Towards an automated prediction and uncertainty quantification system for nuclear models and nuclear data

Georg Schnabel
Holy grail

- Reliable evaluation methods
- Use all the available information
- Deal properly with non-linear models
- Tractable computations

Assistant

Collect all the data from the Internet, read the original publications, analyze the data, find dependencies in the data, find inconsistencies, compare the data to model calculations, improve model parameters within physical sensible boundaries, identify problems with the model, suggest new measurements to efficiently probe models and get essential data for simulations, ...
Bayesian statistics

- Define the prior
- Gather data
- Apply the Bayesian update formula
- Observe the magic and get awesome results
In practice

- Negative cross sections in linearized evaluation methods
- Uncertainty reductions that provoke broad laughter among experimenters
- Model predictions in sharp disagreement with experiment data

Reasons

- Inappropriate prior for model parameters
- Imperfect model
- Inaccurate likelihood specification for the data

Solutions

Prior rescaling, likelihood broadening, model defects, removing suspicious experimental data sets
Simple objective function

$$\log \rho(D | \bar{p}_0, \bar{\sigma}_{\text{exp}}, M) = -\frac{N}{2} \log(2\pi) - \frac{1}{2} \log|M| - \frac{1}{2} (\bar{\sigma}_{\text{exp}} - S\bar{p}_0)^T M^{-1} (\bar{\sigma}_{\text{exp}} - S\bar{p}_0)$$

Entropy

$M = SA_0 S^T + B_{\text{exp}} + K_{\text{def}}$

Questions

Is it computationally tractable?

Can we efficiently maximize this expression?
Bayesian network

\[ M = S A_0 S^T + B_{\text{exp}} + K_{\text{def}} \]

\[ \tilde{p}_1 = \tilde{p}_0 + A_0 S^T M^{-1} (\tilde{\sigma}_{\text{exp}} - S \tilde{p}_0) \]

\[ \tilde{\varepsilon}_{\text{exp,1}} = B_{\text{exp}} M^{-1} (\tilde{\sigma}_{\text{exp}} - S \tilde{p}_0) \]

\[ A_1 = A_0 - A_0 S^T M^{-1} S A_0 \]

\[ B_{\text{exp,1}} = B_{\text{exp}} - B_{\text{exp}} M^{-1} B_{\text{exp}} \]
Deterministic codes

\[ M = S A_0 S^T + B_{\text{exp}} + K_{\text{def}} \]

- Model parameters
- Model prediction
- Observables true values
- Measurement
- Model defect
- Measurement errors

\[ \tilde{p}_1 = \tilde{p}_0 + A_0 S^T M^{-1} (\tilde{\sigma}_{\text{exp}} - S \tilde{p}_0) \]

\[ \tilde{\varepsilon}_{\text{exp},1} = B_{\text{exp}} M^{-1} (\tilde{\sigma}_{\text{exp}} - S \tilde{p}_0) \]

\[ A_1 = A_0 - A_0 S^T M^{-1} S A_0 \]

\[ B_{\text{exp},1} = B_{\text{exp}} - B_{\text{exp}} M^{-1} B_{\text{exp}} \]
Stochastic codes

\[ M = SA_0S^T + B_{\text{exp}} + K_{\text{def}} \]

\[ \tilde{p}_1 = \tilde{p}_0 + A_0S^TM^{-1}(\tilde{\sigma}_{\text{exp}} - S\tilde{p}_0) \]

\[ \tilde{e}_{\text{exp},1} = B_{\text{exp}}M^{-1}(\tilde{\sigma}_{\text{exp}} - S\tilde{p}_0) \]

\[ A_1 = A_0 - A_0S^TM^{-1}SA_0 \]

\[ B_{\text{exp},1} = B_{\text{exp}} - B_{\text{exp}}M^{-1}B_{\text{exp}} \]
G. Schnabel, *Fitting and Analysis Technique for Inconsistent Nuclear Data*

Proc. Int. Conf. on Mathematics & Computational Methods Applied to Nuclear Science & Engineering, April 2017

\[ M = S A_0 S^T + B_{\text{exp}}(\lambda_1, \lambda_2, \ldots) \]
Today

\[ M = P_{\text{stat}} + K_{\text{def}}(\lambda_1, \lambda_2, \ldots) + B_{\text{exp}} \]
Covariance matrices can represent a variety of things such as normalization uncertainties, linear trends, splines, Fourier series, polynomial expansions, white noise, etc.

\[ y(x) = kx + d, \quad k \sim \mathcal{N}(0, \delta_k^2), \quad d \sim \mathcal{N}(0, \delta_d^2) \]

Observations \((y_{\text{exp}}, x_{\text{exp}})\)

\[ \kappa(x_1, x_2) := \text{Cov}[y(x_1), y(x_2)] = \delta_k^2 x_1 x_2 + \delta_d^2 \]

\[ K_{\text{pred,exp}} = \kappa(x_{\text{pred}}, x_{\text{exp}}) \quad K_{\text{exp,exp}} = \kappa(x_{\text{exp}}, x_{\text{exp}}) \]

\[ \bar{y}_{\text{pred}} = K_{\text{pred,exp}} K_{\text{exp,exp}}^{-1} y_{\text{exp}} \]

**Powerful concept**

Directly parametrize covariance matrix and work implicitly with an infinite number of parameters!
Covariance brewing

Combination rules for covariance functions

\[ \kappa_{1+2}(x_1, x_2) = \kappa_1(x_1, x_2) + \kappa_2(x_1, x_2) \]

\[ \kappa_{1 \times 2}(x_1, x_2) = \kappa_1(x_1, x_2) \times \kappa_2(x_1, x_2) \]

Squared exponential covariance functions

\[ \kappa_1(x_1, x_2) = \delta_1^2 \exp\left(\frac{1}{2\lambda_1^2}(x_1 - x_2)^2\right) \]

\[ \kappa_2(x_1, x_2) = \delta_2^2 \exp\left(\frac{1}{2\lambda_2^2}(x_1 - x_2)^2\right) \]

Transition kernel

\[ \tau_1(x_1, x_2) = \sigma(x_1)\sigma(x_2) \]

\[ \tau_2(x_1, x_2) = (1 - \sigma(x_1))(1 - \sigma(x_2)) \]

\[ \sigma(x) = \frac{1}{1 + \exp(-k(x - x_0))} \]

\[ \kappa_{\text{comp}}(x_1, x_2) = \tau_1(x_1, x_2) \times \kappa_1(x_1, x_2) + \tau_2(x_1, x_2) \times \kappa_2(x_1, x_2) \]
Simple example

\[ \kappa_{\text{comp}}(x_1, x_2) = \tau_1(x_1, x_2) \times \kappa_1(x_1, x_2) + \tau_2(x_1, x_2) \times \kappa_2(x_1, x_2) \]

\[ \delta_1 = 5, \lambda_1 = 5, \delta_2 = 1, \lambda_2 = 1, k = 0.5, x_0 = 10 \]

\[ \delta_1 = 5, \lambda_1 = 500, \delta_2 = 1, \lambda_2 = 1, k = 10, x_0 = 10 \]
Real case

INCL vs experiment data

**Final goal:** Inclusive DDX data over the complete nuclide chart projectile + target -> ejectile + X (~100 000 data points above 100 MeV, INCL gives predictions for ~40 000)

**Here:** Inclusive DDX for \( p + \text{target} \rightarrow X + n \) (9287 data points, 11 targets, incident energies ranging from 300 to 3000 MeV)

5 dimensional space (A, Z, EN, ANG, E)

\[ \kappa_1: \delta_1, \lambda_{11}, ..., \lambda_{15} \]

\[ \kappa_2: \delta_2, \lambda_{21}, ..., \lambda_{25} \]

\[ T: k, x_0, \Phi \]

\[
M = P_{\text{stat}} + K_{\text{def}}(\lambda_1, \lambda_2, \ldots) + B_{\exp}
\]

\[
\log \rho(D | \tilde{p}_0, \tilde{\sigma}_{\exp}, M) = -\frac{N}{2} \log(2\pi) - \frac{1}{2} \log |M| - \frac{1}{2} (\tilde{\sigma}_{\exp} - S\tilde{p}_0)^T M^{-1} (\tilde{\sigma}_{\exp} - S\tilde{p}_0)
\]

\[
\kappa_{\text{comp}}(x_1, x_2) = \tau_1(x_1, x_2) \times \kappa_1(x_1, x_2) + \tau_2(x_1, x_2) \times \kappa_2(x_1, x_2)
\]
Introduce sparsity

Assume latent variables (pseudo-inputs)

\[
\tilde{y}_{\text{obs}} = K_{\text{obs},\psi} K_{\psi,\psi}^{-1} \tilde{y}_{\psi} \quad \tilde{y}_{\psi} \sim \mathcal{N}(\tilde{0}, K_{\psi,\psi})
\]

\[
K_{\text{sparse}} = S K_{\psi,\psi} S^T = K_{\text{obs},\psi} K_{\psi,\psi}^{-1} K_{\psi,\text{obs}}
\]

Diagonal correction (essential for continuous optimization)

\[
K_{\text{sparse}} = \text{diag}[K_{\text{obs},\text{obs}} - K_{\text{obs},\psi} K_{\psi,\psi}^{-1} K_{\psi,\text{obs}}] + K_{\text{obs},\psi} K_{\psi,\psi}^{-1} K_{\psi,\text{obs}}
\]


Joint optimization

Efficient computation of objective function (in $O(m^2n)$ instead of $O(n^3)$ with $m$ pseudo-inputs and $n$ observations) and its gradient (Woodbury matrix identity + matrix determinant lemma)

$$\log \rho(D \mid \bar{p}_0, \bar{\sigma}_{\text{exp}}, M) = -\frac{N}{2} \log(2\pi) - \frac{1}{2} \log |M| - \frac{1}{2} (\bar{\sigma}_{\text{exp}} - S\bar{p}_0)^T M^{-1} (\bar{\sigma}_{\text{exp}} - S\bar{p}_0)$$

Scenario

300 pseudo-input points (1500 parameters)
15 hyperparameters (parameters in covariance function)
9287 experiment data points

Timings (not parallelized)

Objective function: 1.3 sec (4 cores: 0.5 sec)
Gradient wrt hyperpars & pseudo-inputs: 50 sec (4 cores: 17 sec)

Optimization on cluster

3000 iterations with L-BFGS-B algorithm in 8 hours using 25 cores (inefficiency: distributed memory)

$\chi^2 / n = 1.03$
Some results at 60 degree

- Al$_{27}$, 800 MeV
- Al$_{27}$, 1200 MeV
- In, 1600 MeV
- C, 800 MeV
- Cu, 1600 MeV
- Fe, 800 MeV
Some results

p on Al27 at 1500 giving n
Some results

$P + Al^{27} \rightarrow X + n$  \quad EN: 1500 MeV  \quad ANG: 30 \, \text{deg}$
Interpolation

\[ P + \text{Al}^{27} \rightarrow X + n \quad \text{EN: 1500 MeV} \quad \text{ANG: 45 deg} \]
Pseudo-Inputs & Hyperpars

<table>
<thead>
<tr>
<th></th>
<th>$\delta$</th>
<th>$\lambda_{EN}$</th>
<th>$\lambda_A$</th>
<th>$\lambda_Z$</th>
<th>$\lambda_{ANG}$</th>
<th>$\lambda_E$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_1$</td>
<td>0.5</td>
<td>99</td>
<td><strong>103</strong></td>
<td><strong>41</strong></td>
<td>68</td>
<td><strong>5</strong></td>
</tr>
<tr>
<td>$K_2$</td>
<td>0.3</td>
<td>272</td>
<td><strong>115</strong></td>
<td><strong>49</strong></td>
<td>64</td>
<td><strong>42</strong></td>
</tr>
</tbody>
</table>

$\tau$: $k = 0.3$, $x_0 = 2.7$
Conclusion

- Reliable evaluation methods must address model imperfections and inconsistent experimental data
- Covariance matrices are versatile and can be used for that purpose
- Maximization of the marginal likelihood (aka $X^2$-fitting with an entropy penalty) is a sound criterion to fit covariance matrices
- Exploiting sparsity enables us to scale up the methods (maybe complete EXFOR?)
Future work & Outlook

- Put it all together
- Remove the linearity assumption
- Automatic covariance structure selection
- Make it accessible to and tuneable by everyone (cloud computing)