Autoregressive processes AR(p) : introduction

An autoregressive process AR(p) in time series analysis is a type of stochastic model where each data point is linearly dependent on its previous p data points. It is characterized by a regression of the current observation on its own past values.

The general formula for an Autoregressive process or order p - AR(p) process - is defined as follows:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + ... + \phi_p X_{t-p} + Z_t$$

where ϕ_i's are constants and $\{Z_t\}$ is some White Noise $N(0, \sigma_Z^2)$.

Mean

AR(p) could also be expressed differently. For instance, let us consider the AR(1) process

$$X_t = \phi X_{t-1} + Z_t$$

Using sucessive substitutions, we find that X_t can be expressed as an $MA(\infty)$ process. Indeed, we have

$$\begin{aligned} X_t &= \phi X_{t-1} + Z_t \\ &= \phi(\phi X_{t-2} + Z_{t-1}) + Z_t \\ &= \phi^2(\phi X_{t-3} + Z_{t-2}) + \phi Z_{t-1} + Z_t \\ &= \phi^3(\phi X_{t-4} + Z_{t-3}) + \phi^2 Z_{t-2} + \phi Z_{t-1} + Z_t \\ &= ... \\ &= Z_t + \phi Z_{t-1} + \phi^2 Z_{t-2} + \phi^3 Z_{t-3} + ... \end{aligned}$$

From here, it is easy to deduce that $E[X_t] = 0$.

Variance

We can also now deduce the formula for the variance of X_t, which is given by

$$\begin{aligned} var(X_t) &= var(Z_t + \phi Z_{t-1} + \phi^2 Z_{t-2} + \phi^3 Z_{t-3} + ...) \\ &= \sigma_Z^2 (1 + \phi^2 + \phi^4 + \phi^6 + ...) \\ &= \sigma_Z^2 \sum_{i=0}^{\infty} \phi^{2i} \end{aligned}$$

And the sum $\sum_{i=0}^{\infty} \phi^{2i}$ will converge to $\frac{1}{1-\phi^2}$ provided that $\mid \phi \mid < 1$, so that we find that the variance of X_t is

$$var(X_t) = \sigma_X^2 = \frac{\sigma_Z^2}{1 - \phi^2}$$

Confidence Interval for a AR(1) - 1/2

Suppose that in a sample of size 100 from an AR(1) process satisfying

$$X_t - \mu = \phi(X_{t-1} - \mu) + Z_t, Z_t \sim WN(0, \sigma^2)$$

with mean $\mu, \phi = 0.6$, $\sigma^2 = 2$ and $E[Z_t^4] < \infty$. Assume that we observe $\bar{x}_{100} = -0.1308$.

What is an approximate 95% confidence interval for μ. Are the data compatible with the hypothesis that $\mu = 0$?

Confidence Interval for a AR(1) - 2/2

The mean μ is approximately distributed as $N\left(\bar{x}_{100}, \frac{\sigma^2}{n(1-\phi)^2}\right)$. A 95% Confidence Interval is given by

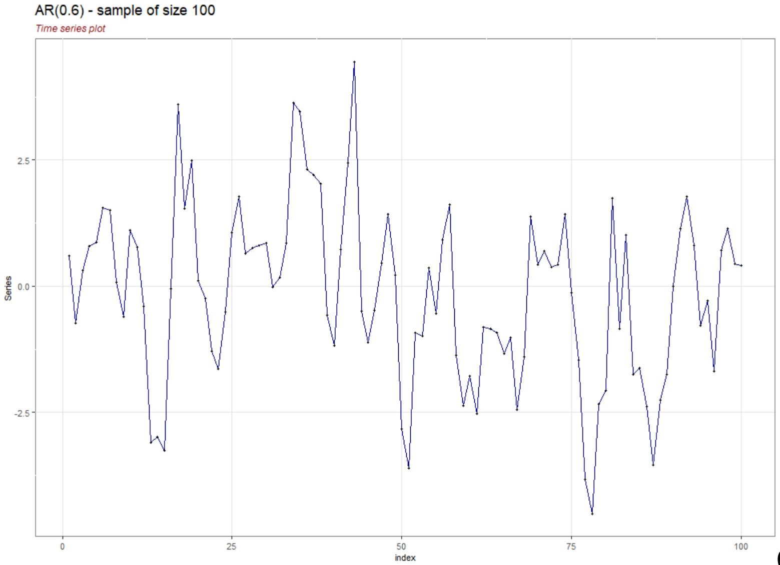
$$CI = [\bar{x}_{100} \pm z_{0.025} \frac{\sigma}{n(1-\phi)}]$$

where $z_{0.975} \approx 1.96$ is the 2.5% quantile of the Normal distribution. Pluging the numeric values, we get

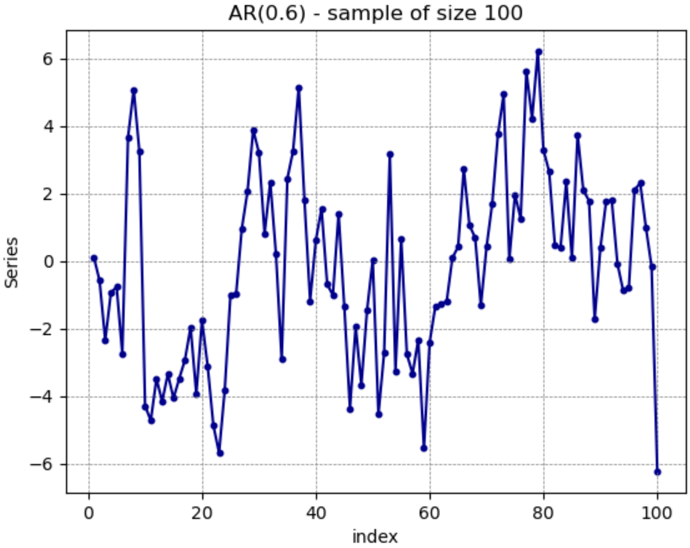
$$CI = -0.1308 \pm 0.693. = [-0.827, 0.562]$$

Since $0 \in CI$, the hypothesis that $\mu = 0$ cannot be rejected.
Note: in the examlple below (code and visualization), we use different time series with different means, hence different Confidence Intervals for R and Python. The conclusion remains valid.

Visualizing the data in R



Visualizing the data in Python



R code

```
# parameters
mu = phi = 0.6; n = 100; sigma = 2
# sample
set.seed(2023)
xt = arima.sim(list(order=c(1,0,0), ar = phi), sd = sqrt(sigma), n = n)
xt - mu
# sample mean
mean(xt - mu)
# [1] -0.1307798
# 95% Confidence Interval
CI = c(mean(xt - mu) - (qnorm(0.025) * (sqrt(2) / (sqrt(n)*(1-phi)))),
       mean(xt - mu) + (qnorm(0.025) * (sqrt(2)/ (sqrt(n)*(1-phi)))))
CI
# -0.8237318  0.5621721
# data reshaping
df = data.frame(series = xt-mu,
                index = seq(1,100))
# visualization
library(tidyverse)
ggplot(df, aes(x = index, y = series)) +
  geom_line(color = 'darkblue') +
  geom_point(size = 0.6) +
  labs(title = 'AR(0.6) - sample of size 100',
       subtitle = 'Time series plot',
       y="Series", x="index") +
       customization
```

Python code

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import norm
# parameters
mu = phi = 0.6; n = 100; sigma = 2
# sample
np.random.seed(2023)
xt = np.zeros(n)
xt[0] = np.random.normal()
for i in range(1, n):
    xt[i] = mu + phi * (xt[i - 1] - mu) + np.random.normal(scale=sigma)
xt_centered = xt - mu
# sample mean
np.mean(xt_centered)
# 95% Confidence Interval
CI = [sample_mean - ( norm.ppf(0.975)  * (np.sqrt(2) / (np.sqrt(n) * (1 - phi)))
      ),
       sample_mean + (norm.ppf(0.975) * (np.sqrt(2) / (np.sqrt(n) * (1 - phi))))]
CI
# [-0.8796616608140211, 0.5062421635356569]
# data reshaping
df = pd.DataFrame({'series': xt_centered, 'index': range(1, n + 1)})
# Visualization
plt.plot(df['index'], df['series'], color='darkblue')
plt.scatter(df['index'], df['series'], s=10, color='darkblue')
plt.title('AR(0.6) - sample of size 100')
plt.xlabel('index'); plt.ylabel('Series')
plt.grid(True, color='grey', linestyle='--', linewidth=0.5)
plt.show()
```

References

Shumway, R.H., Stoffer, D.S. , (2011). Time Series Analysis and its Applications, Third Edition. Springer.

Brockwell, P.J., Davis, R.A, (2002). Introduction to Time Series and Forecasting, Second Edition. Springer.

The R Project for Statistical Computing:
https://www.r-project.org/

Python:
https://www.python.org/