Research objective

We aim at modelling and predicting performance in Team Sport

Provided with the position of Rugby Union player, we want to predict the evolution of a "success" variable. The selected "success" variable is the position of the ball in the field.

Problem formulation

Let $X_{(i,t)}$ be the position vector of player $i \in \{1, ..., 15\}$ at time t. $X_{(i,t)}$ is 2-dimensional.
Let X_t be $\{X_{(1,t)}, ..., X_{(15,t)}\}$.
Let s be the sequence length and p be the prediction length.
For readability purpose, for any t we define $X_{St} = \{X_{(t)}, X_{(t+1)}, ...X_{(t+s)}\}$
Let A_t be the value of the "success" variable at time t, and $A_{\tau t} = A_{(t+s+p)} - A_{(t+s)}$ be the difference between the "success" variable at the end of the prediction horizon and the "success" variable at the end of the sequence.

Our objective can be formulated as : **Predicting** A_τ **given** X_S

Intermediate finding : when trying to predict A_τ, as we do not have access to the ball position, and use a proxy to determine it, we can infer A_τ from the positions. Unexpectedly, in early experiments, applying the proxy to predicted trajectories yields better results than predicting the proxy directly.

Our task could then be formulated as **Predicting** X_P **given** X_S which leads us to consider existing literature as some communities are working on problem whose formulation is rather similar as ours.

Pseudo related work

We distinguished two communities working of problem whose formulation is rather similar as ours :

Time Series :

- Similarities :
  - Predicting future based on past
  - Task can be either multivariate predicts multivariate, multivariate predicts univariate, univariate predicts univariate.
- Differences :
  - One single and uninterrupted Time Series
  - Does not necessarily involve position data (benchmark datasets do not)
  - Stationarity and decomposition matters
- Reference datasets : 8 datasets, used in all papers I read

- Activity : +++
- Code availability : +++
- Metrics : MSE, MAE for LTF, SMAPE and MASE for STF ## **Multi-agent trajectory prediction** :
- Similarities :
  - Predicting future based on past
  - Position data
  - Use of a scene (in our case, the limits of the field)
- Differences :
  - Outputting various trajectories (which is not an issue)
- Reference datasets : 6/7 (Argoverse 1 and 2, Kitti, Waymo, ETH, SDD), not systematically used in all papers
- Activity : +
- Code availability : ~ (ADAPT, MSLR have a github)
- Metrics : Miss Rate, ADE, FDE

Implementation

Compare oneself to reference algorithms / models :

- Re-use the score disclosed in the paper or rerun the experiment and disclose the obtained score ?
- Comparison with a given "budget" ?
- Ability to improve SOTA in TS or MATP ?
- What time is it reasonable to try and implement a model given in a paper ?

From literature :

- Adversarial training ?
- Multiple losses (reproduction, prediction, classification, ...)
- Graphs NN used in MATP

Other :

- How to ensure the code is right ?
- Risk of overfitting in TS with overlap ?
- Ablation study : is there anything I am missing ?
- Best practices ?
- Dataset publication ?
- Number of citation of the review vs number of citation of any article in it → is the review bad ?

Future work :

- “Rate” player position according to the expected outcome of the sequence ?
- Analyze collective behavior metrics vs expected outcome
- Generate counterfactuals
- Reinforcement learning (training an agent to “move the pieces” and see how it does)

Additional research questions

Second question : relevance of feature engineering (dispersion, polarization, . . .) vs Deep Learning approach.

Third question : velocity or position to achieve best results ? (No longer so accurate : standardization issue)