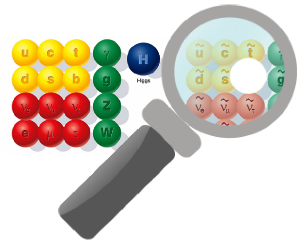
Table of Contents

Search for New Physics in the Multijet and Missing Transverse Momentum Final State (CMSDAS 2013@SINP)



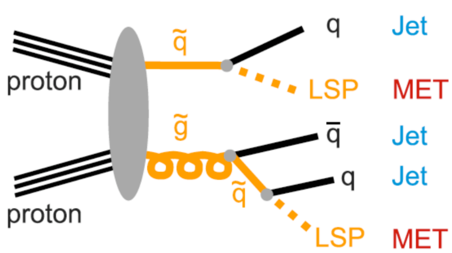
Introduction

For a brief introduction to the analysis motivation and the exercise, please have a look at the introductory presentation. For the overall school's programme, have a look at the indico agenda.

Overview and Analysis Strategy

This analysis is a **generic search for new physics**. It is **motivated by models of R-parity conserving Supersymmetry** (*SUSY*) that generally assume the existence of additional elementary particles. In the simplest case, there is one supersymmetric partner-particle for each Standard Model (*SM*) particle. The existence of these additional particles has many interesting consequences: Most strikingly, they can provide excellent candidates for Dark Matter, the substance that constitutes about 25% of our universe and for which there is no candidate particle in the SM! Moreover, the presence of supersymmetric partner particles in virtual loops can cancel the otherwise large corrections occurring in the renormalisation of e.g. the Higgs-boson mass. As a consequence, the theory becomes less dependent on the exact values of its parameters (becomes *more natural*) - a feature typically thought desirable. Also, the presence of supersymmetric particles in virtual loops leads to common gauge-coupling strengths when extrapolated to higher scales, which hints to some underlying, unified theory.

However, so far, SUSY particles have not been observed, and hence, they must have higher masses than the know SM particles. If somewhere in the TeV range, SUSY particles could be directly produced in the proton-proton collisions at the LHC. In many SUSY models, gluino-gluino, gluino-squark, and squark-squark pair production can occur with particularly **large cross sections** compared to other SUSY-production channels. The squarks and gluinos will predominantly decay into coloured SM particles and pairs of stable LSPs (*Lightest Supersymmetric Particles*). In realistic models, the LSPs are electrically neutral and only weakly interacting such as neutrinos.



Visible energy

$$H_T = \sum_{\text{jets}} |\vec{p}_T|$$

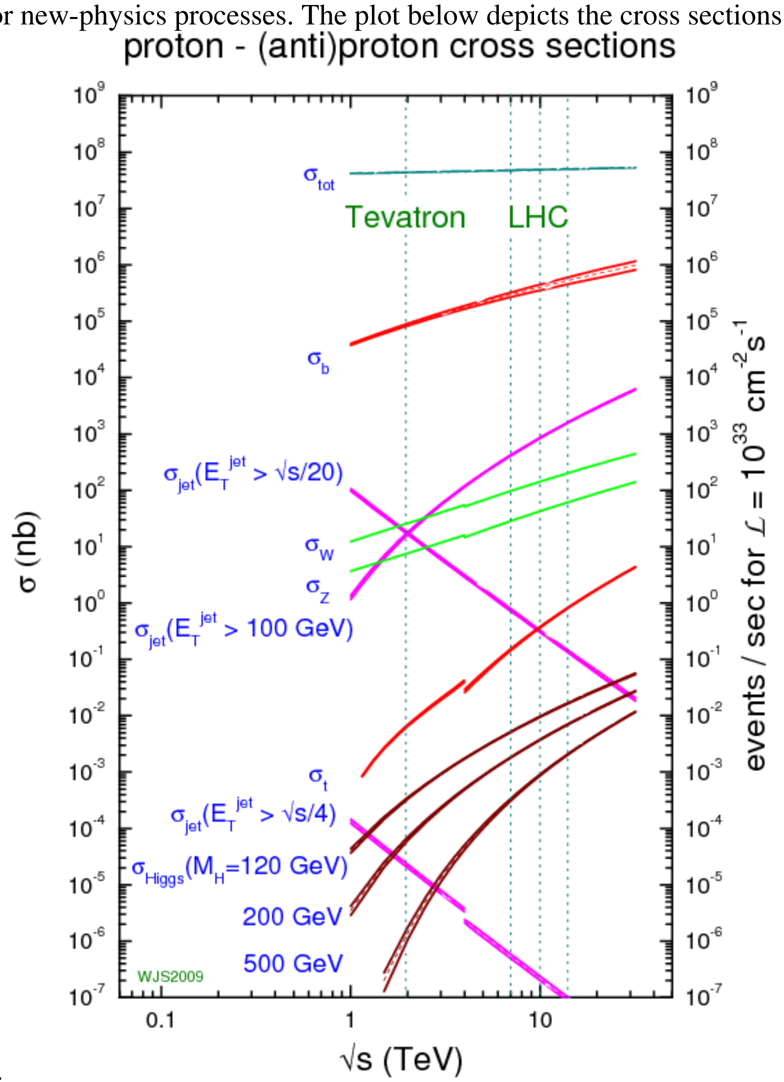
Energy of undetected particles

$$\not{H}_T = \Big| -\sum_{\text{jets}} \vec{p}_T \Big|$$

Thus, the SUSY signature we are looking for in the detector consists of **several jets with high p_T, large missing transverse momentum due to the LSPs, and no leptons**. Accordingly, the sensitive variables of our analysis are HT, MHT, and N(jets):

- HT, the scalar sum of the p_T of all jets with $p_T > 50$ GeV and | | < 2.5
- MHT, the negative vectorial sum of the p_T of all jets with $p_T > 30$ GeV and | | < 5
- N(jets), the number of jets with $p_T > 50$ GeV and | | < 2.5 (same selection as for HT)

However, there are also SM processes with the same signature (*SM background*). Therefore, we can only claim to observe a new process (like SUSY) when the observed number of selected events is significantly larger than the expected number of SM events. Hence, a precise knowledge of the expected SM background is extremely important when searching for new-physics processes. The plot below depicts the cross sections of

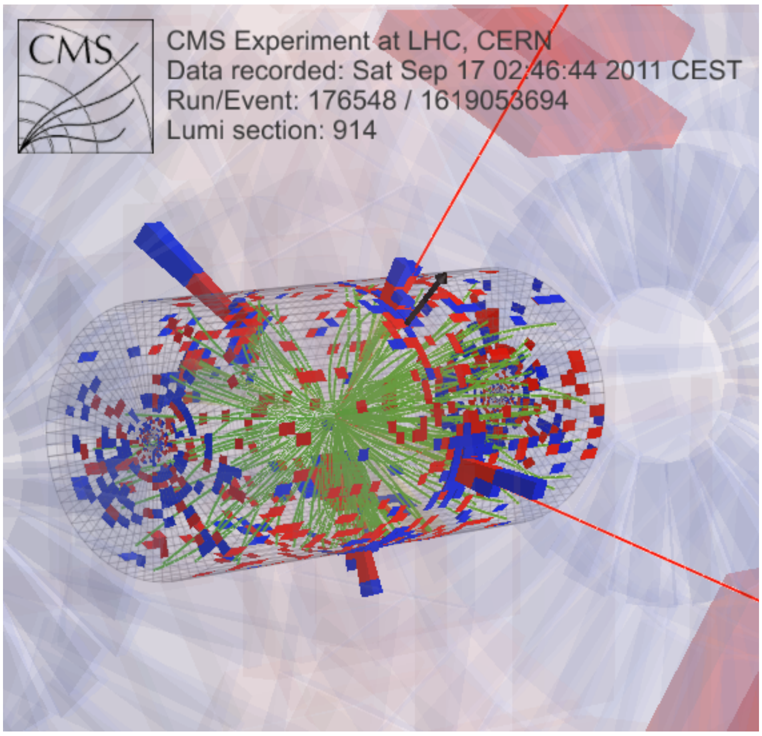


different physics processes at the LHC.
Expected SUSY cross sections are typically somewhere in the <1 pb regime, i.e. orders of magnitudes smaller than the SM background. The typical strategy to cope with this situation is to

- apply suitable event selection criteria to reduce the SM background compared to the expected signal;
- gain a precise understanding of the residual SM backgrounds to be sensitive to tiny excesses in the data.

In this analysis, the SM background is determined almost entirely from data. This is beneficial because the reliance on the simulation - and with it the uncertainity on the background rate - is minimised. **The data-based determination of the SM backgrounds is one of the key features of this analysis**.

Below, you see the visualised detector signature of a candidate event found in the 2011 data.



Publications and Documentation

The analysis, internally also termed RA2 (*Reference Analysis 2*), has been performed since the first LHC-data taking. By now, several versions have been published that successively included more data and ever-improved analysis methods.

- "Search for New Physics with Jets and Missing Transverse Momentum in *pp* collisions at √s=7 TeV", JHEP **1108** (2011) 155, arXiv:1106.4503 (36/pb)
- "Search for supersymmetry in all-hadronic events with missing energy", CMS-PAS-SUS-11-004 (2011) (1.1/fb)
- "Search for new physics in the multijet and missing transverse momentum final state in proton-proton collisions at √s=7 TeV", Phys. Rev. Lett. **109** (2012) 171803, arXiv:1207.1898 (5/fb)
- "Search for New Physics in the Multijet and Missing Momentum Final State in Proton-Proton Collisions at 8 TeV", CMS-PAS-SUS-13-012 (2013) (19.5/fb) (Currently in CWR)

Detailed information about the analysis techniques can be found in the corresponding public Physics Analysis Summaries (*PAS*) and internal Analysis Notes (*AN*) linked below in the References section.

The CMSDAS Exercise

This exercise will follow the recently published analysis of the 19.5/fb of data collected in 2012 at a center-of-mass energy of 8 TeV. The primary results are summarised below. **The focus of the exercise will be on understanding (some of) the data-based background determination methods** which are the heart of this analysis.

The exercise is performed with a series of simple ROOT-level C++ scripts and plain C++ classes. Some of them are incomplete, and you will be instructed on how to fill them up so that they can be run successfully. You will also be asked some questions meant to gauge your understanding of the topics being discussed. The data you will use have been prepared in a reduced format suitable for easy, interactive access and a fast return (ROOT *ntuple* format). The events store kinematic and other properties of higher-level objects, such as muons and jets, that have been reconstructed from the detector signals. In case of data, events have been preselected according to certain quality criteria that reject e.g. events affected by detector noise or malfuctioning detector components (*cleaning filters*). You can have a look at Ntuple Production if you want to know more about the creation of the ntuples, but this is not necessary for the exercise.

Preparation: Setting up the Code

This exercise is based on standalone ROOT scripts. It has been tested to work within the ROOT environment of CMSSW_5_3_5 and CMSSW_5_3_9 but should also work outside that with any not-too-old ROOT version.

Setup at SINP

To access the ntuples, it is necessary that you have access to the `/storage/cmsdas/exercise/long/Susy` directory on the school server. In order that we all have the same working environment

1. login the home-directory of your school account following the computing instructions
2. setup CMSSW_5_3_9 following the computing instructions
3. go to the working directory: `cd ${CMSSW_BASE}/src` and configure the environment: `cmsenv`
4. download the exercise code
5. unpack the archive: `tar -xvf RA2Exercise-v1.tar`

Others

WARNING: WILL NOT WORK YET, NTUPLES NEED TO BE COPIED TO ACCESSIBLE LOCATION.

The complete exercise code, including the solutions and the ntuple maker, is also available as a git repository from github.

First, clone a local copy of the repository into your local working area,

```
git clone git@github.com:mschrode/DAS.git
```

Then, change to the exercise branch,

```
git checkout 2013-SINP-exercise
```

Now you have all the scripts available for the exercise. As mentioned above, they are sometimes incomplete, the completions will be implemented as part of the exercise. If you want to have a complete working versions including the solutions, you can switch to the branch

```
git checkout 2013-SINP
```

Sample Definition and Composition

Introduction

In this section of our long exercise, we will learn how to access the data and simulated events and how to apply the selection cuts. We will investigate the data and compare it to the expected properties of the SM backgrounds. You will also get a feeling of how possible SUSY signals look like.

In this analysis, candidate events are required to pass the **baseline selection** defined as follows:

1. no well-reconstructed and isolated leptons (electrons or muons) with p_T > 10 GeV and | | < 2.4 are present (*lepton veto*);
2. N_{jets} >= 3, where only jets with p_T > 50 GeV and | | < 2.5 are considered;
3. HT > 500 GeV, where again only jets with p_T > 50 GeV and | | < 2.5 are considered. These jets are also referred to as *HT jets*, which is defined as the scalar sum of all selected jet pt;
4. MHT > 200 GeV, where jets with p_T > 30 GeV and | | < 5 are considered. These jets are referred to as *MHT jets*, which is defined as the negatif vectorial sum of all selected jet pt;
5. Δ (jet$_{1,2}$,MHT) > 0.5 and Δ (jet$_3$,MHT) > 0.3.

Lateron, the selected events are distributed in 36 exclusive search bins that are defined on top of this baseline selection by even tighter N(jets), HT, and MHT selection criteria. But for now, we will stick with the baseline selection.

Consider the topology of the selected events.

- **Question 1.1.1:** Can you list possible SM processes that result in this final state? Given these processes, can you motivate the baseline selection? Which of the background processes do you expect to dominate at large N_{jets}, at large HT, and which at large MHT? Don't do any calculation now, just use your intuition.

In the following, you will be asked to perform several exercises that will help us better understand the properties of the processes we are investigating and to check whether your answers to Question 1.1.1 are correct. It will also give us some introduction to the technical aspects of how to use the ntuples and perform an analysis with ROOT.

Before you start, make sure you have executed the initial commands shown above at Preparation: Setting up the Code. Then, go to the `General` directory, i.e. from your working area `$CMSSW_BASE/src/RA2Exercises` do

```
cd General
```

Kinematic Properties of the SM-Background and Signal Processes

We will now investigate the SM background processes using simulated events and compare them to potential new-physics signal processes. There are ntuples available for several different SM as well as potential SUSY-signal processes. They are listed in the table below together with their cross-sections in units of picobarns (pb) at 8 TeV and with the total number of simulated events. (Beware of the kinematic preselection applied to some of the MC samples.)

Id	Process	Cross-section [pb]	Total Nr. Events	Comment
11	W(l)+jets	30.08	6619654	for HT > 400 GeV
12	ttbar+jets	127.06	37543831	only semi- and full-leptonic decays; total xs = 234 pb
13	Z( )+jets	6.26	1006928	HT>400 GeV
14	QCD	8630.0	44443155	HT>500 GeV
21	LM6	0.502	1000000	low mass CMSSM m_0 = 85 GeV, $m_{1/2}$ = 400 GeV, tan = 10
22	LM9	9.287	1000000	

To analyse the simulated events, execute the script `general1.C` with ROOT by typing

```
root -l -b -q general1.C+\(id\)
```

where `id` should be replaced with the values shown in the first column of the table above. We will start with analysing the W(l)+jets sample, so set `id` to 11. (Notice the + after the name of the script in the above command. This will tell ROOT to compile the script before execution which leads to a faster programme.)

The script `general1.C` will produce a ROOT file `General_WJets.root` that contains various control distributions. Browse the file and investigate its content. You can conveniently plot the distributions with the script `plotSample.C`:

```
root -l plotSample.C\(id\)
```

where `id` is again set to 11. The script also stores the plots as `png` files in the current directory.

Discuss the distributions. The events are selected if no isolated lepton (electron or muon) has been reconstructed and if there are at least 3 jets with $p_T > 50$ GeV and | | < 2.5, i.e. if the baseline-selection steps 1 and 2 in the list above are fulfilled. Do the shapes of the distributions meet your expectations?

- **Question 1.2.1**: What is the reason for the lower cut-off in the N(jets) and HT distributions? There is also a small dip at very low MHT; what is the reason for this?

To exit the ROOT environment, you will have to type at the command prompt `.q` and hit `Enter`. (In case of the first script, the `-q` switch told ROOT to exit after execution of the script. However, since we wanted to look at the plots, we needed to keep the environment open.)

Open now the two scripts we have just run, i.e. `general1.C` and `plotSample1.C`, in your favourite editor, e.g. `emacs`. Familiarise yourself with the code. Make sure you understand what is being done because the following exercises will build up on this!

- How is the event content accessed? How is the content read from file?
- Where are the histograms declared and filled? Where are they drawn?
- How are the event selection steps 1 and 2 performed?

We still want to understand the motivation for the baseline selection cuts. Therefore, we will compare the relevant kinematic distributions for the different processes. As you have noticed, so far only N(jets), HT, and MHT are calculated in `general1.C`. However, the baseline selection also includes the Δ cuts (step 5), and thus, we also want to investigate the Δ distributions. Please add the following code to compute the Δ to `general1.C` below the line saying

```
//>>> PLACE DELTA PHI COMPUTATION HERE
```

Show code Hide code

```
    // Delta phi between the MHT vector and the jet for the leading MHT jets
    std::vector<float> deltaPhis(3,9999.);
    const float phiMht = std::atan2(selMhtY,selMhtX);
    // Loop over reco jets: remember, they are ordered in pt!
    unsigned int nMhtJets = 0;
    for(int jetIdx = 0; jetIdx <evt->jetsN(); ++jetIdx) {

      // Select MHT jets
      if( evt->jetsPt()[jetIdx] > 30 && std::abs(evt->jetsEta()[jetIdx])  <5.0 ) {

        // Compute delta phi (per convention in sector between -Pi and Pi)
        // between this jet and the MHT vector
        const float deltaPhi = TVector2::Phi_mpi_pi(evt->jetsPhi()[jetIdx]-phiMht);
        // Store deltaPhi
        deltaPhis.at(nMhtJets) = std::abs(deltaPhi);

        // Increase counter for MHT jets
        ++nMhtJets;
        // DeltaPhi cut only for first three jets
        // Leave jet loop if the first 3 MHT jets tested
        if( nMhtJets == 3 ) break;

      } // End of MHT-jet criterion
```

```
    } // End of loop over reco jets
```

Note the usage of `TVector2::Phi_mpi_pi(Double_t x)`, described in http://root.cern.ch/root/html532/TVector2.html#TVector2:Phi_mpi_pi, to compute Δ . This is a very useful function that puts the result automatically into the correct sector.
The Δ histograms are already implemented, but they are not filled yet. Please add the relevant code.

Now we have the full set of relevant kinematic distributions at hand, and we can compare them for different samples. At the moment, `general1.C` runs over a small number of simulated events. To analyse the full samples, you will need to replace the current input file name

```
Sample::fileNameSubSample(id)
```

by

```
Sample::fileNameFullSample(id)
```

Then, execute

```
root -l -b -q general1.C+\(id\)
```

again and again and again for the background samples 11 to 14 and the signal sample 21 to 22 in the above table. You can automate this using simple for-loops

```
for i in {11..14}; do root -l -b -q general1.C+\(${i}\); done
for i in {21..22}; do root -l -b -q general1.C+\(${i}\); done
```

We will now compare the shapes of the different kinematic distributions for the different processes. This can be done with the prepared script `plotSampleComparison.C` by executing

```
root -l plotSampleComparison.C
```

- **Question 1.2.2**: How do the processes differ?
    - Explain the differences of the N(jets) distributions of QCD and ttbar events.
    - Explain the different HT and MHT distributions of the QCD and the Z( )+jets events.
    - Explain the behaviour of the Δ distributions.

Now, modify the script `plotSampleComparison.C` to also plot the signal samples.

- **Question 1.2.3**: What is the motivation for the different baseline-selection cuts?

Event Yields Expected From the Simulation

We will now use the simulated events to estimate the number of SM-background events after the full baseline selection. In the script `general2.C`, the corresponding cuts, i.e. the selection steps 1 to 5 in the list above, are already implemented using the auxiliary class `Selection` in `../Utils/Selection.h`. Also note that we do not have to compute the selection variables N(jets), HT, MHT, and Δ ourselves, they are in fact already present in the event content! Execute

```
root -l -b -q general2.C+\(id\)
```

where again we start with the W+jets sample, i.e. with `id` is 11. It will produce the output file `General_WJets-Yields.root` that contains the known kinematic distributions as well as the histogram `hYields` that stores the number of events passing the selection. The first bin contains the number of events after the baseline selection, the further bins the number of events after the different search-bin selections.

- **Question 1.3.1**: How many W+jet events are expected in 19.5/fb of data?

Let us determine the expected event yields also for the other background processes. We do not have to compute the cross-section normalisation ourselves every time. Instead, we can use the weight already stored in the ntuples. It is returned by the `Event::weight()` function and includes the cross-section normalisation as well as a correction to properly describe the impact from pile-up collisions.

- **Question 1.3.2**: Why is the pile-up reweighting required?

Adapt the `general2.C` script such that the histogram entries are weighted by the event weights. Then, run the script for the SM-background samples 11 to 14. Afterwards, you can use the script `plotYields.C` to conveniently print the yields:

```
root -l plotYields.C
```

Discuss the result. In what phase-space regions do the different backgrounds dominate? Does this match your initial expectation (Question 1.1.1)?

Data

We will now study what these distributions look like for real *pp* collision data collected by CMS in 2012 at √(s) = 8 TeV. Remember, we can only do this since about 18 months! Before, no one has ever seen collisions at such high energies. **So be amazed :)**

Execute the script `general2.C`. This also takes 1 as an argument to run over data:

```
root -l -b -q general2.C+\(1\)
```

You can look at the data distributions with the script plotSample.C:

```
root -l plotSample.C\(1,true\)
```

(The second argument, `true`, tells the script to use the output of `general2.C` instead of `general1.C`.)

Also run `general1.C`, i.e. fill the histograms without the HT, MHT, N(jets), and Δ   cuts. Then, plot only the data distributions with

```
root -l plotSample.C\(1\)
```

- **Question 1.4.1**: How do you explain the low-HT and low-MHT behaviour of the data (in case of no HT, MHT, and N(jets) cuts)?

Data vs Simulated SM Background

As last part of this exercise, we want to compare the N(jets), HT, and MHT distributions in data with the sum of the background distributions obtained from simulation. For this, you can use the script `plotDataVsMC.C` that adds (*stacks*) the simulated background distributions and superimposes the data distribution.

- **Question 1.5.1:** Do you observe any deviations of the data from the SM background expectation? What can you say about the existence of any new-physics processes?
- **Question 1.5.2:** What uncertainty is represented by the error bars? Are there any further uncertainties to be considered?

In the Jets+MHT analysis, the background yields are not taken from the simulation but, as explained in the introductory presentation, obtained using data-based methods. In the following exercise, we will discuss two of these background-prediction methods in detail.

- **Question 1.5.1:** Why do we not use the simulated background yields but measure them from data?

W+Jets and ttbar Background Determination

Introduction

An important SM background, which dominates in most search regions, arises from the production of W bosons in association with jets (*W+jets background*) and from the production of top-antitop quark pairs (*ttbar background*).

Since top quarks decay almost exclusively to W bosons and bottom quarks, both processes lead to the presence W bosons and jets in the event. The W bosons decay either into a quark-antiquark pair (*hadronically*) or into a lepton-neutrino pair (*leptonically*). In the first case, there is only very little MHT in the event, produced only from jet mismeasurements. Hence, the events are efficiently rejected by the selection criteria because MHT > 200 GeV is required. In the latter case of leptonically decaying W bosons, the events are to first approximation also rejected because events with isolated leptons are rejected in the baseline selection.

However, there are two important cases in which the lepton veto fails, and hence, W+jets and ttbar events enter the search region:

- *Lost Lepton*: The leptons from the W decay are not reconstructed due to either
    - the **limited geometrical acceptance** of the detector; or
    - the **inefficiency of the reconstruction** algorithm; or
    - the leptons are **not isolated** because they geometrically overlap with a jet.
- *Hadronic Tau*: The W boson decays into a tau lepton that decays to hadrons that form a jet. In this case, there will be no isolated leptons in the event such that the lepton veto is passed.

In both cases, there can be sufficient MHT to pass the MHT > 200 GeV selection criterion caused by the neutrinos from the W (and subsequent tau) decays.

- **Question: 2.1.1** How would you define the lepton isolation? Why do we actually only consider isolated leptons when applying the lepton veto?

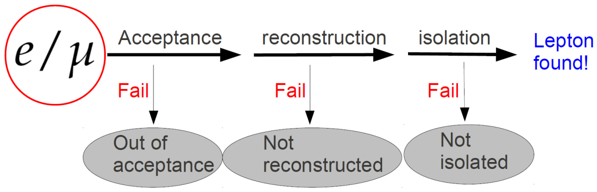
In this analysis, the W+jets and ttbar backgrounds are determined via two methods which address separately the two stated cases. The two methods are discussed in the following because

- they are good examples for data-based background prediction methods on which many important analyses at the LHC are based, in particular searches for new phyiscs; and
- the background processes are important for the Jets+MHT search. At large N(jets), the SM background stems in fact almost entirely from ttbar production.

Remember: Both the lost-lepton and the hadronic-tau method are used to predict the background expected from W+jets and ttbar events, but each method predicts different parts of the backgrounds that occur due to different effects. The total number of W+jets and ttbar background events is obtained by adding the predictions from the lost-lepton and the hadronic-tau methods!

Lost-Lepton Method

This method aims to determine the number of events in which a W boson decays into a lepton that escapes identification, and hence, does not trigger the lepton veto, because it is either out-of-acceptance, not well-reconstructed, or not isolated.



The procedure starts from events with exactly one well-reconstructed, isolated muon. For each event, the search variables HT, MHT, and N(jets) are computed and the baseline selection steps **except for the lepton veto** are applied. The sample of events surviving this selection forms our *control sample*; these are events that almost entirely originate in W+jets and ttbar production. Then, the expected number of background events, i.e. the number of events surviving the full baseline selection including the lepton veto, is estimated from the control sample by essentially weighting each event by the probability that the lepton survives the lepton veto. In this way, we obtain the full kinematic properties of the background events directly from data and do not need to worry about its simulation! The only thing we need to correctly work out is the lost-lepton probability. This probability depends on the geometrical and kinematic acceptance as well as the reconstruction and isolation efficiencies $_{RECO}$ and $_{ISO}$, respectively, of the lepton.

- **Question 2.2.1:** For the muon channel, what would the weight factor w (, $_{RECO}$, $_{ISO}$) look like? In order to answer this question, think about how exactly you would define , $_{RECO}$, and $_{ISO}$.
- **Question 2.2.2:** Could the same muon control sample be used to estimate the contributions from the electron channel? What would be the weight factor in this case?
- **Question 2.2.3:** In this analysis, the acceptance and efficiencies are determined from a MC simulation. (Essentially, we replace the uncertainty one would get due to using MC simulation to determine the background by the uncertainty on the simulation of the acceptance and efficiencies.) Can you think of a way to verify the numbers with data?

Lost-Lepton 1: Acceptance and Efficiency Determination

We will now learn how , $_{RECO}$, and $_{ISO}$ can be determined from simulated events. Here, we will do this only for muons using W+jets events; in the analysis, the numbers are determined also for electrons, of course, but it works analogous to the muon case and one does not learn anything new from it. Before starting the implementation, answer the following questions.

- **Question 2.2.1.1:** How would you technically determine , $_{RECO}$, and $_{ISO}$? Is it sufficient to determine one number each?
- **Question 2.2.1.2:** Would you apply any event selection?
- **Question 2.2.1.3:** Is it sufficient to consider the W+jets sample?

Please go the `LostLepton` directory in your `$CMSSW_BASE/src/RA2Exercises` working area. Have a look at the script `lostLepton1.C`. It is an example how to determine the muon acceptance and the muon-reconstruction efficiency $_{RECO}$ from the W+jets sample (`id 11`). Investigate how this is implemented.

- What event variables are being used?
- How are the acceptance and efficiency computed and stored?

Now, run the script by executing

```
root -l -b -q lostLepton1.C+
```

By default, this runs over the W+jet sample. The produced acceptance and efficiency maps are stored in `LostLepton_MuonEfficienciesFromWJetMC.root` in `$CMSSW_BASE/src/RA2Exercises/data`. Investigate the result; you can conveniently execute the plotting script

```
root -l plotMuonEfficiencies.C
```

for this.

So far, we have only determined the acceptance and the reconstruction efficiency. Extend the `lostLepton1.C` script to also determine the isolation efficiency. (The required histograms are already present, you just need to fill them.)

Below is a proposed solution how to compute the isolation efficiency. However, before looking at it, try for yourself first!

Show solution Hide solution
After the statement

```
// Reconstruction-efficiency determination 2: Counter for those events
// with generator-level muons inside acceptance where the muon has also
// been reconstructed.
hRecoPass.at(nJetIdx)->Fill(evt->ht(),evt->mht());
```

place the following code

```
// Isolation-efficiency determination 1: Counter for all events with a
// reconstructed muon that has a generator-level muon match inside the
// the acceptance, regardless of whether the reconstructed muon is also
// isolated or not.
hIsoAll.at(nJetIdx)->Fill(evt->ht(),evt->mht());

// Check if the muon is also isolated: check if an isolated muon is present
// in the event that matches the reconstructed muon in R
int matchedIsoMuonIdx = -1;
if( utils::findMatchedObject(matchedIsoMuonIdx,evt->muonsEta()[matchedMuonIdx],evt->muo
  // Muon is isolated

  // Isolation-efficiency determination 2: Counter for those events where
  // the muon is also isolated.
  hIsoPass.at(nJetIdx)->Fill(evt->ht(),evt->mht());

} // End of muon is isolated
```

Lost-Lepton 2: Validation of the Method

We will now validate the determined acceptance and efficiencies as well as the lost-lepton method itself. We will do this by verifying that we can correctly predict the number of lost-lepton background events in the W+jets MC sample. We use the simulated events because here we have the generator-truth information at hand, and thus, can compare the prediction to the truth. This kind of test is often referred to as *closure test*.

At first, we will only validate the determined muon acceptance.

- **Question 2.2.2.1:** How could we do this?

Have a look at `lostLepton2.C`. This is an example of how the closure test of the acceptance can be performed. Make sure you understand how the prediction is obtained! Now execute the script

```
root -l -b -q lostLepton2.C+
```

which again by default runs over the W+jet sample. Its output, the predicted and true HT, MHT, and N(jets) distributions for out-of-acceptance muons, are written to the file `LostLepton_ClosureMuonAcceptance.root`. They can be conveniently plotted and compared using the script `plotClosureMuons.C`,

```
root -l plotClosureMuons.C\(\"LostLepton_ClosureMuonAcceptance.root\"\)
```

As you notice, also the predicted and the true event yields after the baseline selection and in the individual 36 search bins are printed.

- **Question 2.2.2.2:** Are the predicted yields compatible with the truth? What are the printed uncertainties? Are the uncertainties sufficient?
- **Question 2.2.2.3:** How could you determine the uncertainties in case of zero predicted events?

Now, we will perform a full closure test of the lost-lepton method. Hence, we will perform the prediction as we would do it on real data and compare to the expectation from the MC truth. Have a look at `lostLepton3.C`. Here, a full data-based prediction from the muon control sample for the number of lost leptons is implemented. Note how the prediction does not rely on any MC-truth based quantities! The essential part is the computation of the lost-lepton weight

```
const double llw = lostLeptonProb(muAcc,muEffReco,muEffIso) * controlSampleCorr(muAcc,muEffReco,m
```

which is the product of the probability for the muon to be out-of acceptance, not reconstructed, or not isolated and a correction to account for the fact that the control sample itself contains only events with well-reconstructed, isolated muons inside the acceptance, as discussed above. So far, the two functions `lostLeptonProb` and `controlSampleCorrection` are dummies, however. Implement them as discussed above!

Below is a proposed solution... and as usual, before looking at it, try for yourself first!

Show solution Hide solution

```
double lostLeptonProb(double acc, double reco, double iso) {
  return (1.-acc) + acc*(1.-reco) + acc*reco*(1.-iso);
}

double controlSampleCorr(double acc, double reco, double iso) {
  return 1. / ( acc * reco * iso );
}
```

Now, we can run the script, which will produce the output `LostLepton_ClosureMuon.root`. Again, prediction and truth can be compared using the script `plotClosureMuons.C`,

```
root -l plotClosureMuons.C\(\"LostLepton_ClosureMuon.root\"\)
```

The script also plots the kinematic properties of the muon in the control sample.

Lost-Lepton 3: Background Prediction from Data

Finally, we will perform the full lost-lepton prediction on data! However, we should not use our efficiency maps because they were determined only for lost muons and only from W+jets events while the full lost-lepton background comprises also ttbar events and, of course, lost electrons. Therefore, we have to determine the acceptance and efficiency maps also for electrons and from a realistic a mixture of W+jets and ttbar events, as discussed above. We will not do this here, because we would not learn anything new, but instead use the efficiency maps from the analysis. They are provided in `RA2Exercises/data/LLEff-SUS-13-012.root`. Use these maps to perform the lost-lepton prediction on the data sample (`id` 1). You can use the `lostLepton4.C` script, which is an implementation of the lost-lepton method for data.

- **Question 2.2.3.1:** In `lostLepton4.C`, what has been added with respect to the prediction of only lost muons?

The result of the prediction can be plotted via `plotPrediction.C`, which also prints the event yields in the various search bins. Compare our prediction to the paper results. Do they agree?

Let us consider some more important aspects about the lost-lepton method.

- **Question 2.2.3.2:** Recap how we select the control sample. Do we need to apply any further corrections to the obtained yields?
- **Question 2.2.3.3:** Could potential new-physics events in data bias our prediction? How? What could be done to mitigate such an effect?

Finally, let us think about the systematic uncertainties we would assign to our prediction. In fact, without proper uncertainties the result is just meaningless!

- **Question 2.2.3.4:** What effects do you think impact the prediction and should be considered as a source of systematic uncertainties? How would you determine the size of the uncertainties?

Hadronic-Tau Method

This method aims to determine the number of events in which a W boson decays into a tau lepton that decays further into hadrons. In almost all cases, these are either one or three charged pions accompanied by a number of neutral hadrons that decay to photons. They are typically clustered into a jet, referred to as *tau jet*, during the jet-finding process of the event reconstruction. The energy deposited by the hadrons, i.e. the energy of the tau jet, is referred to as *visible tau-energy*. In our SUSY search, we cannot distinguish such tau jets from QCD-like jets that originated e.g. in the hard scattering process. Hence, the tau jets enter the HT, MHT, and N(jets) calculation, and we need to determine the expected number of W+jets and ttbar background events with tau jets.

- **Question 2.3.1:** The visible tau-energy is in general smaller than the energy of the tau lepton. Why?

The procedure starts with a control sample from data that contains events with exactly one isolated muon, as in the lost-lepton case; these are events that almost entirely originate in W+jets and ttbar production. The p_T spectrum of the muon from W decays is expected to be similar to that of the tau lepton from W decays. This feature is used to simulate the tau jet by replacing the muon p_T with the visible p_T expected for a tau jet that originates in a tau lepton with the same p_T as the muon. Essentially, the p_T of the muon is scaled to reproduce the expected, visible tau-jet p_T. The muon with the scaled p_T (the *simulated tau jet*) together with the other jets in the event are used to re-compute the event variables relevant for the analysis, i.e. HT, MHT, and N(jets). Then, the baseline selection steps are applied and the number of surviving events is our estimate for the hadronic-tau background. In this way, we obtain the full kinematic properties of the background events directly from data and do not need to worry about its simulation! The only thing we need to correctly work out is the visible tau-energy expected for a given muon p_T.

- **Question 2.3.2:** Why are the muon- and the tau-p_T spectra expected to be similar?
- **Question 2.3.3**: Why do we use isolated muons?

In the following, you will be asked to perform several exercises that will illustrate this method. We will start with developing and testing the method using simulated events. Here, we will work with the W+jets sample only; in real life, both W+jets and ttbar events are used in a mixture corresponding to the expected mixture in data. We will use the same helper classes in `Utils/` as in case of the lost lepton method to access the data and perform some event selection steps. Now, go to the `RA2Exercises/HadTau` directory in your working area.

Hadronic-Tau 1: Tau Response Templates

The key part of the method is to assume that the muon in the control sample events is a tau lepton and to replace the measured muon p_T by the expected tau-jet p_T. Hence, we need to know the tau-jet p_T response, i.e. the probability to measure a certain tau-jet p_T given a certain true tau-lepton p_T. Therefore, as a first exercise, we will determine the tau response templates from simulated W+jets events. (Of course, we need to use simulated events because we need to know the true tau-lepton p_T!)

Using generator-truth information, we will select only W+jets events where the W decayed into hadronically-decaying taus. Then, we will identify the reconstructed jet that originates in the tau-decay products. The identification is done via a geometric matching: we look for the jet that is closest in ΔR = √(Δ ^2 + Δ ^2) to the generator-level tau. In addition, we select only events with at leat 2 "HT jets" (jets with $p_T > 50$ GeV and | | < 2.5), where the matched tau jet is not considered for the jet counting. Also, we require the generator-level tau to fall into the kinematic acceptance of our well-reconstructed muons. This corresponds to the selection applied lateron to select the muon control-sample used in the prediction, and we want to determine the response templates from events resembling the control sample as close as possible. The tau response template is then simply obtained by plotting the ratio of the reconstructed tau-jet p_T and the generated tau-lepton p_T. We will do this in different bins of p_T of the generated tau.

To produce the tau templates, please execute the script `hadTau1.C`

```
root -l -b -q hadTau1.C+
```

This will run over the W+jets sample (`id` 11), fill the response templates, and store them in `RA2Exercises/data/HadTau_TauResponseTemplates.root`. Analyse the code and make sure you understand what is being done.

- Identify the event selection steps described above.
- How is the jet-tau matching being done?

To plot the templates, execute the script `plotTauResponseTemplates.C`:

```
root -l plotTauResponseTemplates.C
```

Investigate the response templates.

- **Question 2.3.1.1**: Why are the mean values in general smaller than 1?
- **Question 2.3.1.2**: What is the reason for a response greater than 1?
- **Question 2.3.1.3**: Can you explain the shift of the mean of the templates as the p_T of the tau lepton increases?

Hadronic-Tau 2: Validation of the Method

We will now perform a closure test with the events of the W+jets MC sample to validate the hadronic-tau method. We will do this by verifying that we can correctly predict the number and the HT, MHT, and N(jets) distributions of background events due to hadronically decaying taus, where we will analyse the events **as if they were real collider data**, i.e. we will only use measurable quantities. We still use the simulated events because here we have the generator-truth information at hand, and thus, can compare the prediction to the truth.

As described in the introductory section, we will select the muon to simulate the tau jet. For technical reasons, we then have to identify the jet corresponding to the muon and remove it from the jet collection (remember, each muon is also contained in the jet collections!). Now, we select only events with at leat 2 "HT jets" (jets with p_T > 50 GeV and | | < 2.5), where the matched muon jet is not considered for the jet counting. Then, we treat the muon as a tau lepton and scale (*smear*) its p_T by a random factor drawn from the tau-response templates. Remember that several templates were determined in bins of tau-lepton p_T. Hence, depending on the muon p_T, we have to chose the corresponding template. The scaled muon p_T simulates the measured tau-jet p_T. This, together with the other jets in the event, is used to recompute HT, MHT, and N(jets). Based on these recomputed quantities, the analysis event selection is finally applied to obtain the background prediction.

- **Question 2.3.2.1**: Can you motivate the control-sample event selection? Why do we initially require at least 2 HT jets for the control sample (in contrast to the "at least 3 jets" requirement for the final event selection)?

The closure test can be performed with the script `hadTau2.C`. In the second part of the event loop, the expectation from simulation for the background events from hadronically decaying taus, i.e. the 'truth', is obtained. This is straight-forward: using generator-truth information,events are selected where the W decayed into a tau that itself decayed to hadrons. Then, the baseline selection criteria are applied, and the HT, MHT, and N(jets) histograms are filled.

We will focus on the first part of the event loop. Here, the data-based prediction is performed. The code is not complete yet, and we need to add a few essential things to perform the prediction. But first, investigate the code and make sure you understand the following:

- What selection requirements are applied to the muon?
- Why is the muon removed from the jet collection? How is it done, and how are the search variables HT, MHT, and N(jets) recomputed without the muon?
- How are the kinematic properties of the tau jet simulated from the muon?

Now, we want to use the simulated tau jet to recompute HT, MHT, and N(jets). Add the relevant implementation where it says `>>> PLACE HT, MHT, AND NJETS COMPUTATION HERE`. As always, you find a proposed solution below.

Show solution Hide solution

```
// If simulted tau-jet meets same criteria as as HT jets,
// recompute NJets and HT
if( simTauJetPt > Selection::htJetPtMin() && std::abs(muEta) <Selection::htJetEtaMax() ) {
  simNJet++;
  simHt += simTauJetPt;
}
// If simulated tau-jet meets same criteria as MHT jets,
// recompute MHT
if( simTauJetPt > Selection::mhtJetPtMin() && std::abs(muEta)  <Selection::mhtJetEtaMax() )
  simMhtX -= simTauJetPt*cos(muPhi);
  simMhtY -= simTauJetPt*sin(muPhi);
}
```

Now, run the closure test by executing

```
root -l -b -q hadTau2.C+
```

This will by default run over the W+jets sample (`id` 11) and produce a ROOT file `HadTau_WJetMC_Closure.root`, which contains the predicted and true HT, MHT, and N(jets) spectra. You can conveniently compare the predicted with the true distributions by executing script

```
root -l plotClosureTest.C
```

Compare the predicted with the true distributions. As you observe, they do not agree very well. Answer the following questions to find the reason for this.

- **Question 2.3.2.2:** Do you expect the number of muons from the W decays to be equal to the number of hadronically decaying taus?
- **Question 2.3.2.3:** Are there any other sources than the W decay for isolated muons in the events?
- **Question 2.3.2.4:** How are the control sample events selected?

Apparently, some corrections to the prediction are required! Please add them to the script. You can modify the existing variable `corr`, which is applied as a weight when filling the histograms (right now, corr is set to 1). Below, you find a proposed solution

Show solution Hide solution

```
// Corrections to control sample
 const double corrBRWToTauHad = 0.65;  // Correction for the BR of hadronic tau decays
 const double corrBRTauToMu = 1./1.15; // Correction for the fact that some muons could come
 const double corrMuAcc = 1./muonAcc(evt->mht(),evt->nJets()); // Correction for muon accept
 const double corrMuRecoEff = 1./muonRecoEff(evt->ht(),evt->mht(),evt->nJets()); // Correcti
 const double corrMuIsoEff = 1./muonIsoEff(evt->ht(),evt->mht(),evt->nJets()); // Correction

 // The overall correction factor
 const double corr = corrBRTauToMu * corrBRWToTauHad * corrMuAcc * corrMuRecoEff * corrMuIso
```

Run `plotClosureTest.C` again and convince yourself that we get better agreement now. Are you satisfied with the performance?

- **Question 2.3.2.5:** Do the plotted error bars represent the total statistical uncertainties?

Hadronic-Tau 3: Background Prediction from Data

Finally, we will perform the full hadronic-tau background prediction on data! However, we will not use the data sample used so far (`id` 1) because they have been recorded using trigger on HT and MHT with quite high thresholds (recall our discussion during the General exercise). Instead, we will use data (`id` 2) collected by a different trigger path that requires less HT and MHT but an isolated muon instead; the amount of collected data is the same, i.e. the full 19.5 / fb of 2012.

- **Question 2.3.3.1:** Why do we not use the data collected with the HTMHT triggers as before?
- **Question 2.3.3.2:** Why can we lower the HT and MHT thresholds when requiring in addition an isolated muon?

Remember that we had to apply a correction for the muon acceptance as well as reconstruction and isolation efficiencies. However, the efficiency maps used so far were determined only from W+jets events while the full lost-lepton background comprises also ttbar events. Therefore, we have to determine the acceptance and efficiency maps from a realistic a mixture of W+jets and ttbar events. We will not do this here, because we would not learn anything new, but instead use the efficiency maps from the analysis. They are provided in `RA2Exercises/data/LLEff-SUS-13-012.root`.

We can now use the script `hadTau2.C` to perform the hadronic-tau background prediction on data:

```
root -l -b -q hadTau2.C+\(2,\"../data/LLEff-SUS-13-012.root\"\)
```

(The two explicit arguments specify to run over data and to use the different efficiency maps.) The result of the prediction can be plotted via `plotPrediction.C`, which also prints the event yields in the various search bins. Compare our prediction to the paper results. Do they agree?

Let us consider some more important aspects about the hadronic-tau method.

- **Question 2.3.3.3:** Could potential new-phyiscs events in data bias our prediction? How? What could be done to mitigate such an effect?

- **Question 2.3.3.4:** The number of events in the control sample limits the statistical precision of the prediction. Could this be improved?

Finally, let us think about the systematic uncertainties we would assign to our prediction. In fact, without proper uncertainties the result is just meaningless!

- **Question 2.3.3.5:** What effects do you think impact the prediction and should be considered as a source of systematic uncertainties? How would you determine the size of the uncertainties?

Results and Interpretation

Finally, let us combine our data-based predictions of the lost-lepton and the hadronic-tau background to estimate the W+jets and ttbar background. We can use our data-based result and replace the corresponding simulated predictions in the data-vs-background plot we produced before.

Please follow the recipe below:

- Got to the `RA2Exercises/Result` directory.
- Copy the ROOT file with the HT, MHT, and NJets distributions observed in data, i.e. `General_Data-Yields.root`, in this directory.
- Copy the background predictions from simulation, i.e. `General_*-Yields.root`, in this directory.
- Now, we don't want to use the W+jets and ttbar prediction from simulation. Therefore, delete them! Instead, copy our data-based lost-lepton and hadronic-tau predictions, i.e. `LostLepton_Data_Prediction.root`, `HadTau_Data_Prediction.root` in this directory.
- Finally, also copy our signal expectiations `General_LM*-Yields.root`.

Now you can execute

```
root -l plotDataVsBkg.C
```

This will compare the HT, MHT, and N(jets) spectra observed in data to the expected SM background contributions. As you can see, instead of the individual W+jet and ttbar processes, the contributions from the lost-lepton and the hadronic-tau background are shown - our results measured from data!

- **Question 4.1:** Do you observe any sign of new physics? Is this actually a valid comparison of data and expected backgrounds?

For comparison, also the signal expectation are superimposed.

- **Question 4.2:** What uncertainties have to be considered for the shown signal expectations? Are these uncertainties also relevant for the backgrounds (assuming they are determined from data)?

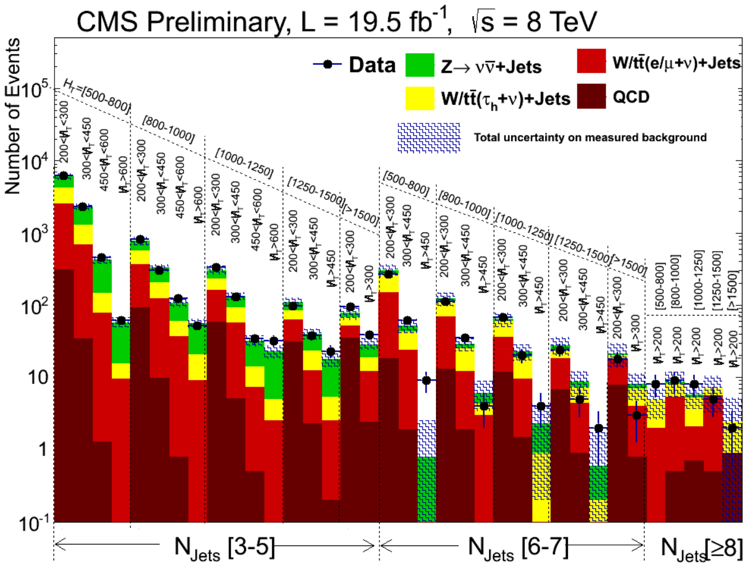
In this long exercise, the ttbar and the W+jets backgrounds have been determined from data.

- **Question 4.3:** Do you have an idea how the remaining backgrounds from Z(   )+jets and QCD-multijet production could be determined from data?

The student's presentation of their results can be found here.

Results of CMS-PAS-SUS-13-012

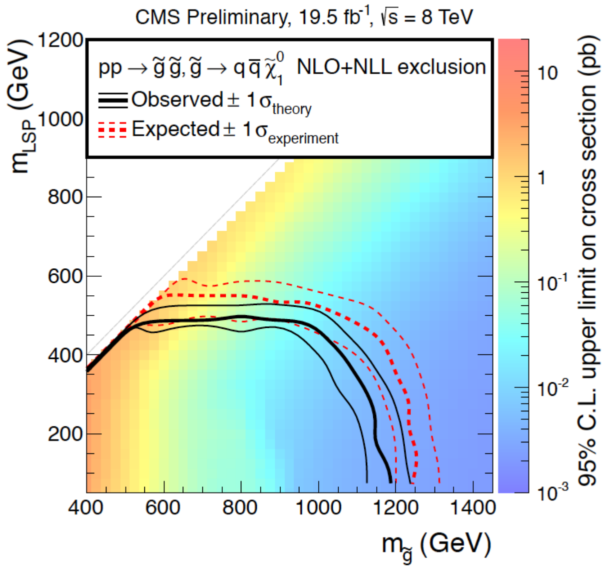
Below, you find the event yields observed in 19.5/fb of data at 8 TeV compared to the fully data-based

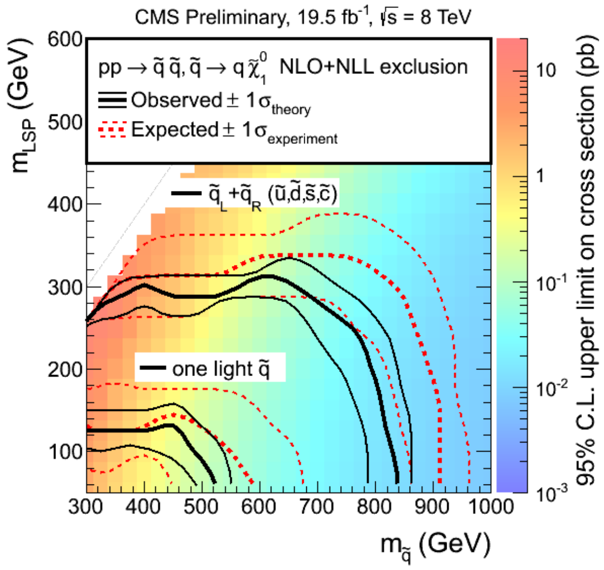


background prediction.

Selection N_{jets}	H_T	$\not{H}_T$	$Z \to \nu\bar{\nu}$ from γ+jets	$t\bar{t}$/W $\to$ e, μ+X	$t\bar{t}$/W $\to \tau_h$+X	QCD	Total background	Obs. data
3-5	500-800	200-300	1821.3±326.5	2210.7±447.8	1683.7±171.4	307.4±219.4	6023.1±620.2	6159
3-5	500-800	300-450	993.6±177.9	660.1±133.3	591.9± 62.5	34.5± 23.8	2280.0±232.1	2305
3-5	500-800	450-600	273.2± 51.1	77.3± 17.9	67.6± 9.5	1.3± 1.5	419.5± 55.0	454
3-5	500-800	> 600	42.0± 8.7	9.5± 4.0	6.0± 1.9	0.1± 0.3	57.6± 9.7	62
3-5	800-1000	200-300	215.8± 40.0	277.5± 62.4	191.6± 23.2	91.7± 65.5	776.7±101.6	808
3-5	800-1000	300-450	124.1± 23.7	112.8± 26.9	83.3± 11.2	9.9± 7.4	330.1± 38.3	305
3-5	800-1000	450-600	46.9± 9.8	36.1± 9.9	23.6± 3.9	0.8± 1.3	107.5± 14.5	124
3-5	800-1000	> 600	35.3± 7.5	9.0± 3.7	11.4± 3.2	0.1± 0.4	55.8± 9.0	52
3-5	1000-1250	200-300	76.3± 14.8	103.5± 25.9	66.8± 10.0	59.0± 24.7	305.6± 40.1	335
3-5	1000-1250	300-450	39.3± 8.2	52.4± 13.6	35.7± 6.2	5.1± 2.7	132.6± 17.3	129
3-5	1000-1250	450-600	18.1± 4.4	6.9± 3.2	6.6± 2.1	0.5± 0.7	32.1± 5.9	34
3-5	1000-1250	> 600	17.8± 4.3	2.4± 1.8	2.5± 1.0	0.1± 0.3	22.8± 4.7	32
3-5	1250-1500	200-300	25.3± 5.5	31.0± 9.5	22.2± 3.9	31.2± 13.1	109.7± 17.5	98
3-5	1250-1500	300-450	16.7± 4.0	10.1± 4.4	11.1± 3.6	2.3± 1.6	40.2± 7.1	38
3-5	1250-1500	> 450	12.3± 3.2	2.3± 1.7	2.8± 1.5	0.2± 0.5	17.6± 4.0	23
3-5	>1500	200-300	10.5± 2.8	16.7± 6.2	15.2± 3.4	35.1± 14.1	77.6± 16.1	94
3-5	>1500	> 300	10.9± 2.9	9.7± 4.3	6.5± 2.0	2.4± 2.0	29.6± 5.8	39
6-7	500-800	200-300	22.7± 6.1	132.5± 58.6	127.1± 21.5	18.2± 9.2	300.5± 63.4	266
6-7	500-800	300-450	9.9± 3.1	22.0± 10.8	18.6± 4.3	1.9± 1.7	52.3± 12.1	62
6-7	500-800	> 450	0.7± 0.6	0.0± 1.6	0.1± 0.3	0.0± 0.1	0.8± 1.7	9
6-7	800-1000	200-300	9.1± 2.8	55.8± 25.4	44.6± 8.2	13.1± 6.6	122.6± 27.7	111
6-7	800-1000	300-450	4.2± 1.6	10.4± 5.5	12.8± 3.1	1.9± 1.4	29.3± 6.6	35
6-7	800-1000	> 450	1.8± 1.0	2.9± 2.5	1.3± 0.5	0.1± 0.4	6.1± 2.7	4
6-7	1000-1250	200-300	4.4± 1.6	24.1± 12.0	24.0± 5.5	11.9± 6.0	64.4± 14.6	67
6-7	1000-1250	300-450	3.5± 1.4	8.0± 4.7	9.6± 2.5	1.5± 1.5	22.6± 5.7	20
6-7	1000-1250	> 450	1.4± 0.8	0.0± 1.8	0.8± 0.5	0.1± 0.3	2.3± 2.1	4
6-7	1250-1500	200-300	3.3± 1.3	11.5± 6.5	6.1± 2.5	6.8± 3.9	27.7± 8.1	24
6-7	1250-1500	300-450	1.4± 0.8	3.5± 2.6	2.9± 1.5	0.9± 1.3	8.8± 3.4	5
6-7	1250-1500	> 450	0.4± 0.4	0.0± 1.2	0.1± 0.2	0.1± 0.3	0.5± 1.3	2
6-7	>1500	200-300	1.3± 0.8	10.0± 6.9	2.3± 1.3	7.8± 4.0	21.5± 8.1	18
6-7	>1500	> 300	1.1± 0.7	3.2± 2.8	2.9± 1.2	0.8± 1.1	8.0± 3.3	3
≥8	500-800	> 200	0.0± 0.6	1.9± 1.5	2.8± 1.3	0.1± 0.4	4.8± 2.1	8
≥8	800-1000	> 200	0.6± 0.5	4.8± 2.9	2.7± 1.1	0.5± 0.9	8.7± 3.3	9
≥8	1000-1250	> 200	0.6± 0.5	1.4± 1.5	3.1± 1.2	0.7± 0.9	5.8± 2.2	8
≥8	1250-1500	> 200	0.0± 0.7	5.1± 3.5	1.3± 0.8	0.5± 0.9	6.9± 3.7	5
≥8	1500-	> 200	0.0± 0.6	0.0± 2.1	1.5± 1.0	0.9± 1.3	2.4± 2.8	2

Since no significant excess is found above the SM expectation, exclusion limits have been derived on the masses of possible new-physics particles. Below you see such limits for two so-called *Simplified Models* that are defined by their final-state topologies. Limits derived in Simplified Models can be translated into limits on realistic models such as supersymmetric models, provided all possible final states are covered by the Simplified Models.





References

- "Search for New Physics with Jets and Missing Transverse Momentum in *pp* collisions at √s=7 TeV", JHEP **1108** (2011) 155, arXiv:1106.4503
- "Search for supersymmetry in all-hadronic events with missing energy", CMS-PAS-SUS-11-004
- "Search for new physics in the multijet and missing transverse momentum final state in proton-proton collisions at √s=7 TeV", Phys. Rev. Lett. **109** (2012) 171803, arXiv:1207.1898
- "Search for New Physics in the Multijet and Missing Momentum Final State in Proton-Proton Collisions at 8 TeV", CMS-PAS-SUS-13-012 (2013)
- Public TWiki with result plots of the 8 TeV version
- Internal Analysis Note AN-12-350 of the 8 TeV version
- Internal TWiki documentation of the 8 TeV version
- ROOT reference guide: documentation of all ROOT classes
- C++ Language Library Reference: detailed descriptions of its elements and examples on how to use its functions

* Bradley Efron, "The Jackknife, the Bootstrap, and Other Resampling Plans", ISBN 0-898-71179-7

Ntuple Production

Below is a recipe how to produce the ntuples used in this exercise. This is not important for you in order to perform the exercise but intended as some additional information.

Show recipe Hide recipe

Code Setup

Setup a CMSSW_5_3_5 environment and go to the `src` directory:

```
setenv SCRAM_ARCH slc5_amd64_gcc462
cmsrel CMSSW_5_3_5
cd CMSSW_5_3_5/src
cmsenv
```

Checkout the SUSY PAT recipe corresponding to CMSSW_5_3_5 from SUSY PAT twiki and compile the code:

```
addpkg DataFormats/PatCandidates    V06-05-06-03
addpkg PhysicsTools/PatAlgos        V08-09-42
addpkg CommonTools/RecoUtils        V00-00-13
addpkg DataFormats/StdDictionaries  V00-02-14
addpkg RecoParticleFlow/PFProducer  V15-02-06
addpkg JetMETCorrections/Type1MET   V04-06-09
addpkg PhysicsTools/Configuration   V00-12-06
scram b -j9
```

Check out and compile the code for using the POG implementation of the electron working points (we are using electrons identified as isolated according to the working point "VETO" to veto the event):

```
cvs co -r CutBasedId_V00-00-05 -d EGamma/EGammaAnalysisTools UserCode/EGamma/EGammaAnalysisTools
scram b -j9
```

To be able to use the new TrackingPOG filters, the following packages should be checked out

```
cvs co -r V00-00-13 RecoMET/METFilters
cvs co -r V01-00-11-01 DPGAnalysis/Skims
cvs co -r V00-11-17 DPGAnalysis/SiStripTools
cvs co -r V00-00-08 DataFormats/TrackerCommon
cvs co -r V01-09-05 RecoLocalTracker/SubCollectionProducers
```

To apply the patched (event-number based) hcalLaserEventFilter, check out the list of bad events

```
cvs co -r V01-00-04 -d EventFilter EventFilter/HcalRawToDigi/data/AllBadHCALLaser.txt
```

Checkout and compile the RA2 object producers and event cleaning filters (including the latest HO-noise filter):

```
cvs co -r  RA253XAN_07Feb2013V1  -d SandBox/Skims  UserCode/seema/SandBox/Skims
cvs co -d SandBox/Skims/plugins  UserCode/seema/SandBox/Skims/plugins/HONoiseFilter.cc
cvs co -d SandBox/Skims/plugins  UserCode/seema/SandBox/Skims/python/hoNoiseFilter_cfi.py
cvs co -r V04JAN2012_v1 -d UserCode/DataFormats UserCode/lhx/DataFormats
cvs co -r RA2_2013-01-23 -d RA2Classic UserCode/kheine/RA2Classic
scram b -j9
```

Finally, add the ntupler code from the git repository at github

```
git clone git@github.com:mschrode/DAS.git
```

Then, change to the branch of the 2013 school at SINP

```
git checkout 2013-SINP
```

To produce the ntuples, only the `DASTreeMaker` package is required. You can delete the `RA2Exercises` directory. Then, compile the code

```
scram b -j4
```

The `makeDASTreeFromAOD_cfg.py` config in `DAS/DASTreeMaker/test` can be used to produce ntuples from AOD datasets. (There is also a version that runs over the RA2 PAT skims, but since the skims are likely to be deleted soon, it will probably not work.) You will also find crab-config files in the `test` directory.

Datasets and Global Tags

As the analysis, the DAS exercise is performed on the 535 version of the 2012 data, i.e. A, B, C-v1 ReReco and C-v2, D PromptReco.

- The AOD input datasets are listed in Analysis Note AN-12-350. As explained above, the ntuple can also be produced from these RA2 PAT skims; however, the latter are likely to be deleted soon.
- The JSON file to select good lumi sections.

The following global tags have been used:

Dataset	Global tag
2012A+B (Jul13 rereco, 53X)	`FT_53_V6_AN3::All`
2012A (Aug06 rereco, 53X)	`FT_53_V6C_AN3::All`
2012Cv1 (Aug24 rereco, 53X)	`FT53_V10A_AN3::All`
2012Cv2 (prompt reco, 53X)	`FT_P_V42C_AN3::All`
2012D (prompt reco, 53X)	`FT_P_V42_AN3::All`
Summer12_DR53X 53X	`START53_V7G::All`

Contacts

- Matthias Schröder: matthias.schroederATdesy.de

-- MatthiasSchroederHH - 26-Oct-20133

This topic: CMS > SWGuideCMSDataAnalysisSchool2013SUSYJetsPlusMHTSINP
Topic revision: r28 - 15 Nov 2013 - MatthiasSchroederHH