Investigation for Enhancing Rare Top-Antitop Quark Background for Higgs-Pair Production and its Identification using Machine Learning

Untersuchung zur Anhäufung seltener Top-Antitop Quark Ereignisse im Hintergrund zur Higgs-Paar Erzeugung und ihre Identifizierung durch Benutzung von maschinellen Lernen

Masterarbeit der Fakultät für Physik
der
Ludwig-Maximilians-Universität München

vorgelegt von
Tim Marvin Rexrodt
geboren in Kassel

München, den 20.01.2026

Contents

Chapter 1

Basics

As Higgs bosons have a neutral charge, as well as a spin of 0, because of CPT symmetry, a Higgs boson pair emerging is equivalent to two Higgs bosons colliding with time reversed. As such, this can be used to better understand Higgs bosons.

Proton-Proton collisions can produce Higgs boson pairs. The most likely way for this to happen would be a gluon fusion, with a cross section of 36.7±5.8fb at 14TeV (12). Higgs bosons have a 25.7±2.5% of branching into W^+W^- and a 53±8% of branching into $b\bar{b}$ (14). Combined, this results in a 27±5% chance for Higgs pairs to decay into $W^+W^-b\bar{b}$ with an overall 10.0±1.7fb cross section times branching ratio with gaussian error calculation.

Also, as a Higgs boson has an expected invariant mass of 125.2GeV (14), while a W boson has an expected invariant mass of 80.4GeV (14), for a Higgs boson to decay into two W bosons, at least one of the W-bosons has to be off shell.

However, a much more common result of proton-proton collisions with a cross section of 953.6±38.3pb at 14TeV is a top-antitop pair (7). Top quarks generally decay into a W-boson and another quark, which in 95.7±3.4% of cases is a bottom quark (14), which results in an overall cross section times branching ratio of 873.4±5.7pb with gaussian error calculation.

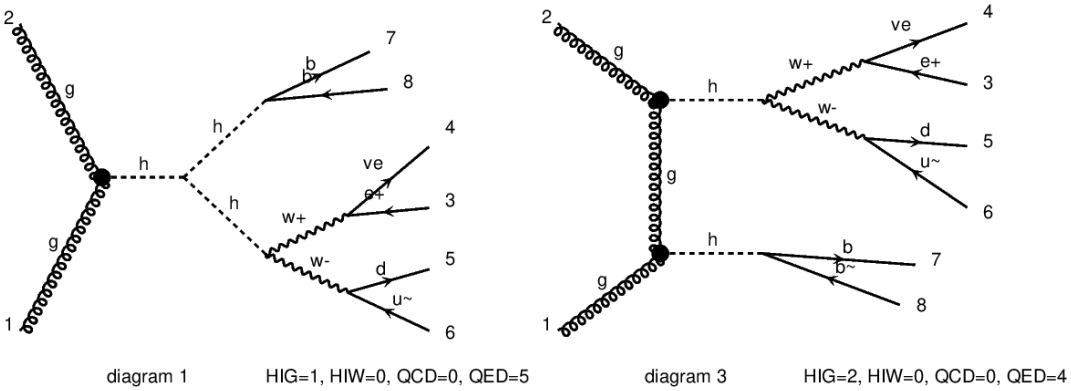
Following this, the ratio of their events is roughly 87.3k $t\bar{t} \rightarrow W^+bW^-\bar{b}$ events for 1 $HH \rightarrow W^+W^-\bar{b}b$ event, so even the fringes of uncertainty for $t\bar{t}$ can easily overshadow all HH-events.

Chapter 2

Dataset Analysis

2.1 Datasets

For HH-pair production events, MadGraph 5 (1) simulates a gluon-gluon fusion resulting in $pp \to HH \to W^+W^-b\bar{b} \to q\bar{q}l\nu_l b\bar{b}$ with both $W^+ \to \bar{l}\nu_l, W^- \to \bar{q}q$ and $W^- \to l\bar{\nu}_l, W^+ \to \bar{q}q$ considered and q, $\bar{q}$ and l being first or second generation particles. The cross section times branching ratio of this is given by MadGraph as 253.3±0.13ab at $\sqrt{s} = 14TeV$.



(a) (b)

Figure 2.1: Examples of the Feynman diagrams created for the HH-pair production by MadGraph. Only the first generation quarks and leptons are displayed, e can also be μ, d can also be s and u can also be c.

For $t\bar{t}$-pair production events, MCatNLO (1) only considers the $W^- \to l\bar{\nu}_l, W^+ \to \bar{q}q$ case, however, for simplicity, the other case can be assumed to be symmetrical and as such be used to represent both, as long as the charge of the W bosons is not used to differentiate between HH-events and $t\bar{t}$-events. MadGraph calculates for the $t\bar{t}$-events a cross section times branching ratio of 209.7±0.17pb. However, as there is a similar process with a cross section times branching ratio of 246.1±0.71pb, where an additional gluon is released somewhere in between with a similar cross section, the functional cross section times branching ratio can be combined to 455.8±0.74pb.

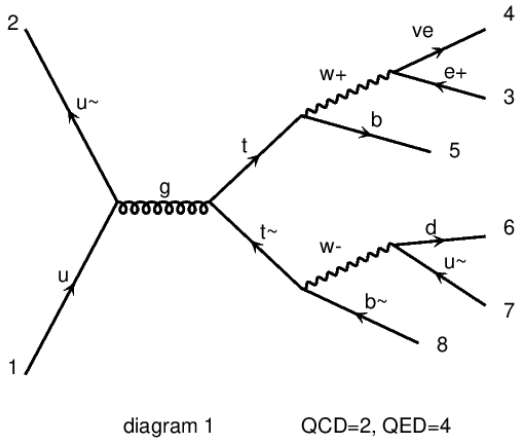


Figure 2.2: Example of the Feynman diagrams created for the $t\bar{t}$-pair production by MadGraph. Only the first generation quarks and leptons are displayed, e can also be μ, d can also be s and u can also be c.

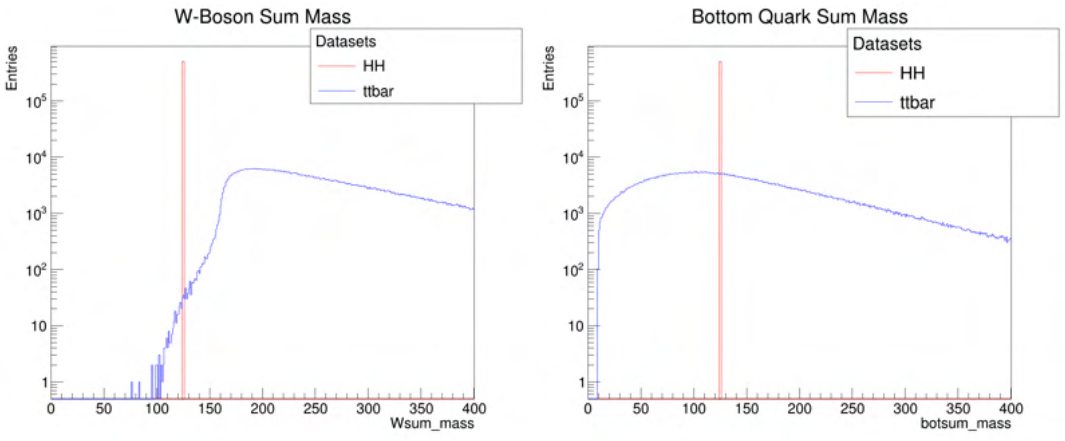
This puts the expected ratio between the events at roughly 1 HH-event to 1.8 million $t\bar{t}$-events at $\sqrt{s} = 14TeV$. This is more than a magnitude bigger than the ratio calculated from outside sources in Chapter 1.

After generating the wanted decay chains into $b\bar{b}q\bar{q}l\nu$, the events are given to Pythia 6.4 (16), which models the further decay into particles for the jet reconstruction. This happens with Final State Interactions and Multiparton Interactions being deactivated.

2.2 Target

As shown in Chapter 1 and Section 2.1, $t\bar{t}$-events appear much more often than HH-events, so events with parameters, that are already much more common among $t\bar{t}$-events, can for simplicity be assumed to be $t\bar{t}$.

The first parameters looked at for this are the equivalents to the Higgs boson masses in the HH-dataset: the $b\bar{b}$ pair mass and W^+W^- pair mass, respectively.



(a) W^+W^- pair mass (b) $b\bar{b}$ pair mass

Figure 2.3: Higgs boson mass equivalent comparison, with 1M $t\bar{t}$-events (blue) and 1M HH-events (red), with logarithmic scaling

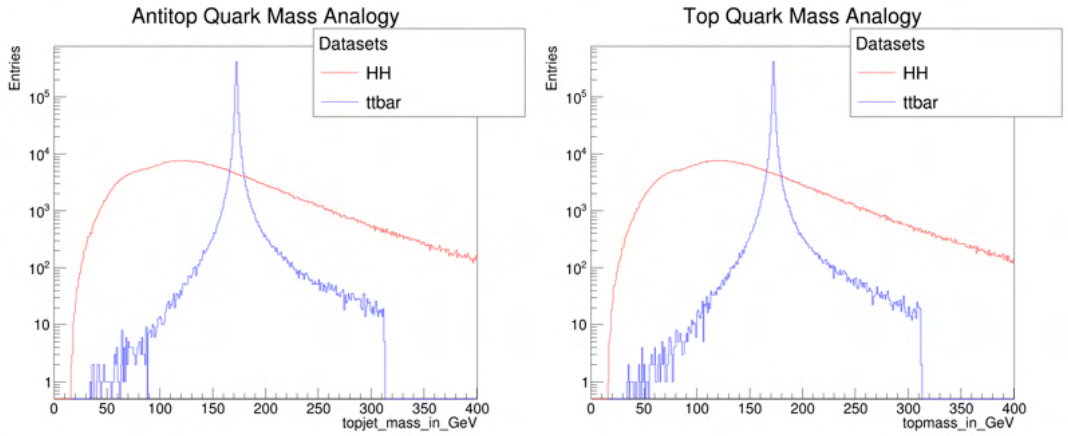
As Figure 2.3 demonstrates, the Higgs boson mass is $125GeV$ with a small width, as such, and to leave some space for smearing, the first conditions for the targets are set as

$$B : 100GeV \leq m_{b\bar{b}} \leq 150GeV \tag{2.1}$$

and

$$W : 100GeV \leq m_{W^+W^-} \leq 150GeV \tag{2.2}$$

The same goes for the equivalents of the top quark mass and antitop quark mass, which are the W^+b pair mass, and the $W^-\bar{b}$ pair mass, respectively.



(a) $W^-\bar{b}$ pair mass (b) W^+b pair mass

Figure 2.4: Comparison of the Wb pair masses, with 1M $t\bar{t}$-events (blue) and 1M HH-events (red), with logarithmic scaling

Here, events with a mass between $150GeV$ and $195GeV$ are excluded for both top and antitop quarks. Because of the symmetry of both cases, they are combined into one condition:

$$T : \overline{(150GeV \leq m_{W^+b} \leq 195GeV)} \wedge \overline{(150GeV \leq m_{W^-\bar{b}} \leq 195GeV)} \tag{2.3}$$

2.3 Increasing the Number of Events on Target

After applying those three restrictions, only a small amount of events are left over, in fact, of a dataset of 5M $t\bar{t}$-events, only 1 event passes all three conditions TBW.

	All	T	B	W
Abs	5,000,000	1,224	1,241,493	11,932
Rel	100.00000%	0.02448%	24.82986%	0.23864%
	TBW	TB	TW	BW
Abs	1	263	8	2,399
Rel	0.00002%	0.00526%	0.00016%	0.04798%

Table 2.1: Number of events meeting the target conditions out of 5,000,000 $t\bar{t}$-events

As Table 2.1 shows, the biggest reason for the low pass rate are conditions T and W, while roughly a quarter of the events pass condition B, so, to get a useful amount of events with these conditions, making more events pass conditions T and W should be a priority.

As comparison, here are the same conditions TBW applied to a HH-dataset:

	All	T	B	W
Abs	998,162	614,090	998,162	998,162
Rel	100.00%	61.52%	100.00%	100.00%
	TBW	TB	TW	BW
Abs	614,090	614,090	614,090	998,162
Rel	61.52%	61.52%	61.52%	100.00%

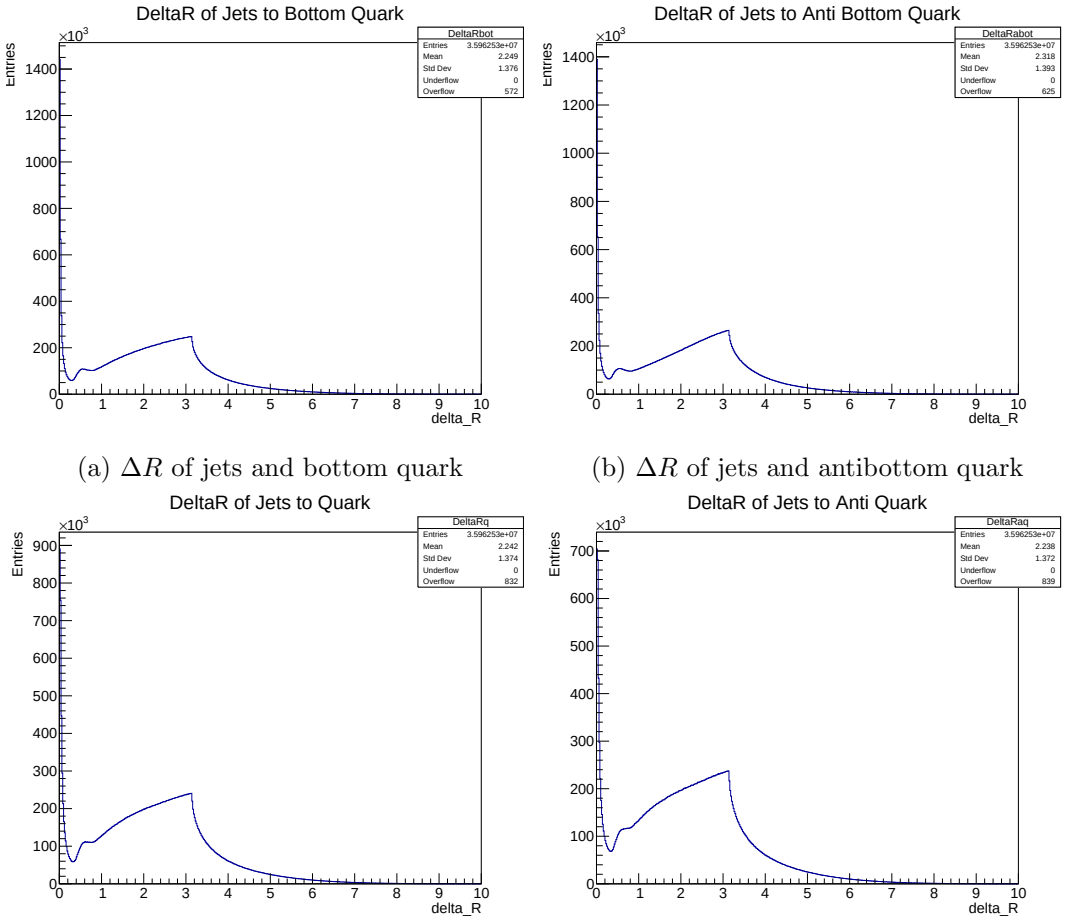
Table 2.2: Number of events meeting the target conditions out of 998,162 HH-events

As Table 2.2 shows, all HH-events pass conditions B and W, and more than half pass condition T, much more than $t\bar{t}$-events.

2.3.1 Jet Reconstruction

Using an anti-kt jet reconstruction with the minimum jet energy of $Ptmin = 5GeV$ and $\Delta R = 0.4$ for removing overlaps between leptons and jets, with FastJet 3.3.4. (3)

Then the Jets get matched to the truth particles of the bottom quarks $(b, \bar{b})$, as well as the decay products from one of the W bosons, $(q, \bar{q})$. The decay products of the other W-boson (ν_l, l) are ignored in the jet reconstruction, as l can be directly detected by a detector and ν_l cannot be reliably detected and needs to be reconstructed by energy from all visible particles. In both cases, the data of the truth particle is used directly instead.



(a) ΔR of jets and bottom quark

(b) ΔR of jets and antibottom quark

(c) ΔR of jets and quark from W boson

(d) ΔR of jets and antiquark from W boson

Figure 2.5: The ΔR between the respective truth particle and all jets of their event, for 4M $t\bar{t}$-events

Looking at the ΔR between the particles to be matched and all jets, $\Delta R < 0.5$ seems like a sensible cutoff point.

The matching of jets to truth particles works as followed:

A jet is matched to the particle with its smallest ΔR, ignoring all particles that are already matched to a jet with a smaller ΔR to it. If another jet is already matched to that particle, this displaced jet repeats this process with the rest of the particles. This then repeats, until either one of the displaced jets finds no particle, it has a smaller ΔR with, or all four particles have a new jet. Then the entire process repeats with the next jet, until all jets have gone through this.

Step	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Jet		new 1		new 2		new 3	old ab 1	old b 2		new 4	old b 1		new 5	old aq 2		new 6.	
ab	0.5	0.342	0.5	0.493	0.493	0.784	0.342	X	0.342	0.205	X	0.205	0.931	0.493	0.205	1.804	0.205
b	0.5	0.322	0.322	0.891	0.322	0.234	X	X	0.234	0.818	0.322	0.234	1.109	0.891	0.234	1.999	0.234
q	0.5	1.939	0.5	1.317	0.5	1.877	1.939	1.317	0.5	1.931	1.939	0.5	0.915	1.317	0.5	0.021	0.021
aq	0.5	1.050	0.5	0.496	0.5	1.022	1.050	0.496	0.496	1.114	1.050	0.496	0.027	X	0.027	0.882	0.027

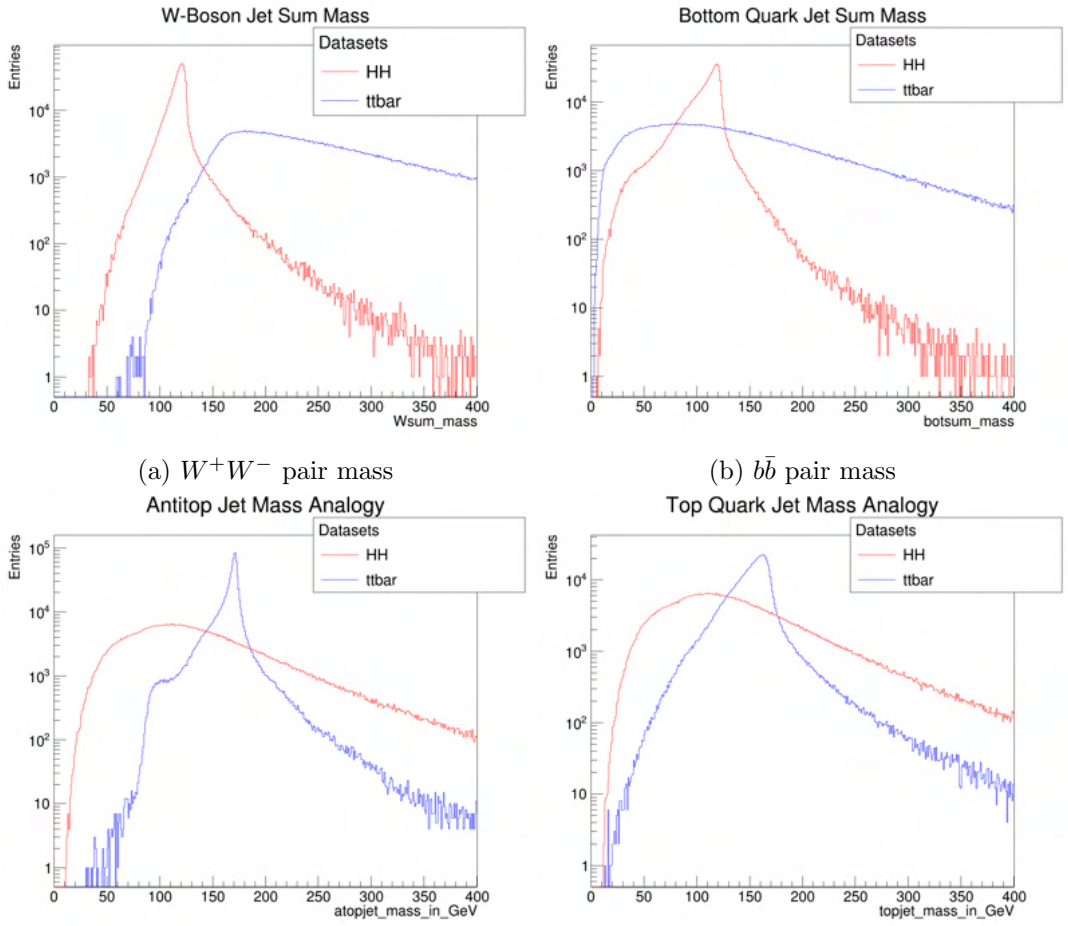
Figure 2.6: Example of the matching algorithm in action on an event. The numbers are all ΔR values. The black columns show the current ΔR threshold for a new jet to be matched, the columns between them show the ΔR of the jets to be matched, "new" is the new jet, "old" is a displaced jet previously matched to another truth particle. All jets, which have no $\Delta R < 0.5$, have been omitted for brevity.

Step	Description
0	no jets assigned
1	test jet #1
2	jet#1 assigned
3	test jet #2
4	jets#1,#2 assigned
5	test jet #3
6	jet #3 assigned, reassign jet #1
7	jet #1 assigned, reassign jet #2
8	jets#1,#2,#3 assigned
9	test jet #4
10	jet #4 assigned, reject jet #1
11	jets#2,#3,#4 assigned
12	test jet #5
13	jet #5 assigned, reject jet #2
14	jets#3,#4,#5 assigned
15	test jet #6
16	jets#3,#4,#5,#6 assigned

Table 2.3: Description of steps from Figure 2.6

After that, the results are checked for unmatched Jets with a smaller ΔR to a particle and a subsets of the matched particles, which would all achieve smaller ΔRs by switching their matched particles. Although the theoretical possibility of this has not been excluded, there is no event among all the tested datasets where this happens.

The result on the data is a further smearing.



(a) W^+W^- pair mass

(b) $b\bar{b}$ pair mass

(c) $W^-\bar{b}$ pair mass

(d) W^+b pair mass

Figure 2.7: Jet reconstructed mass equivalents comparison, of 1M $t\bar{t}$-events (blue) and 1M HH-events (red), with logarithmic scaling

After the reconstruction, only events, where all four truth particles are matched to jets, are considered for passing the conditions, as it is not feasible to classify missing values.

	All	T	B	W
Abs	3,718,227	275,279	848,744	151,816
Rel	74.365%	5.506%	16.975%	3.036%
	TBW	TB	TW	BW
Abs	4,543	46,259	21,053	36,205
Rel	0.091%	0.925%	0.421%	0.724%

Table 2.4: Number of events meeting the target conditions out of 5,000,000 $t\bar{t}$-events after reconstruction

As Table 2.4 shows, the reconstruction improves the number of $t\bar{t}$-events passing the conditions, with large improvements in passing condition T and condition W making up for the slight worsening of condition B.

	All	T	B	W
Abs	637,364	432,970	586,659	578,276
Rel	63.85%	43.38%	58.77%	57.93%
	TBW	TB	TW	BW
Abs	255,730	284,306	389,261	394,397
Rel	25.62%	28.48%	39.00%	39.51%

Table 2.5: Number of events meeting the target conditions out of 998,162 HH-events after reconstruction

A quarter of the HH-events are remaining, mainly, because after reconstruction, not all HH-events pass condition B and condition W.

2.3.2 Smearing

After Reconstruction, Gaussian smearing applied to the jets, as well as the antineutrino and lepton truth particles, to simulate the finite resolution of a detector. As experiments with $\sqrt{s} = 14TeV$ have not happened yet, for the resolution the equation from (5) for $\sqrt{s} = 13TeV$ is used:

$$\frac{\sigma(p_T)}{p_T} = \frac{N}{p_T} \oplus \frac{S}{\sqrt{p_T}} \oplus C \tag{2.4}$$

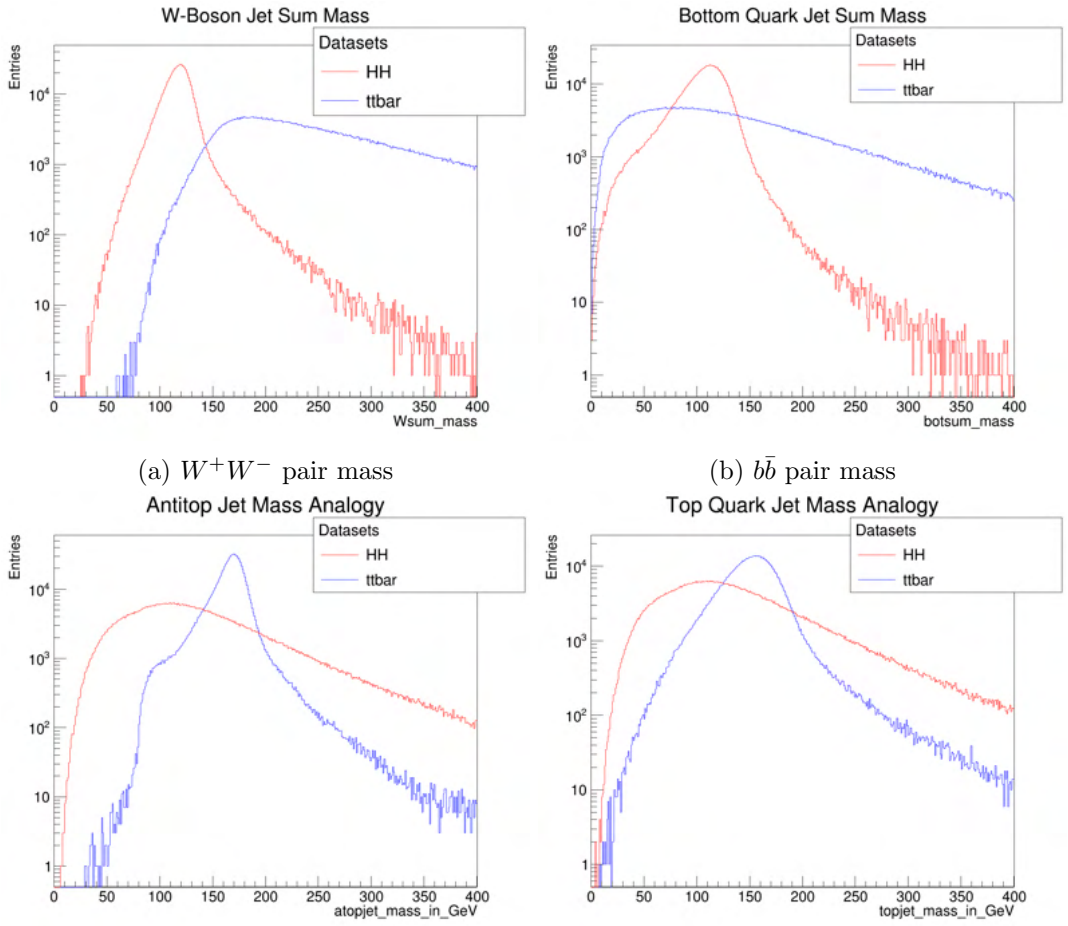
with adapted parameters $C = 0.03$, $S = 0.95$ and $N = 2.1$ (5)

the smearing equation itself looks like follows:

$$\mathbf{j}_{smear} = \mathbf{j}(1 + N(0, \sigma)) \tag{2.5}$$

Where $N(0, \sigma)$ is a normal Gaussian distribution with mean 0 and a width of σ.

Negative results in smearing are avoided, by rerolling the function, if it happens.



(a) W^+W^- pair mass

(b) $b\bar{b}$ pair mass

(c) $W^-\bar{b}$ pair mass

(d) W^+b pair mass

Figure 2.8: Jet reconstructed smeared mass equivalents comparison, with 1M $t\bar{t}$-events (blue) and 1M HH-events (red), with logarithmic scaling

	All	T	B	W
Abs	3,718,227	377,272	834,993	188,141
Rel	74.365%	7.545%	16.700%	3.763%
	TBW	TB	TW	BW
Abs	6,582	63,850	31,005	44,921
Rel	0.132%	1.277%	0.620%	0.898%

Table 2.6: Number of events meeting the target conditions out of 5,000,000 $t\bar{t}$-events, after reconstruction and smearing

This mainly further increases the number of events passing condition T, which along with a smaller improvement of passing condition W makes up for the slight decrease in passing condition B.

	All	T	B	W
Abs	637,364	433,580	536,356	550,353
Rel	63.85%	43.44%	53.73%	55.14%
	TBW	TB	TW	BW
Abs	219,423	258,164	369,132	343,210
Rel	21.98%	25.86%	36.98%	34.38%

Table 2.7: Number of events meeting the target conditions out of 998,162 HH-events, after reconstruction and smearing

A small decrease in the number of HH-events passing the target conditions T, B and W is observed.

2.3.3 Widening

To try to get more $t\bar{t}$-events to fulfill the conditions and become potential background to the HH-events, datasets with wider W boson mass distributions and top quark mass distributions are generated.

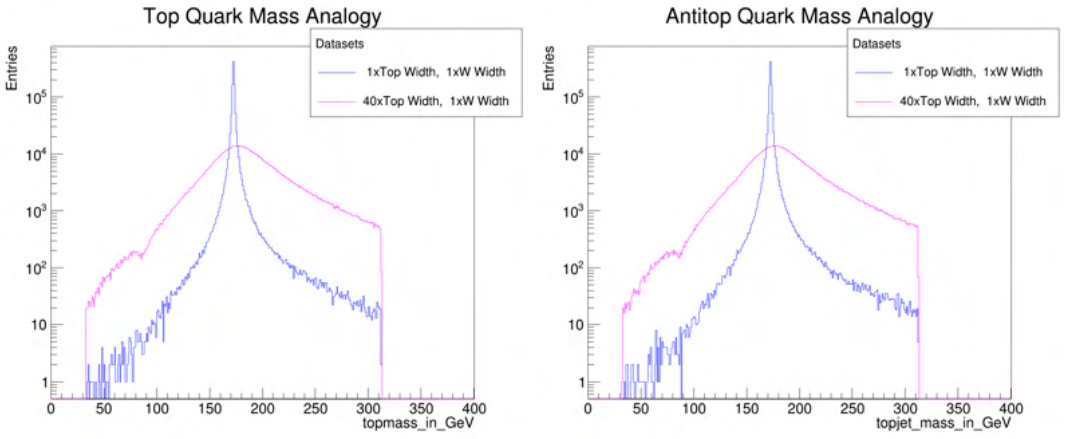
The widening of the mass distributions also changes the distributions of many of the particle energies and momenta.

The mass distributions are given by a Breit-Wigner distribution:

$$f(M, M_{peak}, \Gamma) = \frac{\Gamma \cdot M_{peak}}{\pi((M_{peak}^2 - M^2)^2 + \Gamma^2 \cdot M_{peak}^2)} \tag{2.6}$$

To widen the distribution, the value of Γ is increased.

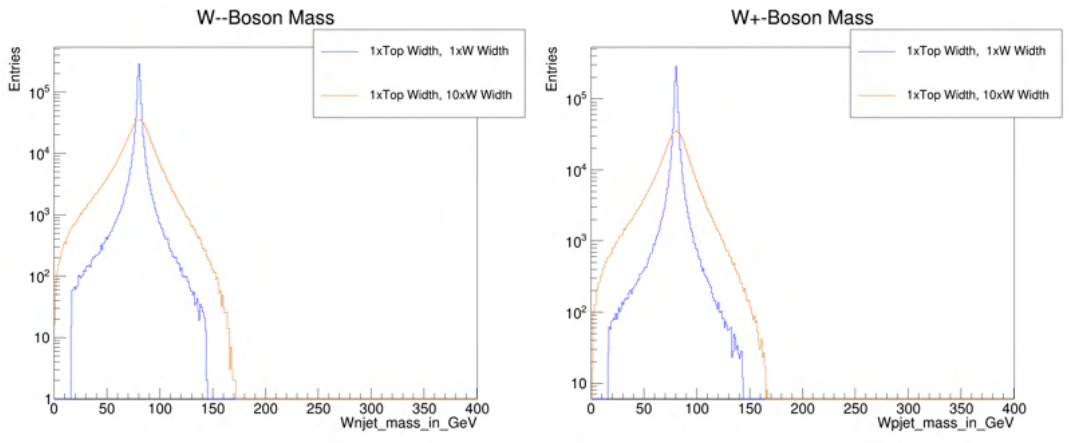
Widening the top quark mass distributions is achieved by increasing their Γ by a factor of 40 and the W boson mass distributions by increasing their Γ by a factor of 10. First separately, then simultaneously.



(a) Mass of top quark

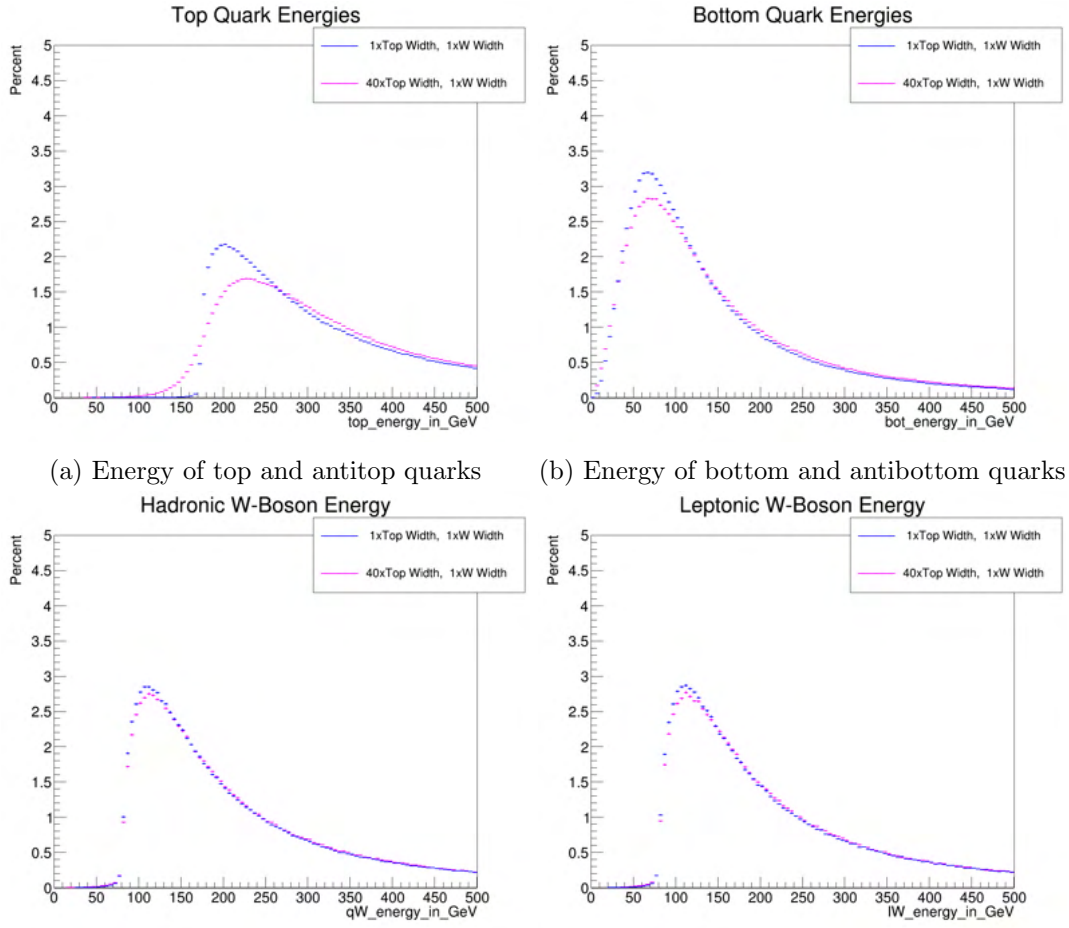
(b) Mass of antitop quark

Figure 2.9: Comparison of 1M top quark masses between the dataset with 40 times larger top quark width (magenta) and the dataset with standard top quark width (blue), with logarithmic scaling



(a) mass of W^--boson (b) mass of W^+-boson

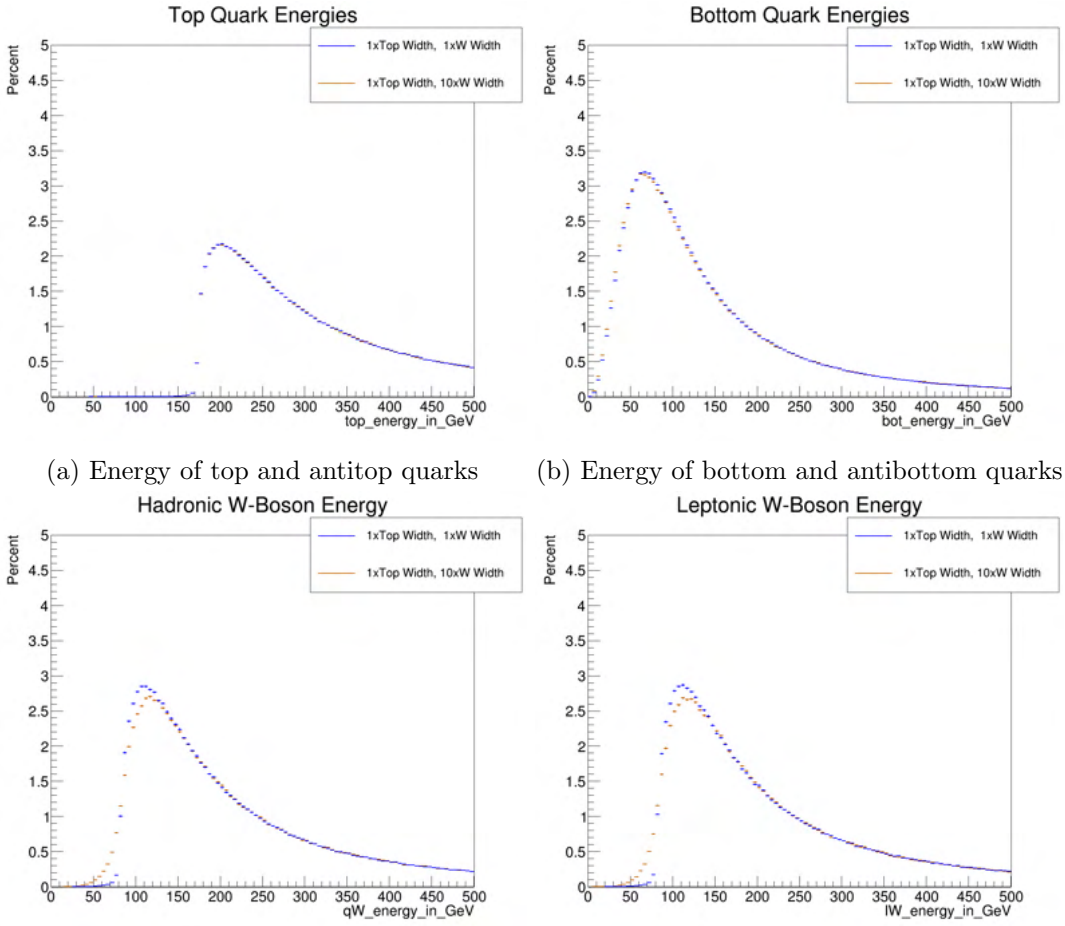
Figure 2.10: Comparison of 1M W boson masses between the dataset with 10 times larger W boson width (orange) and the dataset with standard W boson width (blue), with logarithmic scaling



(a) Energy of top and antitop quarks (b) Energy of bottom and antibottom quarks

(c) Energy of hadronic W boson (d) Energy of leptonic W boson

Figure 2.11: Effect on energy curves after increasing the Γ of top quarks by a factor of 40 with standard $t\bar{t}$-dataset (blue) and widened $t\bar{t}$-dataset (magenta), relative to the number of events



(a) Energy of top and antitop quarks

(b) Energy of bottom and antibottom quarks

(c) Energy of hadronic W boson

(d) Energy of leptonic W boson

Figure 2.12: Effect on energy curves after increasing the Γ of the W bosons by a factor of 10 with standard $t\bar{t}$-dataset (blue) and widened $t\bar{t}$-dataset (orange), relative to the number of events

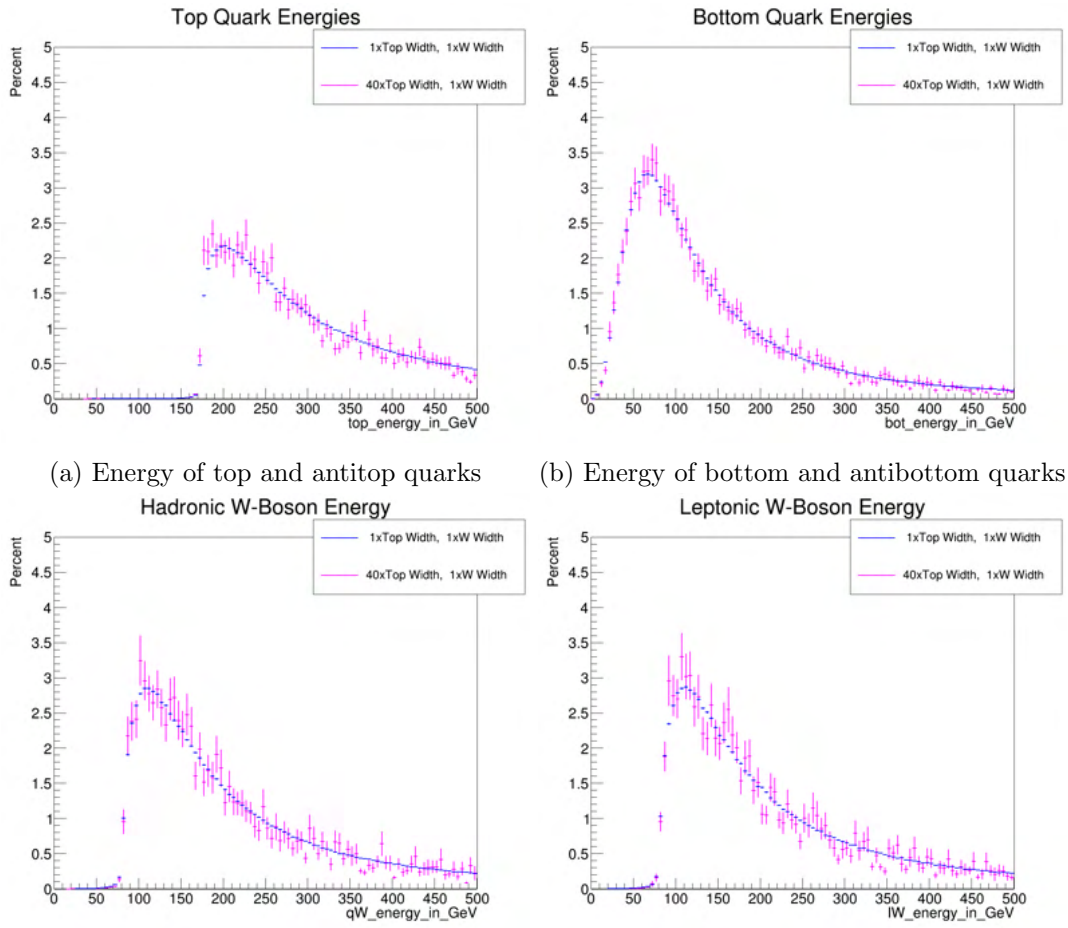
To determine, whether the differences in Figure 2.11 and Figure 2.12 are simply a direct result of the changed mass distributions, or whether there are some side effects leading to a changed kinematic, the events in the widened dataset can be weighted according to the ratio between the original and widened distributions. As the top quark masses and the W boson masses are respectively independent from each other, the weight in each case is simply the product of the ratios.

$t\bar{t}$:

$$W_{t\bar{t}}(M_t, M_{\bar{t}}) = \frac{f(M_t, M_{t_{peak}}, \Gamma_{t,orig})}{f(M_t, M_{t_{peak}}, \Gamma_{t,wide})} \cdot \frac{f(M_{\bar{t}}, M_{t_{peak}}, \Gamma_{t,orig})}{f(M_{\bar{t}}, M_{t_{peak}}, \Gamma_{t,wide})} \tag{2.7}$$

W^+W^-:

$$W_{W^+W^-}(M_{W^+}, M_{W^-}) = \frac{f(M_{W^+}, M_{W_{peak}}, \Gamma_{W,orig})}{f(M_{W^+}, M_{W_{peak}}, \Gamma_{W,wide})} \cdot \frac{f(M_{W^-}, M_{W_{peak}}, \Gamma_{W,orig})}{f(M_{W^-}, M_{W_{peak}}, \Gamma_{W,wide})} \tag{2.8}$$



(a) Energy of top and antitop quarks

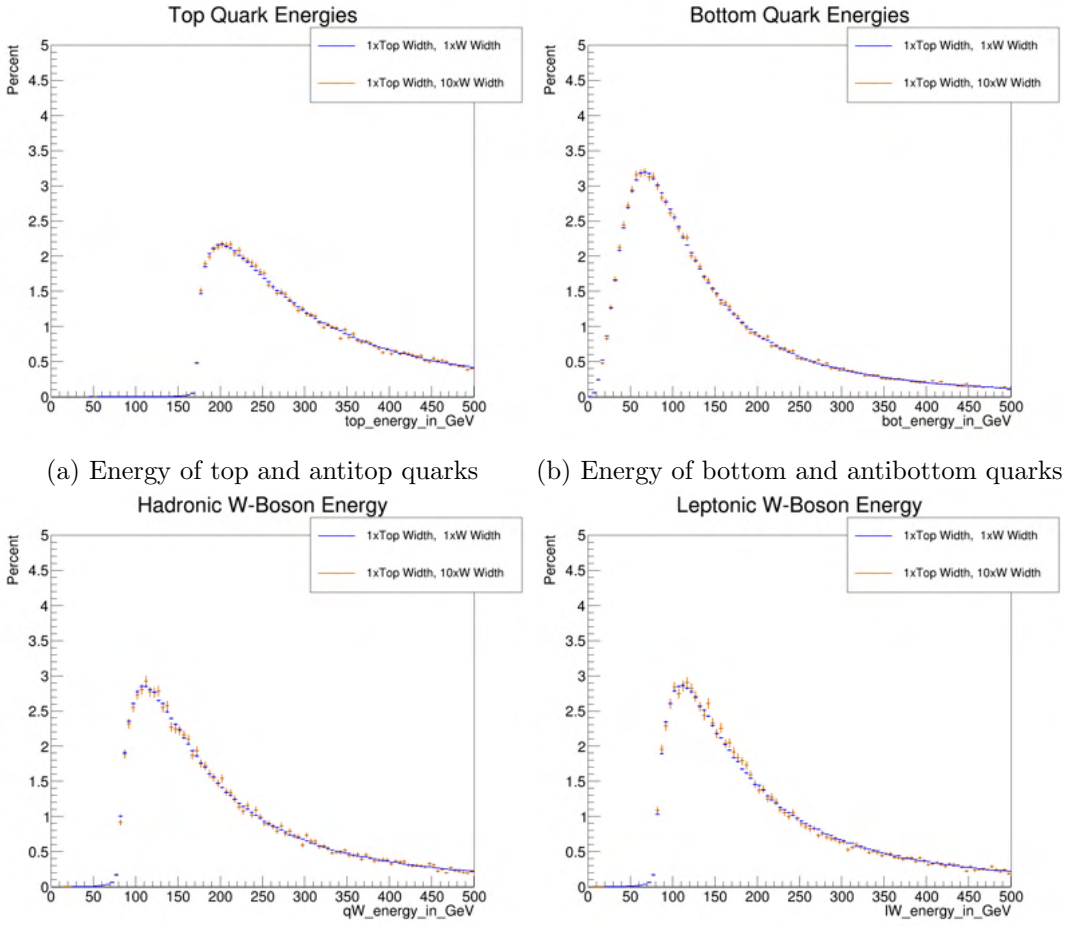
(b) Energy of bottom and antibottom quarks

(c) Energy of hadronic W boson

(d) Energy of leptonic W boson

Figure 2.13: Effect on energy curves by increasing the Γ of the top quarks by a factor of 40, after weighting, with standard $t\bar{t}$-dataset (blue) and widened $t\bar{t}$-dataset (magenta), relative to the number of events

Despite the increase in error bars, the matches between the energies of the original and reweighted wide top quark mass distributions in Figure 2.13 are apparent.



(a) Energy of top and antitop quarks

(b) Energy of bottom and antibottom quarks

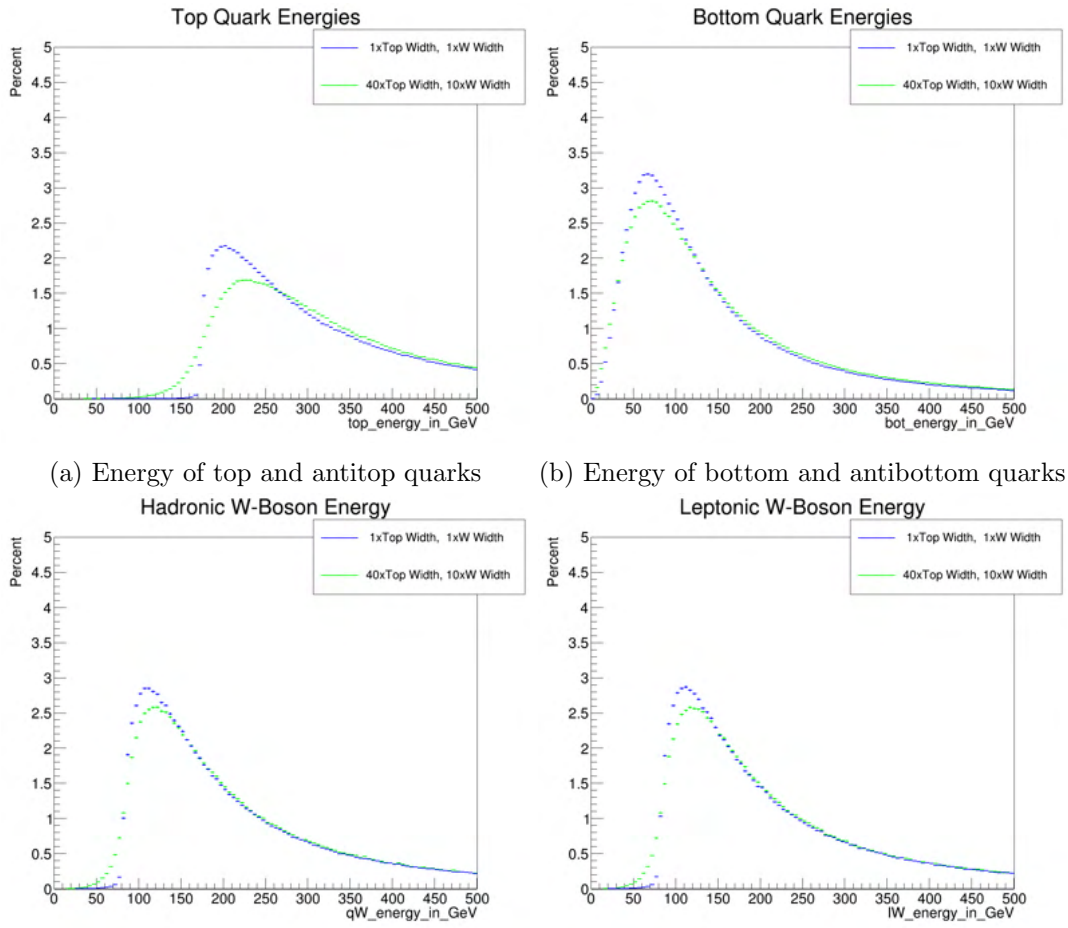
(c) Energy of hadronic W boson

(d) Energy of leptonic W boson

Figure 2.14: Effect on energy curves by increasing the Γ of the Γ of the W bosons by a factor of 10, after weighting, with standard $t\bar{t}$-dataset (blue) and widened $t\bar{t}$-dataset (orange), relative to the number of events

In the case of the reweighted widened W boson mass distribution, as Figure 2.14 shows, the energy curves match.

Next, both approaches are combined in a dataset with both widened top quark mass distributions and widened W boson mass distributions.



(a) Energy of top and antitop quarks

(b) Energy of bottom and antibottom quarks

(c) Energy of hadronic W boson

(d) Energy of leptonic W boson

Figure 2.15: Effect on energy curves by increasing the Γ of the top quarks by a factor of 40 and the Γ of the W bosons by a factor of 10, with standard $t\bar{t}$-dataset (blue) and widened $t\bar{t}$-dataset (green), relative to the number of events

As the sum of the masses of the decay products of the top quark can not be larger than the mass of top quark itself, there is a dependency between the mass of a top quark and its W boson. As a result, the overall probability of the mass of a W boson being between 0 and the difference of the mass of the top quark and the mass of the bottom quark has to be 1 for any specific value of the mass of the top quark larger than the mass of the bottom quark.

Bottom quark mass distributions have a very small width, so this width can be ignored.

$$\int_0^{M_t - M_{b_{peak}}} f_{Wt}(M_W, M_t, \Gamma_W, \Gamma_t) dM_W = f_t(M_t, \Gamma_t) \tag{2.9}$$

From this follows:

$$\begin{aligned} f_{Wt}(M_W, M_t, \Gamma_W, \Gamma_t) =& \frac{f_t(M_t, \Gamma_t) f_W(M_W, \Gamma_W)}{\int_0^{M_t - M_{b_{peak}}} f_W(M_W, \Gamma_W) dM_W} \\ =& \frac{f_t(M_t, \Gamma_t) f_W(M_W, \Gamma_W)}{\arctan(\frac{M_{W_{peak}}}{\Gamma_W}) + \arctan(\frac{(M_t - M_{b_{peak}})^2 - M_{W_{peak}}^2}{\Gamma_W M_{W_{peak}}})} \end{aligned} \tag{2.10}$$

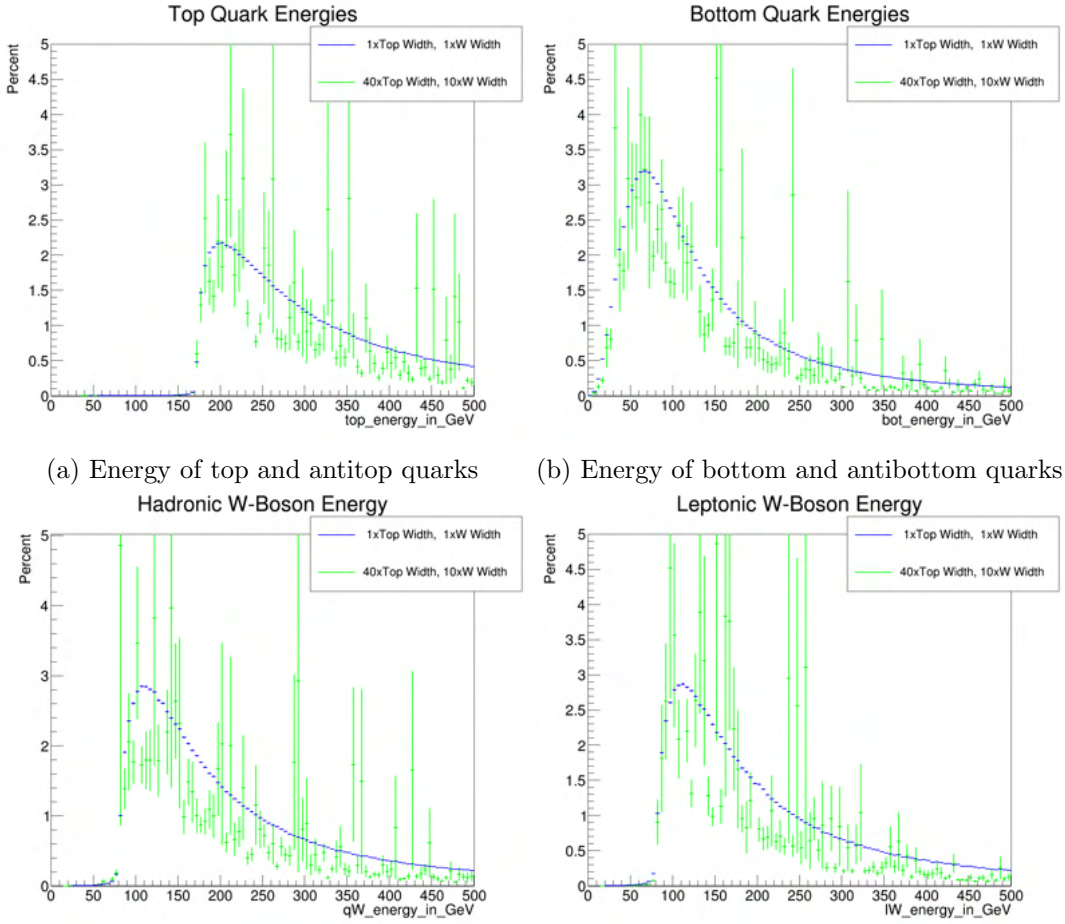
The weight according to factors M_W and M_t has to be the quotient of their mass distributions

for the original and the widened datasets.

$$W_{Wt}(M_W, M_t) = \frac{f_{Wt}(M_W, M_t, \Gamma_{W2}, \Gamma_{t2})}{f_{Wt}(M_W, M_t, \Gamma_{W1}, \Gamma_{t1})} \tag{2.11}$$

This puts the overall weight according to all four factors at:

$$W_{W^+W^-t\bar{t}}(M_{W^+}, M_{W^-}, M_t, M_{\bar{t}}) = W_{W^-t}(M_{W^-}, M_t) \cdot W_{W^+\bar{t}}(M_{W^+}, M_{\bar{t}}) \tag{2.12}$$



(a) Energy of top and antitop quarks

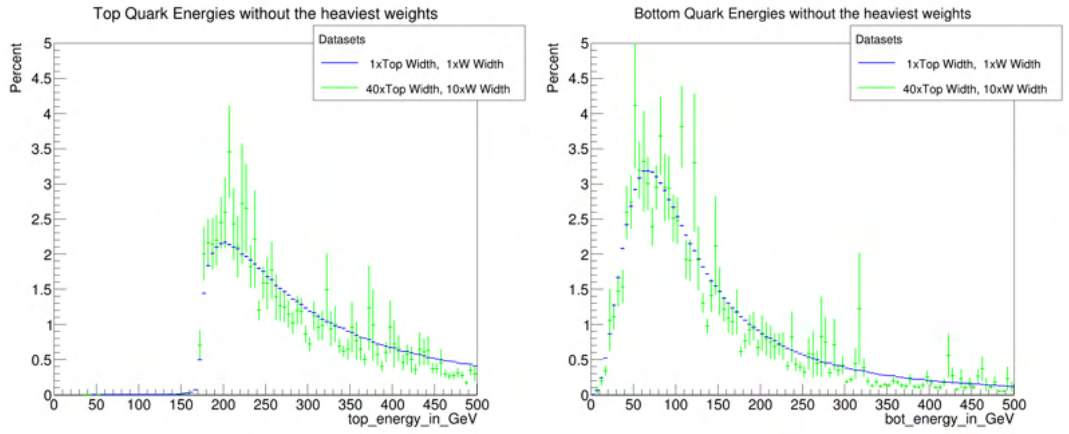
(b) Energy of bottom and antibottom quarks

(c) Energy of hadronic W boson

(d) Energy of leptonic W boson

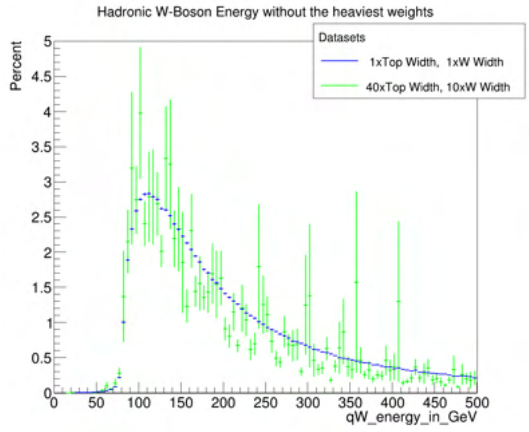
Figure 2.16: Effect on energy curves by increasing the Γ of the top quarks by a factor of 40 and the Γ of the W bosons by a factor of 10, after weighting, with standard $t\bar{t}$-dataset (blue) and widened $t\bar{t}$-dataset (green), relative to the number of events

As seen in Figure 2.16, in the combined case, the error bars become too large to visually compare, as the number of events that are on the peak of each respective mass distribution becomes much smaller for a case with all four distributions widened. There are 256,539 events, which are within the respective standard Γ around the peak in all four distributions, while for the widened dataset this only applies to 34 events. As such, those 34 events weighted to represent 256,539 exceed everything else and are therefore removed.

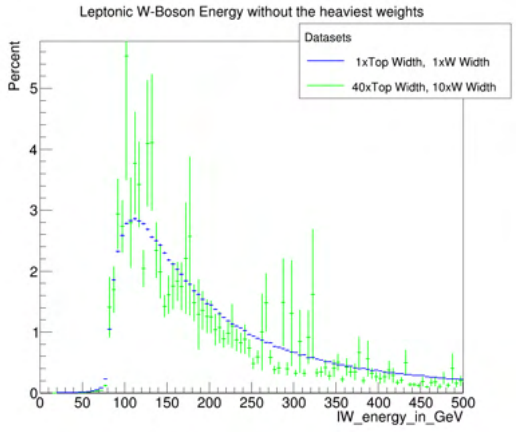


(a) Energy of top and antitop quarks

(b) Energy of bottom and antibottom quarks



(c) Energy of hadronic W boson



(d) Energy of leptonic W boson

Figure 2.17: Effect on energy curves by increasing the Γ of the top quarks by a factor of 40 and the Γ of the W bosons by a factor of 10, without the peaks, after weighting, with standard $t\bar{t}$-dataset (blue) and widened $t\bar{t}$-dataset (green), relative to the number of events

Alternatively, it is possible to compare the datasets using the Kolmogorov-Smirnov test. (14)

	top	bot	W_{had}	W_{lep}
All Events	0.715785	0.728821	0.540875	0.444878
No Peak	0.000131	0.247097	0.274831	0.009196

Table 2.8: Kolmogorov-Smirnov Test on the energies of top quarks, bottom quarks, the hadronic W boson and the leptonic W boson. To make the distribution more compatible with the test's requirements, the bin size has been lowered from 5GeV to 1GeV, over and underflow have not been considered.

There might actually be a problem, if we ignore the peak, but outside of this, it seems fine, though we would need more data to be sure.

The results of using the TBW-conditions on $t\bar{t}$-events with widened top quark mass and W boson mass, are:

	All	T	B	W
Abs	918,718	195,745	206,204	21,860
Rel	100.000%	21.306%	22.445%	2.379%
	TBW	TB	TW	BW
Abs	933	39,631	4,882	4,452
Rel	0.102%	4.314%	0.531%	0.485%

Table 2.9: Number of events meeting the target conditions out of 918,718 widened $t\bar{t}$-events

	All	T	B	W
Abs	686,999	206,734	145,961	43,875
Rel	74.778%	22.502%	15.887%	4.776%
	TBW	TB	TW	BW
Abs	3,163	39,679	15,963	9,273
Rel	0.344%	4.319%	1.738%	1.009%

Table 2.10: Number of events meeting the target conditions out of 918,718 widened $t\bar{t}$-events, after reconstruction and smearing

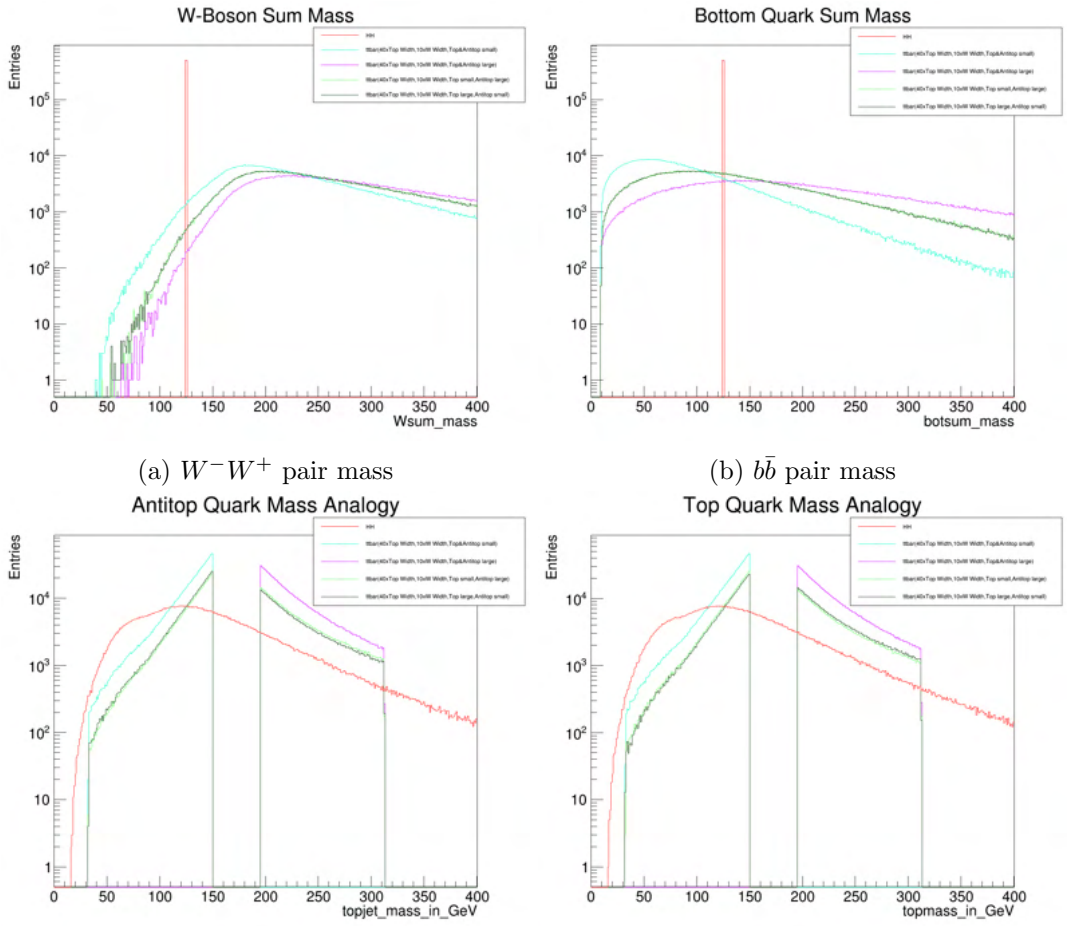
While 3,163 events is less than passed in Section 2.3.2, it is to note, that the dataset there has 5M events while this one has only ca. 1M events, so it still represents an increase in relative terms.

Looking at this, widening helps, but is still not enough to train a neural network.[1]

2.3.4 Generating a dataset using the condition

As the widening is not enough, as a more drastic measure, four 1M event $t\bar{t}$-datasets are generated, with the widened Γs for W boson and top mass distributions, one with top and antitop quark masses $< 150GeV$, one with top and antitop quark masses $> 195GeV$ and two with either top $< 150GeV$ and antitop $> 195GeV$ or vice versa. For the last two, the way they were supposed to be set, was, one has small topmass and large antitopmass, and the other large topmass and small antitopmass. This distinction seems to have gone lost somewhere during generation, so both have both scenarios mixed, with only a slight preference for the intended one. As they complement each other, when combined, this can be ignored.

[1] This dataset with wide top and W boson mass distributions is older and was generated with slightly different settings than the other datasets featured here: $\sqrt{s} = 13TeV$ and Final State Interactions and Multiparton Interactions switched on, despite this, the major points should not change.



(a) W^-W^+ pair mass

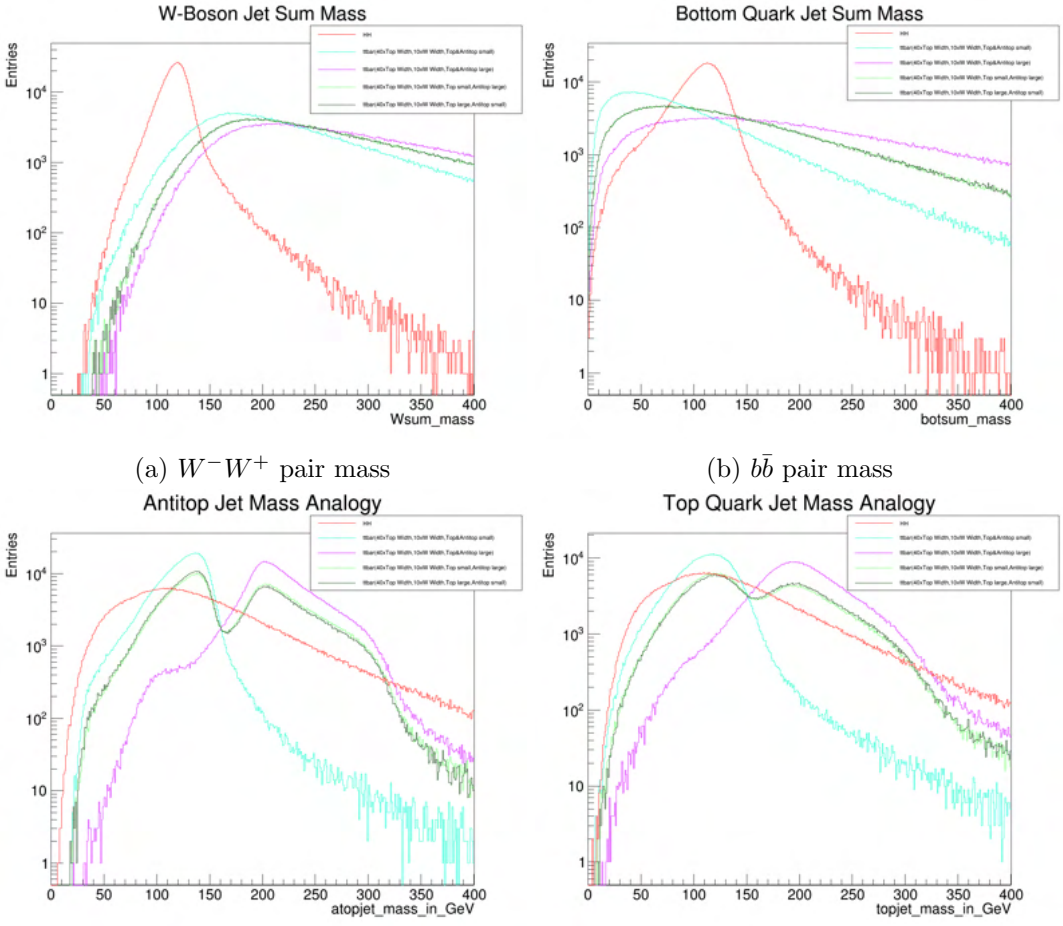
(b) $b\bar{b}$ pair mass

(c) $W^-\bar{b}$ pair mass

(d) W^+b pair mass

Figure 2.18: Higgs boson and top quark mass equivalents comparison in logarithmic scaling between 1M HH-Events (red), and four sets of 1M $t\bar{t}$-Events: one with both top and antitop quark masses small (cyan), one with both top and antitop quark mass large (purple) and two with one of them small and the other large (green)

After reconstruction, due to there being no events outside the T-restriction moving in to replace the ones inside moving out, the peak moves away from the gap.



(a) W^-W^+ pair mass

(b) $b\bar{b}$ pair mass

(c) $W^-\bar{b}$ pair mass

(d) W^+b pair mass

Figure 2.19: Higgs boson and top quark mass equivalents comparison after jet reconstruction in logarithmic scaling between 1M HH-Events (red), and four sets of 1M $t\bar{t}$-Events: one with both top and antitop quark masses small (cyan), one with both top and antitop quark mass large (purple) and two with one of them small and the other large (green)

After the datasets go through jet reconstruction, 32,920 events pass the TBW-conditions.

	All	T	B	W
Abs	2,855,412	2,032,572	566,006	221,184
Rel	71.39%	50.81%	14.15%	5.53%
	TBW	TB	TW	BW
Abs	32,920	391,400	170,278	43,837
Rel	0.82%	9.78%	4.26%	1.10%

Table 2.11: Number of events with target conditions out of 3,999,999 $t\bar{t}$-events

With smearing, that number goes slightly down, to 32,802.

	All	T	B	W
Abs	2,855,412	1,943,094	557,686	236,823
Rel	71.39%	48.58%	13.94%	5.92%
	TBW	TB	TW	BW
Abs	32,802	365,981	174,214	46,275
Rel	0.82%	9.15%	4.36%	1.16%

Table 2.12: Number of events with target conditions out of 3,999,999 $t\bar{t}$-events

Even though this number is not very large, there is not much more to be done.

2.4 Angle between W bosons

For HH-events, as the invariant mass of a Higgs boson has a peak at 125GeV and the invariant mass of a W bosons has a peak at 80.42GeV, which forces at least one of the resulting W bosons to be off shell and also leaves only a small amount of energy over to be converted into kinetic energy. Thus, a small angle near 0° between the W bosons in the HH rest frame is preferred (9).

$$m_H^2 = (\mathbf{p}_{W^+} + \mathbf{p}_{W^-})^2 = m_{W^+}^2 + m_{W^-}^2 + 2(E_{W^+}E_{W^-} - \vec{p}_{W^+}\vec{p}_{W^-}) \tag{2.13}$$

$$= m_{W^+}^2 + m_{W^-}^2 + 2E_{W^+}E_{W^-}(1 - \beta_{W^+}\beta_{W^-}\cos\angle(W^+, W^-)) \tag{2.14}$$

Thus both W bosons are the same weight: $m_{W^+} = m_{W^-} = m_W$ and following that, because of conservation of momentum: $E_{W^+} = E_{W^-} = E_W$ and $\beta_{W^+} = \beta_{W^-} = \beta_W$

$$m_H^2 = 2m_W^2 + 2E_W^2(1 - \beta_W^2\cos\angle(W^+, W^-)) \tag{2.15}$$

From this follows:

$$\beta_W^2\cos\angle(W^+, W^-) = \frac{2m_W^2 - m_H^2}{2E_W^2} + 1 = 1 - \frac{m_H^2 - 2m_W^2}{2E_W^2} \tag{2.16}$$

With $E_W^2 \geq m_W^2$ a maximum value for $\cos\angle(W^+, W^-)\beta_W^2$ can be found:

$$\cos\angle(W^+, W^-)\beta_W^2 \geq 1 - \frac{m_H^2 - 2m_W^2}{2m_W^2} = 2 - \frac{m_H^2}{2m_W^2} \tag{2.17}$$

Putting all masses on-shell.

$$\cos\angle(W^+, W^-)\beta_W^2 \geq 2 - 0.5\left(\frac{125GeV}{80.4GeV}\right)^2 \approx 0.7914 \tag{2.18}$$

Though, this scenario is not possible, as the sum of the invariant masses of the W bosons cannot be larger than the mass of the Higgs boson, they decay from, a scenario, where both W bosons are 62.5 GeV should also be considered, in which case:

$$\cos\angle(W^+, W^-)\beta_W^2 \geq 2 - 0.5\left(\frac{125GeV}{62.5GeV}\right)^2 = 0 \tag{2.19}$$

And as $\beta_W^2 \geq 0$ by definition, as the value of the velocity has to be a real number, unless $\beta_W^2 = 0$, it follows, that $\cos \angle(W^+, W^-) \geq 0$

For $t\bar{t}$-events, as top quarks have spin $\frac{1}{2}$, and the gluon they come from has a spin of 1, the W bosons with spin 1 should point apart, leading to a spike at 180°.

Even ignoring the spin, if the kinetic energy of the top quark is much larger than the kinetic energy released by its decay, the angle between the W bosons in the $t\bar{t}$ rest frame should be near 180°, the angle of the top quarks to one another in the $t\bar{t}$ rest frame, in the reverse extreme, it should resemble the relative angle of two random three dimensional vectors, in which case the angle would peak at 90°, as the cosine between the z-coordinate unit vector and a another unit vector is:

$$\int_0^{2\pi} \left(\int_0^{\pi} \hat{z} \cdot \hat{v}(\phi, \theta) d\theta \right) d\phi = \int_0^{2\pi} \left(\int_0^{\pi} \cos(\theta) d\theta \right) d\phi = 0 \tag{2.20}$$

So in either way, the angle should not be small.

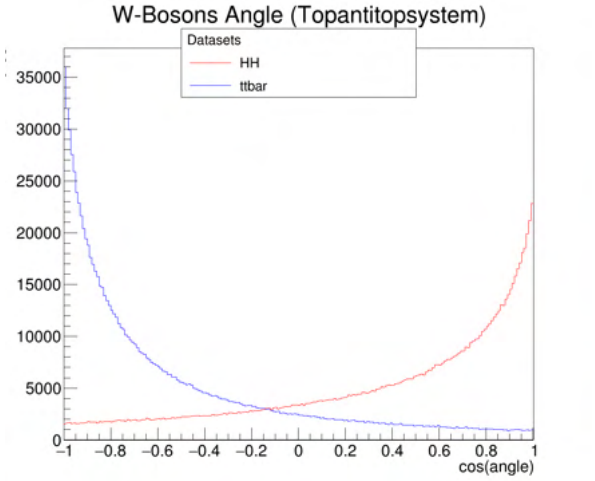


Figure 2.20: Comparison between the cosines of the angles of the W bosons of 1M $t\bar{t}$-events in the $t\bar{t}$ rest frame (blue) and 1M HH-events in the HH rest frame (red).

This difference with the angle between W-Bosons tending to be small in $t\bar{t}$-events and large in HH-events, seems like a good indicator for differentiating between the two in the general case.

The last test would be, how it looks, after applying all the steps from Section 2.3:

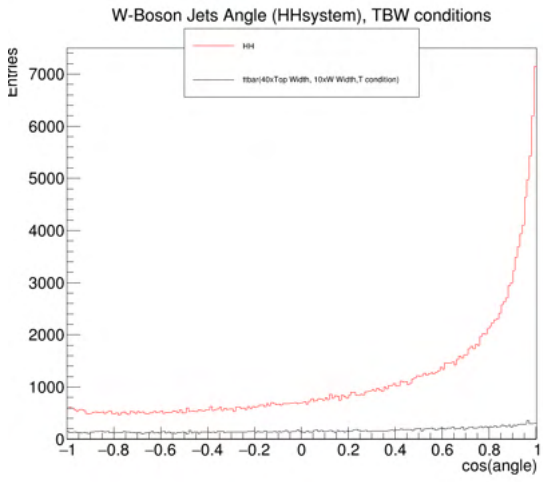


Figure 2.21: Comparison between the cosines of the angles of the W bosons of the 4M T-restricted $t\bar{t}$-events in the $t\bar{t}$ rest frame (blue) and 1M HH-events in the HH rest frame (red), after applying jet reconstruction, smearing, and the TBW conditions.

As Figure 2.21 shows, after applying all of the TBW-conditions, as well as jet reconstruction and smearing on a T-restricted dataset, the difference becomes much less pronounced. The angle distribution of $t\bar{t}$-events becomes much more flat with a small tendency towards large angles, weakening its usefulness for differentiating between the two different events, but possibly not eliminating it entirely.

2.5 Angle between bottom quarks

As the bottom quarks have a very small invariant mass of 4.8GeV, for HH-events, the peak of a Higgs boson at 125GeV, which leaves 57.7GeV for kinetic energy per bottom quark in the Higgs boson rest frame. To have an angle smaller than 90° between the boosted bottom quarks, the HH pair needs to have a combined energy of over 353GeV, and even then, only if the direction of the bottom quarks in the $b\bar{b}$ rest frame were perpendicular to the direction of the Higgs boson in the HH rest frame, the energy has to be at least about 3.26TeV, so the angle between the bottom quarks becomes smaller than 90° for every orientation of the Higgs boson decay to its direction of movement.

Overall, the equation for the angle between the bottom quarks in the HH rest frame for decay perpendicular to the direction of the Higgs boson is (derivation of Equation 2.21 can be found in Appendix C):

$$cos(\theta_{bb,perpend}) = 1 - \frac{2m_H^2 - 8m_b^2}{E_H^2 - 4m_b^2} \tag{2.21}$$

as for the angle to become 90°, $cos(\theta_{bb}) = 0$, E_H is

$$E_{H,min,perpend} = \sqrt{2m_H^2 - 4m_b^2} \approx 176.5GeV \tag{2.22}$$

making $E_{HH} \approx 353GeV$

If the decay is parallel to the direction of the Higgs boson, the angle between the bottom quarks in the HH rest frame stays 180°, unless the velocity of the Higgs boson in the HH-event is larger, than the velocity of bottom quarks, in which case:

$$E_{H,min,parallel} = \frac{m_H^2}{2m_b} \approx 1.628TeV \tag{2.23}$$

making $m_{HH} \approx 3.255TeV$.

Meanwhile, the actual distribution of m_{HH} is:

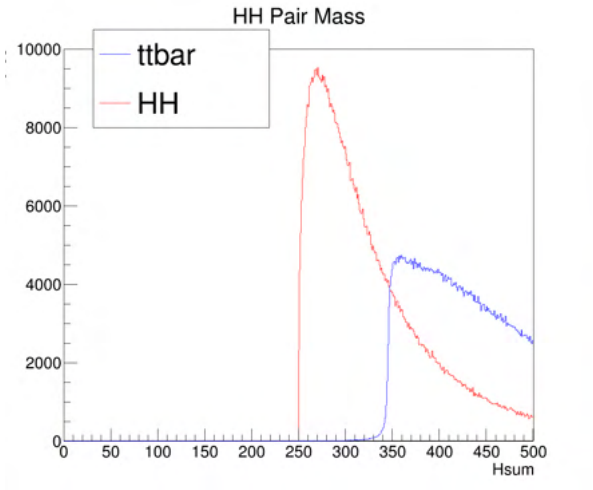


Figure 2.22: The masses of the HH pair of HH-events (red) and the $t\bar{t}$ pair of $t\bar{t}$-events (blue) in comparison, for 1M events

with a peak at roughly 270GeV, where according to Equation 2.21 θ_{bb} cannot be smaller than around 135.5°. θ_{bb} should be expected to peak at a small value near 180°.

For $t\bar{t}$-events, as a consequence of the direction of the W bosons tending towards 180° because of their spins as described in section 2.4, the same applies to bottom quarks, which go into the opposite direction.

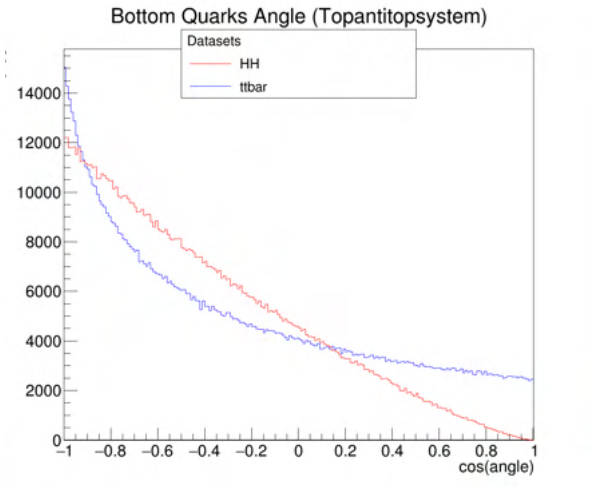


Figure 2.23: Comparison between the cosines of the angles of the bottom quarks of 1M $t\bar{t}$-events in the $t\bar{t}$ rest frame (blue) and 1M HH-events in the HH rest frame (red).

In conclusion, the angle between bottom quarks tends to be small for both $t\bar{t}$- and HH-events, making it less useful for differentiating between the two, than the angle between the W quarks.

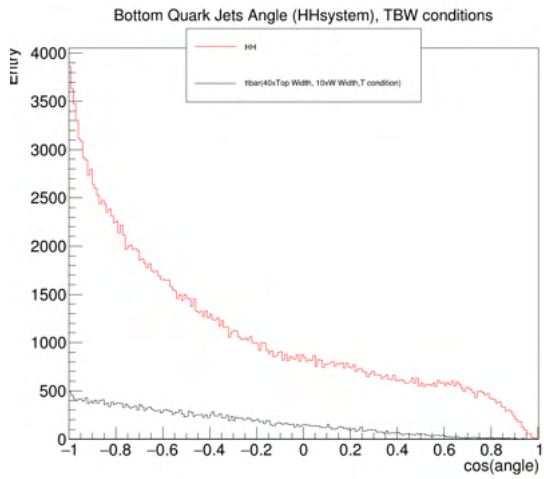


Figure 2.24: Comparison between the cosines of the angles of the bottom quarks of the 4M T-restricted $t\bar{t}$-events in the $t\bar{t}$ rest frame (blue) and 1M HH-events in the HH rest frame (red), after applying jet reconstruction, smearing and the TBW-conditions.

And as Figure 2.24 shows, applying jet reconstruction, smearing and the TBW-conditions on a T-restricted dataset, changes little about this.

Chapter 3

Neural Network

3.1 Data

Looking at the events passing all the TBW-conditions:

	All	Jet	Truth TBW	Jet TBW	Smeared Jet TBW
Standard $t\bar{t}$	5,000,000	3,718,227	1	4,542	6,582
T-Restricted $t\bar{t}$	3,999,999	2,855,412	30,846	32,920	32,802
HH	998,162	637,364	614,090	255,730	219,423

Table 3.1: Overview over the different candidates of datasets considered for training a neural network

The TBW of the standard $t\bar{t}$ seems like a useful benchmark for a neural network of the changes to the mass distribution of top quarks and W bosons in the T-restricted $t\bar{t}$ datasets, because of this, despite the small hit in the number of events, the smeared reconstructed T-restricted events passing the TBW-conditions seem like the best candidate to train a neural network on.

3.2 Structure

The code used for generating the neural network is based on an internal tutorial (10).

3.2.1 Inputs

The 20 parameters chosen for inputs can be sorted into four groups:

-JetP: The kinetic energies in x- and y-direction of the four jets (botPx, botPy, abotPx, abotPy, qPx, qPy, aqPx, aqPy)

-TruthP: The kinetic energies in x-and y-direction of the two truth particles (lepPx, lepPy, nyPx, nyPy)

-Rest: The energies of the sum of the jets not matched to any particle (RestPx, RestPy, RestM)

-Indicators: Parameters, which have shown to behave differently between HH- and $t\bar{t}$-Events (Wangle, topM, atopM, WHiggsM, bHiggsM)

Only the x- and y-directions are used, as the z-direction is affected by the unknown longitudinal boost of the parton-parton collision system.

3.2.2 Nodes

The value a in each node j of a layer i is calculated by:

$$a_{i+1,j} = f(\sum_{j=1}^{n_i}(w_{i,j}a_{i,j}) - b_i) \tag{3.1}$$

with f(x) here being a leaky ReLU function (2).

$$f(x) = \begin{cases} x & \forall x > 0 \\ \alpha x & \forall x \leq 0 \end{cases} \tag{3.2}$$

with $\alpha = 0.1$.

The only exception is the output layer, where the softmax function is used instead:

$$f(x_j) = \frac{e^{x_j}}{\sum_j e^{x_j}} \tag{3.3}$$

In every layer, there are 20 nodes, except for the output layer, which has two, one for HH-events and one for $t\bar{t}$-events. For evaluation of the neural network, the output node with the higher value is treated as the classification for the event by the network.

3.3 Method

First 10% of the HH- and $t\bar{t}$-datasets are set aside for validation testing, then the rest is used for training.

Every hidden node has a 20% chance of Dropout, to combat overfitting.

Every epoch, the training data is split into random batches of a given size.

For every batch, every weight and bias is changed according to the average of their respective derivative of the loss function, times a constant called the learning rate, which is set to 0.01.

Also, to have a consistent indicator between different training datasets and to make sure, that the changes done to the datasets in Sections 2.3.3 and 2.3.4 do not mislead the network, events that pass the TBW-conditions from smeared, jet reconstructed standard HH- and $t\bar{t}$-datasets are used for separate validation in the target area. To prevent overlap between training sets and the target set, for this, only the last 10k events of the HH-dataset in particular are used.

The loss function used is PyTorch's *CrossEntropyLoss*, with default settings (15).

The cross entropy function can be written as (13):

$$l_n = -\sum_{c=1}^{C} y_{n,c} \log(x_{n,c}) \tag{3.4}$$

which for two classes can be expressed as the Binary Cross Entropy function:

$$l_n = -w_n[y_n \cdot \log(x_n) + (1 - y_n) \cdot \log(1 - x_n)] \tag{3.5}$$

With x = $x_1, \ldots, x_n$ being the values of the output nodes, y= $y_1, \ldots, y_n$ representing the correct values, $C = 2$ being the number of output nodes and N being the batch size.

The last batch of the epoch covers the leftover of the events and is smaller, unless the number of events in the training data is divisible by the batch size.

3.4 Systematic Investigations

The parameters mainly looked at for the evaluation are loss, accuracy, sensitivity, and specificity. Be t_p the true positive, t_n the true negative, f_p the false positive and f_n the false negative, with positive being HH-events and negative being $t\bar{t}$-events.

Accuracy describes, what percentage of the total events were identified correctly:

$$accuracy = \frac{t_p + t_n}{t_p + t_n + f_p + f_n} \tag{3.6}$$

Loss is the average value of the loss function between all the events, between all the batches m in an epoch.

$$loss = \sum_{m=1}^{M} \frac{l_m(x, y)}{M} \tag{3.7}$$

Sensitivity describes, what percentage of the signal events (HH-events) were identified correctly:

$$sensitivity = \frac{t_p}{t_p + f_n} \tag{3.8}$$

Specificity describes, what percentage of the background events ($t\bar{t}$-events) were identified correctly:

$$specificity = \frac{t_n}{t_n + f_p} \tag{3.9}$$

The three different datasets, where those parameters are looked at, are called training data, validation test data and target data.

Training data uses the events, the neural network is trained on. The training data is tested every epoch.

Validation test data uses the validation load, the 10% of events that were separated from the training data, before training started. This is useful for recognizing overfitting. The validation test data is tested every 5th epoch.

Target data uses a set of smeared, jet reconstructed, events of standard HH- and $t\bar{t}$-events, that pass the TBW-conditions. It represents the goal of differentiating HH-events from $t\bar{t}$-events in the target area the closest and provides a common benchmark for comparing the results of neural networks trained with different datasets. To prevent overlap with the other datasets, the last 10k of the HH-events are used. The target data is tested every 5th epoch.

3.4.1 Normalization

To simulate $t\bar{t}$-events being much more likely than HH-events, the normalization is based on the full 5M $t\bar{t}$-dataset, which is the 1M and 4M $t\bar{t}$-datasets combined.

The first choice for normalization seems to be, using the average and variance of the dataset for each input variable, this however runs into two problems:

-HH-events are much more rarer than $t\bar{t}$-events and as such would not have much of an influence on the overall average in reality.

-After being trained on the T-restricted dataset, the neural network will be also tested on the events of a standard dataset, the target data. This would not work, if both had different values at normalization.

Because of this, the values used are instead the ones calculated from the full 5M events of the standard $t\bar{t}$-dataset.

The normalization formula used, is:

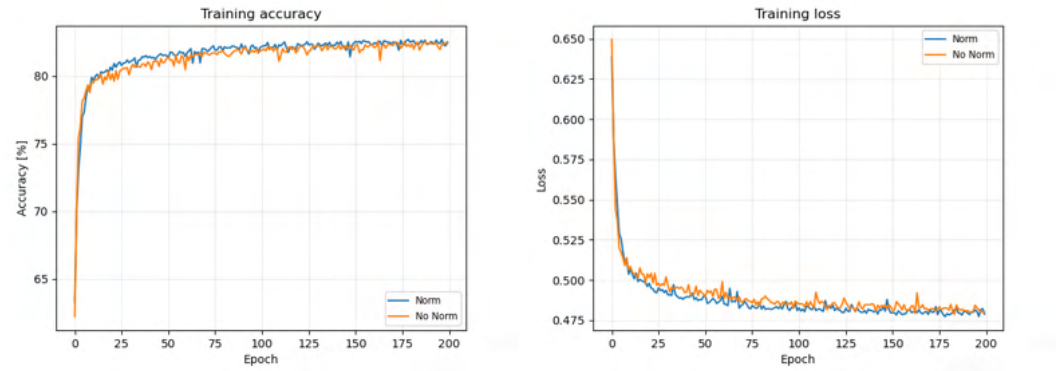
$$x_{normalized} = \frac{x - m_x}{\sigma_x} \tag{3.10}$$

With $m_x = \langle x \rangle$ and $\sigma_x = \sqrt{\langle x^2 \rangle - \langle x \rangle^2}$

Their values are:

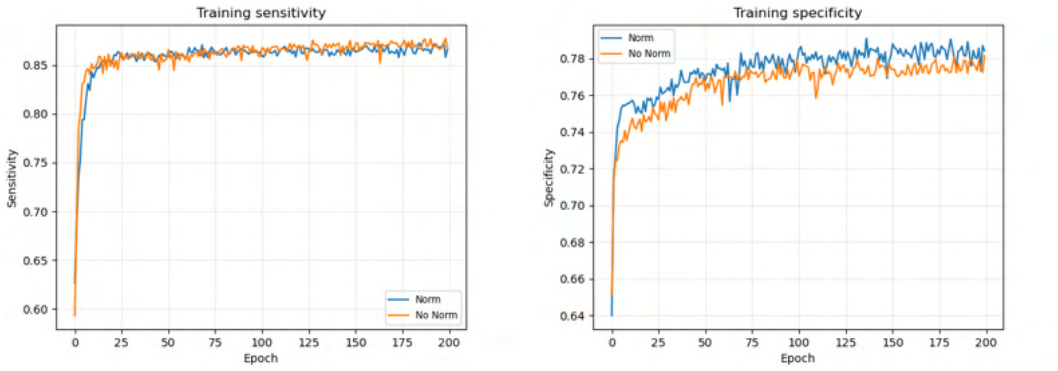
x	botPx	botPy	abotPx	abotPy
m_x	-1.1463797	-0.0031285295	0.40313664	-0.0022223252
σ_x	59.45657	61.457508	62.31287	62.65113
x	qPx	qPy	aqPx	aqPy
m_x	-0.16311064	0.03135924	-0.03135924	-0.019642025
σ_x	52.722122	52.925797	45.418373	45.43017
x	lepPx	lepPy	nyPx	nyPy
m_x	0.2221839	0.0009224076	0.27653328	-0.04254994
σ_x	46.210194	46.434994	54.953907	55.286407
x	restPx	restPy	restM	Wangle
m_x	0.5324904	0.03802755	458.50708	-0.52072614
σ_x	83.37618	83.99087	594.8185	0.503769
x	topM	atopM	WHiggsM	bHiggsM
m_x	151.68784	165.01224	277.04214	146.92615
σ_x	34.211666	23.067703	133.67793	109.85789

Table 3.2: Expected values and width of the parameters of the inputs for 5M standard $t\bar{t}$-events, used for normalization



(a) Accuracy training data

(b) Loss training data



(c) Sensitivity training data

(d) Specificity training data

Figure 3.1: Comparison of the training data of runs normalized (blue) and not normalized (orange), with 3 hidden layers and batch size 1000, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

As Figure 3.1 shows, the normalized run performs better in the training data, though not by much.

(a) Accuracy validation test data

(b) Loss validation test data

(c) Sensitivity validation test data

(d) Specificity validation test data

Figure 3.2: Comparison of the validation test data of runs normalized (blue) and not normalized (orange), with 3 hidden layers and batch size 1000, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

Figure 3.2 demonstrates, that this better performance seems to translate over to the validation test data, showing, the normalized runs performance improvement cannot be solely attributed to overfitting.

(a) Accuracy target data

(b) Loss target data

(c) Sensitivity target data

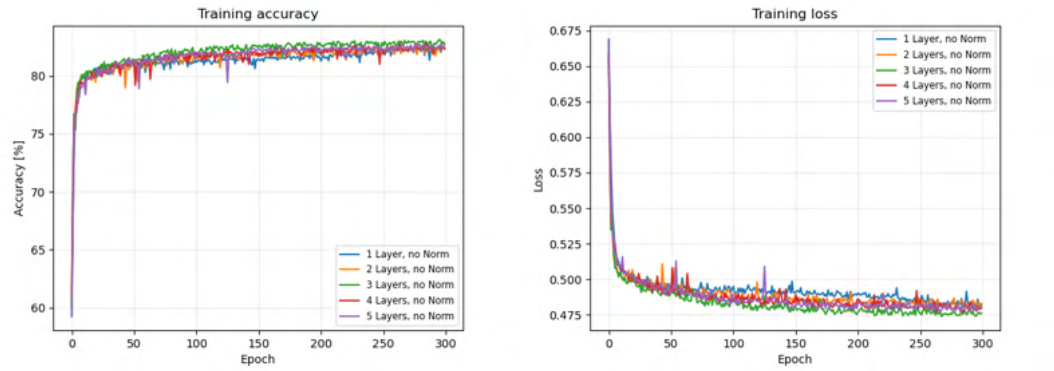
(d) Specificity target data

Figure 3.3: Comparison of the target data on runs normalized (blue) and not normalized (orange), with 3 hidden layers and batch size 1000, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

But, as Figure 3.3 shows, this difference seems to vanish, when looking at the target data.

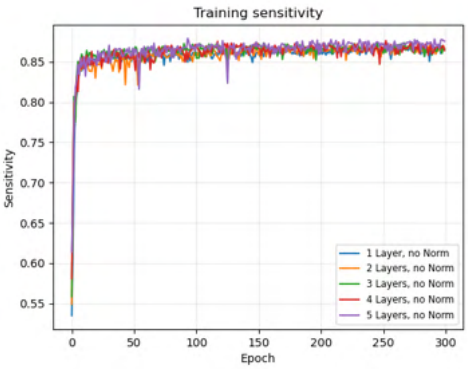
3.4.2 Layers

Next, it is investigated, if and how the number of hidden layers used affects the performance. The results of accuracy, loss, sensitivity and specificity of runs with a layer number of 1-5 is shown, when tested on training data, validation test data and target data.

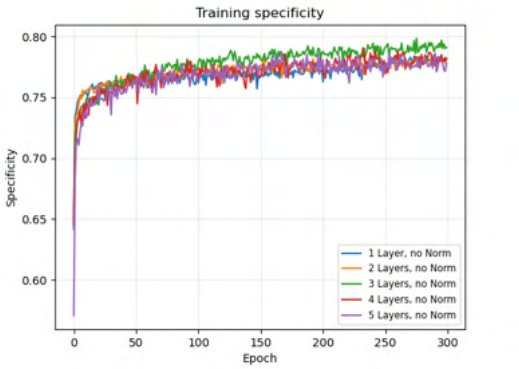


(a) Accuracy training data

(b) Loss training data



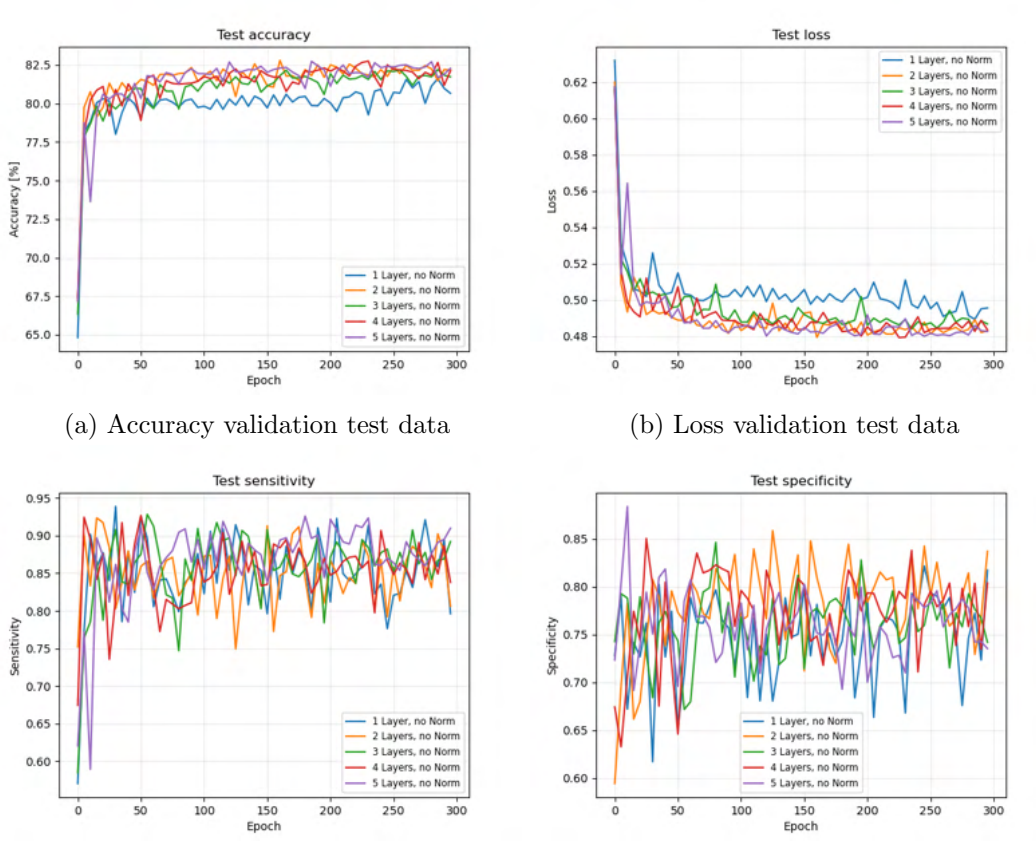
(c) Sensitivity training data



(d) Specificity training data

Figure 3.4: Comparison of the training data of runs with different amounts of hidden layers: 1 layer (blue), 2 layers (orange), 3 layers (green), 4 layer (red) and 5 layers (purple); not normalized and batch size 1000, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

As Figure 3.4 shows, for the training data, the 1 layer run seems to perform the worst and the 3 layer run performs the best.



(a) Accuracy validation test data

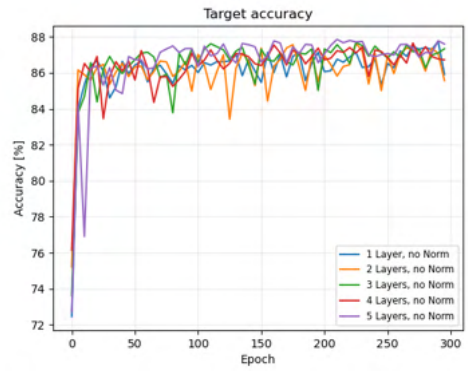
(b) Loss validation test data

(c) Sensitivity validation test data

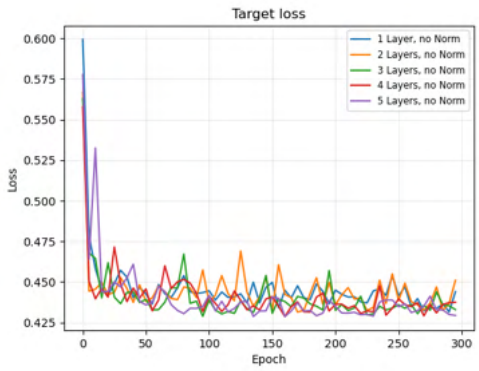
(d) Specificity validation test data

Figure 3.5: Comparison of the validation test data of runs with different amounts of hidden layers: 1 layer (blue), 2 layers (orange), 3 layers (green), 4 layer (red) and 5 layers (purple); not normalized and with batch size 1000, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

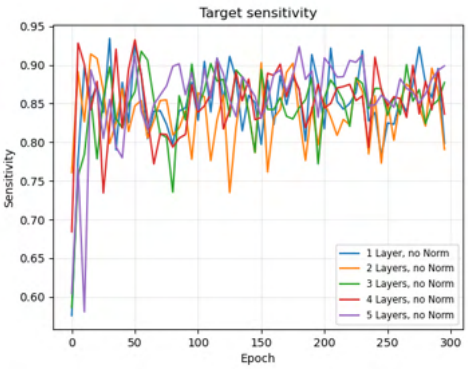
Figure 3.5, shows, for the validation test data, the 1 layer runs still performs the worst, but outside of that, there seems to be no noticeable difference between the runs, suggesting, that the performance increase for the 3 layer run for the training data was just the result of overfitting.



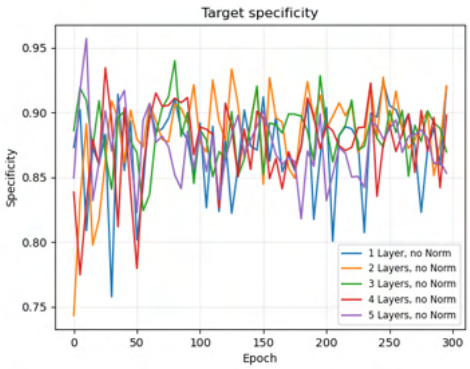
(a) Accuracy target data



(b) Loss target data



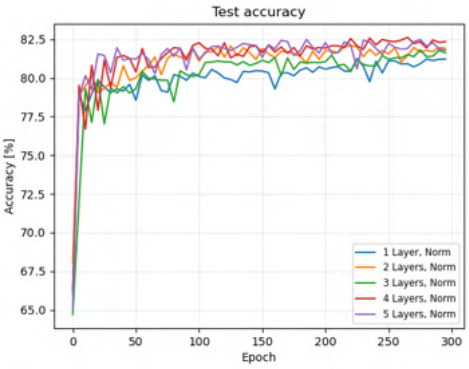
(c) Sensitivity target data



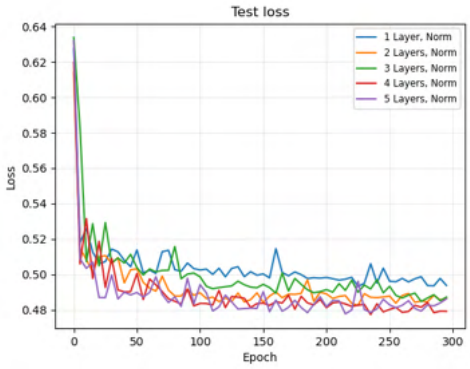
(d) Specificity target data

Figure 3.6: Comparison of the target data on runs with different amounts of hidden layers: 1 layer (blue), 2 layers (orange), 3 layers (green), 4 layer (red) and 5 layers (purple); not normalized and with batch size 1000, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the target data, the differences vanish completely in noise. Looking at the accuracy and loss of the validation test data of some similar runs for batch sizes of 1000 and 2000 with and without normalization in Figures 3.7 to 3.9

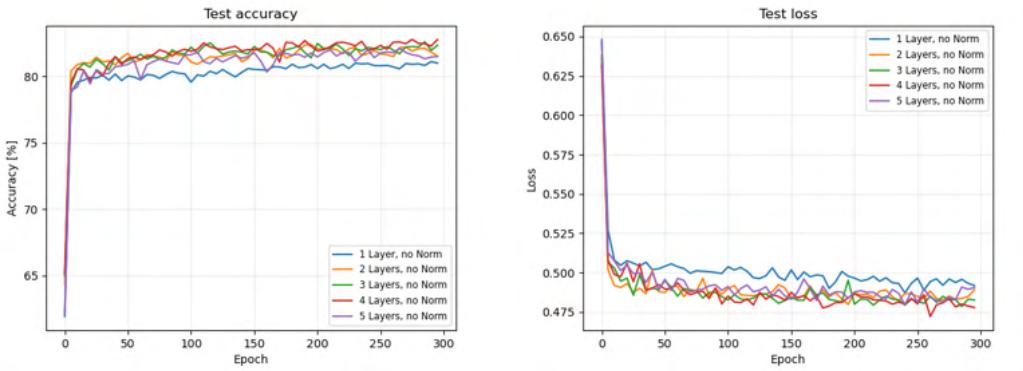


(a) Accuracy validation test data



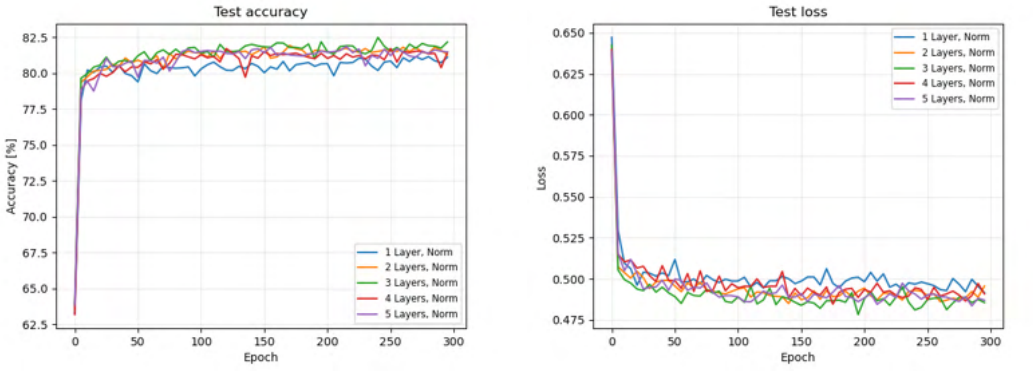
(b) Loss validation test data

Figure 3.7: Comparison of the validation test data of runs with different amounts of hidden layers: 1 layer (blue), 2 layers (orange), 3 layers (green), 4 layer (red) and 5 layers (purple); **normalized** and with **batch size 1000**, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions



(a) Accuracy validation test data (b) Loss validation test data

Figure 3.8: Comparison of the validation test data of runs with different amounts of hidden layers: 1 layer (blue), 2 layers (orange), 3 layers (green), 4 layer (red) and 5 layers (purple); **not normalized** and **batch size 2000**, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions



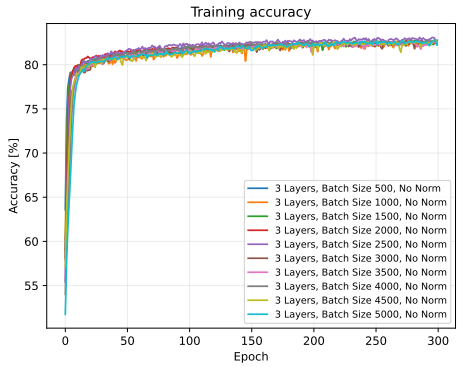
(a) Accuracy validation test data (b) Loss validation test data

Figure 3.9: Comparison of the validation test data of runs with different amounts of hidden layers: 1 layer (blue), 2 layers (orange), 3 layers (green), 4 layer (red) and 5 layers (purple); **normalized** and with **batch size 2000**, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

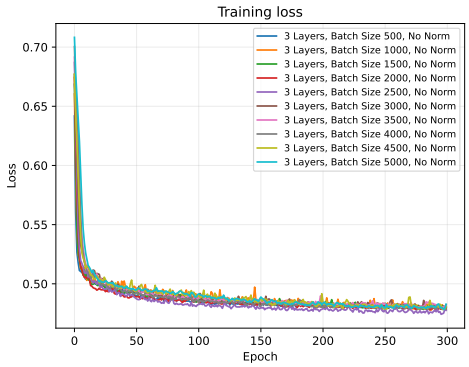
Outside of ”1 layer” performing worse, any difference in the performance between the different layers seems to be more noise than any actual difference.

3.4.3 Batch Size

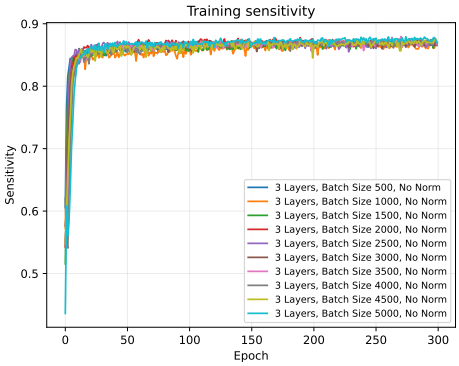
What effect have batch sizes on the runs, this is investigated without normalization and for 3 hidden layers:



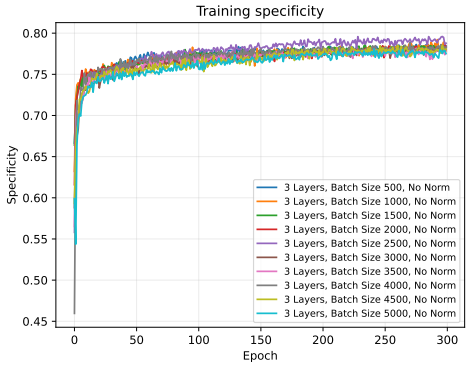
(a) Accuracy training data



(b) Loss



(c) Sensitivity training data

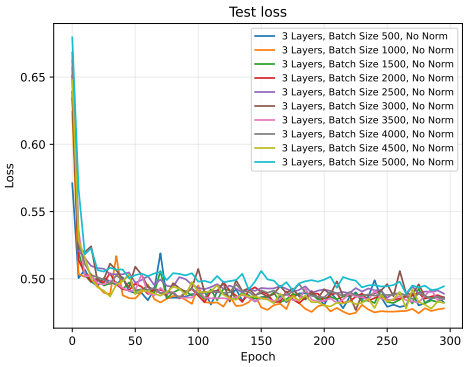


(d) Specificity training data

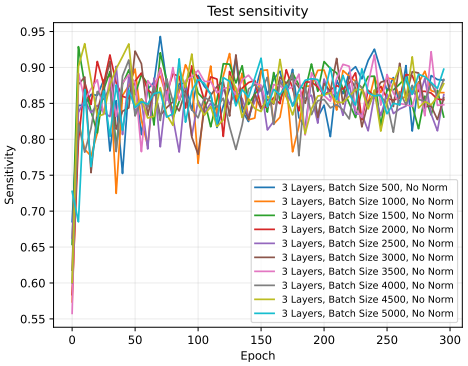
Figure 3.10: Comparison of the training data of runs with different batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); not normalized and with 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For training data, no run seems to really stand out, outside of the batch size 2500 run having a higher specificity and as a result a slightly better loss and accuracy than the rest. But this might just be a fluke

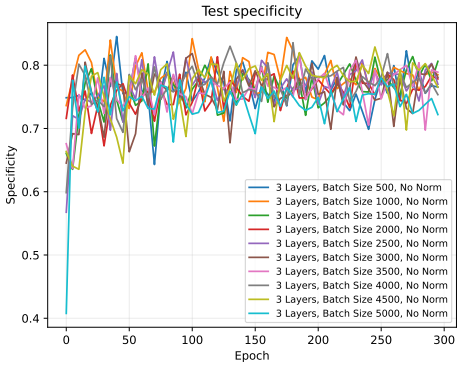
(a) Accuracy validation test data



(b) Loss validation test data



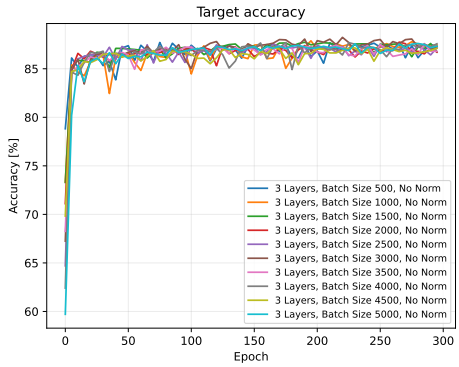
(c) Sensitivity validation test data



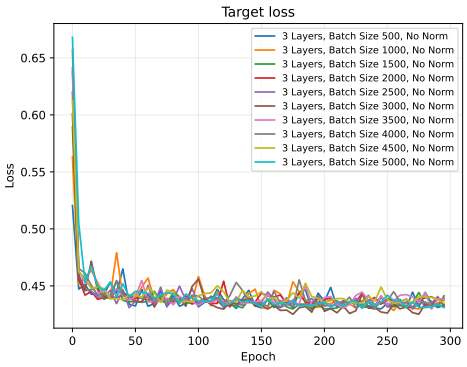
(d) Specificity validation test data

Figure 3.11: Comparison of the validation test data of runs with different batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); not normalized and with 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

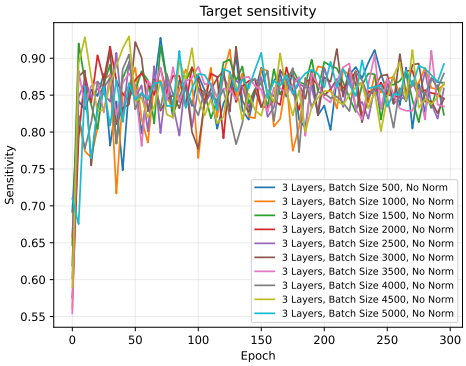
For the validation test data, as Figure 3.11 shows, the batch size 2500 run vanishes in the rest of the runs. In contrast, the new standouts are the batch size 5000 run for performing slightly worse and the batch size 1000 for performing slightly better than the rest.



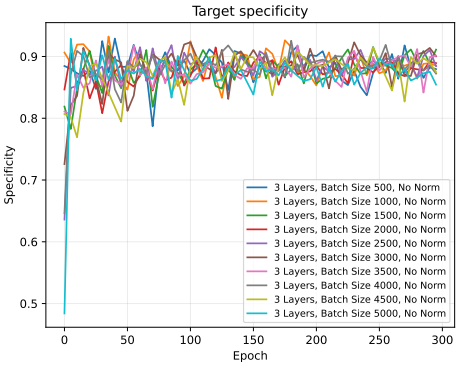
(a) Accuracy target data



(b) Loss target data



(c) Sensitivity target data



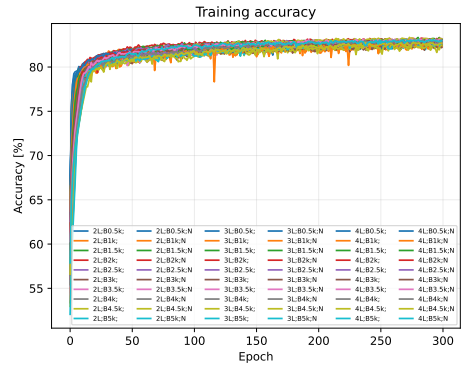
(d) Specificity target data

Figure 3.12: Comparison of the target data on runs with different batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); not normalized and with 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

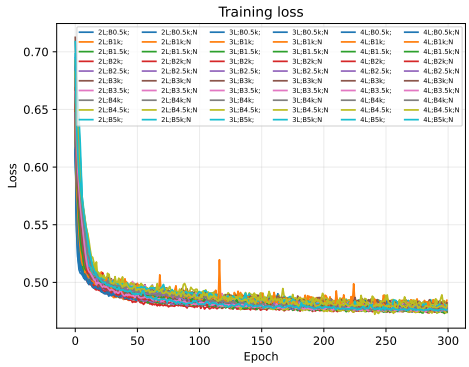
For the target data, no runs are noteworthy.

3.4.4 Combined

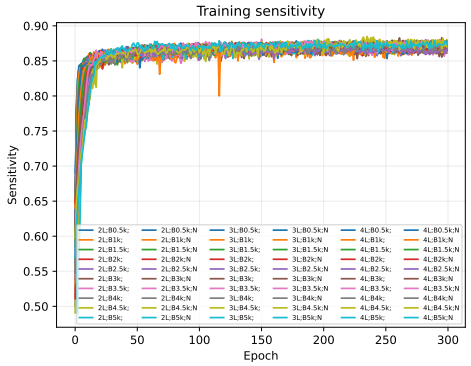
To look, if there are any standouts, a set of runs with all combinations of normalization on/off, 2 to 4 layers and batch size of 500 to 5000 are done. Looking at all of them together, yields the following observation:



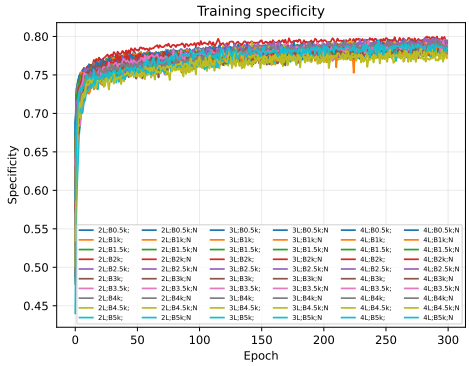
(a) Accuracy training data



(b) Loss training data



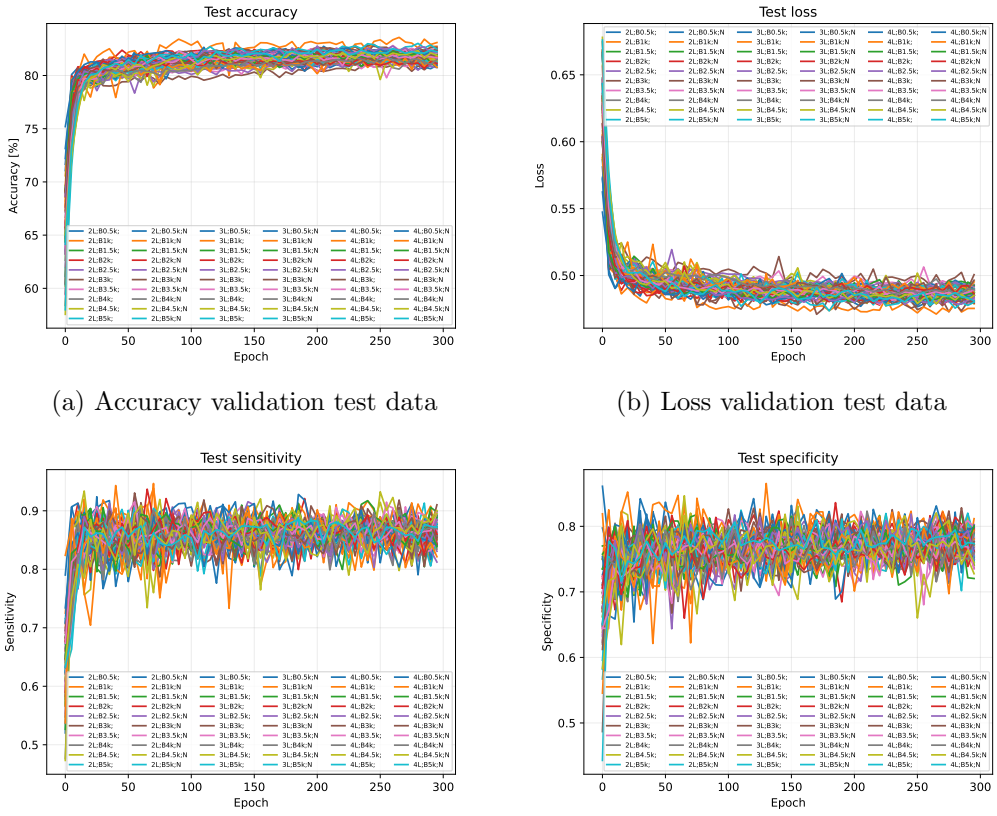
(c) Sensitivity training data



(d) Specificity training data

Figure 3.13: Comparison of the training data of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); normalized and not normalized; with 2 to 4 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

The only notable run for training data, would be the normalized 3 layer run with batch size 1000, for having a slightly higher specificity than the rest.



(a) Accuracy validation test data

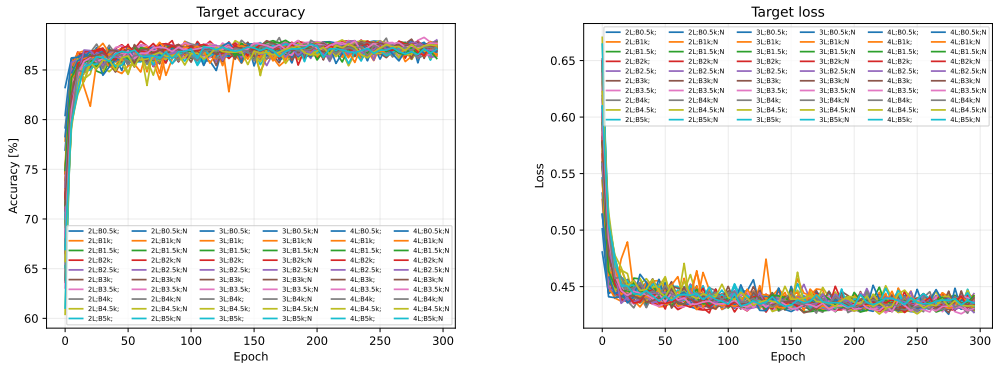
(b) Loss validation test data

(c) Sensitivity validation test data

(d) Specificity validation test data

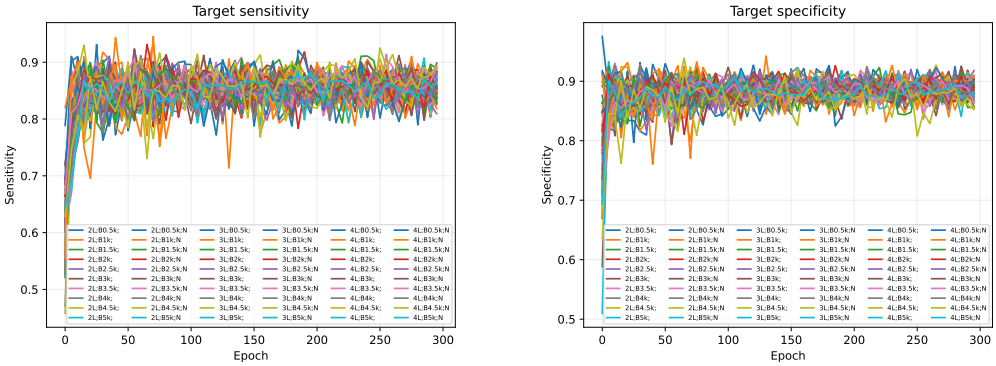
Figure 3.14: Comparison of the validation test data of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); normalized and not normalized; with 2 to 4 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

The one run, that tends to perform slightly better than the others in validation is the not normalized 3 layer batch size 1000 run, which is notable, as another run with the same settings already stood out in Figure 3.11. While it could still be a coincidence, there might be something about this specific setting.



(a) Accuracy target data

(b) Loss target data



(c) Sensitivity target data

(d) Specificity target data

Figure 3.15: Comparison of the target data on runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); normalized and not normalized; with 2 to 4 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

In the target data, there are no standouts.

For keeping track of the individual runs of this section, see Appendix D.

3.4.5 Dropout

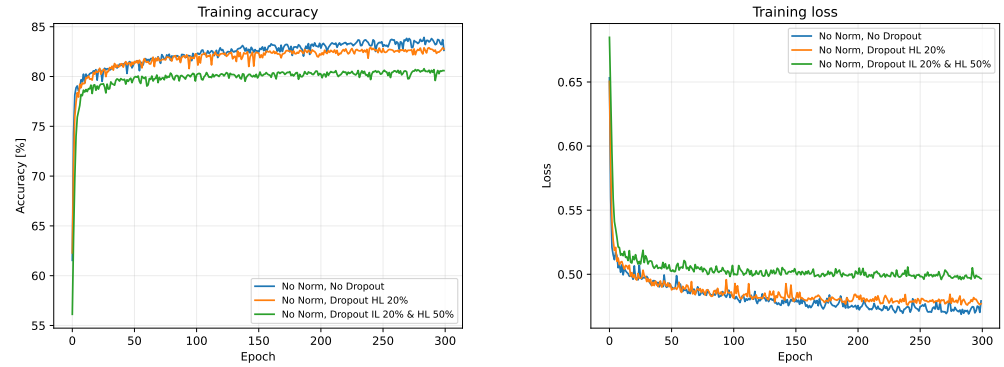
A process to prevent overfitting is called Dropout, it refers to a chance for any input of a node getting set to 0 for any event, and scales every output in training by $\frac{1}{1-p}$. (15)

Until now, 20% Dropout on every hidden layer has been used, but what about other Dropout settings?

Three different setting for Dropout are tested, one without, one with 20% Dropout for every hidden layer from (10) and one with 20% Dropout for the input layer and 50% for every hidden layer from (8).[1]

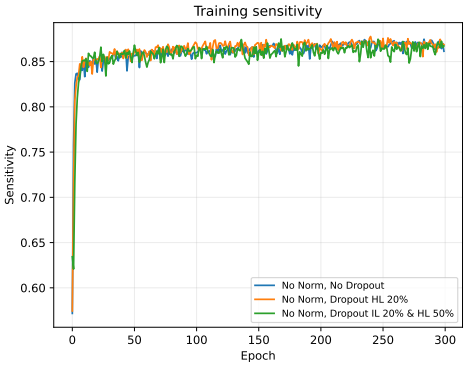
First without Normalization, then with Normalization.

[1] As the layers in PyTorch seem to define the connections between the nodes, instead of the nodes themselves, there is one hidden layer more, than directly described in the code, because of this, it is possible, that there is one hidden layer, which does not have dropout used on it

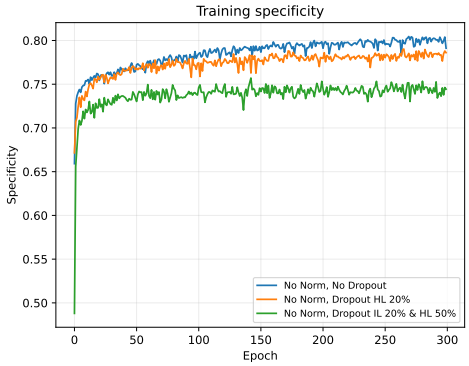


(a) Accuracy training data

(b) Loss training data



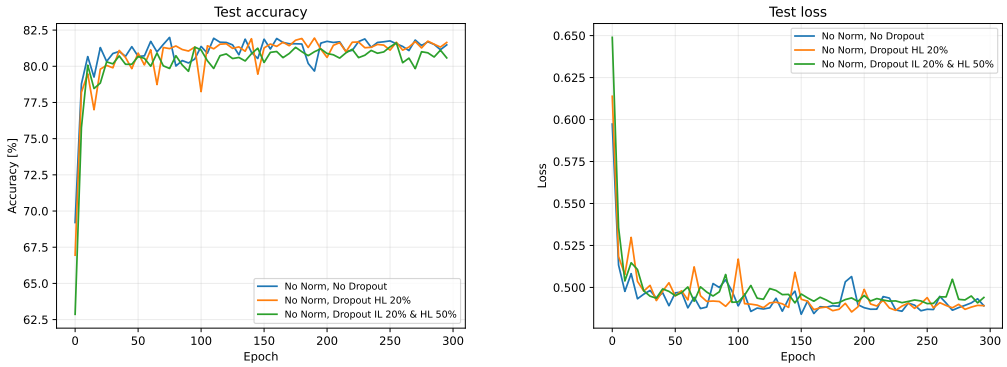
(c) Sensitivity training data



(d) Specificity training data

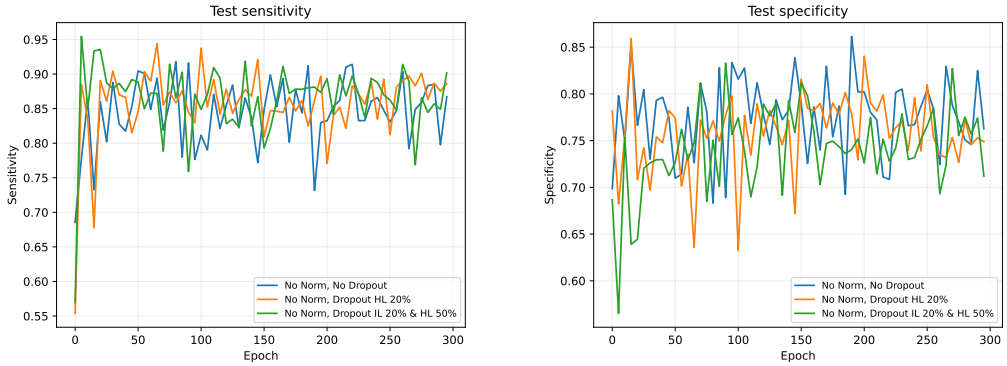
Figure 3.16: Comparison of the training data of runs with different dropout settings: no dropout (blue), hidden layers 20% dropout (orange), input layer 20% and hidden layers 50% dropout (green); not normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

Without Normalization, the stronger Dropout settings seems to lower accuracy and raise loss for the training data, mainly by lowering specificity with the more drastic Dropout being stronger.



(a) Accuracy validation test data

(b) Loss validation test data



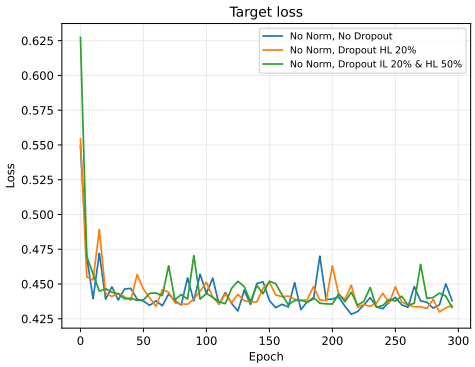
(c) Sensitivity validation test data

(d) Specificity validation test data

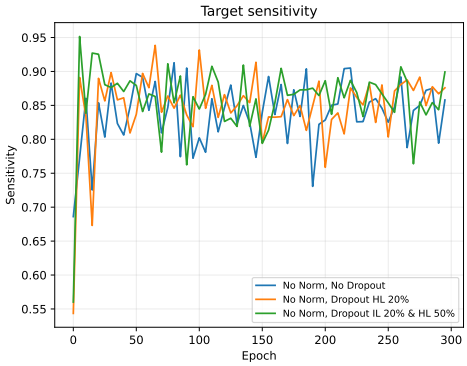
Figure 3.17: Comparison of the validation test data of runs with different dropout settings: no dropout (blue), hidden layers 20% dropout (orange), input layer 20% and hidden layers 50% dropout (green); not normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the validation test data, the different Dropout settings seem to not change much for the not normalized runs, in contrast to the training data, this is in line with the purpose of Dropout minimizing overfitting.

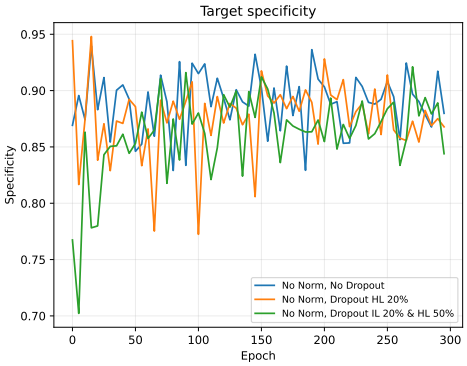
(a) Accuracy target data



(b) Loss target data



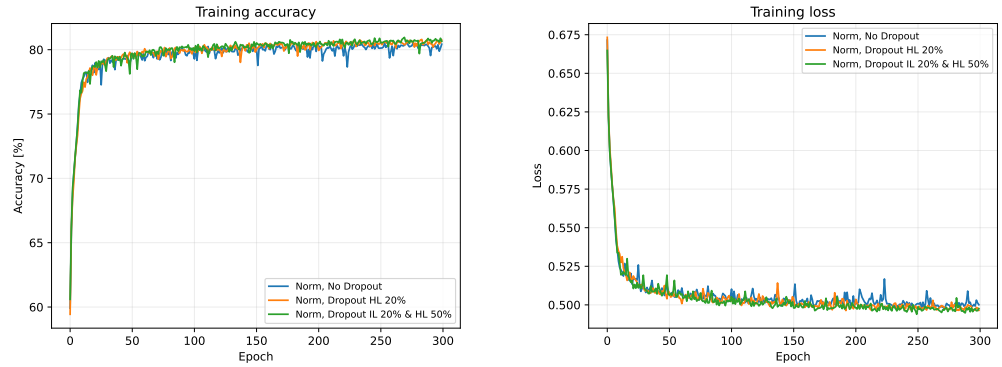
(c) Sensitivity target data



(d) Specificity target data

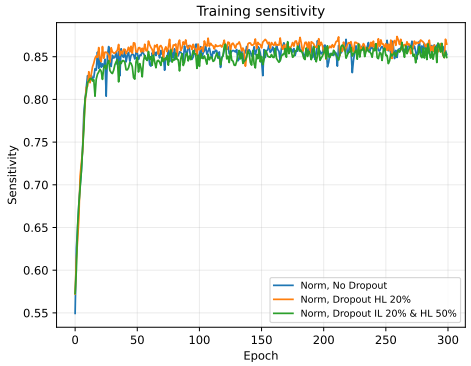
Figure 3.18: Comparison of the target data on runs with different dropout settings: no dropout (blue), hidden layers 20% dropout (orange), input layer 20% and hidden layers 50% dropout (green); not normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the target data, the different Dropout settings do not seem to change anything in performance for the not normalized runs.

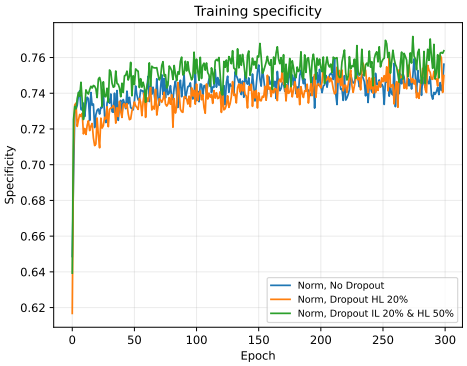


(a) Accuracy training data

(b) Loss training data



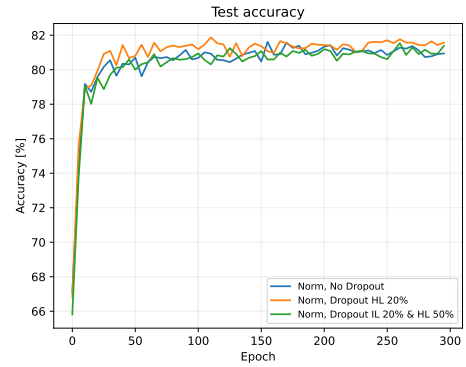
(c) Sensitivity training data



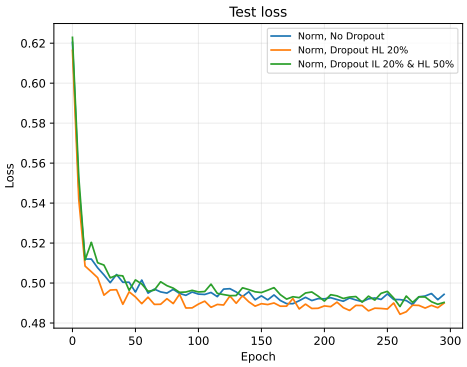
(d) Specificity training data

Figure 3.19: Comparison of the training data of runs with different dropout settings: no dropout (blue), hidden layers 20% dropout (orange), input layer 20% and hidden layers 50% dropout (green); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

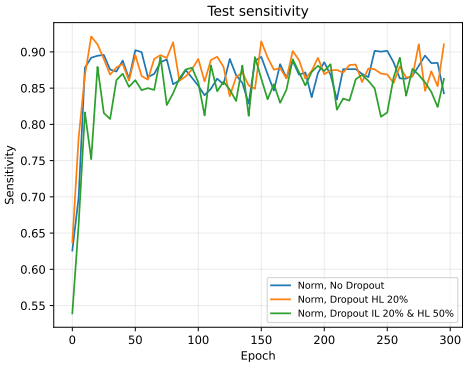
The normalized runs perform roughly equally in training data for different Dropout settings, with an improvement to sensitivity making up for a worse specificity.



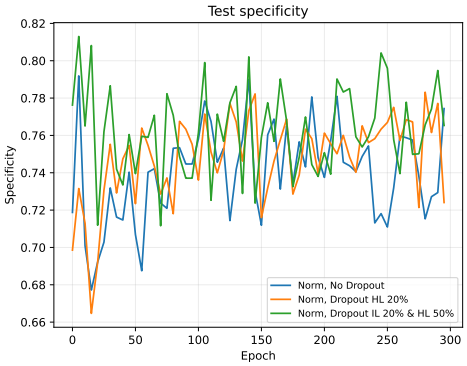
(a) Accuracy validation test data



(b) Loss validation test data



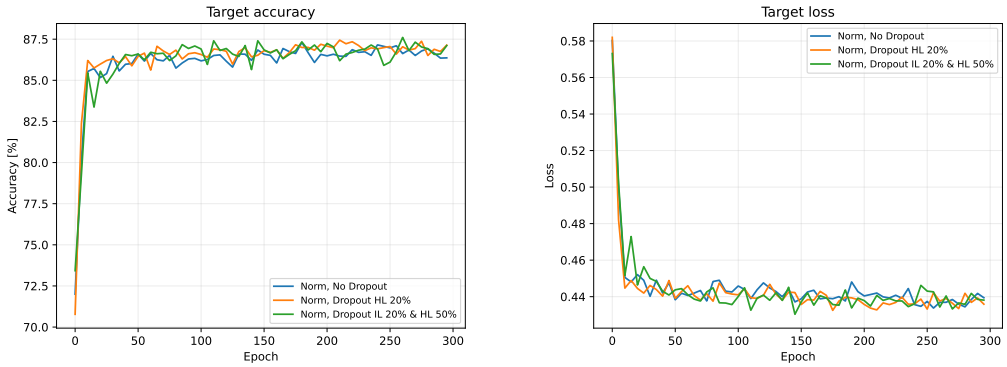
(c) Sensitivity validation test data



(d) Specificity validation test data

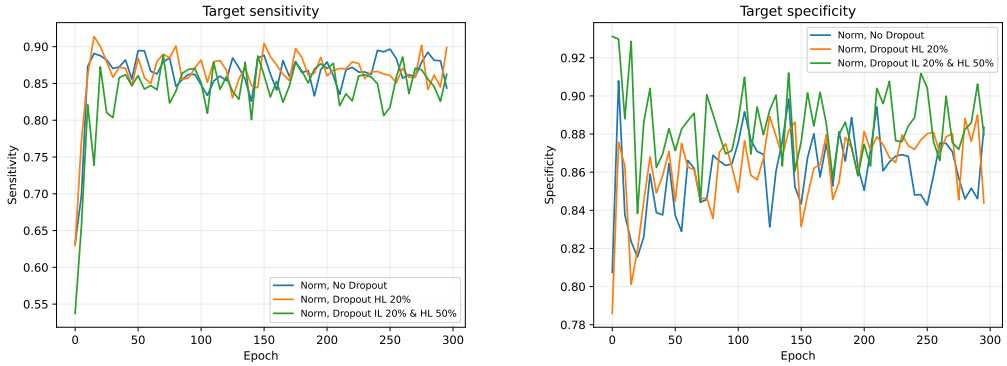
Figure 3.20: Comparison of the validation test data of runs with different dropout settings: no dropout (blue), hidden layers 20% dropout (orange), input layer 20% and hidden layers 50% dropout (green); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the validation test data, 20% Dropout for every hidden layer seems to perform better than the others for normalized runs.



(a) Accuracy target data

(b) Loss target data



(c) Sensitivity target data

(d) Specificity target data

Figure 3.21: Comparison of the target data on runs with different dropout settings: no dropout (blue), hidden layers 20% dropout (orange), input layer 20% and hidden layers 50% dropout (green); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

In target data, the normalized runs also perform equally for different Dropout settings.

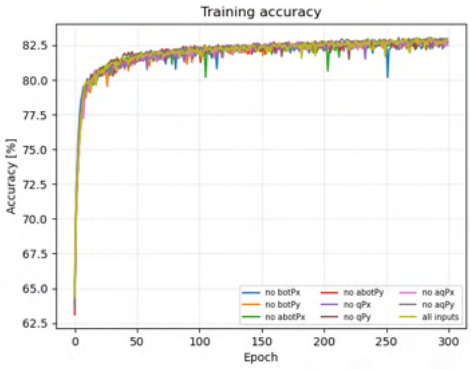
Based on this, the Dropout setting of 20% for hidden layers is kept, mainly because of its better performance in the validation test data for normalized runs seen in Figure 3.20.

3.5 Missing Input

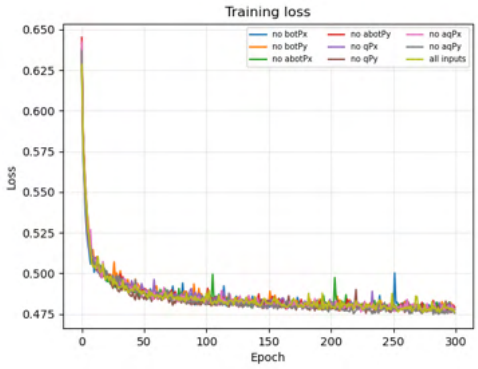
To measure, how much the specific inputs matter, runs without those inputs are made. For a better overview, the inputs are looked at category by category and compared to a run with all inputs.

3.5.1 JetP

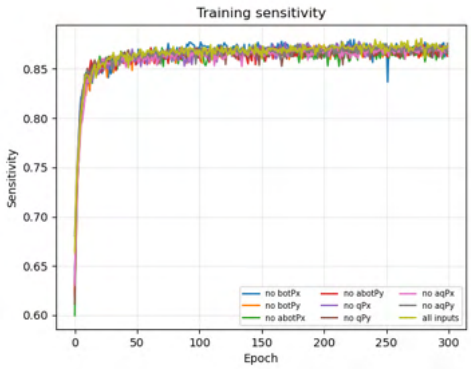
The inputs of the input group "JetP" are the x- and y-momenta of the bottom quark jet b, the antibottom quark jet $\bar{b}$, the quark jet from the hadronic W boson q and the antiquark jet from the hadronic W boson $\bar{q}$.



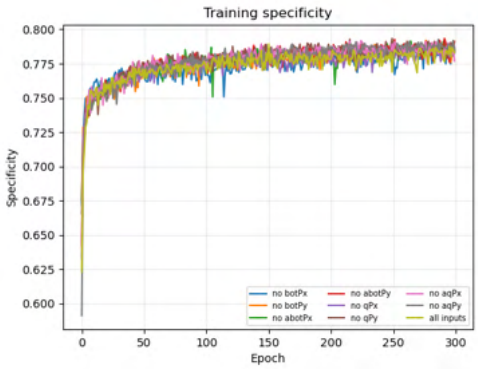
(a) Accuracy training data



(b) Loss training data



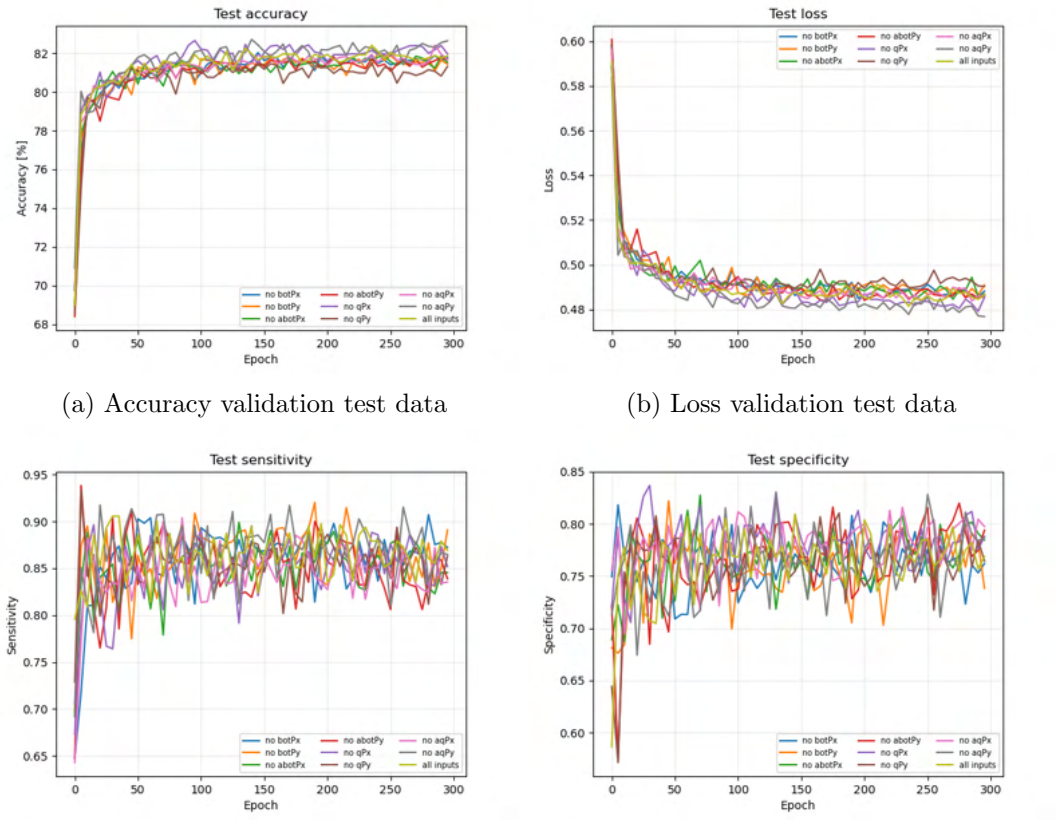
(c) Sensitivity training data



(d) Specificity training data

Figure 3.22: Comparison of the training data of runs with different inputs of the group JetP left out: x-momentum of b (blue), y-momentum of b (orange), x-momentum of $\bar{b}$ (dark green), y-momentum of $\bar{b}$ (red), x-momentum of q (purple), y-momentum of q (brown), x-momentum of $\bar{q}$ (pink), y-momentum of $\bar{q}$ (grey) and none (olive green); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For training data, no noticeable standout can be found among the missing JetP inputs, neither from each other, nor from the run without missing inputs.



(a) Accuracy validation test data

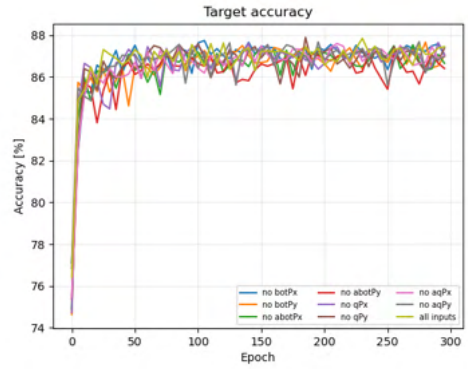
(b) Loss validation test data

(c) Sensitivity validation test data

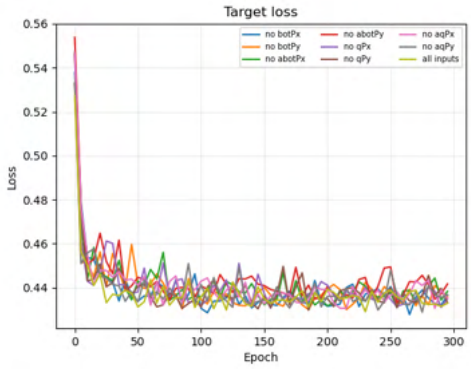
(d) Specificity validation test data

Figure 3.23: Comparison of the validation test data of runs with different inputs of the group JetP left out: x-momentum of b (blue), y-momentum of b (orange), x-momentum of $\bar{b}$ (dark green), y-momentum of $\bar{b}$ (red), x-momentum of q (purple), y-momentum of q (brown), x-momentum of $\bar{q}$ (pink), y-momentum of $\bar{q}$ (grey) and none (olive green); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

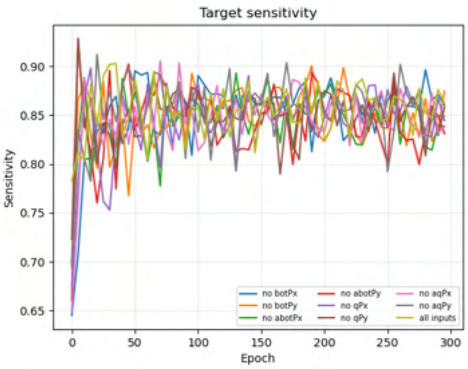
For validation test data, the situation is not much different from the training data, while single runs seem to perform a bit better or worse than the others, the difference is more likely to be a result of the smaller size of the validation test data, than anything about the input data, seen most clearly by the feature, that some runs performed better than the all inputs control case by a similar amount, as the worse case performed worse.



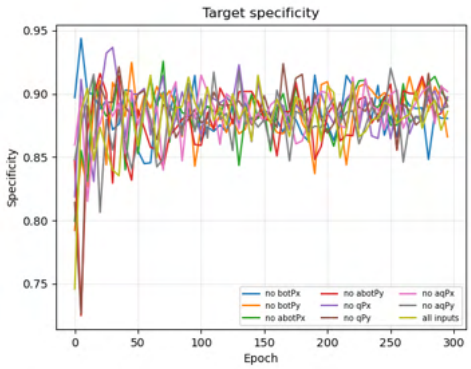
(a) Accuracy target data



(b) Loss target data



(c) Sensitivity target data



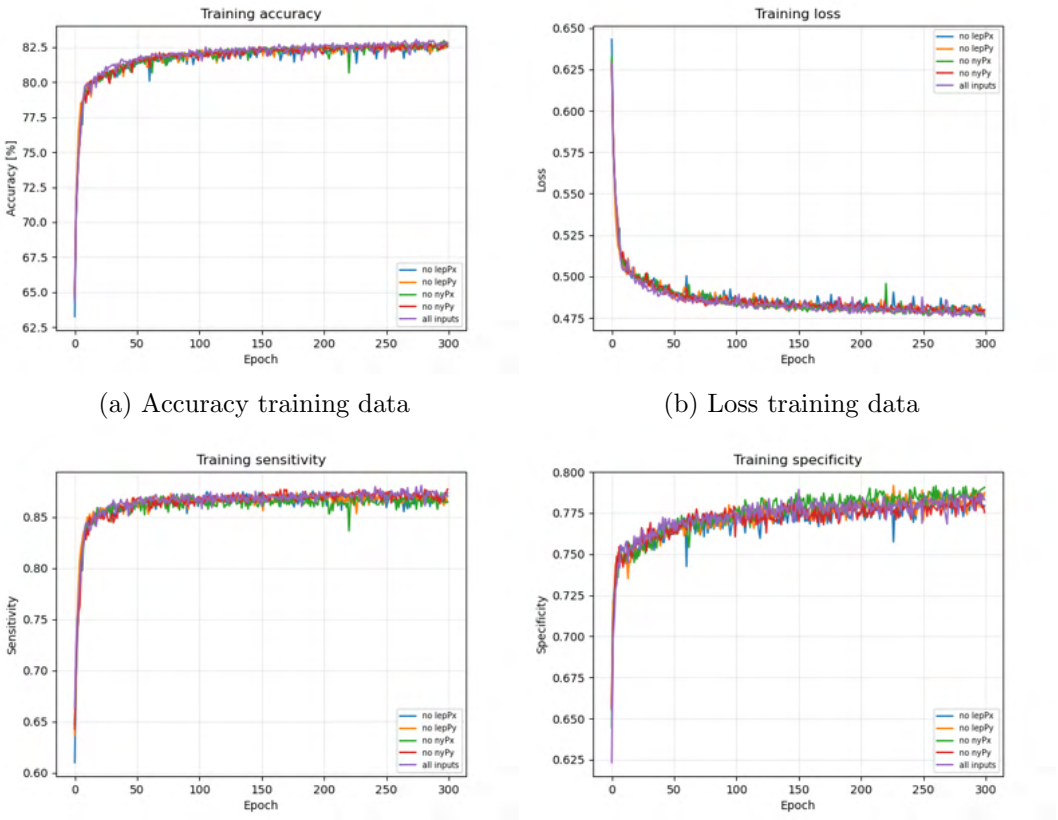
(d) Specificity target data

Figure 3.24: Comparison of the target data on runs with different inputs of the group JetP left out: x-momentum of b (blue), y-momentum of b (orange), x-momentum of $\bar{b}$ (dark green), y-momentum of $\bar{b}$ (red), x-momentum of q (purple), y-momentum of q (brown), x-momentum of $\bar{q}$ (pink), y-momentum of $\bar{q}$ (grey) and none (olive green); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

The target data also seems to have no standout runs.

3.5.2 TruthP

The inputs of the input group "TruthP" are the x- and y-momenta of the lepton truth particle from the leptonic W boson l and the neutrino truth particle from the leptonic W boson ν.



(a) Accuracy training data

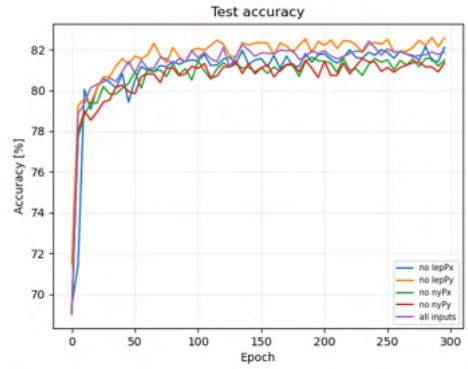
(b) Loss training data

(c) Sensitivity training data

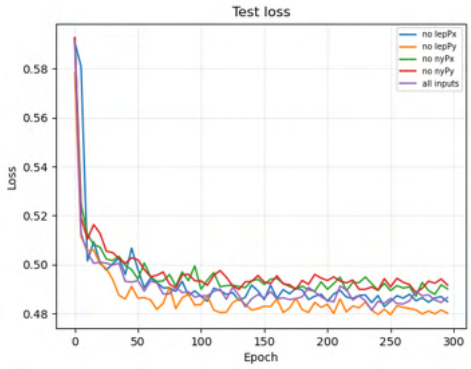
(d) Specificity training data

Figure 3.25: Comparison of the training data of runs with different inputs of the group TruthP left out: x-momentum of l (blue), y-momentum of l (orange), x-momentum of ν (green), y-momentum of ν (red) and none (purple); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

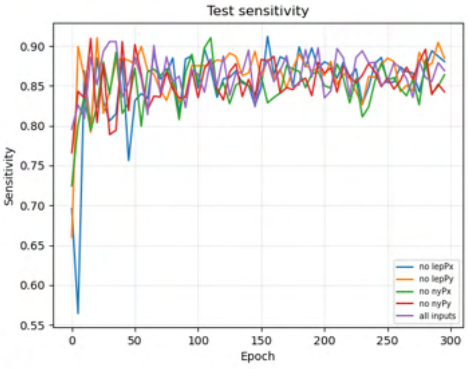
For the training data, the most that stands out, is, that the run with out x-momentum of ν seems to have a bit lower sensitivity and higher specificity than the others. In terms of performance nothing stands out as better or worse than the rest.



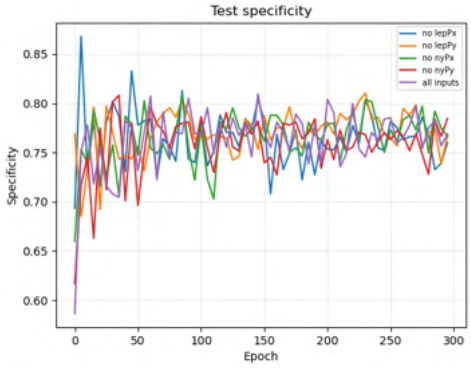
(a) Accuracy validation test data



(b) Loss validation test data



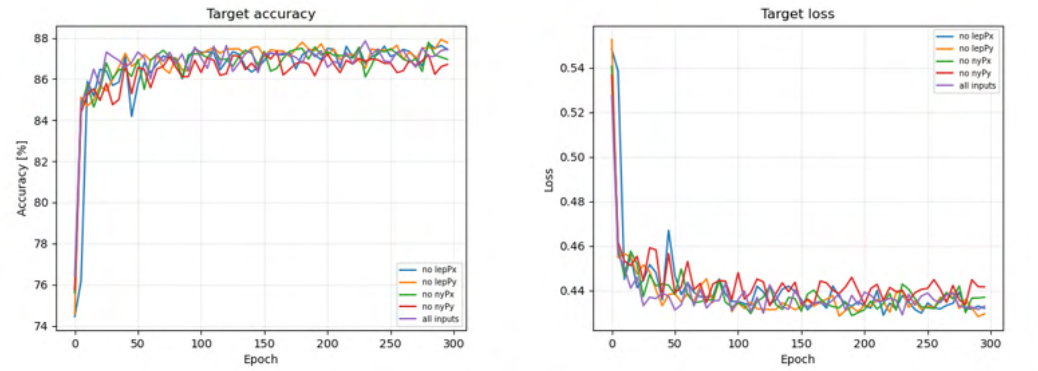
(c) Sensitivity validation test data



(d) Specificity validation test data

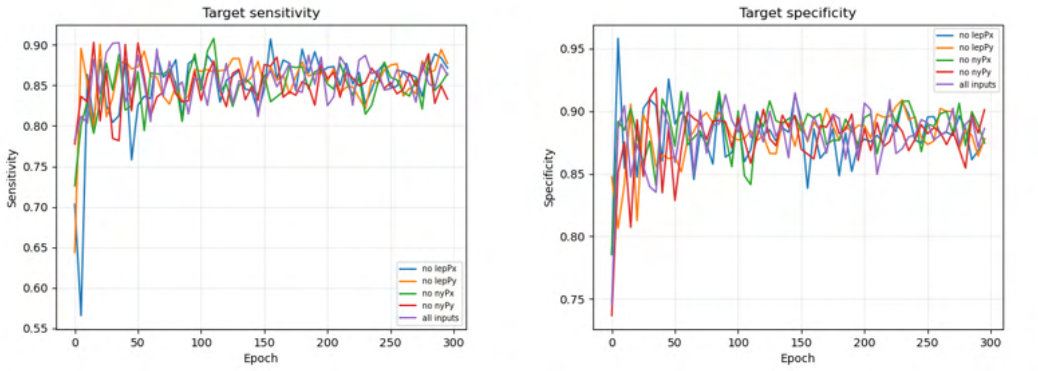
Figure 3.26: Comparison of the validation test data of runs with different inputs of the group TruthP left out: x-momentum of l (blue), y-momentum of l (orange), x-momentum of ν (green), y-momentum of ν (red) and none (purple); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the validation test data, again, while there is seemingly enough difference in the performance of the runs to single out single runs, this is likely just the result of the validation test data being smaller, especially, as the control run lies roughly in the middle. However, it is noticeable here, that the missing ν momenta runs both performed worse than the all inputs control run, while of the missing l momenta runs, one performed better and the other roughly equal to the control run. So, while it is likely just random chance, there is a possibility, that the ν momenta are a bit more important than the l momenta.



(a) Accuracy target data

(b) Loss target data



(c) Sensitivity target data

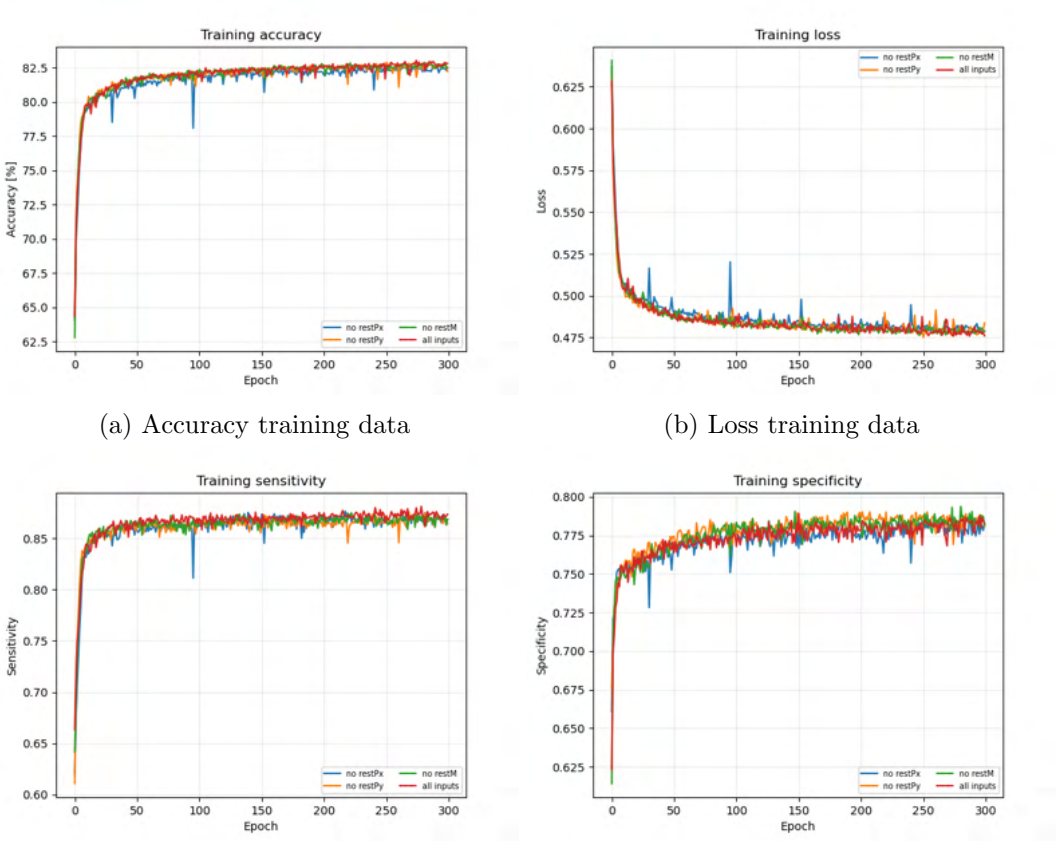
(d) Specificity target data

Figure 3.27: Comparison of the target data on runs with different inputs of the group TruthP left out: x-momentum of l (blue), y-momentum of l (orange), x-momentum of ν (green), y-momentum of ν (red) and none (purple); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the target data, the run performance again seems to have no clear order.

3.5.3 Rest

The inputs of the input group "Rest" are the x- and y-momenta, as well as the mass of the sum of all the jets, that are not matched to a particle.



(a) Accuracy training data

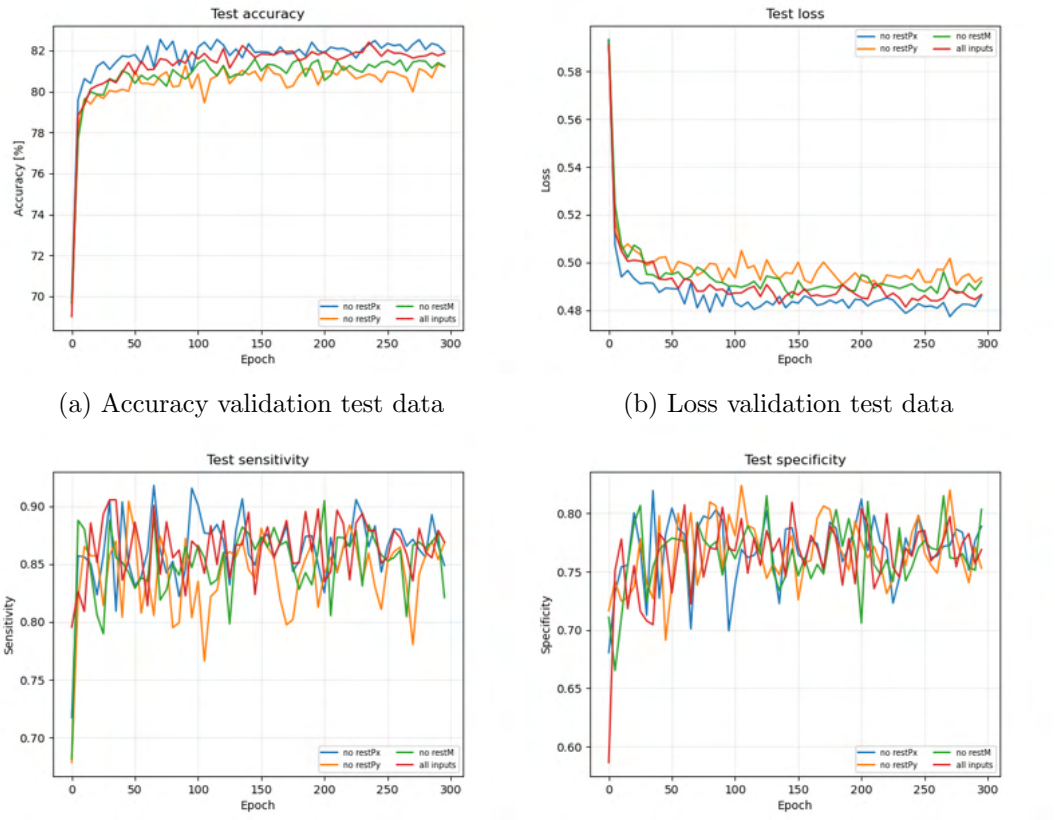
(b) Loss training data

(c) Sensitivity training data

(d) Specificity training data

Figure 3.28: Comparison of the training data of runs with different inputs of the group Rest left out: x-momentum of the sum of the unmatched jets (blue), y-momentum of the sum of the unmatched jets (orange), mass of the sum of the unmatched jets (green) and none (red); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the training data, there are no runs standing out.



(a) Accuracy validation test data

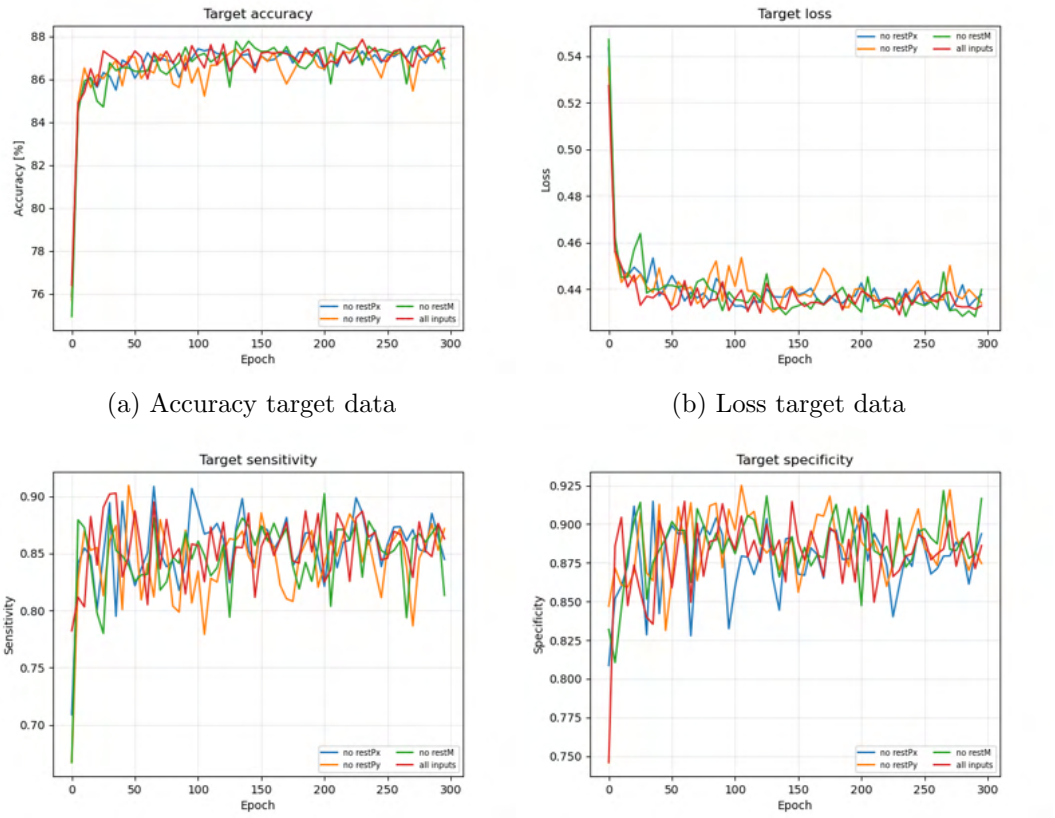
(b) Loss validation test data

(c) Sensitivity validation test data

(d) Specificity validation test data

Figure 3.29: Comparison of the validation test data of runs with different inputs of the group Rest left out: x-momentum of the sum of the unmatched jets (blue), y-momentum of the sum of the unmatched jets (orange), mass of the sum of the unmatched jets (green) and none (red); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the validation test data, the order of performance from best to worst seems to be: missing x-momentum, all inputs, missing mass and missing y-momentum. With the control all input run not performing best and the best and worst being the x and y momentum, this is likely just random chance.



(a) Accuracy target data

(b) Loss target data

(c) Sensitivity target data

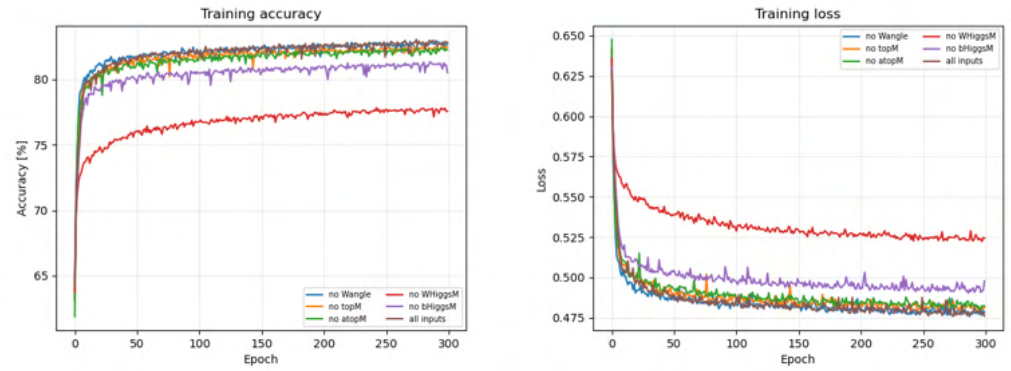
(d) Specificity target data

Figure 3.30: Comparison of the target data on runs with different inputs of the group Rest left out: x-momentum of the sum of the unmatched jets (blue), y-momentum of the sum of the unmatched jets (orange), mass of the sum of the unmatched jets (green) and none (red); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the target data, any difference in performance seems to vanish, again.

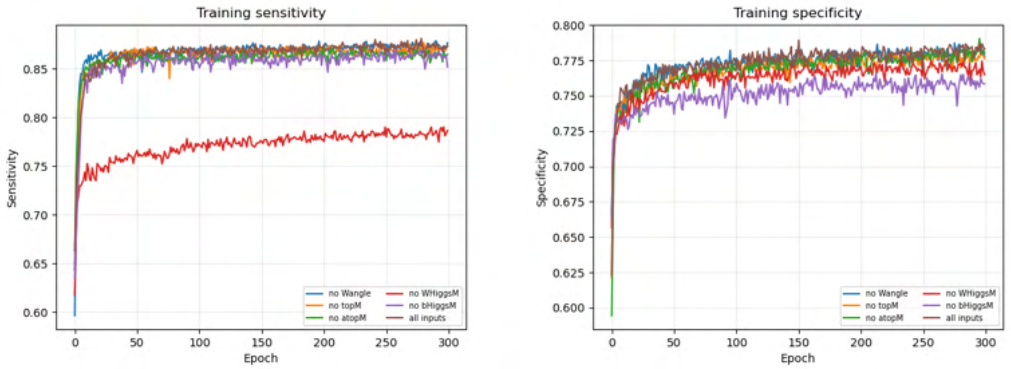
3.5.4 Indicators

The inputs of the input group "Indicators" are the angle between the W bosons in the HH or $t\bar{t}$ rest frame respectively, the W^+b pair mass, the $W^-\bar{b}$ pair mass, the W^+W^- pair mass and the $b\bar{b}$ pair mass.



(a) Accuracy training data

(b) Loss training data



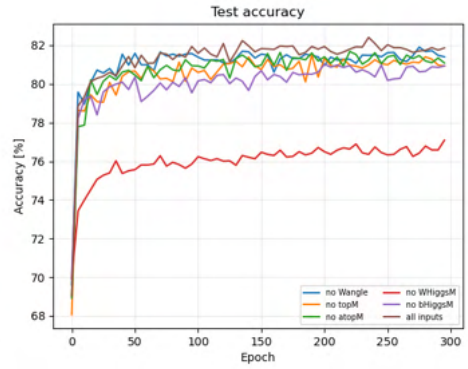
(c) Sensitivity training data

(d) Specificity training data

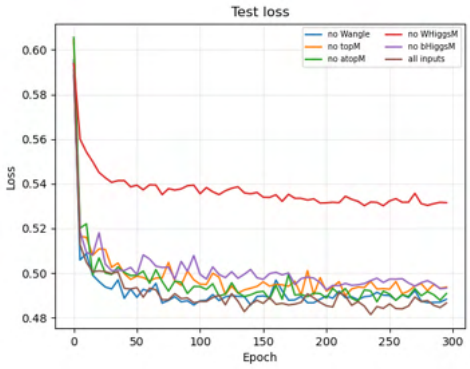
Figure 3.31: Comparison of the training data of runs with different inputs of the group Indicators left out: angle of the W bosons in the HH or $t\bar{t}$ rest frame respectively (blue), W^+b pair mass (orange), $W^-\bar{b}$ pair mass (green), W^+W^- pair mass (red), $b\bar{b}$ pair mass (purple) and none (brown); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

In contrast to the other input groups, for "Indicators", missing single inputs seems to significantly affect the performance for training data. The run performing by far the worst, is the run missing the W^+W^- pair mass, with the run missing the $b\bar{b}$ pair mass being a distant, but still significant second. The runs missing the W^+b pair mass and $W^-\bar{b}$ pair mass respectively, perform only slightly but still noticeable worse than control, with only the run missing the angle between the W bosons having no noticeable performance drop the control with all inputs.

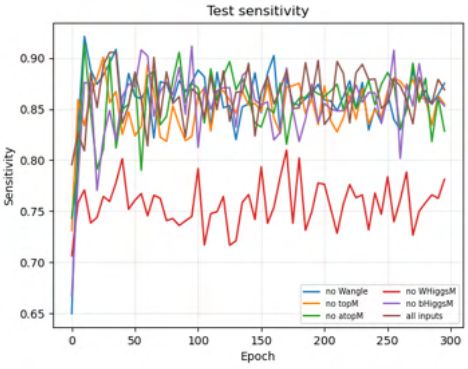
Looking at the sensitivity and specificity, it becomes apparent, that missing the W^+W^- pair mass, mainly makes the sensitivity far worse and with that makes it much harder for the neural network to correctly detect HH-events, while missing the $b\bar{b}$ pair mass mainly has a noticeable impact on the specificity, making the neural network less reliable in excluding false positives for HH-events.



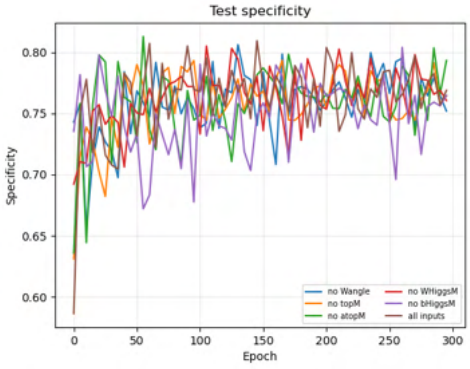
(a) Accuracy validation test data



(b) Loss validation test data



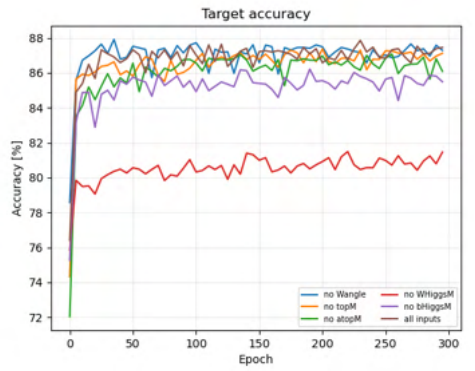
(c) Sensitivity validation test data



(d) Specificity validation test data

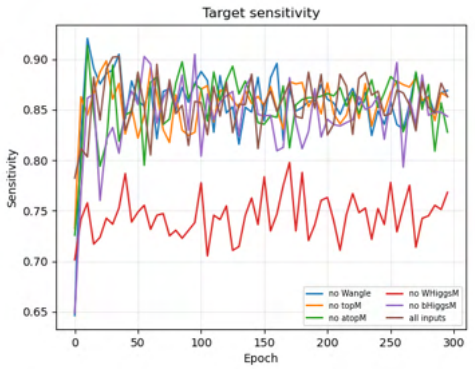
Figure 3.32: Comparison of the validation test data of runs with different inputs of the group Indicators left out: angle of the W bosons in the HH or $t\bar{t}$ rest frame respectively (blue), W^+b pair mass (orange), $W^-\bar{b}$ pair mass (green), W^+W^- pair mass (red), $b\bar{b}$ pair mass (purple) and none (brown); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For validation test data, there are no changes to the order of performance from the training data, though with the exception of the run missing the W^+W^- pair mass, the spread is similar to the one in the validation test data of the other groups.

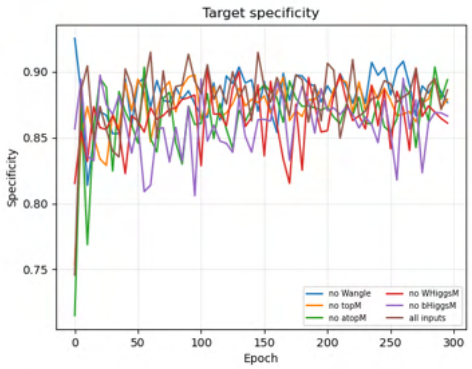


(a) Accuracy target data

(b) Loss target data



(c) Sensitivity target data



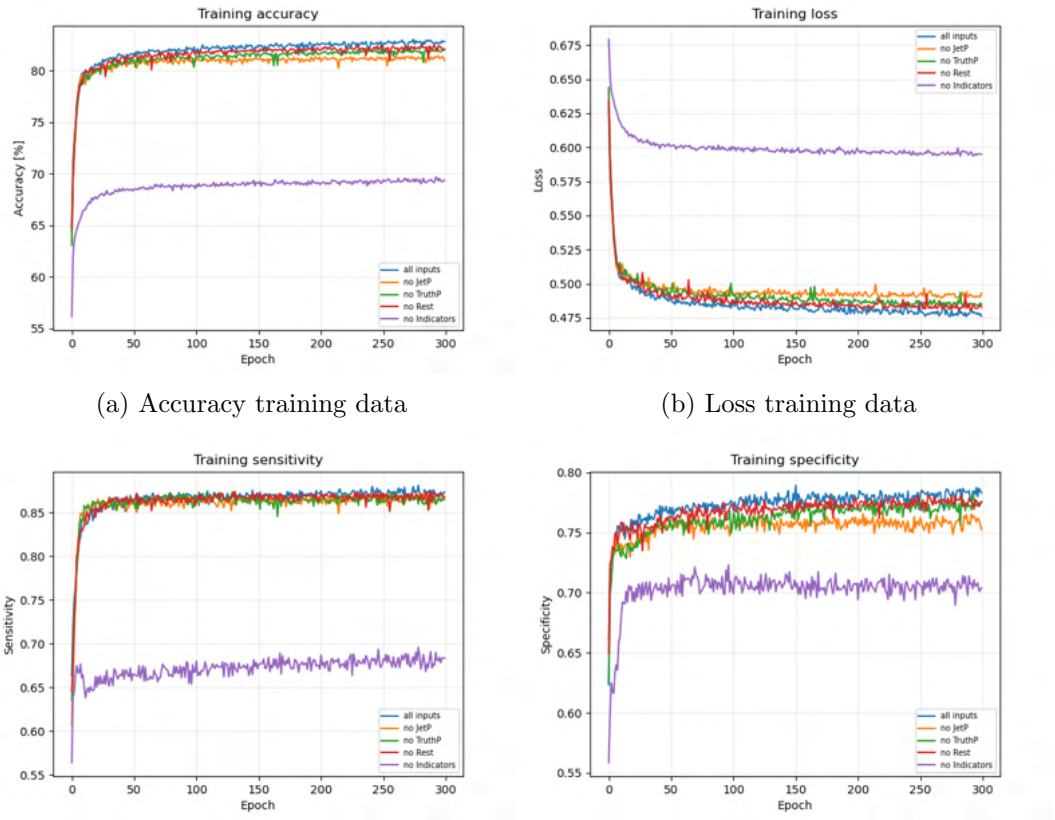
(d) Specificity target data

Figure 3.33: Comparison of the target data on runs with different inputs of the group Indicators left out: angle of the W bosons in the HH or $t\bar{t}$ rest frame respectively (blue), W^+b pair mass (orange), $W^-\bar{b}$ pair mass (green), W^+W^- pair mass (red), $b\bar{b}$ pair mass (purple) and none (brown); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For the target data, again the order of performance is the same as it was for validation test data and training data, though interestingly, the run missing the $b\bar{b}$ pair mass does stand out a bit more, than it did in the validation test data.

3.5.5 Groups

To check, if the negligible impact of missing single inputs in every category except "Indicators" is the result of multiple inputs in the same input group being redundant, there were also runs made, with entire input groups missing.



(a) Accuracy training data

(b) Loss training data

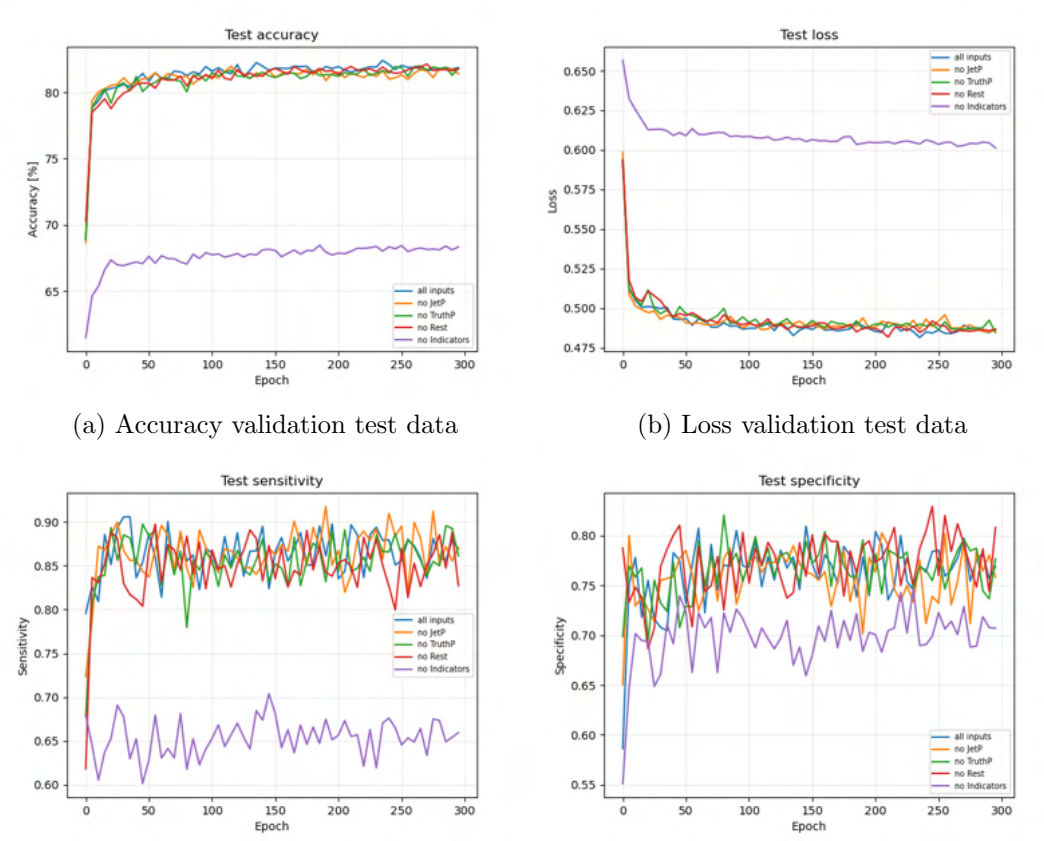
(c) Sensitivity training data

(d) Specificity training data

Figure 3.34: Comparison of the training data on runs with different groups of inputs left out: none (blue), JetP (orange), TruthP (green), Rest (red) and Indicators (purple); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

For training data, a clear order can be seen in the performance. From worst to best: missing Indicators, missing JetP, missing TruthP, missing Rest and the all inputs control run. However, as with the exception of Indicators, which already showed its large importance in Section 3.5.4, this order seems to follow the number of inputs in each group, with 8 in JetP, 4 in TruthP and 3 in Rest.

This might possibly be a sign for this difference to simply be a result of overfitting, as, the more inputs there are for a limited dataset, the more chances there could be for patterns between them seemingly emerging in random fluctuations for the neural network to pick up on.



(a) Accuracy validation test data

(b) Loss validation test data

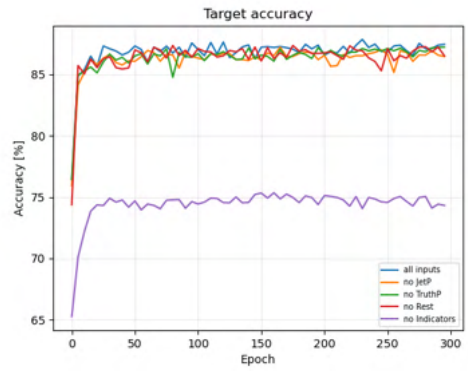
(c) Sensitivity validation test data

(d) Specificity validation test data

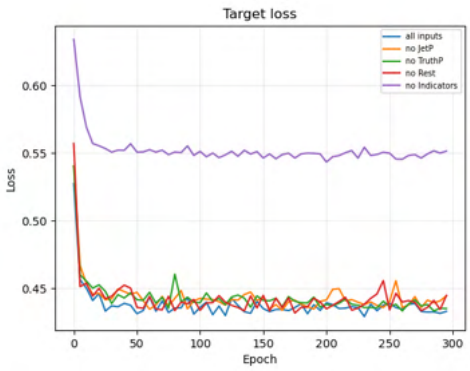
Figure 3.35: Comparison of the validation test data on runs with different groups of inputs left out: none (blue), JetP (orange), TruthP (green), Rest (red) and Indicators (purple); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

Looking at the validation test data, seems to speak for the hypothesis of the trainings data being a sign of overfitting, as, with the exception of the run missing Indicators, the differences between the different missing group runs and the all inputs control run seems to vanish, with the performances being overall closer together, than for the training data, making this being simply a result of the validation test data being smaller, unlikely.

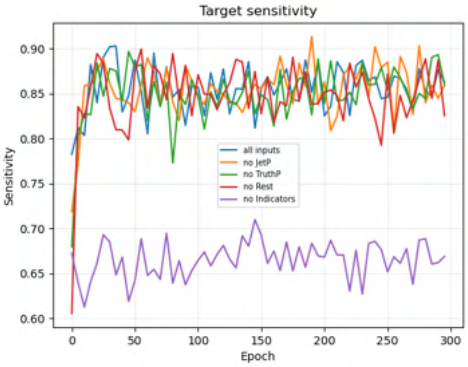
Outside of that, it is interesting to note, that the gap to the run missing Indicators is so large, that it seems to converge at a worse performance, than the other runs start at.



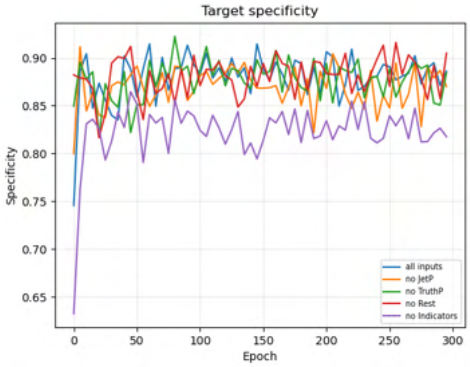
(a) Accuracy target data



(b) Loss target data



(c) Sensitivity target data



(d) Specificity target data

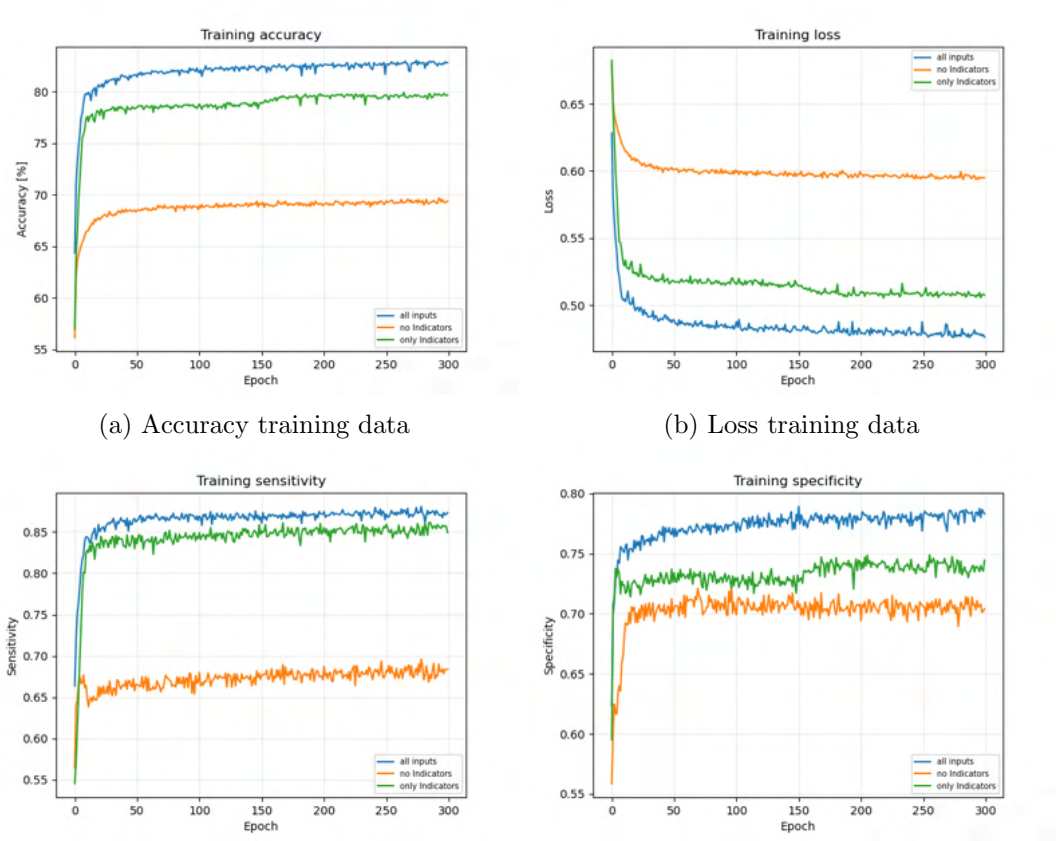
Figure 3.36: Comparison of the target data on runs with different groups of inputs left out: none (blue), JetP (orange), TruthP (green), Rest (red) and Indicators (purple); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

The target data looks basically the same as the validation test data. The run missing Indicators performs much worse than the others, which perform roughly the same.

3.5.6 Only Indicators

The lack of showing any decrease in performance from missing the groups other than Indicators in Section 3.5.5, raises the question, if these other inputs are completely superfluous, when the Indicators inputs are used.

To check if the other inputs even have any impact on the performance, a run is made, which uses only the Indicator inputs and is compared with the run missing the Indicator inputs and the control run with all inputs.



(a) Accuracy training data

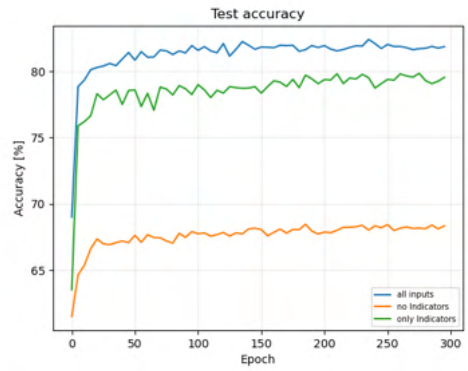
(b) Loss training data

(c) Sensitivity training data

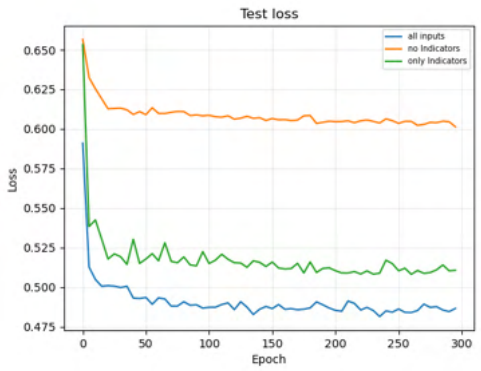
(d) Specificity training data

Figure 3.37: Comparison of the training data of runs trained on: all inputs (blue), all inputs except Indicators (orange) and only Indicators (green); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

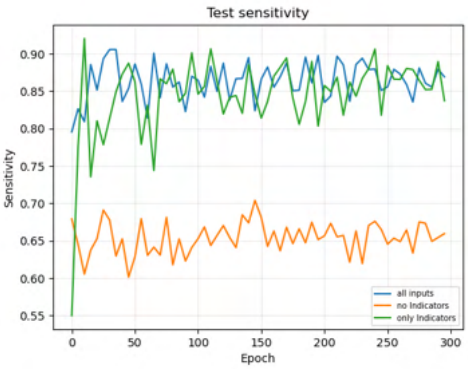
Looking at the training data, using only the Indicators inputs does seem to perform significantly worse than the control run, but not as bad, as missing the Indicator inputs. However, as Section 3.5.5 has shown, this could simply be the result of the run with only Indicators having 15 inputs less to overfit on.



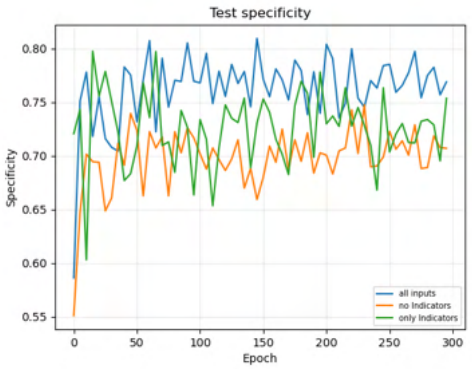
(a) Accuracy validation test data



(b) Loss validation test data



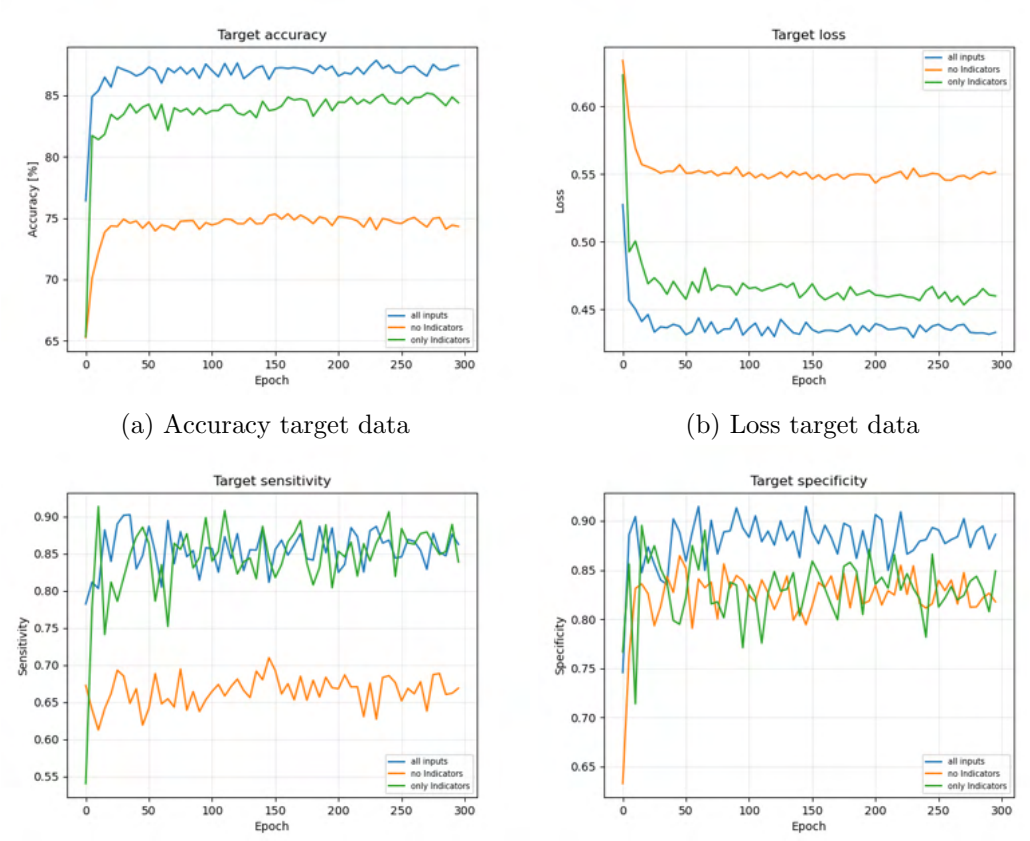
(c) Sensitivity validation test data



(d) Specificity validation test data

Figure 3.38: Comparison of the validation test data of runs trained on: all inputs (blue), all inputs except Indicators (orange) and only Indicators (green); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

The validation test data shows, that the worse performance of the run with only Indicators, is not just a result of overfitting, but the 15 other inputs collectively do have a real impact on the performance.



(a) Accuracy target data

(b) Loss target data

(c) Sensitivity target data

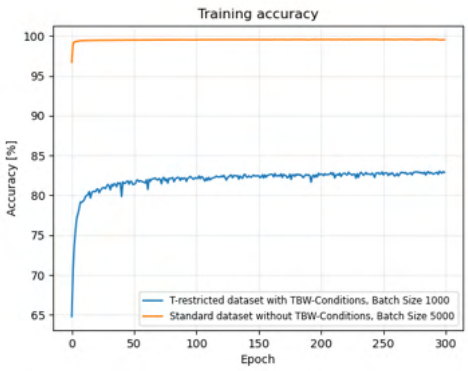
(d) Specificity target data

Figure 3.39: Comparison of the target data on runs trained on: all inputs (blue), all inputs except Indicators (orange) and only Indicators (green); normalized, with batch size 1000 and 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

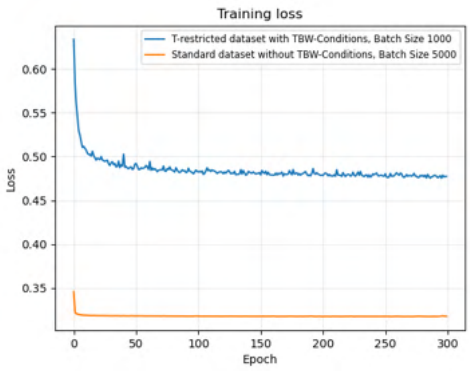
The target data results are basically the same as the validation test data. That the impact, that the 16 other inputs collective have, translates to the target data, shows, that they are not completely superfluous, and should be kept for now.

3.6 Evaluation

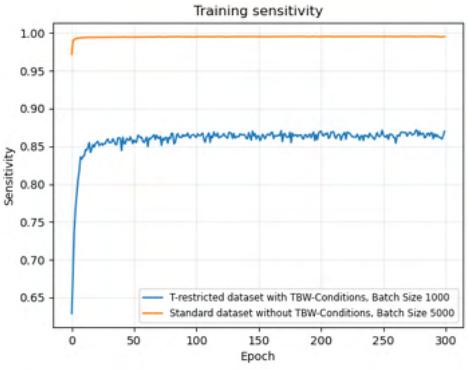
Before looking at the performance of a neural network trained on a T-restricted smeared jet reconstructed $t\bar{t}$-dataset in absolute terms, it is helpful to compare it to a neural network trained on a jet reconstructed standard $t\bar{t}$-dataset as control point. The size of the datasets used are 627,364, which is the size of the full HH dataset, minus the 10k events reserved for the target data.



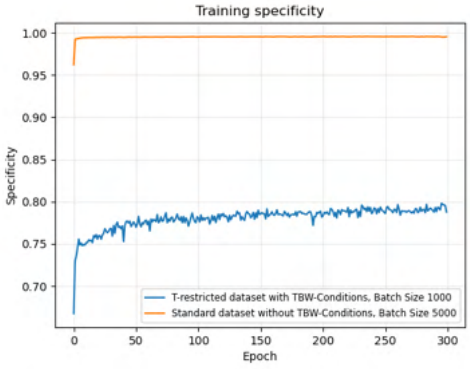
(a) Accuracy training data



(b) Loss training data



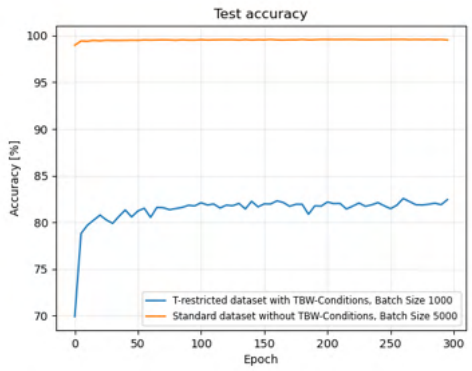
(c) Sensitivity training data



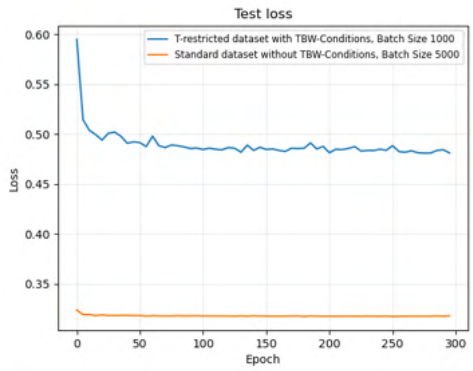
(d) Specificity training data

Figure 3.40: Comparison of the training data of a run trained using a T restricted $t\bar{t}$ dataset after applying the TBW-conditions and batch size 1000 (blue) with a run trained using a standard $t\bar{t}$ dataset with batch size 5000 (orange); Both runs are normalized and have 3 hidden layers. All datasets are smeared and reconstructed.

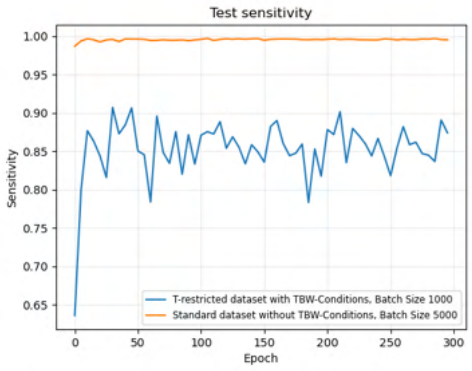
As the two training datas are completely different, any comparison between the two is not really helpful, however, the control standard settings neural network performs almost perfectly, with an over 99% accuracy rate, while the T-restricted neural network still has potential room to be improved, with an accuracy of ca. 83%.



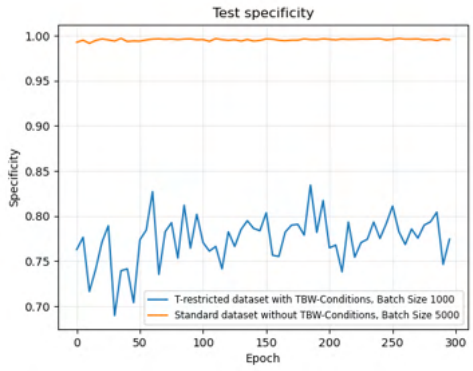
(a) Accuracy validation test data



(b) Loss validation test data



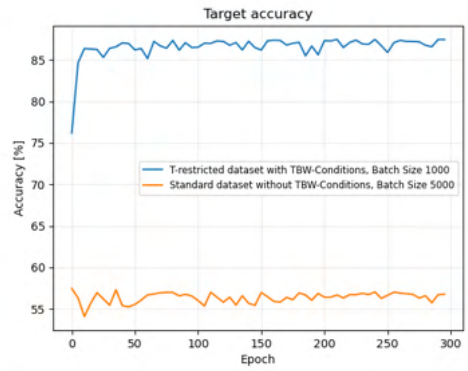
(c) Sensitivity validation test data



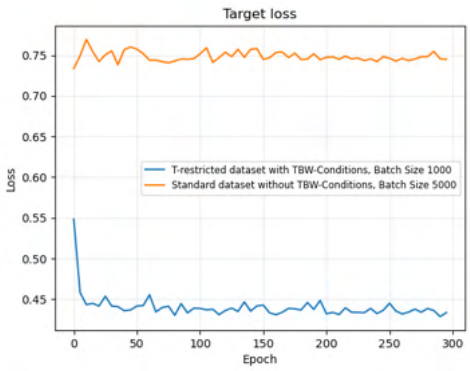
(d) Specificity validation test data

Figure 3.41: Comparison of the validation test data of a run trained using a T restricted $t\bar{t}$ dataset after applying the TBW-conditions and batch size 1000 (blue) with a run trained using a standard $t\bar{t}$ dataset with batch size 5000 (orange); both are normalized and have 3 hidden layers. All datasets are smeared and reconstructed.

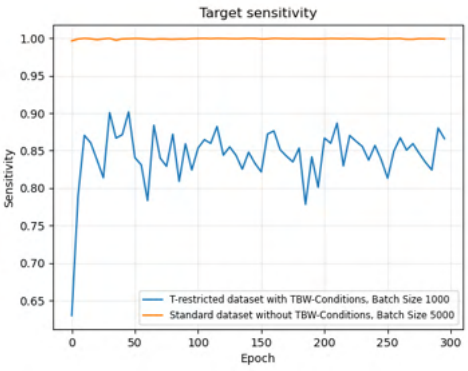
Not much changes between the training data and test data, but it shows, that the results of the training data are real.



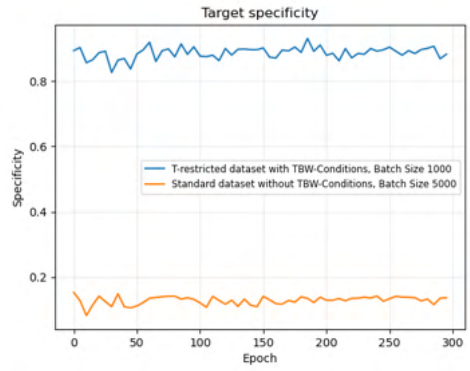
(a) Accuracy target data



(b) Loss target data



(c) Sensitivity target data



(d) Specificity target data

Figure 3.42: Comparison of the target data of a run trained using a T restricted $t\bar{t}$ dataset after applying the TBW-conditions and batch size 1000 (blue) with a run trained using a standard $t\bar{t}$ dataset with batch size 5000 (orange); both are normalized and have 3 hidden layers. All datasets are smeared and reconstructed.

As the target data is based on the same dataset for both, a direct comparison can be made, the only caveat being, that, as the dataset the target data is sourced from is the same jet reconstructed standard $t\bar{t}$-dataset, the standard settings neural network could perform a bit better than it actually would. But even if this was the case, in the target data, the standard dataset neural network ends up with an accuracy of ca. 57%, while the T-restricted dataset neural network performs much better, ending up with an accuracy of ca. 87%.

The reason for the worse performance of the standard dataset run at the target data seems to be, that in standard settings datasets the target area is dominated by the HH-events, leading to the standard settings neural network classifying many $t\bar{t}$-events in the target data as HH-events. This can also be seen in the sensitivity being at almost 1 and the specificity being around 0.13, for the target data of the standard dataset run.

Meanwhile, the T-restricted dataset run ends with a sensitivity (aka the rate of HH-events correctly identified) of around 0.87 and a specificity (aka the rate of $t\bar{t}$-events correctly identified) of around 0.89 for the target data. Transferring this to the ratio of the events in reality:

	HH	$t\bar{t}$
Cross section times branching ratio Literature	10.0fb	873.4pb
Cross section times branching ratio MadGraph	253.3ab	455.8pb
TBW after Rec and Smear	21.98%	0.132%
Target Performance	0.87	0.89

Table 3.3: Overview of the stats relevant for calculating the expected performance of the neural network trained on a T-restricted dataset, when used on real events in the target area.

Two scenarios are considered, one based on the cross sections and branching ratios found in Literature (7)(12)(14) seen in Chapter 1 and one based on the cross sections times branching ratio calculated by MadGraph seen in Chapter 2.

Accuracy then is:

$$\begin{aligned} accuracy =& \frac{t_p + t_n}{t_p + t_n + f_p + f_n} = \frac{t_p + f_n}{t_p + t_n + f_p + f_n} \frac{t_p}{t_p + f_n} + \frac{t_n + f_p}{t_p + t_n + f_p + f_n} \frac{t_n}{t_n + f_p} \\ =& \left(1 - \frac{t_n + f_p}{t_p + t_n + f_p + f_n}\right) \frac{t_p}{t_p + f_n} + \frac{t_n + f_p}{t_p + t_n + f_p + f_n} \frac{t_n}{t_n + f_p} \\ =& \left(1 - \frac{n_{t\bar{t}}}{n_{HH} + n_{t\bar{t}}}\right) sensitivity + \frac{n_{t\bar{t}}}{n_{HH} + n_{t\bar{t}}} specificity \\ =& \left(1 - \frac{\frac{n_{t\bar{t}}}{n_{HH}}}{1 + \frac{n_{t\bar{t}}}{n_{HH}}}\right) sensitivity + \frac{\frac{n_{t\bar{t}}}{n_{HH}}}{1 + \frac{n_{t\bar{t}}}{n_{HH}}} specificity \end{aligned} \tag{3.11}$$

While the share of actual HH-events among events classified by the neural network as HH-events is:

$$\frac{t_p}{t_p + f_p} = \frac{n_{HH} sensitivity}{n_{HH} sensitivity + n_{t\bar{t}}(1 - specificity)} = \frac{sensitivity}{sensitivity + \frac{n_{t\bar{t}}}{n_{HH}}(1 - specificity)} \tag{3.12}$$

And the share of actual $t\bar{t}$-events among events classified by the neural network as $t\bar{t}$-events is:

$$\frac{t_n}{t_n + f_n} = \frac{n_{t\bar{t}} specificity}{n_{HH}(1 - sensitivity) + n_{t\bar{t}} specificity} = \frac{\frac{n_{t\bar{t}}}{n_{HH}} specificity}{(1 - sensitivity) + \frac{n_{t\bar{t}}}{n_{HH}} specificity} \tag{3.13}$$

In both cases is:

$$\frac{n_{t\bar{t}}}{n_{HH}} = \frac{\sigma_{t\bar{t}} p_{TBW,t\bar{t}}}{\sigma_{HH} p_{TBW,HH}} \tag{3.14}$$

with σ being the respective cross section times branching ratio and p_{TBW} being the respective percentage of events, that pass the TBW-conditions after smearing and jet reconstruction.

	Accuracy	$\frac{t_p}{t_p+f_p}$	$\frac{t_n}{t_n+f_n}$
Literature	89.00%	1.49%	99.97%
MadGraph	89.000%	0.073%	99.999%

Table 3.4: Expected performance of the neural network trained on a T-restricted dataset, when used on real events in the target area

In conclusion, while the approach shows promise, it is by far not good enough, to alone identify a HH-Event among $t\bar{t}$-Events. However, in the target area a neural network trained on events in the target are by far outperforms a neural network trained on the full datasets, and this might improve, if given more data in the target area to train on.

Appendix A

Herwig

Originally Herwig 6.520 (6) (11) was used for the further generation of $t\bar{t}$-events after taking them from NCatNLO. The switch to Pythia happened, after attempts of generating datasets with wide top quark mass distribution lead to diminished returns in the numbers of events:

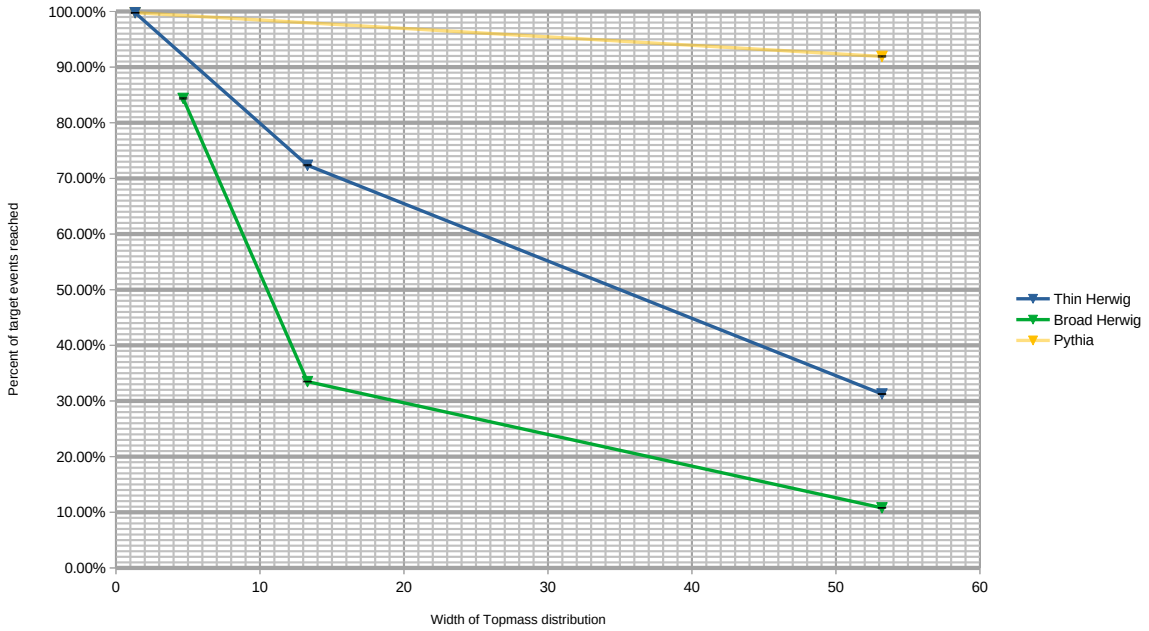


Figure A.1: Overview of the percentage of $t\bar{t}$-events of the target amount generated with different top quark width, Herwig with small boundaries for top mass (blue), Herwig with large boundaries for top mass (green), Pythia for top mass (yellow), the two data points of Pythia have different settings, with small boundaries (left) and with large boundaries (right); The x-axis here has the absolute value of Γ, with, instead of relative term used in Chapter 2, the conversion factor here is 1.33. The original target amount for events generated is 1M for every dataset except the thin Herwig with $\Gamma = 1.33$, where it is 12.5M.

As seen in Figure A.1, the $t\bar{t}$-datasets generated with Pythia are less affected by this loss of events. Pythia goes down to around 92% of the target amount of events, where Herwig only generates around 11% of the target amount of events.

For Pythia the already small loss of events eventually vanished almost completely, as can be seen by the Datasets of Sections 2.2 and 2.3.4 being unaffected, so there is a possibility, that

this could also be fixable for Herwig.

Appendix B

Rotation

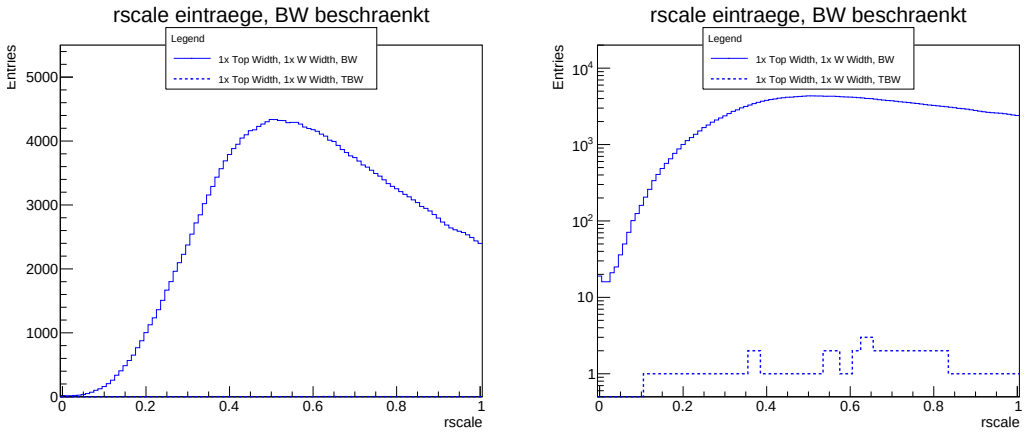
There were also attempts to improve the number of $t\bar{t}$-events passing the TBW-Conditions, by rotating the top quark decay products to reduce the W^+W^- pair mass. Based on the code of (4), though with all scaling factors set to 1, different rotation settings are systematically explored, as well as other improvements.

It basically works by looking at each W boson in the rest frame of its respective top quark, measuring the angle between those two W-boson vectors and then reducing that angle by a factor called r_{scale} by rotating the top quark decay products around the cross product of the W bosons by $(1 - r_{scale})$ times the *angle*.

The code goes through all different values of r_{scale} between 0 and 1 in 0.01 increments and looks at, how many events pass the conditions B and W, which are the conditions directly affected by this rotation, as well as how many events pass condition T in addition to that.

To note here is, that this code works on the datasets events as they were spit out by MCatNLO, unaffected by Pythia, which saves computation time, but also means, that the effects of smearing or jet reconstruction can not be studied here, as those are only applied in a later state in the pipeline.

First, the 5M events standard $t\bar{t}$-dataset:



(a) Linear Scaling

(b) Logarithmic Scaling

Figure B.1: Effect of r_{scale} on the number of events among a 5M standard $t\bar{t}$-dataset passing the BW-conditions (continuous line), and the TBW-conditions (dotted line)

While the number of events passing conditions B and W seems to peak somewhere around

$r_{scale} = 0.5$, there are too few events passing the TBW-conditions, to make any further comments on this.

How does the widening of the dataset from section 2.3.3 affect this:

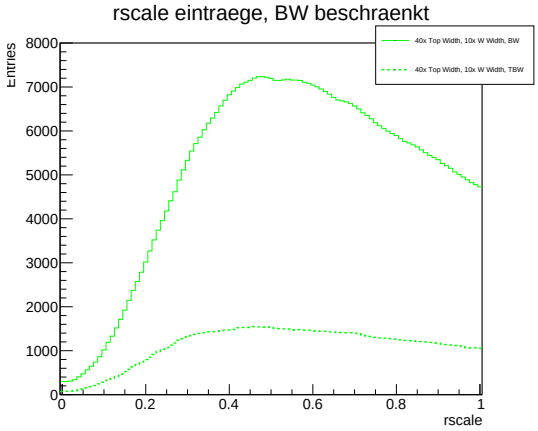


Figure B.2: Effect of r_{scale} on the number of events among a ca. 919k $t\bar{t}$-events dataset with widened top quark and W boson mass distributions passing the BW-conditions (continuous line), and the TBW-conditions (dotted line)

For the widened dataset, the number of events also peaks somewhere around $r_{scale} = 0.5$, though here there are enough events passing the TBW-conditions to say, that they seem to also peak somewhere around there.

And last the T-restricted datasets from Section 2.3.4:

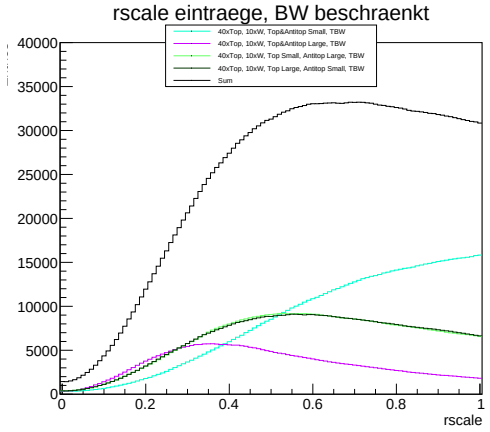


Figure B.3: Comparison, between the effects of r_{scale} on the number of events passing the TBW-conditions among different 1M T-restricted $t\bar{t}$-events datasets with widened top quark and W boson mass distributions: one with both top and antitop quark masses small (cyan), one with both top and antitop quark mass large (purple), two with one of them small and the other large (green) and their 4M events sum (black).

To note here is that, as all T-restricted events pass the T-condition before reconstruction and smearing by definition, there have to be no separate listings for passing the TBW- and BW-conditions.

The different T-restricted datasets peak at different values of r_{scale}, with the one for small top and antitop quark masses being at its best in the control case of $r_{scale} = 1$.

The sum of the datasets has a peak somewhere around $r_{scale} = 0.7$, but the amount of events won by the rotation is much smaller than before the T-restriction.

The diminishing returns of Rotation for a T-restricted $t\bar{t}$-dataset, the difficulty of studying the effects of it after smearing and jet reconstruction and the found possible relevance of the angle between the W bosons in distinguishing $t\bar{t}$-events from HH-events in section 2.4, lead to the decision of not pursuing the use of rotation to increase the number of $t\bar{t}$-events in the target area further.

Appendix C

Derivation of Equation 2.21

Be θ the angle between both bottom quarks in the HH rest frame. Everything here is in the HH rest frame, unless it is marked by $b\bar{b}$, in which case it is in the $b\bar{b}$ rest frame.

$$\mathbf{p}_H = \begin{pmatrix} \gamma_H m_H \\ 0 \\ 0 \\ \beta_H \gamma_H m_H \end{pmatrix} = \begin{pmatrix} E_H \\ 0 \\ 0 \\ p_H \end{pmatrix} = \begin{pmatrix} E_H \\ 0 \\ 0 \\ \sqrt{E_H^2 - m_H^2} \end{pmatrix}$$

$$\beta_H = \frac{\beta_H \gamma_H m_H}{\gamma_H m_H} = \frac{p_H}{E_H}$$

$$\gamma_H = \frac{\gamma_H m_H}{m_H} = \frac{E_H}{m_H}$$

$$\mathbf{p}_{b,b\bar{b}} = \begin{pmatrix} E_{b,b\bar{b}} \\ \sqrt{E_{b,b\bar{b}}^2 - m_b^2} \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{m_H}{2} \\ \sqrt{(\frac{m_H}{2})^2 - m_b^2} \\ 0 \\ 0 \end{pmatrix}$$

$$\mathbf{p}_b = \Lambda_H \mathbf{p}_{b,b\bar{b}} = \begin{pmatrix} \frac{E_H}{m_H} & 0 & 0 & \frac{-\sqrt{E_H^2 - m_H^2}}{m_H} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{-\sqrt{E_H^2 - m_H^2}}{m_H} & 0 & 0 & \frac{E_H}{m_H} \end{pmatrix} \begin{pmatrix} \frac{m_H}{2} \\ \sqrt{(\frac{m_H}{2})^2 - m_b^2} \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{E_H}{2} \\ \sqrt{(\frac{m_H}{2})^2 - m_b^2} \\ 0 \\ \frac{-\sqrt{E_H^2 - m_H^2}}{2} \end{pmatrix}$$

$$\mathbf{p}_{\bar{b}} = \begin{pmatrix} \frac{E_H}{2} \\ -\sqrt{(\frac{m_H}{2})^2 - m_b^2} \\ 0 \\ \frac{-\sqrt{E_H^2 - m_H^2}}{2} \end{pmatrix}$$

$$\cos(\theta) = \frac{\vec{p}_b \cdot \vec{p}_{\bar{b}}}{|\vec{p}_b||\vec{p}_{\bar{b}}|} = \frac{-(\sqrt{(\frac{m_H}{2})^2 - m_b^2})^2 + (\frac{-\sqrt{E_H^2 - m_H^2}}{2})^2}{(\sqrt{(\frac{m_H}{2})^2 - m_b^2})^2 + (\frac{-\sqrt{E_H^2 - m_H^2}}{2})^2} = \frac{-\frac{m_H^2}{4} + m_b^2 + \frac{E_H^2 - m_H^2}{4}}{\frac{m_H^2}{4} - m_b^2 + \frac{E_H^2 - m_H^2}{4}}$$

$$= \frac{\frac{E_H^2}{4} - \frac{m_H^2}{2} + m_b^2}{\frac{E_H^2}{4} - m_b^2} = 1 - \frac{\frac{E_H^2}{4} - m_b^2}{\frac{E_H^2}{4} - m_b^2} + \frac{\frac{E_H^2}{4} - \frac{m_H^2}{2} + m_b^2}{\frac{E_H^2}{4} - m_b^2} = 1 - \frac{\frac{m_H^2}{2} - 2m_b^2}{\frac{E_H^2}{4} - m_b^2}$$

$$= 1 - \frac{2m_H^2 - 8m_b^2}{E_H^2 - 4m_b^2}$$

For comparison, the general case with $\theta_{b\bar{b}}$ being the angle between the bottom quark in the $b\bar{b}$ rest frame and the Higgs boson in the HH rest frame:

$$\mathbf{p}_{b,b\bar{b}} = \begin{pmatrix} \frac{m_H}{2} \\ \sqrt{(\frac{m_H}{2})^2 - m_b^2}\sin\theta_{b\bar{b}} \\ 0 \\ \sqrt{(\frac{m_H}{2})^2 - m_b^2}\cos\theta_{b\bar{b}} \end{pmatrix}$$

$$\mathbf{p}_b = \Lambda_H \mathbf{p}_{b,b\bar{b}} = \begin{pmatrix} \frac{E_H}{m_H} & 0 & 0 & \frac{-\sqrt{E_H^2 - m_H^2}}{m_H} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{-\sqrt{E_H^2 - m_H^2}}{m_H} & 0 & 0 & \frac{E_H}{m_H} \end{pmatrix} \begin{pmatrix} \frac{m_H}{2} \\ \sqrt{(\frac{m_H}{2})^2 - m_b^2}\sin\theta_{b\bar{b}} \\ 0 \\ \sqrt{(\frac{m_H}{2})^2 - m_b^2}\cos\theta_{b\bar{b}} \end{pmatrix}$$

$$= \begin{pmatrix} \frac{E_H}{2} - \frac{\sqrt{E_H^2 - m_H^2}}{m_H}\sqrt{(\frac{m_H}{2})^2 - m_b^2}\cos\theta_{b\bar{b}} \\ \sqrt{(\frac{m_H}{2})^2 - m_b^2}\sin\theta_{b\bar{b}} \\ 0 \\ \frac{-\sqrt{E_H^2 - m_H^2}}{2} + \frac{E_H}{m_H}\sqrt{(\frac{m_H}{2})^2 - m_b^2}\cos\theta_{b\bar{b}} \end{pmatrix} = \begin{pmatrix} \frac{E_H}{2} - \gamma_H\beta_H p_{b,b\bar{b}}\cos\theta_{b\bar{b}} \\ p_{b,b\bar{b}}\sin\theta_{b\bar{b}} \\ 0 \\ \frac{-p_H}{2} + \gamma_H p_{b,b\bar{b}}\cos\theta_{b\bar{b}} \end{pmatrix}$$

$$\mathbf{p}_{\bar{b},b\bar{b}} = \begin{pmatrix} \frac{m_H}{2} \\ -\sqrt{(\frac{m_H}{2})^2 - m_b^2}\sin\theta_{b\bar{b}} \\ 0 \\ -\sqrt{(\frac{m_H}{2})^2 - m_b^2}\cos\theta_{b\bar{b}} \end{pmatrix}$$

$$\mathbf{p}_{\bar{b}} = \Lambda_H \mathbf{p}_{\bar{b},b\bar{b}} = \begin{pmatrix} \frac{E_H}{m_H} & 0 & 0 & \frac{-\sqrt{E_H^2 - m_H^2}}{m_H} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{-\sqrt{E_H^2 - m_H^2}}{m_H} & 0 & 0 & \frac{E_H}{m_H} \end{pmatrix} \begin{pmatrix} \frac{m_H}{2} \\ -\sqrt{(\frac{m_H}{2})^2 - m_b^2}\sin\theta_H \\ 0 \\ -\sqrt{(\frac{m_H}{2})^2 - m_b^2}\cos\theta_{b\bar{b}} \end{pmatrix}$$

$$= \begin{pmatrix} \frac{E_H}{2} + \frac{\sqrt{E_H^2 - m_H^2}}{m_H}\sqrt{(\frac{m_H}{2})^2 - m_b^2}\cos\theta_{b\bar{b}} \\ -\sqrt{(\frac{m_H}{2})^2 - m_b^2}\sin\theta_{b\bar{b}} \\ 0 \\ \frac{-\sqrt{E_H^2 - m_H^2}}{2} - \frac{E_H}{m_H}\sqrt{(\frac{m_H}{2})^2 - m_b^2}\cos\theta_{b\bar{b}} \end{pmatrix} = \begin{pmatrix} \frac{E_H}{2} + \gamma_H\beta_H p_{b,b\bar{b}}\cos\theta_{b\bar{b}} \\ -p_{b,b\bar{b}}\sin\theta_{b\bar{b}} \\ 0 \\ \frac{-p_H}{2} - \gamma_H p_{b,b\bar{b}}\cos\theta_{b\bar{b}} \end{pmatrix}$$

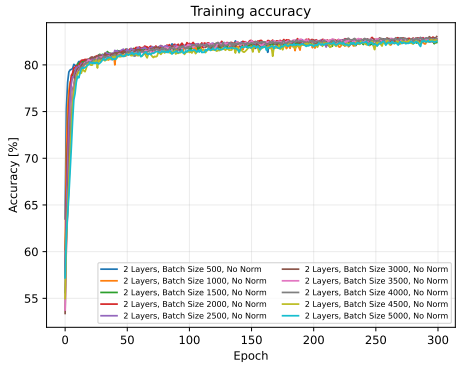
$$\cos(\theta) = \frac{\vec{p_b}\cdot\vec{p_{\bar{b}}}}{|\vec{p_b}||\vec{p_{\bar{b}}}|} = \frac{-p_{b,b\bar{b}}^2\sin^2\theta_{b\bar{b}} + \frac{p_H^2}{4} - \gamma_H p_{b,b\bar{b}}^2\cos^2(\theta_{b\bar{b}})}{\sqrt{p_{b,b\bar{b}}^2\sin^2(\theta_{b\bar{b}}) + (\frac{-p_H}{2} - \gamma_H p_{b,b\bar{b}}\cos\theta_{b\bar{b}})^2}\sqrt{p_{b,b\bar{b}}^2\sin^2(\theta_{b\bar{b}}) + (\frac{-p_H}{2} + \gamma_H p_{b,b\bar{b}}\cos\theta_{b\bar{b}})^2}}$$

Appendix D

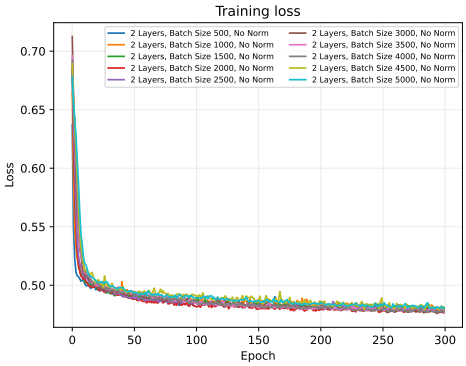
Combined Run options Overview

D.1 2 Hidden Layers

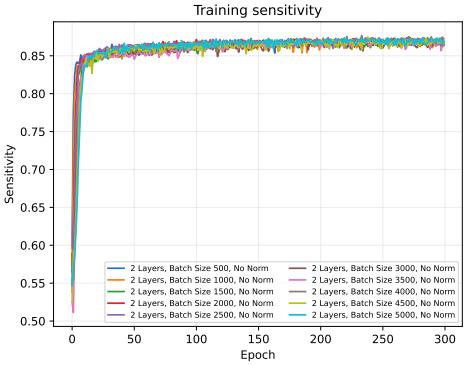
D.1.1 Not Normalized



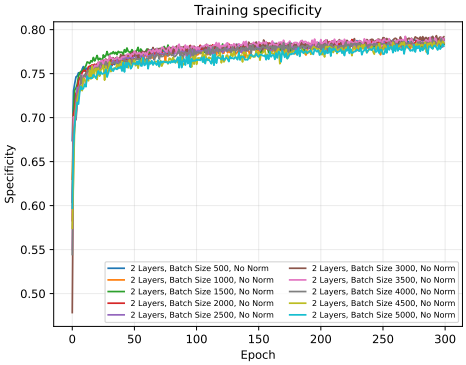
(a) Accuracy training data



(b) Loss training data

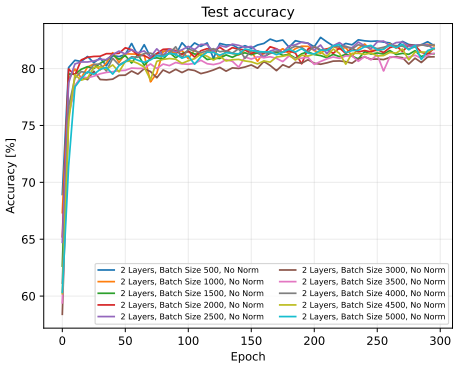


(c) Sensitivity training data

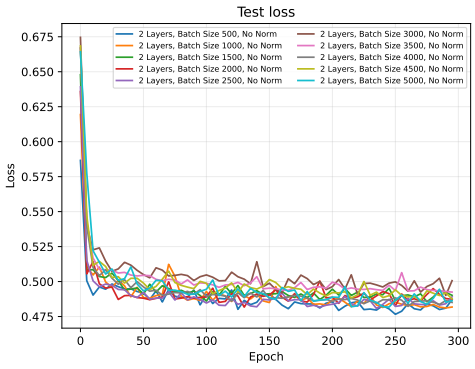


(d) Specificity training data

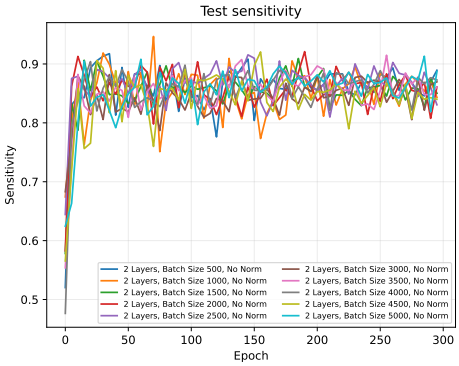
Figure D.1: Comparison of the training data of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); not normalized, with 2 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions



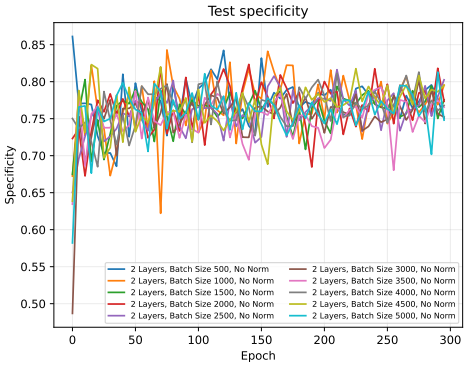
(a) Accuracy validation test data



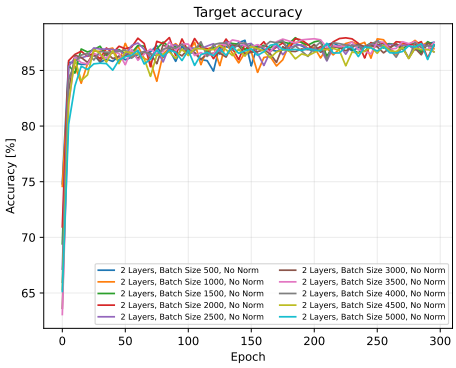
(b) Loss validation test data



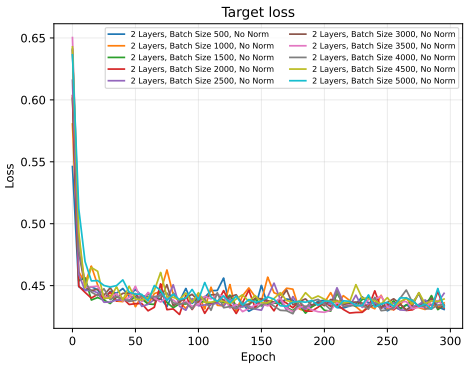
(c) Sensitivity validation test data



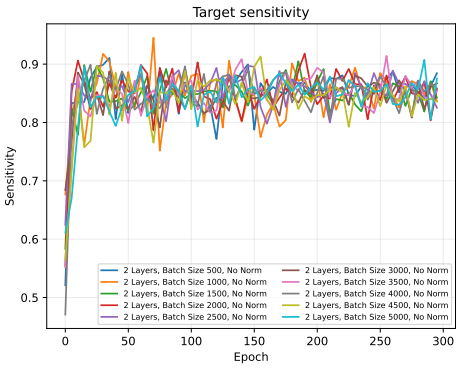
(d) Specificity validation test data



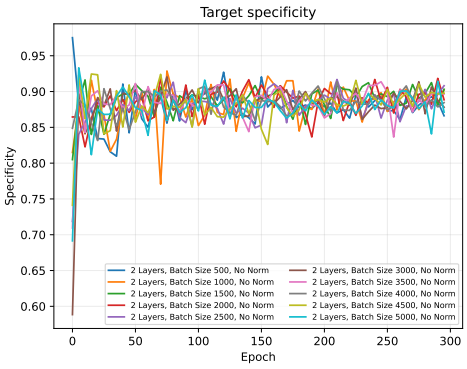
(e) Accuracy target data



(f) Loss target data



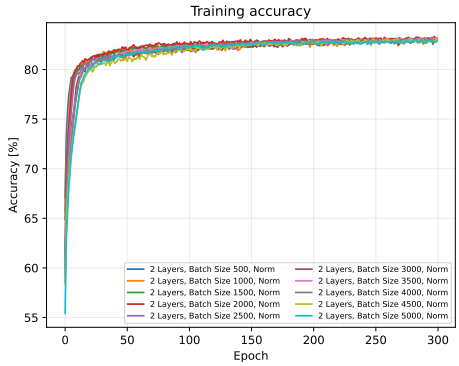
(g) Sensitivity target data



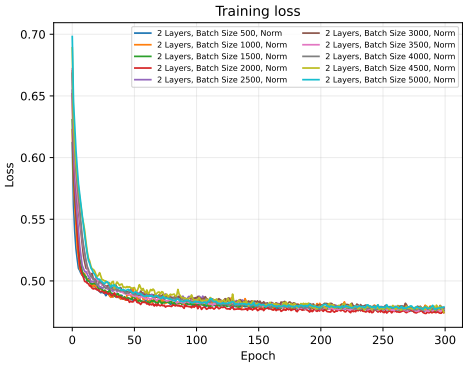
(h) Specificity target data

Figure D.2: Comparison of the validation test data (a-d) and target data (e-h) of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); not normalized, with 2 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

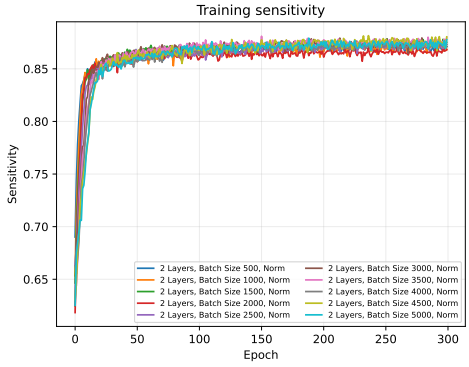
D.1.2 Normalized



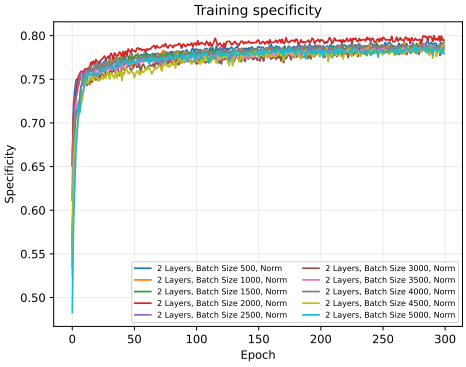
(a) Accuracy training data



(b) Loss training data

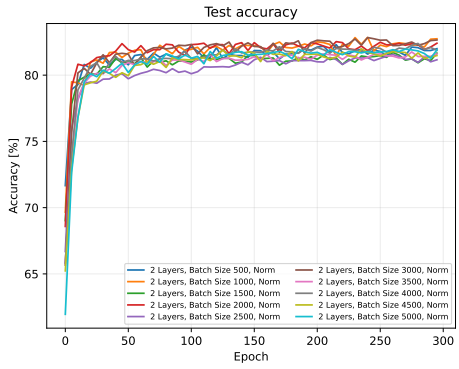


(c) Sensitivity training data

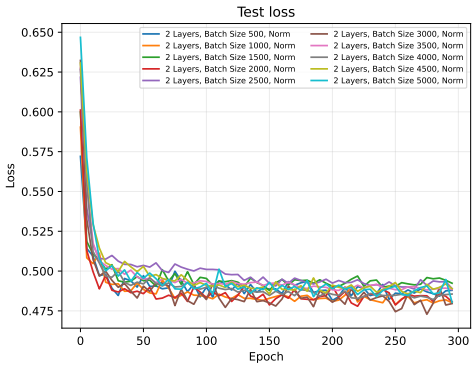


(d) Specificity training data

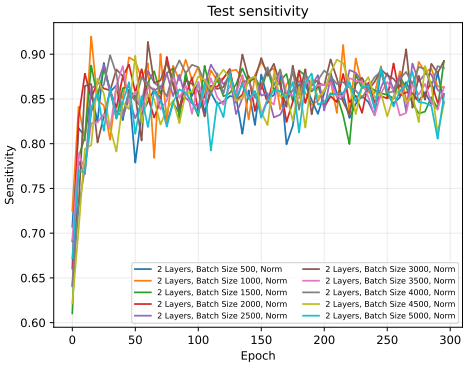
Figure D.3: Comparison of the training data of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); normalized, with 2 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions



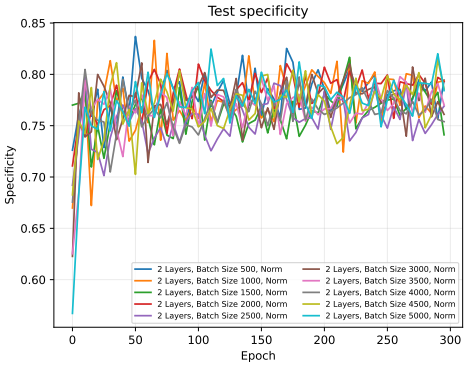
(a) Accuracy validation test data



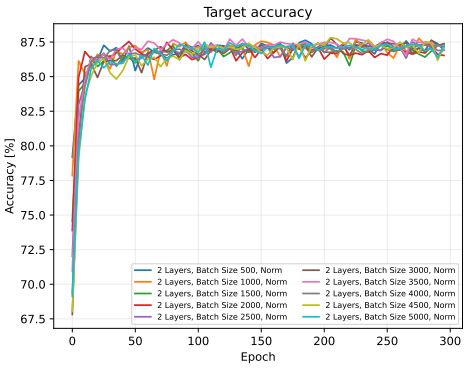
(b) Loss validation test data



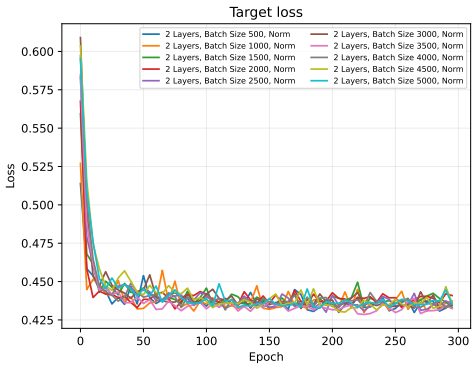
(c) Sensitivity validation test data



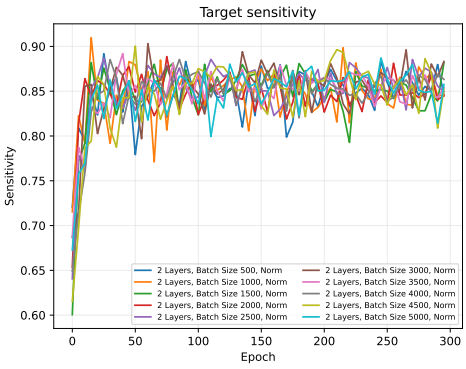
(d) Specificity validation test data



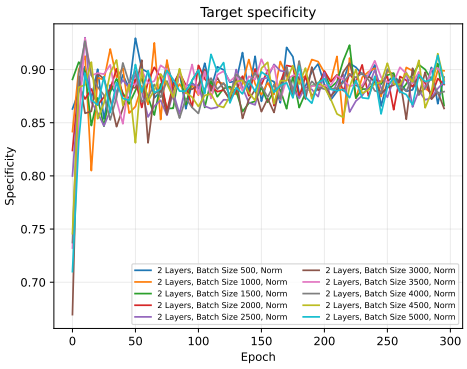
(e) Accuracy target data



(f) Loss target data



(g) Sensitivity target data

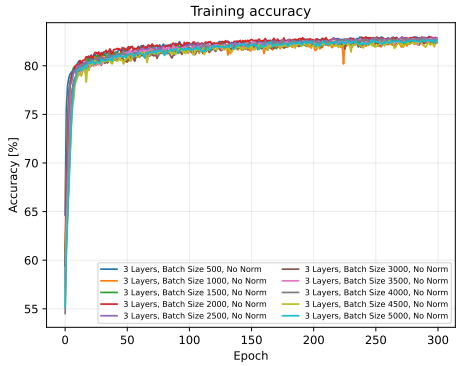


(h) Specificity target data

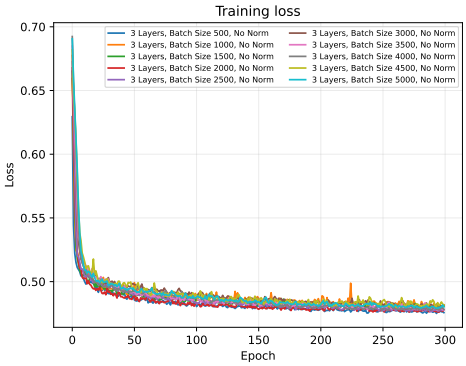
Figure D.4: Comparison of the validation test data (a-d) and the target data (e-h) of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); normalized, with 2 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

D.2 3 Hidden Layers

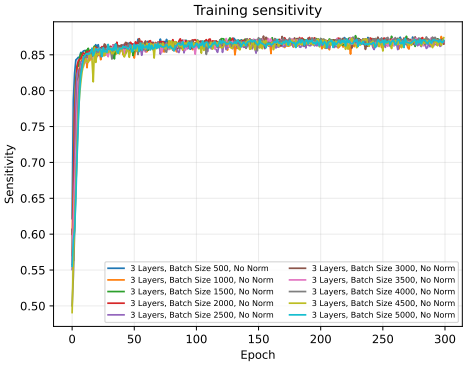
D.2.1 Not Normalized



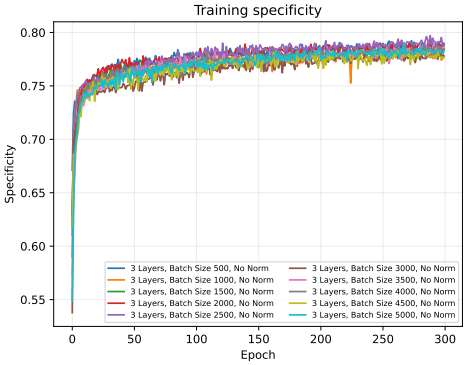
(a) Accuracy training data



(b) Loss training data

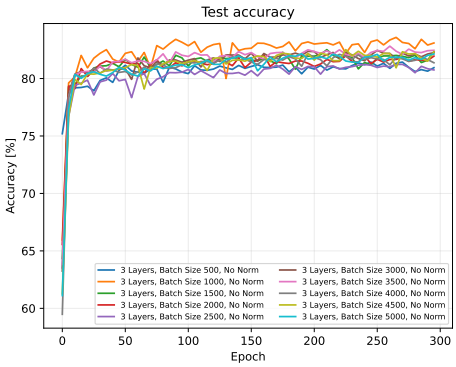


(c) Sensitivity training data

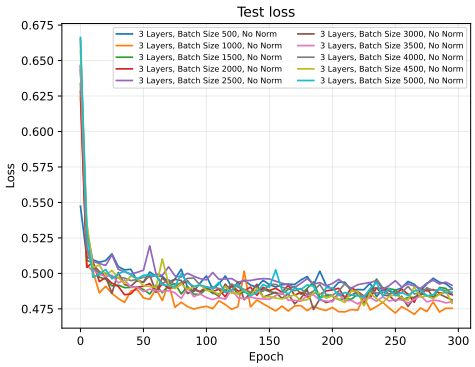


(d) Specificity training data

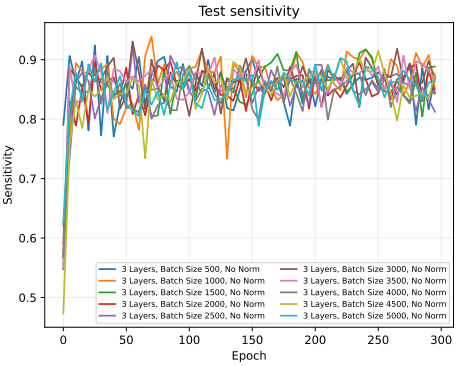
Figure D.5: Comparison of the training data of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); not normalized, with 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions



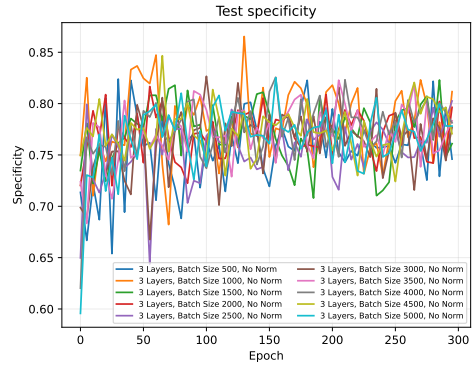
(a) Accuracy validation test data



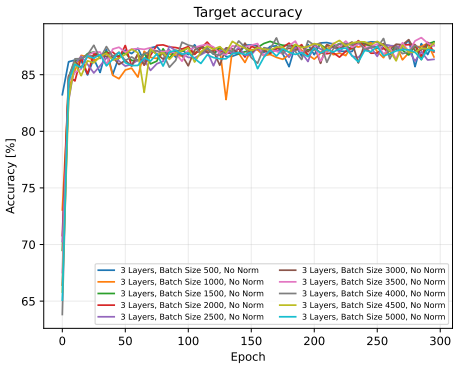
(b) Loss validation test data



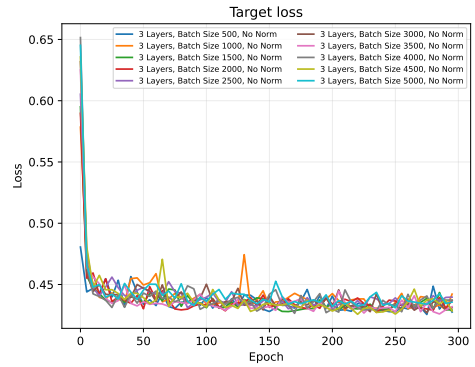
(c) Sensitivity validation test data



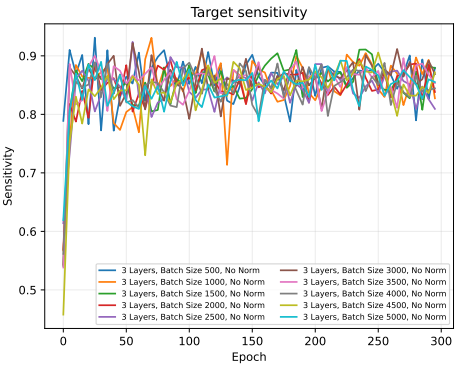
(d) Specificity validation test data



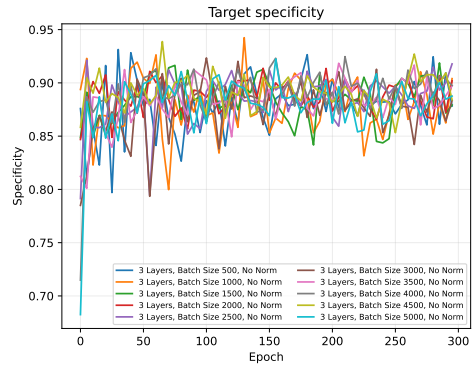
(e) Accuracy target data



(f) Loss target data



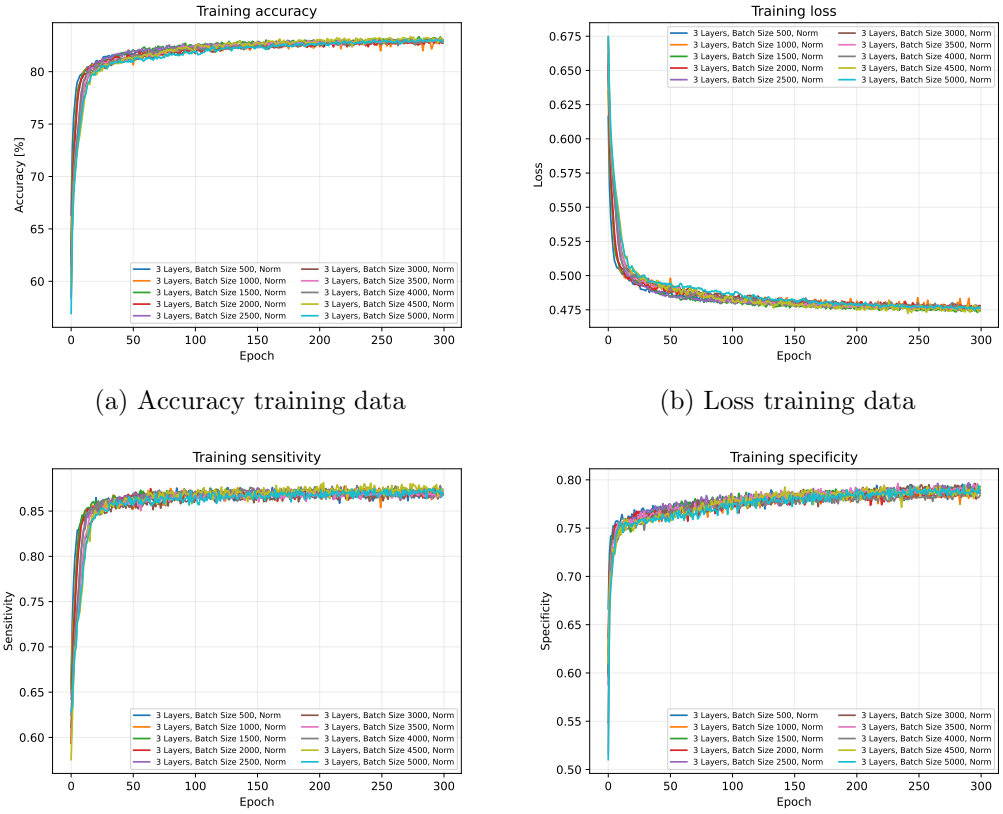
(g) Sensitivity target data



(h) Specificity target data

Figure D.6: Comparison of the validation test data (a-d) and the target data (e-h) of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); not normalized, with 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

D.2.2 Normalized



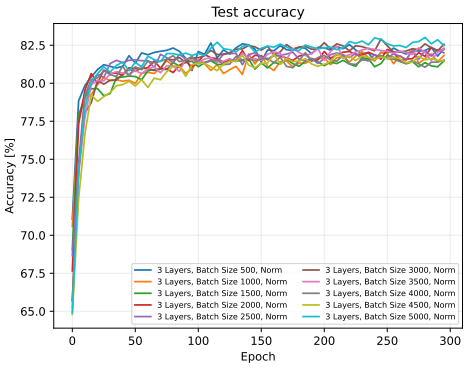
(a) Accuracy training data

(b) Loss training data

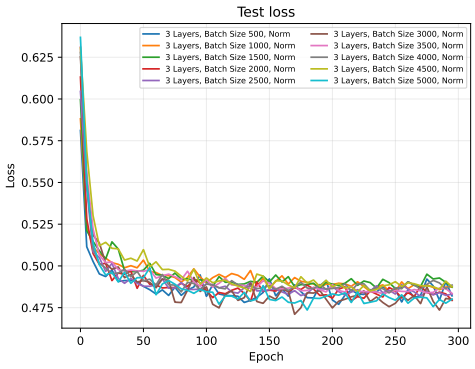
(c) Sensitivity training data

(d) Specificity training data

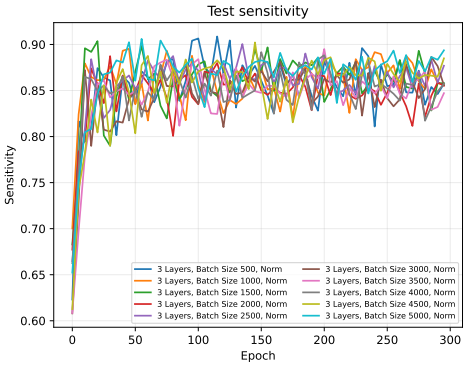
Figure D.7: Comparison of the training data of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); normalized, with 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions



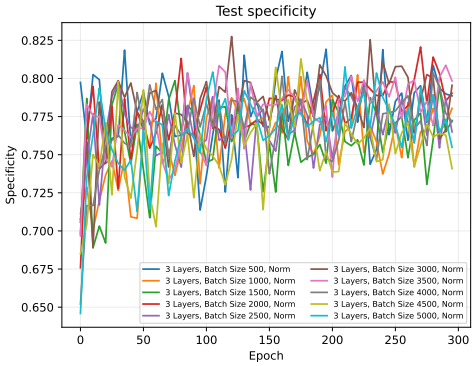
(a) Accuracy validation test data



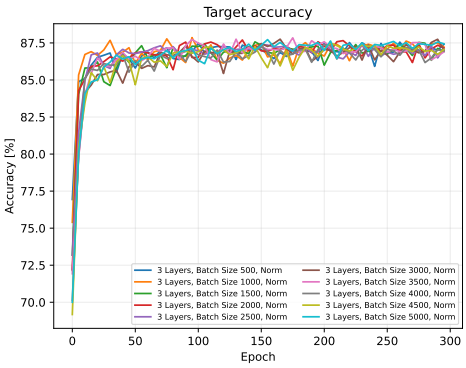
(b) Loss validation test data



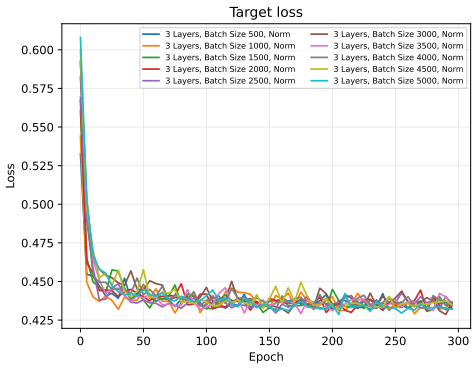
(c) Sensitivity validation test data



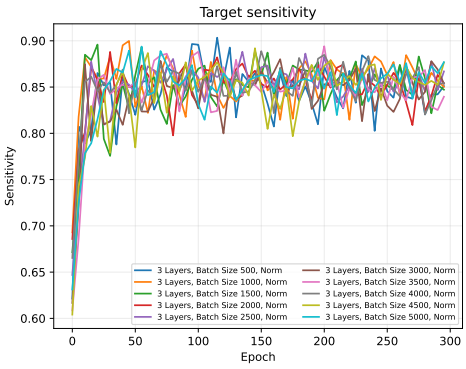
(d) Specificity validation test data



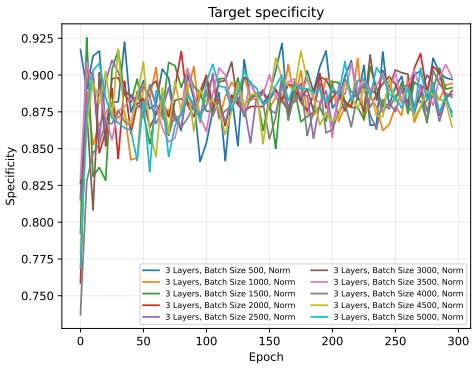
(e) Accuracy target data



(f) Loss target data



(g) Sensitivity target data

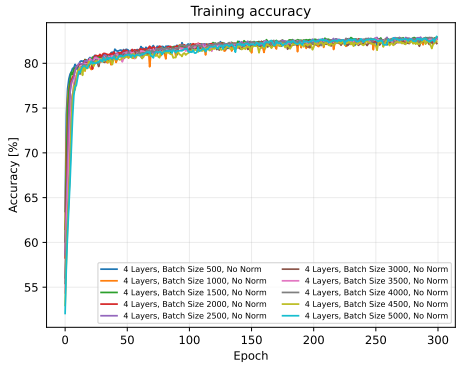


(h) Specificity target data

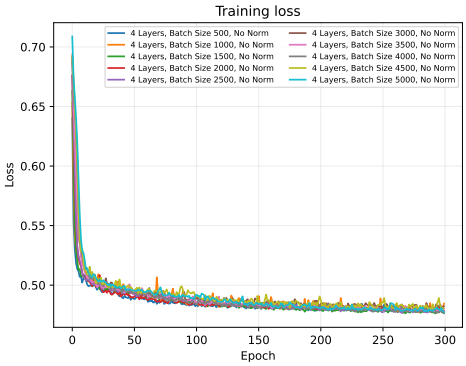
Figure D.8: Comparison of the validation test data (a-d) and the target data (e-h) of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); normalized, with 3 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

D.3 4 Hidden Layers

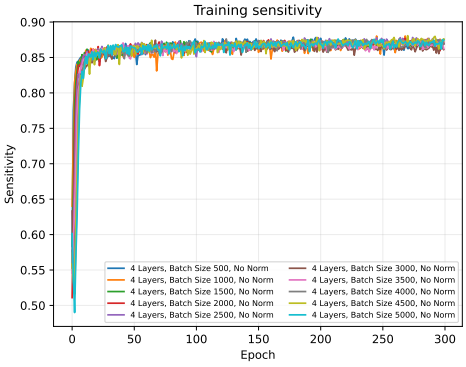
D.3.1 Not Normalized



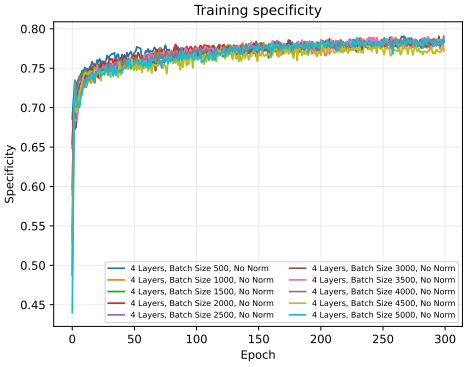
(a) Accuracy training data



(b) Loss training data

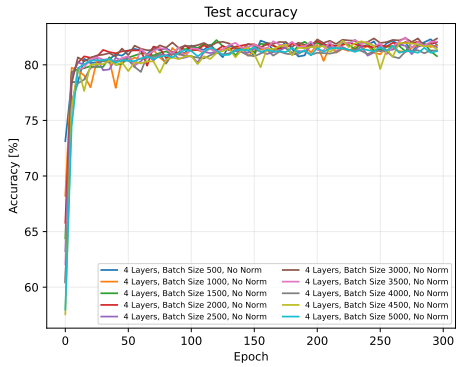


(c) Sensitivity training data

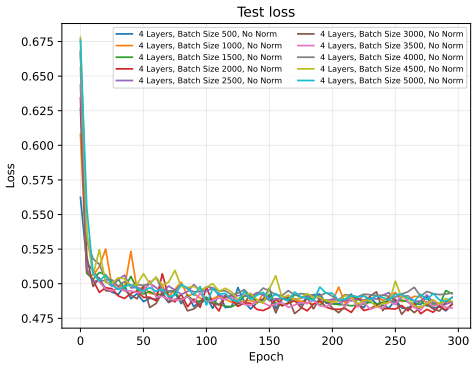


(d) Specificity training data

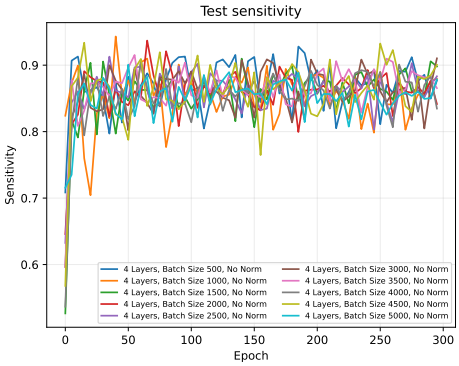
Figure D.9: Comparison of the training data of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); not normalized, with 4 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions



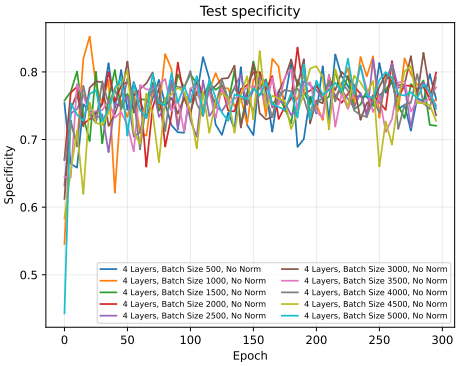
(a) Accuracy validation test data



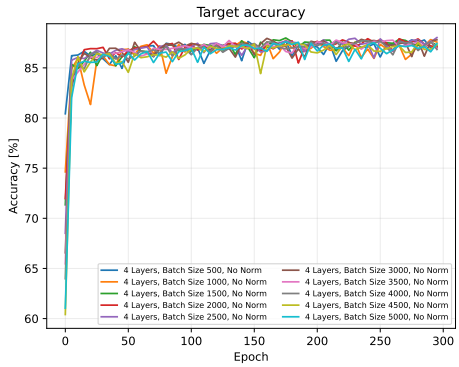
(b) Loss validation test data



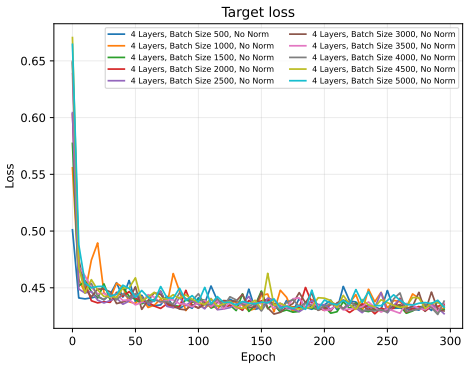
(c) Sensitivity validation test data



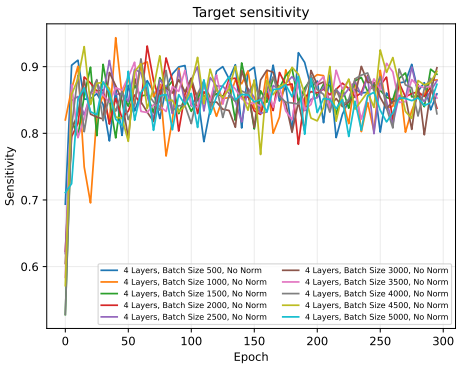
(d) Specificity validation test data



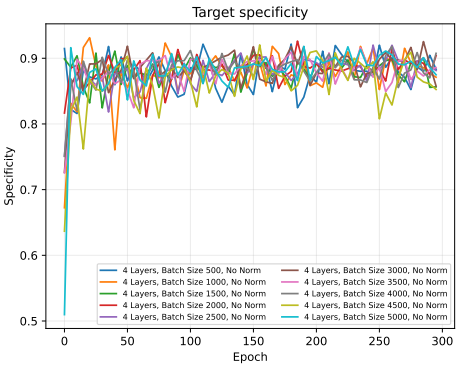
(e) Accuracy target data



(f) Loss target data



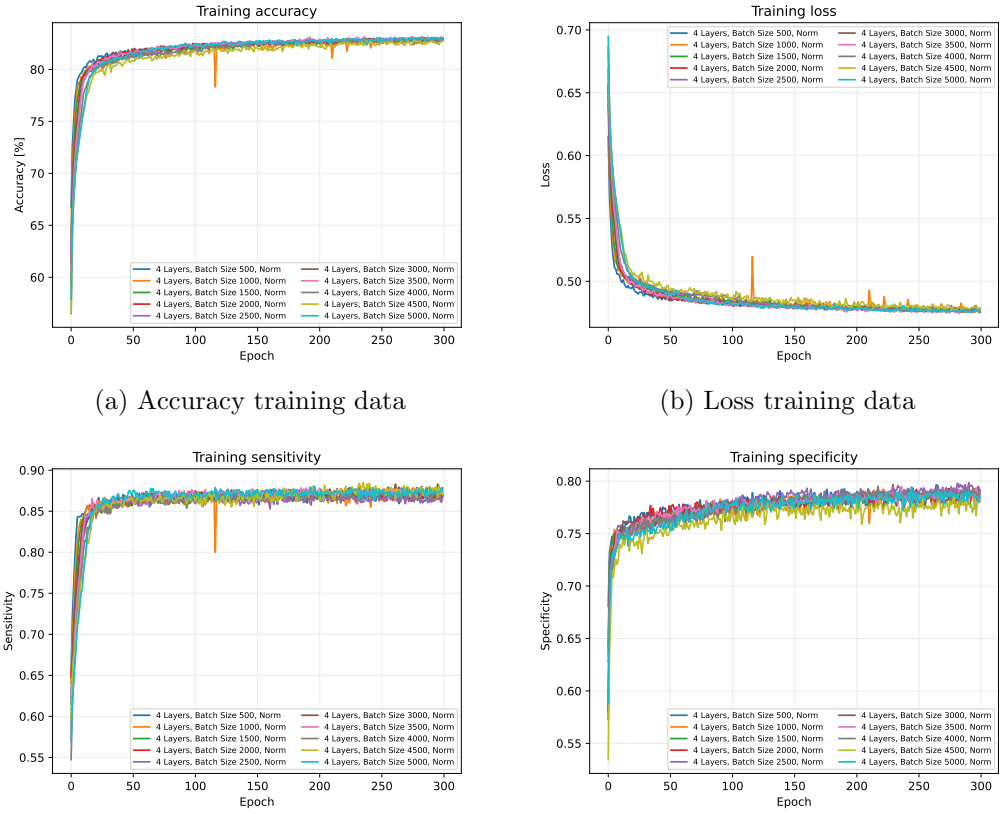
(g) Sensitivity target data



(h) Specificity target data

Figure D.10: Comparison of the validation test data (a-d) and the target data (e-h) of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); not normalized, with 4 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

D.3.2 Normalized



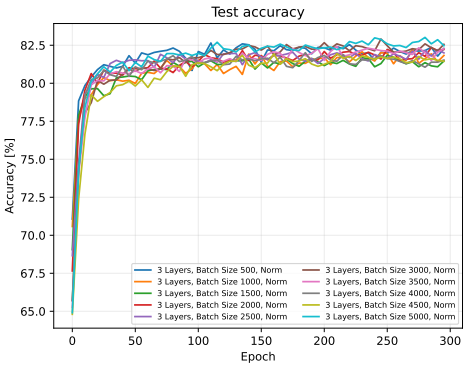
(a) Accuracy training data

(b) Loss training data

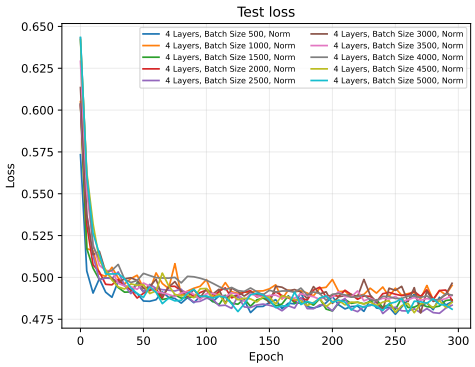
(c) Sensitivity training data

(d) Specificity training data

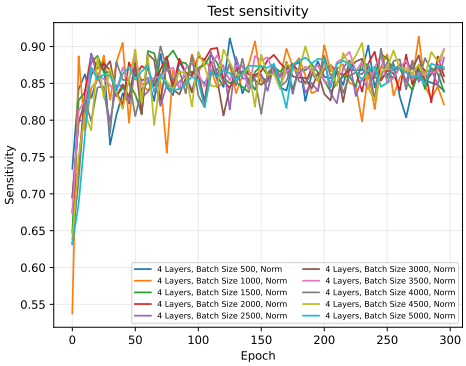
Figure D.11: Comparison of the training data of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); normalized, with 4 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions



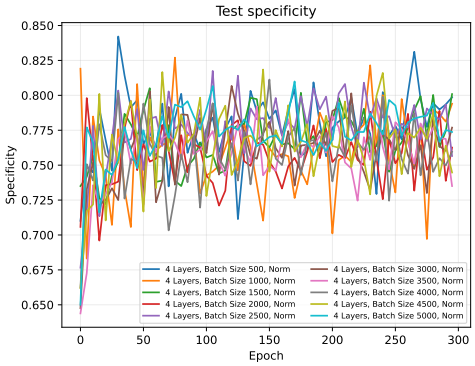
(a) Accuracy validation test data



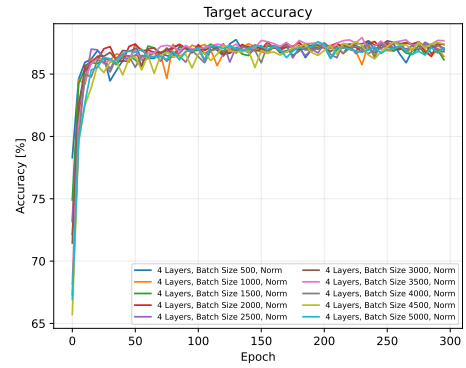
(b) Loss validation test data



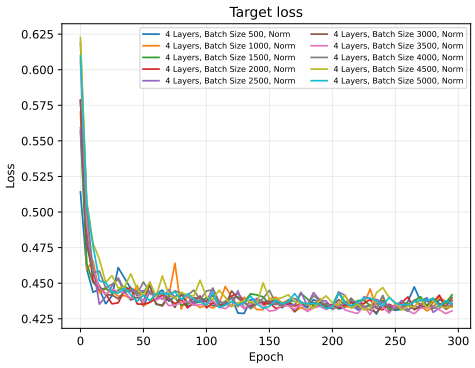
(c) Sensitivity validation test data



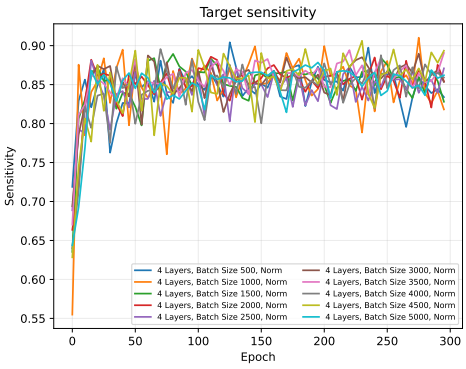
(d) Specificity validation test data



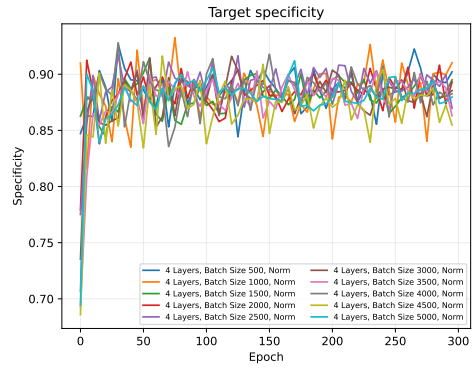
(e) Accuracy target data



(f) Loss target data



(g) Sensitivity target data



(h) Specificity target data

Figure D.12: Comparison of the validation test data (a-d) and the target data (e-h) of runs with different settings: batch sizes: 500 (blue), 1000 (orange), 1500 (dark green), 2000 (red), 2500 (purple), 3000 (brown), 3500 (pink), 4000 (grey), 4500 (olive green) and 5000 (cyan); normalized, with 4 hidden layers, trained on a smeared, jet reconstructed, T-restricted dataset after applying the TBW-conditions

Bibliography

[1] Alwall, J. ; Frederix, R. ; Frixione, S. ; Hirschi, V. ; Maltoni, F. ; Mattelaer, O. ; Shao, H.-S. ; Stelzer, T. ; Torrielli, P. ; Zaro, M.: The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations. In: *Journal of High Energy Physics* 2014 (2014), Juli, Nr. 7. http://dx.doi.org/10.1007/jhep07(2014)079. – DOI 10.1007/jhep07(2014)079. – ISSN 1029–8479

[2] Andrew L. Maas, Andrew Y. N. Awni Y. Hannun H. Awni Y. Hannun: *Rectifier Nonlinearities Improve Neural Network Acoustic Models*. https://ai.stanford.edu/~amaas/papers/relu_hybrid_icml2013_final.pdf. Version: 2013

[3] Cacciari, Matteo ; Salam, Gavin P. ; Soyez, Gregory: FastJet user manual: (for version 3.0.2). In: *The European Physical Journal C* 72 (2012), März, Nr. 3. http://dx.doi.org/10.1140/epjc/s10052-012-1896-2. – DOI 10.1140/epjc/s10052–012–1896–2. – ISSN 1434–6052

[4] Cho, Youn J.: *Seperating $t\bar{t}$ and HH Final States Using Neural Networks*, LMU, Master's thesis, 2024

[5] Collaboration, ATLAS: Jet energy scale and resolution measured in proton–proton collisions at $\sqrt{s} = 13$·TeV with the ATLAS detector. In: *The European Physical Journal C* 81 (2021), aug, Nr. 8. http://dx.doi.org/10.1140/epjc/s10052-021-09402-3. – DOI 10.1140/epjc/s10052–021–09402–3. – ISSN 1434–6052

[6] Corcella, Gennaro ; Knowles, Ian G. ; Marchesini, Giuseppe ; Moretti, Stefano ; Odagiri, Kosuke ; Richardson, Peter ; Seymour, Michael H. ; Webber, Bryan R.: HERWIG 6: an event generator for hadron emission reactions with interfering gluons (including supersymmetric processes). In: *Journal of High Energy Physics* 2001 (2001), Januar, Nr. 01, 010–010. http://dx.doi.org/10.1088/1126-6708/2001/01/010. – DOI 10.1088/1126–6708/2001/01/010. – ISSN 1029–8479

[7] Czakon, Michał ; Fiedler, Paul ; Mitov, Alexander: Total Top-Quark Pair-Production Cross Section at Hadron Colliders Through $O(\alpha_S^4)$. In: *Physical Review Letters* 110 (2013), Juni, Nr. 25. http://dx.doi.org/10.1103/physrevlett.110.252004. – DOI 10.1103/physrevlett.110.252004. – ISSN 1079–7114

[8] Hinton, Geoffrey E. ; Srivastava, Nitish ; Krizhevsky, Alex ; Sutskever, Ilya ; Salakhutdinov, Ruslan R.: *Improving neural networks by preventing co-adaptation of feature detectors*. https://arxiv.org/abs/1207.0580. Version: 2012

[9] Holzner, T.: *Trennung von HH- und t-Endzuständen für niedrige Top-Quark Massen mittels Spinkorrelation und Machine-Learning Methoden*, LMU, Bachelor's thesis, 2025

[10] Internal neural network tutorial by Lars Lindon

[11] Marchesini, G. ; Webber, B.R. ; Abbiendi, G. ; Knowles, I.G. ; Seymour, M.H. ; Stanco, L.: HERWIG 5.1 - a Monte Carlo event generator for simulating hadron emission reactions with interfering gluons. In: *Computer Physics Communications* 67 (1992), Nr. 3, 465-508. http://dx.doi.org/https://doi.org/10.1016/0010-4655(92)90055-4. – DOI https://doi.org/10.1016/0010–4655(92)90055–4

[12] Micco, Biagio D. ; Gouzevitch, Maxime ; Mazzitelli, Javier ; Vernieri, Caterina: Higgs boson potential at colliders: Status and perspectives. In: *Reviews in Physics* 5 (2020), November, 100045. http://dx.doi.org/10.1016/j.revip.2020.100045. – DOI 10.1016/j.revip.2020.100045. – ISSN 2405–4283

[13] https://ml-cheatsheet.readthedocs.io/en/latest/loss_functions.html

[14] Navas, S. u. a.: Review of particle physics. In: *Phys. Rev. D* 110 (2024), Nr. 3, S. 030001. http://dx.doi.org/10.1103/PhysRevD.110.030001. – DOI 10.1103/PhysRevD.110.030001

[15] https://docs.pytorch.org

[16] Sjöstrand, Torbjörn ; Mrenna, Stephen ; Skands, Peter: PYTHIA 6.4 physics and manual. In: *Journal of High Energy Physics* 2006 (2006), Mai, Nr. 05, 026–026. http://dx.doi.org/10.1088/1126-6708/2006/05/026. – DOI 10.1088/1126–6708/2006/05/026. – ISSN 1029–8479

Selbständigkeitserklärung

Ich versichere hiermit, die vorliegende Arbeit mit dem Titel

Untersuchung zur Anhäufung seltener Top-Antitop Quark Ereignisse im Hintergrund zur Higgs-Paar Erzeugung und ihre Identifizierung durch Benutzung von maschinellen Lernen

selbständig verfasst zu haben und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet zu haben.

Tim Marvin Rexrodt

München, den 20. Januar 2026