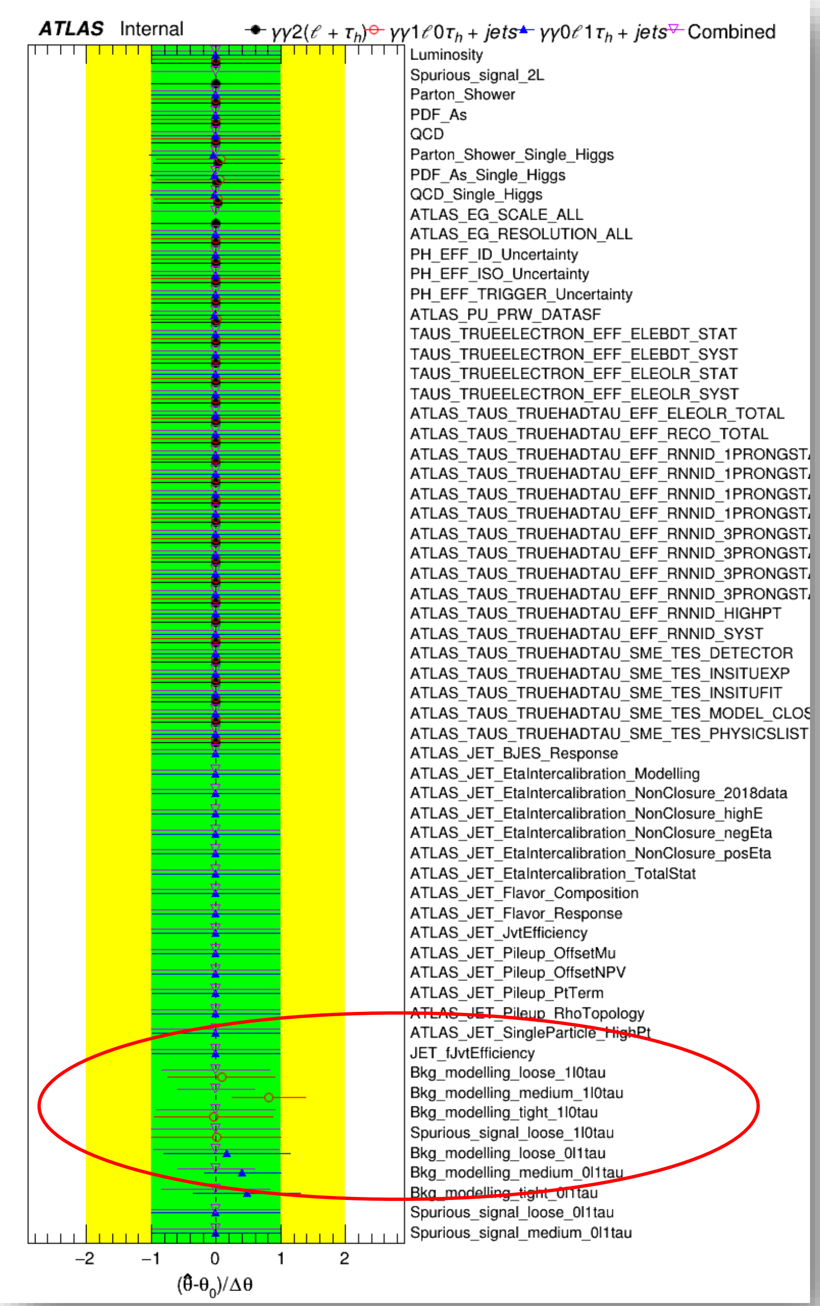
Some update

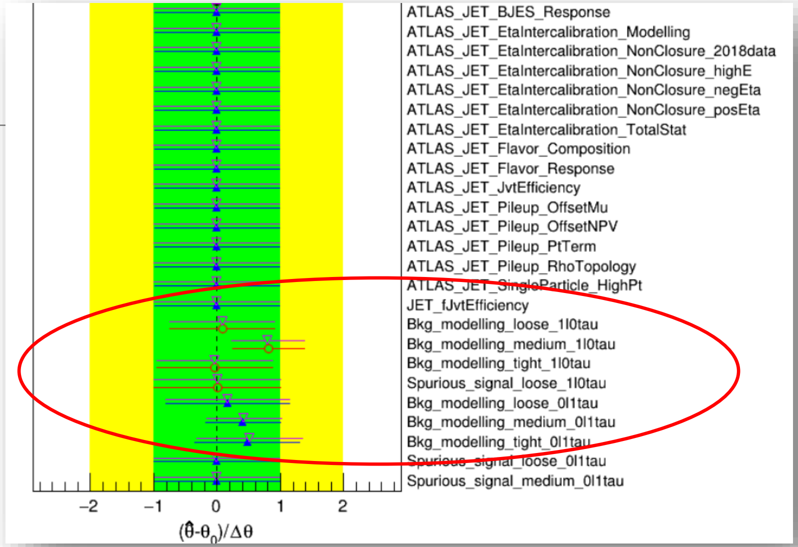
- Check the pull and constrain in unblinding pull plots.

---Will give a report about unblinding results in the Friday HH meeting. (not finish yet)

- Complete the framework using the AnaWSBuilder for computing the limit outcome in the Zprime -> mumu analysis.

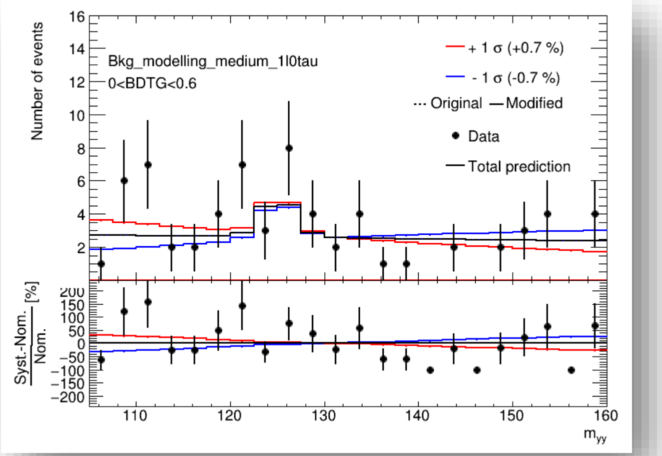


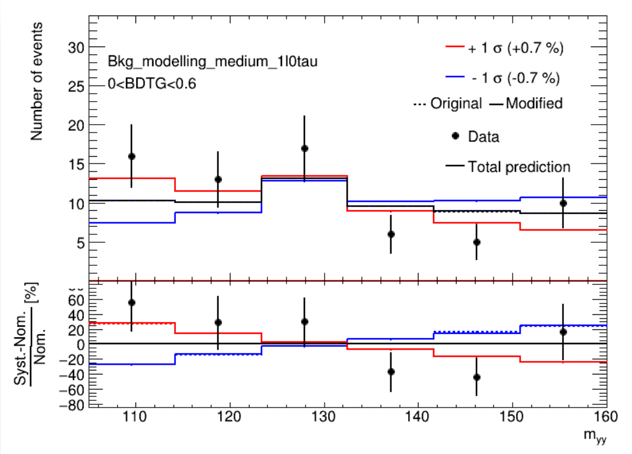
Check the pull and constrain



- **These pulls come from sideband data and reasonable.**
  - The sys plots with data(e.g. Bkg_modelling_medium_1l0tau, the NP with most pull.)

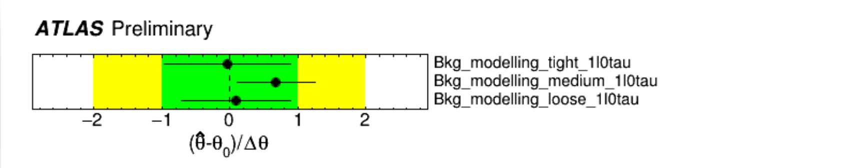
For better understand, we can rebin the plot and find the shape of sideband data is obviously conforms to the shape with $+1\sigma$ bkg uncertainties.





- Another check: fit to data sideband only:

We can get the similar pull: (So these pull come from sideband data)



Since our background modeling is based on the 0l0tau+1j data sideband, we compared the MC 1l0tau and 0l0tau+1j to detect the bin-by-bin deviations. These deviations were then applied to the fake sample (nominal one) for background modeling with $+1\sigma$ bkg uncertainties. (Red one)

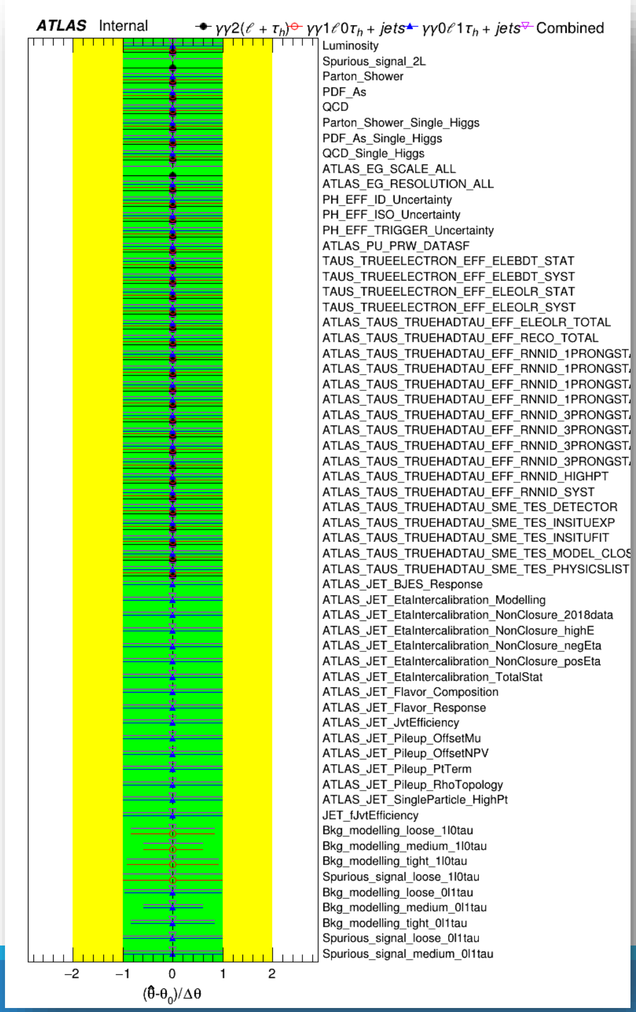
Therefore, it is reasonable to expect that the red shape (with $+1\sigma$) more closely matches the data.

Check the pull and constrain

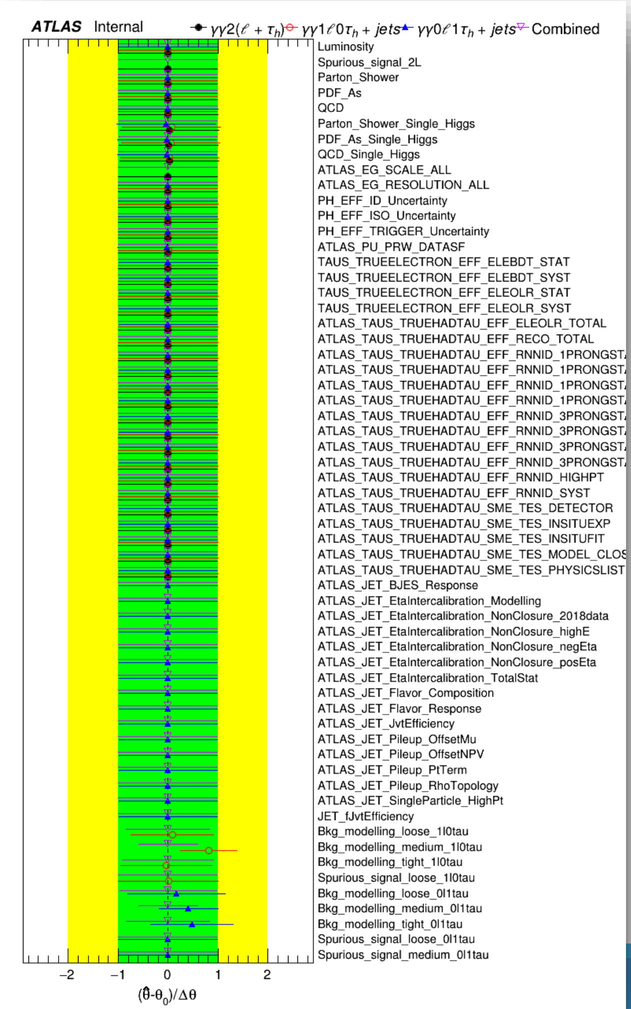
➢Regarding constraints, they are indication of overestimation of the systematics and not from data.(Just come from the background uncertainties.)

(Same constraints in Asimov results. It is not a problem if no one ask before unblinding)

Asimov:



Data:

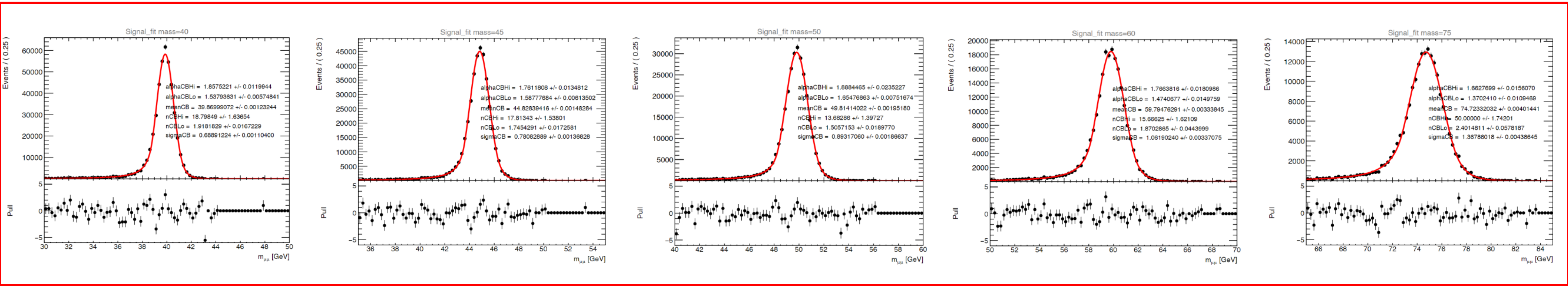


Zprime - Signal modeling

- MC signal samples (now use Dark photon samples to mimic) produced for several mass points:

$m_Z = 30, 35, 40, 45, 50, 60, 75$ (GeV); Now we just use the 40~75 GeV.

- Parameterization derived on those points and provided function to continuously cover the mass range [38, 75] GeV.
- We can get the signal modeled with a Double-Sided Crystal Ball (DSCB) function.



Mass(GeV)	alphaH	alphaL	mean	nH	nL	sigma
40	1.8575221	1.53793631	39.86999072	18.79849	1.9181829	0.68891224
45	1.7611808	1.58777684	44.82839416	17.81343	1.7454291	0.78082889
50	1.8884465	1.65476863	49.81414022	13.68286	1.5057153	0.89317060
60	1.7663816	1.4740677	59.79476291	15.66625	1.8702865	1.06190240
75	1.6627699	1.3702410	74.72332032	50.00000	2.4014811	1.36786018

Background Estimation-Fit Function Method

Damiano

$$f_{LowMass}(\mu\mu) = a_1/(m_Z - m_{\mu\mu})^{a_2} + a_3 \cdot m_{\mu\mu}^{a_4}(1 - a_5/m_{\mu\mu}^{a_6}) \qquad (6.3)$$

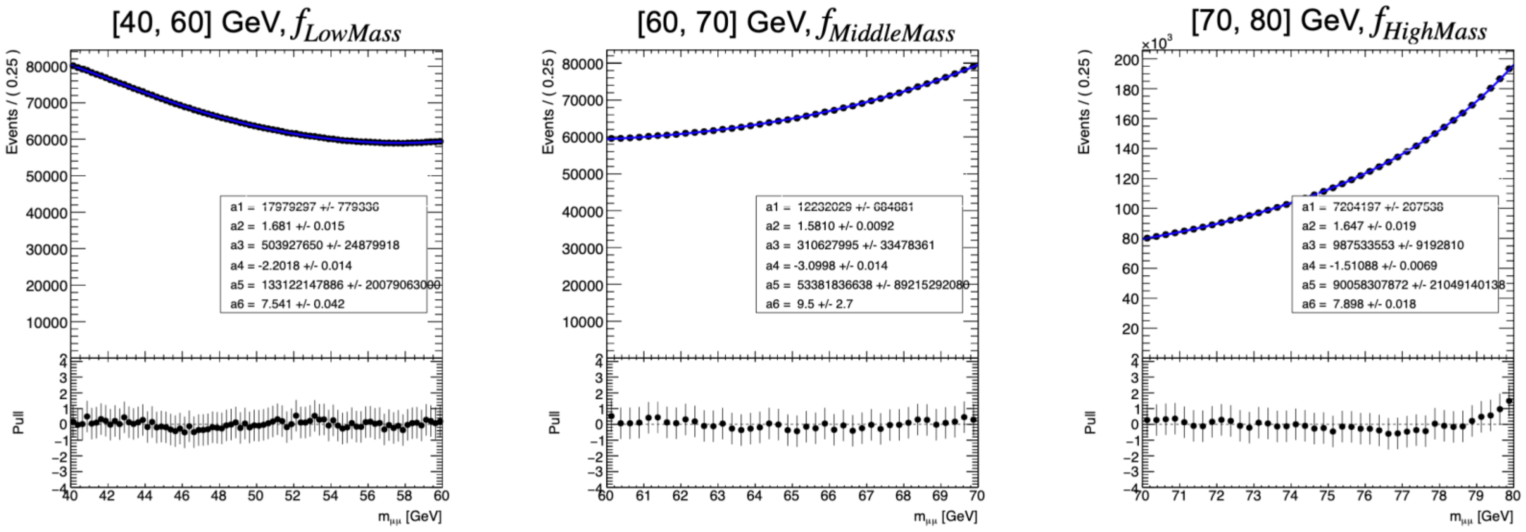
$$f_{MiddleMass}(\mu\mu) = a_1 \cdot m_{\mu\mu}/(m_Z^2 - m_{\mu\mu}^2)^{a_2} + a_3 \cdot m_{\mu\mu}^{a_4}(1 - a_5/m_{\mu\mu}^{a_6}) \qquad (6.4)$$

$$f_{HighMass}(\mu\mu) = a_1 \cdot m_{\mu\mu}/(m_Z - m_{\mu\mu})^{a_2} + a_3 \cdot m_{\mu\mu}^{a_4}(1 - a_5/m_{\mu\mu}^{a_6}) \qquad (6.5)$$

Background Modeling

Jie

- Break down [40, 80] GeV into 3 ranges, and fit the background distributions in the 3 range with $f_{LowMass}$,$f_{MiddleMass}$ and $f_{HighMass}$ separately:

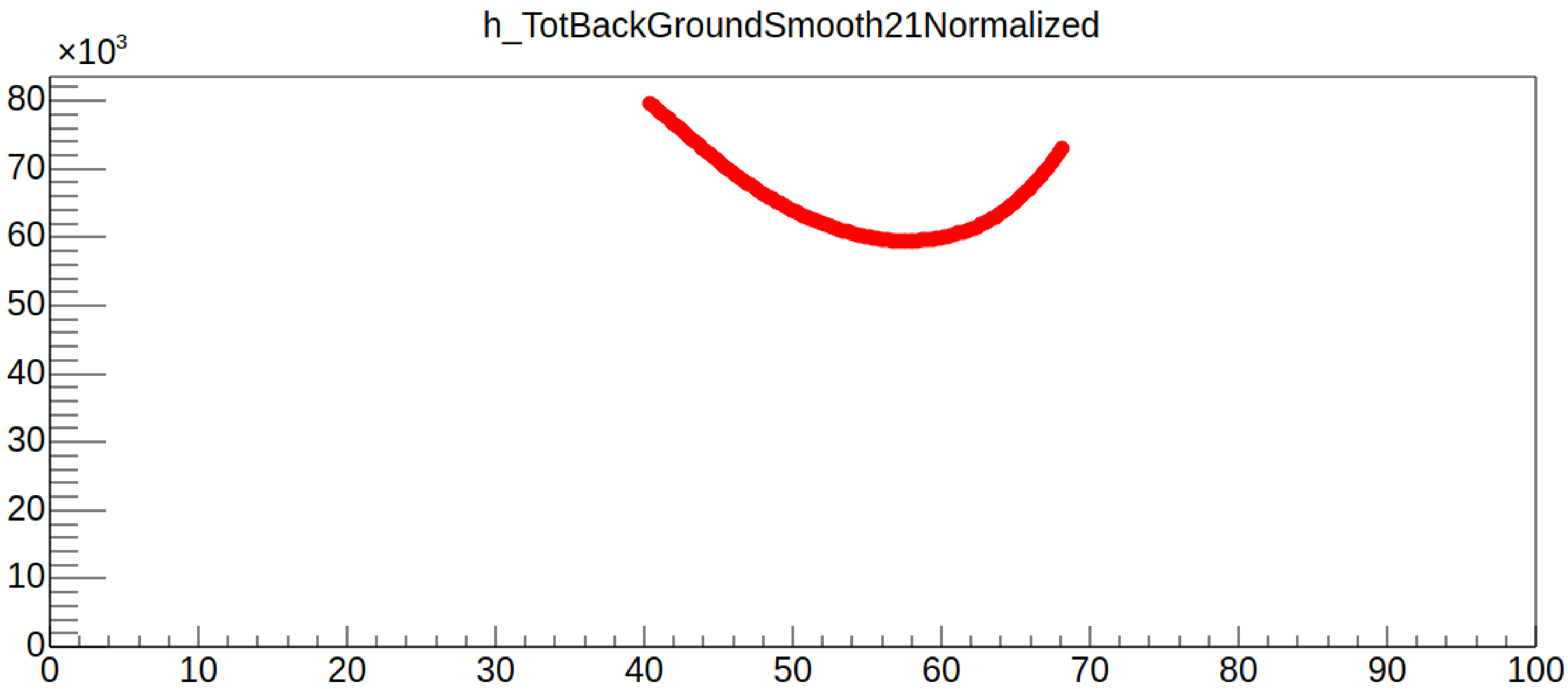


- Use different mass range affect the model.

(No decision on which interval should be used)

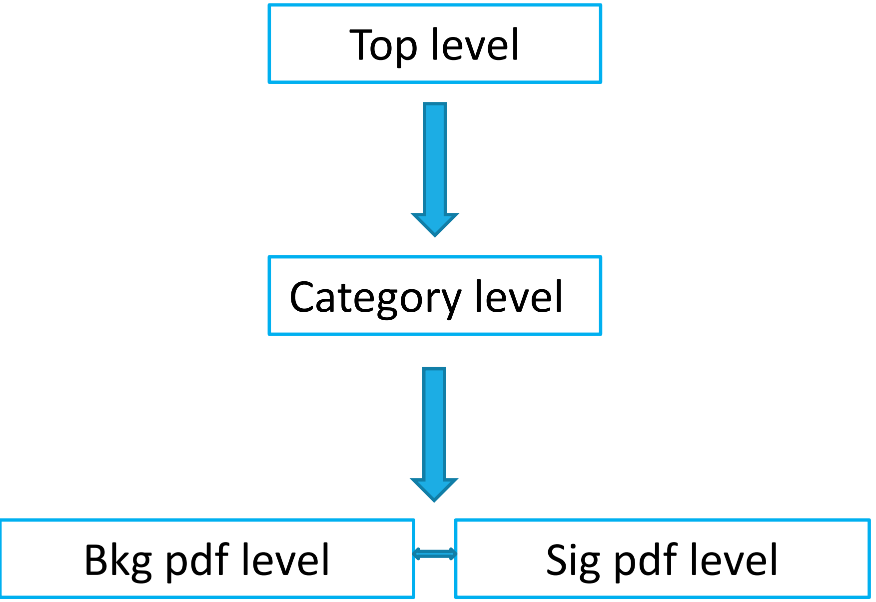
Background Estimation-GP

- Rough repeat of other's work --- Using gaussian progress to estimate the background.



Upper limit-Using AnaWSBuilder

XML Twiki



The **top-level card** is a single XML file which we feed to the executable.

The top-level card points to several **category-level cards**, each describes how to construct the likelihood model for the corresponding category.

Finally, for each physics process, the category-level card will point to a **pdf-level card** constructs the pdf that describes the shape of the process.

Upper limit-Using AnaWSBuilder---Example

Top level

```
<!DOCTYPE Combination  SYSTEM 'AnaWSBuilder.dtd'>
<Combination WorkspaceName="combWS" ModelConfigName="ModelConfig" DataName="combData" OutputFile="workspace/lowmass_Zmumu/lowmass_Zmumu_MC_mass60.root"
Blind="false" >
  <Input>projects/config/category_level_mass60.xml</Input>
  <POI>mu</POI>
  <Asimov Name="asimovData_0" Setup="mu=0" Action="fixsyst:fit:genasimov:float:savesnapshot" SnapshotGlob="nominalGlobs" SnapshotNuis="nominalNuis"/>
  <Asimov Name="asimovData_1_prefit" Setup="mu=1" Action="genasimov:reset"/>
</Combination>
```

Category level

```
<!DOCTYPE Channel SYSTEM 'AnaWSBuilder.dtd'>
<Channel Name="lowmass_Zmumu" Type="shape">
  <Data InputFile="projects/data/outputHistSuperFastNominal.root" FileType="histogram" HistName="h_TotBackGroundSmooth21Normalized" Observable="invMass_mumu[40,
  60]" Binning="80" InjectGhost="true"/>
  <Sample Name="signal" InputFile="projects/config/pdf_level_signal_mass60.xml">
    <NormFactor Name="n_sig[295691.05]"/>
    <NormFactor Name="mu[1,0,5]"/>
  </Sample>
  <Sample Name="background" InputFile="projects/config/pdf_level_background_mass60.xml">
    <NormFactor Name="n_bkg[5259949.1,0,1e9]"/>
  </Sample>
</Channel>
```

Bkg pdf level

Sig pdf level

Bkg function /GP:use histpdf

```
<!DOCTYPE Model SYSTEM 'AnaWSBuilder.dtd'>
<Model Type="UserDef">
  <ModelItem Name="EXPR::bkgPdf('@1/TMath::Power(91.2-@0,@2)+@3*TMath::Power(@0,@4)*(1-@5/TMath::Power(@0,@6))', :observable:, a1[17979297,0.9e+07,2e+07], a2[1.
  681,1,3], a3[503927650,4e+08,5.5e+08], a4[-2.2018,-3,-2], a5[133122147886,0.5e+11,2e+11], a6[7.541,5,10])"/>
</Model>
```

```
<!DOCTYPE Model SYSTEM 'AnaWSBuilder.dtd'>
<Model Type="External" Input="projects/model/template.root" WSName="template" ModelName="basicBkgFrom4ParamFit" ObservableName="x" >
  <Item Name="unit[1]"/>
  <Rename OldName="Lumi" NewName="unit"/>
</Model>
```

Sig function(DSCB)

```
<!DOCTYPE Model SYSTEM 'AnaWSBuilder.dtd'>
<Model Type="UserDef">
  <Item Name="alphaCBLo[1.4828267]"/>
  <Item Name="meanCB[49.82619287]"/>
<!--  <Item Name="meanCBNom[44.84342163]"/> -->
  <Item Name="nCBLo[2.1703524]"/>
  <Item Name="sigmaCB[0.87957827]"/>
<!--  <Item Name="sigmaCBNom[0.80026177]"/> -->
  <Item Name="alphaCBHi[1.8124855]"/>
  <Item Name="nCBHi[22.73915]"/>
<!--  <Item Name="prod::sigmaCB(sigmaCBNom, respRes)"/> -->
<!--  <Item Name="expr::meanCB('(@0+@1-125)*@2', meanCBNom, mH[125], respScale)"/> -->
  <ModelItem Name="RooTwoSidedCBShape::signalPdf(:observable:, meanCB, sigmaCB, alphaCBLo, nCBLo, alphaCBHi, nCBHi)"/>
</Model>
```