873629
23	0.290444	0.105726	0.000000	5.366642	11.941903	16.878099





"In total, five Convivo companies have filed for insolvency: 'The group of companies is aware of its responsibility towards people and will conscientiously accompany the next steps," Convivo

https://www.refire-online.com/companies/as-curata-and-convivo-file-for-partial-bankruptcy-has-health/



06) LINEAR REGRESSION

To examine relation between physical risk, transition risk and firm‘s distance to default:

$$DD_t = \alpha + \beta_1 X_{1t} + \beta_2 X_{2t} + \beta_3 X_{3t} + \beta_4 X_{4t} + \beta_5 X_{5t} + \epsilon_t$$

- X_{1t}: Greenium

- X_{2t}: Average country drought
- X_{3t}: Average country flood

- X_{4t}: Debt Ratio = total liabilities/ total assets
- X_{5t}: Operating margin = operating income/ sales

07) GREENIUM & PHYSICAL RISK

GREENIUM:

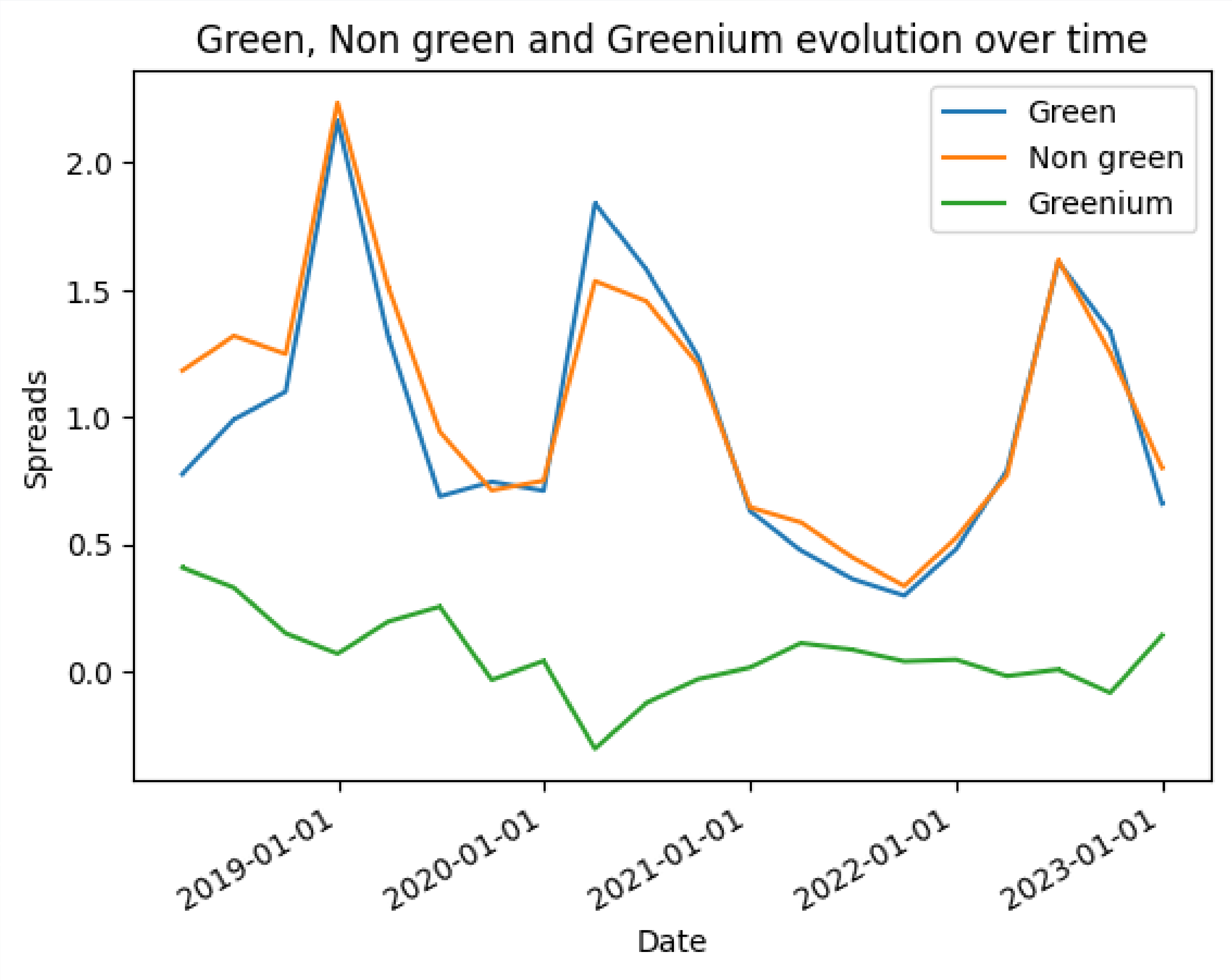
What premium investors are willing to pay for green bonds over comparable conventional bonds.

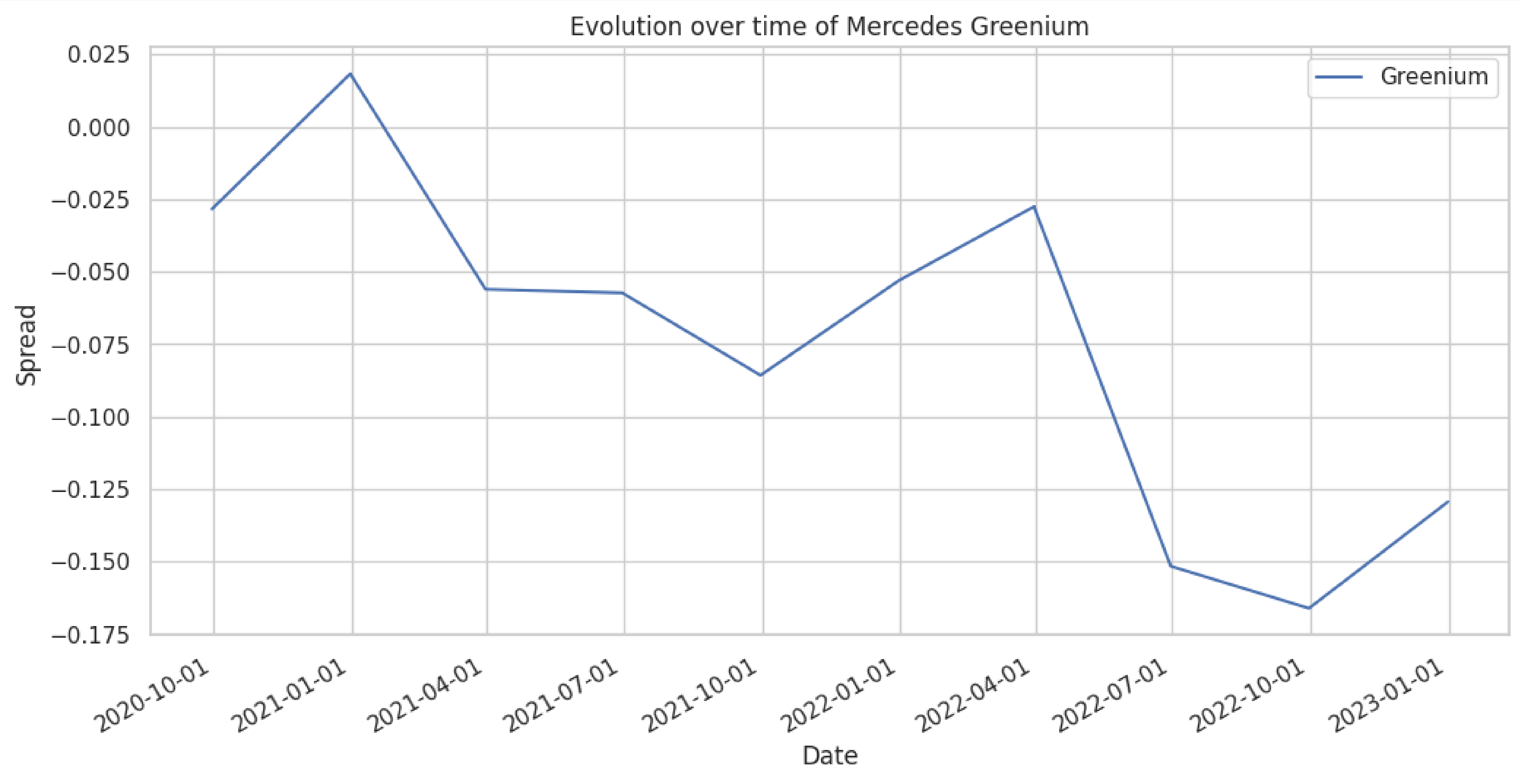
Greenium is driver for the transition risk.

Physical Risks:

Risks arising from the physical impacts of climate change.

- **Acute physical risk:** event-driven risks
  - → Extreme weather events, e.g. floods, wildfire
  - → Natural disaster, e.g. droughts, hurricanes
- **Chronic physical risk:** long-term shift in climate patterns that can have sustained impacts
  - → Rising temperature
  - → Sea-level rise
  - → Changes in precipitations patterns


Green, Non green and Greenium evolution over time
Green
Non green
Greenium
Spreads
2.0
1.5
1.0
0.5
0.0
2019-01-01
2020-01-01
2021-01-01
2022-01-01
2023-01-01
Date


Evolution over time of Mercedes Greenium
Greenium
Spread
Date
0.025
0.000
−0.025
−0.050
−0.075
−0.100
−0.125
−0.150
−0.175
2020-10-01
2021-01-01
2021-04-01
2021-07-01
2021-10-01
2022-01-01
2022-04-01
2022-07-01
2022-10-01
2023-01-01

08) CONVIVO REGRESSION: 1 YEAR DD

		coef	std err	t	P>|t|	[0.025	0.975]
1	const	-268.8754	59.858	-4.492	0.0	-394.632	-143.119
2	x1	-17.0795	12.728	-1.342	0.196	-43.82	9.661
3	x2	468.0143	72.959	6.415	0.0	314.734	621.295
4	x3	0.724	0.441	1.641	0.118	-0.203	1.651
5	x4	-8.9237	8.136	-1.097	0.287	-26.018	8.17
6	x5	-6.479	4.16	-1.557	0.137	-15.219	2.261
7	R^2	0.74					
8	F statistic	14.3					

08) CONVIVO REGRESSION: 5 YEAR DD

		coef	std err	t	P>|t|	[0.025	0.975]
1	const	-106.7911	105.209	-1.015	0.324	-327.827	114.245
2	x1	-1.5238	22.371	-0.068	0.946	-48.524	45.476
3	x2	128.8839	128.236	1.005	0.328	-140.53	398.298
4	x3	0.9915	0.776	1.278	0.217	-0.638	2.621
5	x4	-11.6864	14.301	-0.817	0.425	-41.732	18.359
6	x5	-6.0724	7.312	-0.83	0.417	-21.435	9.29
7							
8	R^2	0.77					
9	F statistic	15.0					

08) CONVIVO REGRESSION: 10 YEAR DD

		coef	std err	t	P>|t|	[0.025	0.975]
1	const	-73.1003	74.52	-0.981	0.34	-229.661	83.461
2	x1	-1.2429	15.846	-0.078	0.938	-34.533	32.048
3	x2	90.6259	90.83	0.998	0.332	-100.201	281.453
4	x3	0.7031	0.549	1.28	0.217	-0.451	1.857
5	x4	-8.3386	10.129	-0.823	0.421	-29.62	12.943
6	x5	-4.2912	5.179	-0.829	0.418	-15.173	6.59
7							
8	R^2	0.78					
9	F statistic	10.0					

08) EON REGRESSION: 1, 5, 10 YEAR DD

		coef	P>|t|
1	const	3.003	0.0
2	x1	-0.0334	0.776
3	x2	1.9482	0.0
4	x3	0.0027	0.392
5	x4	3.811e-07	0.984
6	x5	-0.0097	0.715
7			
8	R^2	0.6	
9	F statistic	5.0	

		coef	P>|t|
1	const	9.6113	0.0
2	x1	-0.0152	0.775
3	x2	0.8785	0.0
4	x3	0.0012	0.402
5	x4	1.668e-06	0.846
6	x5	-0.0043	0.723
7			
8	R^2	0.59	
9	F statistic	5.1	

		coef	P>|t|
1	const	14.1196	0.0
2	x1	-0.0119	0.74
3	x2	0.6109	0.0
4	x3	0.0009	0.359
5	x4	1.835e-08	0.997
6	x5	-0.004	0.625
7			
8	R^2	0.62	
9	F statistic	5.4	

08) HERA REGRESSION: 1, 5, 10 YEARS DD

		coef	P>|t|
1	const	5.6806	0.0
2	x1	-0.0086	0.129
3	x2	-0.4051	0.065
4	x3	-0.009	0.013
5	x4	-0.0006	0.876
6	x5	-0.0002	0.979
7			
8	R^2	0.48	
9	F statistic	3.3	

		coef	P>|t|
1	const	12.0987	0.0
2	x1	-0.0034	0.189
3	x2	-0.2018	0.05
4	x3	-0.0042	0.014
5	x4	-0.0004	0.786
6	x5	-0.0005	0.875
7			
8	R^2	0.42	
9	F statistic	2.7	

		coef	P>|t|
1	const	16.9961	0.0
2	x1	-0.0023	0.205
3	x2	-0.153	0.035
4	x3	-0.0028	0.018
5	x4	-0.0005	0.675
6	x5	-0.0005	0.843
7			
8	R^2	0.4	
9	F statistic	2.4	

08) Comparison of Results

- X_{1t}: Greenium
- X_{2t}: Average country drought
- X_{3t}: Average country flood
- X_{4t}: Debt Ratio = total liabilities/ total assets
- X_{5t}: Operating margin = operating income/ sales

		x1	x2	x3	x4	x5
CONVIVO (FR)	I YR	-17,0795	468,0143***	0,724	-8,9237	-6,479
	5 YR	-1,5238	128,8839	0,9915	-11,6864	-6,0724
	10 YR	-1,2429	90,6259	0,7031	-8,3386	-4,2912

		x1	x2	x3	x4	x5
EON (DE)	I YR	-0,0334	1,9482***	0,0027	0	-0,0097
	5 YR	-0,0152	0,8785***	0,0012	0	-0,0043
	10 YR	-0,0119	0,6109***	0,0009	0	-0,004

		x1	x2	x3	x4	x5
HERA (IT)	I YR	-0,0086	-0,4051*	-0,009*	-0,0006	-0,0002
	5 YR	-0,0034	-0,2018**	-0,0042**	-0,0004	-0,0005
	10 YR	-0,0023	-0,153**	-0,0028**	-0,0005	-0,0005

Greenium: price of brown bonds - price of green bonds
Negative coefficient: greenium lower – price of green bonds higher - yield on green lower - safer - longer distant to default
Positive cycle: Good reputation and management ability -- Higher demand -- lower yield -- lower financing cost for companies –less risky

THANKS FOR WATCHING