Dynamic Soccer models: checks using simulated data

Contents

Summary

This R-notebook checks whether the dynamic Skellam model has been properly implemented in Stan. It does so by simulating data from the data generating process, and checking if the fitted model parameter estimates are similar to the true parameter values used to simulate the data.

```r
rm(list = ls())
library(data.table)
library(ggplot2)
library(rstan)

rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
source("code/simulateData.R")
source("code/plot_ground_truth_vs_estimate.R")
source("code/ppc_coverage_plot.R")
source("code/MakeTimeSeriesPlot.R")
source("code/Create_model_data_for_TS2.R")
source("code/addTeamIds.R")
source("code/create_league_table.R")
source("code/MakeEPLPlotAbility.R")
source("code/games_predicted_vs_actual_intervals.R")
source("code/ppc_coverage_plot.R")
source("code/calc_rps_scores.R")
source("code/odds_to_probability.R")
source("code/ReadfitsandCalculateRPS.R")
source("code/FitOneStepAhead.R")

# create output folder
if(!dir.exists("output")) {dir.create("output")}
```

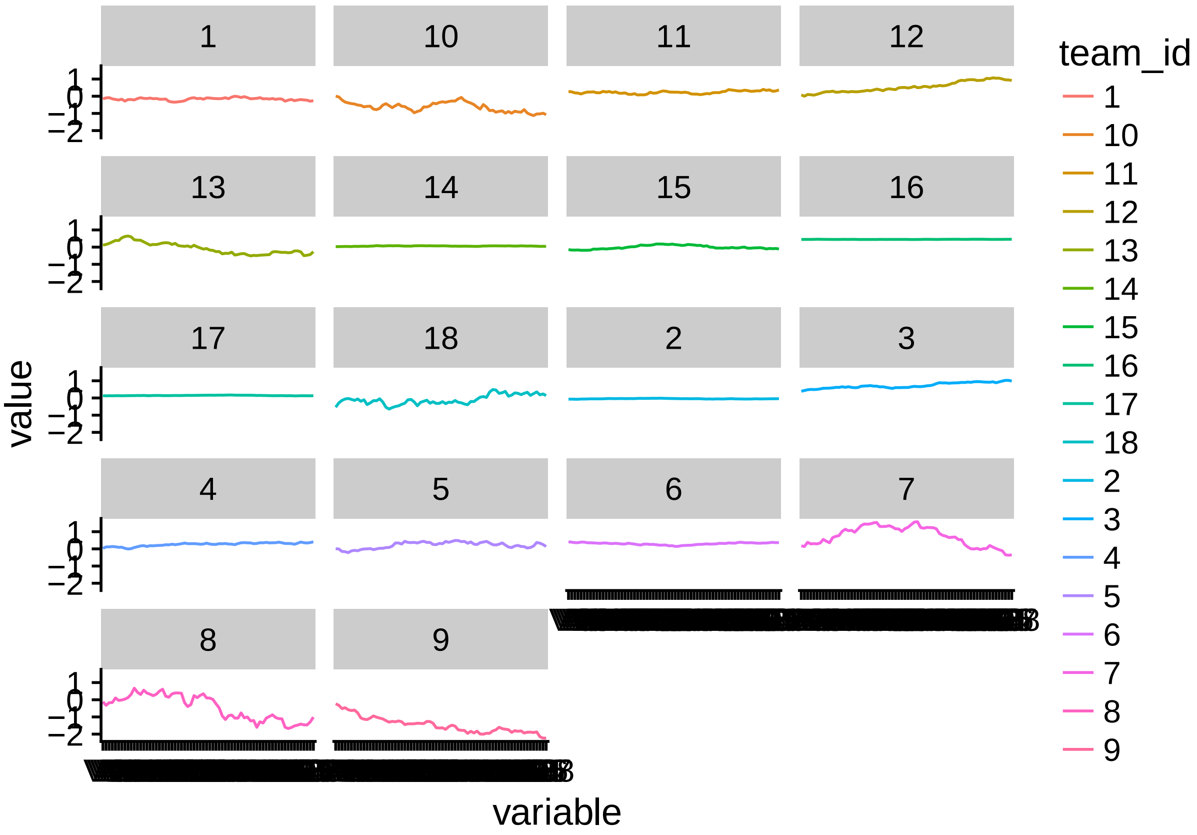
Simulate data acc to dynamic Skellam model

```
source("code/simulateData.R")
my_par_list <- list(nseasons = 2,
                 nteams = 18,
                 nrounds = 34,
                 offense_sigma = 0.25,
                 defense_sigma = 0.25,
                 mixing_proportion = 0.05, # 5% excess draws to test the zero inflation
                 mu_const = 0.2,
                 home_advantage = 0.4,
                 hyper_variance = 0.1,
                 art_turf_vec = c(rep(0, 12), rep(1, 6)), # not used
                 art_turf_advantage = 0 # not used
)


model_data <- Simulate_dynamic_ZPD_data(seed = 123, my_par_list)
```

`Simulate_dynamic_ZPD_data()` first simulates for each team two time series, one for its attack ability, and one for its defense ability. It then uses these abilities to generate match outcomes for random pairings of teams. The match outcomes are simulated according to the Skellam model (difference of two Poisson distributions), with a zero inflation component added.

plot the true (simulated) team abilities

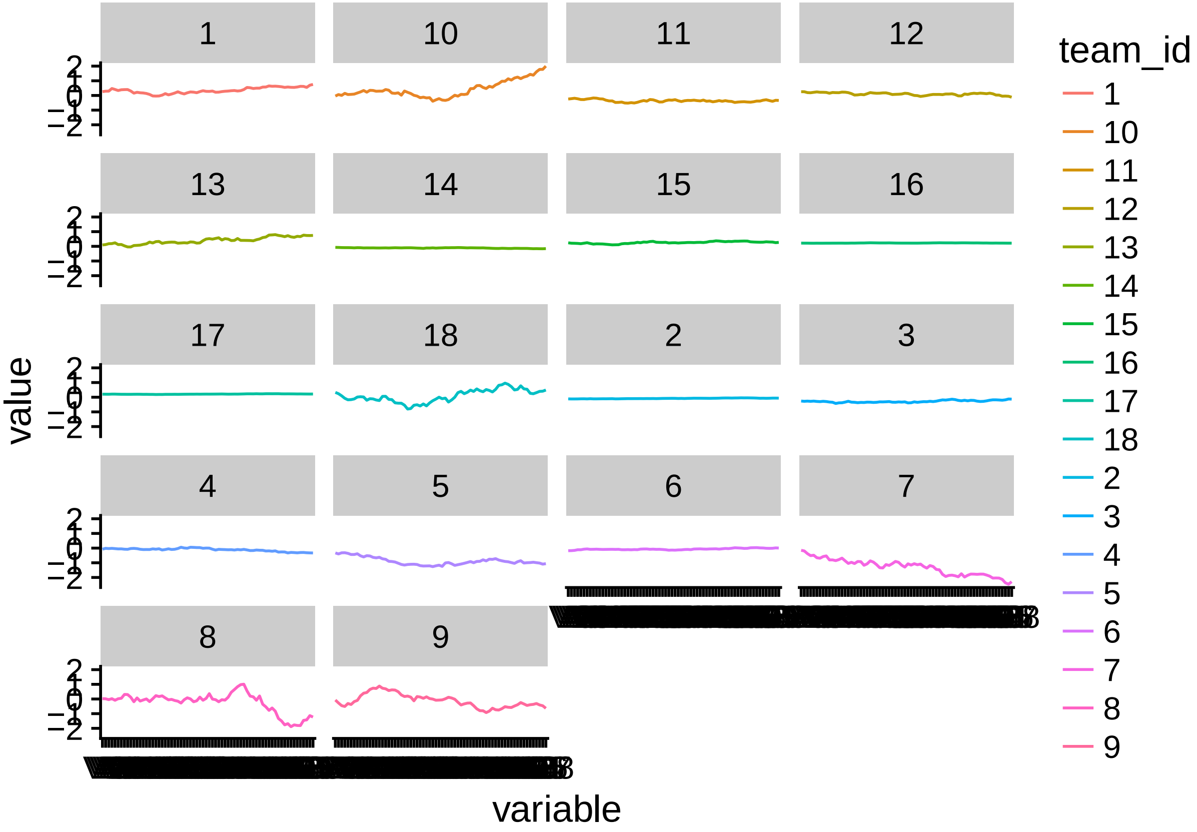
```
ma_offense <- data.table(melt(model_data$a_offense))

ggplot(ma_offense[variable %in% c(paste("V", 1:(34*2), sep = '')),], aes(x= variable, y = value,
                       group = team_id, col = team_id)) + geom_line() + facet_wrap(~ team_id, ncol = 4)
```



```r
ma_defense <- data.table(melt(model_data$a_defense))
```

```
## Using team_id as id variables
```

```r
ggplot(ma_defense[variable %in% c(paste("V", 1:(34*2), sep = '')),], aes(x= variable, y = value,
                                      group = team_id, col = team_id)) + geom_line() + facet_wrap(~ team_id, ncol = 4)
```



Fit dynamic Skellam model on simulated data

```
fullrun <- 0

if(fullrun) {
stanfit_sim_skellam <- stan(
  file = "models/skellam_dynamic.stan",
  data = model_data,
  chains = 4,
  warmup = 200,
  init_r = 0.1, # instead of 2
  iter = 500,
  cores = 4,
  control = list(adapt_delta = 0.95)
  )
  saveRDS(stanfit_sim_skellam, "FITS/skellam_dynamic_sim.rds")
}
```

Analyse results

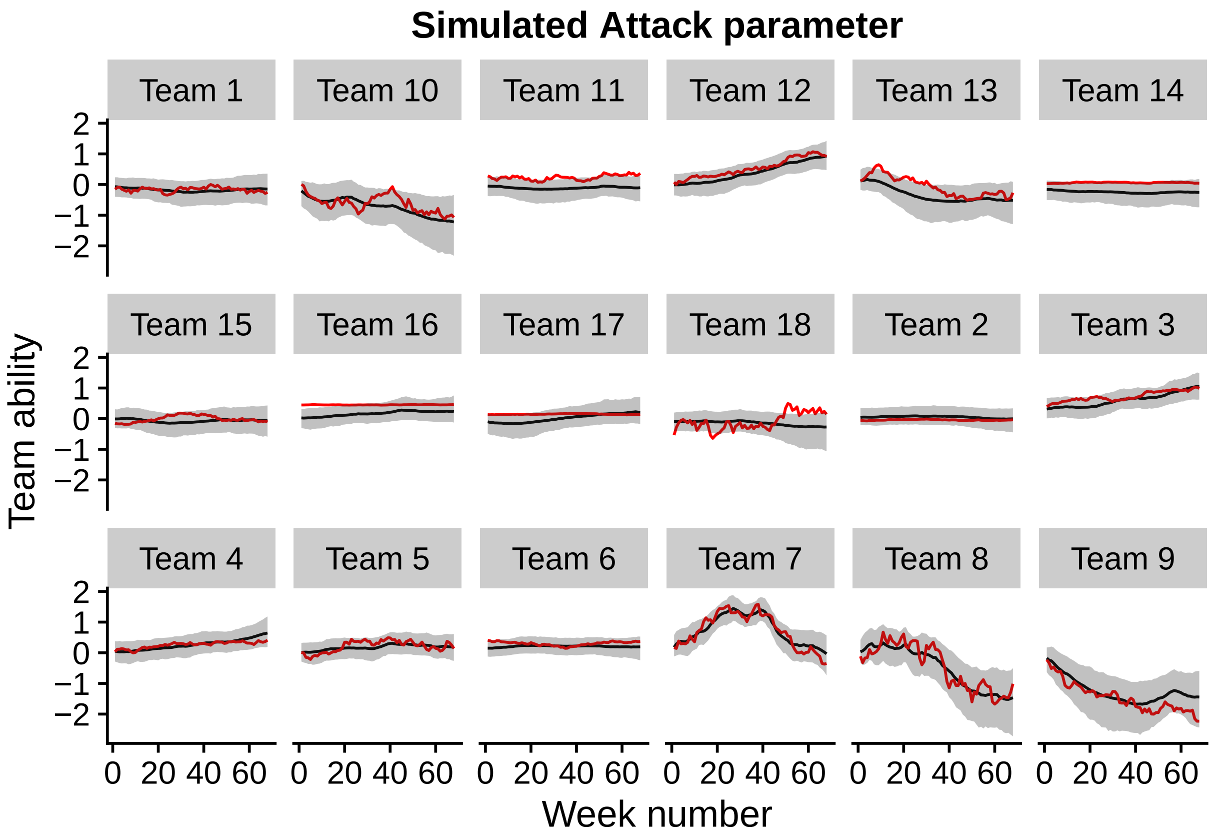
```
stanfit_sim_skellam <- readRDS("FITS/skellam_dynamic_sim.rds")
print(stanfit_sim_skellam, c("constant_mu", "home_advantage", "mixing_proportion"))
```

```
## Inference for Stan model: skellam_dynamic.
## 4 chains, each with iter=500; warmup=200; thin=1;
## post-warmup draws per chain=300, total post-warmup draws=1200.
##
##                   mean se_mean   sd  2.5%  25%  50%  75% 97.5% n_eff Rhat
## constant_mu       0.14    0.01 0.15 -0.17 0.05 0.15 0.24  0.41   861    1
## home_advantage    0.43    0.00 0.05  0.33 0.40 0.43 0.47  0.54  1200    1
## mixing_proportion 0.02    0.00 0.01  0.00 0.01 0.02 0.03  0.05  1200    1
##
## Samples were drawn using NUTS(diag_e) at Wed Jul 18 15:55:31 2018.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

Compare the estimated team abilities with the true team abilities

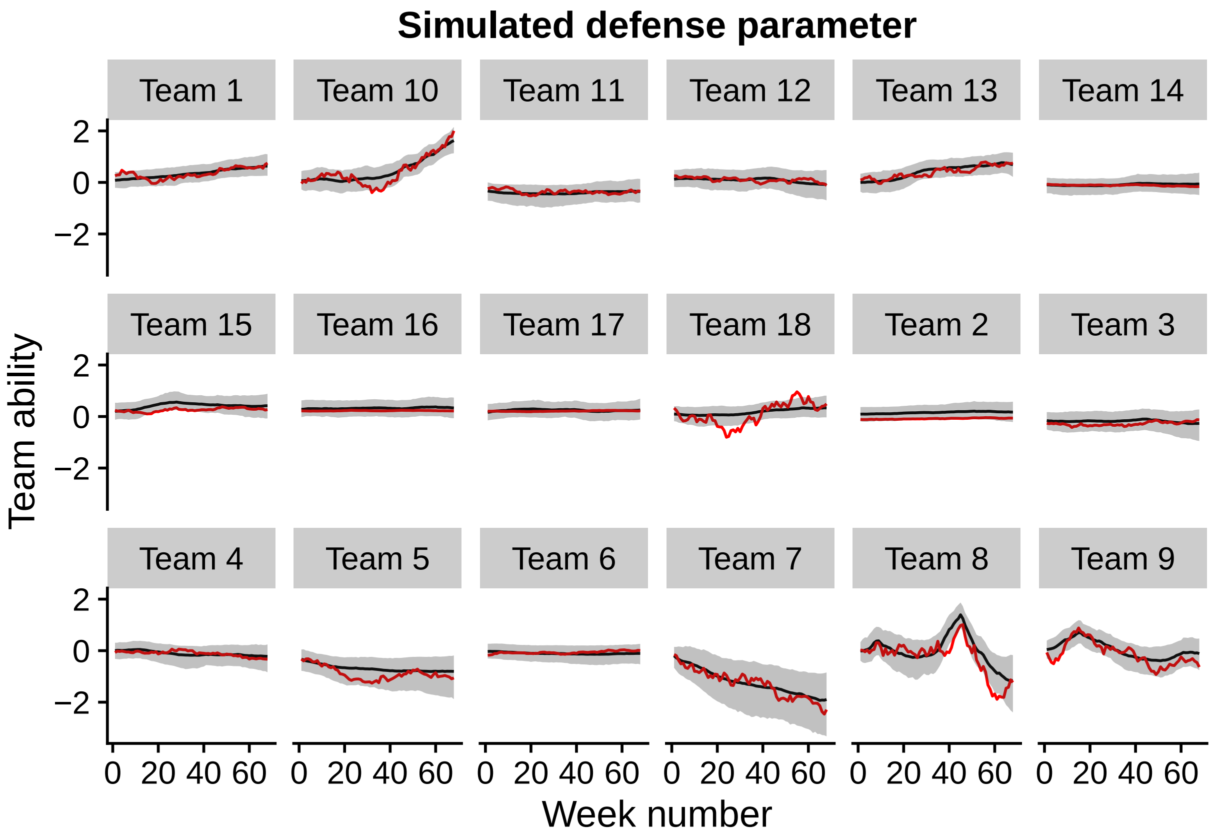
```
sims <- extract(stanfit_sim_skellam)

a_sims <- sims$a_offense

id_lut <- model_data$id_lut

MakeTimeSeriesPlotSim(a_sims, model_data$a_offense, id_lut, title = "Simulated Attack parameter")
```

```
## Using team_id as id variables
```



```
source("code/MakeTimeSeriesPlot.R")
sims <- extract(stanfit_sim_skellam)

a_sims <- sims$a_defense

id_lut <- model_data$id_lut

MakeTimeSeriesPlotSim(a_sims, model_data$a_defense, id_lut, title = "Simulated defense parameter")
```

```
## Using team_id as id variables
```


Simulated defense parameter
Team 1
Team 10
Team 11
Team 12
Team 13
Team 14
Team 15
Team 16
Team 17
Team 18
Team 2
Team 3
Team 4
Team 5
Team 6
Team 7
Team 8
Team 9
2
0
−2
Team ability
0 20 40 60
Week number