Developing a population PK model using Pumas

```
using AlgebraOfGraphics
    using CSV
    using CairoMakie
    using Loess
    using Pumas
    using PumasUtilities
    using Random
```

Read data

```
pkdata = CSV.read("iv_sd.csv", DataFrame, missingstring=".");
```

Coverting the DataFrame to a collection of Subjects

```
population = read_pumas(pkdata;
                        id           = :ID,
                        time         = :TIME,
                        observations = [:CONC],
                        evid         = :EVID,
                        amt          = :AMT,
                        cmt          = :CMT,
                        covariates   = [:WEIGHT, :eGFR],
                        )
```

```
Population
  Subjects: 100
  Covariates: WEIGHT, eGFR
  Observations: CONC
```

Model definition

```
iv1cmt = @model begin
  @param begin
    θcl ∈ RealDomain(lower=0.1)
    θvc ∈ RealDomain(lower=1.0)
    Ω ∈ PDiagDomain(2)
    σ_add ∈ RealDomain(lower=0.0)
    σ_prop ∈ RealDomain(lower=0.0)
```

```
  end
  @random begin
    η ~ MvNormal(Ω)
  end
  @covariates WEIGHT eGFR
  @pre begin
    CL = θcl * exp(η[1])
    Vc = θvc * exp(η[2])
  end
  @dynamics Central1
  @derived begin
    conc_model := @. Central / Vc
    CONC ~ @. Normal(conc_model, sqrt(σ_add^2 + (conc_model*σ_prop)^2))
  end
end
```

```
PumasModel
  Parameters: θcl, θvc, Ω, σ_add, σ_prop
  Random effects: η
  Covariates: WEIGHT, eGFR
  Dynamical variables: Central
  Derived: CONC
  Observed: CONC
```

Initial parameters

```
initial_est_iv1cmt = (θcl = 1.0,
                      θvc = 10.0,
                      Ω = Diagonal([0.09, 0.09]),
                      σ_add = sqrt(10.0),
                      σ_prop = sqrt(0.01),
                      )
```

```
(θcl = 1.0,
 θvc = 10.0,
 Ω = [0.09 0.0; 0.0 0.09],
 σ_add = 3.1622776601683795,
 σ_prop = 0.1,)
```

Fit base model

```
@time iv1cmt_results = fit(iv1cmt, population, initial_est_iv1cmt, Pumas
```

```
Iter     Function value   Gradient norm
     0     4.793467e+03     6.038857e+02
 * time: 3.886222839355469e-5
     1     4.452849e+03     2.701932e+02
 * time: 0.004850864410400391
     2     4.444815e+03     3.023106e+02
 * time: 0.008612871170043945
     3     4.408458e+03     2.733233e+01
 * time: 0.012093067169189453
     4     4.406848e+03     2.393277e+01
 * time: 0.015462875366210938
     5     4.403159e+03     1.912437e+01
 * time: 0.018949031829833984
     6     4.402384e+03     1.522726e+01
 * time: 0.0222928524017334
     7     4.401749e+03     4.543511e+00
 * time: 0.025703907012939453
     8     4.401666e+03     1.522852e+00
 * time: 0.028828859329223633
     9     4.401656e+03     9.513180e-01
 * time: 0.0319209098815918
    10     4.401653e+03     8.885659e-01
 * time: 0.03486990928649902
    11     4.401649e+03     7.376923e-01
 * time: 0.2514309883117676
    12     4.401644e+03     6.903798e-01
 * time: 0.2555508613586426
    13     4.401640e+03     5.160329e-01
 * time: 0.259458065032959
    14     4.401639e+03     1.943664e-01
 * time: 0.26360297203063965
    15     4.401639e+03     4.342882e-02
 * time: 0.26775693893432617
    16     4.401639e+03     1.590737e-02
 * time: 0.27225685119628906
    17     4.401639e+03     1.484691e-02
 * time: 0.2783169746398926
    18     4.401639e+03     1.470245e-02
 * time: 0.2885880470275879
    19     4.401639e+03     1.468561e-02
 * time: 0.296314001083374
    20     4.401639e+03     1.467079e-02
 * time: 0.3054800033569336
    21     4.401639e+03     1.467064e-02
 * time: 0.3150289058685303
    22     4.401639e+03     1.467049e-02
 * time: 0.32541799545288086
    23     4.401639e+03     1.467035e-02
 * time: 0.33480000495910645
    24     4.401639e+03     1.467033e-02
```

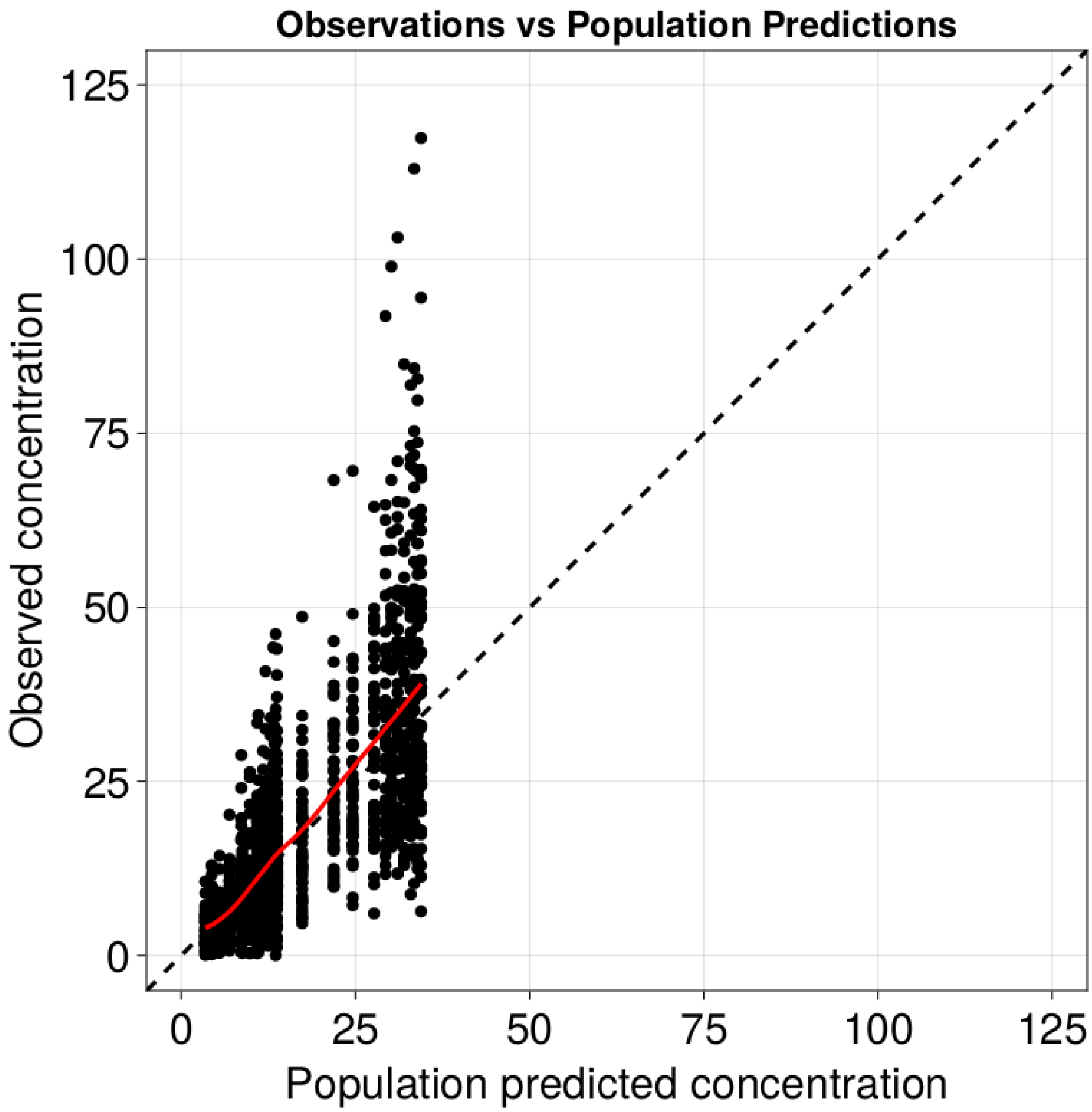
```
 * time: 0.34432005882263184
    25     4.401639e+03     1.467032e-02
 * time: 0.3501319885253906
    26     4.401639e+03     1.467031e-02
 * time: 0.3561398983001709
    27     4.401639e+03     1.467031e-02
 * time: 0.3622109889984131
    28     4.401639e+03     1.467031e-02
 * time: 0.36820292472839355
    29     4.401639e+03     1.467031e-02
 * time: 0.3742818832397461
    30     4.401639e+03     1.467031e-02
 * time: 0.38030195236206055
    31     4.401639e+03     1.467031e-02
 * time: 0.3863818645477295
    32     4.401639e+03     1.467031e-02
 * time: 0.3935699462890625
    33     4.401639e+03     1.467031e-02
 * time: 0.4007558822631836
  3.049752 seconds (9.31 M allocations: 1.013 GiB, 9.56% gc time, 86.51%
mpilation time)
FittedPumasModel

Successful minimization:                      true

Likelihood approximation:                     FOCE
Log-likelihood value:                    -4401.639
Number of subjects:                            100
Number of parameters:         Fixed      Optimized
                                  0              6
Observation records:         Active        Missing
    CONC:                      1500              0
    Total:                     1500              0

--------------------
           Estimate
--------------------
θcl         0.41706
θvc         7.1608
Ω₁,₁        0.17141
Ω₂,₂        0.19818
σ_add       2.9282
σ_prop      0.12114
--------------------
```

Inspection

```
iv1cmt_inspect = inspect(iv1cmt_results)
iv1cmt_inspect_df = DataFrame(iv1cmt_inspect)
iv1cmt_inspect_df_plot = dropmissing(iv1cmt_inspect_df, :CONC);
```

Observed vs Population predicted plot

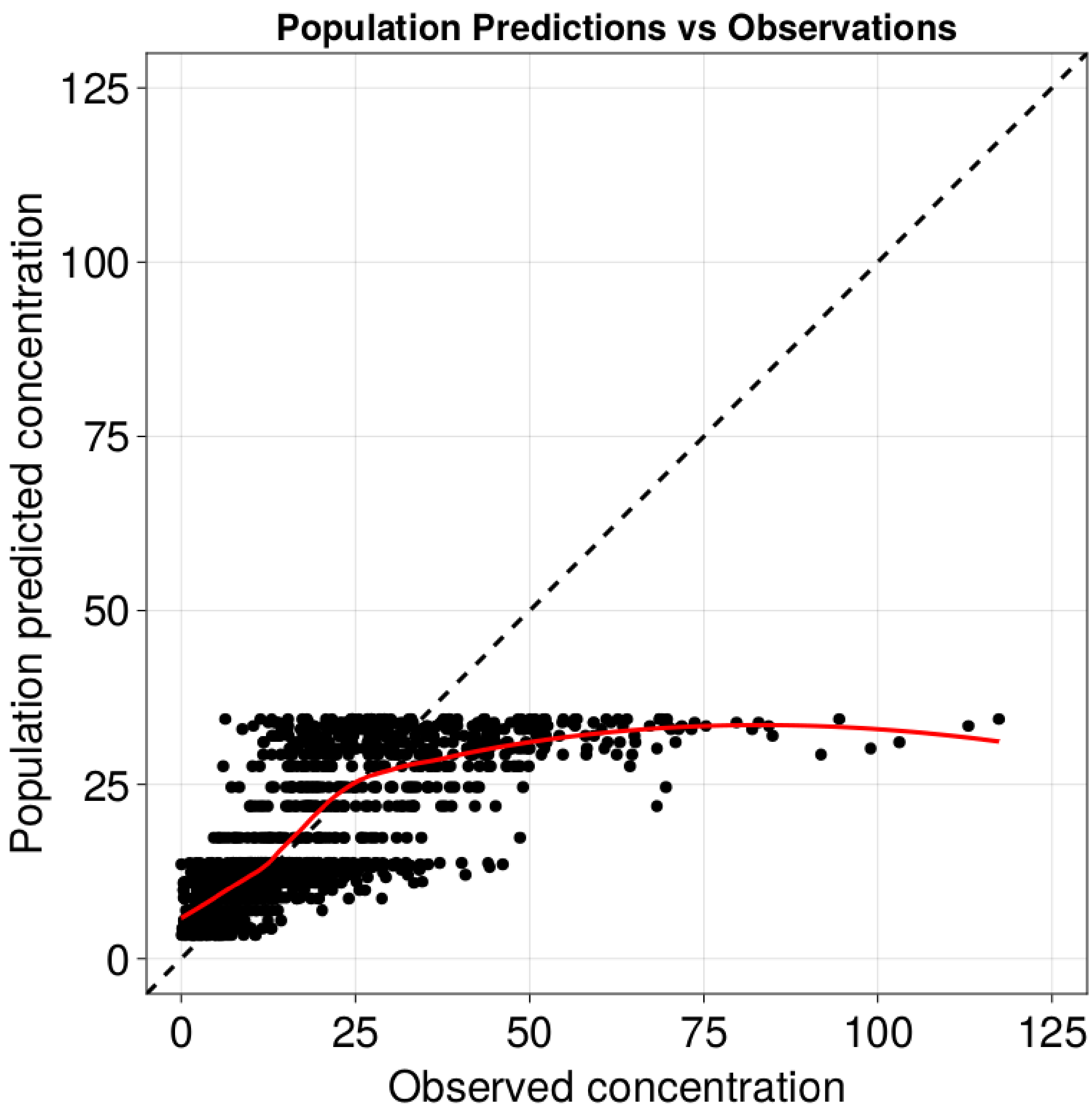
```
pred_vs_obs_loess_model = loess(Float64.(iv1cmt_inspect_df_plot.CONC_pre
                                 Float64.(iv1cmt_inspect_df_plot.CONC),

range_x_axis = range(extrema(Float64.(iv1cmt_inspect_df_plot.CONC_pred))
loess_pred = predict(pred_vs_obs_loess_model, range_x_axis)
loess_pred_df = DataFrame(x_axis = range_x_axis, y_axis = loess_pred)

pred_vs_obs_iv_sd = Figure(fontsize=30,resolution=(800,800))

pred_vs_obs_plt=
        data(iv1cmt_inspect_df_plot) *
        mapping(:CONC_pred => "Population predicted concentration",
                :CONC => "Observed concentration") *
        (visual(Scatter))

plt_ablines = mapping([0], [1]) * visual(ABLines; linewidth = 3 ,linesty


plt_loessline = data(loess_pred_df) *
                mapping(:x_axis => "Population predicted concentration",
                        :y_axis => "Observed concentration") *
                        visual(Lines; linewidth = 3, color=(:red,1.5));


draw!(pred_vs_obs_iv_sd[1,1],pred_vs_obs_plt + plt_ablines + plt_loessli
    axis=(; aspect=1,
    title=("Observations vs Population Predictions"),
    titlesize =25,xticks = 0:25:125,yticks=0:25:125,limits=((-5,130),(-5

pred_vs_obs_iv_sd
```



Population predicted vs Observed plot

```
obs_vs_pred_loess_model = loess(Float64.(iv1cmt_inspect_df_plot.CONC),
                                Float64.(iv1cmt_inspect_df_plot.CONC_pre

range_x_axis = range(extrema(Float64.(iv1cmt_inspect_df_plot.CONC))...;
loess_pred = predict(obs_vs_pred_loess_model, range_x_axis)
loess_pred_df = DataFrame(x_axis = range_x_axis, y_axis = loess_pred)

obs_vs_pred_iv_sd = Figure(fontsize=30,resolution=(800,800))

obs_vs_pred_plt=
        data(iv1cmt_inspect_df_plot) *
        mapping(:CONC => "Observed concentration",
                :CONC_pred => "Population predicted concentration") *
        (visual(Scatter))

plt_ablines = mapping([0], [1]) * visual(ABLines; linewidth = 3 ,linesty
```

```
plt_loessline = data(loess_pred_df) *
                mapping(:x_axis => "Observed concentration",
                        :y_axis => "Population predicted concentration")
                        visual(Lines; linewidth = 3, color=(:red,1.5));

draw!(obs_vs_pred_iv_sd[1,1],obs_vs_pred_plt + plt_ablines + plt_loessli
    axis=(; aspect=1,
    title=("Population Predictions vs Observations"),
    titlesize =25,xticks = 0:25:125,yticks=0:25:125,limits=((-5,130),(-5

obs_vs_pred_iv_sd
```



Diagnostic check

```
icoef_result_iv1cmt = reduce(vcat, DataFrame.(icoef(iv1cmt_results)))

app1 = evaluate_diagnostics(iv1cmt_inspect)
close(app1)
```

Save report

```
report(iv1cmt_inspect; output="reports/base_model")
```

```
note: Running TeX ...
note: Rerunning TeX because "main.out" changed ...
note: Rerunning TeX because "main.out" changed ...
note: Rerunning TeX because "main.toc" changed ...
note: Running xdvipdfmx ...
note: Writing `/tmp/jl_IZ2XR5/main.pdf` (959.78 KiB)
note: Skipped writing 5 intermediate files (use --keep-intermediates to
p them)
"reports/base_model/Report.pdf"
```

Published from iv_sd.jmd using Weave.jl v0.10.12 on 2023-08-16.