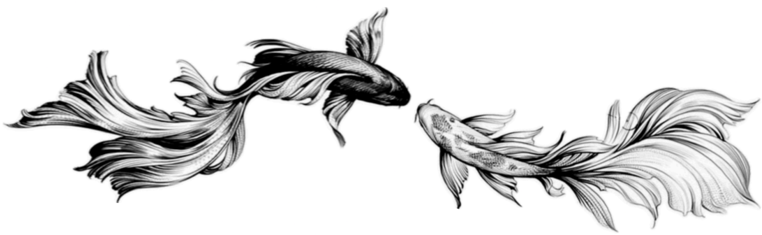
CKLS Model Parameter Estimation

National Higher School of Mathematics

Department of Statistics for Econometrics and Actuarial Sciences



May, 21th 2025

Mandi Souhil

CKLS Model Parameter Estimation

Contents

CKLS Model Parameter Estimation .... - 1 -
Generalized Method of Moments .... - 1 -
Parameter Estimation on Synthetic Data .... - 4 -
Parameter Estimation on Real Data .... - 5 -
Alternative Methods for Estimation .... - 9 -
Remark: .... - 9 -
Bibliography .... - 9 -

CKLS Model Parameter Estimation

> In this report, we estimate the CKLS Model [1] parameters. The model is defined as:

$$dr_t = (\alpha + \beta r_t)dt + \sigma r_t^\gamma dB_t$$

where r_t represents the interest rate we aim to predict, and α, β, σ, and γ are the parameters to be estimated.

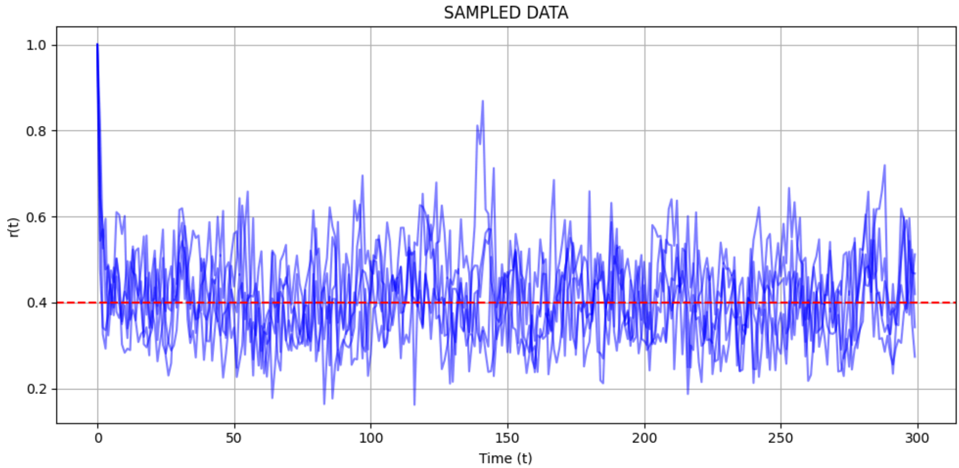


Figure 2: 20 paths simulated with parameters $(0.2, -0.5, 0.02, 0.6)$. These processes are *mean-reverting* (red line $y = -\frac{\alpha}{\beta}$ is the long-run mean)

In the original paper [1], the target parameters were estimated using the **Generalized Method of Moments** (abbreviated as **GMM**). We will reproduce the same method.

Generalized Method of Moments

Let $\{r_i\}_1^N$ be our dataset, in which we aim to estimate the parameter vector given by $\theta^* = (\alpha, \beta, \sigma, \gamma)$. Since the theoretical model is given by the *stochastic dynamics* of $\{r_t\}$, the distribution at each time t is unknown. Therefore, methods like *likelihood maximization* and *moment matching* are not directly applicable. The *GMM* method aims to provide an equivalent to the *moment matching* method by finding a function

$$m(\theta) = \mathbb{E}[g(r_t, \theta)]$$

where $m(\theta) = 0$ if and only if $\theta = \theta^*$, our target parameter. Hence, we need to find the function $(r_t, \theta) \mapsto g(r_t, \theta)$. In the original paper, this function is given as follows:

First, we provide a discretization of the stochastic dynamics of $\{r_t\}$, assuming a constant step size of one. We get:

$$r_{t+1} = r_t + \alpha + \beta r_t + \varepsilon_{t+1}$$
$$\mathbb{E}(\varepsilon_{t+1}) = 0 \quad \mathbb{E}(\varepsilon_{t+1}^2) = \sigma^2 r_t^{2\gamma}$$

From the above expression, we can derive four conditions (since we have four parameters to estimate) related to the model's residuals:

$$\begin{aligned}
\mathbb{E}[\varepsilon_{t+1}] &= 0 \quad \text{Expectation of diffusion term} \\
\mathbb{E}[\varepsilon_{t+1}.r_t] &= 0 \quad \text{First orthogonality condition} \\
\mathbb{E}\left[\varepsilon_{t+1}^2 - \sigma^2 r_t^{2\gamma}\right] &= 0 \quad \text{Variance of diffusion term} \\
\mathbb{E}\left[\left(\varepsilon_{t+1}^2 - \sigma^2 r_t^{2\gamma}\right).r_t\right] &= 0 \quad \text{Second orthogonality condition}
\end{aligned}$$

Hence, the function is:

$$g(r_t, \theta) = \begin{pmatrix} \varepsilon_{t+1} \\ \varepsilon_{t+1}.r_t \\ \varepsilon_{t+1}^2 - \sigma^2 r_t^{2\gamma} \\ \left(\varepsilon_{t+1}^2 - \sigma^2 r_t^{2\gamma}\right).r_t \end{pmatrix}$$

Next, we replace $g(r_t, \theta)$ with its empirical value $\frac{1}{N}\sum_{i=1}^{N} g(r_i, \theta)$. Finding the target parameter θ^* is equivalent to the following minimization problem:

$$\theta^* = \arg\min_{\theta \in \mathbb{R}^4} \left(\frac{1}{N}\sum_{i=1}^{N} g(r_i, \theta)\right)^T W \left(\frac{1}{N}\sum_{i=1}^{N} g(r_i, \theta)\right)$$

where W is a positive-definite weighting matrix. The optimal matrix W is:

$$W = \mathbb{E}\left[g(r_t, \theta^*) g(r_t, \theta^*)^T\right]$$

which cannot be obtained since the target parameter θ^* is unknown. But we can proceed in two steps:

Two-step Feasible GMM:

- Step 1: Take $W = I$ (the identity matrix) or another positive-definite matrix, and compute a preliminary GMM estimate θ_1.
- Step 2: Use the preliminary GMM estimate θ_1 to calculate:

$$\widehat{W}(\theta_1) = \left(\frac{1}{N} \sum_{i=1}^{N} g(r_i, \theta) g(r_i, \theta)^T \right)^{-1}$$

  Plug this matrix into the objective function and minimize it with respect to θ.

The following Python code implements this method:

```
# moments conditions of the the original paper :
def moment_conditions_original(theta, r):
    alpha, beta, sigma2, gamma = theta
    rt = r[:-1]
    rt1 = r[1:]
    rt_safe = np.where(rt > 1e-6, rt, 1e-6)
    eps = rt1 - rt - alpha - beta * rt
    m1 = np.mean(eps)
    m2 = np.mean(eps * rt)
    m3 = np.mean(eps**2 - sigma2 * rt_safe**(2 * gamma))
    m4 = np.mean((eps**2 - sigma2 * rt_safe**(2 * gamma)) * rt)
    return np.array([m1, m2, m3, m4])
def gmm_objective(theta, r, W):
    g = moment_conditions_original(theta, r)
    # print(g.shape)
    return g.T @ W @ g

def compute_optimal_weight_matrix(r, theta_hat):
    g_values = []
    T = len(r) - 1
    for t in range(T):
        rt = r[t]
        rt1 = r[t+1]
        rt_safe = max(rt, 1e-6)
        eps = rt1 - rt - theta_hat[0] - theta_hat[1] * rt
        sigma2 = theta_hat[2]
        gamma = theta_hat[3]
        moment = np.array([
            eps,
```

```
            eps * rt,
            eps**2 - sigma2 * rt_safe**(2 * gamma),
            (eps**2 - sigma2 * rt_safe**(2 * gamma)) * rt
        ])
        g_values.append(moment)
    g_matrix = np.vstack(g_values)
    return np.linalg.inv(np.cov(g_matrix, rowvar=False))
```

```
# Two-step GMM estimation
def estimate_two_step_gmm(r, theta_init):
    W1 = np.eye(4)
    result1 = minimize(gmm_objective, theta_init, args=(r, W1),
method='BFGS')
    theta1 = result1.x
    W_opt = compute_optimal_weight_matrix(r, theta1)
    result2 = minimize(gmm_objective, theta1, args=(r, W_opt),
method='BFGS')
    return result1, result2, W_opt
```

Parameter Estimation on Synthetic Data

Estimating parameters on synthetic data gives the following results:

$$
\begin{aligned}
\text{True parameters:} \quad & [0.2, \quad -0.5, 0.02, 0.6] \\
\text{Estimated parameters:} \quad & [0.22, -0.55, 0.02, 0.63]
\end{aligned}
$$

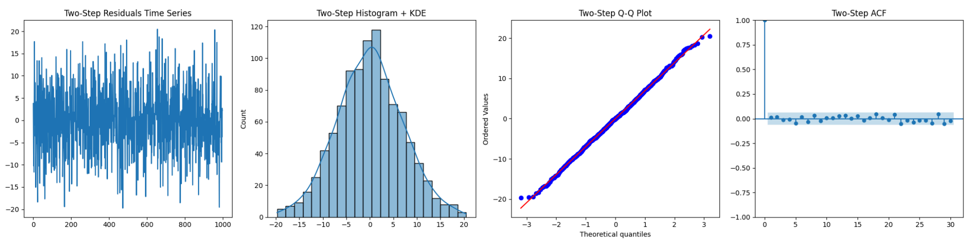


Figure 3: Two-step GMM residual statistics

Two-Step Residuals:

- Mean Zero Test: t = −0.048, p = 0.961 ⟶ Mean zero

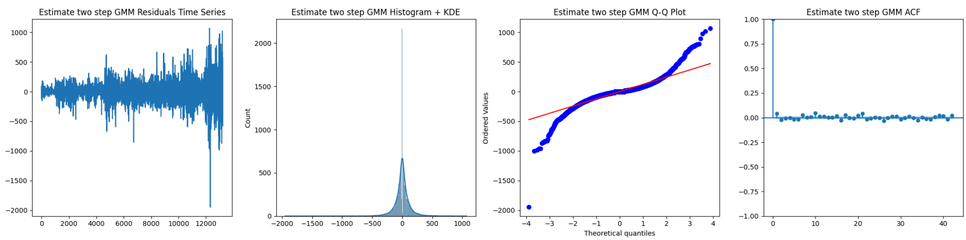
- Ljung-Box: Q(10) = 0.117, p = 0.732 ⟶ No autocorrelation
- Breusch-Pagan Test: p = 0.676 ⟶ Homoskedastic
- Jarque-Bera Normality Test: JB stat = 0.229, p = 0.892 ⟶ Normal

According to the results obtained, the estimated parameters are fairly close to the true parameters. Residuals are normally distributed, uncorrelated, with zero mean and constant variance.

Parameter Estimation on Real Data

In this section, we used a dataset from the ***Market Yield on U.S. Treasury Securities at 10-Year Constant Maturity, Quoted on an Investment Basis***[2]. We performed *two-step GMM* on **daily**, **weekly**, and **monthly** records to observe how the estimation varies with the time step.

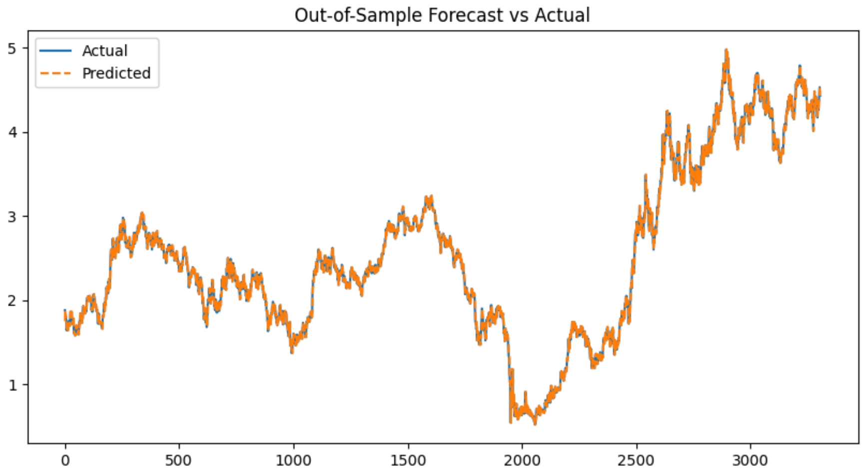
Estimation on Daily Records:



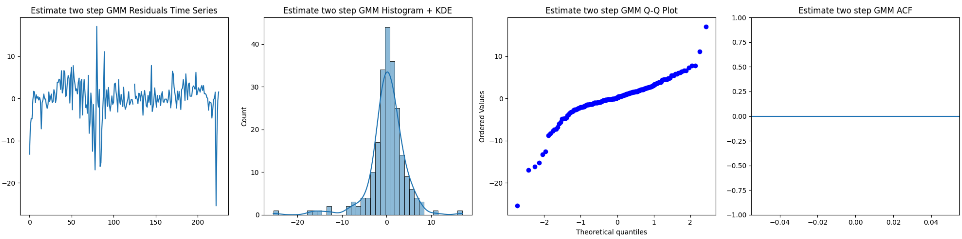
Estimated parameters: $\hat{\theta} = \begin{pmatrix} 0.001 \\ -0.0002 \\ 0.0008 \\ 0.9 \end{pmatrix}$. For the statistical tests performed on the residuals, we have:

- Mean Zero Test: t = −0.076, p = 0.939 ⟶ Mean zero
- Ljung-Box: Q(10) = 21.307, p = 0.000 ⟶ Autocorrelation detected
- Breusch-Pagan Test: p = 0.000 ⟶ Heteroskedasticity detected
- Jarque-Bera Normality Test: JB stat = 74645.773, p = 0.000 ⟶ Non-normal

For *Backtesting on 20% of the data*, we obtained the following prediction:



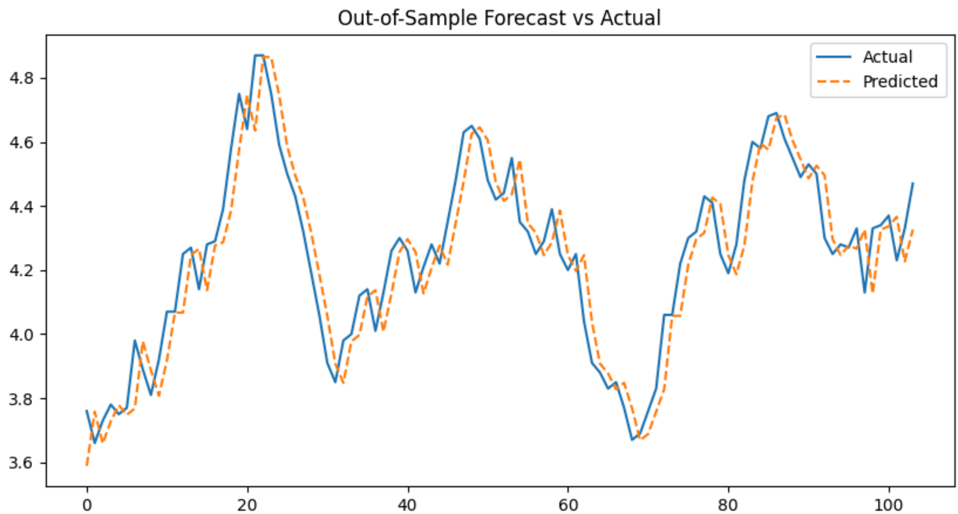
Estimation on Weekly Records:



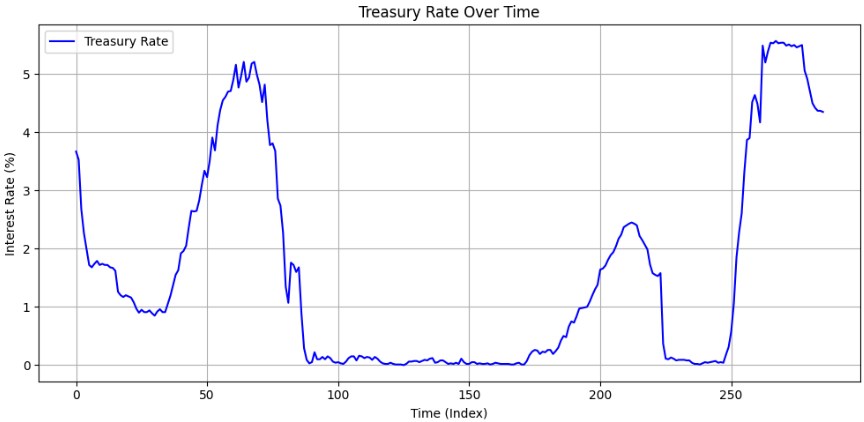
Estimated parameters: $\hat{\theta} = \begin{pmatrix} 0.010 \\ -0.003 \\ 0.004 \\ 0.434 \end{pmatrix}$. For the statistical tests performed on the residuals, we have:

- Mean Zero Test: t = 0.048, p = 0.962 $\longrightarrow$ Mean zero
- Ljung-Box: Q(10) = 18.846, p = 0.000 $\longrightarrow$ Autocorrelation detected
- Breusch-Pagan Test: p = 0.225 $\longrightarrow$ Homoskedastic
- Jarque-Bera Normality Test: JB stat = 49.013, p = 0.000 $\longrightarrow$ Non-normal

For *Backtesting on 20% of the data*, we obtained the following prediction:



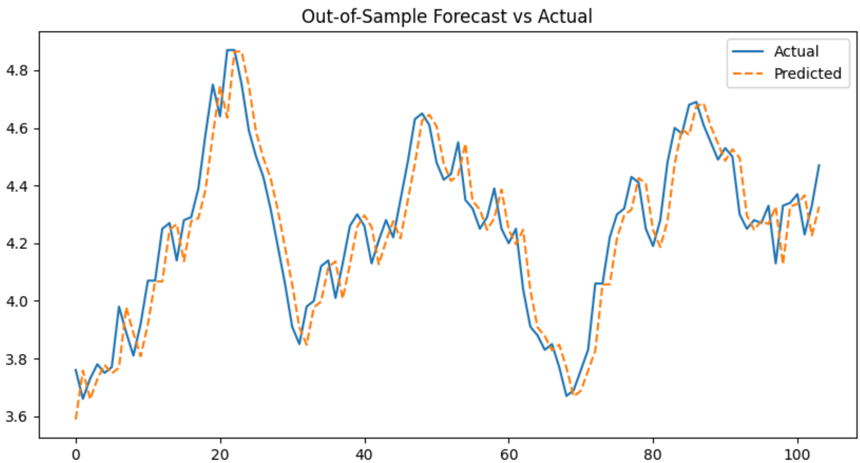
Estimation on Monthly Records:



Estimated parameters: $\hat{\theta} = \begin{pmatrix} 0.002 \\ -0.014 \\ 0.04 \\ 0.325 \end{pmatrix}$. For the statistical tests performed on the residuals, we have:

- Mean Zero Test: t = nan, p = nan $\longrightarrow$ Mean not zero
- Ljung-Box: Q(10) = nan, p = nan $\longrightarrow$ Autocorrelation detected
- Breusch-Pagan Test: p = nan $\longrightarrow$ Heteroskedasticity detected
- Jarque-Bera Normality Test: JB stat = ∞, p = 0 $\longrightarrow$ Non-normal

For *Backtesting on 20% of the data*, we obtained the following prediction:



Alternative Methods for Estimation

As we can notice from the above analysis, the **GMM** method gives a fairly acceptable estimation of the target parameters on synthetic data. However, the method fails to estimate the parameters on real data, since the residuals do not adhere to the theoretical characteristics. This may be due to the dataset being far from a mean-reverting process, which makes the model itself inadequate to describe the dynamics of this time series.

If we proceed alternatively by trying to find a linear regression framework to estimate the target parameters as follows:

$$r_{t+1} = r_t + \alpha + \beta r_t + \sigma r_t^\gamma Z_t \quad Z_t \sim N(0,1)$$
$$\Delta r_t - (\alpha + \beta r_t) = \sigma r_t^\gamma Z_t$$

which gives :

$$\log(\Delta r_t - (\alpha + \beta r_t)) = \log(\sigma) + \log(Z_t) + \gamma \log(r_t) \quad (\star)$$
$$\Delta r_t \approx \alpha + \beta r_t + \varepsilon_t \quad (\star\star)$$

The equations $(\star)$ and $(\star\star)$ allow for linear regression and derivation of the target parameters. However, the residuals in linear regression are assumed to be i.i.d., which is not valid -in our case- since the residuals in $(\star\star)$ are $\varepsilon_t = \sigma r_t^\gamma Z_t$. Hence, linear regression is excluded.

Remark:

The complete implementation is provided in **gmm_estim.py**. The results obtained from both synthetic and real data are presented in the notebook **GMM.ipynb**. The real datasets used are **data_day.csv**, **data_week.csv**, and **data_month.csv**, corresponding to daily, weekly, and monthly observations, respectively.

Bibliography

[1] K. C. Chan, G. A. Karolyi, F. A. Longstaff, and A. B. Sanders, "An Empirical Comparison of Alternative Models of the Short-Term Interest Rate," *The Journal of Finance*, vol. 47, no. 3, pp. 1209–1227, 1992, doi: 10.1111/j.1540-6261.1992.tb04650.x.

[2] Board of Governors of the Federal Reserve System (US), "Market Yield on U.S. Treasury Securities at 10-Year Constant Maturity, Quoted on an Investment Basis." [Online]. Available: https://fred.stlouisfed.org/series/GS10