Dynamic system classifier

Daniel Pumpe, Maksim Greiner, Ewald Müller, Torsten Enßlin

22.01.2016
OUTLINE

INTRODUCTION

MODEL TRAINING

MODEL SELECTION

CONCLUSION
Motivation - Complex Systems

To classify complex dynamical systems
**Motivation - Complex Systems**

To classify complex dynamical systems
Motivation- Complex Systems

To classify complex dynamical systems
Motivation - Complex Systems

To classify complex dynamical systems
The Goal

To classify complex dynamical systems

\[ \text{s_1} \rightarrow \text{d} \rightarrow \text{s_2} \rightarrow \text{d} \rightarrow \text{s_3} \]

system classes

\[ x(t) [\text{m}] \]

data
Bayes Theorem

"Information is what forces a change in belief" by Caticha

\[ P(s \mid d) = \frac{P(d \mid s)P(s)}{P(d)} \]
STOCHASTIC DIFFERENTIAL EQUATION (SDE)

oscillating dynamical systems

\[
\frac{d^2x(t)}{dt^2} + \gamma \frac{dx(t)}{dt} + \omega^2 x(t) = F(t)
\]
STOCHASTIC DIFFERENTIAL EQUATION (SDE)

complex dynamical systems

\[
\frac{d^2 x(t)}{dt^2} + \gamma(t) \frac{dx(t)}{dt} + \omega_0 e^{\beta(t)} x(t) = \xi(t)
\]
Stochastic differential equation (SDE)

Complex dynamical systems

\[
\frac{d^2 x(t)}{dt^2} + \gamma(t) \frac{dx(t)}{dt} + \omega_0 e^{\beta(t)} x(t) = \xi(t)
\]

Operator form

\[
x_t = R_{tt'}^{(s)} \xi_{t'} \\
\left(R_{tt'}^{(s)}\right)^{-1} = \delta^{(2)} (t - t') - \gamma_t \delta^{(1)} (t - t') + \omega_0 e^{\beta_t} \delta (t - t')
\]
TRAINING DATA
CONSTRUCTION OF THE LIKELIHOOD

\[ \beta_t \quad \gamma_t \quad s \quad R^{(s)} \quad x_t \]

- \( \beta_t \) and \( \gamma_t \): Signal field
- \( s \): Response operator
- \( R^{(s)} \): Response operator
- \( x_t \): Training data
The Likelihood of a SDE

\[ \mathcal{P}(x|s) = \mathcal{G}(x, R^{(s)}\hat{\Xi} R^{(s)}) \]

- temporarily structured covariance
- characterizes a non-stationary processes
THE PRIOR

\[ \mathcal{P}(\beta_t|\Omega) = \mathcal{G}(\beta_t, \Omega) \]
assuming statistical stationarity:

\[ \Omega = \sum_{k} e^{\tau_k} \Omega_k \]
A HIERARCHICAL PRIOR MODEL

inverse Gamma Distribution

\[ \mathcal{P}(\tau^\alpha | \alpha, q) \]
A hierarchical Prior model

inverse Gamma Distribution

\[ \mathcal{P}(e^\tau | \alpha, q) \]

\[ \alpha_{\beta}, \ q_{\beta} \]

\[ \sigma_{\beta} \]

smoothness enforcing

\[ \mathcal{P}(\tau | \sigma) \]
The model training
TRAINING DATA
**RECONSTRUCTED $\gamma_{\text{REC}}$**

![Graph showing reconstructed $\gamma_{\text{REC}}$ over time](image-url)

- **Orginal $\gamma$**
- $x_{1,2}$
- $x_{1,2,3}$
- $x_{1,2,3,4}$
- $x_{1,2,3,4,5}$
- $x_{1,2,3,4,5,6}$
Reconstructed $P(k)$

- $P(k)$
- $P_{\text{Original}}(k)$
- $x_{1,2}$
- $x_{1,2,3}$
- $x_{1,2,3,4}$
- $x_{1,2,3,4,5}$
- $x_{1,2,3,4,5,6}$

Graph showing the comparison between the original and reconstructed distributions of $P(k)$ for different combinations of $x$ values.
MODEL SELECTION

\[ d = R_{\text{OBS}} x + n = R_{\text{OBS}} R^{(s)} \xi + n \]
Model selection

\[ d = R_{OBS} x + n = R_{OBS} R^{(s)} \xi + n \]
Model selection

\[ d = R_{\text{OBS}} x + n = R_{\text{OBS}} R^{(s)} \xi + n \]

\[ \mathcal{P}(s_i | d) = \frac{\mathcal{P}(d | s_i) \mathcal{P}(s_i)}{\mathcal{P}(d)} \]
The Bayesian Network of DSC

\[ \alpha_\beta, q_\beta \quad \sigma_\beta \quad \alpha_\gamma, q_\gamma \quad \sigma_\gamma \]

\[ \tau_\beta \quad \beta_t \quad \gamma_t \quad \tau_\gamma \quad s \quad x_t \quad d \]

\[ R^{(s)} \quad s_1 \quad s_2 \quad s_3 \]

training

classification
MODEL SELECTION - THE SYSTEM CLASSES
**Test Case - SNR=10**

\[ \Delta_{i,j} = \log \mathcal{P}(d|s_i) - \log \mathcal{P}(d|s_j) \]
**Test case - SNR= 10**

\[ \Delta_{i,j} = \log P(d|s_i) - \log P(d|s_j) \]

<table>
<thead>
<tr>
<th>d_{s_1}</th>
<th>SNR=10</th>
<th>\Delta_{i,j=1}</th>
<th>\Delta_{i,j=2}</th>
<th>\Delta_{i,j=3}</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>-4352</td>
<td>-1757</td>
<td></td>
</tr>
</tbody>
</table>
TEST CASE - SNR = 0.01

\[ \Delta_{i,j} = \log P(d|s_i) - \log P(d|s_j) \]
## Performance of DSC

\[ \Delta_{i,j} = \log \mathcal{P}(d|s_i) - \log \mathcal{P}(d|s_j) \]

<table>
<thead>
<tr>
<th>SNR=0.01</th>
<th>( \Delta_{i,j=1} )</th>
<th>( \Delta_{i,j=2} )</th>
<th>( \Delta_{i,j=3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d_{s1} )</td>
<td>0</td>
<td>-6</td>
<td>-5</td>
</tr>
<tr>
<td>( d_{s2} )</td>
<td>-8</td>
<td>0</td>
<td>-10</td>
</tr>
<tr>
<td>( d_{s3} )</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SNR=0.1</th>
<th>( \Delta_{i,j=1} )</th>
<th>( \Delta_{i,j=2} )</th>
<th>( \Delta_{i,j=3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d_{s1} )</td>
<td>0</td>
<td>-144</td>
<td>-55</td>
</tr>
<tr>
<td>( d_{s2} )</td>
<td>-151</td>
<td>0</td>
<td>-139</td>
</tr>
<tr>
<td>( d_{s3} )</td>
<td>-1</td>
<td>-12</td>
<td>-0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SNR=10</th>
<th>( \Delta_{i,j=1} )</th>
<th>( \Delta_{i,j=2} )</th>
<th>( \Delta_{i,j=3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d_{s1} )</td>
<td>0</td>
<td>-4352</td>
<td>-1757</td>
</tr>
<tr>
<td>( d_{s2} )</td>
<td>-6355</td>
<td>0</td>
<td>-5724</td>
</tr>
<tr>
<td>( d_{s3} )</td>
<td>-60</td>
<td>-136</td>
<td>0</td>
</tr>
</tbody>
</table>
CONCLUSION

- DSC algorithm is established:
  1. Analyzes training data from system classes to construct abstract classifying information
  2. Confronts data with the system classes, to state the probability which system class explains observations best

- The classification ability of the DSC-algorithm has successfully been demonstrated in realistic numerical tests

- The DSC-algorithm should be applicable to a wide range of model selection problems
Thanks for your attention!
**Classification - The likelihood**

\[ d = R_{\text{OBS}} x + n = R_{\text{OBS}} R^{(s)} \xi + n. \]

\[ P(d|s_i) = \int D x \ P(d|x) P(x|s_i) \]

\[ = \int D x \ G(d - R_{\text{OBS}} x, N) \]

\[ \times G(x, R^{(s)\dagger} \Xi R^{(s)}) \]

\[ \propto \frac{1}{\sqrt{|D|}} \exp \left( \frac{1}{2} j^\dagger D j \right) \]

with

\[ j = R^{(s)\dagger} R_{\text{OBS}} N^{-1} d \]

and

\[ D^{-1} = R^{(s)\dagger} R_{\text{OBS}} N^{-1} R_{\text{OBS}} R^{(s)} + \Xi^{-1}. \]