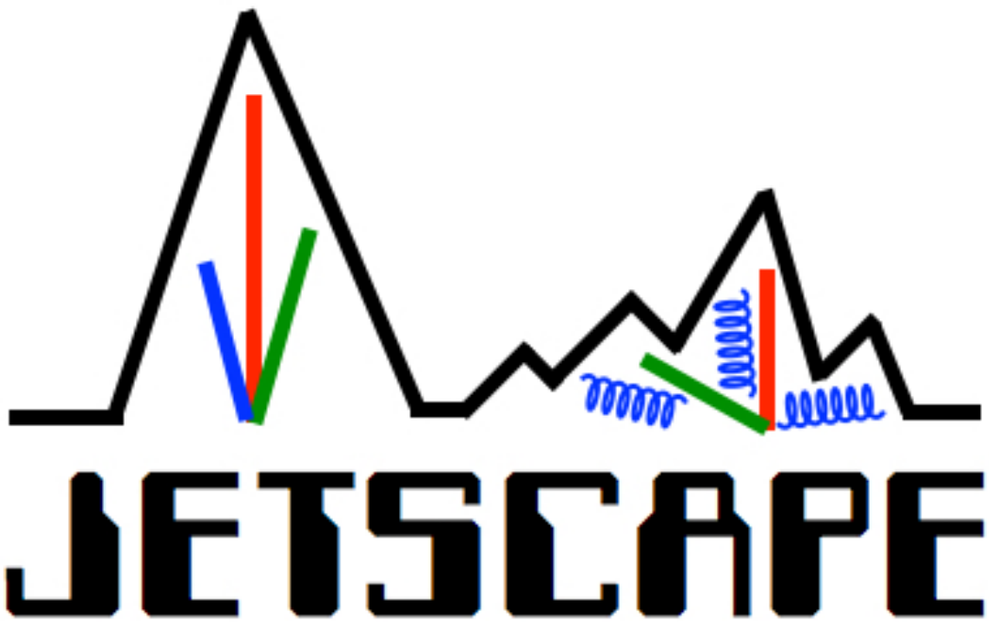


Data Curation & Error Analysis

Ron Soltz, Wayne State University, LLNL
JETSCAPE Winter School, Texas A&M
January 10, 2019

JETSCAPE Data & Errors

- JETSCAPE Data Philosophy
  - Compare to published data
  - Automatically fetch and cache data and errors from HEPdata site
  - Use experimentally determined statistical and systematic errors as specified by experiments
- CAVEATS:
  - Experimental treatment of systematic errors is not (yet) standard
  - May detailed understanding of experimental analyses (i.e. active participation from each experiment is essential)

HEPdata Extraction Mechanics (1)

- Currently handled within hic-param-2017 repo
  - src/_init.py

__init__.py

New Open Recent Revert Save Print Undo Redo Cut Copy Paste Search Preferences Help

scratch 1 | __init__.py 2 | expt.py 3

```
#: Sets the collision systems for the entire project,
#: where each system is a string of the form
#: ``'<projectile 1><projectile 2><beam energy in GeV>'``,
#: such as ``'PbPb2760'``, ``'AuAu200'``, ``'pPb5020'``.
#: Even if the project uses only a single system,
#: this should still be a list of one system string.
systems = ['PbPb2760','PbPb5020']

#: Dictionary of the experimental data.
#: Form MUST be exp_data_list[system][observable][subobservable][{'y':,'x':,'yerr':{'stat':,'sys'}}].
#:      'y' is a (1 x p) numpy array of experimental data.
#:
#:      'x' is a (1 x p) numpy array of numeric index of columns of Y (if exists). In the example data, x is p_T.
#:
#:      'yerr' is a dictionary with keys 'stat' and 'sys'.
#:
#:      'stat' is a (1 x p) array of statistical errors.
#:
#:      'sys' is a (1 x p) array of systematic errors.
```

-:**- __init__.py 18% (33,2) Git-fetch-hepdata (Py Outl)

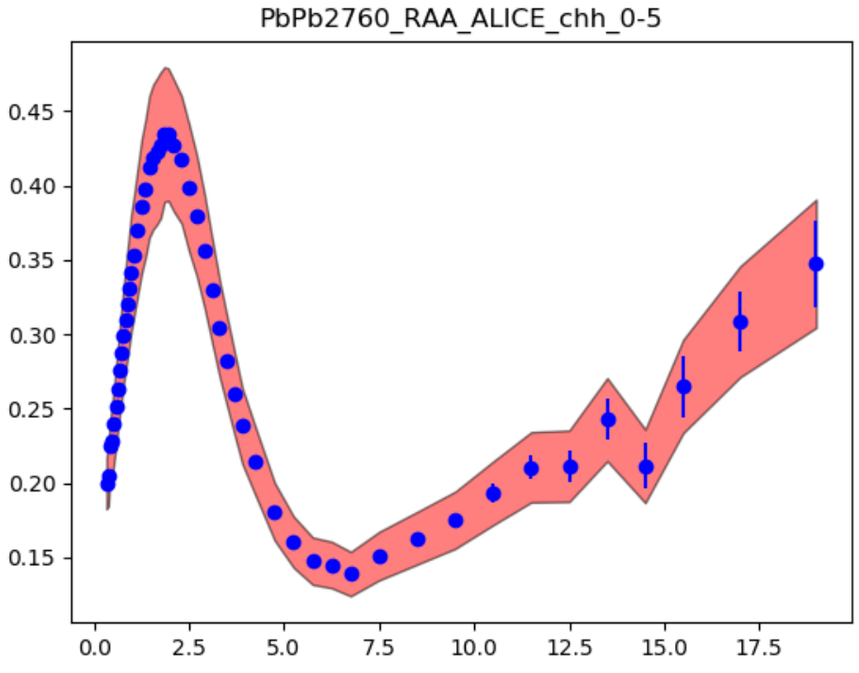
HEPdata Extraction Mechanics (2)

scratch 1 | __init__.py 2 | expt.py 3

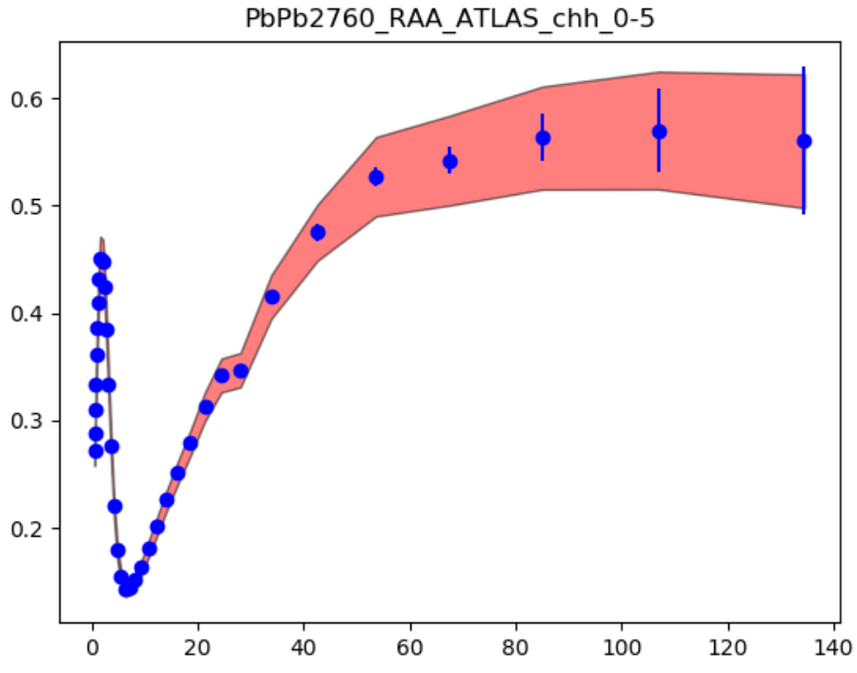
```
def _data():
    """
    Curate the experimental data using the `HEPData` class and return a nested
    dict with levels

    - system
    - observable
    - subobservable
    - dataset (created by :meth:`HEPData.dataset`)

    For example, ``data['PbPb2760']['dN_dy']['pion']`` retrieves the dataset
    for pion dN/dy in Pb+Pb collisions at 2.76 TeV.

    Some observables, such as charged-particle multiplicity, don't have a
    natural subobservable, in which case the subobservable is set to `None`.

    The best way to understand the nested dict structure is to explore the
    object in an interactive Python session.

    """
    data = {s: {} for s in systems}

#    data['PbPb5020']['dNch_deta'] = {None : HEPData(1410589,2).dataset('centrality',r'$\mathrm{d}N_\mathrm{ch}/\mathrm{d}\eta$')}
    data['PbPb2760']['RAA'] = {}
    data['PbPb2760']['RAA']['ALICE_chh_0-5'] = HEPData(879583,3).dataset('PT','RAA')
    data['PbPb2760']['RAA']['ATLAS_chh_0-5'] = HEPData(1360290,33).dataset('PT','RAA',CENTRALITY='0-5')
    data['PbPb2760']['RAA']['ALICE_jetR0.2_0-10'] = HEPData(1343112,6).dataset('PT','RAA')
    data['PbPb2760']['RAA']['ATLAS_jetR0.4_0-10'] = HEPData(1326911,27).dataset('PT','RAA')

    data['PbPb5020']['RAA'] = {}
    data['PbPb5020']['RAA']['CMS_chh_0-5'] = HEPData(1496050,8).dataset('PT','RAA')

    return data

#: A nested dict containing all the experimental data, created by the
#: :func:`_data` function.
data = _data()


def cov(
        system, obs1, subobs1, obs2, subobs2,
        stat_frac=1e-4, sys_corr_length=100, cross_factor=.8,
        corr_obs={
            frozenset({'dNch_deta', 'dET_deta', 'dN_dy'}),
        }
):
    """
-:---  expt.py      51% (237,43)  Git-fetch-hepdata  (Py Outl)
```

HEPdata Extraction Plots

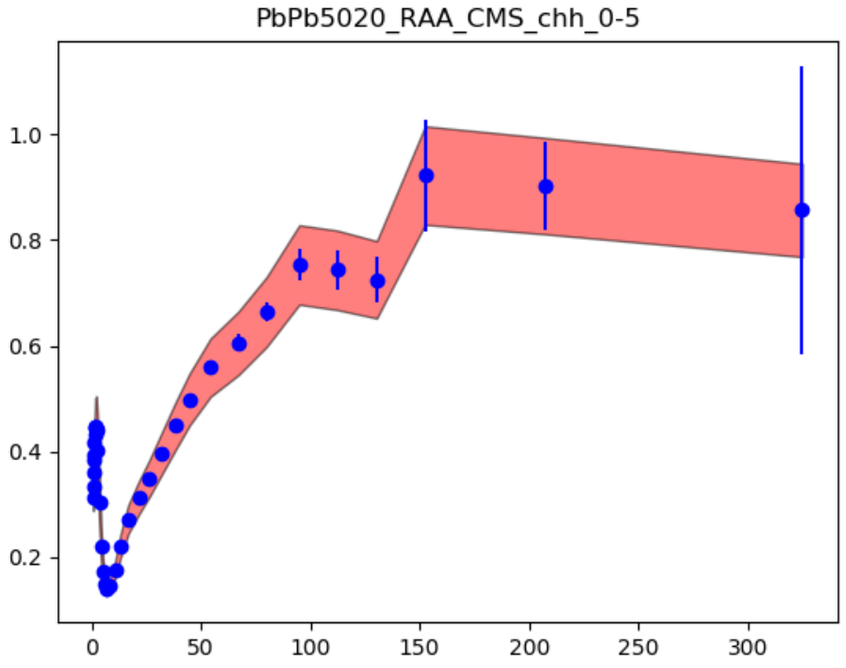
ALICE https://www.hepdata.net/record/ins879583 Table 3



ATLAS https://www.hepdata.net/record/ins1360290 Table 33



CMS https://www.hepdata.net/record/ins1496050 Table 8



Systematic Errors

- Two common methods to specify

  1. Parametric, usually assumed fully-correlated

$$\tilde{\chi}^2(\epsilon_j) = \left[\left(\sum_{i=1}^{n} \frac{(y_i - \mu_i + \sum_k \epsilon_k \sigma_{k,i})^2}{\sigma_i^2}\right) + \sum_k \epsilon_k^2\right]$$

generalize from PRC77.064907 (2008)

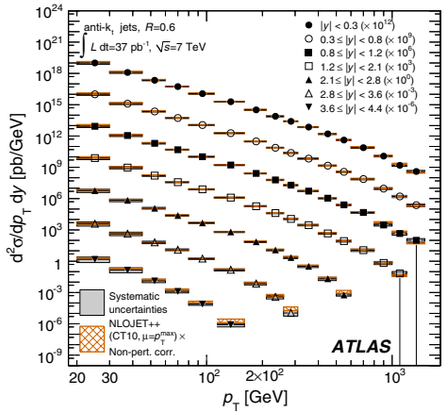
  2. Covariance matrix

- Parametric can be mapped to matrix if unconstrained

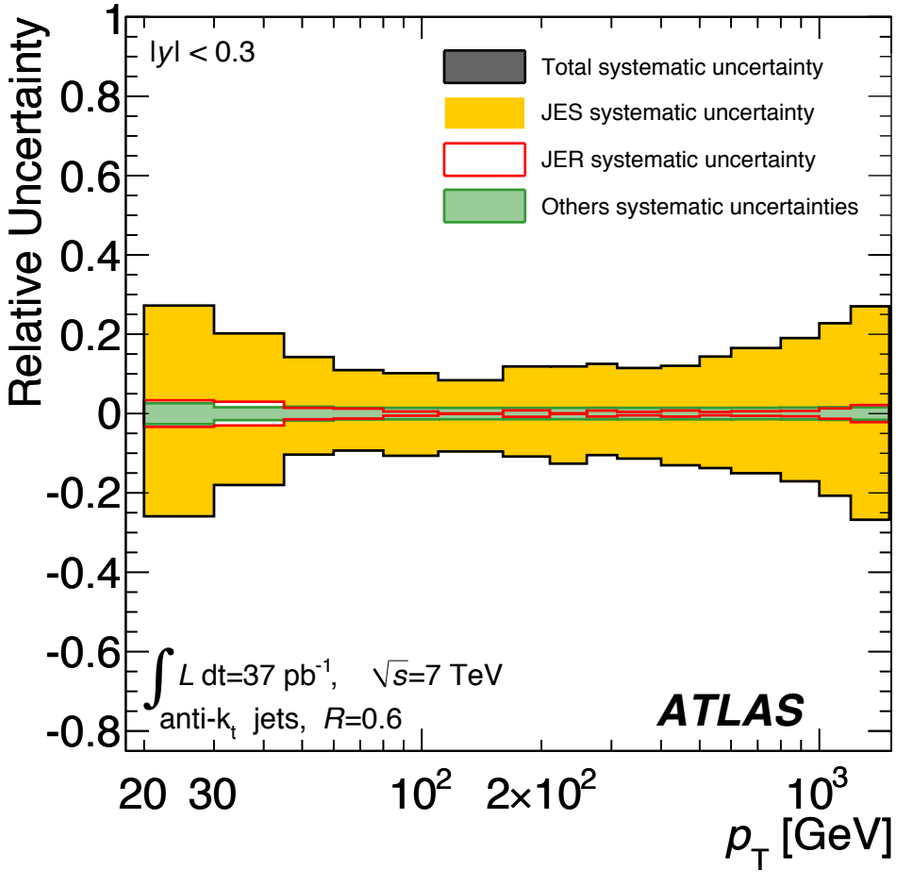
$$\begin{aligned} \chi^2_{min} &= \Delta^T C^{-1} \Delta \\ \Delta_i &= y_i - \mu_i \\ C_{ij} &= \sigma_i^2 \delta_{ij} + \sum_k \sigma_{k,i}\sigma_{k,j} \end{aligned}$$

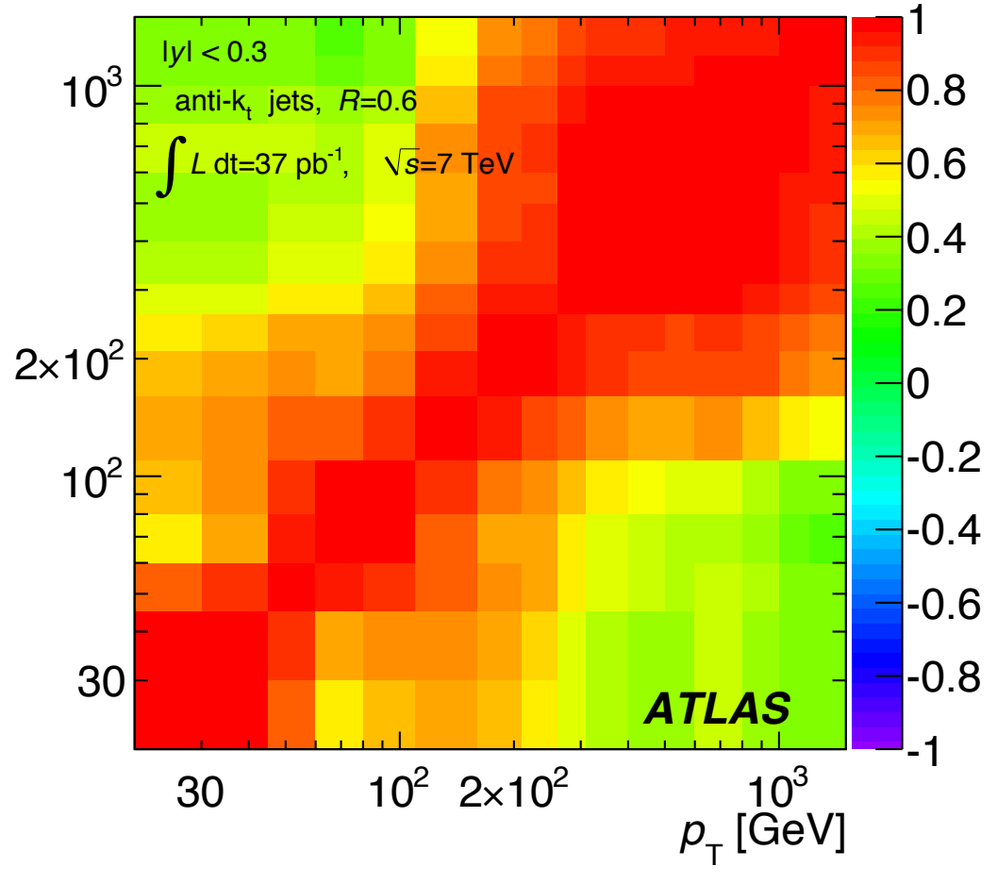
CDF-MEMO-8661 (1999), PRD.89.03309 (2014)

ATLAS Covariance Error Matrix for Inclusive Jets



Measurement of the jet radius and transverse momentum dependence of inclusive jet suppression in lead–lead collisions at sqrt(s_{NN})= 2.76 TeV with the ATLAS detector - PRD.86.014022 Fig 6





note: HEPdata does not provide standard, but can store error matrix in 2D table

JETSCAPE Data & Error Plans

- Tasks for JETSCAPE Collaborators
  - Incorporate Data/Stats package in JETSCAPE 2.0 release
  - Include configurations for all published results for jet/hadron R_{AA}, jet/hadron v_2, di-jet asymmetry, fragmentation functions
- Tasks for JETSCAPE Associates (Experimentalist Collaborators)
  - Confirm/improve treatment of systematic errors
  - Extend data-comparisons to jet sub-structure measurements

For Discussion

- Two approaches for Model-to-Data Comparisons

  1. Remove Experimental Effects from Measurements

  2. Incorporate Experimental Effects in Model Outputs

- My answer: it depends (acceptance vs. resolution)

revisit within context of RIVET and Background Subtraction discussions