ex2_e

March 21, 2025

[1]:
```python
from scipy.stats import norm
import numpy as np
from statsmodels.tsa.stattools import acovf

def simulate_ar1(rho, sigma, T, reps, alpha=0.05, random_seed=1234):
    """
    Simulates an AR(1) process y_t = rho*y_{t-1} + eps_t (eps ~ N(0,sigma^2)).
    Tests H0: E[y_t] = 0 against H1: E[y_t] != 0.
    Returns the empirical rejection frequency.
    """
    np.random.seed(random_seed)
    reject_count = 0
    critical_value_2_sides = norm.ppf(1 - alpha/2)

    for _ in range(reps):
        # Generate data
        e = np.random.normal(0, sigma, T)
        y = np.zeros(T)
        for t in range(1, T):
            y[t] = rho * y[t-1] + e[t]

        # y = np.random.normal(0, sigma, T) # check if for normal distribution␣
↪rejection is 5%

        sample_mean = np.mean(y)
        sample_var = np.var(y)

        # Compute t-statistic
        t_stat = sample_mean / np.sqrt(sample_var/T)

        # Check if t-statistic is greater than critical value
        if np.abs(t_stat) > critical_value_2_sides:
            reject_count += 1

    return reject_count / reps
```

```python
# Parameters
rho = 0.8       # True phi under H0
sigma = 0.1     # Std dev of eps
T = 50          # Sample size
reps = 10000    # Number of Monte Carlo replications

for seed in range(10):
    empirical_size = simulate_ar1(rho, sigma, T, reps, random_seed=37*seed**2)
    print(f"Empirical rejection frequency at nominal 5% = {empirical_size*100:.
 ↪2f}%")
```

```
Empirical rejection frequency at nominal 5% = 54.61%
Empirical rejection frequency at nominal 5% = 53.45%
Empirical rejection frequency at nominal 5% = 53.43%
Empirical rejection frequency at nominal 5% = 54.28%
Empirical rejection frequency at nominal 5% = 53.57%
Empirical rejection frequency at nominal 5% = 54.07%
Empirical rejection frequency at nominal 5% = 54.41%
Empirical rejection frequency at nominal 5% = 54.59%
Empirical rejection frequency at nominal 5% = 54.24%
Empirical rejection frequency at nominal 5% = 53.35%
```

[ ]: