Lehrstuhl E IV March 2016

Technische Universität Dortmund

Lehrstuhlversuch
“Search for $t\bar{t}$ resonances with ATLAS data”

Sonja Bartkowski, Dr. Johannes Erdmann, Prof. Dr. Kevin Kröninger
(sonja.verena.bartkowski@cern.ch, johannes.erdmann@cern.ch,
kevin.kroeninger@cern.ch)

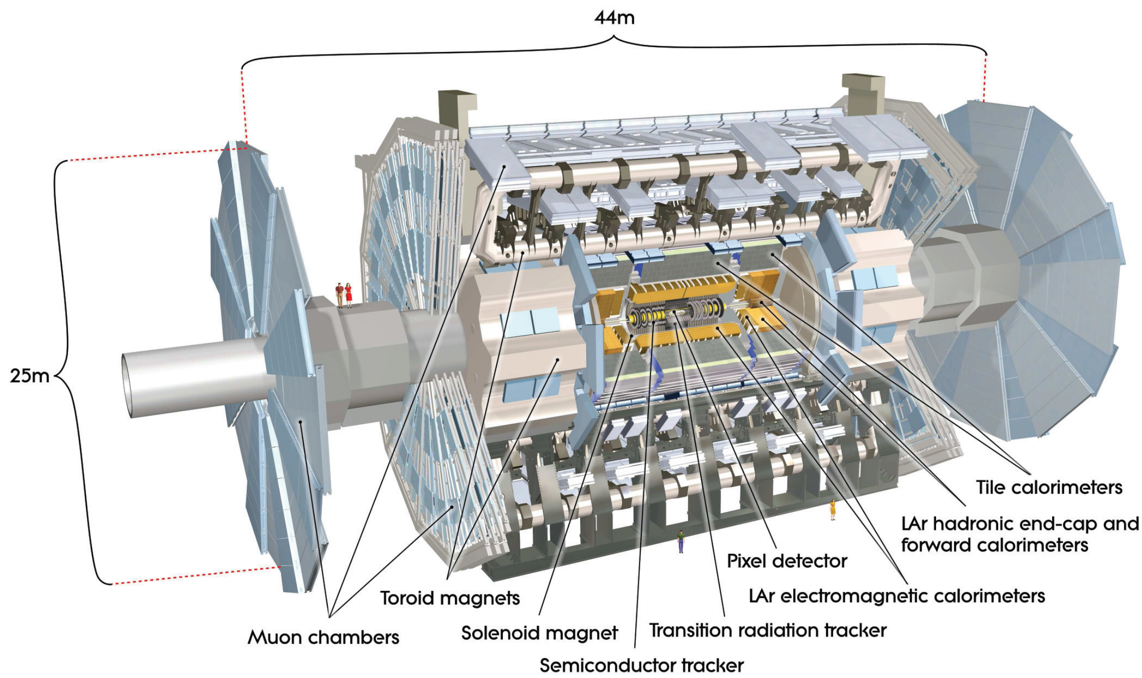

Figure 1: Sketch of the ATLAS detector [1].

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 an 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 / KET*) and basic knowledge about programming (at the level of, for example, *Einführung in die Programmierung / EINI*), 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*) 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 to 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

[1]The data were taken with the ATLAS detector in proton-proton collisions at $\sqrt{s} = 8$ TeV in 2012 and correspond to an integrated luminosity of $1\,\text{fb}^{-1}$. The collision data are available on the CERN Open Data Portal `http://opendata.cern.ch` together with detailed Monte Carlo simulations for several processes.

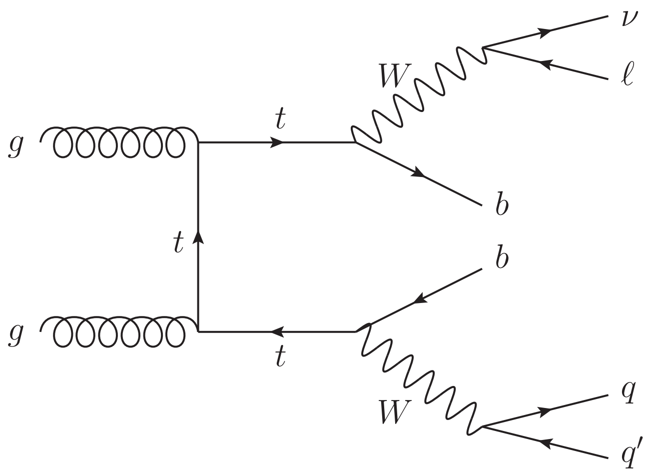

Figure 2: Feynman diagram of $t\bar{t}$ production via gluon-gluon fusion and its decay in the lepton+jets channel.

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*.

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*, $\vec{E}_{\mathrm{T}}^{\mathrm{miss}}$. 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. However, 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 hierarcy 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. [2–4].

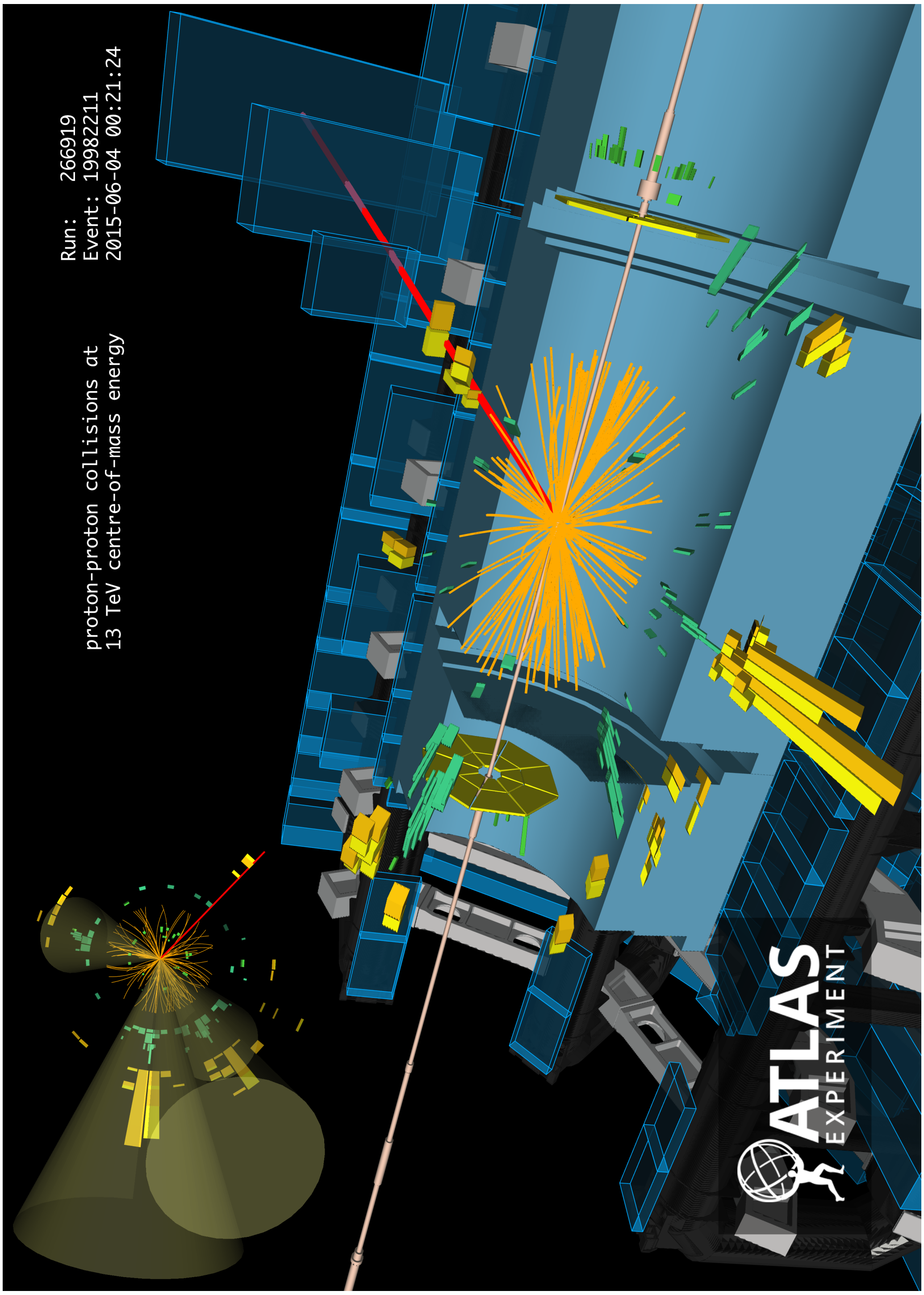

Figure 3: Example event display of a $t\bar{t}$ candidate event in 13 TeV proton-proton collisions recorded with the ATLAS experiment. The event fulfills the requirements for a μ+jets event with one reconstructed muon and four jets with transverse momentum larger than 25 GeV. The green and yellow bars 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 [5].

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

   1. quark-antiquark annihiliation $q\bar{q} \to t\bar{t}$,
   2. single top-quark production via the weak interaction in the t-channel, $qb \to q't$.

b) Have top quarks 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 fraction 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, leptons+jets and all-hadronic $t\bar{t}$ decay channels.

e) What are the strengths 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 pseudorapidity η? Which regions in the detector do $\eta = 0$ and $\eta = \pm\infty$ correspond to? Why is the geometrical 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 are the signatures of high-momentum electrons, muons and jets in the ATLAS detector?

i) 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.

j) What is the relativistic energy-momentum relation written in components? What is the definition of the invariant mass?

k) Assume that a particle A with mass m_A decays to particles b and c with masses m_b and m_c. Calculate the energy of particle b in the rest frame of particle A using four-momenta.

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 studied?

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 it is only updated once every two years.*)

n) Research the current 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.

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

To prepare for the lab course, create your working directory and copy over the content needed for the course[2]:

```
cd /afs/e4.physik.uni-dortmund.de/user/zprime
mkdir Mylastname_Myfirstname_Year_Month_Day
cd Mylastname_Myfirstname_Year_Month_Day
cp -r /afs/e4.physik.uni-dortmund.de/group/e4/teaching/zprime/Final/code .
cd code
```

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 for 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 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 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 files and writing the 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 selection (see Sec. 3).

[2] A username as well as the password to log in will be provided by the supervisor of the lab course.

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:
`ls /afs/e4.physik.uni-dortmund.de/group/e4/teaching/zprime/Final/samples`

Different files correspond to different processes. The suffix `.el` or `.mu` refer to files which contain events selected with an electron or muon trigger, respectively. All files not having the prefix `data` contain MC simulated events for different processes:

- `diboson`: pair production of W and Z bosons, i.e. WW, WZ and ZZ production;
- `singletop`: single top quark production;
- `ttbar`: $t\bar{t}$ production;
- `wjets`: W boson production in 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.

The different files are referred to as *data samples* or just *samples*. They are in ROOT[3] [6] 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 you can start by first setting up the ROOT environment on your work station:

```
swmod load root-system
```

and then just typing `root -l` followed by the path to a file in ROOT format:
```
root -l <path to file>
```

Now, you have opened an interactive ROOT session, which 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

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

about the reconstructed 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. A short description of their content is shown in Table 1. You can look up the data type of each branch in the file `mini.h`.

`runNumber`	run number
`eventNumber`	event number
`channelNumber`	channel number
`lbNumber`	luminosity block number
`rndRunNumber`	random run number
`mu`	average number of interactions per bunch crossing
`mcWeight`	MC event weight
`eventWeight_PRETAG`	total event weight before b-tagging requirement
`eventWeight_BTAG`	total event weight after b-tagging requirement
`pvxp_n`	number of primary vertices
`vxp_z`	vertex z-position
`scaleFactor_PILEUP`	pile-up scale factor
`scaleFactor_ELE`	electron efficiency scale factor
`scaleFactor_MUON`	muon efficiency scale factor
`scaleFactor_BTAG`	b-tagging efficiency scale factor
`scaleFactor_TRIGGER`	trigger efficiency scale factor
`scaleFactor_WJETSNORM`	W+jets normalization scale factor
`scaleFactor_WJETSSHAPE`	W+jets shape scale factor
`scaleFactor_JVFSF`	jet vertex fraction efficiency scale factor
`scaleFactor_ZVERTEX`	weight for vertex z-position
`scaleFactor_ALLPRETAG`	product of all scale factors before b-tagging requirement
`scaleFactor_ALLBTAG`	product of all scale factors after b-tagging requirement
`scaleFactor_COMBINED`	product of all scale factors (*use this one!*)
`trigE`	electron trigger fired
`trigM`	muon trigger fired
`passGRL`	passes the good runs list
`hasGoodVertex`	has a good vertex
`scaledWeight`	scaled weight
`lep_n`	number of leptons
`lep_truthMatched`	lepton is matched to truth lepton
`lep_trigMatched`	lepton is matched to the trigger object
`lep_pt`	lepton transverse momentum
`lep_eta`	lepton pseudorapidity
`lep_phi`	lepton azimuthal angle
`lep_E`	lepton energy
`lep_z0`	distance in z-direction of the lepton track to the vertex
`lep_charge`	lepton charge
`lep_isTight`	lepton fulfills tight identification criteria
`lep_type`	lepton type
`lep_flag`	lepton flag
`el_cl_eta`	electron cluster pseudorapidity
`lep_ptcone30`	lepton track isolation

`lep_etcone20`	lepton calorimeter isolation
`lep_miniIso10_4`	lepton mini-isolation
`lep_trackd0pvunbiased`	lepton impact parameter
`lep_tracksigd0pvunbiased`	lepton impact parameter uncertainty
`met_et`	magnitude of missing transverse momentum
`met_phi`	azimuthal angle of missing transverse momentum
`jet_n`	number of jets passing the jet vertex fraction requirement
`alljet_n`	number of all jets
`jet_pt`	jet transverse momentum
`jet_eta`	jet pseudorapidity
`jet_phi`	jet azimuthal angle
`jet_E`	jet energy
`jet_m`	jet mass
`jet_jvf`	jet vertex fraction
`jet_trueflav`	jet truth flavor
`jet_truthMatched`	jet is truth matched
`jet_SV0`	jet b-tagging discriminant from SV0 algorithm
`jet_MV1`	jet b-tagging discriminant from MV1 algorithm
`jet_flag`	jet flag
`jet_BadMediumBCH`	jet fails medium cleaning criteria for bad calorimeter regions
`jet_BadTightBCH`	jet fails tight cleaning criteria for bad calorimeter regions
`jet_emfrac`	jet energy fraction in the electromagnetic calorimeters
`jet_BCH_CORR_CELL`	jet energy bad channel correction
`jet_good`	jet is good, i.e. to be used

Table 1: Short description of the branches in the ntuples.

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) 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?

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: A jet is called b-tagged if its MV1 value is larger than 0.7892.

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?

b) How do you propose to set up an event selection targeting these signatures? Why is it necessary to require a minimum p_{T} value and a maximum $|\eta|$ value for the objects (as already implemented in the example code)?

c) 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. The background cross sections are listed in
`/afs/e4.physik.uni-dortmund.de/group/e4/teaching/zprime/Final/infofile.txt`
that you may either also copy to your working directory or open with a command line tool, such as `less`, `more` or `cat`.

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 an example file:

```
./runSelection.exe <full path to file>/data1.el.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 a file called `data.el_selected.root`, which is of the same format as the original ntuple. Unfortunately, it also has the same size as the original ntuple, because no event selection is currently implemented in `runSelection.C`.

d) Implement the event selection which you agreed on with the supervisor of the lab course. Test it on the $t\bar{t}$ sample in either the electron or muon channel by verifying in a `TBrowser` that the distributions in the output ROOT folder look different compared to the distributions in the original ntuple 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 ($A \cdot \varepsilon$), 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. The total number of events can be found in the `infofile.txt`.

f) Once you have tested your implementation of the event selection, run it on all data and MC 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 will be 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 /afs/path/*; do
    ./runScript.exe $filename;
done
```

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? Also consider the cross sections in the `infofile.txt`.

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 separately for events with electrons and events with muons, but 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 good jets;
- the number of good jets;
- the p_{T}, η, ϕ, E of the jet with the largest p_{T};
- the number of b-tagged jets (MV1 value larger than 0.7892);
- 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. Investigate the discriminating power of these variables 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;
- $\Delta\phi(E_{\mathrm{T}}^{\mathrm{miss}}, \text{lepton})$, the difference in azimuthal angle between $E_{\mathrm{T}}^{\mathrm{miss}}$ and the lepton direction (*be sure to always take the smaller difference of* $|\phi_1 - \phi_2|$ *and* $|2\pi - |\phi_1 - \phi_2||$, *because* $2\pi = 0$);
- the invariant mass of the system formed by the three jets with largest p_{T}, 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 p_{T}, 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 p_{T}, the lepton and the neutrino with the neutrino vector calculated in the same way.

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. $\Delta\phi$ 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)`.

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!

6.1 Questions and tasks

a) Calculate the number of expected events in the dataset for each background source using

- the total integrated luminosity of the dataset, $\mathcal{L} = 1\,\mathrm{fb}^{-1}$,
- the cross section, σ, for each process (see the `infofile.txt`),
- the selection efficiency from Sec. 3

using the well-known formula $N = \mathcal{L} \cdot \sigma \cdot (A \cdot \varepsilon)$.

b) Compare the sum of the expected background events to the number of events observed in data after the event selection.

c) Also calculate the expected number of events for the Z' signal for the different mass hypotheses.

Run your implemention of `plotDistribution` on real data and on all MC samples. Decide on one of the data samples and only use this sample in the following. If you run `plotDistribution` on real data, be sure to set the flag `isdata` to `true`! 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.

The proper MC normalization is achieved by using the total number of MC events before the preselection, N_{MC}, as given in the `infofile.txt`, and by multiplying each distribution by the following factor or *weight*: $w = \mathcal{L} \cdot \sigma / N_{\mathrm{MC}}$. The integral of each MC distribution will then correspond to the normalization you have calculated above.

6.2 Questions and tasks

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

c) 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?

d) 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?

7 Perform a statistical analysis

Use the file `chiSquare.C` to judge the agreement of data with the background-only hypothesis quantitatively using a χ^2 test. In case you cannot claim to have found signs of a Z' signal, set 95% confidence level (CL) limits on the production cross section of a Z' boson for the different Z' mass hypotheses. Comparing with the theory prediction for the cross section of the Z' signal, you can then claim to have excluded the presence of Z' bosons in nature up to certain Z' masses at 95% CL using the experimental data. 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 $2.7 \cdot 10^{-3}$ (3σ). Observation of a signal can be claimed if the p-value for the background-only hypothesis is below $5.7 \cdot 10^{-7}$ (5σ). Can you claim either evidence or observation for Z' production with your analysis?

c) If you cannot claim any of the above, move on to set 95% CL exclusions limits on the production cross section, i.e., for each mass hypothesis calculate the maximal signal on top of the background distribution still compatible with the data. Identify the value of the cross section from which on 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 production cross section. Plot the 95% CL excluded cross section as a function of the Z' mass.

d) On top of this plot, also show the expected Z' cross section as a function of the Z' mass, as given in the `infofile.txt`. Can you exclude a range of Z' masses at 95% CL? If yes, which range?

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).
- Thoroughly document your answers to the "Questions and tasks" sections from these instructions.
- 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*, JINST 3 (2008) S08003.

[2] T. Liss, F. Maltoni and A. Quadt, *The Top Quark* in Particle Data Group, *Review of Particle Physics*, Chin. Phys. C 38 (2014) 090001, `http://pdg.lbl.gov/2015/reviews/rpp2015-rev-top-quark.pdf` (update 2015).

[3] K. Kröninger, A. Meyer and P. Uwer, *Top-Quark Physics at the LHC* in T. Schörner-Sadenius (ed.), *The Large Hadron Collider – Harvest of Run 1*, Springer, 2015, arXiv:1506.02800 [hep-ex].

[4] G. Cortiana, *Top-quark mass measurements: review and perspectives*, submitted to Reviews in Physics, arXiv:1510.04483 [hep-ex].

[5] ATLAS Collaboration, *TopPublicResults twiki*, `https://twiki.cern.ch/twiki/bin/view/AtlasPublic/TopPublicResults#Event_displays_13_TeV`, accessed March 23$^{\mathrm{rd}}$, 2016.

[6] `https://root.cern.ch`.