

# Describe Model

TIME SERIES ANALYSIS IN PYTHON



**Rob Reider**

Adjunct Professor, NYU-Courant  
Consultant, Quantopian

# Mathematical Description of MA(1) Model

$$R_t \text{ equals } \mu + \epsilon_t + \theta \epsilon_{t-1}$$

- Since only one lagged error on right hand side, this is called:
  - MA model of order 1, or
  - MA(1) model
- MA parameter is  $\theta$
- Stationary for all values of  $\theta$

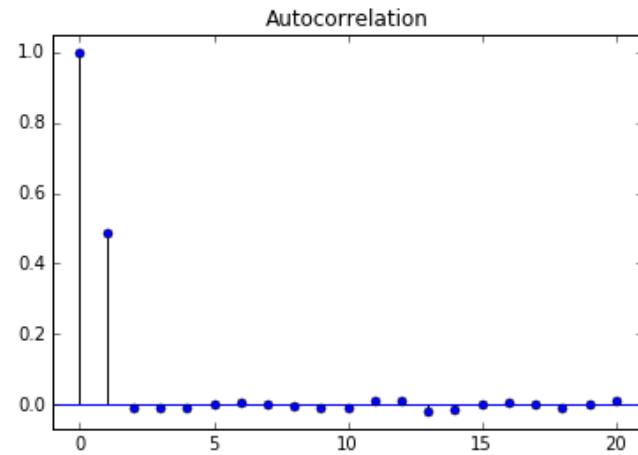
# Interpretation of MA(1) Parameter

$$R_t = \mu + \epsilon_t + \theta \epsilon_{t-1}$$

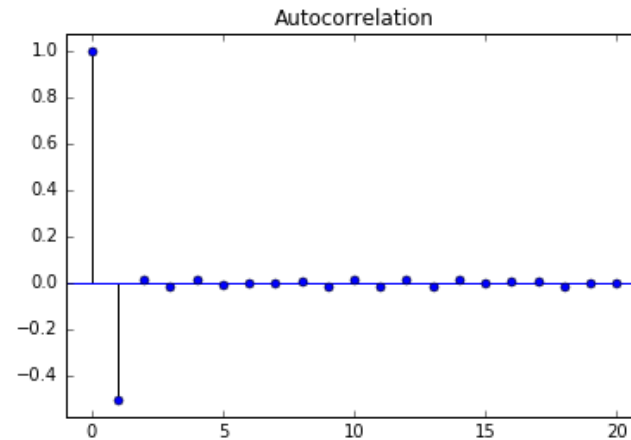
- Negative  $\theta$ : One-Period Mean Reversion
- Positive  $\theta$ : One-Period Momentum
- Note: One-period autocorrelation is  $\theta / (1 + \theta^2)$ , not  $\theta$

# Comparison of MA(1) Autocorrelation Functions

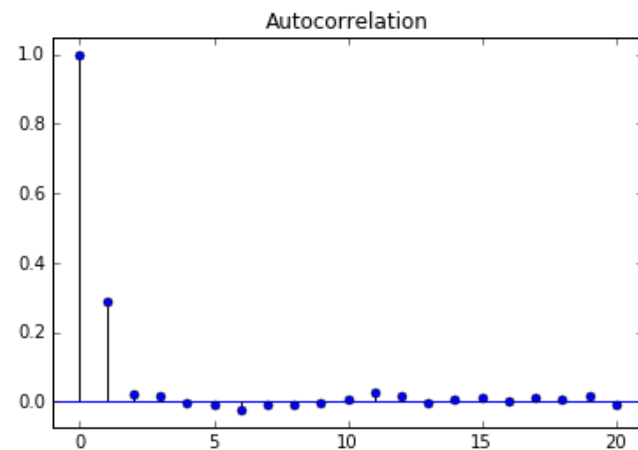
- $\theta = 0.9$



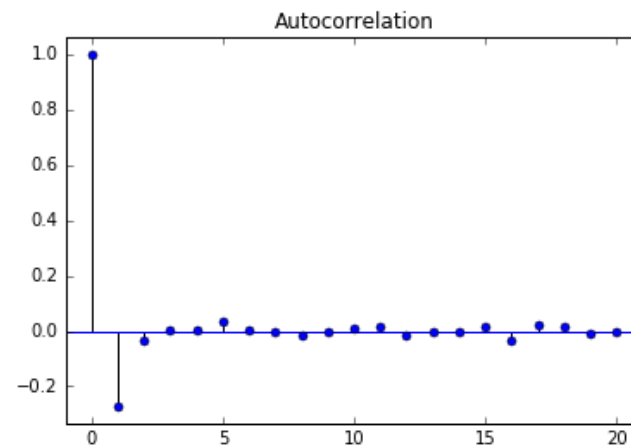
- $\theta = -0.9$



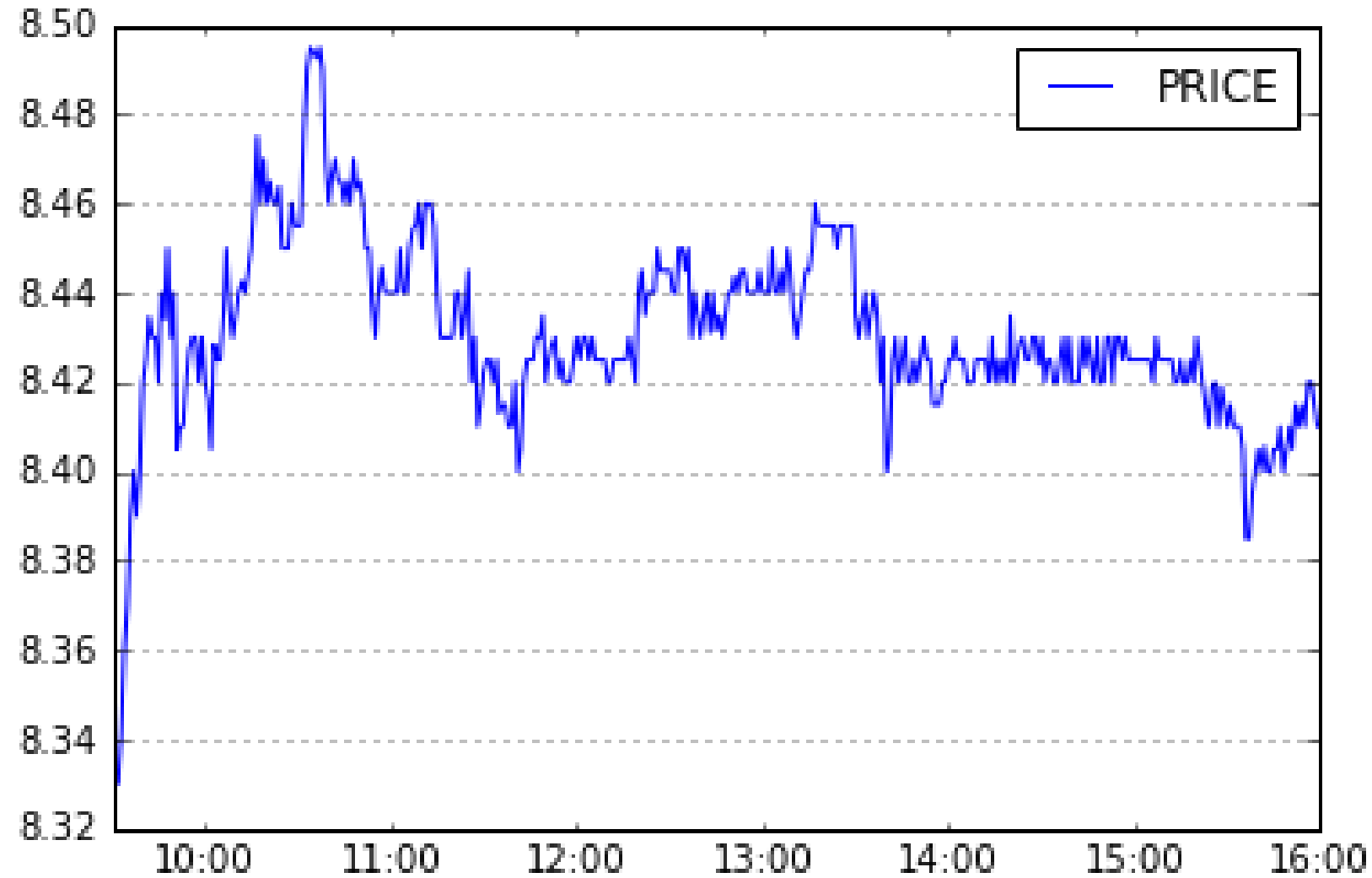
- $\theta = 0.5$



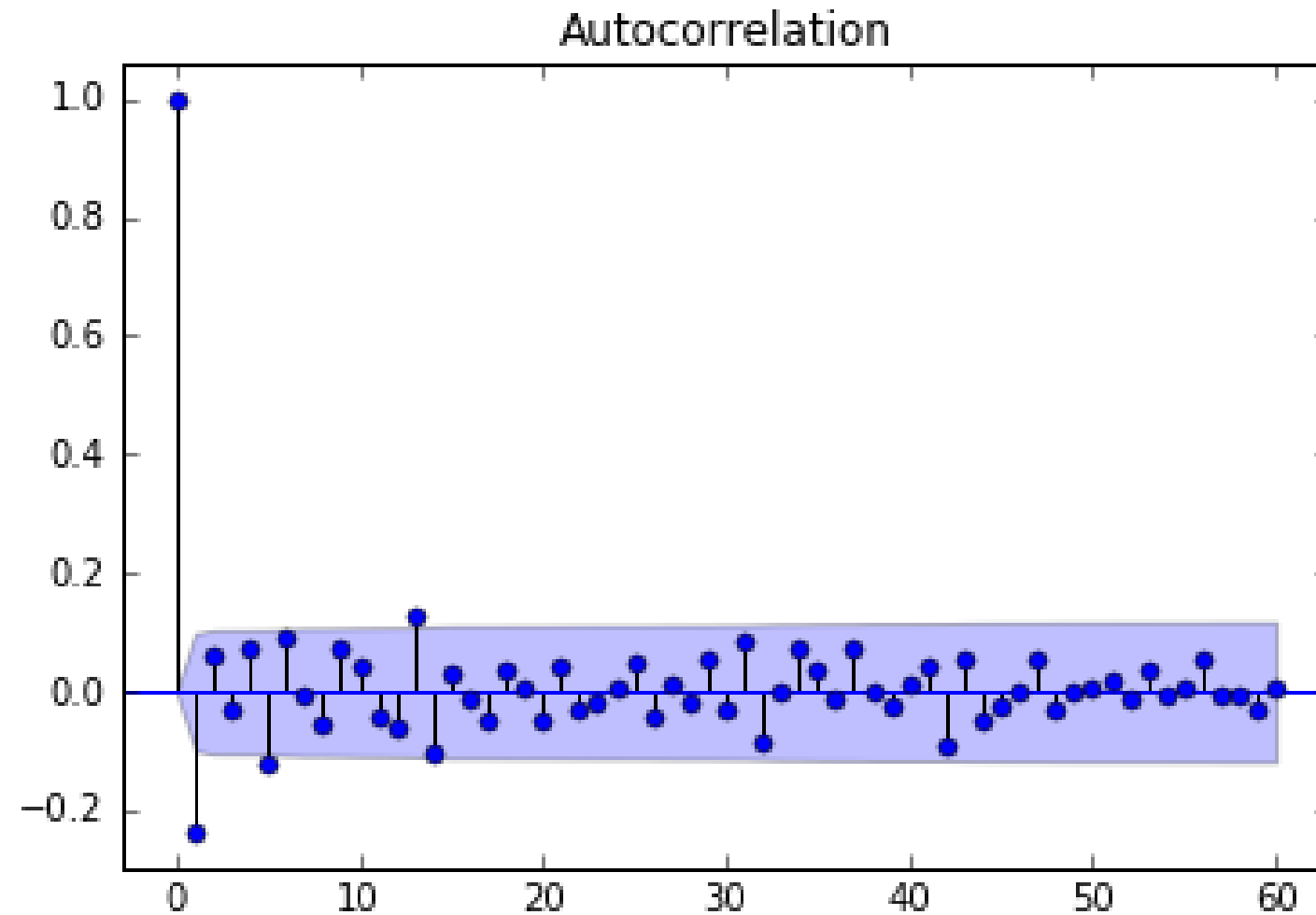
- $\theta = -0.5$



# Example of MA(1) Process: Intraday Stock Returns



# Autocorrelation Function of Intraday Stock Returns



# Higher Order MA Models

- MA(1)

$$R_t = \mu + \epsilon_t - \theta_1 \epsilon_{t-1}$$

- MA(2)

$$R_t = \mu + \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2}$$

- MA(3)

$$R_t = \mu + \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \theta_3 \epsilon_{t-3}$$

- ...

# Simulating an MA Process

```
from statsmodels.tsa.arima_process import ArmaProcess
ar = np.array([1])
ma = np.array([1, 0.5])
AR_object = ArmaProcess(ar, ma)
simulated_data = AR_object.generate_sample(nsample=1000)
plt.plot(simulated_data)
```

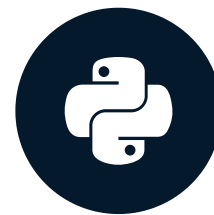


# Let's practice!

TIME SERIES ANALYSIS IN PYTHON

# Estimation and Forecasting an MA Model

TIME SERIES ANALYSIS IN PYTHON



**Rob Reider**

Adjunct Professor, NYU-Courant  
Consultant, Quantopian

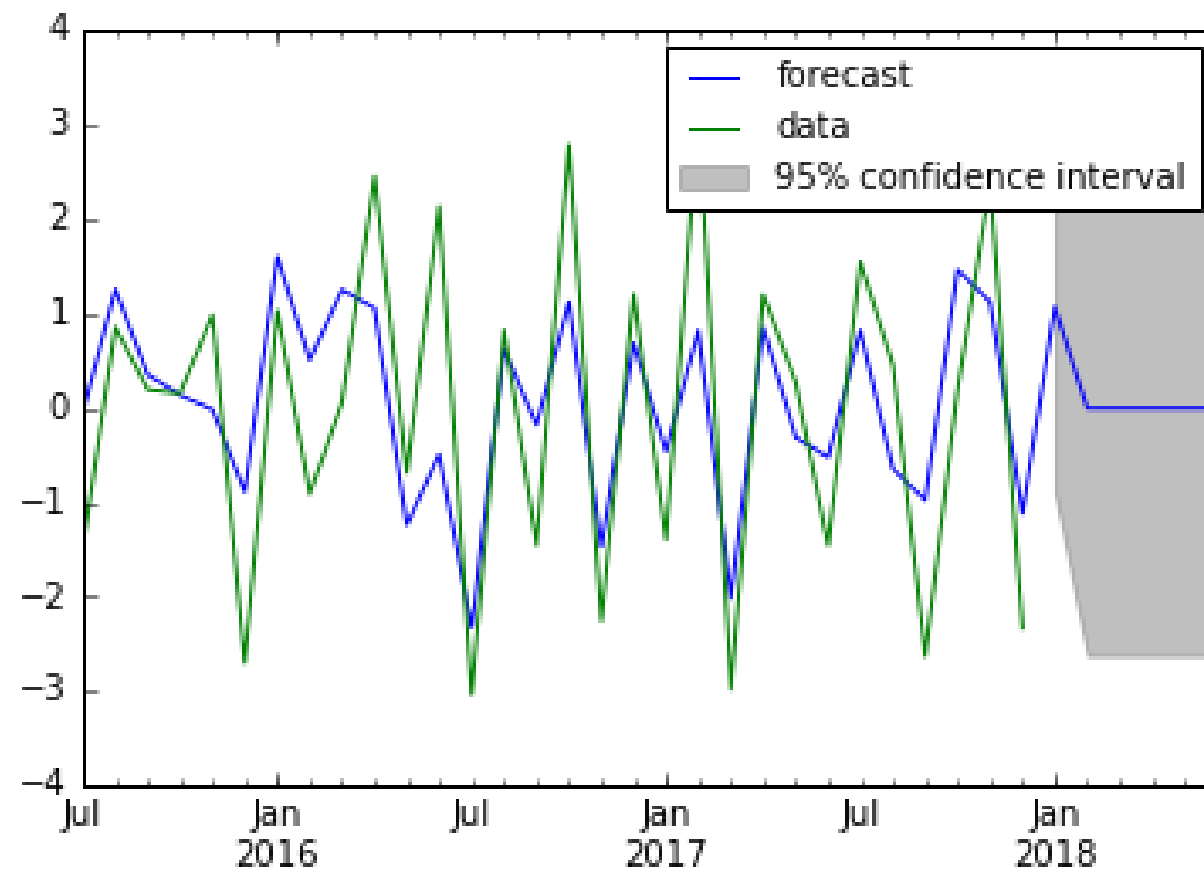
# Estimating an MA Model

- Same as estimating an AR model (except `order=(0,1)` )

```
from statsmodels.tsa.arima_model import ARMA
mod = ARMA(simulated_data, order=(0,1))
result = mod.fit()
```

# Forecasting an MA Model

```
from statsmodels.tsa.arima_model import ARMA
mod = ARMA(simulated_data, order=(0,1))
res = mod.fit()
res.plot_predict(start='2016-07-01', end='2017-06-01')
plt.show()
```



# Let's practice!

TIME SERIES ANALYSIS IN PYTHON

# ARMA models

TIME SERIES ANALYSIS IN PYTHON



**Rob Reider**

Adjunct Professor, NYU-Courant  
Consultant, Quantopian

# ARMA Model

- ARMA(1,1) model:

$$R_t = \mu + \phi R_{t-1} + \epsilon_t + \theta \epsilon_{t-1}$$

# Converting Between ARMA, AR, and MA Models

- Converting AR(1) into an MA(infinity)

$$R_t = \mu + \phi R_{t-1} + \epsilon_t$$

$$R_t = \mu + \phi(\mu + \phi R_{t-2} + \epsilon_{t-1}) + \epsilon_t$$

⋮

$$R_t = \frac{\mu}{1 - \phi} + \epsilon_t + \phi\epsilon_{t-1} - \phi^2\epsilon_{t-2} + \phi^3\epsilon_{t-3} + \dots$$



# Let's practice!

TIME SERIES ANALYSIS IN PYTHON