PAW

– Introduktion til programmet PAW.

Physics Analysis Workstation (PAW) er et program udviklet på CERN til fitting og visualisering af data. Det er hverken det nyeste eller smarteste, men har et nemt interface og er forholdsvist simpelt.

I PAW skriver man så meget af en kommando, som det kræves for at den ikke kan misforstås, fx. `histogram/plot` bliver til `h/pl` og `vector/create` til `v/cre`. Ved at skrive `help` kan man få en liste over kommandoerne og deres virkemåde; både i PAW og i nedenstående betyder firkantede parenteser [] at noget er valgfrit (optional). Muligheder (options) og indstillinger (settings) reguleres med kommandoerne `set` og `opt`.

h/file LUN filename `h/file 1 distributions.hbook`
Ovenstående kommando bruges til at åbne en fil, *filename*, der indeholder histogrammer. Logical Unit Number (LUN) kan vælges frit og bruges senere i kommandoer vedrørrende filen, fx. *close LUN*. Histogrammerne skrives dog også i hukommelsen, og for at slette dem skrives `h/del 0`. Det skal nævnes at PAW *ikke* forstår store bogstaver, så brug kun små.

h/pl ID [c,e,s,lego,surf] `h/pl 1 e`
Plotter histogrammet med nummeret ID. De forskellige options er: *c* continuous – som en kurve, *e* errors – med fejlbarer, *s* superimpose – ovenpå det forrige histogram, *lego* lego – som et lego plot, *surf* surface – som en overflade. *lego* og *surf* er beregnet til 2D histogrammer.

h/fit ID[interval] function `h/fit 1(5:15) g`
Fitter histogrammet med nummeret ID. PAW har indbygget følgende funktioner: *g* Gaussisk, *e* exponentiel, *pX* polynomium af grad X (fx. *p3*). Andre funktioner fittes ved: *h/fit ID function ! Nparameters parameter-vector*, hvor *function* er en brugerdefineret funktion i Fortran. `h/fit 1 func.f ! 3 par`.

v/cre name(Nparameters) [R values] `v/cre par(3) R 100.0 5.0 0.1`
Definerer en vektor, *par*, med tre indgange (entries). Det valgfrie *R* giver mulighed for at indtaste værdier. Arrays defineres på samme måde; `v/cre data(3,10)`.

v/read vector filename `v/read par data.txt`
Indlæser en fil, *filename*, ind i en vektor, *vector*. Hvis vektoren ikke eksisterer vil den blive defineret.

v/fit x y ey function `v/fit xvalue yvalue yerror p3`
Fitter punkter med x-værdier, *x*, y-værdier, *y* og usikkerhed på y-værdierne *ey* med funktionen, *function*.

zone [x,y] `zone 2 3`
Opdeler vinduet til plots i undervinduer. Uden argument giver kommandoen den oprindelige opdeling (1,1) tilbage.

opt stat og *set stat* `set stat 0010`
opt stat indsætter histogrammets statestik i øverste højre hjørne af plottet, og *set stat* udvælger hvilke informationer der skal vises. I eksemplet vises kun antallet af indgange (entires) i histogrammet.

opt fit og *set fit* `set fit 1111`
opt fit indsætter fittets parametre i øverste højre hjørne af plottet, og *set fit* udvælger hvilke informationer der skal vises. I eksemplet vises samtlige parametre med deres fejl.

for/file LUN filename og *meta LUN* `for/file 99 plot.eps` og `meta 99 -113`
Gemmer plot til filen *filename*. Afsluttes med *close LUN*. Disse kommandoer giver mulighed for at gemme alt visualiseret data på en fil.

exe filename.kumac `exe fit.kumac`
Udfører kommandoerne i filen *filename*. Ved ofte brugte serier af kommandoer kan dette være tidsbesparende, ikke mindst hvis man laver mere komplicerede figurer.

shell UNIX command `shell make`
Denne kommando giver mulighed for at udføre en UNIX-kommando uden at gå ud af PAW.

Eksempel på sekvens der fitter og plotter to histogrammer i filen ***data.hbook****:*

`h/file 1 data.hbook`	Åben filen.
`opt fit`	Print parametre fra fit i plottet.
`set fit 1111`	Print parametre fra fit med fejl.
`zone 1 2`	Opdel vinduet til plots i to dele.
`h/fit 10 g`	Fit histogram med ID=10 med en Gaussisk.
`h/pl 20 lego`	Plot histogram med ID=20 i lego-stil.

Eksempel på sekvens der fitter og plotter en vektor i filen ***values.txt****:*

`v/cre data(3,20)`	Definer vektoren `data`.
`v/read data values.txt`	Hent data ind i vektoren `data`.
`v/copy data(1,1:20) x`	Kopier x-værdierne fra `data` til ny vektor, `x`.
`v/copy data(2,1:20) y`	Kopier y-værdierne fra `data` til ny vektor, `y`.
`v/copy data(3,1:20) ey`	Kopier y-usikkerhederne fra `data` til ny vektor, `ey`.
`v/fit x y ey p1`	Fit punkterne med en lineær funktion.