A State-Space Perspective on Modelling and Inference for Online Skill Rating

(published at JRSS-C, package abile on GitHub)

Sam Power, SigProC Lab Seminar, Thursday 7 November, 2024

(joint work with collaborators)

Sam Duffield
(Normal Computing)

Lorenzo Rimella
(University of Torino)

$\boxed{\text{feel free to stop me at any point}}$

Overview

- Skill Rating in Competitive Sports
- State-Space Models
- Inference Tasks for State-Space Models
- Inference Algorithms for State-Space Models
- Applications to Real Data

The Skill Rating Problem

Prediction in Competitive Sports

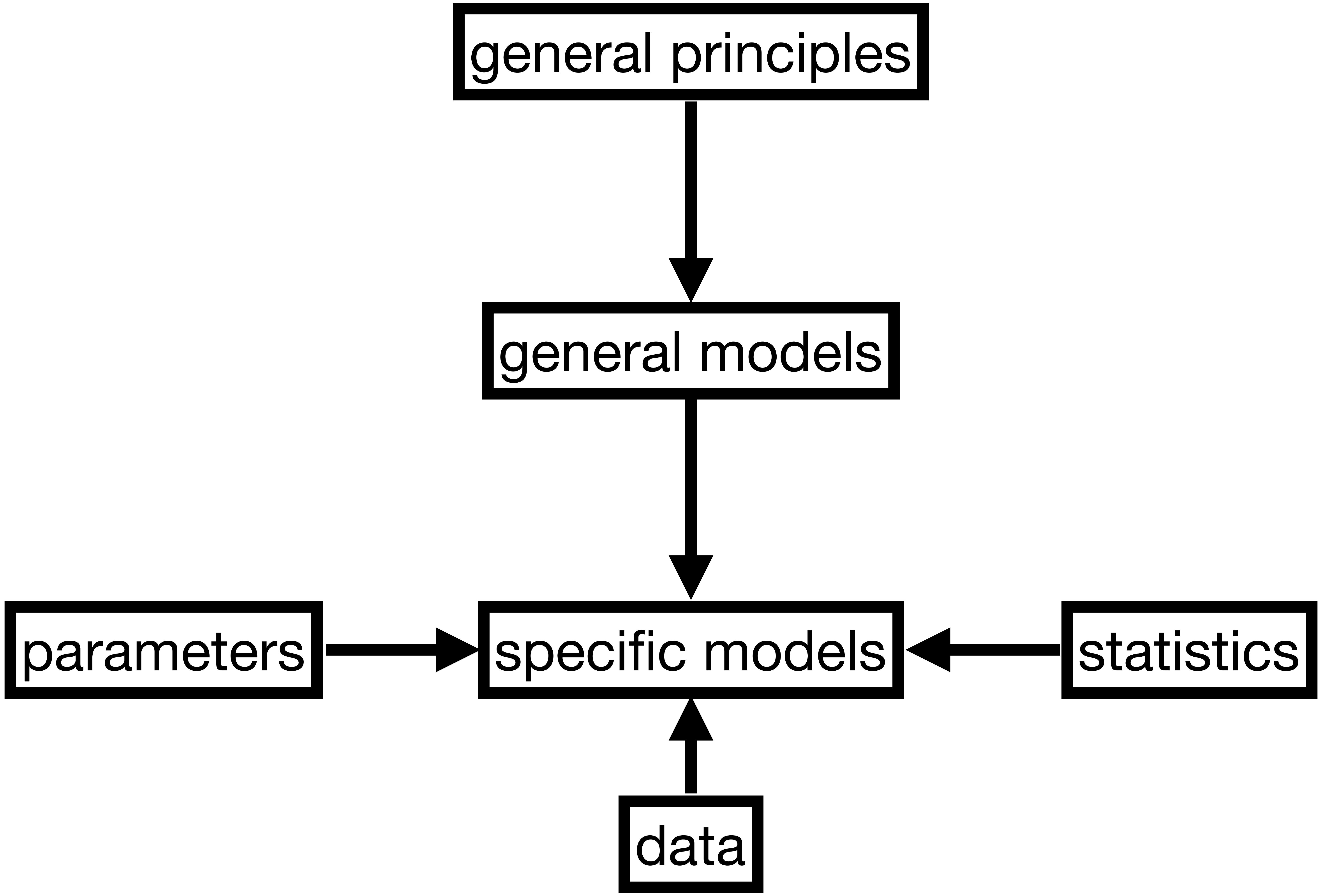
- ‘sports’ $\supseteq$ { ‘players’, ‘matches’, ‘results’ }
  - $\ni$ { tennis, football, basketball, chess, online gaming, education apps, … }
- basic task: observe past results, predict future results
- refined task: infer ‘skills’ of ‘players’
  - applications to e.g. { seeding, team matchups, evaluating interventions, … }

A Non-Mathematical Observation

- broad interest, even from a <u>non-mathematical</u> audience
- approaches can be …
    - ‘non-mathematical’,
    - mathematical, ‘non-statistical’ / ‘quasi-statistical’,
    - ‘fully-statistical’.
- important: what are your goals?

Mathematical and Statistical Approaches

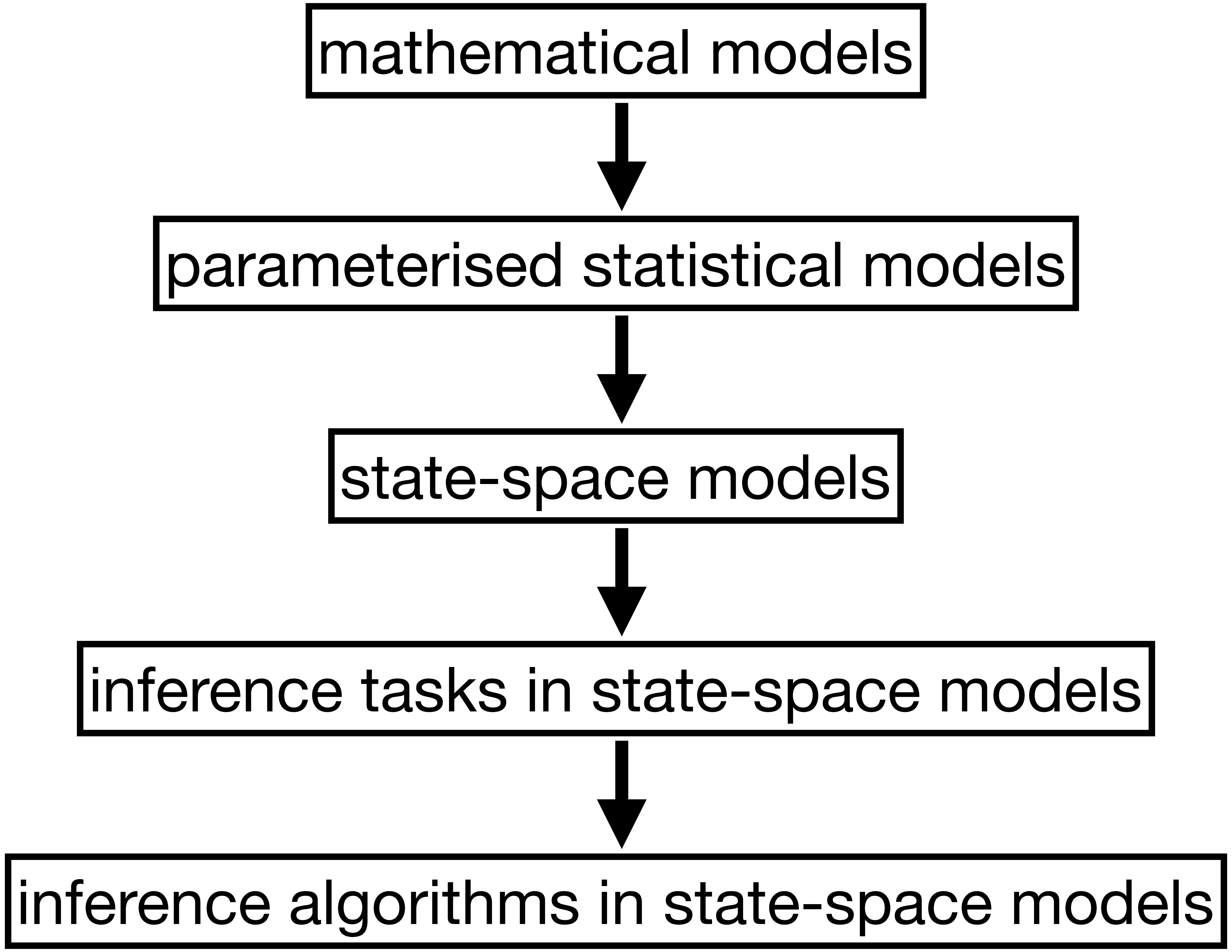
- models are devices, to *use*, to *critique*, and to *refine*
- mathematical models facilitate extrapolation, extension
- general (sporting) principles can yield general (skill) models
- specific (sporting) problems should have specific features
- with statistical methods, we can calibrate general models to specific sports
- statistical formulations facilitate treatment of uncertainty


general principles
general models
parameters
specific models
statistics
data

Our Approach to Skill Rating

- general, structured mathematical models for the skill rating problem
- equip mathematical models with interpretable statistical parameters
- assess inference objectives within model class
- develop algorithmic strategies for solving these tasks

- focus on high-level modelling framework, facilitate a generic workflow
- limited commitment to low-level details of specific models.


mathematical models
parameterised statistical models
state-space models
inference tasks in state-space models
inference algorithms in state-space models

Latent Variable Models

Warm-up: Latent Variable Models

- given two players of a sport, what influences their match results?
  - a first-order answer: their 'skill' at the sport
  - mathematically: let player i have skill $x^i \in \mathscr{X}$
- simple model: $\mathbf{P}(\text{player } i \text{ beats player } j) = F\left(x^i, x^j; \theta\right)$

State-Space Models

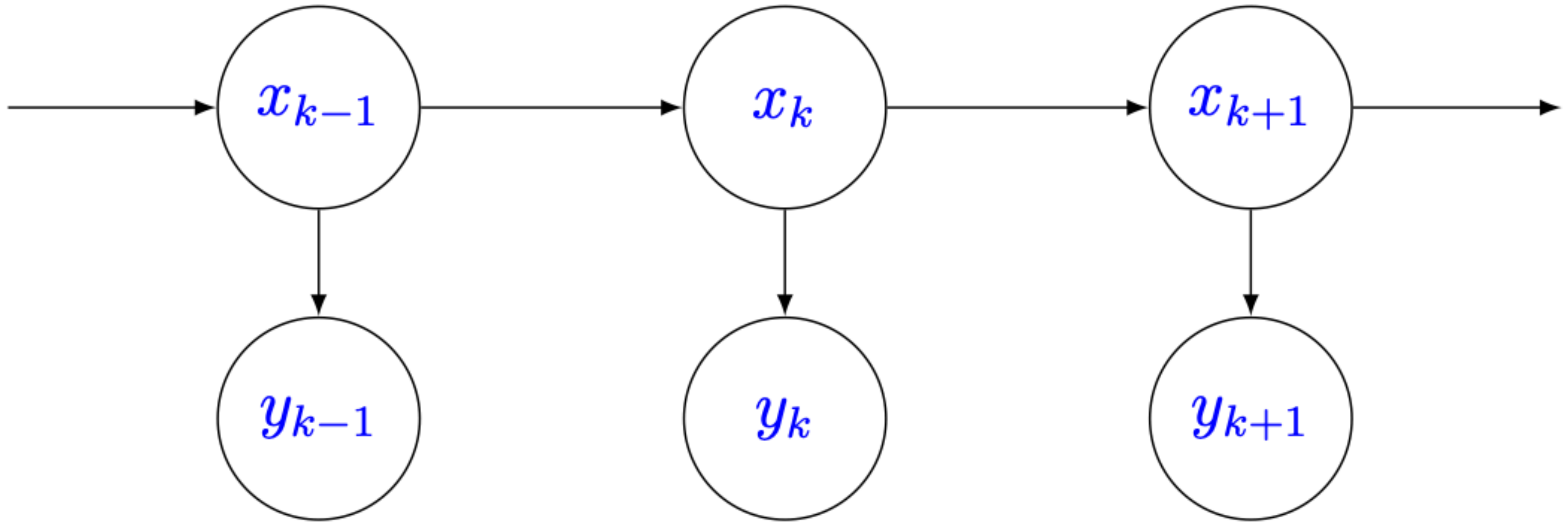
Latent Variable Models through Time

- question: should a player's skill level be static in time?
  - basic answer: 'probably not!'
  - principled answer: 'write down a model, then let the data decide'
    - empirically: indeed often worthwhile for skills to vary over time
- simplest choice: player skills evolve as a *Markov chain* in time
  - ⇝"State Space Models"

State-Space Models in One Slide

$$p(x) = \mu_0\left(x_0\right) \cdot \prod_k M_{k-1,k}\left(x_{k-1}, x_k\right)$$

$$p\left(y \mid x\right) = \prod_k G_k\left(x_k, y_k\right)$$



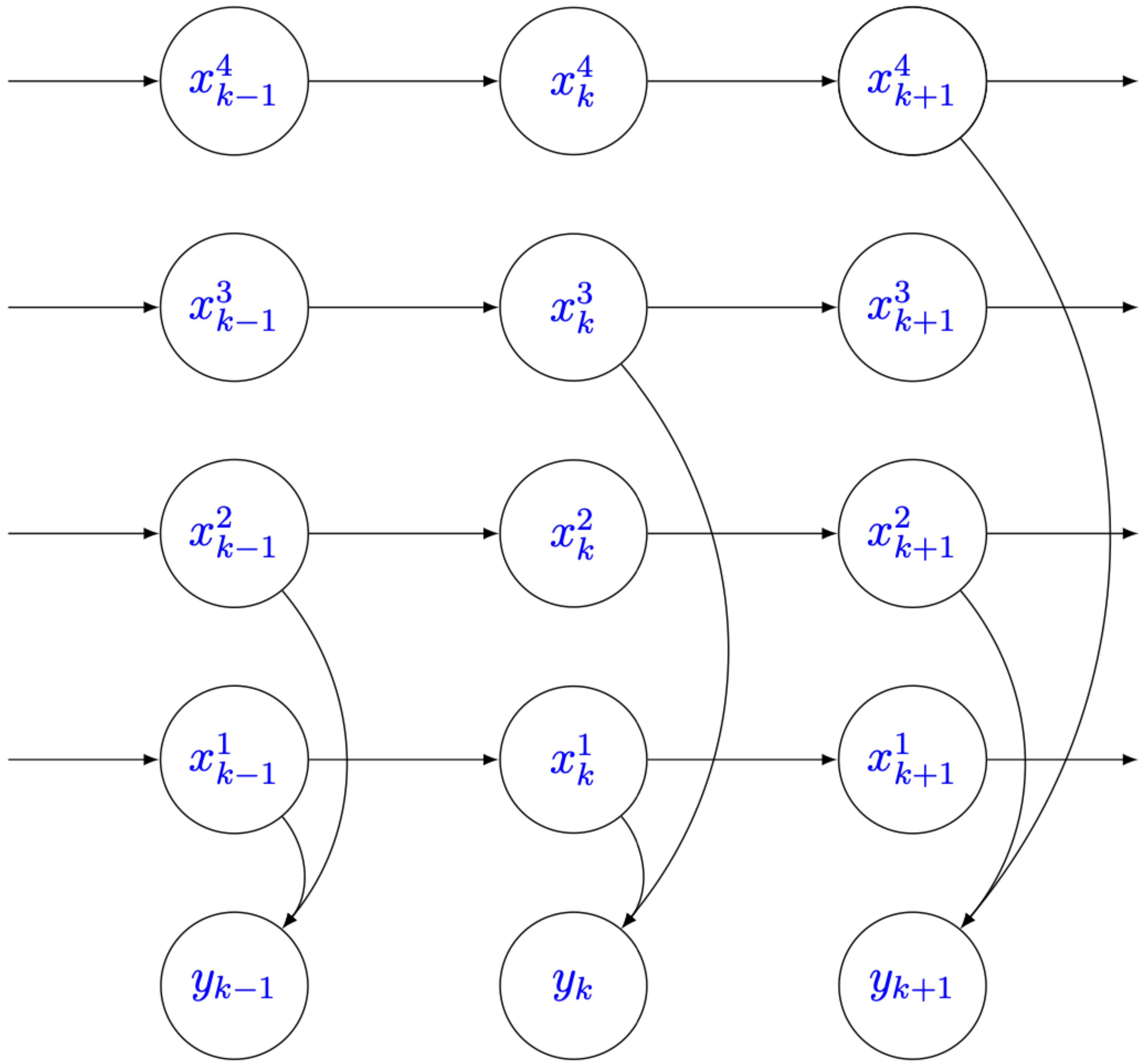
Factorial State-Space Models

- for us, x is really $\left\{x^i : i \in [N]\right\}$; N can be quite large
  - curse: high dimensionality makes SSMs very difficult
    - antidote: players only interact during matches
      - $\rightsquigarrow$ model player skills as evolving independently
        - $\rightsquigarrow$"Factorial" State Space Models
- note also: observation model is *sparse* w.r.t players

Factorial State-Space Models in One Slide

$$p(x) = \prod_i \left(\mu_0^i \left(x_0^i\right) \cdot \prod_k M_{k-1,k}^i \left(x_{k-1}^i, x_k^i\right) \right)$$

$$p\left(y \mid x\right) = \prod_k G_k \left(x_k, y_k\right)$$



Some Concrete Choices

Dynamical Models

- $\mathscr{X} = \mathbf{R}$: can take $M \in$ { Brownian motion, OU Process }

$$M_{s,t}^{\mathsf{BM}}(x, x') = \mathcal{N}\left(x' \mid x, \sigma^2 \cdot (t-s)\right)$$

$$M_{s,t}^{\mathsf{OU}}(x, x') = \mathcal{N}\left(x' \mid \mathrm{e}^{-\gamma(t-s)} \cdot x, \sigma^2 \cdot \left(1 - \mathrm{e}^{-2\gamma(t-s)}\right)\right)$$

- $\mathscr{X} = \left[S\right]$: can take M = Reflected Random Walk, with jump rates

$$1 \leqslant x < S \implies \lambda(x, x+1) = \lambda_0$$

$$1 < x \leqslant S \implies \lambda(x, x-1) = \lambda_0$$

Some Concrete Choices

Observation Models

- $\mathscr{X} = \mathbf{R}$, $\mathscr{Y} = \{\text{home win, away win}\}$: can take

$$\mathbf{P}\left(y = \mathrm{h} \mid x^h, x^a\right) = \sigma\left(x^h - x^a\right), \text{ with } \sigma \in \left\{\mathbf{logit}, \mathbf{probit}, \cdots\right\}$$

- $\mathscr{X} = \left[S\right]$: can do the same, or parametrise directly

- straightforward extension to $\mathscr{Y} = \{\text{home win, away win, draw}\}$, etc.

Inference in State-Space Models

Inference in State-Space Models

- back to ‘real’ tasks:
  1. predict, in real time, the outcome of current matches
     - $\rightsquigarrow$ need estimates of $p\left(x_k \mid y_{\leqslant k}\right)$ (“filtering”)
  2. evaluate past performance of players
     - $\rightsquigarrow$ need estimates of $p\left(x_k \mid y_{\leqslant K}\right)$ (“smoothing”)
  3. calibrate parameters of general model to specific sports
     - $\rightsquigarrow$ need estimates of $p\left(y_{\leqslant K} \mid \theta\right)$ (“likelihood estimation”, “parameter estimation”)

Feedback Loops in State-Space Models

- even if only one of these tasks is of applied interest, all three are intertwined
  - good filtering requires good parameter estimation
  - good parameter estimation requires good smoothing
  - good smoothing requires good filtering
- takeaway: in many cases, aim to do all three tasks well

Inference in Factorial State-Space Models

- high dimension $\rightsquigarrow$ hard to even *represent* full tracking distributions
- practically: often sufficient to only track skills of individual players
  - computationally feasible
  - incurs some (controllable) bias

Algorithms for State-Space Models

Filtering

- object of interest: $\mathrm{Filter}_k = \mathbf{P}\left(x_k \mid y_{1:k}\right)$
- for streamlined computation, rely on key abstract recursions

$$\mathrm{Predict}_{k|k-1} = \text{Propagate}\left(\mathrm{Filter}_{k-1}; M_{k-1,k}\right)$$

$$\mathrm{Filter}_k = \text{Assimilate}\left(\mathrm{Predict}_{k|k-1}; G_k\right)$$

- most filters (exact or approximate) are based around these recursions

Smoothing

- object of interest: $\mathrm{Smooth}_{k|K} = \mathbf{P}\left(x_k \mid y_{1:K}\right)$
- for streamlined computation, rely on key abstract recursions

$$\mathrm{Smooth}_{k,k+1|K} = \mathsf{Bridge}\left(\mathrm{Filter}_k, \mathrm{Smooth}_{k+1|K}; M_{k,k+1}\right)$$

$$\mathrm{Smooth}_{k|K} = \mathsf{Marginalise}\left(\mathrm{Smooth}_{k,k+1|K}; k\right)$$

- most smoothers (exact or approximate) are based around these recursions

Parameter Estimation

- object of interest: $\mathbf{P}\left(y_{1:K} \mid \theta\right)$
- often not analytically available
- common, generic strategy for latent variable models: EM algorithm

$$\log \mathbf{P}\left(y \mid \theta\right) = \sup \left\{ \mathscr{F}\left(Q, \theta\right) := \mathbf{E}_Q \left[\log \left(\frac{\mathbf{P}\left(x, y \mid \theta\right)}{Q(x)} \right) \right] : Q \in \mathscr{P}\left(\mathscr{X}\right) \right\}$$

  - alternating maximisation of $\mathscr{F}$ w.r.t. (Q, θ)
  - optimal Q is $\mathbf{P}\left(x \mid y, \theta\right)$, i.e. smoothing distribution in SSMs

Coping with Scale

! Terminology Warning !

- back to discussing the skill rating problem:
  - $t \in [0,T]$ denotes a generic time
  - $k \in [K]$ denotes a match-time, corresponding to time $t = t_k$
  - we only monitor skill levels on match-times
  - we write x_k^i for what is in some sense ‘technically’ $x_{t_k}^i$, etc.
  - t, T will largely be suppressed in favour of k, K

State of Play: Scalability

- for several interesting sporting applications, one has
  1. many players ($N \to \infty$).
  2. many matches ($K \to \infty$).
  3. high-frequency matches.
- hence, we focus on methods which
  1. can be implemented *online*, and
  2. whose computational complexity scales *linearly* with both N and K.
  - (realistic and worthwhile)

Decoupling Approximation

- for tracking players' skills, every method under discussion approximates

$$\mathrm{Filter}_k \approx \prod_{i \in [N]} \mathrm{Filter}_k^i$$

$$\mathrm{Smooth}_k \approx \prod_{i \in [N]} \mathrm{Smooth}_k^i$$

- under weak dependence, this is provably sensible
- computationally, this approximation opens many doors

Match Sparsity and Parallelism

- observation: at any given time, any player can play in at most one match
- observation: any match involves at most two players
- consequence:
  - upon receiving the result of a single match,
  - update our filtering distribution only for the two players who were involved

Match Sparsity and Parallelism

- consequence:
  - upon receiving the results of several matches, involving disjoint pairs of players,
    - update our filtering distributions only for those pairs of players,
      - and do so in parallel
- similar economies are available when computing smoothing distributions

$$\text{Filter}_k = \mathsf{Assimilate}\left(\text{Predict}_{k|k-1}; G_k\right)$$

$$\text{Predict}_{k|k-1} = \prod_{i \in [N]} \text{Predict}^i_{k|k-1}$$

$$G_k\left(x_k, y_k\right) = \prod_{(h,a) \in \mathsf{Opp}(k)} G_k^{h,a}\left(x_k^h, x_k^a, y_k^{h,a}\right)$$

$$\text{for } \left(h, a\right) \in \mathsf{Opp}\left(k\right), \quad \text{Filter}_k^{h,a} = \mathsf{Assimilate}\left(\text{Predict}^{h,a}_{k|k-1}; G_k^{h,a}\right)$$

Assimilating the Result of one Match

upon receiving the result of a match at time t involving players (h, a):

1. compute the times at which these two players each last played
2. retrieve the filtering distributions of the two players' skills
3. compute the current predictive distributions of the two players' skills
4. compute the joint filtering distribution of the two players' skills
5. compute the marginal filtering distributions of the two players' skills

Algorithms for Online Skill Rating

Algorithms for Skill Rating

- there are many algorithms for treating this skill ranking problem
- i present some here, in (subjectively!) ~increasing order of statistical sophistication
- their practical performance will be addressed in the experiments section

Elo
(online stochastic gradient)

- very widely-used (most famously in chess)
- incomplete model: $\mathscr{X} = \mathbf{R}, \mathbf{P}\left(y = \mathrm{h} \mid x^h, x^a\right) = \mathbf{logit}\left(x^h - x^a\right)$
- directly increment skill estimates via

$$x^h \leftarrow x^h + K \cdot \left(\mathbb{1}\left[y_k = \mathrm{h}\right] - \mathbf{logit}\left(x^h - x^a\right)\right)$$
$$x^a \leftarrow x^a + K \cdot \left(\mathbb{1}\left[y_k = \mathrm{a}\right] - \mathbf{logit}\left(x^a - x^h\right)\right)$$

- intuition: compare outcome to predicted outcome, increment skills accordingly

Glicko
(extended Kalman filter)

- $\mathscr{X} = \mathbf{R}$, Gaussian tracking distributions
- $M_{s,t}(x, x') = \mathcal{N}\left(x' \mid x, \sigma^2 \cdot (t - s)\right)$, $G\left(y = \mathrm{h} \mid x^h, x^a\right) = \mathbf{logit}\left(x^h - x^a\right)$
- Propagate step is closed-form by Gaussian conjugacy
- Assimilate step is approximated via Taylor expansion of observation model

TrueSkill (through time)
(expectation propagation / moment matching)

- $\mathcal{X} = \mathbf{R}$, Gaussian tracking distributions
- $M_{s,t}(x, x') = \mathcal{N}\left(x' \mid x, \sigma^2 \cdot (t-s)\right)$, $G\left(y = \mathrm{h} \mid x^h, x^a\right) = \mathbf{probit}\left(x^h - x^a\right)$
- Propagate step is closed-form by Gaussian conjugacy
- Assimilate step is approximated via moment-matching step

Local Sequential Monte Carlo
(stochastic particle methods)

- idea: represent tracking laws by adaptive system of J stochastic particles
- $\mathscr{X}$ generic, $M_{s,t}$ generic (simulable), G_t generic (evaluable)
- Propagate step is treated by simulation.
- Assimilate step is treated by importance resampling.

Graph Filter-Smoother
(finite state-space recursions)

- $\mathscr{X} = [S]$, discrete tracking distributions
- $M_{s,t}$ from continuous-time Markov process, G_t generic
- Propagate step is closed-form (matexp, matmul)
- (joint) Assimilate step is closed-form (element-wise product)
- no systematic bias beyond decoupling approximation

Table 1: Considered approaches and their features. All approaches are linear in the number of players $\mathcal{O}(N)$ and the number of matches $\mathcal{O}(K)$.

Method	Skills	Filtering	Smoothing	Parameter Estimation	Sources of Error (Beyond Factorial)
Elo	Continuous	Location , $\mathcal{O}(1)$	N/A	N/A	Not model-based
Glicko	Continuous	Location and Spread, $\mathcal{O}(1)$	Location and Spread, $\mathcal{O}(1)$	N/A	Not model-based
Extended Kalman	Continuous	Location and Spread, $\mathcal{O}(1)$	Location and Spread, $\mathcal{O}(1)$	EM	Gaussian Approximation
TrueSkill2	Continuous	Location and Spread, $\mathcal{O}(1)$	Location and Spread, $\mathcal{O}(1)$	EM	Gaussian Approximation
SMC	General	Full Distribution, $\mathcal{O}(J)$	Full Distribution, $\mathcal{O}(J)$ [2]	EM	Monte Carlo Variance
Discrete	Discrete	Full Distribution, $\mathcal{O}(S^2)$	Full Distribution, $\mathcal{O}(S^2)$ [3]	(Gradient) EM	N/A

Applications

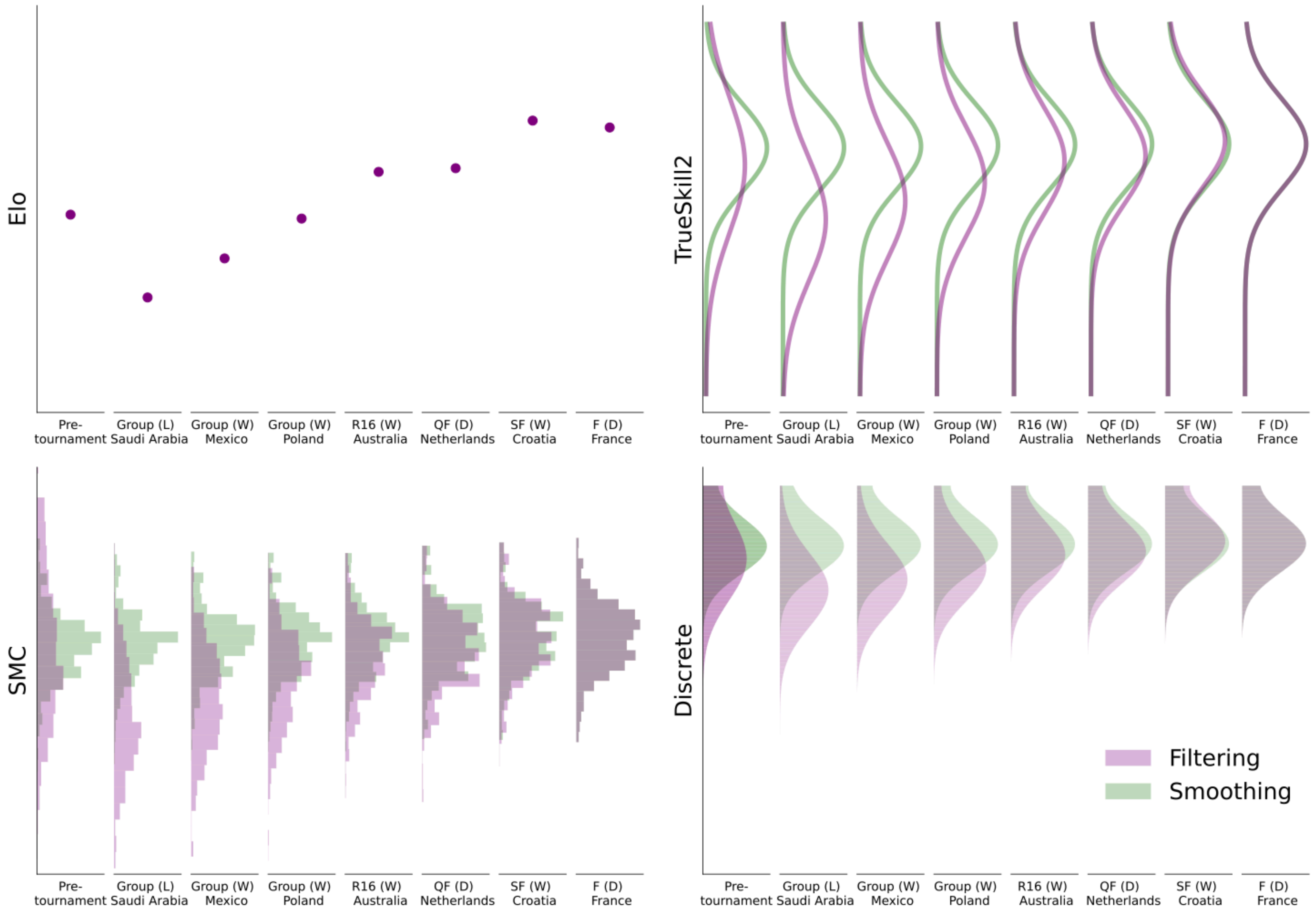
Goal of Case Studies

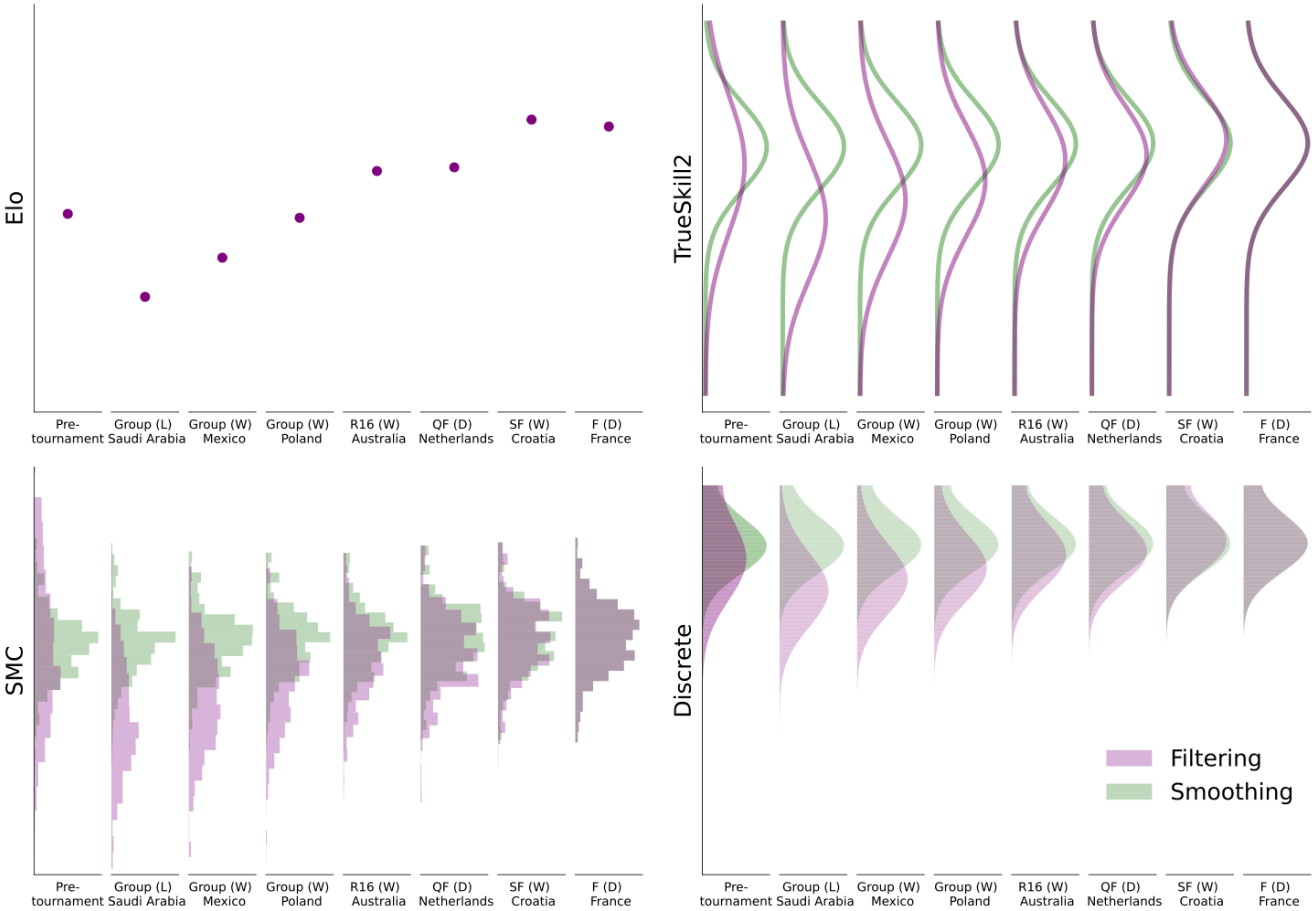
- “replicate a realistic workflow”
  - evaluating different models, quantitative and qualitative comparison
    - filtering and smoothing with static hyperparameters
    - parameter estimation from historical data
    - filtering and smoothing for online prediction and retrospective evaluation
- broad aim: separate modeling concerns from inference concerns
- python package with experiments: github.com/SamDuffield/abile

Exploratory Analysis

(Football, Argentina National Team, 2020-2023 WC)

- observe different skill representations, uncertainty quantification
- confirming intuitions: influence of { wins, draws, losses, surprise losses }
- stabilisation of smoothing distribution, reduction of uncertainty





Elo
TrueSkill2
SMC
Discrete
Pre-tournament
Group (L) Saudi Arabia
Group (W) Mexico
Group (W) Poland
R16 (W) Australia
QF (D) Netherlands
SF (W) Croatia
F (D) France
Filtering
Smoothing

WTA Tennis
(Women's, 2019-2022)

- visualisation of estimate of log marginal likelihood
- EM iterations converge on same basins
- bias from Gaussian approximation leads to distorted trajectory
- Less systematic bias for SMC, discrete approach

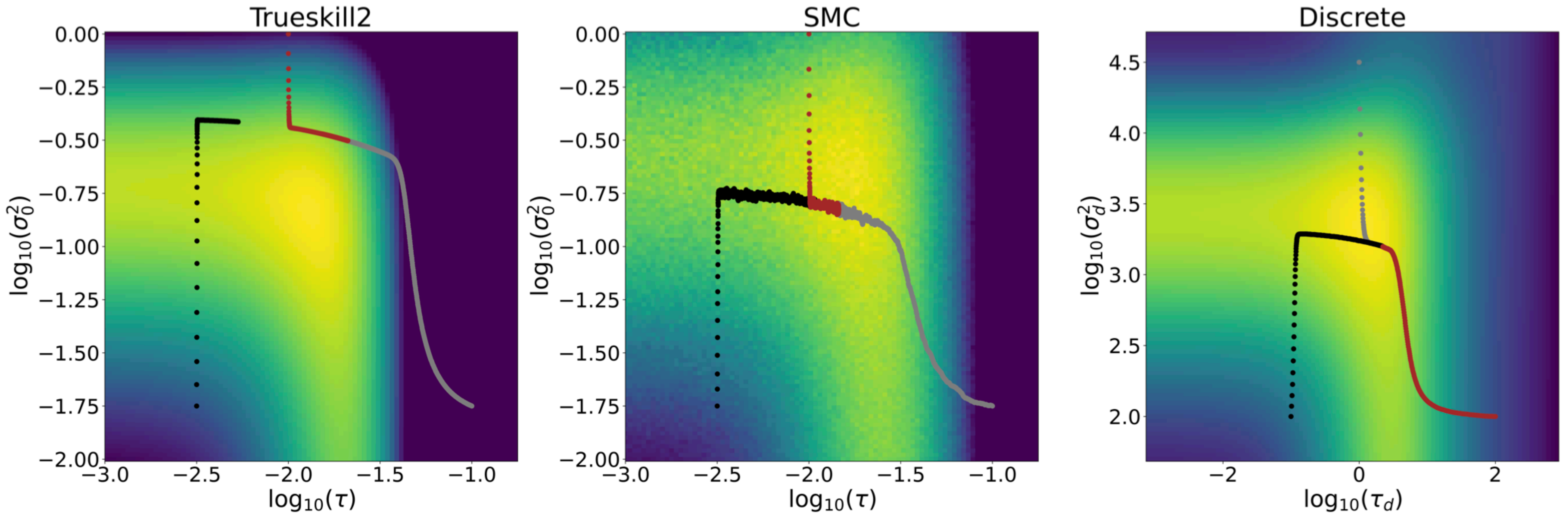


Figure 3: Log-likelihood grid and parameter estimation for WTA tennis data. Note that TrueSkill2 and SMC share the same model.

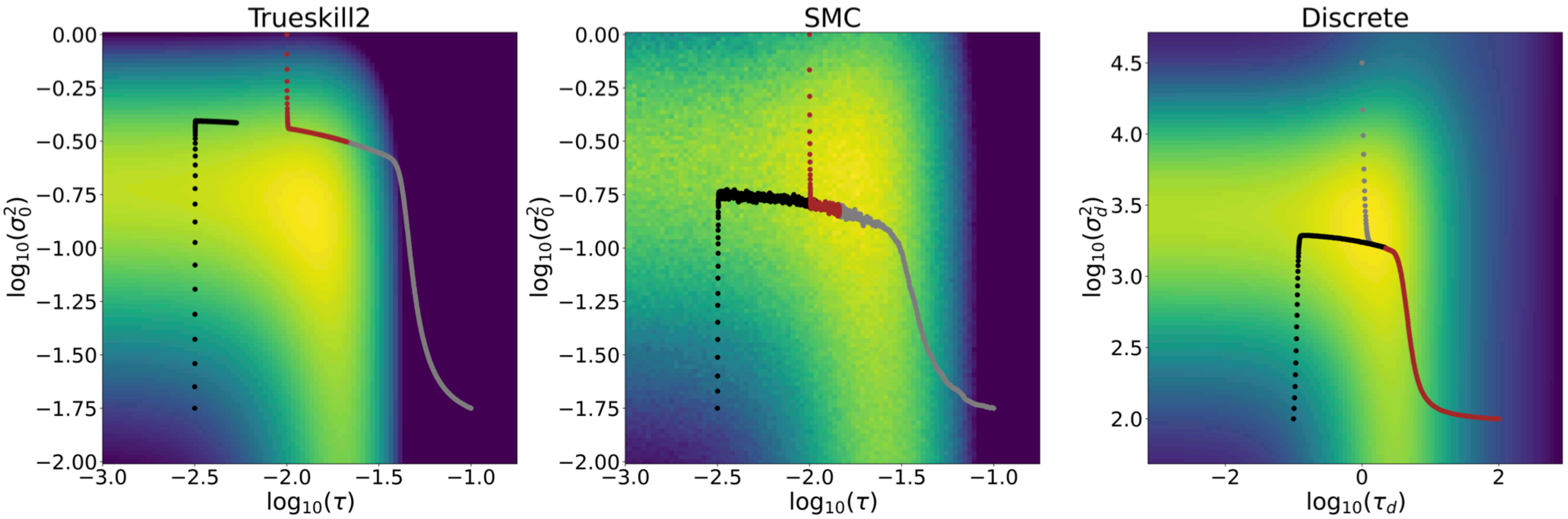


Figure 3: Log-likelihood grid and parameter estimation for WTA tennis data. Note that TrueSkill2 and SMC share the same model.

EPL Football
(Tottenham, 2011-2023)

- use smoothing laws to retrospectively evaluate impact of managers
- naturally, smoothing is less reactive than filtering
- story is roughly consistent across model-based approaches
- harder to address with e.g. Elo

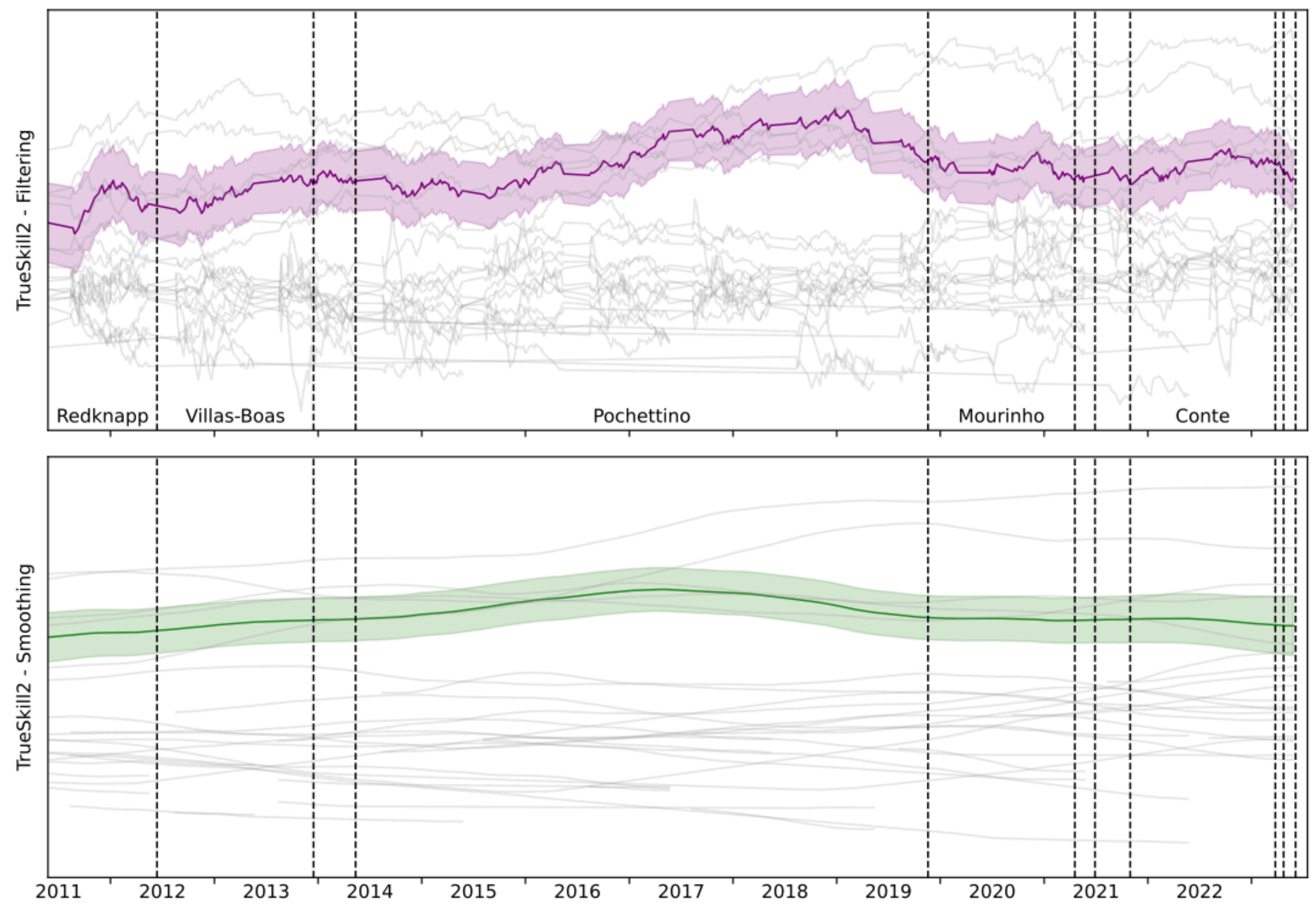


Figure 4: Filtering and smoothing with TrueSkill2 for Tottenham's EPL matches from 2011-2023. Filtering in purple, smoothing in green (error bars represent one standard deviation) with the other teams' mean skills in faded grey. Black dashed lines represent a change in Tottenham manager with long-serving ones named.

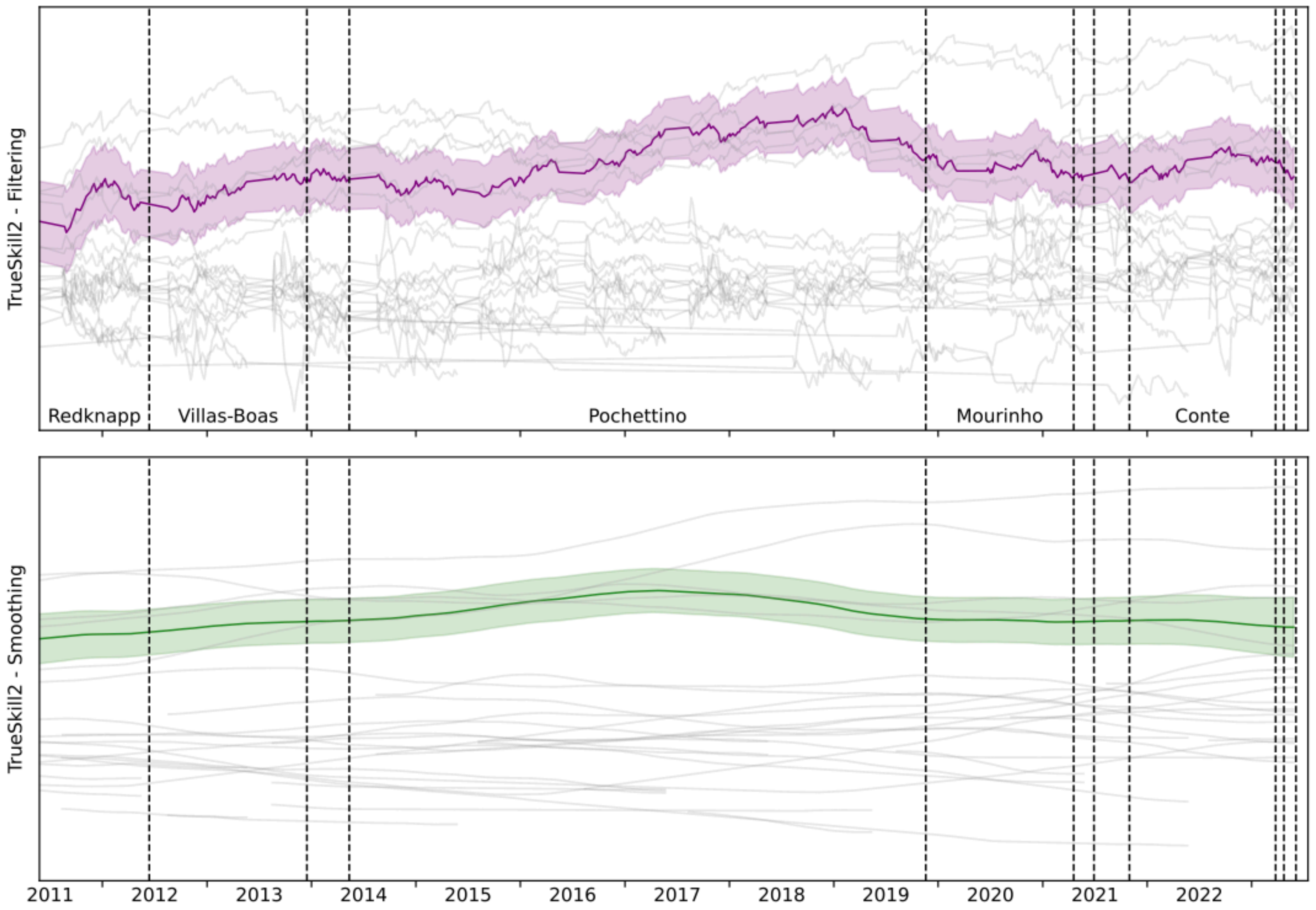


Figure 4: Filtering and smoothing with TrueSkill2 for Tottenham's EPL matches from 2011-2023. Filtering in purple, smoothing in green (error bars represent one standard deviation) with the other teams' mean skills in faded grey. Black dashed lines represent a change in Tottenham manager with long-serving ones named.

Prediction

General Quantitative Evaluation

- fairly similar for tennis, modulo TrueSkill (param. est. issues)
  - binary outcomes, simpler task, performance saturates
- introduction of draws gives Elo difficulties, models seem to help

Table 2: Average negative log-likelihood (low is good) for presented models and algorithms across a variety of sports. In each case, the training period was 3 years and the test period was the subsequent year. Note the draw percentages were 0% for tennis, 22% for football and 65% for chess.

Method	Tennis (WTA)		Football (EPL)		Chess	
	Train	Test	Train	Test	Train	Test
Elo-Davidson	0.640	0.636	1.000	0.973	0.802	1.001
Glicko	0.640	0.636	-	-	-	-
Extended Kalman	0.640	**0.635**	0.988	0.965	**0.801**	**0.972**
TrueSkill2	0.650	0.668	1.006	**0.961**	0.802	0.978
SMC	0.640	0.639	0.988	0.962	**0.801**	0.974
Discrete	**0.639**	0.636	**0.987**	**0.961**	**0.801**	0.976

Table 2: Average negative log-likelihood (low is good) for presented models and algorithms across a variety of sports. In each case, the training period was 3 years and the test period was the subsequent year. Note the draw percentages were 0% for tennis, 22% for football and 65% for chess.

Method	Tennis (WTA)		Football (EPL)		Chess	
	Train	Test	Train	Test	Train	Test
Elo-Davidson	0.640	0.636	1.000	0.973	0.802	1.001
Glicko	0.640	0.636	-	-	-	-
Extended Kalman	0.640	**0.635**	0.988	0.965	**0.801**	**0.972**
TrueSkill2	0.650	0.668	1.006	**0.961**	0.802	0.978
SMC	0.640	0.639	0.988	0.962	**0.801**	0.974
Discrete	**0.639**	0.636	**0.987**	**0.961**	**0.801**	0.976

Discussion

- skill rating problem for competitive sports
- (statistical) models, state-space formulation, generalities
- decoupling modelling decisions from algorithmic decisions
- intertwining of { filtering, smoothing, parameter estimation }
- model-centric approach is particularly accommodating of <u>extensions</u>
  - { covariates, contexts, richer observation models, random effects, multivariate skill representations, … }
- algorithmic extensions: { parallel-in-time, variance reduction, online param. est., … }