AG Kröninger May 2025
Technische Universität Dortmund

Lab course
"Search for $t\bar{t}$ resonances with ATLAS data"

Benedikt Gocke, Aaron van der Graaf,
Dr. Salvatore La Cagnina, Prof. Dr. Kevin Kröninger
(benedikt.gocke@cern.ch, aaron.vandergraaf@tu-dortmund.de,
salvatore.lacagnina@tu-dortmund.de kevin.kroeninger@cern.ch)

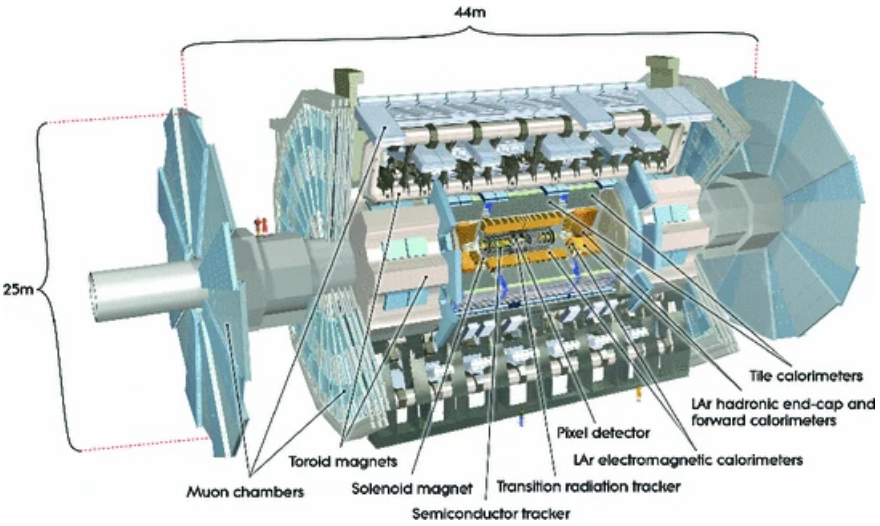

Figure 1: Sketch of the ATLAS detector [1].

1 Introduction

The goal of this lab course is to get an impression of data analysis in modern high-energy physics, where research is done in large collaborations due to the complexity of the machines and detectors. For this lab course, data from the ATLAS experiment[1] (Fig. 1) at the Large Hadron Collider (LHC) at CERN are used to search for new particles decaying to a top quark and a anti top-quark. At the LHC, protons are collided at a rate of up to 40 MHz. Hence, the detectors positioned around the collision points collect datasets of enormous size. In order to analyze these big data specialized methods are used. The structure of this lab course follows the steps of a prototypal data analysis: Analysis code is written in C++ to reduce the size of the dataset using an *event selection* which increases the signal-to-background ratio. The properties of the selected events are then studied to define a final discriminant , which may show a falling background spectrum on top of which the signal would show up as a sharp peak. Monte Carlo (MC) simulations are used to model the background spectrum and the *agreement of MC simulations and data* is studied in order to test the validity of the MC simulations. Finally, a *statistical analysis* is performed using the final discriminant in order to measure the amount of signal on top of the predicted background distribution.

Prerequirements for this lab course are introductory knowledge about particle physics (at the level of *Einführung in die Kern- und Teilchenphysik / introduction into particle physics lecture*) and basic knowledge about programming in C++ (at the level of *Einführung in die Programmierung, Eidp / introductory lectures to C++*), preferrably including some experience with Unix-based operating systems, such as Linux or OSX. Basic knowledge of the concepts of statistical data analysis (*Statistische Methoden der Datenanalyse, SMD / lecture on statistical methods*) are helpful, but not required.

The top quark is the most massive elementary particle known to date. It was discovered in 1995 in $p\bar{p}$ collisions at the Tevatron at Fermilab and completes the third quark generation as the partner of the b-quark. Since 2010, top quarks are also produced at the LHC. The dominant production mechanism is top-quark pair ($t\bar{t}$) production via the strong interaction. Top quarks decay via the weak interaction into a W boson and a down-type quark. In almost 100% of the cases, the down-type quark is a b-quark. The W boson decays further to either a charged lepton and the corresponding neutrino, $W^+ \to \ell^+\nu_\ell$, or to a pair of quarks, $W^+ \to u\bar{d}$ or $W^+ \to c\bar{s}$. The anti-top quark decays correspondingly. Hence, top-quark pair production is categorized according to the charged lepton multiplicity into the dilepton channel, where both W bosons decay leptonically, the lepton+jets channel, where one of the W bosons decays leptonically, and the all-hadronic channel, where both W bosons decay hadronically. A Feynman diagram of $t\bar{t}$ production in gluon-gluon fusion and its decay in the lepton+jets channel is shown in Fig. 2.

The final-state particles from the $t\bar{t}$ decay give rise to a distinct signature in the ATLAS detector. In most top-quark analyses, final states with at least one charged lepton are used, because background processes with high-momentum leptons only arise via the electroweak interaction, and thus they have much lower cross sections than processes without leptons via the strong interaction. Final states with electrons and muons are preferred because the identification of τ leptons is challenging given the variety of different τ lepton decays. The quarks from the top-quark decay hadronize and form jets which consist of the color neutral products of the hadronization process. The particles from the $t\bar{t}$ decay interact with the detector and are reconstructed from the detector signature by sophisticated algorithms specialized for the identification of electrons, muons and jets. It is important to distinguish the particles from the $t\bar{t}$ decay, which are not directly accessible from the detector information, and their reconstructed counterparts, often referred to as *reconstructed objects* or *offline objects*.

[1] The data were taken with the ATLAS detector in proton-proton collisions at $\sqrt{s} = 13$ TeV and correspond to an integrated luminosity of 1 fb^{-1}. The collision data are available on the CERN Open Data portal *http://opendata.atlas.cern* together with detailed Monte Carlo simulations for several processes.

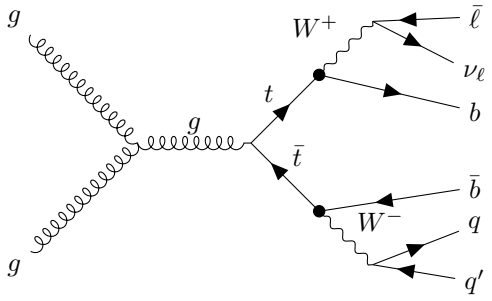

Figure 2: Feynman diagram of $t\bar{t}$ production via gluon-gluon fusion and its decay in the letpon+jets channel.

For almost all parts of data analysis in high-energy physics, MC simulations play an essential role. Such simulations are designed to look like real data and provide indeed a quite accurate description of data in most cases. Such MC simulations use specialized *generators* to produce events for the process studied, for example $pp \to t\bar{t}$. The particles produced are then decayed, hadronized etc. until stable particles are obtained. Here, "stable" means stable on the length scales of the ATLAS detector, so that for example charged pions, muons or kaons classify as stable. These particles are then propagated to enter a very detailed simulation of the ATLAS detector, which includes every single readout channel, the exact layout of each sensor and its readout electronics, the exact position of cables, and other non-active material etc. In simulation, information is available on the generated final state before entering the detector. Information on these *truth particles* are not available in data, of course. However, they can be used to optimize the strategy of the data analysis using MC simulations.

Given the large amount of collisions per second, data reduction happens on-the-fly or online using specialized triggers. Collision data can only be stored permanently with a rate of a couple of 100 Hz, so that the decision on whether a specific event is to be stored or not has to be taken on very short timescales. For analyses with electrons or muons in the final state, lepton triggers are used, which are based on a simplified and fast version of the electron and muon identification algorithms used for the definition of the offline objects. Triggers require these identification criteria and a minimum (transverse) momentum in order to reject background contributions with low-energy leptons. A typical threshold is 25 GeV.

Further information is available which helps distinguishing events with top quarks in the final state from background events. In $t\bar{t}$ decays with charged leptons in the final state, also one or two neutrinos are present. Since neutrinos do not interact with the detector, they cause an apparent imbalance of momentum in the detector, the so called *missing transverse momentum*, $E_{\mathrm{T}}^{\mathrm{miss}}$. The missing transverse momentum is defined as the vectorial sum of the transverse momenta of all reconstructed objects in an event. Transverse momenta are defined in the plane transverse to the beam axis. Given that in pp collisions the partons inside the protons interact to produce top quarks, only the total momentum of the colliding partons in the transverse plane vanishes before the collision, but the total momentum along the beam axis (the z-direction) may be non-zero and remains unknown. Hence, the momentum balance in z-direction cannot be used to estimate the z-direction of the neutrino(s). Also, jets originating from b-quarks can be distinguished from jets originating from so-called *light quarks* (i.e. u-, d-, s- and c-quarks) and from gluons by the identification of the decay vertex of a B-hadron within the jet. Such jets are called *b-jets* or *b-tagged jets*.

While the focus of this lab course is on techniques for studying top quarks, the example studied is an extension of the Standard Model of particle physics (SM) in which a massive resonance decaying to top quarks is predicted. The details of such extensions are not important for this lab course. How-

ever, it should be known that the SM is believed to only be the limit of a more global theory given a set of shortcomings of the SM. Such a global theory would limit the validity of the SM to energies below a certain new energy scale, which in many theories, or extensions of the SM, is of the order of one or several TeV, i.e. in reach with the energy scales accessible at the LHC. Famous examples of shortcomings of the SM are: (i) it does not explain the presence of dark matter in the universe, (ii) it does not include gravity in the theoretical framework, (iii) it does not explain the mass hierarchy of the fermions, (iv) it does not describe lepton-flavor violation in neutrino oscillations, (v) it does not explain the corrections to the Higgs boson mass, which diverge at high energies ("hierarchy problem"). Extensions of the SM, or *beyond-the-SM* (BSM) models, target one or several of these shortcomings by the prediction of new dynamics (some new force) and/or new particles, such as particles decaying to $t\bar{t}$. Typically, the exact mass of such a resonance is not predicted by the new model, but it rather is one of its free parameters. One such example is a massive Z' boson decaying to a top quark and an anti-top quark with a mass larger than 500 GeV and otherwise properties similar to those of the SM Z boson. This hypothetical particle serves as a benchmark for the $t\bar{t}$ resonance search in this lab course. It is called a benchmark, because if it was found, it would not be clear if it was indeed a Z' or some other particle decaying to $t\bar{t}$. However, it would certainly cause a scientific sensation!

Further information on top-quark physics can be found in Refs. [2, 3].

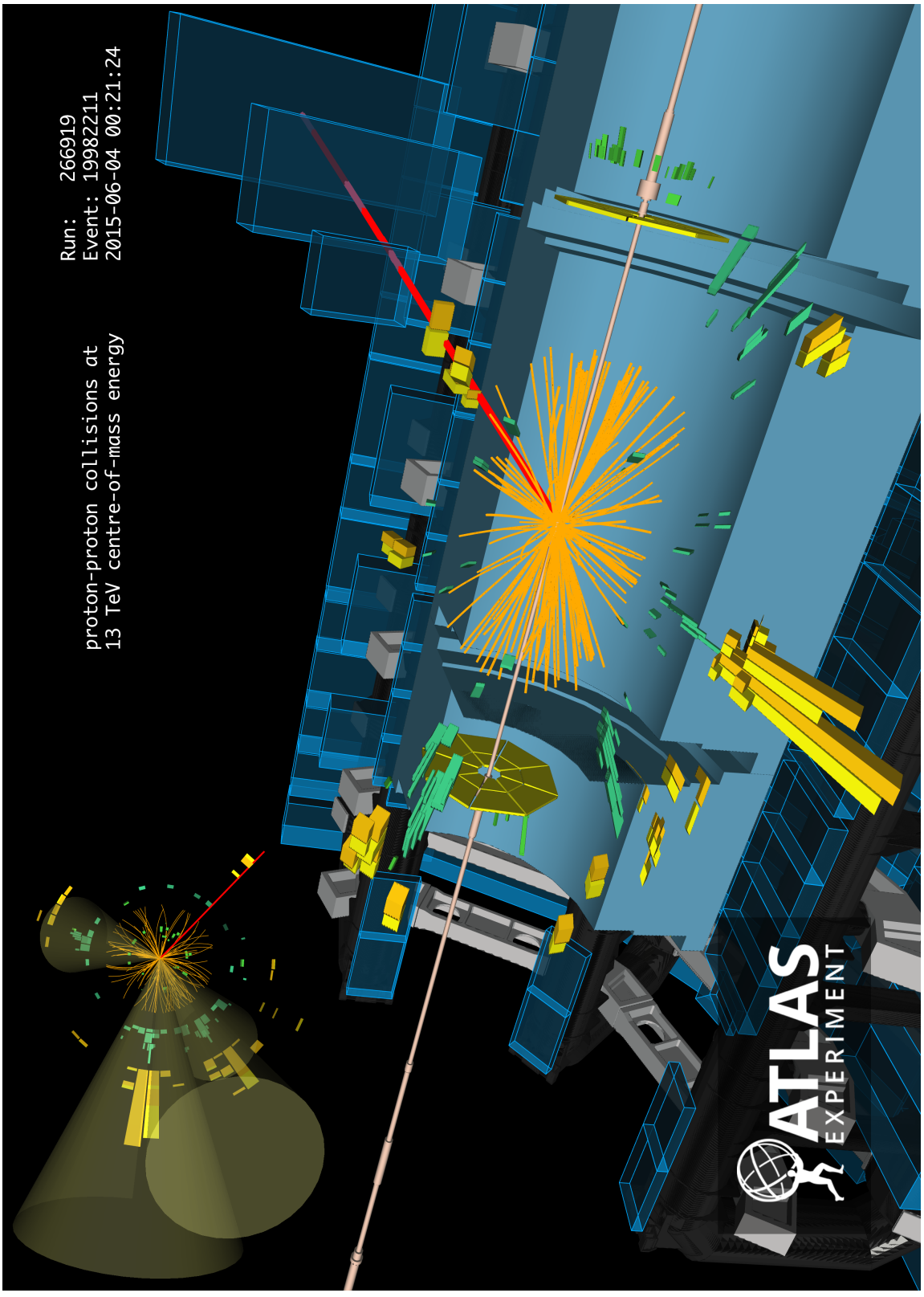

Figure 3: Example event display of a $t\bar{t}$ candidate event in 13 TeV proton-proton collisions recorded with the ATLAS experiment. The event fulfills the requirements for a lepton+jets event with one reconstructed muon and four jets with transverse momentum larger than 25 GeV. The green and yellow bars indicate energy deposits in the liquid argon and scintillating-tile calorimeters. Tracks reconstructed from hits in the inner tracking detector are shown as arcs curving in the solenoidal magnetic field [4].

1.1 Questions and tasks

In order to prepare for the discussion at the start of the lab course, the studies during the lab course, as well as the written report, remind yourself of some basic concepts of particle physics. Look up some special properties of the top quark and learn about a basic concept of statistical data analysis:

a) Draw leading-order Feynman diagrams for $t\bar{t}$ production at the LHC for

   (a) quark-antiquark annihilation $q\bar{q} \to t\bar{t}$,

   (b) single top-quark prduction via the weak interaction in the t-channel, $qb \to q't$.

b) Have top quark pairs been produced at an e^+e^- collider, yet? - If yes: At which collider? If not: Why not?

c) Assume a Z' with a mass of 700 GeV was produced in e^+e^- collisions. Sketch the total e^+e^- cross section as a function of the center-of-mass energy, $\sqrt{s}$, in the range 20-1000 GeV.

d) Estimate the branching faction of the W boson to its different final states by simply counting all possible final states (*quarks have color!*). Now, estimate the branching fractions of the dilepton, lepons+jets and all-hadronic $t\bar{t}$ decay channels.

e) What are the strenghts and weaknesses of the dilepton, lepton+jets and all-hadronic $t\bar{t}$ decay channels in terms of branching fraction and background suppression?

f) Identify the different components of the ATLAS detector and explain their purpose (Fig. 1).

g) What is the definition of the pseudorapdity η? Which regions in the detector do $\eta = 0$ and $\eta = \pm\infty$ correspond to? Why is the geomerical acceptance (A) of the detector not 100% and what is the rough range of the geometrical acceptance in $|\eta|$ of the ATLAS detector?

h) What does the quantity $\Delta R = \sqrt{|\Delta\eta|^2 + |\Delta\phi|^2}$ describe?

i) What are *boosted* objects or *boosted* events? How do these occur? Are (objects) in events with a Z' boson expected to be boosted? Why? What applies to ΔR between two boosted objects?

j) What are the signatures of high-momentum electrons, muons and jets in the ATLAS detector?

k) Have a look at the event display in Fig. 3, i.e. a sketch of the ATLAS detector with indicated signatures of the reconstructed objects. Identify the different objects based on their signature in the detector.

l) What is the χ^2 probability and what is the role of the χ^2 value divided by the number of degrees of freedom, often referred to as χ^2/n.d.f., of the problem stashed?

m) What is the world's best measurement of the top-quark mass to date and its publication reference? (*Wikipedia is NOT a scientific reference. Also, the PDG book(let) may not contain the latest top-quark mass measurement as itis only updated every two years.*)

n) Research the curent experimental status of the search for $t\bar{t}$ resonances from either the ATLAS or the CMS collaboration. Have hints for the existence of $t\bar{t}$ resonance been found yet? If yes: At which experiment and in which mass range? If not: What are the current experimental limits on its production cross section and its mass? Decide on one paper, read it and bring it to the lab course for discussion. (pdf document is sufficient!)

2 Prepare the analysis code as well as the data and simulation samples

To do the lab course you need to use a LDAP account. When you do this lab course, you get a *ssh configuration file* send from the instructor, as well as an *ssh key*. Both needs to be placed in your `.ssh` folder. In case you already have a customized config yourself, you need to add the content of our ssh config to your config file. Afterwards you need to open a new shell and type

```
ssh Zprime
```

This command should connect you to the TU Dortmund university infrastructure of the AG Kröninger. You can test if everything worked using

```
pwd
```

and the output should be

```
/nfs/homes/zprime/zprimeX
```

where X depends on your assigned zprime account number.

The folder CODE contains twelve files and one folder for the analysis in C++.

- `Makefile`: This file is used to compile the relevant pieces of C++ code using the command MAKE. The command `make clean` can be used to clean the folder from all files created during compilation. This may be useful in case the compilation partially failed. *There is no need to modify this file during the lab course!*
- `runSelection.C`: This is the main script for running and implementing the event selection. (see Sec. 3). Compiling with make will create an executable `runSelection.exe`.
- `plotDistribution.C`: This is the main script fo creating histograms of properties of the events obtained after the event selection (see Sec.4 and 5). Compiling with make will create an executable `plotDistribution.exe`.
- `stackedPlots.C`: This is the main script for creating so-called *stack plots* (see Sec. 6) where the contributions from different background sources are stacked on top of each other in order to compare with the measured data. Compiling with `make` will create an executable `stackedPlots.exe`.
- `chiSquare.C`: This is the main script for performing the χ^2 test for the statistical analysis on the final discriminant (see Sec. 7). Compiling with `make` will create an executable `chiSquare.exe`.
- `mini.h` and `mini.cxx`: These two files provide a C++ class that mirrors the structure of the data and simulation files for handy access to their content. *There is no need to modify these files during the lab course!*
- `fileHelper.h` and `fileHelper.cxx`: These two files provide helper functions for reading the data and simulation files and writing output histograms. *There is no need to modify these files during the lab course!*
- `Neutrino.h`, `NeutrinoReco.cc` and `physicsHelper.h`: These three files hold an implementation of how to calculate the momentum of the neutrino in lepton+jets $t\bar{t}$ decays from the momenta of the other identified objects (see Sec. 5). *There is no need to modify these files during the lab course!*
- `output_runSelection` is an empty folder which will hold the output of the event selection (see Sec. 3)

The data files are located at the following path. Please do not copy these files to a different location, because the code for this lab course accesses the files at this location (and also other participants of this course!): `/ceph/e4/users/bgocke/Zprime/Samples`

Different files correspond to different processes. All files not having the prefix data contain MC simulated events for the different processes.

- `diboson`: pair production of W and Zboson, i.e. WW, WZ and ZZ production
- `singleTop`: single top quark production
- `ttbar`: $t\bar{t}$ production
- `Wjets`: W boson production is association with jets
- `Zjets`: Z boson production in association with jets
- `Zprime`: production of a hypothetical Z' boson with varying mass (from 400 GeV to 3000 GeV): this is the particle we search for in this lab course. The mass of the Z' boson is a free parameter of the theory

There are also two example files, which have the suffix *example*. These can be found in your `zprime` home directory. Please only use these for the following instructions in this section!

The different files are referred to as data samples or just samples. They are in ROOT[2] [5] format and contain so-called *ntuples* in ROOT's data type `TTree`. ROOT is a framework providing a large selection of classes useful for data analysis in particle physics. It is widely used in the scientific community. ROOT also comes with a C++ interpreter, which is automatically setup when you log in onto one of the workstations. Caution: Whenever you need to save root files into a new directory, ROOT does not create it but you have to create it beforehand! Otherwise, it will throw an error.

To open an interactive root session you just type

```
root -l <path to file>
```

in the terminal. Once you have opened an interactive ROOT session, it allows you to interactively type C++ commands, including specific ROOT commands. This interpreter is different from the typical use of C++ as a compiled language. The prompt should look similar to this:

```
root [0]
```

The interpreter is not used for more than getting familiar with the content of the ntuples in this lab course. For the rest of the lab course, properly compiled C++ code is used. Open a ROOT browser in the interactive ROOT session:

```
new TBrowser()
```

Then, double-click on the file name you opened. You should see an object with name `mini`, which is the name of the `TTree` in all your ntuples. Open it by double.clicking on it. You should now see the content of the ntuple starting with `runNumber, eventNumber` etc. For each event, the ntuple contains general information (such as the event number) and the information about the reconsructed objects. You can double-click on the different *branches* of the `TTree` and `TBrowser` will plot the content of the respective branch *for all events*. Have a look at some of the branches. Warning: Some branches are not resovled properly via the TBrowser, as the automatically chosen binning is not appropriate. In this case, you can use the `Draw` command in the ROOT interpreter to plot the content of the branch. For example, to plot the transverse momentum of the jets, you can type

[2]Introductory ROOT tutorials can be found here: `https://root.cern.ch/introductory-tutorials`.

```
root -l <path to file>
TTree *mini = (TTree*) _file0->Get("mini")
mini->Draw("jet_pt>>h1(50,0,200000)")
```

and change the numbers to your needs. The first number is the number of bins, the second number is the lower limit and the third number is the upper limit of the histogram. You can also use the `Draw` command to plot two branches in one histogram.

A short description of their content is shown in the following table. You can look up the data type of each branch in the file `mini.h`

In order to reduce the size of the datasets to a manageable level, the datasets that are provided to you have a so-called preselection already applied. You will clearly see the effect of the preselection when analyzing the ntuples. Obviously, the preselection is designed not to reject many signal events, but to mostly remove background events.

2.1 Questions and tasks

a) Open the `ttbar_example.root` file with ROOT. Select one plot from the `TBrowser` and save it. What does the plot show? Briefly describe the features of the plot.

b) In the interactive ROOT session you can access the number of events in the `mini` ntuple by typing `mini->GetEntries()`. How many entries does the ntuple contain? Does the number of entries in the ntuple match the number of entries in the plot from a)? If it does, find a branch, where the number of entries in the plot differs from the number of entries in the ntuple. Why are there more entries in the plot than entries in the ntuple and what is the data type of the branch?

c) The samples provided have a preselection applied in order to reduce their size to a manageable level. Guess the preselection from the distributions in the `TBrowser`

- Was there probably a requirement on the electron or muon trigger?
- What was probably the requirement on the number of reconstructed leptons?
- Were there probably requirements on the minimal transverse momentum (p_{T}), the pseudorapidity (η) and the azimuthal angle (ϕ) of the leptons and jets?

runNumber	number uniquely identifing ATLAS data-taking run
eventNumber	event number and run numer combined uniquely identifies event
channelNumber	number uniquely identifying ATLAS simulated dataset
mcWeight	weight of a simulated event
SumWeights	generated sum of weights for MC process
XSection	total cross-section, including selection efficiency and higher-order correction factor
jet_E	energy of the jet
jet_MV2c10	output from the multivariate b-tagging algorithm MV2c10 of the jet
jet_eta	pseudo-rapidity, η, of the jet
jet_jvt	jet vertex tagger discriminant of the jet
jet_n	number of pre-selected jets
jet_phi	azimuthal angle, ϕ, of the jet
jet_pt	transverse momentum, p_{T}, of the jet
jet_trueflav	truth flavour of the simulated jet
jet_truthMatched	information whether the jet is matched to a simulated jet
lep_E	energy of the lepton
lep_charge	electric charge of the lepton
lep_eta	pseudo-rapidity, η, of the lepton
lep_etcone20	scalar sum of track E_T in a cone of $R = 0.2$ around lepton
lep_isTightID	information whether the lepton satisfies the tight ID reconstruction criteria
lep_n	number of pre-selected leptons
lep_phi	azimuthal angle, phi, of the lepton
lep_pt	transverse momentum, p_{T}, of the lepton
lep_ptcone30	scalar sum of track p_{T}, in a cone of $R = 0.3$ around lepton
lep_trackd0pvbiased	d_0 of track associated to lepton at point of closest approach (pca)
lep_trackd0pvunbiased	d_0 significance of the track associated to the lepton at the (pca)
lep_trigMatched	information whether the lepton is the one triggering the event
lep_truthMatched	information indicating whether the lepton is mathced to simulated lepton
lep_type	number signifying the lepton type (e or μ)
lep_z0	z-coordinate of the track associated to the lepton wrt. primary vertex
met_et	transverse energy of the missing momentum vector
met_phi	azimuthal angle, ϕ, of the missing momentum vector
scaleFactor_BTAG	scale-factor for b-tagging algorithm at 70% efficiency working point
scaleFactor_ELE	scale-factor for electron efficiency
scaleFactor_LepTRIGGER	scale-factor for lepton triggers
scaleFactor_MUON	scale-factor for muon efficiency
scaleFactor_PILEUP	scale-factor for pileup reweighting
scaleFactor_PhotonTRIGGER	scale-factor for photon triggers (not used here!)
scaleFactor_COOMBINED	product of all scale-factors (use this one!)
trigE	information whether the event passes a single-electron trigger
trigM	information whether the event passes a single-muon trigger

3 Define and implement the event selection

In order to search for a signal, the contributions from background processes need to be reduced, as background processes frequently have much higher production cross sections than the signal. The signal-to-background ratio is improved by applying an event selection targeting the final state objects expected from the signal process.

Clue I: A jet is called b-tagged at a certain efficiency working point (WP) if its MV2c10 value is larger than some threshold. For MV2c10 the threshold for the four calibrated b-tagging efficiency working points are in the following table:

WP	MV2c10 threshold
85%	0.11
77%	0.64
70%	0.83
60%	0.94

3.1 Questions and tasks

a) Which reconstructed objects do you expect from the Z' signal in the all-hadronic, lepton+jets and dileptonic $t\bar{t}$ decay modes? Also think about expected separation between the reconstructed objects in the detector!

b) In general, analyses targeting top quarks use b-tagging. Why? What are the advantages and disadvantages of requiring a jet to pass a more/less efficient b-tagging WP? The `scaleFactor_BTAG` variable stored in the ntuples is for the 70% WP. What does this mean to you if you want to use another b-tagging WP?

c) How do you propose to set up an event selection targeting the signatures in the all-hadronic, lepton+jets and dileptonic $t\bar{t}$ decay modes? Why is it necessary to require a minimum p_{T} value and a maximum $|\eta|$ value for the objects?

d) List all relevant background processes for the three different topologies. Which topology do you suggest to study in this lab course? Consider the amount of expected background, but also the branching fraction of the signal in the different decay modes.

Discuss your answers to these questions and to those from Sec. 2.1 with the supervisor of the lab course ***before*** *moving on.*

Compile `runSelection.C` by typing

```
make
```

If compilation succeeds, test running over the example file in simulation

```
./runSelection.exe <full path to file>/ttbar_example.root
```

and data:

```
./runSelection.exe <full path to file>/data_example.root
```

You always need to specify the full path to the input file. Now, have a look at the code in `runSelection.C`: The core part of the code is a for loop over all events, where each event is investigated and it is tested

whether the event fulfills the event selection requirements (to be) implemented in `runSelection.C`. All selected events are stored in an output ROOT file in the folder `output runSelection`. In the example above, you should have gotten two files called data_example_selected.root and data_example_selected.root, respectively. Both are of the same format as the original ntuples. Unfortunately, both also have the same size as the original ntuple, because no event selection is currently implemented in `runSelection.C`.

Clue II:The class `TLorentzVector` can be used to define the four-vectors of the reconstructed objects. It has several useful functions that can be used to calculate different quantities, like e.g. ΔR or the invariant mass. A four-vector can be defined with the known quantities using the function `SetPtEtaPhiE(Double t pt, Double t eta, Double t phi, Double t e)`.

d) Implement the event selection which you agreed on with the supervisor of the lab course. Test it on the $t\bar{t}$ example ntuple by verifying in a `TBrowser` that the distributions in the output ROOT folder look different compared to the distributions in the original ntuples in the way you expect them to vary after the event selection.

e) Also print out an table for the combined value of acceptance times efficiency ($A \cdot \epsilon$), i.e. the ratio of events after each of the different requirements in the event selection to the total number of events in the sample.

f) Once you have tested your implementation of the event selection, run it on all data and MC files (excluding the example files!) and document the efficiency tables for all processes. For running over all files, using a bash script with a loop over the different input files can be very useful. You can find an example script below which can e.g. be saved as a file `script.sh`. The script can be run by using the command `source script.sh`.

```
#!/bin/bash
for filename in <full path to files>/*; do
./runScript.exe $filename;
done
```

Attention: This step might take a few hours for all files combined! You can think about using `screen` or `tmux` for this!

g) Different requirements in the event selection target different background processes. Discuss which requirements suppress which background processes efficiently and which background processes have high selection efficiencies for certain requirements. How efficient is the event selection for the signal process for a Z' mass of 1000 GeV?

h) What is the main background process after the event selection?

4 Plot several fundamental distributions

Use the file `plotDistribution.C` to plot several basic distributions in order to have a detailed look at the content of the samples. Do this only for the $t\bar{t}$ sample! You can run the sample by first compiling the package with `make` and then running the executable with:

```
./plotDistribution.exe <full path to file>
```

One example plot is already implemented (the lepton p_{T}), but it is not, yet, stored in an output file. You can easily store the resulting histograms in a ROOT file using the helper function in the class `fileHelper`.

4.1 Questions and tasks

a) Plot the following distributions. Always choose a reasonable range for the x-axis and a reasonable amount of bins. Properly label the x- and the y-axis including units! Take a look at the class `TLatex` if you want to use Latex expressions for your labels.

   - the lepton p_{T}, η, ϕ, E
   - the p_{T}, η, ϕ, E of all jets
   - the p_{T}, η, ϕ, E of the jet with the largest p_{T}
   - the number jets
   - the number of b-tagged jets
   - the magnitude of the missing transverse momentum

b) Discuss the features of each plot and comment on whether these are in agreement with your expectations for a $t\bar{t}$ sample.

5 Plot several derived quantities

After the event selection, the signal-to-background ratio is significantly better than at preselection level. Still, a better discrimination of signal and background processes may be possible. Below, you can find a list of several quantities derived from the four-momenta of the reconstructed objects. The derived quantities are often referred to as *high level* observables, while the four-momenta of reconstructed objects are often denoted as *low level* observables. Investigate the discriminating power of the high level variables below and decide on which to choose as a final discriminating variable to identify a potential signal on top of the expected background distribution:

- $E_{\mathrm{T}}^{\mathrm{miss}}$, the magnitude of the missing transverse momentum;
- $\Delta\phi\left(E_{\mathrm{T}}^{\mathrm{miss}},\mathrm{lepton}\right)$, the difference in azimuthal angle between $E_{\mathrm{T}}^{\mathrm{miss}}$ and the lepton;
- the invariant mass of the system formed by the three jets with largest pT , i.e. the invariant mass of the vectorial sum of the three four-vectors;
- the invariant mass of the system formed by the four jets with largest pT, the lepton and the neutrino – the neutrino vector can be estimated using the function `Neutrino()` in `physicsHelper.h` from the lepton four-vector and the $E_{\mathrm{T}}^{\mathrm{miss}}$ two-vector;
- the pseudorapidity of the system formed by the four jets with largest pT, the lepton and the neutrino with the neutrino vector calculated in the same way.

5.1 Questions and tasks

a) Plot the distributions explained above for a Z' signal of mass 1000 GeV and for the $t\bar{t}$ background using `plotDistribution.C`. Always choose a reasonable range for the x-axis and a reasonable amount of bins. Properly label the x- and the y-axis including units!

b) Discuss the discrimination power of the different variables and decide on one. *Discuss your choice with the supervisor of the lab course.*

6 Check the agreement of simulation and data

A search for BSM physics on top of the background can only be performed if the sum of the background distributions are in reasonable agreement with the observed data in background-dominated regions of the phase space. Once good data-MC agreement is established, the distribution of the final discriminant is investigated for signs of discrepancies from the background-only hypothesis, i.e. signs of BSM physics!

Run your implementation of `plotDistribution` on real data and on all MC samples. If you run `plotDistribution` on real data, be sure to set the flag `isdata` to true!

You may have noticed the *weight*, w, already implemented in `plotDistribution.C`. This weight is needed to ensure the correct normalization of the MC samples. All simulated events need to be normalized to the integrated luminosity of the dataset, hence for data events the weight is equal to unity. The proper MC normalization is obtained by applying the mentioned *weight*, w, to each MC event. The expected number of events, assuming a perfect detector, is then given by

$$N_{\text{exp.}} = \mathcal{L} \cdot \sigma \tag{1}$$

with the integrated luminosity of the dataset $\mathcal{L}$ and the cross section of the process σ. The weight is then calculated as

$$w = \frac{\mathcal{L} \cdot \sigma}{N_{\text{exp.}}}. \tag{2}$$

This ensures when counting all events in a histogram (=sum over all events over the weights), it gives again $N_{\text{exp.}}$ for a perfect detector (=all objects reconstructed and detected). In reality, this is more complicated unfortunately. The simulated events themselves have weights, coming from the MC generator. The weight then becomes

$$w = \frac{\text{weight}_{\text{mc}} \cdot \mathcal{L} \cdot \sigma}{\text{SumWeights}} \tag{3}$$

Here, $\texttt{weight}_{mc}$ is the MC generator weight, and `SumWeights` are the sum of all MC generator weights for each event *before* any selection applied. This is the weight which is already implemented in `plotDistribution.C`.
If the sum over all selected events in a histogram is build now it becomes a constant factor $\mathcal{L} \cdot \sigma/\text{SumWeights}$ times the sum over the $\text{weight}_{\text{mc}}$ for each event. If all events are selected, this again leads to the weight for a perfect detector. But since a selection efficiency (e.g. from trigger) and detector acceptance effects have to be taken into account, the number of expected events becomes:

$$N_{\text{exp.}} = \mathcal{L} \cdot \sigma \cdot \epsilon \cdot A \tag{4}$$

You can get the (expected) number of events from the filled histograms with taking the integral over all bins.

6.1 Questions and tasks

a) Compare the number of background events in simulation with the number of observed events in data after the event selection.

b) Why is just comparing the number of events not sufficient to quantify good agreement between data and simulation?

c) Also retrieve the expected number of events for Z' events. How do the expected events differ for each assumed mass?

Use the file `stackedPlots.C` to implement the data-MC comparison plots, where the distribution in data is compared to the sum of all background processes properly normalized to the integrated luminosity of the dataset. The sum of all background processes is easily implemented in a so-called stack plot, where the distributions from the different background sources are stacked on top of each other. Different background sources are distinguished by using a different `SetFillColor(<color number>)` for each of them.

6.2 Questions and tasks

a) Make data-MC comparison plots for all distributions from Sec. 4 and for the final discriminant you have chosen in Sec. 5.

b) Judge the data-MC agreement by eye. Do you see any MC mismodeling in the distributions from Sec. 4 that we should hence worry about?

c) Judge the data-MC agreement of the final discriminant by eye. Why do you think you might have or might not have discovered signs of a Z' signal?

d) Use the `PlotStackWithRatio` function to quantify the data-MC agreement for the distributions from Sec. 4 and the final discriminant. Are there certain bins where the agreement is worse/better? Is a trend visible?

7 Statistical analysis

Use the file `chiSquare.C` to judge the agreement of data with the background-only hypothesis quantitatively using a χ^2 test. Check if you can claim an evidence or even an observation. What happens if you not only include MC stat uncertainties but also other experimental uncertainties?
In case there are no signs of a Z' signal, one can calculate exclusion limits. Even if no signs of new particles are found, exclusion limits are of prime interest for theorists developing BSM models, because any new model must respect the observed experimental bounds. Some popular models have lost support in this way in the past years in the light of searches performed with LHC data.

7.1 Questions and tasks

a) Use `chiSquare.C` to calculate the χ^2 value between data and the background-only hypothesis using all bins in the distribution of the final discriminant. What is the calculated χ^2 probability (p-value) for the given number of degrees of freedom?

b) Evidence of a signal can be claimed if the p-value for the background-only hypothesis is below $2.7 \cdot 10^{-3}(3\sigma)$. Observation of a signal can be claimed if the p-value for the background-only hypothesis is below $5.7 \cdot 10^{-7}(5\sigma)$. Can you claim either evidence or observation for Z' production with your analysis?

c) Repeat the χ^2 test, but introduce a flat uncertainty of 14% (= 4% uncertainty from luminosity + 10% estimated systematic uncertainty from different sources like b-tagging, lepton identification...). Does the result of the χ^2 test changes? If so, discuss why. Also check again if you can claim evidence or an observation. Did the p-values change?

d) When one cannot claim any of the above, the next step is to set 95% CL exclusion limits on the production cross section, i.e., for each mass hypothesis the maximal signal on top of the background distribution is calculated which is still compatible with the data. This is the case when the χ^2 probability of the comparison of background+signal (!) and data is smaller than $1-0.95 = 0.05$. This is the so-called observed 95% CL exclusion on the Z' production cross section.
In Figure 4 you can see a plot showing the theoretical prediction and these exclusion limits. Explain the plot. What is shown? Discuss what information the plot gives about Z'-bosons and their hypothetical masses?

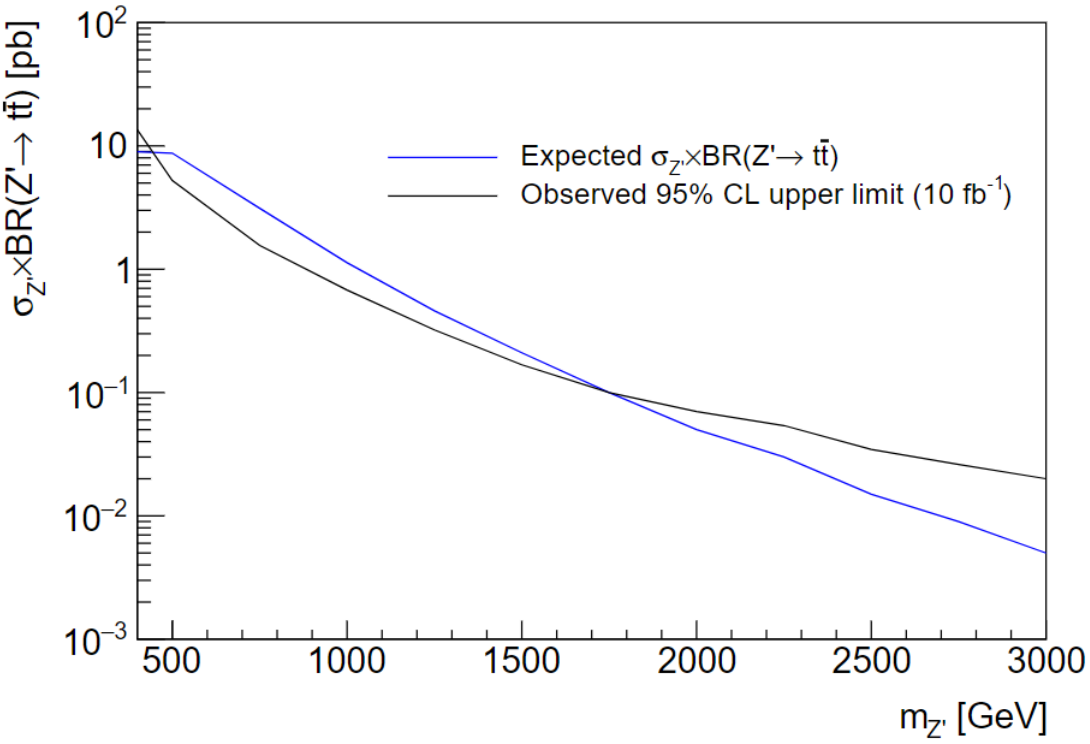

Figure 4: Theoretical limits and 95% CL exclusion limits on the production cross section for different Z'-boson masses.

8 Remarks on the written report

This lab course is organized in a similar way as published analyses in high-energy physics. As you have observed while reading the paper as part of the tasks in Sec. 1.1, scientific publications in the field are also written up in a similar order. Hence, the written report should naturally reflect the structure of such publications. Please use the following structure for your report:

- Start with a *short* (!) introduction (max. 1 page).
- Include important answers from the "Questions and tasks" of each section. Think about which ones make sense and are essential to demonstrate how you did the analysis and obtained the results. Pay attention to the flow of the report!
- Close with a paragraph with conclusions, which should not just be a summary of what is included in the report, but should in particular reflect on the lab course, possibly touching what you have learned, how you judge the work done in this lab course in the larger scientific context, whether important steps towards a publishable data analysis would still need to be done etc.

References

[1] ATLAS Collaboration. "The ATLAS Experiment at the CERN Large Hadron Collider: A Description of the Detector Configuration for Run 3". In: (May 2023). arXiv: 2305.16623 [physics.ins-det].

[2] R. L. Workman et al. "Review of Particle Physics". In: *PTEP* 2022 (2022), p. 083C01. DOI: 10.1093/ptep/ptac097. URL: https://pdg.lbl.gov/2022/web/viewer.html?file=../reviews/rpp2022-rev-top-quark.pdf.

[3] K. Kröninger, A. B. Meyer, and P. Uwer. "Top-Quark Physics at the LHC". In: *The Large Hadron Collider*. Springer International Publishing, 2015, pp. 259–300. DOI: 10.1007/978-3-319-15001-7_7. URL: https://doi.org/10.1007%2F978-3-319-15001-7_7.

[4] ATLAS Collaboration. accessed 09.10.2023. URL: https://twiki.cern.ch/twiki/bin/view/AtlasPublic/TopPublicResults.

[5] URL: https://root.cern.ch.