Statistical Research: Water Resource Analysis with R

01.05.2025

Seunghyun Kim
Niran Raj Pradhan
Heejung Heo

Table of Contents

Research Objectives

Data Overview & Preprocessing Steps

Supervised Learning: Predicting Population Growth Categories

Unsupervised Learning: PCA and Clustering Analysis

1. **Introduction**
2. **R Libraries Used**
3. **Clustering Methodology**
   3.1 Feature Selection & Normalization
   3.2 Model Selection
   3.3 Principal Component Analysis
   3.4 Grid Search
4. **Results and Interpretation**
   4.1 Clustering Configuration Selection
   4.2 Evaluation and Visualization
   4.3 Cluster Profiles
5. **Discussion on Policy Implications**
6. **Conclusion**
7. **Conclusion**

Time Series Forecasting: Freshwater Withdrawals Prediction

1. **Introduction**
2. **R Libraries Used**
3. **Variable and Country Selection Rationale**
   3.1 Target Variable: Freshwater Withdrawals
   3.2 Target Country: Germany
4. **Forecasting Framework**
   4.1 Arima Forecasting
   4.2 Prophet Forecasting
5. **Forecasting Results and Evaluation**
   5.1 Arima Forecasting
   5.2 Prophet Forecasting
   5.3 Model Performance Evaluation and Comparison
6. **Interpretation of Forecast Results and Conditional Policy Recommendations**
   6.1 Potential Drivers Behind the Forecasted Withdrawal Trend
   6.2. Scenario-Based Policy Recommendations
7. **Conclusion**

Data Sources, References, and R Scripts

- Data Sources
- References
- R Scripts

Appendix

- Model Explanation

Research Objectives

Water is a fundamental resource essential for human survival, economic development, and environmental stability. Yet growing water stress concerns—driven by population dynamics, climate change, and economic activity—pose a major global challenge. A deep understanding of water-use dynamics, and their links to economic and climate factors, is therefore critical for informed policymaking and sustainable management.

This study analyzes global and country-level water-use patterns by combining classical statistical techniques with machine learning workflows in R. Water-related variables were examined through a structured pipeline comprising exploratory data analysis, supervised learning(binary classification), unsupervised learning(PCA & clustering), and time-series forecasting. A major emphasis was placed on rigorous validation and evaluation, such as constructing a reliable analytical workflow, optimizing model performance, critically reviewing assumptions and limitations, and examining generalizability of the models—to ensure that all findings are both reliable and applicable.

Data Overview & Preprocessing Steps

This section provides an overview of the data used in this research and the preprocessing steps applied prior to its utilization, including filtering, regression imputation, labeling, and feature derivation. Normalization has not been performed at this stage and will be applied according to each subsequent task's requirements. Although these procedures are time-consuming and technically demanding, they are crucial for ensuring the accuracy and reliability of all subsequent analyses and thus should be performed with the utmost care.

1. Data Description

The initial dataset was obtained from the World Bank, an international financial institution and open-data provider, and includes the following features:

Name	Description
Water productivity	GDP in constant prices divided by annual total water withdrawal
Total freshwater withdrawals	Annual total freshwater withdrawals in billion cubic meters
Sectoral withdrawals (agricultural/domestic/industrial)	Percentage of freshwater withdrawals used for each sector (agricultural/domestic/industrial)
Water stress	The ratio between total freshwater withdrawals and total renewable freshwater resources
Precipitation	Average precipitation in depth (mm per year)
Private investment	Investment in water and sanitation with private participation(current US$)
Natural disaster	Percentage of the population affected by droughts, floods, and extreme temperatures (Average from 1990 to 2009)
GDP per capita(PPP)	Gross Domestic Product per capita, in purchasing power parity (current international $)
Population	Total population by each country
Income level	Classification of each country according to the World Bank's income groups (e.g., Low income, Lower middle income, Upper

	middle income, High income).
Region	Geographical grouping of each country as defined by the World Bank (e.g., East Asia & Pacific, Latin America & Caribbean, Sub-Saharan Africa, etc.).

Note: “Natural Disaster” contains only the average value of the annual data from 1990 to 2009.

2. Data Preprocessing

2.1. R Libraries used

Library	Purpose
tidyverse	Data manipulation and visualization
rnaturalearth & rnaturalearthdata	Retrieves and manages natural Earth geographic data
sf	Spatial data handling for map visualization

2.2. Missing Data Assessment and Handling

Even though the dataset originally spanned 1960–2023 for 266 countries, a substantial share of values turned out to be missing. In order to assess and address these gaps, missing data proportions by feature were first evaluated and then visualized using a horizontal bar chart.

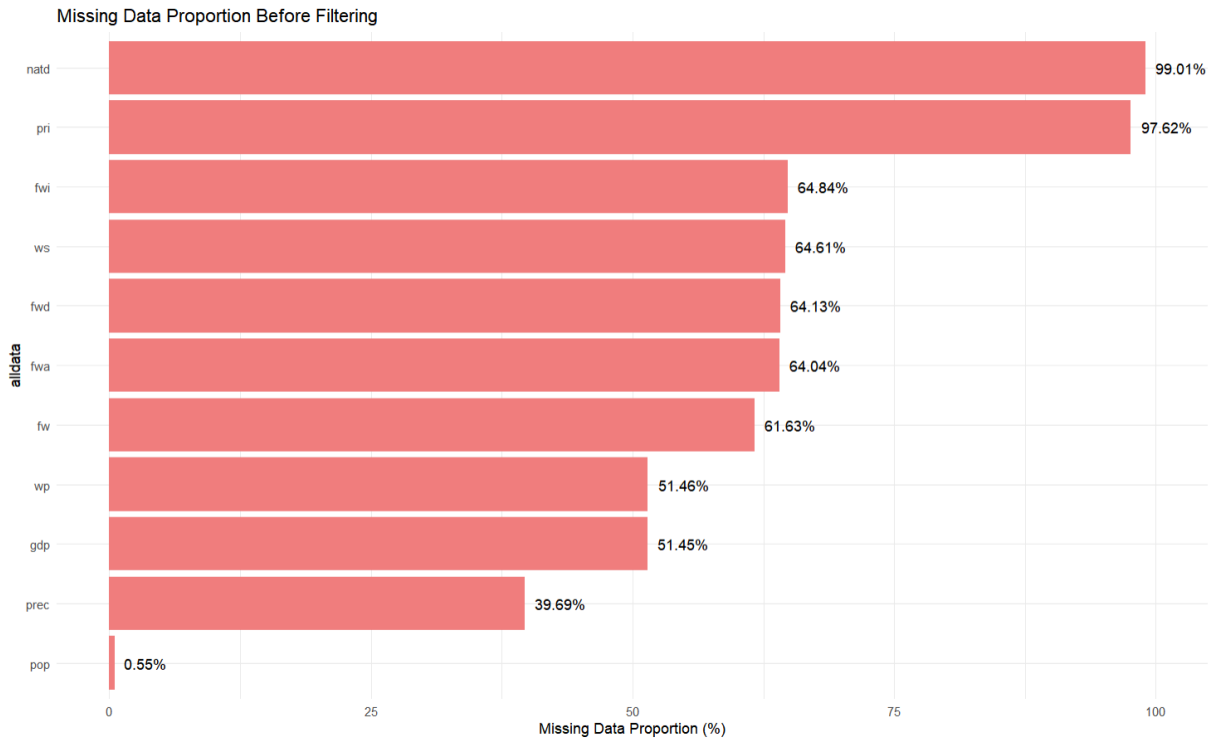


Figure - Missing Data Proportion by Feature

According to the generated plot, natural disaster and private investment exhibited extremely high missing rates (99.01% and 97.62% respectively), while population had an exceptionally low rate (0.55%). The remaining features displayed a missingness range of 39.69%–64.84%. Features were subsequently classified into two subgroups based on their missing value proportions: “normal” and “problematic.” The “problematic” subgroup consisted solely of natural disaster and private investment, while all other features fell into the “normal” category. This grouping was designed to facilitate the use of distinct missing data handling approaches by subgroup.

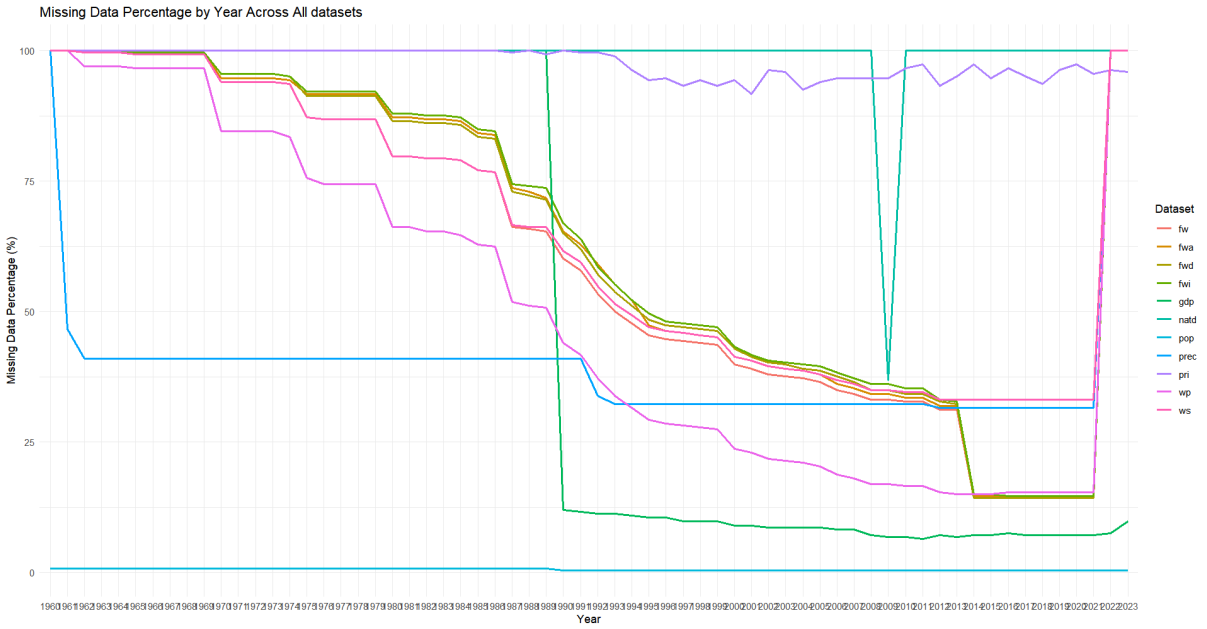


Figure - Missing Data Percentage by Year Across All Features

This line chart tracks missing value proportions for each feature from 1960 to 2023. Missingness is extremely high for all features in the very early decades but steadily declines. This downward trend likely reflects the gradual establishment of standardized reporting protocols, expansion of national statistical agencies, or improvements in data collection technologies. Two anomalies stand out in the chart: the 2009 spike in natural disaster data due to the aggregation of 20 years of data and a modest rebound in missingness for 2022 and 2023 in most features, likely due to reporting delays for the most recent time.

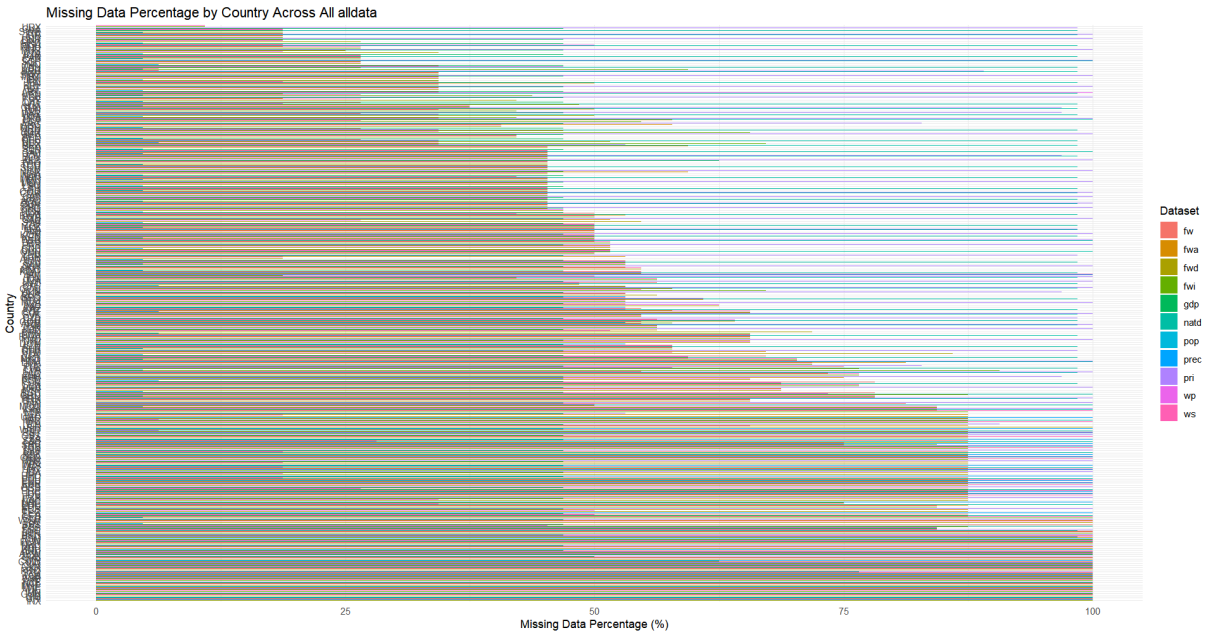


Figure - Missing Data Percentage by Country Across All Datasets

This horizontal bar chart displays the share of missing values for each country. It is observed that missingness varies considerably by different countries as well, possibly due to a lack of statistical infrastructure, political or social instability, or reporting lags.

```
> year_threshold
[1] 0.55
> country_threshold
[1] 0.57
```

R Console Output - Thresholds for the Normal Dataset

Since missingness was concentrated in earlier years and in certain countries, a two-stage filtering strategy was applied to the “normal” datasets (all features except natural disaster and private investment) to address missing values. First, years with more than 55% missing values were dropped by intersecting, across all nine features, the sets of years whose missing value proportions did not exceed the 0.55 threshold. Next, countries exceeding 57% missingness were removed by the same logic. Finally, each dataset was reduced to retain only those valid years and countries, yielding filtered “normal” data frames with substantially improved completeness. This approach also ensures that features share a common set of years and countries, facilitating consistent, panel-wide analyses and seamless data frame manipulation.

	Country.Name	Country.Code	Indicator.Name	Indicator.Code	has_data
1	Angola	AGO	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
2	Albania	ALB	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	1
3	United Arab Emirates	ARE	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
4	Antigua and Barbuda	ATG	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
5	Austria	AUT	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
6	Azerbaijan	AZE	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
7	Burundi	BDI	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
8	Belgium	BEL	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
9	Benin	BEN	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
10	Burkina Faso	BFA	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
11	Bulgaria	BGR	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	1
12	Bahrain	BHR	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	0
13	Belize	BLZ	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	1
14	Bolivia	BOL	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	1
15	Botswana	BWA	Investment in water and sanitation with private participation...	IE.PPI.WATR.CD	1

Figure - Country Level Labelling in Private Investment Data

For the “problematic” dataset, a different strategy was adopted. Since the private investment data exhibited a 97.62% of missingness, it was converted into a binary label at the country level. Any country with at least one valid observation received a label of 1, while countries with no data at all were labeled 0. This approach is intended to avoid numeric imputation, which would be unreliable given the extreme sparsity and lack of domain knowledge.

```
> table(pri_labelled$has_data)
```

```
  0   1
205  61

> table(pri_labelled_final$has_data)

 0  1
83 32
```

R Console Output - Class Distribution of Private Investment Labels Before and After Filtering

Initially, the number of countries with no records at all was significantly higher than the number of countries with at least one record. Therefore, the labeled panel was filtered to include only those countries retained by the normal dataset, in order not only to ensure consistency across datasets, but also to further improve class balance. This approach is based on the assumption that countries removed earlier for excessive missingness in normal features are also likely to not have private investment records. After applying this filter, the class size ratio narrowed from 3.4:1 (205 zeros vs. 61 ones) to 2.6:1 (83 zeros vs. 32 ones), making the potential subsequent classification task more manageable. However, class imbalance is still significant—positives account for only about 28% of cases—so applying resampling techniques (e.g. SMOTE) or incorporating class weights in the model would be advisable.

	Country.Name	Country.Code	Indicator.Name	Indicator.Code	X2000
1	Angola	AGO	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	1.0117646764
2	Albania	ALB	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	5.2695769934
3	United Arab Emirates	ARE	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	*NA*
4	Antigua and Barbuda	ATG	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	*NA*
5	Austria	AUT	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	0.0381529783
6	Azerbaijan	AZE	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	1.1055650667
7	Burundi	BDI	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	2.3774112889
8	Belgium	BEL	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	0.0016921681
9	Benin	BEN	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	0.8582747993
10	Burkina Faso	BFA	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	1.2500368274
11	Bulgaria	BGR	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	0.0085531604
12	Bahrain	BHR	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	*NA*
13	Belize	BLZ	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	0.8060855888
14	Bolivia	BOL	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	1.2974291636
15	Botswana	BWA	Droughts, floods, extreme temperatures (% of population, a...	EN.CLC.MDAT.ZS	0.7404187840

Figure - Overview of the Natural Disaster Data

Natural Disaster data—originally recorded as a 1990–2009 average in the 2009 column—was processed by extracting the X2009 column and renaming it to X2000, which is the midpoint year of 1990–2009. The resulting data frame was then filtered to the same set of valid countries retained in the normal dataset, in order to ensure consistency across all panels.

```
> sum(is.na(natural_Disaster_filtered$X2000)) /
nrow(natural_Disaster_filtered) * 100
```

```
[1] 36.84211
> sum(is.na(natural_Disaster_filtered_final$X2000)) /
nrow(natural_Disaster_filtered_final) * 100
[1] 8.695652
```

R Console Output - Missingness of Natural Disaster Data Before and After Country Filtering

Filtering also reduced missingness in the natural disaster data from 36.84% to 8.70%, indicating that missingness in this feature is not independent of missingness in other features.

```r
# Regression Imputation for natural disasters
natural_disasters <- natural_disasters %>%
  filter(Country.Code %in% water_stress$Country.Code)

nd <- natural_disasters %>%
  left_join(info, by = "Country.Code") %>%
  mutate(Region = as.factor(Region))

nd_model <- lm(X2000 ~ Region, data = nd, na.action = na.exclude)

nd$X2000_imputed <- ifelse(
  is.na(nd$X2000),
  predict(nd_model, newdata = nd),
  nd$X2000
)
```

Code Snippet - Regression Imputation for Natural Disaster Data

Finally, remaining gaps in the natural disaster data were filled by regressing the midpoint year value (X2000) on each country's region. By definition, the natural disaster data represents the percentage of the population affected by droughts, floods, and extreme temperatures. Therefore, an assumption has been made that countries within the same region experience broadly similar exposure to such extreme weather events, and conditional means can serve as plausible proxies for missing national values.
Given the relatively low post-filter missing rate of 8.7%, leveraging regional conditional means was not expected to introduce substantial error. However, more rigorous validation—such as comparing the imputed values against independent disaster-impact datasets or conducting cross validation within the panel—would strengthen confidence in the assumption of regional homogeneity.

```
> print(missing_data)
     Dataset Missing_Proportion
wp        wp                  0
ws        ws                  0
fw        fw                  0
fwa      fwa                  0
fwd      fwd                  0
fwi      fwi                  0
gdp      gdp                  0
pop      pop                  0
prec    prec                  0
pri       pri                  0
natd     natd                  0
```

R output console - Final Missing Data Proportion

As a result of the applied filtering, labeling, and imputation steps, all datasets no longer contained missing values.

2.3 Data Overview After Handling Missing Data

filtered_all	list [11]	List of length 11
wp	list [115 x 32] (S3: data.frame)	A data.frame with 115 rows and 32 columns
ws	list [115 x 32] (S3: data.frame)	A data.frame with 115 rows and 32 columns
fw	list [115 x 32] (S3: data.frame)	A data.frame with 115 rows and 32 columns
fwa	list [115 x 32] (S3: data.frame)	A data.frame with 115 rows and 32 columns
fwd	list [115 x 32] (S3: data.frame)	A data.frame with 115 rows and 32 columns
fwi	list [115 x 32] (S3: data.frame)	A data.frame with 115 rows and 32 columns
gdp	list [115 x 32] (S3: data.frame)	A data.frame with 115 rows and 32 columns
pop	list [115 x 32] (S3: data.frame)	A data.frame with 115 rows and 32 columns
prec	list [115 x 32] (S3: data.frame)	A data.frame with 115 rows and 32 columns
pri	list [115 x 5] (S3: data.frame)	A data.frame with 115 rows and 5 columns
natd	list [115 x 5] (S3: data.frame)	A data.frame with 115 rows and 5 columns

Figure - Dataset Overview After handling Missing Data

All nine “normal” features now have identical dimensions—115 countries by 28 years (plus 4 meta columns)—facilitating manipulation and analysis across multiple indicators. The natural disaster data is reduced to a single column (X2000), representing the 1990–2009 average, and the private investment data likewise consists of a single binary label column, reflecting its recoding from a sparse time series into a presence/absence indicator.

```
> valid_years
 [1] "X1994" "X1995" "X1996" "X1997" "X1998" "X1999" "X2000" "X2001"
"X2002" "X2003" "X2004" "X2005" "X2006"
[14] "X2007" "X2008" "X2009" "X2010" "X2011" "X2012" "X2013" "X2014"
"X2015" "X2016" "X2017" "X2018" "X2019"
[27] "X2020" "X2021"
```

R Console Output - List of retained Years after handling Missing Data

This is the list of retained years across all features after handling missing data. These years form a 28-year span from 1994 through 2021 with no discontinuity, which allows focused analysis on recent decades without any temporal gaps.

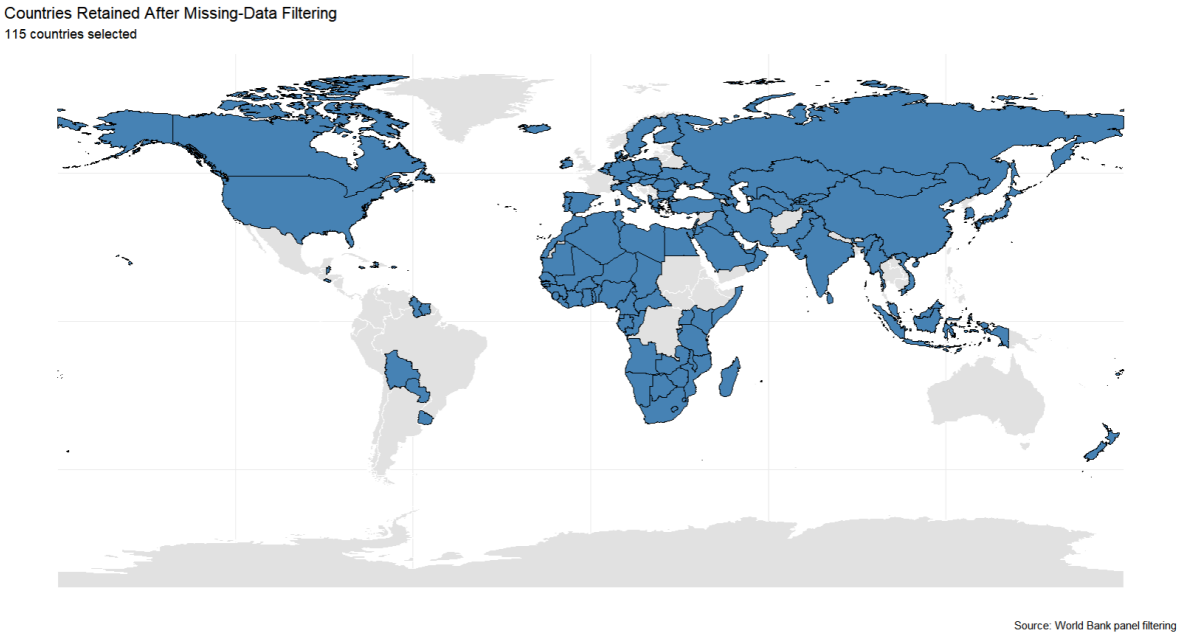


Figure - Global Coverage After handling Missing Data

This world map shows the countries retained across all indicators after handling missing data. While some exclusions occur in Asia, Europe, Africa, and Oceania, the greatest concentration of drop-outs is in Central and Latin America. This pattern may reflect limited statistical capacity, resource constraints, or challenges in data collection and dissemination of that region. Therefore, analyses focusing on Central or Latin America should be interpreted with caution or supplemented by additional sources, as they rely on a very narrow subset of countries and may skew conclusions.

```
> print(size_summary)
     Dataset Cells_Before Cells_After Retained_Pct
fw        fw        17024        3220        18.91
fwa      fwa        17024        3220        18.91
fwd      fwd        17024        3220        18.91
fwi      fwi        17024        3220        18.91
gdp      gdp        17024        3220        18.91
natd    natd        17024         115         0.68
pop      pop        17024        3220        18.91
prec    prec        17024        3220        18.91
pri      pri        17024         115         0.68
wp        wp        17024        3220        18.91
ws        ws        17024        3220        18.91
```

R Console Output - Cell-Count Retention After Filtering

```
> print(non_na_summary)
     Dataset NonNA_Before NonNA_After Retained_Pct
fw        fw         6532        3220        49.30
```

fwa	fwa	6121	3220	52.61
fwd	fwd	6106	3220	52.74
fwi	fwi	5985	3220	53.80
gdp	gdp	8266	3220	38.95
natd	natd	168	115	68.45
pop	pop	16930	3220	19.02
prec	prec	10268	3220	31.36
pri	pri	405	115	28.40
wp	wp	8263	3220	38.97
ws	ws	6024	3220	53.45

R Console Output - Non-NA Cell Retention After Filtering

These two summaries illustrate different aspects of information loss from the filtering procedure. The first R console output shows the loss in total cell slots by the formula

$$retention\ percentage\ = \frac{countries \times years\ (\ after\ filtering)}{countries \times years\ (\ before\ filtering)}$$

, making natural disaster and private investment appear heavily diminished because they were collapsed to a single column. However, most of these removed slots were empty from the beginning, and therefore a second measure of information loss was deemed necessary. The second R console output quantifies loss of actual observations by the formula

$$retention\ percentage\ = \frac{non{-}NA\ entries\ after\ filtering}{non{-}NA\ entries\ before\ filtering}$$

. It shows that “normal” features typically retain 30–55% of their initial non-NA values, while natural disaster and private investment data preserve 68.5% and 28.4%, respectively. The population data is an exception: since its original missing rate was exceptionally low (under 1%), there is little difference between structural loss (cell-count reduction) and observational loss (non-NA reduction) for that feature.

Balancing the trade-off between dataset completeness and information loss guided the threshold choices. Looser thresholds allow more cells to remain but at the cost of higher overall missingness; stricter thresholds reduce missingness but eliminate more data. Intermediate cutoffs were chosen to maintain missing-data rates within reasonable bounds while preserving as many cells as possible. However, these thresholds were based on subjective judgment and can be revisited if further analyses call for a different balance. For example, if a classification task exhibits high variance, suggesting the dataset is too small, the filters can be loosened to include more cells even if that increases the missing-data rate.

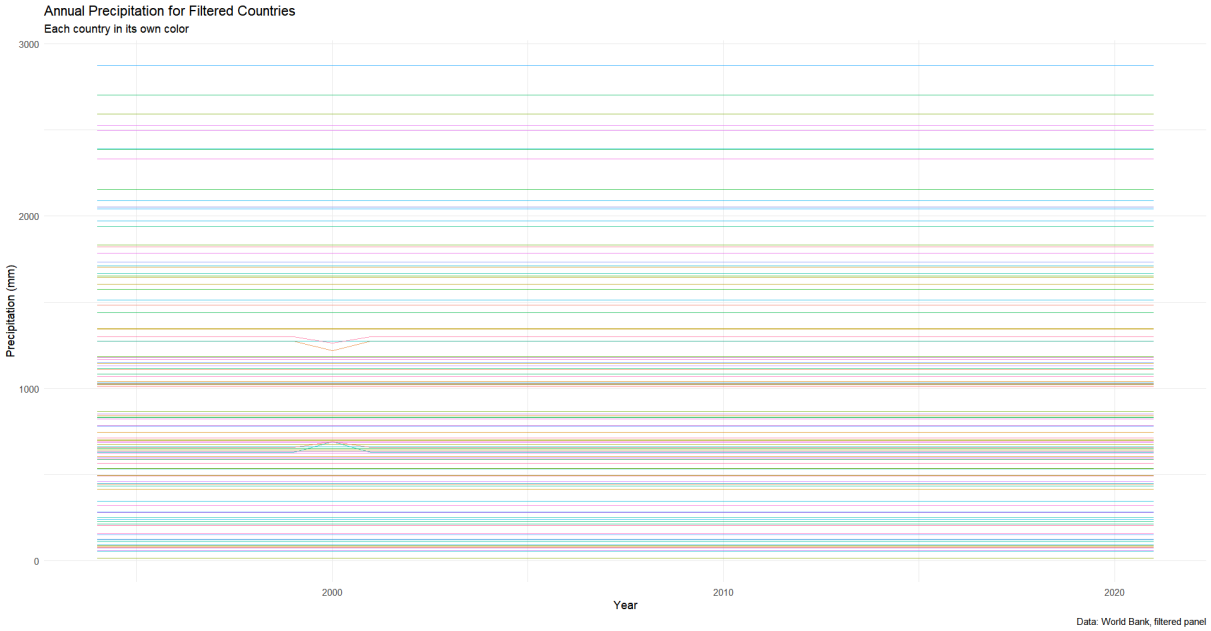


Figure - Annual Precipitation Time Series after Filtering

This is the time-series plot of the filtered panel of annual precipitation, with each country drawn in its own color. A striking feature is that the lines are almost flat, with little year-to-year variation.

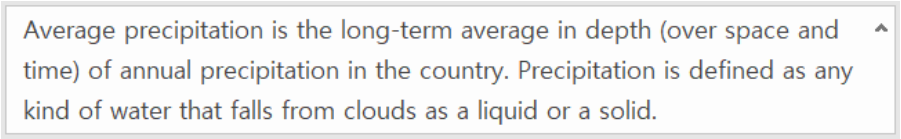


Figure - Detailed Description of Precipitation stated in the Metadata

The reason for the near-flat lines became clear once the metadata was consulted. By definition, “average precipitation” is not the annual total for each specific year but the long-term spatial and temporal average depth of precipitation in the country. In other words, each “year” in the panel simply repeats the same summary value rather than reflecting true year-to-year variability. This characteristic may limit the usefulness of the precipitation data for time-series analysis or for any other approach that requires year-to-year variability.

2.4. Additional Feature Derivation

By definition, several original features are functionally dependent. To recover possible latent information from these relationships, three additional features have been derived:

- Available resources:
  Derived from
  $$water\ stress\ =\ \frac{total\ withdrawals}{available\ resources}\ \Rightarrow\ available\ resources\ =\ \frac{total\ withdrawals}{water\ stress}$$

- GDP (USD):
  Derived from

$$water\ productivity\ =\ \frac{gdp\ (usd)}{total\ withdrawals}\ \Rightarrow\ gdp\ (usd)\ =\ water\ productivity\ \times\ total\ withdrawals$$

- GDP (PPP):
  Derived from
  $gdp\ per\ capita(PPP)\ \times\ population\ =\ gdp\ (ppp)$

Introducing these level-based features would uncover variation in absolute resource endowment and economic scale that percentage- or ratio-only metrics does not capture. However, for any given analysis, either the original ratios or the new levels—but not both—would have to be excluded to avoid multicollinearity while retaining the most relevant information for the task at hand.

3. Summary

The original World Bank dataset encompassed 266 countries over the period 1960–2023 but suffered from an extensive amount of missing values. By applying a filtering step, all years and countries with high missingness were removed to yield a continuous panel of 115 countries from 1994 to 2021, ensuring every core indicator could be directly compared across both space and time. Regression based imputation was then used to fill the remaining holes in the natural disaster series.
To handle the nearly empty private investment series, countries were recoded into a simple presence/absence label, enabling the creation of a complete panel from an otherwise sparse time-series dataset. Finally, three derived features—available water resources and GDP in both USD and PPP terms—were calculated to expose scale effects that ratios alone would not reveal. Normalization was deliberately postponed in order to allow each subsequent task to apply the most appropriate scaling.
Together, these steps produce a clean, analysis-ready dataset that supports exploratory analysis, statistical inference, and machine-learning applications. Even though these preprocessing steps are generally time-consuming and technically demanding, they are crucial for ensuring the accuracy and reliability of all subsequent analyses. If any errors or logical flaws are detected at this stage, the entire subsequent workflow would need to be revisited and reevaluated, which is an even more time-consuming endeavor.

Supervised Learning: Predicting Population Growth Categories from Water-Related Features

1. Introduction

Population-growth dynamics influence economic development, resource allocation, and environmental sustainability worldwide. This study classifies countries as “slow growth” or “rapid growth” based on demographic changes from 2002 to 2021, using water-related indicators to represent socio-economic and ecological pressures.
Each indicator was summarized in terms of its mean, variability, extremes, and long-term trend. Three machine-learning models—random forest, XGBoost, and GLMNET—were evaluated using nested cross-validation, with a random-baseline model included for comparison via hypothesis testing. Interpretability was enhanced through variable-importance rankings, odds-ratio estimates, and partial-dependence plots, and a global choropleth map was used to observe spatial patterns of misclassification.

2. R Libraries used

Library	Purpose
tidyverse	Data manipulation and visualization
caret	Training and tuning of machine learning models
MLmetrics	Provides the F1-score, precision, recall, and other classification metrics
pROC	Computes ROC curves and AUC (Area Under the Curve)
pdp	Generation of partial-dependence plots for model interpretation
rnaturalearth	Retrieval of global country geometries for mapping
sf	Handling and plotting of spatial data

3. Labels and Features

3.1. Label Construction

The population percentage growth rate for each country was calculated as:

$$growth\ rate\ = \frac{X2021 - X2002}{X2002} \times 100$$

where X2002 and X2021 are the population counts in those years. Countries with a growth rate below 30% were labeled Slow growth, and those at or above 30% were labeled Rapid growth.
.

The resulting label counts are:

- Rapid growth: 58
- Slow growth: 57

(total 115 countries), yielding an almost perfect 50/50 split. The 30% cutoff was deliberately chosen to achieve this balance, preventing majority-class bias during model training and ensuring that performance metrics such as accuracy and F1-score remain reliable. By balancing the classes up front, model comparisons focus on genuine predictive performance instead of compensating for label skew.

3.2. Feature Selection and Summary Statistics

A total of nine water-related time-series indicators were initially considered for inclusion—water_productivity, available_resources, water_stress, withdrawals, agriculture, domestic, industry, plus precipitation and natural_disasters. However, several were dropped for the following reasons:

1. Functional redundancies:
   Because agriculture, domestic, and industry sum to 100, the “domestic” feature was removed; likewise, since water_productivity and water_stress are calculated as GDP / withdrawals and withdrawals / available_resources respectively, the “withdrawals” feature was also excluded to avoid functional redundancy and prevent multicollinearity.

2. Lack of temporal dynamics:
   Precipitation and natural_disasters offered only long-term averages without year-to-year variation—precluding the computation of slopes, extrema, and standard deviations—so they were excluded for lacking the dynamic information required for feature engineering.

This left five core datasets—water_productivity, available_resources, water_stress, agriculture, and industry—each spanning 2002–2021. These indicators collectively capture distinct facets of a country’s water dynamics: water_productivity relates economic output to withdrawal volume, water_stress measures withdrawal pressure against renewable supply, available_resources quantifies the total endowment, and the agriculture and industry percentages reveal how water use is allocated between food production and industrial activity. It was deemed that no additional exclusions were necessary, since each feature seems to contribute complementary information beyond the functional redundancies already removed (domestic and withdrawals). Domain knowledge suggests these variables can indeed vary independently—for instance, high productivity in water-rich nations may coexist with low stress, and sectoral allocation often differs even among countries with similar

overall stress—indicating a reasonably comprehensive, non-redundant feature set for modeling.
For each retained indicator, five summary statistics were calculated per country—mean, standard deviation, minimum, maximum, and the linear trend (slope) over the twenty-year period. By including these statistics, the model can account for extreme events and long-term shifts as well as average behavior—insights that would be unavailable if only mean values were used.

4. Classification Methodology

4.1. Candidate Model Selection

Model	Description
Random Forest	Ensemble of decision trees using bagging; captures non-linear interactions; robust to noise and overfitting
XGBoost	Gradient-boosted decision trees; emphasizes errors from prior trees; strong performance with good tuning
GLMNET	Regularized logistic regression (L1/L2); interpretable coefficients; automatic feature selection
Linear SVM	Maximizes class margin with a linear boundary; effective in high-dimensional and small sample settings; limited interpretability

Three models were chosen to span distinct learning paradigms: a tree-based ensemble (random forest), a gradient-boosted method (XGBoost), and a penalized linear model (GLMNET). Random forests excel at capturing high-order interactions without overfitting, XGBoost refines those interactions through sequential boosting and built-in regularization, and GLMNET offers a transparent, coefficient-based view of linear relationships with automatic feature selection. Though only one model is ultimately selected for deployment, evaluating complementary models ensures that the decision is grounded in a comprehensive and well-contextualized framework.
Linear support vector machine could similarly provide a robust linear decision boundary and would likely perform competitively, given the modest size (`115 countries, 25 features`) of the dataset. However, it falls into the same linear paradigm as GLMNET and does not natively yield readily interpretable coefficients or odds ratios without additional post-hoc procedures. By contrast, GLMNET's regularization path explicitly shrinks and selects features, making its internal mechanics directly transparent to practitioners. Thus, GLMNET was chosen over SVM not due to anticipated performance advantages but to its interpretability and transparency.

4.2. Nested Cross-Validation Procedure

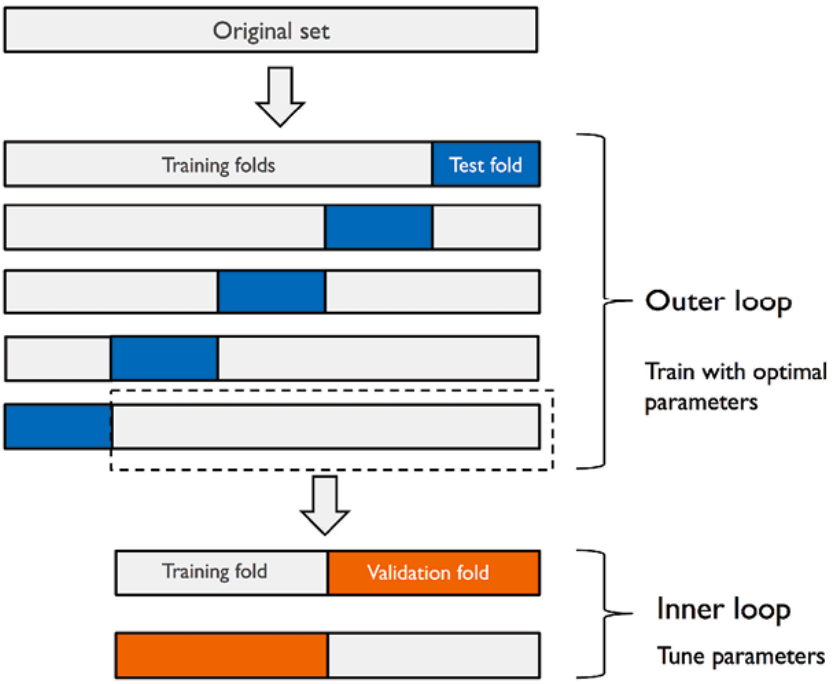


Figure - Structure of the nested cross-validation framework used in this study.

To robustly evaluate model performance and optimize the parameters of each model, a nested cross-validation framework was employed. The outer loop consisted of a 5-fold split of the full dataset. For each outer fold, the data was partitioned into training and testing subsets, with the testing portion held out entirely for final evaluation. This structure ensures that hyperparameter tuning occurs independently of the evaluation data, yielding more reliable performance estimates.

Within each outer training fold, the hyperparameters of the three models—random forest, XGBoost, and GLMNET—were tuned using an inner loop of repeated 5-fold cross-validation with 10 repetitions. A random search approach was used for hyperparameter tuning, with F1-score selected as the primary optimization metric, as further discussed in Section 4.3. Preprocessing steps were applied according to model requirements: for GLMNET, features were centered and scaled to ensure stable coefficient estimation and effective regularization. Range scaling was applied for random forest and XGBoost. However, it proved unnecessary: the models were run again without scaling, and the results remained unchanged. This is likely because these algorithms rely on threshold-based splits and are inherently insensitive to feature magnitude.

After hyperparameter tuning within the inner loop, each model was used to generate predictions on the outer test data. Performance was assessed using both F1-score and accuracy, and results were aggregated across all outer folds. As will be discussed in Section 6, GLMNET was ultimately selected as the final model; thus, additional metrics including AUC, sensitivity, specificity, and out-of-fold predictions were specifically recorded for GLMNET to support further interpretation and mapping.

Note on Feature Selection:
During preliminary modeling, feature selection procedures based on variable importance were explored for the Random Forest and XGBoost models. However, these steps did not yield consistent performance improvements, and any potential benefit appeared to be

masked by variability introduced from different seed initializations. Therefore, these procedures were ultimately discarded to streamline computation and reduce unnecessary complexity.

4.3. Evaluation Metric Selection

Metric	Description
Accuracy	The proportion of total correct predictions (both positive and negative) out of all predictions. Used for model comparison but not for tuning in this study.
F1-score	The harmonic mean of precision and recall. Prioritizes balance between false positives and false negatives. Used for both hyperparameter tuning and model comparison.
AUC	Area Under the ROC Curve. Evaluates a model's ability to distinguish between classes by ranking predicted probabilities. Not used for tuning or comparison, but provides complementary insight into ranking ability.

As previously stated, a random search approach was used for hyperparameter tuning, with F1-score selected as the primary optimization metric. Although the dataset was nearly balanced and accuracy was initially considered as the optimization metric, it was later replaced by F1-score due to an observed imbalance between precision and recall in the result. This choice reflects the recognition that both false positives and false negatives may carry significant and distinct costs in the classification context. However, when comparing the final performance of the three models, both accuracy and F1-score were reported to provide a more comprehensive evaluation.
While AUC provides valuable insight into a model's ability to rank predictions, it was not used for either tuning or final model comparison, as the objective of this study was to produce discrete class predictions rather than probability-based rankings.

5. Model Evaluation and Comparison

5.1. Baseline Model Introduction

```
> nested_results[16:20,]
   fold          model        F1  Accuracy
16    1 RandomBaseline 0.4444444 0.5454545
17    2 RandomBaseline 0.3846154 0.3043478
18    3 RandomBaseline 0.6153846 0.5833333
19    4 RandomBaseline 0.5600000 0.5217391
20    5 RandomBaseline 0.4545455 0.4782609
```

Table - Cross validation performance of the random baseline model

To provide a meaningful performance reference, a random baseline classifier was implemented. This model predicts each class label ("Slow growth" or "Rapid growth") uniformly at random, simulating the performance of a no-skill model in a balanced binary classification setting.

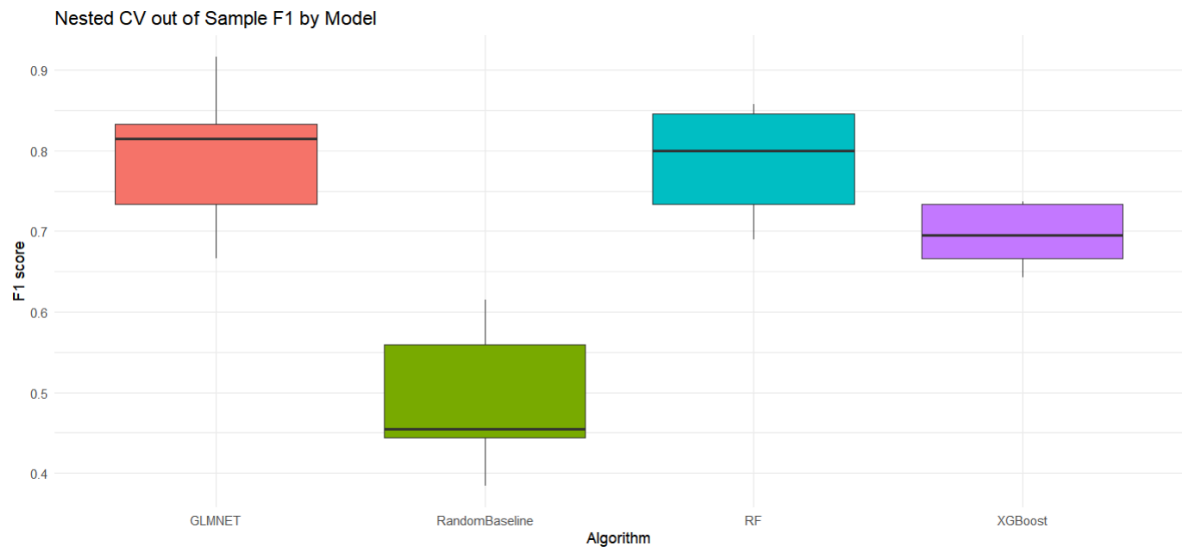
The baseline model was evaluated across the same five outer folds as the candidate models. As shown in the table above, the baseline yielded a mean F1-score of 0.49 (±0.09) and a mean accuracy of 0.49 (±0.10). These results reflect the expected performance level of a naive classifier and serve as a lower bound for meaningful model performance.

5.2. Model Performance Summary and Visualization

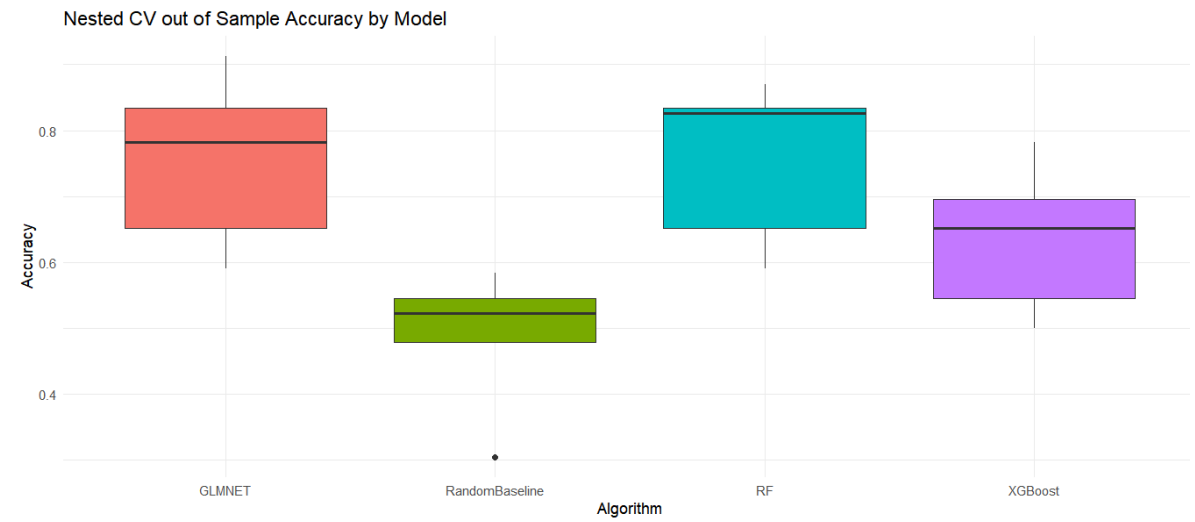
F1-score and accuracy were calculated across the five outer folds to assess model performance. The table below reports the mean and standard deviation for each model, while the two boxplots visualize their distribution:

```
> print(summary_tbl)
# A tibble: 4 × 5
  model           mean_F1  sd_F1 mean_Accuracy sd_Accuracy
  <chr>             <dbl>  <dbl>         <dbl>       <dbl>
1 GLMNET            0.793 0.0961         0.754       0.132
2 RF                0.785 0.0723         0.754       0.124
3 RandomBaseline    0.492 0.0936         0.487       0.109
4 XGBoost           0.695 0.0411         0.635       0.114
```

Table - Mean and Standard Deviation of F1-score and Accuracy across Outer Folds



Boxplot - Distribution of F1 score across Outer Folds



Boxplot - Distribution of Accuracy across Outer Folds

Both GLMNET and Random Forest achieved comparable performance, with nearly identical mean accuracy and F1-scores. XGBoost lagged slightly behind, in terms of both F1-score and accuracy. However, all three models substantially outperformed the random baseline, suggesting that the water-related features provided meaningful signal for classification. The standard deviations (SD) of F1-score across models were moderate—0.096 for GLMNET, 0.072 for Random Forest, and 0.041 for XGBoost—indicating reasonably consistent performance across folds. While the limited dataset size could have introduced greater variability, the use of nested cross-validation appears to have mitigated this risk to the present extent.

```
> print(glmnet_summary_ci)
  mean_Sensitivity ci_Sensitivity_L ci_Sensitivity_U mean_Specificity
ci_Specificity_L
1        0.8969697        0.8056244         0.988315        0.6106061
0.2931717
  ci_Specificity_U  mean_AUC  ci_AUC_L ci_AUC_U
1        0.9280404 0.7868113 0.5695778 1.004045
```

Table - Summary of Additional Performance Metrics for GLMNET

As GLMNET was selected as the representative model for interpretation (see Section 6), its sensitivity, specificity, and AUC across the outer folds were additionally evaluated to provide a more comprehensive understanding of model performance. The results were:

- Sensitivity: 0.897 [95% CI: 0.806, 0.988]
- Specificity: 0.611 [95% CI: 0.293, 0.928]
- AUC: 0.787 [95% CI: 0.570, 1.004]

These results suggest that the model performs strongly in identifying rapid-growth countries, as indicated by the high sensitivity with a relatively narrow confidence interval. However, the broader confidence interval for specificity reflects less stable performance in correctly

identifying slow-growth cases, primarily due to the modest dataset size and associated variability across folds. The AUC value indicates a solid overall discriminative ability, although the upper bound slightly exceeding 1 is a statistical artifact resulting from normal approximation; AUC is theoretically bounded between 0 and 1 and should be interpreted as approaching but not exceeding 1.

5.3. Pairwise Model Comparison via Statistical Testing

```
> print(results_df)
# A tibble: 3 × 5
  Comparison        wilcox_p_f1 mean_diff_F1 wilcox_p_accuracy mean_diff_Acc
  <chr>                   <dbl>        <dbl>             <dbl>         <dbl>
1 GLMNET vs RF            0.584      0.00771               1             0
2 GLMNET vs XGBoost       0.100      0.0979                0.100         0.119
3 RF vs XGBoost           0.100      0.0902                0.100         0.119
```

Table - Wilcoxon Signed-Rank Test Results and Mean Differences in F1-score and Accuracy Between Models

Pairwise Wilcoxon signed-rank tests were conducted to assess whether differences in F1-score and accuracy between models were statistically significant. These non-parametric tests were chosen over paired t-tests due to the small sample size and potential violations of normality and homogeneity of variances in the performance metrics. The results are summarized in the above Table.

The comparison between GLMNET and Random Forest yielded high p-values (0.584 for F1-score and 1.000 for accuracy), indicating no statistically significant difference in performance between these two models. In contrast, comparisons involving XGBoost showed lower p-values (0.100 for both F1-score and accuracy), suggesting a possible trend toward better performance by GLMNET and Random Forest. The mean differences in F1-score were approximately 0.098 (GLMNET vs. XGBoost) and 0.090 (RF vs. XGBoost), while the corresponding differences in accuracy were both around 0.119.

Although these differences did not reach conventional thresholds for statistical significance ($p < 0.05$), the consistency in direction across both metrics indicates that GLMNET and Random Forest may have offered better performance than XGBoost in this context. Nevertheless, due to the limited sample size and moderate variance, these findings should be interpreted with caution.

6. GLMNET Interpretation and Error Analysis

For further interpretation, GLMNET was chosen due to its strong and stable performance shown in earlier testings, as well as its clear advantage in interpretability. While both GLMNET and Random Forest achieved similar performance, GLMNET provides direct access to model coefficients and supports straightforward techniques such as odds ratio analysis and partial dependence plots. The following subsections therefore focus on interpretive analysis of the GLMNET model and explore misclassification patterns across countries.

6.1. Variable Importance, Coefficients, and Feature Effects

6.1.1. Variable Importance Rankings

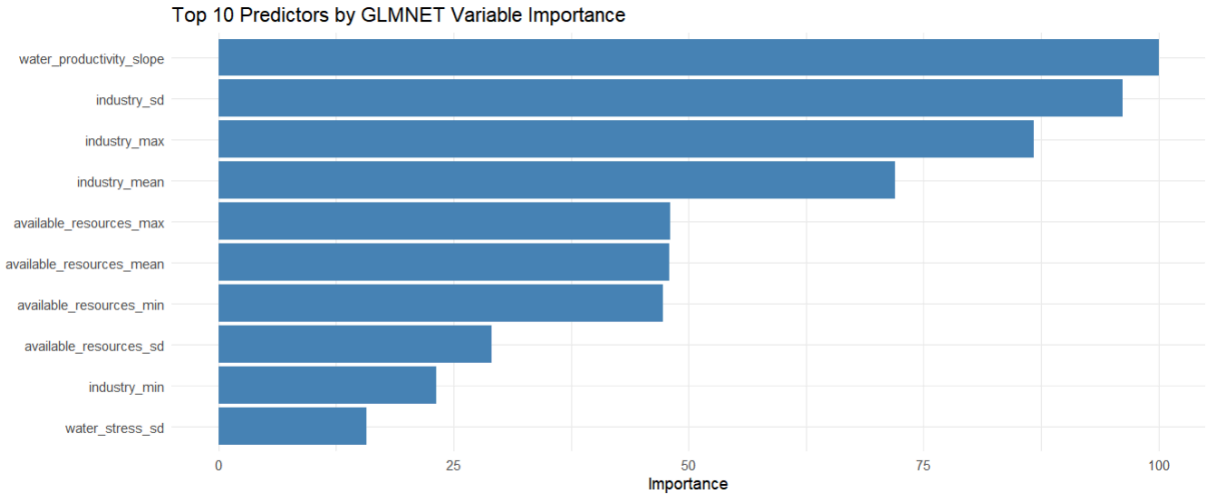


Figure - Top 10 predictors by GLMNET variable importance
(Higher values indicate stronger contribution to model predictions.)

To better understand which features contributed most strongly to the classification decision, the GLMNET model was first re-tuned on the full dataset using repeated cross-validation, and then refit using the best hyperparameters for interpretation. Variable importance scores were subsequently computed from this final model to highlight the most influential predictors. As shown in the figure above, the top-ranked feature was the slope of water productivity over the 2002–2021 period. This feature captures the direction and rate of long-term change in how efficiently water is used for economic output, rather than a static snapshot. Its prominence suggests that sustained improvements or declines in water productivity are strongly associated with a country's population growth category.
Several indicators related to industrial water use followed closely, reflecting the influence of both magnitude and variability in this sector. Available water resources also emerged as important predictors (with importance scores around 50%), suggesting that countries with greater or more stable access to natural water resources may follow distinct demographic trajectories. Lastly, variability(standard deviation) in water stress appeared as the tenth most influential feature, although its importance score was below 20%—considerably lower than that of the leading predictors. This suggests that while it was not a dominant factor, fluctuations in water system pressure may still contribute meaningfully to the model's decision-making, particularly when considered alongside more influential variables.

6.1.2. Coefficient Interpretation

In addition to variable importance, GLMNET coefficients were examined to assess the direction and magnitude of each feature's effect. Coefficients were converted to odds ratios and labeled to indicate whether higher values increased the likelihood of a country being classified as "Rapid growth" or "Slow growth," offering interpretable insight into how water-related trends relate to demographic patterns. A summary of the top 10 coefficients is provided in the table below.

```
> print(coefs_df)
```

	term	estimate	odds_ratio	direction
1	water_productivity_slope	0.6091431	1.8388551	↑ Rapid_growth
2	industry_sd	-0.5856955	0.5567186	↑ Slow_growth
3	industry_max	-0.5285421	0.5894637	↑ Slow_growth
4	industry_mean	-0.4387476	0.6448435	↑ Slow_growth
5	available_resources_max	-0.2926670	0.7462706	↑ Slow_growth
6	available_resources_mean	-0.2922546	0.7465785	↑ Slow_growth
7	available_resources_min	-0.2881981	0.7496131	↑ Slow_growth
8	available_resources_sd	-0.1767930	0.8379532	↑ Slow_growth
9	industry_min	-0.1412537	0.8682690	↑ Slow_growth
10	water_stress_sd	0.0957436	1.1004769	↑ Rapid_growth

Table - Top 10 GLMNET Coefficients with Odds Ratios and Classification Direction. Positive coefficients indicate association with the "Rapid growth" class; negative values indicate association with "Slow growth".

The coefficient patterns reveal a clear contrast between features associated with rapid versus slow growth. All indicators related to industrial water use and available water resources had negative coefficients and odds ratios below one, pointing to a consistent association with the “Slow growth” category. In contrast, water productivity slope had the largest positive coefficient, with an odds ratio of approximately 1.84, linking sustained improvements in water productivity to rapid population growth. A smaller positive effect was observed for water stress variability, suggesting that fluctuating pressure on water systems may also play a role in characterizing fast-growing countries. Collectively, these results suggest that both long-term trends and resource variability are relevant factors in demographic outcomes.

6.1.3. Partial Dependence Analysis

Partial dependence plots (PDPs) were generated for the three most influential predictors: water_productivity_slope, industry_sd, and industry_max. These plots illustrate the isolated effect of each feature on the predicted probability of a country being classified as “Rapid growth,” by marginalizing over the joint distribution of all other features. This approach enables an interpretable approximation of how the model responds to changes in a single variable, independent of interactions with other predictors.

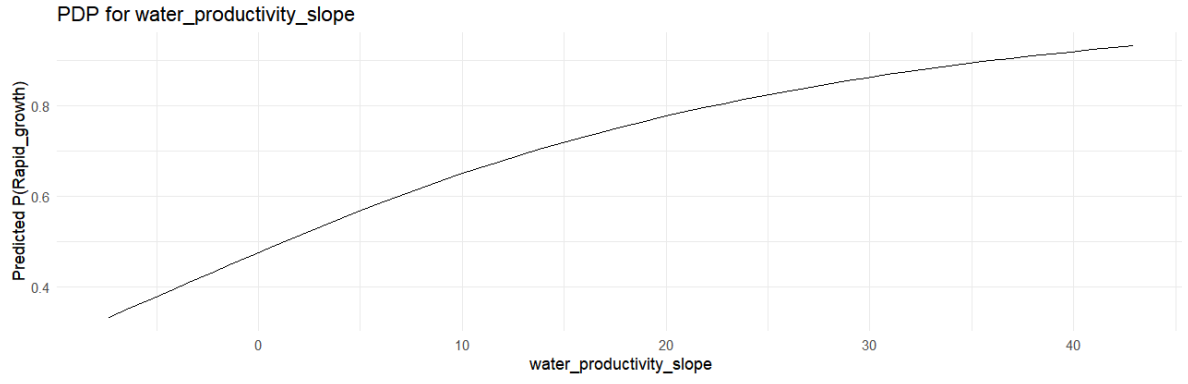


Figure - Partial dependence plot for water productivity slope

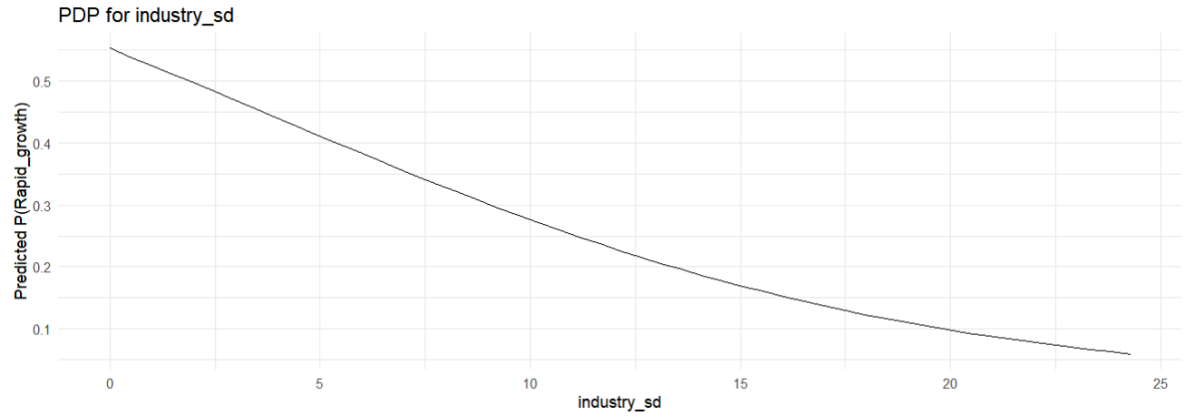


Figure - Partial dependence plot for industrial withdrawals sd(standard deviation)

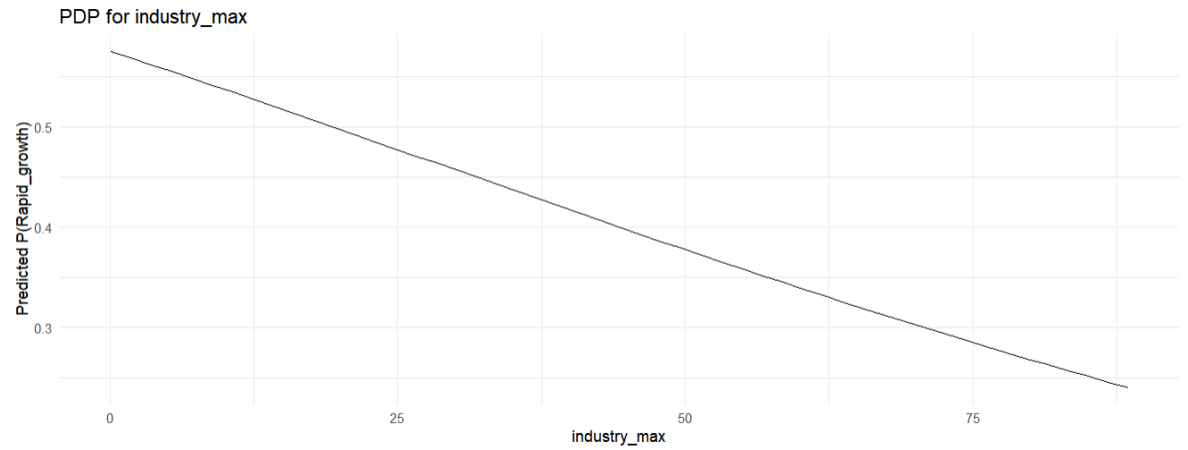


Figure - Partial dependence plot for industrial withdrawals max

The first figure above shows that higher values of water_productivity_slope are associated with a steadily increasing probability of rapid growth. This aligns with previous findings that countries experiencing consistent improvements in water productivity are more likely to fall into the high-growth category. The effect appears smoothly positive and nonlinear, with a gradually decreasing slope.

In contrast, both industry_sd and industry_max exhibit strong negative relationships with the probability of rapid growth (second and third plots). Also, the effect of industry_max is nearly linear, suggesting a consistent downward influence across its range. These trends support the interpretation that excessive or unstable industrial water demand may be more characteristic of slower-growing populations.

Together, these PDPs illustrate how the GLMNET model integrates both static and dynamic features into its decision-making, offering interpretable insights into the relationships between water dynamics and population growth.

Note: The interpretations provided here are specific to the GLMNET model. Other algorithms, such as Random Forest or XGBoost, which capture nonlinear interactions and complex feature hierarchies, may prioritize different features and yield different patterns of importance. While GLMNET offers a transparent and interpretable view of the data, its

perspective should be understood as one among several possible representations learned by different model classes.

6.2. Country-Level Misclassification Analysis

To complement the interpretation of model behavior, performance was also examined at the country level. The following map visualizes which countries were misclassified by the GLMNET model during cross-validation.

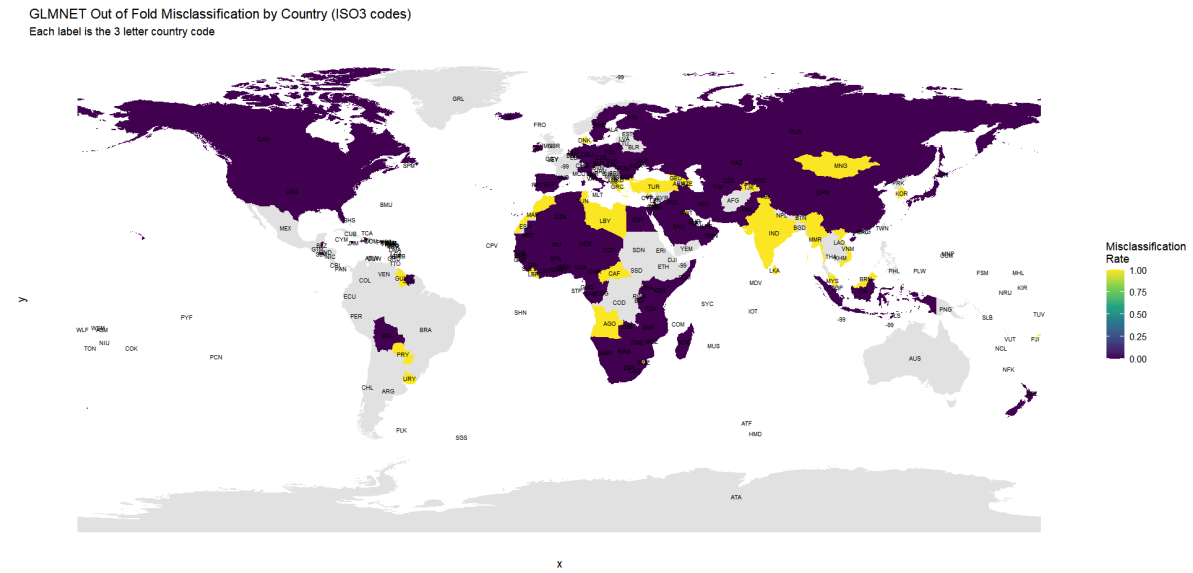


Figure – Country-Level Classification Outcomes from GLMNET (Out-of-Fold Predictions)

This choropleth map highlights spatial patterns in misclassification based on out-of-fold predictions from the GLMNET model. Each country is shaded according to whether its single out-of-fold prediction was correct or incorrect during nested cross-validation. While most countries were classified correctly (shown in purple), a subset—particularly in Sub-Saharan Africa, Latin America, parts of the Middle East and Asia—were misclassified. These geographic patterns may reflect regional limitations in the feature set, data quality inconsistencies, or contextual factors not captured by the model. Such spatial diagnostics can help identify areas where predictive performance is less reliable and warrant further investigation.

7. Reflections and Possible Applications

While the analysis yielded interpretable results and moderately strong predictive performance, several limitations warrant cautious interpretation. Chief among these is the modest dataset size: with only 115 countries and a limited number of temporally aggregated features, the models faced inherent constraints in generalizability. Although the standard deviations of F1-score and accuracy across folds were not excessively high, some variability was still present. This was further reflected in the wide confidence intervals of secondary metrics such as specificity and AUC, highlighting not only variability across folds but also the model's limited reliability in identifying slow-growth countries.

Moreover, the interpretation of model behavior was specific to the GLMNET classifier and may not generalize across algorithms. Models such as Random Forest or XGBoost—which capture nonlinear interactions and hierarchical structures—would likely have yielded different feature rankings and response patterns.
Future analyses could benefit from expanding the dataset both in terms of size and the variety of features. This might involve including more countries, using longer time periods, or adding more detailed variables. For example, more complex patterns could be captured by using lagged inputs (e.g., previous year's values), or by retrieving dynamic precipitation and natural disaster data from alternative sources, as these were previously excluded due to being available only as long-term averages.
On the methodological side, techniques such as averaging across multiple models (ensemble methods) may help improve the reliability and generalizability of the results, while repeating the training process with different random seeds (a seed sensitivity test) would ensure that the findings are not overly dependent on chance.
Lastly, since the current threshold was chosen primarily to ensure a balanced split between the two classes, refining the labeling strategy—based on theoretical reasoning or expert guidance—could lead to a more meaningful and interpretable classification of population growth.

Despite its limitations, the current modeling framework offers a tentative foundation for exploring the relationship between water-related characteristics and demographic trends. While the results are not definitive, they suggest a few possible directions for practical reflection:

1. Early signals of demographic pressure: Countries showing consistent improvements in water productivity may be more likely to sustain rapid population growth. Monitoring long-term gains in water productivity could thus offer an early indication of developmental momentum.

2. Industrial water use as a potential constraint: High variability or peaks in industrial water withdrawals appeared to be associated with slower-growing populations. While the causal relationship is unclear, such patterns may signal inefficiencies or infrastructural limitations that could impact demographic outcomes over time.

3. Resource availability and planning: While not the most dominant predictors, the static measures of available water resources occupied a solid mid-range in importance. Their consistent association with slower growth may suggest underlying structural dependencies—such as the potential for complacency or inefficiencies in settings where water is relatively abundant.

While these insights are exploratory and model-dependent, they may nonetheless contribute to broader conversations in areas such as water resource governance, urban planning, and development strategy. That said, the model's difficulty in consistently identifying slow-growth countries warrants particular caution when interpreting results for more demographically stagnant regions, where predictive uncertainty may be higher. Ultimately, this study illustrates that even relatively simple water indicators, when systematically modeled, can provide valuable perspectives on the complex interplay between environmental factors and

population dynamics—though more robust, theory-driven approaches would be necessary to support confident, real-world decision-making.

8. Conclusion

This study explored whether long-term water-related indicators could help classify countries into distinct population growth categories. By leveraging three supervised learning models—GLMNET, Random Forest, and XGBoost—and applying a nested cross-validation framework, the analysis aimed to capture the most comprehensive model capacity, especially in terms of both predictive performance and interpretability.

Among the models tested, GLMNET emerged as the most interpretable and comparably effective, offering insights into how variables such as water productivity trends and industrial water use variability relate to demographic dynamics. The findings suggested that countries with sustained improvements in water productivity were more likely to exhibit rapid growth, while high variability or peaks in industrial water use tended to align with slower growth. Available water resources—although not the most influential predictors—were also consistently associated with slower-growing populations, hinting at underlying resource management or infrastructure patterns.

Nevertheless, these findings should be interpreted cautiously. The modest dataset and limited set of aggregated features could affect generalizability. Additionally, since interpretation was based solely on the GLMNET model, the results may not reflect the behavior of more complex algorithms. Future studies could enhance robustness by expanding the dataset, incorporating more diverse or dynamic features, applying ensemble methods, and refining the classification threshold based on theoretical or expert input.

Despite its exploratory nature, this analysis illustrates that even modest datasets and simple indicators—when analyzed systematically—can reveal suggestive patterns worth further investigation. While not definitive, the framework may offer a useful starting point for future work in water governance, resource planning, and environmental-demographic modeling.

Unsupervised Learning: Principal Component Analysis(PCA) and Clustering Analysis of Water-Related Indicators

1. Introduction

This study applies unsupervised learning to group countries based on key water-related indicators. K-means, hierarchical clustering, and DBSCAN were used, together with Principal Component Analysis (PCA) for dimensionality reduction. All features were averaged over 2012–2021 to emphasize recent trends and reduce short-term fluctuations. Multiple feature combinations and clustering parameters were evaluated, with the best configuration selected based on internal validity metrics.

2. R Libraries used

Library	Purpose
tidyverse	Data wrangling and visualization
cluster	Computes silhouette scores for clustering evaluation
factoextra	PCA and cluster visualization
clusterSim	Calculates the Davies-Bouldin index for cluster evaluation
dbscan	Density-based clustering (DBSCAN) and outlier detection
rnaturalearth & rnaturalearthdata	Retrieves and manages natural Earth geographic data
sf	Spatial data handling for map visualization
stringr	String manipulation (used in feature set handling)
scales	Data rescaling for metric normalization

3. Clustering Methodology

3.1. Feature Selection & Normalization

The purpose of clustering was defined as grouping countries based solely on water-related characteristics, with the aim of uncovering underlying patterns in water usage, availability, and stress—independent of direct economic or demographic influences.
Accordingly, the following indicators were selected as potential clustering features:

- Available Water Resources
- Water Productivity
- Total Withdrawals
- Sectoral Withdrawals Proportions (Industry, Agriculture, Domestic)
- Precipitation
- Water Stress
- Natural Disasters

As will be discussed later, almost all possible subsets of these features were examined in the clustering process.

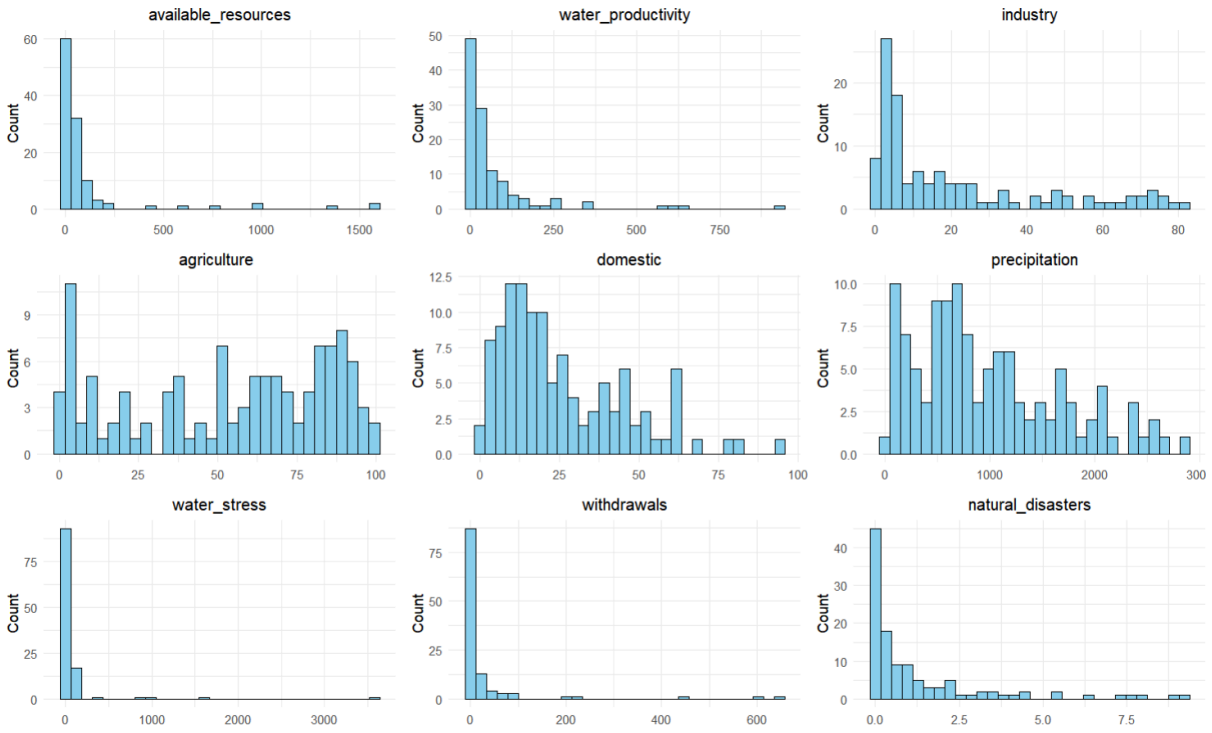


Figure - Distribution of Selected Features (2012–2021 Average)

The above figure presents the distribution of the selected water-related indicators based on their 2012–2021 averages.
As visualized in the feature-wise histograms, most variables exhibit non-normal, right-skewed, and highly heterogeneous distributions. Several features, including available water resources, withdrawals, and natural disasters, show extreme outliers and heavy tails, while others like agriculture and precipitation suggest multi-modal patterns.
These characteristics point to a dataset that may challenge traditional clustering methods relying on assumptions of convexity, uniform density, or balanced variance across clusters. Accordingly, clustering algorithms capable of handling irregular shapes, density variation, and outlier sensitivity might be more appropriate in this context.

```
> for (features in feature_sets) {
+   scaled_data <- scale(aggregated_data[, features, drop = FALSE])
```

Code Snippet – Normalization of Clustering Features

The selected clustering features were normalized (scaled) to ensure comparability across variables and to prevent any single feature from dominating the clustering results due to differences in scale.

3.2. Model Selection

Three clustering algorithms were selected for this study: K-means, hierarchical clustering, and DBSCAN. These models were chosen based on their widespread use, complementary strengths, and alignment with the interpretability goals of the analysis.
Although the data distributions presented earlier were skewed, non-spherical, and heterogeneous—conditions that are not ideal for K-means—this method was included as a baseline. Its simplicity, efficiency, and ease of implementation make it a valuable reference point. Moreover, K-means remains a widely adopted standard in exploratory clustering, offering a quick overview of global grouping tendencies.
Hierarchical clustering was selected to address structural limitations observed during the K-means exploration. In some clusters, a majority of countries came from two geographically distinct regions—for example, Europe and Africa—suggesting internal regional heterogeneity. Further subdivision could reveal more meaningful groupings. Hierarchical clustering supports this by capturing multi-level relationships and enabling flexible interpretation through dendrograms, without requiring a predefined number of clusters.
DBSCAN was included for its robustness in handling irregular shapes and outliers—both clearly present in the data. Earlier feature distributions showed skewed, multi-modal, and non-spherical patterns, making conventional centroid-based methods less effective. Unlike K-means and hierarchical clustering, DBSCAN relies on local density rather than global distance metrics or a fixed number of clusters. This allows it to identify arbitrarily shaped groupings and exclude noise points, making it well-suited for the heterogeneous distributions found in cross-national water-related datasets.
Together, the selected models offer a well-rounded approach: K-means provides a baseline, hierarchical clustering reveals nested structures, and DBSCAN captures non-linear patterns and outliers—all aligned with the study's goal of uncovering meaningful country groupings based on water-related characteristics.
Although Gaussian Mixture Models (GMM) might be considered appropriate for this dataset—given their ability to handle non-spherical clusters and overlapping group structures—they were excluded from the model selection. While GMM may offer advantages in identifying transitional or mixed-pattern countries, its soft clustering approach, which assigns probabilistic rather than fixed labels, was not aligned with the study's goal of producing clear and interpretable groupings for policy use. The decision was based not on performance concerns, but on the reduced practicality of applying probabilistic assignments in a real-world policy context. Hard clustering methods such as K-means, hierarchical clustering, and DBSCAN were prioritized for their clearer interpretability.

3.3. Principal Component Analysis

```
> pca_all <- prcomp(scale(aggregated_data), scale. = FALSE)
```

```
> pca_summary <- summary(pca_all)
> print(pca_summary)
Importance of components:
                          PC1    PC2    PC3    PC4     PC5     PC6    PC7
PC8     PC9
Standard deviation     1.6557 1.3350 1.1158 1.1082 0.87717 0.79418 0.6163
0.47229 0.01969
Proportion of Variance 0.3046 0.1980 0.1383 0.1364 0.08549 0.07008 0.0422
0.02478 0.00004
Cumulative Proportion  0.3046 0.5026 0.6410 0.7774 0.86289 0.93297 0.9752
0.99996 1.00000
```

Code Snippet – Dimensionality Reduction: PCA Variance Analysis

Principal Component Analysis (PCA) was conducted to assess the variance contributions of each principal component and, based on this information, to determine an appropriate number of components for dimensionality reduction prior to clustering. The analysis showed that the first four components captured approximately 77.7% of the total variance, and the first six explained over 93%, indicating that a reduced representation using the first 4 to 6 components retains most of the information in the original feature space.

3.4. Grid Search

Clustering tasks inherently lack a clear ground truth, making it difficult to assert with certainty that any one model or configuration is universally optimal. Therefore, this study exhaustively evaluated all possible combinations of feature subsets, clustering algorithms (K-means, hierarchical, DBSCAN), algorithm-specific parameters, and the number of principal components used for dimensionality reduction. This was implemented through fully nested loops, ensuring that no plausible configuration was left untested within the defined search space.

```
> feature_sets <- list()
> combo_id <- 1
> for (k in 2:length(raw_feature_sources)) {
+  combos <- combn(raw_feature_sources, k, simplify = FALSE)
+  for (combo in combos) {
+    feature_sets[[paste0("Set_", combo_id)]] <- combo
+    combo_id <- combo_id + 1
+  }
+ }
```

Code Snippet – Feature Subset Generation for Grid Search

As part of the grid search, all possible combinations of water-related features from size 2 to the full set, were systematically generated. One-variable sets were excluded—not due to performance concerns, but to ensure interpretability. Clusters based on a single feature would lack dimensional context and likely be monotonous, making it difficult to derive meaningful conclusions for analysis or policy interpretation.

```
> num_pc_options <- 2:8
> epsilon_values <- seq(0.0, 2.0, by = 0.2)
> minpts_values <- 2:8
> cluster_counts <- 2:4
> results <- data.frame()
```

Code Snippet – Parameter Ranges for Grid Search

Based on the previous PCA results, which showed that the first 4 to 6 components explained approximately 78% to 93% of the total variance, principal components from 2 to 8 were tested to allow some margin around this core range.
For DBSCAN, initial experiments tested a wide parameter range — eps values from 0 to 4, and minPts values from 2 to 12 — to account for potential variability in cluster density across countries. However, well-performing configurations (particularly those with silhouette scores above 0.5) were consistently concentrated within the narrower range of eps = 0 to 2 and minPts = 2 to 8. Based on these observations, the parameter space was refined in the final grid search to enhance computational efficiency and reproducibility.
The number of clusters for K-means and hierarchical clustering was limited to 2–4, since one cluster offers no useful grouping, and having too many clusters could make interpretation difficult.

4. Results and Interpretation

4.1. Clustering Configuration Selection

	Method	Feature_Set	PCs	Param	Num_Clusters	Silhouette	DB_Index	Labels
579	Hierarchical	available_resources, withdrawals	2	2	2	0.9154100	0.28347158	1, 1, 1,....
2803	DBSCAN	water_stress, withdrawals	2	eps=1.8, minPts=2	2	0.9112008	0.24136186	1, 1, 0,....
2804	DBSCAN	water_stress, withdrawals	2	eps=1.8, minPts=3	2	0.9112008	0.24136186	1, 1, 0,....
2818	KMeans	water_stress, withdrawals	2	3	3	0.9112008	0.42321758	2, 2, 1,....
2821	Hierarchical	water_stress, withdrawals	2	3	3	0.9065226	0.20548475	1, 1, 1,....
2810	DBSCAN	water_stress, withdrawals	2	eps=2, minPts=2	2	0.9065226	0.26617159	1, 1, 1,....
2811	DBSCAN	water_stress, withdrawals	2	eps=2, minPts=3	2	0.9065226	0.26617159	1, 1, 1,....
576	KMeans	available_resources, withdrawals	2	2	2	0.9058169	0.53377466	2, 2, 2,....
2782	DBSCAN	water_stress, withdrawals	2	eps=1.2, minPts=2	2	0.9014863	0.11988371	1, 1, 0,....
2789	DBSCAN	water_stress, withdrawals	2	eps=1.4, minPts=2	2	0.9014863	0.11988371	1, 1, 0,....
2796	DBSCAN	water_stress, withdrawals	2	eps=1.6, minPts=2	2	0.9014863	0.11988371	1, 1, 0,....
2817	KMeans	water_stress, withdrawals	2	2	2	0.8971020	0.34276329	2, 2, 2,....
2820	Hierarchical	water_stress, withdrawals	2	2	2	0.8971020	0.34276329	1, 1, 1,....
562	DBSCAN	available_resources, withdrawals	2	eps=1.8, minPts=2	2	0.8904171	0.25632549	1, 1, 1,....
563	DBSCAN	available_resources, withdrawals	2	eps=1.8, minPts=3	2	0.8904171	0.25632549	1, 1, 1,....

Showing 1 to 15 of 148,736 entries, 8 total columns

Figure – Top Clustering Results by Silhouette Score

This figure shows a subset of the 148,736 clustering configurations, sorted in descending order of silhouette score. Although several top results show silhouette scores above 0.9 — suggesting high internal cohesion — further inspection revealed significant cluster size imbalance. In some cases, for example, the model produced two clusters where one cluster contained only two countries, while the remaining countries were all grouped into the other

cluster. Such results, despite their high internal metrics, do not yield meaningful or interpretable insights.
This also suggests a limitation of widely used clustering evaluation metrics, silhouette score and DB index, which do not account for cluster size balance.
To address this, a new cluster balance metric called MaxMinRatio was introduced, which measures the ratio between the largest and smallest cluster sizes (excluding noise points). Configurations with a MaxMinRatio above 10 were considered unbalanced and were filtered out, ensuring that only well-structured and interpretable clustering results were retained for further evaluation.
In addition, configurations with more than 20% missing or unassigned labels were excluded to ensure sufficient coverage of countries in the analysis.

```
> # Normalize Silhouette and DBI, then compute composite score
> results_clean$Silhouette_Norm <- rescale(results_clean$Silhouette, to = c(0, 1))
> results_clean$DBI_Norm_Inv <- 1 - rescale(results_clean$DB_Index, to = c(0, 1))
> results_clean$CompositeScore <- 0.5 * results_clean$Silhouette_Norm + 0.5 * results_clean$DBI_Norm_Inv
```

Code Snippet – Composite Clustering Score Based on Silhouette and DB Index

To build a reliable ranking of clustering configurations, a composite score combining the silhouette score and the Davies–Bouldin index (DBI) was constructed. This was necessary because a high silhouette score does not always correspond to a low (desirable) DBI, and vice versa. Also, these two metrics were selected for their complementary focus: silhouette captures separation and cohesion, while DBI emphasizes intra-cluster compactness relative to inter-cluster separation.
Both metrics were normalized to a 0–1 scale, with DBI inverted via (1 − normalized DBI) to ensure that higher values consistently indicated better clustering performance. Then, an equal weighting of 0.5 was applied to balance both perspectives without introducing subjective bias.
While alternative scoring schemes are possible, this design prioritizes interpretability and neutrality, which are critical in unsupervised model selection.

	Method	Feature_Set	PCs	Param	Num_Clusters	Silhouette	DB_Index	Labels	MaxMinRatio	MissingRate	Silhouette_Norm	DBI_Norm_Inv	CompositeScore
1	DBSCAN	available_resources, industry	2	eps=0.4, minPts=8	2	0.6664886	0.3776357	1, 1, 1,...	6.133333	0.06956522	0.8201664	0.9284760	0.8743212
2	DBSCAN	industry, water_stress	2	eps=0.2, minPts=7	2	0.6074117	0.2164016	1, 1, 0,...	7.727273	0.16521739	0.7774863	0.9614308	0.8694585
3	DBSCAN	industry, withdrawals	2	eps=0.4, minPts=8	2	0.6333523	0.3739639	1, 1, 1,...	6.714286	0.06086957	0.7962271	0.9292265	0.8627268
4	DBSCAN	available_resources, industry, withdrawals	2	eps=0.4, minPts=8	2	0.6268690	0.3836774	1, 1, 1,...	6.000000	0.08695652	0.7915432	0.9272411	0.8593922
5	DBSCAN	water_productivity, industry	2	eps=0.4, minPts=3	2	0.6150827	0.3684267	0, 1, 1,...	6.769231	0.12173913	0.7830282	0.9303582	0.8566932
6	DBSCAN	water_productivity, industry	2	eps=0.4, minPts=4	2	0.6020309	0.3606021	0, 1, 1,...	6.615385	0.13913043	0.7735989	0.9319575	0.8527782
7	DBSCAN	water_productivity, industry	2	eps=0.4, minPts=5	2	0.6020309	0.3606021	0, 1, 1,...	6.615385	0.13913043	0.7735989	0.9319575	0.8527782
8	DBSCAN	available_resources, industry, withdrawals	3	eps=0.4, minPts=8	2	0.6071918	0.3901438	1, 1, 1,...	5.933333	0.09565217	0.7773274	0.9259195	0.8516234
9	DBSCAN	water_productivity, industry	2	eps=0.4, minPts=6	2	0.5930882	0.3582433	0, 1, 0,...	6.538462	0.14782609	0.7671383	0.9324396	0.8497890
10	DBSCAN	water_productivity, industry	2	eps=0.4, minPts=7	2	0.5930882	0.3582433	0, 1, 0,...	6.538462	0.14782609	0.7671383	0.9324396	0.8497890
11	DBSCAN	available_resources, industry, water_stress	3	eps=0.4, minPts=8	2	0.5852545	0.3762419	1, 1, 0,...	5.733333	0.12173913	0.7614788	0.9287609	0.8451198
12	DBSCAN	water_productivity, industry, withdrawals	2	eps=0.4, minPts=7	2	0.5596253	0.3114408	0, 1, 1,...	5.200000	0.19130435	0.7429629	0.9420056	0.8424843
13	DBSCAN	water_productivity, industry, withdrawals	2	eps=0.4, minPts=8	2	0.5596253	0.3114408	0, 1, 1,...	5.200000	0.19130435	0.7429629	0.9420056	0.8424843
14	DBSCAN	water_productivity, industry, withdrawals	2	eps=0.4, minPts=4	2	0.5688889	0.3564821	0, 1, 1,...	5.533333	0.14782609	0.7496554	0.9327996	0.8412275
15	DBSCAN	industry, agriculture	2	eps=0.4, minPts=7	2	0.5516598	0.3002723	0, 1, 1,...	6.307692	0.17391304	0.7372082	0.9442884	0.8407483

Showing 1 to 15 of 4,547 entries, 13 total columns

Figure - Top Clustering Results After Applying All Filters, by Composite Score

A total of 4,547 configurations remained after filtering from the original 148,736 results, applying constraints on cluster balance (MaxMinRatio < 10) and missing label rate (≤ 20%).

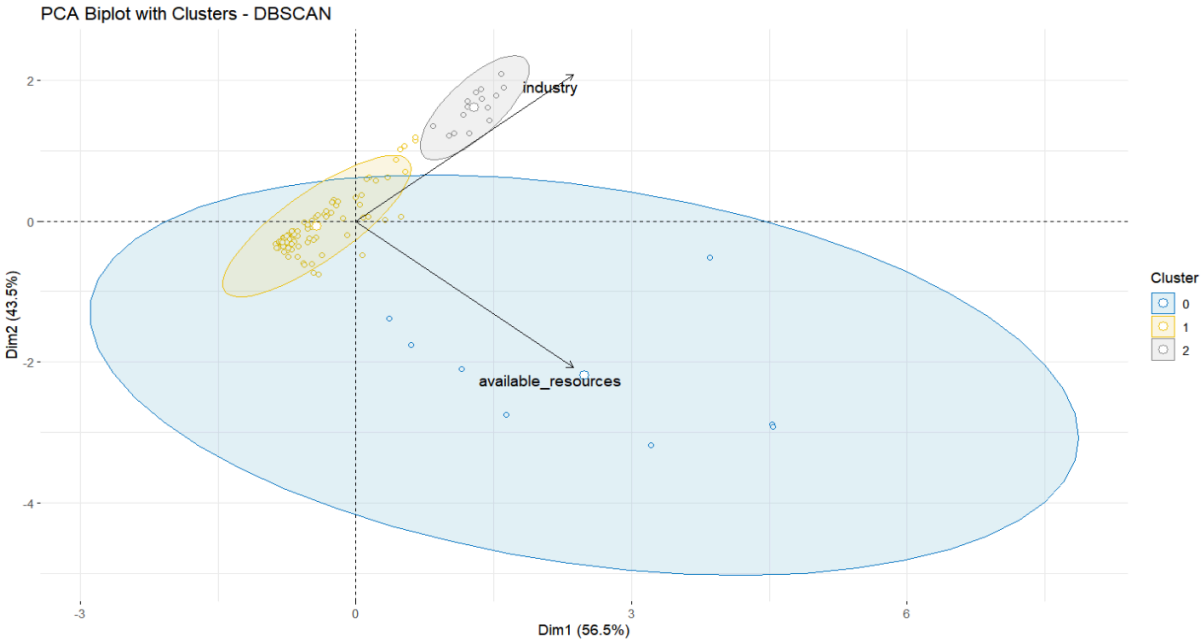
Notably, all top-ranked configurations based on the composite score were produced by DBSCAN. This outcome may be partly explained by its ability to detect arbitrarily shaped clusters and handle outliers — features that align well with the structure of the data.
As shown in the earlier section on Feature Selection & Normalization, country-level water indicators exhibit highly uneven distributions, non-convex groupings, and frequent regional outliers, reflecting the diversity of geographic, climatic, and economic conditions. These characteristics tend to favor density-based methods over centroid-based ones.
However, the dominance of DBSCAN is also shaped by evaluation constraints. For example, when the thresholds on MaxMinRatio and missing label rate were tightened to 5, the top configurations shifted toward K-means and hierarchical clustering. Therefore, while DBSCAN performed best under the applied conditions, its selection reflects both empirical results and structural compatibility with the data.
The highest-ranked configuration among the filtered results was selected for in-depth analysis and interpretation in the subsequent sections.

4.2. Evaluation and Visualization

```
> cat("Best Clustering Configuration:\n")
Best Clustering Configuration:
> cat("Silhouette Score:", round(best_result$Silhouette, 4), "\n")
Silhouette Score: 0.6665
> cat("DB Index:", round(best_result$DB_Index, 4), "\n")
DB Index: 0.3776
> cat("MaxMinRatio:", round(best_result$MaxMinRatio, 2), "\n")
MaxMinRatio: 6.13
> cat("Missing Label Rate:", round(best_result$MissingRate * 100, 2),
"%\n")
Missing Label Rate: 6.96 %
```

Code Snippet – Summary of the Best Clustering Configuration

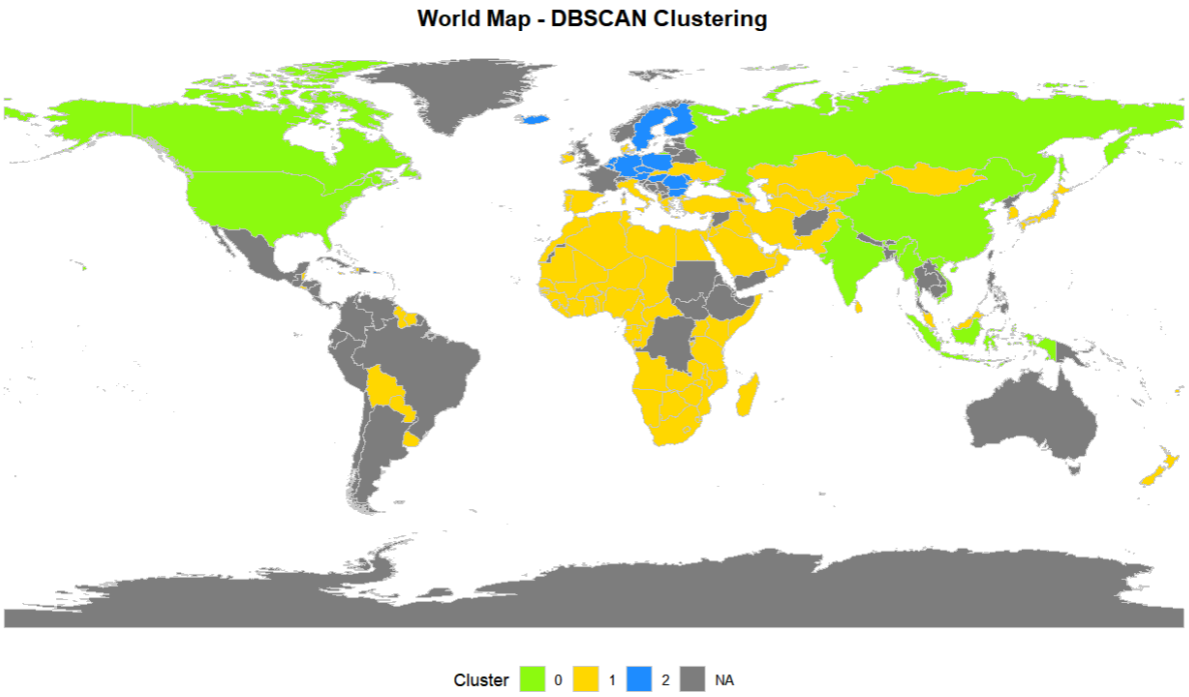
The best clustering configuration achieved a silhouette score of 0.6665, which indicates a moderate level of cohesion within clusters and separation between them. The DB index was 0.3776, a relatively low value that generally reflects compact and well-separated clusters. The MaxMinRatio was 6.13, meaning the largest cluster was about six times the size of the smallest; while this is within the threshold of 10, it suggests some level of imbalance. The missing label rate was 6.96%, indicating that a small portion of countries were not assigned to any cluster. Although all metrics fall within acceptable ranges, the imbalance in cluster sizes and the presence of unassigned data points may be worth considering when interpreting the results.



The figure shows the result of the best clustering configuration, projected onto the first two principal components (PC1 and PC2), which together explain 100% of the variance, as the analysis was conducted on two features: available water resources and industry. Each point represents a country, colored by its assigned cluster label.

- Cluster 1 (yellow) appears tightly grouped, indicating relatively strong internal cohesion among countries in this group.
- Cluster 2 (gray) forms a smaller but distinct group, positioned in the direction of higher values in the industry dimension.
- Cluster 0 (blue) consists of points labeled as noise by DBSCAN — countries that were not assigned to any cluster under the chosen parameter settings.

The plot provides a visual summary of how the two selected features contribute to differentiating countries into distinct groups under the chosen clustering method.
The final model, based on only available water resources and industry withdrawals, was not manually selected for interpretability, but rather emerged as the top-performing configuration after applying filtering criteria on cluster balance and label completeness.
Notably, this low-dimensional model also facilitated intuitive interpretation and effective 2D visualization, which was not consistently achievable with higher-dimensional alternatives. While more complex models were considered during the grid search, they often led to fragmented or less coherent cluster structures. In this sense, the final configuration represents a fortunate convergence of statistical performance and semantic clarity.
Additionally, although DBSCAN labeled some countries as noise (Cluster 0), their geopolitical and economic relevance warranted cautious interpretation. These observations were not treated as tightly grouped clusters but were still examined due to their potential policy implications.



In addition to the PCA projection, a geographic map of cluster assignments provides further insight into regional patterns.

- Cluster 1 (yellow) is concentrated in Sub-Saharan Africa, the Middle East, Central Asia, and parts of Latin America, possibly reflecting countries with similar agricultural reliance, infrastructure limitations, or climatic stress factors.

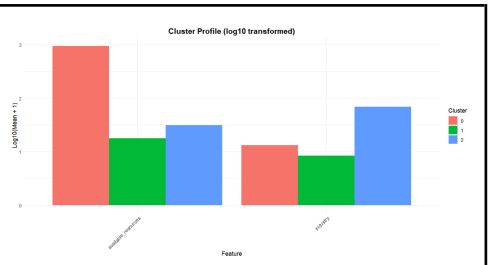
- Cluster 2 (blue) is mostly composed of Northern and Central European countries, forming a compact and contiguous region. This grouping could be associated with relatively high industrial development or consistent water governance practices.

- Cluster 0 (green) consists of countries labeled as noise by DBSCAN, including the United States, Canada, Russia, China, India, and several others across Southeast Asia and Eastern Europe. Although these countries were not assigned to any cluster under the current DBSCAN configuration, they will be individually discussed in the following section based on their feature characteristics.

- Countries shown in gray were previously excluded from clustering due to missing data. The missingness is notably concentrated in parts of Africa, Latin America, and Oceania, suggesting potential gaps in global data reporting or coverage.

While no assumptions about the exact drivers can be made from geography alone, this spatial distribution suggests that the clustering reflects more than random separation — and warrants further exploration through feature-based profiling in the following section.

4.3. Cluster Profiles

`> View(cluster_summary)`
`> cluster_summary`
`# A tibble: 3 × 3`
`  Cluster available_resources industry`
`  <fct>  <dbl>  <dbl>`
`1 0  1030.  25.7`
`2 1  38.9  12.4`
`3 2  46.2  68.9`
	

Average feature values by cluster(Table)	Cluster Profiles (Log-Scaled, Barplot)

The table and bar plot above shows the average values of the two selected features — Available Water Resources and Industry Withdrawals — for each cluster. These profiles provide insight into the distinct characteristics that define each group:

Cluster	Characteristics
Cluster1	Characterized by low available water resources (38.9) and low industrial use (12.4), this cluster likely includes countries with limited water supply and relatively low levels of industrialization. Many of these countries are located in Sub-Saharan Africa, the Middle East, and parts of Central and South Asia, suggesting environments where water scarcity and underdeveloped infrastructure may play important roles in shaping usage patterns.
Cluster2	This group is defined by moderate available resources (46.2) and the highest industrial use (68.9) among all clusters. The profile suggests countries with limited natural water availability but strong industrial activity, potentially supported by efficient management or heavy reliance on imports and infrastructure. These countries are mainly located in Europe, possibly reflecting developed economies with structured industrial sectors and controlled water use.
Cluster0	This group consists of countries originally labeled as noise by DBSCAN, including several large and globally influential states. Although not tightly clustered, they display relatively high water availability (1030.0) and moderate industrial use (25.7), suggesting a loosely shared profile. Given their economic and geopolitical weight, they are included here for reference despite their exclusion from the formal clustering structure.

This interpretation provides contextual understanding of the clustering outcome and serves as a foundation for potential policy recommendations tailored to the water resource profiles of each cluster.
However, it is important to acknowledge that while clustering reveals consistent patterns, it does not explain the underlying causes. As an unsupervised method, it identifies what exists, not why it exists. Accordingly, interpreting clusters as policy groups assumes internal homogeneity that may not fully hold.
Policy suggestions derived from cluster membership should therefore be treated cautiously, especially in the absence of causal inference and control for unobserved factors.

5. Discussion on Policy Implications

Drawing from the broad patterns observed across clusters, several exploratory policy considerations can be outlined.
While these groupings provide a useful lens for comparative analysis, any policy application must be context-sensitive and account for country-specific conditions beyond the scope of this clustering exercise.

[Cluster 1: Low Resource, Low Industrial Demand]

This group consists of countries with generally limited water availability and relatively low industrial withdrawals. While specific national contexts vary, the combination suggests potential vulnerabilities in both water access and economic capacity related to industrial use.

Tentative policy directions may include:

- Improving access to basic water infrastructure.
- Strengthening local water resilience planning.
- Aligning future industrial development with available water resources.

[Cluster 2: High Industrial Use with Limited Resources]

Countries in this group exhibit relatively high industrial water demand despite having only moderate to limited water availability. This configuration may reflect effective infrastructure and management, but also indicates a potential imbalance between usage and long-term sustainability.

Policy considerations may include:

- Enhancing water use monitoring in industrial areas.
- Supporting circular water practices such as recycling and reuse.
- Reviewing allocation frameworks to balance industrial needs with ecological and social priorities.

[Cluster 0: Resource-Rich with Moderate Industrial Activity (Unclustered Group)]

This group includes countries that were not formally clustered but share a general pattern of abundant water resources and moderate industrial withdrawals. A limited interpretation is offered here as a reference.
Possible considerations — highly dependent on country-specific contexts — include:

- Preserving resource security in anticipation of future industrial or demographic shifts.
- Promoting efficient technologies to avoid complacency in water-abundant systems.
- Strengthening monitoring in resource-intensive sectors to safeguard long-term sustainability.

Overall, this clustering framework highlights how data-driven groupings can guide region-specific policy interventions, ensuring that water governance aligns with actual usage patterns and resource constraints. However, these insights are exploratory rather than prescriptive and should be tailored to national-level contexts, given the absence of causal inference and the influence of unobserved political, economic, and geographic factors.

6. Conclusion

This study applied unsupervised learning techniques to identify meaningful clusters among countries based on water-related indicators, focusing on available water resources and industrial withdrawals. Principal Component Analysis (PCA) was used for dimensionality reduction, and an extensive grid search explored various clustering algorithms, feature subsets, and parameter combinations.

DBSCAN yielded the highest-performing configuration under the defined evaluation criteria, which combined silhouette score and Davies–Bouldin index into a composite metric. Additional constraints on cluster balance and label coverage were applied during post-processing to ensure interpretability and robustness. However, this outcome was shaped by the chosen thresholds and scoring design, and alternative methods such as K-means or hierarchical clustering may perform better under different assumptions.

The final model revealed three distinct groups, each with its own water-use profile. While one of these groups consisted of countries initially labeled as noise by DBSCAN, it was retained for reference due to its geopolitical and economic importance.

Although the study explored potential policy implications associated with the observed patterns, such recommendations remain exploratory in nature. Clustering reveals what patterns exist, not why they occur. Therefore, any use of these groupings for policy design must be approached with caution, tailored to national-level contexts, and supplemented by more detailed causal and institutional analysis.

Overall, the study demonstrates how a rigorous unsupervised learning framework — when properly validated and interpreted — can uncover latent structure in complex global datasets and inform further inquiry into context-specific water management strategies.

Time Series Forecasting of Freshwater Withdrawals in Germany Using ARIMA and Prophet

1. Introduction

This section aims to forecast freshwater withdrawals in Germany using time series analysis. Freshwater withdrawals are a key indicator of national water demand and are closely linked to sustainability metrics such as GDP and water stress. Two forecasting models are employed: ARIMA, a classical statistical method, and Prophet, a modern additive time series model developed by Facebook. The goal is to evaluate their performance and generate reliable forecasts.

2. R Libraries used

Library	Purpose
forecast	Classical ARIMA modeling
prophet	Decomposable time series forecasting with automatic changepoint detection
tidyverse	Data wrangling and visualization
lubridate	Date parsing and manipulation
tseries	Time series stationarity testing (e.g., Augmented Dickey-Fuller Test)

3. Variable and Country Selection Rationale

3.1. Target Variable: Freshwater Withdrawals

Freshwater withdrawals were chosen as the target variable because they represent total water demand across all economic sectors. This variable is directly involved in the calculation of other key indicators: GDP (USD) is computed as the product of water productivity and withdrawals, while water stress is derived from the ratio of withdrawals to renewable water resources, multiplied by 100. These mathematical relationships highlight the centrality of withdrawals in understanding both economic performance and environmental pressure. Forecasting this variable therefore supports broader insights into sustainability and development trends.

3.2. Target Country: Germany

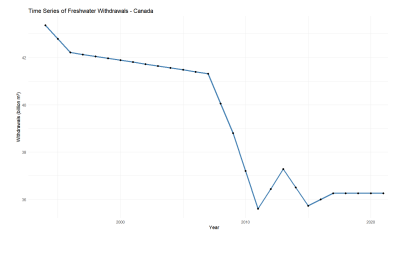
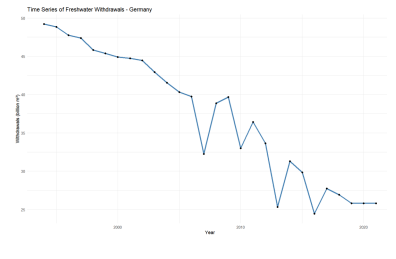
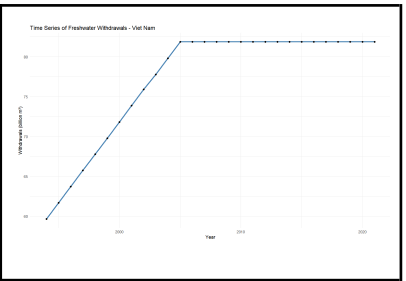
		
Canada(“CAN”)	Germany(“DEU”)	Vietnam(“VNM”)

Figure 1. Time Series of Freshwater Withdrawals of various countries

Germany was selected after reviewing the freshwater withdrawal patterns of numerous countries. Its time series exhibits a clear and consistent long-term downward trend, with moderate year-to-year variability that provides structure without excessive noise. Also, there are no long flat segments, which often limit model learning, and the values remain within a stable, realistic range, avoiding outliers or sudden regime shifts. In contrast, many countries showed less favorable patterns—some, like Vietnam, had time series that became completely flat in recent years, offering no meaningful variation to model. Others, such as Canada, displayed irregular drops and prolonged stagnation, which reduced forecastability. Therefore, Germany proved to be the most suitable candidate for time series forecasting.

4. Forecasting Framework

As previously mentioned, both the AutoRegressive Integrated Moving Average (ARIMA) model and the Facebook Prophet model were applied to model the historical freshwater withdrawals in Germany. Conceptual explanations of these models are provided in Appendix D.

4.1. ARIMA Forecasting

The forecasting procedure followed a two-step approach. First, the model was trained on the full available time span to produce a five-year forecast for future withdrawals. This was used primarily for visualization and exploratory insight. Then, two separate evaluation methods were applied to assess model performance. The first was a traditional holdout approach: the dataset was split into a training period (up to 2017) and a test period (2018–2021). A new ARIMA model was fitted on the training data, and its predictive accuracy was evaluated on the test set using MAE, RMSE, and MAPE. The second method was a rolling forecast evaluation, in which the model was repeatedly trained on an expanding window and used to predict the following year. This rolling process, spanning from 2003 to 2020, enabled a more comprehensive view of the model’s generalization over time.

```r
# Convert to time series object
ts_germany <- ts(df$Value, start = min(df$Year), frequency = 1)
# frequency = 1 means yearly data (as opposed to monthly = 12, quarterly = 4).

# Estimate lambda for Box-Cox transformation
lambda <- BoxCox.lambda(ts_germany)
cat("Suggested Box-Cox lambda:", lambda, "\n")

# Check for stationarity using Augmented Dickey-Fuller Test
adf_test <- tseries::adf.test(ts_germany, alternative = "stationary")
print(adf_test)

# Plot ACF and PACF to visually inspect autocorrelation
acf(ts_germany, main = "ACF - Germany Withdrawals")
pacf(ts_germany, main = "PACF - Germany Withdrawals")

# Fit ARIMA model: This uses the auto.arima() function to automatically determine
# the best-fitting ARIMA model for the time series.
arima_model <- auto.arima(ts_germany)
```

Figure 2. Code snippet – ARIMA model design and forecasting logic.

The time series was explicitly defined with frequency = 1 to indicate annual observations, distinguishing it from monthly or quarterly data where seasonal components would be more relevant. The auto.arima() function from the forecast package was used to automatically select the best-fitting ARIMA model by minimizing the corrected Akaike Information Criterion (AICc). This process identifies the optimal combination of autoregressive (AR), differencing (I), and moving average (MA) terms without requiring manual specification.

```
> lambda <- BoxCox.lambda(ts_germany)
> cat("Suggested Box-Cox lambda:", lambda, "\n")
Suggested Box-Cox lambda: 1.999924
```

Figure 3. Code snippet - Box-Cox lambda estimation for variance stability

Before the analysis was conducted, three diagnostic tests—the Box-Cox lambda estimation for variance stability, the Augmented Dickey-Fuller (ADF) test for stationarity, and the autocorrelation function (ACF/PACF) plots for autocorrelation structure —were performed to verify the assumptions required for ARIMA modeling. First, the Box-Cox lambda was estimated to evaluate whether a variance-stabilizing transformation was necessary. The result indicated a lambda of approximately 2.0, suggesting a square transformation. However, since the data exhibited no clear signs of heteroscedasticity and interpretability would be compromised, no transformation was applied.

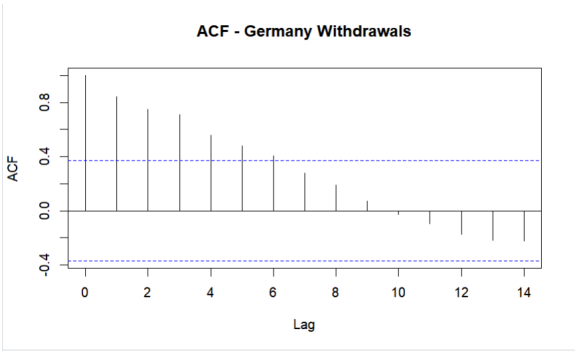
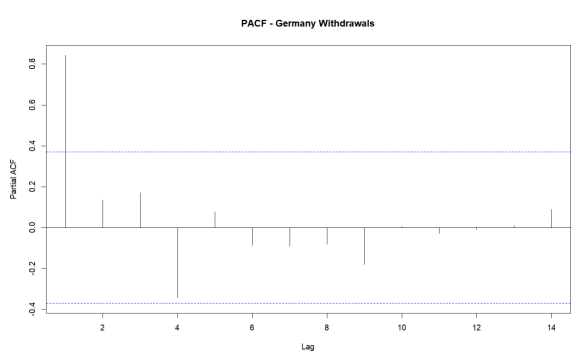
```
> adf_test <- tseries::adf.test(ts_germany, alternative = "stationary")
> print(adf_test)

        Augmented Dickey-Fuller Test

data:  ts_germany
Dickey-Fuller = -2.6909, Lag order = 3, p-value = 0.3082
alternative hypothesis: stationary
```

Figure 4. Code snippet - Augmented Dickey-Fuller (ADF) stationarity test applied to the original time series

ARIMA models assume that the underlying time series is stationary, meaning its statistical properties (mean, variance, autocorrelation) do not change over time. The resulting p-value of 0.3082 indicated non-stationarity. However, since the auto.arima() function automatically identifies and applies the appropriate differencing required for stationarity, no manual differencing was performed.

	
ACF plot of Germany's freshwater withdrawals.	PACF plot of Germany's freshwater withdrawals.

The autocorrelation function (ACF) plot shows significant positive autocorrelations at multiple lags, particularly from lag 1 to lag 6, all exceeding the 95% confidence bounds (blue dashed lines). This pattern indicates strong temporal dependence, suggesting that past values heavily influence future values. The gradual decline is characteristic of a non-stationary series, supporting the need for differencing before ARIMA modeling(which will be done by auto.arima()).
The partial autocorrelation function (PACF) plot shows a strong spike at lag 1, followed by a sharp cutoff with all values within the confidence bounds. This pattern is typical of an autoregressive process of order 1 (AR(1)), suggesting that only the immediate previous value contributes significantly to the current value when controlling for intermediate lags. This structure further supports the use of an ARIMA model with an autoregressive component.

4.2 Prophet Forecasting

As with ARIMA, Prophet was first fitted on the full dataset to produce a five-year forecast for visualization. Then, two evaluation methods were applied: one model was trained on data up to 2017 and tested on the 2018–2021 period using MAE, RMSE, and MAPE; another was evaluated using a rolling forecast approach, where the model was repeatedly retrained on expanding windows to forecast the following year. This consistent two-step approach enabled a direct and fair comparison between the Prophet and ARIMA models across both static and dynamic evaluation settings.

```r
# Prepare data for Prophet
prophet_df <- df %>%
  rename(ds = Year, y = Value) %>%
  mutate(ds = ymd(paste0(ds, "-01-01")))

# Fit Prophet model with yearly seasonality disabled (not needed for annual data)
prophet_model <- prophet(prophet_df, yearly.seasonality = FALSE)

# Create future dates (5 years ahead)
future <- make_future_dataframe(prophet_model, periods = 5, freq = "year")

# Forecast future withdrawals
forecast_prophet <- predict(prophet_model, future)
```

Figure 5. Code snippet – Prophet model setup and forecasting logic.

Prophet is an additive time series model that decomposes the data into trend, seasonality, and holiday effects, and is particularly effective for handling irregular trends and missing values. In this case, only the trend component was modeled, as the data is annual and does not exhibit seasonal patterns. The input data was formatted to match Prophet's required structure, with the time variable renamed to ds and the target variable to y.

5. Forecasting Results and Evaluation

5.1. ARIMA Forecasting

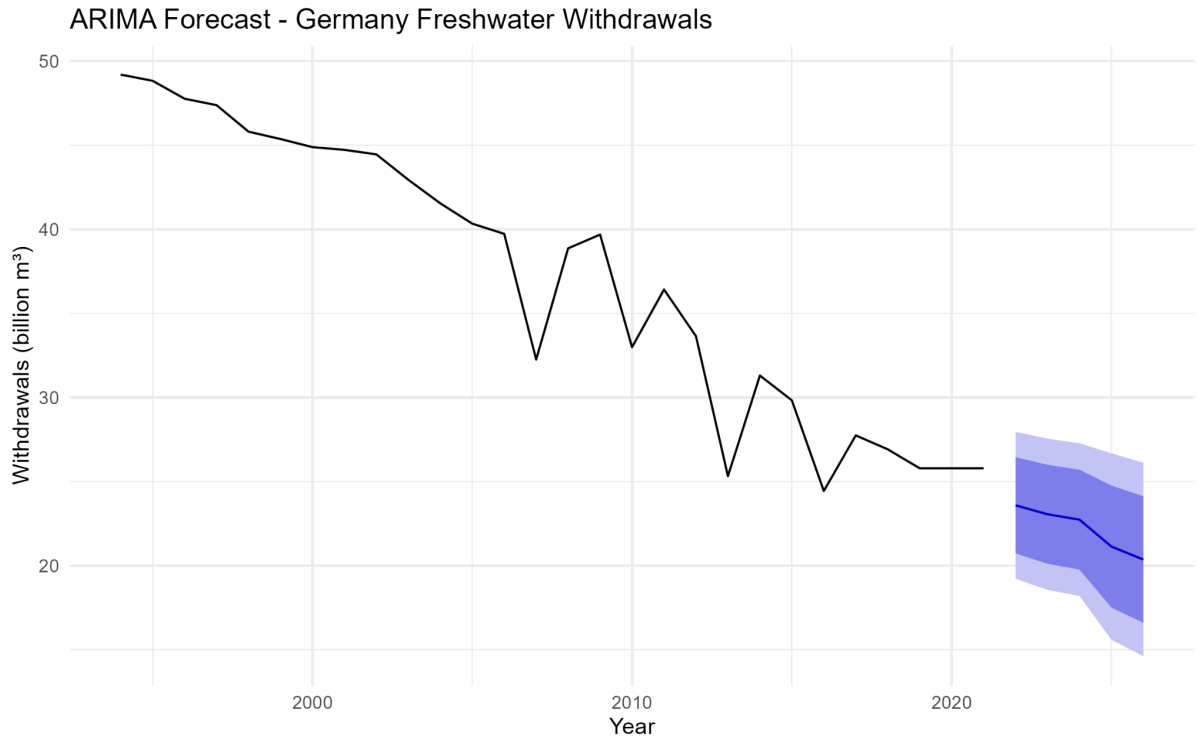


Figure 6. ARIMA Forecast of Germany's Freshwater Withdrawals (2022–2026)

This plot illustrates the five-year forecast generated by the ARIMA model trained on the full historical dataset. The black line represents the observed data up to 2021, while the blue line extends the forecasted trend through 2026. The shaded areas denote the 80% and 95%

confidence intervals, capturing increasing uncertainty further into the forecast horizon. The model projects a continued decline in freshwater withdrawals, consistent with the long-term trend observed in the past data.

	A	B
1	Year	Forecast
2	2022	23.59159
3	2023	23.05934
4	2024	22.73899
5	2025	21.13809
6	2026	20.36587

Figure 7. Forecasted Values from the ARIMA Model (2022–2026)

This table lists the exact forecasted values corresponding to the ARIMA plot shown earlier, which provides a numerical reference. Together with the visual forecast in Figure 4, these results offer a practical basis for future decision-making and water resource policy planning.

```
> summary(arima_model)
Series: ts_germany
ARIMA(2,1,0) with drift

Coefficients:
          ar1      ar2    drift
      -0.7582  -0.6711  -0.9060
s.e.   0.1335   0.1259   0.1721

sigma^2 = 4.968:  log likelihood = -59.08
AIC=126.15   AICc=127.97   BIC=131.34

Training set error measures:
                     ME     RMSE      MAE        MPE     MAPE      MASE      ACF1
Training set 0.01225276 2.063632 1.283608 -0.2315765 3.962897 0.5439712 0.114933
```

Figure 8. ARIMA Model Summary

The final model selected by auto.arima() was an ARIMA(2,1,0) with drift, indicating that the time series was differenced once and modeled using two autoregressive terms. This aligns with the results of the stationarity test, which confirmed the need for differencing. While the PACF plot of the original series suggested a strong AR(1) structure, the automated model selection process identified that including a second autoregressive term improved the overall model fit based on information criteria - possibly due to residual autocorrelation not fully captured by a single lag. The inclusion of drift captures the underlying trend in the differenced series.

All estimated coefficients, including the drift term (−0.9060), were statistically significant, as their absolute values exceeded approximately twice their respective standard errors—a common rule of thumb when formal p-values are not reported. The negative drift reflects a consistent downward trend in the data.

Model selection was based on the corrected Akaike Information Criterion (AICc), which is the default criterion. The selected model yielded AIC = 126.15, AICc = 127.97, and BIC = 131.34. On the full dataset, the model achieved a MAPE of 3.96%, indicating a strong in-sample fit. The residual autocorrelation at lag 1 (ACF1 = 0.11) was low, suggesting that

most temporal patterns were successfully captured. While MAPE is a widely used metric, it can be sensitive to low actual values; however, in this case, the scale and stability of the data mitigate that concern.

```
> checkresiduals(arima_model)  # Includes Ljung-Box test, residual ACF, histogram

        Ljung-Box test

data:  Residuals from ARIMA(2,1,0) with drift
Q* = 6.8181, df = 4, p-value = 0.1458

Model df: 2.   Total lags used: 6
```

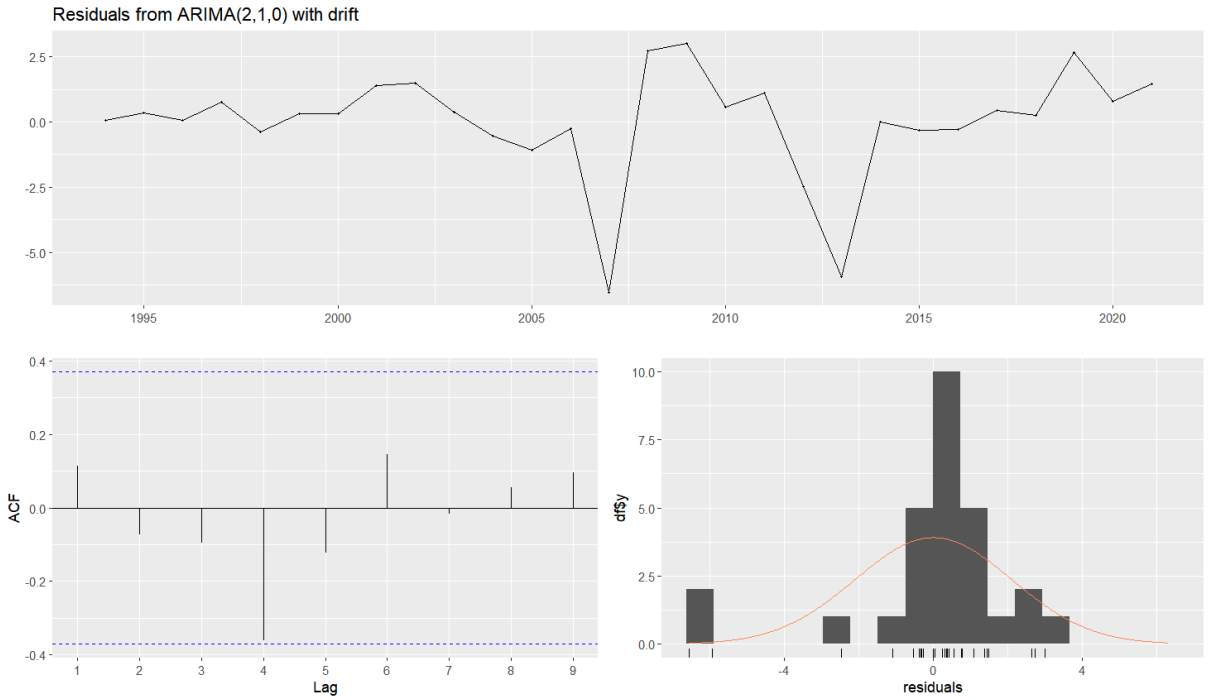


Figure 9, 10. Residual diagnostics for the ARIMA(2,1,0) with drift model

The residual diagnostics show no major violations of model assumptions. In the time series plot (top panel), the residuals appear randomly scattered around zero, with no visible trend or seasonality—suggesting no systematic error remains. In the autocorrelation function (ACF) plot, all lags fall within the 95% confidence bounds, which is a common threshold for determining statistical significance. The histogram of residuals is roughly bell-shaped, supporting the assumption of normality, although slight skewness is observed. Most importantly, the Ljung-Box test returned a p-value of 0.1458 (above the 0.05 significance level), indicating that there is insufficient evidence of residual autocorrelation. Based on these criteria, the model is considered a statistically valid fit for the data.

5.2. Prophet Forecasting

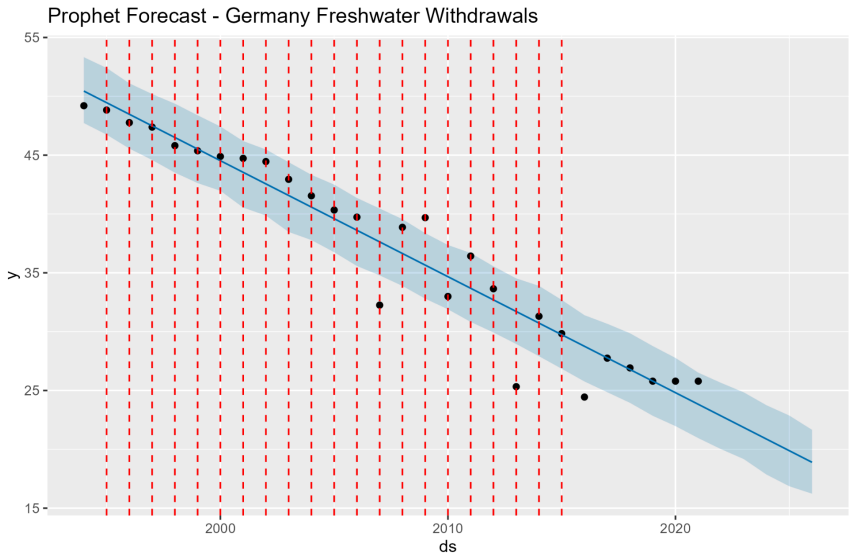


Figure 11. Prophet Forecast of Germany's Freshwater Withdrawals (2022–2026)

This plot presents the five-year forecast generated by the Prophet model trained on the full dataset. The blue line represents the model's predicted trend, while the shaded region shows the forecast uncertainty interval. The black dots correspond to the actual historical observations, most of which fall within the confidence band—indicating a good model fit. The vertical red dashed lines mark the changepoints automatically detected by Prophet, where the trend is allowed to shift. Although no abrupt structural breaks are visible in the series, these changepoints allow the model to adapt flexibly to subtle changes in trend. Overall, Prophet forecasts a continued decline in freshwater withdrawals, consistent with the observed long-term downward trajectory.

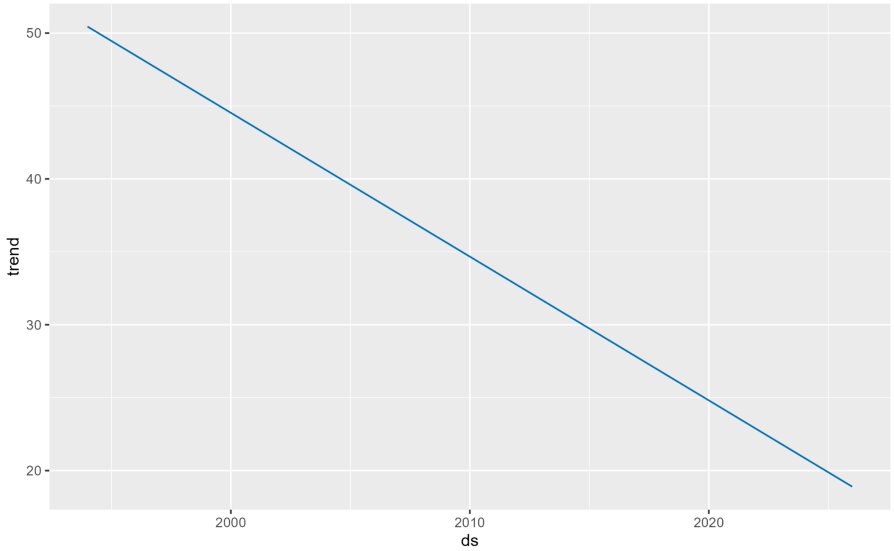


Figure 12. Prophet Forecast Components – Germany's Freshwater Withdrawals (2022–2026)

Note: Since the seasonality and holiday effect are absent, the line in the graph is identical to that of *Figure 7*.

#######	22.84169	20.01963	25.64703
#######	21.85656	19.17608	24.84236
#######	20.87142	17.8599	23.74378
#######	19.88358	16.86555	22.86875
#######	18.89845	16.23397	21.65674

Figure 13. Forecasted and Fitted Values from the Prophet Model

This table contains the Prophet model's output across the full time span. The yhat column contains fitted values for historical years (1994–2021) and forecasted values for future years (2022–2026), with yhat_lower and yhat_upper indicating uncertainty bounds. (Only the five forecasted values are displayed in the figure due to image size concerns.)

5.3. Model Performance Evaluation and Comparison

As previously mentioned, both ARIMA and Prophet were first trained on data up to 2017 and evaluated on a held-out test set spanning 2018 to 2021 to compare model performance. Forecast accuracy was measured using three standard metrics: MAE, RMSE, and MAPE. The results for each model are summarized below.

`> cat("ARIMA Metrics:\n")`
`ARIMA Metrics:`
`> cat("MAE: ", MAE, "\nRMSE:", RMSE, "\nMAPE:", MAPE, "%\n")`
`MAE:  2.011173`
`RMSE: 2.262085`
`MAPE: 7.778954 %`
	`> cat("Prophet Metrics:\n")`
	`Prophet Metrics:`
	`> cat("MAE: ", MAE, "\nRMSE:", RMSE, "\nMAPE:", MAPE, "%\n")`
	`MAE:  1.457615`
	`RMSE: 1.676682`
	`MAPE: 5.620193 %`

ARIMA	Prophet

Metric	Description
MAE (Mean Absolute Error)	Average magnitude of the forecast errors, regardless of direction. Simple and easy to interpret. However, does not penalize large errors more than small ones.
RMSE (Root Mean Squared Error)	Heavily penalizes large errors, making it useful when large deviations are especially undesirable. However, less interpretable than MAE due to squaring and square-rooting.
MAPE (Mean Absolute Percentage Error)	Average absolute error expressed as a percentage of the actual values. Intuitive and unit-free, making it easy to compare across different scales. However, can be misleading when actual values are very small or zero (division by small numbers inflates error).

On the held-out test set (2018–2021), the Prophet model slightly outperformed ARIMA across all three error metrics. Prophet achieved a lower MAE (1.46 vs. 2.01), RMSE (1.68 vs. 2.26), and MAPE (5.62% vs. 7.78%). While these results might suggest that Prophet provides a better in-sample fit on the test window, model performance can vary depending on the time segment evaluated. Therefore, to obtain a more robust and generalizable assessment, we proceeded with a rolling forecast evaluation over a longer time horizon.

```
# Define rolling parameters
start_year <- 2003
end_year <- 2020
horizon <- 1        # Forecast 1 year ahead in each loop
```

Figure 14. Rolling forecast parameter settings used for evaluation

The rolling forecast was configured with a start year of 2003 and an end year of 2020, using a 1-year forecast horizon. This means the model was repeatedly trained on data up to each year from 2003 to 2020 and used to predict the value for the following year. This setup produced 18 evaluation rounds, enabling a more reliable estimate of model performance over time.

`> cat("ARIMA Rolling Forecast Metrics:\n")`
`ARIMA Rolling Forecast Metrics:`
`> cat("Mean MAE: ", mean(rolling_mae), "\n")`
`Mean MAE:  2.726816`
`> cat("Mean RMSE:", mean(rolling_rmse), "\n")`
`Mean RMSE: 2.726816`
`> cat("Mean MAPE:", mean(rolling_mape), "%\n")`
`Mean MAPE: 8.651565 %`
	`> # Print average results`
	`> cat("Prophet Rolling Forecast Metrics:\n")`
	`Prophet Rolling Forecast Metrics:`
	`> cat("Mean MAE: ", mean(rolling_mae_prophet), "\n")`
	`Mean MAE:  2.160427`
	`> cat("Mean RMSE:", mean(rolling_rmse_prophet), "\n")`
	`Mean RMSE: 2.160427`
	`> cat("Mean MAPE:", mean(rolling_mape_prophet), "%\n")`
	`Mean MAPE: 7.199758 %`

ARIMA	Prophet

The results from the rolling forecast evaluation indicate that Prophet achieved slightly lower average error metrics compared to ARIMA across all measures. Specifically, Prophet recorded a mean MAE of 2.16 and a mean MAPE of 7.20%, while ARIMA showed higher errors, with a mean MAE of 2.73 and MAPE of 8.65%. The RMSE values were equal to the MAEs in both models, likely due to the relatively stable error distribution.
This pattern aligns with the earlier held-out test results, where Prophet also showed a modest advantage over ARIMA. A possible reason for this outcome—at least in the context of the present dataset—is Prophet's use of piecewise linear trend fitting and automatic changepoint detection, which may have helped it adapt more flexibly to subtle structural changes in the declining trend. This characteristic could have contributed to its slightly improved accuracy in this particular case. However, it should be noted that in cases where the underlying time series exhibits stable linear trends without significant structural changes,

ARIMA may provide more robust and interpretable results than Prophet, due to its reliance on well-established statistical assumptions and simpler model structure.
While the overall differences in performance are not dramatic, both models produced relatively low average error rates, suggesting that they are reasonably effective forecasters for this type of national-level water withdrawal data. That said, the slightly lower and more consistent errors observed with Prophet across both evaluation settings provide tentative support for its use in similar long-term forecasting tasks.

6. Interpretation of Forecast Results and Conditional Policy Recommendations

The forecasting results for Germany's freshwater withdrawals, generated by both ARIMA and Prophet models, suggest a continued downward trend in the near future. By 2026, ARIMA projects withdrawals to decline to approximately 20.4 billion m^3, while Prophet offers a similar estimate of 18.9 billion m^3 — both down from roughly 26 billion m^3 in 2021. While the models demonstrate strong statistical performance, it is important to recognize that such forecasts may reflect a range of underlying factors. A declining withdrawal trend could indicate positive developments, such as improved water-use efficiency or structural economic shifts, but it might also result from less favorable conditions, including economic downturns or climate-related constraints. Interpreting these results in context is therefore essential, particularly when deriving policy recommendations.

6.1. Potential Drivers Behind the Forecasted Withdrawal Trend

Several plausible scenarios may explain the observed and predicted reduction in withdrawals:

Cause Type	Possible Explanation
Efficiency Gains	Improved irrigation systems, industrial reuse, household conservation.
Economic Restructuring	Shift toward less water-intensive sectors such as services.
Economic Decline	Contraction in manufacturing or agriculture reducing demand.
Climate Impact	Droughts or groundwater depletion limiting actual withdrawal capacity.
Data Artefacts	Changes in monitoring methods or definitions over time.

Each scenario carries very different policy implications. For example, efficiency gains may justify infrastructure optimization, whereas climate-driven reductions may require emergency preparedness and adaptation investment.
To clarify which possibilities may be driving the observed trend, further analysis should be conducted using exploratory data analysis (EDA), linear regression, and potentially classification techniques. These methods help uncover patterns and predictors that explain the trajectory of withdrawals more clearly and inform targeted policy interventions.

6.2. Scenario-Based Policy Recommendations

The table below outlines appropriate responses based on the dominant driver of the withdrawal trend:

Scenario	Policy Direction
Efficiency improvements	Expand support for innovation, precision agriculture, smart water systems.
Sectoral shift	Align infrastructure plans with new economic structures and long-term demand.
Economic downturn	Avoid irreversible cuts in supply capacity; adopt flexible, modular planning.
Climate-related decline	Strengthen drought resilience, water storage, and non-traditional supply methods.
Measurement changes	Improve data transparency, harmonize indicators, and monitor consistency.

These recommendations are not mutually exclusive; multiple causes may operate simultaneously. Therefore, it is desirable that Policymakers adopt adaptive frameworks that can respond dynamically as new information emerges.

7. Conclusion

This study has presented a time series analysis of Germany's freshwater withdrawals using two forecasting approaches: ARIMA and Prophet. In both the held-out test and rolling forecast evaluations, Prophet slightly outperformed ARIMA across all error metrics. This pattern, observed in this specific dataset, may suggest Prophet's advantage in capturing non-linear trends and adapting to structural changes, particularly through its automatic changepoint detection. While the difference in performance was modest, both models proved to be suitable for forecasting long-term national-level water usage trends.

Both models projected a continued downward trend in withdrawals through 2026. While these projections are statistically reliable, their real-world interpretation requires caution. A declining trend may reflect favorable developments such as improved efficiency or structural economic shifts, but it could also signal more concerning drivers such as economic slowdown or climate-related constraints. The time series models capture the "what," but not the "why."

To address this limitation, further analysis should be conducted using exploratory data analysis, regression analysis, and potentially classification techniques. These tools will help clarify the underlying causes of withdrawal changes and provide a more nuanced understanding of the observed patterns.

Ultimately, this study lays the foundation for a broader research framework that combines accurate forecasting with causal interpretation. By integrating time series forecasting with complementary analytical methods, this research aims to support more informed, adaptive,

and sustainable water resource planning — both within Germany and in other countries facing similar water management challenges.

Data Sources, References, and R scripts

Data Sources:

- Water Productivity (GDP per unit of water withdrawal)
  https://data.worldbank.org/indicator/ER.GDP.FWTL.M3.KD
- Water Stress (withdrawals as % of renewable resources)
  https://data.worldbank.org/indicator/ER.H2O.FWST.ZS
- Total Freshwater Withdrawals (billion m^3/year)
  https://data.worldbank.org/indicator/ER.H2O.FWTL.K3
- Agricultural Withdrawal (% of total withdrawals)
  https://data.worldbank.org/indicator/ER.H2O.FWAG.ZS
- Domestic Withdrawal (% of total withdrawals)
  https://data.worldbank.org/indicator/ER.H2O.FWDM.ZS
- Industrial Withdrawal (% of total withdrawals)
  https://data.worldbank.org/indicator/ER.H2O.FWIN.ZS
- Population (total)
  https://data.worldbank.org/indicator/SP.POP.TOTL
- GDP per Capita (PPP, current international $)
  https://data.worldbank.org/indicator/NY.GDP.PCAP.PP.CD
- Private Investment in Water & Sanitation (current US$)
  https://data.worldbank.org/indicator/IE.PPI.WATR.CD
- Natural Disaster Impact (% of population; avg. 1990–2009)
  https://data.worldbank.org/indicator/EN.CLC.MDAT.ZS
- Precipitation (average annual depth, mm)
  https://data.worldbank.org/indicator/AG.LND.PRCP.MM

References:

- Clere A., & Bansal V. (2022) *Machine Learning with Dynamics 365 and Power Platform : The Ultimate Guide to Apply Predictive Analytics*. Wiley.

- Rocha, A. M. A. C., Murgante, B., Garau, C., Gervasi, O., & Misra, S. (Eds.). (2022). *Computational science and its applications – ICCSA 2022 workshops*. Springer International Publishing.

- Prevos P. (2023) * Data Science for Water Utilities : Data as a Source of Value*. RC Press

- van Delden, A., Snijkers, G., Jones, J., Sakshaug, J. W., Thompson, K. J., Bavdaž, M., Bender, S., & MacFeely, S. (2022). *Advances in business statistics, methods and data collection*. Wiley.

- Ertz, F., Burgard, J. P., & Münnich, R. (2024). Lecture notes for the course Statistical Programming with R. Trier University.

- Münnich, R., Burgard, J. P., & Ertz, F. (2024). Lecture notes for the course Elements of Statistics. Trier University.

- Krause, J. (2025). Lecture notes for the course Statistical Methods of Data Science. Trier University.

- Bergmann, R. (2025). Lecture notes for the course Data Mining. Trier University.

R scripts:

- Data Preprocessing
  https://github.com/1798bebe/Statistical-Research-with-R/blob/main/preprocessing/preprocessing.R

- Supervised Learning
  https://github.com/1798bebe/Statistical-Research-with-R/blob/main/Supervised%20Learning%20%28Regression%2C%20Classification%29/binary%20classification.R

- Unsupervised Learning
  https://github.com/1798bebe/Statistical-Research-with-R/blob/main/unsupervised%20learning(PCA%2C%20clustering)/unsupervised_learning.R

- Time-series Forecasting
  https://github.com/1798bebe/Statistical-Research-with-R/blob/main/time%20series%20forecasting/time_series_forecasting.R

Appendix

Model Overview

1 Classification Model Overview

1.1 Random Forest (RF)

Random Forest (RF) is an ensemble learning algorithm used primarily for classification and regression tasks. It creates a multitude of decision trees during the training phase, combining their individual predictions to improve accuracy and reduce overfitting.

1.1.1 Mathematical Formulation

Given a training dataset, Random Forest builds K decision trees, each trained independently on a randomly sampled subset of the data (with replacement), using random subsets of features at each split. The final prediction is made by aggregating the predictions of all individual trees:

For classification, the final predicted class $\hat{y}$ is the majority vote across all trees:

$$\hat{y} = \text{majority\_vote}\left(T_1(x), T_2(x), \ldots, T_K(x)\right)$$

For regression, the final predicted value is the average of the predictions:

$$\hat{y} = \frac{1}{K} \sum_{k=1}^{K} T_k(x)$$

Where:

- x: Input feature vector.
- K: Total number of decision trees in the forest.
- $T_k(x)$: Prediction from the k-th decision tree.
- $\hat{y}$: Final aggregated prediction.

1.1.2 Key Components of the Algorithm

- **Bootstrap Aggregating (Bagging)**: Each decision tree is trained on a random subset (sampled with replacement) from the original training data, ensuring diversity among trees.
- **Random Feature Selection**: For each split in a tree, a random subset of features is chosen. This reduces correlation between trees and leads to better generalization.
- **Aggregation of Predictions**: The individual tree predictions are combined by majority voting (classification) or averaging (regression).

1.1.3 Advantages and Characteristics

- Robustness against overfitting due to ensemble averaging.
- High predictive accuracy by aggregating multiple trees.
- Easy interpretation of feature importance.
- Capability to handle large datasets and numerous features effectively.
- Effective handling of missing values and noisy data.

1.2 XGBoost (Extreme Gradient Boosting)

XGBoost is an optimized gradient boosting algorithm designed for supervised learning tasks, including classification and regression. It builds an ensemble of decision trees sequentially, where each new tree focuses on correcting errors made by previous trees, significantly enhancing predictive performance.

1.2.1 Mathematical Formulation

XGBoost minimizes the following objective function:

$$\text{Obj}(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t)$$

This objective consists of two components:

1. **Loss Function (Fit Term)**:

   $$\sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i))$$

   Represents the discrepancy between the true labels y_i and the predicted values after adding the current tree $f_t(x_i)$. Here, $\hat{y}_i^{(t-1)}$ is the prediction obtained from all previous trees combined, and $f_t(x_i)$ is the current tree prediction.

2. **Regularization Term**:

   $$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

   Controls model complexity to prevent overfitting:

   - γ: Penalizes the number of leaf nodes T in each tree, controlling tree complexity.
   - λ: Penalizes large leaf weights (w_j), encouraging simpler models.

The final prediction after K boosting rounds (trees) is:

$$\hat{y}_i = \sum_{t=1}^{K} f_t(x_i)$$

1.2.2 Notation and Definitions

- x_i: Feature vector for the i-th sample.
- y_i: True label for the i-th sample.
- $\hat{y}_i^{(t-1)}$: Prediction from previous $t-1$ trees for the i-th sample.
- $f_t(x_i)$: Prediction of the current tree t for the i-th sample.
- $l(y_i, \hat{y}_i)$: Loss function (e.g., squared error for regression, log-loss for classification).
- K: Total number of boosting rounds (trees).
- T: Number of leaf nodes in the current tree.
- w_j: Weight (prediction score) of leaf node j.
- γ: Regularization parameter controlling tree complexity.
- λ: Regularization parameter controlling leaf weight magnitude.

1.2.3 Advantages and Characteristics

- High predictive accuracy through sequential error correction.
- Efficient parallelization and distributed computing capabilities.
- Built-in regularization prevents overfitting.
- Automatic handling of missing values.
- Scalable and computationally efficient.

1.3 GLMNET (Elastic Net)

GLMNET is a supervised learning algorithm that fits a generalized linear model (GLM) using **Elastic Net regularization**. Elastic Net combines L1 (Lasso) and L2 (Ridge) penalties, shrinking coefficients toward zero to reduce complexity and improve model generalization.

1.3.1 Mathematical Formulation

GLMNET solves the following optimization problem:

$$\min_{\beta_0,\beta} \left[\frac{1}{N} \sum_{i=1}^{N} w_i \, \ell\left(y_i, \beta_0 + \beta^\top x_i\right) + \lambda \left(\frac{1-\alpha}{2} \|\beta\|_2^2 + \alpha \|\beta\|_1 \right) \right]$$

The objective function consists of two parts:

1. **Loss Term**:
$$\frac{1}{N} \sum_{i=1}^{N} w_i \, \ell(y_i, \beta_0 + \beta^\top x_i)$$
Measures the discrepancy between the predicted values $\hat{y}_i$ and true labels y_i.

2. **Elastic Net Penalty** (Regularization Term):

$$\lambda \left(\frac{1-\alpha}{2} \|\beta\|_2^2 + \alpha \|\beta\|_1 \right)$$

Combines two regularizations:

- L2 penalty ($\|\beta\|_2^2$): Shrinks coefficients toward zero, preventing large values.
- L1 penalty ($\|\beta\|_1$): Encourages sparsity, pushing some coefficients exactly to zero, thus enabling feature selection.

1.3.2 Notation and Definitions

- β_0: Intercept term (bias).
- β: Vector of coefficients corresponding to the input features.
- N: Number of training samples.
- x_i: Feature vector for the i-th sample.
- y_i: True label for the i-th sample.
- w_i: Sample weight (optional; default typically 1).
- $\ell(y_i, \hat{y}_i)$: Loss function (typically negative log-likelihood), measuring error between true and predicted values.
- λ: Regularization parameter controlling penalty strength.
- α: Elastic net mixing parameter:
  - $\alpha = 1$: Pure Lasso (L1)
  - $\alpha = 0$: Pure Ridge (L2)
  - $0 < \alpha < 1$: Elastic Net (combination)
- $\|\beta\|_1$: L1 norm (sum of absolute values of coefficients).
- $\|\beta\|_2^2$: Squared L2 norm (sum of squared coefficients).

1.3.3 Advantages and Characteristics

- Combines strengths of L1 and L2 regularization.
- Good interpretability.
- Encourages sparsity (feature selection).
- Flexible tuning through parameters λ and α.

1.3.4 Remark

This model was selected as the final model for further analysis due to its strong performance and its ease of interpretability. Its interpretability stems from maintaining a linear relationship between predictors and the response variable, which allows for straightforward interpretation of feature effects. Additionally, the Elastic Net regularization induces sparsity by shrinking less relevant coefficients toward zero, resulting in a simpler and more focused model that highlights the most important variables.

2 Clustering Model Overview

2.1 K-means Clustering

K-means clustering is an unsupervised learning algorithm that partitions a dataset into K distinct, non-overlapping clusters based on feature similarity. It aims to minimize the variance within each cluster, thus grouping similar data points together.

2.1.1 Mathematical Formulation

Given a dataset $\{x_1, x_2, \ldots, x_n\}$ where each $x_i \in \mathbb{R}^d$, K-means seeks to minimize the following objective function:

$$\underset{C}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - \mu_k\|^2$$

Where:

- $C = \{C_1, C_2, \ldots, C_K\}$ is the set of clusters.
- μ_k is the centroid (mean) of cluster C_k.
- $\|x_i - \mu_k\|^2$ is the squared Euclidean distance between data point x_i and the centroid μ_k.

2.1.2 Algorithm Steps

1. Initialize K centroids randomly.
2. Assign each data point to the nearest centroid based on Euclidean distance.
3. Recompute the centroids as the mean of all data points assigned to each cluster.
4. Repeat steps 2 and 3 until convergence (no further changes in cluster assignments or centroids).

2.1.3 Advantages and Characteristics

- Simple and computationally efficient, making it scalable to large datasets.
- Works well when clusters are spherical and similarly sized.
- Sensitive to initial centroid placement; different initializations can lead to different results.
- Requires specification of the number of clusters K in advance.

2.2 Hierarchical Clustering

Hierarchical clustering is an unsupervised learning algorithm that builds a hierarchy of clusters either in a bottom-up (agglomerative) or top-down (divisive) manner. It does not require the number of clusters to be specified in advance and produces a dendrogram that visualizes the nested clustering structure.

2.2.1 Mathematical Formulation

Given a dataset $\{x_1, x_2, \ldots, x_n\}$, hierarchical clustering iteratively merges or splits clusters based on a linkage criterion, minimizing or maximizing the dissimilarity between clusters.

The linkage between two clusters A and B is defined differently depending on the chosen strategy:

- **Single linkage** (minimum distance):

$$d(A, B) = \min_{x \in A, y \in B} \|x - y\|$$

- **Complete linkage** (maximum distance):

$$d(A, B) = \max_{x \in A, y \in B} \|x - y\|$$

- **Average linkage** (average distance):

$$d(A, B) = \frac{1}{|A||B|} \sum_{x \in A} \sum_{y \in B} \|x - y\|$$

Where $\|x - y\|$ denotes the Euclidean distance between points x and y.

2.2.2 Algorithm Steps

- **Agglomerative Approach** (bottom-up):
  1. Start with each data point as a separate cluster.
  2. At each step, merge the two closest clusters based on the linkage criterion.
  3. Repeat until all points are merged into a single cluster or until a stopping condition is met.
- **Divisive Approach** (top-down):
  1. Start with all data points in a single cluster.
  2. Recursively split clusters until each cluster contains a single point or meets a stopping condition.

Common stopping conditions include reaching a predefined number of clusters, exceeding a distance threshold between clusters, or achieving a desired cluster quality based on a linkage criterion.

2.2.3 Advantages and Characteristics

- Does not require specification of the number of clusters in advance.
- Produces a dendrogram, providing rich information about the data's structure.
- Flexible with different linkage methods and distance metrics.
- Sensitive to noise and outliers, especially in the agglomerative approach.
- Computationally intensive for large datasets.

2.3 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

DBSCAN is a density-based unsupervised learning algorithm that identifies clusters as areas of high point density, separating noise points in sparse regions. It is particularly effective at discovering clusters of arbitrary shape and handling outliers.

2.3.1 Parameters

DBSCAN relies on two key parameters:

- ε (epsilon): The maximum distance between two points for one to be considered as in the neighborhood of the other.
- MinPts: The minimum number of points required to form a dense region (including the point itself).

2.3.2 Key Definitions

- **Core point**: A point with at least MinPts points (including itself) within its ε-neighborhood.
- **Directly density-reachable**: A point y is directly density-reachable from a core point x if y lies within the ε-neighborhood of x.
- **Density-reachable**: A point y is density-reachable from x if there exists a chain of points $x_1, x_2, \ldots, x_n$ with $x_1 = x$ and $x_n = y$, where each x_{i+1} is directly density-reachable from x_i.
- **Noise**: Points that are not density-reachable from any core point.

2.3.3 Algorithm Steps

1. For each unvisited point, retrieve its ε-neighborhood.
2. If the point is a core point, create a new cluster and expand it by recursively including all points that are density-reachable.
3. If the point is not a core point and not density-reachable from any other core point, label it as noise.
4. Repeat until all points have been processed.

2.3.4 Advantages and Limitations

- Can discover clusters of arbitrary shape and size.
- Robust to noise and outliers.
- Does not require prior specification of the number of clusters.
- Sensitive to the choice of ε and MinPts.
- Struggles with datasets containing clusters of varying density.

2.4 Gaussian Mixture Model (GMM)

Gaussian Mixture Model (GMM) is a probabilistic unsupervised learning algorithm that models a dataset as a mixture of several Gaussian distributions with unknown parameters. It provides a soft clustering approach where each data point is assigned a probability of belonging to each cluster.

2.4.1 Mathematical Formulation

Given a dataset $\{x_1, x_2, \ldots, x_n\}$, GMM assumes that each data point is generated from one of K Gaussian components, each defined by a mean vector μ_k, a covariance matrix Σ_k, and a mixing coefficient π_k.

The probability density function for a point x is:

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k)$$

Where:

- $\mathcal{N}(x \mid \mu_k, \Sigma_k)$ denotes the multivariate Gaussian distribution:

$$\mathcal{N}(x \mid \mu_k, \Sigma_k) = \frac{1}{(2\pi)^{d/2}|\Sigma_k|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k)\right)$$

  where d is the dimensionality of the data, i.e., the number of features in each data point.

- π_k is the mixing coefficient for the k-th Gaussian component, with $\sum_{k=1}^{K} \pi_k = 1$ and $0 \leq \pi_k \leq 1$.

Model parameters (π_k, μ_k, Σ_k) are typically estimated using the Expectation-Maximization (EM) algorithm.

2.4.2 Algorithm Steps

1. Initialize parameters (π_k, μ_k, Σ_k) randomly or using K-means clustering.
2. **Expectation step (E-step)**: Compute the responsibility $\gamma(z_{ik})$, the probability that data point x_i belongs to cluster k.

$$\gamma(z_{ik}) = \frac{\pi_k \mathcal{N}(x_i \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_i \mid \mu_j, \Sigma_j)}$$

3. **Maximization step (M-step)**: Update the parameters (π_k, μ_k, Σ_k) using the computed responsibilities:
   - Update mixing coefficients:

$$\pi_k = \frac{1}{n} \sum_{i=1}^{n} \gamma(z_{ik})$$

   - Update means:

$$\mu_k = \frac{\sum_{i=1}^{n} \gamma(z_{ik}) x_i}{\sum_{i=1}^{n} \gamma(z_{ik})}$$

   - Update covariance matrices:

$$\Sigma_k = \frac{\sum_{i=1}^{n} \gamma(z_{ik})(x_i - \mu_k)(x_i - \mu_k)^T}{\sum_{i=1}^{n} \gamma(z_{ik})}$$

4. Repeat E-step and M-step until convergence (i.e., parameters stabilize).

2.4.3 Advantages and Characteristics

- Provides soft cluster assignments (probabilistic membership).
- Can model complex cluster shapes through covariance matrices.
- More flexible than K-means, which assumes spherical clusters.
- Sensitive to initialization and may converge to local optima.
- Assumes underlying data distribution is Gaussian.

2.4.4 Remark

Gaussian Mixture Models (GMM) were excluded from the research due to their soft clustering nature, which produces probabilistic rather than fixed cluster assignments. Although GMM is capable of modeling complex, non-spherical structures, it was deemed less practical for the study's objective of creating clear and interpretable groupings for policy applications. Hard clustering methods such as K-means, hierarchical clustering, and DBSCAN were preferred for their greater interpretability in real-world contexts.

3 Forecasting Model Overview

3.1 ARIMA: AutoRegressive Integrated Moving Average

The ARIMA model is a widely used approach in time series forecasting. It is particularly effective for data that exhibit autocorrelation and non-stationarity. The acronym stands for AutoRegressive Integrated Moving Average, and the model is defined by three components:

- **AR (AutoRegressive)**: The relationship between an observation and a number of lagged observations.
- **I (Integrated)**: The differencing of raw observations to make the time series stationary.
- **MA (Moving Average)**: The dependency between an observation and a residual error from a moving average model applied to lagged observations.

Mathematically, the ARIMA(p, d, q) model is expressed as:

$$Y_t = c + \sum_{i=1}^{p} \phi_i Y_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t$$

Note: this equation is applied after the original time series has been differenced d times to ensure stationarity.

- Y_t is the observed value at time t (after differencing).
- c is a constant term.
- ϕ_i are the autoregressive coefficients.
- θ_j are the moving average coefficients.
- ε_t is white noise.
- d is the number of times the original series has been differenced to remove trends and achieve stationarity.

In this study, the `auto.arima()` function from the `forecast` package in R was used to automatically select the optimal model order by minimizing the Akaike Information Criterion (AIC).

3.2 Prophet: Decomposable Time Series Model

Prophet is a decomposable time series forecasting model developed by Facebook, designed to handle time series with strong trends and seasonality, even when data is missing or contains outliers. The model represents the time series as the sum of several components:

$$y(t) = g(t) + s(t) + h(t) + \varepsilon_t$$

Where:

- $g(t)$ represents the trend component (linear or logistic growth),
- $s(t)$ captures periodic seasonal effects using Fourier series,
- $h(t)$ represents the effects of holidays (if specified),
- ε_t is the error term.

For this study, the dataset consists of annual observations. As a result, the seasonality term $s(t)$ was disabled. The holiday component $h(t)$ was excluded as well, since holiday effects require higher-frequency data and cannot be meaningfully captured at the annual level. Despite the absence of these components, however, Prophet remains practically useful as a flexible, trend-focused forecasting model, well suited for capturing long-term patterns. Therefore, the use of the Prophet model remains appropriate for this study. The model was implemented using the `prophet` package in R, with forecasts generated through automatic trend fitting and uncertainty estimation. Prophet is widely used in applied forecasting for its flexibility, interpretability, and robust performance on real-world data.