Table of Contents

PAT cleaning exercises

Introduction

This page will guide you through a set of exercises in order to learn more about the importance of event cleaning and the way it is done in PAT. Though there will be hints and reminders throughout the exercises it will require the following knowledge:

requirement	recessing link(s)
create your own EDAnalyzer	WorkBookWriteFrameworkModule
read collections of pat objects in your EDAnalyzer	PAT Example
use the TFileService	SWGuideTFileService
create a PAT Layer1	PAT Layer 1

If you feel uncomfortable in one of these points please follow the corresponding links and make yourself familiar with it. For all exercises the use of a ttbar RelVal sample is suggested which is located at:

/afs/cern.ch/cms/PRS/top/cmssw-data/relval200-for-pat-testing/FullSimTTBar-2_2_X_2008-11-03-STARTUP_V7-AO

Exercise 1

During this exercise you should check the distance in deltaR of the closest electron to a given jet and a view more control histograms which might be of interest.

In your module you should:

- read in both the jet collection and the electron collection via a *cfi* file,
- loop the jet collection,
- in the jet collection loop the electron collection and find the closest electron to the jet in deltaR,
- plot this distance and a view more histograms of interest.

In your configuration file you should:

- make sure you read the proper data,
- produce a standard PAT Layer1
- produce a ttGenEvent, which will allow you to select on different decay modes of the top quark
- select for semi-leptonic ttbar events with an electron in the final state
- add your analyzer.

A few hints on how to create the module/configuration file will be given below:

Module

To read in the corresponding particle collections in the *analyze* function of your EDAnalyzer you should add the following lines:

```
   edm::Handle<edm::View<pat::Jet> > jetHandle;
   iEvent.getByLabel(jetLabel_,jetHandle);
   edm::Handle<edm::View<pat::Electron> > electronHandle;
   iEvent.getByLabel(eleLabel_,electronHandle);
```

The event loop should roughly look like this:

```
  for(edm::View<pat::Jet>::const_iterator jet_iter = jetHandle->begin();
        jet_iter!=jetHandle->end(); ++jet_iter){
          for(edm::View<pat::Electron>::const_iterator electron_iter = electronHandle->begin();
                electron_iter!=electronHandle->end(); ++electron_iter){
                ...
```

A useful function to determine the distance between two pat::Objects you find in the *CMS.PhysicsTools/Utilities*. The use is demonstrated below:

```
#include PhysicsTools/Utilities/interface/deltaR.h
float deltaR = DeltaR<pat::Electron, pat::Jet>()(myElectron, myJet);
```

Apart from the distance of the closest electron to the jet the following distributions might be of interest:

- Number of jets above 10 Gev
- Energy of the closest electron
- Energy of the closest electron over energy of the jet
- Electromagnetic fraction of the jet
- Track isolation of the closest electron

Fill free to add more histograms containing whatever you think might also be of interest.

Configuration

To read the proper data your *source* in the configuration file should look like this:

```
process.source = cms.Source("PoolSource",
    fileNames = cms.untracked.vstring('file:/afs/cern.ch/cms/PRS/top/cmssw-data/relval200-for-pat
    )
)
```

The configuration file for standard PAT Layer1 production can be found here. You should refer to this file as a starting point, when starting from scratch. Run it first to see if it works. Introduce your modifications then and check step by step. This will ease your life when tracing down errors and mis-configurations.

The main sequences for the PAT Layer1 production are these:

```
# PAT Layer 0+1
process.load("CMS.PhysicsTools.PatAlgos.patLayer0_cff")
process.load("CMS.PhysicsTools.PatAlgos.patLayer1_cff")
```

This example is based on a sample of ttbar pair production. To ease the event selection on generator level we make a small excursion to the *Top Quark Analysis Framework* (SWGuideTQAF), which provides a bundle of tools to facilitate top analyses. One of the tools we will use is a pre-selection of semi-leptonic ttbar events with an electron in the final state on generator level. Follow the compilation instructions given on the SWGuideTQAFRecipes for CMSSW_2_2_X and add the following lines to your configuration file:

```
## produce ttGenEvent
process.load("TopQuarkAnalysis.TopEventProducers.sequences.ttGenEvent_cff")

## select semi-leptonic ttbar events with an electron in the final state
process.load("TopQuarkAnalysis.TopEventProducers.producers.TtDecaySelection_cfi")
process.ttDecaySelection.channel_1  = [1, 0, 0]
process.ttDecaySelection.channel_2  = [0, 0, 0]
process.ttDecaySelection.tauDecays  = [0, 0, 0]

## add the two sequences to the process path p1
process.p1 = cms.Path(
    process.makeGenEvt *
    process.ttDecaySelection
)
```

Don't forget also to add your own module. In case you want to make use of the TFileService add the following lines to your configuration file:

```
# register TFileService
process.TFileService = cms.Service("TFileService",
    fileName = cms.string('myHistogramFile.root')
)
```

Now you should be ready to go. Process all events on the file and have a look at your plots.

Exercise 2:

During this exercise you should start playing around with the options of the cleaning at PAT Layer0 for pat::Jets. This will mostly refer to changes on the level of python configuration, which take place during the production of the PAT Layer1. Remember that you may reject overlapping jet, create new collections of overlapping jets or flag them.

In your module you should:

- introduce the possibility to discriminate between flagged and non flagged jets,
- add an steering parameter in your *cfi* file to determine whether you like to use flagged or non flagged jets

In your configuration file you should:

- change the overlap treatment to flag only jets with overlap to *Electrons* (the default is *All*, do you expect any difference?),
- change the overlap treatment to discard jets with overlap to *Electrons*,
- change the overlap treatment to reject jets with overlap to *Electrons* but to keep all jets as an extra collection,
- change the overlap treatment to reject jets with overlap to *Electrons* but to write them into a separate collection,
- change the pt cut of the reference collection of electrons,
- change the isolation of the reference collection of electrons.

A few hints on how to modify the module/configuration file will be given below:

Module

You may find a code snipped how to check pat::Objects like jets for flags here.

Configuration

You should not change the corresponding configuration parameters in the *cfi* file but make use of the parameter replacement in your configuration file. Keep in mind that before you change a parameter of a module in your process you should make it known to the process. This could look like this in your configuration file:

```
process.load("CMS.PhysicsTools.PatAlgos.cleaningLayer0.caloJetCleaner_cfi")
process.caloJetCleaner.bitsToIgnore = ['Overlap/Electrons']
```

Make sure once again that you know how to replace parameters and what kind of parameters you are going to replace. You will find a table of useful links below:

link	comment
SWGuideAboutConfigFile#Modifying_Parameters	how to modify parameters in your configuration file
SWGuideAboutConfigFile#Parameters	list of existing parameters
caloJetCleaner_cfi	calo jet cleaner
electronCleaner_cfi	electron cleaner
here	list of potential flags

Exercise 3:

Try to switch from the semi-leptonic decay channel with electrons to muons in the final state and repeat Exercise 2. Therefore you should change to the following lines in your configuration file:

```
## select semi-leptonic ttbar events with an electron in the final state
process.load("TopQuarkAnalysis.TopEventProducers.producers.TtDecaySelection_cfi")
process.ttDecaySelection.channel_1  = [0, 1, 0]
process.ttDecaySelection.channel_2  = [0, 0, 0]
process.ttDecaySelection.tauDecays  = [0, 0, 0]
```

For a hint what the logic behind this is have a look into the *cfi* file. You will also need to uncomment a view lines in the *caloJetCleaner_cfi*. Do you expect the same behaviour as for electrons?

Exercise 4:

Make yourself familiar with the event cleaning for photons. You will find the corresponding cleaner in *CMS.PhysicsTools/PatAlgos/pytho/cleaningLayer0.*

-- RogerWolf - 27 Jan 2009

This topic: CMSPublic > SWGuidePATv1CleaningExercises
Topic revision: r4 - 2010-04-28 - RogerWolf