A Reproduction Report of: "Civil War Exposure and Violence" by Edward Miguel, Sebastian M. Saiegh & Shanker Satyanath *

Colin Asbill, Chloe Byun, Emma Eeckhout & Christian Stec

April 26, 2024

Abstract

"Civil War Exposure and Violence" explores the psychological and health consequences of civil war exposure in the setting of major professional soccer leagues. Using a negative binomial regression analysis, the authors found a positive correlation between the number of years of civil war in a player's native country and the number of yellow and red cards earned by that player. Their findings suggest that personally experiencing civil conflict is correlated with violent behavior with a predicted 0.0076 increase in yellow cards per year of civil war. Furthermore, for every 1 unit increase in years of civil war, there is an even greater coefficient estimate (0.0126) for red cards given. This estimates are robust to removing outlier countries.

The positive correlation between violence and exposure to civil war seems to be limited to civil conflict post-birth, as civil conflict pre-birth did not have a statistically significant correlation with violence in regressions 6 and 7 of Table 2. We were able to computationally reproduce these results below, other than regression 5, which we were not able to reproduce as see in the Miguel, Et. Al. paper. This most likely is a result of reproducing the regressions in Python and not accounting for certain data within the negative binomial regression in Stata by default.

*Authors: Colin Asbill, Chloe Byun, Emma Eeckhout, Christian Stec. Professor: Eric Van Dusen. Student Instructors: Rohan Jha and Peter Flo Grinde-Hollevik. Econ 148 at the University of California, Berkeley.

1 Introduction

The adverse consequences of exposure to civil war have been largely studied in recent years in the form of survey questioning and analysis. Although helpful in demonstrating the negative impact on physical and psychological health, it is not as useful to examine changes in behavior. Additionally, researchers cannot simply compare violence across countries to determine a relationship between conflict exposure and violent behaviors due to the inconsistency of institutions and laws between nations. Furthermore, data collection is unreliable and the standards and quality of violence reporting are directly related to the level of violence within a country. For these reasons, it is difficult to shift from survey analysis to actual behavior analysis when studying this topic.

Miguel Et. Al. attempts to overcome these challenges by investigating the relationship between civil war exposure and violence levels in the context of professional soccer. The authors focus on 6 major European Football leagues with players from 70 countries. These countries experience a range of internal conflicts, from no civil war to continuous civil war between the years 1980 and 2005. The empirical evidence to measure violent behavior is represented by the number of yellow and red cards given to each player. The main dataset was a combination of information from multiple sources to produce the soccer_data dataset that contains soccer statistics, notably yellow cards and player salaries, as well as civil war statistics. The proxy measure for exposure to violence is the number of years a player's native country was involved in a civil war from 1980 to 2005. The analysis reveals a notable empirical trend: "There is a strong correlation between the number of years of civil conflict in a player's native country and his likelihood of earning yellow and red cards in Europe. This main result is robust to extensive controls for player and country characteristics and team and continent fixed effects, where we effectively compare nearby countries" (Miguel et al. 60). The results are statistically significant due to fixed effects (player, country characteristics, team/continent, civil wars before birth, etc) that will be discussed further in the report.

In this reproduction report, we hope to replicate the research done by Miguel et al. and determine whether his paper is indeed reproducible and accurate. We attempt to do this by using the replication package posted on the Harvard Dataverse. We downloaded the datasets, files, and code done by the original authors to a joint Google Drive folder and reproduced the code on Google Colab. We then used our models to explore the paper's main findings and conclusions on the relationship between yellow cards, violent behavior, and civil war exposure. We also reproduced the Table of Descriptive Statistics (Table 1) for each of the variables that would later be used in the Regression Table (Table 2).

We found that the last table had a few coding errors which caused small discrepancies in regression 5 of Table 2, specifically with the addition of the "Goalie" variable. Finally, we successfully reproduced the figures that represented each regression with ease, although there may be minor differences when representing regression 5.

2 Reproduction: Data & Estimation

The data frame used in this paper contains information from the 2004–2005 and 2005–2006 soccer seasons in 5 national leagues (England, France, Germany, Italy, and Spain) and 1 supranational league (the UEFA Champions League) from the ESPN *Soccernet* website. Weekly player salary and transfer fee information in 2011 USD came from *Football Manager*, 2005 and *World Soccer Manager*, 2006. Civil war information was found in the PRIO/Uppsala Armed Conflict database, while the rule of law variable came from the WGI project. Finally, income per capita data was pulled from the World Bank's *World Development Indicators* (2007). All of these sources were combined into onedata set, which was provided by Miguel Et. Al. for reproduction streamlining the reproduction process, since combining multiple datasets can be extremely challenging. To ensure at least a moderate amount of information per country, the authors only considered countries with 5 or more player seasons represented in their sample. While most players in the European leagues are from wealthy OECD countries, many are also from Africa, Eastern Europe, and Latin America, as well as a smaller number of Asians. In cases of dual nationality or lack of information, Miguel used what national team they played on; this is limited because it could lead to an underestimation of the effect of civil war on players who play for developed (European) countries.

The professional European soccer league setting offers the study a "common institutional backdrop" that allowed the authors to accurately isolate the effect of civil war exposure on violent behavior. Yellow cards are given for excessively violent fouls; swearing at an opponent; humiliating the opponent after scoring a goal (with excessive celebration); "diving" to falsely pin a foul on an opponent; or disobeying the referee's instructions, among other behaviors (including time wasting). Red cards are used as a secondary measure since it is less frequent.

Turning our attention to our replication and the process that led us to similar results to the authors, we imitated Figures 1 and Table 1 with relative ease. Later, through analysis of the soccer data, we noticed the dependent variable, yellow card count, cannot be a negative number, hence a count model was introduced following a negative binomial regression.

Figures 1/2: We started by replicating Figure 1A/B, which helped to determine whether yellow cards were in fact due to aggressive and violent behavior. In their replication package, the authors left two very simple tables in Excel format, only with the resulting data that would be attributed to each figure. In this case, the coding was simple, since we only had to upload these tables to our Google Colab notebook and then code the data into two bar charts. The final result is identical to that of the authors, and we were able to visualize that yellow cards are attributed to "Assault", meaning violent behavior. Figure 1, represented as Figure 1A in the original paper, represents the Italian league during the 2005–2006, 2006–2007, and 2007–2008 seasons in which nearly three-quarters of yellow cards were due to "Assault." Figure 2, represented as Figure 1B in the original paper, represents the

Union of European Football Associations (UEFA) Champions League in 2004–2005 and 2005–2006 where the proportion is around two-thirds.

We decided to recreate the Python graphs in LaTeX for by feeding Python code into ChatGPT and with some trial and error. The first few attempts were unsuccessful because it was trying to use the soccer_data dataset that was not in the LaTeX file resulting in a blank graph. This was resolved by having LaTeX take in the right bar heights using add plot coordinates. We then added a legend, title, and separate colors for each type of yellow card. The figures are not exactly the same as in the paper because the paper labeled the figures as 1A/B, while we have them labeled as Figure 1 and Figure 2. This is because we inserted them as separate figures in LaTeX.

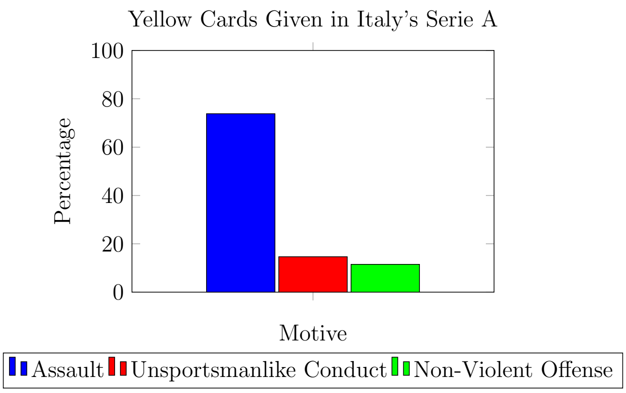


Figure 1: Source: Research Department of Lega Nazionale Professionisti (Italy)

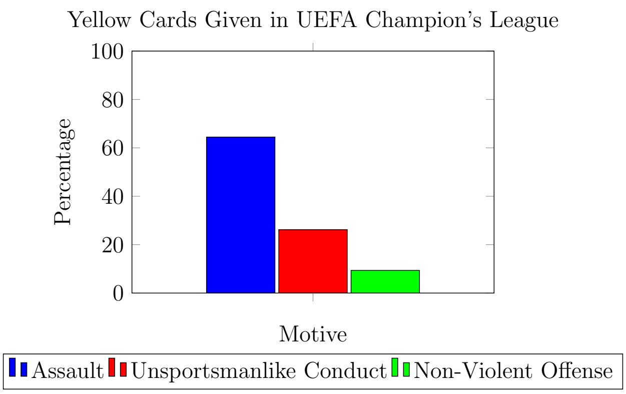


Figure 2: Source: UEFA

Table 1: Table 1 acts as a starting point for our reproduction given that it contains the descriptive statistics of each variable. The main error we encountered was that our observations were higher than those of the original paper. Initially, we thought we had not filtered correctly for the amount of countries because each country needed to have at least 5 player seasons included in the data table. We still had too many observations but after some trial and error, we realized that our data table was including players that had missing log transfer fee values. Filtering out missing log transfer fee values fixed the problem and the data table finally had the same number of observations as Table 1.

There are a few differences between Table 1 and Table 1 in the original paper. We assume that these differences are caused by the way Python rounds significant figures compared to Stata. For example, in the GNI per capita (2006) row, our Table 1 has the mean as 26,203.86, while the original has 26,203. In the research paper, there is also a note attached that explains the origin of the data coming from multiple sources.

Table 1: Descriptive Statistics

Variable	Mean	Standard deviation	Minimum	Maximum	Observations
Rule Infractions					
Yellow cards per player-season	2.43	2.73	0	16	5035
Red cards per player-season	0.16	0.42	0	3	5035
Country Characteristics					
Years of civil war (1980–2005)	2.74	4.74	0	26	5035
Rule of law (2005–2006)	0.85	0.89	-1.76	2.10	5035
GNI per capita (2006)	26203.86	10923.41	720	44260	4965
Player Characteristics					
Age	25.99	4.40	17	41	5035
Weekly salary (in '000 USD)	23.99	27.01	0	190	5034
Transfer fee (in '000 USD)	6323.51	8198.48	3	78000.06	5035
Games Started	13.80	11.49	0	40	5035
Substitute	3.13	3.89	0	29	5035
Goalie	0.08	0.27	0	1	5035
Defender	0.33	0.47	0	1	5035
Forward	0.23	0.42	0	1	5035
Midfield	0.36	0.48	0	1	5035
Goals scored per player-season	1.65	3.12	0	31	5035
Player Region of Origin					
Africa	0.07	0.26	0	1	5035
Asia	0.004	0.06	0	1	5035
Latin America/Caribbean	0.12	0.33	0	1	5035
Eastern Europe	0.07	0.25	0	1	5035
OECD	0.72	0.45	0	1	5035
Soccer Leagues					
English League	0.17	0.38	0	1	5035
European Champions League	0.19	0.39	0	1	5035
French League	0.15	0.36	0	1	5035
German League	0.14	0.35	0	1	5035
Italian League	0.17	0.38	0	1	5035
Spanish League	0.16	0.37	0	1	5035

3 Reproduction: Empirical Results

Table 2: The aim of Table 2 was to explore the influence of civil conflict on yellow cards, red cards, and goals scored with a series of different regressions to test the robustness of the results. Table 2 is comprised of a series of 7 regressions using the data from Table 1. As described above, we encountered multiple challenges and method-based decisions that we had to solve in order to reproduce Table 1 and prepare the data for Table 2.

The regression model employed was the negative binomial regression, chosen for its suitability in handling count data (cards issued, goals scored). The models were adjusted for clustering at the national level to account for correlated observations within the same country. Any challenges with syntax and variable encoding were resolved by using the Patsy library to properly format model formulas.

Finally, there were additional complexities regarding integrating league-specific dummy variables and regional indicators. Ensuring the inclusion of all relevant predictors and interactions, while managing data sparsity in categories like "Goalie," was particularly demanding.

The biggest issue we ran into while creating Table 2 was in the 5th regression, with the inclusion of the "Goalie" variable. When trying to run the regression with the same type of code as the previous regressions, we received many warnings about dividing by zero and failing to invert the hessian. This was also resulting in NaN values for all of the standard errors, and coefficients that were very dissimilar to the ones from the original paper. After researching the issue and the differences between Stata and Python, we believe that a potential cause is the different way that these two programming languages handle negative binomial regression, specifically regarding zero-inflation. Our best theory of why the regression was not working was that the "Goalie" variable is mostly zeros, with just 8 percent of the values being ones. Normal negative-binomial models may struggle with this and one may need to introduce a zero-inflated model to better model this. Stata's nbreg automatically handles zero inflation, but statsmodel does not and we struggled to implement this. Finally, after trying a different negative binomial regression API within statsmodels, we were able to get the coefficients and standard errors to be very similar to the original paper, all except for the "Goalie" variable. The failure to invert Hessian warning remains, and it is likely still due to the sparseness of this variable as well as its high correlation with the endogenous variable.

This led to our reproduction of regression 5 to be different from the paper with our "Goalie" variable value as -52.4183 instead of -18.216. There are also some slight rounding differences like those seen in Table 1, which caused a few results to be less statistically significant such as Years of Civil War in regression 1. The standard errors we have in parentheses are completely different compared to the paper. It mentions that Z-statistics are in parentheses, so it is possible that they are representing something other than the standard error; if the standard errors were far off from the estimates, statistical significance in our reproduced regressions would not be similar to the results seen in Table 2.

Table 2 is the cornerstone of the Miguel Et. Al Civil War paper as it displays all of the regressions in the paper. These regressions show a positive correlation between violence and civil war exposure by showing a statistically significant increase in violent behavior (Yellow Cards/Red Cards) with predicted yellow cards increasing 0.0075 per unit increase of years of civil war after filtering out outlier peaceful and violent countries in regression 3. There is an even greater increase in red cards of 0.0126 found in regression 4. Regressions 6 and 7 show that this trend is limited to civil war years post-birth, with years pre-birth having a negligible non-statistically significant effect.

Figure 3: The aim of Figure 3 (Figure 2 in the original paper) was to combine soccer player statistics with the World Bank country codes dataset to visualize the hypothesis of the paper: the relationship between civil war exposure and violent behavior. To do this, we plotted the number of our outcomes conditional on control variables in Table 2, regression 1 for all countries.

Although this was simply a visualization of the previous table, we still ran into a couple of challenges. Our toughest struggle was realizing that the aim was not to predict the total number of yellow cards but to find the average per player season. The correct approach was to use averages to standardize outcomes across countries that had different player-season counts. Additionally, while coding the plotting aspect of the graph, we assumed we had to use the same regression calculated in Table 2, but had to run a new one. Finally, we were confused about the CIV circle at the top of our graph given that it is far from that location in the graph in the paper. We realized it was the country code for the Côte d'Ivoire (Ivory Coast) which is included in the data of the paper. In Table 1A, the authors have 49 observations with 3.26 yellow cards on average, as well as 3 years of civil war exposure. We do not understand why the circle for CIV is so far off from where it is in the original

Table 2: Empirical Results: Civil War

	Yellow cards	Yellow cards	Yellow cards	Red cards	Goals scored	Yellow cards	Red cards
Years of civil war	0.008** (0.003)	0.008* (0.003)	0.008** (0.003)	0.013 (0.007)	8.995e-05 (0.005)		
Civil war years post-birth						0.005 (0.003)	0.014* (0.007)
Civil war years pre-birth						0.004 (0.005)	-0.004 (0.010)
Log GNI per capita		0.047 (0.044)					
Rule of Law			-0.020 (0.049)	-0.143 (0.098)	0.0061 (0.042)		
Age	0.013*** (0.002)	0.013*** (0.002)	0.013*** (0.002)	0.013 (0.007)	0.0205*** (0.006)	0.013*** (0.002)	0.011 (0.007)
Log transfer fee	0.032* (0.014)	0.031* (0.014)	0.033* (0.014)	0.063* (0.030)	0.3222*** (0.027)	0.032* (0.014)	0.063* (0.030)
Games Started	0.068*** (0.002)	0.068*** (0.002)	0.068*** (0.002)	0.051*** (0.003)	0.0869*** (0.002)	0.068*** (0.002)	0.051*** (0.003)
Substitute	0.040*** (0.004)	0.040*** (0.004)	0.040*** (0.004)	0.011 (0.012)	0.0689*** (0.005)	0.041*** (0.004)	0.011 (0.012)
Defender	1.715*** (0.117)	1.714*** (0.116)	1.714*** (0.117)	1.113*** (0.156)		1.715*** (0.117)	1.120*** (0.155)
Forward	1.397*** (0.127)	1.399*** (0.126)	1.396*** (0.127)	0.720*** (0.181)	1.6466*** (0.078)	1.397*** (0.127)	0.726*** (0.180)
Midfield	1.729*** (0.137)	1.728*** (0.137)	1.729*** (0.137)	0.889*** (0.201)	0.6796*** (0.060)	1.729*** (0.137)	0.892*** (0.199)
Goalie					-52.4183*** (0.336)		
Goals	-0.022*** (0.004)	-0.023*** (0.004)	-0.022*** (0.004)	-0.028*** (0.008)		-0.022*** (0.004)	-0.028*** (0.008)
European Champions League	-0.030 (0.060)	-0.023 (0.061)	-0.036 (0.058)	-0.504* (0.208)	0.2112* (0.086)	-0.028 (0.062)	-0.455* (0.205)
French League	0.265*** (0.060)	0.266*** (0.062)	0.260*** (0.062)	0.297** (0.114)	0.0782 (0.063)	0.263*** (0.064)	0.334** (0.114)
German League	0.318*** (0.048)	0.321*** (0.049)	0.317*** (0.048)	0.097 (0.155)	0.2440*** (0.059)	0.320*** (0.051)	0.111 (0.165)
Italian League	0.353*** (0.054)	0.356*** (0.057)	0.337*** (0.066)	0.629*** (0.138)	-0.0127 (0.057)	0.353*** (0.056)	0.749*** (0.106)
Spanish League	0.544*** (0.050)	0.548*** (0.051)	0.535*** (0.056)	0.648*** (0.106)	0.0020 (0.063)	0.552*** (0.055)	0.719*** (0.108)
Observations	5035	4965	5035	5035	5035	5033	5033
Pseudo R^2	0.190	0.190	0.190	0.107	0.2396	0.190	0.107
Residual Std. Error	1.992 (df=5016)	1.990 (df=4945)	1.992 (df=5015)	0.397 (df=5015)	1.931 (df=5016)	1.991 (df=5013)	0.398 (df=5013)

Note: *p<0.05; **p<0.01; ***p<0.001

paper. In our soccer_data dataframe, there are two names for the same country, Côte d'Ivoire and Ivory Coast, but we have confirmed that these both share the same country code and get mapped to CIV in the merge. Overall, the circles are generally in the right places but are not perfect. We believe that the differences are due to not knowing how they predicted average yellow cards instead of total, as this was not obvious in the provided Stata code.

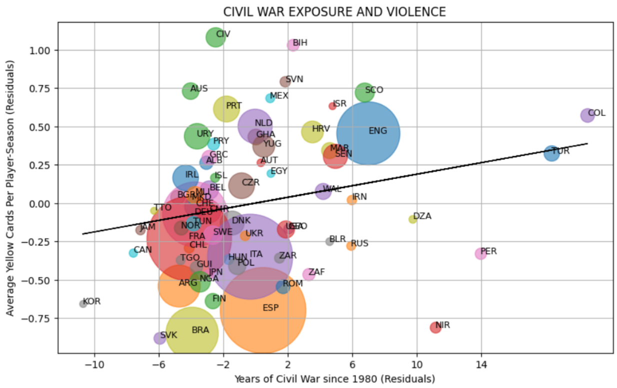


Figure 3: Yellow card sand civil war(conditional on control variables in Table 2, regression 1)– all countries.

Figure 4: This is Figure 3 in the original paper. Similar to the previous figure, we wanted to graph the regression comparing yellow cards to civil war exposure conditional on control variables in Table 2, regression 1 without OECD countries (members of the Organization for Economic Cooperation and Development). In this case, we only had to adjust the previous figure by filtering for non-OECD countries. No real issues were encountered. Filtering out non-OECD countries was important to test the robustness of the trend to outlier OECD countries that had little civil conflict.

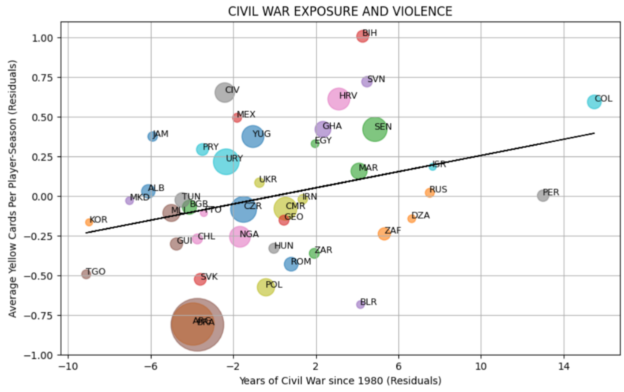


Figure 4: Yellow cards and civil war (conditional on control variables in Table 2, regression 1)non-OECD countries.

Figure 5: Finally, in Figure 5 (Figure 4 in the original paper), we continued using Table 2, regression 1 except this time we filtered for non-OECD countries and also excluded the countries Colombia, Iran, Israel, Peru, and Turkey. These countries were outliers that either had very small or large amounts of civil conflict during this time period. The new graph still shows a positive trend line implying the trend is robust to outliers.

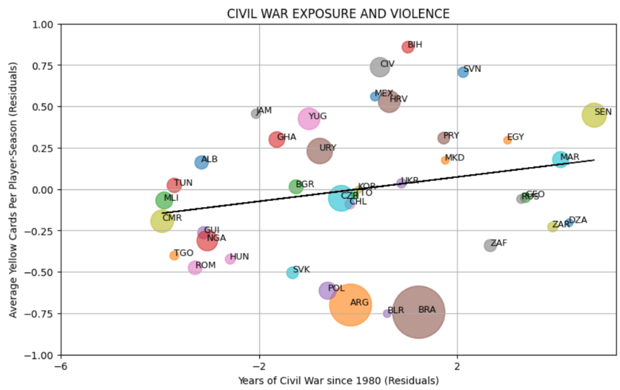


Figure 5: Yellow cards and civil war (conditional on control variables in Table 2, regression 1)non-OECD countries, excluding Colombia, Iran, Israel, Peru, and Turkey.

4 Conclusion

The study by Miguel et al. provides a compelling investigation into the effects of civil war exposure on professional soccer players' behavior, as measured by the issuance of yellow and red cards in major European leagues. This unique approach not only circumvents the typical constraints encountered in survey-based research, which often struggles with unreliable data and varying standards of violence reporting, but it also offers a more direct measure of aggressive behavior in a uniform institutional setting.

Our reproduction efforts, using the original authors' dataset and methodologies from the Harvard Dataverse, reaffirm the findings of Miguel et al.'s investigation. The results are strongly suggestive of a causal link between exposure to civil conflict and subsequent violent behavior among citizens. We successfully reproduced the analytical results and confirmed the main thesis that players from nations with prolonged civil conflicts are more likely to exhibit violent behavior on the field, given the higher issue rates of yellow and red cards. Nevertheless, despite the successful reproduction of most results, several challenges were noted. There was some data inconsistency in which minor discrepancies arose due to initial misunderstandings of the dataset filtering criteria, which was corrected to align with the original study's parameters. Additionally, the choice of model and the formulation of variables required meticulous verification to ensure consistency, specifically for variables with missing data. Still, the successful reproduction of descriptive statistics in visualizations provided clear insights into the data's underlying distributions and helped validate the quantitative analysis.

Finally, while this study provides significant insights, it is not without its limitations. Firstly, the reliance on yellow and red cards to determine aggression could overlook subtler forms of violence. Additionally, the paper assumes that referee discretion is consistent across all leagues and games and free of bias. Realistically, different referees might have different thresholds of what they consider worthy of a yellow/red card, fluctuating by the importance of a game, a reputation of a player, etc.

Further research could incorporate more granular data on a referee's decision-making. Additionally, there could be more research done on other forms of violent behavior on and maybe even off the field (eg. locker rooms and time outs). Furthermore, these athletes retire as young adults, which gives way to a longitudinal study that could assess whether these patterns hold across different generations of players. This paper quantifies the unrealized mental health effects of civil conflict, and we believe that this paper introduces a promising area of interest for future studies.

5 Appendix

We also recreated the first table from the appendix (Table A1). Table A1 was fairly straightforward to recreate in Python and LaTeX because it is just a summary table of the soccer data set. Our table is listed as Table 3 instead of A1 in our reproduction. We did not include the footnotes at the bottom of the table. We were unable to reproduce Table A2 because the Stata code was not provided, so the exact method for removing outliers is unknown.

Table 3: Countries and Players Represented in the Main Sample

Country	Observations	Yellow cards	Civil war years	Country	Observations	Yellow cards	Civil war years
Albania (ALB)	18	2.88	0	Macedonia (MKD)	6	4.16	1
Algeria (DZA)	6	1.50	15	Mali (MLI)	29	3.03	2
Argentina (ARG)	178	2.91	0	Mexico (MEX)	8	3.62	2
Australia (AUS)	28	2.57	0	Morocco (MAR)	26	3.15	10
Austria (AUT)	6	1.66	0	Netherlands (NLD)	118	2.06	0
Belarus (BLR)	6	1.50	0	Nigeria (NGA)	43	1.81	1
Belgium (BEL)	34	1.91	0	Northern Ireland (NIR)	12	1.00	13
Bosnia and Herzegovina (BIH)	14	2.92	4	Norway (NOR)	20	1.75	0
Brazil (BRA)	277	2.44	0	Paraguay (PRY)	14	2.42	1
Bulgaria (BGR)	20	2.55	0	Peru (PER)	13	1.38	19
Cameroon (CMR)	52	2.28	1	Poland (POL)	30	3.00	0
Canada (CAN)	7	3.71	0	Portugal (PRT)	68	3.02	0
Chile (CHL)	10	3.80	0	Romania (ROM)	19	1.21	1
Colombia (COL)	19	4.79	26	Russia (RUS)	8	1.75	13
Congo DR (ZAR)	10	2.50	6	Saudi Arabia (SAU)	37	2.16	13
Croatia (HRV)	48	2.37	3	Senegal (SEN)	59	2.25	10
Czech Republic (CZE)	67	2.34	0	Serbia (SRB)	8	1.75	3
Denmark (DNK)	58	1.84	0	Sierra Leone (SLE)	5	2.00	10
Egypt (EGY)	6	1.00	6	Slovakia (SVK)	14	0.92	0
England (ENG)	402	2.17	13	Slovenia (SVN)	11	1.63	0
Finland (FIN)	24	1.08	0	South Africa (ZAF)	15	1.06	9
France (FRA)	721	2.48	0	South Korea (KOR)	5	1.00	0
Georgia (GEO)	10	3.20	4	Spain (ESP)	742	2.91	5
Germany (DEU)	424	2.00	0	Sweden (SWE)	35	1.77	0
Ghana (GHA)	25	2.40	2	Switzerland (CHE)	49	2.40	0
Greece (GRC)	22	2.13	0	Togo (TGO)	8	0.75	2
Guinea (GIN)	15	2.33	0	Trinidad and Tobago (TTO)	5	0.20	1
Hungary (HUN)	10	0.90	0	Tunisia (TUN)	21	2.33	1
Iceland (ISL)	8	2.00	0	Turkey (TUR)	24	2.25	22
Iran (IRN)	9	2.33	19	Ukraine (UKR)	9	1.44	0
Ireland (IRL)	67	1.89	0	United States (USA)	30	0.96	4
Israel (ISR)	5	4.80	26	Uruguay (URY)	66	2.89	0
Italy (ITA)	730	2.81	0	Wales (WAL)	26	2.19	13
Ivory Coast (CIV)	38	3.63	3				
Japan (JPN)	10	1.50	0				
Jamaica (JAM)	9	1.77	0				

6 References

Miguel, Edward, Sebastian M. Saiegh, and Shanker Satyanath. 2011. “Civil War Exposure and Violence.” Economics and Politics 23 (1): 59-73

In [ ]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

In [ ]:
```
pip install statsmodels stargazer
```

```
Requirement already satisfied: statsmodels in /usr/local/lib/python3.10/dist-packages
(0.14.2)
Collecting stargazer
  Downloading stargazer-0.0.7-py3-none-any.whl (15 kB)
Requirement already satisfied: numpy>=1.22.3 in /usr/local/lib/python3.10/dist-packag
es (from statsmodels) (1.25.2)
Requirement already satisfied: scipy!=1.9.2,>=1.8 in /usr/local/lib/python3.10/dist-p
ackages (from statsmodels) (1.11.4)
Requirement already satisfied: pandas!=2.1.0,>=1.4 in /usr/local/lib/python3.10/dist-
packages (from statsmodels) (2.0.3)
Requirement already satisfied: patsy>=0.5.6 in /usr/local/lib/python3.10/dist-package
s (from statsmodels) (0.5.6)
Requirement already satisfied: packaging>=21.3 in /usr/local/lib/python3.10/dist-pack
ages (from statsmodels) (24.0)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/di
st-packages (from pandas!=2.1.0,>=1.4->statsmodels) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-package
s (from pandas!=2.1.0,>=1.4->statsmodels) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packa
ges (from pandas!=2.1.0,>=1.4->statsmodels) (2024.1)
Requirement already satisfied: six in /usr/local/lib/python3.10/dist-packages (from p
atsy>=0.5.6->statsmodels) (1.16.0)
Installing collected packages: stargazer
Successfully installed stargazer-0.0.7
```

In [ ]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import statsmodels.api as sm
import statsmodels.formula.api as smf
from statsmodels.formula.api import glm
from statsmodels.genmod.families import NegativeBinomial
from patsy import dmatrices
from stargazer.stargazer import Stargazer
from statsmodels.iolib.summary2 import summary_col
```

CIVIL WAR EXPOSURE AND VIOLENCE REPRODUCTION

Table of contents

Data Ingestion

Load in the data and examine it

In [ ]:
```
# Load in soccer data
soccer_data = pd.read_stata("/content/drive/My Drive/Econ 148 Project 3/Soccer_Replica
soccer_data.head()
```

Out[ ]:

	player_id	player_name	war_before	war_after	year	team	nationality	position	age	I
0	2726	Nelson de Jesus Dida	0.0	0.0	2004/05 Statistics	AC Milan	Brazil	G	31	Char I
1	2726	Nelson de Jesus Dida	0.0	0.0	2004/05 Statistics	AC Milan	Brazil	G	31	:
2	2741	Juliano Belletti	0.0	0.0	2004/05 Statistics	Barcelona	Brazil	D	28	S F D
3	2741	Juliano Belletti	0.0	0.0	2004/05 Statistics	Barcelona	Brazil	D	28	Char I
4	2749	Cris	0.0	0.0	2004/05 Statistics	Lyon	Brazil	D	27	I

5 rows × 40 columns

In [ ]:
```
figure1a_data = pd.read_excel("/content/drive/My Drive/Econ 148 Project 3/Soccer_Repli
figure1a_data.head()
```

Out[ ]:

	League	cause	2005-2006	2006-2007	2007-2008	Average	Percent
0	Serie A TIM	Assault	1299	1357.0	1413.0	1356.333333	73.834150
1	Serie A TIM	Unsportsmanlike Conduct	207	320.0	281.0	269.333333	14.661586
2	Serie A TIM	Other Non-Violent	174	235.0	225.0	211.333333	11.504264
3	Serie A TIM	Total	1680	1912.0	1919.0	1837.000000	100.000000
4	NaN	NaN		NaN	NaN	NaN	NaN

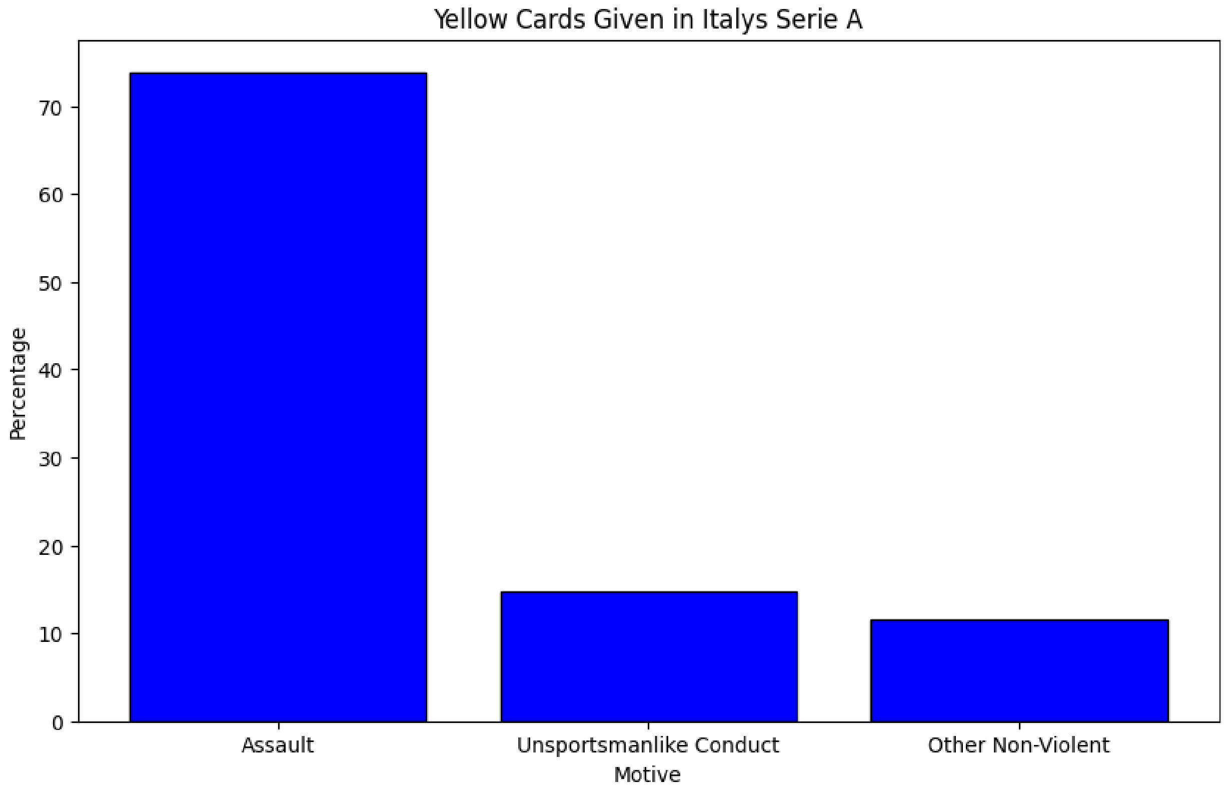
In [ ]:
```
figure1b_data = pd.read_excel("/content/drive/My Drive/Econ 148 Project 3/Soccer_Repli
figure1b_data.head()
```

Out[ ]:

	Motive	2004-2005	2005-2006	Sum	Percent
0	Violent Foul	446	464	910	64.447592
1	Unsporting Behavior	177	193	370	26.203966
2	Non-Violent Offense	56	76	132	9.348442
3	Grand Total	679	733	1412	100.000000

Figure 1a and 1b

In [ ]:
```
figure1a_graph = figure1a_data.dropna()
figure1a_graph = figure1a_graph.drop(figure1a_graph.index[-1])

plt.figure(figsize=(10, 6))

# Plotting the histogram
plt.bar(figure1a_graph['cause'], figure1a_graph['Percent'], color='blue', edgecolor='b

plt.title('Yellow Cards Given in Italys Serie A')
plt.xlabel('Motive')
plt.ylabel('Percentage')

plt.show()



figure1b_graph = figure1b_data.drop(figure1b_data.index[-1])

plt.figure(figsize=(10, 6))

# Plotting the histogram
plt.bar(figure1b_graph['Motive'], figure1b_graph['Percent'], color='blue', edgecolor='

plt.title('Yellow Cards Given in UEFA Champions League')
plt.xlabel('Motive')
plt.ylabel('Percentage')

plt.show()
```



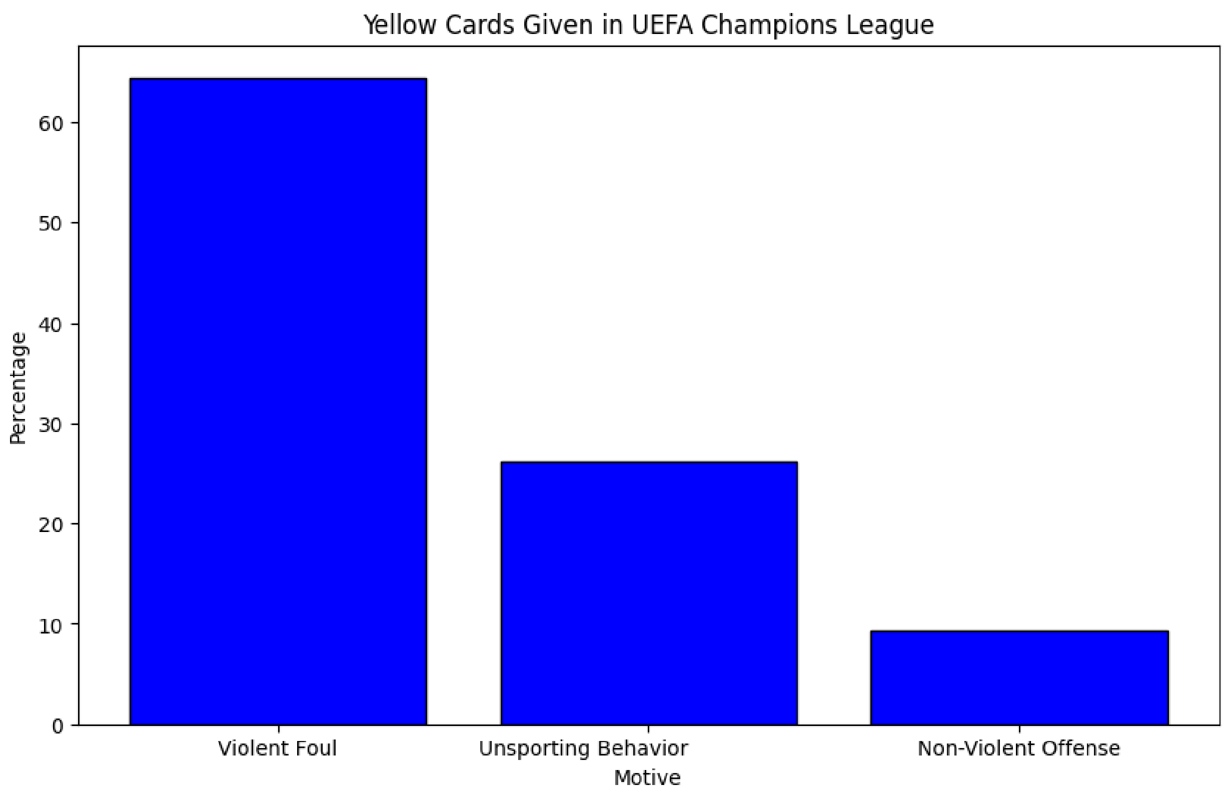


Table 1

In [ ]:
```
x_region = ["africa", "asia", "lac", "east_europe"]

# Renaming variables with labels (Stata labels are just for display, here we actually
data_for_table_1 = soccer_data.rename(columns={
    "civwar": "Years of civil war",
```

```
    "ln_income": "Log GNI per capita",
    "r_law": "Rule of Law",
    "age": "Age",
    "ln_contract": "Log transfer fee",
    "games_start": "Games Started",
    "games_sub": "Substitute",
    "defender": "Defender",
    "forward": "Forward",
    "midfield": "Midfield",
    "goalie": "Goalie",
    "goals": "Goals",
    "champions": "European Champions League",
    "french": "French League",
    "german": "German League",
    "italian": "Italian League",
    "spanish": "Spanish League",
    "english": "English League",
    "war_after": "Civil war years post-birth",
    "war_before": "Civil war years pre-birth"
})

# Data cleaning: Dropping missing values in 'Log transfer fee'
data_for_table_1 = data_for_table_1.dropna(subset=["Log transfer fee"])

data_for_table_1['GNI per capita (2006)'] = np.exp(data_for_table_1['Log GNI per capit
data_for_table_1['Weekly salary (in ’000 USD)'] = data_for_table_1['weekly_wage'] / 10
data_for_table_1['Transfer fee (in ’000 USD)'] = np.exp(data_for_table_1['Log transfer

# Define the conditions for filtering
condition = (
    data_for_table_1["Italian League"].notna() |
    data_for_table_1["European Champions League"].notna() |
    data_for_table_1["English League"].notna() |
    data_for_table_1["French League"].notna() |
    data_for_table_1["German League"].notna() |
    data_for_table_1["Spanish League"].notna()
) & (data_for_table_1['num_country'] >= 5)

# Columns to summarize
columns_to_summarize = [
    "yellow_card", "red_card", "Years of civil war", "Rule of Law", 'GNI per capita (2
    "Age", "Weekly salary (in ’000 USD)", "Transfer fee (in ’000 USD)", "Games Startec
    "Goalie", "Defender", "Forward", "Midfield", "Goals"
] + x_region + ["oecd", "English League", "European Champions League",
                "French League", "German League", "Italian League", "Spanish League"]



# Summary statistics for filtered data
summary = data_for_table_1.loc[condition, columns_to_summarize].describe()
Table1 = summary.T[['mean', 'std', 'min', 'max', 'count']].rename(columns={'mean': 'Me
Table1['Mean'] = np.round(Table1['Mean'], 2)
Table1 = Table1.rename(index={'yellow_card': 'Yellow cards per player-season',
                              'red_card': 'Red cards per player-season',
                              'Years of civil war': 'Years of civil war (1980–2005)',
                              'Rule of Law': 'Rule of law (2005–2006)',
                              'Goals' : 'Goals scored per player-season',
                              'africa': 'Africa',
                              'asia': 'Asia',
                              'lac': 'Latin America/Caribbean',
```

```
                                'east_europe': 'Eastern Europe',
                                'oecd': 'OECD'
                                })
Table1['Observations'] = Table1['Observations'].round(0).astype(int)
#Table1.loc[:, 'GNI per capita (2006)'] = Table1.loc[:, 'GNI per capita (2006)'].round

Table1
```

Out[ ]:

	Mean	Standard deviation	Minimum	Maximum	Observations
Yellow cards per player-season	2.43	2.734036	0.000000	16.000000	5035
Red cards per player-season	0.16	0.416225	0.000000	3.000000	5035
Years of civil war (1980–2005)	2.74	4.742952	0.000000	26.000000	5035
Rule of law (2005–2006)	0.85	0.886973	-1.760000	2.100000	5035
GNI per capita (2006)	26203.86	10923.413086	720.000061	44260.011719	4965
Age	25.99	4.404231	17.000000	41.000000	5035
Weekly salary (in ’000 USD)	23.99	27.014382	0.000000	190.000000	5034
Transfer fee (in ’000 USD)	6323.51	8198.477539	3.000000	78000.062500	5035
Games Started	13.80	11.485606	0.000000	40.000000	5035
Substitute	3.13	3.897660	0.000000	29.000000	5035
Goalie	0.08	0.267343	0.000000	1.000000	5035
Defender	0.33	0.471428	0.000000	1.000000	5035
Forward	0.23	0.419973	0.000000	1.000000	5035
Midfield	0.36	0.480244	0.000000	1.000000	5035
Goals scored per player-season	1.65	3.125677	0.000000	31.000000	5035
Africa	0.07	0.260633	0.000000	1.000000	5035
Asia	0.00	0.061320	0.000000	1.000000	5035
Latin America/Caribbean	0.12	0.330667	0.000000	1.000000	5035
Eastern Europe	0.07	0.253001	0.000000	1.000000	5035
OECD	0.74	0.441242	0.000000	1.000000	5035
English League	0.18	0.381337	0.000000	1.000000	5035
European Champions League	0.19	0.389544	0.000000	1.000000	5035
French League	0.15	0.360529	0.000000	1.000000	5035
German League	0.14	0.352121	0.000000	1.000000	5035
Italian League	0.17	0.378447	0.000000	1.000000	5035
Spanish League	0.17	0.371437	0.000000	1.000000	5035

Table 2

In [ ]:

```
# cant drop nans more columns than gni and civil war years pre and post because each c
# 4,965, and the ones with the pre and post have 5033
```

```
# numbers may not be exactly the same due to differences in algorithms used by stata a

# difficulties have been getting the syntax right and making sure the names line up an
# need to read fine print and make sure u include everything even if it isnt in the ta
# Also getting a lot of warnings about convergence and dividing by 0

# First regression

import statsmodels.api as sm
import statsmodels.formula.api as smf
from patsy import dmatrices

m = 10000

data_for_table_2 = soccer_data.rename(columns={
    "civwar": "Years of civil war",
    "ln_income": "Log GNI per capita",
    "r_law": "Rule of Law",
    "age": "Age",
    "ln_contract": "Log transfer fee",
    "games_start": "Games Started",
    "games_sub": "Substitute",
    "defender": "Defender",
    "forward": "Forward",
    "midfield": "Midfield",
    "goalie": "Goalie",
    "goals": "Goals",
    "champions": "European Champions League",
    "french": "French League",
    "german": "German League",
    "italian": "Italian League",
    "spanish": "Spanish League",
    "english": "English League",
    "war_after": "Civil war years post-birth",
    "war_before": "Civil war years pre-birth"
})
data_for_table_2 = data_for_table_2.dropna(subset=["Log transfer fee"])

condition = (
    data_for_table_2["Italian League"].notna() |
    data_for_table_2["European Champions League"].notna() |
    data_for_table_2["English League"].notna() |
    data_for_table_2["French League"].notna() |
    data_for_table_2["German League"].notna() |
    data_for_table_2["Spanish League"].notna()
) & (data_for_table_2['num_country'] >= 5)


data_for_table_2 = data_for_table_2.loc[condition, :]

formula1 = ('yellow_card ~ Q("Years of civil war") + Age + Q("Log transfer fee") + Q("
           'Q("Forward") + Q("Midfield") + Goals + Q("European Champions League") + Q(
            'Q("Spanish League") + africa + asia + lac + east_europe')


model_col1 = smf.negativebinomial(formula1, data=data_for_table_2).fit(cov_type='clust
print(model_col1.summary())


data_for_table_2_col_2 = data_for_table_2[~data_for_table_2['Log GNI per capita'].isna
```

```
formula2 = ('yellow_card ~ Q("Years of civil war") + Q("Log GNI per capita") + Age + Q
           'Q("Forward") + Q("Midfield") + Goals + Q("European Champions League") + Q(
            'Q("Spanish League") + africa + asia + lac + east_europe')


model_col2 = smf.negativebinomial(formula2, data=data_for_table_2_col_2).fit(cov_type=
print(model_col2.summary())


formula3 = ('yellow_card ~ Q("Years of civil war") + Q("Rule of Law") + Age + Q("Log t
           'Q("Forward") + Q("Midfield") + Goals +Q("European Champions League") + Q("
            'Q("Spanish League") + africa + asia + lac + east_europe')

model_col3 = smf.negativebinomial(formula3, data=data_for_table_2).fit(cov_type='clust
print(model_col3.summary())


formula4 = ('red_card ~ Q("Years of civil war") + Q("Rule of Law") + Age + Q("Log tran
           'Q("Forward") + Q("Midfield") + Goals +Q("European Champions League") + Q("
            'Q("Spanish League") + africa + asia + lac + east_europe')

model_col4 = smf.negativebinomial(formula4, data=data_for_table_2).fit(cov_type='clust
print(model_col4.summary())

#running into issue where introducing the goalie variable in causing divisions by zerc
X = data_for_table_2[["Years of civil war", "Rule of Law", 'Age', "Log transfer fee",
           "Forward", "Midfield", 'Goalie', "European Champions League", "French Leagu
            "Spanish League", 'africa', 'asia', 'lac', 'east_europe']]
X = sm.add_constant(X)
X_infl = sm.add_constant(data_for_table_2['Goalie'])

y = data_for_table_2['Goals']

model_col5 = sm.NegativeBinomialP(y, X)
model_col5 = model_col5.fit(cov_type='cluster', cov_kwds={'groups': data_for_table_2['


print(model_col5.summary())

data_for_table_2_col_6_and_7 = data_for_table_2[~(data_for_table_2["Civil war years pc

formula6 = ('yellow_card ~ Q("Civil war years post-birth") + Q("Civil war years pre-bi
           'Q("Forward") + Q("Midfield") + Goals + Q("European Champions League") + Q(
            'Q("Spanish League") + africa + asia + lac + east_europe')


model_col6 = smf.negativebinomial(formula6, data=data_for_table_2_col_6_and_7).fit(cov
print(model_col6.summary())


formula7 = ('red_card ~ Q("Civil war years post-birth") + Q("Civil war years pre-birth
           'Q("Forward") + Q("Midfield") + Goals + Q("European Champions League") + Q(
            'Q("Spanish League") + africa + asia + lac + east_europe')


model_col7 = smf.negativebinomial(formula7, data=data_for_table_2_col_6_and_7).fit(cov
print(model_col7.summary())
```

```
Optimization terminated successfully.
         Current function value: 1.677944
         Iterations: 68
         Function evaluations: 71
         Gradient evaluations: 71
                     NegativeBinomial Regression Results
==============================================================================
Dep. Variable:            yellow_card   No. Observations:                 5035
Model:               NegativeBinomial   Df Residuals:                     5016
Method:                           MLE   Df Model:                           18
Date:                Thu, 25 Apr 2024   Pseudo R-squ.:                  0.1900
Time:                        23:04:07   Log-Likelihood:                -8448.4
converged:                       True   LL-Null:                       -10430.
Covariance Type:              cluster   LLR p-value:                     0.000
=======================================================================================
=============
                                      coef    std err          z      P>|z|      [0.02
5      0.975]
---------------------------------------------------------------------------------------
-------------
Intercept                          -3.1473      0.316     -9.953      0.000      -3.76
7      -2.528
Q("Years of civil war")             0.0076      0.003      2.621      0.009       0.00
2       0.013
Age                                 0.0132      0.002      5.635      0.000       0.00
9       0.018
Q("Log transfer fee")               0.0325      0.014      2.329      0.020       0.00
5       0.060
Q("Games Started")                  0.0678      0.002     36.022      0.000       0.06
4       0.072
Q("Substitute")                     0.0405      0.004     10.910      0.000       0.03
3       0.048
Q("Defender")                       1.7153      0.117     14.704      0.000       1.48
7       1.944
Q("Forward")                        1.3973      0.127     11.034      0.000       1.14
9       1.645
Q("Midfield")                       1.7293      0.137     12.649      0.000       1.46
1       1.997
Goals                              -0.0221      0.004     -5.802      0.000      -0.03
0      -0.015
Q("European Champions League")    -0.0305      0.060     -0.507      0.612      -0.14
8       0.087
Q("French League")                  0.2646      0.060      4.418      0.000       0.14
7       0.382
Q("German League")                  0.3183      0.048      6.696      0.000       0.22
5       0.412
Q("Italian League")                 0.3532      0.054      6.497      0.000       0.24
7       0.460
Q("Spanish League")                 0.5437      0.050     10.859      0.000       0.44
6       0.642
africa                              0.0554      0.052      1.060      0.289      -0.04
7       0.158
asia                               -0.4109      0.201     -2.049      0.040      -0.80
4      -0.018
lac                                 0.0338      0.074      0.459      0.646      -0.11
1       0.178
east_europe                        -0.0273      0.064     -0.431      0.667      -0.15
2       0.097
alpha                               0.1590      0.012     13.325      0.000       0.13
6       0.182
```

```
=====================================================================================
=============
Optimization terminated successfully.
         Current function value: 1.676763
         Iterations: 71
         Function evaluations: 74
         Gradient evaluations: 74
                     NegativeBinomial Regression Results
==============================================================================
Dep. Variable:            yellow_card   No. Observations:                 4965
Model:               NegativeBinomial   Df Residuals:                     4945
Method:                           MLE   Df Model:                           19
Date:                Thu, 25 Apr 2024   Pseudo R-squ.:                  0.1900
Time:                        23:04:07   Log-Likelihood:                -8325.1
converged:                       True   LL-Null:                       -10278.
Covariance Type:              cluster   LLR p-value:                     0.000
=====================================================================================
=============
                                     coef    std err          z      P>|z|      [0.02
5      0.975]
-------------------------------------------------------------------------------------
-------------
Intercept                         -3.5972      0.550     -6.542      0.000      -4.67
5     -2.520
Q("Years of civil war")            0.0078      0.003      2.508      0.012       0.00
2      0.014
Q("Log GNI per capita")            0.0467      0.044      1.057      0.290      -0.04
0      0.133
Age                                0.0130      0.002      5.391      0.000       0.00
8      0.018
Q("Log transfer fee")              0.0307      0.014      2.219      0.026       0.00
4      0.058
Q("Games Started")                 0.0679      0.002     35.704      0.000       0.06
4      0.072
Q("Substitute")                    0.0405      0.004     10.813      0.000       0.03
3      0.048
Q("Defender")                      1.7137      0.116     14.757      0.000       1.48
6      1.941
Q("Forward")                       1.3990      0.126     11.110      0.000       1.15
2      1.646
Q("Midfield")                      1.7284      0.137     12.651      0.000       1.46
1      1.996
Goals                             -0.0229      0.004     -6.280      0.000      -0.03
0     -0.016
Q("European Champions League")   -0.0232      0.061     -0.381      0.703      -0.14
3      0.096
Q("French League")                 0.2657      0.062      4.257      0.000       0.14
3      0.388
Q("German League")                 0.3210      0.049      6.567      0.000       0.22
5      0.417
Q("Italian League")                0.3557      0.057      6.254      0.000       0.24
4      0.467
Q("Spanish League")                0.5480      0.051     10.724      0.000       0.44
8      0.648
africa                             0.1734      0.120      1.447      0.148      -0.06
2      0.408
asia                              -0.3807      0.190     -2.000      0.046      -0.75
4     -0.008
lac                                0.0827      0.075      1.108      0.268      -0.06
4      0.229
```

```
east_europe                  -0.0372      0.087     -0.428      0.669      -0.20
7       0.133
alpha                         0.1594      0.012     13.282      0.000       0.13
6       0.183
==============================================================================
=============
Optimization terminated successfully.
         Current function value: 1.677901
         Iterations: 70
         Function evaluations: 73
         Gradient evaluations: 73
                     NegativeBinomial Regression Results
==========================================================================
Dep. Variable:           yellow_card   No. Observations:                 5035
Model:              NegativeBinomial   Df Residuals:                     5015
Method:                          MLE   Df Model:                           19
Date:               Thu, 25 Apr 2024   Pseudo R-squ.:                  0.1900
Time:                       23:04:09   Log-Likelihood:                -8448.2
converged:                      True   LL-Null:                       -10430.
Covariance Type:             cluster   LLR p-value:                     0.000
==============================================================================
=============
                                   coef    std err          z      P>|z|      [0.02
5      0.975]
------------------------------------------------------------------------------
-------------
Intercept                    -3.1143      0.341     -9.134      0.000      -3.78
3      -2.446
Q("Years of civil war")       0.0076      0.003      2.582      0.010       0.00
2       0.013
Q("Rule of Law")             -0.0197      0.049     -0.399      0.690      -0.11
6       0.077
Age                           0.0132      0.002      5.629      0.000       0.00
9       0.018
Q("Log transfer fee")         0.0325      0.014      2.327      0.020       0.00
5       0.060
Q("Games Started")            0.0678      0.002     36.096      0.000       0.06
4       0.072
Q("Substitute")               0.0405      0.004     10.870      0.000       0.03
3       0.048
Q("Defender")                 1.7141      0.117     14.683      0.000       1.48
5       1.943
Q("Forward")                  1.3962      0.127     11.028      0.000       1.14
8       1.644
Q("Midfield")                 1.7285      0.137     12.634      0.000       1.46
0       1.997
Goals                        -0.0222      0.004     -5.816      0.000      -0.03
0      -0.015
Q("European Champions League")  -0.0365      0.058     -0.627      0.531      -0.15
1       0.078
Q("French League")            0.2597      0.062      4.199      0.000       0.13
8       0.381
Q("German League")            0.3174      0.048      6.566      0.000       0.22
3       0.412
Q("Italian League")           0.3370      0.066      5.098      0.000       0.20
7       0.467
Q("Spanish League")           0.5348      0.056      9.523      0.000       0.42
5       0.645
africa                        0.0187      0.104      0.180      0.857      -0.18
4       0.222
```

```
asia                              -0.4351      0.216     -2.012      0.044      -0.85
9      -0.011
lac                                0.0040      0.129      0.031      0.976      -0.24
9       0.257
east_europe                       -0.0569      0.101     -0.565      0.572      -0.25
4       0.140
alpha                              0.1590      0.012     13.330      0.000       0.13
6       0.182
=====================================================================================
=============
Optimization terminated successfully.
         Current function value: 0.410759
         Iterations: 106
         Function evaluations: 108
         Gradient evaluations: 108
                     NegativeBinomial Regression Results
==============================================================================
Dep. Variable:               red_card   No. Observations:                 5035
Model:               NegativeBinomial   Df Residuals:                     5015
Method:                           MLE   Df Model:                           19
Date:                Thu, 25 Apr 2024   Pseudo R-squ.:                  0.1074
Time:                        23:04:11   Log-Likelihood:                -2068.2
converged:                       True   LL-Null:                       -2316.9
Covariance Type:              cluster   LLR p-value:                 1.925e-93
=====================================================================================
=============
                                     coef    std err          z      P>|z|      [0.02
5      0.975]
-------------------------------------------------------------------------------------
-------------
Intercept                         -5.1123      0.660     -7.741      0.000      -6.40
7      -3.818
Q("Years of civil war")            0.0126      0.007      1.926      0.054      -0.00
0       0.026
Q("Rule of Law")                  -0.1431      0.098     -1.455      0.146      -0.33
6       0.050
Age                                0.0129      0.007      1.751      0.080      -0.00
2       0.027
Q("Log transfer fee")              0.0633      0.030      2.131      0.033       0.00
5       0.121
Q("Games Started")                 0.0506      0.003     18.693      0.000       0.04
5       0.056
Q("Substitute")                    0.0110      0.012      0.884      0.377      -0.01
3       0.035
Q("Defender")                      1.1134      0.156      7.145      0.000       0.80
8       1.419
Q("Forward")                       0.7200      0.181      3.986      0.000       0.36
6       1.074
Q("Midfield")                      0.8891      0.201      4.432      0.000       0.49
6       1.282
Goals                             -0.0284      0.008     -3.396      0.001      -0.04
5      -0.012
Q("European Champions League")    -0.5042      0.208     -2.420      0.016      -0.91
3      -0.096
Q("French League")                 0.2969      0.114      2.611      0.009       0.07
4       0.520
Q("German League")                 0.0971      0.155      0.626      0.531      -0.20
7       0.401
Q("Italian League")                0.6288      0.138      4.549      0.000       0.35
8       0.900
```

```
Q("Spanish League")               0.6481      0.106      6.133      0.000       0.44
1       0.855
africa                           -0.1127      0.231     -0.487      0.626      -0.56
6       0.341
asia                             -0.6467      0.377     -1.716      0.086      -1.38
5       0.092
lac                              -0.0602      0.185     -0.325      0.745      -0.42
3       0.303
east_europe                      -0.1093      0.224     -0.489      0.625      -0.54
7       0.329
alpha                             0.0005      0.064      0.008      0.994      -0.12
4       0.125
=====================================================================================
=============
Optimization terminated successfully.
         Current function value: 1.254629
         Iterations: 79
         Function evaluations: 84
         Gradient evaluations: 84
```

```
/usr/local/lib/python3.10/dist-packages/statsmodels/base/model.py:595: HessianInversi
onWarning: Inverting hessian failed, no bse or cov_params available
  warnings.warn('Inverting hessian failed, no bse or cov_params '
```

```
                     NegativeBinomialP Regression Results
==============================================================================
Dep. Variable:                  Goals   No. Observations:                 5035
Model:              NegativeBinomialP   Df Residuals:                     5016
Method:                           MLE   Df Model:                           18
Date:                Thu, 25 Apr 2024   Pseudo R-squ.:                  0.2396
Time:                        23:04:13   Log-Likelihood:                -6317.1
converged:                       True   LL-Null:                       -8307.5
Covariance Type:              cluster   LLR p-value:                     0.000
=======================================================================================
========
                               coef    std err          z      P>|z|      [0.025
0.975]
---------------------------------------------------------------------------------------
--------
const                       -7.8566      0.434    -18.101      0.000      -8.707
-7.006
Years of civil war        8.995e-05      0.005      0.019      0.985      -0.009
0.009
Rule of Law                  0.0061      0.042      0.145      0.885      -0.076
0.088
Age                          0.0205      0.006      3.191      0.001       0.008
0.033
Log transfer fee             0.3222      0.027     11.853      0.000       0.269
0.375
Games Started                0.0869      0.002     40.082      0.000       0.083
0.091
Substitute                   0.0689      0.005     13.622      0.000       0.059
0.079
Forward                      1.6466      0.078     21.220      0.000       1.495
1.799
Midfield                     0.6796      0.060     11.286      0.000       0.562
0.798
Goalie                     -52.4183      0.336   -155.986      0.000     -53.077
-51.760
European Champions League    0.2112      0.086      2.449      0.014       0.042
0.380
French League                0.0782      0.063      1.240      0.215      -0.045
0.202
German League                0.2440      0.059      4.113      0.000       0.128
0.360
Italian League              -0.0127      0.057     -0.221      0.825      -0.125
0.100
Spanish League               0.0020      0.063      0.032      0.974      -0.121
0.125
africa                       0.0997      0.114      0.875      0.382      -0.124
0.323
asia                        -0.1246      0.244     -0.510      0.610      -0.603
0.354
lac                          0.1720      0.103      1.677      0.093      -0.029
0.373
east_europe                  0.1161      0.074      1.573      0.116      -0.029
0.261
alpha                        0.3601      0.029     12.579      0.000       0.304
0.416
=======================================================================================
========
Optimization terminated successfully.
         Current function value: 1.677994
         Iterations: 68
```

```
         Function evaluations: 73
         Gradient evaluations: 73
                     NegativeBinomial Regression Results                      
==============================================================================
Dep. Variable:            yellow_card   No. Observations:                 5033
Model:               NegativeBinomial   Df Residuals:                     5013
Method:                           MLE   Df Model:                           19
Date:                Thu, 25 Apr 2024   Pseudo R-squ.:                  0.1900
Time:                        23:04:15   Log-Likelihood:                -8445.3
converged:                       True   LL-Null:                       -10426.
Covariance Type:              cluster   LLR p-value:                     0.000
=====================================================================================
==============
                                      coef    std err          z      P>|z|      [0.0
25      0.975]
-------------------------------------------------------------------------------------
--------------
Intercept                          -3.1418      0.319     -9.841      0.000      -3.7
68      -2.516
Q("Civil war years post-birth")     0.0052      0.003      1.860      0.063      -0.0
00       0.011
Q("Civil war years pre-birth")      0.0036      0.005      0.732      0.464      -0.0
06       0.013
Age                                 0.0131      0.002      5.756      0.000       0.0
09       0.017
Q("Log transfer fee")               0.0324      0.014      2.336      0.019       0.0
05       0.060
Q("Games Started")                  0.0678      0.002     36.005      0.000       0.0
64       0.072
Q("Substitute")                     0.0406      0.004     10.969      0.000       0.0
33       0.048
Q("Defender")                       1.7149      0.117     14.666      0.000       1.4
86       1.944
Q("Forward")                        1.3970      0.127     10.991      0.000       1.1
48       1.646
Q("Midfield")                       1.7286      0.137     12.633      0.000       1.4
60       1.997
Goals                              -0.0221      0.004     -5.828      0.000      -0.0
29      -0.015
Q("European Champions League")     -0.0283      0.062     -0.458      0.647      -0.1
49       0.093
Q("French League")                  0.2634      0.064      4.138      0.000       0.1
39       0.388
Q("German League")                  0.3199      0.051      6.318      0.000       0.2
21       0.419
Q("Italian League")                 0.3533      0.056      6.272      0.000       0.2
43       0.464
Q("Spanish League")                 0.5516      0.055     10.065      0.000       0.4
44       0.659
africa                              0.0590      0.056      1.048      0.295      -0.0
51       0.169
asia                               -0.4204      0.178     -2.362      0.018      -0.7
69      -0.072
lac                                 0.0233      0.074      0.313      0.754      -0.1
23       0.169
east_europe                        -0.0254      0.064     -0.394      0.694      -0.1
51       0.101
alpha                               0.1590      0.012     13.273      0.000       0.1
36       0.183
=====================================================================================
```

```
==============
Optimization terminated successfully.
         Current function value: 0.411010
         Iterations: 97
         Function evaluations: 99
         Gradient evaluations: 99
                     NegativeBinomial Regression Results
==============================================================================
Dep. Variable:               red_card   No. Observations:                 5033
Model:               NegativeBinomial   Df Residuals:                     5013
Method:                           MLE   Df Model:                           19
Date:                Thu, 25 Apr 2024   Pseudo R-squ.:                  0.1070
Time:                        23:04:16   Log-Likelihood:                -2068.6
converged:                       True   LL-Null:                       -2316.6
Covariance Type:              cluster   LLR p-value:                 3.924e-93
=====================================================================================
==============
                                     coef    std err          z      P>|z|      [0.0
25      0.975]
-------------------------------------------------------------------------------------
--------------
Intercept                         -5.2844      0.612     -8.639      0.000      -6.4
83      -4.086
Q("Civil war years post-birth")    0.0142      0.007      2.142      0.032       0.0
01       0.027
Q("Civil war years pre-birth")    -0.0040      0.010     -0.407      0.684      -0.0
23       0.015
Age                                0.0105      0.007      1.552      0.121      -0.0
03       0.024
Q("Log transfer fee")              0.0633      0.030      2.119      0.034       0.0
05       0.122
Q("Games Started")                 0.0506      0.003     18.751      0.000       0.0
45       0.056
Q("Substitute")                    0.0110      0.012      0.890      0.374      -0.0
13       0.035
Q("Defender")                      1.1198      0.155      7.226      0.000       0.8
16       1.424
Q("Forward")                       0.7258      0.180      4.042      0.000       0.3
74       1.078
Q("Midfield")                      0.8917      0.199      4.487      0.000       0.5
02       1.281
Goals                             -0.0279      0.008     -3.296      0.001      -0.0
45      -0.011
Q("European Champions League")    -0.4551      0.205     -2.219      0.026      -0.8
57      -0.053
Q("French League")                 0.3339      0.114      2.925      0.003       0.1
10       0.558
Q("German League")                 0.1114      0.165      0.675      0.500      -0.2
12       0.435
Q("Italian League")                0.7486      0.106      7.063      0.000       0.5
41       0.956
Q("Spanish League")                0.7187      0.108      6.682      0.000       0.5
08       0.930
africa                             0.1490      0.107      1.395      0.163      -0.0
60       0.358
asia                              -0.4523      0.351     -1.290      0.197      -1.1
40       0.235
lac                                0.1490      0.068      2.184      0.029       0.0
15       0.283
east_europe                        0.0954      0.182      0.524      0.600      -0.2
```

```
62       0.452
alpha                                  0.0014      0.064      0.022      0.982      -0.1
24       0.127
=====================================================================================
==============
```

In [ ]:

```
# Assuming `model1`, `model2`, ..., `model7` are your fitted Negative Binomial models
models = [model_col1, model_col2, model_col3, model_col4, model_col5, model_col6, mode

# Create a Stargazer object
stargazer = Stargazer(models)

# Customize the stargazer object as needed, e.g.:
stargazer.title('Empirical Results: Civil War')
stargazer.custom_columns(['Model 1', 'Model 2', 'Model 3', 'Model 4', 'Model 5', 'Mode
stargazer.show_model_numbers(False)
stargazer.significance_levels([0.05, 0.01, 0.001])
print(stargazer.render_latex())
```

```
\begin{table}[!htbp] \centering
  \caption{Empirical Results: Civil War}
\begin{tabular}{@{\extracolsep{5pt}}lccccccc}
\\[-1.8ex]\hline
\hline \\[-1.8ex]
\\[-1.8ex] & \multicolumn{1}{c}{Model 1} & \multicolumn{1}{c}{Model 2} & \multicolumn
{1}{c}{Model 3} & \multicolumn{1}{c}{Model 4} & \multicolumn{1}{c}{Model 5} & \multic
olumn{1}{c}{Model 6} & \multicolumn{1}{c}{Model 7}  \\
\hline \\[-1.8ex]
 Age & 0.013$^{***}$ & 0.013$^{***}$ & 0.013$^{***}$ & 0.013$^{}$ & 0.021$^{**}$ & 0.
013$^{***}$ & 0.011$^{}$ \\
& (0.002) & (0.002) & (0.002) & (0.007) & (0.006) & (0.002) & (0.007) \\
 European Champions League & & & & & 0.211$^{*}$ & & \\
& & & & & (0.086) & & \\
 Forward & & & & & 1.647$^{***}$ & & \\
& & & & & (0.078) & & \\
 French League & & & & & 0.078$^{}$ & & \\
& & & & & (0.063) & & \\
 Games Started & & & & & 0.087$^{***}$ & & \\
& & & & & (0.002) & & \\
 German League & & & & & 0.244$^{***}$ & & \\
& & & & & (0.059) & & \\
 Goalie & & & & & -52.418$^{***}$ & & \\
& & & & & (0.336) & & \\
 Goals & -0.022$^{***}$ & -0.023$^{***}$ & -0.022$^{***}$ & -0.028$^{***}$ & & -0.022
$^{***}$ & -0.028$^{***}$ \\
& (0.004) & (0.004) & (0.004) & (0.008) & & (0.004) & (0.008) \\
 Intercept & -3.147$^{***}$ & -3.597$^{***}$ & -3.114$^{***}$ & -5.112$^{***}$ & & -
3.142$^{***}$ & -5.284$^{***}$ \\
& (0.316) & (0.550) & (0.341) & (0.660) & & (0.319) & (0.612) \\
 Italian League & & & & & -0.013$^{}$ & & \\
& & & & & (0.057) & & \\
 Log transfer fee & & & & & 0.322$^{***}$ & & \\
& & & & & (0.027) & & \\
 Midfield & & & & & 0.680$^{***}$ & & \\
& & & & & (0.060) & & \\
 Q("Civil war years post-birth") & & & & & & 0.005$^{}$ & 0.014$^{*}$ \\
& & & & & & (0.003) & (0.007) \\
 Q("Civil war years pre-birth") & & & & & & 0.004$^{}$ & -0.004$^{}$ \\
& & & & & & (0.005) & (0.010) \\
 Q("Defender") & 1.715$^{***}$ & 1.714$^{***}$ & 1.714$^{***}$ & 1.113$^{***}$ & & 1.
715$^{***}$ & 1.120$^{***}$ \\
& (0.117) & (0.116) & (0.117) & (0.156) & & (0.117) & (0.155) \\
 Q("European Champions League") & -0.030$^{}$ & -0.023$^{}$ & -0.036$^{}$ & -0.504$^
{*}$ & & -0.028$^{}$ & -0.455$^{*}$ \\
& (0.060) & (0.061) & (0.058) & (0.208) & & (0.062) & (0.205) \\
 Q("Forward") & 1.397$^{***}$ & 1.399$^{***}$ & 1.396$^{***}$ & 0.720$^{***}$ & & 1.3
97$^{***}$ & 0.726$^{***}$ \\
& (0.127) & (0.126) & (0.127) & (0.181) & & (0.127) & (0.180) \\
 Q("French League") & 0.265$^{***}$ & 0.266$^{***}$ & 0.260$^{***}$ & 0.297$^{**}$ &
& 0.263$^{***}$ & 0.334$^{**}$ \\
& (0.060) & (0.062) & (0.062) & (0.114) & & (0.064) & (0.114) \\
 Q("Games Started") & 0.068$^{***}$ & 0.068$^{***}$ & 0.068$^{***}$ & 0.051$^{***}$ &
& 0.068$^{***}$ & 0.051$^{***}$ \\
& (0.002) & (0.002) & (0.002) & (0.003) & & (0.002) & (0.003) \\
 Q("German League") & 0.318$^{***}$ & 0.321$^{***}$ & 0.317$^{***}$ & 0.097$^{}$ & &
0.320$^{***}$ & 0.111$^{}$ \\
& (0.048) & (0.049) & (0.048) & (0.155) & & (0.051) & (0.165) \\
 Q("Italian League") & 0.353$^{***}$ & 0.356$^{***}$ & 0.337$^{***}$ & 0.629$^{***}$
& & 0.353$^{***}$ & 0.749$^{***}$ \\
```

```
& (0.054) & (0.057) & (0.066) & (0.138) & & (0.056) & (0.106) \\
 Q("Log GNI per capita") & & 0.047$^{}$ & & & & & \\
& & (0.044) & & & & & \\
 Q("Log transfer fee") & 0.032$^{*}$ & 0.031$^{*}$ & 0.033$^{*}$ & 0.063$^{*}$ & & 0.
032$^{*}$ & 0.063$^{*}$ \\
& (0.014) & (0.014) & (0.014) & (0.030) & & (0.014) & (0.030) \\
 Q("Midfield") & 1.729$^{***}$ & 1.728$^{***}$ & 1.729$^{***}$ & 0.889$^{***}$ & & 1.
729$^{***}$ & 0.892$^{***}$ \\
& (0.137) & (0.137) & (0.137) & (0.201) & & (0.137) & (0.199) \\
 Q("Rule of Law") & & & -0.020$^{}$ & -0.143$^{}$ & & & \\
& & & (0.049) & (0.098) & & & \\
 Q("Spanish League") & 0.544$^{***}$ & 0.548$^{***}$ & 0.535$^{***}$ & 0.648$^{***}$
& & 0.552$^{***}$ & 0.719$^{***}$ \\
& (0.050) & (0.051) & (0.056) & (0.106) & & (0.055) & (0.108) \\
 Q("Substitute") & 0.040$^{***}$ & 0.040$^{***}$ & 0.040$^{***}$ & 0.011$^{}$ & & 0.0
41$^{***}$ & 0.011$^{}$ \\
& (0.004) & (0.004) & (0.004) & (0.012) & & (0.004) & (0.012) \\
 Q("Years of civil war") & 0.008$^{**}$ & 0.008$^{*}$ & 0.008$^{**}$ & 0.013$^{}$ & &
& \\
& (0.003) & (0.003) & (0.003) & (0.007) & & & \\
 Rule of Law & & & & & 0.006$^{}$ & & \\
& & & & & (0.042) & & \\
 Spanish League & & & & & 0.002$^{}$ & & \\
& & & & & (0.063) & & \\
 Substitute & & & & & 0.069$^{***}$ & & \\
& & & & & (0.005) & & \\
 Years of civil war & & & & & 0.000$^{}$ & & \\
& & & & & (0.005) & & \\
 africa & 0.055$^{}$ & 0.173$^{}$ & 0.019$^{}$ & -0.113$^{}$ & 0.100$^{}$ & 0.059$^{}
$ & 0.149$^{}$ \\
& (0.052) & (0.120) & (0.104) & (0.231) & (0.114) & (0.056) & (0.107) \\
 alpha & 0.159$^{***}$ & 0.159$^{***}$ & 0.159$^{***}$ & 0.000$^{}$ & 0.360$^{***}$ &
0.159$^{***}$ & 0.001$^{}$ \\
& (0.012) & (0.012) & (0.012) & (0.064) & (0.029) & (0.012) & (0.064) \\
 asia & -0.411$^{*}$ & -0.381$^{*}$ & -0.435$^{*}$ & -0.647$^{}$ & -0.125$^{}$ & -0.4
20$^{*}$ & -0.452$^{}$ \\
& (0.201) & (0.190) & (0.216) & (0.377) & (0.244) & (0.178) & (0.351) \\
 const & & & & & -7.857$^{***}$ & & \\
& & & & & (0.434) & & \\
 east_europe & -0.027$^{}$ & -0.037$^{}$ & -0.057$^{}$ & -0.109$^{}$ & 0.116$^{}$ & -
0.025$^{}$ & 0.095$^{}$ \\
& (0.064) & (0.087) & (0.101) & (0.224) & (0.074) & (0.064) & (0.182) \\
 lac & 0.034$^{}$ & 0.083$^{}$ & 0.004$^{}$ & -0.060$^{}$ & 0.172$^{}$ & 0.023$^{}$ &
0.149$^{*}$ \\
& (0.074) & (0.075) & (0.129) & (0.185) & (0.103) & (0.074) & (0.068) \\
\hline \\[-1.8ex]
 Observations & 5035 & 4965 & 5035 & 5035 & 5035 & 5033 & 5033 \\
 Pseudo $R^2$ & 0.190 & 0.190 & 0.190 & 0.107 & 0.240 & 0.190 & 0.107 \\
 Residual Std. Error & 1.992 (df=5016) & 1.990 (df=4945) & 1.992 (df=5015) & 0.397 (d
f=5015) & 1.887 (df=5016) & 1.991 (df=5013) & 0.398 (df=5013) \\
\hline
\hline \\[-1.8ex]
\textit{Note:} & \multicolumn{7}{r}{$^{*}$p$<$0.05; $^{**}$p$<$0.01; $^{***}$p$<$0.00
1} \\
\end{tabular}
\end{table}
```

Figure 2

Yellow cards and civil war (conditional on control variables in Table 2, regression 1) – all countries.

Biggest struggle here was realizing that we do not want to predict the total yellow cards, we need to divide by number of player seasons. In addition, for a long time I thought we had to use our original regression but it turns out we just do a new one for these plots. However, the plots are still not exactly the same and I am not sure what is causing these differences. I had to change the original stata code a bit because it was trying to predict total cards instead of average, and I am not sure exactly how they did this originally. Also, the CIV circle, which is Cote de Ivore, is showing up much higher than in the paper for all three of these similar figures. In soccer_data, there are two names for the same country, Cote de Ivore and Ivory Coast, but I have confirmed that these both share the same country code and both get mapped to CIV in the merge. I am not sure of the origin of this issue.

In [ ]:
```
wb_codes = pd.read_csv("/content/drive/My Drive/Econ 148 Project 3/Soccer_Replication/
soccer_data = pd.read_stata("/content/drive/My Drive/Econ 148 Project 3/Soccer_Replica
```

In [ ]:
```
wb_codes.rename(columns={'id': 'nation'}, inplace=True)
wb_codes.sort_values('nation', inplace=True)
soccer_data.drop(columns='num_country', inplace=True, errors='ignore')
soccer_data.dropna(subset=['contract'], inplace=True)

conditions = (
    (soccer_data['italian'] == 1) | (soccer_data['champions'] == 1) |
    (soccer_data['english'] == 1) | (soccer_data['french'] == 1) |
    (soccer_data['german'] == 1) | (soccer_data['spanish'] == 1)
)
soccer_data = soccer_data[conditions]
soccer_data['num_country'] = soccer_data.groupby('nationality')['player_id'].transform

merged_data = pd.merge(soccer_data, wb_codes, on='nation', how='left')

grouped = merged_data[merged_data['num_country'] >= 5].groupby('wb_code').agg({
    'yellow_card': 'sum',
    'civwar': 'sum',
    'nation': 'first',
    'num_country': 'first',
    'age': 'mean',
    'games_start': 'mean',
    'games_sub': 'mean',
    'defender': 'mean',
    'forward': 'mean',
    'midfield': 'mean',
    'goals': 'mean',
    'ln_contract': 'mean',
    'italian': 'mean',
    'champions': 'mean',
    'english': 'mean',
    'french': 'mean',
    'german': 'mean',
    'spanish': 'mean',
    'africa': 'mean',
    'asia': 'mean',
    'lac': 'mean',
```

```
    'east_europe': 'mean',
    'oecd': 'mean'
})

grouped['yellow_avg'] = grouped['yellow_card'] / grouped['num_country']
grouped['war_avg'] = grouped['civwar'] / grouped['num_country']

formula_yellow = 'yellow_avg ~ age + games_start + games_sub + defender + forward + mi
model_yellow = smf.ols(formula=formula_yellow, data=grouped).fit()
grouped['yellowhat'] = model_yellow.predict(grouped)

formula_civwar = 'war_avg ~ age + games_start + games_sub + defender + forward + midfi
model_civwar = smf.ols(formula=formula_civwar, data=grouped).fit()
grouped['warhat'] = model_civwar.predict(grouped)

grouped['yellow_res'] = grouped['yellow_avg'] - grouped['yellowhat']
grouped['war_res'] = grouped['war_avg'] - grouped['warhat']

plt.figure(figsize=(10, 6))
for idx, row in grouped.iterrows():
    plt.scatter(row['war_res'], row['yellow_res'], s=row['num_country'] * 10, alpha=0.
    plt.text(row['war_res'], row['yellow_res'], idx, fontsize=9)  # Add text labels


y = grouped['yellow_res']
x = grouped['war_res']
slope, intercept = np.polyfit(x, y, 1)
plt.plot(x, slope*x + intercept, 'k-', linewidth=1)

plt.ylabel('Average Yellow Cards Per Player-Season (Residuals)')
plt.xlabel('Years of Civil War since 1980 (Residuals)')
plt.title('CIVIL WAR EXPOSURE AND VIOLENCE')
plt.grid(True)
plt.xticks(np.arange(-10, 18, 4))
plt.grid(True)
plt.show()
```

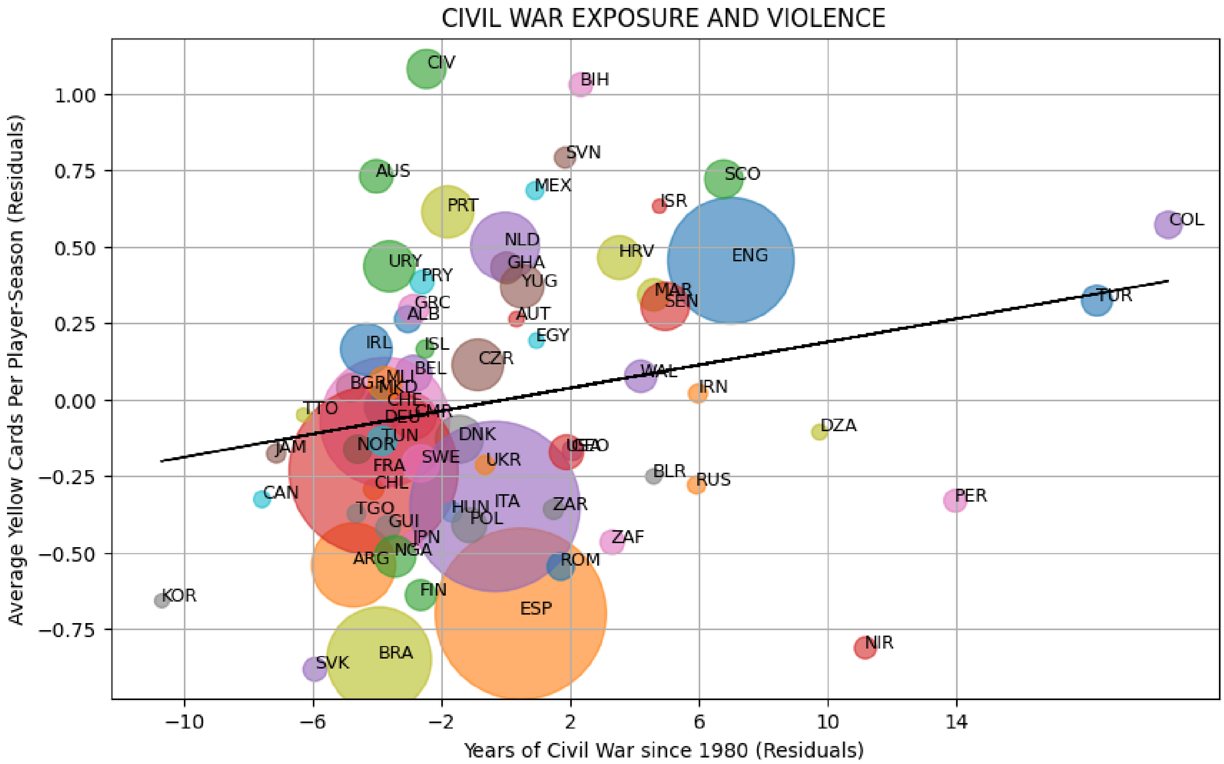


Figure 3

Yellow cards and civil war (conditional on control variables in Table 2, regression 1) – non-OECD countries.

Added a simple one line of code to filter out OECD countries. Still not sure what the CIV circle is. Maybe just drop it

In [ ]:
```
wb_codes = pd.read_csv("/content/drive/My Drive/Econ 148 Project 3/Soccer_Replication/
soccer_data = pd.read_stata("/content/drive/My Drive/Econ 148 Project 3/Soccer_Replica
```

In [ ]:
```
wb_codes.rename(columns={'id': 'nation'}, inplace=True)
wb_codes.sort_values('nation', inplace=True)
soccer_data.drop(columns='num_country', inplace=True, errors='ignore')
soccer_data.dropna(subset=['contract'], inplace=True)

conditions = (
    (soccer_data['italian'] == 1) | (soccer_data['champions'] == 1) |
    (soccer_data['english'] == 1) | (soccer_data['french'] == 1) |
    (soccer_data['german'] == 1) | (soccer_data['spanish'] == 1)
)
soccer_data = soccer_data[conditions]
soccer_data['num_country'] = soccer_data.groupby('nationality')['player_id'].transform

soccer_data = soccer_data[soccer_data['oecd'] == 0]
merged_data = pd.merge(soccer_data, wb_codes, on='nation', how='left')

grouped = merged_data[merged_data['num_country'] >= 5].groupby('wb_code').agg({
    'yellow_card': 'sum',
    'civwar': 'sum',
    'nation': 'first',
```

```
    'num_country': 'first',
    'age': 'mean',
    'games_start': 'mean',
    'games_sub': 'mean',
    'defender': 'mean',
    'forward': 'mean',
    'midfield': 'mean',
    'goals': 'mean',
    'ln_contract': 'mean',
    'italian': 'mean',
    'champions': 'mean',
    'english': 'mean',
    'french': 'mean',
    'german': 'mean',
    'spanish': 'mean',
    'africa': 'mean',
    'asia': 'mean',
    'lac': 'mean',
    'east_europe': 'mean',
    'oecd': 'mean'
})

grouped['yellow_avg'] = grouped['yellow_card'] / grouped['num_country']
grouped['war_avg'] = grouped['civwar'] / grouped['num_country']

formula_yellow = 'yellow_avg ~ age + games_start + games_sub + defender + forward + mi
model_yellow = smf.ols(formula=formula_yellow, data=grouped).fit()
grouped['yellowhat'] = model_yellow.predict(grouped)

formula_civwar = 'war_avg ~ age + games_start + games_sub + defender + forward + midfi
model_civwar = smf.ols(formula=formula_civwar, data=grouped).fit()
grouped['warhat'] = model_civwar.predict(grouped)

grouped['yellow_res'] = grouped['yellow_avg'] - grouped['yellowhat']
grouped['war_res'] = grouped['war_avg'] - grouped['warhat']

plt.figure(figsize=(10, 6))
for idx, row in grouped.iterrows():
    plt.scatter(row['war_res'], row['yellow_res'], s=row['num_country'] * 10, alpha=0.
    plt.text(row['war_res'], row['yellow_res'], idx, fontsize=9)  # Add text labels


y = grouped['yellow_res']
x = grouped['war_res']
slope, intercept = np.polyfit(x, y, 1)
plt.plot(x, slope*x + intercept, 'k-', linewidth=1)

plt.ylabel('Average Yellow Cards Per Player-Season (Residuals)')
plt.xlabel('Years of Civil War since 1980 (Residuals)')
plt.title('CIVIL WAR EXPOSURE AND VIOLENCE')
plt.grid(True)
plt.xticks(np.arange(-10, 18, 4))
plt.yticks(np.arange(-1, 1.1, .25))
plt.grid(True)
plt.show()
```

```
Exception ignored in: <function ZipFile.__del__ at 0x7c77b387a320>
Traceback (most recent call last):
  File "/usr/lib/python3.10/zipfile.py", line 1819, in __del__
    def __del__(self):
KeyboardInterrupt:
```

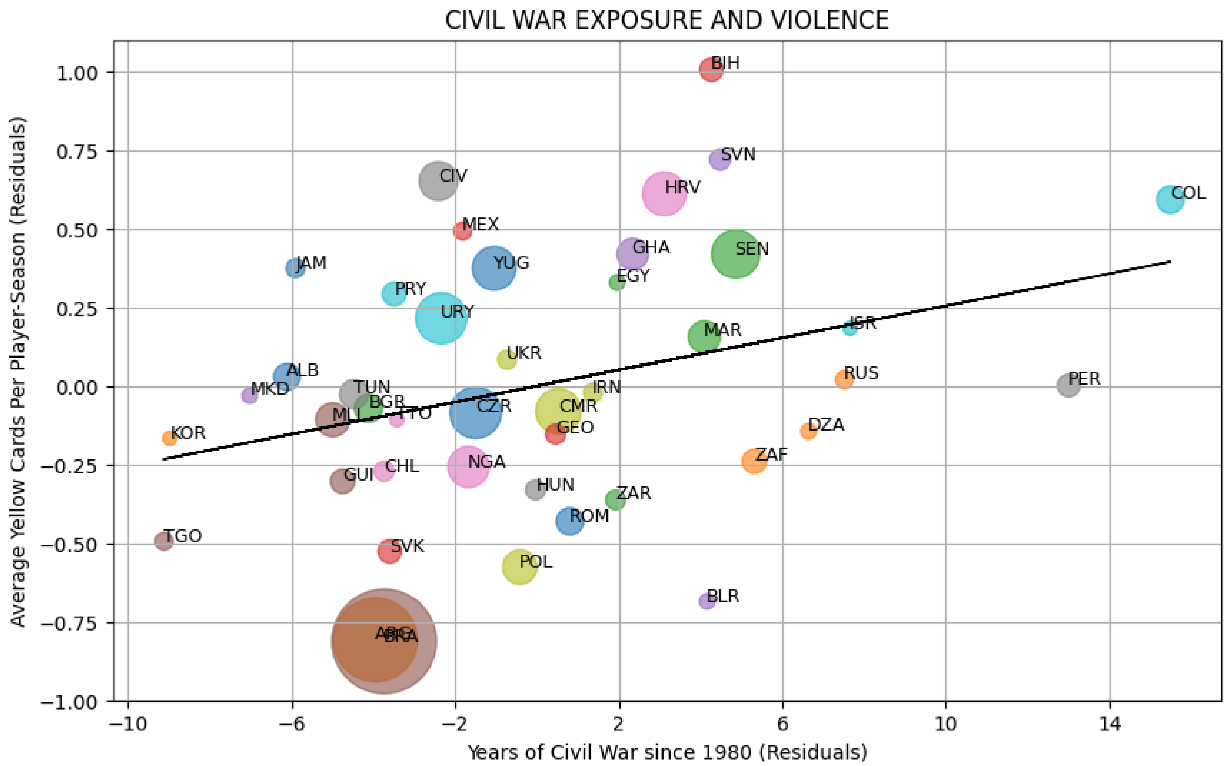


Figure 4

Yellow cards and civil war (conditional on control variables in Table 2, regression 1) – non-OECD countries, excluding Colombia, Iran, Israel, Peru, and Turkey.

Simply added another column

In [ ]:
```
wb_codes = pd.read_csv("/content/drive/My Drive/Econ 148 Project 3/Soccer_Replication/
soccer_data = pd.read_stata("/content/drive/My Drive/Econ 148 Project 3/Soccer_Replica
```

In [ ]:
```
wb_codes.rename(columns={'id': 'nation'}, inplace=True)
wb_codes.sort_values('nation', inplace=True)
soccer_data.drop(columns='num_country', inplace=True, errors='ignore')
soccer_data.dropna(subset=['contract'], inplace=True)

conditions = (
    (soccer_data['italian'] == 1) | (soccer_data['champions'] == 1) |
    (soccer_data['english'] == 1) | (soccer_data['french'] == 1) |
    (soccer_data['german'] == 1) | (soccer_data['spanish'] == 1)
)
soccer_data = soccer_data[conditions]
soccer_data['num_country'] = soccer_data.groupby('nationality')['player_id'].transform

soccer_data = soccer_data[soccer_data['oecd'] == 0]
exclude_nations = [23, 52, 54, 78, 101]
soccer_data = soccer_data[~soccer_data['nation'].isin(exclude_nations)]
```

```
merged_data = pd.merge(soccer_data, wb_codes, on='nation', how='left')

grouped = merged_data[merged_data['num_country'] >= 5].groupby('wb_code').agg({
    'yellow_card': 'sum',
    'civwar': 'sum',
    'nation': 'first',
    'num_country': 'first',
    'age': 'mean',
    'games_start': 'mean',
    'games_sub': 'mean',
    'defender': 'mean',
    'forward': 'mean',
    'midfield': 'mean',
    'goals': 'mean',
    'ln_contract': 'mean',
    'italian': 'mean',
    'champions': 'mean',
    'english': 'mean',
    'french': 'mean',
    'german': 'mean',
    'spanish': 'mean',
    'africa': 'mean',
    'asia': 'mean',
    'lac': 'mean',
    'east_europe': 'mean',
    'oecd': 'mean'
})

grouped['yellow_avg'] = grouped['yellow_card'] / grouped['num_country']
grouped['war_avg'] = grouped['civwar'] / grouped['num_country']

formula_yellow = 'yellow_avg ~ age + games_start + games_sub + defender + forward + mi
model_yellow = smf.ols(formula=formula_yellow, data=grouped).fit()
grouped['yellowhat'] = model_yellow.predict(grouped)

formula_civwar = 'war_avg ~ age + games_start + games_sub + defender + forward + midfi
model_civwar = smf.ols(formula=formula_civwar, data=grouped).fit()
grouped['warhat'] = model_civwar.predict(grouped)

grouped['yellow_res'] = grouped['yellow_avg'] - grouped['yellowhat']
grouped['war_res'] = grouped['war_avg'] - grouped['warhat']

plt.figure(figsize=(10, 6))
for idx, row in grouped.iterrows():
    plt.scatter(row['war_res'], row['yellow_res'], s=row['num_country'] * 10, alpha=0.
    plt.text(row['war_res'], row['yellow_res'], idx, fontsize=9)  # Add text labels


y = grouped['yellow_res']
x = grouped['war_res']
slope, intercept = np.polyfit(x, y, 1)
plt.plot(x, slope*x + intercept, 'k-', linewidth=1)

plt.ylabel('Average Yellow Cards Per Player-Season (Residuals)')
plt.xlabel('Years of Civil War since 1980 (Residuals)')
plt.title('CIVIL WAR EXPOSURE AND VIOLENCE')
plt.grid(True)
plt.xticks(np.arange(-6, 6, 4))
plt.yticks(np.arange(-1, 1.1, .25))
```

```
plt.grid(True)
plt.show()
```

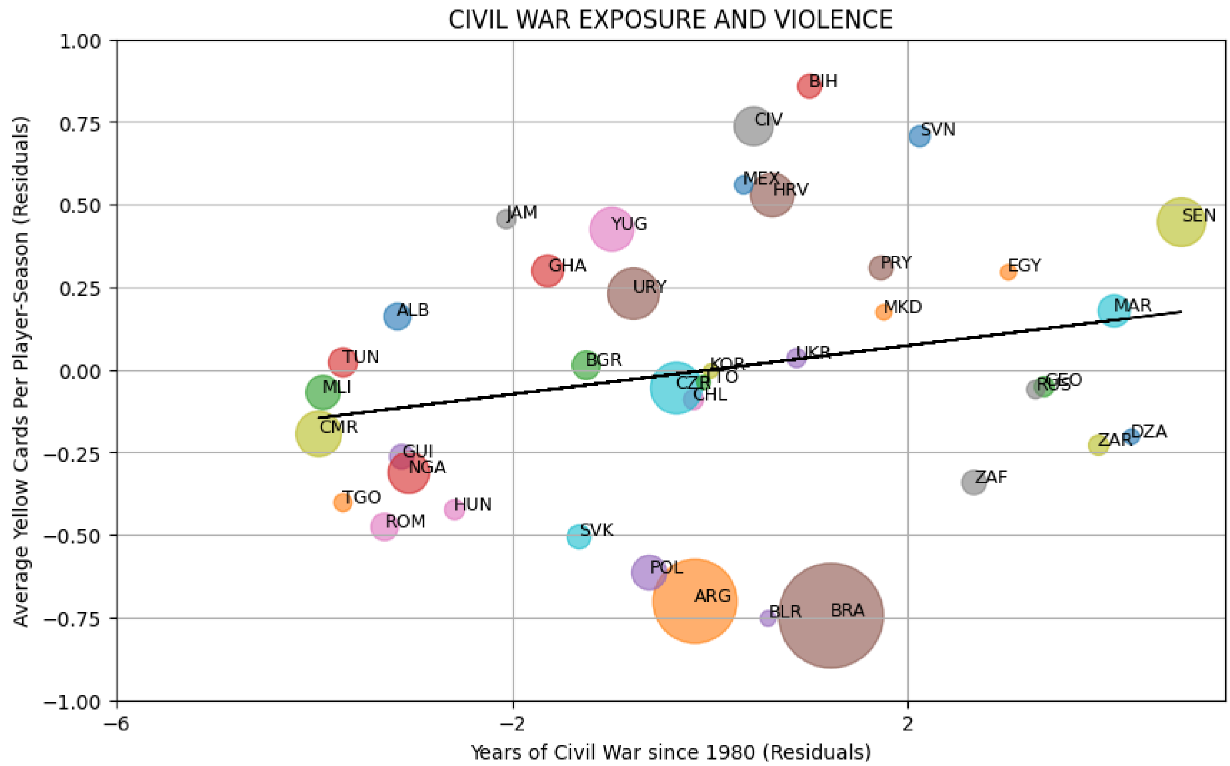


Table A1

```
In [ ]: wb_codes = pd.read_csv("/content/drive/My Drive/Econ 148 Project 3/Soccer_Replication/
soccer_data = pd.read_stata("/content/drive/My Drive/Econ 148 Project 3/Soccer_Replica

# Merge the soccer_data with wb_codes based on the country code
soccer_data_merged = soccer_data.merge(wb_codes, how='left', left_on='nation', right_c

# Define the conditions for filtering


condition = (
    soccer_data_merged["italian"].notna() |
    soccer_data_merged["champions"].notna() |
    soccer_data_merged["english"].notna() |
    soccer_data_merged["french"].notna() |
    soccer_data_merged["german"].notna() |
    soccer_data_merged["spanish"].notna()
) & (soccer_data_merged['num_country'] >= 5)

soccer_data_merged = soccer_data_merged[condition]
soccer_data_merged = soccer_data_merged.dropna(subset=["ln_contract"])
soccer_data_merged['Country with code'] = soccer_data_merged['nationality'] + ' (' +sc

# We are interested in the average number of yellow cards and civil war years per play
# We will first group by the country code and then calculate the mean for yellow cards
aggregated_data = soccer_data_merged.groupby('Country with code').agg(
    Observations=('player_name', 'size'),
    Yellow_cards=('yellow_card', 'mean'),
    Civil_war_years=('civwar', 'mean')
```

```
).reset_index()

# Since the table in the image has two separate sections, we might need to split the d
# and format it in a way that matches the provided table structure.
# However, we'll create a single dataframe and leave the splitting and styling for the

# Sort the countries alphabetically
aggregated_data_sorted = aggregated_data.sort_values(by='Country with code')

# Now, let's prepare it in a tabular format that can be printed or displayed similarly
# We will round the means to two decimal places to match the formatting in the example
aggregated_data_sorted['Yellow_cards'] = aggregated_data_sorted['Yellow_cards'].round(
aggregated_data_sorted['Civil_war_years'] = aggregated_data_sorted['Civil_war_years'].

# Output the data to be checked and for further processing
aggregated_data_sorted['Civil_war_years'] =  aggregated_data_sorted['Civil_war_years']
aggregated_data_sorted.rename(columns={'Country with code': 'Country', 'Yellow_cards':
aggregated_data_sorted
```

Out[ ]:

	Country	Observations	Yellow cards	Civil war years
0	Albania (ALB)	18	2.89	0
1	Algeria (DZA)	6	1.50	15
2	Argentina (ARG)	178	2.92	0
3	Australia (AUS)	28	2.57	0
4	Austria (AUT)	6	1.67	0
...	...	...	...	...
66	Turkey (TUR)	24	2.25	22
67	Ukraine (UKR)	9	1.44	0
68	United States (USA)	30	0.97	4
69	Uruguay (URY)	66	2.89	0
70	Wales (WAL)	26	2.19	13

71 rows × 4 columns

In [ ]:

```
num_rows = len(aggregated_data_sorted)
midpoint = num_rows // 2  # Find the midpoint to split the table

# Split the table into two
left_table = aggregated_data_sorted.iloc[:midpoint, :]
right_table = aggregated_data_sorted.iloc[midpoint:, :]

# Reset index on the right table to start from 0
right_table.reset_index(drop=True, inplace=True)

# Convert the left and right halves of the DataFrame to LaTeX
left_latex = left_table.to_latex(index=False, header=True, float_format="%.2f", longta
right_latex = right_table.to_latex(index=False, header=True, float_format="%.2f", long

# Manually combine the left and right LaTeX tables into a single LaTeX table with two
latex_code = "\\begin{table}[ht]\n\\centering\n"
```

```
latex_code += "\\caption{Countries and Players Represented in the Main Sample}\n"
latex_code += "\\label{tab:countries_players}\n"
latex_code += "\\begin{tabular}{llrrllrr}\n"
latex_code += "\\toprule\n"

# Add column headers
latex_code += " & ".join(left_table.columns) + " & " + " & ".join(right_table.columns)
latex_code += "\\midrule\n"

# Add rows from both tables side by side
for i in range(max(len(left_table), len(right_table))):
    left_row = " & ".join(left_table.iloc[i].astype(str)) if i < len(left_table) else
    right_row = " & ".join(right_table.iloc[i].astype(str)) if i < len(right_table) el
    latex_code += left_row + " & " + right_row + " \\\\\n"

latex_code += "\\bottomrule\n"
latex_code += "\\end{tabular}\n"
latex_code += "\\end{table}"
```

In [ ]:
```
latex_code
```

Out[ ]:
```
"\\begin{table}[ht]\n\\centering\n\\caption{Countries and Players Represented in the Main Sample}\n\\label{tab:countries_players}\n\\begin{tabular}{llrrllrr}\n\\toprule\nCountry & Observations & Yellow cards & Civil war years & Country & Observations & Yellow cards & Civil war years \\\\\n\\midrule\nAlbania (ALB) & 18 & 2.89 & 0 & Jamaica (JAM) & 9 & 1.78 & 0 \\\\\nAlgeria (DZA) & 6 & 1.5 & 15 & Japan (JPN) & 10 & 1.5 & 0 \\\\\nArgentina (ARG) & 178 & 2.92 & 0 & Macedonia (MKD) & 6 & 4.17 & 1 \\\\\nAustralia (AUS) & 28 & 2.57 & 0 & Mali (MLI) & 29 & 3.03 & 2 \\\\\nAustria (AUT) & 6 & 1.67 & 0 & Mexico (MEX) & 8 & 3.62 & 2 \\\\\nBelarus (BLR) & 6 & 1.5 & 0 & Morocco (MAR) & 26 & 3.15 & 10 \\\\\nBelgium (BEL) & 34 & 1.91 & 0 & Netherlands (NLD) & 118 & 2.07 & 0 \\\\\nBosnia and Herzegovina (BIH) & 14 & 2.93 & 4 & Nigeria (NGA) & 43 & 1.81 & 1 \\\\\nBrazil (BRA) & 277 & 2.44 & 0 & Northern Ireland (NIR) & 12 & 1.0 & 13 \\\\\nBulgaria (BGR) & 20 & 2.55 & 0 & Norway (NOR) & 20 & 1.75 & 0 \\\\\nCameroon (CMR) & 52 & 2.29 & 1 & Paraguay (PRY) & 14 & 2.43 & 1 \\\\\nCanada (CAN) & 7 & 3.71 & 0 & Peru (PER) & 13 & 1.38 & 19 \\\\\nChile (CHL) & 10 & 3.8 & 0 & Poland (POL) & 30 & 1.0 & 0 \\\\\nColombia (COL) & 19 & 4.79 & 26 & Portugal (PRT) & 68 & 3.03 & 0 \\\\\nCongo DR (ZAR) & 10 & 2.5 & 6 & Romania (ROM) & 19 & 1.21 & 1 \\\\\nCroatia (HRV) & 48 & 2.38 & 3 & Russia (RUS) & 8 & 1.75 & 13 \\\\\nCzech Republic (CZR) & 67 & 2.24 & 0 & Scotland (SCO) & 37 & 2.16 & 13 \\\\\nCÙte d'Ivoire (CIV) & 12 & 2.08 & 3 & Senegal (SEN) & 59 & 2.25 & 10 \\\\\nDenmark (DNK) & 58 & 1.84 & 0 & Serbia & Montenegro (YUG) & 48 & 2.83 & 3 \\\\\nEgypt (EGY) & 6 & 1.0 & 6 & Serbia (YUG) & 8 & 1.75 & 3 \\\\\nEngland (ENG) & 402 & 2.18 & 13 & Sierra Leone (SLE) & 4 & 2.0 & 10 \\\\\nFinland (FIN) & 24 & 1.08 & 0 & Slovakia (SVK) & 14 & 0.93 & 0 \\\\\nFrance (FRA) & 721 & 2.25 & 0 & Slovenia (SVN) & 11 & 1.64 & 0 \\\\\nGeorgia (GEO) & 10 & 3.2 & 4 & South Africa (ZAF) & 15 & 1.07 & 9 \\\\\nGermany (DEU) & 424 & 2.01 & 0 & South Korea (KOR) & 5 & 1.0 & 0 \\\\\nGhana (GHA) & 25 & 2.4 & 2 & Spain (ESP) & 742 & 2.91 & 5 \\\\\nGreece (GRC) & 22 & 2.14 & 0 & Sweden (SWE) & 35 & 1.77 & 0 \\\\\nGuinea (GUI) & 15 & 2.33 & 2 & Switzerland (CHE) & 49 & 2.41 & 0 \\\\\nHungary (HUN) & 10 & 0.9 & 0 & Togo (TGO) & 8 & 0.75 & 2 \\\\\nIceland (ISL) & 8 & 2.0 & 0 & Trinidad & Tobago (TTO) & 5 & 0.2 & 1 \\\\\nIran (IRN) & 9 & 2.33 & 19 & Tunisia (TUN) & 21 & 2.33 & 1 \\\\\nIreland (IRL) & 67 & 1.9 & 0 & Turkey (TUR) & 24 & 2.25 & 22 \\\\\nIsrael (ISR) & 5 & 4.8 & 26 & Ukraine (UKR) & 9 & 1.44 & 0 \\\\\nItaly (ITA) & 730 & 2.81 & 0 & United States (USA) & 30 & 0.97 & 4 \\\\\nIvory Coast (CIV) & 38 & 3.63 & 3 & Uruguay (URY) & 66 & 2.89 & 0 \\\\\n & & &  & Wales (WAL) & 26 & 2.19 & 13 \\\\\n\\bottomrule\n\\end{tabular}\n\\end{table}"
```

In [ ]:
```
!jupyter nbconvert --to html "/content/drive/My Drive/Econ 148 Project 3/Soccer
```