Model Validation in Non-Linear Continuous-Discrete Grey-Box Models

Jan Holst†, Erik Lindström†, Henrik Madsen* and Henrik Aalborg Nielsen

†Division of Mathematical Statistics, Centre for Mathematical Sciences
Lund Institute of Technology, Lund, Sweden

* Informatics and Mathematical Modeling
Technical University of Denmark, Kongens Lyngby, Denmark
Class of models

General model:

\[ dX(t) = \mu(t, u(t), X(t), \theta)dt + \sigma(t, u(t), X(t), \theta)dW(t) \]

\[ Y(t_k) = h(t_k, u(t_k), X(t_k), \theta) + e_k \]

where

- \( \mu(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \mapsto \mathbb{R}^n \)

The integrals are interpreted in the sense of Itô.
Class of models

General model:

\[ dX(t) = \mu(t, u(t), X(t), \theta)dt + \sigma(t, u(t), X(t), \theta)dW(t) \]
\[ Y(t_k) = h(t_k, u(t_k), X(t_k), \theta) + e_k \]

where

- \( \mu(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}^n \)
- \( \sigma(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}^{n \times m} \)

The integrals are interpreted in the sense of Itô.
Class of models

General model:

\[ dX(t) = \mu(t, u(t), X(t), \theta)dt + \sigma(t, u(t), X(t), \theta)dW(t) \]
\[ Y(t_k) = h(t_k, u(t_k), X(t_k), \theta) + e_k \]

where

- \( \mu(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}^n \)
- \( \sigma(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}^{n \times m} \)
- \( dW(t) \) is a \( m \)-dimensional Wiener process.

The integrals are interpreted in the sense of Itô.
Class of models

General model:

\[ dX(t) = \mu(t, u(t), X(t), \theta) dt + \sigma(t, u(t), X(t), \theta) dW(t) \]
\[ Y(t_k) = h(t_k, u(t_k), X(t_k), \theta) + e_k \]

where

- \( \mu(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}^n \)
- \( \sigma(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}^{n \times m} \)
- \( dW(t) \) is a \( m \)-dimensional Wiener process.
- \( h(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}^p \)

The integrals are interpreted in the sense of Itô.
Class of models

General model:

\[ dX(t) = \mu(t, u(t), X(t), \theta)dt + \sigma(t, u(t), X(t), \theta)dW(t) \]
\[ Y(t_k) = h(t_k, u(t_k), X(t_k), \theta) + e_k \]

where

- \( \mu(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \mapsto \mathbb{R}^n \)
- \( \sigma(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \mapsto \mathbb{R}^{n \times m} \)
- \( dW(t) \) is a \( m \)-dimensional Wiener process.
- \( h(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R}^n \times \mathbb{R}^d \mapsto \mathbb{R}^p \)
- \( e_k \in \mathcal{N}(0, \mathbf{S}), e_k \in \mathbb{R}^p \)

The integrals are interpreted in the sense of Itô.
Overview

Three tools on model validation in continuous-discrete time grey-box models:

• Dependence identification.
Overview

Three tools on model validation in continuous-discrete time grey-box models:

- Dependence identification.
- Model structure identification.
Overview

Three tools on model validation in continuous-discrete time grey-box models:

- Dependence identification.
- Model structure identification.
- Generalized Gaussian residuals.
Dependence identification

It is often reasonable, under some regularity conditions, to approximate the system as Gaussian. This means that filter methods can be used to calculate conditional mean and covariance.

- Define the one-step prediction error:

\[
r_k = \frac{Y_{tk} - E[Y_{tk} | \mathcal{F}_{k-1}]}{\sqrt{V[Y_{tk} | \mathcal{F}_{k-1}]}} ,
\]

where \( \mathcal{F}_{k-1} \) is the available information at time \( t_{k-1} \).
Dependence identification

It is often reasonable, under some regularity conditions, to approximate the system as Gaussian. This means that filter methods can be used to calculate conditional mean and covariance.

- Define the one-step prediction error:

\[ r_k = \frac{Y_{tk} - E[Y_{tk} | \mathcal{F}_{k-1}]}{\sqrt{V[Y_{tk} | \mathcal{F}_{k-1}]}}, \]

where \( \mathcal{F}_{k-1} \) is the available information at time \( t_{k-1} \).

- Non-equidistantly sampled data is normalized.
**Dependence identification, cont.**

- The squared multiple correlation coefficient is given by

\[
\rho^2_{0(1\ldots k)} = \frac{V[Y] - V[Y|X_1, \ldots, X_k]}{V[Y]}. 
\]
Dependence identification, cont.

- The squared multiple correlation coefficient is given by

\[ \rho^2_{0(1...k)} = \frac{V[Y] - V[Y|X_1, \ldots, X_k]}{V[Y]} . \]

- Assuming normality and maximum likelihood estimates yields

\[ R^2_{0(1...k)} = \frac{SS_0 - SS_{0(1...k)}}{SS_0} , \]

where \( SS_0 = \sum (y_i - \sum (y_i/N))^2 \) and \( SS_{0(1...k)} \) is calculated conditional on \( X_1, \ldots, X_k \).
Dependence identification, cont.

- $\rho^2$ is very similar to squared Sample AutoCorrelation Function!

The SACF is obtained as a special case of the LDF if a linear model is used to calculate $q R^2_0(k)$. 
Dependence identification, cont.

- $\rho^2$ is very similar to squared Sample AutoCorrelation Function!
- Let $f_k(r) = E[r_k | r_{k-n} = r]$ and define Lag Dependent Function as

$$LDF(k) = \text{sign} \left( \hat{f}_k(r_{\text{max}}) - \hat{f}_k(r_{\text{min}}) \right) \sqrt{(R^2_{0(k)})^+}.$$
Dependence identification, cont.

- \( \rho^2 \) is very similar to squared Sample AutoCorrelation Function!
- Let \( f_k(r) = E[r_k | r_{k-n} = r] \) and define *Lag Dependent Function* as

\[
LDF(k) = \text{sign} \left( \hat{f}_k(r_{\text{max}}) - \hat{f}_k(r_{\text{min}}) \right) \sqrt{(R^2_{0(k)})^+}.
\]

- The SACF is obtained as a special case of the LDF if a linear model is used to calculate \( \sqrt{R^2_{0(k)}} \).
Dependence identification, cont.

- We can define *Partial Lag Dependent Functions* in a similar fashion.
Dependence identification, cont.

- We can define *Partial Lag Dependent Functions* in a similar fashion.
- The partial square correlation coefficient is given by

\[
\rho^2_{(0k)|(1\ldots k-1)} = \frac{V[Y|X_1, \ldots, X_{k-1}] - V[Y|X_1, \ldots, X_k]}{V[Y|X_1, \ldots, X_{k-1}]}.
\]
Dependence identification, cont.

- We can define *Partial Lag Dependent Functions* in a similar fashion.
- The partial square correlation coefficient is given by

$$\rho^2_{(0k)|(1...k-1)} = \frac{V[Y|X_1, \ldots, X_{k-1}] - V[Y|X_1, \ldots, X_k]}{V[Y|X_1, \ldots, X_{k-1}]}.$$  

- Warning: Defining the *Partial Lag Dependent Functions* for general non-linear autoregressive models

$$X(n) = g(X(n - 1), \ldots, X(n - k)) + \varepsilon(n)$$

is not feasible for large $k$. 
Dependence identification, cont.

- Confidence intervals are obtained by bootstrap under the hypothesis of a i.i.d. process (for LDF and PLDF).
Dependence identification, cont.

- Confidence intervals are obtained by bootstrap under the hypothesis of a i.i.d. process (for LDF and PLDF).
- The distribution for the LDF will be symmetric about zero. Hence, an upper confidence limit for $|LDF(k)|$ or $|PLDF(k)|$ is to be approximated.
Dependence identification, cont.

- Confidence intervals are obtained by bootstrap under the hypothesis of a i.i.d. process (for LDF and PLDF).
- The distribution for the LDF will be symmetric about zero. Hence, an upper confidence limit for $|LDF(k)|$ or $|PLDF(k)|$ is to be approximated.
- Confidence intervals will in general be wider than corresponding intervals for linear models (SACF and/or SPACF).
Dependence identification, cont.

1000 observations from a non-linear MA(1) model:

\[ X_t = e_t + 2 \cos e_{t-1}, \]

Sample Autocorrelation (left) and Lag dependent function (right)
Model structure identification

- Most physical, chemical or biological systems are thought of having a complex deterministic behavior and only small state noise.
Model structure identification

- Most physical, chemical or biological systems are thought of having a complex deterministic behavior and only small state noise.
- The system noise can sometimes be interpreted as model deficiency.
Model structure identification

- Most physical, chemical or biological systems are thought of having a complex deterministic behavior and only small state noise.
- The system noise can sometimes be interpreted as model deficiency.
- Filtering techniques allow us to separate the state noise from the measurement noise.
Model structure identification

- Most physical, chemical or biological systems are thought of having a complex deterministic behavior and only small state noise.
- The system noise can sometimes be interpreted as model deficiency.
- Filtering techniques allow us to separate the state noise from the measurement noise.
- Idea: Use the estimated elements in the diffusion term to pinpoint model deficiencies.
Model structure identification, cont.

Pseudoalgorithm:

- Test the residuals for dependence using e.g. LDF or PLDF etc.

Repeat until no significant dependence remains.
Model structure identification, cont.

Pseudoalgorithm:

- Test the residuals for dependence using e.g. LDF or PLDF etc.
- Extending the state space, guided by large elements in the diffusion term.

Repeat until no significant dependence remains.
Model structure identification, cont.

Pseudoalgorithm:

- Test the residuals for dependence using e.g. LDF or PLDF etc.
- Extending the state space, guided by large elements in the diffusion term.
- Pinpoint dependence by non-parametric regression.

Repeat until no significant dependence remains.
Model structure identification, cont.

Pseudoalgorithm:

- Test the residuals for dependence using e.g. LDF or PLDF etc.
- Extending the state space, guided by large elements in the diffusion term.
- Pinpoint dependence by non-parametric regression.
- Expand the state space model to model dependence.

Repeat until no significant dependence remains.
Model structure identification, cont.

We can test parameter dependence by extending the state space. This can also be generalized to other functional relations $r(t) = \varphi(t, u(t), X(t), \theta)$.

Example: Let $r(t) = \theta^{(j)}$. We test this by extending the state space

$$
\begin{pmatrix}
    dX(t) \\
    dr(t)
\end{pmatrix} =
\begin{pmatrix}
    \mu(\cdot) \\
    0
\end{pmatrix} dt +
\begin{pmatrix}
    \sigma(\cdot) & 0 \\
    0 & \sigma_r
\end{pmatrix}
\begin{pmatrix}
    dW(t) \\
    dW^*(t)
\end{pmatrix}.
$$

We expect the added state (a fixed parameter) to be independent of $t, u(t)$ and $X(t)$. 
Model structure identification, cont.

Example: Simulated model of a fed-batch bioreactor.

\[
\begin{align*}
    dX(t) &= \left( \mu(S(t))X(t) - F(t)X(t)/V(t) \right) \, dt + \sigma_{11} dW(t)^{(1)} \\
    dS(t) &= \left( -\mu(S(t))X(t)/Y(t) + F(t)(S^F(t) - S(t))/V(t) \right) \, dt + \sigma_{22} dW(t)^{(2)} \\
    dV(t) &= F(t) \, dt + \sigma_{33} dW(t)^{(3)}
\end{align*}
\]

where

- \( X(t) \) is biomass concentration
- \( S(t) \) is substrate concentration
- \( V(t) \) is the volume
- \( F(t) \) is the feed flow rate, \( S^F(t) \) is the feed concentration of substrate and \( Y(t) \) is the yield coefficient of biomass.
Finally, $\mu(S)$ is the biomass growth rate given by

$$
\mu(S) = \mu_{\text{max}} \frac{S}{K_2 S^2 + S + K_1}.
$$

Measurement equation:

$$
\begin{bmatrix}
y_1 \\
y_2 \\
y_3
\end{bmatrix}_k =
\begin{bmatrix}
X \\
S \\
V
\end{bmatrix}_k + e_k
$$

where

$$
e_k \in N(0, \mathbf{S}), \quad \mathbf{S} =
\begin{bmatrix}
S_{11} & 0 & 0 \\
0 & S_{22} & 0 \\
0 & 0 & S_{33}
\end{bmatrix}.
$$
Model structure identification, cont.

Test the model \( (\mu(S) = \mu) \) where we have extended the state space.

\[
\begin{align*}
    dX(t) &= \left( \mu(t)X(t) - F(t)X(t)/V(t) \right) dt + \sigma_{11}dW(t) \\
    dS(t) &= \left( -\mu(t)X(t)/Y(t) + F(t)(S(t)^F - S(t))/V(t) \right) dt + \sigma_{22}dW(t) \\
    dV(t) &= F(t)dt + \sigma_{33}dW(t) \\
    d\mu(t) &= 0dt + \sigma_{44}dW(t)
\end{align*}
\]

The parameters and states are estimated using an Extended Kalman filter.
Model structure identification, cont.

Parameter estimates (Quasi Maximum Likelihood):

<table>
<thead>
<tr>
<th>Parameter</th>
<th>True</th>
<th>Estimate</th>
<th>Std dev</th>
<th>t-score</th>
<th>Significant?</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_0$</td>
<td>1.0</td>
<td>1.0239E+00</td>
<td>4.9566E-03</td>
<td>206.5723</td>
<td>Yes</td>
</tr>
<tr>
<td>$S_0$</td>
<td>0.25</td>
<td>2.3282E-01</td>
<td>1.1735E-02</td>
<td>19.8405</td>
<td>Yes</td>
</tr>
<tr>
<td>$V_0$</td>
<td>1.0</td>
<td>1.0099E+00</td>
<td>3.8148E-03</td>
<td>264.7290</td>
<td>Yes</td>
</tr>
<tr>
<td>$\mu_0$</td>
<td>-</td>
<td>7.8658E-01</td>
<td>2.4653E-02</td>
<td>31.9061</td>
<td>Yes</td>
</tr>
<tr>
<td>$\sigma_{11}$</td>
<td>0</td>
<td>2.0791E-18</td>
<td>1.4367E-17</td>
<td>0.1447</td>
<td>No</td>
</tr>
<tr>
<td>$\sigma_{22}$</td>
<td>0</td>
<td>1.1811E-30</td>
<td>1.6162E-29</td>
<td>0.0731</td>
<td>No</td>
</tr>
<tr>
<td>$\sigma_{33}$</td>
<td>0</td>
<td>3.1429E-04</td>
<td>2.0546E-04</td>
<td>1.5297</td>
<td>No</td>
</tr>
<tr>
<td>$\sigma_{44}$</td>
<td>-</td>
<td>1.2276E-01</td>
<td>2.5751E-02</td>
<td>4.7674</td>
<td>Yes</td>
</tr>
<tr>
<td>$S_{11}$</td>
<td>0.01</td>
<td>7.5085E-03</td>
<td>9.9625E-04</td>
<td>7.5368</td>
<td>Yes</td>
</tr>
<tr>
<td>$S_{22}$</td>
<td>0.001</td>
<td>1.1743E-03</td>
<td>1.6803E-04</td>
<td>6.9887</td>
<td>Yes</td>
</tr>
<tr>
<td>$S_{33}$</td>
<td>0.001</td>
<td>1.1317E-02</td>
<td>1.3637E-03</td>
<td>8.2990</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Model structure identification, cont.

Partial dependence plot of $\hat{u}_k|k$ vs. $\hat{X}_k|k$ and $\hat{u}_k|k$ vs. $\hat{S}_k|k$. 
Generalized Gaussian residuals

Univariate but state dependence in diffusion term is no theoretical problem.

\[ dX(t) = \mu(t, u(t), X(t), \theta)dt + \sigma(t, u(t), X(t), \theta)dW(t), \]
\[ Y(t_k) = X(t_k). \]

where

- \( \mu(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R} \times \mathbb{R}^d \mapsto \mathbb{R} \)
Generalized Gaussian residuals

Univariate but state dependence in diffusion term is no theoretical problem.

\[ dX(t) = \mu(t, u(t), X(t), \theta) dt + \sigma(t, u(t), X(t), \theta) dW(t), \]
\[ Y(t_k) = X(t_k). \]

where

- \( \mu(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R} \times \mathbb{R}^d \mapsto \mathbb{R} \)
- \( \sigma(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R} \times \mathbb{R}^d \mapsto \mathbb{R} \)
Generalized Gaussian residuals

Univariate but state dependence in diffusion term is no theoretical problem.

\[
dX(t) = \mu(t, u(t), X(t), \theta)dt + \sigma(t, u(t), X(t), \theta)dW(t),
\]
\[
Y(t_k) = X(t_k).
\]

where

- \(\mu(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R} \times \mathbb{R}^d \mapsto \mathbb{R}\)
- \(\sigma(t, u(t), X(t), \theta) : [0, T] \times \mathbb{R}^l \times \mathbb{R} \times \mathbb{R}^d \mapsto \mathbb{R}\)
- \(dW(t)\) is the increment of a Wiener process.
Generalized Gaussian residuals, cont.

How do we define residuals?

- Discrete time models

\[ X(n + 1) = f(X(n), \theta) + g(X(n), \theta)\varepsilon(n + 1). \]

The i.i.d. sequence \( \{\varepsilon(n)\} \) is the natural choice.
Generalized Gaussian residuals, cont.

How do we define residuals?

- **Discrete time models**

  \[ X(n + 1) = f(X(n), \theta) + g(X(n), \theta)\varepsilon(n + 1). \]

  The i.i.d. sequence \( \{\varepsilon(n)\} \) is the natural choice.

- **Continuous time models**

  \[ X(n+1) = X(n) + \int \mu(s, u(s), X(s), \theta)ds + \int \sigma(s, u(s), X(s), \theta)dW(s). \]

  No natural, general choice!
Generalized Gaussian residuals, cont.

We follow the approach taken in (Pedersen, 1994) but will define Generalized Gaussian Residuals. The reason is threefold

- Uncorrelated Gaussian $\Rightarrow$ independent random variables!
Generalized Gaussian residuals, cont.

We follow the approach taken in (Pedersen, 1994) but will define *Generalized Gaussian Residuals*. The reason is threefold

- Uncorrelated Gaussian $\Rightarrow$ independent random variables!
- Better outlier detection.
Generalized Gaussian residuals, cont.

We follow the approach taken in (Pedersen, 1994) but will define *Generalized Gaussian Residuals*. The reason is threefold

- Uncorrelated Gaussian $\Rightarrow$ independent random variables!
- Better outlier detection.
- More powerful distributional tests.
Generalized Gaussian residuals, cont.

Pseudoalgorithm: Define

\[ x = F^{-1}_{X(n)|X(n-1)}(y) = \inf \{ x : F_{X(n)|X(n-1)}(x) \geq y \} \]

as the generalized inverse. It follows that

- \( U(n) = F_{X(n)|X(n-1)}(X(n)) \in U(0, 1) \)
- \( U(n) \) is the generalized (uniform) residuals introduced by (Pedersen, 1994).

This is the inverse method!

We will define the sequence \( \{ Y(n) \} \) as the 

**Generalized Gaussian Residuals.**
Generalized Gaussian residuals, cont.

Pseudoalgorithm: Define

\[ x = F^{-1}_{X(n)|X(n-1)}(y) = \inf\{ x : F_{X(n)|X(n-1)}(x) \geq y \} \]

as the generalized inverse. It follows that

- \( U(n) = F_{X(n)|X(n-1)}(X(n)) \in U(0, 1) \)
  - \( U(n) \) is the generalized (uniform) residuals introduced by (Pedersen, 1994).

- \( Y(n) = \Phi^{-1}(U(n)) \in N(0, 1) \).

This is the inverse method!

We will define the sequence \( \{Y(n)\} \) as the Generalized Gaussian Residuals.
Algorithm, cont.

Calculation of $F_{X(n)|X(n-1)}(x)$ can be done by using:

- Monte Carlo techniques.
Algorithm, cont.

Calculation of $F_{X(n)|X(n-1)}(x)$ can be done by using:

- Monte Carlo techniques.
- Solving a Partial Differential Equation (Fokker-Planck).
Algorithm, cont.

Calculation of $F_{X(n)|X(n-1)}(x)$ can be done by using:

- Monte Carlo techniques.
- Solving a Partial Differential Equation (Fokker-Planck).
- Other approaches, such as binomial trees, path integration etc.
Algorithm, cont.

- It turns out that simulation is not well suited for this problem as

\[
U(n) = P(X(n) \leq x(n)|X(n-1)) = \frac{1}{N} \sum_{j=1}^{N} 1\{X(n)^{(j)} \leq x(n)\},
\]

which is a discrete approximation of the distribution function with low accuracy in the tails.
Algorithm, cont.

- It turns out that simulation is not well suited for this problem as

\[
U(n) = P(X(n) \leq x(n) | X(n - 1)) = \frac{1}{N} \sum_{j=1}^{N} 1_{\{X(n)^{(j)} \leq x(n)\}},
\]

which is a discrete approximation of the distribution function with low accuracy in the tails.

- Applying \( \Phi^{-1} \) will amplify the errors leading to a bad approximation (possibly \( \Phi^{-1}(0) = -\infty \)).
Algorithm, cont.

- It turns out that simulation is not well suited for this problem as

\[ U(n) = P(X(n) \leq x(n)|X(n - 1)) = \frac{1}{N} \sum_{j=1}^{N} 1\{X(n)^{(j)} \leq x(n)\}, \]

which is a discrete approximation of the distribution function with low accuracy in the tails.

- Applying \( \Phi^{-1} \) will amplify the errors leading to a bad approximation (possibly \( \Phi^{-1}(0) = -\infty \)).

- This problem is magnified by heavy tails, c.f. Jarque-Bera tests.
Algorithm, cont.

The Fokker-Planck (or Kolmogorov forward) equation for a diffusion

\[ dX(t) = \mu(t, u(t), X(t), \theta) \, dt + \sigma(t, u(t), X(t), \theta) \, dW(t), \]

is given by:

\[ \frac{\partial p}{\partial t} = A^* p(x_{t_{i-1}}, t_{i-1}; x_{t_i}, t_i), \]

where the operator \( A^* \) is defined as

\[ A^* f = -\frac{\partial}{\partial x} \left( \mu(t, u(t), x, \theta) f \right) + \frac{1}{2} \frac{\partial^2}{\partial x^2} \left( \sigma^2(t, u(t), x, \theta) f \right). \]

The equation is solved by

- Finite difference approximation of the derivatives
Algorithm, cont.

The Fokker-Planck (or Kolmogorov forward) equation for a diffusion
\[ dX(t) = \mu(t, u(t), X(t), \theta)dt + \sigma(t, u(t), X(t), \theta)dW(t), \]
is given by:
\[ \frac{\partial p}{\partial t} = A^* p(x_{t_{i-1}}, t_{i-1}; x_{t_i}, t_i), \]
where the operator \( A^* \) is defined as
\[ A^* f = -\frac{\partial}{\partial x}(\mu(t, u(t), x, \theta)f) + \frac{1}{2}\frac{\partial^2}{\partial x^2}(\sigma^2(t, u(t), x, \theta)f). \]
The equation is solved by
- Finite difference approximation of the derivatives
- Pade approximation of matrix exponentials
Finally, \( F_{X(n)|X(n-1)}(x) = \int_{-\infty}^{x} p(s, x_s; t, y) dy \) is calculated by a Gauss-Cotes method of the same order of accuracy as the finite difference approximation.

The calculation of the Generalized Gaussian residuals is closely connected to Maximum Likelihood estimation. It is therefore suggested that the model is identified by

- Estimating the parameters in an approximative discrete time model.
Finally, \( F_{X(n)|X(n-1)}(x) = \int_\infty^{-\infty} p(s, x_s; t, y) dy \) is calculated by a Gauss-Cotes method of the same order of accuracy as the finite difference approximation.

The calculation of the Generalized Gaussian residuals is closely connected to Maximum Likelihood estimation. It is therefore suggested that the model is identified by

- Estimating the parameters in an approximative discrete time model.
- Using these estimates as initial values for the Maximum Likelihood estimator.
Example, Cox-Ingersoll-Ross

Model: \( dr_t = 0.17 \cdot (0.05 - r_t) dt + 0.07 \sqrt{r_t} dW_t \).

500 observations, \( \Delta t=1 \). (Simulated data).
Example, Cox-Ingersoll-Ross

Generalized Gaussian residuals.
Example, CKLS

Model: \( dr_t = \alpha(\beta - r_t)dt + \sigma r_t^\gamma dW_t \).

Example, CKLS

Generalized Gaussian residuals. Parameter estimated using an approximative Maximum Likelihood estimator.
Summary

We have presented three tools

- Lag dependent functions to identify dependence.
Summary

We have presented three tools

• Lag dependent functions to identify dependence.
• Non-parametric regression and the estimated state diffusion term to identify model deficiencies.
Summary

We have presented three tools

- Lag dependent functions to identify dependence.
- Non-parametric regression and the estimated state diffusion term to identify model deficiencies.
- Generalized Gaussian Residuals as a definition of residuals to examine the “true” residuals.