Application of Gaussian process regression techniques to experimental plasma profile fitting and model validation

A. Ho¹, J. Citrin¹, C. Bourdelle², F. J. Casson³, K. L. van de Plassche¹, H. Weisen⁴, and JET Contributors*

¹DIFFER - Dutch Institute for Fundamental Energy Research
²CEA, IRFM, Saint-Paul-lez-Durance, France
³CCFE, Culham Science Centre, Abingdon, UK
⁴Swiss Plasma Center, EPFL, Lausanne, Switzerland

*See the author list of Overview of the JET results in support to ITER by X. Litaudon et al.

May 28, 2019
Translating experimental data to analysis model inputs

**Status**
Meaningful interpretation of plasma experiments via modelling rely heavily on fitted plasma profiles

- Requirements on smoothness and feature resolution

**Problem**
Criteria are application-dependent and occasionally user-dependent

- Fit uncertainties improve error propagation if provided
- Difficult to calculate for fit derivative uncertainties for standard routines
Translating experimental data to analysis model inputs

**Status**
Meaningful interpretation of plasma experiments via modelling rely heavily on fitted plasma profiles

- Requirements on smoothness and feature resolution

**Problem**
Criteria are application-dependent and occasionally user-dependent

- Fit uncertainties improve error propagation if provided
- Difficult to calculate for fit derivative uncertainties for standard routines
Brief description of Gaussian process regression

Based on Bayesian statistical principles, probabilistic inference of models (ie. fits in this case) based on given data and other prior knowledge\(^1\)

- Covariance between \(x\) and \(x'\) given by kernel, \(k(x, x', \Theta)\)
- Model selection \(\longrightarrow\) Kernel selection, free parameters \(\longrightarrow\) \(\Theta\)

**Pros:**
- Not limited to a set of basis functions
- Allows simple but rigorous estimation of derivative errors

**Cons:**
- Not robust to outliers, requires careful data filtering
- Not purely convex, initial guess becomes important (prior)

---

\(^1\) C. Rasmussen and C. Williams, (MIT Press, Cambridge, MA, 2006)
Implementation of Gaussian process regression

Developed a 1D regression tool based on Bayesian statistical principles\(^2\)

- Improved speed by using hard-coded kernels and derivatives, implementing optimisation algorithms from machine learning
- Consistently accounts for spatially-varying measurement errors
- Provides self-consistency in sampling of output fit and derivative distributions, each alone can reproduce the other

Available via public git: [https://gitlab.com/aaronkho/GPR1D.git](https://gitlab.com/aaronkho/GPR1D.git)

\(^2\)similar to M. Chilenski et al., Nuclear Fusion 57, 126013 (2017)
Rational quadratic kernel

\[ k(x, x') = \sigma^2 \left( 1 + \frac{(x - x')^2}{2\alpha l^2} \right)^{-\alpha}, \quad \Theta = \{ \sigma, l, \alpha \} \]

- Imposes required smoothness for sufficiently large \( l \)
- Generally fails to resolve sharp features
Gibbs kernel with inverse Gaussian warp

\[ k(x, x') = \sigma^2 \sqrt{\frac{2 l(x) l(x')}{{l}^2(x) + {l}^2(x')}} \exp \left( \frac{(x - x')^2}{{l}^2(x) + {l}^2(x')} \right) \]

\[ l(x) = l_0 - l_1 \exp \left( \frac{(x - \mu)^2}{2\sigma_l^2} \right), \quad \Theta = \{\sigma, l_0, l_1, \mu, \sigma_l\} \]

- Imposes general smoothness through sufficiently large \( l_0 \), like RQ
- Improves fitting of prominent features via \( l_1 \)-term, \( \mu \) near feature
- Not perfectly stable during optimisation, \( \mu \) defined as a fixed parameter
importance of data filtering on fit performances

- Due to Gaussian assumptions, outliers greatly impact optimisation routine
- Removing data points and / or adjusting errors essential to good fits
- Difficult to automate without expert knowledge, but not impossible
GPR fitting of time-averaged JET plasma profiles

- Processed diagnostic measurements not suitable as model inputs as is
- Filtered to remove outliers, errors shown as $2\sigma$ ($\approx 95\%$ confidence)
GPR fitting of time-averaged JET plasma profiles

- Processed diagnostic measurements not suitable as model inputs as is
- Filtered to remove outliers, errors shown as $2\sigma$ ($\approx 95\%$ confidence)

- Fitted for smoothing and interpolation with GPR (via GPR1D tool)
Application to validation of integrated transport model

- JINTRAC transport code$^3$ + QuaLiKiz quasilinear GK turbulent flux$^4$
- Predictive $n_e, T_e, T_i, \Omega_{tor}$: agreement within $2\sigma$ of GPR$^5$
- Monte Carlo propagation of input uncertainty (green) to code outputs (red)

---

3 M. Romanelli et al., Plasma and Fusion Research 9 (2014)
5 A. Ho et al., Nuclear Fusion (accepted 2019) https://doi.org/10.1088/1741-4326/ab065a
Proposed metric for comparing Gaussian distributions

\[ M = \exp \left( -\frac{1}{2} \left( \frac{(\mu_o - \mu_i)^2}{(\sigma_o + \sigma_i)^2} - \frac{(3\sigma_i)^2}{\mu_i^2} - \frac{(3\sigma_o)^2}{\mu_o^2} \right) \right) \]

- Proposed metric\(^6\) only applies to comparison of Gaussian distributions
- Standard metric\(^7\) does not penalize distribution width, can lead to misinterpretation of validation efforts

\(^6\) A. Ho et al., Nuclear Fusion (accepted 2019) [https://doi.org/10.1088/1741-4326/ab065a]

\(^7\) P. Ricci et al., Physics of Plasmas 22, 055704 (2015)
Robustness of GPR1D across wide variety of discharges

- ~13000 time windows from over 2000 discharges processed using the same workflow, same GPR kernels selection criteria and initial conditions
- Provides sufficient fits for initial analysis and uncertainties to evaluate sufficiency with respect to model
Linear analysis shows ITG instabilities exhibit threshold behaviour according to\(^8\):

\[
\left( \frac{R}{L_{T_i}} \right)_{\text{crit}} \simeq \frac{4}{3} \left( 1 + \frac{T_i}{T_e} \right) \left( 1 + 2 \hat{s} \right)
\]

- Steady-state gradients share dependencies with threshold \((T_i/T_e \text{ unclear})\)
- Provides evidence of ITG turbulence optimization via \(q\)-profile shaping

---

Distribution of kernel hyperparameters – Gibbs kernel

$R^2$ shows no trend in hyperparameter space, indicates diversity in data quality and profile shapes.
Using statistical mean of optimized hyperparameters

<table>
<thead>
<tr>
<th>Quantity</th>
<th>$\sigma$</th>
<th>$l_0$</th>
<th>$l_1$</th>
<th>$\sigma_l$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n_e$</td>
<td>0.5</td>
<td>0.35</td>
<td>0.2</td>
<td>0.125</td>
</tr>
<tr>
<td>$T_e$</td>
<td>0.55</td>
<td>0.35</td>
<td>0.15</td>
<td>0.1</td>
</tr>
<tr>
<td>$T_i$</td>
<td>0.55</td>
<td>0.35</td>
<td>0.15</td>
<td>0.05</td>
</tr>
</tbody>
</table>

- Removal of optimization reduces fitting time to $\sim 1$ s per time window
- Trade-off: reduced occurrence of fits with $R^2 \approx 1$
- Similarity of $R^2$ distributions indicate general shallow optimum
- Further analysis of individual profiles needed to draw stronger conclusions
Summary of progress

▸ Developed **Gaussian process regression (GPR) fitting tool, GPR1D, optimised for 1D profile fitting**
  ▸ Can process single time slice / time window in $\sim 30$ s
  ▸ Disadvantage: Not robust to outliers, needs careful data filtering
  ▸ Public git repo: [https://gitlab.com/aaronkho/GPR1D.git](https://gitlab.com/aaronkho/GPR1D.git)

▸ Performed **code validation** exercise (JETTO + QuaLiKiz) with improved statistical rigour using GPR uncertainty information
  ▸ Confidence ranges for model input sensitivity studies
  ▸ Error bars for model output comparison
  ▸ High density sensitivity of integrated model to boundary value

▸ Compiled **JET 1D profile database** of $\sim 13000$ time slices from over 2000 discharges for sampling
Next steps

- Continue with construction of multi-machine profile database using GPR1D tool (addition of AUG)

- Use database as basis for large-scale quasilinear transport model (QuaLiKiz) runs for neural network training sets, for fast and accurate surrogate turbulent transport models (K. L. v/d Plassche)

- Incorporate as standard community tool for preparation and uncertainty quantification of integrated modelling applications

Interesting topics of open discussion:

- Interpretation and usage of error bars: quantification of data variation vs. qualitative representation of trust in data

- Automation of data filtering: is this even necessary with new machine learning paradigm?
Extra Slides
Essential components of Gaussian process regression algorithm

Data: \((x, y)\), Basis function(s): \(\Phi(x)\), Weight(s): \(w\), Error / noise: \(\varepsilon\)

\[
p(w) \sim \mathcal{N}(0, \Sigma_w) \quad p(\varepsilon) \sim \mathcal{N}(0, \sigma_n^2) \quad \Rightarrow \quad p(y|x, w)
\]

Place into Bayesian framework

\[
p(y_*|x_*, x, y) = \int p(y_*|x_*, w) \frac{p(y|x,w)}{p(y|x)} \, dw
\]

Apply kernel trick (covariance of model)

\[
\Phi(x)^T \Sigma_w \Phi(x') + \sigma_n^2(x) \delta_{xx'} \equiv k(x, x', \Theta) + \Sigma_n
\]

Notation: \(K(x, x, \Theta) = k(x = x, x' = x, \theta = \Theta)\)

Data: \((x, y)\), Kernel: \(K(x, x, \Theta)\), Hyperparameter(s): \(\Theta\), Uncertainty: \(\Sigma_n\)
Hyperparameter search with type II ML optimization

Objective / cost function provided by the **log-marginal-likelihood** in Bayesian statistical framework

- Operates as a "goodness-of-fit" measure
- Accounts for all possible fits weighted by its probability, flat prior on $\Theta$

$$
\log p(y|x) = -\frac{1}{2} y^T (K + \Sigma_n)^{-1} y - \frac{\lambda}{2} \log |K + \Sigma_n| - \frac{1}{2} n_x \log 2\pi
$$

- **Goodness of fit**
- **Kernel complexity**
- **Size of data set**

Optimization maximizes log-marginal-likelihood with respect to $\Theta$:

- Gives $\Theta$ with maximum probability to match input data
- $\lambda$ adjusts complexity penalty, used to prevent overfitting (regularization)
Making predictions using the Gaussian process regression

The number and interpretation of the free parameters, $\Theta$, are defined by choice of covariance / kernel function

- Proper choice of covariance function effectively allows an infinite set of basis functions (from a specific family)

Predictive fit equations:

$$
\mathbb{E}[y_*] = K(x_*, x) [K(x, x) + \Sigma_n]^{-1} y
$$

$$
\nabla[y_*] = [K(x_*, x_*) + \Sigma_{n,*}] - K(x_*, x) [K(x, x) + \Sigma_n]^{-1} K(x, x_*)
$$

Predictive fit derivative equations:

$$
\mathbb{E}\left[ \frac{dy}{dx_*} \right] = \frac{dK(x_*, x)}{dx} [K(x, x) + \Sigma_n]^{-1} y
$$

$$
\nabla\left[ \frac{dy}{dx_*} \right] = \frac{d^2[K(x_*, x_*) + \Sigma_{n,*}]}{dx dx'} - \frac{dK(x_*, x)}{dx} [K(x, x) + \Sigma_n]^{-1} \frac{dK(x, x_*)}{dx'}
$$
Possible extension of validation to derivative predictions

- GPR provides consistent profile derivatives and derivative errors (shown as $\pm 2\sigma$)
- Low variability in simulated profile gradients as shown by output envelopes
- Large errors for fit derivatives not always present but possible
Implementation first performs fit on uncertainty values, $\sigma^2 I \rightarrow r(x, x')$

Technique allows spatial dependence of errors with conditions