EBDS20102 - Supervised Machine Learning - *Assignment 1*

Explaining Air Quality with Economic and Environmental Factors

Mathijs de Jong 380891
Chao Liang 588140
Yuchou Peng 481487
Eva Mynott 495602

1 Introduction

Air pollution, both indoor and outdoor, is considered as an invisible killer which has contributed to the deaths of around 500,000 newborns in 2019, according to the report of State of Global Air 2020[1]. Poor air quality could lead to a destructive impact on the global environment, human health, social property, industrial and agricultural production in both the short term and long term. In recent times, reducing air pollution has become a pressing issue that receives a lot of media and academic attention. It is therefore essential to investigate the factors that potentially influence air quality. This paper uses a cross-sectional data set on air quality in 30 U.S. regions, to examine the relationship between air quality and various regional economic and environmental factors. These factors include the value added of companies, average income, amount of rain, and whether the region is located at the coast. Our research question is

To what extend can air quality be explained by economic and environmental factors?

In order to answer this question, we use the better subset regression algorithm that uses the MM-algorithm. This paper is structured as follows. Section 2 presents a brief overview of the data. Section 3 contains a detailed outline of the methodology. Empirical results are reported and interpreted in Section 4. Finally, Section 5 presents the conclusion.

2 Data

The data set in this study is obtained from the Ecdat R package[2] and contains information on air quality in Californian Metropolitan areas in the United States. It is a cross-sectional data set from 1972 and consists of 30 regional observations. It contains altogether six variables; one response variable which is the variable of interest and the remaining variables are predictors. First, the response variable air quality index (**airq**) is an integer that ranges from 0 to 500, where a lower index number indicates better air quality. To model air quality, we use five predictors. The value added of companies in thousands of US dollars (**vala**), the amount of rain in inches (**rain**) in the data, an indicator whether the area is a coastal area (**coas**), the population density per square mile (**dens**), and the average income per head in US dollars (**medi**) rounded to dollars (Verbeek, 2004).

Table 1 presents summary statistics on the six variables in the data set. The total number of observations in the data set is 30. First, the air quality index has an overall average of 104.7 with a relatively low standard deviation of 28. The region with the worst air quality has an index of 165, whereas the region with the best air quality has an index of 59. Second, company value added in thousands of US dollars has a mean of 4188.5 thousands of US dollars, where the region with the most economic activity has a value added of 19733.8 thousands of US dollars. Third, the amount of rain in inches is on average 36.1, where the region with the least rain had 12.6 inches and the region with the most rain had 68.1 inches of rain. Fourth, the indicator variable coastal area has a mean of 0.7 and a standard deviation of 0.5. Hence, 21 regions are coastal areas whereas nine are not located at the coast. Fifth, the population density per square mile is 1728.6 on average with large variation: a standard deviation of 2827.8. Finally, the average income per head in US dollars is 9476.7, with a minimum of 271.6 and a maximum of 12957.5 US dollars. Note that these dollar amounts should be interpreted as values in the year 1972.

[1] Health Effects Institute (2020). State of Global Air 2020. Special Report. URL: `https://www.stateofglobalair.org/`.

[2] Y. Croissant and S. Graves (2020). Data Sets for Econometrics. URL: `https://cran.r-project.org/web/packages/Ecdat/Ecdat.pdf`.

Table 1: Airq Data Summary

This table reports the mean, standard deviation, first and third quantile, minimum and maximum for the data set on regional air quality in California in 1972. The right section includes a correlation matrix between all six variables in the data set. The total numbers of observations is 30.

	Descriptive statistics							Correlation					
variable	n	mean	std. dev.	min	q_{25}	q_{75}	max	`airq`	`vala`	`rain`	`coas`	`dens`	`medi`
Air quality	30	104.7	28.0	59.0	81.0	126.2	165.0	1.000					
Value added	30	4188.5	4630.2	992.9	1535.8	4141.4	19733.8	0.325	1.000				
Rain	30	36.1	13.5	12.6	31.0	42.7	68.1	-0.026	-0.149	1.000			
Coast dummy	30	0.7	0.5	0.0	0.0	1.0	1.0	-0.490	0.010	0.185	1.000		
Population density	30	1728.6	2827.8	271.6	365.2	1635.2	12957.5	-0.039	0.158	0.009	0.005	1.000	
Medium income	30	9476.7	12499.0	853.0	3339.8	8715.0	59460.0	0.251	0.890	-0.086	0.170	0.195	1.000

The right section of Table 1 presents correlations between the six variables in the air quality data set. We observe that the correlation of both the amount of rain in inches (**rain**) and the population density per square mile (**dens**) with the air quality index (**airq**) are near zero. This indicates that these variables may not explain much of the regional variation in air quality. Using similar reasoning, company value added (**vala**), the coastal indicator (**coas**), and average income (**medi**) are likely to do a better job in explaining the air quality indicator. Based on the correlation sign, we would expect a positive relationship between company value added and air quality, as well as between average income and air quality. Furthermore, we expect air quality to improve (i.e. the indicator to decrease) if the region is a coastal area. Moreover, it should be noted that the value added of companies in thousands of US dollars (**vala**) is highly correlated with the average income per head in US dollars (**medi**). The correlation coefficient between these two variables is 0.89, which may indicate redundant predictors and the presence of multicollinearity. Hence, it might not be optimal to include both variables in the model at the same time.

3 Methodology

We consider a multiple linear regression model to investigate to what extend air quality can be explained by the other variables in our data set. Let n denote the number of observations and k the number of explanatory variables with index sets $\mathcal{I} = \{1, 2, \ldots, n\}$ and $\mathcal{J} = \{1, 2, \ldots, k\}$, respectively. The objective that we are trying to predict is called the dependent variable, denoted by $\mathbf{y} = (y_1, y_2, \ldots, y_n)^{'}$. In this paper, this is air quality. Its values are assumed to be independent and not identically distributed. The constant of the model together with the k explanatory variables are combined as column vectors in the $n \times (k+1)$ matrix $\mathbf{X}$. Hence, this matrix is given by $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^{'}$, where $\mathbf{x}_i = (1, x_{i,1}, \ldots, x_{i,k})^{'}$ are $(k+1)$-dimensional, possibly stochastic, with finite second moment, independent and not all identically distributed, for $i \in \mathcal{I}$. Following Cameron and Trivedi (2005), the linear regression model is then given by

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} = \begin{bmatrix} 1 & x_{1,1} & \cdots & x_{1,k} \\ 1 & x_{2,1} & \cdots & x_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \cdots & x_{n,k} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}, \tag{1}$$

where ε_i denotes the error term in the model for observation i, for $i \in \mathcal{I}$, and $\boldsymbol{\beta}$ are the parameters of the model. As β_0 is multiplied with the vector of ones, this is the intercept. The other parameters β_j denote the effect of x_{ij} on y_i, for $i \in \mathcal{I}$ and $j \in \mathcal{J}$.

To obtain an estimate for $\boldsymbol{\beta}$ in model (1), we assume that the true data generating process is $\mathbf{y} = \mathbf{X}\boldsymbol{\beta}_0 + \boldsymbol{\varepsilon}^*$, where $\boldsymbol{\beta}_0$ denotes the true parameter vector and $\boldsymbol{\varepsilon}^*$ are the corresponding realizations of the error terms. The objective now is to minimize the sum of squared residuals, that is,

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \, (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^{'} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = \mathbf{y}^{'}\mathbf{y} + \boldsymbol{\beta}^{'}\mathbf{X}^{'}\mathbf{X}\boldsymbol{\beta} - 2\boldsymbol{\beta}^{'}\mathbf{X}^{'}\mathbf{y}. \tag{2}$$

Under the additional identification assumption that $\mathbf{X}^{'}\mathbf{X}$ is full rank, optimization problem (2) has an analytical solution given by $\hat{\boldsymbol{\beta}}_{\text{OLS}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$. This estimator is unbiased and consistent under the assumption of zero mean errors conditional on the explanatory variables, that is, $\mathbb{E}\left[\boldsymbol{\varepsilon}|\mathbf{X}\right] = \mathbf{0}$ (Cameron and Trivedi, 2005).

An alternative method that can be used to estimate $\hat{\boldsymbol{\beta}}_{\text{OLS}}$ is the MM algorithm as shown in Groenen (2020). The idea behind the MM algorithm is to replace an objective function $f(x)$ by a simpler function, the majorizing function $g(x, y)$, and iteratively minimize the objective function with the majorizing function. This function should satisfy the following three conditions: 1) $f(y) = g(y, y)$ at some support point y; 2) $\forall x, y : f(x) \leq g(x, y)$; 3) $g(x, y)$ must be simpler than $f(x)$, usually linear or quadratic. For solving the linear regression model, the difficult term is $\boldsymbol{\beta}'\mathbf{X}'\mathbf{X}\boldsymbol{\beta}$ which can be replaced by $\lambda\boldsymbol{\beta}'\boldsymbol{\beta}$ where λ is the largest eigenvalue of $\mathbf{X}'\mathbf{X}$ (again requiring that $\mathbf{X}'\mathbf{X}$ is of full rank). For computational convenience, $\mathbf{y}$ and the k columns of the explanatory variables in $\mathbf{X}$ are usually standardized before applying this procedure (without the constant in $\mathbf{X}$) and afterwards the estimate $\hat{\boldsymbol{\beta}}_{\text{scaled}}$ is scaled back to the original data.

For statistical inference of the parameter estimates, we can use the (asymptotic) distribution of the estimator. Assuming heteroskedastic errors, that is, $\text{Var}\left(\varepsilon|\mathbf{X}\right) = \sigma^2\mathbf{I}_{k+1}$, and asymptotic normality of the matrix $n^{-1/2}\mathbf{X}'\boldsymbol{\varepsilon}$, that is, $n^{-1/2}\mathbf{X}'\boldsymbol{\varepsilon} \xrightarrow{d} \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{X}'\mathbf{X}\right)$, it follows that

$$\sqrt{n}\left(\hat{\boldsymbol{\beta}}_{OLS} - \boldsymbol{\beta}_0\right) \xrightarrow{d} \mathcal{N}\left(\mathbf{0}, \sigma^2\left(\mathbf{X}'\mathbf{X}\right)^{-1}\right). \tag{3}$$

To obtain a similar result for small samples, the assumption of asymptotic normality, is replaced by normality. The unknown value σ^2 in (3) can be replaced by its estimate $\hat{\sigma}^2 = \frac{(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})'(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})}{n-k-1}$. Hence, we can test for individual significance of the parameter estimate $\hat{\beta}_j$ using the univariate t-test, for $j \in \mathcal{J} \cup \{0\}$. The null hypothesis of this test is that $\beta_j = 0$ against the alternative that $\beta_j \neq 0$. The null hypothesis is rejected when $\hat{\beta}_j$ differs significantly from zero, that is, when $\left|\hat{t}_j\right| > c$, where the test statistic or t-value is given by $\hat{t}_j = \frac{\hat{\beta}_j\sqrt{n-k-1}}{\sqrt{\text{Var}\left(\hat{\boldsymbol{\beta}}_j\right)}} \sim t(n-k-1)$, where $\text{Var}\left(\hat{\boldsymbol{\beta}}_j\right) = \hat{\sigma}^2\left[\left(\mathbf{X}'\mathbf{X}\right)^{-1}\right]_{(j,j)}$ and c is a chosen critical value.

The fit of the regression model can be evaluated using the ratio between the explained variance and the total variance of $\mathbf{y}$. This statistic is called the multiple R^2 and is defined as the squared correlation between $\mathbf{y}^m$, the vector $\mathbf{y}$ demeaned, and $\hat{\mathbf{y}}^m = \mathbf{X}^m\hat{\boldsymbol{\beta}}$, where $\mathbf{X}^m$ is the matrix $\mathbf{X}$ demeaned column-wise. When the model (1) includes a constant, this statistic is between 0 and 1. Moreover, this statistic is increasing in the number of regressors. In order to evaluate the fit of models with different numbers of regressors, the adjusted R^2 by Theil (1961) can be used. When there is a constant in the model, this statistic is defined as

$$\text{Adjusted R}^2 = 1 - \left(1 - \text{R}^2\right) \times \left(\frac{n}{n-k-1}\right). \tag{4}$$

The scaling factor adjusts for the number of explanatory terms $(k+1)$ in the model relative to the number of observations n.

One method for determining which explanatory variables are important, is the best subset method. In this method, the regression model is estimated for all combinations of explanatory variables, and the combination that fits the data best according to some metric is chosen. Evaluating all subsets is NP-hard and hence inconvenient in practice. Fortunately, several approximate algorithms have been proposed. In this paper, we apply one of these approximation algorithms, namely, the better subset regression based on the MM algorithm proposed by Xiong (2014). For a given number of explanatory variables m, this algorithm is obtained by only selecting the m values with the largest absolute coefficient in each iteration of the MM algorithm for scaled multiple regression. The optimal number of explanatory variables can then be evaluated using some metric, we choose the in-sample adjusted R-squared as defined in equation (4). Note that this algorithm is an approximation and could converge to a local minimum. The code used for this study is given in Appendix A

4 Results

In this section the results of this study are discussed. Applying the better subset algorithm to the air quality data set, we find that it is optimal to select the coastal region dummy when allowing at most one regressor besides the constant. Based on the adjusted R-squared metric, including more regressors does not lead to an improvement in the predictive power of the model that is larger than one would expect by adding an unrelated variable. Table 2 presents the ordinary least squares regression results of the final model. The regression results have been validated by comparing it to the standard linear model package in R.

Table 2: Results better subset regression

This table reports the regression coefficients, standard errors, t-statistics, corresponding p-values, R^2 and adjusted R^2 for the better subset regression algorithm applied to air quality using all explanatory variables. Note that a constant is always included. *, ** and *** denote significance at the 5%, 1%, and 0.1% level, respectively.

Summary statistics			
Nr. regr.	5	R^2	0.240
Opt. nr. regr.	1	Adjusted R^2	0.213

Parameter estimates					
variable	coefficient	std. dev.	t-stat.	p-value	sign.
`Intercept`	125.314	8.288	15.121	0.000	***
`coas`	-29.449	9.906	-2.973	0.006	**

In Table 2, we observe that the model is able to explain approximately 24% of the total variation in air quality across Californian regions. Similarly, if we base our judgment on the adjusted R-squared, the model explains 21.3% of the variation in the air quality index. The intercept of the model is significant at the 0.1% level, with a t-statistic of 15.1. The constant indicates that, for a non-coastal area, the air quality index on a scale of 0-500 is on average 125.3 in this sample. Moreover, the estimated coefficient on the coastal area indicator variable, which equals one for a coastal area and zero for a non-coastal area, is significant at the 1% level. This means that non-coastal areas have an air quality index that is approximately 29.5 lower than the non-coastal areas on average.

5 Conclusion

In this study, we have analyzed the air quality of Californian regions using a cross-sectional data set from 1972. Employing a better subset regression MM-algorithm, we found that it is optimal to include only one regressor in an ordinary least squares regression model. This regressor is an indicator variable which equals one if the Californian region is a coastal area and zero otherwise. The regression results indicate a significant improvement of air quality in case of a coastal region: the air quality index decreases from 125.3 to 95.8 on average. All in all, we conclude that air quality is best explained by the location in terms of a coastal or non-coastal area. The effect of other regional variables including company value added, amount of rain, population density, and average income on air quality, is negligible in this sample. A limitation of the data set used in this study is its age: it is from 1972, and the small number of observations in the sample.

References

Cameron, A. C. and Trivedi, P. K. (2005). *Microeconometrics: methods and applications.* Cambridge university press.

Groenen, P. (2020). Supervised machine learning week 1.

Theil, H. (1961). Economic forecasts and policy.

Verbeek, M. (2004). *A Guide to Modern Econometrics.* John Wiley and Sons.

Xiong, S. (2014). Better subset regression. *Biometrika*, 101(1):71–84.

Appendix A R Code

The following function has been created for the better subset regression.

```r
better.subset.lm = function(X, y, m.vals, tol=1e-6, verbose=0) {
  # Implementation of the better subsets selection estimator by Xiong 2014.
  # Inputs:
  #   X:          Table containing explanatory variables.
  #   y:          Column, vector or list containing dependent variables.
  #   m.vals:     Vector containing number of regressors to select.
  #   tol:        Tolerated rounding error, default is 1e-6.
  #   verbose:    Integer indicating the step-size of printing iterations.
  # Output:
  #   Dataframe containing the results of the best beta parameter obtained
  #   through better subset regression.

  # Initiallize algorithm and define helper functions
  y = data.matrix(y); X = data.matrix(X)   ## Ensure data is numerical
  ## Transform data to scaled numeric so MM algorithm can be applied
  y.scale = scale(y); X.scale = scale(X)

  ## Define constants
  rss_tot = sum((y - mean(y)) ^ 2)
  Xt.X.scale = crossprod(X.scale)
  N = nrow(X); P = ncol(X)
  best.metric = Inf
  k = 0
  inv.lambda = 1 / eigen(Xt.X.scale)$values[1]
  inv.lambda.Xt.y.scale = inv.lambda * crossprod(X.scale, y.scale)

  ## Define helper functions for computing statistics, descaling and printing
  rss = function(beta) sum((y.scale - X.scale %*% beta) ^ 2)
  r2 = function(beta) 1 - sum((y - cbind(1, X) %*% beta) ^ 2) / rss_tot
  adj.r2 = function(beta, m) 1 - (1 - r2(beta)) * (N - 1) / (N - m - 1)
  descale.b = function(b) {
    b = sd(y) * b / apply(X, 2, sd)
    b = c(mean(y) - sum(colMeans(X) * b), b)
    names(b) = c('(Intercept)', colnames(X))
    return(b)
  }
  progress.line = function(var, val, var_width=21, width=15, nsmall=0)
    return(paste0(format(paste0(var, ':'), width=var_width), format(val,
      width=width, justify='right', nsmall=nsmall), '\n'))
  progress.str = function(m, k, rss.new, rss.old, delta) {
    paste0(progress.line('Number of regressors', m), progress.line('Iteration', k),
      progress.line('RSS.new', rss.new, nsmall=10), progress.line('RSS.old',
        rss.old, nsmall=10), progress.line('delta', delta, nsmall=10))
  }

  # Execute algorithm for all possible number of explanatory variables
  for (m in m.vals) {
    # Choose some inital beta_0
    b.new = rep(0, P); b.new[sample(P, m)] = runif(m)

    rss.new = rss(b.new)     # Compute RSS(beta_0)
    # Update beta_k until convergence
    while (TRUE) {
      # Update iteration
      k = k + 1;

      # Replace old parameters by previous
      b.old = b.new; rss.old = rss.new

      # Apply MM-step
      u = b.old - inv.lambda * (Xt.X.scale %*% b.old) + inv.lambda.Xt.y.scale

      # Evaluate parameter importance
      abs.u = abs(u); abs.u.max = sort(abs.u, decreasing=T)[m]

      # Update parameters
      b.new = ifelse(abs.u >= abs.u.max, u, 0); rss.new = rss(b.new)

      # Retrieve improvement
      delta = rss.old - rss.new

      # Display progress if verbose
      if (verbose & (k %% verbose == 0)) {
        cat(progress.str(m, k, rss.new, rss.old, delta), '\n\n')
      }

      # Break if improvement smaller than tol; sufficient convergence
      if (delta / rss.old < tol) break
    }

    # Ensure information of last iteration is displayed
    if (verbose & (k %% verbose != 0)) {
      cat(progress.str(m, k, rss.new, rss.old, delta), '\n\n')
    }

    # 'Descale' estimator to original data
    b = descale.b(b.new)

    # Update best estimate if better performance
    metric = adj.r2(b, m)
    if (metric < best.metric) {
      best.metric = metric; best.b = b
    }
  }

  # Generate summary statistics
  best.r2 = r2(best.b)
  ids = (best.b != 0)
  best.b = best.b[ids]
  X.full = cbind(1, X)[, ids]
  best.rss = sum((y - (X.full %*% best.b)) ^ 2)
  dof = (N - sum(ids))
  best.cov.b = solve(crossprod(X.full)) * best.rss / dof
  best.std.b = sqrt(diag(best.cov.b))
```

```
  # Compute t-values of best beta
  best.t.b = best.b / best.std.b

  # Return results best beta
  return(data.frame(
    'coefficients' = best.b,
    'standard error' = best.std.b,
    't-values' = best.t.b,
    'p-values' = 2 * pt(-abs(best.t.b), dof),
    'R^2' = best.r2,
    'adjusted R^2' = c(best.metric, rep(NULL, length(best.b) - 1))
  ))
}
```

The following code has been used to generate the results of this study.

```
################################################################################
# Load dependencies
################################################################################

# Install packages
if (!require('Ecdat')) install.packages('Ecdat', quiet=T)

# Load dependencies
source('better.subset.lm.R')

# Specify options
options(scipen=999)


################################################################################
# Generate results
################################################################################

# Load data
df = Ecdat::Airq

# Define dependent and independent variables
y = df$airq
X = subset(df, select = -c(airq))

# Transform yes/no variables to 1/0
X = apply(X, 2, function(x) {
  if (setequal(x, c('yes', 'no'))) return(ifelse(x == 'yes', 1L, 0L))
  return(as.numeric(x))
})

# Estimate and show results of better subset regression
better.subset.lm(X, y, m.vals=c(1:ncol(X)), verbose=0)
```