Predicting Economic Growth: An Analysis of GDP Drivers with Penalized Regression

FEM11149 - Introduction to Data Science

Michael van Walsum 756104 (25%), Aleksandra Tatko 648925 (25%)
Stijn Hooijman 620083 (25%), Francesco Di Presa 771382 (25%)

September, 2025

1. Executive Summary

This report aims at exploring whether one can predict the growth of a country's GDP, given a set of economic and demographic variables. The main objective is to identify which modeling approaches provide the most reliable predictions of GDP growth. If a country wants to borrow money to stimulate growth through increased spending, it may be helpful to understand what factors drive economic growth, as expressed in terms of GDP. After preparing the data and conducting descriptive analysis, this report will mostly explore the varying explanatory power of different regression methods and provide reasoning as to why penalised regression techniques such as LASSO and Ridge regression are the most suitable method for this data. Finally, the report will conclude with the best model for predicting annual % GDP growth, as well as outline some interesting data limitations and considerations when working with penalised regression. The report concludes that Traditional regression (OLS) has weak explanatory power for this data set. Non-linear effects modestly improved diagnostics but overall, penalised regressions (LASSO, Ridge and Elastic Net) provided more stable models through variable selection. Finally, we conclude that a logistic extension using a binary variable gives practical classification insights.

2. Data, Variable Section and Methodology

2.1 Variable Selection

In the report, the models are created using a selection of 74 variables, across 150 instances, corresponding to different countries. Our first linear regression was performed with the following chosen variables: gdp per capita (PPP), net trade, unemployment, age dependency, urban percentage of population, life expectancy of females, tax payments, population growth and the main dependent variable: gdp growth. However, later in the report, penalised regression uses all variables. We selected the initial regression variables as they capture key economic, demographic, and structural drivers of growth, consistent with current literature. GDP per capita was included as Barro (1991) shows that poorer countries tend to grow faster. Demographic measures such as the age dependency ratio and population growth reflect labor force dynamics, with Bloom and Williamson (1998) linking favorable demographics to higher growth. Urban population percentage, highlighted by Henderson (2003), drives productivity, while trade openness has been shown by Frankel and Romer (1999) to foster growth. Finally, female life expectancy serves as a proxy for overall health outcomes, which Bloom, Canning, and Sevilla (2004) associate with stronger long-run growth.

$$\begin{aligned} y_i = \beta_0 &+ \beta_1 \,\text{GDP per capita (PPP)} + \beta_2 \,\text{Unemployment} + \beta_3 \,\text{Age dependency} \\ &+ \beta_4 \,\text{Unemployment}^2 + \beta_5 \,\text{Urban population} + \beta_6 \,\text{Life expectancy (female)} \\ &+ \beta_7 \,\text{Population growth} + \beta_8 \,\text{Tax payments} + \varepsilon_i \end{aligned}$$

Formula 1. Baseline linear specification for GDP growth as a function of income, unemployment (quadratic), demographics, urbanization, health, population growth, and tax burden.

2.2 Methodology

We began by running linear regressions (OLS) as a baseline, because it is simple, transparent, and provides a benchmark for comparison. We ran 2 different linear regressions, one that included net trade, and one without. We tested models with and without Net Trade because, while trade is a key component of GDP in theory (exports minus imports), our descriptive statistics showed extreme outliers (e.g. major oil exporters) that risk distorting its explanatory power for most countries. (Figure 1, Appendix 1) We extended Model A by adding a nonlinear effect of unemployment, since labor market conditions are a key driver of growth and their impact is not linear—for example, a 1% increase in unemployment may have very different effects on GDP growth at 4% compared to 25%. Next, we applied penalized regression: LASSO, Ridge, and Elastic Net. These methods reduce noise, handle correlated predictors, and avoid overfitting by shrinking or eliminating weak variables. We split the dataset into 110 training and 40 test observations to balance training strength with reliable out-of-sample evaluation. We posed one final question with the data, and ran a Logistic Ridge regression with a binary growth indicator to test whether we could more usefully classify countries as growing above or below a practical threshold (2.7%) instead of predicting exact growth rates. This approach is often more practical for policy or lending decisions, where a clear yes/no signal is easier to act on than a precise forecast.

3. Results

3.1 Baseline Linear Regression

Our initial baseline regression without Net Trade (Model A, adj. $R^2 = 0.0138$) (Table 1, Appendix 2.1) performed slightly better than the model with Net Trade (Model B, adj. $R^2 = 0.0072$) (Table 2, Appendix 2.2). This shows that Net Trade added noise rather than useful insight, so we dropped it from further analysis. We then extended Model A by including a squared term for unemployment. This improved the model's adjusted R^2 to 0.0323, with unemployment and its squared term appearing marginally significant (Model C, Table 3, Appendix 2.3). However, the overall explanatory power of the OLS model remained weak, with most other predictors statistically insignificant. As shown in Appendix 2.4, none of the predictors exhibit a strong individual relationship with GDP growth. At the same time, standard validation checks confirmed the model did not suffer from major violations (Appendix 2.5), so the issue lay not in misspecified assumptions but in the lack of predictive strength of the variables. For this reason, we moved beyond OLS and applied penalized regression techniques such as LASSO. Unlike OLS, LASSO penalizes weak predictors, shrinking some coefficients to zero. This allowed us to test whether any of our pre-selected variables were truly robust drivers of GDP growth, while filtering out noise and improving out-of-sample reliability.

3.2 Lasso Penalised Regression

We applied a 10-fold cross-validated LASSO regression to identify the most relevant predictors of GDP growth. Two solutions emerged from the cross-validation process. At the minimum penalty level (lambda.min = 0.2740), the model retained two predictors—GDP per capita (PPP) and unemployment—suggesting these have modest predictive power. At the more conservative penalty (lambda.1se = 0.6329), however, all coefficients were shrunk to zero, suggesting that none of the predictors added robust explanatory power across folds

(Table 4, Appendix 3.1). In practical terms, lambda (lambda) is the penalty strength: higher values shrink coefficients more strongly, reducing noise but also removing weaker signals. Looking at both lambda.min and lambda.1se is standard practice. lambda.min gives the "best fit" inside the training data but risks overfitting, while lambda.1se offers a simpler, more conservative model that generalizes better. Compared to Model C, the coefficients from LASSO were notably smaller. This is expected, since LASSO imposes a penalty on coefficient size to prevent overfitting, which both shrinks estimates and can set uninformative predictors to exactly zero. Finally, it is important to note that all predictors were standardized before applying LASSO, so that differences in scale (e.g, GDP per capita in dollars vs. unemployment in percentages) did not distort the penalization process. Without standardization, variables measured on larger scales would be penalized less, biasing variable selection. Overall, LASSO confirmed that only income levels and labor market conditions carry rather weak signals for GDP growth, while most other predictors appear irrelevant in this dataset.

3.3 Ridge and Elastic Net Regression

After the LASSO, we shifted focus to Ridge and Elastic Net, which keep all predictors and are better suited for capturing joint effects across many small but correlated variables. Using cross-validation, we compared performance at the lambda that minimizes error (lambda.min) and the more conservative 1-SE solution (lambda.1se). The results were clear: the 1-SE models consistently outperformed the lambda.min versions, achieving lower prediction errors (RMSE 2.81 vs. 3.11 for Ridge, and 2.81 vs. 3.02 for Elastic Net) (Figure 1). At the 1-SE level, Ridge kept all predictors with small effects, while Elastic Net went further by dropping the weakest ones and keeping only a smaller, more meaningful subset.

Model	TN	FP	FN	TP	Accuracy
Ridge (lambda.min)	11	3	6	20	0.775
Ridge (lambda.1se)	11	3	8	18	0.725

Table 1. Comparison between Ridge models using minimum Lambda and 1-Se Lambda

In our dataset, Elastic Net selected an alpha close to zero, effectively behaving like Ridge, which suggests that GDP growth here is shaped by many small, correlated drivers rather than a few dominant ones. The consistent negative signs on unemployment variables reinforce their role as headwinds to growth, while other coefficients remain small. Full coefficient tables are provided in the Appendix 3.2. These results highlight the importance of selecting lambda carefully: more penalization often produces models that are simpler and more robust out of sample. Choosing lambda solely to minimize cross-validation error can lead to overfitting, because Ridge and Elastic Net estimate lambda based only on the training data. This may result in a model that predicts the training outcomes very accurately but performs poorly on new, unseen data. The 1-SE rule mitigates this risk by selecting a slightly larger, more conservative lambda within one standard error of the minimum, which shrinks the coefficients further. This reduces the tendency of the model to fit the training data too closely, improving generalization and predictive performance. Consequently, it is good practice to evaluate both the lambda that minimizes cross-validation error and the lambda suggested by the 1-SE rule. Finally, there are a number of reasons why running a penalised regression is more useful than multiple linear regression. Firstly, penalised regression tackles the problem of multicollinearity. In economic data, predictors are often correlated. OLS struggles with this and predictor significance can change dramatically when variables are added or dropped. Penalised regression handles this by stabilising estimates by shrinking coefficients toward zero. Penalised regression also counters overfitting risk. Regularisation penalises 'noisy' predictors meaning results are more reliable out-of-sample. Additionally, allowing LASSO to choose for significant variables and Ridge to reduce insignificant predictors impact, there can be consistency in variable selection making the model more robust and reproducible than if this selection was done manually.

3.4 Binary Classification of Growth

Because many policy decisions are yes/no rather than continuous, we also reframed the question: will a country grow faster than a practical benchmark? This makes the model easier to act on in practice: instead of predicting a noisy growth rate, we can give a simple yes/no on whether growth is likely to beat the benchmark. We created a binary flag, Growing_more, equal to 1 if growth exceeds 2.7 percent and 0 otherwise and applied the ridge regression. At the lambda that minimizes error, the model achieved about 77.5% accuracy, while the more conservative 1-SE solution reached 72.5% (Figure 2). Ridge was a sensible choice here, as it stabilizes predictions across many correlated economic indicators rather than relying on just one or two. We repeated the classification with a different train–test split. Performance moved only slightly, which is exactly what we would expect given a modest signal: the model isn't brittle, and small resampling differences don't overturn the result.

Model	TN	FP	FN	TP	Accuracy
Ridge (lambda.min)	11	3	6	20	0.775
Ridge (lambda.1se)	11	3	8	18	0.725

Table 2. Baseline linear specification for GDP growth as a function of income, unemployment (quadratic), demographics, urbanization, health, population growth, and tax burden.

4. Conclusion

To conclude, the results show that penalised regressions clearly outperform OLS, with the Ridge/Elastic Net models at the 1-SE solution providing the most robust and stable predictions of annual GDP growth, despite the weak predictive outcomes of the variables seen across the sample set. If additional data were available, more countries (rows) may improve the predictive power of these models as it will make cross-validation steadier and out of sample results more accurate. One of the main limitations of the data is in sample size rather than dimensionality. Larger samples would improve both the stability of coefficient estimates and the reliability of cross-validation. Predicting continuous GDP growth is useful for economic research and detailed forecasting, but in practice policymakers often need a simpler, binary answer. In this case, predicting whether growth is on par, or below a certain level, (the Growing_more model) is more actionable, as it aligns with policy decisions such as whether to stimulate the economy or adjust borrowing.

References

Barro, Robert J. 1991. "Economic Growth in a Cross Section of Countries." *The Quarterly Journal of Economics* 106 (2): 407–43.

Bloom, David E., David Canning, and Jaypee Sevilla. 2004. "The Effect of Health on Economic Growth: A Production Function Approach." *World Development* 32 (1): 1–13.

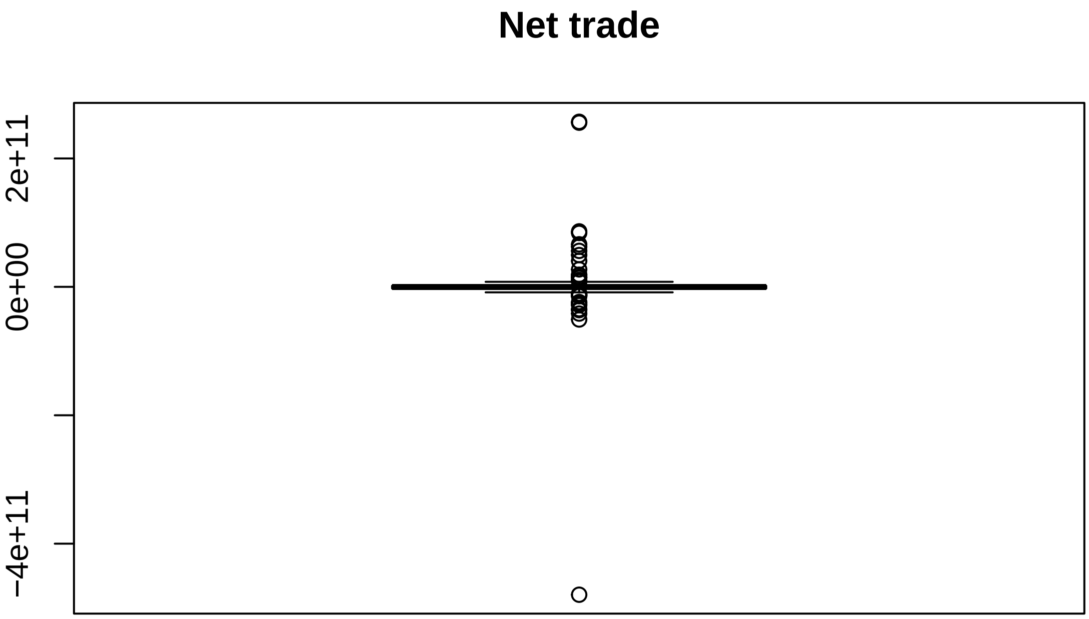
Bloom, David E., and Jeffrey G. Williamson. 1998. "Demographic Transitions and Economic Miracles in Emerging Asia." *The World Bank Economic Review* 12 (3): 419–55.

Frankel, Jeffrey A., and David H. Romer. 1999. "Does Trade Cause Growth?" *American Economic Review* 89 (3): 379–99.

Henderson, J. Vernon. 2003. "The Urbanization Process and Economic Growth: The so-What Question." *Journal of Economic Growth* 8 (1): 47–71.

Appendix

Appendix 1



Appendix 2

Appedix 2.1 Table 1

```
##
## Call:
## lm(formula = formula_A, data = model_df_linear, singular.ok = FALSE)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -36.619  -0.967   0.402   2.098  11.142
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -5.677e-01  7.087e+00  -0.080   0.9363
## gdp_pc_ppp             -4.734e-05  2.657e-05  -1.781   0.0770 .
## unemployment_total     -1.242e-01  6.373e-02  -1.949   0.0533 .
## age_dependency         -1.479e-02  3.155e-02  -0.469   0.6399
## urban_pct              -1.674e-02  2.271e-02  -0.737   0.4621
## life_expectancy_female  8.928e-02  8.075e-02   1.106   0.2708
## tax_payments           -6.856e-03  2.992e-02  -0.229   0.8191
## population_growth       2.888e-01  4.070e-01   0.710   0.4791
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 4.333 on 142 degrees of freedom
## Multiple R-squared:  0.06016,    Adjusted R-squared:  0.01383
## F-statistic: 1.298 on 7 and 142 DF,  p-value: 0.2553
```

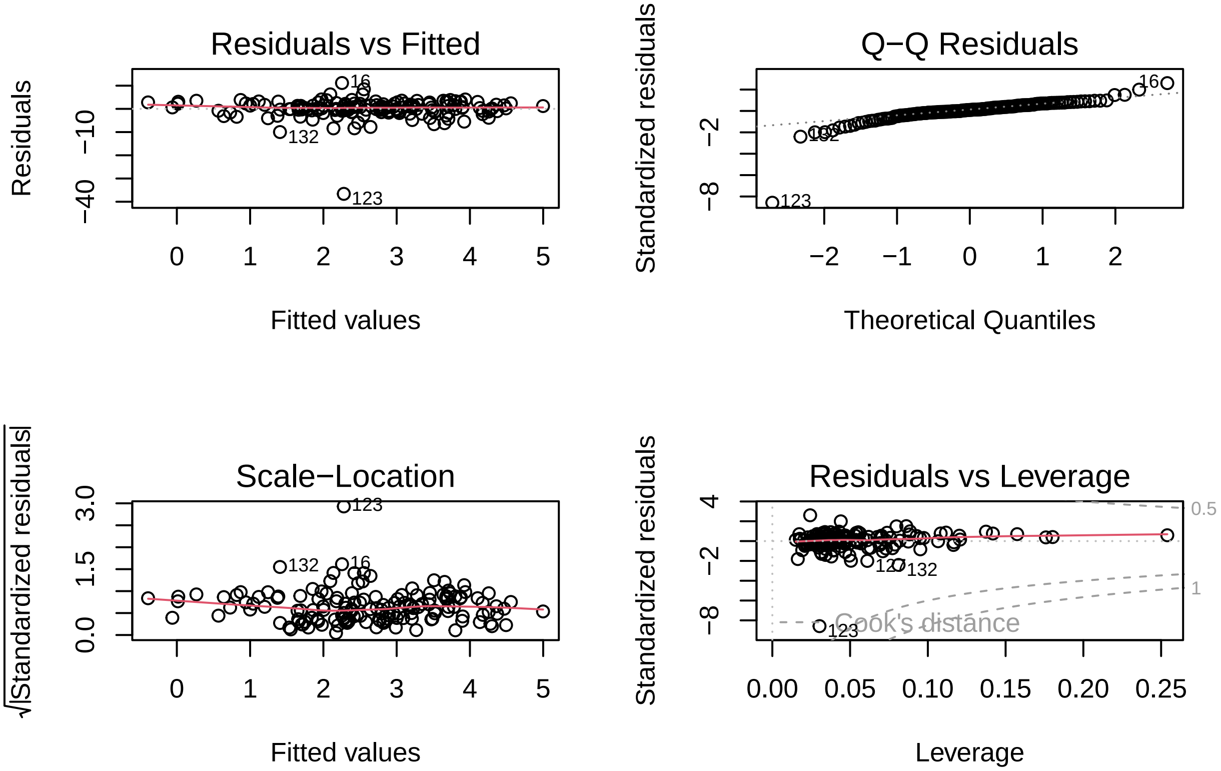
Appedix 2.2 Table 2

```
##
## Call:
## lm(formula = formula_B, data = model_df_linear, singular.ok = FALSE)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -36.618  -0.983   0.424   2.120  11.146
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -5.881e-01  7.111e+00  -0.083   0.9342
## gdp_pc_ppp             -4.734e-05  2.666e-05  -1.776   0.0779 .
## unemployment_total     -1.235e-01  6.401e-02  -1.929   0.0557 .
## age_dependency         -1.459e-02  3.166e-02  -0.461   0.6457
## urban_pct              -1.672e-02  2.278e-02  -0.734   0.4641
## life_expectancy_female  8.919e-02  8.102e-02   1.101   0.2729
## tax_payments           -6.689e-03  3.002e-02  -0.223   0.8240
## net_trade               1.651e-12  6.860e-12   0.241   0.8101
## population_growth       2.910e-01  4.084e-01   0.713   0.4773
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 4.347 on 141 degrees of freedom
## Multiple R-squared:  0.06054,    Adjusted R-squared:  0.00724
## F-statistic: 1.136 on 8 and 141 DF,  p-value: 0.343
```

Appedix 2.3 Table 3

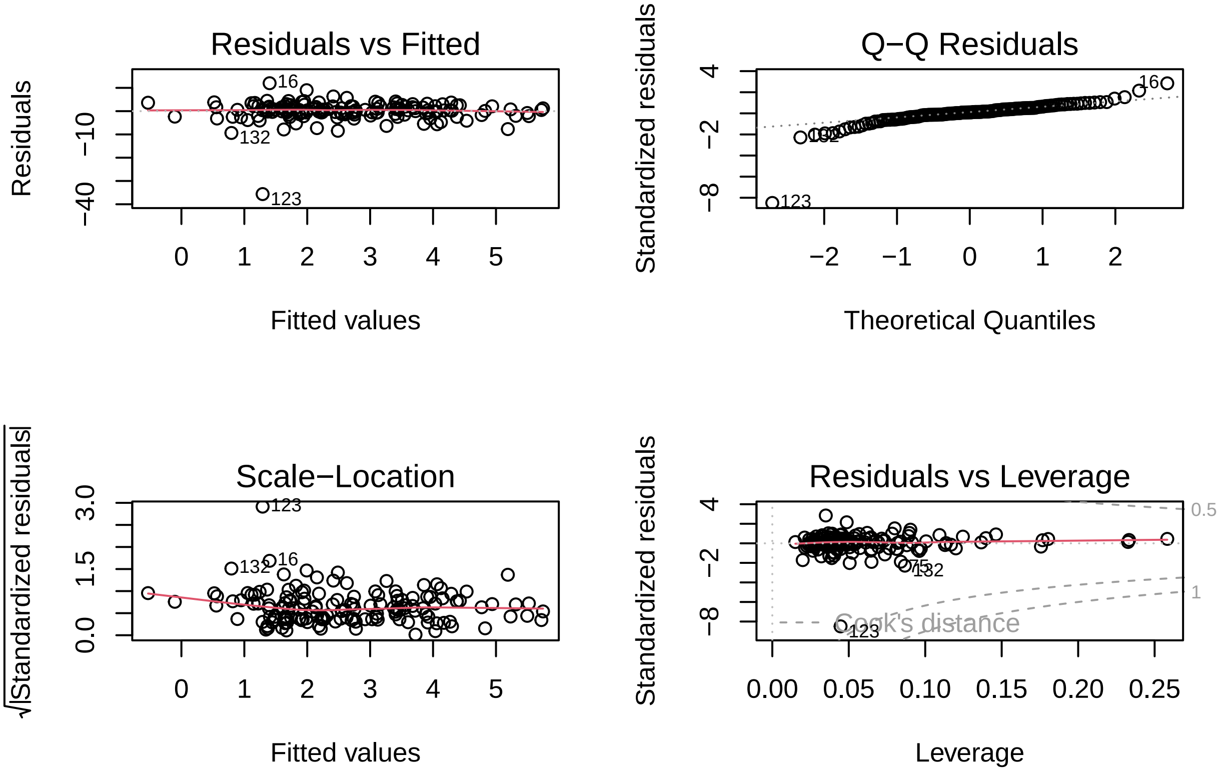
```
##
## Call:
## lm(formula = formula_C, data = model_df_linear, singular.ok = FALSE)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.631  -0.927   0.377   1.892  11.994
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -1.519e+00  7.038e+00  -0.216   0.8294
## gdp_pc_ppp              -5.435e-05  2.657e-05  -2.045   0.0427 *
## unemployment_total      -5.091e-01  2.097e-01  -2.428   0.0165 *
## age_dependency          -6.204e-03  3.157e-02  -0.197   0.8445
## I(unemployment_total^2)  1.539e-02  7.998e-03   1.925   0.0563 .
## urban_pct               -7.888e-03  2.296e-02  -0.344   0.7317
## life_expectancy_female   1.107e-01  8.076e-02   1.370   0.1728
## population_growth        1.790e-01  4.072e-01   0.440   0.6609
## tax_payments            -1.377e-03  2.977e-02  -0.046   0.9632
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.292 on 141 degrees of freedom
## Multiple R-squared:  0.08422,    Adjusted R-squared:  0.03226
## F-statistic: 1.621 on 8 and 141 DF,  p-value: 0.1239
```

Appedix 2.4



Model A

```
##            gdp_pc_ppp     unemployment_total         age_dependency
##              2.441310               1.149864               2.508492
##             urban_pct life_expectancy_female           tax_payments
##              2.180537               3.586917               1.885227
##     population_growth
##              1.860747
```

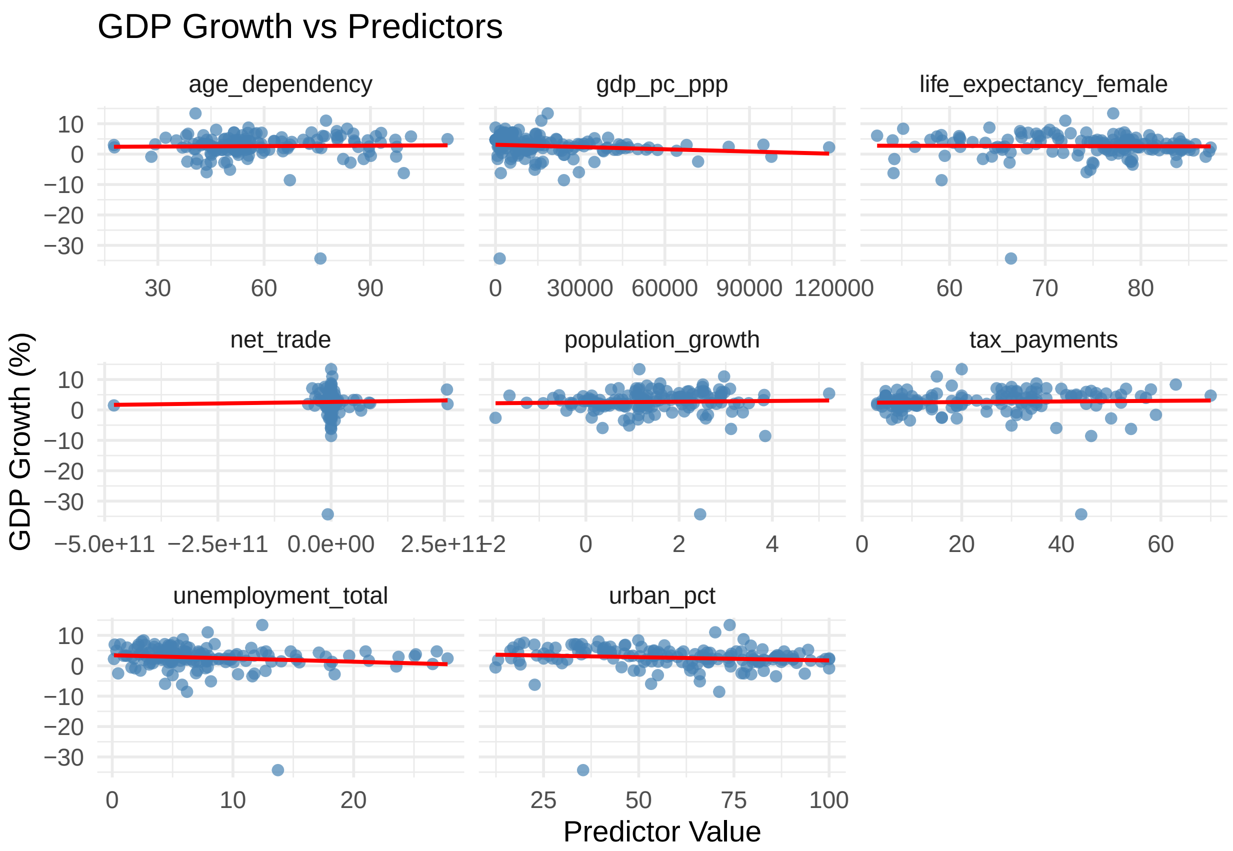


Model C

```
##              gdp_pc_ppp      unemployment_total          age_dependency
##                2.488173               12.690382                2.559617
## I(unemployment_total^2)               urban_pct  life_expectancy_female
##               12.094107                2.271751                3.656133
##       population_growth            tax_payments
##                1.898010                1.902621
```

Appedix 2.5

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



Appendix 3

Appendix 3.1

```
## 9 x 1 sparse Matrix of class "dgCMatrix"
##                             lambda.min
## (Intercept)              3.431018e+00
## gdp_pc_ppp              -1.477082e-05
## unemployment_total      -6.864171e-02
## age_dependency           .
## I(unemployment_total^2)  .
## urban_pct                .
## life_expectancy_female   .
## population_growth        .
## tax_payments             .
```

```
## 9 x 1 sparse Matrix of class "dgCMatrix"
##                          lambda.1se
## (Intercept)               2.624503
## gdp_pc_ppp                .
## unemployment_total        .
## age_dependency            .
## I(unemployment_total^2)   .
## urban_pct                 .
## life_expectancy_female    .
## population_growth         .
## tax_payments              .
```

Appendix 3.2

3.2.1 Ridge lambda-1se coefficients

```
## 73 x 1 sparse Matrix of class "dgCMatrix"
##                                   s0
## (Intercept)              2.415564e+00
## electricity_access       3.028858e-06
## adolescent_fertility    -1.426701e-05
## age_dependency          -4.694881e-06
## contrib_family_fem       4.595036e-05
## contrib_family_male      1.594978e-04
## contrib_family_total     1.652610e-04
## credit_info_index        4.558960e-04
## employers_fem           -2.915627e-03
## employers_male          -1.447401e-03
## employers_total         -1.921476e-03
## emp_agriculture_total    3.263641e-05
## emp_agriculture_fem      1.310494e-05
## emp_agriculture_male     3.605653e-05
## emp_industry_total       1.689826e-04
## emp_industry_fem         1.074809e-04
## emp_industry_male        1.025238e-04
## emp_services_total      -8.548647e-05
## emp_services_fem        -2.802237e-05
## emp_services_male       -1.134050e-04
## export_value_index       4.413186e-06
## export_volume_index      3.252746e-06
## fertility_rate_total    -1.369764e-04
## broadband_subs          -5.488028e-05
## gdp_pc_const2005        -6.005174e-08
## gdp_pc_ppp              -5.133555e-08
## internet_users          -3.219750e-05
## lfpr_fem                 3.304499e-05
## lfpr_male                1.433193e-04
## lfpr_total               9.562140e-05
## labor_force_total        1.551653e-11
## life_expectancy_female   7.774498e-06
## life_expectancy_male     3.716307e-05
## mobile_subs              3.344221e-06
## own_account_fem          1.987298e-05
## own_account_male         8.479930e-05
## own_account_total        5.543618e-05
## pop_0_14_pct             9.583884e-06
## pop_0_14_total           3.332911e-11
## pop_15_64_pct            7.976191e-06
## pop_15_64_total          1.151470e-11
## pop_65plus_pct          -3.810223e-05
## pop_65plus_total         7.394797e-11
## pop_density             -2.714983e-07
## population_growth      -8.500107e-05
## pop_total               7.912561e-12
## credit_coverage_priv    1.852401e-06
## credit_coverage_pub     2.111608e-05
## rural_pop_total         1.534271e-11
## rural_pop_pct           3.836757e-05
## self_emp_fem            1.455672e-05
## self_emp_male           4.709158e-05
## self_emp_total          3.874544e-05
## tax_payments            1.446303e-05
## telephone_lines        -2.375785e-05
## contract_days          -2.576590e-06
## start_business_days    -4.506466e-05
## tax_prep_hours         -4.987756e-06
## unemployment_fem       -2.227386e-04
## unemployment_male      -3.861373e-04
## unemployment_total     -3.199327e-04
## unemp_youth_fem        -7.471961e-05
## unemp_youth_male       -1.545431e-04
## unemp_youth_total      -1.307832e-04
## urban_pop_total         1.354283e-11
## urban_pct              -3.836753e-05
## vulner_emp_fem          2.331695e-05
## vulner_emp_male         7.052348e-05
## vulner_emp_total        5.610468e-05
## waged_emp_fem          -1.455670e-05
## waged_emp_male         -4.709158e-05
## waged_emp_total        -3.874545e-05
## net_trade               9.419828e-15
```

3.2.2 Ridge lambda-min coefficients

```
## 73 x 1 sparse Matrix of class "dgCMatrix"
##                                   s0
## (Intercept)             2.185818e+00
## electricity_access      1.212976e-03
## adolescent_fertility   -9.530809e-04
## age_dependency         -6.274850e-04
## contrib_family_fem      3.316413e-04
## contrib_family_male     3.503657e-03
## contrib_family_total    5.151960e-03
## credit_info_index       2.296204e-02
## employers_fem          -1.241778e-01
## employers_male         -5.612937e-02
## employers_total        -7.770964e-02
## emp_agriculture_total   3.552525e-04
## emp_agriculture_fem    -4.401330e-04
## emp_agriculture_male    4.230492e-04
## emp_industry_total      8.409673e-03
## emp_industry_fem        3.920077e-03
## emp_industry_male       5.826627e-03
## emp_services_total     -2.474725e-03
## emp_services_fem       -8.993554e-05
## emp_services_male      -3.418904e-03
## export_value_index      1.812926e-04
## export_volume_index     1.297501e-04
## fertility_rate_total   -1.915798e-02
## broadband_subs         -1.109780e-03
## gdp_pc_const2005       -2.315548e-06
## gdp_pc_ppp             -2.359331e-06
## internet_users         -8.639382e-04
## lfpr_fem                4.615096e-04
## lfpr_male               4.207983e-03
## lfpr_total              2.102273e-03
## labor_force_total       3.755613e-10
## life_expectancy_female  3.925169e-03
## life_expectancy_male    5.399811e-03
## mobile_subs             1.936810e-04
## own_account_fem        -4.772895e-04
## own_account_male        2.333292e-03
## own_account_total       1.028636e-03
## pop_0_14_pct           -7.588029e-04
## pop_0_14_total          8.136241e-10
## pop_15_64_pct           1.003016e-03
## pop_15_64_total         2.875231e-10
## pop_65plus_pct          1.041016e-03
## pop_65plus_total        1.556436e-09
## pop_density            -1.224629e-05
## population_growth      -1.588637e-02
## pop_total               1.943332e-10
## credit_coverage_priv    2.061769e-04
## credit_coverage_pub     1.208136e-03
## rural_pop_total         3.969891e-10
## rural_pop_pct           5.584753e-04
## self_emp_fem           -5.792277e-04
## self_emp_male           8.777562e-04
## self_emp_total          5.704671e-04
## tax_payments            2.857341e-04
## telephone_lines         5.896603e-04
## contract_days          -1.606202e-04
## start_business_days    -2.441060e-03
## tax_prep_hours         -2.627322e-04
## unemployment_fem       -6.537956e-03
## unemployment_male      -1.222118e-02
## unemployment_total     -8.765505e-03
## unemp_youth_fem        -1.152225e-03
## unemp_youth_male       -4.244909e-03
## unemp_youth_total      -3.045090e-03
## urban_pop_total         3.068793e-10
## urban_pct              -5.604294e-04
## vulner_emp_fem         -1.895280e-04
## vulner_emp_male         1.859822e-03
## vulner_emp_total        1.324326e-03
## waged_emp_fem           5.713420e-04
## waged_emp_male         -8.911309e-04
## waged_emp_total        -5.841588e-04
## net_trade               3.466179e-13
```

3.2.3 Elastic net lambda-1se coefficients

```
## 73 x 1 sparse Matrix of class "dgCMatrix"
##                                   s0
## (Intercept)             2.422786e+00
## electricity_access      .
## adolescent_fertility   -1.245303e-06
## age_dependency          .
## contrib_family_fem      1.515249e-05
## contrib_family_male     8.705468e-05
## contrib_family_total    1.151474e-04
## credit_info_index       2.934770e-04
## employers_fem          -2.622795e-03
## employers_male         -1.291619e-03
## employers_total        -1.732106e-03
## emp_agriculture_total   1.264720e-05
## emp_agriculture_fem     .
## emp_agriculture_male    1.490914e-05
## emp_industry_total      1.172051e-04
## emp_industry_fem        5.881060e-05
## emp_industry_male       5.661298e-05
## emp_services_total     -6.186943e-05
## emp_services_fem       -1.068993e-05
## emp_services_male      -8.459671e-05
## export_value_index      3.452043e-06
## export_volume_index     2.312558e-06
## fertility_rate_total    .
## broadband_subs         -1.986962e-05
## gdp_pc_const2005       -3.394855e-08
## gdp_pc_ppp             -2.915673e-08
## internet_users         -1.555244e-05
## lfpr_fem                4.622448e-06
## lfpr_male               9.004175e-05
## lfpr_total              5.184000e-05
## labor_force_total       1.035006e-11
## life_expectancy_female  .
## life_expectancy_male    .
## mobile_subs             .
## own_account_fem         .
## own_account_male        5.954069e-05
## own_account_total       3.164719e-05
## pop_0_14_pct            .
## pop_0_14_total          2.268548e-11
## pop_15_64_pct           .
## pop_15_64_total         7.849802e-12
## pop_65plus_pct          .
## pop_65plus_total        4.452052e-11
## pop_density            -4.994884e-08
## population_growth       .
## pop_total               5.383910e-12
## credit_coverage_priv    .
## credit_coverage_pub     .
## rural_pop_total         1.076828e-11
## rural_pop_pct           1.808071e-05
## self_emp_fem            .           
## self_emp_male           2.556880e-05
## self_emp_total          1.991348e-05
## tax_payments            .           
## telephone_lines         .           
## contract_days          -1.063154e-06
## start_business_days    -2.897854e-05
## tax_prep_hours         -2.968314e-06
## unemployment_fem       -1.637885e-04
## unemployment_male      -2.916808e-04
## unemployment_total     -2.336506e-04
## unemp_youth_fem        -4.496437e-05
## unemp_youth_male       -1.115972e-04
## unemp_youth_total      -9.085908e-05
## urban_pop_total         8.353468e-12
## urban_pct              -1.808070e-05
## vulner_emp_fem          7.673483e-06
## vulner_emp_male         4.984046e-05
## vulner_emp_total        3.773541e-05
## waged_emp_fem           .           
## waged_emp_male         -2.556879e-05
## waged_emp_total        -1.991348e-05
## net_trade               7.615155e-16
```

3.2.4 Elastic net lambda-min coefficients

```
## 73 x 1 sparse Matrix of class "dgCMatrix"
##                                   s0
## (Intercept)             2.253690e+00
## electricity_access      8.737800e-04
## adolescent_fertility   -7.671287e-04
## age_dependency         -5.411143e-04
## contrib_family_fem      3.929762e-04
## contrib_family_male     3.021736e-03
## contrib_family_total    4.219737e-03
## credit_info_index       1.826051e-02
## employers_fem          -1.003579e-01
## employers_male         -4.614546e-02
## employers_total        -6.335639e-02
## emp_agriculture_total   3.227296e-04
## emp_agriculture_fem    -2.675424e-04
## emp_agriculture_male    3.823324e-04
## emp_industry_total      6.760179e-03
## emp_industry_fem        3.312742e-03
## emp_industry_male       4.644601e-03
## emp_services_total     -2.056690e-03
## emp_services_fem       -1.517342e-04
## emp_services_male      -2.840569e-03
## export_value_index      1.457598e-04
## export_volume_index     1.047447e-04
## fertility_rate_total   -1.484726e-02
## broadband_subs         -8.971067e-04
## gdp_pc_const2005       -1.788953e-06
## gdp_pc_ppp             -1.768145e-06
## internet_users         -6.683895e-04
## lfpr_fem                4.652825e-04
## lfpr_male               3.594952e-03
## lfpr_total              1.925040e-03
## labor_force_total       3.351817e-10
## life_expectancy_female  2.852261e-03
## life_expectancy_male    4.013472e-03
## mobile_subs             1.820512e-04
## own_account_fem        -2.649744e-04
## own_account_male        1.924339e-03
## own_account_total       8.867875e-04
## pop_0_14_pct           -6.301831e-04
## pop_0_14_total          7.181159e-10
## pop_15_64_pct           9.535325e-04
## pop_15_64_total         2.543396e-10
## pop_65plus_pct          6.852006e-04
## pop_65plus_total        1.452342e-09
## pop_density            -9.484870e-06
## population_growth      -1.205153e-02
## pop_total               1.724202e-10
## credit_coverage_priv    1.698051e-04
## credit_coverage_pub     9.475387e-04
## rural_pop_total         3.471110e-10
## rural_pop_pct           5.347587e-04
## self_emp_fem           -3.516226e-04
## self_emp_male           7.420641e-04
## self_emp_total          5.019667e-04
## tax_payments            1.412318e-04
## telephone_lines         3.179185e-04
## contract_days          -1.218542e-04
## start_business_days    -1.898254e-03
## tax_prep_hours         -2.048908e-04
## unemployment_fem       -5.572016e-03
## unemployment_male      -1.018651e-02
## unemployment_total     -7.555892e-03
## unemp_youth_fem        -1.154137e-03
## unemp_youth_male       -3.630878e-03
## unemp_youth_total      -2.712789e-03
## urban_pop_total         2.783597e-10
## urban_pct              -5.360388e-04
## vulner_emp_fem         -3.755951e-05
## vulner_emp_male         1.549014e-03
## vulner_emp_total        1.114868e-03
## waged_emp_fem           3.469820e-04
## waged_emp_male         -7.498013e-04
## waged_emp_total        -5.098157e-04
## net_trade               2.793483e-13
```

Code

```
## Preparation
pacman::p_load(tidyverse,ggplot2,dplyr,car,caret,caretEnsemble,elasticnet,glmnet,broom,psych,corrplot)
df <- read_csv("a1_data_group_2.csv")
df <- df %>%
  dplyr::rename(
    country                = `Country`,
    electricity_access     = `Access to electricity (% of population)`,
    adolescent_fertility   = `Adolescent fertility rate (births per 1,000 women ages 15-19)`,
    age_dependency   = `Age dependency ratio (% of working-age population)`,
    contrib_family_fem     = `Contributing family workers, female (% of female employment) (modeled ILO
    contrib_family_male    = `Contributing family workers, male (% of male employment) (modeled ILO est
    contrib_family_total   = `Contributing family workers, total (% of total employment) (modeled ILO e
    credit_info_index      = `Depth of credit information index (0=low to 8=high)`,
    employers_fem          = `Employers, female (% of female employment) (modeled ILO estimate)`,
    employers_male         = `Employers, male (% of male employment) (modeled ILO estimate)`,
    employers_total        = `Employers, total (% of total employment) (modeled ILO estimate)`,
    emp_agriculture_total  = `Employment in agriculture (% of total employment) (modeled ILO estimate)`
    emp_agriculture_fem    = `Employment in agriculture, female (% of female employment) (modeled ILO e
    emp_agriculture_male   = `Employment in agriculture, male (% of male employment) (modeled ILO estim
    emp_industry_total     = `Employment in industry (% of total employment) (modeled ILO estimate)`,
    emp_industry_fem       = `Employment in industry, female (% of female employment) (modeled ILO esti
    emp_industry_male      = `Employment in industry, male (% of male employment) (modeled ILO estimate
    emp_services_total     = `Employment in services (% of total employment) (modeled ILO estimate)`,
    emp_services_fem       = `Employment in services, female (% of female employment) (modeled ILO esti
    emp_services_male      = `Employment in services, male (% of male employment) (modeled ILO estimate
    export_value_index     = `Export value index (2000 = 100)`,
    export_volume_index    = `Export volume index (2000 = 100)`,
    fertility_rate_total   = `Fertility rate, total (births per woman)`,
    broadband_subs         = `Fixed broadband Internet subscribers (per 100 people)`,
    gdp_growth             = `GDP growth (annual %)`,
    gdp_pc_const2005       = `GDP per capita (constant 2005 US$)`,
    gdp_pc_ppp             = `GDP per capita, PPP (constant 2011 international $)`,
    internet_users         = `Individuals using the Internet (% of population)`,
    lfpr_fem               = `Labor force participation rate, female (% of female population ages 15+)
    lfpr_male              = `Labor force participation rate, male (% of male population ages 15+) (mode
    lfpr_total             = `Labor force participation rate, total (% of total population ages 15+) (mo
    labor_force_total      = `Labor force, total`,
    life_expectancy_female           = `Life expectancy at birth, female (years)`,
    life_expectancy_male          = `Life expectancy at birth, male (years)`,
    mobile_subs            = `Mobile cellular subscriptions (per 100 people)`,
    own_account_fem        = `Own-account workers, female (% of female employment) (modeled ILO estimate
    own_account_male       = `Own-account workers, male (% of male employment) (modeled ILO estimate)`,
    own_account_total      = `Own-account workers, total (% of male employment) (modeled ILO estimate)`
    pop_0_14_pct           = `Population ages 0-14 (% of total)`,
    pop_0_14_total         = `Population ages 0-14, total`,
    pop_15_64_pct          = `Population ages 15-64 (% of total)`,
    pop_15_64_total        = `Population ages 15-64, total`,
    pop_65plus_pct         = `Population ages 65 and above (% of total)`,
    pop_65plus_total       = `Population ages 65 and above, total`,
    pop_density            = `Population density (people per sq. km of land area)`,
    population_growth             = `Population growth (annual %)`,
    pop_total              = `Population, total`,
    credit_coverage_priv   = `Private credit bureau coverage (% of adults)`,
    credit_coverage_pub    = `Public credit registry coverage (% of adults)`,
    rural_pop_total        = `Rural population`,
    rural_pop_pct          = `Rural population (% of total population)`,
    self_emp_fem           = `Self-employed, female (% of female employment) (modeled ILO estimate)`,
    self_emp_male          = `Self-employed, male (% of male employment) (modeled ILO estimate)`,
    self_emp_total         = `Self-employed, total (% of total employment) (modeled ILO estimate)`,
    tax_payments       = `Tax payments (number)`,
    telephone_lines        = `Telephone lines (per 100 people)`,
    contract_days          = `Time required to enforce a contract (days)`,
    start_business_days    = `Time required to start a business (days)`,
    tax_prep_hours         = `Time to prepare and pay taxes (hours)`,
    unemployment_fem              = `Unemployment, female (% of female labor force) (modeled ILO estima
    unemployment_male             = `Unemployment, male (% of male labor force) (modeled ILO estimate)`
    unemployment_total            = `Unemployment, total (% of total labor force) (modeled ILO estimate
    unemp_youth_fem        = `Unemployment, youth female (% of female labor force ages 15-24) (modeled
    unemp_youth_male       = `Unemployment, youth male (% of male labor force ages 15-24) (modeled ILO
    unemp_youth_total      = `Unemployment, youth total (% of total labor force ages 15-24) (modeled IL
    urban_pop_total        = `Urban population`,
    urban_pct          = `Urban population (% of total)`,
    vulner_emp_fem         = `Vulnerable employment, female (% of female employment) (modeled ILO estima
    vulner_emp_male        = `Vulnerable employment, male (% of male employment) (modeled ILO estimate)
    vulner_emp_total       = `Vulnerable employment, total (% of total employment) (modeled ILO estimat
    waged_emp_fem          = `Wage and salaried workers, female (% of female employment) (modeled ILO e
    waged_emp_male         = `Wage and salaried workers, male (% of male employment) (modeled ILO estima
    waged_emp_total        = `Wage and salaried workers, total (% of total employment) (modeled ILO est
    net_trade              = `Net trade in goods and services (BoP, current US$)`)

# 1.1 Select variables used in models
model_df_linear <- df %>%
  dplyr::select(country,gdp_growth,gdp_pc_ppp,net_trade,unemployment_total,age_dependency,urban_pct, li
# 1.2 Basic checks
colSums(is.na(df)) %>% sort(decreasing = TRUE) # no missing values
sum(duplicated(model_df_linear$country))
# 1.3 Check for outlieres
summary(model_df_linear)
boxplot(model_df_linear$net_trade, main="Net trade")

# 2. Descriptive Statistics
# Summary stats for linear model dataset
  summary_stats <- psych::describe(model_df_linear %>% dplyr::select(-country))
  summary_stats %>% dplyr::select(mean, sd, min, max, skew, kurtosis)
# Correlation Heatmap
cor_matrix <- cor(model_df_linear %>% dplyr::select(-country), use = "pairwise.complete.obs")
corrplot(cor_matrix, method = "color", type = "upper", tl.col = "black", tl.srt = 45,
         addCoef.col = "black", number.cex = 0.6,
         title = "Correlation Heatmap of Predictors & GDP Growth", mar=c(0,0,1,0))
# Scatterplots: GDP growth vs each predictor
long_df <- model_df_linear %>%
  pivot_longer(-c(country, gdp_growth), names_to = "variable", values_to = "value")
ggplot(long_df, aes(x = value, y = gdp_growth)) +
  geom_point(color = "steelblue", alpha = 0.7) +
```

```r
  geom_smooth(method = "lm", se = FALSE, color = "red", linewidth = 0.7) +
  facet_wrap(~variable, scales = "free_x") +
  theme_minimal() +
  labs(title = "GDP Growth vs Predictors",
       x = "Predictor Value", y = "GDP Growth (%)")

# Q1: Linear Regression Prediction
# Model A: baseline regression without Net trade
formula_A <- gdp_growth ~ gdp_pc_ppp + unemployment_total + age_dependency + urban_pct +
  life_expectancy_female + tax_payments + population_growth
model_A <- lm(formula_A, data = model_df_linear, singular.ok = FALSE)
# Model B: regression with Net Trade
formula_B <- gdp_growth ~ gdp_pc_ppp + unemployment_total + age_dependency + urban_pct +
  life_expectancy_female + tax_payments + net_trade + population_growth
model_B <- lm(formula_B, data = model_df_linear, singular.ok = FALSE)
summary(model_A)
summary(model_B)

## Task 2:
formula_C <- gdp_growth~gdp_pc_ppp+unemployment_total+age_dependency+I(unemployment_total^2)
 +urban_pct+life_expectancy_female+population_growth+tax_payments
model_C <- lm(formula_C, data = model_df_linear, singular.ok = FALSE)
summary(model_C)
## model diagnostics
plot(model_A)
vif(model_A)
plot(model_C)
vif(model_C)

## Task 3
X <- model.matrix(formula_C, data = model_df_linear)[, -1]
y <- model_df_linear$gdp_growth
set.seed(555)
cvfit <- cv.glmnet(x=X, y=y, alpha=1, type.measure = "mse", nfolds = 10)
print(cvfit)
coef(cvfit, s = "lambda.min")
coef(cvfit, s = "lambda.1se")

## Task 4
set.seed(555)
cvfit1 <- cv.glmnet(x=X, y=y, alpha=1, type.measure = "mse", nfolds = 10, standardize=TRUE)
cvfit2 <- cv.glmnet(x=X, y=y, alpha=1, type.measure = "mse", nfolds = 10, standardize=FALSE)
# Coefficients of both models
coef(cvfit2, s = "lambda.min")
coef(cvfit1, s = "lambda.min")

## Task 6: Train/Test split (110/40)
set.seed(555)
n <- nrow(X)
idx <- sample(seq_len(n), size = 110)
trainData <- df[idx,]
y2<-df$gdp_growth
X2 <- model.matrix(gdp_growth ~ . -country, data=df)[,-1]
```

```r
X_train <- X2[idx, ]; y_train <- y2[idx]
X_test <- X2[-idx,]; y_test <- y2[-idx]

## Task 7
## a) Ridge CV on training set
set.seed(555)
cv_ridge <- cv.glmnet(X_train, y_train, alpha = 0, nfolds = 10)
lambda_ridge_min <- cv_ridge$lambda.min
lambda_ridge_1se <- cv_ridge$lambda.1se
### Final ridge models at the two lambdas
ridge_min <- glmnet(X_train, y_train, alpha = 0, lambda = lambda_ridge_min)
ridge_1se <- glmnet(X_train, y_train, alpha = 0, lambda = lambda_ridge_1se)

## Elastic Net
set.seed(555)
# Cross-validation setup
fitControl <- trainControl(method = "repeatedcv",number = 10,repeats = 5,verboseIter = TRUE)
alpha_grid <- seq(0.0001, 1, length = 10)
lambda_seq <- cv_ridge$glmnet.fit$lambda
elasticNet <- train(gdp_growth~.-country,data=trainData,method = "glmnet",
  tuneGrid = expand.grid(alpha=alpha_grid,lambda=lambda_seq),trControl=fitControl)
print(elasticNet$bestTune)
best_alpha  <- elasticNet$bestTune$alpha
best_lambda <- elasticNet$bestTune$lambda
en_final <- glmnet(X_train, y_train,alpha = best_alpha,lambda = best_lambda,standardize = TRUE)
coef(en_final)

## Task 8
# Predictions for Ridge models
pred_ridge_min<-predict(ridge_min, newx = X_test)
pred_ridge_1se<-predict(ridge_1se, newx = X_test)
# Elastic Net models: lambda.min and lambda.1se
en_min<-glmnet(X_train,y_train,alpha=best_alpha,lambda= elasticNet$bestTune$lambda,standardize=TRUE)
# For lambda.1se, pick the largest lambda within 1 SE of min error
results_all <- elasticNet$results
min_rmse <- min(results_all$RMSE)
rmse_1se <- min_rmse + results_all$RMSESD[which.min(results_all$RMSE)]
lambda_en_1se <- max(results_all$lambda[results_all$RMSE <= rmse_1se])
en_1se <- glmnet(X_train, y_train,alpha = best_alpha,lambda = lambda_en_1se,standardize = TRUE)
# Predictions for Elastic Net models
pred_en_min  <- predict(en_min, newx = X_test)
pred_en_1se  <- predict(en_1se, newx = X_test)
# Performance metric functions
rmse <- function(y, yhat) sqrt(mean((y - as.numeric(yhat))^2))
mae  <- function(y, yhat) mean(abs(y - as.numeric(yhat)))
# Collect results in one table
results <- tibble(
  model = c("Ridge (lambda.min)","Ridge (lambda.1se)",
    paste0("Elastic Net (alpha=", round(best_alpha, 2), ", lambda.min)"),
    paste0("Elastic Net (alpha=", round(best_alpha, 2), ", lambda.1se)")),
  RMSE = c(rmse(y_test, pred_ridge_min),rmse(y_test, pred_ridge_1se),
           rmse(y_test,pred_en_min),rmse(y_test, pred_en_1se)),
  MAE = c(mae(y_test, pred_ridge_min),mae(y_test, pred_ridge_1se),
          mae(y_test,pred_en_min),mae(y_test, pred_en_1se)))
print(results)

## Task 11-12
df <- df %>%
  mutate(Growing_more = ifelse(gdp_growth > 2.7, 1, 0))
table(df$Growing_more)
X3 <- model.matrix(Growing_more ~ . - country, data=df)[,-1]
y3 <- df$Growing_more
set.seed(555)
n <- nrow(X3)
idx1 <- sample(seq_len(n), size = 110)  # training set
X_train1 <- X3[idx1, ]; y_train1 <- y3[idx1]
X_test1  <- X3[-idx1,]; y_test1  <- y3[-idx1]

set.seed(555)
cv_ridge_logit <- cv.glmnet(X_train1, y_train1, alpha = 0, family = "binomial", nfolds = 10)
lambda_min <- cv_ridge_logit$lambda.min
lambda_1se <- cv_ridge_logit$lambda.1se
ridge_logit_min <- glmnet(X_train1,y_train1,alpha=0,lambda=lambda_min,family="binomial",nfold=10)
ridge_logit_1se <- glmnet(X_train1,y_train1,alpha=0,lambda=lambda_1se,family="binomial",nfold=10)
# Probabilities
prob_min  <- predict(ridge_logit_min, newx = X_test1, type = "response")
prob_1se  <- predict(ridge_logit_1se, newx = X_test1, type = "response")
# Class predictions (threshold 0.5)
pred_min <- ifelse(prob_min > 0.5, 1, 0)
pred_1se <- ifelse(prob_1se > 0.5, 1, 0)
# Confusion matrices
table(Predicted = pred_min, Actual = y_test1)
mean(pred_min == y_test1)
table(Predicted = pred_1se, Actual = y_test1)
mean(pred_1se == y_test1)

## Task 13
set.seed(555)
n <- nrow(X3)
idx2 <- sample(seq_len(n), size = 100)
X_train2 <- X3[idx2, ]; y_train2 <- y3[idx2]
X_test2  <- X3[-idx2,]; y_test2  <- y3[-idx2]
# Cross-validated logistic Ridge
cv_ridge_logit2 <- cv.glmnet(X_train2, y_train2,alpha = 0,family = "binomial",nfolds = 10)
lambda_min2 <- cv_ridge_logit2$lambda.min
lambda_1se2 <- cv_ridge_logit2$lambda.1se
ridge_logit_min2 <- glmnet(X_train2,y_train2,alpha=0,lambda=lambda_min2,family="binomial")
ridge_logit_1se2 <- glmnet(X_train2,y_train2,alpha=0,lambda=lambda_1se2,family="binomial")
# Predictions on new test set
prob_min2 <- predict(ridge_logit_min2, newx = X_test2, type = "response")
prob_1se2 <- predict(ridge_logit_1se2, newx = X_test2, type = "response")
pred_min2 <- ifelse(prob_min2 > 0.5, 1, 0)
pred_1se2 <- ifelse(prob_1se2 > 0.5, 1, 0)
# Accuracy of different train-test split
mean(pred_min2 == y_test2)
mean(pred_1se2 == y_test2)
```