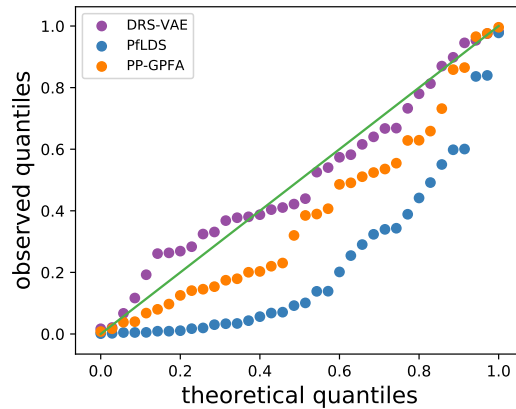
1 **Paper 7339 | Deep Random Splines for Point Process Intensity Estimation of Neural Population Data**

2 We sincerely thank the reviewers for helping us improve the work through their feedback, comments, and suggestions.
3 We now address individual points raised by the reviewers:

4 - **Adding qq-plots (reviewer 2)**: We performed goodness of fit qq-plots as in Brown et al. (2001). At right we display one for an arbitrary neuron and trial, and we can see that our model outperforms PfLDS and PP-GPFA (we do not compare against GPFA as the qq-plot uses the time rescaling theorem, which is not compatible with GPFA). Our method not only recovers better ELBO values and more useful latent representations, it also recovers quantitatively better intensities. This is an important analysis to have in the paper; thank you.

5 - **On comparing against other methods and whether we outperform GP + VI (reviewers 2 and 4)**: Du et al.
6 (2012), Yang et al. (2017), Mei and Eisner (2017) and Du et al. (2016) are all relevant papers which involve
7 estimating intensity functions of point processes: we will carefully include them in our manuscript. Thank you.
8 The first two papers model dynamic networks, making a direct comparison difficult, and the last two do not use
9 latent variables, which is one of the main advantages and goals of our method as a way to perform dimensionality
10 reduction for neural population data. We have also compared against the LFADS model of Pandarinath et al. (2018)
11 and it has deeply underperformed. That said, we will keep working on LFADS and contact the authors to improve
12 how it might perform on our datasets (in case of hyperparameter fickleness or similar). Finally we also note that
13 PP-GPFA, against which we already compared and significantly outperformed on our paper, is a nonparametric
14 model that is GP + VI for neural population data. If the reviewer has a more specific GP + VI model in mind, please
15 let us know and we will gladly compare against it.

16 - **On using B-splines with nonnegative coefficients (reviewer 3)**: We will reference the work of Shen and Ghosal
17 (2015), thank you. In our preliminary experiments, we found B-splines with nonnegative coefficients (and squared
18 splines) to not perform well, which is what motivated the parameterization we used and enabled our good empirical
19 results. Indeed, our parameterization ensures that we can recover every nonnegative spline, in contrast to B-splines:
20 not every nonnegative spline can be written as a linear combination of B-splines with nonnegative coefficients.
21 Thus, while theoretically important, because of this incompleteness, and because there is further no guarantee that
22 their bound is close enough in practice, we believe our parameterization to be a natural choice. We will add this
23 discussion along with our earlier results with B-splines into the paper.

24 - **On GPs and DRS scaling and optimization (reviewers 2 and 4)**: As pointed out, the PP-GPFA method uses GPs
25 and manages to scale using VI (even though we still outperform it). However, this scalability depends critically on
26 the use of inducing points, which is a further approximation scheme. One of the advantages of our method is that
27 it scales well (not cubically, like most GP methods) with respect to most of its parameters like number of trials,
28 number of knots, number of iterations of the alternating projections algorithm, hidden dimension and number of
29 neurons. The only parameter with which our method does not scale as well is the number of spikes. This is due to
30 the RNN-based encoder, which has to process every spike individually (not spike counts over time bins). However,
31 this can be addressed by using a non-amortized inference approach (i.e. not having an encoder and having separate
32 variational parameters for each trial). We found that the amortized approach using our proposed encoder was better
33 for the datasets we analyzed, but even larger datasets might benefit from the non-amortized approach.

34 - **On the comparisons we performed (reviewer 4)**: When comparing against methods that discretize time, we used
35 commonly used bin widths and used those values to select the number of knots for our method. This choice actually
36 makes our comparisons conservative and thus suggests that our results are even more positive than shown, as we
37 can obtain even better results by increasing the number of knots that we use. However, we wanted to make the
38 comparisons as fair as possible by giving methods access to the same number of degrees of freedom. We will make
39 this clearer on the manuscript, thank you for pointing it out. Also, while outperforming GPFA in terms of ELBO
40 may not be surprising as we have access to neural networks, we also obtain more useful latent variables, which
41 again points to the strength of our method. Finally, we also compared against PfLDS, which uses neural networks,
42 and also significantly outperformed it.