Data Combination using Aplcon: Tutorials

Gero Flucke

School on Data Combination and Limit Setting
October 4-7, 2011
DESY, Hamburg

Outline

- Introduction and Technicalities
- Technical Example: How to combine two measurements using Aplcon
- Combining $W \to$ lepton branching ratios: Effect of correlations.
- Multiplicative uncertainties: Combining two m_{top} measurements I
- **Break** - if not earlier...
- Multiplicative uncertainties: Combining two m_{top} measurements II
- Further exercise suggestions.

Software

Environment

- All examples will use Aplcon as tool for data combination:
  - overdoing for simple cases,
  - but helps to get familiar with it.
- Aplcon is Fortran code, written by V. Blobel.
- We will use C++ wrappers to call it from ROOT macros.

Download

- Start a terminal.
- Create directory `aplcon`: `mkdir -p ~/dataComb1/aplcon`
  (Do not choose another location - or adjust `loadAplconLibs.C`!).
- `cd ~/dataComb1/aplcon`
- Download tar ball `aplcon.tgz` from school indico:
  https://indico.desy.de/getFile.py/access?subContId=1&contribId=4&resId=0&materialId=0&confId=4489
- Unpack: `tar xzf aplcon.tgz`

Software II

Aplcon source files

- Fortran code: `*.F`
- Fortran common block definitions (~global variables): `*.inc`
- `Aplcon.h`, `Aplcon.C`: C++ wrappers in namespace `Aplcon`
- `Makefile`

Directories

- `libs`: libaplcon.so (already compiled)
- `templates`: files to edit during tutorials
- `results`: examples for comparison

Calling Aplcon within ROOT

- `cd templates`
- `root -l loadAplconLibs.C <yourMacro>.C+`
  (using '+' to compile macro using ACLiC!)

Basic Calling Sequence

- Initialise with number of variables and constraints:
  `Aplcon::aplcon(nVar, nConstr);`
- Select options (variable pdfs, print options,. . .)
- Set variables and covariance:
  - `double vars[nVar] = {...};`
  - `double cov[...]  = {...};`
- Loop until convergence:
  - calculate values of constraints from current variable values:
    `double constraints[nConstr] = {...};,`
  - pass variables, covariance and constraints to aplcon:
    `Aplcon::aploop(vars, cov, constraints, iret);,`
  - get new variables, covariance in return.

Technical Details

- Fortran knows only floating points (float, double) and integer:
  - no `unsigned int`, no `const` in C++ wrappers.
- Symmetric covariance matrix passed as 1D array:
  - storing lower left triangle,

```
double cov[nVar * (nVar + 1))/2] =
      {c_11,                \\row 1
       c_21, c_22,          \\row 2
       c_31, c_32, c_33,    \\row 3
       ...
      };
```

  - index can also be retrieved via `Aplcon::ijsym(i,j)`:
    `cov[Aplcon::ijsym(2,1)] = c_32.`
    (C++ array index counting!).

Technical Example

See how it works: combineTwo.C

Simple example to combine two correlated measurements x_1, x_2, given σ_1, σ_2 and correlation coeffcient ρ.

1. `cd templates`
2. Open `combineTwo.C` in your favourite editor.
3. Try to understand the program flow.
4. Edit the two places marked as `FIXME` (i.e. fill covariance matrix).
5. Compile and run with ROOT:
   `root -l loadAplconLibs.C combineTwo.C+`.
6. Compilation fails? Edit & (re-)run `.x combineTwo.C+` in ROOT.
7. Try to understand the text output.
8. Bonus: How to access the result in the code?
   Try to print it using `std::cout`.

Compare with `result/combineTwo.C`.

Fictitious Example

BR($W \to e$), BR($W \to \tau$)

- Taken from section 4 of: A. Valassi, NIMA 500 (2003) 391-405.
- Experiments A and B measuring BR($W \to e$) and BR($W \to \tau$):
  - $\mathrm{B}_A^e = (10.5 \pm 1.0)\%$
  - $\mathrm{B}_B^e = (13.5 \pm 3.0)\%$
  - $\mathrm{B}_A^\tau = (\ \ 9.5 \pm 3.0)\%$
  - $\mathrm{B}_B^\tau = (14.0 \pm 3.0)\%$
- Testing effect of correlations on combined results B^e and B^τ:
  - no correlations,
  - correlations between measurements of same observable (B^e),
  - correlations between measurements of different observables (B_B),
  - positive and negative correlations.
- Assuming lepton universality one can further combine to BR($W \to l$).

BR($W \to e$), BR($W \to \tau$): The Task

- `templates/avalassi.C`: no correlations, no lepton universality.
- Run `root -l -q loadAplconLibs.C avalassi.C+`.
- Note down BR($W \to e$) and BR($W \to \tau$), their uncertainties and the χ^2 in table in `templates/avalassi.txt`.
- Add correlations between two measurements according to table (in doubt see `results/avalassi_r12.C` e.g. for $\rho_{12} = 15\%$) and fill upper half of table.
- Change one and add another line to assume lepton universality (in doubt see `results/avalassi_lu.C`) and fill lower half of table.
- Compare each column with no-correlation case:
  - Focus on uncertainties.
  - Which effect has the sign of the correlation?

Multiplicative Uncertainties

Combining two Top Mass Measurements from the Same Experiment

- $m_1 = 170 \pm 5$ GeV.
- $m_2 = 175 \pm 2$ GeV.
- Additional common relative error ϵ: e.g. jet energy scale.
- Approach: Compose covariance matrix from both contributions,
$$\boldsymbol{V} = \begin{pmatrix} 25. & 0. \\ 0. & 4. \end{pmatrix} + \epsilon^2 \begin{pmatrix} m_1^2 & m_1 m_2 \\ m_1 m_2 & m_2^2 \end{pmatrix}.$$
- Pass masses and covariance to e.g. `aplcon`.

Multiplicative Uncertainties: The Task

- Take `templates/simple_top.C` (with $\epsilon = 5\%$).
- Correctly calculate covariance matrix (replacing FIXME).
- In doubt, have a look at `results/simple_top.C`.
- `root -l -q loadAplconLibs.C simple_top.C+`
- Test common relative error ϵ in range 0-20%.
- What happens to the combined result?
- Does this make sense?

Break

Multiplicative Uncertainties: Second Try

Combining two Top Mass Measurements from the Same Experiment

- $m_1 = 170 \pm 5$ GeV.
- $m_2 = 175 \pm 2$ GeV.
- Additional common relative error ϵ: e.g. jet energy scale.

Two new approaches:

1. 3 variables (m_1 m_2, ϵ) and 1 constraint: $\epsilon\, m_1 - \epsilon\, m_2 = 0$,
    - uncertainty of combined m not calculated,
    - add $\epsilon\, m$ 'by hand' afterwards.
2. 4 variables (m_1 m_2, ϵ, unmeasured m_{ave}) and 2 constraints: $\epsilon\, m_{ave} - m_1 = 0$ and $\epsilon\, m_{ave} - m_2 = 0$,
    - directly provides σ_{ave}.

Multiplicative Uncertainties: Second Try, the Task a)

Combining two Top Mass Measurements from the Same Experiment

Approach 1: 3 variables, 1 constraint.

- `templates/precise_top.C`
- `root -l -q loadAplconLibs.C precise_top.C+`
- What happens with ϵ in fit? Why?
- Do error propagation for average and add printout, i.e. uncomment lines with FIXME
  (in doubt see `result/precise_top.C`).
- What happens now if you change ϵ?

Multiplicative Uncertainties: Second Try, the Task b)

Combining two Top Mass Measurements from the Same Experiment

Approach 2: 4 variables, 2 constraints.

- `templates/precise_top2.C`
- Edit FIXMEs to add unmeasured variable m_{ave} (in doubt see `result/precise_top2.C`...).
- `root -l -q loadAplconLibs.C precise_top2.C+`
- See how Aplcon does error propagation for you, i.e. compare for two values of ϵ with `precise_top.C`.

Bonus:

- Switch on profile analysis for m_{ave} (`Aplcon::aprofl(3);`) and whatch new print out.
- Is the uncertainty symmetric? Why?
- As normalisation, ϵ is probably log-normal distributed (`Aplcon::aplogn(2);`) - what changes?

Bonus: No templates

Choose as you like:

- Write triangle fit from lecture (p. 40).
- Write straight line fit with uncertainties in both x and y,
  - use example data as in lecture (p. 28 or 29, read values by eye. . .),
  - switch to parabola instead of straight line.
- Fit branching ratios as in lecture
  (p. 45, but data missing: see black board [?])

Instead of a Summary

- I hope you enjoyed playing around with Aplcon - and recognised its flexibility.
- Refined (C++) interface and some more instruction: will be made available later this year.
- Thanks for attention: **See you at the reception!**