Searching for $t\bar{t}HH$ events with the ATLAS Experiment at the LHC

Giorgio Stucchi

September 2023

Abstract

The Higgs self-coupling, which determines the shape of the Higgs potential, is one of the building blocks of the Standard Model of Particle Physics, directly connected to the electroweak symmetry breaking mechanism and to the stability of the Universe. In fact, probing the self-coupling is among the most important goals of the Particle Physics community and represents a key physics benchmark for the High-Luminosity phase of the Large Hadron Collider experiments.

Searching for Higgs pair production constitutes a direct way to probe the Higgs self-coupling and a comprehensive research programme is ongoing in ATLAS to perform these analyses in Run 2 and Run 3 data. The production of a top quark-antiquark pair in association with a Higgs pair ($t\bar{t}HH$) plays a special role in this context because, even if much rarer than the gluon fusion or Vector boson fusion production modes, is very sensitive to physics Beyond the Standard Model which predicts a quartic interaction between two top quarks and two Higgs bosons.

My project was centered around the initial investigations of $t\bar{t}HH$ processes using ATLAS data, specifically focusing on the 4b decay mode of the two Higgs bosons. My efforts were primarily dedicated to developing the analysis framework by addressing its issues, and I also worked on improving the associated plotting tool. Additionally, I set up a deep neural network and conducted training thereof to differentiate between signal and background events.

Supervisors: Valentina Cairo and Stefano Manzoni

1 Introduction

One crucial property of the Higgs boson is its self-coupling, which has implications for the behavior of the Higgs field and the matter-antimatter asymmetry in the universe. This self-coupling is described by both a trilinear and a quartic vertex in the Standard Model (SM).

The trilinear vertex can be studied by observing final states with two Higgs bosons, while the quartic vertex is more challenging to probe due to its weaker coupling. Researchers are interested in determining if the trilinear coupling differs from the SM prediction, as it could indicate Beyond the Standard Model (BSM) physics.

Several production mechanisms for di-Higgs (HH) events exist, with varying cross-sections. Gluon-gluon fusion (ggF) and vector-boson fusion (VBF) are the most common and well-studied processes. However, around SM values of the self-coupling, these processes suffer from distructive interference effects, reducing their sensitivity in HH searches.

Associated top production ($t\bar{t}HH$) and associated vector-boson production (VHH) modes are being explored as they could supplement existing constraints. ATLAS and CMS collaborations have initiated studies on these modes, with promising results.

Measuring $t\bar{t}HH$ production not only enhances di-Higgs searches but also facilitates the exploration of new physics in the top and Higgs sectors, particularly in effective field theories (EFTs). The $t\bar{t}HH$ vertex in EFTs has different implementations, and measuring $t\bar{t}HH$ would expand our ability to probe BSM physics.

Investigating the $t\bar{t}HH$ process is quite challenging due to its low cross-section. My report specifically focuses on the HH final state involving four bottom quarks (bbbb), accounting for 34% of the branching ratio: with the inclusion of top quarks, which always decay into b quarks and W bosons, it becomes a six-bottom-quark (6b) final state. However, HH→4b is not the only process leading to a 6b final state, which presents a major challenge for the analysis.

A list of other processes that could significantly contribute to background events in our signal region was selected. These processes include $t\bar{t}H$, $t\bar{t}Zjj$ and $t\bar{t}$. In these processes, 'j' represents a jet of any flavor produced alongside other particles.

DataSet	L_{int}	σ	N_{events}
Run2 [13 TeV]	150 fb^{-1}	0.775 fb	116
Run3 [13.6 TeV]	$\approx$ 500 fb^{-1}	$\approx$ 0.879 fb	440
HL-LHC [14 TeV]	$\approx$ 3000 fb^{-1}	0.949 fb	2847

Table 1: Predicted number of $t\bar{t}HH$ events

DataSet	L_{int}	σ	N_{events}
Run2 [13 TeV]	150 fb^{-1}	832 pb	$1.24 \cdot 10^8$
HL-LHC [14 TeV]	$\approx$ 3000 fb^{-1}	10^6 fb	$3 \cdot 10^9$

Table 2: Predicted number of $t\bar{t}$ events

The problem lies in the much bigger cross section for the background events with respect to the signal events: an example comparison for $t\bar{t}$ is shown in tables 1 and 2.

In particular, my analysis only considers as signal the full hadronic decay channel, i.e. the decay channel where the W bosons decay into hadrons (with a branching ratio of 67%). The Feynman diagram for this decay type is shown in Figure 2. The other two possibilities would be the semileptonic channel or the dileptonic channel (respectively one or two W bosons decaying into leptons), but they weren't considered in my report.

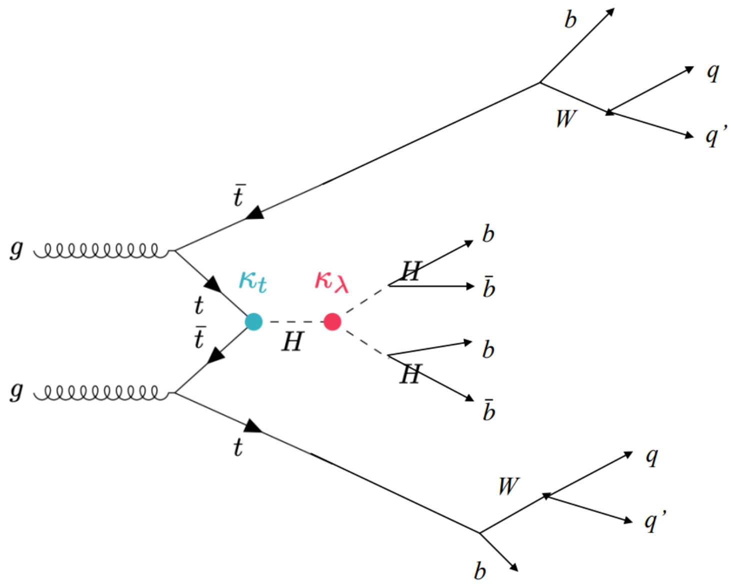


Figure 2: Feynman diagram for full hadronic $t\bar{t}HH$.

2 Work on the analysis program

Given an input of simulated Monte Carlo (MC) events or actual data derivations, the analysis program returns ".root" NTUPLES of relevant kinematic variables. Additionally, the program selects the events on the basis of specified cuts (e.g. number of jets in the event, minimum transverse momentum, etc.). The primary goal in the beginning was making the program run with data, which had previously not been attempted. To achieve this, the code had to modified in various instances; this proved to be a labor-intensive process.

In general, my contribution to the analysis program consisted in:

- I added cross-sections values for the different background processes that were not listed in XSections_mc20.txt.
- I fixed the implementation of the cuts in EventSelection.cxx (events with a number of leptons higher than m_leptonNumberCut were being mistakenly discarded).
- I fixed a problem with the MET containers, which were being filled incorrectly. In fact, it became evident at some point that the MET plot had a number of entries greater than 1/4 of the entries in the recojet_antikt4_pt plots, which was clearly incorrect, given that the MET is a scalar (1 entry per event) and the recojet_antikt_pt a vector of size greater or equal to 4. The problem lay in the containers being filled from input, instead of being created inside the framework as all other containers (electrons and muons containers for example). This did not allow for the implementation of the cuts, since the MET container was not being filled in EventSelection.cxx, where the cuts are applied. The problem was solved by making the MET case analogous to the electron and muon case, creating the container inside TTHH4BTree.cxx and then filling it in EventSelection.cxx only when the event passed the cuts.
- I computed angular variables such as $\Delta\phi$, ΔR, $\Delta\eta$ and saved them alongside the other kinematic variables in the output NTUPLES. The angular separations were computed for various combinations of the jets in the event, for example between the leading and subleading jets.
- I resolved an issue with the number of entries passing the cuts. Previously, the container nrecojet_antikt4 was being filled for every event, even when the event was not passing the cuts, leading to an incorrect number of output events.

- I wrote a bash script to retrieve and select all MC simulations corresponding to a specified year of data taking (or all) based on the filenames of the simulations. The script then runs the program for all selected files. I also added a version that selects the files based on the cut that has been applied on the events (FullHad, Semilep, Dilep).
- I wrote a bash script that retrieves all of the outputs of the simulation and merges them into different files on the basis of one of the following criteria: by number of Higgs bosons in the decay channel, by the name of the channel, by the channel's nature, i.e. whether it is signal or background. This script is fundamental when the outputs become very numerous, as the plotting tool can only open a certain - limited - number of files simultaneously.
- I addressed an issue with the uncertainty in the pt plots of the $t\bar{t}$ samples, which was larger than it should have been (based on the comparison with analogous plots). The problem lay in a missing division by 1000 in the code, which made most of the data overflow.

3 Work on the plotting tool

We used a plotting tool which was updated during the summer into a new framework, which is divided in two parts. The first part is CreateHists.cxx, which creates the root histograms defined in the configuration file config.json. The second part is DrawHists.cxx, which actually plots the histograms. This division allows for easy modifications (such as changes of fill colors) of the plotting code not to require a complete rerun of the whole program, which is quite time consuming. Two usage options are available: the "standard" one, which plots all the kinematic variables, and the "Cutflow" analysis one, which plots trigger and event selection efficiency cut flow.

My contributions to the plotting tool were:

- I added ratio plots (MC simulation / data) and stack plots (sum all MC simulation overlaid with data). In particular, they are both only plotted when the key "data" is present in the json configuration file.
- I implemented the cut flow analysis option inside the new framework.
- I provided the code to produce the significance plot within the cut flow analysis.
- I fixed some minor issues, such as legend size and placement, names of the labels, etc.
- I created a color map, such that each decay channel is associated to one RGB color (to be specified in the configuration file).
- I wrote the code to plot the efficiency of each cut on the events (i.e., how many events are percentually selected).
- I wrote a python script, retrieveFiles.py, that serves as a utility for updating and managing file paths within the JSON configuration file config.json . It constructs file paths by identifying relevant subfolders within the specified input folder based on the year of data taking and the analysis cut mode. These subfolders are assumed to follow a specific naming convention. The script then updates the JSON configuration file by adding or modifying file path entries for the various decay channels, along with associated information like file types and colors.

4 Work on the neural network

A deep neural network was set up for binary classification: signal and background. The only constraints on the structure of the DNN were given by number of input variables and the final layer, which has to be constituted by a single node. Everythings else, i.e., learning rate, batchsize, number of nodes, number of layers, dropout rate, could be played with to optimize the performance of the DNN. We need to process ".root" NTUPLES, therefore we elect to work with the python library uproot, which allows us to upload the data and then convert it to a Pandas format, which can in turn be processed by common machine learning libraries like Keras Tensorflow. As template, a code that was already available for the HHH $\rightarrow$ 6b analysis was used (https://gitlab.cern.ch/trihiggs/dnn_training).

The initial phase of the project involved customizing the code to align with our specific requirements, streamlining and removing unnecessary components, and selecting the input variables for out DNN. Our input files had to be modified ('flattened' by removing vectors) to let the program run without crashing. Secondly, there was a problem with the data size: our signal and background files were 1,4 GB and 800 MB large, respectively. The program could not handle so much data, so we had to momentarily reduce the amount of data in input (by a factor of 8). I later solved this problem: the solution required using HTCondor and requesting additional RAM memory for the execution of the program (compared to what was available locally on Lxplus). Additionally, the program worked poorly as soon as MC weights were added to the events, so we removed them alltogether (see Section 5).

Once the DNN was up and running, I moved on to the optimization of the hyperparameters. This required using a batch system (again, HTCondor) to submit jobs, as each run of the program lasted ≈ 1 h. I created a grid of hyperparameters (learning rate, batchsize, number of nodes, number of layers, dropout rate), submitted each one of the parameters' combinations to a different machine, and ran them simultaneously. Finally, I collected the results and chose the best parameters' combination by selecting the maximum ROC AUC (area under the ROC curve).

In addition, I added the possibility to plot the significance of the neural network prediction within the DNN code (computed as ratio between true positive rate and square root of false positive rate). I also added plots of all the input variables, each of which shows the comparison between signal and background. I also modified the preexisting code to plot correlation matrices (for both signal and background).

5 Results

We provide the following samples as input to the DNN: $t\bar{t}HH$ full hadronic, $t\bar{t}HH$ semileptonic and $t\bar{t}HH$ dileptonic for the signal, and $t\bar{t}$ and $t\bar{t}Zjj$ for the background. In figure 3, we show the comparison between signal and background as relates to the total number of jets in the event.

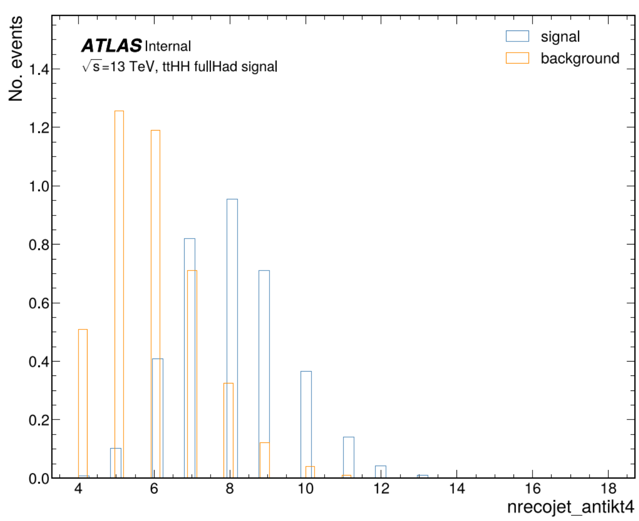
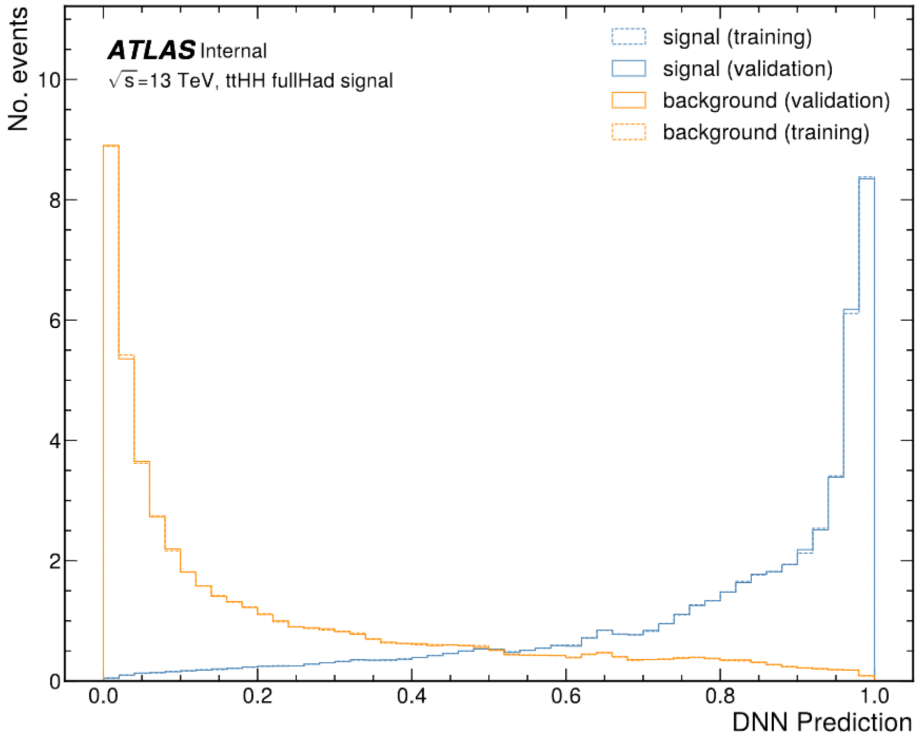


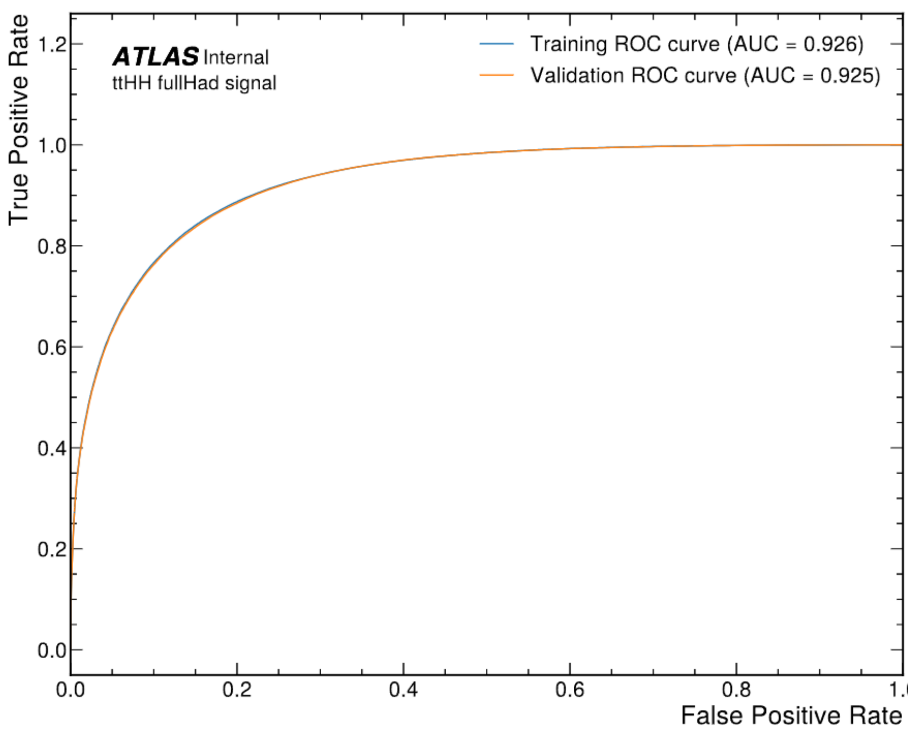
Figure 3: Distribution of number of jets per event: signal vs background.

It can be seen that the number of events with njets ≥ 6 is dramatically low for the background; this means that the background is not actually a "background" in the signal region (njets ≥ 6, given that there are at least 6 b jets in the signal events). The $t\bar{t}$ sample is therefore inadequate. At the moment, it provides great discrimination power for the DNN it is straightforward to implement a cutoff and classify events with fewer than six jets as background.

The predictions of the DNN are shown in Figure 4a.



(a) Output Prediction of the DNN.



(b) ROC curve of the DNN.

Figure 4: DNN outputs.

The distributions for signal and background are well distinguished. It must again be noted, however, that no weights were applied. If that had been the case, then most of the signal would have been identified as background, with a distribution peaking on 0. The problem with the weights is twofold: we need to (temporarily) normalize the weights to balance signal and background for the training, and on the other we need to calculate the total weight, i.e., the MC weight multiplied by cross section and luminosity and divided by the total sum of weights. However, at the moment the DNN cannot handle the last request, given that it takes "signal" and "background" files that are not separated in channels, hence not distinguished as relates to the cross section of the decay. The outcome of not applying the weights is not being able to discuss sensibly about significance.

In figure 4b we show the ROC curve for the DNN. Training and validation score are very similar, probably due to the application of a 0.1 dropout rate.

To get insights on the relative importance of the input variables, we used Shapley plots (Figure 5).

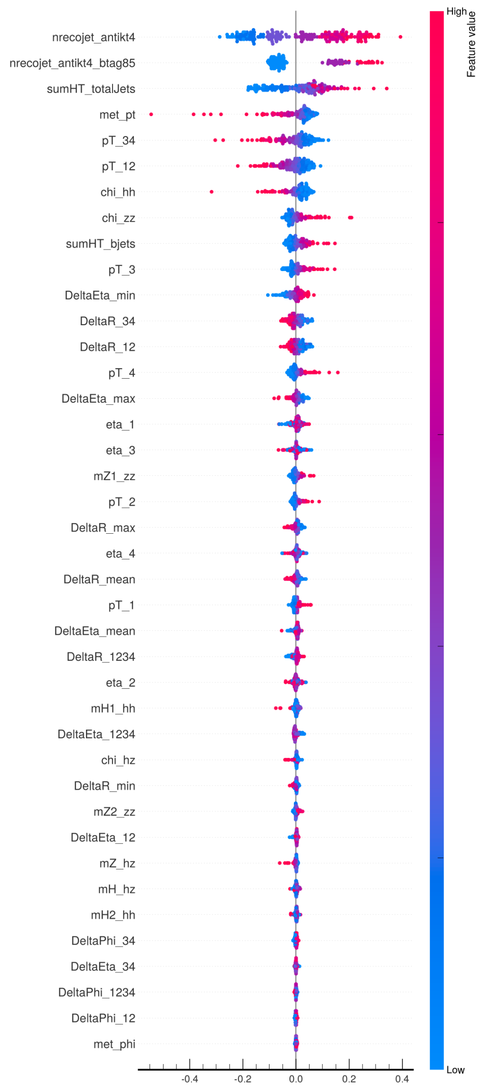
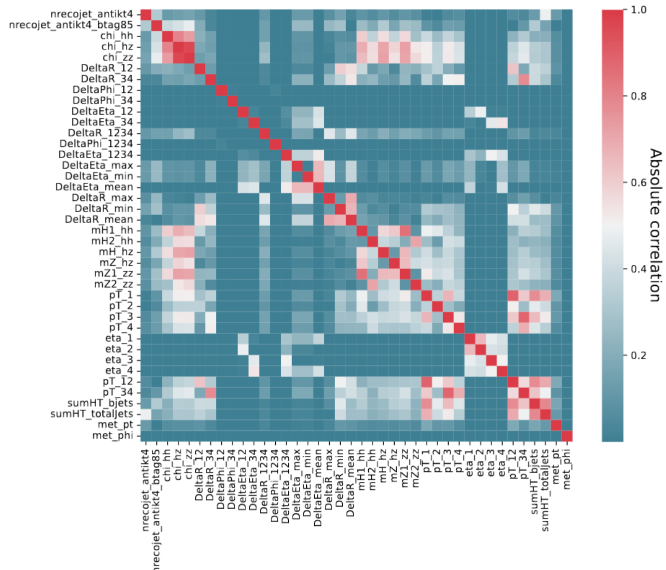
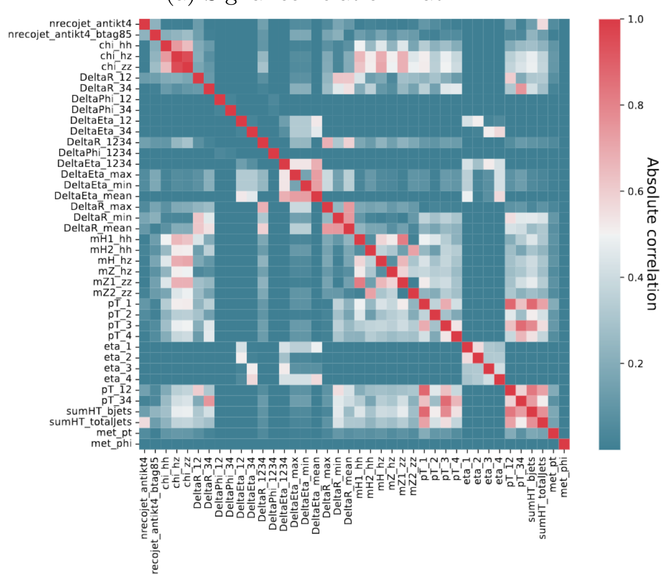


Figure 5: Shapley Beeswarm Plot.

It is clear, once again, that the number of jets in the event is the most important variable in determining whether an event is signal or background. The next most important variable is the sum of transverse momentum from all of the jets, which is therefore necessarily correlated to the number of jets. In general, an analysis of the input variables shows that there is little discriminating power in most of them, which signifies that if we were to apply stricter cuts (minimum 6 jets) the DNN would not be able to discern signal from background. The situation could improve if the background sample contained more events with a high number of jets.



(a) Signal correlation matrix.



(b) Background correlation matrix.

Figure 6: Correlation matrices.

As a last indicator of the quality of the DNN predictions, we plotted correlation matrices for both signal and background (Figure 6). The outcome is somewhat anticipated (e.g., sum of transverse momentum being correlated with the individual momentums). However, there remains uncertainty regarding why ΔR exhibits no correlation with $\Delta\phi$ and $\Delta\eta$, given that it is calculated as the square root of the sum of the squares of these two quantities.

6 Acknowledgments

I wish to express my profound gratitude to CERN and my supervisors for presenting me with the extraordinary opportunity of participating in this summer's internship. Furthermore, I extend my appreciation to Giulia Di Gregorio, Noemi Calace, and Luis Falda Coelho for their invaluable assistance, which I frequently required.