$$\begin{matrix} {}^{t}J_1 \\ {}^{t}J_2 \\ \vdots \\ {}^{t}J_N \end{matrix} \begin{bmatrix} O_{12} & \cdots & O_{1M} \\ O_{22} & \cdots & O_{2M} \\ & & \\ O_{N1} & \cdots & O_{NM} \end{bmatrix} \begin{matrix} {}^{t+1}J_1 \\ {}^{t+1}J_2 \\ \vdots \\ {}^{t+1}J_N \end{matrix}$$

$t+1$ is after 1 period (basically 1 week)

O_{ij} is the j^{th} opponent of the player i during the week, M is the max number of matches. $O_{ij} = (D_{p_{ij}}, X_{ij})$

$\forall 1 \le i \le N$, $J_i \in \mathbb{R}$ Player strenght at time t

$D_{p_{ij}} \in \{J_1, \dots, J_N\} - \{J_i\}$

$X_{ij} = \begin{cases} 1 & \text{if } J_i \text{ plays a match } j \\ 0 & \text{if not} \end{cases}$

1-/ How do we use the matrix of opponents MO to go for ${}^{t}J_i$ to ${}^{t+1}J_i$?

1-1/ How to represent ${}^{t}D_{p_{ij}}$ in term of features vectors of the two players at the moment of the match j.

${}^{t}D_{ij} = [$ player 1 (personal data) | player 1 (match data (game level, tournament level), context data ...) | player 1 (historical data) | player 1 (other data) | player 2 ... player 2 $]$

1-2/ Given ${}^{t}D_{p_{ij}}$, ${}^{t}X_{ij}$, how do we compute F such as $F({}^{t}J_i, {}^{t}MO) = {}^{t+1}J_i$

For F, we have 1 baseline : ATP Ranking

2-) Response

1-2/ To build F,

- Elo system (look at the existing literature)
  - x limitation: - user's odds only based on previous ranking and match binary result
  - don't use game context (every wins don't have the same value)

3-) What next, hop?

- Start looking at the available data that could help to represent $D_{p_{ij}}$
- What would be a good period to evaluate F (assume that we have the data available for all t)

4) Coding

- Build a function for F : def F(inputs) ... return outputs

  inputs: MO : values, J_i^t : ..., other data : ...

- Build the inputs processing function: def process_data(raw_data) ... return inputs

  raw_data: csv players (atp ranking), etc.

- Build a training function depending on F (in case of ML)