The goal of the project is trying to predict the chances of accuracy from a random sample of shots made by Kobe Bryant during his career, using a database provided by a Kaggle competition and Python code (packages pandas, numpy and sklearn).

In order to do so, we handle a dataset with about 30,000 events and 25 variables (**data.csv**), both categorical –such as type of action, area of the field from where the shot was thrown, rival, moment of the season, minute, etc.—and numerical –such as location of the shot using latitude and longitude coordinates, remaining time to finish the period, shot distance, etc.--.

The first step was to explore the dataset and to identify the variables that might be more relevant for the exercise, as well as to eliminate the ones that were clearly superfluous (**01_Exploring_data**). Afterwards, we exported the corrected dataset and used Tableau to try to detect the more relevant variables and to decide whether cross them or not. In the file **Kobe_searching_new_variables.twb** one can see some charts with the variables that turned out to be the most relevant ones.

Also, in these charts one can see that categorical variables such as type of shot or shooting area widely differ from the average (45% shots in vs. 55% shots out). That is why, in order to find new variables that might be relevant, we have created new categorical variables crossing values such as type of shot with shooting position (these new variables can be found in **02_Creating_new_variables**). The rest of the variables related to shots were quite uniform, despite any other factor.

In **03_Preparing_data_for_prediction_model**, we converted some variables into categorical dtype for better summarization. And we also created dummy variables, since the model of Logistic Regression of scikit-learn only accepts numerical values. At this point, it would have been advisable to run a feature selection on the sparse matrix, to avoid inefficiencies in the model (the more variables, the harder to get the Logistic Regression to detect the truly relevant ones). We tried to do so by using Boruta with a Random Forest Classifier, but we found technical problems and were not able to solve them. Therefore, to overcome this situation, we decided to split the datasets into numerical variables, old categorical variables and new categorical variables.

In **04_Prediction_Model** we apply a Logistic Regression to all three datasets. After getting the results, we decided to merge the first and the second ones, in order to check whether it offered a higher rate of accuracy, but we finally concluded that the best possible prediction model was obtained with the dataset with sparse matrix of the original categorical variables.

In **05_Predicting_shot_in_or_out** we again created the model from the mentioned dataset and applied it to the rows where the target value was a NaN value. This way, we finished the process, getting a list of predictions of the probability that an unknown shot goes in or not.

In **Kobe_visualization.twb** one can also find some descriptive visualizations of the accuracy rate, segmented by different relevant variables.

Note: The fifth and last script of the code has not been checked since the cells could not be executed due to memory problems.