Supplementary Material for: **Inferring nonlinear neuronal computation based on physiologically plausible inputs**

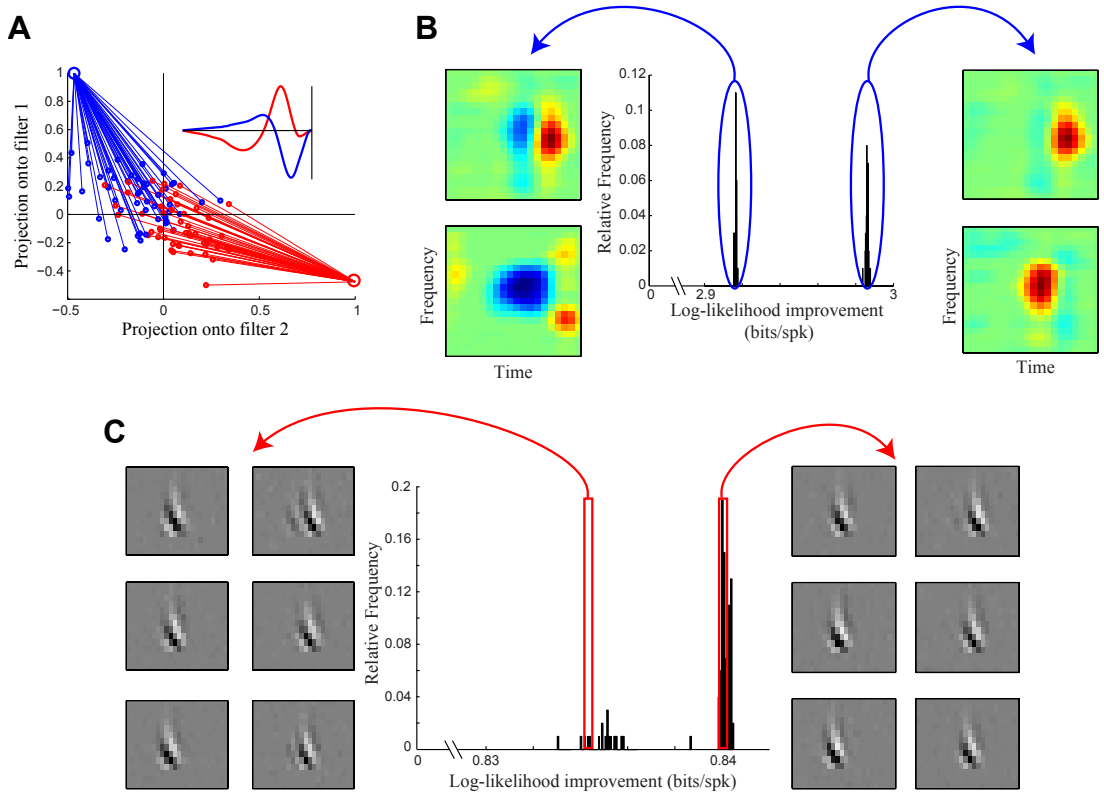

Figure S1: Robustness of filter estimation
A) For the simulated ON-OFF RGC in Figures 1 and 3, the likelihood function with respect to the NIM stimulus filters shows only a single global optimum (up to an interchange of the filters). To illustrate this, 100 iterations of the optimization were performed with random initializations of the filters, and in all cases the correct filters were identified. The initial filters are projected onto the true ON and OFF filters (inset), and are plotted along with the resulting optimized filters. Each iteration of the optimization is thus represented by a pair of optimized filters (large blue and red circles), along with a pair of initial filters (small blue and red circles, color coded based on the resulting filter estimates). **B)** For the example MLd neuron in Figure 5, we found two distinct local maxima when optimizing the NIM stimulus filters, corresponding to the two clusters of the maximized log-likelihood (relative to the null model) across many repetitions of optimizing the filters with random initial conditions. The global optimum (right) corresponds to the set of filters shown in Figure 5, while a locally optimum solution (left) corresponds to the excitatory filter matching the STA. **C)** For the simulated V1 neuron shown in Figure 6A, optimization of the NIM is again well behaved. In this case there are potentially several spurious local maxima, illustrated by the distribution of maximized log-likelihood values. However, these local maxima correspond to very similar models to the identified global maximum, as shown by the similar log-likelihood values, as well as the similarity of the identified filters (example models shown at left and right).

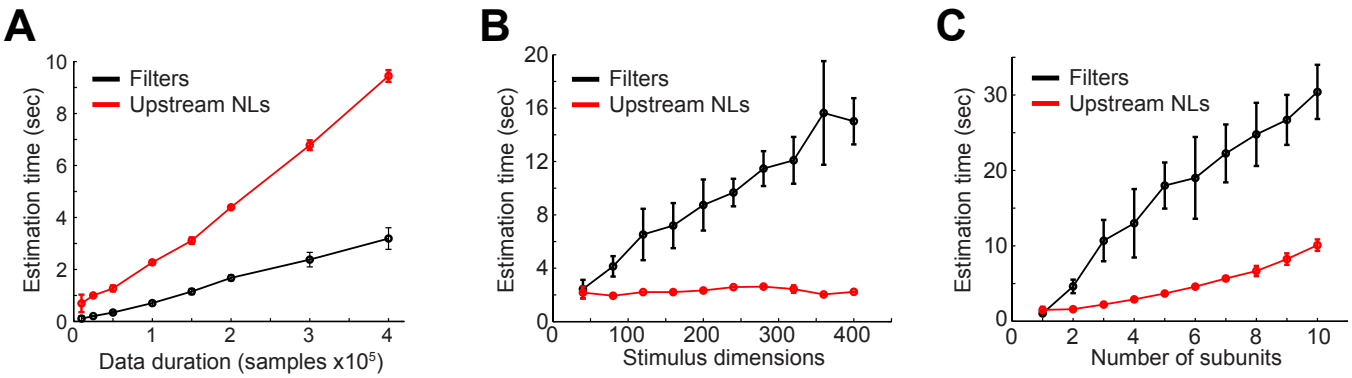

Figure S2: NIM parameter optimization scales approximately linearly

A) The time required to estimate the filters (black) and upstream nonlinearities (red) scales linearly as a function of data duration for the ON-OFF RGC simulation (with two subunits) shown in Figures 1 and 3. The error bars show the region +/- 1 standard deviation around the mean across multiple repetitions of the parameter estimation (with random initialization). Estimation was performed on a machine running Mac OS X 10.6 with two 2.26 GHz quad-core Intel Xeon processors and 16 GB of RAM. **B)** To measure parameter estimation time as a function of the number of stimulus dimensions, we simulated a V1 neuron (similar to that shown in Figure 6A) receiving two rectified inputs (data duration of 10^5 time samples). We then varied the number of time lags used to represent the stimulus and measured the time required for parameter estimation. Estimation of the stimulus filters scales roughly linearly with the number of stimulus dimensions, while estimation of the upstream nonlinearities is largely independent of the number of stimulus dimensions. **C)** Parameter estimation time for the filters and upstream nonlinearities also scales approximately linearly as a function of the number of subunits. Here we again used a simulated V1 neuron similar to that shown in Figure 6A, although with 10 rectified inputs (200 stimulus dimensions and data duration of 10^5 time samples). Note that the additional step of estimating the upstream nonlinearities adds relatively little to the overall parameter estimation time, especially for more complex models.

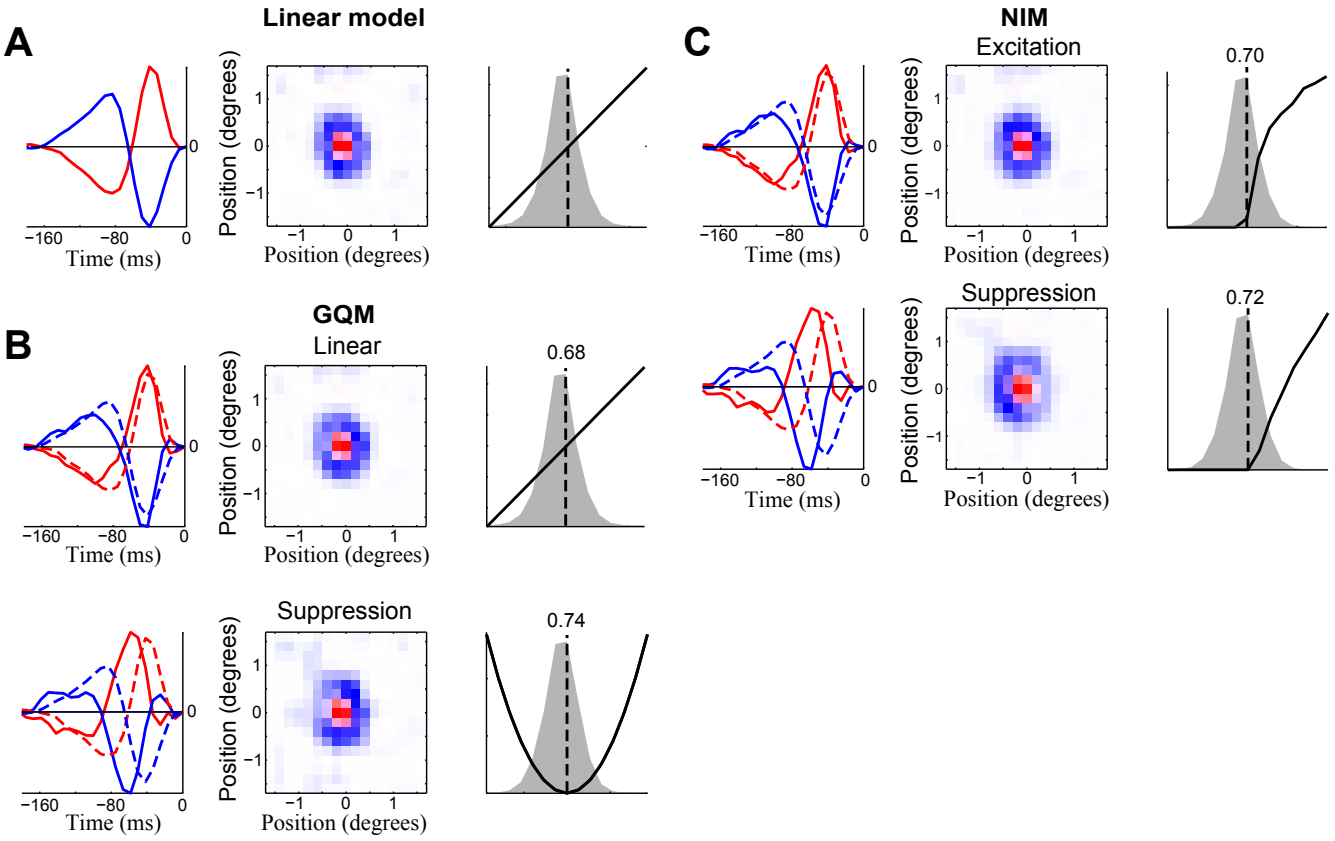

Figure S3: Comparison of the NIM and GQM for the example LGN neuron
The linear model (**A**), NIM (**B**), and GQM (**C**) fit to the example LGN neuron from Figure 4 are shown for comparison. Here (A) and (B) are reproduced from Figures 4A and B respectively. Note that the spatial and temporal profiles of the linear and squared (suppressive) GQM filters are largely similar to the (rectified) excitatory and suppressive filters identified by the NIM. Despite the similarity of the identified filters, however, the NIM and GQM imply a different picture of the neuron's stimulus processing, as illustrated in Figure S4.

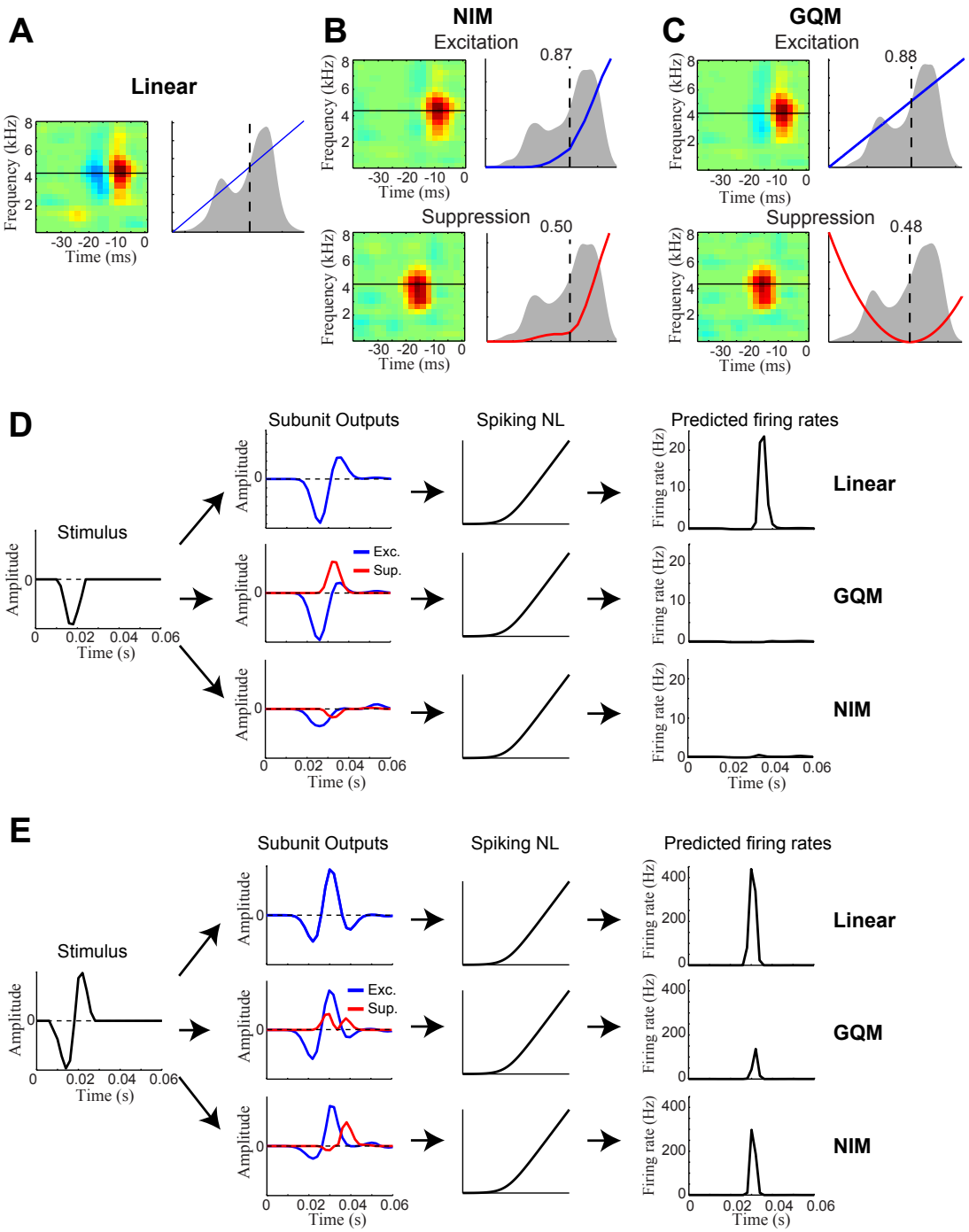

Figure S4: Different predictions of the GQM and NIM with excitation and delayed suppression.
The GLM (**A**), NIM (**B**), and GQM (**C**) fit to the example MLd neuron in Figure 5 (A and B here are reproduced from Figs. 5A and B). The NIM and GQM identify similar excitatory and suppressive filters, but the GQM assumes linear and squaring upstream nonlinearities for these inputs respectively, while the NIM infers the rectified form of these functions. Despite the similarities in the identified filters, the different upstream nonlinearities in these models imply distinct interactions between the excitatory and suppressive inputs. To illustrate this, we consider how these different models process two stimuli in (D) and (E), which highlight these differences. **D)** First, we consider a negative impulse (left) presented at the preferred frequency (horizontal black lines in A-C). The outputs of the excitatory (blue) and suppressive (red) subunits are shown for the linear model (top), GQM (middle), and NIM (bottom). The combined outputs of these subunits are then transformed by the spiking nonlinearity into the corresponding predicted firing rates at right. In this case, only the linear model responds to this stimulus, since the GQM is strongly suppressed, and the NIM is largely unaffected due to the rectification of the negatively driven inputs. **E)** Similar to (D), we consider a biphasic stimulus (left), also presented at the neuron's preferred frequency. This stimulus drives different responses in all three models. The response predicted by the GQM is by far the weakest because the squared suppression driven by the initial negative phase of the stimulus coincides with the excitation driven by the positive phase of the stimulus, causing them to partially cancel each other out. For the NIM, the negative phase of the stimulus does not drive the suppression, due to rectification, and the excitation is able to elicit a much larger response. The response predicted by the linear model is even larger since this is essentially the optimal stimulus for driving the linear filter. This suggests targeted stimuli that might be able to distinguish the computations being performed by MLd neurons.

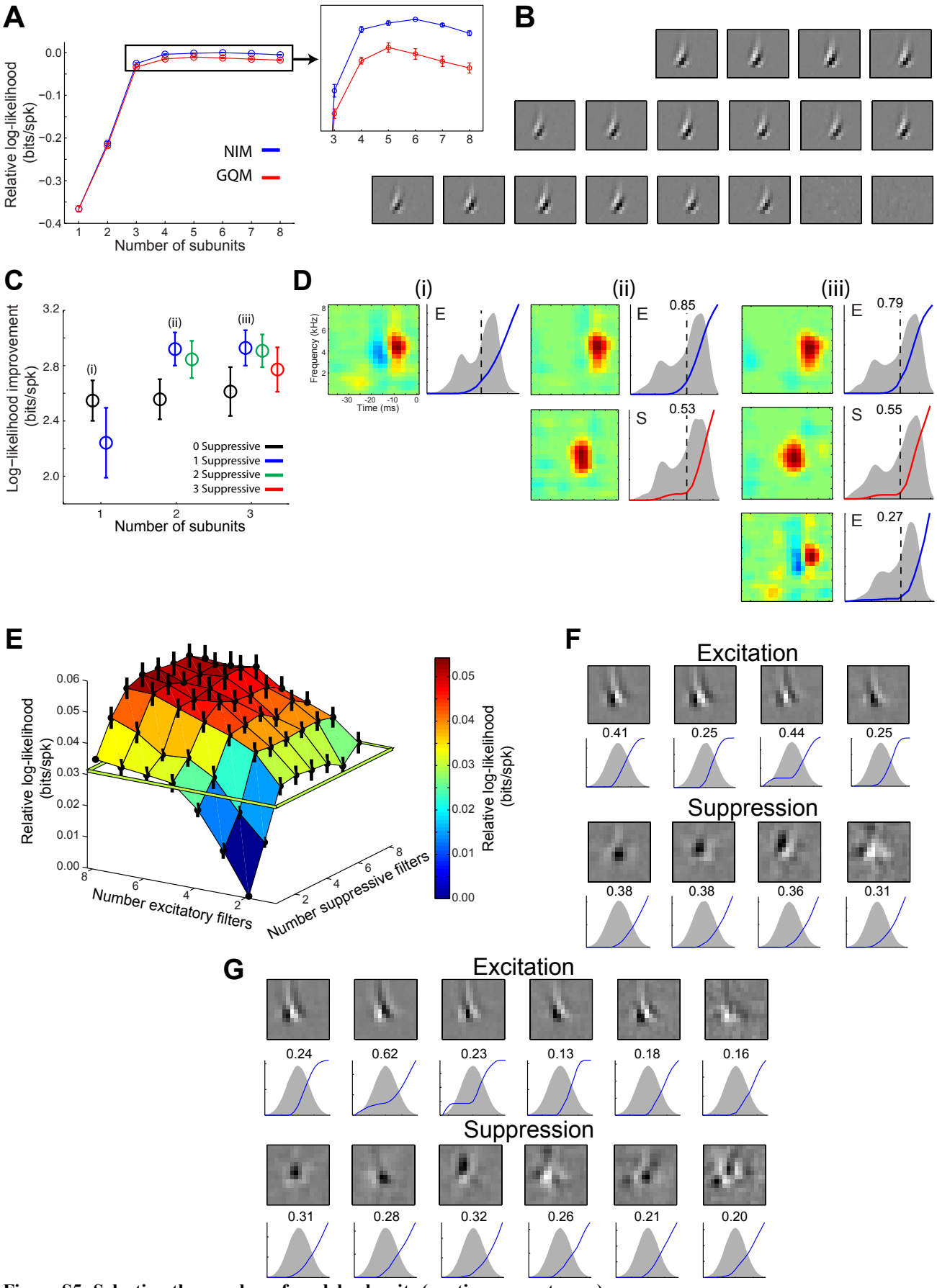

Figure S5: Selecting the number of model subunits (caption on next page):

A) To illustrate the robustness of NIM parameter estimation to specification of the precise number of subunits, we first consider the simulated V1 neuron from Figure 6A, which was constructed from six rectified excitatory subunits. Fitting a sequence of NIMs (blue) and GQMs (red) with increasing numbers of (excitatory) subunits reveals that the log-likelihood (evaluated on a simulated cross-validation data set) initially improves dramatically, but becomes nearly saturated for models with four or more subunits. Here we plot log-likelihood relative to that of the best model, and error bars show one std. dev. about the mean. While it is possible in this case to identify the true number of underlying subunits (six) from the cross-validated model performance of the NIM, the model performance is relatively insensitive to specification of the precise number of subunits. **B)** Stimulus filters from example NIM fits from (A), with four, six, and eight filters. Note that the identified filters are nearly identical across these different models, and when more than the true number (six) of subunits are included in the model, sparseness regularization on the filters tends to drive the extra filters to zero, yielding effectively identical models. **C)** To illustrate the procedure of selecting the number of model subunits with real data, we consider fitting a series of models to the example MLd neuron from Figure 5. In this case there are both excitatory and suppressive stimulus dimensions, so we independently vary the number of each. Average (+/- 1 std. error) cross-validated model performance is depicted for each subunit composition for models with up to three subunits (the color indicates the number of suppressive subunits). While we do not have sufficient data to identify statistically significant differences, a two-filter model with one excitatory and one suppressive filter appears to achieve optimal performance. **D)** Three example NIM fits for one, two, and three-filter models corresponding to the Roman numerals in (C). (i) Model with one excitatory subunit. (ii) Model with one excitatory and one suppressive subunit. (iii) Model with two excitatory and one suppressive subunits. Note that the excitatory and suppressive filters from the two-filter model are also present in the three-filter model, and the addition of a second excitatory subunit (resembling the linear filter) provides little, if any, additional predictive power. **E)** Similar to (C-D), we consider fitting models with different numbers of excitatory and suppressive subunits to the example macaque V1 neuron from Figure 7. In this case, the neuron is selective to a large number of stimulus dimensions (Figure 7A), and thus there are a large number of possible excitatory/suppressive subunit compositions to consider. To greatly speed this process (and illustrate a procedure for rapid model characterization), we fit the NIM filters in a reduced stimulus subspace (see Methods) that is identified by a GQM with four excitatory and six suppressive dimensions. The number of subunits in the GQM was selected in order to ensure that all filters with discernible structure were included. The figure then shows the average (+/- 1 std. error) cross-validated log-likelihood improvement (relative to a model with two excitatory and one suppressive filters) for NIMs with varying numbers of excitatory and suppressive subunits. Note that the model performance increases initially, but tends to saturate for models with more than about four excitatory and four suppressive subunits. For comparison, the cross-validated log-likelihood of the GQM (green line) – which established the stimulus subspace – is below most of the NIM solutions. While fitting the stimulus filters in the full stimulus space provides slightly different (though qualitatively very similar) results, limiting the NIM to the subspace provides a tractable way to fully explore the nonlinear structure of computation, and can then serve as an initial guess for a more computationally-intensive search in the full stimulus space. **F-G)** Two example NIM fits from those depicted in (C). A NIM with four excitatory and four suppressive subunits (F) is compared to a NIM with seven excitatory and seven suppressive subunits (G), the latter providing only a slight improvement relative to the former. Both models provide a qualitatively similar depiction of the neuron's stimulus processing, identifying a largely similar set of excitatory and suppressive inputs.

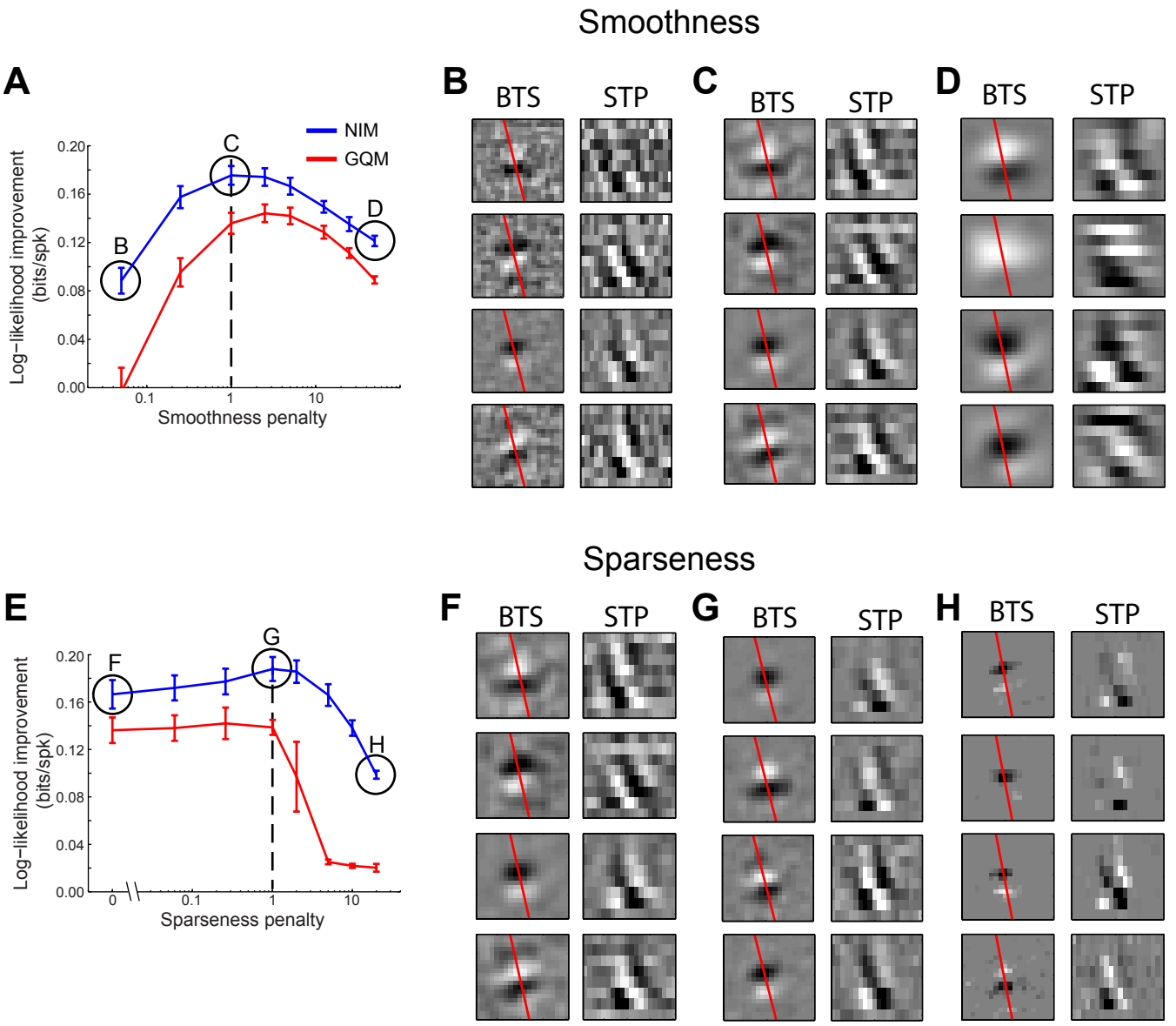

Figure S6: Selection of regularization parameters

To illustrate how the performance of our models depends on selection of the regularization hyperparameters, we fit a series of models to the example V1 neuron from Figure 8. For this example neuron regularization of the stimulus filters is particularly important, given the large number (3200) of parameters associated with each filter. As described in the Methods section, we use both smoothness (L2 penalty on the spatial Laplacian) and sparseness priors on the filters, each of which is governed by a regularization hyperparameter. While in principle we could independently optimize these regularization parameters for each filter, we consider here only the case where all filters are subject to the same regularization penalties. Further, we consider optimizing the smoothness and sparseness penalties independently, which will not in general identify the optimal set of hyperparameters.

A) We first set the sparseness regularization penalty to zero, and systematically vary the strength of the smoothness penalty. The cross-validated log-likelihood improvement is plotted for the NIM (blue trace) and GQM (red trace), showing that the NIM outperforms the GQM over a range of smoothness regularization strengths. **B-D)** Representative filters are shown from model fits at several regularization strengths, as indicated by the black circles in (A). The filters are depicted as the 'best-time slice' (BTS) and the 'space-time projection' (STP), as in Figure 8. **E)** Similar to (A), we next consider varying the strength of sparseness regularization given fixed values for the smoothness regularization (set to the value indicated by the vertical dashed line in A). Note that the performance of the NIM again remains significantly better than the GQM across a range of regularization strengths. **F-H)** Representative filters at several sparseness regularization strengths, as indicated in (E). Note that (F) is identical to (C), reproduced for ease of comparison.