CAPM

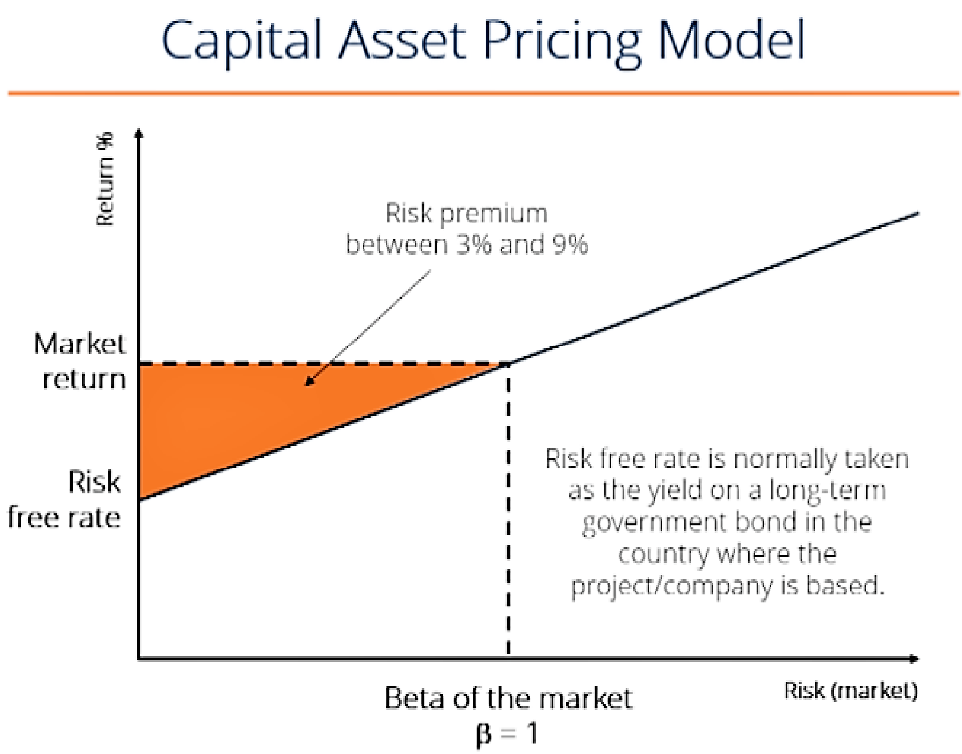
Introduction:

The CAPM was introduced in the early 1960s by William Sharpe, Jack Traynor, John Lintner, and Jan Mossin on diversification and pricing of risky assets. While investing in any asset one always looks into the rate of return and risk associated with it. Here CAPM plays its role taking account of systematic risk (Beta β), expected return of the market, and expected return of risk-free-asset. Further we will deep dive into the assumptions, various terminologies which are needed to understand the CAPM, implications and finally we will code the model.

Lets get started!

Assumptions:

- [ ] Under the risk free rate of interest we can lend and borrow unlimited amounts.
- [ ] Over a single period planning horizon individuals seek to maximise the expected utility of their portfolio.
- [ ] All information is available to all investors at the same time.
- [ ] The market is perfect which means no taxes, zero transaction cost, securities are completely divisible and the market is competitive.



$$E(ri) = Rf + \beta(Rm - Rf)$$

Where,

E(ri) = Expected rate of return on security
Rf = Risk free rate
β = Beta of the security
Rm = Expected returns of the market

Systematic Risk:

It is a risk applicable to all the sectors meaning inherent risk applicable to all the market. We cannot control this risk.

Risk free rate(Rf):

It is the return on investment with zero risk involved. It is typically corresponds to the yield on the 10-year US government bond.

Market Risk Premium: β(Rm - Rf):

Investors need to be compensated for the systematic risk in the form of risk premium.

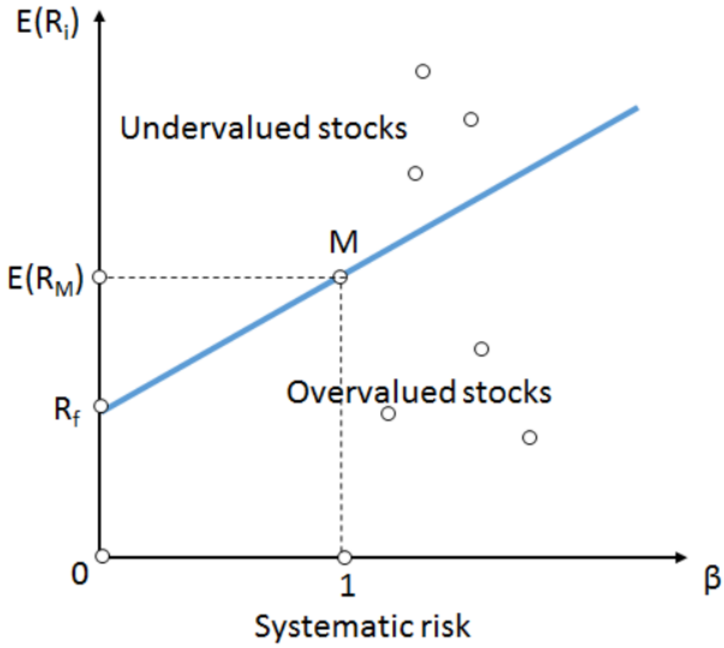
Beta(β):

It is the measurement of volatility or systematic risk. The calculation of beta helps us how the security moves with the market providing how risky is the security/asset with respect to the market. It is calculated by dividing the covariance of excess market returns and excess security returns by the variance of excess market return over risk free return. Statistically it can be calculated by regressing the data points

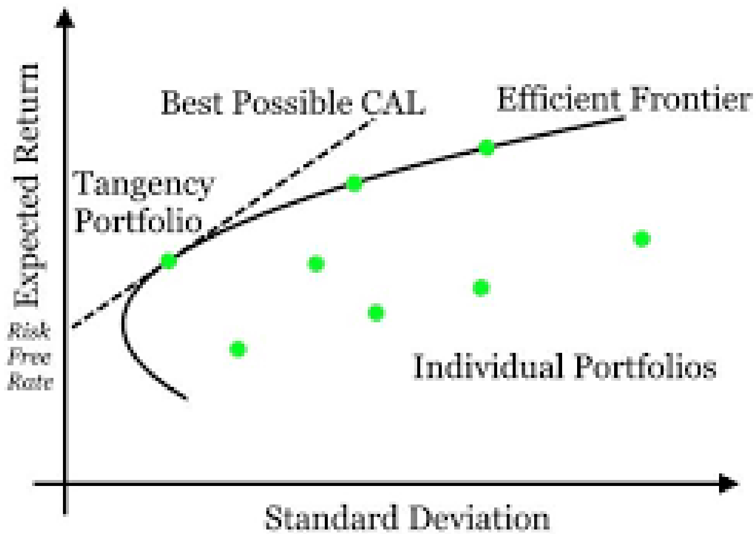
$$\beta = \frac{\text{Covariance } (R_e, R_m)}{\text{Variance } (R_m)}$$

Security Market line(SML):

SML is the graphical visualisation of capital asset pricing model. On the vertical axis it represents market return while on horizontal axis it represents beta. It helps to determine the investment is favourable or not. When the security appears above the SML then it is considered as undervalued and if the security is plotted below the SML it is considered overvalued.



Efficient Frontier and CAPM:



Efficient frontier comprises optimal portfolios that offer the highest expected returns for a specific level of risk. Here the risk is depicted by standard deviation. On the horizontal axis is plotted while on the vertical axis expected returns are plotted.

Capital market line (CML):

The line connecting risk-free return and tangency point on the efficient frontier is called the Capital market line(CML). CML is a linear combination of risk-free assets and portfolios. The tangency portfolio is the optimal portfolio providing maximum return per unit risk taken.

$$\hat{r}_p = \underset{\substack{\uparrow \\ \text{Intercept}}}{r_{RF}} + \left[\frac{\hat{r}_M - r_{RF}}{\underset{\substack{\uparrow \\ \text{Slope}}}{\sigma_M}} \right] \underset{\substack{\uparrow \\ \text{Risk} \\ \text{measure}}}{\sigma_p}.$$

Let’s try to code what we have learned so far!!

```
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
import pandas_datareader as pdr
import datetime as dt
import matplotlib.pyplot as plt
%matplotlib notebook
```

```
stocks = ['AAPL', '^GSPC']
start = dt.datetime(2015, 12, 1)
end = dt.datetime(2022, 1, 1)

data = pdr.get_data_yahoo(stocks, start, end)
data = data['Adj Close']
```

```
data.head()
```

Symbols	AAPL	^GSPC
Date		
2015-11-30	27.232513	2080.409912
2015-12-01	27.011522	2102.629883
2015-12-02	26.767509	2079.510010
2015-12-03	26.518894	2049.620117
2015-12-04	27.400557	2091.689941

```
log_return = np.log(data/data.shift())
log_return.head()
```

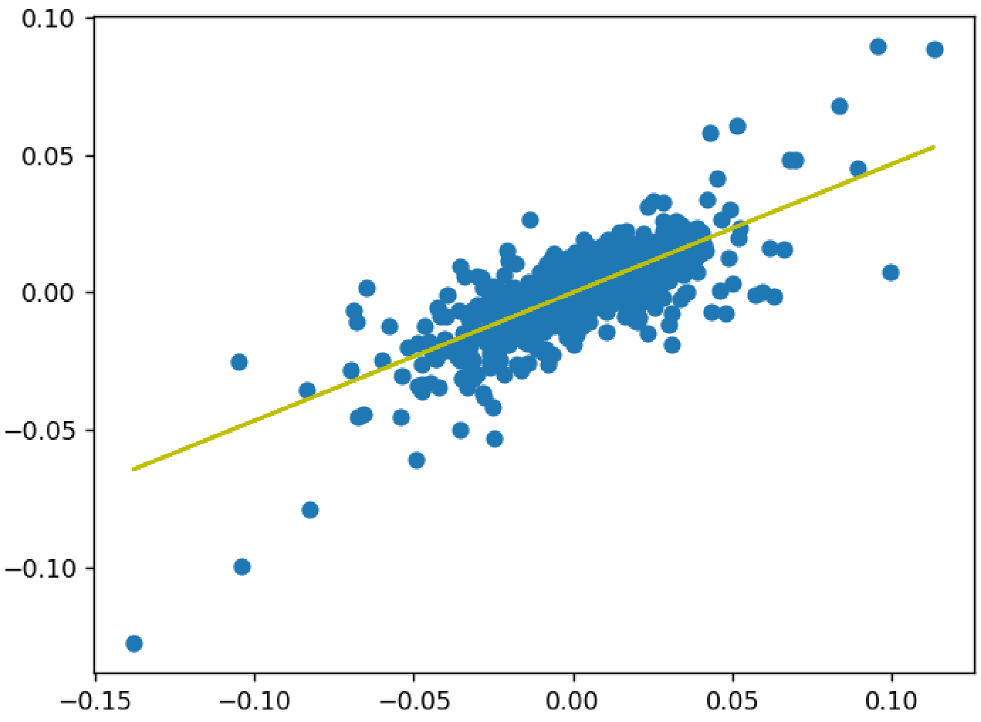
Symbols	AAPL	^GSPC
Date		
2015-11-30	NaN	NaN
2015-12-01	-0.008148	0.010624
2015-12-02	-0.009075	-0.011057
2015-12-03	-0.009331	-0.014478
2015-12-04	0.032706	0.020318

```
stock_1 = 'AAPL'
stock_2 = '^GSPC'
X = log_return[stock_1].iloc[1:].to_numpy().reshape(-1, 1)
Y = log_return[stock_2].iloc[1:].to_numpy().reshape(-1, 1)

lin_reg = LinearRegression()
lin_reg.fit(X, Y)

beta = lin_reg.coef_[0, 0]
Y_predict = lin_reg.predict(X)

fig, ax = plt.subplots()
ax.scatter(X, Y)
ax.plot(X, Y_predict, c = 'y')
```



```
beta
```

0.46751966279741536

```
# Risk free return = 0.056
risk_free_return = 0.052
#Market return = 0.103
market_return = 0.103

#As per the CAPM formula

expected_return = risk_free_return + beta*(market_return - risk_free_return)
```

```
expected_return
```

0.07584350280266818