The ARMAX Model

Full Name:

The **A**uto**r**egressive **M**oving **A**verage e**X**ogenous Model

Mathematical Notation:

$$P_t = c + \beta X + \varphi_1 P_{t-1} + \theta_1 \varepsilon_{t-1} + \varepsilon_t$$

P_t, P_{t-1}	Values in the current period and 1 period ago respectively
ε_t, ε_{t-1}	Error terms for the same two periods
c	Baseline constant factor
φ_1	What part of the value last period is relevant in explaining the current one
θ_1	What part of the error last period is relevant in explaining the current value
X	Exogenous variable
β	Coefficient for the exogenous variable

Description:

The ARMAX is an extension of the ARIMA model, which incorporates other **exogenous** variables.

These variables can be pretty much anything that can have an affect on the values we are trying to estimate. The only requirement is that we have data available for every time period we are interested in. Thus, we often rely on other time series as the exogenous components in the regression.

These models are great, when a big part of the change period to period cannot be explained by past values and past errors alone, so including other relevant values might be of great help (like the prices for an index of a market of a neighbouring country).

The ARMAX Model

Implementation of the Simple Model in Python:

The library the *ARMA* method comes from

The method we are importing

```
from statsmodels.tsa.arima_model import ARMA
```

```
model_ar_1_ma_1_X_spx = ARMA(df.returns[1:], exog = df.returns_spx[1:], order=(1,1))
```

The variable storing the model characteristics that we will fit later

The time series we wish to analyse

The exogenous variable we are adding to the ARMA model

The order of the model

*For an ARMAX(p,q) model, simply change the order from (1,1) to (p,q).