Daniil Sideris

Prediction of round winner in Counter-Strike: Global Offensive using discriminant analysis and logistic regression

Project seminar:
Analysis of Multivariate Data
Supervised by: Dr. Bastian Gribisch

Keywords: Counter-Strike: Global Offensive, eSports, discriminant analysis, logistic regression analysis

Institute of Econometrics and Statistics
Faculty of Management, Economics and Social Sciences
University of Cologne

Cologne, July 17, 2022

Contents

1. Introduction

Since the ancient world competitive sports have been an important part of training the body physically and mentally, where different individuals or teams could compete with each other. Competitive sports can be seen as an source of testing strengths and pushing the limit of the human body and brain. Additionally, sports can be viewed as entertainment for fans, a job market for the athletes and different inside and outside businesses.

Different sport types have been evolved and have their fan base and their spots in the world of competitions, like soccer, basketball or chess. In the last decade electronic sports (eSports) have seen growth in popularity with an fanbase of 495 million viewers (see wepc.com[1]). The world market revenue of eSports is approximately 1.4 billion U.S. dollars in 2022 (see statista.com[2]). eSports organizations set up tournaments where the best eSport athletes compete with each other in various games. eSports are difficult to define, as the sports are presented virtually, on various platforms and in a bright spectrum of games.

The Counter-Strike game series are one of the most popular competitive video games and as well the most played game from the online store Steam - the digital distribution service and storefront by Valve. The last game of the series Counter-Strike: Global Offensive (short name CSGO) has reached an monthly number of simultaneously players of nearly 1 million (see statista.com[3]).

Counter-Strike: Global Offensive is an first-person shooter developed by Valve and Hidden Path Entertainment. The game has two teams of 5 players in each, the Terrorists (T) and the Counter-Terrorists (CT). The main game mode involve the Terrorists planting a bomb while Counter-Terrorists attempt to stop them. Both teams play for a best of 30 rounds, first team to reach 16 rounds wins the game. After the first 15 rounds the teams swap sides. There are 7 main maps. A teams wins a round as Terrorists by either planting the bomb and making sure it explodes, or when all players of the enemy team are eliminated. The Counter-Terrorist team wins by eliminating the other team or by defusing the bomb, in case the Terrorists team has planted it. The game is highly focused on strategical thinking, reflexes and team working.

As in other sports, eSports is collecting, analyzing and presenting different statistical insights before, during and after matches. This insights are used for improving the viewing experience or training and analyzing by the players and coaches. Some statistical insights can be used for predictions of future outcomes in matches or different aspects of the game.

[1]https://www.wepc.com/statistics/esports-gaming/
[2]https://www.statista.com/statistics/490522/global-esports-market-revenue/
[3]https://www.statista.com/statistics/808630/csgo-number-players-steam/

One crucial aspect during a match in Counter-Strike: Global Offensive is the understanding of the outcome of current round, analyzing past rounds and predicting next rounds. This aspect is important as it allows to change the strategy, understand what criteria are involved for success or failure. Understanding this aspect is part of the training of the athletes and can be crucial for winning a whole match.

For predicting and classifying the outcome of an round in CSGO I will use discriminant analysis and logistic regression, as they both are quite popular and accurate techniques and usually not far away from each other. In this paper I want to show how these models classify an round outcome, what the similarities are and where the differences lie.

In the first part of this study I will explain how the data was gathered and give an descriptive analysis of the dataset. Afterwards, I will describe how discriminant analysis and logistic regression work. Another point will be the ROC-curve, specificity and sensitivity. In the last part I will share the results and interpretation of both models together with a discussion and outlook.

2. Problem

For this study, I used a dataset, which was published by Skybox Esports Technologies[1]. This company uses Artificial Intelligence to collect data during matches in CSGO to provide insights and performance analysis for teams and fans. The dataset presents a collection of 700 demos (recorded games) of competitive matches in tournaments in 2019 and 2020. As mentioned before, one match in CSGO contains 16 to 30 rounds, with each round having a maximum duration of 1 minute and 55 seconds.

The dataset has been cleaned before being published. The Skybox team has filtered unnecessarily data from warm-up rounds and restart rounds. The Skybox Esports Technologies Team has collected snapshots of data from the 700 demos each 20 seconds until the end of an round. This gives us a number of snapshots (observations) of 122411. All snapshots are i.i.d in the sense that they each describe the state of a round and can therefore be treated individually. Although multiple snapshots can be taken from the same round. Every snapshot from the original dataset consists 97 variables. The variables can be divided into themed groups and can be observed in the table of variables (see table 1).

[1] https://skybox.gg/

N.	Variable	Definition	Key
X_1	time_left	Time left in the current round	
X_2	ct_score	Current score of the CT team	
X_3	t_score	Current score of the T team	
X_4	map	Map the round is being played on	E.g. de_dust2, de_mirage
X_5	bomb_planted	If the bomb has been planted or not	False = No, True = Yes
X_6	ct_health	Total health of all CT players	Player health in range 0-100.
X_7	t_health	Total health of all T players	Player health in range 0-100.
X_8	ct_armor	Total armor of all CT players	
X_9	t_armor	Total armor of all T players	
X_{10}	ct_money	Total bankroll of all CT players	Amount in USD.
X_{11}	t_money	Total bankroll of all T players	Amount in USD.
X_{12}	ct_helmets	Number of helmets on the CT team	
X_{13}	t_helmets	Number of helmets on the T team	
X_{14}	ct_defuse_kits	Number of defuse kits on the CT team	
X_{15}	ct_players_alive	Number of alive players on the CT team	Range 0 to 5.
X_{16}	t_players_alive	Number of alive players on the T team	Range 0 to 5.
X_{17-84}	ct_weapon_X	Weapon X count on CT team	E.g. Ak47, Deagle and UMP45.
X_{17-84}	t_weapon_X	Weapon X count on T team	E.g. Ak47, Deagle and UMP45.
X_{85-96}	ct_grenade_X	Grenade X count on CT team	E.g. HeGrenade, Flashbang.
X_{85-96}	t_grenade_X	Grenade X count on T team	E.g. HeGrenade, Flashbang.
Y	round_winner	Team winner of the round	Counter-Terrorist = 0, Terrorist = 1.

Table 1: Dataset variables
abbr.: CT = Counter-Terrorist; T = Terrorist; USD = United States Dollar

The dataset is stored in the *CSGO_raw.csv* -file. Before analyzing the dataset it has been cleaned and prepared for the use case with the programming language Python and various packages. Packages which I used here are 4 packages for the data cleaning: (1) pandas, (2) numpy, (3) math and (4) sklearn; as well as 3 packages for visualization: (5) matplotlib, (6) seaborn and (7) plotly. In python I checked for null values and dropped columns with only one value (columns 22, 30, 37, 52, 58, 60), which are different weapon types used very rarely or never in the game. The variables *map*, *bomb_planted* and *round_winner* (see table 1) have been decoded from categorical to numerical variables. The decoding is as follows: 8 numbers for the different maps, 1 or 0 when the bomb has or not been planted, and 0 when the round winner is the Counter-terrorist team and 1 when the winner of the round is the terrorist team. After the operations the dataset contains 91 variables and is stored in the *CSGO_cleaned.csv* -file.

The dataset lists an big amount of variables. To learn more about the dataset descriptive statistics for the variables (except the *weapon_x* -variables) are listed in the table from the python output (see table 2).

Variable	Mean	Std	Min	25%	50%	75%	Max
time_left	97.8869	54.4652	0.01	54.92	94.91	166.9175	175.0
ct_score	6.7092	4.7904	0.00	3.00	6.00	10.00	32.0
t_score	6.7804	4.8235	0.00	3.00	6.00	10.00	33.0
map	3.5202	1.9291	0.00	2.00	3.00	5.00	7.0
bomb_planted	0.1118	0.3151	0.00	0.00	0.00	0.00	1.0
ct_health	412.1066	132.2933	0.00	350.00	500.00	500.00	500.0
t_health	402.7145	139.9190	0.00	322.00	500.00	500.00	600.0
ct_armor	314.1421	171.0297	0.00	194.00	377.00	486.00	500.0
t_armor	298.4447	174.5765	0.00	174.00	334.00	468.00	500.0
ct_money	9789.0238	11215.0423	0.00	1300.00	5500.00	14600.00	80000.0
t_money	11241.0367	12162.8068	0.00	1550.00	7150.00	18000.00	80000.0
ct_players_alive	4.2738	1.2055	0.00	4.00	5.00	5.00	5.0
t_players_alive	4.2662	1.2283	0.00	4.00	5.00	5.00	6.0
round_winner	0.5098	0.4999	0.00	0.00	1.00	1.0000	1.0

Table 2: Dataset descriptive statistics (rounded to 4 decimal places)
abbr.: Std = standard deviation; Min = minimum value; 25% = .25 quantile; 50% = .5 quantile; 75% = 0.75 quantile; Max = maximum value

The objective of this paper is predict the outcome of a round in Counter-Strike: Global Offensive. Therefore, the depended variable can be defined when the round is won by the terrorists or counter-terrorists teams (variable *round_winner*). The dependent variable can be coded as follows (see equation 1):

$$Y = \begin{cases} 0 & \text{, The counter-terrorists team will win the round} \\ 1 & \text{, The terrorists team will win the round} \end{cases} \tag{1}$$

3. Methods

If we have a look at the depended variable, we see that we are facing an binary classification problem. There are different statistical methods which can be applied on an binary classification problem. In this study I will use discriminant analysis and logistic regression.

The theory discussed in this paper has been presented in various literature of Fahrmeir et al. (1996), Rencher et al. (2012) and Backhaus et al. (2018). The chapters have been reviewed and adapted for the use case.

3.1. Discriminant analysis

Fisher's Discriminant analysis is a quite sufficient supervised classification and dimension reduction technique of multivariate data. The model is based on an discriminant function, which represents a linear combination of independent variables that are capable to discriminate between different groups. The discriminant function can be observed from the equation (see equation 2).

$$D = a + b_1X_1 + b_2X_2 + ... + b_jX_j \tag{2}$$

where D is the discriminant score, a is the y - intercept of the regression line, b is the discriminant function coefficient with j $(j = 1, 2, ..., J)$, x is feature variable with j $(j = 1, 2, ..., J)$.

The parameters b_0 and b_j $(j = 1, 2, ..., J)$ are to be estimated based on data for the feature variables. For each element i $(i = 1, ..., I_g)$ of a group g $(g = 1, ..., G)$ with the characteristic values X_{jgi} $(j = 1, ..., J)$, the discriminant function provides a discriminant value D_{gi}. An data point will be allocated to a class, which has the highest discriminant score D_{gi}.

The idea of Fisher's discriminant analysis is to maximize the ratio of the variances between groups and withing groups (see equation 4), and find a linear combination of the predictors (see equation 3) to predict the class.

$$y = a^{'} * x \tag{3}$$

with maximize criterion $Q(a)$

$$Q(a) = \frac{\sum_{g=1}^{G} n_g(\overline{y}_g - y)^2}{\sum_{g=1}^{G} \sum_{i+1}^{n_g}(y_{ig} - y_g)^2} \tag{4}$$

3.2. Logistic regression

Logistic regression is another classification algorithm. As example, logistic regression can be used when discriminant analysis assumptions are not met. Logistic regression belongs to the class of structure-testing methods. As the name suggests, it is a variant of the regression analysis, with the special feature that the dependent variable Y is a categorical variable whose values $(g = 1, ..., G)$ represent the alternatives. Since the occurrence of events is usually subject to uncertainty, Y is considered a random variable and the probabilities for the expressions of Y are predicted.

For $G = 2$ alternatives, Y forms a binary (dichotomous) variable and accordingly we speak of a binary logistic regression. For $G > 3$, we speak of a multinomial logistic regression. In the binary case (which we will deal with in this paper) the groups are usually indexed by 0 and 1 and accordingly Y is then a 0,1 variable. For the probabilities holds a special relationship (see equation 5):

$$P(Y = 0) = 1 - P(Y = 1) \tag{5}$$

For the design of the logistic regression model (see equation 6), the logistic function is used, from which the name results:

$$p = \frac{e^{z(x)}}{1 + e^{z(x)}} = \frac{1}{1 + e^{z(x)}} \quad with\ z(x) = \beta_0 + \beta_1 x_1 + ... + \beta_J x_J \tag{6}$$

The systematic component $z(x)$ forms a predictor for the a-posteriori probability $\pi(x)$. The larger $z(x)$, the larger $\pi(x)$. Accordingly, the larger $z(x)$, the smaller $P(Y = 0|x)$. The logistic function can be also plotted with z on the x-axes and p on the y-axes (see figure 1).

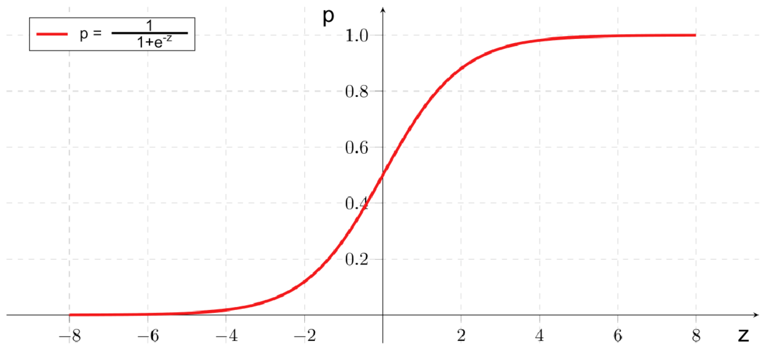

Figure 1: Logistic function

Because of the fact of nonlinearity of the logistic model, the interpretation of the coefficients as measure of the effect of an independent variable is more difficult than, as example, in linear regression. The problem arises from the fact that the effects are not constant, but also change with the dependent variable. Therefore, in general, it is possible to say how the dependent variable changes, when a parameter or independent variable changes, but not how much the change is.

Following can be said about the coefficients: (1) A change in the constant term β_0 causes a horizontal shift of the curve across the x-axis in the logistic model. As β_0 is increased, the curve shifts to the left and the probability at a given value x increases. (2) Increasing the coefficient β_J causes the curve to rise more steeply in the middle region. Since the curve is S-shaped, a steeper slope in the middle region must result in a flatter slope in the outer regions. For $\beta_J = 0$, the curve flattens to a horizontal line. (3) A negative sign of the coefficient β_J causes a sloping curve.

The coefficient β_J determines how the independent variable x_J affects the dependent variable. In linear regression, a change in x_J by one unit causes a change in the dependent variable by β_J. In logistic regression, the effect of a change in x_J also depends on the value of the dependent variable p. The effect is greatest when $p = 0.5$, and it becomes smaller the more p deviates from 0.5.

Speaking about the classification of a data point to a group, it depends on the value of the logistic regression p. There has to be defined an separating value (cut value) p^*. It thus applies to the separation between alternatives (see equation 7)

$$y_k = \begin{cases} 0 & \text{, when } \; p_k > p^* \\ 1 & \text{, when } \; p_k \leq p^* \end{cases} \tag{7}$$

where p_k is the probability of the data point to belong to class k, p^* is the cut-value and y_k the class of the data point.

3.3. ROC-Curve, specificity and sensitivity

To measure the performance of data analysis and classification models, we use various measurements. One important topic is the classification matrix to compare true and false predictions. The ROC-curve (Receiver Operating Characteristic) is an another important topic when we speak about classification with logistic regression and other techniques. While a confusion martix is always valid for a certain cut-value p^*, the ROC-curve gives a summary of the classification tables over the possible values of p^*. On the example figure of the ROC-curve (see figure 2) we see the ROC-curve itself, which represents the probability curve, and the AUC (Area under the Curve).

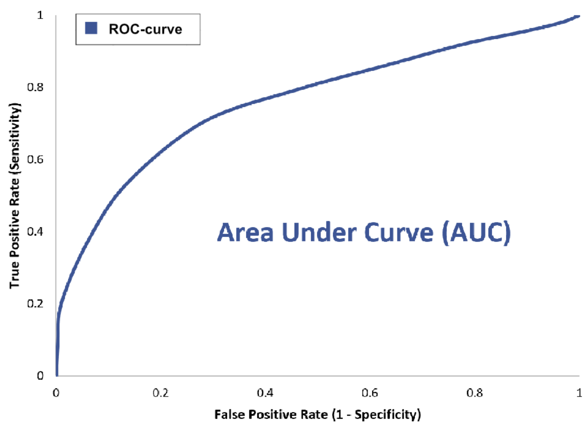

Figure 2: ROC-curve with labeled AUC

The AUC is a measure of the quality of the model's predictive or classification ability. Its maximum is one. There are guideline values which apply for the assessment of the forecast quality expressed by the ROC-curve (see equation 8).

$$
\begin{aligned}
AUC < 0.6 \quad & Unsatisfactory \\
0.6 \leq AUC < 0.7 \quad & Satisfactory \\
0.7 \leq AUC < 0.8 \quad & Good \\
0.8 \leq AUC < 0.9 \quad & Verygood \\
AUC \geq 0.9 \quad & Excellent
\end{aligned} \tag{8}
$$

Another point which is part of the ROC-curve is specificity and sensitivity. This two terms are important for understanding how the ROC-curve is calculated. The sensitivity (y-axes) is the true positive rate, which is a measure of right predicted positive classes (class 1 in a binary case) divided by sum of true positive and false negative predictions. The specificity is the proportion of true negative and the sum of true negative and false positive predictions. For the x-axes the false positive rate is used, which is the substraction of 1 and the specificity. The false positive rate indicates how often a negative group (class 0 in a binary case) will be classified as positive (class 1 in a binary case).

3.4. Similarity and differences

Logistic regression is similar in terms of the problem to discriminant analysis, which was discussed in the previous sections. Both methods are based on a linear model and they are indeed used alternatively. Instead of states or events, we had spoken of separation groups in discriminant analysis.

The essential difference between the two methods is that the logistic regression directly provides probabilities for the occurrence of the alternative events or the memberships to the individual classes. In contrast, the discriminant analysis provides discriminant values, from which the probabilities for group membership can then be calculated in a separate step.

The logistic regression is computationally somewhat more complex, since the estimation of the parameters requires the application of the maximum likelihood method and thus an iterative procedure. In terms of the statistical properties of the methods, one advantage of the logistic regression is that it is based on fewer assumptions about the data used than the discriminant function and is therefore more robust with respect to the data material and, in particular, less sensitive to gross outliers. However, experience shows that both methods provide similarly good results, especially for large samples, even in cases where the assumptions of the discriminant analysis are not met.

3.5. Software and packages

In comparison to the preparation and cleaning of the dataset, which was done with python, the statistical analysis has been done with the programming language *R*. *R* is a programming language which was designed for statistical computing and graphics. Along with the base environment of *R*, I used 4 different packages, as (1) tidyverse, which is a collection of packages like dplyr, tidyr, stringr, lubridate all used for data wrangling and different data operations with arrays or tables, (2) dplyr, which is also a part of tidyverse and an essential tool of shortcuts for subsetting, summarizing, rearranging, and joining together datasets, (3) MASS, which stands for Modern Applied Statistics with S, is used for the disriminant analysis and logistic regression, and (4) pROC for calculating and plotting the ROC-curve and AUC.

4. Statistical Analysis

Now that the theoretical methods and R packages which will be used for predicting an round winner in Counter-Strike: Global Offensive are defined, this section focuses on sharing the results on the statistical analysis. In the R file *CSGO statistical analysis.r* the dataset has been scaled for the 91 independent variables and divided into training (80% of the dataset) and testing samples (20% of the dataset) with a random set seed of 1.

Firstly, from the discriminant analysis output from R, we get the prior probabilities of 0.49 for the group 0 (CT) and 0.51 for the group 1 (T).

Secondly, we observe the coefficients of linear discriminants, which are used for calculating the discriminant score D with the equation from the third section (see equation 2). Only one discriminant function was computed, because there were only two groups (Counter-terrorists and terrorists).The exact coefficients for the discriminant function can be observed in the appendix (see appendix table 6 (part 1) and 7 (part 2)).

Thirdly, for the discriminant model we use the testing sample to evaluate the performance. The confusion matrix (see table 3) gives an accuracy of 0.752907 or 75.2%.

	Actual 0	Actual 1
Predicted 0	9138	3187
Predicted 1	2848	9251

Table 3: Confusion matrix for discriminant analysis

abbr.: actual 0 = actual round wins of counter-terrorists; actual 1 = actual round wins of terrorists
predicted 0 = predicted round wins of counter-terrorists; predicted 1 = actual round wins of terrorists

For the logistic regression we get the deviance residuals (see table 4). And the estimators for the coefficients, their standard errors, z-value and p-value. The exact information for each variable can be observed in the appendix (see appendix table 8 (part 1) and 9 (part 2)).

Min	25%	50%	75%	Max
-3.06398	-0.82959	0.08681	0.79430	3.15703

Table 4: Deviance residuals of the logistic regression
abbr.: Min = minimum value; 25% = .25 quantile;
50% = .5 quantile; 75% = 0.75 quantile; max = maximum values

As for the discriminant model, for the logistic regression model an confusion matrix was also calculated (see table 5). The confusion matrix gives an accuracy of 0.7529889 or 75.30%.

	Actual 0	Actual 1
Predicted 0	9044	2942
Predicted 1	3091	9347

Table 5: Confusion matrix for logistic regression
abbr.: actual 0 = actual round wins of counter-terrorists; actual 1 = actual round wins of terrorists
predicted 0 = predicted round wins of counter-terrorists; predicted 1 = actual round wins of terrorists

The ROC-curve of the CSGO data (see figure 3) shows the relationship between the true positive rate and the false positive rate. The AUC (area under the curve) is 0.851 and shows a very good forecast quality expressed by the ROC-curve

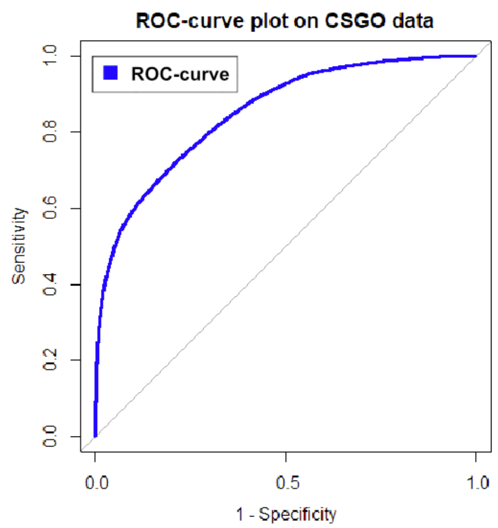

Figure 3: ROC-curve plot on CSGO data

5. Summary

In this paper we investigated how discriminant analysis and logistic regression are performed, what the theory behind his and how to predict an round winner in Counter-Strike: Global Offensive. The objective was to compare both models and show the differences on the classification on big sets of variables and observations.

In section 3.4 we stated that both statistical methods are used alternatively. On the CSGO data both models performed with nearly equal accuracy and complement each other. In other areas or with other data material sometimes one method shows significantly better results. This is why it is suggested not to use one model, but apply different models and compare them with each other.

The variables have different weights and can help to understand what influences an round outcome. This can be the first step for planning the strategy, what weapons to use or on which maps to focus. Although, statistical measures support the decision making of the teams, every new idea and strategy has to be tested in the game itself and against various opponents.

For future analysis and researches it would be interesting to analyze and predict round outcomes with more variables, like athletes career duration, age or, as example, most used weapon by a player. Another approach would be to apply other statistical techniques or even artificial intelligence.

A. Appendix

Variable (A)	Coefficinets LD1
X	-0.0133883475
time_left	0.0987659094
ct_score	-0.0158064487
t_score	0.0082459901
map	-0.0526200770
bomb_planted	0.1642088735
ct_health	-0.2960620039
t_health	0.2182075179
ct_armor	-0.3727365588
t_armor	0.3816095225
ct_money	-0.1422984723
t_money	0.1098278152
ct_helmets	0.0160791580
t_helmets	0.0940687079
ct_defuse_kits	-0.0146239370
ct_players_alive	-0.1300795310
t_players_alive	0.1479829265
ct_weapon_ak47	-0.1296115407
t_weapon_ak47	0.4240876802
ct_weapon_aug	-0.1386297412
t_weapon_aug	0.0249210947
ct_weapon_awp	-0.2180756501
t_weapon_awp	0.1672733960
t_weapon_bizon	-0.0009817992
ct_weapon_cz75auto	0.0033442069
t_weapon_cz75auto	0.0325485640
ct_weapon_elite	-0.0109989556
t_weapon_elite	0.0106851682
ct_weapon_famas	-0.0929309382
t_weapon_famas	0.0320166503
t_weapon_g3sg1	0.0118421427
ct_weapon_galilar	-0.0353909853
t_weapon_galilar	0.0901602236
ct_weapon_glock	0.0132191415
t_weapon_glock	0.0294052551
ct_weapon_m249	0.0067331781
ct_weapon_m4a1s	-0.0659414064
t_weapon_m4a1s	0.0272450019
ct_weapon_m4a4	-0.3223938189
t_weapon_m4a4	0.0728965159
ct_weapon_mac10	-0.0083290602
t_weapon_mac10	0.0732303315
ct_weapon_mag7	-0.0114965081
t_weapon_mag7	0.0048753045
ct_weapon_mp5sd	-0.0050507215
t_weapon_mp5sd	0.0257268194

Table 6: LD1 Coefficients part 1

Variable	Coefficinets LD1
ct_weapon_mp7	0.0027153463
t_weapon_mp7	-0.0033082283
ct_weapon_mp9	-0.0819269615
t_weapon_mp9	0.0218478075
t_weapon_negev	-0.0047323689
ct_weapon_nova	-0.0068935173
t_weapon_nova	-0.0050927800
ct_weapon_p90	-0.0108694178
t_weapon_p90	0.0119152568
t_weapon_r8revolver	-0.0036216209
t_weapon_sawedoff	0.0084788355
ct_weapon_scar20	0.0068002765
t_weapon_scar20	-0.0071945488
ct_weapon_sg553	-0.1448687087
t_weapon_sg553	0.3762212148
ct_weapon_ssg08	-0.0570290496
t_weapon_ssg08	0.0180856979
ct_weapon_ump45	-0.0346217869
t_weapon_ump45	0.0544669653
ct_weapon_xm1014	-0.0130428872
t_weapon_xm1014	0.0000596656
ct_weapon_deagle	0.0095153043
t_weapon_deagle	0.0624803380
ct_weapon_fiveseven	0.0208993515
t_weapon_fiveseven	-0.0167858437
ct_weapon_usps	0.0713885048
t_weapon_usps	0.0560889102
ct_weapon_p250	0.0364552463
t_weapon_p250	0.0359853507
ct_weapon_p2000	0.0374229778
t_weapon_p2000	0.0221654586
ct_weapon_tec9	-0.0062199578
t_weapon_tec9	0.0143849093
ct_grenade_hegrenade	-0.0053675438
t_grenade_hegrenade	-0.0085450972
ct_grenade_flashbang	0.0285413278
t_grenade_flashbang	-0.1325664959
ct_grenade_smokegrenade	0.0411417671
t_grenade_smokegrenade	-0.1467820128
ct_grenade_incendiarygrenade	0.0731313112
t_grenade_incendiarygrenade	-0.0099984453
ct_grenade_molotovgrenade	0.0038472521
t_grenade_molotovgrenade	-0.0828596860
ct_grenade_decoygrenade	0.0046890918
t_grenade_decoygrenade	0.0205078960

Table 7: LD1 Coefficients part 2

Variables	Estimate	Std. Error	z value	$Pr(>\|z\|)$	Signif. codes
(Intercept)	0.121976	0.009817	12.425	< 2e-16	***
X	-0.019382	0.008330	-2.327	0.019985	*
time_left	0.082985	0.019124	4.339	1.43e-05	***
ct_score	-0.020416	0.011263	-1.813	0.069888	.
t_score	0.010437	0.011167	0.935	0.349961	
map	-0.076506	0.008251	-9.272	< 2e-16	***
bomb_planted	0.330855	0.012948	25.553	< 2e-16	***
ct_health	-0.622128	0.043566	-14.280	< 2e-16	***
t_health	0.446476	0.038160	11.700	< 2e-16	***
ct_armor	-0.492807	0.019244	-25.609	< 2e-16	***
t_armor	0.487561	0.022790	21.394	< 2e-16	***
ct_money	-0.177564	0.013011	-13.648	< 2e-16	***
t_money	0.140659	0.012420	11.325	< 2e-16	***
ct_helmets	-0.013544	0.017856	-0.759	0.448135	
t_helmets	0.285940	0.035764	7.995	1.29e-15	***
ct_defuse_kits	-0.024207	0.016154	-1.499	0.133997	
ct_players_alive	-0.412973	0.091623	-4.507	6.57e-06	***
t_players_alive	0.551772	0.083708	6.592	4.35e-11	***
ct_weapon_ak47	-0.196985	0.011211	-17.571	< 2e-16	***
t_weapon_ak47	0.574455	0.024060	23.875	< 2e-16	***
ct_weapon_aug	-0.213286	0.009802	-21.759	< 2e-16	***
t_weapon_aug	0.028940	0.008596	3.367	0.000761	***
ct_weapon_awp	-0.328604	0.012151	-27.042	< 2e-16	***
t_weapon_awp	0.221848	0.011384	19.488	< 2e-16	***
t_weapon_bizon	0.001520	0.008071	0.188	0.850638	
ct_weapon_cz75auto	-0.011656	0.031426	-0.371	0.710702	
t_weapon_cz75auto	0.054185	0.024956	2.171	0.029914	*
ct_weapon_elite	-0.016206	0.008875	-1.826	0.067849	.
t_weapon_elite	0.013273	0.008378	1.584	0.113152	
ct_weapon_famas	-0.147473	0.009328	-15.809	< 2e-16	***
t_weapon_famas	0.046305	0.009594	4.826	1.39e-06	***
t_weapon_g3sg1	0.014120	0.007253	1.947	0.051549	.
ct_weapon_galilar	-0.056132	0.008855	-6.339	2.31e-10	***
t_weapon_galilar	0.120032	0.010504	11.427	< 2e-16	***
ct_weapon_glock	0.021033	0.009607	2.189	0.028582	*
t_weapon_glock	0.011508	0.100250	0.115	0.908607	
ct_weapon_m249	0.033901	0.562972	0.060	0.951982	
ct_weapon_m4a1s	-0.102599	0.008391	-12.228	< 2e-16	***
t_weapon_m4a1s	0.039682	0.009183	4.321	1.55e-05	***
ct_weapon_m4a4	-0.502000	0.017152	-29.267	< 2e-16	***
t_weapon_m4a4	0.085155	0.009347	9.111	< 2e-16	***
ct_weapon_mac10	-0.001453	0.008419	-0.173	0.863010	
t_weapon_mac10	0.103218	0.011528	8.954	< 2e-16	***
ct_weapon_mag7	-0.022437	0.007672	-2.924	0.003452	**
t_weapon_mag7	0.016566	0.022688	0.730	0.465286	
ct_weapon_mp5sd	-0.007452	0.007860	-0.948	0.343101	
t_weapon_mp5sd	0.044157	0.010420	4.238	2.26e-05	***

Table 8: Logistic regression output part 1
abbr.: Std. Error = standard error; Signif. codes = significance codes
levels of significance: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

Variables	Estimate	Std. Error	z value	$Pr(>\|z\|)$	Signif. codes
ct_weapon_mp7	0.003568	0.008152	0.438	0.661655	
t_weapon_mp7	-0.008310	0.008110	-1.025	0.305530	
ct_weapon_mp9	-0.144915	0.010473	-13.837	< 2e-16	***
t_weapon_mp9	0.026658	0.009661	2.759	0.005791	**
t_weapon_negev	-0.031563	0.562972	-0.056	0.955290	
ct_weapon_nova	-0.014753	0.007750	-1.904	0.056961	.
t_weapon_nova	-0.008726	0.009197	-0.949	0.342730	
ct_weapon_p90	-0.015967	0.009274	-1.722	0.085113	.
t_weapon_p90	0.013659	0.007470	1.829	0.067460	.
t_weapon_r8revolver	-0.073215	0.530192	-0.138	0.890168	
t_weapon_sawedoff	0.009795	0.007376	1.328	0.184204	
ct_weapon_scar20	0.009467	0.007626	1.241	0.214430	
t_weapon_scar20	-0.034432	0.562972	-0.061	0.951231	
ct_weapon_sg553	-0.226456	0.011343	-19.964	< 2e-16	***
t_weapon_sg553	0.512515	0.020748	24.702	< 2e-16	***
ct_weapon_ssg08	-0.105733	0.009287	-11.386	< 2e-16	***
t_weapon_ssg08	0.027457	0.008620	3.185	0.001446	**
ct_weapon_ump45	-0.059295	0.008622	-6.877	6.10e-12	***
t_weapon_ump45	0.076670	0.009453	8.110	5.05e-16	***
ct_weapon_xm1014	-0.023748	0.007895	-3.008	0.002629	**
t_weapon_xm1014	-0.003906	0.008431	-0.463	0.643117	
ct_weapon_deagle	0.001980	0.057300	0.035	0.972429	
t_weapon_deagle	0.094066	0.051192	1.838	0.066133	.
ct_weapon_fiveseven	0.022391	0.021279	1.052	0.292664	
t_weapon_fiveseven	-0.019422	0.010819	-1.795	0.072634	.
ct_weapon_usps	0.132444	0.107538	1.232	0.218099	
t_weapon_usps	0.066049	0.028622	2.308	0.021021	*
ct_weapon_p250	0.062975	0.037300	1.688	0.091341	.
t_weapon_p250	0.037045	0.036818	1.006	0.314327	
ct_weapon_p2000	0.060192	0.029766	2.022	0.043162	*
t_weapon_p2000	0.033776	0.010842	3.115	0.001838	**
ct_weapon_tec9	-0.010471	0.010777	-0.972	0.331269	
t_weapon_tec9	0.020381	0.014365	1.419	0.155957	
ct_grenade_hegrenade	0.011917	0.013152	0.906	0.364893	
t_grenade_hegrenade	-0.026664	0.009444	-2.823	0.004752	**
ct_grenade_flashbang	0.036154	0.018120	1.995	0.046014	*
t_grenade_flashbang	-0.186880	0.018046	-10.356	< 2e-16	***
ct_grenade_smokegrenade	0.072684	0.017904	4.060	4.92e-05	***
t_grenade_smokegrenade	-0.235783	0.017666	-13.346	< 2e-16	***
ct_grenade_incendiarygrenade	0.142120	0.015327	9.273	< 2e-16	***
t_grenade_incendiarygrenade	-0.021954	0.008620	-2.547	0.010868	*
ct_grenade_molotovgrenade	0.016171	0.008641	1.871	0.061282	.
t_grenade_molotovgrenade	-0.155460	0.016095	-9.659	< 2e-16	***
ct_grenade_decoygrenade	0.008083	0.007585	1.066	0.286604	
t_grenade_decoygrenade	0.026123	0.007577	3.448	0.000566	***

Table 9: Logistic regression output part 2
abbr.: Std. Error = standard error; Signif. codes = significance codes
levels of significance: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

B. References

[1] Fahrmeir, L., Hamerle, A., Tutz, G. (1996). *Multivariate statistische Verfahren.* 2nd Edition. de Gruyter, Berlin. *Chapter 8: Diskriminanzanalyse*: 357-435

[2] Backhaus, K., Erichson, B., Plinke, R., Weiber, R. (2018). *Multivariate Analysemethoden - Eine anwendungsorientierte Einfuehrung.* 15th Edition. *Chapter 4: Diskriminanzanalyse*: 215-282

[3] Backhaus, K., Erichson, B., Plinke, R., Weiber, R. (2018). *Multivariate Analysemethoden - Eine anwendungsorientierte Einfuehrung.* 15th Edition. *Chapter 5: Logistische Regression*: 283-356

[4] Rencher, A.C., Christensen, W.F. (2012) *Methods of multivariate analysis.* 3rd Edition, Wiley. *Chapter 8: Discriminant Analysis: Description of Group Separation*: 270-298

[5] Rencher, A.C., Christensen, W.F. (2012) *Methods of multivariate analysis.* 3rd Edition, Wiley. *Chapter 9: Classification Analysis: Allocation of Observations to Groups*: 299-321