# Measuring a Classification Approach to Predict Playing Positions of Association Football Players

**Daron Assadourian**
University of California, Berkeley
dassadourian@berkeley.edu

## 1 Abstract

In this project, we use a PCA-reduced European Football players attribute dataset to predict the position of the player. There are two classes that the value of the position can take on, namely goalkeeper that is represented by the number 0 or outfield player that is represented by the number 1. After reducing the dataset to three dimensions, we now train a model to predict the position of each player and we measure the Memory Equivalent Capacity as well as various other measurements in order to determine the the generalizability and learnability of the data with its models. We anticipate meaningful and accurate results as the two classes take on significantly different and distinct values for each feature.

## 2 Introduction

In an effort to retrieve positions of players that were not readily available in the table of 35 features, the project's aim is to explore machine learning techniques to reduce the ambiguity between players' statistics and to identify patterns among the features. The dataset is a modified version of a table taken from the popular and extensively used "European Soccer Database" uploaded by Hugo Mathien onto the Kaggle website[1]. The 35 attributes include useful features such as stamina, sprint speed, goalie diving skills, shot accuracy etc. and although they are many in number, we are aware of the fact that in almost all cases, goalkeepers are terrible outfield players and outfield players are terrible goalkeepers. This known phenomenon brings upon us the idea of reducing the data and focusing on the most important distinguishing features between players to be able to visualize the differences as well as to build efficient models that accurately distinguish between the two classes. In this process, it is necessary to extract a relationship between similar players and to finally label each player as either a goalkeeper or an outfield then proceed to training and validating it on an independent test set. The Information theory is a necessary tool to confidently deduce useful models. Following the spirit of using information theory, in this project we employ memory capacity measuring tools to make inferences on our chosen datasets and chosen models learnability and accuracy.

## 3 Generalization in Machine Learning

Introduced by Dr. Gerald Friedland, Memory Equivalent Capacity is defined as follows:

**Intellectual Capacity:** The number of unique target functions a machine learner is able to represent (as a function of the number of model parameters).

**Memory Equivalent Capacity (MEC):** A machine learner's intellectual capacity is memory-equivalent to N bits when the machine learner is able to represent all 2N binary labeling functions of N uniformly random inputs.

There are 4 rules that define the concept of Memory Equivalent Capacity[2]:

1. The output of a single perceptron yields maximally one bit of information.
2. The capacity of a single perceptron is the number of its parameters (weights and bias) in bits.
3. The total capacity $C_{tot}$ of $M$ perceptrons in parallel is $C_{\text{tot}} = \sum_{i=1}^{M} C_i$ where $C_i$ is the capacity of each neuron.
4. For perceptrons in series (e.g., in subsequent layers), the capacity of a subsequent layer cannot be larger than the output of the previous layer.

Figure 1: The figure establishes the 4 rules of Memory Equivalent Capacity

The Memory equivalent capacity of a dataset has an upper bound and it is defined as follows:

**Maximum Memory Equivalent Capacity:** The size in bits of the lookup table for the data set.

**Estimated Memory Equivalent Capacity:** is found by building a static-parameter(weights set to identity; biases are learned) machine learner to memorize the training data.

The concept of Memory Equivalent Capacity is essential for establishing the definition of Generalization. An ideal Machine learner finds patterns in the data and minimizes the number of parameters that it uses in order to accurately predict the data. Optimal generalization goes hand in hand with Occam's Razor since among equally accurate models , we want to choose the model that has the lowest Memory Equivalent Capacity. In order to ensure that generalization is possible with the given dataset , it is necessary to implement generalization progression. Measure the Estimated Memory Equivalent Capacity for 10%, 20% ... 100% of the data and plot the result. If the capacity continues growing as the percentage increases, then there is not enough data to generalize or the data is too random. A trained model's progress in generalization can be measured by the generalization ratio. The Generalization ratio is measured in bits/bit and is defined as follows:

$$\textbf{Generalization Ratio} = \frac{\text{Number of Correctly classified instances}}{\text{Memory Equivalent Capacity}}$$

A model with GR < 1 requires more data or training, a model with GR = 1 is essentially memorizing hence overfitting. A model that has a GR » 1 where GR is significantly larger than 1 successfully generalizes with no chance of overfitting. A similar concept can be established with Resilience. Resilience is the amount of variance an instance is allowed to assume before changing a predictor's outcome. It is the inverse of generalization. Formally, it is defined as:

$$\textbf{Resilience} = 20 \log_{10}\left(\frac{\text{Memory Equivalent Capacity}}{\text{Number of Correctly classified instances}}\right)$$

## 4 Method

Our main objective is to label the dataset of interest with the two classes mentioned above, implement the measurements and construct models and analyse the results. For clarity, our data processing, research as well as measurement analysis are broken up into several steps and outlined below with details and descriptions:

1. **Preprocessing:** We first Obtain the dataset from the Kaggle database[1]. This database contains a few tables. Our tables of interest are the attributes and players tables. Using pandas module in python, we then merge the two tables such that each player in the players table is now matched to his corresponding row on the attributes table by combining the two tables based on the player_id column. We then drop all rows that contain 'Nan' values in

the data processing stage. Due to the quality of the data, few rows are dropped. Certain ages, in particular ages below 20 and above 42 contained data with high variance. In order to avoid the consequences of using bad-behaving data, we excluded ages below 20 and above 42. We then proceed to visualize the features through scatterplots, correlation matrices, Kernel Density Estimate plots etc. in order to deepen our understanding of the dataset.

2. **Principal Component Analysis:** After we have a detailed understanding of the dataset and its patterns , we decide to implement dimensionality reduction to transform the data from a 35 dimensional space to a much lower 3 dimensional space. We use the PCA library in the scikit-learn module in python to achieve this. Our dataset now contains the names as the indices and the 3 best principal components. We then explore methods to generate the labeling to assign each player a position.

3. **K-Means Clustering:** Now that we have a reduced dataset, we implement K-means clustering using the elbow method heuristic (scikit-learn) and deduce that on the reduced dataset, the optimal K for k means clustering is K=4. Dissatisfied with the given optimal K, 2-Means clustering is implemented and the results are plotted on a 3 dimensional scatterplot as shown in figure 2 below. An extraordinary phenomenon is witnessed. There are two clusters that are completely separated from each other with a negligible amount of outliers.

4. **Labeling and Interpretation:** After manually observing the players in each cluster, we realized that one cluster exclusively contains all goalkeepers while the other contains all the outfield players. This is certainly not due to chance, since many of the original features for goalkeepers such as shooting, dribbling and similar features were extremely low and those features took on high values for the outfield players. Similarly, all the features that involved Goalkeeping skills were extremely high for all the goalkeepers compared to the outfield players. Therefore, we concluded that the labels produced were almost 100% accurate. Since in each team there are on average 2 goalkeepers and about 25 players, the classes in our dataset are unbalanced. In order to balance the classes, we select all the goalkeepers and randomly select the same amount of outfield players. The resulting table has 1208 rows, 3 principal components as features and a position column as the target output. Due to the nature of the resulting dataset accompanied with the reasons described above, we expect a non-trivial model to have near perfect accuracy both on the training set and the validation set. Noise is deemed to be negligible hence ignored since we extensively cleaned the dataset. We randomly sample 1000 instances and keep it as the training set and hold out the remaining 208 instances as the test set.

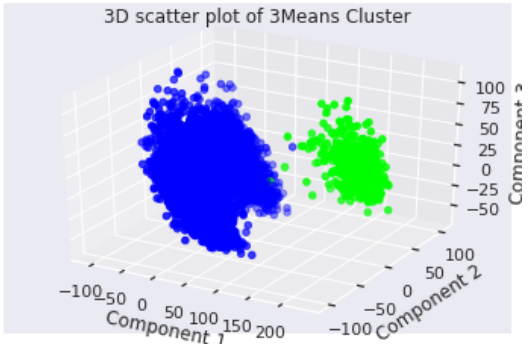


Figure 2: The figure shows a clear separation of the two classes and the separation is across Component 1 (Blue are outfield players, green are goalkeepers).

## 5 Model and Results

Using the Tools and techniques that were established above, we now present our findings. The capacity measurements were accompanied with 2 main types of models. The first model that was widely used across different combinations of the 3 features of data included a neural network of 1 hidden layer of 2 neurons with Rectified Linear Unit (Relu) activation function, batch size of 10,

learning rate of 0.01 and an output layer of binary classification logits. The full rank dataset had a Memory Equivalent Capacity of 4 and as a result an Expected generalization ratio of 250. Following the formula established earlier, the average resilience is -47.958. The noise resilience was noticeably strong. The capacity progression was consistently equal to 2 decision points for all fractions of the training dataset as shown in figure 3 below. This means that there was enough data for training the model. The similar and soporiferous result can be seen with the models performance on both training set and test set for all fractions of data as seen below in figure 4. This model was also implemented on reduced datasets that both contain the first component and exactly one of the other components as features. They yielded boring results in terms of accuracy and capacity progression/variation of the generalization ratio as a function of the fraction of the dataset used.


Figure 3: The figure displays the capacity progression of the training dataset (constant)

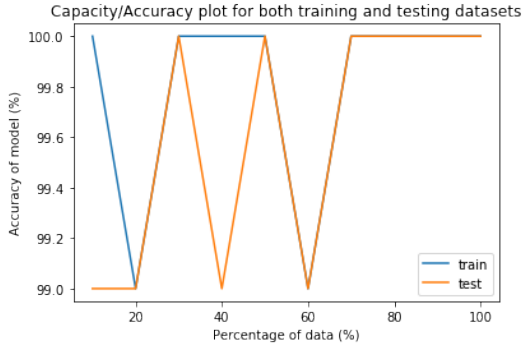


Figure 4: The figure displays the performance of the model based on the amount of training data

The second type of model that was employed was for a more special case: the reduced dataset that only contains the second and third components as the features as evident in figure 5. Due to the overlap of a noticeable amount of points, a more complex neural network of 1 hidden layer and 16 neurons was implemented with a Relu activation function, batch size of 10, learning rate of 0.01 and an output layer of binary classification logits. The results that were found in this experiment were not as trivial as that of the first model and full rank dataset. The capacity progression displayed in figure 6, shows some kind of progression and increase in the number of parameters, suggesting that there is not enough data despite the overall reasonable performance of the model displayed in figure 7 that also shows the generalization ratio of the model. We decided to include the generalization ratio along with the rest on the same plot even though they have different units and interpretations, because the generalization ratio was around 22.2 for all fractions of the dataset. This is slightly counterintuitive when we compare it to the capacity progression plot.

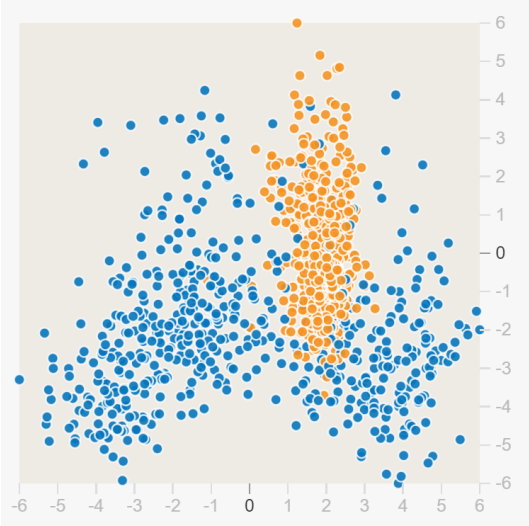

Figure 5: The figure displays the distribution of Components 2 and 3 on a scatterplot depicting the goalkeepers in orange and the outfield players in blue.

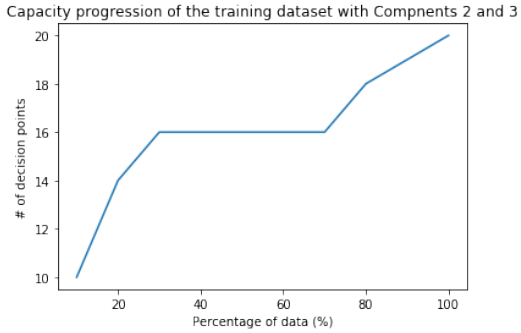


Figure 6: The figure displays the capacity progression of the training dataset consisting of the 2nd and 3rd Components as features.

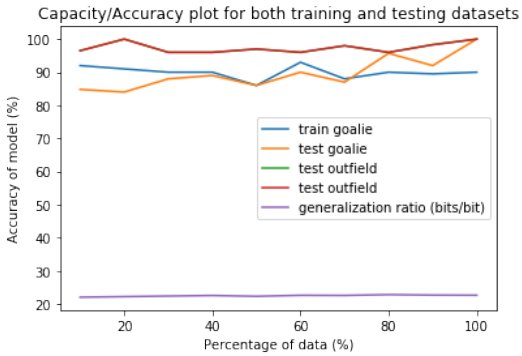


Figure 7: The figure displays the accuracy as well as the generalization ratio for each fraction of the dataset. Note the generalization is not a percentage.

## 6 Analysis

The results observed should not come as a surprise. The 3 dimensional dataset includes all three components obtained from the Principal Component Analysis as features in the process of prediction. Component 1 almost perfectly separates the data, therefore any dataset containing Component 1 will have near perfect predictions. This is because by definition of Principal Component Analysis , the components are sorted according to their variation in decreasing order. We saw above that the dataset containing the latter two components did not enjoy an accuracy of a 100% like the full dataset. For the second model it was necessary to run the model for over a thousand epochs in order for the model to converge. In contrast, any other reduced rank 2 dataset as well as the full dataset trained almost perfectly in a matter of a few epochs. This was a result of the separability of all the datasets that contained the component 1 as a feature. The other reduced rank datasets had less interesting results than the ones explored above due to their separability, and they had a consistent generalization ratio of 302 for all fractions of the training dataset and perfect accuracy on both the training and testing sets.

## 7 Future Work

The entire experiment had many non-trivial steps thus making the process complicated enough to discover interesting patters in the dataset. One method that we wish to explore and employ in the future is to implement 3 Means clustering algorithm to separate the dataset into 3 main categories: attackers, defenders (both belonging to outfield) and goalkeepers. Then there will be a multiclass classification and some of the measurements will be the multiclass version of the ones used in this project. Nevertheless, there are many directions in which we can head towards to stack a significant amount of complexity in order to yield reassuring results.

## 8 Conclusion

We conclude with great confidence that due to the nature of the dataset and setup, prediction of independent test sets were extremely accurate for all the datasets that included Component 1 as a feature, namely the most important feature in this particular experiment. Since almost all the players behave similarly in regards to their statistics and unlabeled roles, any new player can be labeled and classified correctly with about a 99% chance guarantee using the three features that were found in the dataset. The models used were pretty simplistic, because of the linear separability of the data. As a result, the learnability and generalization are efficient and high enough that the model yields great confidence for predicting positions of other football players that were not present at the time of training as well as validation.

# 9 Appendix

In this section, some of the other less significant results are shown.

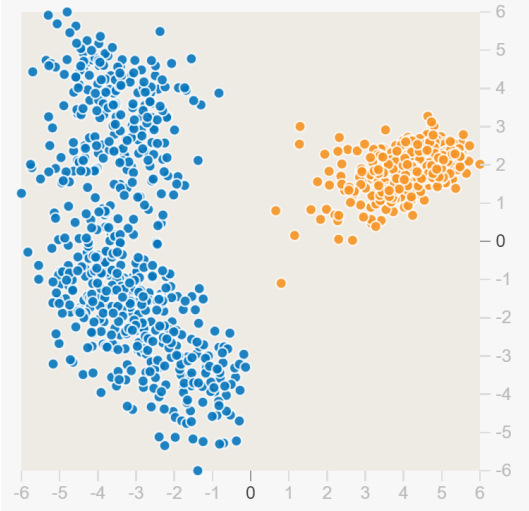


Figure 8: The figure displays the distribution of Components 1 and 2 on a scatterplot depicting the goalkeepers in orange and the outfield players in blue.


Figure 9: The figure displays the capacity progression of the training dataset consisting of the 1st and 2nd Components as features.

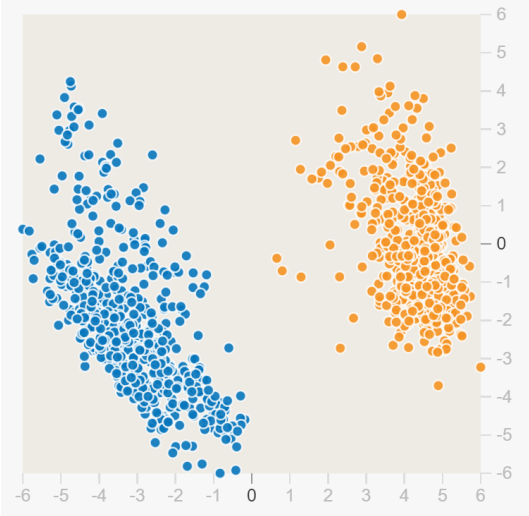

Figure 10: The figure displays the distribution of Components 1 and 3 on a scatterplot depicting the goalkeepers in orange and the outfield players in blue.

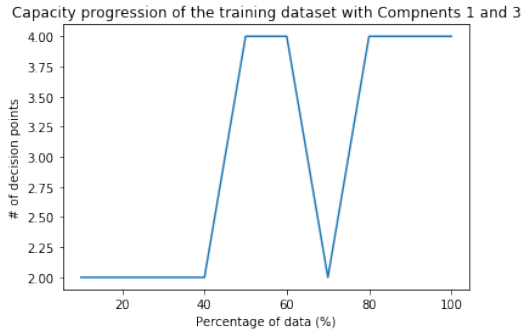


Figure 11: The figure displays the capacity progression of the training dataset consisting of the 1st and 3rd Components as features.

## 10 References

[1] Hugo Mathien. European Soccer Database https://www.kaggle.com/hugomathien/soccer
[2] G. Friedland. Measuring Generalization in Machine Learning.
[3] G. Friedland. Generalization. https://youtu.be/UZ5vhqDKyrY, 2019.
[4] G. Friedland. Reproducibility and Experimental Design for Machine Learning on Audio and Multimedia Data