Contents

Chapter 1

Results

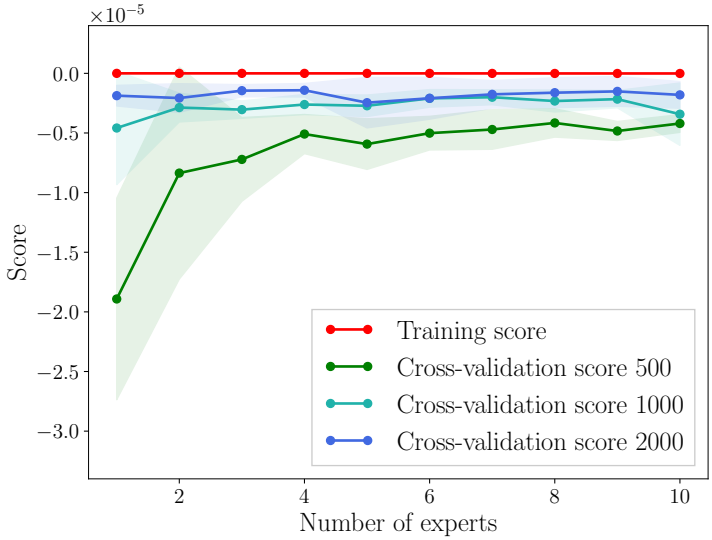
In this chapter Gaussian processes trained on the MSSM-24 dataset are used to predict cross sections for MSSM-24 and CMSSM. The estimator settings are the cummulative settings from the previous chapter, and distributed Gaussian processes are used to include more training data. Learning curves as a function of number of experts are plotted for the cummulative settings. Plots of the relative deviances from the `Prospino` data are shown, and the resulting predictions are compared to data from `Prospino` and `NLL-fast`. Finally, the optimal model with respect to predictive capabilities, model size and computation times is discussed.

1.1 Learning Curves

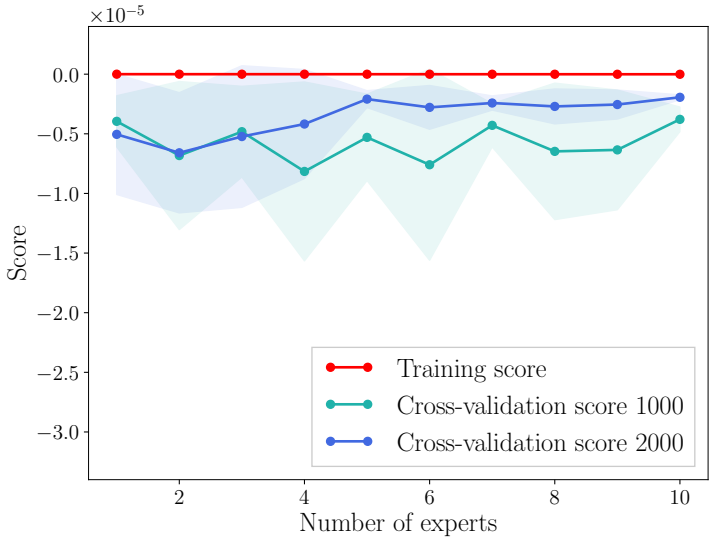
Learning curves for the cummulative settings from Sec. **??** are shown in Fig. 1.1 for $\widetilde{d}_L\widetilde{u}_L$ and $\widetilde{d}_L\widetilde{d}_L$. The experts are trained with 500, 1000 and 2000 points per expert, and learning curves are calculated according to the method described in Sec. **??**.

The training scores for the estimators of $\widetilde{d}_L\widetilde{d}_L$ and $\widetilde{d}_L\widetilde{u}_L$ are 1, indicating that neither model is underfitting. The validation curves for both processes converge towards 1, albeit faster and for less training points per expert for $\widetilde{d}_L\widetilde{d}_L$ than for $\widetilde{d}_L\widetilde{u}_L$. In both cases the training and validation scores are very high, even for few experts. Adding more data, both in the form of more experts and more points per expert, give higher validation scores. Although it appears that including a 'bad' expert can affect the validation score negatively and increase the uncertainty in the score, *e.g.* from 9 to 10 experts with 1000 training points for $\widetilde{d}_L\widetilde{d}_L$, the addition of data generally improves the score. Predictions in this chapter therefore use the largest reasonable[1] models, of 10 experts with 5000 and 8000 training points each, depending on whether or not the models need to be stored.

[1] Taking into account computation times, matrix sizes and model sizes



(a) The process $\widetilde{d}_L\widetilde{d}_L$



(b) The process $\widetilde{d}_L\widetilde{u}_L$

Figure 1.1: Learning curves as a function of number of experts, with 500, 1000 and 2000 training points per expert for the processes **(a)** $\widetilde{d}_L\widetilde{d}_L$ and **(b)** $\widetilde{d}_L\widetilde{u}_L$. The validation curve for $\widetilde{d}_L\widetilde{u}_L$ with 500 training points per expert is omitted because the uncertainty in the validation scores is very large. The k-fold cross validation uses R^2-score as desribed in Sec. **??**, and here $R^2 - 1$ is plotted.

Process	σ_f	$\ell_{m_{\widetilde{g}}}$	$\ell_{m_{\widetilde{d}_L}}$	$\ell_{m_{\widetilde{u}_L}}$	$\ell_{\bar{m}}$	σ_n^2
$\widetilde{d}_L\widetilde{d}_L$						
$\widetilde{d}_L\widetilde{u}_L$						

1.2 Comparison with Prospino and NLL-fast

In this section plots of the relative deviance distributions defined in Sec. **??** are shown for the squark pair-production cross sections from the MSSM-24 and CMSSM datasets. Cross sections predicted by the DGP are compared to cross sections calculated using `Prospino`, and `NLL-fast` where this is possible.

The settings used in this chapter are

- 10 GP experts with 8000 or 5000 training points each
- Features $m_{\widetilde{g}}, m_{\widetilde{q}_i}, \bar{m}$ $(m_{\widetilde{g}}, m_{\widetilde{q}_i}, m_{\widetilde{q}_j}, \bar{m})$ for $\widetilde{q}_i\widetilde{q}_j$ where $i = j$ $(i \neq j)$[2]
- The Matérn kernel with $\nu = 1.5$ and a white noise term

$$\begin{aligned} k(\mathbf{x}_i, \mathbf{x}_j) =& \sigma_f^2 \exp\left(1 + \sqrt{3}\big[(\mathbf{x}_i - \mathbf{x}_j)^T M (\mathbf{x}_i - \mathbf{x}_j)\big]^{1/2}\right) \\ & \times \exp\left(\sqrt{3}\big[(\mathbf{x}_i - \mathbf{x}_j)^T M (\mathbf{x}_i - \mathbf{x}_j)\big]^{1/2}\right) + \sigma_n^2 \delta_{ij}, \end{aligned} \tag{1.1}$$

  where $M = \mathrm{diag}(\boldsymbol{\ell})^{-2}$.
- A lower cut on the cross sections $\sigma > \sigma_{cut} = 10^{-16}$ fb

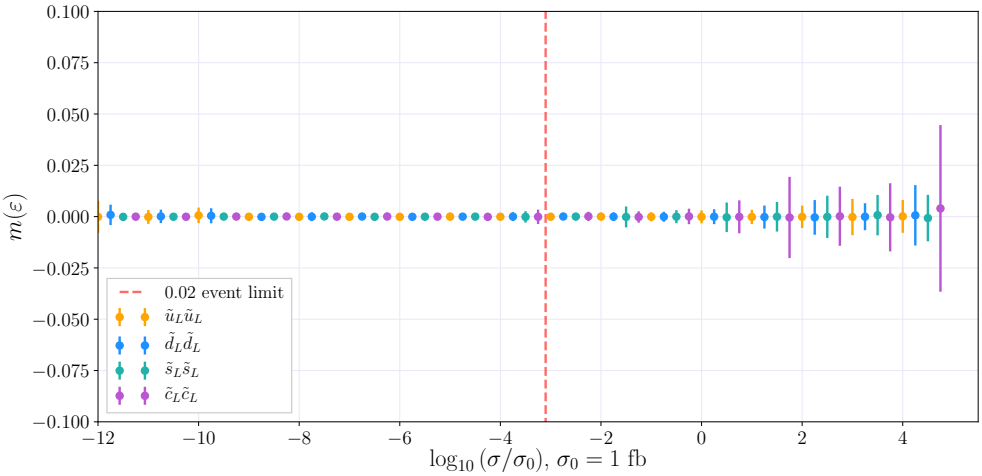
All first- and second generation squarks are considered; $m_{\widetilde{u}_L}$, $m_{\widetilde{d}_L}$, $m_{\widetilde{s}_L}$, $m_{\widetilde{c}_L}$, $m_{\widetilde{u}_R}$, $m_{\widetilde{d}_R}$, $m_{\widetilde{s}_R}$ and $m_{\widetilde{c}_R}$. These make up 36 different processes for squark pair production. A separate distributed Gaussian processes estimator is trained for each squark process, resulting in $36 \times 10 = 360$ trained experts. The optimized kernel parameters from a single GP with 8000 training points for $\widetilde{d}_L\widetilde{d}_L$ and $\widetilde{d}_L\widetilde{u}_L$ are shown in Tab. ().

1.2.1 Relative Deviance

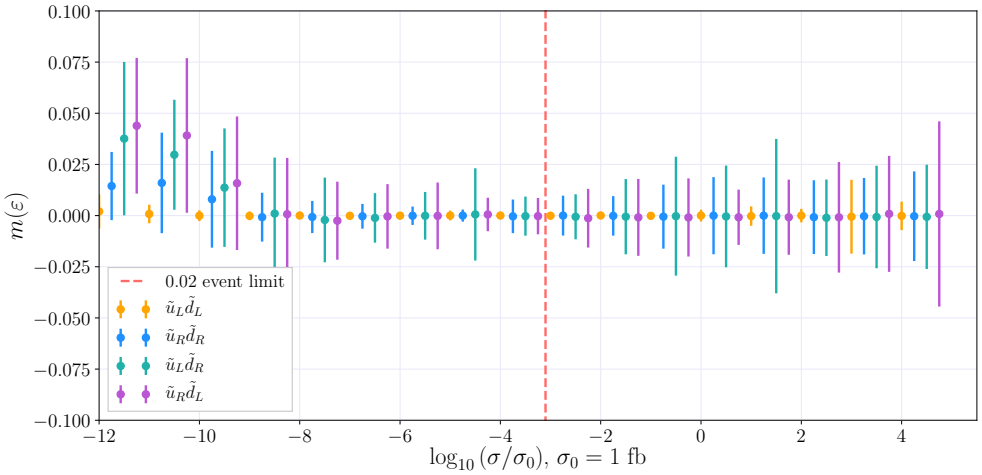
MSSM-24

Relative deviance distributions for 20 000 test points from the MSSM-24 data set are shown in Fig. 1.3, for selected squark processes. In Fig. 1.1a the relative deviance distributions are shown for all processes with equal-flavour left-handed squarks. These processes have only 3 features, as opposed to 4, which appear

[2]Same flavour quarks with different chiralities, *e.g.* $\widetilde{d}_L\widetilde{d}_R$, are regarded as different $i \neq j$, because $m_{\widetilde{d}_R} \neq m_{\widetilde{d}_L}$.



(a) The processes $\widetilde{u}_L\widetilde{u}_L$, $\widetilde{d}_L\widetilde{d}_L$, $\widetilde{s}_L\widetilde{s}_L$ and $\widetilde{c}_L\widetilde{c}_L$.



(b) The processes $\widetilde{u}_L\widetilde{d}_L$, $\widetilde{u}_R\widetilde{d}_R$, $\widetilde{u}_L\widetilde{d}_R$ and $\widetilde{u}_R\widetilde{d}_L$.

Figure 1.2: Relative deviance distributions as a function of the logarithm of the normalized cross sections $\log_{10}\sigma/\sigma_0$. Ten experts with 8000 training points from the MSSM-24 dataset each were combined on 20 000 test points from the MSSM-24 dataset for processes **(a)** $\widetilde{u}_L\widetilde{u}_L$, $\widetilde{d}_L\widetilde{d}_L$, $\widetilde{s}_L\widetilde{s}_L$ and $\widetilde{c}_L\widetilde{c}_L$; and **(b)** $\widetilde{u}_L\widetilde{d}_L$, $\widetilde{u}_R\widetilde{d}_R$, $\widetilde{u}_L\widetilde{d}_R$ and $\widetilde{u}_R\widetilde{d}_L$.

to make them easier to learn for the DGP. The DGP estimators for the corresponding right-handed processes are almost identical as functions of $m_{\widetilde{q}_R}$. This is because `Prospino` calculates cross sections for strong interactions, and the NLO terms only contain QCD corrections, as discussed in Sec. **??**. Since there is no electroweak correction, there is no distinction between left- and right-handed squarks in the calculation. They also share the same underlying pdf from the quark, *e.g.* the pdf's used for calculating $\widetilde{d}_R\widetilde{d}_R$ and $\widetilde{d}_L\widetilde{d}_L$ is the one for the d-flavour squark.

The prediction for equal-flavour equal-chirality squarks in Fig. 1.7a is very stable and close to the true values. All relative deviance distributions have a mean of approximately zero, and a standard deviation well within the desired value of 10%. The largest cross sections have been excluded from the plots, as there are few training and test points there, and so the prediction has a very large uncertainty.

In Fig. 1.7b the relative deviance distributions are shown for the process $\widetilde{u}\widetilde{d}$ for different chirality combinations. The expression for the cross section depends on the chirality combinations, as discussed in Sec. **??**. The prediction for $\widetilde{d}_L\widetilde{u}_L$ is superior to the other chirality combinations, possibly because of the previous argument of the masses $m_{\widetilde{d}_L}$ and $m_{\widetilde{u}_L}$ being very strongly correlated, as discussed in Sec. **??**. The prediction for $\widetilde{d}_R\widetilde{u}_R$ is also better than the processes with different chiralities, but not as good as $\widetilde{u}_L\widetilde{d}_L$. All processes in Fig. 1.7b have 4 features, which seems to make the function more difficult to predict. The mean values of the relative deviance distributions are still close to zero, particularly for cross sections above the 0.02 event limit. Standard deviations are larger than for equal squark processes, but well within 5% for all cross sections larger than $> 10^{-8}$ fb.

CMSSM

The DGPs trained on MSSM-24 data are also tested on an CMSSM data set and compared to cross sections calculated by `NLL-fast` for the same parameter points. The true values are the sums of cross sections for all 36 processes for each parameter point, calculated by `Prospino`. The DGPs estimate the cross sections for each of the 36 processes as well, and these ase summed for each parameter point.

The resulting relative deviance distributions are shown in Fig. 1.3. The cross sections calculated by `NLL-fast` are quite close to the true values for the CMSSM data. This is because the squark masses in CMSSM have much smaller splittings than in MSSM-24, as discussed in Sec. **??**, and `NLL-fast` assumes degenerate squark masses. For large cross sections, however, `NLL-fast` predicts values that are too large, while the DGPs predict cross sections very close to the true value.

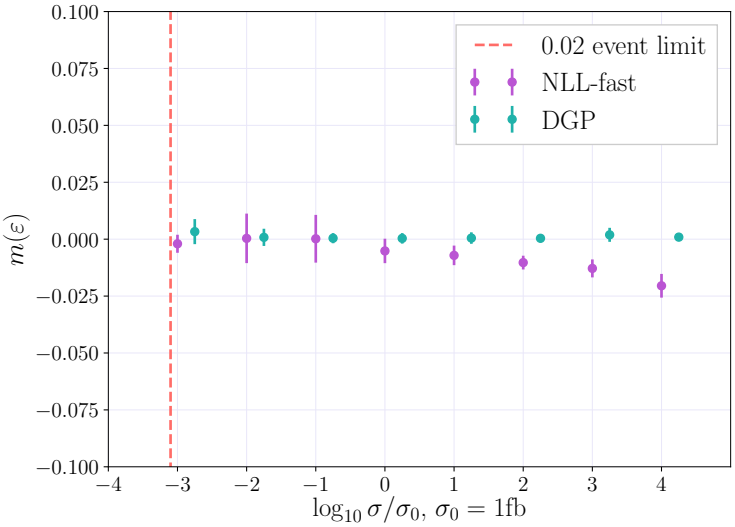


Figure 1.3: Distribution of the relative deviance ε as a function of the logarithm of the total cross section for all squark pair production processes for NLL-fast and Gaussian processes for mSUGRA data. The 'true' values are the values generated by Prospino. 10 experts with 5000 training points each were trained on MSSM-24 data for each process $\widetilde{q}_i\widetilde{q}_j$.

1.2.2 Cross Sections

Cross sections estimated by the distributed Gaussian processes are compared to cross sections from `Prospino` and `NLL-fast`. The estimators consist of 10 experts with 5000 training points each. Note that these experts are smaller than those used in Sec. 1.2.1, where each expert used 8000 training points. This is due to the size of saved models, which will be discussed in Sec. 1.3. In Fig. 1.4 the cross sections for the processes $\widetilde{d}_L\widetilde{d}_L$ and $\widetilde{d}_L\widetilde{u}_L$ are shown as a function of $m_{\widetilde{d}_L}$ for $m_{\widetilde{d}_L}, m_{\widetilde{u}_L} \in [200, 2500]$ GeV, with the approximate mass splitting

$$m^2_{\widetilde{d}_L} - m^2_{\widetilde{u}_L} \approx m^2_W, \tag{1.2}$$

for $m_W = 80$ GeV. All other squark masses are held at 1000 GeV, and the gluino mass is $m_{\widetilde{g}} = 500$ GeV. The DGP prediction is very close to the cross sections from `Prospino`, with a sligthly lower prediction for large $m_{\widetilde{d}_L}$. In addition, the uncertainty in the DGP prediction is very small, with the plots in Fig. 1.4 showing 50σ uncertainty bands.

The uncertainty in the `Prospino` calculation comes from the renormalization scale dependence. Cross sections are calculated for twice the renormalization scale, and half the renormalization scale, to see how scale dependent the cross sections are. As discussed in Sec. **??**, the scale dependence is reduced with the addition of higher order terms to the cross section, as a consequence this is also

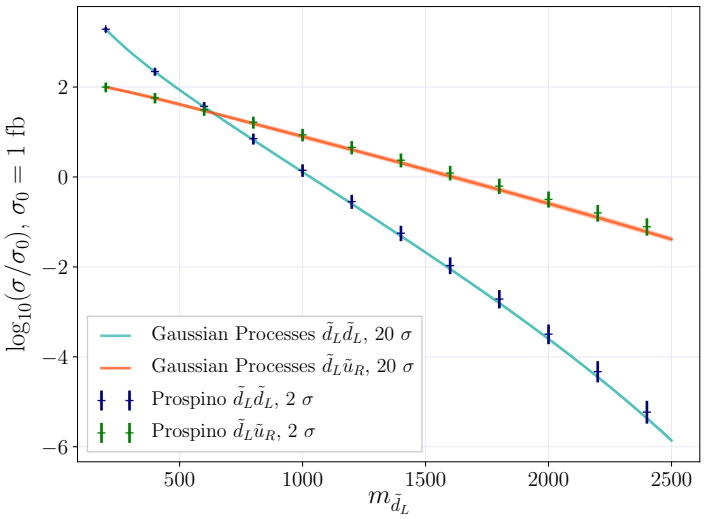


Figure 1.4: Cross sections for $\widetilde{d}_L\widetilde{d}_L$ and $\widetilde{u}_L\widetilde{d}_L$, using $m_{\widetilde{d}_L} = [200, 2400]$,GeV and all other masses set to $m_i = 1000$ GeV generated by prospino (crosses) and predicted by the GP (lines with errors). The GP models used are for $\widetilde{d}_L\widetilde{d}_L$ and $\widetilde{d}_L\widetilde{u}_L$.

a way of estimating the order of magnitude of higher order terms (in this case, next-to-next-to-leading order).

The cross sections calculated by `NLL-fast` for the MSSM-24 are very far from the true values, and only coincide with `Prospino` for $m_{\widetilde{d}_L} = 1000$ GeV, where the squark masses are in fact degenerate. The uncertainty from `NLL-fast` includes the uncertainty from scale dependence, from the pdf's and from α_s.

Cross sections are also calculated as a function of the gluino mass, $m_{\widetilde{g}}$, and shown in Fig. 1.6. All squark masses are here held at 1000 GeV for gluino mass $m_{\widetilde{g}} \in [200, 2400]$ GeV. The uncertainty from `Prospino` is not shown here. The DGP predicted cross sections and cross sections from `Prospino` and `NLL-fast` all coincide, and the DGP gives an uncertainty that is fairly small, but increases with increasing gluino mass.

1.3 Optimizing the Model

The distributed Gaussian process predictions with 10 experts using 8000 training points each are very accurate. Unfortunately, the Gaussian process models take up a lot of space when stored. In addition, larger models take longer to predict values. In this section the model sizes and computation times are discussed, to find the optimal model as a compromise between prediction quality, size and computation time.

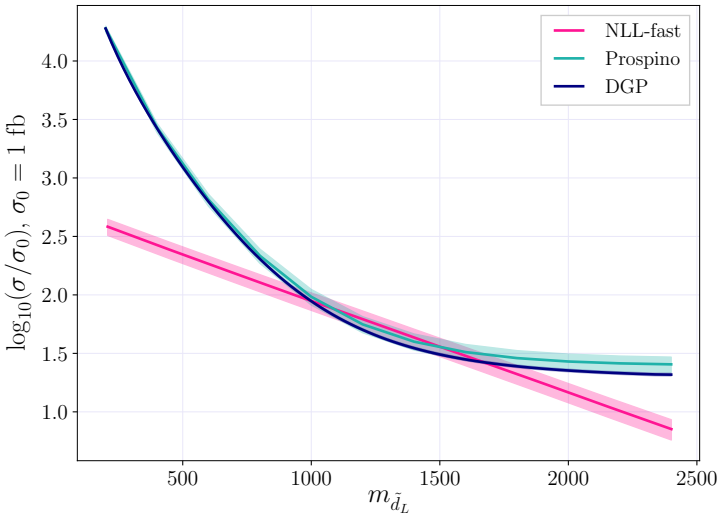


Figure 1.5: Total cross sections for all 36 processes as a function of $m_{\widetilde{d}_L}$ calculated by Prospino and NLL-fast, and estimated by DGP. The masses $m_{\widetilde{d}_L}$ and $m_{\widetilde{u}_L}$ are varied from $[200, 2400]$ GeV while all other squark masses are held at 1000 GeV, and $m_{\widetilde{g}} = 500$ GeV.

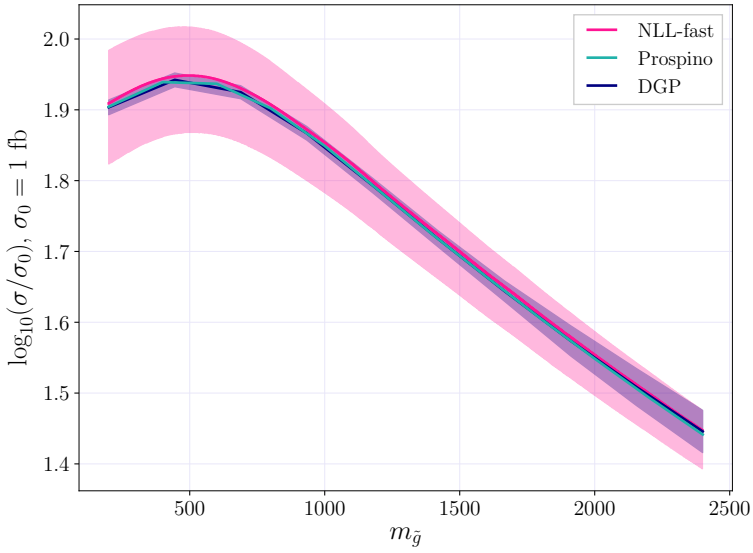


Figure 1.6: Total cross sections for all 36 processes as a function of $m_{\widetilde{g}}$ calculated by Prospino and NLL-fast, and estimated by Gaussian processes. The mass m_g is varied $[200, 2400]$ GeV while all squark masses are held at 1000 GeV.

Training data size	Model size $[MB]$
2000	31
3000	69
5000	191

Table 1.1: Size of saved GP models with 3 or 4 features and the Matérn kernel.

Model Size

Storing the distributed Gaussian processes models requires a lot of memory. A single Gaussian process model trained with 2000 training points takes up 31 MB. Since one model is needed per process, this means that a relatively small model with a single expert with 2000 training points will require

$$36 \times 31 \text{ MB} \approx 1.09 \text{ GB}. \tag{1.3}$$

As seen from Table 1.1 the model size scales approximately as $\mathcal{O}(n^2)$ for a model with n training points. For M experts with n training points each, the model size then scales as $\mathcal{O}(M \cdot n^2)$. Where it is possible it is therefore advisable to add *more* experts as opposed to *larger* experts.

The model size puts limits on the number of training points an optimal model should have. There is very little difference in model sizes for models with 3 and 4 features, and a difference of 29 bytes for all training sizes between using the RBF kernel and the Matérn kernel. A model with 36 processes with 10 experts each would require

$$10 \times 36 \times 191 \text{ MB} = 67.15 \text{ GB}.$$

By comparison, reducing the experts to 3000 training points each would require

$$10 \times 36 \times 69 \text{ MB} = 24.26 \text{ GB}.$$

Comparisons of the relative deviance distributions with 3000, 5000 and 8000 training points per expert for the processes $\widetilde{d}_L\widetilde{d}_L$ and $\widetilde{d}_L\widetilde{u}_R$ are shown in Fig. 1.7. The improvement in the prediction for $\widetilde{d}_L\widetilde{u}_R$ from 3000 to 8000 training points per expert is large for all σ. Processes with different squarks and chiralities should therefore use experts with as many training points as possible, as the main contribution to the prediction error will come from these processes. In contrast, the process $\widetilde{d}_L\widetilde{d}_L$ is modelled very accurately even for 3000 training points per expert. For cross sections larger than 10^2 fb there is a visible improvement from 3000 to 8000 training points per expert, likely rooted in the addition of data[3] in the form of larger experts. For 3000 training points per expert this

[3]The endpoints are usually not well covered

could be remedied simply by adding more experts trained on large cross sections. In conclusion, processes with equal flavour and equal chirality can use smaller experts, while processes with different flavour or different chirality benefit from having as large experts as possible.

Number of Models

In this project 36 Gaussian process models were trained with 10 experts each, resulting in 360 experts. The effective number of models may be reduced, however, as equal-flavour equal-chirality (EFEC) processes are identical as functions of the squark mass, as shown in Sec. **??**. The numer of models can therefore be reduced from 36 to 32.

In addition, the models for EFEC processes, and $\widetilde{d}_L\widetilde{u}_L$ and $\widetilde{c}_L\widetilde{s}_L$ perform better than the other processes for fewer training points. It may therefore not be necessary to use 10 experts for each of these processes. In Fig. 1.1a the learning curve for $\widetilde{d}_L\widetilde{d}_L$ with 2000 training points per expert shows little difference in validation score for more than seven experts. For larger experts it could therefore be sufficient with *e.g.* 4 experts. Reducing the number of experts for all EFEC processes, and $\widetilde{d}_L\widetilde{u}_L$ and $\widetilde{c}_L\widetilde{s}_L$, and using the equal-flavour left-handed models on equal-flavour right-handed processes reduces the number of experts to

$$4 \times 4 \text{ experts} + 2 \times 4 \text{ experts} + 26 \times 10 \text{ experts} = 280 \text{ experts}.$$

Computation Times

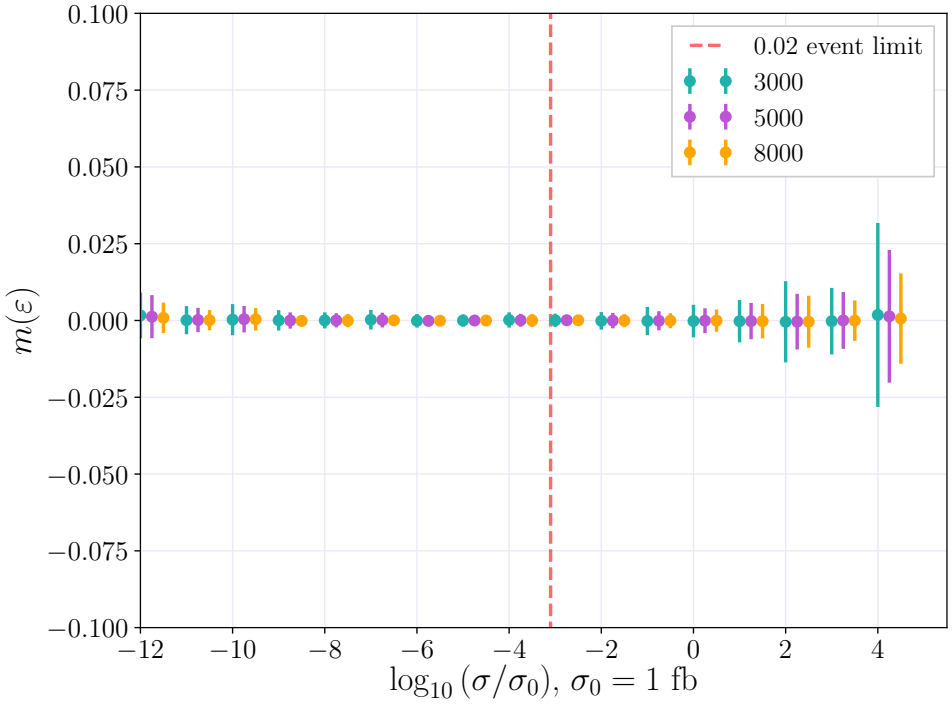
Training 10 experts with 5000 training points each, in parallel, takes approximately 40 minutes for a single process. The prediction time for the DGP was calculated by letting the model predict values for 2000 test points for a single process, and dividing the total computation time by 2000. The average computation time for predicting a single point for a single process is

$$0.46925 \text{ s}. \tag{1.4}$$

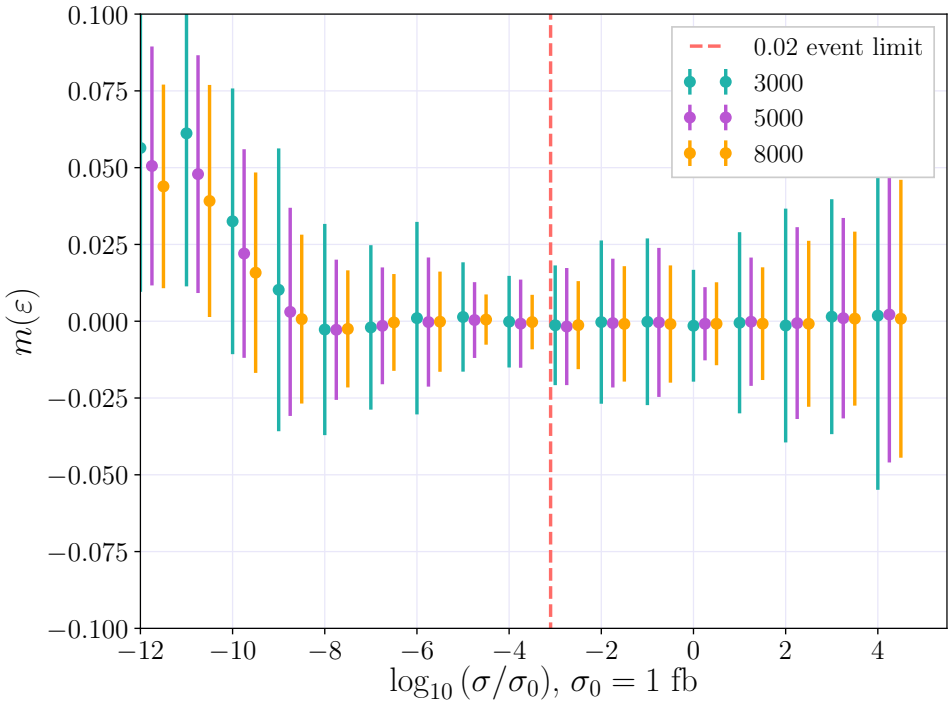
The computation time for predicting all 36 cross sections for a single parameter point is therefore

$$36 \times 0.46925 \text{ s} = 16.893 \text{ s}. \tag{1.5}$$

In Tab. 1.2 the prediction times for all 36 process cross sections for a single parameter point are shown for `Prospino` and distributed Gaussian processes. For `Prospino` three cross sections were calculated for each process, with scale factors 0.5, 1.0 and 2.0, to include the uncertainty from scale dependence. The time for `NLL-fast` is also shown for one parameter point. `NLL-fast` is considerably faster than both DGP and `Prospino`, but has a very large error relative to the



(a) The process $\widetilde{d}_L\widetilde{d}_L$.



(b) The process $\widetilde{d}_L\widetilde{u}_R$.

Figure 1.7: Relative deviance distributions as a function of the logarithm of the normalized cross sections $\log_{10}\sigma/\sigma_0$. Ten experts with 3000 (green), 5000 (violet) and 8000 (orange) training points from the MSSM-24 dataset each were combined on 20 000 test points from the MSSM-24 dataset for processes **(a)** $\widetilde{d}_L\widetilde{d}_L$ and **(b)** $\widetilde{d}_L\widetilde{u}_R$.

Tool	Computation time $[s]$
Prospino	1711.96
NLL-fast	0.00739
Distributed Gaussian Processes	16.893

Table 1.2: Computation times for 1 parameter point for all 36 squark pair production processes.

`Prospino` computation, as seen in Fig. 1.5. Although it is much slower than `NLL-fast`, the DGP is faster than `Prospino` by a factor of approximately 61.

The prediction of each DGP expert is *not* done in parallel. Parallelising the prediction algorithm could reduce the computational time in Tab. 1.2 by a factor of approximately 10

$$16.893 \text{ s} : 10 = 1.689 \text{ s}.$$

The prediction of each of the 36 squark production processes was also done in sequence. Predicting for each process in parallel could further reduce the computation time by a factor of 36

$$1.689 \text{ s} : 36 = 0.0469 \text{ s}.$$

Note that these are idealized times, meant to illustrate how the current algorithm can be improved. The prediction time of Gaussian processes goes as $\mathcal{O}(n^2)$ for n training points, so using smaller experts would also reduce the computation time.