Risk Radar

Minor Data Science

Group 5 - 15/01/2024
Jamey Schaap (0950044)
Thomas Poelman (1008138)
Jarell Wespel (0999541)
Luc Karlas (1017799)
Maurits Hanhart (1009228)
Dominique Kuijten (1009466)

Preface

In the past months, a selected group of six people from Rotterdam University of Applied Sciences will need to find and provide answers to several research questions for their minor study in data science.

During the data selection phase, they faced competition from several project groups who shared the same idea about a data set. In order to secure their preferred data set, they had to present their idea in a pitch presentation in class. Hoping to be selected as the favourites.

Acknowledgment

This project would not have been possible without the guidance and the help of several individuals. We are indebted to our coach Andrej Dobrkovic whose guidance, support and expertise has enabled us to complete our project. We would also like to give a special thanks to Alexandra Wypych for her valuable feedback and insights throughout our project. We do like to acknowledge the feedback given by Tony Busker was very insightful and helped us better our project. We have learned a lot from this minor and would like to further extend our knowledge of this domain.

Abstract

The research paper "Risk Radar" aims to predict the risk for investors to invest in a given country by analysing political and economic factors through data science. Using machine learning models such as K-Nearest Neighbors, Support Vector Machines, and Artificial Neural Networks, the study examines the relationship between political stability, governance, and economic performance in influencing investment outcomes. The report provides investors with actionable insights for strategic decision-making in complex global markets, offering a nuanced understanding of risk variables. The comprehensive approach includes a literature review, data processing, and rigorous model testing to enhance the effectiveness of risk assessment.

The study starts off with data collection and preprocessing, utilising three datasets: Polity5, IMF investment and capital stock data, and population data. Key features selected for analysis include *polity2* (measuring democracy or autocracy), sustainability indicators reflecting political stability, and economic metrics such as *GDP* and *private investments*. Custom Python scripts are employed for extensive data preprocessing, addressing issues like inconsistent country names and missing values.

Model development involves testing various algorithms, with the Feedforward Neural Network emerging as the most promising. After hyperparameter tuning, the final model shows promising training and testing results, displaying minor overfitting. On the test dataset, the model achieves an accuracy of approximately 93%, although facing challenges in predicting 'high' labels due to limited data.

Feature importance analysis, conducted through the Shap Python library, offers valuable insights into the decision-making process of the model. A subset of 20 samples is evaluated 100 times due to time and hardware constraints.

In the validation phase, the 9-label version of the final model is compared against a 27-label version of the final model to test if the 9-label version is precise. The final model is precise with established error margins because the 9-label model would be *100%* correct and the 27-label model would be *95%* correct, thus having a difference of *5%*. The final model has also been tested on a custom validation dataset, which yielded an accuracy of 77% and an accuracy of 100% with established error margins.

In the financial sector, managing risk is crucial for stability and growth. Reducing uncertainty in investments is essential to promote a healthy economy. Accurate predictions of investment risks, grounded in political and economic data, are essential for proactive risk mitigation and investing. Historical events like the 2008 financial crisis underscore the significance of robust risk management practices. This research contributes to enhanced risk management, allows investors to anticipate potential threats and helps a more resilient financial ecosystem.

Contents

Introduction

1.1 Background

In the dynamic landscape of global investments, the ability to accurately assess and manage risks is very important for success. As investors navigate diverse geopolitical and economic terrains, understanding the complex interaction or delicate balance of variables influencing investment risk becomes most important.

This paper addresses this by looking into a comprehensive analysis that combines key political and economic indicators, aiming to construct a nuanced understanding of the diverse landscape of investment risk.

1.2 Research goal

The current investment environment is influenced by a combination of political and economic factors, each playing a pivotal role in shaping the opportunities and challenges that investors can face. Our research is motivated by the impact of the importance of political stability, governance structures, and economic performance in influencing investment outcomes. The relevance of this study lies in its potential to help investors with a more informed and better approach to risk assessment, helping the investor to make more strategic decisions in an increasingly complex global market.

1.3 Research questions

The main research question of this thesis is:

‘How can the investment risk of a country be predicted, based on political and economical data using machine learning?”

This investigation delves into the intricate relationships between political dynamics, economic performance, and the resultant impact on investment risks. Supporting this central exploration are sub-questions that guide our investigation into specific aspects of political stability, governance trends, and economic indicators, providing a comprehensive framework for analysis.

1 What kind of machine learning model is best used to predict the investment risk of a country?

a. What kind of machine learning models are available?
b. Which features should be used in the machine learning model?

2 What are the advantages and limitations of a risk predicting framework?

a. What are the pros and cons of a machine learning model?

3 How can a risk framework handle unexpected events?

a. What are unexpected events?

1.4 Research method

The initial step involves comprehending the data and formulating a well-defined research question. Subsequently, we proceed to establish specific objectives aimed at addressing the research question. The following steps focus more on statistics. In this phase, we will look for correlation and causation between our dependent and independent variables. The next phase entails a comprehensive exploration of existing literature to delve deeper into the identified problem. After that, we will assess the nature of the problem and determine which models are most suitable. Subsequently, we will test these models to select the best possible one. Following this, the (hyper)parameters of the chosen model will be tuned, further testing will be conducted, and the results will be validated. Finally, a conclusion will be written summarising the results.

1.5 Structure

The project is structured to help the reader's comprehension and navigation through the complexities of predicting investment risk based on political and economic data.

Chapter 2 delves into the findings from a comprehensive literature review, exploring the current landscape of machine learning models and decision-making in the context of predicting investment risk.

Chapter 3: Provides a comprehensive overview of the study’s solution design, covering aspects related to data collection, preprocessing and machine learning model development.

Moving on to Chapter 4, the model will be validated to ensure that the precision and accuracy is high enough for this project. It will also be compared against a custom validation dataset to test against data it has never seen.

Finally, Chapter 5 provides a summary and draws final insights from our research and solution of predicting investment risk based on political and economic data.

2. Literature review

This chapter aims to provide information on potentially usable machine learning models and their advantages and disadvantages through literature. In this paper some of the methods provided in Watson & Webster (2002) and Wolfswinkel et al. (2013) are used as a guide to search and analyse literature.

The first order of business was to formulate questions that need to be answered based on the literature research. After this, we created keywords and search queries to be used in Google Scholar, ResearchGate and normal Google search (for extra information on smaller things), which we used to find the papers we ended up using for the project. This allowed us to have a clear overview of the research process and come up with the right path for the project.

2.1. Risk Factor: A literature review

2.1.1. Literature search and research progress

To conduct the necessary literary research, the following research questions will need to be answered:

- What kind of model is best used to predict the investment risk of a country?
    - What variables define investment risk?
- How can a risk framework handle unexpected events?
- What are the limitations of a risk predicting framework?
    - Are there any real world applications of these types of frameworks? Where?

The research and literature selection process went as follows. Firstly, keywords will be written down based on the questions provided above. Afterwards, a small table will be made with the search queries to be used. When the research has been done and usable papers have been found, a table will be made with the findings, filtered by relevance, and relevant page numbers will be provided for every paper. The following keywords were used:

- Investment risk
- Political risk
- Country risk
- Country grade
- Limitations risk prediction framework
- Country risk
- Country risk
- Risk assessment
- Risk analysis
- Risk rating
- Risk variables

Based on these keywords, several search queries were used:

- Predict country risk model
- Investment risk of a country prediction model
- "Investment risk" algorithm
- "Sovereign risk" "machine learning"
- "Country risk" machine learning model
- "Country risk" "Support vector machine"
- Country risk assessment
- Country risk
- Risk limitations
- Country risk analysis investment machine learning
- Risk prediction
- Predicting country investment risk
- Country investment risk

After a thorough literature search, numerous papers were found relating to the questions. The table below provides an overview of all the papers, and to which question they relate.

Title	**Authors**	**Prediction model**	**Handle unexpected events**	**Limitations of model**	**Result and method of models**
Country Risk analysis: a survey of the quantitative methods	Hiranya K Smith	P6 - p17		P17 - p19	P6 - p17
Country Risk and Foreign Direct Investment	Duncan H. Meldrum	p2 - p6		p10 - p13	
Country risk measures: how risky are they?	Jennifer M. Oetzel, Richard A. Bettis, Marc Zenner	p10 - p14	p7 - p9	p14 - p15	
Assessing China’s Investment Risk of the Maritime Silk Road: A Model Based on Multiple Machine Learning Methods	Xu J, Zhang R, Wang Y, Yan H, Liu Q, Guo Y, Ren Y,	p4 - p13			p13 - p14

Risk prediction models: How they work and their benefits	Donald Farmer	p1			
Credit Growth, the Yield Curve and Financial Crisis Prediction: Evidence from a Machine Learning Approach	Kristina Bluwstein, Marcus Buckmann, Andreas Joseph, Miao Kang, S. Kapadia, Özgür Simsek	p15 - p18			p23 - p27

2.1.2. Which models are best used to predict the investment risk of a country?

To identify the optimal model for predicting a country's investment risk, we reviewed various research papers. One such paper that contributed to the understanding of the problem at hand and helped better the approach to tackling it was "Country Risk Analysis: A Survey of Quantitative Methods" by Nath, published in 2008. This paper provided valuable insights into how economists address this challenge, ultimately helping in improving the understanding and tackling of this project.

A country's risk can be defined and measured in many different ways. In general, it refers to the risk associated with those factors which determine or affect the ability and willingness of a particular country to fulfil their obligations towards one or more foreign leaders and/or investors. In general, a country’s risk is defined and split up into two factors: political, economic and financial risk. Therefore, it can be classified as qualitative or quantitative. However, most agencies will most likely combine these sources of information into a single index or rating known as a country risk score.

Logit Model

To calculate a country’s risk factor, economists used some different methods than what programmers would use. In Nath (2008), they used Logit analysis, Probit analysis and Tobit analysis. These will be explained using excerpts from the paper itself.

“The basic assumption of this approach is that the relationship between the probability of debt rescheduling and a set of explanatory variables can be described by the following functional form that represents a logistic distribution:

$$Pr(Y_i = 1) = P_i = \frac{1}{1 + exp\left[-\left(\beta_0 + \sum_{j=1}^{k} \beta_j X_{ij}\right)\right]}, \qquad i = 1, 2, 3, \ldots\ldots, n$$

where $\beta_0 + \sum_{j=1}^{k} \beta_j X_{ij} W$ represents a linear combination of k explanatory variables and a set of coefficients β = (β0 , β1 , ...) which are to be estimated, Yi = 1 for rescheduling cases and Yi = 0 for non-rescheduling cases Note that i indexes country and n is the total number of countries. It is assumed that there is some linear combination of independent variables that is positively related to the probability of rescheduling.”

Probit Model

“Probit analysis is very similar to the logit model except for the fact that the relationship between the probability of debt rescheduling and the explanatory variables is represented by a normal distribution function instead of a logistic distribution function. Thus,

$$Pr(Y_i = 1) = P_i = F(Z_i) = \int_{-\infty}^{Z_i/\sigma} \frac{1}{\sqrt{2\pi}} exp\left(-\frac{t^2}{2}\right) dt$$

where . $Z_i = \beta_0 + \sum_{j=1}^{k} \beta_1 X_{ij}$ and σ is the standard deviation of the distribution to be estimated. Both logit and probit analysis suffer from the lack of any explicit criterion for selecting the critical probability value for distinguishing rescheduling from non-rescheduling countries.”

Tobit Model

“The studies that use the logit and probit model are mainly concerned with predicting the timing of debt rescheduling by a developing country. However, using a Tobit model can help explain both the quantity and timing of a debt rescheduling. A Type 2 Tobit Model suggested for this purpose assumes that the probability of country i rescheduling its debt in a given time period can be represented by a probit equation: $Y_i^* = \beta_0 + \sum_{j=1}^{k} \beta_1 X_{ij} + \varepsilon_i$ where Y*i takes the value 1 if rescheduling takes place and 0 otherwise, and X’s are the variables that influence the rescheduling decision. The quantity of rescheduling is given by linear regression.”

Another paper was Assessing China’s Investment Risk of the Maritime Silk Road, Xu et al. (2022). This paper gave a good inside in machine learning models Support Vector Machine (SVM), XGBoost (XGB), LightGBM, Random Forest, and K-Nearest Neighbors (KNN), as well as a deep learning model (Deep Neural Network or DNN) for assessing China's investment risk in the Maritime Silk Road region.

Using this paper as one of the basis from which to select a usable model. Here it is discussed what the most important findings were and the result of the paper.

Handling Multifaceted Risk Factors:

When investing abroad, investors need to inspect various aspects such as social, economic, legal, diplomatic, religious, and war-related factors. Machine learning models, especially ensemble methods like Random Forest and boosting algorithms like XGBoost and LightGBM, are known for their ability to handle complex relationships and multiple input features. This makes them suitable for capturing the diverse set of risk factors associated with foreign investments.

Non-Linearity and complex data:

The paper highlights the need for considering non-linear relationships and the complexity of the variables in the context of investment risk. SVM, XGBoost, LightGBM, and Random Forest are all capable of capturing non-linear patterns and performing.

Utilising Historical Data for Risk Zoning:

The study uses historical data from ICRG and OFDI to create risk zoning maps for different years. Machine learning models are effective at learning patterns and trends from historical data, allowing for the identification of risk zones and changes over time.

Integration of Various Risk Factors:

The factors considered for risk assessment include government stability, socioeconomic conditions, investment profiles, internal conflict, external conflict, corruption, religious tensions, law and order, ethnic conflict, and bureaucratic quality. Machine learning models, especially ensemble methods, can integrate information from diverse sources and make predictions based on multiple variables.

Comparison and Evaluation:

The paper (Xu et al., 2022) aims to conduct a comprehensive analysis by comparing the performance of different machine learning models. Identifying the model or models that provide the most accurate predictions for China's investment risk in the Maritime Silk Road region.

Based on the results and discussion in the paper Assessing China’s Investment Risk of the Maritime Silk Road, two machine learning models, namely K-nearest neighbours (KNN) and Support Vector Machine (SVM), were prominent in predicting country investment risk. Here's a summary of the results:

1. Overall Prediction Performance:
    - KNN and SVM demonstrated competitive performance among multiple machine learning prediction models.
    - Both models outperformed other methods like XGBoost (XGB), LightGBM, Random Forest (RF), Logistic Regression, and Deep Neural Network (DNN).
    - SVM performed well in predicting high risk, while KNN excelled overall, especially for high-level risk.
2. High Risk Prediction:
    - Both SVM and KNN models achieved optimal values for prediction indicators when facing high risks.
    - SVM, particularly, had high precision, recall, and accuracy in predicting high risk.
    - XGB and LightGBM models had less ideal results in predicting high risk.

3. Low Risk Prediction:
    - KNN models achieved the optimal results for prediction indicators when facing low risks.
    - About every model had high precision, recall, and accuracy in predicting low risk.

The results below are the results from Xu et al. (2022). This means that they are not entirely the same when tested on the project data, but are used here to show contenders for possible models. These models will be tested further on the project data.

Indicator	SVM	XGB	LightGBM	RF	KNN	Logistic	DNN
Accuracy	0.75	0.70	0.71	0.77	0.86	0.42	0.71
F1	0.75	0.71	0.71	0.78	0.86	0.42	0.71
Precision	0.78	0.72	0.73	0.80	0.86	0.44	0.73
Recall	0.75	0.70	0.71	0.77	0.86	0.42	0.71
MAPE	9.1%	20.3%	18.9%	19.1%	4.5%	38.5%	13.7%

Table 2.1) Evaluation Results of Each Prediction Model, Table 2 from Xu et al. (2022)

Indicator	SVM	XGB	LightGBM	RF	KNN
Accuracy	0.88	0.5	0.5	0.62	0.88
F1	0.93	0.67	0.67	0.77	0.93
Precision	1.00	1.00	1.00	1.00	1.00
Recall	0.88	0.5	0.5	0.62	0.88

Table 2.2) High Risk Predictive Model Evaluation Results, Table 3 from Xu et al. (2022)

Indicator	SVM	XGB	LightGBM	RF	KNN
Accuracy	0.74	0.77	0.79	0.82	0.91
F1	0.85	0.87	0.88	0.9	0.95
Precision	1.00	1.00	1.00	1.00	1.00
Recall	0.74	0.77	0.79	0.82	0.91

Table 2.3) Low Risk Predictive Model Evaluation Results , Table 4 from Xu et al. (2022)

A comparison of KNN and SVM:

- Both KNN and SVM exhibited strong predictive capabilities, with KNN having a slight edge in overall performance.
- KNN and SVM performed exceptionally well for high-level risk, making it more concerning for stakeholders.

A comparison of deep learning- and machine learning models:

- Deep learning models, especially the deep neural network (DNN), generally had inferior predictive performance compared to KNN and SVM.
- The poor performance of the DNN model could be attributed to the relatively small amount of data used in the study. We also don’t have a large amount of data and variables, so it's the question if we should consider this?

The paper concludes that as China expands its foreign investment, effective prediction of investment risks is crucial. The proposed model, particularly based on the K-nearest neighbours (KNN) and Support Vector Machine (SVM) algorithms, provides a reliable link between the Outward Foreign Direct Investment (OFDI) indicator and the International Country Risk Guide (ICRG) indicators, offering precise and objective predictions of investment risk. Both KNN and SVM models not only predict an objective and reasonable investment risk level but also provide valuable predictions and suggestions for stakeholders. There is however a limitation regarding this paper, this is because its focus on China's interests in foreign investments might limit the overall applicability and relevance of the risk assessment for other stakeholders. Also, the availability of the data used in the study is restricted, which raises the question of whether similar results could be obtained with a comparable dataset.

This paper is, however, not the only paper we used to answer this question. "Credit Growth, the Yield Curve and Financial Crisis Prediction: Evidence from a Machine Learning Approach" by Bluwstein et al. (2020) offers valuable insights into predicting the investment risk of a country using machine learning models. This study gives insight into the application of machine learning techniques to macrofinancial data across various countries, providing a perspective on the effectiveness of these models in assessing economic and financial risks.

The study found that neural networks (NN), as part of the broader machine learning approach, were effective in identifying complex, non-linear relationships in the data that traditional models might overlook. NN’s ability to process large datasets and learn from them made them particularly adept at handling the multifaceted nature of macroeconomic and financial indicators.

SVMs were also utilised in this research, known for their effectiveness in classification and regression problems. SVMs are particularly adept at finding the optimal hyperplane for data classification, which is crucial in categorising different levels of investment risk based on economic indicators.

The paper by Bluwstein et al. (2020) demonstrates that machine learning models, including neural networks and support vector machines, offer substantial improvements over traditional econometric models in predicting financial crises and assessing investment risk. The use of a comprehensive dataset incorporating both economic and financial indicators allows for a more nuanced understanding of the factors that contribute to a country's investment risk. These findings are valublue for policymakers and investors looking at risk associated with economic downturns and financial instability and so assigning a risk score to a country.

Table 2.4 was created to provide an overview of machine learning algorithm performance as discussed in reviewed papers. This table gives the weak and strong points which are relevant to the specific use case.

Algorithm	Strong Points	Weak Points
SVM	● SVM classifiers perform well in high-dimensional space ● Suitable for a small dataset	● Limited to two-class problems ● SVM doesn’t work well when there is no margin of separation between classes
KNN	● Suitable for multi-class problems ● No assumptions are made ● One hyperparameter ● Works well with heterogeneous distributed points	● Works well with a low K value, but suffers from overfitting ● Features need to have the same scale ● Doesn’t perform well when the distribution of the classes is uneven ● Sensitive to outliers ● Is not capable of dealing with missing values
Neural Networks	● Handling missing data ● Perform well in high-dimensional space ● Feature extraction ● Handling non-linear relationships ● Predictive modelling	● Time-consuming parameter tuning ● Prone to overfitting ● Unforeseen consequences

Table 2.4) Strong and weak points per machine learning algorithm.

What variables define (country) risk?

Based on the findings from the study by Glova et al. (2020), a country's risk can be analysed and defined through a combination of various economic and political factors. While the common approach to defining country risk includes categories like economic risk, transfer risk, exchange rate risk, location or neighbourhood risk, sovereign risk, and political risk, the study suggests focusing on specific measurable variables that are crucial in assessing country risk, especially in the context of European countries. These variables include GDP per capita, inflation, gross government debt, current account balance, and international investment position, along with a political control index of corruption and the rule of law. It's important to note that some common risk categories like location or neighbourhood risk may not be directly available or applicable in all datasets. Therefore, the study's approach can be used as a guide to select the most relevant variables available in your data to create a comprehensive and tailored model for calculating a country's risk factor. Expanding on the findings from the Glova et al. (2020) study, which emphasises the significance of specific economic and political factors in assessing country risk, it is clear that such an approach is consistent with broader research in the field. Another study, "foreign direct investments under impact of political risks: theoretical survey" by Ayhan (2019), further supports this view. Ayhan highlights that political risk indicators, such as the level of democratisation, political/government instability, and internal and external conflicts, significantly affect decisions on investment and the risk.

2.1.3. How can a risk framework handle unexpected events?

Based on the papers to understand AI-models, predicting unexpected events, especially on a worldwide scale, is either so complex it would be completely out of scope, or just not possible to even do. Therefore, it was decided not to factor these unexpected events (such as wars) into future predictions, as this will increase the complexity of the project far beyond what is reasonable within the given timeframe.

2.1.4. What are the limitations of a risk predicting framework?

As stated above, a big limitation of this type of framework, especially one is handling unexpected events such as wars or pandemics. They can be seen in past data available, but cannot be predicted easily without going out of scope.

Are there any real-world applications of these types of frameworks?

From the information that was gathered, it seems that so far, there are some risk-predictors used, for example in the paper about China shown above, but so far these seem to not be all too widespread or supremely accurate, as predicting (investment) risk is a very complex thing, and is dependent on lots of (often not entirely available or accurate) variables.

2.3. Conclusion

After reviewing all of the collected papers, the decision was made to select and test three contenders on the datasets in order to determine the most suitable model for the specific task. The task is a multiclass classification problem with supervised learning, as the risk factor has been labelled with labels ranging from low to high, with those categories also being split into three (more on this in chapter 3.2). These three contenders, which will be discussed in greater detail in chapter three of this paper, are as follows:

- K-Nearest Neighbors (KNN): KNN is known for its simplicity and ability to classify data points based on their proximity to other data points in the feature space. It offers an intuitive approach to classification.
- Support Vector Machines (SVM): SVMs were chosen based on their demonstrated effectiveness in the financial risk prediction domain, as evidenced by Bluwstein et al. (2020). They excel at capturing non-linear relationships and classifying data into distinct categories.
- Artificial Neural Networks (ANN), including Feedforward Neural Networks (FNN): ANNs, specifically FNNs, were selected due to their success in handling large and complex datasets. They have the capacity to uncover intricate patterns and relationships within data, making them a strong candidate for investment risk prediction.

3. Solution Design

3.1 Data

Our data consists of three datasets: the Polity5, IMF investment & capital stock and population per country datasets. The Polity5 dataset was provided by the project course of the Data Science minor at Rotterdam University of Applied Sciences (RUAS), the IMF investment & capital stock was retrieved from the website of the International Monetary Fund (IMF) and the population per country dataset was retrieved from the website of the World Bank.

Polity5 contains data from 1800 to 2018, while the IMF dataset contains data from 1960 to 2019. Therefore, some data needs to be excluded due to this disparity, but that is acceptable because recent data is more applicable to the problem that is addressed for this problem. Information from before the industrial revolution and information age is simply not as useful. All these datasets combined contain a lot of features. Based on literature research and a pair plot generated from features in the dataset (see Appendix B), this set of features has been reduced to those that are most applicable to our problem.

3.1.1 Features used

3.1.1.1 From the Polity5 dataset

Polity2
Polity2 measures how democratic (10) or autocratic (-10) a country is. This score is derived from the Polity feature. We picked this feature instead of polity, because it contains less missing and no special values. It just ranges from -10 to 10 making it easier to work with. A score of -10 to -6 classifies as autocratic, -5 to 5 as anocratic and 6 to 10 as democratic.

Durable
Durable captures how many years a country has gone without regime change. This means that it takes data from previous years into account. A low durable score means that this country has been unstable in the recent past, thus making it a higher risk at this moment.

Fragment
Fragment depicts how many separate polities exist within the territory of the country. A high fragmentation could indicate that the country is currently unstable.

3.1.1.2 Calculated based on the Polity5 dataset

Government instability
Government instability counts how many times a country has had a regime change in the past. A country with a history of instability could be deemed as a higher risk than a country with no previous instability.

3.1.1.3 From the IMF investment & capital stock dataset

All values within this chapter are valuta denoted using constant 2017 international dollars, in other words inflation corrected to 2017 international dollars. This is done to take inflation into account so that rows for example from 1960 and 2000 can be compared fairly, since the value of the international dollar in 2000 is different from the value in 1960. In the dataset, these columns are named with the *_rppp* suffix.

GDP
Gross domestic product (GDP) is the total monetary or market value of all the finished goods and services produced within a country's borders in a specific time period. The calculation of a country's GDP encompasses all private and public consumption, government outlays, investments, additions to private inventories, paid-in construction costs, and the foreign balance of trades. Exports are added to the value and imports are subtracted. (Fernando, 2023)

General government investment
Government investment creates a public infrastructure that is essential for long-term economic growth and societal well-being. Governments spend money on building roads, housing, schools and hospitals, as well as communications networks. In addition, governments can provide grants (transfers) to the private sector to encourage their investment activities. General government investment (gross fixed capital formation), is expressed in billions ("Government At A Glance 2011", 2011).

Private investment
Private investments within a country refer to investments made in privately-held companies; some examples are; private equity funds, private debt funds, real estate investment funds, and direct company investments. Private investments can be an excellent alternative to public investments like, stocks, bonds, mutual funds, and exchange traded funds. Private investment (gross fixed capital formation) is expressed in billions.

Private investments can provide an opportunity for investors to support innovative and entrepreneurial companies that may have a positive impact on society. Investing in private companies can help support job creation, economic growth, and technological innovation.Some private investment sectors include, healthcare, multi-unit real estate, industrial real estate, biotechnology, energy, and manufacturing (Padalino, 2023)

Public-private partnership
In a public-private partnership, companies and government or civil society organisations work together for example housing in a country or the infrastructure. This partnership is mostly financial based in the way of donations and sponsorships. Public-private partnership is based on two main principles. Firstly, That both parties invest in the project in a financial way (manpower, a budget for the materials etc.) and in an expertise related sense in the way of knowledge and network. The second principle is that both parties participate in a social way and often a commercial way of presenting. Public-private partnership is expressed in billions (Netherlands Enterprise Agency, RVO, z.d.).

3.1.1.4 Calculated based on the IMF investment & capital stock and population datasets

GDP per capita
Gross domestic product (GDP) per capita is an economic metric that breaks down a country’s economic output per person. GDP per capita is calculated by dividing the GDP of a country by its population. Countries that have a higher GDP per capita tend to be more developed than countries with a low GDP per capita (Fernando, 2023).

3.2 Data Preprocessing

A Python script was developed to merge and preprocess datasets into a manageable format suitable for feeding into a machine learning model. The benefit of using a script instead of directly modifying the datasets is that it allows us to rectify past errors, make changes and that it is reproducible. To provide further explanations of which steps are taken, let's detail the steps involved in processing and merging the data.

The datasets did not contain ISO 3166 country codes, which means that the rows had to be merged based on the name and year of a given country. The country names across all datasets were inconsistent, which led to problems like Russia being named as the Soviet Union. To resolve this issue, a mapping was created where country names were renamed to standardised names, like the renaming of the Soviet Union to Russia.

The column *government_type* was introduced, categorising the polity2 (-10 to 10) score as either autocratic (-10 to -6), anocratic (-5 to 5) or democratic (6 to 10). This column was mainly used during the plotting of graphs as the colour/hue, because it gave a better view/understanding of trends and clustering.

Several rows were dropped that contained null/NA values in the columns polity2, durable and/or GDP per capita. It is critical that these columns contain values so that our model can be trained correctly. Still, it is important to note that it is unlikely that NA values in these columns were encountered ,as both polity2 and durable came from the preprocessed Polity5 dataset.

Next, a base 10 logarithm was applied to the columns; GDP per capita, GDP and sum of investment. Finally the columns; Durable, GDP per capita, GDP, polity2, sum of investment, government instability, General government investment, Private investment and Public-private partnership were normalised.

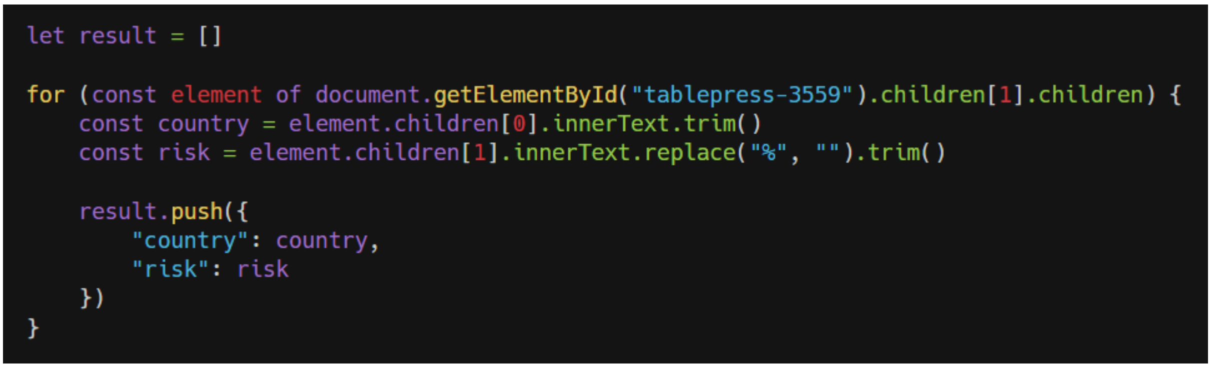
3.2.1 Risk factor

A formula was created to label each row of the dataset with a risk factor. It is important to note that this approach would not be suitable in the real world as it is likely to skew the results. In real-world scenarios, experts would typically provide and/or use accurate data labels. However, the problem arose that labelled data was not available and labelling the data in a more comprehensive approach was beyond the scope.

$$political\_risk \ = \ -((|polity2| \,/\, 10) \ + \ gov\_durability \ - \ (fragment \,/\, 3) \ - \ gov\_instability)$$
$$economical\_risk \ = \ -(norm\_log\_gdp\_pc \ + \ norm\_log\_gdp \ + \ norm\_log\_sum\_investment)$$
$$risk \ = \ political\_risk \ + economical\_risk$$

The risk factor is then normalised to a value ranging from 0 to 1. This is done so that the risk factor can be easily broken down from numerical values into classes, which can then be used directly by the classification algorithm.

To validate the accuracy of the calculated risk factor, a test was performed by testing its correlation with another risk factor. The alternative risk factor is taken from Visual Capitalist's list of country risks in 2023, Neufeld (2023). The values were scraped from their website using the code in Figure 3.1.

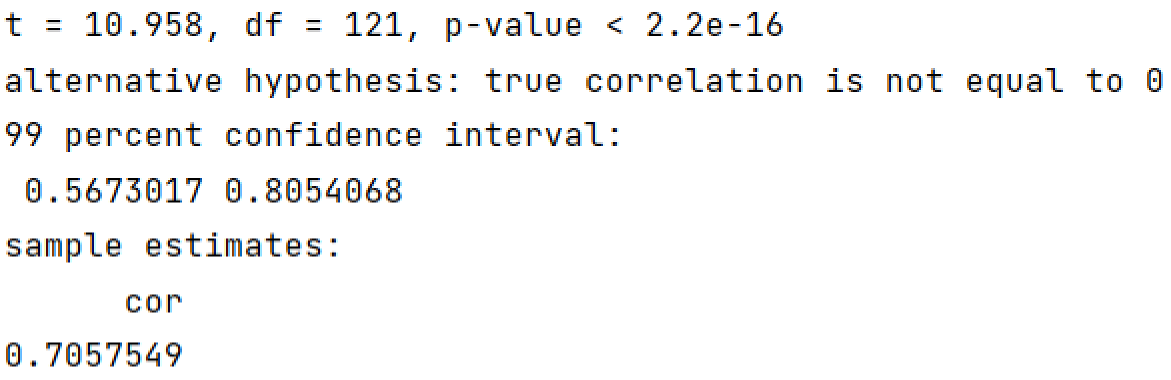


```
let result = []

for (const element of document.getElementById("tablepress-3559").children[1].children) {
    const country = element.children[0].innerText.trim()
    const risk = element.children[1].innerText.replace("%", "").trim()

    result.push({
        "country": country,
        "risk": risk
    })
}
```

Figure 3.1) The JavaScript code used to scrape Visual Capitalist' risk factor.

To test the correlation, a hypothesis was formulated as depicted in Figure 3.2.

> *H0: There is no significant correlation between Visual capitalist's risk factor and our risk factor.*
> *H1: There is a significant positive correlation between Visual capitalist's risk factor and our risk factor.*



```
t = 10.958, df = 121, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
99 percent confidence interval:
 0.5673017 0.8054068
sample estimates:
      cor
0.7057549
```

Figure 3.2) Hypothesis test between our risk factor and risk factor of Visual Capitalist.

Performing this test with a 99% confidence interval determines that our risk factor has sufficient correlation with the risk factor of visual capitalist. While the formula is not perfect, it should be sufficient to create accurate labelling of the data for the machine learning model.

As previously mentioned, a risk factor from industry experts is preferable, but this is not feasible within the scope of this project.

The numerical score, ranging from 0 to 1, is then converted into nine equally sized classes, which serve as labels for the machine learning model. These classes are organised as follows:

- *low_0, low_1, low_2*
- *mid_0, mid_1, mid_2*
- *high_0, high_1, high_2*

Labelling the data in this way turns the problem into a multiclass classification problem. If the risk score had remained numerical, this problem would have been a regression problem. That would also have been a valid approach. When examining the main research questions, it was found that every other study also uses a classification approach to this problem (Allianz, n.d.), which reinforces the use of this approach.

This approach aims to reduce incorrect predictions. It is expected that certain countries may be incorrectly predicted due to the complexity of the problems that certain countries face.

A margin of error was established, where a predicted class would still be considered correct if it differed -1 or +1 from the predicted class. Since our problem is a 9-label multi-class classification problem, a difference of -1 and +1 will have a small impact on the results compared to a difference of -1 and +1 in a 3-label classification problem.

3.2.2 Government instability

Government instability is calculated by analysing the *durable* column. It counts the amount of times where the *durable* value is equal to 0 within a specified lookup period. Through testing it was determined that a search period of 60 years provided the most optimal results for this calculation.

3.2.3 Machine learning dataset

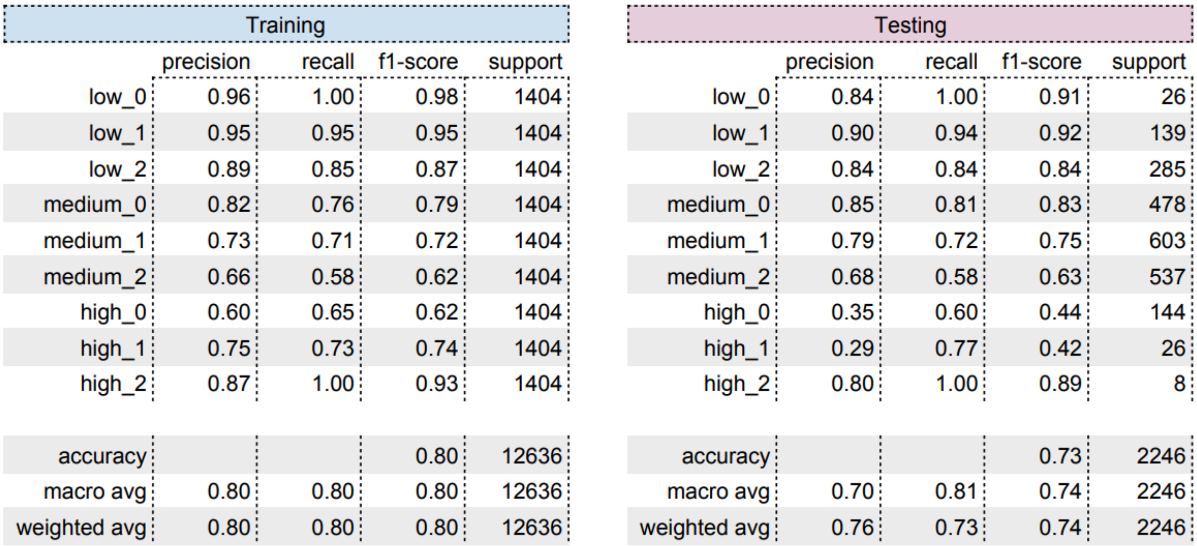
An example of the machine learning dataset is shown in Appendix C. It contains ten rows of ten country-year pairs, with each row containing preprocessed data of the attributes mentioned in Chapter 3.1 Data and the labels encoded via ordinal coding. The labels are coded in the order of low_0 to high_2 to their numerical value 0 to 8.

3.3 Model

3.3.1 Tested models

Over the course of this project, several models were tested to improve understanding and refine the approach to finding the best-suited model.

Support vector machine
After literature research, SVM emerged as a potential model. However, with an SVM model there is a challenge because it does not natively support multi-class classification. However, this can be overcome by using the one-to-one approach (Brownlee, 2021), which decomposes the multi-class problem into multiple binary classifications.



	Training			
	precision	recall	f1-score	support
low_0	0.96	1.00	0.98	1404
low_1	0.95	0.95	0.95	1404
low_2	0.89	0.85	0.87	1404
medium_0	0.82	0.76	0.79	1404
medium_1	0.73	0.71	0.72	1404
medium_2	0.66	0.58	0.62	1404
high_0	0.60	0.65	0.62	1404
high_1	0.75	0.73	0.74	1404
high_2	0.87	1.00	0.93	1404
accuracy			0.80	12636
macro avg	0.80	0.80	0.80	12636
weighted avg	0.80	0.80	0.80	12636

	Testing			
	precision	recall	f1-score	support
low_0	0.84	1.00	0.91	26
low_1	0.90	0.94	0.92	139
low_2	0.84	0.84	0.84	285
medium_0	0.85	0.81	0.83	478
medium_1	0.79	0.72	0.75	603
medium_2	0.68	0.58	0.63	537
high_0	0.35	0.60	0.44	144
high_1	0.29	0.77	0.42	26
high_2	0.80	1.00	0.89	8
accuracy			0.73	2246
macro avg	0.70	0.81	0.74	2246
weighted avg	0.76	0.73	0.74	2246

Table 3.1) Confusion matrices of the fitted SVM model.

For this model, the data was split into a training (*70%*) and test set (*30%*), with the training set using oversampling because there is limited data available for certain classes. Although the overall accuracy is decent, it faces challenges in distinguishing between the medium and high risk classes. The model seems to have particular difficulty in effectively differentiating between the higher risk classes (see Table 3.1).

K-nearest neighbour
During the literature review, KNN also emerged as a potential model. It was noted that KNN is well-suited for multi-class classification. To determine if this model is suitable for this use case, tests were performed with different *K* values to identify the value that provided the best performance. The following formula was used as a guideline:

$$k = \frac{\sqrt{N}}{2}$$

N represents the number of rows in the dataset. Because the data consists of 7,500 rows, applying this formula yields an initial K value of 43. It's important to note that selecting an appropriate *K* value is essential, as opting for a value that is too small may result in unstable decision boundaries, while a value that is too large could lead to overgeneralization. To evaluate the initial *K* value, a test was made to compute the accuracy for each *K* value within the range of 1 to 1000 (see Figure 3.3).

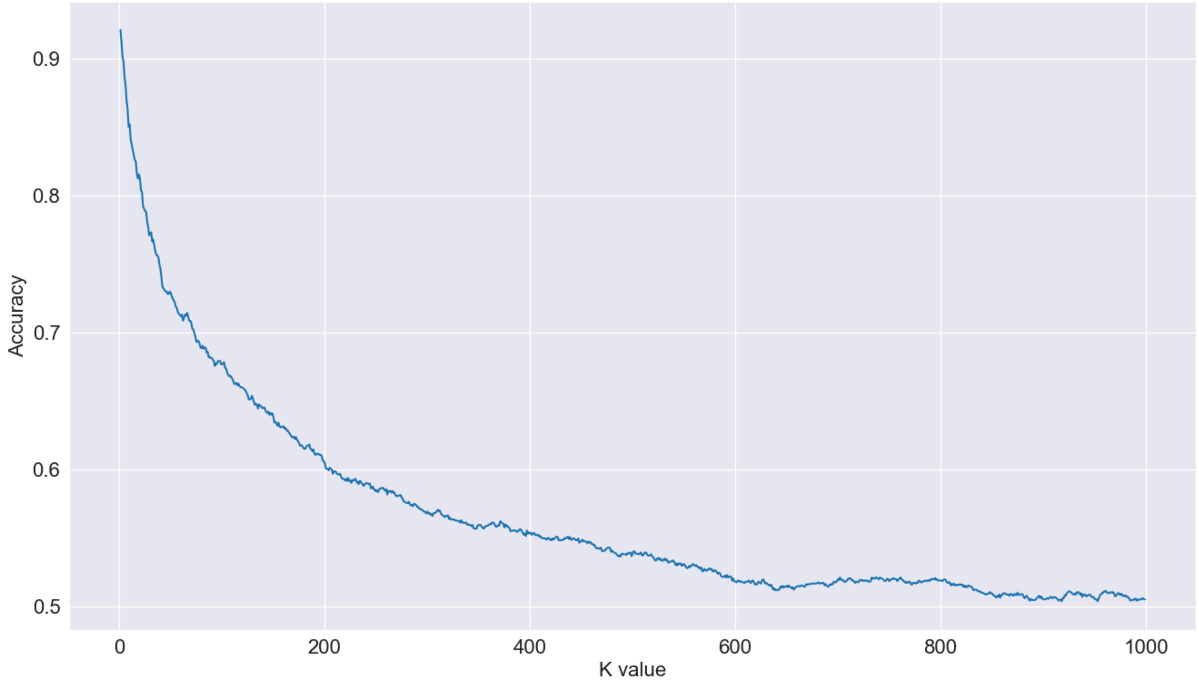
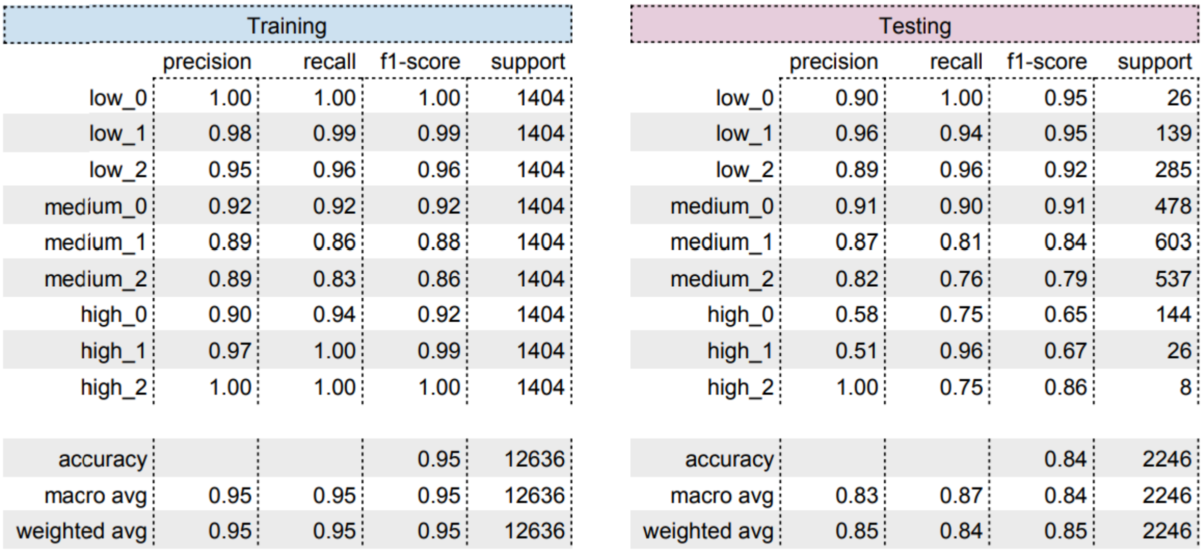


Figure 3.3) Graph depicting the accuracy per K-value from 1 to 1000.

After testing, a *K* value of 10 demonstrated the best performance, although a bit overfitting, leading to further experimentation with this value.



Training				
	precision	recall	f1-score	support
low_0	1.00	1.00	1.00	1404
low_1	0.98	0.99	0.99	1404
low_2	0.95	0.96	0.96	1404
medium_0	0.92	0.92	0.92	1404
medium_1	0.89	0.86	0.88	1404
medium_2	0.89	0.83	0.86	1404
high_0	0.90	0.94	0.92	1404
high_1	0.97	1.00	0.99	1404
high_2	1.00	1.00	1.00	1404
accuracy			0.95	12636
macro avg	0.95	0.95	0.95	12636
weighted avg	0.95	0.95	0.95	12636

Testing				
	precision	recall	f1-score	support
low_0	0.90	1.00	0.95	26
low_1	0.96	0.94	0.95	139
low_2	0.89	0.96	0.92	285
medium_0	0.91	0.90	0.91	478
medium_1	0.87	0.81	0.84	603
medium_2	0.82	0.76	0.79	537
high_0	0.58	0.75	0.65	144
high_1	0.51	0.96	0.67	26
high_2	1.00	0.75	0.86	8
accuracy			0.84	2246
macro avg	0.83	0.87	0.84	2246
weighted avg	0.85	0.84	0.85	2246

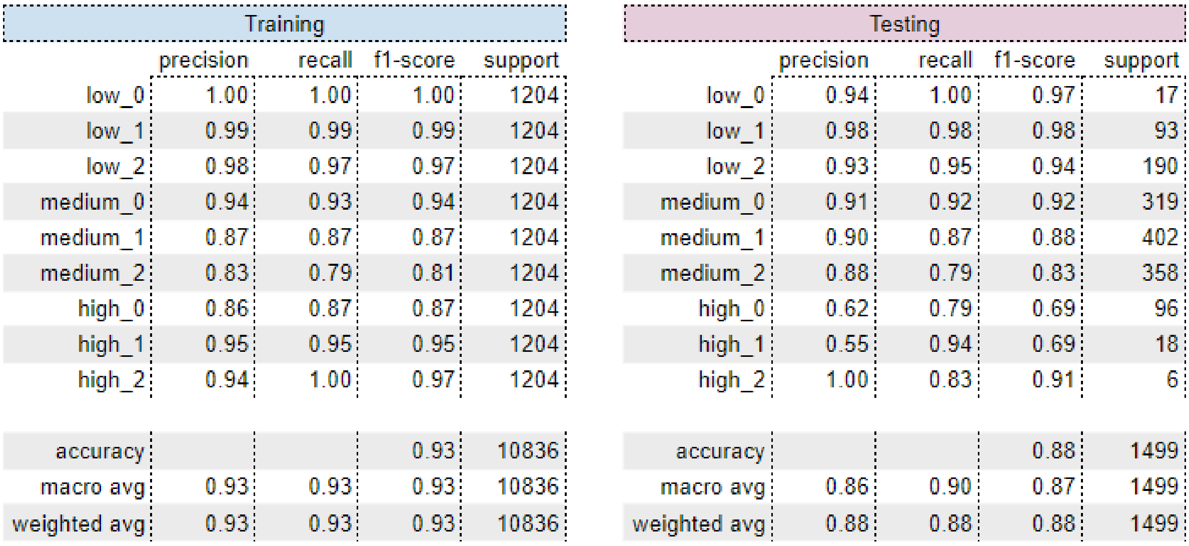
Table 3.2) Confusion matrices of the fitted KNN model, with a K equal to 10.

The same train set split was used for this model as with SVM, the data was split into a training (*70%*) and test set (*30%*), with the training set using oversampling because there is limited data available for certain classes. There appears to be a notable issue of overfitting in the model. Despite achieving a relatively high overall accuracy, the model struggles significantly with identifying high-risk cases (see Table 3.2). This struggle may come from the imbalance within our dataset. Detecting high-risk countries is one of the most important aspects of this business case, making this performance insufficient.

Feedforward neural network

The final model tested is a feedforward neural network (FNN), which shows strong capabilities in handling missing data, effectively operates with nine features and can handle non-linear relationships. The test model was trained with the following (hyper)parameters:

hidden_layers = 1 (ReLU)	*units = 256*	*dropout_rate = 0.2*
patience = 10	*epochs = 2000*	*weight_decay = 0*
epsilon = 1e-7	*beta_1 = 0.9*	*beta_2 = 0.999*
clipnorm = None	*clipvalue = None*	*global_clipnorm = None*
jit_compile = True	*use_ema = False*	*learning_rate = 0.00015*



Training	precision	recall	f1-score	support
low_0	1.00	1.00	1.00	1204
low_1	0.99	0.99	0.99	1204
low_2	0.98	0.97	0.97	1204
medium_0	0.94	0.93	0.94	1204
medium_1	0.87	0.87	0.87	1204
medium_2	0.83	0.79	0.81	1204
high_0	0.86	0.87	0.87	1204
high_1	0.95	0.95	0.95	1204
high_2	0.94	1.00	0.97	1204
accuracy			0.93	10836
macro avg	0.93	0.93	0.93	10836
weighted avg	0.93	0.93	0.93	10836

Testing	precision	recall	f1-score	support
low_0	0.94	1.00	0.97	17
low_1	0.98	0.98	0.98	93
low_2	0.93	0.95	0.94	190
medium_0	0.91	0.92	0.92	319
medium_1	0.90	0.87	0.88	402
medium_2	0.88	0.79	0.83	358
high_0	0.62	0.79	0.69	96
high_1	0.55	0.94	0.69	18
high_2	1.00	0.83	0.91	6
accuracy			0.88	1499
macro avg	0.86	0.90	0.87	1499
weighted avg	0.88	0.88	0.88	1499

Table 3.3) The confusion matrices of the trained FNN test model.

For the FNN model the data was split up into a training (*60%*), validation (*20%*) and test set (*20%*). To address the issue of unbalanced data, oversampling has been employed for the training data, introducing some bias. Notably, unlike the other models, this model does not exhibit overfitting. With an initial accuracy of 88% achieved without any hyperparameter tuning, its performance is impressive (see Table 3.3). This is encouraging, as there is potential to further enhance the model's accuracy by a few percentage points through parameter optimization.

The initial experimentation with the FNN appears to outperform SVM and KNN with our dataset. However, it faces some challenges, particularly in identifying high-risk countries. Finding a solution to this issue will be our next focus as we continue to experiment with the FNN model.

3.3.2 Hyperparameter tuning

The hyperparameters were tuned through random search (RS). Tuning through grid search (GS) would take too long given the amount of hyperparameter combinations possible and the scope of the project. First, an optimizer had to be chosen. Adam, AdamW, Adadelta and SGD were initially tested, where Adam outperformed the other optimizers by a large margin. Using Adam, the following hyperparameters were tuned by means of RS: layers, units (& activation functions), patience, dropout_rate, learning_rate, epsilon, beta_1, beta_2, weight_decay, clipnorm, clipvalue. After numerous testing, the lowest validation loss found was ≈*0.21576*. This was achieved with the following values:

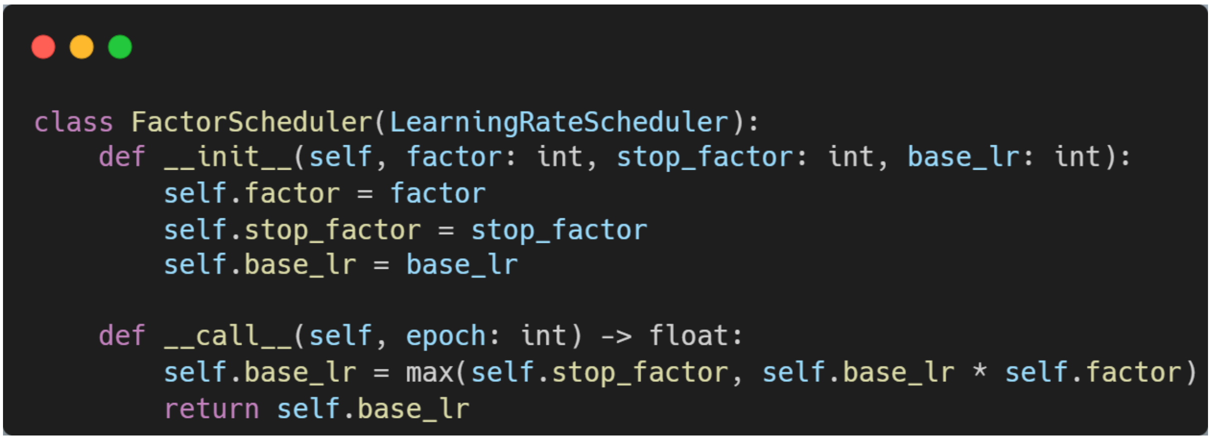
hidden_layers = 1 (ReLU)	*units = 1024*	*dropout_rate = 0*
patience = 25	*epochs = 2000*	*weight_decay = 0*
epsilon = 1e-7	*beta_1 = 0.9*	*beta_2 = 0.999*
clipnorm = None	*clipvalue = None*	*global_clipnorm = None*
jit_compile = True	*use_ema = False*	

learning_rate = FactorScheduler(factor=0.995, stop_factor=0.00075, base_lr=0.002)

Even though the *epochs* parameter is set to *2000*, due to the *EarlyStopping* callback function of Keras (which is used in the model), the model stops its training after ≈*70-110 epochs.*

keras.callbacks.EarlyStopping(monitor="val_loss", patience=25)

The *FactorScheduler* is a learning rate scheduler which will alter the learning rate each epoch by a given factor. The 'optimal' configuration for our model was found to be a *base learning rate* of *0.002*, where each epoch the *base learning rate* will be multiplied by a factor of *0.995*, thus decreasing and eventually stagnating at *0.00075*. The source code of the *FactorScheduler* is shown in Figure 3.4 to give a better idea of the inner workings.



```python
class FactorScheduler(LearningRateScheduler):
    def __init__(self, factor: int, stop_factor: int, base_lr: int):
        self.factor = factor
        self.stop_factor = stop_factor
        self.base_lr = base_lr

    def __call__(self, epoch: int) -> float:
        self.base_lr = max(self.stop_factor, self.base_lr * self.factor)
        return self.base_lr
```

Figure 3.4) The implementation of the FactorScheduler in Python.

Since the *EarlyStopping* callback is used, the model stops training before the *stop factor* is reached. The training started at a *learning rate* of *0.002*, ending at ≈*70-110 epochs* at a *learning rate* of ≈*0.0014-0.0012*.

In both the final model and earlier models, multiple values were tried for the *dropout_rate*. Initially, a *dropout_rate* of *20%* (0.2) always achieved the lowest validation loss. However, this always led to the training accuracy being *≈10%* lower than the testing accuracy. A *dropout_rate* of *0%* resolved this issue and also increased the accuracy and precision by a large margin. The accuracy increased by *≈53-63%*, which also caused the precision to increase significantly. The dropout most likely decreases the accuracy since there is not a lot of data (7475 rows), which is then divided over nine labels. Missing 1495 (20% of 7475) rows data each epoch, will have a significant impact on the training & validation accuracy (and loss). Especially since the extremes of the risk spectrum (i.e. low_0, low_1, high_0, high_1 & high_2) have a lot less data compared to labels closer to the median/mean (i.e. medium_0, medium_1, & medium_2) (see Figure 3.5).

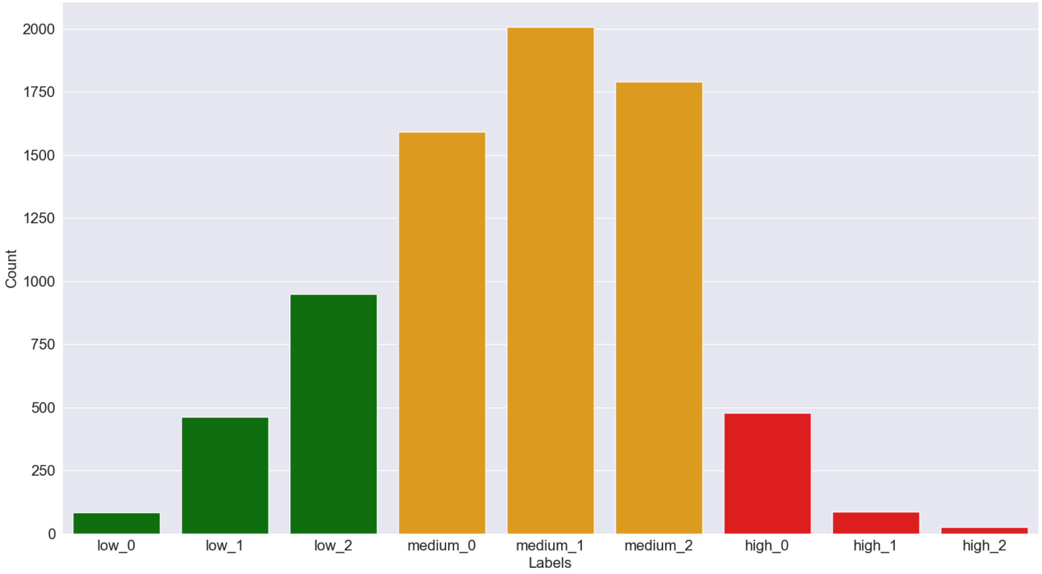


Figure 3.5) A bar graph depicting the amount of rows per label.

The lowest validation loss achieved during tuning was *≈0.2158* with an accuracy of *≈93%*.

4. Results

4.1 The model

As stated in the previous chapter, the lowest validation loss achieved during tuning was *≈0.2158* with an accuracy of *≈93%*. *Chapter 4.1* will go into detail about the training results, the predictions of the test dataset and the feature importance of the best (*lowest validation loss & best accuracy)* model that was trained.

4.1.1 Training

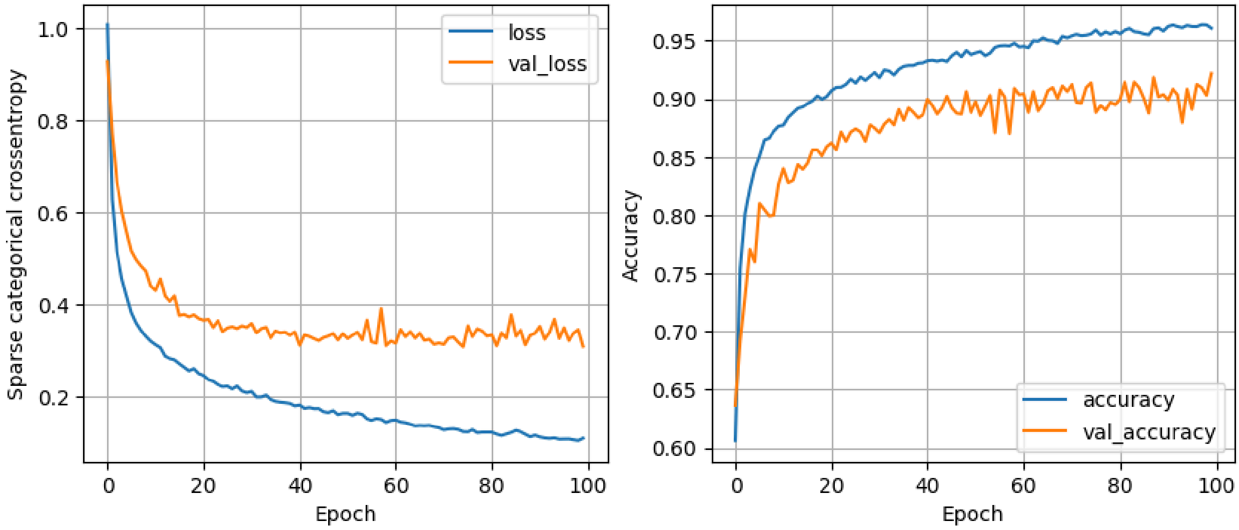


Figure 4.1) The loss and accuracy of the final model during its training.

The graphs in Figure 4.1 show the results of the loss function and accuracy during the training, respectively. From the loss function graph (Figure 4.1 left graph), it can be concluded that our model has a 'good' learning rate (see Figure 4.2 left graph). The loss decreases in such a way that it follows an *exponential decay*. From the accuracy graph (Figure 4.1 right graph), it can be concluded that the *validation accuracy* is slightly overfitting, but follows the *training accuracy* fairly well (see Figure 4.2 right graph). This can be solved by conducting more hyperparameter tuning, it could be that the regularisation should be increased or that a more complex model is needed (Stanford University, z.d.).

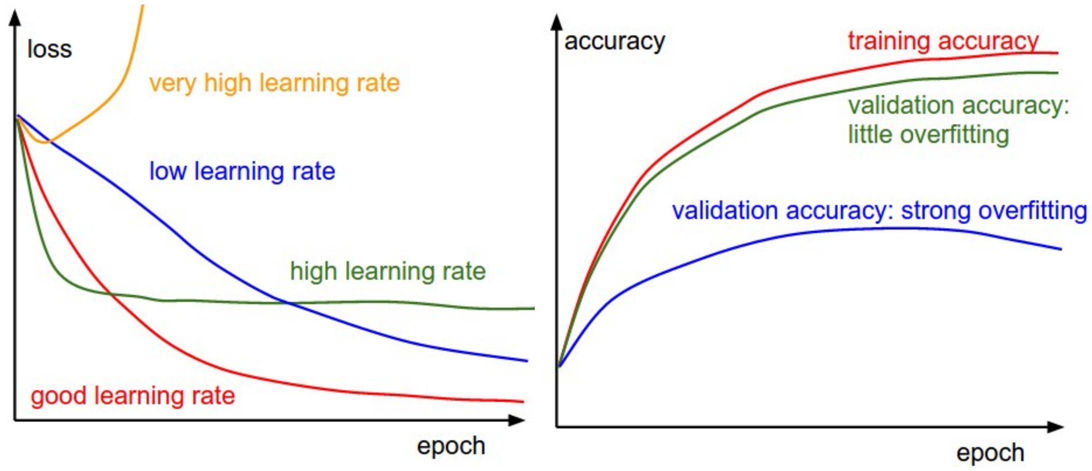
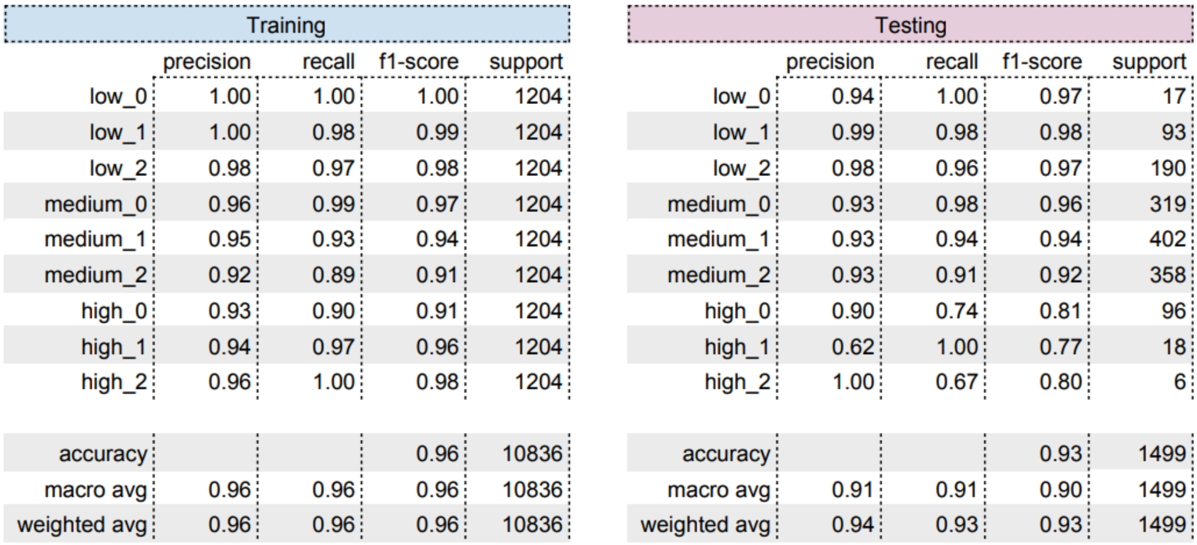


Figure 4.2) Left: ' A cartoon depicting the effects of different learning rates'. Right: 'The gap between the training and validation accuracy indicates the amount of overfitting.' (Stanford University, z.d.).

4.1.2 Predicting

As stated before, if the model is given the test dataset (data which it has not seen before), it predicts ≈*93%* of the labels correct (see Table 4.1). Of the *1499* rows it was given, it predicted *1399* correct and *100* incorrect. The majority of the incorrectly labelled rows are edge cases. Next to this, at a first glance, the model seems to have trouble predicting the '*high'* labels (0..2). This can be explained by the lack of / low amount of *'high'* labelled rows, thus lack of data. The model has less data to train with, thus struggles to generalise the *'high'* rows. Next to this, since there are so few cases, the accuracy metric shown in the table below can be affected greatly when a subset is small. For example, a wrong prediction will affect the accuracy of a set of 6 (*high-2)* a lot, while a wrong prediction on a set of 402 (*medium-1*) will barely affect its accuracy metric. Both of these reasons point to the slight overfitting of the *validation accuracy*.



Training				
	precision	recall	f1-score	support
low_0	1.00	1.00	1.00	1204
low_1	1.00	0.98	0.99	1204
low_2	0.98	0.97	0.98	1204
medium_0	0.96	0.99	0.97	1204
medium_1	0.95	0.93	0.94	1204
medium_2	0.92	0.89	0.91	1204
high_0	0.93	0.90	0.91	1204
high_1	0.94	0.97	0.96	1204
high_2	0.96	1.00	0.98	1204
accuracy			0.96	10836
macro avg	0.96	0.96	0.96	10836
weighted avg	0.96	0.96	0.96	10836

Testing				
	precision	recall	f1-score	support
low_0	0.94	1.00	0.97	17
low_1	0.99	0.98	0.98	93
low_2	0.98	0.96	0.97	190
medium_0	0.93	0.98	0.96	319
medium_1	0.93	0.94	0.94	402
medium_2	0.93	0.91	0.92	358
high_0	0.90	0.74	0.81	96
high_1	0.62	1.00	0.77	18
high_2	1.00	0.67	0.80	6
accuracy			0.93	1499
macro avg	0.91	0.91	0.90	1499
weighted avg	0.94	0.93	0.93	1499

Table 4.1) The confusion matrices of the final model.

Since we calculate the risk score for each row and then categorise it as low, medium or high (0..2), we are able to trace back what issues the model has predicting labels. The risk score is an integer between 0 and 1. This score is then categorised in one of the nine labels, where each label has an equal partition.

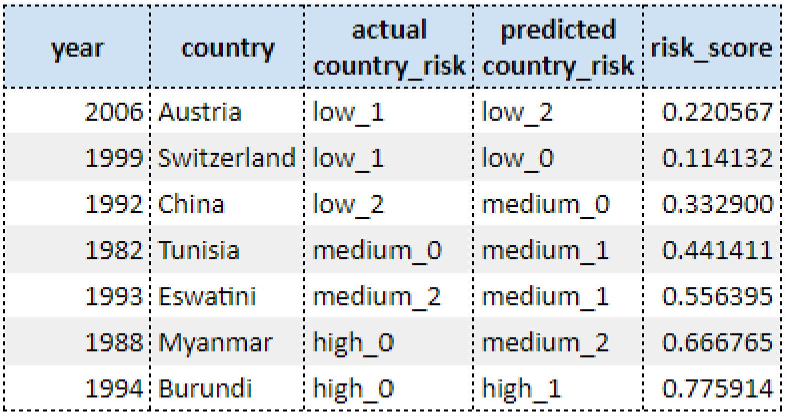
$$\frac{1}{9} \cong 0.111111$$

Both the boundaries and the *risk_score* are rounded at 6 decimals. Here are 2 examples of risk category boundaries:

$$\textit{low\_0} \begin{cases} lowerbound = 0 \times \frac{1}{9} = 0 \\ upperbound = 1 \times \frac{1}{9} \cong 0.111111 \\ lowerbound < y_i <= upperbound \end{cases}$$

$$\textit{medium\_1} \begin{cases} lowerbound = 4 \times \frac{1}{9} \cong 0.444444 \\ upperbound = 5 \times \frac{1}{9} \cong 0.555555 \\ lowerbound < y_i <= upperbound \end{cases}$$

It is important to understand what the boundaries are and how they work when investigating the incorrectly predicted rows. Figure 4.2 shows seven rows that the model has predicted incorrectly. First look at the *actual*- and *predicted country_risk* columns, each row is predicted one label higher (+1) or one label lower (-1) from what the label should be. Next look at the *risk_score* column, each risk score is very close to the boundary of a given risk classification.



year	country	actual country_risk	predicted country_risk	risk_score
2006	Austria	low_1	low_2	0.220567
1999	Switzerland	low_1	low_0	0.114132
1992	China	low_2	medium_0	0.332900
1982	Tunisia	medium_0	medium_1	0.441411
1993	Eswatini	medium_2	medium_1	0.556395
1988	Myanmar	high_0	medium_2	0.666765
1994	Burundi	high_0	high_1	0.775914

Table 4.2) A sample of the set of the incorrectly predicted rows, predicted by the final model.

From all the incorrect rows, there are only a handful of rows that are far away from a given border, with a difference of *0.05*. This, together with the fact that the biggest difference is the actual label and the predicted label is *1 (-1, +1; Table 4.3)*, led us to the conclusion that our model is correct, but struggles with the identification/differentiation of edge cases. Since the decision was made to have nine classes, it is less impactful when the model predicts a row as an adjacent label (*-1, +1*), then for example with a 3 class-multi class classification. Thus, the decision was made to count a *-1* and a *+1* still as a correct prediction, since it has relatively low impact on this scale.

It should also be noted that the boundary closest to a row's *risk_score* is generally the side which the incorrect prediction will go to. For example, the second to last row of Figure 4.2 (Myanmar 1988) has a *risk_score* of *0.666765*. The boundaries of this classification (*high_0*) are *0.666666* and *0.777777*. The *risk_score* of this row is closest to the lower boundary. With the hypothesis stated above, in the case that a row is incorrectly labelled during a prediction, the classification expected would be *medium_2*, which is what the model also predicted.

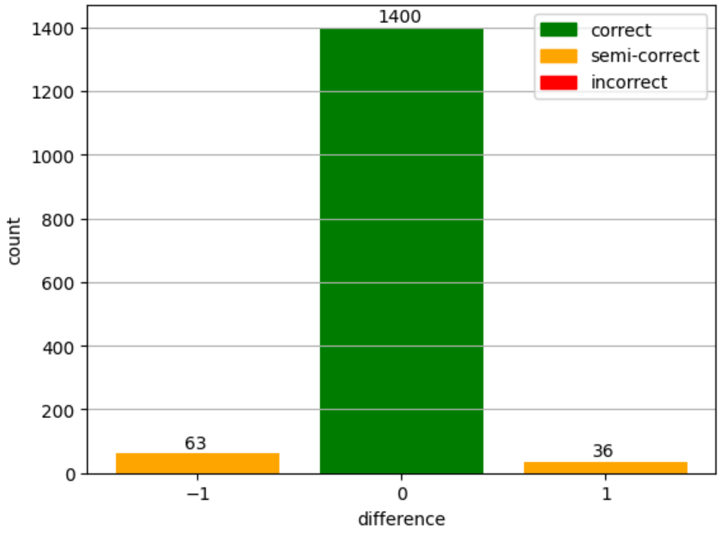
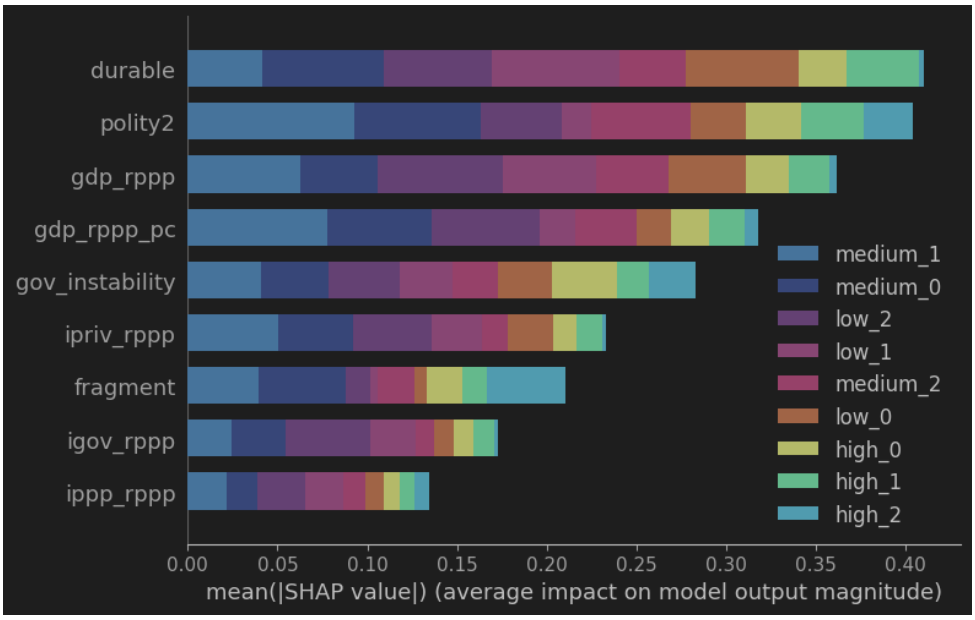


Figure 4.3) The distribution depicting the difference between predicted and observed labels, where the predictions were done by the final model.

The histogram in Figure 4.3 shows that the model tends to classify the rows lower, than they actually are. From the incorrectly predicted rows, the model predicts≈64% of the rows to be in a lower class. This might seem like a lot, but compared to the complete test dataset, the model predicts ≈*4.2%* of the rows to be lower and ≈*2.4%* to be higher than they should be.

4.1.3 Feature importance



*** Figure 4.4) A bar graph, created with the Python library Shap, showing the feature importance, coloured by label.*

Figure 4.4 is created with the Python library Shap. It depicts from top to bottom the most to the least impactful features. The importance of each feature is built up from the importance of said feature per label. The legend is also sorted from the label that is most impacted by features to the least impacted. To get these results, the *shap.,KernelExplainer* was used. This explainer uses the Kernel SHAP method, which is a method that uses a special weighted linear regression to calculate the importance of each feature (SHAP, z.d.).
Due to time constraints and hardware limitations, the feature importance / Shap test was conducted over a small subset. 20 samples were evaluated 100 times to get to the results of Figure 4.4.

shap_sample = shap.sample(X=x_test, nsamples=20)

Where \`*X*\` is a matrix of samples, \`*nsamples*\` is the amount of random samples generated from \`*X*\` and \`*x_test*\` is the test partition of the whole dataset (20%).

shap_values = explainer.shap_values(X=shap_sample, nsamples=100)

Where \`*X*\` is a matrix of samples and \`*nsamples*\` is the number of times to re-evaluate the model when explaining each prediction. If this were to be done on the whole test dataset with a higher evaluation rate, this would have taken more than 72 hours of computation time.

Durable is the most important feature, which is expected, because it counts the amount of years since the last regime change. This is a great indicator of how stable the government of a country has been. For example, most west European countries have a high durable value, while some African countries have a low durable value due to recent turmoil.

Polity2 also proves to be of great importance. Highly Democratic (polity2 score of 8 to 10) and highly autocratic (-8 to -10) countries are generally stable, while anocratic countries (-5 to 5) prove to be less stable. China is a great example of an autocratic country which is doing well.

GDP and GDP per capita are great indicators of the economy of a country. Countries which are poor seem to have a harder time maintaining a stable government, while countries which are rich are more stable. It is interesting that General government investment (igov_rppp), Private investment (ipriv_rppp) and Public-private partnership (ippp_rppp) are not deemed as important.

Government instability counts how many times the country has been unstable in the past. It is derived from durable, so it's interesting that it's not as important. Fragmentation doesn't score very well, because there are a lot of values which are 0 and there are rarely values higher than 0. Since there is so little fluctuation, it has less impact on the risk score and the model deems it as less important.

** 1. Please note that the Shap library did not give any control over the graph.*
** 2. A dark background was chosen for the graph since it increases the readability of the colours. This was achieved with PyCharm from JetBrains.*

4.2 Validation

4.2.1 Accuracy vs Precision

Accuracy and precision are not the same. A model can be accurate but imprecise or inaccurate and precise. Accuracy can be described as the fraction of correctly predicted rows compared to the total amount of rows for a given dataset. Precision (in this context) can be described through an example. Given a multiclass classification problem with three labels and a hyperparameter tuned model. For the 3-label classification, the model has 90% accuracy when predicting the test set. Then, when the model is tested on a 9-label classification, the model has an accuracy of 40%. If the model would be precise, it would have predicted the rows of the 9-label classification with (roughly) the same accuracy as the 3-label classification. But since there is a larger difference between the 3-label and 9-label accuracies, it can be concluded that the model struggles with making a more nuanced prediction and thus is imprecise yet still accurate since the model has a 90% accuracy on the 3-label classification of the problem statement.

To test the accuracy of our model, we compared the results of our model given a 9-label classification (*Chapter 4.1 The model)* and a 27-label classification. In the context of a 27-label classification, our model achieved an accuracy of ≈*70%* with a validation loss of ≈*0.8362*.

4.2.1.1 Training

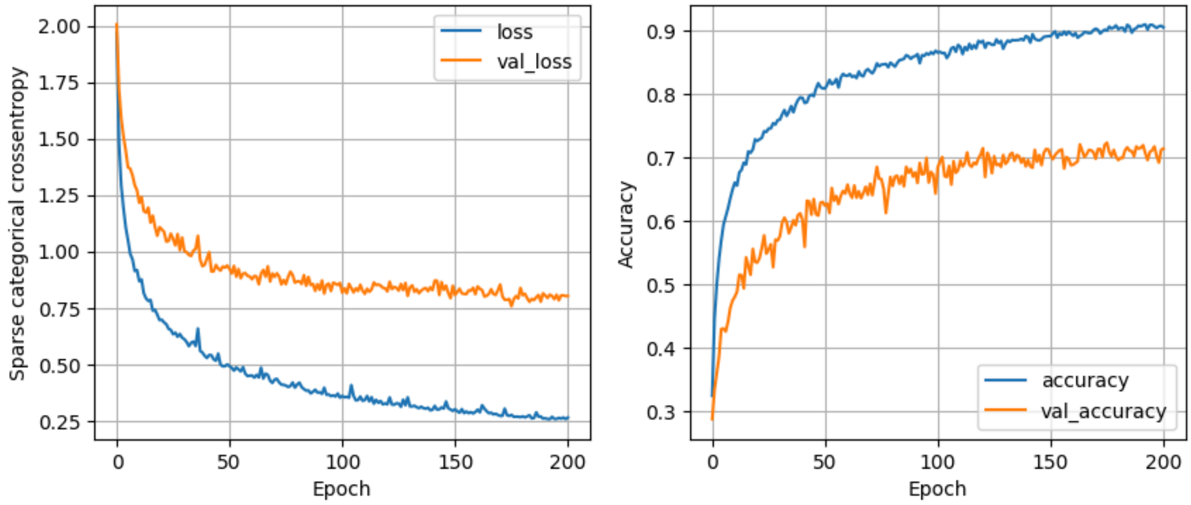


Figure 4.5) The loss and accuracy of the 27-label model during its training.

From the loss function graph from Figure 4.5 (left graph), it can be concluded that the learning rate for the 27-label classification is still a '*good'* learning rate. Like the results of the 9-label classification, the training loss decreases in such a way that it follows an *exponential decay*. This time, the model takes a lot longer (relatively) to train *≈140-200 epochs*, with a *final learning rate* of *≈0.001-0.00075*. Compared to the 9-label classification the model now significantly overfits, this can be seen in Figure 4.5 (right graph) and in chapter *4.2.1.2 Predicting* - Figure 4.2, when comparing the training and validation accuracies. As shown in this graph, there is a significant gap between the training accuracy and validation accuracy, which means that the validation accuracy is significantly overfitting.

4.2.1.2 Predicting

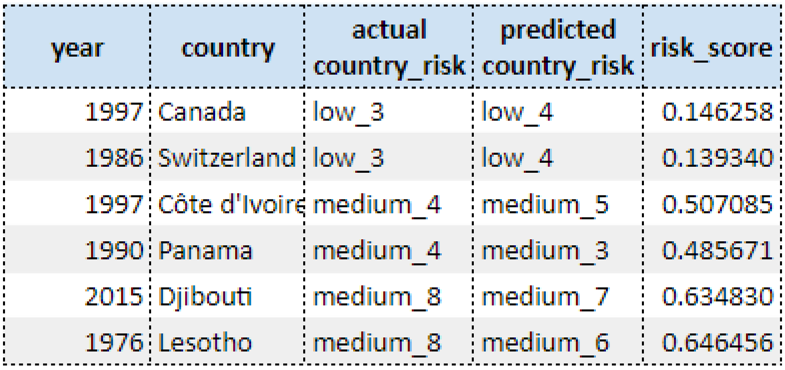
As expected from the training results of the model, the test accuracy (*≈70%*) compared to the training accuracy (*≈91%*) shows that the model is significantly overfitting (see Appendix A). This can be explained by the lack of data, since the limited data is now split up into 27 labels with equal partitions.

Like with the 9-label classification (*Chapter 4.1.2 Predicting*), since we calculate the risk score for each row and then categorise it as low, medium or high (0..8), we are able to trace back what issues the model has predicting labels. The risk score is an integer between 0 and 1. This score is then categorised in one of the 27 labels, where each label has an equal partition.

$$\frac{1}{27} \cong 0.03703703703$$

Here is an example of risk category boundaries:

$$low\_6 \begin{cases} lowerbound = 6 \times \frac{1}{27} \cong 0.222222 \\ upperbound = 7 \times \frac{1}{27} \cong 0.259259 \\ lowerbound < y_i <= upperbound \end{cases}$$



year	country	actual country_risk	predicted country_risk	risk_score
1997	Canada	low_3	low_4	0.146258
1986	Switzerland	low_3	low_4	0.139340
1997	Côte d'Ivoire	medium_4	medium_5	0.507085
1990	Panama	medium_4	medium_3	0.485671
2015	Djibouti	medium_8	medium_7	0.634830
1976	Lesotho	medium_8	medium_6	0.646456

Table 4.3) A sample of the set of the incorrectly predicted rows, predicted by the 27-label model.

Compared to the incorrectly predicted rows of the 9-label model, the rows of the 27-label model are further away from their respective boundaries (see Table 4.3). It does not seem like the incorrectly predicted rows are edge cases. However, just like the 9-label model, the 27-label model, in case of incorrectly predicted rows, generally predicts towards the direction of the boundary closest to the *risk_score*.

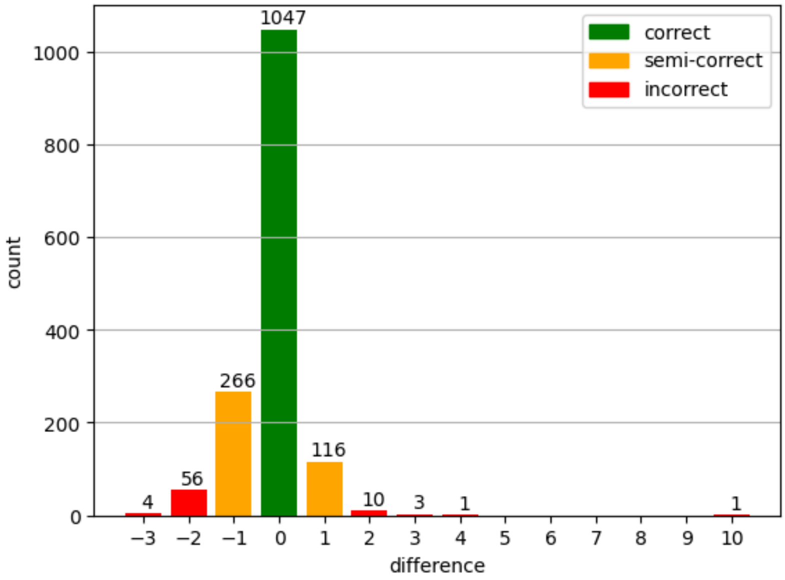


Figure 4.6) The distribution depicting the difference between predicted and observed labels, where the predictions were done by the 27-label model.

Compared to the 9-label model, the 27-model has more incorrectly predicted rows which are more than *-1* and *+1* off the observed value (see Figure 4.6). However, the majority of the predicted rows are within the margin of error (-1, 0 and +1) which was established in chapter *4.1.2 Predicting. 95%* of the predicted rows fall within the established margin of error, of which *≈70%* is truly correct. Of the incorrectly predicted rows, *≈84%* is within the margin of error. Just like the 9-label model, the 27-label model tends to classify rows lower than they are observed to be. Of the incorrectly predicted rows, *≈71%* tends to be lower and *≈29% tends to be higher.* Compared to the complete test dataset, the 27-label model predicted *≈22%* to be lower than observed.

4.2.1.3 Conclusion

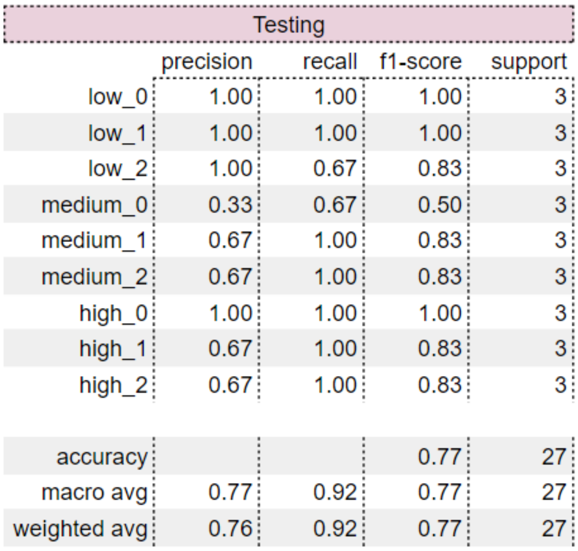
The properties of the 9-label model seem to be enlarged in the 27-label. The amount of overfitting, the error distribution and the tendency of predicted rows lower than observed of the 9-label model are enlarged in the 27-label model. The 27-label model is overfitting, which causes the test accuracy to drop by a lot. This is most likely caused by the lack of data and the hyperparameters not being optimal for 27 labels, thus failing to generalise.

When looking at the true-precision, it can be concluded that the model is somewhat imprecise, since there is a *≈23%* difference between accuracies of the 9- and 27-label models. Compared to earlier versions of the model, this difference is relatively small. Earlier, the accuracy of earlier models dropped to *≈30-40%*, with a difference of *≈53-63%*. Compared to those models, the final model is relatively precise. In contrast to this, when looking at the precision that takes the established error margins into account, it can be concluded that the model is precise. Since, within this context, the 9-label model would be *100%* correct and the 27-label model would be *95%* correct, thus having a difference of *5%*.

Concluding, the model is both accurate and precise, since the model can predict unseen rows with *≈93%* accuracy, which can be further increased to *100%* when taking the error margins into account (accuracy) and has relatively the same accuracy when the 9-label and 27-label versions are compared (precision).

4.2.2 Custom validation dataset

A custom validation dataset consisting of 27 rows was created to independently validate our model (see Table 4.4). This dataset has been manually created. Some prediction errors may occur due to human error as we are not industry experts in manual country risk classification.



	Testing			
	precision	recall	f1-score	support
low_0	1.00	1.00	1.00	3
low_1	1.00	1.00	1.00	3
low_2	1.00	0.67	0.83	3
medium_0	0.33	0.67	0.50	3
medium_1	0.67	1.00	0.83	3
medium_2	0.67	1.00	0.83	3
high_0	1.00	1.00	1.00	3
high_1	0.67	1.00	0.83	3
high_2	0.67	1.00	0.83	3
accuracy			0.77	27
macro avg	0.77	0.92	0.77	27
weighted avg	0.76	0.92	0.77	27

Table 4.4) Confusion matrix depicting the accuracy of the predictions of the final (9-label) model on the custom validation dataset.

The model was able to predict the rows of the custom validation set with ≈*77%* accuracy. All rows which were incorrectly classified fall within the established margin of error (-1 to +1), which is sufficient for our problem (see Table 4.5).

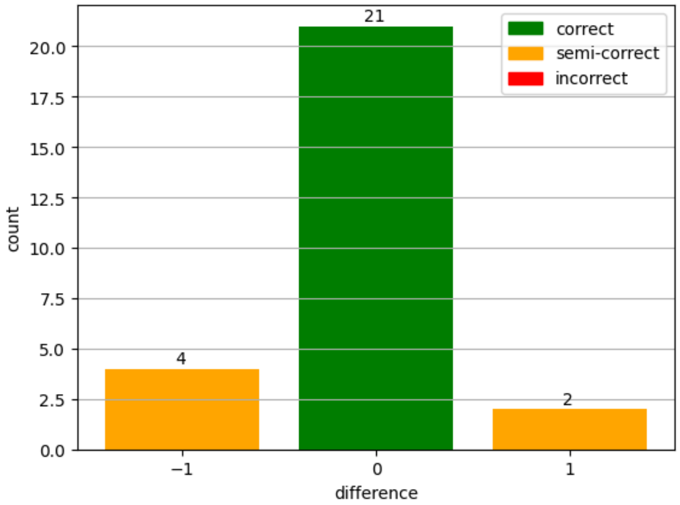


Table 4.5) The distribution depicting the difference between predicted and observed labels, where the predictions were done by the final (9-label) model.

5. Conclusion

The goal for this project was to aid investors/companies in making more strategic decisions in an increasingly complex global market, which can be achieved through the development of the model capable of predicting the risk score of a country. Therefore, the research question is aimed to predict a country's investment risk utilising machine learning, using political and economic data.

After doing background research, it was concluded that this project is a multiclass classification problem. Nine different economical and political features have been used to train a machine learning model to classify the investment risk of a country. The following models seemed suitable to tackle this problem: k-nearest neighbours (KNN), support vector machines (SVM) and feedforward neural networks (FNN). All models were tested using nine classes (low_0..2, medium_0..2 and high_0..2).

After careful experimentation, it was concluded that feedforward neural networks (FNN) work the best on the available data. Our initial FNN model did not overfit, has a decent accuracy and can be further improved using hyperparameter tuning. While the other models did not fall short in terms of accuracy, they did have some problems classifying countries in the high-risk cases.

The final model achieved an accuracy of *≈93%* and validation loss score of *≈0.2158.* When examining the results, it was determined that certain edge cases were classified incorrectly, resulting in an erroneous assignment label (*low_1* or *low_3* instead of *low_1)*. This was solved by counting classification results which are one class off the desired class as correct. This makes the model perform better when predicting edge cases. When taking the established error margin into account, the final model reached *100%* accuracy.

Like every model in the real world, each model has his own limitations. First, the real world is very complex. Not all parameters that define a country's risk score have been included in this model. The number of parameters could be expanded to try to fit the model to the real world. The parameters used in this project are commonly used when trying to create a risk prediction framework. Next to this, data from certain countries was either missing or not publicly available. Minimising missing data is crucial for accurate results.

Allowing a prediction to still be correct when within a margin of error of 1, implies that the scores obtained may not always be completely precise, having a lower true precision. This approach does aid in preventing edge cases from being wrongfully classified.

Finally, the risk label was created using a formula. This creates an enormous bias and probably resulted in the model picking up the patterns from the formula instead of real world trends. It is advised to contact industry experts to classify countries instead, which is out of scope for this project.

There are a few more areas which can be improved. More data can be added to include more features to try to model the real world even better and make the current dataset more complete. This does require a great understanding of what a risk factor should be composed of. A lot more hyperparameter tuning can also be done. Doing this requires powerful hardware, since this process is computationally expensive and takes a substantial amount of time.

We believe that the model can help potential investors and companies make better strategic decisions. The risk score will give them an indicator if it is a good time to invest in a country based on several political and economical factors.

References

Allianz. (z.d.). *Country risk*. Allianz-trade. https://www.allianz-trade.com/en_global/economic-research/country-reports.html

Ayhan, F. (2019). FOREIGN DIRECT INVESTMENTS UNDER IMPACT OF POLITICAL RISKS: THEORETICAL SURVEY. *The EUrASEANs : journal on global socio-economic dynamics*, *1(14)*, 30–40. https://doi.org/10.35678/2539-5645.1(14).2019.30-40

Bluwstein, K., Buckmann, M., Joseph, A., Kang, M., Kapadia, S., & Şimşek, Ö. (2020). Credit Growth, the yield curve and financial crisis prediction: Evidence from a machine learning approach. *Social Science Research Network*. https://doi.org/10.2139/ssrn.3520659

Brownlee, J. (2021, 26 april). *One-vs-Rest and One-vs-One for Multi-Class classification*. MachineLearningMastery.com. https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification/

Farmer, D. (2023, 8 september). *Risk prediction Models: how they work and their benefits*. CIO. https://www.techtarget.com/searchcio/tip/Risk-prediction-models-How-they-work-and-their-benefits

Fernando, J. (2023, 30 november). Gross Domestic Product (GDP): Formula and How to Use It. *Investopedia*. https://www.investopedia.com/terms/g/gdp.asp

Glova, J., Bernatík, W., & Тулай, O. (2020). Determinant effects of political and economic factors on country risk: An evidence from the EU countries. *Montenegrin journal of economics*, *16*(1), 37–53. https://doi.org/10.14254/1800-5845/2020.16-1.3

Government At A Glance 2011. (2011). In *Government at a glance*. https://doi.org/10.1787/gov_glance-2011-en

Meldrum, D. H. (2000). Country risk and foreign direct investment. *Business Economics*, *35*(1), 33. https://www.questia.com/library/journal/1G1-59964458/country-risk-and-foreign-direct-investment

Nath, H. K. (2008). Country Risk Analysis: A survey of the quantitative methods. *Social Science Research Network*. https://doi.org/10.2139/ssrn.1513494

Netherlands Enterprise Agency, RVO. (z.d.). *Public-private partnership (PPP)*. Business.gov.nl. Geraadpleegd op 25 oktober 2023, van https://business.gov.nl/regulation/public-private-partnership/

Neufeld, D. (2023, 9 oktober). Mapped: Investment Risk, by Country. *Visual Capitalist*. https://www.visualcapitalist.com/investment-risk-by-country-map/

Oetzel, J., Bettis, R. A., & Zenner, M. (2001). Country risk measures: How risky are they? *Journal of World Business*, *36*(2), 128–145. https://doi.org/10.1016/s1090-9516(01)00049-9

Padalino, H. (2023, 4 april). What Are Private Investments? *Financial Education Newsletter*. https://www.linkedin.com/pulse/what-private-investments-hayden-padalino#:~:text=Published%20Apr%204%2C%202023,funds%2C%20and%20direct%20company%20investments

SHAP. (z.d.). *SHAP Documentation*. https://shap-lrjball.readthedocs.io/en/latest/generated/shap.KernelExplainer.html#shap.KernelExplainer.shap_values

Stanford University. (z.d.). *CS231n Convolutional Neural Networks for Visual Recognition*. Github. https://cs231n.github.io/neural-networks-3/

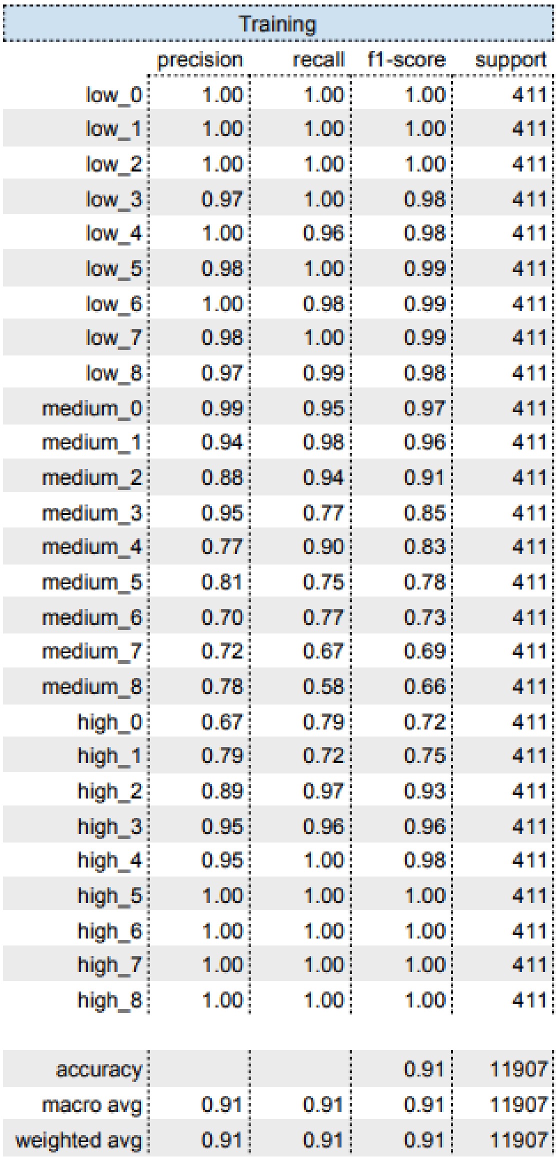
Watson, R. T., & Webster, J. (2002, juni). *Analyzing the Past to Prepare for the Future: Writing a Literature Review*. MIS Quarterly. https://www.jstor.org/stable/4132319

Wolfswinkel, J. F., Furtmueller, E., & Wilderom, C. P. (2013). Using grounded theory as a method for rigorously reviewing literature. *European Journal of Information Systems*, *22*(1), 45–55. https://doi.org/10.1057/ejis.2011.51

Xu, J., Zhang, R., Wang, Y., Yan, H., Liu, Q., Guo, Y., & Ren, Y. (2022). Assessing China's investment risk of the Maritime Silk Road: a model based on multiple machine learning methods. *Energies*, *15*(16), 5780. https://doi.org/10.3390/en15165780

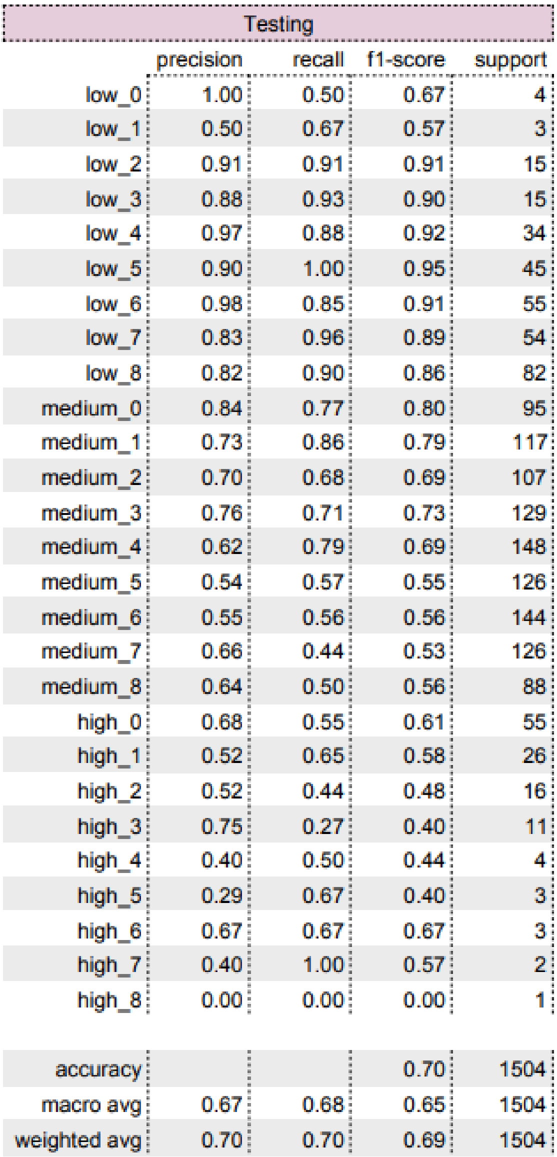
Appendix A. 27-label classification confusion matrix

Table A1. Training confusion matrix



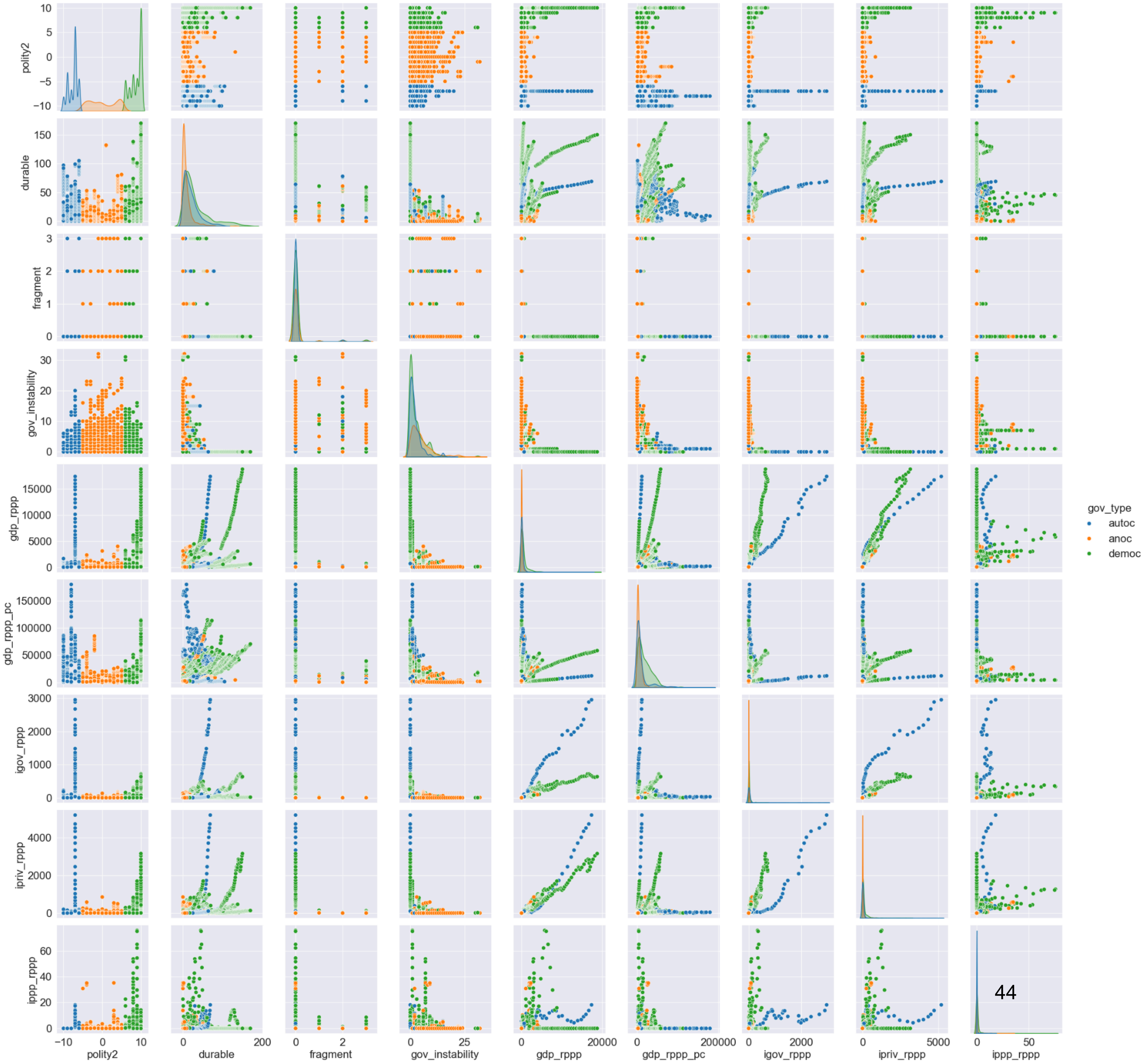
	Training			
	precision	recall	f1-score	support
low_0	1.00	1.00	1.00	411
low_1	1.00	1.00	1.00	411
low_2	1.00	1.00	1.00	411
low_3	0.97	1.00	0.98	411
low_4	1.00	0.96	0.98	411
low_5	0.98	1.00	0.99	411
low_6	1.00	0.98	0.99	411
low_7	0.98	1.00	0.99	411
low_8	0.97	0.99	0.98	411
medium_0	0.99	0.95	0.97	411
medium_1	0.94	0.98	0.96	411
medium_2	0.88	0.94	0.91	411
medium_3	0.95	0.77	0.85	411
medium_4	0.77	0.90	0.83	411
medium_5	0.81	0.75	0.78	411
medium_6	0.70	0.77	0.73	411
medium_7	0.72	0.67	0.69	411
medium_8	0.78	0.58	0.66	411
high_0	0.67	0.79	0.72	411
high_1	0.79	0.72	0.75	411
high_2	0.89	0.97	0.93	411
high_3	0.95	0.96	0.96	411
high_4	0.95	1.00	0.98	411
high_5	1.00	1.00	1.00	411
high_6	1.00	1.00	1.00	411
high_7	1.00	1.00	1.00	411
high_8	1.00	1.00	1.00	411
accuracy			0.91	11907
macro avg	0.91	0.91	0.91	11907
weighted avg	0.91	0.91	0.91	11907

Table A2. Testing confusion matrix

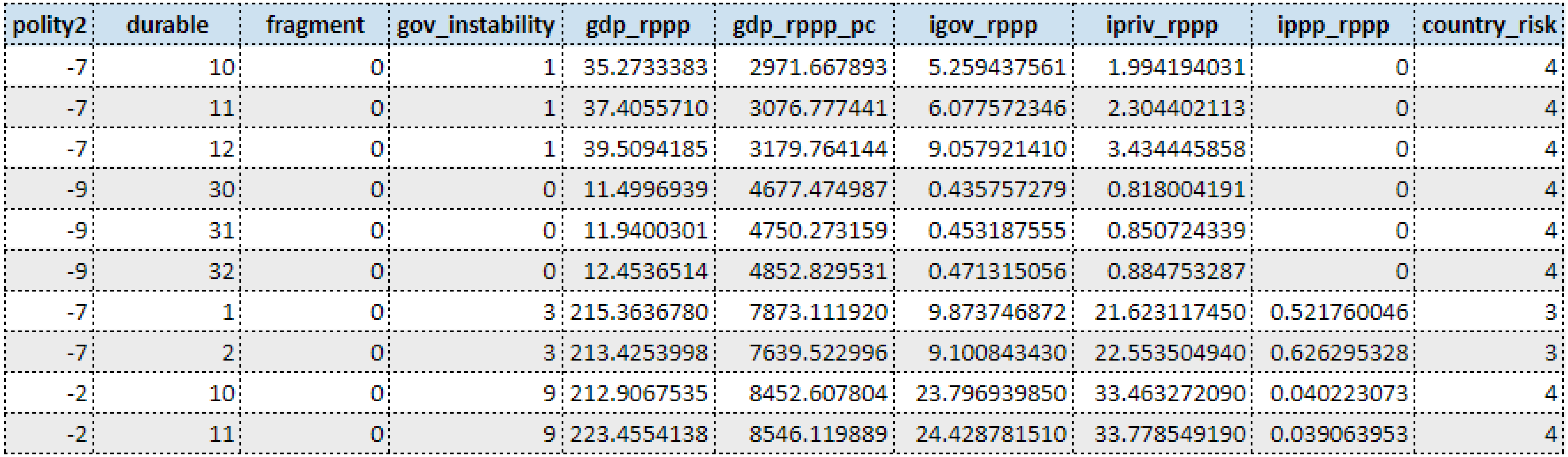


	Testing			
	precision	recall	f1-score	support
low_0	1.00	0.50	0.67	4
low_1	0.50	0.67	0.57	3
low_2	0.91	0.91	0.91	15
low_3	0.88	0.93	0.90	15
low_4	0.97	0.88	0.92	34
low_5	0.90	1.00	0.95	45
low_6	0.98	0.85	0.91	55
low_7	0.83	0.96	0.89	54
low_8	0.82	0.90	0.86	82
medium_0	0.84	0.77	0.80	95
medium_1	0.73	0.86	0.79	117
medium_2	0.70	0.68	0.69	107
medium_3	0.76	0.71	0.73	129
medium_4	0.62	0.79	0.69	148
medium_5	0.54	0.57	0.55	126
medium_6	0.55	0.56	0.56	144
medium_7	0.66	0.44	0.53	126
medium_8	0.64	0.50	0.56	88
high_0	0.68	0.55	0.61	55
high_1	0.52	0.65	0.58	26
high_2	0.52	0.44	0.48	16
high_3	0.75	0.27	0.40	11
high_4	0.40	0.50	0.44	4
high_5	0.29	0.67	0.40	3
high_6	0.67	0.67	0.67	3
high_7	0.40	1.00	0.57	2
high_8	0.00	0.00	0.00	1
accuracy			0.70	1504
macro avg	0.67	0.68	0.65	1504
weighted avg	0.70	0.70	0.69	1504

Appendix B. Pairs plot



Appendix C. Sample of the machine learning dataset



polity2	durable	fragment	gov_instability	gdp_rppp	gdp_rppp_pc	igov_rppp	ipriv_rppp	ippp_rppp	country_risk
-7	10	0	1	35.2733383	2971.667893	5.259437561	1.994194031	0	4
-7	11	0	1	37.4055710	3076.777441	6.077572346	2.304402113	0	4
-7	12	0	1	39.5094185	3179.764144	9.057921410	3.434445858	0	4
-9	30	0	0	11.4996939	4677.474987	0.435757279	0.818004191	0	4
-9	31	0	0	11.9400301	4750.273159	0.453187555	0.850724339	0	4
-9	32	0	0	12.4536514	4852.829531	0.471315056	0.884753287	0	4
-7	1	0	3	215.3636780	7873.111920	9.873746872	21.623117450	0.521760046	3
-7	2	0	3	213.4253998	7639.522996	9.100843430	22.553504940	0.626295328	3
-2	10	0	9	212.9067535	8452.607804	23.796939850	33.463272090	0.040223073	4
-2	11	0	9	223.4554138	8546.119889	24.428781510	33.778549190	0.039063953	4