The Performance of Preliminary Model Selection Using AIC or BIC

Juliet Nwabuzor

Supervised by Assoc. Prof. Paul Kabaila

Department of Mathematical and Physical Sciences
La Trobe University
Victoria 3086, Australia

October 2022

Acknowledgements

Praise and Glory be to the almighty God who has made the completion of this Master's degree possible. Without you Lord, I would never have gone far. Your mercies indeed endure forever.

I would like to thank my husband Ajala Afolabi for been there through these times, giving me the love, support and courage needed to pursue this dream. You have indeed been a blessing to me and loved dearly. To my sons, Mika and Uriel Ajala, thank you both for understanding whilst I stay dedicated to completing this thesis. Love you both.

To my dearest family members, I would never have belonged to a better family than the one I am in now. Your love and support pulled me through. You all are the best.

To my supervisor, Assoc. Prof. Paul Kabaila; your immense support, constant attendance at meetings, extra time allocations and encouragement cannot be overlooked. Each appointment with you shaped this research. I am eternally grateful. Thank you and God bless you sir.

Contents

List of Tables

Chapter 1

Introduction

The aim of this thesis is to evaluate the performance of confidence intervals for a specified parameter of interest, after preliminary model selection is carried out by minimizing a criterion such as the Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC). However, before addressing this, it is of essence to focus on some important aspects that will aid in understanding the purpose of this thesis.

Regression analysis is a statistical technique applied for investigating how the probability distribution of a response variable (also called a dependent variable) is influenced by the explanatory variables (also called independent or predictor variables). It is utilized whilst determining inherent correlation between variables in a dataset and makes predictions using the relations. The application of regression analysis spans through many fields including business, mathematics and statistics, economics, and many more.

Linear regression analysis is the technique of choice for many because it provides the simplest form to model a regression function as a linear combination of predictors. This aids in the interpretation of model parameters specifically in small sample sizes; as it usually yields a satisfactory approximation to the regression function (Xiaogang *et al.* 2019). There are two kinds of linear regression analysis that can occur; which is strongly dependent on the number of regressors involved.

According to Lakshmi *et al.* (2021), assuming a linear relationship exists between a dependent variable Y and an independent variable X, such that the points align along a straight line; then the mathematical representation of the linear regression model is shown as:

$$Y_i = \beta_0 + \beta_1 X_i \text{ with } i = 1, 2, \ldots n, \tag{1.1}$$

where β_0 and β_1 represents the intercept term and the slope respectively.

They further stated that if equation (1.1) is viewed graphically, there is a likelihood of the data points not aligning accurately along the straight line. This implies

that equation (1.1) require a modification that takes account of this. Let the difference between Y and $(\beta_0 + \beta_1 X_i)$ be denoted by ε. Therefore ε is the statistical error describing the lack of ability to accurately fit the data using model 1.1. Therefore, with this error term included, yields the model:

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \text{ with } i = 1, 2, \ldots n. \tag{1.2}$$

Equation (1.2) is called a simple linear regression model, where X can be referred to as an explanatory or predictor or regressor variable, while Y is known as the dependent or response variable. As in equation (1.2), the presence of a single independent and a single dependent variable makes it a simple linear regression model. When a model comprises of a single dependent variable and several independent variable, the model is called a multiple linear regression model (Lakshmi *et al.* 2021).

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \ldots + \beta_k X_{ki} + \varepsilon_i \text{ with } i = 1, 2, \ldots n. \tag{1.3}$$

Equation (1.3) is an example of a multiple linear regression model with the dependent variable Y related to k number of regressors whilst the parameters $\beta_j, j = 0, 1, 2, \ldots, k$ are known as regression coefficients. Note that the parameter β_j defines the change in Y per unit change in X_j, when all other X variables are kept fixed.

Although the relationship between Y and X are unknown in a multiple linear regression model, they are often used as approximation models. This is due to the fact that over a range of the predictor variables, the linear statistical model becomes an adequate approximation to the true unknown function (Lakshmi *et al.* 2021). This is one purpose of regression analysis.

There are several estimation methods available for linear models, with the most common being the least squares estimation (LSE) method and the maximum likelihood estimation (MLE) method. The details of these estimation methods will not be addressed here. However in brief terms, the LSE method estimates parameters by minimizing the distance from the observed response to the predicted values whilst the MLE method estimates parameters of an assumed probability distribution for the responses. It does this by maximizing a likelihood function; so that under the assumed statistical model, the observed data is most likely (Xiaogang *et al.* 2012).

In Chapter 2 the concept of linear regression models in establishing relationships between variables (response and explanatory) is introduced. This section then discusses the relevance of variable selection in producing accurate estimates of coefficients and predicted values of the response variable.

In Chapter 3, analysis on case study I was performed with emphasis on the coverage probability measure of performance of subsequently - constructed confidence

interval (CI) using the information criterion AIC. The result and conclusion of this analysis is highlighted.

In Chapter 4, an outline of the techniques and procedural approach applied in case study II is discussed. Including its analysis; with emphasis on the coverage probability measure of performance of subsequently - constructed confidence interval (CI) using the information criterion BIC. In addition, the result and conclusion of the analysis is highlighted.

Chapter 2

Theoretical Background

2.1 Overview of Regression Models

Regression models form the fundamentals of many disciples; most especially the mathematically related discipline. Researchers in these fields frequently estimate a variety of statistical models using different kinds of dataset; however, the majority of these include regression models or their close relatives.

In chapter 1, we saw how the structure of a simple linear and multiple linear regression models looked like. It is worthy of note that the main essence of formulating a model is to explain the observed values of the dependent variable in relation to the explanatory variables. And in most cases, a linear model is one utilized in capturing this inherent relationship.

2.2 Computational Techniques for variable selection

According to Xiaogang & Xin (2009), model misspecification exists and occurs in numerous ways. There are several instances where the model underfits or overfits the data; with reasons due to either unintentionally excluding significant predictors or including irrelevant ones. Simple linear functions may be adequate or inadequate for some predictors in a dataset. For these predictors where its inadequate, they may require a more complex function for their analysis. As such, evaluating the adequacy of linearity for predictors cannot be overemphasized, and if inadequacy is discovered, applying appropriate functional forms that fit better is the key.

It is important to know that not all variables (regressors) of a dataset make it to form part of that model that is deemed best in a study. The actual set of explanatory variables utilized as part of the final regression model is determined by the analysis of the observed data itself. Determining this subset of variables have

been described by many as a variable selection problem (Edward, 2000).

There are certain objectives involved whilst sourcing for the subset of regressor variables. Amongst them are: the need to obtain a regression model that is complete at the same time realistic; and the need to include a small number of variables as much as possible. This is because the presence of insignificant regressor in a model often leads to decreases in the accuracy of the estimated coefficients as well as predicted values. As such, the goal of variable selection is to balance simplicity with fit (Edward, 2000).

With this in mind, a commonly used method for variable selection as discussed in Xiaogang & Xin (2009) is described in the next subsection.

2.2.1 All possible regressions Method

Also referred to as all subset regression; this algorithm fits all regressors; with the selection criterion recorded for each regressor. When this procedure is completed, the list for each subset size is determined. Then, the analyst can evaluate all model choices by attempting every possible combination of predictors and comparing each using a model selection criterion (Xiaogang & Xin, 2009).

Xiaogang & Xin (2009) stated that assuming there are k predictors in total; exclusive of the null model with the intercept term, this means there are a total of $2k - 1$ different combinations. Therefore if $k \geq 20$ for example, this means that $2k - 1 \geq 1048575$ possible distinct combinations is to be considered. This makes the all possible regressions method only effective for dataset involving fewer variables. According to the authors, a major assumption exist for this method; which is a lack of linear dependency between variables. This is because one of the models that must be solved includes all candidate variables (known as the full model). Implying if one variable is the sum of several variables, it shouldn't be included in the pool of candidates.

2.3 Model Selection Criteria

In real world applications, one is faced with the problem of model selection as there are numerous models to describe a population from a sample of observations. For nested models as in polynomial regression, the complexity of the model (parameter number) improves the model fit until it begins to deteriorate. Basically because the model increasingly shapes itself to the sample features rather than the true model that characterises the population (Linhart & Zucchini, 1986).

Model selection criteria is a useful tool for recognising a model with a suitable structure and dimension amongst many others. It assesses a fitted model for optimum balance between goodness-of-fit and parsimony. And when the balance exists in the chosen model, the model should effectively be able to predict new data arising from the same phenomenon. The usage of a selection criterion eliminates candidate models too simplistic to effectively accommodate the data or unnecessarily complex (Cavanaugh *et al.* 2019).

Knowing this, a question that resonates in the mind is which approximation or model is deemed best? Zucchini *et al.* (2011), says the answer to this strongly depends on the approach applied in the measurement of the fit quality.

There are several approaches to measuring the quality of fit and amongst them are: the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC).These approaches will be discussed extensively below.

2.3.1 Akaike Information Criterion (AIC)

The Akaike information criterion (AIC) introduced by Akaike (1974), was the first model selection criterion to gain relevance in the field of statistics, and more so, a widely used model selection tool (Cavanaugh *et al.* 2019).

The Akaike Information Criterion was birthed from the usage of Kullback-Leibler distance (Zucchini *et al.* 2011); with the equation shown below.

$$\mathrm{AIC}(M) = 2\log(L(\hat{\theta})) - 2p \tag{2.1}$$

where M is the model, L the likelihood, and $\hat{\theta}$ is the maximum likelihood estimator of the vector of the model's p parameters.

According to Akaike (1974), the first part of equation (2.1) measures the fit of the model to the observed sample; where the fit improves with a corresponding increase in the number of parameters. However, an improvement in the model fit to the sample doesn't correspond to an improvement in the fit to the population. The latter part is the penalty term which offsets for the model complexity. In literature, AIC is defined as minus the above expression. This means choosing the model that minimizes it (Zucchini *et al.* 2011). When a model selection criterion is applied, comparison between models can be carried out. In other words, AIC approach ensure a simultaneous achievement of model estimation and selection.

2.3.2 Bayesian Information Criterion (BIC)

The Bayesian approach of parameter estimation consider models available for selection as candidate models; each one with equal possibility of being the true model

(Schwarz, 1978). It is represented mathematically as:

$$\mathrm{BIC}(M) = 2\log(L(\hat{\theta})) - p\log(n) \tag{2.2}$$

where n denote the sample size and p the number of unknown parameters in the model (Schwarz, 1978). Using BIC, the model with the highest posterior probability is chosen.

Note that compared with AIC, BIC uses an approach that differ only in the penalty term. Also, AIC addresses the question of 'the best model' in terms of the least wrong while BIC addresses the question of 'which model is most likely to be true' (Zucchini *et al.* 2011).

2.3.3 Decision on selection criterion choice

Zucchini *et al.* (2011) stated that different selection criteria yields different model selections. As such, there isn't a direct answer to the question of 'which criterion is appropriate for use'?. In saying that, whilst some analysts use a single criterion, some consider the orderings indicated by two or more differing criteria (for example AIC and BIC) and others may tailor the criterion chosen to meet their research objectives. From which a choice is made based on the most reasonable, interpretable, or suitable in the context of their individual application.

.

Chapter 3

Case Study I

Simulations are a powerful method for the assessment of the performance of commonly-used tools of Data Science. One such tool is the preliminary data-based choice of a model by minimizing the Akaike Information Criterion (AIC) or the Bayesian Information Criterion(BIC).

Our focus is on the coverage probability performance of subsequently constructed confidence intervals. The assessment of this performance is made difficult because of the need to effectively consider all possible parameter values. However, a method that has been suggested for dealing with this difficulty is to randomly choose values of the parameters and then to use simulations to find the coverage probability **averaged** over these randomly-chosen parameter values. This section of thesis will assess the validity of this method.

In practice, a data scientist will use the results of existing software packages to obtain their results. To emulate this situation for the thesis project, the supervisor has provided the following `R` functions:
`parameter_estimates.R`
`confidence_interval_M1.R`
`confidence_interval_M2.R`
`choose_model_using_AIC.R`

To gain insight into this topic, I considered a 2-by-2 factorial design using r replicates. The model for the data resulting from this design involves the following unknown parameters: $\beta_0, \beta_1, \beta_2, \beta_{12}$ and σ^2, where σ^2 denotes the variance of the random error, so that $\sigma^2 > \sigma$.

For simplicity, only two possible models were considered.

1. The full model M_2 and
2. The simpler model M_1

The simpler model M_1 is the full model, but with β_{12} assumed to be 0.

Suppose our aim is to find a $1-\alpha$ confidence interval for $\theta = \beta_0 + \beta_1 + \beta_2 + \beta_{12}$. Then the structure of the model and the resulting parameter estimates is:

$$\text{2-by-2 factorial design with } r \text{ replicates (with parameters } \beta_0, \beta_1, \beta_2, \beta_{12} \text{ and } \sigma^2)$$
$$\downarrow$$
$$\text{Data}$$
$$\downarrow$$
$$\text{parameter estimates } \hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_{12} \text{ and } \hat{\sigma}^2$$

The above structure is given by executing the function named `parameter_estimates.R`

The `R` functions `confidence_interval_M1.R` and `confidence_interval_M2.R` compute the confidence intervals, with coverage $1-\alpha$, assuming model M1 and model M2, respectively, are true. We choose between the models M_2 and M_1 using the Akaike Information Criterion (AIC), based on the parameter estimates $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_{12}$ and $\hat{\sigma}^2$. This is done by the function named `choose_model_using_AIC.R`. After selecting the model by minimizing AIC, the confidence interval (CI) for θ with desired coverage probability $1-\alpha$, is obtained. Because the model is chosen first, the resulting confidence interval is called a post-model-selection CI.

What has been proved by Kabaila and co-authors is that, for each given value of the number of replicates r, the coverage probability of the post-model-selection CI is determined by the unknown parameter

$$\gamma = \frac{\beta_{12}}{\sigma/(2\sqrt{r})}.$$

The proof of this result is quite sophisticated and consequently does not form part of the thesis. As a consequence of this result, the coverage probability does not depend on β_0, β_1 and β_2. Such knowledge is usually not available in practice. In other words, in practice the Data Scientist would be concerned to vary over all of the values of the parameters β_0, β_1, β_2 and σ^2, for given r. However, for simplicity, we will assume that it is known that the coverage probability of the post-model-selection CI is determined by the unknown parameter γ, so that the minimum coverage probability of this CI can be found by simply setting $\sigma^2 = 1$ and β_0, β_1, β_2 to arbitrary values and then varying over β_{12} only.

To assess the coverage probability of the resulting CI, we use Monte Carlo Simulation. This simulation procedure provides:

1. The number of times out of `nsim` simulations that produces a CI that includes θ from which the natural estimate of the coverage probability is obtained and

2. The standard error of the estimate of the coverage probability which gives some idea of the difference between the estimate and the true value of the coverage probability of the CI based on the selected model.

This Monte Carlo simulation is implemented in the following `R` code.

```
# We consider choosing the model (either the full model
# M2 or the simpler model M1) by minimizing AIC and then
# using the confidence interval, with nominal coverage
# 1 - alpha, corresponding to this chosen model.

# The coverage probability of the resulting confidence
# interval is found using Monte Carlo simulation.

# Set the true parameter values
# Try beta12 set to 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7

beta0 <- 1
beta1 <- 2
beta2 <- 3
beta12 <- 0.7
sigmasq <- 1

r <- 8

# The desired converage probability of the CI is 1 - alpha
alpha <- 0.05

# The parameter of interest is theta
theta <- beta0 + beta1 + beta2 + beta12
theta

# nsim is the total number of simulation runs
nsim <- 1000000

# count is used to count how many of the simulation runs
```

```
# produce a confidence interval that includes theta. The
# initial value of count is 0
count <- 0

# The system.time(...) command reports that time taken
# for the computations ...


system.time(
for (i in c(1:nsim)){
simul.vec <- parameter_estimates(beta0, beta1, beta2,
beta12, sigmasq, r)
beta0.hat <- simul.vec[1]
beta1.hat <- simul.vec[2]
beta2.hat <- simul.vec[3]
beta12.hat <- simul.vec[4]
sigmasq.hat <- simul.vec[5]

# Choose the model that minimizes AIC
# which is specified by its number (either 1 or 2)
model.number <-
choose_model_using_AIC(beta12.hat, sigmasq.hat, r)


# If model M2 is chosen then the CI is constructed
# assuming that the model M2 is true
if (model.number == 2){
ci.vec <-
confidence_interval_M2(beta0.hat, beta1.hat, beta2.hat,
beta12.hat, sigmasq.hat, r, alpha)
#
# If model M1 is chosen then the CI is constructed
# assuming that the model M1 is true
}else{
ci.vec <-
confidence_interval_M1(beta0.hat, beta1.hat, beta2.hat,
beta12.hat, sigmasq.hat, r, alpha)
}

# If the confidence interval includes theta then
```

```
# increase count by 1.
if (theta >= ci.vec[1] && theta <= ci.vec[2]){
count <- count + 1
}

}
)

# CP.est is the natural estimate of the coverage
# probability of the confidence interval based
# on the model M2.
CP.est <- count / nsim
CP.est

# s.e.CP stands for the ‘‘standard error’’ of this
# estimate. This standard error gives us some idea
# of the difference between the estimate and the
# true value of the coverage probability of the
# confidence interval based on model M2.
s.e.CP <- sqrt(CP.est * (1 - CP.est) / nsim)
s.e.CP
```

3.1 Result of the first part of Case Study I

The result of the analysis of this case study is summarized in Table 3.1. It shows that for the post-model-selection CI, with nominal coverage 0.95, the minimum coverage probabilty estimate is approximately 0.916, which is achieved at $\beta_{12} = 0.3$. This emphasizes the usage of simulation methods in providing valid information about the coverage probability of the post-model-seleciton CI.

Table 3.1: Coverage probability estimate of the post-model-selection CI

β_{12}	CP.est	s.e.CP
0	0.93692	0.0007687712
0.1	0.9308	0.0008025669
0.2	0.92126	0.0008517042
0.3	0.91574	0.0008784091
0.4	0.92276	0.0008442392
0.5	0.93586	0.0007747649
0.6	0.94264	0.0007353219
0.7	0.94871	0.0006975624

3.2 Coverage probability of the post-model-selection CI averaged over randomly-chosen parameter values

Searching through parameter values to find the minimum coverage probabililty of the post-model-selection CI can require some considerable effort. Because of this, it has been suggested that one finds the coverage probability of this CI averaged over randomly-chosen values of the parameters.

Suppose that the parameter values $\beta_0, \beta_1, \beta_2, \beta_{12}$ and σ^2 are randomly chosen; and simulation used to find the coverage probability **averaged** over these randomly-chosen parameter values. How valid is this method?

For simplicity, we make use of the result proved by Kabaila and co-authors that, for each given value of r, the coverage probability of the post-model-selection CI is determined by the unknown parameter γ and that the coverage probability at $\gamma = c$ is the same as for $\gamma = -c$.

We now consider randomly choosing `nsim.beta12` values of β_{12} from a normal distribution with mean 0 and specified standard deviation `sd.rand.beta12` and for each of these randomly chosen values to use simulation to estimate the coverage probability for this randomly chosen value of β_{12}. We then average these estimated coverage probabilities to obtain an estimated average coverage probability.

The `R` code for carrying out this method is the following.

```
r <- 8
alpha <- 0.05

beta0 <- 0
beta1 <- 0
beta2 <- 0
sigmasq <- 1

n.sim.beta12 <- 200
# Consider the following values of
# sd.rand.beta12: 0.15, 0.1, 0.2, 0.3, 0.4, 0.6, 0.8, 1, 1.4, 2, 10
sd.rand.beta12 <- 0.05

nsim <- 20000

# Where 0 is the mean and n.sim.beta12 is the number of simulations of beta12
est.CP.vec <- rep(0, n.sim.beta12)
```

```
system.time(
for (i in c(1:n.sim.beta12)){
beta12 <- rnorm(1, 0, sd.rand.beta12)
temp <- cp_post_model_selection_aic_ci(beta0,
beta1, beta2, beta12, sigmasq, r, alpha, nsim)
est.CP.vec[i] <- temp[1]
}
)

average.est.CP <- mean(est.CP.vec)
average.est.CP
```

3.3 Result of the second part of Case Study I

The result of the analysis of the coverage probability estimate of the post-model-selection CI averaged over randomly-chosen parameter values is summarized in Table 3.2. It shows that the minimum coverage probability for the values of `sd.rand.beta12` is 0.927023; which is achieved at `sd.rand.beta12`=0.2.

Table 3.2: Coverage probability estimate of the post-model-selection CI averaged over randomly-chosen β_{12} parameter values

sd.rand.beta12	average.est.CP
0.05	0.9350373
0.1	0.931893
0.15	0.9290255
0.2	0.927023
0.3	0.928123
0.4	0.928849
0.6	0.933654
0.8	0.936966
1	0.938041
1.4	0.940961
2	0.945006
10	0.949126

3.4 Conclusion for Case Study I

In the context of a 2-by-2 factorial design with r replicates and model selection between only two models by minimizing AIC, we have found that the coverage probability of the post-model-selection CI averaged over randomly-chosen values

of the parameters is not as informative a measure of coverage performance as one would like. In the next chapter, we consider a similar question but in the context of a much more complicated model selection situation.

Chapter 4

Case study II

To critically assess and make logical inferences based on the research subject matter, a detailed procedural approach of the tools used, algorithm applied and a description of terms is considered. This section details all programs and practices employed in this research.

4.1 Methodology

4.1.1 Data source and description

This thesis used a dataset named 'cloud seeding data'; one first presented by Biondini *et al.* (1977). Who evaluated the effect of cloud seeding on rainfall. The data is divided into two (2) groups according to CAT 'echo motion category' with ($1 =$ *moving*, $2 =$ *stationary*). Each of these groups are further divided into three subgroups according to TRT (Treatment code) into ($1 =$ *seeded*, $2 =$ *random control*, $3 =$ *non-random control*). The variables are a combination of five predictor variables: c, p, s, n, v, their cross-product, and square terms. Kabaila (2005), further coded the variable TRT= 1 and TRT = 2 with 1 and 0 respectively.

The full (complete) 'cloud seeding data' therefore consists of a response variable; referred to as the floating target rainfall volume (FT) and twenty two (22) explanatory variables in the order TRT, $const, c, p, s, n, v, cp, cs, cn, cv, ps,$
$pn, pv, sn, sv, nv, c.sq, p.sq, s.sq, n.s,$ and $v.sq$.

Note that for the purpose of this thesis, the coefficients associated with these variables in similar order, will be defined as $\beta_1, \beta_2, \ldots \beta_{22}$. The variable of interest is TRT; associated with the coefficient β_1; where β_1 is the parameter of interest. β_1 measures the effect of cloud seeding on rainfall. In addition, the first and second columns of X are coded TRT and a column of ones (1)'s. The total sample size of the cloud seeding data is 33.

4.1.2 Software Application

For the statistical and analytical computation in this thesis, the programming language R-studio (version 2022.02.0+443) was utilized. This program allows for the installation of packages and the usage of inbuilt functions associated with them. This allows for the enhancement of the `R` language, whilst executing written codes accordingly.

A main package utilized is the Leaps() package. This was installed by selecting the install packages tab, indicating the package required, ensuring all dependency options are checked and clicking the install button. This package forms a vital part of this thesis as it enables the performance of an exhaustive search for the best subsets of the variables in the 'cloud seeding data' used in predicting FT. Embedded in the leaps package is the function regsubsets(); used in discoverying optimal subsets of predictors in the 'cloud seeding data' based on clearly defined criterion statistic.

Note that all packages are stored in libraries within `R`. As such, attaching these packages require the use of the library () command; with the package name as the argument.

4.1.3 Procedure

Data preprocessing was not carried out on the 'cloud seeeding data' as it was already in a clean state. To perform analysis on this data, the steps undertaken and the description of each step are as outlined below:

- Data importation - Done by clicking the file tab, browsing through the already set working directory, selecting the data named `cloud.seeding.full` and clicking the import button
- Fitted the full model using the data in `cloud.seeding.full` as follows. `lm(FT 0+., data = cloud.seeding.full`; which fits the linear model with all of the explanatory variables included but with the intercept term excluded. However, the column labelled $'const'$ already does the job of including an intercept term.
- Loaded the Leaps () package and performed model selection by minimizing BIC across all subsets of the coefficients of $c, p, s, n, v, cp, cs, cn, cv, ps, pn,$ $pv, sn, sv, nv, c.sq, p.sq, s.sq, n.sq, v.sq$. The columns TRT & *const* belonging to columns 1 & 2 are automatically included (forced in).
- With the best model `cloud.seeding.chosen` that includes only the chosen explanatory variables (depicted by TRUE) known; it was fitted and then the resulting post-model-selection CI was computed for θ.

- Simulated a vector of responses `responses.vec` with length 33, using same columns of data as `X.cloud.seeding`. `X.cloud.seeding` is a matrix containing same rows and columns (except column FT) as `cloud.seeding.full`; with 33 rows and 22 columns. Thereafter, the true values of the vector of regression coefficients `(beta.vec)`, with length 22; whose components are $\beta_1, \beta_2, \ldots, \beta_{22}$ was specified. Also, the population standard deviation `sd.rand.errors` of the random errors was specified. The vector `rand.errors` of random errors has length 33.

- Fitted the full model to the simulated data. Then placed `responses.vec` and `X.cloud.seeding` into a single matrix with 33 rows and 23 columns. Thereafter, obtained a CI for θ; which was obtained by fitting the full model to the simulated data. Then tested if θ is contained in this CI

- Carried out `nsimul` runs, where the simulated data is found independently for each of these runs, the CI for θ is computed from this simulated data and we count how many of these CI contain θ. From this run, the `estimate.cov.prob` is obtained

- Chose a model for the simulated data by minimizing BIC for 'best of all subsets' choice of explanatory variables and then computed the CI for θ, with nominal coverage 1 - α, using the chosen model. That computes the post-model-selection CI, with nominal coverage 1 - α

- Carried out `nsimul` simulation runs, where the simulated data is found independently for each of these runs. We count how many of these post-model-selection CI contain θ and then use this count to estimate the coverage probability of the post-model-selection CI.

- Computed a coverage probability averaged of randomly-chosen values of `sd.rand.errs` and the parameter values of $\beta_1, \beta_2, \ldots \beta_{22}$ and also keeping `sd.rand.errs` fixed at 1.

4.2 Overview of the full model of Case Study II

The full model of the "cloud seeding data", can be mathematically represented as:

$$\mathrm{FT}_i = \beta_1 \mathrm{TRT}_i + \beta_2 + \beta_3 c + \cdots + \beta_{22}\, v.sq_i + \varepsilon_i \text{ with } i = 1, 2, \ldots, n. \tag{4.1}$$

A summary of the parameter estimates for the full model is shown in Table 4.1. The estimate $\widehat{\theta}$ of the parameter of interest $\theta = \beta_1$, which is associated with the variable

TRT, is 1.5470. The standard 95% CI for θ is $[-0.3272445, 3.4213232]$. As the value of zero is not included within this interval, it implies that the effect of TRT on FT is not statistically significant. This CI is computed using the following `R` code.

```
# We calculate a 95% confidence interval for theta = beta1,
# which is the parameter associated with the column TRT,
# as follows.

summary.full.model <- summary(full.model)
est.theta.full <- summary.full.model$coefficients[1,1]
se.est.theta.full <- summary.full.model$coefficients[1,2]

# The acronymn df[2] refers to degree of freedom in column 2
df.full <- summary.full.model$df[2]
tquant.full <- qt(1 - alpha/2, df.full)

est.theta.full + c(-1, 1) * tquant.full * se.est.theta.full
```

Table 4.1: Parameter estimates for the full model

	Estimate	Std. Error	t value	Pr(>|t|)
TRT	1.5470	0.8516	1.82	0.0966
const	24.9358	14.3132	1.74	0.1093
c	1.2996	1.3549	0.96	0.3581
p	−36.0938	24.6945	−1.46	0.1718
s	1.1636	3.9297	0.30	0.7727
n	−0.0046	5.7067	−0.00	0.9994
v	−4.7989	2.3451	−2.05	0.0654
cp	0.8864	0.7201	1.23	0.2440
cs	−0.2037	0.3905	−0.52	0.6123
cn	0.1269	0.1753	0.72	0.4842
cv	−0.0174	0.0954	−0.18	0.8583
ps	−0.1804	5.1996	−0.03	0.9729
pn	−1.5628	2.7912	−0.56	0.5868
pv	2.9936	1.0688	2.80	0.0172
sn	0.2614	0.9599	0.27	0.7904
sv	0.1299	0.5854	0.22	0.8285
nv	−0.4599	0.3547	−1.30	0.2212
c.sq	−0.0346	0.0183	−1.89	0.0852
p.sq	9.3252	8.1699	1.14	0.2779
s.sq	−0.3111	0.9203	−0.34	0.7416
n.sq	2.0187	1.0027	2.01	0.0692
v.sq	0.1871	0.0818	2.29	0.0429

For given parameter values, the following R code uses Monte Carlo simulation to estimate the coverage probability of the post-model-selection CI, with nominal coverage $1-\alpha$, when the model is obtained by minimizing BIC for best of all subset choice of explanatory variables.

```
# Carry out nsimul simulation runs, where the simulated data
# is found independently for each of these runs.
#
# For each run, the model is chosen by minimizing
# BIC for "best of all subsets" choice of explanatory
# variables and then we compute the confidence interval for
# theta, with nominal coverage 1 - alpha, using the
# chosen model i.e. we compute the post-model-selection
# confidence interval, with nominal coverage 1 - alpha.
#
# We count how many of these post-model-selection
# confidence intervals contain theta and then use this
# count to estimate the coverage probability of the
# post-model-selection confidence interval.

alpha <- 0.05

sd.rand.errs <- 1

beta1 <- 0
beta2 <- 0.5
beta3 <- 1.1
beta4 <- -2.4
beta5 <- -0.3
beta6 <- 0.2
beta7 <- 0.1
beta8 <- 0.05
beta9 <- -0.01
beta10 <- -0.02
beta11 <- 0.1
beta12 <- 0.08
beta13 <- -0.6
beta14 <- 0.7
beta15 <- 0.01
```

```
beta16 <- -0.02
beta17 <- 0.005
beta18 <- -0.003
beta19 <- 0.007
beta20 <- -0.004
beta21 <- 0.0001
beta22 <- 0.0003

beta.vec <- c(beta1, beta2, beta3, beta4, beta5,
beta6,  beta7, beta8, beta9, beta10,
beta11, beta12, beta13, beta14, beta15,
beta16, beta17, beta18, beta19, beta20,
beta21, beta22)

theta <- beta.vec[1]

nsim <- 5000

counter <- 0
for (i in c(1:nsim)){

rand.errs.vec <- rnorm(n=33, mean=0, sd=sd.rand.errs)
responses.vec <- X.cloud.seeding %*% beta.vec + rand.errs.vec

temp <- regsubsets(x = X.cloud.seeding, y = responses.vec,
force.in = c(1,2), intercept = FALSE,
method = "exhaustive", nvmax = 20,
id = TRUE, vcov = TRUE)

criterion.summary <- summary(temp)
index.min <- which.min(criterion.summary$bic)
indices.vector <- which(criterion.summary$which[(index.min - 2),])

cloud.seeding.chosen <-
as.data.frame(cbind(responses.vec, X.cloud.seeding[,indices.vector]))

chosen.model <- lm(V1 ~ 0 + ., data = cloud.seeding.chosen)
summary.chosen.model <- summary(chosen.model)
```

```
est.theta.chosen <- summary.chosen.model$coefficients[1,1]

se.est.theta.chosen <- summary.chosen.model$coefficients[1,2]

df.chosen <- summary.chosen.model$df[2]

tquant.chosen <- qt(1 - alpha/2, df.chosen)

CI.simul.chosen <-
est.theta.chosen + c(-1, 1) * tquant.chosen * se.est.theta.chosen

if (theta >= CI.simul.chosen[1] && theta <= CI.simul.chosen[2]){
counter <- counter + 1
}

}
estimate.cov.prob.chosen <- counter / nsim
estimate.cov.prob.chosen
```

4.3 Result of the first part of Case Study II

When the above R code was run, the estimate of the coverage probability of the post-model-selection CI, with nominal coverage 0.95, is 0.8952, which took approximately 27 seconds to compute. To compute the minimum coverage probability of this CI would require search over all possible values of the parameters $\beta_1, \beta_2, \ldots \beta_{22}$ and σ, which is a search over a space of 23 dimensions. Without some additional computational techniques of the type described by Kabaila(2005), such a search is not computationally feasible. This is an example of a phenomenon known as the 'curse of dimensionality'. This motivates the method of coverage probability assessment described in the next section.

4.4 Coverage probability of the post-model-selection CI averaged of randomly-chosen parameter values and fixed standard deviation of the random errors in Case Study II

In this section we compute the coverage probability of the post-model-selection CI, averaged of randomly-chosen values of $\beta_1, \beta_2, \ldots \beta_{22}$. In this case study, σ is a fixed value equal to 1.

The R code for carrying out this method is the following.

```
est_aver_cov_prob_chosen <-
function(alpha, sd.randomly.chosen.beta, nsim.outer, nsim){
# This module computes the estimated coverage
# probability averaged of randomly-chosen values of
# beta1, beta2, ..., beta22. For simplicity,
# we suppose that sd.rand.errs <- 1.
#
# Inputs
# alpha: the nominal coverage probability is 1 - alpha
# sd.randomly.chosen.beta: beta1, beta2, ..., beta22
#    are independently normally distributed with mean 0
#    and standard deviation sd.randomly.chosen.beta
# nsim.outer: the number of observations of the random
#    vector (beta1, beta2, ..., beta22)
# nsim: the number of independent simulation runs for
#    each observed (beta1, beta2, ..., beta22)
#
# Output
# A vector with 2 components: the first component is the
# estimate of the coverage probability averaged over
# randomly chosen values of (beta1, beta2, ..., beta22)
# and the second component is the standard error of this
# estimate

sd.rand.errs <- 0.5

CP.est.vec <- rep(0, nsim.outer)
var.CP.est.vec <- rep(0, nsim.outer)
```

```
for (j in c(1:nsim.outer)){

beta.vec <- rnorm(22, mean = 0, sd = sd.randomly.chosen.beta)
theta <- beta.vec[1]

counter <- 0

for (i in c(1:nsim)){

rand.errs.vec <- rnorm(n=33, mean=0, sd=sd.rand.errs)
responses.vec <- X.cloud.seeding %*% beta.vec + rand.errs.vec

temp <- regsubsets(x = X.cloud.seeding, y = responses.vec,
force.in = c(1,2), intercept = FALSE,
method = "exhaustive", nvmax = 20,
id = TRUE, vcov = TRUE)

criterion.summary <- summary(temp)
index.min <- which.min(criterion.summary$bic)
indices.vector <- which(criterion.summary$which[(index.min - 2),])

cloud.seeding.chosen <-
as.data.frame(cbind(responses.vec, X.cloud.seeding[,indices.vector]))

chosen.model <- lm(V1 ~ 0 + ., data = cloud.seeding.chosen)
summary.chosen.model <- summary(chosen.model)

est.theta.chosen <- summary.chosen.model$coefficients[1,1]
se.est.theta.chosen <- summary.chosen.model$coefficients[1,2]
df.chosen <- summary.chosen.model$df[2]
tquant.chosen <- qt(1 - alpha/2, df.chosen)

CI.simul.chosen <-
est.theta.chosen + c(-1, 1) * tquant.chosen * se.est.theta.chosen

if (theta >= CI.simul.chosen[1] && theta <= CI.simul.chosen[2]){
counter <- counter + 1
```

```
}

CP.est <- counter / nsim

CP.est.vec[j] <- CP.est
var.CP.est.vec[j] <- CP.est * (1 - CP.est) / nsim

}

}

CP.av.est <- sum(CP.est.vec) / nsim.outer
CP.av.est.se <- sqrt(sum(var.CP.est.vec)) / nsim.outer

c(CP.av.est, CP.av.est.se)

}

alpha <- 0.05

sd.randomly.chosen.beta <- 0.1

nsim.outer <- 100
nsim <- 500

cat("alpha=", alpha, ",  sd.randomly.chosen.beta=",
sd.randomly.chosen.beta, ",  nsim.outer=", nsim.outer,
",  nsim=", nsim, "\n")

system.time(est.aver.cov.prob.chosen <-
est_aver_cov_prob_chosen(alpha, sd.randomly.chosen.beta, nsim.outer, nsim))

cat("Estimated average coverage probability is",
est.aver.cov.prob.chosen[1], "\n")
```

4.5 Result of the second part of Case Study II

The result of the analysis of the coverage probability estimate of the post-model-selection CI averaged over randomly-chosen parameter values is summarized in Table 4.2.

Table 4.2: Estimates of the coverage probability averaged over randomly-chosen parameter values

`sd.randomly.chosen.beta`	`est.aver.cov.prob.chosen`	standard error of estimates
0.1	0.88008	0.001450063
0.5	0.8717	0.001491183
1	0.87014	0.001498702
1.5	0.89214	0.001382567
2	0.89494	0.001366094
2.5	0.90198	0.001324876
3	0.90326	0.001315515
10	0.93058	0.001095981
20	0.94642	0.0009275702
40	0.95312	0.0006702943

Kabaila (2005) found that the minimum coverage probability of the post-model-selection CI is less than 0.7. However, irrespective of the value of `sd.randomly.chosen.beta` the coverage probability of this CI, averaged over randomly-chosen parameter values, was always substantially larger than 0.7, resulting in an unduly positive assessment of the coverage probability properties of this CI. Even for the smallest possible value of this average coverage probability, namely 0.87014 (achieved at `sd.randomly.chosen.beta` = 1) is substantially larger than 0.7.

4.6 Conclusion for Case Study II

In the context of the real-life cloud seeding data and model selection by minimizing BIC for "best of all subsets" choice of explanatory variables, we have found that the coverage probability of the post-model-selection CI averaged over randomly-chosen values of the parameters is not as informative a measure of coverage performance as one would like.

Bibliography

Akaike, H. (1974). A new look at statistical model identification. *IEEE Transactions on Automatic Control*, 19, 716 – 723.

Biondini, R., Simpson, J. & Woodley, W. (1977). Empirical predictors for natural and seeded rainfall in the Florida area cumulus experiment (FACE), 1970–1975. *Journal of Applied Meteorology*, 16, 585-–594.

Cavanaugh, J.E. & Neath, A. A. (2019). The Akaike information criterion: Background, derivation, properties, application, interpretation, and refinements. *Computational Statistics*, 11(3). https://doi.org/10.1002/wics.1460

Edward, I.G. (2000). The variable selection problem. *Journal of the American Statistical Association*, 95, 1304–1308.

Kabaila, P. (2005). On the coverage probability of confidence intervals in regression after variable selection. *Australian & New Zealand Journal of Statistics*, 47, 549—562.

Lakshmi, K., Mahaboob, B., Kumar, D.S., Prakash, G.B. & Rao, T.N. (2021). A new vision on ordinary least squares estimation of parameters of linear model. AIP Conference Proceedings, 2375(1). https://doi.org/10.1063/5.0066922

Linhart, H. & Zucchini, W. (1986). *Model Selection*. New York, Wiley Interscience

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6, 461-–464.

Zucchini, W., Claeskens, G. & Nguefack-Tsague, G. (2011). Model Selection. *Encyclopedia of Statistical Science*. ISBN : 978-3-642-04897-5

Xiaogang, S., Xin, Y. & Chih-Ling, T. (2012). Linear regression. *WIREs Computational Statistics*. DOI: 10.1002/wics.1198

Xin, Y & Xiaogang, S. (2009). Linear Regression Analysis: Theory and Computing. 348, https://doi.org/10.1142/6986