# PSV Project

## Context

This is a project that consists of a collaboration between the football club PSV and students from the Applied Data Science minor at Fontys. The client is Ruud van Elk who is Head of Sport Science & Analytics. PSV already applies data science in multiple different aspects of their business. One of the ways that PSV uses data science is to evaluate played games. This can be games played by PSV or games that are played by different club and maybe even different leagues. PSV uses the data of their own games to evaluate how well individual players and the team are playing. PSV uses the data from other teams for scouting purposes.

Right now, PSV uses xG (expected goals) for these analyses. PSV gave as assignment to research the xT (expected threat) model. The difference between xG and xT is that xG only looks at the chance that the result of an action will result in a goal, and xT is able to look more layers deep. For example: if someone passes the ball to the location where players often create assists than xT will reward that player for that, xG will not because players do not often score from that position. The goal of the improved model should be that valuable passes and dribbles should also get recognition instead of just the goals.

While researching xT the team came across another soccer action evaluation model called VAEP (Valuing Actions by Estimating Probabilities). The team took the initiative to read some literature on both models and presented the findings to the client. After a discussion the assignment was changed to research on VAEP since it takes more features into account and looked like it had more potential.

## Dataset

The dataset contained every so-called on-the-ball action of the Eredivisie from multiple seasons. This means that every row is an individual action that belongs to a game, player, and time. Each action also contains a lot of information about the action itself such as: the action type, the starting- and end location, and the name of the next player that received the ball. Apart from just the action type each action also comes with a set of extra labels that sometimes contain information like the body part, if the action is forward, or whether it is completed.

An important note is that the data is not auto generated and is created by humans looking at the model, with the help of software of course. This does mean that labels can be inconsistent and mistakes in other aspects can also occur. Another import factor about the data is that it only contains events that the ball went through, the data does not have information about other friendly or enemy players that are not at the ball.

## Data pre-processing

Both xT and VAEP require the data to be in a specific format. This format is called SPADL which stands for Soccer Player Action Description Language. Some of the attributes were already inside the data provided by PSV. However, others had to be calculated or created with the help of the attributes from the original dataset. Furthermore, an EDA was performed on the data to understand the structure and the content, as well as what problems are in there and need fixing. All the necessary manipulations to turn PSV's data into to the SPADL format were saved in methods distributed in two python scripts.

Preparing the data was the biggest, most complex, and, of course, the most important part of the project.

## Models and tools

While researching on xT and VAEP the team found a python-based library that already provides both models and some of the data pre-processing methods. The library is called socceraction and it is open-source. Unfortunately, the data pre-processing had to be re-done but the source code was used as an inspiration and guidance. Thankfully the code for the model seemed to work. However, once the data was completely prepared to be fed to the model the team already had a way better understanding of how the code works. Therefore, they started to read through the source code of the VAEP model. There were changes that had to be made due to the difference in the datasets. The main problem was that the same actions had different names. Some of the changes that had to be made were spotted immediately and others were overlooked. In the end, however, everything was fixed or at least what the team managed to find. It should also be noted that during the project the team put the most focus on the working with results from the VAEP. This means that the team made different kind of summaries that showed the top teams, players and actions. The team also made a summary that showed the best and worst actions from PSV from the entire season. This was also used for validation of the model since the scores could be compared with how valuable the client and the team though that the action actually was.

## Results

During the project, a lot of results were yielded due to the different changes of the model. Firstly, the team spent time analysing why the base model was not working. Then, after figuring it out and making the necessary changes, the first results were received. The output was a dataframe with the ratings and there were three categories - offensive, defensive, and vaep value. The team had to draw and explain the insights to the client. They were presented in the form of a dataframe. Then, it turned out more of the source code needed changing so there were new results. The new results were drastically different. They were different in a way that the first batch of results resembled the real-life results better, but the summaries about the best/worst actions made more sense in the second batch of results. Using the first model AJAX is rated as the best team, and PSV is 3rd or 4th, whereas according to the second model they are almost at the bottom. However, for the best/worst actions the first model rewarded with the highest defending value to an action that was an own goal, which makes no sense. With the fully changed source code, this issue was resolved, and the highest offensive values were not goal attempts anymore, but there were fouls and passes. After taking a different approach and changing some of the action names to match the list in the spadl.config file, the obtained results were similar to the previous ones but probably slightly better. In both scenarios there is a big negative number for the defensive value, however, at least in the latter case, the vaep value is not a negative number anymore. Therefore**,** this might be a step in the right direction. It is interesting why all of this is happening, but since the team does not have any more time to figure it out, this task is for the team of professionals from PSV and ASML.

## Future work

Even though, there was not enough time in the end to investigate the results, based on what was learned during the project the team has a few suggestions for the future.

First, and probably foremost, a clear criterion of how the success of an action is decided should be created. This is something that might be affecting the results greatly. During the project, the score from the Effectivity column in the original dataset was used, but that is not optimal.

Moreover, all the methods inside the python scripts regarding VAEP should be checked again, because something there could be causing the strange results.

In the examples in the socceraction library when the actions are calculated the home team id is taken to order the actions left to right. This is something that the team skipped because it is done inside the VAEP scripts, as well, but it might be causing problems.

And last but not least, try to change all action names in PSV's data, so they match the list from the spadl.config script. After changing just a few the results were slightly better, so probably adapting the data to the source code, and not the other way around as was done, is the better option.